Renesas Electronics America Inc - UPD78F0078GC-8BS-A Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

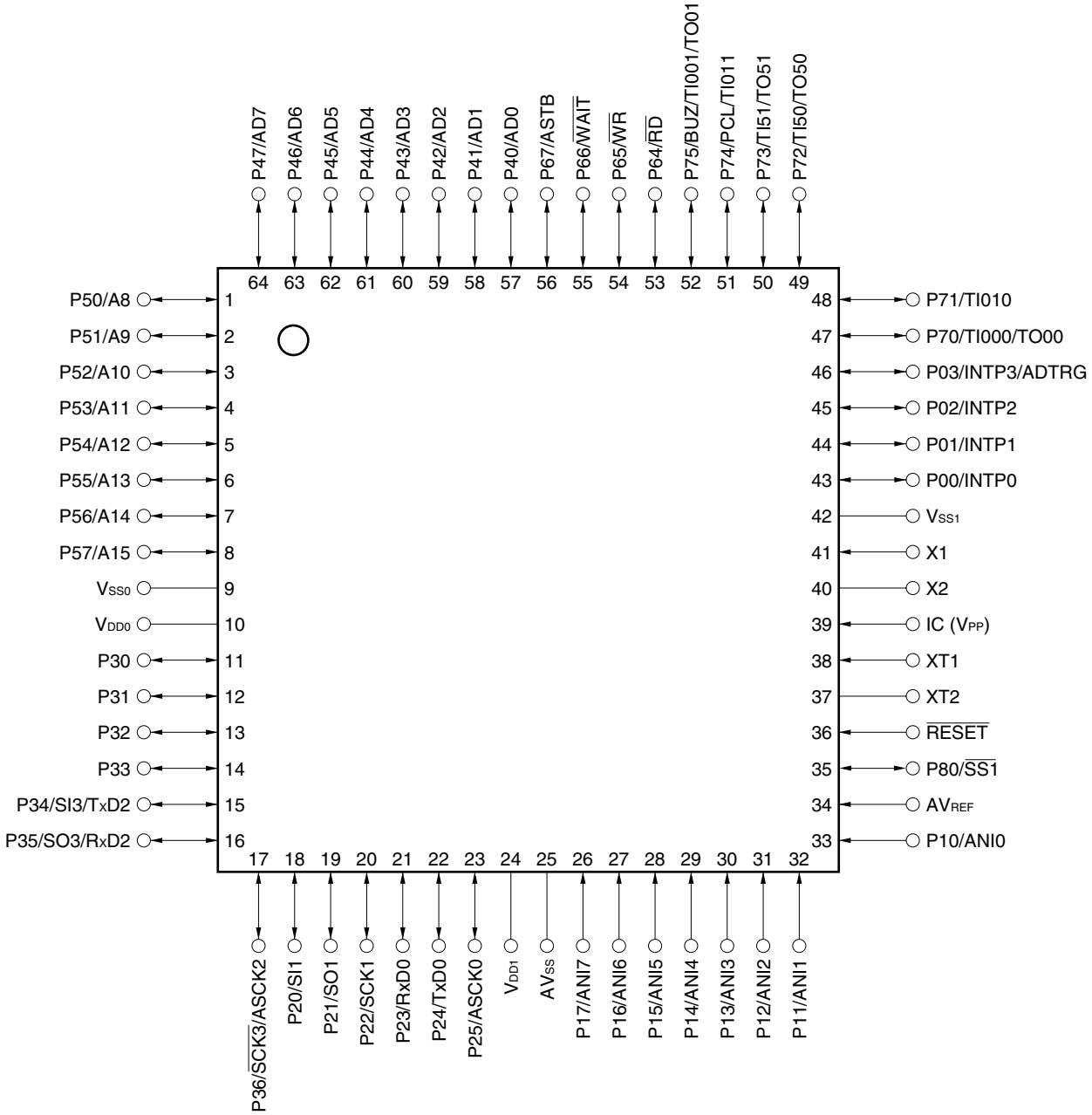
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	-
Core Size	-
Speed	-
Connectivity	-
Peripherals	-
Number of I/O	-
Program Memory Size	-
Program Memory Type	-
EEPROM Size	-
RAM Size	-
Voltage - Supply (Vcc/Vdd)	-
Data Converters	-
Oscillator Type	-
Operating Temperature	-
Mounting Type	-
Package / Case	-
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f0078gc-8bs-a

1.5 Pin Configuration (Top View)

- **64-pin plastic LQFP (14 × 14)**
- **64-pin plastic QFP (14 × 14)**
- **64-pin plastic TQFP (12 × 12)**



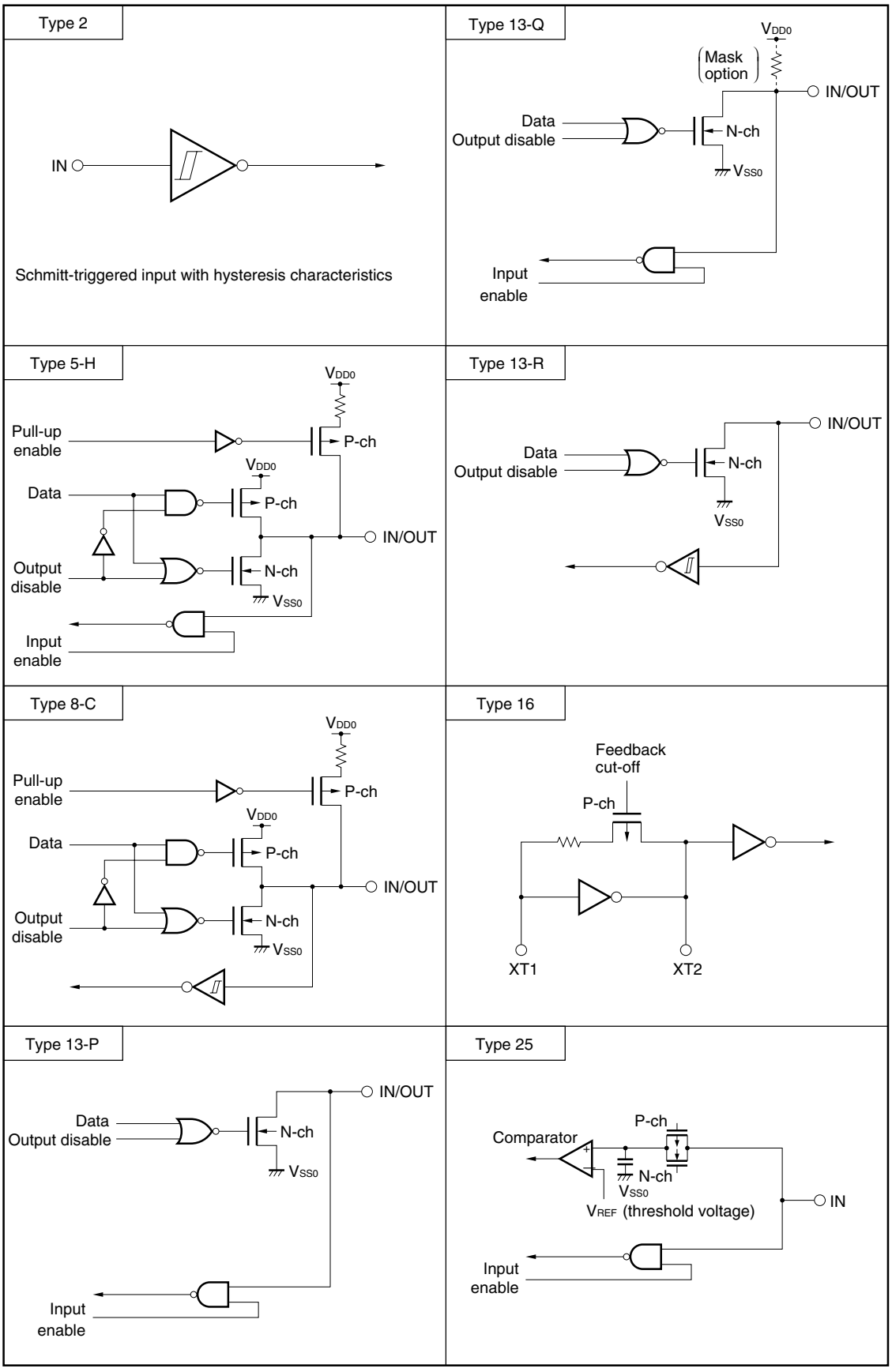
Cautions 1. Connect the IC (internally connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

Remarks 1. When these devices are used in applications that require the reduction of noise generated from an on-chip microcontroller, the implementation of noise measures is recommended, such as supplying V_{DD0} and V_{DD1} independently, connecting V_{SS0} and V_{SS1} independently to ground lines, and so on.

2. Pin connection in parentheses is intended for the μPD78F0078.

Figure 4-1. Pin I/O Circuits



6.2.7 Port 5

Port 5 is an 8-bit I/O port with an output latch. Port 5 can be set to the input or output mode in 1-bit units using port mode register 5 (PM5). An on-chip pull-up resistor can be connected to P50 to P57 in 1-bit units using pull-up resistor option register 5 (PU5).

Port 5 can drive LEDs directly.

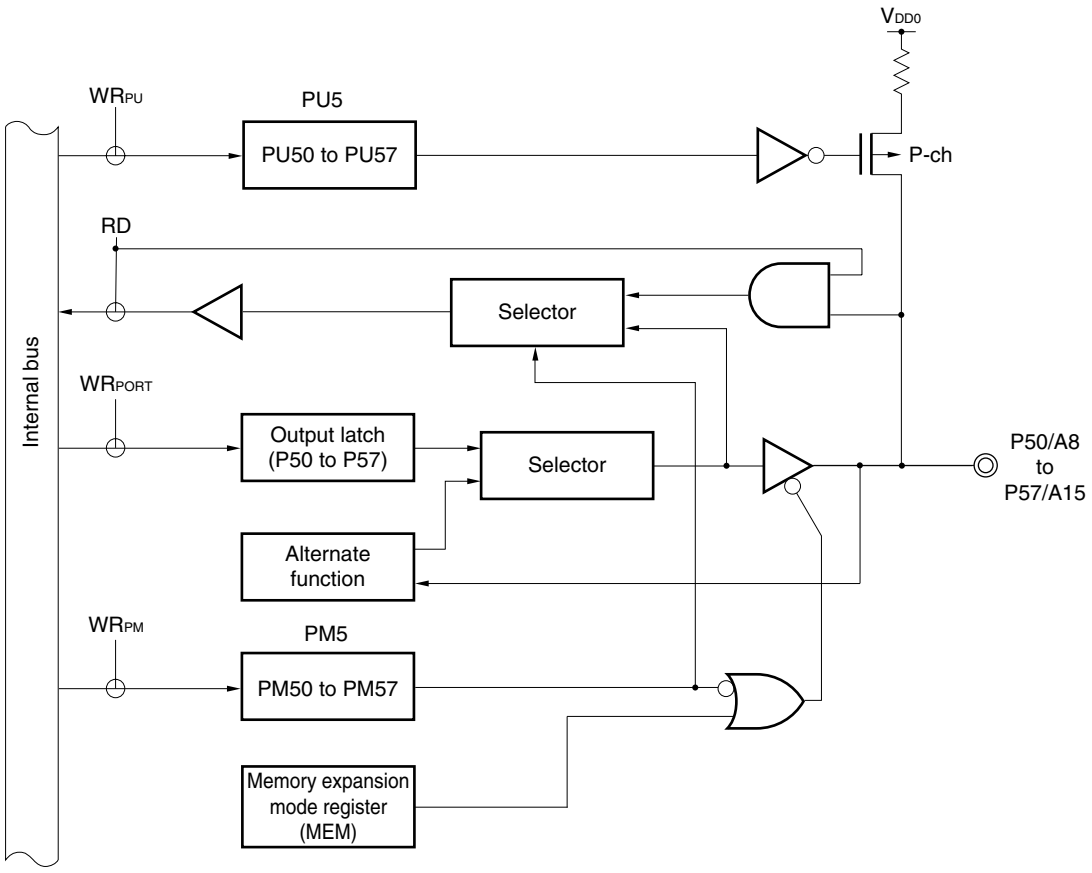
This port can also be used as an address bus in external memory expansion mode.

$\overline{\text{RESET}}$ input sets port 5 to input mode.

Figure 6-16 shows a block diagram of port 5.

Caution An on-chip pull-up resistor is not disconnected even if the external memory expansion mode is set when PU5n = 1 (n = 0 to 7).

Figure 6-16. Block Diagram of P50 to P57



PU5: Pull-up resistor option register 5
PM5: Port mode register 5
RD: Read signal
WR××: Write signal

9.4 Operation of 8-Bit Timer/Event Counters 50, 51

9.4.1 8-bit interval timer operation

The 8-bit timer/event counters operate as interval timers that generate interrupt requests repeatedly at intervals of the count value preset to 8-bit timer compare register 5n (CR5n).

When the count value of 8-bit timer counter 5n (TM5n) matches the value set to CR5n, counting continues with the TM5n value cleared to 0 and an interrupt request signal (INTTM5n) is generated.

The count clock of TM5n can be selected with bits 0 to 2 (TCL5n0 to TCL5n2) of timer clock select register 5n (TCL5n).

Setting

<1> Set each register.
- TCL5n: Select count clock.
- CR5n: Compare value
- TMC5n: Count operation stop, clear & start mode on match between TM5n and CR5n.
  (TMC5n = 0000×××0B × = don't care)

<2> After TCE5n = 1 is set, count operation starts.

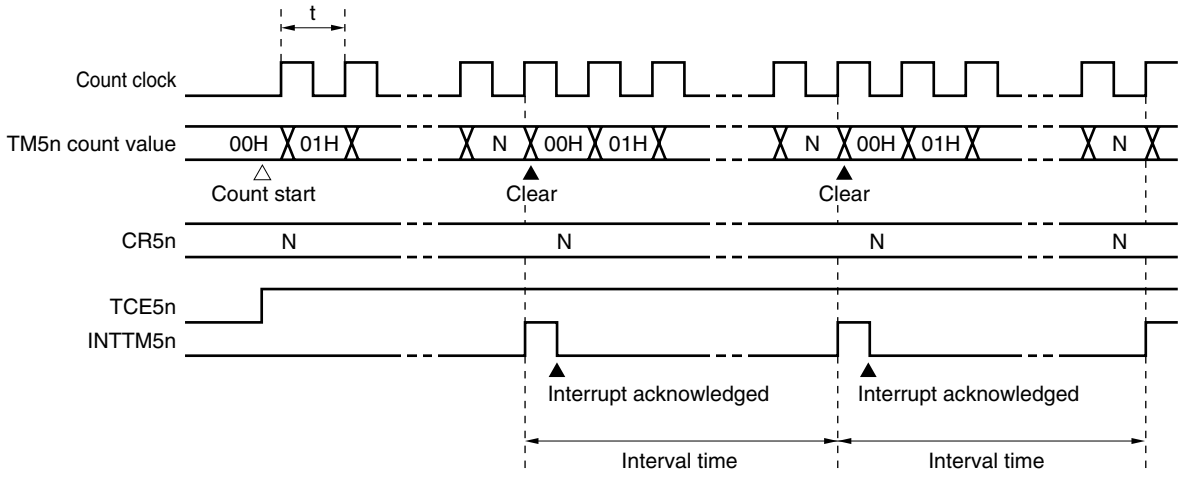
<3> If the values of TM5n and CR5n match, INTTM5n is generated (TM5n is cleared to 00H).

<4> INTTM5n is generated repeatedly at the same interval. Set TCE5n to 0 to stop the count operation.

Remark n = 0, 1

Figure 9-10. Interval Timer Operation Timing (1/3)

(a) Basic operation



Remarks 1. Interval time = (N + 1) × t
N = 00H to FFH

2. n = 0, 1

Figure 18-5. Format of IIC Control Register 0 (IICC0) (3/4)

STT0	Start condition trigger
0	Do not generate a start condition.
1	When bus is released (during STOP mode): Generate a start condition (for starting as master). The SDA0 line is changed from high level to low level and then the start condition is generated. Next, after the rated amount of time has elapsed, SCL0 is changed to low level. When bus is not used: This trigger functions as a start condition reservation flag. When set, it releases the bus and then automatically generates a start condition. Wait status (during master mode): Generate a restart condition after wait is released.

Cautions concerning set timing

- For master reception: Cannot be set during transfer. Can be set only in the waiting period when ACKE0 has been set to 0 and slave has been notified of final reception.
- For master transmission: A start condition may not be generated normally during the acknowledgment period. Therefore, set it during the waiting period.
- Cannot be set at the same time as SPT0.

Condition for clearing (STT0 = 0)	Condition for setting (STT0 = 1)
• Cleared by loss in arbitration • Cleared after start condition is generated by master device • Cleared by LREL0 = 1 (exit from communications) • When IICE0 = 0 (operation stop) • When $\overline{\text{RESET}}$ is input	• Set by instruction

Remark Bit 1 (STT0) is 0 when read after data has been set.

18.5.7 Interrupt request (INTIIC0) generation timing and wait control

The setting of bit 3 (WTIM0) of IIC control register 0 (IICC0) determines the timing by which INTIIC0 is generated and the corresponding wait control, as shown in Table 18-2.

Table 18-2. INTIIC0 Generation Timing and Wait Control

WTIM0	During Slave Device Operation			During Master Device Operation		
	Address	Data Reception	Data Transmission	Address	Data Reception	Data Transmission
0	9 Notes 1, 2	8 Note 2	8 Note 2	9	8	8
1	9 Notes 1, 2	9 Note 2	9 Note 2	9	9	9

Notes 1. The slave device's INTIIC0 signal and wait period occurs at the falling edge of the ninth clock only when there is a match with the address set to slave address register 0 (SVA0).
At this point, $\overline{ACK}$ is generated regardless of the value set to IICC0's bit 2 (ACKE0). For a slave device that has received an extension code, INTIIC0 occurs at the falling edge of the eighth clock.
However, if the address does not match after restart, INTIIC0 is generated at the falling edge of the 9th clock, but wait does not occur.

2. If the received address does not match the contents of slave address register 0 (SVA0) and extension code is not received, neither INTIIC0 nor a wait occurs.

Remark The numbers in the table indicate the number of the serial clock's clock signals. Interrupt requests and wait control are both synchronized with the falling edge of these clock signals.

(1) During address transmission/reception

- Slave device operation: Interrupt and wait timing are determined depending on the conditions described in Notes 1 and 2 above, regardless of the WTIM0 bit.
- Master device operation: Interrupt and wait timing occur at the falling edge of the ninth clock regardless of the WTIM0 bit.

(2) During data reception

- Master/slave device operation: Interrupt and wait timing are determined according to the WTIM0 bit.

(3) During data transmission

- Master/slave device operation: Interrupt and wait timing are determined according to the WTIM0 bit.

(4) Wait cancellation method

The four wait cancellation methods are as follows.

- By setting bit 5 (WREL0) of IIC control register 0 (IICC0) to 1
- By writing to IIC shift register 0 (IIC0)
- By setting a start condition (setting bit 1 (STT0) of IICC0 to 1) Note
- By setting a stop condition (setting bit 0 (SPT0) of IICC0 to 1) Note

Note Master only.

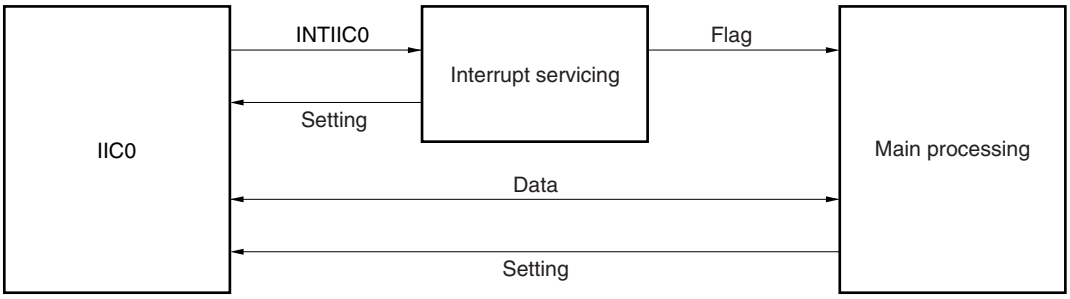
When an 8-clock wait has been selected (WTIM0 = 0), the presence/absence of $\overline{ACK}$ generation must be determined prior to wait cancellation.

(2) Slave operation

The processing procedure of the slave operation is as follows.

Basically, the slave operation is event-driven. Therefore, processing by the INTIIC0 interrupt (processing that must substantially change the operation status such as detection of a stop condition during communication) is necessary.

In the following explanation, it is assumed that the extension code is not supported for data communication. It is also assumed that the INTIIC0 interrupt servicing only performs status transition processing, and that actual data communication is performed by the main processing.



Therefore, data communication processing is performed by preparing the following three flags and passing them to the main processing instead of INTIIC0.

<1> Communication mode flag

This flag indicates the following two communication statuses.

- Clear mode: Status in which data communication is not performed
- Communication mode: Status in which data communication is performed (from valid address detection to stop condition detection, no detection of $\overline{\text{ACK}}$ from master, address mismatch)

<2> Ready flag

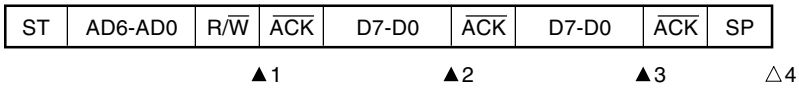
This flag indicates that data communication is enabled. Its function is the same as the INTIIC0 interrupt for ordinary data communication. This flag is set by interrupt servicing and cleared by the main processing. Clear this flag by interrupt servicing when communication is started. However, the ready flag is not set by interrupt servicing when the first data is transmitted. Therefore, the first data is transmitted without the flag being cleared (an address match is interpreted as a request for the next data).

<3> Communication direction flag

This flag indicates the direction of communication. Its value is the same as TRC0.

(b) When arbitration loss occurs during transmission of extension code

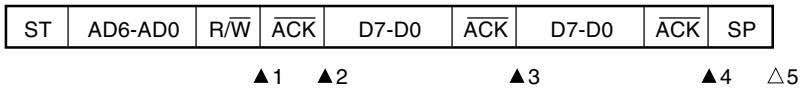
(i) When WTIM0 = 0



▲1 : IICS0 = 0110×010B (**Example** When ALD0 is read during interrupt servicing)
▲2 : IICS0 = 0010×000B
▲3 : IICS0 = 0010×000B
△4 : IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

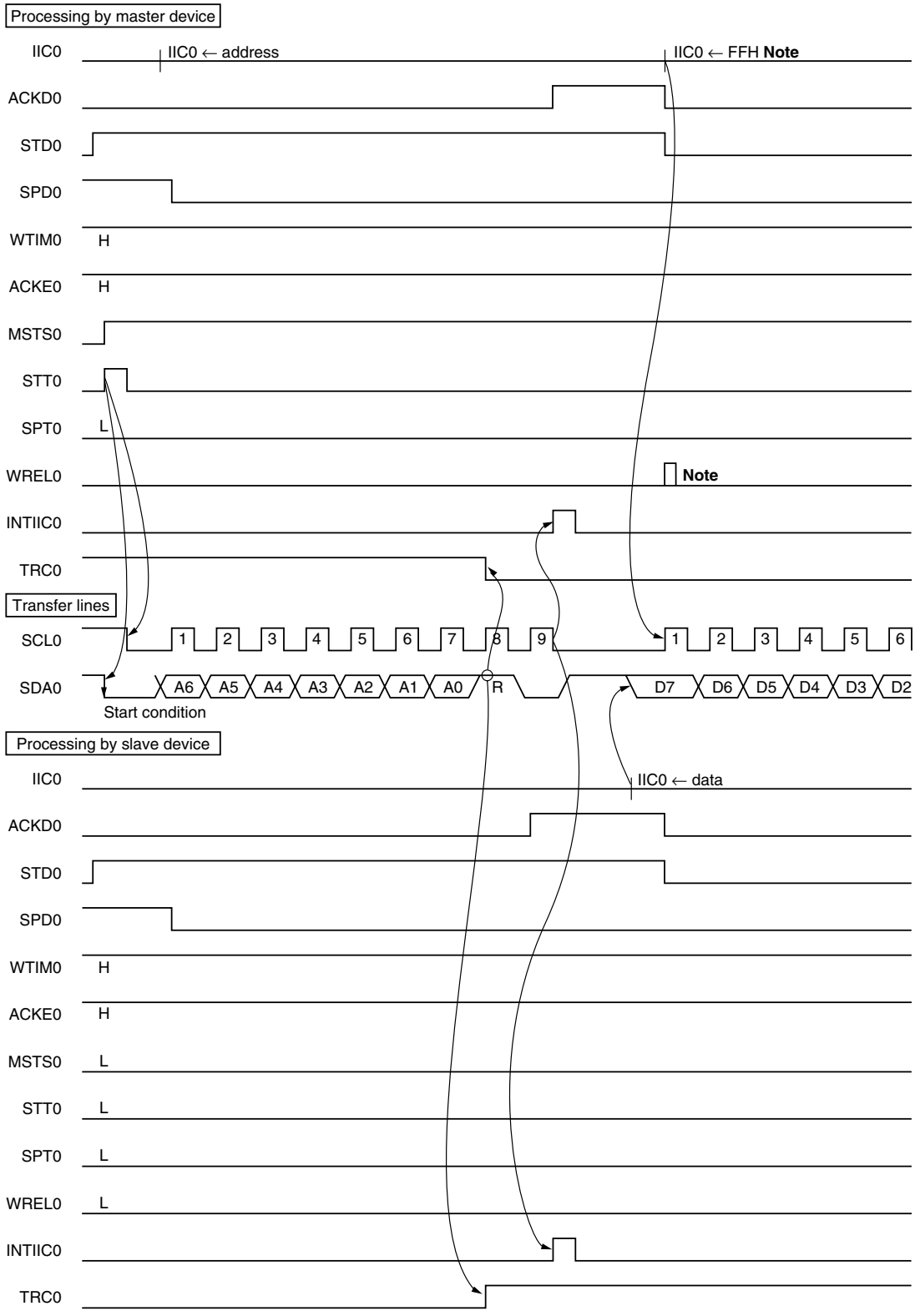


▲1 : IICS0 = 0110×010B (**Example** When ALD0 is read during interrupt servicing)
▲2 : IICS0 = 0010×110B
▲3 : IICS0 = 0010×100B
▲4 : IICS0 = 0010××00B
△5 : IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

Figure 18-24. Example of Slave to Master Communication (When 9-Clock Wait Is Selected for Both Master and Slave) (1/3)

(1) Start condition ~ address



Note To cancel master wait, write “FFH” to IIC0 or set WREL0.

19.4 Interrupt Servicing Operations

19.4.1 Non-maskable interrupt request acknowledgment operation

A non-maskable interrupt request is unconditionally acknowledged even in an interrupt acknowledgment disabled state. It does not undergo interrupt priority control and has the highest priority of all interrupts.

If a non-maskable interrupt request is acknowledged, the contents are saved into the stacks in the order of PSW, then PC, the IE flag and ISP flag are reset (0), and the contents of the vector table are loaded into the PC and branched. This disables the acknowledgment of multiple interrupts.

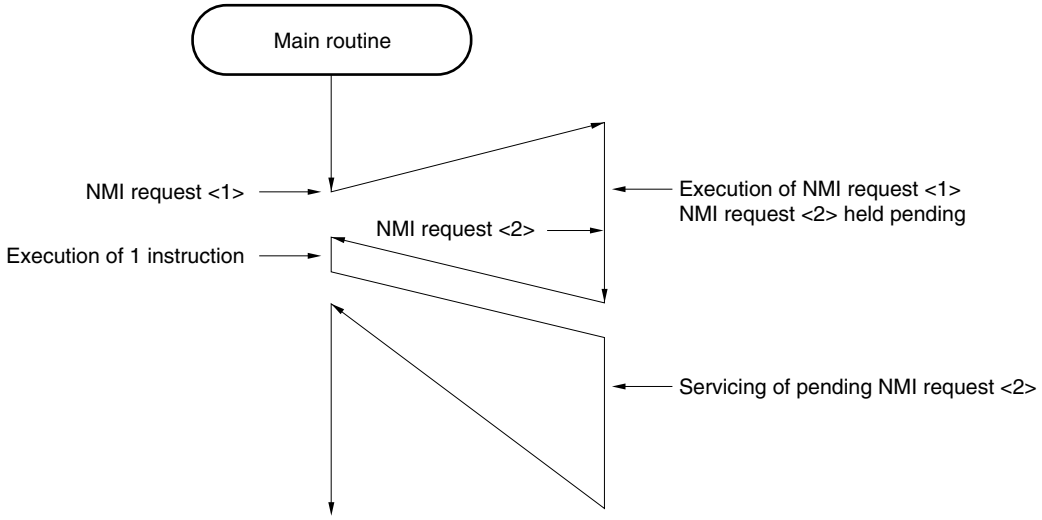
A new non-maskable interrupt request generated during execution of a non-maskable interrupt servicing program is acknowledged after the current non-maskable interrupt servicing program is terminated (following RETI instruction execution) and one main routine instruction has been executed. However, if a new non-maskable interrupt request is generated twice or more during non-maskable interrupt servicing program execution, only one non-maskable interrupt request is acknowledged after termination of the non-maskable interrupt servicing program.

Figures 19-7, 19-8, and 19-9 show the flowchart of non-maskable interrupt request generation through acknowledgment, the acknowledgment timing of a non-maskable interrupt request, and the acknowledgment operation when multiple non-maskable interrupt requests are generated, respectively.

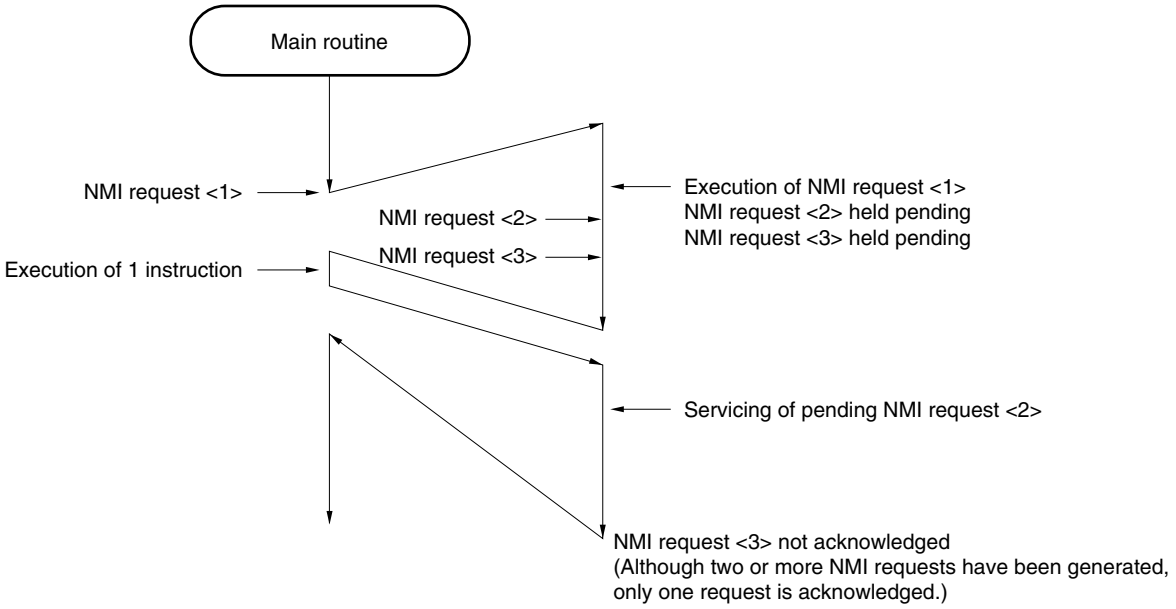
Caution Be sure to use the RETI instruction to restore processing from the non-maskable interrupt.

Figure 19-9. Non-Maskable Interrupt Request Acknowledgment Operation

(a) If a non-maskable interrupt request is generated during non-maskable interrupt servicing program execution



(b) If two non-maskable interrupt requests are generated during non-maskable interrupt servicing program execution



Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag
				Note 1	Note 2		Z AC CY
Bit manipu-late	**AND1**	CY, saddr.bit	3	6	7	CY ← CY ∧ (saddr.bit)	×
		CY, sfr.bit	3	–	7	CY ← CY ∧ sfr.bit	×
		CY, A.bit	2	4	–	CY ← CY ∧ A.bit	×
		CY, PSW.bit	3	–	7	CY ← CY ∧ PSW.bit	×
		CY, [HL].bit	2	6	7 + n	CY ← CY ∧ (HL).bit	×
	OR1	CY, saddr.bit	3	6	7	CY ← CY ∨ (saddr.bit)	×
		CY, sfr.bit	3	–	7	CY ← CY ∨ sfr.bit	×
		CY, A.bit	2	4	–	CY ← CY ∨ A.bit	×
		CY, PSW.bit	3	–	7	CY ← CY ∨ PSW.bit	×
		CY, [HL].bit	2	6	7 + n	CY ← CY ∨ (HL).bit	×
	XOR1	CY, saddr.bit	3	6	7	CY ← CY ⊻ (saddr.bit)	×
		CY, sfr.bit	3	–	7	CY ← CY ⊻ sfr.bit	×
		CY, A.bit	2	4	–	CY ← CY ⊻ A.bit	×
		CY, PSW. bit	3	–	7	CY ← CY ⊻ PSW.bit	×
		CY, [HL].bit	2	6	7 + n	CY ← CY ⊻ (HL).bit	×
	SET1	saddr.bit	2	4	6	(saddr.bit) ← 1	
		sfr.bit	3	–	8	sfr.bit ← 1	
		A.bit	2	4	–	A.bit ← 1	
		PSW.bit	2	–	6	PSW.bit ← 1	× × ×
		[HL].bit	2	6	8 + n + m	(HL).bit ← 1	
	CLR1	saddr.bit	2	4	6	(saddr.bit) ← 0	
		sfr.bit	3	–	8	sfr.bit ← 0	
		A.bit	2	4	–	A.bit ← 0	
		PSW.bit	2	–	6	PSW.bit ← 0	× × ×
		[HL].bit	2	6	8 + n + m	(HL).bit ← 0	
	SET1	CY	1	2	–	CY ← 1	1
	CLR1	CY	1	2	–	CY ← 0	0
	NOT1	CY	1	2	–	CY ← $\overline{CY}$	×

Notes 1. When the internal high-speed RAM area is accessed or an instruction with no data access is executed.

2. When an area except the internal high-speed RAM area is accessed

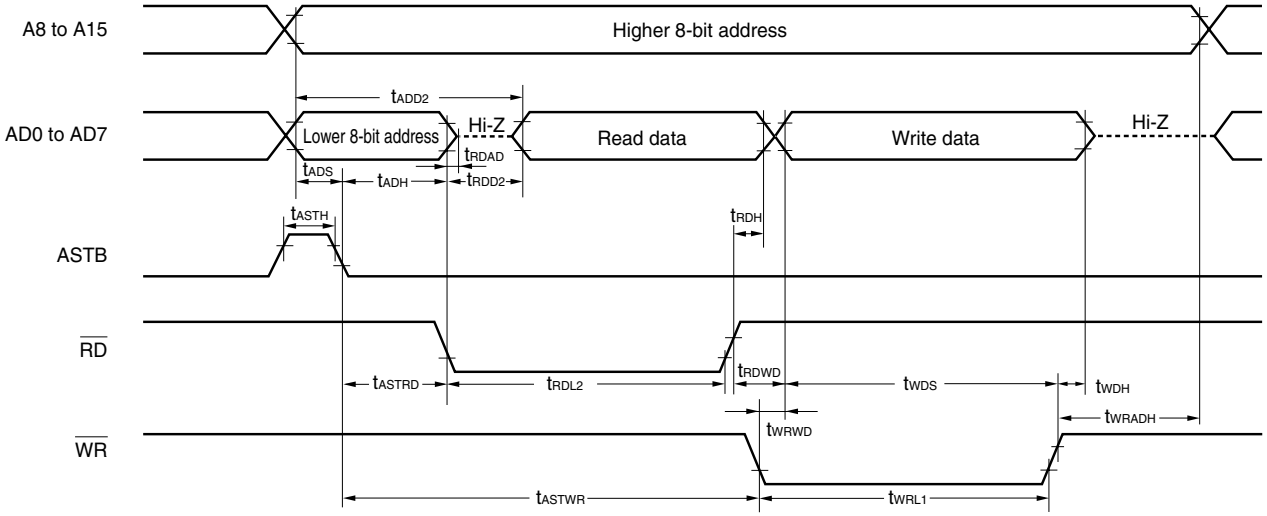
Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).

2. The number of clocks applies when there is a program in the internal ROM.

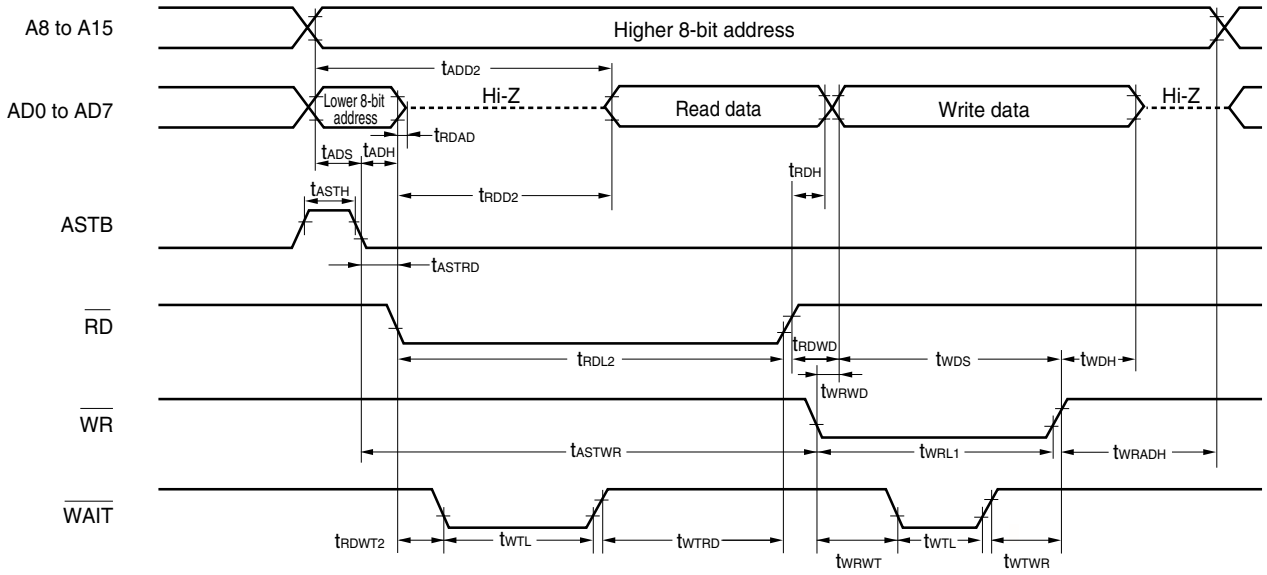
3. n is the number of waits when external memory expansion area is read from.

4. m is the number of waits when external memory expansion area is written to.

External data access (no wait):



External data access (wait insertion):



Flash Memory Programming Characteristics (T_A = +10 to +40°C, V_{DD} = 2.7 to 5.5 V, V_{SS} = AV_{SS} = 0 V)

(1) Write erase characteristics

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Operating frequency	f_X	4.0 V ≤ V_{DD} ≤ 5.5 V			1.0		8.38	MHz
		2.7 V ≤ V_{DD} < 4.0 V			1.0		5.00	MHz
V_{PP} supply voltage	V_{PP2}	During flash memory programming			9.7	10.0	10.3	V
V_{DD} supply current	I_{DD}	When	f_X = 8.38 MHz	V_{DD} = 5.0 V ±10%			24	mA
		V_{PP} = V_{PP2}	f_X = 5.00 MHz	V_{DD} = 3.0 V ±10%			12	mA
V_{PP} supply current	I_{PP}	When V_{PP} = V_{PP2}				75	100	mA
Step erase time[Note 1]	T_{er}				0.99	1.0	1.01	s
Overall erase time per area[Note 2]	T_{era}	When step erase time = 1 s					20	s/area
Step write time	T_{wr}				50		100	μs
Overall write time per word[Note 3]	T_{wrw}	When step write time = 100 μs					1000	μs
Number of rewrites per area[Note 4]	C_{erwr}	1 erase + 1 write after erase = 1 rewrite					20	Times/area

Notes 1. The recommended setting value of the step erase time is 1 s.

2. The prewrite time before erasure and the erase verify time (writeback time) are not included.

3. The actual write time per word is 100 μs longer. The internal verify time during or after a write is not included.

4. When a product is first written after shipment, "erase → write" and "write only" are both taken as one rewrite.

Example: P: Write, E: Erase

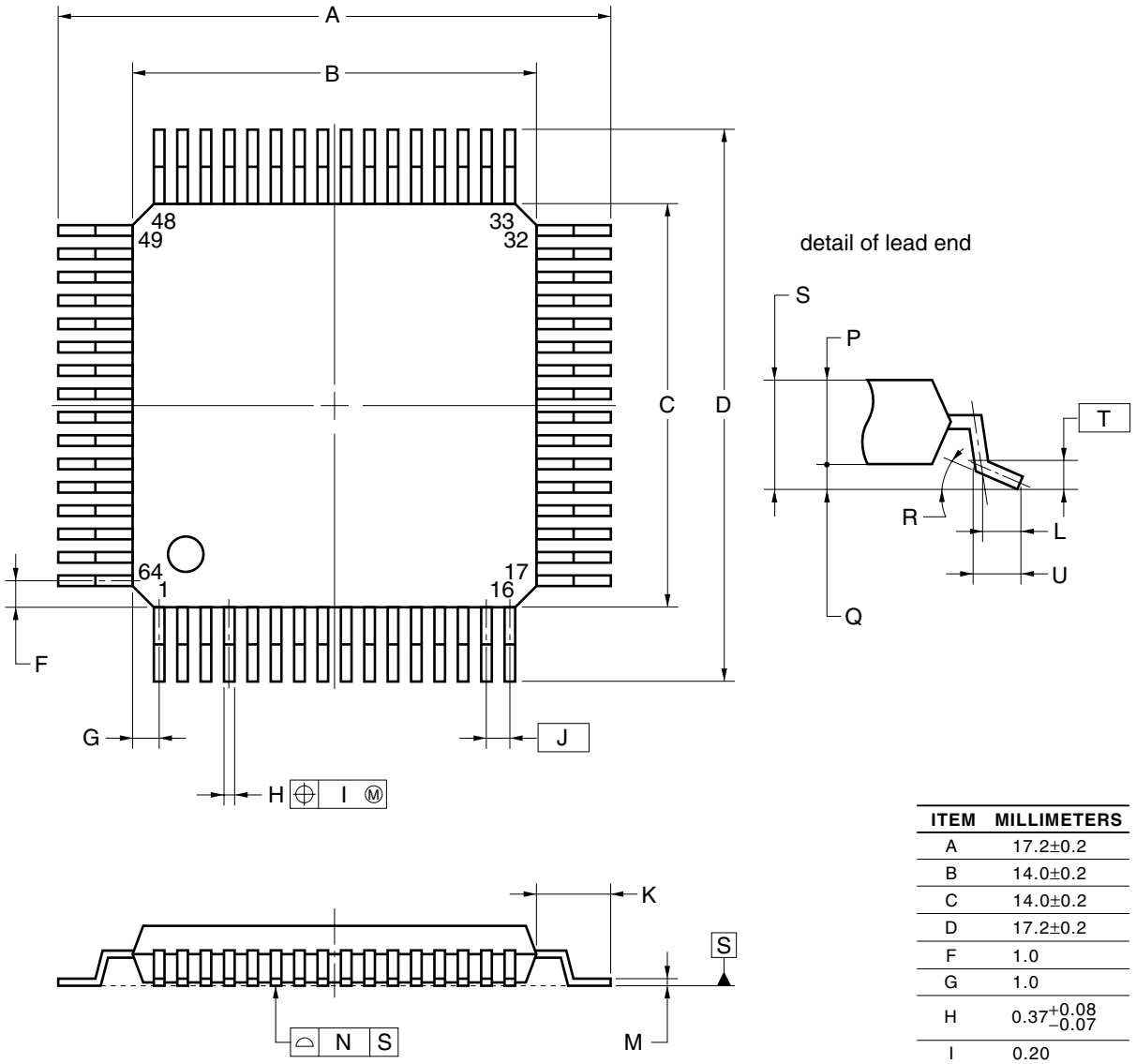
Shipped product →P→E→P→E→P: 3 rewrites

Shipped product →E→P→E→P→E→P: 3 rewrites

(2) Serial write operation characteristics

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Set time from V_{DD}↑ to V_{PP}↑	t_{DP}		10			μs
Release time from V_{PP}↑ to $\overline{RESET}$↑	t_{PR}		1.0			μs
V_{PP} pulse input start time from $\overline{RESET}$↑	t_{RP}		1.0			μs
V_{PP} pulse high-/low-level width	t_{PW}		8.0			μs
V_{PP} pulse input end time from $\overline{RESET}$↑	t_{RPE}				20	ms
V_{PP} pulse low-level input voltage	V_{PPL}		0.8V_{DD}	V_{DD}	1.2V_{DD}	V
V_{PP} pulse high-level input voltage	V_{PPH}		9.7	10.0	10.3	V

64-PIN PLASTIC LQFP (14x14)



ITEM	MILLIMETERS
A	17.2±0.2
B	14.0±0.2
C	14.0±0.2
D	17.2±0.2
F	1.0
G	1.0
H	$0.37^{+0.08}_{-0.07}$
I	0.20
J	0.8 (T.P.)
K	1.6±0.2
L	0.8
M	$0.17^{+0.03}_{-0.06}$
N	0.10
P	1.4±0.1
Q	0.127±0.075
R	$3^{\circ}{}^{+4^{\circ}}_{-3^{\circ}}$
S	1.7 MAX.
T	0.25
U	0.886±0.15

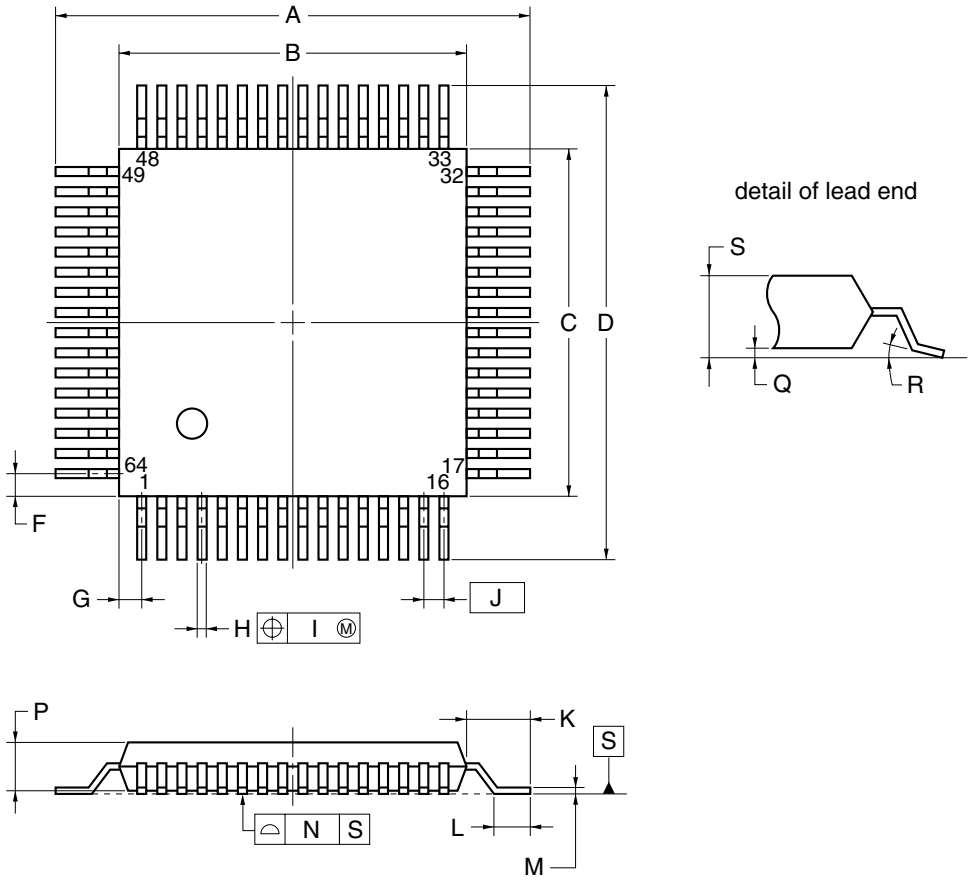
P64GC-80-8BS

NOTE

Each lead centerline is located within 0.20 mm of its true position (T.P.) at maximum material condition.

Remark The external dimensions and materials of the ES version are the same as those of the mass-produced version.

64-PIN PLASTIC QFP (14x14)



NOTE

Each lead centerline is located within 0.15 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	17.6±0.4
B	14.0±0.2
C	14.0±0.2
D	17.6±0.4
F	1.0
G	1.0
H	$0.37^{+0.08}_{-0.07}$
I	0.15
J	0.8 (T.P.)
K	1.8±0.2
L	0.8±0.2
M	$0.17^{+0.08}_{-0.07}$
N	0.10
P	2.55±0.1
Q	0.1±0.1
R	5°±5°
S	2.85 MAX.

P64GC-80-AB8-5

Remark The external dimensions and materials of the ES version are the same as those of the mass-produced version.

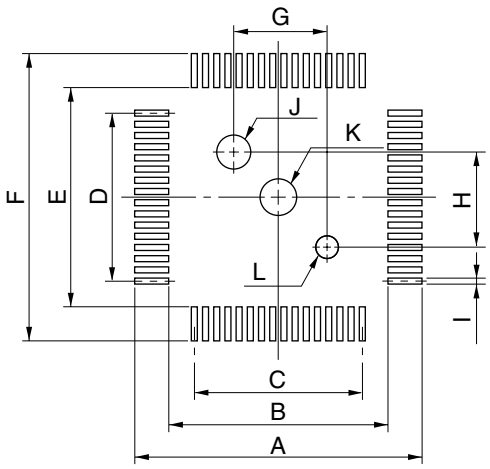
Table A-2. Major Differences Between μPD78018F, 780024A, 780034A, and 780078 Subseries (Software) (1/2)

Item \ Name	μPD78018F Subseries[Note 1]	μPD780024A, 780034A Subseries	μPD780078 Subseries	
A/D converter	—	Take the appropriate measures for the first A/D conversion result immediately after the A/D conversion operation is started (ADCS0 is set to 1), such as discarding it, because it may not satisfy the rating.		
			However, if a wait time of 14 μs (MIN.) has been secured after ADCE0 was set to 1 before starting operation (ADCS0 is set to 1), the first data can be used.	
16-bit timer/event counter	1 ch	1 ch	2 ch	
	TM0	TM0	TM00	TM01
Interval timer	○	○	○	
PWM output	○	—	—	
PPG output	—	○	○	
Pulse width measurement	○	○	○	
External event counter	○	○	○	
Square wave output	○	○	○	
Count clock	$f_X/2$, $f_X/2^2$, $f_X/2^3$, TI0	f_X, $f_X/2^2$, $f_X/2^6$, TI00	f_X, $f_X/2^2$, $f_X/2^6$, TI000	$f_X/2$, $f_X/2^3$ $f_X/2^9$, TI001
Control register	TMC0	TMC0	TMC00	TMC01
Output control register	TOC0	TOC0	TOC00	TOC01
Compare/capture register	CR00, CR01 (Capture only)	CR00, CR01	CR000, CR010	CR001, CR011
Prescaler mode register	TCL0[Note 2]	PRM0	PRM00	PRM01
Capture/compare control register	—	CRC0	CRC00	CRC01
Interrupt	INTTM0	INTTM00, INTTM01	INTTM000, INTTM010	INTTM001, INTTM011

Notes 1. Maintenance product

2. TCL0: Timer clock select register 0

Figure B-3. EV-9200GC-64 Recommended Board Mounting Pattern (for Reference Only)



EV-9200GC-64-P1E

ITEM	MILLIMETERS	INCHES
A	19.5	0.768
B	14.8	0.583
C	$0.8-0.02 \times 15=12.0-0.05$	$0.031^{+0.002}_{-0.001} \times 0.591=0.472^{+0.003}_{-0.002}$
D	$0.8-0.02 \times 15=12.0-0.05$	$0.031^{+0.002}_{-0.001} \times 0.591=0.472^{+0.003}_{-0.002}$
E	14.8	0.583
F	19.5	0.768
G	$6.00-0.08$	$0.236^{+0.004}_{-0.003}$
H	$6.00-0.08$	$0.236^{+0.004}_{-0.003}$
I	$0.5-0.02$	$0.197^{+0.001}_{-0.002}$
J	$\phi 2.36-0.03$	$\phi 0.093^{+0.001}_{-0.002}$
K	$\phi 2.2-0.1$	$\phi 0.087^{+0.004}_{-0.005}$
L	$\phi 1.57-0.03$	$\phi 0.062^{+0.001}_{-0.002}$

Caution Dimensions of mount pad for EV-9200 and that for target device (QFP) may be different in some parts. For the recommended mount pad dimensions for QFP, refer to "Semiconductor Device Mount Manual" (http://www.necel.com/pkg/en/mount/index.html).

(4/7)

Edition	Contents	Applied to:
2nd	Modification of **18.5.16 (3) (d) (ii) When WTIM0 = 1 (after restart, does not match with address (= not extension code))**	**CHAPTER 18 SERIAL INTERFACE IIC0 (μPD780078Y SUBSERIES ONLY)**
	Modification of **(1) Start condition ~ address** and **(2) Data** in **Figure 18-23 Example of Master to Slave Communication (When 9-Clock Wait Is Selected for Both Master and Slave)**	
	Modification of **Figure 18-24 Example of Slave to Master Communication (When 9-Clock Wait Is Selected for Both Master and Slave)**	
	Modification of **(E) Software interrupt** in **Figure 19-1 Basic Configuration of Interrupt Function**	**CHAPTER 19 INTERRUPT FUNCTIONS**
	Addition of **Cautions 3** and **4** in **Figure 19-2 Format of Interrupt Request Flag Register (IF0L, IF0H, IF1L)**	
	Addition of **Caution** in **Figure 19-5 Format of External Interrupt Rising Edge Enable Register (EGP), External Interrupt Falling Edge Enable Register (EGN)**	
	Addition of description and **Remark** in **19.4.1 Non-maskable interrupt request acknowledgment operation**	
	Addition of description in **19.4.2 Maskable interrupt acknowledgment operation**	
	Addition of item to **Table 19-4 Interrupt Requests Enabled for Multiple Interrupt Servicing**	
	Addition of description about when using expanded-specification products	**CHAPTER 20 EXTERNAL DEVICE EXPANSION FUNCTION**
	Addition of clock output and buzzer output in **Table 21-1 HALT Mode Operating Statuses**	**CHAPTER 21 STANDBY FUNCTION**
	Modification of clock output in **Table 21-3 STOP Mode Operating Statuses**	
	Modification of chapter	**CHAPTER 23 μPD78F0078, 78F0078Y**
	Addition of chapters	**CHAPTER 25 ELECTRICAL SPECIFICATIONS**
		CHAPTER 26 PACKAGE DRAWINGS
		CHAPTER 27 RECOMMENDED SOLDERING CONDITIONS
	Addition of **Table A-2 Major Differences Between μPD78018F, 780024A, 780034A, and 780078 Subseries (Software)**	**APPENDIX A DIFFERENCES BETWEEN μPD78018F, 780024A, 780034A, AND 780078 SUBSERIES**
	Modification of chapter	**APPENDIX B DEVELOPMENT TOOLS**
	Addition of chapters	**APPENDIX C NOTES ON TARGET SYSTEM DESIGN**
		APPENDIX E REVISION HISTORY

(7/7)

Edition	Contents	Applied to:
3rd	Modification of description in **(1) Registers to be used** in **16.4.2 3-wire serial I/O mode**	**CHAPTER 16 SERIAL INTERFACE SIO3**
	Modification of **Table 16-3 Register Settings**	
	Partial modification of **Figure 17-1 Block Diagram of Serial Interface CSI1**	**CHAPTER 17 SERIAL INTERFACE CSI1**
	Partial modification of **Figure 17-3 Format of Serial Clock Select Register 1 (CSIC1)**	
	Modification of description in **(1) Registers to be used** in **17.4.2 3-wire serial I/O mode**	
	Addition of **(5) SO1 output** to **17.4.2 3-wire serial I/O mode**	
	Modification of **Figure 18-18 Communication Reservation Timing**	**CHAPTER 18 SERIAL INTERFACE IIC0 (μPD780078Y SUBSERIES ONLY)**
	Modification of **Figure 18-21 Master Operation Flowchart (5/5)**	
	Modification of **Figure 18-21 Master Operation Flowchart**	
	Modification of **(2) Slave operation** in **18.5.15 Communication operations**	
	Addition of **Table 19-3 Ports Corresponding to EGPn and EGNn**	**CHAPTER 19 INTERRUPT FUNCTIONS**
	Modification of part of description in **19.4.1 Non-maskable interrupt request acknowledgment operation**	
	Modification of part of description in **19.4.2 Maskable interrupt request acknowledgment operation**	
	Addition of **Note** to **Figure 20-2 Format of Memory Expansion Mode Register (MEM)** and addition of **Figure 20-3 Pins Specified for Address (with μPD780076 and 780076Y)**	**CHAPTER 20 EXTERNAL DEVICE EXPANSION FUNCTION**
	Partial modification of **Table 23-3 Communication Mode List**	**CHAPTER 23 μPD78F0078, 78F0078Y**
	Division of Note in previous edition of **Figure 23-5 Example of Connection with Dedicated Flash Programmer** to **Notes 1** and **2** and modification of contents	
	Addition of description on voltage monitoring of dedicated flash programmer to **<Power supply>** in **23.3.3 On-board pin processing**	
	Revision of **CHAPTER 25 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS OF μPD780076, 780078, 78F0078)**	**CHAPTER 25 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS OF μPD780076, 780078, 78F0078)**
	Addition of **CHAPTER 26 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS OF μPD780076Y, 780078Y, 78F0078Y)**	**CHAPTER 26 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS OF μPD780076Y, 780078Y, 78F0078Y)**
	Revision of **CHAPTER 27 ELECTRICAL SPECIFICATIONS (CONVENTIONAL PRODUCTS)**	**CHAPTER 27 ELECTRICAL SPECIFICATIONS (CONVENTIONAL PRODUCTS)**
	Partial modification of **Table 29-1 Surface Mounting Type Soldering Conditions**	**CHAPTER 29 RECOMMENDED SOLDERING CONDITIONS**
	Deletion of **B.7 Embedded Software** and **B.8 System Upgrade from Former In-circuit Emulator for 78K/0 Series to IE-78001-R-A** in the previous edition	**APPENDIX B DEVELOPMENT TOOLS**