MF1236-01

EPSON

CMOS 32-BIT SINGLE CHIP MICROCOMPUTER

E0C33208/204/202 TECHNICAL MANUAL

E0C33208/204/202 Technical Hardware

E0C33 Family ASIC Macro Manual

ENERGY SAVING EPSON

SEIKO EPSON CORPORATION

E0C33208/204/202 Technical Manual

This manual describes the hardware specifications of the Seiko Epson original 32-bit microcomputers E0C33208, E0C33204 and E0C33202.

E0C33208/204/202 Technical Hardware

Describes the hardware specifications of the E0C33208/204/202 except for details of the peripheral circuits.

E0C33 Family ASIC Macro Manual

Describes details of all the peripheral circuit blocks for the E0C33 Family microcomputers.

Refer to the "E0C33000 Core CPU Manual" for details of the E0C33000 32-bit RISC CPU.

E0C33208/204/202
Technical Hardware

Table of Contents

1 Outline

The E0C33208/204/202 is a Seiko Epson original 32-bit microcomputer that features high speed, low power and low-voltage operation. It is designed for portable equipment that needs advanced data processing.
The E0C33208/204/202 consists of the E0C33000 32-bit RISC type CPU as the core, a bus control unit, a DMA controller, an interrupt controller, timers, serial interface circuits, an optional A/D converter, and RAM. It also includes a high-speed oscillation circuit, PLL and low-speed oscillation circuit allowing high-speed operation and low-power operation and a clock timer that provides excellent clock functions.
The E0C33208/204/202 also provides a DSP function, by using the internal MAC (multiplication and accumulation) operation function with the A/D converter, it makes it possible to design simply speech recognition and voice synthesis systems.

Each model has a different internal RAM size, package and data bus interface as shown in Table 1.1.

Table 1.1 Model Configuration

Model	Package	Internal RAM	Data bus I/F
E0C33208F0A	QFP5-128pin	8K bytes	TTL
E0C33204F0A	QFP5-128pin	4K bytes	TTL
E0C33202F0A	QFP5-128pin	2K bytes	TTL
E0C33208F0E	QFP5-128pin	8K bytes	CMOS/LVTTL
E0C33204F0E	QFP5-128pin	4K bytes	CMOS/LVTTL
E0C33202F0E	QFP5-128pin	2K bytes	CMOS/LVTTL
E0C33208F1E	QFP15-128pin	8K bytes	CMOS/LVTTL
E0C33204F1E	QFP15-128pin	4K bytes	CMOS/LVTTL
E0C33202F1E	QFP15-128pin	2K bytes	CMOS/LVTTL

The E0C3320xF0A that is configured with a TTL level data bus interface must be used only for 5 V data bus interface systems.

This manual explains the common functions using E0C33208/204/202 as the model name. Note that the independent functions available only for one or the other model (E0C33208, E0C33204 or E0C33202) are described with the specific model name.

1.1 Features

Core CPU

Seiko Epson original 32-bit RISC CPU E0C33000 built-in

- Basic instruction set: 105 instructions (16-bit fixed size)
- Sixteen 32-bit general-purpose register
- 32-bit ALU and 8-bit shifter
- Multiplication/division instructions and MAC (multiplication and accumulation) instruction are available
- 16.7 ns of minimum instruction execution time at 60 MHz operation

Internal memory

RAM: 8K bytes (E0C33208)
4K bytes (E0C33204)
2K bytes (E0C33202)

Internal peripheral circuits

Oscillation circuit:	High-speed (OSC3) oscillation circuit	33 MHz max.	
		Crystal/ceramic oscillator or external clock input	
	Low-speed (OSC1) oscillation circuit	32.768 kHz typ.	
		Crystal oscillator or external clock input	
Timers:	8-bit timer	4 channels	
	16-bit timer	6 channels	
	Watchdog timer	(16-bit timer 0's function)	
	Clock timer	1 channel (with alarm function)	
Serial interface:	2 channels (clock-synchronous system, asynchronous system and IrDA interface are selectable)		
A/D converter:	10 bits × 8 channels		
DMA controller:	High-speed DMA	4 channels	
	Intelligent DMA	128 channels	
Interrupt controller:	Possible to invoke DMA		
	Input interrupt	10 types (programmable)	
	DMA controller interrupt		5 types
	16-bit programmable timer interrupt		12 types
	8-bit programmable timer interrupt		4 types
	Serial interface interrupt		6 types
	A/D converter interrupt		1 type
	Clock timer interrupt		1 type
General-purpose input and output ports:	Shared with the I/O pins for internal peripheral circuits		
	Input port	13 bits	
	I/O port	29 bits	

External bus interface

BCU (bus control unit) built-in

- 24-bit address bus (internal 28-bit processing)
- 16-bit data bus
  Data size is selectable from 8 bits and 16 bits in each area.
- Little-endian memory access; big-endian may be set in each area.
- Memory mapped I/O
- Chip enable and wait control circuits built-in
- DRAM direct interface function built-in
  Supports fast page mode and EDO page mode.
  Supports self-refresh and CAS-before RAS refresh.
- Supports burst ROM.

Operating conditions and power consumption

Operating voltage:	Core (V_{DD})	1.8 V to 3.6 V
	I/O (V_{DDE})	1.8 V to 5.5 V
Operating clock frequency:	Max. 60 MHz (when core voltage = I/O voltage = 3.3 V ±0.3 V)	
Operating temperature:	-40 to 85°C	
Power consumption:	During SLEEP	4 µW typ.
	During HALT	100 mW typ.(3.3 V, 50 MHz)
	During execution	215 mW typ.(3.3 V, 50 MHz)

Note: The values of power consumption during execution were measured when a test program that consisted of 55% load instructions, 23% arithmetic operation instructions, 1% mac instruction, 12% branch instructions and 9% ext instruction was being continuously executed.

Supply form

QFP5-128pin or QFP15-128pin plastic package, or chip
(Note that a chip is available only for the E0C33208.)

1.2 Block Diagram

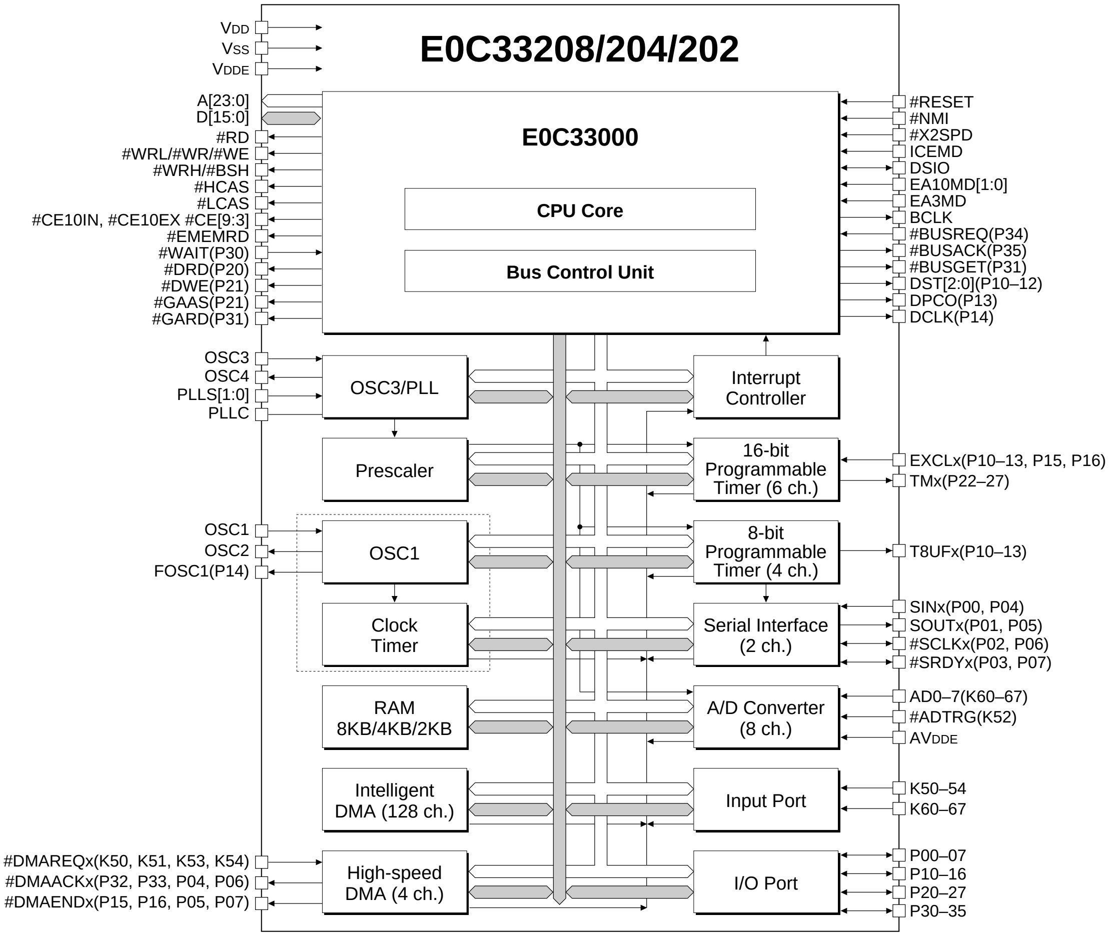
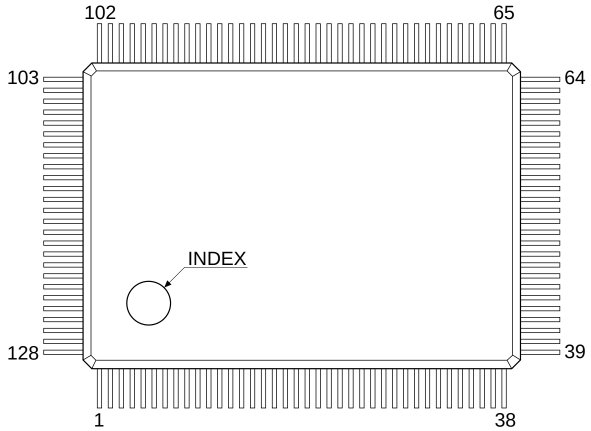


Figure 1.2.1 E0C33208/204/202 Block Diagram

1.3 Pin Description

1.3.1 Pin Layout Diagram (plastic package)

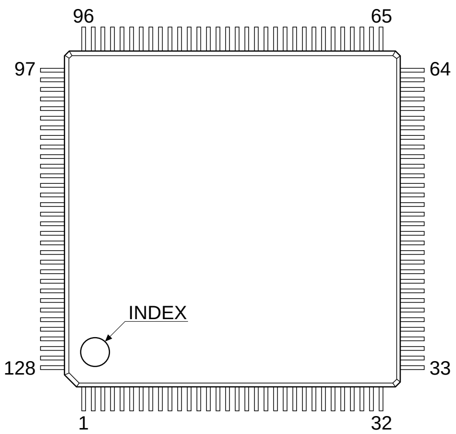
QFP5-128pin



No.	Pin name	No.	Pin name	No.	Pin name	No.	Pin name
1	P24/TM2	33	K65/AD5	65	#RESET	97	A16
2	V_{SS}	34	K50/#DMAREQ0	66	#NMI	98	ICEMD
3	P25/TM3	35	K64/AD4	67	A0/#BSL	99	A17
4	P26/TM4	36	K63/AD3	68	A1	100	A18
5	P15/EXCL4/#DMAEND0	37	K62/AD2	69	P34/#BUSREQ/#CE6	101	A19
6	P27/TM5	38	AV_{DDE}	70	V_{SS}	102	P04/SIN1/#DMAACK2
7	BCLK	39	K61/AD1	71	A2	103	P05/SOUT1/#DMAEND2
8	P00/SIN0	40	K60/AD0	72	A3	104	P06/#SCLK1/DMAACK3
9	P01/SOUT0	41	D6	73	A4	105	V_{SS}
10	D15	42	V_{SS}	74	A5	106	PLLC
11	V_{DD}	43	D5	75	A6	107	V_{SS}
12	P03/#SRDY0	44	D4	76	#CE10IN	108	PLLS1
13	D14	45	D3	77	V_{DD}	109	PLLS0
14	P31/#BUSGET/#GARD	46	D2	78	#EMEMRD	110	P07/#SRDY1/#DMAEND3
15	D13	47	D1	79	A7	111	#X2SPD
16	P32/#DMAACK0	48	D0	80	#HCAS	112	EA10MD0
17	D12	49	P35/#BUSACK	81	A8	113	EA10MD1
18	P33/#DMAACK1	50	V_{DDE}	82	#LCAS	114	V_{DD}
19	D11	51	#CE9/#CE17/#CE17&18	83	A9	115	EA3MD
20	K54/#DMAREQ3	52	OSC2	84	P16/EXCL5/#DMAEND1	116	OSC4
21	D10	53	#CE7/#RAS0/#CE13/#RAS2	85	A10	117	P20/#DRD
22	K53/#DMAREQ2	54	OSC1	86	A20	118	OSC3
23	D9	55	#CE6/#CE7&8	87	A11	119	P21/#DWE/#GAAS
24	K52/#ADTRG	56	#RD	88	A21	120	#CE3
25	V_{SS}	57	V_{SS}	89	A12	121	P22/TM0
26	K51/#DMAREQ1	58	#WRL/#WR/#WE	90	A22	122	P23/TM1
27	P02/#SCLK0	59	#WRH/#BSH	91	A13	123	DSIO
28	D8	60	#CE10EX/#CE9&10EX	92	A23	124	P10/EXCL0/T8UF0/DST0
29	D7	61	#CE8/#RAS1/#CE14/#RAS3	93	V_{SS}	125	P11/EXCL1/T8UF1/DST1
30	V_{DDE}	62	#CE5/#CE15/#CE15&16	94	A14	126	P12/EXCL2/T8UF2/DST2
31	K67/AD7	63	#CE4/#CE11/#CE11&12	95	A15	127	P13/EXCL3/T8UF3/DPCO
32	K66/AD6	64	P30/#WAIT/#CE4&5	96	V_{DDE}	128	P14/FOSC1/DCLK

Figure 1.3.1 Pin Layout Diagram (QFP5-128pin)

QFP15-128pin



No.	Pin name	No.	Pin name	No.	Pin name	No.	Pin name
1	P26/TM4	33	K63/AD3	65	A1	97	A18
2	P15/EXCL4/#DMAEND0	34	K62/AD2	66	P34/#BUSREQ/#CE6	98	A19
3	P27/TM5	35	AVDDE	67	VSS	99	P04/SIN1/#DMAACK2
4	BCLK	36	K61/AD1	68	A2	100	P05/SOUT1/#DMAEND2
5	P00/SIN0	37	K60/AD0	69	A3	101	P06/#SCLK1/DMAACK3
6	P01/SOUT0	38	D6	70	A4	102	VSS
7	D15	39	VSS	71	A5	103	PLLC
8	VDD	40	D5	72	A6	104	VSS
9	P03/#SRDY0	41	D4	73	#CE10IN	105	PLLS1
10	D14	42	D3	74	VDD	106	PLLS0
11	P31/#BUSGET/#GARD	43	D2	75	#EMEMRD	107	P07/#SRDY1/#DMAEND3
12	D13	44	D1	76	A7	108	#X2SPD
13	P32/#DMAACK0	45	D0	77	#HCAS	109	EA10MD0
14	D12	46	P35/#BUSACK	78	A8	110	EA10MD1
15	P33/#DMAACK1	47	VDDE	79	#LCAS	111	VDD
16	D11	48	#CE9/#CE17/#CE17&18	80	A9	112	EA3MD
17	K54/#DMAREQ3	49	OSC2	81	P16/EXCL5/#DMAEND1	113	OSC4
18	D10	50	#CE7/#RAS0/#CE13/#RAS2	82	A10	114	P20/#DRD
19	K53/#DMAREQ2	51	OSC1	83	A20	115	OSC3
20	D9	52	#CE6/#CE7&8	84	A11	116	P21/#DWE/#GAAS
21	K52/#ADTRG	53	#RD	85	A21	117	#CE3
22	VSS	54	VSS	86	A12	118	P22/TM0
23	K51/#DMAREQ1	55	#WRL/#WR/#WE	87	A22	119	P23/TM1
24	P02/#SCLK0	56	#WRH/#BSH	88	A13	120	DSIO
25	D8	57	#CE10EX/#CE9&10EX	89	A23	121	P10/EXCL0/T8UF0/DST0
26	D7	58	#CE8/#RAS1/#CE14/#RAS3	90	VSS	122	P11/EXCL1/T8UF1/DST1
27	VDDE	59	#CE5/#CE15/#CE15&16	91	A14	123	P12/EXCL2/T8UF2/DST2
28	K67/AD7	60	#CE4/#CE11/#CE11&12	92	A15	124	P13/EXCL3/T8UF3/DPCO
29	K66/AD6	61	P30/#WAIT/#CE4&5	93	VDDE	125	P14/FOSC1/DCLK
30	K65/AD5	62	#RESET	94	A16	126	P24/TM2
31	K50/#DMAREQ0	63	#NMI	95	ICEMD	127	VSS
32	K64/AD4	64	A0/#BSL	96	A17	128	P25/TM3

Figure 1.3.2 Pin Layout Diagram (QFP15-128pin)

1.3.2 Pin Functions

Table 1.3.1 List of Pins for Power Supply System

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
VDD	11,77,114	8,74,111	–	–	Power supply (+) for the internal logic
VSS	2,25,42,57, 70,93,105, 107	127,22,39, 54,67,90, 102,104	–	–	Power supply (-); GND
VDDE	30,50,96	27,47,93	–	–	Power supply (+) for the I/O block
AVDDE	38	35	–	–	Analog system power supply (+); AVDDE = VDDE

Table 1.3.2 List of Pins for External Bus Interface Signals

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
A0 #BSL	67	64	O	–	A0: Address bus (A0) when SBUSST(D3/0x4812E) = "0" (default) #BSL: Bus strobe (low byte) signal when SBUSST(D3/0x4812E) = "1"
A[23:1]	68,71–75, 79,81,83, 85–92,94, 95, 97, 99–101	65,68–72, 76,78,80, 82–89,91, 92,94, 96–98	O	–	Address bus (A1 to A23)
D[15:0]	10,13,15,17, 19,21,23,28, 29,41,43–48	7,10,12,14, 16,18,20,25, 26,38,40–45	I/O	–	Data bus (D0 to D15)
#CE10EX	60	57	O	–	Area 10 chip enable for external memory ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE9+#CE10EX signal.
#CE10IN	76	73	O	–	Area 10 chip enable for internal ROM emulation memory
#CE9 #CE17	51	48	O	–	#CE9: Area 9 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" (default) #CE17: Area 17 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE17+#CE18 signal.
#CE8 #RAS1 #CE14 #RAS3	61	58	O	–	#CE8: Area 8 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" and A8DRA(D8/0x48128) = "0" (default) #RAS1: Area 8 DRAM row strobe when CEFUNC[1:0](D[A:9])/0x48130) = "00" and A8DRA(D8/0x48128) = "1" #CE14: Area 14 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" or "1x" and A14DRA(D8/0x48122) = "0" #RAS3: Area 14 DRAM row strobe when CEFUNC[1:0](D[A:9])/0x48130) = "01" or "1x" and A14DRA(D8/0x48122) = "1"
#CE7 #RAS0 #CE13 #RAS2	53	50	O	–	#CE7: Area 7 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" and A7DRA(D7/0x48128) = "0" (default) #RAS0: Area 7 DRAM row strobe when CEFUNC[1:0](D[A:9])/0x48130) = "00" and A7DRA(D7/0x48128) = "1" #CE13: Area 13 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" or "1x" and A13DRA(D7/0x48122) = "0" #RAS2: Area 13 DRAM row strobe when CEFUNC[1:0](D[A:9])/0x48130) = "01" or "1x" and A13DRA(D7/0x48122) = "1"
#CE6	55	52	O	–	Area 6 chip enable ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE7+#CE8 signal.
#CE5 #CE15	62	59	O	–	#CE5: Area 5 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" (default) #CE15: Area 15 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01"
#CE4 #CE11	63	60	O	–	#CE4: Area 4 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" (default) #CE11: Area 11 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE11+#CE12 signal.
#CE3	120	117	O	–	Area 3 chip enable
#RD	56	53	O	–	Read signal
#EMEMRD	78	75	O	–	Read signal for internal ROM emulation memory

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
#WRL #WR #WE	58	55	O	–	#WRL: Write (low byte) signal when SBUSST(D3/0x4812E) = "0" (default) #WR: Write signal when SBUSST(D3/0x4812E) = "1" #WE: DRAM write signal (default)
#WRH #BSH	59	56	O	–	#WRH: Write (high byte) signal when SBUSST(D3/0x4812E) = "0" (default) #BSH: Bus strobe (high byte) signal when SBUSST(D3/0x4812E) = "1"
#HCAS	80	77	O	–	#HCAS: DRAM column address strobe (high byte) signal
#LCAS	82	79	O	–	#LCAS: DRAM column address strobe (low byte) signal
BCLK	7	4	O	–	Bus clock output
P34 #BUSREQ #CE6	69	66	I/O	–	P34: I/O port when CFP34(D4/0x402DC) = "0" (default) #BUSREQ: Bus release request input when CFP34(D4/0x402DC) = "1" #CE6: Area 6 chip enable when CFP34(D4/0x402DC) = "1" and IOC34(D4/0x402DE) = "1"
P35 #BUSACK	49	46	I/O	–	P35: I/O port when CFP35(D5/0x402DC) = "0" (default) #BUSACK: Bus acknowledge output when CFP35(D5/0x402DC) = "1"
P30 #WAIT #CE4&5	64	61	I/O	–	P30: I/O port when CFP30(D0/0x402DC) = "0" (default) #WAIT: Wait cycle request input when CFP30(D0/0x402DC) = "1" #CE4&5: Areas 4&5 chip enable when CFP30(D0/0x402DC) = "1" and IOC30(D0/0x402DE) = "1"
P20 #DRD	117	114	I/O	–	P20: I/O port when CFP20(D0/0x402D8) = "0" (default) #DRD: DRAM read signal output for successive RAS mode when CFP20(D0/0x402D8) = "1"
P21 #DWE #GAAS	119	116	I/O		P21: I/O port when CFP21(D1/0x402D8) = "0" and CFEX2(D2/0x402DF) = "0" (default) #DWE: DRAM write signal output for successive RAS mode when CFP21(D1/0x402D8) = "1" and CFEX2(D2/0x402DF) = "0" #GAAS: Area address strobe output for GA when CFEX2(D2/0x402DF) = "1"
P31 #BUSGET #GARD	14	11	I/O		P31: I/O port when CFP31(D1/0x402DC) = "0" and CFEX3(D3/0x402DF) = "0" (default) #BUSGET: Bus status monitor signal output for bus request when CFP31(D1/0x402DC) = "1" and CFEX3(D3/0x402DF) = "0" #GARD: Area read signal output for GA when CFEX3(D3/0x402DF) = "1"
EA10MD1	113	110	I	With pull-up	Area 10 boot mode selection EA10MD1 EA10MD0 Mode 1 1 External ROM mode
EA10MD0	112	109	I	–	1 0 Internal ROM mode 0 1 OTP mode 0 0 Internal ROM emulation
EA3MD	115	112	I	With pull-up	Area 3 mode selection 1: Internal ROM mode, 0: Emulation mode

Table 1.3.3 List of Pins for HSDMA Control Signals

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
K50 #DMAREQ0	34	31	I	With pull-up	K50: Input port when CFK50(D0/0x402C0) = "0" (default) #DMAREQ0: HSDMA Ch. 0 request input when CFK50(D0/0x402C0) = "1"
K51 #DMAREQ1	26	23	I	With pull-up	K51: Input port when CFK51(D1/0x402C0) = "0" (default) #DMAREQ1: HSDMA Ch. 1 request input when CFK51(D1/0x402C0) = "1"
K53 #DMAREQ2	22	19	I	With pull-up	K53: Input port when CFK53(D3/0x402C0) = "0" (default) #DMAREQ2: HSDMA Ch. 2 request input when CFK53(D3/0x402C0) = "1"
K54 #DMAREQ3	20	17	I	With pull-up	K54: Input port when CFK54(D4/0x402C0) = "0" (default) #DMAREQ3: HSDMA Ch. 3 request input when CFK54(D4/0x402C0) = "1"
P32 #DMAACK0	16	13	I/O	–	P32: I/O port when CFP32(D2/0x402DC) = "0" (default) #DMAACK0: HSDMA Ch. 0 acknowledge output when CFP32(D2/0x402DC) = "1"

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
P33 #DMAACK1	18	15	I/O	–	P33: I/O port when CFP33(D3/0x402DC) = "0" (default) #DMAACK1: HSDMA Ch. 1 acknowledge output when CFP33(D3/0x402DC) = "1"
P04 SIN1 #DMAACK2	102	99	I/O	–	P04: I/O port when CFP04(D4/0x402D0) = "0" and CFEX4(D4/0x402DF) = "0" (default) SIN1: Serial I/F Ch. 1 data input when CFP04(D4/0x402D0) = "1" and CFEX4(D4/0x402DF) = "0" #DMAACK2: HSDMA Ch. 2 acknowledge output when CFEX4(D4/0x402DF) = "1"
P06 #SCLK1 #DMAACK3	104	101	I/O	–	P06: I/O port when CFP06(D6/0x402D0) = "0" and CFEX6(D6/0x402DF) = "0" (default) #SCLK1: Serial I/F Ch. 1 clock input/output when CFP06(D6/0x402D0) = "1" and CFEX6(D6/0x402DF) = "0" #DMAACK3: HSDMA Ch. 3 acknowledge output when CFEX6(D6/0x402DF) = "1"
P15 EXCL4 #DMAEND0	5	2	I/O	–	P15: I/O port when CFP15(D5/0x402D4) = "0" (default) EXCL4: 16-bit timer 4 event counter input when CFP15(D5/0x402D4) = "1" and IOC15(D5/0x402D6) = "0" #DMAEND0: HSDMA Ch. 0 end-of-transfer signal output when CFP15(D5/0x402D4) = "1" and IOC15(D5/0x402D6) = "1"
P16 EXCL5 #DMAEND1	84	81	I/O	–	P16: I/O port when CFP16(D6/0x402D4) = "0" (default) EXCL5: 16-bit timer 5 event counter input when CFP16(D6/0x402D4) = "1" and IOC16(D6/0x402D6) = "0" #DMAEND1: HSDMA Ch. 1 end-of-transfer signal output when CFP16(D6/0x402D4) = "1" and IOC16(D6/0x402D6) = "1"
P05 SOUT1 #DMAEND2	103	100	I/O	–	P05: I/O port when CFP05(D5/0x402D0) = "0" and CFEX5(D5/0x402DF) = "0" (default) SOUT1: Serial I/F Ch. 1 data output when CFP05(D5/0x402D0) = "1" and CFEX5(D5/0x402DF) = "0" #DMAEND2: HSDMA Ch. 2 end-of-transfer signal output when CFEX5(D5/0x402DF) = "1"
P07 #SRDY1 #DMAEND3	110	107	I/O	–	P07: I/O port when CFP07(D7/0x402D0) = "0" and CFEX7(D7/0x402DF) = "0" (default) #SRDY1: Serial I/F Ch. 1 ready signal output when CFP07(D7/0x402D0) = "1" and CFEX5(D5/0x402DF) = "0" #DMAEND3: HSDMA Ch. 3 end-of-transfer signal output when CFEX7(D7/0x402DF) = "1"

Table 1.3.4 List of Pins for Internal Peripheral Circuits

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
K52 #ADTRG	24	21	I	With pull-up	K52: Input port when CFK52(D2/0x402C0) = "0" (default) #ADTRG: A/D converter trigger input when CFK52(D2/0x402C0) = "1"
K60 AD0	40	37	I	–	K60: Input port when CFK60(D0/0x402C3) = "0" (default) AD0: A/D converter Ch. 0 input when CFK60(D0/0x402C3) = "1"
K61 AD1	39	36	I	–	K61: Input port when CFK61(D1/0x402C3) = "0" (default) AD1: A/D converter Ch. 1 input when CFK61(D1/0x402C3) = "1"
K62 AD2	37	34	I	–	K62: Input port when CFK62(D2/0x402C3) = "0" (default) AD2: A/D converter Ch. 2 input when CFK62(D2/0x402C3) = "1"
K63 AD3	36	33	I	–	K63: Input port when CFK63(D3/0x402C3) = "0" (default) AD3: A/D converter Ch. 3 input when CFK63(D3/0x402C3) = "1"
K64 AD4	35	32	I	–	K64: Input port when CFK64(D4/0x402C3) = "0" (default) AD4: A/D converter Ch. 4 input when CFK64(D4/0x402C3) = "1"
K65 AD5	33	30	I	–	K65: Input port when CFK65(D5/0x402C3) = "0" (default) AD5: A/D converter Ch. 5 input when CFK65(D5/0x402C3) = "1"
K66 AD6	32	29	I	–	K66: Input port when CFK66(D6/0x402C3) = "0" (default) AD6: A/D converter Ch. 6 input when CFK60(D6/0x402C3) = "1"
K67 AD7	31	28	I	–	K67: Input port when CFK67(D7/0x402C3) = "0" (default) AD7: A/D converter Ch. 7 input when CFK67(D7/0x402C3) = "1"

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
P00 SIN0	8	5	I/O	–	P00: I/O port when CFP00(D0/0x402D0) = "0" (default) SIN0: Serial I/F Ch. 0 data input when CFP00(D0/0x402D0) = "1"
P01 SOUT0	9	6	I/O	–	P01: I/O port when CFP01(D1/0x402D0) = "0" (default) SOUT0: Serial I/F Ch. 0 data output when CFP01(D1/0x402D0) = "1"
P02 #SCLK0	27	24	I/O	–	P02: I/O port when CFP02(D2/0x402D0) = "0" (default) #SCLK0: Serial I/F Ch. 0 clock input/output when CFP02(D2/0x402D0) = "1"
P03 #SRDY0	12	9	I/O	–	P03: I/O port when CFP03(D3/0x402D0) = "0" (default) #SRDY0: Serial I/F Ch. 0 ready signal output when CFP03(D3/0x402D0) = "1"
P04 SIN1 #DMAACK2	102	99	I/O	–	P04: I/O port when CFP04(D4/0x402D0) = "0" and CFEX4(D4/0x402DF) = "0" (default) SIN1: Serial I/F Ch. 1 data input when CFP04(D4/0x402D0) = "1" and CFEX4(D4/0x402DF) = "0" #DMAACK2: HSDMA Ch. 2 acknowledge output when CFEX4(D4/0x402DF) = "1"
P05 SOUT1 #DMAEND2	103	100	I/O	–	P05: I/O port when CFP05(D5/0x402D0) = "0" and CFEX5(D5/0x402DF) = "0" (default) SOUT1: Serial I/F Ch. 1 data output when CFP05(D5/0x402D0) = "1" and CFEX5(D5/0x402DF) = "0" #DMAEND2: HSDMA Ch. 2 end-of-transfer signal output when CFEX5(D5/0x402DF) = "1"
P06 #SCLK1 #DMAACK3	104	101	I/O	–	P06: I/O port when CFP06(D6/0x402D0) = "0" and CFEX6(D6/0x402DF) = "0" (default) #SCLK1: Serial I/F Ch. 1 clock input/output when CFP06(D6/0x402D0) = "1" and CFEX6(D6/0x402DF) = "0" #DMAACK3: HSDMA Ch. 3 acknowledge output when CFEX6(D6/0x402DF) = "1"
P07 #SRDY1 #DMAEND3	110	107	I/O	–	P07: I/O port when CFP07(D7/0x402D0) = "0" and CFEX7(D7/0x402DF) = "0" (default) #SRDY1: Serial I/F Ch. 1 ready signal output when CFP07(D7/0x402D0) = "1" and CFEX5(D5/0x402DF) = "0" #DMAEND3: HSDMA Ch. 3 end-of-transfer signal output when CFEX7(D7/0x402DF) = "1"
P10 EXCL0 T8UF0 DST0	124	121	I/O	–	P10: I/O port when CFP10(D0/0x402D4) = "0" and CFEX1(D1/0x402DF) = "0" EXCL0: 16-bit timer 0 event counter input when CFP10(D0/0x402D4) = "1", IOC10(D0/0x402D6) = "0" and CFEX1(D1/0x402DF) = "0" T8UF0: 8-bit timer 0 output when CFP10(D0/0x402D4) = "1", IOC10(D0/0x402D6) = "1" and CFEX1(D1/0x402DF) = "0" DST0: DST0 signal output when CFEX1(D1/0x402DF) = "1" (default)
P11 EXCL1 T8UF1 DST1	125	122	I/O	–	P11: I/O port when CFP11(D1/0x402D4) = "0" and CFEX1(D1/0x402DF) = "0" EXCL1: 16-bit timer 1 event counter input when CFP11(D1/0x402D4) = "1", IOC11(D1/0x402D6) = "0" and CFEX1(D1/0x402DF) = "0" T8UF1: 8-bit timer 1 output when CFP11(D1/0x402D4) = "1", IOC11(D1/0x402D6) = "1" and CFEX1(D1/0x402DF) = "0" DST1: DST1 signal output when CFEX1(D1/0x402DF) = "1" (default)
P12 EXCL2 T8UF2 DST2	126	123	I/O	–	P12: I/O port when CFP12(D2/0x402D4) = "0" and CFEX0(D0/0x402DF) = "0" EXCL2: 16-bit timer 2 event counter input when CFP12(D2/0x402D4) = "1", IOC12(D2/0x402D6) = "0" and CFEX0(D0/0x402DF) = "0" T8UF2: 8-bit timer 2 output when CFP12(D2/0x402D4) = "1", IOC12(D2/0x402D6) = "1" and CFEX0(D0/0x402DF) = "0" DST2: DST2 signal output when CFEX0(D0/0x402DF) = "1" (default)

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
P13 EXCL3 T8UF3 DPCO	127	124	I/O	–	P13: I/O port when CFP13(D3/0x402D4) = "0" and CFEX1(D1/0x402DF) = "0" EXCL3: 16-bit timer 3 event counter input when CFP13(D3/0x402D4) = "1", IOC13(D3/0x402D6) = "0" and CFEX1(D1/0x402DF) = "0" T8UF3: 8-bit timer 3 output when CFP13(D3/0x402D4) = "1", IOC13(D3/0x402D6) = "1" and CFEX1(D1/0x402DF) = "0" DPCO: DPCO signal output when CFEX1(D1/0x402DF) = "1" (default)
P14 FOSC1 DCLK	128	125	I/O	–	P14: I/O port when CFP14(D4/0x402D4) = "0" and CFEX0(D0/0x402DF) = "0" FOSC1: OSC1 clock output when CFP14(D4/0x402D4) = "1" and CFEX0(D0/0x402DF) = "0" DCLK: DCLK signal output when CFEX0(D0/0x402DF) = "1" (default)
P15 EXCL4 #DMAEND0	5	2	I/O	–	P15: I/O port when CFP15(D5/0x402D4) = "0" (default) EXCL4: 16-bit timer 4 event counter input when CFP15(D5/0x402D4) = "1" and IOC15(D5/0x402D6) = "0" #DMAEND0: HSDMA Ch. 0 end-of-transfer signal output when CFP15(D5/0x402D4) = "1" and IOC15(D5/0x402D6) = "1"
P16 EXCL5 #DMAEND1	84	81	I/O	–	P16: I/O port when CFP16(D6/0x402D4) = "0" (default) EXCL5: 16-bit timer 5 event counter input when CFP16(D6/0x402D4) = "1" and IOC16(D6/0x402D6) = "0" #DMAEND1: HSDMA Ch. 1 end-of-transfer signal output when CFP16(D6/0x402D4) = "1" and IOC16(D6/0x402D6) = "1"
P22 TM0	121	118	I/O	–	P22: I/O port when CFP22(D2/0x402D8) = "0" (default) TM0: 16-bit timer 0 output when CFP22(D2/0x402D8) = "1"
P23 TM1	122	119	I/O	–	P23: I/O port when CFP23(D3/0x402D8) = "0" (default) TM1: 16-bit timer 1 output when CFP23(D3/0x402D8) = "1"
P24 TM2	1	126	I/O	–	P24: I/O port when CFP24(D4/0x402D8) = "0" (default) TM2: 16-bit timer 2 output when CFP24(D4/0x402D8) = "1"
P25 TM3	3	128	I/O	–	P25: I/O port when CFP25(D5/0x402D8) = "0" (default) TM3: 16-bit timer 3 output when CFP25(D5/0x402D8) = "1"
P26 TM4	4	1	I/O	–	P26: I/O port when CFP26(D6/0x402D8) = "0" (default) TM4: 16-bit timer 4 output when CFP26(D6/0x402D8) = "1"
P27 TM5	6	3	I/O	–	P27: I/O port when CFP27(D7/0x402D8) = "0" (default) TM5: 16-bit timer 5 output when CFP27(D7/0x402D8) = "1"

Table 1.3.5 List of Pins for Clock Generator

Pin name	Pin No.		I/O	Pull-up	Function
	QFP5-128	QFP15-128			
OSC1	54	51	I	–	Low-speed (OSC1) oscillation input (32 kHz crystal oscillator or external clock input)
OSC2	52	49	O	–	Low-speed (OSC1) oscillation output
OSC3	118	115	I	–	High-speed (OSC3) oscillation input (crystal/ceramic oscillator or external clock input)
OSC4	116	113	O	–	High-speed (OSC3) oscillation output
PLLS[1:0]	108,109	105,106	I	–	PLL set-up pins PLLS1 / PLLS0 / fin (f_{OSC3}) / fout (f_{PSCIN}) 1 / 1 / 10–30MHz / 20–60MHz ∗1 10–25MHz / 20–50MHz ∗2 0 / 1 / 10–15MHz / 40–60MHz ∗1 10–12.5MHz / 40–50MHz ∗2 0 / 0 / PLL is not used / L ∗1: ROM-less model with 3.3 V ± 0.3 V operating voltage ∗2: ROM built-in model, or 3.0 V ± 0.3 V operating voltage
PLLC	106	103	–	–	Capacitor connecting pin for PLL

Table 1.3.6 List of Other Pins

Pin name	Pin No.		I/O	Pull-down Pull-up	Function
	QFP5-128	QFP15-128			
ICEMD	98	95	I	With pull-down	High-impedance control input pin When this pin is set to High, all the output pins go into high-impedance state. This makes it possible to disable the E0C33 chip on the board.
DSIO	123	120	I/O	With Pull-up	Serial I/O pin for debugging This pin is used to communicate with the debugging tool ICD33.
#X2SPD	111	108	I	–	Clock doubling mode set-up pin 1: CPU clock = bus clock x 1, 0: CPU clock = bus clock x 2
#NMI	66	63	I	With Pull-up	NMI request input pin
#RESET	65	62	I	With Pull-up	Initial reset input pin

Note: "#" in the pin names indicates that the signal is low active.

2 Power Supply

This chapter explains the operating voltage of the E0C33208/204/202.

2.1 Power Supply Pins

The E0C33208/204/202 has the power supply pins shown in Table 2.1.1.

Table 2.1.1 Power Supply Pins

Pin name	Pin No.		Function
	QFP5-128pin	QFP15-128pin	
V_{DD}	11,77,114	8,74,111	Power supply (+) for the internal logic
V_{SS}	2,25,42,57,70,93,105,107	22,39,54,67,90,102,104,127	Power supply (-); GND
V_{DDE}	30,50,96	27,47,93	Power supply (+) for the I/O block
AV_{DDE}	38	35	Analog system power supply (+); $AV_{DDE} = V_{DDE}$

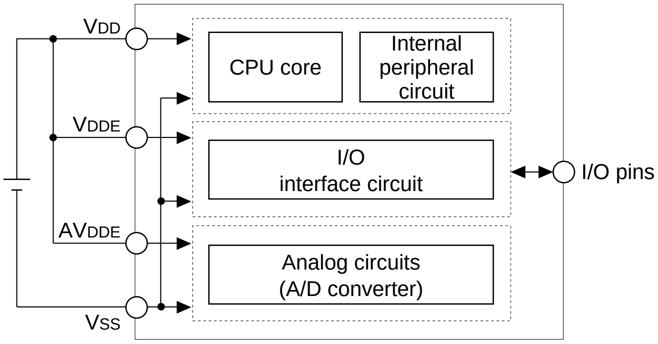


Figure 2.1.1 Power Supply System

2.2 Operating Voltage (V_{DD})

The core CPU and internal peripheral circuits operate with a voltage supplied between the V_{DD} and V_{SS} pins. The following operating voltage can be used:

V_{DD} = 1.8 V to 3.6 V (V_{SS} = GND)

Note: The E0C33208/204/202 has three V_{DD} pins and eight V_{SS} pins. Be sure to supply the operating voltage to all the pins. Do not open any of them.

The operating clock frequency range (OSC3) is 5 MHz to 50 MHz with this voltage.

2.3 Power Supply for I/O Interface (V_{DDE})

The V_{DDE} voltage is used for interfacing with external I/O signals. For the I/O interface of the E0C33208/204/202, the V_{DDE} voltage is used as high level and the V_{SS} voltage as low level.
Normally, supply the same voltage level as V_{DD}. It can be supplied separately from V_{DD} for 5 V interface. The V_{SS} pin is used for the ground common with V_{DD}.
The following voltage is enabled for V_{DDE}:

V_{DDE} = 1.8 V to 5.5 V (V_{SS} = GND)

Notes:
- The E0C33208/204/202 has three V_{DDE} pins. Be sure to supply a voltage to all the pins. Do not open any of them.
- When an external clock is input to the OSC1 or OSC3 pin, the clock signal level must be V_{DD}.
- The interface voltage level of the DSIO, P10, P11, P12, P13 and P14 pins is V_{DD}.

2.4 Power Supply for Analog Circuits (AV_{DDE})

The analog power supply pin (AV_{DDE}) is provided separately from the V_{DD} pin in order that the digital circuits do not affect the analog circuit (A/D converter). The AV_{DDE} pin is used to supply an analog power voltage and the V_{SS} pin is used as the analog ground.
Supply the same voltage level as the V_{DDE} to the AV_{DDE} pin.

$AV_{DDE} = V_{DDE}$ (V_{SS} = GND)

Note: Be sure to supply V_{DDE} to the AV_{DDE} pin even if the analog circuit is not used.

Noise on the analog power lines decrease the A/D converting precision, so use a stabilized power supply and make the board pattern with consideration given to that.

3 Internal Memory

This chapter explains the internal memory configuration.

3.1 RAM

The E0C33208/204/202 has a built-in RAM. Table 3.1.1 shows the size and mapping address.

Table 3.1.1 Built-in RAM

Model	Size	Address
E0C33208	8K bytes	0x0–0x1FFF
E0C33204	4K bytes	0x0–0x0FFF
E0C33202	2K bytes	0x0–0x07FF

The internal RAM is a 32-bit sized device and data can be read/written in 1 cycle regardless of data size (byte, half-word or word).
Figure 3.1.1 shows the basic internal memory map.

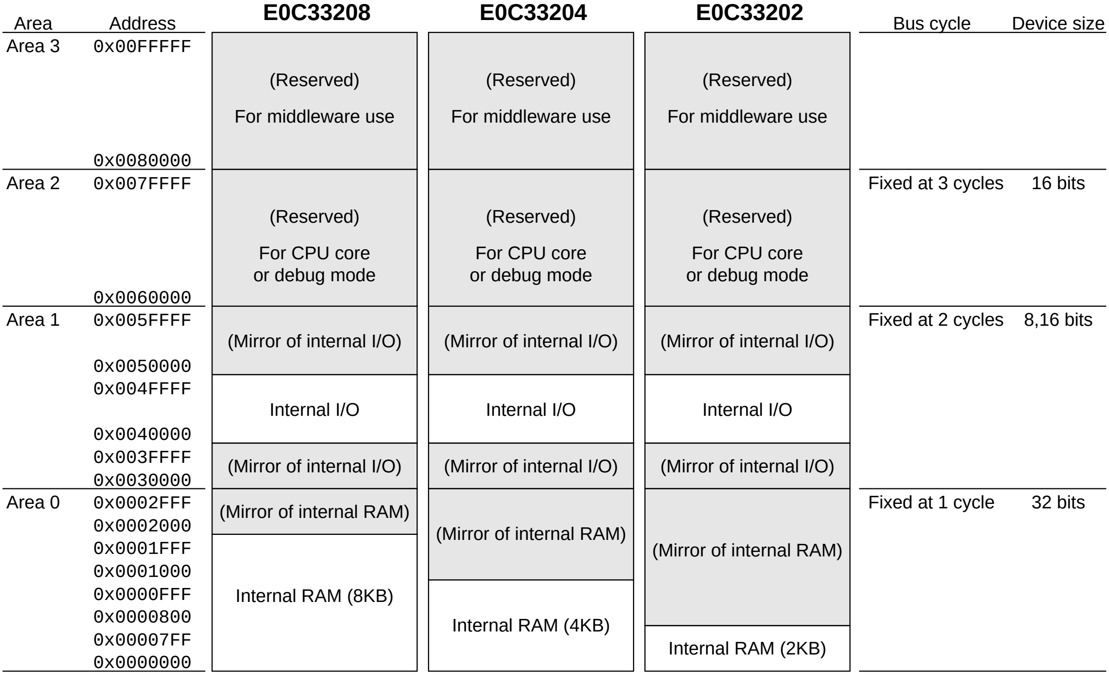


Figure 3.1.1 Basic Internal Memory Map

Area 2 is used in debug mode only and it cannot be accessed in user mode (normal program execution status).

3.2 ROM and Boot Address

The E0C33208/204/202 has no built-in ROM. Since the boot address is fixed at 0x0C00000 in the E0C33208/204/202, an external ROM/Flash must be placed from the top of Area 10.
For setting up Area 10, refer to the "BCU (Bus Controller Unit)" section in the "E0C33 Family ASIC Macro Manual".

4 Peripheral Circuits

This chapter lists the built-in peripheral circuits and the I/O memory map. For details of the circuits, refer to the "E0C33 Family ASIC Macro Manual".

4.1 List of Peripheral Circuits

The E0C33208/204/202 consists of the C33 ASIC Macro Blocks: C33 Core Block, C33 Peripheral Block, C33 DMA Block and C33 Analog Block.

C33 Core Block

CPU	E0C33000 32-bit RISC type CPU
BCU (Bus Control Unit)	24-bit external address bus and 16-bit data bus All the BCU functions can be used.
ITC (Interrupt Controller)	39 types of interrupts are available.
CLG (Clock Generator)	OSC3 oscillation circuit (33 MHz Max.), PLL and OSC1 oscillation circuit (32.768 kHz Typ.) built-in
DBG (Debug Unit)	Functional block for debugging with the ICD33 (In-Circuit Debugger for E0C33 Family)

C33 Peripheral Block

Prescaler	Programmable clock generator for peripheral circuits
8-bit programmable timer	4 channels with clock output function
16-bit programmable timer	6 channels with event counter, clock output and watchdog timer functions
Serial interface	2 channels (asynchronous mode, clock synchronous mode and IrDA are selectable.)
Input and I/O ports	13 bits of input ports and 29 bits of I/O ports (used for peripheral I/O)
Clock timer	1 channel with alarm function

C33 DMA Block

HSDMA (High-Speed DMA)	4 channels
IDMA (Intelligent DMA)	128 channels

C33 Analog Block

A/D converter	10-bit A/D converter with 8 input channels

4.2 I/O Memory Map

Table 4.2.1 I/O Memory Map

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer	**0040146**	D7–4	–	reserved	–				–	–	0 when being read.
clock select	**(B)**	D3	**P8TPCK3**	8-bit timer 3 clock selection	1	θ/1	0	Divided clk.	0	R/W	θ: selected by
register		D2	**P8TPCK2**	8-bit timer 2 clock selection	1	θ/1	0	Divided clk.	0	R/W	Prescaler clock select
		D1	**P8TPCK1**	8-bit timer 1 clock selection	1	θ/1	0	Divided clk.	0	R/W	register (0x40181)
		D0	**P8TPCK0**	8-bit timer 0 clock selection	1	θ/1	0	Divided clk.	0	R/W	
16-bit timer 0	**0040147**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON0**	16-bit timer 0 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS02**	16-bit timer 0	P16TS0[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS01**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS00**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			16-bit timer 0 can be
					0	1	1	θ/16			used as a watchdog
					0	1	0	θ/4			timer.
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 1	**0040148**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON1**	16-bit timer 1 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS12**	16-bit timer 1	P16TS1[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS11**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS10**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 2	**0040149**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON2**	16-bit timer 2 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS22**	16-bit timer 2	P16TS2[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS21**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS20**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 3	**004014A**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON3**	16-bit timer 3 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS32**	16-bit timer 3	P16TS3[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS31**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS30**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 4	**004014B**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON4**	16-bit timer 4 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS42**	16-bit timer 4	P16TS4[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS41**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS40**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			

The meaning of the symbols described in [Init.] are listed below:

0, 1: Initial values that are set at initial reset.
(However, the registers for the bus and input/output ports are not initialized at hot start.)

X: Not initialized at initial reset.

–: Not set in the circuit.

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
16-bit timer 5	**004014C**	D7–4	–	reserved	–	–	–	0 when being read.
clock control	**(B)**	D3	**P16TON5**	16-bit timer 5 clock control	1 On / 0 Off	0	R/W	
register		D2	**P16TS52**	16-bit timer 5	P16TS5[2:0] — Division ratio	0	R/W	θ: selected by
		D1	**P16TS51**	clock division ratio selection	1 1 1 — θ/4096	0		Prescaler clock select
		D0	**P16TS50**		1 1 0 — θ/1024	0		register (0x40181)
					1 0 1 — θ/256			
					1 0 0 — θ/64			
					0 1 1 — θ/16			
					0 1 0 — θ/4			
					0 0 1 — θ/2			
					0 0 0 — θ/1			
8-bit timer 0/1	**004014D**	D7	**P8TON1**	8-bit timer 1 clock control	1 On / 0 Off	0	R/W	
clock control	**(B)**	D6	**P8TS12**	8-bit timer 1	P8TS1[2:0] — Division ratio	0	R/W	θ: selected by
register		D5	**P8TS11**	clock division ratio selection	1 1 1 — θ/4096	0		Prescaler clock select
		D4	**P8TS10**		1 1 0 — θ/2048	0		register (0x40181)
					1 0 1 — θ/1024			
					1 0 0 — θ/512			8-bit timer 1 can
					0 1 1 — θ/256			generate the OSC3
					0 1 0 — θ/128			oscillation-stabilize
					0 0 1 — θ/64			waiting period.
					0 0 0 — θ/32			
		D3	**P8TON0**	8-bit timer 0 clock control	1 On / 0 Off	0	R/W	
		D2	**P8TS02**	8-bit timer 0	P8TS0[2:0] — Division ratio	0	R/W	θ: selected by
		D1	**P8TS01**	clock division ratio selection	1 1 1 — θ/256	0		Prescaler clock select
		D0	**P8TS00**		1 1 0 — θ/128	0		register (0x40181)
					1 0 1 — θ/64			
					1 0 0 — θ/32			8-bit timer 0 can
					0 1 1 — θ/16			generate the DRAM
					0 1 0 — θ/8			refresh clock.
					0 0 1 — θ/4			
					0 0 0 — θ/2			
8-bit timer 2/3	**004014E**	D7	**P8TON3**	8-bit timer 3 clock control	1 On / 0 Off	0	R/W	
clock control	**(B)**	D6	**P8TS32**	8-bit timer 3	P8TS3[2:0] — Division ratio	0	R/W	θ: selected by
register		D5	**P8TS31**	clock division ratio selection	1 1 1 — θ/256	0		Prescaler clock select
		D4	**P8TS30**		1 1 0 — θ/128	0		register (0x40181)
					1 0 1 — θ/64			
					1 0 0 — θ/32			8-bit timer 3 can
					0 1 1 — θ/16			generate the clock for
					0 1 0 — θ/8			the serial I/F Ch.1.
					0 0 1 — θ/4			
					0 0 0 — θ/2			
		D3	**P8TON2**	8-bit timer 2 clock control	1 On / 0 Off	0	R/W	
		D2	**P8TS22**	8-bit timer 2	P8TS2[2:0] — Division ratio	0	R/W	θ: selected by
		D1	**P8TS21**	clock division ratio selection	1 1 1 — θ/4096	0		Prescaler clock select
		D0	**P8TS20**		1 1 0 — θ/2048	0		register (0x40181)
					1 0 1 — θ/64			
					1 0 0 — θ/32			8-bit timer 2 can
					0 1 1 — θ/16			generate the clock for
					0 1 0 — θ/8			the serial I/F Ch.0.
					0 0 1 — θ/4			
					0 0 0 — θ/2			
A/D clock	**004014F**	D7–4	–	reserved	–	–	–	0 when being read.
control register	**(B)**	D3	**PSONAD**	A/D converter clock control	1 On / 0 Off	0	R/W	
		D2	**PSAD2**	A/D converter clock division ratio	P8TS0[2:0] — Division ratio	0	R/W	θ: selected by
		D1	**PSAD1**	selection	1 1 1 — θ/256	0		Prescaler clock select
		D0	**PSAD0**		1 1 0 — θ/128	0		register (0x40181)
					1 0 1 — θ/64			
					1 0 0 — θ/32			
					0 1 1 — θ/16			
					0 1 0 — θ/8			
					0 0 1 — θ/4			
					0 0 0 — θ/2			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Clock timer Run/Stop register	**0040151 (B)**	D7–2	–	reserved	–	–	–	0 when being read.
		D1	**TCRST**	Clock timer reset	1 Reset / 0 Invalid	X	W	0 when being read.
		D0	**TCRUN**	Clock timer Run/Stop control	1 Run / 0 Stop	X	R/W	
Clock timer interrupt control register	**0040152 (B)**	D7	**TCISE2**	Clock timer interrupt factor selection	TCISE[2:0] – Interrupt factor	X	R/W	
		D6	**TCISE1**		1 1 1 – None	X		
		D5	**TCISE0**		1 1 0 – Day	X		
					1 0 1 – Hour			
					1 0 0 – Minute			
					0 1 1 – 1 Hz			
					0 1 0 – 2 Hz			
					0 0 1 – 8 Hz			
					0 0 0 – 32 Hz			
		D4	**TCASE2**	Clock timer alarm factor selection	TCASE[2:0] – Alarm factor	X	R/W	
		D3	**TCASE1**		1 X X – Day	X		
		D2	**TCASE0**		X 1 X – Hour	X		
					X X 1 – Minute			
					0 0 0 – None			
		D1	**TCIF**	Interrupt factor generation flag	1 Generated / 0 Not generated	X	R/W	Reset by writing 1.
		D0	**TCAF**	Alarm factor generation flag	1 Generated / 0 Not generated	X	R/W	Reset by writing 1.
Clock timer divider register	**0040153 (B)**	D7	**TCD7**	Clock timer data 1 Hz	1 High / 0 Low	X	R	
		D6	**TCD6**	Clock timer data 2 Hz	1 High / 0 Low	X	R	
		D5	**TCD5**	Clock timer data 4 Hz	1 High / 0 Low	X	R	
		D4	**TCD4**	Clock timer data 8 Hz	1 High / 0 Low	X	R	
		D3	**TCD3**	Clock timer data 16 Hz	1 High / 0 Low	X	R	
		D2	**TCD2**	Clock timer data 32 Hz	1 High / 0 Low	X	R	
		D1	**TCD1**	Clock timer data 64 Hz	1 High / 0 Low	X	R	
		D0	**TCD0**	Clock timer data 128 Hz	1 High / 0 Low	X	R	
Clock timer second register	**0040154 (B)**	D7–6	–	reserved	–	–	–	0 when being read.
		D5	**TCMD5**	Clock timer second counter data	0 to 59 seconds	X	R	
		D4	**TCMD4**	TCMD5 = MSB		X		
		D3	**TCMD3**	TCMD0 = LSB		X		
		D2	**TCMD2**			X		
		D1	**TCMD1**			X		
		D0	**TCMD0**			X		
Clock timer minute register	**0040155 (B)**	D7–6	–	reserved	–	–	–	0 when being read.
		D5	**TCHD5**	Clock timer minute counter data	0 to 59 minutes	X	R/W	
		D4	**TCHD4**	TCHD5 = MSB		X		
		D3	**TCHD3**	TCHD0 = LSB		X		
		D2	**TCHD2**			X		
		D1	**TCHD1**			X		
		D0	**TCHD0**			X		
Clock timer hour register	**0040156 (B)**	D7–5	–	reserved	–	–	–	0 when being read.
		D4	**TCDD4**	Clock timer hour counter data	0 to 23 hours	X	R/W	
		D3	**TCDD3**	TCDD4 = MSB		X		
		D2	**TCDD2**	TCDD0 = LSB		X		
		D1	**TCDD1**			X		
		D0	**TCDD0**			X		
Clock timer day (low-order) register	**0040157 (B)**	D7	**TCND7**	Clock timer day counter data	0 to 65535 days	X	R/W	
		D6	**TCND6**	(low-order 8 bits)	(low-order 8 bits)	X		
		D5	**TCND5**	TCND0 = LSB		X		
		D4	**TCND4**			X		
		D3	**TCND3**			X		
		D2	**TCND2**			X		
		D1	**TCND1**			X		
		D0	**TCND0**			X		
Clock timer day (high-order) register	**0040158 (B)**	D7	**TCND15**	Clock timer day counter data	0 to 65535 days	X	R/W	
		D6	**TCND14**	(high-order 8 bits)	(high-order 8 bits)	X		
		D5	**TCND13**	TCND15 = MSB		X		
		D4	**TCND12**			X		
		D3	**TCND11**			X		
		D2	**TCND10**			X		
		D1	**TCND9**			X		
		D0	**TCND8**			X		
Clock timer minute comparison register	**0040159 (B)**	D7–6	–	reserved	–	–	–	0 when being read.
		D5	**TCCH5**	Clock timer minute comparison	0 to 59 minutes	X	R/W	
		D4	**TCCH4**	data	(Note) Can be set within 0–63.	X		
		D3	**TCCH3**	TCCH5 = MSB		X		
		D2	**TCCH2**	TCCH0 = LSB		X		
		D1	**TCCH1**			X		
		D0	**TCCH0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Clock timer hour comparison register	**004015A (B)**	D7–5	**–**	reserved	–	–	–	0 when being read.
		D4	**TCCD4**	Clock timer hour comparison data	0 to 23 hours	X	R/W	
		D3	**TCCD3**	TCCD4 = MSB	(Note) Can be set within 0–31.	X		
		D2	**TCCD2**	TCCD0 = LSB		X		
		D1	**TCCD1**			X		
		D0	**TCCD0**			X		
Clock timer day comparison register	**004015B (B)**	D7–5	**–**	reserved	–	–	–	0 when being read.
		D4	**TCCN4**	Clock timer day comparison data	0 to 31 days	X	R/W	Compared with TCND[4:0].
		D3	**TCCN3**	TCCN4 = MSB		X		
		D2	**TCCN2**	TCCN0 = LSB		X		
		D1	**TCCN1**			X		
		D0	**TCCN0**			X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer 0 control register	**0040160 (B)**	D7–3	**–**	reserved	–				–	–	0 when being read.
		D2	**PTOUT0**	8-bit timer 0 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET0**	8-bit timer 0 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN0**	8-bit timer 0 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 0 reload data register	**0040161 (B)**	D7	**RLD07**	8-bit timer 0 reload data	0 to 255				X	R/W	
		D6	**RLD06**	RLD07 = MSB					X		
		D5	**RLD05**	RLD00 = LSB					X		
		D4	**RLD04**						X		
		D3	**RLD03**						X		
		D2	**RLD02**						X		
		D1	**RLD01**						X		
		D0	**RLD00**						X		
8-bit timer 0 counter data register	**0040162 (B)**	D7	**PTD07**	8-bit timer 0 counter data	0 to 255				X	R	
		D6	**PTD06**	PTD07 = MSB					X		
		D5	**PTD05**	PTD00 = LSB					X		
		D4	**PTD04**						X		
		D3	**PTD03**						X		
		D2	**PTD02**						X		
		D1	**PTD01**						X		
		D0	**PTD00**						X		
8-bit timer 1 control register	**0040164 (B)**	D7–3	**–**	reserved	–				–	–	0 when being read.
		D2	**PTOUT1**	8-bit timer 1 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET1**	8-bit timer 1 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN1**	8-bit timer 1 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 1 reload data register	**0040165 (B)**	D7	**RLD17**	8-bit timer 1 reload data	0 to 255				X	R/W	
		D6	**RLD16**	RLD17 = MSB					X		
		D5	**RLD15**	RLD10 = LSB					X		
		D4	**RLD14**						X		
		D3	**RLD13**						X		
		D2	**RLD12**						X		
		D1	**RLD11**						X		
		D0	**RLD10**						X		
8-bit timer 1 counter data register	**0040166 (B)**	D7	**PTD17**	8-bit timer 1 counter data	0 to 255				X	R	
		D6	**PTD16**	PTD17 = MSB					X		
		D5	**PTD15**	PTD10 = LSB					X		
		D4	**PTD14**						X		
		D3	**PTD13**						X		
		D2	**PTD12**						X		
		D1	**PTD11**						X		
		D0	**PTD10**						X		
8-bit timer 2 control register	**0040168 (B)**	D7–3	**–**	reserved	–				–	–	0 when being read.
		D2	**PTOUT2**	8-bit timer 2 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET2**	8-bit timer 2 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN2**	8-bit timer 2 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 2 reload data register	**0040169 (B)**	D7	**RLD27**	8-bit timer 2 reload data	0 to 255				X	R/W	
		D6	**RLD26**	RLD27 = MSB					X		
		D5	**RLD25**	RLD20 = LSB					X		
		D4	**RLD24**						X		
		D3	**RLD23**						X		
		D2	**RLD22**						X		
		D1	**RLD21**						X		
		D0	**RLD20**						X		
8-bit timer 2 counter data register	**004016A (B)**	D7	**PTD27**	8-bit timer 2 counter data	0 to 255				X	R	
		D6	**PTD26**	PTD27 = MSB					X		
		D5	**PTD25**	PTD20 = LSB					X		
		D4	**PTD24**						X		
		D3	**PTD23**						X		
		D2	**PTD22**						X		
		D1	**PTD21**						X		
		D0	**PTD20**						X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer 3 control register	**004016C (B)**	D7–3	–	reserved	–				–	–	0 when being read.
		D2	**PTOUT3**	8-bit timer 3 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET3**	8-bit timer 3 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN3**	8-bit timer 3 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 3 reload data register	**004016D (B)**	D7	**RLD37**	8-bit timer 3 reload data	0 to 255				X	R/W	
		D6	**RLD36**	RLD37 = MSB					X		
		D5	**RLD35**	RLD30 = LSB					X		
		D4	**RLD34**						X		
		D3	**RLD33**						X		
		D2	**RLD32**						X		
		D1	**RLD31**						X		
		D0	**RLD30**						X		
8-bit timer 3 counter data register	**004016E (B)**	D7	**PTD37**	8-bit timer 3 counter data	0 to 255				X	R	
		D6	**PTD36**	PTD37 = MSB					X		
		D5	**PTD35**	PTD30 = LSB					X		
		D4	**PTD34**						X		
		D3	**PTD33**						X		
		D2	**PTD32**						X		
		D1	**PTD31**						X		
		D0	**PTD30**						X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Watchdog timer write-protect register	**0040170 (B)**	D7	**WRWD**	EWD write protection	1	Write enabled	0	Write-protect	0	R/W	
		D6–0	–	–	–				–	–	0 when being read.
Watchdog timer enable register	**0040171 (B)**	D7–2	–	–	–				–	–	0 when being read.
		D1	**EWD**	Watchdog timer enable	1	NMI enabled	0	NMI disabled	0	R/W	
		D0	–	–	–				–	–	0 when being read.

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Power control register	**0040180 (B)**	D7 D6	**CLKDT1** **CLKDT0**	System clock division ratio selection	CLKDT[1:0]		Division ratio		0 0	R/W	
					1	1	1/8				
					1	0	1/4				
					0	1	1/2				
					0	0	1/1				
		D5	**PSCON**	Prescaler On/Off control	1	On	0	Off	1	R/W	
		D4–3	**–**	reserved	–				0	–	Writing 1 not allowed.
		D2	**CLKCHG**	CPU operating clock switch	1	OSC3	0	OSC1	1	R/W	
		D1	**SOSC3**	High-speed (OSC3) oscillation On/Off	1	On	0	Off	1	R/W	
		D0	**SOSC1**	Low-speed (OSC1) oscillation On/Off	1	On	0	Off	1	R/W	
Prescaler clock select register	**0040181 (B)**	D7–1	**–**	reserved	–				0	–	
		D0	**PSCDT0**	Prescaler clock selection	1	OSC1	0	OSC3/PLL	0	R/W	
Clock option register	**0040190 (B)**	D7–4	**–**	–	–				–	–	0 when being read.
		D3	**HLT2OP**	HALT clock option	1	On	0	Off	0	R/W	
		D2	**8T1ON**	OSC3-stabilize waiting function	1	Off	0	On	1	R/W	
		D1	**–**	reserved	–				0	–	Do not write 1.
		D0	**PF1ON**	OSC1 external output control	1	On	0	Off	0	R/W	
Power control protect register	**004019E (B)**	D7 D6 D5 D4 D3 D2 D1 D0	**CLGP7** **CLGP6** **CLGP5** **CLGP4** **CLGP3** **CLGP2** **CLGP1** **CLGP0**	Power control register protect flag	Writing 10010110 (0x96) removes the write protection of the power control register (0x40180). Writing another value set the write protection.				0 0 0 0 0 0 0 0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Serial I/F Ch.0	**00401E0**	D7	**TXD07**	Serial I/F Ch.0 transmit data	0x0 to 0xFF(0x7F)				X	R/W	7-bit asynchronous
transmit data	**(B)**	D6	**TXD06**	TXD07(06) = MSB					X		mode does not use
register		D5	**TXD05**	TXD00 = LSB					X		TXD07.
		D4	**TXD04**						X		
		D3	**TXD03**						X		
		D2	**TXD02**						X		
		D1	**TXD01**						X		
		D0	**TXD00**						X		
Serial I/F Ch.0	**00401E1**	D7	**RXD07**	Serial I/F Ch.0 receive data	0x0 to 0xFF(0x7F)				X	R	7-bit asynchronous
receive data	**(B)**	D6	**RXD06**	RXD07(06) = MSB					X		mode does not use
register		D5	**RXD05**	RXD00 = LSB					X		RXD07 (fixed at 0).
		D4	**RXD04**						X		
		D3	**RXD03**						X		
		D2	**RXD02**						X		
		D1	**RXD01**						X		
		D0	**RXD00**						X		
Serial I/F Ch.0	**00401E2**	D7–6	**–**	–	–				–	–	0 when being read.
status register	**(B)**	D5	**TEND0**	Ch.0 transmit-completion flag	1	Transmitting	0	End	0	R	
		D4	**FER0**	Ch.0 flaming error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D3	**PER0**	Ch.0 parity error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D2	**OER0**	Ch.0 overrun error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D1	**TDBE0**	Ch.0 transmit data buffer empty	1	Empty	0	Buffer full	1	R	
		D0	**RDBF0**	Ch.0 receive data buffer full	1	Buffer full	0	Empty	0	R	
Serial I/F Ch.0	**00401E3**	D7	**TXEN0**	Ch.0 transmit enable	1	Enabled	0	Disabled	0	R/W	
control register	**(B)**	D6	**RXEN0**	Ch.0 receive enable	1	Enabled	0	Disabled	0	R/W	
		D5	**EPR0**	Ch.0 parity enable	1	With parity	0	No parity	X	R/W	Valid only in
		D4	**PMD0**	Ch.0 parity mode selection	1	Odd	0	Even	X	R/W	asynchronous mode.
		D3	**STPB0**	Ch.0 stop bit selection	1	2 bits	0	1 bit	X	R/W	
		D2	**SSCK0**	Ch.0 input clock selection	1	#SCLK0	0	Internal clock	X	R/W	
		D1	**SMD01**	Ch.0 transfer mode selection	SMD0[1:0]		Transfer mode		X	R/W	
		D0	**SMD00**		1	1	8-bit asynchronous		X		
					1	0	7-bit asynchronous				
					0	1	Clock sync. Slave				
					0	0	Clock sync. Master				
Serial I/F Ch.0	**00401E4**	D7–5	**–**	–	–				–	–	0 when being read.
IrDA register	**(B)**	D4	**DIVMD0**	Ch.0 async. clock division ratio	1	1/8	0	1/16	X	R/W	
		D3	**IRTL0**	Ch.0 IrDA I/F output logic inversion	1	Inverted	0	Direct	X	R/W	Valid only in
		D2	**IRRL0**	Ch.0 IrDA I/F input logic inversion	1	Inverted	0	Direct	X	R/W	asynchronous mode.
		D1	**IRMD01**	Ch.0 interface mode selection	IRMD0[1:0]		I/F mode		X	R/W	
		D0	**IRMD00**		1	1	reserved		X		
					1	0	IrDA 1.0				
					0	1	reserved				
					0	0	General I/F				

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Serial I/F Ch.1 transmit data register	**00401E5 (B)**	D7	**TXD17**	Serial I/F Ch.1 transmit data	0x0 to 0xFF(0x7F)	X	R/W	7-bit asynchronous mode does not use TXD17.
		D6	**TXD16**	TXD17(16) = MSB		X		
		D5	**TXD15**	TXD10 = LSB		X		
		D4	**TXD14**			X		
		D3	**TXD13**			X		
		D2	**TXD12**			X		
		D1	**TXD11**			X		
		D0	**TXD10**			X		
Serial I/F Ch.1 receive data register	**00401E6 (B)**	D7	**RXD17**	Serial I/F Ch.1 receive data	0x0 to 0xFF(0x7F)	X	R	7-bit asynchronous mode does not use RXD17 (fixed at 0).
		D6	**RXD16**	RXD17(16) = MSB		X		
		D5	**RXD15**	RXD10 = LSB		X		
		D4	**RXD14**			X		
		D3	**RXD13**			X		
		D2	**RXD12**			X		
		D1	**RXD11**			X		
		D0	**RXD10**			X		
Serial I/F Ch.1 status register	**00401E7 (B)**	D7–6	–	–	–	–	–	0 when being read.
		D5	**TEND1**	Ch.1 transmit-completion flag	1 Transmitting / 0 End	0	R	
		D4	**FER1**	Ch.1 flaming error flag	1 Error / 0 Normal	0	R/W	Reset by writing 0.
		D3	**PER1**	Ch.1 parity error flag	1 Error / 0 Normal	0	R/W	Reset by writing 0.
		D2	**OER1**	Ch.1 overrun error flag	1 Error / 0 Normal	0	R/W	Reset by writing 0.
		D1	**TDBE1**	Ch.1 transmit data buffer empty	1 Empty / 0 Buffer full	1	R	
		D0	**RDBF1**	Ch.1 receive data buffer full	1 Buffer full / 0 Empty	0	R	
Serial I/F Ch.1 control register	**00401E8 (B)**	D7	**TXEN1**	Ch.1 transmit enable	1 Enabled / 0 Disabled	0	R/W	
		D6	**RXEN1**	Ch.1 receive enable	1 Enabled / 0 Disabled	0	R/W	
		D5	**EPR1**	Ch.1 parity enable	1 With parity / 0 No parity	X	R/W	Valid only in asynchronous mode.
		D4	**PMD1**	Ch.1 parity mode selection	1 Odd / 0 Even	X	R/W	
		D3	**STPB1**	Ch.1 stop bit selection	1 2 bits / 0 1 bit	X	R/W	
		D2	**SSCK1**	Ch.1 input clock selection	1 #SCLK1 / 0 Internal clock	X	R/W	
		D1	**SMD11**	Ch.1 transfer mode selection	SMD1[1:0] – Transfer mode	X	R/W	
		D0	**SMD10**		1 1 – 8-bit asynchronous	X		
					1 0 – 7-bit asynchronous			
					0 1 – Clock sync. Slave			
					0 0 – Clock sync. Master			
Serial I/F Ch.1 IrDA register	**00401E9 (B)**	D7–5	–	–	–	–	–	0 when being read.
		D4	**DIVMD1**	Ch.1 async. clock division ratio	1 1/8 / 0 1/16	X	R/W	
		D3	**IRTL1**	Ch.1 IrDA I/F output logic inversion	1 Inverted / 0 Direct	X	R/W	Valid only in asynchronous mode.
		D2	**IRRL1**	Ch.1 IrDA I/F input logic inversion	1 Inverted / 0 Direct	X	R/W	
		D1	**IRMD11**	Ch.1 interface mode selection	IRMD1[1:0] – I/F mode	X	R/W	
		D0	**IRMD10**		1 1 – reserved	X		
					1 0 – IrDA 1.0			
					0 1 – reserved			
					0 0 – General I/F			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
A/D conversion result (low-order) register	**0040240 (B)**	D7	**ADD7**	A/D converted data	0x0 to 0x3FF	0	R	
		D6	**ADD6**	(low-order 8 bits)	(low-order 8 bits)	0		
		D5	**ADD5**	ADD0 = LSB		0		
		D4	**ADD4**			0		
		D3	**ADD3**			0		
		D2	**ADD2**			0		
		D1	**ADD1**			0		
		D0	**ADD0**			0		
A/D conversion result (high-order) register	**0040241 (B)**	D7–2	–	–	–	–	–	0 when being read.
		D1	**ADD9**	A/D converted data	0x0 to 0x3FF	0	R	
		D0	**ADD8**	(high-order 2 bits) ADD9 = MSB	(high-order 2 bits)	0		
A/D trigger register	**0040242 (B)**	D7–6	–	–	–	–	–	0 when being read.
		D5	**MS**	A/D conversion mode selection	1 Continuous / 0 Normal	0	R/W	
		D4	**TS1**	A/D conversion trigger selection	TS[1:0] : Trigger	0	R/W	
		D3	**TS0**		1 1 : #ADTRG pin	0		
					1 0 : 8-bit timer 0			
					0 1 : 16-bit timer 0			
					0 0 : Software			
		D2	**CH2**	A/D conversion channel status	CH[2:0] : Channel	0	R	
		D1	**CH1**		1 1 1 : AD7	0		
		D0	**CH0**		1 1 0 : AD6	0		
					1 0 1 : AD5			
					1 0 0 : AD4			
					0 1 1 : AD3			
					0 1 0 : AD2			
					0 0 1 : AD1			
					0 0 0 : AD0			
A/D channel register	**0040243 (B)**	D7–6	–	–	–	–	–	0 when being read.
		D5	**CE2**	A/D converter	CE[2:0] : End channel	0	R/W	
		D4	**CE1**	end channel selection	1 1 1 : AD7	0		
		D3	**CE0**		1 1 0 : AD6	0		
					1 0 1 : AD5			
					1 0 0 : AD4			
					0 1 1 : AD3			
					0 1 0 : AD2			
					0 0 1 : AD1			
					0 0 0 : AD0			
		D2	**CS2**	A/D converter	CS[2:0] : Start channel	0	R/W	
		D1	**CS1**	start channel selection	1 1 1 : AD7	0		
		D0	**CS0**		1 1 0 : AD6	0		
					1 0 1 : AD5			
					1 0 0 : AD4			
					0 1 1 : AD3			
					0 1 0 : AD2			
					0 0 1 : AD1			
					0 0 0 : AD0			
A/D enable register	**0040244 (B)**	D7–4	–	–	–	–	–	0 when being read.
		D3	**ADF**	Conversion-complete flag	1 Completed / 0 Run/Standby	0	R	Reset when ADD is read.
		D2	**ADE**	A/D enable	1 Enabled / 0 Disabled	0	R/W	
		D1	**ADST**	A/D conversion control/status	1 Start/Run / 0 Stop	0	R/W	
		D0	**OWE**	Overwrite error flag	1 Error / 0 Normal	0	R/W	Reset by writing 0.
A/D sampling register	**0040245 (B)**	D7–2	–	–	–	–	–	0 when being read.
		D1	**ST1**	Input signal sampling time setup	ST[1:0] : Sampring time	1	R/W	Use with 9 clocks.
		D0	**ST0**		1 1 : 9 clocks	1		
					1 0 : 7 clocks			
					0 1 : 5 clocks			
					0 0 : 3 clocks			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Port input 0/1 interrupt priority register	**0040260 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PP1L2**	Port input 1 interrupt level	0 to 7	X	R/W	
		D5	**PP1L1**			X		
		D4	**PP1L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PP0L2**	Port input 0 interrupt level	0 to 7	X	R/W	
		D1	**PP0L1**			X		
		D0	**PP0L0**			X		
Port input 2/3 interrupt priority register	**0040261 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PP3L2**	Port input 3 interrupt level	0 to 7	X	R/W	
		D5	**PP3L1**			X		
		D4	**PP3L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PP2L2**	Port input 2 interrupt level	0 to 7	X	R/W	
		D1	**PP2L1**			X		
		D0	**PP2L0**			X		
Key input interrupt priority register	**0040262 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PK1L2**	Key input 1 interrupt level	0 to 7	X	R/W	
		D5	**PK1L1**			X		
		D4	**PK1L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PK0L2**	Key input 0 interrupt level	0 to 7	X	R/W	
		D1	**PK0L1**			X		
		D0	**PK0L0**			X		
High-speed DMA Ch.0/1 interrupt priority register	**0040263 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PHSD1L2**	High-speed DMA Ch.1 interrupt level	0 to 7	X	R/W	
		D5	**PHSD1L1**			X		
		D4	**PHSD1L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PHSD0L2**	High-speed DMA Ch.0 interrupt level	0 to 7	X	R/W	
		D1	**PHSD0L1**			X		
		D0	**PHSD0L0**			X		
High-speed DMA Ch.2/3 interrupt priority register	**0040264 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PHSD3L2**	High-speed DMA Ch.3 interrupt level	0 to 7	X	R/W	
		D5	**PHSD3L1**			X		
		D4	**PHSD3L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PHSD2L2**	High-speed DMA Ch.2 interrupt level	0 to 7	X	R/W	
		D1	**PHSD2L1**			X		
		D0	**PHSD2L0**			X		
IDMA interrupt priority register	**0040265 (B)**	D7–3	–	reserved	–	–	–	0 when being read.
		D2	**PDM2**	IDMA interrupt level	0 to 7	X	R/W	
		D1	**PDM1**			X		
		D0	**PDM0**			X		
16-bit timer 0/1 interrupt priority register	**0040266 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**P16T12**	16-bit timer 1 interrupt level	0 to 7	X	R/W	
		D5	**P16T11**			X		
		D4	**P16T10**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**P16T02**	16-bit timer 0 interrupt level	0 to 7	X	R/W	
		D1	**P16T01**			X		
		D0	**P16T00**			X		
16-bit timer 2/3 interrupt priority register	**0040267 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**P16T32**	16-bit timer 3 interrupt level	0 to 7	X	R/W	
		D5	**P16T31**			X		
		D4	**P16T30**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**P16T22**	16-bit timer 2 interrupt level	0 to 7	X	R/W	
		D1	**P16T21**			X		
		D0	**P16T20**			X		
16-bit timer 4/5 interrupt priority register	**0040268 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**P16T52**	16-bit timer 5 interrupt level	0 to 7	X	R/W	
		D5	**P16T51**			X		
		D4	**P16T50**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**P16T42**	16-bit timer 4 interrupt level	0 to 7	X	R/W	
		D1	**P16T41**			X		
		D0	**P16T40**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
8-bit timer, serial I/F Ch.0 interrupt priority register	**0040269 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PSIO02**	Serial interface Ch.0 interrupt level	0 to 7	X	R/W	
		D5	**PSIO01**			X		
		D4	**PSIO00**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**P8TM2**	8-bit timer 0–3 interrupt level	0 to 7	X	R/W	
		D1	**P8TM1**			X		
		D0	**P8TM0**			X		
Serial I/F Ch.1, A/D interrupt priority register	**004026A (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PAD2**	A/D converter interrupt level	0 to 7	X	R/W	
		D5	**PAD1**			X		
		D4	**PAD0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PSIO12**	Serial interface Ch.1 interrupt level	0 to 7	X	R/W	
		D1	**PSIO11**			X		
		D0	**PSIO10**			X		
Clock timer interrupt priority register	**004026B (B)**	D7–3	–	reserved	–	–	–	Writing 1 not allowed.
		D2	**PCTM2**	Clock timer interrupt level	0 to 7	X	R/W	
		D1	**PCTM1**			X		
		D0	**PCTM0**			X		
Port input 4/5 interrupt priority register	**004026C (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PP5L2**	Port input 5 interrupt level	0 to 7	X	R/W	
		D5	**PP5L1**			X		
		D4	**PP5L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PP4L2**	Port input 4 interrupt level	0 to 7	X	R/W	
		D1	**PP4L1**			X		
		D0	**PP4L0**			X		
Port input 6/7 interrupt priority register	**004026D (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PP7L2**	Port input 7 interrupt level	0 to 7	X	R/W	
		D5	**PP7L1**			X		
		D4	**PP7L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PP6L2**	Port input 6 interrupt level	0 to 7	X	R/W	
		D1	**PP6L1**			X		
		D0	**PP6L0**			X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Key input, port input 0–3 interrupt enable register	**0040270 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**EK1**	Key input 1	1	Enabled	0	Disabled	0	R/W	
		D4	**EK0**	Key input 0					0	R/W	
		D3	**EP3**	Port input 3					0	R/W	
		D2	**EP2**	Port input 2					0	R/W	
		D1	**EP1**	Port input 1					0	R/W	
		D0	**EP0**	Port input 0					0	R/W	
DMA interrupt enable register	**0040271 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**EIDMA**	IDMA	1	Enabled	0	Disabled	0	R/W	
		D3	**EHDM3**	High-speed DMA Ch.3					0	R/W	
		D2	**EHDM2**	High-speed DMA Ch.2					0	R/W	
		D1	**EHDM1**	High-speed DMA Ch.1					0	R/W	
		D0	**EHDM0**	High-speed DMA Ch.0					0	R/W	
16-bit timer 0/1 interrupt enable register	**0040272 (B)**	D7	**E16TC1**	16-bit timer 1 comparison A	1	Enabled	0	Disabled	0	R/W	
		D6	**E16TU1**	16-bit timer 1 comparison B					0	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**E16TC0**	16-bit timer 0 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU0**	16-bit timer 0 comparison B					0	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
16-bit timer 2/3 interrupt enable register	**0040273 (B)**	D7	**E16TC3**	16-bit timer 3 comparison A	1	Enabled	0	Disabled	0	R/W	
		D6	**E16TU3**	16-bit timer 3 comparison B					0	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**E16TC2**	16-bit timer 2 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
16-bit timer 4/5 interrupt enable register	**0040274 (B)**	D7	**E16TC5**	16-bit timer 5 comparison A	1	Enabled	0	Disabled	0	R/W	
		D6	**E16TU5**	16-bit timer 5 comparison B					0	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**E16TC4**	16-bit timer 4 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU4**	16-bit timer 4 comparison B					0	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
8-bit timer interrupt enable register	**0040275 (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**E8TU3**	8-bit timer 3 underflow	1	Enabled	0	Disabled	0	R/W	
		D2	**E8TU2**	8-bit timer 2 underflow					0	R/W	
		D1	**E8TU1**	8-bit timer 1 underflow					0	R/W	
		D0	**E8TU0**	8-bit timer 0 underflow					0	R/W	
Serial I/F interrupt enable register	**0040276 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**ESTX1**	SIF Ch.1 transmit buffer empty	1	Enabled	0	Disabled	0	R/W	
		D4	**ESRX1**	SIF Ch.1 receive buffer full					0	R/W	
		D3	**ESERR1**	SIF Ch.1 receive error					0	R/W	
		D2	**ESTX0**	SIF Ch.0 transmit buffer empty					0	R/W	
		D1	**ESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D0	**ESERR0**	SIF Ch.0 receive error					0	R/W	
Port input 4–7, clock timer, A/D interrupt enable register	**0040277 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**EP7**	Port input 7	1	Enabled	0	Disabled	0	R/W	
		D4	**EP6**	Port input 6					0	R/W	
		D3	**EP5**	Port input 5					0	R/W	
		D2	**EP4**	Port input 4					0	R/W	
		D1	**ECTM**	Clock timer					0	R/W	
		D0	**EADE**	A/D converter					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Key input, port input 0–3 interrupt factor flag register	**0040280 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**FK1**	Key input 1	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FK0**	Key input 0					X	R/W	
		D3	**FP3**	Port input 3					X	R/W	
		D2	**FP2**	Port input 2					X	R/W	
		D1	**FP1**	Port input 1					X	R/W	
		D0	**FP0**	Port input 0					X	R/W	
DMA interrupt factor flag register	**0040281 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**FIDMA**	IDMA	1	Factor is generated	0	No factor is generated	X	R/W	
		D3	**FHDM3**	High-speed DMA Ch.3					X	R/W	
		D2	**FHDM2**	High-speed DMA Ch.2					X	R/W	
		D1	**FHDM1**	High-speed DMA Ch.1					X	R/W	
		D0	**FHDM0**	High-speed DMA Ch.0					X	R/W	
16-bit timer 0/1 interrupt factor flag register	**0040282 (B)**	D7	**F16TC1**	16-bit timer 1 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU1**	16-bit timer 1 comparison B					X	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**F16TC0**	16-bit timer 0 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU0**	16-bit timer 0 comparison B					X	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
16-bit timer 2/3 interrupt factor flag register	**0040283 (B)**	D7	**F16TC3**	16-bit timer 3 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU3**	16-bit timer 3 comparison B					X	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**F16TC2**	16-bit timer 2 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU2**	16-bit timer 2 comparison B					X	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
16-bit timer 4/5 interrupt factor flag register	**0040284 (B)**	D7	**F16TC5**	16-bit timer 5 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU5**	16-bit timer 5 comparison B					X	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**F16TC4**	16-bit timer 4 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU4**	16-bit timer 4 comparison B					X	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
8-bit timer interrupt factor flag register	**0040285 (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**F8TU3**	8-bit timer 3 underflow	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F8TU2**	8-bit timer 2 underflow					X	R/W	
		D1	**F8TU1**	8-bit timer 1 underflow					X	R/W	
		D0	**F8TU0**	8-bit timer 0 underflow					X	R/W	
Serial I/F interrupt factor flag register	**0040286 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**FSTX1**	SIF Ch.1 transmit buffer empty	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FSRX1**	SIF Ch.1 receive buffer full					X	R/W	
		D3	**FSERR1**	SIF Ch.1 receive error					X	R/W	
		D2	**FSTX0**	SIF Ch.0 transmit buffer empty					X	R/W	
		D1	**FSRX0**	SIF Ch.0 receive buffer full					X	R/W	
		D0	**FSERR0**	SIF Ch.0 receive error					X	R/W	
Port input 4–7, clock timer, A/D interrupt factor flag register	**0040287 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**FP7**	Port input 7	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FP6**	Port input 6					X	R/W	
		D3	**FP5**	Port input 5					X	R/W	
		D2	**FP4**	Port input 4					X	R/W	
		D1	**FCTM**	Clock timer					X	R/W	
		D0	**FADE**	A/D converter					X	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA request register	**0040290 (B)**	D7	**R16TC0**	16-bit timer 0 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**RHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**RHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**RP3**	Port input 3					0	R/W	
		D2	**RP2**	Port input 2					0	R/W	
		D1	**RP1**	Port input 1					0	R/W	
		D0	**RP0**	Port input 0					0	R/W	
16-bit timer 1–4 IDMA request register	**0040291 (B)**	D7	**R16TC4**	16-bit timer 4 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU4**	16-bit timer 4 comparison B					0	R/W	
		D5	**R16TC3**	16-bit timer 3 comparison A					0	R/W	
		D4	**R16TU3**	16-bit timer 3 comparison B					0	R/W	
		D3	**R16TC2**	16-bit timer 2 comparison A					0	R/W	
		D2	**R16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1	**R16TC1**	16-bit timer 1 comparison A					0	R/W	
		D0	**R16TU1**	16-bit timer 1 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register	**0040292 (B)**	D7	**RSTX0**	SIF Ch.0 transmit buffer empty	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RSRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**R8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**R8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**R8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**R8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**R16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**R16TU5**	16-bit timer 5 comparison B					0	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA request register	**0040293 (B)**	D7	**RP7**	Port input 7	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RP6**	Port input 6					0	R/W	
		D5	**RP5**	Port input 5					0	R/W	
		D4	**RP4**	Port input 4					0	R/W	
		D3	–	reserved	–				–	–	0 when being read.
		D2	**RADE**	A/D converter	1	IDMA request	0	Interrupt request	0	R/W	
		D1	**RSTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**RSRX1**	SIF Ch.1 receive buffer full					0	R/W	
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA enable register	**0040294 (B)**	D7	**DE16TC0**	16-bit timer 0 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**DEHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**DEHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**DEP3**	Port input 3					0	R/W	
		D2	**DEP2**	Port input 2					0	R/W	
		D1	**DEP1**	Port input 1					0	R/W	
		D0	**DEP0**	Port input 0					0	R/W	
16-bit timer 1–4 IDMA enable register	**0040295 (B)**	D7	**DE16TC4**	16-bit timer 4 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU4**	16-bit timer 4 comparison B					0	R/W	
		D5	**DE16TC3**	16-bit timer 3 comparison A					0	R/W	
		D4	**DE16TU3**	16-bit timer 3 comparison B					0	R/W	
		D3	**DE16TC2**	16-bit timer 2 comparison A					0	R/W	
		D2	**DE16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1	**DE16TC1**	16-bit timer 1 comparison A					0	R/W	
		D0	**DE16TU1**	16-bit timer 1 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register	**0040296 (B)**	D7	**DESTX0**	SIF Ch.0 transmit buffer empty	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**DE8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**DE8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**DE8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**DE8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**DE16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**DE16TU5**	16-bit timer 5 comparison B					0	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA enable register	**0040297 (B)**	D7	**DEP7**	Port input 7	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DEP6**	Port input 6					0	R/W	
		D5	**DEP5**	Port input 5					0	R/W	
		D4	**DEP4**	Port input 4					0	R/W	
		D3	–	reserved	–				–	–	0 when being read.
		D2	**DEADE**	A/D converter	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D1	**DESTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**DESRX1**	SIF Ch.1 receive buffer full					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
High-speed DMA Ch.0/1 trigger set-up register	**0040298 (B)**	D7 D6 D5 D4	**HSD1S3** **HSD1S2** **HSD1S1** **HSD1S0**	High-speed DMA Ch.1 trigger set-up	0 1 2 3 4 5 6 7 8 9 A B C	Software trigger K51 input (falling edge) K51 input (rising edge) Port 1 input Port 5 input 8-bit timer Ch.1 underflow 16-bit timer Ch.1 compare B 16-bit timer Ch.1 compare A 16-bit timer Ch.5 compare B 16-bit timer Ch.5 compare A SI/F Ch.1 Rx buffer full SI/F Ch.1 Tx buffer empty A/D conversion completion			0 0 0 0	R/W	
		D3 D2 D1 D0	**HSD0S3** **HSD0S2** **HSD0S1** **HSD0S0**	High-speed DMA Ch.0 trigger set-up	0 1 2 3 4 5 6 7 8 9 A B C	Software trigger K50 input (falling edge) K50 input (rising edge) Port 0 input Port 4 input 8-bit timer Ch.0 underflow 16-bit timer Ch.0 compare B 16-bit timer Ch.0 compare A 16-bit timer Ch.4 compare B 16-bit timer Ch.4 compare A SI/F Ch.0 Rx buffer full SI/F Ch.0 Tx buffer empty A/D conversion completion			0 0 0 0	R/W	
High-speed DMA Ch.2/3 trigger set-up register	**0040299 (B)**	D7 D6 D5 D4	**HSD3S3** **HSD3S2** **HSD3S1** **HSD3S0**	High-speed DMA Ch.3 trigger set-up	0 1 2 3 4 5 6 7 8 9 A B C	Software trigger K54 input (falling edge) K54 input (rising edge) Port 3 input Port 7 input 8-bit timer Ch.3 underflow 16-bit timer Ch.3 compare B 16-bit timer Ch.3 compare A 16-bit timer Ch.5 compare B 16-bit timer Ch.5 compare A SI/F Ch.1 Rx buffer full SI/F Ch.1 Tx buffer empty A/D conversion completion			0 0 0 0	R/W	
		D3 D2 D1 D0	**HSD2S3** **HSD2S2** **HSD2S1** **HSD2S0**	High-speed DMA Ch.2 trigger set-up	0 1 2 3 4 5 6 7 8 9 A B C	Software trigger K53 input (falling edge) K53 input (rising edge) Port 2 input Port 6 input 8-bit timer Ch.2 underflow 16-bit timer Ch.2 compare B 16-bit timer Ch.2 compare A 16-bit timer Ch.4 compare B 16-bit timer Ch.4 compare A SI/F Ch.0 Rx buffer full SI/F Ch.0 Tx buffer empty A/D conversion completion			0 0 0 0	R/W	
High-speed DMA software trigger register	**004029A (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**HST3**	HSDMA Ch.3 software trigger	1	Trigger	0	Invalid	0	W	
		D2	**HST2**	HSDMA Ch.2 software trigger					0	W	
		D1	**HST1**	HSDMA Ch.1 software trigger					0	W	
		D0	**HST0**	HSDMA Ch.0 software trigger					0	W	
Flag set/reset method select register	**004029F (B)**	D7–3	**–**	reserved	–				–	–	
		D2	**DENONLY**	IDMA enable register set method selection	1	Set only	0	RD/WR	1	R/W	
		D1	**IDMAONLY**	IDMA request register set method selection	1	Set only	0	RD/WR	1	R/W	
		D0	**RSTONLY**	Interrupt factor flag reset method selection	1	Reset only	0	RD/WR	1	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
K5 function select register	**00402C0 (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**CFK54**	K54 function selection	1	#DMAREQ3	0	K54	0	R/W	
		D3	**CFK53**	K53 function selection	1	#DMAREQ2	0	K53	0	R/W	
		D2	**CFK52**	K52 function selection	1	#ADTRG	0	K52	0	R/W	
		D1	**CFK51**	K51 function selection	1	#DMAREQ1	0	K51	0	R/W	
		D0	**CFK50**	K50 function selection	1	#DMAREQ0	0	K50	0	R/W	
K5 input port data register	**00402C1 (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**K54D**	K54 input port data	1	High	0	Low	–	R	
		D3	**K53D**	K53 input port data					–	R	
		D2	**K52D**	K52 input port data					–	R	
		D1	**K51D**	K51 input port data					–	R	
		D0	**K50D**	K50 input port data					–	R	
K6 function select register	**00402C3 (B)**	D7	**CFK67**	K67 function selection	1	AD7	0	K67	0	R/W	
		D6	**CFK66**	K66 function selection	1	AD6	0	K66	0	R/W	
		D5	**CFK65**	K65 function selection	1	AD5	0	K65	0	R/W	
		D4	**CFK64**	K64 function selection	1	AD4	0	K64	0	R/W	
		D3	**CFK63**	K63 function selection	1	AD3	0	K63	0	R/W	
		D2	**CFK62**	K62 function selection	1	AD2	0	K62	0	R/W	
		D1	**CFK61**	K61 function selection	1	AD1	0	K61	0	R/W	
		D0	**CFK60**	K60 function selection	1	AD0	0	K60	0	R/W	
K6 input port data register	**00402C4 (B)**	D7	**K67D**	K67 input port data	1	High	0	Low	–	R	
		D6	**K66D**	K66 input port data					–	R	
		D5	**K65D**	K65 input port data					–	R	
		D4	**K64D**	K64 input port data					–	R	
		D3	**K63D**	K63 input port data					–	R	
		D2	**K62D**	K62 input port data					–	R	
		D1	**K61D**	K61 input port data					–	R	
		D0	**K60D**	K60 input port data					–	R	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Port input interrupt select register 1	**00402C6 (B)**	D7	**SPT31**	FPT3 interrupt input port selection	11	10	01	00	0	R/W	
		D6	**SPT30**		P23	P03	K53	K63	0		
		D5	**SPT21**	FPT2 interrupt input port selection	11	10	01	00	0	R/W	
		D4	**SPT20**		P22	P02	K52	K62	0		
		D3	**SPT11**	FPT1 interrupt input port selection	11	10	01	00	0	R/W	
		D2	**SPT10**		P21	P01	K51	K61	0		
		D1	**SPT01**	FPT0 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPT00**		P20	P00	K50	K60	0		
Port input interrupt select register 2	**00402C7 (B)**	D7	**SPT71**	FPT7 interrupt input port selection	11	10	01	00	0	R/W	
		D6	**SPT70**		P27	P07	P33	K67	0		
		D5	**SPT61**	FPT6 interrupt input port selection	11	10	01	00	0	R/W	
		D4	**SPT60**		P26	P06	P32	K66	0		
		D3	**SPT51**	FPT5 interrupt input port selection	11	10	01	00	0	R/W	
		D2	**SPT50**		P25	P05	P31	K65	0		
		D1	**SPT41**	FPT4 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPT40**		P24	P04	K54	K64	0		
Port input interrupt input polarity select register	**00402C8 (B)**	D7	**SPPT7**	FPT7 input polarity selection	1	High level or Rising edge	0	Low level or Falling edge	1	R/W	
		D6	**SPPT6**	FPT6 input polarity selection					1	R/W	
		D5	**SPPT5**	FPT5 input polarity selection					1	R/W	
		D4	**SPPT4**	FPT4 input polarity selection					1	R/W	
		D3	**SPPT3**	FPT3 input polarity selection					1	R/W	
		D2	**SPPT2**	FPT2 input polarity selection					1	R/W	
		D1	**SPPT1**	FPT1 input polarity selection					1	R/W	
		D0	**SPPT0**	FPT0 input polarity selection					1	R/W	
Port input interrupt edge/level select register	**00402C9 (B)**	D7	**SEPT7**	FPT7 edge/level selection	1	Edge	0	Level	1	R/W	
		D6	**SEPT6**	FPT6 edge/level selection					1	R/W	
		D5	**SEPT5**	FPT5 edge/level selection					1	R/W	
		D4	**SEPT4**	FPT4 edge/level selection					1	R/W	
		D3	**SEPT3**	FPT3 edge/level selection					1	R/W	
		D2	**SEPT2**	FPT2 edge/level selection					1	R/W	
		D1	**SEPT1**	FPT1 edge/level selection					1	R/W	
		D0	**SEPT0**	FPT0 edge/level selection					1	R/W	
Key input interrupt select register	**00402CA (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**SPPK11**	FPK1 interrupt input port selection	11	10	01	00	0	R/W	
		D2	**SPPK10**		P2[7:4]	P0[7:4]	K6[7:4]	K6[3:0]	0		
		D1	**SPPK01**	FPK0 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPPK00**		P2[4:0]	P0[4:0]	K6[4:0]	K5[4:0]	0		
Key input interrupt (FPK0) input comparison register	**00402CC (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**SCPK04**	FPK04 input comparison	1	High	0	Low	0	R/W	
		D3	**SCPK03**	FPK03 input comparison					0	R/W	
		D2	**SCPK02**	FPK02 input comparison					0	R/W	
		D1	**SCPK01**	FPK01 input comparison					0	R/W	
		D0	**SCPK00**	FPK00 input comparison					0	R/W	
Key input interrupt (FPK1) input comparison register	**00402CD (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**SCPK13**	FPK13 input comparison	1	High	0	Low	0	R/W	
		D2	**SCPK12**	FPK12 input comparison					0	R/W	
		D1	**SCPK11**	FPK11 input comparison					0	R/W	
		D0	**SCPK10**	FPK10 input comparison					0	R/W	
Key input interrupt (FPK0) input mask register	**00402CE (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**SMPK04**	FPK04 input mask	1	Interrupt enabled	0	Interrupt disabled	0	R/W	
		D3	**SMPK03**	FPK03 input mask					0	R/W	
		D2	**SMPK02**	FPK02 input mask					0	R/W	
		D1	**SMPK01**	FPK01 input mask					0	R/W	
		D0	**SMPK00**	FPK00 input mask					0	R/W	
Key input interrupt (FPK1) input mask register	**00402CF (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**SMPK13**	FPK13 input mask	1	Interrupt enabled	0	Interrupt disabled	0	R/W	
		D2	**SMPK12**	FPK12 input mask					0	R/W	
		D1	**SMPK11**	FPK11 input mask					0	R/W	
		D0	**SMPK10**	FPK10 input mask					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
P0 function select register	**00402D0 (B)**	D7	**CFP07**	P07 function selection	1	#SRDY1	0	P07	0	R/W	Extended functions (0x402DF)
		D6	**CFP06**	P06 function selection	1	#SCLK1	0	P06	0	R/W	
		D5	**CFP05**	P05 function selection	1	SOUT1	0	P05	0	R/W	
		D4	**CFP04**	P04 function selection	1	SIN1	0	P04	0	R/W	
		D3	**CFP03**	P03 function selection	1	#SRDY0	0	P03	0	R/W	
		D2	**CFP02**	P02 function selection	1	#SCLK0	0	P02	0	R/W	
		D1	**CFP01**	P01 function selection	1	SOUT0	0	P01	0	R/W	
		D0	**CFP00**	P00 function selection	1	SIN0	0	P00	0	R/W	
P0 I/O port data register	**00402D1 (B)**	D7	**P07D**	P07 I/O port data	1	High	0	Low	0	R/W	
		D6	**P06D**	P06 I/O port data					0	R/W	
		D5	**P05D**	P05 I/O port data					0	R/W	
		D4	**P04D**	P04 I/O port data					0	R/W	
		D3	**P03D**	P03 I/O port data					0	R/W	
		D2	**P02D**	P02 I/O port data					0	R/W	
		D1	**P01D**	P01 I/O port data					0	R/W	
		D0	**P00D**	P00 I/O port data					0	R/W	
P0 I/O control register	**00402D2 (B)**	D7	**IOC07**	P07 I/O control	1	Output	0	Input	0	R/W	
		D6	**IOC06**	P06 I/O control					0	R/W	
		D5	**IOC05**	P05 I/O control					0	R/W	
		D4	**IOC04**	P04 I/O control					0	R/W	
		D3	**IOC03**	P03 I/O control					0	R/W	
		D2	**IOC02**	P02 I/O control					0	R/W	
		D1	**IOC01**	P01 I/O control					0	R/W	
		D0	**IOC00**	P00 I/O control					0	R/W	
P1 function select register	**00402D4 (B)**	D7	–	reserved	–				–	–	0 when being read.
		D6	**CFP16**	P16 function selection	1	EXCL5 #DMAEND1	0	P16	0	R/W	
		D5	**CFP15**	P15 function selection	1	EXCL4 #DMAEND0	0	P15	0	R/W	
		D4	**CFP14**	P14 function selection	1	FOSC1	0	P14	0	R/W	Extended functions (0x402DF)
		D3	**CFP13**	P13 function selection	1	EXCL3 T8UF3	0	P13	0	R/W	
		D2	**CFP12**	P12 function selection	1	EXCL2 T8UF2	0	P12	0	R/W	
		D1	**CFP11**	P11 function selection	1	EXCL1 T8UF1	0	P11	0	R/W	
		D0	**CFP10**	P10 function selection	1	EXCL0 T8UF0	0	P10	0	R/W	
P1 I/O port data register	**00402D5 (B)**	D7	–	reserved	–				–	–	0 when being read.
		D6	**P16D**	P16 I/O port data	1	High	0	Low	0	R/W	
		D5	**P15D**	P15 I/O port data					0	R/W	
		D4	**P14D**	P14 I/O port data					0	R/W	
		D3	**P13D**	P13 I/O port data					0	R/W	
		D2	**P12D**	P12 I/O port data					0	R/W	
		D1	**P11D**	P11 I/O port data					0	R/W	
		D0	**P10D**	P10 I/O port data					0	R/W	
P1 I/O control register	**00402D6 (B)**	D7	–	reserved	–				–	–	0 when being read.
		D6	**IOC16**	P16 I/O control	1	Output	0	Input	0	R/W	
		D5	**IOC15**	P15 I/O control					0	R/W	
		D4	**IOC14**	P14 I/O control					0	R/W	
		D3	**IOC13**	P13 I/O control					0	R/W	
		D2	**IOC12**	P12 I/O control					0	R/W	
		D1	**IOC11**	P11 I/O control					0	R/W	
		D0	**IOC10**	P10 I/O control					0	R/W	
P2 function select register	**00402D8 (B)**	D7	**CFP27**	P27 function selection	1	TM5	0	P27	0	R/W	
		D6	**CFP26**	P26 function selection	1	TM4	0	P26	0	R/W	
		D5	**CFP25**	P25 function selection	1	TM3	0	P25	0	R/W	
		D4	**CFP24**	P24 function selection	1	TM2	0	P24	0	R/W	
		D3	**CFP23**	P23 function selection	1	TM1	0	P23	0	R/W	
		D2	**CFP22**	P22 function selection	1	TM0	0	P22	0	R/W	
		D1	**CFP21**	P21 function selection	1	#DWE	0	P21	0	R/W	Ext. func.(0x402DF)
		D0	**CFP20**	P20 function selection	1	#DRD	0	P20	0	R/W	
P2 I/O port data register	**00402D9 (B)**	D7	**P27D**	P27 I/O port data	1	High	0	Low	0	R/W	
		D6	**P26D**	P26 I/O port data					0	R/W	
		D5	**P25D**	P25 I/O port data					0	R/W	
		D4	**P24D**	P24 I/O port data					0	R/W	
		D3	**P23D**	P23 I/O port data					0	R/W	
		D2	**P22D**	P22 I/O port data					0	R/W	
		D1	**P21D**	P21 I/O port data					0	R/W	
		D0	**P20D**	P20 I/O port data					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
P2 I/O control register	**00402DA (B)**	D7	**IOC27**	P27 I/O control	1	Output	0	Input	0	R/W	
		D6	**IOC26**	P26 I/O control					0	R/W	
		D5	**IOC25**	P25 I/O control					0	R/W	
		D4	**IOC24**	P24 I/O control					0	R/W	
		D3	**IOC23**	P23 I/O control					0	R/W	
		D2	**IOC22**	P22 I/O control					0	R/W	
		D1	**IOC21**	P21 I/O control					0	R/W	
		D0	**IOC20**	P20 I/O control					0	R/W	
P3 function select register	**00402DC (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**CFP35**	P35 function selection	1	#BUSACK	0	P35	0	R/W	
		D4	**CFP34**	P34 function selection	1	#BUSREQ #CE6	0	P34	0	R/W	
		D3	**CFP33**	P33 function selection	1	#DMAACK1	0	P33	0	R/W	
		D2	**CFP32**	P32 function selection	1	#DMAACK0	0	P32	0	R/W	
		D1	**CFP31**	P31 function selection	1	#BUSGET	0	P31	0	R/W	Ext. func.(0x402DF)
		D0	**CFP30**	P30 function selection	1	#WAIT #CE4/#CE5	0	P30	0	R/W	
P3 I/O port data register	**00402DD (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**P35D**	P35 I/O port data	1	High	0	Low	0	R/W	
		D4	**P34D**	P34 I/O port data					0	R/W	
		D3	**P33D**	P33 I/O port data					0	R/W	
		D2	**P32D**	P32 I/O port data					0	R/W	
		D1	**P31D**	P31 I/O port data					0	R/W	
		D0	**P30D**	P30 I/O port data					0	R/W	
P3 I/O control register	**00402DE (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**IOC35**	P35 I/O control	1	Output	0	Input	0	R/W	
		D4	**IOC34**	P34 I/O control					0	R/W	
		D3	**IOC33**	P33 I/O control					0	R/W	
		D2	**IOC32**	P32 I/O control					0	R/W	
		D1	**IOC31**	P31 I/O control					0	R/W	
		D0	**IOC30**	P30 I/O control					0	R/W	
Port function extension register	**00402DF (B)**	D7	**CFEX7**	P07 port extended function	1	#DMAEND3	0	P07, etc.	0	R/W	
		D6	**CFEX6**	P06 port extended function	1	#DMAACK3	0	P06, etc.	0	R/W	
		D5	**CFEX5**	P05 port extended function	1	#DMAEND2	0	P05, etc.	0	R/W	
		D4	**CFEX4**	P04 port extended function	1	#DMAACK2	0	P04, etc.	0	R/W	
		D3	**CFEX3**	P31 port extended function	1	#GARD	0	P31, etc.	0	R/W	
		D2	**CFEX2**	P21 port extended function	1	#GAAS	0	P21, etc.	0	R/W	
		D1	**CFEX1**	P10, P11, P13 port extended function	1	DST0 DST1 DPC0	0	P10, etc. P11, etc. P13, etc.	1	R/W	
		D0	**CFEX0**	P12, P14 port extended function	1	DST2 DCLK	0	P12, etc. P14, etc.	1	R/W	

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Areas 18–15	**0048120**	DF	–	reserved	–	–	–	0 when being read.
set-up register	**(HW)**	DE	**A18SZ**	Areas 18–17 device size selection	1 8 bits / 0 16 bits	0	R/W	
		DD	**A18DF1**	Areas 18–17	A18DF[1:0] / Number of cycles	1	R/W	
		DC	**A18DF0**	output disable delay time	1 1 / 3.5	1		
					1 0 / 2.5			
					0 1 / 1.5			
					0 0 / 0.5			
		DB	–	reserved	–	–	–	0 when being read.
		DA	**A18WT2**	Areas 18–17 wait control	A18WT[2:0] / Wait cycles	1	R/W	
		D9	**A18WT1**		1 1 1 / 7	1		
		D8	**A18WT0**		1 1 0 / 6	1		
					1 0 1 / 5			
					1 0 0 / 4			
					0 1 1 / 3			
					0 1 0 / 2			
					0 0 1 / 1			
					0 0 0 / 0			
		D7	–	reserved	–	–	–	0 when being read.
		D6	**A16SZ**	Areas 16–15 device size selection	1 8 bits / 0 16 bits	0	R/W	
		D5	**A16DF1**	Areas 16–15	A16DF[1:0] / Number of cycles	1	R/W	
		D4	**A16DF0**	output disable delay time	1 1 / 3.5	1		
					1 0 / 2.5			
					0 1 / 1.5			
					0 0 / 0.5			
		D3	–	reserved	–	–	–	0 when being read.
		D2	**A16WT2**	Areas 16–15 wait control	A16WT[2:0] / Wait cycles	1	R/W	
		D1	**A16WT1**		1 1 1 / 7	1		
		D0	**A16WT0**		1 1 0 / 6	1		
					1 0 1 / 5			
					1 0 0 / 4			
					0 1 1 / 3			
					0 1 0 / 2			
					0 0 1 / 1			
					0 0 0 / 0			
Areas 14–13	**0048122**	DF–9	–	reserved	–	–	–	0 when being read.
set-up register	**(HW)**	D8	**A14DRA**	Area 14 DRAM selection	1 Used / 0 Not used	0	R/W	
		D7	**A13DRA**	Area 13 DRAM selection	1 Used / 0 Not used	0	R/W	
		D6	**A14SZ**	Areas 14–13 device size selection	1 8 bits / 0 16 bits	0	R/W	
		D5	**A14DF1**	Areas 14–13	A14DF[1:0] / Number of cycles	1	R/W	
		D4	**A14DF0**	output disable delay time	1 1 / 3.5	1		
					1 0 / 2.5			
					0 1 / 1.5			
					0 0 / 0.5			
		D3	–	reserved	–	–	–	0 when being read.
		D2	**A14WT2**	Areas 14–13 wait control	A14WT[2:0] / Wait cycles	1	R/W	
		D1	**A14WT1**		1 1 1 / 7	1		
		D0	**A14WT0**		1 1 0 / 6	1		
					1 0 1 / 5			
					1 0 0 / 4			
					0 1 1 / 3			
					0 1 0 / 2			
					0 0 1 / 1			
					0 0 0 / 0			
Areas 12–11	**0048124**	DF–7	–	reserved	–	–	–	0 when being read.
set-up register	**(HW)**	D6	**A12SZ**	Areas 12–11 device size selection	1 8 bits / 0 16 bits	0	R/W	
		D5	**A12DF1**	Areas 12–11	A18DF[1:0] / Number of cycles	1	R/W	
		D4	**A12DF0**	output disable delay time	1 1 / 3.5	1		
					1 0 / 2.5			
					0 1 / 1.5			
					0 0 / 0.5			
		D3	–	reserved	–	–	–	0 when being read.
		D2	**A12WT2**	Areas 12–11 wait control	A18WT[2:0] / Wait cycles	1	R/W	
		D1	**A12WT1**		1 1 1 / 7	1		
		D0	**A12WT0**		1 1 0 / 6	1		
					1 0 1 / 5			
					1 0 0 / 4			
					0 1 1 / 3			
					0 1 0 / 2			
					0 0 1 / 1			
					0 0 0 / 0			

Register name	Address	Bit	Name	Function	Setting		Init.	R/W	Remarks
Areas 10–9	**0048126**	DF	–	reserved	–		–	–	0 when being read.
set-up register	**(HW)**	DE	**A10IR2**	Area 10 internal ROM wait control	A10IR[2:0]	ROM size	1	R/W	
		DD	**A10IR1**	Area 10 internal ROM size	1 1 1	2MB	1		
		DC	**A10IR0**	selection	1 1 0	1MB	1		
					1 0 1	512KB			
					1 0 0	256KB			
					0 1 1	128KB			
					0 1 0	64KB			
					0 0 1	32KB			
					0 0 0	16KB			
		DB	–	reserved	–		–	–	0 when being read.
		DA	**A10BW1**	Areas 10–9	A10BW[1:0]	Wait cycles	0	R/W	
		D9	**A10BW0**	burst ROM	1 1	3	0		
				burst read cycle wait control	1 0	2			
					0 1	1			
					0 0	0			
		D8	**A10DRA**	Area 10 burst ROM selection	1 Used	0 Not used	0	R/W	
		D7	**A9DRA**	Area 9 burst ROM selection	1 Used	0 Not used	0	R/W	
		D6	**A10SZ**	Areas 10–9 device size selection	1 8 bits	0 16 bits	0	R/W	
		D5	**A10DF1**	Areas 10–9	A10DF[1:0]	Number of cycles	1	R/W	
		D4	**A10DF0**	output disable delay time	1 1	3.5	1		
					1 0	2.5			
					0 1	1.5			
					0 0	0.5			
		D3	–	reserved	–		–	–	0 when being read.
		D2	**A10WT2**	Areas 10–9 wait control	A10WT[2:0]	Wait cycles	1	R/W	
		D1	**A10WT1**		1 1 1	7	1		
		D0	**A10WT0**		1 1 0	6	1		
					1 0 1	5			
					1 0 0	4			
					0 1 1	3			
					0 1 0	2			
					0 0 1	1			
					0 0 0	0			
Areas 8–7	**0048128**	DF–9	–	reserved	–		–	–	0 when being read.
set-up register	**(HW)**	D8	**A8DRA**	Area 8 DRAM selection	1 Used	0 Not used	0	R/W	
		D7	**A7DRA**	Area 7 DRAM selection	1 Used	0 Not used	0	R/W	
		D6	**A8SZ**	Areas 8–7 device size selection	1 8 bits	0 16 bits	0	R/W	
		D5	**A8DF1**	Areas 8–7	A8DF[1:0]	Number of cycles	1	R/W	
		D4	**A8DF0**	output disable delay time	1 1	3.5	1		
					1 0	2.5			
					0 1	1.5			
					0 0	0.5			
		D3	–	reserved	–		–	–	0 when being read.
		D2	**A8WT2**	Areas 8–7 wait control	A8WT[2:0]	Wait cycles	1	R/W	
		D1	**A8WT1**		1 1 1	7	1		
		D0	**A8WT0**		1 1 0	6	1		
					1 0 1	5			
					1 0 0	4			
					0 1 1	3			
					0 1 0	2			
					0 0 1	1			
					0 0 0	0			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Areas 6–4 set-up register	**004812A (HW)**	DF–E	–	reserved	–	–	–	0 when being read.
		DD	**A6DF1**	Area 6 output disable delay time	A6DF[1:0] : Number of cycles 1 1 : 3.5 1 0 : 2.5 0 1 : 1.5 0 0 : 0.5	1	R/W	
		DC	**A6DF0**			1		
		DB	–	reserved	–	–	–	0 when being read.
		DA	**A6WT2**	Area 6 wait control	A6WT[2:0] : Wait cycles 1 1 1 : 7 1 1 0 : 6 1 0 1 : 5 1 0 0 : 4 0 1 1 : 3 0 1 0 : 2 0 0 1 : 1 0 0 0 : 0	1	R/W	
		D9	**A6WT1**			1		
		D8	**A6WT0**			1		
		D7	–	reserved	–	–	–	0 when being read.
		D6	**A5SZ**	Areas 5–4 device size selection	1: 8 bits / 0: 16 bits	0	R/W	
		D5	**A5DF1**	Areas 5–4 output disable delay time	A5DF[1:0] : Number of cycles 1 1 : 3.5 1 0 : 2.5 0 1 : 1.5 0 0 : 0.5	1	R/W	
		D4	**A5DF0**			1		
		D3	–	reserved		–	–	0 when being read.
		D2	**A5WT2**	Areas 5–4 wait control	A5WT[2:0] : Wait cycles 1 1 1 : 7 1 1 0 : 6 1 0 1 : 5 1 0 0 : 4 0 1 1 : 3 0 1 0 : 2 0 0 1 : 1 0 0 0 : 0	1	R/W	
		D1	**A5WT1**			1		
		D0	**A5WT0**			1		
TTBR write protect register	**004812D (B)**	D7	**TBRP7**	TTBR register write protect	Writing 01011001(0x59) removes the TTBR (0x48134) write protection. Writing other data sets the write protection.	0	W	Undefined in read.
		D6	**TBRP6**			0		
		D5	**TBRP5**			0		
		D4	**TBRP4**			0		
		D3	**TBRP3**			0		
		D2	**TBRP2**			0		
		D1	**TBRP1**			0		
		D0	**TBRP0**			0		
Bus control register	**004812E (HW)**	DF	**RBCLK**	BCLK output control	1: Fixed at H / 0: Enabled	0	R/W	
		DE	–	reserved	–	0	–	Writing 1 not allowed.
		DD	**RBST8**	Burst ROM burst mode selection	1: 8-successive / 0: 4-successive	0	R/W	
		DC	**REDO**	DRAM page mode selection	1: EDO / 0: Fast page	0	R/W	
		DB	**RCA1**	Column address size selection	RCA[1:0] : Size 1 1 : 11 1 0 : 10 0 1 : 9 0 0 : 8	0	R/W	
		DA	**RCA0**			0		
		D9	**RPC2**	Refresh enable	1: Enabled / 0: Disabled	0	R/W	
		D8	**RPC1**	Refresh method selection	1: Self-refresh / 0: CBR-refresh	0	R/W	
		D7	**RPC0**	Refresh RPC delay setup	1: 2.0 / 0: 1.0	0	R/W	
		D6	**RRA1**	Refresh RAS pulse width selection	RRA[1:0] : Number of cycles 1 1 : 5 1 0 : 4 0 1 : 3 0 0 : 2	0	R/W	
		D5	**RRA0**			0		
		D4	–	reserved	–	0	–	Writing 1 not allowed.
		D3	**SBUSST**	External interface method selection	1: #BSL / 0: A0	0	R/W	
		D2	**SEMAS**	External bus master setup	1: Existing / 0: Nonexistent	0	R/W	
		D1	**SEPD**	External power-down control	1: Enabled / 0: Disabled	0	R/W	
		D0	**SWAITE**	#WAIT enable	1: Enabled / 0: Disabled	0	R/W	

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
DRAM timing set-up register	**0048130 (HW)**	DF–C	**–**	reserved	–	–	–	0 when being read.
		DB	**A3EEN**	Area 3 emulation	1 Internal ROM / 0 Emulation	1	R/W	
		DA	**CEFUNC1**	#CE pin function selection	CFFUNC[1:0] / #CE output	0	R/W	
		D9	**CEFUNC0**		1 x #CE7/8..#CE17/18	0		
					0 1 #CE6..#CE17			
					0 0 #CE4..#CE10			
		D8	**CRAS**	Successive RAS mode setup	1 Successive / 0 Normal	0	R/W	
		D7	**RPRC1**	DRAM	RPRC[1:0] / Number of cycles	0	R/W	
		D6	**RPRC0**	RAS precharge cycles selection	1 1 4	0		
					1 0 3			
					0 1 2			
					0 0 1			
		D5	**–**	reserved	–	–	–	0 when being read.
		D4	**CASC1**	DRAM	CASC[1:0] / Number of cycles	0	R/W	
		D3	**CASC0**	CAS cycles selection	1 1 4	0		
					1 0 3			
					0 1 2			
					0 0 1			
		D2	**–**	reserved	–	–	–	0 when being read.
		D1	**RASC1**	DRAM	RASC[1:0] / Number of cycles	0	R/W	
		D0	**RASC0**	RAS cycles selection	1 1 4	0		
					1 0 3			
					0 1 2			
					0 0 1			
Access control register	**0048132 (HW)**	DF	**A18IO**	Area 18, 17 internal/external access	1 Internal access / 0 External access	0	R/W	
		DE	**A16IO**	Area 16, 15 internal/external access		0	R/W	
		DD	**A14IO**	Area 14, 13 internal/external access		0	R/W	
		DC	**A12IO**	Area 12, 11 internal/external access		0	R/W	
		DB	**–**	reserved	–	0	–	0 when being read.
		DA	**A8IO**	Area 8, 7 internal/external access	1 Internal access / 0 External access	0	R/W	
		D9	**A6IO**	Area 6 internal/external access		0	R/W	
		D8	**A5IO**	Area 5, 4 internal/external access		0	R/W	
		D7	**A18EC**	Area 18, 17 endian control	1 Big endian / 0 Little endian	0	R/W	
		D6	**A16EC**	Area 16, 15 endian control		0	R/W	
		D5	**A14EC**	Area 14, 13 endian control		0	R/W	
		D4	**A12EC**	Area 12, 11 endian control		0	R/W	
		D3	**A10EC**	Area 10, 9 endian control		0	R/W	
		D2	**A8EC**	Area 8, 7 endian control		0	R/W	
		D1	**A6EC**	Area 6 endian control		0	R/W	
		D0	**A5EC**	Area 5, 4 endian control		0	R/W	
TTBR low-order register	**0048134 (HW)**	DF	**TTBR15**	Trap table base address [15:10]		0	R/W	
		DE	**TTBR14**			0		
		DD	**TTBR13**			0		
		DC	**TTBR12**			0		
		DB	**TTBR11**			0		
		DA	**TTBR10**			0		
		D9	**TTBR09**	Trap table base address [9:0]	Fixed at 0	0	R	0 when being read. Writing 1 not allowed.
		D8	**TTBR08**			0		
		D7	**TTBR07**			0		
		D6	**TTBR06**			0		
		D5	**TTBR05**			0		
		D4	**TTBR04**			0		
		D3	**TTBR03**			0		
		D2	**TTBR02**			0		
		D1	**TTBR01**			0		
		D0	**TTBR00**			0		
TTBR high-order register	**0048136 (HW)**	DF	**TTBR33**	Trap table base address [31:28]	Fixed at 0	0	R	0 when being read. Writing 1 not allowed.
		DE	**TTBR32**			0		
		DD	**TTBR31**			0		
		DC	**TTBR30**			0		
		DB	**TTBR2B**	Trap table base address [27:16]	The initial value is set according to the BTA3 pin status. BTA3 = "1": 0x008 BTA3 = "0": 0x0C0	←	R/W	
		DA	**TTBR2A**					
		D9	**TTBR29**					
		D8	**TTBR28**					
		D7	**TTBR27**					
		D6	**TTBR26**					
		D5	**TTBR25**					
		D4	**TTBR24**					
		D3	**TTBR23**					
		D2	**TTBR22**					
		D1	**TTBR21**					
		D0	**TTBR20**					

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
G/A read signal control register	**0048138 (HW)**	DF	**A18AS**	Area 18, 17 address strobe signal	1	Enabled	0	Disabled	0	R/W	
		DE	**A16AS**	Area 16, 15 address strobe signal					0	R/W	
		DD	**A14AS**	Area 14, 13 address strobe signal					0	R/W	
		DC	**A12AS**	Area 12, 11 address strobe signal					0	R/W	
		DB	**–**	reserved	–				0	–	0 when being read.
		DA	**A8AS**	Area 8, 7 address strobe signal	1	Enabled	0	Disabled	0	R/W	
		D9	**A6AS**	Area 6 address strobe signal					0	R/W	
		D8	**A5AS**	Area 5, 4 address strobe signal					0	R/W	
		D7	**A18RD**	Area 18, 17 read signal	1	Enabled	0	Disabled	0	R/W	
		D6	**A16RD**	Area 16, 15 read signal					0	R/W	
		D5	**A14RD**	Area 14, 13 read signal					0	R/W	
		D4	**A12RD**	Area 12, 11 read signal					0	R/W	
		D3	**–**	reserved	–				0	–	0 when being read.
		D2	**A8RD**	Area 8, 7 read signal	1	Enabled	0	Disabled	0	R/W	
		D1	**A6RD**	Area 6 read signal					0	R/W	
		D0	**A5RD**	Area 5, 4 read signal					0	R/W	
BCLK select register	**004813A (B)**	D7–4	**–**	reserved	–				0	–	0 when being read.
		D3	**A1X1MD**	Area 1 access-speed	1	2 cycles	0	4 cycles	0	R/W	x2 speed mode only
		D2	**–**	reserved	–				0	–	0 when being read.
		D1	**BCLKSEL1**	BCLK output clock selection	BCLKSEL[1:0]		BCLK		0	R/W	
		D0	**BCLKSEL0**		1	1	PLL_CLK		0		
					1	0	OSC3_CLK				
					0	1	BCU_CLK				
					0	0	CPU_CLK				

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 0	**0048180**	DF	**CR0A15**	16-bit timer 0 comparison data A	0 to 65535				X	R/W	
comparison	**(HW)**	DE	**CR0A14**	CR0A15 = MSB					X		
register A		DD	**CR0A13**	CR0A0 = LSB					X		
		DC	**CR0A12**						X		
		DB	**CR0A11**						X		
		DA	**CR0A10**						X		
		D9	**CR0A9**						X		
		D8	**CR0A8**						X		
		D7	**CR0A7**						X		
		D6	**CR0A6**						X		
		D5	**CR0A5**						X		
		D4	**CR0A4**						X		
		D3	**CR0A3**						X		
		D2	**CR0A2**						X		
		D1	**CR0A1**						X		
		D0	**CR0A0**						X		
16-bit timer 0	**0048182**	DF	**CR0B15**	16-bit timer 0 comparison data B	0 to 65535				X	R/W	
comparison	**(HW)**	DE	**CR0B14**	CR0B15 = MSB					X		
register B		DD	**CR0B13**	CR0B0 = LSB					X		
		DC	**CR0B12**						X		
		DB	**CR0B11**						X		
		DA	**CR0B10**						X		
		D9	**CR0B9**						X		
		D8	**CR0B8**						X		
		D7	**CR0B7**						X		
		D6	**CR0B6**						X		
		D5	**CR0B5**						X		
		D4	**CR0B4**						X		
		D3	**CR0B3**						X		
		D2	**CR0B2**						X		
		D1	**CR0B1**						X		
		D0	**CR0B0**						X		
16-bit timer 0	**0048184**	DF	**TC015**	16-bit timer 0 counter data	0 to 65535				X	R	
counter data	**(HW)**	DE	**TC014**	TC015 = MSB					X		
register		DD	**TC013**	TC00 = LSB					X		
		DC	**TC012**						X		
		DB	**TC011**						X		
		DA	**TC010**						X		
		D9	**TC09**						X		
		D8	**TC08**						X		
		D7	**TC07**						X		
		D6	**TC06**						X		
		D5	**TC05**						X		
		D4	**TC04**						X		
		D3	**TC03**						X		
		D2	**TC02**						X		
		D1	**TC01**						X		
		D0	**TC00**						X		
16-bit timer 0	**0048186**	D7	**–**	reserved	–				0	–	0 when being read.
control register	**(B)**	D6	**SELFM0**	16-bit timer 0 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB0**	16-bit timer 0 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV0**	16-bit timer 0 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL0**	16-bit timer 0 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM0**	16-bit timer 0 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET0**	16-bit timer 0 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN0**	16-bit timer 0 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 1 comparison register A	**0048188 (HW)**	DF	**CR1A15**	16-bit timer 1 comparison data A	0 to 65535				X	R/W	
		DE	**CR1A14**	CR1A15 = MSB					X		
		DD	**CR1A13**	CR1A0 = LSB					X		
		DC	**CR1A12**						X		
		DB	**CR1A11**						X		
		DA	**CR1A10**						X		
		D9	**CR1A9**						X		
		D8	**CR1A8**						X		
		D7	**CR1A7**						X		
		D6	**CR1A6**						X		
		D5	**CR1A5**						X		
		D4	**CR1A4**						X		
		D3	**CR1A3**						X		
		D2	**CR1A2**						X		
		D1	**CR1A1**						X		
		D0	**CR1A0**						X		
16-bit timer 1 comparison register B	**004818A (HW)**	DF	**CR1B15**	16-bit timer 1 comparison data B	0 to 65535				X	R/W	
		DE	**CR1B14**	CR1B15 = MSB					X		
		DD	**CR1B13**	CR1B0 = LSB					X		
		DC	**CR1B12**						X		
		DB	**CR1B11**						X		
		DA	**CR1B10**						X		
		D9	**CR1B9**						X		
		D8	**CR1B8**						X		
		D7	**CR1B7**						X		
		D6	**CR1B6**						X		
		D5	**CR1B5**						X		
		D4	**CR1B4**						X		
		D3	**CR1B3**						X		
		D2	**CR1B2**						X		
		D1	**CR1B1**						X		
		D0	**CR1B0**						X		
16-bit timer 1 counter data register	**004818C (HW)**	DF	**TC115**	16-bit timer 1 counter data	0 to 65535				X	R	
		DE	**TC114**	TC115 = MSB					X		
		DD	**TC113**	TC10 = LSB					X		
		DC	**TC112**						X		
		DB	**TC111**						X		
		DA	**TC110**						X		
		D9	**TC19**						X		
		D8	**TC18**						X		
		D7	**TC17**						X		
		D6	**TC16**						X		
		D5	**TC15**						X		
		D4	**TC14**						X		
		D3	**TC13**						X		
		D2	**TC12**						X		
		D1	**TC11**						X		
		D0	**TC10**						X		
16-bit timer 1 control register	**004818E (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM1**	16-bit timer 1 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB1**	16-bit timer 1 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV1**	16-bit timer 1 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL1**	16-bit timer 1 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM1**	16-bit timer 1 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET1**	16-bit timer 1 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN1**	16-bit timer 1 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 2 comparison register A	**0048190 (HW)**	DF	**CR2A15**	16-bit timer 2 comparison data A	0 to 65535				X	R/W	
		DE	**CR2A14**	CR2A15 = MSB					X		
		DD	**CR2A13**	CR2A0 = LSB					X		
		DC	**CR2A12**						X		
		DB	**CR2A11**						X		
		DA	**CR2A10**						X		
		D9	**CR2A9**						X		
		D8	**CR2A8**						X		
		D7	**CR2A7**						X		
		D6	**CR2A6**						X		
		D5	**CR2A5**						X		
		D4	**CR2A4**						X		
		D3	**CR2A3**						X		
		D2	**CR2A2**						X		
		D1	**CR2A1**						X		
		D0	**CR2A0**						X		
16-bit timer 2 comparison register B	**0048192 (HW)**	DF	**CR2B15**	16-bit timer 2 comparison data B	0 to 65535				X	R/W	
		DE	**CR2B14**	CR2B15 = MSB					X		
		DD	**CR2B13**	CR2B0 = LSB					X		
		DC	**CR2B12**						X		
		DB	**CR2B11**						X		
		DA	**CR2B10**						X		
		D9	**CR2B9**						X		
		D8	**CR2B8**						X		
		D7	**CR2B7**						X		
		D6	**CR2B6**						X		
		D5	**CR2B5**						X		
		D4	**CR2B4**						X		
		D3	**CR2B3**						X		
		D2	**CR2B2**						X		
		D1	**CR2B1**						X		
		D0	**CR2B0**						X		
16-bit timer 2 counter data register	**0048194 (HW)**	DF	**TC215**	16-bit timer 2 counter data	0 to 65535				X	R	
		DE	**TC214**	TC215 = MSB					X		
		DD	**TC213**	TC20 = LSB					X		
		DC	**TC212**						X		
		DB	**TC211**						X		
		DA	**TC210**						X		
		D9	**TC29**						X		
		D8	**TC28**						X		
		D7	**TC27**						X		
		D6	**TC26**						X		
		D5	**TC25**						X		
		D4	**TC24**						X		
		D3	**TC23**						X		
		D2	**TC22**						X		
		D1	**TC21**						X		
		D0	**TC20**						X		
16-bit timer 2 control register	**0048196 (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM2**	16-bit timer 2 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB2**	16-bit timer 2 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV2**	16-bit timer 2 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL2**	16-bit timer 2 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM2**	16-bit timer 2 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET2**	16-bit timer 2 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN2**	16-bit timer 2 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 3 comparison register A	**0048198 (HW)**	DF	**CR3A15**	16-bit timer 3 comparison data A	0 to 65535				X	R/W	
		DE	**CR3A14**	CR3A15 = MSB					X		
		DD	**CR3A13**	CR3A0 = LSB					X		
		DC	**CR3A12**						X		
		DB	**CR3A11**						X		
		DA	**CR3A10**						X		
		D9	**CR3A9**						X		
		D8	**CR3A8**						X		
		D7	**CR3A7**						X		
		D6	**CR3A6**						X		
		D5	**CR3A5**						X		
		D4	**CR3A4**						X		
		D3	**CR3A3**						X		
		D2	**CR3A2**						X		
		D1	**CR3A1**						X		
		D0	**CR3A0**						X		
16-bit timer 3 comparison register B	**004819A (HW)**	DF	**CR3B15**	16-bit timer 3 comparison data B	0 to 65535				X	R/W	
		DE	**CR3B14**	CR3B15 = MSB					X		
		DD	**CR3B13**	CR3B0 = LSB					X		
		DC	**CR3B12**						X		
		DB	**CR3B11**						X		
		DA	**CR3B10**						X		
		D9	**CR3B9**						X		
		D8	**CR3B8**						X		
		D7	**CR3B7**						X		
		D6	**CR3B6**						X		
		D5	**CR3B5**						X		
		D4	**CR3B4**						X		
		D3	**CR3B3**						X		
		D2	**CR3B2**						X		
		D1	**CR3B1**						X		
		D0	**CR3B0**						X		
16-bit timer 3 counter data register	**004819C (HW)**	DF	**TC315**	16-bit timer 3 counter data	0 to 65535				X	R	
		DE	**TC314**	TC315 = MSB					X		
		DD	**TC313**	TC30 = LSB					X		
		DC	**TC312**						X		
		DB	**TC311**						X		
		DA	**TC310**						X		
		D9	**TC39**						X		
		D8	**TC38**						X		
		D7	**TC37**						X		
		D6	**TC36**						X		
		D5	**TC35**						X		
		D4	**TC34**						X		
		D3	**TC33**						X		
		D2	**TC32**						X		
		D1	**TC31**						X		
		D0	**TC30**						X		
16-bit timer 3 control register	**004819E (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM3**	16-bit timer 3 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB3**	16-bit timer 3 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV3**	16-bit timer 3 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL3**	16-bit timer 3 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM3**	16-bit timer 3 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET3**	16-bit timer 3 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN3**	16-bit timer 3 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 4 comparison register A	**00481A0 (HW)**	DF	**CR4A15**	16-bit timer 4 comparison data A	0 to 65535				X	R/W	
		DE	**CR4A14**	CR4A15 = MSB					X		
		DD	**CR4A13**	CR4A0 = LSB					X		
		DC	**CR4A12**						X		
		DB	**CR4A11**						X		
		DA	**CR4A10**						X		
		D9	**CR4A9**						X		
		D8	**CR4A8**						X		
		D7	**CR4A7**						X		
		D6	**CR4A6**						X		
		D5	**CR4A5**						X		
		D4	**CR4A4**						X		
		D3	**CR4A3**						X		
		D2	**CR4A2**						X		
		D1	**CR4A1**						X		
		D0	**CR4A0**						X		
16-bit timer 4 comparison register B	**00481A2 (HW)**	DF	**CR4B15**	16-bit timer 4 comparison data B	0 to 65535				X	R/W	
		DE	**CR4B14**	CR4B15 = MSB					X		
		DD	**CR4B13**	CR4B0 = LSB					X		
		DC	**CR4B12**						X		
		DB	**CR4B11**						X		
		DA	**CR4B10**						X		
		D9	**CR4B9**						X		
		D8	**CR4B8**						X		
		D7	**CR4B7**						X		
		D6	**CR4B6**						X		
		D5	**CR4B5**						X		
		D4	**CR4B4**						X		
		D3	**CR4B3**						X		
		D2	**CR4B2**						X		
		D1	**CR4B1**						X		
		D0	**CR4B0**						X		
16-bit timer 4 counter data register	**00481A4 (HW)**	DF	**TC415**	16-bit timer 4 counter data	0 to 65535				X	R	
		DE	**TC414**	TC415 = MSB					X		
		DD	**TC413**	TC40 = LSB					X		
		DC	**TC412**						X		
		DB	**TC411**						X		
		DA	**TC410**						X		
		D9	**TC49**						X		
		D8	**TC48**						X		
		D7	**TC47**						X		
		D6	**TC46**						X		
		D5	**TC45**						X		
		D4	**TC44**						X		
		D3	**TC43**						X		
		D2	**TC42**						X		
		D1	**TC41**						X		
		D0	**TC40**						X		
16-bit timer 4 control register	**00481A6 (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM4**	16-bit timer 4 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB4**	16-bit timer 4 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV4**	16-bit timer 4 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL4**	16-bit timer 4 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM4**	16-bit timer 4 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET4**	16-bit timer 4 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN4**	16-bit timer 4 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 5 comparison register A	**00481A8 (HW)**	DF	**CR5A15**	16-bit timer 5 comparison data A	0 to 65535				X	R/W	
		DE	**CR5A14**	CR5A15 = MSB					X		
		DD	**CR5A13**	CR5A0 = LSB					X		
		DC	**CR5A12**						X		
		DB	**CR5A11**						X		
		DA	**CR5A10**						X		
		D9	**CR5A9**						X		
		D8	**CR5A8**						X		
		D7	**CR5A7**						X		
		D6	**CR5A6**						X		
		D5	**CR5A5**						X		
		D4	**CR5A4**						X		
		D3	**CR5A3**						X		
		D2	**CR5A2**						X		
		D1	**CR5A1**						X		
		D0	**CR5A0**						X		
16-bit timer 5 comparison register B	**00481AA (HW)**	DF	**CR5B15**	16-bit timer 5 comparison data B	0 to 65535				X	R/W	
		DE	**CR5B14**	CR5B15 = MSB					X		
		DD	**CR5B13**	CR5B0 = LSB					X		
		DC	**CR5B12**						X		
		DB	**CR5B11**						X		
		DA	**CR5B10**						X		
		D9	**CR5B9**						X		
		D8	**CR5B8**						X		
		D7	**CR5B7**						X		
		D6	**CR5B6**						X		
		D5	**CR5B5**						X		
		D4	**CR5B4**						X		
		D3	**CR5B3**						X		
		D2	**CR5B2**						X		
		D1	**CR5B1**						X		
		D0	**CR5B0**						X		
16-bit timer 5 counter data register	**00481AC (HW)**	DF	**TC515**	16-bit timer 5 counter data	0 to 65535				X	R	
		DE	**TC514**	TC515 = MSB					X		
		DD	**TC513**	TC50 = LSB					X		
		DC	**TC512**						X		
		DB	**TC511**						X		
		DA	**TC510**						X		
		D9	**TC59**						X		
		D8	**TC58**						X		
		D7	**TC57**						X		
		D6	**TC56**						X		
		D5	**TC55**						X		
		D4	**TC54**						X		
		D3	**TC53**						X		
		D2	**TC52**						X		
		D1	**TC51**						X		
		D0	**TC50**						X		
16-bit timer 5 control register	**00481AE (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM5**	16-bit timer 5 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB5**	16-bit timer 5 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV5**	16-bit timer 5 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL5**	16-bit timer 5 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM5**	16-bit timer 5 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET5**	16-bit timer 5 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN5**	16-bit timer 5 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
IDMA base address low-order register	**0048200 (HW)**	DF	**DBASEL15**	IDMA base address low-order 16 bits (Initial value: 0x0C003A0)					0	R/W	
		DE	**DBASEL14**						0		
		DD	**DBASEL13**						0		
		DC	**DBASEL12**						0		
		DB	**DBASEL11**						0		
		DA	**DBASEL10**						0		
		D9	**DBASEL9**						1		
		D8	**DBASEL8**						1		
		D7	**DBASEL7**						1		
		D6	**DBASEL6**						0		
		D5	**DBASEL5**						1		
		D4	**DBASEL4**						0		
		D3	**DBASEL3**						0		
		D2	**DBASEL2**						0		
		D1	**DBASEL1**						0		
		D0	**DBASEL0**						0		
IDMA base address high-order register	**0048202 (HW)**	DF–C	**–**	reserved	–				–	–	Undefined in read.
		DB	**DBASEH11**	IDMA base address high-order 12 bits (Initial value: 0x0C003A0)					0	R/W	
		DA	**DBASEH10**						0		
		D9	**DBASEH9**						0		
		D8	**DBASEH8**						0		
		D7	**DBASEH7**						1		
		D6	**DBASEH6**						1		
		D5	**DBASEH5**						0		
		D4	**DBASEH4**						0		
		D3	**DBASEH3**						0		
		D2	**DBASEH2**						0		
		D1	**DBASEH1**						0		
		D0	**DBASEH0**						0		
IDMA start register	**0048204 (B)**	D7	**DSTART**	IDMA start	1	IDMA start	0	Stop	0	R/W	
		D6–0	**DCHN**	IDMA channel number	0 to 127				0	R/W	
IDMA enable register	**0048205 (B)**	D7–1	**–**	reserved	–				–	–	
		D0	**IDMAEN**	IDMA enable	1	Enabled	0	Disabled	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
High-speed DMA Ch.0 transfer counter register	**0048220 (HW)**	DF	**TC0_L7**	Ch.0 transfer counter[7:0]					X	R/W	
		DE	**TC0_L6**	(block transfer mode)					X		
		DD	**TC0_L5**						X		
		DC	**TC0_L4**	Ch.0 transfer counter[15:8]					X		
		DB	**TC0_L3**	(single/successive transfer mode)					X		
		DA	**TC0_L2**						X		
		D9	**TC0_L1**						X		
		D8	**TC0_L0**						X		
		D7	**BLKLEN07**	Ch.0 block length					X	R/W	
		D6	**BLKLEN06**	(block transfer mode)					X		
		D5	**BLKLEN05**						X		
		D4	**BLKLEN04**	Ch.0 transfer counter[7:0]					X		
		D3	**BLKLEN03**	(single/successive transfer mode)					X		
		D2	**BLKLEN02**						X		
		D1	**BLKLEN01**						X		
		D0	**BLKLEN00**						X		
High-speed DMA Ch.0 control register	**0048222 (HW)**	DF	**DUALM0**	Ch.0 address mode selection	1	Dual addr	0	Single addr	0	R/W	
		DE	**D0DIR**	D) Invalid	–				–	–	
				S) Ch.0 transfer direction control	1	Memory WR	0	Memory RD	0	R/W	
		DD–8	–	reserved	–				–	–	Undefined in read.
Note:		D7	**TC0_H7**	Ch.0 transfer counter[15:8]					X	R/W	
D) Dual address mode		D6	**TC0_H6**	(block transfer mode)					X		
S) Single address mode		D5	**TC0_H5**						X		
		D4	**TC0_H4**	Ch.0 transfer counter[23:16]					X		
		D3	**TC0_H3**	(single/successive transfer mode)					X		
		D2	**TC0_H2**						X		
		D1	**TC0_H1**						X		
		D0	**TC0_H0**						X		
High-speed DMA Ch.0 low-order source address set-up register	**0048224 (HW)**	DF	**S0ADRL15**	D) Ch.0 source address[15:0]					X	R/W	
		DE	**S0ADRL14**	S) Ch.0 memory address[15:0]					X		
		DD	**S0ADRL13**						X		
		DC	**S0ADRL12**						X		
		DB	**S0ADRL11**						X		
		DA	**S0ADRL10**						X		
Note:		D9	**S0ADRL9**						X		
D) Dual address mode		A8	**S0ADRL8**						X		
S) Single address mode		D7	**S0ADRL7**						X		
		D6	**S0ADRL6**						X		
		D5	**S0ADRL5**						X		
		D4	**S0ADRL4**						X		
		D3	**S0ADRL3**						X		
		D2	**S0ADRL2**						X		
		D1	**S0ADRL1**						X		
		D0	**S0ADRL0**						X		
High-speed DMA Ch.0 high-order source address set-up register	**0048226 (HW)**	DF	**DINTEN0**	Ch.0 interrupt enable	1	Enabled	0	Disabled	0	R/W	
		DE	**DATSIZE0**	Ch.0 transfer data size	1	Half word	0	Byte	0	R/W	
		DD	**S0IN1**	D) Ch.0 source address control	S0IN[1:0]		Inc/dec		0	R/W	
		DC	**S0IN0**	S) Ch.0 memory address control	1	1	Inc.(no init)		0		
					1	0	Inc.(init)				
					0	1	Dec.(no init)				
Note:					0	0	Fixed				
D) Dual address mode		DB	**S0ADRH11**	D) Ch.0 source address[27:16]					X	R/W	
S) Single address mode		DA	**S0ADRH10**	S) Ch.0 memory address[27:16]					X		
		D9	**S0ADRH9**						X		
		A8	**S0ADRH8**						X		
		D7	**S0ADRH7**						X		
		D6	**S0ADRH6**						X		
		D5	**S0ADRH5**						X		
		D4	**S0ADRH4**						X		
		D3	**S0ADRH3**						X		
		D2	**S0ADRH2**						X		
		D1	**S0ADRH1**						X		
		D0	**S0ADRH0**						X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
High-speed DMA Ch.0 low-order destination address set-up register	**0048228 (HW)**	DF	**D0ADRL15**	D) Ch.0 destination address[15:0]					X	R/W	
Note:		DE	**D0ADRL14**	S) Invalid					X		
D) Dual address mode		DD	**D0ADRL13**						X		
S) Single address mode		DC	**D0ADRL12**						X		
		DB	**D0ADRL11**						X		
		DA	**D0ADRL10**						X		
		D9	**D0ADRL9**						X		
		A8	**D0ADRL8**						X		
		D7	**D0ADRL7**						X		
		D6	**D0ADRL6**						X		
		D5	**D0ADRL5**						X		
		D4	**D0ADRL4**						X		
		D3	**D0ADRL3**						X		
		D2	**D0ADRL2**						X		
		D1	**D0ADRL1**						X		
		D0	**D0ADRL0**						X		
High-speed DMA Ch.0 high-order destination address set-up register	**004822A (HW)**	DF	**D0MOD1**	Ch.0 transfer mode	D0MOD[1:0]		Mode		0	R/W	
Note:		DE	**D0MOD0**		1	1	Invalid		0		
D) Dual address mode					1	0	Block				
S) Single address mode					0	1	Successive				
					0	0	Single				
		DD	**D0IN1**	D) Ch.0 destination address control	D0IN[1:0]		Inc/dec		0	R/W	
		DC	**D0IN0**	S) Invalid	1	1	Inc.(no init)		0		
					1	0	Inc.(init)				
					0	1	Dec.(no init)				
					0	0	Fixed				
		DB	**D0ADRH11**	D) Ch.0 destination address[27:16]					X	R/W	
		DA	**D0ADRH10**	S) Invalid					X		
		D9	**D0ADRH9**						X		
		A8	**D0ADRH8**						X		
		D7	**D0ADRH7**						X		
		D6	**D0ADRH6**						X		
		D5	**D0ADRH5**						X		
		D4	**D0ADRH4**						X		
		D3	**D0ADRH3**						X		
		D2	**D0ADRH2**						X		
		D1	**D0ADRH1**						X		
		D0	**D0ADRH0**						X		
High-speed DMA Ch.0 enable register	**004822C (HW)**	DF–1	–	reserved	–				–	–	Undefined in read.
		D0	**HS0_EN**	Ch.0 enable	1	Enable	0	Disable	0	R/W	
High-speed DMA Ch.0 trigger flag register	**004822E (HW)**	DF–1	–	reserved	–				–	–	Undefined in read.
		D0	**HS0_TF**	Ch.0 trigger flag clear (writing)	1	Clear	0	No operation	0	R/W	
				Ch.0 trigger flag status (reading)	1	Set	0	Cleared			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048230**	DF	**TC1_L7**	Ch.1 transfer counter[7:0]		X	R/W	
DMA Ch.1	**(HW)**	DE	**TC1_L6**	(block transfer mode)		X		
transfer		DD	**TC1_L5**			X		
counter		DC	**TC1_L4**	Ch.1 transfer counter[15:8]		X		
register		DB	**TC1_L3**	(single/successive transfer mode)		X		
		DA	**TC1_L2**			X		
		D9	**TC1_L1**			X		
		D8	**TC1_L0**			X		
		D7	**BLKLEN17**	Ch.1 block length		X	R/W	
		D6	**BLKLEN16**	(block transfer mode)		X		
		D5	**BLKLEN15**			X		
		D4	**BLKLEN14**	Ch.1 transfer counter[7:0]		X		
		D3	**BLKLEN13**	(single/successive transfer mode)		X		
		D2	**BLKLEN12**			X		
		D1	**BLKLEN11**			X		
		D0	**BLKLEN10**			X		
High-speed	**0048232**	DF	**DUALM1**	Ch.1 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
DMA Ch.1	**(HW)**	DE	**D1DIR**	D) Invalid	–	–	–	
control register				S) Ch.1 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	**–**	reserved	–	–	–	Undefined in read.
Note:		D7	**TC1_H7**	Ch.1 transfer counter[15:8]		X	R/W	
D) Dual address mode		D6	**TC1_H6**	(block transfer mode)		X		
S) Single address mode		D5	**TC1_H5**			X		
		D4	**TC1_H4**	Ch.1 transfer counter[23:16]		X		
		D3	**TC1_H3**	(single/successive transfer mode)		X		
		D2	**TC1_H2**			X		
		D1	**TC1_H1**			X		
		D0	**TC1_H0**			X		
High-speed	**0048234**	DF	**S1ADRL15**	D) Ch.1 source address[15:0]		X	R/W	
DMA Ch.1	**(HW)**	DE	**S1ADRL14**	S) Ch.1 memory address[15:0]		X		
low-order		DD	**S1ADRL13**			X		
source address		DC	**S1ADRL12**			X		
set-up register		DB	**S1ADRL11**			X		
		DA	**S1ADRL10**			X		
Note:		D9	**S1ADRL9**			X		
D) Dual address mode		A8	**S1ADRL8**			X		
S) Single address mode		D7	**S1ADRL7**			X		
		D6	**S1ADRL6**			X		
		D5	**S1ADRL5**			X		
		D4	**S1ADRL4**			X		
		D3	**S1ADRL3**			X		
		D2	**S1ADRL2**			X		
		D1	**S1ADRL1**			X		
		D0	**S1ADRL0**			X		
High-speed	**0048236**	DF	**DINTEN1**	Ch.1 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
DMA Ch.1	**(HW)**	DE	**DATSIZE1**	Ch.1 transfer data size	1 Half word / 0 Byte	0	R/W	
high-order		DD	**S1IN1**	D) Ch.1 source address control	S1IN[1:0] — Inc/dec	0	R/W	
source address		DC	**S1IN0**	S) Ch.1 memory address control	1 1 — Inc.(no init)	0		
set-up register					1 0 — Inc.(init)			
					0 1 — Dec.(no init)			
Note:					0 0 — Fixed			
D) Dual address mode		DB	**S1ADRH11**	D) Ch.1 source address[27:16]		X	R/W	
S) Single address mode		DA	**S1ADRH10**	S) Ch.1 memory address[27:16]		X		
		D9	**S1ADRH9**			X		
		A8	**S1ADRH8**			X		
		D7	**S1ADRH7**			X		
		D6	**S1ADRH6**			X		
		D5	**S1ADRH5**			X		
		D4	**S1ADRH4**			X		
		D3	**S1ADRH3**			X		
		D2	**S1ADRH2**			X		
		D1	**S1ADRH1**			X		
		D0	**S1ADRH0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048238**	DF	**D1ADRL15**	D) Ch.1 destination address[15:0]		X	R/W	
DMA Ch.1	**(HW)**	DE	**D1ADRL14**	S) Invalid		X		
low-order		DD	**D1ADRL13**			X		
destination		DC	**D1ADRL12**			X		
address set-up		DB	**D1ADRL11**			X		
register		DA	**D1ADRL10**			X		
		D9	**D1ADRL9**			X		
Note:		A8	**D1ADRL8**			X		
D) Dual address mode		D7	**D1ADRL7**			X		
S) Single address mode		D6	**D1ADRL6**			X		
		D5	**D1ADRL5**			X		
		D4	**D1ADRL4**			X		
		D3	**D1ADRL3**			X		
		D2	**D1ADRL2**			X		
		D1	**D1ADRL1**			X		
		D0	**D1ADRL0**			X		
High-speed	**004823A**	DF	**D1MOD1**	Ch.1 transfer mode	D1MOD[1:0] / Mode	0	R/W	
DMA Ch.1	**(HW)**	DE	**D1MOD0**		1 1 Invalid	0		
high-order					1 0 Block			
destination					0 1 Successive			
address set-up					0 0 Single			
register		DD	**D1IN1**	D) Ch.1 destination address	D1IN[1:0] / Inc/dec	0	R/W	
		DC	**D1IN0**	control	1 1 Inc.(no init)	0		
Note:				S) Invalid	1 0 Inc.(init)			
D) Dual address mode					0 1 Dec.(no init)			
S) Single address mode					0 0 Fixed			
		DB	**D1ADRH11**	D) Ch.1 destination		X	R/W	
		DA	**D1ADRH10**	address[27:16]		X		
		D9	**D1ADRH9**	S) Invalid		X		
		A8	**D1ADRH8**			X		
		D7	**D1ADRH7**			X		
		D6	**D1ADRH6**			X		
		D5	**D1ADRH5**			X		
		D4	**D1ADRH4**			X		
		D3	**D1ADRH3**			X		
		D2	**D1ADRH2**			X		
		D1	**D1ADRH1**			X		
		D0	**D1ADRH0**			X		
High-speed DMA Ch.1	**004823C (HW)**	DF–1	**–**	reserved	–	–	–	Undefined in read.
enable register		D0	**HS1_EN**	Ch.1 enable	1 Enable / 0 Disable	0	R/W	
High-speed DMA Ch.1	**004823E (HW)**	DF–1	**–**	reserved	–	–	–	Undefined in read.
trigger flag		D0	**HS1_TF**	Ch.1 trigger flag clear (writing)	1 Clear / 0 No operation	0	R/W	
register				Ch.1 trigger flag status (reading)	1 Set / 0 Cleared			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed DMA Ch.2 transfer counter register	**0048240 (HW)**	DF	**TC2_L7**	Ch.2 transfer counter[7:0]		X	R/W	
		DE	**TC2_L6**	(block transfer mode)		X		
		DD	**TC2_L5**			X		
		DC	**TC2_L4**	Ch.2 transfer counter[15:8]		X		
		DB	**TC2_L3**	(single/successive transfer mode)		X		
		DA	**TC2_L2**			X		
		D9	**TC2_L1**			X		
		D8	**TC2_L0**			X		
		D7	**BLKLEN27**	Ch.2 block length		X	R/W	
		D6	**BLKLEN26**	(block transfer mode)		X		
		D5	**BLKLEN25**			X		
		D4	**BLKLEN24**	Ch.2 transfer counter[7:0]		X		
		D3	**BLKLEN23**	(single/successive transfer mode)		X		
		D2	**BLKLEN22**			X		
		D1	**BLKLEN21**			X		
		D0	**BLKLEN20**			X		
High-speed DMA Ch.2 control register	**0048242 (HW)**	DF	**DUALM2**	Ch.2 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
		DE	**D2DIR**	D) Invalid	–	–	–	
				S) Ch.2 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	–	reserved	–	–	–	Undefined in read.
Note: D) Dual address mode S) Single address mode		D7	**TC2_H7**	Ch.2 transfer counter[15:8]		X	R/W	
		D6	**TC2_H6**	(block transfer mode)		X		
		D5	**TC2_H5**			X		
		D4	**TC2_H4**	Ch.2 transfer counter[23:16]		X		
		D3	**TC2_H3**	(single/successive transfer mode)		X		
		D2	**TC2_H2**			X		
		D1	**TC2_H1**			X		
		D0	**TC2_H0**			X		
High-speed DMA Ch.2 low-order source address set-up register	**0048244 (HW)**	DF	**S2ADRL15**	D) Ch.2 source address[15:0]		X	R/W	
		DE	**S2ADRL14**	S) Ch.2 memory address[15:0]		X		
		DD	**S2ADRL13**			X		
		DC	**S2ADRL12**			X		
		DB	**S2ADRL11**			X		
		DA	**S2ADRL10**			X		
Note: D) Dual address mode S) Single address mode		D9	**S2ADRL9**			X		
		A8	**S2ADRL8**			X		
		D7	**S2ADRL7**			X		
		D6	**S2ADRL6**			X		
		D5	**S2ADRL5**			X		
		D4	**S2ADRL4**			X		
		D3	**S2ADRL3**			X		
		D2	**S2ADRL2**			X		
		D1	**S2ADRL1**			X		
		D0	**S2ADRL0**			X		
High-speed DMA Ch.2 high-order source address set-up register	**0048246 (HW)**	DF	**DINTEN2**	Ch.2 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
		DE	**DATSIZE2**	Ch.2 transfer data size	1 Half word / 0 Byte	0	R/W	
		DD	**S2IN1**	D) Ch.2 source address control	S2IN[1:0] : Inc/dec	0	R/W	
		DC	**S2IN0**	S) Ch.2 memory address control	1 1 : Inc.(no init)	0		
					1 0 : Inc.(init)			
					0 1 : Dec.(no init)			
Note: D) Dual address mode S) Single address mode					0 0 : Fixed			
		DB	**S2ADRH11**	D) Ch.2 source address[27:16]		X	R/W	
		DA	**S2ADRH10**	S) Ch.2 memory address[27:16]		X		
		D9	**S2ADRH9**			X		
		A8	**S2ADRH8**			X		
		D7	**S2ADRH7**			X		
		D6	**S2ADRH6**			X		
		D5	**S2ADRH5**			X		
		D4	**S2ADRH4**			X		
		D3	**S2ADRH3**			X		
		D2	**S2ADRH2**			X		
		D1	**S2ADRH1**			X		
		D0	**S2ADRH0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048248**	DF	**D2ADRL15**	D) Ch.2 destination address[15:0]		X	R/W	
DMA Ch.2	**(HW)**	DE	**D2ADRL14**	S) Invalid		X		
low-order		DD	**D2ADRL13**			X		
destination		DC	**D2ADRL12**			X		
address set-up		DB	**D2ADRL11**			X		
register		DA	**D2ADRL10**			X		
		D9	**D2ADRL9**			X		
Note:		A8	**D2ADRL8**			X		
D) Dual address mode		D7	**D2ADRL7**			X		
S) Single address mode		D6	**D2ADRL6**			X		
		D5	**D2ADRL5**			X		
		D4	**D2ADRL4**			X		
		D3	**D2ADRL3**			X		
		D2	**D2ADRL2**			X		
		D1	**D2ADRL1**			X		
		D0	**D2ADRL0**			X		
High-speed	**004824A**	DF	**D2MOD1**	Ch.2 transfer mode	D2MOD[1:0] / Mode	0	R/W	
DMA Ch.2	**(HW)**	DE	**D2MOD0**		1 1 Invalid	0		
high-order					1 0 Block			
destination					0 1 Successive			
address set-up					0 0 Single			
register		DD	**D2IN1**	D) Ch.2 destination address	D2IN[1:0] / Inc/dec	0	R/W	
		DC	**D2IN0**	control	1 1 Inc.(no init)	0		
Note:				S) Invalid	1 0 Inc.(init)			
D) Dual address mode					0 1 Dec.(no init)			
S) Single address mode					0 0 Fixed			
		DB	**D2ADRH11**	D) Ch.2 destination		X	R/W	
		DA	**D2ADRH10**	address[27:16]		X		
		D9	**D2ADRH9**	S) Invalid		X		
		A8	**D2ADRH8**			X		
		D7	**D2ADRH7**			X		
		D6	**D2ADRH6**			X		
		D5	**D2ADRH5**			X		
		D4	**D2ADRH4**			X		
		D3	**D2ADRH3**			X		
		D2	**D2ADRH2**			X		
		D1	**D2ADRH1**			X		
		D0	**D2ADRH0**			X		
High-speed DMA Ch.2 enable register	**004824C (HW)**	DF–1	–	reserved	–	–	–	Undefined in read.
		D0	**HS2_EN**	Ch.2 enable	1 Enable / 0 Disable	0	R/W	
High-speed DMA Ch.2 trigger flag register	**004824E (HW)**	DF–1	–	reserved	–	–	–	Undefined in read.
		D0	**HS2_TF**	Ch.2 trigger flag clear (writing)	1 Clear / 0 No operation	0	R/W	
				Ch.2 trigger flag status (reading)	1 Set / 0 Cleared			

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
High-speed DMA Ch.3 transfer counter register	**0048250 (HW)**	DF	**TC3_L7**	Ch.3 transfer counter[7:0] (block transfer mode)					X	R/W	
		DE	**TC3_L6**						X		
		DD	**TC3_L5**						X		
		DC	**TC3_L4**	Ch.3 transfer counter[15:8] (single/successive transfer mode)					X		
		DB	**TC3_L3**						X		
		DA	**TC3_L2**						X		
		D9	**TC3_L1**						X		
		D8	**TC3_L0**						X		
		D7	**BLKLEN37**	Ch.3 block length (block transfer mode)					X	R/W	
		D6	**BLKLEN36**						X		
		D5	**BLKLEN35**						X		
		D4	**BLKLEN34**	Ch.3 transfer counter[7:0] (single/successive transfer mode)					X		
		D3	**BLKLEN33**						X		
		D2	**BLKLEN32**						X		
		D1	**BLKLEN31**						X		
		D0	**BLKLEN30**						X		
High-speed DMA Ch.3 control register	**0048252 (HW)**	DF	**DUALM3**	Ch.3 address mode selection	1	Dual addr	0	Single addr	0	R/W	
		DE	**D3DIR**	D) Invalid	–				–	–	
				S) Ch.3 transfer direction control	1	Memory WR	0	Memory RD	0	R/W	
		DD–8	–	reserved	–				–	–	Undefined in read.
Note: D) Dual address mode S) Single address mode		D7	**TC3_H7**	Ch.3 transfer counter[15:8] (block transfer mode)					X	R/W	
		D6	**TC3_H6**						X		
		D5	**TC3_H5**						X		
		D4	**TC3_H4**	Ch.3 transfer counter[23:16] (single/successive transfer mode)					X		
		D3	**TC3_H3**						X		
		D2	**TC3_H2**						X		
		D1	**TC3_H1**						X		
		D0	**TC3_H0**						X		
High-speed DMA Ch.3 low-order source address set-up register	**0048254 (HW)**	DF	**S3ADRL15**	D) Ch.3 source address[15:0] S) Ch.3 memory address[15:0]					X	R/W	
		DE	**S3ADRL14**						X		
		DD	**S3ADRL13**						X		
		DC	**S3ADRL12**						X		
		DB	**S3ADRL11**						X		
		DA	**S3ADRL10**						X		
Note: D) Dual address mode S) Single address mode		D9	**S3ADRL9**						X		
		A8	**S3ADRL8**						X		
		D7	**S3ADRL7**						X		
		D6	**S3ADRL6**						X		
		D5	**S3ADRL5**						X		
		D4	**S3ADRL4**						X		
		D3	**S3ADRL3**						X		
		D2	**S3ADRL2**						X		
		D1	**S3ADRL1**						X		
		D0	**S3ADRL0**						X		
High-speed DMA Ch.3 high-order source address set-up register	**0048256 (HW)**	DF	**DINTEN3**	Ch.3 interrupt enable	1	Enabled	0	Disabled	0	R/W	
		DE	**DATSIZE3**	Ch.3 transfer data size	1	Half word	0	Byte	0	R/W	
		DD	**S3IN1**	D) Ch.3 source address control	S3IN[1:0]		Inc/dec		0	R/W	
		DC	**S3IN0**	S) Ch.3 memory address control	1	1	Inc.(no init)		0		
					1	0	Inc.(init)				
					0	1	Dec.(no init)				
Note: D) Dual address mode S) Single address mode					0	0	Fixed				
		DB	**S3ADRH11**	D) Ch.3 source address[27:16] S) Ch.3 memory address[27:16]					X	R/W	
		DA	**S3ADRH10**						X		
		D9	**S3ADRH9**						X		
		A8	**S3ADRH8**						X		
		D7	**S3ADRH7**						X		
		D6	**S3ADRH6**						X		
		D5	**S3ADRH5**						X		
		D4	**S3ADRH4**						X		
		D3	**S3ADRH3**						X		
		D2	**S3ADRH2**						X		
		D1	**S3ADRH1**						X		
		D0	**S3ADRH0**						X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048258**	DF	**D3ADRL15**	D) Ch.3 destination address[15:0]		X	R/W	
DMA Ch.3	**(HW)**	DE	**D3ADRL14**	S) Invalid		X		
low-order		DD	**D3ADRL13**			X		
destination		DC	**D3ADRL12**			X		
address set-up		DB	**D3ADRL11**			X		
register		DA	**D3ADRL10**			X		
		D9	**D3ADRL9**			X		
Note:		A8	**D3ADRL8**			X		
D) Dual address mode		D7	**D3ADRL7**			X		
S) Single address mode		D6	**D3ADRL6**			X		
		D5	**D3ADRL5**			X		
		D4	**D3ADRL4**			X		
		D3	**D3ADRL3**			X		
		D2	**D3ADRL2**			X		
		D1	**D3ADRL1**			X		
		D0	**D3ADRL0**			X		
High-speed	**004825A**	DF	**D3MOD1**	Ch.3 transfer mode	D3MOD[1:0] / Mode	0	R/W	
DMA Ch.3	**(HW)**	DE	**D3MOD0**		1 1 Invalid	0		
high-order					1 0 Block			
destination					0 1 Successive			
address set-up					0 0 Single			
register		DD	**D3IN1**	D) Ch.3 destination address	D3IN[1:0] / Inc/dec	0	R/W	
		DC	**D3IN0**	control	1 1 Inc.(no init)	0		
Note:				S) Invalid	1 0 Inc.(init)			
D) Dual address mode					0 1 Dec.(no init)			
S) Single address mode					0 0 Fixed			
		DB	**D3ADRH11**	D) Ch.3 destination		X	R/W	
		DA	**D3ADRH10**	address[27:16]		X		
		D9	**D3ADRH9**	S) Invalid		X		
		A8	**D3ADRH8**			X		
		D7	**D3ADRH7**			X		
		D6	**D3ADRH6**			X		
		D5	**D3ADRH5**			X		
		D4	**D3ADRH4**			X		
		D3	**D3ADRH3**			X		
		D2	**D3ADRH2**			X		
		D1	**D3ADRH1**			X		
		D0	**D3ADRH0**			X		
High-speed DMA Ch.3 enable register	**004825C (HW)**	DF–1	–	reserved	–	–	–	Undefined in read.
		D0	**HS3_EN**	Ch.3 enable	1 Enable / 0 Disable	0	R/W	
High-speed DMA Ch.3 trigger flag register	**004825E (HW)**	DF–1	–	reserved	–	–	–	Undefined in read.
		D0	**HS3_TF**	Ch.3 trigger flag clear (writing)	1 Clear / 0 No operation	0	R/W	
				Ch.3 trigger flag status (reading)	1 Set / 0 Cleared			

5 Power-Down Control

This chapter describes the controls used to reduce power consumption of the device.

Points on power saving

The current consumption of the device varies greatly with the CPU's operation mode, the system clocks used, and the peripheral circuits operated.

Current consumption	low←					→high
CPU/BCU	SLEEP	HALT2	Operating	HALT2	HALT(basic)	Operating
System clock	–	OSC1	OSC1	OSC3	OSC3	OSC3
OSC3 oscillation circuit	OFF	OFF	OFF	ON	ON	ON
Prescaler/peripheral circuit	STOP					RUN

To reduce power consumption of the device, it is important that as many unnecessary circuits as possible be turned off. In particular, peripheral circuits operating at a fast-clock rate consume a large amount of current, so design the program so that these circuits are turned off whenever unnecessary.

Power-saving in standby modes

When CPU processing is unnecessary, such as when waiting for an interrupt from key entries or peripheral circuits, place the device in standby mode to reduce current consumption.

Standby mode	Method to enter the mode	Circuits/functions stopped
Basic HALT mode	Execute the halt instruction after setting HLT2OP (D3)/Clock option register (0x40190) to "0". When the #BUSREQ signal is asserted from an external bus master while SEPD (D1)/Bus control register (0x4812E) = "1".	CPU only
HALT2 mode	Execute the halt instruction after setting HLT2OP to "1".	CPU, BCU, bus clock, and DMA
SLEEP mode	Execute the slp instruction.	CPU, BCU, bus clock, DMA, high-speed (OSC3) oscillation circuit, prescaler, and peripheral circuits that use the prescaler output clocks

HLT2OP (D3)/Clock option register (0x40190) that is used to select a HALT mode is set to "0" (basic HALT mode) at initial reset.

Notes: • In systems in which DRAM is connected directly to the device, the refresh function is turned off during HALT2 and SLEEP modes.

• The standby mode is cleared by interrupt generation (except for the basic HALT mode, which is set using an external bus master). Therefore, before entering standby mode, set the related registers to allow an interrupt to be used to clear the standby mode to be generated.

The low-speed (OSC1) oscillation circuit and clock timer continue operating even during SLEEP mode. If they are unnecessary, these circuits can also be turned off.

Function	Control bit	"1"	"0"	Default
Low-speed (OSC1) oscillation ON/OFF control	SOSC1(D0)/ Power control register(0x40180)	ON	OFF	ON

Switching over the system clocks

Normally, the system is clocked by the high-speed (OSC3) oscillation clock. If high-speed operation is unnecessary, switch the system clock to the low-speed (OSC1) oscillation clock and turn off the high-speed (OSC3) oscillation circuit. This helps to reduce current consumption. However, if DRAM is connected directly to the device, note that the refresh function is also turned off.
Even during operation using the high-speed (OSC3) oscillation clock, power reduction can also be achieved through the use of a system clock derived from the OSC3 clock by dividing it (1/1, 1/2, 1/4, or 1/8).

Function	Control bit	"1"	"0"	Default
System clock switch over	CLKCHG(D2)/ Power control register(0x40180)	OSC3	OSC1	OSC3
High-speed (OSC3) oscillation ON/OFF control	SOSC3(D1)/ Power control register(0x40180)	ON	OFF	ON
System clock division ratio selection	CLKDT(D[7:6])/ Power control register(0x40180)	"11" = 1/8 "10" = 1/4 "01" = 1/2 "00" = 1/1		1/1

Turning off the prescaler and peripheral circuits

Current consumption can be reduced by turning off the peripheral circuits operating at high speed as much as possible. The circuits listed below are operated using a clock generated by the prescaler:

• 16-bit programmable timers 0 to 5 (watchdog timer)
• 8-bit programmable timers 0 to 3 (DRAM refresh, serial interface)
• A/D converter

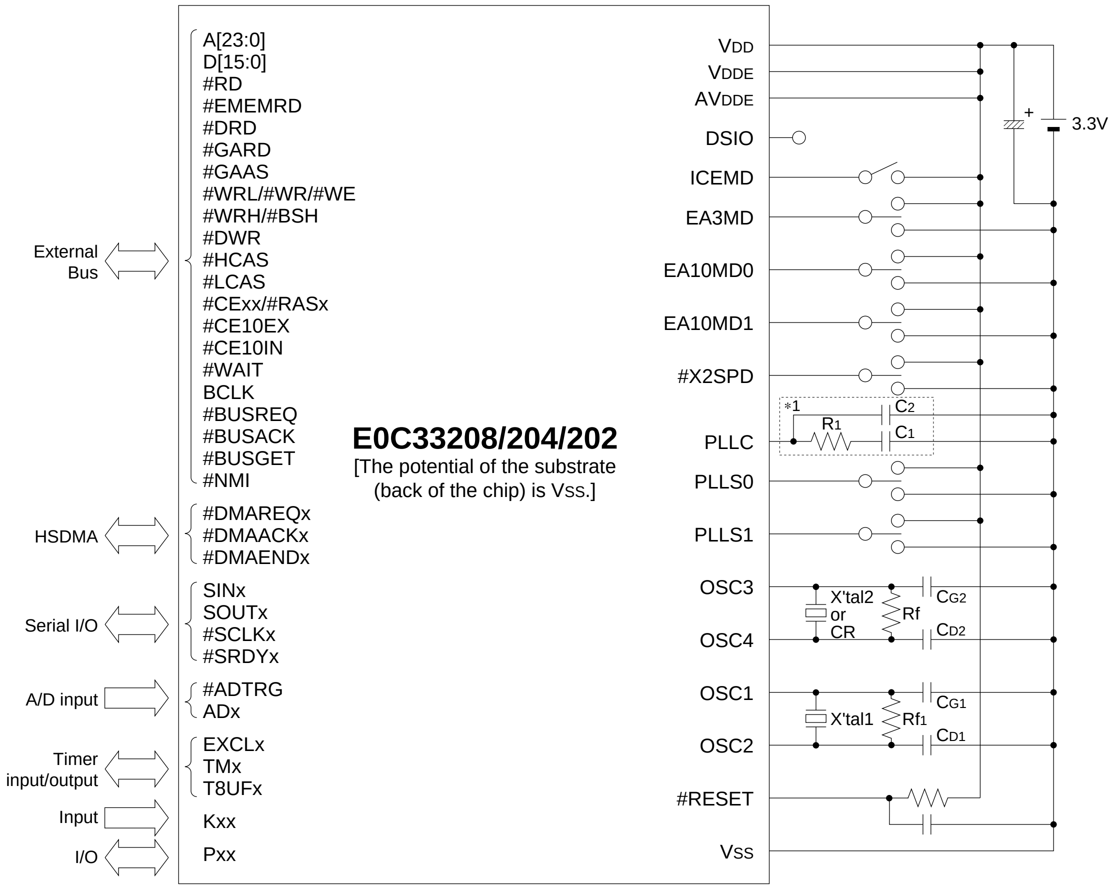
If none of these circuits need to be used, turn off the prescaler. If some of these circuits need to be used, turn off all other unnecessary circuits and stop clock supply from the prescaler to those circuits.

Function	Control bit	"1"	"0"	Default
Prescaler ON/OFF	PSCON(D5)/Power control register(0x40180)	ON	OFF	ON
16-bit timer 0 clock control	P16TON0(D3)/16-bit timer 0 clock control register(0x40147)	ON	OFF	OFF
16-bit timer 0 Run/Stop	PRUN0(D0)/16-bit timer 0 control register(0x48186)	RUN	STOP	STOP
16-bit timer 1 clock control	P16TON1(D3)/16-bit timer 1 clock control register(0x40148)	ON	OFF	OFF
16-bit timer 1 Run/Stop	PRUN1(D0)/16-bit timer 1 control register(0x4818E)	RUN	STOP	STOP
16-bit timer 2 clock control	P16TON2(D3)/16-bit timer 2 clock control register(0x40149)	ON	OFF	OFF
16-bit timer 2 Run/Stop	PRUN2(D0)/16-bit timer 2 control register(0x48196)	RUN	STOP	STOP
16-bit timer 3 clock control	P16TON3(D3)/16-bit timer 3 clock control register(0x4014A)	ON	OFF	OFF
16-bit timer 3 Run/Stop	PRUN3(D0)/16-bit timer 3 control register(0x4819E)	RUN	STOP	STOP
16-bit timer 4 clock control	P16TON4(D3)/16-bit timer 4 clock control register(0x4014B)	ON	OFF	OFF
16-bit timer 4 Run/Stop	PRUN4(D0)/16-bit timer 4 control register(0x481A6)	RUN	STOP	STOP
16-bit timer 5 clock control	P16TON5(D3)/16-bit timer 5 clock control register(0x4014C)	ON	OFF	OFF
16-bit timer 5 Run/Stop	PRUN5(D0)/16-bit timer 5 control register(0x481AE)	RUN	STOP	STOP
8-bit timer 0 clock control	P8TON0(D3)/8-bit timer 0/1 clock control register(0x4014D)	ON	OFF	OFF
8-bit timer 0 Run/Stop	PTRUN0(D0)/8-bit timer 0 control register(0x40160)	RUN	STOP	STOP
8-bit timer 1 clock control	P8TON1(D7)/8-bit timer 0/1 clock control register(0x4014D)	ON	OFF	OFF
8-bit timer 1 Run/Stop	PTRUN1(D0)/8-bit timer 1 control register(0x40164)	RUN	STOP	STOP
8-bit timer 2 clock control	P8TON2(D3)/8-bit timer 2/3 clock control register(0x4014E)	ON	OFF	OFF
8-bit timer 2 Run/Stop	PTRUN2(D0)/8-bit timer 2 control register(0x40168)	RUN	STOP	STOP
8-bit timer 3 clock control	P8TON3(D7)/8-bit timer 2/3 clock control register(0x4014E)	ON	OFF	OFF
8-bit timer 3 Run/Stop	PTRUN3(D0)/8-bit timer 3 control register(0x4016C)	RUN	STOP	STOP
A/D converter clock control	PSONAD(D3)/A/D clock control register(0x4014F)	ON	OFF	OFF
A/D conversion enable	ADE(D2)/A/D enable register(0x40244)	RUN	STOP	STOP

The same clock source must be used for the prescaler operating clock and the CPU operating clock. Therefore, when operating the CPU in low-speed with the OSC1 clock, the prescaler input clock must be switched according to the CPU operating clock. In this case, in order to prevent a malfunction in the peripheral circuit, the prescaler should be turned off before switching the CPU operating clock. After the CPU operating clock has been switched, switch the prescaler operating clock and then turn the prescaler on.

Function	Control bit	"1"	"0"	Default
Prescaler operating clock switch over	PSCDT0 (D0)/Prescaler clock select register(0x40181)	OSC1	OSC3/ PLL	OSC3/ PLL

6 Basic External Wiring Diagram



X'tal1	Crystal oscillator	32.768 kHz, C_I(Max.) = 34 kΩ
C_{G1}	Gate capacitor	10 pF
C_{D1}	Drain capacitor	10 pF
Rf_1	Feedback resistor	10 MΩ
X'tal2	Crystal oscillator	33 MHz (Max.)
CR	Ceramic oscillator	33 MHz (Max.)
C_{G2}	Gate capacitor	10 pF
C_{D2}	Drain capacitor	10 pF
Rf_2	Feedback resistor	1 MΩ
R_1	Resistor	4.7 kΩ
C_1	Capacitor	100 pF
C_2	Capacitor	5 pF

∗1: When the PLL is not used, leave the PLLC pin open.

Note: The above table is simply an example, and is not guaranteed to work.

7 Precautions on Mounting

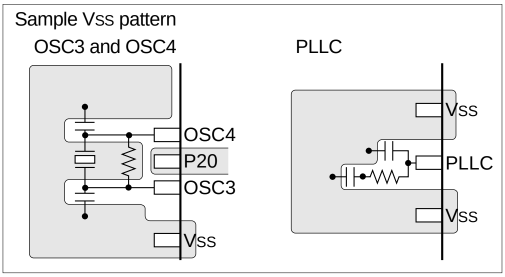
The following shows the precautions when designing the board and mounting the IC.

Oscillation Circuit

- Oscillation characteristics change depending on conditions (board pattern, components used, etc.).
  In particular, when a ceramic oscillator or crystal oscillator is used, use the oscillator manufacturer's recommended values for constants such as capacitance and resistance.
- Disturbances of the oscillation clock due to noise may cause a malfunction. Consider the following points to prevent this:

  (1) Components which are connected to the OSC3 (OSC1), OSC4 (OSC2) and PLLC pins, such as oscillators, resistors and capacitors, should be connected in the shortest line.

  (2) As shown in the figure below, make a VSS pattern as large as possible at circumscription of the OSC3 (OSC1) and OSC4 (OSC2) pins and the components connected to these pins. The same applies to the PLLC pin.
  Furthermore, do not use this VSS pattern to connect other components than the oscillation system.



In the QFP5-128pin/QFP15-128pin package, the P20 pin is located between the OSC3 and OSC4 pins. The P20 signal line should be laid under the package, not parallel with the OSC3 and OSC4 patterns. Furthermore, the #CE7 between the OSC1 and OSC2 pins should be wired in the same way.

  (3) When supplying an external clock to the OSC3 (OSC1) pin, the clock source should be connected to the OSC3 (OSC1) pin in the shortest line.
  Furthermore, do not connect anything else to the OSC4 (OSC2) pin.

- In order to prevent unstable operation of the oscillation circuit due to current leak between OSC3 (OSC1) and VDD, please keep enough distance between OSC3 (OSC1) and VDD or other signals on the board pattern.

Reset Circuit

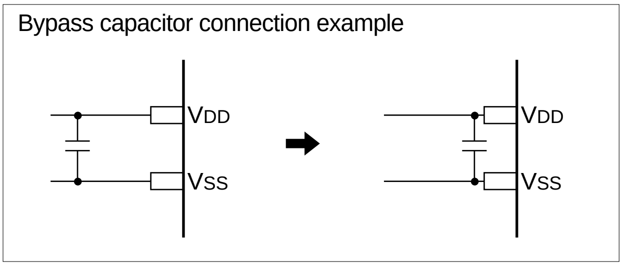
- The power-on reset signal which is input to the #RESET pin changes depending on conditions (power rise time, components used, board pattern, etc.). Decide the time constant of the capacitor and resistor after enough tests have been completed with the application product.
- In order to prevent any occurrences of unnecessary resetting caused by noise during operating, components such as capacitors and resistors should be connected to the #RESET pin in the shortest line.

Power Supply Circuit

- Sudden power supply variation due to noise may cause malfunction. Consider the following points to prevent this:

  (1) The power supply should be connected to the VDD, VDDE, VSS and AVDDE pins with patterns as short and large as possible.
  In particular, the power supply for AVDDE affects A/D conversion precision.

(2) When connecting between the VDD/VDDE and VSS pins with a bypass capacitor, the pins should be connected as short as possible.

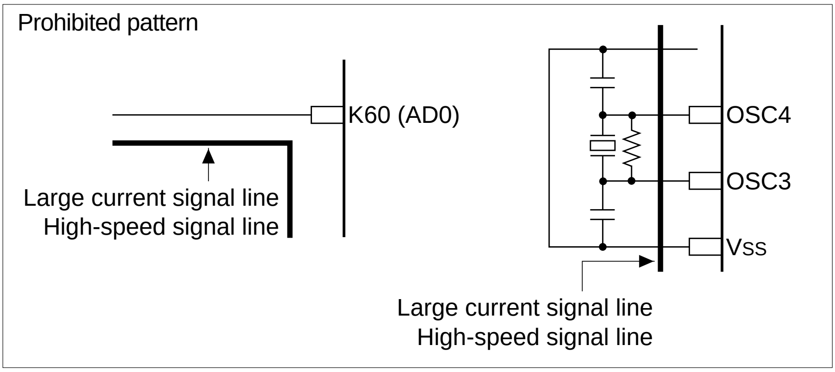


A/D Converter

- When the A/D converter is not used, the power supply pin AVDDE for the analog system should be connected to VDDE.

Arrangement of Signal Lines

- In order to prevent generation of electromagnetic induction noise caused by mutual inductance, do not arrange a large current signal line near the circuits that are sensitive to noise such as the oscillation unit and analog input unit.
- When a signal line is parallel with a high-speed line in long distance or intersects a high-speed line, noise may generated by mutual interference between the signals and it may cause a malfunction.
  Do not arrange a high-speed signal line especially near circuits that are sensitive to noise such as the oscillation unit and analog input unit.



Precautions for Visible Radiation (when bare chip is mounted)

- Visible radiation causes semiconductor devices to change the electrical characteristics. It may cause this IC to malfunction. When developing products which use this IC, consider the following precautions to prevent malfunctions caused by visible radiation.

(1) Design the product and implement the IC on the board so that it is shielded from visible radiation in actual use.

(2) The inspection process of the product needs an environment that shields the IC from visible radiation.

(3) As well as the face of the IC, shield the back and side too.

8 Electrical Characteristics

8.1 Absolute Maximum Rating

(V_{SS}=0V)

Item	Symbol	Condition	Rated value	Unit	*
Supply voltage	V_{DD}		-0.3 to +4.0	V	
I/O power voltage	V_{DDE}		-0.3 to +7.0	V	
Input voltage	V_I		-0.3 to V_{DDE}+0.5	V	
High-level output current	I_{OH}	1 pin	-10	mA	
		Total of all pins	-40	mA	
Low-level output current	I_{OL}	1 pin	10	mA	
		Total of all pins	40	mA	
Analog power voltage	AV_{DDE}		-0.3 to +7.0	V	
Analog input voltage	AV_{IN}		-0.3 to AV_{DDE}+0.3	V	
Storage temperature	T_{STG}		-65 to +150	°C	

8.2 Recommended Operating Conditions

1) 3.3 V/5.0 V dual power source

(V_{SS}=0V)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Supply voltage (high voltage)	V_{DDE}		4.50	5.00	5.50	V	
Supply voltage (low voltage)	V_{DD}		2.70	–	3.60	V	
Input voltage	HV_I		V_{SS}	–	V_{DDE}	V	
	LV_I		V_{SS}	–	V_{DD}	V	
CPU operating clock frequency	f_{CPU}	ROM-less model and 3.3±0.3V	–	–	60	MHz	
		ROM model or 3.0±0.3V	–	–	50	MHz	
External bus clock frequency	f_{BUS}		–	–	33	MHz	
Low-speed oscillation frequency	f_{OSC1}		–	32.768	–	kHz	
Operating temperature	Ta		-40	25	85	°C	
Input rise time (normal input)	tri		–	–	50	nS	
Input fall time (normal input)	tfi		–	–	50	nS	
Input rise time (schmitt input)	tri		–	–	5	mS	
Input fall time (schmitt input)	tfi		–	–	5	mS	

2) 3.3 V single power source

(V_{DDE}=V_{DD}, V_{SS}=0V)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Supply voltage	V_{DD}		2.70	–	3.60	V	
Input voltage	V_I		V_{SS}	–	V_{DD}	V	
CPU operating clock frequency	f_{CPU}	ROM-less model and 3.3±0.3V	–	–	60	MHz	
		ROM model or 3.0±0.3V	–	–	50	MHz	
External bus clock frequency	f_{BUS}		–	–	33	MHz	
Low-speed oscillation frequency	f_{OSC1}		–	32.768	–	kHz	
Operating temperature	Ta		-40	25	85	°C	
Input rise time (normal input)	tri		–	–	50	nS	
Input fall time (normal input)	tfi		–	–	50	nS	
Input rise time (schmitt input)	tri		–	–	5	mS	
Input fall time (schmitt input)	tfi		–	–	5	mS	

3) 2.0 V single power source

(V_{DDE}=V_{DD}, V_{SS}=0V)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Supply voltage	V_{DD}		1.80	2.00	2.20	V	
Input voltage	V_I		V_{SS}	–	V_{DD}	V	
CPU operating clock frequency	f_{CPU}		–	–	20	MHz	
External bus clock frequency	f_{BUS}		–	–	20	MHz	
Low-speed oscillation frequency	f_{OSC1}		–	32.768	–	kHz	
Operating temperature	Ta		-40	25	85	°C	
Input rise time (normal input)	tri		–	–	100	nS	
Input fall time (normal input)	tfi		–	–	100	nS	
Input rise time (schmitt input)	tri		–	–	10	mS	
Input fall time (schmitt input)	tfi		–	–	10	mS	

8.3 DC Characteristics

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Input leakage current	I_{LI}		-1	–	1	µA	
Off-state leakage current	I_{OZ}		-1	–	1	µA	
High-level output voltage	V_{OH}	I_{OH}=-3mA (Type1), V_{DDE}=Min.	V_{DDE} -0.4	–	–	V	
Low-level output voltage	V_{OL}	I_{OL}=3mA (Type1), V_{DDE}=Min.	–	–	0.4	V	
High-level input voltage	V_{IH}	CMOS level, V_{DDE}=Max.	3.5	–	–	V	1
Low-level input voltage	V_{IL}	CMOS level, V_{DDE}=Min.	–	–	1.0	V	1
Positive trigger input voltage	V_{T+}	CMOS schmitt	2.0	–	4.0	V	
Negative trigger input voltage	V_{T-}	CMOS schmitt	0.8	–	3.1	V	
Hysteresis voltage	V_H	CMOS schmitt	0.3	–	–	V	
High-level input voltage	V_{IH2}	TTL level, V_{DDE}=Max.	2.0	–	–	V	2
Low-level input voltage	V_{IL2}	TTL level, V_{DDE}=Min.	–	–	0.8	V	2
Pull-up resistor	R_{PU}	V_I=0V	60	120	288	kΩ	
Pull-down resistor	R_{PD}	V_I=V_{DDE} (#ICEMD)	30	60	144	kΩ	
Input pin capacitance	C_I	f=1MHz, V_{DDE}=0V	–	–	10	pF	
Output pin capacitance	C_O	f=1MHz, V_{DDE}=0V	–	–	10	pF	
I/O pin capacitance	C_{IO}	f=1MHz, V_{DDE}=0V	–	–	10	pF	

∗ note 1) In the E0C3320xFXE, the data bus (D[15:0]) is interfaced with CMOS level.
2) In the E0C3320xF0A, the data bus (D[15:0]) is interfaced with TTL level.

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Condition		Min.	Typ.	Max.	Unit	*
Static current consumption	I_{DDS}	Static state, Tj=85°C		–	–	90	µA	
Input leakage current	I_{LI}			-1	–	1	µA	
Off-state leakage current	I_{OZ}			-1	–	1	µA	
High-level output voltage	V_{OH}	I_{OH}=-2mA (Type1), I_{OH}=-6mA (Type2), V_{DD}=Min.		V_{DD} -0.4	–	–	V	
Low-level output voltage	V_{OL}	I_{OL}=2mA (Type1), I_{OL}=6mA (Type2), V_{DD}=Min.		–	–	0.4	V	
High-level input voltage	V_{IH}	CMOS level, V_{DD}=Max.		2.4	–	–	V	1
Low-level input voltage	V_{IL}	CMOS level, V_{DD}=Min.		–	–	0.4	V	1
Positive trigger input voltage	V_{T+}	LVTTL schmitt		1.1	–	2.4	V	
Negative trigger input voltage	V_{T-}	LVTTL schmitt		0.6	–	1.8	V	
Hysteresis voltage	V_H	LVTTL schmitt		0.1	–	–	V	
Pull-up resistor	R_{PU}	V_I=0V	Other than DSIO	80	200	480	kΩ	
			DSIO	40	100	240	kΩ	
Pull-down resistor	R_{PD}	V_I=V_{DD} (#ICEMD)		40	100	240	kΩ	
Input pin capacitance	C_I	f=1MHz, V_{DD}=0V		–	–	10	pF	
Output pin capacitance	C_O	f=1MHz, V_{DD}=0V		–	–	10	pF	
I/O pin capacitance	C_{IO}	f=1MHz, V_{DD}=0V		–	–	10	pF	

∗ note 1) In the E0C3320xFXE, the data bus (D[15:0]) is interfaced with CMOS level.

Note: See Appendix B for pin characteristics.

3) 2.0 V single power source

(Unless otherwise specified: $V_{DDE}=V_{DD}=2V\pm0.2V$, $V_{SS}=0V$, Ta=-40 to +85°C)

Item	Symbol	Condition		Min.	Typ.	Max.	Unit	*
Static current consumption	I_{DDS}	Static state, Tj=85°C		–	–	80	µA	
Input leakage current	I_{LI}			-1	–	1	µA	
Off-state leakage current	I_{OZ}			-1	–	1	µA	
High-level output voltage	V_{OH}	I_{OH}=-0.6mA (Type1), I_{OH}=-2mA (Type2), V_{DD}=Min.		V_{DD} -0.2	–	–	V	
Low-level output voltage	V_{OL}	I_{OL}=0.6mA (Type1), I_{OL}=2mA (Type2), V_{DD}=Min.		–	–	0.2	V	
High-level input voltage	V_{IH}	CMOS level, V_{DD}=Max.		1.6	–	–	V	1
Low-level input voltage	V_{IL}	CMOS level, V_{DD}=Min.		–	–	0.3	V	1
Positive trigger input voltage	V_{T+}	CMOS schmitt		0.4	–	1.6	V	
Negative trigger input voltage	V_{T-}	CMOS schmitt		0.3	–	1.4	V	
Hysteresis voltage	V_H	CMOS schmitt		0	–	–	V	
Pull-up resistor	R_{PU}	V_I=0V	Other than DSIO	120	480	1200	kΩ	
			DSIO	60	240	600	kΩ	
Pull-down resistor	R_{PD}	$V_I=V_{DD}$ (#ICEMD)		60	240	600	kΩ	
Input pin capacitance	C_I	f=1MHz, V_{DD}=0V		–	–	10	pF	
Output pin capacitance	C_O	f=1MHz, V_{DD}=0V		–	–	10	pF	
I/O pin capacitance	C_{IO}	f=1MHz, V_{DD}=0V		–	–	10	pF	

* note 1) In the E0C3320xF$_{xE}$, the data bus (D[15:0]) is interfaced with CMOS level.

Note: See Appendix B for pin characteristics.

8.4 Current Consumption

1) 3.3 V power source

(Unless otherwise specified: V_{DDE}=2.7V to 5.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Condition		Min.	Typ.	Max.	Unit	*
Operating current	I_{DD1}	When CPU is operating	20MHz	–	25	35	mA	1
			33MHz	–	40	60		
			50MHz	–	65	85		
	I_{DD2}	HALT mode	20MHz	–	12	16	mA	2
			33MHz	–	20	26		
			50MHz	–	30	40		
	I_{DD3}	HALT2 mode, 20 to 50MHz		–	1.8	2.5	mA	3
	I_{DD4}	SLEEP mode		–	1	30	µA	4
Clock timer operation current	I_{DDCT}	When clock timer only is operating OSC1 oscillation: 32kHz		–	7	–	µA	5

2) 2.0 V power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Operating current	I_{DD1}	When CPU is operating, 20MHz	–	13	19	mA	1
	I_{DD2}	HALT mode, 20MHz	–	6	9	mA	2
	I_{DD3}	HALT2 mode, 20MHz	–	0.4	1.0	mA	3
	I_{DD4}	SLEEP mode	–	1	30	µA	4
Clock timer operation current	I_{DDCT}	When clock timer only is operating OSC1 oscillation: 32kHz	–	1.5	–	µA	5

3) Analog power current

(Unless otherwise specified: V_{DD}=2.0V±0.2V, V_{DDE}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
A/D converter operating current	AI_{DD1}	V_{DD}=3.6V, V_{DDE}=AV_{DD}=5.0V±0.5V	–	800	1400	µA	6
		V_{DD}=V_{DDE}=AV_{DD}=2.7V to 3.6V	–	500	800		

Current consumption measurement condition: V_{IH}=V_{DD}, V_{IL}=0V, output pins are open, V_{DDE} current is not included

* note)

No.	OSC3	OSC1	CPU	Clock timer	Other peripheral circuits
1	On	Off	Normal operation *1	Stop	Stop
2	On	Off	HALT mode	Stop	Stop
3	On	Off	HALT2 mode	Stop	Stop
4	Off	Off	SLEEP mode	Stop	Stop
5	Off	On	HALT mode	Run	Stop
6	On	Off	HALT mode	Stop	A/D converter only operated, conversion clock frequency=2MHz

*1: The values of current consumption while the CPU is operating were measured when a test program that consists of 55% load instructions, 23% arithmetic operation instructions, 1% mac instruction, 12% branch instructions and 9% ext instruction is being executed in the built-in RAM continuously.

8.5 A/D Converter Characteristics

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=AV_{DD}=4.5V to 5.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C, ST[1:0]=11)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	∗
Resolution	–		–	10	–	bit	
Conversion time	–		5	–	–	µS	1
Zero scale error	E_{ZS}		0	2	4	LSB	
Full scale error	E_{FS}		-2	–	2	LSB	
Integral linearity error	E_L		-3	–	3	LSB	
Differential linearity error	E_D		-3	–	3	LSB	
Permissible signal source impedance	–		–	–	5	kΩ	
Analog input capacitance	–		–	–	45	pF	

∗ note 1) Indicates the minimum value when A/D clock = 4MHz (maximum clock frequency in 5V system).

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=AV_{DD}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C, ST[1:0]=11)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	∗
Resolution	–		–	10	–	bit	
Conversion time	–		10	–	–	µS	1
Zero scale error	E_{ZS}		0	2	4	LSB	
Full scale error	E_{FS}		-2	–	2	LSB	
Integral linearity error	E_L		-3	–	3	LSB	
Differential linearity error	E_D		-3	–	3	LSB	
Permissible signal source impedance	–		–	–	5	kΩ	
Analog input capacitance	–		–	–	45	pF	

∗ note 1) Indicates the minimum value when A/D clock = 2MHz (maximum clock frequency in 3V system).

Note:
- Be sure to use as V_{DDE} = AV_{DD}.
- The A/D converter cannot be used when the E0C33208/204/202 is used with a 2V power source.

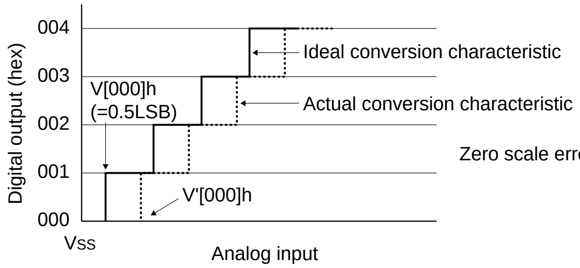
A/D conversion error

V[000]h = Ideal voltage at zero-scale point (=0.5LSB)
V'[000]h = Actual voltage at zero-scale point
V[3FF]h = Ideal voltage at full-scale point (=1022.5LSB)
V'[3FF]h = Actual voltage at full-scale point

$$1LSB = \frac{AV_{DD} - V_{SS}}{2^{10} - 1}$$

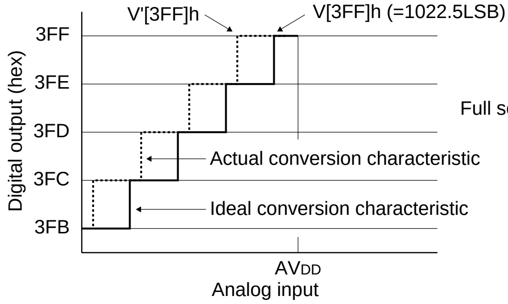
$$1LSB' = \frac{V'[3FF]h - V'[000]h}{2^{10} - 2}$$

■ **Zero scale error**



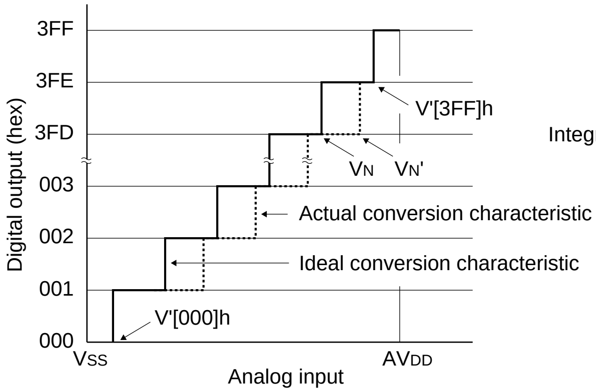
$$\text{Zero scale error } E_{ZS} = \frac{(V'[000]h - 0.5LSB') - (V[000]h - 0.5LSB)}{1LSB} \text{ [LSB]}$$

■ **Full scale error**



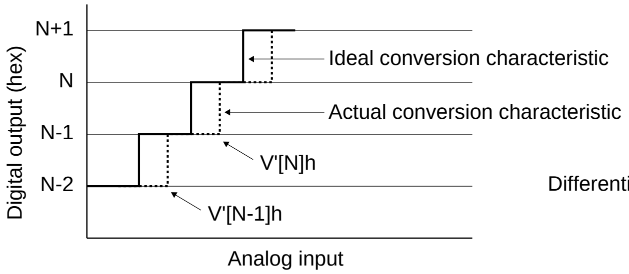
$$\text{Full scale error } E_{FS} = \frac{(V'[3FF]h + 0.5LSB') - (V[3FF]h + 0.5LSB)}{1LSB} \text{ [LSB]}$$

■ Integral linearity error



Integral linearity error $E_L = \frac{V_N' - V_N}{1LSB'}$ [LSB]

■ Differential linearity error



Differential linearity error $E_D = \frac{V'[N]h - V'[N-1]h}{1LSB'} - 1$ [LSB]

8.6 AC Characteristics

8.6.1 Symbol Description

t_{CYC}: Bus-clock cycle time

- In x1 mode, t_{CYC} = 50 nS (20 MHz) when the CPU is operated with a 20-MHz clock
  t_{CYC} = 30 nS (33 MHz) when the CPU is operated with a 33-MHz clock
- In x2 mode, t_{CYC} = 50 nS (20 MHz) when the CPU is operated with a 40-MHz clock
  t_{CYC} = 40 nS (25 MHz) when the CPU is operated with a 50-MHz clock
  t_{CYC} = 33 nS (30 MHz) when the CPU is operated with a 60-MHz clock

WC: Number of wait cycles

Up to 7 cycles can be set for the number of cycles using the BCU control register. Furthermore, it can be extended to a desired number of cycles by setting the #WAIT pin from outside of the IC.

The minimum number of read cycles with no wait (0) inserted is 1 cycle.

The minimum number of write cycles with no wait cycle (0) inserted is 2 cycles. It does not change even if 1-wait cycle is set. The write cycle is actually extended when 2 or more wait cycles are set.

When inserting wait cycles by controlling the #WAIT pin from outside of the IC, pay attention to the timing of the #WAIT signal sampling. Read cycles are terminated at the cycle in which the #WAIT signal is negated. Write cycles are terminated at the following cycle after the #WAIT signal is negated.

C1, C2, C3, Cn: Cycle number

C1 indicates the first cycle when the BCU transfers data from/to an external memory or another device. Similarly, C2 and Cn indicate the second cycle and nth cycle, respectively.

Cw: Wait cycle

Indicates that the cycle is wait cycle inserted.

8.6.2 AC Characteristics Measurement Condition

Signal detection level:	Input signal	High level	$V_{IH} = V_{DDE} - 0.4$ V
		Low level	V_{IL} = 0.4 V
	Output signal	High level	V_{OH} = 1/2 V_{DDE}
		Low level	V_{OL} = 1/2 V_{DDE}
	The following applies when OSC3 is external clock input:		
	Input signal	High level	V_{IH} = 1/2 V_{DD}
		Low level	V_{IL} = 1/2 V_{DD}
Input signal waveform:	Rise time (10% → 90% V_{DD})	5 nS	
	Fall time (90% → 10% V_{DD})	5 nS	
Output load capacitance:	C_L = 50 pF		

8.6.3 AC Characteristic Tables

External clock input characteristics

(Note) These AC characteristics apply to input signals from outside the IC.
The OSC3 input clock must be within VDD to VSS voltage range.

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: VDDE=5.0V±0.5V, VDD=2.7V to 3.6V, VSS=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
High-speed clock cycle time	t_{C3}	30		nS	
OSC3 clock input duty	t_{C3ED}	45	55	%	
OSC3 clock input rise time	t_{IF}		5	nS	
OSC3 clock input fall time	t_{IR}		5	nS	
BCLK high-level output delay time	t_{CD1}		35	nS	
BCLK low-level output delay time	t_{CD2}		35	nS	
Minimum reset pulse width	t_{RST}	$6 \cdot t_{CYC}$		nS	

2) 3.3 V single power source

(Unless otherwise specified: VDDE=VDD=2.7V to 3.6V, VSS=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
High-speed clock cycle time	t_{C3}	30		nS	
OSC3 clock input duty	t_{C3ED}	45	55	%	
OSC3 clock input rise time	t_{IF}		5	nS	
OSC3 clock input fall time	t_{IR}		5	nS	
BCLK high-level output delay time	t_{CD1}		35	nS	
BCLK low-level output delay time	t_{CD2}		35	nS	
Minimum reset pulse width	t_{RST}	$6 \cdot t_{CYC}$		nS	

3) 2.0 V single power source

(Unless otherwise specified: VDDE=VDD=2.0V±0.2V, VSS=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
High-speed clock cycle time	t_{C3}	50		nS	
OSC3 clock input duty	t_{C3ED}	45	55	%	
OSC3 clock input rise time	t_{IF}		5	nS	
OSC3 clock input fall time	t_{IR}		5	nS	
BCLK high-level output delay time	t_{CD1}		60	nS	
BCLK low-level output delay time	t_{CD2}		60	nS	
Minimum reset pulse width	t_{RST}	$6 \cdot t_{CYC}$		nS	

BCLK clock output characteristics

(Note) These AC characteristic values are applied only when the high-speed oscillation circuit is used.

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: VDDE=5.0V±0.5V, VDD=2.7V to 3.6V, VSS=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
BCLK clock output duty	t_{CBD}	40	60	%	

2) 3.3 V single power source

(Unless otherwise specified: VDDE=VDD=2.7V to 3.6V, VSS=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
BCLK clock output duty	t_{CBD}	40	60	%	

3) 2.0 V single power source

(Unless otherwise specified: VDDE=VDD=2.0V±0.2V, VSS=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
BCLK clock output duty	t_{CBD}	40	60	%	

Common characteristics

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Address delay time	t_{AD}	–	8	nS	
#CEx delay time (1)	t_{CE1}	–	8	nS	
#CEx delay time (2)	t_{CE2}	–	8	nS	
Wait setup time	t_{WTS}	15	–	nS	
Wait hold time	t_{WTH}	0	–	nS	
Read signal delay time (1)	t_{RDD1}		8	nS	
Read data setup time	t_{RDS}	12		nS	
Read data hold time	t_{RDH}	0		nS	
Write signal delay time (1)	t_{WRD1}		8	nS	
Write data delay time (1)	t_{WDD1}		10	nS	
Write data delay time (2)	t_{WDD2}	0	10	nS	
Write data hold time	t_{WDH}	0		nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Address delay time	t_{AD}	–	10	nS	
#CEx delay time (1)	t_{CE1}	–	10	nS	
#CEx delay time (2)	t_{CE2}	–	10	nS	
Wait setup time	t_{WTS}	15	–	nS	
Wait hold time	t_{WTH}	0	–	nS	
Read signal delay time (1)	t_{RDD1}		10	nS	
Read data setup time	t_{RDS}	15		nS	
Read data hold time	t_{RDH}	0		nS	
Write signal delay time (1)	t_{WRD1}		10	nS	
Write data delay time (1)	t_{WDD1}		10	nS	
Write data delay time (2)	t_{WDD2}	0	10	nS	
Write data hold time	t_{WDH}	0		nS	

3) 2.0 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Address delay time	t_{AD}	–	20	nS	
#CEx delay time (1)	t_{CE1}	–	20	nS	
#CEx delay time (2)	t_{CE2}	–	20	nS	
Wait setup time	t_{WTS}	40	–	nS	
Wait hold time	t_{WTH}	0	–	nS	
Read signal delay time (1)	t_{RDD1}		20	nS	
Read data setup time	t_{RDS}	40		nS	
Read data hold time	t_{RDH}	0		nS	
Write signal delay time (1)	t_{WRD1}		20	nS	
Write data delay time (1)	t_{WDD1}		20	nS	
Write data delay time (2)	t_{WDD2}	0	20	nS	
Write data hold time	t_{WDH}	0		nS	

SRAM read cycle

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Read signal delay time (2)	t_{RDD2}		8	nS	
Read signal pulse width	t_{RDW}	t_{CYC}(0.5+WC)-8		nS	
Read address access time (1)	t_{ACC1}		t_{CYC}(1+WC)-20	nS	
Chip enable access time (1)	t_{CEAC1}		t_{CYC}(1+WC)-20	nS	
Read signal access time (1)	t_{RDAC1}		t_{CYC}(0.5+WC)-20	nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Read signal delay time (2)	t_{RDD2}		10	nS	
Read signal pulse width	t_{RDW}	t_{CYC}(0.5+WC)-10		nS	
Read address access time (1)	t_{ACC1}		t_{CYC}(1+WC)-25	nS	
Chip enable access time (1)	t_{CEAC1}		t_{CYC}(1+WC)-25	nS	
Read signal access time (1)	t_{RDAC1}		t_{CYC}(0.5+WC)-25	nS	

3) 2.0 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Read signal delay time (2)	t_{RDD2}		10	nS	
Read signal pulse width	t_{RDW}	t_{CYC}(0.5+WC)-10		nS	
Read address access time (1)	t_{ACC1}		t_{CYC}(1+WC)-60	nS	
Chip enable access time (1)	t_{CEAC1}		t_{CYC}(1+WC)-60	nS	
Read signal access time (1)	t_{RDAC1}		t_{CYC}(0.5+WC)-60	nS	

SRAM write cycle

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Write signal delay time (2)	t_{WRD2}		8	nS	
Write signal pulse width	t_{WRW}	t_{CYC}(1+WC)-10		nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Write signal delay time (2)	t_{WRD2}		10	nS	
Write signal pulse width	t_{WRW}	t_{CYC}(1+WC)-10		nS	

3) 2.0 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Write signal delay time (2)	t_{WRD2}		20	nS	
Write signal pulse width	t_{WRW}	t_{CYC}(1+WC)-20		nS	

DRAM access cycle common characteristics

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
#RAS signal delay time (1)	t_{RASD1}		10	nS	
#RAS signal delay time (2)	t_{RASD2}		10	nS	
#RAS signal pulse width	t_{RASW}	t_{CYC}(2+WC)-10		nS	
#CAS signal delay time (1)	t_{CASD1}		10	nS	
#CAS signal delay time (2)	t_{CASD2}		10	nS	
#CAS signal pulse width	t_{CASW}	t_{CYC}(0.5+WC)-5		nS	
Read signal delay time (3)	t_{RDD3}		10	nS	
Read signal pulse width (2)	t_{RDW2}	t_{CYC}(2+WC)-10		nS	
Write signal delay time (3)	t_{WRD3}		10	nS	
Write signal pulse width (2)	t_{WRW2}	t_{CYC}(2+WC)-10		nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
#RAS signal delay time (1)	t_{RASD1}		10	nS	
#RAS signal delay time (2)	t_{RASD2}		10	nS	
#RAS signal pulse width	t_{RASW}	t_{CYC}(2+WC)-10		nS	
#CAS signal delay time (1)	t_{CASD1}		10	nS	
#CAS signal delay time (2)	t_{CASD2}		10	nS	
#CAS signal pulse width	t_{CASW}	t_{CYC}(0.5+WC)-10		nS	
Read signal delay time (3)	t_{RDD3}		10	nS	
Read signal pulse width (2)	t_{RDW2}	t_{CYC}(2+WC)-10		nS	
Write signal delay time (3)	t_{WRD3}		10	nS	
Write signal pulse width (2)	t_{WRW2}	t_{CYC}(2+WC)-10		nS	

3) 2.0 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
#RAS signal delay time (1)	t_{RASD1}		20	nS	
#RAS signal delay time (2)	t_{RASD2}		20	nS	
#RAS signal pulse width	t_{RASW}	t_{CYC}(2+WC)-20		nS	
#CAS signal delay time (1)	t_{CASD1}		20	nS	
#CAS signal delay time (2)	t_{CASD2}		20	nS	
#CAS signal pulse width	t_{CASW}	t_{CYC}(0.5+WC)-20		nS	
Read signal delay time (3)	t_{RDD3}		20	nS	
Read signal pulse width (2)	t_{RDW2}	t_{CYC}(2+WC)-20		nS	
Write signal delay time (3)	t_{WRD3}		20	nS	
Write signal pulse width (2)	t_{WRW2}	t_{CYC}(2+WC)-20		nS	

DRAM random access cycle and DRAM fast-page cycle

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Column address access time	t_{ACCF}		t_{CYC}(1+WC)-25	nS	
#RAS access time	t_{RACF}		t_{CYC}(1.5+WC)-25	nS	
#CAS access time	t_{CACF}		t_{CYC}(0.5+WC)-25	nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Column address access time	t_{ACCF}		t_{CYC}(1+WC)-25	nS	
#RAS access time	t_{RACF}		t_{CYC}(1.5+WC)-25	nS	
#CAS access time	t_{CACF}		t_{CYC}(0.5+WC)-25	nS	

3) 2.0 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Column address access time	t_{ACCF}		t_{CYC}(1+WC)-60	nS	
#RAS access time	t_{RACF}		t_{CYC}(1.5+WC)-60	nS	
#CAS access time	t_{CACF}		t_{CYC}(0.5+WC)-60	nS	

EDO DRAM random access cycle and EDO DRAM page cycle

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Column address access time	t_{ACCE}		t_{CYC}(1.5+WC)-25	nS	
#RAS access time	t_{RACE}		t_{CYC}(2+WC)-25	nS	
#CAS access time	t_{CACE}		t_{CYC}(1+WC)-15	nS	
Read data setup time	t_{RDS2}	20		nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Column address access time	t_{ACCE}		t_{CYC}(1.5+WC)-25	nS	
#RAS access time	t_{RACE}		t_{CYC}(2+WC)-25	nS	
#CAS access time	t_{CACE}		t_{CYC}(1+WC)-20	nS	
Read data setup time	t_{RDS2}	20		nS	

3) 2.0 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Column address access time	t_{ACCE}		t_{CYC}(1.5+WC)-60	nS	
#RAS access time	t_{RACE}		t_{CYC}(2+WC)-60	nS	
#CAS access time	t_{CACE}		t_{CYC}(1+WC)-60	nS	
Read data setup time	t_{RDS2}	20		nS	

Burst ROM read cycle

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Read address access time (2)	t_{ACC2}		t_{CYC}(1+WC)-20	nS	
Chip enable access time (2)	t_{CEAC2}		t_{CYC}(1+WC)-20	nS	
Read signal access time (2)	t_{RDAC2}		t_{CYC}(0.5+WC)-20	nS	
Burst address access time	t_{ACCB}		t_{CYC}(1+WC)-20	nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Read address access time (2)	t_{ACC2}		t_{CYC}(1+WC)-25	nS	
Chip enable access time (2)	t_{CEAC2}		t_{CYC}(1+WC)-25	nS	
Read signal access time (2)	t_{RDAC2}		t_{CYC}(0.5+WC)-25	nS	
Burst address access time	t_{ACCB}		t_{CYC}(1+WC)-25	nS	

3) 2.0 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
Read address access time (2)	t_{ACC2}		t_{CYC}(1+WC)-60	nS	
Chip enable access time (2)	t_{CEAC2}		t_{CYC}(1+WC)-60	nS	
Read signal access time (2)	t_{RDAC2}		t_{CYC}(0.5+WC)-60	nS	
Burst address access time	t_{ACCB}		t_{CYC}(1+WC)-60	nS	

External bus master and NMI

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
#BUSREQ signal setup time	t_{BRQS}	15		nS	
#BUSREQ signal hold time	t_{BRQH}	0		nS	
#BUSACK signal output delay time	t_{BAKD}		10	nS	
High-impedance → output delay time	t_{Z2E}		10	nS	
Output → high-impedance delay time	t_{B2Z}		10	nS	
#NMI pulse width	t_{NMIW}	30		nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
#BUSREQ signal setup time	t_{BRQS}	15		nS	
#BUSREQ signal hold time	t_{BRQH}	0		nS	
#BUSACK signal output delay time	t_{BAKD}		10	nS	
High-impedance → output delay time	t_{Z2E}		10	nS	
Output → high-impedance delay time	t_{B2Z}		10	nS	
#NMI pulse width	t_{NMIW}	30		nS	

3) 2.0 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item	Symbol	Min.	Max.	Unit	*
#BUSREQ signal setup time	t_{BRQS}	40		nS	
#BUSREQ signal hold time	t_{BRQH}	0		nS	
#BUSACK signal output delay time	t_{BAKD}		20	nS	
High-impedance → output delay time	t_{Z2E}		20	nS	
Output → high-impedance delay time	t_{B2Z}		20	nS	
#NMI pulse width	t_{NMIW}	90		nS	

Input, Output and I/O port

1) 3.3 V/5.0 V dual power source

(Unless otherwise specified: V_{DDE}=5.0V±0.5V, V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item		Symbol	Min.	Max.	Unit	*
Input data setup time		t_{INPS}	20		nS	
Input data hold time		t_{INPH}	10		nS	
Output data delay time		t_{OUTD}		20	nS	
K-port interrupt	SLEEP, HALT2 mode	t_{KINW}	30		nS	
input pulse width	Others		$2 \times t_{CYC}$		nS	

2) 3.3 V single power source

(Unless otherwise specified: V_{DDE}=V_{DD}=2.7V to 3.6V, V_{SS}=0V, Ta=-40 to +85°C)

Item		Symbol	Min.	Max.	Unit	*
Input data setup time		t_{INPS}	20		nS	
Input data hold time		t_{INPH}	10		nS	
Output data delay time		t_{OUTD}		20	nS	
K-port interrupt	SLEEP, HALT2 mode	t_{KINW}	30		nS	
input pulse width	Others		$2 \times t_{CYC}$		nS	

3) 2.0 V single power source

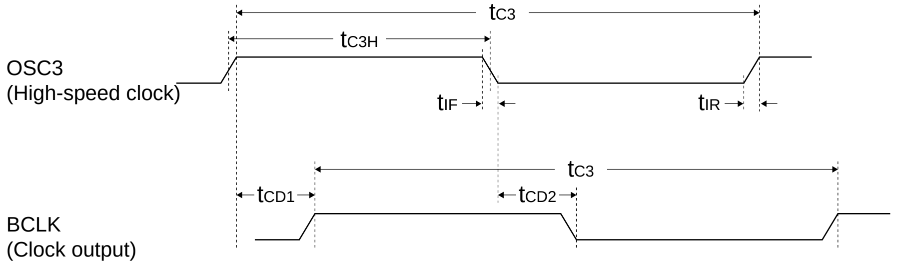
(Unless otherwise specified: V_{DDE}=V_{DD}=2.0V±0.2V, V_{SS}=0V, Ta=-40 to +85°C)

Item		Symbol	Min.	Max.	Unit	*
Input data setup time		t_{INPS}	40		nS	
Input data hold time		t_{INPH}	20		nS	
Output data delay time		t_{OUTD}		30	nS	
K-port interrupt	SLEEP, HALT2 mode	t_{KINW}	90		nS	
input pulse width	Others		$2 \times t_{CYC}$		nS	

8.6.4 AC Characteristic Timing Charts

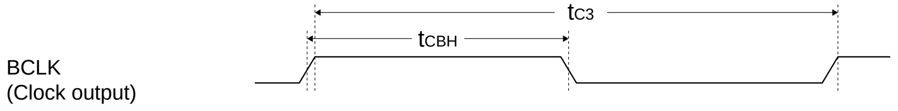
Clock

(1) When an external clock is input (in x1 speed mode):



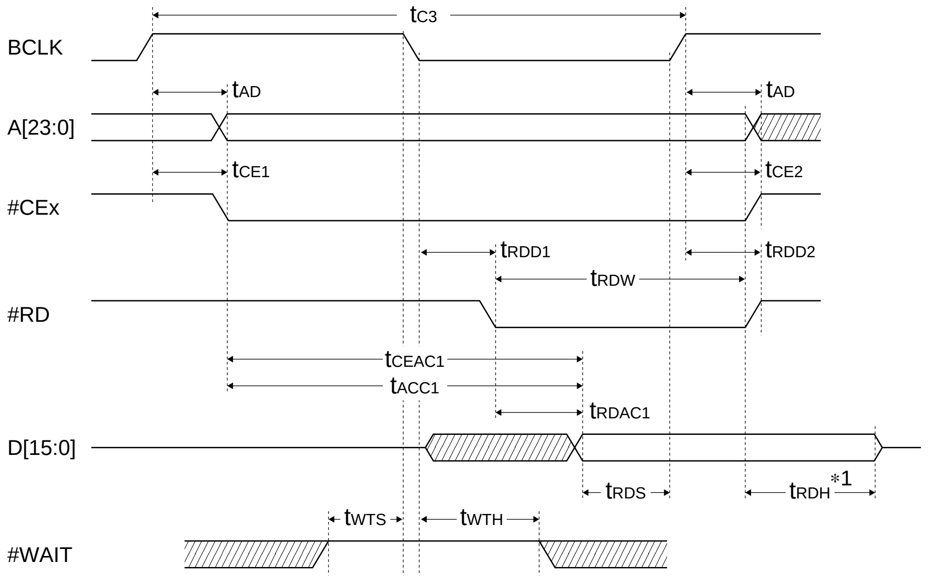
$t_{C3ED} = t_{C3H}/t_{C3}$

(2) When the high-speed oscillation circuit is used for the operating clock:



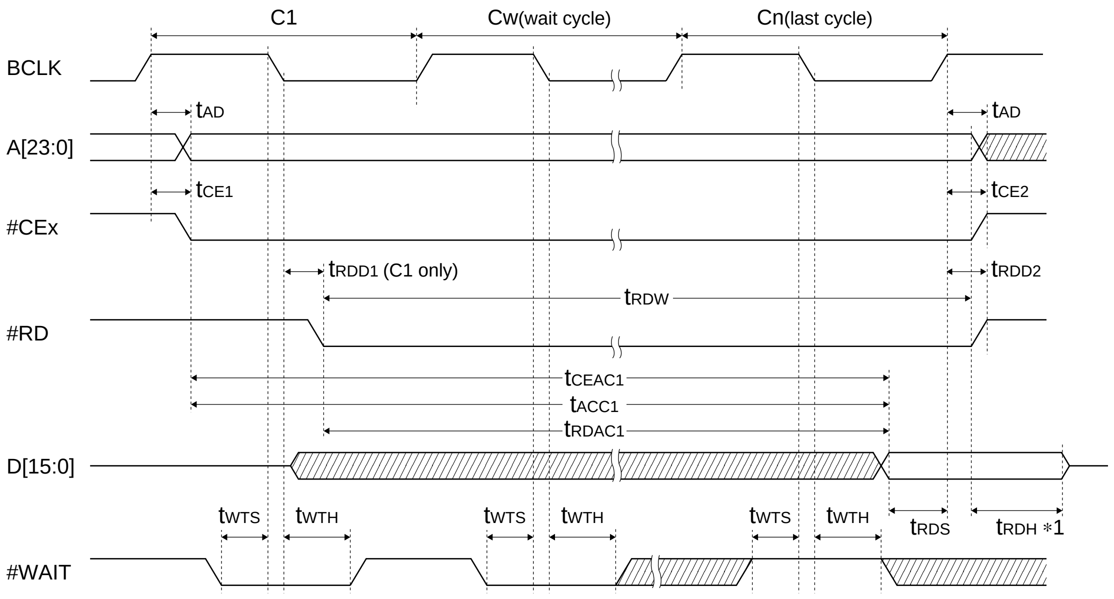
$t_{CBD} = t_{CBH}/t_{C3}$

SRAM read cycle (basic cycle: 1 cycle)



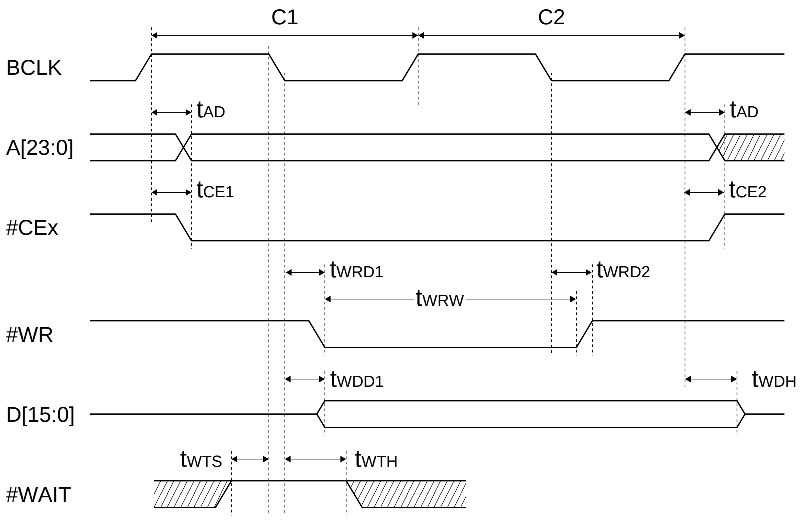
*1 tRDH is measured with respect to the first signal change (negation) from among the #RD, #CEx and A[23:0] signals.

SRAM read cycle (when a wait cycle is inserted)

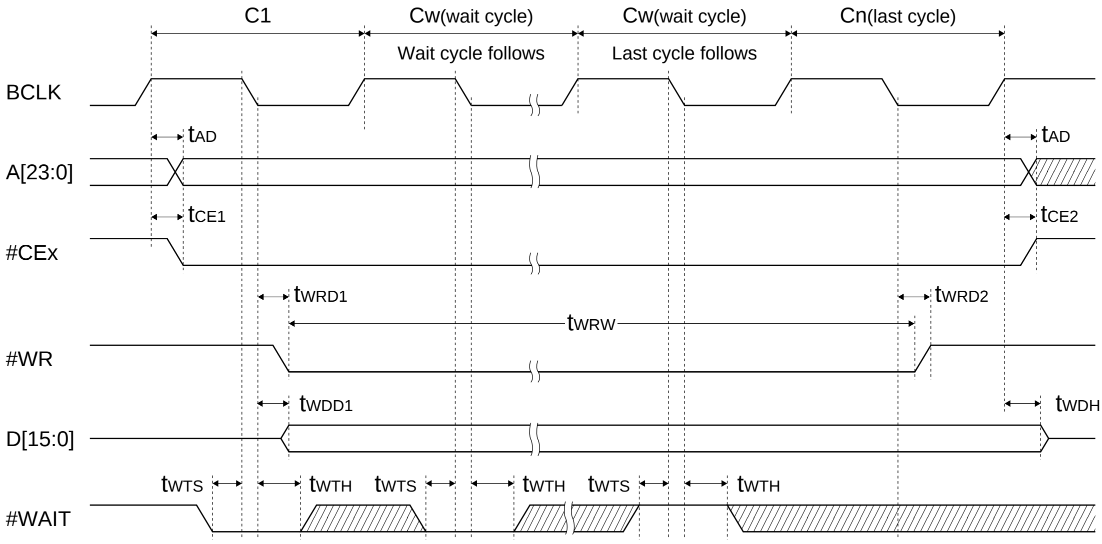


*1 tRDH is measured with respect to the first signal change (negation) from among the #RD, #CEx and A[23:0] signals.

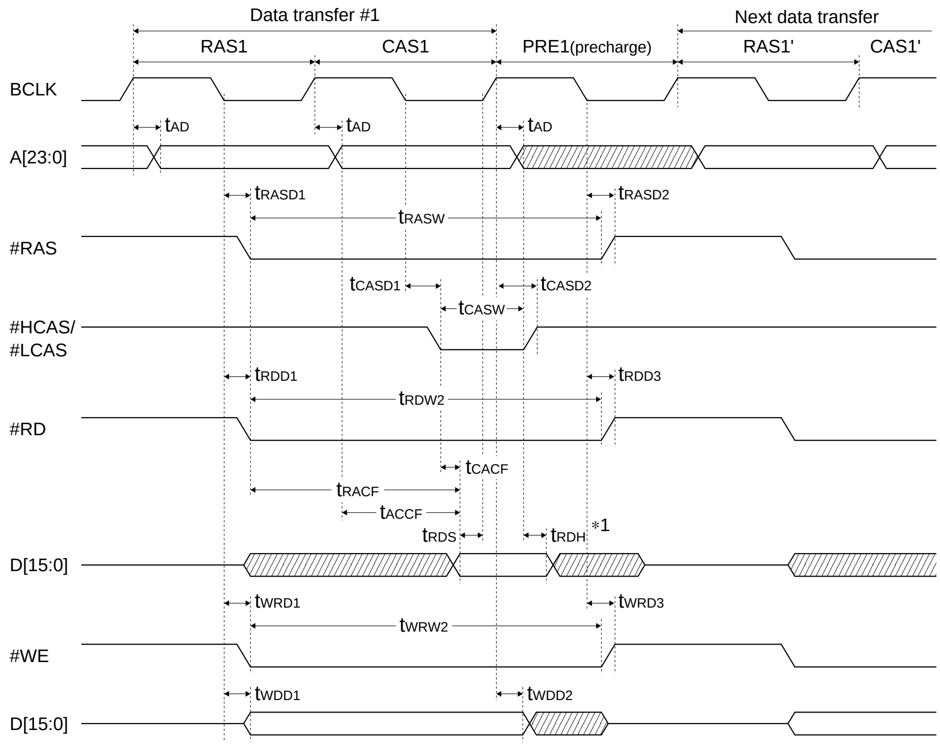
SRAM write cycle (basic cycle: 2 cycles)



SRAM write cycle (when wait cycles are inserted)

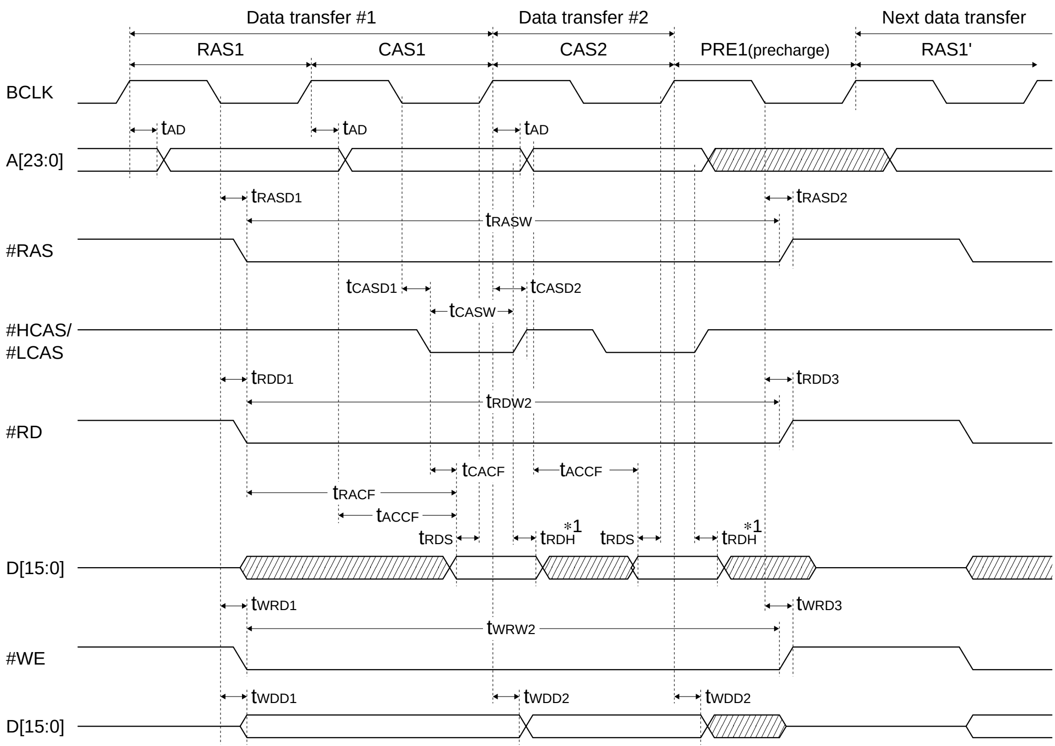


DRAM random access cycle (basic cycle)



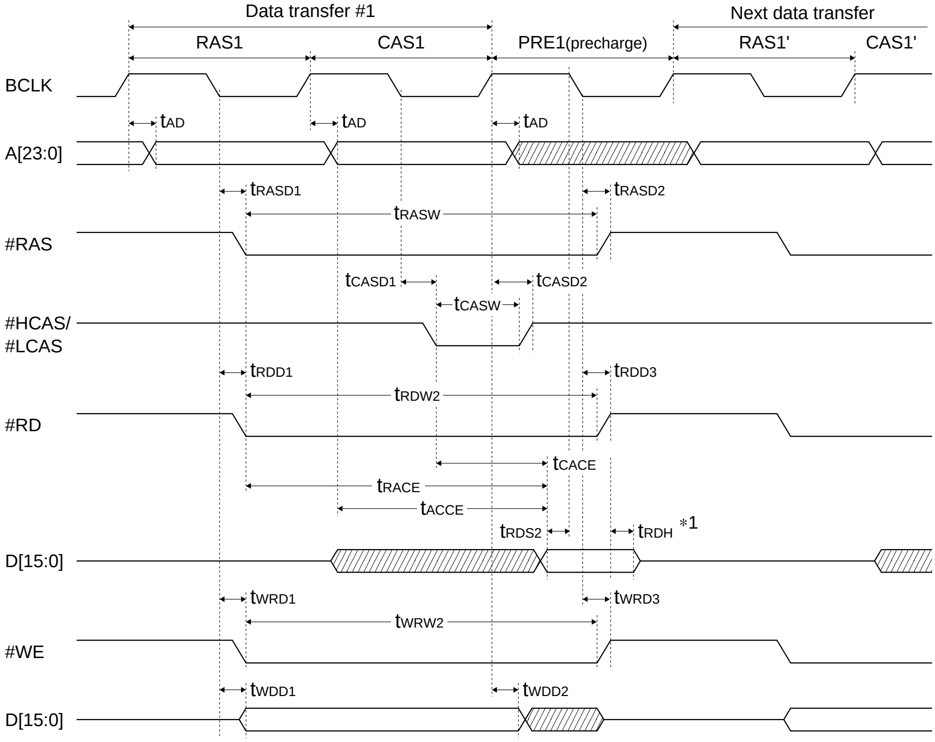
*1 tRDH is measured with respect to the first signal change (negation) of either the #RD or the A[23:0] signals.

DRAM fast-page access cycle



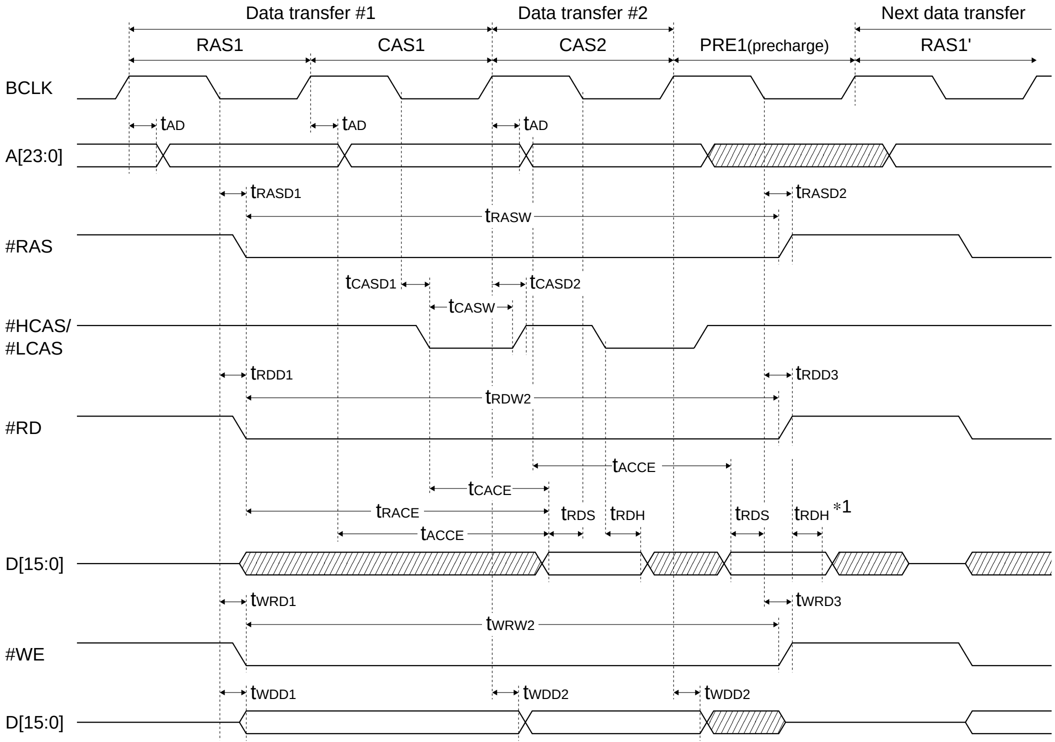
*1 tRDH is measured with respect to the first signal change (negation) of either the #RD or the A[23:0] signals.

EDO DRAM random access cycle (basic cycle)



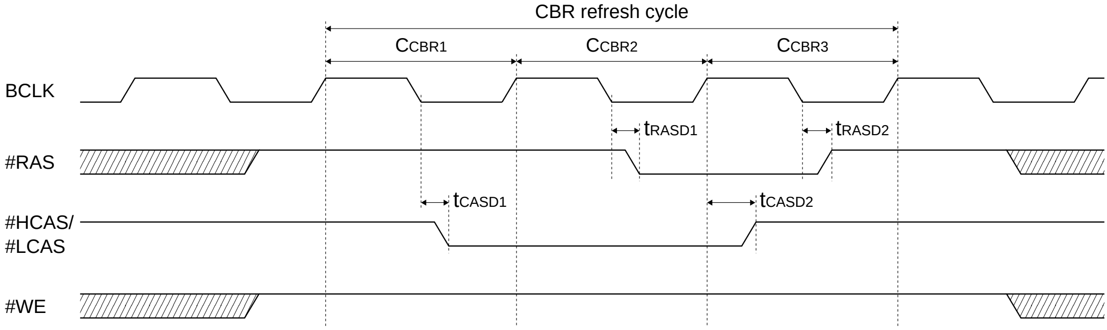
*1 tRDH is measured with respect to the first signal change (negation) of either the #RD or the #RASx signals.

EDO DRAM page access cycle

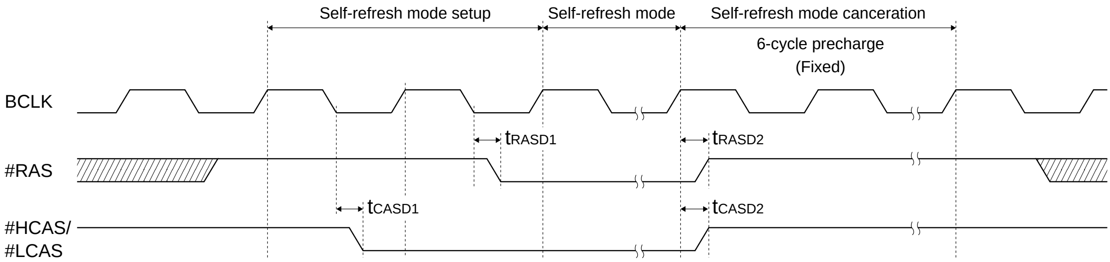


*1 tRDH is measured with respect to the first signal change from among the #RD (negation), #RASx (negation) and #CAS (rise) signals.

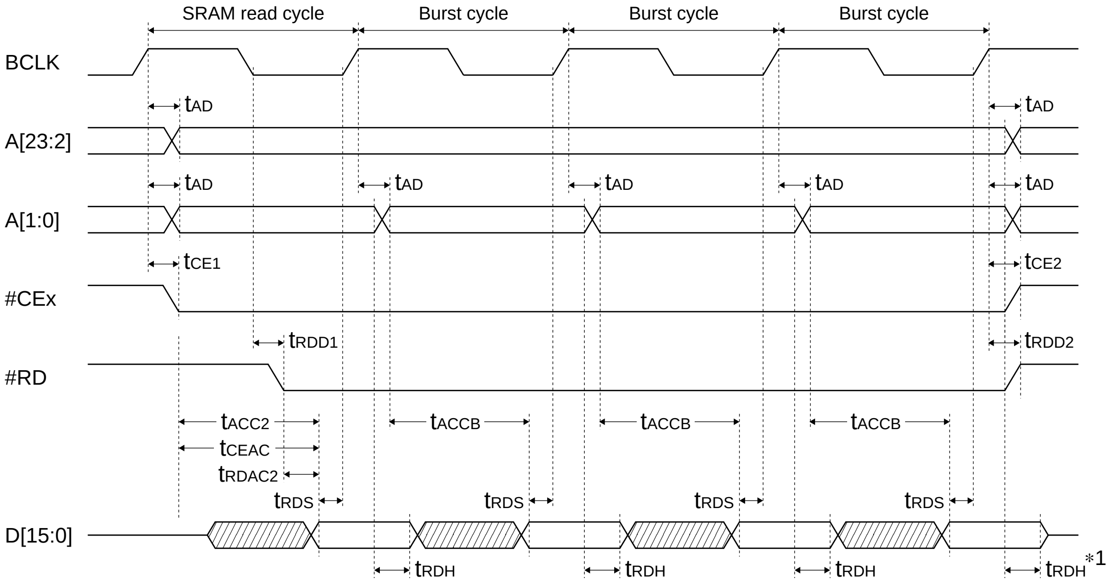
DRAM CAS-before-RAS refresh cycle



DRAM self-refresh cycle

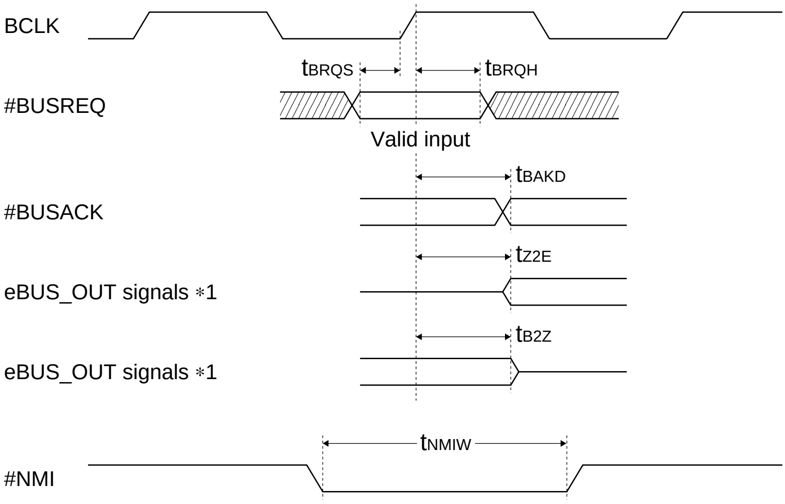


Burst ROM read cycle



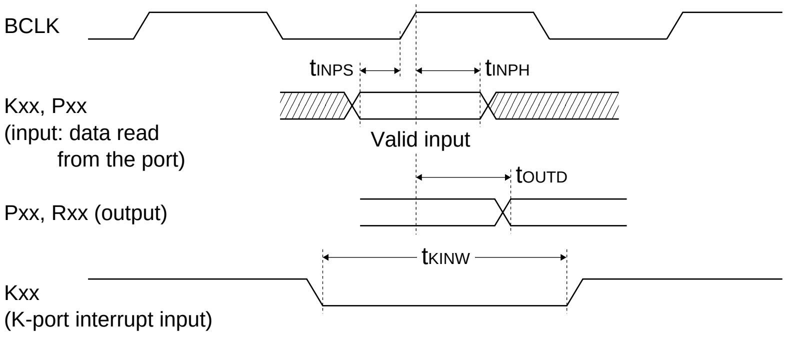
*1 t_{RDH} is measured with respect to the first signal change (negation) from among the #RD, #CEx and A[23:0] signals.

#BUSREQ, #BUSACK and #NMI timing



∗1 eBUS_OUT indicates the following pins:
A[23:0], #RD, #WRL, #WRH, #HCAS, #LCAS, #CE[17:4], D[15:0]

Input, output and I/O port timing



8.7 Oscillation Characteristics

Oscillation characteristics change depending on conditions (board pattern, components used, etc.). Use the following characteristics as reference values. In particular, when a ceramic or crystal oscillator is used, use the oscillator manufacturer recommended values for constants such as capacitance and resistance.

OSC1 crystal oscillation

(Unless otherwise specified: crystal=C-002RX#1 32.768kHz, Rf_1=20MΩ, C_{G1}=C_{D1}=15pF#2)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Operating temperature	Ta	V_{DD}=2.7V to 3.6V	-40		85	°C	
		V_{DD}=1.9V to 2.2V	-40		85	°C	
		V_{DD}=1.8V to 2.2V	0		70	°C	

#1 C-002RX: Crystal resonator made by Seiko Epson
#2 "C_{G1}=C_{D1}=15pF" includes board capacitance.

(Unless otherwise specified: V_{DD}=3.3V, V_{SS}=0V, crystal=C-002RX#1 32.768kHz, Rf_1=20MΩ, C_{G1}=C_{D1}=15pF#2, Ta=25°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Oscillation start time	t_{STA1}				3	Sec	
External gate/drain capacitance	C_{G1}, C_{D1}	C_{G1}=C_{D1}, including board capacitance	5		25	pF	
Frequency/IC deviation	Δf/ΔIC		-10		10	ppm	
Frequency/power voltage deviation	Δf/ΔV		-10		10	ppm/V	
Frequency adjustment range	$\Delta f/\Delta C_G$	C_G=5 to 25pF	50			ppm	

#1 C-002RX: Crystal resonator made by Seiko Epson
#2 "C_{G1}=C_{D1}=15pF" includes board capacitance.

(Unless otherwise specified: V_{DD}=2.0V, V_{SS}=0V, crystal=C-002RX#1 32.768kHz, Rf_1=20MΩ, C_{G1}=C_{D1}=15pF#2, Ta=25°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Oscillation start time	t_{STA1}				20	Sec	
External gate/drain capacitance	C_{G1}, C_{D1}	C_{G1}=C_{D1}, including board capacitance	5		25	pF	
Frequency/IC deviation	Δf/ΔIC		-10		10	ppm	
Frequency/power voltage deviation	Δf/ΔV		-10		10	ppm/V	
Frequency adjustment range	$\Delta f/\Delta C_G$	C_G=5 to 25pF	50			ppm	

#1 C-002RX: Crystal resonator made by Seiko Epson
#2 "C_{G1}=C_{D1}=15pF" includes board capacitance.

OSC3 crystal oscillation

Note: A "crystal resonator that uses a fundamental" should be used for the OSC3 crystal oscillation circuit.

(Unless otherwise specified: V_{DD}=3.3V, V_{SS}=0V, crystal=MA-306#1 33.8688MHz, Rf_2=1MΩ, C_{G1}=C_{D1}=15pF#2, Ta=25°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Oscillation start time	t_{STA3}	V_{DD}=3.3V			10	mS	
		V_{DD}=2.0V			25	mS	

#1 MA-306: Crystal resonator made by Seiko Epson
#2 "C_{G1}=C_{D1}=15pF" includes board capacitance.

OSC3 ceramic oscillation

(Unless otherwise specified: V_{SS}=0V, Ta=25°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Oscillation start time	t_{STA3}	10MHz ceramic oscillator			10	mS	1
		16MHz ceramic oscillator			10	mS	2
		20MHz ceramic oscillator			10	mS	3
		25MHz ceramic oscillator			5	mS	4
		33MHz ceramic oscillator			5	mS	5

* note)

No.	Ceramic oscillator	Recommended constants			Power voltage range (V)	Remarks (Manufacturer)
		C_{G2} (pF)	C_{D2} (pF)	R_{f2} (MΩ)		
1	CST10.0MTW	30	30	1	1.8 to 2.2	(Murata Mfg. corporation) *1
2	CST16.00MXW0C1	5	5	1	1.8 to 2.2	(Murata Mfg. corporation)
3	CST20.00MXW0H1	5	5	1	1.8 to 2.2	(Murata Mfg. corporation)
4	CST25.00MXW0H1	5	5	1	2.7 to 3.6	(Murata Mfg. corporation)
5	CST33.00MXZ040	Open	Open	1	2.7 to 3.6	(Murata Mfg. corporation)

*1 This oscillator has a tendency to rise to the frequency of 0.3%.

8.8 PLL Characteristics

Setting the PLLS0 and PLLS1 pins (recommended operating condition)

V_{DD}=2.7V to 3.6V

PLLS1	PLLS0	Mode	Fin (OSC3 clock)	Fout
1	1	x2	10 to 25MHz	20 to 50MHz
0	1	x4	10 to 12.5MHz	40 to 50MHz
0	0	PLL not used	–	–

V_{DD}=2.0V±0.2V

PLLS1	PLLS0	Mode	Fin (OSC3 clock)	Fout
1	1	x2	10MHz	20MHz
0	0	PLL not used	–	–

PLL characteristics

(Unless otherwise specified: V_{DD}=2.7V to 3.6V, V_{SS}=0V, crystal oscillator=SG-8002[#1],
R_1=4.7kΩ, C_1=100pF, C_2=5pF, Ta=-40 to +85°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Jitter (peak jitter)	t_{pj}		-1		1	nS	
Lockup time	t_{pll}				1	mS	

#1 SG-8002: Crystal oscillator made by Seiko Epson

(Unless otherwise specified: V_{DD}=2.0V±0.2V, V_{SS}=0V, crystal oscillator=SG-8002[#1],
R_1=4.7kΩ, C_1=100pF, C_2=5pF, Ta=-40 to +85°C)

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	*
Jitter (peak jitter)	t_{pj}		-2		2	nS	
Lockup time	t_{pll}				2	mS	

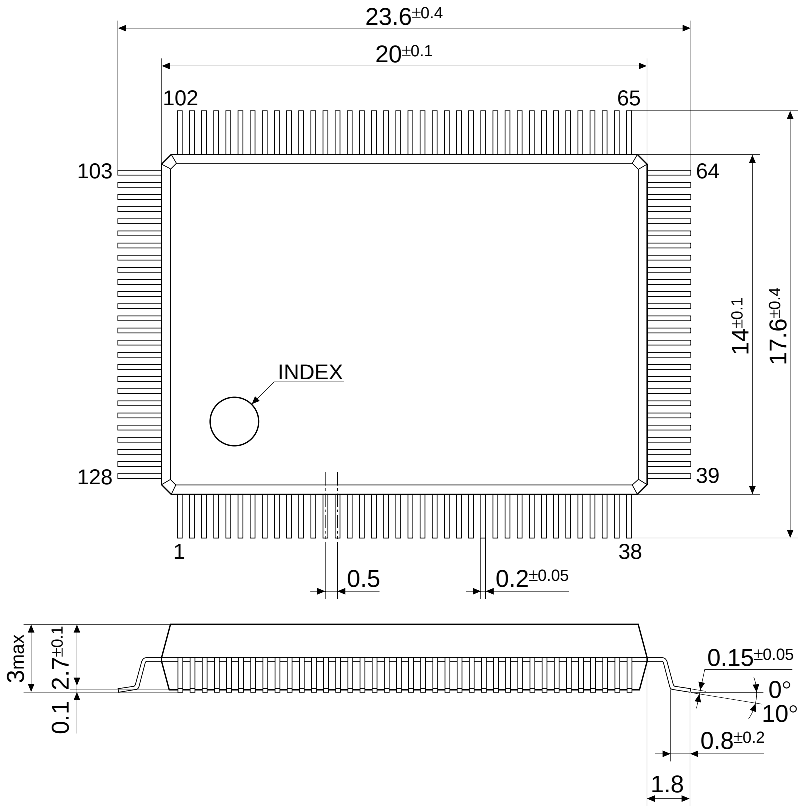
#1 SG-8002: Crystal oscillator made by Seiko Epson

9 Package

9.1 Plastic Package

QFP5-128pin

(Unit: mm)



Limit of power consumption

The chip temperature of an LSI rises according to power consumption. The chip temperature can be calculated from environment temperature (Ta), thermal resistance (θ) and power consumption (P_D).

Chip temperature (Tj) = Ta + ($P_D \times \theta$) (°C)

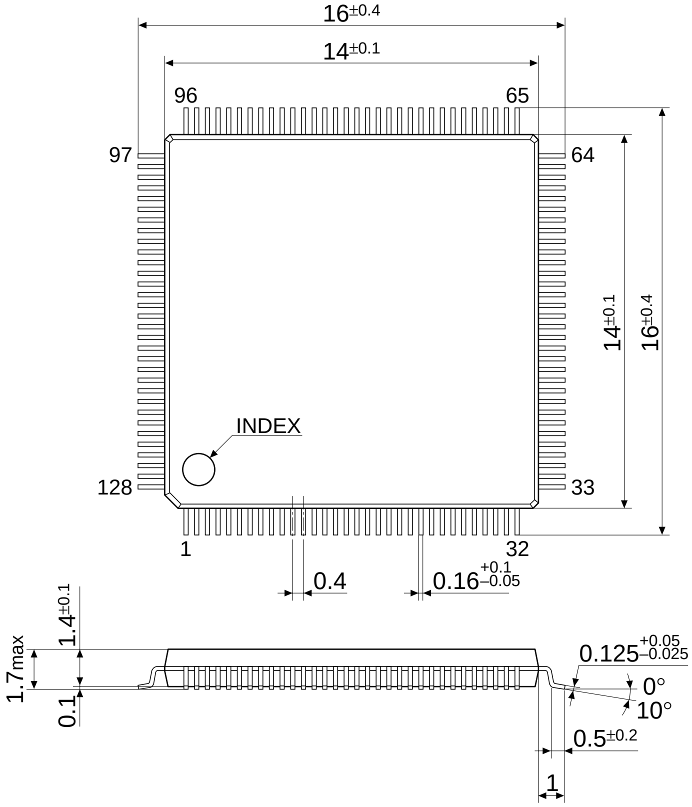
As a guide, normally keep the chip temperature (Tj) lower than 85°C.
The thermal resistance of the QFP5-128pin package is as follows:

Thermal resistance (°C/W) = 105 to 115 (75 to 85 for Cu lead frame)

This thermal resistance is a value under the condition that the measured device is hanging in the air and has no air-cooling. Thermal resistance greatly varies according to the mounting condition on the board and air-cooling condition.

QFP15-128pin

(Unit: mm)



Limit of power consumption

The chip temperature of an LSI rises according to power consumption. The chip temperature can be calculated from environment temperature (Ta), thermal resistance (θ) and power consumption (PD).

Chip temperature (Tj) = Ta + (PD × θ) (°C)

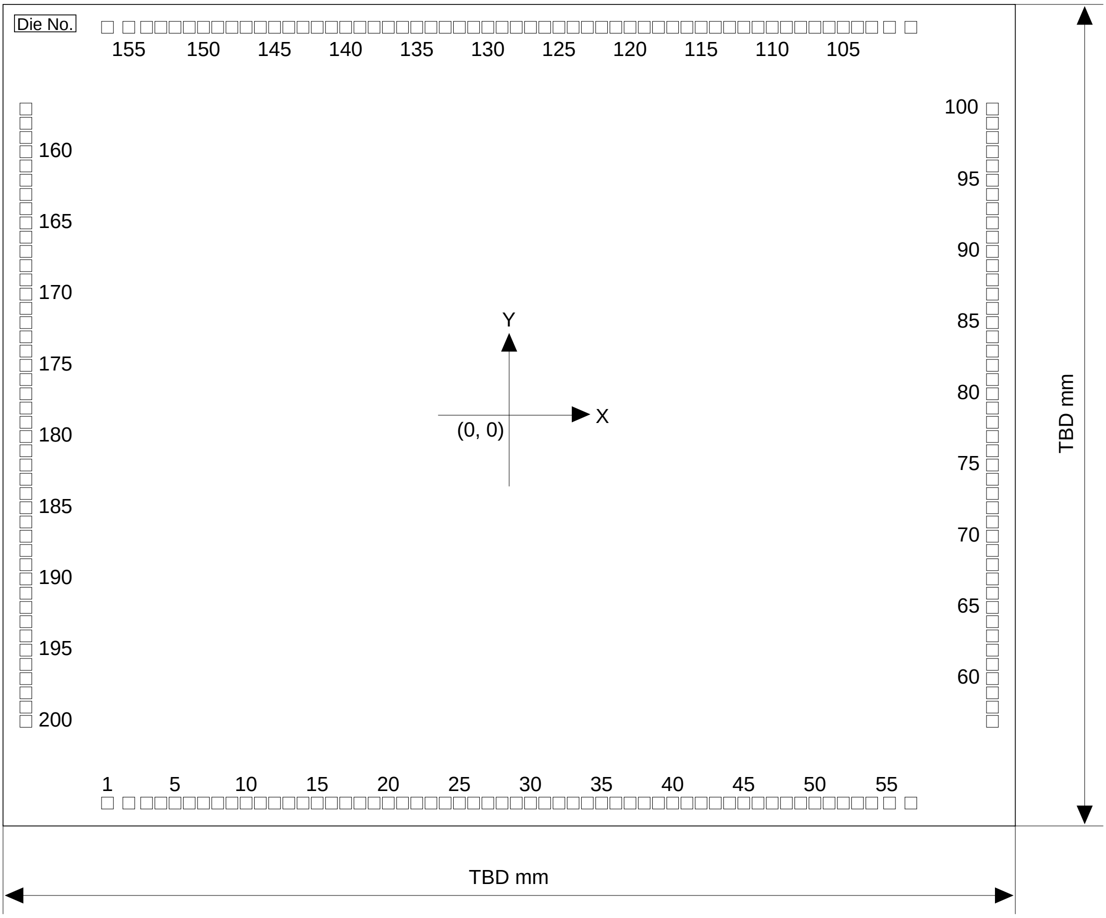
As a guide, normally keep the chip temperature (Tj) lower than 85°C.
The thermal resistance of the QFP15-128pin package is as follows:

Thermal resistance (°C/W) = 110 to 120 (90 to 100 for Cu lead frame)

This thermal resistance is a value under the condition that the measured device is hanging in the air and has no air-cooling. Thermal resistance greatly varies according to the mounting condition on the board and air-cooling condition.

10 Pad Layout

10.1 Pad Layout Diagram



10.2 Pad Coordinate

(Unit: µm)

No.	Pad name	X	Y	No.	Pad name	X	Y
1	P24/TM2	-2,373	-2,288	51	N.C.	1,890	-2,288
2	N.C.	-2,247	-2,288	52	K63/AD3	1,974	-2,288
3	VSS	-2,142	-2,288	53	N.C.	2,058	-2,288
4	N.C.	-2,058	-2,288	54	K62/AD2	2,142	-2,288
5	P25/TM3	-1,974	-2,288	55	N.C.	2,247	-2,288
6	N.C.	-1,890	-2,288	56	AVDDE	2,373	-2,288
7	P26/TM4	-1,806	-2,288	57	K61/AD1	2,855	-1,806
8	P15/EXCL4/#DMAEND0	-1,722	-2,288	58	N.C.	2,855	-1,722
9	N.C.	-1,638	-2,288	59	K60/AD0	2,855	-1,638
10	P27/TM5	-1,554	-2,288	60	N.C.	2,855	-1,554
11	N.C.	-1,470	-2,288	61	D6	2,855	-1,470
12	BCLK	-1,386	-2,288	62	N.C. (VDD)	2,855	-1,386
13	N.C.	-1,302	-2,288	63	VSS	2,855	-1,302
14	P00/SIN0	-1,218	-2,288	64	N.C.	2,855	-1,218
15	N.C.	-1,134	-2,288	65	D5	2,855	-1,134
16	P01/SOUT0	-1,050	-2,288	66	N.C.	2,855	-1,050
17	N.C.	-966	-2,288	67	D4	2,855	-966
18	D15	-882	-2,288	68	N.C.	2,855	-882
19	N.C.	-798	-2,288	69	D3	2,855	-798
20	VDD	-714	-2,288	70	N.C.	2,855	-714
21	P03/#SRDY0	-630	-2,288	71	D2	2,855	-630
22	D14	-546	-2,288	72	N.C.	2,855	-546
23	P31/#BUSGET/#GARD	-462	-2,288	73	D1	2,855	-462
24	D13	-378	-2,288	74	N.C. (VSS)	2,855	-378
25	P32/#DMAACK0	-294	-2,288	75	D0	2,855	-294
26	D12	-210	-2,288	76	P35/#BUSACK	2,855	-210
27	P33/#DMAACK1	-126	-2,288	77	VDDE	2,855	-126
28	D11	-42	-2,288	78	#CE9/#CE17	2,855	-42
29	K54/#DMAREQ3	42	-2,288	79	OSC2	2,855	42
30	D10	126	-2,288	80	#CE7/#RAS0/#CE13/#RAS2	2,855	126
31	K53/#DMAREQ2	210	-2,288	81	OSC1	2,855	210
32	D9	294	-2,288	82	#CE6	2,855	294
33	K52/#ADTRG	378	-2,288	83	N.C.	2,855	378
34	VSS	462	-2,288	84	#RD	2,855	462
35	K51/#DMAREQ1	546	-2,288	85	N.C.	2,855	546
36	P02/#SCLK0	630	-2,288	86	VSS	2,855	630
37	D8	714	-2,288	87	N.C.	2,855	714
38	N.C.	798	-2,288	88	#WRL/#WR/#WE/#LWE	2,855	798
39	D7	882	-2,288	89	N.C.	2,855	882
40	N.C.	966	-2,288	90	#WRH/#BSH/#UWE	2,855	966
41	VDDE	1,050	-2,288	91	N.C.	2,855	1,050
42	N.C.	1,134	-2,288	92	#CE10EX	2,855	1,134
43	K67/AD7	1,218	-2,288	93	N.C.	2,855	1,218
44	N.C.	1,302	-2,288	94	#CE8/#RAS1/#CE14/#RAS3	2,855	1,302
45	K66/AD6	1,386	-2,288	95	N.C.	2,855	1,386
46	N.C.	1,470	-2,288	96	#CE5/#CE15	2,855	1,470
47	K65/AD5	1,554	-2,288	97	N.C. (VDD)	2,855	1,554
48	N.C.	1,638	-2,288	98	#CE4/#CE11	2,855	1,638
49	K50/#DMAREQ0	1,722	-2,288	99	N.C.	2,855	1,722
50	K64/AD4	1,806	-2,288	100	P30/#WAIT/#CE4&5	2,855	1,806

No.	Pad name	X	Y	No.	Pad name	X	Y
101	#RESET	2,373	2,288	151	N.C.	-1,890	2,288
102	N.C.	2,247	2,288	152	A18	-1,974	2,288
103	#NMI	2,142	2,288	153	N.C.	-2,058	2,288
104	N.C.	2,058	2,288	154	A19	-2,142	2,288
105	A0/#BSL	1,974	2,288	155	N.C.	-2,247	2,288
106	N.C.	1,890	2,288	156	P04/SIN1/#DMAACK2	-2,373	2,288
107	A1	1,806	2,288	157	P05/SOUT1/#DMAEND2	-2,855	1,806
108	P34/#BUSREQ/#CE6	1,722	2,288	158	N.C.	-2,855	1,722
109	N.C.	1,638	2,288	159	P06/#SCLK1/DMAACK3	-2,855	1,638
110	V_{SS}	1,554	2,288	160	N.C.	-2,855	1,154
111	N.C.	1,470	2,288	161	V_{SS}	-2,855	1,470
112	A2	1,386	2,288	162	N.C.	-2,855	1,386
113	N.C.	1,302	2,288	163	PLLC	-2,855	1,302
114	A3	1,218	2,288	164	N.C.	-2,855	1,218
115	N.C.	1,134	2,288	165	V_{SS}	-2,855	1,134
116	A4	1,050	2,288	166	N.C.	-2,855	1,050
117	N.C.	966	2,288	167	PLLS1	-2,855	966
118	A5	882	2,288	168	N.C.	-2,855	882
119	N.C.	798	2,288	169	PLLS0	-2,855	798
120	A6	714	2,288	170	V_{DD}	-2,855	714
121	#CE10IN	630	2,288	171	P07/#SRDY1/#DMAEND3	-2,855	630
122	V_{DD}	546	2,288	172	N.C.	-2,855	546
123	#EMEMRD	462	2,288	173	#X2SPD	-2,855	462
124	A7	378	2,288	174	N.C.	-2,855	378
125	#HCAS/#UWE	294	2,288	175	EA10MD0	-2,855	294
126	A8	210	2,288	176	EA10MD1	-2,855	210
127	#LCAS/#CAS	126	2,288	177	V_{DD}	-2,855	126
128	A9	42	2,288	178	EA3MD	-2,855	42
129	P16/EXCL5/#DMAEND1	-42	2,288	179	OSC4	-2,855	-42
130	A10	-126	2,288	180	P20/#DRD	-2,855	-126
131	A20	-210	2,288	181	OSC3	-2,855	-210
132	A11	-294	2,288	182	P21/#DWE/#GAAS	-2,855	-294
133	A21	-378	2,288	183	N.C.	-2,855	-378
134	A12	-462	2,288	184	#CE3	-2,855	-462
135	A22	-546	2,288	185	N.C. (BSTB)	-2,855	-546
136	A13	-630	2,288	186	P22/TM0	-2,855	-630
137	A23	-714	2,288	187	N.C.	-2,855	-714
138	N.C.	-798	2,288	188	P23/TM1	-2,855	-798
139	V_{SS}	-882	2,288	189	N.C.	-2,855	-882
140	N.C.	-966	2,288	190	DSIO	-2,855	-966
141	A14	-1,050	2,288	191	N.C.	-2,855	-1,050
142	N.C.	-1,134	2,288	192	P10/EXCL0/T8UF0/DST0	-2,855	-1,134
143	A15	-1,218	2,288	193	N.C.	-2,855	-1,218
144	N.C.	-1,302	2,288	194	P11/EXCL1/T8UF1/DST1	-2,855	-1,302
145	V_{DDE}	-1,386	2,288	195	N.C.	-2,855	-1,386
146	N.C.	-1,470	2,288	196	P12/EXCL2/T8UF2/DST2	-2,855	-1,470
147	A16	-1,554	2,288	197	N.C.	-2,855	-1,554
148	N.C.	-1,638	2,288	198	P13/EXCL3/T8UF3/DST3	-2,855	-1,638
149	ICEMD	-1,722	2,288	199	N.C.	-2,855	-1,722
150	A17	-1,806	2,288	200	P14/FOSC1/DCLK	-2,855	-1,806

Note: The E0C33208 is constructed with 0.35 µm process technology. Since the pad pitch is to small, it is impossible to use all pads when mounting the chip (die form) on the board. When mounting the chip, use the pads other than "N.C." and "N.C. (xxxx)". The pads which is indicated with "N.C. (xxxx)" in the table is available in the QFP5-128pin/QFP15-128pin package.
The E0C33204/202 cannot be shipped in chip form.

Appendix A <Reference> External Device Interface Timings

This section shows setup examples for setting timing conditions of the external system interface as a reference material used when configuring a system with external devices.
Pay attention to the following precautions when using this material.

- The described AC characteristic values of external devices are standard values. They may differ from those of the devices actually used, so the actual setup values (number of cycles) should be determined by referring the manual or specification of the device to be used.
- It is necessary to set the timing values allowing ample margin according to the load capacitance of the bus and signal lines, number of devices to be connected, operating temperature range, I/O levels and other conditions. The number of cycles described in this section is an example and the conditions are not considered.
- The values described in "Time" column of the tables are simply calculated by multiplying the number of cycles by the cycle time. Conditions such as the output delay time of the device, delay due to wiring and load capacitance, and input setup time are not considered.
- The described contents are reference data and cannot be guaranteed to work.

A.1 DRAM (70nS)

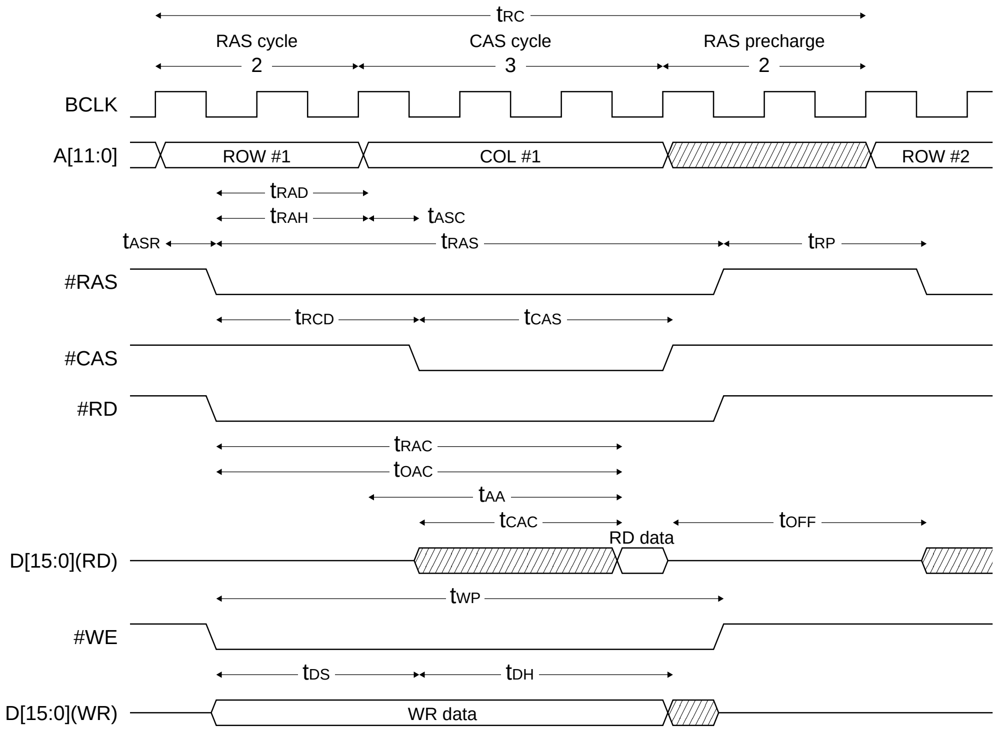
DRAM interface setup examples – 70nS

Operating frequency	RAS precharge cycle	RAS cycle	CAS cycle	Refresh RAS pulse width	Refresh RPC delay
20MHz	2	1	2	2	1
25MHz	2	1	2	2	1
33MHz	2	2	3	3	1

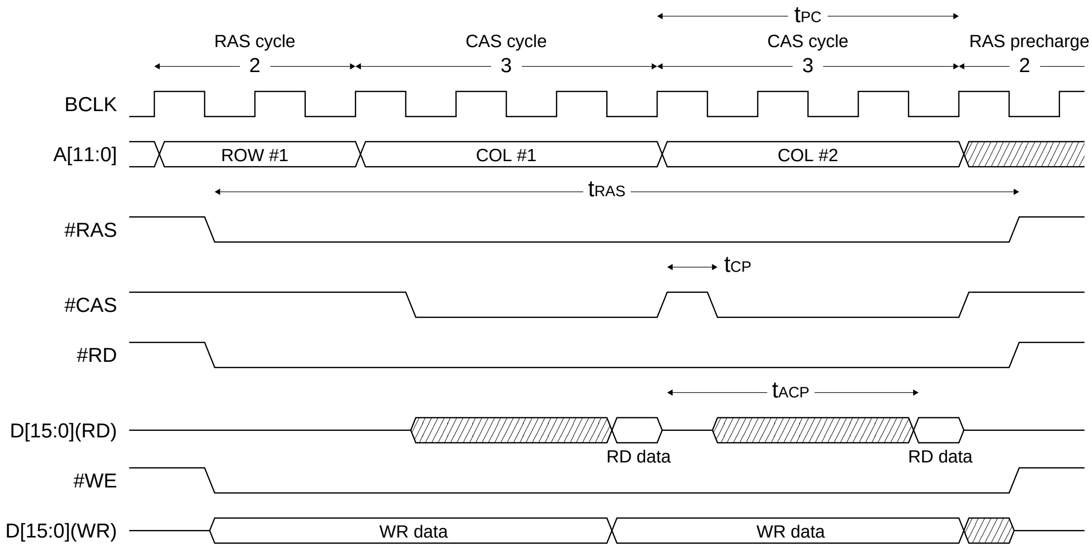
DRAM interface timing – 70nS

DRAM interface		Unit: nS		33MHz		25MHz		20MHz	
Parameter	Symbol	Min.	Max.	Cycle	Time	Cycle	Time	Cycle	Time
<Common parameters>									
Random read/random write cycle time	t_{RC}	130	–	7	210	5	200	5	250
#RAS precharge time	t_{RP}	50	–	2	60	2	80	2	100
#RAS pulse width	t_{RAS}	70	10000	5	150	3	120	3	150
#CAS pulse width	t_{CAS}	20	10000	2.5	75	1.5	60	1.5	75
Row address setup time	t_{ASR}	0	–	0.5	15	0.5	20	0.5	25
Row address hold time	t_{RAH}	10	–	1.5	45	0.5	20	0.5	25
Column address setup time	t_{ASC}	0	–	0.5	15	0.5	20	0.5	25
#RAS→#CAS delay time	t_{RCD}	20	–	2.0	60	1.0	40	1.0	50
#RAS→column address delay time	t_{RAD}	15	–	1.5	45	0.5	20	0.5	25
<Read-cycle parameters>									
#RAS access time	t_{RAC}	–	70	4.5	135	2.5	100	2.5	125
#CAS access time	t_{CAC}	–	20	2.5	75	1.5	60	1.5	75
Address access time	t_{AA}	–	35	3.0	90	2.0	80	2.0	100
#OE access time	t_{OAC}	–	20	4.5	135	2.5	100	2.5	125
Output buffer turn-off time	t_{OFF}	0	20	2	60	2	80	2	100
<Write-cycle parameters>									
Data input hold time	t_{DH}	15	–	2.5	75	1.5	60	1.5	75
<Fast-page mode>									
Fast-page mode cycle time	t_{PC}	45	–	3.0	90	2.0	80	2.0	100
Fast-page mode #CAS precharge time	t_{CP}	10	–	0.5	15	0.5	20	0.5	25
Access time after #CAS precharge	t_{ACP}	–	40	3.0	90	2.0	80	2.0	100
<Refresh cycle>									
#CAS setup time	t_{CSR}	10	–	1.0	30	1.0	40	1.0	50
#CAS hold time	t_{CHR}	10	–	2.5	75	1.5	60	1.5	75
#RAS precharge→#CAS hold time	t_{PPC}	10	–	1.0	30	1.0	40	1.0	50
#RAS pulse width (only in refresh cycle)	t_{RAS}	70	10000	3.0	90	2.0	80	2.0	100

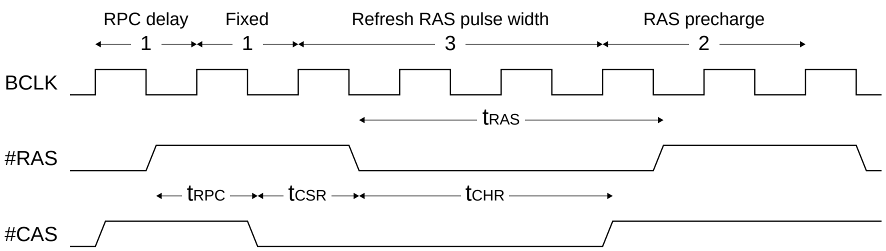
DRAM: 70nS, CPU: 33MHz, random read/write cycle



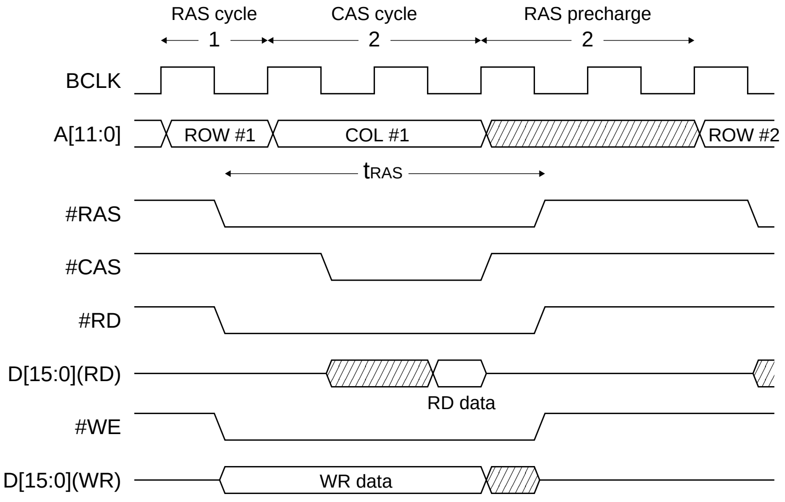
DRAM: 70nS, CPU: 33MHz, page-mode read/write cycle



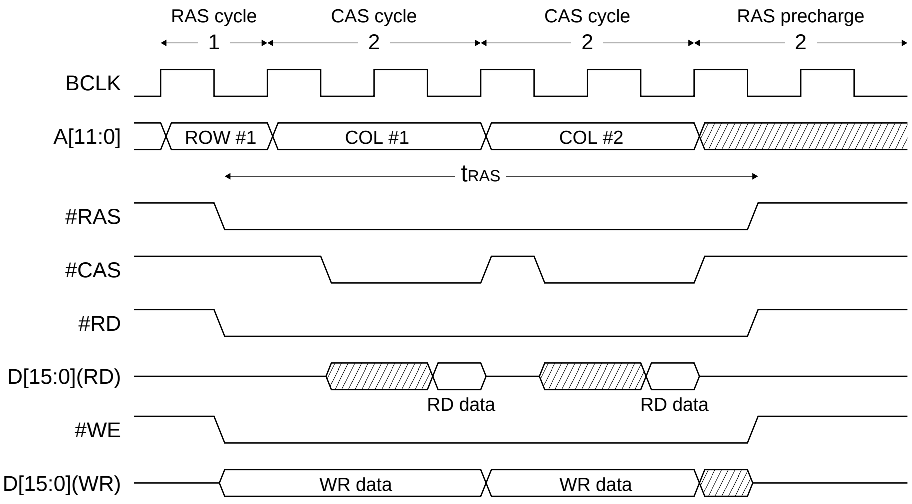
DRAM: 70nS, CPU: 33MHz, CAS-before-RAS refresh cycle



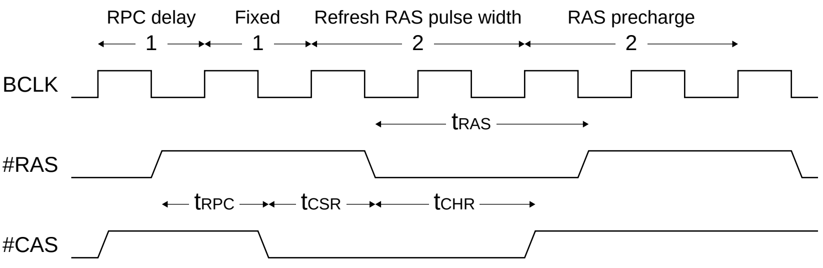
DRAM: 70nS, CPU: 25/20MHz, random read/write cycle



DRAM: 70nS, CPU: 25/20MHz, page-mode read/write cycle



DRAM: 70nS, CPU: 25/20MHz, CAS-before-RAS refresh cycle



A.2 DRAM (60nS)

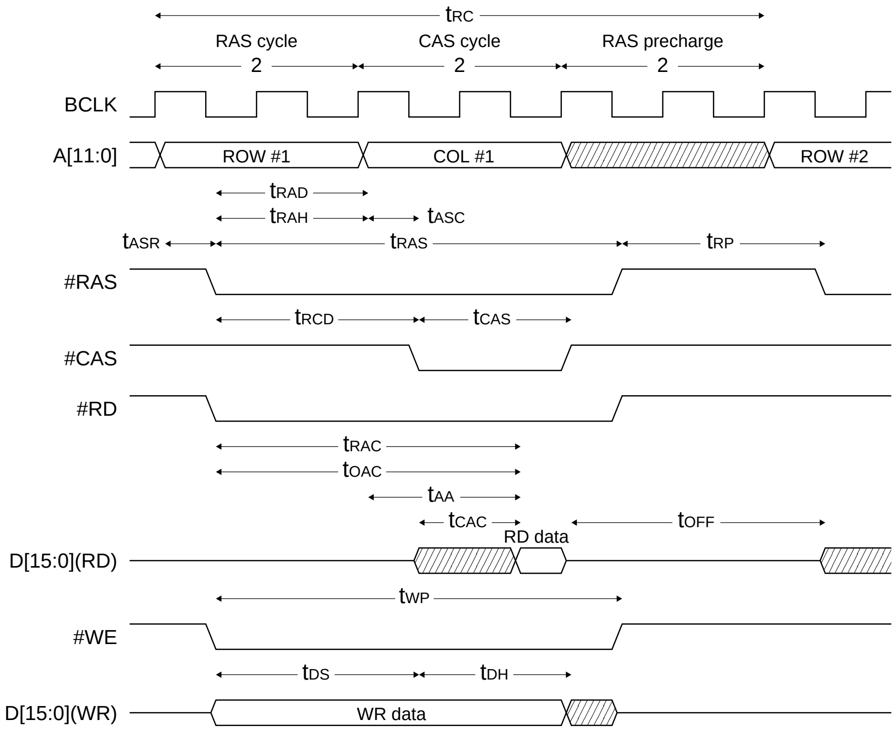
DRAM interface setup examples – 60nS

Operating frequency	RAS precharge cycle	RAS cycle	CAS cycle	Refresh RAS pulse width	Refresh RPC delay
20MHz	1	1	2	2	1
25MHz	2	1	2	2	1
33MHz	2	2	2	3	1

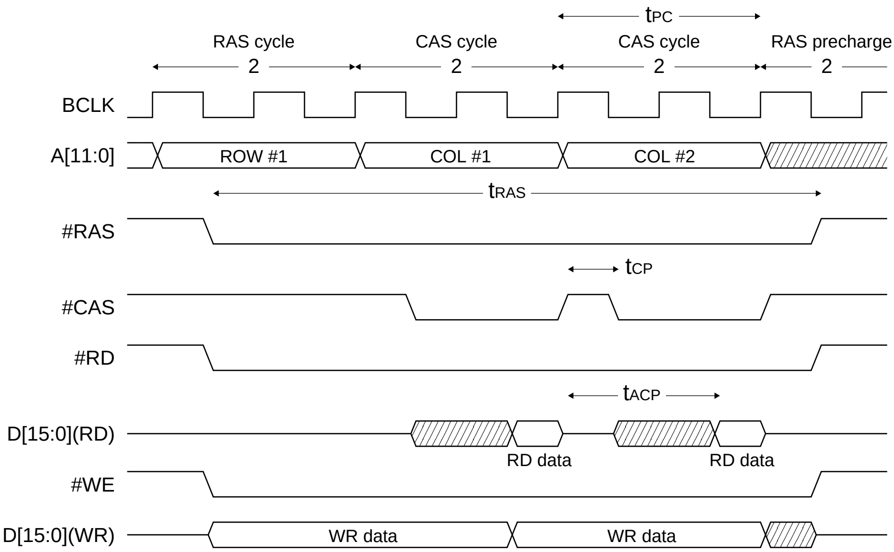
DRAM interface timing – 60nS

DRAM interface		Unit: nS		33MHz		25MHz		20MHz	
Parameter	**Symbol**	**Min.**	**Max.**	**Cycle**	**Time**	**Cycle**	**Time**	**Cycle**	**Time**
<Common parameters>									
Random read/random write cycle time	t_{RC}	110	–	6	180	5	200	4	200
#RAS precharge time	t_{RP}	40	–	2	60	2	80	1	50
#RAS pulse width	t_{RAS}	60	10000	4	120	3	120	3	150
#CAS pulse width	t_{CAS}	15	10000	1.5	45	1.5	60	1.5	75
Row address setup time	t_{ASR}	0	–	0.5	15	0.5	20	0.5	25
Row address hold time	t_{RAH}	10	–	1.5	45	0.5	20	0.5	25
Column address setup time	t_{ASC}	0	–	0.5	15	0.5	20	0.5	25
#RAS→#CAS delay time	t_{RCD}	20	–	2.0	60	1.0	40	1.0	50
#RAS→column address delay time	t_{RAD}	15	–	1.5	45	0.5	20	0.5	25
<Read-cycle parameters>									
#RAS access time	t_{RAC}	–	60	3.5	105	2.5	100	2.5	125
#CAS access time	t_{CAC}	–	15	1.5	45	1.5	60	1.5	75
Address access time	t_{AA}	–	30	2.0	60	2.0	80	2.0	100
#OE access time	t_{OAC}	–	15	3.5	105	2.5	100	2.5	125
Output buffer turn-off time	t_{OFF}	0	15	2	60	2	80	1	50
<Write-cycle parameters>									
Data input hold time	t_{DH}	10	–	1.5	45	1.5	60	1.5	75
<Fast-page mode>									
Fast-page mode cycle time	t_{PC}	40	–	2.0	60	2.0	80	2.0	100
Fast-page mode #CAS precharge time	t_{CP}	10	–	0.5	15	0.5	20	0.5	25
Access time after #CAS precharge	t_{ACP}	–	35	2.0	60	2.0	80	2.0	100
<Refresh cycle>									
#CAS setup time	t_{CSR}	10	–	1.0	30	1.0	40	1.0	50
#CAS hold time	t_{CHR}	10	–	2.5	75	1.5	60	1.5	75
#RAS precharge→#CAS hold time	t_{PPC}	10	–	1.0	30	1.0	40	1.0	50
#RAS pulse width (only in refresh cycle)	t_{RAS}	60	10000	3.0	90	2.0	80	2.0	100

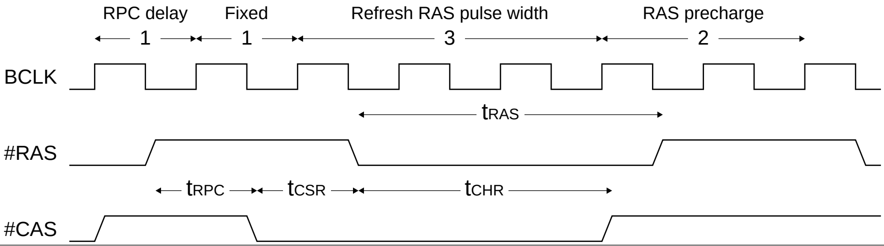
DRAM: 60nS, CPU: 33MHz, random read/write cycle



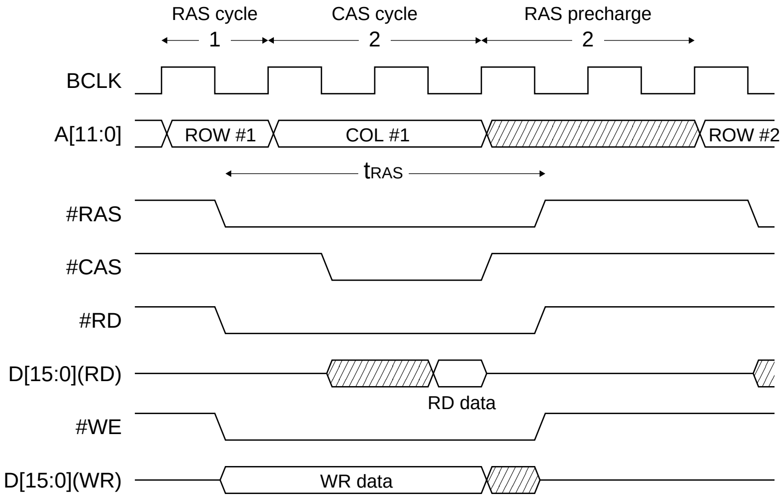
DRAM: 60nS, CPU: 33MHz, page-mode read/write cycle



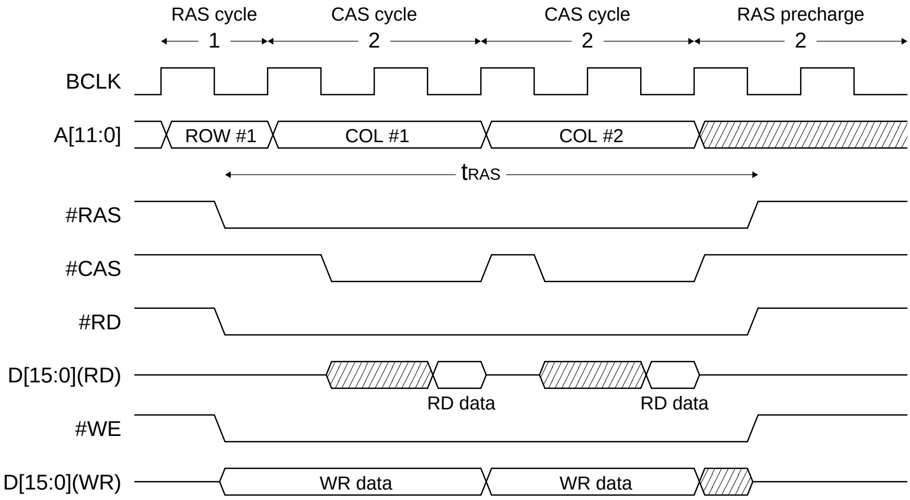
DRAM: 60nS, CPU: 33MHz, CAS-before-RAS refresh cycle



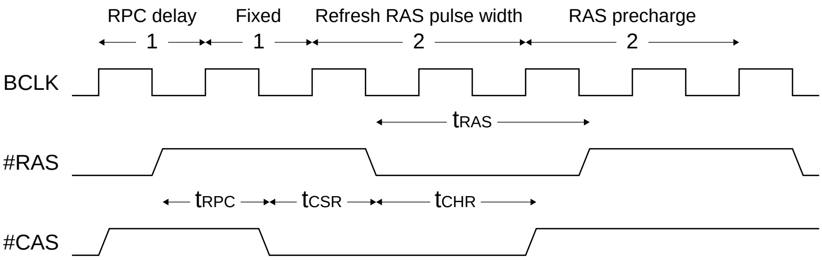
DRAM: 60nS, CPU: 25MHz, random read/write cycle



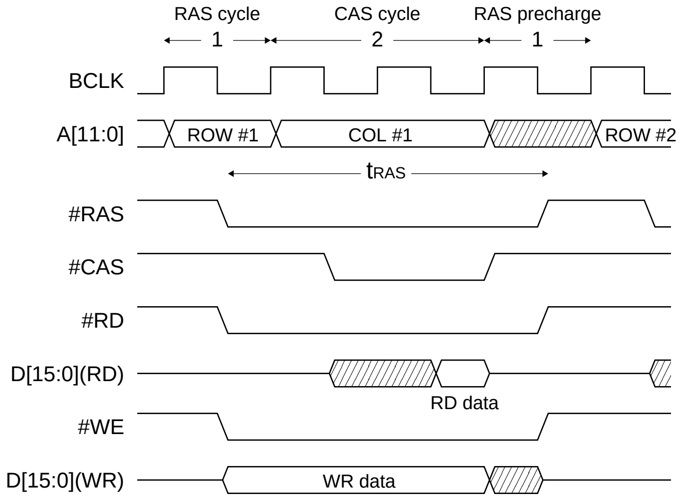
DRAM: 60nS, CPU: 25MHz, page-mode read/write cycle



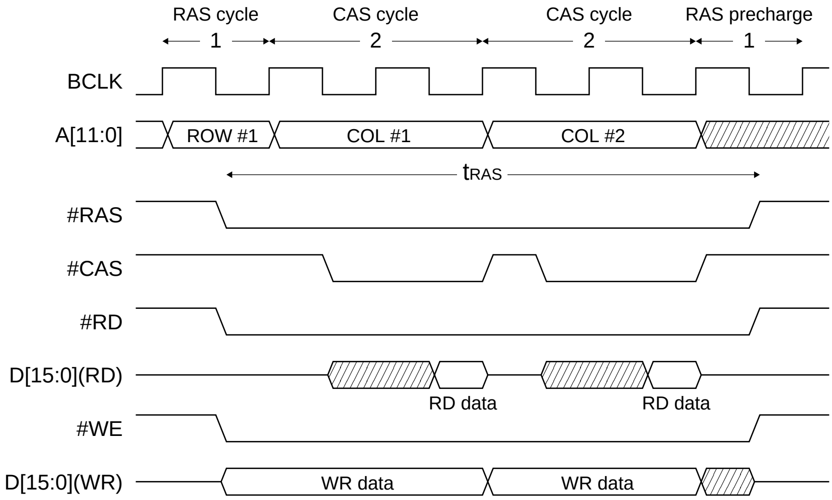
DRAM: 60nS, CPU: 25MHz, CAS-before-RAS refresh cycle



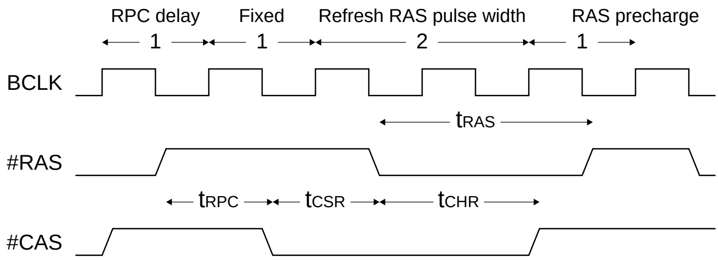
DRAM: 60nS, CPU: 20MHz, random read/write cycle



DRAM: 60nS, CPU: 20MHz, page-mode read/write cycle



DRAM: 60nS, CPU: 20MHz, CAS-before-RAS refresh cycle



A.3 ROM and Burst ROM

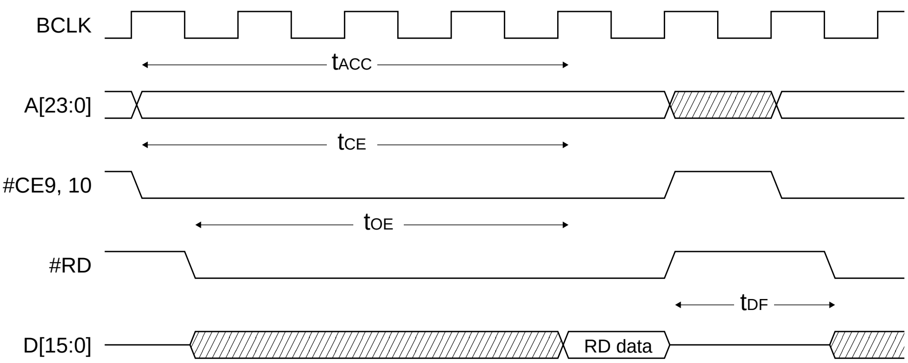
Burst ROM and mask ROM interface setup examples

Operating frequency	Normal read cycle		Burst read cycle		Output disable delay cycle
	Wait cycle	Read cycle	Wait cycle	Read cycle	
20MHz	2	3	1	2	1.5
25MHz	3	4	1	2	1.5
33MHz	4	5	2	3	1.5

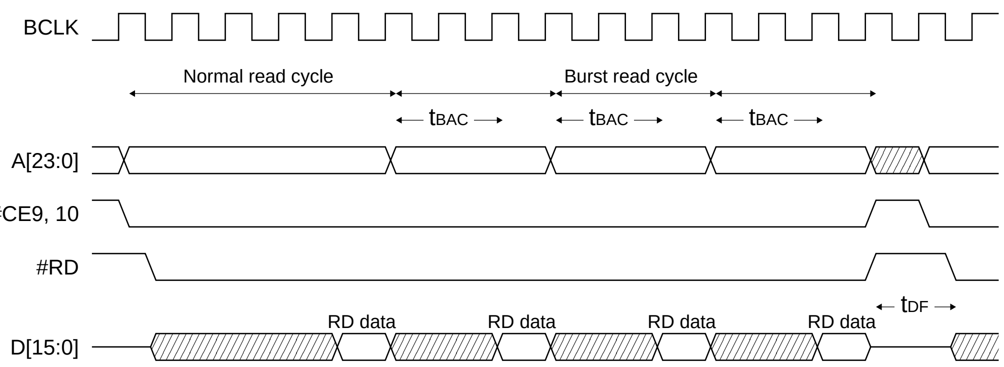
Burst ROM and mask ROM interface timing

Burst ROM and mask ROM interface				33MHz		25MHz		20MHz	
Parameter	Symbol	Min.	Max.	Cycle	Time	Cycle	Time	Cycle	Time
Access time	t_{ACC}	–	100	5	150	4	160	3	150
#CE output delay time	t_{CE}	–	100	5	150	4	160	3	150
#OE output delay time	t_{OE}	–	50	4.5	135	3.5	140	2.5	125
Burst access time	t_{BAC}	–	50	3	90	2	80	2	100
Output disable delay time	t_{DF}	0	40	1.5	45	1.5	60	1.5	75

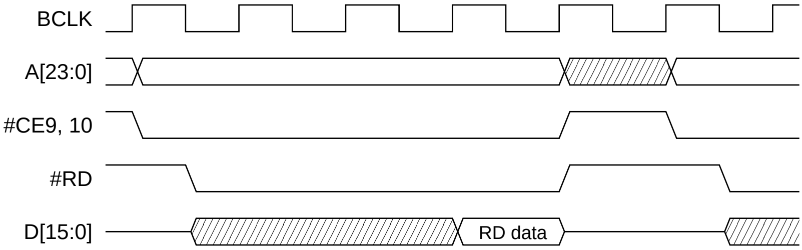
ROM: 100nS, CPU: 33MHz, normal read



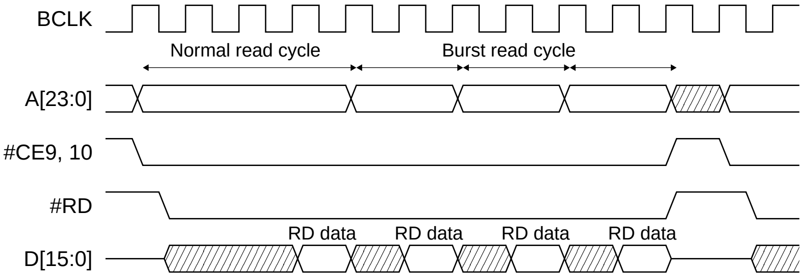
ROM: 100nS, CPU: 33MHz, burst read



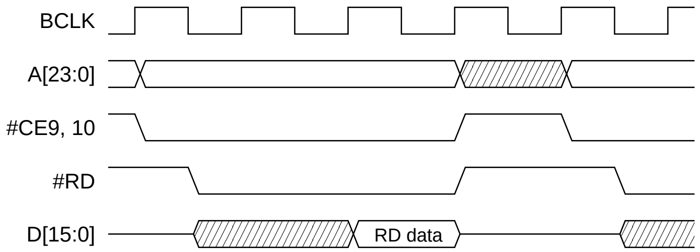
ROM: 100nS, CPU: 25MHz, normal read



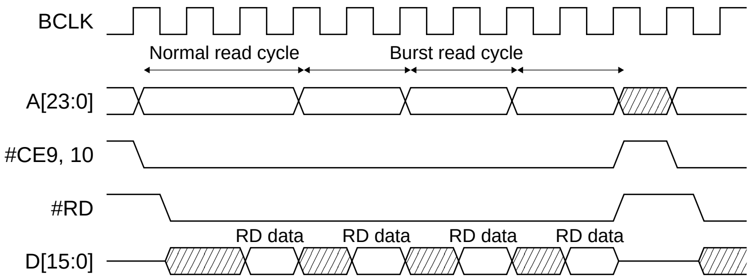
ROM: 100nS, CPU: 25MHz, burst read



ROM: 100nS, CPU: 20MHz, normal read



ROM: 100nS, CPU: 20MHz, burst read



A.4 SRAM (55nS)

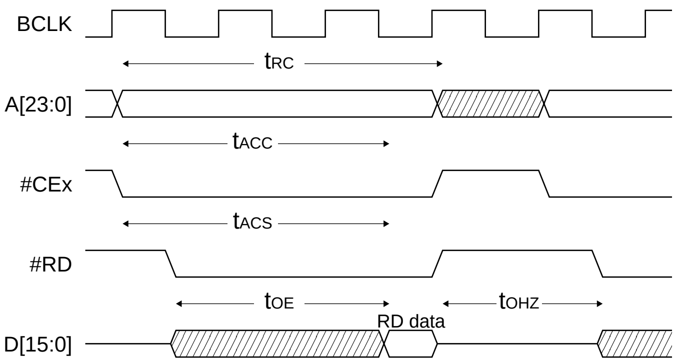
SRAM interface setup examples – 55nS

Operating frequency	Read cycle		Write cycle	Output disable delay time
	Wait cycle	Read cycle		
20MHz	1	2	2	1.5
25MHz	2	3	3	1.5
33MHz	2	3	3	1.5

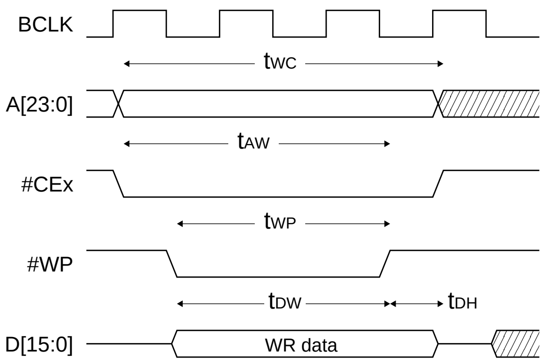
SRAM interface timing – 55nS

SRAM interface				33MHz		25MHz		20MHz	
Parameter	Symbol	Min.	Max.	Cycle	Time	Cycle	Time	Cycle	Time
<Read cycle>									
Read cycle time	t_{RC}	55	–	3	90	3	120	2	100
Address access time	t_{ACC}	–	55	3	90	3	120	2	100
#CE access time	t_{ACS}	–	55	3	90	3	120	2	100
#OE access time	t_{OE}	–	30	2.5	75	2.5	100	1.5	75
Output disable delay time	t_{OHZ}	0	30	1.5	45	1.5	60	1.5	75
<Write cycle>									
Write cycle time	t_{WC}	55	–	3	90	3	120	2	100
Address enable time	t_{AW}	50	–	2.5	75	2.5	100	1.5	75
Write pulse width	t_{WP}	45	–	2	60	2	80	1	50
Input data setup time	t_{DW}	30	–	2	60	2	80	1	50
Input data hold time	t_{DH}	0	–	0.5	15	0.5	20	0.5	25

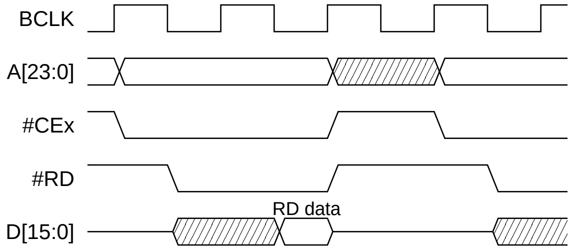
SRAM: 55nS, CPU: 33/25MHz, read cycle



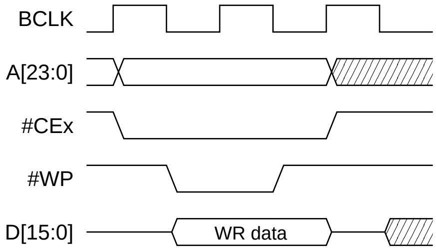
SRAM: 55nS, CPU: 33/25MHz, write cycle



SRAM: 55nS, CPU: 20MHz, read cycle



SRAM: 55nS, CPU: 20MHz, write cycle



A.5 SRAM (70nS)

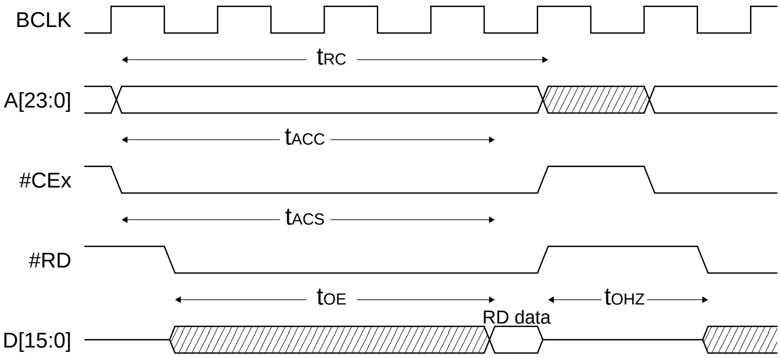
SRAM interface setup examples – 70nS

Operating frequency	Read cycle		Write cycle	Output disable delay time
	Wait cycle	Read cycle		
20MHz	2	3	3	1.5
25MHz	2	3	3	1.5
33MHz	3	4	4	1.5

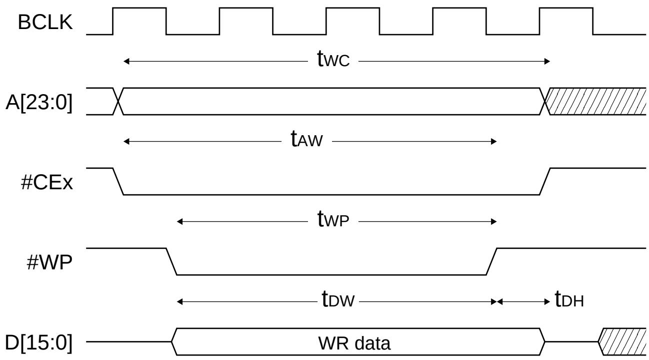
SRAM interface timing – 70nS

SRAM interface				33MHz		25MHz		20MHz	
Parameter	Symbol	Min.	Max.	Cycle	Time	Cycle	Time	Cycle	Time
<Read cycle>									
Read cycle time	t_{RC}	70	–	4	120	3	120	3	150
Address access time	t_{ACC}	–	70	4	120	3	120	3	150
#CE access time	t_{ACS}	–	70	4	120	3	120	3	150
#OE access time	t_{OE}	–	40	3.5	105	2.5	100	2.5	125
Output disable delay time	t_{OHZ}	0	30	1.5	45	1.5	60	1.5	75
<Write cycle>									
Write cycle time	t_{WC}	70	–	4	120	3	120	3	150
Address enable time	t_{AW}	60	–	3.5	105	2.5	100	2.5	125
Write pulse width	t_{WP}	55	–	3	90	2	80	2	100
Input data setup time	t_{DW}	30	–	3	90	2	80	2	100
Input data hold time	t_{DH}	0	–	0.5	15	0.5	20	0.5	25

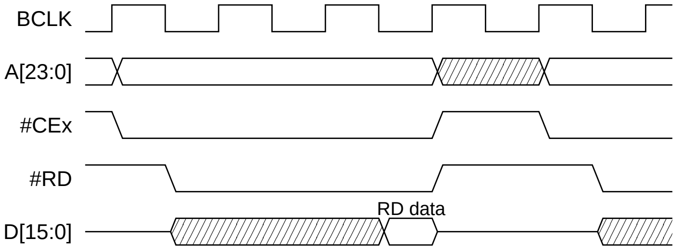
SRAM: 70nS, CPU: 33MHz, read cycle



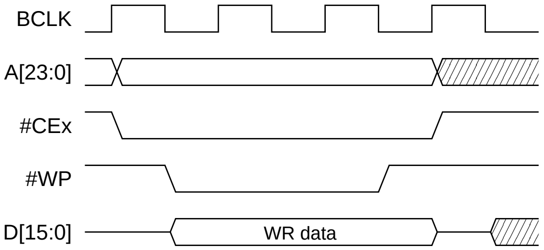
SRAM: 70nS, CPU: 33MHz, write cycle



SRAM: 70nS, CPU: 25/20MHz, read cycle



SRAM: 70nS, CPU: 25/20MHz, write cycle



A.6 8255A

8255A interface setup examples

Operating frequency	Read cycle: Wait cycle	Read cycle: Read cycle	Write cycle	Output disable delay time
20MHz	9 ∗1	10	10	3.5
25MHz	11	12	12	3.5
33MHz	14	15	15	3.5 ∗2

8255A interface timing

SRAM interface				33MHz		25MHz		20MHz	
Parameter	Symbol	Min.	Max.	Cycle	Time	Cycle	Time	Cycle	Time
<Read cycle>									
Read cycle time	t_{RC}	300	–	15	450	12	480	10	500
Address access time	t_{ACC}	–	250	15	450	12	480	10	500
#CE access time	t_{ACS}	–	250	15	450	12	480	10	500
#OE access time	t_{OE}	–	250	14.5	435	11.5	460	9.5	475
Output disable delay time	t_{OHZ}	10	150	3.5	105	3.5	140	3.5	175
<Write cycle>									
Write cycle time	t_{WC}	430	–	15	450	12	480	10	500
Address enable time	t_{AW}	400	–	14.5	435	11.5	460	9.5	475
Write pulse width	t_{WP}	400	–	14	420	11	440	9	450
Input data setup time	t_{DW}	100	–	14	420	11	440	9	450
Input data hold time ∗3	t_{DH}	30	–	0.5	15	0.5	20	0.5	25

∗1 The E0C33208/204/202 enables up to 7 cycles of wait-cycle insertion. If a number of wait cycles more than 7 cycles needs to be inserted, input the #WAIT signal from external hardware. Note that the interface must be set for SRAM type devices to insert wait cycles using the #WAIT pin. (Refer to "BCU (Bus Control Unit)" in the "C33 ASIC Macro Manual", for more information.)

∗2 This setting cannot satisfy the 150 nS of output-disable delay time specification required for the 8255A. When implementing such a low-speed device in the system, the external bus must be separated by inserting a 3-state bus buffer at the output side (when viewed from the CPU) of the external system bus.

∗3 If the data hold time that can be set is sufficient for the device, secure it by connecting a bus repeater to the external data bus D[15:0] or by inserting a latch at the output side of the external system interface.

Appendix B Pin Characteristics

Pin No.		Signal name	I/O cell name	Characteristic		Pull-up /down	Power supply	Remarks
QFP5	QFP15			Input	Output			
1	126	P24/TM2	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
2	127	VSS	XVSS					
3	128	P25/TM3	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
4	1	P26/TM4	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
5	2	P15/EXCL4/#DMAEND0	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
6	3	P27/TM5	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
7	4	BCLK	XHTB1T		Type1		VDDE	
8	5	P00/SIN0	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
9	6	P01/SOUT0	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
10	7	D15	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
11	8	VDD	XLVDD					
12	9	P03/#SRDY0	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
13	10	D14	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
14	11	P31/#BUSGET/#GARD	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
15	12	D13	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
16	13	P32/#DMAACK0	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
17	14	D12	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
18	15	P33/#DMAACK1	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
19	16	D11	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
20	17	K54/#DMAREQ3	XHIBHP2	CMOS/LVTTL SCHMITT		Pull-up	VDDE	
21	18	D10	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
22	19	K53/#DMAREQ2	XHIBHP2	CMOS/LVTTL SCHMITT		Pull-up	VDDE	
23	20	D9	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
24	21	K52/#ADTRG	XHIBHP2	CMOS/LVTTL SCHMITT		Pull-up	VDDE	
25	22	VSS	XVSS					
26	23	K51/#DMAREQ1	XHIBHP2	CMOS/LVTTL SCHMITT		Pull-up	VDDE	
27	24	P02/#SCLK0	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
28	25	D8	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
29	26	D7	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
30	27	VDDE	XHVDD					
31	28	K67/AD7	XHIBCLIN	CMOS/LVTTL			AVDDE	note2
32	29	K66/AD6	XHIBCLIN	CMOS/LVTTL			AVDDE	note2
33	30	K65/AD5	XHIBCLIN	CMOS/LVTTL			AVDDE	note2
34	31	K50/#DMAREQ0	XHIBHP2	CMOS/LVTTL SCHMITT		Pull-up	AVDDE	note2
35	32	K64/AD4	XHIBCLIN	CMOS/LVTTL			AVDDE	note2
36	33	K63/AD3	XHIBCLIN	CMOS/LVTTL			AVDDE	note2
37	34	K62/AD2	XHIBCLIN	CMOS/LVTTL			AVDDE	note2
38	35	AVDDE						
39	36	K61/AD1	XHIBCLIN	CMOS/LVTTL			AVDDE	note2
40	37	K60/AD0	XHIBCLIN	CMOS/LVTTL			AVDDE	note2
41	38	D6	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
42	39	VSS	XVSS					
43	40	D5	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
44	41	D4	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
45	42	D3	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
46	43	D2	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
47	44	D1	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
48	45	D0	XHBC1T	CMOS/LVTTL	Type1		VDDE	note1
49	46	P35/#BUSACK	XHBH1T	CMOS/LVTTL SCHMITT	Type1		VDDE	
50	47	VDDE	XHVDD					

Pin No.		Signal name	I/O cell	Characteristic		Pull-up	Power	Remarks
QFP5	QFP15		name	Input	Output	/down	supply	
51	48	#CE9/#CE17	XHBC1T	note4	Type1		V_{DDE}	
52	49	OSC2	XLLOT				V_{DD}	
53	50	#CE7/#RAS0/#CE13/#RAS2	XHBC1T	note4	Type1		V_{DDE}	
54	51	OSC1	XLLIN				V_{DD}	note3
55	52	#CE6	XHBC1T	note4	Type1		V_{DDE}	
56	53	#RD	XHBC1T	note4	Type1		V_{DDE}	
57	54	V_{SS}	XVSS					
58	55	#WRL/#WR/#WE/#LWE	XHBC1T	note4	Type1		V_{DDE}	
59	56	#WRH/#BSH/#UWE	XHBC1T	note4	Type1		V_{DDE}	
60	57	#CE10EX	XHBC1T	note4	Type1		V_{DDE}	
61	58	#CE8/#RAS1/#CE14/#RAS3	XHBC1T	note4	Type1		V_{DDE}	
62	59	#CE5/#CE15	XHBC1T	note4	Type1		V_{DDE}	
63	60	#CE4/#CE11	XHBC1T	note4	Type1		V_{DDE}	
64	61	P30/#WAIT/#CE4&5	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
65	62	#RESET	XHIBHP2	CMOS/LVTTL SCHMITT		Pull-up	V_{DDE}	
66	63	#NMI	XHIBHP2	CMOS/LVTTL SCHMITT		Pull-up	V_{DDE}	
67	64	A0/#BSL	XHBC1T	note4	Type1		V_{DDE}	
68	65	A1	XHBC1T	note4	Type1		V_{DDE}	
69	66	P34/#BUSREQ/#CE6	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
70	67	V_{SS}	XVSS					
71	68	A2	XHBC1T	note4	Type1		V_{DDE}	
72	69	A3	XHBC1T	note4	Type1		V_{DDE}	
73	70	A4	XHBC1T	note4	Type1		V_{DDE}	
74	71	A5	XHBC1T	note4	Type1		V_{DDE}	
75	72	A6	XHBC1T	note4	Type1		V_{DDE}	
76	73	#CE10IN	XHTB1T		Type1		V_{DDE}	
77	74	V_{DD}	XLVDD					
78	75	#EMEMRD	XHTB1T		Type1		V_{DDE}	
79	76	A7	XHBC1T	note4	Type1		V_{DDE}	
80	77	#HCAS/#UWE	XHTB1T		Type1		V_{DDE}	
81	78	A8	XHBC1T	note4	Type1		V_{DDE}	
82	79	#LCAS/#CAS	XHTB1T		Type1		V_{DDE}	
83	80	A9	XHBC1T	note4	Type1		V_{DDE}	
84	81	P16/EXCL5/#DMAEND1	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
85	82	A10	XHBC1T	note4	Type1		V_{DDE}	
86	83	A20	XHBC1T	note4	Type1		V_{DDE}	
87	84	A11	XHBC1T	note4	Type1		V_{DDE}	
88	85	A21	XHBC1T	note4	Type1		V_{DDE}	
89	86	A12	XHBC1T	note4	Type1		V_{DDE}	
90	87	A22	XHBC1T	note4	Type1		V_{DDE}	
91	88	A13	XHBC1T	note4	Type1		V_{DDE}	
92	89	A23	XHBC1T	note4	Type1		V_{DDE}	
93	90	V_{SS}	XVSS					
94	91	A14	XHBC1T	note4	Type1		V_{DDE}	
95	92	A15	XHBC1T	note4	Type1		V_{DDE}	
96	93	V_{DDE}	XHVDD					
97	94	A16	XHBC1T	note4	Type1		V_{DDE}	
98	95	ICEMD	XITST1			Pull-down		Testing pin
99	96	A17	XHBC1T	note4	Type1		V_{DDE}	
100	97	A18	XHBC1T	note4	Type1		V_{DDE}	

Pin No.		Signal name	I/O cell name	Characteristic		Pull-up /down	Power supply	Remarks
QFP5	QFP15			Input	Output			
101	98	A19	XHBC1T	note4	Type1		V_{DDE}	
102	99	P04/SIN1/#DMAACK2	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
103	100	P05/SOUT1/#DMAEND2	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
104	101	P06/#SCLK1/#DMAACK3	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
105	102	V_{SS}	XVSS					
106	103	PLLC	XLLIN					
107	104	V_{SS}	XVSS					
108	105	PLLS1	XHIBC	CMOS/LVTTL			V_{DDE}	
109	106	PLLS0	XHIBC	CMOS/LVTTL			V_{DDE}	
110	107	P07/#SRDY1/#DMAEND3	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
111	108	#X2SPD	XHIBC	CMOS/LVTTL			V_{DDE}	
112	109	EA10MD0	XHIBC	CMOS/LVTTL			V_{DDE}	
113	110	EA10MD1	XHIBCP2	CMOS/LVTTL		Pull-up	V_{DDE}	
114	111	V_{DD}	XLVDD					
115	112	EA3MD	XHIBCP2	CMOS/LVTTL	Type1	Pull-up	V_{DDE}	
116	113	OSC4	XLLOT				V_{DD}	
117	114	P20/#DRD	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
118	115	OSC3	XLLIN				V_{DD}	note3
119	116	P21/#DWE/#GAAS	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
120	117	#CE3	XHTB1T		Type1		V_{DDE}	
121	118	P22/TM0	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
122	119	P23/TM1	XHBH1T	CMOS/LVTTL SCHMITT	Type1		V_{DDE}	
123	120	DSIO	XLBH2P2T	CMOS/LVTTL SCHMITT	Type2	Pull-up	V_{DD}	note3
124	121	P10/EXCL0/T8UF0/DST0	XLBH2T	CMOS/LVTTL SCHMITT	Type2		V_{DD}	note3
125	122	P11/EXCL1/T8UF1/DST1	XLBH2T	CMOS/LVTTL SCHMITT	Type2		V_{DD}	note3
126	123	P12/EXCL2/T8UF2/DST2	XLBH2T	CMOS/LVTTL SCHMITT	Type2		V_{DD}	note3
127	124	P13/EXCL3/T8UF3/DPCO	XLBH2T	CMOS/LVTTL SCHMITT	Type2		V_{DD}	note3
128	125	P14/FOSC1/DCLK	XLBH2T	CMOS/LVTTL SCHMITT	Type2		V_{DD}	note3

note1) In the E0C33xxxF0A, the pins for the data bus are set for the TTL interface (I/O cell name: XHBT1T, output characteristics: Type1).
note2) The voltage applied to this pin must be $0V \leq V_{IN} \leq AV_{DDE}$.
Note that the input voltage range for the K50 pin differs from other K5 pins.
note3) The voltage applied to this pin must be $0V \leq V_{IN} \leq V_{DD}$.
note4) This pin is set as an input pin during device testing. Normally it is an output pin.

The following table lists output current characteristics.

Output current (I_{OL}/I_{OH})

	5.0V	3.3V	2.0V
Type1	3mA	2mA	0.6mA
Type2	–	6mA	2mA

E0C33 Family
ASIC Macro Manual

E0C33 Family ASIC Macro Manual

I OUTLINE

Table of Contents

I-1 INTRODUCTION

The C33 ASIC Macro is a library of macro cells including SEIKO EPSON original 32-bit CPU core E0C33000 and its various peripheral circuits. It is integrated on the SSL50000, SEIKO EPSON's 0.35 µm embedded ASIC Family. Since it shares common process technology, SRAM/ROM/Flash ASIC memory macro can also be integrated on the same chip. Thus SEIKO EPSON provides complete ASIC-microcomputer design environment.
The C33 CPU architecture is RISC type. So the CPU core size is small but the instruction set is very strong to achieve small compiled-code size.
The C33 ASIC Macro has the following features:

• Small CPU core:	25K gates
• Fast and high performance:	DC to 60 MHz operation
• Strong instruction set:	16-bit fixed length, 105 basic instructions
• Execution cycle:	Major instructions are executed in 1 cycle per instruction
• MAC function:	16 bits × 16 bits + 64 bits, 2 clock per MAC (25 MOPS in 50 MHz)
• Registers:	32 bits × 16 general registers and 32 bits × 5 special registers
• Memory space:	256M bytes (28 bits) linear space, code-data-IO shared type
• External bus I/F:	15 configurable memory areas Direct connection to external memory
• Interrupts:	Reset, NMI, up to 128 external interrupts, 4 software interrupts, 2 exceptions
• Reset, boot:	Cold reset, hot reset
• Power down mode:	Sleep, Halt
• Others:	Little endian (partial big endian can be configured) Harvard architecture (fetch, load/store parallel execution)
• User logic interface:	Programmable wait state (up to 7 cycles) #WAIT pin hand shake is possible. Large memory space for the user logic (up to 16M bytes) BCU configuration registers allow internal use of the external areas (Areas 4 to 18). Many interrupt requests from the user logic are acceptable.

THIS PAGE IS BLANK.

I-2 BLOCK DIAGRAM

The C33 Macro Library consists of five major blocks: C33 Core Block, C33 Peripheral Block, C33 Analog Block, C33 DMA Block and C33 Internal Memory Block.

Figure 2.1 shows the configuration of the C33 macro blocks.

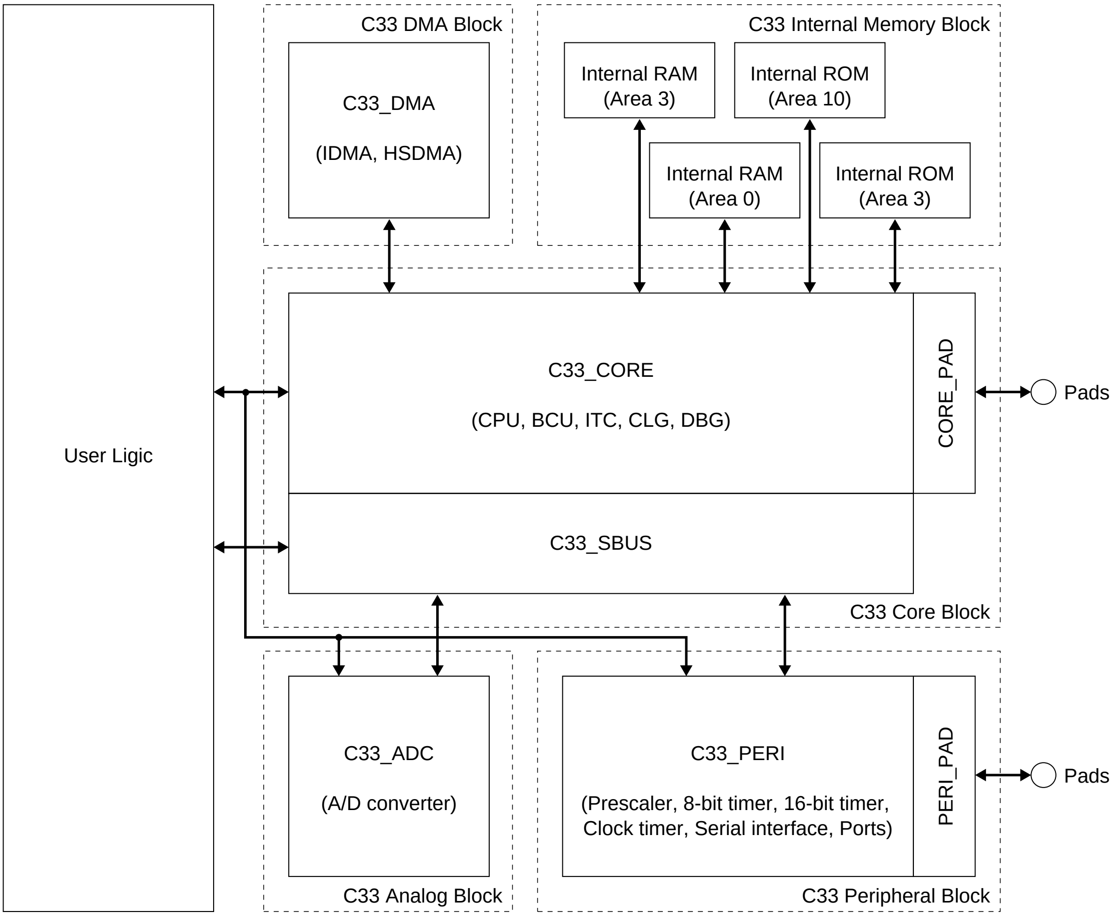


Figure 2.1 Block Configuration

C33 Core Block

The C33 Core Block consists of a functional block C33_CORE including CPU, BCU (Bus Control Unit), ITC (Interrupt Controller), CLG (Clock Generator) and DBG (Debug Unit), an I/O pad block for external interface, and an SBUS (Internal Silicon Integration Bus) for interfacing with on-chip Peripheral Macro Cells.
The C33 Core Block employs the E0C33000 32-bit RISC type CPU as the core CPU.

C33 Peripheral Block

The C33 Peripheral Block consists of a prescaler, four channels of 8-bit programmable timer, six channels of 16-bit programmable timer including watchdog timer function, two channels of serial interface, input and I/O ports, and a clock timer.

C33 Analog Block

The analog block consists of an A/D converter with eight input channels.

C33 DMA Block

The DMA block is configured with two types of DMA controllers: HSDMA (High-Speed DMA) that has on-chip registers for controlling DMA command information and IDMA (Intelligent DMA) that uses a memory area for storing DMA command information.

C33 Memory Block

For the internal memory space (Areas 0, 3 and 10), SRAM, Flash or mask ROM are provided. The memory size can be decided according to the system.

User Block

The user can design custom blocks using the SSL50000 Series ASIC libraries.

For details of the blocks, refer to the respective section in this manual.

I-3 LIST OF PINS

List of External I/O Pins

The following lists the external I/O pins of the C33 Core Block and Peripheral Block. Note that some pins are listed in two or more tables.

Table 3.1 List of Pins for External Bus Interface Signals

Pin name	I/O	Pull-up	Function
A0 #BSL	O	–	A0: Address bus (A0) when SBUSST(D3/0x4812E) = "0" (default) #BSL: Bus strobe (low byte) signal when SBUSST(D3/0x4812E) = "1"
A[23:1]	O	–	Address bus (A1 to A23)
D[15:0]	I/O	–	Data bus (D0 to D15)
#CE10EX	O	–	Area 10 chip enable for external memory ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE9+#CE10EX signal.
#CE10IN	O	–	Area 10 chip enable for internal ROM emulation memory
#CE9 #CE17	O	–	#CE9: Area 9 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" (default) #CE17: Area 17 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE17+#CE18 signal.
#CE8 #RAS1 #CE14 #RAS3	O	–	#CE8: Area 8 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" and A8DRA(D8/0x48128) = "0" (default) #RAS1: Area 8 DRAM row strobe when CEFUNC[1:0](D[A:9])/0x48130) = "00" and A8DRA(D8/0x48128) = "1" #CE14: Area 14 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" or "1x" and A14DRA(D8/0x48122) = "0" #RAS3: Area 14 DRAM row strobe when CEFUNC[1:0](D[A:9])/0x48130) = "01" or "1x" and A14DRA(D8/0x48122) = "1"
#CE7 #RAS0 #CE13 #RAS2	O	–	#CE7: Area 7 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" and A7DRA(D7/0x48128) = "0" (default) #RAS0: Area 7 DRAM row strobe when CEFUNC[1:0](D[A:9])/0x48130) = "00" and A7DRA(D7/0x48128) = "1" #CE13: Area 13 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" or "1x" and A13DRA(D7/0x48122) = "0" #RAS2: Area 13 DRAM row strobe when CEFUNC[1:0](D[A:9])/0x48130) = "01" or "1x" and A13DRA(D7/0x48122) = "1"
#CE6	O	–	Area 6 chip enable ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE7+#CE8 signal.
#CE5 #CE15	O	–	#CE5: Area 5 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" (default) #CE15: Area 15 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE15+#CE16 signal.
#CE4 #CE11	O	–	#CE4: Area 4 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "00" (default) #CE11: Area 11 chip enable when CEFUNC[1:0](D[A:9])/0x48130) = "01" ∗ When CEFUNC[1:0] = "1x", this pin outputs #CE11+#CE12 signal.
#CE3	O	–	Area 3 chip enable
#RD	O	–	Read signal
#EMEMRD	O	–	Read signal for internal ROM emulation memory
#WRL #WR #WE	O	–	#WRL: Write (low byte) signal when SBUSST(D3/0x4812E) = "0" (default) #WR: Write signal when SBUSST(D3/0x4812E) = "1" #WE: DRAM write signal (default)
#WRH #BSH	O	–	#WRH: Write (high byte) signal when SBUSST(D3/0x4812E) = "0" (default) #BSH: Bus strobe (high byte) signal when SBUSST(D3/0x4812E) = "1"
#HCAS	O	–	#HCAS: DRAM column address strobe (high byte) signal
#LCAS	O	–	#LCAS: DRAM column address strobe (low byte) signal
BCLK	O	–	Bus clock output
P34 #BUSREQ #CE6	I	–	P34: I/O port when CFP34(D4/0x402DC) = "0" (default) #BUSREQ: Bus release request input when CFP34(D4/0x402DC) = "1" and IOC34(D4/0x402DE) = "0" #CE6: Area 6 chip enable when CFP34(D4/0x402DC) = "1" and IOC34(D4/0x402DE) = "1"
P35 #BUSACK	I/O	–	P35: I/O port when CFP35(D5/0x402DC) = "0" (default) #BUSACK: Bus acknowledge output when CFP35(D5/0x402DC) = "1"

Pin name	I/O	Pull-up	Function
P30 #WAIT #CE4&5	I/O	–	P30: I/O port when CFP30(D0/0x402DC) = "0" (default) #WAIT: Wait cycle request input when CFP30(D0/0x402DC) = "1" and IOC30(D0/0x402DE) = "0" #CE4&5: Areas 4&5 chip enable when CFP30(D0/0x402DC) = "1" and IOC30(D0/0x402DE) = "1"
P20 #DRD	I/O	–	P20: I/O port when CFP20(D0/0x402D8) = "0" (default) #DRD: DRAM read signal output for successive RAS mode when CFP20(D0/0x402D8) = "1"
P21 #DWE #GAAS	I/O		P21: I/O port when CFP21(D1/0x402D8) = "0" and CFEX2(D2/0x402DF) = "0" (default) #DWE: DRAM write signal output for successive RAS mode when CFP21(D1/0x402D8) = "1" and CFEX2(D2/0x402DF) = "0" #GAAS: Area address strobe output for GA when CFEX2(D2/0x402DF) = "1"
P31 #BUSGET #GARD	I/O		P31: I/O port when CFP31(D1/0x402DC) = "0" and CFEX3(D3/0x402DF) = "0" (default) #BUSGET: Bus status monitor signal output for bus request when CFP31(D1/0x402DC) = "1" and CFEX3(D3/0x402DF) = "0" #GARD: Area read signal output for GA when CFEX3(D3/0x402DF) = "1"
EA10MD1	I	With pull-up	Area 10 boot mode selection EA10MD1 EA10MD0 Mode 1 1 External ROM mode 1 0 Internal ROM mode 0 1 OTP mode 0 0 Internal ROM emulation
EA10MD0	I	–	
EA3MD	I	With pull-up	Area 3 mode selection 1: Internal ROM mode, 0: Emulation mode

Table 3.2 List of Pins for HSDMA Control Signals

Pin name	I/O	Pull-up	Function
K50 #DMAREQ0	I	With pull-up	K50: Input port when CFK50(D0/0x402C0) = "0" (default) #DMAREQ0: HSDMA Ch. 0 request input when CFK50(D0/0x402C0) = "1"
K51 #DMAREQ1	I	With pull-up	K51: Input port when CFK51(D1/0x402C0) = "0" (default) #DMAREQ1: HSDMA Ch. 1 request input when CFK51(D1/0x402C0) = "1"
K53 #DMAREQ2	I	With pull-up	K53: Input port when CFK53(D3/0x402C0) = "0" (default) #DMAREQ2: HSDMA Ch. 2 request input when CFK53(D3/0x402C0) = "1"
K54 #DMAREQ3	I	With pull-up	K54: Input port when CFK54(D4/0x402C0) = "0" (default) #DMAREQ3: HSDMA Ch. 3 request input when CFK54(D4/0x402C0) = "1"
P32 #DMAACK0	I/O	–	P32: I/O port when CFP32(D2/0x402DC) = "0" (default) #DMAACK0: HSDMA Ch. 0 acknowledge output when CFP32(D2/0x402DC) = "1"
P33 #DMAACK1	I/O	–	P33: I/O port when CFP33(D3/0x402DC) = "0" (default) #DMAACK1: HSDMA Ch. 1 acknowledge output when CFP33(D3/0x402DC) = "1"
P04 SIN1 #DMAACK2	I/O	–	P04: I/O port when CFP04(D4/0x402D0) = "0" and CFEX4(D4/0x402DF) = "0" (default) SIN1: Serial I/F Ch. 1 data input when CFP04(D4/0x402D0) = "1" and CFEX4(D4/0x402DF) = "0" #DMAACK2: HSDMA Ch. 2 acknowledge output when CFEX4(D4/0x402DF) = "1"
P06 #SCLK1 #DMAACK3	I/O	–	P06: I/O port when CFP06(D6/0x402D0) = "0" and CFEX6(D6/0x402DF) = "0"(default) #SCLK1: Serial I/F Ch. 1 clock input/output when CFP06(D6/0x402D0) = "1" and CFEX6(D6/0x402DF) = "0" #DMAACK3: HSDMA Ch. 3 acknowledge output when CFEX6(D6/0x402DF) = "1"
P15 EXCL4 #DMAEND0	I/O	–	P15: I/O port when CFP15(D5/0x402D4) = "0" (default) EXCL4: 16-bit timer 4 event counter input when CFP15(D5/0x402D4) = "1" and IOC15(D5/0x402D6) = "0" #DMAEND0: HSDMA Ch. 0 end-of-transfer signal output when CFP15(D5/0x402D4) = "1" and IOC15(D5/0x402D6) = "1"
P16 EXCL5 #DMAEND1	I/O	–	P16: I/O port when CFP16(D6/0x402D4) = "0" (default) EXCL5: 16-bit timer 5 event counter input when CFP16(D6/0x402D4) = "1" and IOC16(D6/0x402D6) = "0" #DMAEND1: HSDMA Ch. 1 end-of-transfer signal output when CFP16(D6/0x402D4) = "1" and IOC16(D6/0x402D6) = "1"
P05 SOUT1 #DMAEND2	I/O	–	P05: I/O port when CFP05(D5/0x402D0) = "0" and CFEX5(D5/0x402DF) = "0" (default) SOUT1: Serial I/F Ch. 1 data output when CFP05(D5/0x402D0) = "1" and CFEX5(D5/0x402DF) = "0" #DMAEND2: HSDMA Ch. 2 end-of-transfer signal output when CFEX5(D5/0x402DF) = "1"
P07 #SRDY1 #DMAEND3	I/O	–	P07: I/O port when CFP07(D7/0x402D0) = "0" and CFEX7(D7/0x402DF) = "0" (default) #SRDY1: Serial I/F Ch. 1 ready signal output when CFP07(D7/0x402D0) = "1" and CFEX5(D5/0x402DF) = "0" #DMAEND3: HSDMA Ch. 3 end-of-transfer signal output when CFEX7(D7/0x402DF) = "1"

Table 3.3 List of Pins for Internal Peripheral Circuits

Pin name	I/O	Pull-up	Function
K50 #DMAREQ0	I	With pull-up	K50: Input port when CFK50(D0/0x402C0) = "0" (default) #DMAREQ0: HSDMA Ch. 0 request input when CFK50(D0/0x402C0) = "1"
K51 #DMAREQ1	I	With pull-up	K51: Input port when CFK51(D1/0x402C0) = "0" (default) #DMAREQ1: HSDMA Ch. 1 request input when CFK51(D1/0x402C0) = "1"
K52 #ADTRG	I	With pull-up	K52: Input port when CFK52(D2/0x402C0) = "0" (default) #ADTRG: A/D converter trigger input when CFK52(D2/0x402C0) = "1"
K53 #DMAREQ2	I	With pull-up	K53: Input port when CFK53(D3/0x402C0) = "0" (default) #DMAREQ2: HSDMA Ch. 2 request input when CFK53(D3/0x402C0) = "1"
K54 #DMAREQ3	I	With pull-up	K54: Input port when CFK54(D4/0x402C0) = "0" (default) #DMAREQ3: HSDMA Ch. 3 request input when CFK54(D4/0x402C0) = "1"
K60 AD0	I	–	K60: Input port when CFK60(D0/0x402C3) = "0" (default) AD0: A/D converter Ch. 0 input when CFK60(D0/0x402C3) = "1"
K61 AD1	I	–	K61: Input port when CFK61(D1/0x402C3) = "0" (default) AD1: A/D converter Ch. 1 input when CFK61(D1/0x402C3) = "1"
K62 AD2	I	–	K62: Input port when CFK62(D2/0x402C3) = "0" (default) AD2: A/D converter Ch. 2 input when CFK62(D2/0x402C3) = "1"
K63 AD3	I	–	K63: Input port when CFK63(D3/0x402C3) = "0" (default) AD3: A/D converter Ch. 3 input when CFK63(D3/0x402C3) = "1"
K64 AD4	I	–	K64: Input port when CFK64(D4/0x402C3) = "0" (default) AD4: A/D converter Ch. 4 input when CFK64(D4/0x402C3) = "1"
K65 AD5	I	–	K65: Input port when CFK65(D5/0x402C3) = "0" (default) AD5: A/D converter Ch. 5 input when CFK65(D5/0x402C3) = "1"
K66 AD6	I	–	K66: Input port when CFK66(D6/0x402C3) = "0" (default) AD6: A/D converter Ch. 6 input when CFK60(D6/0x402C3) = "1"
K67 AD7	I	–	K67: Input port when CFK67(D7/0x402C3) = "0" (default) AD7: A/D converter Ch. 7 input when CFK67(D7/0x402C3) = "1"
P00 SIN0	I/O	–	P00: I/O port when CFP00(D0/0x402D0) = "0" (default) SIN0: Serial I/F Ch. 0 data input when CFP00(D0/0x402D0) = "1"
P01 SOUT0	I/O	–	P01: I/O port when CFP01(D1/0x402D0) = "0" (default) SOUT0: Serial I/F Ch. 0 data output when CFP01(D1/0x402D0) = "1"
P02 #SCLK0	I/O	–	P02: I/O port when CFP02(D2/0x402D0) = "0" (default) #SCLK0: Serial I/F Ch. 0 clock input/output when CFP02(D2/0x402D0) = "1"
P03 #SRDY0	I/O	–	P03: I/O port when CFP03(D3/0x402D0) = "0" (default) #SRDY0: Serial I/F Ch. 0 ready signal output when CFP03(D3/0x402D0) = "1"
P04 SIN1 #DMAACK2	I/O	–	P04: I/O port when CFP04(D4/0x402D0) = "0" and CFEX4(D4/0x402DF) = "0" (default) SIN1: Serial I/F Ch. 1 data input when CFP04(D4/0x402D0) = "1" and CFEX4(D4/0x402DF) = "0" #DMAACK2: HSDMA Ch. 2 acknowledge output when CFEX4(D4/0x402DF) = "1"
P05 SOUT1 #DMAEND2	I/O	–	P05: I/O port when CFP05(D5/0x402D0) = "0" and CFEX5(D5/0x402DF) = "0" (default) SOUT1: Serial I/F Ch. 1 data output when CFP05(D5/0x402D0) = "1" and CFEX5(D5/0x402DF) = "0" #DMAEND2: HSDMA Ch. 2 end-of-transfer signal output when CFEX5(D5/0x402DF) = "1"
P06 #SCLK1 #DMAACK3	I/O	–	P06: I/O port when CFP06(D6/0x402D0) = "0" and CFEX6(D6/0x402DF) = "0" (default) #SCLK1: Serial I/F Ch. 1 clock input/output when CFP06(D6/0x402D0) = "1" and CFEX6(D6/0x402DF) = "0" #DMAACK3: HSDMA Ch. 3 acknowledge output when CFEX6(D6/0x402DF) = "1"
P07 #SRDY1 #DMAEND3	I/O	–	P07: I/O port when CFP07(D7/0x402D0) = "0" and CFEX7(D7/0x402DF) = "0" (default) #SRDY1: Serial I/F Ch. 1 ready signal output when CFP07(D7/0x402D0) = "1" and CFEX5(D5/0x402DF) = "0" #DMAEND3: HSDMA Ch. 3 end-of-transfer signal output when CFEX7(D7/0x402DF) = "1"
P10 EXCL0 T8UF0 DST0	I/O	–	P10: I/O port when CFP10(D0/0x402D4) = "0" and CFEX1(D1/0x402DF) = "0" EXCL0: 16-bit timer 0 event counter input when CFP10(D0/0x402D4) = "1", IOC10(D0/0x402D6) = "0" and CFEX1(D1/0x402DF) = "0" T8UF0: 8-bit timer 0 output when CFP10(D0/0x402D4) = "1", IOC10(D0/0x402D6) = "1" and CFEX1(D1/0x402DF) = "0" DST0: DST0 signal output when CFEX1(D1/0x402DF) = "1" (default)

Pin name	I/O	Pull-up	Function
P11 EXCL1 T8UF1 DST1	I/O	–	P11: I/O port when CFP11(D1/0x402D4) = "0" and CFEX1(D1/0x402DF) = "0" EXCL1: 16-bit timer 1 event counter input when CFP11(D1/0x402D4) = "1", IOC11(D1/0x402D6) = "0" and CFEX1(D1/0x402DF) = "0" T8UF1: 8-bit timer 1 output when CFP11(D1/0x402D4) = "1", IOC11(D1/0x402D6) = "1" and CFEX1(D1/0x402DF) = "0" DST1: DST1 signal output when CFEX1(D1/0x402DF) = "1" (default)
P12 EXCL2 T8UF2 DST2	I/O	–	P12: I/O port when CFP12(D2/0x402D4) = "0" and CFEX0(D0/0x402DF) = "0" EXCL2: 16-bit timer 2 event counter input when CFP12(D2/0x402D4) = "1", IOC12(D2/0x402D6) = "0" and CFEX0(D0/0x402DF) = "0" T8UF2: 8-bit timer 2 output when CFP12(D2/0x402D4) = "1", IOC12(D2/0x402D6) = "1" and CFEX0(D0/0x402DF) = "0" DST2: DST2 signal output when CFEX0(D0/0x402DF) = "1" (default)
P13 EXCL3 T8UF3 DPCO	I/O	–	P13: I/O port when CFP13(D3/0x402D4) = "0" and CFEX1(D1/0x402DF) = "0" EXCL3: 16-bit timer 3 event counter input when CFP13(D3/0x402D4) = "1", IOC13(D3/0x402D6) = "0" and CFEX1(D1/0x402DF) = "0" T8UF3: 8-bit timer 3 output when CFP13(D3/0x402D4) = "1", IOC13(D3/0x402D6) = "1" and CFEX1(D1/0x402DF) = "0" DPCO: DPCO signal output when CFEX1(D1/0x402DF) = "1" (default)
P14 FOSC1 DCLK	I/O	–	P14: I/O port when CFP14(D4/0x402D4) = "0" and CFEX0(D0/0x402DF) = "0" FOSC1: OSC1 clock output when CFP14(D4/0x402D4) = "1" and CFEX0(D0/0x402DF) = "0" DCLK: DCLK signal output when CFEX0(D0/0x402DF) = "1" (default)
P15 EXCL4 #DMAEND0	I/O	–	P15: I/O port when CFP15(D5/0x402D4) = "0" (default) EXCL4: 16-bit timer 4 event counter input when CFP15(D5/0x402D4) = "1" and IOC15(D5/0x402D6) = "0" #DMAEND0: HSDMA Ch. 0 end-of-transfer signal output when CFP15(D5/0x402D4) = "1" and IOC15(D5/0x402D6) = "1"
P16 EXCL5 #DMAEND1	I/O	–	P16: I/O port when CFP16(D6/0x402D4) = "0" (default) EXCL5: 16-bit timer 5 event counter input when CFP16(D6/0x402D4) = "1" and IOC16(D6/0x402D6) = "0" #DMAEND1: HSDMA Ch. 1 end-of-transfer signal output when CFP16(D6/0x402D4) = "1" and IOC16(D6/0x402D6) = "1"
P20 #DRD	I/O	–	P20: I/O port when CFP20(D0/0x402D8) = "0" (default) #DRD: DRAM read signal output for successive RAS mode when CFP20(D0/0x402D8) = "1"
P21 #DWE #GAAS	I/O		P21: I/O port when CFP21(D1/0x402D8) = "0" and CFEX2(D2/0x402DF) = "0" (default) #DWE: DRAM write signal output for successive RAS mode when CFP21(D1/0x402D8) = "1" and CFEX2(D2/0x402DF) = "0" #GAAS: Area address strobe output for GA when CFEX2(D2/0x402DF) = "1"
P22 TM0	I/O	–	P22: I/O port when CFP22(D2/0x402D8) = "0" (default) TM0: 16-bit timer 0 output when CFP22(D2/0x402D8) = "1"
P23 TM1	I/O	–	P23: I/O port when CFP23(D3/0x402D8) = "0" (default) TM1: 16-bit timer 1 output when CFP23(D3/0x402D8) = "1"
P24 TM2	I/O	–	P24: I/O port when CFP24(D4/0x402D8) = "0" (default) TM2: 16-bit timer 2 output when CFP24(D4/0x402D8) = "1"
P25 TM3	I/O	–	P25: I/O port when CFP25(D5/0x402D8) = "0" (default) TM3: 16-bit timer 3 output when CFP25(D5/0x402D8) = "1"
P26 TM4	I/O	–	P26: I/O port when CFP26(D6/0x402D8) = "0" (default) TM4: 16-bit timer 4 output when CFP26(D6/0x402D8) = "1"
P27 TM5	I/O	–	P27: I/O port when CFP27(D7/0x402D8) = "0" (default) TM5: 16-bit timer 5 output when CFP27(D7/0x402D8) = "1"
P31 #BUSGET #GARD	I/O		P31: I/O port when CFP31(D1/0x402DC) = "0" and CFEX3(D3/0x402DF) = "0" (default) #BUSGET: Bus status monitor signal output for bus request when CFP31(D1/0x402DC) = "1" and CFEX3(D3/0x402DF) = "0" #GARD: Area read signal output for GA when CFEX3(D3/0x402DF) = "1"
P32 #DMAACK0	I/O	–	P32: I/O port when CFP32(D2/0x402DC) = "0" (default) #DMAACK0: HSDMA Ch. 0 acknowledge output when CFP32(D2/0x402DC) = "1"
P33 #DMAACK1	I/O	–	P33: I/O port when CFP33(D3/0x402DC) = "0" (default) #DMAACK1: HSDMA Ch. 1 acknowledge output when CFP33(D3/0x402DC) = "1"
P34 #BUSREQ #CE6	I	–	P34: I/O port when CFP34(D4/0x402DC) = "0" (default) #BUSREQ: Bus release request input when CFP34(D4/0x402DC) = "1" and IOC34(D4/0x402DE) = "0" #CE6: Area 6 chip enable when CFP34(D4/0x402DC) = "1" and IOC34(D4/0x402DE) = "1"
P35 #BUSACK	I/O	–	P35: I/O port when CFP35(D5/0x402DC) = "0" (default) #BUSACK: Bus acknowledge output when CFP35(D5/0x402DC) = "1"

Table 3.4 List of Pins for Clock Generator and Oscillation Circuits

Pin name	I/O	Pull-up	Function
OSC1	I	–	Low-speed (OSC1) oscillation input (32 kHz crystal oscillator or external clock input)
OSC2	O	–	Low-speed (OSC1) oscillation output
OSC3	I	–	High-speed (OSC3) oscillation input (crystal/ceramic oscillator or external clock input)
OSC4	O	–	High-speed (OSC3) oscillation output

PLLS[1:0]	I	–	PLL set-up pins
			PLLS1 / PLLS0 / fin (f_{OSC3}) / fout (f_{PSCIN})
			1 / 1 / 10–30MHz / 20–60MHz ∗1
			    10–25MHz / 20–50MHz ∗2
			0 / 1 / 10–15MHz / 40–60MHz ∗1
			    10–12.5MHz / 40–50MHz ∗2
			0 / 0 / PLL is not used / L
			∗1: ROM-less model with 3.3 V ± 0.3 V operating voltage
			∗2: ROM built-in model, or 3.0 V ± 0.3 V operating voltage

PLLC	–	–	Capacitor connecting pin for PLL

Table 3.5 List of Other Pins

Pin name	I/O	Pull-up/down	Function
ICEMD	I	With pull-down	High-impedance control input pin When this pin is set to High, all the output pins go into high-impedance state. This makes it possible to disable the E0C33 chip on the board.
DSIO	I/O	With pull-up	Serial I/O pin for debugging This pin is used to communicate with the debugging tool ICD33.
#X2SPD	I	–	Clock doubling mode set-up pin 1: CPU clock = bus clock x 1, 0: CPU clock = bus clock x 2
#NMI	I	With pull-up	NMI request input pin
#RESET	I	With pull-up	Initial reset input pin

Note: "#" in the pin names indicates that the signal is low active.

THIS PAGE IS BLANK.

E0C33 Family ASIC Macro Manual

II CORE BLOCK

Table of Contents

II-1 INTRODUCTION

The core block consists of a functional block C33_CORE including CPU, BCU (Bus Control Unit), ITC (Interrupt Controller), CLG (Clock Generator) and DBG (Debug Unit), an I/O pad block for external interface, and an SBUS (Internal Silicon Integration Bus) for interfacing with on-chip Peripheral Macro Cells.

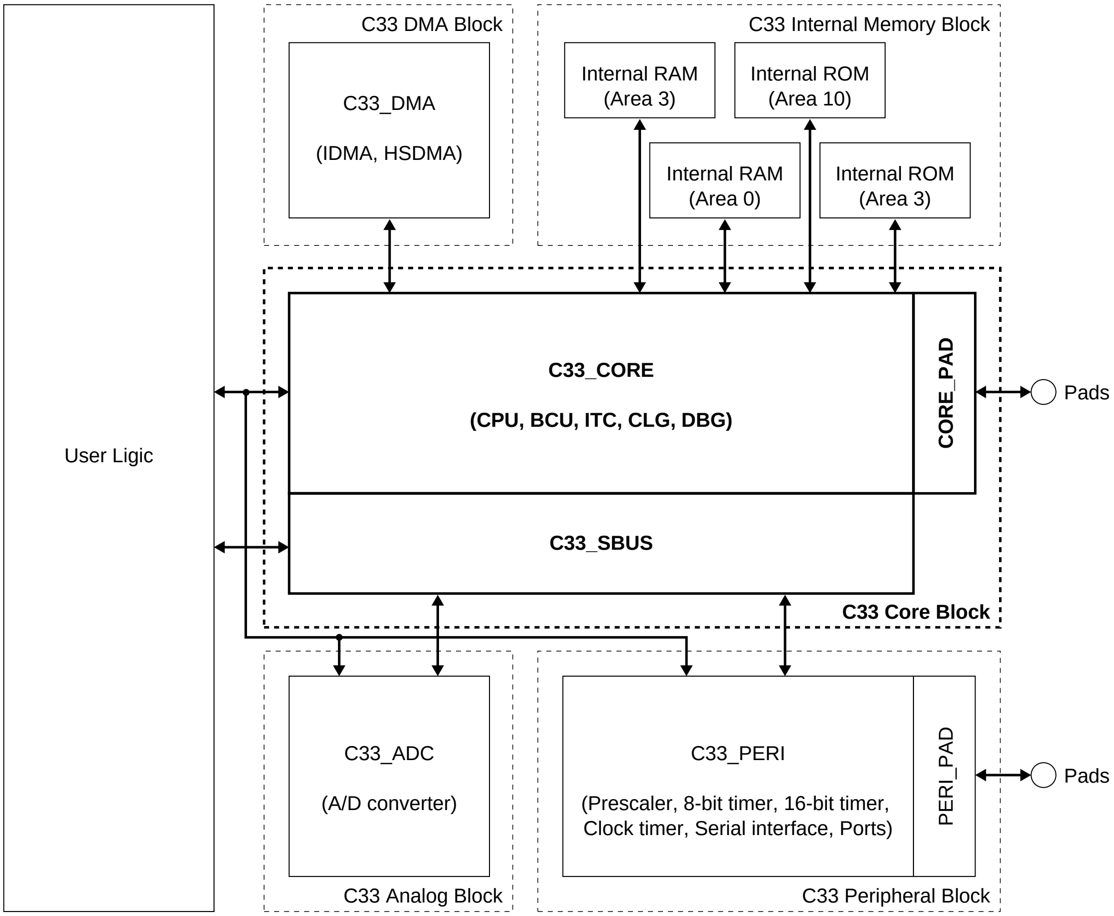


Figure 1.1 Core Block

THIS PAGE IS BLANK.

II-2 CPU AND OPERATING MODE

CPU

The C33 Core Block employs the E0C33000 32-bit RISC type CPU as the core CPU. Since it has a built-in multiplier, all instructions (105 instructions) in the E0C33000 instruction set including the MAC (multiplication and accumulation) instruction and the multiplication/division instructions are available.
All the internal registers of the E0C33000 can be used. The CPU registers and CPU address bus can handle 28-bit addresses. However, the core block has a 24-bit external address bus (A[0:23]), so the low-order 24 bits of address data can only be delivered to the external address bus and the internal address bus which is connected to the User Logic Block.

Refer to the "E0C33000 Core CPU Manual" for details of the E0C33000.

Standby Mode

The CPU supports three standby modes: two HALT modes and a SLEEP mode.
By setting the CPU in the standby mode, power consumption can greatly be reduced.

HALT Mode

When the CPU executes the halt instruction, it suspends the program execution and enters the HALT mode.
The CPU supports two types of HALT modes (basic HALT mode and HALT2 mode) and either can be selected using the HLT2OP (D3) / Clock option register (0x40190).
The CPU stops operating in basic HALT mode, so the amount of current consumption can be reduced. The internal peripheral circuits including the oscillation circuit keep operating in basic HALT mode.
HALT2 mode stops the external bus control functions including DMA and the bus clock as well as the CPU similar to basic HALT mode. Consequently, HALT2 mode realizes more power saving than the basic HALT mode.
The HALT mode is canceled by an initial reset or an interrupt including NMI. This mode is useful for saving power when waiting for an external input or completion of the peripheral circuit operations that do not need to execute the CPU.
The CPU transits to program execution status through trap processing when the HALT mode is canceled by an interrupt and executes the interrupt processing routine. The trap processing of the CPU saves the address of the instruction that follows the executed halt instruction into the stack. Therefore, when the interrupt processing routine is terminated by the reti instruction, the program flow returns to the instruction that follows the halt instruction.
Note that the HALT mode cannot be canceled with an interrupt factor except for reset and NMI if the PSR is set into interrupt disabled status.

SLEEP Mode

When the CPU executes the slp instruction, it suspends the program execution and enters SLEEP mode.
In SLEEP mode, the CPU and the internal peripheral circuits including the high-speed (OSC3) oscillation circuit stop operating. Thus SLEEP mode can greatly reduce current consumption in comparison to HALT mode.
Moreover, the low-speed (OSC1) oscillation circuit and clock timer do not stop operating. The clock function keeps operating in SLEEP mode.
SLEEP mode is canceled by an initial reset or an interrupt (NMI, clock timer interrupt, external interrupt such as a key entry). Note that other interrupts by the internal peripheral circuits that use the OSC3 clock cannot be used for canceling SLEEP mode.
The CPU transits to program execution status through trap processing when the SLEEP mode is canceled by an interrupt and executes the interrupt processing routine. The trap processing of the CPU saves the address of the instruction that follows the executed slp instruction into the stack. Therefore, when the interrupt processing routine is terminated by the reti instruction, the program flow returns to the instruction that follows the slp instruction.
Note that SLEEP mode cannot be canceled with an interrupt factor except for reset and NMI if the PSR is set into interrupt disabled status.

Notes on Standby Mode

Interrupts

The standby mode can be canceled by an interrupt. Therefore, it is necessary to enable the interrupt to be used for canceling the standby mode before setting the CPU in the standby mode. It is also necessary to set the IE (interrupt enable) and IL (interrupt level) bits in the PSR to a condition that can accept the interrupt. Otherwise, the standby mode cannot be canceled even when an interrupt occurs. Refer to "ITC (Interrupt Controller)", for interrupt settings.

Oscillation circuit

The high-speed (OSC3) oscillation circuit stops in SLEEP mode and restarts oscillating when SLEEP mode is canceled. If the CPU had operated with the OSC3 clock before entering SLEEP mode, the CPU restarts operating with the OSC3 clock immediately after canceling SLEEP mode. However, the OSC3 oscillation needs appropriate stabilization time (10 ms max. under the standard condition in 3.3 V). To restart the CPU after the oscillation stabilizes, a programmable interval can be inserted between cancellation of SLEEP mode and starting the CPU operation. Refer to "CLG (Clock Generator)", for details.

The oscillation start time of the high-speed (OSC3) oscillation circuit varies according to the components to be used, board pattern and operating environment. The interval must be set to allow enough margin.

BCU

When the CPU enters the standby mode, the BCU (bus control unit) stops after the current bus cycle has completed. All the chip enable signals are negated.
In basic HALT mode, the BCLK (bus clock) signal is output and DRAM refresh cycles are generated. DMA also operates.
In HALT2 or SLEEP mode, the BCLK signal stops, therefore DRAM refresh cycles cannot be generated and DMA stops.

Additional

The contents of the CPU registers and input/output port status are retained in the standby mode. Almost all control and data registers of the internal peripheral circuits are also retained, note, however, some registers may be changed at the transition to SLEEP mode. Refer to the section of each peripheral circuit for other precautions.

Test Mode

The C33 Core Block has the ICEMD pin for testing the chip. When this pin is set to High, the IC enters the following state:

- All output pins go into high-impedance state except for the clock output pins (OSC2: H, OSC4 H, PLLC: L).
- Clock inputs are disabled. OSC1, OSC3 and PLL stop operating. OSC2: H, OSC4 H, PLLC: L
- All the pull-up and pull-down resistors enter an inactive state.

Leave this pin open or connect to V_{SS} for normal operation. The ICEMD pin has a built-in pull-down resistor.

Debug Mode

The C33 Core Block supports the debug mode.
The debug mode is a CPU function, and realizes single step operation and break functions in the chip itself. Refer to the "E0C33000 Core CPU Manual" for details of the debug mode and the functions.

Area 2 in the memory map can only be accessed in the debug mode.
In the debug mode, the OSC3 clock is used as the CPU operating clock. Therefore, do not stop the high-speed (OSC3) oscillation circuit when using the debugging functions. Furthermore, only the C33 Core Block operates in the debug mode and other internal peripheral circuits stop operating.

Trap Table

Table 2.1 shows the trap table in the C33 Core. Refer to the "E0C33000 Core CPU Manual" for details of exceptions and Section II-5 in this manual, "ITC (Interrupt Controller)", for interrupts.

Table 2.1 Trap Table

HEX No.	Vector number (Hex address)	Exception/interrupt name	Exception/interrupt factor	IDMA Ch.	Priority
0	0(Base)	Reset	Low input to the reset pin	–	High ↑
	1–3	reserved	–	–	
4	4(Base+10)	Zero division	Division instruction	–	
5	5	reserved	–	–	
6	6(Base+18)	Address error exception	Memory access instruction	–	
7	0x0 or 0x60000	Debugging exception	brk instruction, etc.	–	
8	8(Base+1C)	NMI	Low input to the NMI pin	–	
	9–11	reserved	–	–	
C	12(Base+30)	Software exception 0	int instruction	–	
D	13(Base+34)	Software exception 1	int instruction	–	
E	14(Base+38)	Software exception 2	int instruction	–	
F	15(Base+3C)	Software exception 3	int instruction	–	
10	16(Base+40)	Port input interrupt 0	Edge (rising or falling) or level (High or Low)	1	
11	17(Base+44)	Port input interrupt 1	Edge (rising or falling) or level (High or Low)	2	
12	18(Base+48)	Port input interrupt 2	Edge (rising or falling) or level (High or Low)	3	
13	19(Base+4C)	Port input interrupt 3	Edge (rising or falling) or level (High or Low)	4	
14	20(Base+50)	Key input interrupt 0	Rising or falling edge	–	
15	21(Base+54)	Key input interrupt 1	Rising or falling edge	–	
16	22(Base+58)	High-speed DMA Ch.0	High-speed DMA Ch.0, end of transfer	5	
17	23(Base+5C)	High-speed DMA Ch.1	High-speed DMA Ch.1, end of transfer	6	
18	24(Base+60)	High-speed DMA Ch.2	High-speed DMA Ch.2, end of transfer	–	
19	25(Base+64)	High-speed DMA Ch.3	High-speed DMA Ch.3, end of transfer	–	
1A	26(Base+68)	IDMA	Intelligent DMA, end of transfer	–	
	27–29	reserved	–	–	
1E	30(Base+78)	16-bit programmable timer 0	Timer 0 comparison B	7	
1F	31(Base+7C)		Timer 0 comparison A	8	
	32–33	reserved	–	–	
22	34(Base+88)	16-bit programmable timer 1	Timer 1 comparison B	9	
23	35(Base+8C)		Timer 1 comparison A	10	
	36–37	reserved	–	–	
26	38(Base+98)	16-bit programmable timer 2	Timer 2 comparison B	11	
27	39(Base+9C)		Timer 2 comparison A	12	
	40–41	reserved	–	–	
2A	42(Base+A8)	16-bit programmable timer 3	Timer 3 comparison B	13	
2B	43(Base+AC)		Timer 3 comparison A	14	
	44–45	reserved	–	–	
2E	46(Base+B8)	16-bit programmable timer 4	Timer 4 comparison B	15	
2F	47(Base+BC)		Timer 4 comparison A	16	
	48–49	reserved	–	–	
32	50(Base+C8)	16-bit programmable timer 5	Timer 5 comparison B	17	
33	51(Base+CC)		Timer 5 comparison A	18	
34	52(Base+D0)	8-bit programmable timer	Timer 0 underflow	19	
35	53(Base+D4)		Timer 1 underflow	20	
36	54(Base+D8)		Timer 2 underflow	21	
37	55(Base+DC)		Timer 3 underflow	22	
38	56(Base+E0)	Serial interface Ch.0	Receive error	–	
39	57(Base+E4)		Receive buffer full	23	
3A	58(Base+E8)		Transmit buffer empty	24	
	59	reserved	–	–	
3C	60(Base+F0)	Serial interface Ch.1	Receive error	–	
3D	61(Base+F4)		Receive buffer full	25	
3E	62(Base+F8)		Transmit buffer empty	26	
	63	reserved	–	–	
40	64(Base+100)	A/D converter	A/D converter, end of conversion	27	
41	65(Base+104)	Clock timer	Falling edge of 32 Hz, 8 Hz, 2 Hz or 1 Hz signal 1-minuet, 1-hour or specified time count up	–	
	66–67	reserved	–	–	
44	68(Base+110)	Port input interrupt 4	Edge (rising or falling) or level (High or Low)	28	
45	69(Base+114)	Port input interrupt 5	Edge (rising or falling) or level (High or Low)	29	
46	70(Base+118)	Port input interrupt 6	Edge (rising or falling) or level (High or Low)	30	↓
47	71(Base+11C)	Port input interrupt 7	Edge (rising or falling) or level (High or Low)	31	Low

∗ Base = Set value in the TTBR register (0x48134 to 0x48137); 0xC00000 by default.

II-3 INITIAL RESET

Pins for Initial Reset

Table 3.1 shows the pins used for initial reset.

Table 3.1 Pins for Initial Reset

Pin name	I/O	Function
#RESET	I	Initial reset input pin (Low active) Low: Resets the CPU.
#MNI	I	NMI request input pin This pin is also used for selecting a reset method. High: Cold start Low: Hot start

The chip is reset when the #RESET pin goes low and starts operating at the rising edge of the reset signal. The CPU and internal peripheral circuits are initialized while the #RESET pin is low.

Cold Start and Hot Start

The CPU supports two initial reset methods: cold start and hot start. The #MNI pin is used with the #RESET pin to set this condition.
The differences between cold start and hot start are shown in Table 3.2.

Table 3.2 Differences between Cold Start and Hot Start

Setup contents	Cold start	Hot start
Reset condition	#RESET = low & #MNI = high	#RESET = low & #MNI = low
CPU: PC	The vector at the boot address is loaded to the PC.	
CPU: PSR	All the PSR bits are reset to 0.	
CPU: Other registers	Undefined	
CPU: Operating clock	The CPU operates with the OSC3 clock.	
External bus status (0x48120–0x4813F)	Initialized	Status is retained.
Oscillation circuit	Both the OSC1 and OSC3 circuits start oscillating.	
I/O pin status (0x402C0–0x402DF)	Initialized	Status is retained.
Other peripheral circuit	Initialized or undefined	

Since cold start initializes all the internal peripheral circuits as well as the CPU, it is useful as a power-on reset.
Hot start initializes the CPU and peripheral circuits, but does not reset the bus control unit and the input, output and I/O port status. It is useful as a reset that maintains the external memory, external I/O and the port status.

The #NMI pin that specifies the reset method should be set following the timing chart shown in Figure 3.1.

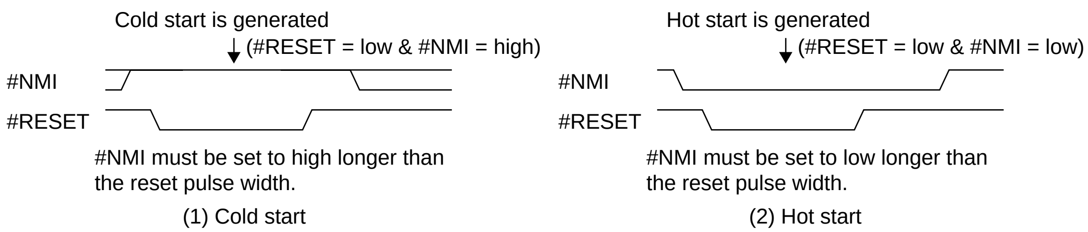


Figure 3.1 Setup of pad_rset_x and #MNI Pins

Power-on Reset

Be sure to reset (cold start) the chip after turning on the power to start operating.
Since the #RESET pin is directly connected to an input gate, a power-on reset circuit should be configured outside the chip.

An initial reset (#RESET = low) turns the high-speed (OSC3) oscillation circuit on. The CPU starts operating with the OSC3 clock at the rising edge of the reset signal. The high-speed (OSC3) oscillation circuit takes time (10 ms max. under the standard condition in 3.3 V) for the oscillation to stabilize, therefore initial reset must be released after an appropriate oscillation-stabilization time has passed in order to start up the CPU without fault. The initial reset pulse width must be exceeded the oscillation-stabilization time.
Figure 3.2 shows a power-on reset timing chart.

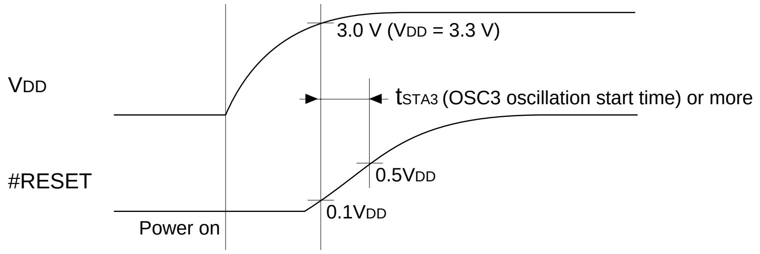


Figure 3.2 Power-on Reset Timing

Maintain the #RESET pin at 0.1•V_{DD} or less (low level) after turning the power on until the supply voltage rises at least to the oscillation start voltage (3.0 V). Furthermore, maintain the #RESET pin at 0.5•V_{DD} or less until the high-speed (OSC3) oscillation circuit stabilizes oscillating.

Note: The OSC3 oscillation start time varies due to the elements used, board pattern and operating environment, therefore allow enough margin for the reset-release time. Refer to "Oscillation Characteristics", in which an example of oscillation start time is provided.

Reset Pulse

A low pulse can be input to the #RESET pin for resetting the chip being operated.
The minimum reset pulse width is provided in "AC Characteristics". Be sure to input a pulse that has a pulse width longer than the minimum value.

To reset the chip when the high-speed (OSC3) oscillation circuit is in off status, the pulse width must be extended until the oscillation stabilizes similarly to the power-on reset. Be aware that a short reset pulse may cause an operation error.

Boot Address

When the core CPU is initially reset, it reads the reset vector (program start address) from the boot address (0x0C00000) and loads the vector to the PC (program counter). Then the CPU starts executing the program from the address when the #RESET pin goes high.

The trap table in which trap vectors for interrupts and other trap factors are written also begins from the boot address by the default setting. (Refer to the "E0C33000 Core CPU Manual" for details of the trap table.)
The trap table base address can also be changed to a 1KB boundary address using the TTBR register (0x48134 to 0x48137).

Notes Related to Initial Reset

Core CPU

Since the all registers except for the PC and PSR are indeterminate at initial reset, they should be initialized by a program. In particular, the SP (stack pointer) must be initialized before accessing the stack area. NMI requests are disabled until any value is written to the SP. The initialization is necessary when the CPU is cold-started.

Internal RAM

The contents of the internal RAM are indeterminate at initial reset. Initialize the area to be used if necessary.

High-speed (OSC3) oscillation circuit

An initial reset activates the high-speed (OSC3) oscillation circuit and the CPU starts operating with the OSC3 clock after the initial reset is released. In order to prevent a malfunction of the CPU due to an unstabilized clock, the #RESET pin must be maintained at low until the OSC3 oscillation stabilizes when performing a power-on reset or resetting while the high-speed (OSC3) oscillation circuit is stopped.

Low-speed (OSC1) oscillation circuit

A power-on reset or an initial reset when the low-speed (OSC1) oscillation circuit is off starts the OSC1 oscillation. The low-speed (OSC1) oscillation circuit takes a longer stabilization time (3 sec max. under the standard condition) than the high-speed (OSC3) oscillation circuit. In order to prevent a malfunction due to an unstabilized clock, do not use the OSC1 clock until the stabilization time has passed.

BCU (Bus Control Unit)

Cold-start initializes the control registers for the BCU (bus control unit). Therefore, it is necessary to set up all the bus conditions.
Hot-start retains the previous bus conditions before an initial reset.

Input/output ports and input/output pins

Cold start initializes the control and data registers for the input and I/O ports.
Hot start retains the contents of the control registers and input/output pin status before an initial reset. However, when the pins are used for the internal peripheral circuits, it is necessary to set up the control registers of the peripheral circuit because they are initialized by an initial reset.

Other internal peripheral circuits

The control and data registers of peripheral circuits other than those listed above are initialized with the predefined values or become indeterminate regardless of the reset method (cold start or hot start). Therefore, it is necessary to set up the peripheral circuit conditions.
Refer to the I/O maps or explanation of each peripheral circuit section for initial settings of the peripheral circuits.

THIS PAGE IS BLANK.

II-4 BCU (Bus Control Unit)

The BCU (Bus Control Unit) provides an interface for external devices and on-chip user logic block. The types and sizes of memory and peripheral I/O devices can be set for each area of the memory map and can be controlled directly by the BCU. This unit also supports a direct interface for DRAM and burst ROM. This chapter describes how to control the external and internal system interface, and how it operates.

Note: The control registers of the external system interface shown in this chapter are mapped to the internal 16-bit I/O area. Therefore, the addresses of these control registers are indicated by half-word (16-bit) addresses unless otherwise specified. Note that the control registers can be accessed in bytes, half-words, or words.

Pin Assignment for External System Interface

I/O Pin List

External I/O pins

Table 4.1 lists the pins used for the external system interface.

Table 4.1 I/O Pin List

Pin name	I/O	Function
A[0]/#BSL	O	Address bus (A0) / Bus strobe (Low-byte)
A[23:1]	O	Address bus (A1–A23)
D[15:0]	I/O	Data bus (D0–D15)
#CE10EX	O	Area 10 external memory chip enable
#CE10IN	O	Area 10 chip enable for internal ROM emulation mode
#CE9/#CE17	O	Area 9/17 chip enable
#CE8/#RAS1/#CE14/#RAS3	O	Area 8/14 chip enable / DRAM Row strobe
#CE7/#RAS0/#CE13/#RAS2	O	Area 7/13 chip enable / DRAM Row strobe
#CE6	O	Area 6 chip enable
#CE5/#CE15	O	Area 5/15 chip enable
#CE4/#CE11	O	Area 4/11 chip enable
#CE3	O	Area 3 chip enable for ROM emulation mode
#RD	O	Read signal
#EMEMRD	O	Read signal for area 3/10 emulation mode
#WRL/#WR/#WE	O	Write (Low-byte) / Write / DRAM write
#WRH/#BSH	O	Write (High-byte) / Bus strobe (High-byte)
#HCAS	O	DRAM column address strobe (High-byte)
#LCAS	O	DRAM column address strobe (Low-byte)
BCLK	O	Bus clock output
#BUSREQ/#CE6/P34	I/O	Bus release request / Area 6 chip enable / I/O port
#BUSACK/P35	O	Bus request acknowledge / I/O port
#WAIT/#CE4&5/P30	I/O	Wait cycle request / Areas 4&5 chip enable / I/O port
#DRD/P20	O	DRAM read signal / I/O port
#DWE/P21	O	DRAM write (Low-byte) / I/O port
#X2SPD	I	CPU - BCLK clock ratio 1: CPU clock = Bus clock, 0: CPU clock = Bus clock x 2
EA10MD[1:0]	I	Area 10 boot mode selection 11: External ROM, 10: Internal ROM, 01: OTP, 00: Internal ROM emulation
EA3MD	I	Area 3 mode selection 1: Internal ROM mode, 0: Emulation mode

User interface signals

Table 4.2 List of User Interface Signals

Signal name	I/O	Function
Internal_addr0	O	• Address bus (a0) when SBUSST(D3/0x4812E) = "0" (default) • Bus strobe (low byte) signal (#BSL) when SBUSST(D3/0x4812E) = "1"
Internal_addr[23:1]	O	Address bus (a1 to a23)
Internal_dout[15:0]	O	Output data bus (dout0 to dout15) This data bus is used when the CPU writes data to the on-chip user logic.
Internal_din[15:0]	I	Input data bus (din0 to din15) This data bus is used when the CPU reads data from the on-chip user logic.
Internal_ce4_x Internal_ce5_x Internal_ce6_x	O	Areas 6–4 chip enable signals These signals go low when the CPU accesses the user logic circuits that are mapped to Areas 6–4.
Internal_rd_x	O	Read signal This signal goes low when the CPU reads data from the user logic.
Internal_wrl_x	O	• Write (low byte) signal (#WRL) when SBUSST(D3/0x4812E) = "0" (default) • Write signal (#WR) when SBUSST(D3/0x4812E) = "1" This signal goes low when the CPU write 8 low-order bit data to the user logic.
Internal_wrh_x	O	• Write (high byte) signal (#WRH) when SBUSST(D3/0x4812E) = "0" (default) • Bus strobe (high byte) signal (#BSH) when SBUSST(D3/0x4812E) = "1" This signal goes low when the CPU write 8 high-order bit data to the user logic.
Internal_osc3_clk	O	High-speed (OSC3) oscillation clock output This can be used as a source clock for the user logic.
Internal_pll_clk	O	PLL output clock This can be used as a source clock for the user logic.
Internal_wait_x	I	Wait cycle request input The user logic can request to insert wait cycles by setting this signal to low.
Internal_irrd_x	O	Instruction fetch indicator signal This signal goes low when the CPU is in an instruction fetch cycle.
Internal_k60-k67	I	Input signals These signals are connected to the input ports K60–K67. The user logic can request HSDMA, IDMA and interrupts using these signals. The user logic can also be used as input ports with these signals.

The internal bus signals are available when an internal access area is set using the BCU register.
The bus conditions can be programmed using the BCU registers similar to the external bus.

Combination of System Bus Control Signals

The bus control signal pins that have two or more functions have their functionality determined when an interface method is selected by a program. The BCU contains an ordinary external system interface (two interface method are supported) and a DRAM interface.

Table 4.3 Interface Selection

Interface type	Interface method	Control bit
External system interface	A0 system (default)	SBUSST(D3/0x4812E) = "0"
	#BSL system	SBUSST(D3/0x4812E) = "1"
DRAM interface	2CAS system (fixed)	None

SBUSST is initialized to "0" at cold start.
When the IC is hot-started, these bits retain their status before the chip was reset.

Table 4.4 shows combinations of control signals classified by each interface method.

Table 4.4 Combinations of Bus Control Signals

External system interface		DRAM interface
A0 system	**#BSL system**	**2CAS system**
A0	#BSL (little endian) / #BSH (big endian) ∗1	–
#WRL	#WR	#WE
#WRH	#BSL (little endian) / #BSH (big endian) ∗1	–
–	–	#HCAS
–	–	#LCAS
#CEx	#CEx	#RASx ∗2

∗1 In the #BSL system, the A0 and #WRH pin functions change according to the endian selected (little endian or big endian).

∗2 When using DRAM, the #CE output pins in areas 7–8 (areas 13–14) function as the #RAS1–2 (#RAS3–4) pins.

Memory Area

Memory Map

Figure 4.1 shows the memory map supported by the BCU.

Area	Address	
Area 9 SRAM type Burst ROM type 8 or 16 bits	0x0BFFFFF 0x0800000	External memory (4MB)
Area 8 SRAM type DRAM type 8 or 16 bits	0x07FFFFF 0x0600000	External memory (2MB)
Area 7 SRAM type DRAM type 8 or 16 bits	0x05FFFFF 0x0400000	External memory (2MB)
Area 6 SRAM type	0x03FFFFF 0x0380000	External I/O (16-bit device)
	0x037FFFF 0x0300000	External I/O (8-bit device)
Area 5 SRAM type 8 or 16 bits	0x02FFFFF 0x0200000	External memory (1MB)
Area 4 SRAM type 8 or 16 bits	0x01FFFFF 0x0100000	External memory (1MB)
Area 3 16 bits Fixed at 1 cycle	0x00FFFFF 0x0080000	(Reserved) For middleware use
Area 2 16 bits Fixed at 3 cycles	0x007FFFF 0x0060000	(Reserved) For CPU core or debug mode
Area 1 8, 16 bits 2 or 4 cycles	0x005FFFF 0x0050000	(Mirror of internal I/O)
	0x004FFFF 0x0040000	Internal I/O
	0x003FFFF 0x0030000	(Mirror of internal I/O)
Area 0 32 bits Fixed at 1 cycle	0x002FFFF 0x0000000	Internal RAM

Area	Address	
Area 18 SRAM type 8 or 16 bits	0xFFFFFFF 0xD000000	
	0xCFFFFFF 0xC000000	External memory (16MB)
Area 17 SRAM type 8 or 16 bits	0xBFFFFFF 0x9000000	
	0x8FFFFFF 0x8000000	External memory (16MB)
Area 16 SRAM type 8 or 16 bits	0x7FFFFFF 0x7000000	
	0x6FFFFFF 0x6000000	External memory (16MB)
Area 15 SRAM type 8 or 16 bits	0x5FFFFFF 0x5000000	
	0x4FFFFFF 0x4000000	External memory (16MB)
Area 14 SRAM type DRAM type 8 or 16 bits	0x3FFFFFF 0x3000000	External memory (16MB)
Area 13 SRAM type DRAM type 8 or 16 bits	0x2FFFFFF 0x2000000	External memory (16MB)
Area 12 SRAM type 8 or 16 bits	0x1FFFFFF 0x1800000	External memory (8MB)
Area 11 SRAM type 8 or 16 bits	0x17FFFFF 0x1000000	External memory (8MB)
Area 10 SRAM type Burst ROM type 8 or 16 bits	0x0FFFFFF 0x0C00000	External memory (4MB)

Figure 4.1 Memory Map

Basically, Areas 0 to 3 are internal memory areas and Areas 4 to 18 are external memory areas.
Area 0 is normally used for a built-in RAM. The built-in memory is mapped from the beginning of the area and the remained space is used as the mirror area of the mapped memory.
Area 1 is reserved for the I/O memory of the on-chip functional blocks. Address 0x0040000 to address 0x004FFFF are used as the control registers and address 0x0050000 to 0x005FFFF are used as the mirror area.
Area 2 is used in debug mode only and it cannot be accessed in user mode (normal program execution status).
Area 3 is reserved for E0C33 middlewares.
Area 4 to 18 can also be configured as internal memory areas using the control register and they can be used for user logic circuits.

External Memory Map and Chip Enable

The BCU has a 24-bit external address bus (A[23:0]) and a 16-bit external data bus (D[15:0]), allowing an address space of up to 16MB to be accessed with one chip enable signal. By default, the address space is divided into 11 areas (areas 0 to 10) for management purposes. Of these, areas 4 to 10 are open to an external system, each provided with an independent chip-enable pin (#CE[10:4]).

The C33 Core Block is limited to 24 available pins for the address bus and 7 pins for the #CE output due to its package structure. However, the #CE[4:10] output pins can be switched to the high-order area chip enable output pins as shown in Table 4.5 using software. CEFUNC[1:0] (D[A:9]) / DRAM timing set-up register (0x48130) is used for this switching.

Table 4.5 Switching of #CE Output

Pin	CEFUNC = "00"	CEFUNC = "01"	CEFUNC = "1x"
#CE4	#CE4	#CE11	#CE11+#CE12
#CE5	#CE5	#CE15	#CE15+#CE16
#CE6	#CE6	#CE6	#CE7+#CE8
#CE7/#RAS0	#CE7/#RAS0	#CE13/#RAS2	#CE13/#RAS2
#CE8/#RAS1	#CE8/#RAS1	#CE14/#RAS3	#CE14/#RAS3
#CE9	#CE9	#CE17	#CE17+#CE18
#CE10EX	#CE10EX	#CE10EX	#CE9+#CE10EX

(Default: CEFUNC = "00")

The high-order areas that are made available for use by writing "01" to CEFUNC can be larger in size than the default low-order areas. For example, when using DRAM in default settings, the available space is 4MB in areas 7 and 8. However, if areas 13 and 14 are used, up to 32MB of DRAM can be used. The same applies to the other areas. Furthermore, when CEFUNC is set to "10" or "11", five chip enable signals are expanded into two area size. Although the C33 Core Block has only 24 address output pins, it features 28-bit internal address processing. Figure 4.2 shows a memory map for an external system.

Area	Address	
Area 10 (#CE10) SRAM type Burst ROM type 8 or 16 bits	0x0FFFFFF 0x0C00000	External memory 6 (4MB)
Area 9 (#CE9) SRAM type Burst ROM type 8 or 16 bits	0x0BFFFFF 0x0800000	External memory 5 (4MB)
Area 8 (#CE8/#RAS1) SRAM type DRAM type 8 or 16 bits	0x07FFFFF 0x0600000	External memory 4 (2MB)
Area 7 (#CE7/#RAS0) SRAM type DRAM type 8 or 16 bits	0x05FFFFF 0x0400000	External memory 3 (2MB)
Area 6 (#CE6) SRAM type	0x03FFFFF 0x0380000	External I/O (16-bit device)
	0x037FFFF 0x0300000	External I/O (8-bit device)
Area 5 (#CE5) SRAM type 8 or 16 bits	0x02FFFFF 0x0200000	External memory 2 (1MB)
Area 4 (#CE4) SRAM type 8 or 16 bits	0x01FFFFF 0x0100000	External memory 1 (1MB)

CEFUNC = "00"

Area	Address	
Area 17 (#CE17) SRAM type 8 or 16 bits	0xBFFFFFF 0x9000000	(Mirror of External memory 6)
	0x8FFFFFF 0x8000000	External memory 6 (16MB)
Area 15 (#CE15) SRAM type 8 or 16 bits	0x5FFFFFF 0x5000000	(Mirror of External memory 5)
	0x4FFFFFF 0x4000000	External memory 5 (16MB)
Area 14 (#CE14/#RAS3) SRAM type DRAM type 8 or 16 bits	0x3FFFFFF 0x3000000	External memory 4 (16MB)
Area 13 (#CE13/#RAS2) SRAM type DRAM type 8 or 16 bits	0x2FFFFFF 0x2000000	External memory 3 (16MB)
Area 11 (#CE11) SRAM type 8 or 16 bits	0x17FFFFF 0x1000000	External memory 2 (8MB)
Area 10 (#CE10) SRAM type Burst ROM type 8 or 16 bits	0x0FFFFFF 0x0C00000	External memory 1 (4MB)
Area 6 (#CE6) SRAM type	0x03FFFFF 0x0380000	External I/O (16-bit device)
	0x037FFFF 0x0300000	External I/O (8-bit device)

CEFUNC = "01"

Area	Address	
Area 17+18 (#CE17+18) SRAM type 8 or 16 bits	0xFFFFFFF 0xD000000	(Mirror of External memory 7')
	0xCFFFFFF 0xC000000	External memory 7' (16MB)
	0xBFFFFFF 0x9000000	(Mirror of External memory 7)
	0x8FFFFFF 0x8000000	External memory 7 (16MB)
Areas 15–16 (#CE15+16) SRAM type 8 or 16 bits	0x7FFFFFF 0x7000000	(Mirror of External memory 6')
	0x6FFFFFF 0x6000000	External memory 6' (16MB)
	0x5FFFFFF 0x5000000	(Mirror of External memory 6)
	0x4FFFFFF 0x4000000	External memory 6 (16MB)
Area 14 (#CE14/#RAS3) SRAM type DRAM type 8 or 16 bits	0x3FFFFFF 0x3000000	External memory 5 (16MB)
Area 13 (#CE13/#RAS2) SRAM type DRAM type 8 or 16 bits	0x2FFFFFF 0x2000000	External memory 4 (16MB)
Areas 11–12 (#CE11+12) SRAM type 8 or 16 bits	0x1FFFFFF 0x1000000	External memory 3 (16MB)
Areas 9–10 (#CE9+10EX) SRAM type Burst ROM type 8 or 16 bits	0x0FFFFFF 0x0800000	External memory 2 (8MB)
Areas 7–8 (#CE7+8) SRAM type 8 or 16 bits	0x07FFFFF 0x0400000	External memory 1 (4MB)

CEFUNC = "10" or "11"

Figure 4.2 External System Memory Map

Furthermore, the #CE4+#CE5 and #CE6 signals can be output from the P30 and P34 terminals, respectively.
This function expands the accessible area when CEFUNC is set to "01, "10" or "11".
To output the #CE4+#CE5 signal from the P30 terminal:

CFP30 (D0)/P3 function select register (0x402DC) = "1"
IOC30 (D0)/P3 I/O control register (0x402DE) = "1"

To output the #CE6 signal from the P34 terminal:

CFP34 (D4)/P3 function select register (0x402DC) = "1"
IOC34 (D4)/P3 I/O control register (0x402DE) = "1"

The P30 and P34 terminals are set for the general I/O ports at initial reset.
The P30 and P34 terminals are shared with the #WAIT input and the #BUSREQ input, respectively. Therefore, when using the #WAIT and #BUSREQ signals, there terminal cannot be used for #CE4+#CE5 and #CE6 outputs.

Using Internal Memory on External Memory Area

The BCU allows using of an internal memory in the external memory areas.
The AxxIO bit in the access control register (0x48132) is used to select either internal access or external access. When "1" is written, the internal device will be accessed and when "0" is written, the external device is accessed (external access by default). The bit names and the corresponding areas are as follows:

A18IO (DF): Areas 17 and 18
A16IO (DE): Areas 15 and 16
A14IO (DD): Areas 13 and 14
A12IO (DC): Areas 11 and 12
A8IO (DA): Areas 7 and 8
A6IO (D9): Area 6
A5IO (D8): Areas 4 and 5

Exclusive Signals for Areas

Areas can be accessed using the exclusive signals (address strobe and read signals) as well as the common control signals.
To use these exclusive signals, they should be configured using G/A read signal control register (0x48138).
The AxxAS bit is used to enable/disable the address strobe signal, and the AxxRD bit is used to enable/disable the read signal. When "1" is written to the bit, the exclusive signal for the corresponding area(s) is enabled and when "0" is written, it is disabled (disabled by default). The bit names and the corresponding areas are as follows:

A18AS (DF), A18RD (D7): Areas 17 and 18
A16AS (DE), A16RD (D6): Areas 15 and 16
A14AS (DD), A14RD (D5): Areas 13 and 14
A12AS (DC), A12RD (D4): Areas 11 and 12
A8AS (DA), A8RD (D2): Areas 7 and 8
A6AS (D9), A6RD (D1): Area 6
A5AS (D8), A5RD (D0): Areas 4 and 5

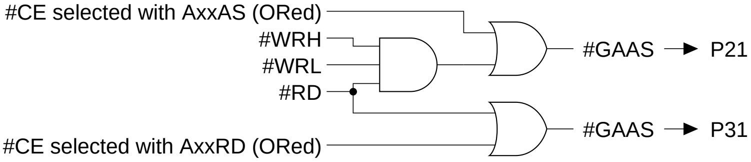


Figure 4.3 #GAAS and #GARD Signals

The address strobe signal and the read signal are output from the P21 pin and P31 pin, respectively. Therefore, when using these signals, the pin(s) must be configured for exclusive signal output using the port function select register and port function extension register.

To output the exclusive address strobe signal #GAAS:

CFEX2 (D2)/Port function extension register (0x402DF) = "1"

To output the exclusive address strobe signal #GARD:

CFEX3 (D3)/Port function extension register (0x402DF) = "1"

These signals are common used to all the above areas, so when two or more areas are selected to output the exclusive signal, OR condition is applied.

Area 10

Area 10 is an external memory area that includes the boot address (0xC00000). This area supports four boot mode and a high-speed internal ROM can also be mapped.

Area 10 boot mode

The boot mode can be configured using the external pins EA10MD[1:0].

Table 4.6 Area 10 Boot Mode Selection

EA10MD[1:0] pins	Area 10 boot mode
00	Internal ROM emulation mode
01	OTP and internal ROM emulation mode
10	Internal ROM boot mode
11	External ROM boot mode

Internal ROM boot mode

The CPU boots by the internal ROM mapped to area 10. The internal ROM size should be selected from among eight types (min. 16KB, max. 2MB) using the A10IR[2:0] (D[E:C])/Areas 10–9 set-up register (0x48126).
This ROM begines with address 0xC00000 and can be read in one cysle the same as that of area 3. For the remained area within area 10, the external memory will be accessed if it is available.

Internal ROM emulation mode

The CPU boots by the external memory that emulates an internal ROM. This mode accesses the ROM emulation area set by the A10IR[2:0] (D[E:C])/Areas 10–9 set-up register (0x48126) using the same condition as internal ROM boot mode. The emulation memory is accessed using the #CE10IN chip enable signal.

OTP and internal ROM emulation mode

In this mode, channel 0 of IDMA starts up for tranfering the program codes in the Flash memory to the high-speed SRAM immediately after an initial reset is released. Then the system boots by the SRAM. After that, this mode functions the same as internal ROM emulation mode.
Since the Flash memory and SRAM are mapped to the same address range, the Flash memory is accessed using the #CE10EX chip enable signal and the SRAM is accessed using the #CE10IN chip enable signal.

External ROM boot mode

The CPU boots by the external ROM (ROM, Flash, SRAM, etc.). This mode uses the bus condition set by the BCU registers for area 10.

Setting the internal ROM size

When a boot mode other than external ROM boot mode is used, the internal ROM or emulation memory size should be set using A10IR[2:0] (D[E:C)/Areas 10–9 set-up register (0x48126).

Table 4.7 Area 10 Internal ROM Size

A10IR2	A10IR1	A10IR0	ROM size
0	0	0	16KB
0	0	1	32KB
0	1	0	64KB
0	1	1	128KB
1	0	0	256KB
1	0	1	512KB
1	1	0	1MB
1	1	1	2MB (default)

Area 10 memory map

Figure 4.4 shows the memory map of area 10.

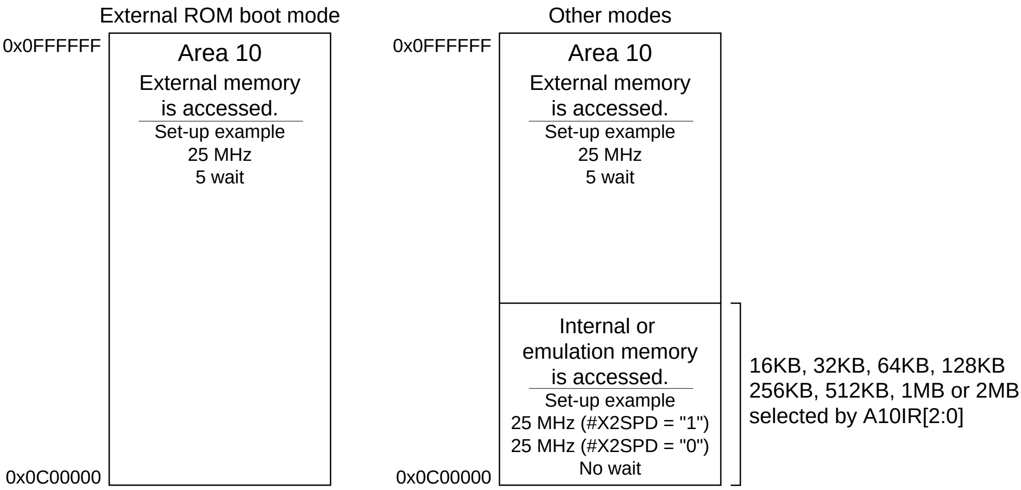


Figure 4.4 Area 10 Memory Map

Area 3

Area 3 is an internal memory area and reserved for E0C33 middlewares.
This area can be configured with an internal ROM or an external emulation memory. The EA3MD pin is used for this selection.

Table 4.8 Area 3 Mode Selection

EA3MD pin	Area 3 mode
0	Emulation mode
1	Internal ROM mode

System Configuration in Emulation Mode

Figure 4.5 shows a development environment example using the emulation mode.

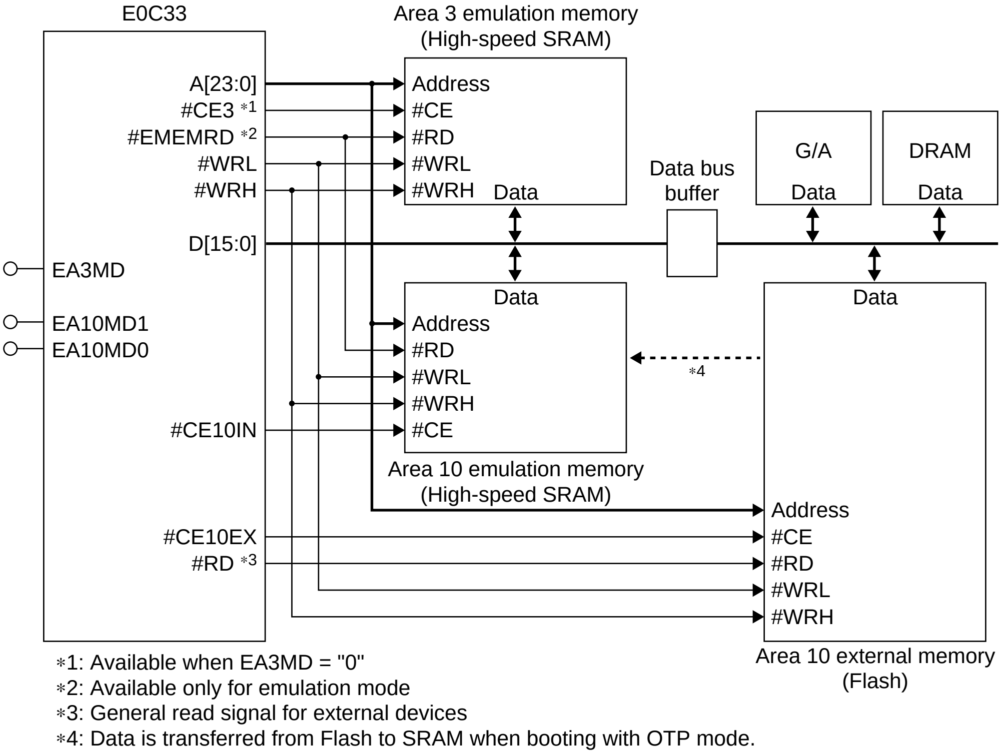


Figure 4.5 Development Environment Example

The emulation memory is accessed without wait state as shown in Figure 4.6.

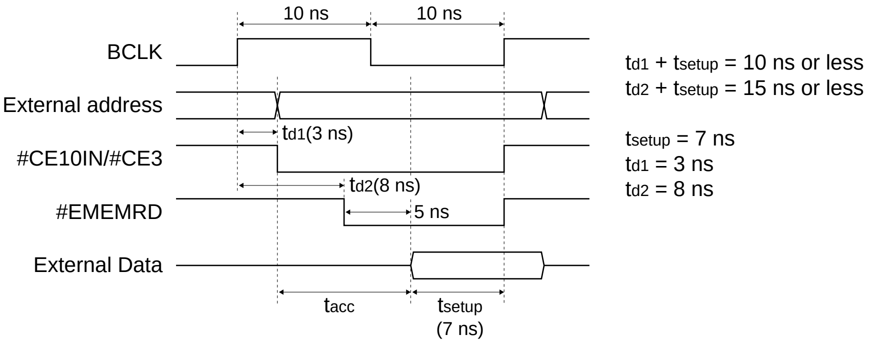


Figure 4.6 Read Timing in Emulation Mode

In the emulation mode, the exclusive read signal #EMEMRD is used.

Setting External Bus Conditions

The type, size, and wait conditions of a device connected to the external bus can be individually set for each area using the control register (0x48120 to 0x48130). The following explains the available setup conditions individually for each area. For details on how to set the DRAM interface conditions, refer to "DRAM Direct Interface".

The control register used to set bus conditions is initialized at cold start. Therefore, please set up these registers again using software according to the external device configuration and specifications.
When the IC is hot-started, the setup contents and pins retain their previous status before a reset.

Setting Device Type and Size

Table 4.9 shows the types of devices that can be connected directly to each area.

Table 4.9 Device Type

Area	SRAM type	DRAM type	Burst ROM type	Control bit
18–15	○	X	X	None
14	○	○	X	A14DRA(D8)/Areas 14–13 set-up register(0x48122)
13	○	○	X	A13DRA(D7)/Areas 14–13 set-up register(0x48122)
12,11	○	X	X	None
10	○	X	○	A10DRA(D8)/Areas 10–9 set-up register(0x48126)
9	○	X	○	A9DRA(D7)/Areas 10–9 set-up register(0x48126)
8	○	○	X	A8DRA(D8)/Areas 8–7 set-up register(0x48128)
7	○	○	X	A7DRA(D7)/Areas 8–7 set-up register(0x48128)
6–4	○	X	X	None

○: Can be connected X: Cannot be connected

When connecting burst ROM or DRAM, write "1" to each corresponding control bit. These control bits are reset to "0" (SRAM type) at cold start.

The device size can be set to 8 or 16 bits once every two areas except for area 6. Area 6 alone has its first half (0x300000–0x37FFFF) fixed to an 8-bit device and the second half (0x380000–0x3FFFFF) fixed to a 16-bit device.

Table 4.10 Device Size Control Bits

Area	Control bit
18, 17	A18SZ(DE)/Areas 18–15 set-up register(0x48120)
16, 15	A16SZ(D6)/Areas 18–15 set-up register(0x48120)
14, 13	A14SZ(D6)/Areas 14–13 set-up register(0x48122)
12, 11	A12SZ(D6)/Areas 12–11 set-up register(0x48124)
10, 9	A10SZ(D6)/Areas 10–9 set-up register(0x48126)
8, 7	A8SZ(D6)/Areas 8–7 set-up register(0x48128)
5, 4	A5SZ(D6)/Areas 6–4 set-up register(0x4812A)

At cold start, each area by default is set to 16 bits.
When using an 8-bit device, write "1" to the control bit.

Note: The BCU supports 16-bit burst ROM. Therefore, when connecting burst ROM to area 10 or area 9, do not set the device size to 8 bits (A10SZ = "1").

For differences in bus operation due to the device size and access data size, refer to "Bus Operation of External Memory".

Setting SRAM Timing Conditions

The areas set for the SRAM allow wait cycles and output disable delay time to be set.

Number of wait cycles: 0 to 7 (incremented in units of one cycle)
Output disable delay time: 0.5, 1.5, 2.5, 3.5 cycles

This selection can be made once every two areas except for area 6.

Table 4.11 Timing Condition Setting Bits (for SRAM type)

Area	Number of wait cycles	Output disable delay time	Control register
18, 17	A18WT[2:0](D[A:8])	A18DF[1:0](D[D:C])	Areas 18–15 set-up register(0x48120)
16, 15	A16WT[2:0](D[2:0])	A16DF[1:0](D5:4])	Areas 18–15 set-up register(0x48120)
14, 13	A14WT[2:0](D[2:0])	A14DF[1:0](D5:4])	Areas 14–13 set-up register(0x48122)
12, 11	A12WT[2:0](D[2:0])	A12DF[1:0](D5:4])	Areas 12–11 set-up register(0x48124)
10, 9	A10WT[2:0](D[2:0])	A10DF[1:0](D5:4])	Areas 10–9 set-up register(0x48126)
8, 7	A8WT[2:0](D[2:0])	A8DF[1:0](D[5:4])	Areas 8–7 set-up register(0x48128)
6	A6WT[2:0](D[A:8])	A6DF[1:0](D[D:C])	Areas 6–4 set-up register(0x4812A)
5, 4	A5WT[2:0](D[2:0])	A5DF[1:0](D[5:4])	Areas 6–4 set-up register(0x4812A)

At cold start, the number of wait cycles is set to 7 and the output disable delay time is set to 3.5 cycles. Reset up these parameters as necessary using software according to specifications of the connected device.
At hot start, these parameters retain their previous settings before a reset.

Wait cycles

When the number of wait cycles is set for an area using the control bit, the BCU extends the bus cycle for a duration equivalent to the wait cycles set when it accesses the area. Set the desired wait cycles according to the bus clock frequency and the external device's access time. Separately from the wait cycles set here, a wait request from an external device can also be accepted using the #WAIT pin. Since the settings of wait cycles using software are made once every two areas, use this external wait request function if you want the wait cycles to be controlled individually in each area or if you need 7 or more wait cycles. The #WAIT pin is shared with the P30 I/O port. For an external wait request to be accepted, write "1" to CFP30 (D0) / P3 function select register (0x402DC [Byte]) and write "1" (default = "0") to SWAIT (D0) / Bus control register (0x4812E) to enable the #WAIT pin.
For timing charts for bus cycles and when wait cycles are inserted, refer to "Bus Cycles in External System Interface".

If the number of wait cycles is set to 0 and no external wait is requested, the basic read cycle (read in byte or half-word) for the SRAM external device consists of one cycle. If wait cycles are set, because these cycles are added, the bus read cycle consists of [number of wait cycles + 1] (providing that there is no external wait).
On the other hand, the basic write cycle consists of at least two cycles. This does not change regardless of whether zero or one wait cycle is set. If the number of wait cycles set is 2 or more, the bus cycle is actually extended. In this case, the bus write cycle consists of [number of wait cycles + 1], as in the case of read cycles (providing that there is no external wait).

Output disable delay time

In cases when a device having a long output disable time is connected, if a read cycle for that device is followed by the next access, contention for the data bus may occur. (Due to the fact the read device's data bus is not placed in the high-impedance state.) The output disable delay time is provided to prevent such data bus contention. This is accomplished by inserting a specified number of cycles between a read cycle and the next bus operation.
Check the specifications of the device to be connected before setting the output disable delay time.
By default, the output disable delay time is inserted only in the following cases:

- when a read cycle from the external device that has had an output disable delay time set is followed by a write cycle performed by the CPU; and
- when a read cycle from the external device that has had an output disable delay time set is followed by a read cycle for a different area (including the internal device).

Conversely, no output disable delay time is inserted in the following conditions:

- immediately after a write cycle, and
- during a successive read from the same external device.

Setting Timing Conditions of Burst ROM

Wait cycles

If burst ROM is selected for area 10 or 9, the wait cycles to be inserted in the burst read cycle can be selected in a range from 0 to 3 cycles. A10BW[1:0] (D[A:9]) / Areas 10–9 set-up register (0x48126) is used for this selection. This selection is applied simultaneously to areas 10 and 9, so wait cycles can not be chosen individually for each area. The wait cycles set at cold start is 0.

Even for a burst read, the SRAM settings of wait cycles in the first bus operation are valid. (Refer to A10WT[2:0] in the foregoing section.)
The wait cycles set by A10BW[1:0] are inserted into the burst cycles after the first bus operation.

In addition, when burst ROM is selected, no wait cycles can be inserted into the read cycle via the #WAIT pin. For writing to an area that has had burst ROM selected, an SRAM write cycle is executed. In this case, both the SRAM settings of wait cycles and those input via the #WAIT pin are valid.

Burst mode

The burst mode can be selected between an eight-consecutive-burst and a four-consecutive-burst mode. RBST8 (DD) / Bus control register (0x4812E) is used for this selection. The eight-consecutive-burst mode is selected by writing "1" to RBST8 and the four-consecutive-burst mode is selected by setting the bit to "0". At cold start, the four-consecutive-burst mode is set by default.

Bus Operation

Data Arrangement in Memory

The E0C33 Family of devices handle data in bytes (8 bits), half-words (16 bits), and words (32 bits). When accessing data in memory, it is necessary to specify a boundary address that conforms to the data size involved. Specification of an invalid address causes an address error exception. For instructions (e.g., stack manipulation or branch instructions) that rewrite the SP (stack pointer) or PC (program counter), the specified addresses are forcibly modified to appropriate boundary addresses. Therefore, no address error exception occurs in this type of instruction. For details about the address error exception, refer to the "E0C33000 Core CPU Manual".

Table 4.12 shows the data arrangement in memory, classified by data type.

Table 4.12 Data Arrangement in Memory

Data type	Arranged location
Byte data	Byte boundary address (all addresses)
Half-word data	Half-word boundary address (A[0]="0")
Word data	Word boundary address (A[1:0]="00")

The half-word and word data in memory area accessed in little-endian format by default. It can be changed to big-endian format using AxxEC (D[7:0])/Access control register (0x48132). When "1" is written to AxxEC, the corresponding area is accessed in big-endian method. The bit names and the corresponding areas are as follows:

A18EC (D7): Areas 17 and 18
A16EC (D6): Areas 15 and 16
A14EC (D5): Areas 13 and 14
A12EC (D4): Areas 11 and 12
A10EC (D3): Areas 9 and 10 ... Fixed at "0" (little-endian) for booting.
A8EC (D2): Areas 7 and 8
A6EC (D1): Area 6
A5EC (D0): Areas 4 and 5

To increase memory efficiency, try to locate the same type of data at continuous locations on exact boundary addresses in order to minimize invalid areas.

Bus Operation of External Memory

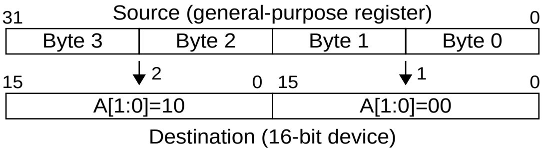
The external data bus is 16-bits wide. For this reason, more than one bus operation occurs depending on the device size and the data size of the instruction executed, as shown in Table 4.13.

Table 4.13 Number of Bus Operation Cycles

Data size to be accessed	Devise size	Number of bus operation cycles	Remarks
32 bits	16 bits	2	
16 bits	16 bits	1	
8 bits	16 bits	1	In little-endian method, the low-order byte is accessed when the LSB of the address (A[0]) is "0" or the #BSL signal is L. The high-order byte is accessed when the LSB of the address (A[0]) is "1" or the #BSH signal is H. In big-endian method, the high-order byte is accessed when the LSB of the address (A[0]) is "0" or the #BSL signal is L. The low-order byte is accessed when the LSB of the address (A[0]) is "1" or the #BSH signal is H.
32 bits	8 bits	4	In little-endian method, the 8-bit device must be connected to the low-order 8 bits of the data bus. In big-endian method, the 8-bit device must be connected to the high-order 8 bits of the data bus.
16 bits	8 bits	2	In little-endian method, the 8-bit device must be connected to the low-order 8 bits of the data bus. In big-endian method, the 8-bit device must be connected to the high-order 8 bits of the data bus.
8 bits	8 bits	1	In little-endian method, the 8-bit device must be connected to the low-order 8 bits of the data bus. In big-endian method, the 8-bit device must be connected to the high-order 8 bits of the data bus.

The following diagram shows sample bus operations where the A0 system is used.

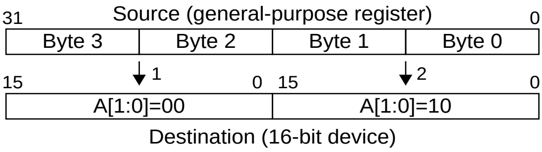
Little-endian



Bus operation

No.	A1	A0	#WRH	#WRL	Data bus (15)	Data bus (0)
1	0	0	0	0	Byte 1	Byte 0
2	1	0	0	0	Byte 3	Byte 2

Big-endian

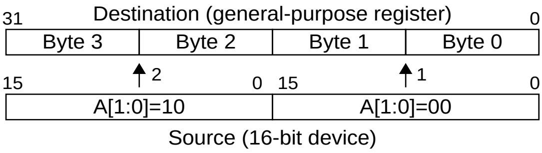


Bus operation

No.	A1	A0	#WRH	#WRL	Data bus (15)	Data bus (0)
1	0	0	0	0	Byte 3	Byte 2
2	1	0	0	0	Byte 1	Byte 0

Figure 4.7 Word Data Writing to a 16-bit Device

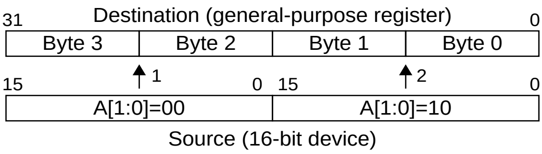
Little-endian



Bus operation

No.	A1	A0	#WRH	#WRL	Data bus (15)	Data bus (0)
1	0	0	1	1	Byte 1	Byte 0
2	1	0	1	1	Byte 3	Byte 2

Big-endian

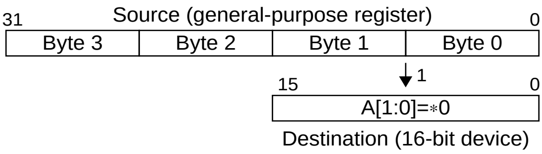


Bus operation

No.	A1	A0	#WRH	#WRL	Data bus (15)	Data bus (0)
1	0	0	1	1	Byte 3	Byte 2
2	1	0	1	1	Byte 1	Byte 0

Figure 4.8 Word Data Reading from a 16-bit Device

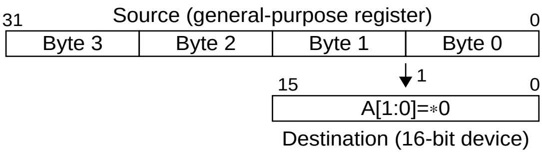
Little-endian



Bus operation

No.	A1	A0	#WRH	#WRL	Data bus (15)	Data bus (0)
1	*	0	0	0	Byte 1	Byte 0

Big-endian

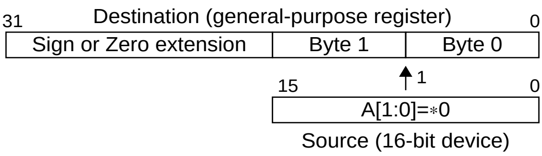


Bus operation

No.	A1	A0	#WRH	#WRL	Data bus (15)	Data bus (0)
1	*	0	0	0	Byte 1	Byte 0

Figure 4.9 Half-word Data Writing to a 16-bit Device

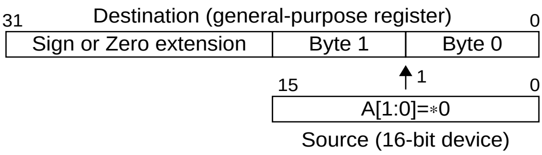
Little-endian



Bus operation

No.	A1	A0	#WRH	#WRL	Data bus (15)	Data bus (0)
1	*	0	1	1	Byte 1	Byte 0

Big-endian

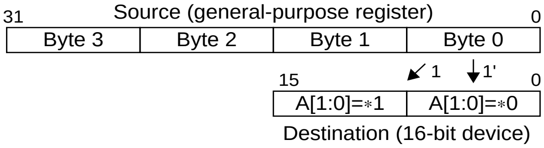


Bus operation

No.	A1	A0	#WRH	#WRL	Data bus (15)	Data bus (0)
1	*	0	1	1	Byte 1	Byte 0

Figure 4.10 Half-word Data Reading from a 16-bit Device

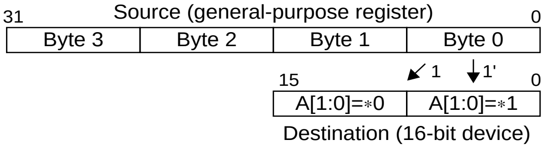
Little-endian



Bus operation

No.	A1	A0	#WRH	#WRL	15 Data bus	0
1	*	1	0	1	Byte 0	Data retained
1'	*	0	1	0	Data retained	Byte 0

Big-endian

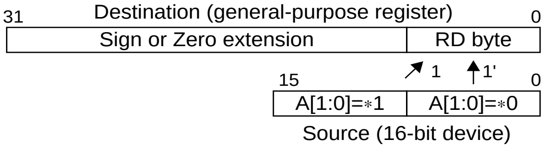


Bus operation

No.	A1	A0	#WRH	#WRL	15 Data bus	0
1	*	0	0	1	Byte 0	Data retained
1'	*	1	1	0	Data retained	Byte 0

Figure 4.11 Byte Data Writing to a 16-bit Device

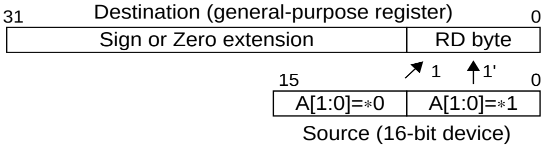
Little-endian



Bus operation

No.	A1	A0	#WRH	#WRL	15 Data bus	0
1	*	1	1	1	RD byte	Ignored
1'	*	0	1	1	Ignored	RD byte

Big-endian

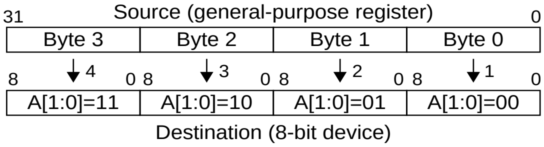


Bus operation

No.	A1	A0	#WRH	#WRL	15 Data bus	0
1	*	0	1	1	RD byte	Ignored
1'	*	1	1	1	Ignored	RD byte

Figure 4.12 Byte Data Reading from a 16-bit Device

Little-endian

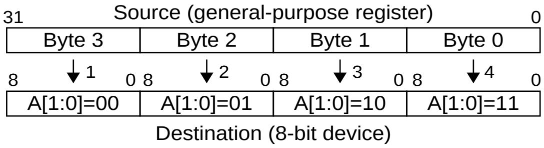


Bus operation

No.	A1	A0	#WRH	#WRL	15 Data bus	0
1	0	0	X	0	Data retained	Byte 0
2	0	1	X	0	Data retained	Byte 1
3	1	0	X	0	Data retained	Byte 2
4	1	1	X	0	Data retained	Byte 3

(X: Not connected/Unused)

Big-endian

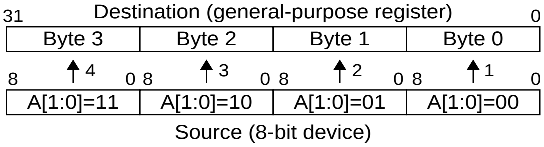


Bus operation

No.	A1	A0	#WRH	#WRL	15 Data bus	0
1	0	0	0	1	Byte 3	Data retained
2	0	1	0	1	Byte 2	Data retained
3	1	0	0	1	Byte 1	Data retained
4	1	1	0	1	Byte 0	Data retained

Figure 4.13 Word Data Writing to an 8-bit Device

Little-endian

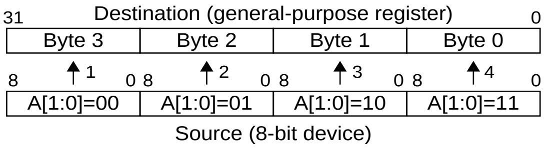


Bus operation

No.	A1	A0	#WRH	#WRL	15 Data bus	0
1	0	0	X	1	Ignored	Byte 0
2	0	1	X	1	Ignored	Byte 1
3	1	0	X	1	Ignored	Byte 2
4	1	1	X	1	Ignored	Byte 3

(X: Not connected/Unused)

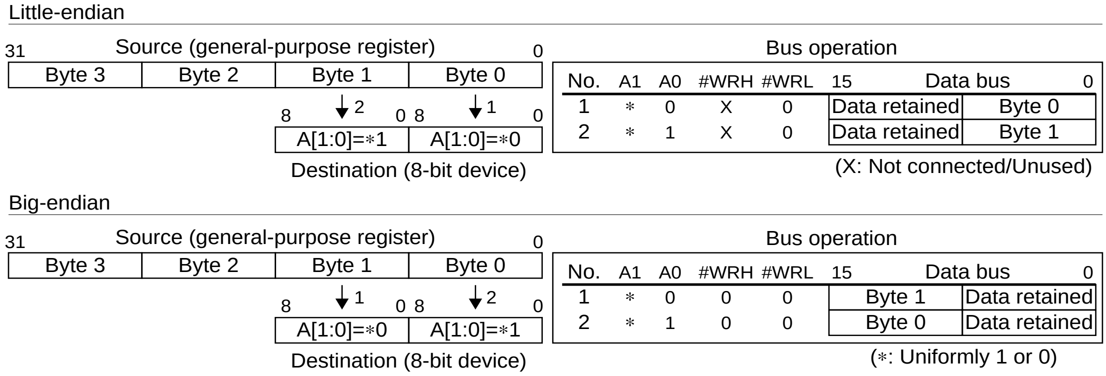
Big-endian



Bus operation

No.	A1	A0	#WRH	#WRL	15 Data bus	0
1	0	0	1	1	Byte 3	Ignored
2	0	1	1	1	Byte 2	Ignored
3	1	0	1	1	Byte 1	Ignored
4	1	1	1	1	Byte 0	Ignored

Figure 4.14 Word Data Reading from an 8-bit Device



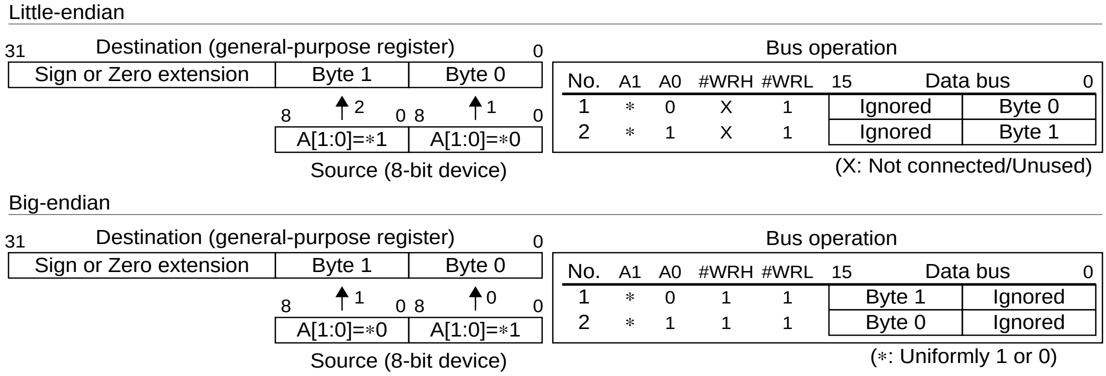
Little-endian:

No.	A1	A0	#WRH	#WRL	Data bus 15–8	Data bus 7–0
1	∗	0	X	0	Data retained	Byte 0
2	∗	1	X	0	Data retained	Byte 1

Big-endian:

No.	A1	A0	#WRH	#WRL	Data bus 15–8	Data bus 7–0
1	∗	0	0	0	Byte 1	Data retained
2	∗	1	0	0	Byte 0	Data retained

Figure 4.15 Half-word Data Writing to an 8-bit Device



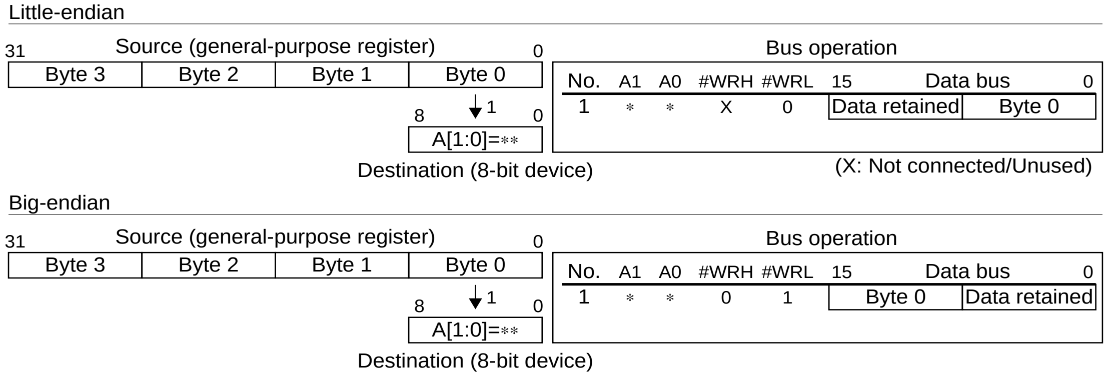
Little-endian:

No.	A1	A0	#WRH	#WRL	Data bus 15–8	Data bus 7–0
1	∗	0	X	1	Ignored	Byte 0
2	∗	1	X	1	Ignored	Byte 1

Big-endian:

No.	A1	A0	#WRH	#WRL	Data bus 15–8	Data bus 7–0
1	∗	0	1	1	Byte 1	Ignored
2	∗	1	1	1	Byte 0	Ignored

Figure 4.16 Half-word Data Reading from an 8-bit Device



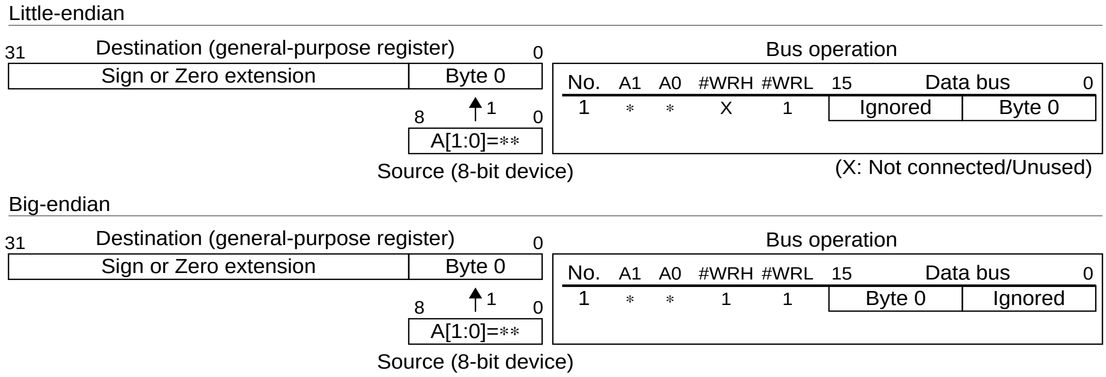
Little-endian:

No.	A1	A0	#WRH	#WRL	Data bus 15–8	Data bus 7–0
1	∗	∗	X	0	Data retained	Byte 0

Big-endian:

No.	A1	A0	#WRH	#WRL	Data bus 15–8	Data bus 7–0
1	∗	∗	0	1	Byte 0	Data retained

Figure 4.17 Byte Data Writing to an 8-bit Device



Little-endian:

No.	A1	A0	#WRH	#WRL	Data bus 15–8	Data bus 7–0
1	∗	∗	X	1	Ignored	Byte 0

Big-endian:

No.	A1	A0	#WRH	#WRL	Data bus 15–8	Data bus 7–0
1	∗	∗	1	1	Byte 0	Ignored

Figure 4.18 Byte Data Reading from an 8-bit Device

Bus Clock

The bus clock is generated by the BCU using the CPU system clock output from the clock generator.
Figure 4.19 shows the clock system.

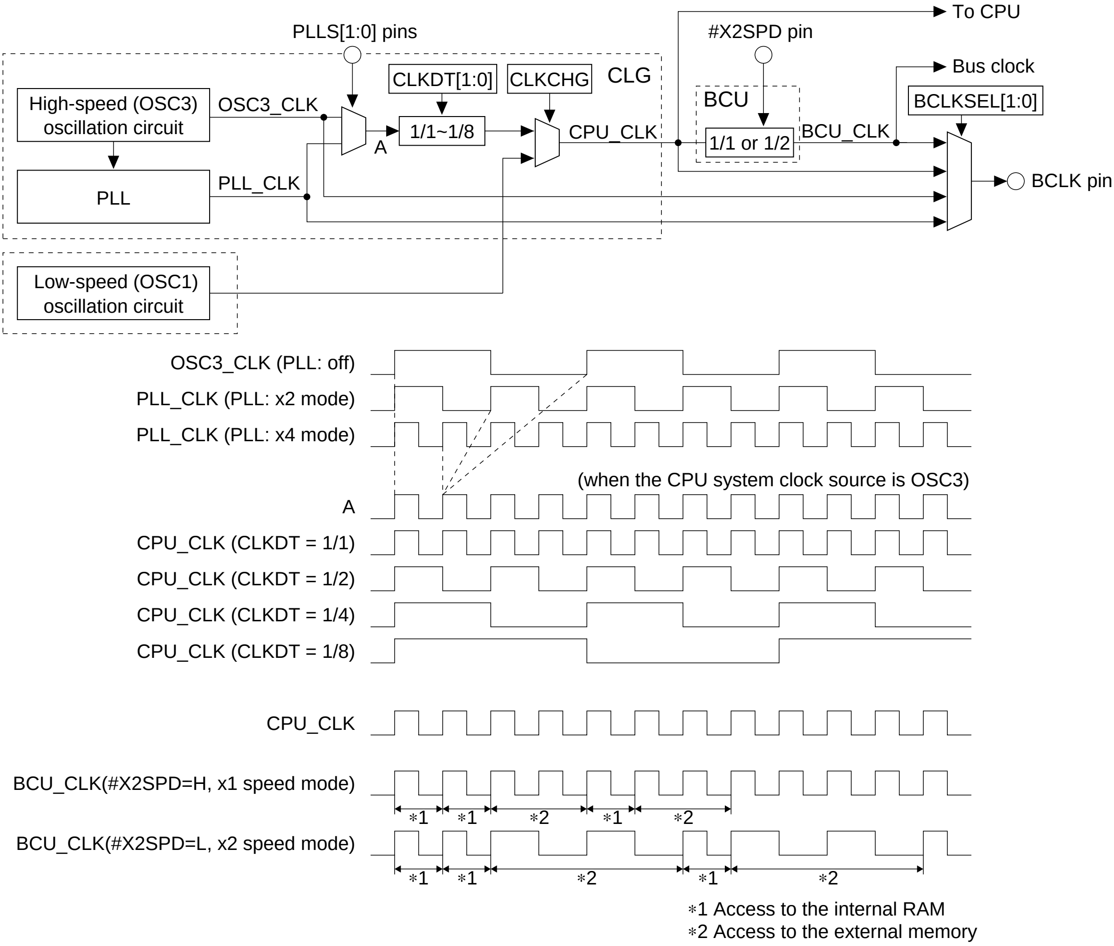


Figure 4.19 Clock System

Since the bus clock is generated from the CPU system clock (CPU_CLK), the following settings affect the bus clock:

1. Selection of an oscillation circuit (OSC3 or OSC1)
2. PLL configuration (OSC3_CLK x 1, x2 or x4)
3. CPU clock division ratio for power saving

Items 2 and 3 apply when the high-speed (OSC3) oscillation circuit is selected as the CPU clock source.
For details about the settings of the system clock, refer to "CLG (Clock Generator)".

Bus clock operation during standby is as follows:

Basic HALT mode: the BCU and bus clock continue operating. DRAM can be refreshed.
HALT2 mode: the BCU and bus clock are stopped.
SLEEP mode: the BCU and bus clock are stopped.

Bus Speed Mode

The CPU - bus clock ratio can be set using the #X2SPD pin as follows:
When #X2SPD = High, x1 speed mode (CPU - bus clock ratio is 1 : 1) is set. The bus clock and the CPU system clock will be the same.
When #X2SPD = Low, x2 speed mode (CPU - bus clock ratio is 2 : 1) is set. In x2 speed mode, the bus clock will be dynamically varied according to the memory to be accessed.

- When an external memory area is accessed, the bus clock frequency becomes half of the CPU system clock.
- When the internal RAM/ROM area is accessed, the bus clock frequency becomes equal to the CPU system clock.

In x1 speed mode, area 1 (internal I/O area) is accessed in 4 cycles of the CPU system clock, while in x2 speed mode, the number of access cycles can be selected using A1X1MD (D3) / BCLK select register (0x4813A).
When A1X1MD = "1", area 1 is accessed in 2 cycles of the CPU system clock.
When A1X1MD = "0", area 1 is accessed in 4 cycles of the CPU system clock. (default)

Bus Clock Output

The bus clock is also output from the BCLK pin to an external device. The BCLK output clock can be selected from among four types using BCLKSEL[1:0] (D[1:0]) / BCLK select register (0x4813A).

Table 4.14 Selection of BCLK Output Clock

BCLKSEL1	BCLKSEL0	Output clock
1	1	PLL_CLK (PLL output clock)
1	0	OSC3_CLK (OSC3 oscillation clock)
0	1	BCU_CLK (BCU operating clock)
0	0	CPU_CLK (CPU operating clock)

Bus Cycles in External System Interface

The following shows a sample SRAM connection the basic bus cycles.

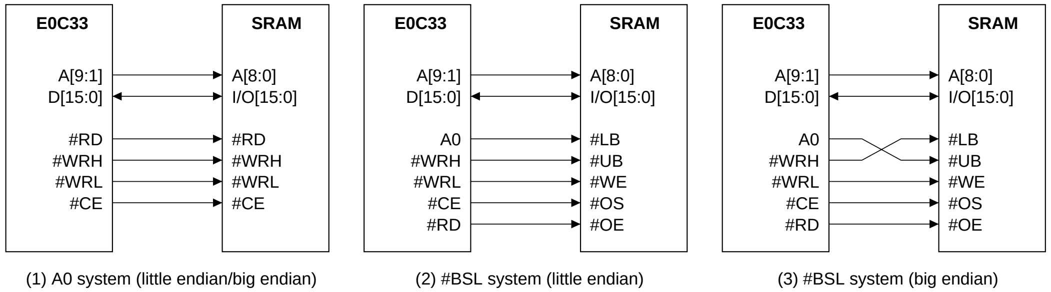


Figure 4.20 Sample DRAM Connection

SRAM Read Cycles

Basic read cycle with no wait mode

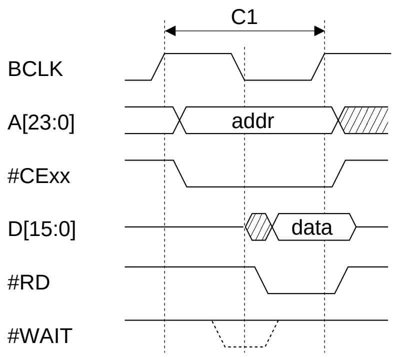


Figure 4.21 Basic Read Cycle with No Wait

Read cycle with wait mode

Example: When the BCU has no internal wait mode and 2 wait cycles via #WAIT pin are inserted

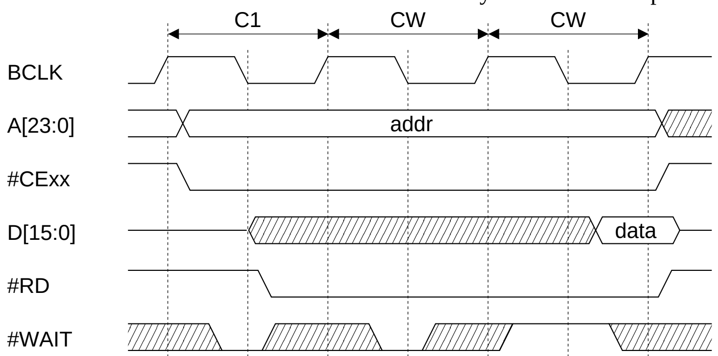


Figure 4.22 Read Cycle with Wait

The #WAIT signal is sampled at the falling edge of the transition of BCLK (bus clock) and when it is sampled on an inactive (high level), the read cycle is terminated.

Note: Insertion of wait cycles via the #WAIT pin is possible only when the device for bus conditions is set for SRAM, and SWAIT (D0) / Bus control register (0x4812E) is enabled for waiting.

The above example shows a read cycle when a wait mode is inserted via the #WAIT signal. A wait mode consisting of 0 to 7 cycles can also be inserted using the wait control bits. The settings of these bits can also be used in combination with the #WAIT signal. In this case as well, the #WAIT signal is sampled at the falling edge of the transition of BCLK. However, even when the #WAIT signal is inactive before the wait cycles set by the wait control bits are terminated, the read cycle is not terminated at that time.

Precaution

#CE and address hold times at the rising edge of the #RD signal

In read cycles of this BCU, negating the #RD signal, negating the chip enable (#CExx) signal and changing the address (A[23:0]) occur simultaneously at the same clock edge. No hold time is inserted to the chip enable and address signals. The same applies even when an output disable delay time is inserted.
Therefore when connecting a peripheral circuit, which changes its internal state by reading, to the bus, take a measure to insert a delay to the address and chip enable signals.

Output disable cycle

When an output disable cycle (set with output disable delay time parameter) is inserted, the chip enable (#CExx) signal temporarily goes high. This makes an interval between the next read cycle.
Note, however, that no output disable cycle is inserted when reading is continuously performed to the area that is accessed with the same chip enable signal.

SRAM Write Cycles

Basic write cycle with no wait mode

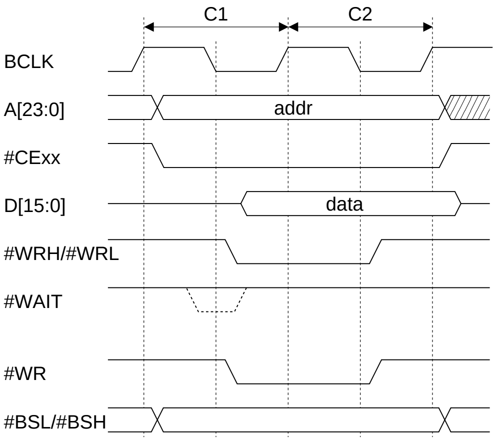


Figure 4.23 Half-word Write Cycle with No Wait

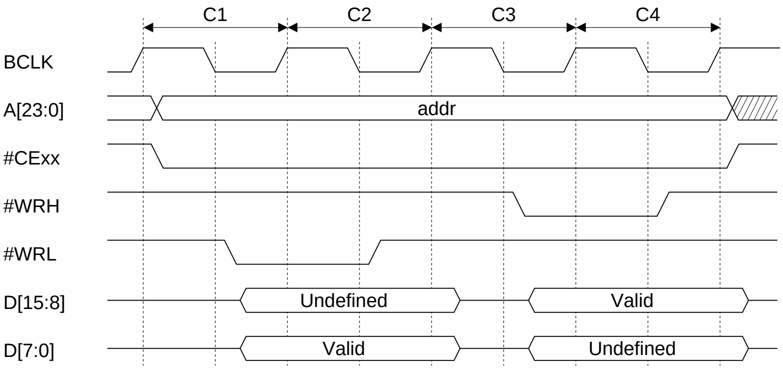


Figure 4.24 Byte Write Cycle with No Wait (A0 system, little endian)

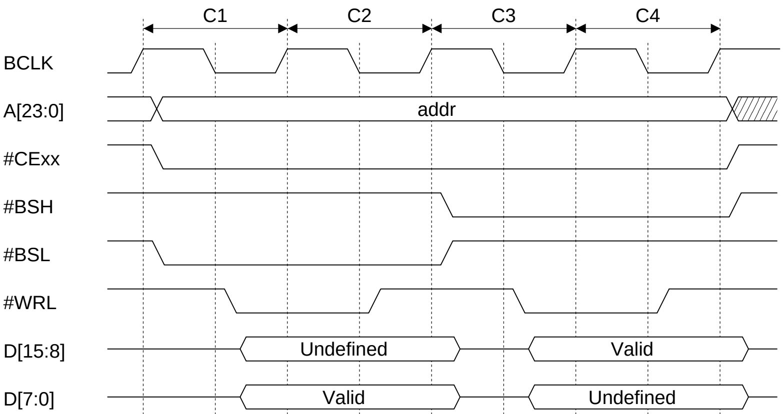


Figure 4.25 Byte Write Cycle with No Wait (#BSL system, little endian)

Write cycle with wait mode

Example: When the BCU has no internal wait mode, and 1 wait cycle is inserted via the #WAIT pin

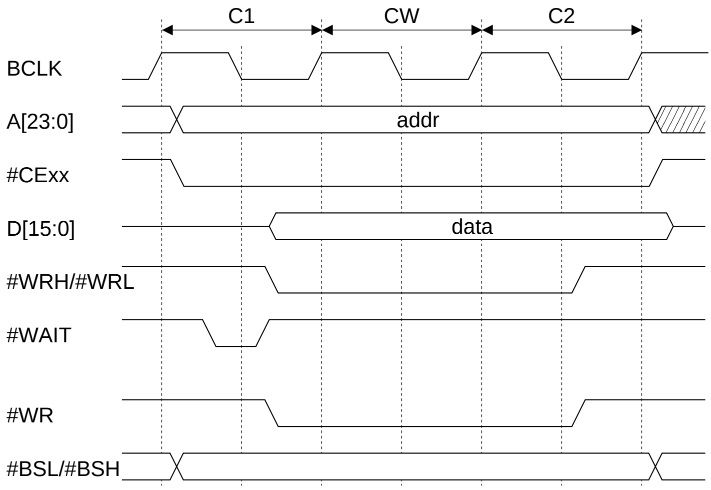


Figure 4.26 Half-word Write Cycle with Wait

The #WAIT signal is sampled at the falling edge of the transition of BCLK (bus clock), and the write cycle is terminated in the cycle immediately following the cycle in which the #WAIT signal was sampled in an inactive (high level).

Note: Insertion of wait cycles via the #WAIT pin is possible only when the device for bus conditions is set to SRAM and SWAIT (D0) / Bus control register (0x4812E) is enabled for waiting.

The above example shows a write cycle when a wait mode is inserted via the #WAIT signal. A wait mode consisting of 2 to 7 cycles can also be inserted using the wait control bits. The settings of these bits also can be used in combination with the #WAIT signal. In this case as well, the #WAIT signal is sampled at the falling edge of the transition of BCLK. However, even when the #WAIT signal is inactive before the wait cycles set by the wait control bits are terminated, the write cycle is not terminated at that time.

Note: The basic write cycle consists of at least two cycles. This does not change regardless of whether zero or one wait cycle is set by the wait control bits. If the number of wait cycles set is 2 or more, the bus cycle is actually extended. In this case, the bus write cycle consists of [number of wait cycles + 1], as in the case of read cycles (providing that there is no external wait).

Burst ROM Read Cycles

Burst read cycle

Example: When 4-consecutive-burst and 2-wait cycles are set during the first access

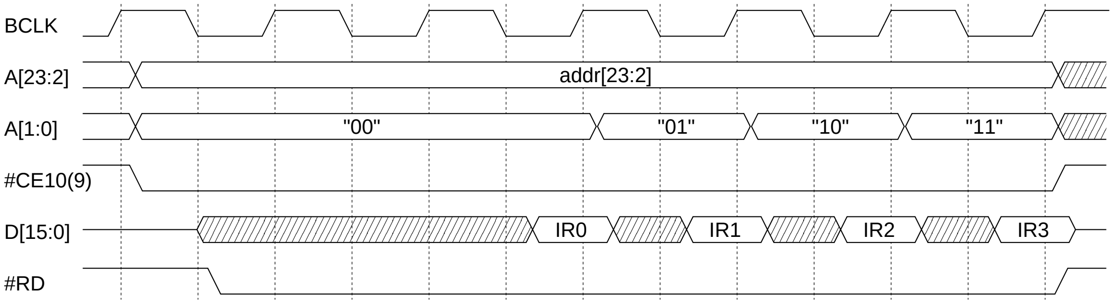


Figure 4.27 Burst Read Cycle

A burst read cycle occurs when area 10 or 9 is set for burst ROM and one of those areas is accessed for the following reasons:

1) Instruction fetch

The burst read cycle is executed as long as a instruction fetch from contiguous addresses continues until
A[2:1] = "11" (for 4-consecutive bursts); or
A[3:1] = "111" (for 8-consecutive bursts)

2) Word (32-bit) data readout

Note: A 16-bit output is supported for the burst ROM. Set the device size to 16 bits.

Wait cycles during burst read

In the first bus operation, 0 to 7 wait cycles can be inserted using the wait control bits A10WT[2:0] (D[2:0]) / Areas 10–9 set-up register (0x48126) in the same way as for ordinary SRAM. For the wait cycles to be inserted in the burst cycle that follows, use a dedicated wait control bits, A10BW[1:0], which is only used for reading bursts. The wait cycles can be set in the range from 0 to 3 using these bits.
Note that no wait cycle via the #WAIT pin can be inserted into the burst-read cycle.

Write cycle to burst ROM area

If area 10 or 9 is set for burst ROM, a SRAM write cycle is executed when a write to that area is attempted. In this case, wait cycles via the #WAIT pin can be inserted.

DRAM Direct Interface

Outline of DRAM Interface

The BCU incorporates a DRAM direct interface that allows DRAM to be connected directly to areas 8 and 7 or areas 14 and 13. This interface supports the 2CAS method, so that column addresses can be set at between 8 and 11 bits. In addition, this interface supports a fast-page or an EDO-page mode (EDO DRAM directly connectable to areas) as well as random cycles. The refresh method (CAS-before-RAS refresh or self-refresh) and timing conditions (e.g., number of RAS/CAS cycles and number of precharge cycles) can be programmed using a control bit.

When selecting areas 8 and 7 or areas 14 and 13 to be used for DRAM, it depends on chip-enable settings using CEFUNC (D9) / DRAM timing set-up register (0x48130).

CEFUNC = "00": DRAM can be connected to areas 8 and 7 (default)
#CE8 and #CE7 function as #RAS0 and #RAS1, respectively.

CEFUNC ≠ "00": DRAM can be connected to areas 14 and 13.
#CE14 and #CE13 function as #RAS2 and #RAS3, respectively.

Figure 4.28 shows a sample DRAM connection. Table 4.15 and Table 4.16 show examples of connectable DRAMs and typical configurations.

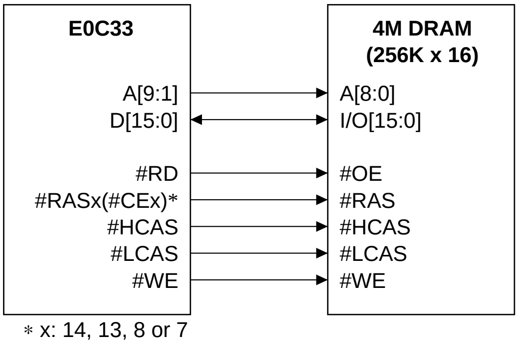


Figure 4.28 Sample DRAM Connection

Table 4.15 Connectable DRAM Example

DRAM	Number of devices	Number of Row bits	Number of Column bits	Memory size
1M (64K x 16)	1	8	8	128K bytes
4M (256K x 16)	1	9	9	512K bytes
16M (1M x 16)	1	12	8	2M bytes

Table 4.16 DRAM Configuration Example (areas 7 and 8 only)

	Area 7	Area 8	Total memory size
1	I/O	DRAM (1M)	1M bits (128K bytes)
2	I/O	DRAM (4M)	4M bits (512K bytes)
3	I/O	DRAM (16M)	16M bits (2M bytes)
4	DRAM (1M)	DRAM (1M)	2M bits (256K bytes)
5	DRAM (4M)	DRAM (4M)	8M bits (1M bytes)
6	DRAM (16M)	DRAM (16M)	32M bits (4M bytes)

DRAM Setting Conditions

The DRAM interface allows the following conditions to be selected. Although DRAM can be used in areas 8 and 7 or areas 14 and 13, these condition are applied to all four areas and cannot be set individually for each area.

Table 4.17 DRAM Interface Parameters

Parameter	Selectable condition	Initial setting	Control bits
Page mode	EDO page mode or Fast page mode	Fast page mode	REDO(DC)/Bus control register(0x4812E)
RAS mode	Successive RAS mode or Normal mode	Normal mode	CRAS(D8)/DRAM timing set-up register(0x48130)
Column address size	8, 9, 10 or 11 bits	8 bits	RCA[1:0](D[B:A])/Bus control register(0x4812E)
Refresh enable	Enabled or Disabled	Disabled	RPC2(D9)/Bus control register(0x4812E)
Refresh method	Self-refresh or CBR refresh	CBR refresh	RPC1(D8)/Bus control register(0x4812E)
Refresh RPC delay	2.0 or 1.0	1.0	RPC0(D7)/Bus control register(0x4812E)
Refresh RAS pulse width	2, 3, 4 or 5 cycles	2 cycles	RRA[1:0](D[6:5])/Bus control register(0x4812E)
Number of RAS precharge cycles	1, 2, 3 or 4 cycles	1 cycle	RPRC[1:0](D[7:6])/DRAM timing set-up register(0x48130)
CAS cycle control	1, 2, 3 or 4 cycles	1 cycle	CASC[1:0](D[4:3])/DRAM timing set-up register(0x48130)
RAS cycle control	1, 2, 3 or 4 cycles	1 cycle	RASC[1:0](D[1:0])/DRAM timing set-up register(0x48130)

Page mode

The DRAM interface allows EDO DRAM to be connected directly. Therefore, the EDO-page mode is supported along with the fast-page mode.
Use REDO to choose the desired page mode that suits the DRAM to be used.
REDO = "1": EDO page mode
REDO = "0": Fast page mode (default)

Successive RAS mode

For applications that require high-speed DRAM access, the DRAM interface supports a successive RAS mode. In this mode, even when successive accesses to the DRAM are not requested by the CPU or DMA, the #RAS signal is kept low and operation is continued without inserting any precharge cycle. Therefore, when accessing the same page (row address) of the DRAM that has been accessed previously, the page mode remains active, allowing read/write to be performed at high speeds.
However, to maintain the rated AC characteristics, one idle cycle is inserted when access in the page mode is begun and when finished.
CRAS is used to set the successive RAS mode.
CRAS = "1": Successive RAS mode
CRAS = "0": Normal mode (default)

The successive RAS mode is suspended by one of the following causes:
• a refresh cycle has occurred;
• bus control is requested by an external bus master;
• the requested device and page are not compatible with DRAM memory; and
• the slp or halt instruction is executed.
If the successive RAS mode is suspended, a precharge cycle is inserted before the next bus cycle begins.

Note: When using the successive RAS mode, always be sure to use #DRD for the read signal and #DWE for the low-byte write signal.

Column address size

When accessing DRAM, addresses are divided into a row address and a column address as they are output. Choose the size of this column address using RCA, as shown below.

Table 4.18 Column Address Size

RCA1	RCA0	Column address size
1	1	11
1	0	10
0	1	9
0	0	8

The initial default size is 8 bits. Choose the desired size according to the address input pins of the DRAM to be used.

The row addresses output synchronously with falling edges of the #RAS signal are derived from the CPU's internal 28-bit addresses by logically shifting them to the right by an amount equal to the column address size. The MSB contains a 1. The column addresses are output to the address bus along with the falling edges of the #CAS signal. These addresses are derived directly from the CPU's internal 28-bit addresses.

Figure 4.29 shows the contents of the row addresses thus output.

28-bit CPU internal address

27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

(1) Row address when column address is set to 8 bits

T	T	T	T	T	T	T	T	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8

(2) Row address when column address is set to 9 bits

T	T	T	T	T	T	T	T	T	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9

(3) Row address when column address is set to 10 bits

T	T	T	T	T	T	T	T	T	T	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10

(4) Row address when column address is set to 11 bits

T	T	T	T	T	T	T	T	T	T	T	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11

T = "1", 0–27: Bit number of CPU internal address

Figure 4.29 Example of Row/Column Address Mapping

Refresh enable

Use RPC2 to enable or disable the internal refresh function.

RPC2 = "1": Enabled

RPC2 = "0": Disabled (default)

After choosing the desired refresh method using RPC1, write "1" to RPC2.

Refresh method

The DRAM interface supports both a CAS-before-RAS refresh cycle and a self-refresh cycle. Choose the desired method using RPC1.

RPC1 = "1": Self-refresh

RPC1 = "0": CAS-before-RAS refresh

The generation interval of the CAS-before-RAS refresh is determined by the underflow signal of an 8-bit programmable timer 0. Consequently, before the CAS-before-RAS refresh can be executed, the 8-bit programmable timer 0 must be set to obtain the necessary underflow timing. When this method is selected and RPC2 is enabled, the refresh cycle is generated each time the 8-bit programmable timer 0 underflows.

The self-refresh is started by writing "1" to RPC2 while RPC1 = "1" and is terminated by clearing RPC1 or RPC2 to "0".

If RPC1 is switched over when RPC2 = "1" (refresh enabled), an undesirable self-refresh cycle is generated. So be sure to clear RPC2 to "0" (refresh disabled) before selecting the refresh method.

Refresh RPC delay

Use RPC0 to set the RPC delay value of a refresh cycle (a delay time from the immediately preceding precharge to the fall of #CAS).

RPC0 = "1": 2 cycles

RPC0 = "0": 1 cycle

Refresh RAS pulse width

Use RRA to set the #RAS pulse width of a CAS-before-RAS refresh cycle.

Table 4.19 Refresh RAS Pulse Width

RRA1	RRA0	Pulse width
1	1	5 cycles
1	0	4 cycles
0	1	3 cycles
0	0	2 cycles

The initial default value is 2 cycles.

Number of RAS precharge cycles

Use RPRC to choose the number of RAS precharge cycles.

Table 4.20 Number of RAS Precharge Cycles

RPRC1	RPRC0	Number of cycles
1	1	4 cycles
1	0	3 cycles
0	1	2 cycles
0	0	1 cycle

The initial default value is 1 cycle.

CAS cycle control

Use CASC to choose the number of CAS cycles when accessing DRAM.

Table 4.21 Number of CAS Cycles

CASC1	CASC0	Number of cycles
1	1	4 cycles
1	0	3 cycles
0	1	2 cycles
0	0	1 cycle

The initial default value is 1 cycle.

RAS cycle control

Use RASC to choose the number of RAS cycles when accessing DRAM.

Table 4.22 Number of RAS Cycles

RASC1	RASC0	Number of cycles
1	1	4 cycles
1	0	3 cycles
0	1	2 cycles
0	0	1 cycle

The initial default value is 1 cycle.

DRAM Read/Write Cycles

The following shows the basic bus cycles of DRAM.
The DRAM interface does not accept wait cycles inserted via the #WAIT pin.

DRAM random read cycle

Example: RAS: 1 cycle; CAS: 2 cycles; Precharge: 1 cycle

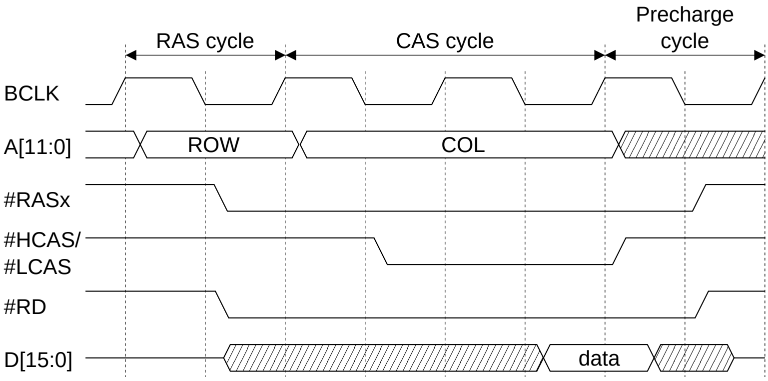


Figure 4.30 DRAM Random Read Cycle

DRAM read cycle (fast page mode)

Example: RAS: 1 cycle; CAS: 2 cycles; Precharge: 1 cycle

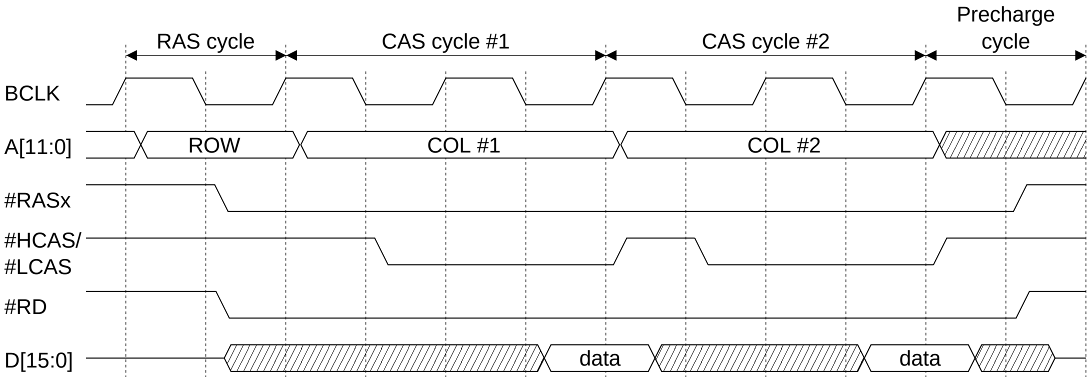


Figure 4.31 DRAM Read Cycle (fast page mode)

DRAM read cycle (EDO page mode)

Example: RAS: 1 cycle; CAS: 2 cycles; Precharge: 1 cycle

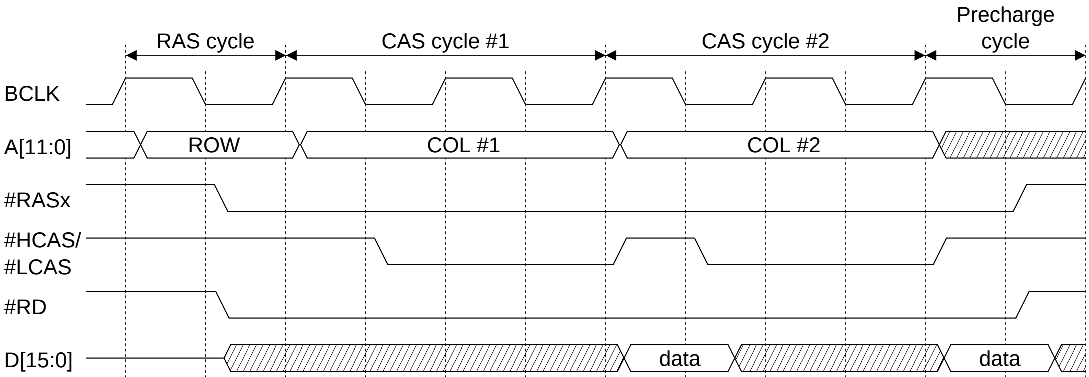


Figure 4.32 DRAM Read Cycle (EDO page mode)

The read timing in EDO page-mode lags 0.5 cycles behind that in fast page mode.

DRAM random write cycle

Example: RAS: 1 cycle; CAS: 2 cycles; Precharge: 1 cycle

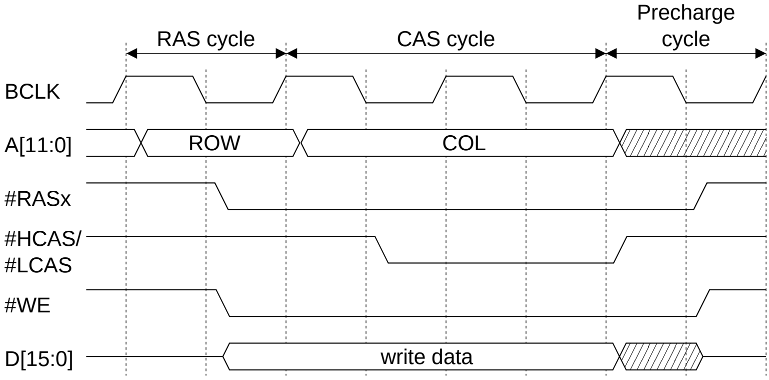


Figure 4.33 2CAS Type DRAM Random Write Cycle

DRAM write cycle (fast page or EDO page mode)

Example: RAS: 1 cycle; CAS: 2 cycles; Precharge: 1 cycle; word-write sample

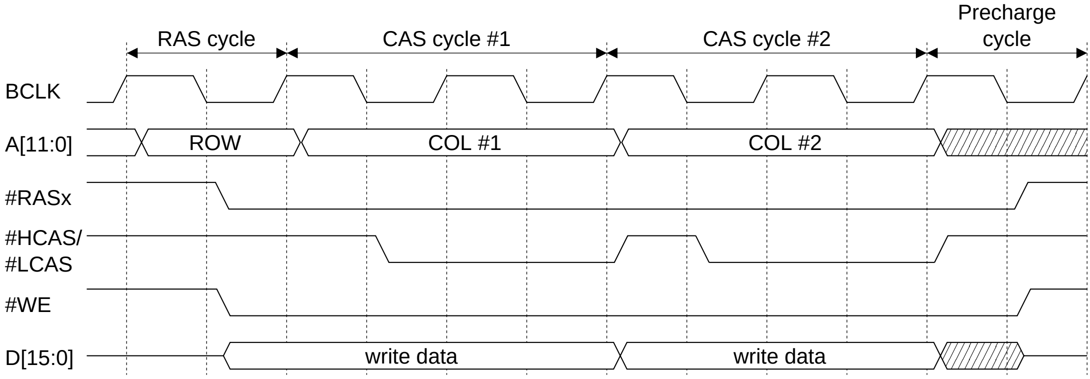


Figure 4.34 DRAM Word-Write Cycle (fast page or EDO page mode)

Example: RAS: 1 cycle; CAS: 2 cycles; Precharge: 1 cycle; byte-write sample (little endian)

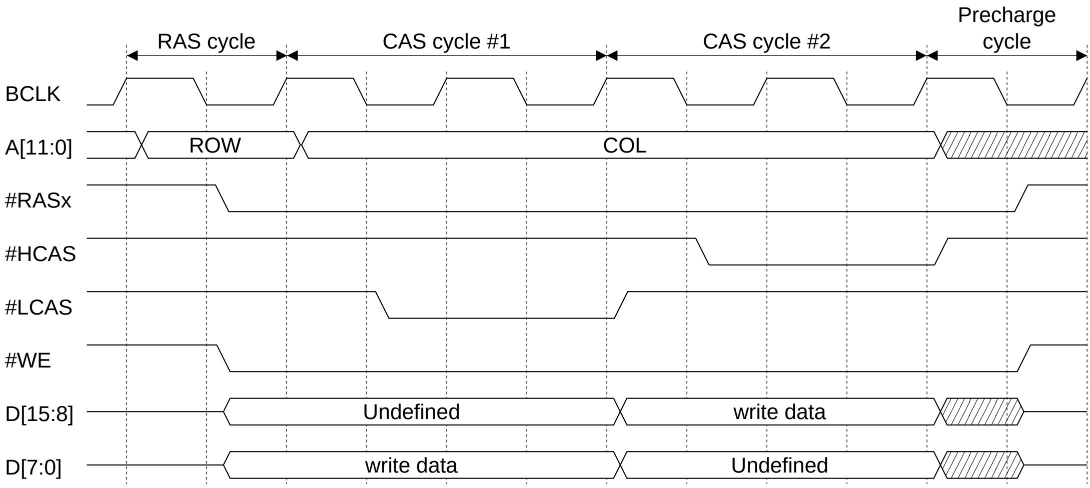


Figure 4.35 DRAM Byte-Write Cycle (fast page or EDO page mode)

Operation in successive RAS mode

Example: RAS: 2 cycles; CAS: 1 cycle; Precharge: 2 cycles

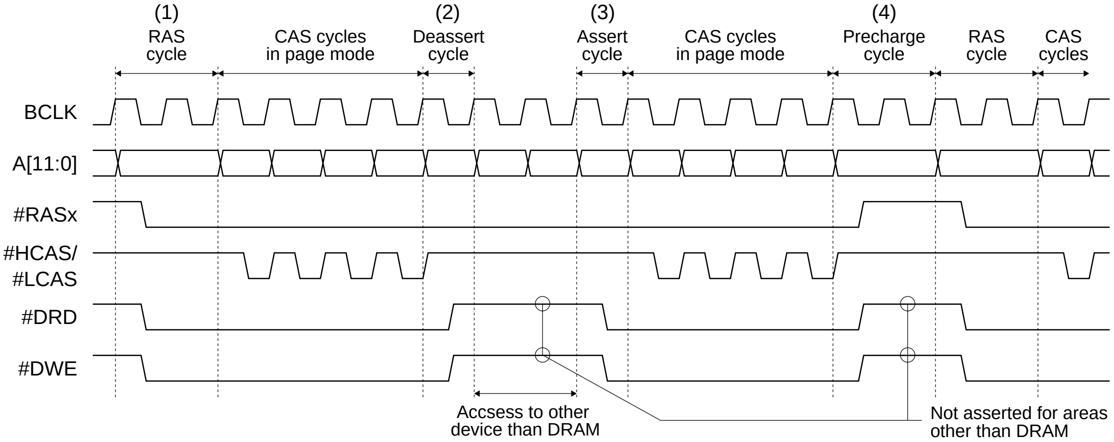


Figure 4.36 Operation in Successive RAS Mode

(1) When accessing the DRAM area, an ordinary RAS cycle is executed first.

(2) If access to the same DRAM is suspended during a page mode, #RASx remains asserted while some other device is accessed. In this case, a cycle to temporarily deassert #DRD/#DWE is inserted before accessing the other device.

(3) If access to the same page in the same DRAM area as in (1) is requested after (2), #DRD/#DWE is asserted back again to restart the page mode.

(4) A precharge cycle is executed when one of the following conditions that cause the page mode to suspend is encountered:
- access to different DRAM is requested;
- access to a different page in the same DRAM area is requested;
- access to some other device than DRAM is requested;
- CAS-before-RAS refresh is requested; and
- relinquishing of bus control is requested by an external bus master.

Note: When using the successive RAS mode, always be sure to use #DRD for the read signal and #DWE for the low-byte write signal.

DRAM Refresh Cycles

The DRAM interface supports a CAS-before-RAS refresh cycle and a self-refresh cycle.

CAS-before-RAS refresh cycle

Before performing a CAS-before-RAS refresh, set RPC2 to "1" while RPC1 = "0" in order to enable the DRAM refresh function. Once this is done, the BCU executes a CAS-before-RAS refresh by using the underflow signal that is output by the 8-bit programmable timer 0 as a trigger. Therefore, refresh generation timing can be programmed using the internal prescaler and 8-bit programmable timer 0.

For details on how to control the prescaler and 8-bit programmable timer 0, refer to "Prescaler and Operating Clock for Peripheral Circuits", and "8-Bit Programmable Timers".

Example: RPC delay: 1 cycle; Refresh RAS pulse width: 2 cycles; Precharge: 1 cycle

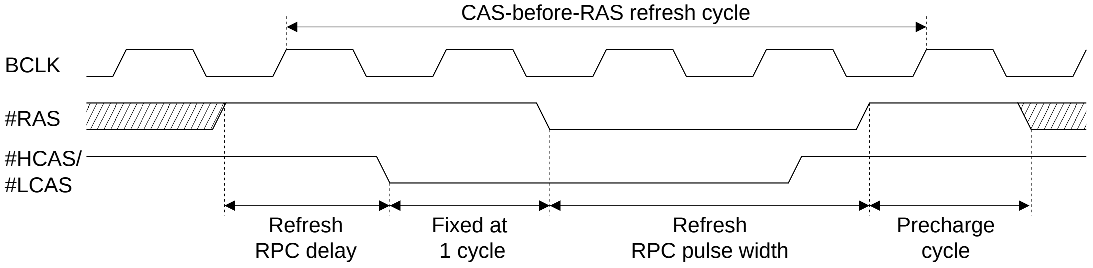


Figure 4.37 CAS-Before-RAS Refresh

When the refresh cycle is terminated, the #HCAS/#LCAS signal boot timing is 0.5 cycles before that of #RAS. Consequently, the pulse width of #HCAS/#LCAS is determined by the refresh RAS pulse width that was set using RRA. The number of precharge cycles after the refresh cycle is defined by the value that was set using RPRC, the same value that is used for both random cycles and page mode accesses.

Self-refresh

To support DRAM chips equipped with a self-refresh function, the BCU has a function to generate a self-refresh cycle.

To start a self-refresh cycle, set RPC2 to "1" after setting RPC1 to "1". To deactivate a self-refresh cycle, write "0" to RPC1 or RPC2.

Example: RPC delay: 1 cycle

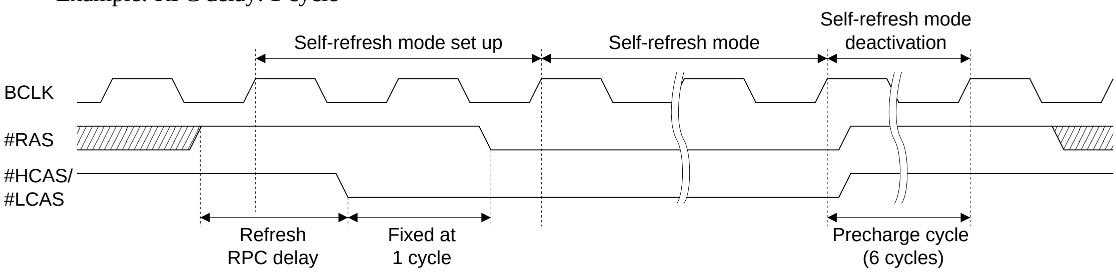


Figure 4.38 Self-Refresh

For a self-refresh function as well, the RPC delay is determined by setting RPC0 in the same way as for a CAS-before-RAS refresh.

The refresh RAS pulse width is determined by the timing at which the refresh is deactivated in software and is unaffected by settings of RRA.

#RAS and #HCAS/#LCAS are booted up simultaneously upon completion of a self-refresh and the precharge duration that follows is fixed at 6 cycles.

Normally, DRAM specifications require that the contents of all row addresses be refreshed within a certain time before and after a self-refresh. To meet this requirement, make sure a CAS-before-RAS refresh is executed by a program. In this case, set the 8-bit programmable timer 0 so that the contents of all row addresses are refreshed within a predetermined time.

Note: If read from or write to the DRAM under a self-refresh is attempted, the BCU keeps #RAS and #HCAS/#LCAS low as it executes a read/write cycle. Other bus signals than #RAS and #HCAS/#LCAS (e.g., address, data, and control signals) change their state according to the specified conditions. Since said attempt initiates an invalid access to the DRAM, do not read from or write to the DRAM during a self-refresh.

Releasing External Bus

The external bus is normally controlled by the CPU, but the BCU is designed to release control of the bus ownership to an external device. This function is enabled by writing "1" to SEMAS (D2) / Bus control register (0x4812E) (disabled by default). The #BUSREQ (P34) and #BUSACK (P35) pins are used for control of the bus ownership. To direct the P34 and P35 pins for input/output of the #BUSREQ and #BUSACK signals, write "1" to CFP34 (D4) and CFP35 (D5) / P3 function select register (0x402DC [Byte]).

Sequence in which control of the bus is released

This sequence is described below.

1. The external bus master device requesting control of the bus ownership lowers the #BUSREQ pin.
2. The CPU keeps monitoring the status of the #BUSREQ pin, so that when this pin is lower, the CPU terminates the bus cycle being executed and places the signals listed below in high-impedance state one cycle later:
   A[23:0], D[15:0], #RD, #WRL, #WRH, #HCAS, #LCAS, #CExx
   Then the CPU lowers the #BUSACK pin to inform the external device that control of the bus ownership has been released.
3. One cycle later, the external bus starts its own bus cycle. The external bus master must hold the #BUSREQ pin low until the bus cycle is completed.
4. After completing the necessary bus cycles, the external bus master places the bus in high-impedance state and releases the #BUSREQ pin back high.
5. After confirming that the #BUSREQ pin is raised again, the CPU raises the #BUSACK pin one cycle later and resumes the processing that has been suspended.

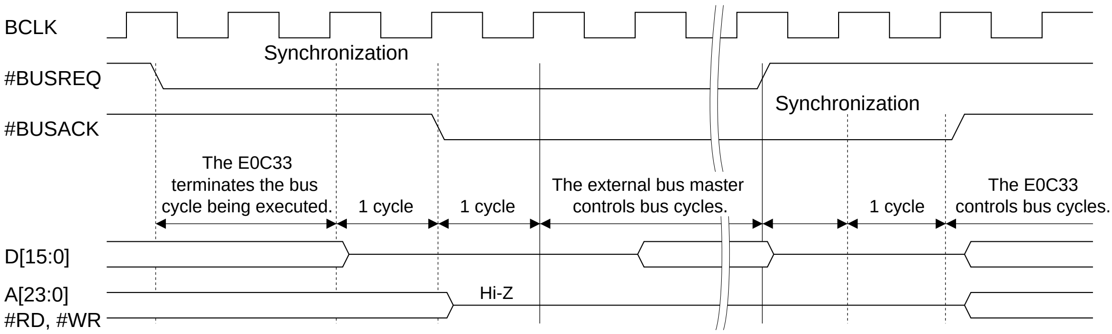


Figure 4.39 External Bus Release Timing

If control of the bus ownership is requested during a DMA transfer by the internal DMA controller, the DMA transfer under way is suspended at a break in data to accept the request for bus ownership control. The DMA transfer that has been kept pending is restarted when the CPU gains control of the bus ownership.

DRAM refresh when bus ownership control is released

In systems where DRAM is connected directly, a refresh request could arise while control of the bus ownership is released from the CPU. In such a case, take one of the corrective measures described below.

• Monitoring the output signal of the 8-bit programmable timer 0

The underflow signal (DRAM refresh request) of the 8-bit programmable timer 0 can be output from the P10 I/O port pin.
If a refresh request arises while the external bus master is monitoring this output, release #BUSREQ back high to drop the request for bus ownership control.
Start a DRAM refresh cycle when control of the bus ownership is returned to the CPU.

To direct the P10 pin in order to output the underflow signal of the 8-bit programmable timer 0, write "1" to CFP10 (D0) / P1 function select register (0x402D4 [Byte]) and IOC10 (D0) / P1 I/O control register (0x402D6 [Byte]). Also, to output the underflow signal to an external device, write "1" to PTOUT0 (D2) / 8-bit Timer 0 control register (0x40160 [Byte]). For details about output control, refer to "8-Bit Programmable Timers".

• Monitoring the #BUSGET signal

The #BUSGET signal can be output from the P31 I/O port pin.
The #BUSGET signal is derived from logical sum of the following signals:
1. DRAM refresh request signal (output from the 8-bit programmable timer 0)
2. Interrupt request signal from the interrupt controller to the CPU
3. Startup request signal from the interrupt controller to the IDMA

If the #BUSGET signal is found to be active when the external bus master is monitoring it, release #BUSREQ back high to drop the request for bus ownership control.
When using the #BUSGET signal to only monitor a refresh request, set the interrupt controller in such a way that no interrupt request or IDMA startup request will be generated.

To direct the P31 pin for output of the #BUSGET signal, write "1" to CFP31 (D1) / P3 function select register (0x402DC [Byte]) and CFEX3 (D3) / Port function extension register (0x402DF [Byte]).

Power-down Control by External Device

In addition to requesting the releasing of bus ownership control described above, it is possible to place the CPU in a HALT state by using the #BUSREQ signal. This allows the CPU to be stopped during bus operation by an external bus master in order to conserve power.
This function is enabled by writing "1" to SEPD (D1) / Bus control register (0x4812E).
If SEPD = "1", the CPU and the BCU stop operating when the #BUSREQ pin is lowered, thus entering a HALT state. This HALT state is not cleared by an interrupt from the internal peripheral circuits and remains set until the #BUSREQ pin is released back high. Unlike in the case of ordinary releasing of the bus by #BUSREQ, the address bus and bus control signals are not placed in high-impedance state.

For a DRAM refresh request that may arise in this HALT state, take one of the corrective measures described above.

I/O Memory of BCU

Table 4.23 shows the control bits of the BCU. These I/O memories are mapped into the area (0x48000 and following addresses) used for the internal 16-bit peripheral circuits. However, these I/O memories can be accessed in bytes or words, as well as in half-words.
For the control bits of the external system interface pins assigned to the I/O ports, and for details on how to control the 8-bit programmable timer 0 in order to generate a DRAM refresh cycle, refer to each corresponding section in this manual.

Table 4.23 Control Bits of External System Interface

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Areas 18–15	**0048120**	DF	–	reserved	–	–	–	0 when being read.
set-up register	**(HW)**	DE	**A18SZ**	Areas 18–17 device size selection	1 8 bits / 0 16 bits	0	R/W	
		DD	**A18DF1**	Areas 18–17	A18DF[1:0] Number of cycles	1	R/W	
		DC	**A18DF0**	output disable delay time	1 1 3.5	1		
					1 0 2.5			
					0 1 1.5			
					0 0 0.5			
		DB	–	reserved	–	–	–	0 when being read.
		DA	**A18WT2**	Areas 18–17 wait control	A18WT[2:0] Wait cycles	1	R/W	
		D9	**A18WT1**		1 1 1 7	1		
		D8	**A18WT0**		1 1 0 6	1		
					1 0 1 5			
					1 0 0 4			
					0 1 1 3			
					0 1 0 2			
					0 0 1 1			
					0 0 0 0			
		D7	–	reserved	–	–	–	0 when being read.
		D6	**A16SZ**	Areas 16–15 device size selection	1 8 bits / 0 16 bits	0	R/W	
		D5	**A16DF1**	Areas 16–15	A16DF[1:0] Number of cycles	1	R/W	
		D4	**A16DF0**	output disable delay time	1 1 3.5	1		
					1 0 2.5			
					0 1 1.5			
					0 0 0.5			
		D3	–	reserved	–	–	–	0 when being read.
		D2	**A16WT2**	Areas 16–15 wait control	A16WT[2:0] Wait cycles	1	R/W	
		D1	**A16WT1**		1 1 1 7	1		
		D0	**A16WT0**		1 1 0 6	1		
					1 0 1 5			
					1 0 0 4			
					0 1 1 3			
					0 1 0 2			
					0 0 1 1			
					0 0 0 0			
Areas 14–13	**0048122**	DF–9	–	reserved	–	–	–	0 when being read.
set-up register	**(HW)**	D8	**A14DRA**	Area 14 DRAM selection	1 Used / 0 Not used	0	R/W	
		D7	**A13DRA**	Area 13 DRAM selection	1 Used / 0 Not used	0	R/W	
		D6	**A14SZ**	Areas 14–13 device size selection	1 8 bits / 0 16 bits	0	R/W	
		D5	**A14DF1**	Areas 14–13	A14DF[1:0] Number of cycles	1	R/W	
		D4	**A14DF0**	output disable delay time	1 1 3.5	1		
					1 0 2.5			
					0 1 1.5			
					0 0 0.5			
		D3	–	reserved	–	–	–	0 when being read.
		D2	**A14WT2**	Areas 14–13 wait control	A14WT[2:0] Wait cycles	1	R/W	
		D1	**A14WT1**		1 1 1 7	1		
		D0	**A14WT0**		1 1 0 6	1		
					1 0 1 5			
					1 0 0 4			
					0 1 1 3			
					0 1 0 2			
					0 0 1 1			
					0 0 0 0			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Areas 12–11	**0048124**	DF–7	**–**	reserved	–	–	–	0 when being read.
set-up register	**(HW)**	D6	**A12SZ**	Areas 12–11 device size selection	1 8 bits 0 16 bits	0	R/W	
		D5	**A12DF1**	Areas 12–11	A18DF[1:0] Number of cycles	1	R/W	
		D4	**A12DF0**	output disable delay time	1 1 3.5	1		
					1 0 2.5			
					0 1 1.5			
					0 0 0.5			
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**A12WT2**	Areas 12–11 wait control	A18WT[2:0] Wait cycles	1	R/W	
		D1	**A12WT1**		1 1 1 7	1		
		D0	**A12WT0**		1 1 0 6	1		
					1 0 1 5			
					1 0 0 4			
					0 1 1 3			
					0 1 0 2			
					0 0 1 1			
					0 0 0 0			
Areas 10–9	**0048126**	DF	**–**	reserved	–	–	–	0 when being read.
set-up register	**(HW)**	DE	**A10IR2**	Area 10 internal ROM wait control	A10IR[2:0] ROM size	1	R/W	
		DD	**A10IR1**	Area 10 internal ROM size	1 1 1 2MB	1		
		DC	**A10IR0**	selection	1 1 0 1MB	1		
					1 0 1 512KB			
					1 0 0 256KB			
					0 1 1 128KB			
					0 1 0 64KB			
					0 0 1 32KB			
					0 0 0 16KB			
		DB	**–**	reserved	–	–	–	0 when being read.
		DA	**A10BW1**	Areas 10–9	A10BW[1:0] Wait cycles	0	R/W	
		D9	**A10BW0**	burst ROM	1 1 3	0		
				burst read cycle wait control	1 0 2			
					0 1 1			
					0 0 0			
		D8	**A10DRA**	Area 10 burst ROM selection	1 Used 0 Not used	0	R/W	
		D7	**A9DRA**	Area 9 burst ROM selection	1 Used 0 Not used	0	R/W	
		D6	**A10SZ**	Areas 10–9 device size selection	1 8 bits 0 16 bits	0	R/W	
		D5	**A10DF1**	Areas 10–9	A10DF[1:0] Number of cycles	1	R/W	
		D4	**A10DF0**	output disable delay time	1 1 3.5	1		
					1 0 2.5			
					0 1 1.5			
					0 0 0.5			
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**A10WT2**	Areas 10–9 wait control	A10WT[2:0] Wait cycles	1	R/W	
		D1	**A10WT1**		1 1 1 7	1		
		D0	**A10WT0**		1 1 0 6	1		
					1 0 1 5			
					1 0 0 4			
					0 1 1 3			
					0 1 0 2			
					0 0 1 1			
					0 0 0 0			
Areas 8–7	**0048128**	DF–9	**–**	reserved	–	–	–	0 when being read.
set-up register	**(HW)**	D8	**A8DRA**	Area 8 DRAM selection	1 Used 0 Not used	0	R/W	
		D7	**A7DRA**	Area 7 DRAM selection	1 Used 0 Not used	0	R/W	
		D6	**A8SZ**	Areas 8–7 device size selection	1 8 bits 0 16 bits	0	R/W	
		D5	**A8DF1**	Areas 8–7	A8DF[1:0] Number of cycles	1	R/W	
		D4	**A8DF0**	output disable delay time	1 1 3.5	1		
					1 0 2.5			
					0 1 1.5			
					0 0 0.5			
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**A8WT2**	Areas 8–7 wait control	A8WT[2:0] Wait cycles	1	R/W	
		D1	**A8WT1**		1 1 1 7	1		
		D0	**A8WT0**		1 1 0 6	1		
					1 0 1 5			
					1 0 0 4			
					0 1 1 3			
					0 1 0 2			
					0 0 1 1			
					0 0 0 0			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Areas 6–4 set-up register	**004812A (HW)**	DF–E	–	reserved	–	–	–	0 when being read.
		DD	**A6DF1**	Area 6	A6DF[1:0] Number of cycles	1	R/W	
		DC	**A6DF0**	output disable delay time	1 1 3.5	1		
					1 0 2.5			
					0 1 1.5			
					0 0 0.5			
		DB	–	reserved	–	–	–	0 when being read.
		DA	**A6WT2**	Area 6 wait control	A6WT[2:0] Wait cycles	1	R/W	
		D9	**A6WT1**		1 1 1 7	1		
		D8	**A6WT0**		1 1 0 6	1		
					1 0 1 5			
					1 0 0 4			
					0 1 1 3			
					0 1 0 2			
					0 0 1 1			
					0 0 0 0			
		D7	–	reserved	–	–	–	0 when being read.
		D6	**A5SZ**	Areas 5–4 device size selection	1 8 bits / 0 16 bits	0	R/W	
		D5	**A5DF1**	Areas 5–4	A5DF[1:0] Number of cycles	1	R/W	
		D4	**A5DF0**	output disable delay time	1 1 3.5	1		
					1 0 2.5			
					0 1 1.5			
					0 0 0.5			
		D3	–	reserved		–	–	0 when being read.
		D2	**A5WT2**	Areas 5–4 wait control	A5WT[2:0] Wait cycles	1	R/W	
		D1	**A5WT1**		1 1 1 7	1		
		D0	**A5WT0**		1 1 0 6	1		
					1 0 1 5			
					1 0 0 4			
					0 1 1 3			
					0 1 0 2			
					0 0 1 1			
					0 0 0 0			
Bus control register	**004812E (HW)**	DF	**RBCLK**	BCLK output control	1 Fixed at H / 0 Enabled	0	R/W	
		DE	–	reserved	–	0	–	Writing 1 not allowed.
		DD	**RBST8**	Burst ROM burst mode selection	1 8-successive / 0 4-successive	0	R/W	
		DC	**REDO**	DRAM page mode selection	1 EDO / 0 Fast page	0	R/W	
		DB	**RCA1**	Column address size selection	RCA[1:0] Size	0	R/W	
		DA	**RCA0**		1 1 11	0		
					1 0 10			
					0 1 9			
					0 0 8			
		D9	**RPC2**	Refresh enable	1 Enabled / 0 Disabled	0	R/W	
		D8	**RPC1**	Refresh method selection	1 Self-refresh / 0 CBR-refresh	0	R/W	
		D7	**RPC0**	Refresh RPC delay setup	1 2.0 / 0 1.0	0	R/W	
		D6	**RRA1**	Refresh RAS pulse width	RRA[1:0] Number of cycles	0	R/W	
		D5	**RRA0**	selection	1 1 5	0		
					1 0 4			
					0 1 3			
					0 0 2			
		D4	–	reserved	–	0	–	Writing 1 not allowed.
		D3	**SBUSST**	External interface method selection	1 #BSL / 0 A0	0	R/W	
		D2	**SEMAS**	External bus master setup	1 Existing / 0 Nonexistent	0	R/W	
		D1	**SEPD**	External power-down control	1 Enabled / 0 Disabled	0	R/W	
		D0	**SWAITE**	#WAIT enable	1 Enabled / 0 Disabled	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
DRAM timing set-up register	**0048130 (HW)**	DF–C	–	reserved	–				–	–	0 when being read.
		DB	**A3EEN**	Area 3 emulation	1	Internal ROM	0	Emulation	1	R/W	
		DA	**CEFUNC1**	#CE pin function selection	CFFUNC[1:0]		#CE output		0	R/W	
		D9	**CEFUNC0**		1	x	#CE7/8..#CE17/18		0		
					0	1	#CE6..#CE17				
					0	0	#CE4..#CE10				
		D8	**CRAS**	Successive RAS mode setup	1	Successive	0	Normal	0	R/W	
		D7	**RPRC1**	DRAM	RPRC[1:0]		Number of cycles		0	R/W	
		D6	**RPRC0**	RAS precharge cycles selection	1	1	4		0		
					1	0	3				
					0	1	2				
					0	0	1				
		D5	–	reserved	–				–	–	0 when being read.
		D4	**CASC1**	DRAM	CASC[1:0]		Number of cycles		0	R/W	
		D3	**CASC0**	CAS cycles selection	1	1	4		0		
					1	0	3				
					0	1	2				
					0	0	1				
		D2	–	reserved	–				–	–	0 when being read.
		D1	**RASC1**	DRAM	RASC[1:0]		Number of cycles		0	R/W	
		D0	**RASC0**	RAS cycles selection	1	1	4		0		
					1	0	3				
					0	1	2				
					0	0	1				
Access control register	**0048132 (HW)**	DF	**A18IO**	Area 18, 17 internal/external access	1	Internal access	0	External access	0	R/W	
		DE	**A16IO**	Area 16, 15 internal/external access					0	R/W	
		DD	**A14IO**	Area 14, 13 internal/external access					0	R/W	
		DC	**A12IO**	Area 12, 11 internal/external access					0	R/W	
		DB	–	reserved	–				0	–	0 when being read.
		DA	**A8IO**	Area 8, 7 internal/external access	1	Internal access	0	External access	0	R/W	
		D9	**A6IO**	Area 6 internal/external access					0	R/W	
		D8	**A5IO**	Area 5, 4 internal/external access					0	R/W	
		D7	**A18EC**	Area 18, 17 endian control	1	Big endian	0	Little endian	0	R/W	
		D6	**A16EC**	Area 16, 15 endian control					0	R/W	
		D5	**A14EC**	Area 14, 13 endian control					0	R/W	
		D4	**A12EC**	Area 12, 11 endian control					0	R/W	
		D3	**A10EC**	Area 10, 9 endian control					0	R/W	
		D2	**A8EC**	Area 8, 7 endian control					0	R/W	
		D1	**A6EC**	Area 6 endian control					0	R/W	
		D0	**A5EC**	Area 5, 4 endian control					0	R/W	
G/A read signal control register	**0048138 (HW)**	DF	**A18AS**	Area 18, 17 address strobe signal	1	Enabled	0	Disabled	0	R/W	
		DE	**A16AS**	Area 16, 15 address strobe signal					0	R/W	
		DD	**A14AS**	Area 14, 13 address strobe signal					0	R/W	
		DC	**A12AS**	Area 12, 11 address strobe signal					0	R/W	
		DB	–	reserved	–				0	–	0 when being read.
		DA	**A8AS**	Area 8, 7 address strobe signal	1	Enabled	0	Disabled	0	R/W	
		D9	**A6AS**	Area 6 address strobe signal					0	R/W	
		D8	**A5AS**	Area 5, 4 address strobe signal					0	R/W	
		D7	**A18RD**	Area 18, 17 read signal	1	Enabled	0	Disabled	0	R/W	
		D6	**A16RD**	Area 16, 15 read signal					0	R/W	
		D5	**A14RD**	Area 14, 13 read signal					0	R/W	
		D4	**A12RD**	Area 12, 11 read signal					0	R/W	
		D3	–	reserved	–				0	–	0 when being read.
		D2	**A8RD**	Area 8, 7 read signal	1	Enabled	0	Disabled	0	R/W	
		D1	**A6RD**	Area 6 read signal					0	R/W	
		D0	**A5RD**	Area 5, 4 read signal					0	R/W	
BCLK select register	**004813A (B)**	D7–4	–	reserved	–				0	–	0 when being read.
		D3	**A1X1MD**	Area 1 access-speed	1	2 cycles	0	4 cycles	0	R/W	x2 speed mode only
		D2	–	reserved	–				0	–	0 when being read.
		D1	**BCLKSEL1**	BCLK output clock selection	BCLKSEL[1:0]		BCLK		0	R/W	
		D0	**BCLKSEL0**		1	1	PLL_CLK		0		
					1	0	OSC3_CLK				
					0	1	BCU_CLK				
					0	0	CPU_CLK				

A18SZ: Areas 18–17 device size selection (DE) / Areas 18–15 set-up register (0x48120)
A16SZ: Areas 16–15 device size selection (D6) / Areas 18–15 set-up register (0x48120)
A14SZ: Areas 14–13 device size selection (D6) / Areas 14–13 set-up register (0x48122)
A12SZ: Areas 12–11 device size selection (D6) / Areas 12–11 set-up register (0x48124)
A10SZ: Areas 10–9 device size selection (D6) / Areas 10–9 set-up register (0x48126)
A8SZ: Areas 8–7 device size selection (D6) / Areas 8–7 set-up register (0x48128)
A5SZ: Areas 5–4 device size selection (D6) / Areas 6–4 set-up register (0x4812A)

Select the size of the device connected to each area.

Write "1": 8 bits
Write "0": 16 bits
Read: Valid

A device size can be selected for every two areas.
An 8-bit size is selected by writing "1" to AxxSZ and a 16-bit size is selected by writing "0" to AxxSZ.
Area 6 has its first half (0x300000 through 0x37FFFF) fixed to an 8-bit device and the last half (0x380000 through 0x3FFFFF) fixed to a 16-bit device.
At cold start, these bits are set to "0" (16 bits). At hot start, these bits retain their status before being initialized.

A18DF1–A18DF0: Areas 18–17 output disable delay time (D[D:C]) / Areas 18–15 set-up register (0x48120)
A16DF1–A16DF0: Areas 16–15 output disable delay time (D[5:4]) / Areas 18–15 set-up register (0x48120)
A14DF1–A14DF0: Areas 14–13 output disable delay time (D[5:4]) / Areas 14–13 set-up register (0x48122)
A12DF1–A12DF0: Areas 12–11 output disable delay time (D[5:4]) / Areas 12–11 set-up register (0x48124)
A10DF1–A10DF0: Areas 10–9 output disable delay time (D[5:4]) / Areas 10–9 set-up register (0x48126)
A8DF1–A8DF0: Areas 8–7 output disable delay time (D[5:4]) / Areas 8–7 set-up register (0x48128)
A6DF1–A6DF0: Area 6 output disable delay time (D[D:C]) / Areas 6–4 set-up register (0x4812A)
A5DF1–A5DF0: Areas 5–4 output disable delay time (D[5:4]) / Areas 6–4 set-up register (0x4812A)

Set the output-disable delay time.

Table 4.24 Output Disable Delay Time

AxxDF1	AxxDF0	Delay time
1	1	3.5 cycles
1	0	2.5 cycles
0	1	1.5 cycles
0	0	0.5 cycles

When using a device that has a long output-disable time, set a delay time to ensure that no contention for the data bus occurs during the bus operation immediately after a device is read.
At cold start, these bits are set to "11" (3.5 cycles). At hot start, the bits retain their status before being initialized.

A18WT2–A18WT0: Areas 18–17 wait control (D[A:8]) / Areas 18–15 set-up register (0x48120)
A16WT2–A16WT0: Areas 16–15 wait control (D[2:0]) / Areas 18–15 set-up register (0x48120)
A14WT2–A14WT0: Areas 14–13 wait control (D[2:0]) / Areas 14–13 set-up register (0x48122)
A12WT2–A12WT0: Areas 12–11 wait control (D[2:0]) / Areas 12–11 set-up register (0x48124)
A10WT2–A10WT0: Areas 10–9 wait control (D[2:0]) / Areas 10–9 set-up register (0x48126)
A8WT2–A8WT0: Areas 8–7 wait control (D[2:0]) / Areas 8–7 set-up register (0x48128)
A6WT2–A6WT0: Area 6 wait control (D[A:8]) / Areas 6–4 set-up register (0x4812A)
A5WT2–A5WT0: Areas 5–4 wait control (D[2:0]) / Areas 6–4 set-up register (0x4812A)

Set the number of wait cycles to be inserted when accessing an SRAM device.
The values 0 through 7 written to the control bits equal the number of wait cycles inserted.
Note that the write cycle consists of a minimum of two cycles, so that a writing 0 or 1 is invalid.
When an SRAM device is connected, wait cycles derived via the #WAIT pin can also be inserted. In this case too, the wait cycles set by AxxWT are valid.
The DRAM read/write cycles do not have wait cycles inserted that are set by AxxWT or derived from the #WAIT pin.

The burst read cycle of a burst ROM (except for the first access) also does not have any wait cycle inserted. The first read cycle of a burst ROM and the write cycle to the burst ROM area have wait cycles inserted that are set by AxxWT . Wait cycles derived from the #WAIT pin also can be inserted in the cycle for writing to the burst ROM area.
At cold start, these bits are set to "111" (7 cycles). At hot start, the bits retain their status before being initialized.

A14DRA: Area 14 DRAM selection (D8) / Areas 14–13 set-up register (0x48122)
A13DRA: Area 13 DRAM selection (D7) / Areas 14–13 set-up register (0x48122)
A8DRA: Area 8 DRAM selection (D8) / Areas 8–7 set-up register (0x48128)
A7DRA: Area 7 DRAM selection (D7) / Areas 8–7 set-up register (0x48128)

Select the DRAM direct interface.

Write "1": DRAM is used
Write "0": DRAM is not used
Read: Valid

When DRAM is used by connecting it directly to the BCU, write "1" to this bit. The ordinary SRAM interface is selected by writing "0" to the control bit.
The areas to which DRAM can be connected are areas 8 and 7 when the CEFUNC = "0", or areas 14 and 13 when the bit = "1".
At cold start, these bits are set to "0" (DRAM not used). At hot start, the bits retain their status before being initialized.

A10IR2–A10IR0: Area 10 internal ROM size selection (D[D:B]) / Areas 10–9 set-up register (0x48126)

Select an area 10 internal/emulation memory size.

Table 4.25 Area 10 Internal ROM Size

A10IR2	A10IR1	A10IR0	ROM size
0	0	0	16KB
0	0	1	32KB
0	1	0	64KB
0	1	1	128KB
1	0	0	256KB
1	0	1	512KB
1	1	0	1MB
1	1	1	2MB

At cold start, A10IR is set to "111" (2MB). At hot start, A10IR retains its status before being initialized.

A10BW1–A10BW0: Burst read cycle wait control (D[A:9]) / Areas 10–9 set-up register (0x48126)

Set the number of wait cycles inserted during a burst read.
The values 0 to 3 written to the bits constitute the number of wait cycles inserted. The contents set here are applied to both areas 10 and 9. The wait cycles set by AxxWT are inserted in the first read cycle of burst ROM and in the burst ROM write cycle. For the burst ROM write cycle, the wait cycles set via the #WAIT pin can also be used.
At cold start, A10BW is set to "0" (no wait cycle). At hot start, A10BW retains its status before being initialized.

A10DRA: Area 10 burst ROM selection (D8) / Areas 10–9 set-up register (0x48126)
A9DRA: Area 9 burst ROM selection (D7) / Areas 10–9 set-up register (0x48126)

Set areas 10 and 9 for use of burst ROM.

Write "1": Burst ROM is used
Write "0": Burst ROM is not used
Read: Valid

When using burst ROM, write "1" to the control bit. The ordinary SRAM interface is selected by writing "0" to the bit.
Area 9 can only be used when the CEFUNC = "00".
At cold start, these bits are set to "0" (burst ROM not used). At hot start, the bits retain their status before being initialized.

RBCLK: BCLK output control (DF) / Bus control register (0x4812E)

Control the bus clock BCLK to enable or disable external output.

Write "1": Fixed at high level
Write "0": Output enabled
Read: Valid

To stop outputting the bus clock from the BCLK pin, write "1" to RBCLK. When the clock output is stopped, the BCLK pin is fixed at high level. The bus clock output from the BCLK pin is enabled by writing "0" to RBCLK.
The bus clock output from the BCLK pin also is stopped in the HALT2 and the SLEEP modes.
At cold start, the RBCLK is set to "0" (output enabled). At hot start, RBCLK retains its status before being initialized.

RBST8: Burst mode selection (DD) / Bus control register (0x4812E)

Set the operation mode during a burst read.

Write "1": 8-successive-burst mode
Write "0": 4-successive-burst mode
Read: Valid

The 8-successive-burst mode is selected by writing "1" to RBST8 and the 4-successive-burst mode is selected by writing "0" to RBST8. This setting is valid when areas 10 and 9 are set for burst ROM, and the setting is applied to both areas simultaneously.
At cold start, RBST8 is set to "0" (4-successive-burst mode). At hot start, RBST8 retains its status before being initialized.

REDO: Page mode selection (DC) / Bus control register (0x4812E)

Select the page mode of DRAM.

Write "1": EDO-page mode
Write "0": Fast-page mode
Read: Valid

When using EDO DRAM, write "1" to REDO to select the EDO-page mode.
The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
At cold start, REDO is set to "0" (fast-page mode). At hot start, REDO retains its status before being initialized.

RCA1–RCA0: Column address size selection (D[B:A]) / Bus control register (0x4812E)

Select the column address size of DRAM.

Table 4.26 Column Address Size

RCA1	RCA0	Column address size
1	1	11
1	0	10
0	1	9
0	0	8

The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
RCA can be read to obtain its set value.
At cold start, RCA is set to "0" (8 bits). At hot start, RCA retain its status before being initialized.

RPC2: Refresh enable (D9) / Bus control register (0x4812E)

Control the DRAM refresh function.

Write "1": Enabled
Write "0": Disabled
Read: Valid

When DRAM is connected directly, a refresh cycle is generated by writing "1" to RPC2. The internal refresh function is disabled by writing "0" to RPC2.
Since the BCU stops operating in the HALT2 and the SLEEP modes, no refresh cycle is generated regardless of how this bit is set.
The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
At cold start, RPC2 is set to "0" (disabled). At hot start, RPC2 retains its status before being initialized.

RPC1: Refresh method selection (D8) / Bus control register (0x4812E)

Select the DRAM refresh method.

Write "1": Self-refresh
Write "0": CAS-before-RAS refresh
Read: Valid

To perform a CAS-before-RAS refresh, set RPC1 to "0" and then RPC2 to "1". This causes the underflow output signal of the 8-bit programmable timer 0 is fed to the DRAM interface, at which timing a refresh cycle is generated.
To start a self-refresh, set RPC1 to "1" and then RPC2 to "1". The self-refresh is disabled by writing "0" to RPC2.
The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
At cold start, RPC1 is set to "0" (CAS-before-RAS refresh). At hot start, RPC1 retains its status before being initialized.

RPC0: Refresh RPC delay (D7) / Bus control register (0x4812E)

Set a RPC delay when at start of refresh.

Write "1": 2 cycles
Write "0": 1 cycle
Read: Valid

Set a time from the immediately preceding precharge to the falling transition of #HCAS/#LCAS necessary in order to perform a refresh. This time is 2 cycles when RPC0 = "1" or 1 cycle when RPC0 = "0".
The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
At cold start, RPC0 is set to "0" (1 cycle). At hot start, RPC0 retains its status before being initialized.

RRA1–RRA0: Refresh RAS pulse width selection (D[6:5]) / Bus control register (0x4812E)

Select the RAS pulse width of a CAS-before-RAS refresh.

Table 4.27 Refresh RAS Pulse Width

RRA1	RRA0	Pulse width
1	1	5 cycles
1	0	4 cycles
0	1	3 cycles
0	0	2 cycles

The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
The RRA can be read to obtain their set value.
At cold start, RRA is set to "0" (2 cycles). At hot start, RRA retains its status before being initialized.

SBUSST: External interface method select register (D3) / Bus control register (0x4812E)

Select the interface method of an SRAM device.

Write "1": #BSL system
Write "0": A0 system
Read: Valid

When using the #BSL system, write "1" to SBUSST.
The contents set here are applied to all areas that are set for the SRAM type.
At cold start, SBUSST is set to "0" (A0 system). At hot start, SBUSST retains its status before being initialized.

SEMAS: External bus master setup (D2) / Bus control register (0x4812E)

Specify whether an external bus master exists.

Write "1": Existing
Write "0": Nonexistent
Read: Valid

A request for bus ownership control via the #BUSREQ pin is made acceptable by writing "1" to SEMAS. If the system does not have any external bus master, fix this register at "0".
At cold start, SEMAS is set to "0" (nonexistent). At hot start, SEMAS retains its status before being initialized.

SEPD: External power-down control (D1) / Bus control register (0x4812E)

Enable or disable the CPU's power-down control by an external bus master.

Write "1": Enabled
Write "0": Disabled
Read: Valid

Power-down control via an external pin (#BUSREQ) is enabled by writing "1" to SEPD. If the #BUSREQ pin is lowered when external power-down control is thus enabled, the CPU is placed in a HALT state, allowing for reduction in power consumption.
At cold start, SEPD is set to "0" (disabled). At hot start, SEPD retains its status before being initialized.

SWAITE: #WAIT enable (D0) / Bus control register (0x4812E)

Enable or disable wait cycle control via the #WAIT pin.

Write "1": Enabled
Write "0": Disabled
Read: Valid

A wait request from an SRAM device is made acceptable by writing "1" to SWAITE. The wait request signal input from the #WAIT pin is sampled at each falling edge of the bus clock when executing an SRAM read/write cycle. Wait cycles are inserted until the wait request signal is sampled and detected as high (inactive).
Wait control for 0 to 7 cycles can be accomplished by AxxWT without using the #WAIT pin. However, since the setting via AxxWT is applied to every two areas, the number of wait cycles may be controlled individually in each area or more than 7 wait cycles may be set. In such a case, use an external wait request via the #WAIT pin.
Wait requests from the #WAIT pin are ignored when SWAITE = "0".
The contents set here are applied to all areas that are set for SRAM, and are also effective for write cycles in the areas that are set for burst ROM.
At cold start, SWAITE is set to "0" (disabled). At hot start, SWAITE retains its status before being initialized.

A3EEN: Area 3 emulation (DB) / DRAM timing set-up register (0x48130)

Select area 3 emulation mode.

Write "1": Internal ROM mode
Write "0": Emulation mode
Read: Valid

When "1" is written to A3EEN, internal ROM emulation mode is selected and the external device will be accessed with the same condition as the internal ROM. When "0" is written, the internal ROM will be used for accessing area 3. This bit functions the same as the EA3MD pin. The bit status and the pin status are logically ORed.
At cold start, A3EEN is set to "1" (internal ROM mode). At hot start, A3EEN retains its status before being initialized.

CEFUNC1–CEFUNC0: #CE pin function selection (D[A:9]) / DRAM timing set-up register (0x48130)

Change the #CE pin-assigned area.

Table 4.28 #CE Output Assignment

Pin	CEFUNC = "00"	CEFUNC = "01"	CEFUNC = "1x"
#CE4	#CE4	#CE11	#CE11+#CE12
#CE5	#CE5	#CE15	#CE15+#CE16
#CE6	#CE6	#CE6	#CE7+#CE8
#CE7/#RAS0	#CE7/#RAS0	#CE13/#RAS2	#CE13/#RAS2
#CE8/#RAS1	#CE8/#RAS1	#CE14/#RAS3	#CE14/#RAS3
#CE9	#CE9	#CE17	#CE17+#CE18
#CE10EX	#CE10EX	#CE10EX	#CE9+#CE10EX

(Default: CEFUNC = "00")

The high-order areas that are made available for use by writing "01" to CEFUNC can be larger in size than the default low-order areas. For example, when using DRAM in default settings, the available space is 4MB in areas 7 and 8. However, if areas 13 and 14 are used, up to 32MB of DRAM can be used. The same applies to the other areas.
Furthermore, when CEFUNC is set to "10" or "11", four chip enable signal is expanded into two area size.
At cold start, CEFUNC is set to "00". At hot start, CEFUNC retains its status before being initialized.

CRAS: Successive RAS mode (D8) / DRAM timing set-up register (0x48130)

Set the successive RAS mode.

Write "1": Successive RAS mode
Write "0": Normal mode
Read: Valid

In systems using DRAM, the successive RAS mode is entered by writing "1" to CRAS. In this mode, read/write operations can be performed in page mode even when DRAM accesses do not occur back-to-back.
When using the successive RAS mode, be sure to use #DRD for the read signal and #DWE for the write signal for low-byte.
When CRAS = "0", random read/write cycles are used for non-successive DRAM accesses.
The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
At cold start, CRAS is set to "0" (normal mode). At hot start, CRAS retains its status before being initialized.

RPRC1–RPRC0: Number of RAS precharge cycles (D[7:6]) / DRAM timing set-up register (0x48130)

Select the number of precharge cycles during a DRAM access.

Table 4.29 Number of RAS Precharge Cycles

RPRC1	RPRC0	Number of cycles
1	1	4 cycles
1	0	3 cycles
0	1	2 cycles
0	0	1 cycle

The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
At cold start, RPRC is set to "0" (1 cycle). At hot start, RPRC retains its status before being initialized.

CASC1–CASC0: Number of CAS cycles (D[4:3]) / DRAM timing set-up register (0x48130)

Select the number of CAS cycles during a DRAM access.

Table 4.30 Number of CAS Cycles

CASC1	CASC0	Number of cycles
1	1	4 cycles
1	0	3 cycles
0	1	2 cycles
0	0	1 cycle

The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
At cold start, CASC is set to "0" (1 cycle). At hot start, CASC retains its status before being initialized.

RASC1–RASC0: Number of RAS cycles (D[1:0]) / DRAM timing set-up register (0x48130)

Select the number of RAS cycles during a DRAM access.

Table 4.31 Number of RAS Cycles

RASC1	RASC0	Number of cycles
1	1	4 cycles
1	0	3 cycles
0	1	2 cycles
0	0	1 cycle

The contents set here are applied to all of areas 14, 13, 8, and 7 that are set for DRAM.
At cold start, RASC is set to "0" (1 cycle). At hot start, RASC retains its status before being initialized.

A18IO: Areas 18–17 internal/external access selection (DF) / Access control register (0x48132)
A16IO: Areas 16–15 internal/external access selection (DE) / Access control register (0x48132)
A14IO: Areas 14–13 internal/external access selection (DD) / Access control register (0x48132)
A12IO: Areas 12–11 internal/external access selection (DC) / Access control register (0x48132)
A8IO: Areas 8–7 internal/external access selection (DA) / Access control register (0x48132)
A6IO: Area 6 internal/external access selection (D9) / Access control register (0x48132)
A5IO: Areas 5–4 internal/external access selection (D8) / Access control register (0x48132)

Select either internal access or external access for each area.

Write "1": Internal access
Write "0": External access
Read: Valid

When AxxIO is set to "1", the internal device that mapped to the corresponding area is accessed. When AxxIO is set to "0", the external device is accessed.
At cold start, these bits are set to "0" (external access). At hot start, these bits retain their status before being initialized.

A18EC: Areas 18–17 little/big endian method selection (D7) / Access control register (0x48132)
A16EC: Areas 16–15 little/big endian method selection (D6) / Access control register (0x48132)
A14EC: Areas 14–13 little/big endian method selection (D5) / Access control register (0x48132)
A12EC: Areas 12–11 little/big endian method selection (D4) / Access control register (0x48132)
A10EC: Areas 10–9 little/big endian method selection (D3) / Access control register (0x48132)
A8EC: Areas 8–7 little/big endian method selection (D2) / Access control register (0x48132)
A6EC: Area 6 little/big endian method selection (D1) / Access control register (0x48132)
A5EC: Areas 5–4 little/big endian method selection (D0) / Access control register (0x48132)

Select either little endian or big-endian method for accessing each area.

Write "1": Big-endian
Write "0": Little-endian
Read: Valid

When AxxEC is set to "1", the corresponding area is accessed in big-endian method. When AxxEC is set to "0", the area is accessed in little-endian method. When using area 10 as the boot area, fix A10EC at "0" (little-endian).
At cold start, these bits are set to "0" (little-endian). At hot start, these bits retain their status before being initialized.

A18AS: Areas 18–17 address strobe signal (DF) / G/A read signal control register (0x48138)
A16AS: Areas 16–15 address strobe signal (DE) / G/A read signal control register (0x48138)
A14AS: Areas 14–13 address strobe signal (DD) / G/A read signal control register (0x48138)
A12AS: Areas 12–11 address strobe signal (DC) / G/A read signal control register (0x48138)
A8AS: Areas 8–7 address strobe signal (DA) / G/A read signal control register (0x48138)
A6AS: Area 6 address strobe signal (D9) / G/A read signal control register (0x48138)
A5AS: Areas 5–4 address strobe signal (D8) / G/A read signal control register (0x48138)

Enable/disable the exclusive address strobe signal output.

Write "1": Enabled
Write "0": Disabled
Read: Valid

If AxxAS is set to "1", the exclusive address strobe signal is output from #GAAS (P21) pin when the corresponding area is accessed. If AxxAS is set to "0", the signal output is disabled.
At cold start, these bits are set to "0" (disabled). At hot start, these bits retain their status before being initialized.

A18RD: Areas 18–17 read signal (D7) / G/A read signal control register (0x48138)
A16RD: Areas 16–15 read signal (D6) / G/A read signal control register (0x48138)
A14RD: Areas 14–13 read signal (D5) / G/A read signal control register (0x48138)
A12RD: Areas 12–11 read signal (D4) / G/A read signal control register (0x48138)
A8RD: Areas 8–7 read signal (D2) / G/A read signal control register (0x48138)
A6RD: Area 6 read signal (D1) / G/A read signal control register (0x48138)
A5RD: Areas 5–4 read signal (D0) / G/A read signal control register (0x48138)

Enable/disable the exclusive read signal output.

Write "1": Enabled
Write "0": Disabled
Read: Valid

If AxxRD is set to "1", the exclusive read signal is output from #GARD (P31) pin when the corresponding area is read. If AxxRD is set to "0", the signal output is disabled.
At cold start, these bits are set to "0" (disabled). At hot start, these bits retain their status before being initialized.

BCLKSEL1–BCLKSEL0: BCLK output clock selection (D[1:0]) / BCLK select register (0x4813A)

Select a clock to be output from the BCLK pin.

Table 4.32 Selection of BCLK Output Clock

BCLKSEL1	BCLKSEL0	Output clock
1	1	PLL_CLK (PLL output clock)
1	0	OSC3_CLK (OSC3 oscillation clock)
0	1	BCU_CLK (BCU operating clock)
0	0	CPU_CLK (CPU operating clock)

PLL_CLK: PLL output clock. This clock is stable and kept as output except in the following cases:
1. When the PLL is off by setting the PLLS[1:0] pins.
2. When the OSC3 (high-speed) oscillation is stopped by executing the SLP instruction.
3. When the OSC3 (high-speed) oscillation is stopped using the CLG register.

Note that the PLL_CLK clock is out of phase with the CPU operating clock.

OSC3_CLK: OSC3 (high-speed) oscillation circuit output clock. This clock is stable and kept as output except in the following cases:
1. When the OSC3 (high-speed) oscillation is stopped by executing the SLP instruction.
2. When the OSC3 (high-speed) oscillation is stopped using the CLG register.

Note that the OSC3_CLK clock is out of phase with the CPU operating clock.

BCU_CLK: Bus clock in the bus controller. This clock varies according to the bus cycle speed. Furthermore, the clock frequency changes dynamically in x2 speed mode as follows:
1. When the internal RAM/ROM is accessed, x2 clock (e.g., 50 MHz same as the CPU operating clock) is output.
2. When an external device is accessed via the external bus, x1 clock (e.g., 25 MHz) is output.

This dynamic change (e.g., between 50 MHz and 25 MHz) does not affect the external memory access timing, such as position relationship between the rising or falling edge of the 25 MHz clock and the falling edge of the #WR signal. (It is the same as that in the x1 speed mode with 25 MHz clock.)

CPU_CLK: The CPU operating clock. The clock frequency is as follows:
1. Equals to the PLL output clock frequency when the PLL is on.
2. Equals to the OSC3 (high-speed) oscillation circuit output clock frequency when the PLL is off.
3. However, it equals to the divided frequency when the CLG is set to generate the CPU operating clock by dividing the source clock.
4. When the CPU stops by the HALT or SLP instruction, this clock is also stopped.

This clock is almost in phase with the bus clock.

At initial reset, BCLKSEL is set to "0" (CPU_CLK).

A1X1MD: Area 1 access speed (D3) / BCLK select register (0x4813A)

Select a number of access cycles for area 1 in x2 speed mode.

Write "1": 2 cycles
Write "0": 4 cycles
Read: Valid

When x2 speed mode is set (#X2SPD pin = L) and A1X1MD = "1", area 1 is read/written in 2 cycles of the CPU system clock.
When A1X1MD = "0", area 1 is read/written in 4 cycles.
When x1 speed mode is set (#X2SPD pin = H), area 1 is always accessed in 2 cycles regardless of the A1X1MD value.
At cold start, A1X1MD is set to "0" (4 cycles). At hot start, A1X1MD retains its status before being initialized.

II-5 ITC (Interrupt Controller)

The C33 Core Block contains an interrupt controller, making it possible to control all interrupts generated by the internal peripheral circuits. This section explains the functions of this interrupt controller centering around the method for controlling maskable interrupts. For details about the various factors and conditions under which interrupts are generated, refer to the description of each peripheral circuit in this manual.

Outline of Interrupt Functions

Maskable Interrupts

The ITC can handle 39 kinds of maskable interrupts as shown in the table below.

Table 5.1 List of Maskable Interrupts

No.	HEX No.	Vector number (Hex address)	Interrupt system (Peripheral circuit)	Interrupt factor	IDMA Ch.	Priority
1	10	16(Base+40)	Port input interrupt 0	Edge (rising or falling) or level (High or Low)	1	High
2	11	17(Base+44)	Port input interrupt 1	Edge (rising or falling) or level (High or Low)	2	↑
3	12	18(Base+48)	Port input interrupt 2	Edge (rising or falling) or level (High or Low)	3	
4	13	19(Base+4C)	Port input interrupt 3	Edge (rising or falling) or level (High or Low)	4	
5	14	20(Base+50)	Key input interrupt 0	Rising or falling edge	–	
6	15	21(Base+54)	Key input interrupt 1	Rising or falling edge	–	
7	16	22(Base+58)	High-speed DMA Ch.0	High-speed DMA Ch.0, end of transfer	5	
8	17	23(Base+5C)	High-speed DMA Ch.1	High-speed DMA Ch.1, end of transfer	6	
9	18	24(Base+60)	High-speed DMA Ch.2	High-speed DMA Ch.2, end of transfer	–	
10	19	25(Base+64)	High-speed DMA Ch.3	High-speed DMA Ch.3, end of transfer	–	
11	1A	26(Base+68)	IDMA	Intelligent DMA, end of transfer	–	
–		27–29	reserved	–	–	
12	1E	30(Base+78)	16-bit programmable timer 0	Timer 0 comparison B	7	
13	1F	31(Base+7C)		Timer 0 comparison A	8	
–		32–33	reserved	–	–	
14	22	34(Base+88)	16-bit programmable timer 1	Timer 1 comparison B	9	
15	23	35(Base+8C)		Timer 1 comparison A	10	
–		36–37	reserved	–	–	
16	26	38(Base+98)	16-bit programmable timer 2	Timer 2 comparison B	11	
17	27	39(Base+9C)		Timer 2 comparison A	12	
–		40–41	reserved	–	–	
18	2A	42(Base+A8)	16-bit programmable timer 3	Timer 3 comparison B	13	
19	2B	43(Base+AC)		Timer 3 comparison A	14	
–		44–45	reserved	–	–	
20	2E	46(Base+B8)	16-bit programmable timer 4	Timer 4 comparison B	15	
21	2F	47(Base+BC)		Timer 4 comparison A	16	
–		48–49	reserved	–	–	
22	32	50(Base+C8)	16-bit programmable timer 5	Timer 5 comparison B	17	
23	33	51(Base+CC)		Timer 5 comparison A	18	
24	34	52(Base+D0)	8-bit programmable timer	Timer 0 underflow	19	
25	35	53(Base+D4)		Timer 1 underflow	20	
26	36	54(Base+D8)		Timer 2 underflow	21	
27	37	55(Base+DC)		Timer 3 underflow	22	
28	38	56(Base+E0)	Serial interface Ch.0	Receive error	–	
29	39	57(Base+E4)		Receive buffer full	23	
30	3A	58(Base+E8)		Transmit buffer empty	24	
–		59	reserved	–	–	
31	3C	60(Base+F0)	Serial interface Ch.1	Receive error	–	
32	3D	61(Base+F4)		Receive buffer full	25	
33	3E	62(Base+F8)		Transmit buffer empty	26	
–		63	reserved	–	–	
34	40	64(Base+100)	A/D converter	A/D converter, end of conversion	27	
35	41	65(Base+104)	Clock timer	Falling edge of 32 Hz, 8 Hz, 2 Hz or 1 Hz signal 1-minuet, 1-hour or specified time count up	–	
–		66–67	reserved	–	–	
36	44	68(Base+110)	Port input interrupt 4	Edge (rising or falling) or level (High or Low)	28	
37	45	69(Base+114)	Port input interrupt 5	Edge (rising or falling) or level (High or Low)	29	
38	46	70(Base+118)	Port input interrupt 6	Edge (rising or falling) or level (High or Low)	30	↓
39	47	71(Base+11C)	Port input interrupt 7	Edge (rising or falling) or level (High or Low)	31	Low

Contents of table

"Hex No." indicates an interrupt number in hexadecimal value.

"Vector number (Address)" indicates the trap table's vector number. The numerals in parentheses show an offset (in bytes) from the starting address (Base) of the trap table. The starting address (Base) of the trap table by default is the boot address, 0xC00000 set at an initial reset. This address can be changed using the TTBR register (0x48134 to 0x48137).

For details about the trap table contents including exception factors, etc., refer to the "E0C33000 Core CPU Manual".

"Interrupt system (Peripheral circuit)" indicates that interrupt levels can be programmed for each peripheral circuit written.

"Interrupt factor" indicates the factor of the interrupt occurring in each interrupt system.

"IDMA Ch." indicates that an interrupt factor which has a numeric value in this column can start up the intelligent DMA (IDMA) to transfer data when an interrupt factor occurs. The numeric value indicates the IDMA's channel number. Interrupt factors that do not have a numeric value here cannot start up the IDMA.

"Priority" indicates the priority of interrupts in cases when all interrupt systems are set to the same interrupt level. If two or more interrupt factors occur simultaneously, interrupt requests are accepted in order of highest priority. Interrupt priority varies depending on the interrupt levels set in each interrupt system. However, the priorities of interrupt factors in the same interrupt system are fixed in the order that they are written here.

Maskable interrupt generating conditions

A maskable interrupt to the CPU occurs when all of the conditions described below are met.

- The interrupt enable register for the interrupt factor that has occurred is set to "1".
- The IE (Interrupt Enable) bit of the Processor Status Register (PSR) in the CPU is set to "1".
- The interrupt factor that has occurred has a higher priority level than the value that is set in the PSR's Interrupt Level (IL). (The interrupt levels can be set using the interrupt priority register in each interrupt system.)
- No other trap factor having higher priority, such as NMI, has occurred.
- The interrupt factor does not invoke IDMA (the IDMA request bit is set to "0").

When an interrupt factor occurs, the corresponding interrupt factor flag is set to "1" and the flag remains set until it is reset in the software program. Therefore, in no cases can the generated interrupt factor be inadvertently cleared even if the above conditions are not met when the interrupt factor has occurred. The interrupt will occur when the above conditions are met.
However, when the interrupt factor invokes IDMA, the interrupt factor is reset if the following condition is met.

- The IDMA transfer counter is not "0".
- Interrupts are disabled in the IDMA control information even if the transfer counter is "0".

If two or more maskable interrupt factors occur simultaneously, the interrupt factor that has the highest priority is allowed to signal an interrupt request to the CPU. The other interrupts with lower priorities are kept pending until the above conditions are met.

The PSR and interrupt control register will be detailed later.
For details about interrupt factor generating conditions, refer to the description of each peripheral circuit in this manual.

Interrupt Factors and Intelligent DMA

Several interrupt factors can be set so that they can invoke IDMA startup. When one of these interrupt factors occurs, IDMA is started up before an interrupt request to the CPU. The interrupt request to the CPU is generated after IDMA is completed. (The interrupt request can be disabled by a program.)
IDMA is always started up regardless of how the PSR is set. For details, refer to "IDMA Invocation".

Nonmaskable Interrupt (NMI)

The nonmaskable interrupt (NMI) can be generated by pulling the #NMI pin low or using the internal watchdog timer. The vector number of NMI is 7, with the vector address set to the trap table's starting address + 28 bytes.
This interrupt is prioritized over other interrupts and is unconditionally accepted by the CPU.
However, since this interrupt may operate erratically if it occurs before the stack pointer (SP) is set up, it is masked in hardware until a write to the SP is completed after an initial reset.

Interrupt Processing by the CPU

The CPU keeps sampling interrupt requests every cycle. When the CPU accepts an interrupt request, it enters trap processing after completing execution of the instruction that was being executed.
The following lists the contents executed in trap processing.

(1) The PSR and the current program counter (PC) value are saved to the stack.
(2) The IE bit of the PSR is reset to "0" (following maskable interrupts are disabled).
(3) The IL of the PSR is set to the priority level of the accepted interrupt (NMI does not have its interrupt level changed).
(4) The vector of the generated interrupt factor is loaded into the PC, thus executing the interrupt processing routine.

Thus, once an interrupt is accepted, all maskable interrupts that may follow are disabled in (2). Multiple interrupts can also be handled by setting the IE bit to "1" in the interrupt processing routine. In this case, since the IL has been changed in (3), only an interrupt that has a higher priority than that of the currently processed interrupt is accepted.
When the interrupt processing routine is terminated by the reti instruction, the PSR is restored to its previous status before the interrupt has occurred. The program restarts processing after branching to the instruction next to the one that was being executed when the interrupt occurred.

Clearing Standby Mode by Interrupts

The standby modes (HALT and SLEEP) are cleared by an NMI or a maskable interrupt.
All maskable interrupts can be used to clear HALT mode. However, if the bus clock has stopped in HALT2 mode, a DMA interrupt cannot be used.
In SLEEP mode, since the high-speed (OSC3) oscillation circuit is deactivated, interrupts from the peripheral circuits that operate with the OSC3 clock cannot be used.

Interrupts that can be used to clear basic HALT mode:	NMI and all maskable interrupts
Interrupts that can be used to clear HALT2 mode:	NMI and all maskable interrupts (except DMA interrupts)
Interrupts that can be used to clear SLEEP mode:	NMI, input port interrupts, and clock timer interrupts

Clearing of the standby modes is accomplished by an interrupt request to the CPU. Therefore, this requires that the PSR be set in such a way that the requested interrupt will be accepted, and that the interrupt enable register for the interrupt factor be set to accept the interrupt.
When standby mode is cleared and the CPU has accepted the interrupt, it returns to the instruction next to the halt or slp instruction after executing the interrupt processing routine.

Note: If the interrupt factor used to restart from the standby mode has been set to invoke the IDMA, the IDMA is started up by that interrupt.
In the case of SLEEP mode, the high-speed (OSC3) oscillation circuit also starts operating.
If an interrupt to be generated upon completion of IDMA is disabled at the setting of the IDMA side, no interrupt request is signaled to the CPU. Therefore, the CPU remains idle until the next interrupt request is generated.

Trap Table

The C33 Core Block allows the base (starting) address of the trap table to be set by the TTBR register.
TTBR0 (D[9:0]) / TTBR low-order register (0x48134): Trap table base address [9:0] (fixed at "0")
TTBR1 (D[F:A]) / TTBR low-order register (0x48134): Trap table base address [15:10]
TTBR2 (D[B:0]) / TTBR high-order register (0x48136): Trap table base address [27:16]
TTBR3 (D[F:C]) / TTBR high-order register (0x48136): Trap table base address [31:28] (fixed at "0")

After an initial reset, the TTBR register is set to the boot address determined by the BTA3 pin status.
BTA3 = high: 0x0080000
BTA3 = low: 0x0C00000

Therefore, even when the trap table position is changed, it is necessary that at least the reset vector be written to the above address.

TTBR0 and TTBR3 are read-only bits which are fixed at "0". Therefore, the trap table starting address always begins with a 1KB boundary address.
The TTBR register is normally write-protected to prevent them from being inadvertently rewritten. To remove this write protection function, another register, TBRP (D[7:0]) / TTBR write-protect register (0x4812D [byte]), is provided. A write to the TTBR register is enabled by writing "0x59" to TBRP and is disabled back again by a write to the most significant byte of the TTBR register (0x48137). Consequently, a write to the TTBR register needs to begin with the low-order half-word first. However, since an occurrence of NMI or the like between writes of the low-order and high-order half-words would cause a malfunction, it is recommended that the register be written in words.

Control of Maskable Interrupts

Structure of the Interrupt Controller

The interrupt controller is configured as shown in Figure 5.1.

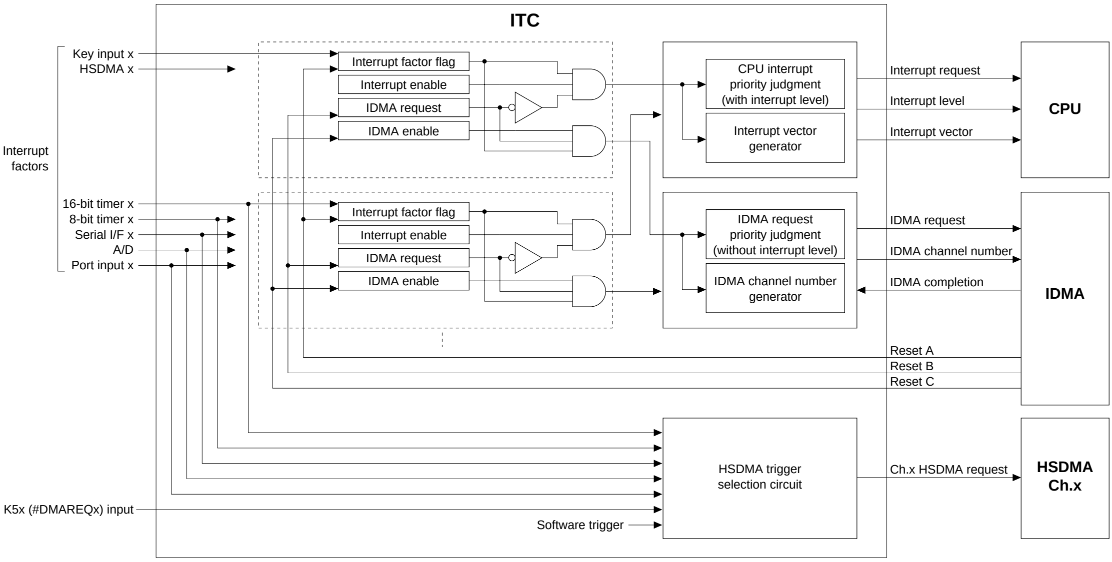


Figure 5.1 Configuration of Interrupt Controller

The following sections explain the functions of the registers used to control interrupts.

Processor Status Register (PSR)

The PSR is a special register incorporated in the core CPU and contains control bits to enable or disable an interrupt request to the CPU.

Interrupt Enable (IE) bit: PSR[4]

This bit is used to enable or disable an interrupt request to the CPU. When this bit is set to "1", the CPU is enabled to accept a maskable interrupt request. When this bit is reset to "0", no maskable interrupt request is accepted by the CPU.

When the CPU accepts an interrupt request (or some other trap occurs), it saves the PSR to the stack and resets the IE bit to "0". Consequently, no maskable interrupt request occurring thereafter will be accepted unless the IE bit is set to "1" in software program or the interrupt (trap) processing routine is terminated by the reti instruction.

The IE bit is initialized to "0" (interrupts disabled) by an initial reset.

Interrupt Level (IL): PSR[11:8]

The IL bits disable the interrupts whose priorities are below the set interrupt level. For example, if the interrupt level set in the IL is 3, the interrupts whose priorities are set below 3 in the interrupt priority register (described later) are not accepted by the CPU even if the IE bit is set to "1". The IL and the interrupt priority register together allow you to control the interrupt priorities in each interrupt system. For details about the interrupt levels, refer to "Interrupt Priority Register and Interrupt Levels".

When the CPU accepts a maskable interrupt request, it saves the PSR to the stack and sets the IL to the accepted interrupt's priority level. Therefore, even when the IE bit is set to "1" in the interrupt processing routine, no interrupts whose priority levels are equal or below that of the interrupt currently being processed are accepted unless the IL is rewritten.

The IL is restored to its previous status when the interrupt processing routine is terminated by the reti instruction.

The IL is rewritten for only maskable interrupts and not for any other traps (except a reset).
The IL is set to level 0 (that is, all interrupts above level 1 are enabled) by an initial reset.

Note: As the E0C33000 Core CPU function, the IL allows interrupt levels to be set in the range of 0 to 15. However, since the interrupt priority register in the ITC consists of three bits, interrupt levels in each interrupt system can only be set for up to 8.

Interrupt Factor Flag and Interrupt Enable Register

An interrupt factor flag and an interrupt enable register are provided for each maskable interrupt factor.

Interrupt factor flag

The interrupt factor flag is set to "1" when the corresponding interrupt factor occurs. Reading the flag enables you to determine what caused an interrupt, making it unnecessary to resort to the CPU's trap processing. The interrupt factor flag is reset only by writing data in software. Note that the method by which this flag is reset can be selected from the software application using either of the two methods described below. This selection is accomplished using RSTONLY (D0) / Interrupt factor flag reset method select register (0x4029F).

- Reset-only method (default)
  This method is selected (RSTONLY = "1") when initially reset.
  With this method, the interrupt factor flag is reset by writing "1". Although multiple interrupt factor flags are located at the same address of the interrupt control register, the interrupt factor flags for which "0" has been written can be neither set nor reset. Therefore, this method ensures that only a specific factor flag is reset. However, when using read-modify-write instructions (e.g., bset, bclr, or bnot), note that an interrupt factor flag that has been set to "1" is reset by writing.

  In this method, no interrupt factor flag can be set in the software application.

- Read/write method
  This method is selected by writing "0" to RSTONLY.
  When this method is used, interrupt factor flags can be read and written as for other registers. Therefore, the flag is reset by writing "0" and set by writing "1". In this case, all factor flags for which "0" has been written are reset. Even in a read-modify-write operation, an interrupt factor can occur between the read and the write, so be careful when using this method.

Since interrupt factor flags are not initialized by an initial reset, be sure to reset them before enabling interrupts.

Note: Even when a maskable interrupt request is accepted by the CPU and control branches off to the interrupt processing routine, the interrupt factor flag is not reset. Consequently, if control is returned from the interrupt processing routine by the reti instruction without resetting the interrupt factor flag in a program, the same interrupt factor occurs again.

For details about interrupt factor generating conditions, refer to the description of each peripheral circuit in this manual.

Interrupt enable register

This register controls the output of an interrupt request to the CPU. Only when the interrupt enable bit of this register is set to "1" can an interrupt request to the CPU be enabled by an occurrence of the corresponding interrupt factor. If the bit is set to "0", no interrupt request is made to the CPU even when the corresponding interrupt factor occurs.
Interrupt enable bits can be read and written as for other registers. Therefore, the interrupt enable bit is reset by writing "0" and set by writing "1". By reading this register, its setup status can be checked at any time.
Settings of the interrupt enable register do not affect the operation of interrupt factor flags, so when an interrupt factor occurs the interrupt factor flag is set to "1" even if the corresponding interrupt enable bit is set to "0".
When initially reset, the interrupt enable register is set to "0" (interrupts are disabled).

In cases when IDMA is started up by occurrence of an interrupt factor or when clearing standby mode (HALT or SLEEP mode) too, the corresponding interrupt enable bit must be set to "1".

The interrupt controller outputs an interrupt request to the CPU when the following conditions are met:
- An interrupt factor has occurred and the interrupt factor flag is set to "1".
- The bit of the interrupt enable register for the interrupt factor that has occurred is set to "1" (interrupt enable).
- The bit of the IDMA request register for the interrupt factor that has occurred is set to "0" (interrupt request).

If two or more interrupt factors occur simultaneously, the interrupt factor that has the highest priority is allowed to signal an interrupt request to the CPU. (See the following section.)

When these conditions are met, the interrupt controller outputs an interrupt request signal to the CPU along with the setup content (interrupt level) of the interrupt priority register for the generated interrupt system and its vector number.
These signals remain asserted until the interrupt factor flag is reset to "0" or the corresponding bit of the interrupt enable register is set to "0" (interrupts are disabled) or until some other interrupt factor of higher priority occurs. They are not cleared if the CPU simply accepts the interrupt request.

Interrupt Priority Register and Interrupt Levels

The interrupt priority register is a 3-bit register provided for each interrupt system. It allows the interrupt levels of a given interrupt system to be set in the range of 0 to 7. The default priorities shown in Table 5.1 can be modified according to system requirements by this setting.
The value set in this register is used by the interrupt controller and the CPU as described below.

Roles of the interrupt priority register in the interrupt controller

If two or more interrupt factors that have been enabled by the interrupt enable register occur simultaneously, the interrupt factor in the interrupt system whose interrupt priority register contains the greatest value is allowed by the interrupt controller to signal an interrupt request to the CPU.
If an interrupt factor occurs in two or more interrupt systems having the same value, the interrupt priority is resolved according to the default priorities in Table 5.1. Interrupt factors in the same interrupt system also have their priorities resolved according to the order in Table 5.1.
Other interrupt factors are kept pending until all interrupts of higher priority are accepted by the CPU.

When outputting an interrupt request signal to the CPU, the interrupt controller outputs the content of the interrupt priority register to the CPU along with it.
If another interrupt factor of higher priority occurs during outputting an interrupt request signal, the interrupt controller changes the vector number and interrupt level to those of the new interrupt factor before they are output to the CPU. The first interrupt request is left pending.

Roles of the interrupt priority register in CPU processing

The CPU compares the content of the interrupt priority register received from the interrupt controller with the interrupt level that is set in the IL of the PSR to determine whether or not to accept the interrupt request.

IE bit = "1" & IL < interrupt priority register: the interrupt request is accepted
IE bit = "1" & IL ≥ interrupt priority register: the interrupt request is rejected

Before interrupts can be controlled by an interrupt level, the interrupt disabling level must be written to the IL. For example, if the value written to the IL is 3, only the interrupts whose interrupt levels written in the interrupt priority register are 4 or more will be accepted.
When an interrupt is accepted, the interrupt level that is set in its interrupt priority register is written to the IL. As a result, the interrupt requests below that interrupt level can no longer be accepted.
If the interrupt priority register for an interrupt is set to "0", the interrupt is disabled. However, invoking IDMA by means of an interrupt factor works fine.

Notes:
- As the E0C33000 Core CPU function, the IL allows interrupt levels to be set in the range of 0 to 15. However, since the interrupt priority register in the C33 Core Block consists of three bits, interrupt levels in each interrupt system can only be set for up to 8.
- Multiple interrupts can also be handled by rewriting the interrupt level to the IL in the interrupt processing routine. However, if the interrupt level of the IL is set below the current level and the IE is set to enable interrupts before resetting the interrupt factor flag after an interrupt has occurred, the same interrupt may occur again.

IDMA Invocation

The interrupt factors for which IDMA channel numbers are written in Table 5.1 have the function to invoke the intelligent DMA (IDMA).

IDMA request register

The IDMA request register is used to specify the interrupt factor that invoke an IDMA transfer. If an IDMA request bit is set to "1", the IDMA request will be generated when the corresponding interrupt factor occurs. When the IDMA request bit is set to "0", the corresponding interrupt factor does not invoke IDMA and a normal interrupt processing will be performed. The IDMA request register is set to "0" by an initial reset.
The method by which this register is set can be selected from the software application using either of the two methods described below. This selection is accomplished using IDMAONLY (D1) / Flag set/reset method select register (0x4029F).

- Set-only method (default)
  This method is selected (IDMAONLY = "1") when initially reset.
  With this method, an IDMA request bit is set by writing "1". Although multiple IDMA request bits are located in the IDMA request register, the IDMA request bits for which "0" has been written can be neither set nor reset. Therefore, this method ensures that only a specific IDMA request bit is set.
  However, when using read-modify-write instructions (e.g., bset, bclr, or bnot), note that an IDMA request bit that has been set to "1" is not reset by writing.

- Read/write method
  This method is selected by writing "0" to IDMAONLY.
  When this method is used, IDMA request bits can be read and written as for other registers. Therefore, the IDMA request bit is reset by writing "0" and set by writing "1". In this case, all IDMA request bits for which "0" has been written are reset. Even in a read-modify-write operation, an IDMA request bit can be reset by the hardware between the read and the write, so be careful when using this method.

IDMA enable register

To perform IDMA transfer using an interrupt factor, the corresponding bit of the IDMA enable register must be set to "1". If this bit is set to "0", the interrupt factor cannot invoke the IDMA channel. The IDMA enable register is set to "0" by an initial reset.
The IDMA enable register allows selection of a set method (set-only method or Read/write method) similar to the IDMA request register. This selection is accomplished using DENONLY (D2) / Flag set/reset method select register (0x4029F). See the above explanation for the set method.

Invoking IDMA

Before IDMA can be invoked by the occurrence of an interrupt factor, the corresponding bits of the IDMA request and IDMA enable registers must be set to "1". Then when an interrupt factor occurs, the interrupt request to the CPU is made pending and the corresponding IDMA channel is invoked. The DMA transfer is performed according to the control information of that IDMA channel. The interrupt level set by the interrupt priority register of the ITC does not affect the IDMA invocation. The IDMA request can be accepted even if the interrupt level of the CPU is higher than the set value of the interrupt priority register. However, when generating the interrupt request to the CPU after the IDMA transfer is completed, the interrupt is controlled using the interrupt level set by the interrupt priority register.
An IDMA invocation request is accepted even when the interrupt enable register and PSR of the CPU is set to disable interrupts. It is also necessary that the control information for the IDMA channel has been set.

Interrupt after IDMA transfer

To generate an interrupt after completion of IDMA transfer:

The interrupt request that has been kept pending can be generated after completion of the DMA transfer.
In this case, the interrupt must be enabled by the IDMA control information (DINTEM = "1") in adition to the interrupt controller and the PSR register settings.
However, if the transfer counter set for the selected IDMA channel does not reach the terminal count of 0 after the number of transfers set have been performed, the interrupt factor flag is reset and no interrupt request is generated. The transfer counter is decremented by 1 for each transfer performed.
If the transfer counter is decremented to 0 when DINTEN is set to "1", the interrupt factor flag is not reset and the IDMA request bit is cleared to "0". An interrupt request is generated if other interrupt conditions are met.
The IDMA request bit must be set up again in order for IDMA to be invoked when an interrupt factor occurs next time as well. To ensure that no unwanted IDMA request occurs, this setup must be performed after resetting the interrupt factor flag.
Figure 5.2 shows the hardware sequence when DINTEN is set to "1".

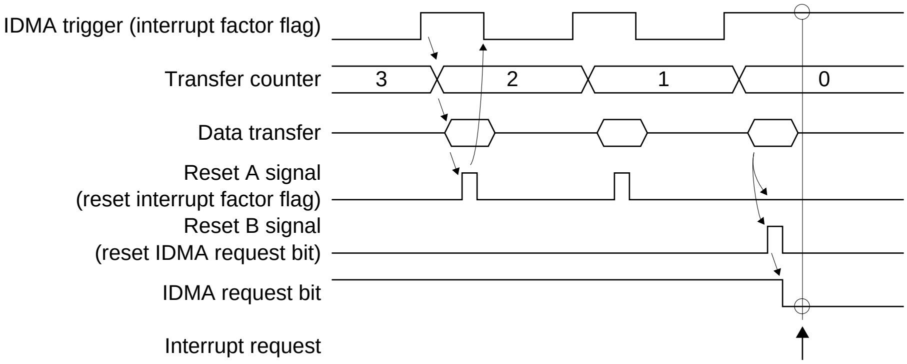


Figure 5.2 Sequence when DINTEN = "1"

To disable an interrupt after completion of IDMA transfer:

If an interrupt has been disabled in the IDMA control information (DINTEN = "0"), the interrupt is not generated since the interrupt factor flag is reset when the transfer counter becomes 0.
In this case, the IDMA request bit remains set to "1" without being cleared. However, the IDMA enable bit is cleared, so the following IDMA request by the same interrupt factor will be disabled.
Figure 5.3 shows the hardware sequence when DINTEN is set to "0".

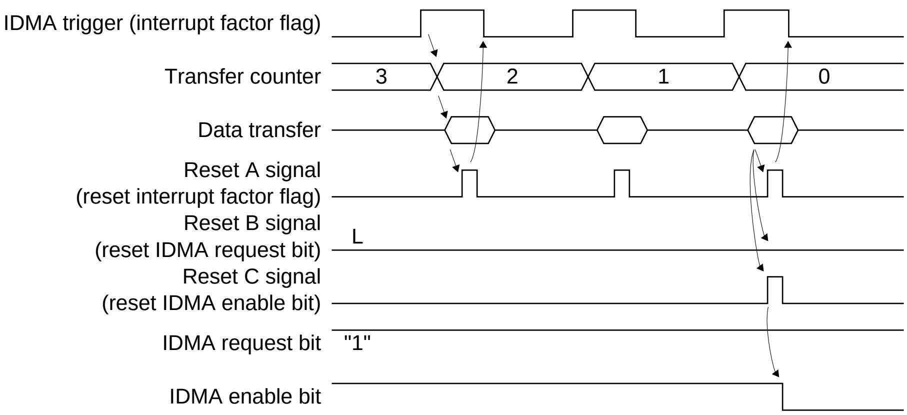


Figure 5.3 Sequence when DINTEN = "0"

For details on IDMA, refer to "IDMA (Intelligent DMA)".

HSDMA Invocation

Some interrupt factors can invoke high-speed DMAs (HSDMA).

HSDMA trigger set-up register

The DMA block contains four channel of HSDMA circuit. Each channel allows selection of an interrupt factor as the trigger. The HSDMA trigger set-up registers are used for this selection.

HSDMA Ch.0: HSD0S[3:0] (D[3:0])/HSDMA Ch.0/1 trigger set-up register (0x40298)
HSDMA Ch.1: HSD1S[3:0] (D[7:4])/HSDMA Ch.0/1 trigger set-up register (0x40298)
HSDMA Ch.2: HSD2S[3:0] (D[3:0])/HSDMA Ch.2/3 trigger set-up register (0x40299)
HSDMA Ch.3: HSD3S[3:0] (D[7:4])/HSDMA Ch.2/3 trigger set-up register (0x40299)

Table 5.2 shows the setting value and the corresponding trigger factor.

Table 5.2 HSDMA Trigger Factor

Value	Ch.0 trigger factor	Ch.1 trigger factor	Ch.2 trigger factor	Ch.3 trigger factor
0000	Software trigger	Software trigger	Software trigger	Software trigger
0001	K50 port input (falling edge)	K51 port input (falling edge)	K53 port input (falling edge)	K54 port input (falling edge)
0010	K50 port input (rising edge)	K51 port input (rising edge)	K53 port input (rising edge)	K54 port input (rising edge)
0011	Port 0 input	Port 1 input	Port 2 input	Port 3 input
0100	Port 4 input	Port 5 input	Port 6 input	Port 7 input
0101	8-bit timer 0 underflow	8-bit timer 1 underflow	8-bit timer 2 underflow	8-bit timer 3 underflow
0110	16-bit timer 0 compare B	16-bit timer 1 compare B	16-bit timer 2 compare B	16-bit timer 3 compare B
0111	16-bit timer 0 compare A	16-bit timer 1 compare A	16-bit timer 2 compare A	16-bit timer 3 compare A
1000	16-bit timer 4 compare B	16-bit timer 5 compare B	16-bit timer 4 compare B	16-bit timer 5 compare B
1001	16-bit timer 4 compare A	16-bit timer 5 compare A	16-bit timer 4 compare A	16-bit timer 5 compare A
1010	Serial I/F Ch.0 Rx buffer full	Serial I/F Ch.1 Rx buffer full	Serial I/F Ch.0 Rx buffer full	Serial I/F Ch.1 Rx buffer full
1011	Serial I/F Ch.0 Tx buffer empty	Serial I/F Ch.1 Tx buffer empty	Serial I/F Ch.0 Tx buffer empty	Serial I/F Ch.1 Tx buffer empty
1100	A/D conversion completion	A/D conversion completion	A/D conversion completion	A/D conversion completion

Invoking HSDMA

By selecting an interrupt factor with the HSDMA trigger set-up register, the HSDMA channel is invoked when the selected interrupt factor occurs. The interrupt control bits (interrupt factor flag, interrupt enable register, IDMA request register, interrupt priority register) do not affect this invocation.

Since HSDMA does not reset the interrupt factor flag, an interrupt will occur when the DMA transfer is completed if the interrupt is enabled by ITC.

Before HSDMA can be invoked by the occurrence of an interrupt factor, it is necessary that DMA be enabled on the HSDMA side by setting the control register for HSDMA transfer.

For details about HSDMA, refer to "HSDMA (High-Speed DMA)".

I/O Memory of Interrupt Controller

Table 5.3 shows the control bits of the interrupt controller.

Table 5.3 Control Bits of Interrupt Controller

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Port input 0/1 interrupt priority register	**0040260 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**PP1L2**	Port input 1 interrupt level	0 to 7	X	R/W	
		D5	**PP1L1**			X		
		D4	**PP1L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PP0L2**	Port input 0 interrupt level	0 to 7	X	R/W	
		D1	**PP0L1**			X		
		D0	**PP0L0**			X		
Port input 2/3 interrupt priority register	**0040261 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**PP3L2**	Port input 3 interrupt level	0 to 7	X	R/W	
		D5	**PP3L1**			X		
		D4	**PP3L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PP2L2**	Port input 2 interrupt level	0 to 7	X	R/W	
		D1	**PP2L1**			X		
		D0	**PP2L0**			X		
Key input interrupt priority register	**0040262 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**PK1L2**	Key input 1 interrupt level	0 to 7	X	R/W	
		D5	**PK1L1**			X		
		D4	**PK1L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PK0L2**	Key input 0 interrupt level	0 to 7	X	R/W	
		D1	**PK0L1**			X		
		D0	**PK0L0**			X		
High-speed DMA Ch.0/1 interrupt priority register	**0040263 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PHSD1L2**	High-speed DMA Ch.1 interrupt level	0 to 7	X	R/W	
		D5	**PHSD1L1**			X		
		D4	**PHSD1L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PHSD0L2**	High-speed DMA Ch.0 interrupt level	0 to 7	X	R/W	
		D1	**PHSD0L1**			X		
		D0	**PHSD0L0**			X		
High-speed DMA Ch.2/3 interrupt priority register	**0040264 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PHSD3L2**	High-speed DMA Ch.3 interrupt level	0 to 7	X	R/W	
		D5	**PHSD3L1**			X		
		D4	**PHSD3L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PHSD2L2**	High-speed DMA Ch.2 interrupt level	0 to 7	X	R/W	
		D1	**PHSD2L1**			X		
		D0	**PHSD2L0**			X		
IDMA interrupt priority register	**0040265 (B)**	D7–3	**–**	reserved	–	–	–	0 when being read.
		D2	**PDM2**	IDMA interrupt level	0 to 7	X	R/W	
		D1	**PDM1**			X		
		D0	**PDM0**			X		
16-bit timer 0/1 interrupt priority register	**0040266 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**P16T12**	16-bit timer 1 interrupt level	0 to 7	X	R/W	
		D5	**P16T11**			X		
		D4	**P16T10**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**P16T02**	16-bit timer 0 interrupt level	0 to 7	X	R/W	
		D1	**P16T01**			X		
		D0	**P16T00**			X		
16-bit timer 2/3 interrupt priority register	**0040267 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**P16T32**	16-bit timer 3 interrupt level	0 to 7	X	R/W	
		D5	**P16T31**			X		
		D4	**P16T30**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**P16T22**	16-bit timer 2 interrupt level	0 to 7	X	R/W	
		D1	**P16T21**			X		
		D0	**P16T20**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
16-bit timer 4/5	**0040268**	D7	–	reserved	–	–	–	0 when being read.
interrupt	**(B)**	D6	**P16T52**	16-bit timer 5 interrupt level	0 to 7	X	R/W	
priority register		D5	**P16T51**			X		
		D4	**P16T50**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**P16T42**	16-bit timer 4 interrupt level	0 to 7	X	R/W	
		D1	**P16T41**			X		
		D0	**P16T40**			X		
8-bit timer,	**0040269**	D7	–	reserved	–	–	–	0 when being read.
serial I/F Ch.0	**(B)**	D6	**PSI002**	Serial interface Ch.0	0 to 7	X	R/W	
interrupt		D5	**PSI001**	interrupt level		X		
priority register		D4	**PSI000**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**P8TM2**	8-bit timer 0–3 interrupt level	0 to 7	X	R/W	
		D1	**P8TM1**			X		
		D0	**P8TM0**			X		
Serial I/F Ch.1,	**004026A**	D7	–	reserved	–	–	–	0 when being read.
A/D interrupt	**(B)**	D6	**PAD2**	A/D converter interrupt level	0 to 7	X	R/W	
priority register		D5	**PAD1**			X		
		D4	**PAD0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PSI012**	Serial interface Ch.1	0 to 7	X	R/W	
		D1	**PSI011**	interrupt level		X		
		D0	**PSI010**			X		
Clock timer	**004026B**	D7–3	–	reserved	–	–	–	Writing 1 not allowed.
interrupt	**(B)**	D2	**PCTM2**	Clock timer interrupt level	0 to 7	X	R/W	
priority register		D1	**PCTM1**			X		
		D0	**PCTM0**			X		
Port input 4/5	**004026C**	D7	–	reserved	–	–	–	0 when being read.
interrupt	**(B)**	D6	**PP5L2**	Port input 5 interrupt level	0 to 7	X	R/W	
priority register		D5	**PP5L1**			X		
		D4	**PP5L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PP4L2**	Port input 4 interrupt level	0 to 7	X	R/W	
		D1	**PP4L1**			X		
		D0	**PP4L0**			X		
Port input 6/7	**004026D**	D7	–	reserved	–	–	–	0 when being read.
interrupt	**(B)**	D6	**PP7L2**	Port input 7 interrupt level	0 to 7	X	R/W	
priority register		D5	**PP7L1**			X		
		D4	**PP7L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PP6L2**	Port input 6 interrupt level	0 to 7	X	R/W	
		D1	**PP6L1**			X		
		D0	**PP6L0**			X		
Key input,	**0040270**	D7–6	–	reserved	–	–	–	0 when being read.
port input 0–3	**(B)**	D5	**EK1**	Key input 1	1 Enabled / 0 Disabled	0	R/W	
interrupt		D4	**EK0**	Key input 0		0	R/W	
enable register		D3	**EP3**	Port input 3		0	R/W	
		D2	**EP2**	Port input 2		0	R/W	
		D1	**EP1**	Port input 1		0	R/W	
		D0	**EP0**	Port input 0		0	R/W	
DMA interrupt	**0040271**	D7–5	–	reserved	–	–	–	0 when being read.
enable register	**(B)**	D4	**EIDMA**	IDMA	1 Enabled / 0 Disabled	0	R/W	
		D3	**EHDM3**	High-speed DMA Ch.3		0	R/W	
		D2	**EHDM2**	High-speed DMA Ch.2		0	R/W	
		D1	**EHDM1**	High-speed DMA Ch.1		0	R/W	
		D0	**EHDM0**	High-speed DMA Ch.0		0	R/W	
16-bit timer 0/1	**0040272**	D7	**E16TC1**	16-bit timer 1 comparison A	1 Enabled / 0 Disabled	0	R/W	
interrupt	**(B)**	D6	**E16TU1**	16-bit timer 1 comparison B		0	R/W	
enable register		D5–4	–	reserved	–	–	–	0 when being read.
		D3	**E16TC0**	16-bit timer 0 comparison A	1 Enabled / 0 Disabled	0	R/W	
		D2	**E16TU0**	16-bit timer 0 comparison B		0	R/W	
		D1–0	–	reserved	–	–	–	0 when being read.
16-bit timer 2/3	**0040273**	D7	**E16TC3**	16-bit timer 3 comparison A	1 Enabled / 0 Disabled	0	R/W	
interrupt	**(B)**	D6	**E16TU3**	16-bit timer 3 comparison B		0	R/W	
enable register		D5–4	–	reserved	–	–	–	0 when being read.
		D3	**E16TC2**	16-bit timer 2 comparison A	1 Enabled / 0 Disabled	0	R/W	
		D2	**E16TU2**	16-bit timer 2 comparison B		0	R/W	
		D1–0	–	reserved	–	–	–	0 when being read.

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 4/5 interrupt enable register	**0040274 (B)**	D7	**E16TC5**	16-bit timer 5 comparison A	1	Enabled	0	Disabled	0	R/W	
		D6	**E16TU5**	16-bit timer 5 comparison B					0	R/W	
		D5–4	**–**	reserved		–			–	–	0 when being read.
		D3	**E16TC4**	16-bit timer 4 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU4**	16-bit timer 4 comparison B					0	R/W	
		D1–0	**–**	reserved		–			–	–	0 when being read.
8-bit timer interrupt enable register	**0040275 (B)**	D7–4	**–**	reserved		–			–	–	0 when being read.
		D3	**E8TU3**	8-bit timer 3 underflow	1	Enabled	0	Disabled	0	R/W	
		D2	**E8TU2**	8-bit timer 2 underflow					0	R/W	
		D1	**E8TU1**	8-bit timer 1 underflow					0	R/W	
		D0	**E8TU0**	8-bit timer 0 underflow					0	R/W	
Serial I/F interrupt enable register	**0040276 (B)**	D7–6	**–**	reserved		–			–	–	0 when being read.
		D5	**ESTX1**	SIF Ch.1 transmit buffer empty	1	Enabled	0	Disabled	0	R/W	
		D4	**ESRX1**	SIF Ch.1 receive buffer full					0	R/W	
		D3	**ESERR1**	SIF Ch.1 receive error					0	R/W	
		D2	**ESTX0**	SIF Ch.0 transmit buffer empty					0	R/W	
		D1	**ESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D0	**ESERR0**	SIF Ch.0 receive error					0	R/W	
Port input 4–7, clock timer, A/D interrupt enable register	**0040277 (B)**	D7–6	**–**	reserved		–			–	–	0 when being read.
		D5	**EP7**	Port input 7	1	Enabled	0	Disabled	0	R/W	
		D4	**EP6**	Port input 6					0	R/W	
		D3	**EP5**	Port input 5					0	R/W	
		D2	**EP4**	Port input 4					0	R/W	
		D1	**ECTM**	Clock timer					0	R/W	
		D0	**EADE**	A/D converter					0	R/W	
Key input, port input 0–3 interrupt factor flag register	**0040280 (B)**	D7–6	**–**	reserved		–			–	–	0 when being read.
		D5	**FK1**	Key input 1	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FK0**	Key input 0					X	R/W	
		D3	**FP3**	Port input 3					X	R/W	
		D2	**FP2**	Port input 2					X	R/W	
		D1	**FP1**	Port input 1					X	R/W	
		D0	**FP0**	Port input 0					X	R/W	
DMA interrupt factor flag register	**0040281 (B)**	D7–5	**–**	reserved		–			–	–	0 when being read.
		D4	**FIDMA**	IDMA	1	Factor is generated	0	No factor is generated	X	R/W	
		D3	**FHDM3**	High-speed DMA Ch.3					X	R/W	
		D2	**FHDM2**	High-speed DMA Ch.2					X	R/W	
		D1	**FHDM1**	High-speed DMA Ch.1					X	R/W	
		D0	**FHDM0**	High-speed DMA Ch.0					X	R/W	
16-bit timer 0/1 interrupt factor flag register	**0040282 (B)**	D7	**F16TC1**	16-bit timer 1 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU1**	16-bit timer 1 comparison B					X	R/W	
		D5–4	**–**	reserved		–			–	–	0 when being read.
		D3	**F16TC0**	16-bit timer 0 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU0**	16-bit timer 0 comparison B					X	R/W	
		D1–0	**–**	reserved		–			–	–	0 when being read.
16-bit timer 2/3 interrupt factor flag register	**0040283 (B)**	D7	**F16TC3**	16-bit timer 3 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU3**	16-bit timer 3 comparison B					X	R/W	
		D5–4	**–**	reserved		–			–	–	0 when being read.
		D3	**F16TC2**	16-bit timer 2 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU2**	16-bit timer 2 comparison B					X	R/W	
		D1–0	**–**	reserved		–			–	–	0 when being read.
16-bit timer 4/5 interrupt factor flag register	**0040284 (B)**	D7	**F16TC5**	16-bit timer 5 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU5**	16-bit timer 5 comparison B					X	R/W	
		D5–4	**–**	reserved		–			–	–	0 when being read.
		D3	**F16TC4**	16-bit timer 4 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU4**	16-bit timer 4 comparison B					X	R/W	
		D1–0	**–**	reserved		–			–	–	0 when being read.
8-bit timer interrupt factor flag register	**0040285 (B)**	D7–4	**–**	reserved		–			–	–	0 when being read.
		D3	**F8TU3**	8-bit timer 3 underflow	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F8TU2**	8-bit timer 2 underflow					X	R/W	
		D1	**F8TU1**	8-bit timer 1 underflow					X	R/W	
		D0	**F8TU0**	8-bit timer 0 underflow					X	R/W	
Serial I/F interrupt factor flag register	**0040286 (B)**	D7–6	**–**	reserved		–			–	–	0 when being read.
		D5	**FSTX1**	SIF Ch.1 transmit buffer empty	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FSRX1**	SIF Ch.1 receive buffer full					X	R/W	
		D3	**FSERR1**	SIF Ch.1 receive error					X	R/W	
		D2	**FSTX0**	SIF Ch.0 transmit buffer empty					X	R/W	
		D1	**FSRX0**	SIF Ch.0 receive buffer full					X	R/W	
		D0	**FSERR0**	SIF Ch.0 receive error					X	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Port input 4–7, clock timer, A/D interrupt factor flag register	**0040287 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**FP7**	Port input 7	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FP6**	Port input 6					X	R/W	
		D3	**FP5**	Port input 5					X	R/W	
		D2	**FP4**	Port input 4					X	R/W	
		D1	**FCTM**	Clock timer					X	R/W	
		D0	**FADE**	A/D converter					X	R/W	
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA request register	**0040290 (B)**	D7	**R16TC0**	16-bit timer 0 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**RHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**RHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**RP3**	Port input 3					0	R/W	
		D2	**RP2**	Port input 2					0	R/W	
		D1	**RP1**	Port input 1					0	R/W	
		D0	**RP0**	Port input 0					0	R/W	
16-bit timer 1–4 IDMA request register	**0040291 (B)**	D7	**R16TC4**	16-bit timer 4 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU4**	16-bit timer 4 comparison B					0	R/W	
		D5	**R16TC3**	16-bit timer 3 comparison A					0	R/W	
		D4	**R16TU3**	16-bit timer 3 comparison B					0	R/W	
		D3	**R16TC2**	16-bit timer 2 comparison A					0	R/W	
		D2	**R16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1	**R16TC1**	16-bit timer 1 comparison A					0	R/W	
		D0	**R16TU1**	16-bit timer 1 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register	**0040292 (B)**	D7	**RSTX0**	SIF Ch.0 transmit buffer empty	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RSRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**R8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**R8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**R8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**R8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**R16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**R16TU5**	16-bit timer 5 comparison B					0	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA request register	**0040293 (B)**	D7	**RP7**	Port input 7	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RP6**	Port input 6					0	R/W	
		D5	**RP5**	Port input 5					0	R/W	
		D4	**RP4**	Port input 4					0	R/W	
		D3	–	reserved	–				–	–	0 when being read.
		D2	**RADE**	A/D converter	1	IDMA request	0	Interrupt request	0	R/W	
		D1	**RSTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**RSRX1**	SIF Ch.1 receive buffer full					0	R/W	
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA enable register	**0040294 (B)**	D7	**DE16TC0**	16-bit timer 0 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**DEHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**DEHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**DEP3**	Port input 3					0	R/W	
		D2	**DEP2**	Port input 2					0	R/W	
		D1	**DEP1**	Port input 1					0	R/W	
		D0	**DEP0**	Port input 0					0	R/W	
16-bit timer 1–4 IDMA enable register	**0040295 (B)**	D7	**DE16TC4**	16-bit timer 4 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU4**	16-bit timer 4 comparison B					0	R/W	
		D5	**DE16TC3**	16-bit timer 3 comparison A					0	R/W	
		D4	**DE16TU3**	16-bit timer 3 comparison B					0	R/W	
		D3	**DE16TC2**	16-bit timer 2 comparison A					0	R/W	
		D2	**DE16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1	**DE16TC1**	16-bit timer 1 comparison A					0	R/W	
		D0	**DE16TU1**	16-bit timer 1 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register	**0040296 (B)**	D7	**DESTX0**	SIF Ch.0 transmit buffer empty	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**DE8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**DE8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**DE8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**DE8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**DE16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**DE16TU5**	16-bit timer 5 comparison B					0	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA enable register	**0040297 (B)**	D7	**DEP7**	Port input 7	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DEP6**	Port input 6					0	R/W	
		D5	**DEP5**	Port input 5					0	R/W	
		D4	**DEP4**	Port input 4					0	R/W	
		D3	–	reserved	–				–	–	0 when being read.
		D2	**DEADE**	A/D converter	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D1	**DESTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**DESRX1**	SIF Ch.1 receive buffer full					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
High-speed	**0040298**	D7	**HSD1S3**	High-speed DMA Ch.1	0	Software trigger			0	R/W	
DMA Ch.0/1	**(B)**	D6	**HSD1S2**	trigger set-up	1	K51 input (falling edge)			0		
trigger set-up		D5	**HSD1S1**		2	K51 input (rising edge)			0		
register		D4	**HSD1S0**		3	Port 1 input			0		
					4	Port 5 input					
					5	8-bit timer Ch.1 underflow					
					6	16-bit timer Ch.1 compare B					
					7	16-bit timer Ch.1 compare A					
					8	16-bit timer Ch.5 compare B					
					9	16-bit timer Ch.5 compare A					
					A	SI/F Ch.1 Rx buffer full					
					B	SI/F Ch.1 Tx buffer empty					
					C	A/D conversion completion					
		D3	**HSD0S3**	High-speed DMA Ch.0	0	Software trigger			0	R/W	
		D2	**HSD0S2**	trigger set-up	1	K50 input (falling edge)			0		
		D1	**HSD0S1**		2	K50 input (rising edge)			0		
		D0	**HSD0S0**		3	Port 0 input			0		
					4	Port 4 input					
					5	8-bit timer Ch.0 underflow					
					6	16-bit timer Ch.0 compare B					
					7	16-bit timer Ch.0 compare A					
					8	16-bit timer Ch.4 compare B					
					9	16-bit timer Ch.4 compare A					
					A	SI/F Ch.0 Rx buffer full					
					B	SI/F Ch.0 Tx buffer empty					
					C	A/D conversion completion					
High-speed	**0040299**	D7	**HSD3S3**	High-speed DMA Ch.3	0	Software trigger			0	R/W	
DMA Ch.2/3	**(B)**	D6	**HSD3S2**	trigger set-up	1	K54 input (falling edge)			0		
trigger set-up		D5	**HSD3S1**		2	K54 input (rising edge)			0		
register		D4	**HSD3S0**		3	Port 3 input			0		
					4	Port 7 input					
					5	8-bit timer Ch.3 underflow					
					6	16-bit timer Ch.3 compare B					
					7	16-bit timer Ch.3 compare A					
					8	16-bit timer Ch.5 compare B					
					9	16-bit timer Ch.5 compare A					
					A	SI/F Ch.1 Rx buffer full					
					B	SI/F Ch.1 Tx buffer empty					
					C	A/D conversion completion					
		D3	**HSD2S3**	High-speed DMA Ch.2	0	Software trigger			0	R/W	
		D2	**HSD2S2**	trigger set-up	1	K53 input (falling edge)			0		
		D1	**HSD2S1**		2	K53 input (rising edge)			0		
		D0	**HSD2S0**		3	Port 2 input			0		
					4	Port 6 input					
					5	8-bit timer Ch.2 underflow					
					6	16-bit timer Ch.2 compare B					
					7	16-bit timer Ch.2 compare A					
					8	16-bit timer Ch.4 compare B					
					9	16-bit timer Ch.4 compare A					
					A	SI/F Ch.0 Rx buffer full					
					B	SI/F Ch.0 Tx buffer empty					
					C	A/D conversion completion					
Flag set/reset	**004029F**	D7–3	–	reserved	–				–	–	
method select	**(B)**	D2	**DENONLY**	IDMA enable register set method selection	1	Set only	0	RD/WR	1	R/W	
register		D1	**IDMAONLY**	IDMA request register set method selection	1	Set only	0	RD/WR	1	R/W	
		D0	**RSTONLY**	Interrupt factor flag reset method selection	1	Reset only	0	RD/WR	1	R/W	

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
TTBR write protect register	**004812D (B)**	D7	**TBRP7**	TTBR register write protect	Writing 01011001(0x59) removes the TTBR (0x48134) write protection. Writing other data sets the write protection.	0	W	Undefined in read.
		D6	**TBRP6**			0		
		D5	**TBRP5**			0		
		D4	**TBRP4**			0		
		D3	**TBRP3**			0		
		D2	**TBRP2**			0		
		D1	**TBRP1**			0		
		D0	**TBRP0**			0		
TTBR low-order register	**0048134 (HW)**	DF	**TTBR15**	Trap table base address [15:10]		0	R/W	
		DE	**TTBR14**			0		
		DD	**TTBR13**			0		
		DC	**TTBR12**			0		
		DB	**TTBR11**			0		
		DA	**TTBR10**			0		
		D9	**TTBR09**	Trap table base address [9:0]	Fixed at 0	0	R	0 when being read. Writing 1 not allowed.
		D8	**TTBR08**			0		
		D7	**TTBR07**			0		
		D6	**TTBR06**			0		
		D5	**TTBR05**			0		
		D4	**TTBR04**			0		
		D3	**TTBR03**			0		
		D2	**TTBR02**			0		
		D1	**TTBR01**			0		
		D0	**TTBR00**			0		
TTBR high-order register	**0048136 (HW)**	DF	**TTBR33**	Trap table base address [31:28]	Fixed at 0	0	R	0 when being read. Writing 1 not allowed.
		DE	**TTBR32**			0		
		DD	**TTBR31**			0		
		DC	**TTBR30**			0		
		DB	**TTBR2B**	Trap table base address [27:16]	The initial value is set according to the BTA3 pin status. BTA3 = "1": 0x008 BTA3 = "0": 0x0C0	←	R/W	
		DA	**TTBR2A**					
		D9	**TTBR29**					
		D8	**TTBR28**					
		D7	**TTBR27**					
		D6	**TTBR26**					
		D5	**TTBR25**					
		D4	**TTBR24**					
		D3	**TTBR23**					
		D2	**TTBR22**					
		D1	**TTBR21**					
		D0	**TTBR20**					

The following collectively explains the basic functions of each control register/bit. For details about individual interrupt systems and the contents classified by an interrupt factor, refer to the descriptions of the peripheral circuits in this manual.

Pxxx2–Pxxx0: Interrupt priority register

Set the priority levels of each interrupt system in the range of 0 to 7.
If this register is set below the IL value of the PSR, no interrupt is generated. The value of this register when initially reset is indeterminate.

Exxx: Interrupt enable register

Enable or disable interrupt generation to the CPU.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

Interrupts are enabled when the corresponding bits of this register are set to "1" and are disabled when the bits are set to "0".
For the interrupt factors used to request IDMA invocation or clear the standby mode, the corresponding interrupt enable register bit must be set for interrupt enable.
When initially reset, this register is set to "0" (interrupt disabled).

Fxxx: Interrupt factor flag

Indicate the status of interrupt factors generated.

When read

Read "1": Interrupt factor generated
Read "0": No interrupt factor generated

When written using the reset-only method (default)

Write "1": Factor flag is reset
Write "0": Invalid

When written using the read/write method

Write "1": Factor flag is set
Write "0": Factor flag is reset

The interrupt factor flag is set to "1" when an interrupt factor occurs in each peripheral circuit.
If the following conditions are met at this time, an interrupt is generated to the CPU:
1. The corresponding bit of the interrupt enable register is set to "1".
2. No other interrupt request of higher priority has occurred.
3. The IE bit of the PSR is set to "1" (interrupt enabled).
4. The corresponding interrupt priority register is set to a level higher than the CPU's interrupt level (IL).
When using an interrupt factor to request IDMA, note that even when the above conditions are met, no interrupt request to the CPU is generated for the interrupt factor that has occurred. If interrupts are enabled at the setting of IDMA, an interrupt is generated under the above conditions after the data transfer by IDMA is completed.
The interrupt factor flag is always set to "1" when an interrupt factor occurs no matter how the interrupt enable and interrupt priority registers are set.
In order for the next interrupt to be accepted after interrupt generation, the interrupt factor flag must be reset and the PSR must be set up again (by setting the IL below the level indicated by the interrupt priority register and setting the IE bit to "1" or executing the reti instruction).
The interrupt factor flag can only be reset by a write instruction in the software application. If the PSR is again set up to accept interrupts (or the reti instruction is executed) without resetting the interrupt factor flag, the same interrupt may occur again. Note also that the value to be written to reset the flag is "1" when using the reset-only method (RSTONLY = "1") and "0" when using the read/write method (RSTONLY = "0"). Be careful not to confuse these two conditions.
The interrupt factor flag becomes indeterminate when initially reset, so be sure to reset the flag in the software application.

Rxxx: IDMA request register

Specify whether or not to invoke IDMA when an interrupt factor occurs.

When using the set-only method (default)

Write "1": IDMA request
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA request
Write "0": Interrupt request
Read: Valid

If a bit of this register is set to "1", IDMA is invoked when the corresponding interrupt factor occurs and the programmed data transfer is performed. If the register bit is set to "0", regular interrupt processing is performed, without ever invoking IDMA.
For details about IDMA, refer to "IDMA (Intelligent DMA)".
If interrupts are enabled on the IDMA side and the transfer counter reaches the terminal count of 0 after completion of DMA transfer, the IDMA request register is reset to "0" and an interrupt request for the interrupt factor that enabled IDMA invoking is generated.
After an initial reset, this register is set to "0" (Interrupt is requested).

DExxx: IDMA enable register

Enable or disable the IDMA request.

When using the set-only method (default)

Write "1": IDMA enabled
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA enabled
Write "0": IDMA disabled
Read: Valid

If a bit of this register is set to "1", the IDMA request by the interrupt factor is enabled. If the register bit is set to "0", the IDMA request is disabled.
After an initial reset, this register is set to "0" (IDMA is disabled).

RSTONLY: Interrupt factor flag reset method selection (D0) / Flag set/reset method select register (0x4029F)

Select the method for resetting the interrupt factor flag.

Write "1": Reset-only method
Write "0": Read/write method
Read: Valid

With the reset-only method, the interrupt factor flag is reset by writing "1".
The interrupt factor flags for which "0" has been written can neither be set nor reset. Therefore, this method ensures that only a specific factor flag is reset. However, when using read-modify-write instructions (e.g., bset, bclr, or bnot), note that an interrupt factor flag that has been set to "1" is reset by writing. This method cannot be used to set any interrupt factor flag in the software application.
The read/write method is selected by writing "0" to RSTONLY. When this method is selected, interrupt factor flags can be read and written as for other registers. Therefore, the flag is reset by writing "0" and set by writing "1". In this case all factor flags for which "0" has been written are reset. Even in a read-modify-write operation, an interrupt factor can occur between read and write instructions, so be careful when using this method.
After an initial reset, RSTONLY is set to "1" (reset-only method).

IDMAONLY: IDMA request register set method selection (D1) / Flag set/reset method select register (0x4029F)

Select the method for setting the IDMA request registers.

Write "1": Set-only method
Write "0": Read/write method
Read: Valid

With the set-only method, IDMA request bits are set by writing "1".
The IDMA request bits for which "0" has been written can neither be set nor reset. Therefore, this method ensures that only a specific IDMA request bit is set. However, when using read-modify-write instructions (e.g., bset, bclr, or bnot), note that an IDMA request bit that has been set to "1" is not reset by writing.
The read/write method is selected by writing "0" to IDMAONLY. When this method is selected, IDMA request bits can be read and written as for other registers. Therefore, the IDMA request bit is reset by writing "0" and set by writing "1". In this case all IDMA request bits for which "0" has been written are reset. Even in a read-modify-write operation, an IDMA request bit can be reset by the hardware between the read and the write, so be careful when using this method.
After an initial reset, IDMAONLY is set to "1" (set-only method).

DENONLY: IDMA enable register set method selection
(D2) / Flag set/reset method select register (0x4029F)

Select the method for setting the IDMA enable registers.

Write "1": Set-only method
Write "0": Read/write method
Read: Valid

With the set-only method, IDMA enable bits are set by writing "1".
The IDMA enable bits for which "0" has been written can neither be set nor reset. Therefore, this method ensures that only a specific IDMA enable bit is set. However, when using read-modify-write instructions (e.g., bset, bclr, or bnot), note that an IDMA enable bit that has been set to "1" is not reset by writing.
The read/write method is selected by writing "0" to DENONLY. When this method is selected, IDMA enable bits can be read and written as for other registers. Therefore, the IDMA enable bit is reset by writing "0" and set by writing "1". In this case all IDMA enable bits for which "0" has been written are reset. Even in a read-modify-write operation, an interrupt enable bit can be reset by the hardware between the read and the write, so be careful when using this method.
After an initial reset, DENONLY is set to "1" (set-only method).

TBRP7–TBRP0: TTBR register write protection ([D[7:0]) / TTBR write-protect register (0x4812D)

Remove write protection for the TTBR register.

Write 0x59: Write protection is removed
Write not the above: No operation (write protected)
Read: Valid

Before writing to the TTBR register, set TBRP to "0x59" to remove the write protection. Then when data is written to the most significant byte (0x48137) of the TTBR, the register once again becomes write-protected.
After an initial reset, TBRP is set to "0x0" (write protected).

TTBR09–TTBR00: Trap table base address [9:0] (D[9:0]) / TTBR low-order register (0x48134[HW])
TTBR15–TTBR10: Trap table base address [15:10] (D[F:A]) / TTBR low-order register (0x48134[HW])
TTBR2B–TTBR20: Trap table base address [27:16] (D[B:0]) / TTBR high-order register (0x48136[HW])
TTBR33–TTBR30: Trap table base address [31:28] (D[F:C]) / TTBR high-order register (0x48136[HW])

Set the starting address of the trap table.
TTBR0 and TTBR3 are read-only registers and are fixed to "0". For this reason, the trap table starting address always begins with a 1KB boundary address.
The TTBR registers normally are write-protected to prevent them from being inadvertently rewritten. To remove this write protect function, another register, TBRP (D[F:8]) / TTBR write-protect register (0x4812D), is provided.
A write to the TTBR register is enabled by writing "0x59" to TBRP and is disabled back again by a write to the most significant byte of the TTBR register (0x48137). Consequently, writes to the TTBR register need to begin with the low-order half-word first. However, since occurrences of NMI and the like between writes of the low-order and high-order half-words cause malfunctions, it is recommended that the register be written in words.
After an initial reset, the TTBR register is set to the boot address that is determined by the BTA3 pin status (BTA3 = high: 0x0080000; BTA3 = low: 0x0C00000).

Programming Notes

(1) In cases when an interrupt factor that is used for restarting from the standby mode has been set to invoke IDMA, IDMA is started up by the interrupt at its occurrence. In SLEEP mode, the high-speed (OSC3) oscillation circuit also starts operating. However, if an interrupt to be generated upon completion of IDMA is disabled at the setting of IDMA side, no interrupt request is signaled to the CPU. Therefore, the CPU remains idle until the next interrupt request is generated.

(2) As the E0C33000 Core CPU function, the IL allows interrupt levels to be set in the range of 0 to 15. However, since the interrupt priority register in the C33 Core Block consists of three bits, interrupt levels in each interrupt system can only be set for up to 8.

(3) When the reset-only method is used to reset the interrupt factor flag (by writing "1"), if a read-modify-write instruction (e.g., bset, bclr, or bnot) is executed, the other interrupt factor flags at the same address that have been set to "1" are reset by a write. This requires caution. In cases when the read/write method is used to reset the interrupt factor flag (by writing "0"), all factor flags for which "0" has been written are reset. When a read-modify-write operation is performed, an interrupt factor may occur between reads and writes, so be careful when using this method.
The same applies to the set-only method and read/write method for the IDMA request and IDMA enable registers.

(4) After an initial reset, the interrupt factor flags and interrupt priority registers all become indeterminate. To prevent unwanted interrupts or IDMA requests from being generated inadvertently, be sure to reset these flags and registers in the software application.

(5) To prevent another interrupt from being generated for the same factor again after generation of an interrupt, be sure to reset the interrupt factor flag before enabling interrupts and setting the PSR again or executing the reti instruction.

THIS PAGE IS BLANK.

II-6 CLG (Clock Generator)

This section describes the method for controlling the system clock.

Configuration of Clock Generator

The C33 Core Block has a built-in clock generator that consists of a high-speed oscillation circuit (OSC3) and a PLL. The high-speed (OSC3) oscillation circuit generates the main clock for the CPU and internal peripheral circuits (e.g., DMA, serial interface, programmable timer, and A/D converter).
Furthermore, the clock generator can input a sub clock, such as low-speed (OSC1, 32.768 kHz, Typ.) clock generated by the Peripheral Block, for the clock timer and for operating the CPU at a low clock speed in order to reduce current consumption.

Note: When the Peripheral Block including the low-speed (OSC1) oscillation circuit is used, the source clocks for the CPU and the peripheral circuits (e.g., DMA, serial interface, programmable timer, and A/D converter) can be selected between the OSC3 clock and the OSC1 clock independently. For details, refer to "Setting and Switching Over the CPU Operating Clock" in this section and "Prescaler" and "Low-Speed (OSC1) oscillation circuit" of the Peripheral Block.

Figure 6.1 shows the configuration of the clock generator.

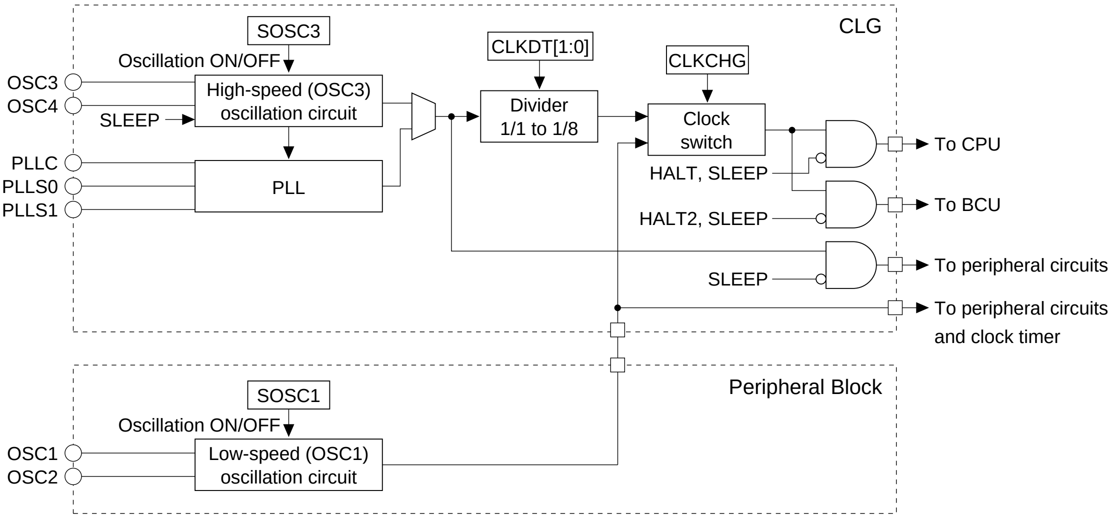


Figure 6.1 Configuration of Clock Generator

After an initial reset, the output (OSC3 clock) of the high-speed (OSC3) oscillation circuit is set for the CPU operating clock.
When the low-speed (OSC1) oscillation circuit is used, the CPU operating clock can be switched to the output (OSC1 clock) of the low-speed (OSC1) oscillation circuit in a program. Furthermore, each oscillation circuit can be stopped in a program.
If the OSC3 clock is unnecessary such as when performing clock processing only, set the OSC1 clock for operation of the CPU and turn off the high-speed (OSC3) oscillation circuit in order to reduce current consumption. In addition, when SLEEP mode is set, the high-speed (OSC3) oscillation circuit is turned off, greatly reducing current consumption (no internal units except for the clock timer need to be operated).

I/O Pins of Clock Generator

Table 6.1 lists the I/O pins of the clock generator.

Table 6.1 I/O Pins of Clock Generator

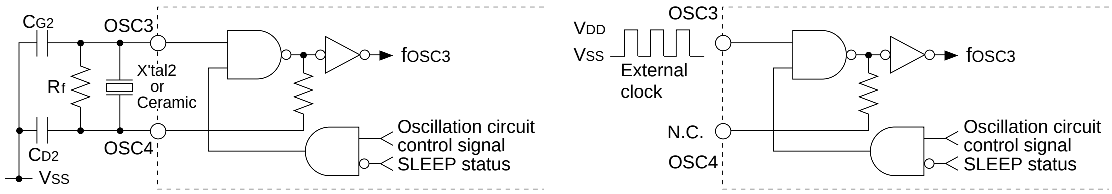
Pin name	I/O	Function
OSC3	I	High-speed (OSC3) oscillation input pin Crystal/ceramic oscillation or external clock input
OSC4	O	High-speed (OSC3) oscillation output pin Crystal/ceramic oscillation (open when external clock is used)
PLLC	–	Capasitor connecting pin for PLL
PLLS[1:0]	I	PLL set-up pins

PLLS1	PLLS0	fin (f_{OSC3})	fout (f_{PSCIN})	
1	1	10–30MHz	20–60MHz	∗1
		10–25MHz	20–50MHz	∗2
0	1	10–15MHz	40–60MHz	∗1
		10–12.5MHz	40–50MHz	∗2
0	0	PLL is not used	L	∗3

∗1: ROM-less model with 3.3 V ±0.3 V operating voltage
∗2: ROM built-in model, or 3.0 V ±0.3 V operating voltage
∗3: When the PLL is not used, the OSC3 clock is used directly.

High-Speed (OSC3) Oscillation Circuit

The high-speed (OSC3) oscillation circuit generates the main clock for the CPU and internal peripheral circuits (e.g., DMA, serial interface, programmable timer, and A/D converter).
This circuit can be a crystal or a ceramic oscillation circuit. Optionally an external clock source can be used.
Figure 6.2 shows the structure of the high-speed (OSC3) oscillation circuit.



(1) Crystal/ceramic oscillation circuit (2) External clock input

Figure 6.2 High-Speed (OSC3) Oscillation Circuit

When using a crystal or a ceramic oscillation for this circuit, connect a crystal (X'tal2) or ceramic (Ceramic) resonator and feedback resistor (Rf) between the OSC3 and OSC4 pins, and two capacitors (C_{G2}, C_{D2}) between the OSC3 pin and V_{SS} and the OSC4 pin and V_{SS}, respectively.
When an external clock is used, leave the OSC4 pin open and input a square-wave clock to the OSC3 pin.
The range of oscillation frequencies is 10 MHz to 33 MHz. This frequency range also applies when an external clock is used.

Note: When using the PLL, the oscillation frequency range changes according to the PLL setting. See Table 6.2.

For details on oscillation characteristics and the external clock input characteristics, refer to "Electrical Characteristics".

PLL

The PLL inputs the OSC3 clock and multiply its frequency. The multiply mode should be set using the PLLS[1:0] pins according to the OSC3 clock frequency.

Table 6.2 Setting the PLLS[1:0] Pins

PLLS1	PLLS0	Mode	fin (OSC3 clock)	fout
1	1	x2	10 to 20 MHz	20 to 40 MHz
1	0	x2	20 to 25 MHz	40 to 50 MHz
0	1	x4	10 to 12.5 MHz	40 to 50 MHz
0	0	PLL disabled	–	Unused

Figure 6.3 shows a basic external connection diagram for the PLL pins.

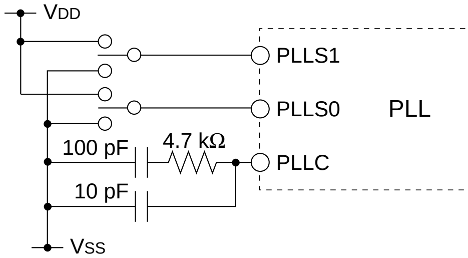


Figure 6.3 External Connection Diagram

Note: When the PLL is not used, the OSC3 oscillation output is used as the source clock. In this case, the oscillation frequency range is 10 MHz to 33 MHz. Furthermore, leave the PLLC pin open.

Controlling Oscillation

The high-speed (OSC3) oscillation circuit can be turned on or off using SOSC3 (D1) / Power control register (0x40180).
The oscillation circuit is turned off by writing "0" to SOSC3 and turned back on again by writing "1". SOSC3 is set to "1" at initial reset, so the oscillation circuit is turned on.

Notes:
- When the high-speed (OSC3) oscillation circuit is used as the clock source for the CPU operating clock, it cannot be turned off. In this case, writing "0" to SOSC3 is ignored. Note also that writing to SOSC3 is allowed only when the power-control register protection flag is set to "0b10010110".
- Immediately after the oscillation circuit is turned on, a certain period of time is required for oscillation to stabilize (for 3.3-V crystal resonator, this time is 10 ms max.). To prevent the device from operating erratically, do not use the clock until its oscillation has stabilized.

The high-speed (OSC3) oscillation circuit turns off when the CPU is set in SLEEP mode.

Setting and Switching Over the CPU Operating Clock

Setting the CPU operating clock frequency

When operating the CPU with the high-speed (OSC3) clock, the operating frequency can be switched over in four steps. Use CLKDT[1:0] (D[7:6]) / Power control register (0x40180) for this switchover.

Table 6.3 Setting of CPU Operating Clock

CLKDT1	CLKDT0	Division ratio
1	1	fout/8
1	0	fout/4
0	1	fout/2
0	0	fout/1

fout: PLL output

The clock thus set becomes the system clock, which is used as the CPU operating clock and the bus clock.
At initial reset, the division ratio is set to fout/1, so the CPU is operated directly by the PLL output clock.
Since the device's current consumption can be decreased by reducing the CPU operating speed, switch over the operating frequency as necessary.
This setting is effective only for the high-speed (OSC3) clock, and has no effect when the low-speed (OSC1) clock is used as the system clock.

Note: Writing to CLKDT[1:0] is effective only when the power-control register protection flag is set to "0b10010110".

Switching over the CPU operating clock

Note: The CPU operating clock can be switched from OSC3 to OSC1 only when the low-speed (OSC1) oscillation circuit in the Peripheral Block is used.

After an initial reset, the CPU starts operating using the OSC3 clock. All internal peripheral circuits also operate.
In cases in which some peripheral circuits (e.g., programmable timer, serial interface, and A/D converter) that are clocked by the OSC3 clock do not need to be operate and the CPU can process its jobs at a low clock speed, the CPU operating clock can be switched to the OSC1 clock, thereby reducing current consumption. Use CLKCHG (D2) / Power control register (0x40180) to switch over the operating clock.

Procedure for switching over from the OSC3 clock to the OSC1 clock

1. Turn on the low-speed (OSC1) oscillation circuit (by writing "1" to SOSC1).
2. Wait until the OSC1 oscillation stabilizes (three seconds or more).
3. Change the CPU operating clock (by writing "0" to CLKCHG).
4. Turn off the high-speed (OSC3) oscillation circuit (by writing "0" to SOSC3).

* Steps 1 and 2 are required only when the low-speed (OSC1) oscillation circuit is inactive.

Notes:
- Use separate instructions to switch from OSC3 to OSC1 and turn the OSC3 oscillation off. If these operations are processed simultaneously using one instruction, the CPU may operate erratically.
- Make sure the operation of the peripheral circuits, such as the programmable timer, A/D converter, and serial interface, which are clocked by the OSC3 oscillation circuit, is terminated before the OSC3 oscillation is turned off in order to prevent them from operating erratically.

Procedure for switching over from the OSC1 clock to the OSC3 clock

1. Turn on the high-speed (OSC3) oscillation circuit (by writing "1" to SOSC3).
2. Wait until the OSC3 oscillation stabilizes (10 ms or more for a 3.3-V crystal resonator).
3. Switch over the CPU operating clock (by writing "1" to CLKCHG).

Note: The operating clock switchover by CLKCHG is effective only when both oscillation circuits are on and the power-control register protection flag is set to "0b10010110".

Power-Control Register Protection Flag

The power-control register at address 0x40180, which is used to control the oscillation circuits and the CPU operating clock, is normally disabled against writing in order to prevent it from malfunctioning due to unnecessary writing.
To enable this register for writing, the power-control register protection flag CLGP[7:0] (D[7:0]) / Power-control protection register (0x4019E) must be set to "0b10010110". Note that this setting allows for the power-control register (0x40180) to be written to only once, so all bits of CLGP[7:0] are cleared to "0" when this address is written to. Therefore, CLGP[7:0] must be set to "0b10010110" each time the power-control register (0x40180) is written to.
The flag CLGP[7:0] does not affect the readout from the power-control register (0x40180).

Operation in Standby Mode

In HALT mode, which is entered by executing the halt instruction, the high-speed (OSC3) and low-speed (OSC1) oscillation circuits both retain their status before HALT mode is entered. Under normal conditions, therefore, there is no need to control the oscillation circuits before entering or after exiting HALT mode.

The high-speed (OSC3) oscillation circuit stops operating after SLEEP mode is entered, which is done by executing the slp (sleep) instruction. If the high-speed (OSC3) oscillation circuit was operating before SLEEP mode was entered, it automatically starts oscillating again after SLEEP mode is exited.
In addition, if the CPU was operating using the OSC3 clock before SLEEP mode was entered, the CPU starts operating using the OSC3 clock again even after SLEEP mode is exited. The high-speed (OSC3) oscillation circuit requires 10 ms max. (when using a 3.3-V crystal resonator) for its oscillation to stabilize after oscillation starts. To prevent the CPU from operating erratically upon restart during this period, the C33 Core Block is designed to allow the OSC3 clock supply to the CPU to be disabled in the hardware after SLEEP mode is exited. Use 8T1ON (D2) / Clock option register (0x40190) to select this function. Use 8-bit programmable timer 1 to set the waitting time before clock supply is started.
The processing procedure and the operations to be performed when this function is used are as follows:

1. Disable the 8-bit programmable timer 1 interrupt.
2. Preset the initial count to 8-bit programmable timer 1.
   Set a value that will provide an ample stabilization waiting time. It is also necessary to set the input clock for 8-bit programmable timer 1 using the prescaler.
3. Enable the interrupt used to exit SLEEP mode.
   Before enabling the interrupt, be sure to reset the interrupt factor flag.
4. Write "0" to 8T1ON (turn on the function for waiting until the oscillation stabilizes after exiting SLEEP mode).
5. Activate 8-bit programmable timer 1 to start counting.
6. Enter SLEEP mode using the slp instruction.
   :
   SLEEP mode
   :
7. Exit SLEEP mode using an NMI, input port, or timer interrupt.
8. The high-speed (OSC3) oscillation circuit starts oscillating when SLEEP mode is exited. 8-bit programmable timer 1 also is made to start counting using the OSC3 clock.
9. 8-bit programmable timer 1 underflows.
   The operating clock supply to the CPU is begun by the underflow signal, so that the CPU restarts.

For details on how to control the 8-bit programmable timer, prescaler, and interrupts, refer to the description of each item in this manual.

Note: The function for waiting until the high-speed (OSC3) oscillation is stabilized by 8T1ON is effective only when SLEEP mode is exited.
Writing to 8T1ON is effective only when the power-control register protection flag is set to "0b10010110".

I/O Memory of Clock Generator

Table 6.4 lists the control bits of clock generator.

Table 6.4 Control Bits of Clock Generator

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Power control register	**0040180 (B)**	D7	**CLKDT1**	System clock division ratio selection	CLKDT[1:0] / Division ratio	0	R/W	
		D6	**CLKDT0**		1 1 / 1/8	0		
					1 0 / 1/4			
					0 1 / 1/2			
					0 0 / 1/1			
		D5	**PSCON**	Prescaler On/Off control	1 On / 0 Off	1	R/W	
		D4–3	–	reserved	–	0	–	Writing 1 not allowed.
		D2	**CLKCHG**	CPU operating clock switch	1 OSC3 / 0 OSC1	1	R/W	
		D1	**SOSC3**	High-speed (OSC3) oscillation On/Off	1 On / 0 Off	1	R/W	
		D0	**SOSC1**	Low-speed (OSC1) oscillation On/Off	1 On / 0 Off	1	R/W	
Clock option register	**0040190 (B)**	D7–4	–	–	–	–	–	0 when being read.
		D3	**HLT2OP**	HALT clock option	1 On / 0 Off	0	R/W	
		D2	**8T1ON**	OSC3-stabilize waiting function	1 Off / 0 On	1	R/W	
		D1	–	reserved	–	0	–	Do not write 1.
		D0	**PF1ON**	OSC1 external output control	1 On / 0 Off	0	R/W	
Power control protect register	**004019E (B)**	D7	**CLGP7**	Power control register protect flag	Writing 10010110 (0x96) removes the write protection of the power control register (0x40180). Writing another value set the write protection.	0	R/W	
		D6	**CLGP6**			0		
		D5	**CLGP5**			0		
		D4	**CLGP4**			0		
		D3	**CLGP3**			0		
		D2	**CLGP2**			0		
		D1	**CLGP1**			0		
		D0	**CLGP0**			0		

SOSC1: Low-speed (OSC1) oscillation control (D0) / Power control register (0x40180)

Turns the low-speed (OSC1) oscillation on or off.

Write "1": OSC1 oscillation turned on
Write "0": OSC1 oscillation turned off
Read: Valid

The oscillation of the low-speed (OSC1) oscillation circuit is stopped by writing "0" to SOSC1, and started again by writing "1".
Since a duration of maximum three seconds is required for oscillation to stabilize after the oscillation has been restarted, at least this length of time must pass before the OSC1 clock can be used.
Writing to SOSC1 is allowed only when CLGP[7:0] is set to "0b10010110". Note also that if the CPU is operating using the OSC1 clock, writing "0" to SOSC1 is ignored and the oscillation is not turned off.
At initial reset, SOSC1 is set to "1" (OSC1 oscillation turned on).

Note: This control bit is effective only when the low-speed (OSC1) oscillation circuit in the Peripheral Block is used.

SOSC3: High-speed (OSC3) oscillation control (D1) / Power control register (0x40180)

Turns the high-speed (OSC3) oscillation on or off.

Write "1": OSC3 oscillation turned on
Write "0": OSC3 oscillation turned off
Read: Valid

The oscillation of the high-speed (OSC3) oscillation circuit is stopped by writing "0" to SOSC3, and started again by writing "1".
Since a duration of maximum 10 ms (for a 3.3-V crystal resonator) is required for oscillation to stabilize after the oscillation has been restarted, at least this length of time must pass before the OSC3 clock can be used.
Writing to SOSC3 is allowed only when CLGP[7:0] is set to "0b10010110". Note also that if the CPU is operating using the OSC3 clock, writing "0" to SOSC3 is ignored and the oscillation is not turned off.
At initial reset, SOSC3 is set to "1" (OSC3 oscillation turned on).

CLKCHG: CPU operating clock switch (D2) / Power control register (0x40180)

Selects the CPU operating clock.

Write "1": OSC3 clock
Write "0": OSC1 clock
Read: Valid

The OSC3 clock is selected as the CPU operating clock by writing "1" to CLKCHG, and OSC1 is selected by writing "0". The operating clock can be switched over in this way only when both the high-speed (OSC3) and low-speed (OSC1) oscillation circuits are on. In addition, writing to CLKCHG is effective only when CLGP[7:0] is set to "0b10010110". Immediately after the oscillation circuit has started oscillating, wait for the oscillation to stabilize before switching over the CPU operating clock.
At initial reset, CLKCHG is set to "1" (OSC3 clock).

Note: This control bit is effective only when the low-speed (OSC1) oscillation circuit in the Peripheral Block is used.

CLKDT1–CLKDT0: CPU operating frequency selection (D[7:6]) / Power control register (0x40180)

Select the CPU operating clock frequency.

Table 6.5 Setting of CPU Operating Clock

CLKDT1	CLKDT0	Division ratio
1	1	fout/8
1	0	fout/4
0	1	fout/2
0	0	fout/1

fout: PLL output

This setting is effective when the CPU is operated using the high-speed (OSC3) clock and has no effect on the low-speed (OSC1) clock. Writing to CLKDT[1:0] is allowed only when CLGP[7:0] is set to "0b10010110".
At initial reset, CLKDT is set to "0" (fout/1).

8T1ON: High-speed (OSC3) oscillation waiting function (D2) / Clock option register (0x40190)

Sets the function for waiting until the high-speed (OSC3) oscillation stabilizes after SLEEP mode is exited.

Write "1": On
Write "0": Off
Read: Valid

After SLEEP mode is exited, the high-speed (OSC3) oscillation waiting function is effective by writing "1" to 8T1ON. For this function to be used, the waiting time must be set in 8-bit programmable timer 1 to allow it to start counting before entering SLEEP mode. After SLEEP mode is exited, the OSC3 clock is not supplied to the CPU until 8-bit programmable timer 1 underflows. This function will not work when 8T1ON is set to "0".
The high-speed (OSC3) oscillation waiting function is effective only when SLEEP mode is exited.
Writing to 8T1ON is effective only when CLGP[7:0] is set to "0b10010110".
When writing to 8T1ON, always be sure to write "0" to the reserved bits at address 0x40190.
At initial reset, 8T1ON is set to "0" (Off).

HLT2OP: HALT clock option (D3) / Clock option register (0x40190)

Select a HALT condition (basic mode or HALT2 mode).

Write "1": HALT2 mode
Write "0": Basic mode
Read: Valid

When "1" is written to HLT2OP, the CPU will enter HALT2 mode when the HALT instruction is executed. When "0" is written, the CPU will enter basic mode.
Writing to HLT2OP is allowed only when CLGP[7:0] is set to "0b10010110".
At initial reset, HLT2OP is set to "0" (basic mode).

The following shows the operating status in HALT mode (basic mode and HALT2 mode) and SLEEP mode.

Table 6.6 Operating Status in Standby Mode

Standby mode		Operating status	Reactivating factor
HALT mode	Basic mode	(1) The CPU clock is stopped. (CPU stop status) (2) BCU clock is supplied. (BCU run status) (3) Clocks for the peripheral circuits maintain the status before entering HALT mode. (run or stop) (4) The high-speed oscillation circuit maintains the status before entering HALT mode. (5) The low-speed oscillation circuit maintains the status before entering HALT mode.	(1) Reset, NMI (2) Enabled (not masked) interrupt factors
	HALT2 mode	(1) The CPU clock is stopped. (CPU stop status) (2) BCU clock is stopped. (BCU stop status) (3) Clocks for the peripheral circuits maintain the status before entering HALT mode. (run or stop) (4) The high-speed oscillation circuit maintains the status before entering HALT mode. (5) The low-speed oscillation circuit maintains the status before entering HALT mode.	
SLEEP mode		(1) The CPU clock is stopped. (CPU stop status) (2) BCU clock is stopped. (BCU stop status) (3) Clocks for the peripheral circuits are stopped. (4) The high-speed oscillation circuit is stopped. (5) The low-speed oscillation circuit maintains the status before entering SLEEP mode.	(1) Reset, NMI (2) Enabled (not masked) input port interrupt factors (3) Clock timer interrupt when the low-speed oscillation circuit is being operated

CLGP7–CLGP0:Power-control register protection flag ([D[7:0]) / Power control protection register (0x4019E)

These bits remove the protection against writing to addresses 0x40180 and 0x40190.

Write "0b10010110": Write protection removed
Write other than the above: No operation (write-protected)
Read: Valid

Before writing to address 0x40180 or 0x40190, set CLGP[7:0] to "0b10010110" to remove the protection against writing to that address. This clearing of write protection is effective for only one writing, so the bits are cleared to "0b00000000" by one writing. Therefore, CLGP[7:0] must be set each time the protected address is written to.
At initial reset, CLGP is set to "0b00000000" (write-protected).

Programming Notes

(1) Immediately after the high-speed (OSC3) oscillation circuit is turned on, a certain period of time is required for oscillation to stabilize (for a 3.3-V crystal resonator, this time is 10 ms max.). To prevent the device from operating erratically, do not use the clock until its oscillation has stabilized.
In particular, if the CPU is set in SLEEP mode during operation using the OSC3 clock, the high-speed (OSC3) oscillation circuit is turned off during in SLEEP mode and starts oscillating again after SLEEP mode is exited. To prevent the CPU from operating erratically at restart due to an unstable OSC3 clock, set a sufficient stabilization waiting time in 8-bit programmable timer 1 to turn on the oscillation stabilization waiting function after SLEEP mode is exited before entering SLEEP mode.

(2) The oscillation circuit used for the CPU operating clock cannot be turned off.

(3) The CPU operating clock can only be switched over when both the OSC3 and OSC1 oscillation circuits are on. Furthermore, when turning off an oscillation circuit that has become unnecessary as a result of the CPU operating clock switchover, be sure to use separate instructions for switchover and oscillation turnoff. If these two operations are processed simultaneously using one instruction, the CPU may operate erratically.

(4) If the high-speed (OSC3) oscillation circuit is turned off, all peripheral circuits operated using the OSC3 clock will be inactive.

(5) If the OSC3 clock is unnecessary, use the OSC1 clock to operate the CPU and turn the high-speed (OSC3) oscillation circuit off. This helps reduce current consumption.

THIS PAGE IS BLANK.

II-7 DBG (Debug Unit)

Debug Circuit

The C33 Core Block has a built-in debug circuit.
This functional block is provided to simply realize an advanced software development environment.

Note: The debug circuit does not work during normal operation. To construct a software development environment using the debug circuit, the ICD33 (In-Circuit Debugger for E0C33 Family) is separately required.

I/O Pins of Debug Circuit

Six pins used to exclusively connect the ICD33 (In-Circuit Debugger for E0C33 Family) are reserved for the debug circuit. The I/O voltage level of these pins is 3.3 V.
Table 7.1 lists the I/O pins of the debug circuit.

Table 7.1 I/O Pins of Debug Circuit

Pin name	I/O	Pull-up	Initial status	Voltage level	Function
DCLK	O	–	1	3.3 V	Clock output for debugging
DST2	O	–	0	3.3 V	Status output 2 for debugging
DST1	O	–	1	3.3 V	Status output 1 for debugging
DST0	O	–	1	3.3 V	Status output 0 for debugging
DPCO	O	–	1	3.3 V	PC output for debugging
DSIO	I/O	With pull-up	1 (Input)	3.3 V	Serial I/O for debugging

The DCLK, DST[2:0] and DPCO outputs are extended functions of the I/O port pins P14, P1[2:0] and P13, respectively. At initial reset, these pins are set as debug signal outputs.
If the debug circuit is not used, these pins can be used for I/O ports or the redefined peripheral circuits by writing "0" to CFEX[1:0] (D[1:0]) / Port function extension register (0x402DF). Refer to "I/O Ports (P Ports)" for the pin functions.

Note: When these pins are set as debug signal outputs, only the ICD33 (In-Circuit Debugger for E0C33 Family) can be connected to these pins. Leave these pins open if the ICD33 is not connected. For connecting the ICD33, refer to the "In-Circuit Debugger (ICD33) Manual".
Furthermore, the pin status is fixed as shown in the above table after a user reset.

THIS PAGE IS BLANK.

E0C33 Family ASIC Macro Manual

III PERIPHERAL BLOCK

Table of Contents

III-1 INTRODUCTION

The peripheral block consists of a prescaler, four channels of 8-bit programmable timer, six channels of 16-bit programmable timer including watchdog timer function, two channels of serial interface, input and I/O ports, low-speed (OSC1) oscillation circuit and a clock timer.

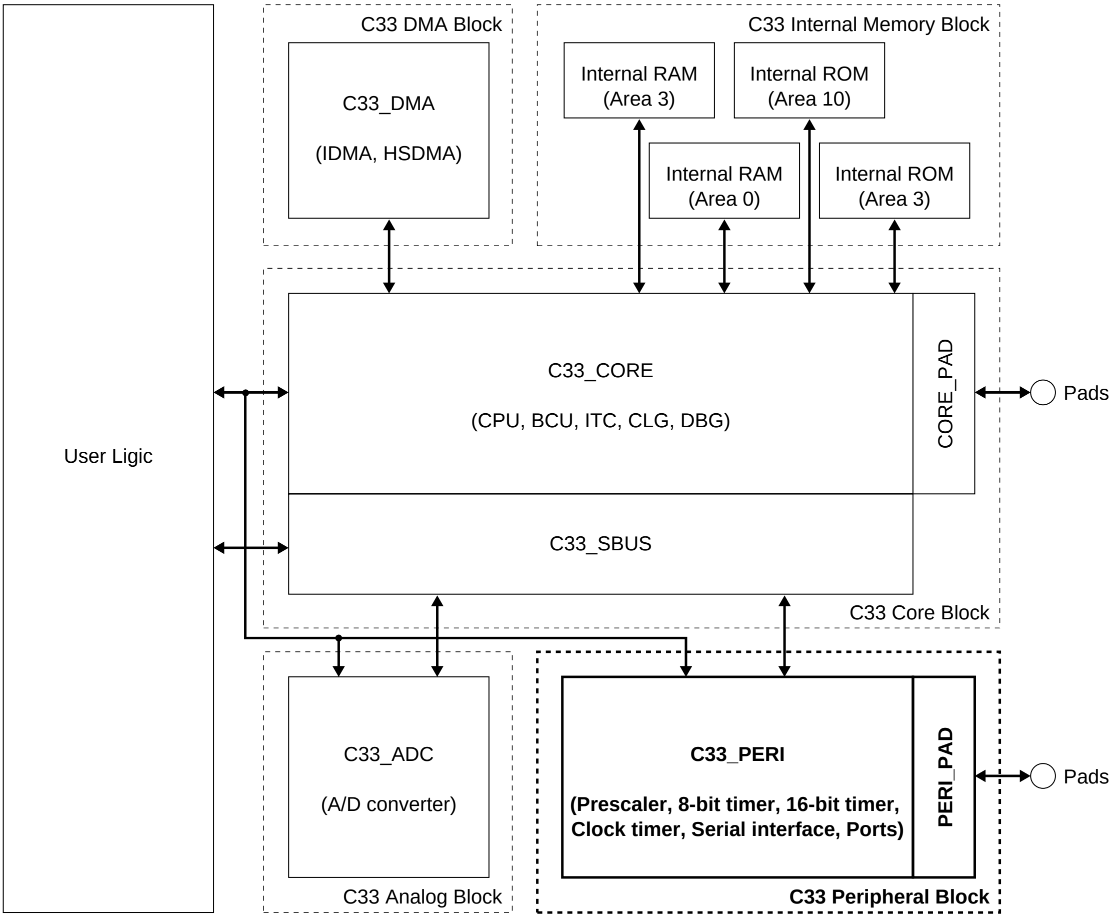


Figure 1.1 Peripheral Block

THIS PAGE IS BLANK.

III-2 PRESCALER

Configuration of Prescaler

The prescaler divides the source clock (OSC3/PLL output clock or OSC1 clock) to generate the clocks for the internal peripheral circuits. The prescaler division ratio can be selected for each peripheral circuit in a program. A clock control circuit to control the clock supply to each peripheral circuit is also included.
The following are the peripheral circuits that use the output clock:

- 16-bit programmable timers 5 to 0 (and watchdog timer)
- 8-bit programmable timers 3 to 0 (and serial interface)
- A/D converter

Figure 2.1 shows the configuration of the prescaler.
For details on control of each peripheral circuit, refer to each corresponding section in this manual.

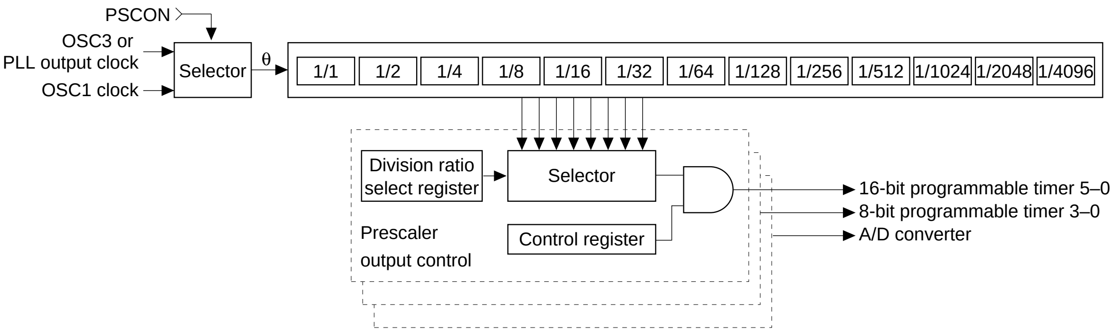


Figure 2.1 Configuration of Prescaler and Clock Control Circuit

Source Clock

The source clock for the prescaler can be selected using PSCDT0 (D0) / Prescaer clock select register (0x40181).
When PSCDT0 = "0", the OSC3 clock (when the PLL is not used) or the PLL output clock (when the PLL is used) is selected.
When PSCDT0 = "1", the OSC1 clock (typ. 32 kHz) is selected.
At initial reset, the OSC3/PLL output clock is selected.

Note: For the prescaler clock, the clock source same as the CPU operating clock must be selected.

For details on how to control the oscillation circuit and CPU operating clock, refer to "CLG (Clock Generator)".
At initial reset, the OSC3 clock is selected.

The source clock is supplied to the prescaler by writing "1" to PSCON (D5) / Power control register (0x40180). At initial reset, PSCON is set to "1", so the prescaler is in an operating state. If all of said peripheral circuits can be turned off, stop the prescaler by writing "0" to PSCON. This helps to reduce current consumption.

Selecting Division Ratio and Output Control for Prescaler

The prescaler has registers for selecting the division ratio and clock output control separately for each peripheral circuit described above, allowing each peripheral circuit to be controlled.
The prescaler's division ratio can be selected from among eight ratios set for each peripheral circuit through the use of the division ratio selection bits. The divided clock is output to the corresponding peripheral circuit by writing "1" to the clock control bit.

Table 2.1 Control Bits of the Clock Control Registers

Peripheral circuit	Division ratio selection bit	Clock control bit
16-bit programmable timer 0	P16TS0[2:0] (D[2:0]/0x40147)∗1	P16TON0 (D3/0x40147)
16-bit programmable timer 1	P16TS1[2:0] (D[2:0]/0x40148)∗1	P16TON1 (D3/0x40148)
16-bit programmable timer 2	P16TS2[2:0] (D[2:0]/0x40149)∗1	P16TON2 (D3/0x40149)
16-bit programmable timer 3	P16TS3[2:0] (D[2:0]/0x4014A)∗1	P16TON3 (D3/0x4014A)
16-bit programmable timer 4	P16TS4[2:0] (D[2:0]/0x4014B)∗1	P16TON4 (D3/0x4014B)
16-bit programmable timer 5	P16TS5[2:0] (D[2:0]/0x4014C)∗1	P16TON5 (D3/0x4014C)
8-bit programmable timer 0	P8TS0[2:0] (D[2:0]/0x4014D)∗2	P8TON0 (D3/0x4014D)
8-bit programmable timer 1	P8TS1[2:0] (D[6:4]/0x4014D)∗3	P8TON1 (D7/0x4014D)
8-bit programmable timer 2	P8TS2[2:0] (D[2:0]/0x4014E)∗4	P8TON2 (D3/0x4014E)
8-bit programmable timer 3	P8TS3[2:0] (D[6:4]/0x4014E)∗2	P8TON3 (D7/0x4014E)
A/D converter	PSAD[2:0] (D[2:0]/0x4014F)∗2	PSONAD (D3/0x4014F)

∗1 to ∗4: See Table 2.2.

Table 2.2 Division Ratio

Bit setting	7	6	5	4	3	2	1	0
∗1	θ/4096	θ/1024	θ/256	θ/64	θ/16	θ/4	θ/2	θ/1
∗2	θ/256	θ/128	θ/64	θ/32	θ/16	θ/8	θ/4	θ/2
∗3	θ/4096	θ/2048	θ/1024	θ/512	θ/256	θ/128	θ/64	θ/32
∗4	θ/4096	θ/2048	θ/64	θ/32	θ/16	θ/8	θ/4	θ/2

(θ = Source clock selected by PSCDT0)

Current consumption can be reduced by turning off the clock output to the peripheral circuits that are unused among those listed above.

Note: In the following cases, the prescaler output clock may contain a hazard:
- If, when a clock is output, its division ratio is changed
- When the clock output is switched between on and off
- When the oscillation circuit is turned off or the CPU operating clock is switched over

Before performing these operations, make sure the 16-bit and 8-bit programmable timers and the A/D converter are turned off.

Source Clock Output to 8-Bit Programmable Timer

In addition to the divided clock, the prescaler can output the source clock directly to the 8-bit programmable timer. This function can be selected for each 8-bit timer using P8TPCKx bit.
8-bit timer 0: P8TPCK0 (D0) / 8-bit timer clock select register (0x40146)
8-bit timer 1: P8TPCK1 (D1) / 8-bit timer clock select register (0x40146)
8-bit timer 2: P8TPCK2 (D2) / 8-bit timer clock select register (0x40146)
8-bit timer 3: P8TPCK3 (D3) / 8-bit timer clock select register (0x40146)

When P8TPCKx is set to "1", the prescaler input clock (θ/1) is selected for the 8-bit timer x operating clock.
The clock output is controlled by the P8TONx bit even if P8TPCKx is set to "1".
When P8TPCKx is "0", the divided clock that is selected by P8TSx[2:0] will be output to the 8-bit timer x.
At initial reset, P8TPCKx is set to "0" and P8TSx[2:0] becomes effective.

I/O Memory of Prescaler

Table 2.3 shows the control bits of the prescaler.

Table 2.3 Control Bits of Prescaler

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer	**0040146**	D7–4	–	reserved	–				–	–	0 when being read.
clock select	**(B)**	D3	**P8TPCK3**	8-bit timer 3 clock selection	1	θ/1	0	Divided clk.	0	R/W	θ: selected by
register		D2	**P8TPCK2**	8-bit timer 2 clock selection	1	θ/1	0	Divided clk.	0	R/W	Prescaler clock select
		D1	**P8TPCK1**	8-bit timer 1 clock selection	1	θ/1	0	Divided clk.	0	R/W	register (0x40181)
		D0	**P8TPCK0**	8-bit timer 0 clock selection	1	θ/1	0	Divided clk.	0	R/W	
16-bit timer 0	**0040147**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON0**	16-bit timer 0 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS02**	16-bit timer 0	P16TS0[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS01**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS00**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			16-bit timer 0 can be
					0	1	1	θ/16			used as a watchdog
					0	1	0	θ/4			timer.
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 1	**0040148**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON1**	16-bit timer 1 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS12**	16-bit timer 1	P16TS1[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS11**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS10**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 2	**0040149**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON2**	16-bit timer 2 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS22**	16-bit timer 2	P16TS2[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS21**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS20**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 3	**004014A**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON3**	16-bit timer 3 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS32**	16-bit timer 3	P16TS3[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS31**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS30**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 4	**004014B**	D7–4	–	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON4**	16-bit timer 4 clock control	1	On	0	Off	0	R/W	
register		D2	**P16TS42**	16-bit timer 4	P16TS4[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS41**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS40**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 5 clock control register	**004014C (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**P16TON5**	16-bit timer 5 clock control	1 On		0 Off		0	R/W	
		D2	**P16TS52**	16-bit timer 5 clock division ratio selection	P16TS5[2:0]			Division ratio	0	R/W	θ: selected by Prescaler clock select register (0x40181)
		D1	**P16TS51**		1	1	1	θ/4096	0		
		D0	**P16TS50**		1	1	0	θ/1024	0		
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
8-bit timer 0/1 clock control register	**004014D (B)**	D7	**P8TON1**	8-bit timer 1 clock control	1 On		0 Off		0	R/W	
		D6	**P8TS12**	8-bit timer 1 clock division ratio selection	P8TS1[2:0]			Division ratio	0	R/W	θ: selected by Prescaler clock select register (0x40181)
		D5	**P8TS11**		1	1	1	θ/4096	0		
		D4	**P8TS10**		1	1	0	θ/2048	0		
					1	0	1	θ/1024			
					1	0	0	θ/512			8-bit timer 1 can generate the OSC3 oscillation-stabilize waiting period.
					0	1	1	θ/256			
					0	1	0	θ/128			
					0	0	1	θ/64			
					0	0	0	θ/32			
		D3	**P8TON0**	8-bit timer 0 clock control	1 On		0 Off		0	R/W	
		D2	**P8TS02**	8-bit timer 0 clock division ratio selection	P8TS0[2:0]			Division ratio	0	R/W	θ: selected by Prescaler clock select register (0x40181)
		D1	**P8TS01**		1	1	1	θ/256	0		
		D0	**P8TS00**		1	1	0	θ/128	0		
					1	0	1	θ/64			
					1	0	0	θ/32			8-bit timer 0 can generate the DRAM refresh clock.
					0	1	1	θ/16			
					0	1	0	θ/8			
					0	0	1	θ/4			
					0	0	0	θ/2			
8-bit timer 2/3 clock control register	**004014E (B)**	D7	**P8TON3**	8-bit timer 3 clock control	1 On		0 Off		0	R/W	
		D6	**P8TS32**	8-bit timer 3 clock division ratio selection	P8TS3[2:0]			Division ratio	0	R/W	θ: selected by Prescaler clock select register (0x40181)
		D5	**P8TS31**		1	1	1	θ/256	0		
		D4	**P8TS30**		1	1	0	θ/128	0		
					1	0	1	θ/64			
					1	0	0	θ/32			8-bit timer 3 can generate the clock for the serial I/F Ch.1.
					0	1	1	θ/16			
					0	1	0	θ/8			
					0	0	1	θ/4			
					0	0	0	θ/2			
		D3	**P8TON2**	8-bit timer 2 clock control	1 On		0 Off		0	R/W	
		D2	**P8TS22**	8-bit timer 2 clock division ratio selection	P8TS2[2:0]			Division ratio	0	R/W	θ: selected by Prescaler clock select register (0x40181)
		D1	**P8TS21**		1	1	1	θ/4096	0		
		D0	**P8TS20**		1	1	0	θ/2048	0		
					1	0	1	θ/64			
					1	0	0	θ/32			8-bit timer 2 can generate the clock for the serial I/F Ch.0.
					0	1	1	θ/16			
					0	1	0	θ/8			
					0	0	1	θ/4			
					0	0	0	θ/2			
A/D clock control register	**004014F (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**PSONAD**	A/D converter clock control	1 On		0 Off		0	R/W	
		D2	**PSAD2**	A/D converter clock division ratio selection	P8TS0[2:0]			Division ratio	0	R/W	θ: selected by Prescaler clock select register (0x40181)
		D1	**PSAD1**		1	1	1	θ/256	0		
		D0	**PSAD0**		1	1	0	θ/128	0		
					1	0	1	θ/64			
					1	0	0	θ/32			
					0	1	1	θ/16			
					0	1	0	θ/8			
					0	0	1	θ/4			
					0	0	0	θ/2			

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Power control register	**0040180 (B)**	D7	**CLKDT1**	System clock division ratio selection	CLKDT[1:0]		Division ratio		0	R/W	
		D6	**CLKDT0**		1	1	1/8		0		
					1	0	1/4				
					0	1	1/2				
					0	0	1/1				
		D5	**PSCON**	Prescaler On/Off control	1 On		0 Off		1	R/W	
		D4–3	–	reserved	–				0	–	Writing 1 not allowed.
		D2	**CLKCHG**	CPU operating clock switch	1 OSC3		0 OSC1		1	R/W	
		D1	**SOSC3**	High-speed (OSC3) oscillation On/Off	1 On		0 Off		1	R/W	
		D0	**SOSC1**	Low-speed (OSC1) oscillation On/Off	1 On		0 Off		1	R/W	
Prescaler clock select register	**0040181 (B)**	D7–1	–	reserved	–				0	–	
		D0	**PSCDT0**	Prescaler clock selection	1 OSC1		0 OSC3/PLL		0	R/W	
Power control protect register	**004019E (B)**	D7	**CLGP7**	Power control register protect flag	Writing 10010110 (0x96) removes the write protection of the power control register (0x40180). Writing another value set the write protection.				0	R/W	
		D6	**CLGP6**						0		
		D5	**CLGP5**						0		
		D4	**CLGP4**						0		
		D3	**CLGP3**						0		
		D2	**CLGP2**						0		
		D1	**CLGP1**						0		
		D0	**CLGP0**						0		

PSCON: Prescaler on/off control (D5) / Power control register (0x40180)

Turns the prescaler on or off.

Write "1": On
Write "0": Off
Read: Valid

The source clock is input to the prescaler by writing "1" to PSCON, thereby starting a dividing operation.
The prescaler is turned off by writing "0". If the peripheral circuits do not need to be operated, write "0" to this bit to reduce current consumption. Since PSCON is protected against writing the same as SOSC1, SOSC3, CLKCHG and CLKDT[1:0], CLGP[7:0] must be set to "0b10010110" before PSCON can be changed.
At initial reset, PSCON is set to "1" (On).

CLGP7–CLGP0:Power-control register protection flag ([D[7:0]) / Power control protection register (0x4019E)

These bits remove the protection against writing to addresses 0x40180 and 0x40190.

Write "0b10010110": Write protection removed
Write other than the above: No operation (write-protected)
Read: Valid

Before writing to address 0x40180 or 0x40190, set CLGP[7:0] to "0b10010110" to remove the protection against writing to that address. This clearing of write protection is effective for only one writing, so the bits are cleared to "0b00000000" by one writing. Therefore, CLGP[7:0] must be set each time the protected address is written to.
At initial reset, CLGP is set to "0b00000000" (write-protected).

PSCDT0: Prescaler clock selection (D0) / Prescaler clock select register (0x40181)

Select the source clock for the prescaler.

Write "1": OSC1 clock
Write "0": OSC3 clock/PLL output clock
Read: Valid

When "1" is written to PSCDT0, the OSC1 clock (typ. 32 kHz) is selected.
When "0" is written, the OSC3 clock (when the PLL is not used) or the PLL output clock (when the PLL is used) is selected.
For the prescaler clock, the clock source same as the CPU operating clock must be selected.
At initial reset, PSCDT0 is set to "0" (OSC3 clock/PLL output clock).

P16TS0[2:0]: 16-bit timer 0 clock division ratio (D[2:0]) / 16-bit timer 0 clock control register (0x40147)
P16TS1[2:0]: 16-bit timer 1 clock division ratio (D[2:0]) / 16-bit timer 1 clock control register (0x40148)
P16TS2[2:0]: 16-bit timer 2 clock division ratio (D[2:0]) / 16-bit timer 2 clock control register (0x40149)
P16TS3[2:0]: 16-bit timer 3 clock division ratio (D[2:0]) / 16-bit timer 3 clock control register (0x4014A)
P16TS4[2:0]: 16-bit timer 4 clock division ratio (D[2:0]) / 16-bit timer 4 clock control register (0x4014B)
P16TS5[2:0]: 16-bit timer 5 clock division ratio (D[2:0]) / 16-bit timer 5 clock control register (0x4014C)
P8TS0[2:0]: 8-bit timer 0 clock division ratio (D[2:0]) / 8-bit timer 0/1 clock control register (0x4014D)
P8TS1[2:0]: 8-bit timer 1 clock division ratio (D[6:4]) / 8-bit timer 0/1 clock control register (0x4014D)
P8TS2[2:0]: 8-bit timer 2 clock division ratio (D[2:0]) / 8-bit timer 2/3 clock control register (0x4014E)
P8TS3[2:0]: 8-bit timer 3 clock division ratio (D[6:4]) / 8-bit timer 2/3 clock control register (0x4014E)
PSAD[2:0]: A/D converter clock division ratio (D[2:0]) / A/D clock control register (0x4014F)

Select a clock for each peripheral circuit.
The desired division ratio can be selected from among the eight ratios shown on the I/O map. Note that the division ratio differs for each peripheral circuit.
These bits can also be read out.
At initial reset, all of these bits are set to "0b000" (highest frequency available).

P16TON0: 16-bit timer 0 clock control (D3) / 16-bit timer 0 clock control register (0x40147)
P16TON1: 16-bit timer 1 clock control (D3) / 16-bit timer 1 clock control register (0x40148)
P16TON2: 16-bit timer 2 clock control (D3) / 16-bit timer 2 clock control register (0x40149)
P16TON3: 16-bit timer 3 clock control (D3) / 16-bit timer 3 clock control register (0x4014A)
P16TON4: 16-bit timer 4 clock control (D3) / 16-bit timer 4 clock control register (0x4014B)
P16TON5: 16-bit timer 5 clock control (D3) / 16-bit timer 5 clock control register (0x4014C)
P8TON0: 8-bit timer 0 clock control (D3) / 8-bit timer 0/1 clock control register (0x4014D)
P8TON1: 8-bit timer 1 clock control (D7) / 8-bit timer 0/1 clock control register (0x4014D)
P8TON2: 8-bit timer 2 clock control (D3) / 8-bit timer 2/3 clock control register (0x4014E)
P8TON3: 8-bit timer 3 clock control (D7) / 8-bit timer 2/3 clock control register (0x4014E)
PSONAD: A/D converter clock control (D3) / A/D clock control register (0x4014F)

Control the clock supply to each peripheral circuit.

Write "1": On
Write "0": Off
Read: Valid

The clock selected using the division ratio setup bits is output to the corresponding peripheral circuit by writing "1" to these bits.
The clock is not output by writing "0". If the peripheral circuits do not need to be operated, write "0" to these bits. This helps to reduce current consumption.
At initial reset, all of these bits are set to "0" (Off).

P8TPCK0: 8-bit timer 0 clock selection (D0) / 8-bit timer clock select register (0x40146)
P8TPCK1: 8-bit timer 1 clock selection (D1) / 8-bit timer clock select register (0x40146)
P8TPCK2: 8-bit timer 2 clock selection (D2) / 8-bit timer clock select register (0x40146)
P8TPCK3: 8-bit timer 3 clock selection (D3) / 8-bit timer clock select register (0x40146)

Select the operating clock for the 8-bit programmable timer.

Write "1": Prescaler input clock ($\theta/1$)
Write "0": Divided clock
Read: Valid

When "1" is written to P8TPCKx, the prescaler input clock ($\theta/1$) is selected for the 8-bit timer x operating clock.
The clock output is controlled by the P8TONx bit even if P8TPCKx is set to "1".
When "0" is written, the divided clock that is selected by P8TSx[2:0] will be output to the 8-bit timer x.
At initial reset, P8TPCKx is set to "0" (divided clock).

Programming Notes

(1) For the prescaler clock, the clock source same as the CPU operating clock must be selected.

(2) In the following cases, the prescaler output clock may contain a hazard:
- If, during outputting of a clock, its division ratio is changed
- When the clock output is switched between on and off
- When the oscillation circuit is turned off or the CPU operating clock is switched over

Before performing these operations, make sure the 16-bit and 8-bit programmable timers and the A/D converter are turned off.

(3) When the 16-bit and 8-bit programmable timers and the A/D converter do not need to be operated, turn off the clock supply to those peripheral circuits. This helps to reduce current consumption.

THIS PAGE IS BLANK.

III-3 8-BIT PROGRAMMABLE TIMERS

Configuration of 8-Bit Programmable Timer

The Peripheral Block contains four channels of 8-bit programmable timers (timers 0 to 3).
Figure 3.1 shows the structure of the 8-bit programmable timer.

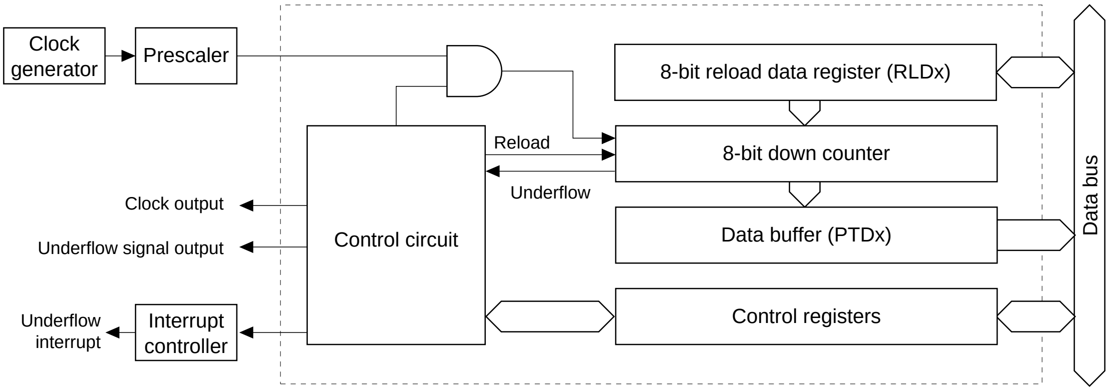


Figure 3.1 Structure of 8-Bit Programmable Timer

Each timer consists of an 8-bit presentable counter and can output a clock generated by the counter's underflow signal to the internal peripheral circuits or external devices. The output clock cycle can be selected from a wide range of cycles by setting the preset data that can be set in the software and the input clock in the prescaler.

Output Pins of 8-Bit Programmable Timers

Table 3.1 shows the pins that are used to output the underflow signals of the 8-bit programmable timers to external devices.

Table 3.1 Output Pins of 8-Bit Programmable Timers

Pin name	I/O	Function	Function select bit
P10/EXCL0/ T8UF0	I/O	I/O port / 16-bit timer 0 event counter input / 8-bit timer 0 output / DST0 output	CFP10(D0)/P1 function select register (0x402D4) CFEX1(D1)/Port function extension register (0x402DF)
P11/EXCL1/ T8UF1	I/O	I/O port / 16-bit timer 1 event counter input / 8-bit timer 1 output / DST1 output	CFP11(D1/P1 function select register (0x402D4) CFEX1(D1)/Port function extension register (0x402DF)
P12/EXCL2/ T8UF2	I/O	I/O port / 16-bit timer 2 event counter input / 8-bit timer 2 output / DST2 output	CFP12(D2/P1 function select register (0x402D4) CFEX0(D0)/Port function extension register (0x402DF)
P13/EXCL3/ T8UF3	I/O	I/O port / 16-bit timer 3 event counter input / 8-bit timer 3 output / DPCO output	CFP13(D3/P1 function select register (0x402D4) CFEX1(D1)/Port function extension register (0x402DF)

T8UFx (output pin of the 8-bit programmable timer)

This pin outputs a clock divided in each 8-bit programmable timer. The pulse width is equal to that of input clock of the 8-bit programmable timer (prescaler output). Therefore, the pulse width varies according to the prescaler setting.

How to set the output pins of the 8-bit programmable timer

All pins used by the 8-bit programmable timers are shared with I/O ports, event counter inputs of the 16-bit programmable timers and debug signal outputs. At cold start, all these pins are set for the debug signal outputs (function select bit CFP1[3:0] = "0", port extended function bit CFEX[1:0] = "1"). When using the clock output function of the 8-bit programmable timer, write "0" to the port extended function bit CFEXx and write "1" to the function select bit CFP1x for the corresponding pin.

Then, after setting the above, write "1" to the I/O port's I/O control bit IOC1x (D[3:0]) / P1 I/O control register (0x402D6) to set to output mode. In input mode, the pin functions as the 16-bit programmable timer's event counter input and cannot be used to output a clock of the 8-bit programmable timer. At cold start, the register is set to input mode. At hot start, the register retains its status from prior to the reset.

Uses of 8-Bit Programmable Timers

The down-counter of the 8-bit programmable timer cyclically outputs an underflow signal according to the preset data that is set in the software. This underflow signal is used to generate an interrupt request to the CPU or to control the internal peripheral circuits. In addition, this signal can be output to external devices.
Furthermore, each 8-bit programmable timer generates a clock from the underflow signal by dividing it by 2, and the resulting clock is output to a specific internal peripheral circuit.

CPU interrupt request/IDMA invocation request

Each timer's underflow condition can be used as an interrupt factor to output an interrupt request to the CPU. Therefore, an interrupt can be generated at an interval that is set in the software.
This interrupt factor also can be used to invoke IDMA or HSDMA.

Clock output to external devices

The underflow signal can be output from the chip to the outside. This output can be used to control external devices. The output pins of each timer are described in the preceding section.

Control of and clock supply to internal peripheral circuits

The following describes the functions controlled by the underflow signal from the 8-bit programmable timer and the internal peripheral circuits that use the timer's output clock.

8-bit programmable timer 0

• DRAM refresh

When the BCU has a DRAM directly connected to its external bus, the underflow signal from timer 0 can be used as a DRAM refresh request signal. This enables the intervals of the refresh cycle to be programmed.
To use this function, write "1" to the BCU's control bit RPC (D9) / Bus control register (0x4812E) to enable the DRAM refresh.

• A/D conversion start trigger

The A/D converter enables a trigger for starting the A/D conversion to be selected from among four available types. One of these is the underflow signal of the 8-bit programmable timer 0. This makes it possible to perform the A/D conversion at programmable intervals.
To use this function, write "10" to the A/D converter control bit TS[1:0] (D[4:3]) / A/D trigger register (0x40242) to select the 8-bit programmable timer 0 as the trigger.

8-bit programmable timer 1

• Oscillation stabilization wait time of the high-speed (OSC3) oscillation circuit

When SLEEP mode is cleared by an external interrupt, the high-speed (OSC3) oscillation circuit starts oscillating. To prevent the CPU from being operated erratically by an unstable clock before the oscillation stabilizes, the C33 Core Block enables setting of the waiting time before the CPU starts operating after SLEEP is cleared. Use the 8-bit programmable timer 1 to generate this waiting time. If the 8-bit programmable timer 1 is set so that the timer is actuated when the high-speed (OSC3) oscillation circuit starts oscillating the timer and, after the oscillation stabilization time elapses, an underflow signal is generated, then the CPU can be started up by that underflow signal.
To use this function, write "0" to the oscillation circuit control bit 8T1ON (D2) / Clock option register (0x40190) to enable the oscillation stabilization waiting function.

8-bit programmable timer 2

• Clock supply to the Ch.0 serial interface

When using the Ch.0 serial interface in the clock-synchronized master mode or the internal clock-based asynchronous mode, the output clock derived from the underflow signal of the 8-bit programmable timer 2 by dividing it by 2 is supplied to the serial interface as its operating clock. This enables the transfer rate of the serial interface to be programmed.

To use this function, write "0" to the serial interface control bit SSCK0 (D2) / Serial I/F Ch.0 control register (0x401E3) to select the internal clock.

8-bit programmable timer 3

• Clock supply to the Ch.1 serial interface

When using the Ch.1 serial interface in the clock-synchronized master mode or the internal clock-based asynchronous mode, the output clock derived from the underflow signal of the 8-bit programmable timer 3 by dividing it by 2 is supplied to the serial interface as its operating clock. This enables the transfer rate of the serial interface to be programmed.

To use this function, write "0" to the serial interface control bit SSCK1 (D2) / Serial I/F Ch.1 control register (0x401E8) to select the internal clock.

Control and Operation of 8-Bit Programmable Timer

With the 8-bit programmable timer, the following settings must first be made before it starts counting:
1. Setting the output pin (only when necessary)
2. Setting the input clock
3. Setting the preset data (initial counter value)
4. Setting the interrupt/IDMA/HSDMA

Setting of an output pin is necessary only when the output clock of the 8-bit programmable timer is supplied to external devices. For details on how to set the pin, refer to "Output Pins of 8-Bit Programmable Timers".
For details on how to set interrupts and DMA, refer to "8-Bit Programmable Timer Interrupts and DMA".

Note: The 8-bit programmable timers 0 through 3 all operate in the same way during counting, and the structure of their control registers is also the same. The control bit names are assigned the numerals "0" through "3" to denote the timer numbers. Since all these timers have common functions, timer numbers here are represented it is by "x" unless necessary to specify a timer number.

Setting the input clock

The 8-bit programmable timer is operated by the prescaler's output clock. The prescaler's division ratio can be selected for each timer.

	Division ratio select bit	Clock control bit	Register
8-bit timer 0:	P8TS0[2:0] (D2:0])	P8TON0 (D3)	8-bit timer 0/1 clock control register (0x4014D)
8-bit timer 1:	P8TS1[2:0] (D6:4])	P8TON1 (D7)	8-bit timer 0/1 clock control register (0x4014D)
8-bit timer 2:	P8TS2[2:0] (D2:0])	P8TON2 (D3)	8-bit timer 2/3 clock control register (0x4014E)
8-bit timer 3:	P8TS3[2:0] (D6:4])	P8TON3 (D7)	8-bit timer 2/3 clock control register (0x4014E)

Note that the division ratios differ for each timer (see Table 3.2).
Furthermore, the prescaler input clock can be directly supplied to the 8-bit timer by writing "1" to the P8TCPKx bit in the 8-bit timer clock select register (0x40146).
Timer 0 clock selection: P8TCPK0 (D0) / 8-bit timer clock select register (0x40146)
Timer 1 clock selection: P8TCPK1 (D1) / 8-bit timer clock select register (0x40146)
Timer 2 clock selection: P8TCPK2 (D2) / 8-bit timer clock select register (0x40146)
Timer 3 clock selection: P8TCPK3 (D3) / 8-bit timer clock select register (0x40146)

When using the divided clock selected by P8TSx, set P8TCPKx to "0".

Table 3.2 Input Clock Selection

Timer	P8TSx = 7	P8TSx = 6	P8TSx = 5	P8TSx = 4	P8TSx = 3	P8TSx = 2	P8TSx = 1	P8TSx = 0	P8TCPK = 1
Timer 0	f_{PSCIN}/256	f_{PSCIN}/128	f_{PSCIN}/64	f_{PSCIN}/32	f_{PSCIN}/16	f_{PSCIN}/8	f_{PSCIN}/4	f_{PSCIN}/2	f_{PSCIN}/1
Timer 1	f_{PSCIN}/4096	f_{PSCIN}/2048	f_{PSCIN}/1024	f_{PSCIN}/512	f_{PSCIN}/256	f_{PSCIN}/128	f_{PSCIN}/64	f_{PSCIN}/32	f_{PSCIN}/1
Timer 2	f_{PSCIN}/4096	f_{PSCIN}/2048	f_{PSCIN}/64	f_{PSCIN}/32	f_{PSCIN}/16	f_{PSCIN}/8	f_{PSCIN}/4	f_{PSCIN}/2	f_{PSCIN}/1
Timer 3	f_{PSCIN}/256	f_{PSCIN}/128	f_{PSCIN}/64	f_{PSCIN}/32	f_{PSCIN}/16	f_{PSCIN}/8	f_{PSCIN}/4	f_{PSCIN}/2	f_{PSCIN}/1

f_{PSCIN}: Prescaler input clock frequency

The selected clock is output from the prescaler to the 8-bit programmable timer by writing "1" to P8TONx.

Notes:
- The 8-bit programmable timer operates only when the prescaler is operating. (Refer to "Prescaler".)
- Do not use a clock that is faster than the CPU operating clock as the 8-bit programmable timer.
- When setting an input clock, make sure the 8-bit programmable timer is turned off.

Setting preset data (initial counter value)

Each timer has an 8-bit down-counter and a reload data register. The reload data register RLDx is used to set the initial value of the down-counter of each timer.

Timer 0 reload data: RLD0[7:0] (D[7:0]) / 8-bit timer 0 reload data register (0x40161)
Timer 1 reload data: RLD1[7:0] (D[7:0]) / 8-bit timer 1 reload data register (0x40165)
Timer 2 reload data: RLD2[7:0] (D[7:0]) / 8-bit timer 2 reload data register (0x40169)
Timer 3 reload data: RLD3[7:0] (D[7:0]) / 8-bit timer 3 reload data register (0x4016D)

The reload data registers can be read and written. At initial reset, the reload data registers are not initialized.

The data written to this register is preset in the down-counter, and the counter starts counting down from the preset value.

Data is thus preset in the down-counter in the following two cases:

1. **When it is preset in the software**

   Presetting in the software is performed using the preset control bit PSETx. When this bit is set to "1", the content of the reload data register is loaded into the down-counter at that point.

   Timer 0 preset: PSET0 (D1) / 8-bit timer 0 control register (0x40160)
   Timer 1 preset: PSET1 (D1) / 8-bit timer 1 control register (0x40164)
   Timer 2 preset: PSET2 (D1) / 8-bit timer 2 control register (0x40168)
   Timer 3 preset: PSET3 (D1) / 8-bit timer 3 control register (0x4016C)

2. **When the down-counter underflown during counting**

   Since the reload data is preset in the down-counter upon underflow, its underflow cycle is determined by the value that is set in the reload data register. This underflow signal controls each function described in the preceding section.

Before starting the 8-bit programmable timer, set the initial value in the reload data register and use the PSETx bit to preset the data in the down-counter.

The underflow cycle is determined by the prescaler setting and the reload data. The relationship between these two parameters is expressed by the following equation:

$$\text{Under flow cycle} = \frac{\text{RLDx} + 1}{f_{\text{PSCIN}} \times \text{dr}} \text{ [sec.]}$$

f_{PSCIN}: Prescaler input clock frequency [Hz]
dr: Prescaler division ratio set by P8TSx
RLDx: Set value of the RLDx register (0 to 255)

Timer RUN/STOP control

Each timer has a PTRUNx bit to control RUN/STOP.

Timer 0 RUN/STOP control: PTRUN0 (D0) / 8-bit timer 0 control register (0x40160)
Timer 1 RUN/STOP control: PTRUN1 (D0) / 8-bit timer 1 control register (0x40164)
Timer 2 RUN/STOP control: PTRUN2 (D0) / 8-bit timer 2 control register (0x40168)
Timer 3 RUN/STOP control: PTRUN3 (D0) / 8-bit timer 3 control register (0x4016C)

The timer is initiated to start counting down by writing "1" to PTRUNx. Writing "0" to PTRUNx disables the clock input and causes the timer to stop counting.

This RUN/STOP control does not affect the counter data. Even when the timer has stopped counting, the counter retains its count so that it can start counting again from that point.

When the terminal count is reached and the counter underflows, the initial value is reloaded from the reload data register into the counter.

When both the timer RUN/STOP control bit (PTRUNx) and the timer preset bit (PSETx) are set to "1" at the same time, the timer starts counting after presetting the reload register value into the counter.

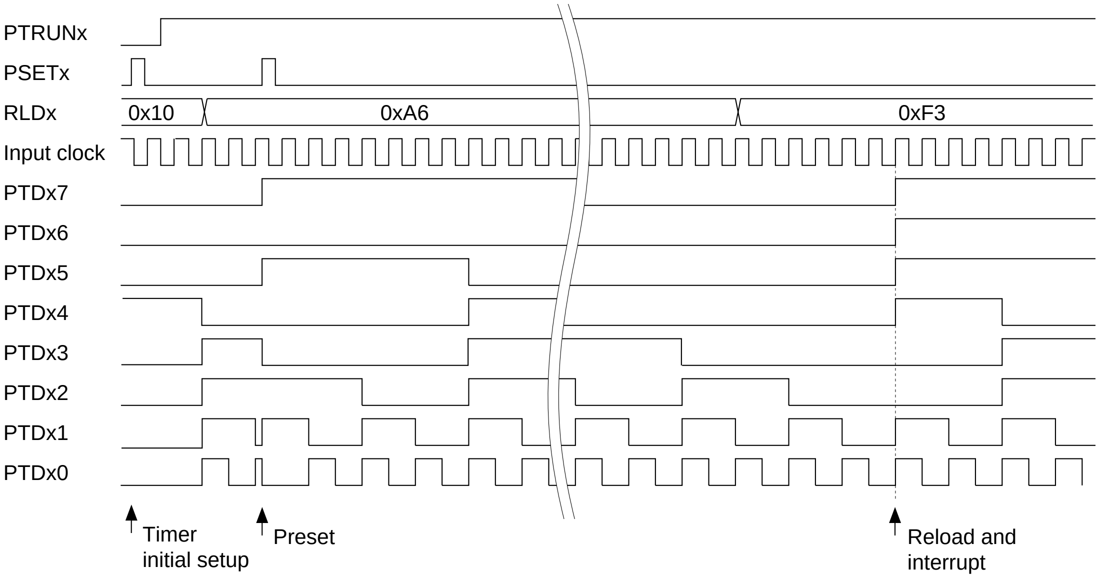


Figure 3.2 Basic Operation Timing of Counter

Reading out counter data

The counter data is read out via a PTDx data buffer. The counter data can be read out at any time.

Timer 0 data: PTD0[7:0] (D[7:0]) / 8-bit timer 0 counter data register (0x40162)
Timer 1 data: PTD1[7:0] (D[7:0]) / 8-bit timer 1 counter data register (0x40166)
Timer 2 data: PTD2[7:0] (D[7:0]) / 8-bit timer 2 counter data register (0x4016A)
Timer 3 data: PTD3[7:0] (D[7:0]) / 8-bit timer 3 counter data register (0x4016E)

Control of Clock Output

When outputting an underflow signal of the 8-bit programmable timer to external devices, or when supplying a clock generated by the underflow signal to the serial interface, it is necessary to control the clock output of the timer.
Timer 0 clock output control: PTOUT0 (D2) / 8-bit timer 0 control register (0x40160)
Timer 1 clock output control: PTOUT1 (D2) / 8-bit timer 1 control register (0x40164)
Timer 2 clock output control: PTOUT2 (D2) / 8-bit timer 2 control register (0x40168)
Timer 3 clock output control: PTOUT3 (D2) / 8-bit timer 3 control register (0x4016C)

To output the underflow signal/clock, write "1" to PTOUTx. If an output pin has been set, the underflow signal is output from that pin.
The same applies when timer 2 or 3 has been set as the clock source of the serial interface. A clock generated from the underflow signal by dividing it by 2 is output to the serial interface through this control. The clock output is turned off by writing "0" to PTOUTx, and the external output is fixed at "0" and the internal clock output is fixed at "1".
Figure 3.3 shows the waveforms of the output signals.

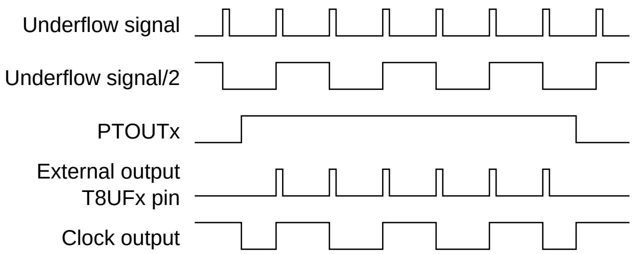


Figure 3.3 8-Bit Programmable Timer Output Waveform

The underflow signal's pulse width (duration of the high period) is equal to that of the timer's input clock (prescaler's output).

8-bit timer external output (P10–P13 ports)

1) After an initial reset (cold start), the ports (P10–P13) are set to debug signal putput ports.
2) The port (P10–P13) outputs "0" when it is set to the 8-bit timer output (timer output is off status).
3) The timer output is left as "0" when the timer output is turned on after setting the input clock and timer initial value.
4) When an underflow occurs after starting the timer, the port outputs a pulse with the same width as the 8-bit timer input clock pulse (prescaler's output).

8-Bit Programmable Timer Interrupts and DMA

The 8-bit programmable timer has a function to generate an interrupt based on the underflow state of each timer. The timing at which an interrupt is generated is shown in Figure 3.2 in the preceding section.

Control registers of the interrupt controller

Table 3.3 shows the interrupt controller's control register provided for each timer.

Table 3.3 Control Registers of Interrupt Controller

Timer	Interrupt factor flag	Interrupt enable register	Interrupt priority register
Timer 0	F8TU0(D0/0x40285)	E8TU0(D0/0x40275)	P8TM[2:0](D[2:0]/0x40269)
Timer 1	F8TU1(D1/0x40285)	E8TU1(D1/0x40275)	
Timer 2	F8TU2(D2/0x40285)	E8TU2(D2/0x40275)	
Timer 3	F8TU3(D3/0x40285)	E8TU3(D3/0x40275)	

When the timer underflows, the corresponding interrupt factor flag is set to "1". If the interrupt enable register bit corresponding to that interrupt factor flag has been set to "1", an interrupt request is generated.
An interrupt caused by a timer can be disabled by leaving the interrupt enable register bit for that timer set to "0". The interrupt factor flag is set to "1" whenever the timer underflows, regardless of how the interrupt enable register is set (even when it is set to "0").
The interrupt priority register sets an interrupt priority level (0 to 7) for the four timers as one interrupt source. Within 8-bit programmable timers, timer 0 has the highest priority and timer 3 the lowest. An interrupt request to the CPU is accepted on the condition that no other interrupt request of a higher priority has been generated.
It is only when the PSR's IE bit = "1" (interrupts enabled) and the set value of the IL is smaller than the timer interrupt level set by the interrupt priority register, that a timer interrupt request is actually accepted by the CPU.
For details on these interrupt control registers and device operation when an interrupt has occurred, refer to "ITC (Interrupt Controller)".

Intelligent DMA

The underflow interrupt factor of each timer can invoke intelligent DMA (IDMA). This enables memory-to-memory DMA transfers to be performed cyclically.
The following shows the IDMA channel numbers set to each timer:

	IDMA channel
Timer 0:	0x13
Timer 1:	0x14
Timer 2:	0x15
Timer 3:	0x16

For IDMA to be invoked, the IDMA request and IDMA enable bits shown in Table 3.4 must be set to "1" in advance. Transfer conditions, etc. must also be set on the IDMA side in advance.

Table 3.4 Control Bits for IDMA Transfer

Timer	IDMA request bit	IDMA enable bit
Timer 0	R8TU0(D2/0x40292)	DE8TU0(D2/0x40296)
Timer 1	R8TU1(D3/0x40292)	DE8TU1(D3/0x40296)
Timer 2	R8TU2(D4/0x40292)	DE8TU2(D4/0x40296)
Timer 3	R8TU3(D5/0x40292)	DE8TU3(D5/0x40296)

If the IDMA request and enable bits are set to "1", IDMA is invoked through generation of an interrupt factor. No interrupt request is generated at that point. An interrupt request is generated after the DMA transfer is completed. The registers can also be set so as not to generate an interrupt, with only a DMA transfer performed.
For details on IDMA transfers and interrupt control upon completion of IDMA transfer, refer to "IDMA (Intelligent DMA)".

High-speed DMA

The underflow interrupt factor of each timer can also invoke high-speed DMA (HSDMA).

The following shows the HSDMA channel number and trigger set-up bit corresponding to each timer:

Table 3.5 HSDMA Trigger Set-up Bits

Timer	HSDMA channel	Trigger set-up bits
Timer 0	0	HSD0S[3:0] (D[3:0]) / HSDMA Ch.0/1 trigger set-up register (0x40298)
Timer 1	1	HSD1S[3:0] (D[7:4]) / HSDMA Ch.0/1 trigger set-up register (0x40298)
Timer 2	2	HSD2S[3:0] (D[3:0]) / HSDMA Ch.2/3 trigger set-up register (0x40299)
Timer 3	3	HSD3S[3:0] (D[7:4]) / HSDMA Ch.2/3 trigger set-up register (0x40299)

For HSDMA to be invoked, the trigger set-up bits should be set to "0101" in advance. Transfer conditions, etc. must also be set on the HSDMA side.
If the 8-bit timer is selected as the HSDMA trigger, the HSDMA channel is invoked through generation of the interrupt factor.
For details on HSDMA transfer, refer to "HSDMA (High-Speed DMA)".

Trap vectors

The trap vector addresses for individual underflow interrupt factors are set by default as shown below:

	(BTA3 = high)	(BTA3 = low)
Timer 0 underflow interrupt:	0x00800D0	0x0C000D0
Timer 1 underflow interrupt:	0x00800D4	0x0C000D4
Timer 2 underflow interrupt:	0x00800D8	0x0C000D8
Timer 3 underflow interrupt:	0x00800DC	0x0C000DC

The base address of the trap table can be changed using the TTBR register (0x48134 to 0x48137).

I/O Memory of 8-Bit Programmable Timers

Table 3.6 shows the control bits of the 8-bit programmable timers.
For details on the I/O memory of the prescaler used to set a clock, refer to "Prescaler".

Table 3.6 Control Bits of 8-Bit Programmable Timer

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer 0	**0040160**	D7–3	**–**	reserved	–				–	–	0 when being read.
control register	**(B)**	D2	**PTOUT0**	8-bit timer 0 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET0**	8-bit timer 0 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN0**	8-bit timer 0 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 0	**0040161**	D7	**RLD07**	8-bit timer 0 reload data	0 to 255				X	R/W	
reload data	**(B)**	D6	**RLD06**	RLD07 = MSB					X		
register		D5	**RLD05**	RLD00 = LSB					X		
		D4	**RLD04**						X		
		D3	**RLD03**						X		
		D2	**RLD02**						X		
		D1	**RLD01**						X		
		D0	**RLD00**						X		
8-bit timer 0	**0040162**	D7	**PTD07**	8-bit timer 0 counter data	0 to 255				X	R	
counter data	**(B)**	D6	**PTD06**	PTD07 = MSB					X		
register		D5	**PTD05**	PTD00 = LSB					X		
		D4	**PTD04**						X		
		D3	**PTD03**						X		
		D2	**PTD02**						X		
		D1	**PTD01**						X		
		D0	**PTD00**						X		
8-bit timer 1	**0040164**	D7–3	**–**	reserved	–				–	–	0 when being read.
control register	**(B)**	D2	**PTOUT1**	8-bit timer 1 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET1**	8-bit timer 1 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN1**	8-bit timer 1 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 1	**0040165**	D7	**RLD17**	8-bit timer 1 reload data	0 to 255				X	R/W	
reload data	**(B)**	D6	**RLD16**	RLD17 = MSB					X		
register		D5	**RLD15**	RLD10 = LSB					X		
		D4	**RLD14**						X		
		D3	**RLD13**						X		
		D2	**RLD12**						X		
		D1	**RLD11**						X		
		D0	**RLD10**						X		
8-bit timer 1	**0040166**	D7	**PTD17**	8-bit timer 1 counter data	0 to 255				X	R	
counter data	**(B)**	D6	**PTD16**	PTD17 = MSB					X		
register		D5	**PTD15**	PTD10 = LSB					X		
		D4	**PTD14**						X		
		D3	**PTD13**						X		
		D2	**PTD12**						X		
		D1	**PTD11**						X		
		D0	**PTD10**						X		
8-bit timer 2	**0040168**	D7–3	**–**	reserved	–				–	–	0 when being read.
control register	**(B)**	D2	**PTOUT2**	8-bit timer 2 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET2**	8-bit timer 2 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN2**	8-bit timer 2 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 2	**0040169**	D7	**RLD27**	8-bit timer 2 reload data	0 to 255				X	R/W	
reload data	**(B)**	D6	**RLD26**	RLD27 = MSB					X		
register		D5	**RLD25**	RLD20 = LSB					X		
		D4	**RLD24**						X		
		D3	**RLD23**						X		
		D2	**RLD22**						X		
		D1	**RLD21**						X		
		D0	**RLD20**						X		
8-bit timer 2	**004016A**	D7	**PTD27**	8-bit timer 2 counter data	0 to 255				X	R	
counter data	**(B)**	D6	**PTD26**	PTD27 = MSB					X		
register		D5	**PTD25**	PTD20 = LSB					X		
		D4	**PTD24**						X		
		D3	**PTD23**						X		
		D2	**PTD22**						X		
		D1	**PTD21**						X		
		D0	**PTD20**						X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer 3 control register	**004016C (B)**	D7–3	–	reserved	–				–	–	0 when being read.
		D2	**PTOUT3**	8-bit timer 3 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET3**	8-bit timer 3 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN3**	8-bit timer 3 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 3 reload data register	**004016D (B)**	D7	**RLD37**	8-bit timer 3 reload data	0 to 255				X	R/W	
		D6	**RLD36**	RLD37 = MSB					X		
		D5	**RLD35**	RLD30 = LSB					X		
		D4	**RLD34**						X		
		D3	**RLD33**						X		
		D2	**RLD32**						X		
		D1	**RLD31**						X		
		D0	**RLD30**						X		
8-bit timer 3 counter data register	**004016E (B)**	D7	**PTD37**	8-bit timer 3 counter data	0 to 255				X	R	
		D6	**PTD36**	PTD37 = MSB					X		
		D5	**PTD35**	PTD30 = LSB					X		
		D4	**PTD34**						X		
		D3	**PTD33**						X		
		D2	**PTD32**						X		
		D1	**PTD31**						X		
		D0	**PTD30**						X		
8-bit timer, serial I/F Ch.0 interrupt priority register	**0040269 (B)**	D7	–	reserved	–				–	–	0 when being read.
		D6	**PSIO02**	Serial interface Ch.0	0 to 7				X	R/W	
		D5	**PSIO01**	interrupt level					X		
		D4	**PSIO00**						X		
		D3	–	reserved	–				–	–	0 when being read.
		D2	**P8TM2**	8-bit timer 0–3 interrupt level	0 to 7				X	R/W	
		D1	**P8TM1**						X		
		D0	**P8TM0**						X		
8-bit timer interrupt enable register	**0040275 (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**E8TU3**	8-bit timer 3 underflow	1	Enabled	0	Disabled	0	R/W	
		D2	**E8TU2**	8-bit timer 2 underflow					0	R/W	
		D1	**E8TU1**	8-bit timer 1 underflow					0	R/W	
		D0	**E8TU0**	8-bit timer 0 underflow					0	R/W	
8-bit timer interrupt factor flag register	**0040285 (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**F8TU3**	8-bit timer 3 underflow	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F8TU2**	8-bit timer 2 underflow					X	R/W	
		D1	**F8TU1**	8-bit timer 1 underflow					X	R/W	
		D0	**F8TU0**	8-bit timer 0 underflow					X	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register	**0040292 (B)**	D7	**RSTX0**	SIF Ch.0 transmit buffer empty	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RSRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**R8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**R8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**R8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**R8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**R16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**R16TU5**	16-bit timer 5 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register	**0040296 (B)**	D7	**DESTX0**	SIF Ch.0 transmit buffer empty	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**DE8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**DE8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**DE8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**DE8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**DE16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**DE16TU5**	16-bit timer 5 comparison B					0	R/W	
P1 function select register	**00402D4 (B)**	D7	–	reserved	–				–	–	0 when being read.
		D6	**CFP16**	P16 function selection	1	EXCL5 #DMAEND1	0	P16	0	R/W	
		D5	**CFP15**	P15 function selection	1	EXCL4 #DMAEND0	0	P15	0	R/W	
		D4	**CFP14**	P14 function selection	1	FOSC1	0	P14	0	R/W	Extended functions (0x402DF)
		D3	**CFP13**	P13 function selection	1	EXCL3 T8UF3	0	P13	0	R/W	
		D2	**CFP12**	P12 function selection	1	EXCL2 T8UF2	0	P12	0	R/W	
		D1	**CFP11**	P11 function selection	1	EXCL1 T8UF1	0	P11	0	R/W	
		D0	**CFP10**	P10 function selection	1	EXCL0 T8UF0	0	P10	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
P1 I/O control register	**00402D6 (B)**	D7	–	reserved	–				–	–	0 when being read.
		D6	**IOC16**	P16 I/O control	1	Output	0	Input	0	R/W	
		D5	**IOC15**	P15 I/O control					0	R/W	
		D4	**IOC14**	P14 I/O control					0	R/W	
		D3	**IOC13**	P13 I/O control					0	R/W	
		D2	**IOC12**	P12 I/O control					0	R/W	
		D1	**IOC11**	P11 I/O control					0	R/W	
		D0	**IOC10**	P10 I/O control					0	R/W	
Port function extension register	**00402DF (B)**	D7	**CFEX7**	P07 port extended function	1	#DMAEND3	0	P07, etc.	0	R/W	
		D6	**CFEX6**	P06 port extended function	1	#DMAACK3	0	P06, etc.	0	R/W	
		D5	**CFEX5**	P05 port extended function	1	#DMAEND2	0	P05, etc.	0	R/W	
		D4	**CFEX4**	P04 port extended function	1	#DMAACK2	0	P04, etc.	0	R/W	
		D3	**CFEX3**	P31 port extended function	1	#GARD	0	P31, etc.	0	R/W	
		D2	**CFEX2**	P21 port extended function	1	#GAAS	0	P21, etc.	0	R/W	
		D1	**CFEX1**	P10, P11, P13 port extended function	1	DST0 DST1 DPC0	0	P10, etc. P11, etc. P13, etc.	1	R/W	
		D0	**CFEX0**	P12, P14 port extended function	1	DST2 DCLK	0	P12, etc. P14, etc.	1	R/W	

CFP13–CFP10: P1[3:0] pin function selection (D[3:0]) / P1 function select register (0x402D4)

Selects the pin that is used to output a timer underflow signal to external devices.

Write "1": Underflow signal output pin
Write "0": I/O port pin
Read: Valid

Select the pin used to output a timer underflow signal to external devices from among P10 through P13 by writing "1" to the corresponding bit, CFP10 through CFP13. P10 through P13 correspond to timers 0 through 3, respectively. If "0" is written to CFP1x, the pin is set for an I/O port.
At cold start, CFP1x is set to "0" (I/O port). At hot start, the bit retains its state from prior to the initial reset.

IOC13–IOC10: P1[3:0] port I/O control (D[3:0]) / P1 I/O control register (0x402D6)

Sets input or output mode for P10 through P13.

Write "1": Output mode
Write "0": Input mode
Read: Valid

If a pin chosen from among P10 through P13 is used to output an underflow signal, write "1" to the corresponding I/O control bit to set it to output mode. If the pin is set to input mode, even if its CFP1x is set to "1", it functions as the event counter input pin of a 16-bit programmable timer cannot be used to output a timer underflow signal.
At cold start, IOC1x is set to "0" (input mode). At hot start, the bit retains its state from prior to the initial reset.

CFEX1: P10, P11, P13 port extended function (D1) / Port function extension register (0x402DF)
CFEX0: P12, P14 port extended function (D0) / Port function extension register (0x402DF)

Sets whether the function of an I/O-port pin is to be extended.

Write "1": Function-extended pin
Write "0": I/O-port/peripheral-circuit pin
Read: Valid

When CFEX[1:0] is set to "1", the P13–P10 ports function as debug signal output ports. When CFEX[1:0] = "0", the CFP1[3:0] bit becomes effective, so the settings of these bits determine whether the P13–P10 ports function as I/O port s or timer underflow signal output ports.
At cold start, CFEX[1:0] is set to "1" (function-extended pins). At hot start, CFEX[1:0] retains its state from prior to the initial reset.

RLD07–RLD00: Timer 0 reload data (D[7:0]) / 8-bit timer 0 reload data register (0x40161)
RLD17–RLD10: Timer 1 reload data (D[7:0]) / 8-bit timer 1 reload data register (0x40165)
RLD27–RLD20: Timer 2 reload data (D[7:0]) / 8-bit timer 2 reload data register (0x40169)
RLD37–RLD30: Timer 3 reload data (D[7:0]) / 8-bit timer 3 reload data register (0x4016D)

Set the initial counter value of each timer.
The reload data set in this register is loaded into each counter, and the counter starts counting down beginning with this data, which is used as the initial count.
There are two cases in which the reload data is loaded into the counter: when data is preset after "1" is written to PSETx, or when data is automatically reloaded upon counter underflow.
At initial reset, RLD is not initialized.

PTD07–PTD00: Timer 0 counter data (D[7:0]) / 8-bit timer 0 counter data (0x40162)
PTD17–PTD10: Timer 1 counter data (D[7:0]) / 8-bit timer 1 counter data (0x40166)
PTD27–PTD20: Timer 2 counter data (D[7:0]) / 8-bit timer 2 counter data (0x4016A)
PTD37–PTD30: Timer 3 counter data (D[7:0]) / 8-bit timer 3 counter data (0x4016E)

The 8-bit programmable timer data can be read out from these bits.
These bits function as buffers that retain the counter data when read out, enabling the data to be read out at any time.
At initial reset, PTD is not initialized.

PSET0: Timer 0 preset (D1) / 8-bit timer 0 control register (0x40160)
PSET1: Timer 1 preset (D1) / 8-bit timer 1 control register (0x40164)
PSET2: Timer 2 preset (D1) / 8-bit timer 2 control register (0x40168)
PSET3: Timer 3 preset (D1) / 8-bit timer 3 control register (0x4016C)

Preset the reload data in the counter.

Write "1": Preset
Write "0": Invalid
Read: Always "0"

The reload data of RLDx is preset in the counter of timer x by writing "1" to PSETx. If the counter is preset when in a RUN state, the counter starts counting immediately after the reload data is preset.
If the counter is preset when in a STOP state, the reload data that has been preset is retained.
Writing "0" results in No Operation.
Since PSETx is a write-only bit, its content when read is always "0".

PTRUN0: Timer 0 RUN/STOP control (D0) / 8-bit timer 0 control register (0x40160)
PTRUN1: Timer 1 RUN/STOP control (D0) / 8-bit timer 1 control register (0x40164)
PTRUN2: Timer 2 RUN/STOP control (D0) / 8-bit timer 2 control register (0x40168)
PTRUN3: Timer 3 RUN/STOP control (D0) / 8-bit timer 3 control register (0x4016C)

Controls the counter's RUN/STOP states.

Write "1": RUN
Write "0": STOP
Read: Valid

The counter of each timer starts counting down when "1" written to PTRUNx, and stops counting when "0" is written.
While in a STOP state, the counter retains its count until it is preset with reload data or placed in a RUN state. When the state is changed from STOP to RUN, the counter can restart counting beginning with the retained count.
At initial reset, PTRUNx is set to "0" (STOP).

PTOUT0: Timer 0 clock output control register (D2) / 8-bit timer 0 control register (0x40160)
PTOUT1: Timer 1 clock output control register (D2) / 8-bit timer 1 control register (0x40164)
PTOUT2: Timer 2 clock output control register (D2) / 8-bit timer 2 control register (0x40168)
PTOUT3: Timer 3 clock output control register (D2) / 8-bit timer 3 control register (0x4016C)

Controls the clock output of each timer.

Write "1": On
Write "0": Off
Read: Valid

The underflow signal of timer x is output from the external output pin set by CFP1x by writing "1" to PTOUTx.
When using timer 2 or 3 as the clock source of the serial interface, a clock generated from the underflow signal by dividing it by 2 is output to the corresponding channel of the serial interface.
The clock output is turned off by writing "0" to PTOUT, and the external output is fixed at "0" and the internal clock output is fixed at "1".
At initial reset, PTOUT is set to "0" (off).

P8TM2–P8TM0: 8-bit timer interrupt level (D[2:0]) / 8-bit timer, serial I/F Ch.0 interrupt priority register (0x40269)

Set the priority level of the 8-bit programmable timer interrupt in the range of 0 to 7.
At initial reset, the content of the P8TM register becomes indeterminate.

E8TU0: Timer 0 interrupt enable (D0) / 8-bit timer interrupt enable register (0x40275)
E8TU1: Timer 1 interrupt enable (D1) / 8-bit timer interrupt enable register (0x40275)
E8TU2: Timer 2 interrupt enable (D2) / 8-bit timer interrupt enable register (0x40275)
E8TU3: Timer 3 interrupt enable (D3) / 8-bit timer interrupt enable register (0x40275)

Enables or disables generation of an interrupt to the CPU.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

E8TUx is the interrupt enable bit which controls the interrupt generated by each timer. The interrupt set to "1" by this bit is enabled, and the interrupt set to "0" by this bit is disabled.
At initial reset, E8TUx is set to "0" (interrupt disabled).

F8TU0: Timer 0 interrupt factor flag (D0) / 8-bit timer interrupt factor flag register (0x40285)
F8TU1: Timer 1 interrupt factor flag (D1) / 8-bit timer interrupt factor flag register (0x40285)
F8TU2: Timer 2 interrupt factor flag (D2) / 8-bit timer interrupt factor flag register (0x40285)
F8TU3: Timer 3 interrupt factor flag (D3) / 8-bit timer interrupt factor flag register (0x40285)

Indicates the interrupt generation status of the 8-bit programmable timer.

When read

Read "1": Interrupt factor has occurred
Read "0": No interrupt factor has occurred

When written using the reset-only method (default)

Write "1": Interrupt factor flag is reset
Write "0": Invalid

When written using the read/write method

Write "1": Interrupt flag is set
Write "0": Interrupt flag is reset

F8TUx is the interrupt factor flag corresponding to each timer. It is set to "1" when the counter underflows.
At this time, if the following conditions are met, an interrupt to the CPU is generated:
1. The corresponding interrupt enable register bit is set to "1".
2. No other interrupt request of a higher priority has been generated.
3. The IE bit of the PSR is set to "1" (interrupts enabled).
4. The value set in the corresponding interrupt priority register is higher than the interrupt level (IL) of the CPU.

When using the interrupt factor of the 8-bit programmable timer to request IDMA, note that even when the above conditions are met, no interrupt request to the CPU is generated for the interrupt factor that has occurred. If interrupts are enabled at the setting of IDMA, an interrupt is generated under the above conditions after the data transfer by IDMA is completed.

The interrupt factor flag is set to "1" whenever interrupt generation conditions are met, regardless of how the interrupt enable and interrupt priority registers are set.
If the next interrupt is to be accepted after an interrupt has occurred, it is necessary that the interrupt factor flag be reset, and that the PSR be set again (by setting the IE bit to "1" after setting the IL to a value lower than the level indicated by the interrupt priority register, or by executing the reti instruction).
The interrupt factor flag can be reset only by writing to it in the software. Note that if the PSR is set again to accept interrupts generated (or if the reti instruction is executed) without resetting the interrupt factor flag, the same interrupt occurs again. Note also that the value to be written to reset the flag is "1" when the reset-only method (RSTONLY = "1") is used, and "0" when the read/write method (RSTONLY = "0") is used.
At initial reset, the content of F8TUx becomes indeterminate, so be sure to reset it in the software.

R8TU0: Timer 0 IDMA request (D2) / 16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register (0x40292)
R8TU1: Timer 1 IDMA request (D3) / 16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register (0x40292)
R8TU2: Timer 2 IDMA request (D4) / 16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register (0x40292)
R8TU3: Timer 3 IDMA request (D5) / 16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register (0x40292)

Specifies whether IDMA is to be invoked at the occurrence of an interrupt factor.

When using the set-only method (default)

Write "1": IDMA request
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA request
Write "0": Interrupt request
Read: Valid

R8TUx is the IDMA request bit for each timer. If this bit is set to "1", IDMA can be invoked when an interrupt factor occurs, and thus programmed data transfers are performed. If the bit is set to "0", normal interrupt processing is performed and IDMA is not invoked.
For details on IDMA, refer to "IDMA (Intelligent DMA)".
At initial reset, R8TUx is set to "0" (interrupt request).

DE8TU0: Timer 0 IDMA enable (D2) / 16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register (0x40296)
DE8TU1: Timer 1 IDMA enable (D3) / 16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register (0x40296)
DE8TU2: Timer 2 IDMA enable (D4) / 16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register (0x40296)
DE8TU3: Timer 3 IDMA enable (D5) / 16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register (0x40296)

Enables IDMA transfer by means of an interrupt factor.

When using the set-only method (default)

Write "1": IDMA enabled
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA enabled
Write "0": IDMA disabled
Read: Valid

If DE8TUx is set to "1", the IDMA request by the interrupt factor is enabled. If the register bit is set to "0", the IDMA request is disabled.
After an initial reset, DE8TUx is set to "0" (IDMA disabled).

Programming Notes

(1) The 8-bit programmable timer operates only when the prescaler is operating.

(2) Do not use a clock that is faster than the CPU operating clock for the 8-bit programmable timer.

(3) When setting an input clock, make sure the 8-bit programmable timer is turned off.

(4) Since the underflow interrupt condition and the timer output status are undefined after an initial reset, the counter initial value should be set to the 8-bit timer before resetting the interrupt factor flag or turning the timer output on.

(5) After an initial reset, the interrupt factor flag (F8TUx) becomes indeterminate. To prevent generation of an unwanted interrupt or IDMA request, be sure to reset this flag in the software.

(6) To prevent another interrupt from being generated again by the same factor after an interrupt has occurred, be sure to reset the interrupt factor flag (F8TUx) before setting the PSR again or executing the reti instruction.

III-4 16-BIT PROGRAMMABLE TIMERS

Configuration of 16-Bit Programmable Timer

The Peripheral Block contains six systems of 16-bit programmable timers (timers 0 to 5). They also have an event counter function using an I/O port pin.

Note: On the following pages, each timer is identified as timer x (x = 0 to 5). The functions and control register structures of 16-bit programmable timers 0 to 5 are the same. Control bit names are assigned numerals "0" to "5" denoting timer numbers. Since explanations are common to all timers, timer numbers are represented by "x" unless it is necessary to specify a timer number.

Figure 4.1 shows the structure of one channel of the 16-bit programmable timer.

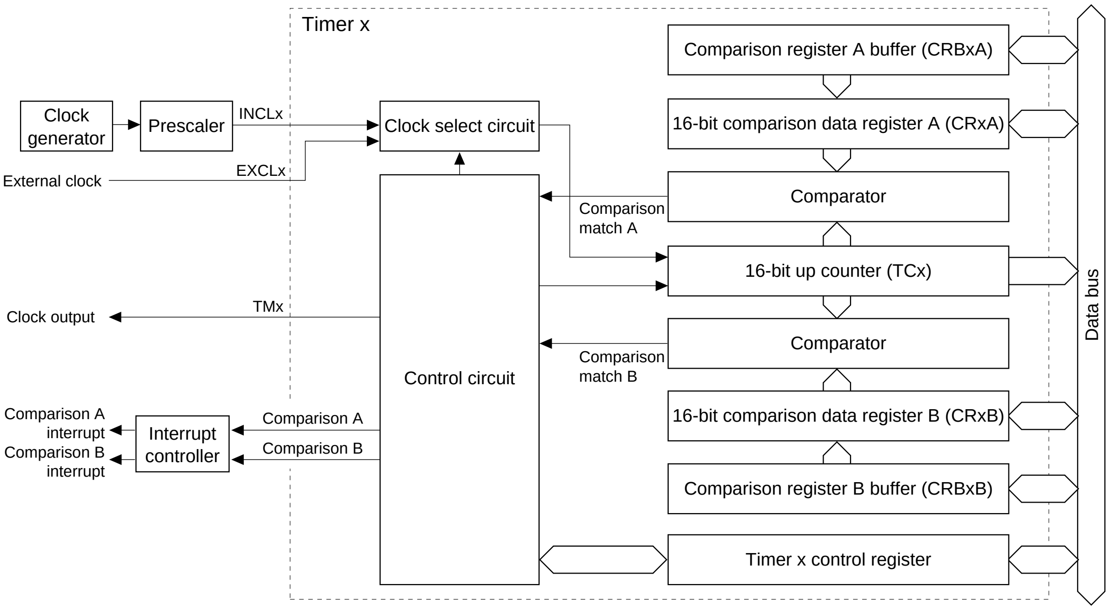


Figure 4.1 Structure of 16-Bit Programmable Timer

In each timer, a 16-bit up-counter (TCx), as well as two 16-bit comparison data registers (CRxA, CRxB) and their buffers (CRBxA, CRBxB), are provided.
The 16-bit counter can be reset to "0" by software and counts up using the prescaler output clock or an external signal input from the I/O port. The counter value can be read by software.
The comparison data registers A and B are used to store the data to be compared with the content of the up-counter. This register can be directly read and written. Furthermore, comparison data can be set via the comparison register buffer. In this case, the set value is loaded to the comparison data register when the counter is reset by the comparison match B signal or software (by writing "1" to PRESETx bit). The software can select whether comparison data is written to the comparison data register or the buffer.
When the counter value matches to the content of each comparison data register, the comparator outputs a signal that controls the interrupt and the output signal. Thus the registers allow interrupt generating intervals and the timer's output clock frequency and duty ratio to be programmed.

I/O Pins of 16-Bit Programmable Timers

Table 4.1 shows the input/output pins used for the 16-bit programmable timers.

Table 4.1 I/O Pins of 16-Bit Programmable Timer

Pin name	I/O	Function	Function select bit
P10/EXCL0/ T8UF0/DST0	I/O	I/O port / 16-bit timer 0 event counter input (I) / 8-bit timer 0 output (O) / DST0 output (Ex)	CFP10(D0)/P1 function select register(0x402D4) CFEX1(D1)/Port function extension register(0x402DF)
P11/EXCL1/ T8UF1/DST1	I/O	I/O port / 16-bit timer 1 event counter input (I) / 8-bit timer 1 output (O) / DST1 output (Ex)	CFP11(D1)/P1 function select register(0x402D4) CFEX1(D1)/Port function extension register(0x402DF)
P12/EXCL2/ T8UF2/DST2	I/O	I/O port / 16-bit timer 2 event counter input (I) / 8-bit timer 2 output (O) / DST2 output (Ex)	CFP12(D2)/P1 function select register(0x402D4) CFEX0(D0)/Port function extension register(0x402DF)
P13/EXCL3/ T8UF3/DPCO	I/O	I/O port / 16-bit timer 3 event counter input (I) / 8-bit timer 3 output (O) / DPCO output (Ex)	CFP13(D3)/P1 function select register(0x402D4) CFEX1(D1)/Port function extension register(0x402DF)
P15/EXCL4 /#DMAEND0	I/O	I/O port / 16-bit timer 4 event counter input (I) / High-speed DMA Ch.0 end signal output (O)	CFP15(D5)/P1 function select register(0x402D4)
P16/EXCL5 /#DMAEND1	I/O	I/O port / 16-bit timer 5 event counter input (I) / High-speed DMA Ch.1 end signal output (O)	CFP16(D6)/P1 function select register(0x402D4)
P22/TM0	I/O	I/O port / 16-bit timer 0 output	CFP22(D2)/P2 function select register(0x402D8)
P23/TM1	I/O	I/O port / 16-bit timer 1 output	CFP23(D3)/P2 function select register(0x402D8)
P24/TM2	I/O	I/O port / 16-bit timer 2 output	CFP24(D4)/P2 function select register(0x402D8)
P25/TM3	I/O	I/O port / 16-bit timer 3 output	CFP25(D5)/P2 function select register(0x402D8)
P26/TM4	I/O	I/O port / 16-bit timer 4 output	CFP26(D6)/P2 function select register(0x402D8)
P27/TM5	I/O	I/O port / 16-bit timer 5 output	CFP27(D7)/P2 function select register(0x402D8)

(I): Input mode, (O): Output mode, (Ex): Extended function

TMx (output pin of the 16-bit programmable timer)

This pin outputs a clock generated by the timer x.

EXCLx (event counter input pin)

When using the timer x as an event counter, input count pulses from an external source to this pin.

How to set the input/output pins of 16-bit programmable timers

All clock output pins used by the 16-bit programmable timers are shared with I/O ports. At cold start, all these pins are set for the I/O port pins P2x (function select bit CFP2x = "0"), and go into high-impedance.
When using the clock output function of the 16-bit programmable timer, select the desired timer and write "1" to the function select bit CFP2x for the corresponding pin. At hot start, these pins retain their status before from prior to the reset.

All event-counter input pins are also shared with I/O-ports. At cold start, the EXCL[3:0] pins are set for debug signal output pins (function extension bit CFEX[1:0] = "1") and the EXCL[5:4] pins are set for I/O-port pins P1[5:4] (function select bit CFP1[5:4] = "0"). When using the event counter function, select the desired timer and write "1" to the function select bit CFP1x and write "0" to the function select bit CFEXx for the corresponding pin.
Note that these pins are also shared with output pins for the 8-bit programmer timers, etc. When the input/output pins are set in input mode, they function as event counter inputs. Therefore, it is necessary to set the I/O port's I/O control bit IOC1x to "0" in advance. At cold start, these pins are set in input mode. At hot start, they retain their status from prior to the reset.

Uses of 16-Bit Programmable Timers

The up-counters of the 16-bit programmable timer cyclically output a comparison-match signal in accordance with the comparison data that are set in the software. This signal is used to generate an interrupt request to the CPU or control the internal peripheral circuits. A clock generated from the signal can also be output to external devices.

CPU interrupt request/IDMA invocation request

Each timer's comparison match (matching of counter and comparison data) can be used as an interrupt factor to generate an interrupt request to the CPU. Therefore, an interrupt can be generated at an interval that is set in the software.

Furthermore, this interrupt factor can also be used to invoke IDMA or HSDMA.

Clock output to external devices

A clock generated from the comparison-match signal can be output from the chip to the outside. The clock cycle is determined by comparison data B, and the duty ratio is determined by comparison data A. This output can be used to control external devices. The output pins of each timer are described in the preceding section.

A/D converter start trigger

The A/D converter allows a trigger to start the A/D conversion to be selected from among four available types. One is the comparison-match B of the 16-bit programmable timer 0. This makes it possible to perform the A/D conversion at programmable intervals.

To use this function, write "01" to the A/D converter control TS[1:0] (D[4:3]) / A/D trigger register (0x40242) to select the 16-bit programmable timer 0 as the trigger.

Watchdog timer

The 16-bit programmable timer 0 can be used as a watchdog timer to monitor CPU crash. In this case, the comparison-match B of this timer serves as an NMI request signal to the CPU.

To use this function, write "1" to the watchdog timer control bit EWD (D1) / Watchdog timer enable register (0x40171) to enable the NMI. For details on how to control the watchdog timer, refer to "Watchdog Timer".

Control and Operation of 16-Bit Programmable Timer

The following settings must first be made before the 16-bit programmable timer starts counting:
1. Setting pins for input/output (only when necessary)
2. Setting input clock
3. Selecting comparison data register/buffer
4. Setting clock output conditions (signal active level, fine mode)
5. Setting comparison data
6. Setting interrupt/IDMA

For details on how to set clock output conditions and interrupts and DMA, refer to "Controlling Clock Output" and "16-Bit Programmable Timer Interrupts and DMA".

Setting pin for input/output

The pin must be set for output for the output clock of the 16-bit programmable timer to be fed to external devices.

The pin for input must be set for the 16-bit programmable timer to be used as an event counter that counts external clock pulses.

For details on how to set the pin, refer to "I/O Pins of 16-Bit Programmable Timers".

Setting the input clock

The count clock for each timer can be selected from between an internal clock and an external clock. Use the following control bits to select the input clock:

Timer 0 input clock selection: CKSL0 (D3) / 16-bit timer 0 control register (0x48186)
Timer 1 input clock selection: CKSL1 (D3) / 16-bit timer 1 control register (0X4818E)
Timer 2 input clock selection: CKSL2 (D3) / 16-bit timer 2 control register (0x48196)
Timer 3 input clock selection: CKSL3 (D3) / 16-bit timer 3 control register (0x4819E)
Timer 4 input clock selection: CKSL4 (D3) / 16-bit timer 4 control register (0x481A6)
Timer 5 input clock selection: CKSL5 (D3) / 16-bit timer 5 control register (0x481AE)

An external clock is selected by writing "1" to CKSLx, and the internal clock is selected by writing "0".
At initial reset, CKSLx is set for the internal clock.
An external clock can be used for the timer for which the pin is set for input.

- **Internal clock**

When the internal clock is selected as a timer, the timer is operated by the prescaler output clock. The prescaler division ratio can be selected for each timer.

Table 4.2 Setting the Internal Clock

Timer	Control register	Division ratio select bit	Clock control bit
Timer 0	16-bit timer 0 clock control register (0x40147)	P16TS0[2:0] (D2:0])	P16TON0 (D3)
Timer 1	16-bit timer 1 clock control register (0x40148)	P16TS1[2:0] (D2:0])	P16TON1 (D3)
Timer 2	16-bit timer 2 clock control register (0x40149)	P16TS2[2:0] (D2:0])	P16TON2 (D3)
Timer 3	16-bit timer 3 clock control register (0x4014A)	P16TS3[2:0] (D2:0])	P16TON3 (D3)
Timer 4	16-bit timer 4 clock control register (0x4014B)	P16TS4[2:0] (D2:0])	P16TON4 (D3)
Timer 5	16-bit timer 5 clock control register (0x4014C)	P16TS5[2:0] (D2:0])	P16TON5 (D3)

The division ratio can be selected from among eight types as shown in Table 4.3.

Table 4.3 Input Clock Selection

P16TS = 7	P16TS = 6	P16TS = 5	P16TS = 4	P16TS = 3	P16TS = 2	P16TS = 1	P16TS = 0
f_{PSCIN}/4096	f_{PSCIN}/1024	f_{PSCIN}/256	f_{PSCIN}/64	f_{PSCIN}/16	f_{PSCIN}/4	f_{PSCIN}/2	f_{PSCIN}/1

f_{PSCIN}: Prescaler input clock frequency

The selected clock is output from the prescaler to the 16-bit programmable timer by writing "1" to P16TONx.

Notes: • When the internal clock is used, the 16-bit programmable timer operates only when the prescaler is operating (refer to "Prescaler").
• When setting an input clock, make sure the 16-bit programmable timer is turned off.

- **External clock**
  When using the timer as an event counter by supplying clock pulses from an external source, make sure the event cycle is at least the CPU operating clock period.

Selecting comparison data register/buffer

The comparison data registers A and B are used to store the data to be compared with the content of the up-counter. This register can be directly read and written. Furthermore, comparison data can be set via the comparison register buffer. In this case, the set value is loaded to the comparison data register when the counter is reset by the comparison match B signal or software (by writing "1" to PRESETx bit).
Select whether comparison data is written to the comparison data register or the buffer using the following control bits:
Timer 0 comparison register buffer enable: SELCRB0 (D5) / 16-bit timer 0 control register (0x48186)
Timer 1 comparison register buffer enable: SELCRB1 (D5) / 16-bit timer 1 control register (0x4818E)
Timer 2 comparison register buffer enable: SELCRB2 (D5) / 16-bit timer 2 control register (0x48196)
Timer 3 comparison register buffer enable: SELCRB3 (D5) / 16-bit timer 3 control register (0x4819E)
Timer 4 comparison register buffer enable: SELCRB4 (D5) / 16-bit timer 4 control register (0x481A6)
Timer 5 comparison register buffer enable: SELCRB5 (D5) / 16-bit timer 5 control register (0x481AE)

When "1" is written to SELCRBx, the comparison register buffer is selected and when "0" is written, the comparison data register is selected.
At initial reset, the comparison data register is selected.

Setting comparison data

The programmable timer contains two data comparators that allows the count data to be compared with given values. The following registers are used to set these values.
Timer 0 comparison data A: CR0A[15:0] (D[F:0]) / 16-bit timer 0 comparison data A set-up register (0x48180)
Timer 0 comparison data B: CR0B[15:0] (D[F:0]) / 16-bit timer 0 comparison data B set-up register (0x48182)
Timer 1 comparison data A: CR1A[15:0] (D[F:0]) / 16-bit timer 1 comparison data A set-up register (0x48188)
Timer 1 comparison data B: CR1B[15:0] (D[F:0]) / 16-bit timer 1 comparison data B set-up register (0x4818A)
Timer 2 comparison data A: CR2A[15:0] (D[F:0]) / 16-bit timer 2 comparison data A set-up register (0x48190)
Timer 2 comparison data B: CR2B[15:0] (D[F:0]) / 16-bit timer 2 comparison data B set-up register (0x48192)
Timer 3 comparison data A: CR3A[15:0] (D[F:0]) / 16-bit timer 3 comparison data A set-up register (0x48198)
Timer 3 comparison data B: CR3B[15:0] (D[F:0]) / 16-bit timer 3 comparison data B set-up register (0x4819A)
Timer 4 comparison data A: CR4A[15:0] (D[F:0]) / 16-bit timer 4 comparison data A set-up register (0x481A0)
Timer 4 comparison data B: CR4B[15:0] (D[F:0]) / 16-bit timer 4 comparison data B set-up register (0x481A2)
Timer 5 comparison data A: CR5A[15:0] (D[F:0]) / 16-bit timer 5 comparison data A set-up register (0x481A8)
Timer 5 comparison data B: CR5B[15:0] (D[F:0]) / 16-bit timer 5 comparison data B set-up register (0x481AA)

When SELCRBx is set to "0", these registers allow direct reading/writing from/to the comparison data register. When SELCRBx is set to "1", these registers are used to read/write from/to the comparison register buffer. The content of the buffer is loaded to the comparison data register when the counter is reset.
At initial reset, the comparison data registers/buffers are not initialized.

The programmable timer compares the comparison data register and count data and, when the two values are equal, generates a comparison match signal. This comparison match signal controls the clock output (TMx signal) to external devices, in addition to generating an interrupt.
The comparison data B is also used to reset the counter.

Resetting the counter

Each timer includes the PRESETx bit to reset the counter.
Timer 0 reset: PRESET0 (D1) / 16-bit timer 0 control register (0x48186)
Timer 1 reset: PRESET1 (D1) / 16-bit timer 1 control register (0x4818E)
Timer 2 reset: PRESET2 (D1) / 16-bit timer 2 control register (0x48196)
Timer 3 reset: PRESET3 (D1) / 16-bit timer 3 control register (0x4819E)
Timer 4 reset: PRESET4 (D1) / 16-bit timer 4 control register (0x481A6)
Timer 5 reset: PRESET5 (D1) / 16-bit timer 5 control register (0x481AE)

Normally, reset the counter before starting count-up by writing "1" to this control bit.
After the counter starts counting, it will be reset by comparison match B.

Timer RUN/STOP control

Each timer includes the PRUNx bit to control RUN/STOP.
Timer 0 RUN/STOP control: PRUN0 (D0) / 16-bit timer 0 control register (0x48186)
Timer 1 RUN/STOP control: PRUN1 (D0) / 16-bit timer 1 control register (0x4818E)
Timer 2 RUN/STOP control: PRUN2 (D0) / 16-bit timer 2 control register (0x48196)
Timer 3 RUN/STOP control: PRUN3 (D0) / 16-bit timer 3 control register (0x4819E)
Timer 4 RUN/STOP control: PRUN4 (D0) / 16-bit timer 4 control register (0x481A6)
Timer 5 RUN/STOP control: PRUN5 (D0) / 16-bit timer 5 control register (0x481AE)

The timer starts counting when "1" is written to PRUNx. The clock input is disabled and the timer stops counting when "0" is written to PRUNx.
This RUN/STOP control does not affect the counter data. Even when the timer has stopped counting, the counter retains its count so that the timer can start counting again from that point.
If the count of the counter matches the set value of the comparison data register during count-up, the timer generates a comparison match interrupt.
When the counter matches comparison data B, an interrupt is generated and the counter is reset. At the same time, the values set in the compare register buffer are loaded to the compare data register if SELCRBx is set to "1".
The counter continues counting up regardless of which interrupt has occurred. In the case of a comparison B interrupt, the counter starts counting beginning with 0.
When both the timer RUN/STOP control bit (PRUNx) and the timer reset bit (PRESETx) are set to "1" at the same time, the timer starts counting after resetting the counter.

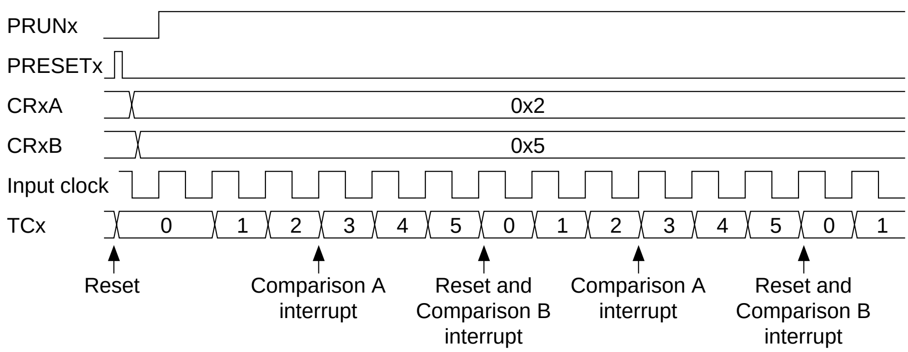


Figure 4.2 Basic Operation Timing of Counter

Reading counter data

The counter data can be read out from the following addresses shown below at any time:
Timer 0 counter data: TC0[15:0] (D[F:0]) / 16-bit timer 0 counter data register (0x48184)
Timer 1 counter data: TC1[15:0] (D[F:0]) / 16-bit timer 1 counter data register (0x4818C)
Timer 2 counter data: TC2[15:0] (D[F:0]) / 16-bit timer 0 counter data register (0x48194)
Timer 3 counter data: TC3[15:0] (D[F:0]) / 16-bit timer 1 counter data register (0x4819C)
Timer 4 counter data: TC4[15:0] (D[F:0]) / 16-bit timer 0 counter data register (0x481A4)
Timer 5 counter data: TC5[15:0] (D[F:0]) / 16-bit timer 1 counter data register (0x481AC)

Controlling Clock Output

The timers can generate a TMx signal using the comparison match signals from the counter.

Setting the signal active level

By default, an active high signal (normal low) is generated. This logic can be inverted using the OUTINVx bit. When "1" is written to the OUTINVx bit, the timer generates an active low (normal high) signal.
Timer 0 clock output inversion: OUTINV0 (D4) / 16-bit timer 0 control register (0x48186)
Timer 1 clock output inversion: OUTINV1 (D4) / 16-bit timer 1 control register (0x4818E)
Timer 2 clock output inversion: OUTINV2 (D4) / 16-bit timer 2 control register (0x48196)
Timer 3 clock output inversion: OUTINV3 (D4) / 16-bit timer 3 control register (0x4819E)
Timer 4 clock output inversion: OUTINV4 (D4) / 16-bit timer 4 control register (0x481A6)
Timer 5 clock output inversion: OUTINV5 (D4) / 16-bit timer 5 control register (0x481AE)

See Figure 4.3 for the waveforms.

Setting the output port

The TMx signal generated here can be output from the clock output pins (see Table 4.1), enabling a programmable clock to be supplied to external devices.
After a cold start, the output pins are set for the I/O ports and set in input mode. The pins go into high-impedance status.
When the pin function is switched to the timer output, the pin goes low if OUTINVx is set to "0" or goes high if OUTINVx is set to "1".

Starting clock output

To output the TMx clock, write "1" to the clock output control bit PTMx. Clock output is stopped by writing "0" to PTMx and goes to the off level according to the OUTINVx setting (low when OUTINVx = "0" or high when OUTINVx = "1").
Timer 0 clock output control: PTM0 (D2) / 16-bit timer 0 control register (0x48186)
Timer 1 clock output control: PTM1 (D2) / 16-bit timer 1 control register (0x4818E)
Timer 2 clock output control: PTM2 (D2) / 16-bit timer 2 control register (0x48196)
Timer 3 clock output control: PTM3 (D2) / 16-bit timer 3 control register (0x4819E)
Timer 4 clock output control: PTM4 (D2) / 16-bit timer 4 control register (0x481A6)
Timer 5 clock output control: PTM5 (D2) / 16-bit timer 5 control register (0x481AE)

Figure 4.3 shows the waveform of the output signal.

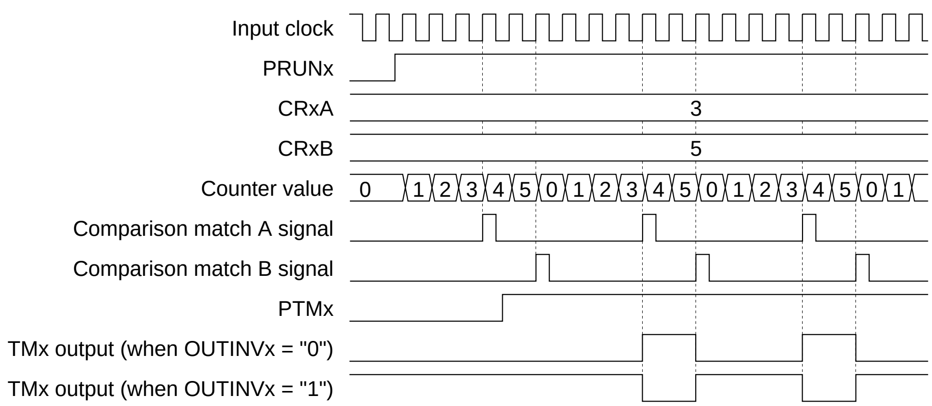


Figure 4.3 Waveform of 16-Bit Programmable Timer Output

When OUTINVx = "0" (active high):

The timer outputs a low level until the counter becomes equal to the comparison data A set in the CRxA register. When the counter is incremented to the next value from the comparison data A, the output pin goes high and a comparison A interrupt occurs. When the counter becomes equal to the comparison data B set in the CRxB register, the counter is reset and the output pin goes low. At the same time a comparison B interrupt occurs.

When OUTINVx = "1" (active low):

The timer outputs a high level until the counter becomes equal to the comparison data A set in the CRxA register. When the counter is incremented to the next value from the comparison data A, the output pin goes low and a comparison A interrupt occurs. When the counter becomes equal to the comparison data B set in the CRxB register, the counter is reset and the output pin goes high. At the same time a comparison B interrupt occurs.

Setting clock output fine mode

By default (after an initial reset), the clock output signal changes at the rising edge of the input clock when CRxA[15:0] becomes equal to TCx[15:0].
In fine mode, the output signal changes according to CRxA[0] when CRxA[15:1] becomes equal to TCx[14:0].
When CRxA[0] is "0", the output signal changes at the rising edge of the input clock.
When CRxA[0] is "1", the output signal changes at the falling edge of the input clock a half cycle from the default setting.

Example) CRxA = 3, CRxB = 5

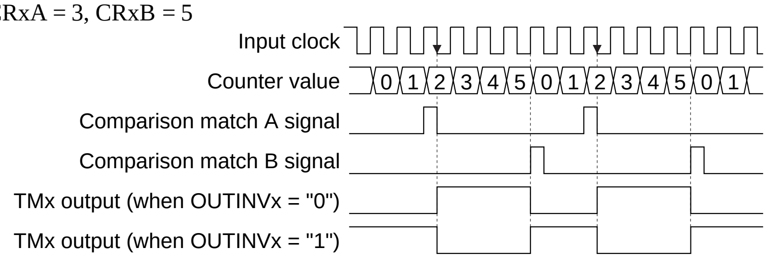


Figure 4.4 Clock Output in Fine Mode

As shown in the figure above, in fine mode the output clock duty ratio can be adjusted in the half cycle of the input clock. However, when the CRxA value is "0", the timer outputs a pulse with a 1-cycle width as the input clock, the same as the default setting.
In fine mode, the maximum value of CRxB is $2^{15} - 1 = 32{,}767$ and the range of CRxA that can be set is 0 to ($2 \times$ CRxB - 1).

The fine mode is set by the following registers:
Timer 0 fine mode selection: SELFM0 (D6) / 16-bit timer 0 control register (0x48186)
Timer 1 fine mode selection: SELFM1 (D6) / 16-bit timer 1 control register (0x4818E)
Timer 2 fine mode selection: SELFM2 (D6) / 16-bit timer 2 control register (0x48196)
Timer 3 fine mode selection: SELFM3 (D6) / 16-bit timer 3 control register (0x4819E)
Timer 4 fine mode selection: SELFM4 (D6) / 16-bit timer 4 control register (0x481A6)
Timer 5 fine mode selection: SELFM5 (D6) / 16-bit timer 5 control register (0x481AE)

When "1" is written to the SELFMx bit, fine mode is set. At initial reset, the fine mode is disabled.

Precautions

1) If a same value is set to the comparison data A and B registers, a hazard may be generated in the output signal. Therefore, do not set the comparison registers as A = B.
   There is no problem when the interrupt function only is used.

2) When using the output clock, set the comparison data registers as A ≥ 0 and B ≥ 1. The minimum settings are A = 0 and B = 1. In this case, the timer output clock cycle is the input clock × 1/2.

3) When the comparison data registers are set as A > B, no comparison A signal is generated. In this case, the output signal is fixed at the off level.

16-Bit Programmable Timer Interrupts and DMA

The 16-bit programmable timer has a function for generating an interrupt using the comparison match A and B states.
The timing at which an interrupt is generated is shown in Figure 4.2 in the preceding section.

Control registers of the interrupt controller

Table 4.4 shows the control registers of the interrupt controller provided for each timer.

Table 4.4 Control Registers of Interrupt Controller

Interrupt factor	Interrupt factor flag	Interrupt enable register	Interrupt priority register
Timer 0 comparison A	F16TC0 (D3/0x40282)	E16TC0 (D3/0x40272)	P16T0[2:0] (D[2:0]/0x40266)
Timer 0 comparison B	F16TU0 (D2/0x40282)	E16TU0 (D2/0x40272)	
Timer 1 comparison A	F16TC1 (D7/0x40282)	E16TC1 (D7/0x40272)	P16T1[2:0] (D[6:4]/0x40266)
Timer 1 comparison B	F16TU1 (D6/0x40282)	E16TU1 (D6/0x40272)	
Timer 2 comparison A	F16TC2 (D3/0x40283)	E16TC2 (D3/0x40273)	P16T2[2:0] (D[2:0]/0x40267)
Timer 2 comparison B	F16TU2 (D2/0x40283)	E16TU2 (D2/0x40273)	
Timer 3 comparison A	F16TC3 (D7/0x40283)	E16TC3 (D7/0x40273)	P16T3[2:0] (D[6:4]/0x40267)
Timer 3 comparison B	F16TU3 (D6/0x40283)	E16TU3 (D6/0x40273)	
Timer 4 comparison A	F16TC4 (D3/0x40284)	E16TC4 (D3/0x40274)	P16T4[2:0] (D[2:0]/0x40268)
Timer 4 comparison B	F16TU4 (D2/0x40284)	E16TU4 (D2/0x40274)	
Timer 5 comparison A	F16TC5 (D7/0x40284)	E16TC5 (D7/0x40274)	P16T5[2:0] (D[6:4]/0x40268)
Timer 5 comparison B	F16TU5 (D6/0x40284)	E16TU5 (D6/0x40274)	

When a comparison match state occurs in the timer, the corresponding interrupt factor flag is set to "1".
If the interrupt enable register bit corresponding to that interrupt factor flag has been set to "1", an interrupt request is generated.
An interrupt caused by a timer can be disabled by leaving the interrupt enable register bit for that timer set to "0". The interrupt factor flag is always set to "1" by the timer's comparison match state, regardless of how the interrupt enable register is set (even when set to "0").
The interrupt priority register sets an interrupt priority level (0 to 7) for each timer. Priorities within a timer block are such that timers of smaller numbers have a higher priority. Priorities between interrupt types are such that the comparison B interrupt has priority over the comparison A interrupt. An interrupt request to the CPU is accepted only when no other interrupt request of a higher priority has been generated.
It is only when the PSR's IE bit = "1" (interrupts enabled) and the set value of the IL is smaller than the timer interrupt level set by the interrupt priority register, that a timer interrupt request is actually accepted by the CPU.
For details on these interrupt control registers, as well as the device operation when an interrupt has occurred, refer to "ITC (Interrupt Controller)".

Intelligent DMA

The interrupt factor of each timer can also invoke intelligent DMA (IDMA). This allows memory-to-memory DMA transfers to be performed cyclically.
The following shows the IDMA channel numbers set for each interrupt factor of timer:

	IDMA Ch.		IDMA Ch.
Timer 0 comparison B:	0x07	Timer 0 comparison A:	0x08
Timer 1 comparison B:	0x09	Timer 1 comparison A:	0x0A
Timer 2 comparison B:	0x0B	Timer 2 comparison A:	0x0C
Timer 3 comparison B:	0x0D	Timer 3 comparison A:	0x0E
Timer 4 comparison B:	0x0F	Timer 4 comparison A:	0x10
Timer 5 comparison B:	0x11	Timer 5 comparison A:	0x12

For IDMA to be invoked, the IDMA request and IDMA enable bits shown in Table 4.5 must be set to "1" in advance. Transfer conditions, etc. must also be set on the IDMA side in advance.

Table 4.5 Control Bits for IDMA Transfer

Interrupt factor	IDMA request bit	IDMA enable bit
Timer 0 comparison A	R16TC0(D7/0x40290)	DE16TC0(D7/0x40294)
Timer 0 comparison B	R16TU0(D6/0x40290)	DE16TU0(D6/0x40294)
Timer 1 comparison A	R16TC1(D1/0x40291)	DE16TC1(D1/0x40295)
Timer 1 comparison B	R16TU1(D0/0x40291)	DE16TU1(D0/0x40295)
Timer 2 comparison A	R16TC2(D3/0x40291)	DE16TC2(D3/0x40295)
Timer 2 comparison B	R16TU2(D2/0x40291)	DE16TU2(D2/0x40295)
Timer 3 comparison A	R16TC3(D5/0x40291)	DE16TC3(D5/0x40295)
Timer 3 comparison B	R16TU3(D4/0x40291)	DE16TU3(D4/0x40295)
Timer 4 comparison A	R16TC4(D7/0x40291)	DE16TC4(D7/0x40295)
Timer 4 comparison B	R16TU4(D6/0x40291)	DE16TU4(D6/0x40295)
Timer 5 comparison A	R16TC5(D1/0x40292)	DE16TC5(D1/0x40296)
Timer 5 comparison B	R16TU5(D0/0x40292)	DE16TU5(D0/0x40296)

If the IDMA request and enable bits are set to "1", IDMA is invoked through generation of an interrupt factor. No interrupt request is generated at that point. An interrupt request is generated after the DMA transfer is completed. The registers can also be set so as not to generate an interrupt, with only a DMA transfer performed. For details on IDMA transfers and interrupt control upon completion of IDMA transfer, refer to "IDMA (Intelligent DMA)".

High-speed DMA

The interrupt factor of each timer can also invoke high-speed DMA (HSDMA).
The following shows the HSDMA channel number and trigger set-up bit corresponding to each timer:

Table 4.6 HSDMA Trigger Set-up Bits

Interrupt factor	HSDMA Ch.	Trigger set-up bits
Timer 0 comparison A	0	HSD0S[3:0] (D[3:0]) / HSDMA Ch.0/1 trigger set-up register (0x40298) = "0111"
Timer 0 comparison B	0	HSD0S[3:0] (D[3:0]) / HSDMA Ch.0/1 trigger set-up register (0x40298) = "0110"
Timer 1 comparison A	1	HSD1S[3:0] (D[7:4]) / HSDMA Ch.0/1 trigger set-up register (0x40298) = "0111"
Timer 1 comparison B	1	HSD1S[3:0] (D[7:4]) / HSDMA Ch.0/1 trigger set-up register (0x40298) = "0110"
Timer 2 comparison A	2	HSD2S[3:0] (D[3:0]) / HSDMA Ch.2/3 trigger set-up register (0x40299) = "0111"
Timer 2 comparison B	2	HSD2S[3:0] (D[3:0]) / HSDMA Ch.2/3 trigger set-up register (0x40299) = "0110"
Timer 3 comparison A	3	HSD3S[3:0] (D[7:4]) / HSDMA Ch.2/3 trigger set-up register (0x40299) = "0111"
Timer 3 comparison B	3	HSD3S[3:0] (D[7:4]) / HSDMA Ch.2/3 trigger set-up register (0x40299) = "0110"
Timer 4 comparison A	0	HSD0S[3:0] (D[3:0]) / HSDMA Ch.0/1 trigger set-up register (0x40298) = "1001"
	2	HSD2S[3:0] (D[3:0]) / HSDMA Ch.2/3 trigger set-up register (0x40299) = "1001"
Timer 4 comparison B	0	HSD0S[3:0] (D[3:0]) / HSDMA Ch.0/1 trigger set-up register (0x40298) = "1000"
	2	HSD2S[3:0] (D[3:0]) / HSDMA Ch.2/3 trigger set-up register (0x40299) = "1000"
Timer 5 comparison A	1	HSD1S[3:0] (D[7:4]) / HSDMA Ch.0/1 trigger set-up register (0x40298) = "1001"
	3	HSD3S[3:0] (D[7:4]) / HSDMA Ch.2/3 trigger set-up register (0x40299) = "1001"
Timer 5 comparison B	1	HSD1S[3:0] (D[7:4]) / HSDMA Ch.0/1 trigger set-up register (0x40298) = "1000"
	3	HSD3S[3:0] (D[7:4]) / HSDMA Ch.2/3 trigger set-up register (0x40299) = "1000"

For HSDMA to be invoked, a 16-bit timer interrupt factor should be selected using the trigger set-up bits in advance. Transfer conditions, etc. must also be set on the HSDMA side.
If a 16-bit timer is selected as the HSDMA trigger, the HSDMA channel is invoked through generation of the interrupt factor.
For details on HSDMA transfer, refer to "HSDMA (High-Speed DMA)".

Trap vectors

The trap vector addresses for each default interrupt factor are set as shown below:

	(BTA3 = high)	(BTA3 = low)
Timer 0 comparison B:	0x0080078	0x0C00078
Timer 0 comparison A:	0x008007C	0x0C0007C
Timer 1 comparison B:	0x0080088	0x0C00088
Timer 1 comparison A:	0x008008C	0x0C0008C
Timer 2 comparison B:	0x0080098	0x0C00098
Timer 2 comparison A:	0x008009C	0x0C0009C
Timer 3 comparison B:	0x00800A8	0x0C000A8
Timer 3 comparison A:	0x00800AC	0x0C000AC
Timer 4 comparison B:	0x00800B8	0x0C000B8
Timer 4 comparison A:	0x00800BC	0x0C000BC
Timer 5 comparison B:	0x00800C8	0x0C000C8
Timer 5 comparison A:	0x00800CC	0x0C000CC

The base address of the trap table can be changed using the TTBR register (0x48134 to 0x48137).

I/O Memory of 16-Bit Programmable Timers

Table 4.7 shows the control bits of the 16-bit programmable timers.
For details on the I/O memory of the prescaler used to set a clock, refer to "Prescaler".

Table 4.7 Control Bits of 16-Bit Programmable Timer

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 0/1	**0040266**	D7	–	reserved	–				–	–	0 when being read.
interrupt	**(B)**	D6	**P16T12**	16-bit timer 1 interrupt level	0 to 7				X	R/W	
priority register		D5	**P16T11**						X		
		D4	**P16T10**						X		
		D3	–	reserved	–				–	–	0 when being read.
		D2	**P16T02**	16-bit timer 0 interrupt level	0 to 7				X	R/W	
		D1	**P16T01**						X		
		D0	**P16T00**						X		
16-bit timer 2/3	**0040267**	D7	–	reserved	–				–	–	0 when being read.
interrupt	**(B)**	D6	**P16T32**	16-bit timer 3 interrupt level	0 to 7				X	R/W	
priority register		D5	**P16T31**						X		
		D4	**P16T30**						X		
		D3	–	reserved	–				–	–	0 when being read.
		D2	**P16T22**	16-bit timer 2 interrupt level	0 to 7				X	R/W	
		D1	**P16T21**						X		
		D0	**P16T20**						X		
16-bit timer 4/5	**0040268**	D7	–	reserved	–				–	–	0 when being read.
interrupt	**(B)**	D6	**P16T52**	16-bit timer 5 interrupt level	0 to 7				X	R/W	
priority register		D5	**P16T51**						X		
		D4	**P16T50**						X		
		D3	–	reserved	–				–	–	0 when being read.
		D2	**P16T42**	16-bit timer 4 interrupt level	0 to 7				X	R/W	
		D1	**P16T41**						X		
		D0	**P16T40**						X		
16-bit timer 0/1	**0040272**	D7	**E16TC1**	16-bit timer 1 comparison A	1	Enabled	0	Disabled	0	R/W	
interrupt	**(B)**	D6	**E16TU1**	16-bit timer 1 comparison B					0	R/W	
enable register		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**E16TC0**	16-bit timer 0 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU0**	16-bit timer 0 comparison B					0	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.
16-bit timer 2/3	**0040273**	D7	**E16TC3**	16-bit timer 3 comparison A	1	Enabled	0	Disabled	0	R/W	
interrupt	**(B)**	D6	**E16TU3**	16-bit timer 3 comparison B					0	R/W	
enable register		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**E16TC2**	16-bit timer 2 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.
16-bit timer 4/5	**0040274**	D7	**E16TC5**	16-bit timer 5 comparison A	1	Enabled	0	Disabled	0	R/W	
interrupt	**(B)**	D6	**E16TU5**	16-bit timer 5 comparison B					0	R/W	
enable register		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**E16TC4**	16-bit timer 4 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU4**	16-bit timer 4 comparison B					0	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.
16-bit timer 0/1	**0040282**	D7	**F16TC1**	16-bit timer 1 comparison A	1	Factor is	0	No factor is	X	R/W	
interrupt factor	**(B)**	D6	**F16TU1**	16-bit timer 1 comparison B		generated		generated	X	R/W	
flag register		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**F16TC0**	16-bit timer 0 comparison A	1	Factor is	0	No factor is	X	R/W	
		D2	**F16TU0**	16-bit timer 0 comparison B		generated		generated	X	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.
16-bit timer 2/3	**0040283**	D7	**F16TC3**	16-bit timer 3 comparison A	1	Factor is	0	No factor is	X	R/W	
interrupt factor	**(B)**	D6	**F16TU3**	16-bit timer 3 comparison B		generated		generated	X	R/W	
flag register		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**F16TC2**	16-bit timer 2 comparison A	1	Factor is	0	No factor is	X	R/W	
		D2	**F16TU2**	16-bit timer 2 comparison B		generated		generated	X	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.
16-bit timer 4/5	**0040284**	D7	**F16TC5**	16-bit timer 5 comparison A	1	Factor is	0	No factor is	X	R/W	
interrupt factor	**(B)**	D6	**F16TU5**	16-bit timer 5 comparison B		generated		generated	X	R/W	
flag register		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**F16TC4**	16-bit timer 4 comparison A	1	Factor is	0	No factor is	X	R/W	
		D2	**F16TU4**	16-bit timer 4 comparison B		generated		generated	X	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA request register	**0040290 (B)**	D7	**R16TC0**	16-bit timer 0 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**RHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**RHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**RP3**	Port input 3					0	R/W	
		D2	**RP2**	Port input 2					0	R/W	
		D1	**RP1**	Port input 1					0	R/W	
		D0	**RP0**	Port input 0					0	R/W	
16-bit timer 1–4 IDMA request register	**0040291 (B)**	D7	**R16TC4**	16-bit timer 4 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU4**	16-bit timer 4 comparison B					0	R/W	
		D5	**R16TC3**	16-bit timer 3 comparison A					0	R/W	
		D4	**R16TU3**	16-bit timer 3 comparison B					0	R/W	
		D3	**R16TC2**	16-bit timer 2 comparison A					0	R/W	
		D2	**R16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1	**R16TC1**	16-bit timer 1 comparison A					0	R/W	
		D0	**R16TU1**	16-bit timer 1 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register	**0040292 (B)**	D7	**RSTX0**	SIF Ch.0 transmit buffer empty	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RSRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**R8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**R8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**R8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**R8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**R16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**R16TU5**	16-bit timer 5 comparison B					0	R/W	
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA enable register	**0040294 (B)**	D7	**DE16TC0**	16-bit timer 0 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**DEHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**DEHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**DEP3**	Port input 3					0	R/W	
		D2	**DEP2**	Port input 2					0	R/W	
		D1	**DEP1**	Port input 1					0	R/W	
		D0	**DEP0**	Port input 0					0	R/W	
16-bit timer 1–4 IDMA enable register	**0040295 (B)**	D7	**DE16TC4**	16-bit timer 4 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU4**	16-bit timer 4 comparison B					0	R/W	
		D5	**DE16TC3**	16-bit timer 3 comparison A					0	R/W	
		D4	**DE16TU3**	16-bit timer 3 comparison B					0	R/W	
		D3	**DE16TC2**	16-bit timer 2 comparison A					0	R/W	
		D2	**DE16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1	**DE16TC1**	16-bit timer 1 comparison A					0	R/W	
		D0	**DE16TU1**	16-bit timer 1 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register	**0040296 (B)**	D7	**DESTX0**	SIF Ch.0 transmit buffer empty	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**DE8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**DE8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**DE8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**DE8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**DE16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**DE16TU5**	16-bit timer 5 comparison B					0	R/W	
P1 function select register	**00402D4 (B)**	D7	–	reserved	–				–	–	0 when being read.
		D6	**CFP16**	P16 function selection	1	EXCL5 #DMAEND1	0	P16	0	R/W	
		D5	**CFP15**	P15 function selection	1	EXCL4 #DMAEND0	0	P15	0	R/W	
		D4	**CFP14**	P14 function selection	1	FOSC1	0	P14	0	R/W	Extended functions (0x402DF)
		D3	**CFP13**	P13 function selection	1	EXCL3 T8UF3	0	P13	0	R/W	
		D2	**CFP12**	P12 function selection	1	EXCL2 T8UF2	0	P12	0	R/W	
		D1	**CFP11**	P11 function selection	1	EXCL1 T8UF1	0	P11	0	R/W	
		D0	**CFP10**	P10 function selection	1	EXCL0 T8UF0	0	P10	0	R/W	
P2 function select register	**00402D8 (B)**	D7	**CFP27**	P27 function selection	1	TM5	0	P27	0	R/W	
		D6	**CFP26**	P26 function selection	1	TM4	0	P26	0	R/W	
		D5	**CFP25**	P25 function selection	1	TM3	0	P25	0	R/W	
		D4	**CFP24**	P24 function selection	1	TM2	0	P24	0	R/W	
		D3	**CFP23**	P23 function selection	1	TM1	0	P23	0	R/W	
		D2	**CFP22**	P22 function selection	1	TM0	0	P22	0	R/W	
		D1	**CFP21**	P21 function selection	1	#DWE	0	P21	0	R/W	Ext. func.(0x402DF)
		D0	**CFP20**	P20 function selection	1	#DRD	0	P20	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Port function extension register	**00402DF (B)**	D7	**CFEX7**	P07 port extended function	1	#DMAEND3	0	P07, etc.	0	R/W	
		D6	**CFEX6**	P06 port extended function	1	#DMAACK3	0	P06, etc.	0	R/W	
		D5	**CFEX5**	P05 port extended function	1	#DMAEND2	0	P05, etc.	0	R/W	
		D4	**CFEX4**	P04 port extended function	1	#DMAACK2	0	P04, etc.	0	R/W	
		D3	**CFEX3**	P31 port extended function	1	#GARD	0	P31, etc.	0	R/W	
		D2	**CFEX2**	P21 port extended function	1	#GAAS	0	P21, etc.	0	R/W	
		D1	**CFEX1**	P10, P11, P13 port extended function	1	DST0 DST1 DPC0	0	P10, etc. P11, etc. P13, etc.	1	R/W	
		D0	**CFEX0**	P12, P14 port extended function	1	DST2 DCLK	0	P12, etc. P14, etc.	1	R/W	
16-bit timer 0 comparison data A set-up register	**0048180 (HW)**	DF	**CR0A15**	16-bit timer 0 comparison data A	0 to 65535				X	R/W	
		DE	**CR0A14**	CR0A15 = MSB					X		
		DD	**CR0A13**	CR0A0 = LSB					X		
		DC	**CR0A12**						X		
		DB	**CR0A11**						X		
		DA	**CR0A10**						X		
		D9	**CR0A9**						X		
		D8	**CR0A8**						X		
		D7	**CR0A7**						X		
		D6	**CR0A6**						X		
		D5	**CR0A5**						X		
		D4	**CR0A4**						X		
		D3	**CR0A3**						X		
		D2	**CR0A2**						X		
		D1	**CR0A1**						X		
		D0	**CR0A0**						X		
16-bit timer 0 comparison data B set-up register	**0048182 (HW)**	DF	**CR0B15**	16-bit timer 0 comparison data B	0 to 65535				X	R/W	
		DE	**CR0B14**	CR0B15 = MSB					X		
		DD	**CR0B13**	CR0B0 = LSB					X		
		DC	**CR0B12**						X		
		DB	**CR0B11**						X		
		DA	**CR0B10**						X		
		D9	**CR0B9**						X		
		D8	**CR0B8**						X		
		D7	**CR0B7**						X		
		D6	**CR0B6**						X		
		D5	**CR0B5**						X		
		D4	**CR0B4**						X		
		D3	**CR0B3**						X		
		D2	**CR0B2**						X		
		D1	**CR0B1**						X		
		D0	**CR0B0**						X		
16-bit timer 0 counter data register	**0048184 (HW)**	DF	**TC015**	16-bit timer 0 counter data	0 to 65535				X	R	
		DE	**TC014**	TC015 = MSB					X		
		DD	**TC013**	TC00 = LSB					X		
		DC	**TC012**						X		
		DB	**TC011**						X		
		DA	**TC010**						X		
		D9	**TC09**						X		
		D8	**TC08**						X		
		D7	**TC07**						X		
		D6	**TC06**						X		
		D5	**TC05**						X		
		D4	**TC04**						X		
		D3	**TC03**						X		
		D2	**TC02**						X		
		D1	**TC01**						X		
		D0	**TC00**						X		
16-bit timer 0 control register	**0048186 (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM0**	16-bit timer 0 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB0**	16-bit timer 0 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV0**	16-bit timer 0 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL0**	16-bit timer 0 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM0**	16-bit timer 0 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET0**	16-bit timer 0 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN0**	16-bit timer 0 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 1 comparison data A set-up register	**0048188 (HW)**	DF	**CR1A15**	16-bit timer 1 comparison data A	0 to 65535				X	R/W	
		DE	**CR1A14**	CR1A15 = MSB					X		
		DD	**CR1A13**	CR1A0 = LSB					X		
		DC	**CR1A12**						X		
		DB	**CR1A11**						X		
		DA	**CR1A10**						X		
		D9	**CR1A9**						X		
		D8	**CR1A8**						X		
		D7	**CR1A7**						X		
		D6	**CR1A6**						X		
		D5	**CR1A5**						X		
		D4	**CR1A4**						X		
		D3	**CR1A3**						X		
		D2	**CR1A2**						X		
		D1	**CR1A1**						X		
		D0	**CR1A0**						X		
16-bit timer 1 comparison data B set-up register	**004818A (HW)**	DF	**CR1B15**	16-bit timer 1 comparison data B	0 to 65535				X	R/W	
		DE	**CR1B14**	CR1B15 = MSB					X		
		DD	**CR1B13**	CR1B0 = LSB					X		
		DC	**CR1B12**						X		
		DB	**CR1B11**						X		
		DA	**CR1B10**						X		
		D9	**CR1B9**						X		
		D8	**CR1B8**						X		
		D7	**CR1B7**						X		
		D6	**CR1B6**						X		
		D5	**CR1B5**						X		
		D4	**CR1B4**						X		
		D3	**CR1B3**						X		
		D2	**CR1B2**						X		
		D1	**CR1B1**						X		
		D0	**CR1B0**						X		
16-bit timer 1 counter data register	**004818C (HW)**	DF	**TC115**	16-bit timer 1 counter data	0 to 65535				X	R	
		DE	**TC114**	TC115 = MSB					X		
		DD	**TC113**	TC10 = LSB					X		
		DC	**TC112**						X		
		DB	**TC111**						X		
		DA	**TC110**						X		
		D9	**TC19**						X		
		D8	**TC18**						X		
		D7	**TC17**						X		
		D6	**TC16**						X		
		D5	**TC15**						X		
		D4	**TC14**						X		
		D3	**TC13**						X		
		D2	**TC12**						X		
		D1	**TC11**						X		
		D0	**TC10**						X		
16-bit timer 1 control register	**004818E (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM1**	16-bit timer 1 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB1**	16-bit timer 1 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV1**	16-bit timer 1 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL1**	16-bit timer 1 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM1**	16-bit timer 1 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET1**	16-bit timer 1 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN1**	16-bit timer 1 Run/Stop control	1	Run	0	Stop	0	R/W	
16-bit timer 2 comparison data A set-up register	**0048190 (HW)**	DF	**CR2A15**	16-bit timer 2 comparison data A	0 to 65535				X	R/W	
		DE	**CR2A14**	CR2A15 = MSB					X		
		DD	**CR2A13**	CR2A0 = LSB					X		
		DC	**CR2A12**						X		
		DB	**CR2A11**						X		
		DA	**CR2A10**						X		
		D9	**CR2A9**						X		
		D8	**CR2A8**						X		
		D7	**CR2A7**						X		
		D6	**CR2A6**						X		
		D5	**CR2A5**						X		
		D4	**CR2A4**						X		
		D3	**CR2A3**						X		
		D2	**CR2A2**						X		
		D1	**CR2A1**						X		
		D0	**CR2A0**						X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 2	**0048192**	DF	**CR2B15**	16-bit timer 2 comparison data B	0 to 65535				X	R/W	
comparison	**(HW)**	DE	**CR2B14**	CR2B15 = MSB					X		
data B set-up		DD	**CR2B13**	CR2B0 = LSB					X		
register		DC	**CR2B12**						X		
		DB	**CR2B11**						X		
		DA	**CR2B10**						X		
		D9	**CR2B9**						X		
		D8	**CR2B8**						X		
		D7	**CR2B7**						X		
		D6	**CR2B6**						X		
		D5	**CR2B5**						X		
		D4	**CR2B4**						X		
		D3	**CR2B3**						X		
		D2	**CR2B2**						X		
		D1	**CR2B1**						X		
		D0	**CR2B0**						X		
16-bit timer 2	**0048194**	DF	**TC215**	16-bit timer 2 counter data	0 to 65535				X	R	
counter data	**(HW)**	DE	**TC214**	TC215 = MSB					X		
register		DD	**TC213**	TC20 = LSB					X		
		DC	**TC212**						X		
		DB	**TC211**						X		
		DA	**TC210**						X		
		D9	**TC29**						X		
		D8	**TC28**						X		
		D7	**TC27**						X		
		D6	**TC26**						X		
		D5	**TC25**						X		
		D4	**TC24**						X		
		D3	**TC23**						X		
		D2	**TC22**						X		
		D1	**TC21**						X		
		D0	**TC20**						X		
16-bit timer 2	**0048196**	D7	**–**	reserved	–				0	–	0 when being read.
control register	**(B)**	D6	**SELFM2**	16-bit timer 2 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB2**	16-bit timer 2 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV2**	16-bit timer 2 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL2**	16-bit timer 2 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM2**	16-bit timer 2 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET2**	16-bit timer 2 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN2**	16-bit timer 2 Run/Stop control	1	Run	0	Stop	0	R/W	
16-bit timer 3	**0048198**	DF	**CR3A15**	16-bit timer 3 comparison data A	0 to 65535				X	R/W	
comparison	**(HW)**	DE	**CR3A14**	CR3A15 = MSB					X		
data A set-up		DD	**CR3A13**	CR3A0 = LSB					X		
register		DC	**CR3A12**						X		
		DB	**CR3A11**						X		
		DA	**CR3A10**						X		
		D9	**CR3A9**						X		
		D8	**CR3A8**						X		
		D7	**CR3A7**						X		
		D6	**CR3A6**						X		
		D5	**CR3A5**						X		
		D4	**CR3A4**						X		
		D3	**CR3A3**						X		
		D2	**CR3A2**						X		
		D1	**CR3A1**						X		
		D0	**CR3A0**						X		
16-bit timer 3	**004819A**	DF	**CR3B15**	16-bit timer 3 comparison data B	0 to 65535				X	R/W	
comparison	**(HW)**	DE	**CR3B14**	CR3B15 = MSB					X		
data B set-up		DD	**CR3B13**	CR3B0 = LSB					X		
register		DC	**CR3B12**						X		
		DB	**CR3B11**						X		
		DA	**CR3B10**						X		
		D9	**CR3B9**						X		
		D8	**CR3B8**						X		
		D7	**CR3B7**						X		
		D6	**CR3B6**						X		
		D5	**CR3B5**						X		
		D4	**CR3B4**						X		
		D3	**CR3B3**						X		
		D2	**CR3B2**						X		
		D1	**CR3B1**						X		
		D0	**CR3B0**						X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 3 counter data register	**004819C (HW)**	DF	**TC315**	16-bit timer 3 counter data	0 to 65535				X	R	
		DE	**TC314**	TC315 = MSB					X		
		DD	**TC313**	TC30 = LSB					X		
		DC	**TC312**						X		
		DB	**TC311**						X		
		DA	**TC310**						X		
		D9	**TC39**						X		
		D8	**TC38**						X		
		D7	**TC37**						X		
		D6	**TC36**						X		
		D5	**TC35**						X		
		D4	**TC34**						X		
		D3	**TC33**						X		
		D2	**TC32**						X		
		D1	**TC31**						X		
		D0	**TC30**						X		
16-bit timer 3 control register	**004819E (B)**	D7	–	reserved	–				0	–	0 when being read.
		D6	**SELFM3**	16-bit timer 3 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB3**	16-bit timer 3 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV3**	16-bit timer 3 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL3**	16-bit timer 3 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM3**	16-bit timer 3 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET3**	16-bit timer 3 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN3**	16-bit timer 3 Run/Stop control	1	Run	0	Stop	0	R/W	
16-bit timer 4 comparison data A set-up register	**00481A0 (HW)**	DF	**CR4A15**	16-bit timer 4 comparison data A	0 to 65535				X	R/W	
		DE	**CR4A14**	CR4A15 = MSB					X		
		DD	**CR4A13**	CR4A0 = LSB					X		
		DC	**CR4A12**						X		
		DB	**CR4A11**						X		
		DA	**CR4A10**						X		
		D9	**CR4A9**						X		
		D8	**CR4A8**						X		
		D7	**CR4A7**						X		
		D6	**CR4A6**						X		
		D5	**CR4A5**						X		
		D4	**CR4A4**						X		
		D3	**CR4A3**						X		
		D2	**CR4A2**						X		
		D1	**CR4A1**						X		
		D0	**CR4A0**						X		
16-bit timer 4 comparison data B set-up register	**00481A2 (HW)**	DF	**CR4B15**	16-bit timer 4 comparison data B	0 to 65535				X	R/W	
		DE	**CR4B14**	CR4B15 = MSB					X		
		DD	**CR4B13**	CR4B0 = LSB					X		
		DC	**CR4B12**						X		
		DB	**CR4B11**						X		
		DA	**CR4B10**						X		
		D9	**CR4B9**						X		
		D8	**CR4B8**						X		
		D7	**CR4B7**						X		
		D6	**CR4B6**						X		
		D5	**CR4B5**						X		
		D4	**CR4B4**						X		
		D3	**CR4B3**						X		
		D2	**CR4B2**						X		
		D1	**CR4B1**						X		
		D0	**CR4B0**						X		
16-bit timer 4 counter data register	**00481A4 (HW)**	DF	**TC415**	16-bit timer 4 counter data	0 to 65535				X	R	
		DE	**TC414**	TC415 = MSB					X		
		DD	**TC413**	TC40 = LSB					X		
		DC	**TC412**						X		
		DB	**TC411**						X		
		DA	**TC410**						X		
		D9	**TC49**						X		
		D8	**TC48**						X		
		D7	**TC47**						X		
		D6	**TC46**						X		
		D5	**TC45**						X		
		D4	**TC44**						X		
		D3	**TC43**						X		
		D2	**TC42**						X		
		D1	**TC41**						X		
		D0	**TC40**						X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 4 control register	**00481A6 (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM4**	16-bit timer 4 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB4**	16-bit timer 4 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV4**	16-bit timer 4 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL4**	16-bit timer 4 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM4**	16-bit timer 4 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET4**	16-bit timer 4 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN4**	16-bit timer 4 Run/Stop control	1	Run	0	Stop	0	R/W	
16-bit timer 5 comparison data A set-up register	**00481A8 (HW)**	DF	**CR5A15**	16-bit timer 5 comparison data A	0 to 65535				X	R/W	
		DE	**CR5A14**	CR5A15 = MSB					X		
		DD	**CR5A13**	CR5A0 = LSB					X		
		DC	**CR5A12**						X		
		DB	**CR5A11**						X		
		DA	**CR5A10**						X		
		D9	**CR5A9**						X		
		D8	**CR5A8**						X		
		D7	**CR5A7**						X		
		D6	**CR5A6**						X		
		D5	**CR5A5**						X		
		D4	**CR5A4**						X		
		D3	**CR5A3**						X		
		D2	**CR5A2**						X		
		D1	**CR5A1**						X		
		D0	**CR5A0**						X		
16-bit timer 5 comparison data B set-up register	**00481AA (HW)**	DF	**CR5B15**	16-bit timer 5 comparison data B	0 to 65535				X	R/W	
		DE	**CR5B14**	CR5B15 = MSB					X		
		DD	**CR5B13**	CR5B0 = LSB					X		
		DC	**CR5B12**						X		
		DB	**CR5B11**						X		
		DA	**CR5B10**						X		
		D9	**CR5B9**						X		
		D8	**CR5B8**						X		
		D7	**CR5B7**						X		
		D6	**CR5B6**						X		
		D5	**CR5B5**						X		
		D4	**CR5B4**						X		
		D3	**CR5B3**						X		
		D2	**CR5B2**						X		
		D1	**CR5B1**						X		
		D0	**CR5B0**						X		
16-bit timer 5 counter data register	**00481AC (HW)**	DF	**TC515**	16-bit timer 5 counter data	0 to 65535				X	R	
		DE	**TC514**	TC515 = MSB					X		
		DD	**TC513**	TC50 = LSB					X		
		DC	**TC512**						X		
		DB	**TC511**						X		
		DA	**TC510**						X		
		D9	**TC59**						X		
		D8	**TC58**						X		
		D7	**TC57**						X		
		D6	**TC56**						X		
		D5	**TC55**						X		
		D4	**TC54**						X		
		D3	**TC53**						X		
		D2	**TC52**						X		
		D1	**TC51**						X		
		D0	**TC50**						X		
16-bit timer 5 control register	**00481AE (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM5**	16-bit timer 5 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB5**	16-bit timer 5 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV5**	16-bit timer 5 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL5**	16-bit timer 5 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM5**	16-bit timer 5 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET5**	16-bit timer 5 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN5**	16-bit timer 5 Run/Stop control	1	Run	0	Stop	0	R/W	

CFP16–CFP10: P1[6:0] pin function selection (D[6:0]) / P1 function select register (0x402D4)

Selects the pin to be used for input of an external count clock to the timer.

Write "1": Clock input pin
Write "0": I/O port pin
Read: Valid

Select clock input pins for the timers that are used as an event counter from among P10 through P16, by writing "1" to CFP10–CFP16. For the relationship between each pin and timer, refer to Table 4.1. The pin is set for an I/O port by writing "0" to CFP1x.
In addition to pin selection here, the pin to be used for clock input to the 16-bit programmable timer must be set to input mode using the I/O control register.
At cold start, CFP1x is set to "0" (I/O port). At hot start, CFP1x retains its status from prior to the initial reset.

CFP27–CFP22: P2[7:2] pin function selection (D[7:2]) / P2 function select register (0x402D8)

Selects the pin used for clock output.

Write "1": Clock output pin
Write "0": I/O port pin
Read: Valid

Select the pin to be used to output a timer-generated clock to external devices from among P22 through P27, by writing "1" to CFP22–CFP27. For the relationship between each pin and timer, refer to Table 4.1. The pin is set for an I/O port by writing "0" to CFP2x.
At cold start, CFP2x is set to "0" (I/O port). At hot start, CFP2x retains its status from prior to the initial reset.

CFEX1: P10, P11, P13 port extended function (D1) / Port function extension register (0x402DF)
CFEX0: P12, P14 port extended function (D0) / Port function extension register (0x402DF)

Sets whether the function of an I/O-port pin is to be extended.

Write "1": Function-extended pin
Write "0": I/O-port/peripheral-circuit pin
Read: Valid

When CFEX[1:0] is set to "1", the P14–P10 ports function as debug signal output ports. When CFEX[1:0] = "0", the CFP1[4:0] bit becomes effective, so the settings of these bits determine whether the P14–P10 ports function as I/O port s or external clock input ports.
At cold start, CFEX[1:0] is set to "1" (function-extended pins). At hot start, CFEX[1:0] retains its state from prior to the initial reset.

IOC16–IOC10: P1[6:0] port I/O control (D[6:0]) / P1 I/O control register (0x402D6)

Directs P10 through P16 for input or output.

Write "1": Output mode
Write "0": Input mode
Read: Valid

For the pin selected from among P10 through P16 for use for external clock input, write "0" to the corresponding I/O control bit to set it to input mode . If the pin is set to output mode, even though its CFP1x may be set to "1", it functions as the output pin of an 8-bit programmable timer and cannot be used to receive an external clock.
At cold start, all IOC1x is set to "0" (input mode). At hot start, IOC1x retains its state from prior to the initial reset.

SELFM0: Timer 0 fine mode selection (D6) / 16-bit timer 0 control register (0x48186)
SELFM1: Timer 1 fine mode selection (D6) / 16-bit timer 1 control register (0x4818E)
SELFM2: Timer 2 fine mode selection (D6) / 16-bit timer 2 control register (0x48196)
SELFM3: Timer 3 fine mode selection (D6) / 16-bit timer 3 control register (0x4819E)
SELFM4: Timer 4 fine mode selection (D6) / 16-bit timer 4 control register (0x481A6)
SELFM5: Timer 5 fine mode selection (D6) / 16-bit timer 5 control register (0x481AE)

Sets fine mode for clock output.

Write "1": Fine mode
Write "0": Normal output
Read: Valid

When SELFMx is set to "1", clock output is set in fine mode which allows adjustment of the output signal duty ratio in units of a half cycle for the input clock.
When SELFMx is set to "0", normal clock output will be performed.
At initial reset, SELCFMx is set to "0" (normal output).

SELCRB0: Timer 0 comparison register buffer enable (D5) / 16-bit timer 0 control register (0x48186)
SELCRB1: Timer 1 comparison register buffer enable (D5) / 16-bit timer 1 control register (0x4818E)
SELCRB2: Timer 2 comparison register buffer enable (D5) / 16-bit timer 2 control register (0x48196)
SELCRB3: Timer 3 comparison register buffer enable (D5) / 16-bit timer 3 control register (0x4819E)
SELCRB4: Timer 4 comparison register buffer enable (D5) / 16-bit timer 4 control register (0x481A6)
SELCRB5: Timer 5 comparison register buffer enable (D5) / 16-bit timer 5 control register (0x481AE)

Enables or disables writing to the comparison register buffer.

Write "1": Enabled
Write "0": Disabled
Read: Valid

When SELCRBx is set to "1", comparison data is read and written from/to the comparison register buffer. The content of the buffer is loaded to the comparison data register when the counter is reset by the software or the comparison B signal.
When SELCRBx is set to "0", comparison data is read and written from/to the comparison data register.
At initial reset, SELCRBx is set to "0" (disabled).

OUTINV0: Timer 0 output inversion (D4) / 16-bit timer 0 control register (0x48186)
OUTINV1: Timer 1 output inversion (D4) / 16-bit timer 1 control register (0x4818E)
OUTINV2: Timer 2 output inversion (D4) / 16-bit timer 2 control register (0x48196)
OUTINV3: Timer 3 output inversion (D4) / 16-bit timer 3 control register (0x4819E)
OUTINV4: Timer 4 output inversion (D4) / 16-bit timer 4 control register (0x481A6)
OUTINV5: Timer 5 output inversion (D4) / 16-bit timer 5 control register (0x481AE)

Selects a logic of the output signal.

Write "1": Inverted (active low)
Write "0": Normal (active high)
Read: Valid

By writing "1" to OUTINVx, an active-low signal (off level = high) is generated for the TMx output. When OUTINVx is set to "0", an active-high signal (off level = low) is generated.
At initial reset, OUTINVx is set to "0" (normal).

CKSL0: Timer 0 input clock selection (D3) / 16-bit timer 0 control register (0x48186)
CKSL1: Timer 1 input clock selection (D3) / 16-bit timer 1 control register (0x4818E)
CKSL2: Timer 2 input clock selection (D3) / 16-bit timer 2 control register (0x48196)
CKSL3: Timer 3 input clock selection (D3) / 16-bit timer 3 control register (0x4819E)
CKSL4: Timer 4 input clock selection (D3) / 16-bit timer 4 control register (0x481A6)
CKSL5: Timer 5 input clock selection (D3) / 16-bit timer 5 control register (0x481AE)

Selects the input clock of each timer.

Write "1": External clock
Write "0": Internal clock
Read: Valid

The internal clock (prescaler output) is selected for the input clock of each timer by writing "0" to CKSLx. An external clock (one that is fed from the clock input pin) is selected by writing "1", and the timer functions as an event counter. In this case, the clock input pin must be set using CFP1x before an external clock is selected here.
At initial reset, CKSLx is set to "0" (internal clock).

PTM0: Timer 0 clock output control (D2) / 16-bit timer 0 control register (0x48186)
PTM1: Timer 1 clock output control (D2) / 16-bit timer 1 control register (0x4818E)
PTM2: Timer 2 clock output control (D2) / 16-bit timer 2 control register (0x48196)
PTM3: Timer 3 clock output control (D2) / 16-bit timer 3 control register (0x4819E)
PTM4: Timer 4 clock output control (D2) / 16-bit timer 4 control register (0x481A6)
PTM5: Timer 5 clock output control (D2) / 16-bit timer 5 control register (0x481AE)

Controls the output of the TMx signal (timer output clock).

Write "1": On
Write "0": Off
Read: Valid

The TMx signal is output from the clock output pin by writing "1" to PTMx. Clock output is stopped by writing "0" to PTMx and goes to the off level according to the OUTINVx setting (low when OUTINVx = "0" or high when OUTINVx = "1"). In this case, the clock output pin must be set using CFP2x before outputting the TMx signal here.
At initial reset, PTMx is set to "0" (off).

PRESET0: Timer 0 reset (D1) / 16-bit timer 0 control register (0x48186)
PRESET1: Timer 1 reset (D1) / 16-bit timer 1 control register (0x4818E)
PRESET2: Timer 2 reset (D1) / 16-bit timer 2 control register (0x48196)
PRESET3: Timer 3 reset (D1) / 16-bit timer 3 control register (0x4819E)
PRESET4: Timer 4 reset (D1) / 16-bit timer 4 control register (0x481A6)
PRESET5: Timer 5 reset (D1) / 16-bit timer 5 control register (0x481AE)

Resets the counter.

Write "1": Reset
Write "0": Invalid
Read: Always "0"

The counter of timer x is reset by writing "1" to PRESETx.
Writing "0" results in No Operation.
Since PRESETx is a write-only bit, its content when read is always "0".

PRUN0: Timer 0 RUN/STOP control (D0) / 16-bit timer 0 control register (0x48186)
PRUN1: Timer 1 RUN/STOP control (D0) / 16-bit timer 1 control register (0x4818E)
PRUN2: Timer 2 RUN/STOP control (D0) / 16-bit timer 2 control register (0x48196)
PRUN3: Timer 3 RUN/STOP control (D0) / 16-bit timer 3 control register (0x4819E)
PRUN4: Timer 4 RUN/STOP control (D0) / 16-bit timer 4 control register (0x481A6)
PRUN5: Timer 5 RUN/STOP control (D0) / 16-bit timer 5 control register (0x481AE)

Controls the timer's RUN/STOP state.

Write "1": RUN
Write "0": STOP
Read: Valid

Each timer is made to start counting up by writing "1" to PRUNx and made to stop counting by writing "0".
In the STOP state, the counter data is retained until the timer is reset or placed in a RUN state. By changing states from STOP to RUN, the timer can restart counting beginning at the retained count.
At initial reset, PRUNx is set to "0" (STOP).

CR0A15–CR0A0: Timer 0 comparison data A (D[F:0]) / 16-bit timer 0 comparison data A set-up register (0x48180)
CR1A15–CR1A0: Timer 1 comparison data A (D[F:0]) / 16-bit timer 1 comparison data A set-up register (0x48188)
CR2A15–CR2A0: Timer 2 comparison data A (D[F:0]) / 16-bit timer 2 comparison data A set-up register (0x48190)
CR3A15–CR3A0: Timer 3 comparison data A (D[F:0]) / 16-bit timer 3 comparison data A set-up register (0x48198)
CR4A15–CR4A0: Timer 4 comparison data A (D[F:0]) / 16-bit timer 4 comparison data A set-up register (0x481A0)
CR5A15–CR5A0: Timer 5 comparison data A (D[F:0]) / 16-bit timer 5 comparison data A set-up register (0x481A8)

Sets the comparison data A of each timer.
When SELCRBx is set to "0", comparison data is directly read or writing from/to the comparison data register A.
When SELCRBx is set to "1", comparison data is read or written from/to the comparison register buffer A. The content of the buffer is loaded to the comparison data register A when the counter is reset.
The data set in this register is compared with each corresponding counter data. When the contents match, a comparison A interrupt is generated and the output signal rises (OUTINVx = "0") or falls (OUTINVx = "1"). This does not affect the counter value and count-up operation.
At initial reset, CRxA is not initialized.

CR0B15–CR0B0: Timer 0 comparison data B (D[F:0]) / 16-bit timer 0 comparison data B set-up register (0x48182)
CR1B15–CR1B0: Timer 1 comparison data B (D[F:0]) / 16-bit timer 1 comparison data B set-up register (0x4818A)
CR2B15–CR2B0: Timer 2 comparison data B (D[F:0]) / 16-bit timer 2 comparison data B set-up register (0x48192)
CR3B15–CR3B0: Timer 3 comparison data B (D[F:0]) / 16-bit timer 3 comparison data B set-up register (0x4819A)
CR4B15–CR4B0: Timer 4 comparison data B (D[F:0]) / 16-bit timer 4 comparison data B set-up register (0x481A2)
CR5B15–CR5B0: Timer 5 comparison data B (D[F:0]) / 16-bit timer 5 comparison data B set-up register (0x481AA)

Sets the comparison data B of each timer.
When SELCRBx is set to "0", comparison data is directly read or writing from/to the comparison data register B.
When SELCRBx is set to "1", comparison data is read or written from/to the comparison register buffer B. The content of the buffer is loaded to the comparison data register B when the counter is reset.
The data set in this register is compared with each corresponding counter data. When the contents match, a comparison B interrupt is generated and the output signal falls (OUTINVx = "0") or rises (OUTINVx = "1").
Furthermore, the counter is reset to "0".
At initial reset, CRxB is not initialized.

TC015–TC00: Timer 0 counter data (D[F:0]) / 16-bit timer 0 counter data register (0x48184)
TC115–TC10: Timer 1 counter data (D[F:0]) / 16-bit timer 1 counter data register (0x4818C)
TC215–TC20: Timer 2 counter data (D[F:0]) / 16-bit timer 2 counter data register (0x48194)
TC315–TC30: Timer 3 counter data (D[F:0]) / 16-bit timer 3 counter data register (0x4819C)
TC415–TC40: Timer 4 counter data (D[F:0]) / 16-bit timer 4 counter data register (0x481A4)
TC515–TC50: Timer 5 counter data (D[F:0]) / 16-bit timer 5 counter data register (0x481AC)

The counter data of each timer can be read from this register.
The data can be read out at any time.
Since TCx is a read-only register, writing to this register is ignored.
At initial reset, TCx is not initialized.

P16T02–P16T00: Timer 0 interrupt level (D[2:0]) / 16-bit timer 0/1 interrupt priority register (0x40266)
P16T12–P16T10: Timer 1 interrupt level (D[6:4]) / 16-bit timer 0/1 interrupt priority register (0x40266)
P16T22–P16T20: Timer 2 interrupt level (D[2:0]) / 16-bit timer 2/3 interrupt priority register (0x40267)
P16T32–P16T30: Timer 3 interrupt level (D[6:4]) / 16-bit timer 2/3 interrupt priority register (0x40267)
P16T42–P16T40: Timer 4 interrupt level (D[2:0]) / 16-bit timer 4/5 interrupt priority register (0x40268)
P16T52–P16T50: Timer 5 interrupt level (D[6:4]) / 16-bit timer 4/5 interrupt priority register (0x40268)

Sets the priority levels of 16-bit programmable timer interrupts.
The priority level can be set in the range of 0 to 7.
At initial reset, P16Tx becomes indeterminate.

E16TU0, E16TC0: Timer 0 interrupt enable (D2, D3) / 16-bit timer 0/1 interrupt enable register (0x40272)
E16TU1, E16TC1: Timer 1 interrupt enable (D6, D7) / 16-bit timer 0/1 interrupt enable register (0x40272)
E16TU2, E16TC2: Timer 2 interrupt enable (D2, D3) / 16-bit timer 2/3 interrupt enable register (0x40273)
E16TU3, E16TC3: Timer 3 interrupt enable (D6, D7) / 16-bit timer 2/3 interrupt enable register (0x40273)
E16TU4, E16TC4: Timer 4 interrupt enable (D2, D3) / 16-bit timer 4/5 interrupt enable register (0x40274)
E16TU5, E16TC5: Timer 5 interrupt enable (D6, D7) / 16-bit timer 4/5 interrupt enable register (0x40274)

Enables or disables the generation of an interrupt to the CPU.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

The E16TUx and E16TCx are provided for the comparison B and comparison A interrupt factors, respectively. The interrupt for which the bit is set to "1" is enabled, and the interrupt for which the bit is set to "0" is disabled.
At initial reset, these bits are set to "0" (interrupt disabled).

F16TU0, F16TC0: Timer 0 interrupt factor flag (D2, D3) / 16-bit timer 0/1 interrupt factor flag register (0x40282)
F16TU1, F16TC1: Timer 1 interrupt factor flag (D6, D7) / 16-bit timer 0/1 interrupt factor flag register (0x40282)
F16TU2, F16TC2: Timer 2 interrupt factor flag (D2, D3) / 16-bit timer 2/3 interrupt factor flag register (0x40283)
F16TU3, F16TC3: Timer 3 interrupt factor flag (D6, D7) / 16-bit timer 2/3 interrupt factor flag register (0x40283)
F16TU4, F16TC4: Timer 4 interrupt factor flag (D2, D3) / 16-bit timer 4/5 interrupt factor flag register (0x40284)
F16TU5, F16TC5: Timer 5 interrupt factor flag (D6, D7) / 16-bit timer 4/5 interrupt factor flag register (0x40284)

Indicates the status of 16-bit programmable timer interrupt generation.

When read

Read "1": Interrupt factor has occurred
Read "0": No interrupt factor has occurred

When written using the reset-only method (default)

Write "1": Interrupt factor flag is reset
Write "0": Invalid

When written using the read/write method

Write "1": Interrupt flag is set
Write "0": Interrupt flag is reset

F16TUx and F16TCx are the interrupt factor flags corresponding to the comparison B and comparison A interrupts, respectively. The flag is set to "1" when each interrupt factor occurs.
At this time, if the following conditions are met, an interrupt to the CPU is generated:
1. The corresponding interrupt enable register bit is set to "1".
2. No other interrupt request of a higher priority has been generated.
3. The PSR's IE bit is set to "1" (interrupts enabled).
4. The value set in the corresponding interrupt priority register is higher than the CPU's interrupt level (IL).
When using the interrupt factor of the 16-bit programmable timer to request IDMA, note that even when the above conditions are met, no interrupt request to the CPU is generated for the interrupt factor that has occurred. If interrupts are enabled at the setting of IDMA, an interrupt is generated under the above conditions after the data transfer by IDMA is completed.

The interrupt factor flag is set to "1" whenever interrupt generation conditions are met, regardless of how the interrupt enable and interrupt priority registers are set.
If the next interrupt is to be accepted after an interrupt has occurred, it is necessary that the interrupt factor flag be reset, and that the PSR be set again (by setting the IE bit to "1" after setting the IL to a value lower than the level indicated by the interrupt priority register, or by executing the reti instruction).
The interrupt factor flag can be reset only by writing to it in the software. Note that if the PSR is set again to accept interrupts generated (or if the reti instruction is executed) without resetting the interrupt factor flag, the same interrupt occurs again. Note also that the value to be written to reset the flag is "1" when the reset-only method (RSTONLY = "1") is used, and "0" when the read/write method (RSTONLY = "0") is used.
At initial reset, all these flags become indeterminate, so be sure to reset them in the software.

R16TU0, R16TC0: Timer 0 IDMA request (D6, D7) /
Port input 0–3, HSDMA, 16-bit timer 0 IDMA request register (0x40290)
R16TU1, R16TC1: Timer 1 IDMA request (D0, D1) / 16-bit timer 1–4 IDMA request register (0x40291)
R16TU2, R16TC2: Timer 2 IDMA request (D2, D3) / 16-bit timer 1–4 IDMA request register (0x40291)
R16TU3, R16TC3: Timer 3 IDMA request (D4, D5) / 16-bit timer 1–4 IDMA request register (0x40291)
R16TU4, R16TC4: Timer 4 IDMA request (D6, D7) / 16-bit timer 1–4 IDMA request register (0x40291)
R16TU5, R16TC5: Timer 5 IDMA request (D0, D1) /
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register (0x40292)

Specifies whether to invoke IDMA when an interrupt factor occurs.

When using the set-only method (default)
Write "1": IDMA request
Write "0": Not changed
Read: Valid

When using the read/write method
Write "1": IDMA request
Write "0": Interrupt request
Read: Valid

R16TUx and R16TCx are IDMA request bits corresponding to the comparison B and comparison A interrupt factors, respectively. When the bit is set to "1", IDMA is invoked when the interrupt factor occurs, thereby performing programmed data transfers. When the register is set to "0", normal interrupt processing is performed and IDMA is not invoked. For details on IDMA, refer to "IDMA (Intelligent DMA)".
At initial reset, these bits are set to "0" (interrupt request).

DE16TU0, DE16TC0: Timer 0 IDMA enable (D6, D7) /
Port input 0–3, HSDMA, 16-bit timer 0 IDMA enable register (0x40294)
DE16TU1, DE16TC1: Timer 1 IDMA enable (D0, D1) / 16-bit timer 1–4 IDMA enable register (0x40295)
DE16TU2, DE16TC2: Timer 2 IDMA enable (D2, D3) / 16-bit timer 1–4 IDMA enable register (0x40295)
DE16TU3, DE16TC3: Timer 3 IDMA enable (D4, D5) / 16-bit timer 1–4 IDMA enable register (0x40295)
DE16TU4, DE16TC4: Timer 4 IDMA enable (D6, D7) / 16-bit timer 1–4 IDMA enable register (0x40295)
DE16TU5, DE16TC5: Timer 5 IDMA enable (D0, D1) /
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register (0x40296)

Enables IDMA transfer by means of an interrupt factor.

When using the set-only method (default)

Write "1": IDMA enabled
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA enabled
Write "0": IDMA disabled
Read: Valid

DE16TUx and DE16TCx are IDMA enable bits corresponding to the comparison B and comparison A interrupt factors, respectively. If the bit is set to "1", the IDMA request by the interrupt factor is enabled. If the bit is set to "0", the IDMA request is disabled.
After an initial reset, these bits are set to "0" (IDMA disabled).

Programming Notes

(1) The 16-bit programmable timers clocked by the internal clock operate only when the prescaler is operating.

(2) When setting the input clock or operation mode, make sure the 16-bit programmable timer is turned off.

(3) If a same value is set to the comparison data A and B registers, a hazard may be generated in the output signal. Therefore, do not set the comparison registers as A = B.
There is no problem when the interrupt function only is used.

(4) When using the output clock, set the comparison data registers as $A \geq 0$ and $B \geq 1$. The minimum settings are A = 0 and B = 1. In this case, the timer output clock cycle is the input clock $\times$ 1/2.

(5) When the comparison data registers are set as A > B in normal mode, no comparison A interrupt is generated. In this case, the output signal is fixed at the off level.
In fine mode, no comparison A interrupt is generated when the comparison data registers are set as $A > 2 \times B + 1$.

(6) After an initial reset, the interrupt factor flag becomes indeterminate. To prevent generation of an unwanted interrupt or IDMA request, be sure to reset this flag and register in the software.

(7) To prevent another interrupt from being generated by the same factor after an interrupt has occurred, be sure to reset the interrupt factor flag before setting the PSR again or executing the reti instruction.

(8) Be aware that unnecessary pulse may be generated according to the control of the clock output and port configuration when a 16-bit programmable timer is used to output the TMx clock.
For example, when TMx is set as inverted output (OUTINVx = "1"), the output waveform falls with the comparison B signal and it rises with the comparison A signal. Furthermore, the output pin is fixed at high level when PTMx is set to "0" to stop the clock output. When switching the output pin to the I/O port pin and then setting the port to low after the TMx signal falls with the comparison A signal, a high level pulse will be generated if "0" is written to PTMx before setting the port to low. It can be prevented by writing "0" to PTMx after setting the port to low.

THIS PAGE IS BLANK.

III-5 WATCHDOG TIMER

Configuration of Watchdog Timer

The Periheral Block incorporates a watchdog timer function to detect the CPU's crash.
This function is implemented through the use of the 16-bit programmable timer 0. When this function is enabled, an NMI (nonmaskable interrupt) is generated by the comparison B signal from the 16-bit programmable timer 0 (generating intervals can be set through the use of software). The 16-bit programmable timer 0 set in the software so as not to generate the NMI, making it possible to detect a program crash that may not pass through this processing routine.
Figure 5.1 shows the block diagram of the watchdog timer.

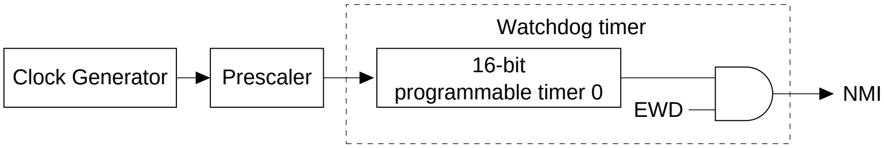


Figure 5.1 Watchdog Timer Block Diagram

Control of Watchdog Timer

Setting the operating clock and NMI generating interval

The watchdog timer is operated by the prescaler's output clock. Therefore, the watchdog timer function cannot be used when the prescaler is inactive.
The NMI is generated every time the 16-bit programmable timer 0 is reset by the comparison B setting. Therefore, this interval is determined by the prescaler's P16TS0[2:0] (D[2:0]) / 16-bit timer 0 clock control register (0x40147), and the comparison data B set in CR0B[15:0] (D[F:0]) / 16-bit timer 0 comparison register B (0x48182).

The NMI generating interval is calculated using the following equation:

$$\text{NMI generating interval} = \frac{\text{CR0B} + 1}{f_{\text{PSCIN}} \times \text{dr}} \text{ [sec.]}$$

f_{PSCIN}: Prescaler input clock frequency [Hz]
dr: Prescaler's division ratio set by the P16TS0 register (1/4096, 1/1024, 1/256, 1/64, 1/16, 1/4, 1/2, 1/1)
CR0B: Set value of the CR0B register (0 to 65,535)

For details on how to control the prescaler and the 16-bit programmable timer 0, refer to "Prescaler" and "16-Bit Programmable Timers".

Setting the watchdog timer function

To use the watchdog timer function, enable the NMI that is generated by the comparison B signal from the 16-bit programmable timer 0. For this purpose, use EWD (D1) / Watchdog timer enable register (0x40171). The NMI is enabled by writing "1" to EWD. At initial reset, EWD is set to "0", so generation of the NMI is disabled.
To prevent an unwanted NMI from being generated by erroneous writing to EWD, this register is normally write-protected. To write-enable EWD, write "1" to WRWD (D7) / Watchdog timer write-protect register (0x40170). Only one writing to EWD is enabled in this way by the WRWD bit. When data is written to EWD after it is write-enabled, the WRWD bit is reset back to "0", thus making EWD write-protected again.

For the 16-bit programmable timer 0, set an appropriate comparison B value to make it start operating.

If the watchdog timer function is not to be used, set EWD to "0" and do not change it.

Resetting the watchdog timer

When using the watchdog timer, prepare a routine to reset the 16-bit programmable timer 0 before an NMI is generated in a location where it will be periodically processed. Make sure this routine is processed within the NMI generation interval described above.
The 16-bit programmable timer 0 is reset by writing "1" to PRESET0 (D1) / 16-bit timer 0 control register (0x48186). At this point, the timer counter is set to 0, and the timer starts counting the NMI generation interval over again from that point.

If the watchdog timer is not reset within the set interval for any reason, the CPU is made to enter trap processing by an NMI and starts executing the processing routine indicated by the NMI vector.
The NMI trap vector address is set by default as follows:

When BTA3 = high, 0x008001C
When BTA3 = low, 0x0C0001C

The trap table base address can be changed using the TTBR registers (0x48134 to 0x48137).

Operation in Standby Modes

During HALT mode

In HALT mode (basic mode or HALT2 mode), the prescaler and watchdog timer are operating. Consequently, if HALT mode continues beyond the NMI generation interval, HALT mode is cleared by the NMI.
To disable the watchdog timer in HALT mode, set EWD to "0" before executing the halt instruction or turn off the 16-bit programmable timer 0.
If the NMI is disabled by EWD, the 16-bit programmable timer 0 continues counting even in HALT mode. To reenable the NMI after clearing HALT mode, reset the 16-bit programmable timer 0 in advance.
If HALT mode was entered after the 16-bit programmable timer 0 was turned off, reset the timer before restarting it.

During SLEEP mode

In SLEEP mode, the prescaler is turned off. Therefore, the watchdog timer also stops operating. To prevent generation of an unwanted NMI after clearing SLEEP mode, reset the 16-bit programmable timer 0 before executing the slp instruction. In addition, disable generation of the NMI by EWD as necessary.

I/O Memory of Watchdog Timer

Table 5.1 shows the control bits of the watchdog timer.

Table 5.1 Control Bits of Watchdog Timer

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Watchdog timer write-protect register	**0040170 (B)**	D7	**WRWD**	EWD write protection	1	Write enabled	0	Write-protect	0	R/W	
		D6–0	–	–	–				–	–	0 when being read.
Watchdog timer enable register	**0040171 (B)**	D7–2	–	–	–				–	–	0 when being read.
		D1	**EWD**	Watchdog timer enable	1	NMI enabled	0	NMI disabled	0	R/W	
		D0	–	–	–				–	–	0 when being read.

WRWD: EWD write protection (D7) / Watchdog timer write-protect register (0x40170)

Enables writing to the EWD register.

Write "1": Writing enabled
Write "0": Write-protected
Read: Valid

The EWD bit is write-protected to prevent unwanted modifications. Writing to this bit is enabled for only one writing by setting WRWD to "1". WRWD is reset back to "0" by writing to EWD, so EWD is write-protected again. If WRWD is reset to "0" when EWD is write-enabled (WRWD = "1"), EWD becomes write-protected again.
At initial reset, WRWD is set to "0" (write-protected).

EWD: NMI enable (D1) / Watchdog timer enable register (0x40171)

Controls the generation of a nonmaskable interrupt (NMI) by the watchdog timer.

Write "1": NMI is enabled
Write "0": NMI is disabled
Read: Valid

The watchdog timer's interrupt signal is masked by writing "0" to EWD, so a nonmaskable interrupt (NMI) to the CPU is not generated. If EWD is set to "1", an NMI is generated by the 16-bit programmable timer 0 comparison B signal.
Writing to EWD is valid only when WRWD = "1".
Even when EWD is set to "0", the 16-bit programmable timer 0 does not stop counting. Therefore, if the NMI has been temporarily disabled, be sure to reset the 16-bit programmable timer 0 before setting the EWD register back to "1".
At initial reset, EWD is set to "0" (NMI disabled).

Programming Notes

(1) If the watchdog timer's NMI is enabled, the watchdog timer must be reset in the software before the 16-bit programmable timer 0 outputs the comparison B signal.

(2) Even when EWD is set to "0", the 16-bit programmable timer 0 does not stop counting. Therefore, if the NMI has been temporarily disabled, be sure to reset the 16-bit programmable timer 0 before setting EWD back to "1".

THIS PAGE IS BLANK.

III-6 LOW-SPEED (OSC1) OSCILLATION CIRCUIT

Configuration of Low-Speed (OSC1) Oscillation Circuit

The Peripheral Block has a built-in low-speed (OSC1) oscillation circuit.
The low-speed (OSC1) oscillation circuit generates a 32.768-kHz (Typ.) subclock.
The OSC1 clock output by this circuit is delivered to the CLG (clock generator) in the Core Block and is used as the source clock for the clock timer. It can also be used as a sub-clock for the low-speed (low-power) operation of the CPU and peripheral circuits (switchable in a program).
Figure 6.1 shows the configuration of the clock system.

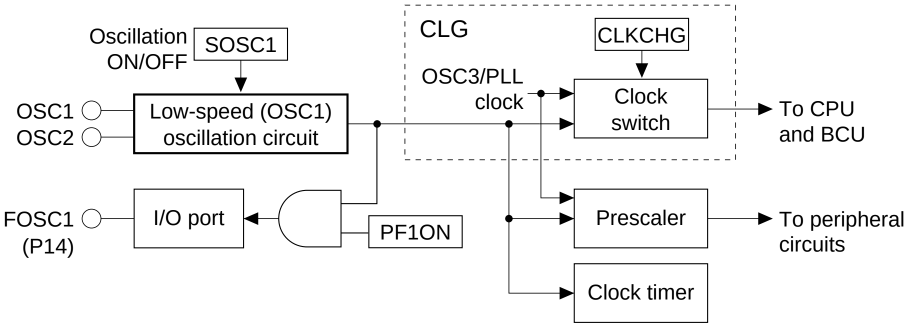


Figure 6.1 Configuration of Clock System

The CPU operating clock can be switched to the output (OSC1 clock) of the low-speed (OSC1) oscillation circuit in a program. Furthermore, the oscillation circuit can be stopped in a program.
If the OSC3 clock is unnecessary such as when performing clock processing only, set the OSC1 clock for operation of the CPU/peripheral circuits and turn off the high-speed (OSC3) oscillation circuit in order to reduce current consumption.
The low-speed (OSC1) oscillation circuit does not stop in SLEEP mode.
For the control method when using the OSC1 clock for the operating clock of the peripheral circuits, refer to "Prescaler".

I/O Pins of Low-Speed (OSC1) Oscillation Circuit

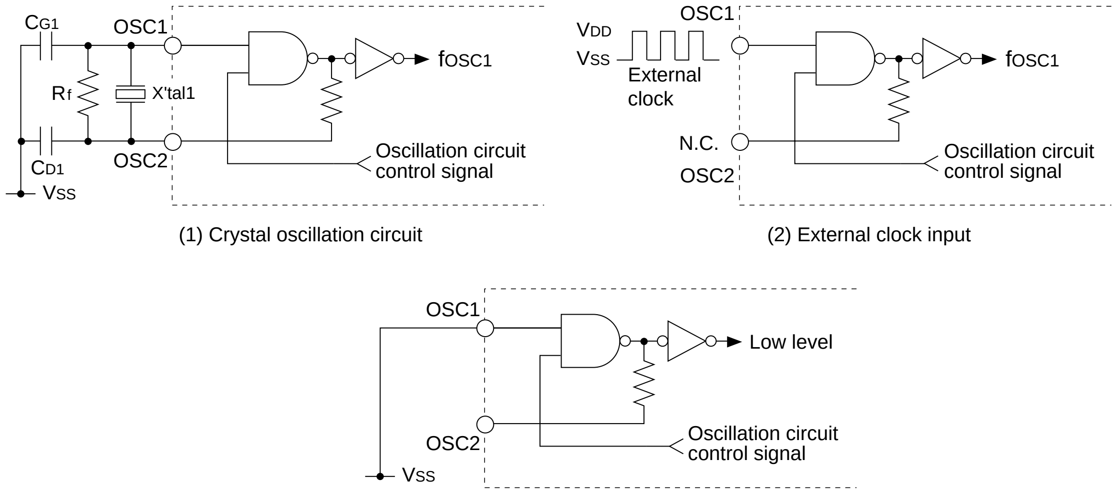
Table 6.1 lists the I/O pins of the low-speed (OSC1) oscillation circuit.

Table 6.1 I/O Pins of Low-Speed (OSC1) Oscillation Circuit

Pin name	I/O	Function
OSC1	I	Low-speed (OSC1) oscillation input pin Crystal oscillation or external clock input
OSC2	O	Low-speed (OSC1) oscillation output pin Crystal oscillation (open when external clock is used)
P14/FOSC1/DCLK	O	I/O port / Low-speed (OSC1) oscillation clock output / DCLK signal output

Oscillator Types

In the low-speed (OSC1) oscillation circuit, either a crystal oscillation or an external clock input can be selected as the type of oscillation circuit.
Figure 6.2 shows the structure of the low-speed (OSC1) oscillation circuit.



(1) Crystal oscillation circuit
(2) External clock input
(3) When not used

Figure 6.2 Low-Speed (OSC1) Oscillation Circuit

When using a crystal oscillation for this circuit, connect a crystal resonator X'tal1 (32.768 kHz, Typ.) and feedback resistor (Rf) between the OSC1 and OSC2 pins, and two capacitors (C_{G1}, C_{D1}) between the OSC1 pin and V_{SS} and the OSC2 pin and V_{SS}, respectively.
When an external clock source is used, leave the OSC2 pin open and input a square-wave clock to the OSC1 pin.
If the low-speed (OSC1) oscillation circuit is not used, connect the OSC1 pin to V_{SS} and leave the OSC2 pin open.

The oscillation frequency is 32.768 kHz (Typ.). Use a crystal resonator or external clock that oscillates at this frequency. No other frequency can be used for clock applications.

For details on oscillation characteristics and the external clock input characteristics, refer to "Electrical Characteristics".

Controlling Oscillation

The low-speed (OSC1) oscillation circuit can be turned on or off using SOSC1 (D0) / Power control register (0x40180).
The oscillation circuit is turned off by writing "0" to SOSC1 and turned back on again by writing "1". SOSC1 is set to "1" at initial reset, so the oscillation circuit is turned on.

Notes:
- When the low-speed (OSC1) oscillation circuit is used as the clock source for the CPU operating clock, it cannot be turned off. In this case, writing "0" to SOSC1 is ignored. Note also that writing to SOSC1 is allowed only when the power-control register protection flag is set to "0b10010110".
- Immediately after the oscillation circuit is turned on, a certain period of time is required for oscillation to stabilize (3 sec max.). To prevent the device from operating erratically, do not use the clock until its oscillation has stabilized.

The low-speed (OSC1) oscillation circuit does not stop when the CPU is set in SLEEP mode.

Switching Over the CPU Operating Clock

After an initial reset, the CPU starts operating using the OSC3 clock.
In cases in which some peripheral circuits (e.g., programmable timer, serial interface, and A/D converter) that are clocked by the OSC3 clock do not need to be operate and the CPU can process its jobs at a low clock speed, the CPU operating clock can be switched to the OSC1 clock, thereby reducing current consumption. Use CLKCHG (D2) / Power control register (0x40180) to switch over the operating clock.

Procedure for switching over from the OSC3 clock to the OSC1 clock

1. Turn on the low-speed (OSC1) oscillation circuit (by writing "1" to SOSC1).
2. Wait until the OSC1 oscillation stabilizes (three seconds or more).
3. Change the CPU operating clock (by writing "0" to CLKCHG).
4. Turn off the high-speed (OSC3) oscillation circuit (by writing "0" to SOSC3).

* Steps 1 and 2 are required only when the low-speed (OSC1) oscillation circuit is inactive.

Notes:
- Use separate instructions to switch from OSC3 to OSC1 and turn the OSC3 oscillation off. If these operations are processed simultaneously using one instruction, the CPU may operate erratically.
- Make sure the operation of the peripheral circuits, such as the programmable timer, A/D converter, and serial interface, which are clocked by the OSC3 oscillation circuit, is terminated before the OSC3 oscillation is turned off in order to prevent them from operating erratically.

Procedure for switching over from the OSC1 clock to the OSC3 clock

1. Turn on the high-speed (OSC3) oscillation circuit (by writing "1" to SOSC3).
2. Wait until the OSC3 oscillation stabilizes (10 ms or more for a 3.3-V crystal resonator).
3. Switch over the CPU operating clock (by writing "1" to CLKCHG).

Note: The operating clock switchover by CLKCHG is effective only when both oscillation circuits are on and the power-control register protection flag is set to "0b10010110".

Power-Control Register Protection Flag

The power-control register (SOSC1, SOSC3, CLKCHG, CLKDT[1:0]) at address 0x40180, which is used to control the oscillation circuits and the CPU operating clock, is normally disabled against writing in order to prevent it from malfunctioning due to unnecessary writing.
To enable this register for writing, the power-control register protection flag CLGP[7:0] (D[7:0]) / Power-control protection register (0x4019E) must be set to "0b10010110". Note that this setting allows for the power-control register (0x40180) to be written to only once, so all bits of CLGP[7:0] are cleared to "0" when this address is written to. Therefore, CLGP[7:0] must be set to "0b10010110" each time the power-control register (0x40180) is written to. The flag CLGP[7:0] does not affect the readout from the power-control register (0x40180).

Operation in Standby Mode

In HALT mode, which is entered by executing the halt instruction, the low-speed (OSC1) oscillation circuits retains its status before HALT mode is entered. Under normal conditions, therefore, there is no need to control the oscillation circuit before entering or after exiting HALT mode.

The low-speed (OSC1) oscillation circuit does not stop operating in SLEEP mode set by executing the slp (sleep) instruction. Therefore, if the CPU was operating using the OSC1 clock before SLEEP mode was entered, the CPU keeps operating using the OSC1 clock in SLEEP mode.

OSC1 Clock Output to External Devices

The low-speed (OSC1) oscillation clock can be output from the FOSC1 (P14) pin to external devices.

Table 6.2 OSC1 Clock Output Pin

Pin name	I/O	Function	Function select bit
P14/FOSC1/ DCLK	O	I/O port / Low-speed (OSC1) oscillation clock output / DCLK signal output	CFP14(D4) / P1 function select register (0x402D4) CFEX0 (D0) / Port function extension register (0x402DF)

Setting the clock output pin

The pin used to output the OSC1 clock to external devices is shared with the P14 I/O port and the debug clock signal DCLK.
At cold start, it is set for the DCLK signal output (CFP14 = "0" and CFEX0 = "1"). When using the clock output function, write "1" to CFP14 and "0" to CFEX0 (refer to "I/O Ports").
At hot start, the pin retains its pre-reset status.

Output control

To start clock output, write "1" to PF1ON (D0) / Clock option register (0x40190). The clock output is stopped by writing "0".
At initial reset, PF1ON is set to "0" (output disabled).

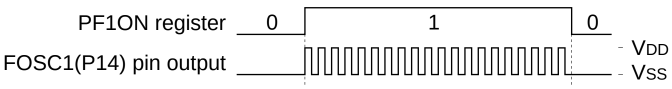


Figure 6.3 OSC1 Clock Output

I/O Memory of Clock Generator

Table 6.3 lists the control bits of clock generator.

Table 6.3 Control Bits of Clock Generator

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Power control register	**0040180 (B)**	D7	**CLKDT1**	System clock division ratio selection	CLKDT[1:0]		Division ratio		0	R/W	
		D6	**CLKDT0**		1	1	1/8		0		
					1	0	1/4				
					0	1	1/2				
					0	0	1/1				
		D5	**PSCON**	Prescaler On/Off control	1	On	0	Off	1	R/W	
		D4–3	**–**	reserved	–				0	–	Writing 1 not allowed.
		D2	**CLKCHG**	CPU operating clock switch	1	OSC3	0	OSC1	1	R/W	
		D1	**SOSC3**	High-speed (OSC3) oscillation On/Off	1	On	0	Off	1	R/W	
		D0	**SOSC1**	Low-speed (OSC1) oscillation On/Off	1	On	0	Off	1	R/W	
Clock option register	**0040190 (B)**	D7–4	**–**	–	–				–	–	0 when being read.
		D3	**HLT2OP**	HALT clock option	1	On	0	Off	0	R/W	
		D2	**8T1ON**	OSC3-stabilize waiting function	1	Off	0	On	1	R/W	
		D1	**–**	reserved	–				0	–	Do not write 1.
		D0	**PF1ON**	OSC1 external output control	1	On	0	Off	0	R/W	
Power control protect register	**004019E (B)**	D7	**CLGP7**	Power control register protect flag	Writing 10010110 (0x96) removes the write protection of the power control register (0x40180). Writing another value set the write protection.				0	R/W	
		D6	**CLGP6**						0		
		D5	**CLGP5**						0		
		D4	**CLGP4**						0		
		D3	**CLGP3**						0		
		D2	**CLGP2**						0		
		D1	**CLGP1**						0		
		D0	**CLGP0**						0		
P1 function select register	**00402D4 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**CFP16**	P16 function selection	1	EXCL5 #DMAEND1	0	P16	0	R/W	
		D5	**CFP15**	P15 function selection	1	EXCL4 #DMAEND0	0	P15	0	R/W	
		D4	**CFP14**	P14 function selection	1	FOSC1	0	P14	0	R/W	Extended functions (0x402DF)
		D3	**CFP13**	P13 function selection	1	EXCL3 T8UF3	0	P13	0	R/W	
		D2	**CFP12**	P12 function selection	1	EXCL2 T8UF2	0	P12	0	R/W	
		D1	**CFP11**	P11 function selection	1	EXCL1 T8UF1	0	P11	0	R/W	
		D0	**CFP10**	P10 function selection	1	EXCL0 T8UF0	0	P10	0	R/W	
Port function extension register	**00402DF (B)**	D7	**CFEX7**	P07 port extended function	1	#DMAEND3	0	P07, etc.	0	R/W	
		D6	**CFEX6**	P06 port extended function	1	#DMAACK3	0	P06, etc.	0	R/W	
		D5	**CFEX5**	P05 port extended function	1	#DMAEND2	0	P05, etc.	0	R/W	
		D4	**CFEX4**	P04 port extended function	1	#DMAACK2	0	P04, etc.	0	R/W	
		D3	**CFEX3**	P31 port extended function	1	#GARD	0	P31, etc.	0	R/W	
		D2	**CFEX2**	P21 port extended function	1	#GAAS	0	P21, etc.	0	R/W	
		D1	**CFEX1**	P10, P11, P13 port extended function	1	DST0 DST1 DPC0	0	P10, etc. P11, etc. P13, etc.	1	R/W	
		D0	**CFEX0**	P12, P14 port extended function	1	DST2 DCLK	0	P12, etc. P14, etc.	1	R/W	

SOSC1: Low-speed (OSC1) oscillation control (D0) / Power control register (0x40180)

Turns the low-speed (OSC1) oscillation on or off.

Write "1": OSC1 oscillation turned on
Write "0": OSC1 oscillation turned off
Read: Valid

The oscillation of the low-speed (OSC1) oscillation circuit is stopped by writing "0" to SOSC1, and started again by writing "1".
Since a duration of maximum three seconds is required for oscillation to stabilize after the oscillation has been restarted, at least this length of time must pass before the OSC1 clock can be used.
Writing to SOSC1 is allowed only when CLGP[7:0] is set to "0b10010110". Note also that if the CPU is operating using the OSC1 clock, writing "0" to SOSC1 is ignored and the oscillation is not turned off.
At initial reset, SOSC1 is set to "1" (OSC1 oscillation turned on).

CLKCHG: CPU operating clock switch (D2) / Power control register (0x40180)

Selects the CPU operating clock.

Write "1": OSC3 clock
Write "0": OSC1 clock
Read: Valid

The OSC3 clock is selected as the CPU operating clock by writing "1" to CLKCHG, and OSC1 is selected by writing "0". The operating clock can be switched over in this way only when both the high-speed (OSC3) and low-speed (OSC1) oscillation circuits are on. In addition, writing to CLKCHG is effective only when CLGP[7:0] is set to "0b10010110". Immediately after the oscillation circuit has started oscillating, wait for the oscillation to stabilize before switching over the CPU operating clock.
At initial reset, CLKCHG is set to "1" (OSC3 clock).
For controlling the high-speed (OSC3) oscillation circuit, refer to "CLG (Clock Generator)" in the Core Block.

HLT2OP: HALT clock option (D3) / Clock option register (0x40190)

Select a HALT condition (basic mode or HALT2 mode).

Write "1": HALT2 mode
Write "0": Basic mode
Read: Valid

When "1" is written to HLT2OP, the CPU will enter HALT2 mode when the HALT instruction is executed. When "0" is written, the CPU will enter basic mode.
Writing to HLT2OP is allowed only when CLGP[7:0] is set to "0b10010110".
At initial reset, HLT2OP is set to "0" (basic mode).

The following shows the operating status in HALT mode (basic mode and HALT2 mode) and SLEEP mode.

Table 6.4 Operating Status in Standby Mode

Standby mode		Operating status	Reactivating factor
HALT mode	Basic mode	(1) The CPU clock is stopped. (CPU stop status) (2) BCU clock is supplied. (BCU run status) (3) Clocks for the peripheral circuits maintain the status before entering HALT mode. (run or stop) (4) The high-speed oscillation circuit maintains the status before entering HALT mode. (5) The low-speed oscillation circuit maintains the status before entering HALT mode.	(1) Reset, NMI (2) Enabled (not masked) interrupt factors
	HALT2 mode	(1) The CPU clock is stopped. (CPU stop status) (2) BCU clock is stopped. (BCU stop status) (3) Clocks for the peripheral circuits maintain the status before entering HALT mode. (run or stop) (4) The high-speed oscillation circuit maintains the status before entering HALT mode. (5) The low-speed oscillation circuit maintains the status before entering HALT mode.	
SLEEP mode		(1) The CPU clock is stopped. (CPU stop status) (2) BCU clock is stopped. (BCU stop status) (3) Clocks for the peripheral circuits are stopped. (4) The high-speed oscillation circuit is stopped. (5) The low-speed oscillation circuit maintains the status before entering SLEEP mode.	(1) Reset, NMI (2) Enabled (not masked) input port interrupt factors (3) Clock timer interrupt when the low-speed oscillation circuit is being operated

PF1ON: OSC1 external output control (D0) / Clock option register (0x40190)

Turns the low-speed (OSC1) clock output to external devices on or off.

Write "1": On
Write "0": Off
Read: Valid

The low-speed (OSC1) clock is output from the FOSC1 pin to an external device by writing "1" to PF1ON. However, for this setting to be effective, the P14 pin must be set for the FOSC1 pin by CFP14 and CFEX0.
The clock output is disabled by writing "0".
Writing to PF1ON is allowed only when CLGP[7:0] is set to "0b10010110".
At initial reset, PF1ON is set to "0" (Off).

CLGP7–CLGP0: Power-control register protection flag ([D[7:0]) / Power control protection register (0x4019E)

These bits remove the protection against writing to addresses 0x40180 and 0x40190.

Write "0b10010110": Write protection removed
Write other than the above: No operation (write-protected)
Read: Valid

Before writing to address 0x40180 or 0x40190, set CLGP[7:0] to "0b10010110" to remove the protection against writing to that address. This clearing of write protection is effective for only one writing, so the bits are cleared to "0b00000000" by one writing. Therefore, CLGP[7:0] must be set each time the protected address is written to.
At initial reset, CLGP is set to "0b00000000" (write-protected).

CFP14: P14 function selection (D4) / P1 function select register (0x402D4)

Selects the pin function of the P14 I/O port.

Write "1": OSC1 clock output pin
Write "0": I/O port pin
Read: Invalid

The P14 pin is set for OSC1 clock output (FOSC1) by writing "1" to CFP14.
At cold start, CFP14 is set to "0" (I/O port pin). At hot start, CFP14 retains its status from before the initial reset.

CFEX0: P12, P14 extended function (D0) / Port function extension register (0x402DF)

Sets whether the function of the P14 pin is to be extended.

Write "1": DCLK output pin
Write "0": P14/FOSC1 output pin
Read: Invalid

When CFEX0 is set to "1", the P14 pin functions as a debug clock DCLK output pin. When CFEX0 = "0", the CFP14 register becomes effective, so the settings of this register determine whether the P14 pin functions as an P14 I/O port or a FOUT1 output pin.
At cold start, CFEX0 is set to "1" (DCLK output pin). At hot start, CFEX0 retains its state from prior to the initial reset.

Programming Notes

(1) Immediately after the low-speed (OSC1) oscillation circuit is turned on, a certain period of time is required for oscillation to stabilize (3 sec max.). To prevent the device from operating erratically, do not use the clock until its oscillation has stabilized.

(2) The oscillation circuit used for the CPU operating clock cannot be turned off.

(3) The CPU operating clock can only be switched over when both the OSC3 and OSC1 oscillation circuits are on. Furthermore, when turning off an oscillation circuit that has become unnecessary as a result of the CPU operating clock switchover, be sure to use separate instructions for switchover and oscillation turnoff. If these two operations are processed simultaneously using one instruction, the CPU may operate erratically.

(4) If the low-speed (OSC1) oscillation circuit is turned off, all peripheral circuits operated using the OSC1 clock will be inactive.

(5) If the OSC3 clock is unnecessary, use the OSC1 clock to operate the CPU and turn the high-speed (OSC3) oscillation circuit off. This helps reduce current consumption.

III-7 CLOCK TIMER

Configuration of Clock Timer

The clock timer consists of an 8-bit binary counter that is clocked by a 256-Hz signal derived from the low-speed (OSC1) oscillation clock f_{OSC1}, and second, minute, hour, and day counters, allowing all data (128 Hz to 1 Hz, seconds, minutes, hours, and day) to be read out in a software. It can also generate an interrupt using a 32-Hz, 8-Hz, 2-Hz, or 1-Hz (1-second) signal or when a one-minute, one-hour, or one-day count is up, in addition to generating an alarm at a specified time (minute or hour) or day.

The low-speed (OSC1) oscillation circuit and the clock timer can be kept operating even when the CPU and other internal peripheral circuits are placed in standby mode (HALT or SLEEP).

Normally, this clock timer should be used for a clock and various other clocking functions.

Figure 7.1 shows the structure of the clock timer.

Note: Since the clock timer is driven by a clock originating from the low-speed (OSC1) oscillation circuit, this timer cannot be used unless the low-speed (OSC1) oscillation circuit (32.768 kHz, Typ.) is used.

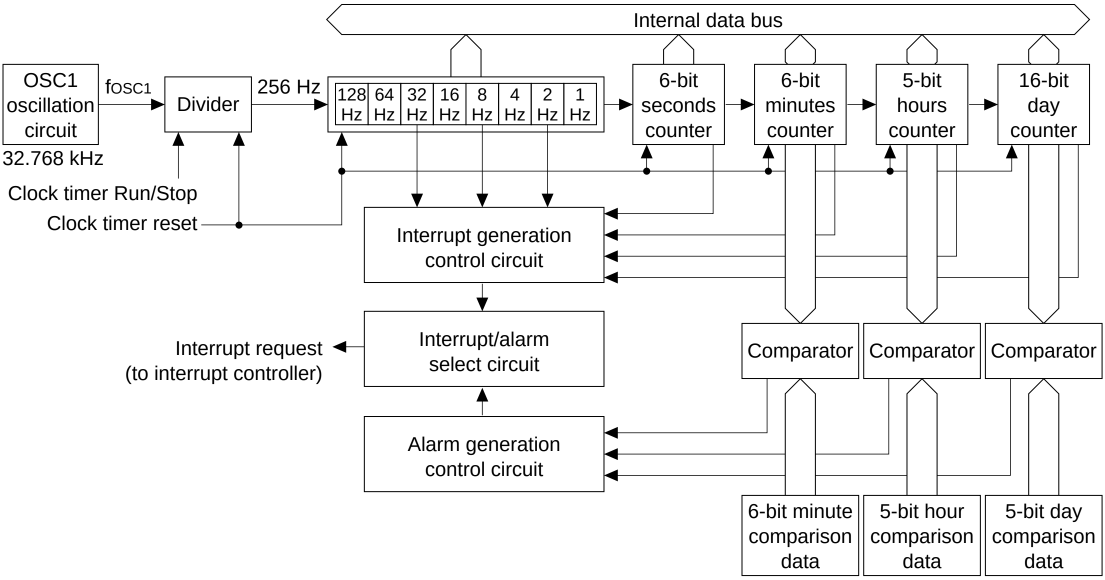


Figure 7.1 Structure of Clock Timer

Control and Operation of the Clock Timer

Initial setting

At initial reset, the clock timer's counter data, setup contents of alarms, and control bits including RUN/STOP, are not initialized. (This does not include the CPU core power on/off flag TCHVOF or OSC1 auto-off flag TCAOFF.)

Therefore, when using the clock timer, initialize it as follows:

1. Before you start setting up, stop the clock timer and disable the clock timer interrupt.
2. Reset the counters.
3. Preset the minute, hour, and day data (only when necessary).
4. Select an interrupt factor.
5. Select the alarm function.
6. Enable the interrupt.
7. Start the clock timer.

The following shows how to set and control each of the above. For details on interrupt control, refer to "Interrupt Function".

Resetting the counters

Each counter of the clock timer can only be reset to "0" in the software. Note that they are not reset by an initial reset or the auto-off function.

To reset the clock timer, write "1" to TCRST (D1) / Clock timer Run/Stop register (0x40151). Note, however, that this reset input is accepted only when the clock timer is inactive, and is ignored when the timer is operating.

Notes:
- The clock timer reset bit TCRST and the clock timer RUN/STOP control bit TCRUN are located at the same address (0x40151). However, the clock timer cannot be reset at the same time it is set to RUN by writing "1" to both. In this case, the reset input is ignored and the timer starts counting up from the counter values then in effect. Always make sure TCRUN = "0" before resetting the timer.
- When the counters are cleared as the clock timer is reset, an interrupt may be generated depending on the timer settings. Therefore, first disable the clock timer interrupt before resetting the clock timer, and after resetting the clock timer, reset the interrupt factor flag, interrupt factor generation flag, and alarm factor generation flag.

Presetting minute, hour, and day data

The clock timer's minute, hour, and day counters have a data preset function, enabling the desired time and day to be set.

Table 7.1 Presetting the Counters

Counter	Data register	Preset value
Minute counter	TCHD[5:0] (D[5:0]) / Clock timer minute register (0x40155)	0 to 59
Hour counter	TCDD[4:0] (D[4:0]) / Clock timer hour register (0x40156)	0 to 23
Day counter	TCND[15:0] (D[7:0]) / Clock timer day (high-order) register (0x40158) (D[7:0]) / Clock timer day (low-order) register (0x40157)	0 to 65535

When using the clock timer as an RTC, be sure to set these counter values before starting operating of the clock timer. For the day counter, set a number of days starting from the reference day (e.g., January 1, 1990).

RUN/STOP the clock timer

The clock timer starts counting when "1" is written to TCRUN (D0) / Clock timer Run/Stop register (0x40151) and stops counting when "0" is written.
When the clock timer is made to RUN, the 256-Hz clock input is enabled at a falling edge of the low-speed (OSC1) oscillation clock pulse, and the 8-bit binary counter counts up at each falling edge of this 256-Hz clock. Figure 7.2 shows the operation of the 8-bit binary counter.

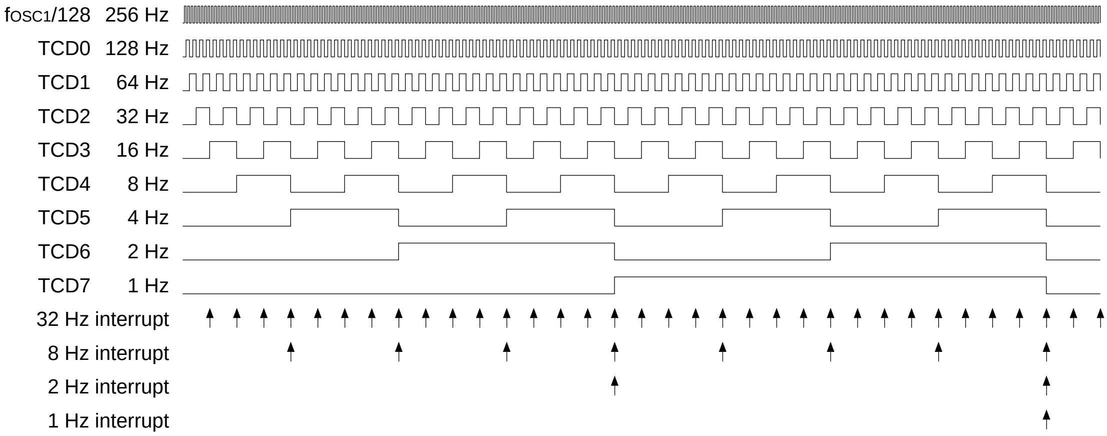


Figure 7.2 Timing Chart of 8-Bit Binary Counter

The 8-bit binary counter outputs a 1-Hz signal in its final stage.
The second counter counts the 1-Hz signal thus output. When it counts 60 seconds, the counter outputs a 60-second signal and is reset to 0 seconds.
Similarly, the minute and hour counters count 60 minutes and 24 hours, respectively, using the signals output by each preceding counter.
The day counter is a 16-bit binary counter and can count up to 65,536 days using the 24-hour signal output by the hour counter.
One of the following signals output by each counter can be selected to generate an interrupt:
32 Hz, 8 Hz, 2 Hz, 1 Hz (1 second), 1 minute, 1 hour, 1 day

If "0" is written to TCRUN, the clock timer is stopped at a rising edge of the low-speed (OSC1) oscillation clock to prevent device malfunction caused by the concurrent termination of counting (falling edge of the 256-Hz clock).

Even when the clock timer is stopped, each counter retains the data set at that point. When the timer is made to RUN again while in that state, each counter restarts counting from the retained value.

Reading out counter data

The data in each counter can be read out in a software as binary data.

Table 7.2 Reading Out Counter Data

Counter	Counter data
1 Hz to 128 Hz	TCD[7:0] (D[7:0]) / Clock timer divider register (0x40153)
Second counter	TCMD[5:0] (D[5:0]) / Clock timer second counter (0x40154)
Minute counter	TCHD[5:0] (D[5:0]) / Clock timer minute counter (0x40155)
Hour counter	TCDD[4:0] (D[4:0]) / Clock timer hour counter (0x40156)
Day counter	TCND[15:0] (D[7:0]) / Clock timer day (high-order) counter (0x40158) (D[7:0]) / Clock timer day (low-order) counter (0x40157)

Data is read directly from the counter during operation. For this reason, a counter can overflow while reading data from each counter, so the data thus read may not be exact. For example, if the 8-bit binary counter is read at 0xFF and then overflows before reading the next seconds counter, the value of the seconds counter is its count plus the one second that has elapsed since the 8-bit binary counter was read. To prevent this problem, try reading out each counter several times and make sure data has not been modified.

Setting alarm function

The clock timer has an alarm function, enabling an interrupt to be generated at a specified time and day. This specification can be made in minutes, hours, and days for each alarm or a combination of multiple alarms.
Use TCASE[2:0] (D[4:2) / Clock timer interrupt control register (0x40152) for this specification.

Table 7.3 Alarm Factor Selection

TCASE2	TCASE1	TCASE0	Alarm factor
X	X	1	Minutes alarm
X	1	X	Hours alarm
1	X	X	Day alarm
0	0	0	None

For example, if TCASE is set to "001", only a minutes alarm is enabled and an alarm is generated at a specified minute every hour. If TCASE is set to "111", an alarm is generated on each specified day at each specified hour and minute. If alarms are not to be used, set TCASE to "000".
An interrupt can be generated every minute, every hour, and every day through the use of the counter's interrupt function instead of the alarm function.
To specify a day, hours, and minutes, use the registers shown below:
To specify minutes: TCCH[5:0] (D[5:0]) / Minute-comparison data register (90x40159) 0 to 59 minutes*
To specify hours: TCCD[4:0] (D[4:0]) / Hour-comparison data register (0x4015A) 0 to 23 hours*
To specify day: TCCN4[4:0] (D[4:0]) / Day-comparison data register 0x4015B) 0 to 31 days after

* The minute-comparison data register (6 bits) and hour-comparison data register (5 bits) can be set for up to 63 minutes and 31 hours, respectively. Note that even when the data set in these registers exceeds 59 minutes or 23 hours, the data is not considered invalid.

The values set in these registers are compared with those of each counter, and when they match, the alarm factor generation flag TCAF (D0) / Clock timer interrupt control register (0x40152) is set to "1". If clock timer interrupts have been enabled using the interrupt controller, an interrupt is generated when the flag is set.
The day-comparison data register is a 5-bit register, and its value is compared with the five low-order bits of the day counter. Therefore, an alarm can be generated for up to 31 days after the register is set.

Interrupt Function

Clock timer interrupt factors

The clock timer can generate an interrupt using a 32-Hz, 8-Hz, 2-Hz, 1-Hz (1-second), 1-minute, 1-hour, or 1-day signal. The interrupt factor to be used from among these signals can be selected using the interrupt factor selection bit TCISE[2:0] (D[7:5]) / Clock timer interrupt control register (0x40152).

Table 7.4 Selecting Interrupt Factor

TCISE2	TCISE1	TCISE0	Interrupt factor
1	1	1	None
1	1	0	1 day
1	0	1	1 hour
1	0	0	1 minute
0	1	1	1 Hz
0	1	0	2 Hz
0	0	1	8 Hz
0	0	0	32 Hz

An interrupt factor is generated at intervals of a selected signal (each falling edge of the signal).
If interrupts based on these signals are not to be used, set TCISE to "111".
When a selected interrupt factor is generated, the interrupt factor generation flag TCIF (D1) / Clock timer interrupt control register (0x40152) is set to "1". At the same time, the clock timer interrupt factor flag FCTM (D1) / Port input 4–7, clock timer, A/D interrupt factor flag register (0x40287) also is set to "1". At this time, if the interrupt conditions set by the interrupt control registers are met, an interrupt to the CPU is generated.

An interrupt can be generated on a specified alarm day at a specified time as described in the preceding section. Interrupts generated by a signal and those generated by an alarm can both be used. However, since the clock timer has only one interrupt factor flag, it is the same interrupt that is generated by the timer. Therefore, if both types of interrupts are used, when an interrupt occurs, read the interrupt factor generation flag TCIF and alarm factor generation flag TCAF to determine which factor has generated the interrupt.
Once the factor generation flag is set to "1", it remains set until it is reset by writing "1" in the software. After confirming that the flag is set, write "1" to reset it.
The interrupt factor generation flag TCIF and alarm factor generation flag TCAF should be reset after at least 4 ms have passed from generation of an interrupt or an alarm.

Note: To prevent generation of an unwanted interrupt, disable the clock timer interrupt before selecting the interrupt and alarm factors. Then, before reenabling the interrupt, reset each factor generation flag and the interrupt factor flag.

Control registers of the interrupt controller

The following lists the clock timer interrupt control registers:
Interrupt factor flag: FCTM (D1) / Port input 4–7, clock timer, A/D interrupt factor flag register (0x40287)
Interrupt enable: ECTM (D1) / Port input 4–7, clock timer, A/D interrupt enable register (0x40277)
Interrupt level: PCTM[2:0] (D[2:0]) / Clock timer interrupt priority register (0x4026B)

When an interrupt factor occurs, the clock timer sets the interrupt actor flag to "1" as described above. At this time, if the interrupt enable register bit is set to "1", an interrupt request is generated.
Interrupts can be disabled by leaving the interrupt enable register bit reset to "0". The interrupt factor flag is always set to "1" when an interrupt factor is generated, regardless of the setting of the interrupt enable register (even when it is set to "0").
The interrupt priority register sets the priority levels (0 to 7) of interrupts. An interrupt request to the CPU is accepted on the condition that no other interrupt request has been generated that is of a higher priority.
It is only when the PSR's IE bit = "1" (interrupts enabled) and the set value of the IL is smaller than the clock timer interrupt level set by the interrupt priority register that a clock timer interrupt request is actually accepted by the CPU.
For details on these interrupt control registers, as well as the device operation when an interrupt has occurred, refer to "ITC (Interrupt Controller)".
Note that the clock timer interrupt factor does not have a function to invoke an intelligent DMA.

Trap vectors

The trap vector addresses for the clock-timer interrupt by default are set as shown below:
When BTA3 = high, 0x0080104
When BTA3 = low, 0x0C00104

The trap table base address can be changed using the TTBR registers (0x48134 to 0x48137).

Examples of Use of Clock Timer

The following shows examples of use of the clock timer and how to control the timer in each case.

To use the clock timer as a timer/counter

Example in which while the CPU is inactive, the clock timer is kept operating in order to start again the CPU after a specified length of time has elapsed (e.g., three days):

1. Make sure the low-speed (OSC1) oscillation circuit is oscillating stably (SOSC1 = "1").
   Wait for approximately three seconds after the oscillation starts for its oscillation to stabilize.
2. Disable the clock timer interrupt using the interrupt controller (ECTM = "0").
3. Stop the clock timer and set "3 days" in the day-comparison register (TCRUN = "0", TCCN = "3").
4. Choose a "day-specified alarm" using the alarm-factor select bit and set "none" in the interrupt-factor select bit (TCASE = "100", TCISE = "111").
5. Reset the interrupt factor and alarm factor generation flags (FCTM = "0", TCAF = "0").
6. Reenable the clock timer interrupt using the interrupt controller (ECTM = "1").
7. Switch the CPU operating clock to the low-speed (OSC1) clock (CLKCHG = "0").
8. Turn off the high-speed (OSC3) oscillation circuit (SOSC3 = "0").
9. Reset the clock timer (TCRST = "0").
10. Start the clock timer (TCRUN = "1").
11. Execute the halt instruction to stop the CPU.

    :

    Wait until an interrupt is generated by a day-specified alarm from the clock timer. When an interrupt occurs, the CPU starts up using the OSC1 clock.

    :

12. If necessary, turn on the high-speed (OSC3) oscillation circuit and change the CPU operating clock back to the OSC3 clock.

In the above example, if the device is reset before a three-day period has elapsed, the device operates as follows:

• The CPU starts up using the OSC3 clock.

• The clock timer counters are not reset. They remain in the RUN state.

The time during which the CPU has been idle can be checked by reading out the clock timer counters.

For using the clock timer as RTC

Example in which the clock timer is kept operating and an alarm is generated at 10:00 A.M. every day:

1. Disable the clock timer interrupt using the interrupt controller (ECTM = "0").
2. Stop the clock timer (TCRUN = "0").
3. Reset the clock timer (TCRST = "1").
4. Set the current day and time in the minute (TCHD), hour (TCDD), and day (TCND) counters. For the day counter, set a number of days starting from the reference day (e.g., January 1, 1990). When the count is read, it is converted into the current date by the software.
5. Set "10:00" in the hour-compare register (TCCD = "0x0A").
6. Select an a "hour-specified alarm" using the alarm factor select bit, and set "none" in the interrupt factor select bit (TCASE = "010", TCISE = "111").
7. Reset the interrupt factor and alarm-factor generation flags (FCTM = "1", TCAF = "0").
8. Reenable the clock timer interrupt using the interrupt controller (ECTM = "1").
9. Start the clock timer (TCRUN = "1").

   :

   The clock timer is made to generate an interrupt at 10:00 every day by an hour-specified alarm.

   :

In the above example, if any interrupt factor other than an alarm is selected, an interrupt is also generated by that interrupt factor. To determine which factor caused the interrupt generated, read the interrupt factor generation flag TCIF and alarm factor generation flag TCAF. If TCAF is set to 1, the interrupt has been caused by an alarm. If you select an interrupt factor (other than a 1-day factor) along with the hour-specified alarm, the selected interrupt factor occurs at the same time as the alarm factor.

I/O Memory of Clock Timer

Table 7.5 shows the control bits of the clock timer.

Table 7.5 Control Bits of Clock Timer

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Clock timer	**0040151**	D7–2	**–**	reserved	–	–	–	0 when being read.
Run/Stop	**(B)**	D1	**TCRST**	Clock timer reset	1 Reset / 0 Invalid	X	W	0 when being read.
register		D0	**TCRUN**	Clock timer Run/Stop control	1 Run / 0 Stop	X	R/W	
Clock timer	**0040152**	D7	**TCISE2**	Clock timer interrupt factor	TCISE[2:0] — Interrupt factor	X	R/W	
interrupt	**(B)**	D6	**TCISE1**	selection	1 1 1 — None	X		
control register		D5	**TCISE0**		1 1 0 — Day	X		
					1 0 1 — Hour			
					1 0 0 — Minute			
					0 1 1 — 1 Hz			
					0 1 0 — 2 Hz			
					0 0 1 — 8 Hz			
					0 0 0 — 32 Hz			
		D4	**TCASE2**	Clock timer alarm factor selection	TCASE[2:0] — Alarm factor	X	R/W	
		D3	**TCASE1**		1 X X — Day	X		
		D2	**TCASE0**		X 1 X — Hour	X		
					X X 1 — Minute			
					0 0 0 — None			
		D1	**TCIF**	Interrupt factor generation flag	1 Generated / 0 Not generated	X	R/W	Reset by writing 1.
		D0	**TCAF**	Alarm factor generation flag	1 Generated / 0 Not generated	X	R/W	Reset by writing 1.
Clock timer	**0040153**	D7	**TCD7**	Clock timer data 1 Hz	1 High / 0 Low	X	R	
divider register	**(B)**	D6	**TCD6**	Clock timer data 2 Hz	1 High / 0 Low	X	R	
		D5	**TCD5**	Clock timer data 4 Hz	1 High / 0 Low	X	R	
		D4	**TCD4**	Clock timer data 8 Hz	1 High / 0 Low	X	R	
		D3	**TCD3**	Clock timer data 16 Hz	1 High / 0 Low	X	R	
		D2	**TCD2**	Clock timer data 32 Hz	1 High / 0 Low	X	R	
		D1	**TCD1**	Clock timer data 64 Hz	1 High / 0 Low	X	R	
		D0	**TCD0**	Clock timer data 128 Hz	1 High / 0 Low	X	R	
Clock timer	**0040154**	D7–6	**–**	reserved	–	–	–	0 when being read.
second	**(B)**	D5	**TCMD5**	Clock timer second counter data	0 to 59 seconds	X	R	
register		D4	**TCMD4**	TCMD5 = MSB		X		
		D3	**TCMD3**	TCMD0 = LSB		X		
		D2	**TCMD2**			X		
		D1	**TCMD1**			X		
		D0	**TCMD0**			X		
Clock timer	**0040155**	D7–6	**–**	reserved	–	–	–	0 when being read.
minute register	**(B)**	D5	**TCHD5**	Clock timer minute counter data	0 to 59 minutes	X	R/W	
		D4	**TCHD4**	TCHD5 = MSB		X		
		D3	**TCHD3**	TCHD0 = LSB		X		
		D2	**TCHD2**			X		
		D1	**TCHD1**			X		
		D0	**TCHD0**			X		
Clock timer	**0040156**	D7–5	**–**	reserved	–	–	–	0 when being read.
hour register	**(B)**	D4	**TCDD4**	Clock timer hour counter data	0 to 23 hours	X	R/W	
		D3	**TCDD3**	TCDD4 = MSB		X		
		D2	**TCDD2**	TCDD0 = LSB		X		
		D1	**TCDD1**			X		
		D0	**TCDD0**			X		
Clock timer	**0040157**	D7	**TCND7**	Clock timer day counter data	0 to 65535 days	X	R/W	
day (low-order)	**(B)**	D6	**TCND6**	(low-order 8 bits)	(low-order 8 bits)	X		
register		D5	**TCND5**	TCND0 = LSB		X		
		D4	**TCND4**			X		
		D3	**TCND3**			X		
		D2	**TCND2**			X		
		D1	**TCND1**			X		
		D0	**TCND0**			X		
Clock timer	**0040158**	D7	**TCND15**	Clock timer day counter data	0 to 65535 days	X	R/W	
day (high-	**(B)**	D6	**TCND14**	(high-order 8 bits)	(high-order 8 bits)	X		
order) register		D5	**TCND13**	TCND15 = MSB		X		
		D4	**TCND12**			X		
		D3	**TCND11**			X		
		D2	**TCND10**			X		
		D1	**TCND9**			X		
		D0	**TCND8**			X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Clock timer minute comparison register	**0040159 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**TCCH5**	Clock timer minute comparison data	0 to 59 minutes (Note) Can be set within 0–63.				X	R/W	
		D4	**TCCH4**	TCCH5 = MSB					X		
		D3	**TCCH3**	TCCH0 = LSB					X		
		D2	**TCCH2**						X		
		D1	**TCCH1**						X		
		D0	**TCCH0**						X		
Clock timer hour comparison register	**004015A (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**TCCD4**	Clock timer hour comparison data	0 to 23 hours (Note) Can be set within 0–31.				X	R/W	
		D3	**TCCD3**	TCCD4 = MSB					X		
		D2	**TCCD2**	TCCD0 = LSB					X		
		D1	**TCCD1**						X		
		D0	**TCCD0**						X		
Clock timer day comparison register	**004015B (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**TCCN4**	Clock timer day comparison data	0 to 31 days				X	R/W	Compared with TCND[4:0].
		D3	**TCCN3**	TCCN4 = MSB					X		
		D2	**TCCN2**	TCCN0 = LSB					X		
		D1	**TCCN1**						X		
		D0	**TCCN0**						X		
Clock timer interrupt priority register	**004026B (B)**	D7–3	–	reserved	–				–	–	Writing 1 not allowed.
		D2	**PCTM2**	Clock timer interrupt level	0 to 7				X	R/W	
		D1	**PCTM1**						X		
		D0	**PCTM0**						X		
Port input 4–7, clock timer, A/D interrupt enable register	**0040277 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**EP7**	Port input 7	1	Enabled	0	Disabled	0	R/W	
		D4	**EP6**	Port input 6					0	R/W	
		D3	**EP5**	Port input 5					0	R/W	
		D2	**EP4**	Port input 4					0	R/W	
		D1	**ECTM**	Clock timer					0	R/W	
		D0	**EADE**	A/D converter					0	R/W	
Port input 4–7, clock timer, A/D interrupt factor flag register	**0040287 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**FP7**	Port input 7	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FP6**	Port input 6					X	R/W	
		D3	**FP5**	Port input 5					X	R/W	
		D2	**FP4**	Port input 4					X	R/W	
		D1	**FCTM**	Clock timer					X	R/W	
		D0	**FADE**	A/D converter					X	R/W	

TCRST: Clock timer reset (D1) / Clock timer Run/Stop register (0x40151)

Resets the clock timer.

Write "1": The clock timer is reset
Write "0": Invalid
Read: Always "0"

The clock timer is reset by writing "1" to TCRST when the timer is inactive. All timer counters are cleared to "0". The clock timer cannot be reset when in the RUN state, nor can it be reset at the same time it is made to RUN through the execution of one write to address 0x40151. (The clock timer is started, but not reset.) In this case, first reset the clock timer and then use another instruction to RUN the clock timer. When the counters are cleared as the clock timer is reset, an interrupt may be generated, depending on the register settings. Therefore, before resetting the clock timer, first disable the clock timer interrupt, and after resetting the clock timer, reset the interrupt factor flag and the interrupt factor and alarm factor generation flags.
Writing "0" to TCRST results in No Operation. Since this TCRST is a write-only bit, its value when read is always "0".
The clock timer is not reset by an initial reset.

TCRUN: Clock timer RUN/STOP control (D0) / Clock timer Run/Stop register (0x40151)

Controls the RUN/STOP of the clock timer.

Write "1": RUN
Write "0": STOP
Read: Valid

The clock timer is made to start counting by writing "1" to the TCRUN register and made to stop by writing "0". The timer data is retained even in the STOP state. The timer can also be made to start counting from the retained data by changing its state from STOP to RUN.
The TCRUN register is not initialized at initial reset.

TCD7–TCD0: 1–128 Hz counter data (D[7:0]) / Clock timer divider register (0x40153)
TCMD5–TCMD0: Second counter data (D[5:0]) / Clock timer second register (0x40154)
TCHD5–TCHD0: Minute counter data (D[5:0]) / Clock timer minute register (0x40155)
TCDD4–TCDD0: Hour counter data (D[4:0]) / Clock timer hour register (0x40156)
TCND15–TCND0: Day counter data (D[7:0]) / Clock timer day (high-order) register (0x40158)
(D[7:0]) / Clock timer day (low-order) register (0x40157)

Data can be read out from each counter.
The minute, hour, and day counters allow data to be written to, in addition to being read out.
The 1–128 Hz counter and seconds counter are read-only, so writing to these registers is ignored.
The unused high-order bits at each address of the second, minute, and hour counter data are always "0" when read out.
The counter data is not initialized at initial reset.

TCCH5–TCCH0: Minute-comparison data (D[5:0]) / Clock timer minute-comparison register (0x40159)
TCCD4–TCCD0: Hour-comparison data (D[4:0]) / Clock timer hour-comparison register (0x4015A)
TCCN4–TCCN0: Day-comparison data (D[4:0]) / Clock timer day-comparison register (0x4015B)

Set a day on which and a time at which an alarm is to be generated.
The comparison data register corresponding to the alarm factor selected using the TCASE register is compared with the counter data, and when the data matches, an alarm interrupt request is generated.
The day-comparison data is compared with the 5 low-order bits of the day counter.
Each register can be read out.
These registers are not initialized at initial reset.

TCISE2–TCISE0: Interrupt factor selection (D[7:5]) / Clock timer interrupt control register (0x40152)

Selects the factor for which the clock timer interrupt is to be generated.

Table 7.6 Selecting Interrupt Factor

TCISE2	TCISE1	TCISE0	Interrupt factor
1	1	1	None
1	1	0	1 day
1	0	1	1 hour
1	0	0	1 minute
0	1	1	1 Hz
0	1	0	2 Hz
0	0	1	8 Hz
0	0	0	32 Hz

When the clock timer interrupt is enabled, an interrupt is generated cyclically at each falling edge of the selected signal. If you the interrupt caused by these factors is not be used set TCISE to "111".
TCISE is not initialized at initial reset.

TCASE2–TCASE0: Alarm factor select register (D[4:2]) / Clock timer interrupt control register (0x40152)

Selects the factor for which an alarm is to be generated.

Table 7.7 Selecting Alarm Factor

TCASE2	TCASE1	TCASE0	Alarm factor
X	X	1	Minute alarm
X	1	X	Hour alarm
1	X	X	Day alarm
0	0	0	None

Use the TCASE2, TCASE1, and TCASE0 bits to select a day, hour, and minute alarm, respectively. It is therefore possible to select multiple alarm factors. When one of these bits is set to "1", the contents of the comparison data register that corresponds to the selected alarm factor is compared with the counter. If the comparison data of all selected alarm factors matches the counter data, an alarm interrupt request is generated. The comparison data register from which the alarm factor is unselected by writing "0" is not compared with the counter data.
TCASE is not initialized at initial reset.

TCIF: Interrupt factor generation flag (D1) / Clock timer interrupt control register (0x40152)

Indicates whether an interrupt factor has occurred.

Read "1": Interrupt factor has occurred
Read "0": No interrupt factor has occurred
Write "1": Flag is reset
Write "0": Invalid

TCIF is set to "1" when an interrupt factor selected using TCISE occurs. Since there is only one source for the clock timer interrupt, use this flag to differentiate it from interrupts caused by an alarm.
Once set to "1", TCIF remains set until it is reset by writing "1".
TCIF is not initialized at initial reset.
This bit does not affect generation of an interrupt even if it is set to "1" or "0".

TCAF: Alarm factor generation flag (D0) / Clock timer interrupt control register (0x40152)

Indicates whether an alarm factor has occurred.

Read "1": Alarm factor has occurred
Read "0": No alarm factor has occurred
Write "1": Flag is reset
Write "0": Invalid

TCAF is set to "1" when all alarm factors selected using the TCASE register occur. Since there is only one source for the clock timer interrupt, use this flag to differentiate it from interrupts due to other interrupt factors.
Once set to "1", TCAF remains set until it is reset by writing "1".
TCAF is not initialized at initial reset.
This bit does not affect generation of an alarm even if it is set to "1" or "0".

PCTM2–PCTM0: Clock timer interrupt level (D[2:0]) / Clock timer interrupt priority register (0x4026B)

Sets the priority level of the clock timer interrupt between 0 and 7.
At initial reset, PCTM becomes indeterminate.

ECTM: Clock timer interrupt enable (D1) / Port input 4–7, clock timer, A/D interrupt enable register (0x40277)

Enables or disables generation of an interrupt to the CPU.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

This bit controls the clock timer interrupt. The interrupt is enabled by setting ECTM to "1" and is disabled by setting it to "0".
At initial reset, ECTM is set to "0" (interrupt disabled).

FCTM: Clock timer interrupt factor flag (D1) / Port input 4–7, clock timer, A/D interrupt factor flag register (0x40287)

Indicates whether the clock timer interrupt factor has occurred.

When read

Read "1": Interrupt factor has occurred
Read "0": No interrupt factor has occurred

When written using the reset-only method (default)

Write "1": Interrupt factor flag is reset
Write "0": Invalid

When written using the read/write method

Write "1": Interrupt flag is set
Write "0": Interrupt flag is reset

FCTM is set to "1" when the selected interrupt factor or alarm factor occurs.
At this time, if the following conditions are met, an interrupt to the CPU is generated:
1. The corresponding interrupt enable register bit is set to "1".
2. No other interrupt request of a higher interrupt priority is generated.
3. The IE bit of the PSR is set to "1" (interrupt enabled).
4. The corresponding interrupt priority register is set to a value higher than the CPU interrupt level (IL).

The interrupt factor flag is always set to "1" when an interrupt factor occurs, no matter how the interrupt enable and interrupt priority registers are set.
For the next interrupt to be accepted after an interrupt has occurred, it is necessary that the interrupt factor flag be reset, and that the PSR be set again (by setting the IE bit to "1" after setting the IL to a value lower than the level indicated by the interrupt priority register, or by executing the reti instruction).
The interrupt factor flag can be reset only by writing to it in the software. Note that if the PSR is set again to accept generated interrupts (or if the reti instruction is executed) without the interrupt factor flag being reset, the same interrupt occurs again. Note also that the value to be written to reset the flag is "1" when the reset-only method (RSTONLY = "1") is used, and "0" when the read/write method (RSTONLY = "0") is used.
The FCTM flag becomes indeterminate at initial reset, so be sure to reset it in the software.

Programming Notes

(1) The low-speed (OSC1) oscillation circuit, which is the clock source for the clock timer, requires a muxmum of three seconds for its oscillation to stabilize after it is started up. Therefore, immediately after power-on, wait until the oscillation stabilizes before starting the clock timer.

(2) At initial reset, the clock timer counter data, the setup contents of alarms, and control bits, including RUN/STOP, are not initialized. Therefore, always initialize the clock timer in the software following power-on.

(3) The clock timer reset bit TCRST and the clock timer RUN/STOP control bit TCRUN are located at the same address (0x40151). However, the clock timer cannot be reset at the same time it is set to RUN by writing "1" to both. In this case, the reset input is ignored and the timer starts counting up from the counter values then in effect. When resetting the timer, always make sure TCRUN = "0" (timer stopped).

(4) When the counters are cleared as the clock timer is reset, an interrupt may be generated depending on the register settings. Therefore, before resetting the clock timer, first disable the clock timer interrupt and, after resetting the clock timer, reset the interrupt factor flag and the interrupt factor generation and alarm factor generation flags.

(5) To prevent generation of an unwanted interrupt, disable the clock timer interrupt before selecting the interrupt and alarm factors. Then, before reenabling the interrupt, reset each factor generation flag and the interrupt factor flag.

(6) The interrupt factor flag (FCTM) becomes indeterminate at initial reset. To prevent generation of an unwanted interrupt, be sure to reset the flag in a program.

(7) To prevent regeneration of interrupts with the same factor after an interrupt has occurred, be sure to reset the interrupt factor flag (FCTM) before setting the PSR again or executing the reti instruction.

III-8 SERIAL INTERFACE

Configuration of Serial Interfaces

Features of Serial Interfaces

The Peripheral Block contains two channels (Ch.0, Ch.1) of serial interfaces, the features of which are described below. The functions of these two serial interfaces are the same.

- A clock-synchronized or asynchronous mode can be selected for the transfer method.
  Clock-synchronized mode
  - Data length: 8 bits, fixed (No start, stop, and parity bits)
  - Receive error: An overrun error can been detected.

  Asynchronous mode
  - Data length: 7 or 8 bits, selectable
  - Receive error: Overrun, framing, or parity errors can been detected.
  - Start bit: 1 bit, fixed
  - Stop bit: 1 or 2 bits, selectable
  - Parity bit: Even, odd, or none; selectable

  Since the transmit and receive units are independent, full-duplex communication is possible.
- Baud-rate setting: Any desired baud rate can be set by selecting the prescaler's division ratio, setting the 8-bit programmable timer, or using external clock input (asynchronous mode only).
- The receive and transmit units are constructed with a double-buffer structure, allowing for successive receive and transmit operations.
- Data transfers using IDMA or HSDMA are possible.
- Three types of interrupts (transmit data empty, receive data full, and receive error) can be generated.

Figure 8.1 shows the configuration of the serial interface (one channel).

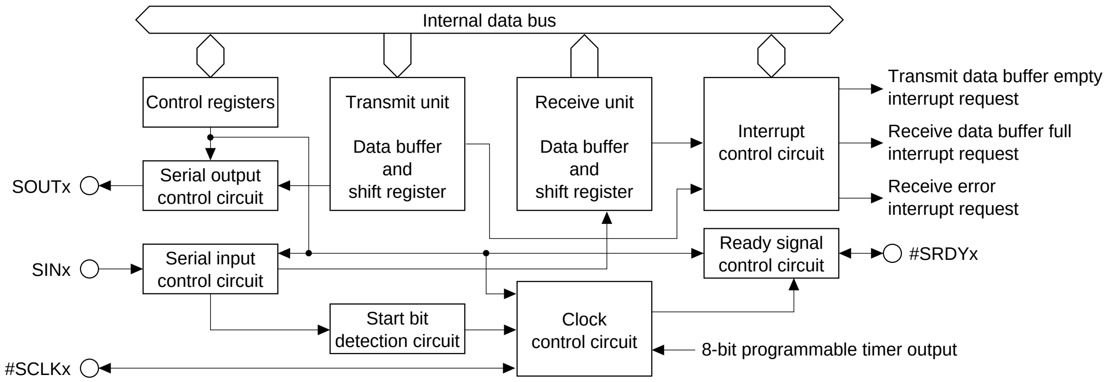


Figure 8.1 Configuration of Serial Interface

Note: Ch.0 and Ch.1 have the same configuration and the same function. The signal and control bit names are suffixed by a 0 or a 1 to indicate the channel number, enabling discrimination between channels 0 and 1. In this manual, however, channel numbers 0 and 1 are replaced with "x" unless discrimination is necessary, because explanations are common to both channels.

I/O Pins of Serial Interface

Table 8.1 lists the I/O pins used by the serial interface.

Table 8.1 Serial-Interface Pin Configuration

Pin name	I/O	Function	Function select bit
P00/SIN0	I/O	I/O port / Serial IF Ch.0 data input	CFP00(D0)/P0 function select register(0x402D0)
P01/SOUT0	I/O	I/O port / Serial IF Ch.0 data output	CFP01(D1)/P0 function select register(0x402D0)
P02/#SCLK0	I/O	I/O port / Serial IF Ch.0 clock input/output	CFP02(D2)/P0 function select register(0x402D0)
P03/#SRDY0	I/O	I/O port / Serial IF Ch.0 ready input/output	CFP03(D3)/P0 function select register(0x402D0)
P04/SIN1/ #DMAACK2	I/O	I/O port / Serial IF Ch.1 data input	CFP04(D4)/P0 function select register(0x402D0) CFEX4(D4)/Port function extension register(0x402DF)
P05/SOUT1/ #DMAEND2	I/O	I/O port / Serial IF Ch.1 data output	CFP05(D5)/P0 function select register(0x402D0) CFEX5(D5)/Port function extension register(0x402DF)
P06/#SCLK1/ #DMAACK3	I/O	I/O port / Serial IF Ch.1 clock input/output	CFP06(D6)/P0 function select register(0x402D0) CFEX6(D6)/Port function extension register(0x402DF)
P07/#SRDY1/ #DMAEND3	I/O	I/O port / Serial IF Ch.1 ready input/output	CFP07(D7)/P0 function select register(0x402D0) CFEX7(D7)/Port function extension register(0x402DF)

SINx (serial-data input pin)

This pin is used to input serial data to the device, regardless of the transfer mode.

SOUTx (serial-data output pin)

This pin is used to output serial data from the device, regardless of the transfer mode.

#SCLKx (clock input/output pin)

This pin is used to input or output a clock.

In the clock-synchronized slave mode, it is used as a clock input pin; in the clock-synchronized master mode, it is used as a clock output pin.

In the asynchronous mode, this pin is used as clock input when an external clock is used. This pin is not used when the internal clock is used, so it can be used as an I/O port.

#SRDYx (ready-signal input/output pin)

This pin is used to input or output the ready signal that is used in the clock-synchronized mode.

In the clock-synchronized slave mode, it is used as a ready-signal output pin; in the clock-synchronized master mode, it is used as a ready-signal input pin.

This pin is not used in the asynchronous mode, so it can be used as an I/O port.

Method for setting the serial-interface input/output pins

All of the pins used in the serial interface are shared with I/O ports. At cold start, they are all set for I/O port pins P0x (function select bit CFP0x = "0"). When using the serial interface, write "1" to CFP0x for the pin to be used in accordance with the channel and transfer mode used.

Furthermore, the P04–P07 ports that are used for channel 1 may be used for channels 2 and 3 of HSDMA. When using the serial interface Ch.1, fix CFEX[7:4] at "0" (default).

At hot start, the pins retain their status from prior to the reset.

Setting Transfer Mode

The transfer mode of the serial interface can be set using SMDx[1:0] individually for each channel as shown in Table 8.2 below.

Table 8.2 Transfer Mode

SMDx1	SMDx0	Transfer mode
1	1	8-bit asynchronous mode
1	0	7-bit asynchronous mode
0	1	Clock-synchronized slave mode
0	0	Clock-synchronized master mode

At initial reset, SMDx becomes indeterminate, so be sure to initialize it in the software.
When using the IrDA interface, set the transfer mode for the asynchronous 7-bit or asynchronous 8-bit mode.
The input/output pins are configured differently, depending on the transfer mode. The pin configuration in each mode is shown in Table 8.3.

Table 8.3 Pin Configuration by Transfer Mode

Transfer mode	SINx (P00/P04)	SOUTx (P01/P05)	#SCLKx (P02/P06)	#SRDYx (P03/P07)
8-bit asynchronous	Data input	Data output	Clock input/P port	P port
7-bit asynchronous	Data input	Data output	Clock input/P port	P port
Clock-synchronized slave	Data input	Data output	Clock input	Ready output
Clock-synchronized master	Data input	Data output	Clock output	Ready input

All four pins are used in the clock-synchronized mode.
In the asynchronous mode, since #SRDYx is unused, P03 (or P07) can be used as an I/O (P) port. In addition, when an external clock is not used, P02 (or P06) can also be used as an I/O port.
The I/O control and data registers for the I/O ports used in the serial interface can be used as general-purpose read/write registers.

Note: To enable the IrDA interface to be set, IRMDx[1:0] (D[1:0]) / Serial I/F Ch.0 IrDA register (0x401E4) or Serial I/F Ch.1 IrDA register (0x401E9) is provided. Since these bits become indeterminate at initial reset, be sure to initialize them by writing "00" when using as the normal interface or "10" when using as the IrDA interface.

Clock-Synchronized Interface

Outline of Clock-Synchronized Interface

In the clock-synchronized transfer mode, 8 bits of data are synchronized to the common clock on both the transmit and receive sides when the data is transferred. Since the transmit and receive units both have a double-buffer structure, successive transmit and receive operations are possible. Since the clock line is shared between the transmit and receive units, the communication mode is half-duplex.

Master and slave modes

Either the clock-synchronized master mode or the clock-synchronized slave mode can be selected using SMDx[1:0].

Clock-synchronized master mode (SMDx[1:0] = "00")

In this mode, clock-synchronized 8-bit serial transfers, in which the serial interface functions as the master, can be performed using the internal clock to synchronize the operation of the internal shift registers.
The synchronizing clock is output from the #SCLKx pin, enabling an external (slave side) serial input/output device to be controlled. The #SRDYx pin is also used to input a signal that indicates whether the external serial input/output device is ready to transmit or receive (when ready in a low level).

Clock-synchronized slave mode (SMDx[1:0] = "01")

In this mode, clock-synchronized 8-bit serial transfers, in which the serial interface functions as a slave, can be performed using the synchronizing clock that is supplied by an external (master side) serial input/output device. The synchronizing clock is input from the #SCLKx pin for use as the synchronizing clock of the serial interface. In addition, a #SRDYx signal indicating whether the serial interface is ready to transmit or receive (when ready in a low level) is output from the #SRDYx pin.

Figure 8.2 shows an example of how the input/output pins are connected in the clock-synchronized mode.

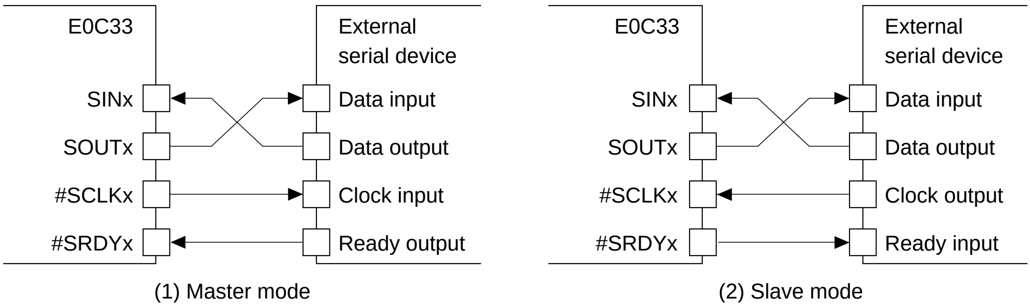


Figure 8.2 Example of Connection in Clock-Synchronized Mode

Clock-synchronized transfer data format

In clock-synchronized transfers, the data format is fixed as shown below.

Data length: 8 bits
Start bit: None
Stop bit: None
Parity bit: None

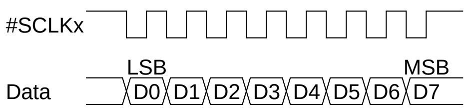


Figure 8.3 Clock-Synchronized Transfer Data Format

Serial data is transmitted and received starting with the LSB.

Setting Clock-Synchronized Interface

When performing clock-synchronized transfers via the serial interface, the following settings must be made before data transfer is actually begun:

1. Setting input/output pins
2. Setting the interface mode
3. Setting the transfer mode
4. Setting the input clock
5. Setting interrupts and IDMA/HSDMA

The following explains the content of each setting. For details on interrupt/DMA settings, refer to "Serial Interface Interrupts and DMA".

Note: Always make sure the serial interface is inactive (TXENx and RXENx = "0") before these settings are made. A change of settings during operation may cause a malfunction.

Setting input/output pins

All four pins—SINx, SOUTx, #SCLKx, and #SRDYx—are used in the clock-synchronized mode. When using Ch.0, set CFP0[3:0] (D[3:0]) / P0 function select register (0x402D0) to "1111" and when using Ch.1, set CFP0[7:4] (D[7:4]) / P0 function select register (0x402D0) to "1111". (It is possible to use both channels.)

Setting the interface mode

IRMDx[1:0] (D[1:0]) / Serial I/F Ch.0 IrDA register (0x401E4) or Serial I/F Ch.1 IrDA register (0x401E9) is used to set the interface mode (normal or IrDA interface). Write "00" to IRMDx[1:0] to choose the ordinary interface. Since IRMDx[1:0] becomes indeterminate at initial reset, it must be initialized.

Setting the transfer mode

Use SMDx to set the transfer mode of the serial interface as described earlier. When using the serial interface as the master for clock-synchronized transfer, set SMDx[1:0] to "00"; when using the serial interface as a slave, set SMDx[1:0] to "01".

Setting the input clock

- **Clock-synchronized master mode**

This mode operates using an internally derived clock. The clock source for each channel is as follows:
Ch.0: A clock output by 8-bit programmable timer 2
Ch.1: A clock output by 8-bit programmable timer 3

Therefore, in order for the serial interface to be used in the clock-synchronized master mode, the following conditions must be met:

1. The prescaler is feeding a clock to 8-bit programmable timer 2 (3).
2. The 8-bit programmable timer 2 (3) is generating a clock.

Any desired clock frequency can be selected by setting the division ratio of the prescaler and the reload data of the 8-bit programmable timer as necessary. The relationship between the contents of these settings and the transfer rate is expressed by Eq. 1 below.
To ensure that the duty ratio of the clock to be fed to the serial interface is 50%, the 8-bit programmable timer further divides the underflow signal frequency by 2 internally. This 1/2 frequency division is factored into Eq. 1.

$$RLD = \frac{f_{PSCIN} \times dr}{2 \times bps} - 1 \qquad \text{(Eq. 1)}$$

RLD: Reload data register setup value of the 8-bit programmable timer
f_{PSCIN}: Prescaler input clock frequency (Hz)
bps: Transfer rate (bits/second)
dr: Division ratio of the prescaler

Note: The division ratios selected by the prescaler differ between 8-bit programmable timers 2 and 3, so be careful when setting the ratio.
8-bit programmable timer 2: 1/2, 1/4, 1/8, 1/32, 1/64, 1/2048, 1/4096
8-bit programmable timer 3: 1/2, 1/4, 1/8, 1/32, 1/64, 1/128, 1/256

For details on how to control the prescaler and 8-bit programmable timers, refer to "Prescaler", and "8-Bit Programmable Timers".

The serial-interface control register contains an SSCKx bit to select the clock source used for the asynchronous mode. Although this bit does not affect the clock in the clock-synchronized mode, its content becomes indeterminate at initial reset. Therefore, be sure to initialize this bit by writing "0" (Internal clock), even when using the serial interface in the clock-synchronized master mode.

- **Clock-synchronized slave mode**

This mode operates using the clock that is output by the external master. This clock is input from the #SCLK pin.
Therefore, there is no need to control the prescaler or 8-bit programmable timer.

Initialize SSCKx by writing "1" (#SCLKx).

Control and Operation of Clock-Synchronized Transfer

Transmit control

(1) Enabling transmit operation

Use the transmit-enable bit TXENx for transmit control.
Ch.0 transmit-enable: TXEN0 (D7) / Serial I/F Ch.0 control register (0x401E3)
Ch.1 transmit-enable: TXEN1 (D7) / Serial I/F Ch.1 control register (0x401E8)
When transmit is enabled by writing "1" to this bit, the clock input to the shift register is enabled (ready for input), thus allowing for data to be transmitted. The synchronizing clock input/output of the #SCLKx pin is also enabled (ready for input/output).
Transmit is disabled by writing "0" to TXENx.

After the P0 function select register is set for the serial interface, the I/O direction of the #SRDY and #SCLK pins are changed at follows:
#SRDY: When transmission is enabled in slave mode, P03 (P07) enters output mode.
Otherwise, P03 (P07) stays in input mode.
#SCLK: When transmission is enabled in master mode, P02 (P06) enters output mode.
Otherwise, P02 (P06) stays in input mode.

Note: In clock-synchronized transfers, the clock line is shared between the transmit and receive units, so the communication mode is half-duplex. Therefore, TXENx and receive-enable bit RXENx cannot be enabled simultaneously. When transmitting data, fix RXENx at "0" and do not change it during a transmit operation.
In addition, make sure TXENx is not set to "0" during a transmit operation.

(2) Transmit procedure

The serial interface contains a transmit shift register and a transmit data register (transmit data buffer), which are provided independently of those used for a receive operation.
Ch.0 transmit data: TXD0[7:0] (D[7:0]) / Serial I/F Ch.0 transmit data register (0x401E0)
Ch.1 transmit data: TXD1[7:0] (D[7:0]) / Serial I/F Ch.1 transmit data register (0x401E5)
The serial interface contains a status bit to indicate the status of the transmit data register.
Ch.0 transmit data buffer empty: TDBE0(D1) / Serial I/F Ch.0 status register (0x401E2)
Ch.1 transmit data buffer empty: TDBE1(D1) / Serial I/F Ch.1 status register (0x401E7)
This bit is reset to "0" by writing data to the transmit-data register, and set to "1" again (buffer empty) when the data is transferred to the shift register.
The serial interface starts transmitting when data is written to the transmit data register.

The transfer status can be checked using the transmit-completion flag (TENDx).
Ch.0 transmit-completion flag: TEND0 (D5) / Serial I/F Ch.0 status register (0x401E2)
Ch.1 transmit-completion flag: TEND1 (D5) / Serial I/F Ch.1 status register (0x401E7)
This bit goes "1" when data is being transmitted and goes "0" when the transmission has completed.
When data is transmitted successively in clock-synchronized master mode, TENDx maintains "1" until all data is transmitted (Figure 8.4). In slave mode, TENDx goes "0" every time 1-byte data is transmitted (Figure 8.5).

Following explains transmit operation in both the master and slave modes.

- Clock-synchronized master mode

The timing at which the device starts transmitting in the master mode is as follows:
When #SRDY is on a low level while TDBEx = "0" (the transmit-data register contains data written to it) or when TDBEx is set to "0" (data has been written to the transmit-data register) while #SRDY is on a low level.
Figure 8.4 shows a transmit timing chart in the clock-synchronized master mode.

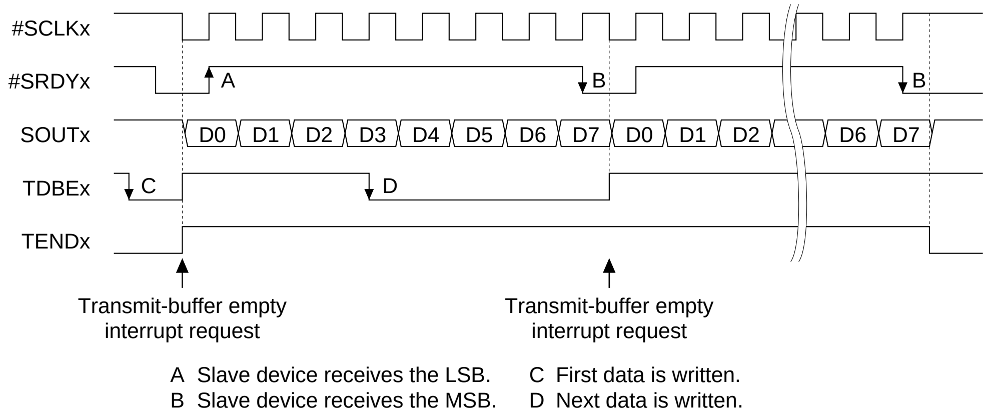


Figure 8.4 Transmit Timing Chart in Clock-Synchronized Master Mode

1. If the #SRDYx signal from the slave is on a high level, the master waits until it is on a low level (ready to receive).
2. If #SRDYx is on a low level, the synchronizing clock input to the serial interface begins. The synchronizing clock is also output from the #SCLKx pin to the slave device.
3. The content of the data register is transferred to the shift register synchronously with the first falling edge of the clock. At the same time, the LSB of the data transferred to the shift register is output from the SOUTx pin.
4. The data in the shift register is shifted 1 bit by the next falling edge of the clock, and the bit following the LSB is output from SOUTx. This operation is repeated until all 8 bits of data are transmitted.

The slave device must take in each bit synchronously with the rising edges of the synchronizing clock.

- Clock-synchronized slave mode

Figure 8.5 shows a transmit timing chart in the clock-synchronized slave mode.

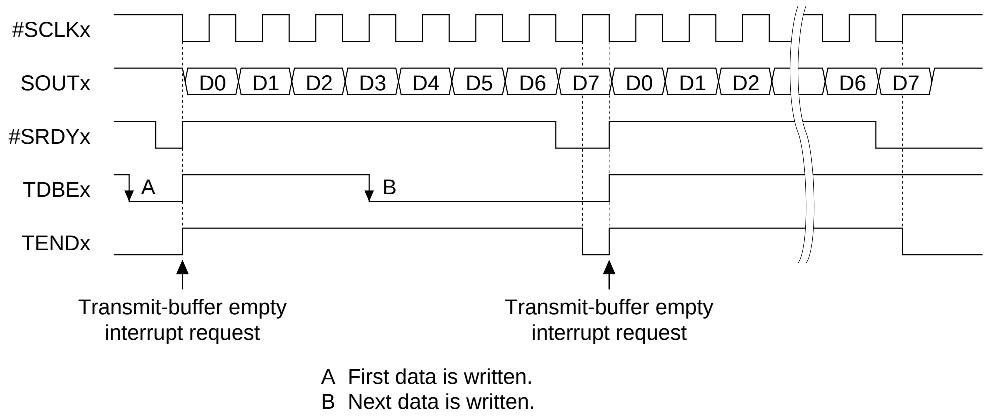


Figure 8.5 Transmit Timing Chart in Clock-Synchronized Slave Mode

1. After setting the #SRDYx signal to a low level (ready to transmit), the slave waits for clock input from the master.
2. When the synchronizing clock is input from the #SCLKx pin, the content of the data register is transferred to the shift register synchronously with the first falling edge of the clock. At the same time, the LSB of the data transferred to the shift register is output from the SOUTx pin.
   The #SRDYx signal is returned to a high level at this point.
3. The data in the shift register is shifted 1 bit by the next falling edge of the clock, and the bit following the LSB is output from SOUTx. This operation is repeated until all 8 bits of data are transmitted.
4. The #SRDYx signal is set to a low level when the last bit (8th bit) is output from the SOUTx pin.

The master device must take in each bit synchronously with the rising edges of the synchronizing clock.

- Successive transmit operations

  When the data in the transmit data register is transferred to the shift register, TDBEx is reset to "1" (buffer empty). Once this occurs, the next transmit data can be written to the transmit data register, even during data transmission.

  This allows data to be transmitted successively. The transmit procedure is described above.

  When TDBEx is set to "1", a transmit-data empty interrupt factor occurs. Since an interrupt can be generated as set by the interrupt controller, the next piece of transmit data can be written using an interrupt processing routine. In addition, since this interrupt factor can be used to invoke DMA, the data prepared in memory can be transmitted successively to the transmit-data register through DMA transfers.

  For details on how to control interrupts and DMA requests, refer to "Serial Interface Interrupts and DMA".

(3) Terminating transmit operation

Upon completion of data transmission, write "0" to the transmit-enable bit TXENx to disable transmit operation.

Receive control

(1) Enabling receive operation

Use the receive-enable bit RXENx for receive control.
Ch.0 receive-enable: RXEN0 (D6) / Serial I/F Ch.0 receive-enable register (0x401E3)
Ch.1 receive-enable: RXEN1 (D6) / Serial I/F Ch.1 receive-enable register (0x401E8)
When receive operations are enabled by writing "1" to this bit, clock input to the shift register is enabled (ready for input), thereby starting a data-receive operation. The synchronizing clock input/output on the #SCLKx pin also is enabled (ready for input/output). Receive operations are disabled by writing "0" to RXENx.

After the P0 function select register is set for the serial interface, the I/O direction of the #SRDY and #SCLK pins are changed at follows:
#SRDY: When receive operation is enabled in slave mode, P03 (P07) enters output mode.
Otherwise, P03 (P07) stays in input mode.
#SCLK: When receive operation is enabled in master mode, P02 (P06) enters output mode.
Otherwise, P02 (P06) stays in input mode.

Note: In clock-synchronized transfers, the clock line is shared between the transmit and receive units, so the communication mode is half-duplex. Therefore, RXENx and transmit-enable bit TXENx cannot be enabled simultaneously. When receiving data, fix TXENx at "0" and do not change it during a receive operation. In addition, make sure RXENx is not set to "0" during a receive operation.

(2) Receive procedure

This serial interface has a receive shift register and a receive data register (receive data buffer) that are provided independently of those used for transmit operations.
Ch.0 receive data: RXD0[7:0] (D[7:0]) / Serial I/F Ch.0 receive data register (0x401E1)
Ch.1 receive data: RXD1[7:0] (D[7:0]) / Serial I/F Ch.1 receive data register (0x401E6)
The receive data can be read out from this register.
A status bit is also provided that indicates the status of the receive data register.
Ch.0 receive data buffer full: RDBF0 (D0) / Serial I/F Ch.0 status register (0x401E2)
Ch.1 receive data buffer full: RDBF1 (D0) / Serial I/F Ch.1 status register (0x401E7)
This bit is set to "1" (buffer full) when the MSB of serial data is received and the data in the shift register is transferred to the receive data register, indicating that the received data can be read out. When the data is read out, the bit is reset to "0".

The following describes a receive operation in the master and slave modes.

- Clock-synchronized master mode

Figure 8.6 shows a receive timing chart in the clock-synchronized master mode.

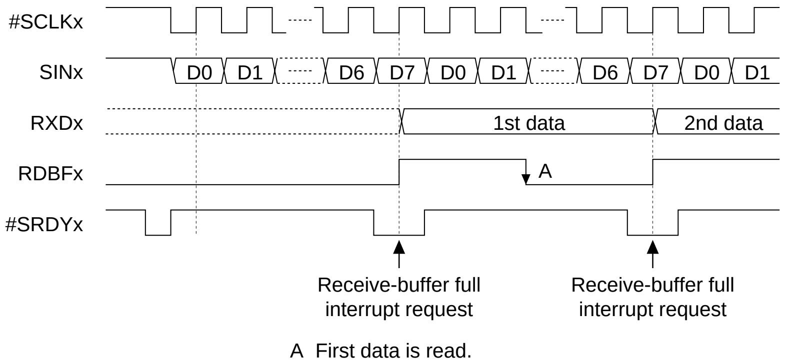


Figure 8.6 Receive Timing Chart in Clock-Synchronized Master Mode

1. If the #SRDYx signal from the slave is on a high level, the master waits until it turns to a low level (ready to receive).
2. If #SRDYx is on a low level, synchronizing clock input to the serial interface begins. The synchronizing clock is also output from the #SCLKx pin to the slave device.
3. The slave device outputs each bit of data synchronously with the falling edges of the clock. The LSB is output first.
4. This serial interface takes the SIN input into the shift register at the rising edges of the clock. The data in the shift register is sequentially shifted as bits are taken in. This operation is repeated until the MSB of data is received.
5. When the MSB is taken in, the data in the shift register is transferred to the receive data register, enabling the data to be read out.

- Clock-synchronized slave mode

Figure 8.7 shows a receive timing chart in the clock-synchronized slave mode.

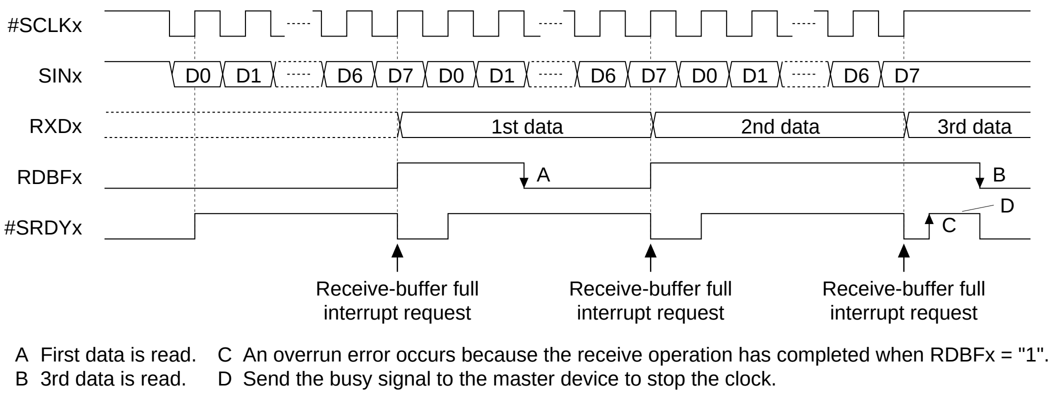


Figure 8.7 Receive Timing Chart in Clock-Synchronized Slave Mode

1. After setting the #SRDYx signal to a low level (ready to receive), the slave waits for clock input from the master.
2. The master device outputs each bit of data synchronously with the falling edges of the clock. The LSB is output first.
3. This serial interface takes the SIN input into the shift register at the rising edges of the clock that is input from #SCLKx. The data in the shift register is sequentially shifted as bits are taken in. This operation is repeated until the MSB of data is received.
4. When the MSB is taken in, the data in the shift register is transferred to the receive data register, enabling the data to be read out.

- Successive receive operations

  When the data received in the shift register is transferred to the receive data register, RDBFx is set to "1" (buffer full), indicating that the received data can be read out.

  Since the receive data register can be read out while receiving the next data, data can be received successively. The procedure for receiving is described above.

  When RDBFx is set to "1", a receive-data full interrupt factor occurs. Since an interrupt can be generated as set by the interrupt controller, the received data can be read by an interrupt processing routine. In addition, since this interrupt factor can be used to invoke DMA, the received data can be received successively in locations prepared in memory through DMA transfers.

  For details on how to control interrupts/DMA, refer to "Serial Interface Interrupts and DMA".

(3) Overrun error

If, during successive receive operation, a receive operation for the next data is completed before the receive data register is read out, the receive data register is overwritten with the new data. Therefore, the receive data register must always be read out before a receive operation for the next data is completed.

When the receive data register is overwritten, an overrun error is generated and the overrun error flag is set to "1".

Ch.0 overrun error flag: OER0 (D2) / Serial I/F Ch.0 status register (0x401E2)

Ch.1 overrun error flag: OER1 (D2) / Serial I/F Ch.1 status register (0x401E7)

Once the overrun error flag is set to "1", it remains set until it is reset by writing "0" to it in the software.

The overrun error is one of the receive-error interrupt factors in the serial interface. An interrupt can be generated for this error by setting the interrupt controller as necessary, so that the error can be processed by an interrupt processing routine.

(4) #SRDYx in slave mode

When receive operations are enabled by writing "1" to RXENx, the #SRDYx signal is turned to a low level, thereby indicating to the master device that the slave is ready to receive. When the LSB of serial data is received, #SRDYx is turned to a high level; when the MSB is received, #SRDYx is returned to a low level, in preparation for the next receive operation.

If an overrun error occurs, #SRDYx is turned to a high level (unable to receive) at that point, with receive operations for the following data thus suspended. In this case, #SRDYx is returned to a low by reading out the data overwritten in the receive data register, and if any receive data follows, the slave restarts receiving data.

(5) Terminating receive operation

Upon completion of a data receive operation, write "0" to the receive-enable bit RXENx to disable receive operations.

Asynchronous Interface

Outline of Asynchronous Interface

Asynchronous transfers are performed by adding a start bit and a stop bit to the start and end points of each serial-converted data. With this method, there is no need to use a clock that is fully synchronized on the transmit and receive sides; instead, transfer operations are timed by the start and stop bits added to the start and end points of each data.

In the 8-bit asynchronous mode (SMDx[1:0] = "11"), 8 bits of data can be transferred; in the 7-bit asynchronous mode (SMDx[1:0] = "10"), 7 bits of data can be transferred.

In either mode, it is possible to select the stop-bit length, add a parity bit, and choose between even and odd parity. The start bit is fixed at "1".

The operating clock can be selected between an internal clock generated by an 8-bit programmable timer or an external clock that is input from the #SCLKx pin.

Since the transmit and receive units are both constructed with a double-buffer structure, successive transmit and receive operations are possible. Furthermore, since the transmit and receive units are independent, full-duplex communication in which transmit and receive operations are performed simultaneously is also possible.

Figure 8.8 shows an example of how input/output pins are connected for transfers in the asynchronous mode.

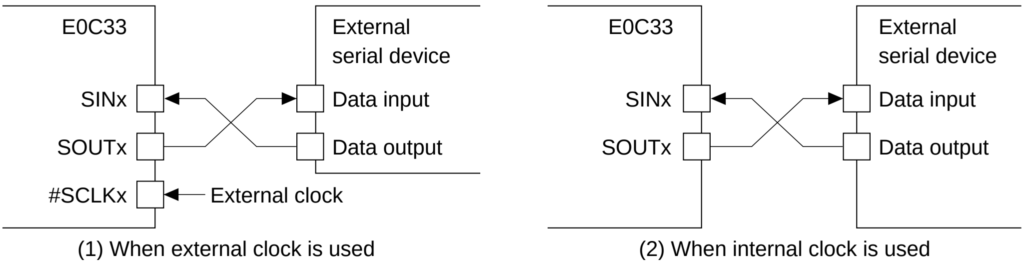


Figure 8.8 Example of Connection in Asynchronous Mode

When the asynchronous mode is selected, it is possible to use the IrDA interface function.

Asynchronous-transfer data format

The data format for asynchronous transfer is shown below.

Data length: 7 or 8 bits (determined by the selected transfer mode)
Start bit: 1 bit, fixed
Stop bit: 1 or 2 bits
Parity bit: Even or odd parity, or none

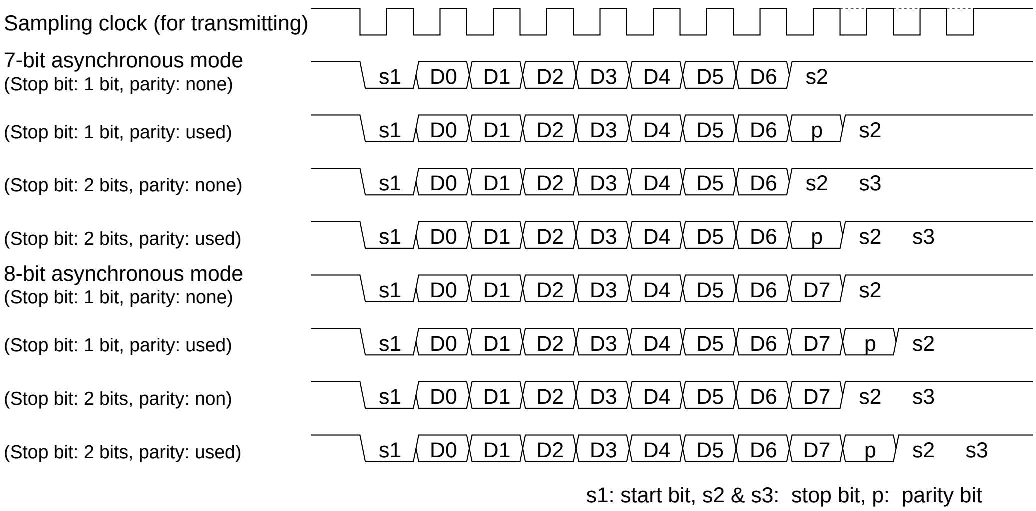


Figure 8.9 Data Format for Asynchronous Transfer

Serial data is transmitted and received, starting with the LSB.

Setting Asynchronous Interface

When performing asynchronous transfer via the serial interface, the following must be done before data transfer can be started:

1. Setting input/output pins
2. Setting the interface mode
3. Setting the transfer mode
4. Setting the input clock
5. Setting the data format
6. Setting interrupt/IDMA/HSDMA

The following describes how to set each of the above. For details on interrupt/DMA settings, refer to "Serial Interface Interrupts and DMA".

Note: Always make sure the serial interface is inactive (TXENx and RXENx = "0") before making these settings. A change in settings during operation may result in a malfunction.

Setting input/output pins

In the asynchronous mode, two pins–SINx and SOUTx–are used. When external clock input is used, one more pin, #SCLKx, is also used.

Set CFP0[7:0] (D[7:0]) / P0 function select register (0x402D0) according to the pins used. (Both channels can be used, if necessary.) Since the #SRDYx pin is not used, P03 or P07 can be used as an I/O port. During operation using the internal clock, P03 or P06 can also be used as an I/O port.

Setting the interface mode

IRMDx[1:0] (D[1:0]) / Serial I/F Ch.0 IrDA register (0x401E4) is used to set the IrDA interface. Since IRMDx[1:0] becomes indeterminate at initial reset, initialize it by writing "00" when using the serial interface as a normal interface, or "10" when using the serial interface as an IrDA interface. This setting must be made before a transfer mode is set.

Setting the transfer mode

Use SMDx to set the transfer mode of the serial interface as described earlier. When using the serial interface in the 8-bit asynchronous mode, set SMDx[1:0] to "11", when using the serial interface in the 7-bit asynchronous mode, set SMDx[1:0] to "10".

Setting the input clock

In the asynchronous mode, the operating clock can be selected between the internal clock and an external clock.

Ch.0 input clock selection: SSCK0 (D2) / Serial I/F Ch.0 control register (0x401E3)

Ch.1 input clock selection: SSCK1 (D2) / Serial I/F Ch.1 control register (0x401E8)

The external clock is selected (input from the #SCLKx pin) by writing "1" to SSCKx, and an internal clock is selected by writing "0".

Note: SSCKx becomes indeterminate at initial reset, so be sure to reset it in the software.

- **Internal clock**

When the internal clock is selected, the serial interface is clocked by a clock generated using an 8-bit programmable timer. The clock source for each channel is as follows:

Ch.0: Clock output by 8-bit programmable timer 2

Ch.1: Clock output by 8-bit programmable timer 3

Therefore, before the internal clock can be used, the following conditions must be met:

1. The prescaler is outputting a clock to the 8-bit programmable timer 2 (or 3).
2. The 8-bit programmable timer 2 (or 3) is outputting a clock.

Any desired clock frequency can be obtained by setting the prescaler division ratio and the reload data of the 8-bit programmable timer as necessary. The relationship between the contents of these setting and the transfer rate is expressed by Eq. 2.
The 8-bit programmable timer has its underflow signal further divided by 2 internally, in order to ensure that the duty ratio of the clock supplied to the serial interface is 50%.
Furthermore, the clock output by the 8-bit programmable timer is divided by 16 or 8 internally in the serial interface, in order to create a sampling clock (refer to "Sampling clock"). This division ratio must also be considered when setting the transfer rate.
These division ratios are taken into account in Eq. 2.

$$RLD = \frac{f_{PSCIN} \times pdr \times sdr}{2 \times bps} - 1 \qquad \text{(Eq. 2)}$$

RLD: Set value of the 8-bit programmable timer's reload data register
f_{PSCIN}: Prescaler input clock frequency (Hz)
bps: Transfer rate (bits/second)
pdr: Division ratio of the prescaler
sdr: Internal division ratio of the serial interface (1/16 or 1/8)

Note: The division ratio selected using the prescaler differs between 8-bit programmable timers 2 and 3. Take this into account when setting a division ratio.
8-bit programmable timer 2: 1/2, 1/4, 1/8, 1/32, 1/64, 1/2048, 1/4096
8-bit programmable timer 3: 1/2, 1/4, 1/8, 1/32, 1/64, 1/128, 1/256

Table 8.4 shows examples of prescaler division ratios and the reload data settings of the programmable timer, in cases in which the internal division ratio of the serial interface is set to 1/16.

Table 8.4 Example of Transfer Rate Settings

Transfer rate	f_{PSCIN} = 20 MHz			f_{PSCIN} = 25 MHz			f_{PSCIN} = 33 MHz		
(bps)	RLD	dr	Error (%)	RLD	dr	Error (%)	RLD	dr	Error (%)
300	129	1/16	0.16025	162	1/16	-0.14698	216	1/16	0.00640
1200	129	1/4	0.16025	162	1/4	-0.14698	216	1/4	0.00640
2400	129	1/2	0.16025	162	1/2	-0.14698	216	1/2	0.00640
4800	64	1/2	0.16025	80	1/2	-0.46939	108	1/2	-0.45234
9600	32	1/2	-1.35732	40	1/2	-0.75584	53	1/2	0.46939
14400	21	1/2	-1.35732	13	1/4	-3.11880	35	1/2	0.46939
28800	10	1/2	-1.35732	13	1/2	-3.11880	17	1/2	0.46939

Make sure the error is within 1%. Calculate the error using the following equation:

$$Error = \left\{ \frac{f_{PSCIN} \times dr}{(RLD + 1) \times 32 \times bps} - 1 \right\} \times 100 \ [\%]$$

For details on how to control the prescaler and 8-bit programmable timers, refer to "Prescaler" and "8-Bit Programmable Timers".

- **External clock**

When an external clock is selected, the serial interface is clocked by a clock input from the #SCLKx pin. Therefore, there is no need to control the prescaler and 8-bit programmable timers.

Any desired clock frequency can be set. The clock input from the #SCLKx pin is internally divided by 16 or 8 in the serial interface, in order to create a sampling clock (refer to "Sampling clock"). This division ratio must also be considered when setting the transfer rate.

- **Sampling clock**

In the asynchronous mode, TCLK (the clock output by the 8-bit programmable timer or input from the #SCLKx pin) is internally divided in the serial interface, in order to create a sampling clock.
A 1/16 division ratio is selected by writing "1" to DIVMDx , and a 1/8 ratio is selected by writing "0".
Ch.0 clock division ratio selection: DIVMD0 (D4) / Serial I/F Ch.0 IrDA register (0x401E4)
Ch.1 clock division ratio selection: DIVMD1 (D4) / Serial I/F Ch.1 IrDA register (0x401E9)

Note: The DIVMDx bit becomes indeterminate at initial reset, so be sure to reset it in the software. Settings of this bit are valid only in the asynchronous mode (and when using the IrDA interface).

For receiving

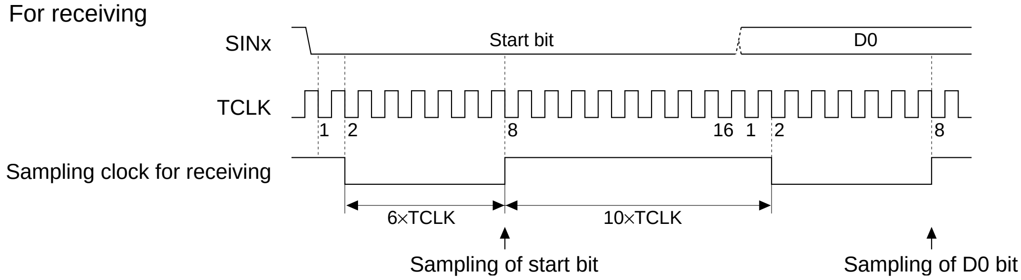


Figure 8.10 Sampling Clock for Asynchronous Receive Operation (when 1/16 division is selected)

As shown in Figure 8.10, the sampling clock is created by dividing TCLK by 16 (or 8). Its duty ratio (low: high ratio) is 6:10 (or 2:6 when divided by 8), and not 50%. Since the receive data is sampled in the middle point of each bit, the sampling clock recognizes the start bit first, and then changes the level from high to low at the second falling edge of TCLK. And at the 8th (4th for 1/8) falling edge of TCLK, it changes the level from low to high. This change in levels is repeated for the following bits of data:
Each bit of data is sampled at each rising edge of this sampling clock. When the stop bit is sampled, the sampling clock is fixed at high level until the next start bit is sampled.
If the SINx pin is returned to high level at the second falling edge of TCLK when it recognize the start bit, the data is assumed to be noise, and generation of the sampling clock is stopped.
If the SINx pin is not on a low level when the start bit is sampled at the 8th (4th for 1/8) clock, such as when the baud rate is not matched between the transmit and receive units, the serial interface stops sampling the following data and returns to a start-bit detection mode. In this case, no error is generated.

For transmitting

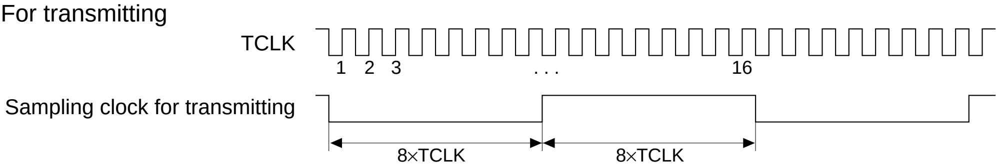


Figure 8.11 Sampling Clock for Asynchronous Transmit Operation (when 1/16 division is selected)

When transmitting data, a sampling clock of a 50% duty cycle is generated from TCLK by dividing it by 16 (or 8), and each bit of data is output synchronously with this clock.

Setting the data format

In the asynchronous mode, the data length is 7 or 8 bits as determined by the transfer mode set. The start bit is fixed at 1.

The stop and parity bits can be set as shown in the Table 8.5 using the following control bits:

Stop-bit selection	Ch.0:	STPB0 (D3) / Serial I/F Ch.0 control register (0x401E3)
	Ch.1:	STPB1 (D3) / Serial I/F Ch.1 control register (0x401E8)
Parity enable	Ch.0:	EPR0 (D5) / Serial I/F Ch.0 control register (0x401E3)
	Ch.1:	EPR1 (D5) / Serial I/F Ch.1 control register (0x401E8)
Parity-mode selection	Ch.0:	PMD0 (D4) / Serial I/F Ch.0 control register (0x401E3)
	Ch.1:	PMD1 (D4) / Serial I/F Ch.1 control register (0x401E8)

Table 8.5 Stop Bit and Parity Bit Settings

STPBx	EPRx	PMDx	Stop bit	Parity bit
1	1	1	2 bits	Odd
		0	2 bits	Even
	0	∗	2 bits	None
0	1	1	1 bit	Odd
		0	1 bit	Even
	0	∗	1 bit	Non

∗ Setting PMDx is invalid when EPRx = "0".

Note: These bits become indeterminate at initial reset, so be sure to initialize them in the software.

Control and Operation of Asynchronous Transfer

Transmit control

(1) Enabling transmit operation

Use the transmit-enable bit TXENx for transmit control.

Ch.0 transmit-enable: TXEN0 (D7) / Serial I/F Ch.0 control register (0x401E3)

Ch.1 transmit-enable: TXEN1 (D7) / Serial I/F Ch.1 control register (0x401E8)

When transmit is enabled by writing "1" to this bit, the clock input to the shift register is enabled (ready for input), thus allowing data to be transmitted.

Transmit is disabled by writing "0" to TXENx.

Note: Do not set TXENx to "0" during a transmit operation.

(2) Transmit procedure

The serial interface has a transmit shift register and a transmit data register (transmit data buffer) that are provided independently of those used for receive operations.

Ch.0 transmit data: TXD0[7:0] (D[7:0]) / Serial I/F Ch.0 transmit data register (0x401E0)

Ch.1 transmit data: TXD1[7:0] (D[7:0]) / Serial I/F Ch.1 transmit data register (0x401E5)

The serial interface starts a transmit operation by writing data to this register. In the 7-bit asynchronous mode, bit 7 (MSB) in each register is ignored.

The serial interface also contains a status bit to indicate the status of the transmit data register.

Ch.0 transmit data buffer empty: TDBE0 (D1) / Serial I/F Ch.0 status register (0x401E2)

Ch.1 transmit data buffer empty: TDBE1 (D1) / Serial I/F Ch.1 status register (0x401E7)

This bit is reset to "0" by writing data to the transmit data register, and set back to "1" (buffer empty) when the data is transferred to the shift register. The transfer begins when the serial interface starts sending the start bit.

The transfer status can be checked using the transmit-completion flag (TENDx).

Ch.0 transmit-completion flag: TEND0 (D5) / Serial I/F Ch.0 status register (0x401E2)

Ch.1 transmit-completion flag: TEND1 (D5) / Serial I/F Ch.1 status register (0x401E7)

This bit goes "1" when data is being transmitted and goes "0" when the transmission has completed.

When data is transmitted successively in asynchronous mode, TENDx maintains "1" until all data is transmitted.

Figure 8.12 shows a transmit timing chart in the asynchronous mode.

Example: Data length 8 bits
Stop bit 1 bit
Parity bit Included

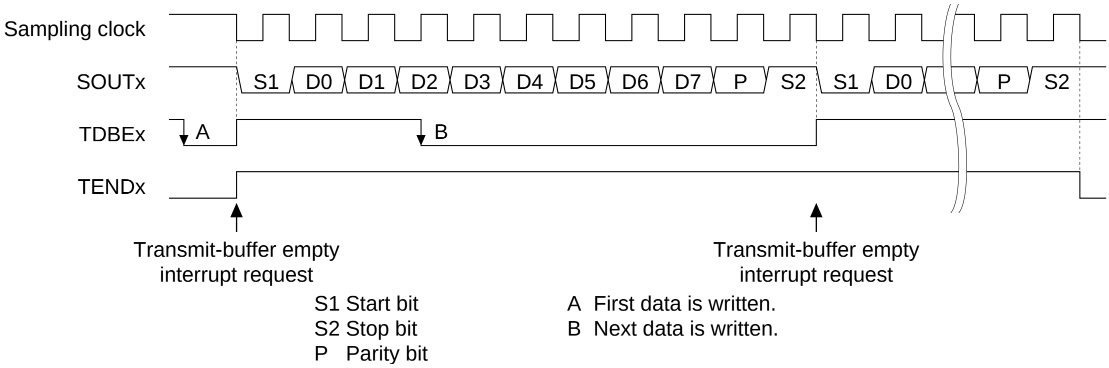


Figure 8.12 Transmit Timing Chart in Asynchronous Mode

1. The contents of the data register are transferred to the shift register synchronously with the first falling edge of the sampling clock. At the same time, the SOUTx pin is setting to a low level to send the start bit.
2. Each bit of data in the shift register is transmitted beginning with the LSB at each falling edge of the subsequent sampling clock. This operation is repeated until all 8 (or 7) bits of data are transmitted.
3. After sending the MSB, the parity bit (if EPRx = "1") and the stop bit are transmitted insuccession.

- Successive transmit operation

When the data in the transmit data register is transferred to the shift register, TDBEx is reset to "1" (buffer empty). Once this occurs, the next transmit data can be written to the transmit data register, even during data transmission.
This allows data to be transmitted successively. The transmit procedure is described above.
When TDBEx is set to "1", a transmit-data empty interrupt factor simultaneously occurs. Since an interrupt can be generated as set by the interrupt controller, the next transmit data can be written using an interrupt processing routine. In addition, since this interrupt factor can be used to invoke IDMA, the data prepared in memory can be transmitted successively to the transmit data register through DMA transfers.
For details on how to control interrupts and IDMA requests, refer to "Serial Interface Interrupts and DMA".

(3) Terminating transmit operations

When data transmission is completed, write "0" to the transmit-enable bit TXENx to disable transmit operations.

Receive control

(1) Enabling receive operations

Use the receive-enable bit RXENx for receive control.
Ch.0 receive-enable: RXEN0 (D6) / Serial I/F Ch.0 control register (0x401E3)
Ch.1 receive-enable: RXEN1 (D6) / Serial I/F Ch.1 control register (0x401E8)
When receiving enabled by writing "1" to this bit, clock input to the shift register is enabled (ready for input), meaning that it is ready to receive data.
Receive operations are disabled by writing "0" to RXENx.

Note: Do not set RXENx to "0" during a receive operation.

(2) Receive procedure

This serial interface has a receive shift register and a receive data register (receive data buffer) that are provided independently of those used for transmit operations.
Ch.0 receive data: RXD0[7:0] (D[7:0]) / Serial I/F Ch.0 receive data register (0x401E1)
Ch.1 receive data: RXD1[7:0] (D[7:0]) / Serial I/F Ch.1 receive data register (0x401E6)
Receive data can be read out from this register.

A status bit is also provided to indicate the status of the receive data register.
Ch.0 receive data buffer full: RDBF0 (D0) / Serial I/F Ch.0 status register (0x401E2)
Ch.1 receive data buffer full: RDBF1 (D0) / Serial I/F Ch.1 status register (0x401E7)
This bit is set to "1" (buffer full) when data is transferred from the shift register to the receive data register after the stop bit is sampled (the second bit if two stop bits are used), indicating that the received data can be read out. When the data is read out, the bit is reset to "0".

Figure 8.13 shows a receive timing chart in the asynchronous mode.

Example: Data length 8 bits
Stop bit 1 bit
Parity bit Included

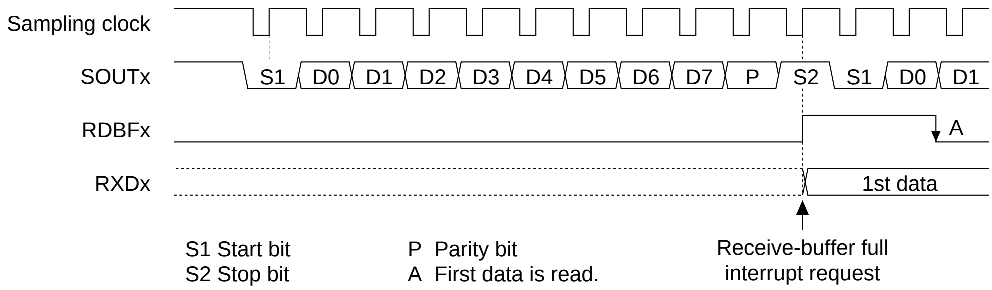


Figure 8.13 Receive Timing Chart in Asynchronous Mode

1. The serial interface starts sampling when the start bit is input (SINx = low).
2. When the start bit is sampled at the first rising edge of the sampling clock, each bit of receive data is taken into the shift register, beginning with the LSB at each rising edge of the subsequent clock. This operation is repeated until the MSB of data is received.
3. When the MSB is taken in, the parity bit that follows is also taken in (if EPRx = "1").
4. When the stop bit is sampled, the data in the shift register is transferred to the receive data register, enabling the data to be read out.
   The parity is checked when data is transferred to the receive data register (if EPRx = "1").

Note: The receive operation is terminated when the first stop bit is sampled even if the stop bit is configured with two bits.

- Successive receive operations

When the data received in the shift register is transferred to the receive data register, RDBFx is set to "1" (buffer full), indicating that the received data can be read out. Thereafter, data can be received successively because the receive data register can be read out while the next data is received. The procedure for receiving is described above.
When RDBFx is set to "1", a receive-data full interrupt factor occurs. Since an interrupt can be generated as set by the interrupt controller, the received data can be read using an interrupt processing routine. In addition, since this interrupt factor can be used to invoke IDMA, the received data can be received successively in locations prepared in memory through DMA transfers.
For details on how to control interrupts and IDMA requests, refer to "Serial Interface Interrupts and DMA".

(3) Receive errors

Three types of receive errors can be detected when receiving data in the asynchronous mode.
Since an interrupt can be generated by setting the interrupt controller, the error can be processed using an interrupt processing routine. For details on receive error interrupts, refer to "Serial Interface Interrupts and DMA".

- Parity error

If EPRx is set to "1" (parity added), the parity is checked when data is received.
This parity check is performed when the data received in the shift register is transferred to the receive data register in order to check conformity with PMDx settings (odd or even parity). If any nonconformity is found in this check, a parity error is assumed and the parity error flag is set to "1".
Ch.0 parity error flag: PER0 (D3) / Serial I/F Ch.0 status register (0x401E2)
Ch.1 parity error flag: PER1 (D3) / Serial I/F Ch.1 status register (0x401E7)
Even when this error occurs, the received data in error is transferred to the receive data register and the receive operation is continued. However, the content of the received data for which a parity error is flagged cannot be guaranteed.
The PERx flag is reset to "0" by writing "0".

- Framing error

If data with a stop bit = "0" is received, the serial interface assumes that the data is out of synchronization and generates a framing error.
If two stop bits are used, only the first stop bit is checked.
When this error occurs, the framing-error flag is set to "1".
Ch.0 framing-error flag: FER0 (D4) / Serial I/F Ch.0 status register (0x401E2)
Ch.1 framing-error flag: FER1 (D4) / Serial I/F Ch.1 status register (0x401E7)
Even when this error occurs, the received data in error is transferred to the receive data register and the receive operation is continued. However, the content of the received data for which a framing error is flagged cannot be guaranteed, even if no framing error is found in the following data received.
The FERx flag is reset to "0" by writing "0".

- Overrun error

If during successive receive operations, a receive operation for the next data is completed before the receive data register is read out, the receive data register is overwritten with the new data. Therefore, the receive data register must always be read out before a receive operation for the next data is completed.
When the receive data register is overwritten, an overrun error is generated and the overrun-error flag is set to "1".
Ch.0 overrun-error flag: OER0 (D2) / Serial I/F Ch.0 status register (0x401E2)
Ch.1 overrun-error flag: OER1 (D2) / Serial I/F Ch.1 status register (0x401E7)
Even when this error occurs, the received data in error is transferred to the receive data register and the receive operation is continued.
The OERx flag is reset to "0" by writing "0".

(4) Terminating receive operation

When a data receive operation is completed, write "0" to the receive-enable bit RXENx to disable receive operations.

IrDA Interface

Outline of IrDA Interface

Each channel of the serial interface contains a PPM modulator circuit, allowing an infrared-ray communication circuit to be configured based on IrDA 1.0 simply by adding a simple external circuit.

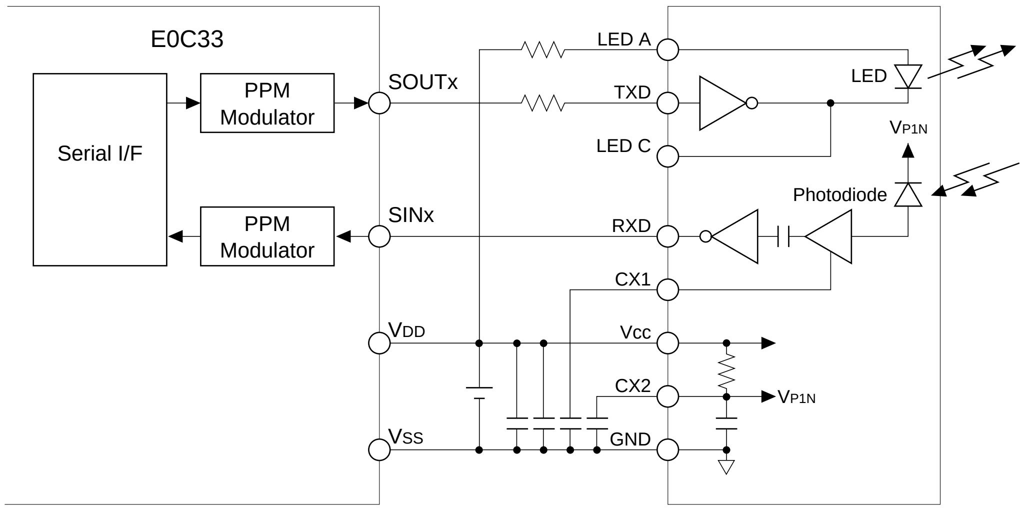


Figure 8.14 Configuration Example of IrDA Interface

This IrDA interface function can be used only when the selected transfer mode is an asynchronous mode.
Since the contents of the asynchronous mode are applied directly for the serial-interface functions other than the IrDA interface unit, refer to "Asynchronous Interface", for details on how to set and control the data formats and data transfers.

Setting IrDA Interface

When performing infrared-ray communication, the following settings must be made before communication can be started:

1. Setting input/output pins
2. Selecting the interface mode (IrDA interface function)
3. Setting the transfer mode
4. Setting the input clock
5. Setting the data format
6. Setting the interrupt/IDMA
7. Setting the input/output logic

The contents for items 1 through 5 have been explained in connection with the asynchronous interface. For details, refer to "Asynchronous Interface". For details on item 6, refer to "Serial Interface Interrupts and DMA".

Note: Before making these settings, always make sure the serial interface is inactive (TXENx and RXENx are both set to "0"), as a change in settings during operation could cause a malfunction.
In addition, be sure to set the transfer mode in (3) and the following items before selecting the IrDA interface function in (2).

Selecting the IrDA interface function

To use the IrDA interface function, select it using the control bits shown below and then set the 8-bit (or 7-bit) asynchronous mode as the transfer mode.

Ch.0 IrDA interface-function selection: IRMD0[1:0] (D[1:0]) / Serial I/F Ch.0 IrDA register (0x401E4)
Ch.1 IrDA interface-function selection: IRMD1[1:0] (D[1:0]) / Serial I/F Ch.1 IrDA register (0x401E9)

Table 8.6 Setting of IrDA Interface

IRMDx1	IRMDx0	Interface mode
1	1	Do not set. (reserved)
1	0	IrDA 1.0 interface
0	1	Do not set. (reserved)
0	0	Normal interface

Note: The IRMDx bit becomes indeterminate when initially reset, so be sure to initialize it in the software.

Setting the input/output logic

When using the IrDA interface, the logic of the input/output signals of the PPM modulator circuit can be changed in accordance with the infrared-ray communication module or the circuit connected externally to the chip. The logic of the internal serial interface is "active-low". If the input/output signals are active-high, the logic of these signals must be inverted before they can be used. The input SINx and output SOUTx logic can be set individually through the use of the IRRLx and IRTLx bits, respectively.

IrDA input logic inversion Ch.0: IRRL0 (D2) / Serial I/F Ch.0 IrDA register (0x401E4)
Ch.1: IRRL1 (D2) / Serial I/F Ch.1 IrDA register (0x401E9)
IrDA output logic inversion Ch.0: IRTL0 (D3) / Serial I/F Ch.0 IrDA register (0x401E4)
Ch.1: IRTL1 (D3) / Serial I/F Ch.1 IrDA register (0x401E9)

The logic of the input/output signal is inverted by writing "1" to each corresponding bit. Logic is not inverted if the bit is set to "0".

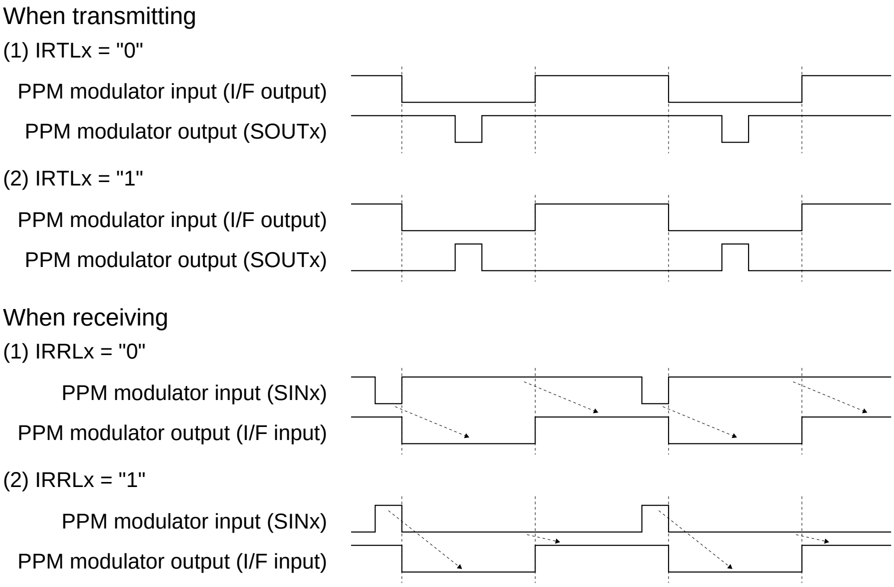


Figure 8.15 IRRLx and IRTLx Settings

Note: The IRRLx and IRTLx bits become indeterminate at initial reset, so be sure to initialize them in the software.

Control and Operation of IrDA Interface

The transmit/receive procedures have been explained in the section on the asynchronous interface, so refer to "Control and Operation of Asynchronous Transfer".

The following describes the data modulation and demodulation performed using the PPM modulator circuit:

When transmitting

During data transmission, the pulse width of the serial interface output signal is set to 3/16 before the signal is output from the SOUTx pin.

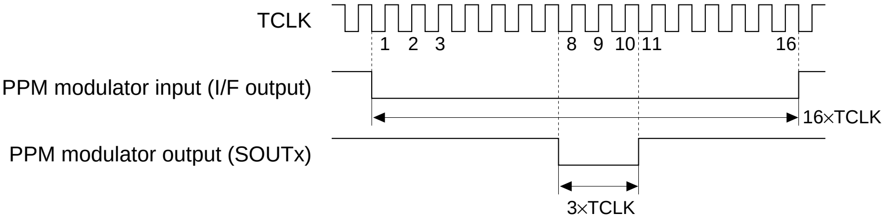


Figure 8.16 Data Modulation by PPM Circuit

When receiving

During data reception, the pulse width of the input signal from SINx is set to 16/3 before the signal is transferred to the serial interface.

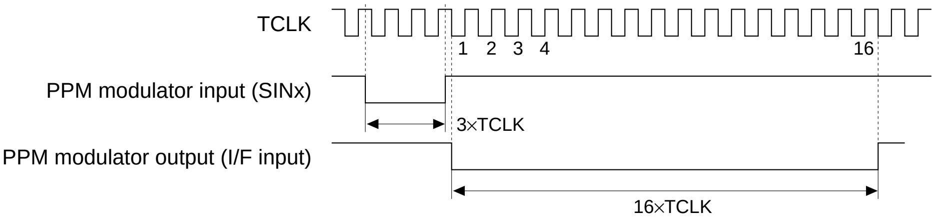


Figure 8.17 Demodulation by PPM Circuit

Note: When using the IrDA interface, set the internal division ratio of the serial interface 1/16 (DIVMDx = "1"), rather than 1/8 (DIVMDx = "0").

Serial Interface Interrupts and DMA

The serial interface can generate the following three types of interrupts in each channel:
• Transmit-buffer empty interrupt
• Receive-buffer full interrupt
• Receive-error interrupt

Transmit-buffer empty interrupt factor

This interrupt factor occurs when the transmit data set in the transmit data register is transferred to the shift register, in which case the interrupt factor flag FSTXx is set to "1". At this time, if the interrupt conditions set using the interrupt control register are met, an interrupt to the CPU is generated.
Occurrence of this interrupt factor indicates that the next transmit data can be written to the transmit data register.
This interrupt factor can also be used to invoke IDMA, enabling transmit data to be written to the register by means of a DMA transfer.

Receive-completion interrupt

This interrupt factor occurs when a receive operation is completed and the receive data taken into the shift register is transferred to the receive data register, in which case the interrupt factor flag FSRXx is set to "1". At this time, if the interrupt conditions set using the interrupt control register are met, an interrupt to the CPU is generated. Occurrence of this interrupt factor indicates that the received data can be read out.
This interrupt factor can also be used to invoke IDMA, enabling the received data to be written into specified memory locations by means of a DMA transfer.

Receive-error interrupt

This interrupt factor occurs when a parity, framing, or overrun error is detected during data reception, in which case the interrupt factor flag FSERRx is set to "1". At this time, if the interrupt conditions set using the interrupt control register are met, an interrupt to the CPU is generated.
Since all three types of errors generate the same interrupt factor, check the error flags PERx (parity error), OERx (overrun error), and FERx (framing error) to identify the type of error that has occurred. In the clock-synchronized mode, parity and framing errors do not occur.

Note: If a receive error (parity or framing error) occurs, the receive-error interrupt and receive-buffer full interrupt factors occur simultaneously. However, since the receive-error interrupt has priority over the receive-buffer full interrupt, the receive-error interrupt is processed first. It is therefore necessary for the receive-buffer full interrupt factor flag be cleared through the use of the receive-error interrupt processing routine.

Control registers of the interrupt controller

Table 8.7 shows the interrupt controller's control registers provided for each interrupt source (channel).

Table 8.7 Control Register of Interrupt Controller

Channel	Interrupt factor	Interrupt factor flag	Interrupt enable register	Interrupt priority register
Ch.0	Receive-error interrupt	FSERR0(D0/0x40286)	ESERR0(D0/0x40276)	PSIO0[2:0](D[6:4]/0x40269)
	Receive-buffer full	FSRX0(D1/0x40286)	ESRX0(D1/0x40276)	
	Transmit-buffer empty	FSTX0(D2/0x40286)	ESTX0(D2/0x40276)	
Ch.1	Receive-error interrupt	FSERR1(D3/0x40286)	ESERR1(D3/0x40276)	PSIO1[2:0](D[2:0]/0x4026A)
	Receive-buffer full	FSRX1(D4/0x40286)	ESRX1(D4/0x40276)	
	Transmit-buffer empty	FSTX1(D5/0x40286)	ESTX1(D5/0x40276)	

When the interrupt factor described above occurs, the corresponding interrupt factor flag is set to "1". If the interrupt enable register bit for that interrupt factor has been set to "1", an interrupt request is generated.
Interrupts caused by an interrupt factor can be disabled by leaving the interrupt enable register bit for that factor set to "0". The interrupt factor flag is set to "1" whenever interrupt conditions are met, regardless of the setting of the interrupt enable register (even if it is set to "0").

The interrupt priority register sets the interrupt priority level of each interrupt source in a range between 0 and 7. An interrupt request to the CPU is accepted only when no other interrupt request of a higher priority has been generated.
In addition, only when the PSR's IE bit = "1" (interrupts enabled) and the set value of the IL is smaller than the input interrupt level set by the interrupt priority register, will the input interrupt request actually be accepted by the CPU.
For details on these interrupt control registers, as well as the device operation when an interrupt has occurred, refer to "ITC (Interrupt Controller)".

Intelligent DMA

The receive-buffer full interrupt and transmit-buffer empty interrupt factors can be used to invoke intelligent DMA (IDMA). This enables successive transmit/receive operations between memory and the transmit/receive-buffer to be performed by means of a DAM transfer.
The following shows the IDMA channel numbers set for each interrupt factor:

	IDMA Ch.
Ch.0 receive-buffer full interrupt:	0x17
Ch.0 transmit-buffer empty interrupt:	0x18
Ch.1 receive-buffer full interrupt:	0x19
Ch.1 transmit-buffer empty interrupt:	0x1A

The IDMA request and enable bits shown in Table 8.8 must be set to "1" for IDMA to be invoked. Transfer conditions, etc. on the IDMA side must also be set in advance.

Table 8.8 Control Bits for IDMA Transfer

Channel	Interrupt factor	IDMA request bit	IDMA enable bit
Ch.0	Receive-buffer full	RSRX0(D6/0x40292)	DESRX0(D6/0x40296)
	Transmit-buffer empty	RSTX0(D7/0x40292)	DESTX0(D7/0x40296)
Ch.1	Receive-buffer full	RSRX1(D0/0x40293)	DESRX1(D0/0x40297)
	Transmit-buffer empty	RSTX1(D1/0x40293)	DESTX1(D1/0x40297)

If an interrupt factor occurs when the IDMA request and enable bits are set to "1", IDMA is invoked. No interrupt request is generated at that point. An interrupt request is generated upon completion of the DMA transfer. The bits can also be set so as not to generate an interrupt, with only a DAM transfer performed.
For details on DMA transfer and how to control interrupts upon completion of DMA transfer, refer to "IDMA (Intelligent DMA)".

High-speed DMA

The receive-buffer full interrupt and transmit-buffer empty interrupt factors can also invoke high-speed DMA (HSDMA).
The following shows the HSDMA channel number and trigger set-up bit corresponding to each channel:

Table 8.9 HSDMA Trigger Set-up Bits

SIF Ch.	HSDMA Ch.	Trigger set-up bits
0	0	HSD0S[3:0] (D[3:0]) / HSDMA Ch.0/1 trigger set-up register (0x40298)
1	1	HSD1S[3:0] (D[7:4]) / HSDMA Ch.0/1 trigger set-up register (0x40298)
0	2	HSD2S[3:0] (D[3:0]) / HSDMA Ch.2/3 trigger set-up register (0x40299)
1	3	HSD3S[3:0] (D[7:4]) / HSDMA Ch.2/3 trigger set-up register (0x40299)

For HSDMA to be invoked by the receive-buffer full interrupt factor, the trigger set-up bits should be set to "1010". For HSDMA to be invoked by the transmit-buffer empty interrupt factor, the trigger set-up bits should be set to "1011". Transfer conditions, etc. must also be set on the HSDMA side.
The HSDMA channel is invoked through generation of the interrupt factor.
For details on HSDMA transfer, refer to "HSDMA (High-Speed DMA)".

Trap vectors

The trap-vector address of each default interrupt factor is set as follows:

	(BTA3=high)	(BTA3=low)
Ch.0 receive-error interrupt:	0x00800E0	0x0C000E0
Ch.0 receive-buffer full interrupt:	0x00800E4	0x0C000E4
Ch.0 transmit-buffer empty interrupt:	0x00800E8	0x0C000E8
Ch.1 receive-error interrupt:	0x00800EC	0x0C000EC
Ch.1 receive-buffer full interrupt:	0x00800F0	0x0C000F0
Ch.1 transmit-buffer empty interrupt:	0x00800F4	0x0C000F4

The base address of the trap table can be changed using the TTBR register (0x48134 to 0x48137).

I/O Memory of Serial Interface

Table 8.10 shows the control bits of the serial interface.
For details on the I/O memory of the prescaler that is used to set clocks, as well of that of 8-bit programmable timers, refer to "Prescaler" and "8-Bit Programmable Timers", respectively.

Table 8.10 Control Bits of Serial Interface

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Serial I/F Ch.0	**00401E0**	D7	**TXD07**	Serial I/F Ch.0 transmit data	0x0 to 0xFF(0x7F)	X	R/W	7-bit asynchronous
transmit data	**(B)**	D6	**TXD06**	TXD07(06) = MSB		X		mode does not use
register		D5	**TXD05**	TXD00 = LSB		X		TXD07.
		D4	**TXD04**			X		
		D3	**TXD03**			X		
		D2	**TXD02**			X		
		D1	**TXD01**			X		
		D0	**TXD00**			X		
Serial I/F Ch.0	**00401E1**	D7	**RXD07**	Serial I/F Ch.0 receive data	0x0 to 0xFF(0x7F)	X	R	7-bit asynchronous
receive data	**(B)**	D6	**RXD06**	RXD07(06) = MSB		X		mode does not use
register		D5	**RXD05**	RXD00 = LSB		X		RXD07 (fixed at 0).
		D4	**RXD04**			X		
		D3	**RXD03**			X		
		D2	**RXD02**			X		
		D1	**RXD01**			X		
		D0	**RXD00**			X		
Serial I/F Ch.0	**00401E2**	D7–6	**–**	–	–	–	–	0 when being read.
status register	**(B)**	D5	**TEND0**	Ch.0 transmit-completion flag	1 Transmitting / 0 End	0	R	
		D4	**FER0**	Ch.0 flaming error flag	1 Error / 0 Normal	0	R/W	Reset by writing 0.
		D3	**PER0**	Ch.0 parity error flag	1 Error / 0 Normal	0	R/W	Reset by writing 0.
		D2	**OER0**	Ch.0 overrun error flag	1 Error / 0 Normal	0	R/W	Reset by writing 0.
		D1	**TDBE0**	Ch.0 transmit data buffer empty	1 Empty / 0 Buffer full	1	R	
		D0	**RDBF0**	Ch.0 receive data buffer full	1 Buffer full / 0 Empty	0	R	
Serial I/F Ch.0	**00401E3**	D7	**TXEN0**	Ch.0 transmit enable	1 Enabled / 0 Disabled	0	R/W	
control register	**(B)**	D6	**RXEN0**	Ch.0 receive enable	1 Enabled / 0 Disabled	0	R/W	
		D5	**EPR0**	Ch.0 parity enable	1 With parity / 0 No parity	X	R/W	Valid only in
		D4	**PMD0**	Ch.0 parity mode selection	1 Odd / 0 Even	X	R/W	asynchronous mode.
		D3	**STPB0**	Ch.0 stop bit selection	1 2 bits / 0 1 bit	X	R/W	
		D2	**SSCK0**	Ch.0 input clock selection	1 #SCLK0 / 0 Internal clock	X	R/W	
		D1	**SMD01**	Ch.0 transfer mode selection	SMD0[1:0] / Transfer mode	X	R/W	
		D0	**SMD00**		1 1 8-bit asynchronous	X		
					1 0 7-bit asynchronous			
					0 1 Clock sync. Slave			
					0 0 Clock sync. Master			
Serial I/F Ch.0	**00401E4**	D7–5	**–**	–	–	–	–	0 when being read.
IrDA register	**(B)**	D4	**DIVMD0**	Ch.0 async. clock division ratio	1 1/8 / 0 1/16	X	R/W	
		D3	**IRTL0**	Ch.0 IrDA I/F output logic inversion	1 Inverted / 0 Direct	X	R/W	Valid only in
		D2	**IRRL0**	Ch.0 IrDA I/F input logic inversion	1 Inverted / 0 Direct	X	R/W	asynchronous mode.
		D1	**IRMD01**	Ch.0 interface mode selection	IRMD0[1:0] / I/F mode	X	R/W	
		D0	**IRMD00**		1 1 reserved	X		
					1 0 IrDA 1.0			
					0 1 reserved			
					0 0 General I/F			
Serial I/F Ch.1	**00401E5**	D7	**TXD17**	Serial I/F Ch.1 transmit data	0x0 to 0xFF(0x7F)	X	R/W	7-bit asynchronous
transmit data	**(B)**	D6	**TXD16**	TXD17(16) = MSB		X		mode does not use
register		D5	**TXD15**	TXD10 = LSB		X		TXD17.
		D4	**TXD14**			X		
		D3	**TXD13**			X		
		D2	**TXD12**			X		
		D1	**TXD11**			X		
		D0	**TXD10**			X		
Serial I/F Ch.1	**00401E6**	D7	**RXD17**	Serial I/F Ch.1 receive data	0x0 to 0xFF(0x7F)	X	R	7-bit asynchronous
receive data	**(B)**	D6	**RXD16**	RXD17(16) = MSB		X		mode does not use
register		D5	**RXD15**	RXD10 = LSB		X		RXD17 (fixed at 0).
		D4	**RXD14**			X		
		D3	**RXD13**			X		
		D2	**RXD12**			X		
		D1	**RXD11**			X		
		D0	**RXD10**			X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Serial I/F Ch.1	**00401E7**	D7–6	–	–	–				–	–	0 when being read.
status register	**(B)**	D5	**TEND1**	Ch.1 transmit-completion flag	1	Transmitting	0	End	0	R	
		D4	**FER1**	Ch.1 flaming error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D3	**PER1**	Ch.1 parity error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D2	**OER1**	Ch.1 overrun error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D1	**TDBE1**	Ch.1 transmit data buffer empty	1	Empty	0	Buffer full	1	R	
		D0	**RDBF1**	Ch.1 receive data buffer full	1	Buffer full	0	Empty	0	R	
Serial I/F Ch.1	**00401E8**	D7	**TXEN1**	Ch.1 transmit enable	1	Enabled	0	Disabled	0	R/W	
control register	**(B)**	D6	**RXEN1**	Ch.1 receive enable	1	Enabled	0	Disabled	0	R/W	
		D5	**EPR1**	Ch.1 parity enable	1	With parity	0	No parity	X	R/W	Valid only in
		D4	**PMD1**	Ch.1 parity mode selection	1	Odd	0	Even	X	R/W	asynchronous mode.
		D3	**STPB1**	Ch.1 stop bit selection	1	2 bits	0	1 bit	X	R/W	
		D2	**SSCK1**	Ch.1 input clock selection	1	#SCLK1	0	Internal clock	X	R/W	
		D1	**SMD11**	Ch.1 transfer mode selection	SMD1[1:0]			Transfer mode	X	R/W	
		D0	**SMD10**		1	1		8-bit asynchronous	X		
					1	0		7-bit asynchronous			
					0	1		Clock sync. Slave			
					0	0		Clock sync. Master			
Serial I/F Ch.1	**00401E9**	D7–5	–	–	–				–	–	0 when being read.
IrDA register	**(B)**	D4	**DIVMD1**	Ch.1 async. clock division ratio	1	1/8	0	1/16	X	R/W	
		D3	**IRTL1**	Ch.1 IrDA I/F output logic inversion	1	Inverted	0	Direct	X	R/W	Valid only in
		D2	**IRRL1**	Ch.1 IrDA I/F input logic inversion	1	Inverted	0	Direct	X	R/W	asynchronous mode.
		D1	**IRMD11**	Ch.1 interface mode selection	IRMD1[1:0]			I/F mode	X	R/W	
		D0	**IRMD10**		1	1		reserved	X		
					1	0		IrDA 1.0			
					0	1		reserved			
					0	0		General I/F			
8-bit timer,	**0040269**	D7	–	reserved	–				–	–	0 when being read.
serial I/F Ch.0	**(B)**	D6	**PSIO02**	Serial interface Ch.0	0 to 7				X	R/W	
interrupt		D5	**PSIO01**	interrupt level					X		
priority register		D4	**PSIO00**						X		
		D3	–	reserved	–				–	–	0 when being read.
		D2	**P8TM2**	8-bit timer 0–3 interrupt level	0 to 7				X	R/W	
		D1	**P8TM1**						X		
		D0	**P8TM0**						X		
Serial I/F Ch.1,	**004026A**	D7	–	reserved	–				–	–	0 when being read.
A/D interrupt	**(B)**	D6	**PAD2**	A/D converter interrupt level	0 to 7				X	R/W	
priority register		D5	**PAD1**						X		
		D4	**PAD0**						X		
		D3	–	reserved	–				–	–	0 when being read.
		D2	**PSIO12**	Serial interface Ch.1	0 to 7				X	R/W	
		D1	**PSIO11**	interrupt level					X		
		D0	**PSIO10**						X		
Serial I/F	**0040276**	D7–6	–	reserved	–				–	–	0 when being read.
interrupt	**(B)**	D5	**ESTX1**	SIF Ch.1 transmit buffer empty	1	Enabled	0	Disabled	0	R/W	
enable register		D4	**ESRX1**	SIF Ch.1 receive buffer full					0	R/W	
		D3	**ESERR1**	SIF Ch.1 receive error					0	R/W	
		D2	**ESTX0**	SIF Ch.0 transmit buffer empty					0	R/W	
		D1	**ESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D0	**ESERR0**	SIF Ch.0 receive error					0	R/W	
Serial I/F	**0040286**	D7–6	–	reserved	–				–	–	0 when being read.
interrupt factor	**(B)**	D5	**FSTX1**	SIF Ch.1 transmit buffer empty	1	Factor is generated	0	No factor is generated	X	R/W	
flag register		D4	**FSRX1**	SIF Ch.1 receive buffer full					X	R/W	
		D3	**FSERR1**	SIF Ch.1 receive error					X	R/W	
		D2	**FSTX0**	SIF Ch.0 transmit buffer empty					X	R/W	
		D1	**FSRX0**	SIF Ch.0 receive buffer full					X	R/W	
		D0	**FSERR0**	SIF Ch.0 receive error					X	R/W	
16-bit timer 5,	**0040292**	D7	**RSTX0**	SIF Ch.0 transmit buffer empty	1	IDMA request	0	Interrupt request	0	R/W	
8-bit timer,	**(B)**	D6	**RSRX0**	SIF Ch.0 receive buffer full					0	R/W	
serial I/F Ch.0		D5	**R8TU3**	8-bit timer 3 underflow					0	R/W	
IDMA request		D4	**R8TU2**	8-bit timer 2 underflow					0	R/W	
register		D3	**R8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**R8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**R16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**R16TU5**	16-bit timer 5 comparison B					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Serial I/F Ch.1, A/D, port input 4–7 IDMA request register	**0040293 (B)**	D7	**RP7**	Port input 7	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RP6**	Port input 6					0	R/W	
		D5	**RP5**	Port input 5					0	R/W	
		D4	**RP4**	Port input 4					0	R/W	
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**RADE**	A/D converter	1	IDMA request	0	Interrupt request	0	R/W	
		D1	**RSTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**RSRX1**	SIF Ch.1 receive buffer full					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register	**0040296 (B)**	D7	**DESTX0**	SIF Ch.0 transmit buffer empty	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**DE8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**DE8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**DE8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**DE8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**DE16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**DE16TU5**	16-bit timer 5 comparison B					0	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA enable register	**0040297 (B)**	D7	**DEP7**	Port input 7	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DEP6**	Port input 6					0	R/W	
		D5	**DEP5**	Port input 5					0	R/W	
		D4	**DEP4**	Port input 4					0	R/W	
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**DEADE**	A/D converter	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D1	**DESTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**DESRX1**	SIF Ch.1 receive buffer full					0	R/W	
P0 function select register	**00402D0 (B)**	D7	**CFP07**	P07 function selection	1	#SRDY1	0	P07	0	R/W	Extended functions (0x402DF)
		D6	**CFP06**	P06 function selection	1	#SCLK1	0	P06	0	R/W	
		D5	**CFP05**	P05 function selection	1	SOUT1	0	P05	0	R/W	
		D4	**CFP04**	P04 function selection	1	SIN1	0	P04	0	R/W	
		D3	**CFP03**	P03 function selection	1	#SRDY0	0	P03	0	R/W	
		D2	**CFP02**	P02 function selection	1	#SCLK0	0	P02	0	R/W	
		D1	**CFP01**	P01 function selection	1	SOUT0	0	P01	0	R/W	
		D0	**CFP00**	P00 function selection	1	SIN0	0	P00	0	R/W	
Port function extension register	**00402DF (B)**	D7	**CFEX7**	P07 port extended function	1	#DMAEND3	0	P07, etc.	0	R/W	
		D6	**CFEX6**	P06 port extended function	1	#DMAACK3	0	P06, etc.	0	R/W	
		D5	**CFEX5**	P05 port extended function	1	#DMAEND2	0	P05, etc.	0	R/W	
		D4	**CFEX4**	P04 port extended function	1	#DMAACK2	0	P04, etc.	0	R/W	
		D3	**CFEX3**	P31 port extended function	1	#GARD	0	P31, etc.	0	R/W	
		D2	**CFEX2**	P21 port extended function	1	#GAAS	0	P21, etc.	0	R/W	
		D1	**CFEX1**	P10, P11, P13 port extended function	1	DST0 DST1 DPC0	0	P10, etc. P11, etc. P13, etc.	1	R/W	
		D0	**CFEX0**	P12, P14 port extended function	1	DST2 DCLK	0	P12, etc. P14, etc.	1	R/W	

CFP07–CFP00: P0[7:0] pin function selection (D[7:0]) / P0 function select register (0x402D0)

Selects the pins used for the serial interface.

Write "1": Serial-interface input/output pin
Write "0": I/O port pin
Read: Valid

Select the pins used for the serial interface from among P00 through P07 by writing "1" to CFP00 through CFP07. P00–P03 (SIN0, SOUT0, #SCLK0, #SRDY0) are used for channel 0; P04–P07 (SIN1, SOUT1, #SCLK1, #SRDY1) are used for channel 1. If the bit for a pin is set to "0", the pin functions as an I/O port.
The necessary input/output pins differ depending on the transfer mode set (see Table 8.3).
At cold start, CFP is set to "0" (I/O port). At hot start, CFP retains its state from prior to the initial reset.

CFEX7–CFEX4: P0[7:4] pin function selection (D[7:4]) / Port function extension register (0x402DF)

Selects the extended function of pins P07–P04.

Write "1": Function-extended pin
Write "0": I/O-port/serial I/O pin
Read: Valid

When CFEX[7:4] is set to "1", the P07–P04 ports function as DMA signal output ports. When CFEX[7:4] = "0", the CFP0[7:4] bit becomes effective, so the settings of these bits determine whether the P07–P04 ports function as I/O port s or serial interface Ch.1 signal output ports.
At cold start, CFEX[7:4] is set to "0" (I/O-port/serial I/O pin). At hot start, CFEX[7:4] retains its state from prior to the initial reset.

TXD07–TXD00: Ch.0 transmit data (D[7:0]) / Serial I/F Ch.0 transmit data register (0x401E0)
TXD17–TXD10: Ch.1 transmit data (D[7:0]) / Serial I/F Ch.1 transmit data register (0x401E5)

Sets transmit data.
When data is written to this register (transmit buffer) after "1" is written to TXENx, a transmit operation is begun. TDBEx is set to "1" (transmit-buffer empty) when the data is transferred to the shift register. A transmit-buffer empty interrupt factor is simultaneously generated. The next transmit data can be written to the buffer at any time thereafter, even when the serial interface is sending data.
In the 7-bit asynchronous mode, TXDx7 (MSB) is ignored.
The serial-converted data is output from the SOUT pin beginning with the LSB, in which the bits set to "1" are output as high-level signals and those set to "0" output as low-level signals.
This register can be read as well as written.
At initial reset, the content of TXDx becomes indeterminate.

RXD07–RXD00: Ch.0 receive data (D[7:0]) / Serial I/F Ch.0 receive data register (0x401E1)
RXD17–RXD10: Ch.1 receive data (D[7:0]) / Serial I/F Ch.1 receive data register (0x401E6)

Stores received data.
When a receive operation is completed and the data received in the shift register is transferred to this register (receive buffer), RDBFx is set to "1" (receive buffer full). At the same time, a receive-buffer full interrupt factor is generated. Thereafter, the data can be read out at any time before a receive operation for the next data is completed.
If the next data receive operation is completed before this register is read out, the data in it is overwritten with the newly received data, causing an overrun error to occur.
In the 7-bit asynchronous mode, "0" is stored in RXDx7.
The serial data input from the SINx pin is converted into parallel data beginning with the LSB, with the high-level signals changed to "1"s and the low-level signals changed to "0"s. The resulting data is stored in this buffer.
This register is a read-only register, so no data can be written to it.
At initial reset, the content of RXDx becomes indeterminate.

TEND0: Ch.0 transmit-completion flag (D5) / Serial I/F Ch.0 status register (0x401E2)
TEND1: Ch.1 transmit-completion flag (D5) / Serial I/F Ch.1 status register (0x401E7)

Indicates the transmission status.

Read "1": During transmitting
Read "0": End of transmission
Write: Invalid

TENDx goes "1" when data is being transmitted and goes "0" when the transmission has completed.
When data is transmitted successively in clock-synchronized master mode or asynchronous mode, TENDx maintains "1" until all data is transmitted (see Figure 8.4 and Figure 8.12). In clock-synchronized slave mode, TENDx goes "0" every time 1-byte data is transmitted (see Figure 8.5).
At initial reset, TENDx is set to "0" (End of transmission).

FER0: Ch.0 framing-error flag (D4) / Serial I/F Ch.0 status register (0x401E2)
FER1: Ch.1 framing-error flag (D4) / Serial I/F Ch.1 status register (0x401E7)

Indicates whether a framing error occurred.

Read "1": An error occurred
Read "0": No error occurred
Write "1": Invalid
Write "0": Reset to "0"

The FERx flag is an error flag indicating whether a framing error occurred. When an error has occurred, it is set to "1". A framing error occurs when data with a stop bit = "0" is received in the asynchronous mode.
The FERx flag is reset by writing "0".
At initial reset, as well as when RXENx and TXENx both are set to "0", the FERx flag is set to "0" (no error).

PER0: Ch.0 parity-error flag (D3) / Serial I/F Ch.0 status register (0x401E2)
PER1: Ch.1 parity-error flag (D3) / Serial I/F Ch.1 status register (0x401E7)

Indicates whether a parity error occurred.

Read "1": An error occurred
Read "0": No error occurred
Write "1": Invalid
Write "0": Reset to "0"

The PERx flag is an error flag indicating whether a parity error occurred. When an error has occurred, it is set to "1". Parity checks are valid only in the asynchronous mode with EPRx set to "1" (parity added). This check is performed when the received data is transferred from the shift register to the receive data register.
The PERx flag is reset by writing "0".
At initial reset, as well as when RXENx and TXENx both are set to "0", PERx is set to "0" (no error).

OER0: Ch.0 overrun-error flag (D2) / Serial I/F Ch.0 status register (0x401E2)
OER1: Ch.1 overrun-error flag (D2) / Serial I/F Ch.1 status register (0x401E7)

Indicates whether an overrun error occurred.

Read "1": An error occurred
Read "0": No error occurred
Write "1": Invalid
Write "0": Reset to "0"

The OERx flag is an error flag indicating whether an overrun error occurred. When an error has occurred, it is set to "1". An overrun error occurs when the next receive operation is completed before the receive data register is read out, resulting in the receive data register being overwritten.
The OERx flag is reset by writing "0".
At initial reset, as well as when RXENx and TXENx both are set to "0", OERx is set to "0" (no error).

TDBE0: Ch.0 transmit data buffer empty (D1) / Serial I/F Ch.0 status register (0x401E2)
TDBE1: Ch.1 transmit data buffer empty (D1) / Serial I/F Ch.1 status register (0x401E7)

Indicates the status of the transmit data register (buffer).

Read "1": Buffer empty
Read "0": Buffer full
Write: Invalid

TDBEx is set to "0" when transmit data is written to the transmit data register, and is set to "1" when this data is transferred to the shift register (transmit operation started).
Transmit data is written to the transmit data register when this bit = "1".
At initial reset, TDBEx is set to "1" (buffer empty).

RDBF0: Ch.0 receive data buffer full (D0) / Serial I/F Ch.0 status register (0x401E2)
RDBF1: Ch.1 receive data buffer full (D0) / Serial I/F Ch.1 status register (0x401E7)

Indicates the status of the receive data register (buffer).

Read "1": Buffer full
Read "0": Buffer empty
Write: Invalid

RDBFx is set to "1" when the data received in the shift register is transferred to the receive data register (receive operation completed), indicating that the received data can be read out. This bit is reset to "0" when the data is read out.
At initial reset, RDBFx is set to "0" (buffer empty).

TXEN0: Ch.0 transmit enable (D7) / Serial I/F Ch.0 control register (0x401E3)
TXEN1: Ch.1 transmit enable (D7) / Serial I/F Ch.1 control register (0x401E8)

Enables each channel for transmit operations.

Write "1": Transmit enabled
Write "0": Transmit disabled
Read: Valid

When TXENx for a channel is set to "1", the channel is enabled for transmit operations. When TXENx is set to "0", the channel is disabled for transmit operations.
Always make sure the TXENx = "0" before setting the transfer mode and other conditions.
At initial reset, TXENx is set to "0" (transmit disabled).

RXEN0: Ch.0 receive enable (D6) / Serial I/F Ch.0 control register (0x401E3)
RXEN1: Ch.1 receive enable (D6) / Serial I/F Ch.1 control register (0x401E8)

Enables each channel for receive operations.

Write "1": Receive enabled
Write "0": Receive disabled
Read: Valid

When RXENx for a channel is set to "1", the channel is enabled for receive operations. When RXENx is set to "0", the channel is disabled for receive operations.
Always make sure the RXENx = "0" before setting the transfer mode and other conditions.
At initial reset, RXENx is set to "0" (receive disabled).

EPR0: Ch.0 parity enable (D5) / Serial I/F Ch.0 control register (0x401E3)
EPR1: Ch.1 parity enable (D5) / Serial I/F Ch.1 control register (0x401E8)

Selects a parity function.

Write "1": Parity added
Write "0": No parity added
Read: Valid

EPRx is used to select whether receive data is to be checked for parity, and whether a parity bit is to be added to transmit data. When EPRx is set to "1", the receive data is checked for parity. A parity bit is automatically added to the transmit data. When EPRx is set to "0", parity is not checked and no parity bit is added.
The parity function is only valid in the asynchronous mode. Settings of EPRx have no effect in the clock-synchronized mode.
At initial reset, EPRx becomes indeterminate.

PMD0: Ch.0 parity mode selection (D4) / Serial I/F Ch.0 control register (0x401E3)
PMD1: Ch.1 parity mode selection (D4) / Serial I/F Ch.1 control register (0x401E8)

Selects an odd or even parity.

Write "1": Odd parity
Write "0": Even parity
Read: Valid

Odd parity is selected by writing "1" to PMDx, and even parity is selected by writing "0". Parity check and the addition of a parity bit are only effective in asynchronous transfers in which EPRx is set to "1". If EPRx = "0", settings of PMDx do not have any effect.
At initial reset, PMDx becomes indeterminate.

STPB0: Ch.0 stop bit selection (D3) / Serial I/F Ch.0 control register (0x401E3)
STPB1: Ch.1 stop bit selection (D3) / Serial I/F Ch.1 control register (0x401E8)

Selects a stop-bit length during the performance of an asynchronous transfer.

Write "1": 2 bits
Write "0": 1 bit
Read: Valid

STPBx is only valid in an asynchronous transfer. Two stop bits are selected by writing "1" to STPBx , and one stop bit is selected by writing "0". The start bit is fixed at 1 bit.
Settings of STPBx are ignored during the performance of a clock-synchronized transfer.
At initial reset, STPBx becomes indeterminate.

SSCK0: Ch.0 input clock selection (D2) / Serial I/F Ch.0 control register (0x401E3)
SSCK1: Ch.1 input clock selection (D2) / Serial I/F Ch.1 control register (0x401E8)

Selects the clock source for an asynchronous transfer.

Write "1": #SCLK (external clock)
Write "0": Internal clock
Read: Valid

During operation in the asynchronous mode, this bit is used to select the clock source between an internal clock (output by an 8-bit programmable timer) and an external clock (input from the #SCLKx pin). An external clock is selected by writing "1" to this bit, and an internal clock is selected by writing "0".
At initial reset, SSCKx becomes indeterminate.

SMD01–SMD00: Ch.0 transfer mode selection (D[1:0]) / Serial I/F Ch.0 control register (0x401E3)
SMD11–SMD10: Ch.1 transfer mode selection (D[1:0]) / Serial I/F Ch.1 control register (0x401E8)

Sets the transfer mode of the serial interface as shown in Table 8.11 below.

Table 8.11 Setting of Transfer Mode

SMDx1	SMDx0	Transfer mode
1	1	8-bit asynchronous mode
1	0	7-bit asynchronous mode
0	1	Clock-synchronized slave mode
0	0	Clock-synchronized master mode

The SMDx bit can be read as well as written.
When using the IrDA interface, always be sure to set an asynchronous mode for the transfer mode.
At initial reset, SMDx becomes indeterminate.

DIVMD0: Sampling clock division ratio (D4) / Serial I/F Ch.0 IrDA register (0x401E4)
DIVMD1: Sampling clock division ratio (D4) / Serial I/F Ch.1 IrDA register (0x401E9)

Selects the division ratio of the sampling clock.

Write "1": 1/8
Write "0": 1/16
Read: Valid

Select the division ratio necessary to generate the sampling clock for asynchronous transfers. When DIVMDx is set to "1", the sampling clock is generated from the input clock of the serial interface (output by an 8-bit programmable timer or input from #SCLKx) by dividing it by 8. When DIVMDx is set to "0", the input clock is divided by 16.
At initial reset, DIVMDx becomes indeterminate.

IRTL0: Ch.0 IrDA output logic inversion (D3) / Serial I/F Ch.0 IrDA register (0x401E4)
IRTL1: Ch.1 IrDA output logic inversion (D3) / Serial I/F Ch.1 IrDA register (0x401E9)

Inverts the logic of the IrDA output signal.

Write "1": Inverted
Write "0": Not inverted
Read: Valid

When using the IrDA interface, set the logic of the SOUTx output signal to suit the infrared-ray communication circuit that is connected external to the chip. If IRTLx is set to "1", a high pulse is output when the output data = "0" (held low-level when the output data = "1"). If IRTLx is set to "0", a low pulse is output when the output data = "0" (held high-level when the output data = "1").
At initial reset, IRTLx becomes indeterminate.

IRRL0: Ch.0 IrDA input logic inversion (D2) / Serial I/F Ch.0 IrDA register (0x401E4)
IRRL1: Ch.1 IrDA input logic inversion (D2) / Serial I/F Ch.1 IrDA register (0x401E9)

Inverts the logic of the IrDA input signal.

Write "1": Inverted
Write "0": Not inverted
Read: Valid

When using the IrDA interface, set the logic of the signal that is input from an external infrared-ray communication circuit to the chip to suit the serial interface. If IRRLx is set to "1", a high pulse is input as a logic "0". If IRRLx is set to "0", a low pulse is input as a logic "0".
At initial reset, IRRLx becomes indeterminate.

IRMD01–IRMD00: Ch.0 IrDA interface mode selection (D[1:0]) / Serial I/F Ch.0 IrDA register (0x401E4)
IRMD11–IRMD10: Ch.1 IrDA interface mode selection (D[1:0]) / Serial I/F Ch.1 IrDA register (0x401E9)

Selects the IrDA interface function.

Table 8.12 IrDA Interface Setting

IRMDx1	IRMDx0	Interface mode
1	1	Do not set. (reserved)
1	0	IrDA 1.0 interface
0	1	Do not set. (reserved)
0	0	Normal interface

When using the IrDA interface function, write "10" to IRMDx while setting to an asynchronous mode for the transfer mode. If the IrDA interface function is not to be used, write "00" to IRMDx.
At initial reset, IRMDx becomes indeterminate.

Note: This selection must always be performed before the transfer mode and other conditions are set.

PSIO02–PSIO00: Ch.0 interrupt level (D[6:4]) / 8-bit timer, serial I/F Ch.0 interrupt priority register (0x40269)
PSIO12–PSIO10: Ch.1 interrupt level (D[2:0]) / Serial I/F Ch.1, A/D interrupt priority register (0x4026A)

Sets the priority level of the serial-interface interrupt.
The interrupt priority level can be set for each channel in the range of 0 to 7.
At initial reset, PSIOx becomes indeterminate.

ESERR0, ESRX0, ESTX0: Ch.0 interrupt enable (D0,D1,D2) / Serial I/F interrupt enable register (0x40276)
ESERR1, ESRX1, ESTX1: Ch.1 interrupt enable (D3,D4,D5) / Serial I/F interrupt enable register (0x40276)

Enable or disable interrupt generation to the CPU.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

The ESERRx, ESRXx, and ESTXx bits are interrupt enable bits corresponding to receive-error, receive-buffer full, and transmit-buffer empty interrupt factors, respectively, in each channel. The interrupts for which this bit is set to "1" are enabled, and the interrupts for which this bit is set to "0" are disabled.
At initial reset, all these bits are set to "0" (interrupts disabled).

FSERR0, FSRX0, FSTX0: Ch.0 interrupt factor flags (D0,D1,D2) / Serial I/F interrupt factor flag register (0x40286)
FSERR1, FSRX1, FSTX1: Ch.1 interrupt factor flags (D3,D4,D5) / Serial I/F interrupt factor flag register (0x40286)

Indicate the status of serial-interface interrupt generation.

When read

Read "1": An interrupt factor occurred
Read "0": No interrupt factor occurred

When written using the reset-only method (default)

Write "1": Flag is reset
Write "0": Invalid

When written using the read/write method

Write "1": Flag is set
Write "0": Flag is reset

The FSERRx, FSRXx, and FSTXx flags are interrupt factor flags corresponding to receive-error, receive-buffer full, and transmit-buffer empty interrupts, respectively, in each channel. The flag is set to "1" when each interrupt factor occurs.
A transmit-buffer empty interrupt factor occurs when transmit data is transferred from the transmit data register to the shift register.
A receive-buffer full interrupt factor occurs when receive data is transferred from the shift register to the receive data register.

A receive-error interrupt factor occurs when a parity, framing, or overrun error is detected during reception of data. At this time, if the following conditions are met, an interrupt to the CPU is generated:

1. The corresponding interrupt enable register bit is set to "1".
2. No other interrupt request of a higher priority has been generated.
3. The PSR's IE bit is set to "1" (interrupts enabled).
4. The set value of the corresponding interrupt priority register is higher than the CPU interrupt level (IL).

When using the receive-buffer full or transmit-buffer empty interrupt factor as an IDMA request, the fact that the above conditions are met does not necessarily mean that an interrupt request to the CPU has been output simultaneously when an interrupt factor occurs. An interrupt is generated under the above conditions upon completion of the data transfer by IDMA, provided that interrupts are enabled by settings on the IDMA side.

The interrupt factor flag is set to "1" whenever an interrupt factor occurs, regardless of the settings of the interrupt-enable and interrupt priority registers.

If the next interrupt is to be accepted following the occurrence of an interrupt, it is necessary that the interrupt factor flag be reset, and that the PSR be set up again (by setting the IE bit to "1" after setting the IL to a value lower than the level indicated by the interrupt priority register, or by executing the reti instruction).

The interrupt factor flag can only be reset by writing to it in the software. Note that if the PSR is set up again to accept interrupts generated (or if the reti instruction is executed) without resetting the interrupt factor flag, the same interrupt occurs again. Note also that the value to be written to reset the flag is "1" when the reset-only method (RSTONLY = "1") is used, and "0" when the read/write method (RSTONLY = "0") is used.

At initial reset, all of these flags become indeterminate, so be sure to reset them in the software.

RSRX0, RSTX0: Ch.0 IDMA request (D6, D7) /
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register (0x40292)

RSRX1, RSTX1: Ch.1 IDMA request (D0, D1) / Serial I/F Ch.1, A/D IDMA request register (0x40293)

Specifies whether to invoke IDMA when an interrupt factor occurs.

When using the set-only method (default)

Write "1": IDMA request
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA request
Write "0": Interrupt request
Read: Valid

The RSRXx and RSTXx bits are IDMA request bits corresponding to receive-buffer full and transmit-buffer empty interrupt factors, respectively. If the bit is set to "1", IDMA is invoked when an interrupt factor occurs, thus performing a programmed data transfer. If this bit is set to "0", normal interrupt processing is performed, without invoking IDMA.

For details on IDMA, refer to "IDMA (Intelligent DMA)".

At initial reset, these bits are set to "0" (interrupt request).

DESRX0, DESTX0: Ch.0 IDMA enable (D6, D7) /
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register (0x40296)
DESRX1, DESTX1: Ch.1 IDMA enable (D0, D1) / Serial I/F Ch.1, A/D IDMA enable register (0x40297)

Enables IDMA transfer by means of an interrupt factor.

When using the set-only method (default)

Write "1": IDMA enabled
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA enabled
Write "0": IDMA disabled
Read: Valid

The DESRXx and DESTXx bits are IDMA enable bits corresponding to receive-buffer full and transmit-buffer empty interrupt factors, respectively. If the bit is set to "1", the IDMA request by the interrupt factor is enabled. If the bit is set to "0", the IDMA request is disabled.
At initial reset, these bits are set to "0" (IDMA disabled).

Programming Notes

(1) Before setting various serial-interface parameters, make sure the transmit and receive operations are disabled (TXENx = RXENx = "0").

(2) When the serial interface is transmitting or receiving data, do not set TXENx or RXENx to "0", and do not execute the slp instruction.

(3) In clock-synchronized transfers, the mode of communication is half-duplex, in which the clock line is shared between the transmit and receive units. Therefore, RXENx and TXENx cannot be enabled simultaneously.

(4) After an initial reset, the interrupt factor flag becomes indeterminate. To prevent generation of an unwanted interrupt or IDMA request, reset this flag in the program.

(5) If a receive error occurs, the receive-error interrupt and receive-buffer full interrupt factors occur simultaneously. However, since the receive-error interrupt has priority over the receive-buffer full interrupt, the receive-error interrupt is processed first. Therefore, it is necessary to reset the receive-buffer full interrupt factor flag through the use of the receive-error interrupt processing routine

(6) To prevent the regeneration of interrupts due to the same factor following the occurrence of an interrupt, always be sure to reset the interrupt factor flag before setting the PSR again or executing the reti instruction.

(7) Follow the procedure described below to initialize the serial interface.

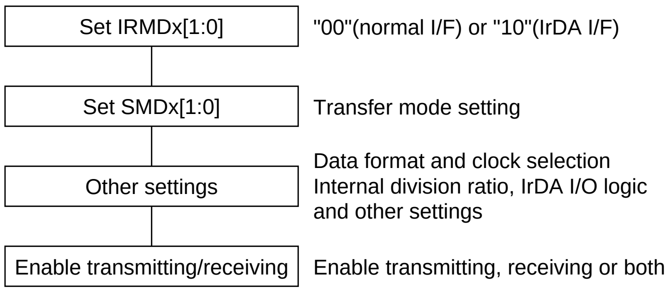


Figure 8.18 Serial Interface Initialize Procedure

(8) When transmitting data in the clock-synchronized master mode, transmit data is written to the transmit data register after the initial setting is performed following the flow in item (7). However, the clock generated by the 8-bit timer must be supplied to the serial interface (at least one underflow has had to have occurred in the 8-bit tier) before this writing. Otherwise, 0xFF will be transmitted prior to the written data.

(9) The maximum transfer rate of the serial interface is limited to 1 Mbps.

THIS PAGE IS BLANK.

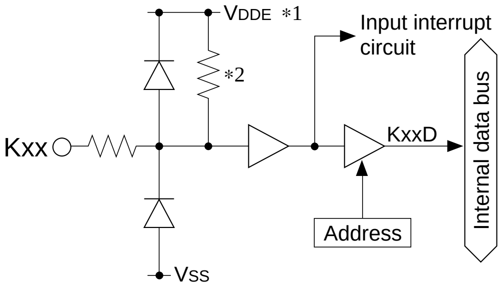
III-9 INPUT/OUTPUT PORTS

The Peripheral Block has a total of 42 input/output ports. Although each pin is used for input/output from/to the internal peripheral circuits, some pins can be used as general-purpose input/output ports unless they are used for the peripheral circuits.

Input Ports (K Ports)

Structure of Input Port

The Peripheral Block contains 13 bits of input ports (K50 to K54, K60 to K67).
Figure 9.1 shows the structure of a typical input port.



*1 AVDD for K50 and K60–K67
*2 Available only for K50–K54

Figure 9.1 Structure of Input Port

Each input-port pin is connected directly to the internal data bus via a three-state buffer. The state of the input signal when read at an input port is directly taken into the internal circuit as data.

Input-Port Pins

The input pins concurrently serve as the input pins for peripheral circuits, as shown in Table 9.1. Whether they are used as input ports or for peripheral circuits can be set bit-for-bit using a function select register. All pins not used for peripheral circuits can be used as general-purpose input ports that have an interrupt function.

Table 9.1 Input Pins

Pin name	I/O	Pull-up	Function	Function select bit
K50/#DMAREQ0	I	Available	Input port / High-speed DMA request 0	CFK50(D0)/K5 function select register(0x402C0)
K51/#DMAREQ1	I	Available	Input port / High-speed DMA request 1	CFK51(D1)/K5 function select register(0x402C0)
K52/#ADTRG	I	Available	Input port / AD converter trigger	CFK52(D2)/K5 function select register(0x402C0)
K53/#DMAREQ2	I	Available	Input port / High-speed DMA request 2	CFK53(D3)/K5 function select register(0x402C0)
K54/#DMAREQ3	I	Available	Input port / High-speed DMA request 3	CFK54(D4)/K5 function select register(0x402C0)
K60/AD0	I	–	Input port / AD converter input 0	CFK60(D0)/K6 function select register(0x402C3)
K61/AD1	I	–	Input port / AD converter input 1	CFK61(D1)/K6 function select register(0x402C3)
K62/AD2	I	–	Input port / AD converter input 2	CFK62(D2)/K6 function select register(0x402C3)
K63/AD3	I	–	Input port / AD converter input 3	CFK63(D3)/K6 function select register(0x402C3)
K64/AD4	I	–	Input port / AD converter input 4	CFK64(D4)/K6 function select register(0x402C3)
K65/AD5	I	–	Input port / AD converter input 5	CFK65(D5)/K6 function select register(0x402C3)
K66/AD6	I	–	Input port / AD converter input 6	CFK66(D6)/K6 function select register(0x402C3)
K67/AD7	I	–	Input port / AD converter input 7	CFK67(D7)/K6 function select register(0x402C3)

At cold start, all pins are set for input ports Kxx (function select register CFKxx = "0"). When these pins are used for the internal peripheral circuits, write "1" to CFKxx. For details on pin functions in this case, refer to the description of each peripheral circuit in this manual.
At hot start, the pins retain their state from prior to the reset.
When the ports set for A/D converter input are read, the value obtained is always "0".

Notes on Use

The input buffers of the K50 and K60 to K67 ports use AV_{DD} (power voltage for A/D converter) as their power source. Furthermore, the K50 pull-up resistor is connected to AV_{DD}. Therefore, the following precautions must be taken.

1) When using K50 and K60–K67 as general-purpose input ports, the voltage input to the port must be high level = AV_{DD} and low level = V_{SS}.

2) When using V_{DDE} as high level similar to other ports, V_{DDE} must be the same voltage level as AV_{DD}. If the input V_{DDE} level is lower than the AV_{DD} level, current flows in the input buffer, or if the input V_{DDE} level is higher than the AV_{DD} level, current flows from the V_{DDE} power supply to the AV_{DD} power supply.

3) To fix the input level externally when the port is not used, the input pin should be connected to V_{SS} or AV_{DD}.

I/O Memory of Input Ports

Table 9.2 shows the control bits of the input ports.

Table 9.2 Control Bits of Input Ports

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
K5 function select register	**00402C0 (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**CFK54**	K54 function selection	1	#DMAREQ3	0	K54	0	R/W	
		D3	**CFK53**	K53 function selection	1	#DMAREQ2	0	K53	0	R/W	
		D2	**CFK52**	K52 function selection	1	#ADTRG	0	K52	0	R/W	
		D1	**CFK51**	K51 function selection	1	#DMAREQ1	0	K51	0	R/W	
		D0	**CFK50**	K50 function selection	1	#DMAREQ0	0	K50	0	R/W	
K5 input port data register	**00402C1 (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**K54D**	K54 input port data	1	High	0	Low	–	R	
		D3	**K53D**	K53 input port data					–	R	
		D2	**K52D**	K52 input port data					–	R	
		D1	**K51D**	K51 input port data					–	R	
		D0	**K50D**	K50 input port data					–	R	
K6 function select register	**00402C3 (B)**	D7	**CFK67**	K67 function selection	1	AD7	0	K67	0	R/W	
		D6	**CFK66**	K66 function selection	1	AD6	0	K66	0	R/W	
		D5	**CFK65**	K65 function selection	1	AD5	0	K65	0	R/W	
		D4	**CFK64**	K64 function selection	1	AD4	0	K64	0	R/W	
		D3	**CFK63**	K63 function selection	1	AD3	0	K63	0	R/W	
		D2	**CFK62**	K62 function selection	1	AD2	0	K62	0	R/W	
		D1	**CFK61**	K61 function selection	1	AD1	0	K61	0	R/W	
		D0	**CFK60**	K60 function selection	1	AD0	0	K60	0	R/W	
K6 input port data register	**00402C4 (B)**	D7	**K67D**	K67 input port data	1	High	0	Low	–	R	
		D6	**K66D**	K66 input port data					–	R	
		D5	**K65D**	K65 input port data					–	R	
		D4	**K64D**	K64 input port data					–	R	
		D3	**K63D**	K63 input port data					–	R	
		D2	**K62D**	K62 input port data					–	R	
		D1	**K61D**	K61 input port data					–	R	
		D0	**K60D**	K60 input port data					–	R	

CFK54–CFK50: K5[4:0] function selection (D[4:0]) / K5 function select register (0x402C0)
CFK67–CFK60: K6[7:0] function selection (D[7:0]) / K6 function select register (0x402C3)

Selects the function of each input-port pin.

Write "1": Used for peripheral circuit
Write "0": Input port pin
Read: Invalid

When a bit of the CFK register is set to "1", the corresponding pin is set for use with the peripheral circuit (see Table 9.1). The pins for which register bits are set to "0" can be used as general-purpose input ports.
At cold start, CFK is set to "0" (input port). At hot start, CFK retains its state from prior to the initial reset.

K54D–K50D: K5[4:0] input port data (D[4:0]) / K5 input port data register (0x402C1)
K67D–K60D: K6[7:0] input port data (D[7:0]) / K6 input port data register (0x402C4)

The input data on each input port pin can be read from this register.

Read "1": High level
Read "0": Low level
Write: Invalid

The pin voltage of each input port can be read out "1" directly when the voltage is high (VDD) or "0" when the voltage is low (VSS) respectively.
Since this register is a read-only register, writing to the register is ignored.
When the ports set for A/D converter input are read, the value obtained is always "0".

I/O Ports (P Ports)

Structure of I/O Port

The Peripheral Block contains 29 bits of I/O ports (P00 to P07, P10 to P16, P20 to P27, P30 to P35) that can be directed for input or output through the use of a program.
Figure 9.2 shows the structure of a typical I/O port.

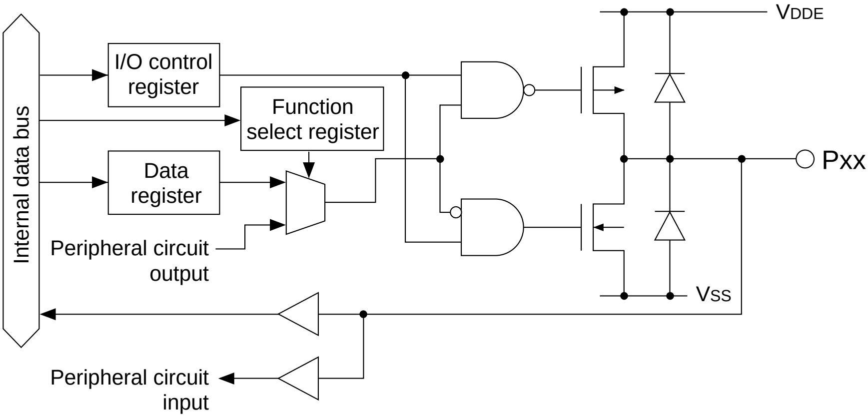


Figure 9.2 Structure of I/O Port

I/O Port Pins

The I/O ports concurrently serve as the input/output pins for peripheral circuits, as shown in Table 9.3. Whether they are used as I/O ports or for peripheral circuits can be set bit-for-bit using a function select register. All pins not used for peripheral circuits can be used as general-purpose I/O ports.

Table 9.3 I/O Pins

Pin name	I/O	Pull-up	Function	Function select bit
P00/SIN0	I/O	–	I/O port / Serial IF Ch.0 data input	CFP00(D0)/P0 function select register(0x402D0)
P01/SOUT0	I/O	–	I/O port / Serial IF Ch.0 data output	CFP01(D1)/P0 function select register(0x402D0)
P02/#SCLK0	I/O	–	I/O port / Serial IF Ch.0 clock input/output	CFP02(D2)/P0 function select register(0x402D0)
P03/#SRDY0	I/O	–	I/O port / Serial IF Ch.0 ready input/output	CFP03(D3)/P0 function select register(0x402D0)
P04/SIN1/ #DMAACK2	I/O	–	I/O port / Serial IF Ch.1 data input / #DMAACK2 output (Ex)	CFP04(D4)/P0 function select register(0x402D0) CFEX4(D4)/Port function extension register(0x402DF)
P05/SOUT1/ #DMAEND2	I/O	–	I/O port / Serial IF Ch.1 data output / #DMAEND2 output (Ex)	CFP05(D5)/P0 function select register(0x402D0) CFEX5(D5)/Port function extension register(0x402DF)
P06/#SCLK1/ #DMAACK3	I/O	–	I/O port / Serial IF Ch.1 clock input/output / #DMAACK3 output (Ex)	CFP06(D6)/P0 function select register(0x402D0) CFEX6(D6)/Port function extension register(0x402DF)
P07/#SRDY1/ #DMAEND3	I/O	–	I/O port / Serial IF Ch.1 ready input/output / #DMAEND3 output (Ex)	CFP07(D7)/P0 function select register(0x402D0) CFEX7(D7)/Port function extension register(0x402DF)
P10/EXCL0/ T8UF0/DST0 *	I/O	–	I/O port / 16-bit timer 0 event counter input (I) / 8-bit timer 0 output (O) / DST0 output (Ex)	CFP10(D0)/P1 function select register(0x402D4) CFEX1(D1)/Port function extension register(0x402DF)
P11/EXCL1/ T8UF1/DST1 *	I/O	–	I/O port / 16-bit timer 1 event counter input (I) / 8-bit timer 1 output (O) / DST1 output (Ex)	CFP11(D1)/P1 function select register(0x402D4) CFEX1(D1)/Port function extension register(0x402DF)
P12/EXCL2/ T8UF2/DST2 *	I/O	–	I/O port / 16-bit timer 2 event counter input (I) / 8-bit timer 2 output (O) / DST2 output (Ex)	CFP12(D2)/P1 function select register(0x402D4) CFEX0(D0)/Port function extension register(0x402DF)
P13/EXCL3/ T8UF3/DPCO *	I/O	–	I/O port / 16-bit timer 3 event counter input (I) / 8-bit timer 3 output (O) / DPCO output (Ex)	CFP13(D3)/P1 function select register(0x402D4) CFEX1(D1)/Port function extension register(0x402DF)
P14/FOSC1/ DCLK *	I/O	–	I/O port / Low-speed (OSC1) clock output / DCLK output (Ex)	CFP14(D4)/P1 function select register(0x402D4) CFEX0(D0)/Port function extension register(0x402DF)
P15/EXCL4/ #DMAEND0	I/O	–	I/O port / 16-bit timer 4 event counter input (I) / #DMAEND0 output (O)	CFP15(D5)/P1 function select register(0x402D4)
P16/EXCL5/ #DMAEND1	I/O	–	I/O port / 16-bit timer 5 event counter input (I) / #DMAEND1 output (O)	CFP16(D6)/P1 function select register(0x402D4)

(I): Input mode, (O): Output mode, (Ex): Extended function

*: A 3-V system I/O voltage can only be used for the P10–P14 pins.

Pin name	I/O	Pull-up	Function	Function select bit
P20/#DRD	I/O	–	I/O port / #DRD output	CFP20(D0)/P2 function select register(0x402D8)
P21/#DWE/#GAAS	I/O	–	I/O port / #DWE output / GA address strobe output (Ex)	CFP21(D1)/P2 function select register(0x402D8) CFEX2(D2)/Port function extension register(0x402DF)
P22/TM0	I/O	–	I/O port / 16-bit timer 0 output	CFP22(D2)/P2 function select register(0x402D8)
P23/TM1	I/O	–	I/O port / 16-bit timer 1 output	CFP23(D3)/P2 function select register(0x402D8)
P24/TM2	I/O	–	I/O port / 16-bit timer 2 output	CFP24(D4)/P2 function select register(0x402D8)
P25/TM3	I/O	–	I/O port / 16-bit timer 3 output	CFP25(D5)/P2 function select register(0x402D8)
P26/TM4	I/O	–	I/O port / 16-bit timer 4 output	CFP26(D6)/P2 function select register(0x402D8)
P27/TM5	I/O	–	I/O port / 16-bit timer 5 output	CFP27(D7)/P2 function select register(0x402D8)
P30/#WAIT/#CE4&5	I/O	–	I/O port / #WAIT input (I) / #CE4&5 output (O)	CFP30(D0)/P3 function select register(0x402DC)
P31/#BUSGET/#GARD	I/O	–	I/O port / #BUSGET output / GA read signal output (Ex)	CFP31(D1)/P3 function select register(0x402DC) CFEX3(D3)/Port function extension register(0x402DF)
P32/#DMAACK0	I/O	–	I/O port / #DMAACK0 output	CFP32(D2)/P3 function select register(0x402DC)
P33/#DMAACK1	I/O	–	I/O port / #DMAACK1 output	CFP33(D3)/P3 function select register(0x402DC)
P34/#BUSREQ/#CE6	I/O	–	I/O port / #BUSREQ input (I) / #CE6 output (O)	CFP34(D4)/P3 function select register(0x402DC)
P35/#BUSACK	I/O	–	I/O port / #BUSACK output	CFP35(D5)/P3 function select register(0x402DC)

(I): Input mode, (O): Output mode, (Ex): Extended function

At cold start, all pins are set for I/O ports Pxx (function select register CFPxx = "0"). When these pins are used for the internal peripheral circuits, write "1" to CFPxx. For details on pin functions in this case, refer to the description of each peripheral circuit in this manual.
At hot start, the pins retain their state from prior to the reset.
In addition to being an I/O port, the P10–P13, P15–P16, P30 and P34 pins are shared with two types (three types for P10–P13) of peripheral circuits. The type of peripheral circuit for which these pins are used is determined by the direction (input or output) in which the pin is set using an I/O control register, as will be described later.
The P04–P07, P10–P14, P21 and P31 ports have extended functions indicated with (Ex) in the table. They can be selected by writing "1" to CFEXx / Port function extension register (0x402DF).
The setting of CFEXx has priority over the CFPxx.
At cold start, CFEX1 and CFEX0 are set to "1", so the P10–P14 pins are set for debug signal outputs.

I/O Control Register and I/O Modes

The I/O ports are directed for input or output modes by writing data to an I/O control register corresponding to each port bit.
P07–P00 I/O control: IOC0[7:0] (D[7:0]) / P0 I/O control register (0x402D2)
P16–P10 I/O control: IOC1[6:0] (D[6:0]) / P1 I/O control register (0x402D6)
P27–P20 I/O control: IOC2[7:0] (D[7:0]) / P2 I/O control register (0x402DA)
P35–P30 I/O control: IOC3[5:0] (D[5:0]) / P3 I/O control register (0x402DE)

To set an I/O port for input, write "0" to the I/O control bit. I/O ports set for input mode are placed in the high-impedance state, and thus function as input ports.
In the input mode, the state of the input pin is read directly, so the data is "1" when the pin state is high (VDD level) or "0" when the pin state is low (VSS level).
Even in the input mode, data can be written to the data register without affecting the pin state.

To set an I/O port for output, write "1" to the I/O control bit. I/O port set for output function as output ports. When the port output data is "1", the port outputs a high level (VDD level); when the data is "0", the port outputs a low level (VSS level).

At cold start, the I/O control register is set to "0" (input mode).
At hot start, the pins retain their state from prior to the reset.

Note: If pins P10–P13, P15–P16, P30 and P34 are set for use with peripheral circuits, their pin functions vary depending on the input/output direction control by the IOC1x register.

I/O Memory of I/O Ports

Table 9.4 shows the control bits of the I/O ports.

Table 9.4 Control Bits of I/O Ports

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
P0 function select register	**00402D0 (B)**	D7	**CFP07**	P07 function selection	1	#SRDY1	0	P07	0	R/W	Extended functions (0x402DF)
		D6	**CFP06**	P06 function selection	1	#SCLK1	0	P06	0	R/W	
		D5	**CFP05**	P05 function selection	1	SOUT1	0	P05	0	R/W	
		D4	**CFP04**	P04 function selection	1	SIN1	0	P04	0	R/W	
		D3	**CFP03**	P03 function selection	1	#SRDY0	0	P03	0	R/W	
		D2	**CFP02**	P02 function selection	1	#SCLK0	0	P02	0	R/W	
		D1	**CFP01**	P01 function selection	1	SOUT0	0	P01	0	R/W	
		D0	**CFP00**	P00 function selection	1	SIN0	0	P00	0	R/W	
P0 I/O port data register	**00402D1 (B)**	D7	**P07D**	P07 I/O port data	1	High	0	Low	0	R/W	
		D6	**P06D**	P06 I/O port data					0	R/W	
		D5	**P05D**	P05 I/O port data					0	R/W	
		D4	**P04D**	P04 I/O port data					0	R/W	
		D3	**P03D**	P03 I/O port data					0	R/W	
		D2	**P02D**	P02 I/O port data					0	R/W	
		D1	**P01D**	P01 I/O port data					0	R/W	
		D0	**P00D**	P00 I/O port data					0	R/W	
P0 I/O control register	**00402D2 (B)**	D7	**IOC07**	P07 I/O control	1	Output	0	Input	0	R/W	
		D6	**IOC06**	P06 I/O control					0	R/W	
		D5	**IOC05**	P05 I/O control					0	R/W	
		D4	**IOC04**	P04 I/O control					0	R/W	
		D3	**IOC03**	P03 I/O control					0	R/W	
		D2	**IOC02**	P02 I/O control					0	R/W	
		D1	**IOC01**	P01 I/O control					0	R/W	
		D0	**IOC00**	P00 I/O control					0	R/W	
P1 function select register	**00402D4 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**CFP16**	P16 function selection	1	EXCL5 #DMAEND1	0	P16	0	R/W	
		D5	**CFP15**	P15 function selection	1	EXCL4 #DMAEND0	0	P15	0	R/W	
		D4	**CFP14**	P14 function selection	1	FOSC1	0	P14	0	R/W	Extended functions (0x402DF)
		D3	**CFP13**	P13 function selection	1	EXCL3 T8UF3	0	P13	0	R/W	
		D2	**CFP12**	P12 function selection	1	EXCL2 T8UF2	0	P12	0	R/W	
		D1	**CFP11**	P11 function selection	1	EXCL1 T8UF1	0	P11	0	R/W	
		D0	**CFP10**	P10 function selection	1	EXCL0 T8UF0	0	P10	0	R/W	
P1 I/O port data register	**00402D5 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**P16D**	P16 I/O port data	1	High	0	Low	0	R/W	
		D5	**P15D**	P15 I/O port data					0	R/W	
		D4	**P14D**	P14 I/O port data					0	R/W	
		D3	**P13D**	P13 I/O port data					0	R/W	
		D2	**P12D**	P12 I/O port data					0	R/W	
		D1	**P11D**	P11 I/O port data					0	R/W	
		D0	**P10D**	P10 I/O port data					0	R/W	
P1 I/O control register	**00402D6 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**IOC16**	P16 I/O control	1	Output	0	Input	0	R/W	
		D5	**IOC15**	P15 I/O control					0	R/W	
		D4	**IOC14**	P14 I/O control					0	R/W	
		D3	**IOC13**	P13 I/O control					0	R/W	
		D2	**IOC12**	P12 I/O control					0	R/W	
		D1	**IOC11**	P11 I/O control					0	R/W	
		D0	**IOC10**	P10 I/O control					0	R/W	
P2 function select register	**00402D8 (B)**	D7	**CFP27**	P27 function selection	1	TM5	0	P27	0	R/W	
		D6	**CFP26**	P26 function selection	1	TM4	0	P26	0	R/W	
		D5	**CFP25**	P25 function selection	1	TM3	0	P25	0	R/W	
		D4	**CFP24**	P24 function selection	1	TM2	0	P24	0	R/W	
		D3	**CFP23**	P23 function selection	1	TM1	0	P23	0	R/W	
		D2	**CFP22**	P22 function selection	1	TM0	0	P22	0	R/W	
		D1	**CFP21**	P21 function selection	1	#DWE	0	P21	0	R/W	Ext. func.(0x402DF)
		D0	**CFP20**	P20 function selection	1	#DRD	0	P20	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
P2 I/O port data register	**00402D9 (B)**	D7	**P27D**	P27 I/O port data	1	High	0	Low	0	R/W	
		D6	**P26D**	P26 I/O port data					0	R/W	
		D5	**P25D**	P25 I/O port data					0	R/W	
		D4	**P24D**	P24 I/O port data					0	R/W	
		D3	**P23D**	P23 I/O port data					0	R/W	
		D2	**P22D**	P22 I/O port data					0	R/W	
		D1	**P21D**	P21 I/O port data					0	R/W	
		D0	**P20D**	P20 I/O port data					0	R/W	
P2 I/O control register	**00402DA (B)**	D7	**IOC27**	P27 I/O control	1	Output	0	Input	0	R/W	
		D6	**IOC26**	P26 I/O control					0	R/W	
		D5	**IOC25**	P25 I/O control					0	R/W	
		D4	**IOC24**	P24 I/O control					0	R/W	
		D3	**IOC23**	P23 I/O control					0	R/W	
		D2	**IOC22**	P22 I/O control					0	R/W	
		D1	**IOC21**	P21 I/O control					0	R/W	
		D0	**IOC20**	P20 I/O control					0	R/W	
P3 function select register	**00402DC (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**CFP35**	P35 function selection	1	#BUSACK	0	P35	0	R/W	
		D4	**CFP34**	P34 function selection	1	#BUSREQ #CE6	0	P34	0	R/W	
		D3	**CFP33**	P33 function selection	1	#DMAACK1	0	P33	0	R/W	
		D2	**CFP32**	P32 function selection	1	#DMAACK0	0	P32	0	R/W	
		D1	**CFP31**	P31 function selection	1	#BUSGET	0	P31	0	R/W	Ext. func.(0x402DF)
		D0	**CFP30**	P30 function selection	1	#WAIT #CE4/#CE5	0	P30	0	R/W	
P3 I/O port data register	**00402DD (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**P35D**	P35 I/O port data	1	High	0	Low	0	R/W	
		D4	**P34D**	P34 I/O port data					0	R/W	
		D3	**P33D**	P33 I/O port data					0	R/W	
		D2	**P32D**	P32 I/O port data					0	R/W	
		D1	**P31D**	P31 I/O port data					0	R/W	
		D0	**P30D**	P30 I/O port data					0	R/W	
P3 I/O control register	**00402DE (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**IOC35**	P35 I/O control	1	Output	0	Input	0	R/W	
		D4	**IOC34**	P34 I/O control					0	R/W	
		D3	**IOC33**	P33 I/O control					0	R/W	
		D2	**IOC32**	P32 I/O control					0	R/W	
		D1	**IOC31**	P31 I/O control					0	R/W	
		D0	**IOC30**	P30 I/O control					0	R/W	
Port function extension register	**00402DF (B)**	D7	**CFEX7**	P07 port extended function	1	#DMAEND3	0	P07, etc.	0	R/W	
		D6	**CFEX6**	P06 port extended function	1	#DMAACK3	0	P06, etc.	0	R/W	
		D5	**CFEX5**	P05 port extended function	1	#DMAEND2	0	P05, etc.	0	R/W	
		D4	**CFEX4**	P04 port extended function	1	#DMAACK2	0	P04, etc.	0	R/W	
		D3	**CFEX3**	P31 port extended function	1	#GARD	0	P31, etc.	0	R/W	
		D2	**CFEX2**	P21 port extended function	1	#GAAS	0	P21, etc.	0	R/W	
		D1	**CFEX1**	P10, P11, P13 port extended function	1	DST0 DST1 DPC0	0	P10, etc. P11, etc. P13, etc.	1	R/W	
		D0	**CFEX0**	P12, P14 port extended function	1	DST2 DCLK	0	P12, etc. P14, etc.	1	R/W	

CFP07–CFP00: P0[7:0] function selection (D[7:0]) / P0 function select register (0x402D0)
CFP16–CFP10: P1[6:0] function selection (D[6:0]) / P1 function select register (0x402D4)
CFP27–CFP20: P2[7:0] function selection (D[7:0]) / P2 function select register (0x402D8)
CFP35–CFP30: P3[5:0] function selection (D[5:0]) / P3 function select register (0x402DC)

Selects the function of each I/O port pin.

Write "1": Used for peripheral circuit
Write "0": I/O port pin
Read: Valid

When a bit of the CFP register is set to "1", the corresponding pin is set for use with peripheral circuits (see Table 9.3). The pins for which register bits are set to "0" can be used as general-purpose I/O ports.
At cold start, CFP is set to "0" (I/O port). At hot start, CFP retains its state from prior to the initial reset.

P07D–P00D: P0[7:0] I/O port data (D[7:0]) / P0 I/O port data register (0x402D1)
P16D–P10D: P1[6:0] I/O port data (D[6:0]) / P1 I/O port data register (0x402D5)
P27D–P20D: P2[7:0] I/O port data (D[7:0]) / P2 I/O port data register (0x402D9)
P35D–P30D: P3[5:0] I/O port data (D[5:0]) / P3 I/O port data register (0x402DD)

This register reads data from I/O-port pins or sets output data.

When writing data

Write "1": High level
Write "0": Low level

When an I/O port is set for output, the data written to it is directly output to the I/O port pin. If the data written to the port is "1", the port pin is set high (VDD level); if the data is "0", the port pin is set low (VSS level).
Even in the input mode, data can be written to the port data register.

When reading data

Read "1": High level
Read "0": Low level

The voltage level on the port pin is read out regardless of whether an I/O port is set for input or output mode. If the pin voltage is high (VDD level), "1" is read out as input data; if the pin voltage is low (VSS level), "0" is read out as input data.
At cold start, all data bits are set to "0". At hot start, they retain their state from prior to the initial reset.

IOC07–IOC00: P0[7:0] port I/O control (D[7:0]) / P0 port I/O control register (0x402D2)
IOC16–IOC10: P1[6:0] port I/O control (D[6:0]) / P1 port I/O control register (0x402D6)
IOC27–IOC20: P2[7:0] port I/O control (D[7:0]) / P2 port I/O control register (0x402DA)
IOC35–IOC30: P3[5:0] port I/O control (D[5:0]) / P3 port I/O control register (0x402DE)

Directs an I/O port for input or output.

Write "1": Output mode
Write "0": Input mode
Read: Valid

This I/O control register corresponds bit-for-bit to each I/O port. When an IOC bit is set to "1", the corresponding I/O port is directed for output; if it is set to "0", the I/O port is directed for input.
At cold start, all IOC bits are set to "0" (input). At hot start, IOC retains its state from prior to the initial reset.
If pins P10–P13, P15–P16, P30 and P34 are set for use with peripheral circuits, their pin functions vary depending on the input/output direction control by the IOC1x register.

CFEX0: P12, P14 function extension (D0) / Port function extension register (0x402DF)
CFEX1: P10, P11, P13 function extension (D1) / Port function extension register (0x402DF)
CFEX2: P21 function extension (D2) / Port function extension register (0x402DF)
CFEX3: P31 function extension (D3) / Port function extension register (0x402DF)
CFEX4: P04 function extension (D4) / Port function extension register (0x402DF)
CFEX5: P05 function extension (D5) / Port function extension register (0x402DF)
CFEX6: P06 function extension (D6) / Port function extension register (0x402DF)
CFEX7: P07 function extension (D7) / Port function extension register (0x402DF)

Sets whether the function of an I/O-port pin is to be extended.

Write "1": Function-extended pin
Write "0": I/O-port/peripheral-circuit pin
Read: Valid

When CFEXx is set to "1", the corresponding pin is set to the extended function input/output pin. When CFEXx = "0", the corresponding CFP bit becomes effective.
At cold start, CFEX0 and CFEX1 are set to "1" (function-extended pin) and other bits are set to "0" (I/O-port/peripheral-circuit pin). At hot start, CFEX retains its state from prior to the initial reset.

Input Interrupt

The input ports and the I/O ports support eight system of port input interrupts and two systems of key input interrupts.

Port Input Interrupt

The port input interrupt circuit has eight interrupt systems (FPT7–FPT0) and a port can be selected for generating each interrupt factor.
The interrupt condition can also be selected from between input signal edge and input signal level.
Figure 9.3 shows the configuration of the port input interrupt circuit.

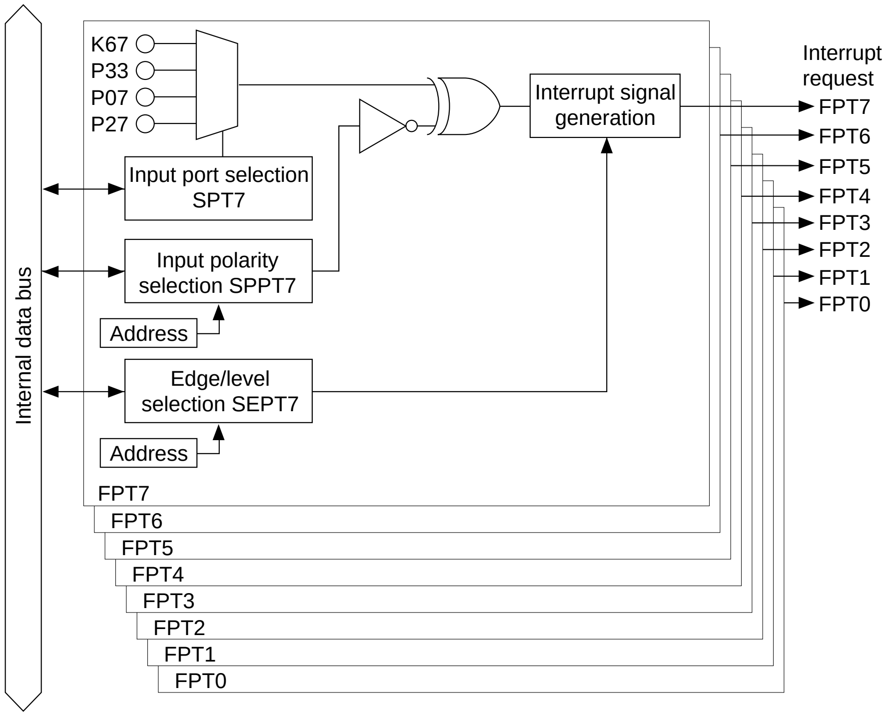


Figure 9.3 Configuration of Port Input Interrupt Circuit

Selecting input pins

The interrupt factors allows selection of an input pin from the four predefined pins independently.
Table 9.5 shows the control bits and the selectable pins for each factor.

Table 9.5 Selecting Pins for Port Input Interrupts

Interrupt factor	Control bit	SPT settings			
		11	10	01	00
FPT7	SPT7[1:0] (D[7:6])/Port input interrupt select register 2 (0x402C7)	P27	P07	P33	K67
FPT6	SPT6[1:0] (D[5:4])/Port input interrupt select register 2 (0x402C7)	P26	P06	P32	K66
FPT5	SPT5[1:0] (D[3:2])/Port input interrupt select register 2 (0x402C7)	P25	P05	P31	K65
FPT4	SPT4[1:0] (D[1:0])/Port input interrupt select register 2 (0x402C7)	P24	P04	K54	K64
FPT3	SPT3[1:0] (D[7:6])/Port input interrupt select register 1 (0x402C6)	P23	P03	K53	K63
FPT2	SPT2[1:0] (D[5:4])/Port input interrupt select register 1 (0x402C6)	P22	P02	K52	K62
FPT1	SPT1[1:0] (D[3:2])/Port input interrupt select register 1 (0x402C6)	P21	P01	K51	K61
FPT0	SPT0[1:0] (D[1:0])/Port input interrupt select register 1 (0x402C6)	P20	P00	K50	K60

Conditions for port input-interrupt generation

Each port input interrupt can be generated by the edge or level of the input signal. The SEPTx bit of the edge/level select register (0x402C9) is used for this selection. When SEPTx is set to "1", the FPTx interrupt will be generated at the signal edge. When SEPTx is set to "0", the FPTx interrupt will be generated by the input signal level.

Furthermore, the signal polarity can be selected using the SPPTx bit of the input porarity select register (0x402C8).

With these registers, the port input interrupt condition is decided as shown in Table 9.6.

Table 9.6 Port Input Interrupt Condition

SEPTx	SPPTx	FPTx interrupt condition
1	1	Rising edge
1	0	Falling edge
0	1	High level
0	0	Low level

When the input signal goes to the selected status, the interrupt factor flag FP is set to "1" and, if other interrupt conditions set by the interrupt controller are met, an interrupt is generated.

Key Input Interrupt

The key input interrupt circuit has two interrupt systems (FPK1 and FPK0) and a port group can be selected for generating each interrupt factor.

The interrupt condition can also be set by software.

Figure 9.4 shows the configuration of the port input interrupt circuit.

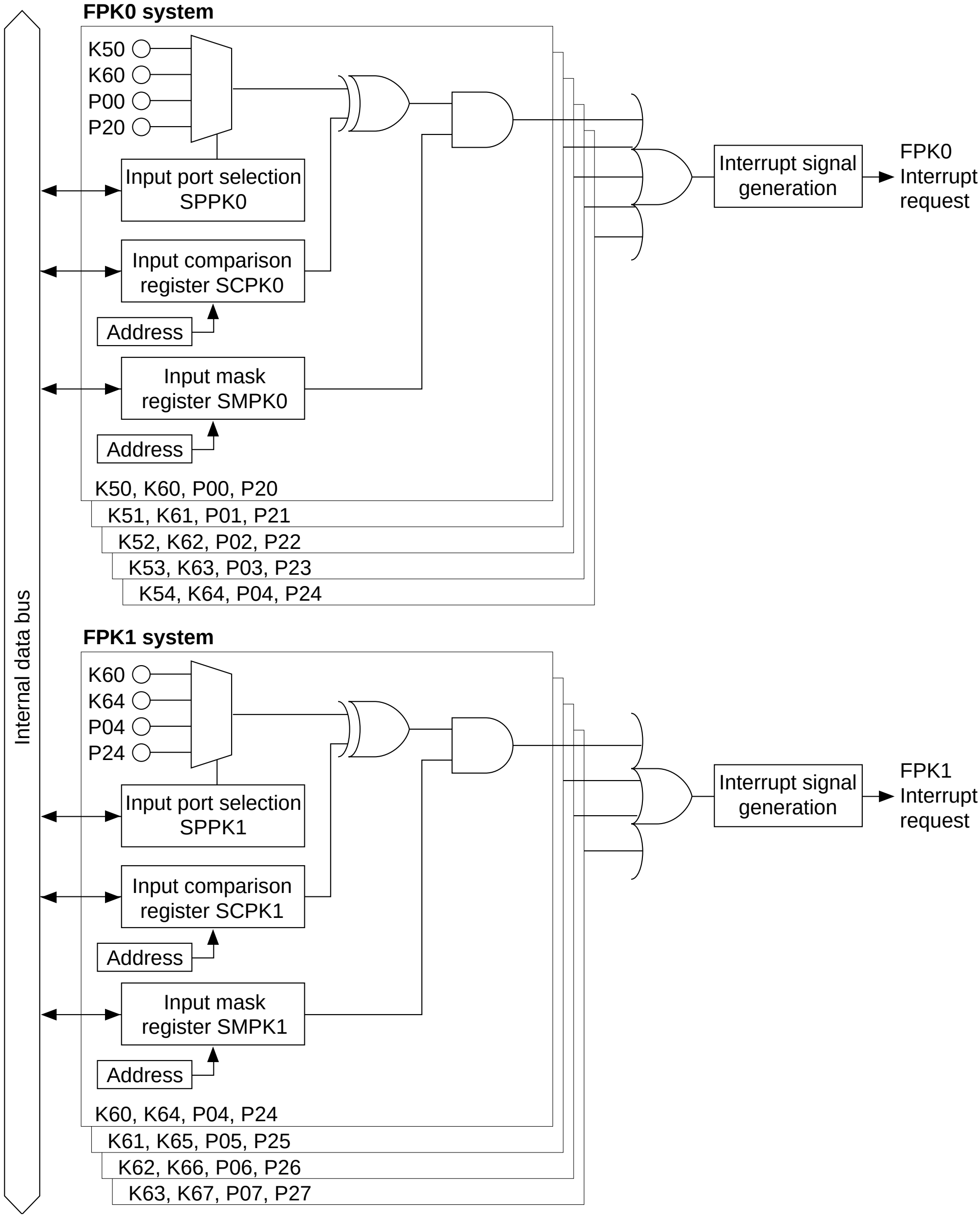


Figure 9.4 Configuration of Key Input Interrupt Circuit

Selecting input pins

For the FPK1 interrupt system, a four-bit input pin group can be selected from the four predefined groups. For the FPK0 system, a five-bit input pin group can be selected.

Table 9.7 shows the control bits and the selectable groups for each factor.

Table 9.7 Selecting Pins for Key Input Interrupts

Interrupt factor	Control bit	SPPK settings			
		11	10	01	00
FPK1	SPPK1[1:0] (D[3:2])/Key input interrupt select register (0x402CA)	P2[7:4]	P0[7:4]	K6[7:4]	K6[3:0]
FPK0	SPPK0[1:0] (D[1:0])/Key input interrupt select register (0x402CA)	P2[4:0]	P0[4:0]	K6[4:0]	K5[4:0]

Conditions for key input-interrupt generation

The key input interrupt circuit has two input mask registers (SMPK0[4:0] for FPK0 and SMPK1[3:0] for FPK1) and two input comparison registers (SCPK0[4:0] for FPK0 and SCPK1[3:0] for FPK1) to set input-interrupt conditions.

The input mask register SMPK is used to mask the input pin that is not used for an interrupt. This register masks each input pin, whereas the interrupt enable register of the interrupt controller masks the interrupt factor for each interrupt group.

The input comparison register SCPK is used to select whether an interrupt for each input port is to be generated at the rising or falling edge of the input.

A change in state occurs so that the input pin enabled for interrupt by the interrupt mask register SMPK and the content of the input comparison register SCPK become unmatched after being matched, the interrupt factor flag FK is set to "1" and, if other interrupt conditions are met, an interrupt is generated.

Figure 9.5 shows cases in which a FPK0 interrupt is generated. Here, it is assumed that the K5[4:0] pins are selected for the input-pin group and the control register of the interrupt controller is set so as to enable generation of a FPK0 interrupt.

Intput mask register SMPK0

SMPK04	SMPK03	SMPK02	SMPK01	SMPK00
1	1	1	1	0

Input comparison register SCPK0

SCPK04	SCPK03	SCPK02	SCPK01	SCPK00
1	1	0	1	0

With the settings shown above, FPK0 interrupt is generated under the condition shown below.

Input port K5

(1)

K54	K53	K52	K51	K50
1	1	0	1	0

(Initial value)

↓

(2)

K54	K53	K52	K51	K50
1	1	0	1	1

↓

(3)

K54	K53	K52	K51	K50
1	0	0	1	0

→ Interrupt generation

Because interrupt has been disabled for K50, interrupt will be generated when non-conformity occurs between the contents of the four bits K51–K54 and the four bits input comparison register SCPK0[4:1].

↓

(4)

K54	K53	K52	K51	K50
1	0	1	1	0

Figure 9.5 FPK0 Interrupt Generation Example (when K5[4:0] is selected by SPPK[1:0])

Since K50 is masked from interrupt by SMPK00, no interrupt occurs at that point (2) above.
Next, because K53 becomes "0" at (3), an interrupt is generated due to the lack of a match between the data of the input pin K5[4:1] that is enabled for interrupt and that of the input comparison register SCPK0[4:1].
Since only a change in states in which the input data and the content of the input comparison register SCPK become unmatched after being matched constitutes an interrupt generation condition as described above, no interrupt is generated when a change in states from one unmatched state to another, as in (4), occurs.
Consequently, if another interrupt is to be generated again following the occurrence of an interrupt, the state of the input pin must be temporarily restored to the same content as that of the input comparison register SCPK, or the input comparison register SCPK must be set again. Note that the input pins masked from interrupt by the SMPK register do not affect interrupt generation conditions.

An interrupt is generated for FPK1 in the same way as described above.

Control Registers of the Interrupt Controller

Table 9.8 shows the control registers of the interrupt controller that are provided for each input-interrupt system.

Table 9.8 Control Registers of Interrupt Controller

System	Interrupt factor flag	Interrupt enable register	Interrupt priority register
FPT7	FP7(D5/0x40287)	EP7(D5/0x40277)	PP7L[2:0](D[6:4]/0x4026D)
FPT6	FP6(D4/0x40287)	EP6(D4/0x40277)	PP6L[2:0](D[2:0]/0x4026D)
FPT5	FP5(D3/0x40287)	EP5(D3/0x40277)	PP5L[2:0](D[6:4]/0x4026C)
FPT4	FP4(D2/0x40287)	EP4(D2/0x40277)	PP4L[2:0](D[2:0]/0x4026C)
FPT3	FP3(D3/0x40280)	EP3(D3/0x40270)	PP3L[2:0](D[6:4]/0x40261)
FPT2	FP2(D2/0x40280)	EP2(D2/0x40270)	PP2L[2:0](D[2:0]/0x40261)
FPT1	FP1(D1/0x40280)	EP1(D1/0x40270)	PP1L[2:0](D[6:4]/0x40260)
FPT0	FP0(D0/0x40280)	EP0(D0/0x40270)	PP0L[2:0](D[2:0]/0x40260)
FPK1	FK1(D5/0x40280)	EK1(D5/0x40270)	PK1L[2:0](D[6:4]/0x40262)
FPK0	FK0(D4/0x40280)	EK0(D4/0x40270)	PK0L[2:0](D[2:0]/0x40262)

When the interrupt generation condition described above is met, the corresponding interrupt factor flag is set to "1". If the interrupt enable register bit for that interrupt factor has been set to "1", an interrupt request is generated.
Interrupts due to an interrupt factor can be disabled by leaving the interrupt enable register bit for that factor set to "0". The interrupt factor flag is set to "1" whenever interrupt generation conditions are met, regardless of the setting of the interrupt enable register.
The interrupt priority register sets the interrupt priority level (0 to 7) for each interrupt system. An interrupt request to the CPU is accepted only when no other interrupt request of a higher priority has been generated.
In addition, only when the PSR's IE bit = "1" (interrupts enabled) and the set value of the IL is smaller than the input interrupt level set using the interrupt priority register will the input interrupt request actually be accepted by the CPU.
For details on these interrupt control registers, as well as the device operation when an interrupt has occurred, refer to "ITC (Interrupt Controller)".

Intelligent DMA

The port input interrupt system can invoke an intelligent DMA (IDMA) through the use of its interrupt factor. This enables the port inputs to be used as a trigger to perform DMA transfer.
The following shows the IDMA channel numbers assigned to each interrupt factor:

	IDMA Ch.		IDMA Ch.
FPT0 input interrupt:	1	FPT4 input interrupt:	28
FPT1 input interrupt:	2	FPT5 input interrupt:	29
FPT2 input interrupt:	3	FPT6 input interrupt:	30
FPT3 input interrupt:	4	FPT7 input interrupt:	31

For IDMA to be invoked, the IDMA request and IDMA enable bits shown in Table 9.9 must be set to "1" in advance. Transfer conditions, etc. must also be set on the IDMA side in advance.

Table 9.9 Control Bits for IDMA Transfer

System	IDMA request bit	IDMA enable bit
FPT7	RP7(D7/0x40293)	DEP7(D7/0x40297)
FPT6	RP6(D6/0x40293)	DEP6(D6/0x40297)
FPT5	RP5(D5/0x40293)	DEP5(D5/0x40297)
FPT4	RP4(D4/0x40293)	DEP4(D4/0x40297)
FPT3	RP3(D3/0x40290)	DEP3(D3/0x40294)
FPT2	RP2(D2/0x40290)	DEP2(D2/0x40294)
FPT1	RP1(D1/0x40290)	DEP1(D1/0x40294)
FPT0	RP0(D0/0x40290)	DEP0(D0/0x40294)

If the IDMA request and enable bits are set to "1", IDMA is invoked through generation of an interrupt factor. No interrupt request is generated at that point. An interrupt request is generated after the DMA transfer is completed. The registers can also be set so as not to generate an interrupt, with only DMA transfers performed. For details on IDMA transfers and interrupt control upon completion of IDMA transfer, refer to "IDMA (Intelligent DMA)".

Trap vectors

The trap-vector address of each input default interrupt factor is set as follows:

	(BTA3 = high)	(BTA3 = low)
FPT0 input interrupt:	0x0080040	0x0C00040
FPT1 input interrupt:	0x0080044	0x0C00044
FPT2 input interrupt:	0x0080048	0x0C00048
FPT3 input interrupt:	0x008004C	0x0C0004C
FPK0 input interrupt:	0x0080050	0x0C00050
FPK1 input interrupt:	0x0080054	0x0C00054
FPT4 input interrupt:	0x0080110	0x0C00110
FPT5 input interrupt:	0x0080114	0x0C00114
FPT6 input interrupt:	0x0080118	0x0C00118
FPT7 input interrupt:	0x008011C	0x0C0011C

The base address of the trap table can be changed using the TTBR register (0x48134 to 0x48137).

I/O Memory for Input Interrupts

Table 9.10 shows the control bits for the port input and key input interrupts.

Table 9.10 Control Bits for Input Interrupts

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Port input 0/1 interrupt priority register	**0040260 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**PP1L2**	Port input 1 interrupt level	0 to 7				X	R/W	
		D5	**PP1L1**						X		
		D4	**PP1L0**						X		
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**PP0L2**	Port input 0 interrupt level	0 to 7				X	R/W	
		D1	**PP0L1**						X		
		D0	**PP0L0**						X		
Port input 2/3 interrupt priority register	**0040261 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**PP3L2**	Port input 3 interrupt level	0 to 7				X	R/W	
		D5	**PP3L1**						X		
		D4	**PP3L0**						X		
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**PP2L2**	Port input 2 interrupt level	0 to 7				X	R/W	
		D1	**PP2L1**						X		
		D0	**PP2L0**						X		
Key input interrupt priority register	**0040262 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**PK1L2**	Key input 1 interrupt level	0 to 7				X	R/W	
		D5	**PK1L1**						X		
		D4	**PK1L0**						X		
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**PK0L2**	Key input 0 interrupt level	0 to 7				X	R/W	
		D1	**PK0L1**						X		
		D0	**PK0L0**						X		
Key input, port input 0–3 interrupt enable register	**0040270 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**EK1**	Key input 1	1	Enabled	0	Disabled	0	R/W	
		D4	**EK0**	Key input 0					0	R/W	
		D3	**EP3**	Port input 3					0	R/W	
		D2	**EP2**	Port input 2					0	R/W	
		D1	**EP1**	Port input 1					0	R/W	
		D0	**EP0**	Port input 0					0	R/W	
Port input 4–7, clock timer, A/D interrupt enable register	**0040277 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**EP7**	Port input 7	1	Enabled	0	Disabled	0	R/W	
		D4	**EP6**	Port input 6					0	R/W	
		D3	**EP5**	Port input 5					0	R/W	
		D2	**EP4**	Port input 4					0	R/W	
		D1	**ECTM**	Clock timer					0	R/W	
		D0	**EADE**	A/D converter					0	R/W	
Key input, port input 0–3 interrupt factor flag register	**0040280 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**FK1**	Key input 1	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FK0**	Key input 0					X	R/W	
		D3	**FP3**	Port input 3					X	R/W	
		D2	**FP2**	Port input 2					X	R/W	
		D1	**FP1**	Port input 1					X	R/W	
		D0	**FP0**	Port input 0					X	R/W	
Port input 4–7, clock timer, A/D interrupt factor flag register	**0040287 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**FP7**	Port input 7	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FP6**	Port input 6					X	R/W	
		D3	**FP5**	Port input 5					X	R/W	
		D2	**FP4**	Port input 4					X	R/W	
		D1	**FCTM**	Clock timer					X	R/W	
		D0	**FADE**	A/D converter					X	R/W	
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA request register	**0040290 (B)**	D7	**R16TC0**	16-bit timer 0 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**RHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**RHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**RP3**	Port input 3					0	R/W	
		D2	**RP2**	Port input 2					0	R/W	
		D1	**RP1**	Port input 1					0	R/W	
		D0	**RP0**	Port input 0					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Serial I/F Ch.1,	**0040293**	D7	**RP7**	Port input 7	1	IDMA	0	Interrupt	0	R/W	
A/D,	**(B)**	D6	**RP6**	Port input 6		request		request	0	R/W	
port input 4–7		D5	**RP5**	Port input 5					0	R/W	
IDMA request		D4	**RP4**	Port input 4					0	R/W	
register		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**RADE**	A/D converter	1	IDMA	0	Interrupt	0	R/W	
		D1	**RSTX1**	SIF Ch.1 transmit buffer empty		request		request	0	R/W	
		D0	**RSRX1**	SIF Ch.1 receive buffer full					0	R/W	
Port input 0–3,	**0040294**	D7	**DE16TC0**	16-bit timer 0 comparison A	1	IDMA	0	IDMA	0	R/W	
high-speed	**(B)**	D6	**DE16TU0**	16-bit timer 0 comparison B		enabled		disabled	0	R/W	
DMA, 16-bit		D5	**DEHDM1**	High-speed DMA Ch.1					0	R/W	
timer 0		D4	**DEHDM0**	High-speed DMA Ch.0					0	R/W	
IDMA enable		D3	**DEP3**	Port input 3					0	R/W	
register		D2	**DEP2**	Port input 2					0	R/W	
		D1	**DEP1**	Port input 1					0	R/W	
		D0	**DEP0**	Port input 0					0	R/W	
Serial I/F Ch.1,	**0040297**	D7	**DEP7**	Port input 7	1	IDMA	0	IDMA	0	R/W	
A/D,	**(B)**	D6	**DEP6**	Port input 6		enabled		disabled	0	R/W	
port input 4–7		D5	**DEP5**	Port input 5					0	R/W	
IDMA enable		D4	**DEP4**	Port input 4					0	R/W	
register		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**DEADE**	A/D converter	1	IDMA	0	IDMA	0	R/W	
		D1	**DESTX1**	SIF Ch.1 transmit buffer empty		enabled		disabled	0	R/W	
		D0	**DESRX1**	SIF Ch.1 receive buffer full					0	R/W	
Port input	**00402C6**	D7	**SPT31**	FPT3 interrupt input port selection	11	10	01	00	0	R/W	
interrupt select	**(B)**	D6	**SPT30**		P23	P03	K53	K63	0		
register 1		D5	**SPT21**	FPT2 interrupt input port selection	11	10	01	00	0	R/W	
		D4	**SPT20**		P22	P02	K52	K62	0		
		D3	**SPT11**	FPT1 interrupt input port selection	11	10	01	00	0	R/W	
		D2	**SPT10**		P21	P01	K51	K61	0		
		D1	**SPT01**	FPT0 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPT00**		P20	P00	K50	K60	0		
Port input	**00402C7**	D7	**SPT71**	FPT7 interrupt input port selection	11	10	01	00	0	R/W	
interrupt select	**(B)**	D6	**SPT70**		P27	P07	P33	K67	0		
register 2		D5	**SPT61**	FPT6 interrupt input port selection	11	10	01	00	0	R/W	
		D4	**SPT60**		P26	P06	P32	K66	0		
		D3	**SPT51**	FPT5 interrupt input port selection	11	10	01	00	0	R/W	
		D2	**SPT50**		P25	P05	P31	K65	0		
		D1	**SPT41**	FPT4 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPT40**		P24	P04	K54	K64	0		
Port input	**00402C8**	D7	**SPPT7**	FPT7 input polarity selection	1	High level	0	Low level	1	R/W	
interrupt	**(B)**	D6	**SPPT6**	FPT6 input polarity selection		or		or	1	R/W	
input polarity		D5	**SPPT5**	FPT5 input polarity selection		Rising edge		Falling	1	R/W	
select register		D4	**SPPT4**	FPT4 input polarity selection				edge	1	R/W	
		D3	**SPPT3**	FPT3 input polarity selection					1	R/W	
		D2	**SPPT2**	FPT2 input polarity selection					1	R/W	
		D1	**SPPT1**	FPT1 input polarity selection					1	R/W	
		D0	**SPPT0**	FPT0 input polarity selection					1	R/W	
Port input	**00402C9**	D7	**SEPT7**	FPT7 edge/level selection	1	Edge	0	Level	1	R/W	
interrupt	**(B)**	D6	**SEPT6**	FPT6 edge/level selection					1	R/W	
edge/level		D5	**SEPT5**	FPT5 edge/level selection					1	R/W	
select register		D4	**SEPT4**	FPT4 edge/level selection					1	R/W	
		D3	**SEPT3**	FPT3 edge/level selection					1	R/W	
		D2	**SEPT2**	FPT2 edge/level selection					1	R/W	
		D1	**SEPT1**	FPT1 edge/level selection					1	R/W	
		D0	**SEPT0**	FPT0 edge/level selection					1	R/W	
Key input	**00402CA**	D7–4	**–**	reserved	–				–	–	0 when being read.
interrupt select	**(B)**	D3	**SPPK11**	FPK1 interrupt input port selection	11	10	01	00	0	R/W	
register		D2	**SPPK10**		P2[7:4]	P0[7:4]	K6[7:4]	K6[3:0]	0		
		D1	**SPPK01**	FPK0 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPPK00**		P2[4:0]	P0[4:0]	K6[4:0]	K5[4:0]	0		
Key input	**00402CC**	D7–5	**–**	reserved	–				–	–	0 when being read.
interrupt	**(B)**	D4	**SCPK04**	FPK04 input comparison	1	High	0	Low	0	R/W	
(FPK0) input		D3	**SCPK03**	FPK03 input comparison					0	R/W	
comparison		D2	**SCPK02**	FPK02 input comparison					0	R/W	
register		D1	**SCPK01**	FPK01 input comparison					0	R/W	
		D0	**SCPK00**	FPK00 input comparison					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Key input interrupt (FPK1) input comparison register	**00402CD (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**SCPK13**	FPK13 input comparison	1	High	0	Low	0	R/W	
		D2	**SCPK12**	FPK12 input comparison					0	R/W	
		D1	**SCPK11**	FPK11 input comparison					0	R/W	
		D0	**SCPK10**	FPK10 input comparison					0	R/W	
Key input interrupt (FPK0) input mask register	**00402CE (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**SMPK04**	FPK04 input mask	1	Interrupt enabled	0	Interrupt disabled	0	R/W	
		D3	**SMPK03**	FPK03 input mask					0	R/W	
		D2	**SMPK02**	FPK02 input mask					0	R/W	
		D1	**SMPK01**	FPK01 input mask					0	R/W	
		D0	**SMPK00**	FPK00 input mask					0	R/W	
Key input interrupt (FPK1) input mask register	**00402CF (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**SMPK13**	FPK13 input mask	1	Interrupt enabled	0	Interrupt disabled	0	R/W	
		D2	**SMPK12**	FPK12 input mask					0	R/W	
		D1	**SMPK11**	FPK11 input mask					0	R/W	
		D0	**SMPK10**	FPK10 input mask					0	R/W	

SPT71–SPT70: FPT7 interrupt input port selection (D[7:6]) / Port input interrupt select register 2 (0x402C7)
SPT61–SPT60: FPT6 interrupt input port selection (D[5:4]) / Port input interrupt select register 2 (0x402C7)
SPT51–SPT50: FPT5 interrupt input port selection (D[3:2]) / Port input interrupt select register 2 (0x402C7)
SPT41–SPT40: FPT4 interrupt input port selection (D[1:0]) / Port input interrupt select register 2 (0x402C7)
SPT31–SPT30: FPT3 interrupt input port selection (D[7:6]) / Port input interrupt select register 1 (0x402C6)
SPT21–SPT20: FPT2 interrupt input port selection (D[5:4]) / Port input interrupt select register 1 (0x402C6)
SPT11–SPT10: FPT1 interrupt input port selection (D[3:2]) / Port input interrupt select register 1 (0x402C6)
SPT01–SPT00: FPT0 interrupt input port selection (D[1:0]) / Port input interrupt select register 1 (0x402C6)

Select an input pin for port interrupt generation.

Table 9.11 Selecting Pins for Port Input Interrupts

Interrupt system	SPT settings			
	11	**10**	**01**	**00**
FPT7	P27	P07	P33	K67
FPT6	P26	P06	P32	K66
FPT5	P25	P05	P31	K65
FPT4	P24	P04	K54	K64
FPT3	P23	P03	K53	K63
FPT2	P22	P02	K52	K62
FPT1	P21	P01	K51	K61
FPT0	P20	P00	K50	K60

At cold start, SPT is set to "00". At hot start, SPT retains its state from prior to the initial reset.

SPPT7–SPPT0: Input polarity selection (D[7:0]) / Port interrupt input polarity select register (0x402C8)

Selects input signal porarity for port interrupt generation.

Write "1": High level or Rising edge
Write "0": Low level or Falling edge
Read: Valid

SPPTx is the input polarity select bit corresponding to the FPTx interrupt. When SPPTx is set to "1", the FPTx interrupt will be generated by a high level input or at the rising edge. When SPPTx is set to "0", the interrupt will be generated by a low level input or at the falling edge. An edge or a level interrupt is selected by the SEPTx bit.
At cold start, SPPT is set to "0" (low level). At hot start, SPPT retains its state from prior to the initial reset.

SEPT7–SEPT0: Edge/level selection (D[7:0]) / Port interrupt edge/level select register (0x402C9)

Selects an edge trigger or a level trigger for port interrupt generation.

Write "1": Edge
Write "0": Level
Read: Valid

SEPTx is the edge/level select bit corresponding to the FPTx interrupt. When SEPTx is set to "1", the FPTx interrupt will be generated at the signal edge. Either falling edge or rising edge can be selected by the SPPTx bit. When SEPTx is set to "0", the interrupt will be generated by the level (high or low) specified with the SPPTx bit.
At cold start, SEPT is set to "0" (level). At hot start, SEPT retains its state from prior to the initial reset.

SPPK11–SPPK10: FPK1 interrupt input port selection (D[3:2]) / Key input interrupt select register (0x402CA)
SPPK01–SPPK00: FPK0 interrupt input port selection (D[1:0]) / Key input interrupt select register (0x402CA)

Select an input-pin group for key interrupt generation.

Table 9.12 Selecting Pins for Key Input Interrupts

Interrupt system	SPPK settings			
	11	10	01	00
FPK1	P2[7:4]	P0[7:4]	K6[7:4]	K6[3:0]
FPK0	P2[4:0]	P0[4:0]	K6[4:0]	K5[4:0]

At cold start, SPPK is set to "00". At hot start, SPPK retains its state from prior to the initial reset.

SCPK13–SCPK10: FPK1 input comparison (D[3:0]) / FPK1 input comparison register (0x402CD)
SCPK04–SCPK00: FPK0 input comparison (D[4:0]) / FPK0 input comparison register (0x402CC)

Sets the conditions for key-input interrupt generation (timing of interrupt generation).

Write "1": Generated at falling edge
Write "0": Generated at rising edge
Read: Valid

SCPK0[4:0] is compared with the input state of five bits of the FPK0 input ports, and SCPK1[3:0] is compared with the input state of four bits of the FPK1 input ports, and when a change in states from a matched to an unmatched state occurs in either, an interrupt is generated (except for the inputs disabled from interrupt by the SMPK register).
At cold start, SCPK is set to "0" (rising edge). At hot start, SCPK retains its state from prior to the initial reset.

SMPK13–SMPK10: FPK1 input mask (D[3:0]) / FPK1 input mask register (0x402CF)
SMPK04–SMPK00: FPK0 input mask (D[4:0]) / FPK0 input mask register (0x402CE)

Sets conditions for key-input interrupt generation (interrupt enabled/disabled).

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

SMPK is an input mask register for each key-input interrupt system. Interrupts for bits set to "1" are enabled, and interrupts for bits set to "0" are disabled. A change in the state of an input pin that is disabled from interrupt does not affect interrupt generation.
At cold start, SMPK is set to "0" (interrupt disabled). At hot start, SMPK retains its state from prior to the initial reset.

PP0L2–PP0L0: Port input 0 interrupt level (D[2:0]) / Port input 0/1 interrupt priority register (0x40260)
PP1L2–PP1L0: Port input 1 interrupt level (D[6:4]) / Port input 0/1 interrupt priority register (0x40260)
PP2L2–PP2L0: Port input 2 interrupt level (D[2:0]) / Port input 2/3 interrupt priority register (0x40261)
PP3L2–PP3L0: Port input 3 interrupt level (D[6:4]) / Port input 2/3 interrupt priority register (0x40261)
PP4L2–PP4L0: Port input 4 interrupt level (D[2:0]) / Port input 4/5 interrupt priority register (0x4026C)
PP5L2–PP5L0: Port input 5 interrupt level (D[6:4]) / Port input 4/5 interrupt priority register (0x4026C)
PP6L2–PP6L0: Port input 6 interrupt level (D[2:0]) / Port input 6/7 interrupt priority register (0x4026D)
PP7L2–PP7L0: Port input 7 interrupt level (D[6:4]) / Port input 6/7 interrupt priority register (0x4026D)
PK0L2–PK0L0: Key input 0 interrupt level (D[2:0]) / Key input interrupt priority register (0x40262)
PK1L2–PK1L0: Key input 1 interrupt level (D[6:4]) / Key input interrupt priority register (0x40262)

Sets the priority level of the input interrupt.
PPxL and PKxL are interrupt priority registers corresponding to each port-input interrupt and key-input interrupt, respectively.
The priority level can be set for each interrupt group in the range of 0 to 7.
At initial reset, these registers becomes indeterminate.

EP3–EP0: Port input 3–0 interrupt enable (D[3:0]) /
Key input, port input 0–3 interrupt enable register (0x40270)
EP7–EP4: Port input 7–4 interrupt enable (D[5:2]) /
Port input 4–7, clock timer, A/D interrupt enable register (0x40277)
EK1, EK0: Key input 1, 0 interrupt enable (D[5:4]) /
Key input, port input 0–3 interrupt enable register (0x40270)

Enables or disables the generation of an interrupt to the CPU.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

EP and EK are interrupt enable bits corresponding to the port-input interrupt and the key-input interrupt, respectively. Interrupts for input systems set to "1" are enabled, and interrupts for input systems set to "0" are disabled.
At initial reset, these bits are set to "0" (interrupt disabled).

FP3–FP0: Port input 3–0 interrupt factor flag (D[3:0]) /
Key input, port input 0–3 interrupt factor flag register (0x40280)
FP7–FP4: Port input 7–4 interrupt factor flag (D[5:2]) /
Port input 4–7, clock timer, A/D interrupt factor flag register (0x40287)
FK1, FK0: Key input 1, 0 interrupt factor flag (D[5:4]) /
Key input, port input 0–3 interrupt factor flag register (0x40280)

Indicates the status of an input interrupt factor generated.

When read
Read "1": Interrupt factor has occurred
Read "0": No interrupt factor has occurred

When written using the reset-only method (default)
Write "1": Interrupt factor flag is reset
Write "0": Invalid

When written using the read/write method
Write "1": Interrupt flag is set
Write "0": Interrupt flag is reset

FP and FK are an interrupt factor flags corresponding to the port-input interrupt and the key-input interrupt, respectively. The flag is set to "1" when interrupt generation conditions are met.

At this time, if the following conditions are met, an interrupt to the CPU is generated:
1. The corresponding interrupt enable register bit is set to "1".
2. No other interrupt request of a higher priority has been generated.
3. The IE bit of the PSR is set to "1" (interrupts enabled).
4. The value set in the corresponding interrupt priority register is higher than the interrupt level (IL) of the CPU.
When using the interrupt factor of the port-input to request IDMA, note that even when the above conditions are met, no interrupt request to the CPU is generated for the interrupt factor that has occurred. If interrupts are enabled at the setting of IDMA, an interrupt is generated under the above conditions after the data transfer by IDMA is completed.

The interrupt factor flag is set to "1" whenever interrupt generation conditions are met, regardless of how the interrupt enable and interrupt priority registers are set.
If the next interrupt is to be accepted after an interrupt has occurred, it is necessary that the interrupt factor flag be reset, and that the PSR be set again (by setting the IE bit to "1" after setting the IL to a value lower than the level indicated by the interrupt priority register, or by executing the reti instruction).
The interrupt factor flag can be reset only by writing to it in the software. Note that if the PSR is set again to accept interrupts generated (or if the reti instruction is executed) without resetting the interrupt factor flag, the same interrupt occurs again. Note also that the value to be written to reset the flag is "1" when the reset-only method (RSTONLY = "1") is used, and "0" when the read/write method (RSTONLY = "0") is used.
At initial reset, all the flags become indeterminate, so be sure to reset them in the software.

RP3–RP0: Port input 3–0 IDMA request (D[3:0]) /
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA request register (0x40290)
RP7–RP4: Port input 7–4 IDMA request (D[7:4]) /
Serial I/F Ch.1, A/D, Port input 4–7 IDMA request register (0x40293)

Specifies whether to invoke IDMA when an interrupt factor occurs.

When using the set-only method (default)

Write "1": IDMA request
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA request
Write "0": Interrupt request
Read: Valid

RP7 to RP0 are IDMA request bits corresponding to the port-input 7 to 0 interrupts, respectively. If the bit is set to "1", IDMA is invoked when an interrupt factor occurs, thereby performing a programmed data transfer. If the bit is set to "0", normal interrupt processing is performed, without invoking IDMA.
For details on IDMA, refer to "IDMA (Intelligent DMA)".
At initial reset, RP set to "0" (interrupt request).

DEP3–DEP0: Port input 3–0 IDMA enable (D[3:0]) /
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA enable register (0x40294)
DEP7–DEP4: Port input 7–4 IDMA enable (D[7:4]) /
Serial I/F Ch.1, A/D, Port input 4–7 IDMA enable register (0x40297)

Enables IDMA transfer by means of an interrupt factor.

When using the set-only method (default)

Write "1": IDMA enabled
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA enabled
Write "0": IDMA disabled
Read: Valid

If DEP is set to "1", the IDMA request by the interrupt factor is enabled. If the register bit is set to "0", the IDMA request is disabled.
After an initial reset, DEP is set to "0" (IDMA disabled).

Programming Notes

(1) After an initial reset, the interrupt factor flags become indeterminate. To prevent generation of an unwanted interrupt or IDMA request, be sure to reset the flags in a program.

(2) To prevent regeneration of interrupts due to the same factor following the occurrence of an interrupt, always be sure to reset the interrupt factor flag before resetting the PSR or executing the reti instruction.

THIS PAGE IS BLANK.

E0C33 Family ASIC Macro Manual

IV ANALOG BLOCK

Table of Contents

IV-1 INTRODUCTION

The analog block consists of an A/D converter with 8 input channels.

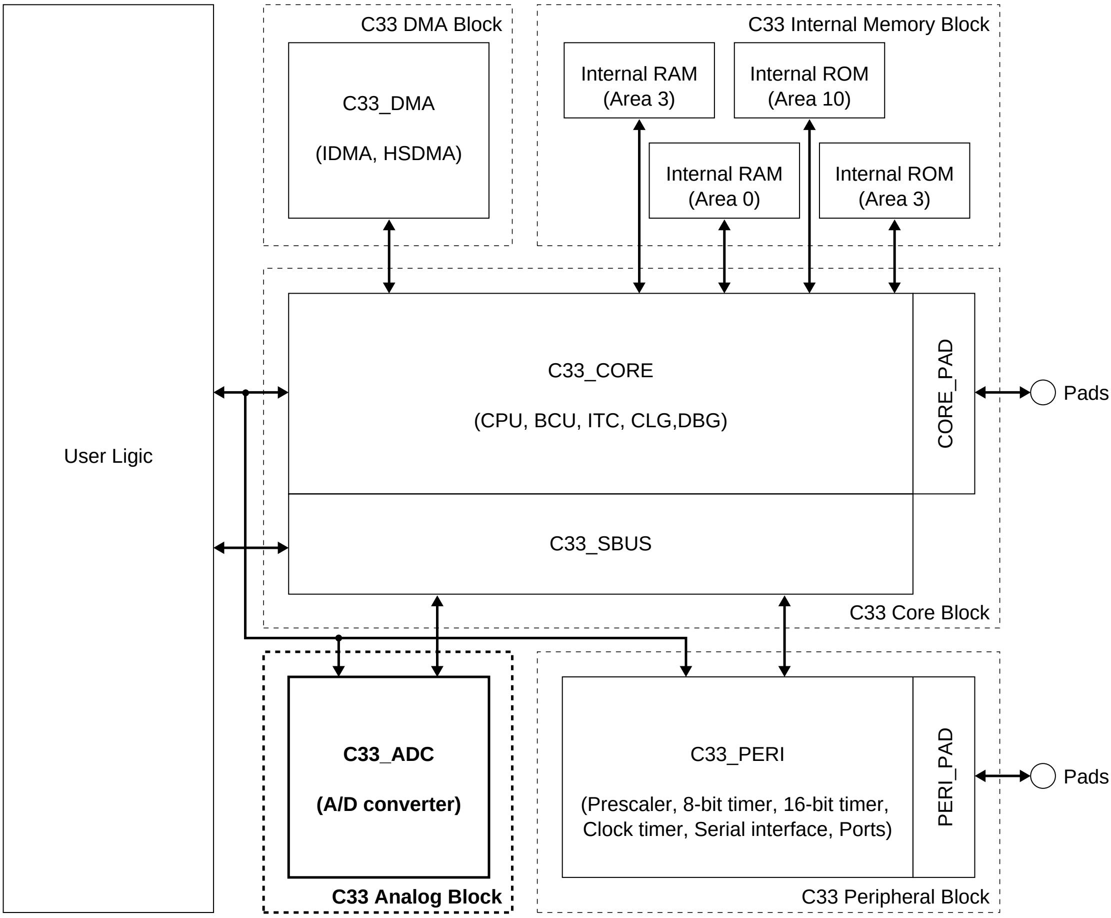


Figure 1.1 Analog Block

THIS PAGE IS BLANK.

IV-2 A/D CONVERTER

Features and Structure of A/D Converter

The Analog Block contains an A/D converter with the following features:

- Conversion method: Successive comparison
- Resolution: 10 bits
- Input channels: Maximum of 8
- Conversion time: Maximum of 10 µs (when a 2-MHz input clock is selected)
- Conversion range: Between AV_{RL} and AV_{DD} ($AV_{RL} = V_{SS}$ when the A/D converter is used)
- Two conversion modes can be selected:
  - Normal mode: Conversion is completed in one operation.
  - Continuous mode: Conversion is continuous and terminated through software control.
  - Continuous conversion of multiple channels can be performed in each mode.
- Four types of A/D-conversion start triggers can be selected:
  - Triggered by the external pin (#ADTRG)
  - Triggered by the compare match B of the 16-bit programmable timer 0
  - Triggered by the underflow of the 8-bit programmable timer 0
  - Triggered by the software
- A/D conversion results can be read out from a 10-bit data register.
- An interrupt is generated upon completion of A/D conversion.

Figure 2.1 shows the structure of the A/D converter.

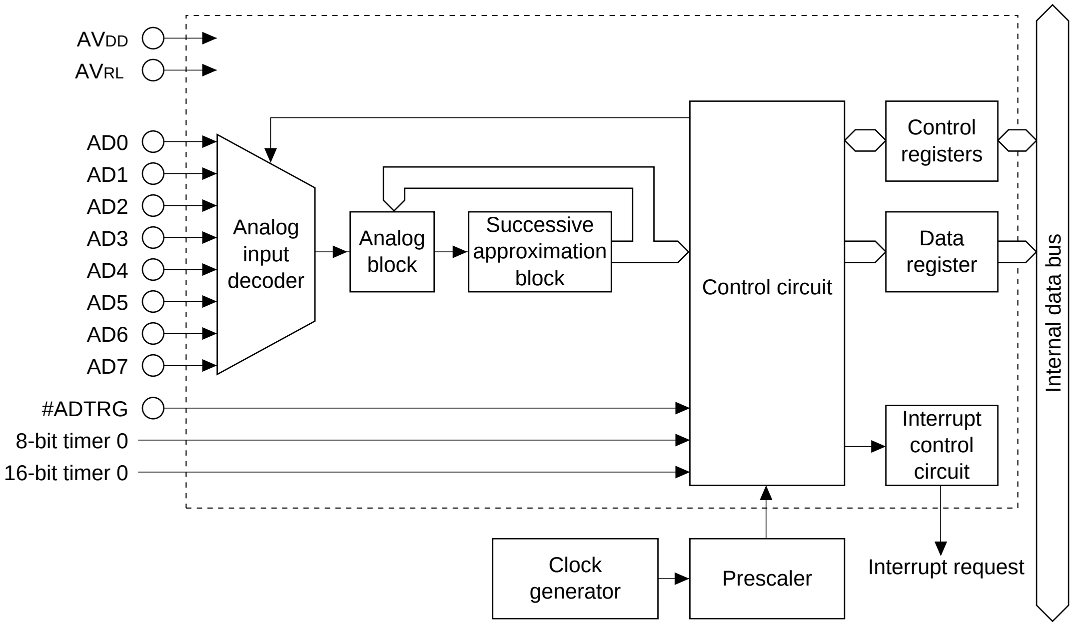


Figure 2.1 Structure of A/D Converter

I/O Pins of A/D Converter

Table 2.1 shows the pins used by the A/D converter.

Table 2.1 I/O Pins of A/D Converter

Pin name	I/O	Function	Function select bit
K52/#ADTRG	I	Input port / AD trigger	CFK52(D2)/K5 function select register(0x402C0)
K60/AD0	I	Input port / AD converter input 0	CFK60(D0)/K6 function select register(0x402C3)
K61/AD1	I	Input port / AD converter input 1	CFK61(D1)/K6 function select register(0x402C3)
K62/AD2	I	Input port / AD converter input 2	CFK62(D2)/K6 function select register(0x402C3)
K63/AD3	I	Input port / AD converter input 3	CFK63(D3)/K6 function select register(0x402C3)
K64/AD4	I	Input port / AD converter input 4	CFK64(D4)/K6 function select register(0x402C3)
K65/AD5	I	Input port / AD converter input 5	CFK65(D5)/K6 function select register(0x402C3)
K66/AD6	I	Input port / AD converter input 6	CFK66(D6)/K6 function select register(0x402C3)
K67/AD7	I	Input port / AD converter input 7	CFK67(D7)/K6 function select register(0x402C3)
AV_{DD}	–	Power supply for analog system (+)	–
AV_{RL}	–	Analog reference voltage (-)	–

AV_{DD} (analog power-supply pin)

AV_{DD} is the power-supply pin for the analog circuit. The voltage level supplied to this pin must be $AV_{DD} = V_{DDE}$.

AV_{RL} (reference-voltage input pin)

AV_{RL} is the reference-voltage input pin for the A/D converter. The input voltage range of the A/D converter is determined by the reference voltage of this pin (AV_{RL} to AV_{DD}). The voltage input to this pin must be $AV_{RL} = V_{SS}$.

Note: These pins should be set as $AV_{RL} = AV_{DD}$ if the A/D converter is not used.

AD[7:0] (analog-signal input pins)

The analog input pins AD7 (Ch.7) through AD0 (Ch.0) are shared with input port pins K67 through K60. Therefore, when these pins are used for analog input, they must be set for use with the A/D converter in the software. This setting can be made individually for each pin. At cold start, all these pins are set for input ports. The analog input voltage AV_{IN} can be input in the range of $AV_{RL} \leq AV_{IN} \leq AV_{RH}$.

#ADTRG (external-trigger input pin)

This pin is used to input a trigger signal to start A/D conversion from an external source. Since this pin is shared with input port K52, it must be set for use with the A/D converter in the software before an external trigger can be applied to the pin. At cold start, this pin is set for an input port.

Method for setting A/D-converter input pins

At cold start, the #ADTRG and AD[7:0] pins all are set for input ports Kxx (function select bit CFKxx = "0"). When using these pins for the A/D converter, write "1" to the function select bit CFKxx.
At hot start, these pins retain their state from prior to the reset.

Setting A/D Converter

When the A/D converter is used, the following settings must be made before an A/D conversion can be performed:

1. Setting analog input pins
2. Setting the input clock
3. Selecting the analog-conversion start and end channels
4. Setting the A/D conversion mode
5. Selecting a trigger
6. Setting the sampling time
7. Setting interrupt/IDMA/HSDMA

The following describes how to set each item. For details on how to set the analog input pins, refer to the preceding section. For details on how to set interrupt/DMA, refer to "A/D Converter Interrupt and DMA".

Note: Before making these settings, make sure the A/D converter is disabled (ADE (D2) / A/D enable register (0x40244) = "0"). Changing the settings while the A/D converter is enabled could cause a malfunction.

Setting the input clock

As explained in "Prescaler", the A/D conversion clock can be selected from among the eight types shown in Table 2.2 below. Use PSAD[2:0] (D[2:0]) / A/D clock control register (0x4014F) for this selection.

Table 2.2 Input Clock Selection

PSAD2	PSAD1	PSAD0	Division ratio
1	1	1	f_{PSCIN}/256
1	1	0	f_{PSCIN}/128
1	0	1	f_{PSCIN}/64
1	0	0	f_{PSCIN}/32
0	1	1	f_{PSCIN}/16
0	1	0	f_{PSCIN}/8
0	0	1	f_{PSCIN}/4
0	0	0	f_{PSCIN}/2

f_{PSCIN}: Prescaler input clock frequency

The selected clock is output from the prescaler to the A/D converter by writing "1" to PSONAD (D3) / A/D clock control register (0x4014F).

Notes:
- The A/D converter operates only when the prescaler is operating.
- The recommended input clock frequency is a maximum of 2 MHz.
- Do not start an A/D conversion when the clock output from the prescaler to the A/D converter is turned off, and do not turn off the prescaler's clock output when an A/D conversion is underway. This could cause the A/D converter to operate erratically.

Selecting analog-conversion start and end channels

Select the channel in which the A/D conversion is to be performed from among the pins (channels) that have been set for analog input. To enable A/D conversions in multiple channels to be performed successively through one convert operation, specify the conversion start and conversion end channels.

Conversion start channel: CS[2:0] (D[2:0]) / A/D channel register (0x40243)

Conversion end channel: CE[2:0] (D[5:3]) / A/D channel register (0x40243)

Table 2.3 Relationship between CS/CE and Input Channel

CS2/CE2	CS1/CE1	CS0/CE0	Channel selected
1	1	1	AD7
1	1	0	AD6
1	0	1	AD5
1	0	0	AD4
0	1	1	AD3
0	1	0	AD2
0	0	1	AD1
0	0	0	AD0

Example: Operation of one A/D conversion
CS[2:0] = "0", CE[2:0] = "0": Converted only in AD0
CS[2:0] = "0", CE[2:0] = "3": Converted in the following order: AD0→AD1→AD2→AD3
CS[2:0] = "5", CE[2:0] = "1": Converted in the following order: AD5→AD6→AD7→AD0→AD1

Note: Only conversion-channel input pins that have been set for use with the A/D converter can be set using the CS and CE bits.

Setting the A/D conversion mode

The A/D converter can operate in one of the following two modes. This operation mode is selected using MS (D5) / A/D trigger register (0x40242).

1. Normal mode (MS = "0")

   All inputs in the range of channels set using the CS and CE bits are A/D converted once and then stopped.

2. Continuous mode (MS = "1")

   A/D conversions in the range of channels set using the CS and CE bits are executed successively until stopped by the software.

At initial reset, the normal mode is selected.

Selecting a trigger

Use TS[1:0] (D[4:3]) / A/D trigger register (0x40242) to select a trigger to start A/D conversion from among the four types shown in Table 2.4.

Table 2.4 Trigger Selection

TS1	TS0	Trigger
1	1	External trigger (K52/#ADTRG)
1	0	8-bit programmable timer 0
0	1	16-bit programmable timer 0
0	0	Software

1. External trigger

The signal input to the #ADTRG pin is used as a trigger.
When this trigger is used, the K52 pin must be set for #ADTRG in advance by writing "1" to CFK52 (D2) / K5 function select register (0x402C0).
A/D conversion is started at a falling edge of the #ADTRG signal.

2. Programmable timer

The underflow signal of 8-bit programmable timer 0 or the comarison match B signal of the 16-bit programmable timer 0 is used as a trigger. Since the cycle can be programmed using each timer, this trigger is effective when cyclic A/D conversions are required.
For details on how to set a timer, refer to the explanation of each programmable timer in this manual.

3. Software trigger

Writing "1" to ADST (D1) / A/D enable register (0x40244) in the software serves as a trigger to start A/D conversion.

Setting the sampling time

The A/D converter contains ST[1:0] (D[1:0]) / A/D sampling register (0x40245) that allows the analog-signal input sampling time to be set in four steps (3, 5, 7, or 9 times the input clock period).
However, this register should be used as set by default (ST = "11"; x9 clock periods).

Control and Operation of A/D Conversion

Figure 2.2 shows the operation of the A/D converter.

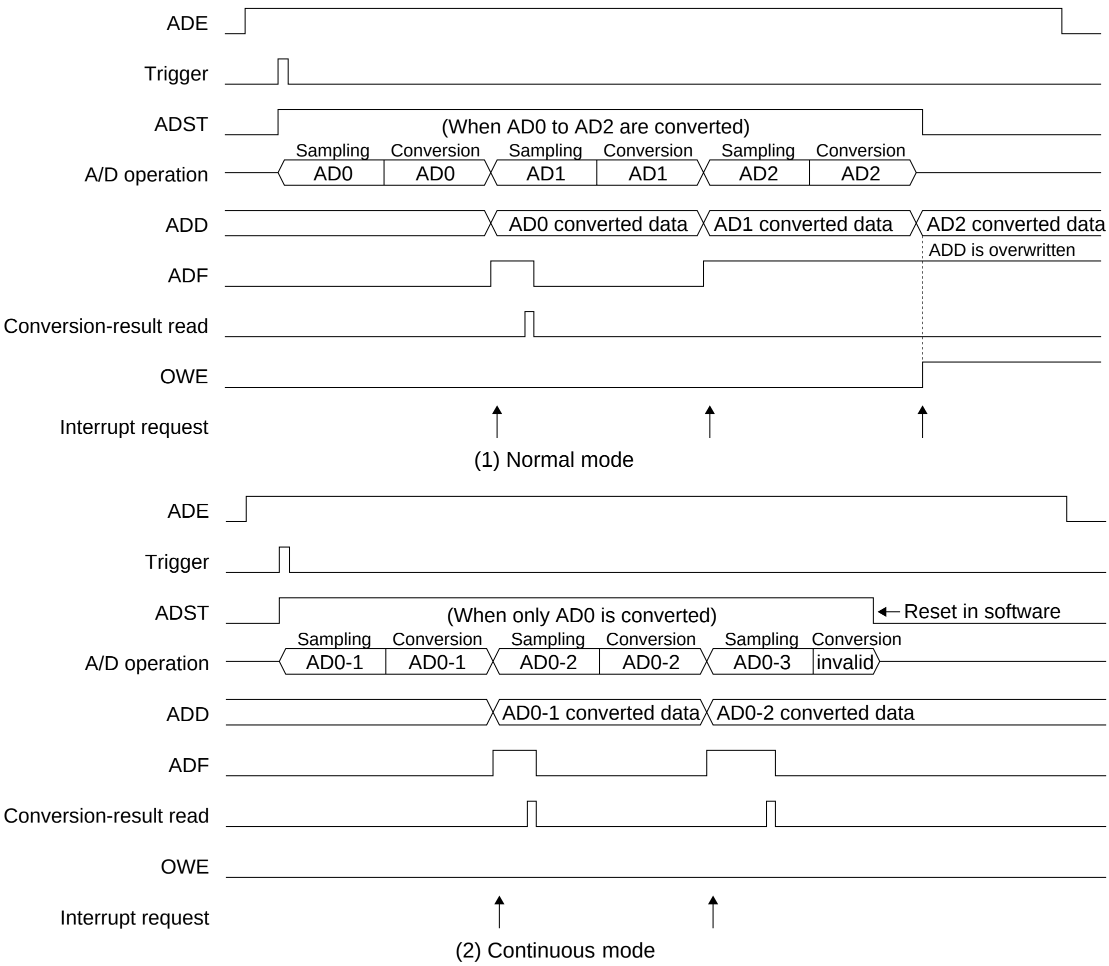


Figure 2.2 Operation of A/D Converter

Starting up the A/D converter circuit

After the settings specified in the preceding section have been made, write "1" to ADE (D2) / A/D enable register (0x40244) to enable the A/D converter. The A/D converter is thereby readied to accept a trigger to start A/D conversion. To set the A/D converter again, or if it is not be used, set ADE to "0".

Starting A/D conversion

When a trigger is input while ADE = "1", A/D conversion is started. If a software trigger has been selected, A/D conversion is started by writing "1" to ADST (D1) / A/D enable register (0x40244).
Only the trigger selected using TS[1:0] (D[4:3]) / A/D trigger register (0x40242) are valid; no other trigger is accepted.

When a trigger is input, the A/D converter samples and A/D-converts the analog input signal, beginning with the conversion start channel selected by CS[2:0].
Upon completion of the A/D conversion in that channel, the A/D converter stores the conversion result, in 10-bit data registers ADD[9:0] (ADD[9:8] = D[1:0]/0x40241, ADD[7:0] = D[7:0]/0x40240), and sets the conversion-complete flag ADF (D3) / A/D enable register (0x40244) and interrupt factor flag FADE (D0) / Port input 4–7, clock timer and A/D interrupt factor flag register (0x40287). If multiple channels are specified using CS[2:0] and CE[2:0], A/D conversions in the subsequent channels are performed in succession.

The ADST used for the software trigger is set to "1" during A/D conversion, even when it is started by some other trigger, so it can be used as an A/D-conversion status bit.
The channel in which conversion is underway can be identified by reading CH[2:0] (D[2:0]) / A/D trigger register (0x40242).

Reading out A/D conversion results

As explained earlier, the results of A/D conversion are stored in the ADD[9:0] register each time conversion in one channel is completed. Since an interrupt can be generated simultaneously, this interrupt is normally used to read out the converted data. In addition, be sure to reset the interrupt factor flag (by writing "0") to prepare the A/D converter for the next operation.
Since the interrupt factor of the A/D converter can also be used to invoke DMA, the conversion results can automatically be transferred to a specified memory location.

If multiple A/D conversion channels are specified, the conversion results in one channel must be read out prior to completion of conversion in the next channel. If the A/D conversion currently under way is completed before the previous conversion results are read out, the ADD[9:0] register is overwritten with the new conversion results.
If ADD[9:0] is updated when the conversion-complete flag ADF = "1" (before the converted data is read out), the overwrite-error flag OWE (D0) / A/D enable register (0x40244) is set to "1". The conversion-complete flag ADF is reset to "0" when the converted data is read out. If ADD[9:0] is updated when ADF = "0", OWE remains at "0", indicating that the operation has been completed normally. When reading out data, also read the OWE flag also to make sure the data is valid. Once OWE is set, it remains set until it is reset to "0" in the software. Note also that if OWE is set, ADF also is set. In this case, read out the converted data and reset ADF.

Terminating A/D conversion

- **For normal mode (MS = "1")**
  In the normal mode, A/D conversion is performed successively from the conversion start channel specified using CS[2:0] to the conversion end channel specified using CE[2:0], and is completed after these conversions are executed in one operation. ADST is reset to "0" upon completion of the conversion.

- **For continuous mode (MS = "0")**
  In the continuous mode, A/D conversion from the conversion-start to the conversion-end channels is executed repeatedly, without being stopped in the hardware. To terminate conversion, therefore, ADST must be reset to "0" in the software. The A/D conversion being executed when ADST is reset to "0" is forcibly stopped. Therefore, the results of this conversion cannot be obtained.

- **Forced termination**
  In either normal or continuous mode, A/D conversion is immediately terminated by writing "0" to ADST. The results of the conversion then under-way cannot be obtained.
  In addition, ADST is reset to "0" by writing "0" to ADE, so the conversion under-way is terminated.

A/D Converter Interrupt and DMA

Upon completion of A/D conversion in each channel, the A/D converter generates an interrupt and invokes the DMA if necessary.

Control registers of the interrupt controller

The following shows the interrupt control registers available for the A/D converter:

Interrupt factor flag: FADE (D0) / Port input 4–7, clock timer, A/D interrupt factor flag register (0x40287)
Interrupt enable: EADE (D0) / Port input 4–7, clock timer, A/D interrupt enable register (0x40277)
Interrupt level: PAD[2:0] (D[6:4]) / Serial I/F Ch.1, A/D interrupt priority register (0x4026A)

The A/D converter sets the interrupt factor flag to "1" when A/D conversion in one channel is completed, and the conversion results are stored in the ADD register. At this time, if the interrupt enable register bit has been set to "1", an interrupt request is generated.
Interrupts can be disabled by leaving the interrupt enable register bit set to "0". The interrupt factor flag is set to "1" upon completion of A/D conversion in each channel, regardless of the setting of the interrupt enable register (even when it is set to "0").
The interrupt priority register sets the priority level (0 to 7) of an interrupt. An interrupt request to the CPU is accepted no other interrupt request of a higher priority has been generated.
In addition, it is only when the PSR's IE bit = "1" (interrupts enabled) and the set value of the IL is smaller than the A/D-converter interrupt level set by the interrupt priority register, that the A/D converter's interrupt request is actually accepted by the CPU.
For details on these interrupt control registers, as well as the device operation when an interrupt has occurred, refer to "ITC (Interrupt Controller)".

Intelligent DMA

The A/D converter can invoke the intelligent DMA (IDMA) through the use of its interrupt factor. This allows the conversion results to be transferred to a specified memory location with no need to execute an interrupt processing routine.
The IDMA channel number assigned to the A/D converter is 0x1B.

Before IDMA can be invoked, the IDMA request and IDMA enable bits must be set to "1". Transfer conditions on the IDMA side must also be set in advance.
IDMA request: RADE (D2)/ Serial I/F Ch.1, A/D, Port input 4–7 IDMA request register (0x40293)
IDMA enable: DEADE (D2)/ Serial I/F Ch.1, A/D, Port input 4–7 IDMA enable register (0x40297)

If an interrupt factor occurs when the IDMA request and IDMA enable bits are set to "1", IDMA is invoked. No interrupt request is generated at that point. An interrupt request is generated upon completion of the DMA transfer. Otherwise, the bit can be set so as not to generate an interrupt, with only a DMA transfer performed.
For details on DMA transfers and how to control interrupts upon completion of a DMA transfer, refer to "IDMA (Intelligent DMA)".

High-speed DMA

The A/D interrupt factor can also invoke high-speed DMA (HSDMA).
The following shows the HSDMA channel number and trigger set-up bit:

Table 2.5 HSDMA Trigger Set-up Bits

HSDMA channel	Trigger set-up bits
0	HSD0S[3:0] (D[3:0]) / HSDMA Ch.0/1 trigger set-up register (0x40298)
1	HSD1S[3:0] (D[7:4]) / HSDMA Ch.0/1 trigger set-up register (0x40298)
2	HSD2S[3:0] (D[3:0]) / HSDMA Ch.2/3 trigger set-up register (0x40299)
3	HSD3S[3:0] (D[7:4]) / HSDMA Ch.2/3 trigger set-up register (0x40299)

For HSDMA to be invoked, the trigger set-up bits should be set to "1100" in advance. Transfer conditions, etc. must also be set on the HSDMA side.
If the A/D interrupt factor is selected as the HSDMA trigger, the HSDMA channel is invoked through generation of the interrupt factor.
For details on HSDMA transfer, refer to "HSDMA (High-Speed DMA)".

Trap vector

The A/D converter's interrupt trap-vector default address is set as follows:
When BTA3 = high: 0x0080100
When BTA3 = low: 0x0C00100

The base address of the trap table can be changed using the TTBR register (0x48134 to 0x48137).

I/O Memory of A/D Converter

Table 2.6 shows the control bits of the A/D converter.
For details on the I/O memory of the prescaler used to set clocks, refer to "Prescaler". For details on the I/O memory of the programmable timers used for a trigger, refer to "8-Bit Programmable Timers" or "16-Bit Programmable Timers".

Table 2.6 Control Bits of A/D Converter

Register name	Address	Bit	Name	Function	Setting			Init.	R/W	Remarks
A/D conversion result (low-order) register	**0040240 (B)**	D7	**ADD7**	A/D converted data	0x0 to 0x3FF			0	R	
		D6	**ADD6**	(low-order 8 bits)	(low-order 8 bits)			0		
		D5	**ADD5**	ADD0 = LSB				0		
		D4	**ADD4**					0		
		D3	**ADD3**					0		
		D2	**ADD2**					0		
		D1	**ADD1**					0		
		D0	**ADD0**					0		
A/D conversion result (high-order) register	**0040241 (B)**	D7–2	**–**	–	–			–	–	0 when being read.
		D1	**ADD9**	A/D converted data	0x0 to 0x3FF			0	R	
		D0	**ADD8**	(high-order 2 bits) ADD9 = MSB	(high-order 2 bits)			0		
A/D trigger register	**0040242 (B)**	D7–6	**–**	–	–			–	–	0 when being read.
		D5	**MS**	A/D conversion mode selection	1 Continuous	0 Normal		0	R/W	
		D4	**TS1**	A/D conversion trigger selection	TS[1:0]		Trigger	0	R/W	
		D3	**TS0**		1	1	#ADTRG pin	0		
					1	0	8-bit timer 0			
					0	1	16-bit timer 0			
					0	0	Software			
		D2	**CH2**	A/D conversion channel status	CH[2:0]		Channel	0	R	
		D1	**CH1**		1 1	1	AD7	0		
		D0	**CH0**		1 1	0	AD6	0		
					1 0	1	AD5			
					1 0	0	AD4			
					0 1	1	AD3			
					0 1	0	AD2			
					0 0	1	AD1			
					0 0	0	AD0			
A/D channel register	**0040243 (B)**	D7–6	**–**	–	–			–	–	0 when being read.
		D5	**CE2**	A/D converter	CE[2:0]		End channel	0	R/W	
		D4	**CE1**	end channel selection	1 1	1	AD7	0		
		D3	**CE0**		1 1	0	AD6	0		
					1 0	1	AD5			
					1 0	0	AD4			
					0 1	1	AD3			
					0 1	0	AD2			
					0 0	1	AD1			
					0 0	0	AD0			
		D2	**CS2**	A/D converter	CS[2:0]		Start channel	0	R/W	
		D1	**CS1**	start channel selection	1 1	1	AD7	0		
		D0	**CS0**		1 1	0	AD6	0		
					1 0	1	AD5			
					1 0	0	AD4			
					0 1	1	AD3			
					0 1	0	AD2			
					0 0	1	AD1			
					0 0	0	AD0			
A/D enable register	**0040244 (B)**	D7–4	**–**	–	–			–	–	0 when being read.
		D3	**ADF**	Conversion-complete flag	1 Completed	0 Run/Standby		0	R	Reset when ADD is read.
		D2	**ADE**	A/D enable	1 Enabled	0 Disabled		0	R/W	
		D1	**ADST**	A/D conversion control/status	1 Start/Run	0 Stop		0	R/W	
		D0	**OWE**	Overwrite error flag	1 Error	0 Normal		0	R/W	Reset by writing 0.
A/D sampling register	**0040245 (B)**	D7–2	**–**	–	–			–	–	0 when being read.
		D1	**ST1**	Input signal sampling time setup	ST[1:0]		Sampring time	1	R/W	Use with 9 clocks.
		D0	**ST0**		1	1	9 clocks	1		
					1	0	7 clocks			
					0	1	5 clocks			
					0	0	3 clocks			

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Serial I/F Ch.1, A/D interrupt priority register	**004026A (B)**	D7	–	reserved	–				–	–	0 when being read.
		D6	**PAD2**	A/D converter interrupt level	0 to 7				X	R/W	
		D5	**PAD1**						X		
		D4	**PAD0**						X		
		D3	–	reserved	–				–	–	0 when being read.
		D2	**PSIO12**	Serial interface Ch.1 interrupt level	0 to 7				X	R/W	
		D1	**PSIO11**						X		
		D0	**PSIO10**						X		
Port input 4–7, clock timer, A/D interrupt enable register	**0040277 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**EP7**	Port input 7	1	Enabled	0	Disabled	0	R/W	
		D4	**EP6**	Port input 6					0	R/W	
		D3	**EP5**	Port input 5					0	R/W	
		D2	**EP4**	Port input 4					0	R/W	
		D1	**ECTM**	Clock timer					0	R/W	
		D0	**EADE**	A/D converter					0	R/W	
Port input 4–7, clock timer, A/D interrupt factor flag register	**0040287 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**FP7**	Port input 7	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FP6**	Port input 6					X	R/W	
		D3	**FP5**	Port input 5					X	R/W	
		D2	**FP4**	Port input 4					X	R/W	
		D1	**FCTM**	Clock timer					X	R/W	
		D0	**FADE**	A/D converter					X	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA request register	**0040293 (B)**	D7	**RP7**	Port input 7	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RP6**	Port input 6					0	R/W	
		D5	**RP5**	Port input 5					0	R/W	
		D4	**RP4**	Port input 4					0	R/W	
		D3	–	reserved	–				–	–	0 when being read.
		D2	**RADE**	A/D converter	1	IDMA request	0	Interrupt request	0	R/W	
		D1	**RSTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**RSRX1**	SIF Ch.1 receive buffer full					0	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA enable register	**0040297 (B)**	D7	**DEP7**	Port input 7	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DEP6**	Port input 6					0	R/W	
		D5	**DEP5**	Port input 5					0	R/W	
		D4	**DEP4**	Port input 4					0	R/W	
		D3	–	reserved	–				–	–	0 when being read.
		D2	**DEADE**	A/D converter	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D1	**DESTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**DESRX1**	SIF Ch.1 receive buffer full					0	R/W	
K5 function select register	**00402C0 (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**CFK54**	K54 function selection	1	#DMAREQ3	0	K54	0	R/W	
		D3	**CFK53**	K53 function selection	1	#DMAREQ2	0	K53	0	R/W	
		D2	**CFK52**	K52 function selection	1	#ADTRG	0	K52	0	R/W	
		D1	**CFK51**	K51 function selection	1	#DMAREQ1	0	K51	0	R/W	
		D0	**CFK50**	K50 function selection	1	#DMAREQ0	0	K50	0	R/W	
K6 function select register	**00402C3 (B)**	D7	**CFK67**	K67 function selection	1	AD7	0	K67	0	R/W	
		D6	**CFK66**	K66 function selection	1	AD6	0	K66	0	R/W	
		D5	**CFK65**	K65 function selection	1	AD5	0	K65	0	R/W	
		D4	**CFK64**	K64 function selection	1	AD4	0	K64	0	R/W	
		D3	**CFK63**	K63 function selection	1	AD3	0	K63	0	R/W	
		D2	**CFK62**	K62 function selection	1	AD2	0	K62	0	R/W	
		D1	**CFK61**	K61 function selection	1	AD1	0	K61	0	R/W	
		D0	**CFK60**	K60 function selection	1	AD0	0	K60	0	R/W	

CFK52: K52 pin function selection (D2) / K5 function select register (0x402C0)
CFK67–CFK60: K6[7:0] pin function selection (D[7:0]) / K6 function select register (0x402C3)

Selects the pins used by the A/D converter.

Write "1": A/D converter
Write "0": Input port
Read: Valid

When an external trigger is used, write "1" to CFK52 to set the K52 pin for external trigger input #ADTRG. Select the pin used for analog input from among K60 (AD0) through K67 (AD7) by writing "1" to CFK60 through CFK67. If the function select bit for a pin is set to "0", the pin is set for an input port.
At cold start, CFK is set to "0" (input port). At hot start, CFK retains its state from prior to the initial reset.

ADD9–ADD0: A/D converted data (D[1:0]) / A/D conversion result (high-order) register (0x40241)
(D[7:0]) / A/D conversion result (low-order) register (0x40240)

Stores the results of A/D conversion.
The LSB is stored in ADD0, and the MSB is stored in ADD9. ADD0 and ADD1 are mapped to bits D0 and D1 at the address 0x40241, but bits D2 through D7 are always 0 when read.
This is a read-only register, so writing to this register is ignored.
At initial reset, the data in this register is cleared to "0".

MS: A/D conversion mode selection (D5) / A/D trigger register (0x40242)

Selects an A/D conversion mode.

Write "1": Continuous mode
Write "0": Normal mode
Read: Valid

The A/D converter is set for the continuous mode by writing "1" to MS. In this mode, A/D conversions in the range of the channels selected using CS and CE are executed continuously until stopped in the software.
When MS = "0", the A/D converter operates in the normal mode. In this mode, A/D conversion is completed after all inputs in the range of the channels selected by CS and CE are converted in one operation.
At initial reset, MS is set to "0" (normal mode).

TS1–TS0: Trigger selection (D[4:3]) / A/D trigger register (0X40242)

Selects a trigger to start A/D conversion.

Table 2.7 Trigger Selection

TS1	TS0	Trigger
1	1	External trigger (K52/#ADTRG)
1	0	8-bit programmable timer 0
0	1	16-bit programmable timer 0
0	0	Software

When an external trigger is used, use the CFK52 bit to set the K52 pin for #ADTRG.
When a programmable timer is used, since its underflow signal (8-bit timer) or comarison match B signal (16-bit timer) serves as a trigger, set the cycle and other parameters for the programmable timer.
At initial reset, TS is set to "0" (software trigger).

CH2–CH0: Conversion channel status (D[2:0]) / A/D trigger register (0X40242)

Indicates the channel number (0 to 7) currently being A/D-converted.
When A/D conversion is performed in multiple channels, read this bit to identify the channel in which conversion is underway.
At initial reset, CH is set to "0" (AD0).

CE2–CE0: Conversion end-channel setup (D[5:3]) / A/D channel register (0x40243)

Sets the conversion end channel by selecting a channel number from 0 to 7.
Analog inputs can be A/D-converted successively from the channel set using CS to the channel set using this bit in one operation. If only one channel is to be A/D converted, set the same channel number in both the CS and CE bits.
At initial reset, CE is set to "0" (AD0).

CS2–CS0: Conversion start-channel setup (D[2:0]) / A/D channel register (0x40243)

Sets the conversion start channel by selecting a channel number from 0 to 7.
Analog inputs can be A/D-converted successively from the channel set using this bit to the channel set using CE in one operation. If only one channel is to be A/D converted, set the same channel number in both the CS and CE bits.
At initial reset, CS is set to "0" (AD0).

ADF: Conversion-complete flag (D3) / A/D enable register (0x40244)

Indicates that A/D conversion has been completed.

Read "1": Conversion completed
Read "0": Being converted or standing by
Write: Invalid

This flag is set to "1" when A/D conversion is completed, and the converted data is stored in the data register and is reset to "0" when the converted data is read out. When A/D conversion is performed in multiple channels, if the next A/D conversion is completed while ADF = "1" (before the converted data is read out), the data register is overwritten with the new conversion results, causing an overrun error to occur. Therefore, ADF must be reset by reading out the converted data before the next A/D conversion is completed.
At initial reset, ADF is set to "0" (being converted or standing by).

ADE: A/D enable (D2) / A/D enable register (0x40244)

Enables the A/D converter (readied for conversion).

Write "1": Enabled
Write "0": Disabled
Read: Valid

When ADE is set to "1", the A/D converter is enabled, meaning it is ready to start A/D conversion (i.e., ready to accept a trigger). When ADE = "0", the A/D converter is disabled, meaning it is unable to accept a trigger.
Before setting the conversion mode, start/end channels, etc. for the A/D converter, be sure to reset ADE to "0". This helps to prevent the A/D converter from operating erratically.
At initial reset, ADE is set to "0" (disabled).

ADST: A/D conversion control/status (D1) / A/D enable register (0x40244)

Controls A/D conversion.

Write "1": Software trigger
Write "0": A/D conversion is stopped
Read: Valid

If A/D conversion is to be started by a software trigger, set ADST to "1". If any other trigger is used, ADST is automatically set to "1" by the hardware.
ADST remains set while A/D conversion is underway.
In normal mode, upon completion of A/D conversion in selected channels, ADST is reset to "0" and the A/D conversion circuit is turned off. To stop A/D conversion during operation in continuous mode or forcibly terminate A/D conversion, reset ADST by writing "0".
When ADE = "0" (A/D conversion disabled), ADST is fixed to "0", with no trigger accepted. In addition, ADST is reset to "0" when ADE is reset by writing "0" during A/D conversion.
At initial reset, ADST is set to "0" (A/D conversion stopped).

OWE: Overwrite-error flag (D0) / A/D enable register (0x40244)

Indicates that the converted data has been overwritten.

Read "1": Overwritten
Read "0": Normal
Write "1": Invalid
Write "0": Flag is set

During A/D conversion in multiple channels, if the conversion results for the next channel are written to the converted-data register (overwritten) before the converted data is read out to reset the conversion-complete flag ADF that has been set through conversion of the preceding channel, OWE is set to "1". When ADF is reset, because this means that the converted data has been read out, OWE is not set.
Once OWE is set to "1", it remains set until it is reset by writing "0" in the software.
At initial reset, OWE is set to "0" (normal).

ST1–ST0: Sampling-time setup (D[1:0]) / A/D sampling register (0x40245)

Sets the analog input sampling time.

Table 2.8 Sampling Time

ST1	ST0	Sampling Time
1	1	9-clock period
1	0	7-clock period
0	1	5-clock period
0	0	3-clock period

The A/D converter input clock is used for counting.

At initial reset, ST is set to "11" (9-clock period).
To maintain the conversion accuracy, use ST as set by default (9-clock period).

PAD2–PAD0: A/D converter interrupt level (D[6:4]) / Serial I/F Ch.1, A/D interrupt priority register (0x4026A)

Sets the priority level of the A/D-converter interrupt in the range of 0 to 7.
At initial reset, PAD becomes indeterminate.

EADE: A/D converter interrupt enable (D0) / Port input 4–7, clock timer, A/D interrupt enable register (0x40277)

Enables or disables an interrupt to the CPU generated by the A/D converter.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

EADE is an interrupt enable bit to control the A/D converter interrupt.
When EADE is set to "1", the A/D converter interrupt is enabled. When EADE is set to "0", the A/D-converter interrupt is disabled.
At initial reset, EADE is set to "0" (interrupt disabled).

FADE: A/D converter interrupt factor flag (D0) / Port input 4–7, clock timer, A/D interrupt factor flag register (0x40287)

Indicates the status of an A/D-converter interrupt factor generated.

When read

Read "1": Interrupt factor has occurred
Read "0": No interrupt factor has occurred

When written using the reset-only method (default)

Write "1": Interrupt factor flag is reset
Write "0": Invalid

When written using the read/write method

Write "1": Interrupt flag is set
Write "0": Interrupt flag is reset

FADE is the interrupt factor flag of the A/D converter. It is set to "1" upon completion of A/D conversion in one channel (i.e., when the conversion results are written into the ADD register).
At this time, if the following conditions are met, an interrupt to the CPU is generated:
1. The corresponding interrupt enable register bit is set to "1".
2. No other interrupt request of a higher priority has been generated.
3. The IE bit of the PSR is set to "1" (interrupts enabled).
4. The value set in the corresponding interrupt priority register is higher than the interrupt level (IL) of the CPU.
When using the interrupt factor of the A/D converter to request IDMA, note that even when the above conditions are met, no interrupt request to the CPU is generated for the interrupt factor that has occurred. If interrupts are enabled at the setting of IDMA, an interrupt is generated under the above conditions after the data transfer by IDMA is completed.

The interrupt factor flag is set to "1" whenever interrupt generation conditions are met, regardless of how the interrupt enable and interrupt priority registers are set.

If the next interrupt is to be accepted after an interrupt has occurred, it is necessary that the interrupt factor flag be reset, and that the PSR be set again (by setting the IE bit to "1" after setting the IL to a value lower than the level indicated by the interrupt priority register, or by executing the reti instruction).
The interrupt factor flag can be reset only by writing to it in the software. Note that if the PSR is set again to accept interrupts generated (or if the reti instruction is executed) without resetting the interrupt factor flag, the same interrupt occurs again. Note also that the value to be written to reset the flag is "1" when the reset-only method (RSTONLY = "1") is used, and "0" when the read/write method (RSTONLY = "0") is used.
At initial reset, the content of FADE becomes indeterminate, so be sure to reset it in the software.

RADE: A/D converter IDMA request (D2) / Serial I/F Ch.1, A/D, port input 4–7 IDMA request register (0x40293)

Specifies whether to invoke IDMA when an interrupt factor occurs.

When using the set-only method (default)

Write "1": IDMA request
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA request
Write "0": Interrupt request
Read: Valid

When RADE is set to "1", IDMA is invoked when an interrupt factor occurs, thereby performing a programmed data transfer. If RADE is set to "0", normal interrupt processing is performed, without invoking IDMA.
For details on IDMA, refer to "IDMA (Intelligent DMA)".
At initial reset, RADE is set to "0" (interrupt request).

DEADE: A/D converter IDMA enable (D2) / Serial I/F Ch.1, A/D, port input 4–7 IDMA enable register (0x40297)

Enables IDMA transfer by means of an interrupt factor.

When using the set-only method (default)

Write "1": IDMA enabled
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA enabled
Write "0": IDMA disabled
Read: Valid

If DEADE is set to "1", the IDMA request by the interrupt factor is enabled. If this bit is set to "0", the IDMA request is disabled.
After an initial reset, DEADE is set to "0" (IDMA disabled).

Programming Notes

(1) Before setting the conversion mode, start/end channels, etc. for the A/D converter, be sure to disable the A/D converter (ADE (D2) / A/D enable register (0x40244) = "0"). A change in settings while the A/D converter is enabled could cause it to operate erratically.

(2) The A/D converter operates only when the prescaler is operating.
When the A/D converter registers are set up, the prescaler must be operating. Therefore, start the prescaler first and make sure the A/D converter is supplied with its operating clock before setting up the A/D converter registers.
In consideration of the conversion accuracy, we recommend that the A/D converter operating clock be 2 MHz (max.).

(3) Do not start an A/D conversion when the clock supplied from the prescaler to the A/D converter is turned off, and do not turn off the prescaler's clock output when an A/D conversion is underway, as doing so could cause the A/D converter to operate erratically.

(4) After an initial reset, the interrupt factor flag (FADE) becomes indeterminate. To prevent generation of an unwanted interrupt or IDMA request, be sure to reset this flag and register in a program.

(5) To prevent the regeneration of interrupts due to the same factor following the occurrence an interrupt, always be sure to reset the interrupt factor flag before setting the PSR again or executing the reti instruction.

THIS PAGE IS BLANK.

E0C33 Family ASIC Macro Manual

V DMA BLOCK

Table of Contents

V-1 INTRODUCTION

The DMA block is configured with two types of DMA controllers: HSDMA (High-Speed DMA) that has on-chip registers for controlling DMA command information and IDMA (Intelligent DMA) that uses a memory area for storing DMA command information.

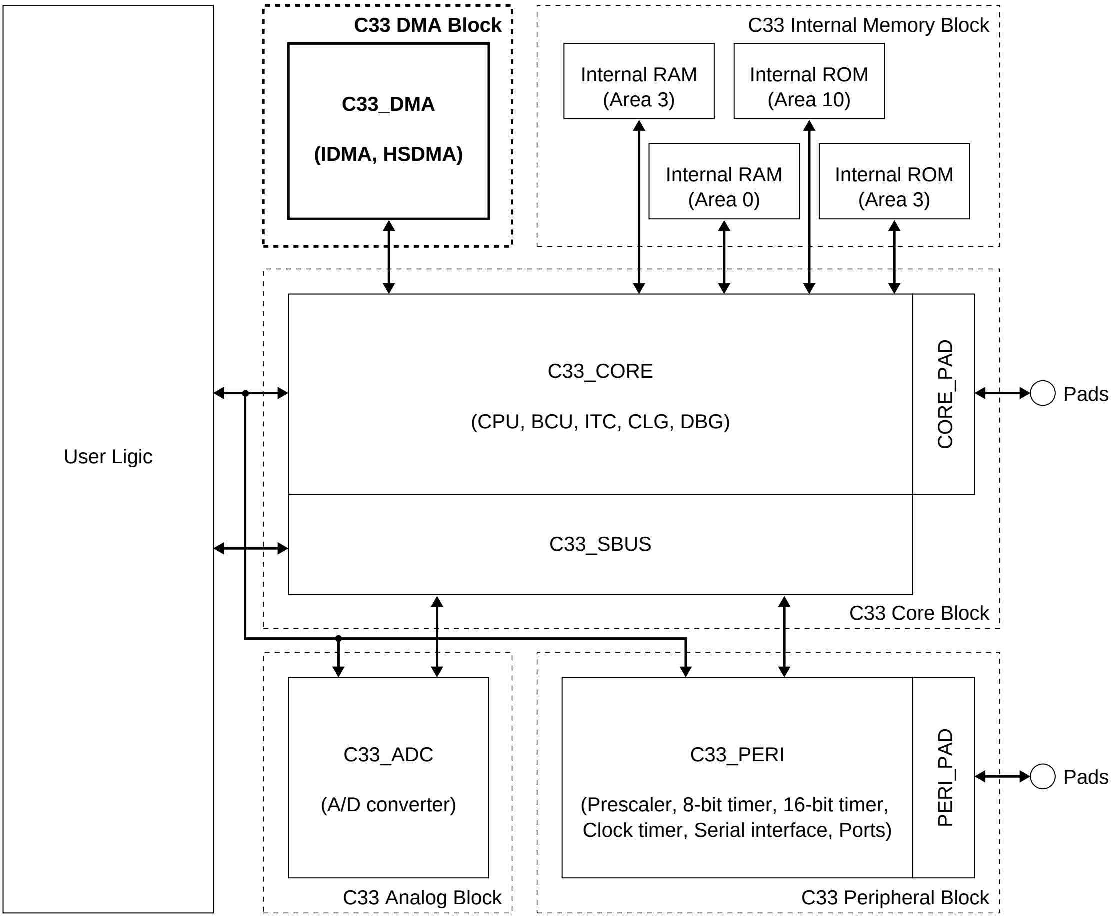


Figure 1.1 DMA Block

THIS PAGE IS BLANK.

V-2 HSDMA (High-Speed DMA)

Functional Outline of HSDMA

The DMA Block contains four channels of HSDMA (High-Speed DMA) circuits that support dual-address transfer and single-address transfer methods.
Since the control registers required for the DMA function are built into the chip, DMA requests for data transfer can be responded to instantaneously.

Dual-address transfer

In this method, a source address and a destination address for DMA transfer can be specified and a DMA transfer is performed in two phases. The first phase reads data at the source address into the on-chip temporary register. The second phase writes the temporary register data to the destination address.
Unlike IDMA (Intelligent DMA), which has transfer information in memory, this DMA method does not support a DMA link function but allows high-speed data transfers because it is not necessary to read transfer information from a memory.

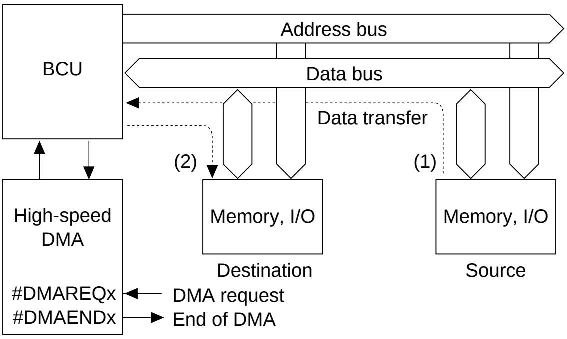


Figure 2.1 Dual-Address Transfer Method

Single-address transfer

In this method, data transfers that are normally accomplished by executing data read and write operations back-to-back are executed on the external bus collectively at one time, thus further speeding up the transfer operation. The #DMAACKx and #DMAENDx signals are used to control data transfer.
Unlike dual-address transfer, this method does not allow memory to memory data transfer but data transfers can be performed in minimum cycles.

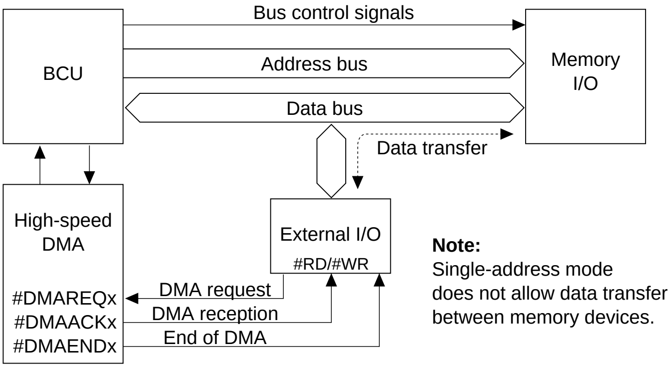


Figure 2.2 Single-Address Transfet Method

Note: Channels 0 to 3 are configured in the same way and have the same functionality. Signal and control bit names are assigned channel numbers 0 to 3 to distinguish them from other channels. In this manual, however, channel numbers 0 to 3 are designated with an "x" except where they must be distinguished, as the explanation is the same for all channels.

I/O Pins of HSDMA

Table 2.1 lists the I/O pins used for HSDMA.

Table 2.1 I/O Pins of HSDMA

Pin name	I/O	Function	Function select bit
K50/#DMAREQ0	I	Input port / High-speed DMA request 0	CFK50(D0)/K5 function select register(0x402C0)
K51/#DMAREQ1	I	Input port / High-speed DMA request 1	CFK51(D1)/K5 function select register(0x402C0)
K53/#DMAREQ2	I	Input port / High-speed DMA request 2	CFK53(D3)/K5 function select register(0x402C0)
K54/#DMAREQ3	I	Input port / High-speed DMA request 3	CFK54(D4)/K5 function select register(0x402C0)
P04/SIN1/ #DMAACK2	I/O	I/O port / Serial IF Ch.1 data input / #DMAACK2 output (Ex)	CFEX4(D4)/Port function extension register(0x402DF)
P05/SOUT1/ #DMAEND2	I/O	I/O port / Serial IF Ch.1 data output / #DMAEND2 output (Ex)	CFEX5(D5)/Port function extension register(0x402DF)
P06/#SCLK1/ #DMAACK3	I/O	I/O port / Serial IF Ch.1 clock input/output / #DMAACK3 output (Ex)	CFEX6(D6)/Port function extension register(0x402DF)
P07/#SRDY1/ #DMAEND3	I/O	I/O port / Serial IF Ch.1 ready input/output / #DMAEND3 output (Ex)	CFEX7(D7)/Port function extension register(0x402DF)
P15/EXCL4/ #DMAEND0	I/O	I/O port / 16-bit timer 4 event counter input (I) / High-speed DMA Ch.0 end signal output (O)	CFP15(D5)/P1 function select register(0x402D4)
P16/EXCL5/ #DMAEND1	I/O	I/O port / 16-bit timer 5 event counter input (I) / High-speed DMA Ch.1 end signal output (O)	CFP16(D6)/P1 function select register(0x402D4)
P32/#DMAACK0	I/O	I/O port / #DMAACK0 output	CFP32(D2)/P3 function select register(0x402DC)
P33/#DMAACK1	I/O	I/O port / #DMAACK1 output	CFP33(D3)/P3 function select register(0x402DC)

(I): Input mode, (O): Output mode, (Ex): Extended function

#DMAREQx (DMA request input pin)

This pin is used to input a DMA request signal from an external peripheral circuit. One data transfer operation is performed by this trigger (either the rising edge or the falling edge of the signal can be selected). The #DMAREQ0 to #DMAREQ3 pins correspond to channel 0 to channel 3, respectively.

In addition to this external input, software trigger or an interrupt factor can be selected for the HSDMA trigger factor using the register in the interrupt controller.

#DMAACKx (DMA acknowledge signal output pin for single-address mode)

This signal is output to indicate that a DMA request has been acknowledged by the DMA controller.

In single-address mode, the I/O device that is the source or destination of transfer outputs data to the external bus or takes in data from the external data synchronously with this signal.

The #DMAACK0 to #DMAACK3 pins correspond to channel 0 to channel 3, respectively.

This signal is not output in dual-address mode.

#DMAENDx (End-of-transfer signal output pin)

This signal is output to indicate that the number of data transfer operations that is set in the control register have been completed. The #DMAEND0 to #DMAEND3 pins correspond to channel 0 to channel 3, respectively.

Method for setting HSDMA I/O pins

As shown in Table 2.1, the pins used for HSDMA are shared with input ports and I/O ports. At cold start, all of these are set as input and I/O port pins (function select register = "0"). According to the signals to be used, set the corresponding pin function select bit by writing "1". At hot start, the register retains the previous status before a reset.

The #DMAEND3, #DMAACK3, #DMAEND2 and #DMAACK2 outputs are the extended functions of the P04 to P07 ports. When using these signals, the extended function bit (CFEX[7:4]) must be set to "1".

In addition, setup of the #DMAEND0 pin or #DMAEND1 pin further requires setting the I/O port's I/O control bit IOC15 (D5) or IOC16 (D6) / P1 I/O control register (0x402D6) by writing "1" in order to direct the pin for output. If this pin is directed for input, it functions as a 16-bit programmable timer's event counter input and cannot be used to output the #DMAENDx signal. At cold start, this pin is set for input. At hot start, it retains the previous status.

Programming Control Information

The HSDMA operates according to the control information set in the registers.
Note that some control bits change their functions according to the address mode.

The following explains how to set the contents of control information. Before using HSDMA, make each the settings described below.

Setting the Registers in Dual-Address Mode

Make sure that the HSDMA channel is disabled (HSx_EN = "0") before setting the control information.

Address mode

The address mode select bit DUALMx should be set to "1" (dual-address mode). This bit is set to "0" (single-address mode) at initial reset.

DUALM0: Ch. 0 address mode selection (DF) / HSDMA Ch. 0 control register (0x48222)
DUALM1: Ch. 1 address mode selection (DF) / HSDMA Ch. 1 control register (0x48232)
DUALM2: Ch. 2 address mode selection (DF) / HSDMA Ch. 2 control register (0x48242)
DUALM3: Ch. 3 address mode selection (DF) / HSDMA Ch. 3 control register (0x48252)

Transfer mode

A transfer mode should be set using the DxMOD[1:0] bits.

D0MOD[1:0]: Ch. 0 transfer mode (D[F:E]) / HSDMA Ch. 0 high-order destination address set-up register (0x4822A)
D1MOD[1:0]: Ch. 1 transfer mode (D[F:E]) / HSDMA Ch. 1 high-order destination address set-up register (0x4823A)
D2MOD[1:0]: Ch. 2 transfer mode (D[F:E]) / HSDMA Ch. 2 high-order destination address set-up register (0x4824A)
D3MOD[1:0]: Ch. 3 transfer mode (D[F:E]) / HSDMA Ch. 3 high-order destination address set-up register (0x4825A)

The following three transfer modes are available:

Single transfer mode (DxMOD = "00", default)

In this mode, a transfer operation invoked by one trigger is completed after transferring one unit of data of the size set by DATSIZEx. If data transfer need to be performed a number of times as set by the transfer counter, an equal number of triggers are required.

Successive transfer mode (DxMOD = "01")

In this mode, data transfer operations are performed by one trigger a number of times as set by the transfer counter. The transfer counter is decremented to 0 each time data is transferred.

Block transfer mode (DxMOD = "10")

In this mode, a transfer operation invoked by one trigger is completed after transferring one block of data of the size set by BLKLENx. If a block transfer need to be performed a number of times as set by the transfer counter, an equal number of triggers are required.

Transfer data size

The DATSIZEx bit is used to set the unit size of data to be transferred.
A half-word size (16 bits) is assumed if this bit is "1" and a byte size (8 bits) is assumed if this bit is "0" (default).

DATSIZE0: Ch. 0 transfer data size (DE) / HSDMA Ch. 0 high-order source address set-up register (0x48226)
DATSIZE1: Ch. 1 transfer data size (DE) / HSDMA Ch. 1 high-order source address set-up register (0x48236)
DATSIZE2: Ch. 2 transfer data size (DE) / HSDMA Ch. 2 high-order source address set-up register (0x48246)
DATSIZE3: Ch. 3 transfer data size (DE) / HSDMA Ch. 3 high-order source address set-up register (0x48256)

Block length

When using block transfer mode (DxMOD = "10"), the data block length (in units of DATSIZEx) should be set using the BLKLENx[7:0] bits.
BLKLEN0[7:0]: Ch. 0 block length (D[7:0]) / HSDMA Ch. 0 transfer counter register (0x48220)
BLKLEN1[7:0]: Ch. 1 block length (D[7:0]) / HSDMA Ch. 1 transfer counter register (0x48230)
BLKLEN2[7:0]: Ch. 2 block length (D[7:0]) / HSDMA Ch. 2 transfer counter register (0x48240)
BLKLEN3[7:0]: Ch. 3 block length (D[7:0]) / HSDMA Ch. 3 transfer counter register (0x48250)

Note: The block size thus set is decremented according to the transfers performed. If the block size is set to 0, it is decremented to all Fs by the first transfer performed. This means that you have set the maximum value that is determined by the number of bits available.

In single transfer and successive transfer modes, these bits are used as the bits7–0 of the transfer counter.

Transfer counter

Block transfer mode

In block transfer mode, up to 16 bits of transfer count can be specified.
TC0_L[7:0]: Ch. 0 transfer counter [7:0] (D[F:8]) / HSDMA Ch. 0 transfer counter register (0x48220)
TC1_L[7:0]: Ch. 1 transfer counter [7:0] (D[F:8]) / HSDMA Ch. 1 transfer counter register (0x48230)
TC2_L[7:0]: Ch. 2 transfer counter [7:0] (D[F:8]) / HSDMA Ch. 2 transfer counter register (0x48240)
TC3_L[7:0]: Ch. 3 transfer counter [7:0] (D[F:8]) / HSDMA Ch. 3 transfer counter register (0x48250)
TC0_H[7:0]: Ch. 0 transfer counter [15:8] (D[7:0]) / HSDMA Ch. 0 control register (0x48222)
TC1_H[7:0]: Ch. 1 transfer counter [15:8] (D[7:0]) / HSDMA Ch. 1 control register (0x48232)
TC2_H[7:0]: Ch. 2 transfer counter [15:8] (D[7:0]) / HSDMA Ch. 2 control register (0x48242)
TC3_H[7:0]: Ch. 3 transfer counter [15:8] (D[7:0]) / HSDMA Ch. 3 control register (0x48252)

Single transfer and successive transfer modes

In single transfer and successive transfer modes, up to 24 bits of transfer count can be specified.
BLKLEN0[7:0]: Ch. 0 transfer counter [7:0] (D[7:0]) / HSDMA Ch.0 transfer counter register (0x48220).
BLKLEN1[7:0]: Ch. 1 transfer counter [7:0] (D[7:0]) / HSDMA Ch.1 transfer counter register (0x48230).
BLKLEN2[7:0]: Ch. 2 transfer counter [7:0] (D[7:0]) / HSDMA Ch.2 transfer counter register (0x48240).
BLKLEN3[7:0]: Ch. 3 transfer counter [7:0] (D[7:0]) / HSDMA Ch.3 transfer counter register (0x48250).
TC0_L[7:0]: Ch. 0 transfer counter [15:8] (D[F:8]) / HSDMA Ch. 0 transfer counter register (0x48220)
TC1_L[7:0]: Ch. 1 transfer counter [15:8] (D[F:8]) / HSDMA Ch. 1 transfer counter register (0x48230)
TC2_L[7:0]: Ch. 2 transfer counter [15:8] (D[F:8]) / HSDMA Ch. 2 transfer counter register (0x48240)
TC3_L[7:0]: Ch. 3 transfer counter [15:8] (D[F:8]) / HSDMA Ch. 3 transfer counter register (0x48250)
TC0_H[7:0]: Ch. 0 transfer counter [23:16] (D[7:0]) / HSDMA Ch. 0 control register (0x48222)
TC1_H[7:0]: Ch. 1 transfer counter [23:16] (D[7:0]) / HSDMA Ch. 1 control register (0x48232)
TC2_H[7:0]: Ch. 2 transfer counter [23:16] (D[7:0]) / HSDMA Ch. 2 control register (0x48242)
TC3_H[7:0]: Ch. 3 transfer counter [23:16] (D[7:0]) / HSDMA Ch. 3 control register (0x48252)

Note: The transfer count thus set is decremented according to the transfers performed. If the transfer count is set to 0, it is decremented to all Fs by the first transfer performed. This means that you have set the maximum value that is determined by the number of bits available.

Source and destination addresses

In dual-address mode, a source address and a destination address for DMA transfer can be specified.
S0ADRL[15:0]: Ch. 0 source address [15:0] (D[F:0]) / Ch. 0 low-order source address set-up register (0x48224)
S1ADRL[15:0]: Ch. 1 source address [15:0] (D[F:0]) / Ch. 1 low-order source address set-up register (0x48234)
S2ADRL[15:0]: Ch. 2 source address [15:0] (D[F:0]) / Ch. 2 low-order source address set-up register (0x48244)
S3ADRL[15:0]: Ch. 3 source address [15:0] (D[F:0]) / Ch. 3 low-order source address set-up register (0x48254)
S0ADRH[11:0]: Ch. 0 source address [27:16] (D[B:0]) / Ch. 0 high-order source address set-up register (0x48226)
S1ADRH[11:0]: Ch. 1 source address [27:16] (D[B:0]) / Ch. 1 high-order source address set-up register (0x48236)
S2ADRH[11:0]: Ch. 2 source address [27:16] (D[B:0]) / Ch. 2 high-order source address set-up register (0x48246)
S3ADRH[11:0]: Ch. 3 source address [27:16] (D[B:0]) / Ch. 3 high-order source address set-up register (0x48256)

D0ADRL[15:0]: Ch. 0 destination address [15:0] (D[F:0]) / Ch. 0 low-order destination address set-up register (0x48228)
D1ADRL[15:0]: Ch. 1 destination address [15:0] (D[F:0]) / Ch. 1 low-order destination address set-up register (0x48238)
D2ADRL[15:0]: Ch. 2 destination address [15:0] (D[F:0]) / Ch. 2 low-order destination address set-up register (0x48248)
D3ADRL[15:0]: Ch. 3 destination address [15:0] (D[F:0]) / Ch. 3 low-order destination address set-up register (0x48258)
D0ADRH[11:0]: Ch. 0 destination address [27:16] (D[B:0]) / Ch. 0 high-order destination address set-up register (0x4822A)
D1ADRH[11:0]: Ch. 1 destination address [27:16] (D[B:0]) / Ch. 1 high-order destination address set-up register (0x4823A)
D2ADRH[11:0]: Ch. 2 destination address [27:16] (D[B:0]) / Ch. 2 high-order destination address set-up register (0x4824A)
D3ADRH[11:0]: Ch. 3 destination address [27:16] (D[B:0]) / Ch. 3 high-order destination address set-up register (0x4825A)

Address increment/decrement control

The source and/or destination addresses can be incremented or decremented when one data transfer is completed. The SxIN[1:0] bits (for source address) and DxIN[1:0] bits (for destination address) are used to set this function.

S0IN[1:0]: Ch. 0 source address control (D[D:C]) / Ch. 0 high-order source address set-up register (0x48226)
S1IN[1:0]: Ch. 1 source address control (D[D:C]) / Ch. 1 high-order source address set-up register (0x48236)
S2IN[1:0]: Ch. 2 source address control (D[D:C]) / Ch. 2 high-order source address set-up register (0x48246)
S3IN[1:0]: Ch. 3 source address control (D[D:C]) / Ch. 3 high-order source address set-up register (0x48256)

D0IN[1:0]: Ch. 0 destination address control (D[D:C]) / Ch. 0 high-order destination address set-up register (0x4822A)
D1IN[1:0]: Ch. 1 destination address control (D[D:C]) / Ch. 1 high-order destination address set-up register (0x4823A)
D2IN[1:0]: Ch. 2 destination address control (D[D:C]) / Ch. 2 high-order destination address set-up register (0x4824A)
D3IN[1:0]: Ch. 3 destination address control (D[D:C]) / Ch. 3 high-order destination address set-up register (0x4825A)

SxIN/DxIN = "00": address fixed (default)

The address is not changed by a data transfer performed. Even when transferring multiple data, the transfer data is always read/write from/to the same address.

SxIN/DxIN = "01": address decremented without initialization

The address is decremented by an amount equal to the data size set by DATSIZEx when one data transfer is completed. The address that has been decremented during transfer does not return to the initial value.

SxIN/DxIN = "10": address incremented with initialization

If this function is selected in single and successive transfer modes, the address is incremented by an amount equal to the data size set by DATSIZEx when one data transfer is completed. The address that has been incremented during transfer does not return to the initial value.
In block transfer mode too, the address is incremented when one data unit is transferred. However, the address that has been incremented during a block transfer recycles returns to the initial value when the block transfer is completed.

SxIN/DxIN = "11": address incremented without initialization

The address is incremented by an amount equal to the data size set by DATSIZEx when one data transfer is completed. The address that has been incremented during transfer does not return to the initial value.

Setting the Registers in Single-Address Mode

Make sure that the HSDMA channel is disabled (HSx_EN = "0") before seffing the control information.

Address mode

The address mode select bit DUALMx should be set to "0" (single-address mode). This bit is set to "0" at initial reset.

Transfer mode

A transfer mode should be set using the DxMOD[1:0] bits.
• Single transfer mode (DxMOD = "00", default)
• Successive transfer mode (DxMOD = "01")
• Block transfer mode (DxMOD = "10")

Refer to the explanation in "Setting the Registers in Dual-Address Mode".

Direction of transfer

The direction of data transfer should be set using DxDIR.
D0DIR: Ch. 0 transfer direction control (DE) / HSDMA Ch. 0 control register (0x48222)
D1DIR: Ch. 1 transfer direction control (DE) / HSDMA Ch. 1 control register (0x48232)
D2DIR: Ch. 2 transfer direction control (DE) / HSDMA Ch. 2 control register (0x48242)
D3DIR: Ch. 3 transfer direction control (DE) / HSDMA Ch. 3 control register (0x48252)

Memory write operations (data transfer from I/O device to memory) are specified by writing "1" and memory read operations (data transfer from memory to I/O device) are specified by writing "0".

Transfer data size

The DATSIZEx bit is used to set the unit size of data to be transferred.
A half-word size (16 bits) is assumed if this bit is "1" and a byte size (8 bits) is assumed if this bit is "0" (default).

Block length

When using block transfer mode (DxMOD = "10"), the data block length (in units of DATSIZEx) should be set using the BLKLENx[7:0] bits.
In single transfer and successive transfer modes, BLKLENx[7:0] is used as the bits7–0 of the transfer counter.

Transfer counter

Block transfer mode

In block transfer mode, up to 16 bits of transfer count can be specified using TCx_L[7:0] and TCx_H[7:0].

Single transfer and successive transfer modes

In single transfer and successive transfer modes, up to 24 bits of transfer count can be specified using BLKLENx[7:0], TCx_L[7:0] and TCx_H[7:0].

Memory address

In single-address mode, SxADRL[15:0] and SxADRH[11:0] are used to specify a memory address.
S0ADRL[15:0]: Ch. 0 memory address [15:0] (D[F:0]) / Ch. 0 low-order source address set-up register (0x48224)
S0ADRH[11:0]: Ch. 0 memory address [27:16] (D[B:0]) / Ch. 0 high-order source address set-up register (0x48226)
S1ADRL[15:0]: Ch. 1 memory address [15:0] (D[F:0]) / Ch. 1 low-order source address set-up register (0x48234)
S1ADRH[11:0]: Ch. 1 memory address [27:16] (D[B:0]) / Ch. 1 high-order source address set-up register (0x48236)
S2ADRL[15:0]: Ch. 2 memory address [15:0] (D[F:0]) / Ch. 2 low-order source address set-up register (0x48244)
S2ADRH[11:0]: Ch. 2 memory address [27:16] (D[B:0]) / Ch. 2 high-order source address set-up register (0x48246)
S3ADRL[15:0]: Ch. 3 memory address [15:0] (D[F:0]) / Ch. 3 low-order source address set-up register (0x48254)
S3ADRH[11:0]: Ch. 3 memory address [27:16] (D[B:0]) / Ch. 3 high-order source address set-up register (0x48256)

In single-address mode, data transfer is performed between the memory connected to the system interface and an external I/O device. The I/O device is accessed directly by the #DMAACKx signal, so it is unnecessary to specify an address. DxADRL[15:0] and DxADRH[11:0] are not used in single-address mode.

Address increment/decrement control

The memory addresses can be incremented or decremented when one data transfer is completed. SxIN[1:0] is used to set this function.

S0IN[1:0]: Ch. 0 memory address control (D[D:C]) / Ch. 0 high-order source address set-up register (0x48226)
S1IN[1:0]: Ch. 1 memory address control (D[D:C]) / Ch. 1 high-order source address set-up register (0x48236)
S2IN[1:0]: Ch. 2 memory address control (D[D:C]) / Ch. 2 high-order source address set-up register (0x48246)
S3IN[1:0]: Ch. 3 memory address control (D[D:C]) / Ch. 3 high-order source address set-up register (0x48256)

SxIN = "00": address fixed (default)
SxIN = "01": address decremented without initialization
SxIN = "10": address incremented with initialization
SxIN = "00": address incremented without initialization

Refer to the explanation in "Setting the Registers in Dual-Address Mode".

D0IN[1:0] is not used in single-address mode.

Enabling/Disabling DMA Transfer

The HSDMA transfer is enabled by writing "1" to the enable bit HSx_EN.
HS0_EN: Ch. 0 enable (D0) / Ch. 0 enable register (0x4822C)
HS1_EN: Ch. 1 enable (D0) / Ch. 1 enable register (0x4823C)
HS2_EN: Ch. 2 enable (D0) / Ch. 2 enable register (0x4824C)
HS3_EN: Ch. 3 enable (D0) / Ch. 3 enable register (0x4825C)

However, the control information must always be set correctly before enabling a DMA transfer.
Note that the control information cannot be set when HSx_EN = "1".
When HSx_EN is set to "0", HSDMA requests are no longer accepted.
When a DMA transfer is completed (transfer counter = 0), HSx_EN is reset to "0" to disable the following trigger inputs.

Trigger Factor

A HSDMA tigger factor can be selected from among 13 types using the HSDMA trigger set-up register for each channel. This function is supported by the interrupt controller.
HSD0S[3:0]: Ch. 0 trigger set-up (D[3:0]) / HSDMA Ch. 0/1 trigger set-up register (0x40298)
HSD1S[3:0]: Ch. 1 trigger set-up (D[7:4]) / HSDMA Ch. 0/1 trigger set-up register (0x40298)
HSD2S[3:0]: Ch. 2 trigger set-up (D[3:0]) / HSDMA Ch. 2/3 trigger set-up register (0x40299)
HSD3S[3:0]: Ch. 3 trigger set-up (D[7:4]) / HSDMA Ch. 2/3 trigger set-up register (0x40299)

Table 2.2 shows the setting value and the corresponding trigger factor.

Table 2.2 HSDMA Trigger Factor

Value	Ch.0 trigger factor	Ch.1 trigger factor	Ch.2 trigger factor	Ch.3 trigger factor
0000	Software trigger	Software trigger	Software trigger	Software trigger
0001	K50 port input (falling edge)	K51 port input (falling edge)	K53 port input (falling edge)	K54 port input (falling edge)
0010	K50 port input (rising edge)	K51 port input (rising edge)	K53 port input (rising edge)	K54 port input (rising edge)
0011	Port 0 input	Port 1 input	Port 2 input	Port 3 input
0100	Port 4 input	Port 5 input	Port 6 input	Port 7 input
0101	8-bit timer 0 underflow	8-bit timer 1 underflow	8-bit timer 2 underflow	8-bit timer 3 underflow
0110	16-bit timer 0 compare B	16-bit timer 1 compare B	16-bit timer 2 compare B	16-bit timer 3 compare B
0111	16-bit timer 0 compare A	16-bit timer 1 compare A	16-bit timer 2 compare A	16-bit timer 3 compare A
1000	16-bit timer 4 compare B	16-bit timer 5 compare B	16-bit timer 4 compare B	16-bit timer 5 compare B
1001	16-bit timer 4 compare A	16-bit timer 5 compare A	16-bit timer 4 compare A	16-bit timer 5 compare A
1010	Serial I/F Ch.0 Rx buffer full	Serial I/F Ch.1 Rx buffer full	Serial I/F Ch.0 Rx buffer full	Serial I/F Ch.1 Rx buffer full
1011	Serial I/F Ch.0 Tx buffer empty	Serial I/F Ch.1 Tx buffer empty	Serial I/F Ch.0 Tx buffer empty	Serial I/F Ch.1 Tx buffer empty
1100	A/D conversion completion	A/D conversion completion	A/D conversion completion	A/D conversion completion

By selecting an interrupt factor with the HSDMA trigger set-up register, the HSDMA channel is invoked when the selected interrupt factor occurs. The interrupt control bits (interrupt factor flag, interrupt enable register, IDMA request register, interrupt priority register) do not affect this invocation. The interrupt factor that invokes HSDMA sets the interrupt factor flag. and HSDMA does not reset the flag. Consequently, when the DMA transfer is completed (even if the transfer counter is not 0), an interrupt request to the CPU will be generated if the interrupt has been enabled. To generate an interrupt only when the transfer counter reaches 0, disable the interrupt by the interrupt factor that invokes HSDMA and use the HSDMA transfer completion interrupt .

When software trigger is selected, the HSDMA channel can be invoked by writing "1" to the HSTx bit.
HST0: Ch. 0 software trigger (D0) / HSDMA software trigger register (0x4029F)
HST1: Ch. 1 software trigger (D1) / HSDMA software trigger register (0x4029F)
HST2: Ch. 2 software trigger (D2) / HSDMA software trigger register (0x4029F)
HST3: Ch. 3 software trigger (D3) / HSDMA software trigger register (0x4029F)

When the selected trigger factor occurs, the trigger flag is set to "1" to invoke the HSDMA channel.
The HSDMA starts a DMA transfer if it has been enabled and the trigger flag is cleared by the hardware at the same time. This makes it possible to queue the HSDMA triggers that have been generated.
The trigger flag can be read and cleared using the HSx_TF bit.
HS0_TF: Ch. 0 trigger flag status/clear (D0) / Ch. 0 trigger flag register (0x4822E)
HS1_TF: Ch. 1 trigger flag status/clear (D0) / Ch. 1 trigger flag register (0x4823E)
HS2_TF: Ch. 2 trigger flag status/clear (D0) / Ch. 2 trigger flag register (0x4824E)
HS3_TF: Ch. 3 trigger flag status/clear (D0) / Ch. 3 trigger flag register (0x4825E)

By writing "1" to this bit, the set trigger flag can be cleared if the DMA transfer has not been started.
When this bit is read, "1" indicates that the flag is set and "0" indicates that the flag is cleared.

Operation of HSDMA

An HSDMA channel starts data transfer by the selected trigger factor.
Make sure that transfer conditions and a trigger factor are set and the HSDMA channel is enabled before starting a DMA transfer.

Operation in Dual-Address Mode

In dual-address mode, both the source and destination addresses are accessed according to the bus condition set by the BCU.
HSDMA has three transfer modes, in each of which data transfer operates differently. The following describes the operation of HSDMA in each transfer mode.

Single transfer mode

The channel for which DxMOD in control information is set to "00" operates in single transfer mode. In this mode, a transfer operation invoked by one trigger is completed after transferring one data unit of the size set by DATSIZEx. If a data transfer needs to be performed a number of times as set by the transfer counter, an equal number of triggers are required.
The operation of HSDMA in single transfer mode is shown by the flow chart in Figure 2.3.

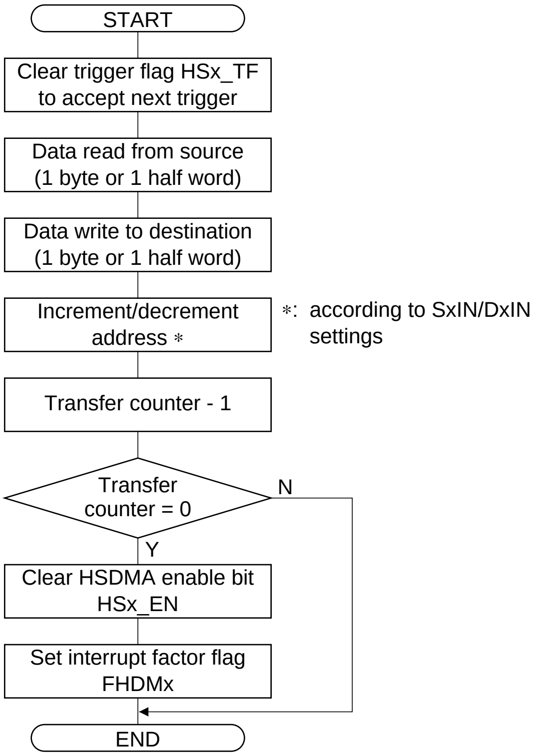


Figure 2.3 Operation Flow in Single Transfer Mode

(1) When a trigger is accepted, the trigger flag HSx_TF is cleared and then data of the size set in the control information is read from the source address.
(2) The read data is written to the destination address.
(3) The addresses are incremented or decremented according to the SxIN/DxIN settings.
(4) The transfer counter is decremented.
(5) The HSDMA enable bit HSx_EN is cleared and HSDMA interrupt factor flag in ITC is set when the transfer counter reaches 0 (when DINTENx = "1").

Successive transfer mode

The channel for which DxMOD in control information is set to "01" operates in successive transfer mode. In this mode, a data transfer is performed by one trigger a number of times as set by the transfer counter. The transfer counter is decremented to "0" by one transfer executed.

The operation of HSDMA in successive transfer mode is shown by the flow chart in Figure 2.4.

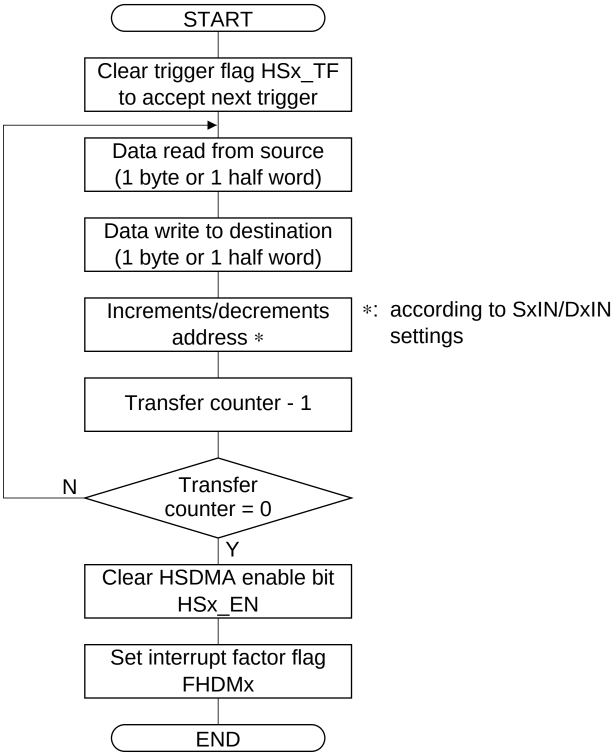


Figure 2.4 Operation Flow in Successive Transfer Mode

(1) When a trigger is accepted, the trigger flag HSx_TF is cleared and then data of the size set in the control information is read from the source address.
(2) The read data is written to the destination address.
(3) The addresses are incremented or decremented according to the SxIN/DxIN settings.
(4) The transfer counter is decremented.
(5) Steps (1) to (4) are repeated until the transfer counter reaches 0.
(6) The HSDMA enable bit HSx_EN is cleared and HSDMA interrupt factor flag in ITC is set when the transfer counter reaches 0 (when DINTENx = "1").

Block transfer mode

The channel for which DxMOD in control information is set to "10" operates in block transfer mode. In this mode, a transfer operation invoked by one trigger is completed after transferring one block of data of the size set by BLKLENx. If a block transfer needs to be performed a number of times as set by the transfer counter, an equal number of triggers are required.

The operation of HSDMA in block transfer mode is shown by the flow chart in Figure 2.5.

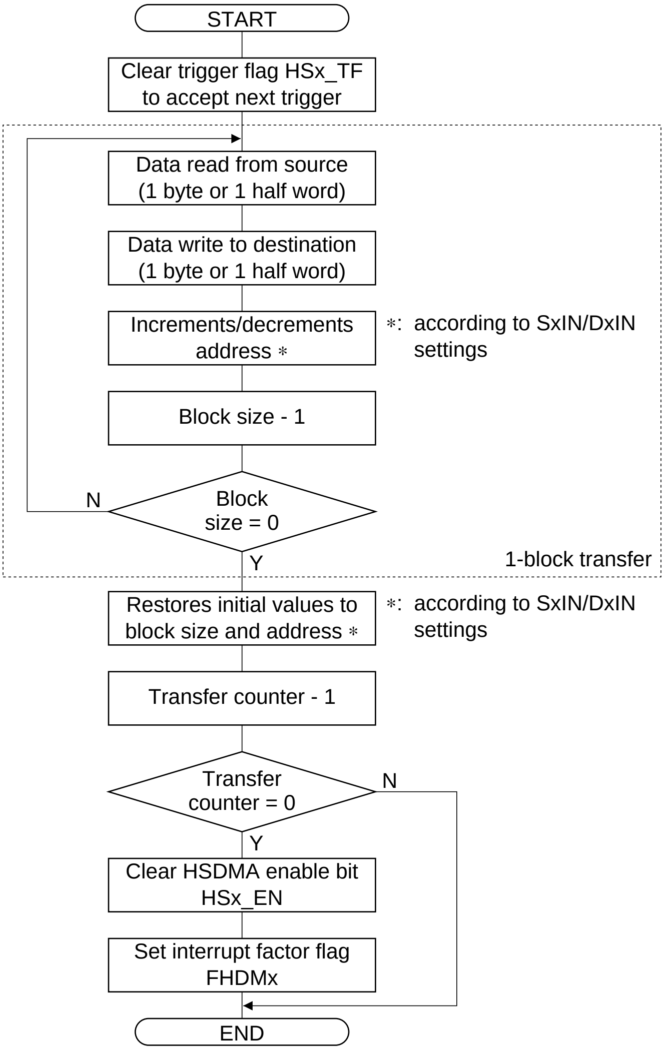


Figure 2.5 Operation Flow in Block Transfer Mode

(1) When a trigger is accepted, the trigger flag HSx_TF is cleared and then data of the size set in the control information is read from the source address.
(2) The read data is written to the destination address.
(3) The address is incremented or decremented and BLKLENx is decremented.
(4) Steps (1) to (3) are repeated until BLKLEN reaches 0.
(5) If SxIN or DxIN is "10", the address is recycled to the initial value.
(6) The transfer counter is decremented.
(7) The HSDMA enable bit HSx_EN is cleared and HSDMA interrupt factor flag in ITC is set when the transfer counter reaches 0 (when DINTENx = "1").

Operation in Single-Address Mode

The operation of each transfer mode is almost the same as that of dual-address mode (see the previous section). However, data read/write operation is performed simultaneously in single-address mode.
The following explains the data transfer operation different from dual-address mode.

#DMAACKx signal output and bus operation

When the HSDMA circuit accepts the DMA request, it outputs a low-level pulse from the #DMAACKx pin and starts bus operation for the memory at the same time.
The contents of this bus operation are as follows:

- **Data transfer from I/O device to memory**
  The address that has been set in the memory address register is output to the address bus.
  A write operation is performed under the interface conditions set on the area to which the memory at the destination of transfer belongs. The data bus is left floating.

  The external I/O device outputs the transfer data onto the data bus using the #DMAACKx signal as the read signal. The memory takes in this data using the write signal.

- **Data transfer from memory to an I/O device**
  The address that has been set in the memory address register is output to the address bus.
  A read operation is performed under the interface conditions set on the area to which the memory at the source of transfer belongs.

  The memory outputs the transfer data onto the data bus using the read signal.
  The external I/O device takes in the data from the data bus using the #DMAACKx signal as the write signal.

If the transfer data size is 16 bits and the I/O device is an 8-bit device, two bus operations are performed. Otherwise, transfer is completed in one bus operation.

#DMAENDx signal output

When the transfer counter reaches 0, the end-of-transfer signal is output from the #DMAENDx pin indicating that a specified number of transfers has been completed. At the same time, the interrupt factor for the completion of HSDMA is generated.

Timing Chart

Dual-address mode

(1) SRAM

Example: When 2 (RD)/1 (WR) wait cycles are inserted

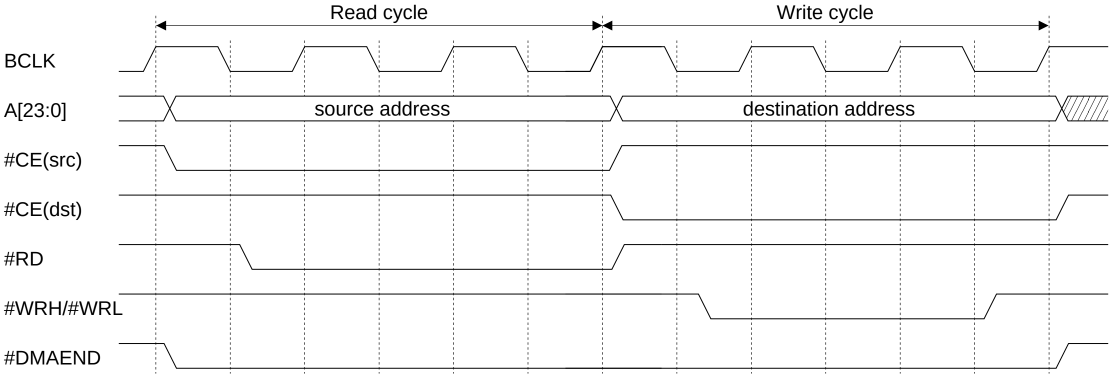


Figure 2.6 #DMAEND Signal Output Timing (SRAM)

(2) DRAM

Example: Page mode, RAS: 1 cycle; CAS: 2 cycles; Precharge: 1 cycle

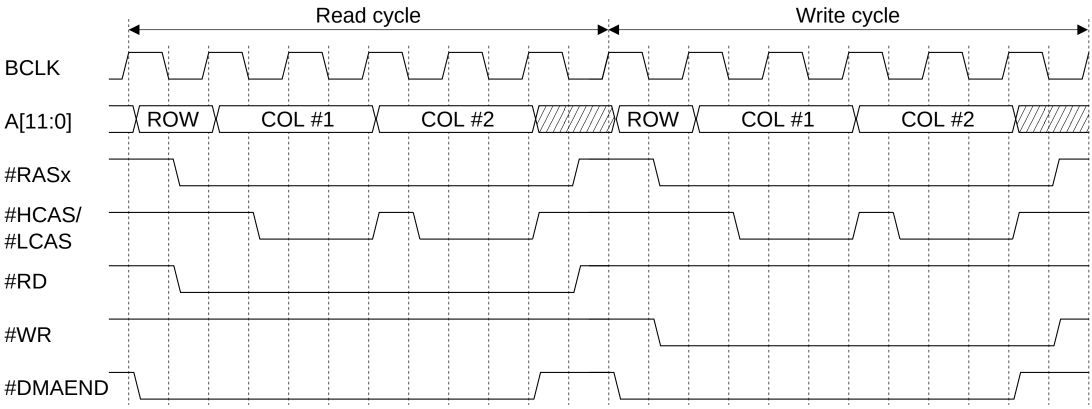


Figure 2.7 #DMAEND Signal Output Timing (DRAM)

Single-address mode

(1) SRAM

Example: When 2 (RD)/1 (WR) wait cycles are inserted

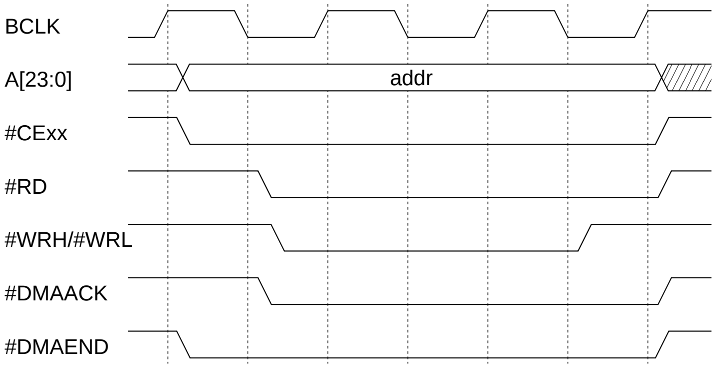


Figure 2.8 #DMAACK/#DMAEND Signal Output Timing (SRAM)

(2) Burst ROM

Example: When 4-consecutive-burst and 2-wait cycles are set during the first access

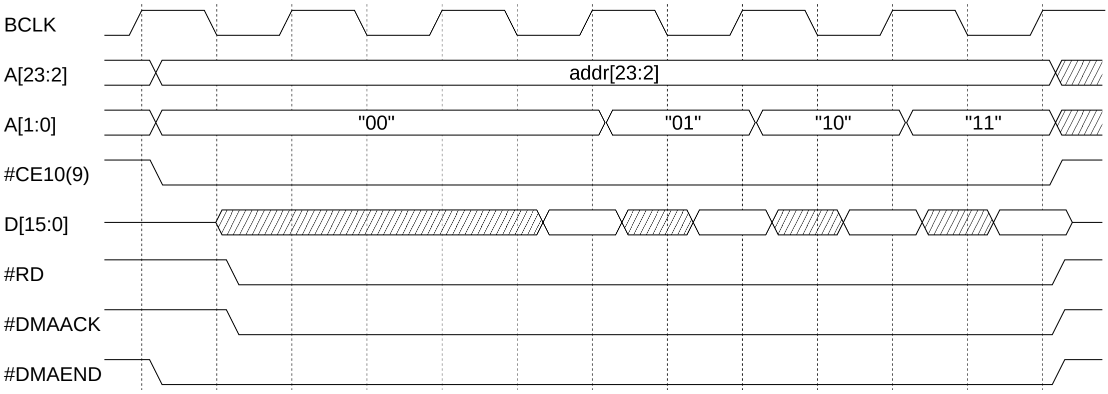


Figure 2.9 #DMAACK/#DMAEND Signal Output Timing (Burst ROM)

(3) DRAM

Example: Page mode, RAS: 1 cycle; CAS: 2 cycles; Precharge: 1 cycle

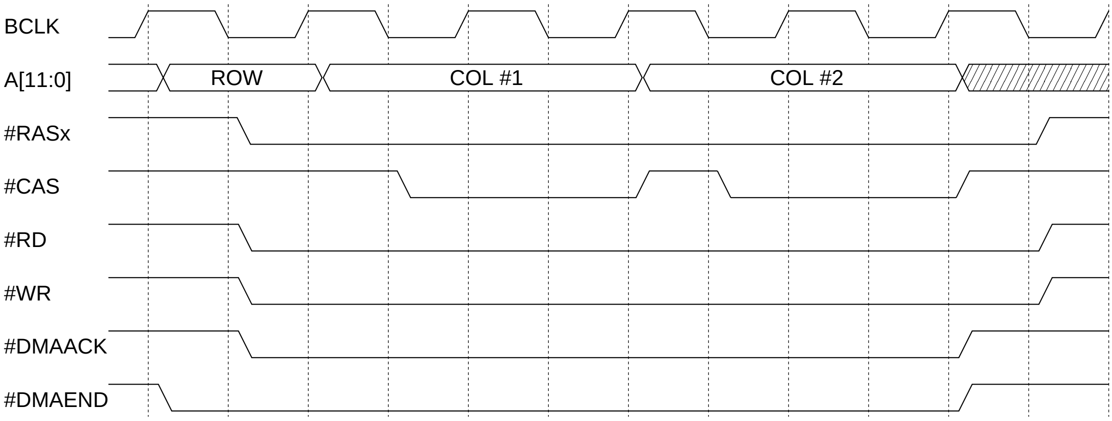


Figure 2.10 #DMAACK/#DMAEND Signal Output Timing (DRAM)

Interrupt Function of HSDMA

The DMA controller can generate an interrupt when the transfer counter in each HSDMA channel reaches 0. Furthermore, channels 0 and 1 can invoke IDMA using their interrupt factor.

Interrupt enable bit in the HSDMA circuit

The DINTENx bit in each channel allows selection whether to set the HSDMA interrupt factor flag in the interrupt controller or not. If DINTENx = "1", the HSDMA interrupt factor flag is set to "1" when the transfer counter reaches 0. If DINTENx = "0", the HSDMA interrupt factor flag is not set and an interrupt does not occur.

DINTEN0: Ch. 0 interrupt enable (DF) / HSDMA Ch. 0 high-order source address set-up register (0x48226)
DINTEN1: Ch. 1 interrupt enable (DF) / HSDMA Ch. 1 high-order source address set-up register (0x48236)
DINTEN2: Ch. 2 interrupt enable (DF) / HSDMA Ch. 2 high-order source address set-up register (0x48246)
DINTEN3: Ch. 3 interrupt enable (DF) / HSDMA Ch. 3 high-order source address set-up register (0x48256)

Control registers of the interrupt controller

Table 2.3 shows the control registers of the interrupt controller that are provided for each channel.

Table 2.3 Control Registers of Interrupt Controller

Channel	Interrupt factor flag	Interrupt enable register	Interrupt priority register
Ch. 0	FHDM0(D0/0x40281)	EHDM0(D0/0x40271)	PHSD0L[2:0](D[2:0]/0x40263)
Ch. 1	FHDM1(D1/0x40281)	EHDM1(D1/0x40271)	PHSD1L[2:0](D[6:4]/0x40263)
Ch. 2	FHDM2(D2/0x40281)	EHDM2(D2/0x40271)	PHSD2L[2:0](D[2:0]/0x40264)
Ch. 3	FHDM3(D3/0x40281)	EHDM3(D3/0x40271)	PHSD3L[2:0](D[6:4]/0x40264)

If DINTENx = "1", the HSDMA controller sets the HSDMA interrupt factor flag to "1" when the transfer counter reaches 0 after completing a series of HSDMA transfers. If the corresponding bit of the interrupt enable register is set to "1" at this time, an interrupt request is generated. Interrupts can be disabled by leaving the interrupt enable register bit set to "0". The HSDMA interrupt factor flag is always set to "1" when the data transfer in each channel is completed no matter what value the interrupt enable register bit is set to. (This is true even when it is set to "0".)

The interrupt priority register sets an interrupt priority level (0 to 7). An interrupt request to the CPU is accepted only when there is no other interrupt request of higher priority. Furthermore, it is only when the PSR's IE bit = "1" (interrupt enable) and the set value of IL is smaller than the HSDMA interrupt level which is set in the interrupt priority register that the CPU actually accepts a HSDMA interrupt. For details about the interrupt control register and for the device operation when an interrupt occurs, refer to "ITC (Interrupt Controller)".

Intelligent DMA

Intelligent DMA (IDMA) can be invoked by the end-of-transfer interrupt factor of channels 0 and 1 of HSDMA. The following shows the IDMA channels set in HSDMA:

	IDMA channel
Channel 0 end-of-transfer interrupt:	0x05
Channel 1 end-of-transfer interrupt:	0x06

Before IDMA can be invoked, the corresponding bits of the IDMA request and IDMA enable registers must be set to "1". Settings of transfer conditions on the IDMA side are also required.

Table 2.4 Control Bits for IDMA Transfer

Channel	IDMA request bit	IDMA enable bit
Ch. 0	RHDM0(D4/0x40290)	DEHDM0(D4/0x40294)
Ch. 1	RHDM1(D5/0x40290)	DEHDM1(D5/0x40294)

If the IDMA request and enable bits are set to "1", IDMA is invoked through generation of an interrupt factor. No interrupt request is generated at that point. An interrupt request is generated after the DMA transfer is completed. The registers can also be set so as not to generate an interrupt, with only a DMA transfer performed. For details on IDMA transfers and interrupt control upon completion of IDMA transfer, refer to "IDMA (Intelligent DMA)".

Trap vector

The trap vector addresses for interrupt factors in each channel are set by default as follows:

	(BTA3 = High)	(BTA3 = Low)
Channel 0 end-of-transfer interrupt:	0x0080058	0x0C00058
Channel 1 end-of-transfer interrupt:	0x008005C	0x0C0005C
Channel 2 end-of-transfer interrupt:	0x0080060	0x0C00060
Channel 3 end-of-transfer interrupt:	0x0080064	0x0C00064

Note that the trap table base address can be modified using the TTBR registers (0x48134 to 0x48137).

I/O Memory of HSDMA

Table 2.5 shows the control bits of HSDMA.

Table 2.5 Control Bits of HSDMA

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
High-speed DMA Ch.0/1 interrupt priority register	**0040263 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**PHSD1L2**	High-speed DMA Ch.1 interrupt level	0 to 7				X	R/W	
		D5	**PHSD1L1**						X		
		D4	**PHSD1L0**						X		
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**PHSD0L2**	High-speed DMA Ch.0 interrupt level	0 to 7				X	R/W	
		D1	**PHSD0L1**						X		
		D0	**PHSD0L0**						X		
High-speed DMA Ch.2/3 interrupt priority register	**0040264 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**PHSD3L2**	High-speed DMA Ch.3 interrupt level	0 to 7				X	R/W	
		D5	**PHSD3L1**						X		
		D4	**PHSD3L0**						X		
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**PHSD2L2**	High-speed DMA Ch.2 interrupt level	0 to 7				X	R/W	
		D1	**PHSD2L1**						X		
		D0	**PHSD2L0**						X		
DMA interrupt enable register	**0040271 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**EIDMA**	IDMA	1	Enabled	0	Disabled	0	R/W	
		D3	**EHDM3**	High-speed DMA Ch.3					0	R/W	
		D2	**EHDM2**	High-speed DMA Ch.2					0	R/W	
		D1	**EHDM1**	High-speed DMA Ch.1					0	R/W	
		D0	**EHDM0**	High-speed DMA Ch.0					0	R/W	
DMA interrupt factor flag register	**0040281 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**FIDMA**	IDMA	1	Factor is generated	0	No factor is generated	X	R/W	
		D3	**FHDM3**	High-speed DMA Ch.3					X	R/W	
		D2	**FHDM2**	High-speed DMA Ch.2					X	R/W	
		D1	**FHDM1**	High-speed DMA Ch.1					X	R/W	
		D0	**FHDM0**	High-speed DMA Ch.0					X	R/W	
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA request register	**0040290 (B)**	D7	**R16TC0**	16-bit timer 0 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**RHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**RHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**RP3**	Port input 3					0	R/W	
		D2	**RP2**	Port input 2					0	R/W	
		D1	**RP1**	Port input 1					0	R/W	
		D0	**RP0**	Port input 0					0	R/W	
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA enable register	**0040294 (B)**	D7	**DE16TC0**	16-bit timer 0 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**DEHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**DEHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**DEP3**	Port input 3					0	R/W	
		D2	**DEP2**	Port input 2					0	R/W	
		D1	**DEP1**	Port input 1					0	R/W	
		D0	**DEP0**	Port input 0					0	R/W	

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed DMA Ch.0/1 trigger set-up register	**0040298 (B)**	D7	**HSD1S3**	High-speed DMA Ch.1 trigger set-up	0 Software trigger	0	R/W	
		D6	**HSD1S2**		1 K51 input (falling edge)	0		
		D5	**HSD1S1**		2 K51 input (rising edge)	0		
		D4	**HSD1S0**		3 Port 1 input	0		
					4 Port 5 input			
					5 8-bit timer Ch.1 underflow			
					6 16-bit timer Ch.1 compare B			
					7 16-bit timer Ch.1 compare A			
					8 16-bit timer Ch.5 compare B			
					9 16-bit timer Ch.5 compare A			
					A SI/F Ch.1 Rx buffer full			
					B SI/F Ch.1 Tx buffer empty			
					C A/D conversion completion			
		D3	**HSD0S3**	High-speed DMA Ch.0 trigger set-up	0 Software trigger	0	R/W	
		D2	**HSD0S2**		1 K50 input (falling edge)	0		
		D1	**HSD0S1**		2 K50 input (rising edge)	0		
		D0	**HSD0S0**		3 Port 0 input	0		
					4 Port 4 input			
					5 8-bit timer Ch.0 underflow			
					6 16-bit timer Ch.0 compare B			
					7 16-bit timer Ch.0 compare A			
					8 16-bit timer Ch.4 compare B			
					9 16-bit timer Ch.4 compare A			
					A SI/F Ch.0 Rx buffer full			
					B SI/F Ch.0 Tx buffer empty			
					C A/D conversion completion			
High-speed DMA Ch.2/3 trigger set-up register	**0040299 (B)**	D7	**HSD3S3**	High-speed DMA Ch.3 trigger set-up	0 Software trigger	0	R/W	
		D6	**HSD3S2**		1 K54 input (falling edge)	0		
		D5	**HSD3S1**		2 K54 input (rising edge)	0		
		D4	**HSD3S0**		3 Port 3 input	0		
					4 Port 7 input			
					5 8-bit timer Ch.3 underflow			
					6 16-bit timer Ch.3 compare B			
					7 16-bit timer Ch.3 compare A			
					8 16-bit timer Ch.5 compare B			
					9 16-bit timer Ch.5 compare A			
					A SI/F Ch.1 Rx buffer full			
					B SI/F Ch.1 Tx buffer empty			
					C A/D conversion completion			
		D3	**HSD2S3**	High-speed DMA Ch.2 trigger set-up	0 Software trigger	0	R/W	
		D2	**HSD2S2**		1 K53 input (falling edge)	0		
		D1	**HSD2S1**		2 K53 input (rising edge)	0		
		D0	**HSD2S0**		3 Port 2 input	0		
					4 Port 6 input			
					5 8-bit timer Ch.2 underflow			
					6 16-bit timer Ch.2 compare B			
					7 16-bit timer Ch.2 compare A			
					8 16-bit timer Ch.4 compare B			
					9 16-bit timer Ch.4 compare A			
					A SI/F Ch.0 Rx buffer full			
					B SI/F Ch.0 Tx buffer empty			
					C A/D conversion completion			
High-speed DMA software trigger register	**004029A (B)**	D7–4	–	reserved	–	–	–	0 when being read.
		D3	**HST3**	HSDMA Ch.3 software trigger	1 Trigger / 0 Invalid	0	W	
		D2	**HST2**	HSDMA Ch.2 software trigger		0	W	
		D1	**HST1**	HSDMA Ch.1 software trigger		0	W	
		D0	**HST0**	HSDMA Ch.0 software trigger		0	W	
K5 function select register	**00402C0 (B)**	D7–5	–	reserved	–	–	–	0 when being read.
		D4	**CFK54**	K54 function selection	1 #DMAREQ3 / 0 K54	0	R/W	
		D3	**CFK53**	K53 function selection	1 #DMAREQ2 / 0 K53	0	R/W	
		D2	**CFK52**	K52 function selection	1 #ADTRG / 0 K52	0	R/W	
		D1	**CFK51**	K51 function selection	1 #DMAREQ1 / 0 K51	0	R/W	
		D0	**CFK50**	K50 function selection	1 #DMAREQ0 / 0 K50	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
P1 function select register	**00402D4 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**CFP16**	P16 function selection	1	EXCL5 #DMAEND1	0	P16	0	R/W	
		D5	**CFP15**	P15 function selection	1	EXCL4 #DMAEND0	0	P15	0	R/W	
		D4	**CFP14**	P14 function selection	1	FOSC1	0	P14	0	R/W	Extended functions (0x402DF)
		D3	**CFP13**	P13 function selection	1	EXCL3 T8UF3	0	P13	0	R/W	
		D2	**CFP12**	P12 function selection	1	EXCL2 T8UF2	0	P12	0	R/W	
		D1	**CFP11**	P11 function selection	1	EXCL1 T8UF1	0	P11	0	R/W	
		D0	**CFP10**	P10 function selection	1	EXCL0 T8UF0	0	P10	0	R/W	
P1 I/O control register	**00402D6 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**IOC16**	P16 I/O control	1	Output	0	Input	0	R/W	
		D5	**IOC15**	P15 I/O control					0	R/W	
		D4	**IOC14**	P14 I/O control					0	R/W	
		D3	**IOC13**	P13 I/O control					0	R/W	
		D2	**IOC12**	P12 I/O control					0	R/W	
		D1	**IOC11**	P11 I/O control					0	R/W	
		D0	**IOC10**	P10 I/O control					0	R/W	
P3 function select register	**00402DC (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**CFP35**	P35 function selection	1	#BUSACK	0	P35	0	R/W	
		D4	**CFP34**	P34 function selection	1	#BUSREQ #CE6	0	P34	0	R/W	
		D3	**CFP33**	P33 function selection	1	#DMAACK1	0	P33	0	R/W	
		D2	**CFP32**	P32 function selection	1	#DMAACK0	0	P32	0	R/W	
		D1	**CFP31**	P31 function selection	1	#BUSGET	0	P31	0	R/W	Ext. func.(0x402DF)
		D0	**CFP30**	P30 function selection	1	#WAIT #CE4/#CE5	0	P30	0	R/W	
Port function extension register	**00402DF (B)**	D7	**CFEX7**	P07 port extended function	1	#DMAEND3	0	P07, etc.	0	R/W	
		D6	**CFEX6**	P06 port extended function	1	#DMAACK3	0	P06, etc.	0	R/W	
		D5	**CFEX5**	P05 port extended function	1	#DMAEND2	0	P05, etc.	0	R/W	
		D4	**CFEX4**	P04 port extended function	1	#DMAACK2	0	P04, etc.	0	R/W	
		D3	**CFEX3**	P31 port extended function	1	#GARD	0	P31, etc.	0	R/W	
		D2	**CFEX2**	P21 port extended function	1	#GAAS	0	P21, etc.	0	R/W	
		D1	**CFEX1**	P10, P11, P13 port extended function	1	DST0 DST1 DPC0	0	P10, etc. P11, etc. P13, etc.	1	R/W	
		D0	**CFEX0**	P12, P14 port extended function	1	DST2 DCLK	0	P12, etc. P14, etc.	1	R/W	
High-speed DMA Ch.0 transfer counter register	**0048220 (HW)**	DF	**TC0_L7**	Ch.0 transfer counter[7:0]					X	R/W	
		DE	**TC0_L6**	(block transfer mode)					X		
		DD	**TC0_L5**						X		
		DC	**TC0_L4**	Ch.0 transfer counter[15:8]					X		
		DB	**TC0_L3**	(single/successive transfer mode)					X		
		DA	**TC0_L2**						X		
		D9	**TC0_L1**						X		
		D8	**TC0_L0**						X		
		D7	**BLKLEN07**	Ch.0 block length					X	R/W	
		D6	**BLKLEN06**	(block transfer mode)					X		
		D5	**BLKLEN05**						X		
		D4	**BLKLEN04**	Ch.0 transfer counter[7:0]					X		
		D3	**BLKLEN03**	(single/successive transfer mode)					X		
		D2	**BLKLEN02**						X		
		D1	**BLKLEN01**						X		
		D0	**BLKLEN00**						X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048222**	DF	**DUALM0**	Ch.0 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
DMA Ch.0	**(HW)**	DE	**D0DIR**	D) Invalid	–	–	–	
control register				S) Ch.0 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	**–**	reserved	–	–	–	Undefined in read.
Note:		D7	**TC0_H7**	Ch.0 transfer counter[15:8]		X	R/W	
D) Dual address mode		D6	**TC0_H6**	(block transfer mode)		X		
S) Single address mode		D5	**TC0_H5**			X		
		D4	**TC0_H4**	Ch.0 transfer counter[23:16]		X		
		D3	**TC0_H3**	(single/successive transfer mode)		X		
		D2	**TC0_H2**			X		
		D1	**TC0_H1**			X		
		D0	**TC0_H0**			X		
High-speed	**0048224**	DF	**S0ADRL15**	D) Ch.0 source address[15:0]		X	R/W	
DMA Ch.0	**(HW)**	DE	**S0ADRL14**	S) Ch.0 memory address[15:0]		X		
low-order		DD	**S0ADRL13**			X		
source address		DC	**S0ADRL12**			X		
set-up register		DB	**S0ADRL11**			X		
		DA	**S0ADRL10**			X		
Note:		D9	**S0ADRL9**			X		
D) Dual address mode		A8	**S0ADRL8**			X		
S) Single address mode		D7	**S0ADRL7**			X		
		D6	**S0ADRL6**			X		
		D5	**S0ADRL5**			X		
		D4	**S0ADRL4**			X		
		D3	**S0ADRL3**			X		
		D2	**S0ADRL2**			X		
		D1	**S0ADRL1**			X		
		D0	**S0ADRL0**			X		
High-speed	**0048226**	DF	**DINTEN0**	Ch.0 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
DMA Ch.0	**(HW)**	DE	**DATSIZE0**	Ch.0 transfer data size	1 Half word / 0 Byte	0	R/W	
high-order		DD	**S0IN1**	D) Ch.0 source address control	S0IN[1:0] : Inc/dec	0	R/W	
source address		DC	**S0IN0**	S) Ch.0 memory address control	1 1 : Inc.(no init)	0		
set-up register					1 0 : Inc.(init)			
					0 1 : Dec.(no init)			
Note:					0 0 : Fixed			
D) Dual address mode		DB	**S0ADRH11**	D) Ch.0 source address[27:16]		X	R/W	
S) Single address mode		DA	**S0ADRH10**	S) Ch.0 memory address[27:16]		X		
		D9	**S0ADRH9**			X		
		A8	**S0ADRH8**			X		
		D7	**S0ADRH7**			X		
		D6	**S0ADRH6**			X		
		D5	**S0ADRH5**			X		
		D4	**S0ADRH4**			X		
		D3	**S0ADRH3**			X		
		D2	**S0ADRH2**			X		
		D1	**S0ADRH1**			X		
		D0	**S0ADRH0**			X		
High-speed	**0048228**	DF	**D0ADRL15**	D) Ch.0 destination address[15:0]		X	R/W	
DMA Ch.0	**(HW)**	DE	**D0ADRL14**	S) Invalid		X		
low-order		DD	**D0ADRL13**			X		
destination		DC	**D0ADRL12**			X		
address set-up		DB	**D0ADRL11**			X		
register		DA	**D0ADRL10**			X		
		D9	**D0ADRL9**			X		
Note:		A8	**D0ADRL8**			X		
D) Dual address mode		D7	**D0ADRL7**			X		
S) Single address mode		D6	**D0ADRL6**			X		
		D5	**D0ADRL5**			X		
		D4	**D0ADRL4**			X		
		D3	**D0ADRL3**			X		
		D2	**D0ADRL2**			X		
		D1	**D0ADRL1**			X		
		D0	**D0ADRL0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed DMA Ch.0 high-order destination address set-up register	**004822A (HW)**	DF	**D0MOD1**	Ch.0 transfer mode	D0MOD[1:0] / Mode	0	R/W	
		DE	**D0MOD0**		1 1 Invalid	0		
					1 0 Block			
					0 1 Successive			
					0 0 Single			
		DD	**D0IN1**	D) Ch.0 destination address control	D0IN[1:0] / Inc/dec	0	R/W	
Note:		DC	**D0IN0**	S) Invalid	1 1 Inc.(no init)	0		
D) Dual address mode					1 0 Inc.(init)			
S) Single address mode					0 1 Dec.(no init)			
					0 0 Fixed			
		DB	**D0ADRH11**	D) Ch.0 destination address[27:16]		X	R/W	
		DA	**D0ADRH10**	S) Invalid		X		
		D9	**D0ADRH9**			X		
		A8	**D0ADRH8**			X		
		D7	**D0ADRH7**			X		
		D6	**D0ADRH6**			X		
		D5	**D0ADRH5**			X		
		D4	**D0ADRH4**			X		
		D3	**D0ADRH3**			X		
		D2	**D0ADRH2**			X		
		D1	**D0ADRH1**			X		
		D0	**D0ADRH0**			X		
High-speed DMA Ch.0 enable register	**004822C (HW)**	DF–1	–	reserved	–	–	–	Undefined in read.
		D0	**HS0_EN**	Ch.0 enable	1 Enable / 0 Disable	0	R/W	
High-speed DMA Ch.0 trigger flag register	**004822E (HW)**	DF–1	–	reserved	–	–	–	Undefined in read.
		D0	**HS0_TF**	Ch.0 trigger flag clear (writing)	1 Clear / 0 No operation	0	R/W	
				Ch.0 trigger flag status (reading)	1 Set / 0 Cleared			
High-speed DMA Ch.1 transfer counter register	**0048230 (HW)**	DF	**TC1_L7**	Ch.1 transfer counter[7:0] (block transfer mode)		X	R/W	
		DE	**TC1_L6**			X		
		DD	**TC1_L5**			X		
		DC	**TC1_L4**	Ch.1 transfer counter[15:8] (single/successive transfer mode)		X		
		DB	**TC1_L3**			X		
		DA	**TC1_L2**			X		
		D9	**TC1_L1**			X		
		D8	**TC1_L0**			X		
		D7	**BLKLEN17**	Ch.1 block length (block transfer mode)		X	R/W	
		D6	**BLKLEN16**			X		
		D5	**BLKLEN15**			X		
		D4	**BLKLEN14**	Ch.1 transfer counter[7:0] (single/successive transfer mode)		X		
		D3	**BLKLEN13**			X		
		D2	**BLKLEN12**			X		
		D1	**BLKLEN11**			X		
		D0	**BLKLEN10**			X		
High-speed DMA Ch.1 control register	**0048232 (HW)**	DF	**DUALM1**	Ch.1 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
		DE	**D1DIR**	D) Invalid	–	–	–	
				S) Ch.1 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	–	reserved	–	–	–	Undefined in read.
Note:		D7	**TC1_H7**	Ch.1 transfer counter[15:8] (block transfer mode)		X	R/W	
D) Dual address mode		D6	**TC1_H6**			X		
S) Single address mode		D5	**TC1_H5**			X		
		D4	**TC1_H4**	Ch.1 transfer counter[23:16] (single/successive transfer mode)		X		
		D3	**TC1_H3**			X		
		D2	**TC1_H2**			X		
		D1	**TC1_H1**			X		
		D0	**TC1_H0**			X		

High-speed	**0048234**	DF	**S1ADRL15**	D) Ch.1 source address[15:0]		X	R/W	
DMA Ch.1	**(HW)**	DE	**S1ADRL14**	S) Ch.1 memory address[15:0]		X		
low-order		DD	**S1ADRL13**			X		
source address		DC	**S1ADRL12**			X		
set-up register		DB	**S1ADRL11**			X		
		DA	**S1ADRL10**			X		
Note:		D9	**S1ADRL9**			X		
D) Dual address mode		A8	**S1ADRL8**			X		
S) Single address mode		D7	**S1ADRL7**			X		
		D6	**S1ADRL6**			X		
		D5	**S1ADRL5**			X		
		D4	**S1ADRL4**			X		
		D3	**S1ADRL3**			X		
		D2	**S1ADRL2**			X		
		D1	**S1ADRL1**			X		
		D0	**S1ADRL0**			X		
High-speed	**0048236**	DF	**DINTEN1**	Ch.1 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
DMA Ch.1	**(HW)**	DE	**DATSIZE1**	Ch.1 transfer data size	1 Half word / 0 Byte	0	R/W	
high-order		DD	**S1IN1**	D) Ch.1 source address control	S1IN[1:0] — Inc/dec	0	R/W	
source address		DC	**S1IN0**	S) Ch.1 memory address control	1 1 Inc.(no init)	0		
set-up register					1 0 Inc.(init)			
					0 1 Dec.(no init)			
Note:					0 0 Fixed			
D) Dual address mode		DB	**S1ADRH11**	D) Ch.1 source address[27:16]		X	R/W	
S) Single address mode		DA	**S1ADRH10**	S) Ch.1 memory address[27:16]		X		
		D9	**S1ADRH9**			X		
		A8	**S1ADRH8**			X		
		D7	**S1ADRH7**			X		
		D6	**S1ADRH6**			X		
		D5	**S1ADRH5**			X		
		D4	**S1ADRH4**			X		
		D3	**S1ADRH3**			X		
		D2	**S1ADRH2**			X		
		D1	**S1ADRH1**			X		
		D0	**S1ADRH0**			X		
High-speed	**0048238**	DF	**D1ADRL15**	D) Ch.1 destination address[15:0]		X	R/W	
DMA Ch.1	**(HW)**	DE	**D1ADRL14**	S) Invalid		X		
low-order		DD	**D1ADRL13**			X		
destination		DC	**D1ADRL12**			X		
address set-up		DB	**D1ADRL11**			X		
register		DA	**D1ADRL10**			X		
		D9	**D1ADRL9**			X		
Note:		A8	**D1ADRL8**			X		
D) Dual address mode		D7	**D1ADRL7**			X		
S) Single address mode		D6	**D1ADRL6**			X		
		D5	**D1ADRL5**			X		
		D4	**D1ADRL4**			X		
		D3	**D1ADRL3**			X		
		D2	**D1ADRL2**			X		
		D1	**D1ADRL1**			X		
		D0	**D1ADRL0**			X		
High-speed	**004823A**	DF	**D1MOD1**	Ch.1 transfer mode	D1MOD[1:0] — Mode	0	R/W	
DMA Ch.1	**(HW)**	DE	**D1MOD0**		1 1 Invalid	0		
high-order					1 0 Block			
destination					0 1 Successive			
address set-up					0 0 Single			
register		DD	**D1IN1**	D) Ch.1 destination address control	D1IN[1:0] — Inc/dec	0	R/W	
		DC	**D1IN0**	S) Invalid	1 1 Inc.(no init)	0		
Note:					1 0 Inc.(init)			
D) Dual address mode					0 1 Dec.(no init)			
S) Single address mode					0 0 Fixed			
		DB	**D1ADRH11**	D) Ch.1 destination address[27:16]		X	R/W	
		DA	**D1ADRH10**	S) Invalid		X		
		D9	**D1ADRH9**			X		
		A8	**D1ADRH8**			X		
		D7	**D1ADRH7**			X		
		D6	**D1ADRH6**			X		
		D5	**D1ADRH5**			X		
		D4	**D1ADRH4**			X		
		D3	**D1ADRH3**			X		
		D2	**D1ADRH2**			X		
		D1	**D1ADRH1**			X		
		D0	**D1ADRH0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed DMA Ch.1 enable register	**004823C (HW)**	DF–1	**–**	reserved	–	–	–	Undefined in read.
		D0	**HS1_EN**	Ch.1 enable	1 Enable 0 Disable	0	R/W	
High-speed DMA Ch.1 trigger flag register	**004823E (HW)**	DF–1	**–**	reserved	–	–	–	Undefined in read.
		D0	**HS1_TF**	Ch.1 trigger flag clear (writing)	1 Clear 0 No operation	0	R/W	
				Ch.1 trigger flag status (reading)	1 Set 0 Cleared			
High-speed DMA Ch.2 transfer counter register	**0048240 (HW)**	DF	**TC2_L7**	Ch.2 transfer counter[7:0]		X	R/W	
		DE	**TC2_L6**	(block transfer mode)		X		
		DD	**TC2_L5**			X		
		DC	**TC2_L4**	Ch.2 transfer counter[15:8]		X		
		DB	**TC2_L3**	(single/successive transfer mode)		X		
		DA	**TC2_L2**			X		
		D9	**TC2_L1**			X		
		D8	**TC2_L0**			X		
		D7	**BLKLEN27**	Ch.2 block length		X	R/W	
		D6	**BLKLEN26**	(block transfer mode)		X		
		D5	**BLKLEN25**			X		
		D4	**BLKLEN24**	Ch.2 transfer counter[7:0]		X		
		D3	**BLKLEN23**	(single/successive transfer mode)		X		
		D2	**BLKLEN22**			X		
		D1	**BLKLEN21**			X		
		D0	**BLKLEN20**			X		
High-speed DMA Ch.2 control register	**0048242 (HW)**	DF	**DUALM2**	Ch.2 address mode selection	1 Dual addr 0 Single addr	0	R/W	
		DE	**D2DIR**	D) Invalid	–	–	–	
				S) Ch.2 transfer direction control	1 Memory WR 0 Memory RD	0	R/W	
		DD–8	**–**	reserved	–	–	–	Undefined in read.
Note:		D7	**TC2_H7**	Ch.2 transfer counter[15:8]		X	R/W	
D) Dual address mode		D6	**TC2_H6**	(block transfer mode)		X		
S) Single address mode		D5	**TC2_H5**			X		
		D4	**TC2_H4**	Ch.2 transfer counter[23:16]		X		
		D3	**TC2_H3**	(single/successive transfer mode)		X		
		D2	**TC2_H2**			X		
		D1	**TC2_H1**			X		
		D0	**TC2_H0**			X		
High-speed DMA Ch.2 low-order source address set-up register	**0048244 (HW)**	DF	**S2ADRL15**	D) Ch.2 source address[15:0]		X	R/W	
		DE	**S2ADRL14**	S) Ch.2 memory address[15:0]		X		
		DD	**S2ADRL13**			X		
		DC	**S2ADRL12**			X		
		DB	**S2ADRL11**			X		
		DA	**S2ADRL10**			X		
Note:		D9	**S2ADRL9**			X		
D) Dual address mode		A8	**S2ADRL8**			X		
S) Single address mode		D7	**S2ADRL7**			X		
		D6	**S2ADRL6**			X		
		D5	**S2ADRL5**			X		
		D4	**S2ADRL4**			X		
		D3	**S2ADRL3**			X		
		D2	**S2ADRL2**			X		
		D1	**S2ADRL1**			X		
		D0	**S2ADRL0**			X		
High-speed DMA Ch.2 high-order source address set-up register	**0048246 (HW)**	DF	**DINTEN2**	Ch.2 interrupt enable	1 Enabled 0 Disabled	0	R/W	
		DE	**DATSIZE2**	Ch.2 transfer data size	1 Half word 0 Byte	0	R/W	
		DD	**S2IN1**	D) Ch.2 source address control	S2IN[1:0] Inc/dec	0	R/W	
		DC	**S2IN0**	S) Ch.2 memory address control	1 1 Inc.(no init)	0		
					1 0 Inc.(init)			
					0 1 Dec.(no init)			
					0 0 Fixed			
Note:		DB	**S2ADRH11**	D) Ch.2 source address[27:16]		X	R/W	
D) Dual address mode		DA	**S2ADRH10**	S) Ch.2 memory address[27:16]		X		
S) Single address mode		D9	**S2ADRH9**			X		
		A8	**S2ADRH8**			X		
		D7	**S2ADRH7**			X		
		D6	**S2ADRH6**			X		
		D5	**S2ADRH5**			X		
		D4	**S2ADRH4**			X		
		D3	**S2ADRH3**			X		
		D2	**S2ADRH2**			X		
		D1	**S2ADRH1**			X		
		D0	**S2ADRH0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048248**	DF	**D2ADRL15**	D) Ch.2 destination address[15:0]		X	R/W	
DMA Ch.2	**(HW)**	DE	**D2ADRL14**	S) Invalid		X		
low-order		DD	**D2ADRL13**			X		
destination		DC	**D2ADRL12**			X		
address set-up		DB	**D2ADRL11**			X		
register		DA	**D2ADRL10**			X		
		D9	**D2ADRL9**			X		
Note:		A8	**D2ADRL8**			X		
D) Dual address mode		D7	**D2ADRL7**			X		
S) Single address mode		D6	**D2ADRL6**			X		
		D5	**D2ADRL5**			X		
		D4	**D2ADRL4**			X		
		D3	**D2ADRL3**			X		
		D2	**D2ADRL2**			X		
		D1	**D2ADRL1**			X		
		D0	**D2ADRL0**			X		
High-speed	**004824A**	DF	**D2MOD1**	Ch.2 transfer mode	D2MOD[1:0] / Mode	0	R/W	
DMA Ch.2	**(HW)**	DE	**D2MOD0**		1 1 Invalid	0		
high-order					1 0 Block			
destination					0 1 Successive			
address set-up					0 0 Single			
register		DD	**D2IN1**	D) Ch.2 destination address control	D2IN[1:0] / Inc/dec	0	R/W	
		DC	**D2IN0**	S) Invalid	1 1 Inc.(no init)	0		
Note:					1 0 Inc.(init)			
D) Dual address mode					0 1 Dec.(no init)			
S) Single address mode					0 0 Fixed			
		DB	**D2ADRH11**	D) Ch.2 destination address[27:16]		X	R/W	
		DA	**D2ADRH10**	S) Invalid		X		
		D9	**D2ADRH9**			X		
		A8	**D2ADRH8**			X		
		D7	**D2ADRH7**			X		
		D6	**D2ADRH6**			X		
		D5	**D2ADRH5**			X		
		D4	**D2ADRH4**			X		
		D3	**D2ADRH3**			X		
		D2	**D2ADRH2**			X		
		D1	**D2ADRH1**			X		
		D0	**D2ADRH0**			X		
High-speed DMA Ch.2	**004824C**	DF–1	–	reserved	–	–	–	Undefined in read.
enable register	**(HW)**	D0	**HS2_EN**	Ch.2 enable	1 Enable / 0 Disable	0	R/W	
High-speed DMA Ch.2	**004824E**	DF–1	–	reserved	–	–	–	Undefined in read.
trigger flag	**(HW)**	D0	**HS2_TF**	Ch.2 trigger flag clear (writing)	1 Clear / 0 No operation	0	R/W	
register				Ch.2 trigger flag status (reading)	1 Set / 0 Cleared			
High-speed	**0048250**	DF	**TC3_L7**	Ch.3 transfer counter[7:0]		X	R/W	
DMA Ch.3	**(HW)**	DE	**TC3_L6**	(block transfer mode)		X		
transfer		DD	**TC3_L5**			X		
counter		DC	**TC3_L4**	Ch.3 transfer counter[15:8]		X		
register		DB	**TC3_L3**	(single/successive transfer mode)		X		
		DA	**TC3_L2**			X		
		D9	**TC3_L1**			X		
		D8	**TC3_L0**			X		
		D7	**BLKLEN37**	Ch.3 block length		X	R/W	
		D6	**BLKLEN36**	(block transfer mode)		X		
		D5	**BLKLEN35**			X		
		D4	**BLKLEN34**	Ch.3 transfer counter[7:0]		X		
		D3	**BLKLEN33**	(single/successive transfer mode)		X		
		D2	**BLKLEN32**			X		
		D1	**BLKLEN31**			X		
		D0	**BLKLEN30**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048252**	DF	**DUALM3**	Ch.3 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
DMA Ch.3	**(HW)**	DE	**D3DIR**	D) Invalid	–	–	–	
control register				S) Ch.3 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	**–**	reserved	–	–	–	Undefined in read.
Note:		D7	**TC3_H7**	Ch.3 transfer counter[15:8]		X	R/W	
D) Dual address mode		D6	**TC3_H6**	(block transfer mode)		X		
S) Single address mode		D5	**TC3_H5**			X		
		D4	**TC3_H4**	Ch.3 transfer counter[23:16]		X		
		D3	**TC3_H3**	(single/successive transfer mode)		X		
		D2	**TC3_H2**			X		
		D1	**TC3_H1**			X		
		D0	**TC3_H0**			X		
High-speed	**0048254**	DF	**S3ADRL15**	D) Ch.3 source address[15:0]		X	R/W	
DMA Ch.3	**(HW)**	DE	**S3ADRL14**	S) Ch.3 memory address[15:0]		X		
low-order		DD	**S3ADRL13**			X		
source address		DC	**S3ADRL12**			X		
set-up register		DB	**S3ADRL11**			X		
		DA	**S3ADRL10**			X		
Note:		D9	**S3ADRL9**			X		
D) Dual address mode		A8	**S3ADRL8**			X		
S) Single address mode		D7	**S3ADRL7**			X		
		D6	**S3ADRL6**			X		
		D5	**S3ADRL5**			X		
		D4	**S3ADRL4**			X		
		D3	**S3ADRL3**			X		
		D2	**S3ADRL2**			X		
		D1	**S3ADRL1**			X		
		D0	**S3ADRL0**			X		
High-speed	**0048256**	DF	**DINTEN3**	Ch.3 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
DMA Ch.3	**(HW)**	DE	**DATSIZE3**	Ch.3 transfer data size	1 Half word / 0 Byte	0	R/W	
high-order		DD	**S3IN1**	D) Ch.3 source address control	S3IN[1:0] — Inc/dec	0	R/W	
source address		DC	**S3IN0**	S) Ch.3 memory address control	1 1 — Inc.(no init)	0		
set-up register					1 0 — Inc.(init)			
					0 1 — Dec.(no init)			
Note:					0 0 — Fixed			
D) Dual address mode		DB	**S3ADRH11**	D) Ch.3 source address[27:16]		X	R/W	
S) Single address mode		DA	**S3ADRH10**	S) Ch.3 memory address[27:16]		X		
		D9	**S3ADRH9**			X		
		A8	**S3ADRH8**			X		
		D7	**S3ADRH7**			X		
		D6	**S3ADRH6**			X		
		D5	**S3ADRH5**			X		
		D4	**S3ADRH4**			X		
		D3	**S3ADRH3**			X		
		D2	**S3ADRH2**			X		
		D1	**S3ADRH1**			X		
		D0	**S3ADRH0**			X		
High-speed	**0048258**	DF	**D3ADRL15**	D) Ch.3 destination address[15:0]		X	R/W	
DMA Ch.3	**(HW)**	DE	**D3ADRL14**	S) Invalid		X		
low-order		DD	**D3ADRL13**			X		
destination		DC	**D3ADRL12**			X		
address set-up		DB	**D3ADRL11**			X		
register		DA	**D3ADRL10**			X		
		D9	**D3ADRL9**			X		
Note:		A8	**D3ADRL8**			X		
D) Dual address mode		D7	**D3ADRL7**			X		
S) Single address mode		D6	**D3ADRL6**			X		
		D5	**D3ADRL5**			X		
		D4	**D3ADRL4**			X		
		D3	**D3ADRL3**			X		
		D2	**D3ADRL2**			X		
		D1	**D3ADRL1**			X		
		D0	**D3ADRL0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed DMA Ch.3 high-order destination address set-up register	**004825A (HW)**	DF DE	**D3MOD1** **D3MOD0**	Ch.3 transfer mode	D3MOD[1:0] / Mode 1 1 Invalid 1 0 Block 0 1 Successive 0 0 Single	0 0	R/W	
Note: D) Dual address mode S) Single address mode		DD DC	**D3IN1** **D3IN0**	D) Ch.3 destination address control S) Invalid	D3IN[1:0] / Inc/dec 1 1 Inc.(no init) 1 0 Inc.(init) 0 1 Dec.(no init) 0 0 Fixed	0 0	R/W	
		DB DA D9 A8 D7 D6 D5 D4 D3 D2 D1 D0	**D3ADRH11** **D3ADRH10** **D3ADRH9** **D3ADRH8** **D3ADRH7** **D3ADRH6** **D3ADRH5** **D3ADRH4** **D3ADRH3** **D3ADRH2** **D3ADRH1** **D3ADRH0**	D) Ch.3 destination address[27:16] S) Invalid		X X X X X X X X X X X X	R/W	
High-speed DMA Ch.3 enable register	**004825C (HW)**	DF–1	**–**	reserved	–	–	–	Undefined in read.
		D0	**HS3_EN**	Ch.3 enable	1 Enable / 0 Disable	0	R/W	
High-speed DMA Ch.3 trigger flag register	**004825E (HW)**	DF–1	**–**	reserved	–	–	–	Undefined in read.
		D0	**HS3_TF**	Ch.3 trigger flag clear (writing) Ch.3 trigger flag status (reading)	1 Clear / 0 No operation 1 Set / 0 Cleared	0	R/W	

CFK51–CFK50: K5[1:0] pin function selection (D[1:0]) / K5 function select register (0x402C0)
CFK54–CFK53: K5[4:3] pin function selection (D[4:3]) / K5 function select register (0x402C0)

Set the #DMAREQx pin of HSDMA.

Write "1": #DMAREQx input
Write "0": Input port
Read: Valid

CFK50, CFK51, CFK53 and CFK54 are the function select bits for K50 (#DMAREQ0), K51 (#DMAREQ1), K53 (#DMAREQ2) and K54 (#DMAREQ3), respectively. When using the #DMAREQx signal, write "1" to CFK5x to set the K5x port for inputting the signal.
If this bit is set to "0", the pin is set for an input port.
At cold start, CFK5x is set to "0" (input port). At hot start, CFK5x retains the previous status before an initial reset.

CFP16–CFP15: P1[6:5] pin function selection (D[6:5]) / P1 function select register (0x402D4)

Set the #DMAENDx pin of HSDMA.

Write "1": #DMAENDx output
Write "0": I/O port
Read: Valid

When using the #DMAEND0 signal, set the P15 pin for the #DMAEND0 output pin by writing "1" to CFP15. Similarly, when using the #DMAEND1 signal, set the P16 pin for the #DMAEND1 output pin by writing "1" to CFP16. Furthermore, direct these pins for output by writing "1" to the corresponding I/O control register.
If CFP1x is set to "0", the pin is set for an I/O port.
At cold start, CFP1x is set to "0" (I/O port). At hot start, CFP1x retains the previous status before an initial reset.

IOC16–IOC15: P1[6:5] port I/O control (D[6:5]) / P1 I/O control register (0x402D6)

Direct the I/O port for input or output.

Write "1": Output mode
Write "0": Input mode
Read: Valid

To use the #DMAEND0 pin (channel 0), direct the pin for output by writing "1" to IOC15; to use the #DMAEND1 pin (channel 1), direct the pin for output by writing "1" to IOC16. If these pins are set for input, the P15 and P16 pins do not function as the #DMAENDx output pins even when CFP15 and CFP16 are set to "1".
At cold start, IOC1x is set to "0" (input mode). At hot start, IOC1x retains the previous state before an initial reset.

CFP33–CFP32: P3[3:2] pin function selection (D[3:2]) / P3 function select register (0x402DC)

Set the #DMAACKx pin of HSDMA.

Write "1": #DMAACKx output
Write "0": I/O port
Read: Valid

When using the #DMAACK0 signal, set the P32 pin for the #DMAACK0 output pin by writing "1" to CFP32. Similarly, when using the #DMAACK1 signal, set the P33 pin for the #DMAACK1 output pin by writing "1" to CFP33.
If CFP3x is set to "0", the pin is set for an I/O port.
At cold start, CFP3x is set to "0" (I/O port). At hot start, CFP3x retains the previous status before an initial reset.

CFEX7–CFEX4: P0[7:4] function extension (D[7:4]) / Port function extension register (0x402DF)

Set the #DMAACKx and #DMAENDx pins of HSDMA.

Write "1": HSDMA output pin
Write "0": I/O-port/serial interface I/O pin
Read: Valid

CFEX4, CFEX5, CFEX6 and CFEX7 are the function extention bits for P04 (#DMAACK2), P05 (#DMAEND2), P06 (#DMAACK3) and P07 (#DMAEND3), respectively. When using the HSDMA signal, write "1" to CFEXx to set the P0x port for outputting the signal.
When CFEXx is set to "0", the corresponding CFP bit becomes effective.
At cold start, these bits are set to "0" (I/O-port/serial interface I/O pin). At hot start, these bits retain the previous status before an initial reset.

HSD0S3–HSD0S0: Ch. 0 trigger set-up (D[3:0]) / HSDMA Ch. 0/1 trigger set-up register (0x40298)
HSD1S3–HSD1S0: Ch. 1 trigger set-up (D[7:4]) / HSDMA Ch. 0/1 trigger set-up register (0x40298)
HSD2S3–HSD2S0: Ch. 2 trigger set-up (D[3:0]) / HSDMA Ch. 2/3 trigger set-up register (0x40299)
HSD3S3–HSD3S0: Ch. 3 trigger set-up (D[7:4]) / HSDMA Ch. 2/3 trigger set-up register (0x40299)

Select a trigger factor for each HSDMA channel.

Table 2.6 HSDMA Trigger Factor

Value	Ch.0 trigger factor	Ch.1 trigger factor	Ch.2 trigger factor	Ch.3 trigger factor
0000	Software trigger	Software trigger	Software trigger	Software trigger
0001	K50 port input (falling edge)	K51 port input (falling edge)	K53 port input (falling edge)	K54 port input (falling edge)
0010	K50 port input (rising edge)	K51 port input (rising edge)	K53 port input (rising edge)	K54 port input (rising edge)
0011	Port 0 input	Port 1 input	Port 2 input	Port 3 input
0100	Port 4 input	Port 5 input	Port 6 input	Port 7 input
0101	8-bit timer 0 underflow	8-bit timer 1 underflow	8-bit timer 2 underflow	8-bit timer 3 underflow
0110	16-bit timer 0 compare B	16-bit timer 1 compare B	16-bit timer 2 compare B	16-bit timer 3 compare B
0111	16-bit timer 0 compare A	16-bit timer 1 compare A	16-bit timer 2 compare A	16-bit timer 3 compare A
1000	16-bit timer 4 compare B	16-bit timer 5 compare B	16-bit timer 4 compare B	16-bit timer 5 compare B
1001	16-bit timer 4 compare A	16-bit timer 5 compare A	16-bit timer 4 compare A	16-bit timer 5 compare A
1010	Serial I/F Ch.0 Rx buffer full	Serial I/F Ch.1 Rx buffer full	Serial I/F Ch.0 Rx buffer full	Serial I/F Ch.1 Rx buffer full
1011	Serial I/F Ch.0 Tx buffer empty	Serial I/F Ch.1 Tx buffer empty	Serial I/F Ch.0 Tx buffer empty	Serial I/F Ch.1 Tx buffer empty
1100	A/D conversion completion	A/D conversion completion	A/D conversion completion	A/D conversion completion

At initial reset, HSDxS is set to "0000" (software trigger).

HST0: Ch. 0 software trigger (D0) / HSDMA software trigger register (0x4029A)
HST1: Ch. 1 software trigger (D1) / HSDMA software trigger register (0x4029A)
HST2: Ch. 2 software trigger (D2) / HSDMA software trigger register (0x4029A)
HST3: Ch. 3 software trigger (D3) / HSDMA software trigger register (0x4029A)

Start a DMA transfer.

Write "1": Trigger
Write "0": Invalid
Read: Invalid

Writing "1" to HSTx generates a trigger pulse that starts a DMA transfer.
HSTx is effective only when software trigger is selected as the trigger factor of the HSDMA channel by the HSDxS bits.
At initial reset, HSTx is set to "0".

HS0_TF: Ch. 0 trigger flag clear/status (D0) / HSDMA Ch. 0 trigger flag register (0x4022E)
HS1_TF: Ch. 1 trigger flag clear/status (D0) / HSDMA Ch. 1 trigger flag register (0x4023E)
HS2_TF: Ch. 2 trigger flag clear/status (D0) / HSDMA Ch. 2 trigger flag register (0x4024E)
HS3_TF: Ch. 3 trigger flag clear/status (D0) / HSDMA Ch. 3 trigger flag register (0x4025E)

These bits are used to check and clear the trigger flag status.

Write "1": Trigger flag clear
Write "0": Invalid
Read "1": Trigger flag has been set
Read "0": Trigger flag has been cleared

The trigger flag is set when the trigger factor is input to the HSDMA channel and is cleared when the HSDMA channel starts a data transfer. By reading HSx_TF, the flag status can be checked. Writing "1" to HSx_TF clears the trigger flag if the DMA transfer has not been started.
At initial reset, HSx_TF is set to "0".

HS0_EN: Ch. 0 enable (D0) / HSDMA Ch. 0 enable register (0x4822C)
HS1_EN: Ch. 1 enable (D1) / HSDMA Ch. 1 enable register (0x4823C)
HS2_EN: Ch. 2 enable (D2) / HSDMA Ch. 2 enable register (0x4824C)
HS3_EN: Ch. 3 enable (D3) / HSDMA Ch. 3 enable register (0x4825C)

Enable a DMA transfer.

Write "1": Enabled
Write "0": Disabled
Read: Valid

DMA transfer is enabled by writing "1" to this bit.
HSDMA is placed in a state ready to accept a DMA request from the #DMAREQx pin or by the selected trigger factor.
DMA transfer is disabled by writing "0" to this bit.
When DMA transfers are completed (transfer counter = 0), HSx_EN is cleared by the hardware.
Be sure to disable DMA transfers (HSx_EN = "0") before setting the transfer condition.
At initial reset, HSx_EN is set to "0" (disabled).

DUALM0: Ch. 0 address mode selection (DF) / HSDMA Ch. 0 control register (0x48222)
DUALM1: Ch. 1 address mode selection (DF) / HSDMA Ch. 1 control register (0x48232)
DUALM2: Ch. 2 address mode selection (DF) / HSDMA Ch. 2 control register (0x48242)
DUALM3: Ch. 3 address mode selection (DF) / HSDMA Ch. 3 control register (0x48252)

Select an address mode.

Write "1": Dual-address mode
Write "0": Single-address mode
Read: Valid

When "1" is written to DUALMx, the HSDMA channel enters dual-address mode that allows specification of source and destination addresses. When "0" is written, the HSDMA channel enters single-address mode for high-speed data transfer between the external memory and an I/O device.
At initial reset, DUALMx is set to "0" (single-address mode).

D0DIR: Ch. 0 transfer direction control (DE) / HSDMA Ch.0 control register (0x48222)
D1DIR: Ch. 1 transfer direction control (DE) / HSDMA Ch.1 control register (0x48232)
D2DIR: Ch. 2 transfer direction control (DE) / HSDMA Ch.2 control register (0x48242)
D3DIR: Ch. 3 transfer direction control (DE) / HSDMA Ch.3 control register (0x48252)

Control the direction of data transfer in single-address mode.

Write "1": Memory write (I/O to memory)
Write "0": Memory read (memory to I/O)
Read: Valid

Data transfer from an external I/O device to external memory is performed by writing "1" to DxDIR. Data transfer from external memory to an external I/O is performed by writing "0".
At initial reset, DxDIR is set to "0" (memory to I/O).

This bit is effective only in single-address mode.

D0MOD1–D0MOD0: Ch. 0 transfer mode (D[F:E]) / Ch. 0 high-order destination address set-up register (0x4822A)
D1MOD1–D1MOD0: Ch. 1 transfer mode (D[F:E]) / Ch. 1 high-order destination address set-up register (0x4823A)
D2MOD1–D2MOD0: Ch. 2 transfer mode (D[F:E]) / Ch. 2 high-order destination address set-up register (0x4824A)
D3MOD1–D3MOD0: Ch. 3 transfer mode (D[F:E]) / Ch. 3 high-order destination address set-up register (0x4825A)

Select a transfer mode.

Table 2.7 Transfer Mode

DxMOD1	DxMOD0	Mode
1	1	Invalid
1	0	Block transfer mode
0	1	Successive transfer mode
0	0	Single transfer mode

In single transfer mode, a transfer operation invoked by one trigger is completed after transferring one unit of data of the size set by DATSIZEx.
In successive transfer mode, data transfer operations are performed by one trigger a number of times as set by the transfer counter.
In block transfer mode, a transfer operation invoked by one trigger is completed after transferring one block of data of the size set by BLKLENx.
At initial reset, DxMOD is set to "00" (single transfer mode).

DATSIZE0: Ch. 0 transfer data size (DE) / Ch. 0 high-order source address register (0x48226)
DATSIZE1: Ch. 1 transfer data size (DE) / Ch. 1 high-order source address register (0x48236)
DATSIZE2: Ch. 2 transfer data size (DE) / Ch. 2 high-order source address register (0x48246)
DATSIZE3: Ch. 3 transfer data size (DE) / Ch. 3 high-order source address register (0x48256)

Select the data size to be transferred.

Write "1": Half-word (16 bits)
Write "0": Byte (8 bits)
Read: Valid

The transfer data size is set to 16 bits by writing "1" to DATSIZEx and set to 8 bits by writing "0".
At initial reset, DATSIZEx is set to "0" (8 bits).

S0IN1–S0IN0: Ch. 0 source address control (D[D:C]) / Ch. 0 high-order source address set-up register (0x48226)
S1IN1–S1IN0: Ch. 1 source address control (D[D:C]) / Ch. 1 high-order source address set-up register (0x48236)
S2IN1–S2IN0: Ch. 2 source address control (D[D:C]) / Ch. 2 high-order source address set-up register (0x48246)
S3IN1–S3IN0: Ch. 3 source address control (D[D:C]) / Ch. 3 high-order source address set-up register (0x48256)

Control the incrementing or decrementing of the memory address.

Table 2.8 Address Control

SxIN1	SxIN0	Address control
1	1	Increment without initialization
1	0	Increment with initialization
0	1	Decrement without initialization
0	0	Fixed

In dual-address mode, this setting applies to the source address. In single-address mode, this setting applies to the external memory address.
When "address fixed" (00) is selected, the source address is not changed by a data transfer performed. Even when transferring multiple data, the transfer data is always read from the same address.
When "address increment" (11 or 10) is selected in single and successive transfer modes, the source address is incremented by an amount equal to the data size set by DATSIZEx when one data transfer is completed.
When "address decrement" (01) is selected, the source address is decremented in the same way.
In block transfer mode too, the source address is incremented or decremented when one data unit is transferred. However, if SxIN is set to "10", the source address that has been incremented during a block transfer recycles back to the initial value when the block transfer is completed.
At initial reset, SxIN is set to "00" (Fixed).

D0IN1–D0IN0: Ch. 0 destination address control (D[D:C]) / Ch. 0 high-order destination address set-up register (0x4822A)
D1IN1–D1IN0: Ch. 1 destination address control (D[D:C]) / Ch. 1 high-order destination address set-up register (0x4823A)
D2IN1–D2IN0: Ch. 2 destination address control (D[D:C]) / Ch. 2 high-order destination address set-up register (0x4824A)
D3IN1–D3IN0: Ch. 3 destination address control (D[D:C]) / Ch. 3 high-order destination address set-up register (0x4825A)

Control the incrementing or decrementing of the memory address.

Table 2.9 Address Control

DxIN1	DxIN0	Address control
1	1	Increment without initialization
1	0	Increment with initialization
0	1	Decrement without initialization
0	0	Fixed

In dual-address mode, this setting applies to the destination address. In single-address mode, these bits are not used.
When "address fixed" (00) is selected, the destination address is not changed by a data transfer performed. Even when transferring multiple data, the transfer data is always written to the same address.
When "address increment" (11 or 10) is selected in single and successive transfer modes, the destination address is incremented by an amount equal to the data size set by DATSIZEx when one data transfer is completed.
When "address decrement" (01) is selected, the destination address is decremented in the same way.
In block transfer mode too, the destination address is incremented or decremented when one data unit is transferred. However, if DxIN is set to "10", the destination address that has been incremented during a block transfer recycles back to the initial value when the block transfer is completed.
At initial reset, DxIN is set to "00" (Fixed).

BLKLEN07–BLKLEN00: Ch. 0 block length/transfer counter[7:0] (D[7:0]) / Ch. 0 transfer counter register (0x48220)
BLKLEN17–BLKLEN10: Ch. 1 block length/transfer counter[7:0] (D[7:0]) / Ch. 1 transfer counter register (0x48230)
BLKLEN27–BLKLEN20: Ch. 2 block length/transfer counter[7:0] (D[7:0]) / Ch. 2 transfer counter register (0x48240)
BLKLEN37–BLKLEN30: Ch. 3 block length/transfer counter[7:0] (D[7:0]) / Ch. 3 transfer counter register (0x48250)

In block transfer mode, these bits are used to specify a transfer block size. A transfer operation invoked by one trigger is completed after transferring one block of data of the size set by BLKLENx.
In single or successive transfer mode, these bits are used to specifythe 8 low-order bits of the transfer counter.
At initial reset, these bits are not initialized.

TC0_L7–TC0_L0: Ch. 0 transfer counter[7:0]/[15:8] (D[F:8]) / Ch. 0 transfer counter register (0x48220)
TC0_H7–TC0_H0: Ch. 0 transfer counter[15:8]/[23:16] (D[F:8]) / Ch. 0 control register (0x48222)
TC1_L7–TC1_L0: Ch. 1 transfer counter[7:0]/[15:8] (D[F:8]) / Ch. 1 transfer counter register (0x48230)
TC1_H7–TC1_H0: Ch. 1 transfer counter[15:8]/[23:16] (D[F:8]) / Ch. 1 control register (0x48232)
TC2_L7–TC2_L0: Ch. 2 transfer counter[7:0]/[15:8] (D[F:8]) / Ch. 2 transfer counter register (0x48240)
TC2_H7–TC2_H0: Ch. 2 transfer counter[15:8]/[23:16] (D[F:8]) / Ch. 2 control register (0x48242)
TC3_L7–TC3_L0: Ch. 3 transfer counter[7:0]/[15:8] (D[F:8]) / Ch. 3 transfer counter register (0x48250)
TC3_H7–TC3_H0: Ch. 3 transfer counter[15:8]/[23:16] (D[F:8]) / Ch. 3 control register (0x48252)

Set the data transfer count.
In block transfer mode, TCx_L[7:0] is bits[7:0] of the transfer counter, and TCx_H[7:0] is bits[15:8] of the transfer counter.
In single or successive transfer mode, TCx_L[7:0] is bits[15:8] of the transfer counter, and TCx_H[7:0] is bits[23:16] of the transfer counter. The 8 low-order bits are specified by BLKLENx[7:0].
This counter is decremented each time a DMA transfer in the corresponding channel is performed. When the counter reaches 0, an interrupt factor is generated. In single-address mode, the end-of-transfer signal is output from the #DMAENDx pin at the same time.
Even when the counter is 0, a DMA request is accepted and the counter is decremented to "0xFFFF" (or "0xFFFFFF").
Be sure to disable DMA transfers (HSx_EN = "0") before writing and reading to and from the counter.
At initial reset, these bits are not initialized.

S0ADRL15–S0ADRL0: Ch. 0 source address[15:0]
(D[F:0]) / Ch. 0 low-order source address set-up register (0x48224)
S0ADRH11–S0ADRH0: Ch. 0 source address[27:16]
(D[B:0]) / Ch. 0 high-order source address set-up register (0x48226)
S1ADRL15–S1ADRL0: Ch. 1 source address[15:0]
(D[F:0]) / Ch. 1 low-order source address set-up register (0x48234)
S1ADRH11–S1ADRH0: Ch. 1 source address[27:16]
(D[B:0]) / Ch. 1 high-order source address set-up register (0x48236)
S2ADRL15–S2ADRL0: Ch. 2 source address[15:0]
(D[F:0]) / Ch. 2 low-order source address set-up register (0x48244)
S2ADRH11–S2ADRH0: Ch. 2 source address[27:16]
(D[B:0]) / Ch. 2 high-order source address set-up register (0x48246)
S3ADRL15–S3ADRL0: Ch. 3 source address[15:0]
(D[F:0]) / Ch. 3 low-order source address set-up register (0x48254)
S3ADRH11–S3ADRH0: Ch. 3 source address[27:16]
(D[B:0]) / Ch. 3 high-order source address set-up register (0x48256)

In dual-address mode, these bits are used to specify a source address. In single-address mode, an external memory address at the destination or source of transfer is specified.
Use SxADRL to set the 16 low-order bits of the address and SxADRH to set the 12 high-order bits.
Be sure to disable DMA transfers (HSx_EN = "0") before writing or reading to and from these registers.
The address is incremented or decremented (as set by SxIN) according to the transfer data size each time a DMA transfer in the corresponding channel is performed.
At initial reset, these bits are not initialized.

D0ADRL15–D0ADRL0: Ch. 0 destination address[15:0]
(D[F:0]) / Ch. 0 low-order destination address set-up register (0x48228)
D0ADRH11–D0ADRH0: Ch. 0 destination address[27:16]
(D[B:0]) / Ch. 0 high-order destination address set-up register (0x4822A)
D1ADRL15–D1ADRL0: Ch. 1 destination address[15:0]
(D[F:0]) / Ch. 1 low-order destination address set-up register (0x48238)
D1ADRH11–D1ADRH0: Ch. 1 destination address[27:16]
(D[B:0]) / Ch. 1 high-order destination address set-up register (0x4823A)
D2ADRL15–D2ADRL0: Ch. 2 destination address[15:0]
(D[F:0]) / Ch. 2 low-order destination address set-up register (0x48248)
D2ADRH11–D2ADRH0: Ch. 2 destination address[27:16]
(D[B:0]) / Ch. 2 high-order destination address set-up register (0x4824A)
D3ADRL15–D3ADRL0: Ch. 3 destination address[15:0]
(D[F:0]) / Ch. 3 low-order destination address set-up register (0x48258)
D3ADRH11–D3ADRH0: Ch. 3 destination address[27:16]
(D[B:0]) / Ch. 3 high-order destination address set-up register (0x4825A)

In dual-address mode, these bits are used to specify a destination address. In single-address mode, these bits are not used.
Be sure to disable DMA transfers (HSx_EN = "0") before writing or reading to and from these registers.
The address is incremented or decremented (as set by DxIN) according to the transfer data size each time a DMA transfer in the corresponding channel is performed.
At initial reset, these bits are not initialized.

DINTEN0: Ch. 0 interrupt enable (DF) / Ch. 0 high-order source address register (0x48226)
DINTEN1: Ch. 1 interrupt enable (DF) / Ch. 1 high-order source address register (0x48236)
DINTEN2: Ch. 2 interrupt enable (DF) / Ch. 2 high-order source address register (0x48246)
DINTEN3: Ch. 3 interrupt enable (DF) / Ch. 3 high-order source address register (0x48256)

Enable to set the interrupt factor flag for each channel.

Write "1": Enabled
Write "0": Disabled
Read: Valid

If DINTENx = "1", the HSDMA interrupt factor flag is set to "1" when the transfer counter reaches 0. If DINTENx = "0", the HSDMA interrupt factor flag is not set and an interrupt does not occur.
At initial reset, DINTENx is set to "0" (Disabled).

PHSD0L2–PHSD0L0: Ch. 0 interrupt level (D[2:0]) / HSDMA Ch. 0/1 interrupt priority register (0x40263)
PHSD1L2–PHSD1L0: Ch. 1 interrupt level (D[6:4]) / HSDMA Ch. 0/1 interrupt priority register (0x40263)
PHSD2L2–PHSD2L0: Ch. 2 interrupt level (D[2:0]) / HSDMA Ch. 2/3 interrupt priority register (0x40264)
PHSD3L2–PHSD3L0: Ch. 3 interrupt level (D[6:4]) / HSDMA Ch. 2/3 interrupt priority register (0x40264)

Set the priority level of an end-of-DMA interrupt in the range of 0 to 7.
At initial reset, these registers become indeterminate.

EHDM0: Ch. 0 interrupt enable (D0) / DMA interrupt enable register (0x40271)
EHDM1: Ch. 1 interrupt enable (D1) / DMA interrupt enable register (0x40271)
EHDM2: Ch. 2 interrupt enable (D2) / DMA interrupt enable register (0x40271)
EHDM3: Ch. 3 interrupt enable (D3) / DMA interrupt enable register (0x40271)

Enable or disable interrupt generation to the CPU.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

EHDMx is the interrupt enable bit for HSDMA channel x. The interrupt is enabled when EHDMx is set to "1" and disabled when EHDMx is set to "0".
At initial reset, EHDMx is set to "0" (interrupt disabled).

FHDM0: Ch. 0 interrupt factor flag (D0) / DMA interrupt factor flag register (0x40281)
FHDM1: Ch. 1 interrupt factor flag (D1) / DMA interrupt factor flag register (0x40281)
FHDM2: Ch. 2 interrupt factor flag (D2) / DMA interrupt factor flag register (0x40281)
FHDM3: Ch. 3 interrupt factor flag (D3) / DMA interrupt factor flag register (0x40281)

Indicate the occurrence status of HSDMA interrupt factor.

When read
Read "1": Interrupt factor generated
Read "0": No interrupt factor generated

When written using the reset-only method (default)
Write "1": Factor flag is reset
Write "0": Invalid

When written using the read/write method
Write "1": Factor flag is set
Write "0": Factor flag is reset

FHDMx is the interrupt factor flag for HSDMA channel x. These flags are set to "1" when the transfer counter reaches 0. An interrupt to the CPU is generated if the following conditions are met at this time:
1. The corresponding interrupt enable register is set to "1".
2. No other interrupt request of higher priority is generated.
3. The IE bit of the PSR is set to "1" (interrupt enable).
4. The corresponding interrupt priority register is set to a level higher than the CPU's interrupt level (IL).

When using an interrupt factor to request IDMA, note that even when the above conditions are met, no interrupt request to the CPU is generated for the interrupt factor that has occurred. If interrupts are enabled at the setting of the IDMA side, an interrupt is generated under the above conditions after the data transfer by IDMA is completed.
The interrupt factor flag is always set to "1" when an interrupt factor occurs no matter how the interrupt enable and interrupt priority registers are set.
In order for the next interrupt to be accepted after interrupt generation, the interrupt factor flag must be reset and the PSR must be set up again (by setting the IL below the level indicated by the interrupt priority register and setting the IE bit to "1" or executing the reti instruction).
The interrupt factor flag can only be reset by a write instruction in the software application. If the PSR is again set up to accept interrupts (or the reti instruction is executed) without resetting the interrupt factor flag, the same interrupt may occur again. Note also that the value to be written to reset the flag is "1" when using the reset-only method (RSTONLY = "1") and "0" when using the read/write method (RSTONLY = "0"). Be careful not to confuse these two cases.
The FHDMx flag becomes indeterminate when initially reset, so be sure to reset the flag in the software application.

RHDM0: Ch.0 IDMA request (D4) / Port input 0–3, HSDMA, 16-bit timer 0 IDMA request register (0x40290)
RHDM1: Ch.1 IDMA request (D5) / Port input 0–3, HSDMA, 16-bit timer 0 IDMA request register (0x40290)

Specify whether IDMA need to be invoked when an interrupt factor occurs.

When using the set-only method (default)

Write "1": IDMA request
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA request
Write "0": Interrupt request
Read: Valid

RHDM0 and RHDM1 are the IDMA request bits for HSDMA channels 0 and 1, respectively. If the bit is set to "1", IDMA is invoked when an interrupt factor occurs, thus performing a programmed data transfer. If the register is set to "0", regular interrupt processing is performed without ever invoking IDMA.
For details on IDMA, refer to "IDMA (Intelligent DMA)".
At initial reset, RHDMx is set to "0" (interrupt request).

DEHDM0: Ch.0 IDMA enable (D4) / Port input 0–3, HSDMA, 16-bit timer 0 IDMA enable register (0x40294)
DEHDM1: Ch.1 IDMA enable (D5) / Port input 0–3, HSDMA, 16-bit timer 0 IDMA enable register (0x40294)

Enables IDMA transfer by means of an interrupt factor.

When using the set-only method (default)

Write "1": IDMA enabled
Write "0": Not changed
Read: Valid

When using the read/write method

Write "1": IDMA enabled
Write "0": IDMA disabled
Read: Valid

DEHDM0 and DEHDM1 are the IDMA enable bits for HSDMA channels 0 and 1, respectively. If DEHDMx is set to "1", the IDMA request by the interrupt factor is enabled. If the bit is set to "0", the IDMA request is disabled.
At initial reset, DEHDMx is set to "0" (IDMA disabled).

Programming Notes

(1) When setting the transfer conditions, always make sure the DMA controller is inactive (HSx_EN = "0").

(2) After an initial reset, the interrupt factor flag (FHDMx) becomes indeterminate. Always be sure to reset the flag to prevent interrupts or IDMA requests from being generated inadvertently.

(3) To prevent an interrupt from being generated repeatedly for the same factor, be sure to reset the interrupt factor flag before setting up the PSR again or executing the reti instruction.

(4) HSDMA is given higher priority over IDMA (intelligent DMA) and the CPU. However, since HSDMA and IDMA share the same circuit, HSDMA cannot gain the bus ownership while an IDMA transfer is under way. Requests for HSDMA invocation that have occurred during an IDMA transfer are kept pending until the IDMA transfer is completed.
A request for IDMA invocation or an interrupt request that has occurred during a HSDMA transfer are accepted after completion of the HSDMA transfer.

THIS PAGE IS BLANK.

V-3 IDMA (Intelligent DMA)

Functional Outline of IDMA

The DMA Block contains an intelligent DMA (IDMA), a function that allows control information to be programmed in RAM. Up to 128 channels can be programmed, including 31 channels that are invoked by an interrupt factor that occurs in some internal peripheral circuit.
Although an additional overhead for loading and storing control information in RAM may be incurred, this intelligent DMA supports such functions as successive transfers, block transfers, and linking to another IDMA.

IDMA is invoked by an interrupt factor that occurs in some internal peripheral circuit or a software trigger, thereby performing a data transfer according to the control information in RAM. When the transfer is completed, IDMA can generate an interrupt or invoke another IDMA according to link settings.

Programming Control Information

The intelligent DMA operates according to the control information prepared in RAM. The control information can be stored in either internal RAM or external RAM should the necessary area be allocated.
The control information is 3 words (12 bytes) per channel in size, and must be located at contiguous addresses beginning with the base address that is set in the software application as the starting address of channel 0.
Consequently, an area of 384 words (1,536 bytes) in RAM is required in order for all of 128 channels to be used.

The following explains how to set the base address and the contents of control information. Before using IDMA, make each the settings described below.

Setting the base address

Set the starting address of control information (starting address of channel 0) in the IDMA base address register.
16 low-order bits: DBASEL[15:0](D[15:0]) / IDMA base address low-order register (0x48200)
12 high-order bits: DBASEH[11:0](D[11:0]) / IDMA base address high-order register (0x48202)

When initially reset, the base address is set to 0x0C003A0.

Notes:
- The address you set in the IDMA base address register must always be a word (32-bit) boundary address.
- Be sure to disable DMA transfers (IDMAEN = "0") before setting the base address. Writing to the IDMA base address register is ignored when the DMA transfer is enabled (IDMAEN = "1"). When the register is read, the read data is indeterminate.

Control information

Write the control information for the IDMA channels used to RAM.
The addresses at which the control information of each channel is placed are determined by the base address and a channel number.

Starting address of channel = base address + (channel number × 12 [bytes])

Note: The control information must be written only when the channel to be set does not start a DMA transfer. If a DMA transfer starts when the control information is being written to the RAM, proper transfer cannot performed. Reading the control information can always be done.

The contents of control information (3 words) in each channel are shown in the table below.

Table 3.1 IDMA Control Information

Word	Bit	Name	Function
1st	D31	LNKEN	IDMA link enable "1" = Enabled, "0" = Disabled
	D30–24	LNKCHN[6:0]	IDMA link field
	D23–8	TC[15:0]	Transfer counter (block transfer mode) Transfer counter - high-order 16 bits (single or successive transfer mode)
	D7–0	BLKLEN[7:0]	Block size (block transfer mode) Transfer counter - low-order 8 bits (single or successive transfer mode)
2nd	D31	DINTEN	End-of-transfer interrupt enable "1" = Enabled, "0" = Disabled
	D30	DATSIZ	Data size control "1" = Half-word, "0" = Byte
	D29–28	SRINC[1:0]	Source address control SRINC1 SRINC0 Setting contents 1 1 Address incremented (In block transfer mode, the transfer address is updated without reset using the initial value.) 1 0 Address incremented (In block transfer mode, the transfer address is updated with the initial value.) 0 1 Address decremented (In block transfer mode, the transfer address is updated without reset using the initial value.) 0 0 Address fixed
	D27–0	SRADR[27:0]	Source address
3rd	D31–30	DMOD[1:0]	Transfer mode (Do not set to "11".) DMOD1 DMOD0 Setting contents 1 0 Block transfer mode 0 1 Successive transfer mode 0 0 Single transfer mode
	D29–28	DSINC[1:0]	Destination address control DSINC1 DSINC0 Setting contents 1 1 Address incremented (In block transfer mode, the transfer address is updated without reset using the initial value.) 1 0 Address incremented (In block transfer mode, the transfer address is updated with the initial value.) 0 1 Address decremented (In block transfer mode, the transfer address is updated without reset using the initial value.) 0 0 Address fixed
	D27–0	DSADR[27:0]	Destination address

LNKEN: IDMA link enable (D31/1st Word)

If this bit remains set (= "1"), the IDMA channel that is set in the IDMA link field is invoked after the completion of a DMA transfer in this channel. DMA transfers in multiple channels can be performed successively by merely triggering the first channel to be executed. There is no limit to the number of channels linked. Set this link in order of the IDMA channels you want to be executed.
If this bit is "0", IDMA is completed by merely executing a DMA transfer in this channel.

LNKCHN[6:0]: IDMA link field (D[30:24]/1st Word)

If you want IDMA to be linked, set the channel numbers (0 to 127) to be executed next.
The data in this field is valid only when LINKEN = "1".

TC[15:0]: Transfer counter (D[23:8]/1st Word)

In block transfer mode, a transfer count can be specified using up to 16 bits. Set this value here. In single transfer and successive transfer modes, a transfer count can be specified using up to 24 bits. Set a 16-bit high-order value here.

BLKLEN[7:0]: Block size/transfer counter (D[7:0]/1st Word)

In block transfer mode, set the size of a block that is transferred in one operation (in units of DATSIZ). In single transfer and successive transfer modes, set an 8-bit low-order value for the transfer count here.

Note: The transfer count and block size thus set are decremented according to the transfers performed. If the transfer count or block size is set to 0, it is decremented to all Fs by the first transfer performed. This means that you have set the maximum value that is determined by the number of bits available.

DINTEN: End-of-transfer interrupt enable (D31/2nd Word)

If this bit is left set (= "1"), when the transfer counter reaches 0, an interrupt request to the CPU is generated based on the interrupt factor flag by which IDMA has been invoked.
If this bit is "0", no interrupt request to the CPU is generated even when the transfer counter has reached 0.

DATSIZ: Data size control (D30/2nd Word)

Set the unit size of data to be transferred.
A half-word size (16 bits) is assumed if this bit is "1" and a byte size (8 bits) is assumed if this bit is "0".

SRINC[1:0]: Source address control (D[29:28]/2nd Word)

Set the source address updating format.
If the format is set for "address fixed" (00), the source address is not changed by a data transfer performed. Even when transferring multiple data, the transfer data is always read from the same address.
If the format is set for "address increment" (11 or 10) in single and successive transfer modes, the source address is incremented by an amount equal to the data size set by DATSIZ when one data transfer is completed. If the format is set for "address decrement" (01), the source address is decremented in the same way.
In block transfer mode too, the source address is incremented or decremented when one data unit is transferred. However, if the set format is "10", the source address that has been incremented during a block transfer recycles back to the initial value when the block transfer is completed.

SRADR[27:0]: Source address (D[27:0]/2nd Word)

Use these bits to set the starting address at the source of transfer. The content set here is updated according to the setting of SRINC.

DMOD[1:0]: Transfer mode (D[31:30]/3rd Word)

Use these bits to set the desired transfer mode.
The transfer modes are outlined below (to be detailed later):

- **Single transfer mode (00)**
  In this mode, a transfer operation invoked by one trigger is completed after transferring one unit of data of the size set by DATSIZ. If data transfer need to be performed a number of times as set by the transfer counter, an equal number of triggers are required.

- **Successive transfer mode (01)**
  In this mode, data transfer operations are performed by one trigger a number of times as set by the transfer counter. The transfer counter is decremented to 0 each time data is transferred.

- **Block transfer mode (10)**
  In this mode, a transfer operation invoked by one trigger is completed after transferring one block of data of the size set by BLKLEN. If a block transfer need to be performed a number of times as set by the transfer counter, an equal number of triggers are required.

DSINC[1:0]: Destination address control (D[29:28]/3rd Word)

Set the destination address update format.
If the format is set for "address fixed" (00), the destination address is not changed by the performance of a data transfer operation. Even when transferring multiple data, the transfer data is always written to the same address.
If the format is set for "address increment" (11 or 10) in single and successive transfer modes, the destination address is incremented by an amount equal to the data size set by DATSIZ when one data transfer is completed.
If the format is set for "address decrement" (01), the destination address is decremented in the same way.
In block transfer mode as well, the destination address is incremented or decremented when one data unit is transferred. However, if the set format is "10", the destination address that has been incremented during a block transfer recycles back to the initial value when the block transfer is completed.

DSADR[27:0]: Destination address (D[27:0]/3rd Word)

Use these bits to set the starting address at the destination of transfer. The content set here is updated according to the setting of DSINC.

Since the control information is placed in RAM, it can be rewritten. However, before rewriting the content of this information, make sure that no DMA transfer is generated in the channel whose information you are going to rewrite.

IDMA Invocation

The triggers by which IDMA is invoked have the following three causes:
1. Interrupt factor in an internal peripheral circuit
2. Trigger in the software application
3. Link setting

Enabling/disabling DMA transfer

The IDMA controller is enabled by writing "1" to the IDMA enable bit IDMAEN (D0) / IDMA enable register (0x48205), and is ready to accept the triggers described above. However, before enabling a DMA transfer, be sure to set the base address and the control information for the channel to be invoked correctly. If IDMAEN is set to "0", no IDMA invocation request is accepted.

IDMA invocation by an interrupt factor in internal peripheral circuits

Some internal peripheral circuits that have an interrupt generating function can invoke IDMA by an interrupt factor in that circuit. The IDMA channel numbers corresponding to such IDMA invocation are predetermined. The relationship between the interrupt factors that have this function and the IDMA channels is shown in Table 3.2.

Table 3.2 Interrupt Factors Used to Invoke IDMA

Peripheral circuit	Interrupt factor	IDMA Ch.	IDMA request bit	IDMA enable bit
Ports	Port input 0	1	RP0 (D0/0x40290)	DEP0 (D0/0x40294)
	Port input 1	2	RP1 (D1/0x40290)	DEP1 (D1/0x40294)
	Port input 2	3	RP2 (D2/0x40290)	DEP2 (D2/0x40294)
	Port input 3	4	RP3 (D3/0x40290)	DEP3 (D3/0x40294)
High-speed DMA	Ch.0, end of transfer	5	RHDM0 (D4/0x40290)	DEHDM0 (D4/0x40294)
	Ch.1, end of transfer	6	RHDM1 (D5/0x40290)	DEHDM1 (D5/0x40294)
16-bit programmable timer	Timer 0 comparison B	7	R16TU0 (D6/0x40290)	DE16TU0 (D6/0x40294)
	Timer 0 comparison A	8	R16TC0 (D7/0x40290)	DE16TC0 (D7/0x40294)
	Timer 1 comparison B	9	R16TU1 (D0/0x40291)	DE16TU1 (D0/0x40295)
	Timer 1 comparison A	10	R16TC1 (D1/0x40291)	DE16TC1 (D1/0x40295)
	Timer 2 comparison B	11	R16TU2 (D2/0x40291)	DE16TU2 (D2/0x40295)
	Timer 2 comparison A	12	R16TC2 (D3/0x40291)	DE16TC2 (D3/0x40295)
	Timer 3 comparison B	13	R16TU3 (D4/0x40291)	DE16TU3 (D4/0x40295)
	Timer 3 comparison A	14	R16TC3 (D5/0x40291)	DE16TC3 (D5/0x40295)
	Timer 4 comparison B	15	R16TU4 (D6/0x40291)	DE16TU4 (D6/0x40295)
	Timer 4 comparison A	16	R16TC4 (D7/0x40291)	DE16TC4 (D7/0x40295)
	Timer 5 comparison B	17	R16TU5 (D0/0x40292)	DE16TU5 (D0/0x40296)
	Timer 5 comparison A	18	R16TC5 (D1/0x40292)	DE16TC5 (D1/0x40296)
8-bit programmable timer	Timer 0 underflow	19	R8TU0 (D2/0x40292)	DE8TU0 (D2/0x40296)
	Timer 1 underflow	20	R8TU1 (D3/0x40292)	DE8TU1 (D3/0x40296)
	Timer 2 underflow	21	R8TU2 (D4/0x40292)	DE8TU2 (D4/0x40296)
	Timer 3 underflow	22	R8TU3 (D5/0x40292)	DE8TU3 (D5/0x40296)
Serial interface	Ch.0 receive buffer full	23	RSRX0 (D6/0x40292)	DESRX0 (D6/0x40296)
	Ch.0 transmit buffer empty	24	RSTX0 (D7/0x40292)	DESTX0 (D7/0x40296)
	Ch.1 receive buffer full	25	RSRX1 (D0/0x40293)	DESRX1 (D0/0x40297)
	Ch.1 transmit buffer empty	26	RSTX1 (D1/0x40293)	DESTX1 (D1/0x40297)
A/D converter	End of A/D conversion	27	RADE (D2/0x40293)	DEADE (D2/0x40297)
Ports	Port input 4	28	RP4 (D4/0x40293)	DEP4 (D4/0x40297)
	Port input 5	29	RP5 (D5/0x40293)	DEP5 (D5/0x40297)
	Port input 6	30	RP4 (D6/0x40293)	DEP4 (D6/0x40297)
	Port input 7	31	RP7 (D7/0x40293)	DEP7 (D7/0x40297)

These interrupt factors are used in common for interrupt requests and IDMA invocation requests.
To invoke IDMA upon the occurrence of an interrupt factor, set the corresponding bits of the IDMA request and IDMA enable registers shown in the table by writing "1". Then when an interrupt factor occurs, an interrupt request to the CPU is kept pending and the corresponding IDMA channel is invoked.
The interrupt factor flag that has been set to "1" remains set until the DMA transfer invoked by it is completed.
If the following two conditions are met when one DMA transfer is completed, an interrupt request is generated without resetting the interrupt factor flag.
• The transfer counter has reached 0.
• DINTEN in control information is set to "1" (interrupt enabled).
In this case, the IDMA request register is cleared to "0". Therefore, if IDMA needs to be invoked when an interrupt factor occurs next time, this register must be set up again. To prevent unwanted IDMA requests from being generated, this setting must be performed before enabling interrupts and after resetting the interrupt factor flag. The IDMA enable bit is not cleared and remains set to "1".

If the transfer counter is not 0, the interrupt factor flag is reset when the DMA transfer is completed, so that no interrupt is generated. In this case, the IDMA request bit and IDMA enable bit are not cleared and remain set to "1".
When DINTEN in control information has been set to "0", the interrupt factor flag is reset even if the transfer counter reaches 0, so that no interrupt is generated. In this case, the IDMA request bit is not cleared but the IDMA enable bit is cleared.

If the IDMA request register bit is left reset to "0", the relevant interrupt factor generates an interrupt request and not a IDMA request.

The control registers (interrupt enable register and interrupt priority register) corresponding to the interrupt factor do not affect IDMA invocation. IDMA can be invoked even if the interrupt enable bit in ITC is set to "0" (interrupt disabled). However, these register must be set to enable the interrupt when generating the interrupt after completing the DMA transfer.

IDMA invocation by a trigger in the software application

All IDMA channels for which control information is set, including those corresponding to interrupt factors described above, can be invoked by a trigger in the software application.
The following bits are used for this control:
IDMA channel number set-up: DCHN[6:0] (D[6:0]) / IDMA start register (0x48204)
IDMA start control: DSTART (D7) / IDMA start register (0x48204)

When the IDMA channel number to be invoked (0 to 127) is written to DCHN and DSTART is set to "1", the specified IDMA channel starts a DMA transfer.
DSTART remains set (= "1") during a DMA transfer and is reset to "0" in hardware when one DMA transfer operation is completed.
Do not modify these bits during a DMA transfer.
If DINTEN is set to "1" (interrupt enabled), an interrupt factor for the completion of IDMA transfer is generated when one DMA transfer is completed.

IDMA invocation by link setting

If LNKEN in the control information is set to "1" (link enabled), the IDMA channel that is set in the IDMA link field "LNKCHN" is invoked successively after a DMA transfer in the link-enabled channel is completed.
The interrupt request by the first channel is generated after transfers in all linked channels are completed if the interrupt conditions are met.
To generate an interrupt at the end of an IDMA transfer, the DINTEN (end-of-transfer interrupt enable) bits in the IDMA control information for the first IDMA channel to be invoked and all the channels to be linked must be set to "1".

IDMA invocation request during a DMA transfer

An IDMA invocation request to another channel that is generated during a DMA transfer is kept pending until the DMA transfer that was being executed at the time is completed. Since an invocation request is not cleared, new requests will be accepted when the DMA transfer under execution is completed.
An IDMA invocation request to the same channel canot be accepted while the channel is executing a DMA transfer because the same interrupt factor is used. Therefore, an interval longer than the DMA transfer period is required when invoking the same channel.

IDMA invocation request when DMA transfer is disabled

An IDMA invocation request generated when IDMAEN is "0" (DMA transfer disabled) is kept pending until IDMAEN is set to "1". Since an invocation request is not cleared, it is accepted when DMA transfer is enabled.

Simultaneous generation of a software trigger and a hardware trigger

When a software trigger and the hardware trigger for the same channel are generated simultaneously, the software trigger starts IDMA transfer. The IDMA transfer by the hardware trigger is not executed since the interrupt factor is reset when the DMA transfer is completed. However, an operation like this cannot be recommended.

Operation of IDMA

IDMA has three transfer modes, in each of which data transfer operates differently. Furthermore, an interrupt factor is processed differently depending on the type of trigger. The following describes the operation of IDMA in each transfer mode and how an interrupt factor is processed for each type of trigger.

Single transfer mode

The channels for which DMOD in control information is set to "00" operate in single transfer mode. In this mode, a transfer operation invoked by one trigger is completed after transferring one data unit of the size set by DATSIZ. If a data transfer needs to be performed a number of times as set by the transfer counter, an equal number of triggers are required.

The operation of IDMA in single transfer mode is shown by the flow chart in Figure 3.1.

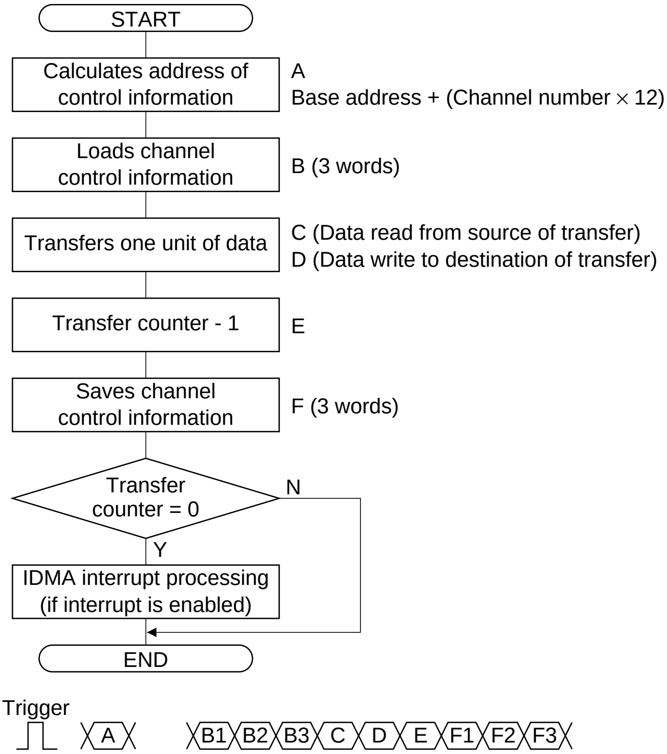


Figure 3.1 Operation Flow in Single Transfer Mode

(1) When a trigger is accepted, the address for control information is calculated from the base address and channel number.

(2) Control information is read from the calculated address into the internal temporary register.

(3) Data of the size set in the control information is read from the source address.

(4) The read data is written to the destination address.

(5) The address is incremented or decremented and the transfer counter is decremented.

(6) The modified control information is written to RAM.

(7) In the case of a hardware trigger, the interrupt control bits are processed before completing IDMA.

Condition	Interrupt factor flag	IDMA request bit	IDMA enable bit
Transfer counter ≠ "0":	Reset ("0")	Not changed ("1")	Not changed ("1")
Transfer counter = "0", DINTEN = "1":	Not changed ("1")	Reset ("0")	Not changed ("1")
Transfer counter = "0", DINTEN = "0":	Reset ("0")	Not changed ("1")	Reset ("0")

Successive transfer mode

The channels for which DMOD in control information is set to "01" operate in successive transfer mode. In this mode, a data transfer is performed by one trigger a number of times as set by the transfer counter. The transfer counter is decremented to "0" by one transfer executed.

The operation of IDMA in successive transfer mode is shown by the flow chart in Figure 3.2.

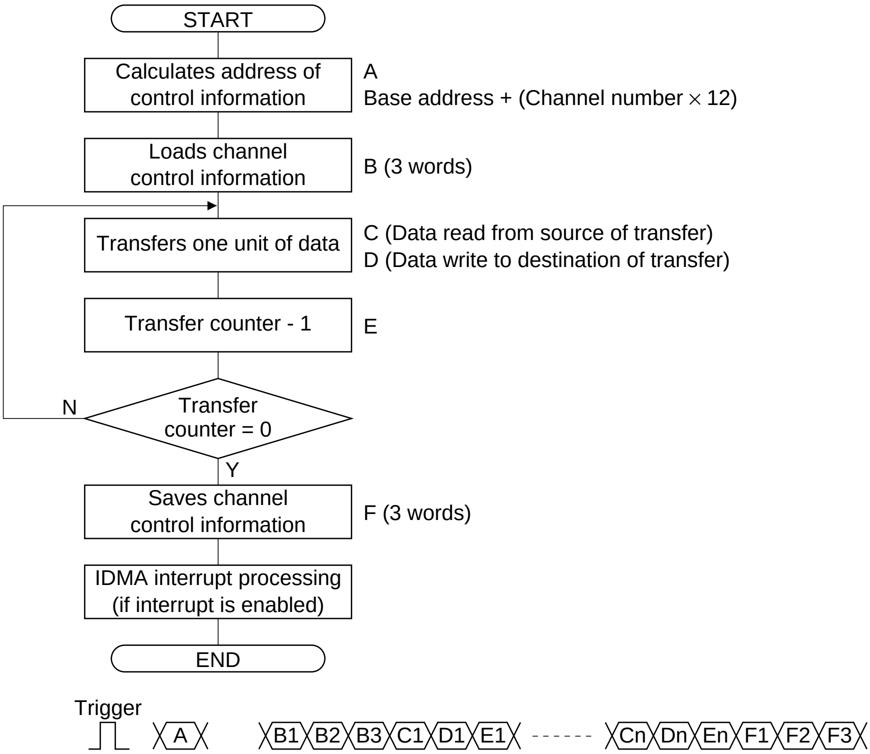


Figure 3.2 Operation Flow in Successive Transfer Mode

(1) When a trigger is accepted, the address for control information is calculated from the base address and channel number.
(2) Control information is read from the calculated address into the internal temporary register.
(3) Data of the size set in the control information is read from the source address.
(4) The read data is written to the destination address.
(5) The address is incremented or decremented and the transfer counter is decremented.
(6) Steps (3) to (5) are repeated until the transfer counter reaches 0.
(7) The modified control information is written to RAM.
(8) In the case of a hardware trigger, the interrupt control bits are processed before completing IDMA.

Condition	Interrupt factor flag	IDMA request bit	IDMA enable bit
Transfer counter ≠ "0":	Reset ("0")	Not changed ("1")	Not changed ("1")
Transfer counter = "0", DINTEN = "1":	Not changed ("1")	Reset ("0")	Not changed ("1")
Transfer counter = "0", DINTEN = "0":	Reset ("0")	Not changed ("1")	Reset ("0")

Block transfer mode

The channels for which DMOD in control information is set to "10" operate in block transfer mode. In this mode, a transfer operation invoked by one trigger is completed after transferring one block of data of the size set by BLKLEN. If a block transfer needs to be performed a number of times as set by the transfer counter, an equal number of triggers are required.

The operation of IDMA in block transfer mode is shown by the flow chart in Figure 3.3.

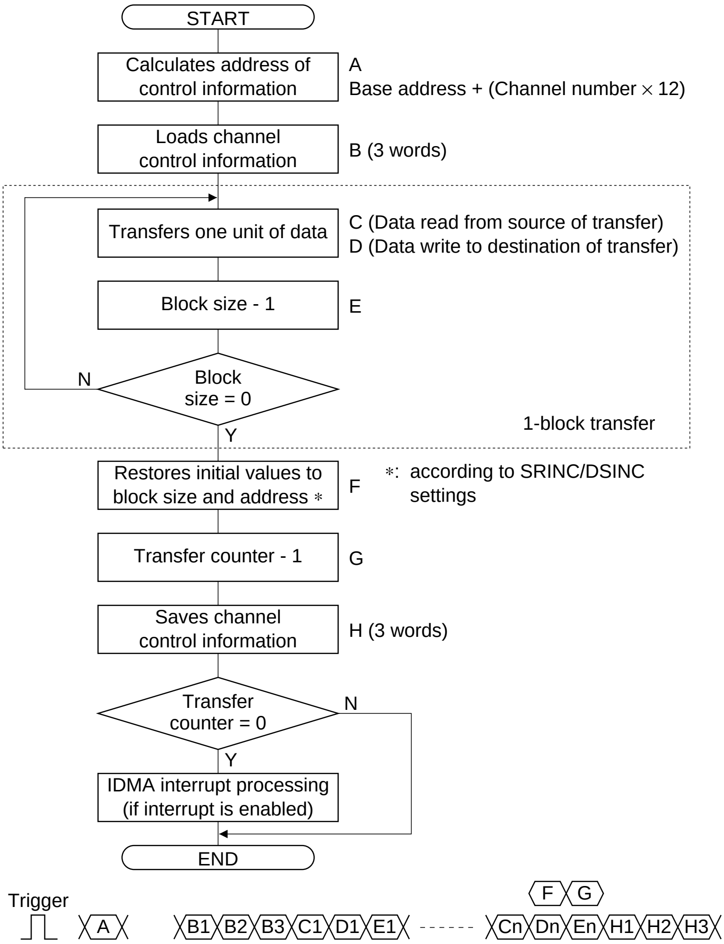


Figure 3.3 Operation Flow in Block Transfer Mode

(1) When a trigger is accepted, the address for control information is calculated from the base address and channel number.
(2) Control information is read from the calculated address into the internal temporary register.
(3) Data of the size set in the control information is read from the source address.
(4) The read data is written to the destination address.
(5) The address is incremented or decremented and BLKLEN is decremented.
(6) Steps (3) to (5) are repeated until BLKLEN reaches 0.
(7) If SRINC and DSINC are "10", the address is recycled to the initial value.
(8) The transfer counter is decremented.
(9) The modified control information is written to RAM.
(10) In the case of a hardware trigger, the interrupt control bits are processed before completing IDMA.

Condition	Interrupt factor flag	IDMA request bit	IDMA enable bit
Transfer counter ≠ "0":	Reset ("0")	Not changed ("1")	Not changed ("1")
Transfer counter = "0", DINTEN = "1":	Not changed ("1")	Reset ("0")	Not changed ("1")
Transfer counter = "0", DINTEN = "0":	Reset ("0")	Not changed ("1")	Reset ("0")

Processing of interrupt factors by type of trigger

- **When invoked by an interrupt factor**

The interrupt factor flag by which IDMA has been invoked remains set even during a DMA transfer.
If the transfer counter is decremented to 0 and DINTEN = "1" (interrupt enabled) when one DMA transfer is completed, the interrupt factor that has invoked IDMA is not reset and an interrupt request is generated. At the same time, the IDMA request register is cleared to "0". The IDMA enable bit is not cleared and remains set to "1".

If the transfer counter is not 0, the interrupt factor flag is reset when the DMA transfer is completed, so that no interrupt is generated. In this case, the IDMA request bit and IDMA enable bit are not cleared and remain set to "1".

When DINTEN has been set to "0" (interrupt disabled), the interrupt factor flag is reset even if the transfer counter reaches 0, so that no interrupt is generated. In this case, the IDMA request bit is not cleared but the IDMA enable bit is cleared.

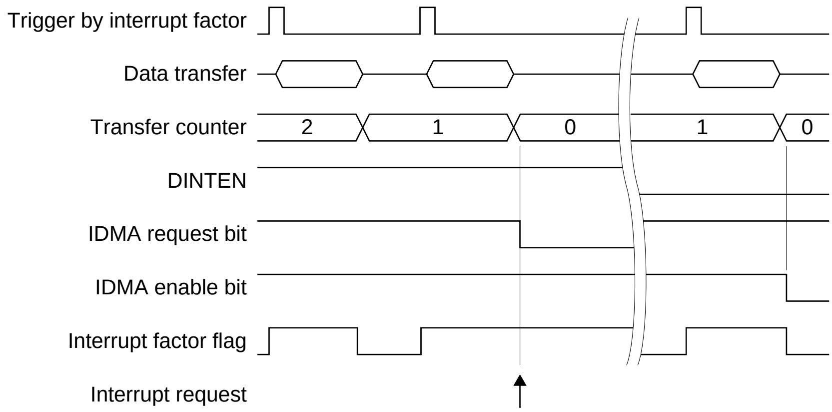


Figure 3.4 Operation when Invoked by Interrupt Factor

When IDMA is invoked by the software trigger, the IDMA interrupt factor flag FIDMA (D4)/DMA interrupt factor flag register (0x40281) will not be set.

- **When invoked by a software trigger**

If the transfer counter is decremented to 0 and DINTEN = "1" (interrupt enabled) when one DMA transfer is completed, the IDMA interrupt factor flag FIDMA (D4)/DMA interrupt factor flag register (0x40281) is set, thereby generating an interrupt request.

If the transfer counter is not 0 or DINTEN = "0" (interrupt disabled), the FIDMA flag is not set.

If the interrupt factor flag for the same channel is set during a software-triggered transfer, the IDMA invocation request by that interrupt factor flag is kept pending. However, the interrupt factor flag will be reset when the current execution is completed, so there will be no DMA transfer by the interrupt factor flag.

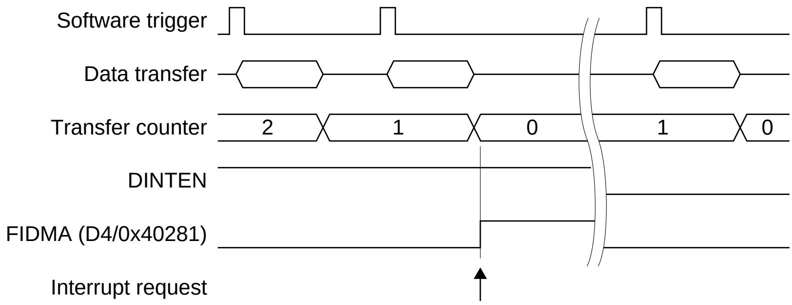


Figure 3.5 Operation when Invoked by Software Trigger

Linking

If the IDMA channel number to be executed next is set in the IDMA link field "LNKCHN" of control information and LNKEN is set to "1" (link enabled), DMA successive transfer in that IDMA channel can be performed.
An example of link setting is shown in Figure 3.6.

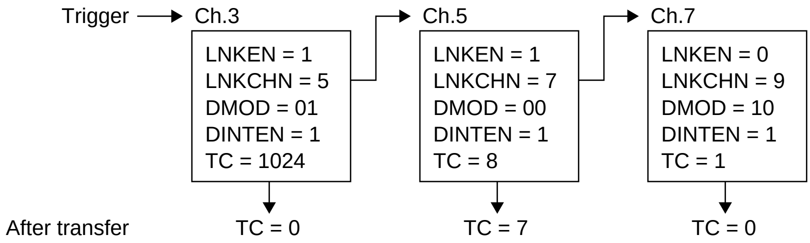


Figure 3.6 Example of Link Setting

For the above example, IDMA operates as described below.

• For trigger in hardware

(1) The IDMA channel 3 is invoked by an interrupt factor and the DMA transfer that is set is performed.
Since the IDMA is operating in successive transfer mode and the transfer counter is decremented to 0 and DINTEN is set to "1", the interrupt factor flag by which the channel 3 has been invoked remains set.

(2) Next, a DMA transfer is performed via the linked IDMA channel 5. Channel 5 is set for single transfer mode and the transfer counter in this transfer is decremented by 1.

(3) Finally, a DMA transfer in IDMA channel 7 is performed. Although the channel 7 is set for block transfer mode, the transfer counter is decremented to 0 when the transfer is completed because the number of transfers to be performed is 1.

(4) Since the interrupt factor flag that has invoked IDMA channel 3 in (1) remains set, an interrupt is generated when the IDMA transfer (channel 7) in (3) is completed. The transfer result does not affect the interrupt factor flag of channel 3.
To generate an interrupt at the end of an IDMA transfer, the DINTEN (end-of-transfer interrupt enable) bits in the IDMA control information for the first IDMA channel to be invoked and all the channels to be linked must be set to "1".

• For trigger in the software application

(1) The IDMA channel 3 is invoked by a trigger in the software application and the DMA transfer that is set is performed.
Since the IDMA is operating in successive transfer mode and the transfer counter is decremented to 0 and DINTEN is set to "1", the IDMA interrupt factor flag FIDMA (D4)/DMA interrupt factor flag register (0x40281) is set when the transfer is completed.

(2) Next, a DMA transfer is performed in the linked IDMA channel 5. The channel 5 is set for the single transfer mode and the transfer counter in this transfer is decremented by 1.

(3) Finally, a DMA transfer in IDMA channel 7 is performed. Although channel 7 is set for the block transfer mode, the transfer counter is decremented to 0 when the transfer is completed because the number of transfers to be performed is 1. The completion of this transfer also causes the FIDMA flag to be set to "1". However, the FIDMA flag has already been set when the transfer is completed in (1) above.

(4) Since the FIDMA flag is set, an interrupt request is generated here. In cases when IDMA has been invoked by a trigger in the software application, if the transfer counter in any one of the linked channels is decremented to 0 and DINTEN for that channel is set to "1", an interrupt request for the completion of IDMA transfer is generated when a transfer operation in each of the linked channels is completed. The channel in which an interrupt request has been generated can be verified by reading out the transfer counter.

Transfer operations in each channel are performed as described earlier.

Interrupt Function of Intelligent DMA

IDMA can generate an interrupt that causes invocation of IDMA and an interrupt for the completion of IDMA transfer itself.

Interrupt when invoked by an interrupt factor

If the corresponding bits of the IDMA request and interrupt enable registers are left set (= "1"), assertion of an interrupt request is kept pending even when the enabled interrupt factor has occurred and the IDMA channel assigned to that interrupt factor is invoked.

If the transfer counter is decremented to 0 and DINTEN = "1" (interrupt enabled) when one DMA transfer is completed, the interrupt factor that has invoked IDMA is not reset and an interrupt request is generated. At the same time, the IDMA request register is cleared to "0". The IDMA enable bit is not cleared and remains set to "1".

If the transfer counter is not 0, the interrupt factor flag is reset when the DMA transfer is completed, so that no interrupt is generated. In this case, the IDMA request bit and IDMA enable bit are not cleared and remain set to "1".

When DINTEN has been set to "0" (interrupt disabled), the interrupt factor flag is reset even if the transfer counter reaches 0, so that no interrupt is generated. In this case, the IDMA request bit is not cleared but the IDMA enable bit is cleared.

When IDMA is invoked by the software trigger, the IDMA interrupt factor flag FIDMA (D4)/DMA interrupt factor flag register (0x40281) will not be set.

For details about the interrupt factors that can be used to invoke IDMA and the interrupt control registers, refer to the descriptions of the peripheral circuits in this manual.
Note that the priority levels of interrupt factors are set by the interrupt priority register. Refer to "ITC (Interrupt Controller)". However, when compared between IDMA and interrupt requests, IDMA is given higher priority over the other. Consequently, even when an interrupt factor occurring during an IDMA transfer has higher priority than the interrupt factor that invoked the IDMA transfer, an interrupt request for it or a new IDMA invocation request is not accepted until after the current IDMA transfer is completed.

Software-triggered interrupts

If the transfer counter is decremented to 0 and DINTEN = "1" (interrupt enabled) when one DMA transfer operation is completed, the IDMA interrupt factor flag FIDMA (D4)/DMA interrupt factor flag register (0x40281) is set, thereby generating an interrupt request. If the transfer counter is not 0 or DINTEN = "0" (interrupt disabled), the FIDMA flag is not set.

IDMA interrupt control register in the interrupt controller

The following registers are used to control an interrupt for the completion of IDMA transfer:
Interrupt factor flag: FIDMA (D4) / DMA interrupt factor flag register (0x40281)
Interrupt enable: EIDMA (D4) / DMA interrupt enable register (0x40271)
Interrupt level: PDM[2:0](D[2:0]) / IDMA interrupt priority register (0x40265)

When a DMA transfer in the IDMA channel invoked by a trigger in the software application or subsequent link is completed and the transfer counter is decremented to 0, the interrupt factor flag for the completion of IDMA transfer is set to "1". However, this requires as a precondition that interrupt be enabled (DINTEN = "1") in the control information for that channel. If the interrupt enable register bit remains set (= "1") when the flag is set, an interrupt request is generated. Interrupts can be disabled by leaving the interrupt enable register bit cleared (= "0"). Use the interrupt priority register to set interrupt priority levels (0 to 7). An interrupt request to the CPU is accepted on condition that no other interrupt request of higher priority is generated.
Furthermore, it is only when the PSR's IE bit = "1" (interrupt enabled) and the set value of IL is smaller than the IDMA interrupt level which is set by the interrupt priority register that the CPU actually accepts an IDMA interrupt request.
For details about these interrupt control registers, and for information on device operation when an interrupt occurs, refer to "ITC (Interrupt Controller)".

Trap vector

The trap vector address for an interrupt upon completion of IDMA transfer by default is set as follows:
When BTA3 = H (high), 0x0080068; when BTA3 = L (low), 0x0C00068

The trap table base address can be changed using the TTBR registers (0x48134 to 0x48137).

I/O Memory of Intelligent DMA

Table 3.3 shows the control bits of IDMA.

Table 3.3 Control Bits of IDMA

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
IDMA interrupt priority register	**0040265 (B)**	D7–3	**–**	reserved	–				–	–	0 when being read.
		D2	**PDM2**	IDMA interrupt level	0 to 7				X	R/W	
		D1	**PDM1**						X		
		D0	**PDM0**						X		
DMA interrupt enable register	**0040271 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**EIDMA**	IDMA	1	Enabled	0	Disabled	0	R/W	
		D3	**EHDM3**	High-speed DMA Ch.3					0	R/W	
		D2	**EHDM2**	High-speed DMA Ch.2					0	R/W	
		D1	**EHDM1**	High-speed DMA Ch.1					0	R/W	
		D0	**EHDM0**	High-speed DMA Ch.0					0	R/W	
DMA interrupt factor flag register	**0040281 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**FIDMA**	IDMA	1	Factor is generated	0	No factor is generated	X	R/W	
		D3	**FHDM3**	High-speed DMA Ch.3					X	R/W	
		D2	**FHDM2**	High-speed DMA Ch.2					X	R/W	
		D1	**FHDM1**	High-speed DMA Ch.1					X	R/W	
		D0	**FHDM0**	High-speed DMA Ch.0					X	R/W	
IDMA base address low-order register	**0048200 (HW)**	DF	**DBASEL15**	IDMA base address low-order 16 bits (Initial value: 0x0C003A0)					0	R/W	
		DE	**DBASEL14**						0		
		DD	**DBASEL13**						0		
		DC	**DBASEL12**						0		
		DB	**DBASEL11**						0		
		DA	**DBASEL10**						0		
		D9	**DBASEL9**						1		
		D8	**DBASEL8**						1		
		D7	**DBASEL7**						1		
		D6	**DBASEL6**						0		
		D5	**DBASEL5**						1		
		D4	**DBASEL4**						0		
		D3	**DBASEL3**						0		
		D2	**DBASEL2**						0		
		D1	**DBASEL1**						0		
		D0	**DBASEL0**						0		
IDMA base address high-order register	**0048202 (HW)**	DF–C	**–**	reserved	–				–	–	Undefined in read.
		DB	**DBASEH11**	IDMA base address high-order 12 bits (Initial value: 0x0C003A0)					0	R/W	
		DA	**DBASEH10**						0		
		D9	**DBASEH9**						0		
		D8	**DBASEH8**						0		
		D7	**DBASEH7**						1		
		D6	**DBASEH6**						1		
		D5	**DBASEH5**						0		
		D4	**DBASEH4**						0		
		D3	**DBASEH3**						0		
		D2	**DBASEH2**						0		
		D1	**DBASEH1**						0		
		D0	**DBASEH0**						0		
IDMA start register	**0048204 (B)**	D7	**DSTART**	IDMA start	1	IDMA start	0	Stop	0	R/W	
		D6–0	**DCHN**	IDMA channel number	0 to 127				0	R/W	
IDMA enable register	**0048205 (B)**	D7–1	**–**	reserved	–				–	–	
		D0	**IDMAEN**	IDMA enable	1	Enabled	0	Disabled	0	R/W	

DBASEL[15:0]: IDMA base address [15:0] (D[F:0]) / IDMA base address low-order register (0x48200)
DDBASEH[11:0]: IDMA base address [27:16] (D[B:0]) / IDMA base address high-order register (0x48202)

Specify the starting address of the control information to be placed in RAM.
Use DBASEL to set the 16 low-order bits of the address and DBASEH to set the 12 high-order bits.
The address to be set in these registers must always be a word (32-bit) boundary address.
These registers cannot be read or written in bytes. The registers must be accessed in words for read/write operations to address 0x48200, and in half-words for read/write operations to addresses 0x48200 and 0x48202. Write operations in half-words must be performed in order of 0x48200 and 0x48202. Read operations in half-words maybe performed in any order.
Write operations to the IDMA base address registers during a DMA transfer are ignored. When the register is read during a DMA transfer, the read data is indeterminate.
At initial reset, the base address is set to 0xC003A0.

IDMAEN: DMA enable (D0) / DMA enable register (0x48205)

Enable a DMA transfer.

Write "1": Enabled
Write "0": Disabled
Read: Valid

A data transfer operation by intelligent DMA and high-speed DMA is enabled by writing "1" to IDMAEN.
DMA transfer is disabled by writing "0" to IDMAEN.
Be sure to disable DMA transfers (IDMAEN = "0") before you set the base address and transfer conditions.
At initial reset, IDMAEN is set to "0" (disabled).

DCHN[6:0]: IDMA channel number (D[6:0]) / IDMA start register (0x48204)

Set the channel numbers (0 to 127) to be invoked by a trigger in the software application.
At initial reset, DCHN is set to "0".

DSTART: IDMA start (D7) / IDMA start register (0x48204)

Use this register for a trigger in the software application and for monitoring the operation of IDMA.

When written
Write "1": IDMA started
Write "0": Invalid

When read
Read "1": IDMA operating (only when invoked by software trigger)
Read "0": IDMA inactive

When DSTART is set to "1", it functions as a trigger in the software application, invoking the IDMA channel that is set in the DCHN register.
At initial reset, DSTART is set to "0".

PDM2–PDM0: DMA interrupt level (D[2:0]) / IDMA interrupt priority register (0x40265)

Set the priority level of the interrupt upon completion of IDMA transfer in the range of 0 to 7.
At initial reset, the contents of this register are indeterminate.

EIDMA: IDMA interrupt enable (D2) / DMA interrupt enable register (0x40271)

Enable or disable occurrence of an interrupt to the CPU.

Write "1": Interrupt enabled
Write "0": Interrupt disabled
Read: Valid

This bit controls the interrupt generated upon completion of IDMA transfer. The interrupt is enabled by setting this bit to "1" and disabled by setting this bit to "0".
At initial reset, EIDMA is set to "0" (interrupt disable).

FIDMA: IDMA interrupt factor flag (D2) / DMA interrupt factor flag register (0x40281)

Indicate the occurrence status of an IDMA interrupt request.

When read

Read "1": Interrupt factor occurred
Read "0": No interrupt factor occurred

When written using reset-only method (default)

Write "1": Interrupt factor flag is reset
Write "0": Invalid

When written using the read/write method

Write "1": Interrupt factor flag is set
Write "0": Interrupt factor flag is reset

This flag is set to "1" when one DMA transfer initiated by a software trigger or subsequent link is completed and the transfer counter is decremented to 0. However, this requires as a precondition that interrupts be enabled in control information (DINTEN = "1").
At this time, an interrupt to the CPU is generated if the following conditions are met:

1. The corresponding interrupt enable register bit is set to "1".
2. No interrupt request of higher priority is generated.
3. The IE bit of the PSR is set to "1" (interrupt enable).
4. The corresponding interrupt priority register is set to a level higher than the CPU's interrupt level (IL).

In order for the next interrupt to be accepted after interrupt generation, the interrupt factor flag must be reset and the PSR must be set up again (by setting the IL below the level indicated by the interrupt priority register and setting the IE bit to "1" or executing a reti instruction).
The interrupt factor flag can only be reset by a write instruction in the software application. If the PSR is set up again to accept interrupts (or the reti instruction is executed) without resetting the interrupt factor flag, the same interrupt may occur again. Note also that the value to be written to reset the flag is "1" when using the reset-only method (RSTONLY = "1") and "0" when using the read/write method (RSTONLY = "0"). Be careful not to confuse these two cases.
This flag becomes indeterminate when initially reset, so be sure to reset it in the software application.

Programming Notes

(1) Before setting the IDMA base address, be sure to disable DMA transfers (IDMAEN = "0"). Writing to the IDMA base address register is ignored when the DMA transfer is enabled (IDMAEN = "1"). Also, when the register is read during a DMA transfer, the data is indeterminate. When setting or rewriting control information for each channel, make sure that DMA transfers will not occur in any channel.

(2) The address that is set in the IDMA base address register must always be a word (32-bit) boundary address.

(3) After an initial reset, the interrupt factor flag (FIDMA) becomes indeterminate. To prevent unwanted interrupts from occurring, be sure to reset the flag in a program.

(4) Once an interrupt occurs, be sure to reset the interrupt factor flag (FIDMA) before setting up the PSR again or executing the reti instruction. This ensures that an interrupt will not be generated for the same factor.

(5) If all the following conditions are met, the transfer counter value becomes invalid during IDMA transfer so data cannot be transferred properly.

1. The IDMA control information (source/destination addresses, transfer counter, etc.) is placed in the external EDO DRAM.
2. The DRAM access timing condition is set to EDO mode by the BCU register.
3. The bus clock is set to x2 speed mode (#X2SPD pin = "0").

When placing the control information in the EDO DRAM in x2 speed mode, the DRAM access timing condition must be set to high-speed page mode.
Or place the control information in the internal RAM. Using the internal RAM increases the performance because the overhead during IDMA transfer is decreased to 6 cycles on both load/store operations.

(6) The current version of the DMA controller (C33 macro Model 2 rev. 2.2) does not set the IDMA interrupt factor flag FIDMA (D4)/DMA interrupt factor flag register (0x40281) even when an IDMA transfer that was started with a software trigger has completed (transfer counter = 0). Therefore, a transfer completion interrupt cannot be used in software trigger mode.

THIS PAGE IS BLANK.

E0C33 Family ASIC Macro Manual

Appendix I/O MAP

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer clock select register	**0040146 (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**P8TPCK3**	8-bit timer 3 clock selection	1	θ/1	0	Divided clk.	0	R/W	θ: selected by Prescaler clock select register (0x40181)
		D2	**P8TPCK2**	8-bit timer 2 clock selection	1	θ/1	0	Divided clk.	0	R/W	
		D1	**P8TPCK1**	8-bit timer 1 clock selection	1	θ/1	0	Divided clk.	0	R/W	
		D0	**P8TPCK0**	8-bit timer 0 clock selection	1	θ/1	0	Divided clk.	0	R/W	
16-bit timer 0 clock control register	**0040147 (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**P16TON0**	16-bit timer 0 clock control	1	On	0	Off	0	R/W	
		D2 D1 D0	**P16TS02** **P16TS01** **P16TS00**	16-bit timer 0 clock division ratio selection	P16TS0[2:0]			Division ratio	0 0 0	R/W	θ: selected by Prescaler clock select register (0x40181) 16-bit timer 0 can be used as a watchdog timer.
					1	1	1	θ/4096			
					1	1	0	θ/1024			
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 1 clock control register	**0040148 (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**P16TON1**	16-bit timer 1 clock control	1	On	0	Off	0	R/W	
		D2 D1 D0	**P16TS12** **P16TS11** **P16TS10**	16-bit timer 1 clock division ratio selection	P16TS1[2:0]			Division ratio	0 0 0	R/W	θ: selected by Prescaler clock select register (0x40181)
					1	1	1	θ/4096			
					1	1	0	θ/1024			
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 2 clock control register	**0040149 (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**P16TON2**	16-bit timer 2 clock control	1	On	0	Off	0	R/W	
		D2 D1 D0	**P16TS22** **P16TS21** **P16TS20**	16-bit timer 2 clock division ratio selection	P16TS2[2:0]			Division ratio	0 0 0	R/W	θ: selected by Prescaler clock select register (0x40181)
					1	1	1	θ/4096			
					1	1	0	θ/1024			
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 3 clock control register	**004014A (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**P16TON3**	16-bit timer 3 clock control	1	On	0	Off	0	R/W	
		D2 D1 D0	**P16TS32** **P16TS31** **P16TS30**	16-bit timer 3 clock division ratio selection	P16TS3[2:0]			Division ratio	0 0 0	R/W	θ: selected by Prescaler clock select register (0x40181)
					1	1	1	θ/4096			
					1	1	0	θ/1024			
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
16-bit timer 4 clock control register	**004014B (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**P16TON4**	16-bit timer 4 clock control	1	On	0	Off	0	R/W	
		D2 D1 D0	**P16TS42** **P16TS41** **P16TS40**	16-bit timer 4 clock division ratio selection	P16TS4[2:0]			Division ratio	0 0 0	R/W	θ: selected by Prescaler clock select register (0x40181)
					1	1	1	θ/4096			
					1	1	0	θ/1024			
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			

(B) in [Address] indicates an 8-bit register and (HW) indicates a 16-bit register.

The meaning of the symbols described in [Init.] are listed below:

0, 1: Initial values that are set at initial reset.
(However, the registers for the bus and input/output ports are not initialized at hot start.)

X: Not initialized at initial reset.

–: Not set in the circuit.

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 5	**004014C**	D7–4	**–**	reserved	–				–	–	0 when being read.
clock control	**(B)**	D3	**P16TON5**	16-bit timer 5 clock control	1 On		0 Off		0	R/W	
register		D2	**P16TS52**	16-bit timer 5	P16TS5[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P16TS51**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P16TS50**		1	1	0	θ/1024	0		register (0x40181)
					1	0	1	θ/256			
					1	0	0	θ/64			
					0	1	1	θ/16			
					0	1	0	θ/4			
					0	0	1	θ/2			
					0	0	0	θ/1			
8-bit timer 0/1	**004014D**	D7	**P8TON1**	8-bit timer 1 clock control	1 On		0 Off		0	R/W	
clock control	**(B)**	D6	**P8TS12**	8-bit timer 1	P8TS1[2:0]			Division ratio	0	R/W	θ: selected by
register		D5	**P8TS11**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D4	**P8TS10**		1	1	0	θ/2048	0		register (0x40181)
					1	0	1	θ/1024			
					1	0	0	θ/512			8-bit timer 1 can
					0	1	1	θ/256			generate the OSC3
					0	1	0	θ/128			oscillation-stabilize
					0	0	1	θ/64			waiting period.
					0	0	0	θ/32			
		D3	**P8TON0**	8-bit timer 0 clock control	1 On		0 Off		0	R/W	
		D2	**P8TS02**	8-bit timer 0	P8TS0[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P8TS01**	clock division ratio selection	1	1	1	θ/256	0		Prescaler clock select
		D0	**P8TS00**		1	1	0	θ/128	0		register (0x40181)
					1	0	1	θ/64			
					1	0	0	θ/32			8-bit timer 0 can
					0	1	1	θ/16			generate the DRAM
					0	1	0	θ/8			refresh clock.
					0	0	1	θ/4			
					0	0	0	θ/2			
8-bit timer 2/3	**004014E**	D7	**P8TON3**	8-bit timer 3 clock control	1 On		0 Off		0	R/W	
clock control	**(B)**	D6	**P8TS32**	8-bit timer 3	P8TS3[2:0]			Division ratio	0	R/W	θ: selected by
register		D5	**P8TS31**	clock division ratio selection	1	1	1	θ/256	0		Prescaler clock select
		D4	**P8TS30**		1	1	0	θ/128	0		register (0x40181)
					1	0	1	θ/64			
					1	0	0	θ/32			8-bit timer 3 can
					0	1	1	θ/16			generate the clock for
					0	1	0	θ/8			the serial I/F Ch.1.
					0	0	1	θ/4			
					0	0	0	θ/2			
		D3	**P8TON2**	8-bit timer 2 clock control	1 On		0 Off		0	R/W	
		D2	**P8TS22**	8-bit timer 2	P8TS2[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**P8TS21**	clock division ratio selection	1	1	1	θ/4096	0		Prescaler clock select
		D0	**P8TS20**		1	1	0	θ/2048	0		register (0x40181)
					1	0	1	θ/64			
					1	0	0	θ/32			8-bit timer 2 can
					0	1	1	θ/16			generate the clock for
					0	1	0	θ/8			the serial I/F Ch.0.
					0	0	1	θ/4			
					0	0	0	θ/2			
A/D clock	**004014F**	D7–4	**–**	reserved	–				–	–	0 when being read.
control register	**(B)**	D3	**PSONAD**	A/D converter clock control	1 On		0 Off		0	R/W	
		D2	**PSAD2**	A/D converter clock division ratio	P8TS0[2:0]			Division ratio	0	R/W	θ: selected by
		D1	**PSAD1**	selection	1	1	1	θ/256	0		Prescaler clock select
		D0	**PSAD0**		1	1	0	θ/128	0		register (0x40181)
					1	0	1	θ/64			
					1	0	0	θ/32			
					0	1	1	θ/16			
					0	1	0	θ/8			
					0	0	1	θ/4			
					0	0	0	θ/2			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Clock timer Run/Stop register	**0040151 (B)**	D7–2	–	reserved	–	–	–	0 when being read.
		D1	**TCRST**	Clock timer reset	1 Reset / 0 Invalid	X	W	0 when being read.
		D0	**TCRUN**	Clock timer Run/Stop control	1 Run / 0 Stop	X	R/W	
Clock timer interrupt control register	**0040152 (B)**	D7	**TCISE2**	Clock timer interrupt factor selection	TCISE[2:0] / Interrupt factor	X	R/W	
		D6	**TCISE1**		1 1 1 / None	X		
		D5	**TCISE0**		1 1 0 / Day	X		
					1 0 1 / Hour			
					1 0 0 / Minute			
					0 1 1 / 1 Hz			
					0 1 0 / 2 Hz			
					0 0 1 / 8 Hz			
					0 0 0 / 32 Hz			
		D4	**TCASE2**	Clock timer alarm factor selection	TCASE[2:0] / Alarm factor	X	R/W	
		D3	**TCASE1**		1 X X / Day	X		
		D2	**TCASE0**		X 1 X / Hour	X		
					X X 1 / Minute			
					0 0 0 / None			
		D1	**TCIF**	Interrupt factor generation flag	1 Generated / 0 Not generated	X	R/W	Reset by writing 1.
		D0	**TCAF**	Alarm factor generation flag	1 Generated / 0 Not generated	X	R/W	Reset by writing 1.
Clock timer divider register	**0040153 (B)**	D7	**TCD7**	Clock timer data 1 Hz	1 High / 0 Low	X	R	
		D6	**TCD6**	Clock timer data 2 Hz	1 High / 0 Low	X	R	
		D5	**TCD5**	Clock timer data 4 Hz	1 High / 0 Low	X	R	
		D4	**TCD4**	Clock timer data 8 Hz	1 High / 0 Low	X	R	
		D3	**TCD3**	Clock timer data 16 Hz	1 High / 0 Low	X	R	
		D2	**TCD2**	Clock timer data 32 Hz	1 High / 0 Low	X	R	
		D1	**TCD1**	Clock timer data 64 Hz	1 High / 0 Low	X	R	
		D0	**TCD0**	Clock timer data 128 Hz	1 High / 0 Low	X	R	
Clock timer second register	**0040154 (B)**	D7–6	–	reserved	–	–	–	0 when being read.
		D5	**TCMD5**	Clock timer second counter data	0 to 59 seconds	X	R	
		D4	**TCMD4**	TCMD5 = MSB		X		
		D3	**TCMD3**	TCMD0 = LSB		X		
		D2	**TCMD2**			X		
		D1	**TCMD1**			X		
		D0	**TCMD0**			X		
Clock timer minute register	**0040155 (B)**	D7–6	–	reserved	–	–	–	0 when being read.
		D5	**TCHD5**	Clock timer minute counter data	0 to 59 minutes	X	R/W	
		D4	**TCHD4**	TCHD5 = MSB		X		
		D3	**TCHD3**	TCHD0 = LSB		X		
		D2	**TCHD2**			X		
		D1	**TCHD1**			X		
		D0	**TCHD0**			X		
Clock timer hour register	**0040156 (B)**	D7–5	–	reserved	–	–	–	0 when being read.
		D4	**TCDD4**	Clock timer hour counter data	0 to 23 hours	X	R/W	
		D3	**TCDD3**	TCDD4 = MSB		X		
		D2	**TCDD2**	TCDD0 = LSB		X		
		D1	**TCDD1**			X		
		D0	**TCDD0**			X		
Clock timer day (low-order) register	**0040157 (B)**	D7	**TCND7**	Clock timer day counter data	0 to 65535 days	X	R/W	
		D6	**TCND6**	(low-order 8 bits)	(low-order 8 bits)	X		
		D5	**TCND5**	TCND0 = LSB		X		
		D4	**TCND4**			X		
		D3	**TCND3**			X		
		D2	**TCND2**			X		
		D1	**TCND1**			X		
		D0	**TCND0**			X		
Clock timer day (high-order) register	**0040158 (B)**	D7	**TCND15**	Clock timer day counter data	0 to 65535 days	X	R/W	
		D6	**TCND14**	(high-order 8 bits)	(high-order 8 bits)	X		
		D5	**TCND13**	TCND15 = MSB		X		
		D4	**TCND12**			X		
		D3	**TCND11**			X		
		D2	**TCND10**			X		
		D1	**TCND9**			X		
		D0	**TCND8**			X		
Clock timer minute comparison register	**0040159 (B)**	D7–6	–	reserved	–	–	–	0 when being read.
		D5	**TCCH5**	Clock timer minute comparison data	0 to 59 minutes	X	R/W	
		D4	**TCCH4**	TCCH5 = MSB	(Note) Can be set within 0–63.	X		
		D3	**TCCH3**	TCCH0 = LSB		X		
		D2	**TCCH2**			X		
		D1	**TCCH1**			X		
		D0	**TCCH0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Clock timer hour comparison register	**004015A (B)**	D7–5	**–**	reserved	–	–	–	0 when being read.
		D4	**TCCD4**	Clock timer hour comparison data	0 to 23 hours	X	R/W	
		D3	**TCCD3**	TCCD4 = MSB	(Note) Can be set within 0–31.	X		
		D2	**TCCD2**	TCCD0 = LSB		X		
		D1	**TCCD1**			X		
		D0	**TCCD0**			X		
Clock timer day comparison register	**004015B (B)**	D7–5	**–**	reserved	–	–	–	0 when being read.
		D4	**TCCN4**	Clock timer day comparison data	0 to 31 days	X	R/W	Compared with TCND[4:0].
		D3	**TCCN3**	TCCN4 = MSB		X		
		D2	**TCCN2**	TCCN0 = LSB		X		
		D1	**TCCN1**			X		
		D0	**TCCN0**			X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer 0 control register	**0040160 (B)**	D7–3	–	reserved	–				–	–	0 when being read.
		D2	**PTOUT0**	8-bit timer 0 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET0**	8-bit timer 0 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN0**	8-bit timer 0 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 0 reload data register	**0040161 (B)**	D7	**RLD07**	8-bit timer 0 reload data	0 to 255				X	R/W	
		D6	**RLD06**	RLD07 = MSB					X		
		D5	**RLD05**	RLD00 = LSB					X		
		D4	**RLD04**						X		
		D3	**RLD03**						X		
		D2	**RLD02**						X		
		D1	**RLD01**						X		
		D0	**RLD00**						X		
8-bit timer 0 counter data register	**0040162 (B)**	D7	**PTD07**	8-bit timer 0 counter data	0 to 255				X	R	
		D6	**PTD06**	PTD07 = MSB					X		
		D5	**PTD05**	PTD00 = LSB					X		
		D4	**PTD04**						X		
		D3	**PTD03**						X		
		D2	**PTD02**						X		
		D1	**PTD01**						X		
		D0	**PTD00**						X		
8-bit timer 1 control register	**0040164 (B)**	D7–3	–	reserved	–				–	–	0 when being read.
		D2	**PTOUT1**	8-bit timer 1 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET1**	8-bit timer 1 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN1**	8-bit timer 1 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 1 reload data register	**0040165 (B)**	D7	**RLD17**	8-bit timer 1 reload data	0 to 255				X	R/W	
		D6	**RLD16**	RLD17 = MSB					X		
		D5	**RLD15**	RLD10 = LSB					X		
		D4	**RLD14**						X		
		D3	**RLD13**						X		
		D2	**RLD12**						X		
		D1	**RLD11**						X		
		D0	**RLD10**						X		
8-bit timer 1 counter data register	**0040166 (B)**	D7	**PTD17**	8-bit timer 1 counter data	0 to 255				X	R	
		D6	**PTD16**	PTD17 = MSB					X		
		D5	**PTD15**	PTD10 = LSB					X		
		D4	**PTD14**						X		
		D3	**PTD13**						X		
		D2	**PTD12**						X		
		D1	**PTD11**						X		
		D0	**PTD10**						X		
8-bit timer 2 control register	**0040168 (B)**	D7–3	–	reserved	–				–	–	0 when being read.
		D2	**PTOUT2**	8-bit timer 2 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET2**	8-bit timer 2 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN2**	8-bit timer 2 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 2 reload data register	**0040169 (B)**	D7	**RLD27**	8-bit timer 2 reload data	0 to 255				X	R/W	
		D6	**RLD26**	RLD27 = MSB					X		
		D5	**RLD25**	RLD20 = LSB					X		
		D4	**RLD24**						X		
		D3	**RLD23**						X		
		D2	**RLD22**						X		
		D1	**RLD21**						X		
		D0	**RLD20**						X		
8-bit timer 2 counter data register	**004016A (B)**	D7	**PTD27**	8-bit timer 2 counter data	0 to 255				X	R	
		D6	**PTD26**	PTD27 = MSB					X		
		D5	**PTD25**	PTD20 = LSB					X		
		D4	**PTD24**						X		
		D3	**PTD23**						X		
		D2	**PTD22**						X		
		D1	**PTD21**						X		
		D0	**PTD20**						X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
8-bit timer 3 control register	**004016C (B)**	D7–3	**–**	reserved	–				–	–	0 when being read.
		D2	**PTOUT3**	8-bit timer 3 clock output control	1	On	0	Off	0	R/W	
		D1	**PSET3**	8-bit timer 3 preset	1	Preset	0	Invalid	–	W	0 when being read.
		D0	**PTRUN3**	8-bit timer 3 Run/Stop control	1	Run	0	Stop	0	R/W	
8-bit timer 3 reload data register	**004016D (B)**	D7	**RLD37**	8-bit timer 3 reload data	0 to 255				X	R/W	
		D6	**RLD36**	RLD37 = MSB					X		
		D5	**RLD35**	RLD30 = LSB					X		
		D4	**RLD34**						X		
		D3	**RLD33**						X		
		D2	**RLD32**						X		
		D1	**RLD31**						X		
		D0	**RLD30**						X		
8-bit timer 3 counter data register	**004016E (B)**	D7	**PTD37**	8-bit timer 3 counter data	0 to 255				X	R	
		D6	**PTD36**	PTD37 = MSB					X		
		D5	**PTD35**	PTD30 = LSB					X		
		D4	**PTD34**						X		
		D3	**PTD33**						X		
		D2	**PTD32**						X		
		D1	**PTD31**						X		
		D0	**PTD30**						X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Watchdog timer write-protect register	**0040170 (B)**	D7	**WRWD**	EWD write protection	1 Write enabled 0 Write-protect	0	R/W	
		D6–0	–	–	–	–	–	0 when being read.
Watchdog timer enable register	**0040171 (B)**	D7–2	–	–	–	–	–	0 when being read.
		D1	**EWD**	Watchdog timer enable	1 NMI enabled 0 NMI disabled	0	R/W	
		D0	–	–	–	–	–	0 when being read.

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Power control register	**0040180 (B)**	D7	**CLKDT1**	System clock division ratio selection	CLKDT[1:0]		Division ratio		0	R/W	
		D6	**CLKDT0**		1	1	1/8		0		
					1	0	1/4				
					0	1	1/2				
					0	0	1/1				
		D5	**PSCON**	Prescaler On/Off control	1	On	0	Off	1	R/W	
		D4–3	–	reserved	–				0	–	Writing 1 not allowed.
		D2	**CLKCHG**	CPU operating clock switch	1	OSC3	0	OSC1	1	R/W	
		D1	**SOSC3**	High-speed (OSC3) oscillation On/Off	1	On	0	Off	1	R/W	
		D0	**SOSC1**	Low-speed (OSC1) oscillation On/Off	1	On	0	Off	1	R/W	
Prescaler clock select register	**0040181 (B)**	D7–1	–	reserved	–				0	–	
		D0	**PSCDT0**	Prescaler clock selection	1	OSC1	0	OSC3/PLL	0	R/W	
Clock option register	**0040190 (B)**	D7–4	–	–	–				–	–	0 when being read.
		D3	**HLT2OP**	HALT clock option	1	On	0	Off	0	R/W	
		D2	**8T1ON**	OSC3-stabilize waiting function	1	Off	0	On	1	R/W	
		D1	–	reserved	–				0	–	Do not write 1.
		D0	**PF1ON**	OSC1 external output control	1	On	0	Off	0	R/W	
Power control protect register	**004019E (B)**	D7	**CLGP7**	Power control register protect flag	Writing 10010110 (0x96) removes the write protection of the power control register (0x40180). Writing another value set the write protection.				0	R/W	
		D6	**CLGP6**						0		
		D5	**CLGP5**						0		
		D4	**CLGP4**						0		
		D3	**CLGP3**						0		
		D2	**CLGP2**						0		
		D1	**CLGP1**						0		
		D0	**CLGP0**						0		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Serial I/F Ch.0 transmit data register	**00401E0 (B)**	D7	**TXD07**	Serial I/F Ch.0 transmit data	0x0 to 0xFF(0x7F)				X	R/W	7-bit asynchronous mode does not use TXD07.
		D6	**TXD06**	TXD07(06) = MSB					X		
		D5	**TXD05**	TXD00 = LSB					X		
		D4	**TXD04**						X		
		D3	**TXD03**						X		
		D2	**TXD02**						X		
		D1	**TXD01**						X		
		D0	**TXD00**						X		
Serial I/F Ch.0 receive data register	**00401E1 (B)**	D7	**RXD07**	Serial I/F Ch.0 receive data	0x0 to 0xFF(0x7F)				X	R	7-bit asynchronous mode does not use RXD07 (fixed at 0).
		D6	**RXD06**	RXD07(06) = MSB					X		
		D5	**RXD05**	RXD00 = LSB					X		
		D4	**RXD04**						X		
		D3	**RXD03**						X		
		D2	**RXD02**						X		
		D1	**RXD01**						X		
		D0	**RXD00**						X		
Serial I/F Ch.0 status register	**00401E2 (B)**	D7–6	–	–	–				–	–	0 when being read.
		D5	**TEND0**	Ch.0 transmit-completion flag	1	Transmitting	0	End	0	R	
		D4	**FER0**	Ch.0 flaming error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D3	**PER0**	Ch.0 parity error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D2	**OER0**	Ch.0 overrun error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D1	**TDBE0**	Ch.0 transmit data buffer empty	1	Empty	0	Buffer full	1	R	
		D0	**RDBF0**	Ch.0 receive data buffer full	1	Buffer full	0	Empty	0	R	
Serial I/F Ch.0 control register	**00401E3 (B)**	D7	**TXEN0**	Ch.0 transmit enable	1	Enabled	0	Disabled	0	R/W	
		D6	**RXEN0**	Ch.0 receive enable	1	Enabled	0	Disabled	0	R/W	
		D5	**EPR0**	Ch.0 parity enable	1	With parity	0	No parity	X	R/W	Valid only in asynchronous mode.
		D4	**PMD0**	Ch.0 parity mode selection	1	Odd	0	Even	X	R/W	
		D3	**STPB0**	Ch.0 stop bit selection	1	2 bits	0	1 bit	X	R/W	
		D2	**SSCK0**	Ch.0 input clock selection	1	#SCLK0	0	Internal clock	X	R/W	
		D1	**SMD01**	Ch.0 transfer mode selection	SMD0[1:0]		Transfer mode		X	R/W	
		D0	**SMD00**		1	1	8-bit asynchronous		X		
					1	0	7-bit asynchronous				
					0	1	Clock sync. Slave				
					0	0	Clock sync. Master				
Serial I/F Ch.0 IrDA register	**00401E4 (B)**	D7–5	–	–	–				–	–	0 when being read.
		D4	**DIVMD0**	Ch.0 async. clock division ratio	1	1/8	0	1/16	X	R/W	
		D3	**IRTL0**	Ch.0 IrDA I/F output logic inversion	1	Inverted	0	Direct	X	R/W	Valid only in asynchronous mode.
		D2	**IRRL0**	Ch.0 IrDA I/F input logic inversion	1	Inverted	0	Direct	X	R/W	
		D1	**IRMD01**	Ch.0 interface mode selection	IRMD0[1:0]		I/F mode		X	R/W	
		D0	**IRMD00**		1	1	reserved		X		
					1	0	IrDA 1.0				
					0	1	reserved				
					0	0	General I/F				

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Serial I/F Ch.1 transmit data register	**00401E5 (B)**	D7	**TXD17**	Serial I/F Ch.1 transmit data	0x0 to 0xFF(0x7F)				X	R/W	7-bit asynchronous mode does not use TXD17.
		D6	**TXD16**	TXD17(16) = MSB					X		
		D5	**TXD15**	TXD10 = LSB					X		
		D4	**TXD14**						X		
		D3	**TXD13**						X		
		D2	**TXD12**						X		
		D1	**TXD11**						X		
		D0	**TXD10**						X		
Serial I/F Ch.1 receive data register	**00401E6 (B)**	D7	**RXD17**	Serial I/F Ch.1 receive data	0x0 to 0xFF(0x7F)				X	R	7-bit asynchronous mode does not use RXD17 (fixed at 0).
		D6	**RXD16**	RXD17(16) = MSB					X		
		D5	**RXD15**	RXD10 = LSB					X		
		D4	**RXD14**						X		
		D3	**RXD13**						X		
		D2	**RXD12**						X		
		D1	**RXD11**						X		
		D0	**RXD10**						X		
Serial I/F Ch.1 status register	**00401E7 (B)**	D7–6	**–**	–	–				–	–	0 when being read.
		D5	**TEND1**	Ch.1 transmit-completion flag	1	Transmitting	0	End	0	R	
		D4	**FER1**	Ch.1 flaming error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D3	**PER1**	Ch.1 parity error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D2	**OER1**	Ch.1 overrun error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
		D1	**TDBE1**	Ch.1 transmit data buffer empty	1	Empty	0	Buffer full	1	R	
		D0	**RDBF1**	Ch.1 receive data buffer full	1	Buffer full	0	Empty	0	R	
Serial I/F Ch.1 control register	**00401E8 (B)**	D7	**TXEN1**	Ch.1 transmit enable	1	Enabled	0	Disabled	0	R/W	
		D6	**RXEN1**	Ch.1 receive enable	1	Enabled	0	Disabled	0	R/W	
		D5	**EPR1**	Ch.1 parity enable	1	With parity	0	No parity	X	R/W	Valid only in asynchronous mode.
		D4	**PMD1**	Ch.1 parity mode selection	1	Odd	0	Even	X	R/W	
		D3	**STPB1**	Ch.1 stop bit selection	1	2 bits	0	1 bit	X	R/W	
		D2	**SSCK1**	Ch.1 input clock selection	1	#SCLK1	0	Internal clock	X	R/W	
		D1	**SMD11**	Ch.1 transfer mode selection	SMD1[1:0]		Transfer mode		X	R/W	
		D0	**SMD10**		1	1	8-bit asynchronous		X		
					1	0	7-bit asynchronous				
					0	1	Clock sync. Slave				
					0	0	Clock sync. Master				
Serial I/F Ch.1 IrDA register	**00401E9 (B)**	D7–5	**–**	–	–				–	–	0 when being read.
		D4	**DIVMD1**	Ch.1 async. clock division ratio	1	1/8	0	1/16	X	R/W	
		D3	**IRTL1**	Ch.1 IrDA I/F output logic inversion	1	Inverted	0	Direct	X	R/W	Valid only in asynchronous mode.
		D2	**IRRL1**	Ch.1 IrDA I/F input logic inversion	1	Inverted	0	Direct	X	R/W	
		D1	**IRMD11**	Ch.1 interface mode selection	IRMD1[1:0]		I/F mode		X	R/W	
		D0	**IRMD10**		1	1	reserved		X		
					1	0	IrDA 1.0				
					0	1	reserved				
					0	0	General I/F				

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
A/D conversion result (low-order) register	**0040240 (B)**	D7	**ADD7**	A/D converted data	0x0 to 0x3FF				0	R	
		D6	**ADD6**	(low-order 8 bits)	(low-order 8 bits)				0		
		D5	**ADD5**	ADD0 = LSB					0		
		D4	**ADD4**						0		
		D3	**ADD3**						0		
		D2	**ADD2**						0		
		D1	**ADD1**						0		
		D0	**ADD0**						0		
A/D conversion result (high-order) register	**0040241 (B)**	D7–2	**–**	–	–				–	–	0 when being read.
		D1	**ADD9**	A/D converted data	0x0 to 0x3FF				0	R	
		D0	**ADD8**	(high-order 2 bits) ADD9 = MSB	(high-order 2 bits)				0		
A/D trigger register	**0040242 (B)**	D7–6	**–**	–	–				–	–	0 when being read.
		D5	**MS**	A/D conversion mode selection	1	Continuous	0	Normal	0	R/W	
		D4	**TS1**	A/D conversion trigger selection	TS[1:0]			Trigger	0	R/W	
		D3	**TS0**		1	1		#ADTRG pin	0		
					1	0		8-bit timer 0			
					0	1		16-bit timer 0			
					0	0		Software			
		D2	**CH2**	A/D conversion channel status	CH[2:0]			Channel	0	R	
		D1	**CH1**		1	1	1	AD7	0		
		D0	**CH0**		1	1	0	AD6	0		
					1	0	1	AD5			
					1	0	0	AD4			
					0	1	1	AD3			
					0	1	0	AD2			
					0	0	1	AD1			
					0	0	0	AD0			
A/D channel register	**0040243 (B)**	D7–6	**–**	–	–				–	–	0 when being read.
		D5	**CE2**	A/D converter	CE[2:0]			End channel	0	R/W	
		D4	**CE1**	end channel selection	1	1	1	AD7	0		
		D3	**CE0**		1	1	0	AD6	0		
					1	0	1	AD5			
					1	0	0	AD4			
					0	1	1	AD3			
					0	1	0	AD2			
					0	0	1	AD1			
					0	0	0	AD0			
		D2	**CS2**	A/D converter	CS[2:0]			Start channel	0	R/W	
		D1	**CS1**	start channel selection	1	1	1	AD7	0		
		D0	**CS0**		1	1	0	AD6	0		
					1	0	1	AD5			
					1	0	0	AD4			
					0	1	1	AD3			
					0	1	0	AD2			
					0	0	1	AD1			
					0	0	0	AD0			
A/D enable register	**0040244 (B)**	D7–4	**–**	–	–				–	–	0 when being read.
		D3	**ADF**	Conversion-complete flag	1	Completed	0	Run/Standby	0	R	Reset when ADD is read.
		D2	**ADE**	A/D enable	1	Enabled	0	Disabled	0	R/W	
		D1	**ADST**	A/D conversion control/status	1	Start/Run	0	Stop	0	R/W	
		D0	**OWE**	Overwrite error flag	1	Error	0	Normal	0	R/W	Reset by writing 0.
A/D sampling register	**0040245 (B)**	D7–2	**–**	–	–				–	–	0 when being read.
		D1	**ST1**	Input signal sampling time setup	ST[1:0]			Sampring time	1	R/W	Use with 9 clocks.
		D0	**ST0**		1	1		9 clocks	1		
					1	0		7 clocks			
					0	1		5 clocks			
					0	0		3 clocks			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
Port input 0/1 interrupt priority register	**0040260 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**PP1L2**	Port input 1 interrupt level	0 to 7	X	R/W	
		D5	**PP1L1**			X		
		D4	**PP1L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PP0L2**	Port input 0 interrupt level	0 to 7	X	R/W	
		D1	**PP0L1**			X		
		D0	**PP0L0**			X		
Port input 2/3 interrupt priority register	**0040261 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**PP3L2**	Port input 3 interrupt level	0 to 7	X	R/W	
		D5	**PP3L1**			X		
		D4	**PP3L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PP2L2**	Port input 2 interrupt level	0 to 7	X	R/W	
		D1	**PP2L1**			X		
		D0	**PP2L0**			X		
Key input interrupt priority register	**0040262 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**PK1L2**	Key input 1 interrupt level	0 to 7	X	R/W	
		D5	**PK1L1**			X		
		D4	**PK1L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PK0L2**	Key input 0 interrupt level	0 to 7	X	R/W	
		D1	**PK0L1**			X		
		D0	**PK0L0**			X		
High-speed DMA Ch.0/1 interrupt priority register	**0040263 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**PHSD1L2**	High-speed DMA Ch.1 interrupt level	0 to 7	X	R/W	
		D5	**PHSD1L1**			X		
		D4	**PHSD1L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PHSD0L2**	High-speed DMA Ch.0 interrupt level	0 to 7	X	R/W	
		D1	**PHSD0L1**			X		
		D0	**PHSD0L0**			X		
High-speed DMA Ch.2/3 interrupt priority register	**0040264 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**PHSD3L2**	High-speed DMA Ch.3 interrupt level	0 to 7	X	R/W	
		D5	**PHSD3L1**			X		
		D4	**PHSD3L0**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**PHSD2L2**	High-speed DMA Ch.2 interrupt level	0 to 7	X	R/W	
		D1	**PHSD2L1**			X		
		D0	**PHSD2L0**			X		
IDMA interrupt priority register	**0040265 (B)**	D7–3	**–**	reserved	–	–	–	0 when being read.
		D2	**PDM2**	IDMA interrupt level	0 to 7	X	R/W	
		D1	**PDM1**			X		
		D0	**PDM0**			X		
16-bit timer 0/1 interrupt priority register	**0040266 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**P16T12**	16-bit timer 1 interrupt level	0 to 7	X	R/W	
		D5	**P16T11**			X		
		D4	**P16T10**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**P16T02**	16-bit timer 0 interrupt level	0 to 7	X	R/W	
		D1	**P16T01**			X		
		D0	**P16T00**			X		
16-bit timer 2/3 interrupt priority register	**0040267 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**P16T32**	16-bit timer 3 interrupt level	0 to 7	X	R/W	
		D5	**P16T31**			X		
		D4	**P16T30**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**P16T22**	16-bit timer 2 interrupt level	0 to 7	X	R/W	
		D1	**P16T21**			X		
		D0	**P16T20**			X		
16-bit timer 4/5 interrupt priority register	**0040268 (B)**	D7	**–**	reserved	–	–	–	0 when being read.
		D6	**P16T52**	16-bit timer 5 interrupt level	0 to 7	X	R/W	
		D5	**P16T51**			X		
		D4	**P16T50**			X		
		D3	**–**	reserved	–	–	–	0 when being read.
		D2	**P16T42**	16-bit timer 4 interrupt level	0 to 7	X	R/W	
		D1	**P16T41**			X		
		D0	**P16T40**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
8-bit timer, serial I/F Ch.0 interrupt priority register	**0040269 (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PSIO02**	Serial interface Ch.0 interrupt level	0 to 7	X	R/W	
		D5	**PSIO01**			X		
		D4	**PSIO00**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**P8TM2**	8-bit timer 0–3 interrupt level	0 to 7	X	R/W	
		D1	**P8TM1**			X		
		D0	**P8TM0**			X		
Serial I/F Ch.1, A/D interrupt priority register	**004026A (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PAD2**	A/D converter interrupt level	0 to 7	X	R/W	
		D5	**PAD1**			X		
		D4	**PAD0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PSIO12**	Serial interface Ch.1 interrupt level	0 to 7	X	R/W	
		D1	**PSIO11**			X		
		D0	**PSIO10**			X		
Clock timer interrupt priority register	**004026B (B)**	D7–3	–	reserved	–	–	–	Writing 1 not allowed.
		D2	**PCTM2**	Clock timer interrupt level	0 to 7	X	R/W	
		D1	**PCTM1**			X		
		D0	**PCTM0**			X		
Port input 4/5 interrupt priority register	**004026C (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PP5L2**	Port input 5 interrupt level	0 to 7	X	R/W	
		D5	**PP5L1**			X		
		D4	**PP5L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PP4L2**	Port input 4 interrupt level	0 to 7	X	R/W	
		D1	**PP4L1**			X		
		D0	**PP4L0**			X		
Port input 6/7 interrupt priority register	**004026D (B)**	D7	–	reserved	–	–	–	0 when being read.
		D6	**PP7L2**	Port input 7 interrupt level	0 to 7	X	R/W	
		D5	**PP7L1**			X		
		D4	**PP7L0**			X		
		D3	–	reserved	–	–	–	0 when being read.
		D2	**PP6L2**	Port input 6 interrupt level	0 to 7	X	R/W	
		D1	**PP6L1**			X		
		D0	**PP6L0**			X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Key input, port input 0–3 interrupt enable register	**0040270 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**EK1**	Key input 1	1	Enabled	0	Disabled	0	R/W	
		D4	**EK0**	Key input 0					0	R/W	
		D3	**EP3**	Port input 3					0	R/W	
		D2	**EP2**	Port input 2					0	R/W	
		D1	**EP1**	Port input 1					0	R/W	
		D0	**EP0**	Port input 0					0	R/W	
DMA interrupt enable register	**0040271 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**EIDMA**	IDMA	1	Enabled	0	Disabled	0	R/W	
		D3	**EHDM3**	High-speed DMA Ch.3					0	R/W	
		D2	**EHDM2**	High-speed DMA Ch.2					0	R/W	
		D1	**EHDM1**	High-speed DMA Ch.1					0	R/W	
		D0	**EHDM0**	High-speed DMA Ch.0					0	R/W	
16-bit timer 0/1 interrupt enable register	**0040272 (B)**	D7	**E16TC1**	16-bit timer 1 comparison A	1	Enabled	0	Disabled	0	R/W	
		D6	**E16TU1**	16-bit timer 1 comparison B					0	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**E16TC0**	16-bit timer 0 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU0**	16-bit timer 0 comparison B					0	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
16-bit timer 2/3 interrupt enable register	**0040273 (B)**	D7	**E16TC3**	16-bit timer 3 comparison A	1	Enabled	0	Disabled	0	R/W	
		D6	**E16TU3**	16-bit timer 3 comparison B					0	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**E16TC2**	16-bit timer 2 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
16-bit timer 4/5 interrupt enable register	**0040274 (B)**	D7	**E16TC5**	16-bit timer 5 comparison A	1	Enabled	0	Disabled	0	R/W	
		D6	**E16TU5**	16-bit timer 5 comparison B					0	R/W	
		D5–4	**–**	reserved	–				–	–	0 when being read.
		D3	**E16TC4**	16-bit timer 4 comparison A	1	Enabled	0	Disabled	0	R/W	
		D2	**E16TU4**	16-bit timer 4 comparison B					0	R/W	
		D1–0	**–**	reserved	–				–	–	0 when being read.
8-bit timer interrupt enable register	**0040275 (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**E8TU3**	8-bit timer 3 underflow	1	Enabled	0	Disabled	0	R/W	
		D2	**E8TU2**	8-bit timer 2 underflow					0	R/W	
		D1	**E8TU1**	8-bit timer 1 underflow					0	R/W	
		D0	**E8TU0**	8-bit timer 0 underflow					0	R/W	
Serial I/F interrupt enable register	**0040276 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**ESTX1**	SIF Ch.1 transmit buffer empty	1	Enabled	0	Disabled	0	R/W	
		D4	**ESRX1**	SIF Ch.1 receive buffer full					0	R/W	
		D3	**ESERR1**	SIF Ch.1 receive error					0	R/W	
		D2	**ESTX0**	SIF Ch.0 transmit buffer empty					0	R/W	
		D1	**ESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D0	**ESERR0**	SIF Ch.0 receive error					0	R/W	
Port input 4–7, clock timer, A/D interrupt enable register	**0040277 (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**EP7**	Port input 7	1	Enabled	0	Disabled	0	R/W	
		D4	**EP6**	Port input 6					0	R/W	
		D3	**EP5**	Port input 5					0	R/W	
		D2	**EP4**	Port input 4					0	R/W	
		D1	**ECTM**	Clock timer					0	R/W	
		D0	**EADE**	A/D converter					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Key input, port input 0–3 interrupt factor flag register	**0040280 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**FK1**	Key input 1	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FK0**	Key input 0					X	R/W	
		D3	**FP3**	Port input 3					X	R/W	
		D2	**FP2**	Port input 2					X	R/W	
		D1	**FP1**	Port input 1					X	R/W	
		D0	**FP0**	Port input 0					X	R/W	
DMA interrupt factor flag register	**0040281 (B)**	D7–5	–	reserved	–				–	–	0 when being read.
		D4	**FIDMA**	IDMA	1	Factor is generated	0	No factor is generated	X	R/W	
		D3	**FHDM3**	High-speed DMA Ch.3					X	R/W	
		D2	**FHDM2**	High-speed DMA Ch.2					X	R/W	
		D1	**FHDM1**	High-speed DMA Ch.1					X	R/W	
		D0	**FHDM0**	High-speed DMA Ch.0					X	R/W	
16-bit timer 0/1 interrupt factor flag register	**0040282 (B)**	D7	**F16TC1**	16-bit timer 1 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU1**	16-bit timer 1 comparison B					X	R/W	
		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**F16TC0**	16-bit timer 0 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU0**	16-bit timer 0 comparison B					X	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.
16-bit timer 2/3 interrupt factor flag register	**0040283 (B)**	D7	**F16TC3**	16-bit timer 3 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU3**	16-bit timer 3 comparison B					X	R/W	
		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**F16TC2**	16-bit timer 2 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU2**	16-bit timer 2 comparison B					X	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.
16-bit timer 4/5 interrupt factor flag register	**0040284 (B)**	D7	**F16TC5**	16-bit timer 5 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D6	**F16TU5**	16-bit timer 5 comparison B					X	R/W	
		D5–4	–	reserved	–				–	–	0 when being read.
		D3	**F16TC4**	16-bit timer 4 comparison A	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F16TU4**	16-bit timer 4 comparison B					X	R/W	
		D1–0	–	reserved	–				–	–	0 when being read.
8-bit timer interrupt factor flag register	**0040285 (B)**	D7–4	–	reserved	–				–	–	0 when being read.
		D3	**F8TU3**	8-bit timer 3 underflow	1	Factor is generated	0	No factor is generated	X	R/W	
		D2	**F8TU2**	8-bit timer 2 underflow					X	R/W	
		D1	**F8TU1**	8-bit timer 1 underflow					X	R/W	
		D0	**F8TU0**	8-bit timer 0 underflow					X	R/W	
Serial I/F interrupt factor flag register	**0040286 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**FSTX1**	SIF Ch.1 transmit buffer empty	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FSRX1**	SIF Ch.1 receive buffer full					X	R/W	
		D3	**FSERR1**	SIF Ch.1 receive error					X	R/W	
		D2	**FSTX0**	SIF Ch.0 transmit buffer empty					X	R/W	
		D1	**FSRX0**	SIF Ch.0 receive buffer full					X	R/W	
		D0	**FSERR0**	SIF Ch.0 receive error					X	R/W	
Port input 4–7, clock timer, A/D interrupt factor flag register	**0040287 (B)**	D7–6	–	reserved	–				–	–	0 when being read.
		D5	**FP7**	Port input 7	1	Factor is generated	0	No factor is generated	X	R/W	
		D4	**FP6**	Port input 6					X	R/W	
		D3	**FP5**	Port input 5					X	R/W	
		D2	**FP4**	Port input 4					X	R/W	
		D1	**FCTM**	Clock timer					X	R/W	
		D0	**FADE**	A/D converter					X	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA request register	**0040290 (B)**	D7	**R16TC0**	16-bit timer 0 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**RHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**RHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**RP3**	Port input 3					0	R/W	
		D2	**RP2**	Port input 2					0	R/W	
		D1	**RP1**	Port input 1					0	R/W	
		D0	**RP0**	Port input 0					0	R/W	
16-bit timer 1–4 IDMA request register	**0040291 (B)**	D7	**R16TC4**	16-bit timer 4 comparison A	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**R16TU4**	16-bit timer 4 comparison B					0	R/W	
		D5	**R16TC3**	16-bit timer 3 comparison A					0	R/W	
		D4	**R16TU3**	16-bit timer 3 comparison B					0	R/W	
		D3	**R16TC2**	16-bit timer 2 comparison A					0	R/W	
		D2	**R16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1	**R16TC1**	16-bit timer 1 comparison A					0	R/W	
		D0	**R16TU1**	16-bit timer 1 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA request register	**0040292 (B)**	D7	**RSTX0**	SIF Ch.0 transmit buffer empty	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RSRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**R8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**R8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**R8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**R8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**R16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**R16TU5**	16-bit timer 5 comparison B					0	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA request register	**0040293 (B)**	D7	**RP7**	Port input 7	1	IDMA request	0	Interrupt request	0	R/W	
		D6	**RP6**	Port input 6					0	R/W	
		D5	**RP5**	Port input 5					0	R/W	
		D4	**RP4**	Port input 4					0	R/W	
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**RADE**	A/D converter	1	IDMA request	0	Interrupt request	0	R/W	
		D1	**RSTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**RSRX1**	SIF Ch.1 receive buffer full					0	R/W	
Port input 0–3, high-speed DMA, 16-bit timer 0 IDMA enable register	**0040294 (B)**	D7	**DE16TC0**	16-bit timer 0 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU0**	16-bit timer 0 comparison B					0	R/W	
		D5	**DEHDM1**	High-speed DMA Ch.1					0	R/W	
		D4	**DEHDM0**	High-speed DMA Ch.0					0	R/W	
		D3	**DEP3**	Port input 3					0	R/W	
		D2	**DEP2**	Port input 2					0	R/W	
		D1	**DEP1**	Port input 1					0	R/W	
		D0	**DEP0**	Port input 0					0	R/W	
16-bit timer 1–4 IDMA enable register	**0040295 (B)**	D7	**DE16TC4**	16-bit timer 4 comparison A	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DE16TU4**	16-bit timer 4 comparison B					0	R/W	
		D5	**DE16TC3**	16-bit timer 3 comparison A					0	R/W	
		D4	**DE16TU3**	16-bit timer 3 comparison B					0	R/W	
		D3	**DE16TC2**	16-bit timer 2 comparison A					0	R/W	
		D2	**DE16TU2**	16-bit timer 2 comparison B					0	R/W	
		D1	**DE16TC1**	16-bit timer 1 comparison A					0	R/W	
		D0	**DE16TU1**	16-bit timer 1 comparison B					0	R/W	
16-bit timer 5, 8-bit timer, serial I/F Ch.0 IDMA enable register	**0040296 (B)**	D7	**DESTX0**	SIF Ch.0 transmit buffer empty	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DESRX0**	SIF Ch.0 receive buffer full					0	R/W	
		D5	**DE8TU3**	8-bit timer 3 underflow					0	R/W	
		D4	**DE8TU2**	8-bit timer 2 underflow					0	R/W	
		D3	**DE8TU1**	8-bit timer 1 underflow					0	R/W	
		D2	**DE8TU0**	8-bit timer 0 underflow					0	R/W	
		D1	**DE16TC5**	16-bit timer 5 comparison A					0	R/W	
		D0	**DE16TU5**	16-bit timer 5 comparison B					0	R/W	
Serial I/F Ch.1, A/D, port input 4–7 IDMA enable register	**0040297 (B)**	D7	**DEP7**	Port input 7	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D6	**DEP6**	Port input 6					0	R/W	
		D5	**DEP5**	Port input 5					0	R/W	
		D4	**DEP4**	Port input 4					0	R/W	
		D3	**–**	reserved	–				–	–	0 when being read.
		D2	**DEADE**	A/D converter	1	IDMA enabled	0	IDMA disabled	0	R/W	
		D1	**DESTX1**	SIF Ch.1 transmit buffer empty					0	R/W	
		D0	**DESRX1**	SIF Ch.1 receive buffer full					0	R/W	

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed DMA Ch.0/1 trigger set-up register	**0040298 (B)**	D7	**HSD1S3**	High-speed DMA Ch.1 trigger set-up	0 Software trigger	0	R/W	
		D6	**HSD1S2**		1 K51 input (falling edge)	0		
		D5	**HSD1S1**		2 K51 input (rising edge)	0		
		D4	**HSD1S0**		3 Port 1 input	0		
					4 Port 5 input			
					5 8-bit timer Ch.1 underflow			
					6 16-bit timer Ch.1 compare B			
					7 16-bit timer Ch.1 compare A			
					8 16-bit timer Ch.5 compare B			
					9 16-bit timer Ch.5 compare A			
					A SI/F Ch.1 Rx buffer full			
					B SI/F Ch.1 Tx buffer empty			
					C A/D conversion completion			
		D3	**HSD0S3**	High-speed DMA Ch.0 trigger set-up	0 Software trigger	0	R/W	
		D2	**HSD0S2**		1 K50 input (falling edge)	0		
		D1	**HSD0S1**		2 K50 input (rising edge)	0		
		D0	**HSD0S0**		3 Port 0 input	0		
					4 Port 4 input			
					5 8-bit timer Ch.0 underflow			
					6 16-bit timer Ch.0 compare B			
					7 16-bit timer Ch.0 compare A			
					8 16-bit timer Ch.4 compare B			
					9 16-bit timer Ch.4 compare A			
					A SI/F Ch.0 Rx buffer full			
					B SI/F Ch.0 Tx buffer empty			
					C A/D conversion completion			
High-speed DMA Ch.2/3 trigger set-up register	**0040299 (B)**	D7	**HSD3S3**	High-speed DMA Ch.3 trigger set-up	0 Software trigger	0	R/W	
		D6	**HSD3S2**		1 K54 input (falling edge)	0		
		D5	**HSD3S1**		2 K54 input (rising edge)	0		
		D4	**HSD3S0**		3 Port 3 input	0		
					4 Port 7 input			
					5 8-bit timer Ch.3 underflow			
					6 16-bit timer Ch.3 compare B			
					7 16-bit timer Ch.3 compare A			
					8 16-bit timer Ch.5 compare B			
					9 16-bit timer Ch.5 compare A			
					A SI/F Ch.1 Rx buffer full			
					B SI/F Ch.1 Tx buffer empty			
					C A/D conversion completion			
		D3	**HSD2S3**	High-speed DMA Ch.2 trigger set-up	0 Software trigger	0	R/W	
		D2	**HSD2S2**		1 K53 input (falling edge)	0		
		D1	**HSD2S1**		2 K53 input (rising edge)	0		
		D0	**HSD2S0**		3 Port 2 input	0		
					4 Port 6 input			
					5 8-bit timer Ch.2 underflow			
					6 16-bit timer Ch.2 compare B			
					7 16-bit timer Ch.2 compare A			
					8 16-bit timer Ch.4 compare B			
					9 16-bit timer Ch.4 compare A			
					A SI/F Ch.0 Rx buffer full			
					B SI/F Ch.0 Tx buffer empty			
					C A/D conversion completion			
High-speed DMA software trigger register	**004029A (B)**	D7–4	–	reserved	–	–	–	0 when being read.
		D3	**HST3**	HSDMA Ch.3 software trigger	1 Trigger / 0 Invalid	0	W	
		D2	**HST2**	HSDMA Ch.2 software trigger		0	W	
		D1	**HST1**	HSDMA Ch.1 software trigger		0	W	
		D0	**HST0**	HSDMA Ch.0 software trigger		0	W	
Flag set/reset method select register	**004029F (B)**	D7–3	–	reserved	–	–	–	
		D2	**DENONLY**	IDMA enable register set method selection	1 Set only / 0 RD/WR	1	R/W	
		D1	**IDMAONLY**	IDMA request register set method selection	1 Set only / 0 RD/WR	1	R/W	
		D0	**RSTONLY**	Interrupt factor flag reset method selection	1 Reset only / 0 RD/WR	1	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
K5 function select register	**00402C0 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**CFK54**	K54 function selection	1	#DMAREQ3	0	K54	0	R/W	
		D3	**CFK53**	K53 function selection	1	#DMAREQ2	0	K53	0	R/W	
		D2	**CFK52**	K52 function selection	1	#ADTRG	0	K52	0	R/W	
		D1	**CFK51**	K51 function selection	1	#DMAREQ1	0	K51	0	R/W	
		D0	**CFK50**	K50 function selection	1	#DMAREQ0	0	K50	0	R/W	
K5 input port data register	**00402C1 (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**K54D**	K54 input port data	1	High	0	Low	–	R	
		D3	**K53D**	K53 input port data					–	R	
		D2	**K52D**	K52 input port data					–	R	
		D1	**K51D**	K51 input port data					–	R	
		D0	**K50D**	K50 input port data					–	R	
K6 function select register	**00402C3 (B)**	D7	**CFK67**	K67 function selection	1	AD7	0	K67	0	R/W	
		D6	**CFK66**	K66 function selection	1	AD6	0	K66	0	R/W	
		D5	**CFK65**	K65 function selection	1	AD5	0	K65	0	R/W	
		D4	**CFK64**	K64 function selection	1	AD4	0	K64	0	R/W	
		D3	**CFK63**	K63 function selection	1	AD3	0	K63	0	R/W	
		D2	**CFK62**	K62 function selection	1	AD2	0	K62	0	R/W	
		D1	**CFK61**	K61 function selection	1	AD1	0	K61	0	R/W	
		D0	**CFK60**	K60 function selection	1	AD0	0	K60	0	R/W	
K6 input port data register	**00402C4 (B)**	D7	**K67D**	K67 input port data	1	High	0	Low	–	R	
		D6	**K66D**	K66 input port data					–	R	
		D5	**K65D**	K65 input port data					–	R	
		D4	**K64D**	K64 input port data					–	R	
		D3	**K63D**	K63 input port data					–	R	
		D2	**K62D**	K62 input port data					–	R	
		D1	**K61D**	K61 input port data					–	R	
		D0	**K60D**	K60 input port data					–	R	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Port input interrupt select register 1	**00402C6 (B)**	D7	**SPT31**	FPT3 interrupt input port selection	11	10	01	00	0	R/W	
		D6	**SPT30**		P23	P03	K53	K63	0		
		D5	**SPT21**	FPT2 interrupt input port selection	11	10	01	00	0	R/W	
		D4	**SPT20**		P22	P02	K52	K62	0		
		D3	**SPT11**	FPT1 interrupt input port selection	11	10	01	00	0	R/W	
		D2	**SPT10**		P21	P01	K51	K61	0		
		D1	**SPT01**	FPT0 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPT00**		P20	P00	K50	K60	0		
Port input interrupt select register 2	**00402C7 (B)**	D7	**SPT71**	FPT7 interrupt input port selection	11	10	01	00	0	R/W	
		D6	**SPT70**		P27	P07	P33	K67	0		
		D5	**SPT61**	FPT6 interrupt input port selection	11	10	01	00	0	R/W	
		D4	**SPT60**		P26	P06	P32	K66	0		
		D3	**SPT51**	FPT5 interrupt input port selection	11	10	01	00	0	R/W	
		D2	**SPT50**		P25	P05	P31	K65	0		
		D1	**SPT41**	FPT4 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPT40**		P24	P04	K54	K64	0		
Port input interrupt input polarity select register	**00402C8 (B)**	D7	**SPPT7**	FPT7 input polarity selection	1	High level or Rising edge	0	Low level or Falling edge	1	R/W	
		D6	**SPPT6**	FPT6 input polarity selection					1	R/W	
		D5	**SPPT5**	FPT5 input polarity selection					1	R/W	
		D4	**SPPT4**	FPT4 input polarity selection					1	R/W	
		D3	**SPPT3**	FPT3 input polarity selection					1	R/W	
		D2	**SPPT2**	FPT2 input polarity selection					1	R/W	
		D1	**SPPT1**	FPT1 input polarity selection					1	R/W	
		D0	**SPPT0**	FPT0 input polarity selection					1	R/W	
Port input interrupt edge/level select register	**00402C9 (B)**	D7	**SEPT7**	FPT7 edge/level selection	1	Edge	0	Level	1	R/W	
		D6	**SEPT6**	FPT6 edge/level selection					1	R/W	
		D5	**SEPT5**	FPT5 edge/level selection					1	R/W	
		D4	**SEPT4**	FPT4 edge/level selection					1	R/W	
		D3	**SEPT3**	FPT3 edge/level selection					1	R/W	
		D2	**SEPT2**	FPT2 edge/level selection					1	R/W	
		D1	**SEPT1**	FPT1 edge/level selection					1	R/W	
		D0	**SEPT0**	FPT0 edge/level selection					1	R/W	
Key input interrupt select register	**00402CA (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**SPPK11**	FPK1 interrupt input port selection	11	10	01	00	0	R/W	
		D2	**SPPK10**		P2[7:4]	P0[7:4]	K6[7:4]	K6[3:0]	0		
		D1	**SPPK01**	FPK0 interrupt input port selection	11	10	01	00	0	R/W	
		D0	**SPPK00**		P2[4:0]	P0[4:0]	K6[4:0]	K5[4:0]	0		
Key input interrupt (FPK0) input comparison register	**00402CC (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**SCPK04**	FPK04 input comparison	1	High	0	Low	0	R/W	
		D3	**SCPK03**	FPK03 input comparison					0	R/W	
		D2	**SCPK02**	FPK02 input comparison					0	R/W	
		D1	**SCPK01**	FPK01 input comparison					0	R/W	
		D0	**SCPK00**	FPK00 input comparison					0	R/W	
Key input interrupt (FPK1) input comparison register	**00402CD (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**SCPK13**	FPK13 input comparison	1	High	0	Low	0	R/W	
		D2	**SCPK12**	FPK12 input comparison					0	R/W	
		D1	**SCPK11**	FPK11 input comparison					0	R/W	
		D0	**SCPK10**	FPK10 input comparison					0	R/W	
Key input interrupt (FPK0) input mask register	**00402CE (B)**	D7–5	**–**	reserved	–				–	–	0 when being read.
		D4	**SMPK04**	FPK04 input mask	1	Interrupt enabled	0	Interrupt disabled	0	R/W	
		D3	**SMPK03**	FPK03 input mask					0	R/W	
		D2	**SMPK02**	FPK02 input mask					0	R/W	
		D1	**SMPK01**	FPK01 input mask					0	R/W	
		D0	**SMPK00**	FPK00 input mask					0	R/W	
Key input interrupt (FPK1) input mask register	**00402CF (B)**	D7–4	**–**	reserved	–				–	–	0 when being read.
		D3	**SMPK13**	FPK13 input mask	1	Interrupt enabled	0	Interrupt disabled	0	R/W	
		D2	**SMPK12**	FPK12 input mask					0	R/W	
		D1	**SMPK11**	FPK11 input mask					0	R/W	
		D0	**SMPK10**	FPK10 input mask					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
P0 function select register	**00402D0 (B)**	D7	**CFP07**	P07 function selection	1	#SRDY1	0	P07	0	R/W	Extended functions (0x402DF)
		D6	**CFP06**	P06 function selection	1	#SCLK1	0	P06	0	R/W	
		D5	**CFP05**	P05 function selection	1	SOUT1	0	P05	0	R/W	
		D4	**CFP04**	P04 function selection	1	SIN1	0	P04	0	R/W	
		D3	**CFP03**	P03 function selection	1	#SRDY0	0	P03	0	R/W	
		D2	**CFP02**	P02 function selection	1	#SCLK0	0	P02	0	R/W	
		D1	**CFP01**	P01 function selection	1	SOUT0	0	P01	0	R/W	
		D0	**CFP00**	P00 function selection	1	SIN0	0	P00	0	R/W	
P0 I/O port data register	**00402D1 (B)**	D7	**P07D**	P07 I/O port data	1	High	0	Low	0	R/W	
		D6	**P06D**	P06 I/O port data					0	R/W	
		D5	**P05D**	P05 I/O port data					0	R/W	
		D4	**P04D**	P04 I/O port data					0	R/W	
		D3	**P03D**	P03 I/O port data					0	R/W	
		D2	**P02D**	P02 I/O port data					0	R/W	
		D1	**P01D**	P01 I/O port data					0	R/W	
		D0	**P00D**	P00 I/O port data					0	R/W	
P0 I/O control register	**00402D2 (B)**	D7	**IOC07**	P07 I/O control	1	Output	0	Input	0	R/W	
		D6	**IOC06**	P06 I/O control					0	R/W	
		D5	**IOC05**	P05 I/O control					0	R/W	
		D4	**IOC04**	P04 I/O control					0	R/W	
		D3	**IOC03**	P03 I/O control					0	R/W	
		D2	**IOC02**	P02 I/O control					0	R/W	
		D1	**IOC01**	P01 I/O control					0	R/W	
		D0	**IOC00**	P00 I/O control					0	R/W	
P1 function select register	**00402D4 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**CFP16**	P16 function selection	1	EXCL5 #DMAEND1	0	P16	0	R/W	
		D5	**CFP15**	P15 function selection	1	EXCL4 #DMAEND0	0	P15	0	R/W	
		D4	**CFP14**	P14 function selection	1	FOSC1	0	P14	0	R/W	Extended functions (0x402DF)
		D3	**CFP13**	P13 function selection	1	EXCL3 T8UF3	0	P13	0	R/W	
		D2	**CFP12**	P12 function selection	1	EXCL2 T8UF2	0	P12	0	R/W	
		D1	**CFP11**	P11 function selection	1	EXCL1 T8UF1	0	P11	0	R/W	
		D0	**CFP10**	P10 function selection	1	EXCL0 T8UF0	0	P10	0	R/W	
P1 I/O port data register	**00402D5 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**P16D**	P16 I/O port data	1	High	0	Low	0	R/W	
		D5	**P15D**	P15 I/O port data					0	R/W	
		D4	**P14D**	P14 I/O port data					0	R/W	
		D3	**P13D**	P13 I/O port data					0	R/W	
		D2	**P12D**	P12 I/O port data					0	R/W	
		D1	**P11D**	P11 I/O port data					0	R/W	
		D0	**P10D**	P10 I/O port data					0	R/W	
P1 I/O control register	**00402D6 (B)**	D7	**–**	reserved	–				–	–	0 when being read.
		D6	**IOC16**	P16 I/O control	1	Output	0	Input	0	R/W	
		D5	**IOC15**	P15 I/O control					0	R/W	
		D4	**IOC14**	P14 I/O control					0	R/W	
		D3	**IOC13**	P13 I/O control					0	R/W	
		D2	**IOC12**	P12 I/O control					0	R/W	
		D1	**IOC11**	P11 I/O control					0	R/W	
		D0	**IOC10**	P10 I/O control					0	R/W	
P2 function select register	**00402D8 (B)**	D7	**CFP27**	P27 function selection	1	TM5	0	P27	0	R/W	
		D6	**CFP26**	P26 function selection	1	TM4	0	P26	0	R/W	
		D5	**CFP25**	P25 function selection	1	TM3	0	P25	0	R/W	
		D4	**CFP24**	P24 function selection	1	TM2	0	P24	0	R/W	
		D3	**CFP23**	P23 function selection	1	TM1	0	P23	0	R/W	
		D2	**CFP22**	P22 function selection	1	TM0	0	P22	0	R/W	
		D1	**CFP21**	P21 function selection	1	#DWE	0	P21	0	R/W	Ext. func.(0x402DF)
		D0	**CFP20**	P20 function selection	1	#DRD	0	P20	0	R/W	
P2 I/O port data register	**00402D9 (B)**	D7	**P27D**	P27 I/O port data	1	High	0	Low	0	R/W	
		D6	**P26D**	P26 I/O port data					0	R/W	
		D5	**P25D**	P25 I/O port data					0	R/W	
		D4	**P24D**	P24 I/O port data					0	R/W	
		D3	**P23D**	P23 I/O port data					0	R/W	
		D2	**P22D**	P22 I/O port data					0	R/W	
		D1	**P21D**	P21 I/O port data					0	R/W	
		D0	**P20D**	P20 I/O port data					0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
P2 I/O control register	**00402DA (B)**	D7	**IOC27**	P27 I/O control	1	Output	0	Input	0	R/W	
		D6	**IOC26**	P26 I/O control					0	R/W	
		D5	**IOC25**	P25 I/O control					0	R/W	
		D4	**IOC24**	P24 I/O control					0	R/W	
		D3	**IOC23**	P23 I/O control					0	R/W	
		D2	**IOC22**	P22 I/O control					0	R/W	
		D1	**IOC21**	P21 I/O control					0	R/W	
		D0	**IOC20**	P20 I/O control					0	R/W	
P3 function select register	**00402DC (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**CFP35**	P35 function selection	1	#BUSACK	0	P35	0	R/W	
		D4	**CFP34**	P34 function selection	1	#BUSREQ #CE6	0	P34	0	R/W	
		D3	**CFP33**	P33 function selection	1	#DMAACK1	0	P33	0	R/W	
		D2	**CFP32**	P32 function selection	1	#DMAACK0	0	P32	0	R/W	
		D1	**CFP31**	P31 function selection	1	#BUSGET	0	P31	0	R/W	Ext. func.(0x402DF)
		D0	**CFP30**	P30 function selection	1	#WAIT #CE4/#CE5	0	P30	0	R/W	
P3 I/O port data register	**00402DD (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**P35D**	P35 I/O port data	1	High	0	Low	0	R/W	
		D4	**P34D**	P34 I/O port data					0	R/W	
		D3	**P33D**	P33 I/O port data					0	R/W	
		D2	**P32D**	P32 I/O port data					0	R/W	
		D1	**P31D**	P31 I/O port data					0	R/W	
		D0	**P30D**	P30 I/O port data					0	R/W	
P3 I/O control register	**00402DE (B)**	D7–6	**–**	reserved	–				–	–	0 when being read.
		D5	**IOC35**	P35 I/O control	1	Output	0	Input	0	R/W	
		D4	**IOC34**	P34 I/O control					0	R/W	
		D3	**IOC33**	P33 I/O control					0	R/W	
		D2	**IOC32**	P32 I/O control					0	R/W	
		D1	**IOC31**	P31 I/O control					0	R/W	
		D0	**IOC30**	P30 I/O control					0	R/W	
Port function extension register	**00402DF (B)**	D7	**CFEX7**	P07 port extended function	1	#DMAEND3	0	P07, etc.	0	R/W	
		D6	**CFEX6**	P06 port extended function	1	#DMAACK3	0	P06, etc.	0	R/W	
		D5	**CFEX5**	P05 port extended function	1	#DMAEND2	0	P05, etc.	0	R/W	
		D4	**CFEX4**	P04 port extended function	1	#DMAACK2	0	P04, etc.	0	R/W	
		D3	**CFEX3**	P31 port extended function	1	#GARD	0	P31, etc.	0	R/W	
		D2	**CFEX2**	P21 port extended function	1	#GAAS	0	P21, etc.	0	R/W	
		D1	**CFEX1**	P10, P11, P13 port extended function	1	DST0 DST1 DPC0	0	P10, etc. P11, etc. P13, etc.	1	R/W	
		D0	**CFEX0**	P12, P14 port extended function	1	DST2 DCLK	0	P12, etc. P14, etc.	1	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Areas 18–15	**0048120**	DF	–	reserved	–				–	–	0 when being read.
set-up register	**(HW)**	DE	**A18SZ**	Areas 18–17 device size selection	1	8 bits	0	16 bits	0	R/W	
		DD	**A18DF1**	Areas 18–17	A18DF[1:0]			Number of cycles	1	R/W	
		DC	**A18DF0**	output disable delay time	1	1		3.5	1		
					1	0		2.5			
					0	1		1.5			
					0	0		0.5			
		DB	–	reserved	–				–	–	0 when being read.
		DA	**A18WT2**	Areas 18–17 wait control	A18WT[2:0]			Wait cycles	1	R/W	
		D9	**A18WT1**		1	1	1	7	1		
		D8	**A18WT0**		1	1	0	6	1		
					1	0	1	5			
					1	0	0	4			
					0	1	1	3			
					0	1	0	2			
					0	0	1	1			
					0	0	0	0			
		D7	–	reserved	–				–	–	0 when being read.
		D6	**A16SZ**	Areas 16–15 device size selection	1	8 bits	0	16 bits	0	R/W	
		D5	**A16DF1**	Areas 16–15	A16DF[1:0]			Number of cycles	1	R/W	
		D4	**A16DF0**	output disable delay time	1	1		3.5	1		
					1	0		2.5			
					0	1		1.5			
					0	0		0.5			
		D3	–	reserved	–				–	–	0 when being read.
		D2	**A16WT2**	Areas 16–15 wait control	A16WT[2:0]			Wait cycles	1	R/W	
		D1	**A16WT1**		1	1	1	7	1		
		D0	**A16WT0**		1	1	0	6	1		
					1	0	1	5			
					1	0	0	4			
					0	1	1	3			
					0	1	0	2			
					0	0	1	1			
					0	0	0	0			
Areas 14–13	**0048122**	DF–9	–	reserved	–				–	–	0 when being read.
set-up register	**(HW)**	D8	**A14DRA**	Area 14 DRAM selection	1	Used	0	Not used	0	R/W	
		D7	**A13DRA**	Area 13 DRAM selection	1	Used	0	Not used	0	R/W	
		D6	**A14SZ**	Areas 14–13 device size selection	1	8 bits	0	16 bits	0	R/W	
		D5	**A14DF1**	Areas 14–13	A14DF[1:0]			Number of cycles	1	R/W	
		D4	**A14DF0**	output disable delay time	1	1		3.5	1		
					1	0		2.5			
					0	1		1.5			
					0	0		0.5			
		D3	–	reserved	–				–	–	0 when being read.
		D2	**A14WT2**	Areas 14–13 wait control	A14WT[2:0]			Wait cycles	1	R/W	
		D1	**A14WT1**		1	1	1	7	1		
		D0	**A14WT0**		1	1	0	6	1		
					1	0	1	5			
					1	0	0	4			
					0	1	1	3			
					0	1	0	2			
					0	0	1	1			
					0	0	0	0			
Areas 12–11	**0048124**	DF–7	–	reserved	–				–	–	0 when being read.
set-up register	**(HW)**	D6	**A12SZ**	Areas 12–11 device size selection	1	8 bits	0	16 bits	0	R/W	
		D5	**A12DF1**	Areas 12–11	A18DF[1:0]			Number of cycles	1	R/W	
		D4	**A12DF0**	output disable delay time	1	1		3.5	1		
					1	0		2.5			
					0	1		1.5			
					0	0		0.5			
		D3	–	reserved	–				–	–	0 when being read.
		D2	**A12WT2**	Areas 12–11 wait control	A18WT[2:0]			Wait cycles	1	R/W	
		D1	**A12WT1**		1	1	1	7	1		
		D0	**A12WT0**		1	1	0	6	1		
					1	0	1	5			
					1	0	0	4			
					0	1	1	3			
					0	1	0	2			
					0	0	1	1			
					0	0	0	0			

Register name	Address	Bit	Name	Function	Setting		Init.	R/W	Remarks
Areas 10–9 set-up register	**0048126 (HW)**	DF	–	reserved	–		–	–	0 when being read.
		DE	**A10IR2**	Area 10 internal ROM wait control	A10IR[2:0]	ROM size	1	R/W	
		DD	**A10IR1**	Area 10 internal ROM size	1 1 1	2MB	1		
		DC	**A10IR0**	selection	1 1 0	1MB	1		
					1 0 1	512KB			
					1 0 0	256KB			
					0 1 1	128KB			
					0 1 0	64KB			
					0 0 1	32KB			
					0 0 0	16KB			
		DB	–	reserved	–		–	–	0 when being read.
		DA	**A10BW1**	Areas 10–9	A10BW[1:0]	Wait cycles	0	R/W	
		D9	**A10BW0**	burst ROM	1 1	3	0		
				burst read cycle wait control	1 0	2			
					0 1	1			
					0 0	0			
		D8	**A10DRA**	Area 10 burst ROM selection	1 Used	0 Not used	0	R/W	
		D7	**A9DRA**	Area 9 burst ROM selection	1 Used	0 Not used	0	R/W	
		D6	**A10SZ**	Areas 10–9 device size selection	1 8 bits	0 16 bits	0	R/W	
		D5	**A10DF1**	Areas 10–9	A10DF[1:0]	Number of cycles	1	R/W	
		D4	**A10DF0**	output disable delay time	1 1	3.5	1		
					1 0	2.5			
					0 1	1.5			
					0 0	0.5			
		D3	–	reserved	–		–	–	0 when being read.
		D2	**A10WT2**	Areas 10–9 wait control	A10WT[2:0]	Wait cycles	1	R/W	
		D1	**A10WT1**		1 1 1	7	1		
		D0	**A10WT0**		1 1 0	6	1		
					1 0 1	5			
					1 0 0	4			
					0 1 1	3			
					0 1 0	2			
					0 0 1	1			
					0 0 0	0			
Areas 8–7 set-up register	**0048128 (HW)**	DF–9	–	reserved	–		–	–	0 when being read.
		D8	**A8DRA**	Area 8 DRAM selection	1 Used	0 Not used	0	R/W	
		D7	**A7DRA**	Area 7 DRAM selection	1 Used	0 Not used	0	R/W	
		D6	**A8SZ**	Areas 8–7 device size selection	1 8 bits	0 16 bits	0	R/W	
		D5	**A8DF1**	Areas 8–7	A8DF[1:0]	Number of cycles	1	R/W	
		D4	**A8DF0**	output disable delay time	1 1	3.5	1		
					1 0	2.5			
					0 1	1.5			
					0 0	0.5			
		D3	–	reserved	–		–	–	0 when being read.
		D2	**A8WT2**	Areas 8–7 wait control	A8WT[2:0]	Wait cycles	1	R/W	
		D1	**A8WT1**		1 1 1	7	1		
		D0	**A8WT0**		1 1 0	6	1		
					1 0 1	5			
					1 0 0	4			
					0 1 1	3			
					0 1 0	2			
					0 0 1	1			
					0 0 0	0			

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
Areas 6–4	**004812A**	DF–E	–	reserved	–				–	–	0 when being read.
set-up register	**(HW)**	DD	**A6DF1**	Area 6	A6DF[1:0]			Number of cycles	1	R/W	
		DC	**A6DF0**	output disable delay time	1	1		3.5	1		
					1	0		2.5			
					0	1		1.5			
					0	0		0.5			
		DB	–	reserved	–				–	–	0 when being read.
		DA	**A6WT2**	Area 6 wait control	A6WT[2:0]			Wait cycles	1	R/W	
		D9	**A6WT1**		1	1	1	7	1		
		D8	**A6WT0**		1	1	0	6	1		
					1	0	1	5			
					1	0	0	4			
					0	1	1	3			
					0	1	0	2			
					0	0	1	1			
					0	0	0	0			
		D7	–	reserved	–				–	–	0 when being read.
		D6	**A5SZ**	Areas 5–4 device size selection	1	8 bits	0	16 bits	0	R/W	
		D5	**A5DF1**	Areas 5–4	A5DF[1:0]			Number of cycles	1	R/W	
		D4	**A5DF0**	output disable delay time	1	1		3.5	1		
					1	0		2.5			
					0	1		1.5			
					0	0		0.5			
		D3	–	reserved					–	–	0 when being read.
		D2	**A5WT2**	Areas 5–4 wait control	A5WT[2:0]			Wait cycles	1	R/W	
		D1	**A5WT1**		1	1	1	7	1		
		D0	**A5WT0**		1	1	0	6	1		
					1	0	1	5			
					1	0	0	4			
					0	1	1	3			
					0	1	0	2			
					0	0	1	1			
					0	0	0	0			
TTBR write	**004812D**	D7	**TBRP7**	TTBR register write protect	Writing 01011001(0x59)				0	W	Undefined in read.
protect register	**(B)**	D6	**TBRP6**		removes the TTBR (0x48134)				0		
		D5	**TBRP5**		write protection.				0		
		D4	**TBRP4**		Writing other data sets the				0		
		D3	**TBRP3**		write protection.				0		
		D2	**TBRP2**						0		
		D1	**TBRP1**						0		
		D0	**TBRP0**						0		
Bus control	**004812E**	DF	**RBCLK**	BCLK output control	1	Fixed at H	0	Enabled	0	R/W	
register	**(HW)**	DE	–	reserved	–				0	–	Writing 1 not allowed.
		DD	**RBST8**	Burst ROM burst mode selection	1	8-successive	0	4-successive	0	R/W	
		DC	**REDO**	DRAM page mode selection	1	EDO	0	Fast page	0	R/W	
		DB	**RCA1**	Column address size selection	RCA[1:0]			Size	0	R/W	
		DA	**RCA0**		1	1		11	0		
					1	0		10			
					0	1		9			
					0	0		8			
		D9	**RPC2**	Refresh enable	1	Enabled	0	Disabled	0	R/W	
		D8	**RPC1**	Refresh method selection	1	Self-refresh	0	CBR-refresh	0	R/W	
		D7	**RPC0**	Refresh RPC delay setup	1	2.0	0	1.0	0	R/W	
		D6	**RRA1**	Refresh RAS pulse width	RRA[1:0]			Number of cycles	0	R/W	
		D5	**RRA0**	selection	1	1		5	0		
					1	0		4			
					0	1		3			
					0	0		2			
		D4	–	reserved	–				0	–	Writing 1 not allowed.
		D3	**SBUSST**	External interface method selection	1	#BSL	0	A0	0	R/W	
		D2	**SEMAS**	External bus master setup	1	Existing	0	Nonexistent	0	R/W	
		D1	**SEPD**	External power-down control	1	Enabled	0	Disabled	0	R/W	
		D0	**SWAITE**	#WAIT enable	1	Enabled	0	Disabled	0	R/W	

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
DRAM timing set-up register	**0048130 (HW)**	DF–C	–	reserved	–	–	–	0 when being read.
		DB	**A3EEN**	Area 3 emulation	1 Internal ROM / 0 Emulation	1	R/W	
		DA	**CEFUNC1**	#CE pin function selection	CFFUNC[1:0] / #CE output	0	R/W	
		D9	**CEFUNC0**		1 x #CE7/8..#CE17/18	0		
					0 1 #CE6..#CE17			
					0 0 #CE4..#CE10			
		D8	**CRAS**	Successive RAS mode setup	1 Successive / 0 Normal	0	R/W	
		D7	**RPRC1**	DRAM	RPRC[1:0] / Number of cycles	0	R/W	
		D6	**RPRC0**	RAS precharge cycles selection	1 1 4	0		
					1 0 3			
					0 1 2			
					0 0 1			
		D5	–	reserved	–	–	–	0 when being read.
		D4	**CASC1**	DRAM	CASC[1:0] / Number of cycles	0	R/W	
		D3	**CASC0**	CAS cycles selection	1 1 4	0		
					1 0 3			
					0 1 2			
					0 0 1			
		D2	–	reserved	–	–	–	0 when being read.
		D1	**RASC1**	DRAM	RASC[1:0] / Number of cycles	0	R/W	
		D0	**RASC0**	RAS cycles selection	1 1 4	0		
					1 0 3			
					0 1 2			
					0 0 1			
Access control register	**0048132 (HW)**	DF	**A18IO**	Area 18, 17 internal/external access	1 Internal access / 0 External access	0	R/W	
		DE	**A16IO**	Area 16, 15 internal/external access		0	R/W	
		DD	**A14IO**	Area 14, 13 internal/external access		0	R/W	
		DC	**A12IO**	Area 12, 11 internal/external access		0	R/W	
		DB	–	reserved	–	0	–	0 when being read.
		DA	**A8IO**	Area 8, 7 internal/external access	1 Internal access / 0 External access	0	R/W	
		D9	**A6IO**	Area 6 internal/external access		0	R/W	
		D8	**A5IO**	Area 5, 4 internal/external access		0	R/W	
		D7	**A18EC**	Area 18, 17 endian control	1 Big endian / 0 Little endian	0	R/W	
		D6	**A16EC**	Area 16, 15 endian control		0	R/W	
		D5	**A14EC**	Area 14, 13 endian control		0	R/W	
		D4	**A12EC**	Area 12, 11 endian control		0	R/W	
		D3	**A10EC**	Area 10, 9 endian control		0	R/W	
		D2	**A8EC**	Area 8, 7 endian control		0	R/W	
		D1	**A6EC**	Area 6 endian control		0	R/W	
		D0	**A5EC**	Area 5, 4 endian control		0	R/W	
TTBR low-order register	**0048134 (HW)**	DF	**TTBR15**	Trap table base address [15:10]		0	R/W	
		DE	**TTBR14**			0		
		DD	**TTBR13**			0		
		DC	**TTBR12**			0		
		DB	**TTBR11**			0		
		DA	**TTBR10**			0		
		D9	**TTBR09**	Trap table base address [9:0]	Fixed at 0	0	R	0 when being read. Writing 1 not allowed.
		D8	**TTBR08**			0		
		D7	**TTBR07**			0		
		D6	**TTBR06**			0		
		D5	**TTBR05**			0		
		D4	**TTBR04**			0		
		D3	**TTBR03**			0		
		D2	**TTBR02**			0		
		D1	**TTBR01**			0		
		D0	**TTBR00**			0		
TTBR high-order register	**0048136 (HW)**	DF	**TTBR33**	Trap table base address [31:28]	Fixed at 0	0	R	0 when being read. Writing 1 not allowed.
		DE	**TTBR32**			0		
		DD	**TTBR31**			0		
		DC	**TTBR30**			0		
		DB	**TTBR2B**	Trap table base address [27:16]	The initial value is set according to the BTA3 pin status. BTA3 = "1": 0x008 BTA3 = "0": 0x0C0	←	R/W	
		DA	**TTBR2A**					
		D9	**TTBR29**					
		D8	**TTBR28**					
		D7	**TTBR27**					
		D6	**TTBR26**					
		D5	**TTBR25**					
		D4	**TTBR24**					
		D3	**TTBR23**					
		D2	**TTBR22**					
		D1	**TTBR21**					
		D0	**TTBR20**					

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
G/A read signal control register	**0048138 (HW)**	DF	**A18AS**	Area 18, 17 address strobe signal	1	Enabled	0	Disabled	0	R/W	
		DE	**A16AS**	Area 16, 15 address strobe signal					0	R/W	
		DD	**A14AS**	Area 14, 13 address strobe signal					0	R/W	
		DC	**A12AS**	Area 12, 11 address strobe signal					0	R/W	
		DB	**–**	reserved	–				0	–	0 when being read.
		DA	**A8AS**	Area 8, 7 address strobe signal	1	Enabled	0	Disabled	0	R/W	
		D9	**A6AS**	Area 6 address strobe signal					0	R/W	
		D8	**A5AS**	Area 5, 4 address strobe signal					0	R/W	
		D7	**A18RD**	Area 18, 17 read signal	1	Enabled	0	Disabled	0	R/W	
		D6	**A16RD**	Area 16, 15 read signal					0	R/W	
		D5	**A14RD**	Area 14, 13 read signal					0	R/W	
		D4	**A12RD**	Area 12, 11 read signal					0	R/W	
		D3	**–**	reserved	–				0	–	0 when being read.
		D2	**A8RD**	Area 8, 7 read signal	1	Enabled	0	Disabled	0	R/W	
		D1	**A6RD**	Area 6 read signal					0	R/W	
		D0	**A5RD**	Area 5, 4 read signal					0	R/W	
BCLK select register	**004813A (B)**	D7–4	**–**	reserved	–				0	–	0 when being read.
		D3	**A1X1MD**	Area 1 access-speed	1	2 cycles	0	4 cycles	0	R/W	x2 speed mode only
		D2	**–**	reserved	–				0	–	0 when being read.
		D1	**BCLKSEL1**	BCLK output clock selection	BCLKSEL[1:0]		BCLK		0	R/W	
		D0	**BCLKSEL0**		1	1	PLL_CLK		0		
					1	0	OSC3_CLK				
					0	1	BCU_CLK				
					0	0	CPU_CLK				

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 0 comparison register A	**0048180 (HW)**	DF	**CR0A15**	16-bit timer 0 comparison data A	0 to 65535				X	R/W	
		DE	**CR0A14**	CR0A15 = MSB					X		
		DD	**CR0A13**	CR0A0 = LSB					X		
		DC	**CR0A12**						X		
		DB	**CR0A11**						X		
		DA	**CR0A10**						X		
		D9	**CR0A9**						X		
		D8	**CR0A8**						X		
		D7	**CR0A7**						X		
		D6	**CR0A6**						X		
		D5	**CR0A5**						X		
		D4	**CR0A4**						X		
		D3	**CR0A3**						X		
		D2	**CR0A2**						X		
		D1	**CR0A1**						X		
		D0	**CR0A0**						X		
16-bit timer 0 comparison register B	**0048182 (HW)**	DF	**CR0B15**	16-bit timer 0 comparison data B	0 to 65535				X	R/W	
		DE	**CR0B14**	CR0B15 = MSB					X		
		DD	**CR0B13**	CR0B0 = LSB					X		
		DC	**CR0B12**						X		
		DB	**CR0B11**						X		
		DA	**CR0B10**						X		
		D9	**CR0B9**						X		
		D8	**CR0B8**						X		
		D7	**CR0B7**						X		
		D6	**CR0B6**						X		
		D5	**CR0B5**						X		
		D4	**CR0B4**						X		
		D3	**CR0B3**						X		
		D2	**CR0B2**						X		
		D1	**CR0B1**						X		
		D0	**CR0B0**						X		
16-bit timer 0 counter data register	**0048184 (HW)**	DF	**TC015**	16-bit timer 0 counter data	0 to 65535				X	R	
		DE	**TC014**	TC015 = MSB					X		
		DD	**TC013**	TC00 = LSB					X		
		DC	**TC012**						X		
		DB	**TC011**						X		
		DA	**TC010**						X		
		D9	**TC09**						X		
		D8	**TC08**						X		
		D7	**TC07**						X		
		D6	**TC06**						X		
		D5	**TC05**						X		
		D4	**TC04**						X		
		D3	**TC03**						X		
		D2	**TC02**						X		
		D1	**TC01**						X		
		D0	**TC00**						X		
16-bit timer 0 control register	**0048186 (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM0**	16-bit timer 0 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB0**	16-bit timer 0 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV0**	16-bit timer 0 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL0**	16-bit timer 0 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM0**	16-bit timer 0 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET0**	16-bit timer 0 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN0**	16-bit timer 0 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 1 comparison register A	**0048188 (HW)**	DF	**CR1A15**	16-bit timer 1 comparison data A	0 to 65535				X	R/W	
		DE	**CR1A14**	CR1A15 = MSB					X		
		DD	**CR1A13**	CR1A0 = LSB					X		
		DC	**CR1A12**						X		
		DB	**CR1A11**						X		
		DA	**CR1A10**						X		
		D9	**CR1A9**						X		
		D8	**CR1A8**						X		
		D7	**CR1A7**						X		
		D6	**CR1A6**						X		
		D5	**CR1A5**						X		
		D4	**CR1A4**						X		
		D3	**CR1A3**						X		
		D2	**CR1A2**						X		
		D1	**CR1A1**						X		
		D0	**CR1A0**						X		
16-bit timer 1 comparison register B	**004818A (HW)**	DF	**CR1B15**	16-bit timer 1 comparison data B	0 to 65535				X	R/W	
		DE	**CR1B14**	CR1B15 = MSB					X		
		DD	**CR1B13**	CR1B0 = LSB					X		
		DC	**CR1B12**						X		
		DB	**CR1B11**						X		
		DA	**CR1B10**						X		
		D9	**CR1B9**						X		
		D8	**CR1B8**						X		
		D7	**CR1B7**						X		
		D6	**CR1B6**						X		
		D5	**CR1B5**						X		
		D4	**CR1B4**						X		
		D3	**CR1B3**						X		
		D2	**CR1B2**						X		
		D1	**CR1B1**						X		
		D0	**CR1B0**						X		
16-bit timer 1 counter data register	**004818C (HW)**	DF	**TC115**	16-bit timer 1 counter data	0 to 65535				X	R	
		DE	**TC114**	TC115 = MSB					X		
		DD	**TC113**	TC10 = LSB					X		
		DC	**TC112**						X		
		DB	**TC111**						X		
		DA	**TC110**						X		
		D9	**TC19**						X		
		D8	**TC18**						X		
		D7	**TC17**						X		
		D6	**TC16**						X		
		D5	**TC15**						X		
		D4	**TC14**						X		
		D3	**TC13**						X		
		D2	**TC12**						X		
		D1	**TC11**						X		
		D0	**TC10**						X		
16-bit timer 1 control register	**004818E (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM1**	16-bit timer 1 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB1**	16-bit timer 1 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV1**	16-bit timer 1 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL1**	16-bit timer 1 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM1**	16-bit timer 1 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET1**	16-bit timer 1 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN1**	16-bit timer 1 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 2 comparison register A	**0048190 (HW)**	DF	**CR2A15**	16-bit timer 2 comparison data A	0 to 65535				X	R/W	
		DE	**CR2A14**	CR2A15 = MSB					X		
		DD	**CR2A13**	CR2A0 = LSB					X		
		DC	**CR2A12**						X		
		DB	**CR2A11**						X		
		DA	**CR2A10**						X		
		D9	**CR2A9**						X		
		D8	**CR2A8**						X		
		D7	**CR2A7**						X		
		D6	**CR2A6**						X		
		D5	**CR2A5**						X		
		D4	**CR2A4**						X		
		D3	**CR2A3**						X		
		D2	**CR2A2**						X		
		D1	**CR2A1**						X		
		D0	**CR2A0**						X		
16-bit timer 2 comparison register B	**0048192 (HW)**	DF	**CR2B15**	16-bit timer 2 comparison data B	0 to 65535				X	R/W	
		DE	**CR2B14**	CR2B15 = MSB					X		
		DD	**CR2B13**	CR2B0 = LSB					X		
		DC	**CR2B12**						X		
		DB	**CR2B11**						X		
		DA	**CR2B10**						X		
		D9	**CR2B9**						X		
		D8	**CR2B8**						X		
		D7	**CR2B7**						X		
		D6	**CR2B6**						X		
		D5	**CR2B5**						X		
		D4	**CR2B4**						X		
		D3	**CR2B3**						X		
		D2	**CR2B2**						X		
		D1	**CR2B1**						X		
		D0	**CR2B0**						X		
16-bit timer 2 counter data register	**0048194 (HW)**	DF	**TC215**	16-bit timer 2 counter data	0 to 65535				X	R	
		DE	**TC214**	TC215 = MSB					X		
		DD	**TC213**	TC20 = LSB					X		
		DC	**TC212**						X		
		DB	**TC211**						X		
		DA	**TC210**						X		
		D9	**TC29**						X		
		D8	**TC28**						X		
		D7	**TC27**						X		
		D6	**TC26**						X		
		D5	**TC25**						X		
		D4	**TC24**						X		
		D3	**TC23**						X		
		D2	**TC22**						X		
		D1	**TC21**						X		
		D0	**TC20**						X		
16-bit timer 2 control register	**0048196 (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM2**	16-bit timer 2 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB2**	16-bit timer 2 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV2**	16-bit timer 2 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL2**	16-bit timer 2 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM2**	16-bit timer 2 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET2**	16-bit timer 2 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN2**	16-bit timer 2 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 3 comparison register A	**0048198 (HW)**	DF	**CR3A15**	16-bit timer 3 comparison data A	0 to 65535				X	R/W	
		DE	**CR3A14**	CR3A15 = MSB					X		
		DD	**CR3A13**	CR3A0 = LSB					X		
		DC	**CR3A12**						X		
		DB	**CR3A11**						X		
		DA	**CR3A10**						X		
		D9	**CR3A9**						X		
		D8	**CR3A8**						X		
		D7	**CR3A7**						X		
		D6	**CR3A6**						X		
		D5	**CR3A5**						X		
		D4	**CR3A4**						X		
		D3	**CR3A3**						X		
		D2	**CR3A2**						X		
		D1	**CR3A1**						X		
		D0	**CR3A0**						X		
16-bit timer 3 comparison register B	**004819A (HW)**	DF	**CR3B15**	16-bit timer 3 comparison data B	0 to 65535				X	R/W	
		DE	**CR3B14**	CR3B15 = MSB					X		
		DD	**CR3B13**	CR3B0 = LSB					X		
		DC	**CR3B12**						X		
		DB	**CR3B11**						X		
		DA	**CR3B10**						X		
		D9	**CR3B9**						X		
		D8	**CR3B8**						X		
		D7	**CR3B7**						X		
		D6	**CR3B6**						X		
		D5	**CR3B5**						X		
		D4	**CR3B4**						X		
		D3	**CR3B3**						X		
		D2	**CR3B2**						X		
		D1	**CR3B1**						X		
		D0	**CR3B0**						X		
16-bit timer 3 counter data register	**004819C (HW)**	DF	**TC315**	16-bit timer 3 counter data	0 to 65535				X	R	
		DE	**TC314**	TC315 = MSB					X		
		DD	**TC313**	TC30 = LSB					X		
		DC	**TC312**						X		
		DB	**TC311**						X		
		DA	**TC310**						X		
		D9	**TC39**						X		
		D8	**TC38**						X		
		D7	**TC37**						X		
		D6	**TC36**						X		
		D5	**TC35**						X		
		D4	**TC34**						X		
		D3	**TC33**						X		
		D2	**TC32**						X		
		D1	**TC31**						X		
		D0	**TC30**						X		
16-bit timer 3 control register	**004819E (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM3**	16-bit timer 3 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB3**	16-bit timer 3 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV3**	16-bit timer 3 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL3**	16-bit timer 3 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM3**	16-bit timer 3 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET3**	16-bit timer 3 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN3**	16-bit timer 3 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 4 comparison register A	**00481A0 (HW)**	DF	**CR4A15**	16-bit timer 4 comparison data A	0 to 65535				X	R/W	
		DE	**CR4A14**	CR4A15 = MSB					X		
		DD	**CR4A13**	CR4A0 = LSB					X		
		DC	**CR4A12**						X		
		DB	**CR4A11**						X		
		DA	**CR4A10**						X		
		D9	**CR4A9**						X		
		D8	**CR4A8**						X		
		D7	**CR4A7**						X		
		D6	**CR4A6**						X		
		D5	**CR4A5**						X		
		D4	**CR4A4**						X		
		D3	**CR4A3**						X		
		D2	**CR4A2**						X		
		D1	**CR4A1**						X		
		D0	**CR4A0**						X		
16-bit timer 4 comparison register B	**00481A2 (HW)**	DF	**CR4B15**	16-bit timer 4 comparison data B	0 to 65535				X	R/W	
		DE	**CR4B14**	CR4B15 = MSB					X		
		DD	**CR4B13**	CR4B0 = LSB					X		
		DC	**CR4B12**						X		
		DB	**CR4B11**						X		
		DA	**CR4B10**						X		
		D9	**CR4B9**						X		
		D8	**CR4B8**						X		
		D7	**CR4B7**						X		
		D6	**CR4B6**						X		
		D5	**CR4B5**						X		
		D4	**CR4B4**						X		
		D3	**CR4B3**						X		
		D2	**CR4B2**						X		
		D1	**CR4B1**						X		
		D0	**CR4B0**						X		
16-bit timer 4 counter data register	**00481A4 (HW)**	DF	**TC415**	16-bit timer 4 counter data	0 to 65535				X	R	
		DE	**TC414**	TC415 = MSB					X		
		DD	**TC413**	TC40 = LSB					X		
		DC	**TC412**						X		
		DB	**TC411**						X		
		DA	**TC410**						X		
		D9	**TC49**						X		
		D8	**TC48**						X		
		D7	**TC47**						X		
		D6	**TC46**						X		
		D5	**TC45**						X		
		D4	**TC44**						X		
		D3	**TC43**						X		
		D2	**TC42**						X		
		D1	**TC41**						X		
		D0	**TC40**						X		
16-bit timer 4 control register	**00481A6 (B)**	D7	**–**	reserved	–				0	–	0 when being read.
		D6	**SELFM4**	16-bit timer 4 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB4**	16-bit timer 4 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV4**	16-bit timer 4 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL4**	16-bit timer 4 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM4**	16-bit timer 4 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET4**	16-bit timer 4 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN4**	16-bit timer 4 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
16-bit timer 5	**00481A8**	DF	**CR5A15**	16-bit timer 5 comparison data A	0 to 65535				X	R/W	
comparison	**(HW)**	DE	**CR5A14**	CR5A15 = MSB					X		
register A		DD	**CR5A13**	CR5A0 = LSB					X		
		DC	**CR5A12**						X		
		DB	**CR5A11**						X		
		DA	**CR5A10**						X		
		D9	**CR5A9**						X		
		D8	**CR5A8**						X		
		D7	**CR5A7**						X		
		D6	**CR5A6**						X		
		D5	**CR5A5**						X		
		D4	**CR5A4**						X		
		D3	**CR5A3**						X		
		D2	**CR5A2**						X		
		D1	**CR5A1**						X		
		D0	**CR5A0**						X		
16-bit timer 5	**00481AA**	DF	**CR5B15**	16-bit timer 5 comparison data B	0 to 65535				X	R/W	
comparison	**(HW)**	DE	**CR5B14**	CR5B15 = MSB					X		
register B		DD	**CR5B13**	CR5B0 = LSB					X		
		DC	**CR5B12**						X		
		DB	**CR5B11**						X		
		DA	**CR5B10**						X		
		D9	**CR5B9**						X		
		D8	**CR5B8**						X		
		D7	**CR5B7**						X		
		D6	**CR5B6**						X		
		D5	**CR5B5**						X		
		D4	**CR5B4**						X		
		D3	**CR5B3**						X		
		D2	**CR5B2**						X		
		D1	**CR5B1**						X		
		D0	**CR5B0**						X		
16-bit timer 5	**00481AC**	DF	**TC515**	16-bit timer 5 counter data	0 to 65535				X	R	
counter data	**(HW)**	DE	**TC514**	TC515 = MSB					X		
register		DD	**TC513**	TC50 = LSB					X		
		DC	**TC512**						X		
		DB	**TC511**						X		
		DA	**TC510**						X		
		D9	**TC59**						X		
		D8	**TC58**						X		
		D7	**TC57**						X		
		D6	**TC56**						X		
		D5	**TC55**						X		
		D4	**TC54**						X		
		D3	**TC53**						X		
		D2	**TC52**						X		
		D1	**TC51**						X		
		D0	**TC50**						X		
16-bit timer 5	**00481AE**	D7	**–**	reserved	–				0	–	0 when being read.
control register	**(B)**	D6	**SELFM5**	16-bit timer 5 fine mode selection	1	Fine mode	0	Normal	0	R/W	
		D5	**SELCRB5**	16-bit timer 5 comparison buffer	1	Enabled	0	Disabled	0	R/W	
		D4	**OUTINV5**	16-bit timer 5 output inversion	1	Invert	0	Normal	0	R/W	
		D3	**CKSL5**	16-bit timer 5 input clock selection	1	External clock	0	Internal clock	0	R/W	
		D2	**PTM5**	16-bit timer 5 clock output control	1	On	0	Off	0	R/W	
		D1	**PRESET5**	16-bit timer 5 reset	1	Reset	0	Invalid	0	W	0 when being read.
		D0	**PRUN5**	16-bit timer 5 Run/Stop control	1	Run	0	Stop	0	R/W	

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
IDMA base address low-order register	**0048200 (HW)**	DF	**DBASEL15**	IDMA base address		0	R/W	
		DE	**DBASEL14**	low-order 16 bits		0		
		DD	**DBASEL13**	(Initial value: 0x0C003A0)		0		
		DC	**DBASEL12**			0		
		DB	**DBASEL11**			0		
		DA	**DBASEL10**			0		
		D9	**DBASEL9**			1		
		D8	**DBASEL8**			1		
		D7	**DBASEL7**			1		
		D6	**DBASEL6**			0		
		D5	**DBASEL5**			1		
		D4	**DBASEL4**			0		
		D3	**DBASEL3**			0		
		D2	**DBASEL2**			0		
		D1	**DBASEL1**			0		
		D0	**DBASEL0**			0		
IDMA base address high-order register	**0048202 (HW)**	DF–C	**–**	reserved	–	–	–	Undefined in read.
		DB	**DBASEH11**	IDMA base address		0	R/W	
		DA	**DBASEH10**	high-order 12 bits		0		
		D9	**DBASEH9**	(Initial value: 0x0C003A0)		0		
		D8	**DBASEH8**			0		
		D7	**DBASEH7**			1		
		D6	**DBASEH6**			1		
		D5	**DBASEH5**			0		
		D4	**DBASEH4**			0		
		D3	**DBASEH3**			0		
		D2	**DBASEH2**			0		
		D1	**DBASEH1**			0		
		D0	**DBASEH0**			0		
IDMA start register	**0048204 (B)**	D7	**DSTART**	IDMA start	1 IDMA start / 0 Stop	0	R/W	
		D6–0	**DCHN**	IDMA channel number	0 to 127	0	R/W	
IDMA enable register	**0048205 (B)**	D7–1	**–**	reserved	–	–	–	
		D0	**IDMAEN**	IDMA enable	1 Enabled / 0 Disabled	0	R/W	

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048220**	DF	**TC0_L7**	Ch.0 transfer counter[7:0]		X	R/W	
DMA Ch.0	**(HW)**	DE	**TC0_L6**	(block transfer mode)		X		
transfer		DD	**TC0_L5**			X		
counter		DC	**TC0_L4**	Ch.0 transfer counter[15:8]		X		
register		DB	**TC0_L3**	(single/successive transfer mode)		X		
		DA	**TC0_L2**			X		
		D9	**TC0_L1**			X		
		D8	**TC0_L0**			X		
		D7	**BLKLEN07**	Ch.0 block length		X	R/W	
		D6	**BLKLEN06**	(block transfer mode)		X		
		D5	**BLKLEN05**			X		
		D4	**BLKLEN04**	Ch.0 transfer counter[7:0]		X		
		D3	**BLKLEN03**	(single/successive transfer mode)		X		
		D2	**BLKLEN02**			X		
		D1	**BLKLEN01**			X		
		D0	**BLKLEN00**			X		
High-speed	**0048222**	DF	**DUALM0**	Ch.0 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
DMA Ch.0	**(HW)**	DE	**D0DIR**	D) Invalid	–	–	–	
control register				S) Ch.0 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	–	reserved	–	–	–	Undefined in read.
Note:		D7	**TC0_H7**	Ch.0 transfer counter[15:8]		X	R/W	
D) Dual address mode		D6	**TC0_H6**	(block transfer mode)		X		
S) Single address mode		D5	**TC0_H5**			X		
		D4	**TC0_H4**	Ch.0 transfer counter[23:16]		X		
		D3	**TC0_H3**	(single/successive transfer mode)		X		
		D2	**TC0_H2**			X		
		D1	**TC0_H1**			X		
		D0	**TC0_H0**			X		
High-speed	**0048224**	DF	**S0ADRL15**	D) Ch.0 source address[15:0]		X	R/W	
DMA Ch.0	**(HW)**	DE	**S0ADRL14**	S) Ch.0 memory address[15:0]		X		
low-order		DD	**S0ADRL13**			X		
source address		DC	**S0ADRL12**			X		
set-up register		DB	**S0ADRL11**			X		
		DA	**S0ADRL10**			X		
Note:		D9	**S0ADRL9**			X		
D) Dual address mode		A8	**S0ADRL8**			X		
S) Single address mode		D7	**S0ADRL7**			X		
		D6	**S0ADRL6**			X		
		D5	**S0ADRL5**			X		
		D4	**S0ADRL4**			X		
		D3	**S0ADRL3**			X		
		D2	**S0ADRL2**			X		
		D1	**S0ADRL1**			X		
		D0	**S0ADRL0**			X		
High-speed	**0048226**	DF	**DINTEN0**	Ch.0 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
DMA Ch.0	**(HW)**	DE	**DATSIZE0**	Ch.0 transfer data size	1 Half word / 0 Byte	0	R/W	
high-order		DD	**S0IN1**	D) Ch.0 source address control	S0IN[1:0] : Inc/dec	0	R/W	
source address		DC	**S0IN0**	S) Ch.0 memory address control	1 1 : Inc.(no init)	0		
set-up register					1 0 : Inc.(init)			
					0 1 : Dec.(no init)			
Note:					0 0 : Fixed			
D) Dual address mode		DB	**S0ADRH11**	D) Ch.0 source address[27:16]		X	R/W	
S) Single address mode		DA	**S0ADRH10**	S) Ch.0 memory address[27:16]		X		
		D9	**S0ADRH9**			X		
		A8	**S0ADRH8**			X		
		D7	**S0ADRH7**			X		
		D6	**S0ADRH6**			X		
		D5	**S0ADRH5**			X		
		D4	**S0ADRH4**			X		
		D3	**S0ADRH3**			X		
		D2	**S0ADRH2**			X		
		D1	**S0ADRH1**			X		
		D0	**S0ADRH0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048228**	DF	**D0ADRL15**	D) Ch.0 destination address[15:0]		X	R/W	
DMA Ch.0	**(HW)**	DE	**D0ADRL14**	S) Invalid		X		
low-order		DD	**D0ADRL13**			X		
destination		DC	**D0ADRL12**			X		
address set-up		DB	**D0ADRL11**			X		
register		DA	**D0ADRL10**			X		
		D9	**D0ADRL9**			X		
Note:		A8	**D0ADRL8**			X		
D) Dual address mode		D7	**D0ADRL7**			X		
S) Single address mode		D6	**D0ADRL6**			X		
		D5	**D0ADRL5**			X		
		D4	**D0ADRL4**			X		
		D3	**D0ADRL3**			X		
		D2	**D0ADRL2**			X		
		D1	**D0ADRL1**			X		
		D0	**D0ADRL0**			X		
High-speed	**004822A**	DF	**D0MOD1**	Ch.0 transfer mode	D0MOD[1:0] / Mode	0	R/W	
DMA Ch.0	**(HW)**	DE	**D0MOD0**		1 1 Invalid	0		
high-order					1 0 Block			
destination					0 1 Successive			
address set-up					0 0 Single			
register		DD	**D0IN1**	D) Ch.0 destination address control	D0IN[1:0] / Inc/dec	0	R/W	
		DC	**D0IN0**	S) Invalid	1 1 Inc.(no init)	0		
Note:					1 0 Inc.(init)			
D) Dual address mode					0 1 Dec.(no init)			
S) Single address mode					0 0 Fixed			
		DB	**D0ADRH11**	D) Ch.0 destination address[27:16]		X	R/W	
		DA	**D0ADRH10**	S) Invalid		X		
		D9	**D0ADRH9**			X		
		A8	**D0ADRH8**			X		
		D7	**D0ADRH7**			X		
		D6	**D0ADRH6**			X		
		D5	**D0ADRH5**			X		
		D4	**D0ADRH4**			X		
		D3	**D0ADRH3**			X		
		D2	**D0ADRH2**			X		
		D1	**D0ADRH1**			X		
		D0	**D0ADRH0**			X		
High-speed DMA Ch.0 enable register	**004822C (HW)**	DF–1	–	reserved	–	–	–	Undefined in read.
		D0	**HS0_EN**	Ch.0 enable	1 Enable / 0 Disable	0	R/W	
High-speed DMA Ch.0 trigger flag register	**004822E (HW)**	DF–1	–	reserved	–	–	–	Undefined in read.
		D0	**HS0_TF**	Ch.0 trigger flag clear (writing)	1 Clear / 0 No operation	0	R/W	
				Ch.0 trigger flag status (reading)	1 Set / 0 Cleared			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed DMA Ch.1 transfer counter register	**0048230 (HW)**	DF	**TC1_L7**	Ch.1 transfer counter[7:0] (block transfer mode)		X	R/W	
		DE	**TC1_L6**			X		
		DD	**TC1_L5**			X		
		DC	**TC1_L4**	Ch.1 transfer counter[15:8] (single/successive transfer mode)		X		
		DB	**TC1_L3**			X		
		DA	**TC1_L2**			X		
		D9	**TC1_L1**			X		
		D8	**TC1_L0**			X		
		D7	**BLKLEN17**	Ch.1 block length (block transfer mode)		X	R/W	
		D6	**BLKLEN16**			X		
		D5	**BLKLEN15**			X		
		D4	**BLKLEN14**	Ch.1 transfer counter[7:0] (single/successive transfer mode)		X		
		D3	**BLKLEN13**			X		
		D2	**BLKLEN12**			X		
		D1	**BLKLEN11**			X		
		D0	**BLKLEN10**			X		
High-speed DMA Ch.1 control register Note: D) Dual address mode S) Single address mode	**0048232 (HW)**	DF	**DUALM1**	Ch.1 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
		DE	**D1DIR**	D) Invalid	–	–	–	
				S) Ch.1 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	–	reserved	–	–	–	Undefined in read.
		D7	**TC1_H7**	Ch.1 transfer counter[15:8] (block transfer mode)		X	R/W	
		D6	**TC1_H6**			X		
		D5	**TC1_H5**			X		
		D4	**TC1_H4**	Ch.1 transfer counter[23:16] (single/successive transfer mode)		X		
		D3	**TC1_H3**			X		
		D2	**TC1_H2**			X		
		D1	**TC1_H1**			X		
		D0	**TC1_H0**			X		
High-speed DMA Ch.1 low-order source address set-up register Note: D) Dual address mode S) Single address mode	**0048234 (HW)**	DF	**S1ADRL15**	D) Ch.1 source address[15:0] S) Ch.1 memory address[15:0]		X	R/W	
		DE	**S1ADRL14**			X		
		DD	**S1ADRL13**			X		
		DC	**S1ADRL12**			X		
		DB	**S1ADRL11**			X		
		DA	**S1ADRL10**			X		
		D9	**S1ADRL9**			X		
		A8	**S1ADRL8**			X		
		D7	**S1ADRL7**			X		
		D6	**S1ADRL6**			X		
		D5	**S1ADRL5**			X		
		D4	**S1ADRL4**			X		
		D3	**S1ADRL3**			X		
		D2	**S1ADRL2**			X		
		D1	**S1ADRL1**			X		
		D0	**S1ADRL0**			X		
High-speed DMA Ch.1 high-order source address set-up register Note: D) Dual address mode S) Single address mode	**0048236 (HW)**	DF	**DINTEN1**	Ch.1 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
		DE	**DATSIZE1**	Ch.1 transfer data size	1 Half word / 0 Byte	0	R/W	
		DD	**S1IN1**	D) Ch.1 source address control	S1IN[1:0] – Inc/dec	0	R/W	
		DC	**S1IN0**	S) Ch.1 memory address control	1 1 – Inc.(no init) 1 0 – Inc.(init) 0 1 – Dec.(no init) 0 0 – Fixed	0		
		DB	**S1ADRH11**	D) Ch.1 source address[27:16] S) Ch.1 memory address[27:16]		X	R/W	
		DA	**S1ADRH10**			X		
		D9	**S1ADRH9**			X		
		A8	**S1ADRH8**			X		
		D7	**S1ADRH7**			X		
		D6	**S1ADRH6**			X		
		D5	**S1ADRH5**			X		
		D4	**S1ADRH4**			X		
		D3	**S1ADRH3**			X		
		D2	**S1ADRH2**			X		
		D1	**S1ADRH1**			X		
		D0	**S1ADRH0**			X		

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048238**	DF	**D1ADRL15**	D) Ch.1 destination address[15:0]		X	R/W	
DMA Ch.1	**(HW)**	DE	**D1ADRL14**	S) Invalid		X		
low-order		DD	**D1ADRL13**			X		
destination		DC	**D1ADRL12**			X		
address set-up		DB	**D1ADRL11**			X		
register		DA	**D1ADRL10**			X		
		D9	**D1ADRL9**			X		
Note:		A8	**D1ADRL8**			X		
D) Dual address mode		D7	**D1ADRL7**			X		
S) Single address mode		D6	**D1ADRL6**			X		
		D5	**D1ADRL5**			X		
		D4	**D1ADRL4**			X		
		D3	**D1ADRL3**			X		
		D2	**D1ADRL2**			X		
		D1	**D1ADRL1**			X		
		D0	**D1ADRL0**			X		
High-speed	**004823A**	DF	**D1MOD1**	Ch.1 transfer mode	D1MOD[1:0] / Mode	0	R/W	
DMA Ch.1	**(HW)**	DE	**D1MOD0**		1 1 Invalid	0		
high-order					1 0 Block			
destination					0 1 Successive			
address set-up					0 0 Single			
register		DD	**D1IN1**	D) Ch.1 destination address	D1IN[1:0] / Inc/dec	0	R/W	
		DC	**D1IN0**	control	1 1 Inc.(no init)	0		
Note:				S) Invalid	1 0 Inc.(init)			
D) Dual address mode					0 1 Dec.(no init)			
S) Single address mode					0 0 Fixed			
		DB	**D1ADRH11**	D) Ch.1 destination		X	R/W	
		DA	**D1ADRH10**	address[27:16]		X		
		D9	**D1ADRH9**	S) Invalid		X		
		A8	**D1ADRH8**			X		
		D7	**D1ADRH7**			X		
		D6	**D1ADRH6**			X		
		D5	**D1ADRH5**			X		
		D4	**D1ADRH4**			X		
		D3	**D1ADRH3**			X		
		D2	**D1ADRH2**			X		
		D1	**D1ADRH1**			X		
		D0	**D1ADRH0**			X		
High-speed DMA Ch.1	**004823C**	DF–1	–	reserved	–	–	–	Undefined in read.
enable register	**(HW)**	D0	**HS1_EN**	Ch.1 enable	1 Enable / 0 Disable	0	R/W	
High-speed DMA Ch.1	**004823E**	DF–1	–	reserved	–	–	–	Undefined in read.
trigger flag	**(HW)**	D0	**HS1_TF**	Ch.1 trigger flag clear (writing)	1 Clear / 0 No operation	0	R/W	
register				Ch.1 trigger flag status (reading)	1 Set / 0 Cleared			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048240**	DF	**TC2_L7**	Ch.2 transfer counter[7:0]		X	R/W	
DMA Ch.2	**(HW)**	DE	**TC2_L6**	(block transfer mode)		X		
transfer		DD	**TC2_L5**			X		
counter		DC	**TC2_L4**	Ch.2 transfer counter[15:8]		X		
register		DB	**TC2_L3**	(single/successive transfer mode)		X		
		DA	**TC2_L2**			X		
		D9	**TC2_L1**			X		
		D8	**TC2_L0**			X		
		D7	**BLKLEN27**	Ch.2 block length		X	R/W	
		D6	**BLKLEN26**	(block transfer mode)		X		
		D5	**BLKLEN25**			X		
		D4	**BLKLEN24**	Ch.2 transfer counter[7:0]		X		
		D3	**BLKLEN23**	(single/successive transfer mode)		X		
		D2	**BLKLEN22**			X		
		D1	**BLKLEN21**			X		
		D0	**BLKLEN20**			X		
High-speed	**0048242**	DF	**DUALM2**	Ch.2 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
DMA Ch.2	**(HW)**	DE	**D2DIR**	D) Invalid	–	–	–	
control register				S) Ch.2 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	–	reserved	–	–	–	Undefined in read.
Note:		D7	**TC2_H7**	Ch.2 transfer counter[15:8]		X	R/W	
D) Dual address mode		D6	**TC2_H6**	(block transfer mode)		X		
S) Single address mode		D5	**TC2_H5**			X		
		D4	**TC2_H4**	Ch.2 transfer counter[23:16]		X		
		D3	**TC2_H3**	(single/successive transfer mode)		X		
		D2	**TC2_H2**			X		
		D1	**TC2_H1**			X		
		D0	**TC2_H0**			X		
High-speed	**0048244**	DF	**S2ADRL15**	D) Ch.2 source address[15:0]		X	R/W	
DMA Ch.2	**(HW)**	DE	**S2ADRL14**	S) Ch.2 memory address[15:0]		X		
low-order		DD	**S2ADRL13**			X		
source address		DC	**S2ADRL12**			X		
set-up register		DB	**S2ADRL11**			X		
		DA	**S2ADRL10**			X		
Note:		D9	**S2ADRL9**			X		
D) Dual address mode		A8	**S2ADRL8**			X		
S) Single address mode		D7	**S2ADRL7**			X		
		D6	**S2ADRL6**			X		
		D5	**S2ADRL5**			X		
		D4	**S2ADRL4**			X		
		D3	**S2ADRL3**			X		
		D2	**S2ADRL2**			X		
		D1	**S2ADRL1**			X		
		D0	**S2ADRL0**			X		
High-speed	**0048246**	DF	**DINTEN2**	Ch.2 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
DMA Ch.2	**(HW)**	DE	**DATSIZE2**	Ch.2 transfer data size	1 Half word / 0 Byte	0	R/W	
high-order		DD	**S2IN1**	D) Ch.2 source address control	S2IN[1:0] — Inc/dec	0	R/W	
source address		DC	**S2IN0**	S) Ch.2 memory address control	1 1 — Inc.(no init)	0		
set-up register					1 0 — Inc.(init)			
					0 1 — Dec.(no init)			
Note:					0 0 — Fixed			
D) Dual address mode		DB	**S2ADRH11**	D) Ch.2 source address[27:16]		X	R/W	
S) Single address mode		DA	**S2ADRH10**	S) Ch.2 memory address[27:16]		X		
		D9	**S2ADRH9**			X		
		A8	**S2ADRH8**			X		
		D7	**S2ADRH7**			X		
		D6	**S2ADRH6**			X		
		D5	**S2ADRH5**			X		
		D4	**S2ADRH4**			X		
		D3	**S2ADRH3**			X		
		D2	**S2ADRH2**			X		
		D1	**S2ADRH1**			X		
		D0	**S2ADRH0**			X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
High-speed DMA Ch.2 low-order destination address set-up register	**0048248 (HW)**	DF	**D2ADRL15**	D) Ch.2 destination address[15:0]					X	R/W	
		DE	**D2ADRL14**	S) Invalid					X		
		DD	**D2ADRL13**						X		
		DC	**D2ADRL12**						X		
		DB	**D2ADRL11**						X		
		DA	**D2ADRL10**						X		
		D9	**D2ADRL9**						X		
Note:		A8	**D2ADRL8**						X		
D) Dual address mode		D7	**D2ADRL7**						X		
S) Single address mode		D6	**D2ADRL6**						X		
		D5	**D2ADRL5**						X		
		D4	**D2ADRL4**						X		
		D3	**D2ADRL3**						X		
		D2	**D2ADRL2**						X		
		D1	**D2ADRL1**						X		
		D0	**D2ADRL0**						X		
High-speed DMA Ch.2 high-order destination address set-up register	**004824A (HW)**	DF	**D2MOD1**	Ch.2 transfer mode	D2MOD[1:0]		Mode		0	R/W	
		DE	**D2MOD0**		1	1	Invalid		0		
					1	0	Block				
					0	1	Successive				
					0	0	Single				
		DD	**D2IN1**	D) Ch.2 destination address control	D2IN[1:0]		Inc/dec		0	R/W	
		DC	**D2IN0**	S) Invalid	1	1	Inc.(no init)		0		
Note:					1	0	Inc.(init)				
D) Dual address mode					0	1	Dec.(no init)				
S) Single address mode					0	0	Fixed				
		DB	**D2ADRH11**	D) Ch.2 destination address[27:16]					X	R/W	
		DA	**D2ADRH10**	S) Invalid					X		
		D9	**D2ADRH9**						X		
		A8	**D2ADRH8**						X		
		D7	**D2ADRH7**						X		
		D6	**D2ADRH6**						X		
		D5	**D2ADRH5**						X		
		D4	**D2ADRH4**						X		
		D3	**D2ADRH3**						X		
		D2	**D2ADRH2**						X		
		D1	**D2ADRH1**						X		
		D0	**D2ADRH0**						X		
High-speed DMA Ch.2 enable register	**004824C (HW)**	DF–1	–	reserved	–				–	–	Undefined in read.
		D0	**HS2_EN**	Ch.2 enable	1	Enable	0	Disable	0	R/W	
High-speed DMA Ch.2 trigger flag register	**004824E (HW)**	DF–1	–	reserved	–				–	–	Undefined in read.
		D0	**HS2_TF**	Ch.2 trigger flag clear (writing)	1	Clear	0	No operation	0	R/W	
				Ch.2 trigger flag status (reading)	1	Set	0	Cleared			

Register name	Address	Bit	Name	Function	Setting	Init.	R/W	Remarks
High-speed	**0048250**	DF	**TC3_L7**	Ch.3 transfer counter[7:0]		X	R/W	
DMA Ch.3	**(HW)**	DE	**TC3_L6**	(block transfer mode)		X		
transfer		DD	**TC3_L5**			X		
counter		DC	**TC3_L4**	Ch.3 transfer counter[15:8]		X		
register		DB	**TC3_L3**	(single/successive transfer mode)		X		
		DA	**TC3_L2**			X		
		D9	**TC3_L1**			X		
		D8	**TC3_L0**			X		
		D7	**BLKLEN37**	Ch.3 block length		X	R/W	
		D6	**BLKLEN36**	(block transfer mode)		X		
		D5	**BLKLEN35**			X		
		D4	**BLKLEN34**	Ch.3 transfer counter[7:0]		X		
		D3	**BLKLEN33**	(single/successive transfer mode)		X		
		D2	**BLKLEN32**			X		
		D1	**BLKLEN31**			X		
		D0	**BLKLEN30**			X		
High-speed	**0048252**	DF	**DUALM3**	Ch.3 address mode selection	1 Dual addr / 0 Single addr	0	R/W	
DMA Ch.3	**(HW)**	DE	**D3DIR**	D) Invalid	–	–	–	
control register				S) Ch.3 transfer direction control	1 Memory WR / 0 Memory RD	0	R/W	
		DD–8	–	reserved	–	–	–	Undefined in read.
Note:		D7	**TC3_H7**	Ch.3 transfer counter[15:8]		X	R/W	
D) Dual address mode		D6	**TC3_H6**	(block transfer mode)		X		
S) Single address mode		D5	**TC3_H5**			X		
		D4	**TC3_H4**	Ch.3 transfer counter[23:16]		X		
		D3	**TC3_H3**	(single/successive transfer mode)		X		
		D2	**TC3_H2**			X		
		D1	**TC3_H1**			X		
		D0	**TC3_H0**			X		
High-speed	**0048254**	DF	**S3ADRL15**	D) Ch.3 source address[15:0]		X	R/W	
DMA Ch.3	**(HW)**	DE	**S3ADRL14**	S) Ch.3 memory address[15:0]		X		
low-order		DD	**S3ADRL13**			X		
source address		DC	**S3ADRL12**			X		
set-up register		DB	**S3ADRL11**			X		
		DA	**S3ADRL10**			X		
Note:		D9	**S3ADRL9**			X		
D) Dual address mode		A8	**S3ADRL8**			X		
S) Single address mode		D7	**S3ADRL7**			X		
		D6	**S3ADRL6**			X		
		D5	**S3ADRL5**			X		
		D4	**S3ADRL4**			X		
		D3	**S3ADRL3**			X		
		D2	**S3ADRL2**			X		
		D1	**S3ADRL1**			X		
		D0	**S3ADRL0**			X		
High-speed	**0048256**	DF	**DINTEN3**	Ch.3 interrupt enable	1 Enabled / 0 Disabled	0	R/W	
DMA Ch.3	**(HW)**	DE	**DATSIZE3**	Ch.3 transfer data size	1 Half word / 0 Byte	0	R/W	
high-order		DD	**S3IN1**	D) Ch.3 source address control	S3IN[1:0] — Inc/dec	0	R/W	
source address		DC	**S3IN0**	S) Ch.3 memory address control	1 1 — Inc.(no init)	0		
set-up register					1 0 — Inc.(init)			
					0 1 — Dec.(no init)			
Note:					0 0 — Fixed			
D) Dual address mode		DB	**S3ADRH11**	D) Ch.3 source address[27:16]		X	R/W	
S) Single address mode		DA	**S3ADRH10**	S) Ch.3 memory address[27:16]		X		
		D9	**S3ADRH9**			X		
		A8	**S3ADRH8**			X		
		D7	**S3ADRH7**			X		
		D6	**S3ADRH6**			X		
		D5	**S3ADRH5**			X		
		D4	**S3ADRH4**			X		
		D3	**S3ADRH3**			X		
		D2	**S3ADRH2**			X		
		D1	**S3ADRH1**			X		
		D0	**S3ADRH0**			X		

Register name	Address	Bit	Name	Function	Setting				Init.	R/W	Remarks
High-speed DMA Ch.3 low-order destination address set-up register Note: D) Dual address mode S) Single address mode	**0048258 (HW)**	DF	**D3ADRL15**	D) Ch.3 destination address[15:0]					X	R/W	
		DE	**D3ADRL14**	S) Invalid					X		
		DD	**D3ADRL13**						X		
		DC	**D3ADRL12**						X		
		DB	**D3ADRL11**						X		
		DA	**D3ADRL10**						X		
		D9	**D3ADRL9**						X		
		A8	**D3ADRL8**						X		
		D7	**D3ADRL7**						X		
		D6	**D3ADRL6**						X		
		D5	**D3ADRL5**						X		
		D4	**D3ADRL4**						X		
		D3	**D3ADRL3**						X		
		D2	**D3ADRL2**						X		
		D1	**D3ADRL1**						X		
		D0	**D3ADRL0**						X		
High-speed DMA Ch.3 high-order destination address set-up register Note: D) Dual address mode S) Single address mode	**004825A (HW)**	DF	**D3MOD1**	Ch.3 transfer mode	D3MOD[1:0]		Mode		0	R/W	
		DE	**D3MOD0**		1	1	Invalid		0		
					1	0	Block				
					0	1	Successive				
					0	0	Single				
		DD	**D3IN1**	D) Ch.3 destination address control	D3IN[1:0]		Inc/dec		0	R/W	
		DC	**D3IN0**	S) Invalid	1	1	Inc.(no init)		0		
					1	0	Inc.(init)				
					0	1	Dec.(no init)				
					0	0	Fixed				
		DB	**D3ADRH11**	D) Ch.3 destination address[27:16]					X	R/W	
		DA	**D3ADRH10**	S) Invalid					X		
		D9	**D3ADRH9**						X		
		A8	**D3ADRH8**						X		
		D7	**D3ADRH7**						X		
		D6	**D3ADRH6**						X		
		D5	**D3ADRH5**						X		
		D4	**D3ADRH4**						X		
		D3	**D3ADRH3**						X		
		D2	**D3ADRH2**						X		
		D1	**D3ADRH1**						X		
		D0	**D3ADRH0**						X		
High-speed DMA Ch.3 enable register	**004825C (HW)**	DF–1	–	reserved	–				–	–	Undefined in read.
		D0	**HS3_EN**	Ch.3 enable	1	Enable	0	Disable	0	R/W	
High-speed DMA Ch.3 trigger flag register	**004825E (HW)**	DF–1	–	reserved	–				–	–	Undefined in read.
		D0	**HS3_TF**	Ch.3 trigger flag clear (writing)	1	Clear	0	No operation	0	R/W	
				Ch.3 trigger flag status (reading)	1	Set	0	Cleared			

THIS PAGE IS BLANK.

EPSON International Sales Operations

AMERICA

EPSON ELECTRONICS AMERICA, INC.

- HEADQUARTERS -
1960 E. Grand Avenue
El Segundo, CA 90245, U.S.A.
Phone: +1-310-955-5300 Fax: +1-310-955-5400

- SALES OFFICES -

West
150 River Oaks Parkway
San Jose, CA 95134, U.S.A.
Phone: +1-408-922-0200 Fax: +1-408-922-0238

Central
101 Virginia Street, Suite 290
Crystal Lake, IL 60014, U.S.A.
Phone: +1-815-455-7630 Fax: +1-815-455-7633

Northeast
301 Edgewater Place, Suite 120
Wakefield, MA 01880, U.S.A.
Phone: +1-781-246-3600 Fax: +1-781-246-5443

Southeast
3010 Royal Blvd. South, Suite 170
Alpharetta, GA 30005, U.S.A.
Phone: +1-877-EEA-0020 Fax: +1-770-777-2637

EUROPE

EPSON EUROPE ELECTRONICS GmbH

- HEADQUARTERS -
Riesstrasse 15
80992 Muenchen, GERMANY
Phone: +49-(0)89-14005-0 Fax: +49-(0)89-14005-110

- GERMANY -
SALES OFFICE
Altstadtstrasse 176
51379 Leverkusen, GERMANY
Phone: +49-(0)217-15045-0 Fax: +49-(0)217-15045-10

- UNITED KINGDOM -
UK BRANCH OFFICE
2.4 Doncastle House, Doncastle Road
Bracknell, Berkshire RG12 8PE, ENGLAND
Phone: +44-(0)1344-381700 Fax: +44-(0)1344-381701

- FRANCE -
FRENCH BRANCH OFFICE
1 Avenue de l' Atlantique, LP 915 Les Conquerants
Z.A. de Courtaboeuf 2, F-91976 Les Ulis Cedex, FRANCE
Phone: +33-(0)1-64862350 Fax: +33-(0)1-64862355

ASIA

- CHINA -
EPSON (CHINA) CO., LTD.
28F, Beijing Silver Tower 2# North RD DongSanHuan
ChaoYang District, Beijing, CHINA
Phone: 64106655 Fax: 64107320

SHANGHAI BRANCH
4F, Bldg., 27, No. 69, Gui Jing Road
Caohejing, Shanghai, CHINA
Phone: 21-6485-5552 Fax: 21-6485-0775

- HONG KONG, CHINA -
EPSON HONG KONG LTD.
20/F., Harbour Centre, 25 Harbour Road
Wanchai, HONG KONG
Phone: +852-2585-4600 Fax: +852-2827-4346
Telex: 65542 EPSCO HX

- TAIWAN, R.O.C. -
EPSON TAIWAN TECHNOLOGY & TRADING LTD.
10F, No. 287, Nanking East Road, Sec. 3
Taipei, TAIWAN, R.O.C.
Phone: 02-2717-7360 Fax: 02-2712-9164
Telex: 24444 EPSONTB

HSINCHU OFFICE
13F-3, No. 295, Kuang-Fu Road, Sec. 2
HsinChu 300, TAIWAN, R.O.C.
Phone: 03-573-9900 Fax: 03-573-9169

- SINGAPORE -
EPSON SINGAPORE PTE., LTD.
No. 1 Temasek Avenue, #36-00
Millenia Tower, SINGAPORE 039192
Phone: +65-337-7911 Fax: +65-334-2716

- KOREA -
SEIKO EPSON CORPORATION KOREA OFFICE
50F, KLI 63 Bldg., 60 Yoido-Dong
Youngdeungpo-Ku, Seoul, 150-010, KOREA
Phone: 02-784-6027 Fax: 02-767-3677

- JAPAN -
SEIKO EPSON CORPORATION
ELECTRONIC DEVICES MARKETING DIVISION

Electronic Device Marketing Department
IC Marketing & Engineering Group
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-(0)42-587-5816 Fax: +81-(0)42-587-5624

ED International Marketing Department I (Europe & U.S.A.)
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-(0)42-587-5812 Fax: +81-(0)42-587-5564

ED International Marketing Department II (Asia)
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-(0)42-587-5814 Fax: +81-(0)42-587-5110

In pursuit of **“Saving” Technology**, Epson electronic devices.
Our lineup of semiconductors, liquid crystal displays and quartz devices
assists in creating the products of our customers’ dreams.
Epson IS energy savings.

EPSON

SEIKO EPSON CORPORATION
ELECTRONIC DEVICES MARKETING DIVISION

■ EPSON Electronic Devices Website

http://www.epson.co.jp/device/

Issue NOVEMBER 1999, Printed in Japan Ⓜ B