NEC

MOS INTEGRATED CIRCUIT

μPD705100

V830™
32-BIT MICROCONTROLLER

The μPD705100 (also called V830) is a microcontroller for incorporation use, which belongs to the V830 family™ of the NEC original V800 series™ microcontrollers. The V830 can achieve high cost-performance for multimedia equipment, by integrating quick real-time responses, high-speed arithmetic/logical instructions, and functions suitable for individual applications.

The following user's manual describes details of the functions of the V830. Be sure to read it before designing an application system.

V830 User's Manual, Hardware : U10064E
V830 User's Manual, Architecture : U12496E

FEATURES

- High-performance 32-bit architecture for incorporation use
  - Built-in cache memory
    Instruction cache : 4K bytes
    Data cache : 4K bytes
  - Built-in RAM
    Instruction RAM : 4K bytes
    Data RAM : 4K bytes
  - One-clock-pitch pipeline structure
  - 16-/32-bit instructions
  - Separate buses for addresses and data
  - 4G-byte linear addresses
  - Thirty-two 32-bit general-purpose registers
  - Hardware-interlocked register/flag hazard
  - 16-level interrupt responses
- 16-bit bus fixing function
  - 16-bit bus system construction
- Instructions suitable for variable application
  - Sum-of-products operation
  - Saturable operation
  - Branch prediction
  - Concatenation shift
  - Block transfer instructions
- Power-saving mode
- Maximum operating frequency
  - 100 MHz (internal)
  - 50/33 MHz (external)
- CMOS operation, 3.3-V operation

ORDERING INFORMATION

Part number	Package
μPD705100GJ-100-8EU	144-pin plastic LQFP (fine pitch) (20 × 20 mm)

The information in this document is subject to change without notice.

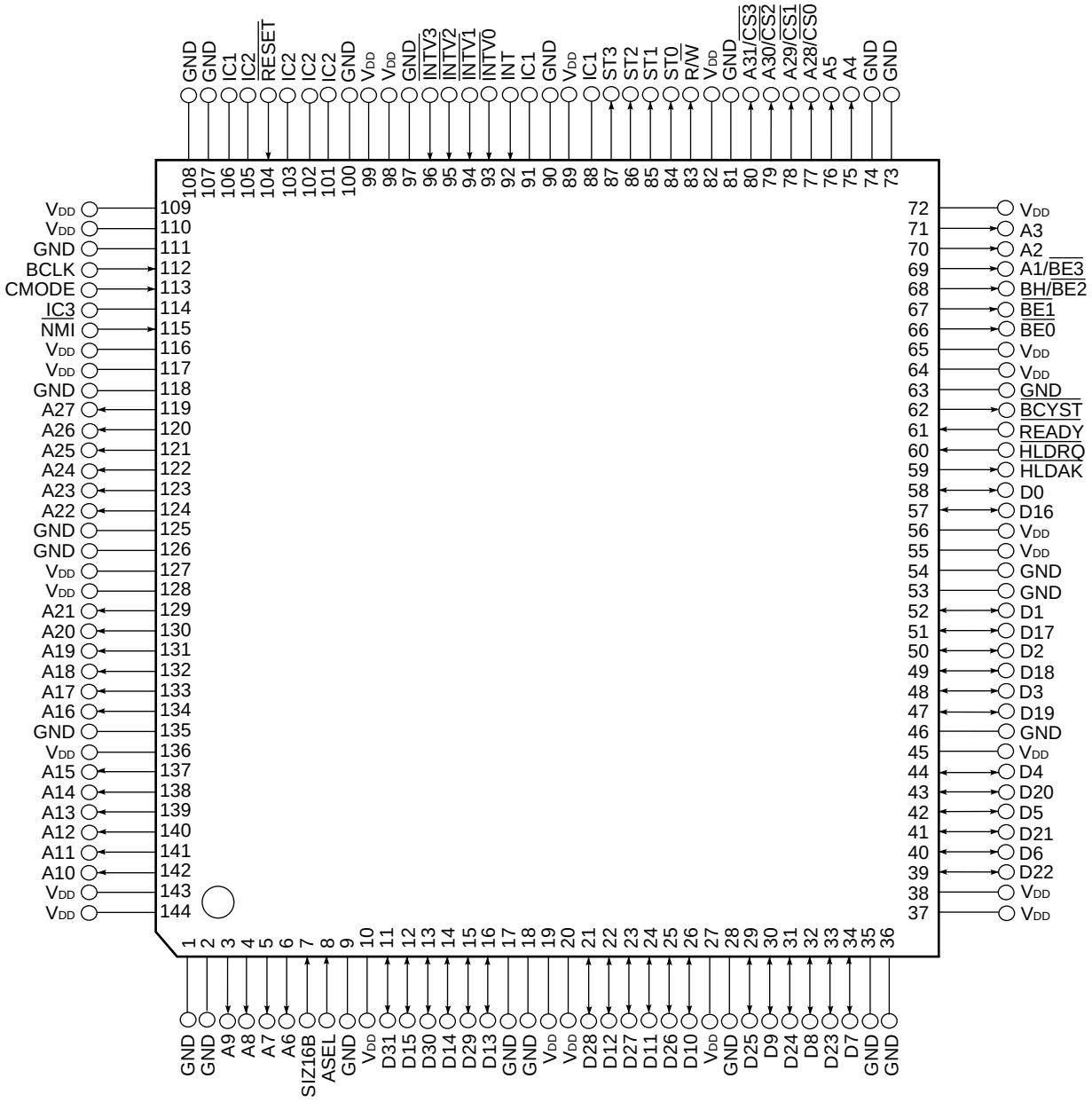
Document No. U11483EJ3V0DS00 (3rd edition)
Date Published January 1998 J CP(K)
Printed in Japan

The mark ★ shows major revised points.

PIN CONFIGURATION

144-pin plastic LQFP (fine pitch) (20 × 20 mm)

μPD705100GJ-100-8EU



Caution Leave the IC1 pins open.

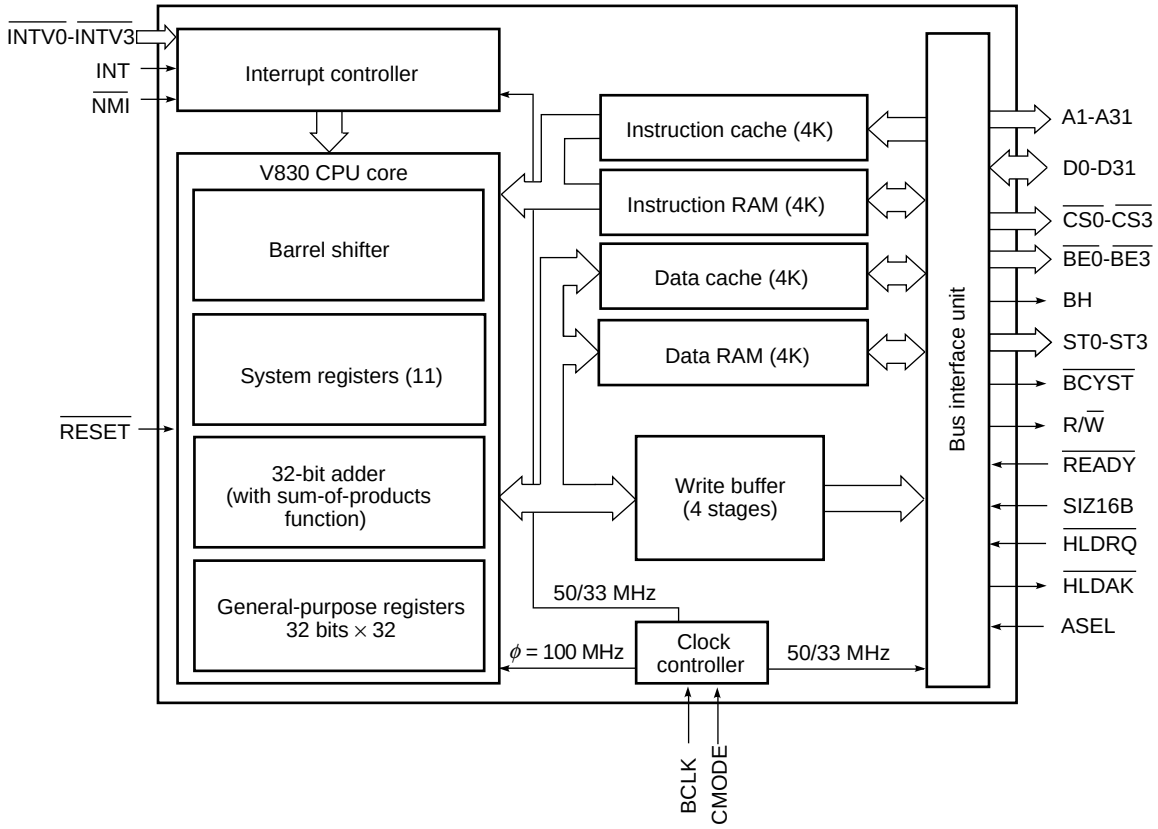
Connect each IC2 pin to GND via a dedicated resistor.

Connect each IC3 pin to V_{DD} via a dedicated resistor.

PIN NAMES

A1-A31	:	Address Bus
$\overline{CS0}$-$\overline{CS3}$	:	Chip Select
D0-D31	:	Data Bus
$\overline{BE0}$-$\overline{BE3}$	:	Byte Enable
BH	:	Byte or Halfword
ST0-ST3	:	Status
$\overline{BCYST}$	:	Bus Cycle Start
R/$\overline{W}$	:	Read/Write
$\overline{READY}$	:	Ready
$\overline{HLDRQ}$	:	Hold Request
$\overline{HLDAK}$	:	Hold Acknowledge
SIZ16B	:	Bus Size 16 bit
$\overline{NMI}$	:	Non-Maskable Interrupt Request
INT	:	Interrupt Request
$\overline{INTV0}$-$\overline{INTV3}$	:	Interrupt Level
BCLK	:	Bus Clock
CMODE	:	Clock Mode
ASEL	:	Address Select
$\overline{RESET}$	:	Reset
V_{DD}	:	Power Supply
GND	:	Ground
IC1-IC3	:	Internally Connected

BLOCK DIAGRAM



CONTENTS

1. PIN FUNCTIONS

1.1 Pin Functions

Pin name	Input/output	Function	At hold	At reset
A2-A27	Tristate output	Address bus	Hi-Z	H
A28-A31/$\overline{CS0}$-$\overline{CS3}$**Note**		Address bus/chip select	Hi-Z/H	H
D0-D31	Tristate input/output	Bidirectional data bus	Hi-Z	Hi-Z
$\overline{BE0}$, $\overline{BE1}$	Tristate output	Indicates which data bus can be used for data access.	Hi-Z	H
$\overline{BE2}$/BH		Indicates access to D16-D23/byte or halfword access.	Hi-Z	H
$\overline{BE3}$/A1		Indicates most significant byte access/A1 address.	Hi-Z	H
ST0-ST3		Indicates the status of a bus.	Hi-Z	0101
$\overline{BCYST}$		Indicates the start of a bus cycle.	Hi-Z	H
R/$\overline{W}$		Indicates whether the bus cycle is a read or write cycle.	Hi-Z	H
$\overline{READY}$	Input	Terminates a bus cycle.	-	-
$\overline{HLDRQ}$		Requests bus mastership.	-	-
$\overline{HLDAK}$	Output	Response to $\overline{HLDRQ}$	L	H
SIZ16B	Input	Fixes the bus width to 16 bits.	-	-
$\overline{NMI}$		Nonmaskable interrupt request	-	-
INT		Maskable interrupt request	-	-
$\overline{INTV0}$-$\overline{INTV3}$		Indicates an interrupt level.	-	-
BCLK		Bus clock input	-	-
CMODE		Specifies the frequency ratio for the external bus and the internal circuit.	-	-
ASEL		Selects A28-A31/$\overline{CS0}$-$\overline{CS3}$.	-	-
$\overline{RESET}$		Resets the internal state.	-	-
V_{DD}	-	Supplies positive power.	-	-
GND		Ground potential	-	-

Note When used for a chip select signal, this is held at the high level.

2. ADDRESS SPACE

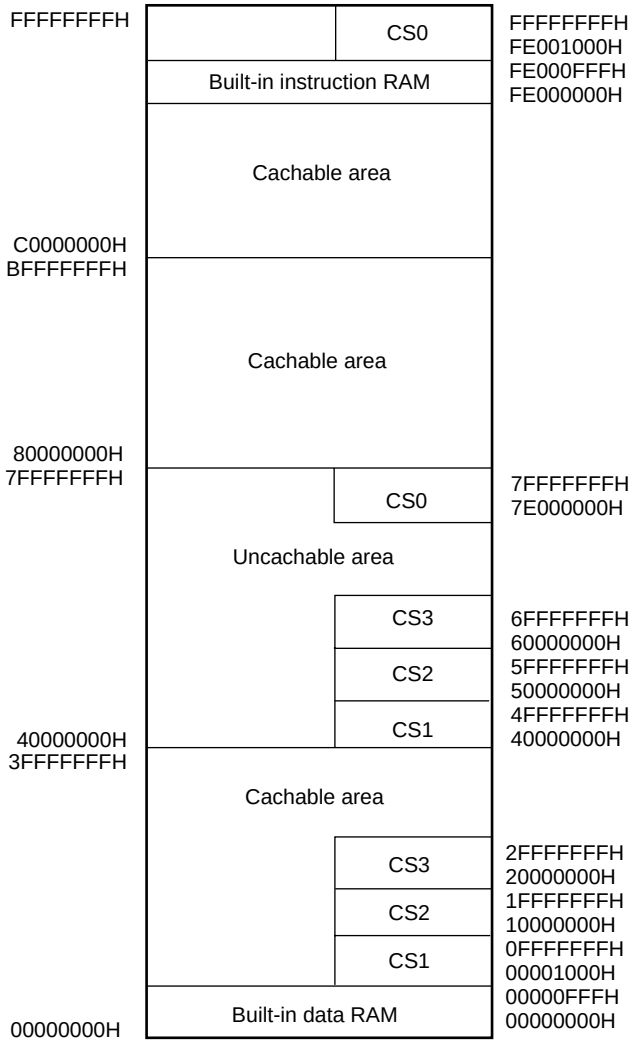
2.1 Memory Space

The V830 uses four chip select/address pins and 26 address bus pins to represent a 32-bit address. When the chip select function is used, a 256M-byte image space is created as three spaces and a 32M-byte image space is created as one space. When the chip select function is not used, a 4G-byte linear address space is created.

Area 40000000H-7FFFFFFFH in the memory space is reserved as an uncachable area. When this area is accessed, the cache function is not effective. For all other areas, the cache function is effective.

Within the memory space, built-in instruction RAM and built-in data RAM are mapped. By accessing these areas, an instruction can be fetched and data loaded/stored within one cycle (internal clock) without activating a bus cycle externally. Data in the built-in instruction RAM, however, cannot be accessed by using the load/store instructions. Nor can instructions be fetched from the built-in data RAM. These built-in RAMs are mapped to the cachable area; however, they are not cached.

Figure 2-1. Memory Map



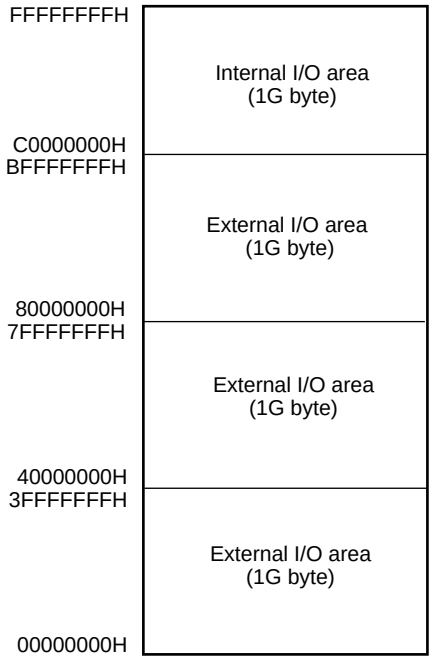
Chip select signal	Address space	Cachable
$\overline{CS0}$	7E000000H-7FFFFFFFH	×
	FE001000H-FFFFFFFFH	○
$\overline{CS1}$	40000000H-4FFFFFFFH	×
	00001000H-0FFFFFFFH	○
$\overline{CS2}$	50000000H-5FFFFFFFH	×
	10000000H-1FFFFFFFH	○
$\overline{CS3}$	60000000H-6FFFFFFFH	×
	20000000H-2FFFFFFFH	○

2.2 I/O Space

The V830 represents the I/O space using 32 bits and supports a linear address space of up to 4G bytes.

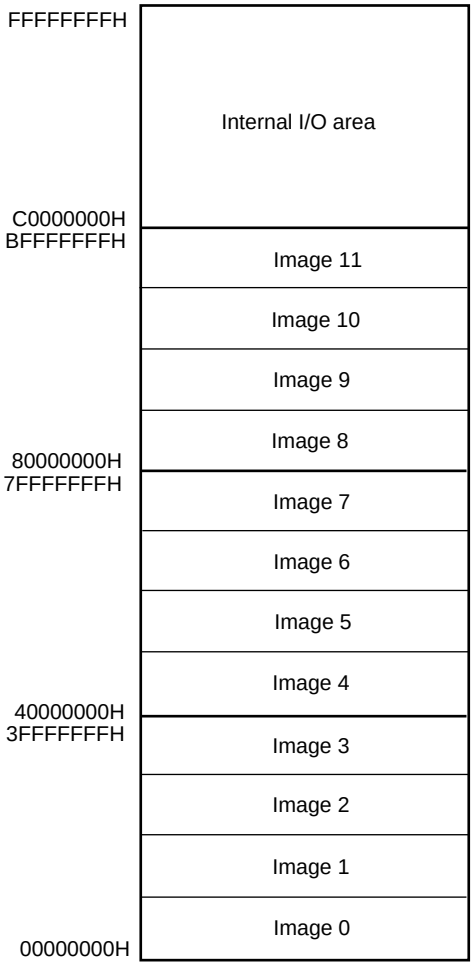
The 1G-byte area C0000000H-FFFFFFFFH is reserved as an internal I/O area. External I/O cannot be placed in this area. When accessing that part of the internal I/O area to which internal I/O is not allocated, normal operation cannot be guaranteed.

Figure 2-2. I/O Map



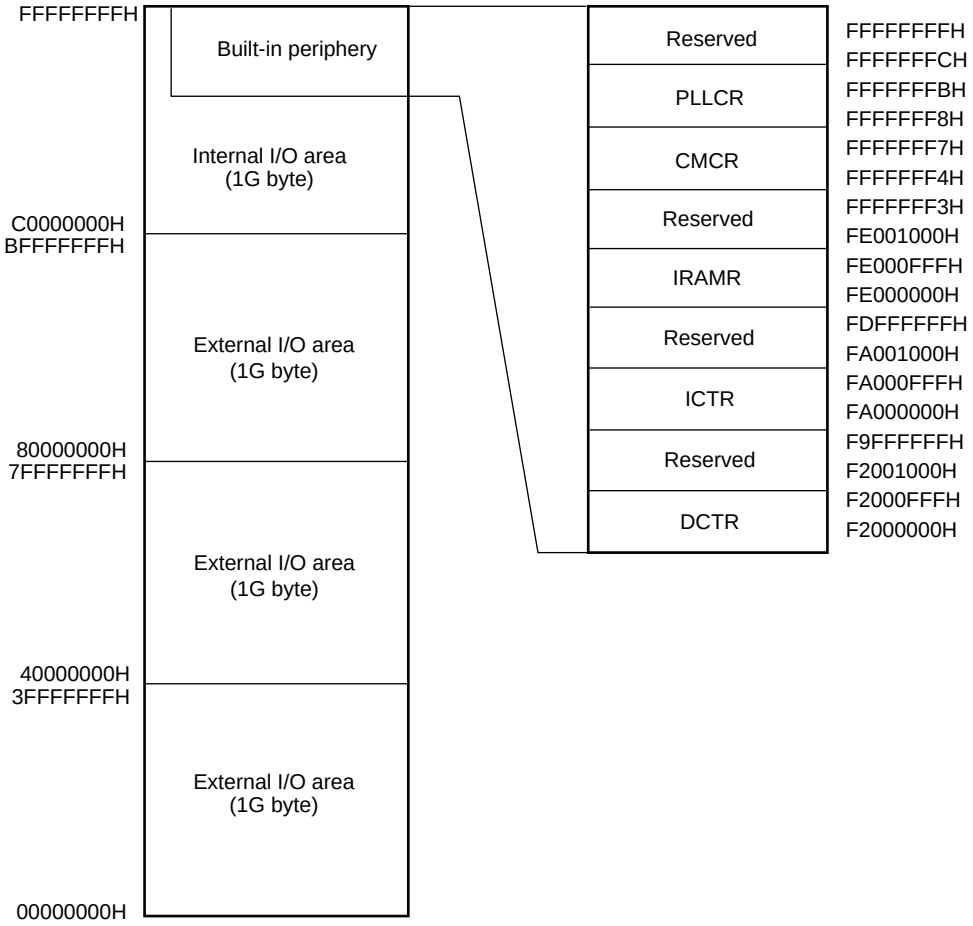
The cache function is not effective within the I/O space. When the chip select function is used, the area is used as the 256M-byte image space represented by A2-A27.

Figure 2-3. Image Space Used When Chip Select Function is Used



The upper 1G-byte area (C0000000H-FFFFFFFFH) in the I/O space is reserved for internal I/O. To access internal I/O, the IN.W/OUT.W instructions (in words) must be used. When the internal I/O area is accessed, an external bus cycle is not activated.

Figure 2-4. Internal I/O Area



3. 32-BIT BUS MODE

If the SIZ16B input, sampled at reset, is inactive, the external bus width becomes 32 bits (32-bit bus mode). In this mode, $\overline{BE2}$/BH acts as $\overline{BE2}$ and $\overline{BE3}$/A1 acts as $\overline{BE3}$.

3.1 Relationship between External Accesses and Byte Enable Signals

In 32-bit bus mode, $\overline{BE0}$-$\overline{BE3}$ are output. External accesses are related to byte enable signals as indicated below.

Table 3-1. 32-Bit Bus Mode

Data length	Operand address		Byte enable				State
	Bit 1	Bit 0	$\overline{BE3}$	$\overline{BE2}$	$\overline{BE1}$	$\overline{BE0}$	
Byte	0	0	1	1	1	0	Ta,Ts
	0	1	1	1	0	1	Ta,Ts
	1	0	1	0	1	1	Ta,Ts
	1	1	0	1	1	1	Ta,Ts
Halfword	0	0	1	1	0	0	Ta,Ts
	1	0	0	0	1	1	Ta,Ts
Word	0	0	0	0	0	0	Ta,Ts
Burst transfer	0	0	0	0	0	0	Ta,Tb1
	0	0	0	0	0	0	Tb2
	0	0	0	0	0	0	Tb3
	0	0	0	0	0	0	Tb4

4. 16-BIT BUS MODE

If the SIZ16B input, sampled at reset, is active, the external bus width becomes 16 bits (16-bit bus mode). In this mode, the low-order 16 bits (D0-D15) of the data bus are valid, $\overline{BE2}$/BH acts as BH and $\overline{BE3}$/A1 acts as A1. The high-order 16 bits (D16-D31) of the data bus enter the high-impedance state.

4.1 16-Bit Bus Sizing

The V830 has a bus sizing function by which, to enable access from the data bus to 16 bits of memory or the I/O space, data can be transferred using only the low-order 16 bits of the 32-bit data bus.

When the SIZ16B input is activated upon a reset, the external data bus width becomes 16 bits (16-bit bus mode). In 16-bit bus mode, D16-D31 are all set to the high-impedance state and $\overline{BE0}$, $\overline{BE1}$, BH, and A1 are output in a way suited to a 16-bit bus system. Connection to D16-D31 is not necessary. The SIZ16B input can be changed only when the V830 is reset. It cannot be changed at any other time.

4.1.1 Byte/halfword access

Bus cycles in either of two bus states (Ta and Ts) are used for byte/halfword access.

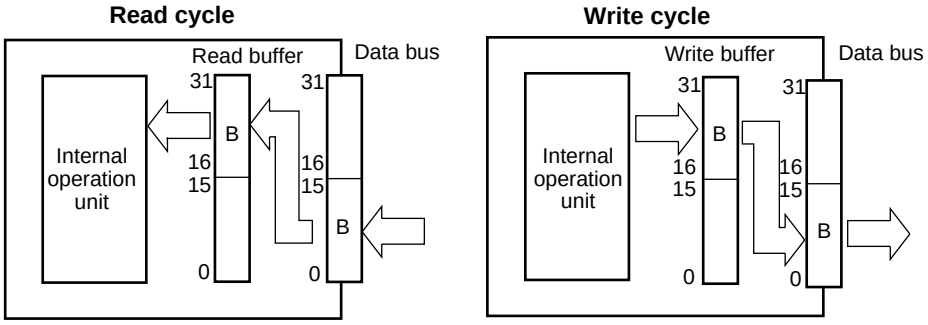
(1) Upper halfword

During read cycles, data is read from D0-D15.

During write cycles, D16-D31 data read from the write buffer is output to D0-D15.

Figure 4-1 illustrates the operation for upper halfword access. In this figure, B indicates the upper halfword (high-order 16 bits of the word).

Figure 4-1. Upper Halfword Access



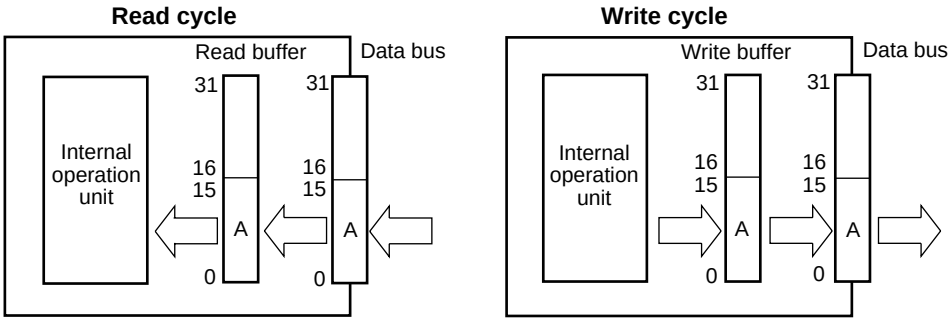
(2) Lower halfword

During read cycles, data is read from D0-D15.

During write cycles, D0-D15 data read from the write buffer is output to D0-D15.

Figure 4-2 shows the operation for lower halfword access. In this figure, A indicates the lower halfword (low-order 16 bits of the word).

Figure 4-2. Lower Halfword Access



4.1.2 Word access

Bus cycles in any of three bus states (Ta, Tw1, and Tw2) are used for word access.

During a read cycle, the low-order 16 bits of data and high-order 16 bits of data are sampled from D0-D15 in the Tw1 and Tw2 state, respectively. During write cycles, the low-order 16 bits of data and high-order 16 bits of data are output to D0-D15 in the Ta/Tw1 state and Tw2 states, respectively.

Figure 4-3. Read Cycle

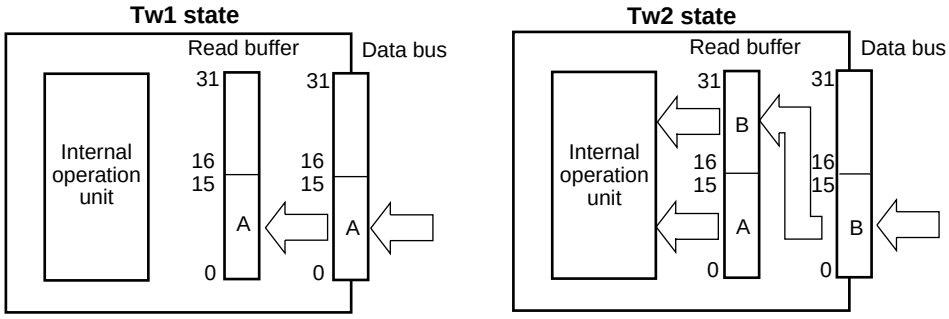
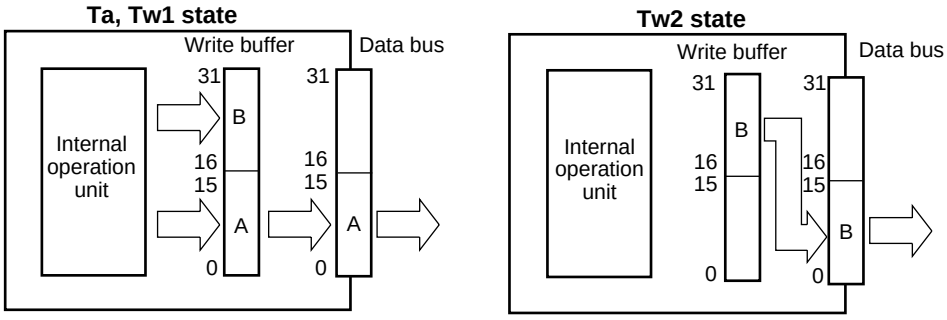


Figure 4-4. Write Cycle



4.2 Relationship between External Access and Byte Enable Signals

In 16-bit bus mode, the $\overline{BE3}$/A1 output acts as A1 and $\overline{BE2}$/BH output acts as BH. External accesses are related to the byte enable signals as indicated below.

Table 4-1. 16-Bit Bus Mode

Data length	Operand address		Byte enable				State
	Bit 1	Bit 0	A1	BH	$\overline{BE1}$	$\overline{BE0}$	
Byte	0	0	0	1	1	0	Ta,Ts
	0	1	0	1	0	1	Ta,Ts
	1	0	1	1	1	0	Ta,Ts
	1	1	1	1	0	1	Ta,Ts
Halfword	0	0	0	1	0	0	Ta,Ts
	1	0	1	1	0	0	Ta,Ts
Word	0	0	0	0	0	0	Ta,Tw1
			1	0	0	0	Tw2
Burst transfer	0	0	0	0	0	0	Ta,Tb1
			1	0	0	0	Tb2
			0	0	0	0	Tb3
			1	0	0	0	Tb4
			0	0	0	0	Tb5
			1	0	0	0	Tb6
			0	0	0	0	Tb7
			1	0	0	0	Tb8

5. INTERRUPTS

V830 interrupts include maskable interrupts, nonmaskable interrupts, and reset operations.

5.1 Maskable Interrupts

Maskable interrupt requests are themselves denoted by INT, and their interrupt levels by $\overline{\text{INTV0}}$ to $\overline{\text{INTV3}}$. The following lists pin states and the corresponding interrupt levels.

Table 5-1. Interrupt Levels

Interrupt level	$\overline{\text{INTV3}}$	$\overline{\text{INTV2}}$	$\overline{\text{INTV1}}$	$\overline{\text{INTV0}}$
15	0	0	0	0
14	0	0	0	1
13	0	0	1	0
12	0	0	1	1
11	0	1	0	0
10	0	1	0	1
9	0	1	1	0
8	0	1	1	1
7	1	0	0	0
6	1	0	0	1
5	1	0	1	0
4	1	0	1	1
3	1	1	0	0
2	1	1	0	1
1	1	1	1	0
0	1	1	1	1

INT and $\overline{\text{INTV0}}$ to $\overline{\text{INTV3}}$ are level inputs. The V830 samples an INT at the rising edge of a bus clock pulse. INT and $\overline{\text{INTV0}}$ to $\overline{\text{INTV3}}$ should be held at the active level until the V830 accepts the interrupt request and posts to a peripheral, by software, notification of the acceptance of the interrupt request. Although a change to a higher interrupt level is possible, the timing at which an interrupt request is detected cannot then be posted to peripheral. Hence, an interrupt request made before such a change may be accepted. If the interrupt request input (INT, $\overline{\text{INTV0}}$-$\overline{\text{INTV3}}$) fails to satisfy the setup time requirement for the bus clock pulse, the interrupt request will be detected at the rising edge of the next bus clock pulse.

Upon accepting an interrupt request, the V830 jumps to a fixed address to start interrupt handling. The target address of the jump is set to FE0000n0H (built-in RAM) or FFFFFEn0H (external memory), where n is the interrupt level, either of which may be specified with the IHA bit of the system register, HCCW.

Caution Interrupt level 15 is reserved for use by development tools (in-circuit emulator, ROM emulator, etc). If the user uses interrupt level 15, those development tools may fail to operate.

5.2 Nonmaskable Interrupts

The V830 samples an $\overline{\text{NMI}}$ at the rising edge of a bus clock pulse. When the $\overline{\text{NMI}}$ changes from the high to low level, an interrupt request is detected. Once a nonmaskable interrupt request has been detected, the NMI can subsequently be deactivated at any time because the $\overline{\text{NMI}}$ is detected at the falling edge. An interrupt request thus detected is retained in the CPU until the CPU starts interrupt handling.

Upon accepting a nonmaskable interrupt, the V830 jumps to the fixed address (FFFFFFD0H). If another nonmaskable interrupt is issued during nonmaskable interrupt handling (the NP bit of PSW is set to1), it is retained in the processor. If, however, another nonmaskable interrupt request is issued during clearing of the latch circuit by internal processing after the start of nonmaskable interrupt handling, it is not retained in the processor.

5.3 Reset

The V830 can be reset by inputting a low-level signal of 20 or more clock pulses to $\overline{\text{RESET}}$. After the V830 has been reset, the CPU starts program execution from address FFFFFFF0H.

If $\overline{\text{RESET}}$ is driven high, the CPU starts instruction fetching from the reset address.

Immediately after power-on or in the stop-mode state, the active pulse width of the $\overline{\text{RESET}}$ should be determined by adding the PLL oscillation settling time to the active level of 20 clock pulses.

6. CLOCK CONTROLLER

6.1 Operation Modes

The V830 supports two clock stop functions, namely, sleep mode and stop mode. Transition from one mode to another is made by executing special instructions HALT or STBY. The following lists the features of these modes:

Table 6-1. Operation Modes

	Sleep mode	Stop mode
V830 Internal state	Internal clock stop PLL operation continuous Bus hold acceptable Built-in RAM/cache data hold	Internal clock stop PLL operation stop Bus hold unacceptable Built-in RAM/cache data hold
Entry to mode	HALT instruction	STBY instruction
Escape from mode	Maskable interrupt/NMI/reset	NMI/reset

6.1.1 Sleep mode

The V830 enters sleep mode upon the execution of a HALT instruction. On the other hand, escape from sleep mode can be realized by a maskable interrupt, NMI, or reset operation.

In sleep mode, bus hold requests can be accepted. During bus hold, the status becomes high impedance and no halt acknowledge status is output. At the end of bus hold, a halt acknowledge status is output in sync with the rising edge of a bus clock pulse.

6.1.2 Stop mode

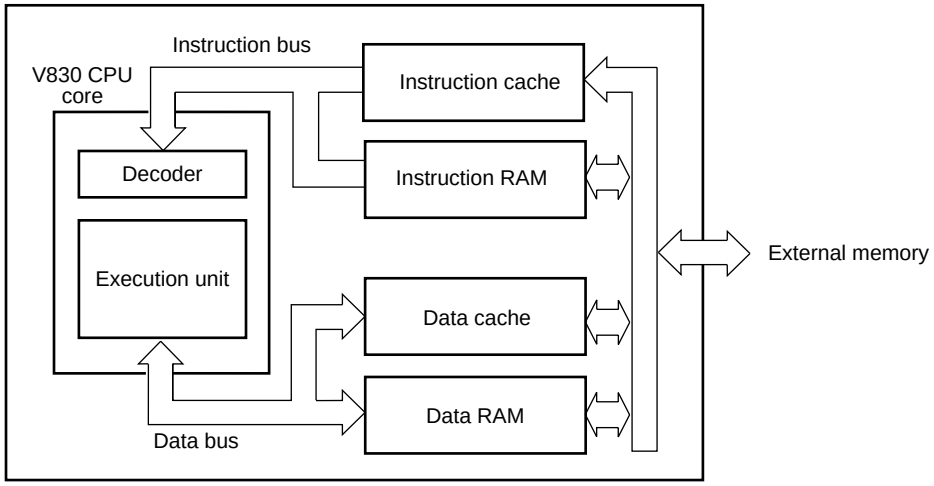
The V830 enters stop mode when an STBY instruction is executed. On the other hand, escape from stop mode can be realized using an NMI or a reset operation. The power consumption in stop mode is less than that in the sleep mode because the PLL circuit stops.

Also, no bus hold requests are accepted in the stop mode.

7. INTERNAL MEMORY

The V830 has a 4K bytes × 4 internal memory, consisting of four blocks (instruction cache, data cache, instruction RAM, and data RAM). The V830 allows any of these internal memory blocks to be accessed in one cycle.

Figure 7-1. Built-In Cache Configuration



Cautions 1. Data can not be written into the instruction cache or instruction RAM.

2. A instruction can not be written into the data cache or data RAM.

8. REGISTER SETS

8.1 Program Register Set

The V830 has two types of register sets: general-purpose register sets which can be used by programmers, and system register sets which control the state of the V830. The width of all registers is 32 bits.

8.1.1 General-purpose register set

(1) General-purpose registers

The V830 has 32 general-purpose registers, r0-r31, which can be used either as data registers or address registers. Note, however, that r0, r30, and r31 contain values that are fixed by hardware or which are used implicitly by instructions.

(a) Hardware-dependent registers

Hardware-dependent registers contain values that are fixed by hardware or which are used implicitly by instructions.

r0 : Zero register
Always contains 0.

r30 : Register reserved for operation
Serves as an auxiliary register which stores the result of a multiplication or division instruction.

r31 : Link pointer
The JAL instruction stores the return address in this register.

(b) Software-reserved registers

These registers are used by assemblers and compilers. To use them as registers for variables, first save their contents to guard against data loss or damage. When their use is no longer required restore the saved contents.

r1 : Assembler-reserved register
Serves as a working register for creating 32 bits of immediate data. It is used implicitly when the assembler calculates an effective address.

r2 : Handler stack pointer
Reserved as the stack pointer for a handler.

r3 : Stack pointer
Reserved for stack frame creation when a function is called.

r4 : Global pointer
Used when accessing a global variable in a data area.

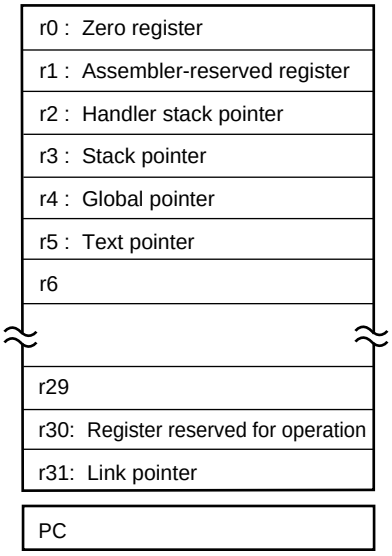
r5 : Text pointer
Points to the beginning of a text area.

8.1.2 Program counter (PC)

The program counter (PC) is a register which holds the first address of the instruction being executed. Bit 0 of the program counter is fixed to 0, but is forcibly masked to 0 upon a branch to a point other than a halfword boundary (bit 0 of the address is 0).

Upon reset, the program counter is initialized to FFFFFFF0H.

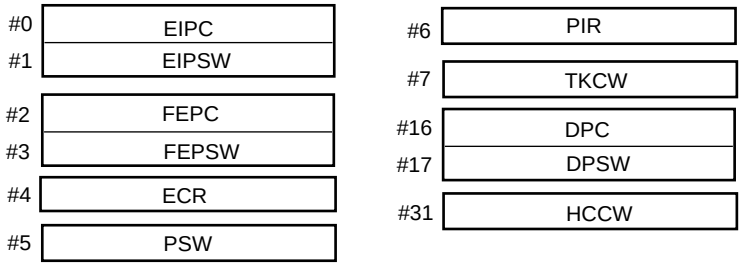
Figure 8-1. Program Registers



8.2 System Register Set

System registers are used to control the processor state, save exception/interruption information, and manage tasks. The V830 has eleven 32-bit system registers. These registers can be accessed using special instructions (LDSR and STSR instructions).

Figure 8-2. System Registers



Remark The system register number is preceded by #.

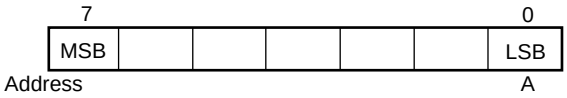
9. DATA SETS

9.1 Data Types

The V830 supports three data types: byte (8 bits), halfword (16 bits), and word (32 bits). Data of these types must be aligned with byte, halfword, or word boundaries, respectively. Addressing is based on little endian.

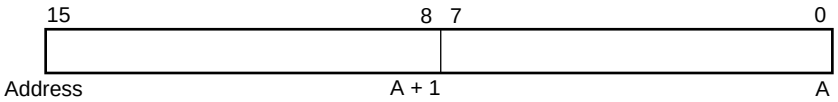
(1) Byte data

One byte of data consists of eight consecutive bits, each of which is named. Bit 0 is the LSB (Least Significant Bit) while bit 7 is the MSB (Most Significant Bit). This data can be placed at any address.



(2) Halfword data

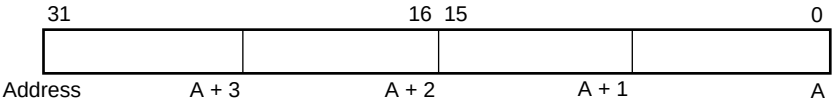
One halfword of data consists of 16 consecutive bits, each of which is named. Bit 0 is the LSB, while bit 15 is the MSB. Halfword data must be aligned with halfword boundaries (in address areas such that bit 0 of the address of the segment containing bit 0 is 0).



A = 2n (where n is a positive integer)

(3) Word data

One word of data consists of 32 consecutive bits, each of which is named. Bit 0 is the LSB and bit 31 is the MSB. Word data must be aligned with word boundaries (in address areas such that bits 0 and 1 of the address of the segment containing bit 0 are 0).



A = 4n (where n is a positive integer)

9.1.1 Integers

In the V830, integers are represented by twos complements. They are expressed by bytes, halfwords, or words. Digit ordering for integers is as follows: Bit 0 is handled as the least significant bit, regardless of the data length. Larger bit numbers correspond to higher orders.

Data length	Range (in decimal notation)
Byte (8 bits)	–128 to +127
Halfword (16 bits)	–32 768 to +32 767
Word (32 bits)	–2 147 483 648 to +2 147 483 647

9.1.2 Unsigned integers

Unsigned integers are of a data type for which the most significant bit is not handled as a sign bit, but all bits represent a positive integer. Data of this data type is represented by a binary number and has a size of a byte, halfword, or word. Digit ordering for unsigned integers is as follows: Bit 0 is handled as the least significant bit, regardless of the data length. Larger bit numbers correspond to higher orders.

Data length	Range (in decimal notation)
Byte (8 bits)	0 to 255
Halfword (16 bits)	0 to 65 535
Word (32 bits)	0 to 4 294 967 295

9.2 Data Alignment

The V830 requires that data be aligned with appropriate boundaries: word boundaries for word data, halfword boundaries for halfword data, and byte boundaries for byte data. If a data alignment error is detected, the data address is automatically changed to an accessible address. It is impossible to predict whether this address change will lead to correct or incorrect data access. This change is made as follows:

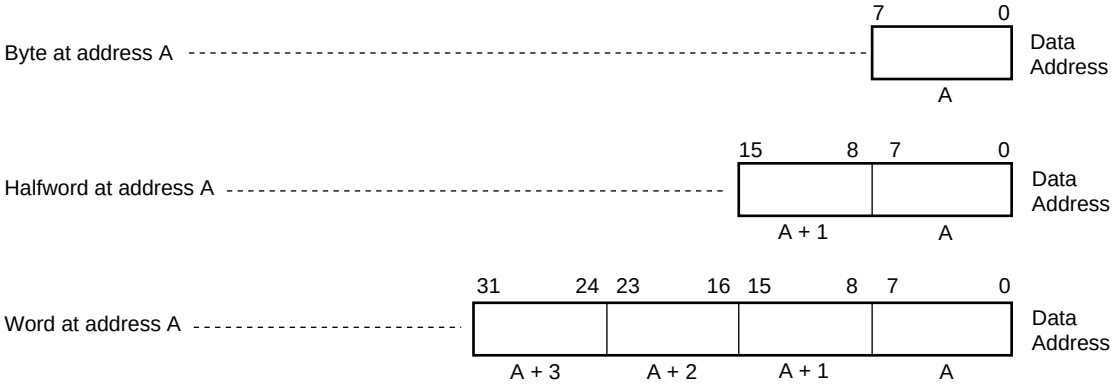
Data size	Method
Byte data	—
Halfword data	Bit 0 is masked to 0.
Word data	Bits 0 and 1 are masked to 0.

10. ADDRESS SPACE

The V830 supports 4G-byte linear address spaces for both the memory space and I/O space. It assigns 32-bit addresses to the memory space. The maximum address is $2^{32} - 1$. It also assigns 32-bit addresses to the I/O space.

Byte data aligned with each address is defined such that bits 0 and 7 are the LSB and MSB, respectively. If data consists of multiple bytes, it is defined such that the byte data at the low-order address contains the LSB and that at the high-order address contains the MSB (little-endian ordering), unless specified otherwise.

According to V830 terminology, data arranged in two-byte format is called halfword data, while that arranged in four-byte format is called word data. For data consisting of multiple bytes, the low-order address on the right and the high-order address on the left, as indicated below.



10.1 Addressing Mode

The V830 generates two types of addresses, as follows:

- Instruction addresses (used by instructions involving branching)
- Operand addresses (used by instructions which access data)

10.1.1 Instruction addresses

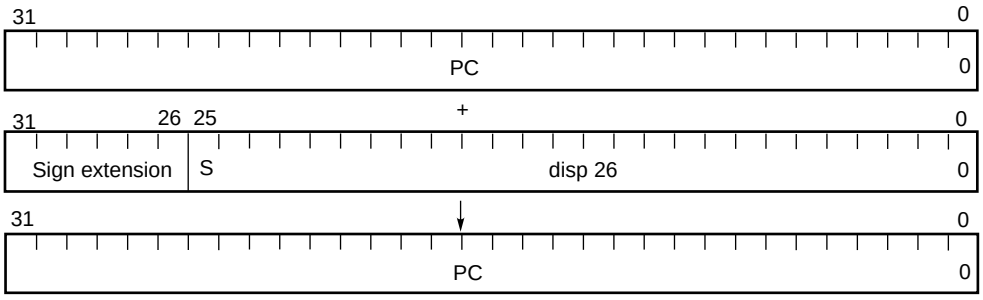
The instruction address is determined by the contents of the program counter (PC). Each time an instruction is executed, it is automatically incremented by 2 or 4, depending on the number of bytes constituting the instruction being fetched. When a branch instruction is executed, the branch address is set in the PC by the following addressing mode:

(1) Relative addressing (to PC)

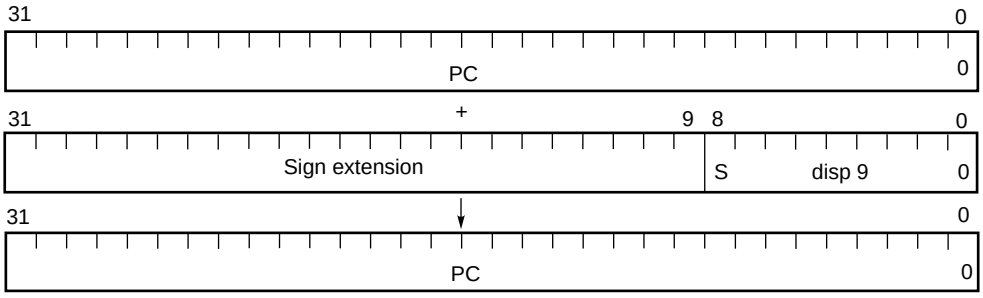
The signed 9 or 26 bits (displacement, or disp) of data contained in the operation code are added to the program counter (PC). For this addition, the displacement is handled as twos complement data. Bit 8 or 25 is the sign bit, respectively.

The JR, JAL, Bcond, and ABcond instructions use this addressing.

Addressing for JR and JAL instructions



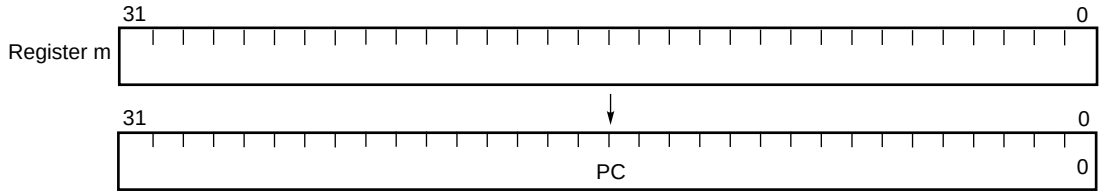
Addressing for Bcond and ABcond instructions



(2) Register addressing (via register)

The contents of the general-purpose register (r0-r31) designated in the instruction are transferred to the program counter (PC).

The JMP instruction uses this addressing.



10.1.2 Operand addresses

(1) Register addressing

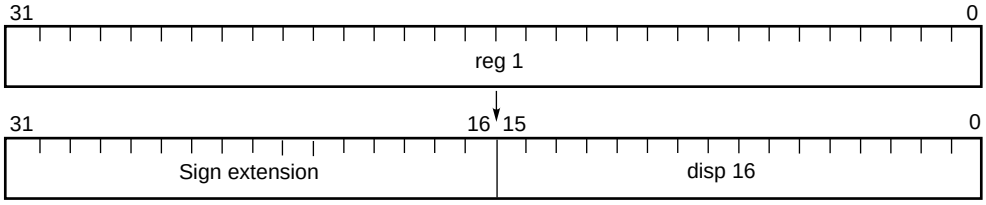
In this addressing mode, the general-purpose register designated in the general-purpose register designation field is accessed as an operand. This addressing is used by instructions whose operand format is reg1 or reg2.

(2) Immediate addressing

In this addressing mode, the 5 or 16 bits of data constituting the operation code are handled as an operand. This addressing is used by those instructions whose operand format is imm5 or imm16.

(3) Based addressing

In this addressing mode, when the memory area containing the operand is accessed, its address is determined from the sum of the contents of the general-purpose register designated by the address designation code and the 16-bit displacement in the instruction. This addressing is used by those instructions having an operand format of disp16[reg1].



11. INSTRUCTIONS

11.1 Instruction Format

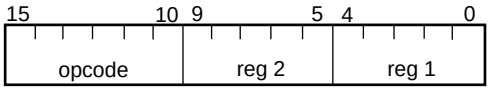
The V830 uses two instruction formats: 16-bit and 32-bit. The 16-bit instructions include binary operation, control, and conditional branch instructions, while the 32-bit instructions include load/store and I/O operation instructions, instructions for handling 16 bits of immediate data, and jump-and-link instructions.

Some instructions contain unused fields, which must be fixed to 0, which are provided for future use. When an instruction is actually loaded into memory, its configuration is as follows:

- Low-order part of each instruction format (including bit 0) → Low-order address
- High-order part of each instruction format (including bit 15 or 31) → High-order address

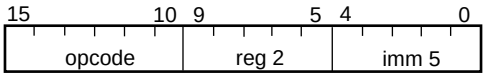
(1) reg-reg instruction format [FORMAT I]

This instruction format has a six-bit operation code field and two general-purpose register designation fields for operand specification, giving a total length of 16 bits.



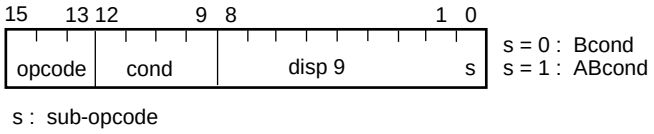
(2) imm-reg instruction format [FORMAT II]

This instruction format has a six-bit operation code field, a five-bit immediate data field, and a general-purpose register designation field, giving a total length of 16 bits.



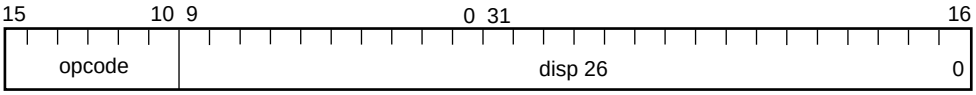
(3) Conditional branch instruction format [FORMAT III]

This instruction format has a three-bit operation code field, a four-bit condition code field, a nine-bit branch displacement field (bit 0 is handled as 0 and need not be specified), and a one-bit sub-operation code, giving a total length of 16 bits.



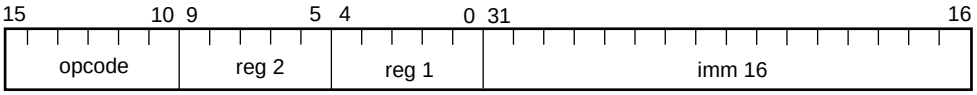
(4) Medium-distance jump instruction format [FORMAT IV]

This instruction format has a six-bit operation code field and a 26-bit displacement field (the lowest-order bit must be 0), giving a total length of 32 bits.



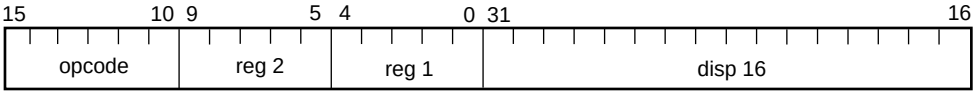
(5) Three-operand instruction format [FORMAT V]

This instruction format has a six-bit operation code field, two general-purpose register designation fields, and a 16-bit immediate data field, giving a total length of 32 bits.



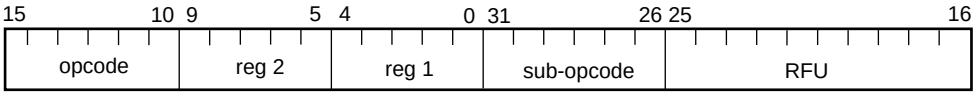
(6) Load/store instruction format [FORMAT VI]

This instruction format has a six-bit operation code field, two general-purpose register designation fields, and a 16-bit displacement field, giving a total length of 32 bits.



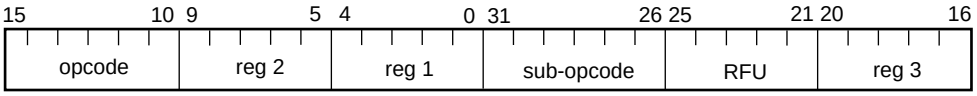
(7) Extended instruction format [FORMAT VII]

This instruction format has a six-bit operation code field, two general-purpose register designation fields, and a six-bit sub-operation code field, giving a total length of 32 bits.



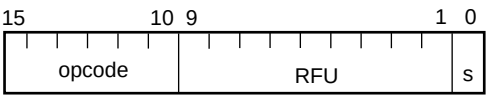
(8) Three-register operand instruction format [FORMAT VIII]

This instruction format has a six-bit operation code field, three general-purpose register designation fields, and a six-bit sub-operation code field, giving a total length of 32 bits.



(9) No-operand instruction format [FORMAT IX]

This instruction format has a six-bit operation code field and a one-bit sub-operation code field, giving a total length of 16 bits.



s : sub-opcode

11.2 Instructions (Listed Alphabetically)

The instructions are listed below in alphabetic order of their mnemonics.

Explanation of list format

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
ADD	reg1, reg2	I	*	*	*	*	

Instruction mnemonic

Instruction format (See **Section 11.1**.)

Indicates how each flag changes.
- : Does not change.
* : Changes.
0 : Becomes 0.
1 : Becomes 1.

Abbreviations of operands

Abbreviation	Meaning
reg1	General-purpose register (used as a source register)
reg2	General-purpose register (used mainly as a destination register, but in some instructions, used as a source register)
reg3	General-purpose register (used mainly as a destination register, but in some instructions, used as a source register)
imm×	× bits of immediate data
disp×	×-bit displacement
regID	System register number
vector adr	Trap handler address corresponding to trap vector

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
ABC	disp9	III	-	-	-	-	High-speed conditional branch (if Carry) relative to PC.
ABE	disp9	III	-	-	-	-	High-speed conditional branch (if Equal) relative to PC.
ABGE	disp9	III	-	-	-	-	High-speed conditional branch (if Greater than or Equal) relative to PC.
ABGT	disp9	III	-	-	-	-	High-speed conditional branch (if Greater than) relative to PC.
ABH	disp9	III	-	-	-	-	High-speed conditional branch (if Higher) relative to PC.
ABL	disp9	III	-	-	-	-	High-speed conditional branch (if Lower) relative to PC.
ABLE	disp9	III	-	-	-	-	High-speed conditional branch (if Less than or Equal) relative to PC.
ABLT	disp9	III	-	-	-	-	High-speed conditional branch (if Less than) relative to PC.
ABN	disp9	III	-	-	-	-	High-speed conditional branch (if Negative) relative to PC.
ABNC	disp9	III	-	-	-	-	High-speed conditional branch (if Not Carry) relative to PC.
ABNE	disp9	III	-	-	-	-	High-speed conditional branch (if Not Equal) relative to PC.
ABNH	disp9	III	-	-	-	-	High-speed conditional branch (if Not Higher) relative to PC.
ABNL	disp9	III	-	-	-	-	High-speed conditional branch (if Not Lower) relative to PC.
ABNV	disp9	III	-	-	-	-	High-speed conditional branch (if Not Overflow) relative to PC.
ABNZ	disp9	III	-	-	-	-	High-speed conditional branch (if Not Zero) relative to PC.
ABP	disp9	III	-	-	-	-	High-speed conditional branch (if Positive) relative to PC.
ABR	disp9	III	-	-	-	-	High-speed unconditional branch (Always) relative to PC.
ABV	disp9	III	-	-	-	-	High-speed conditional branch (if Overflow) relative to PC.
ABZ	disp9	III	-	-	-	-	High-speed conditional branch (if Zero) relative to PC.
ADD	reg1, reg2	I	*	*	*	*	Addition. reg1 is added to reg2 and the sum is written into reg2.
	imm5, reg2	II	*	*	*	*	Addition. imm5, sign-extended to a word, is added to reg2 and the sum is written into reg2.
ADDI	imm16, reg1, reg2	V	*	*	*	*	Addition. imm16, sign-extended to a word, is added to reg1, and the sum is written into reg2.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
AND	reg1, reg2	I	-	0	*	*	AND. reg2 and reg1 are ANDed and the result is written into reg2.
ANDI	imm16, reg1, reg2	V	-	0	0	*	AND. reg1 is ANDed with imm16, zero-extended to a word, and result is written into reg2.
BC	disp9	III	-	-	-	-	Conditional branch (if Carry) relative to PC.
BDLD	[reg1], [reg2]	VII	-	-	-	-	Block transfer. 4 words of data are transferred from external memory to built-in data RAM.
BDST	[reg2], [reg1]	VII	-	-	-	-	Block transfer. 4 words of data are transferred from built-in data RAM to external memory.
BE	disp9	III	-	-	-	-	Conditional branch (if Equal) relative to PC.
BGE	disp9	III	-	-	-	-	Conditional branch (if Greater than or Equal) relative to PC.
BGT	disp9	III	-	-	-	-	Conditional branch (if Greater than) relative to PC.
BH	disp9	III	-	-	-	-	Conditional branch (if Higher) relative to PC.
BILD	[reg1], [reg2]	VII	-	-	-	-	Block transfer. 4 words of data are transferred from external memory to built-in instruction RAM.
BIST	[reg2], [reg1]	VII	-	-	-	-	Block transfer. 4 words of data are transferred from built-in instruction RAM to external memory.
BL	disp9	III	-	-	-	-	Conditional branch (if Lower) relative to PC.
BLE	disp9	III	-	-	-	-	Conditional branch (if Less than or Equal) relative to PC.
BLT	disp9	III	-	-	-	-	Conditional branch (if Less than) relative to PC.
BN	disp9	III	-	-	-	-	Conditional branch (if Negative) relative to PC.
BNC	disp9	III	-	-	-	-	Conditional branch (if Not Carry) relative to PC.
BNE	disp9	III	-	-	-	-	Conditional branch (if Not Equal) relative to PC.
BNH	disp9	III	-	-	-	-	Conditional branch (if Not Higher) relative to PC.
BNL	disp9	III	-	-	-	-	Conditional branch (if Not Lower) relative to PC.
BNV	disp9	III	-	-	-	-	Conditional branch (if Not Overflow) relative to PC.
BNZ	disp9	III	-	-	-	-	Conditional branch (if Not Zero) relative to PC.
BP	disp9	III	-	-	-	-	Conditional branch (if Positive) relative to PC.
BR	disp9	III	-	-	-	-	Unconditional branch (Always) relative to PC.
BRKRET		IX	-	-	-	-	Return from fatal exception handling.
BV	disp9	III	-	-	-	-	Conditional branch (if Overflow) relative to PC.
BZ	disp9	III	-	-	-	-	Conditional branch (if Zero) relative to PC.
CAXI	disp16[reg1], reg2	VI	*	*	*	*	Inter-processor synchronization in multi-processor system.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
CMP	reg1, reg2	I	*	*	*	*	Comparison. reg2 is compared with reg1 sign-extended to a word and the condition flag is set according to the result. The comparison involves subtracting reg1 from reg2.
	imm5, rag2	II	*	*	*	*	Comparison. reg2 is compared with imm5 sign-extended to a word and the condition flag is set according to the result. The comparison involves subtracting imm5, sign-extended to a word, from reg2.
DI		II	-	-	-	-	Disable interrupt. Maskable interrupts are disabled. DI instruction cannot disable nonmaskable interrupts.
DIV	reg1, reg2	I	-	*	*	*	Division of signed operands. reg2 is divided by reg1 (signed operands). The quotient is stored in reg2 and the remainder in r30. The division is performed so that the sign of the remainder will match that of the dividend.
DIVU	reg1, reg2	I	-	0	*	*	Division of unsigned operands. reg2 is divided by reg1 (unsigned operands). The quotient is stored in reg2 and the remainder in r30. The division is performed so that the sign of the remainder will match that of the dividend.
EI		II	-	-	-	-	Enable interrupt. Maskable interrupts are enabled. The EI instruction cannot enable nonmaskable interrupts.
HALT		IX	-	-	-	-	Processor halt. The processor is placed in sleep mode.
IN.B	disp16[reg1], reg2	VI	-	-	-	-	Port input. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. A byte of data is read from the resulting port address, zero-extended to a word, then stored in reg2.
IN.H	disp16[reg1], reg2	VI	-	-	-	-	Port input. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. A halfword of data is read from the produced port address, zero-extended to a word, and stored in reg2. Bit 0 of the unsigned 32-bit port address is masked to 0.
IN.W	disp16[reg1], reg2	VI	-	-	-	-	Port input. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. A word of data is read from the resulting port address, then written into reg2. Bits 0 and 1 of the unsigned 32-bit port address are masked to 0.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
JAL	disp26	IV	-	-	-	-	Jump and link. The sum of the current PC and 4 is written into r31. disp26, sign-extended to a word, is added to the PC and the sum is set to the PC for control transfer. Bit 0 of disp26 is masked.
JMP	[reg1]	I	-	-	-	-	Indirect unconditional branch via register. Control is passed to the address designated by reg1. Bit 0 of the address is masked to 0.
JR	disp26	IV	-	-	-	-	Unconditional branch. disp26, sign-extended to a word, is added to the current PC and control is passed to the address specified by that sum. Bit 0 of disp26 is masked to 0.
LD.B	disp16[reg1], reg2	VI	-	-	-	-	Byte load. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. A byte of data is read from the produced address, sign-extended to a word, then written into reg2.
LD.H	disp16[reg1], reg2	VI	-	-	-	-	Halfword load. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. A halfword of data is read from the produced address, sign-extended to a word, then written into reg2. Bit 0 of the unsigned 32-bit address is masked to 0.
LD.W	disp16[reg1], reg2	VI	-	-	-	-	Word load. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. A word of data is read from the produced address, then written into reg2. Bits 0 and 1 of the unsigned 32-bit address are masked to 0.
LDSR	reg2, regID	II	*	*	*	*	Load into system register. The contents of reg2 are set in the system register identified by the system register number (regID).
MAC3	reg1, reg2, reg3	VIII	-	-	-	-	Saturatable operation on signed 32-bit operands. reg1 and reg2 are multiplied together as signed integers and the product is added to reg3. [If no overflow has occurred:] The result is stored in reg3. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg3; if the result is negative, the negative maximum is written into reg3.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
MACI	imm16, reg1, reg2	V	-	-	-	-	Sum-of-products operation on signed 32-bit operands. reg1 and imm16, sign-extended to 32 bits, are multiplied together as signed integers and the product is added to reg2 as a signed integer. [If no overflow has occurred:] The result is written into reg2. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg2; if the result is negative, the negative maximum is written into reg2.
MACT3	reg1, reg2, reg3	VIII	-	-	-	-	Saturatable operation on signed 32-bit operands. reg1 and reg2 are multiplied together as signed integers and the high-order 32 bits of the product are added to reg3 as signed integers. [If no overflow has occurred:] The result is written into reg3. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg3; if the result is negative, the negative maximum is written into reg3.
MAX3	reg1, reg2, reg3	VIII	-	-	-	-	Maximum. reg2 and reg1 are compared as signed integers. The larger value is written into reg3.
MIN3	reg1, reg2, reg3	VIII	-	-	-	-	Minimum. reg2 and reg1 are compared as signed integers. The smaller value is written into reg3.
MOV	reg1, reg2,	I	-	-	-	-	Data transfer. reg1 is copied to reg2 for data transfer.
	imm5, reg2	II	-	-	-	-	Data transfer. imm5, sign-extended to a word, is copied into reg2 for data transfer.
MOVEA	imm16, reg1, reg2	V	-	-	-	-	Addition. The high-order 16 bits (imm16), sign-extended to a word, are added to reg1 and the sum is written into reg2.
MOVHI	imm16, reg1, reg2	V	-	-	-	-	Addition. A word consisting of the high-order 16 bits (imm16) and low-order 16 bits (0) is added to reg1 and the sum is written into reg2.
MUL	reg1, reg2	I	-	*	*	*	Multiplication of signed operands. reg2 and reg1 are multiplied together as signed values. The high-order 32 bits of the product (double word) are written into r30 and low-order 32 bits are written into reg2.
MUL3	reg1, reg2, reg3	VIII	-	-	-	-	Multiplication of signed 32-bit operands. reg2 and reg1 are multiplied together as signed integers. The high-order 32 bits of the product are written into reg3.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
MULI	imm16, reg1, reg2	V	-	-	-	-	Saturatable multiplication of signed 32-bit operands. reg1 and imm16, sign-extended to 32 bits, are multiplied together as signed integers. [If no overflow has occurred:] The result is written into reg2. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg2; if the result is negative, the negative maximum is written into reg2.
MULT3	reg1, reg2, reg3	VIII	-	-	-	-	Saturatable multiplication of signed 32-bit operands. reg1 and reg2 are multiplied together as signed integers. The high-order 32 bits of the product are written into reg3.
MULU	reg1, reg2	I	-	*	*	*	Multiplication of unsigned operands. reg1 and reg2 are multiplied together as unsigned values. The high-order 32 bits of the product (double word) are written into r30 and the low-order 32 bits are written into reg2.
NOP		III	-	-	-	-	No operation.
NOT	reg1, reg2	I	-	0	*	*	NOT. The NOT (ones complement) of reg1 is taken and written into reg2.
OR	reg1, reg2	I	-	0	*	*	OR. The OR of reg2 and reg1 is taken and written into reg2.
ORI	imm16, reg1, reg2	V	-	0	*	*	OR. The OR of reg1 and imm16, zero-extended to a word, is taken and written into reg2.
OUT.B	reg2, disp16[reg1]	VI	-	-	-	-	Port output. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. The low-order one byte of the data in reg2 is output to the resulting port address.
OUT.H	reg2, disp16[reg1]	VI	-	-	-	-	Port output. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. The low-order two bytes of the data in reg2 are output to the resulting port address. Bit 0 of the unsigned 32-bit port address is masked to 0.
OUT.W	reg2, disp16[reg1]	VI	-	-	-	-	Port output. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit port address. The word of data in reg2 is output to the produced port address. Bits 0 and 1 of the unsigned 32-bit port address are masked to 0.
RETI		IX	*	*	*	*	Return from trap/interrupt handling routine. The return PC and PSW are read from the system registers so that program execution will return from the trap or interrupt handling routine.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
SAR	reg1 ,reg2	I	*	0	*	*	Arithmetic right shift. reg2 is arithmetically shifted to the right by the displacement specified by the low-order five bits of reg1 (MSB value is copied to the MSB in sequence). The result is written into reg2.
	imm5, reg2	II	*	0	*	*	Arithmetic right shift. reg2 is arithmetically shifted to the right by the displacement specified by imm5, zero-extended to a word. The result is written into reg2.
SATADD3	reg1, reg2, reg3	VIII	*	*	*	*	Saturatable addition. reg1 and reg2 are added together as signed integers. [If no overflow has occurred:] The result is written into reg3. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg3; if the result is negative, the negative maximum is written into reg3.
SATSUB3	reg1, reg2, reg3	VIII	*	*	*	*	Saturatable subtraction. reg1 is subtracted from reg2 as signed integers. [If no overflow has occurred:] The result is written into reg3. [If an overflow has occurred:] The SAT bit is set. If the result is positive, the positive maximum is written into reg3; if the result is negative, the negative maximum is written into reg3.
SETF	imm5, reg2	II	-	-	-	-	Set flag condition. reg2 is set to 1 if the condition specified by the low-order four bits of imm5 matches the condition flag; otherwise it is set to 0.
SHL	reg1, reg2	I	*	0	*	*	Logical left shift. reg2 is logically shifted to the left (0 is put on the LSB) by the displacement specified by the low-order five bits of reg1. The result is written into reg2.
	imm5, reg2	II	*	0	*	*	Logical left shift. reg2 is logically shifted to the left by the displacement specified by imm5, zero-extended to a word. The result is written into reg2.
SHLD3	reg1, reg2, reg3	VIII	-	-	-	-	Left shift of concatenation. The 64 bits consisting of reg3 (high order) and reg2 (low order) are logically shifted to the left by the displacement specified by the low-order five bits of reg1. The high-order 32 bits of the result are written into reg3.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
SHR	reg1, reg2	I	*	0	*	*	Logical right shift. reg2 is logically shifted to the right by the displacement specified by the low-order five bits of reg1 (0 is put on the MSB). The result is written into reg2.
	imm5, reg2	II	*	0	*	*	Logical right shift. reg2 is logically shifted to the right by the displacement specified by imm5, zero-extended to a word. The result is written into reg2.
SHRD3	reg1, reg2, reg3	VIII	-	-	-	-	Right shift of concatenation. The 64 bits consisting of reg3 (high order) and reg2 (low order) are logically shifted to the right by the displacement specified by the low-order five bits of reg1. The low-order 32 bits of the result are written into reg3.
ST.B	reg2, disp16[reg1]	VI	-	-	-	-	Byte store. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. The low-order one byte of data in reg2 is stored at the resulting address.
ST.H	reg2, disp16[reg1]	VI	-	-	-	-	Halfword store. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. The low-order two bytes of the data in reg2 are stored at the resulting address. Bit 0 of the unsigned 32-bit address is masked to 0.
ST.W	reg2, disp16[reg1]	VI	-	-	-	-	Word store. disp16, sign-extended to a word, is added to reg1 to produce an unsigned 32-bit address. The word of data in reg2 is stored at the resulting address. Bits 0 and 1 of the unsigned 32-bit address are masked to 0.
STBY		IX	-	-	-	-	Processor stop. The processor is placed in stop mode.
STSR	regID,reg2	II	-	-	-	-	System register store. The contents of the system register identified by the system register number (regID) are set in reg2.
SUB	reg1,reg2	I	*	*	*	*	Subtraction. reg1 is subtracted from reg2. The difference is written into reg2.
TRAP	vector	II	-	-	-	-	Software trap. The return PC and PSW are saved in the system registers: PSW.EP = 1 → Save in FEPC, FEPSW PSW.EP = 0 → Save in EIPC, EIPSW The exception code is set in the ECR: PSW.EP = 1 → Set in FECC PSW.EP = 0 → Set in EICC PSW flags are set: PSW.EP = 1 → Set NP and ID PSW.EP = 0 → Set EP and ID Program execution jumps to the trap handler address corresponding to the trap vector (0-31) specified by vector and begins exception handling.

Instruction	Operand(s)	Format	CY	OV	S	Z	Function
XOR	reg1,reg2	I	-	0	*	*	Exclusive OR. The exclusive OR of reg2 and reg1 is taken and written into reg2.
XORI	imm16, reg1,reg2	V	-	0	*	*	Exclusive OR. The exclusive OR of reg1 and imm16, zero-extended to a word, is taken and written into reg2.

12. INTERRUPTS AND EXCEPTIONS

Interrupts are events which occur independently of program execution. They are classified into maskable and nonmaskable interrupts. In contrast, exceptions are events which are directly related to program execution. Interrupts and exceptions do not differ greatly in their control flow, but interrupts are assigned higher handling priorities than exceptions. Fatal exceptions, however, are assigned higher priorities than interrupts.

Under the V830 architecture, the following interrupts and exceptions may occur. When an exception, maskable interrupt, or nonmaskable interrupt occurs, control is passed to a handler at an address which is predetermined a given cause. The cause of an exception can be identified by means of the exception code stored in the ECR (Exception Cause Register). The pertinent handler analyzes the contents of the ECR so that it can handle the exception or interrupt appropriately.

Table 12-1. Exception/Interrupt Source Codes

Exception/interrupt		Category	Exception code ECR**Note 1**	Handler address**Note1**	Return PC
Reset		Interrupt	FFF0H	FFFFFFF0H	Indefinite
Fatal exception		Exception	-	FFFFFFE0H	current PC
NMI		Interrupt	FFD0H	FFFFFFD0H	next PC
Double exception		Exception	**Note 2**	FFFFFFD0H	current PC
TRAP instruction (parameter 0×1n)		Exception	FFBnH	FFFFFFB0H	next PC
TRAP instruction (parameter 0×0n)		Exception	FFAnH	FFFFFFA0H	next PC
Invalid operation code		Exception	FF90H	FFFFFF90H	current PC
Division by zero		Exception	FF80H	FFFFFF80H	current PC
Interrupt level n (n = 0-15)	HWCC.IHA = 0	Interrupt	FEn0H	FFFFFEn0H	next PC
	HWCC.IHA = 1			FE0000n0H	

Notes 1. Level n is represented by a hexadecimal number (n = 0-F).

2. Exception code of the exception which caused the double exception

13. ELECTRICAL SPECIFICATIONS

ABSOLUTE MAXIMUM RATINGS (T_A = 25°C)

	Parameter	Symbol	Conditions	Rating	Unit
	Power supply voltage	V_{DD}		–0.5 to +4.5	V
	Input voltage	V_I		–0.5 to +5.5	V
	Clock input voltage	V_K	V_{DD} = 3.0 to 3.6 V	–0.5 to V_{DD} + 0.3	V
	Output voltage	V_O	V_{DD} = 3.0 to 3.6 V	–0.5 to +5.5	V
★	Operating ambient temperature	V_A		–10 to +85	°C
	Storage temperature	T_{stg}		–65 to +150	°C

Cautions 1. Do not connect an output (or input/output) pin of an IC device directly to any other output (or input/output) pin of the same device, with the exception of the open-drain and open-collector pins. Also, do not connect the V_{DD} or V_{CC} pin of an IC device directly to its GND pin or a ground. Note, however, that these restrictions do not apply to the high-impedance pins of an external circuit, whose timing has been specifically designed to avoid output collision.

2. Absolute maximum ratings are rated values, beyond which physical damage may be caused to the product; if the rated value of any of the parameters in the above table is exceeded even momentarily, the quality of the product may deteriorate. Always use the product within its rated values, therefore. For IC products, normal operation and quality are guaranteed only when the ratings and conditions described under the DC and AC characteristics are satisfied.

DC CHARACTERISTICS (T_A = –10 to +85°C, V_{DD} = 3.0 to 3.6 V)

	Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
	Low-level clock input voltage	V_{KL}		–0.5		+0.2V_{DD}	V
	High-level clock input voltage	V_{KH}		0.8V_{DD}		V_{DD} + 0.3	V
	Low-level input voltage	V_{IL}		–0.5		+0.6	V
	High-level input voltage	V_{IH}		2.0		5.5	V
	Low-level output voltage	V_{OL}	I_{OL} = 3.2 mA			0.4	V
	High-level output voltage	V_{OH}	I_{OH} = –400 μA	0.85V_{DD}			V
	Low-level input leakage current	I_{LIL}	V_I = 0 V			–5	μA
	High-level input leakage current	I_{LIH}	V_I = V_{DD}			5	μA
			V_I = 5.5 V			25	μA
	Low-level output leakage current	I_{LOL}	V_O = 0 V			–5	μA
	High-level output leakage current	I_{LOH}	V_O = V_{DD}			5	μA
★	Supply current[Note]	I_{DD}	During operation		120	170	mA
			During HALT mode		12	18	mA
			During STOP mode		30	150	μA

Note The supply current (TYP.) is as measured at 3.3 V with the output pin open.
The supply current (MAX.) is as measured at 3.6 V with the output pin open.

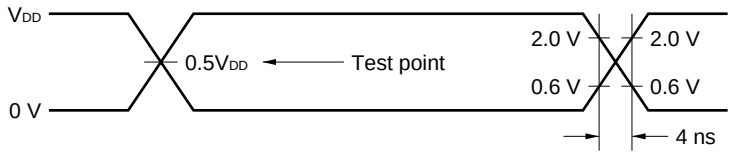
Remark f is the input frequency to the BCLK pin.

CAPACITANCE (T_A = –10 to +85°C, V_{DD} = 3.0 to 3.6 V)

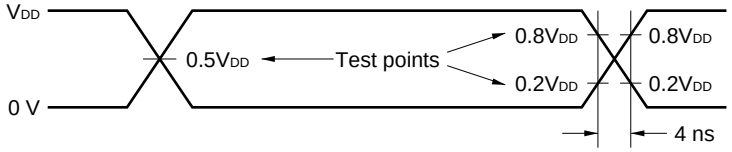
Parameter	Symbol	Conditions	MIN.	MAX.	Unit
Input capacitance	C_I	f_C = 1 MHz		15	pF
I/O capacitance	C_{IO}			15	pF

AC CHARACTERISTICS (T_A = –10 to +85°C, V_{DD} = 3.0 to 3.6 V)

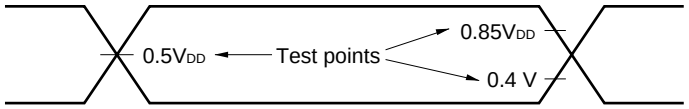
AC test input waveform (except BCLK)



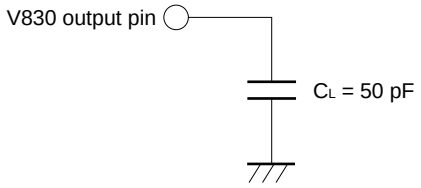
AC test input waveform (BCLK)



AC test output waveform (except BCLK)



Test load



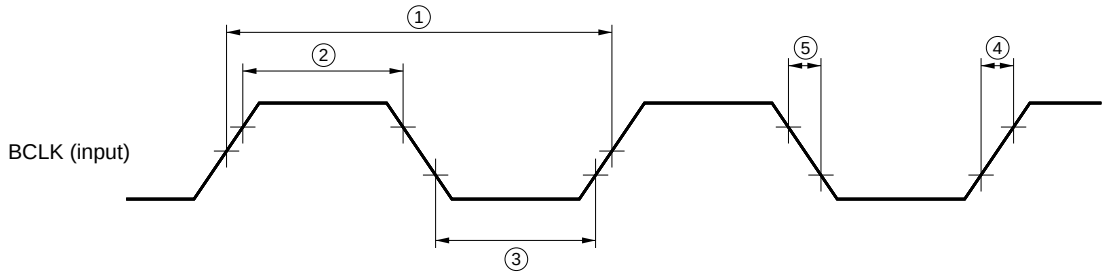
★ **(1) Clock timing**

(a) When the internal operating frequency is 75 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
Clock period	①	t_{CYK}		30	40	20	26.7	ns
Clock pulse high level width	②	t_{KKH}		3		3		ns
Clock pulse low level width	③	t_{KKL}		3		3		ns
Clock pulse rise time	④	t_{KR}			6		5	ns
Clock pulse fall time	⑤	t_{KF}			6		5	ns

(b) When the internal operating frequency is 48 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
Clock period	①	t_{CYK}		30	62.5	20	41.6	ns
Clock pulse high level width	②	t_{KKH}		3		3		ns
Clock pulse low level width	③	t_{KKL}		3		3		ns
Clock pulse rise time	④	t_{KR}			6		5	ns
Clock pulse fall time	⑤	t_{KF}			6		5	ns



Cautions 1. For clock input timings during a reset, see (2), "Reset timing."

2. The BCLK input must settle within 0.1% of t_{CYK} (①).

3. The operation is not guaranteed if t_{KR} (④) or t_{KF} (⑤) exceeds the maximum value.

★ **(2) Reset timing**

(a) When the internal operating frequency is 75 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
$\overline{\text{RESET}}$ hold time (relative to V_{DD} VALID)	⑥	t_{HVR}		0	2	0	2	μs
Clock period (when reset)	⑦	t_{CYKR}		30	40	20	26.7	ns
Clock high level width (when reset)	⑧	t_{KKHR}		8		5		ns
Clock low level width (when reset)	⑨	t_{KKLR}		8		5		ns
$\overline{\text{RESET}}$ set time (relative to BCLK↑, active)	⑩	t_{SRKF}		10		8		ns
$\overline{\text{RESET}}$ set time (relative to BCLK↑, inactive)	⑪	t_{SRKR}		10		8		ns
$\overline{\text{RESET}}$ hold time (relative to BCLK↑)	⑫	t_{HKR}		0		0		ns
$\overline{\text{RESET}}$ pulse low level width	⑬	t_{WRL}	**Note 1**	10		10		ms
			Note 2	20		20		t_{CYKR}

Notes 1. At power-on or return from stop mode

2. Other than the conditions stated in **Note 1**.

Remarks 1. The reset signal need not satisfy t_{SRKF} (⑩) or t_{SRKR} (⑪), provided it is within t_{HVR} (⑥).

2. t_{WRL} (⑬) is the number of clock cycles (t_{CYKR}) counted after the BCLK signal has settled.

(b) When the internal operating frequency is 48 to 100 MHz

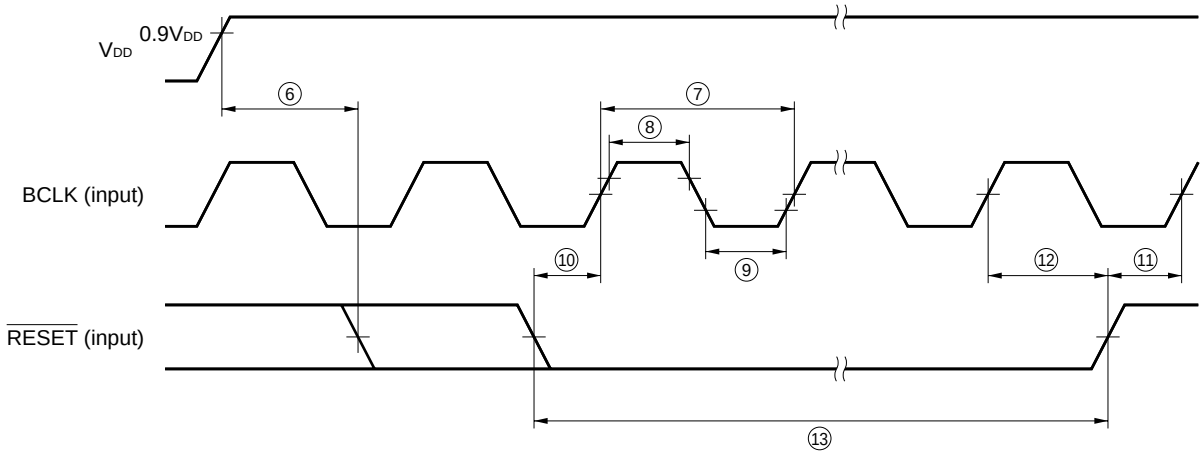
Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
$\overline{\text{RESET}}$ hold time (relative to V_{DD} VALID)	⑥	t_{HVR}		0	2	0	2	μs
Clock period (when reset)	⑦	t_{CYKR}		30	62.5	20	41.6	ns
Clock high level width (when reset)	⑧	t_{KKHR}		8		5		ns
Clock low level width (when reset)	⑨	t_{KKLR}		8		5		ns
$\overline{\text{RESET}}$ set time (relative to BCLK↑, active)	⑩	t_{SRKF}		10		8		ns
$\overline{\text{RESET}}$ set time (relative to BCLK↑, inactive)	⑪	t_{SRKR}		10		8		ns
$\overline{\text{RESET}}$ hold time (relative to BCLK↑)	⑫	t_{HKR}		0		0		ns
$\overline{\text{RESET}}$ pulse low level width	⑬	t_{WRL}	**Note 1**	10		10		ms
			Note 2	20		20		t_{CYKR}

Notes 1. At power-on or return from stop mode

2. Other than the conditions stated in **Note 1**.

Remarks 1. The reset signal need not satisfy t_{SRKF} (⑩) or t_{SRKR} (⑪), provided it is within t_{HVR} (⑥).

2. t_{WRL} (⑬) is the number of clock cycles (t_{CYKR}) counted after the BCLK signal has settled.


VDD
0.9VDD
⑥
⑦
⑧
BCLK (input)
⑩
⑨
⑫
⑪
RESET (input)
⑬

★ **(3) Memory and I/O access timing (single transfer) (1/2)**

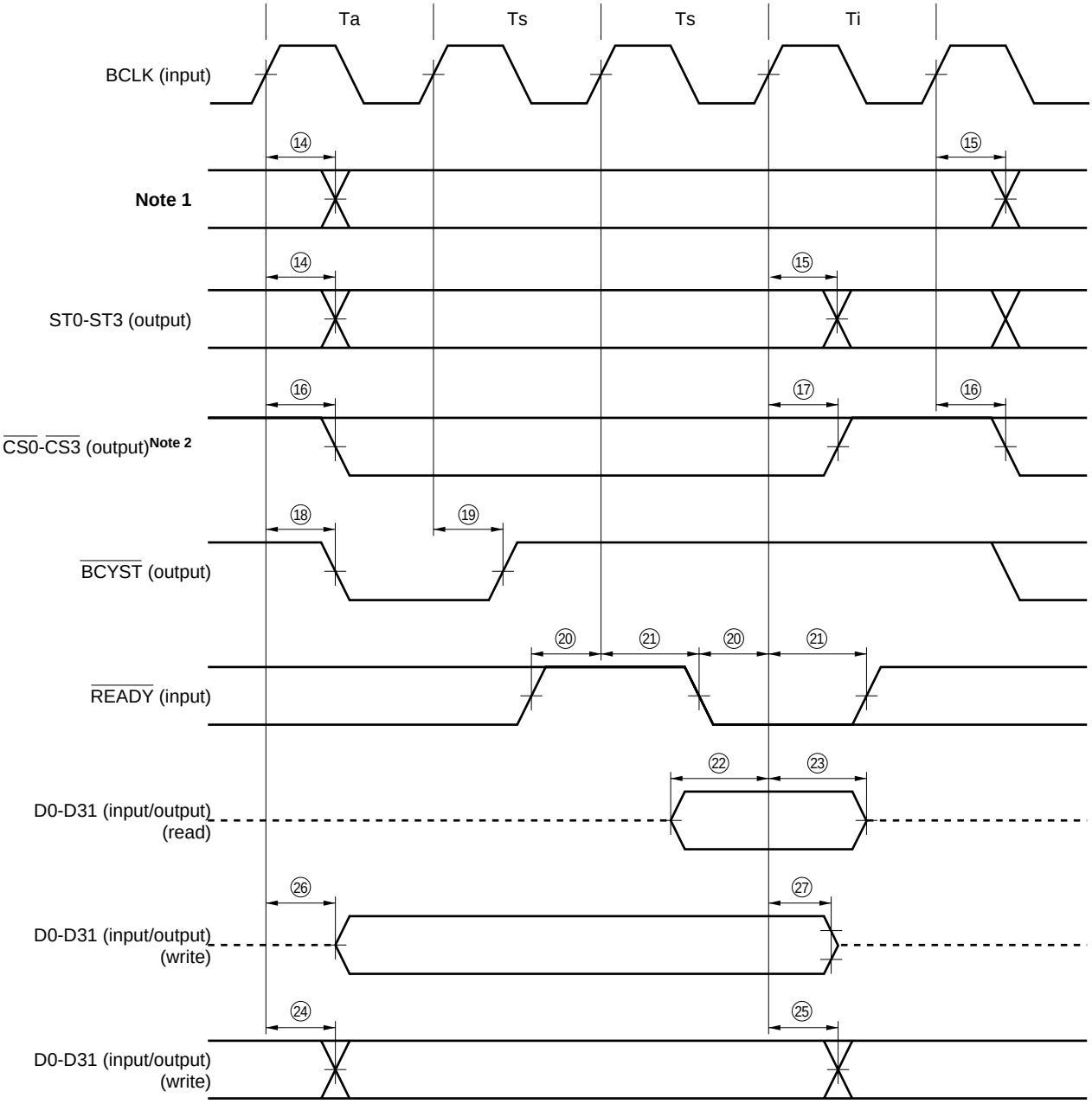
(a) When the internal operating frequency is 75 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
Address output delay (relative to BCLK↑)	⑭	t_{DKA}		2	13	2	13	ns
Address output hold time (relative to BCLK↑)	⑮	t_{HKA}		2	13	2	13	ns
$\overline{CSn}$ output delay (relative to BCLK↑)	⑯	t_{DKCS}		2	13	2	13	ns
$\overline{CSn}$ output hold time (relative to BCLK↑)	⑰	t_{HKCS}		2	13	2	13	ns
$\overline{BCYST}$ output delay (relative to BCLK↑)	⑱	t_{DKBC}		2	13	2	13	ns
$\overline{BCYST}$ output hold time (relative to BCLK↑)	⑲	t_{HKBC}		2	13	2	13	ns
$\overline{READY}$ set time (relative to BCLK↑)	⑳	t_{SRYK}		10		9		ns
$\overline{READY}$ hold time (relative to BCLK↑)	㉑	t_{HKRY}		0		0		ns
Data set time (relative to BCLK↑)	㉒	t_{SDK}		6		6		ns
Data hold time (relative to BCLK↑)	㉓	t_{HKD}		2		1		ns
Data output delay (from active, relative to BCLK↑)	㉔	t_{DKDT}		2	13	2	13	ns
Data output hold time (to active, relative to BCLK↑)	㉕	t_{HKDT}		2	13	2	13	ns
Data output delay (from float, relative to BCLK↑)	㉖	t_{LZKDT}		2	13	2	13	ns
Data output hold time (to float, relative to BCLK↑)	㉗	t_{HZKDT}		3	20	3	20	ns

(b) When the internal operating frequency is 48 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
Address output delay (relative to BCLK↑)	⑭	t_{DKA}		1	13	1	13	ns
Address output hold time (relative to BCLK↑)	⑮	t_{HKA}		1	13	1	13	ns
$\overline{CSn}$ output delay (relative to BCLK↑)	⑯	t_{DKCS}		1	13	1	13	ns
$\overline{CSn}$ output hold time (relative to BCLK↑)	⑰	t_{HKCS}		1	13	1	13	ns
$\overline{BCYST}$ output delay (relative to BCLK↑)	⑱	t_{DKBC}		1	13	1	13	ns
$\overline{BCYST}$ output hold time (relative to BCLK↑)	⑲	t_{HKBC}		1	13	1	13	ns
$\overline{READY}$ set time (relative to BCLK↑)	⑳	t_{SRYK}		10		10		ns
$\overline{READY}$ hold time (relative to BCLK↑)	㉑	t_{HKRY}		0		0		ns
Data set time (relative to BCLK↑)	㉒	t_{SDK}		7		7		ns
Data hold time (relative to BCLK↑)	㉓	t_{HKD}		2		1		ns
Data output delay (from active, relative to BCLK↑)	㉔	t_{DKDT}		1	13	1	13	ns
Data output hold time (to active, relative to BCLK↑)	㉕	t_{HKDT}		1	13	1	13	ns
Data output delay (from float, relative to BCLK↑)	㉖	t_{LZKDT}		1	13	1	13	ns
Data output hold time (to float, relative to BCLK↑)	㉗	t_{HZKDT}		3	20	3	20	ns

(3) Memory and I/O access timing (single transfer) (2/2)



Notes 1. A2-A27 (output), BE0-BE3 (output), R/W (output)

2. A28-A31 are output at CS0-CS3 when the chip select function is not used.

Remark Dotted lines indicate the high-impedance state.

★ **(4) Memory access timing (burst transfer) (1/3)**

(a) When the internal operating frequency is 75 to 100 MHz

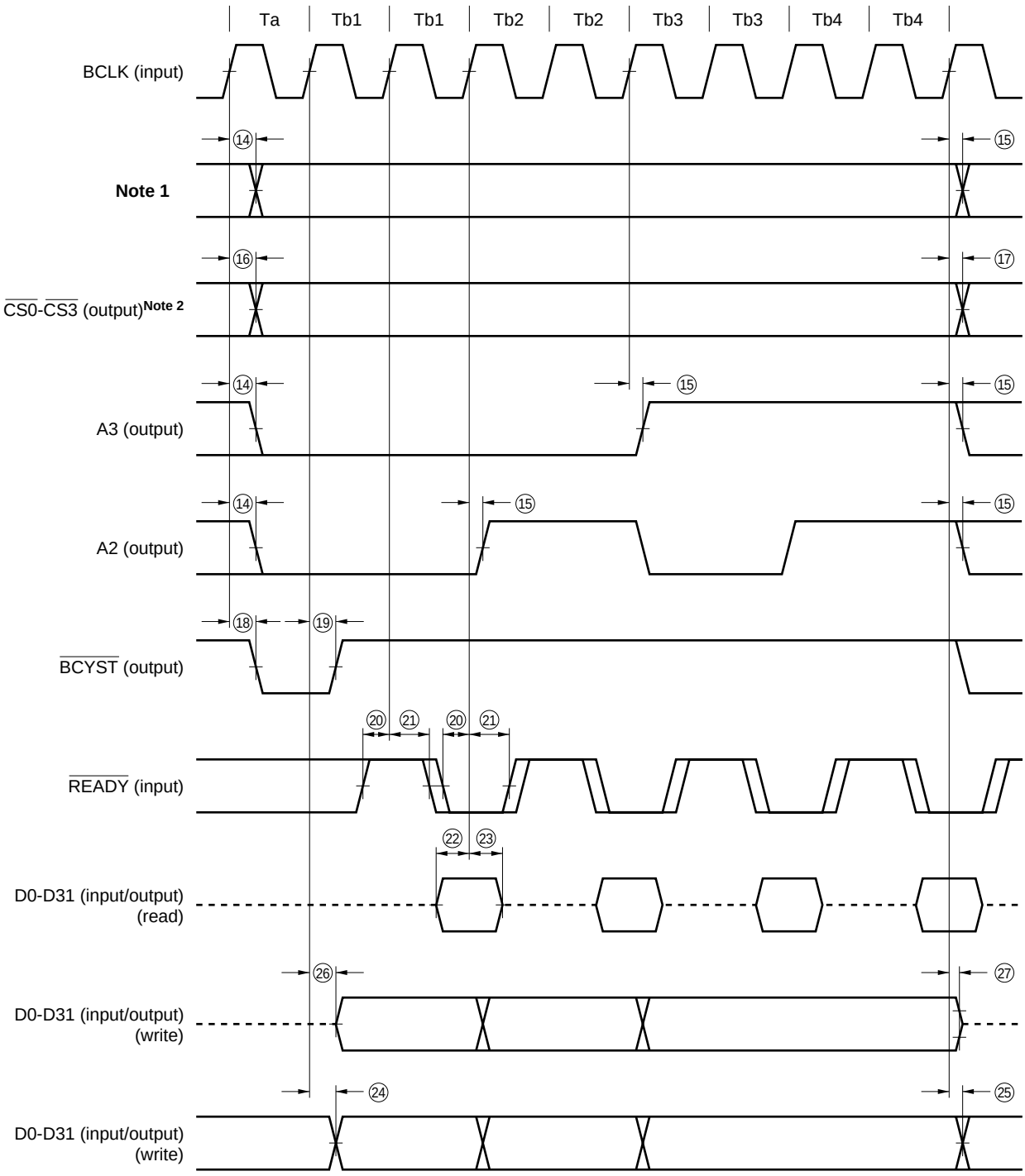
Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
Address output delay (relative to BCLK↑)	⑭	t_{DKA}		2	13	2	13	ns
Address output hold time (relative to BCLK↑)	⑮	t_{HKA}		2	13	2	13	ns
$\overline{CSn}$ output delay (relative to BCLK↑)	⑯	t_{DKCS}		2	13	2	13	ns
$\overline{CSn}$ output hold time (relative to BCLK↑)	⑰	t_{HKCS}		2	13	2	13	ns
$\overline{BCYST}$ output delay (relative to BCLK↑)	⑱	t_{DKBC}		2	13	2	13	ns
$\overline{BCYST}$ output hold time (relative to BCLK↑)	⑲	t_{HKBC}		2	13	2	13	ns
$\overline{READY}$ set time (relative to BCLK↑)	⑳	t_{SRYK}		10		9		ns
$\overline{READY}$ hold time (relative to BCLK↑)	㉑	t_{HKRY}		0		0		ns
Data set time (relative to BCLK↑)	㉒	t_{SDK}		6		6		ns
Data hold time (relative to BCLK↑)	㉓	t_{HKD}		2		1		ns
Data output delay (from active, relative to BCLK↑)	㉔	t_{DKDT}		2	13	2	13	ns
Data output hold time (to active, relative to BCLK↑)	㉕	t_{HKDT}		2	13	2	13	ns
Data output delay (from float, relative to BCLK↑)	㉖	t_{LZKDT}		2	13	2	13	ns
Data output hold time (to float, relative to BCLK↑)	㉗	t_{HZKDT}		3	20	3	20	ns

(b) When the internal operating frequency is 48 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
Address output delay (relative to BCLK↑)	⑭	t_{DKA}		1	13	1	13	ns
Address output hold time (relative to BCLK↑)	⑮	t_{HKA}		1	13	1	13	ns
$\overline{\text{CSn}}$ output delay (relative to BCLK↑)	⑯	t_{DKCS}		1	13	1	13	ns
$\overline{\text{CSn}}$ output hold time (relative to BCLK↑)	⑰	t_{HKCS}		1	13	1	13	ns
$\overline{\text{BCYST}}$ output delay (relative to BCLK↑)	⑱	t_{DKBC}		1	13	1	13	ns
$\overline{\text{BCYST}}$ output hold time (relative to BCLK↑)	⑲	t_{HKBC}		1	13	1	13	ns
$\overline{\text{READY}}$ set time (relative to BCLK↑)	⑳	t_{SRYK}		10		10		ns
$\overline{\text{READY}}$ hold time (relative to BCLK↑)	㉑	t_{HKRY}		0		0		ns
Data set time (relative to BCLK↑)	㉒	t_{SDK}		7		7		ns
Data hold time (relative to BCLK↑)	㉓	t_{HKD}		2		1		ns
Data output delay (from active, relative to BCLK↑)	㉔	t_{DKDT}		1	13	1	13	ns
Data output hold time (to active, relative to BCLK↑)	㉕	t_{HKDT}		1	13	1	13	ns
Data output delay (from float, relative to BCLK↑)	㉖	t_{LZKDT}		1	13	1	13	ns
Data output hold time (to float, relative to BCLK↑)	㉗	t_{HZKDT}		3	20	3	20	ns

(4) Memory access timing (burst transfer) (2/3)

(c) 32-bit bus mode



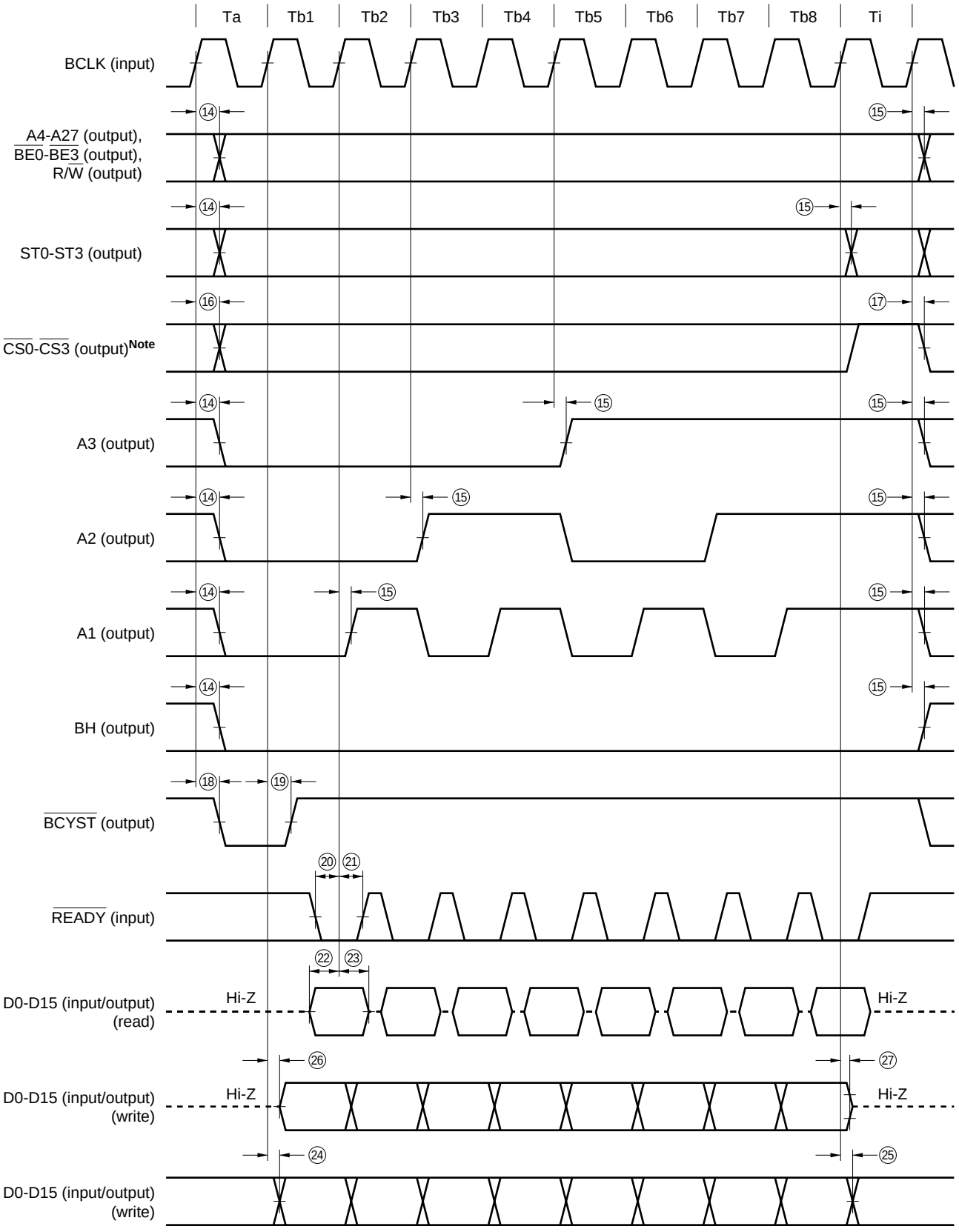
Notes 1. A4-A27 (output), BE0-BE3 (output), ST0-ST3 (output), R/W (output)

2. A28-A31 are output at CS0-CS3 when the chip select function is not used.

Remark Dotted lines indicate the high-impedance state.

(4) Memory access timing (burst transfer) (3/3)

(d) 16-bit bus mode



Note A28-A31 are output at $\overline{CS0}$-$\overline{CS3}$ when the chip select function is not used.

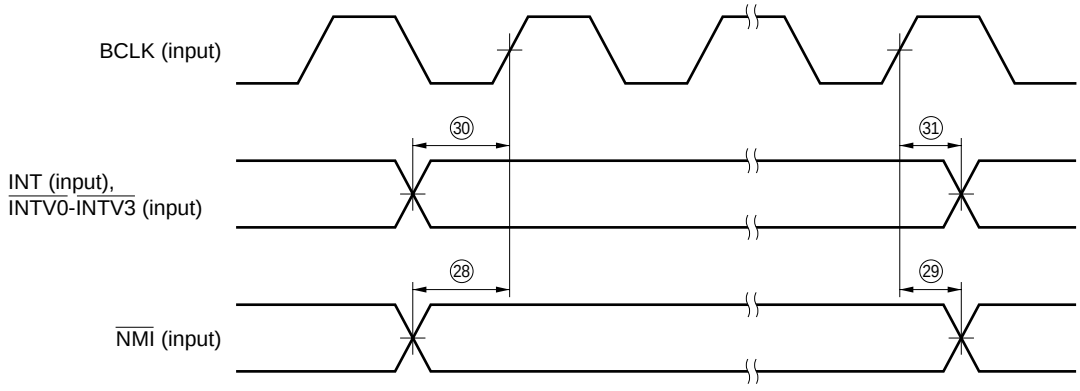
★ **(5) Interrupt timing**

(a) When the internal operating frequency is 75 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
$\overline{\text{NMI}}$ set time (relative to BCLK↑)	㉘	t_{SNK}		6		6		ns
$\overline{\text{NMI}}$ hold time (relative to BCLK↑)	㉙	t_{HKN}		2		1		ns
INT set time (relative to BCLK↑)	㉚	t_{SIK}		6		6		ns
INT hold time (relative to BCLK↑)	㉛	t_{HKI}		2		1		ns

(b) When the internal operating frequency is 48 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
$\overline{\text{NMI}}$ set time (relative to BCLK↑)	㉘	t_{SNK}		7		7		ns
$\overline{\text{NMI}}$ hold time (relative to BCLK↑)	㉙	t_{HKN}		2		1		ns
INT set time (relative to BCLK↑)	㉚	t_{SIK}		7		7		ns
INT hold time (relative to BCLK↑)	㉛	t_{HKI}		2		1		ns



★ **(6) Bus hold timing (1/2)**

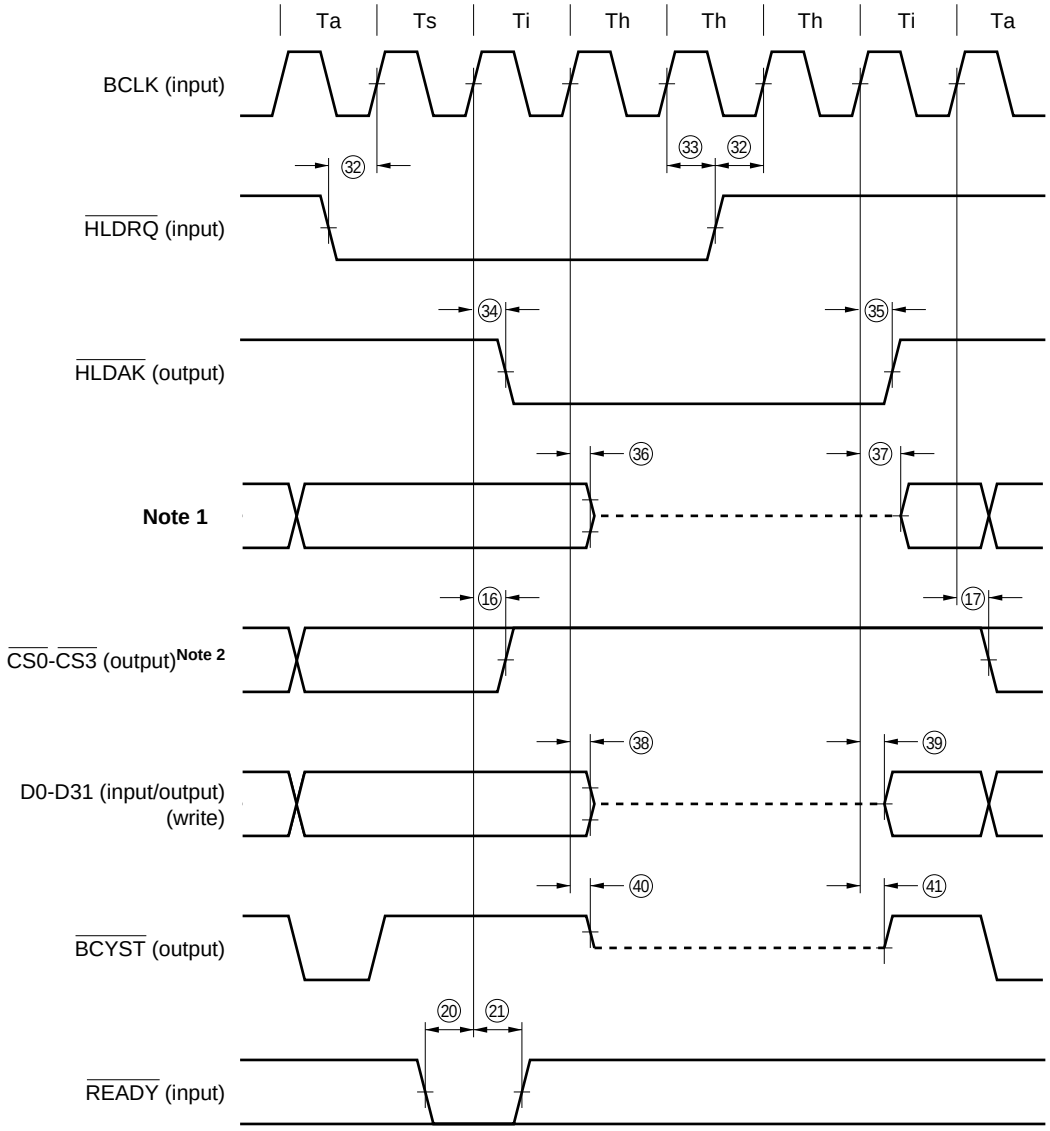
(a) When the internal operating frequency is 75 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
$\overline{CSn}$ output delay (relative to BCLK↑)	⑯	t_{DKCS}		2	13	2	13	ns
$\overline{CSn}$ output hold time (relative to BCLK↑)	⑰	t_{HKCS}		2	13	2	13	ns
$\overline{READY}$ set time (relative to BCLK↑)	⑳	t_{SRYK}		10		9		ns
$\overline{READY}$ hold time (relative to BCLK↑)	㉑	t_{HKRY}		0		0		ns
$\overline{HLDRQ}$ set time (relative to BCLK↑)	㉜	t_{SHQK}		6		6		ns
$\overline{HLDRQ}$ hold time (relative to BCLK↑)	㉝	t_{HKHQ}		2		1		ns
$\overline{HLDAK}$ output delay (relative to BCLK↑)	㉞	t_{DKHA}		2	13	2	13	ns
$\overline{HLDAK}$ output hold time (relative to BCLK↑)	㉟	t_{HKHA}		2	13	2	13	ns
Address delay (from active, relative to BCLK↑)	㊱	t_{HZKA}		3	20	3	20	ns
Address delay (from float, relative to BCLK↑)	㊲	t_{LZKA}		2	13	2	10	ns
Data delay (from active, relative to BCLK↑)	㊳	t_{HZKD}		3	20	3	20	ns
Data delay (from float, relative to BCLK↑)	㊴	t_{LZKD}		2	13	2	10	ns
$\overline{BCYST}$ delay (from active, relative to BCLK↑)	㊵	t_{HZKBC}		3	20	3	20	ns
$\overline{BCYST}$ delay (from float, relative to BCLK↑)	㊶	t_{LZKBC}		2	13	2	10	ns

(b) When the internal operating frequency is 48 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
$\overline{\text{CSn}}$ output delay (relative to BCLK↑)	⑯	t_{DKCS}		1	13	1	13	ns
$\overline{\text{CSn}}$ output hold time (relative to BCLK↑)	⑰	t_{HKCS}		1	13	1	13	ns
$\overline{\text{READY}}$ set time (relative to BCLK↑)	⑳	t_{SRYK}		10		10		ns
$\overline{\text{READY}}$ hold time (relative to BCLK↑)	㉑	t_{HKRY}		0		0		ns
$\overline{\text{HLDRQ}}$ set time (relative to BCLK↑)	㉜	t_{SHQK}		7		7		ns
$\overline{\text{HLDRQ}}$ hold time (relative to BCLK↑)	㉝	t_{HKHQ}		2		1		ns
$\overline{\text{HLDAK}}$ output delay (relative to BCLK↑)	㉞	t_{DKHA}		1	13	1	13	ns
$\overline{\text{HLDAK}}$ output hold time (relative to BCLK↑)	㉟	t_{HKHA}		1	13	1	13	ns
Address delay (from active, relative to BCLK↑)	㊱	t_{HZKA}		3	20	3	20	ns
Address delay (from float, relative to BCLK↑)	㊲	t_{LZKA}		2	13	2	10	ns
Data delay (from active, relative to BCLK↑)	㊳	t_{HZKD}		3	20	3	20	ns
Data delay (from float, relative to BCLK↑)	㊴	t_{LZKD}		2	13	2	10	ns
$\overline{\text{BCYST}}$ delay (from active, relative to BCLK↑)	㊵	t_{HZKBC}		3	20	3	20	ns
$\overline{\text{BCYST}}$ delay (from float, relative to BCLK↑)	㊶	t_{LZKBC}		2	13	2	10	ns

(6) Bus hold timing (2/2)



Notes 1. A2-A27 (output), BE0-BE3 (output), ST0-ST3 (output), R/W (output)

2. A28-A31 are output at CS0-CS3 when the chip select function is not used.
The timings of these signals are the same as stated in **Note 1**.

Remark Dotted lines indicate the high-impedance state.

★ **(7) Halt acknowledge cycle (1/2)**

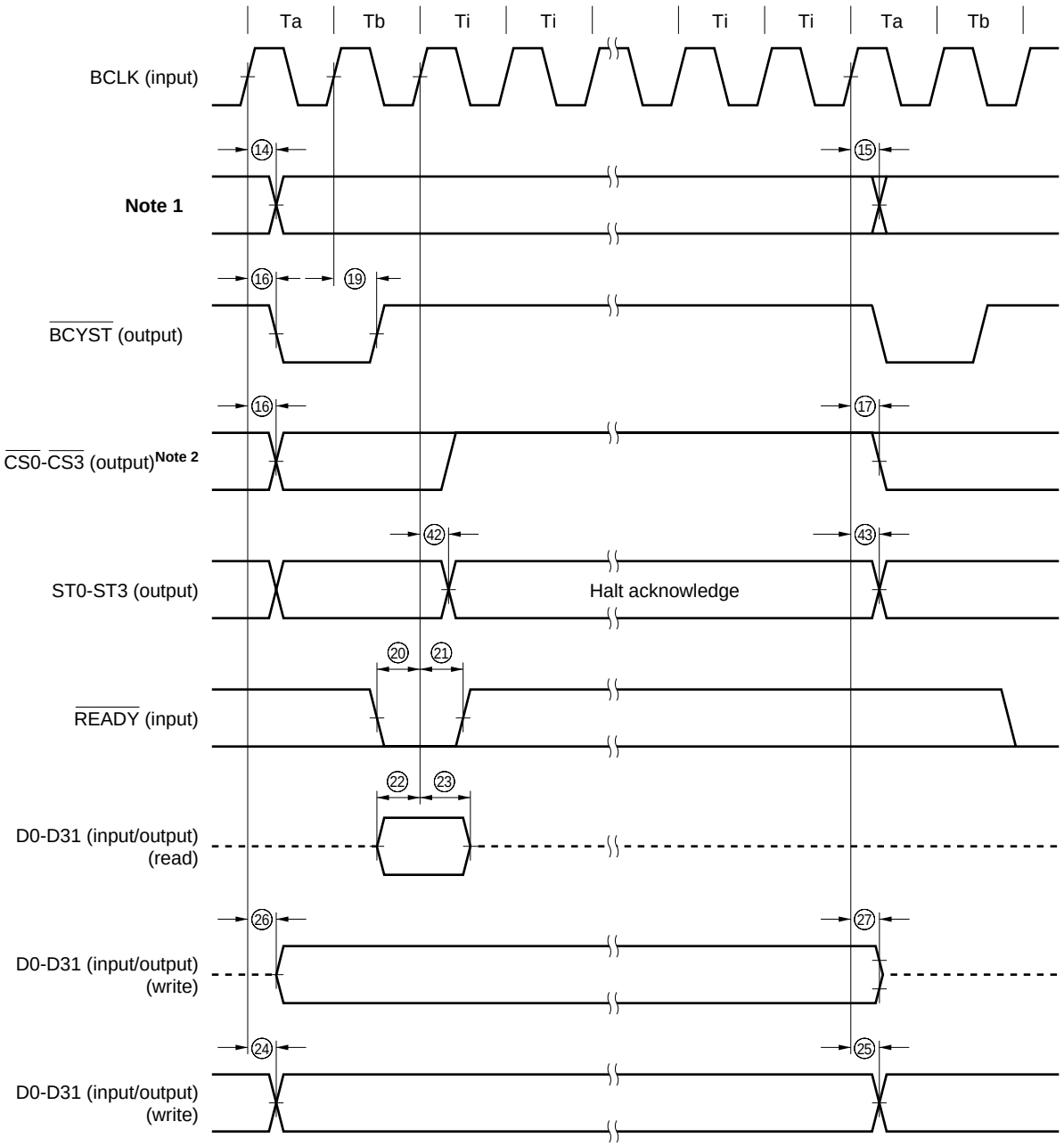
(a) When the internal operating frequency is 75 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
Address output delay (relative to BCLK↑)	⑭	t_{DKA}		2	13	2	13	ns
Address output hold time (relative to BCLK↑)	⑮	t_{HKA}		2	13	2	13	ns
$\overline{CSn}$ output delay (relative to BCLK↑)	⑯	t_{DKCS}		2	13	2	13	ns
$\overline{CSn}$ output hold time (relative to BCLK↑)	⑰	t_{HKCS}		2	13	2	13	ns
$\overline{BCYST}$ output delay (relative to BCLK↑)	⑱	t_{DKBC}		2	13	2	13	ns
$\overline{BCYST}$ output hold time (relative to BCLK↑)	⑲	t_{HKBC}		2	13	2	13	ns
$\overline{READY}$ set time (relative to BCLK↑)	⑳	t_{SRYK}		10		9		ns
$\overline{READY}$ hold time (relative to BCLK↑)	㉑	t_{HKRY}		0		0		ns
Data set time (relative to BCLK↑)	㉒	t_{SDK}		6		6		ns
Data hold time (relative to BCLK↑)	㉓	t_{HKD}		2		1		ns
Data output delay (from active, relative to BCLK↑)	㉔	t_{DKDT}		2	13	2	13	ns
Data output hold time (to active, relative to BCLK↑)	㉕	t_{HKDT}		2	13	2	13	ns
Data output delay (from float, relative to BCLK↑)	㉖	t_{LZKDT}		2	13	2	13	ns
Data output hold time (to float, relative to BCLK↑)	㉗	t_{HZKDT}		3	20	3	20	ns
Stn output delay (relative to BCLK↑)	㊷	t_{DKST}		2	13	2	13	ns
Stn output hold time (relative to BCLK↑)	㊸	t_{HKST}		2	13	2	13	ns

(b) When the internal operating frequency is 48 to 100 MHz

Parameter	Symbol		Conditions	When ϕ = 3f		When ϕ = 2f		Unit
				MIN.	MAX.	MIN.	MAX.	
Address output delay (relative to BCLK↑)	⑭	t_{DKA}		1	13	1	10	ns
Address output hold time (relative to BCLK↑)	⑮	t_{HKA}		1	13	1	10	ns
$\overline{CSn}$ output delay (relative to BCLK↑)	⑯	t_{DKCS}		1	13	1	10	ns
$\overline{CSn}$ output hold time (relative to BCLK↑)	⑰	t_{HKCS}		1	13	1	10	ns
$\overline{BCYST}$ output delay (relative to BCLK↑)	⑱	t_{DKBC}		1	13	1	13	ns
$\overline{BCYST}$ output hold time (relative to BCLK↑)	⑲	t_{HKBC}		1	13	1	13	ns
$\overline{READY}$ set time (relative to BCLK↑)	⑳	t_{SRYK}		10		10		ns
$\overline{READY}$ hold time (relative to BCLK↑)	㉑	t_{HKRY}		0		0		ns
Data set time (relative to BCLK↑)	㉒	t_{SDK}		7		7		ns
Data hold time (relative to BCLK↑)	㉓	t_{HKD}		2		1		ns
Data output delay (from active, relative to BCLK↑)	㉔	t_{DKDT}		1	13	1	13	ns
Data output hold time (to active, relative to BCLK↑)	㉕	t_{HKDT}		1	13	1	13	ns
Data output delay (from float, relative to BCLK↑)	㉖	t_{LZKDT}		1	13	1	13	ns
Data output hold time (to float, relative to BCLK↑)	㉗	t_{HZKDT}		3	20	3	20	ns
Stn output delay (relative to BCLK↑)	㊷	t_{DKST}		1	13	1	13	ns
Stn output hold time (relative to BCLK↑)	㊸	t_{HKST}		1	13	1	13	ns

(7) Halt acknowledge cycle (2/2)



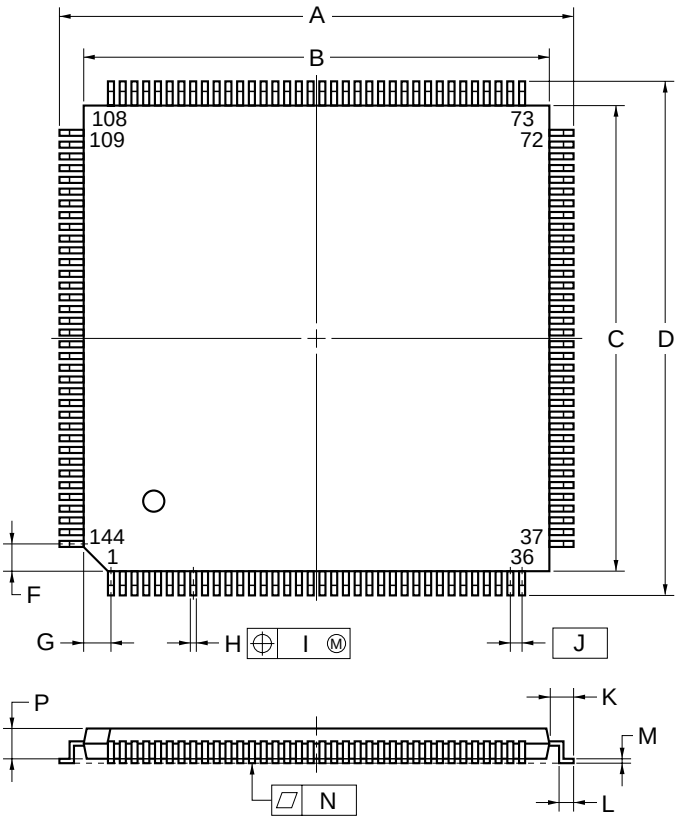
Notes 1. A2-A27 (output), BE0-BE3 (output), R/W (output)

2. A28-A31 are output at CS0-CS3 when the chip select function is not being used. The timings of these signals are the same as stated in **Note 1**.

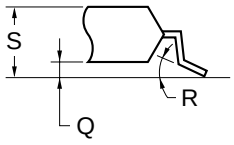
Remark Dotted lines indicate the high-impedance state.

14. PACKAGE DRAWING

144 PIN PLASTIC LQFP (FINE PITCH) (20×20)



detail of lead end



NOTE

Each lead centerline is located within 0.10 mm (0.004 inch) of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS	INCHES
A	22.0±0.2	0.866±0.008
B	20.0±0.2	$0.787^{+0.009}_{-0.008}$
C	20.0±0.2	$0.787^{+0.009}_{-0.008}$
D	22.0±0.2	0.866±0.008
F	1.25	0.049
G	1.25	0.049
H	$0.22^{+0.05}_{-0.04}$	0.009±0.002
I	0.10	0.004
J	0.5 (T.P.)	0.020 (T.P.)
K	1.0±0.2	$0.039^{+0.009}_{-0.008}$
L	0.5±0.2	$0.020^{+0.008}_{-0.009}$
M	$0.145^{+0.055}_{-0.045}$	0.006±0.002
N	0.10	0.004
P	1.4±0.1	0.055±0.004
Q	0.125±0.075	0.005±0.003
R	$3^{\circ}{}^{+7^{\circ}}_{-3^{\circ}}$	$3^{\circ}{}^{+7^{\circ}}_{-3^{\circ}}$
S	1.7 MAX.	0.067 MAX.

S144GJ-50-8EU-2

15. RECOMMENDED SOLDERING CONDITIONS

The conditions listed below shall be met when soldering the μPD705100.

For details of the recommended soldering conditions, refer to our document **Semiconductor Device Mounting Technology Manual** (C10535E).

Please consult with our sales offices in case any other soldering process is used, or in case soldering is done under different conditions.

Table 15-1. Soldering Conditions for Surface-Mount Devices

Soldering process	Soldering conditions	Symbol
Infrared ray reflow	Peak package's surface temperature: 235 °C Reflow time: 30 seconds or less (210 °C or more) Maximum allowable number of reflow processes: 2 Exposure limit: 3 days**Note** (10 hours of pre-baking is required at 125 °C afterward) <Caution> Non-heat-resistant trays, such as magazine and taping trays, cannot be baked before unpacking.	IR35-103-2
VPS	Peak package's surface temperature: 215 °C Reflow time: 40 seconds or less (200 °C or more) Maximum allowable number of reflow processes: 2 Exposure limit: 3 days**Note** (10 hours of pre-baking is required at 125 °C afterward) <Caution> Non-heat-resistant trays, such as magazine and taping trays, cannot be baked before unpacking.	VP15-103-2
Partial heating method	Terminal temperature: 300 °C or less Heat time: 3 seconds or less (for one side of a device)	-

Note Maximum number of days during which the product can be stored at a temperature of 25 °C and a relative humidity of 65 % or less after dry-pack package is opened.

Caution Do not apply two or more different soldering methods to one chip (except for partial heating method for terminal sections).

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note: Strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note: No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS device behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note: Power-on does not necessarily define initial status of MOS device. Production process of MOS does not define the initial operation status of the device. Immediately after the power source is turned ON, the devices with reset function have not yet been initialized. Hence, power-on does not guarantee out-pin levels, I/O settings or contents of registers. Device is not initialized until the reset signal is received. Reset operation must be executed immediately after power-on for devices having reset function.

Regional Information

Some information contained in this document may vary from country to country. Before using any NEC product in your application, please contact the NEC office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

NEC Electronics Inc. (U.S.)
Santa Clara, California
Tel: 408-588-6000
800-366-9782
Fax: 408-588-6130
800-729-9288

NEC Electronics (Germany) GmbH
Duesseldorf, Germany
Tel: 0211-65 03 02
Fax: 0211-65 03 490

NEC Electronics (UK) Ltd.
Milton Keynes, UK
Tel: 01908-691-133
Fax: 01908-670-290

NEC Electronics Italiana s.r.1.
Milano, Italy
Tel: 02-66 75 41
Fax: 02-66 75 42 99

NEC Electronics (Germany) GmbH
Benelux Office
Eindhoven, The Netherlands
Tel: 040-2445845
Fax: 040-2444580

NEC Electronics (France) S.A.
Velizy-Villacoublay, France
Tel: 01-30-67 58 00
Fax: 01-30-67 58 99

NEC Electronics (France) S.A.
Spain Office
Madrid, Spain
Tel: 01-504-2787
Fax: 01-504-2860

NEC Electronics (Germany) GmbH
Scandinavia Office
Taeby, Sweden
Tel: 08-63 80 820
Fax: 08-63 80 388

NEC Electronics Hong Kong Ltd.
Hong Kong
Tel: 2886-9318
Fax: 2886-9022/9044

NEC Electronics Hong Kong Ltd.
Seoul Branch
Seoul, Korea
Tel: 02-528-0303
Fax: 02-528-4411

NEC Electronics Singapore Pte. Ltd.
United Square, Singapore 1130
Tel: 253-8311
Fax: 250-3583

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-719-2377
Fax: 02-719-5951

NEC do Brasil S.A.
Cumbica-Guarulhos-SP, Brasil
Tel: 011-6465-6810
Fax: 011-6465-6829

J97. 8

[MEMO]

Reference: Electrical Characteristics for Microcomputer (IEI-601)

Note that "preliminary" is not indicated in this document, even though the related documents may be preliminary versions.

V800 series, V830, and V830 family are trademarks of NEC Corporation.

M4 96.5