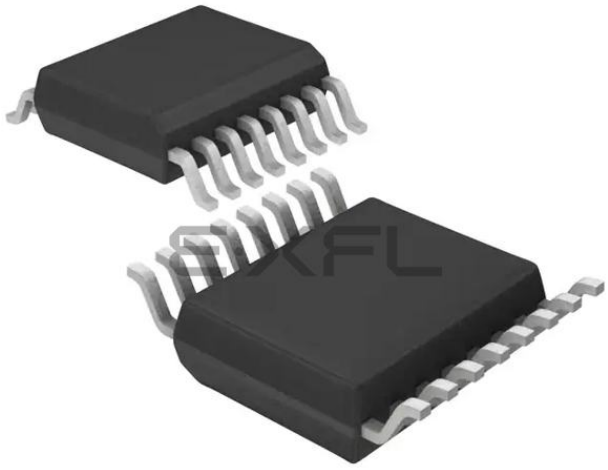
Renesas Electronics America Inc - UPD78F9212MA-FAA-AX Datasheet



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	78K0S
Core Size	8-Bit
Speed	10MHz
Connectivity	-
Peripherals	LVD, POR, PWM, WDT
Number of I/O	14
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	16-SSOP (0.173", 4.40mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f9212ma-faa-ax

RENESAS

User's Manual

78K0S/KY1+

8-bit Single-Chip Microcontrollers

μPD78F9210 **μPD78F9211** **μPD78F9212**
μPD78F9210(A) **μPD78F9211(A)** **μPD78F9212(A)**
μPD78F9210(A2) **μPD78F9211(A2)** **μPD78F9212(A2)**
μPD78F9510 **μPD78F9511** **μPD78F9512**

Document No. U16994EJ6V0UD00 (6th edition)
Date Published November 2009 NS

Target Readers

This manual is intended for user engineers who wish to understand the functions of the 78K0S/KY1+ in order to design and develop its application systems and programs.

The target devices are the following subseries products.

<R> • 78K0S/KY1+: μPD78F9210, 78F9211, 78F9212, 78F9210(A), 78F9211(A), 78F9212(A), 78F9210(A2), 78F9211(A2), 78F9212(A2), 78F9511, 78F9512

Purpose

This manual is intended to give users on understanding of the functions described in the **Organization** below.

Organization

Two manuals are available for 78K0S/KY1+: this manual and the Instruction Manual (common to the 78K/0S Series).

78K0S/KY1+ User's Manual	78K/0S Series Instructions User's Manual
• Pin functions • Internal block functions • Interrupts • Other internal peripheral functions • Electrical specifications	• CPU function • Instruction set • Instruction description

How to Use This Manual

It is assumed that the readers of this manual have general knowledge of electrical engineering, logic circuits, and microcontrollers.

◊ To understand the overall functions of 78K0S/KY1+

→ Read this manual in the order of the **CONTENTS**. The mark <R> shows major revised points. The revised points can be easily searched by copying an "<R>" in the PDF file and specifying it in the "Find what:" field.

◊ How to read register formats

→ For a bit number enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0S, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0S.

◊ To learn the detailed functions of a register whose register name is known

→ See **APPENDIX B REGISTER INDEX**.

◊ To learn the details of the instruction functions of the 78K/0S Series

→ Refer to **78K/0S Series Instructions User's Manual (U11047E)** separately available.

◊ To learn the electrical specifications of the 78K0S/KY1+

→ See **CHAPTER 19** and **CHAPTER 20 ELECTRICAL SPECIFICATIONS**.

Figure 3-10. Data to Be Saved to Stack Memory

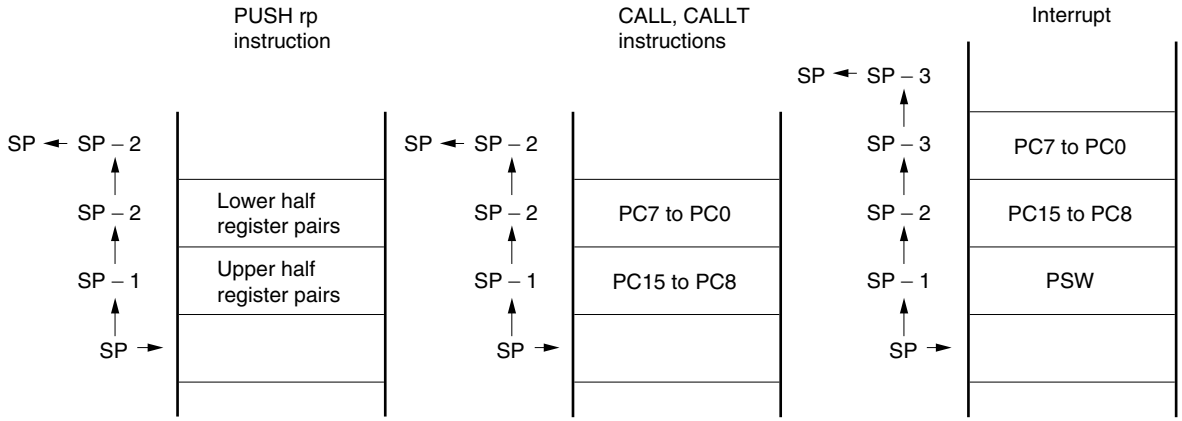


Figure 3-11. Data to Be Restored from Stack Memory

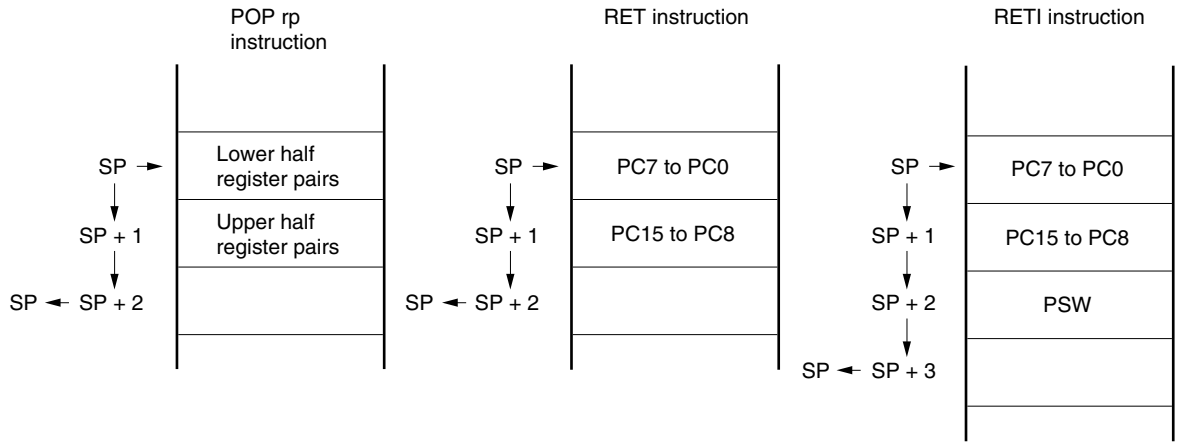
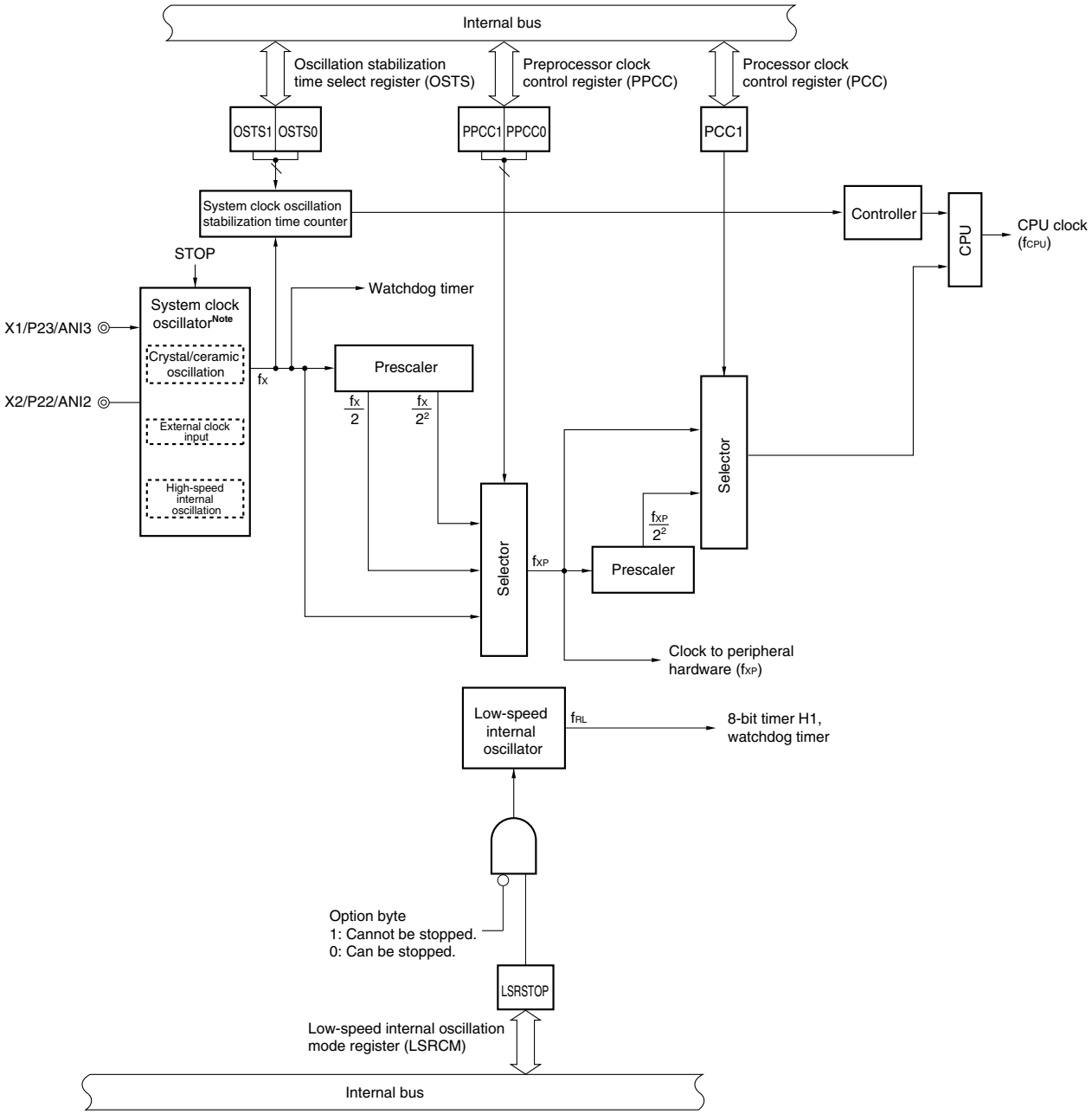
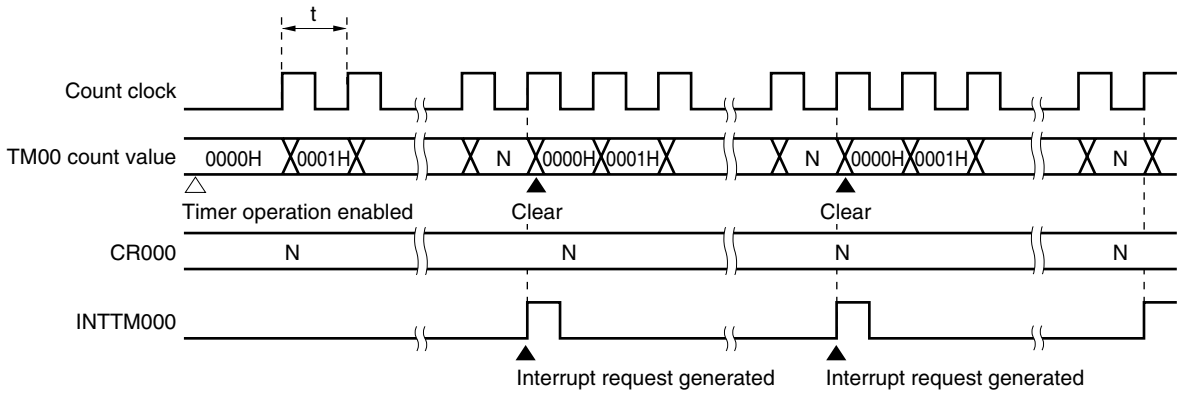


Figure 5-1. Block Diagram of Clock Generators



Note Select the high-speed internal oscillator, crystal/ceramic oscillator, or external clock input circuit as the system clock source by using the option byte.

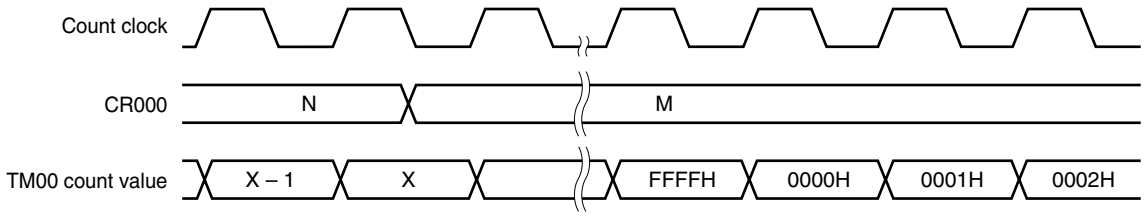
Figure 6-13. Timing of Interval Timer Operation



Remark Interval time = (N + 1) × t
N = 0001H to FFFFH (settable range)

When the compare register is changed during timer count operation, if the value after 16-bit timer capture/compare register 000 (CR000) is changed is smaller than that of 16-bit timer counter 00 (TM00), TM00 continues counting, overflows and then restarts counting from 0. Thus, if the value (M) after the CR000 change is smaller than that (N) before the change, it is necessary to restart the timer after changing CR000.

Figure 6-14. Timing After Change of Compare Register During Timer Count Operation (N → M: N > M)



Remark N > X > M

6.4.2 External event counter operation

Setting

The basic operation setting procedure is as follows.

<1> Set the CRC00 register (see **Figure 6-15** for the set value).
<2> Set the count clock by using the PRM00 register.
<3> Set any value to the CR000 register (0000H cannot be set).
<4> Set the TMC00 register to start the operation (see **Figure 6-15** for the set value).

Remarks 1. For the setting of the TI000 pin, see **6.3 (5) Port mode register 2 (PM2) and port mode control register 2 (PMC2)**.
2. For how to enable the INTTM000 interrupt, see **CHAPTER 10 INTERRUPT FUNCTIONS**.

CHAPTER 8 WATCHDOG TIMER

8.1 Functions of Watchdog Timer

The watchdog timer is used to detect an inadvertent program loop. If a program loop is detected, an internal reset signal is generated.

When a reset occurs due to the watchdog timer, bit 4 (WDTRF) of the reset control flag register (RESF) is set to 1. For details of RESF, see **CHAPTER 12 RESET FUNCTION**.

Table 8-1. Loop Detection Time of Watchdog Timer

Loop Detection Time	
During Low-Speed Internal oscillation Clock Operation	During System Clock Operation
$2^{11}/f_{RL}$ (4.27 ms)	$2^{13}/f_X$ (819.2 μs)
$2^{12}/f_{RL}$ (8.53 ms)	$2^{14}/f_X$ (1.64 ms)
$2^{13}/f_{RL}$ (17.07 ms)	$2^{15}/f_X$ (3.28 ms)
$2^{14}/f_{RL}$ (34.13 ms)	$2^{16}/f_X$ (6.55 ms)
$2^{15}/f_{RL}$ (68.27 ms)	$2^{17}/f_X$ (13.11 ms)
$2^{16}/f_{RL}$ (136.53 ms)	$2^{18}/f_X$ (26.21 ms)
$2^{17}/f_{RL}$ (273.07 ms)	$2^{19}/f_X$ (52.43 ms)
$2^{18}/f_{RL}$ (546.13 ms)	$2^{20}/f_X$ (104.86 ms)

Remarks 1. f_{RL}: Low-speed internal oscillation clock oscillation frequency

2. f_X: System clock oscillation frequency

3. Figures in parentheses apply to operation at f_{RL} = 480 kHz (MAX.), f_X = 10 MHz.

The operation mode of the watchdog timer (WDT) is switched according to the option byte setting of the on-chip low-speed internal oscillator as shown in Table 8-2.

Table 8-2. Option Byte Setting and Watchdog Timer Operation Mode

	Option Byte Setting	
	Low-Speed Internal Oscillator Cannot Be Stopped	Low-Speed Internal Oscillator Can Be Stopped by Software
Watchdog timer clock source	Fixed to f_{RL}[Note 1].	• Selectable by software (f_X, f_{RL} or stopped) • When reset is released: f_{RL}
Operation after reset	Operation starts with the maximum interval ($2^{18}/f_{RL}$).	Operation starts with the maximum interval ($2^{18}/f_{RL}$).
Operation mode selection	The interval can be changed only once.	The clock selection/interval can be changed only once.
Features	The watchdog timer cannot be stopped.	The watchdog timer can be stopped[Note 2].

Notes 1. As long as power is being supplied, low-speed internal oscillator cannot be stopped (except in the reset period).

2. The conditions under which clock supply to the watchdog timer is stopped differ depending on the clock source of the watchdog timer.

<1> If the clock source is f_X, clock supply to the watchdog timer is stopped under the following conditions.

- When f_X is stopped
- In HALT/STOP mode
- During oscillation stabilization time

<2> If the clock source is f_{RL}, clock supply to the watchdog timer is stopped under the following conditions.

- If the CPU clock is f_X and if f_{RL} is stopped by software before execution of the STOP instruction
- In HALT/STOP mode

Remarks 1. f_{RL}: Low-speed internal oscillation clock oscillation frequency

2. f_X: System clock oscillation frequency

8.4.3 Watchdog timer operation in STOP mode (when "low-speed internal oscillator can be stopped by software" is selected by option byte)

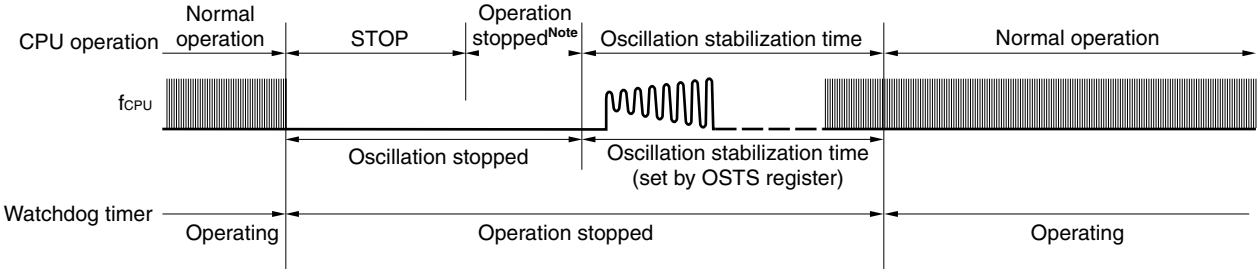
The watchdog timer stops counting during STOP instruction execution regardless of whether the system clock or low-speed internal oscillation clock is being used.

(1) When the watchdog timer operation clock is the system clock (f_X) when the STOP instruction is executed

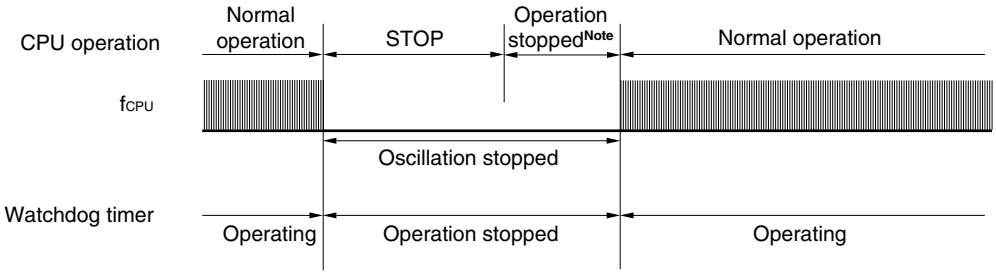
When STOP instruction is executed, operation of the watchdog timer is stopped. After STOP mode is released, operation stops for 34 μs (TYP.) (after waiting for the oscillation stabilization time set by the oscillation stabilization time select register (OSTS) after operation stops in the case of crystal/ceramic oscillation) and then counting is started again using the operation clock before the operation was stopped. At this time, the counter is not cleared to 0 but holds its value.

Figure 8-6. Operation in STOP Mode (WDT Operation Clock: Clock to Peripheral Hardware)

<1> CPU clock: Crystal/ceramic oscillation clock



<2> CPU clock: High-speed internal oscillation clock or external clock input



Note The operation stop time is 17 μs (MIN.), 34 μs (TYP.), and 67 μs (MAX.).

Table 9-1. Sampling Time and A/D Conversion Time

Reference Voltage Range[Note 1]	Sampling Time[Note 2]	Conversion Time[Note 3]	f_{XP} = 8 MHz		f_{XP} = 10 MHz		FR2	FR1	FR0
			Sampling Time[Note 2]	Conversion Time[Note 3]	Sampling Time[Note 2]	Conversion Time[Note 3]			
$V_{DD} \geq 4.5$ V	$12/f_{XP}$	$36/f_{XP}$	1.5 μs	4.5 μs	1.2 μs	3.6 μs	0	0	0
$V_{DD} \geq 4.0$ V	$24/f_{XP}$	$72/f_{XP}$	3.0 μs	9.0 μs	2.4 μs	7.2 μs	1	0	0
$V_{DD} \geq 2.85$ V	$96/f_{XP}$	$144/f_{XP}$	12.0 μs	18.0 μs	9.6 μs	14.4 μs	1	1	0
	$48/f_{XP}$	$96/f_{XP}$	6.0 μs	12.0 μs	4.8 μs	9.6 μs	1	0	1
	$48/f_{XP}$	$72/f_{XP}$	6.0 μs	9.0 μs	4.8 μs	7.2 μs	0	1	0
	$24/f_{XP}$	$48/f_{XP}$	3.0 μs	6.0 μs	Setting prohibited Note 4 (2.4 μs)	Setting prohibited Note 4 (4.8 μs)	0	0	1
$V_{DD} \geq 2.7$ V	$176/f_{XP}$	$224/f_{XP}$	22.0 μs	28.0 μs	17.6 μs	22.4 μs	1	1	1
	$88/f_{XP}$	$112/f_{XP}$	11.0 μs	14.0 μs	Setting prohibited Note 4 (8.8 μs)	Setting prohibited Note 4 (11.2 μs)	0	1	1

Notes 1. Be sure to set the FR2, FR1, and FR0, in accordance with the reference voltage so that **Notes 2** and **3** below are satisfied.

Example When $V_{DD} \geq 2.7$ V, f_{XP} = 8 MHz

- The sampling time is 11.0 μs or more and the A/D conversion time is 14.0 μs or more and 100 μs or less.
- Set FR2, FR1, and FR0 = 0, 1, 1 or 1, 1, 1.

2. Set the sampling time as follows.

- $V_{DD} \geq 4.5$ V: 1.0 μs or more
- $V_{DD} \geq 4.0$ V: 2.4 μs or more
- $V_{DD} \geq 2.85$ V: 3.0 μs or more
- $V_{DD} \geq 2.7$ V: 11.0 μs or more

3. Set the A/D conversion time as follows.

- $V_{DD} \geq 4.5$ V: 3.0 μs or more and less than 100 μs
- $V_{DD} \geq 4.0$ V: 4.8 μs or more and less than 100 μs
- $V_{DD} \geq 2.85$ V: 6.0 μs or more and less than 100 μs
- $V_{DD} \geq 2.7$ V: 14.0 μs or more and less than 100 μs

4. Setting is prohibited because the values do not satisfy the condition of **Notes 2** or **3**.

Caution The above sampling time and conversion time do not include the clock frequency error. Select the sampling time and conversion time such that Notes 2 and 3 above are satisfied, while taking the clock frequency error into consideration (an error margin maximum of ±5% when using the high-speed internal oscillator).

Remarks 1. f_{XP}: Oscillation frequency of clock to peripheral hardware

2. The conversion time refers to the total of the sampling time and the time from successively comparing with the sampling value until the conversion result is output.

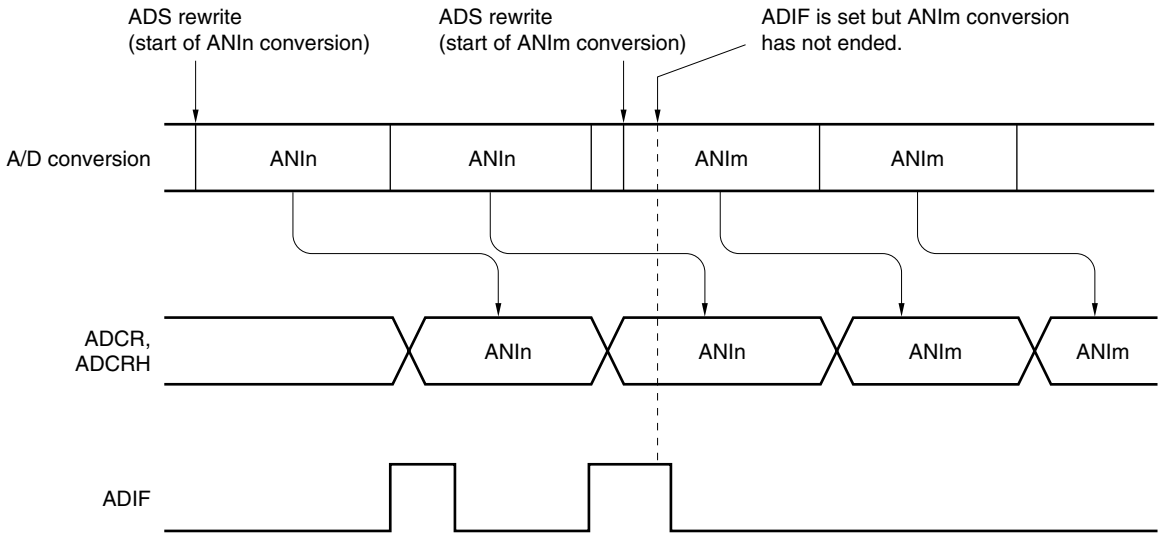
(7) Interrupt request flag (ADIF)

The interrupt request flag (ADIF) is not cleared even if the analog input channel specification register (ADS) is changed.

Therefore, if an analog input pin is changed during A/D conversion, the A/D conversion result and ADIF for the pre-change analog input may be set just before the ADS rewrite. Caution is therefore required since, at this time, when ADIF is read immediately after the ADS rewrite, ADIF is set despite the fact A/D conversion for the post-change analog input has not ended.

When A/D conversion is stopped and then resumed, clear ADIF before the A/D conversion operation is resumed.

Figure 9-20. Timing of A/D Conversion End Interrupt Request Generation



Remarks 1. n = 0 to 3

2. m = 0 to 3

(8) Conversion results just after A/D conversion start

The first A/D conversion value immediately after A/D conversion starts may not fall within the rating range if the ADCS bit is set to 1 within 1 μs after the ADCE bit was set to 1, or if the ADCS bit is set to 1 with the ADCE bit = 0. Take measures such as polling the A/D conversion end interrupt request (INTAD) and removing the first conversion result.

(9) A/D conversion result register (ADCR, ADCRH) read operation

When a write operation is performed to the A/D converter mode register (ADM) and analog input channel specification register (ADS), the contents of ADCR and ADCRH may become undefined. Read the conversion result following conversion completion before writing to ADM and ADS. Using a timing other than the above may cause an incorrect conversion result to be read.

(10) Operating current at conversion waiting mode

The DC characteristic of the operating current during the STOP mode is not satisfied due to the conversion waiting mode (only the comparator consumes power), when bit 7 (ADCS) and bit 0 (ADCE) of the A/D converter mode register (ADM) are set to 0 and 1 respectively.

CHAPTER 10 INTERRUPT FUNCTIONS

10.1 Interrupt Function Types

There are two types of interrupts: maskable interrupts and resets.

- **Maskable interrupts**
  These interrupts undergo mask control. When an interrupt request occurs, the standby release signal occurs, and if an interrupt can be acknowledged then the program corresponding to the address written in the vector table address is executed (vector interrupt servicing). When several interrupt requests are generated at the same time, processing takes place in the priority order of the vector interrupt servicing. For details on the priority order, see Table 10-1.
  There are external sources and internal sources of maskable interrupts.
  - μPD78F921x: external sources: 2, internal sources: 5
  - μPD78F951x: external sources: 2, internal sources: 4

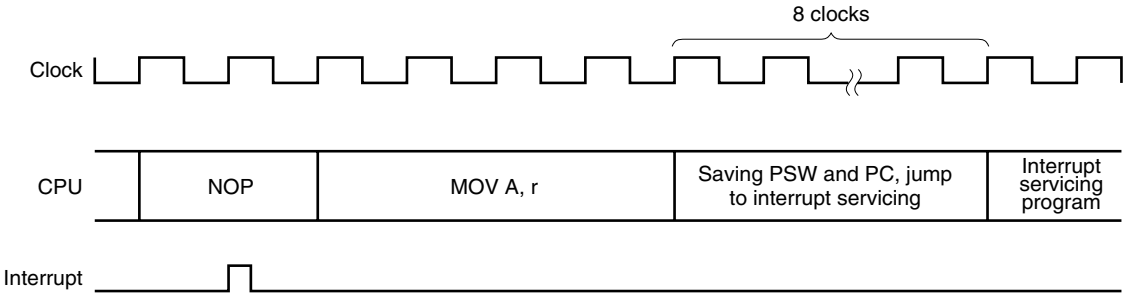
<R>

- **Reset**
  The CPU and SFR are returned to their initial states by the reset signal. The causes for reset signal occurrences are shown in Table 10-1.
  When a reset signal occurs, program execution starts from the programs at the addresses written in addresses 0000H and 0001H.

10.2 Interrupt Sources and Configuration

<R> There are a total of seven maskable interrupt sources in μPD78F921x, and six maskable interrupt sources in μPD78F951x, and up to four reset sources (see **Table 10-1**).

Figure 10-8. Interrupt Request Acknowledgment Timing (When Interrupt Request Flag Is Set at Last Clock During Instruction Execution)



If an interrupt request flag (××IF) is set at the last clock of the instruction, the interrupt acknowledgment processing starts after the next instruction is executed.

Figure 10-8 shows an example of the interrupt request acknowledgment timing for an interrupt request flag that is set at the second clock of NOP (2-clock instruction). In this case, the MOV A, r instruction after the NOP instruction is executed, and then the interrupt acknowledgment processing is performed.

Caution Interrupt requests will be held pending while the interrupt request flag register 0 (IF0) or interrupt mask flag register 0 (MK0) are being accessed.

10.4.2 Multiple interrupt servicing

In order to perform multiple interrupt servicing in which another interrupt is acknowledged while an interrupt is being serviced, the interrupt mask function must be used to mask interrupts for which a low priority is to be set.

11.1.2 Registers used during standby

The oscillation stabilization time after the standby mode is released is controlled by the oscillation stabilization time select register (OSTS).

Remark For the registers that start, stop, or select the clock, see **CHAPTER 5 CLOCK GENERATORS**.

(1) Oscillation stabilization time select register (OSTS)

This register is used to select oscillation stabilization time of the clock supplied from the oscillator when the STOP mode is released. The wait time set by OSTS is valid only when the crystal/ceramic oscillation clock is selected as the system clock and after the STOP mode is released. If the high-speed internal oscillation or external clock input is selected as the system clock source, no wait time elapses.

The system clock oscillator and the oscillation stabilization time that elapses after power application or release of reset are selected by the option byte. For details, refer to **CHAPTER 15 OPTION BYTE**.

OSTS is set by using the 8-bit memory manipulation instruction.

Figure 11-1. Format of Oscillation Stabilization Time Select Register (OSTS)

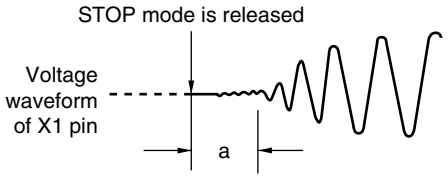
Address: FFF4H, After reset: Undefined, R/W

Symbol	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	0	OSTS1	OSTS0

OSTS1	OSTS0	Selection of oscillation stabilization time
0	0	2^{10}/fx (102.4 μs)
0	1	2^{12}/fx (409.6 μs)
1	0	2^{15}/fx (3.27 ms)
1	1	2^{17}/fx (13.1 ms)

Cautions 1. To set and then release the STOP mode, set the oscillation stabilization time as follows. Expected oscillation stabilization time of resonator ≤ Oscillation stabilization time set by OSTS

2. The wait time after the STOP mode is released does not include the time from the release of the STOP mode to the start of clock oscillation (“a” in the figure below), regardless of whether STOP mode was released by reset signal generation or interrupt generation.



3. The oscillation stabilization time that elapses on power application or after release of reset is selected by the option byte. For details, refer to CHAPTER 15 OPTION BYTE.

Remarks 1. ( ): fx = 10 MHz

2. Determine the oscillation stabilization time of the resonator by checking the characteristics of the resonator to be used.

CHAPTER 14 LOW-VOLTAGE DETECTOR

14.1 Functions of Low-Voltage Detector

The low-voltage detector (LVI) has following functions.

- Compares supply voltage (V_{DD}) and detection voltage (V_{LVI}), and generates an internal interrupt signal or internal reset signal when $V_{DD} < V_{LVI}$.
- Detection levels (ten levels) of supply voltage can be changed by software.
- Interrupt or reset function can be selected by software.
- Operable in STOP mode.

When the low-voltage detector is used to reset, bit 0 (LVIRF) of the reset control flag register (RESF) is set to 1 if reset occurs. For details of RESF, refer to **CHAPTER 12 RESET FUNCTION**.

14.2 Configuration of Low-Voltage Detector

The block diagram of the low-voltage detector is shown in Figure 14-1.

Figure 14-1. Block Diagram of Low-Voltage Detector

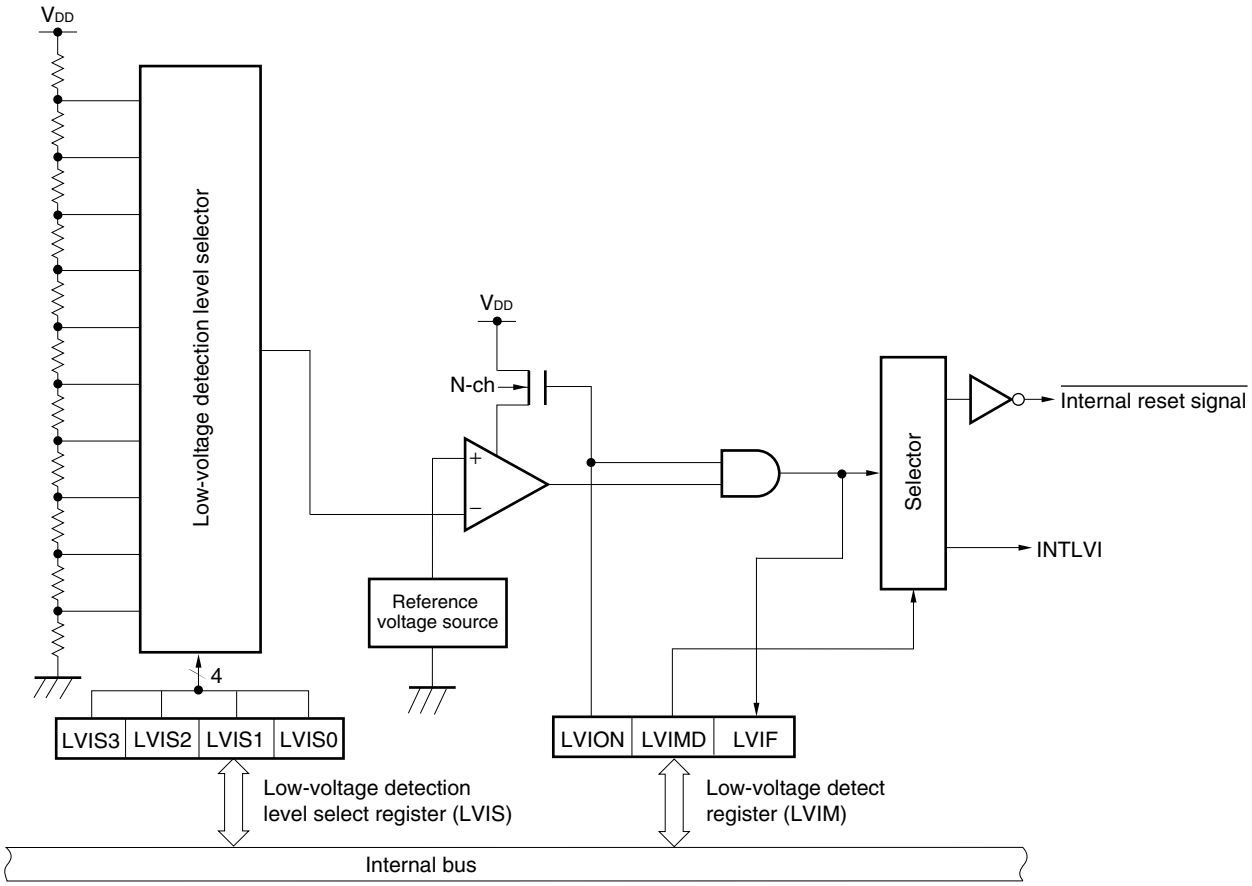
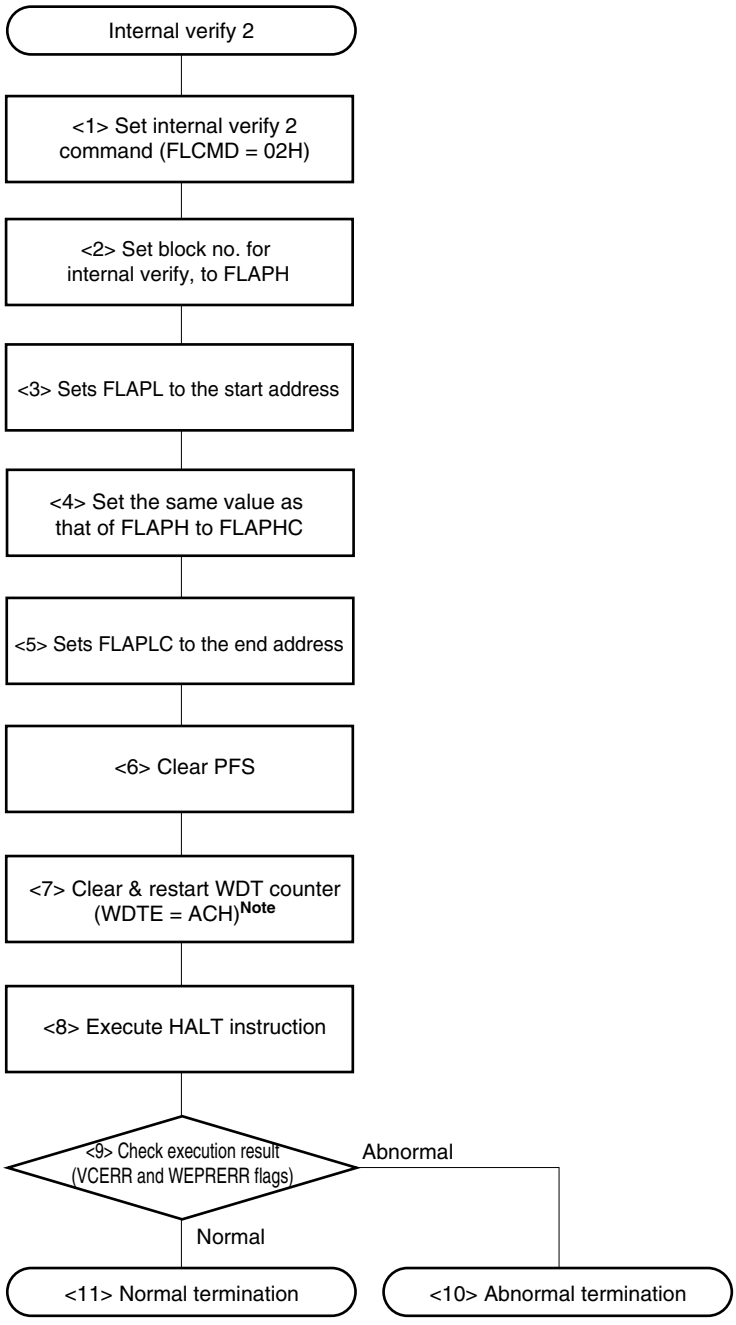


Figure 16-24. Example of Internal Verify 2 Operation in Self Programming Mode



Note This setting is not required when the watchdog timer is not used.

Remark <1> to <11> in Figure 16-24 correspond to Internal verify 2 <1> to <11> in **16.8.9** (the page before last).

An example of a program when the command execution time (from erasure to black check) should be minimized in self programming mode is shown below.

```
;---------------------------------------------------------------------
;START
;---------------------------------------------------------------------
       MOV      MK0,#11111111B   ; Masks all interrupts
       MOV      FLCMD,#00H       ; Clears FLCMD register

       DI

ModeOnLoop:                      ; Configure settings so that the CPU clock ≥ 1 MHz
       MOV      PFS,#00H         ; Clears flash status register
       MOV      PFCMD,#0A5H      ; PFCMD register control
       MOV      FLPMC,#01H       ; FLPMC register control (sets value)
       MOV      FLPMC,#0FEH      ; FLPMC register control (inverts set value)
       MOV      FLPMC,#01H       ; Sets self programming mode with FLPMC register control (sets
                                 ; value)

       NOP
       HALT
       BT       PFS.0,$ModeOnLoop ; Checks completion of write to specific registers
                                  ; Repeats the same processing when an error occurs

FlashBlockErase:
       MOV      FLCMD,#03H       ; Sets flash control command (block erase)
       MOV      FLAPH,#07H       ; Sets number of block to be erased (block 7 is specified
                                 ; here)
       MOV      FLAPL,#00H       ; Fixes FLAPL to “00H”
       MOV      FLAPHC,#07H      ; Sets erase block compare number (same value as that of
                                 ; FLAPH)
       MOV      FLAPLC,#00H      ; Fixes FLAPLC to “00H”

       MOV      WDTE,#0ACH       ; Clears & restarts WDT
       HALT                      ; Self programming is started
       MOV      A,PFS
       CMP      A,#00H
       BNZ      $StatusError     ; Checks erase error
                                 ; Performs abnormal termination processing when an error
                                 ; occurs.

FlashBlockBlankCheck:
       MOV      FLCMD,#04H       ; Sets flash control command (block blank check)
       MOV      FLAPH,#07H       ; Sets number of block for blank check (block 7 is specified
                                 ; here)
       MOV      FLAPL,#00H       ; Fixes FLAPL to “00H”
```

Mnemonic	Operand	Bytes	Clocks	Operation	Flag		
					Z	AC	CY
MOVW	rp, #word	3	6	rp ← word			
	AX, saddrp	2	6	AX ← (saddrp)			
	saddrp, AX	2	8	(saddrp) ← AX			
	AX, rp Note	1	4	AX ← rp			
	rp, AX Note	1	4	rp ← AX			
XCHW	AX, rp Note	1	8	AX ↔ rp			
ADD	A, #byte	2	4	A, CY ← A + byte	×	×	×
	saddr, #byte	3	6	(saddr), CY ← (saddr) + byte	×	×	×
	A, r	2	4	A, CY ← A + r	×	×	×
	A, saddr	2	4	A, CY ← A + (saddr)	×	×	×
	A, !addr16	3	8	A, CY ← A + (addr16)	×	×	×
	A, [HL]	1	6	A, CY ← A + (HL)	×	×	×
	A, [HL + byte]	2	6	A, CY ← A + (HL + byte)	×	×	×
ADDC	A, #byte	2	4	A, CY ← A + byte + CY	×	×	×
	saddr, #byte	3	6	(saddr), CY ← (saddr) + byte + CY	×	×	×
	A, r	2	4	A, CY ← A + r + CY	×	×	×
	A, saddr	2	4	A, CY ← A + (saddr) + CY	×	×	×
	A, !addr16	3	8	A, CY ← A + (addr16) + CY	×	×	×
	A, [HL]	1	6	A, CY ← A + (HL) + CY	×	×	×
	A, [HL + byte]	2	6	A, CY ← A + (HL + byte) + CY	×	×	×
SUB	A, #byte	2	4	A, CY ← A – byte	×	×	×
	saddr, #byte	3	6	(saddr), CY ← (saddr) – byte	×	×	×
	A, r	2	4	A, CY ← A – r	×	×	×
	A, saddr	2	4	A, CY ← A – (saddr)	×	×	×
	A, !addr16	3	8	A, CY ← A – (addr16)	×	×	×
	A, [HL]	1	6	A, CY ← A – (HL)	×	×	×
	A, [HL + byte]	2	6	A, CY ← A – (HL + byte)	×	×	×

Note Only when rp = BC, DE, or HL.

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

Mnemonic	Operand	Bytes	Clocks	Operation	Flag		
					Z	AC	CY
SUBC	A, #byte	2	4	A, CY ← A – byte – CY	×	×	×
	saddr, #byte	3	6	(saddr), CY ← (saddr) – byte – CY	×	×	×
	A, r	2	4	A, CY ← A – r – CY	×	×	×
	A, saddr	2	4	A, CY ← A – (saddr) – CY	×	×	×
	A, !addr16	3	8	A, CY ← A – (addr16) – CY	×	×	×
	A, [HL]	1	6	A, CY ← A – (HL) – CY	×	×	×
	A, [HL + byte]	2	6	A, CY ← A – (HL + byte) – CY	×	×	×
AND	A, #byte	2	4	A ← A ∧ byte	×		
	saddr, #byte	3	6	(saddr) ← (saddr) ∧ byte	×		
	A, r	2	4	A ← A ∧ r	×		
	A, saddr	2	4	A ← A ∧ (saddr)	×		
	A, !addr16	3	8	A ← A ∧ (addr16)	×		
	A, [HL]	1	6	A ← A ∧ (HL)	×		
	A, [HL + byte]	2	6	A ← A ∧ (HL + byte)	×		
OR	A, #byte	2	4	A ← A ∨ byte	×		
	saddr, #byte	3	6	(saddr) ← (saddr) ∨ byte	×		
	A, r	2	4	A ← A ∨ r	×		
	A, saddr	2	4	A ← A ∨ (saddr)	×		
	A, !addr16	3	8	A ← A ∨ (addr16)	×		
	A, [HL]	1	6	A ← A ∨ (HL)	×		
	A, [HL + byte]	2	6	A ← A ∨ (HL + byte)	×		
XOR	A, #byte	2	4	A ← A ⊻ byte	×		
	saddr, #byte	3	6	(saddr) ← (saddr) ⊻ byte	×		
	A, r	2	4	A ← A ⊻ r	×		
	A, saddr	2	4	A ← A ⊻ (saddr)	×		
	A, !addr16	3	8	A ← A ⊻ (addr16)	×		
	A, [HL]	1	6	A ← A ⊻ (HL)	×		
	A, [HL + byte]	2	6	A ← A ⊻ (HL + byte)	×		

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

18.3 Instructions Listed by Addressing Type

(1) 8-bit instructions

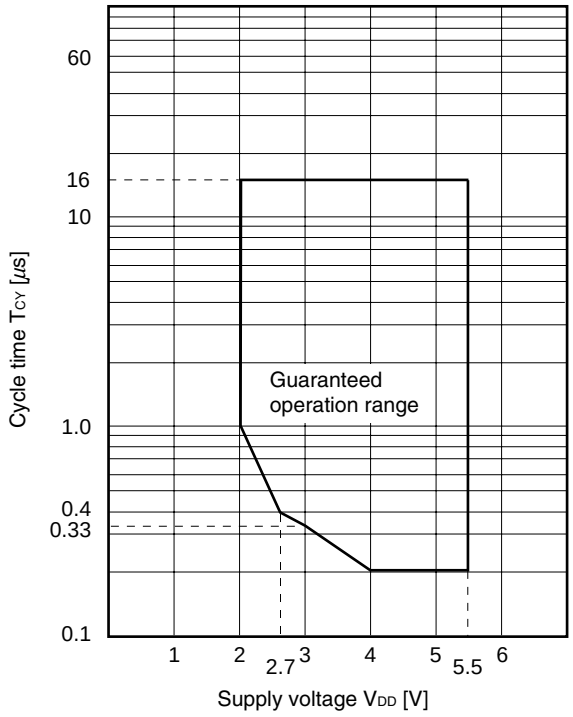
MOV, XCH, ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP, INC, DEC, ROR, ROL, RORC, ROLC, PUSH, POP, DBNZ

1st Operand \ 2nd Operand	#byte	A	r	sfr	saddr	!addr16	PSW	[DE]	[HL]	[HL + byte]	$addr16	1	None
A	ADD ADDC SUB SUBC AND OR XOR CMP		MOV[Note] XCH[Note] ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV ADD ADDC SUB SUBC AND OR XOR CMP	MOV	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP		ROR ROL RORC ROLC	
r	MOV	MOV											INC DEC
B, C											DBNZ		
sfr	MOV	MOV											
saddr	MOV ADD ADDC SUB SUBC AND OR XOR CMP	MOV									DBNZ		INC DEC
!addr16		MOV											
PSW	MOV	MOV											PUSH POP
[DE]		MOV											
[HL]		MOV											
[HL + byte]		MOV											

Note Except r = A.

Standard product, (A) grade product T_A = –40 to +85°C

T_{CY} vs. V_{DD} (Crystal/Ceramic Oscillation Clock, External Clock Input)



T_{CY} vs. V_{DD} (High-speed internal oscillator Clock)