FUJITSU SEMICONDUCTOR
DATA SHEET

DS07-03108-2E

Processor Digital Signal Processor

CMOS

16-bit Fixed-point DSP

MB86330

■ DESCRIPTION

The MB86330 is a 16-bit fixed-point DSP (Digital Signal Processor) that is based on Fujitsu-specific Dual-MAC architecture, and can implement product addition operations and double transfer at a high rate and under low power consumption.

The DSP supports a set of instructions optimum for digital signal processing in communications applications such as handy phones.

The MB86330 consists of a core section and a peripheral section. For detailed specifications of the core section, see MB86330DSP Core Section Specifications.

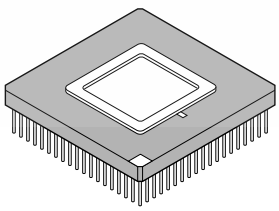
■ FEATURES

- Fixed-point operations
  Multiplication: 16 bits × 16 bits → 31 bits
  Addition: 40 bits + 40 bits → 40 bits
  Product addition: 40 bits ±16 bits × 16 bits → 40 bits
  Maximum operation speed: 100 MIPS at 3.3 V
- Memory configuration
  Data RAM: Two sectors that can be accessed concurrently
  An external RAM (ERAM) is supported.
  Memory mapped I/O system characterized by allocation of I/O devices in the memory space
  Instruction RAM: 48 Kwords × 16 bits
  Table RAM: 16 Kwords × 16 bits

(Continued)

■ PACKAGE

256-pin Ceramic PGA



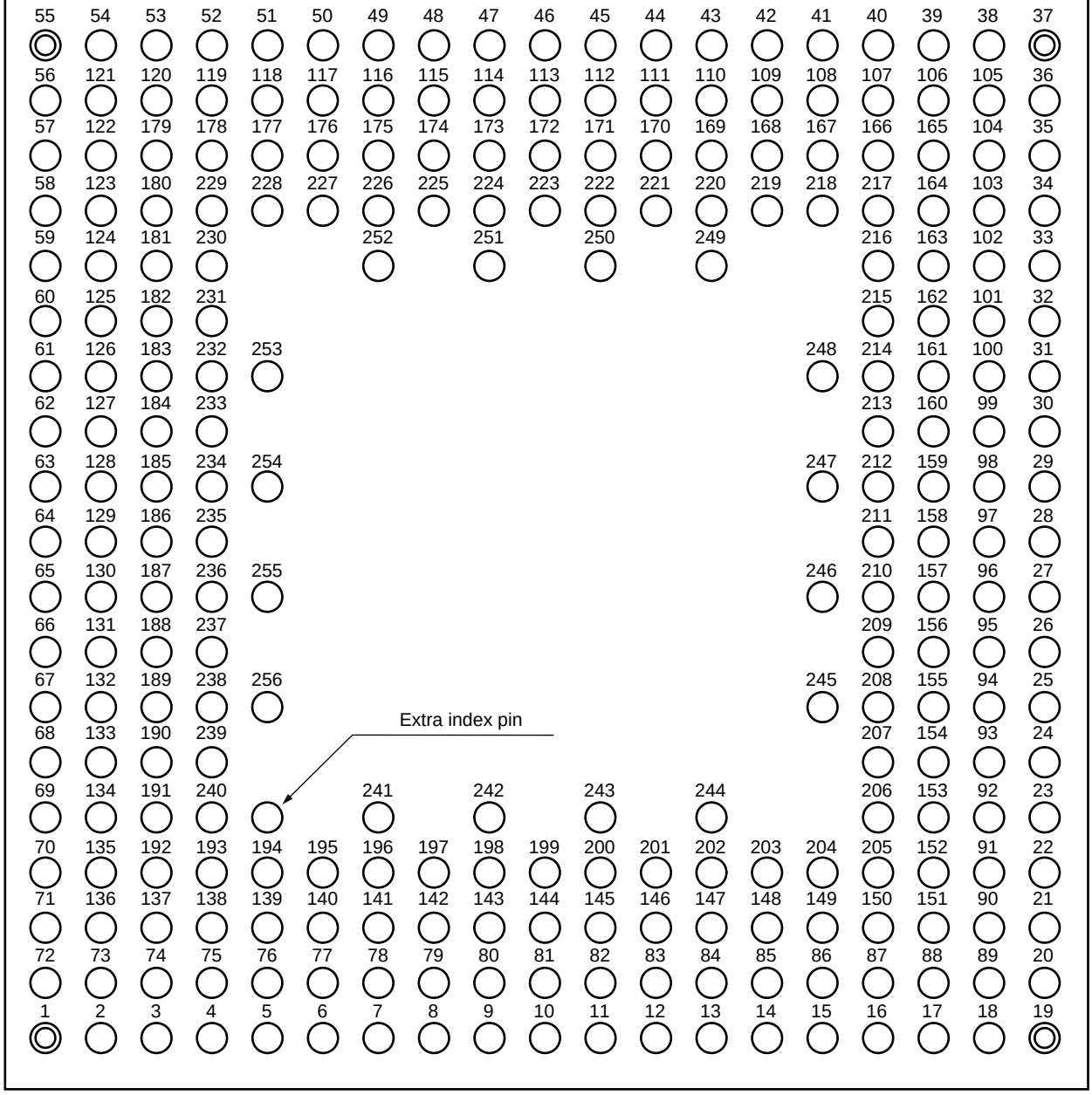
(PGA-256C-A03)

(Continued)

- Addressing
  Two independent address units
  Eight general-purpose registers
  Addressing function that can update a register
  Circular addressing
  Two address update registers
- Supply voltage: 3.3 V (single type of supply voltage)
- Ceramic package: PGA-256

■ PIN ASSIGNMENT

(Top view)



(PGA-256C-A03)

Pin no.	I/O	Pin name	Pin no.	I/O	Pin name	Pin no.	I/O	Pin name	Pin no.	I/O	Pin name
1	O	PAO27	33	I/O	ED13	65	O	PAO7	97	I/O	EDI
2	I	ICAD0	34	—	N.C.	66	O	PAO10	98	I/O	ED3
3	O	PAO30	35	I/O	ICDT12	67	O	ST2	99	—	N.C.
4	O	IRO0	36	I/O	ICDT10	68	O	PAO13	100	I/O	ED10
5	O	IRO3	37	I/O	ICDT6	69	O	PAO15	101	I/O	ED12
6	I	ICAD1	38	—	N.C.	70	O	FF	102	I/O	ED15
7	O	IRO8	39	I/O	ICDT3	71	O	PAO21	103	I/O	ICDT13
8	O	IRO10	40	—	N.C.	72	O	PAO23	104	I/O	ICDT11
9	O	IRO14	41	O	AINT6	73	O	PAO24	105	I/O	ICDT9
10	I	ICAD3	42	O	AINT5	74	O	PAO28	106	I/O	ICDT5
11	O	IRO17	43	I	INT4	75	O	PAO31	107	I/O	ICDT2
12	O	IRO20	44	I	SCZC	76	O	IRO1	108	O	AINT7
13	O	IRO23	45	—	N.C.	77	O	IRO4	109	I	INT6
14	O	IRO26	46	—	N.C.	78	O	IRO6	110	I	INT5
15	O	IRO28	47	—	N.C.	79	I	ICAD2	111	I	INT3
16	O	IRO30	48	—	N.C.	80	O	IRO13	112	—	N.C.
17	—	N.C.	49	I	F0	81	O	IRO16	113	—	N.C.
18	—	N.C.	50	I	MOD0	82	O	IRO18	114	I	MCLK
19	O	PAGE1	51	O	AINT2	83	O	IRO21	115	I	BREAK
20	O	XERD	52	O	AINT1	84	—	N.C.	116	I	MOD2
21	I	WMD0	53	—	N.C.	85	O	IRO27	117	—	N.C.
22	O	EA1	54	—	N.C.	86	—	N.C.	118	I	INT1
23	O	EA4	55	I	SMCK	87	—	N.C.	119	—	N.C.
24	O	EA5	56	I	SMEN	88	—	N.C.	120	—	N.C.
25	O	EA9	57	O	PDXED	89	O	BTACT	121	—	N.C.
26	O	EA11	58	I	SYI1	90	O	XEWR	122	—	VS
27	—	N.C.	59	I	SCI1	91	I	WMD1	123	I	SY10
28	O	EA15	60	O	SDO0	92	O	EA2	124	I	SDI1
29	I/O	ED2	61	O	SDO1	93	—	N.C.	125	I	SYO0
30	I/O	ED6	62	O	PAO0	94	O	EA7	126	I	SYO1
31	I/O	ED8	63	O	PAO4	95	O	EA10	127	O	XMONI
32	—	N.C.	64	I	ICCN	96	O	EA14	128	O	PAO3

N.C.: Pin not connected

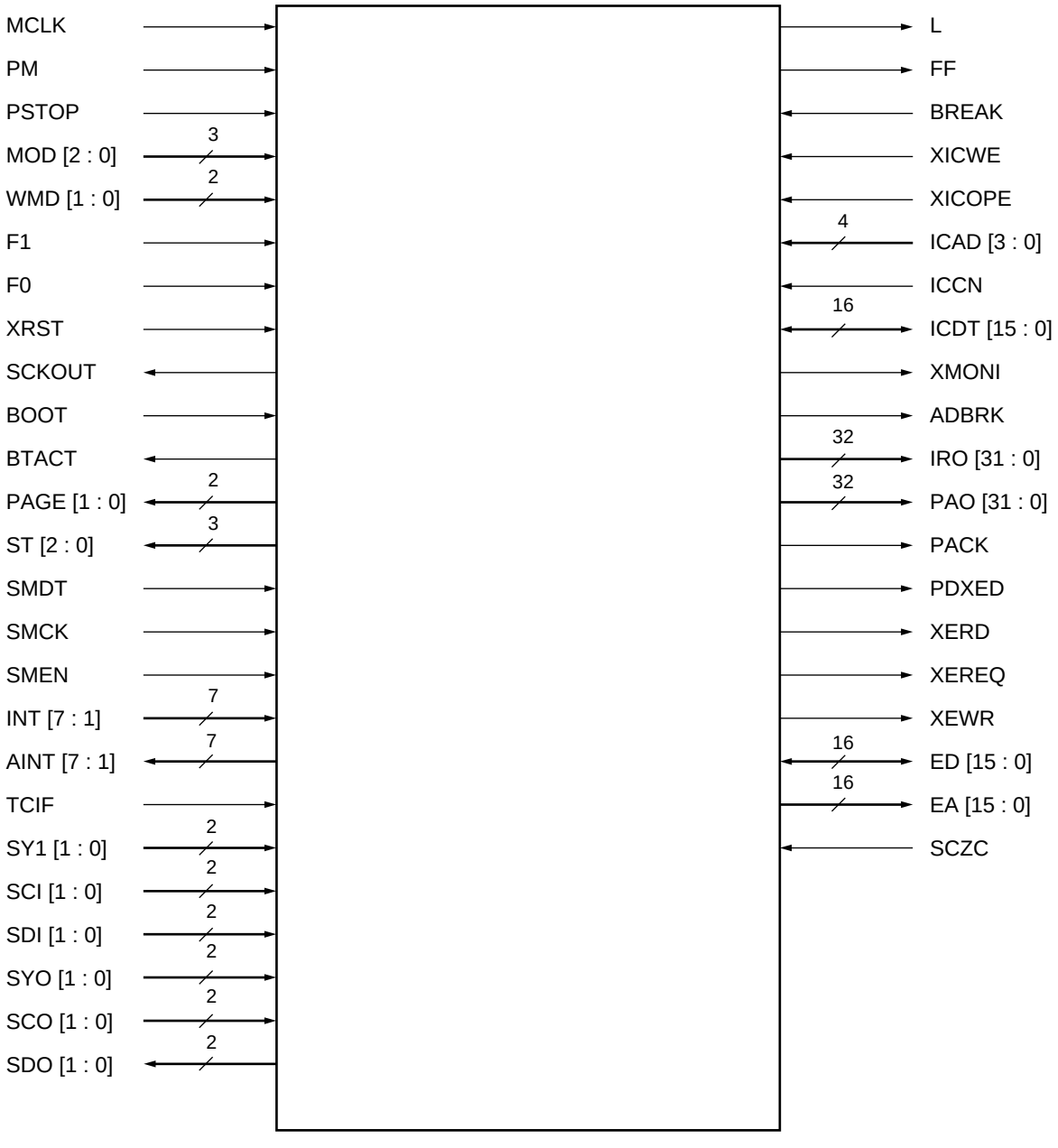
(Continued)

(Continued)

Pin no.	I/O	Pin name	Pin no.	I/O	Pin name	Pin no.	I/O	Pin name	Pin no.	I/O	Pin name
129	O	PAO6	161	I/O	ED11	193	O	PAO26	225	—	N.C.
130	O	PAO8	162	I/O	ED14	194	—	VS	226	—	VS
131	O	PAO11	163	I/O	ICDT14	195	—	N.C.	227	—	N.C.
132	O	ST1	164	—	N.C.	196	—	VS	228	—	VS
133	O	PAO14	165	I/O	ICDT8	197	O	IRO7	229	I	SMDT
134	O	PAO17	166	I/O	ICDT4	198	O	IRO11	230	—	VS
135	O	PAO20	167	I/O	ICDT1	199	—	VS	231	I	SCI0
136	O	PAO22	168	I	INT7	200	—	N.C.	232	—	VS
137	O	PAO25	169	—	N.C.	201	O	IRO24	233	O	ADBRK
138	O	PAO29	170	O	AINT3	201	—	VS	234	O	PAO1
139	—	N.C.	171	—	N.C.	203	O	IRO31	235	—	VS
140	O	IRO2	172	I	PSTOP	204	—	VS	236	I	XICOPE
141	O	IRO5	173	I	PM	205	O	PAGE0	237	O	ST0
142	O	IRO9	174	I	F1	206	—	VS	238	—	VS
143	O	IRO12	175	I	MOD1	207	O	EA0	239	O	PAO18
144	O	IRO15	176	I	INT2	208	—	VS	240	—	VS
145	O	IRO19	177	I	XRST	209	O	EA8	241	—	VD
146	O	IRO22	178	—	N.C.	210	O	EA12	242	—	VD
147	O	IRO25	179	—	N.C.	211	—	VS	243	—	VD
148	O	IRO29	180	I	TCIF	212	I/O	ED5	244	—	VD
149	—	N.C.	181	I	SDI0	213	I/O	ED9	245	—	VD
150	—	N.C.	182	I	SCO0	214	—	VS	246	—	VD
151	I	BOOT	183	O	PACK	215	I/O	ICDT15	247	—	VD
152	O	XEREQ	184	I	SCO1	216	—	VS	248	—	VD
153	—	N.C.	185	O	PAO2	217	I/O	ICDT7	249	—	VD
154	O	EA3	186	O	PAO5	218	—	VS	250	—	VD
155	O	EA6	187	O	PAO9	219	I/O	ICDT0	251	—	VD
156	—	N.C.	188	O	PAO12	220	—	VS	252	—	VD
157	O	EA13	189	I	XICWE	221	O	AINT4	253	—	VD
158	I/O	ED0	190	O	PAO16	222	O	SCKOUT	254	—	VD
159	I/O	ED4	191	O	PAO19	223	—	VS	255	—	VD
160	I/O	ED7	192	O	L	224	—	N.C.	256	—	VD

N.C.: Pin not connected

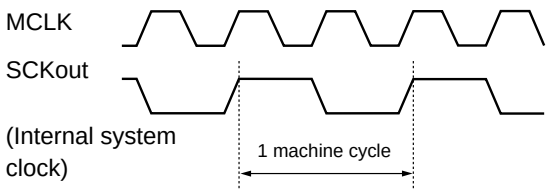
■ EXTERNAL PIN LAYOUT



Other pins

VD, VS, N.C.

■ PIN DESCRIPTION

<table>
<tr><th>Pin no.</th><th>Pin name</th><th>Bit</th><th>I/O</th><th>Active</th><th>Pull up or pull down</th><th>Function</th></tr>
<tr><td>114</td><td>MCLK</td><td>1</td><td>I</td><td>—</td><td>—</td><td>Master clock input
 
</td></tr>
<tr><td>173</td><td>PM</td><td>1</td><td>I</td><td>—</td><td>Pull up</td><td>Internal master clock input can be selected.
0: MCLK, 1: PLL output</td></tr>
<tr><td>172</td><td>PSTOP</td><td>1</td><td>I</td><td>H</td><td>—</td><td>PLL operation setup
0: PLL operation, 1: PLL stop</td></tr>
<tr><td>50,
116,
175</td><td>MOD
[2:0]</td><td>3</td><td>I</td><td>—</td><td>Pull down</td><td>Operating mode
<table>
<tr><th>MOD2</th><th>MOD1</th><th>MOD0</th><th>Operating mode</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Single chip mode</td></tr>
<tr><td colspan="3">Other than above</td><td>Disabled</td></tr>
</table></td></tr>
<tr><td>21,
91</td><td>WMD
[1:0]</td><td>2</td><td>I</td><td>—</td><td>Pull down</td><td>External memory WAIT mode
<table>
<tr><th>WMD1</th><th>SMD0</th><th>Wait cycle</th><th>Can data be rewritten?</th></tr>
<tr><td>0</td><td>0</td><td>0 cyc</td><td>No</td></tr>
<tr><td>0</td><td>1</td><td>5 cyc</td><td>No</td></tr>
<tr><td>1</td><td>0</td><td>15 cyc</td><td>Yes</td></tr>
<tr><td>1</td><td>1</td><td>30 cyc</td><td>Yes</td></tr>
</table></td></tr>
<tr><td>174</td><td>F1</td><td>1</td><td>I</td><td>—</td><td>—</td><td>Flag input 1 (level sense)</td></tr>
<tr><td>49</td><td>F0</td><td>1</td><td>I</td><td>—</td><td>—</td><td>Flag input 0 (level sense)</td></tr>
<tr><td>177</td><td>XRST</td><td>1</td><td>I</td><td>L</td><td>—</td><td>Reset input</td></tr>
<tr><td>222</td><td>SCKOUT</td><td>1</td><td>O</td><td>—</td><td>—</td><td>Internal system clock output</td></tr>
<tr><td>44</td><td>SCZC</td><td>1</td><td>I</td><td>H</td><td>Pull down</td><td>Hi-z control over SCKOOUT, POUT and EA [15:0] (SCZC = “L”)</td></tr>
<tr><td>151</td><td>BOOT</td><td>1</td><td>I</td><td>H</td><td>—</td><td>Input for a BOOT mode control signal</td></tr>
<tr><td>89</td><td>BTACT</td><td>1</td><td>O</td><td>H</td><td>—</td><td>Output for a BOOT mode status indication signal</td></tr>
<tr><td>19,
205</td><td>PAGE
[1:0]</td><td>2</td><td>O</td><td>—</td><td>—</td><td>Output for an external memory/page selection control signal</td></tr>
<tr><td>67,
132,
237</td><td>ST
[2:0]</td><td>3</td><td>O</td><td>—</td><td>—</td><td>Internal status output</td></tr>
</table>

(Continued)

Pin no.	Pin name	Bit	I/O	Active	Pull up or pull down	Function
229	SMDT	1	I	—	—	Serial input data (16 bits) for operating mode (SMODE) setup
55	SMCK	1	I	—	—	Serial input clock for operating mode (SMODE) setup
56	SMEN	1	I	H	—	Pulse input for operating mode (SMODE) setup Upon completion of setup, a positive pulse is entered.
43, 109 to 111, 118, 168, 176	INT [7:1]	7	I	L	Pull up	Input for INT7 to INT1 interrupt request signals
41, 42, 51, 52, 108, 170, 221	AINT [7:1]	7	O	L	—	Output for INT7 to INT1 interrupt acknowl edge signals
180	TCIF	1	I	H	—	Used for DC setup. “0”: PCM, “1”: TCH Used to set serial port 1.
58, 123	SYI [1:0]	2	I	H	—	Input pins for synchronization signals for serial input port 1/0
59, 231	SCI [1:0]	2	I	—	—	Clock input for serial input port 1/0
124, 181	SDI [1:0]	2	I	—	—	Data input for serial input port 1/0
125, 126	SYO [1:0]	2	I	H	—	Synchronization signal input for serial output port 1/0
182, 184	SCO [1:0]	2	I	—	—	Clock input for serial output port 1/0
60, 61	SDO [1:0]	2	O	—	—	Data output for serial output port 1/0
192	L	1	O	—	—	PLL status output
70	FF	1	O	—	—	Output for test
115	BREAK	1	I	L	Pull up	Break input for the emulator
189	XICWE	1	I	L	Pull up	Input for an emulator write signal
236	XICOPE	1	I	L	Pull up	Input for an emulator read signal
2, 6, 10, 79	ICAD [3:0]	4	I	—	Pull down	Address input for the emulator
64	ICCN	1	I	H	Pull down	Input for an emulator connection signal
35 to 37,39, 103 to 107, 163, 165 to 167, 215, 217, 219	ICDT [15:0]	16	I/O	—	Pull down	I/O for a data bus used to access the emulator
127	XMONI	1	O	L	—	Output for indicating emulator monitor mode status
233	ADBRK	1	O	H	—	Output for indicating occurrence of an ADBKP register event for the emulator

(Continued)

(Continued)

Pin no.	Pin name	Bit	I/O	Active	Pull up or pull down	Function
4, 5, 7 to 9, 11 to 16, 76 to 78, 80 to 83, 85, 140 to 148, 197, 198, 201, 203	IRO [31:0]	32	O	—	—	Instruction register output for the emulator
1, 3, 62, 63, 65, 66, 68, 69, 71 to 75, 128 to 131, 133 to 138, 185 to 188, 190, 191, 193, 234, 239	PAO [31:0]	32	O	—	—	Program address output for the emulator
183	PACK	1	O	—	—	Output for a PAO fetch clock for the emulator
57	PDXED	1	O	—	—	Output for test
20	XERD	1	O	L	—	Output for an ERAM reading signal
152	XEREQ	1	O	L	—	Output for an ERAM access request signal
90	XEWR	1	O	L	—	Output for an ERAM writing signal
29 to 31, 33, 97, 98, 100 to 102, 158 to 162, 212, 213	ED [15:0]	16	I/O	—	Pull up	External data bus I/O pins
22 to 26, 28, 92, 94 to 96, 154, 155, 157, 207, 209, 210	EA [15:0]	16	O	—	—	Output for the ERAM address
241 to 256	VD	—	—	—	—	Power supply for the digital circuit (3.3 V, input)
122, 194, 196, 199, 202, 204, 206, 208, 211, 214, 216, 218, 220, 223, 226, 228, 230, 232, 235, 238, 240	VS	—	—	—	—	GND (input) for the digital circuit
17, 18, 27, 32, 34, 38, 40, 45 to 48, 53, 54, 84, 86 to 88, 93, 99, 112, 113, 117,119 to 121, 139, 149, 150, 153, 156, 164, 169, 171, 178, 179, 195, 200, 224, 225, 227	N.C.	—	—	—	—	Pins not connected

■ HANDLING DEVICES

1. Take Care So that the Maximum Rated Value Is Not Exceeded. (Preventing Latchup)

Latchup may occur on CMOS ICs if voltage higher than VD or lower than VS is applied to input and output pins other than medium- and high-voltage pins or if higher than the voltage which shows on “1. Absolute Maximum Ratings” in section “■ Electrical Characteristics” is applied between VD and VS.

When latchup occurs, power supply current increases rapidly and might thermally damage elements. When using, take great care not to exceed the absolute maximum ratings.

2. Treatment of Unused Input Pins

Leaving unused input pins open could cause malfunctions. They should be connected to a pull-up or pull-down resistor.

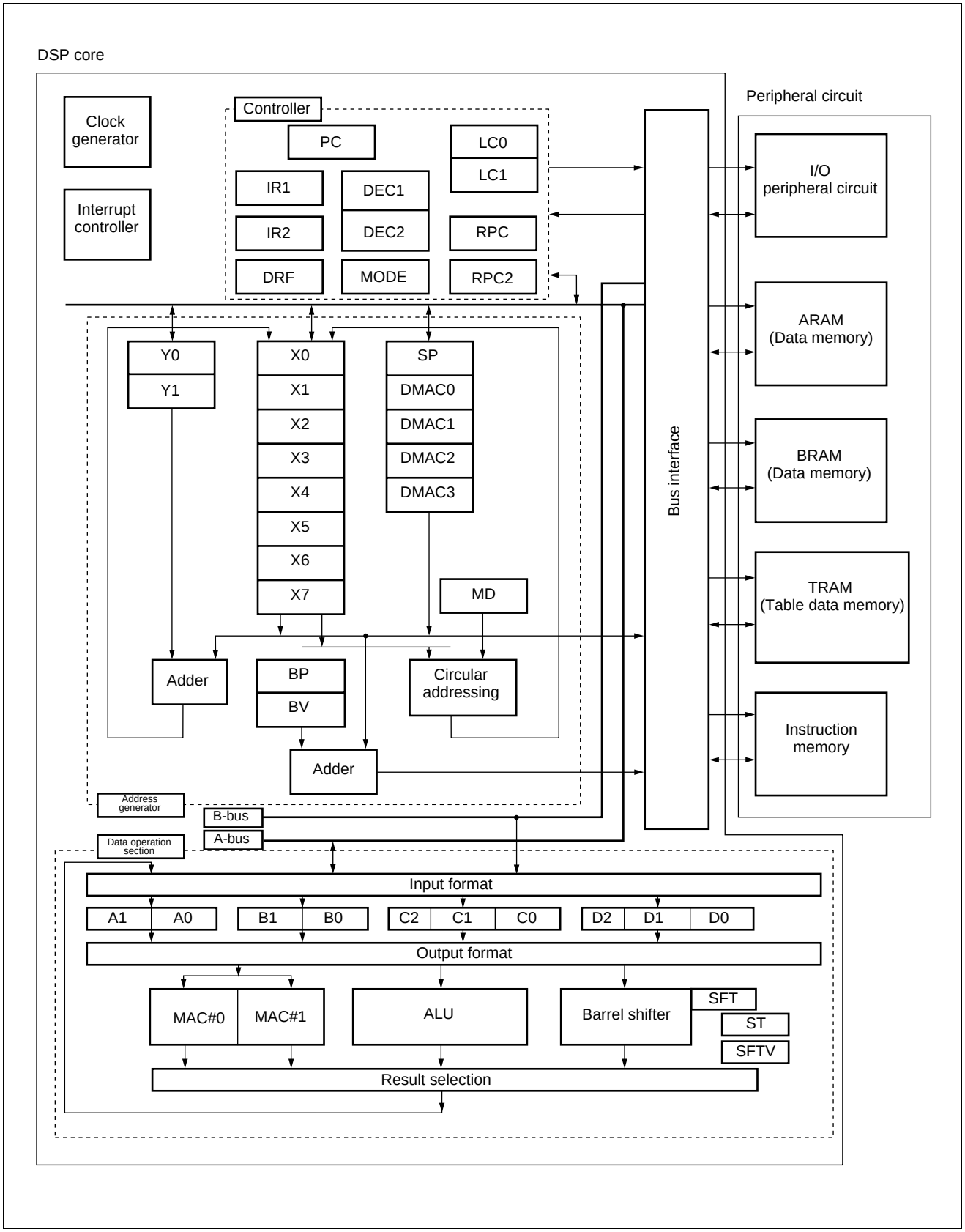
3. Treatment of N.C. Pins

Be sure to leave (internally connected) N.C. pins open.

4. Treatment of Pins Connected to Pull-up/Pull-down Resistors

With neither a pull-up resistor nor a pull-down resistor connected, the pin state is determined depending on the input level that reflects an internal resistor. When controlling the pin state, however, connect a pull-up or pull-down resistor.

■ BLOCK DIAGRAM



■ DESCRIPTION OF BLOCK FUNCTIONS

- **Clock generator (CLOCK Gen.)**
  Generates a clock required for the DSP to control a system clock stop in the waste state and an entire clock stop for sleeping.

- **Interrupt controller**
  Controls an INT interrupt, an overflow interrupt, and a DMA interrupt.

- **Controller**
  Generates a program address and decodes an instruction to control the entire DSP.

- **Address generator**
  Generates an address required for memory access. It supports a circular addressing function to control the DMA access pointer and the stack pointer.

- **Data operation section**
  Performs data operations such as product addition, arithmetic operations (multiplication, division, addition and subtraction), logic operations, and shift operations.

- **Bus interface**
  Controls access to the memory space including instruction reading, data memory access and mapped I/O access.

- **ARAM/BRAM**
  This is a data memory for data operations. A and B use different banks, enabling double transfer without a wait. (ARAM = 4 kwords, BRAM = 4 kwords)

- **TRAM (Table data memory)**
  Sets table data required for applications.

- **Instruction memory**
  Sets program data.

- **Mapped I/O**
  Supports a macro valid for applications
  Serial I/O: Two serial ports for transmitting CODEC data.
  One serial input 1 system for setting operating mode (SMODE)

■ MEMORY SPACE

- **Configuration of the memory space**
  The memory space consists of a data memory space and an instruction memory space. The I/O, ARAM, BRAM, TABLE, and ERAM (external memory) areas are allocated in the data memory space, while the instruction memory is allocated in the program memory space. The I/O, TABLE, ERAM (external memory), ARAM and BRAM areas in the data memory space are, however, assigned a particular data bus. The memory for any two areas can, therefore, be accessed concurrently during one cycle. What memory is accessed is determined automatically by an address value.

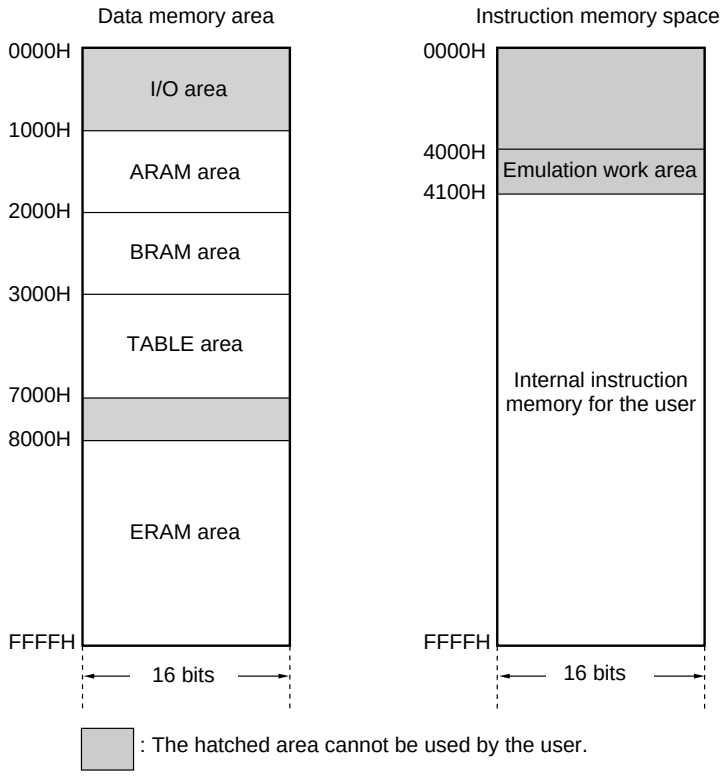
  Addresses allocated for the memories, and their maximum size are determined as follows. Select a memory size to be allocated in this range.

- **Memory space mode**
  You can select a method for allocating the data memory space and the program memory space by operating mode. Determine operating mode by mode pins (MOD2, MOD1 and MOD0) as follows.

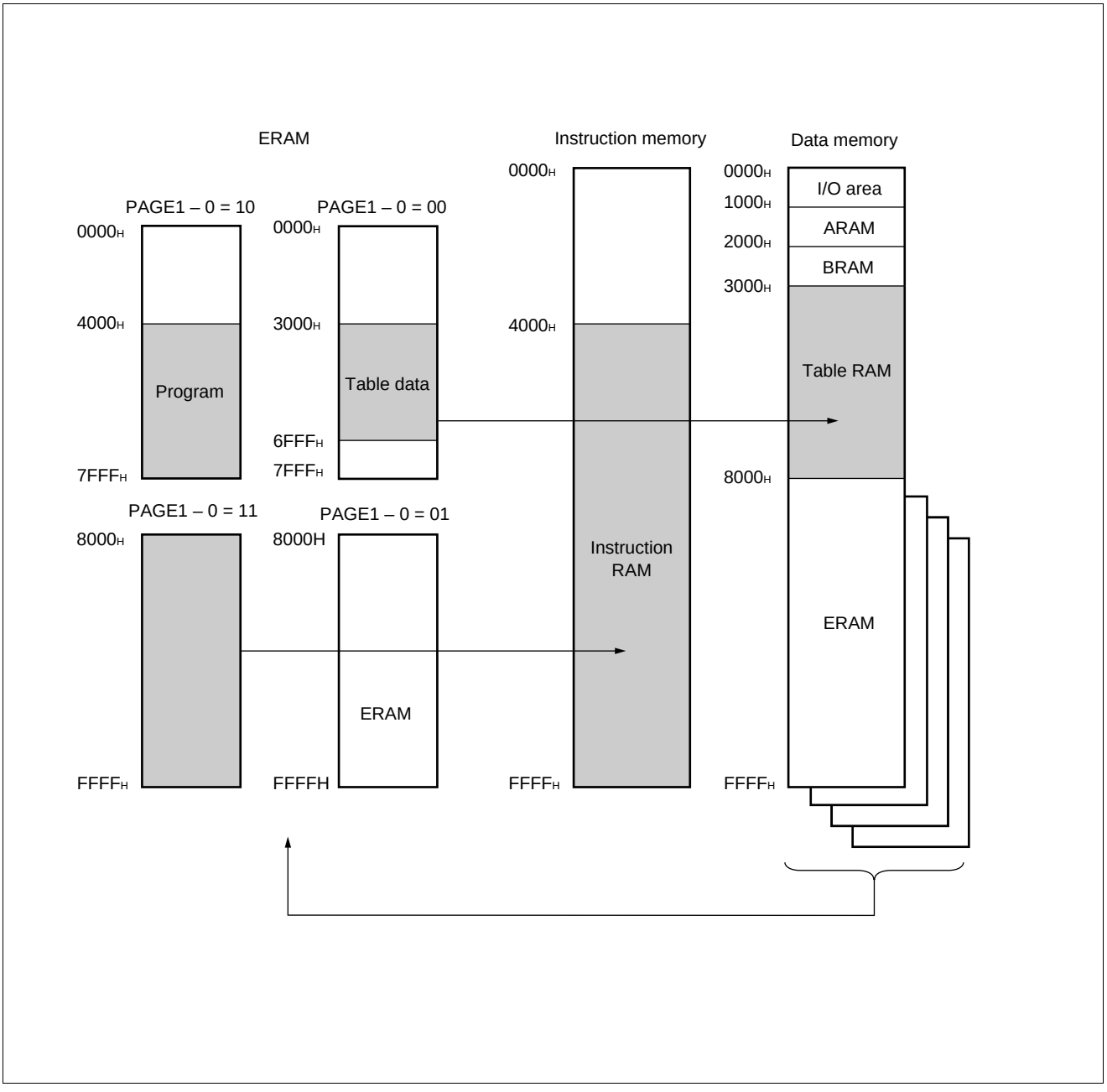
MOD2	MOD1	MOD0	Operating mode
0	0	0	Single chip mode
Other than above			Disabled

- **Memory map for single chip mode**
  The program for single chip mode (MOD2: 0 = “000”) is operated by the internal program RAM (which externally downloads a program). Five areas for I/O, ARAM, BRAM, TABLE and ERAM are allocated on the data memory space, with the instruction memory allocated on a program memory space completely different from the data memory space. Although the instruction memory can have an independent address space for 64 kwords, therefore, you cannot access data in the instruction memory using a program.

- **Single chip mode (MOD2: 0 = “000”)**



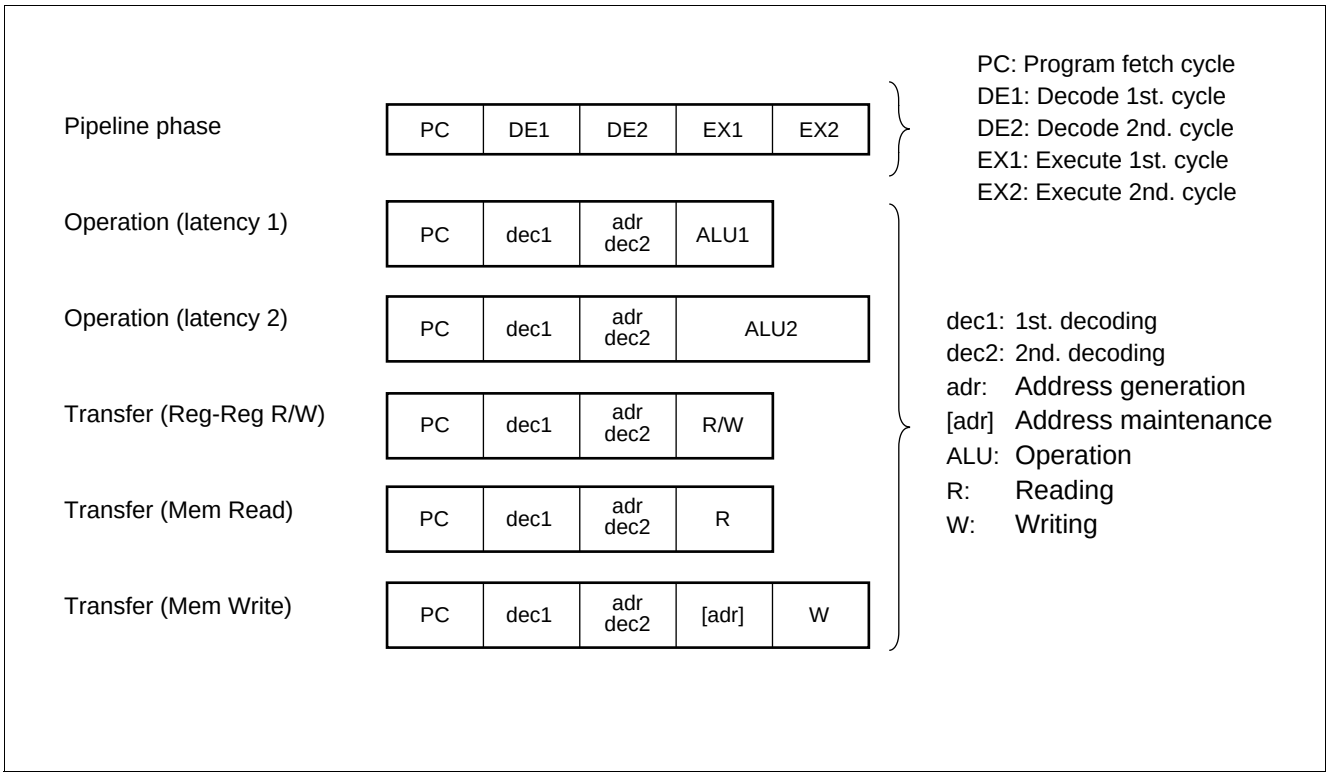
■ MEMORY MAP FOR BOOT STRAP



The ERAM area is used by the DSP to access the data memory space. Because the ERAM area ranges from 8000H to FFFFH, however, PAGE is created in units of 32 kwords. PAGE is selected automatically.

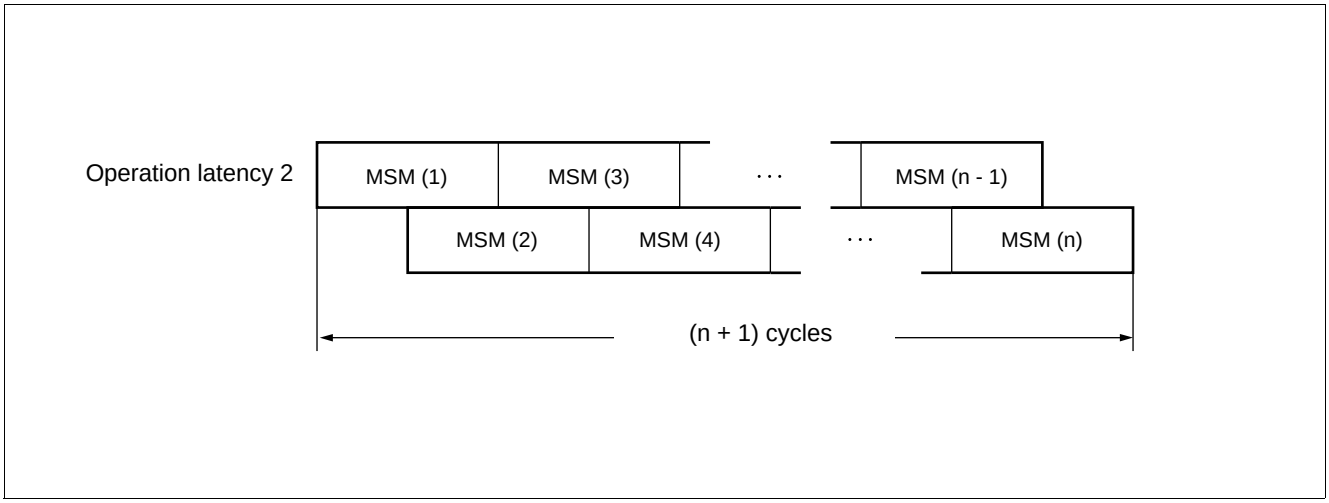
■ BASIC PIPELINE OPERATION

The DSP splits the contents of processing in one cycle to increase the number of pipeline sectors for high-speed operation. For operations using product adders such as product addition and multiplication, and for operations using 40-bit adders such as 40-bit addition, the processing latency is two cycles.

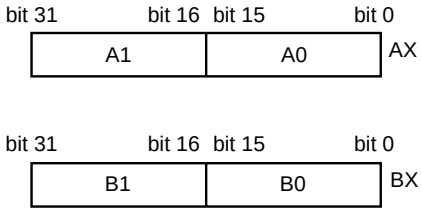
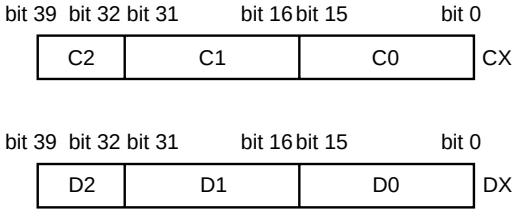
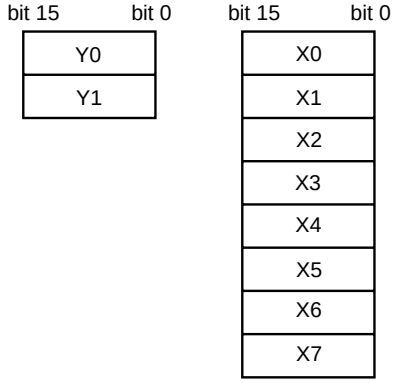


■ PRODUCT ADDITION

For product addition and multiplication, the latency is two cycles. Because it is provided with a dual product adder (MAC) for alternate processing every cycle, however, the DSP can process n successive product addition (multiplication) steps in (n + 1) cycles.

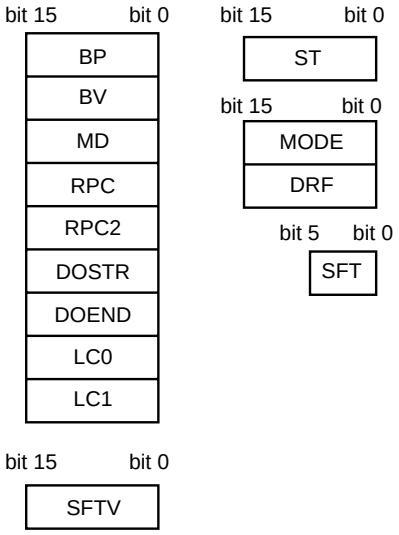
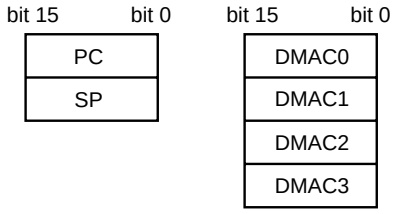


■ REGISTER TABLE

Register name	Bit length	Register type	Initial value	Register configuration
A0	16	Data register	Undefined	
A1	16	Data register		
AX	32	Data register		
B0	16	Data register		
B1	16	Data register		
BX	32	Data register		
C0	16	Accumulator	Undefined	
C1	16	Accumulator		
C2	8	Accumulator (guard register)		
CX	40	Accumulator		
D0	16	Accumulator		
D1	16	Accumulator		
D2	8	Accumulator (guard register)		
DX	40	Accumulator		
X0	16	Address register	Undefined	
X1	16	Address register		
X2	16	Address register		
X3	16	Address register		
X4	16	Address register		
X5	16	Address register		
X6	16	Address register		
X7	16	Address register		
Y0	16	Address register update register		
Y1	16	Address register update register		

(Continued)

(Continued)

Register name	Bit length	Register type	Initial value	Register configuration
BP	16	Base pointer	Undefined	
BV	16	Circular register		
MD	16	Modulo register		
RPC	16	Repeat counter 1		
RPC2	16	Repeat counter 2		
DOSTR	16	DO start address register		
DOEND	16	DO end address register		
LC0	16	Loop counter		
LC1	16	Loop counter		
SFT	6	Shift register		
SFTV	16	Shift register		
ST	16	Status register	00000000B 00000000B	
MODE	16	Mode register		
DRF	16	Flag holding register		
DMAC0	16	DMA counter	Undefined	
DMAC1	16	DMA counter		
DMAC2	16	DMA counter		
DMAC3	16	DMA counter		
PC	16	Program counter	FFFD	
SP	16	Stack pointer	Undefined	

■ REGISTERS

- **Data registers (A0, A1, B0 and B1)**
  Each of the data registers consists of four words (16 bits). They can be used as four word-length registers (16 bits) and two long-word registers (32 bits) to execute various arithmetic operation instructions, logic operation instructions, and transfer instructions.

- **Accumulators (C0 to C1, and D0 to D2)**
  The accumulators can be linked as two 40-bit registers (CX and DX) to execute various arithmetic operation instructions, logic operation instructions, and transfer instructions. The 40-bit length registers (CX and DX) can be specified as destinations for product addition instructions. Four 16-bit length accumulators (C0, C1, D0 and D1), and two 8-bit length accumulators (C2 and D2) are supported.

- **Address registers (X0 to X7)**
  Eight 16-bit address registers are supported. An address register is used to specify an operand address for transfer. Immediate values (1 to –2) or the address update registers (Y0 and Y1) can be used to update address registers. They can also be updated automatically by transfer.

- **Address update registers (Y0 and Y1)**
  Two 16-bit address update registers are supported. The address update registers are used to update address registers during addressing.

- **Base pointer (BP)**
  The base pointer consists of 16 bits. The contents of the base pointer plus a 7-bit immediate value are generated as the address value during direct 7-bit length addressing.

- **Circular register (BV)**
  The circular register, which consists of 16 bits, provides an offset value for circular addressing.

- **Modulo register (MD)**
  The modulo register, which consists of 16 bits, is used to specify an addressing range for circular addressing.

- **Repeat counter (RPC)**
  The repeat counter, which consists of 16 bits, is used to specify the number of times the REP/DO instruction is repeated. During execution of the repeat instruction, the repeat counter is decremented by one every repeat operation cycle.

- **Repeat counter 2 (RPC2)**
  Repeat counter 2, which consists of 16 bits, is used to specify the number of times the REP2 instruction is repeated. During execution of the repeat 2 instruction, repeat counter 2 is decremented by one every repeat operation cycle.

- **DO address registers (DOSTR and DOEND)**
  These registers maintain a loop start address (DOSTR)/end address (DOEND) for the DO instruction. They can process only PUSH/POP.

- **Loop counters (LC0 and LC1)**
  Each of the loop counters consists of 16 bits. They store the number of times repetition is made in a specified address range.

(Continued)

(Continued)

- **Shift register (SFT)**
  The SFT register consists of signed 6 bits. This shift value storage register stores the number of bits shifted during execution of the shift instruction.

- **Shift register (SFTV)**
  The SFTV register, which consists of 16 bits, is used to store the results of CMLT and CMGT instruction.

- **Status register (ST)**
  The status register, which consists of 16 bits, is assigned bits for storing information about results of operations (carry and overflow) and for setting operating mode.

- **Mode register (MODE)**
  This register is used to specify modes of operations and transfer, and interrupts.

- **Flag holding register (DRF)**
  This register holds flags for the DO, REP and REP2 instructions. It can process only PUSH/POP. This register is cleared by the PUSH instruction.

- **DMA counters (DMAC0 to DMAC3)**
  When a DMA interrupt occurs, this register stores the address of the data transfer source or the data transfer destination.

- **Program counter (PC)**
  The program counter, which consists of 16 bits, points to the memory address that stores an instruction code to be executed by the CPU. While it is updated automatically by instruction execution, the program counter can be rewritten by a conditional branch, a subroutine call instruction, an interrupt, and a reset. Executing the repeat instruction stops a program counter update.

- **Stack pointer (SP)**
  The stack pointer, which consists of 16 bits, stores addresses for saving and transferring the contents of registers upon execution of the PUSH/POP instruction, the subroutine call instruction, or an interrupt.

■ DETAILED DESCRIPTION OF SPECIAL REGISTERS

(1) Status Register

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
IT	OV3	OV1	INT2	INT1	INT0	MDMA	CP	RND	ITG	V3	V2	V1	N	Z	C

Bit abbreviation	Bit name	Description
C	Carry flag	Set when carry occurs as a result of operation execution. Reset when no carry occurs. Not changed by transfer instruction execution.
Z	Zero flag	Set when the operation result is 0. Reset when the operation result is not 0. Not changed by transfer instruction execution.
N	Negative flag	Set when the operation result is smaller than 0. Reset when the operation result is equal to or greater than 0. Not changed by transfer instruction execution.
V1	Overflow flag 1	Set when the operation result overflows. Reset when the operation result does not overflow. Not changed by transfer instruction execution.
V2	Overflow flag 2	Set when the operation result overflows. Set V2 is reset only by hardware or by ST programming by the transfer instruction. Not changed by transfer instruction execution.
V3	Overflow flag 3	Set when the operation result of an instruction stored in CX or DX cannot be expressed by 32 bits (but by 40 bits). Reset when the operation result can be expressed by 32 bits.
ITG	Operating mode specification flag	Specify this when executing multiplication in integral mode.
RND	Rounding mode setup	Used to set ON/OFF of rounding processing when data is transferred from a register consisting of 32 or more bits to a 16-bit register.
CP	Clip flag	Used to specify whether the operation result is to be clipped when overflow occurs during the operation.
MDMA	DMA enable flag	Enables a DMA interrupt. (0: Disabled)
INT0	Interrupt enable flag	INT0 (SMODE) interrupt enable flag 0: Disabled, 1: Enabled
INT1	Interrupt enable flag	INT1 interrupt enable flag 0: Disabled, 1: Enabled
INT2	Interrupt enable flag	INT2 interrupt enable flag 0: Disabled, 1: Enabled
OV1	V1 interrupt enable flag	Operation overflow interrupt enable flag. An interrupt is generated when V1 is set. 0: An interrupt is disabled. 1: An interrupt is enabled.
OV3	V3 interrupt enable flag	Operation overflow interrupt enable flag. An interrupt is generated when V3 is set. 0: An interrupt is disabled. 1: An interrupt is enabled.
IT	Interrupt enable flag	OV1, OV3 and INT0 to INT7 interrupt enable flag 0: An interrupt is disabled. 1: An interrupt is enabled.

(2) Mode Register

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
INT7	INT6	INT5	INT4	INT3	—	NCT	NOG

Bit abbreviation	Bit name	Description
NOG	Operating mode specification	0: Ordinary mode The guard bit is used. 1: NOG mode No guard bit is used.
NCT	Transferred data clipping specification	0: Transferred data is clipped. 1: Transferred data is not clipped.
—	Indeterminate	Reserved
INT3	Interrupt enable flag	INT3 interrupt enable flag 0: Disabled, 1: Enabled
INT4	Interrupt enable flag	INT4 interrupt enable flag 0: Disabled, 1: Enabled
INT5	Interrupt enable flag	INT5 interrupt enable flag 0: Disabled, 1: Enabled
INT6	Interrupt enable flag	INT6 interrupt enable flag 0: Disabled, 1: Enabled
INT7	Interrupt enable flag	INT7 interrupt enable flag 0: Disabled, 1: Enabled

(3) DRF Register

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
REPF23	REPF22	REPF21	REPF13	REPF12	REPF11	DOF1	DOF2

Bit abbreviation	Bit name	Description
DOF2	DO flag 2	Internal operation status holding flag. Only PUSH and POP are available. Cleared by the PUSH instruction.
DOF1	DO flag 1	
REPF11	REP1 flag 1	
REPF12	REP1 flag 2	
REPF13	REP1 flag 3	
REPF21	REP2 flag 1	
REPF22	REP2 flag 2	
REPF23	REP2 flag 3	

■ ADDRESSING

- **Types of addressing**
  When reading/writing data from/to the memory, you can use a direct addressing method for specifying a 16-bit length address space with an immediate value, and an indirect addressing method for referencing that space by an address register. The DSP supports eight address registers, two update registers, the base pointer, the circular register, and the modulo register for addressing.

- **Addressing classification**
  Three indirect addressing means (AD0 to AD2) by address registers can be used to transfer a register value to a memory, data from a memory to a register, and data between memories. The available addressing means is determined depending on the type of a register for data transfer, double transfer and transfer accompanied by an operation. All addressing is performed in units of words.

- **Addressing modes**

Mode		Mnemonic	Effective address	Register update	Description
Direct addressing		(imm16)	imm16	Not updated	16-bit direct addressing
Indirect addressing	AD0	(Xk + + 1) (Xk + + 0) (Xk – – 1) (Xk + + Y0)	Xk Xk Xk Xk	+1 Not updated –1 Y0	Xk can be assigned X0, X1, X2 and X3.
	AD1	(Xm + + 3) (Xm + + 2) (Xm + + 1) (Xm + + 0) (Xm – – 1) (Xm – – 2) (Xm – – 3) (Xm + + Y1)	Xm Xm Xm Xm Xm Xm Xm Xm	+3 +2 +1 Not updated –1 –2 –3 Y1	Xm can be assigned X4, X5 and X6.
		[BV+X7 + + 3] [BV+X7 + + 2] [BV+X7 + + 1] [BV+X7 + + 0] [BV+X7 – – 1] [BV+X7 – – 2] [BV+X7 – – 3] [BV+X7 + + Y1]	BV+X7 BV+X7 BV+X7 BV+X7 BV+X7 BV+X7 BV+X7 BV+X7	+3 +2 +1 Not updated –1 –2 –3 Y1	For circular addressing, only X7 can be used.
	AD2	(BP+disp7) (Xn + + 2) (Xn + + 1) (Xn + + 0) (Xn – – 1) (Xn – – 2) (Xn – – 3) (Xn + + Y0) (Xn + + Y1) [BV+Xn + + 2] [BV+Xn + + 1] [BV+Xn + + 0] [BV+Xn – – 1] [BV+Xn – – 2] [BV+Xn – – 3] [BV+Xn + + Y0] [BV+Xn + + Y1]	BP+disp7 Xn Xn Xn Xn Xn Xn Xn Xn BV+Xn BV+Xn BV+Xn BV+Xn BV+Xn BV+Xn BV+Xn BV+Xn	Not updated +2 +1 Not updated –1 –2 –3 Y0 Y1 +2 +1 Not updated –1 –2 –3 Y0 Y1	Xn can be assigned X0, X1, X2, X3, X4, X5, X6 and X7. disp7 is signed 7 bits.

[ ] : Indicates circular addressing.

AD0 : For “AD0,” you cannot specify circular addressing.

AD1 : For “AD1,” circular addressing mode is set automatically when “X7” is selected as the address register. With “0” set in the “MD” and “BV” registers, the same operation as ordinary addressing is performed even if “X7” is selected as the address register.

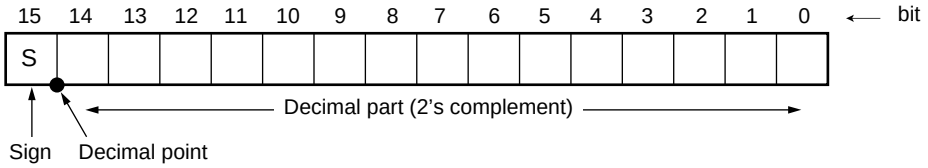
AD2 : disp7 in “AD2” is signed 7 bits.

■ BASIC CONFIGURATION OF THE DATA OPERATION SECTION

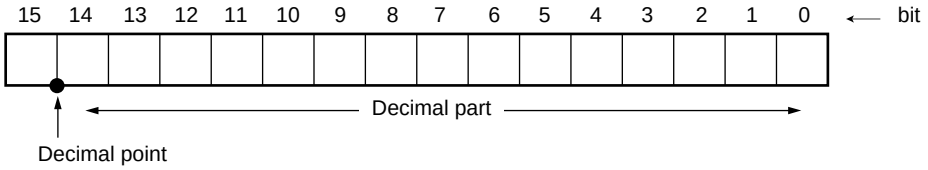
1. Data Format

Integral type and fixed-point data can be handled regardless of whether the data is signed or unsigned. For signed data, the most significant bit indicates a sign. A number of 1 indicates negative data, which is expressed by 2's complement. The decimal point for fixed-point type data is located between the sign bit (bit 14) and its right bit (bit 15). When the accumulator value resulting from execution of a multiplication instruction is transferred to a 16-bit length register or memory, the decimal point is returned to the original position (between bits 14 and 15).

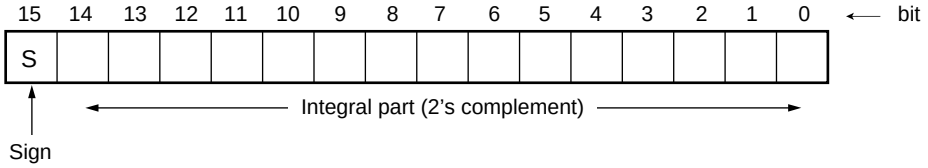
- **Fixed-point type, signed data**



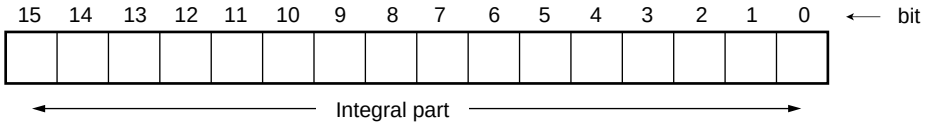
- **Fixed-point type, unsigned data**



- **Integral-type, signed data**



- **Integral-type, unsigned data**



Since a value is handled in 2's complement format for addition and subtraction, no distinction is made in the result of an operation by the above four data format. Since, for multiplication, the result of an operation varies with whether data is signed or unsigned, three combinations of "signed data x signed data," "signed data × unsigned data," and "unsigned data × unsigned data" exist for multiplication instructions.

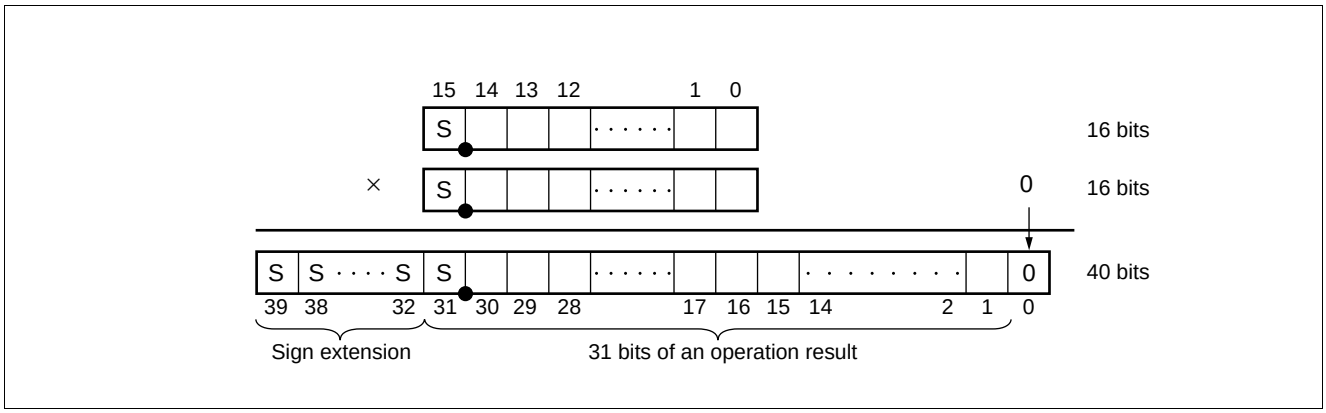
For the fixed-point type and the integral type, the result of multiplication varies with the decimal point position. When fixed-point type data is multiplied, the result of the operation is stored into the accumulator with one bit shifted to the left in comparison with integral type data. The "ITG" bit in the status register is used to switch between the fixed-point type and the integral type. With this bit set at 0, an operation is executed in the fixed-point type format.

2. Multiplication in Fixed-point Type Mode

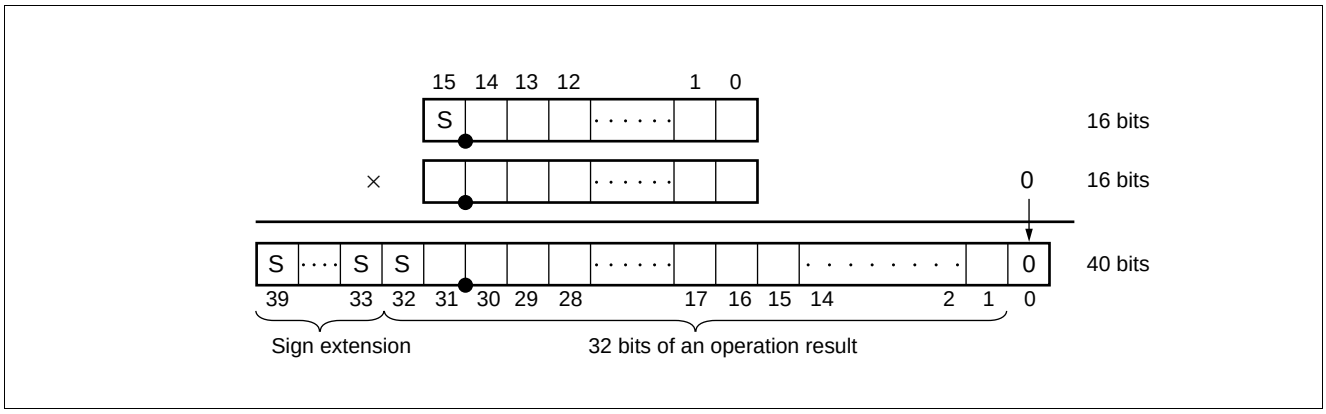
With the “ITG” bit in the status register set at 0, fixed-point type mode is set up, and multiplication is executed in the following format. You can use ordinary mode and NOG mode in which the guard bit is not used.

(1) Ordinary Mode

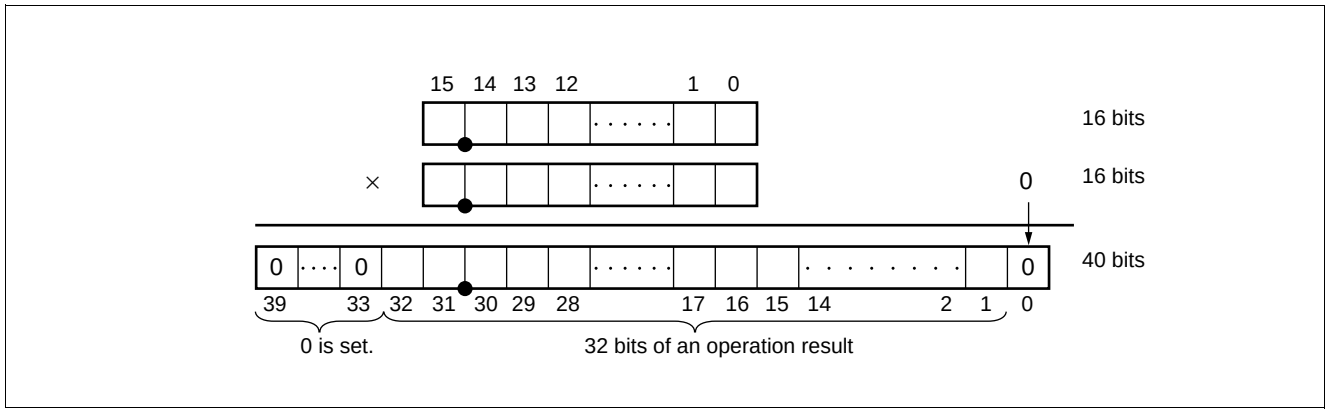
- **Fixed-point type multiplication (signed data × signed data)**
  31 bits of the result of a signed operation are stored into bits 1 to 31 in the accumulator shifted to the left by one bit. “0” is set to bit 0, with the same value as at bit 31 set to bits 32 to 39.



- **Fixed-point type multiplication (signed data × unsigned data)**
  32 bits of the result of a signed operation are stored into bits 1 to 32 in the accumulator shifted to the left by one bit. “0” is set to bit 0, with the same value as at bit 32 set to bits 33 to 39.



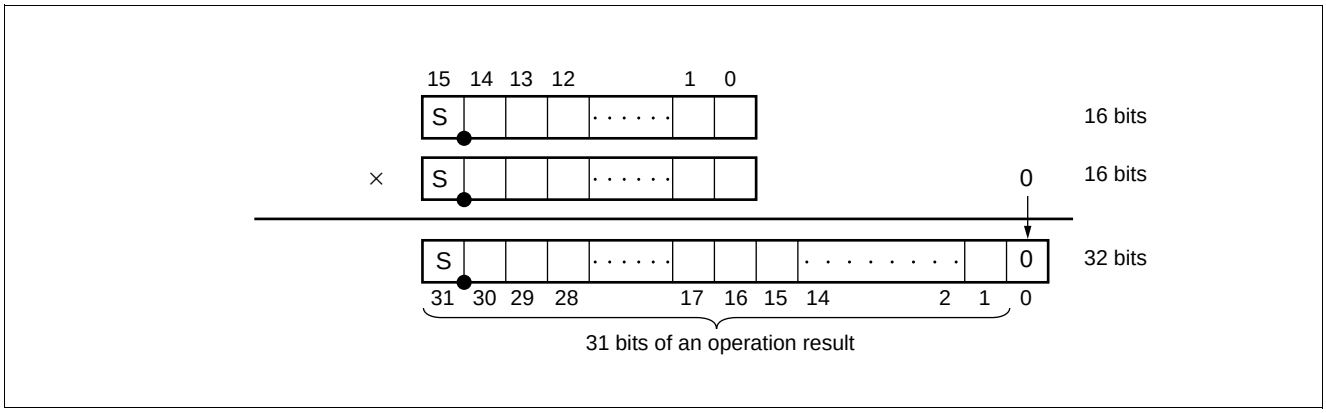
- **Fixed-point type multiplication (unsigned data × unsigned data)**
  32 bits of the result of an unsigned operation are stored into bits 1 to 32 in the accumulator shifted to the left by one bit. “0” is set to bit 0, and bits 33 to 39.



(2) NOG Mode

In this mode, the CX and DX registers are handled as a 32-bit accumulator in which detected overflow is clipped.

- **Fixed-point type multiplication (signed data x signed data)**
  31 bits of the result of a signed operation are stored into bits 1 to 31 in the accumulator shifted to the left by one bit. “0” is set at bit 0.



■ INTERRUPT

A software interrupt and a hardware interrupt are available. Interrupts are assigned specified types of priority. When interrupts occur concurrently, an interrupt with higher priority is executed earlier.

Priority	Soft/hard	Interrupt type	Interrupt branch destination	Cause of an interrupt
1	Hard	RST	0xFFFE	Hardware reset External reset signal input
2	Hard	BREAK	0x0002*	Setup of emulator operation mode External break signal input
	Soft	TRAP		Setup of emulator operation mode Software (TRAP instruction)
3	Soft	V1	0xFFFC	Occurrence of arithmetic operation overflow (V1)
4		V3	0xFFFA	Occurrence of arithmetic operation overflow (V3)
5	Hard	DMA0	—	DMA0 signal input (for data input)
6		DMA1	—	DMA1 signal input (for data input)
7		DMA2	—	DMA2 signal input (for data output)
8		DMA3	—	DMA3 signal input (for data output)
9		INT0	0xFFF8	SDOME interrupt signal input
10		INT1	0xFFF6	External interrupt signal input (INT1)
11		INT2	0xFFF4	External interrupt signal input (INT2)
12		INT3	0xFFF2	External interrupt signal input (INT3)
13		INT4	0xFFF0	External interrupt signal input (INT4)
14		INT5	0xFFEE	External interrupt signal input (INT5)
15		INT6	0xFFEC	External interrupt signal input (INT6)
16		INT7	0xFFEA	External interrupt signal input (INT7)

* : Memory space for a debug instruction

Notes:
- Emulator operation mode is set by a hardware interrupt resulting from external break signal input and by a software interrupt by the TRAP instruction.
- Any interrupts other than a reset are disabled during downloading.

■ INSTRUCTIONS

Mnemonic	Operation overview		Flag change					
			N	Z	C	V1	V2	V3
ABS	Absolute value calculation	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	0	↕	–	↕	↑	↕
ADD	Addition	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
ADD	Addition with transfer	Acc ← S1 + A1 reg ↔ (AD1)	↕	↕	↕	↕	↑	↕
ADDC	Addition with carry	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
ADSB	Addition and subtraction	A0, A1, B0, B1	↕	↕	↕	↕	↑	–
ADX	Address register addition	Xn ← Xn + Immediate value	–	–	–	–	–	–
AND	Logical AND	A0, A1, B0, B1, C0, C1, D0, D1	↕	↕	–	–	–	–
	Logical AND	ST	↓	↓	↓	↓	↓	↓
ASL	MSB □□ ···· □□ ← 0 C ↵	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
ASL	MSB □□ ···· □□ ← 0 C ↵ Shift with transfer	CX, DX reg ↔ (AD1)	↕	↕	↕	↕	↑	↕
ASR	MSB ← □□ ···· □□ → C	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
BRcc	Conditional relative branch		–	–	–	–	–	–
CALL	Subroutine jump		–	–	–	–	–	–
CMGT	Transfer with (major) comparison conditions	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
CMLT	Transfer with (minor) comparison conditions	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
CMP	Comparison	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
DO	Block repetition	RPC/imm 10	–	–	–	–	–	–
DSTP	Division support	AX, BX, CX, DX/A0, A1, B0, B1, C0, C1, D0, D1	↕	↕	–	↕	↕	↕
JPC0	Conditional absolute branch		–	–	–	–	–	–
JPC1	Conditional absolute branch		–	–	–	–	–	–
JUMP	Absolute branch		–	–	–	–	–	–
LSL	C ← □□ ···· □□ ← 0	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	–	–	–
LSR	0 → □□ ···· □□ → C	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	–	–	–

(Continued)

Mnemonic	Operation overview		Flag change					
			N	Z	C	V1	V2	V3
MOV	Data transfer or duplicate transfer	Inter-register transfer REG ↔ REG Transfer between a memory and a register MEM ↔ REG Duplicate transfer B0 ← (AD0), A1 ← (AD1) B0 ← (AD0), B1 ← (AD1) Immediate value transfer to a register REG ← Immediate value Inter-memory transfer (AD0) ← (AD1)	–	–	–	–	–	–
MOVT	Duplicate transfer to the accumulator	CX, DX ← (AD1)	–	–	–	–	–	–
MRD	Signed product addition	A0, A1,B0, B1/CX, DX	↕	↕	0	↕	↑	↕
MRD	Signed product addition with duplicate transfer	acc ← acc – A0 × A1 A0 ← (AD0), A1 ← (AD1) acc ← acc – B0 × B1 B0 ← (AD0), B1 ← (AD1)	↕	↕	0	↕	↑	↕
MRDS	Signed and unsigned product addition	A0, A1, B0, B1/CX, DX	↕	↕	0	↕	↑	–
MRDU	Unsigned and unsigned product addition	A0, A1, B0, B1/CX, DX	↕	↕	0	↕	↑	–
MSM	Signed product addition	A0, A1, B0, B1/CX, DX	↕	↕	0	↕	↑	↕
MSM	Signed product addition with duplicate transfer	acc ← acc + A0 × A1 A0 ← (AD0), A1 ← (AD1) acc ← acc + B0 × B1 B0 ← (AD0), B1 ← (AD1)	↕	↕	0	↕	↑	↕
MSM	Signed product addition with transfer	acc ← acc + (A0 or A1) × A1 reg ↔ (AD1)	↕	↕	0	↕	↑	↕
MSMS	Signed and unsigned product addition	A0, A1, B0, B1/CX, DX	↕	↕	0	↕	↑	–
MSMS	Signed and unsigned product addition with duplicate transfer	acc ← acc + A0 × A1 A0 ← (AD0), A1 ← (AD1) acc ← acc + B0 × B1 B0 ← (AD0), B1 ← (AD1)	↕	↕	0	↕	↑	–
MSMU	Unsigned and unsigned product addition	A0, A1, B0, B1/CX, DX	↕	↕	0	↕	↑	–
MSMU	Unsigned and unsigned product addition with duplicate transfer	acc ← acc + A0 × A1 A0 ← (AD0), A1 ← (AD1) acc ← acc + B0 × B1 B0 ← (AD0), B1 ← (AD1)	↕	↕	0	↕	↑	–

(Continued)

Mnemonic	Operation overview		Flag change					
			N	Z	C	V1	V2	V3
MUL	Signed multiplication	A0, A1, B0, B1, C0, C1, D0, D1/CX, DX	↕	↕	–	–	–	↕
MUL	Signed multiplication with duplicate transfer	acc ← A0 × A1 A0 ← (AD0), A1 ← (AD1) acc ← B0 × B1 B0 ← (AD0), B1 ← (AD1)	↕	↕	–	–	–	↕
MUL	Signed multiplication with transfer	acc ← (A0 or A1) × A1 reg ↔ (AD1)	↕	↕	–	–	–	↕
MULS	Signed and unsigned multiplication	A0, A1, B0, B1, C0, C1, D0, D1/CX, DX	↕	↕	–	–	–	–
MULS	Signed and unsigned multiplication with duplicate transfer	acc ← A0 × A1 A0 ← (AD0), A1 ← (AD1) acc ← B0 × B1 B0 ← (AD0), B1 ← (AD1)	↕	↕	–	–	–	–
MULU	Signed and unsigned multiplication	A0, A1, B0, B1, C0, C1, D0, D1/CX, DX	↕	↕	–	–	–	–
MULU	Signed and unsigned multiplication with duplicate transfer	acc ← A0 × A1 A0 ← (AD0), A1 ← (AD1) acc ← B0 × B1 B0 ← (AD0), B1 ← (AD1)	↕	↕	–	–	–	–
MV$_{CC}$	Conditional transfer	Inter-register transfer REG ↔ REG	–	–	–	–	–	–
NOT	Logical NOT	A0, A1, B0, B1, C0, C1, D0, D1	↕	↕	–	–	–	–
NEG	2's complement operation	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
NOP	None executed		–	–	–	–	–	–
OR	Logical OR	A0, A1, B0, B1, C0, C1, D0, D1	↕	↕	–	–	–	–
		ST	↑	↑	↑	↑	↑	↑
POP	Return of a register from the stack		–	–	–	–	–	–
PUSH	Saving of a register to the stack		–	–	–	–	–	–
REGU	Auxiliary normalization operation	CX, DX	–	↑	–	–	–	–
REP	Repeated execution of the subsequent instruction	RPC/imm 10	–	–	–	–	–	–
REP2	Repeated execution of the subsequent instruction	RPC2/imm 10	–	–	–	–	–	–
RET	Return from a subroutine		–	–	–	–	–	–
RET1	Return from an interrupt routine		–	–	–	–	–	–

(Continued)

(Continued)

Mnemonic	Operation overview		Flag change					
			N	Z	C	V1	V2	V3
RGLT	Auxiliary search for the minimum normalized value	CX, DX	–	↑	–	–	–	–
RVL	Reverse shift	A0, A1, B0, B1, C0, C1, D0, D1	↕	↕	↕	–	–	–
SLP	Standby status		–	–	–	–	–	–
STLD	Store and load	(AD0) ← A0, A0 ← (AD1) (AD0) ← A1, A1 ← (AD1) (AD0) ← CX, CX ← (AD1) (AD0) ← DX, DX ← (AD1)	–	–	–	–	–	–
SBAD	Subtraction and addition	A0, A1, B0, B1	↕	↕	↕	↕	↑	–
SUB	Subtraction	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
SUB	Subtraction with transfer	acc ← acc – A1 reg ← (AD1)	↕	↕	↕	↕	↑	↕
SUBC	Subtraction with carry	A0, A1, B0, B1, C0, C1, D0, D1, AX, BX, CX, DX	↕	↕	↕	↕	↑	↕
XOR	Exclusive logical OR	A0, A1, B0, B1, C0, C1, D0, D1	↕	↕	–	–	–	–

[Flag indications]

↕ : Set or reset by an operation
↓ : Not changed or reset by an operation
– : Not changed by an operation
↑ : Not changed or reset by an operation
0 : Reset by an operation

■ BOOT

1. BOOT Mode

BOOT mode supports the following functions.

- Ordinary BOOT function (Booting the instruction RAM (48 kwords)/the table RAM (16 kwords))
- Simplified BOOT function (Booting some instruction RAMs)

2. Setting BOOT Mode

Only single chip mode is available.

(1) Command Setup

Load the ERAM area ($FFE0_H$ to $FFE3_H$: PAGE1-0 = 11) with the following contents during booting.

- Address FFE0: Command contents
  0001_H to $000B_H$: Reserved
  $000C_H$: Simplified booting
  $000D_H$ to 0010_H : Reserved
  Others : Ordinary booting
- Address FFE1: Start address
  Specify the start address of a program to be booted during simplified booting.
- Address FFE2: Size
  Specify the program size of a program to be booted during simplified booting.
- Address FFE3: Execution start address
  Specify an execution start address after completion of simplified booting.

(2) Ordinary BOOT function (Command = other than 0001_H to 0010_H)

The ordinary BOOT function loads a full-word program (48 kwords) and table data (16 kwords), then moves execution to the user program (jump to address FFFE) or ICE (jump to address 0002).

(3) Simplified BOOT function (Command = $000C_H$)

The Simplified BOOT function loads and executes a program of 48 k or fewer words.
It also sets loaded start address ($FFE1_H$), the number of program words ($FFE2_H$), and execution start address ($FFE3_H$).

3. BOOT Timing

Performing BOOT processing requires satisfaction of the following operation.

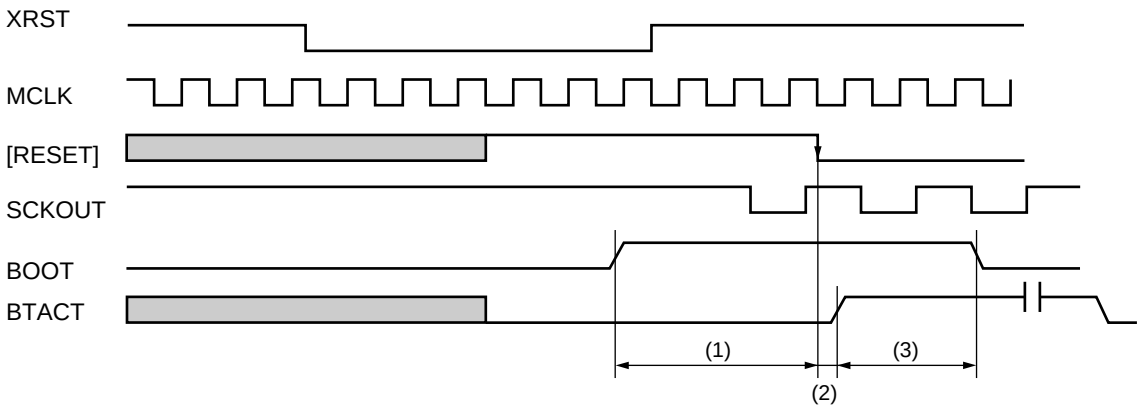
(1) Take setup of two or more MCLK clocks from a fall rising edge.

(2) Fetch information about the BOOT pin at a fall rising edge, and the BTACT pin will be set at “H”.

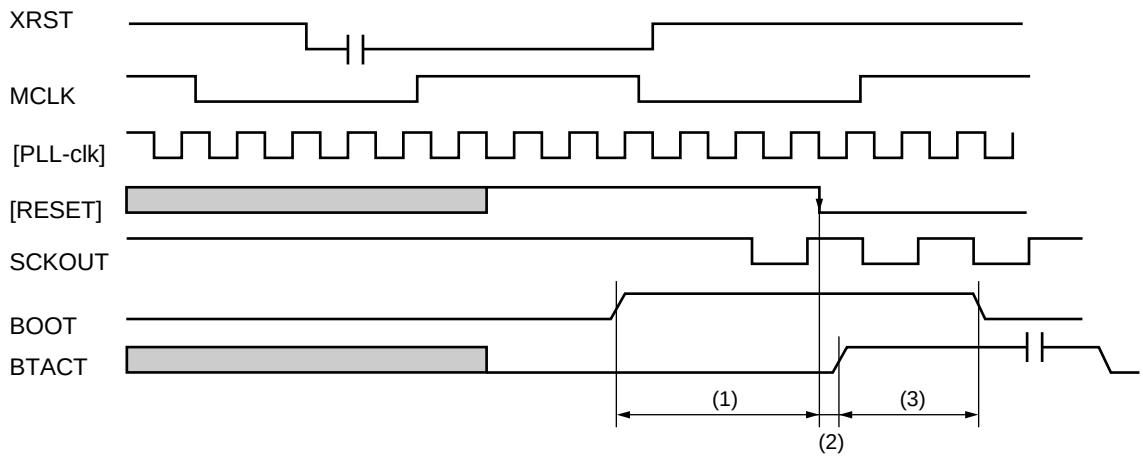
(3) Reset the BOOT pin at least two MCLK clocks after a fall edge observed at the BTACT pin.

(4) When the BTACT pin is changed from “H” to “L”, BOOT operation is terminated.

- **PM = 0 (When PLL is not used)**



- **PM = 1 (When PLL is used)**



■ PLL

1. PLL operation

Performing this DSP operation using PLL requires satisfaction of the following operation.

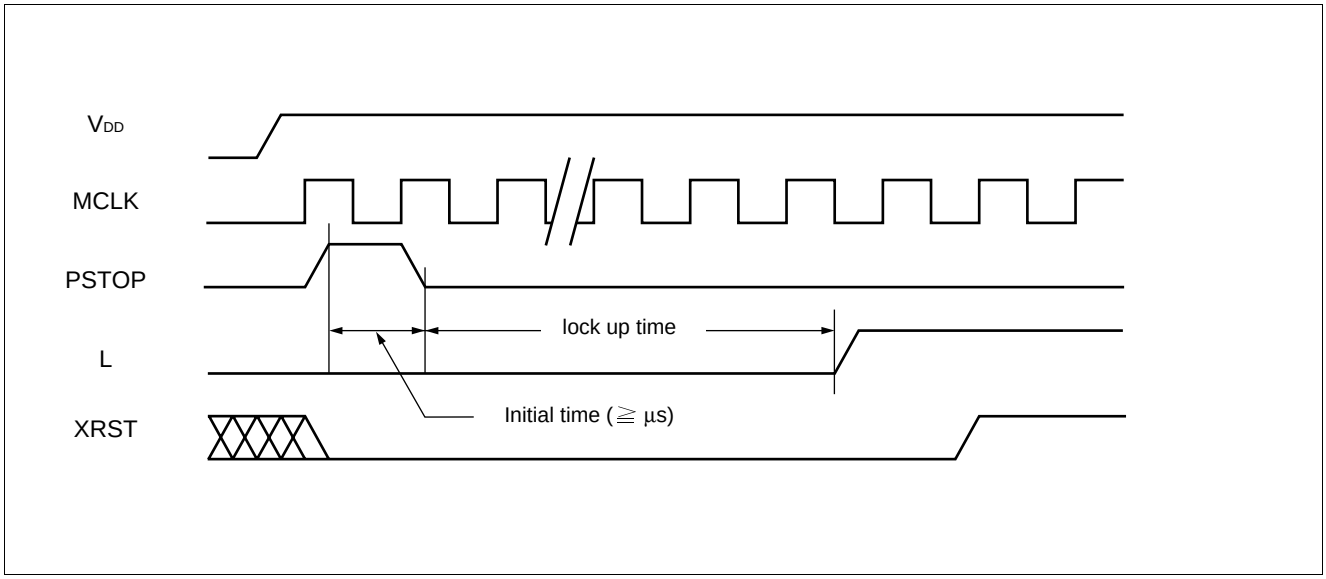
When using PLL, set the PSTOP pin at “H” for 1 µsec or more for a reset, then at “L”, and wait for lockup time or more time.

(1) Take enough time for MCLK input and for lockup at the PLL operation state with PSTOP equal to "L".

(2) Hold the DSP reset state until PLL is locked. (XRST = “L”)

(3) When PLL is locked, the “L” pin goes “H”.

(4) After PLL has been locked, change XRST from “L” to “H” to start DSP operation.



2. PLL standards

Input clock (MHz)	Look up time (µs)	Remarks
20 to 25	200	When PLL is used

■ ELECTRICAL CHARACTERISTICS

1. Absolute Maximum Ratings

(V_{SS} = 0 V)

Parameter	Symbol	Condition	Value		Unit	Remarks
			Min.	Max.		
Power supply voltage	V_{DD}	$V_{DD} - V_{SS}$	V_{SS} – 0.5	4.0	V	
Input voltage	V_I	—	V_{SS} – 0.5	V_{DD} + 0.5	V	Input pin
Output voltage	V_O	—	V_{SS} – 0.5	V_{DD} + 0.5	V	Output pin BUS pin
Maximum output current	I_O	$V_O = V_{DD}$	—	14	mA	Output drive pin (I_{OL}) 4 mA
	I_O	V_O = 0 V	—	–14	mA	
Storage temperature	Tstg	—	–65	+150	°C	Ceramic package

WARNING: Semiconductor devices can be permanently damaged by application of stress (voltage, current, temperature, etc.) in excess of absolute maximum ratings. Do not exceed these ratings.

2. Recommended Operating Conditions

(V_{SS} = 0 V)

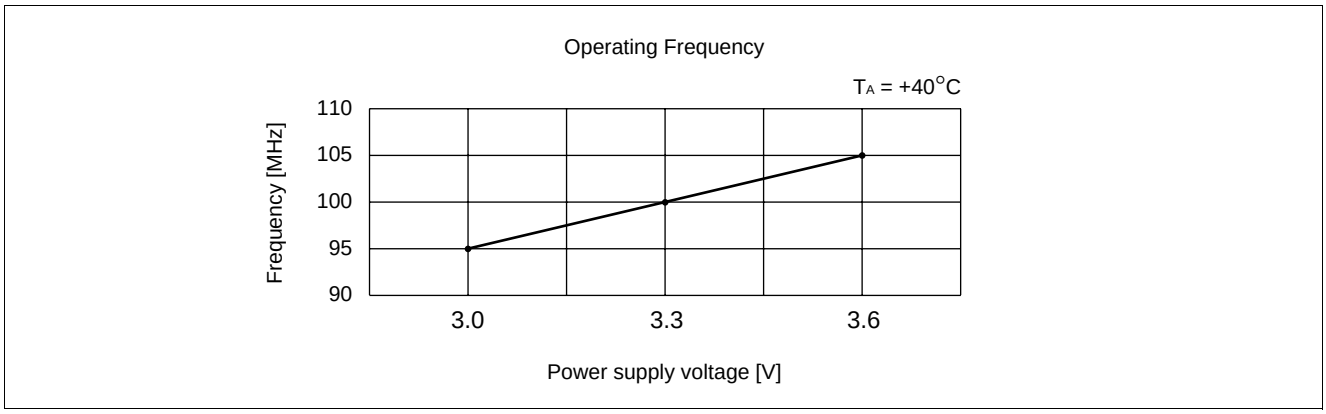
Parameter	Symbol	Value			Unit	Remarks
		Min.	Typ.	Max.		
Power supply voltage	V_{DD}	3.0	3.3	3.6	V	V_{DD}
“H” level input voltage	V_{IH}	0.65 V_{DD}	—	V_{DD} + 0.3	V	
“L” level output voltage	V_{IL}	V_{SS}	—	0.25 V_{DD}	V	
Ambient temperature	T_A	0	—	+40	°C	

WARNING: The recommended operating conditions are required in order to ensure the normal operation of the semiconductor device. All of the device's electrical characteristics are warranted when the device is operated within these ranges.

Always use semiconductor devices within their recommended operating condition ranges. Operation outside these ranges may adversely affect reliability and could result in device failure.

No warranty is made with respect to uses, operating conditions, or combinations not represented on the data sheet. Users considering application outside the listed conditions are advised to contact their FUJITSU representatives beforehand.

3. Operating Frequency



4. DC Characteristics

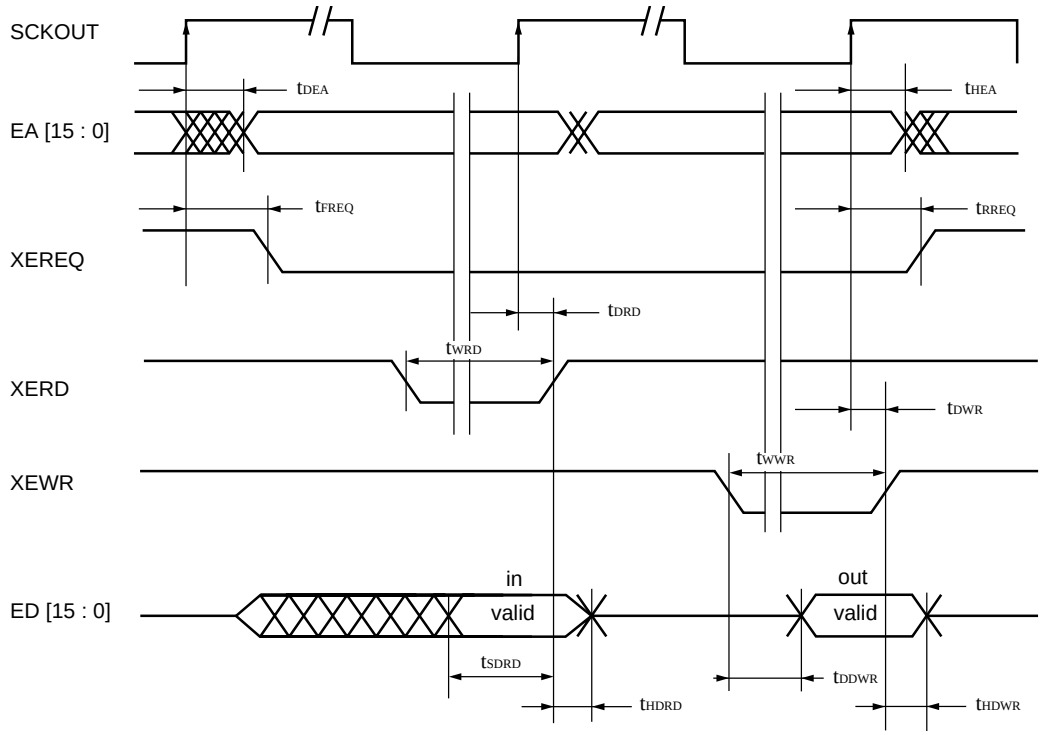
(V_{DD} = 3.0 to 3.6 V, V_{SS} = 0 V, T_A = 0°C to +40°C)

Parameter	Symbol	Condition	Value			Unit
			Min.	Typ.	Max.	
Power supply voltage	I_{DDS}	Standby mode*1	—	50	—	µA
	I_{DD}	Ordinary operation mode	—	62	—	mA
"H" level input voltage	V_{IH}	—	0.65 V_{DD}	—	V_{DD}	V
"L" level input voltage	V_{IL}	—	V_{SS}	—	0.25 V_{SS}	V
"H" level output voltage	V_{OH}	I_{OH} = –4mA	V_{DD} – 0.5	—	V_{DD}	V
"L" level output voltage	V_{OL}	I_{OL} = 4mA	V_{SS}	—	0.4	V
Input leakage current*2 (Tri-state pin)	I_{LI}	V_I = 0 – V_{DD}	–5	—	5	µA
	I_{LZ}		–5	—	5	µA
Pull-up/pull-down resistor	RP	—	25	50	200	kΩ
Output current (Shorting circuit)	I_O*3	State	V_O = V_{DD}	—	V_O = 0 V	mA
		Normal/I_{OL} = 4 mA	+60	—	–60	mA

*1: The memory is set at the standby state with V_{IH} = V_{DD} and V_{IL} = V_{SS}.
*2: With the input pin provided with a pull-up or pull-down resistor, the standard value may be exceeded.
*3: Maximum supply current at the output section, and the V_{DD} or V_{SS} circuit

5. AC Characteristics

(1) ERAM Interface



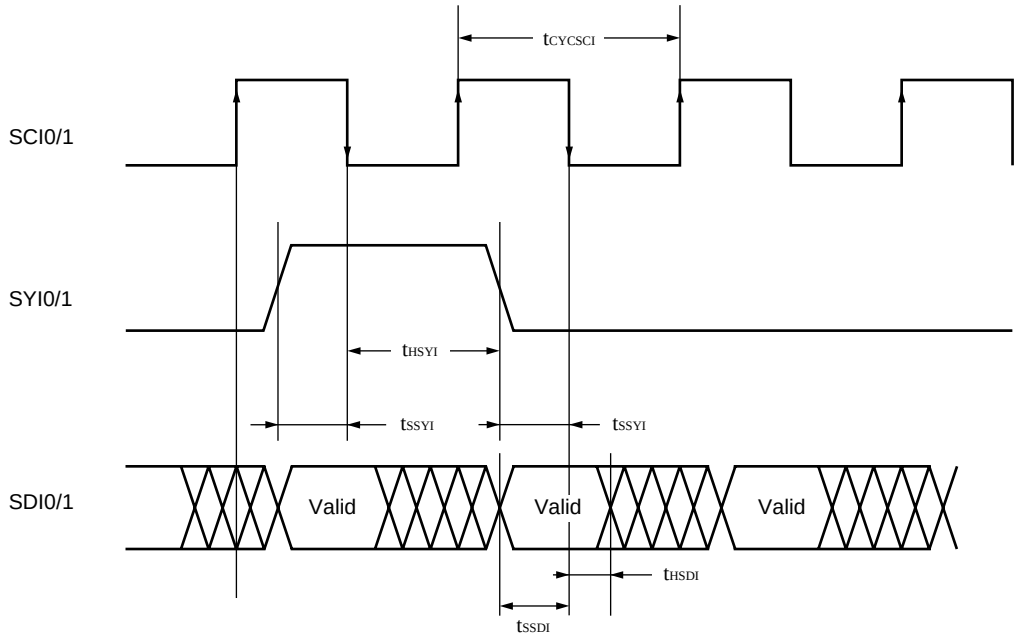
Note: During a wait, the state indicated by double lines above is held for a wait cycle.

(V_{DD} = 3.0 V to 3.6 V, T_A = 0°C to +40°C, output pin load = 50 pF)

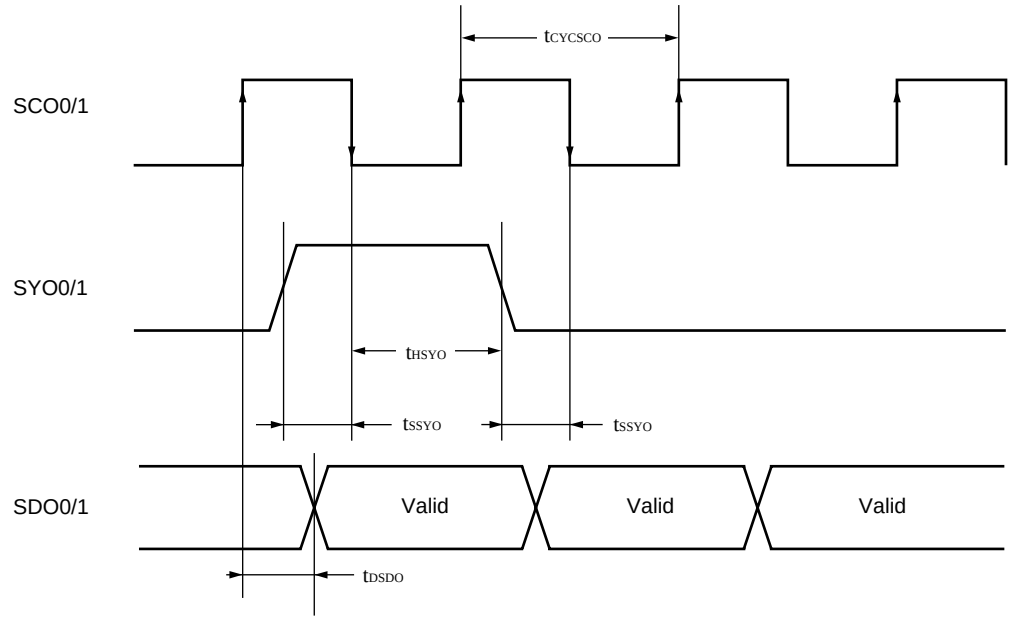
Parameter	Symbol	Pin name	Value			Unit
			Min.	Typ.	Max.	
EA output delay	t_{DEA}	EA	—	—	6.2	ns
EA hold time	t_{HEA}	EA	2.6	—	—	ns
XEREQ falling delay	t_{FREQ}	XEREQ	—	—	5.5	ns
XEREQ rising delay	t_{RREQ}	XEREQ	—	—	4.2	ns
XERD rising delay	t_{DRD}	XERD	—	—	1.0	ns
XERD “L” pulse width	t_{WRD}	XERD	5.8	—	—	ns
XEWR rising delay	t_{DWR}	XEWR	—	—	2.5	ns
XEWR “L” pulse width	t_{WWR}	XEWR	4.2	—	—	ns
ED setup time for XERD	t_{SDRD} (in)	ED	10.3	—	—	ns
ED hold time for XERD	t_{HDRD} (in)	ED	1.5	—	—	ns
ED delay time for XEWR	t_{DDWR} (out)	ED	—	—	8.5	ns
ED hold time for XEWR	t_{HDWR} (out)	ED	0.3	—	—	ns

(2) Serial I/O Interface

- **Serial input**



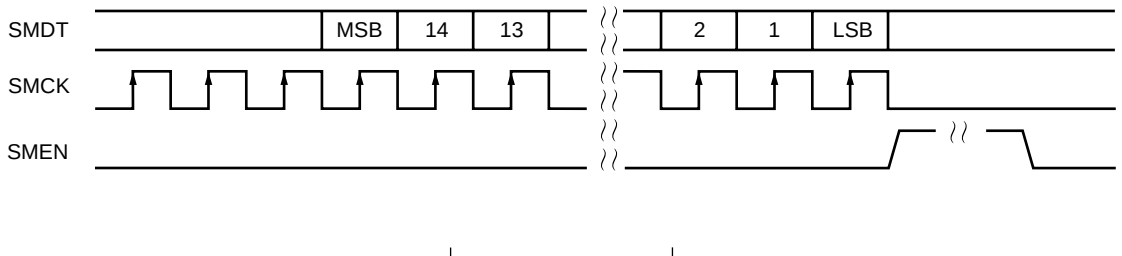
- **Serial output**

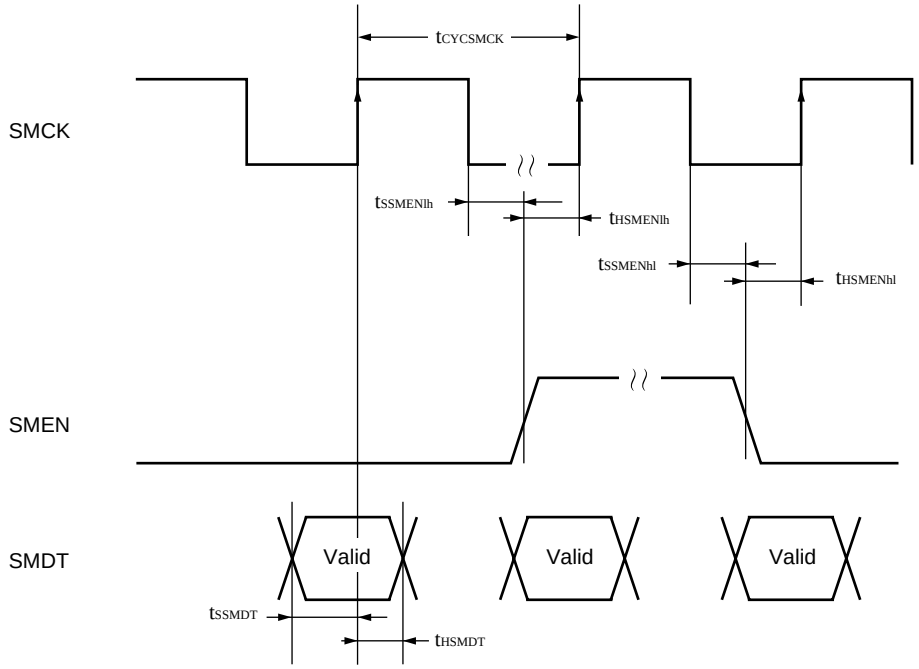


(V_{DD} = 3.0 V to 3.6 V, T_A = 0°C to +40°C, output pin load = 50 pF)

Parameter	Symbol	Pin name	Value			Unit
			Min.	Typ.	Max.	
Serial input clock cycle	t_{CYCSCI}	SCI [1 : 0]		≧ 500		ns
SYI signal setup time	t_{SSTI}	SYI [1 : 0]	2.1	—	—	ns
SYI signal hold time	t_{HSYI}	SYI [1 : 0]	1.4	—		ns
SDI signal setup time	t_{SSDI}	SDI [1 : 0]	1.5	—	—	ns
SDI signal hold time	t_{HSDI}	SDI [1 : 0]	2.0	—	—	ns
Serial output clock cycle	t_{CYCSCO}	SCO [1 : 0]		≧ 500		ns
SYO signal setup time	t_{SSYO}	SYO [1 : 0]	2.2	—	—	ns
SYO signal hold time	t_{HSYO}	SYO [1 : 0]	1.4	—	—	ns
SDO signal output delay	t_{DSDO}	SDO[1 : 0]	—	—	5.6	ns

(3) SMODE

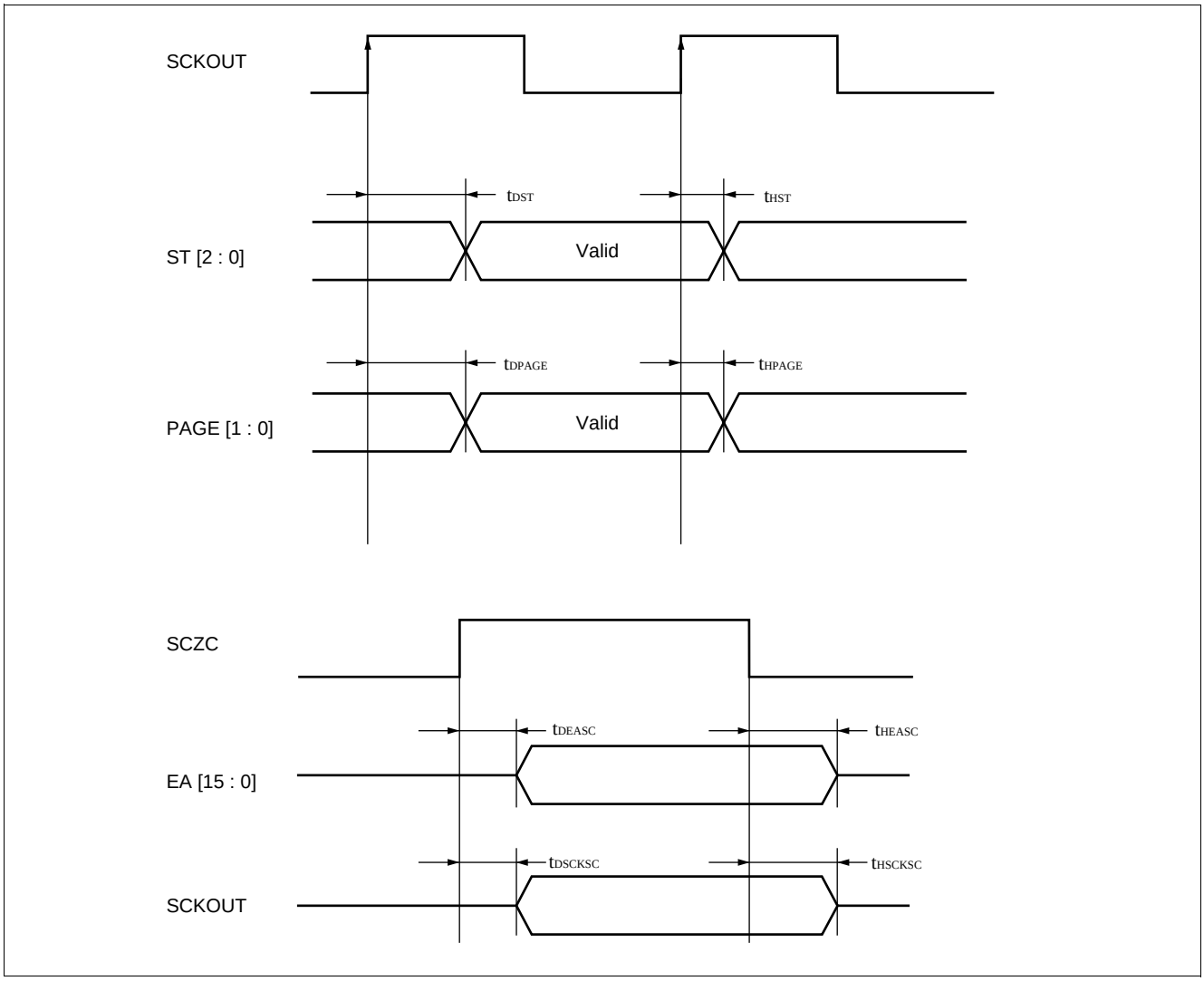




(V_{DD} = 3.0 V to 3.6 V, T_A = 0°C to +40°C, output pin load = 50 pF)

Parameter	Symbol	Pin name	Value			Unit
			Min.	Typ.	Max.	
Serial input clock cycle	$t_{CYCSMCK}$	SMCK		≧ 500		ns
SMDT SMCK setup time	t_{SSMDT}	SMDT	0.8	—	—	ns
SMDT SMCK hold time	t_{HSMDT}	SMDT	1.8	—	—	ns
SMEN SMCK setup time	$t_{SSMENhl}$	SMEN	0.6	—	—	ns
SMEN SMCK hold time	$t_{HSMENhl}$	SMEN	0.6	—	—	ns
SMEN SMCK setup time	$t_{SSMENlh}$	SMEN	0.6	—	—	ns
SMEN SMCK hold time	$t_{HSMENlh}$	SMEN	0.8	—	—	ns

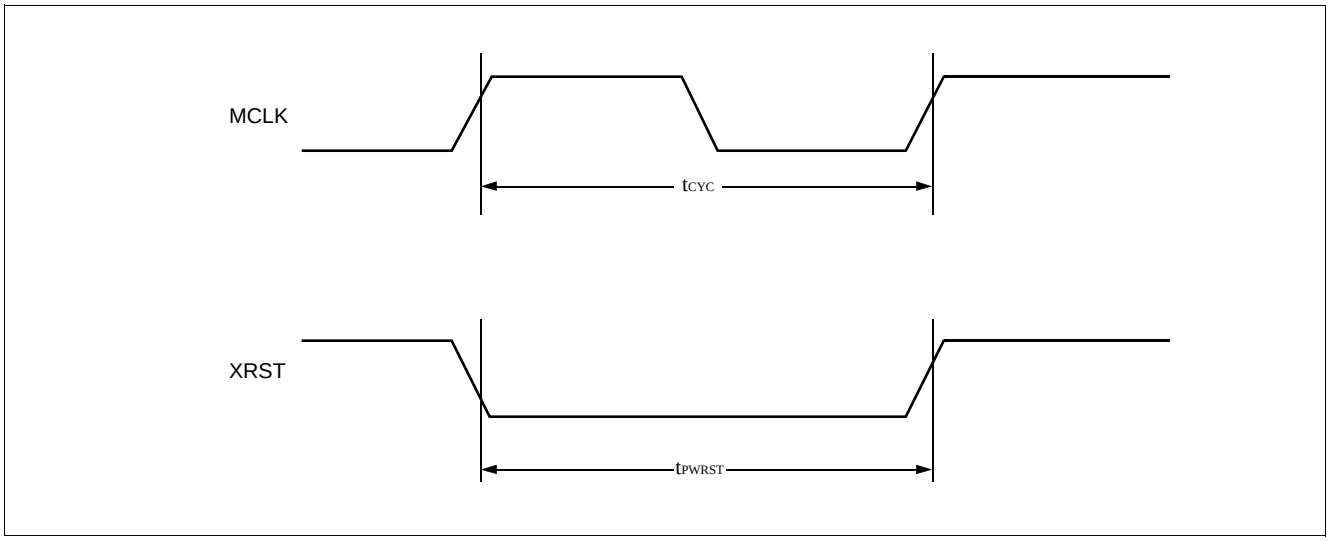
(4) PLL and Others



(V_{DD} = 3.0 V to 3.6 V, T_A = 0°C to +40°C)

Parameter	Symbol	Pin name	Value			Unit
			Min.	Typ.	Max.	
ST output delay	t_{DST}	ST [2 : 0]	—	—	3.3	ns
ST hold time	t_{HST}	ST [2 : 0]	0.8	—	—	ns
PAGE output delay	t_{DPAGE}	PAGE [1 : 0]	—	—	5.3	ns
PAGE hold time	t_{HPAGE}	PAGE [1 : 0]	1.3	—	—	ns
EA output delay for SCZC	t_{DEASC}	EA [15 : 0]	—	—	5.4	ns
EA hold time for SCZC	t_{HEASC}	EA [15 : 0]	2.2	—	—	ns
EA output delay for SCZC	t_{DSCKSC}	SCKOUT	—	—	3.2	ns
EA hold time for SCZC	t_{HSCKSC}	SCKOUT	1.1	—	—	ns

(5) MCLK, XRST



(V_{DD} = 3.0 V to 3.6 V, T_A = 0°C to +40°C)

Parameter	Symbol	Pin name	Value			Unit	Remarks
			Min.	Typ.	Max.		
MCLK cycle (when PLL is used)	f_{CYC}	MCLK	20	—	25	MHz	*
MCLK cycle (when PLL is not used)	f_{CYC}	MCLK	—	—	160	MHz	*

* : Input the MCLK cycle value so that the MCLK duty-cycle becomes 50% ±5%.

Parameter	Symbol	Pin name	Value			Unit
			Min.	Typ.	Max.	
XRST “L” pulse width	t_{PWRST}	XRST	$10t_{CYC}$*	—	—	ns

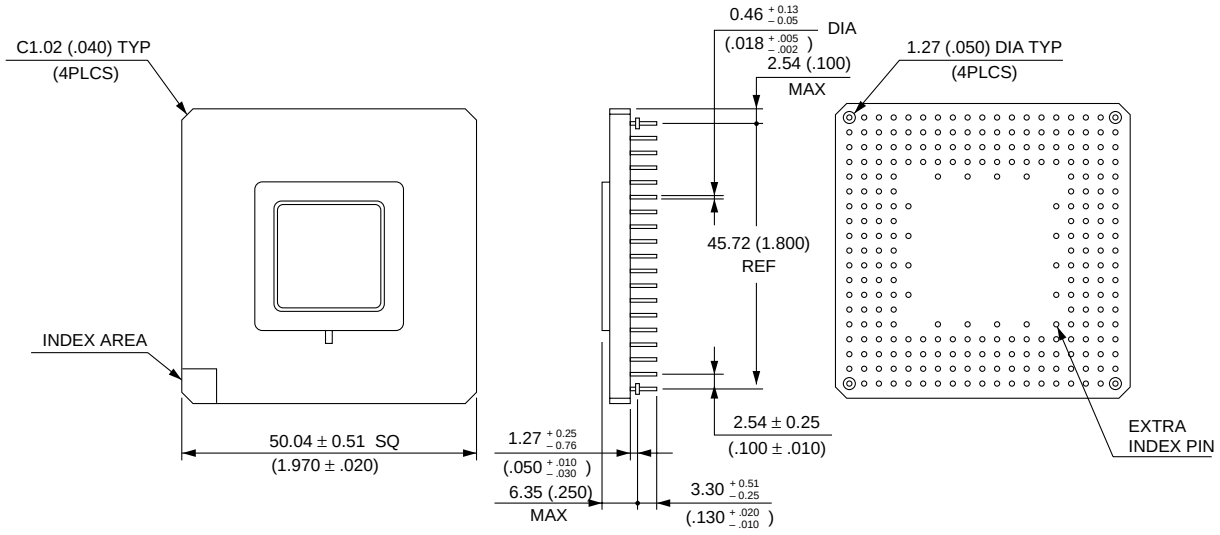
* : $t_{CYC} = 1/f_{CYC}$

■ ORDERING INFORMATION

Part number	Package	Remarks
MB86330CR-ES	256-pin Ceramic PGA (PGA-256C-A03)	

■ PACKAGE DIMENSIONS

256-pin Ceramic PGA
(PGA-256C-A03)



Dimensions in mm (inches)

FUJITSU LIMITED

For further information please contact:

Japan
FUJITSU LIMITED
Corporate Global Business Support Division
Electronic Devices
KAWASAKI PLANT, 4-1-1, Kamikodanaka
Nakahara-ku, Kawasaki-shi
Kanagawa 211-8588, Japan
Tel: 81(44) 754-3763
Fax: 81(44) 754-3329

http://www.fujitsu.co.jp/

North and South America
FUJITSU MICROELECTRONICS, INC.
Semiconductor Division
3545 North First Street
San Jose, CA 95134-1804, USA
Tel: (408) 922-9000
Fax: (408) 922-9179

Customer Response Center
Mon. - Fri.: 7 am - 5 pm (PST)
Tel: (800) 866-8608
Fax: (408) 922-9179

http://www.fujitsumicro.com/

Europe
FUJITSU MIKROELEKTRONIK GmbH
Am Siebenstein 6-10
D-63303 Dreieich-Buchschlag
Germany
Tel: (06103) 690-0
Fax: (06103) 690-122

http://www.fujitsu-ede.com/

Asia Pacific
FUJITSU MICROELECTRONICS ASIA PTE LTD
#05-08, 151 Lorong Chuan
New Tech Park
Singapore 556741
Tel: (65) 281-0770
Fax: (65) 281-0220

http://www.fmap.com.sg/

F9810

The contents of this document are subject to change without notice. Customers are advised to consult with FUJITSU sales representatives before ordering.

The information and circuit diagrams in this document are presented as examples of semiconductor device applications, and are not intended to be incorporated in devices for actual use. Also, FUJITSU is unable to assume responsibility for infringement of any patent rights or other rights of third parties arising from the use of this information or circuit diagrams.

FUJITSU semiconductor devices are intended for use in standard applications (computers, office automation and other office equipment, industrial, communications, and measurement equipment, personal or household devices, etc.).
CAUTION:
Customers considering the use of our products in special applications where failure or abnormal operation may directly affect human lives or cause physical injury or property damage, or where extremely high levels of reliability are demanded (such as aerospace systems, atomic energy controls, sea floor repeaters, vehicle operating controls, medical devices for life support, etc.) are requested to consult with FUJITSU sales representatives before such use. The company will not be responsible for damages arising from such use without prior approval.

Any semiconductor devices have an inherent chance of failure. You must protect against injury, damage or loss from such failures by incorporating safety design measures into your facility and equipment such as redundancy, fire protection, and prevention of over-current levels and other abnormal operating conditions.

If any products described in this document represent goods or technologies subject to certain restrictions on export under the Foreign Exchange and Foreign Trade Law of Japan, the prior authorization by Japanese government will be required for export of those products from Japan.