OTP-Based 8-Bit Microcontroller with 12 bit ADC

Devices Included in this Data Sheet:

- FM8PA76AE : 20-pin OTP device
- FM8PA76BE : 14-pin OTP device
- FM8PA76DE : 16-pin OTP device
- FM8PA76EE : 24-pin OTP device with VR pin
- FM8PA76FE : 16-pin OTP device with VR pin

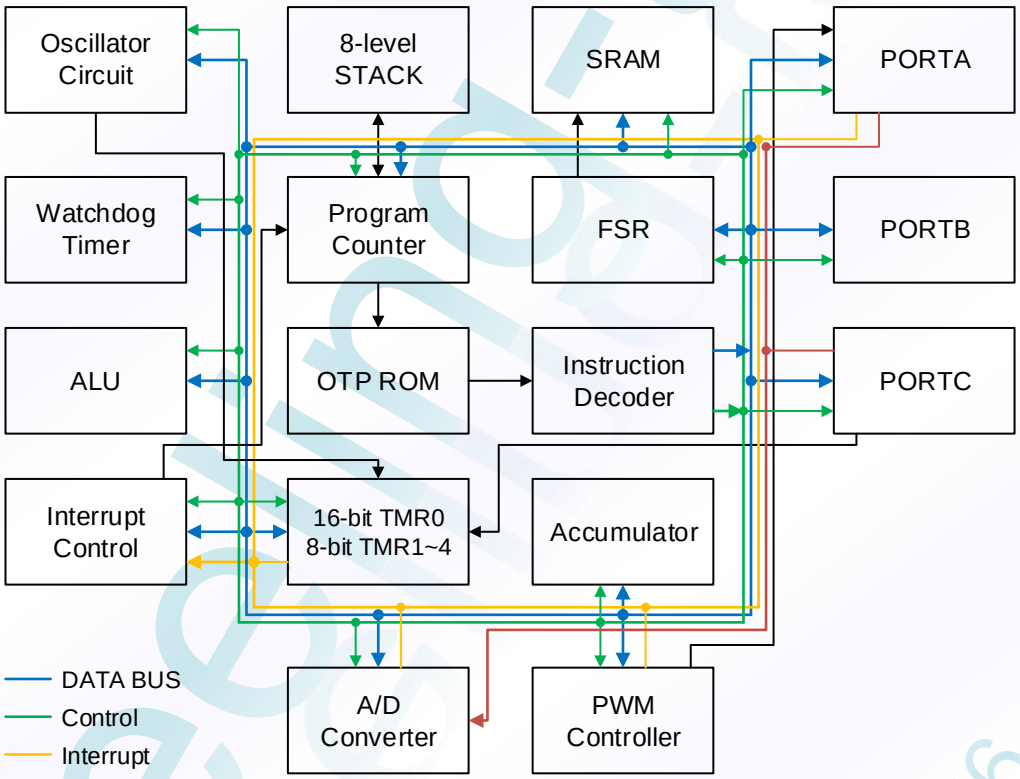
FEATURES

- Total 9 channel 12bit AD converter with ±2LSB resolution
- All instructions are single cycle except for program branches which are two-cycles
- All OTP area LGOTO instruction
- All OTP area subroutine LCALL instruction
- 8-bit wide data path
- 8-level deep hardware stack
- 2K x 16 bits on chip OTP
- 45x8 bits on chip special purpose registers and 128 x 8 bits on chip general purpose registers (SRAM)
- Operating speed: DC-20 MHz clock input, or DC-100 ns instruction cycle
- Direct, indirect addressing modes for data accessing
- Five real time up-count Timer/Counter with 3-bit programmable prescaler
  - TMR0: 16-bit Timer (up-counter)
  - TMR1: 8-bit, PWM1 (Period) & Timer
  - TMR2: 8-bit, PWM1 (Duty) & Timer
  - TMR3: 8-bit, PWM2 (Period) & Timer
  - TMR4: 8-bit, PWM2 (Duty) & Timer
- Built-in 3 levels Low Voltage Detector (LVDT) (2.2V/2.6V/3.7V) for Brown-out Reset (BOR)
- Power-up Reset Timer (PWRT)
- On chip Watchdog Timer (WDT) with internal oscillator for reliable operation and soft-ware watch-dog enable/disable control
- Three I/O ports Port A, Port B and Port C with independent direction control
  - 17 Bi-direction I/O port (Programmable Pull-up enable in Input mode)
  - One Input only port (IOB2/RSTB)
- Four kinds of interrupt source: 5 Timers/Counters, 8 external interrupt sources: IOA0~IOA7, Internal watchdog timer (i_WDT) wakeup, and A/D end of conversion
- Wake-up from SLEEP:
  - Port A (IOA0~IOA7) pin change wakeup
  - WDT overflow
  - i_WDT overflow
- Power saving SLEEP mode
- Programmable Code Protection
- Selectable oscillator options:
  - ERC: External Resistor/ Voltage Controlled Oscillator
  - XT: Crystal/Resonator Oscillator
  - HF: High Frequency Crystal/Resonator Oscillator
  - LF: Low Frequency Crystal Oscillator
  - IRC: Internal Resistor/Capacitor Oscillator
- Wide-operating voltage range:
  - OTP: 2.2V to 5.5V

GENERAL DESCRIPTION

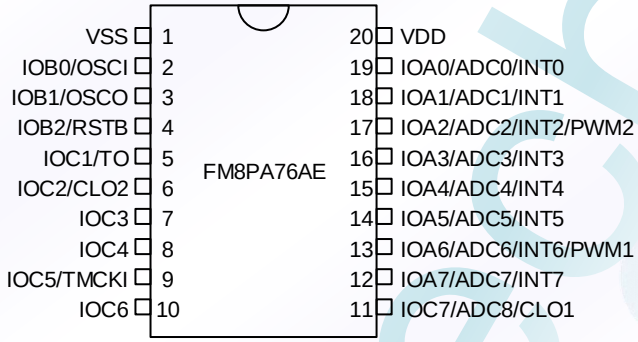
The FM8PA76 is a low-cost, high speed, high noise immunity, OTP-based 8-bit CMOS microcontrollers. It employs a RISC architecture with 54 instructions. All instructions are single cycle except for program branches which take two cycles. The easy to use and easy to remember instruction set reduces development time significantly.
The FM8PA76 consists of Power-on Reset (POR), Brown-out Reset (BOR), Power-up Reset Timer (PWRT), Watchdog Timer, OTP, SRAM, tri-state I/O port, I/O pull-high control, Power saving SLEEP mode, 5 real time programmable clock/counter, Interrupt, Wake-up from SLEEP mode, and Code Protection for OTP products. There are eight oscillator configurations to be chosen from, including the power-saving LF (Low Frequency) oscillator and cost saving internal RC oscillator.
The FM8PA76 address 2K×16 of program memory.
The FM8PA76 can directly or indirectly address its register files and data memory. All special function registers including the program counter are mapped in the data memory.
The FM8PA76 provides total 9 channel 12bit AD converter with ±2LSB resolution.

BLOCK DIAGRAM

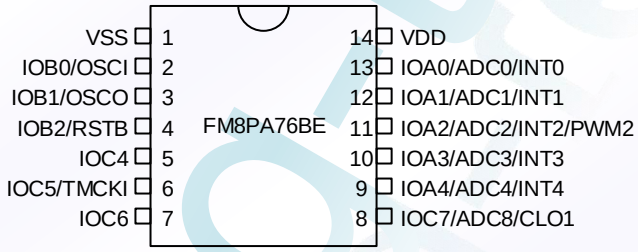


PIN CONNECTION

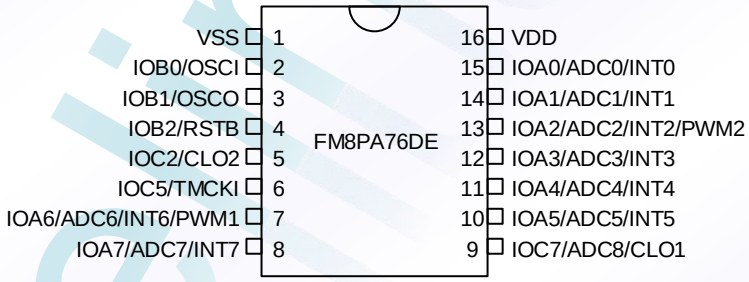
PDIP20, SOP20



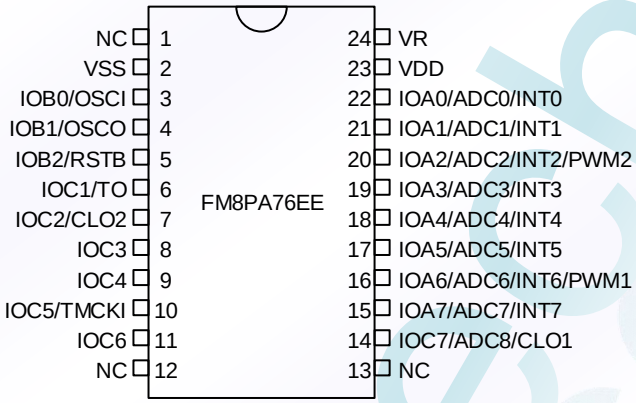
PDIP14, SOP14



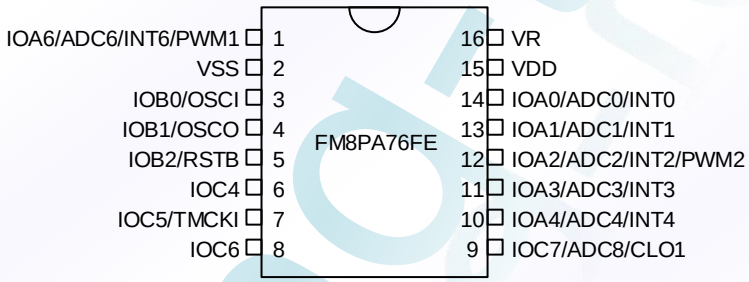
PDIP16, SOP16



PDIP24, SOP24 (With VR PIN)



PDIP16, SOP16 (With VR PIN)



PIN DESCRIPTIONS

Name	I/O	Description
IOA0/AD0/INT0 ~ IOA7/AD7/INT7	I/O	• Bi-direction I/O port (programmable Pull-high in Input mode) • Wake-up on pin change • External interrupt input • A/D converter input • IOA2 is PWM2 output • IOA6 is PWM1 output
IOB0/OSCI	I/O	• Bi-direction I/O port (programmable Pull-high in Input mode) • Oscillator input (HF, XT, LF, ERC mode)
IOB1/OSCO	I/O	• Bi-direction I/O port (programmable Pull-high in Input mode) • Oscillator output (HF, XT, LF, ERC mode)
IOB2/RSTB	I	• Input port • System clear (RESET) input. This pin is an active low RESET to the device, **the voltage on this pin must not exceed VDD.**
IOC1/TO~ IOC2/CLO2	I/O	• IOC1~IOC2 is Bi-direction I/O port (programmable Pull-high in Input mode) • TO (PWM2 interrupt/2) shared with IOC1 • Clock output 2 with prescaler shared with IOC2
IOC3, IOC4, IOC6	I/O	Bi-direction I/O port (programmable Pull-high in Input mode)
IOC5/TMCKI	I/O	• Bi-direction I/O port (programmable Pull-high in Input mode) • TMCKI (External clock input) shared with IOC5
IOC7/AD8/CLO1	I/O	• Bi-direction I/O port (programmable Pull-high in Input mode) • A/D converter input • CLO1 (system clock out) shared with IOC7
VR	-	ADC module reference input, **The voltage on this pin must not exceed VDD.**
VDD	-	Positive supply
VSS	-	Ground

Legend: I=input, O=output, I/O=input/output

Note: Please refer to 2.2 for detail IO type description

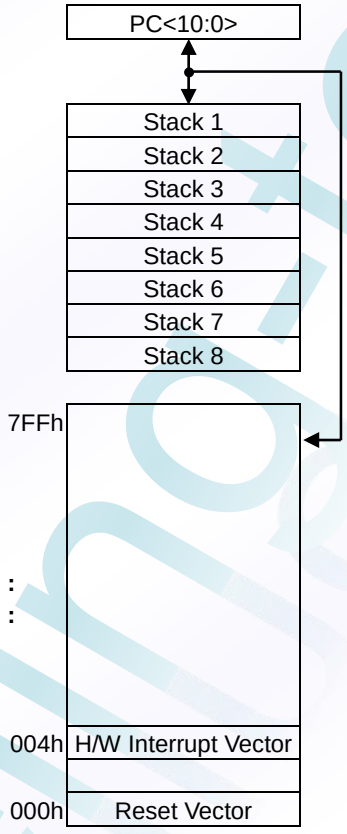
1.0 MEMORY ORGANIZATION

FM8PA76 memory is organized into program memory and data memory.

1.1 Program Memory Organization

The FM8PA76 has a 11-bit Program Counter capable of addressing a 2K×16 program memory space.
The RESET vector for the FM8PA76 is at 000h.
The H/W interrupt vector is at 004h.
User can use “LCALL (far call)/LGOTO (far goto)” instructions to program user's code within entire program area.

Figure 1.1: Program Memory Map and STACK



FM8PA76

1.2 Data Memory Organization

Data memory is composed of 45 bytes Special Function Registers and 128 bytes General Purpose Registers. The data memory can be accessed either directly or indirectly through the FSR register.

Table 1.1: Registers File Map for FM8PA76

Address	Description
00h : : 3Eh	Special Purpose Register
40h : : BFh	General Purpose Register

Table 1.2: Special Purpose Registers Map

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
System									
00h (r/w)	INDF	Uses contents of FSR to address data memory (not a physical register)							
01h (r/w)	PCL	Low order 8 bits of PC							
02h (r/w)	PCHBUF	-	-	-	-	-	High order 3 bits of PC		
03h (r/w)	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C
04h (r/w)	FSR	Indirect data memory address pointer							
IO PAD & CONTROL									
05h (r/w)	IOSTA	IOSTA7	IOSTA6	IOSTA5	IOSTA4	IOSTA3	IOSTA2	IOSTA1	IOSTA0
06h (r/w)	PORTA	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
07h (r/w)	IOSTB	-	-	-	-	-	-	IOSTB1	IOSTB0
08h (r/w)	PORTB	-	-	-	-	-	IOB2	IOB1	IOB0
09h (r/w)	IOSTC	IOSTC7	IOSTC6	IOSTC5	IOSTC4	IOSTC3	IOSTC2	IOSTC1	-
0Ah (r/w)	PORTC	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	-
Timer0: 16-bit timer									
10h (r/w)	TMR0_CTL	T0EN	T0LOAD	T0SO1	T0SO0	T0EDGE	T0PS2	T0PS1	T0PS0
11h (r/w)	TMR0L_LA	16-bit real-time timer/counter latch Low byte							
12h (r/w)	TMR0H_LA	16-bit real-time timer/counter latch High byte							
13h (r)	TMR0L_CNT	16-bit real-time timer/counter count Low byte							
14h (r)	TMR0H_CNT	16-bit real-time timer/counter count High byte							
Timer1: 8-bit Timer & PWM1 Period									
15h (r/w)	TMR1_CTL1	T1EN	T1LOAD	T1SO1	T1SO0	T1EDGE	T1PS2	T1PS1	T1PS0
16h (r/w)	TMR1_CTL2	T12MOD	PWM1_INI	-	-	PWM1R3	PWM1R2	PWM1R1	PWM1R0
17h (r/w)	TMR1_LA	8-bit real-time timer/counter Latch							
18h (r)	TMR1_CNT	8-bit real time timer/counter Count							
Timer2: 8-bit Timer & PWM1 Duty									
19h (r/w)	TMR2_CTL1	T2EN	T2LOAD	T2SO1	T2SO0	T2EDGE	T2PS2	T2PS1	T2PS0
1Ah (r/w)	TMR2_LA	8-bit real-time timer/counter Latch							
1Bh (r)	TMR2_CNT	8-bit real time timer/counter Count							

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
Timer3: 8-bit Timer & PWM2 Period									
1Ch (r/w)	TMR3_CTL1	T3EN	T3LOAD	T3SO1	T3SO0	T3EDGE	T3PS2	T3PS1	T3PS0
1Dh (r/w)	TMR3_CTL2	T34MOD	PWM2_INI	-	-	PWM2R3	PWM2R2	PWM2R1	PWM2R0
1Eh (r/w)	TMR3_LA	8-bit real-time timer/counter Latch							
1Fh (r)	TMR3_CNT	8-bit real-time timer/counter Count							
Timer4: 8-bit Timer & PWM2 Duty									
20h (r/w)	TMR4_CTL1	T4EN	T4LOAD	T4SO1	T4SO0	T4EDGE	T4PS2	T4PS1	T4PS0
21h (r/w)	TMR4_LA	8-bit real-time timer/counter Latch							
22h (r/w)	TMR4_CNT	8-bit real-time timer/counter Count							
IRQ									
25h (r/w)	INTEN	GIE	ADCIE	PAIE	T4IE	T3_PWM2IE	T2IE	T1_PWM1IE	T0IE
26h (r/w)	INTFLAG	-	ADCIF	PAIF	T4IF	T3_PWM2IF	T2IF	T1_PWM1IF	T0IF
ADC Control									
29h (r/w)	AD_CTL1	ADCEN	-	MODE	-	CHSL3	CHSL2	CHSL1	CHSL0
2Ah (r/w)	AD_CTL2	CMP_D	-	-	-	-	CLKSL2	CLKSL1	CLKSL0
2Bh (r/w)	AD_CTL3	-	-	-	-	ANISL3	ANISL2	ANISL1	ANISL0
2Ch (r)	AD_DATL	D3	D2	D1	D0	-	-	-	-
2Dh (r)	AD_DATH	D11	D10	D9	D8	D7	D6	D5	D4
Others									
2Fh (r/w)	SYS_CLK	CLKS	-	-	-	-	-	IRCPD	ECLKPD
30h (r/w)	CLO_CTL	CLO2SO	CLO2PS1	CLO2PS0	-	EXT_CLK	CLO2_E	CLO1_E	TO_E
31h (r/w)	APHCON	PHA7	PHA6	PHA5	PHA4	PHA3	PHA2	PHA1	PHA0
32h (r/w)	BPHCON	-	-	-	-	-	-	PHB1	PHB0
33h (r/w)	CPHCON	PHC7	PHC6	PHC5	PHC4	PHC3	PHC2	PHC1	-
3Ah (r/w)	INT_PA	PA7IEN	PA6IEN	PA5IEN	PA4IEN	PA3IEN	PA2IEN	PA1IEN	PA0IEN
3Dh (r/w)	WDT_CTL	WDTEN	I_WDT	I_TWDT	-	-	WDTPS2	WDTPS1	WDTPS0
3Eh (r/w)	TB_BNK	-	-	-	-	-	BNK2	BNK1	BNK0

Legend: - = unimplemented, read as '0'.

2.0 FUNCTIONAL DESCRIPTIONS

2.1 Operational Registers

2.1.1 INDF (Indirect Addressing Register)

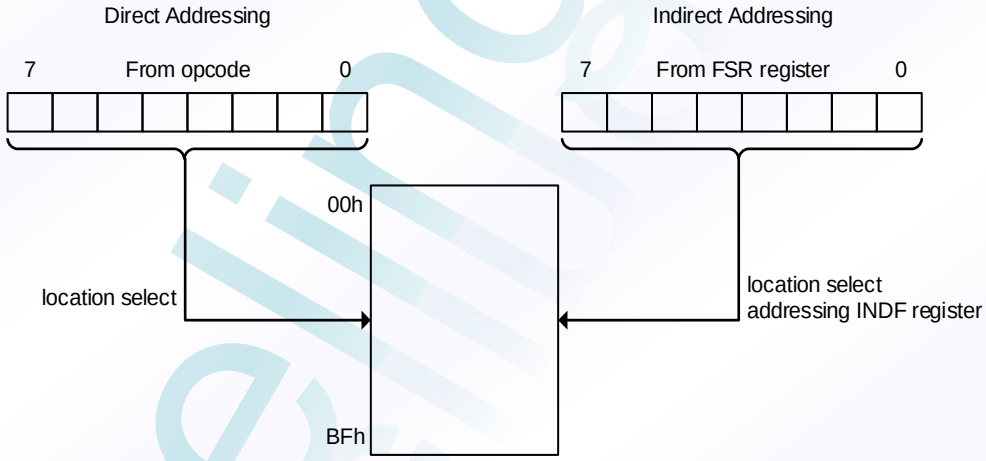
Read/Write-POR		R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
00h	INDF	Uses contents of FSR to address data memory (not a physical register)							

Legend: x = unknown, more bits default state, please refer to Table 2.1.

The INDF Register is not a physical register. Any instruction accessing the INDF register can actually access the register pointed by FSR Register. Reading the INDF register itself indirectly (FSR="0") will read 00h. Writing to the INDF register indirectly results in a no-operation (although status bits may be affected).

Example 2.1: INDIRECT ADDRESSING
Register file 48 contains the value 10h
Register file 49 contains the value 0Ah
Load the value 48 into the FSR Register
A read of the INDF Register will return the value of 10h
Increment the value of the FSR Register by one (@FSR=49h)
A read of the INDF register now will return the value of 0Ah.

Figure 2.1: Direct/Indirect Addressing for FM8PA76



2.1.2 PCL / PCHBUF (Low / High Bytes of Program Counter) & Stack

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
01h	PCL	Low order 8 bits of PC							

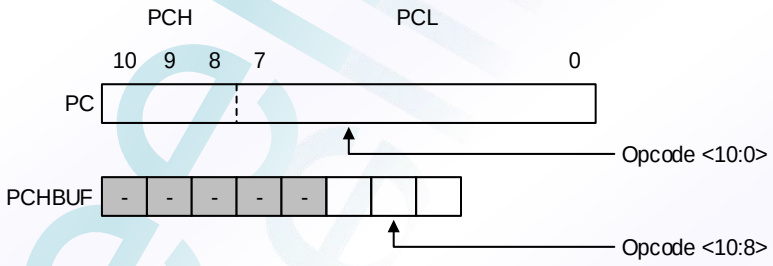
Read/Write-POR		-	-	-	-	-	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
02h	PCHBUF	-	-	-	-	-	High order 3 bits of PC		

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

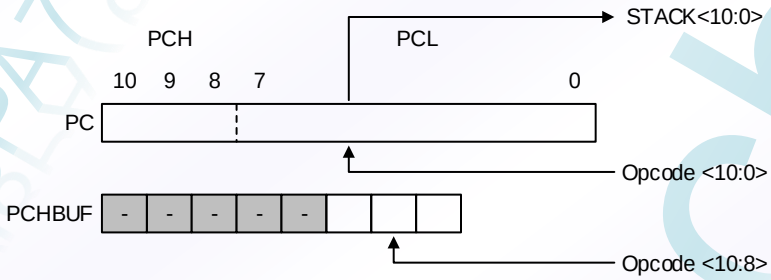
FM8P767 device has an 11-bit wide Program Counter (PC) and eight-level deep 11-bit hardware push/pop stack. The low byte of PC is called the PCL register. This register is readable and writable. The high byte of PC is called the PCH register. This register contains the PC<10:8> bits and is not directly readable or writable. All updates to the PCH register go through the PCHBUF register. As a program instruction is executed, the Program Counter will contain the address of the next program instruction to be executed. The PC value is increased by one, every instruction cycle, unless an instruction changes the PC.
For a LGOTO instruction, the PC<10:0> is provided by the LGOTO instruction word. The PCL register is mapped to PC<7:0>, while PCHBUF register update for PC<10:8>.
For a LCALL instruction, the PC<10:0> is provided by the LCALL instruction word. The next PC will be loaded (PUSHed) onto the top of STACK. The PCL register is mapped to PC<7:0>, while PCHBUF register update for PC<10:8>.
For a RETF or RETFIE instruction, the PC are updated (POPed) from the top of STACK. The PCL register is mapped to PC<7:0>, and the PCHBUF register is not updated.
For a RETIA or RETURN instruction, the PC are updated (POPed) from the top of STACK. The PCL register is mapped to PC<7:0>, while PCHBUF register update for PC<10:8>.
For any instruction where the PCL is the destination, the PC<7:0> is provided by the instruction word or ALU result. However, the PC<10:8> will come from the PCHBUF<2:0> bits (PCHBUF → PCH). If the result of the ALU operation has resulted in a carry, the carry will be updated to PCHBUF and PCH.
PCHBUF only when the PCL is written, will be updated to the PCH.

Figure 2.2: Loading of PC in Different Situations

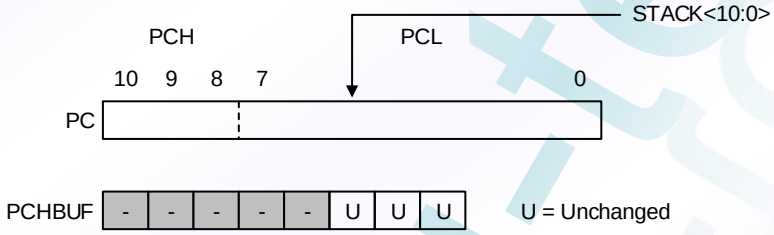
Situation 1: LGOTO Instruction



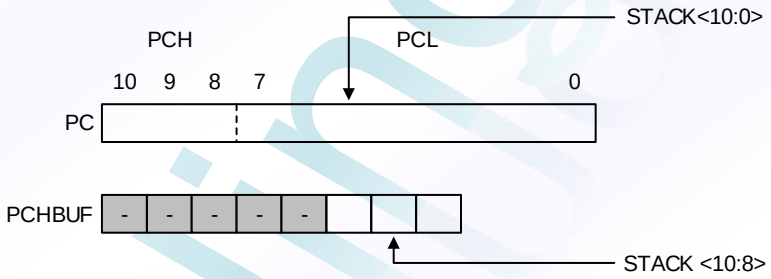
Situation 2: LCALL Instruction



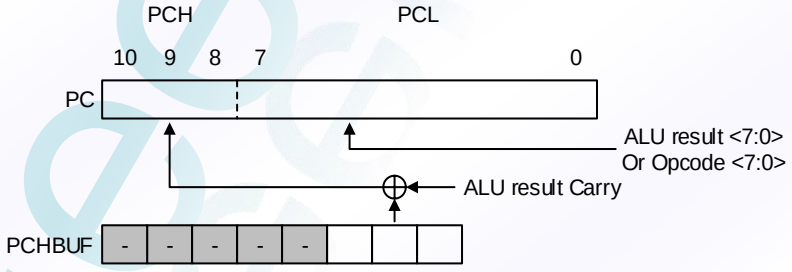
Situation 3: RETF or RETFIE Instruction



Situation 4: RETURN or RETIA Instruction



Situation 4: Instruction with PCL as destination



2.1.3 STATUS (Status Register)

Read/Write-POR		-	-	-	R-#	R-#	R/W-x	R/W-x	R/W-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
03h	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C

Legend: - = unimplemented, read as '0', x = unknown, # refer Table 2.3 for detail description, more bits default state, please refer to Table 2.1.

This register contains the arithmetic status of the ALU, the RESET status.
If the STATUS Register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS Register as destination may be different than intended. For example, CLRR STATUS will clear the upper three bits and set the Z bit. This leaves the STATUS Register as 000u u1uu (where u = unchanged).

C : Carry/borrow bit.
ADDAR
= 1, Carry occurred.
= 0, No Carry occurred.
SUBAR
= 1, No borrow occurred.
= 0, Borrow occurred.
Note : A subtraction is executed by adding the two's complement of the second operand. For rotate (RRR, RLR) instructions, this bit is loaded with either the high or low order bit of the source register.

DC : Half carry/half borrow bit
ADDAR
= 1, Carry from the 4th low order bit of the result occurred.
= 0, No Carry from the 4th low order bit of the result occurred.
SUBAR
= 1, No Borrow from the 4th low order bit of the result occurred.
= 0, Borrow from the 4th low order bit of the result occurred.

Z : Zero bit.
= 1, The result of a logic operation is zero.
= 0, The result of a logic operation is not zero.

$\overline{PD}$: Power down flag bit.
= 1, after power-up or by the CLRWDT instruction.
= 0, by the SLEEP instruction.

$\overline{TO}$: Watch-dog timer overflow flag bit.
= 1, after power-up or by the CLRWDT or SLEEP instruction
= 0, a watch-dog time overflow occurred

2.1.4 FSR (Indirect Data Memory Address Pointer)

Read/Write-POR		R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
04h	FSR	Indirect data memory address pointer							

Legend: x = unknown, more bits default state, please refer to Table 2.1.

Bit7:Bit0 : Select registers address in the indirect addressing mode. See 2.1.1 for detail description.

2.1.5 PORTA, PORTB, PORTC, IOSTA, IOSTB and IOSTC (Port Data Registers and Port Direction Control Registers)

Read/Write-POR		R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
05h	IOSTA	IOSTA7	IOSTA6	IOSTA5	IOSTA4	IOSTA3	IOSTA2	IOSTA1	IOSTA0

Read/Write-POR		R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
06h	PORTA	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0

Read/Write-POR		-	-	-	-	-	-	R/W-1	R/W-1
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
07h	IOSTB	-	-	-	-	-	-	IOSTB1	IOSTB0

Read/Write-POR		-	-	-	-	-	R/W-x	R/W-x	R/W-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
08h	PORTB	-	-	-	-	-	IOB2	IOB1	IOB0

Read/Write-POR		R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	-
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
09h	IOSTC	IOSTC7	IOSTC6	IOSTC5	IOSTC4	IOSTC3	IOSTC2	IOSTC1	-

Read/Write-POR		R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	-
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Ah	PORTC	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	-

Legend: - = unimplemented, read as '0', x = unknown, more bits default state, please refer to Table 2.1.

The registers (IOSTA, IOSTB and IOSTC) are used to define the input or output of each port.

= 1, Input.

= 0, Output.

Reading the port (PORTA, PORTB and PORTC register) reads the status of the pins independent of the pin's input/output modes. Writing to these ports will write to the port data latch. Please refer to 2.2 for detail I/O Port description.

Note: IOB2 is read only.

2.1.6 TMR0: 16-bits Time Clock/Counter

The Timer0 is a 16-bit up count timer/counter which includes high byte (TMR0H_CNT), low byte (TMR0L_CNT) counter register, high byte (TMR0H_LA), and low byte (TMR0L_LA) latch register. Please refer to 2.3 for detail Timer description.

2.1.6.1 TMR0_CTL (Timer0 Control Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
10h	TMR0_CTL	T0EN	T0LOAD	T0SO1	T0SO0	T0EDGE	T0PS2	T0PS1	T0PS0

Note: more bits default state, please refer to Table 2.1.

T0EN : TMR0 Enable/Disable
= 1, TMR0 Enable.
= 0, TMR0 Disable.

T0LOAD : Enable/Disable Latch Buffer automatically load to counter register while writing to latch register
= 1, Enable TMR0 latch buffer automatically load to counter register while writing to latch register.
= 0, Disable TMR0 latch buffer automatically load to counter register while writing to latch register.
Note: This bit is only affected after latch register written. When the timer underflows, the latch register data will automatically load into counter register.

T0SO1:T0SO0 : TMR0 clock source selection

T0SO1	T0SO0	TMR0 clock source
0	0	TMCKI(IOC5)
0	1	Crystal mode OSCI or EXT_RC(In dual RC clock mode)
1	0	Internal 4MHz RC
1	1	No function, don't use

T0EDGE : TMR0 clock edge selection. This bit works only when external clock source TMCKI (IOC5) selected.
= 1, TMR0 increased while external clock H→L (Falling edge).
= 0, TMR0 increased while external clock L→H (Rising edge).

T0PS2:T0PS0 : TMR0 Prescaler selection

T0PS2	T0PS1	T0PS0	TMR0 Prescal rate
0	0	0	1:1
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

2.1.6.2 TMR0L_LA & TMR0H_LA (Timer0 Latch High & Low byte Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
11h	TMR0L_LA	16-bit real-time timer/counter latch Low byte							

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
12h	TMR0H_LA	16-bit real-time timer/counter latch High byte							

Note: more bits default state, please refer to Table 2.1.

TMR0L_LA and TMR0H_LA are Timer0 pre-set latch buffer, please don't write FFFFh to these registers, otherwise it will generate an error. See 2.3 for detail description.

2.1.6.3 TMR0L_CNT & TMR0H_CNT (Timer0 Counter High & Low byte Register)

Read/Write-POR		R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
13h	TMR0L_CNT	16-bit real-time timer/counter count Low byte							

Read/Write-POR		R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
14h	TMR0H_CNT	16-bit real-time timer/counter count High byte							

Note: more bits default state, please refer to Table 2.1.

T0CNT_L and T0CNT_H are Timer1 real-time counter, these register is only read, see 2.3 for detail description.

2.1.7 TMR1: 8-bit Timer & PWM1 Period

The Timer1 is an 8-bit up count timer/counter which includes counter register TMR1_CNT, and latch register TMR1_LA. Please refer to 2.3 for detail Timer description.
The Timer1 can also be combined with Timer2 as PWM1 period and duty and controlled by the register TMR1_CTL2. Please refer to 2.4 for detail PWM description.

2.1.7.1 TMR1_CTL1 (Timer1 Control Register1)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
15h	TMR1_CTL1	T1EN	T1LOAD	T1SO1	T1SO0	T1EDGE	T1PS2	T1PS1	T1PS0

Note: more bits default state, please refer to Table 2.1.

T1EN : TMR1 (PWM1) Enable/Disable
= 1, TMR1 (PWM1) Enable.
= 0, TMR1 (PWM1) Disable.

T1LOAD : Enable/Disable Latch Buffer automatically load to counter register while writing to latch register
= 1, Enable TMR1 latch buffer automatically load to counter register while writing to latch register.
= 0, Disable TMR1 latch buffer automatically load to counter register while writing to latch register.
Note: This bit is only affected after latch register written. When the timer underflows, the latch register data will automatically load into counter register.

T1SO1:T1SO0 : TMR1 clock source selection

T1SO1	T1SO0	TMR1 clock source
0	0	TMCKI(IOC5)
0	1	Crystal mode OSCI or EXT_RC(In dual RC clock mode)
1	0	Internal 4MHz RC
1	1	No function, don't use.

T1EDGE : TMR1 clock edge selection. This bit works only when external clock source TMCKI (IOC5) selected.
= 1, TMR1 increased while external clock H→L (Falling edge).
= 0, TMR1 increased while external clock L→H (Rising edge).

T1PS2:T1PS0 : TMR1 Prescaler selection

T1PS2 : T1PS0			TMR1 Prescal rate
0	0	0	1:1
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

2.1.7.2 TMR1_CTL2 (Timer1 Control Register2)

Read/Write-POR		R/W-0	R/W-0	-	-	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
16h	TMR1_CTL2	T12MOD	PWM1_INI	-	-	PWM1R3	PWM1R2	PWM1R1	PWM1R0

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

T12MOD : TMR1 and TMR2 working mode (TMR / PWM1)
= 1, TMR1 and TMR2 is PWM1.
= 0, TMR1 and TMR2 is Timer.

PW1_INI : Initial State of PWM1 output duty.
= 1, Set the initial state to L, change to H when TMR2 duty overflow.
= 0, Set the initial state to H, change to L when TMR2 duty overflow.

PWM1R3:PWM1R0 : Interrupt Event Rate of PWM1.
"1:N" means interrupt occurred after "N" PWM1 pulses.

PWM1R3 : PWM1R0				PWM1 Interrupt rate
0	0	0	0	1:1
0	0	0	1	1:2
0	0	1	0	1:3
0	0	1	1	1:4
		|		|
1	1	0	1	1:14
1	1	1	0	1:15
1	1	1	1	1:16

2.1.7.3 TMR1_LA (Timer1 Latch Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
17h	TMR1_LA	8-bit real-time timer/counter Latch							

Note: more bits default state, please refer to Table 2.1.

TMR1_LA is a Timer1 pre-set latch buffer, please don't write FFh to this register, otherwise it will generate an error. See 2.3 for detail description.

2.1.7.4 TMR1_CNT (Timer1 Counter Register)

Read/Write-POR		R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
18h	TMR1_CNT	8-bit real time timer/counter Count							

Note: more bits default state, please refer to Table 2.1.

TMR1_CNT is a Timer1 real-time counter, this register is only read, see 2.3 for detail description.

2.1.8 TMR2: 8-bit Timer & PWM1 Duty

The Timer2 is an 8-bit up count timer/counter which includes counter register TMR2_CNT, and latch register TMR2_LA. Please refer to 2.3 for detail Timer description.

2.1.8.1 TMR2_CTL1 (Timer2 Control Register1)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
19h	TMR2_CTL1	T2EN	T2LOAD	T2SO1	T2SO0	T2EDGE	T2PS2	T2PS1	T2PS0

Note: more bits default state, please refer to Table 2.1.

T2EN : TMR2 Enable/Disable
= 1, TMR2 Enable.
= 0, TMR2 Disable.
Note: At PWM mode, Timer2 is controlled by T1EN.

T2LOAD : Enable/Disable Latch Buffer automatically load to counter register while writing to latch register
= 1, Enable TMR2 latch buffer automatically load to counter register while writing to latch register.
= 0, Disable TMR2 latch buffer automatically load to counter register while writing to latch register.
Note: This bit is only affected after latch register written. When the timer underflows, the latch register data will automatically load into counter register.

T2SO1:T2SO0 : TMR2 clock source selection

T2SO1	T2SO0	TMR2 clock source
0	0	TMCKI(IOC5)
0	1	Crystal mode OSCI or EXT_RC(In dual RC clock mode)
1	0	Internal 4MHz RC
1	1	No function, don't use.

T2EDGE : TMR2 clock edge selection. This bit works only when external clock source TMCKI (IOC5) selected.
= 1, TMR2 increased while external clock H→L (Falling edge).
= 0, TMR2 increased while external clock L→H (Rising edge).

T2PS2:T2PS0 : TMR2 Prescaler selection

T2PS2 : T2PS0			TMR2 Prescal rate
0	0	0	1:1
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

2.1.8.2 TMR2_LA (Timer2 Latch Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Ah	TMR2_LA	8-bit real-time timer/counter Latch							

Note: more bits default state, please refer to Table 2.1.

TMR2_LA is a Timer2 pre-set latch buffer, please don't write FFh to this register, otherwise it will generate an error. See 2.3 for detail description.

2.1.8.3 TMR2_CNT (Timer2 Counter Register)

Read/Write-POR		R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Bh	TMR2_CNT	8-bit real time timer/counter Count							

Note: more bits default state, please refer to Table 2.1.

TMR2_CNT is a Timer2 real-time counter, this register is only read, see 2.3 for detail description.

2.1.9 TMR3: 8-bit Timer & PWM2 Period

The Timer3 is an 8-bit up count timer/counter which includes counter register TMR3_CNT, and latch register TMR3_LA. Please refer to 2.3 for detail Timer description.
The Timer3 can also be combined with Timer4 as PWM2 period and duty and controlled by the register TMR3_CTL2. Please refer to 2.4 for detail PWM description.

2.1.9.1 TMR3_CTL1 (Timer3 Control Register1)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Ch	TMR3_CTL1	T3EN	T3LOAD	T3SO1	T3SO0	T3EDGE	T3PS2	T3PS1	T3PS0

Note: more bits default state, please refer to Table 2.1.

T3EN : TMR3 (PWM2) Enable/Disable
= 1, TMR3 (PWM2) Enable.
= 0, TMR3 (PWM2) Disable.

T3LOAD : Enable/Disable Latch Buffer automatically load to counter register while writing to latch register
= 1, Enable TMR3 latch buffer automatically load to counter register while writing to latch register.
= 0, Disable TMR3 latch buffer automatically load to counter register while writing to latch register.
Note: This bit is only affected after latch register written. When the timer underflows, the latch register data will automatically load into counter register.

T3SO1:T3SO0 : TMR3 clock source selection

T3SO1	T3SO0	TMR3 clock source
0	0	TMCKI(IOC5)
0	1	Crystal mode OSCI or EXT_RC(In dual RC clock mode)
1	0	Internal 4MHz RC
1	1	No function, don't use.

T3EDGE : TMR3 clock edge selection. This bit works only when external clock source TMCKI (IOC5) selected.
= 1, TMR3 increased while external clock H→L (Falling edge).
= 0, TMR3 increased while external clock L→H (Rising edge).

T3PS2:T3PS0 : TMR3 Prescaler selection

T3PS2	T3PS1	T3PS0	TMR3 Prescal rate
0	0	0	1:1
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

2.1.9.2 TMR3_CTL2 (Timer3 Control Register2)

Read/Write-POR		R/W-0	R/W-0	-	-	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Dh	TMR3_CTL2	T34MOD	PWM2_INI	-	-	PWM2R3	PWM2R2	PWM2R1	PWM2R0

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

T34MOD : TMR3 and TMR4 working mode (TMR/ PWM2)
= 1, TMR3 and TMR4 is PWM2.
= 0, TMR3 and TMR4 is Timer.

PWM2_INI : Initial State of PWM2 output duty.
= 1, Set the initial state to L, change to H when TMR4 duty overflow.
= 0, Set the initial state to H, change to L when TMR4 duty overflow.

PWM2R3:PWM2R0 : Interrupt Event Rate of PWM2.
"1:N" means interrupt occurred after "N" PWM2 pulses.

PWM2R3 : PWM2R0				PWM2 Interrupt rate
0	0	0	0	1:1
0	0	0	1	1:2
0	0	1	0	1:3
0	0	1	1	1:4
		|		|
1	1	0	1	1:14
1	1	1	0	1:15
1	1	1	1	1:16

2.1.9.3 TMR3_LA (Timer3 Latch Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Eh	TMR3_LA	8-bit real-time timer/counter Latch							

Note: more bits default state, please refer to Table 2.1.

TMR3_LA is a Timer3 pre-set latch buffer, please don't write FFh to this register, otherwise it will generate an error. See 2.3 for detail description.

2.1.9.4 TMR3_CNT (Timer3 Counter Register)

Read/Write-POR		R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Fh	TMR3_CNT	8-bit real-time timer/counter Count							

Note: more bits default state, please refer to Table 2.1.

TMR3_CNT is a Timer3 real-time counter, this register is only read, see 2.3 for detail description.

2.1.10 TMR4: 8-bit Timer & PWM2 Duty

The Timer4 is an 8-bit up count timer/counter which include latch register TMR4_CNT, and latch register TMR4_LA. Please refer to 2.3 for detail Timer description.

2.1.10.1 TMR4_CTL1 (Timer4 Control Register1)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
20h	TMR4_CTL1	T4EN	T4LOAD	T4SO1	T4SO0	T4EDGE	T4PS2	T4PS1	T4PS0

Note: more bits default state, please refer to Table 2.1.

T4EN : TMR4 Enable/Disable
= 1, TMR4 Enable.
= 0, TMR4 Disable.
Note: At PWM mode, Timer4 is controlled by T3EN.

T4LOAD : Enable/Disable Latch Buffer automatically load to counter register while writing to latch register
= 1, Enable TMR4 latch buffer automatically load to counter register while writing to latch register.
= 0, Disable TMR4 latch buffer automatically load to counter register while writing to latch register.
Note: This bit is only affected after latch register written. When the timer underflows, the latch register data will automatically load into counter register.

T4SO1:T4SO0 : TMR4 clock source selection

T4SO1	T4SO0	TMR4 clock source
0	0	TMCKI(IOC5)
0	1	Crystal mode OSCI or EXT_RC(In dual RC clock mode)
1	0	Internal 4MHz RC
1	1	No function, don't use.

T4EDGE : TMR4 clock edge selection. This bit works only when external clock source TMCKI (IOC5) selected.
= 1, TMR4 increased while external clock H→L (Falling edge).
= 0, TMR4 increased while external clock L→H (Rising edge).

T4PS2:T4PS0 : TMR4 Prescaler selection

T4PS2	T4PS1	T4PS0	TMR4 Prescal rate
0	0	0	1:1
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

2.1.10.2 TMR4_LA (Timer4 Latch Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
21h	TMR4_LA	8-bit real-time timer/counter Latch							

Note: more bits default state, please refer to Table 2.1.

TMR4_LA is a Timer4 pre-set latch buffer, please don't write FFh to this register, otherwise it will generate an error. See 2.3 for detail description.

2.1.10.3 TMR4_CNT (Timer4 Counter Register)

Read/Write-POR		R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
22h	TMR4_CNT	8-bit real-time timer/counter Count							

Note: more bits default state, please refer to Table 2.1.

TMR4_CNT is a Timer4 real-time counter, this register is only read, see 2.3 for detail description.

2.1.11 INTEN (Interrupt Mask Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
25h	INTEN	GIE	ADCIE	PAIE	T4IE	T3_PWM2IE	T2IE	T1_PWM1IE	T0IE

Note: more bits default state, please refer to Table 2.1.

GIE : Global interrupt enable bit.
= 1, Enable all un-masked interrupts.
= 0, Disable all interrupts.
Note : When an interrupt event occurred with the GIE bit and its corresponding interrupt enable bits are set, the GIE bit will be cleared by hardware to disable any further interrupts. The RETFIE instruction will exit the interrupt routine and set the GIE bit to re-enable interrupt.

ADCIE : ADC conversion completed interrupt enable bit.
= 1, Enable interrupt.
= 0, Disable interrupt.

PAIE : PORTA interrupt enable
= 1, Enable interrupt.
= 0, Disable interrupt.

T4IE : Timer4 overflow interrupt enable bit.
= 1, Enable interrupt.
= 0, Disable interrupt.

T3_PWM2IE : Timer3 / PWM2 overflow interrupt enable bit.
= 1, Enable interrupt.
= 0, Disable interrupt.

T2IE : Timer2 overflow interrupt enable bit.
= 1, Enable interrupt.
= 0, Disable interrupt.

T1_PWM1IE : Timer1 / PWM1 overflow interrupt enable bit.
= 1, Enable interrupt.
= 0, Disable interrupt.

T0IE : Timer0 overflow interrupt enable bit.
= 1, Enable interrupt.
= 0, Disable interrupt.

2.1.12 INTFLAG (Interrupt Status Register)

Read/Write-POR		-	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
26h	INTFLAG	-	ADCIF	PAIF	T4IF	T3_PWM2IF	T2IF	T1_PWM1IF	T0IF

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

CAUTION: This register is not recommended BCR instruction.

ADCIF : ADC Interrupt flag. Set when ADC conversion is completed, reset by software.

PAIF : PORTA IOA<7~0> Interrupt flag. Set when pin changed on selected IOA by register INT_PA, and reset by software.

T4IF : TMR4 interrupt flag. Set when TMR4 overflows, and reset by software.

T3_PWM2IF : TMR3 interrupt or PWM2 interrupt flag. Set when TMR3 overflows or PWM2 pulse counts to selected interrupt rate, and reset by software.

T2IF : TMR2 interrupt flag. Set when TMR2 overflows, and reset by software.

T1_PWM1IF : TMR1 interrupt or PWM1 interrupt flag. Set when TMR1 overflows or PWM1 pulse counts to selected interrupt rate, and reset by software.

T0IF : TMR0 interrupt flag. Set when TMR0 overflows, and reset by software.

2.1.13 AD_CTL1 (AD converter Control Register1)

Read/Write-POR		R/W-0	-	R/W-0	-	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
29h	AD_CTL1	ADCEN	-	MODE	-	CHSL3	CHSL2	CHSL1	CHSL0

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

ADCEN : ADC enable/disable setting
= 1, Enable.
= 0, Disable.
Note : This bit should be set by software and would be reset by hardware after the ADC end of conversion.

MODE : ADC operation mode selection
= 1, The ADC is operated in Comparator mode.
= 0, The ADC is operated in Analog to Digital Conversion mode.
Note : When the ADC in comparator mode, the converted data of input voltage would be compared to AD_DAT. The compared result would be stored in the bit7 of AD_CTL2 register.

CHSL3:CHSL0 : ADC input channel select

CHSL3	CHSL2	CHSL1	CHSL0	Input channel
0	0	0	0	Channel 0, IOA0 pin
0	0	0	1	Channel 1, IOA1 pin
0	0	1	0	Channel 2, IOA2 pin
0	0	1	1	Channel 3, IOA3 pin
0	1	0	0	Channel 4, IOA4 pin
0	1	0	1	Channel 5, IOA5 pin
0	1	1	0	Channel 6, IOA6 pin
0	1	1	1	Channel 7, IOA7 pin
1	0	0	0	Channel 8, IOC7 pin
Other				No function, don't use.

2.1.14 AD_CTL2 (AD converter Control Register2)

Read/Write-POR		R-0	-	-	-	-	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
2Ah	AD_CTL2	CMP_D	-	-	-	-	CLKSL2	CLKSL1	CLKSL0

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

CMP_D : Comparison result of ADC in Comparator Mode
= 1, Input Voltage ≧ AD_DAT.
= 0, Input Voltage < AD_DAT.

CLKSL2:CLKSL0 : ADC Conversion clock source select bits.

CKSL2	CKSL1	CKSL0	Conversion clock
0	0	0	System clock /2 (fastest result, lowest quality)
0	0	1	System clock /8
0	1	0	System clock /32
0	1	1	System clock /128 (slowest result, best quality)
1	0	0	System clock /64
1	0	1	System clock /16
1	1	0	System clock /4
1	1	1	No function, don't use

Note : This clock is used to control the conversion precision and speed. The precision will be dropped off if faster conversion rate been used. The lowest conversion rate would be recommended in order to acquire most accurate data.

2.1.15 AD_CTL3 (AD converter Control Register3)

Read/Write-POR		-	-	-	-	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
2Bh	AD_CTL3	-	-	-	-	ANISL3	ANISL2	ANISL1	ANISL0

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

ANISL3:ANISL0 : Analog input select bits.

ANISL3	ANISL2	ANISL1	ANISL0	Analog input selection
0	0	0	0	All the ports are digital input
0	0	0	1	AN0
0	0	1	0	AN1
0	0	1	1	AN2
0	1	0	0	AN3
0	1	0	1	AN4
0	1	1	0	AN5
0	1	1	1	AN6
1	0	0	0	AN7
1	0	0	1	AN8
Other				No function, don't use.

Note : To minimize power consumption, all the I/O pins should be carefully managed before entering sleep mode.

2.1.16 AD_DATL, AD_DATH (AD conversion data high-byte and low-byte Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	-	-	-	-
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
2Ch	AD_DATL	D3	D2	D1	D0	-	-	-	-

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
2Dh	AD_DATH	D11	D10	D9	D8	D7	D6	D5	D4

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

The AD_DAT registers contain the Analog to Digital converted data in the AD conversion mode. When operated in comparator mode, the data written to those registers would be used to compare to the converted data of input voltage.

2.1.17 SYS_CLK (System Clock Control Register)

Read/Write-POR		R/W-0	-	-	-	-	-	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
2Fh	SYS_CLK	CLKS	-	-	-	-	-	IRCPD	ECLKPD

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

The FM8PA76 could be operated either dual or single clock system selected by configuration words. Please refer to 2.13 for detail configuration selection description. This register is used to control the switch between different system clocks and power-down function of those clocks.

CLKS : System Clock Selection (only valid in dual clock mode)
= 1, System Clock is External OSC/RC.
= 0, System Clock is Internal 4MHz RC.

IRCPD : Internal 4MHz RC Power down Control (only valid in dual clock mode)
= 1, Internal 4MHz RC Power Down.
= 0, Internal 4MHz RC Power ON.
Note: Make sure the system clock been switch to external OSC/RC before power down internal RC.

ECLKPD : External clock (OSC/RC) Power down Control (only valid in dual clock mode)
= 1, External OSC/RC Power Down.
= 0, External OSC/RC Power ON.
Note: Make sure the system clock been switch to internal 4MHz RC before power down external OSC/RC.

2.1.18 CLO_CTL (Clock output Control Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	-	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
30h	CLO_CTL	CLO2SO	CLO2PS1	CLO2PS0	-	EXT_CLK	CLO2_E	CLO1_E	TO_E

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

The FM8PA76 provides three kinds of clock output. The first one (CLO1) is the system clock output. The 2nd one (CLO2) is the selected internal or external clock output with prescaler function. The 3rd one (TO) is the TMR3 output with the frequency of TMR3 divided by 2.

CLO2SO : System Clock output 2 source select
= 1, Clock Output 2 source is external OSC/RC.
= 0, Clock Output 2 source is internal 4MHz RC (default).

CLO2PS1:CLO2PS0 : Clock Output 2 prescaler setting

CLO2PS1 : CLO2PS0		Clock Output 2 prescaler
0	0	1:2
0	1	1:4
1	0	1:8
1	1	1:16

EXT_CLK : External clock (IOC5/TMCKI) function selection
= 1, IOC5 is external clock input of timer.
= 0, IOC5 is normal I/O.

CLO2_E : Clock Output 2 (IOC2) function selection
= 1, IOC2 is Clock Output 2.
= 0, IOC2 is normal I/O.

CLO1_E : Clock Output (IOC7) function selection
= 1, IOC7 is System Clock Output.
= 0, IOC7 is normal I/O.

TO_E : TMR3 output (IOC1) Enable/Disable
= 1, IOC1 is the frequency of TMR3 (PWM2) divided by 2.
= 0, IOC1 is normal I/O.

2.1.19 APHCON, BPHCON, CPHCON (Port A, Port B, Port C Pull-high Control Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
31h	APHCON	PHA7	PHA6	PHA5	PHA4	PHA3	PHA2	PHA1	PHA0

Read/Write-POR		-	-	-	-	-	-	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
32h	BPHCON	-	-	-	-	-	-	PHB1	PHB0

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	-
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
33h	CPHCON	PHC7	PHC6	PHC5	PHC4	PHC3	PHC2	PHC1	-

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

Those registers are used to setup pull-high resistor enable/disable of each IO pins.
= 1, Pull-high resistor enable.
= 0, Pull-high resistor disable.

2.1.20 INT_PA (Port A Interrupt / Wakeup control Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
3Ah	INT_PA	PA7IEN	PA6IEN	PA5IEN	PA4IEN	PA3IEN	PA2IEN	PA1IEN	PA0IEN

Note: more bits default state, please refer to Table 2.1.

This register is used to enable/disable the interrupt/wakeup function of PORTA. Please refer to 2.7.1 for detail description of External Interrupt and Wake up function.
PA6IEN:PA0IEN : = 1, Selected IO interrupt/wakeup enable.
= 0, Selected IO interrupt/wakeup disable.

PA7IEN : If WDT_CTL<6> = 1:
This bit state is ignored, IOA7 Pin Interrupt / Wakeup function will be forcibly disabled.
If WDT_CTL<6> = 0:
= 1, Selected IO interrupt/wakeup enable.
= 0, Selected IO interrupt/wakeup disable.

2.1.21 WDT_CTL (Watchdog Timer Control Register)

Read/Write-POR		R/W-1	R/W-0	R/W-0	-	-	R/W-1	R/W-1	R/W-1
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
3Dh	WDT_CTL	WDTEN	I_WDT	I_TWDT	-	-	WDTPS2	WDTPS1	WDTPS0

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

The FM8PA76 builds in a watchdog timer with two different modes, normal watchdog reset and internal watchdog wakeup. The watchdog timer is controlled by this register WDT_CTL. Please refer to 2.5 for detail Watchdog Timer description.

WDTEN : Watchdog Timer Enable/ Disable.
= 1, WDT Enable.
= 0, WDT disable.

I_WDT : Internal Watchdog Wakeup mode selection.
= 1, Internal Watchdog Wakeup Enable.
= 0, Internal Watchdog Wakeup Disable.
Note: If this bit is set, IOA7 Pin Interrupt / Wakeup function will be forcibly disabled.

I_TWDT : Watchdog Timer Stable time required when operating in I_WDT mode.
= 1, 1.25ms.
= 0, 5ms (default).

WDTPS2:WDTPS0 : Watchdog timer prescaler setting

WDTPS2 : WDTPS0			WDT prescaler rate
0	0	0	20mS
0	0	1	40mS
0	1	0	80mS
0	1	1	160mS
1	0	0	320mS
1	0	1	640mS
1	1	0	1.28S
1	1	1	2.56S

2.1.22 TB_BNK (Table Look-up function Bank select Register)

Read/Write-POR		-	-	-	-	-	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
3Eh	TB_BNK	-	-	-	-	-	BNK2	BNK1	BNK0

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.1.

The FM8PA76 provides a table look-up function and the bank selection of ROM data is controlled by this register. Please refer to 2.9 for detail operation of look-up table function.

BNK2:BNK0 : Page selection of Look-up table

BNK2 : BNK0			BANK select
0	0	0	000 XXXX XXXX Table location
0	0	1	001 XXXX XXXX Table location
0	1	0	010 XXXX XXXX Table location
	|		|
1	1	1	111 XXXX XXXX Table location

2.1.23 ACC (Accumulator)

Read/Write-POR		R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
N/A	ACC	Accumulator							

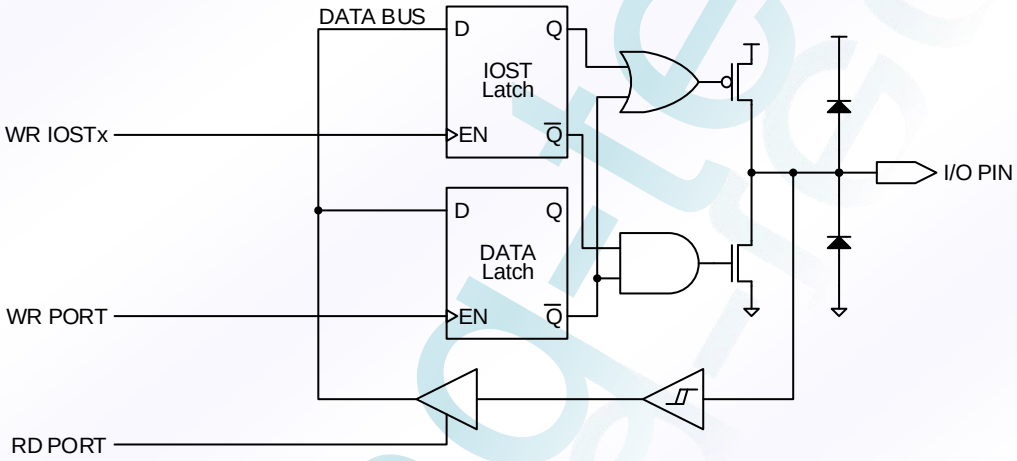
Legend: x = unknown, more bits default state, please refer to Table 2.1.

Accumulator is an internal data transfer, or instruction operand holding. It cannot be addressed.

2.2 I/O Ports

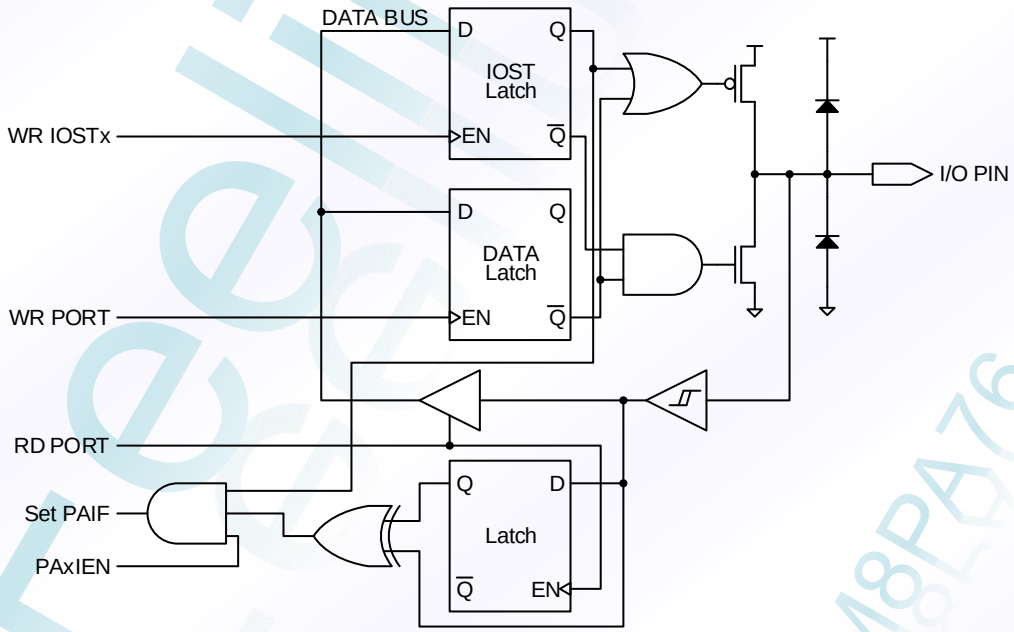
There are totally 17 bi-directional tri-state I/O ports and one (IOB2) input only. All I/O pins (IOA<7:0>, IOB<1:0> and IOC<7:1>) have specified data direction control registers (IOSTA, IOSTB and IOSTC) which can configure these pins as output or input.
All the IO pins can also enable or disable a weak internal pull-high by setting APHCON, BPHCON and CPHCON. This weak pull-high will be automatically turned off when the pin is configured as an output pin.
VR pin is reference voltage input pin of the ADC module, this pin does not have I/O function.
Please note, IOB2 and VR voltage on these pins must not exceed VDD, otherwise it will cause the pin breakdown!!

Figure 2.3: Block Diagram of I/O Pins
IOC7 ~ IOC1, IOB1 and IOB0:



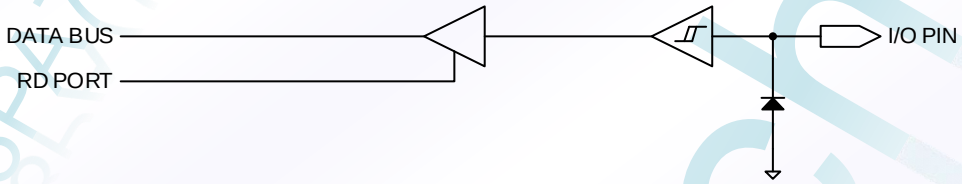
Pull-high control is not shown in this figure

IOA7 ~ IOA0:



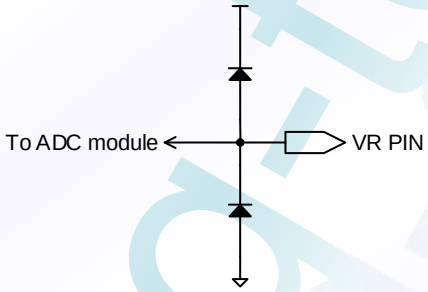
Pull-high/ADC/OSC control is not shown in this figure

IOB2:



Voltage on this pin must not exceed VDD.

VR:

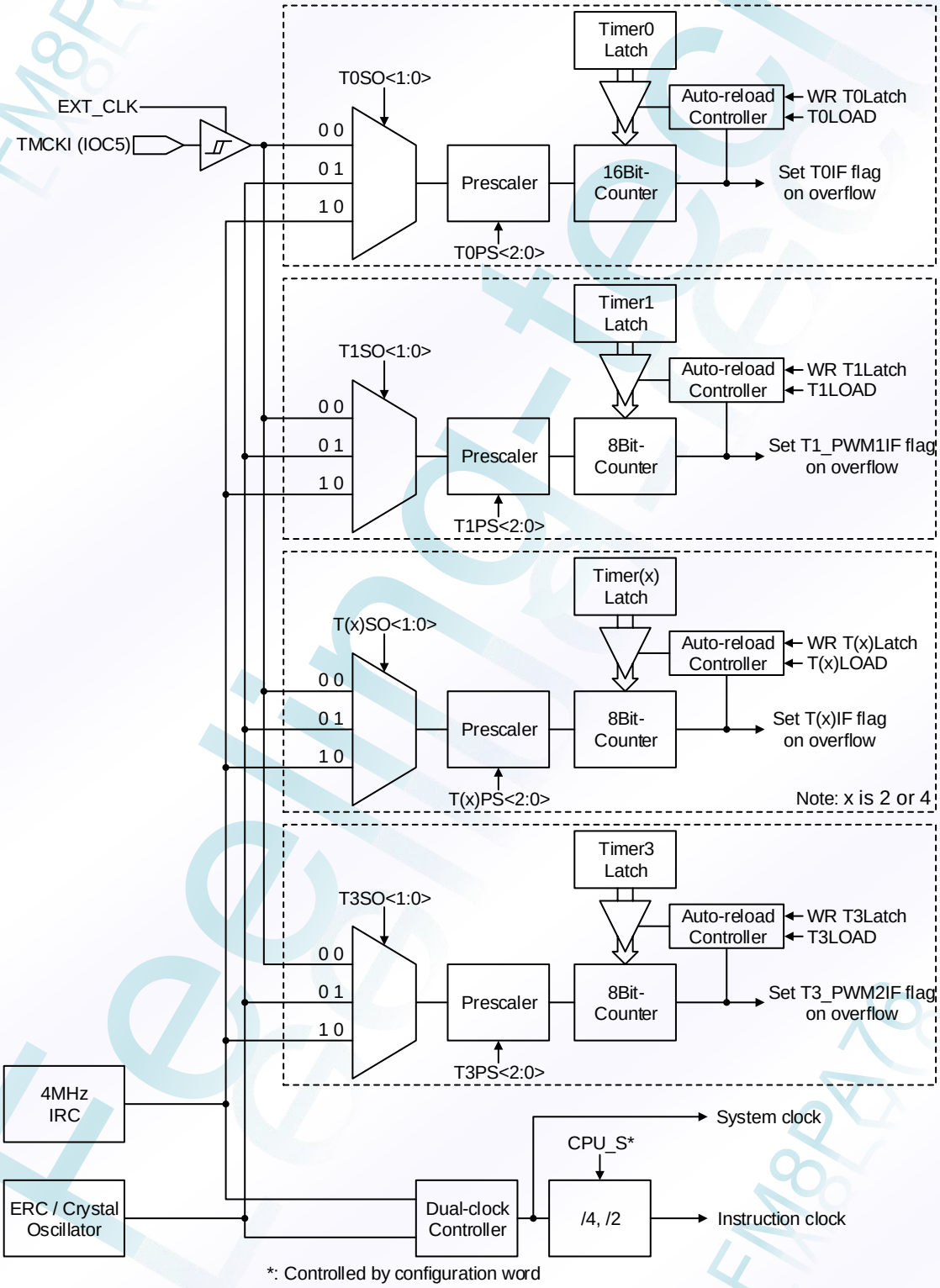


Voltage on this pin must not exceed VDD.

2.3 Timer/Event Counter (TMR0, TMR1, TMR2, TMR3, TMR4)

The FM8PA76 contains one 16-bit up-count, four 8-bit up-counts Timers. All these timers have auto reload function, TMR1/TMR2 and TMR3/TMR4 can be combined to perform PWM function.

Figure 2.4: Simple Block Diagram of the Timer 0 ~ 4



2.3.1 Clock Source

There are 3 clock sources could be selected by each timer separately.

2.3.1.1 TMCKI (IOC5)

The event counter mode would be activated when the source of TMCKI (IOC5) used. At this mode, the rising/ falling edge of the event could also be selected separately.

2.3.1.2 Crystal or External RC Oscillator

In this mode, the timer clock source from Crystal / ERC oscillator module. Oscillator module operating modes are defined by the Fosc bit in the configuration word.

2.3.1.3 Internal 4MHz RC Oscillator

In this mode, timer clock source from internal 4MHz RC oscillator.

2.3.2 Prescaler

Each timer contains a 3-bits prescaler which can scale the timer or counter from 1:1 to 1:128.

TxPS2 : TxPS0			TMRx Prescal rate
0	0	0	1:1
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

2.4 Pulse Width Modulation (PWM)

FM8PA76 provides two PWM output shared with TMR1/2 and TMR3/4. When PWM1 or PWM2 selected, TMR1/TMR3 becomes the period of PWM1/PWM2 and TMR2/TMR4 will be the duty of PWM1/PWM2.
The PWM outputs are on the IOA6/ADC6/INT6/PWM1, and IOA2/ADC2/INT2/PWM2 pins.
PWM1 and PWM2 output has a maximum resolution of 8-bits, the duty cycle of the output can vary from 1% to 99%.
The user needs to set the T12MOD bit (TMR1_CTL2<7>) to enable the PWM1 output, set the T34MOD bit (TMR3_CTL2<7>) to enable the PWM2 output. When T12MOD bit is set, the IOA6/ADC6/INT6/PWM1 pin is configured as PWM1 output and forced as an output, irrespective of the data direct bit (IOSTA<6>). When the T12MOD is clear, the pin behaves as a port pin.
Similarly, the T34MOD bit (TMR3_CTL2<7>) controls the configuration of the IOA2/ADC2/INT2/PWM2 pin.

The PWM1 period time can be calculated as follows:

$$\text{Period time of PWM1} = \frac{[(\text{FFh-T1LA})+1] * \text{TMR1 Prescal rate}}{\text{Clock source frequency}}$$

or

$$\text{T1LA} = 256 - \frac{\text{Period time} * \text{Clock source frequency}}{\text{Prescal rate}}$$

Example:

If the T1LA value is designed to ECh, calculated as follows:

$$\text{Period time of PWM1} = \frac{[(\text{FFh-ECh})+1] * 32(\text{Dec})}{16\text{MHz}} = 40\text{uS (Dec)}$$

or

If the period time is designed to 40uS, calculated as follows:

$$\text{T1LA} = 256 - \frac{40\text{uS} * 16\text{MHz}}{32} = 236(\text{Dec}) = \text{ECh (Hex)}$$

PWM1 duty cycle time is determined by the 8-bit of T2LA, PWM1 duty cycle times are as follows:

$$\text{Duty time of PWM1} = \frac{[(\text{FFh-T2LA})+1] * \text{TMR2 Prescal rate}}{\text{Clock source frequency}}$$

or

$$\text{T2LA} = 256 - \frac{\text{Period time} * \text{Clock source frequency}}{\text{Prescal rate}}$$

Similarly, these formulas can be used directly on PWM2.

Note : **1. When the PWM duty cycle is greater than the PWM period will occur when the wrong result, the user must carefully set.**

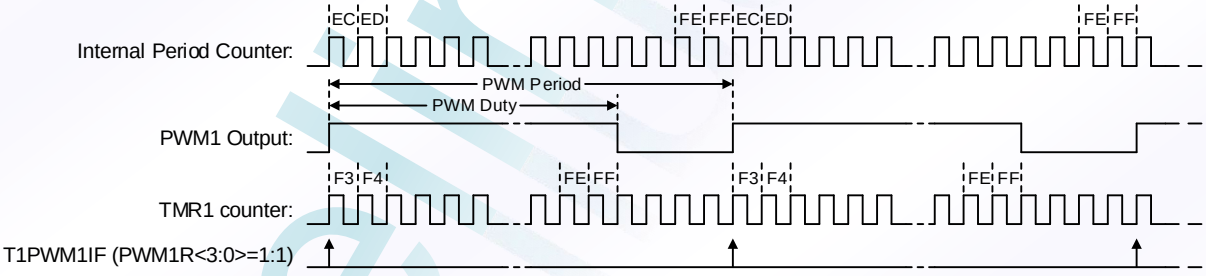
2. When PWM duty or period needed to be changed, the auto-load control bit of the timer (TxLOAD) must be cleared before new data writes to latch register. If this bit still set, the data written to latch register would be load into counter register immediately and cause PWM output anomaly.

Example 2.2: PWM1 Setting (Normal mode)

Address	Code			
NA	#include	<8PA76.ASH>		
		…		
		//Set PWM1 Period		
n		MOVIA	0x15	
n+1		MOVAR	TMR1_CTL1	;CLK source is Crystal, Prescaler 1:32
n+2		MOVIA	0x80	
n+3		MOVAR	TMR1_CTL2	;Set PWM interrupt rate 1:1
n+4		MOVIA	0xEC	
n+5		MOVAR	TMR1_LA	;Set period (0xEC up count to 0x00)
				;Period time = [(0xFF-0xEC)+1]*32*(1/16MHz) = 40uS
		//Set PWM1 Duty		
n+6		MOVIA	0x14	
n+7		MOVAR	TMR2_CTL1	;CLK source is Crystal, Prescaler 1:16
n+8		MOVIA	0xF3	
n+9		MOVAR	TMR2_LA	;Set Duty (0xF3 up count to 0x00)
				;Duty time = [(0xFF-0xF3)+1]*16*(1/16MHz) = 13uS
n+10		BSR	TMR1_CTL1,T1EN_B	;Start PWM1
		//Interrupt setting, not required		
n+11		MOVIA	0x82	
n+12		MOVAR	INTEN	;Enable global & PWM1 interrupt
n+13		MOVIA	0x7D	;Clear interrupt flag
n+14		MOVAR	INTFLAG	;Clear T1_PWM1IF(PWM1) flag

Note: 1. The PWM duty (Timer2) must be smaller than PWM period (Timer1).
2. This example demonstrates the PWM applied in Crystal mode. In this example, the frequency of external OSC is approximately 16MHz.

Figure 2.5 PWM Output Waveform



2.5 Watch Dog Timer (WDT)

The Watchdog Timer (WDT) is a free running on-chip RC oscillator which does not require any external components. So the WDT will still run even if the clock on the OSCI and OSCO pins is turned off, such as in SLEEP mode.
The WDT can be disabled by clearing the control bit WDTEN (WDT_CTL<7>) to "0".

The WDT has a typical time-out period of 20 mS (without prescaler). This period of this timer may be variant slightly because of temperature, voltage, and process variation. If a longer time-out period is desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT controlled by the WDT_CTL register <2:0>. Thus, the longest time-out period is approximately 2.56 seconds.
The CLRWDT instruction clears the WDT and prevents it from timing out and generating a device reset.
The SLEEP instruction also resets the WDT. This gives the maximum SLEEP time before a WDT Wake-up Reset.

There are two type of watchdog timer mode could be selected by I_WDT (WDT_CTL<6>). When I_WDT bit disable, normal watchdog timer reset is selected. During normal operation or in SLEEP mode, a WDT time-out will cause the device reset and the $\overline{TO}$ bit (STATUS<4>) will be cleared.
If I_WDT bit enabled, the internal watchdog timer wakeup will be used. The system wakeups from sleep, then jumps into interrupt vector with external interrupt request PAIF (INTFLAG<5>) and continues from next instruction instead of triggering a reset event. There is a stabilization time required for internal watchdog wakeup could be selected by I_ TWDT (WDT_CTL<5>). The default value of this stabilization timer is 5ms.

Example 2.3: Internal Watchdog Wakeup

Address	Code			
NA	#include	<8PA76.ASH>		
0x003		…		
0x004		… (Backup status code)		1. WDT Wakeup
		…		;User WDT Wakeup ISR code
		MOVIA	0xDF	
		MOVAR	INTFLAG	;Clear PAIF flag(Note1)
		… (Restore status code)		
		RETFIE		
		2. Return from ISR		
n		MOVIA	0xA0	
n+1		MOVAR	INTEN	;Enable global & Port A interrupt
n+2		CLRWDT		
n+3		MOVIA	0xE7	
n+4		MOVAR	WDT_CTL	;Sleep: 2.56S + Wakeup:5mS
n+5		…		
n+6		…		
n+7		SLEEP		
n+8		NOP		
		…		

Note : 1. BCR instruction is not recommended for Clear interrupt flag (INTFLAG register).
2. Interrupt backup / restore status code are not shown in this example.

Example 2.4: Typical Watchdog Reset

Address	Code			
NA	#include	<8PA76.ASH>		
0x000		…		
		…		WDT Reset
n		CLRWDT		
n+1		MOVIA	0x87	
n+2		MOVAR	WDT_CTL	;Sleep: 2.56S + Wakeup:20mS
n+3		…		
n+4		…		
n+5		SLEEP		
n+6		NOP		
n+7		…		
n+8		…		

2.6 Reset

FM8PA76 device may be RESET in one of the following ways:

1. Power-on Reset (POR)
2. Brown-out Reset (BOR)
3. RSTB Pin Reset
4. WDT time-out Reset

Some registers are not affected in any RESET condition. Their status is unknown on Power-on Reset and unchanged in any other RESET. Most other registers are reset to a “reset state” on Power-on Reset, RSTB or WDT Reset.

A Power-on RESET pulse is generated on-chip when Vdd rise is detected. To use this feature, the user merely ties the RSTB pin to Vdd.

On-chip Low Voltage Detector (LVDT) places the device into reset when Vdd is below a fixed voltage. This ensures that the device does not continue program execution outside the valid operation Vdd range. Brown-out RESET is typically used in AC line or heavy loads switched applications.

A RSTB or WDT Wake-up from SLEEP also results in a device RESET, and not a continuation of operation before SLEEP.

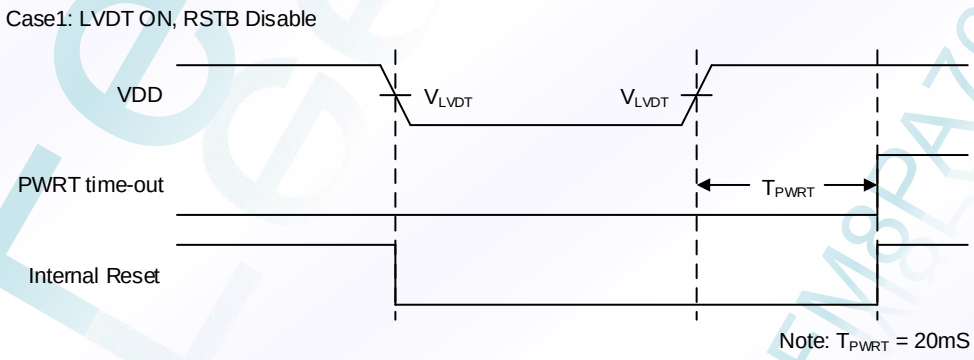
The $\overline{TO}$ and $\overline{PD}$ bits (STATUS<4:3>) are set or cleared depending on the different reset conditions.

2.6.1 Power-up Reset Timer (PWRT)

The Power-up Reset Timer provides a nominal 20ms delay after Power-on Reset (POR), Brown-out Reset (BOR), RSTB Reset or WDT time-out Reset. The device is kept in reset state as long as the PWRT is active.

The PWDT delay will vary from device to device due to Vdd, temperature, and process variation.

Figure 2.6: Reset Timing



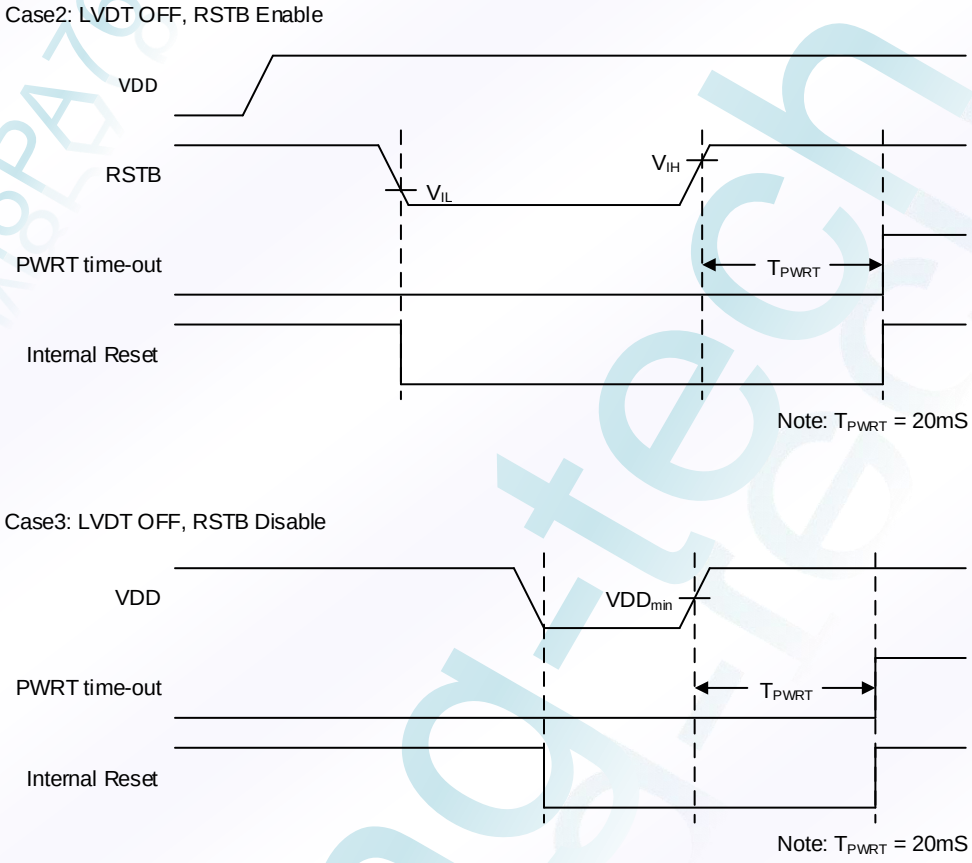


Figure 2.7: Simplified Block Diagram of on-chip Reset Circuit

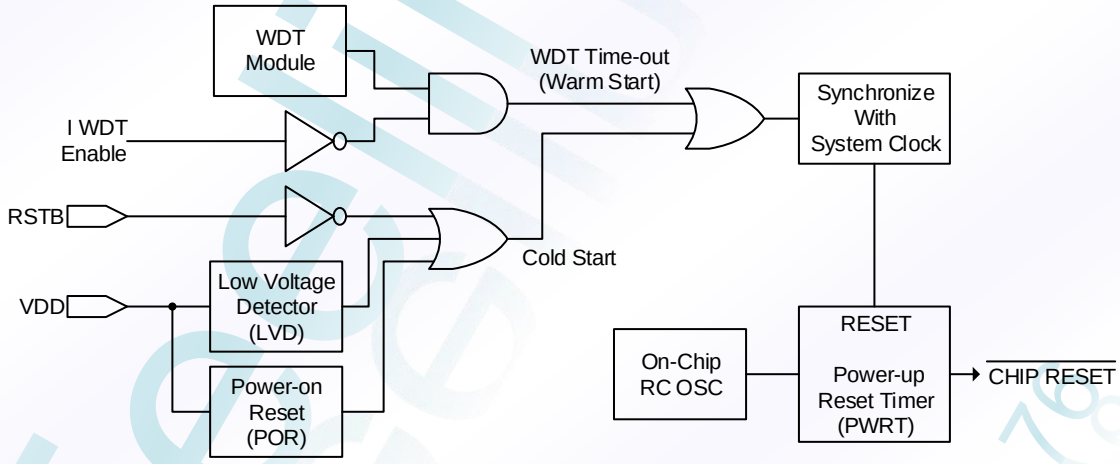


Table 2.1: Reset Conditions for Operational Registers

Register	Address	Power-on Reset Brown-out Reset	WDT Reset RSTB Reset
ACC	N/A	xxxx xxxx	uuuu uuuu
INDF	00h	xxxx xxxx	uuuu uuuu
PCL	01h	0000 0000	0000 0000
PCHBUF	02h	---- -000	---- -000
STATUS	03h	---1 1xxx	---# #xxx
FSR	04h	xxxx xxxx	uuuu uuuu
IOSTA	05h	1111 1111	1111 1111
PORTA	06h	xxxx xxxx	uuuu uuuu
IOSTB	07h	---- --11	---- --11
PORTB	08h	---- -xxx	---- -xuu
IOSTC	09h	1111 111-	1111 111-
PORTC	0Ah	xxxx xxx-	uuuu uuu-
TMR0_CTL	10h	0000 0000	0000 0000
TMR0L_LA	11h	0000 0000	0000 0000
TMR0H_LA	12h	0000 0000	0000 0000
TMR0L_CNT	13h	0000 0000	0000 0000
TMR0H_CNT	14h	0000 0000	0000 0000
TMR1_CTL1	15h	0000 0000	0000 0000
TMR1_CTL2	16h	00-- 0000	00-- 0000
TMR1_LA	17h	0000 0000	0000 0000
TMR1_CNT	18h	0000 0000	0000 0000
TMR2_CTL1	19h	0000 0000	0000 0000
TMR2_LA	1Ah	0000 0000	0000 0000
TMR2_CNT	1Bh	0000 0000	0000 0000
TMR3_CTL1	1Ch	0000 0000	0000 0000
TMR3_CTL2	1Dh	00-- 0000	00-- 0000
TMR3_LA	1Eh	0000 0000	0000 0000
TMR3_CNT	1Fh	0000 0000	0000 0000
TMR4_CTL1	20h	0000 0000	0000 0000
TMR4_LA	21h	0000 0000	0000 0000
TMR4_CNT	22h	0000 0000	0000 0000
INTEN	25h	0000 0000	0000 0000
INTFLAG	26h	-000 0000	-000 0000
AD_CTL1	29h	0-0- 0000	0-0- 0000
AD_CTL2	2Ah	0--- -000	0--- -000
AD_CTL3	2Bh	---- 0000	---- 0000
AD_DATL	2Ch	0000 ----	0000 ----
AD_DATH	2Dh	0000 0000	0000 0000
SYS_CLK	2Fh	0--- --00	0--- --00
CLO_CTL	30h	000- 0000	000- 0000
APHCON	31h	0000 0000	0000 0000
BPHCON	32h	---- --00	---- --00

Register	Address	Power-on Reset Brown-out Reset	WDT Reset RSTB Reset
CPHCON	33h	0000 000-	0000 000-
INT_PA	3Ah	0000 0000	0000 0000
WDT_CTL	3Dh	100- -111	100- -111
TB_BNK	3Eh	---- -000	---- -000
General Purpose Registers	40 ~ BFh	xxxx xxxx	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented, # = refer to the following table for possible values.

Table 2.2: $\overline{TO}$ and $\overline{PD}$ Status after Reset

$\overline{TO}$	$\overline{PD}$	RESET was caused by
0	0	WDT timer overflow from sleep mode
0	1	WDT timer overflow from normal mode
1	0	Set ‘low” at RSTB from sleep mode
1	1	Power on reset
u	u	Set “low” at RSTB from normal mode

Legend: u = unchanged.

Table 2.3: $\overline{TO}$ and $\overline{PD}$ Status after Reset

Event	$\overline{TO}$	$\overline{PD}$
Power-on	1	1
WDT Time-out	0	u
SLEEP instruction	1	0
CLRWDT instruction	1	1

Legend: u = unchanged.

2.7 Interrupt

The FM8PA76 has three kinds of interrupt sources:
1. 8 External IOA<7:0> pin changed interrupt
2. 5 Timers / Counters overflow interrupt (or PWM interrupt)
3. ADC conversion completion interrupt

INTFLAG is the interrupt flag register that recodes the interrupt requests to the relative flags.
A global interrupt enable bit, GIE (INTEN<7>), enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. Individual interrupts can be enabled/ disabled through their corresponding enable bits in INTEN register regardless of the status of the GIE bit.
When an interrupt event occur with the GIE bit and its corresponding interrupt enable bit are all set, the GIE bit will be cleared by hardware to disable any further interrupts, and the next instruction will be fetched from address 004h. The interrupt flag bits must be cleared by software before re-enabling GIE bit to avoid recursive interrupts.
The RETFIE instruction exits the interrupt routine and set the GIE bit to re-enable interrupt.
The RETF instruction exits the interrupt routine and does NOT set the GIE bit.
The flag bit in INTFLAG register is set by interrupt event regardless of the status of its mask bit.

2.7.1 PORTA<7:0> External Interrupt and Wakeup Function

The external interrupt on PORTA<7:0> are selected by INT_PA<7:0> and PAIE (INTEN<5>). When the device is in normal mode and the specified IO status changed, the interrupt event will be triggered and the program will jump to 004h.
When the device is in sleep mode, those interrupts can also be used as an external wakeup signal. The device will restart system clock and the program will jump to 004h after startup timer timeout.
Please note, if I_WDT(WDT_CTL<6>) is set, IOA7 Pin Interrupt / Wakeup function will be forcibly disabled.

Example 2.5: External IOA0 pin change interrupt

Address	Code		
NA	#include	<8PA76.ASH>	
0x003	…		1. IOA0 pin change
0x004		… (Backup status code)	
		…	;User Port A pin change ISR code
		MOVIA 0xDF	
		MOVAR INTFLAG	;Clear PAIF flag(Note1)
		… (Restore status code)	
		RETFIE	
	2. Return from ISR		
n		MOVIA 0xFF	
n+1		MOVAR IOSTA	;Set Port A as input
n+2		MOVIA 0xA0	
n+3		MOVAR INTEN	;Enable global & Port A interrupt
n+4		MOVIA 0xDF	
n+5		MOVAR INTFALG	;Clear PAIF flag(Note1)
n+6		MOVR PORTA,R	;Update Port A pin status
n+7		MOVIA 0x01	
n+8		MOVAR INT_PA	;Set IOA0 pin change
		…	
		…	
		…	

Note : 1. BCR instruction is not recommended for Clear interrupt flag (INTFLAG register).
2. Interrupt backup / restore status code are not shown in this example.

Example 2.6: External IOA0 pin change wakeup interrupt

Address	Code		
NA	#include	<8PA76.ASH>	
0x003		…	
0x004		… (Backup status code)	1. IOA0 pin change
		…	; User Port A pin change wakeup ISR code
		MOVIA 0xDF	
		MOVAR INTFLAG	;Clear PAIF flag(Note1)
		… (Restore status code)	
		RETFIE	
	2. Return from ISR		
n		MOVIA 0xFF	
n+1		MOVAR IOSTA	;Set Port A as input
n+2		MOVIA 0xA0	
n+3		MOVAR INTEN	;Enable global & Port A interrupt
n+4		MOVIA 0xDF	
n+5		MOVAR INTFALG	;Clear PAIF flag(Note1)
n+6		MOVR PORTA,R	;Update Port A pin status
n+7		MOVIA 0x01	
n+8		MOVAR INT_PA	;Set IOA0 pin change wakeup
n+9		SLEEP	
n+10		NOP	
		…	

Note : 1. BCR instruction is not recommended for Clear interrupt flag (INTFLAG register).
2. Interrupt backup / restore status code is not shown in this example.

2.7.2 Timer0~4 Interrupt's

2.7.2.1 Timer0 interrupt

An overflow (FFFFh → 0000h) in the TMR0 counter will set the flag bit T0IF (INTFLAG<0>). This interrupt can be disabled by clearing T0IE bit (INTEN<0>).

2.7.2.2 Timer 1 interrupt

At Timer mode, an overflow (FFh → 00h) in the TMR1 counter will set the flag bit T1_PWM1IF (INTFLAG<1>). This interrupt can be disabled by clearing T1_PWM1IE bit (INTEN<1>).
At PWM mode, the end of each PWM period cycle to generate an interrupt. The interrupt rate can be adjusted by TMR1_CTL2<3:0>. See Figure 2.8 for detail description.

2.7.2.3 Timer 2 interrupt

At Timer mode, an overflow (FFh → 00h) in the TMR2 counter will set the flag bit T2IF (INTFLAG<2>). This interrupt can be disabled by clearing T2IE bit (INTEN<2>).
At PWM mode, TMR2 is PWM1 duty cycle counter. Not generate an interrupt.

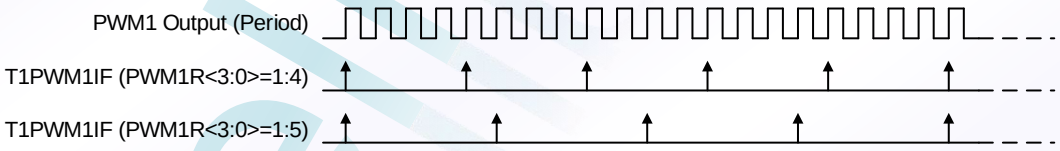
2.7.2.4 Timer 3 interrupt

At Timer mode, an overflow (FFh → 00h) in the TMR3 counter will set the flag bit T3_PWM2IF (INTFLAG<3>). This interrupt can be disabled by clearing T3_PWM2IE bit (INTEN<3>).
At PWM mode, the end of each PWM period cycle to generate an interrupt. The interrupt rate can be adjusted by TMR3_CTL2<3:0>. See Figure 2.8 for detail description.

2.7.2.5 Timer 4 interrupt

At Timer mode, an overflow (FFh → 00h) in the TMR4 counter will set the flag bit T4IF (INTFLAG<4>). This interrupt can be disabled by clearing T4IE bit (INTEN<4>).
At PWM mode, TMR4 is PWM2 duty cycle counter. Not generate an interrupt.

Figure 2.8: PWM Interrupt Waveform



2.7.3 ADC conversion completion interrupt

When the A/D conversion is completed, the flag bit ADCIF (INTFLAG<6>) will be set. And the ADCIF bit can be cleared by software.
This interrupt can be disabled by clearing ADCIE bit (INTEN<6>).

2.8 Analog to Digital Converter (ADC)

This analog to digital converter has 9 channels 12bits (10+2) resolution. The ADC is controlled by three control register, AD_CTL1, AD_CTL2, and AD_CTL3. The FM8PA76 provides two operation modes, AD conversion mode and comparator mode. The operation mode can be selected by MODE (AD_CTL1<5>).
In AD conversion mode, the AD_DATL and AD_DATH register shows the AD conversion result. If the comparator selected, the data written to those two registers will be compared to the converted data of input voltage. The result will be shown in CMP_D (AD_CTL2<7>).

Example 2.7: Analog to Digital Conversion (Channel0 AD conversion)

Address	Code			
NA	#include	<8PA76.ASH>		
	…			
n		BTRSC	AD_CTL1,ADCEN_B	
n+1		LGOTO	$-1	; Make Sure no ADC is processing
n+2		MOVIA	0xBF	
n+3		MOVAR	INTFLAG	; Clear ADCIF flag(Note)
n+4		MOVIA	0x00	
n+5		MOVAR	AD_CTL1	; Select ADC Channel 0 (IOA0) conversion
n+6		MOVIA	0x03	
n+7		MOVAR	AD_CTL2	; Set AD conversion rate: System clock / 128
n+8		MOVIA	0x01	
n+9		MOVAR	AD_CTL3	; Set AN0 analog input
n+10		BSR	AD_CTL1,ADCEN_B	; ADC conversion start
n+11		BTRSS	INTFLAG,ADCIF_B	
n+12		LGOTO	$-1	; Wait AD end of conversion
n+13		MOVR	AD_DATH,A	; Read ADC high byte data
n+14		MOVAR	…	; Transfer ADC value to other register.
n+15		MOVR	AD_DATL,A	; Read ADC low byte data
n+16		MOVAR	…	; Transfer ADC value to other register.
		…		

Note : BCR instruction is not recommended for Clear interrupt flag (INTFLAG register).

2.9 Look-Up Table Function

The Look-up Table function is built-in to access the data table within entire ROM area. The TB_BNK register is used to address the high byte of the location of required ROM. The instructions TABL and TABH are used to read low byte and high byte of the addressed ROM. The result of instructions will be stored at ACC register. Please refer to the following example for detail.

Example 2.8: Look-up Table

Address	Code			
NA	#include	<8PA76.ASH>		
		…		
n		MOVIA	0x03	
n+1		MOVAR	0x5B	;Save offset value 03H to register 0x5B (low bit
				; address)
n+2		MOVIA	0x07	
n+3		MOVAR	TB_BNK	; Save offset value 07H to TB_BNK (high bit
				; address)
n+4		TABL	0x5B	; Read Low byte 0x703 ROM Data, and saved
				; it to ACC. (ACC=0xAA)
n+5		MOVAR	….	; Transfer value to other register.
n+6		TABH	0x5B	; Read High byte 0x703 ROM Data, and saved
				; it to ACC. (ACC=0x55)
n+7		MOVAR	…	; Transfer value to other register.
n+8		…		
0x700		DW	0x1122	
0x701		DW	0x3344	
0x702		DW	0x5566	
0x703		DW	0x55AA	
…		…		

2.10 Hexadecimal Convert to Decimal (HCD)

Decimal format is another number format for FM8PA76. When the content of the data memory has been assigned as decimal format, it is necessary to convert the results to decimal format after the execution of ALU instructions. When the decimal converting operation is processing, all of the operand data (including the contents of the data memory (RAM), accumulator (ACC), immediate data, and look-up table) should be in the decimal format, or the results of conversion will be incorrect.
Instruction DAA can convert the ACC data from hexadecimal to decimal format after any addition operation and restored to ACC.
The conversion operation is illustrated in Example 2.9.

Example 2.9: DAA CONVERSION

Address	Code			
NA	#include	<8PA76.ASH>		
n		…		
n+1		MOVIA	0x90	;Set immediate data = decimal format number “90” (ACC ← 90h)
n+2		MOVAR	0x40	;Load immediate data “90” to data memory address 40H
n+3		MOVIA	0x10	;Set immediate data = decimal format number “10” (ACC ← 10h)
n+4		ADDAR	0x40,A	;Contents of the data memory address 40H and ACC are binary-added
				;the result loads to the ACC (ACC ← A0h, C ← 0)
n+5		DAA	0x40,A	;Convert the content of ACC to decimal format, and restored to ACC
				;The result in the ACC is “00” and the carry bit C is “1”. This represents the
				;decimal number “100”
n+6	…			

Instruction DAS can convert the ACC data from hexadecimal to decimal format after any subtraction operation and restored to ACC.
The conversion operation is illustrated in Example 2.10.

Example 2.10: DAS CONVERSION

Address	Code			
NA	#include	<8PA76.ASH>		
n		…		
n+1		MOVIA	0x10	;Set immediate data = decimal format number “10” (ACC ← 10h)
n+2		MOVAR	0x40	;Load immediate data “90” to data memory address 40H
n+3		MOVIA	0x20	;Set immediate data = decimal format number “20” (ACC ← 20h)
n+4		SUBAR	0x40,A	;Contents of the data memory address 40H and ACC are binary-subtracted
				;the result loads to the ACC (ACC ← F0h, C ← 0)
n+5		DAS	0x40,A	;Convert the content of ACC to decimal format, and restored to ACC
				;The result in the ACC is “90” and the carry bit C is “0”. This represents the
				;decimal number “ -10”
n+6		…		

2.11 Dual Clock Function

The chip can be operated in four different dual clock function, users need to use it, and the configuration word must be set to one of following:

- HF & IRC
- XT & IRC
- LF & IRC
- ERC & IRC

If not in these states, will not be able to use dual clock function. By default, the system is the use of internal IRC frequency as the clock source, and the two oscillator circuit is in the enable state. If not used, turn off unused oscillator power (via SYS_CLK), can be reduce unnecessary current consumption.
When you want to switch clock source, recommend follow these steps:

1. Turn-on another oscillator power.
2. Wait oscillator to stable (HF, XT and LF mode requires this step).
3. Set WDT prescaler to 1:128 and Clear Watch-dog (avoid watchdog overflow).
4. Set or Clear CLKS bit (SYS_CLK<7>) to switch to another clock source.
5. Wait two NOP instruction (Required sequence).
6. Clear Watch-dog and set back to original settings.
7. If original oscillator not used, turn-off it.

Since the oscillator from the off state to the normal output clock oscillator needs some time to wait for a stable, at each oscillation mode, we recommend waiting time should be greater than the following table:

Table 2.4: Recommend typical wait time

Situation	Typical waiting time
Crystal → IRC	10uS
ERC → IRC	10uS
IRC → Crystal (XT or HF, 4 to 20 MHz)	5mS
IRC → Crystal (LF, 32 KHz)	2S
IRC → ERC	1.5mS

Note: 1. This table is for reference only.
2. Quartz crystal characteristics vary according to type, package and manufacturer, the users must be carefully tested and verified.
3. RC oscillator mode will change depending on the operating voltage, the user must carefully tested and verified.

Example 2.11: Switching from IRC to External clock

Address	Code			
NA	#include	<8PA76.ASH>		
		…		
n		BCR	SYS_CLK,ECLKPD_B	;Turn-on External oscillator
n+1		LCALL	Delay	;Wait Crystal oscillator to stable
		MOVIA	0x87	
		MOVAR	WDT_CTL	;If Watch-dog enable, recommend set to 1:128
n+2		CLRWDT		; If Watch-dog enable, clean it!
n+3		BSR	SYS_CLK,CLKS_B	;Switching from IRC to External clock
n+4		NOP		Required sequence
n+5		NOP		
n+6		CLRWDT		; If Watch-dog enable, clean it!
n+7		BSR	SYS_CLK,IRCPD_B	;Turn-off IRC oscillator (if unused)
n+8		MOVIA	0xnn	
n+9		MOVAR	WDT_CTL	;Set back original settings (if Watch-dog used)
		…		

Similarly, switching from External clock to IRC also this procedure.

2.12 Oscillator Configurations

FM8PA76 can be operated in eight different combinations of oscillator modes. Users can program configuration word (F_{OSC}) to select the appropriate modes. The eight different system clock modes are combination of the following oscillators:

- LF: Low Frequency Crystal Oscillator
- XT: Crystal/Resonator Oscillator
- HF: High Frequency Crystal/Resonator Oscillator
- ERC: External Resistor/Voltage Controlled Oscillator
- IRC: Internal Resistor/Capacitor Oscillator

In LF, XT, or HF modes, a crystal or ceramic resonator in connected to the OSCI and OSCO pins to establish oscillation. When in LF, XT, or HF modes, the devices can have an external clock source drive the OSCI pin. The ERC device option offers additional cost savings for timing insensitive applications. The RC oscillator frequency is a function of the supply voltage, the resistor (Rext) and capacitor (Cext), the operating temperature, and the process parameter.
The IRC option offers largest cost savings for timing insensitive applications.

Figure 2.9: HF, XT or LF Oscillator Modes (Crystal Operation or Ceramic Resonator)

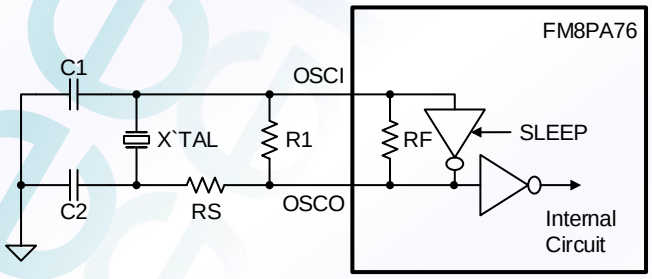


Figure 2.10: HF, XT or LF Oscillator Modes (External Clock Input Operation)

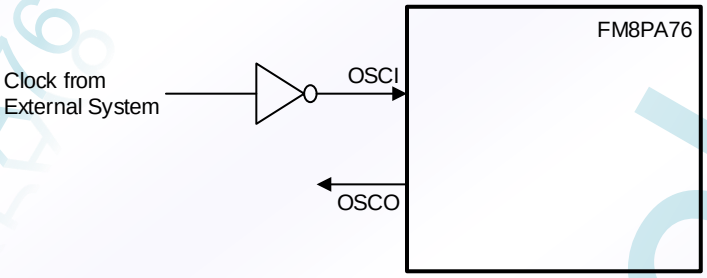
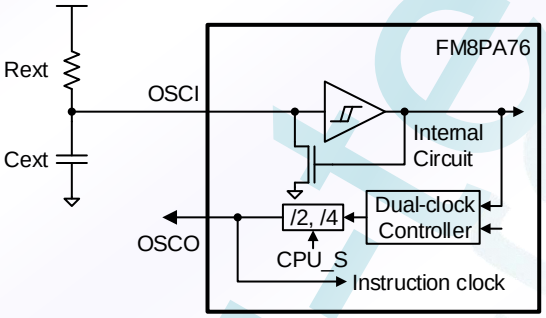


Figure 2.11: ERC Oscillator Mode (External RC Oscillator)

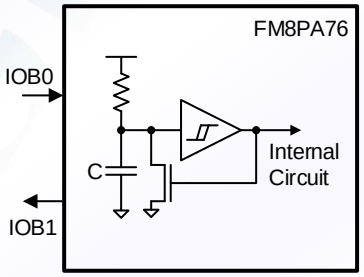


The typical oscillator frequency vs. external resistor is as following table
When Cext = 0.01uf (103)

5V		3V	
Rext	Frequency	Rext	Frequency
4.3M	32 KHz	3.6M	32 KHz
220K	500 KHz	189K	500 KHz
107K	1.0 MHz	100K	1.0 MHz
56K	2.0 MHz	53K	2.0 MHz
30K	4.0 MHz	28K	4.0 MHz
16K	8.0 MHz	15K	8.0 MHz
11K	12.0 MHz	11K	12.0 MHz

Note: Values are provided for design reference only.

Figure 2.12: IRC Oscillator Mode (Internal R, Internal C Oscillator)



2.13 Configuration Words

Table 2.5: Configuration Words

Name	Description
Fosc	Oscillator Selection Bit → IRC (4MHz) mode (default) → HF crystal & IRC(4MHz) mode → XT crystal & IRC(4MHz) mode → LF crystal & IRC(4MHz) mode → ERC & IRC(4MHz) mode → HF crystal mode → XT crystal mode → LF crystal mode
WDTEN	Watchdog Timer Enable Bit → WDT enabled (default) → WDT disabled
LVDT	Low Voltage Detector Selection Bit → Disable* → LVDT = 2.2V* → LVDT = 2.6V (default) → LVDT = 3.7V
RSTBIN	IOB2/RSTB Pin Selection Bit → RSTB pin is selected (default) → IOB2 pin is selected
CPU_S	Instruction Period Selection Bit → four oscillator periods (4T) (default) → two oscillator periods (2T)
PROTECT	Code Protection Bit → NO, OTP code protection off (default) → YES, OTP code protection on

Note: LVDT not recommended select to Disable or 2.2V. If used, the user must carefully tested and verified.

Table 2.6: Selection of IOB0/OSCI and IOB1/OSCO Pin

Mode of oscillation	IOB0/OSCI	IOB1/OSCO
IRC	Force to IOB0	Force to IOB1
ERC	Force to OSCI	Force to OSCO
HF, XT, LF	Force to OSCI	Force to OSCO

3.0 INSTRUCTION SET

Mnemonic, Operands		Description	Operation	Cycles	Status Affected
BCR	**R, bit**	Clear bit in R	0 → R<b>	1	-
BSR	**R, bit**	Set bit in R	1 → R<b>	1	-
BTRSC	**R, bit**	Test bit in R, Skip if Clear	Skip if R<b> = 0	1/2(1)	-
BTRSS	**R, bit**	Test bit in R, Skip if Set	Skip if R<b> = 1	1/2(1)	-
NOP		No Operation	No operation	1	-
CLRWDT		Clear Watchdog Timer	00h → WDT, 00h → WDT prescaler	1	$\overline{TO}$,$\overline{PD}$
SLEEP		Go into power-down mode	00h → WDT, 00h → WDT prescaler	1	$\overline{TO}$,$\overline{PD}$
TABL	**R**	Read low byte ROM table to (acc) ROM table address={TB_BNK, index of R}	ACC=ROM{BANK index: R}[7:0]	2	-
TABH	**R**	Read high byte ROM table to (acc) ROM table address={TB_BNK, index of R}	ACC=ROM{BANK index : R}[15:8]	2	-
DAA	**R, d**	Adjust data format of register from HEX to DEC after any addition operation	R(hex) → dest (dec)	1	C
DAS	**R, d**	Adjust data format of register from HEX to DEC after any subtraction operation	R(hex) → dest (dec)	1	C
RETURN		Return from subroutine	Top of Stack → PC	2	-
RETFIE		Return from interrupt, set GIE bit	Top of Stack → PC, 1 → GIE	2	-
RETF		Return from interrupt	Top of Stack → PC,	2	-
CLRA		Clear ACC	00h → ACC	1	Z
CLRR	**R**	Clear R	00h → R	1	Z
MOVAR	**R**	Move ACC to R	ACC → R	1	-
MOVR	**R, d**	Move R	R → dest	1	Z
MOV2	**R, d**	Move R	R → dest	1	-
DECR	**R, d**	Decrement R	R - 1 → dest	1	Z
DECRSZ	**R, d**	Decrement R, Skip if 0	R - 1 → dest, Skip if result = 0	1/2(1)	-
INCR	**R, d**	Increment R	R + 1 → dest	1	Z
INCRSZ	**R, d**	Increment R, Skip if 0	R + 1 → dest, Skip if result = 0	1/2(1)	-
ADDAR	**R, d**	Add ACC and R	R + ACC → dest	1	C, DC, Z
SUBAR	**R, d**	Subtract ACC from R	R - ACC → dest	1	C, DC, Z
ADCAR	**R, d**	Add ACC and R with Carry	R + ACC + C → dest	1	C, DC, Z
SBCAR	**R, d**	Subtract ACC from R with Carry	R + $\overline{ACC}$ + C → dest	1	C, DC, Z
ANDAR	**R, d**	AND ACC with R	ACC and R → dest	1	Z
IORAR	**R, d**	Inclusive OR ACC with R	ACC or R → dest	1	Z
XORAR	**R, d**	Exclusive OR ACC with R	R xor ACC → dest	1	Z
COMR	**R, d**	Complement R	$\overline{R}$→ dest	1	Z
RL	**R, d**	Rotate left R	R<6:0> → dest<7:1>, R<7>→ dest<0>	1	-
RLR	**R, d**	Rotate left R through Carry	R<7> → C, R<6:0> → dest<7:1>, C → dest<0>	1	C
RL0	**R, d**	Rotate left R through 0	R<6:0> → dest<7:1>, 0 → dest<0>	1	-

Mnemonic, Operands		Description	Operation	Cycles	Status Affected
RL1	**R, d**	Rotate left R through 1	R<6:0> → dest<7:1>, 1 → dest<0>	1	-
RR	**R, d**	Rotate right R	R<7:1> → dest<6:0>, R<0> → dest<7>	1	-
RRR	**R, d**	Rotate right R through Carry	C → dest<7>, R<7:1> → dest<6:0>, R<0> → C	1	C
RR0	**R, d**	Rotate right R with 0	0 → dest<7>, R<7:1> → dest<6:0>,	1	-
RR1	**R, d**	Rotate right R with 1	1 → dest<7>, R<7:1> → dest<6:0>,	1	-
SWAPR	**R, d**	Swap R	R<3:0> → dest<7:4>, R<7:4> → dest<3:0>	1	-
MOVIA	**I**	Move Immediate to ACC	I → ACC	1	-
ADDIA	**I**	Add ACC and Immediate	I + ACC → ACC	1	C, DC, Z
SUBIA	**I**	Subtract ACC from Immediate	I - ACC → ACC	1	C, DC, Z
ANDIA	**I**	AND Immediate with ACC	ACC and I → ACC	1	Z
IORIA	**I**	OR Immediate with ACC	ACC or I → ACC	1	Z
XORIA	**I**	Exclusive OR Immediate to ACC	ACC xor I → ACC	1	Z
RETIA	**I**	Return, place Immediate in ACC	I → ACC, Top of Stack → PC	2	-
LCALL	**I**	Call subroutine	PC + 1 → Top of Stack, I → PC<10:0> I <10:8> → PCHBUF<2:0>	2	-
LGOTO	**I**	Unconditional branch	I → PC<10:0> I <10:8> → PCHBUF<2:0>	2	-
TMSZA		If (ACC) =0, skip next instruction	Skip if ACC = 0	1/2(1)	-
TMSZR	**R**	If (R) =0, skip next instruction	Skip if R = 0	1/2(1)	-
TMSNZR	**R**	If (R) ≠ 0, skip next instruction	Skip if R ≠ 0	1/2(1)	-
TMCOMP	**R**	If (acc) =(R), skip next instruction	Skip if (acc) =(R)	1/2(1)	-
TMCOMPB	**R**	If (acc) ≠(R), skip next instruction	Skip if (acc) ≠ (R)	1/2(1)	-

Note: 1. 2 cycles for skip, else 1 cycle.

2. bit : Bit address within an 8-bit register R
   R : Register address (00h to BFh)
   I : Immediate data
   ACC :Accumulator
   d : Destination select;
   =0 (store result in ACC)
   =1 (store result in file register R)
   dest : Destination
   PC : Program Counter
   PCHBUF : Program Counter High-byte buffer
   TB_BNK : Table Look-up Bank selection register
   WDT : Watchdog Timer Counter
   GIE : Global interrupt enable bit
   $\overline{TO}$: Time-out bit
   $\overline{PD}$: Power-down bit
   C : Carry bit
   DC : Digital carry bit
   Z : Zero bit

ADCAR	**Add ACC and R with Carry**
Syntax:	ADCAR R, d
Operands:	0 ≤ R ≤ 0xBF d ∈ [0,1]
Operation:	R + ACC + C → dest
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register and register 'R' with Carry. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1

ADDAR	**Add ACC and R**
Syntax:	ADDAR R, d
Operands:	0 ≤ R ≤ 0xBF d ∈ [0,1]
Operation:	ACC + R → dest
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register and register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1

ADDIA	**Add ACC and Immediate**
Syntax:	ADDIA I
Operands:	0 ≤ I ≤ 0xFF
Operation:	ACC + I → ACC
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

ANDAR	**AND ACC and R**
Syntax:	ANDAR R, d
Operands:	0 ≤ R ≤ 0xBF d ∈ [0,1]
Operation:	ACC and R → dest
Status Affected:	Z
Description:	The contents of the ACC register are AND'ed with register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1

ANDIA	**AND Immediate with ACC**
Syntax:	ANDIA I
Operands:	0 ≤ I ≤ 0xFF
Operation:	ACC AND I → ACC
Status Affected:	Z
Description:	The contents of the ACC register are AND'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

BCR	**Clear Bit in R**
Syntax:	BCR R, b
Operands:	0 ≤ R ≤ 0xBF
	0 ≤ b ≤ 7
Operation:	0 → R<b>
Status Affected:	None
Description:	Clear bit 'b' in register 'R'.
Cycles:	1

BSR	**Set Bit in R**
Syntax:	BSR R, b
Operands:	0 ≤ R ≤ 0xBF
	0 ≤ b ≤ 7
Operation:	1 → R<b>
Status Affected:	None
Description:	Set bit 'b' in register 'R'.
Cycles:	1

BTRSC	**Test Bit in R, Skip if Clear**
Syntax:	BTRSC R, b
Operands:	0 ≤ R ≤ 0xBF
	0 ≤ b ≤ 7
Operation:	Skip if R<b> = 0
Status Affected:	None
Description:	If bit 'b' in register 'R' is 0 then the next instruction is skipped. If bit 'b' is 0 then next instruction fetched during the current instruction execution is discarded, and a NOP is executed instead making this a 2-cycle instruction.
Cycles:	1/2

BTRSS	**Test Bit in R, Skip if Set**
Syntax:	BTRSS R, b
Operands:	0 ≤ R ≤ 0xBF
	0 ≤ b ≤ 7
Operation:	Skip if R<b> = 1
Status Affected:	None
Description:	If bit 'b' in register 'R' is '1' then the next instruction is skipped. If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a 2-cycle instruction.
Cycles:	1/2

CLRA	**Clear ACC**
Syntax:	CLRA
Operands:	None
Operation:	00h → ACC; 1 → Z
Status Affected:	Z
Description:	The ACC register is cleared. Zero bit (Z) is set.
Cycles:	1

CLRR	Clear R
Syntax:	CLRR R
Operands:	0 ≤ R ≤ 0xBF
Operation:	00h → R; 1 → Z
Status Affected:	Z
Description:	The contents of register 'R' are cleared and the Z bit is set.
Cycles:	1

CLRWDT	Clear Watchdog Timer
Syntax:	CLRWDT
Operands:	None
Operation:	00h → WDT; 1 → $\overline{TO}$; 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	The CLRWDT instruction resets the WDT. The status bits $\overline{TO}$ and $\overline{PD}$ will be set.
Cycles:	1

COMR	Complement R
Syntax:	COMR R, d
Operands:	0 ≤ R ≤ 0xBF d ∈ [0,1]
Operation:	$\overline{R}$ → dest
Status Affected:	Z
Description:	The contents of register 'R' are complemented. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

DAA	Adjust ACC's data format from HEX to DEC
Syntax:	DAA R, d
Operands:	0 ≤ R ≤ 0xBF d ∈ [0,1]
Operation:	R(hex) → dest(dec)
Status Affected:	C
Description:	Convert the register data from hexadecimal to decimal format after any addition operation. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

DAS	Adjust ACC's data format from HEX to DEC
Syntax:	DAS R, d
Operands:	0 ≤ R ≤ 0xBF d ∈ [0,1]
Operation:	R(hex) → dest(dec)
Status Affected:	C
Description:	Convert the register data from hexadecimal to decimal format after any subtraction operation. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

DECR	Decrement R
Syntax:	DECR R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R - 1 → dest
Status Affected:	Z
Description:	Decrement of register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

DECRSZ	Decrement R, Skip if 0
Syntax:	DECRSZ R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R - 1 → dest; skip if result =0
Status Affected:	None
Description:	The contents of register 'R' are decrement. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is stored back in register 'R'. If the result is 0, the next instruction, which is already fetched, is discarded and a NOP is executed instead and making it a two-cycle instruction.
Cycles:	1/2

INCR	Increment R
Syntax:	INCR R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R + 1 → dest
Status Affected:	Z
Description:	The contents of register 'R' are increment. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

INCRSZ	Increment R, Skip if 0
Syntax:	INCRSZ R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R + 1 → dest, skip if result = 0
Status Affected:	None
Description:	The contents of register 'R' are increment. If 'd' is 0 the result is placed in the ACC register. If 'd' is the result is stored back in register 'R'. If the result is 0, then the next instruction, which is already fetched, is discarded and a NOP is executed instead and making it a two-cycle instruction.
Cycles:	1/2

IORAR	OR ACC with R
Syntax:	IORAR R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	ACC or R → dest
Status Affected:	Z
Description:	Inclusive OR the ACC register with register 'R'. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

IORIA	OR Immediate with ACC
Syntax:	IORIA I
Operands:	0≤I≤0xFF
Operation:	ACC or I → ACC
Status Affected:	Z
Description:	The contents of the ACC register are OR'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

LCALL	Subroutine Call
Syntax:	LCALL I
Operands:	0≤I≤0x7FF
Operation:	PC + 1 → Top of Stack, I → PC<10:0> I <10:8> → PCHBUF<2:0>
Status Affected:	None
Description:	Subroutine call. First, return address (PC+1) is pushed onto the stack. The 11-bit immediate address is loaded into PC bits <10:0>.
Cycles:	2

LGOTO	Unconditional Branch
Syntax:	LGOTO I
Operands:	0≤I≤0x7FF
Operation:	I → PC<10:0> I <10:8> → PCHBUF<2:0>
Status Affected:	None
Description:	LGOTO is an unconditional branch. The 11-bit immediate value is loaded into PC bits <10:0>.
Cycles:	2

MOVAR	Move ACC to R
Syntax:	MOVAR R
Operands:	0≤R≤0xBF
Operation:	ACC → R
Status Affected:	None
Description:	Move data from the ACC register to register 'R'.
Cycles:	1

MOVIA	Move Immediate to ACC
Syntax:	MOVIA I
Operands:	0≤I≤0xFF
Operation:	I → ACC
Status Affected:	None
Description:	The 8-bit immediate 'I' is loaded into the ACC register. The don't cares will assemble as 0s.
Cycles:	1

MOVR	Move R
Syntax:	MOVR R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R → dest
Status Affected:	Z
Description:	The contents of register 'R' is moved to destination 'd'. If 'd' is 0, destination is the ACC register. If 'd' is 1, the destination is file register 'R'. 'd' is 1 is useful to test a file register since status flag Z is affected.
Cycles:	1

MOV2	Move R
Syntax:	MOV2 R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R → dest
Status Affected:	None
Description:	The contents of register 'R' is moved to destination 'd'. If 'd' is 0, destination is the ACC register. If 'd' is 1, the destination is file register 'R'. The zero status flag <Z> is not affected.
Cycles:	1

NOP	No Operation
Syntax:	NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.
Cycles:	1

RETF	Return from Interrupt
Syntax:	RETF
Operands:	None
Operation:	Top of Stack → PC
Status Affected:	None
Description:	The program counter is loaded from the top of the stack (the return address). The 'GIE' bit would NOT be set to 1. This is a two-cycle instruction.
Cycles:	2

RETFIE	Return from Interrupt, Set 'GIE' Bit
Syntax:	RETFIE
Operands:	None
Operation:	Top of Stack → PC 1 → GIE
Status Affected:	None
Description:	The program counter is loaded from the top of the stack (the return address). The 'GIE' bit is set to 1. This is a two-cycle instruction.
Cycles:	2

RETIA	Return with Immediate in ACC
Syntax:	RETIA I
Operands:	0≤I≤0xFF
Operation:	I → ACC; Top of Stack → PC
Status Affected:	None
Description:	The ACC register is loaded with the 8-bit immediate 'I'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Cycles:	2

RETURN	Return from Subroutine
Syntax:	RETURN
Operands:	None
Operation:	Top of Stack → PC
Status Affected:	None
Description:	The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Cycles:	2

RL	Rotate Left R
Syntax:	RL R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R<6:0> → dest<7:1>, R<7>→ dest<0>
Status Affected:	None
Description:	The contents of register 'R' are rotated left one bit. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

RL0	Rotate Left R with 0
Syntax:	RL0 R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R<6:0> → dest<7:1>, 0 → dest<0>
Status Affected:	None
Description:	The contents of register 'R' are rotated left one bit to the left and bit0 fills with "0". If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

RL1	Rotate Left R with 1
Syntax:	RL1 R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R<6:0> → dest<7:1>, 1 → dest<0>
Status Affected:	None
Description:	The contents of register 'R' are rotated left one bit to the left and bit0 fills with "1". If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

RLR	**Rotate Left R through Carry**
Syntax:	RLR R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R<7> → C; R<6:0> → dest<7:1>; C → dest<0>
Status Affected:	C
Description:	The contents of register 'R' are rotated left one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

RR	**Rotate Right R**
Syntax:	RR R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	R<7:1> → dest<6:0>, R<0> → dest<7>
Status Affected:	None
Description:	The contents of register 'R' are rotated right one bit. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

RR0	**Rotate Right R with 0**
Syntax:	RR0 R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	0 → dest<7>, R<7:1> → dest<6:0>
Status Affected:	None
Description:	The contents of register 'R' are rotated right one bit and bit7 fills with "0". If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

RR1	**Rotate Right R with 1**
Syntax:	RR1 R, d
Operands:	0≤R≤0xBF d∈ [0,1]
Operation:	1 → dest<7>, R<7:1> → dest<6:0>
Status Affected:	None
Description:	The contents of register 'R' are rotated right one bit and bit7 fills with "1". If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

RRR	Rotate Right R through Carry
Syntax:	RRR R, d
Operands:	0≤R≤0xBF d∈[0,1]
Operation:	C → dest<7>; R<7:1> → dest<6:0>; R<0> → C
Status Affected:	C
Description:	The contents of register 'R' are rotated one bit to the right through the Carry Flag. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

SLEEP	Enter SLEEP Mode
Syntax:	SLEEP
Operands:	None
Operation:	00h → WDT; 1 → $\overline{TO}$; 0 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	Time-out status bit ($\overline{TO}$) is set. The power-down status bit ($\overline{PD}$) is cleared. The WDT is cleared. The processor is put into SLEEP mode.
Cycles:	1

SBCAR	Subtract ACC from R with Carry
Syntax:	SBCAR R, d
Operands:	0≤R≤0xBF d∈[0,1]
Operation:	R + $\overline{ACC}$ + C → dest
Status Affected:	C, DC, Z
Description:	Add the 2's complement data of the ACC register from register 'R' with Carry. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

SUBAR	Subtract ACC from R
Syntax:	SUBAR R, d
Operands:	0≤R≤0xBF d∈[0,1]
Operation:	R - ACC → dest
Status Affected:	C, DC, Z
Description:	Subtract (2's complement method) the ACC register from register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

SUBIA	Subtract ACC from Immediate
Syntax:	SUBIA I
Operands:	0≤I≤0xFF
Operation:	I - ACC → ACC
Status Affected:	C, DC, Z
Description:	Subtract (2's complement method) the ACC register from the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

SWAPR	**Swap nibbles in R**
Syntax:	SWAPR R, d
Operands:	0≤R≤0xBF d∈[0,1]
Operation:	R<3:0> → dest<7:4>; R<7:4> → dest<3:0>
Status Affected:	None
Description:	The upper and lower nibbles of register 'R' are exchanged. If 'd' is 0 the result is placed in ACC register. If 'd' is 1 the result in placed in register 'R'.
Cycles:	1

TABL	**Table Look-up Low Byte**
Syntax:	TABL R
Operands:	0≤R≤0xBF
Operation:	ACC=ROM{TB_BNK index : R}[7:0]
Status Affected:	None
Description:	Read low byte ROM table to (ACC) ROM table address={TB_BNK index : R}
Cycles:	2

TABH	**Table Look-up High Byte**
Syntax:	TABH R
Operands:	0≤R≤0xBF
Operation:	ACC=ROM{TB_BNK index : R}[15:8]
Status Affected:	None
Description:	Read High byte ROM table to (ACC) ROM table address={TB_BNK index : R}
Cycles:	2

TMCOMP	**Test ACC and R, Skip if equal**
Syntax:	TMCOMP R
Operands:	0≤R≤0xBF
Operation:	Skip if ACC = R
Status Affected:	None
Description:	If ACC is equal to R then the next instruction is skipped. If ACC is equal to R then next instruction fetched during the current instruction execution is discarded, a NOP is executed instead and making this a 2-cycle instruction.
Cycles:	1/2

TMCOMPB	**Test ACC and R, Skip if not equal**
Syntax:	TMCOMPB R
Operands:	0≤R≤0xBF
Operation:	Skip if ACC ≠ R
Status Affected:	None
Description:	If ACC is not equal to R then the next instruction is skipped. If ACC is not equal to R then next instruction fetched during the current instruction execution is discarded, a NOP is executed instead and making this a 2-cycle instruction.
Cycles:	1/2

TMSZA	**Test ACC, Skip if equal to 0**
Syntax:	TMSZA
Operands:	
Operation:	Skip if ACC = 0
Status Affected:	None
Description:	If ACC is equal to 0 then the next instruction is skipped. If ACC is equal to 0 then next instruction fetched during the current instruction execution is discarded, a NOP is executed instead and making this a 2-cycle instruction.
Cycles:	1/2

TMSNZR	**Test R, Skip if not equal to 0**
Syntax:	TMSNZR R
Operands:	$0 \leq R \leq 0xBF$
Operation:	Skip if $R \neq 0$
Status Affected:	None
Description:	If R is not equal to 0 then the next instruction is skipped. If R is not equal to 0 then next instruction fetched during the current instruction execution is discarded, a NOP is executed instead and making this a 2-cycle instruction.
Cycles:	1/2

TMSZR	Test R, Skip if equal to 0
Syntax:	TMSZR R
Operands:	$0 \leq R \leq 0xBF$
Operation:	Skip if R = 0
Status Affected:	None
Description:	If R is equal to 0 then the next instruction is skipped. If R is equal to 0 then next instruction fetched during the current instruction execution is discarded, a NOP is executed instead and making this a 2-cycle instruction.
Cycles:	1/2

XORAR	Exclusive OR ACC with R
Syntax:	XORAR R, d
Operands:	$0 \leq R \leq 0xBF$ $d \in [0,1]$
Operation:	ACC xor R → dest
Status Affected:	Z
Description:	Exclusive OR the contents of the ACC register with register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

XORIA	**Exclusive OR Immediate with ACC**
Syntax:	XORIA I
Operands:	$0 \leq I \leq 0xFF$
Operation:	ACC xor I → ACC
Status Affected:	Z
Description:	The contents of the ACC register are XOR'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

4.0 ABSOLUTE MAXIMUM RATINGS

Ambient Operating Temperature	-40℃ to +85℃
Store Temperature	-65℃ to +150℃
DC Supply Voltage (Vdd)	0V to +6.0V
Input Voltage with respect to Ground (Vss)	-0.3V to (Vdd + 0.3)V

5.0 OPERATING CONDITIONS

DC Supply Voltage	+2.2V to +5.5V
Operating Temperature	-40℃ to +85℃

6.0 ELECTRICAL CHARACTERISTICS

6.1 ELECTRICAL CHARACTERISTICS of FM8PA76AE/BE/DE/EE/FE

Ta=25℃

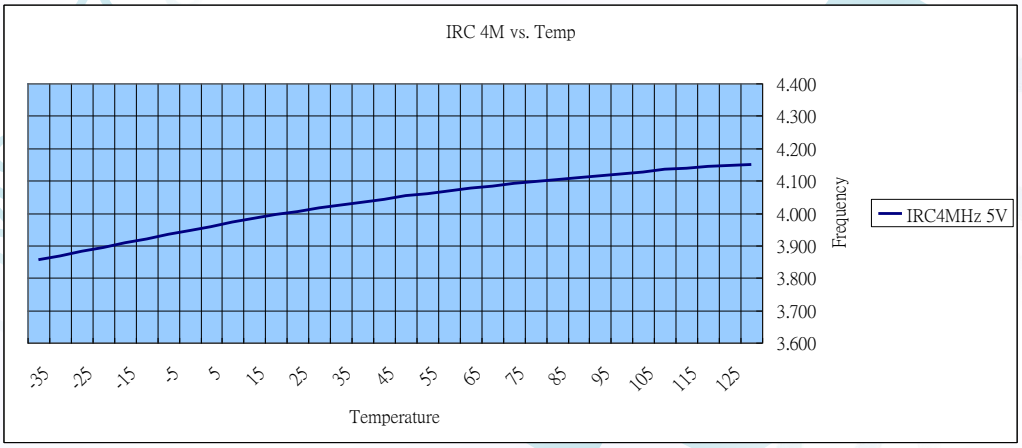
Under Operating Conditions, at four clock instruction cycles and WDT & LVDT are disabled

Sym	Description	Conditions	Min.	Typ.	Max.	Unit
F_{HF}	X'tal oscillation range	HF mode, Vdd=5V, Fcpu=Fosc/2			20	MHz
		HF mode, Vdd=3V, Fcpu=Fosc/2			15	
F_{XT}	X'tal oscillation range	XT mode, Vdd=5V, Fcpu=Fosc/2			10	MHz
		XT mode, Vdd=3V, Fcpu=Fosc/2			10	
F_{LF}	X'tal oscillation range	LF mode, Vdd=5V, Fcpu=Fosc/2			4000	KHZ
		LF mode, Vdd=3V, Fcpu=Fosc/2			1000	
F_{ERC}	RC oscillation range	ERC mode, Vdd=5V, Fcpu=Fosc/2			15	MHz
		ERC mode, Vdd=3V, Fcpu=Fosc/2			7	
V_{IH}	Input high voltage	With schmitter				V
		I/O ports	0.7Vdd		Vdd	
		RSTB pin	0.8Vdd		Vdd	
V_{IL}	Input low voltage	With schmitter				V
		I/O ports	Vss		0.2Vdd	
		RSTB pin	Vss		0.2Vdd	
I_{IL}	Input Leakage Current	Vin = 5V, Vdd=5V			1	uA
		Vin = 0V, Vdd=5V			1	
I_{OH}	IO Drive Current	VOH =4.5V, Vdd = 5V		9		mA
		VOH =4V, Vdd = 5V		17		
I_{OL}	IO Sink Current	VOL =0.5V, Vdd = 5V		19		mA
		VOL =0.75V, Vdd = 5V		26		
R_{PH}	Pull-high resister	Input pin at Vss, vdd=5V	65	130	195	KΩ
		Input pin at Vss, vdd=3V	125	250	375	
I_{WDT}	WDT current	Vdd=5V		8		uA
		Vdd=3V		2		
T_{WDT}	WDT period	Vdd=3V		24		mS
		Vdd=5V		20		
I_{LVDT}	LVDT current	LVDT = 3.7V, vdd=5V		2		uA
		LVDT = 2.6V, vdd=5V		3		
		LVDT = 2.6V, vdd=3V		0.5		
		LVDT = 2.2V, vdd=5V		3		
		LVDT = 2.2V, vdd=3V		0.5		
V_{LVDT}	LVDT voltage	LVDT = 3.7V	3.5	3.7	3.9	V
		LVDT = 2.6V	2.4	2.6	2.8	
		LVDT = 2.2V	2.0	2.2	2.4	
V_{AD}	A/D input Voltage		0		Vdd	V
R_{AD}	Resolution				12	Bits
DNL	A/D Differential Non-Linear				2	LSB
INL	A/D Integral Non-Linear				3	LSB
I_{ADC}	A/D Operation Current	Vdd = 5V, Fcpu=Fosc/4		600		uA
		Vdd = 3V, Fcpu=Fosc/4		100		
T_{AD}	A/D clock period		8			us

Sym	Description	Conditions	Min.	Typ.	Max.	Unit
T_{ADC}	A/D Conversion Time			25		T_{AD}
T_{ADCS}	A/D Sampling Time			8		T_{AD}
I_{SB}	Power down current	Sleep mode, Vdd=5V, WDT enable, LVDT off		9		uA
		Sleep mode, Vdd=5V, WDT disable, LVDT off			1	
		Sleep mode, Vdd=3V, WDT enable, LVDT off		3		
		Sleep mode, Vdd=3V, WDT disable, LVDT off			1	
I_{DD1}	Operating current	IRC mode, vdd=5V, 4 clock instruction		0.9		mA
I_{DD2}	Operating current	IRC mode, vdd=5V, 2 clock instruction		1.4		mA
I_{DD3}	Operating current	IRC mode, vdd=3V, 4 clock instruction		0.5		mA
I_{DD4}	Operating current	IRC mode, vdd=3V, 2 clock instruction		0.7		mA
I_{DD5}	Operating current	HF mode, vdd=5V, 4 clock instruction				mA
		20MHz		4		
I_{DD6}	Operating current	HF mode, vdd=5V, 2 clock instruction				mA
		20MHz		6		
I_{DD7}	Operating current	HF mode, vdd=3V, 4 clock instruction				mA
		20MHZ		2		
I_{DD8}	Operating current	XT mode, Vdd=5V, 4 clock instruction				
		10MHz		3		mA
		4MHz		1.5		
I_{DD9}	Operating current	XT mode, Vdd=5V, 2 clock instruction				mA
		10MHz		3.5		
		4MHz		1.8		
I_{DD10}	Operating current	XT mode, Vdd=3V, 4 clock instruction				mA
		10MHz		1		
		4MHz		0.5		
I_{DD11}	Operating current	XT mode, Vdd=3V, 2 clock instruction				mA
		10MHz		1.5		
		4MHz		0.8		
I_{DD}	Operating current	LF mode, Vdd=5V, 4 clock instruction				uA
		32KHz		30		
I_{DD12}	Operating current	LF mode, Vdd=5V, 2 clock instruction				uA
		32KHz		35		
I_{DD13}	Operating current	LF mode, Vdd=3V, 4 clock instruction				uA
		32KHz		8		
I_{DD14}	Operating current	LF mode, Vdd=3V, 2 clock instruction				uA
		32KHz		10		

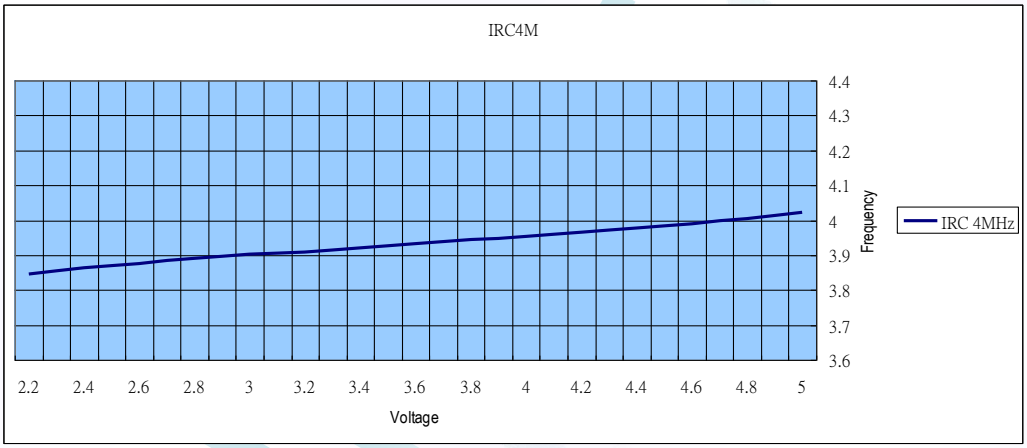
6.2 ELECTRICAL CHARACTERISTICS Charts of FM8PA76AE/BE/DE/EE/FE

6.2.1 Internal 4MHz RC vs. Temperature (VDD=5V)



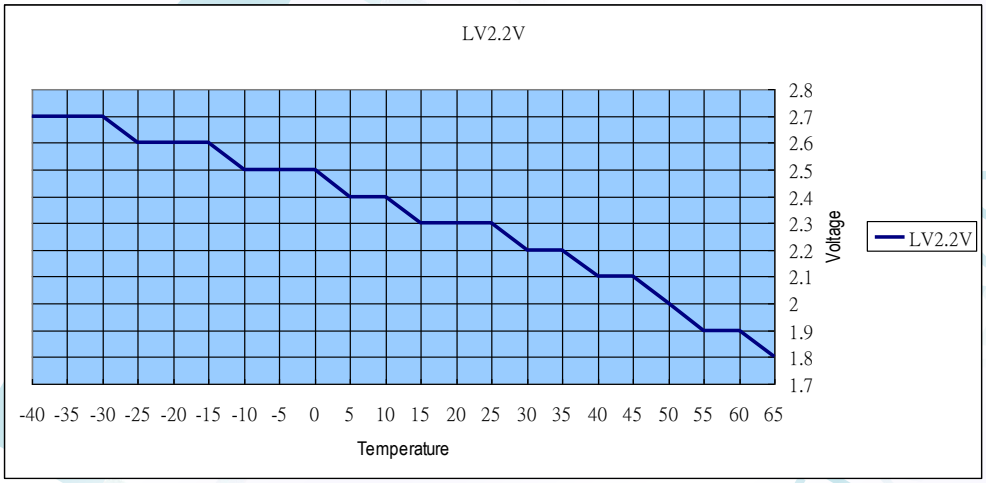
Note: Curves are for design reference only.

6.2.2 Internal 4MHz RC vs. Supply Voltage (Ta=25℃)



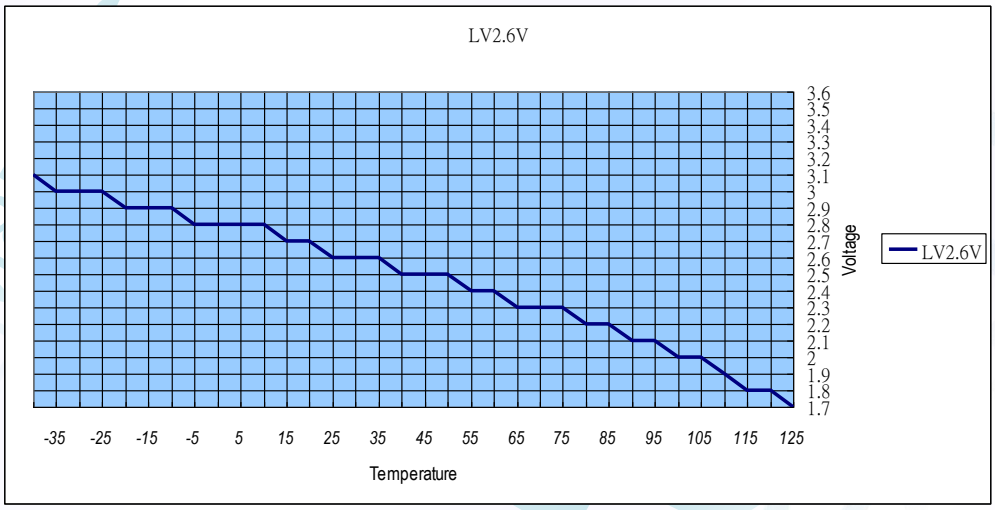
Note: Curves are for design reference only.

6.2.3 Low Voltage Detect (LVDT=2.2V) vs. Temperature



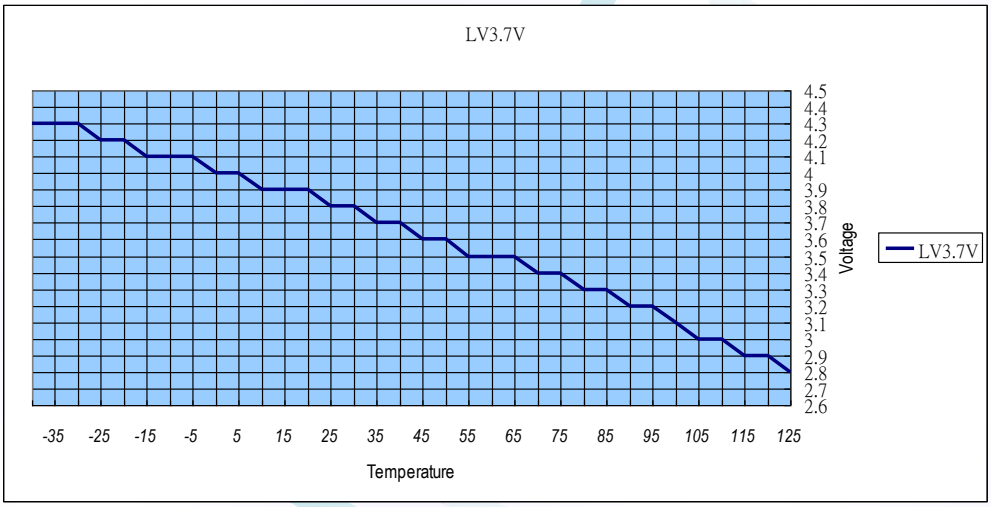
CAUTION: The LVDT 2.2V option can only support temperature range between -40~65℃
Note: Curves are for design reference only.

6.2.4 Low Voltage Detect (LVDT=2.6V) vs. Temperature



Note: Curves are for design reference only.

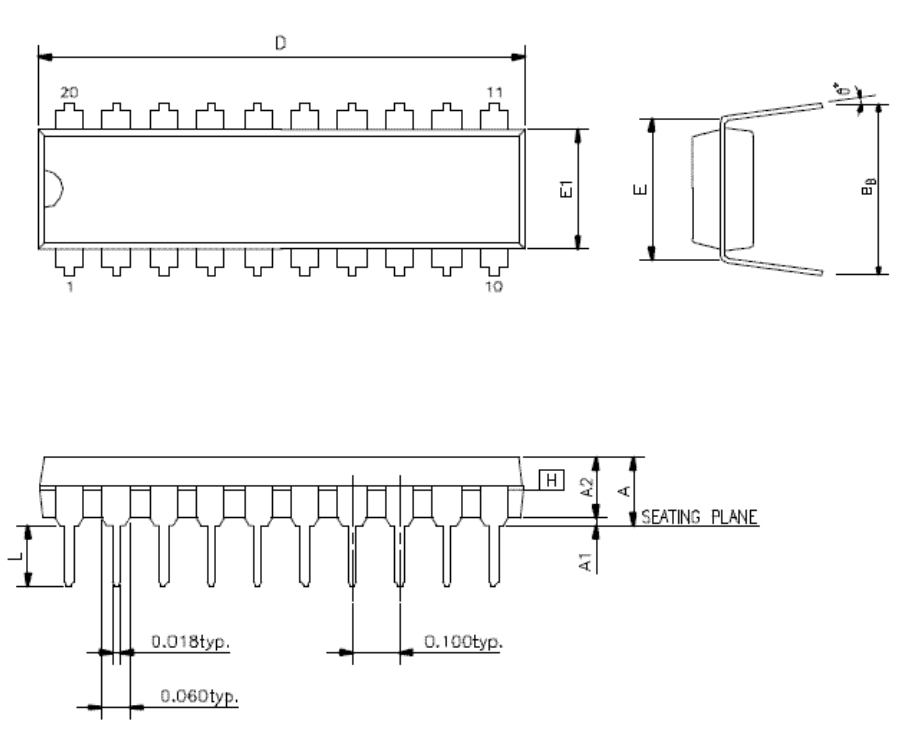
6.2.5 Low Voltage Detect (LVDT=3.7V) vs. Temperature



Note: Curves are for design reference only.

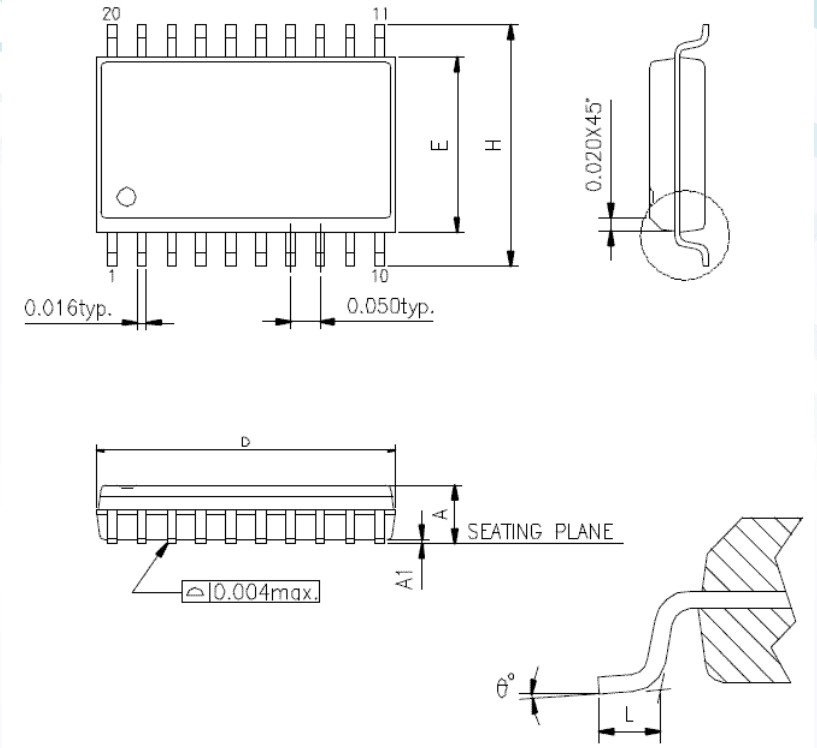
7.0 PACKAGE DIMENSION

7.1 20-PIN PDIP



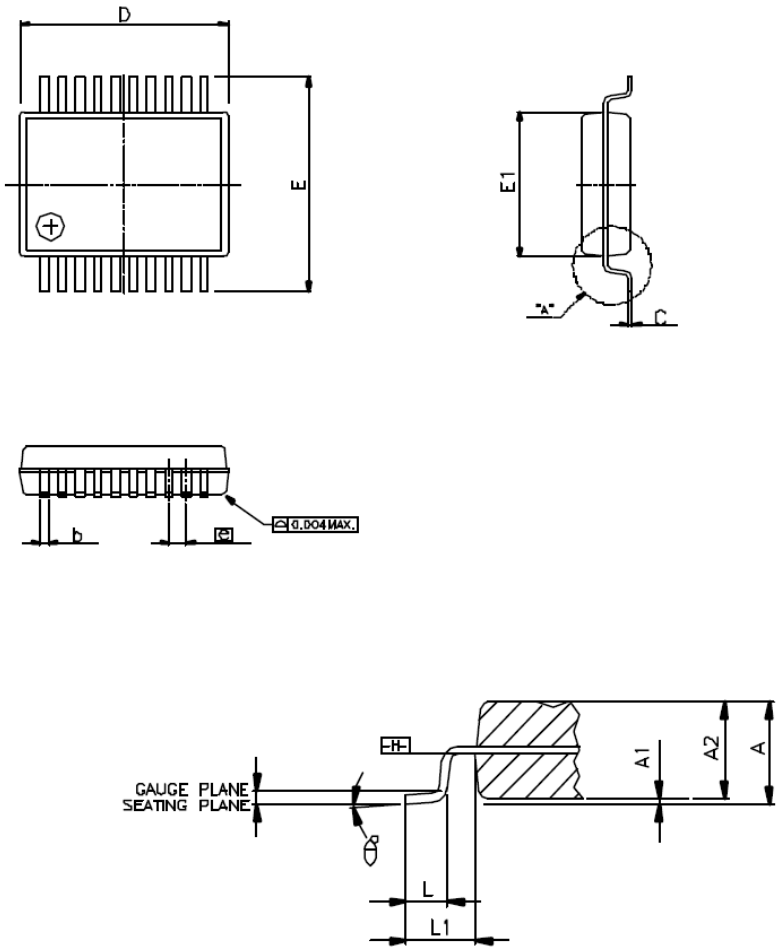
Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.210
A1	0.015	-	-
A2	0.125	0.130	0.135
D	0.98	1.030	1.060
E		0.300 BSC	
E1	0.245	0.250	0.255
L	0.115	0.130	0.150
eB	0.335	0.355	0.375
θ°	0°	7°	15°

7.2 20-PIN SOP



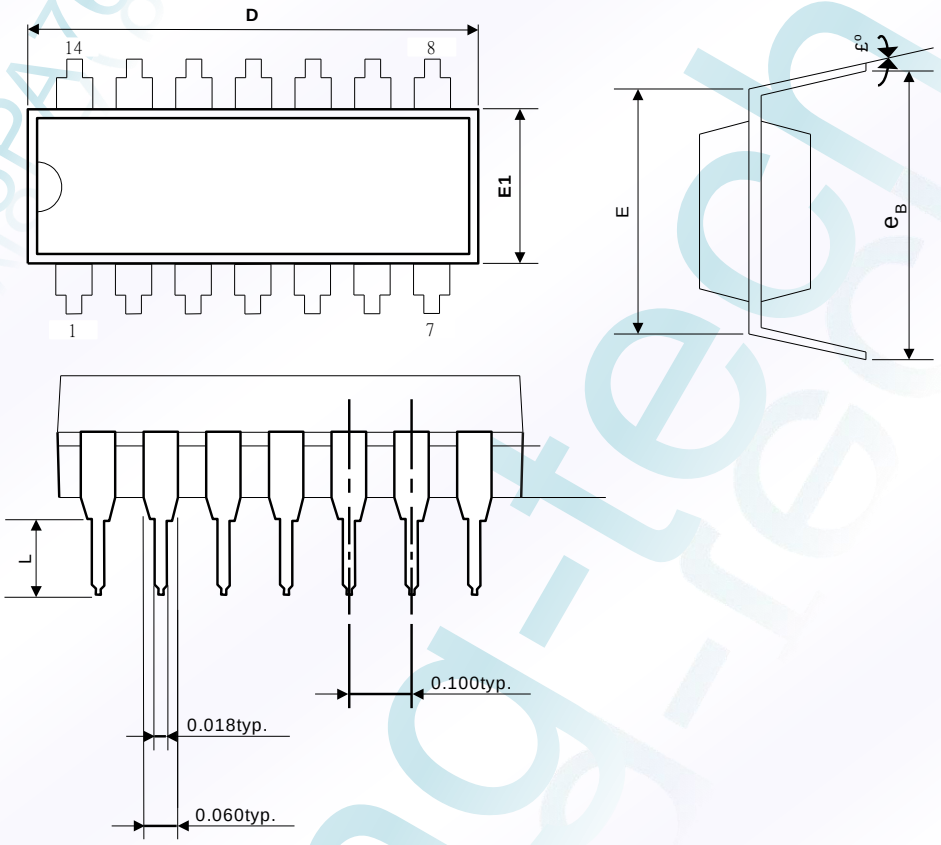
Symbols	Dimension In Inches		
	Min	Nom	Max
A	0.093	-	0.104
A1	0.004	-	0.012
D	0.496	-	0.508
E	0.291	-	0.299
H	0.394	-	0.419
L	0.016	-	0.050
θ°	0°	-	8°

7.3 20-PIN SSOP 209 mil



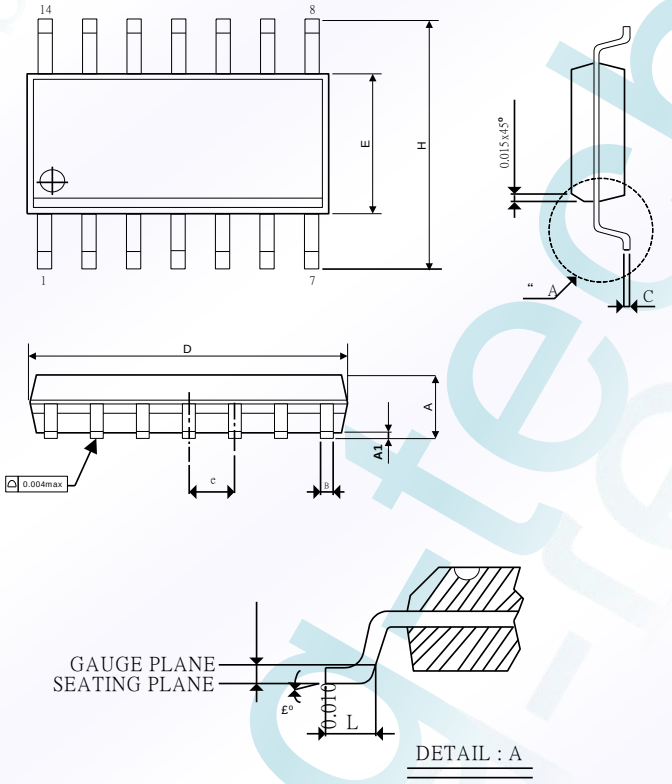
Symbols	Dimension In Millimeters		
	Min	Nom	Max
A	-	-	2.00
A1	0.05	-	-
A2	1.65	1.75	1.85
b	0.22	-	0.38
c	0.09	-	0.21
D	6.90	7.20	7.50
E	7.40	7.80	8.20
E1	5.00	5.30	5.60
e	-	0.65	-
L	0.55	0.75	0.95
L1	-	1.25	-
θ°	0°	4°	8°

7.4 14-PIN PDIP 300mil



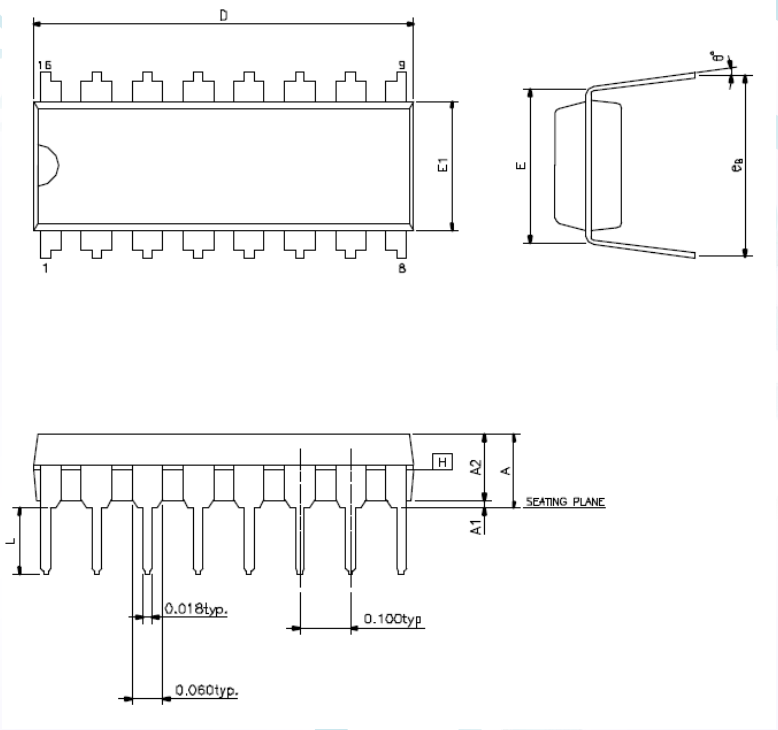
Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.210
A1	0.015	-	-
A2	0.125	0.130	0.135
D	0.735	0.750	0.775
E	0.300 BSC.		
E1	0.245	0.250	0.255
L	0.115	0.130	0.150
eB	0.335	0.355	0.375
θ°	0°	7°	15°

7.5 14-PIN SOP 150mil



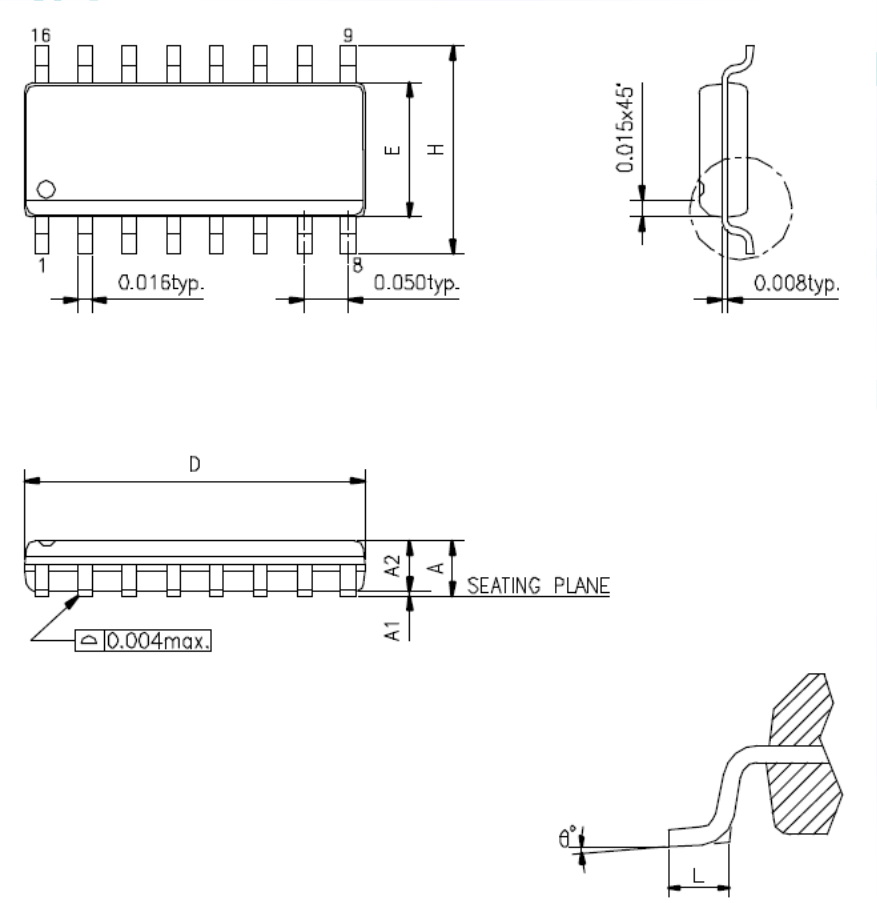
Symbols	Dimension In Inches		
	Min	Nom	Max
A	0.058	0.064	0.068
A1	0.004	-	0.010
B	0.013	0.016	0.020
C	0.0075	0.008	0.0098
D	0.336	0.341	0.344
E	0.150	0.154	0.157
e	-	0.050	-
H	0.228	0.236	0.244
L	0.015	0.025	0.050
θ°	0°	-	8°

7.6 16-PIN PDIP 300mil



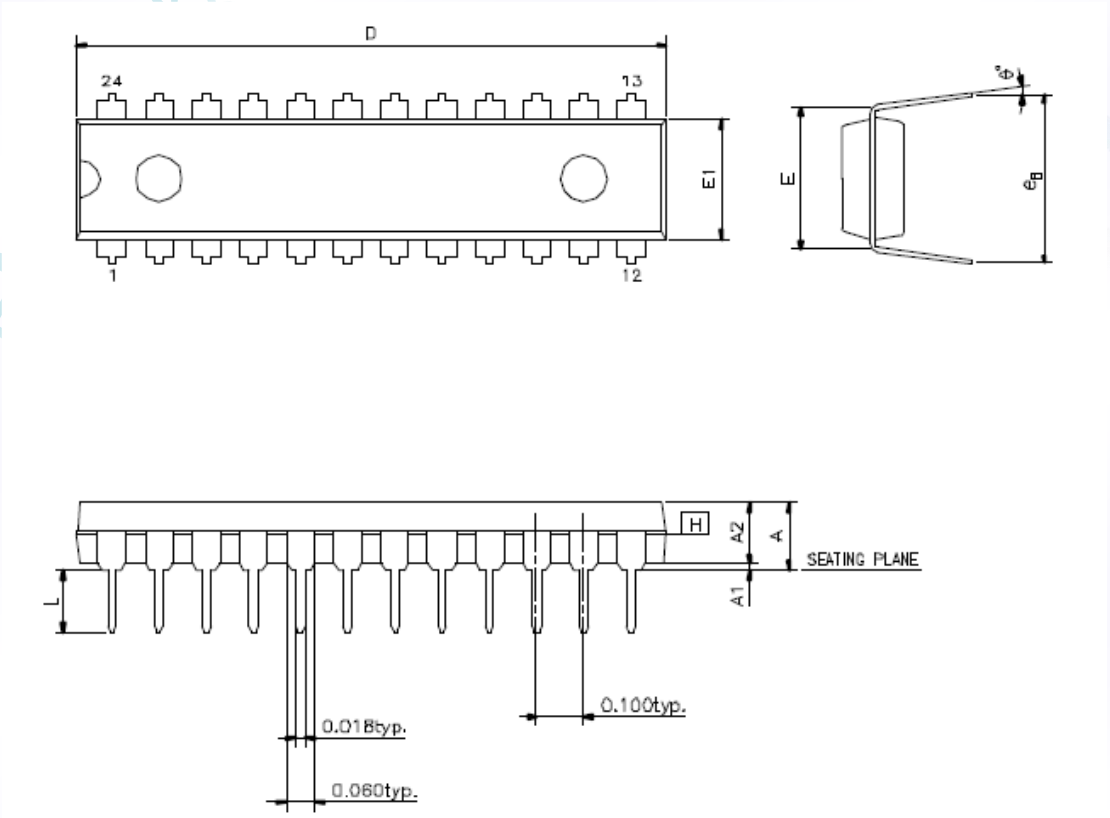
Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.210
A1	0.015	-	-
A2	0.125	0.130	0.135
D	0.735	0.755	0.775
E		0.300 BSC	
E1	0.245	0.250	0.255
L	0.115	0.130	0.150
eB	0.335	0.355	0.375
θ°	0°	7°	15°

7.7 16-PIN SOP 150mil



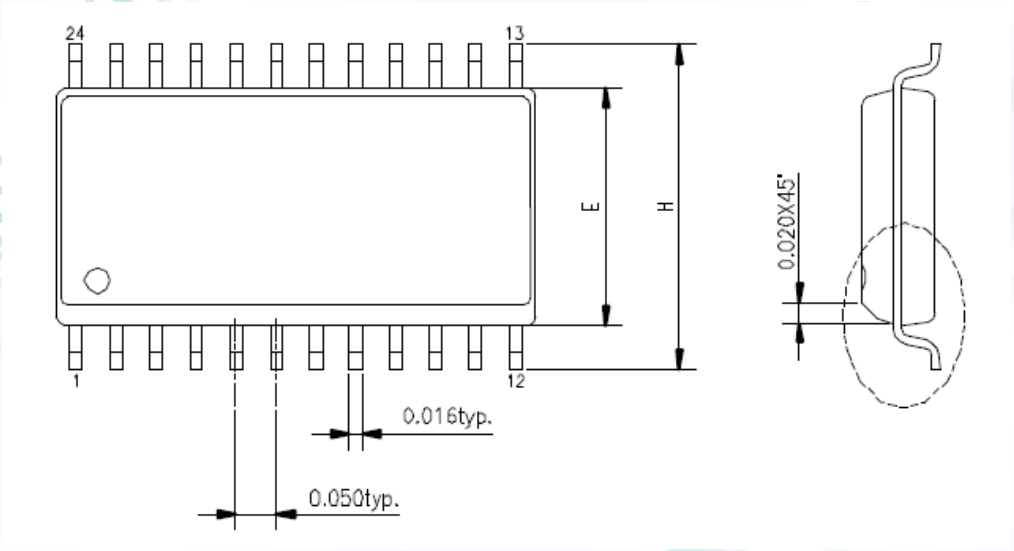
Symbols	Dimension In Inches	
	Min	Max
A	0.053	0.069
A1	0.004	0.010
A2	0.049	0.065
D	0.386	0.394
E	0.150	0.157
H	0.228	0.244
L	0.016	0.050
θ°	0°	8°

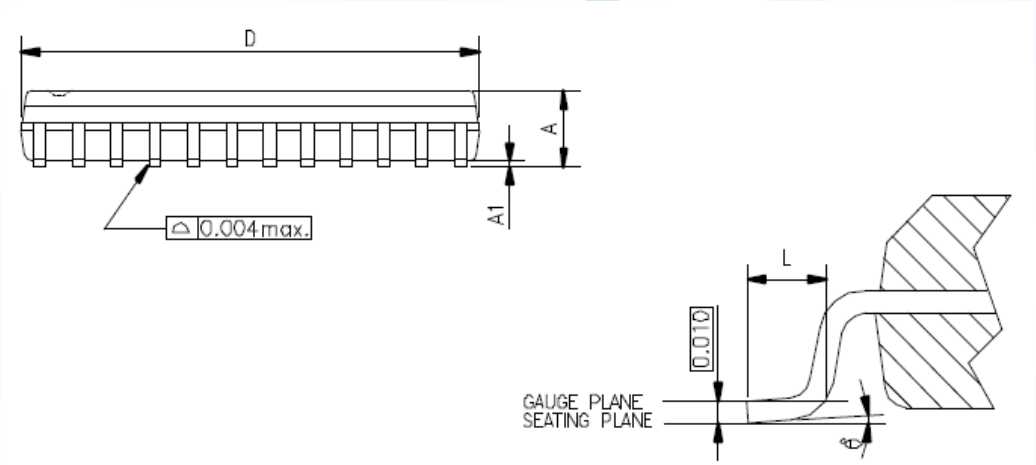
7.8 24-PIN PDIP 300mil (SKINNY)



Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.210
A1	0.015	-	-
A2	0.125	0.130	0.135
D	1.230	1.250	1.280
E	0.300 BSC.		
E1	0.253	0.258	0.263
L	0.115	0.130	0.150
eB	0.335	0.355	0.375
θ°	0°	7°	15°

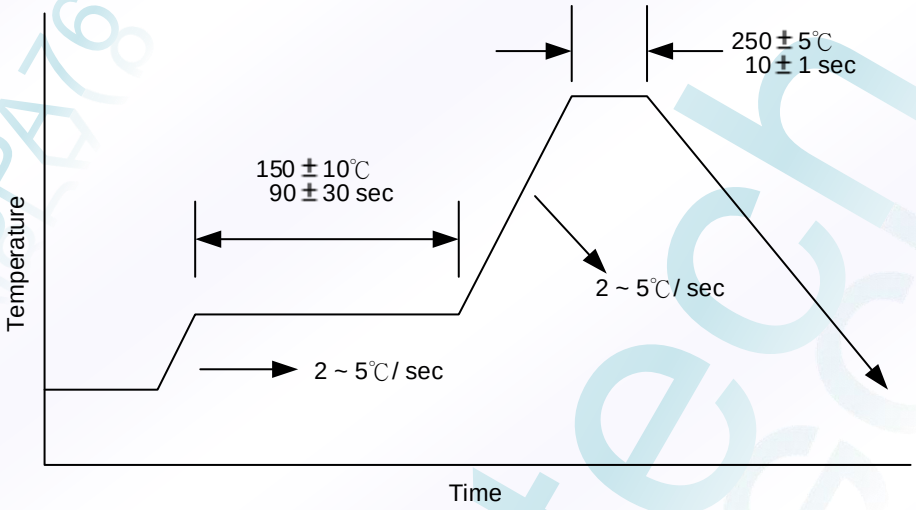
7.9 24-PIN SOP 300mil





Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.104
A1	0.004	-	-
D	0.599	0.600	0.624
E	0.291	0.295	0.299
H	0.394	0.406	0.419
L	0.016	0.035	0.050
θ°	0°	4°	8°

8.0 PACKAGE IR Re-flow Soldering Curve

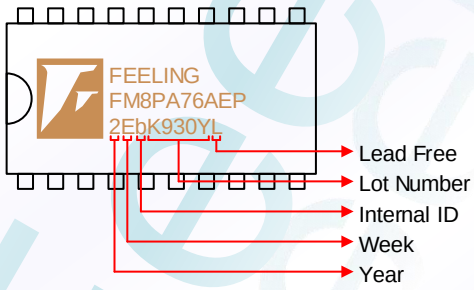


9.0 ORDERING INFORMATION (For Any customer)

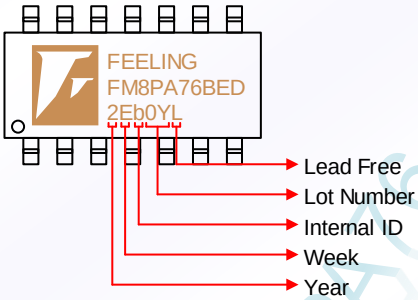
OTP Type MCU	Package Type	Pin Count	Package Size	SAMPLE Stock
FM8PA76AEP	PDIP	20	300 mil	Available
FM8PA76AED	SOP	20	300 mil	Available
FM8PA76AER	SSOP	20	209 mil	Available
FM8PA76BEP	PDIP	14	300 mil	Available
FM8PA76BED	SOP	14	150 mil	Available
FM8PA76DEP	PDIP	16	300 mil	Available
FM8PA76DED	SOP	16	150 mil	Available
FM8PA76EEP	PDIP	24	300 mil	Available
FM8PA76EED	SOP	24	300 mil	Available
FM8PA76FEP	PDIP	16	300 mil	Available
FM8PA76FED	SOP	16	150 mil	Available

9.1 MARKING INFROMATION

Marking type A



Marking type B



Lead Free: Lead free product indicator.
Lot Number: Wafer lot numbers, marking type A is show all digit, marking type B is show last two digit.
Internal ID: Internal identification code.
Week: Production period indicator in week time unit.
Year: Production year's last digit.

10.0 ORDERING INFORMATION (Only for Feeling-tech customer)

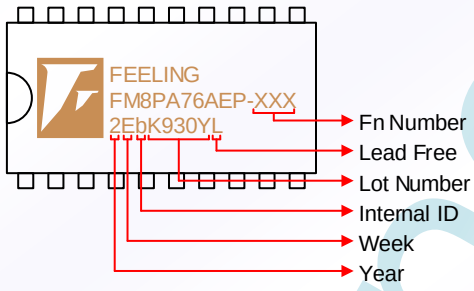
Ordering information in the table applies only to some customers of Feeling-tech corp., before ordering please contact sales representatives:
E-mail: chien_lw@feeling-tech.com.tw
Telephone: +886-3-560-5588 ext. 680

OTP Type MCU	Package Type	Pin Count	Package Size	SAMPLE Stock
FM8PA76AEP-XXX	PDIP	20	300 mil	Call FTC sales
FM8PA76AED-XXX	SOP	20	300 mil	Call FTC sales
FM8PA76AER-XXX	SSOP	20	209 mil	Call FTC sales
FM8PA76BEP-XXX	PDIP	14	300 mil	Call FTC sales
FM8PA76BED-XXX	SOP	14	150 mil	Call FTC sales
FM8PA76DEP-XXX	PDIP	16	300 mil	Call FTC sales
FM8PA76DED-XXX	SOP	16	150 mil	Call FTC sales

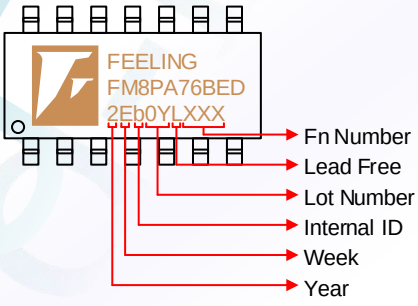
Please note: These products are available only package, does not apply to all sales representatives and vendor.

10.1 MARKING INFROMATION

Marking type A



Marking type B



Fn Number: This device function identification number (Range: Y00 ~ ZZZ).
Lead Free: Lead free product indicator.
Lot Number: Wafer lot numbers, marking type A is show all digit, marking type B is show last two digit.
Internal ID: Internal identification code.
Week: Production period indicator in week time unit.
Year: Production year's last digit.