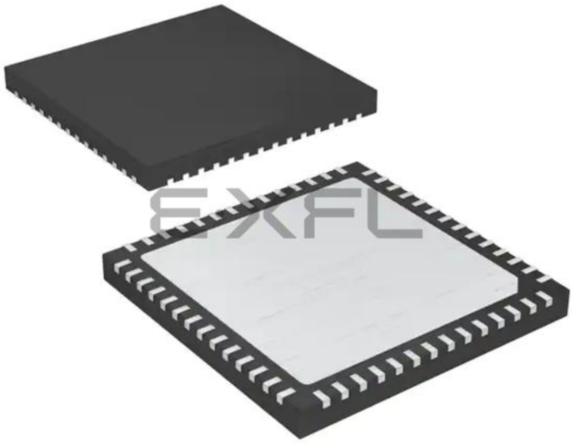
# Renesas Electronics America Inc - UPD78F8042K8(R)-9B4-AX Datasheet



**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| **Details** | |
|---|---|
| Product Status | Obsolete |
| Core Processor | 78K/0 |
| Core Size | 16-Bit |
| Speed | 20MHz |
| Connectivity | CSI, I²C, LINbus, UART/USART |
| Peripherals | LVD, POR, PWM, WDT |
| Number of I/O | 23 |
| Program Memory Size | 96KB (96K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 6K x 8 |
| Voltage - Supply (Vcc/Vdd) | 3V ~ 5.5V |
| Data Converters | A/D 6x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 56-WFQFN Exposed Pad |
| Supplier Device Package | - |
| Purchase URL | https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f8042k8-r-9b4-ax |

## 3.3 Pin I/O Circuits and Recommended Connection of Unused Pins

Table 3-3 shows the types of pin I/O circuits and the recommended connections of unused pins.

**Table 3-3. Connection of Unused Pins (1/2)**

| Pin Name | I/O Circuit Type | I/O | Recommended Connection of Unused Pins |
|---|---|---|---|
| P05/TI05/TO05 | 8-R | I/O | Input: Independently connect to $V_{DD}$/$EV_{DD}$ or $V_{SS}$/$EV_{SS}$ via a resistor.<br>Output: Leave open. |
| P13/TxD3 | 5-AG | | |
| P14/RxD3 | 8-R | | |
| P16/TI01/TO01/INTP5 | 8-R | | |
| P17/TI02/TO02 | | | |
| P26/ANI6, P27/ANI7[Note 1] | 11-G | | Input: Independently connect to $AV_{REF}$ or $AV_{SS}$ via a resistor.<br>Output: Leave open. |
| P31/TI03/TO03/INTP4 | 8-R | | Input: Independently connect to $V_{DD}$/$EV_{DD}$ or $V_{SS}$/$EV_{SS}$ via a resistor.<br>Output: Leave open. |
| P40/TOOL0 | | | <When on-chip debugging is enabled><br>Pull this pin up (pulling it down is prohibited).<br><When on-chip debugging is disabled><br>Input: Independently connect to $V_{DD}$/$EV_{DD}$ or $V_{SS}$/$EV_{SS}$ via a resistor.<br>Output: Leave open. |
| P41/TOOL1 | 5-AG | | Input: Independently connect to $V_{DD}$/$EV_{DD}$ or $V_{SS}$/$EV_{SS}$ via a resistor.<br>Output: Leave open. |
| P60/SCL0 | 13-R | | Input: Connect to $V_{SS}$/$EV_{SS}$.<br>Output: Set the port output latch to 0 and leave these pins open via low-level output. |
| P61/SDA0 | | | |
| P65/TI11/TO11 | 8-R | | Input: Independently connect to $V_{DD}$/$EV_{DD}$ or $V_{SS}$/$EV_{SS}$ via a resistor.<br>Output: Leave open. |
| P67/TI13/TO13 | | | |
| P120/INTP0/EXLVI | 8-R | | |
| P121/X1[Note 2] | 37-B | Input | Independently connect to $V_{DD}$/$EV_{DD}$ or $V_{SS}$/$EV_{SS}$ via a resistor. |
| P142/$\overline{SCK20}$/SCL20 | 5-AN | I/O | Input: Independently connect to $V_{DD}$/$EV_{DD}$ or $V_{SS}$/$EV_{SS}$ via a resistor.<br>Output: Leave open. |
| P143/SI20/RxD2/SDA20 | | | |
| P144/SO20/TxD2 | 5-AG | | |
| P150/ANI8 to P153/ANI11[Note 1] | 11-G | | Input: Independently connect to $AV_{REF}$ or $AV_{SS}$ via a resistor.<br>Output: Leave open. |

**Notes 1.** P26/ANI6, P27/ANI7, and P150/ANI8 to P153/ANI11 are set in the digital input port mode after release of reset.

**2.** Use recommended connection above in input port mode (see **Figure 7-2 Format of Clock Operation Mode Control Register (CMC)**) when these pins are not used.

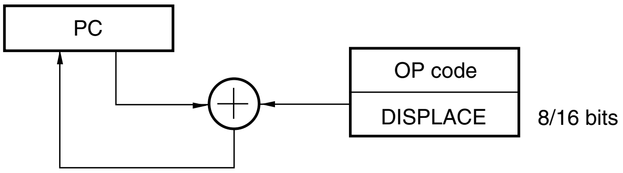
## 5.3 Instruction Address Addressing

### 5.3.1 Relative addressing

**[Function]**

Relative addressing stores in the program counter (PC) the result of adding a displacement value included in the instruction word (signed complement data: −128 to +127 or −32768 to +32767) to the program counter (PC)'s value (the start address of the next instruction), and specifies the program address to be used as the branch destination. Relative addressing is applied only to branch instructions.

**Figure 5-16. Outline of Relative Addressing**



### 5.3.2 Immediate addressing

**[Function]**

Immediate addressing stores immediate data of the instruction word in the program counter, and specifies the program address to be used as the branch destination.

For immediate addressing, CALL !!addr20 or BR !!addr20 is used to specify 20-bit addresses and CALL !addr16 or BR !addr16 is used to specify 16-bit addresses. 0000 is set to the higher 4 bits when specifying 16-bit addresses.

**Figure 5-17. Example of CALL !!addr20/BR !!addr20**

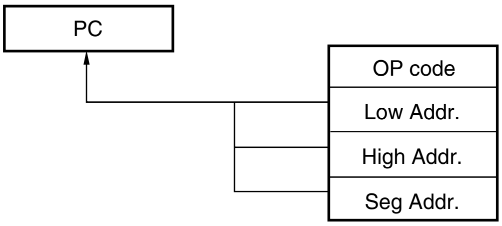
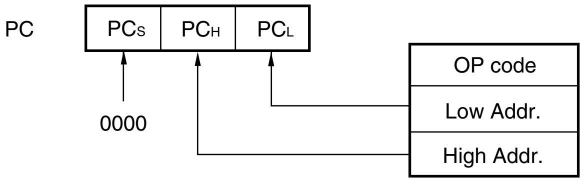


**Figure 5-18. Example of CALL !addr16/BR !addr16**



### 5.4.7 Based addressing

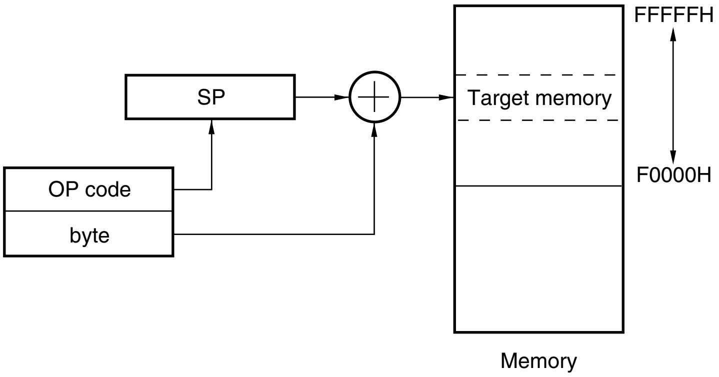
**[Function]**

Based addressing uses the contents of a register pair specified with the instruction word as a base address, and 8-bit immediate data or 16-bit immediate data as offset data. The sum of these values is used to specify the target address.

**[Operand format]**

| Identifier | Description |
|---|---|
| – | [HL + byte], [DE + byte], [SP + byte] (only the space from F0000H to FFFFFH is specifiable) |
| – | word[B], word[C] (only the space from F0000H to FFFFFH is specifiable) |
| – | word[BC] (only the space from F0000H to FFFFFH is specifiable) |
| – | ES:[HL + byte], ES:[DE + byte] (higher 4-bit addresses are specified by the ES register) |
| – | ES:word[B], ES:word[C] (higher 4-bit addresses are specified by the ES register) |
| – | ES:word[BC] (higher 4-bit addresses are specified by the ES register) |

**Figure 5-29. Example of [SP+byte]**



### 6.2.4 Port 3

Port 3 is a 1-bit I/O port with an output latch. Port 3 can be set to the input mode or output mode using port mode register 3 (PM3). When the P31 pin is used as an input port, use of an on-chip pull-up resistor can be specified by pull-up resistor option register 3 (PU3).

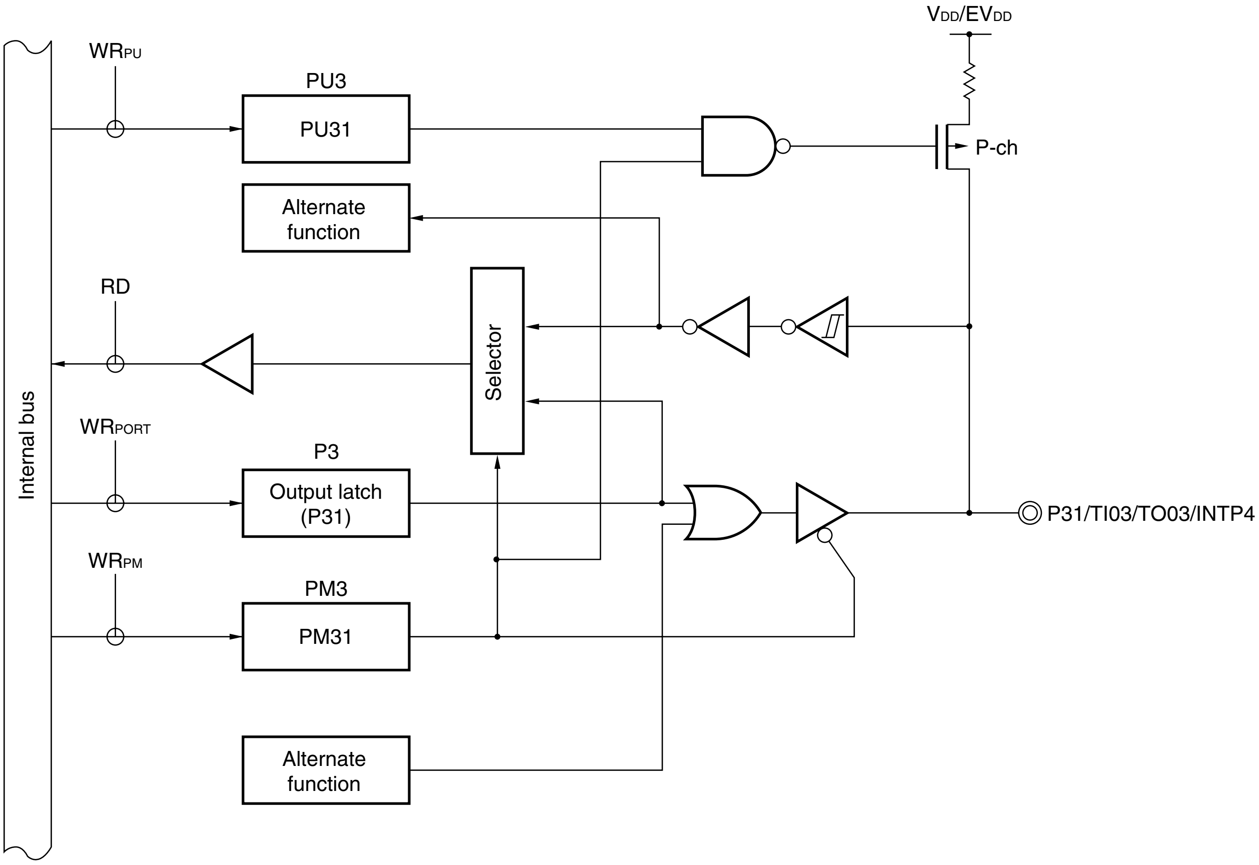
This port can also be used for external interrupt request input and timer I/O.

Reset signal generation sets port 3 to input mode.

Figure 6-8 shows block a diagram of port 3.

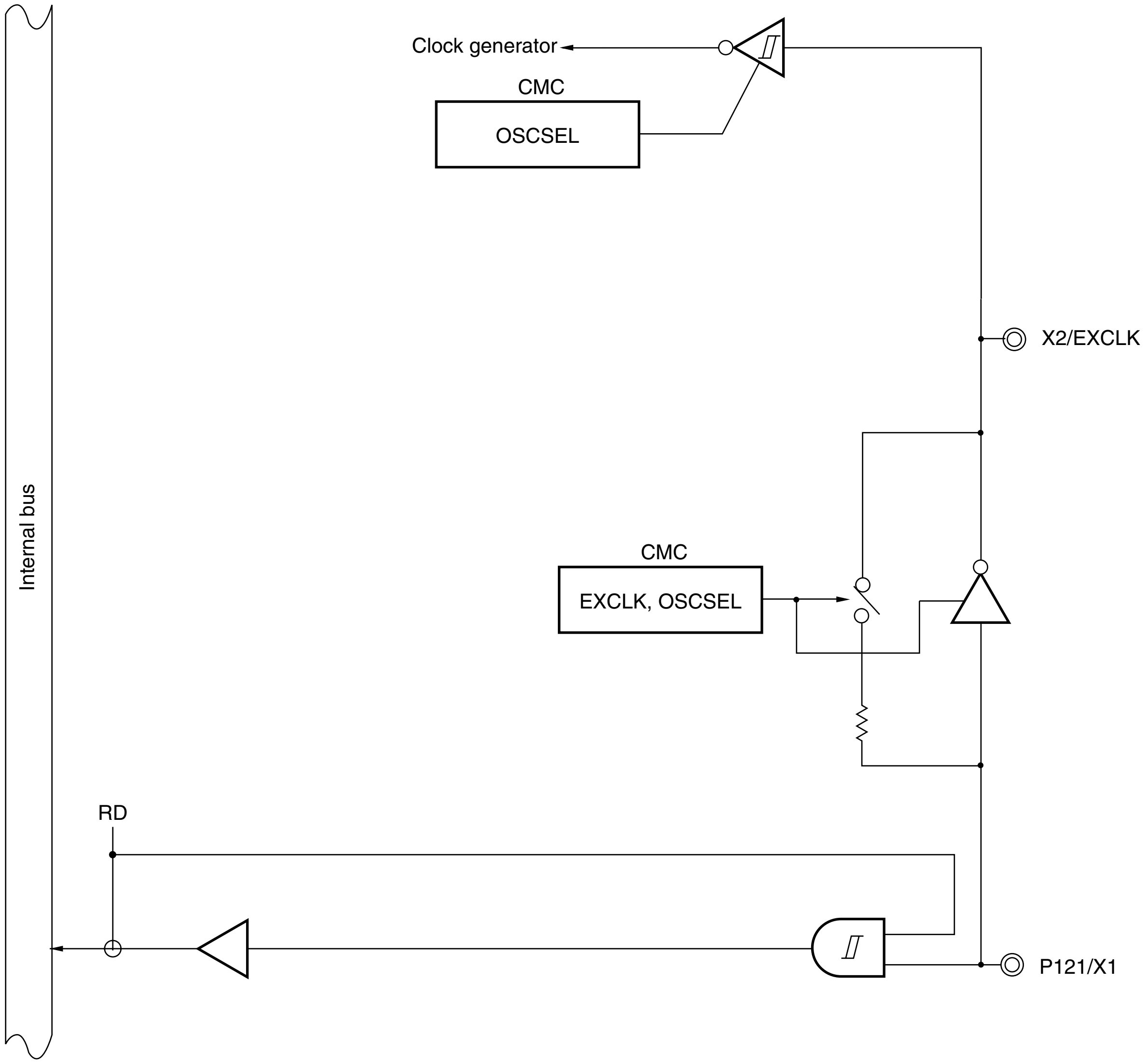
**Cautions 1. To use P31/TI03/TO03/INTP4 as a general-purpose port, set bit 3 (TO03) of timer output register 0 (TO0) and bit 3 (TOE03) of timer output enable register 0 (TOE0) to "0", which is the same as their default status setting.**

**Figure 6-8. Block Diagram of P31**



| | |
|---|---|
| P3: | Port register 3 |
| PU3: | Pull-up resistor option register 3 |
| PM3: | Port mode register 3 |
| RD: | Read signal |
| WR××: | Write signal |

**Figure 6-15. Block Diagram of P121**



| | |
|---|---|
| CMC: | Clock operation mode control register |
| RD: | Read signal |

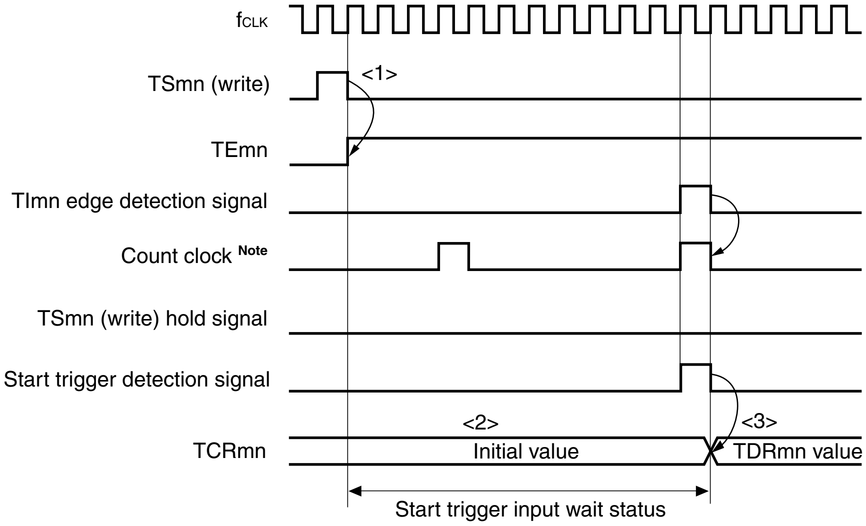
### (d) Start timing in one-count mode

<1> Operation is enabled (TEmn = 1) by writing 1 to the TSmn.

<2> Enters the start trigger input wait status, and timer/counter register mn (TCRmn) holds the initial value.

<3> On start trigger detection, the value of timer data register mn (TDRmn) is loaded to the TCRmn register and count starts.

**Figure 8-15. Start Timing (In One-count Mode)**



**Note** When the one-count mode is set, the operation clock ($f_{MCK}$) is selected as count clock (CCSmn = 0).

**Caution An input signal sampling error is generated since operation starts upon start trigger detection (If the TImn pin input signal is used as a start trigger, an error of one count clock occurs.).**

**Figure 8-52. Operation Procedure When Input Signal High-/Low-Level Width Measurement Function Is Used**

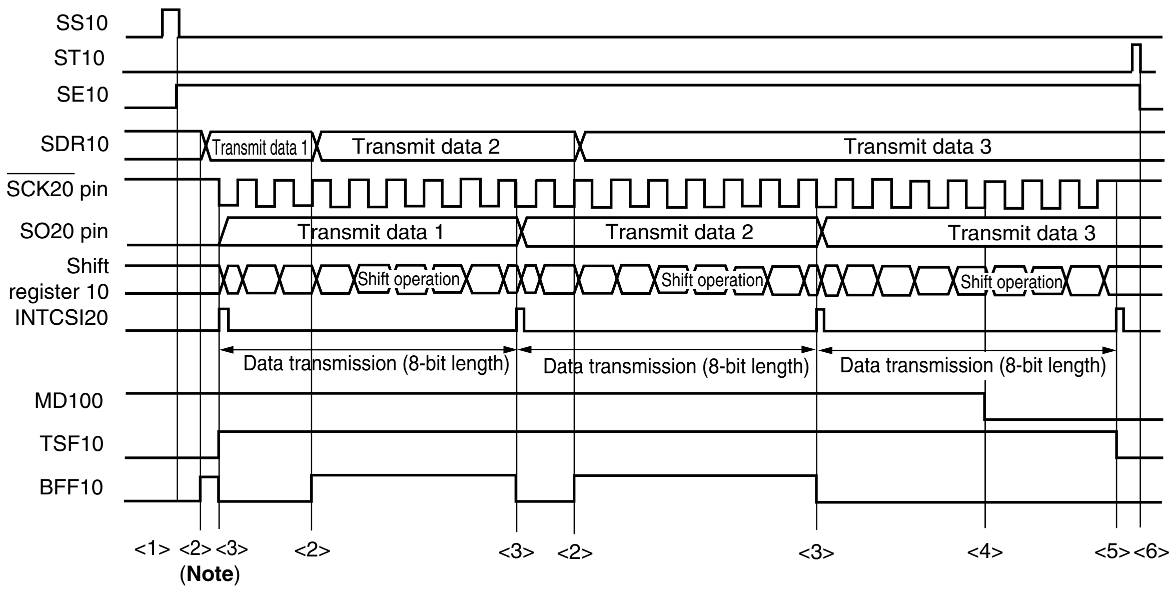
| | Software Operation | Hardware Status |
|---|---|---|
| TAU default setting | | Power-off status<br>(Clock supply is stopped and writing to each register is disabled.) |
| | Sets the TAU0EN and TAU1EN bits of peripheral enable registers 0, 2 (PER0, PER2) to 1. → | Power-on status. Each channel stops operating.<br>(Clock supply is started and writing to each register is enabled.) |
| | Sets timer clock select register m (TPSm).<br>Determines clock frequencies of CKm0 and CKm1. | |
| Channel default setting | Sets timer mode register mn (TMRmn) (determines operation mode of channel).<br>Clears the TOEmn bit to 0 and stops operation of TOmn. | Channel stops operating.<br>(Clock is supplied and some power is consumed.) |
| Operation start | Sets the TSmn bit to 1. →<br>The TSmn bit automatically returns to 0 because it is a trigger bit. | TEmn = 1, and the TImn pin start edge detection wait status is set. |
| | Detects TImn pin input count start valid edge. → | Clears timer/counter register mn (TCRmn) to 0000H and starts counting up. |
| During operation | Set value of the TDRmn register can be changed.<br>The TCRmn register can always be read.<br>The TSRmn register is not used.<br>Set values of the TMRmn register, TOMmn, TOLmn, TOmn, and TOEmn bits cannot be changed. | When the TImn pin start edge is detected, the counter (TCRmn) counts up from 0000H. If a capture edge of the TImn pin is detected, the count value is transferred to timer data register mn (TDRmn) and INTTMmn is generated.<br>If an overflow occurs at this time, the OVF bit of timer status register mn (TSRmn) is set; if an overflow does not occur, the OVF bit is cleared. The TCRmn register stops the count operation until the next TImn pin start edge is detected. |
| Operation stop | The TTmn bit is set to 1. →<br>TTmn bit automatically returns to 0 because it is a trigger bit. | TEmn = 0, and count operation stops.<br>TCRmn register holds count value and stops.<br>The OVF bit of the TSRmn register is also held. |
| TAU stop | The TAU0EN and TAU1EN bits of PER0 register is cleared to 0. → | Power-off status<br>All circuits are initialized and SFR of each channel is also initialized. |

Operation is resumed.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 7),
mn = 00 to 07, 10 to 13 (This is, however, mn = 01 to 03, 05, 11, and 13 in the case of the timer input pin (TImn).)

**(4) Processing flow (in continuous transmission mode)**

**Figure 11-30. Timing Chart of Master Transmission (in Continuous Transmission Mode) (Type 1: DAP10 = 0, CKP10 = 0)**



**Note** If transmit data is written to the SDR10 register while the BFF10 bit is 1 (valid data is stored in serial data register 10 (SDR10)), the transmit data is overwritten.

**Caution The MD100 bit can be rewritten even during operation.**
**However, rewrite it before transfer of the last bit is started, so that it will be rewritten before the transfer end interrupt of the last transmit data.**

### (1) Register setting

**Figure 11-62. Example of Contents of Registers for Slave Transmission/Reception of 3-Wire Serial I/O (CSI20)**

**(a) Serial mode register 10 (SMR10)**

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SMR10 | CKS10<br>0/1 | CCS10<br>1 | 0 | 0 | 0 | 0 | 0 | STS10<br>0 | 0 | SIS100<br>0 | 1 | 0 | 0 | MD102<br>0 | MD101<br>0 | MD100<br>0/1 |

Operation clock ($f_{MCK}$) of channel 0
0: Prescaler output clock CK10 set by the SPS1 register
1: Prescaler output clock CK11 set by the SPS1 register

Interrupt source of channel 0
0: Transfer end interrupt
1: Buffer empty interrupt

**(b) Serial communication operation setting register 10 (SCR10)**

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SCR10 | TXE10<br>1 | RXE10<br>1 | DAP10<br>0/1 | CKP10<br>0/1 | 0 | EOC10<br>0 | PTC101<br>0 | PTC100<br>0 | DIR10<br>0/1 | 0 | SLC101<br>0 | SLC100<br>0 | 0 | DLS102<br>1 | DLS101<br>1 | DLS100<br>0/1 |

Selection of the data and clock phase (For details about the setting, see **11.3 Registers Controlling Serial Array Unit**.)

Selection of data transfer sequence
0: Inputs/outputs data with MSB first
1: Inputs/outputs data with LSB first.

Setting of data length
0: 7-bit data length
1: 8-bit data length

**(c) Serial data register 10 (SDR10) (lower 8 bits: SIO20)**

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SDR10 | 0000000<br>(baud rate setting) | | | | | | | 0 | Transmit data setting/receive data register | | | | | | | |
| | | | | | | | | | SIO20 | | | | | | | |

**(d) Serial output register 1 (SO1) ... Sets only the bits of the target channel.**

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SO1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | CKO10<br>× | 0 | 0 | 0 | 0 | 1 | 1 | 1 | SO10<br>0/1 |

**(e) Serial output enable register 1 (SOE1) ... Sets only the bits of the target channel to 1.**

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SOE1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | SOE10<br>0/1 |

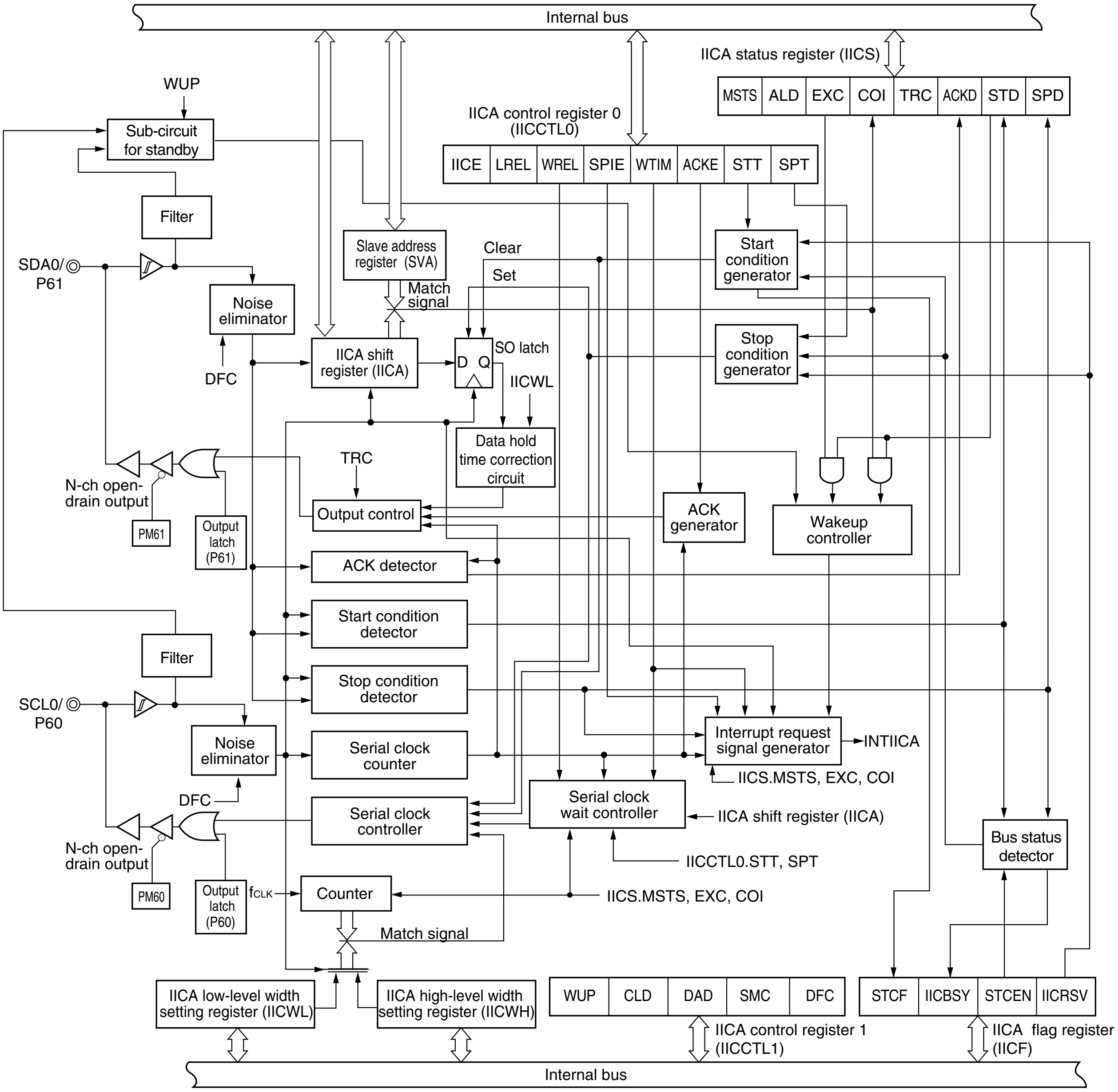
**(f) Serial channel start register 1 (SS1) ... Sets only the bits of the target channel to 1.**

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SS1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | SS11<br>× | SS10<br>0/1 |

**Caution Be sure to set transmit data to the SIO20 register before the clock from the master is started.**

**Remark** □ : Setting is fixed in the CSI slave transmission/reception mode, ▨ : Setting disabled (set to the initial value)
×: Bit that cannot be used in this mode (set to the initial value when not used in any mode)
0/1: Set to 0 or 1 depending on the usage of the user

**Figure 12-1. Block Diagram of Serial Interface IICA**



### 12.5.9 Address match detection method

In I²C bus mode, the master device can select a particular slave device by transmitting the corresponding slave address.

Address match can be detected automatically by hardware. An interrupt request (INTIICA) occurs when the address set to the slave address register (SVA) matches the slave address sent by the master device, or when an extension code has been received.

### 12.5.10 Error detection

In I²C bus mode, the status of the serial data bus (SDA0) during data transmission is captured by the IICA shift register (IICA) of the transmitting device, so the IICA data prior to transmission can be compared with the transmitted IICA data to enable detection of transmission errors. A transmission error is judged as having occurred when the compared data values do not match.

### 12.5.11 Extension code

(1) When the higher 4 bits of the receive address are either "0000" or "1111", the extension code reception flag (EXC) is set to 1 for extension code reception and an interrupt request (INTIICA) is issued at the falling edge of the eighth clock. The local address stored in the slave address register (SVA) is not affected.

(2) The settings below are specified if 11110xx0 is transferred from the master by using a 10-bit address transfer when the SVA register is set to 11110xx0. Note that INTIICA occurs at the falling edge of the eighth clock.

- Higher four bits of data match: EXC = 1
- Seven bits of data match: COI = 1

**Remark** EXC: Bit 5 of IICA status register (IICS)
COI: Bit 4 of IICA status register (IICS)

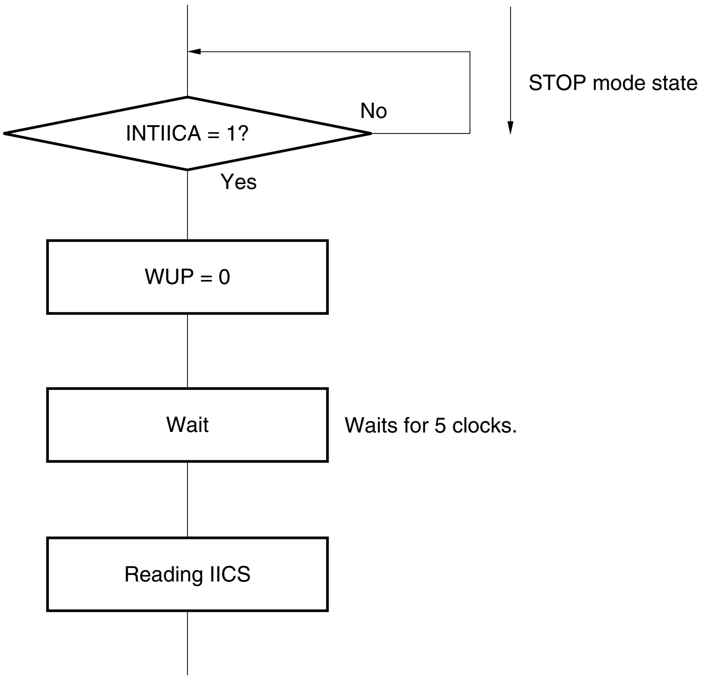
(3) Since the processing after the interrupt request occurs differs according to the data that follows the extension code, such processing is performed by software.
If the extension code is received while a slave device is operating, then the slave device is participating in communication even if its address does not match.
For example, after the extension code is received, if you do not wish to operate the target device as a slave device, set bit 6 (LREL) of the IICA control register 0 (IICCTL0) to 1 to set the standby mode for the next communication operation.

**Table 12-3. Bit Definitions of Major Extension Codes**

| Slave Address | R/W Bit | Description |
|---|---|---|
| 0 0 0 0 0 0 0 | 0 | General call address |
| 1 1 1 1 0 x x | 0 | 10-bit slave address specification (during address authentication) |
| 1 1 1 1 0 x x | 1 | 10-bit slave address specification (after address match, when read command is issued) |

**Remark** See the I²C bus specifications issued by NXP Semiconductors for details of extension codes other than those described above.

**Figure 12-23. Flow When Setting WUP = 0 upon Address Match (Including Extension Code Reception)**

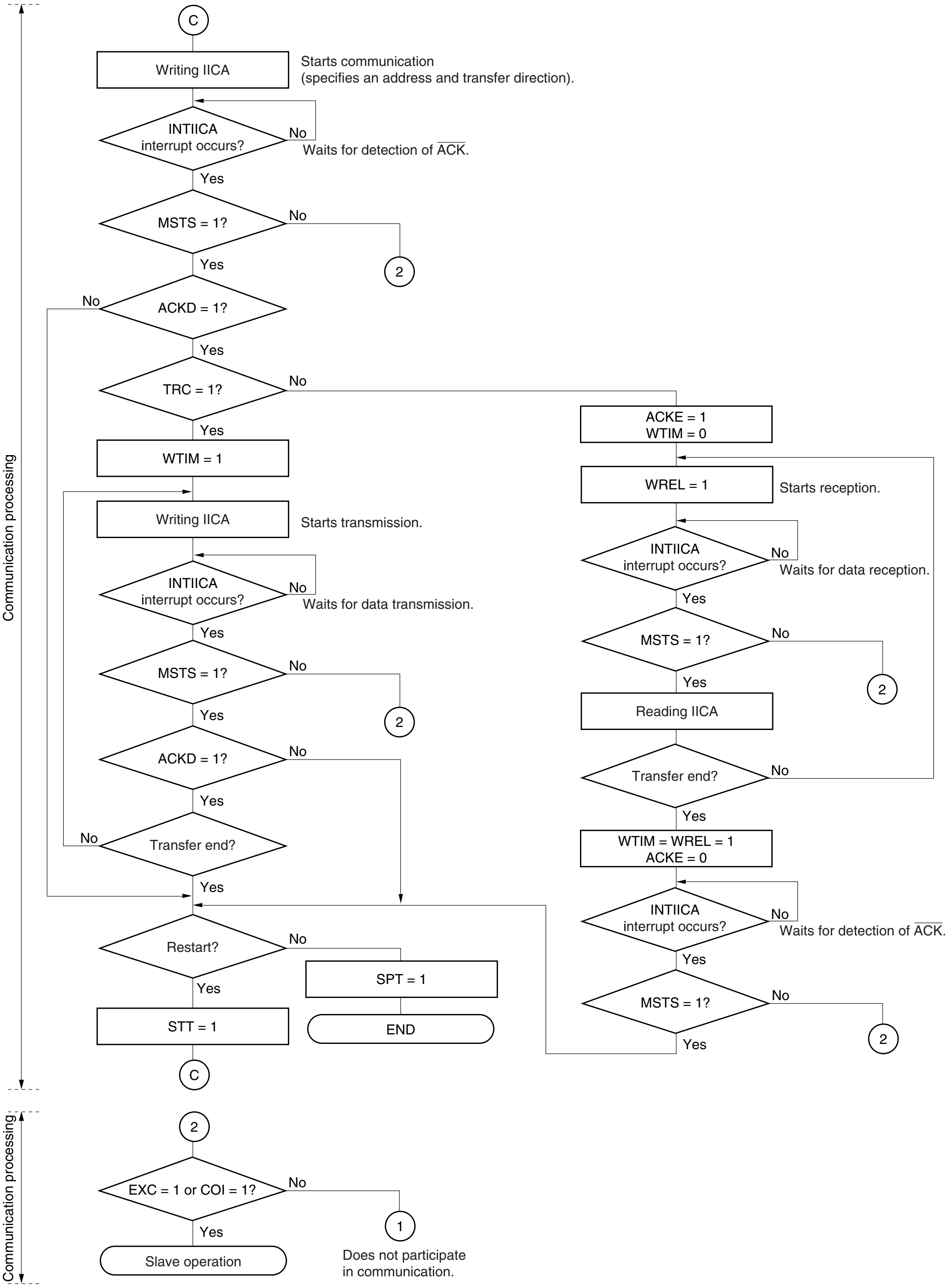


Executes processing corresponding to the operation to be executed after checking the operation state of serial interface IICA.

Use the following flows to perform the processing to release the STOP mode other than by an interrupt request (INTIICA) generated from serial interface IICA.

- Master device operation: Flow shown in Figure 12-24
- Slave device operation: Same as the flow in Figure 12-23

**Figure 12-29. Master Operation in Multi-Master System (3/3)**



**Remarks 1.** Conform to the specifications of the product that is communicating, with respect to the transmission and reception formats.

**2.** To use the device as a master in a multi-master system, read the MSTS bit each time interrupt INTIICA has occurred to check the arbitration result.

**3.** To use the device as a slave in a multi-master system, check the status by using the IICA status register (IICS) and IICA flag register (IICF) each time interrupt INTIICA has occurred, and determine the processing to be performed next.

The meanings of <1> to <6> in (1) Start condition ~ address ~ data in Figure 12-32 are explained below.

<1> The start condition trigger is set by the master device (STT = 1) and a start condition (SDA0 = 0 and SCL0 = 1) is generated once the bus data line goes low (SDA0 = 0). When the start condition is subsequently detected, the master device enters the master device communication status (MSTS = 1). The master device is ready to communicate once the bus clock line goes low (SCL0 = 0) after the hold time has elapsed.

<2> The master device writes the address + W (transmission) to the IICA shift register (IICA) and transmits the slave address.

<3> If the address received matches the address of a slave device[Note], that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (ACKD = 1) at the rising edge of the 9th clock.

<4> The master device issues an interrupt (INTIICA: end of address transmission) at the falling edge of the 9th clock, and the slave device whose address matched the transmitted slave address also issues an interrupt (INTIICA: address match). The master device and slave device also set a wait status (SCL0 = 0)[Note] when the addresses match.

<5> The master device writes the data to transmit to the IICA register and releases the wait status that it set by the master device.

<6> If the slave device releases the wait status (WREL = 1), the master device starts transferring data to the slave device.

**Note** If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK: SDA0 = 1). The slave device also does not issue the INTIICA interrupt (address match) and does not set a wait status. The master device, however, issues the INTIICA interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

**Remark** <1> to <15> in Figure 12-32 represent the entire procedure for communicating data using the I$^2$C bus. Figure 12-32 (1) Start condition ~ address ~ data shows the processing from <1> to <6>, Figure 12-32 (2) Address ~ data ~ data shows the processing from <3> to <10>, and Figure 12-32 (3) Data ~ data ~ stop condition shows the processing from <7> to <15>.

## 14.5 Example of Setting of DMA Controller

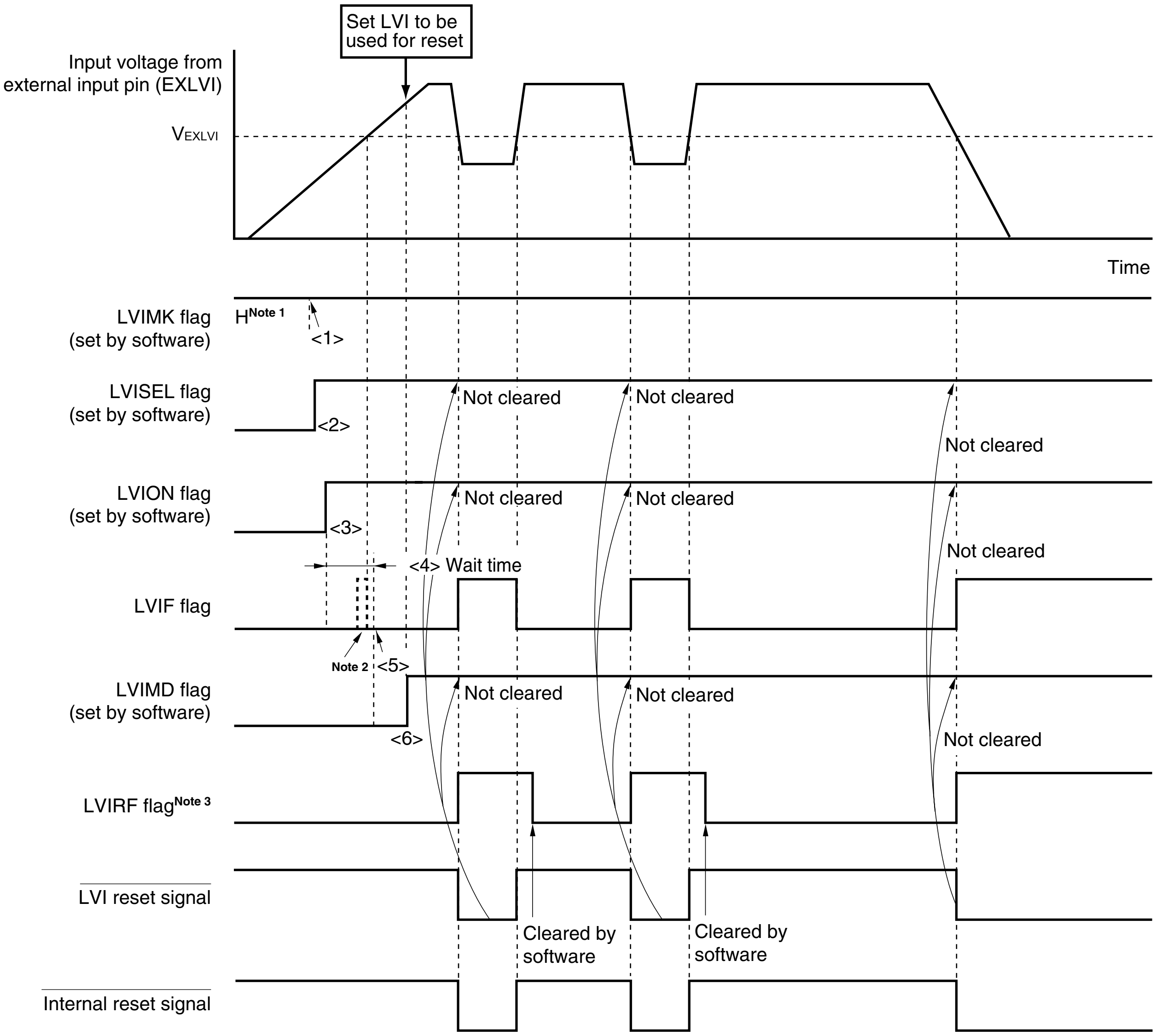
### 14.5.1 Consecutive capturing of A/D conversion results

A flowchart of an example of setting for consecutively capturing A/D conversion results is shown below.

- Consecutive capturing of A/D conversion results.
- DMA channel 1 is used for DMA transfer.
- DMA start source: INTAD
- Interrupt of A/D is specified by IFC13 to IFC10 = 1100B.
- Transfers FFF1EH and FFF1FH (2 bytes) of the 10-bit A/D conversion result register to 512 bytes of FFCE0H to FFEDFH of RAM.

**Remark** IFC13 to IFC10: Bits 3 to 0 of DMA mode control registers 1 (DMC1)

**Figure 19-7. Timing of Low-Voltage Detector Internal Reset Signal Generation (Bit: LVISEL = 1)**



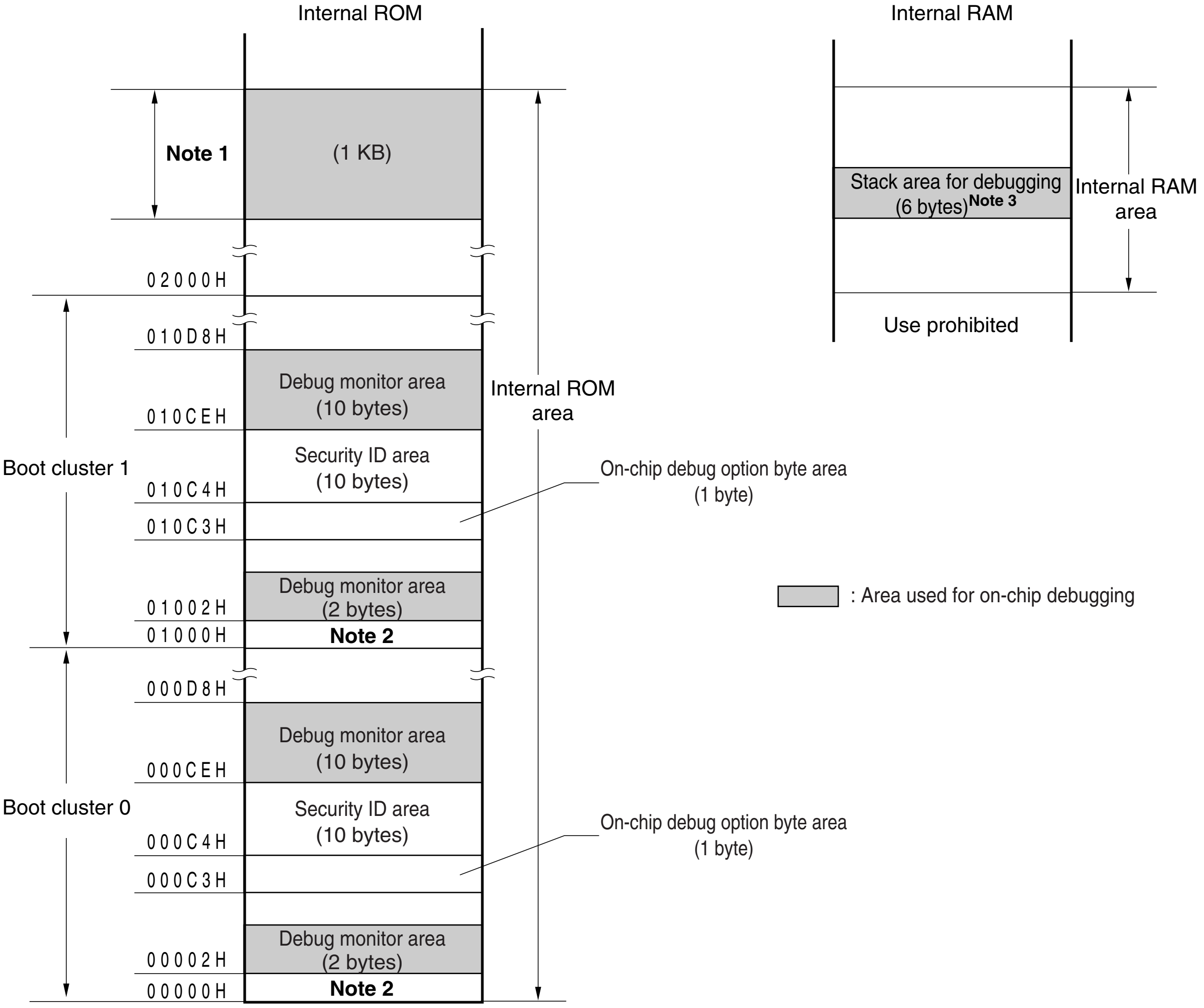
**Notes 1.** The LVIMK flag is set to “1” by reset signal generation.

**2.** The LVIIF flag of the interrupt request flag registers and the LVIF flag may be set (1).

**3.** LVIRF is bit 0 of the reset control flag register (RESF). For details of RESF, see **CHAPTER 17 RESET FUNCTION**.

**Remark** <1> to <6> in Figure 19-7 above correspond to <1> to <6> in the description of “When starting operation” in **19.4.1 (2) When detecting level of input voltage from external input pin (EXLVI)**.

**Figure 23-2. Memory Spaces Where Debug Monitor Programs Are Allocated**



**Notes 1.** Address differs depending on products as follows.

| Products | Internal ROM | Address |
|---|---|---|
| μPD78F8040 | 32 KB | 07C00H to 07FFFH |
| μPD78F8041 | 64 KB | 0FC00H to 0FFFFH |
| μPD78F8042 | 96 KB | 17C00H to 17FFFH |
| μPD78F8043 | 128 KB | 1FC00H to 1FFFFH |

**2.** In debugging, reset vector is rewritten to address allocated to a monitor program.

**3.** Since this area is allocated immediately before the stack area, the address of this area varies depending on the stack increase and decrease. That is, 6 extra bytes are consumed for the stack area used.

For details of the way to secure of the memory space, refer to the **QB-MINI2 On-Chip Debug Emulator with Programming Function User’s Manual (U18371E)**.

## 25.1 Conventions Used in Operation List

### 25.1.1 Operand identifiers and specification methods

Operands are described in the “Operand” column of each instruction in accordance with the description method of the instruction operand identifier (refer to the assembler specifications for details). When there are two or more description methods, select one of them. Alphabetic letters in capitals and the symbols, #, !, !!, $, $!, [ ], and ES: are keywords and are described as they are. Each symbol has the following meaning.

- #: Immediate data specification
- !: 16-bit absolute address specification
- !!: 20-bit absolute address specification
- $: 8-bit relative address specification
- $!: 16-bit relative address specification
- [ ]: Indirect address specification
- ES:: Extension address specification

In the case of immediate data, describe an appropriate numeric value or a label. When using a label, be sure to describe the #, !, !!, $, $!, [ ], and ES: symbols.

For operand register identifiers, r and rp, either function names (X, A, C, etc.) or absolute names (names in parentheses in the table below, R0, R1, R2, etc.) can be used for description.

**Table 25-1. Operand Identifiers and Specification Methods**

| Identifier | Description Method |
|---|---|
| r<br>rp<br>sfr<br>sfrp | X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7)<br>AX (RP0), BC (RP1), DE (RP2), HL (RP3)<br>Special-function register symbol (SFR symbol) FFF00H to FFFFFH<br>Special-function register symbol (16-bit manipulatable SFR symbol. Even addresses only^Note) FFF00H to FFFFFH |
| saddr<br>saddrp | FFE20H to FFF1FH Immediate data or labels<br>FFE20H to FF1FH Immediate data or labels (even addresses only^Note) |
| addr20<br>addr16<br>addr5 | 00000H to FFFFFH Immediate data or labels<br>0000H to FFFFH Immediate data or labels (only even addresses for 16-bit data transfer instructions^Note)<br>0080H to 00BFH Immediate data or labels (even addresses only) |
| word<br>byte<br>bit | 16-bit immediate data or label<br>8-bit immediate data or label<br>3-bit immediate data or label |
| RBn | RB0 to RB3 |

**Note** Bit 0 = 0 when an odd address is specified.

**Remark** The special function registers can be described to operand sfr as symbols. See **Table 5-5 SFR List** for the symbols of the special function registers.

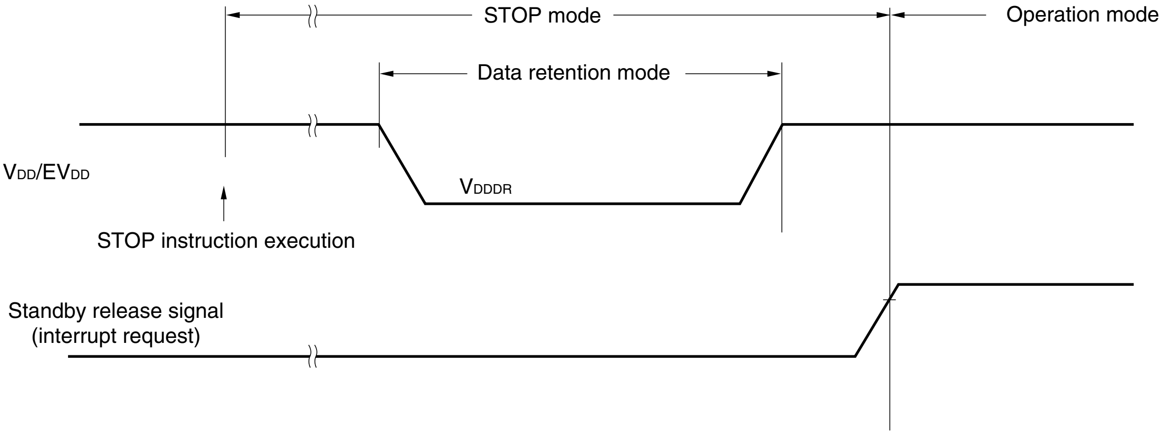
The extended special function registers can be described to operand !addr16 as symbols. See **Table 5-6 Extended SFR (2nd SFR) List** for the symbols of the extended special function registers.

## 26.7 Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics

**($T_A$ = –40 to +85°C)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| Data retention supply voltage | $V_{DDDR}$ | | 1.5[Note] | | 5.5 | V |

**Note** The value depends on the POC detection voltage. When the voltage drops, the data is retained before a POC reset is effected, but data is not retained when a POC reset is effected.



## 26.8 Flash Memory Programming Characteristics

**($T_A$ = –40 to +85°C, 3.0 V ≤ $V_{DD}$/$EV_{DD}$ ≤ 5.5 V, 4.75 V ≤ $IV_{DD}$ ≤ 5.25 V, $V_{SS}$/$EV_{SS}$ = GND1 = GND2 = 0 V)**

| Parameter | Symbol | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| $V_{DD}$/$EV_{DD}$ supply current | $I_{DD}$ | Typ. = 10 MHz, Max. = 20 MHz | | | 6 | 20 | mA |
| CPU/peripheral hardware clock frequency | $f_{CLK}$ | 3.0 V ≤ $V_{DD}$/$EV_{DD}$ ≤ 5.5 V | | 2 | | 20 | MHz |
| Number of rewrites (number of deletes per block) | $C_{erwr}$ | Used for updating programs<br>When using flash memory programmer and Renesas Electronics self programming library | Retained for 15 years | 1,000 | | | Times |
| | | Used for updating data<br>When using Renesas Electronics EEPROM emulation library | Retained for 5 years | 10,000 | | | Times |

**Remark** When updating data multiple times, use the flash memory as one for updating data.