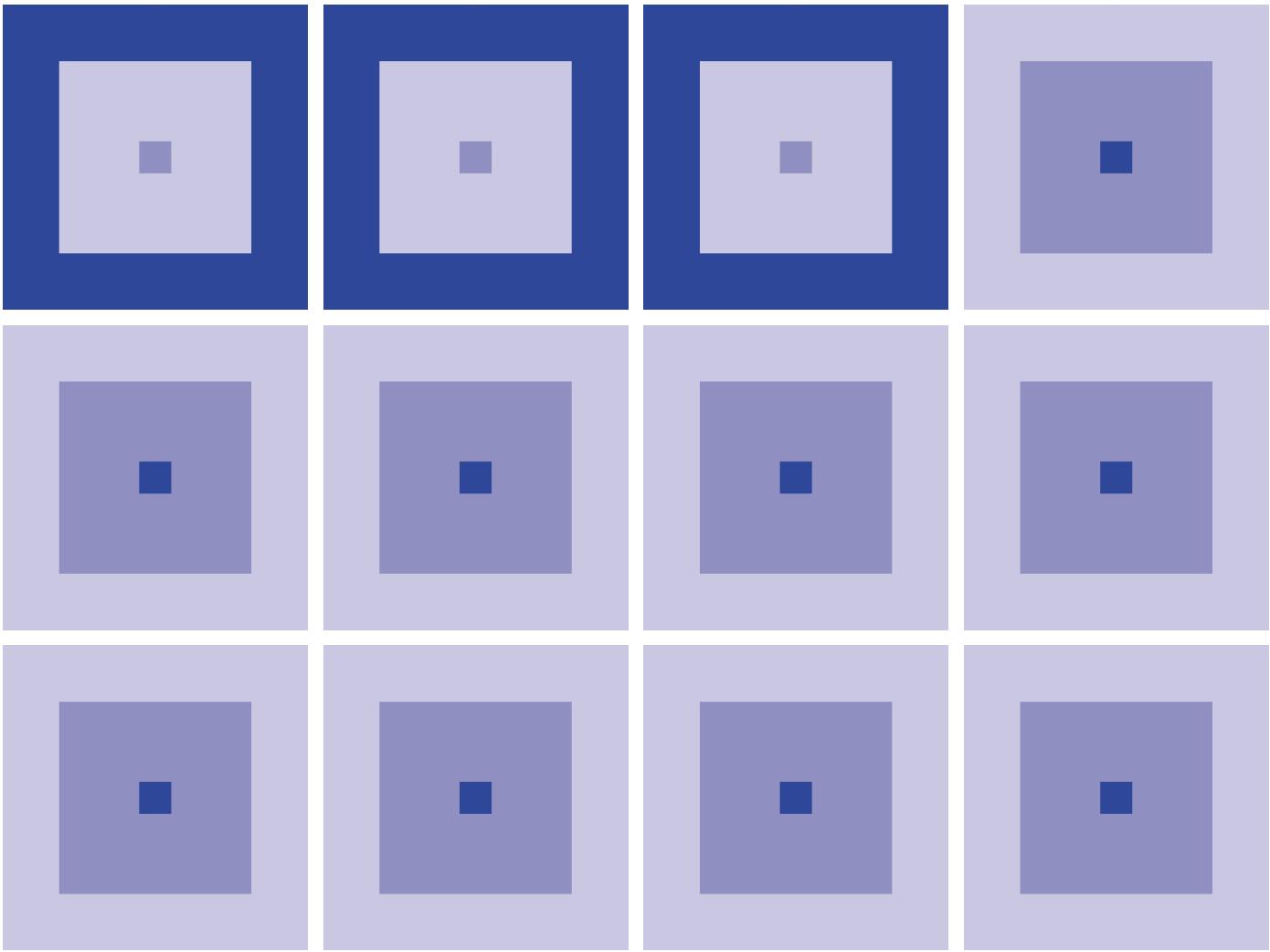
MF1528-01

EPSON

SCSI-2/IDE Interface Controller

S1R72103

Technical Manual



ENERGY SAVING EPSON

SEIKO EPSON CORPORATION

NOTICE

No part of this material may be reproduced or duplicated in any form or by any means without the written permission of Seiko Epson. Seiko Epson reserves the right to make changes to this material without notice. Seiko Epson does not assume any liability of any kind arising out of any inaccuracies contained in this material or due to its application or use in any product or circuit and, further, there is no representation that this material is applicable to products requiring high level reliability, such as, medical products. Moreover, no license to any intellectual property rights is granted by implication or otherwise, and there is no representation or warranty that anything made in accordance with this material will be free from any patent or copyright infringement of a third party. This material or portions thereof may contain technology or the subject relating to strategic products under the control of the Foreign Exchange and Foreign Trade Law of Japan and may require an export license from the Ministry of International Trade and Industry or other approval from anther government agency.

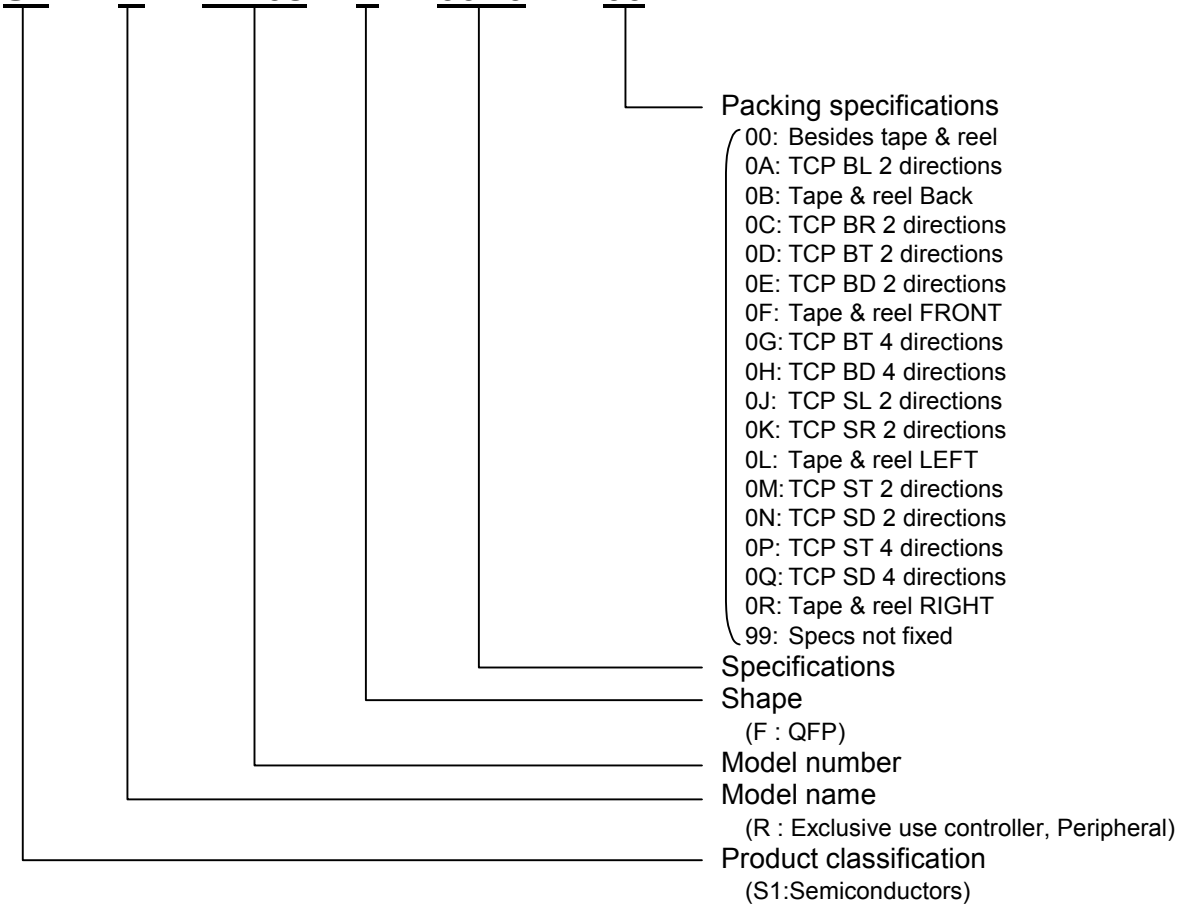
©SEIKO EPSON CORPORATION 2002, All rights reserved.

All other product names mentioned herein are trademarks and/or registered trademarks of their respective companies.

Configuration of product number

●DEVICES

S1 R 72103 F 00A0 00



CONTENTS

1. DESCRIPTION

S1R72103 is an interface control IC that mutually converts IDE and SCSI-2 compatible with SCAM.

2. FEATURES

«CPU Interface»

- Connectable to a general-purpose CPU

«SCSI-2 Interface»

- Compatible with SCSI-2 (10Mbps (synchronous), 5Mbps (asynchronous)) and SCSI-3 Fast-20 (20Mbps (synchronous))
- Compatible with SCAM Lv.1 (compatible with Lv.2 with firmware)
- Automatic processing of phase control
- Built-in single end driver
- Built-in active negation function

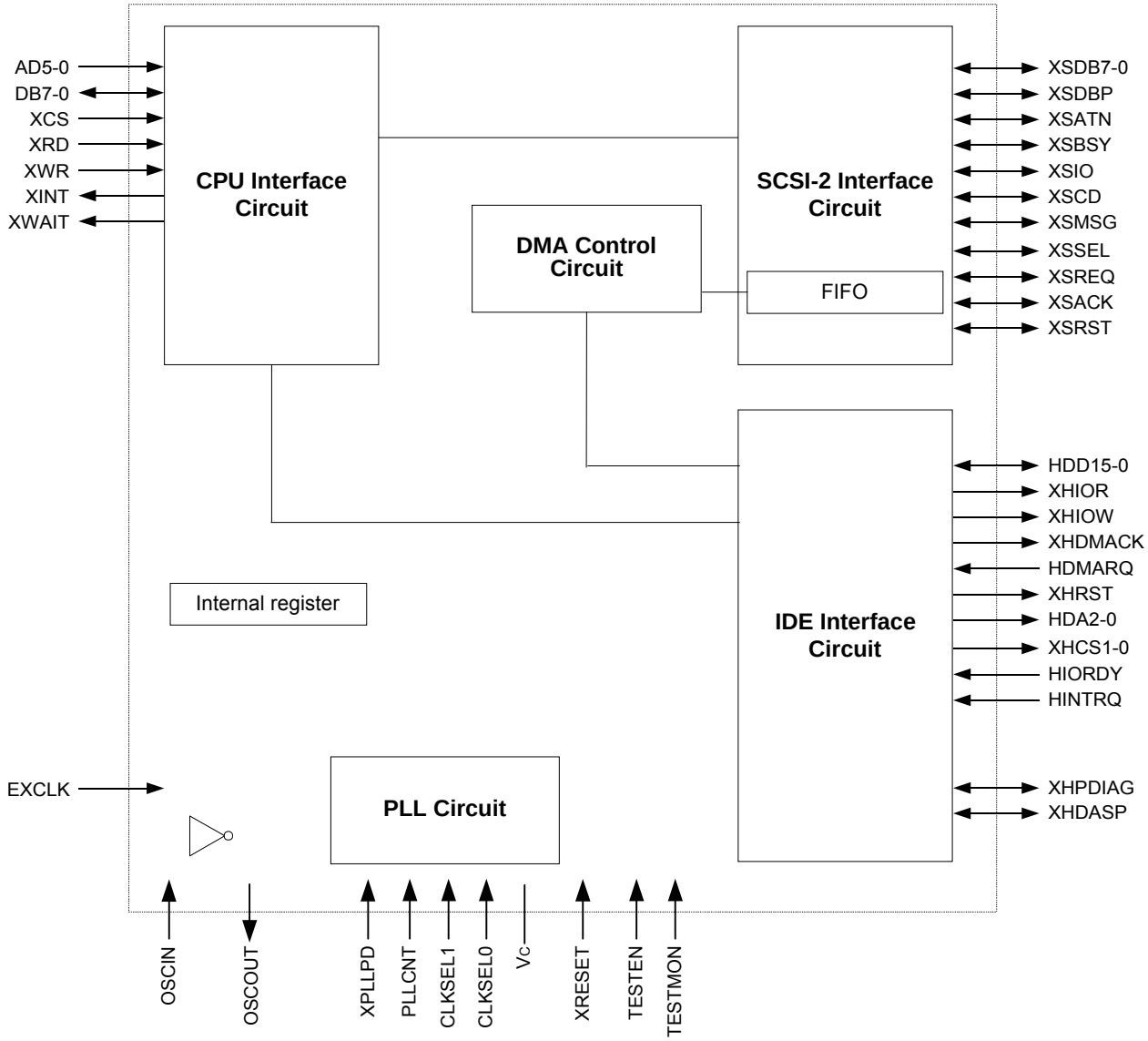
«IDE Interface»

- IDE access function by program modes
- Data transfer function with SCSI by DMA
- Compatible with PIO mode 0/1/2/3/4, multiword DMA mode 0/1/2, and Ultra-DMA mode0/1/2
- IDE interface section can be also used as a general-purpose port interface

«Others»

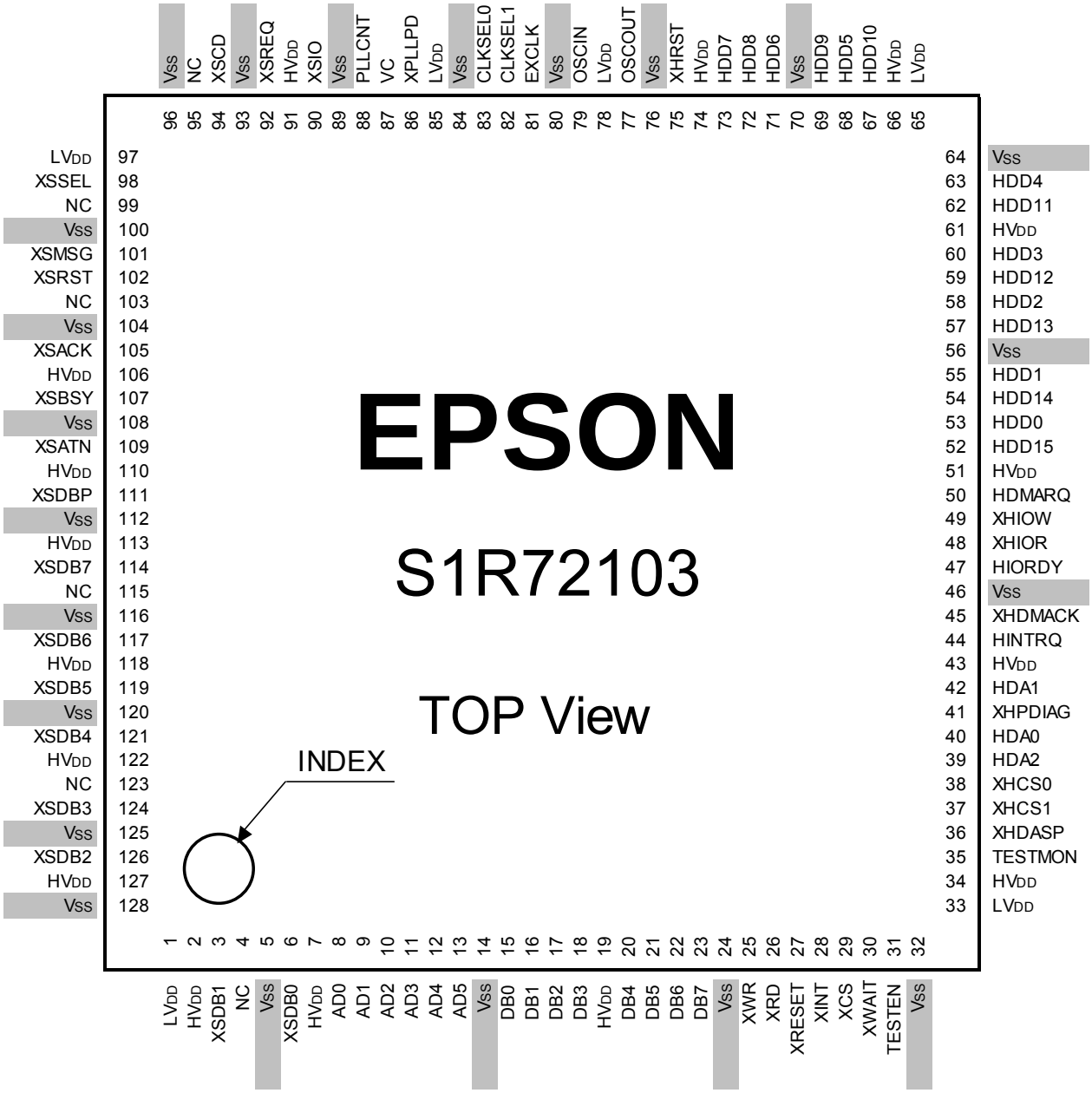
- Built-in 16MHz /20MHz /40MHz oscillation circuit
- Built-in PLL circuit
- 128 pin QFP (0.4 mm pitch)
- Supply voltage: 5.0V±10% and 3.3V±0.3V
- No anti-radiation design

3. BLOCK DIAGRAM



4. PIN ASSIGNMENT

S1R72103 (QFP15-128pin)



5. PIN DESCRIPTION

The control signal with "X" at the head of a pin name is LOW-active.

Pin No.	Symbol	I/O	Functional description	Remarks
SCSI interface-related matters (18)				
6	XSDB0	Is/Otr	SCSI data signal (SD0 to SD7)	Drive capability 48mA
3	XSDB1			
126	XSDB2			
124	XSDB3			
121	XSDB4			
119	XSDB5			
117	XSDB6			
114	XSDB7			
111	XSDBP		SCSI data parity signal	Drive capability 48mA
109	XSATN	I/Ood	SCSI ATN signal	Drive capability 48mA
107	XSBSY	I/Ood	SCSI BSY signal	Drive capability 48mA
105	XSACK	Is/Otr	SCSI ACK signal	Drive capability 48mA
102	XSRST	I/Ood	SCSI RST signal	Drive capability 48mA
101	XSMSG	I/Ood	SCSI MSG signal	Drive capability 48mA
98	XSSEL	I/Ood	SCSI SEL signal	Drive capability 48mA
94	XSCD	I/Ood	SCSI C/D signal	Drive capability 48mA
92	XSREQ	Is/Otr	SCSI REQ signal	Drive capability 48mA
90	XSIO	I /Ood	SCSI I/O signal	Drive capability 48mA
IDE interface-related matters (30)				
75	XHRST	Otr	IDE reset signal	Drive capability 6mA
53	HDD0	I/O	IDE DMA data bus signal (DD0 to 15)	Drive capability 3mA
55	HDD1			
58	HDD2			
60	HDD3			
63	HDD4			
68	HDD5			
71	HDD6			
73	HDD7			
72	HDD8			
69	HDD9			
67	HDD10			
62	HDD11			
59	HDD12			
57	HDD13			
54	HDD14			
52	HDD15			
50	HDMARQ	Is/O	IDE DMA request signal (also operable in negative logic)	Drive capability 6mA
49	XHIOW	Is/O	IDE write signal	Drive capability 3mA
48	XHIOR	Is/O	IDE read signal	Drive capability 3mA
47	HIORDY	Is	IDE IORDY signal	
45	XHDMACK	Is/O	IDE DMA ACK signal	Drive capability 3mA
44	HINTRQ	Is	IDE Interrupt signal	
40	HDA0	Otr	IDE address signal (DA0 to 2)	Drive capability 6mA
42	HDA1			
39	HDA2			
41	XHPDIAG	Is	IDE PDIAG signal	Drive capability 6mA
38	XHCS0	Otr	IDE chip select signal (XCS0 to 1)	Drive capability 6mA
37	XHCS1			
36	XHDASP	Is	IDE DASP signal	

Pin No.	Symbol	I/O	Functional description	Remarks
CPU interface-related matters (19)				
8	AD0	Ipu	Address input pin (AD0 to AD5)	
9	AD1			
10	AD2			
11	AD3			
12	AD4			
13	AD5			
15	DB0	Ipu/O	Data pin (DB0 to DB7)	Drive capability 3mA
16	DB1			
17	DB2			
18	DB3			
20	DB4			
21	DB5			
22	DB6			
23	DB7			
25	XWR	Ispu	Data write signal	
26	XRD	Ispu	Data read signal	
28	XINT	Otr	Interrupt request output signal	Drive capability 6mA
29	XCS	Ispu	Chip select signal for accessing internal register	
30	XWAIT	Ood	Wait signal	Drive capability 6mA
Others (18)				
88	PLLCNT	I	PLL operational setting LOW(GND):20MHz /HIGH(LV_{DD}): 16MHz input	
87	V_C	O	Internal V_{CO} control pin	
86	XPLLPD	I	PLL power-down pin LOW(GND): PLL power down mode/HIGH(LV_{DD}): PLL operation	
81	EXCLK	I	5V level external clock input pin (connected to LOW (GND) when not used)	
83	CLKSEL0	I	Input clock selection LOW(GND): OSCIN/HIGH(LV_{DD}): EXCLK input	
82	CLKSEL1	I	System clock selection LOW(GND): PLL output//HIGH(LV_{DD}): CLKSEL0 selection signal	
79	OSCIN	I	Input to built-in oscillation circuit (40MHz, 20MHz or 16MHz) Connected to LOW (GND) when EXCLK pin is used	
77	OSCOUT	O	Output from built-in oscillation circuit	
31	TESTEN	Ipd	Pin for testing (connected to LOW (GND) usually)	
35	TESTMON	O	Monitor output for testing (open “LOW” output usually)	Drive capability 2mA
27	XRESET	Ipu	System reset input signal	
4,123,115, 103,99,95	NC	—	Not connected to IC chips (open usually)	
HV_{DD}:5V (13)				
127,122,118, 113,110,106, 91,74,66,61, 51,43,34,19, 7,2	HV_{DD}	P	Power supply for 5V interface	
LV_{DD}:3.3V (6)				
97,85,78,65, 33,1	LV_{DD}	P	Power supply for internal operation	
V_{SS}:0V (17)				
128,125,120, 116,112,108, 104,100,96, 93,89,84,80, 76,70,64,56, 46,32,24,14, 5	V_{SS}	P	GND	

Note : I : Input
Is : Schmitt input
Ipu : Pull-up input
Ispu : Pull-up Schmitt input
Ipd : Pull-down input
O : Output
Ood : Open drain output
Otr : Tristate output

6. FUNCTIONAL DESCRIPTION

6.1 CPU Interface Circuit

This block can be interfaced to a general-purpose CPU. It controls the interface with the CPU generally.
If XCS signal from CPU is LOW, the block can access the internal register. It decodes the address bus AD5 to AD0 to generate the address of the internal register. At this time, it generates the read/write strobe signal from XRD/XWR signal, transferring data between the internal register. A wait signal to CPU is not generated because of no-wait operation for register area other than those that are operated by collaboration with IDE bus.

6.2 Internal Registers

Refer to the section of Register Functions as for the addresses of the internal registers and description of each bit. The main functions of this block are as follows:

(1) It generates control signals to each block according to the address, write-data and write-strobe signals generated by the CPU interface circuit.
(2) It stores the status signals from each block, and outputs data according to the address and read-strobe signals sent from the CPU interface circuit.

6.3 IDE Interface Circuit

This is a block controlling the IDE interface. It has the following functions:

(1) It is an access to IDE bus by CPU program mode. For the access to CPU data port, the PIO mode can be only used.
(2) It can monitor various kind of signals of the IDE interface.
(3) It controls the linkage operation of each functional block according to the control signal and the stop-operation signal sent from the DMA control circuit.
(4) It control the transfer status at the time of DMA mode of IDE according to HDMARQ/XHDMACK signals.
(5) It reads/writes the data of the data bus DD15-0 of the IDE from/to FIFO in SCSI-2 block. If the transfer becomes impossible because of FIFO's full/empty state, the block suspends transfer to and from the port according to the timing specified by the XHIOR/XHIOW signals.
(6) The following operations can be used as a general-purpose port as well as IDE.
(7) The port allows selection of bit width 8 or 16.
(8) The port interface allows selection of the master or slave function (toward HDMARQ/XHDMACK/XHIOR/XHIOW direction).

6.4 DMA Control Circuit

This is a block which controls the transfer between IDE interface and FIFO in SCSI-2 block. It has the following functions:

(1) It controls the linkage operation of each functional block according to the control signal from the internal register and the information and stop-operation signals from each block.
(2) It stores the status of each of functional blocks when their linkage operation ends, reporting it to the internal register at the specified timing.

6.5 SCSI-2 Interface Circuit

This is a block which controls the interfaces conforming to the SCSI-2 standard in general. It has the following functions:

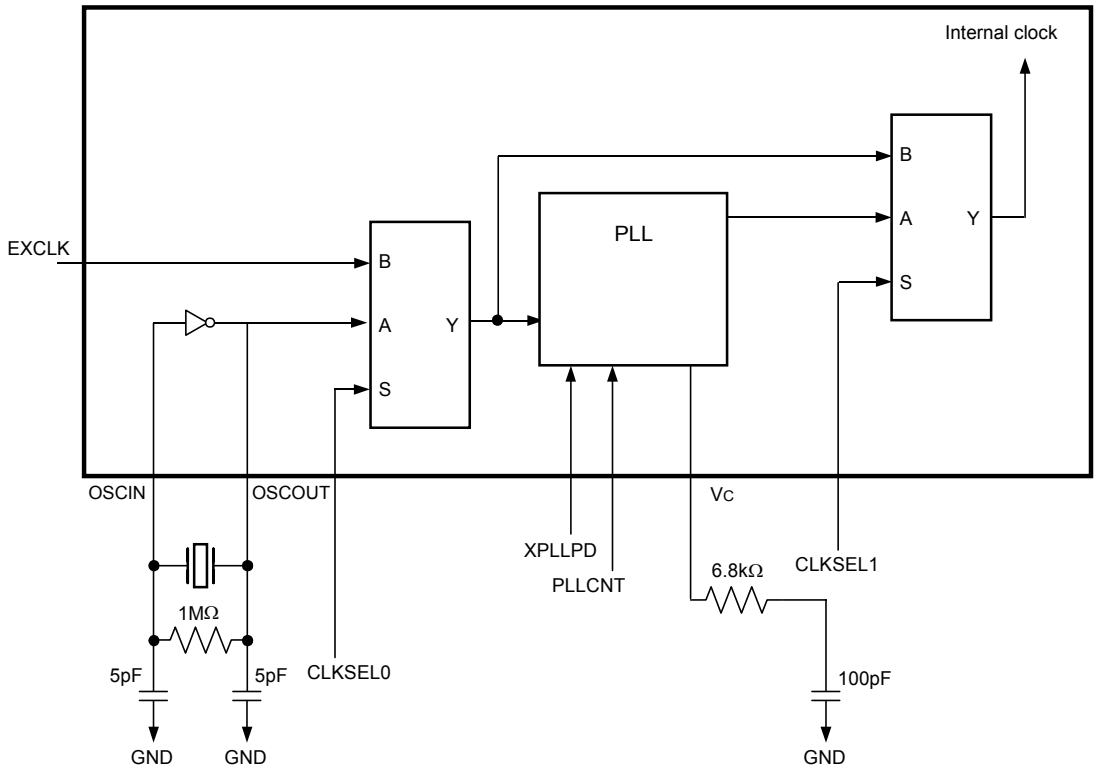
(1) It performs the SCSI protocol control automatically with hardware.
(2) It has 16-staged off-set counter to control the off-set and transfer rate during synchronous transfer.
(3) In the command phase, it distinguishes automatically the groups of commands received (in Target mode).
(4) It controls the automatic status/message transfer function. It supports the messages 00h/0Ah/0Bh. (In Target mode)

SCAM compatibility
Besides the traditional SCSI, this LSI has some additional functions compatible to SCAM (SCSI Configured Auto Magnify) as listed below.
These functions allow a device to operate as a SCAM Lv.1 drive.

(1) It monitors and recognizes SCAM selection and generates interruption.
(2) It responds to the selection response delay of 4ms or more, so it can distinguish SCAM selection from ordinary selection.
(3) It can operate SCSI bus's signal line directly because of its actual operation responding to SCAM selection and sending/receiving data.

6.6 PLL Circuit (Internal System Clock Generating Section)

This IC has the function to generate 40MHz required for the internal circuit from the clock generated by the oscillation circuit or inputted from EXCK pin by using PLL circuit.



- The IC allows to structure a oscillation circuit easily by connecting ceramic vibrator and feedback resistance. (As for the characteristics of the ceramic vibrator, please consult with us separately.)
- It allows oscillation of 16MHz/20MHz/40MHz by means of the oscillation circuit mentioned above.
- It allows inputting external clock of 5V level and 16MHz/20MHz/40MHz from the EXCLK pin.
- It allows inputting external clock of 3.3V level and 16MHz/20MHz/40MHz from the OSCIN pin.
- No PLL circuit is used because the internal clock can get necessary clock if the oscillator section is oscillated at 40MHz, or if 40MHz clock is input from EXCLK pin.
- If EXCLK pin is used, set OSCIN pin to LOW; if OSCIN pin is used, set EXCLK pin to LOW.

Make settings as shown below depending on the ways of use (1:LV_{DD} , 0:V_{SS}):

	If oscillator circuit is used			If external clock of 5V level is input from EXCLK pin		
Oscillation/ input clock	16MHz	20MHz	40MHz	16MHz	20MHz	40MHz
CLKSEL0	0	0	0	1	1	1
XPLLPD	1	1	0	1	1	0
PLLCNT	1	0	0	1	0	0
CLKSEL1	0	0	1	0	0	1

- PLL circuit specifications Ta = 0 to 70°C LV_{DD}=3.3V±0.3V

Item	Specifications
Lock-up time	Within 1ms after oscillation was stabilized
Jitter	Within ±2ns

7. FUNCTION OF REGISTERS

7.1 List of Registers

Address	Register name	Abridged name
00h	Main Interrupt Status	MAININT
01h	SCSI Interrupt Status 1	SCSIINT1
02h	SCSI Interrupt Status 2	SCSIINT2
03h	- Reserved -	—
04h	- Reserved -	—
05h	- Reserved -	—
06h	- Reserved -	—
07h	RESET	RESET
08h	- Reserved -	—
09h	SCSI Mode0	SCSIMODE0
0Ah	SCSI Mode1	SCSIMODE1
0Bh	SCSI Control	SCSICTL
0Ch	SCSI DATA	SCSIDATA
0Dh	SCSI Synchronous data transfer Mode	SYNCMODE
0Eh	SCSI Own ID	OWNID
0Fh	SCSI Source/Destination ID	SDID
10h	SCSI Selection Timeout Counter	SELTIME
11h	SCSI FIFO Control	FIFOCTL
12h	SCSI FIFO Data	FIFODATA
13h	SCSI Non-DMA Transfer Size	NDMASIZE
14h	SCSI Command	COMMAND
15h	- Reserved -	—
16h	- Reserved -	—
17h	DMA Control	DMACTL
18h	- Reserved -	—
19h	Host Transfer Byte Count2	HTBC2
1Ah	Host Transfer Byte Count1	HTBC1
1Bh	Host Transfer Byte Count0	HTBC0
1Ch	Config0	CONFIG0
1Dh	Config1	CONFIG1
1Eh	Test	TEST
1Fh	Revision	REVISION

Address	Register name	Abridged name
20h	IDE Bus Status	IDESTS
21h	IDE Bus Control	IDECTL
22h	IDE Bus Register Mode	IDERMOD
23h	IDE Bus Transfer Mode	IDETMOD
24h	IDE Bus Ultra-DMA Transfer Mode	IDEUMOD
25h	- Reserved -	—
26h	- Reserved -	—
27h	CRC Control	CRCCTL
28h	CRC Data Upper	CRCU
29h	CRC Data Lower	CRCL
2Ah	- Reserved -	—
2Bh	- Reserved -	—
2Ch	- Reserved -	—
2Dh	- Reserved -	—
2Eh	- Reserved -	—
2Fh	- Reserved -	—
30h 31h 32h 33h 34h 35h 36h 37h	IDE to CS0 (Command Block Register)	IDE to CS0
38h 39h 3Ah 3Bh 3Ch 3Dh 3Eh 3Fh	IDE to CS1 (Command Block Register)	IDE to CS1

7.2 List of Registers/Bits

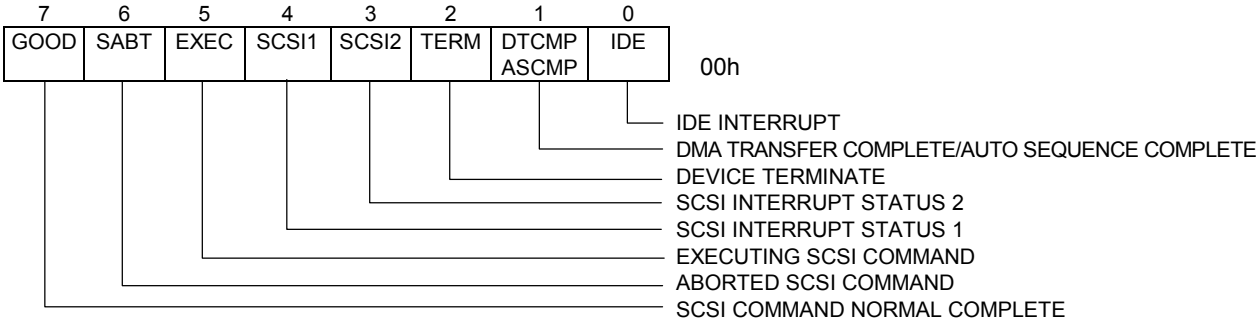
Address	Type	Register Name	Default value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
00h	R/W	MAININT	00h	GOOD	SABT	EXEC	SCSI1	SCSI2	TERM	DTCMP/ ASCMP	IDE
01h	R/W	SCSIINT1	00h	SPERR	IDERR	SELTO	SATN	BFREE	ILPHS	SCSEL	WOATN
02h	R/W	SCSIINT2	00h	—	SRST	OFERR	UNDEF	CMDER	RESEL	SEL	LARBT
03h	—	—	—	—	—	—	—	—	—	—	—
04h	—	—	—	—	—	—	—	—	—	—	—
05h	—	—	—	—	—	—	—	—	—	—	—
06h	—	—	—	—	—	—	—	—	—	—	—
07h	W	RESET	—	—	—	—	—	—	—	—	—
08h	—	—	—	—	—	—	—	—	—	—	—
09h	R/W	SCSIMODE0	80h	SINTEN	DTCD	—	ULTRAS	AUTO1	AUTO2	AN_C	AN_D
0Ah	R/W	SCSIMODE1	00h	STPPE	ATNPE	STATN	AUTO	RINH	SINH	DACS	SPCEN
0Bh	R/W	SCSICTL	00h	ACK	ATN	SEL	BSY	REQ	MSG	I/O	C/D
0Ch	R/W	SCSIDATA	00h	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0Dh	R/W	SYNCMODE	00h	RATE3	RATE2	RATE1	RATE0	OFF3	OFF2	OFF1	OFF0
0Eh	R/W	OWNID	00h	—	—	—	—	—	OID2	OID1	OID0
0Fh	R/W	SDID	00h	—	SID2	SID1	SID0	—	DID2	DID1	DID0
10h	R/W	SELTIME	00h	ST7	ST6	ST5	ST4	ST3	ST2	ST1	ST0
11h	R/W	FIFOCTL	01h	—	—	—	—	—	FCLR	FULL	EMPTY
12h	R/W	FIFODATA	xxh	FD7	FD6	FD5	FD4	FD3	FD2	FD1	FD0
13h	R/W	NDMASIZE	FFh	NSZ7	NSZ6	NSZ5	NSZ4	NSZ3	NSZ2	NSZ1	NSZ0
14h	R/W	COMMAND	00h	CMD7	CMD6	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0
15h	—	—	—	—	—	—	—	—	—	—	—
16h	—	—	—	—	—	—	—	—	—	—	—
17h	R/W	DMACTL	00h	—	—	—	—	—	—	FIFO	DTGO
18h	—	—	—	—	—	—	—	—	—	—	—
19h	R/W	DTBC2	00h	DBC23	DBC22	DBC21	DBC20	DBC19	DBC18	DBC17	DBC16
1Ah	R/W	DTBC1	00h	DBC15	DBC14	DBC13	DBC12	DBC11	DBC10	DBC9	DBC8
1Bh	R/W	DTBC0	00h	DBC7	DBC6	DBC5	DBC4	DBC3	DBC2	DBC1	DBC0
1Ch	R/W	CONFIG0	00h	ACP	INTLV	PSLV	WTDSA	DRQLV	SWAP	ODS	BUS8
1Dh	R/W	CONFIG1	00h	—	—	—	—	—	—	—	INTLK
1Eh	R	TEST	00h	TM2	TM1	TM0	CKOUT	—	OFST	SCBC	DMBC
1Fh	R	REVISION	00h	REV7	REV6	REV5	REV4	REV3	REV2	REV1	REV0

Address	Type	Register Name	Default value	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
20h	R	IDESTS	xxh	DMARQ	DMACK	INTRQ	IORDY	—	—	PDIAG	DASP
21h	R/W	IDECTL	00h	RESET	IMASK	DTIE	DTSE	—	—	ULTRAD	DMA
22h	R/W	IDERMOD	00h	AP3	AP2	AP1	AP0	NP3	NP2	NP1	NP0
23h	R/W	IDETMOD	00h	AP3	AP2	AP1	AP0	NP3	NP2	NP1	NP0
24h	R/W	IDEUMOD	00h	—	—	—	—	CYC3	CYC2	CYC1	CYC0
25h	—	—	—	—	—	—	—	—	—	—	—
26h	—	—	—	—	—	—	—	—	—	—	—
27h	R/W	CRCCTL	00h	CRCCL	—	—	—	—	—	—	—
28h	R	CRCU	4Ah	CRC15	CRC14	CRC13	CRC12	CRC11	CRC10	CRC09	CRC08
29h	R	CRCL	BAh	CRC07	CRC06	CRC05	CRC04	CRC03	CRC02	CRC01	CRC00
2Ah	—	—	—	—	—	—	—	—	—	—	—
2Bh	—	—	—	—	—	—	—	—	—	—	—
2Ch	—	—	—	—	—	—	—	—	—	—	—
2Dh	—	—	—	—	—	—	—	—	—	—	—
2Eh	—	—	—	—	—	—	—	—	—	—	—
2Fh	—	—	—	—	—	—	—	—	—	—	—
30h	R/W	IDE-CS0	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
31h	R/W	IDE-CS0	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
32h	R/W	IDE-CS0	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
33h	R/W	IDE-CS0	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
34h	R/W	IDE-CS0	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
35h	R/W	IDE-CS0	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
36h	R/W	IDE-CS0	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
37h	R/W	IDE-CS0	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
38h	R/W	IDE-CS1	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
39h	R/W	IDE-CS1	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
3Ah	R/W	IDE-CS1	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
3Bh	R/W	IDE-CS1	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
3Ch	R/W	IDE-CS1	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
3Dh	R/W	IDE-CS1	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
3Eh	R/W	IDE-CS1	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
3Fh	R/W	IDE-CS1	xxh	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0

7.3 Detailed Description of Each Register

7.3.1 Main Interrupt Status (MAININT) R/W

When the IC interrupted CPU, the CPU first reads this register for processing the interruption to get to know which interrupt status register is the factor.
After reading this register, the CPU reads the interrupt status register corresponding to each bit to find out the bit that is the source of interrupt, and processes the interruption appropriately. Then it writes the values read to the interrupt status registers corresponding to each bit, then clearing the bits.
If GOOD, SABT, DTCMP, or IDE bit is the interrupt source, the CPU writes the value read to clear the bits. The register has no need to clear directly any other bits.
Moreover, when writing any value to this register, the XINT pin is negated, then the XINT pin is asserted again after 400 ns if any interrupt factor is shown (for a CPU that has edge trigger type interrupt function).

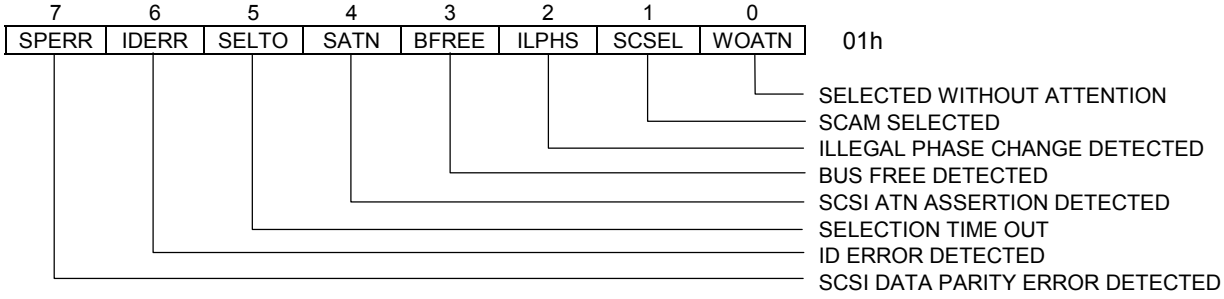


BIT7 SCSI COMMAND NORMAL COMPLETE
This bit becomes HIGH if a SCSI control command closed normally.
BIT6 ABORTED SCSI COMMAND
This bit becomes HIGH if a control command was forced to terminate by Abort command issued.
BIT5 EXECUTING SCSI COMMAND
This bit becomes HIGH while a SCSI control command is under execution. This bit is not a factor causing interrupt to the CPU, so HIGH of this bit causes no interruption. It is used to monitor the execution of SCSI control command.
BIT4 SCSI INTERRUPT STATUS 1
This bit becomes HIGH if any interrupt factor about the SCSI interface is shown on SCSIINT1 register.
BIT3 DMA INTERRUPT STATUS 2
This bit becomes HIGH if any interrupt factor about the SCSI interface is shown on SCSIINT2 register.
BIT2 DEVICE TERMINATE
Sets to "1" and suspends transfer at the same time of DTCMP bit when the device negated HDMAREQ during IDE-Ultra transfer. Then, abotes the SCSI command published in the firm.
Writing "1" to this bit clears the port.
BIT1 DMA TRANSFER COMPLETE / AUTO SEQUENCE COMPLETE
This bit becomes HIGH when DMA data transfer activated by DMACTL register ends.
It becomes HIGH also when the transfer is forced to terminate by "0" written in DTGO bit of DMACTL register.
It becomes HIGH also when a command is aborted by Abort_SCSI command, when a command under execution is aborted by ATN assertion, or when the transfer terminate is sequenced at the time of Ultra-DMA of IDE, after DTGO bit of DMACTL register was set because DMA terminates.
When FIFO bit of DMACTL register is set and DMA is activated, this bit becomes HIGH when transfer between CPU and FIFO in SCSI is terminated by the number of bit counts set to the DTBC register. However, access to FIFO of IDE of CPU does not affect this bit.

This bit becomes HIGH when AUTO1 or AUTO2 bit of SCSIMODE0(09h) is set and the command processing specified is over. This bit is a common bit between DTCMP and ASCMP, so use DTCD bit with set to "Hi" if this bit is used as ASCMP.
BIT0 IDE INTERRUPT
This bit becomes HIGH when the rising edge of HINTRQ pin on the IDE interface was detected.

7.3.2 SCSI Interrupt Status 1 (SCSIINT1) R/W

Shows the result of a SCSI control command executed.
The CPU can recognize the interrupt source by reading this register after receiving the interrupt signal. It clears the bit by writing again the value read.



BIT7 SCSI DATA PARITY ERROR DETECTED

This bit becomes HIGH if a parity error was detected on SCSI data bus.

BIT6 ID ERROR DETECTED

This bit becomes HIGH if an error was detected about ID bit during the selection or reselection phase. The error about ID bit shows that:

- Only one ID bit is asserted. or,
- Three or more ID bits are asserted.

BIT5 SELECTION TIME OUT

This bit becomes HIGH if time-out was detected during the selection or reselection phase.

BIT4 SCSI ATN ASSERTION DETECTED

This bit becomes HIGH if SCSI ATN was asserted. It is not set, though, in the sequence where SCSI ATN is asserted usually, such as the message-out phase subsequent to selection.

BIT3 BUS FREE DETECTED

This bit becomes HIGH if SCSI control command detected the busfree phase during its execution.

BIT2 ILLEGAL PHASE CHANGE DETECTED

This bit becomes HIGH if a SCSI control command detected unexpected phase transition during its execution.
It is valid only in Initiator mode.

BIT1 SCAM SELECTED

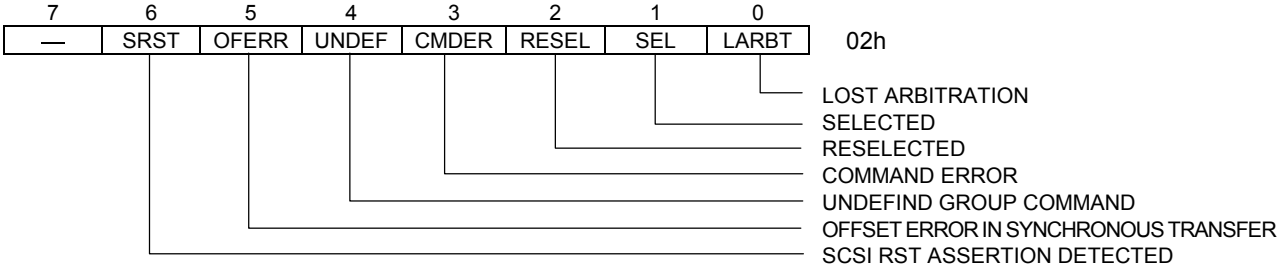
This bit becomes HIGH if SCAM selection was responded to.

BIT0 SELECTED WITHOUT ATTENTION

This bit becomes HIGH if the selection which does not assert ATTENTION was responded to.
Even if this bit is set, a SCSI control command continues execution. If a command block remains received, though, the first byte of SCSI-FIFO has the command code.

7.3.3 SCSI Interrupt Status 2 (SCSIINT2) R/W

Shows the result of a SCSI control command executed.
The CPU can recognize the interrupt source by reading this register after receiving the interrupt signal. It clears the bit by writing again the value read.



BIT7 RESERVED

BIT6 SCSI RST ASSERTION DETECTED

This bit becomes HIGH if SCSI RST was asserted.

BIT5 OFFSET ERROR IN SYNCHRONOUS TRANSFER

This bit becomes HIGH if an off-set error occurred during synchronous transfer. The off-set error means that the off-set counter is not reset to "0" when transfer ends, or that the counter overflows/underflows.

BIT4 UNDEFIND GROUP COMMAND

This bit becomes HIGH if SCSI commands other than group 0, 1, 2, or 5 were received.

BIT3 COMMAND ERROR

This bit becomes HIGH if an undefined SCSI control command was issued or a control command was issued during execution of another command.

BIT2 RESELECTED

This bit becomes HIGH if any other device made re-selection during execution of a command other than the SCSI control command which makes re-selection.

BIT1 SELECTED

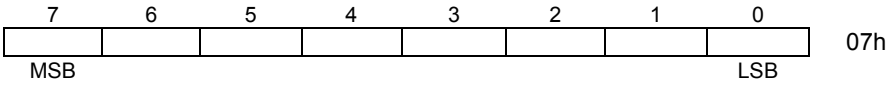
This bit becomes HIGH if any other device made selection during execution of a command other than SCSI control command which makes selection.

BIT0 LOST ARBITRATION

This bit becomes HIGH in the case of defeat in the arbitration phase. If this bit is HIGH and no other device made selection or re-selection, the IC suspends the operation of the control commands.

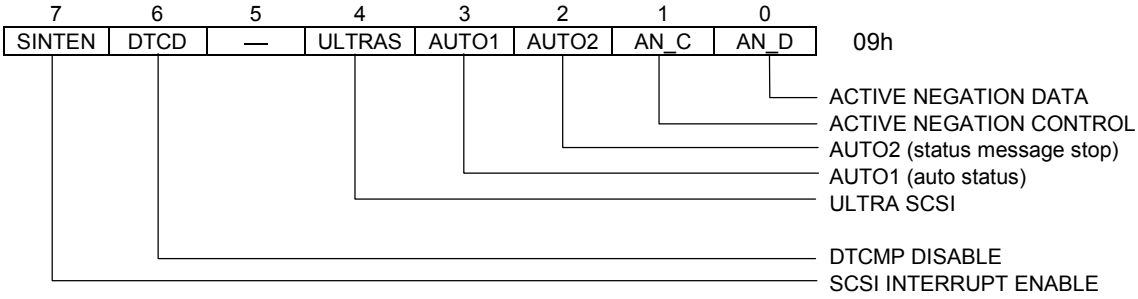
7.3.4 Reset (RESET) W

Writing in this register initializes the inside of the circuit. Any value may be entered.



7.3.5 SCSI Mode Select0 (SCSIMODE0) R/W

Makes the operational settings related to SCSI interface.



BIT7 SCSI INTERRUPT ENABLE

If this bit is HIGH, any SCSI interruption but DTCMP is enabled.

BIT6 DTCMP DISABLE

If this bit is HIGH, DTCMP interrupt is disabled.

If AUTO1 or 2 bit is used, setting this bit to HIGH causes MAININT bit 1 to be only ASCMP.

BIT5 RESERVED

BIT4 ULTRA SCSI

If this bit is HIGH, ULTRA-SCSI transfer is enabled. It is valid only when RATE bit of SYNCMODE register is set to "0" or "1".

BIT3 AUTO1 (auto status)

Executes automatically STS_MSG, Busfree, and Wait_SEL_CMD after DMA_DATA_IN/OUT command was executed.

At the end of execution, ASCMP interrupt occurs. The bit setting is valid also in FIFO-DMA mode. *AUTO

BIT2 AUTO2 (status message stop)

Executes automatically STS_MSG after DMA_DATA_IN/OUT command was executed.

At the end of execution, ASCMP interrupt occurs. The bit setting is valid also in FIFO-DMA mode. *AUTO

*AUTO : During the execution of AUTO1 or 2, AUTO bit of SCSIMODE1 register is deemed as "1".
Also, EXEC bit is "1" during the execution of AUTO1 or 2. Because internal sequencer writes the command in SCSI block for CPU, COMMAND register can read the command value then under execution.

BIT1 ACTIVE NEGATION CONTROL

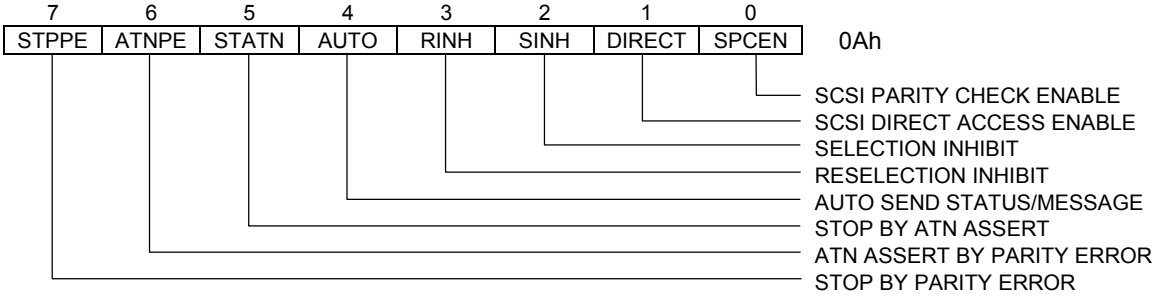
HIGH of this bit enables the function of active negation of SCSI REQ/ACK signals.

BIT0 ACTIVE NEGATION DATA

HIGH of this bit enables the function of active negation of the SCSI data and parity signals.

7.3.6 SCSI Mode Select1 (SCSIMODE1) R/W

Makes the operational settings related to SCSI interface.



BIT7 STOP BY PARITY ERROR

HIGH of this bit suspends a SCSI control command under execution if a parity error was detected on SCSI interface.

BIT6 ATN ASSERT BY PARITY ERROR

HIGH of this bit makes ATN asserted against a target device if a parity error was detected on SCSI interface. Any setting of this bit is invalid if SCSI parity check is disabled.

This bit is valid only in Initiator mode.

BIT5 STOP BY ATN ASSERT

HIGH of this bit suspends a SCSI control command under execution if ATN asserted was detected.

This bit is valid only in Target mode.

BIT4 AUTO SEND STATUS/MESSAGE

HIGH of this bit puts SCSI control command “Status_Message” into Automatic Transmission mode. In this mode, FLAG and LINK bits of SCSI command block which have been received are checked; Status 00h(GOOD) and message 00h(COMMAND COMPLETE), status 10h(INTERMEDIATE GOOD) and message 0Ah(LINKED COMMAND COMPLETE), and status 10h(INTERMEDIATE GOOD) and message 0Bh(LINKED COMMAND COMPLETE WITH FLAG) are sent automatically if LINK bit is LOW, LINK bit is HIGH and FLAG bit is LOW, and LINK and FLAG bits are both HIGH, respectively.

BIT3 RESELECTION INHIBIT

HIGH of this bit disables response to re-selection.

BIT2 SELECTION INHIBIT

HIGH of this bit disables response to re-selection.

BIT1 SCSI DIRECT ACCESS ENABLE

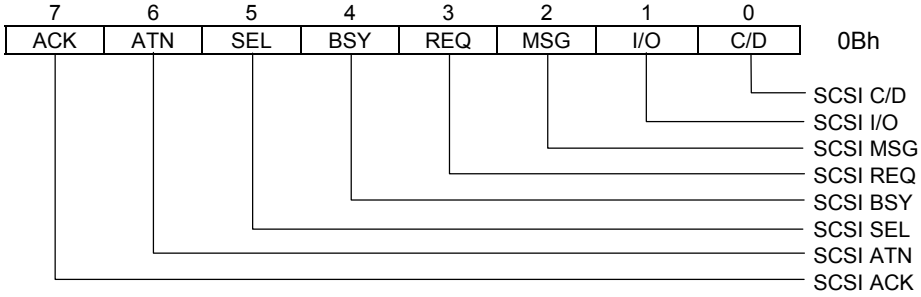
HIGH of this bit allows direct control of SCSI signal lines from CPU by using SCSI data register and SCSI control register. Also, it is possible always to monitor the status of the signal lines irrespective of the status of this bit.

BIT0 SCSI PARITY CHECK ENABLE

HIGH of this bit causes parity check of SCSI data bus during the selection phase (when itself is selected) and when data is input through SCSI.

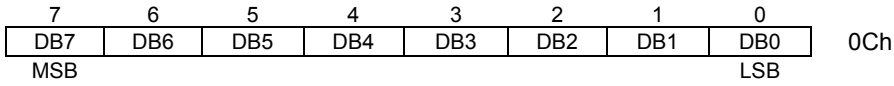
7.3.7 SCSI Control (SCSICTL) R/W

This register is accessed when CPU controls SCSI signal lines directly.
For such direct control, DIRECT (bit 1) must be set in the mode setting register (0Ah).
The status of each signal is stored as “active HIGH”.



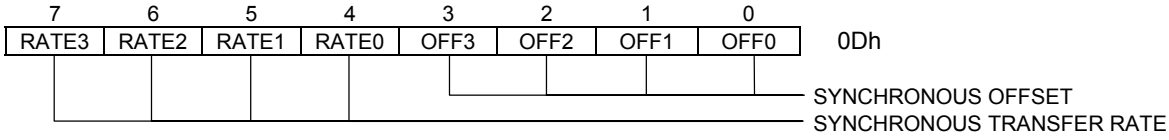
7.3.8 SCSI Data (SCSIDATA) R/W

The CPU accesses this register when it controls SCSI data bus directly. For such direct control, DIRECT (bit 1) must be set in the mode setting register (0Ah).
The status of each signal is stored as “active HIGH”. DIRECT setting does not decide whether parity bit is output or not; it is output if it has been output before setting DIRECT, or it is not otherwise.



7.3.9 Synchronize Transfer Mode (SYNCMODE) R/W

Sets transfer rate and off-set for SCSI synchronous transfer.



RATE3-0	ASSERT	NEGATE	PERIOD
0000	1T	1T	2T
0001	2T	1T	3T
0010	2T	2T	4T
0011	3T	2T	5T
0100	3T	3T	6T
0101	4T	3T	7T
0110	4T	4T	8T
0111	5T	4T	9T
1000	5T	5T	10T
1001	6T	5T	11T
1010	6T	6T	12T
1011	7T	6T	13T
1100	7T	7T	14T
1101	8T	7T	15T
1110	8T	8T	16T
1111	9T	8T	17T

Note 2

OFF3-0	OFFSET
0000	Asynchronous
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

Note 1) T has double the cycle of internal clock (40MHz).
Note 2) If ULTRA bit of SCSIMODE0 register is set, T has the same cycle as internal clock (40MHz) only if the value set for RATE3-0 bit is 1 or less.

7.3.10 SCSI Own ID (OWNID) R/W

Sets the SCSI-ID of this IC itself.

7	6	5	4	3	2	1	0	
—	—	—	—	—	OID2	OID1	OID0	0Eh

7.3.11 Source/Destination ID (SDID) R/W

Sets both the SCSI-ID of the selector side and the target SCSI-ID when selection is made.

7	6	5	4	3	2	1	0	
—	SID2	SID1	SID0	—	DID2	DID1	DID0	0Fh

DESTINATION ID (R/W)
SOURCE ID (R)

In Initiator mode, sets the target SCSI-ID to be selected in DESTINATION ID.
When re-selection is received, the target SCSI-ID which made re-selection is set in SOURCE ID.
In Target mode, sets the initiator SCSI-ID to be re-selected in DESTINATION ID.
When selection is received, the initiator SCSI-ID which made selection is set in SOURCE ID.

7.3.12 Selection Timeout Counter (SLTIME) R/W

Sets time-out delay for selection and re-selection.

7	6	5	4	3	2	1	0	
ST7	ST6	ST5	ST4	ST3	ST2	ST1	ST0	10h

The time-out delay value is calculated according to the following formula:
Delay value = count value x $2^{15} \times T \times 2$
Where, T is internal clock cycle (40MHz).

The IC acts as follows if it detected time-out:
・Suspends to output ID bit.
・Negates XSSEL 4000×T×2 (about 200µs) after such suspension, and outputs selection time-out interrupt.
No time-out is detected if "0" is set to this register.

7.3.13 FIFO Control (FIFOCTL) R/W

Used for clearing the data of SCSI-FIFO and for checking its status.

7	6	5	4	3	2	1	0	
—	—	—	—	—	FCLR	FULL	EMPTY	11h

FIFO EMPTY
FIFO FULL
CLEAR FIFO

BIT7,6,5,4,3 RESERVED
BIT2 CLEAR FIFO
HIGH of this bit clears the data stored in SCSI-FIFO.
The bit returns to LOW automatically after such clearing.
BIT1 FULL
HIGH of this bit means that SCSI-FIFO is full. In such a condition, any data written in SCSI-FIFO is ignored.
BIT0 EMPTY
HIGH of this bit means that SCSI-FIFO is empty. In such a condition, any trial to read data from SCSI-FIFO results in invalid data read out.

7.3.14 FIFO Data (FIFODATA) R/W

This is a register to access SCSI-FIFO from CPU.

7	6	5	4	3	2	1	0	
FD7	FD6	FD5	FD4	FD3	FD2	FD1	FD0	12h

7.3.15 Non DMA Transfer Size (NDMASIZ) R/W

Sets the number of data transfer bytes in Non-DMA mode. In Read mode, the register allows to read out the size of data yet to be transferred.

7	6	5	4	3	2	1	0	
NSZ7	NSZ6	NSZ5	NSZ4	NSZ3	NSZ2	NSZ1	NSZ0	13h

7.3.16 SCSI Command (COMMAND) R/W

Sets SCSI control commands.

7	6	5	4	3	2	1	0	
CMD7	CMD6	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0	14h

Refer to [7.4 SCSI Control Commands] as for the details of each of them.

7.3.17 DMA Control (DMACTL) R/W

7	6	5	4	3	2	1	0	
—	—	—	—	—	—	FIFO	DTGO	17h

DTGO — DMA TRANSFER GO
FIFO — FIFO CONTROL

BIT7-1 RESERVED

BIT1 FIFO CONTROL

Setting this bit and DTGO to HIGH at the same time activates DMA, however, the control circuit in internal DMA does not operate.
CPU must transfer data between FIFO according to FULL/EMPTY status of FIFO. Alternately, it may first write data in FIFO and set HIGH in this bit and DTGO, then make control with remaining data and FULL/EMPTY
Though, CPU may not access FIFO reversely against the direction of transfer.
This bit is the mode that transfers between SCSI bus and CPU, so it does not operate to transfer at the IDE bus side.

This bit is used together with SCSI command for DMA Data In/Out. Note that FIFO is cleared when SCSI phase is switched to the data phase if DMA Data In/Out command was issued without setting this bit (FIFO is not cleared at the IDE side).
FIFO is not cleared if DMA Data In/Out command is issued after this bit was set. So it is possible that FIFO has data written beforehand.

BIT0 DMA TRANSFER GO

Setting this bit to HIGH causes DMA transfer to start between SCSI and IDE interfaces.
Setting this bit to HIGH is also allowed before enterring the data phase.

7.3.18 DMA Transfer Byte Count 2 (DTBC2) R/W

Sets the most significant byte of the byte-length (3 bytes) for DMA transfer.
Setting of DTBC2 to 0 allows the setting of the byte-length up to FFFFFFh.

7	6	5	4	3	2	1	0	
DBC23	DBC22	DBC21	DBC20	DBC19	DBC18	DBC17	DBC16	19h

7.3.19 DMA Transfer Byte Count 1 (DTBC1) R/W

Sets the second byte of the byte-length (3 bytes) for DMA transfer.

7	6	5	4	3	2	1	0	
DBC15	DBC14	DBC13	DBC12	DBC11	DBC10	DBC9	DBC8	1Ah

7.3.20 DMA Transfer Byte Count 1 (DTBC0) R/W

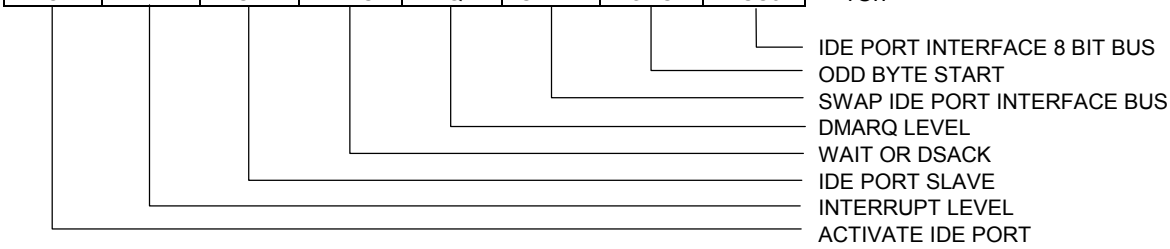
Sets the least significant byte of the byte-length (3 bytes) for DMA transfer.
When the data port of the IDE bus control section is used in word size (when it is used as IDE bus, etc), the uneven number byte is set to this register, and when ODS bit of CONFIG0 register is set, this LSI is automatically padding (data is indefinite) because 1 byte is lacking at the first or end transfer.

7	6	5	4	3	2	1	0	
DBC7	DBC6	DBC5	DBC4	DBC3	DBC2	DBC1	DBC0	1Bh

7.3.21 CONFIG0 (CONFIG0) R/W

Sets the operational mode of the IC.

7	6	5	4	3	2	1	0	
ACP	INTLV	PSLV	WTDSA	DRQLV	SWAP	ODS	BUS8	1Ch



BIT7 ACTIVATE IDE PORT
After reset, the IDE interface is in All Pins Input mode. Setting this bit to HIGH activates the port.

BIT6 INTERRUPT LEVEL
Decides the level of the signal interrupting CPU.
0: Active LOW (Output to XINT is 0/Hi-Z)
1: Active HIGH (Output to XINT is 1/0)

BIT5 IDE PORT SLAVE
Decides the operational mode of IDE interface section. When it is used at IDE bus compatibility, set it to LOW.
DMA bit of IDECTL register is reflected in Master mode.
22h or later register functions cannot be used in Slave mode.
0: Master mode (HDMARQ = input; XHDMACK/XHIOR/XHIOW = output)
1: Slave mode (HDMARQ = output; XHDMACK/XHIOR/XHIOW = input)

BIT4 WAIT OR DSACK
Decides the operation of XWAIT signal to CPU.
0: WAIT mode (Negates XWAIT signal when preparation for the register data is completed.)
1: DSACK mode (Asserts XWAIT signal when preparation for the register data is completed.)

BIT3 DMARQ LEVEL
Decides the operational level of HDMARQ signal. When it is used at IDE bus compatibility, set it to LOW.
0: Positive logic
1: Negative logic

BIT2 SWAP IDE PORT INTERFACE BUS
Swaps the higher 8 bits with lower ones when the IDE interface is used with 16-bit width.
Also reverses access order to 30h address of IDE-CS0 register.
0: Lower 8 bits data is firstly transferred to SCSI bus.
1: Higher 8 bits data is firstly transferred to SCSI bus.

BIT1 ODD BYTE START
When this bit is set to HIGH, 8 bits data to be transferred later is firstly transferred, and the remaining 1 byte is discarded, by the SWAP setting when IDE interface is used with 16-bit width for data to be transferred to and from SCSI block.
It is effective only for the first one byte.

BIT0 IDE PORT INTERFACE 8 BIT BUS
Sets this bit to HIGH to use the IDE interface with 8-bit width.
Only the lower 8 bits are valid. The higher 8 bits are invalid.
Pull up the higher 8 bits to use it with 8-bit width.
Sets this bit to LOW to use it with IDE bus compatibility
If ULTRA bit of IDECTL is set, this bit is ignored.

* Operational settings of the port interface
Shown below is a list of the operational settings made by the bit setting. Moreover, it is used as IDE interface, bit5/3/0 of CONFIG register is not required to set.

1) Switching master/slave of the port by PSLV bit

	HDMARQ	XHDMACK	XHIOR/XHIOW	Remarks
PSLV=0 (Master)	Input	Output	Output	Data input for XHIOR, data output for XHIOW Setting of STB3 to 0 valid XHIOR/XHIOW minimum pulse width: Assert≥40ns Negate≥40ns
PSLV=1 (Slave)	Output	Input	Input	Data output for XHIOR, data input for XHIOW Setting of STB3 to 0 invalid XHIOR/XHIOW minimum pulse width: Assert≥30ns Negate≥30ns

2) Switching operational modes by BUS8/SWAP/ODS bit

BUS8=0	SWAP=0	HDD7 to 0 is transferred to and from SCSI block first. If ODS = 1, HDD7 to 0 is discarded when the first one word is transferred, and only HDD15 to 8 is transferred. HDD7 to 0 is used if the last data to be transferred is not a word but a byte. The first data becomes HDD7 to 0 for the access to 30h address.
	SWAP=1	HDD15 to 8 is transferred to and from SCSI block first. If ODS = 1, HDD15 to 8 is discarded when the first one word is transferred, and only HDD7 to 0 is transferred. HDD15 to 8 is used if the last data to be transferred is not a word but a byte. The first data becomes HDD15 to 8 for the access to 30h address.
BUS8=1		Only HDD7 to 0 is used for transfer. HDD15 to 8 is used for Input mode (connect to GND). Only HDD7 to 0 is used for access to 30h address.

7.3.22 CONFIG1 (CONFIG1) R/W

Sets the operational mode of the IC.

7	6	5	4	3	2	1	0	
—	—	—	—	—	—	—	INTLK	1Dh

PORT INTERLOCK

BIT0 PORT INTERLOCK

Valid only in Master DMA mode

0: Negates XHDMACK if SCSI-FIFO cannot be transferred.

1: No XHDMACK is negated under the condition of SCSI-FIFO. However, negates it when HDMARQ is disappeared.

7.3.23 Test (TEST) R

Used for testing a LSI. Basically, writing in this register is prohibited.

7	6	5	4	3	2	1	0	
TM2	TM1	TM0	CKOUT	—	OFST	SCBC	DMBC	1Eh

7.3.24 Revision Reg. (REVISION) R

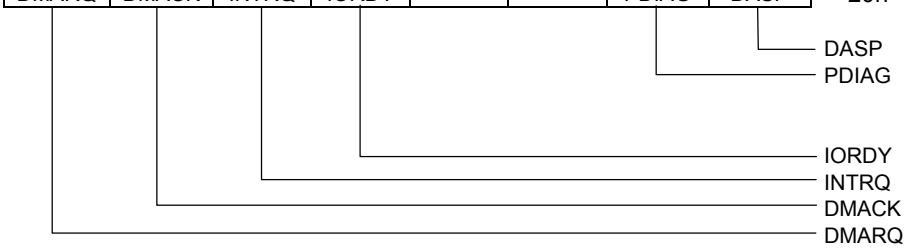
Shows the revision No. of the IC.

7	6	5	4	3	2	1	0	
REV7	REV6	REV5	REV4	REV3	REV2	REV1	REV0	1Fh

7.3.25 IDE Bus Statue (IDESTS) R

Shows the status of IDE interface signal

7	6	5	4	3	2	1	0	
DMARQ	DMACK	INTRQ	IORDY	—	—	PDIAG	DASP	20h



BIT7 DMARQ

Shows the status of HDMARQ signal with positive logic (the status of DRQLV bit of CONFIG register is reflected).

BIT6 DMACK

Shows the status of XHDMACK signal with positive logic.

BIT5 INTRQ

Shows the status of HINTRQ signal with positive logic.

BIT4 IORDY

Shows the status of HIORDY signal with positive logic.

BIT3-2 RESERVED

BIT1 PDIAG

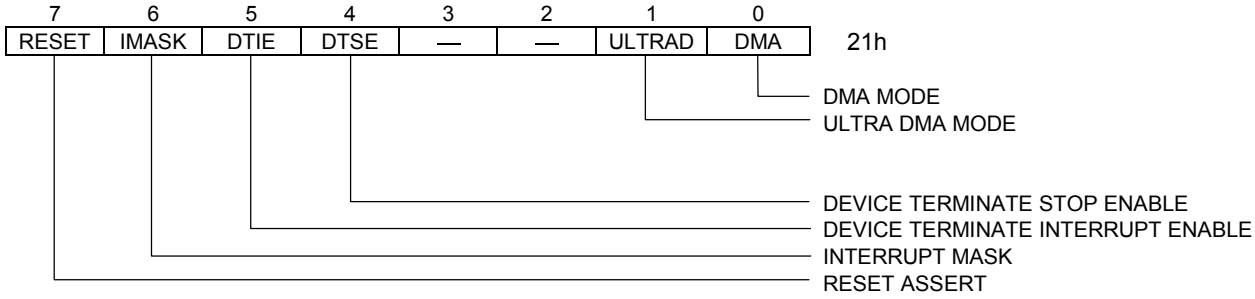
Shows the status of XHPDIAG signal with positive logic.

BIT0 DASP

Shows the status of XHDASP signal with positive logic.

7.3.26 IDE Bus Control (IDECTL) R/W

Shows the status of IDE interface signal



BIT7 RESET ASSERT

Writing HIGH to this bit causes RESET signal to be asserted to IDE interface between 50µs.
HIGH of this bit is read out during XHRESET assert.
When reset is executed during assert, XHRESET between 50µs is output from the time of reset.

BIT6 INTERRUPT MASK

Masks interruption to CPU due to detection at the riding edge of HINTRQ pin of IDE interface.
No display to MAININT register is executed.

BIT5 DEVICE TERMINATE INTERRUPT ENABLE

Interrupt (TERM) permission bit when device is terminated This bit is permitted at HIGH

BIT4 DEVICE TERMINATE STOP ENABLE

Operational stop permission bit when device is terminated. This bit is permitted at HIGH.
When this bit is LOW, do not end DMA command even if device is terminated, and restarts internal ULTRA sequencer until DTBC=0.

BIT3-2 RESERVED

BIT1 ULTRA DMA MODE

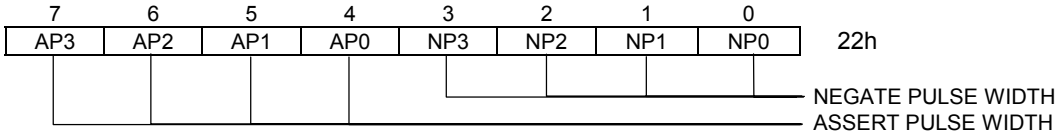
Sets DMA Transfer mode to ULTRA-DMA when Bit0 is set.
Uses it with a combination of DMA bit. BUS8 bit setting is ignored.

BIT0 DMA MODE

Executes Transfer mode in DMA when data is transferred in collaboration with SCSI and IDE. Transfers the data in PIO mode when this bit is not set.

7.3.27 IDE Register Mode (IDERMOD) R/W

Sets Transfer mode when accessing the register area of IDE interface
This command is valid againt 31h to 3Fh access of IDE-CS0 /1 register



BIT7-4 ASSERT PULSE WIDTH

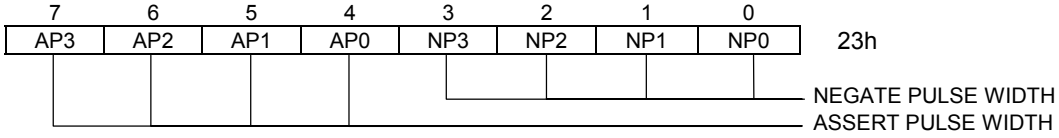
Decides the minimum value of assert period of the strobe signal when accessing the register area of IDE interface.
The width is the internal operation clock cycle (40MHz) multiplied by [(AP3 - 0)+2].

BIT3-0 NEGATE PULSE WIDTH

Decides the minimum value of negate period of the strobe signal when accessing the register area of IDE interface.
The width is the internal operation clock cycle (40MHz) multiplied by [(NP3 - 0)+2].
ex 0000: 2×25ns=50ns
0001: 3×25ns=75ns

7.3.28 IDE Transfer Mode (IDETMOD) R/W

Sets Transfer mode when transferring the data by IDE interface.
This command is also valid againt 30h access of IDE to CS0 register.
It is common to PIO /DMA mode.



BIT7-4 ASSERT PULSE WIDTH

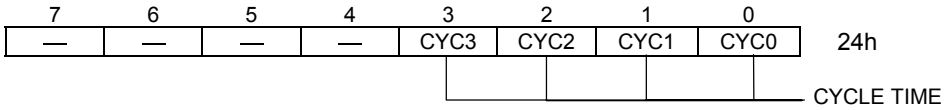
Decides the minimum value of assert period of the strobe signal when transferring the data by IDE interface.
The width is the internal operation clock cycle (40MHz) multiplied by [(AP3 to 0)+2].

BIT3-0 NEGATE PULSE WIDTH

Decides the minimum value of negate period of the strobe signal when transferring the data by IDE interface.
The width is the internal operation clock cycle (40MHz) multiplied by [(NP3 to 0)+2]
ex 0000: 2×25ns=50ns
0001: 3×25ns=75ns

7.3.29 IDE Ultra-DMA Transfer Mode (IDEUMOD) R/W

Sets Transfer mode when transferring the data with Ultra-DMA by IDE interface.

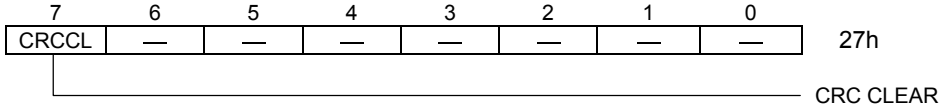


BIT3-0 CYCLE TIME

Decides the minimum cycle time of strobe signal when transferring the data with Ultra-DMA by IDE interface.
The width is the internal operation clock cycle (40MHz) multiplied by [(CYC3 to 0)+2].
ex 0000: 2×25ns=50ns
0001: 3×25ns=75ns

7.3.30 CRC Control (CRCCTL) R/W

Controls CRC when transferring the data with Ultra-DMA by IDE interface.

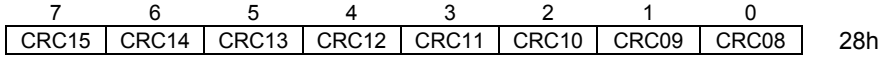


BIT7 CRC CLEAR

Initializes the internal CRC calculation circuit. It is automatically initialized in the internal circuit when DMA is activated.

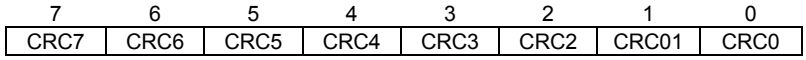
7.3.31 CRC Upper (CRCU) R

Shows the upper of CRC calculation result when transferring the data with Ultra-DMA by IDE interface.



7.3.32 CRC Lower (CRCL) R

Shows the lower of CRC calculation result when transferring the data with Ultra-DMA by IDE interface.



7.3.33 IDE-CS0 (IDE-CS0) R/W

Accesses this area when CPU accesses the data port of IDE interface.
Transfer mode is fixed at PIO mode, accesses it under the condition specified to IDETMOD register.
Setting of BUS8/SWAP bit of CONFIG register is reflected, so accesses to this register twice enable IDE bus to be 16-bit accessed if 16-bit width is used.
If 8 bit mode is used, access to this register once enables IDE bus to be accessed.

7	6	5	4	3	2	1	0	
DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0	30h

7.3.34 IDE-CS0/1 (IDE-CS0/1) R/W

Accesses this area when CPU accesses the register area of IDE interface.
Transfer mode is fixed at PIO mode, and accesses it under the condition specified to IDERMOD register.
Transfer is fixed at 8 bits, and DD7-0 of bus signal is used.

7	6	5	4	3	2	1	0	
DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0	31 to 3Fh

*Note: Access of 30h is prohibited during DMA transfer.
Accesses 31h - 3fh, and then once negates HDMARQ to enter CPU access when INTLK bit is not standing in DMA mode. XHDMACK is negated by HDMARQ off or transfer termination when INTLK bit is standing or during Ultra DMA, so be sure that you should wait for CPU access.

7.4 SCSI Control Commands

7.4.1 Control Commands and Command Codes

Code	Command names	Summary of commands
00h		Reserved
01h	Abort_SCSI	SCSI Abort command
02h		Reserved
03h		
04h	Assert_RST	SCSI Bus Clear command
05h	Busfree	
06h		Reserved
07h	Assert_ATN	
08h	SEL_MSG_clear	SCAM control commands
09h	Select_WithoutATN	Connection system commands
0Ah	Select_WithATN_Commannd	
0Bh	SelectWithoutATN_Command	
0Ch	Wait_Selection_Command	
0Dh	Reselection	
0Eh	Wait_Reselection	
0Fh	Wait_SCAM_Selection_Command	
10h		Reserved
11h	Negate_ACK	Transfer system commands
12h	Command_Out	
13h	DMA_Data_Out	
14h	Non-DMA_Data_Out	
15h	DMA_Data_In	
16h	Non-DMA_Data_In	
17h	Status_In	
18h	Message_In	
19h	Message_Out	
1Ah	Status_Message	

7.4.2 Description of Each Control Command

●Abort_SCSI (01H)
Suspends a SCSI control command under execution. After suspension, this command sets SABT bit of MAININT register and the status block, causing interruption. This command, if issued in a non-operational condition, is ignored.

●Assert_RST (04H)
Asserts SCSI RST signal (XSRST) for 768×T×2 (about 46uS), and then negates it.
This command releases all the signals it asserts, causing the busfree condition. The inside of the IC is not initialized.
It sets SCSIINT1 SRST bit after negating RST signal, causing interruption.
If a SCSI control command is under execution, it is forced to terminate. Only RST bit is set, though.
If SCSI DMA command is under execution, status block is set.

●Busfree(05H)
Executes busfree.
The command is valid only in Target mode. If issued in Initiator mode, it is ignored.
It is invalid if issued when other SCSI-type command is under execution, causing a command error and the command error interruption.

●Assert_ATN(07H)
Asserts SCSI ATN signal (XSATN).
The command is valid only in Initiator mode. If issued in Target mode, it is ignored.
Also, it is not asserted in the busfree condition, though no error occurs.
It causes no interruption after its execution.
Any other command under execution continues execution.
Negation of ATN occurs in any of the following cases:
・When the last byte is ACK-negated after Message_Out command was issued.
・If busfree was detected.
・If Assert_RST command was executed.
・If chip-reset was done.

●SEL_MSG_clear(08H)
Negates SCSI SEL/MSG signal (XSSEL/XSMSG).
When SCAM-selection was made by Wait_SCAM_Selection_Command, SCAM protocol is processed in Direct mode, with SEL/MSG remaining asserted inside. This command is used to clear it. After issuing this command, release Direct mode.
It causes no interruption after its execution.
Any other command under execution continues execution.

●Select_WithoutATN(09H)
Executes selection without asserting SCSI ATN signal.
This command is valid in either disconnected or connected condition. It causes a command error if issued when any other command is under execution.
After the command was issued, the IC acts as follows:
・Waits until SCSI bus becomes busfree.
・Enters arbitration after detecting busfree.
・If it wins arbitration, it asserts XSSEL and ID bit to data bus, entering the selection phase.
・It terminates selection and operation when its rival asserts XSBSY.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
After that, the IC enters Initiator mode.

●Select_WithATN_Command(0AH)
Asserts SCSI ATN signal, executes selection, and then executes the message-out command phase.
This command is valid in either disconnected or connected condition. It causes a command error if issued

when any other command is under execution.

CPU sets the message byte number in NON-DMA data-size register before issuing this command.
Then, the CPU write message data in FIFO. After transferring the message, it sets the command byte number in NON-DMA data-size register, writing the command data in FIFO.
The IC acts as follows:
・Waits for busfree.
・After detecting busfree, enters arbitration.
・If it wins arbitration, it asserts XSSEL and ID bit, entering the selection phase. Asserts XSATN at this time.
・After selection, it checks message-out at the timing when XSREQ is asserted, transferring messages in FIFO.
・Negates XSATN after asserting XSREQ and before asserting XSACK at the last byte of the messages.
・After transferring all the messages, it checks the command phase, detects data accumulated in FIFO, and transfers the command data in FIFO according to the byte number set anew.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
After that, the IC enters Initiator mode.
Caution: Be sure that data is written in FIFO after the number of bytes to be transferred was set.

●Select_WithoutATN_Command(0BH)
Executes selection keeping SCSI ATN negated, and then executes the command phase.
This command is valid in either disconnected or connected condition. It causes a command error if issued when any other command is under execution.

CPU issues this command after setting the command byte number in NON_DMA data-size register.
The command data is written in FIFO.
The IC acts as follows:
・Waits for busfree.
・Enters arbitration after detecting busfree.
・If it wins arbitration, it asserts XSSEL and ID bit, entering the selection phase.
・After selection, it checks the command phase at the timing when XSREQ is asserted, transferring command data from FIFO.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
After that, the IC enters Initiator mode.

●Wait_Select_Command(0CH)
Waits for the selection phase, and executes the command phase after selection.
Valid only when it is not connected.
If issued in the connected condition, it sets SCSIINT2 and CMDER bits, causing interruption. If any other command is under execution then, it continues execution.

If this command is issued, set STATN(bit5) of SCSIMODE register, and clear it when the command is over.
After the command was issued, the IC acts as follows:
① Waits for selection phase.
② If selected, checks XSATN. If it is not asserted, the IC acts as mentioned in ⑤.
If it is asserted, the IC sets the message-out phase, receiving a message.
③ If the message received is not “Identify”, the IC acts as mentioned in ⑥. (CPU checks the message in FIFO and responds to it.)
④ If XSATN remains asserted after 1-byte message (“Identify”) was received, the IC ends its operation by setting SATN bit of SCSIINT1 register and causing interruption.
If XSATN is negated,
⑤ The command phase is set to receive a command.
The IC distinguishes the command groups, deciding the number of bytes received automatically.
⑥ It sets GOOD bit of MAININT register, causing interruption.
After that, the IC enters Target mode.

●Reselect(0DH)
Executes the re-selection phase.
Valid only when it is not connected.
If issued in the connected condition, it sets SCSIINT2 and CMDER bits, causing interruption. If any other command is under execution then, it continues execution.

The IC acts as follows:
・Waits for busfree.
・Enters arbitration after detecting busfree.
・if it wins arbitration, it assets XSSEL, XSIO and ID bits, entering the re-selection phase.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
After that, the IC enters Target mode.

●Wait_Reselect(0EH)
Waits for the re-selection phase.
Valid only when it is not connected.
If issued in the connected condition, it sets SCSIINT2 and CMDER bits, causing interruption. If any other command is under execution then, it continues execution.

The IC acts as follows:
・Enters the condition waiting for re-selection.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
After that, the IC enters Initiator mode.

●Wait_SCAM_Selection_Command(0FH)
Waits for SCAM selection, and causes interruption after the selection is made.
Valid only when it is not connected.
If issued in the connected condition, it sets SCSIINT2 and CMDER bits, causing interruption. If any other command is under execution then, it continues execution.

If this command is issued, set STATN (bit5) of SCSIMODE register, and clear it when the command is over.
After the command was issued, the IC acts as follows:
・Waits for the SCAM/normal selection phase.
・Does not respond to, but ignores, SCSI selection with the selection time-out delay less than 4ms.
・Responds to SCAM selection, causing interruption.
・If no SCAM selection occurs, the IC responds to the selection which continues for 4ms or longer, and acts like Wait_Select_Command (0Ch) after that.
After that, the IC enters Target mode.

●Negate_ACK(11H)
Clears ACK left asserted by the last message transfer in Initiator mode when the LSI stopped operation.

●Command_Out(12H)
Executes SCSI command phase.
Valid only in the connected condition. It can be issued in either Target or Initiator mode.
If it is issued in the disconnected condition,
・It sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
The IC acts as follows:
Enters this command into FIFO after setting its command byte number in NON-DMA data-size register.
This control command does not distinguish command groups automatically. It is used when any group command which has undefined command block length is received or other cases.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.

CPU reads a command from FIFO

○ In Initiator mode
CPU issues this command after setting the command byte number in NON-DMA data-size register.
Then it writes the command data in FIFO.
The IC acts as follows:
・Negates XSACK if it is asserted at the start of execution.
・Transfers the command data in FIFO after checking the command phase at the timing of assertion of XSREQ.
Suspends REQ-ACK hand-shake until FIFO becomes full of data, if FIFO was empty.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
If any other phase is found when the command phase is checked:
・Sets ILPHS of SCSIINT1, causing interruption.
Caution: Be sure that data is written in FIFO after the number of bytes to be transferred was set.

●DMA_Data_Out(13H)
Executes the data-out phase to be transferred between port and SCSI.
(Setting FIFO bit enables transfer between CPU and SCSI.)
Valid only in the connected condition. It can be issued in either Target or Initiator mode.
If issued in the disconnected condition, it sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
Combination of this command issued and the AND condition of DTGO bit of DMACTL register starts DMA transfer.
When transfer of the count value set in DTBC register is over, the command ends, GOOD and DTCMP bits of MAININT register are set, and interruption is caused.

○ In Initiator mode
Negates XSACK if it is asserted at the start of execution.
After the data-out phase was checked at the timing of assertion of XSREQ, the AND condition of DTGO bit of DMACTL register starts actual DMA transfer. When the transfer of the count value set in DTBC register is over, the command ends, GOOD and DTCMP bits of MAININT register are set, and interruption is caused.

If any other phase is found when the data-out phase is checked:
・Sets ILPHS of SCSIINT1, causing interruption.

●Non-DMA_Data_Out(14H)
Executes the data-out phase between SCSI and CPU interface.
Only valid in the connected condition.
Both Target and Initiator mode can be published.
If it is issued in the disconnected condition,
・It sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
Setting data bytes to be transferred in NON-DMA data-size register issues this command.
CPU reads data from inside of FIFO by referring to FIFO status.
The IC acts as follows:
・Enters data bytes set into FIFO after setting the data-out phase.
REQ-ACK hand-shake is suspended until FIFO has some space, if FIFO was full.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.

○ In Initiator mode
Setting data bytes to be transferred in NON-DMA data-size register issues this command.
CPU writes data into FIFO referring to FIFO status.
The IC acts as follows:
・Negates XSACK if it is asserted at the start of execution.

・Transfers data bytes set from FIFO after checking the data-out phase at the timing when XSREQ is asserted. Suspends REQ-ACK hand-shake until FIFO becomes full of data, if FIFO was empty
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
Caution: Be sure that data is written in FIFO after the number of bytes to be transferred was set.

●DMA_Data_In(15H)
Executes the data-in phase between SCSI and CPU interface.
(Setting FIFO bit enables transfer between CPU and SCSI.)
Valid only in the connected condition. It can be issued in either Target or Initiator mode.
If issued in the disconnected condition, it sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
Combination of this command issued and the AND condition of DTGO bit of DMACTL register starts DMA transfer.
When transfer of the count value set in DTBC register is over, the command ends, GOOD and DTCMP bits of MAININT register are set, and interruption is caused.

○ In Initiator mode
Negates XSACK if it is asserted at the start of execution.
After the data-out phase was checked at the timing of assertion of XSREQ, the AND condition of DTGO bit of DMACTL register starts actual DMA transfer. When the transfer of the count value set in DTBC register is over, the command ends, GOOD and DTCMP bits of MAININT register are set, and interruption is caused.

If any other phase is found when the data-in phase is checked:
・Sets ILPHS of SCSIINT1, causing interruption.

●Non-DMA_Data_In(16H)
Executes the data-in phase between SCSI and CPU interface.
Valid only in the connected condition. It can be issued in either Target or Initiator mode.
If it is issued in the disconnected condition,
・It sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
Setting data bytes to be transferred in NON-DMA data-size register issues this command.
CPU writes data into FIFO by referring to FIFO status.
The IC acts as follows:
・Outputs data bytes set from FIFO after setting the data-in phase.
Suspends REQ-ACK hand-shake until FIFO becomes full of data, if FIFO was empty.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
Caution: Be sure that data is written in FIFO after the number of bytes to be transferred was set.

○ In Initiator mode
Setting data bytes to be transferred in NON-DMA data-size register issues this command.
CPU reads data from inside of FIFO by referring to FIFO status.
The IC acts as follows:
・Negates XSACK if it is asserted at the start of execution.
・Enters data into FIFO after checking the data-in phase at the timing of assertion of XSREQ.
Suspends REQ-ACK hand-shake until FIFO has some space, if FIFO was full.

If any other phase is found when the data-in phase is checked:
・Sets ILPHS of SCSIINT1, causing interruption.

●Status_In(17H)
Executes the status phase.
Valid only in the connected condition. It can be issued in either Target or Initiator mode.

If issued in the disconnected condition, it sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
CPU issues this command after writing the status byte in SCSI FIFO.
The IC transfers the status in FIFO after setting the status phase.
After that,
・It sets GOOD bit of MAININT register.
・It causes interruption.

○ In Initiator mode
Negates XSACK if it is asserted at the start of execution.
Enters 1 byte of status into SCSI FIFO after checking the status phase at the first timing when XSREQ is asserted.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
CPU checks the interrupt status, and reads status byte from SCSI FIFO if termination is normal.

●Message_In (18H)
Executes the message-in phase.
Valid only in the connected condition. It can be issued in either Target or Initiator mode.
If issued in the disconnected condition, it sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
CPU sets the message bytes to be sent in NON-DMA data-size register, and issues this command. It writes messages to be transferred in FIFO.
The IC sets the message phase, and sends the data bytes set in FIFO.
It suspends REQ-ACK hand-shake until FIFO becomes full of data, if FIFO was empty.
After that,
・It sets GOOD bit of MAININT register.
・It causes interruption.
Caution: Be sure that data is written in FIFO after the number of bytes to be transferred was set.

○ In Initiator mode
CPU sets the byte size of message to be received in NON-DMA data-size register before issuing this command.
The IC acts as follows:
・Negates XSACK if it is asserted at the start of execution.
・Enters the byte size of message set into FIFO after checking the message-in phase at the timing when REQ is asserted. REQ-ACK hand-shake is suspended until FIFO has some space, if FIFO was full.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
Usually, the size of message is unknown beforehand. So set “1” as the byte to be transferred when the command is issued first, and then decide the number of bytes to be received after the second byte by checking the message code received.

●Message_Out(19H)
Executes the message-out phase.
Valid only in the connected condition. It can be issued in either Target or Initiator mode.
If issued in the disconnected condition, it sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
CPU sets the byte size of message to be received in NON-DMA data-size register before issuing this command.
After setting the message-out phase, the IC enters the byte size of message set into FIFO.
If FIFO becomes full, REQ-ACK hand-shake is suspended until CPU read out message from FIFO to make some space in it.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
Usually, the size of message is unknown beforehand. So set “1” as the byte to be transferred when the

command is issued first, and then decide the number of bytes to be received after the second byte by checking the message code received.

○ In Initiator mode
CPU sets the byte size of message to be sent in NON-DMA data-size register before issuing this command.
CPU writes the message to be transferred in FIFO.
The IC acts as follows:
・Asserts XSATN.
・Negates XSACK if it is asserted at the start of execution.
・Sends data in FIFO after checking the message phase at the timing when XSREQ is asserted.
Suspends REQ-ACK hand-shake until FIFO becomes full of data, if FIFO was empty.
・Negates XSATN after sending the bytes to be transferred.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
Caution: Be sure that data is written in FIFO after the number of bytes to be transferred was set.

●Status_Message(1AH)
Executes the message-in phase after executing the status phase.
Valid only in the connected condition. It can be issued in either Target or Initiator mode.
If issued in the disconnected condition, it sets SCSIINT2 and CMDER bits, causing interruption.

○ In Target mode
Writing the status message to be sent in FIFO issues this command.
It is allowed also to write the status message to be transferred in FIFO after issuing the command.
The IC acts as follows:
・Sets the status phase, and transfer one status byte taken out of FIFO.
・Sets the message-in phase, and transfer one message byte taken out of FIFO.
Suspends REQ-ACK hand-shake until FIFO becomes full of data, if FIFO was empty.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.

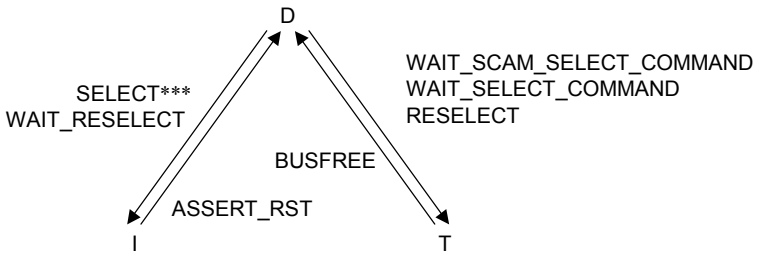
○ In Initiator mode
Issuing this command makes each one byte of status and message taken in FIFO.
CPU reads one status byte, then one message byte, from FIFO.
The IC acts as follows:
・Negates XSACK if it is asserted at the start of execution.
・Enters status into FIFO after checking the status phase at the timing of assertion of XSREQ.
・After receiving status, it enters message into FIFO after checking the message-in phase at the timing of assertion of XSREQ.
・After that, it sets GOOD bit of MAININT register.
・It causes interruption.
Note: The message length is fixed at one byte.

7.4.3 Command Execution and State Transition

The IC has the following three state transitions from the viewpoint of execution of SCSI-type commands:

- Disconnected condition (D)
- Connected condition in Initiator mode (I)
- Connected condition in Target mode (T)

The state transition among those conditions are caused by specific commands, as shown in the figure below:



The transition from I to D is done implicitly except by Assert_RST command.
It means that there is no command which causes such transition explicitly. The disconnected condition is assumed if busfree is found when a next connection-type command is issued, and the command is executed. If busfree is not found, the disconnected condition is waited for, and the connection-related command is suspended.

The connection-type command can be executed in the disconnected condition. It may be issued in such condition.
If it is issued while SCSI control command is under execution, though, a command error will occur.
A transfer-type command can be executed in the connected condition. If it is issued in the disconnected condition, a command error will occur.
Both connection- and transfer-type commands can be executed when the IC is in the condition where no SCSI control command is under execution.
If they are issued while SCSI control command is under execution, a command error will occur.

7.5 Others and Cautions about Operation

●Operation responding to the selection without SCSI-1 arbitration phase

The IC operates responding to the selection of only a target ID of SCSI-1 as mentioned below. Note that there occurs no (automatic) transition to the message or command phase after selection, as in the usual cases after Wait_selection command.

① If only an ID is selected after Wait_select_cmd command was issued, IDERR interrupt occurs and the command ends.
The inside is in the condition where connection is complete, though. So message_out/command_out and other commands can be issued, as in the usual case the selection is made with an initiator/target ID. (Except that message_out/command_out is not executed automatically.)

② If an ID of 3 bits or more is selected, IDERR interrupt occurs. This distinguishes whether selection of 1 bit is completed or IDERR with an ID of 3 bits or more is selected.
If 1 bit is selected, IDERR and SEL interrupts occur. If ATN is not asserted here, WOATN interrupt occurs, too.

* The firm is asked to check that SCSI-1 selection has occurred by observing SEL interrupt at the same timing when IDERR interrupt occurs.
Also, issue message_out/command_out manually while observing the condition of WOATN interrupt, because IDERR terminates Wait_selection command.

● Parity error in SCSI data phase, or command stop operation by detection of ATN

Take note of the following points when port interface is used as slave:
If a setting has been made that a parity error or detection of ATN in SCSI data phase stops the operation of a command (STATN/STPPE/SPCEN bit of SCSIMODE register), occurrence of such factor and subsequent command stop cause negation of HDMARQ being output to IDE interface at the internal timing of the IC.
Accordingly, use such setting after checking that it causes no problem in hand-shake on the LSI side connected to the IC.
In such a case, no problem occurs in the internal sequence of the IC if XHDMACK or XHIOR/XHIOW may come from the port side. Though, data transfer to and from FIFO may be obstructed depending on timing.
(The data may not be written in or read from the FIFO.)
In such a case, the FIFO terminates in uncompleted manner, so it requires clearing before going to the status phase.

8. ELECTRICAL CHARACTERISTICS

8.1 Absolute Maximum Ratings

V_{SS} = 0[V]

Item	Symbol	Ratings	Unit
Supply voltage	HV_{DD}	-0.3 to +6.0	V
	LV_{DD}	-0.3 to +4.6	V
Input voltage	HV_{IN}	-0.3 to HV_{DD} + 0.5	V
	LV_{IN}	-0.3 to LV_{DD} + 0.5	V
Output voltage	HV_{OUT}	-0.3 to HV_{DD} + 0.5	V
	LV_{OUT}	-0.3 to LV_{DD} + 0.5	V
Output current/pins (SCSI output pins)	I_{OUT1}	50	mA
Output current/pins (Other than SCSI output pins)	I_{OUT2}	±30	mA
Storage temperature	T_{stg}	-65 to +150	°C

8.2 Recommended operational conditions

V_{SS} = 0[V]

Item	Symbol	Min.	Typ.	Max.	Unit
Supply voltage	HV_{DD}	4.50	5.00	5.50	V
	LV_{DD}	3.00	3.30	3.60	
Input voltage	HV_{IN}	V_{SS}	—	HV_{DD}	V
	LV_{IN}	V_{SS}	—	LV_{DD}	V
Operating temperature	Topr	0	25	70	°C
Input signal rise time Normal input	t_{ri}	—	—	50	ns
Input signal fall time Normal input	t_{fi}	—	—	50	ns
Input signal rise time Schmitt input	t_{ri}	—	—	5	ms
Input signal fall time Schmitt input	t_{fi}	—	—	5	ms

8.3 DC Characteristics

(1) I/O characteristics in the DC condition(Ta = 0 to 70°C, V_{SS} = 0V)

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Static current	HI_{DDS}	HV_{DD}=5.0V±10%	—	—	80	μA
	LI_{DDS}	LV_{DD}=3.3V±0.3V	—	—	220	
Input leak current	I_{LI}	HV_{DD}=5.0V	-1	—	1	μA
Input pins capacitance	C_I	f=1MHz HV_{DD}=0V	—	—	10	pF
Output pins capacitance	C_O	f=1MHz HV_{DD}=0V	—	—	10	pF
Input/output pins capacitance	C_{IO}	f=1MHz HV_{DD}=0V	—	—	10	pF

(2) TTL input characteristics (Ta = 0 to 70°C, V_{SS} = 0V)
Names of signals covered: AD0 to 5, DB0 to 7, HDD0 to 15, EXCLK, XSATN, XSBSY, XSRST, XSMSG, XSSEL, XSCD, XSIO

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
HIGH level input voltage	V_{IH}	HV_{DD}=5.5V	2.0	—	—	V
LOW level input voltage	V_{IL2H}	HV_{DD}=4.5V	—	—	0.8	V

(3) CMOS input characteristics (Ta = 0 to 70°C, V_{SS} = 0V)
Names of signals covered: CLKSEL1 to 0, XPLLPD, OSCIN, PLLCNT

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
HIGH level input voltage	V_{IHIL}	LV_{DD}=3.6V	1.9	—	—	V
LOW level input voltage	V_{ILIL}	LV_{DD}=3.0V	—	—	0.8	V

(4) TTL Schmitt input characteristics (Ta = 0 to 70°C, V_{SS} = 0V)
Names of signals covered: XSREQ, XSACK, XSDB0 to 7, XSDBP, XCS, XRD, XWR, TESTEN, XRESET, HDMARQ, XHIOW, XHIOR HIORDY, XHDMACK, HINTRQ, XHPDIAG, XHDASP

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
HIGH level trigger input voltage	V_{T2+}	HV_{DD}=5.0V	1.2	—	2.4	V
LOW level trigger input voltage	V_{T2-}	HV_{DD}=5.0V	0.6	—	1.8	V
Hysteresis voltage	ΔV_H	HV_{DD}=5.0V	0.1	—	—	V

(5) Pull-up/down input characteristics (Ta = 0 to 70°C, V_{SS} = 0V)
Names of signals covered: AD0 to 5, DB0 to 7, XCS, XRD, XWR, XRESET, TESTEN

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Pull-up resistance	R_{pu}	V_I=0V HV_{DD}=5.0V	50	100	200	kΩ
Pull-down resistance	R_{pd}	V_I= HV_{DD} HV_{DD}=5.0V	50	100	200	kΩ

(6) Output characteristics (Ta = 0 to 70°C, V_{SS} = 0V) (I_{OL} = 2mA)
Names of signals covered: TESTMON

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
HIGH level output voltage	V_{OH}	LV_{DD}=3.0V I_{OH}=-2mA	LV_{DD} -0.4	—	—	V
LOW level output voltage	V_{OL}	LV_{DD}=Min. I_{OL}=-2mA	—	—	0.4	V

(7) Output characteristics (Ta = 0 to 70°C, V_{SS} = 0V) (I_{OL} = 3mA)
Names of signals covered: HDD0 to 15, DB0 to 7, XHIOW, XHIOR, XHDMACK

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
HIGH level output voltage	V_{OH}	HV_{DD}=5.0V I_{OH}=-1.5mA	HV_{DD} -0.4	—	—	V
LOW level output voltage	V_{OL}	HV_{DD}=4.5V I_{OL}=3mA	—	—	0.4	V

(8) Output characteristics (Ta = 0 to 70°C, V_{SS} = 0V) (I_{OL} = 6mA)
Names of signals covered: XINT, XHRST, HDMARQ, HDA0 to 2, XHCS0 to 1

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
HIGH level output voltage	V_{OH}	HV_{DD}=5.0V I_{OH}=-3mA	HV_{DD} -0.4	—	—	V
LOW level output voltage	V_{OL}	HV_{DD}=4.5V I_{OL}=6mA	—	—	0.4	V

(9) Output characteristics (Ta = 0 to 70°C, V_{SS} = 0V) (Open drain I_{OL} = 6mA)
Names of signals covered: XWAIT

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
3-state leak current	I_{OZ}	HV_{DD}=Max.	-1	—	1	μA
LOW level output voltage	V_{OL5}	HV_{DD}=Min. I_{OL}=6mA	—	—	0.4	V

(10) Output characteristics (Ta = 0 to 70°C, V_{SS} = 0V) (Open drain I_{OL} = 48mA)
Names of signals covered: XSATN, XSBSY, XSRST, XSMSG, XSSEL, XSCD, XSIO

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
3-state leak current	I_{OZ}	HV_{DD}=Max.	-1	—	1	μA
LOW level output voltage	V_{OL5}	HV_{DD}=Min. I_{OL}=48mA	—	—	0.4	V

(11) Output characteristics (Ta = 0 to 70°C, V_{SS} = 0V) (I_{OL} = 48mA)
Names of signals covered: XSDB0 to 7, XSDBP, XSREQ, XSACK

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
LOW level output voltage	V_{OL5}	HV_{DD}=Min. I_{OL}=48mA	—	—	0.4	V

8.4 AC Characteristics

Measurement conditions of AC characteristics:

・Ta = 0 to 70°C HV_{DD} = 5V±10% LV_{DD} = 3.3V±0.3V
V_{SS} = 0V

・ DC level to decide input
0.8V to 2.4V

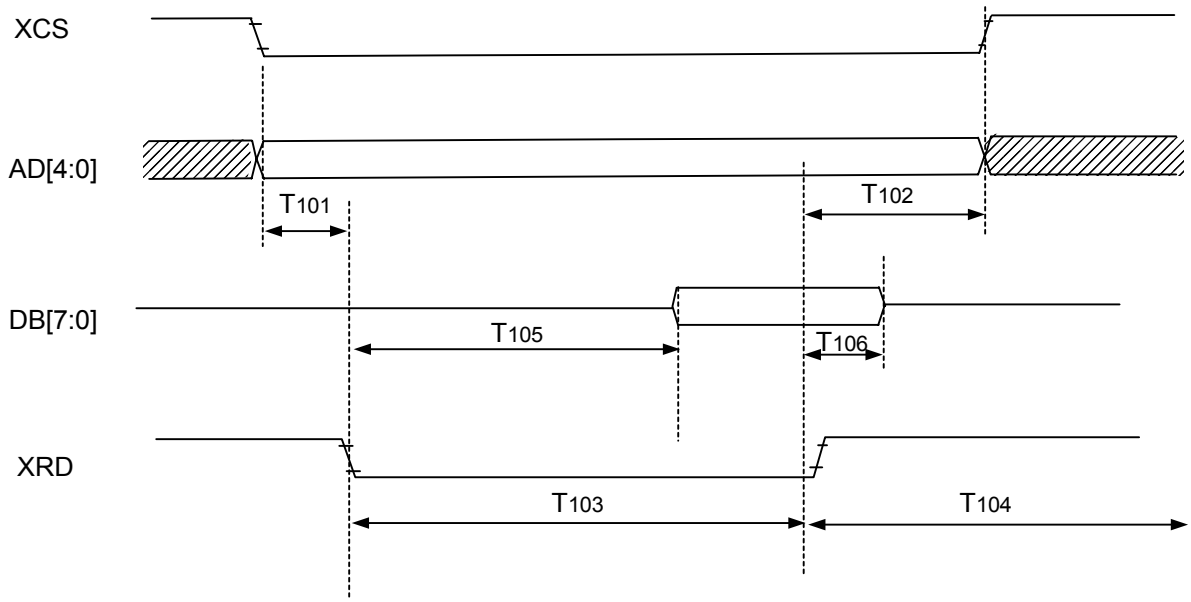
・ Operating clock
f_{oscin} = 40MHz

・ Loading conditions of output pins except SCSI pins
Drives load capacitance of 50pF and 1TTL.

・ Load capacitance of SCSI pins
Load capacitance = 100pF, pull-up resistance = 110Ω/pull-down resistance = 165Ω

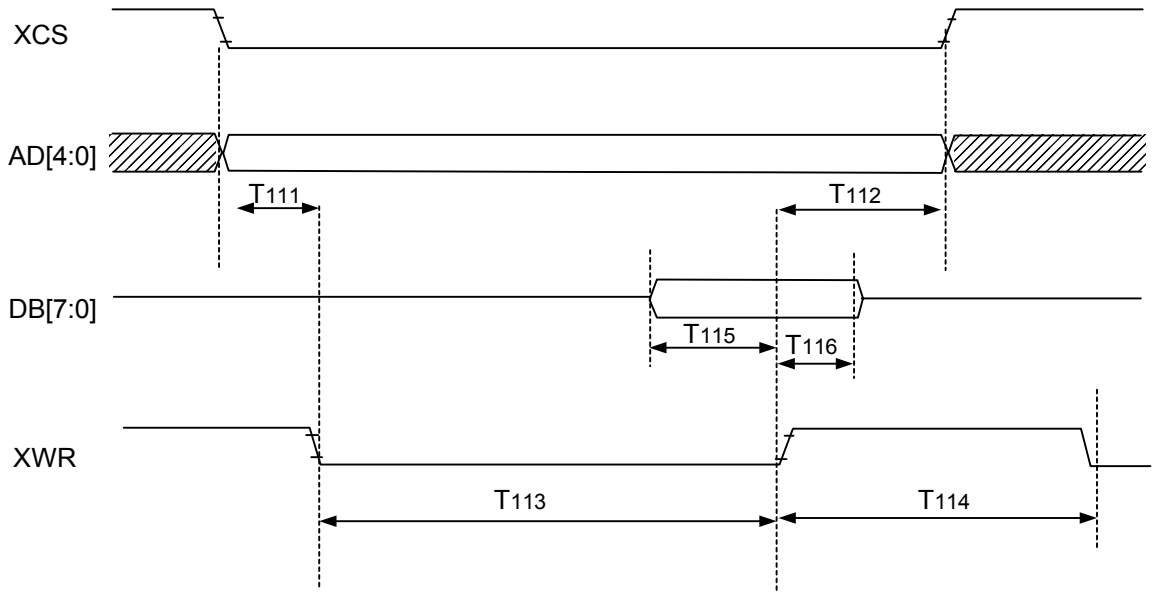
8.4.1 CPU Interface

8.4.1.1 Register Read Timing



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{101}	XCS fall → XRD fall AD [4:0] valid → XRD fall	3	—	—	ns
T_{102}	XRD rise → AD [4:0] invalid XRD rise → XCS rise	0	—	—	ns
T_{103}	XRD LOW level pulse width	60	—	—	ns
T_{104}	XRD HIGH level pulse width	45	—	—	ns
T_{105}	XRD fall → DB [7:0] output	—	—	60	ns
T_{106}	XRD rise → DB [7:0] hold	2	—	—	ns

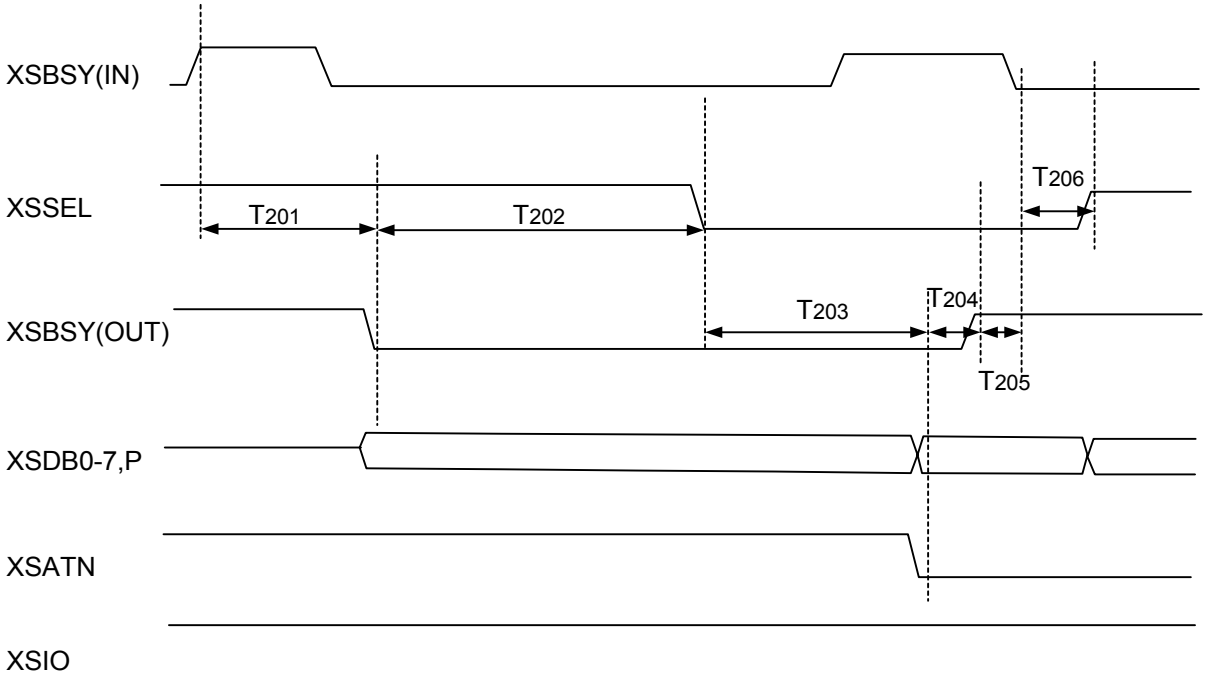
8.4.1.2 Register Write Timing



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{111}	XCS fall → XWR fall AD [4:0] valid → XWR fall	3	—	—	ns
T_{112}	XWR rise → AD [4:0] invalid XWR rise → XCS rise	0	—	—	ns
T_{113}	XWR LOW level pulse width	40	—	—	ns
T_{114}	XWR HIGH level pulse width	45	—	—	ns
T_{115}	DB [7:0] valid → XWR rise	10	—	—	ns
T_{116}	XWR rise → DB [7:0] hold	0	—	—	ns

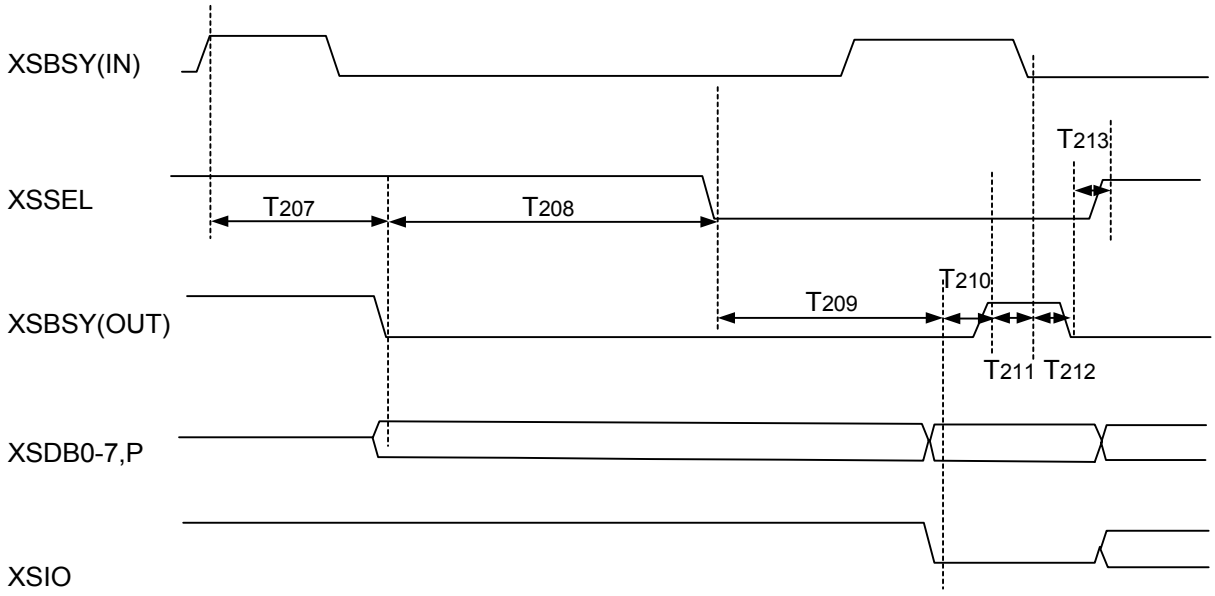
8.4.2 SCSI Interface

8.4.2.1 Selection Timing



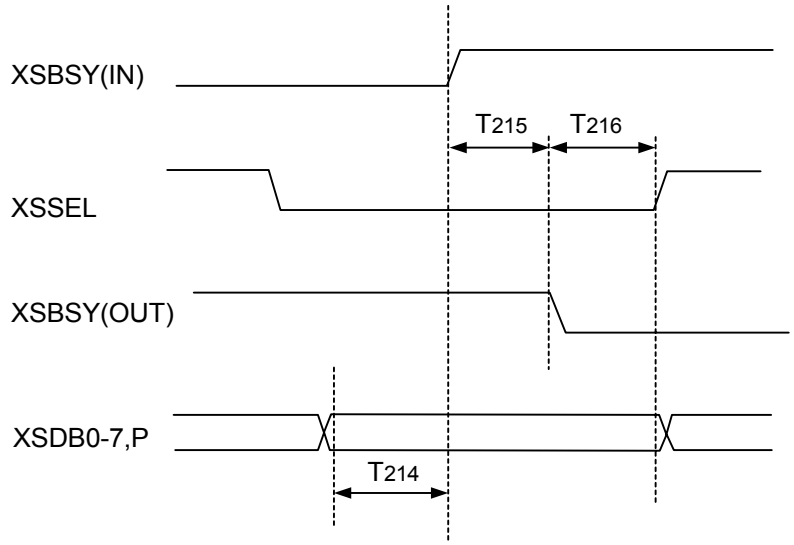
Symbol	Specification	Min.	Typ.	Max.	Unit
T_{201}	XSBSY(IN) ↑ -XSBSY(OUT) ↓, OWNID valid	1600	—	—	ns
T_{202}	XSBSY(OUT) ↓ -XSSEL ↓	2900	—	—	ns
T_{203}	XSSEL ↓ -SELID valid	1500	—	—	ns
T_{204}	SELID valid -XSBSY(OUT) ↑	90	—	—	ns
T_{205}	XSBSY(OUT) ↑ -XSBSY(IN) ↓	500	—	—	ns
T_{206}	XSBSY(IN) ↓ -XSSEL ↑	250	—	—	ns

8.4.2.2 Re-selection Timing



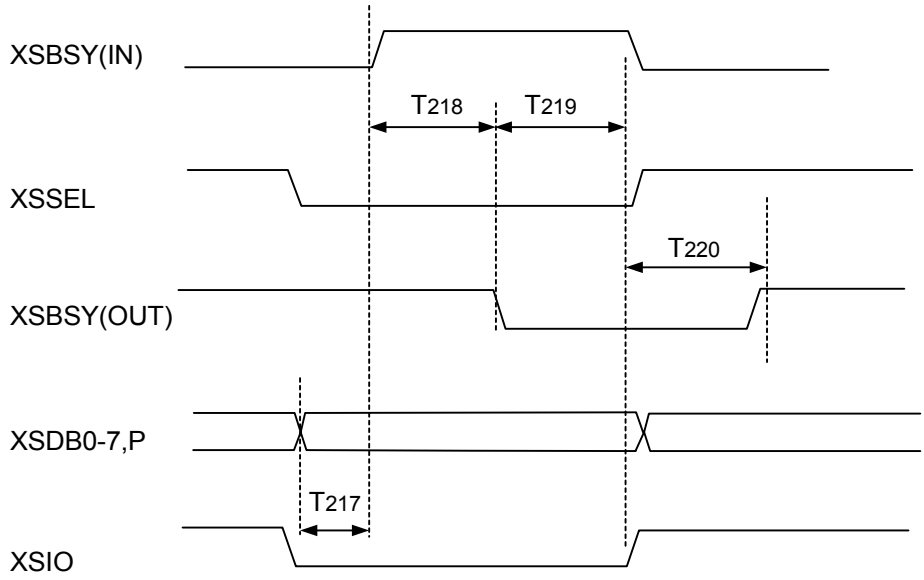
Symbol	Specification	Min.	Typ.	Max.	Unit
T_{207}	XSBSY(IN) ↑ -XSBSY(OUT) ↓, OWNID valid	1600	—	—	ns
T_{208}	XSBSY(OUT) ↓ -XSSEL ↓	2900	—	—	ns
T_{209}	XSSEL ↓ -SELID valid, XSIO ↓	1500	—	—	ns
T_{210}	SELID valid -XSBSY(OUT) ↑	90	—	—	ns
T_{211}	XSBSY(OUT) ↑ -XSBSY(IN) ↓	500	—	—	ns
T_{212}	XSBSY(IN) ↓ -XSBSY(OUT) ↓	100	—	—	ns
T_{213}	XSBSY(OUT) ↓ -XSSEL ↑	150	—	—	ns

8.4.2.3 Timing of Being Selected



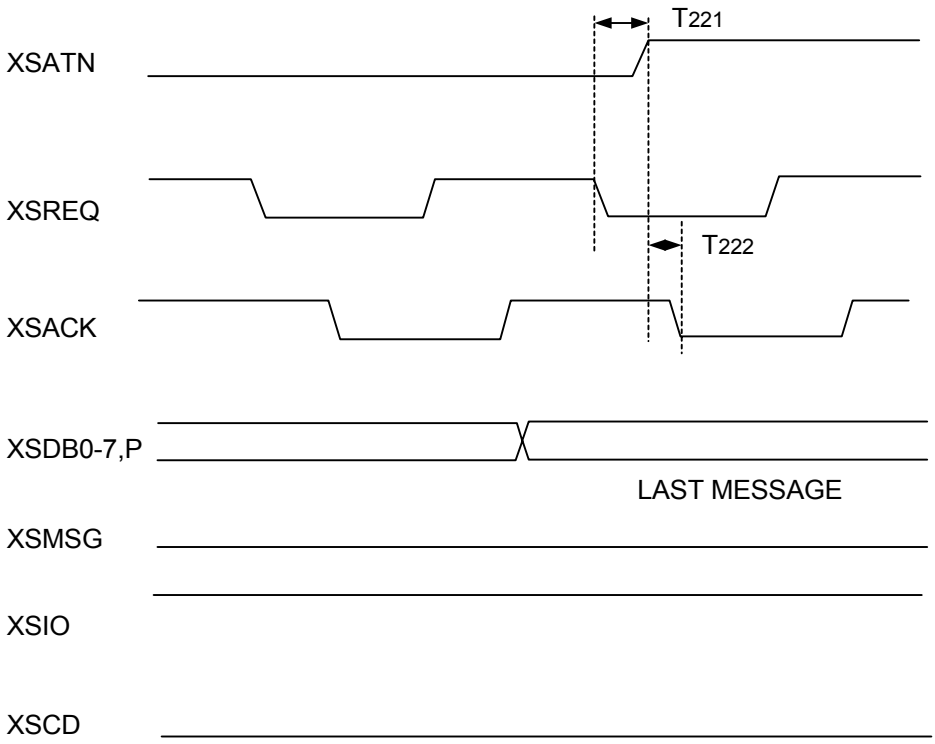
Symbol	Specification	Min.	Typ.	Max.	Unit
T_{214}	SELID valid -XSBSY(IN) ↑	0	—	—	ns
T_{215}	XSBSY(IN) ↑ -XSBSY(OUT) ↓	800	—	—	ns
T_{216}	XSBSY(OUT) ↓ -XSSEL ↑	0	—	—	ns

8.4.2.4 Timing of Being Re-Selected



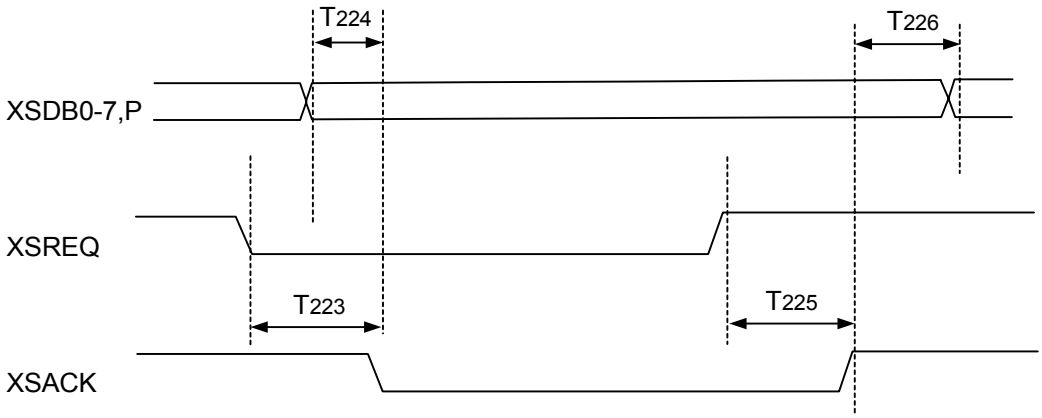
Symbol	Specification	Min.	Typ.	Max.	Unit
T_{217}	SELID valid -XSBSY(IN) ↑	0	—	—	ns
T_{218}	XSBSY(IN) ↑ -XSBSY(OUT) ↓	800	—	—	ns
T_{219}	XSBSY(OUT) ↓ -XSSEL ↑	0	—	—	ns
T_{220}	XSSEL ↑ -XSBSY(OUT) ↑	—	—	200	ns

8.4.2.5 XSATN Output Timing



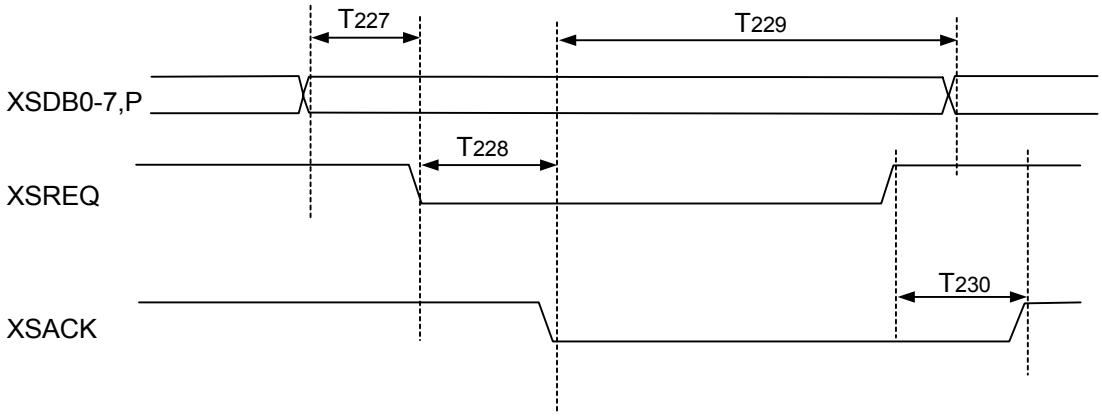
Symbol	Specification	Min.	Typ.	Max.	Unit
T_{221}	XSREQ ↓ -XSATN ↑	25	—	—	ns
T_{222}	XSATN ↑ -XSACK ↓	150	—	—	ns

8.4.2.6 Initiator Asynchronous Data-out Timing (Data output)



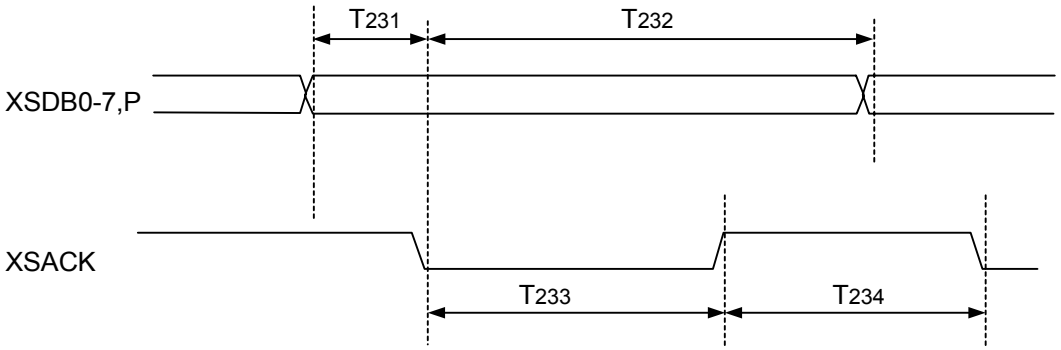
Symbol	Specification	Min.	Typ.	Max.	Unit
T_{223}	XSREQ ↓ -XSACK ↓	25	—	—	ns
T_{224}	XSDB valid -XSACK ↓	90	—	—	ns
T_{225}	XSREQ ↑ -XSACK ↑	25	—	90	ns
T_{226}	XSACK ↑ -XSDB invalid	50	—	—	ns

8.4.2.7 Initiator Asynchronous Data-in Timing (Data input)



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{227}	XSDB valid -XSREQ ↓	30	—	—	ns
T_{228}	XSREQ ↓ -XSACK ↓	25	—	—	ns
T_{229}	XSACK ↓ -XSDB invalid	0	—	—	ns
T_{230}	XSREQ ↑ -XSACK ↑	25	—	90	ns

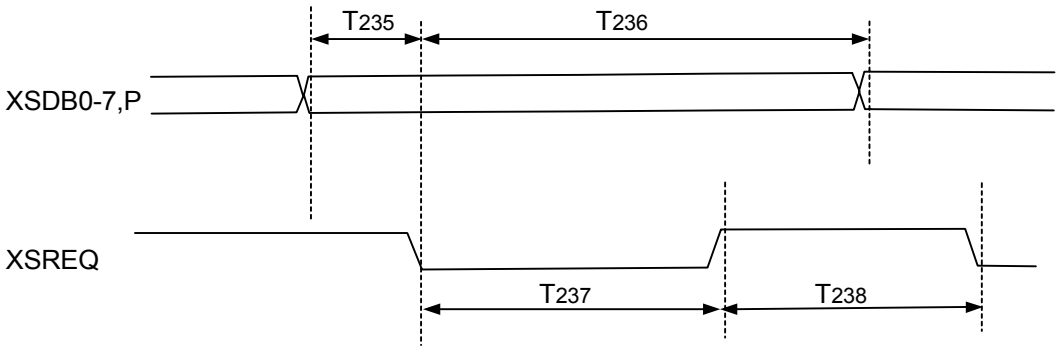
8.4.2.8 Initiator Synchronous Data-out Timing (Data output)



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{231}	XSDB valid –XSACK ↓	45 (18)	—	—	ns
T_{232}	XSACK ↓ -XSDB invalid	45 (18)	—	—	ns
T_{233}	XSACK ↓ -XSACK ↑	45 (20)	—	—	ns
T_{234}	XSACK ↑ - (NEXT) XSACK ↓	45 (20)	—	—	ns

Note: Value of when RATE3 to 0bit is “0000”
The timing of switching data is the same as in the case of XSACK rise.
The value at Ultra set-up is described in ( ).

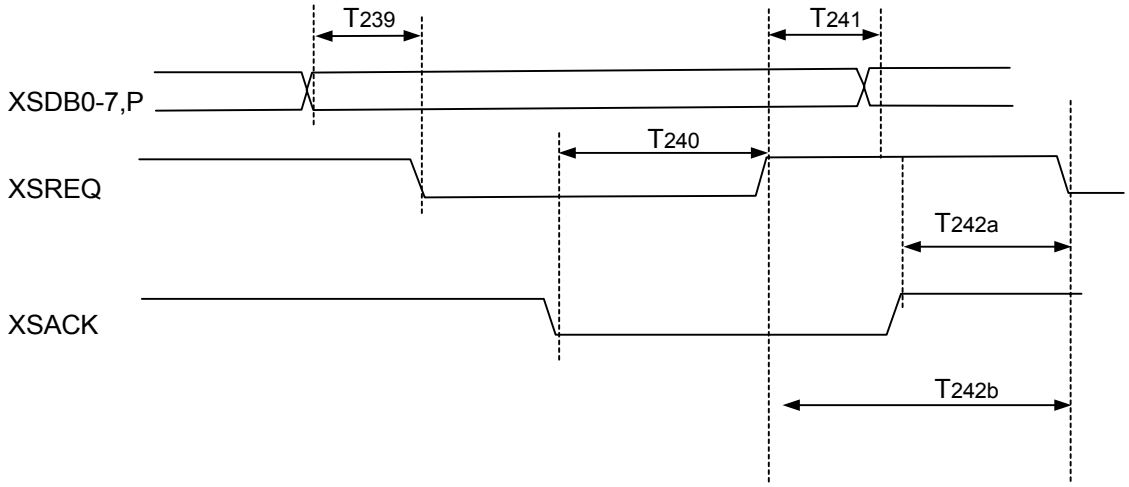
8.4.2.9 Initiator Synchronous Data-in Timing (Data input)



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{235}	XSDB0-7, P valid -XSREQ ↓	15 (6)	—	—	ns
T_{236}	XSREQ ↓ -XSDB0-7, P invalid	5 (11)	—	—	ns
T_{237}	XSREQ ↓ -XSREQ ↑	22 (11)	—	—	ns
T_{238}	XSREQ ↑ -XSREQ ↓	22 (11)	—	—	ns

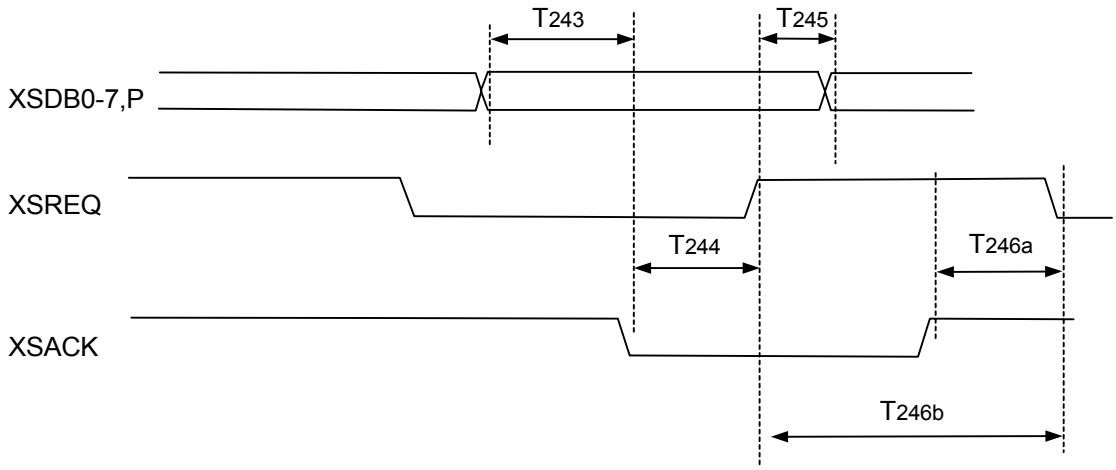
Note: Minimum cycle of T_{237}+T_{238} is 100ns (50ns).
The value at Ultra set-up is described in ( ).

8.4.2.10 Target Asynchronous Data-in Timing (Data output)



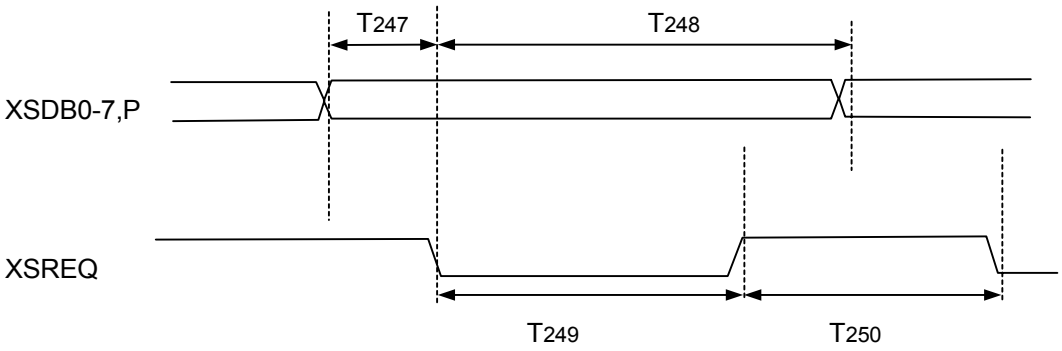
Symbol	Specification	Min.	Typ.	Max.	Unit
T_{239}	XSDB valid -XSREQ ↓	90	—	—	ns
T_{240}	XSACK ↑ -XSREQ ↑	25	—	90	ns
T_{241}	XSREQ ↑ -XSDB invalid	50	—	—	ns
T_{242a}	XSACK ↑ - (NEXT) XSREQ ↓	25	—	—	ns
T_{242b}	XSREQ ↑ -XSREQ ↓	150	—	—	ns

8.4.2.11 Target Asynchronous Data-out Timing (Data input)



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{243}	XSDB valid -XSACK ↓	30	—	—	ns
T_{244}	XSACK ↓ -XSREQ ↑	25	—	90	ns
T_{245}	XSREQ ↑ -XSDB invalid	0	—	—	ns
T_{246a}	XSACK ↑ -XSREQ ↓	25	—	—	ns
T_{246b}	XSREQ ↑ -XSREQ ↓	150	—	—	ns

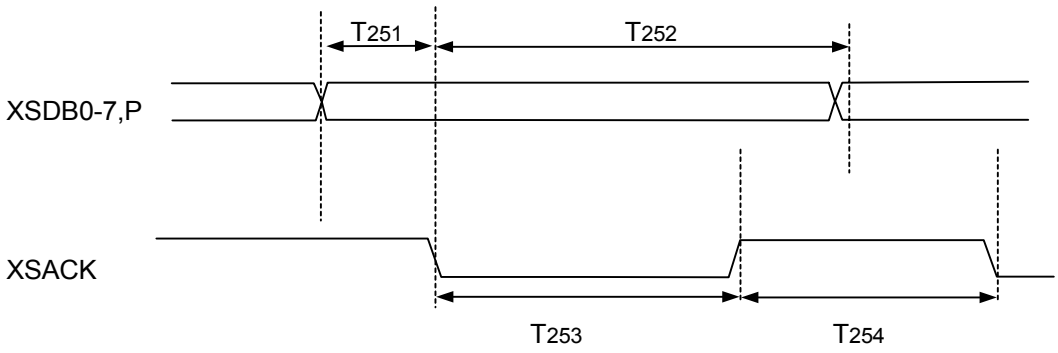
8.4.2.12 Target Synchronous Data-in Timing (Data output)



Symbol	Specification	Min.	Typ.	Max.	Unit
T247	XSDB valid -XSREQ ↓	45 (18)	—	—	ns
T248	XSREQ ↓ -XSDB invalid	45 (18)	—	—	ns
T249	XSREQ ↓ -XSREQ ↑	45 (20)	—	—	ns
T250	XSREQ ↑ -XSREQ ↓	45 (20)	—	—	ns

Note: Value of when RATE3 to 0bit is “0000”
The timing of switching data is the same as in the case of XSREQ rise.
The value at Ultra set-up is described in ( ).

8.4.2.13 Target Synchronous Data-out Timing (Data input)

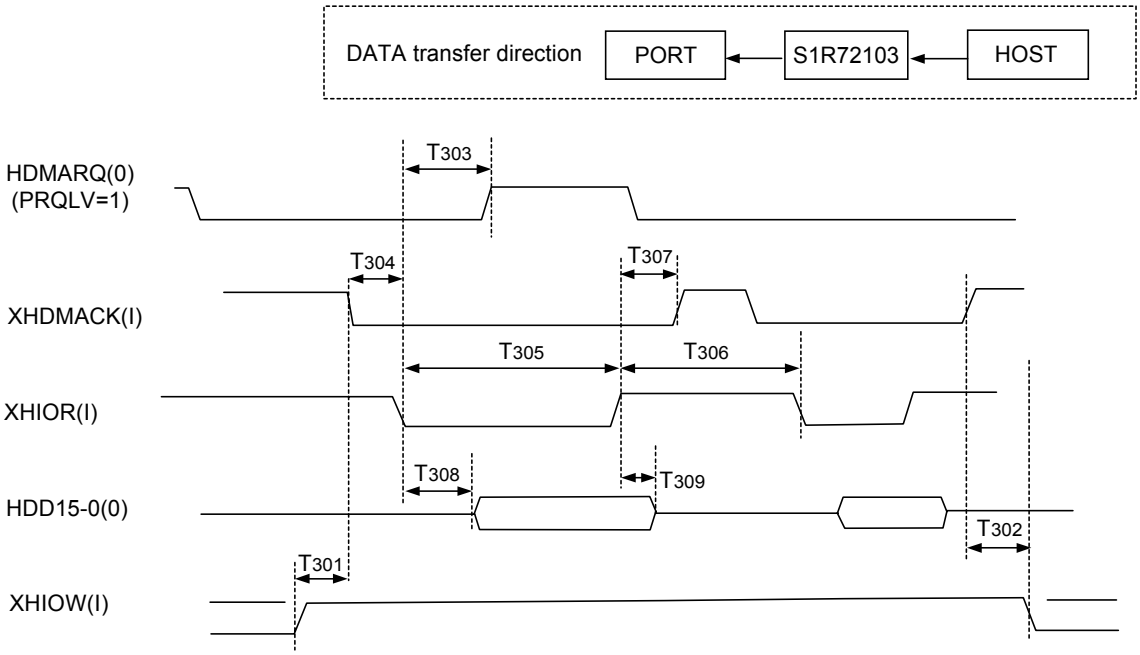


Symbol	Specification	Min.	Typ.	Max.	Unit
T_{251}	XSDB valid -XSACK ↓	15 (6)	—	—	ns
T_{252}	XSACK ↓ -XSDB invalid	5 (11)	—	—	ns
T_{253}	XSACK ↓ -XSACK ↑	22 (11)	—	—	ns
T_{254}	XSACK ↑ -XSACK ↓	22 (11)	—	—	ns

Note: The value at Ultra set-up is described in ( ).

8.4.3 Port Interface

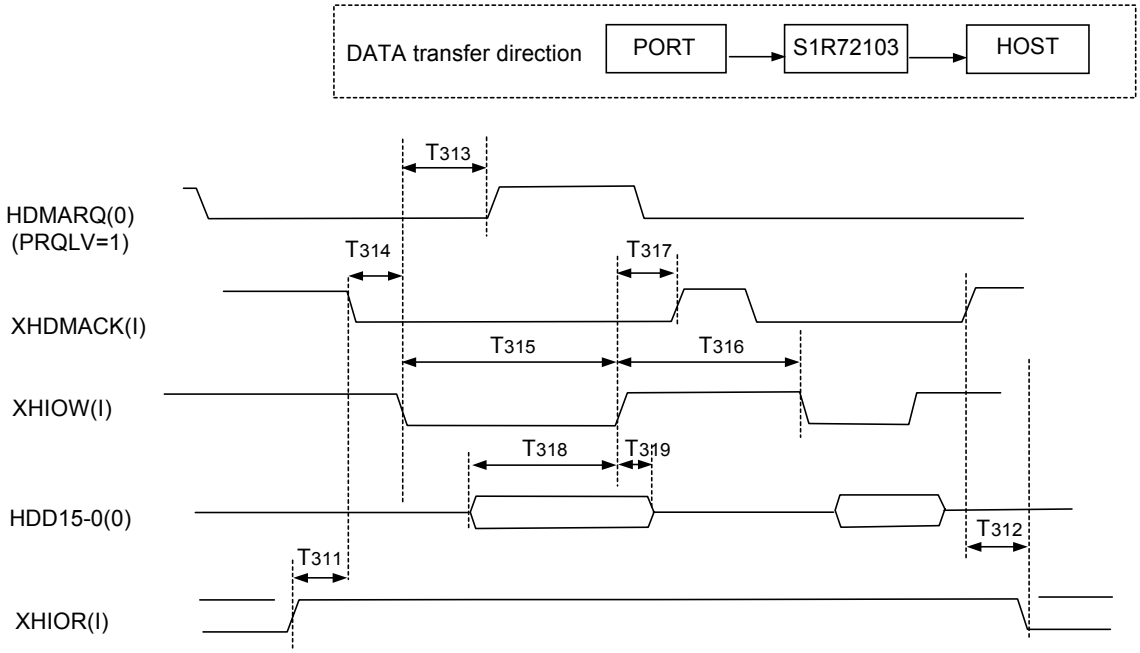
8.4.3.1 DMA read (PSLV=1: Slave mode)



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{301}	XHIOW → XHDMACK ↓ XHDMACK setup time	5	—	—	ns
T_{302}	XHDMACK ↑ → XHIOW XHDMACK hold time	5	—	—	ns
T_{303}	XHIOR ↓ → HDMARQ negate HDMARQ negate delay time	10	—	40	ns
T_{304}	XHDMACK ↓ → XHIOR ↓ XHIOR setup time	0	—	—	ns
T_{305}	XHIOR ↓ → XHIOR ↑ XHIOR assert pulth width	30	—	—	ns
T_{306}	XHIOR ↑ → XHIOR ↓ XHIOR negate pulth width	30	—	—	ns
T_{307}	XHIOR ↑ → XHDMACK ↑ XHIOR hold time	0	—	—	ns
T_{308}	XHIOR ↓ → HDD Data output delay time Note 1	0	—	25	ns
T_{309}	XHIOR ↑ → HDD(Hi-Z) Data bus negate time Note 1	6	—	40	ns

Note 1: Data is output to HDD only while both XHDMACK and XHIOR are asserted.
HDD is always in Input mode except such time.

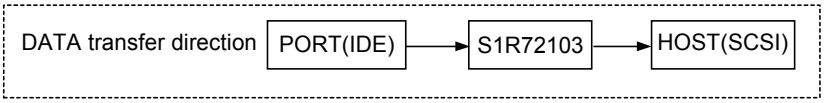
8.4.3.2 DMA write (PSLV=1: Slave mode)

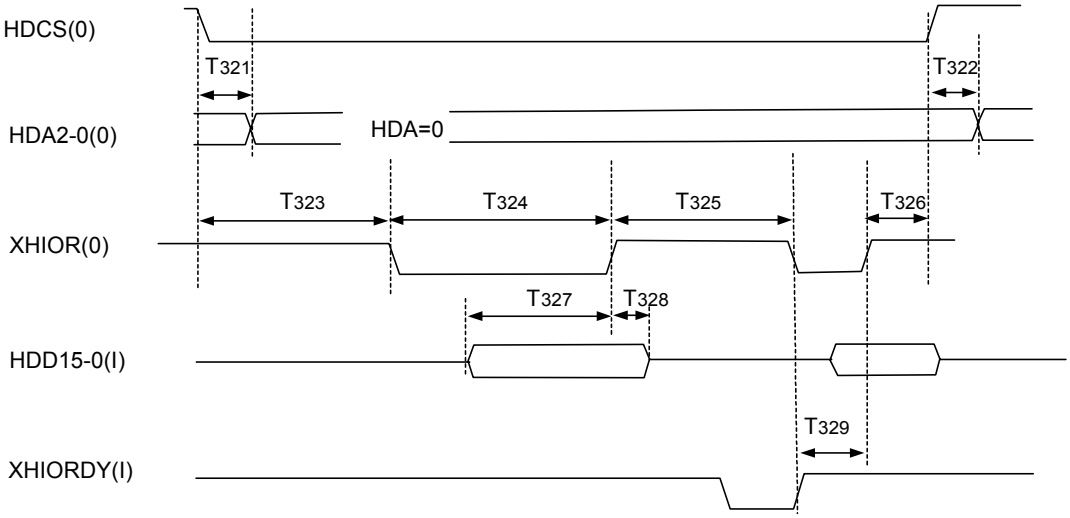


Symbol	Specification	Min.	Typ.	Max.	Unit
T311	XHIOR → XHDMACK ↓ XHDMACK setup time	5	—	—	ns
T312	XHDMACK ↑ → XHIOR XHDMACK hold time	5	—	—	ns
T313	XHIOW ↓ → HDMARQ negate HDMARQ negate delay time	10	—	40	ns
T314	XHDMACK ↓ → XHIOW ↓ XHIOW setup time	0	—	—	ns
T315	XHIOW ↓ → XHIOW ↑ XHIOW assert pulth width	30	—	—	ns
T316	XHIOW ↑ → XHIOW ↓ XHIOW negate pulth width	30	—	—	ns
T317	XHIOW ↑ → XHDMACK ↑ XHIOW hold time	0	—	—	ns
T318	HDD → XHIOR ↑ Data setup time Note 1	10	—	—	ns
T319	XHIOR ↑ → HDD(Hi-Z) Data bus negate time Note 1	0	—	—	ns

8.4.3.3 PIO read (PSLV=0: Master mode)

[DMA=0,ULTRA=0]

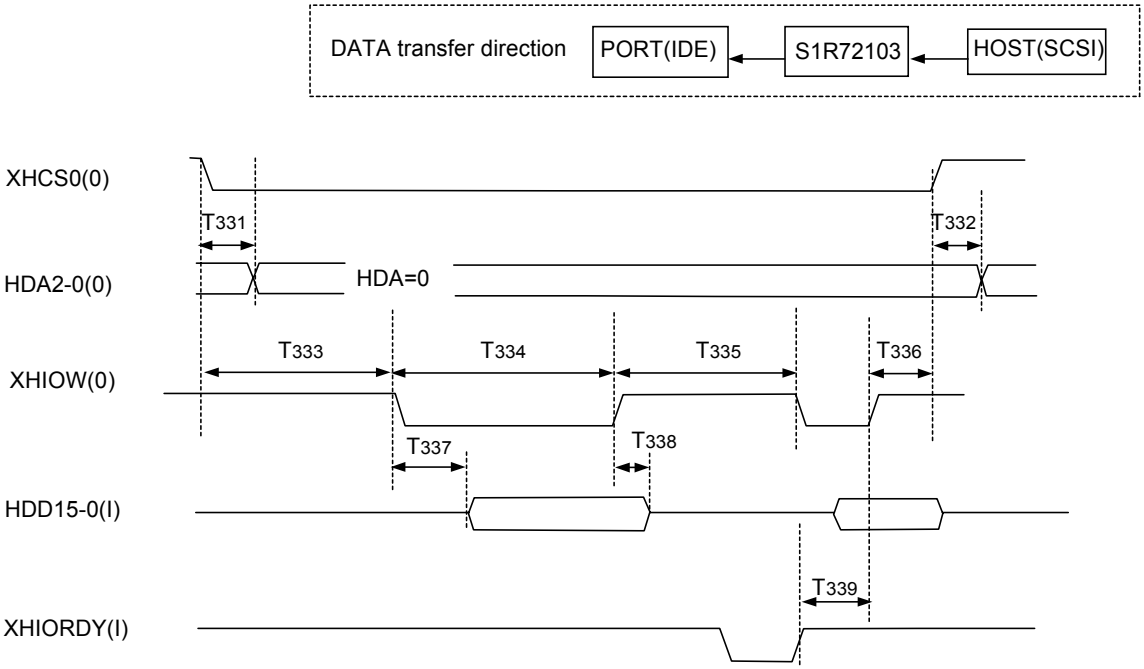




Symbol	Specification	Min.	Typ.	Max.	Unit
T_{321}	XHCS0 ↓ → HAD HAD output delay time	—	0	—	ns
T_{322}	XHCS0 ↑ → HAD HAD hold time	—	0	—	ns
T_{323}	XHCS0 ↓ → XHIOR ↓ XHIOR setup time	75	—	—	ns
T_{324}	XHIOR ↓ → XHIOR ↑ XHIOR assert pulth width	—	IDETMD (AP+2)×25	—	ns
T_{325}	XHIOR ↑ → XHIOR ↓ XHIOR negate pulth width	—	IDETMD (NP+2)×25	—	ns
T_{326}	XHIOR ↑ → XHCS0 ↑ XHIOR hold time	25	—	—	ns
T_{327}	HDD → XHIOR ↑ Data setup time	10	—	—	ns
T_{328}	XHIOR ↑ → HDD Data hold time	0	—	—	ns
T_{329}	HIORDY assert → XHIOR ↑ XHDMACK setup time	—	—	40	ns

8.4.3.4 PIO write (PSLV=0: Master mode)

[DMA=0,ULTRA=0]

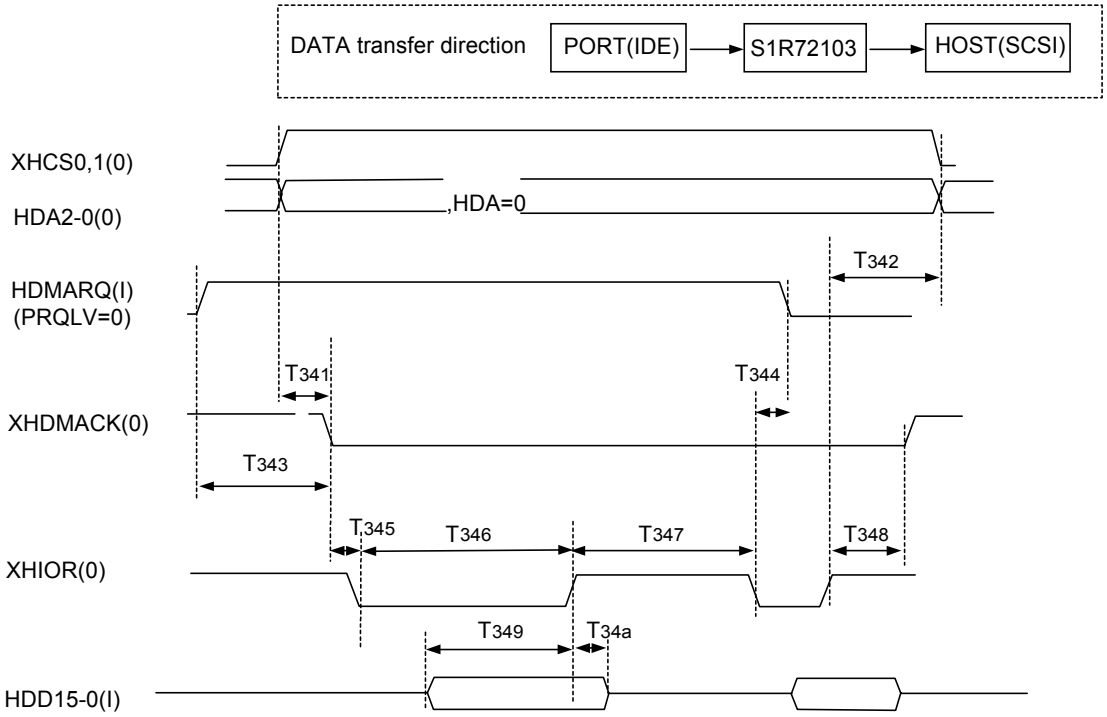


Symbol	Specification	Min.	Typ.	Max.	Unit
T_{331}	XHCS0 ↓ → HAD HAD output delay time	—	0	—	ns
T_{332}	XHCS0 ↑ → HAD HAD hold time	—	0	—	ns
T_{333}	XHCS0 ↓ → XHIOW ↓ XHIOW setup time	75	—	—	ns
T_{334}	XHIOW ↓ → XHIOW ↑ XHIOW assert pulth width Note 1	—	IDETMD (AP+2)×25	—	ns
T_{335}	XHIOW ↑ → XHIOW ↓ XHIOW negate pulth width Note 1	—	IDETMD (NP+2)×25	—	ns
T_{336}	XHIOW ↑ → XHCS0 ↑ XHIOW hold time	25	—	—	ns
T_{337}	XHIOW ↓ → HDD Data output delay time	0	—	25	ns
T_{338}	XHIOW ↑ → HDD Data bus negate time	40	—	60	ns
T_{339}	HIORDY assert → XHIOW ↑ XHDMACK setup time	—	—	40	ns

Note 1: Take setting value of -5ns into consideration for Min. value of pulth width.

8.4.3.5 DMA read (PSLV=0: Master mode)

[DMA=1,ULTRA=0]

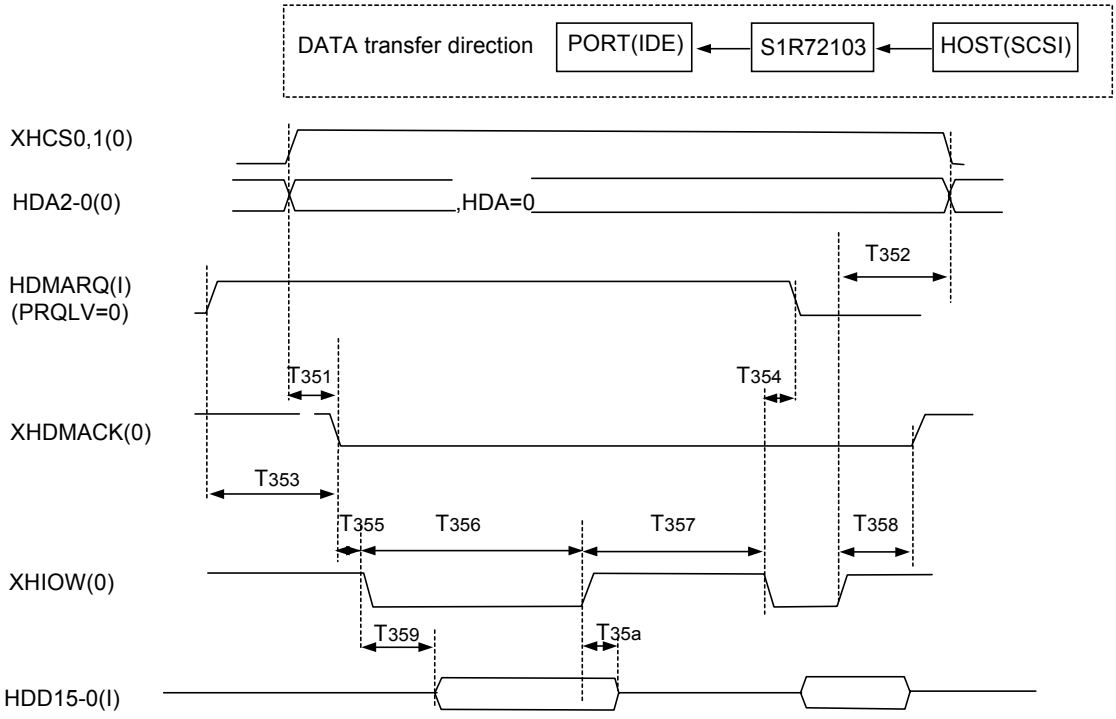


Symbol	Specification	Min.	Typ.	Max.	Unit
T_{341}	XHCS0,1 ↑ → XHDMACK ↓ Address setup time	75	—	—	ns
T_{342}	XHIOR ↑→ XHCS0 ↓ Address hold time	25	—	—	ns
T_{343}	HDMARQ ↑ → XHDMACK ↓ XHDMACK response time	0	—	—	ns
T_{344}	XHIOR ↓ → HDMAQR negate HDMARQ hold time	0	—	—	ns
T_{345}	XHDMACK ↓ → XHIOR ↓ XHIOR setup time	0	—	—	ns
T_{346}	XHIOR ↓ → XHIOR ↑ XHIOR assert pulth width Note 1	—	IDETMD (AP+2)×25	—	ns
T_{347}	XHIOR ↑ → XHIOR ↓ XHIOR negate pulth width	—	IDETMD (NP+2)×25	—	ns
T_{348}	XHIOR ↑ → XHDMACK ↑ XHIOR hold time	20	—	—	ns
T_{349}	HDD →XHIOR ↑ Data setup time	10	—	—	ns
T_{34a}	XHIOR ↑ → HDD Data bus negate time	0	—	—	ns

Note 1: No affected on HIORDY status.

8.4.3.6 DMA write (PSLV=0: Master mode)

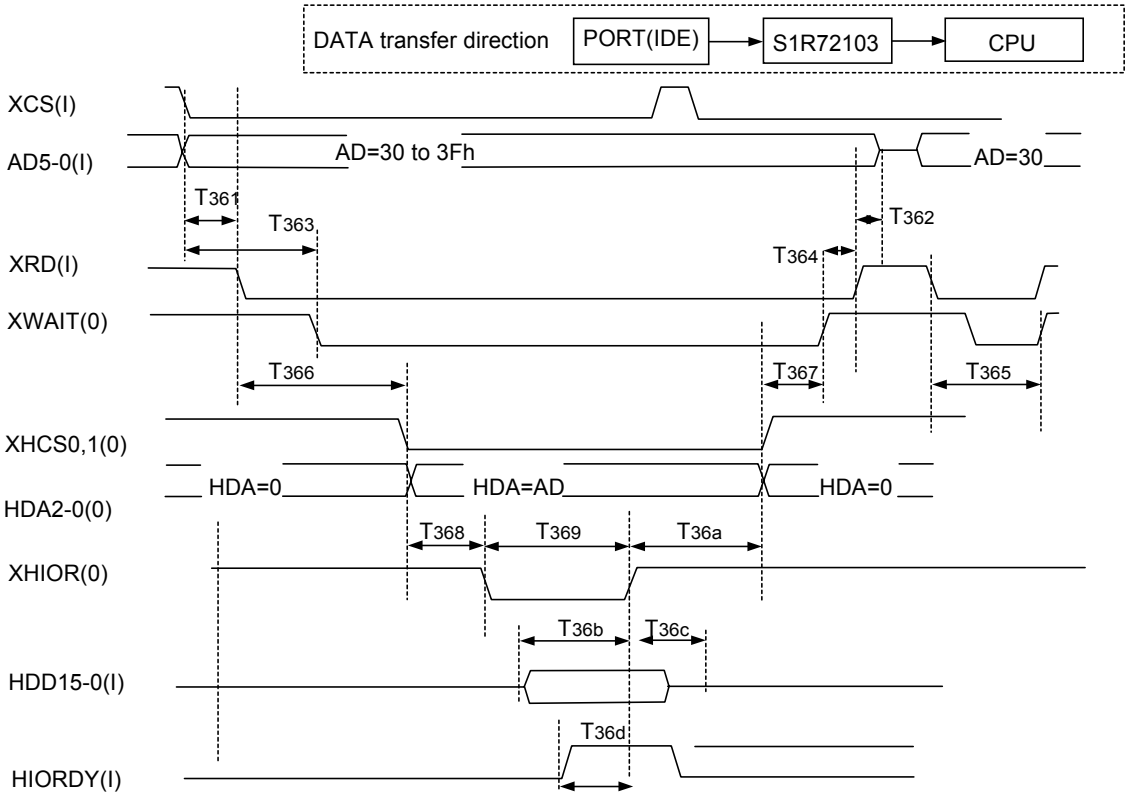
[DMA=1,ULTRA=0]



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{351}	XHCS0 ↑ → XHDMACK ↓ Address setup time	75	—	—	ns
T_{352}	XHIOW ↑ → XHCS0 ↓ Address hold time	25	—	—	ns
T_{353}	HDMARQ ↑ → XHDMACK ↓ XHDMACK response time	0	—	—	ns
T_{354}	XHIOW ↓ → HDMAQR negate HDMARQ hold time	0	—	—	ns
T_{355}	XHDMACK ↓ → XHIOW ↓ XHIOW setup time	0	—	—	ns
T_{356}	XHIOW ↓ → XHIOW ↑ XHIOW assert pulth width Note 1	—	IDETMD (AP+2)×25	—	ns
T_{357}	XHIOW ↑ → XHIOW ↓ XHIOW negate pulth width	—	IDETMD (NP+2)×25	—	ns
T_{358}	XHIOW ↑ → XHDMACK ↑ XHIOW hold time	20	—	—	ns
T_{359}	XHIOW ↓ → HDD Data output delay time	0	—	25	ns
T_{35a}	XHIOW ↑ → HDD Data bus negate time	20	—	40	ns

Note 1: No affected on HIORDY status.

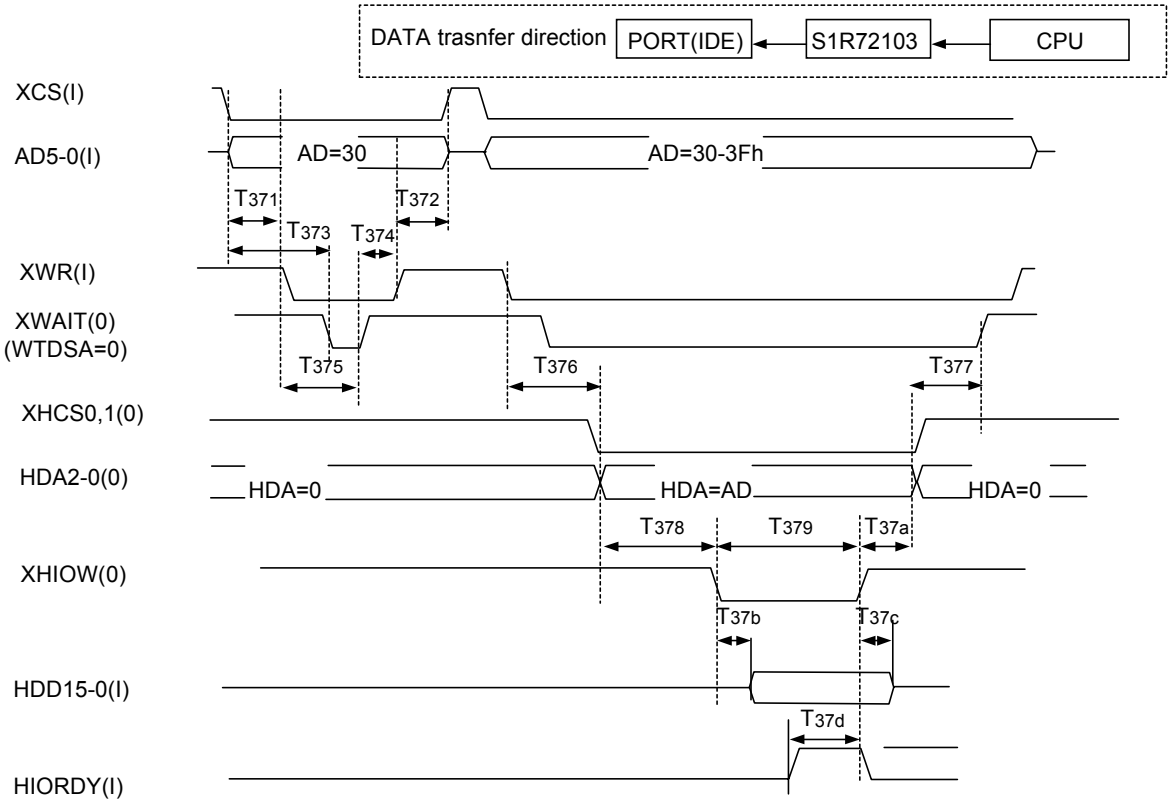
8.4.3.7 CPU PORT access PORT read (PSLV=0: Master mode)



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{361}	XCS,AD → XRD ↓ Address setup time	5	—	—	ns
T_{362}	XRD ↑ → XCS, AD Address hold time	5	—	—	ns
T_{363}	XRD ↓ → XWAIT ↓ XWAIT output delay time	0	—	15	ns
T_{364}	XWAIT ↑ → XRD ↑ XRD hold time	0	—	—	ns
T_{365}	XRD ↓ → XWAIT ↑ XWAIT response time Note 1	120	—	140	ns
T_{366}	XRD ↓ →XHCS0,1 Address output delay time	80	—	—	ns
T_{367}	XHCS0,1 ↑ → XWAIT ↑ XWAIT output delay time	5	—	25	ns
T_{368}	XHCS0,1 ↓ → XHIOR ↓ Address setup time	95	—	—	ns
T_{369}	XHIOR ↓ → XHIOR ↑ XHIOR assert pulth width	—	IDERMD (AP+2)×25	—	ns
T_{36a}	XHIOR ↑ → XHCS0,1 ↑ Address hold time	—	IDERMD (NP+2)×25	—	ns
T_{36b}	HDD → XHIOR ↑ Data setup time	10	—	—	ns
T_{36c}	XHIOR ↑ → HDD Data hold time	0	—	—	ns
T_{36d}	HIORDY assert → XHIOR ↑ HIORDY set-up time	—	—	40	ns

Note1: This cycle does not exist when BUS8=1 or AD ≠ 30h. Only this cycle exists regardless of AD when PSLV=1, and no access to PORT Interface.

8.4.3.8 CPU PORT access PORT write (PSLV=0: Master mode)

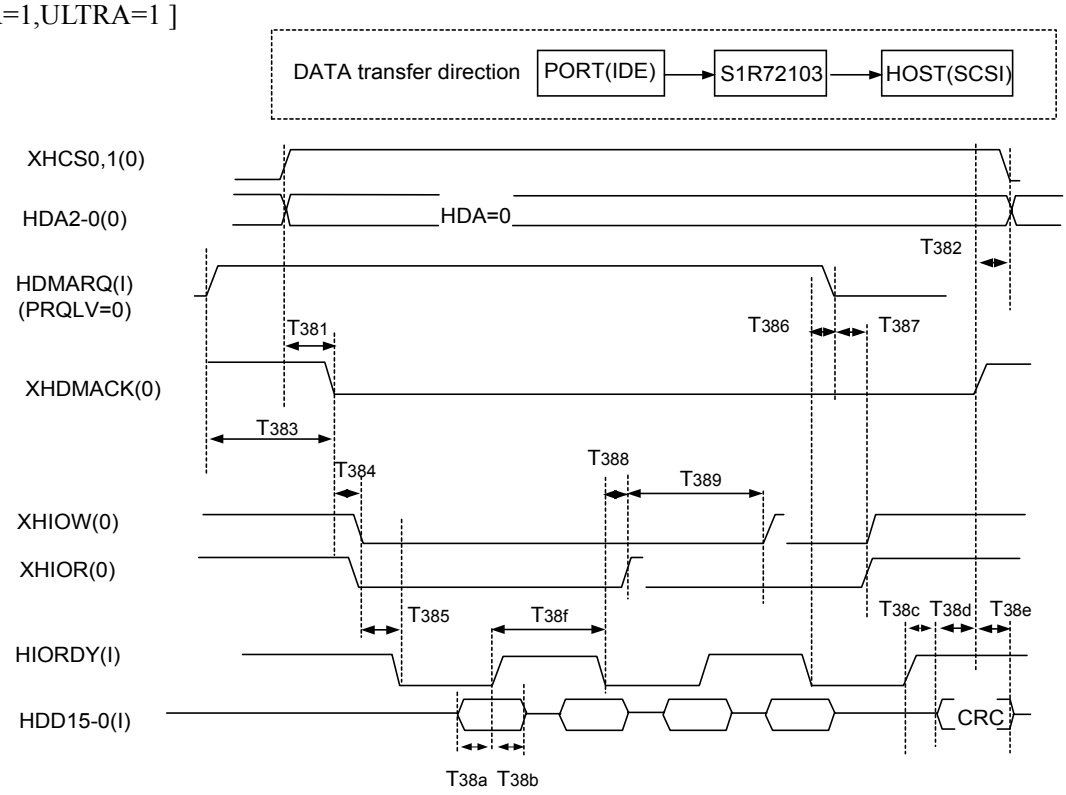


Symbol	Specification	Min.	Typ.	Max.	Unit
T_{371}	XCS,AD → XWR ↓ Address setup time	5	—	—	ns
T_{372}	XWR ↑ → XCS,AD Address hold time	5	—	—	ns
T_{373}	XWR ↓ → XWAIT ↓ XWAIT output delay time	0	—	15	ns
T_{374}	XWAIT ↑ → XWR ↑ XWR hold time	0	—	—	ns
T_{375}	XWR ↓ → XWAIT ↑ XWAIT response time Note 1	120	—	140	ns
T_{376}	XWR ↓ → XHCS0,1 Address output delay time	80	—	—	ns
T_{377}	XHCS0,1 ↑ → XWAIT ↑ XWAIT output delay time	5	—	25	ns
T_{378}	XHCS0,1 ↓ → XHIOW ↓ Address setup time	95	—	—	ns
T_{379}	XHIOW ↓ → XHIOW ↑ XHIOW assert pulth width	—	IDERMD (AP+2)×25	—	ns
T_{37a}	XHIOW ↑ → XHCS0,1 ↑ Address hold time	—	IDERMD (NP+2)×25	—	ns
T_{37b}	XHIOW ↓ → HDD Data output delay time	0	—	25	ns
T_{37c}	XHIOW ↑ → HDD Data bus negate time	40	—	60	ns
T_{37d}	HIORDY assert → XHIOW ↑ HIORDY setup time	—	—	40	ns

Note1: This cycle does not exist when BUS8=1 or AD ≠ 30h. Only this cycle exists regardless of AD when PSLV=1, and no access to PORT Interface.

8.4.3.9 ULTRA-DMA read (PSLV=0: Master mode)

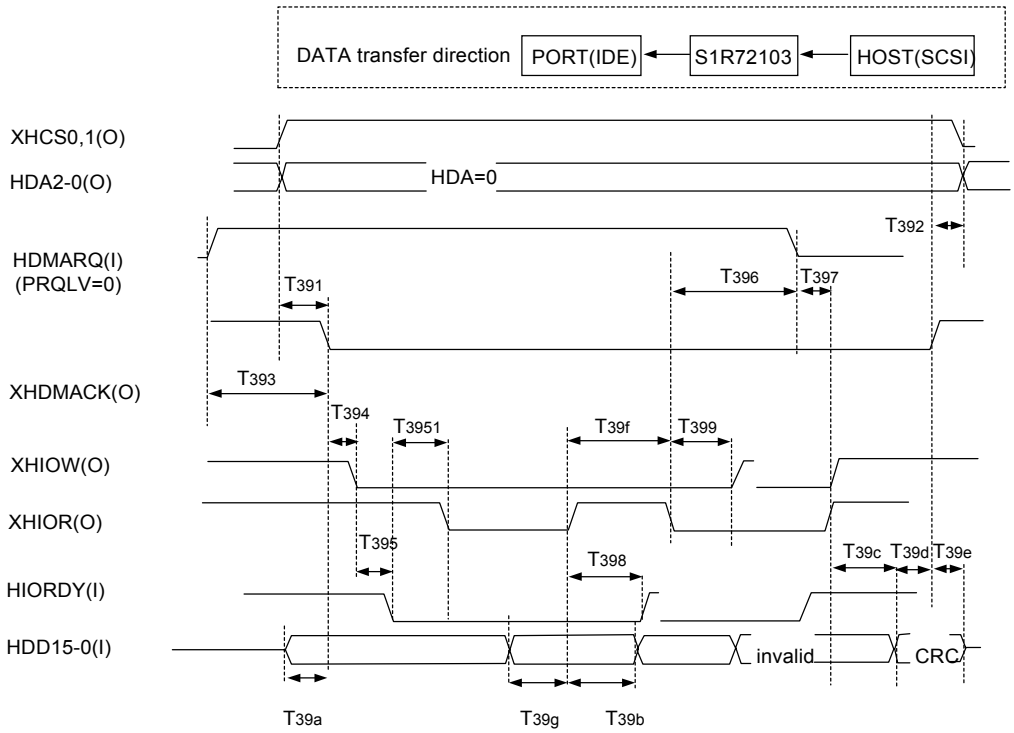
[DMA=1,ULTRA=1]



Symbol	Specification	Min.	Typ.	Max.	Unit
T_{381}	XHCS0,1 ↑ → XHDMACK ↓ Address setup time	20	—	—	ns
T_{382}	XHDMACK ↑ → XHCS0,1 ↓ Address hold time	40	—	—	ns
T_{383}	HDMARQ ↑ → XHDMACK ↓ XHDMACK response time	20	—	—	ns
T_{384}	XHDMACK ↓ → XHIOR, XHIOW ↓ envelop time	20	—	—	ns
T_{385}	XHIOR, XHIOW ↓ → HIORDY ↓ First Strobe Time	0	—	—	ns
T_{386}	HIORDY ↓ → HDMARQ ↓ Strobe edge to negation DMARQ	30	—	—	ns
T_{387}	HDMARQ ↓ → XHIOR,XHIOW ↑ Limited interlock time	40	—	90	ns
T_{388}	HIORDY ↓ → XHIOR ↑ Strobe to DMARDY time	25	—	80	ns
T_{389}	XHIOR ↑ → XHIOW ↑ Ready-to-Pause time	170	—	—	ns
T_{38a}	HDD → HIORDY ↑ Data setup time	7	—	—	ns
T_{38b}	HIORDY ↑ → HDD Data hold time	5	—	—	ns
T_{38c}	HIORDY↑ → HDD CRC data output delay time	0	—	—	ns
T_{38d}	HDD → HDMACK ↑ CRC data setup time	75	—	95	ns
T_{38e}	HDMACK ↑ → HDD Data hold time	6	—	25	ns
T_{38f}	HIORDY ↑ → HIORDY ↓ HIORDY pulth width	55	—	—	ns

8.4.3.10 ULTRA-DMA write (PSLV=0: Master mode)

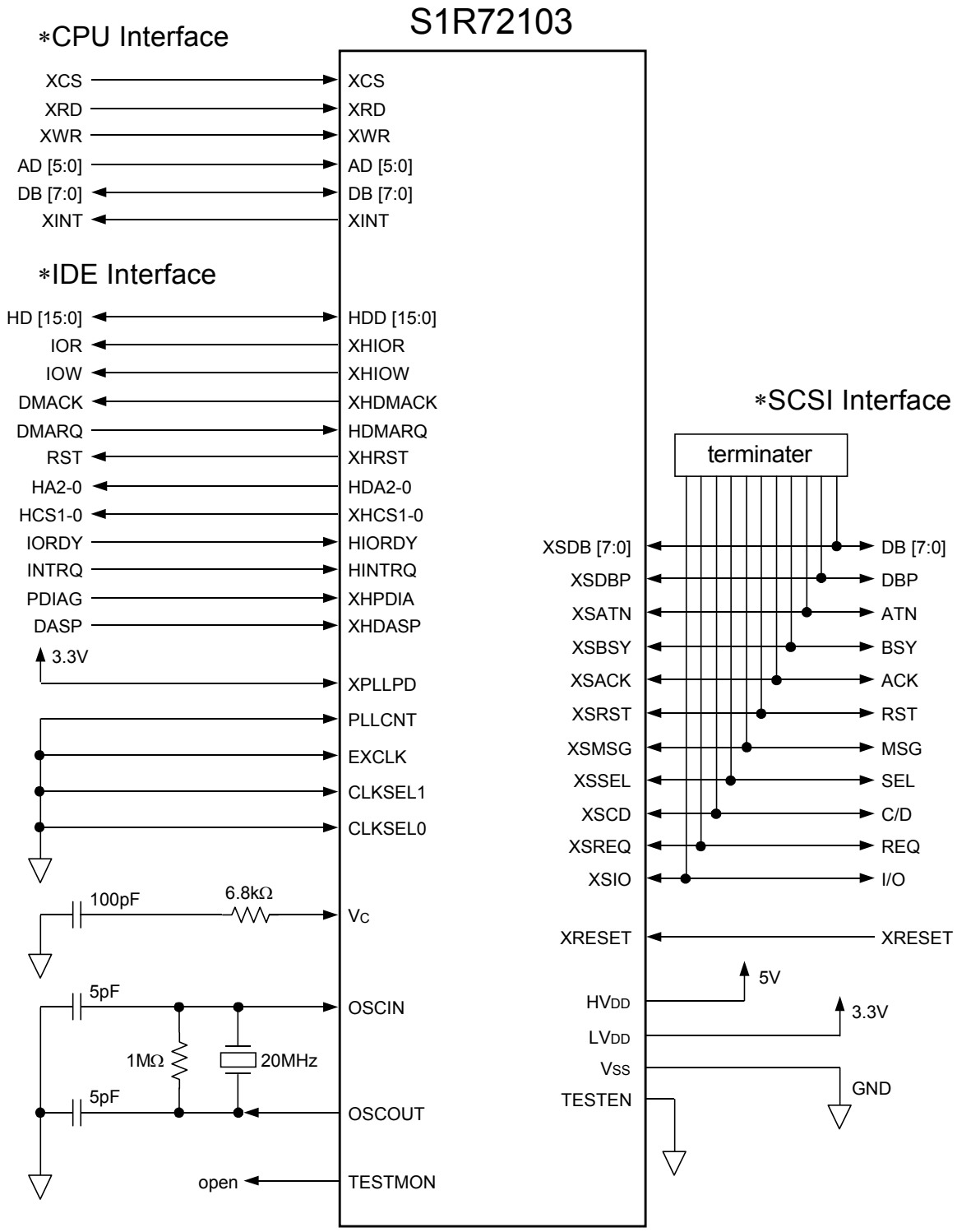
[DMA=1,ULTRA=1]



Symbol	Specification	Min.	Typ.	Max.	単位
T391	XHCS0,1 ↑ → XHDMACK ↓ Address setup time	20	—	—	ns
T392	XHDMACK ↑ → XHCS0,1 ↓ Address hold time	40	—	—	ns
T393	HDMARQ ↑ → XHDMACK ↓ XHDMACK response time	20	—	—	ns
T394	XHDMACK ↓ → XHIOW ↓ envelop time	20	—	—	ns
T395	XHIOW ↓ → HIORDY ↓ Limited interlock Time	40	—	150	ns
T3951	HIORDY ↓ → XHIOR ↓ Unlimited interlock Time	IDEUMD (CYC)×25	—	—	ns
T396	XHIOR ↓ →HDMARQ ↓ Strobe edge to negation DMARQ	30	—	—	ns
T397	HDMARQ ↓ →XHIOR ↑ Limited interlock time	40	—	90	ns
T398	XHIOR ↑ → HIORDY ↑ Strobe to DMARDY time	—	—	IDEUMD (CYC+1)×25-5	ns
T399	XHIOR ↓ → XHIOW ↑ Strobe edge to negation STOP	50	—	—	ns
T39a	HDMACK ↓ → HDD Data bus output delay time	—	—	-125	ns
T39b	XHIOR ↑ → HDD Data hold time	20	—	—	ns
T39c	XHIOR ↑ → HDD CRC data output delay time	—	0	—	ns
T39d	HDD → HDMACK ↑ CRC data setup time	75	—	95	ns
T39e	HDMACK ↑ → HDD CRC data hold time	6	—	25	ns
T39f	HIORDY ↑ → HIORDY ↓ HIORDY pulth width	—	IDEUMD (CYC+2)×25	—	ns
T39g	HDD → XHIOR ↑ Data setup time	—	IDEUMD (CYC+1)×25	—	ns

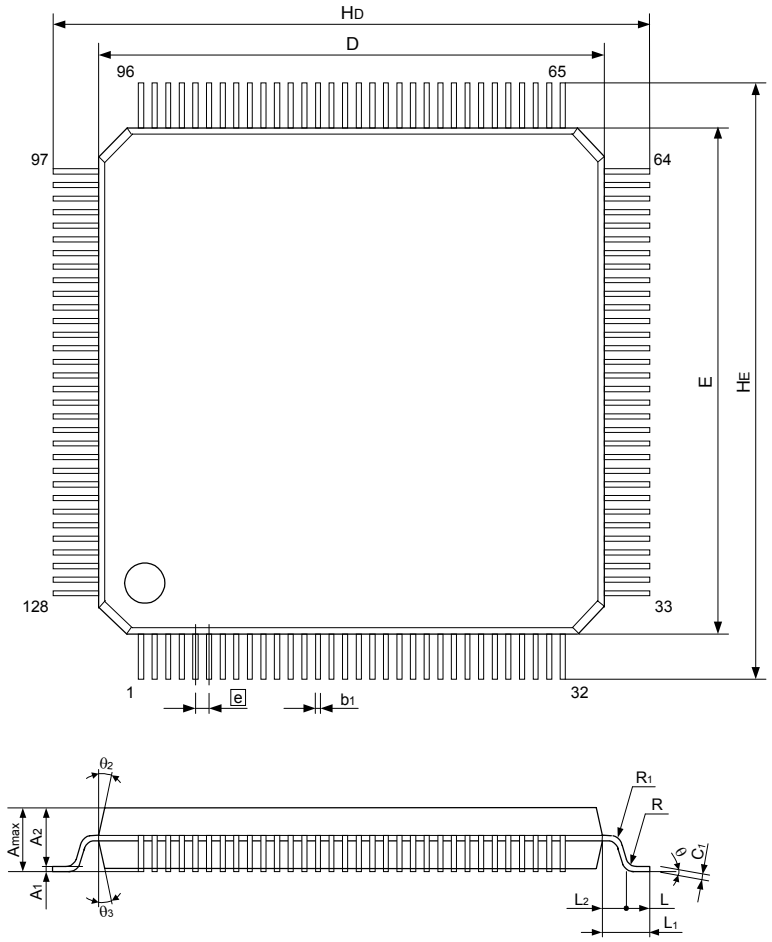
9. EXAMPLES OF CONNECTION

(When 20MHz OSC oscillation is used)



10. EXTERNAL DIMENSIONS DRAWING

Plastic QFP15-100 pin



	Symbol	Dimension in Millimeters		
		Min.	Nom.	Max.
*	E	13.9	14	14.1
*	D	13.9	14	14.1
	AMax.			1.7
	A_1		0.1	
	A_2	1.3	1.4	1.5
	e		0.4	
*	b_1	0.11	0.16	0.26
*	C_1	0.1	0.125	0.175
	θ	0°		10°
	L	0.3	0.5	0.7
	L_1		1	
	L_2		0.5	
	H_E	15.6	16	16.4
	H_D	15.6	16	16.4
	θ_2		12°	
	θ_3		12°	
	R		0.2	
	R_1		0.2	

* E,D Excluding the tie bar cutting stub.

b_1 Lead width of basemetal.

c_1 Lead thickness of basemetal.

1 = 1mm

EPSON

International Sales Operations

AMERICA

EPSON ELECTRONICS AMERICA, INC.
HEADQUARTERS
150 River Oaks Parkway
San Jose, CA 95134, U.S.A.
Phone: +1-408-922-0200 FAX: +1-408-922-0238

SALES OFFICES
West
1960 E.Grand Avenue
El Segundo, CA 90245, U.S.A.
Phone: +1-310-955-5300 FAX: +1-310-955-5400

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 Munich, GERMANY
Phone: +49-(0)89-14005-0 FAX: +49-(0)89-14005-110

DÜSSELDORF BRANCH OFFICE
Altstadtstrasse 176
51379 Leverkusen, GERMANY
Phone: +49-(0)2171-5045-0 FAX: +49-(0)2171-5045-10

UK & IRELAND BRANCH OFFICE
Unit 2.4, Doncastle House, Doncastle Road
Bracknell, Berkshire RG12 8PE, ENGLAND
Phone: +44-(0)1344-381700 FAX: +44-(0)1344-381701

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

BARCELONA BRANCH OFFICE
Barcelona Design Center
Edificio Testa, Avda. Alcalde Barrils num. 64-68
E-08190 Sant Cugat del Vallès, SPAIN
Phone: +34-93-544-2490 FAX: +34-93-544-2491

Scotland Design Center
Integration House, The Alba Campus
Livingston West Lothian, EH54 7EG, SCOTLAND
Phone: +44-1506-605040 FAX: +44-1506-605041

ASIA

EPSON (CHINA) CO., LTD.
23F, Beijing Silver Tower 2# North RD DongSanHuan
ChaoYang District, Beijing, CHINA
Phone: 64106655 FAX: 64107319

SHANGHAI BRANCH
7F, High-Tech Bldg., 900, Yishan Road,
Shanghai 200233, CHINA
Phone: 86-21-5423-5577 FAX: 86-21-5423-4677

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

EPSON TAIWAN TECHNOLOGY & TRADING LTD.
10F, No. 287, Nanking East Road, Sec. 3
Taipei
Phone: 02-2717-7360 FAX: 02-2712-9164
Telex: 24444 EPSONTB

HSINCHU OFFICE
13F-3, No. 295, Kuang-Fu Road, Sec. 2
HsinChu 300
Phone: 03-573-9900 FAX: 03-573-9169

EPSON SINGAPORE PTE., LTD.
No. 1 Temasek Avenue, #36-00
Millenia Tower, SINGAPORE 039192
Phone: +65-6337-7911 FAX: +65-6334-2716

SEIKO EPSON CORPORATION
KOREA OFFICE
50F, KLI 63 Bldg., 60 Yoido-dong
Youngdeungpo-Ku, Seoul, 150-763, KOREA
Phone: 02-784-6027 FAX: 02-767-3677

GUMI OFFICE
6F, Good Morning Securities Bldg., 56 Songjeong-Dong,
Gumi-City, Seoul, 730-090, KOREA
Phone: 054-454-6027 FAX: 054-454-6093

SEIKO EPSON CORPORATION
ELECTRONIC DEVICES MARKETING DIVISION

IC 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
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-(0)42-587-5814 FAX: +81-(0)42-587-5117

In pursuit of **“Saving” Technology**, Epson electronic devices.
Our lineup of semiconductors, displays and quartz devices
assists in creating the products of our customers’ dreams.
Epson IS energy savings.

S1R72103
Technical Manual

SEIKO EPSON CORPORATION
ELECTRONIC DEVICES MARKETING DIVISION

■ EPSON Electronic Devices Website

http://www.epsondevice.com/

First issue October, 2002
Printed in Japan Ⓗ A