FEELING TECHNOLOGY

FM8PE59M

OTP-Based 8-Bit Microcontroller

Devices Included in this Data Sheet:

- FM8PE59MA: 28-pin OTP device
- FM8PE59MB: 32-pin OTP device

FEATURES

- Only 49 single word instructions.
- All instructions are single cycle except for program branches which are two-cycle.
- All OTP area GOTO/FGOTO instruction.
- All OTP area subroutine CALL/FCALL instruction.
- 8-bit wide data path.
- 5-level deep hardware stack.
- 4K Word on chip OTP.
- 144 x 8 bits on chip general purpose registers (SRAM).
- Operating speed: DC-20 M_{HZ} clock input, or DC-100 ns instruction cycle.
- Direct, indirect addressing modes for data accessing.
- One 8-bit real time clock/counter (Timer0) with 8-bit programmable pre-scaler.
- One 8-bit real time clock/counter (Timer1) with 2-bit programmable pre-scaler and period setting.
- Internal Power-on Reset (POR).
- Built-in Low Voltage Detector (LVD) for Brown-out Reset (BOR).
- Power-up Reset Timer (PWRT) and Oscillator Start-up Timer(OST).
- On chip Watchdog Timer (WDT) with internal oscillator for reliable operation and soft-ware watch-dog enable/disable control.
- Three I/O ports IOA, IOB and IOC with independent direction control.
- 16 soft-ware control pull-high pins: Port B/Port C.
- 8 soft-ware control pull-down pins: IOA0~A3/IOB0~B3.
- 2 soft-ware control open-drain pins: IOC6/IOC7.
- IR output channel with programmable frequency and duty cycle.
- Serial Peripheral Interface (SPI).
- Five internal interrupt source: Timer0 overflow, Timer1 match, IROUT, SPI module and Low-voltage detector; Two external interrupt source: INT0 pin, and INT1 pin.
- Wake-up from SLEEP by Port B/IOC4/IOC5 input falling edge.
- Power saving SLEEP mode.
- Built-in 8M_{HZ}, 4M_{HZ}, 1M_{HZ}, and 455K_{HZ} internal RC oscillator.
- Programmable Code Protection.
- Selectable oscillator options:
  - ERC: External Resistor/Capacitor Oscillator.
  - HF: High Frequency Crystal/Resonator Oscillator.
  - XT: Crystal/Resonator Oscillator.
  - LF: Low Frequency Crystal Oscillator.
  - IRC: Internal Resistor/Capacitor Oscillator.
  - ERIC: External Resistor/Internal Capacitor Oscillator.
- Operating voltage range: 2.3V to 5.5V.

GENERAL DESCRIPTION

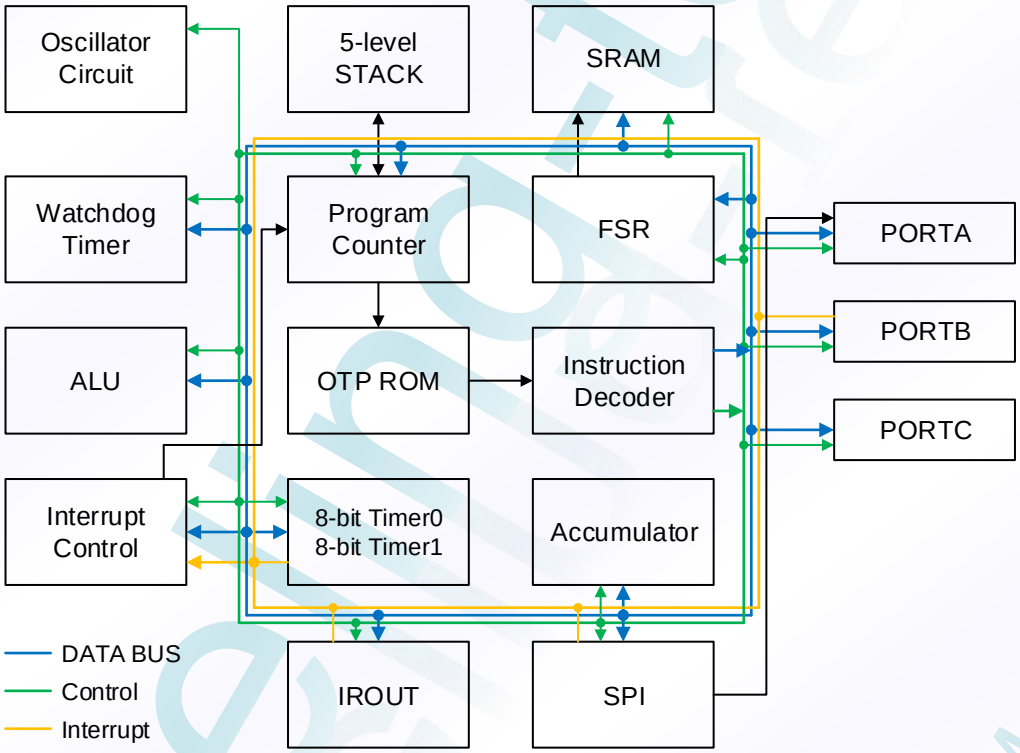
The FM8PE59M is a family of low-cost, high speed, high noise immunity, and OTP-based 8-bit CMOS microcontrollers. It employs a RISC architecture with only 49 instructions. All instructions are single cycle except for program branches which take two cycles. The easy to use and easy to remember instruction set reduces development time significantly.

The FM8PE59M consists of Power-on Reset (POR), Brown-out Reset (BOR), Power-up Reset Timer (PWRT), Oscillator Start-up Timer(OST), Watchdog Timer, OTP, SRAM, tristate I/O port, I/O pull-high/open-drain/pull-down control, Power saving SLEEP mode, 2 real time programmable clock/counter, Interrupt, IROUT, SPI, Wake-up from SLEEP mode, and Code Protection for OTP products. There are six oscillator configurations to choose from, including the power-saving LF (Low Frequency) oscillator and cost saving RC oscillator.

The FM8PE59M address 4K of program memory.

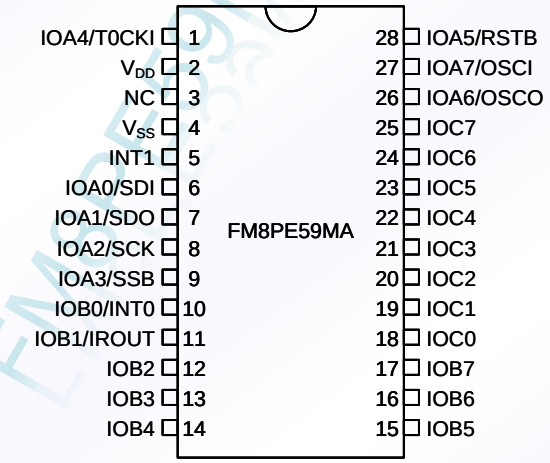
The FM8PE59M can directly or indirectly address its register files and data memory. All special function registers including the program counter are mapped in the data memory.

BLOCK DIAGRAM

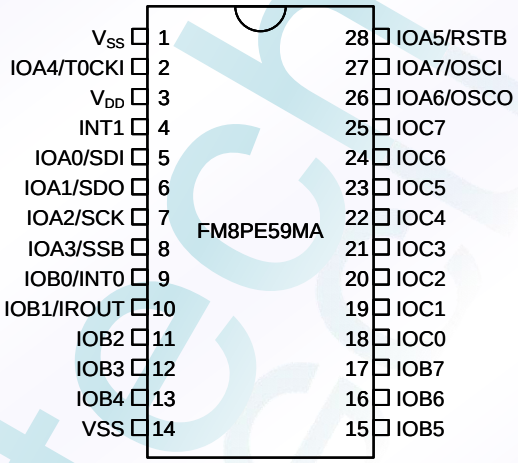


PIN CONNECTION

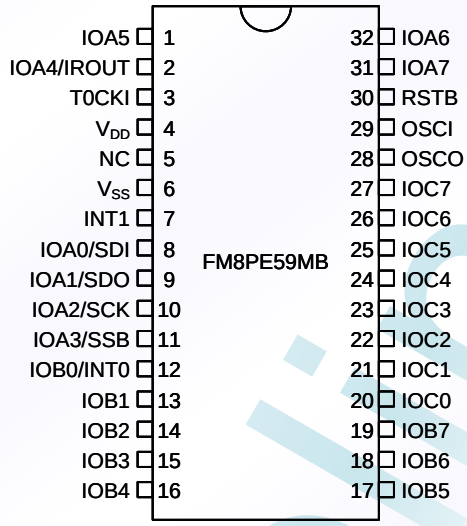
PDIP, SOP



SSOP



PDIP



PIN DESCRIPTIONS

FM8PE59MA

Name	I/O	Description
IOA0 ~ IOA7	I/O	• IOA0 ~ IOA4, IOA6 ~ IOA7 as bi-direction I/O port. • IOA5 is input pin or open-drain output pin, **Voltage on IOA5 pin must not exceed V_{DD}**, See IOA5 diagram for detail description.
IOB0 ~ IOB7	I/O	Bi-direction I/O port with system wake-up function.
IOC0 ~ IOC7	I/O	Bi-direction I/O port.
INT0	I	External interrupt input 0.
INT1	I	External interrupt input 1 triggered by falling edge, Internal weak pull-high.
SDI	I	Serial data in for SPI.
SDO	O	Serial data out for SPI.
SCK	I/O	Serial clock for SPI.
SSB	I	Slave select (active low) for SPI.
IROUT	O	IR output pin.
T0CKI	I	Clock input to Timer0. Must be tied to V_{SS} or V_{DD}, if not in use, to reduce current consumption.
RSTB	I	System clear (RESET) input. This pin is an active low RESET to the device.
OSCI	I	• X'tal type: Oscillator crystal input. • RC type: Clock input of RC oscillator.
OSCO	O	• X'tal type: Oscillator crystal output. • RC mode: Outputs with the instruction cycle rate.
V_{DD}	-	Positive supply.
V_{SS}	-	Ground.

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

FM8PE59MB

Name	I/O	Description
IOA0 ~ IOA7	I/O	Bi-direction I/O port.
IOB0 ~ IOB7	I/O	Bi-direction I/O port with system wake-up function.
IOC0 ~ IOC7	I/O	Bi-direction I/O port.
INT0	I	External interrupt input 0.
INT1	I	External interrupt input 1 triggered by falling edge, Internal weak pull-high.
SDI	I	Serial data in for SPI.
SDO	O	Serial data out for SPI.
SCK	I/O	Serial clock for SPI.
SSB	I	Slave select (active low) for SPI.
IROUT	O	IR output pin.
T0CKI	I	Clock input to Timer0. Must be tied to V_{SS} or V_{DD}, if not in use, to reduce current consumption.
RSTB	I	System clear (RESET) input. This pin is an active low RESET to the device.
OSCI	I	• X'tal type: Oscillator crystal input. • RC type: Clock input of RC oscillator.
OSCO	O	• X'tal type: Oscillator crystal output. • RC mode: Outputs with the instruction cycle rate.
V_{DD}	-	Positive supply.
V_{SS}	-	Ground.

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

Note: Please refer to 2.2 for detail IO type description

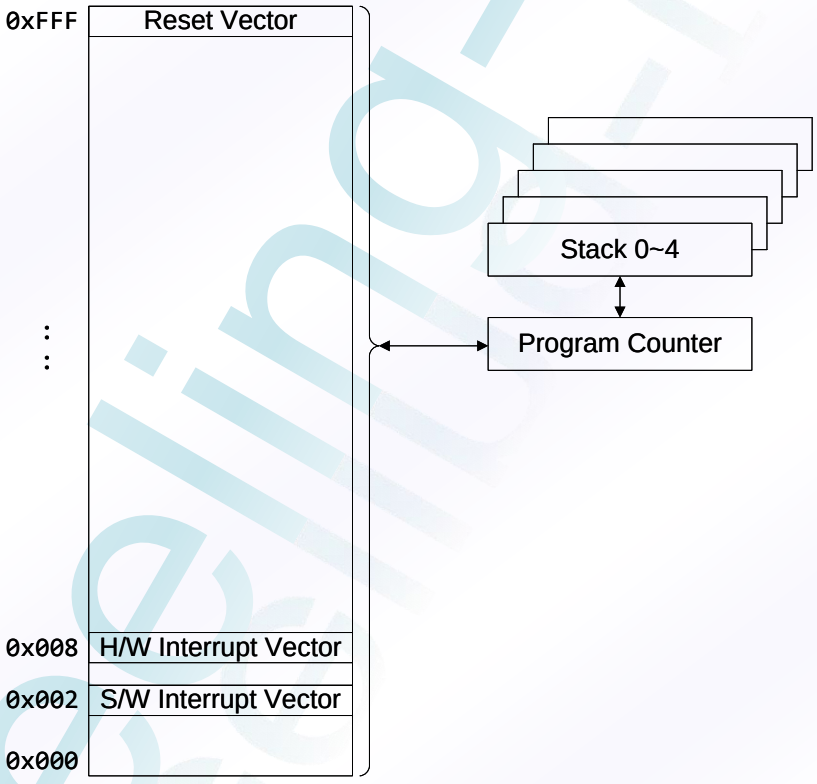
1.0 MEMORY ORGANIZATION

FM8PE59M memory is organized into program memory and data memory.

1.1 Program Memory Organization

The FM8PE59M has a 12-bit Program Counter capable of addressing a 4K program memory space.
The RESET vector for the FM8PE59M is at 0xFFF.
The H/W interrupt vector is at 0x008. And the S/W interrupt vector is at 0x002.
FM8PE59M has program memory size greater than 1K words, but the CALL and GOTO instructions only have a 10-bit address range. This 10-bit address range allows a branch within a 1K program memory page size. To allow CALL and GOTO instructions to address the entire 4K program memory address range for FM8PE59M, there is another two bits to specify the program memory page. This paging bit comes from the PCHBUF<3:2> bits. When doing a CALL or GOTO instruction, the user must ensure that page bit PCHBUF<3:2> are programmed so that the desired program memory page is addressed. When one of the return instructions is executed, the entire 12-bit PC is POPed from the stack. Therefore, manipulation of the PCHBUF<3:2> is not required for the return instructions.
User can use “PAGE” instruction to change memory page and maintains the program memory page. Otherwise, user can use “FCALL (far call)/FGOTO (far goto)” instructions to program user's code.

Figure 1.1: Program Memory Map and STACK



1.2 Data Memory Organization

Data memory is composed of Special Function Registers and General Purpose Registers.
The General Purpose Registers are accessed either directly or indirectly through the FSR register.
The Special Function Registers are registers used by the CPU and peripheral functions to control the operation of the device.
In FM8PE59M, the data memory is partitioned into four banks. Switching between these banks requires the RP1 and RP0 bits in the FSR register to be configured for the desired bank. User can use “BANK” instruction to change the data memory bank.

Table 1.1: Registers File Map for FM8PE59M

FSR<7:6> / Address	Description					
	0 0 Bank 0	0 1 Bank 1	1 0 Bank 2	1 1 Bank 3		
0x00	INDF	Memory back to address in Bank 0				
0x01	TMR0					
0x02	PCL				N/A	OPTION
0x03	STATUS					
0x04	FSR					
0x05	PORTA				0x05	IOSTA
0x06	PORTB				0x06	IOSTB
0x07	PORTC				0x07	IOSTC
0x08	PCON					
0x09	WUCON					
0x0A	PCHBUF					
0x0B	PDCON	T1CON*	PDCON	SPIRXB*		
0x0C	BPHCON	TMR1*	BPHCON	SPITXB*	0x0C	IRCON
0x0D	CPHCON	PR1*	CPHCON	SPISTAT*	0x0D	IRCYCLE
0x0E	INTEN	-*	INTEN	SPICON*	0x0E	IRDUTY
0x0F	INTFLAG	Memory back to address in Bank 0			0x0F	IRCPR
0x10 | 0x1F	General Purpose Registers					
0x20 | 0x3F	General Purpose Registers	General Purpose Registers	General Purpose Registers	General Purpose Registers		

*: Valid only when RBANK = Enable (Configurations bit); if RBANK= Disable, these registers are all memory map back to address in BANK 0.

Table 1.2: The Registers Controlled by OPTION / OPTIONR / IOST / IOSTR Instructions

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
N/A (r/w)	OPTION	*	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
0x05 (r/w)	IOSTA	Port A I/O Control Register							
0x06 (r/w)	IOSTB	Port B I/O Control Register							
0x07 (r/w)	IOSTC	Port C I/O Control Register							
0x0C (r/w)	IRCON	IREN	IROEN	IRCEN	IRSC	-	-	IRPS1	IRPS0
0x0D (r/w)	IRCYCLE	IRC7	IRC6	IRC5	IRC4	IRC3	IRC2	IRC1	IRC0
0x0E (r/w)	IRDUTY	IRD7	IRD6	IRD5	IRD4	IRD3	IRD2	IRD1	IRD0
0x0F (r/w)	IRCPR	IRCPR7	IRCPR6	IRCPR5	IRCPR4	IRCPR3	IRCPR2	IRCPR1	IRCPR0

Legend: - = unimplemented, read as '0', * = unimplemented, read as '1',

Table 1.3: Operational Registers Map

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
Unbanked									
0x00 (r/w)	INDF	Uses contents of FSR to address data memory (not a physical register)							
0x01 (r/w)	TMR0	8-bit real-time clock/counter							
0x02 (r/w)	PCL	Low order 8 bits of PC							
0x03 (r/w)	STATUS	GP2	GP1	GP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x04 (r/w)	FSR	RP1	RP0	Indirect data memory address pointer					
0x05 (r/w)	PORTA	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
0x06 (r/w)	PORTB	IOB7	IOB6	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0
0x07 (r/w)	PORTC	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0
0x08 (r/w)	PCON	WDTE	EIS	LVDTE	ROC	-	-	ODC67	/WUC45
0x09 (r/w)	WUCON	/WUB7	/WUB6	/WUB5	/WUB4	/WUB3	/WUB2	/WUB1	/WUB0
0x0A (r/w)	PCHBUF	-	-	-	-	Upper 4 MSBs Buffer of PC			
Bank 0, 2									
0x0B (r/w)	PDCON	/PDB3	/PDB2	/PDB1	/PDB0	/PDA3	/PDA2	/PDA1	/PDA0
0x0C (r/w)	BPHCON	/PHB7	/PHB6	/PHB5	/PHB4	/PHB3	/PHB2	/PHB1	/PHB0
0x0D (r/w)	CPHCON	/PHC7	/PHC6	/PHC5	/PHC4	/PHC3	/PHC2	/PHC1	/PHC0
0x0E (r/w)	INTEN	GIE	SPIIE	IRIE	LVDTIE	INT1IE	INT0IE	T1IE	T0IE
Bank 1									
0x0B (r/w)	T1CON	-	-	-	-	-	T1ON	T1P1	T1P0
0x0C (r/w)	TMR1	TMR17	TMR16	TMR15	TMR14	TMR13	TMR12	TMR11	TMR10
0x0D (r/w)	PR1	PR17	PR16	PR15	PR14	PR13	PR12	PR11	PR10
0x0E (r/w)	-	Unimplemented, read as "0"s							
Bank 3									
0x0B (r)	SPIRXB	SPI Receive buffer							
0x0C (r/w)	SPITXB	SPI Transmitter buffer							
0x0D (r/w)	SPISTAT	DORD	SDOS	-	-	SDOOD	SCKOD	-	RXBF
0x0E (r/w)	SPICON	CKEDG	SPION	RXOV	SSE	-	SPIM2	SPIM1	SPIM0
Unbanked									
0x0F (r/w)	INTFLAG	-	SPIIF	IRIF	LVDTIF	INT1IF	INT0IF	T1IF	T0IF

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
0x00	INDF	Uses contents of FSR to address data memory (not a physical register)							

Legend: x = unknown, more bits' default state, please refer to Table 2.6.

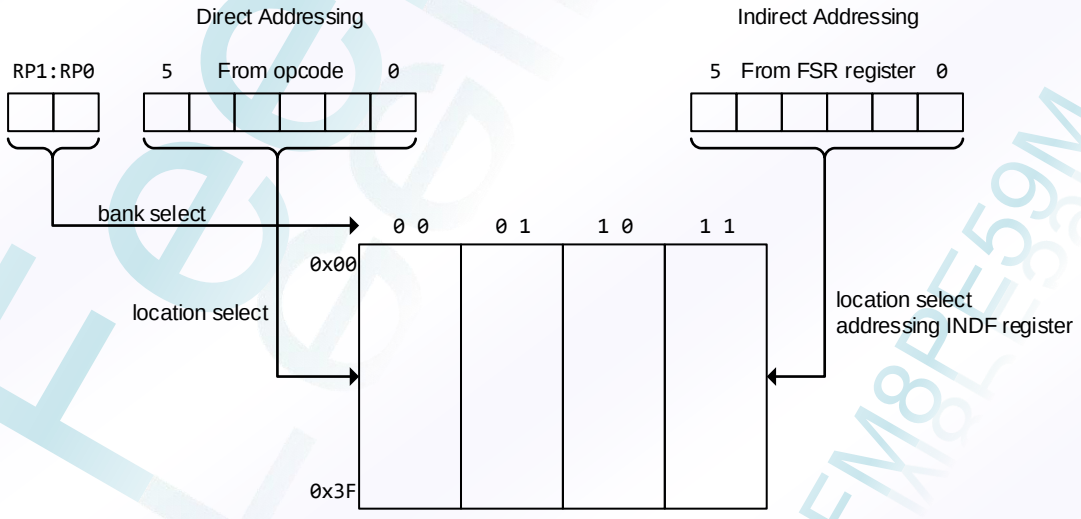
The INDF Register is not a physical register. Any instruction accessing the INDF register can actually access the register pointed by FSR Register. Reading the INDF register itself indirectly (FSR="0x00") will read 0x00. Writing to the INDF register indirectly results in a no-operation (although status bits may be affected).
The bits 5-0 of FSR register are used to select up to 64 registers (address: 0x00 ~ 0x3F).
In FM8PE59M, the data memory is partitioned into four banks. Switching between these banks requires the RP1 and RP0 bits in the FSR register to be configured for the desired bank. The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers. All Special Function Registers and some of General Purpose Registers from other banks are mirrored in bank 0 for code reduction and quicker access.

RP1:RP0		Accessed Bank
0	0	0
0	1	1
1	0	2
1	1	3

Example 2.1: INDIRECT ADDRESSING

- Register file 0x38 contains the value 0x10
- Register file 0x39 contains the value 0x0A
- Load the value 0x38 into the FSR Register
- A read of the INDF Register will return the value of 0x10
- Increment the value of the FSR Register by one (@FSR=0x39)
- A read of the INDF register now will return the value of 0x0A.

Figure 2.1: Direct/Indirect Addressing for FM8PE59M



2.1.2 TMR0 (Time Clock/Counter 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
0x01	TMR0	8-bit real-time clock/counter							

Note: more bits' default state, please refer to Table 2.6.

The Timer0 is a 8-bit timer/counter. The clock source of Timer0 can come from the instruction cycle clock or by an external clock source (T0CKI pin) defined by T0CS bit (OPTION<5>). If T0CKI pin is selected, the Timer0 is increased by T0CKI signal rising/falling edge (selected by T0SE bit (OPTION<4>)).
The pre-scaler is assigned to Timer0 by clearing the PSA bit (OPTION<3>). In this case, the pre-scaler will be cleared when TMR0 register is written with a value.

2.1.3 PCL (Low Bytes of Program Counter) & Stack

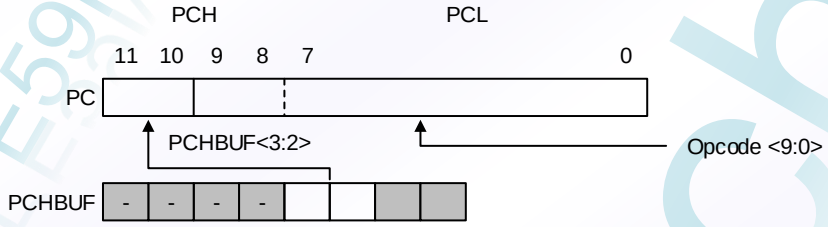
Read/Write-POR		R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x02	PCL	Low order 8 bits of PC							

Note: more bits' default state, please refer to Table 2.6.

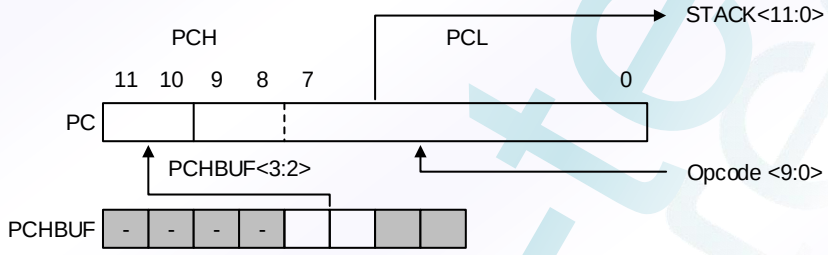
FM8PE59M devices have a 12-bit wide Program Counter (PC) and five-level deep 12-bit hardware push/pop stack. The low byte of PC is called the PCL register. This register is readable and writable. The high byte of PC is called the PCH register. This register contains the PC<11:8> bits and is not directly 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 GOTO instruction, the PC<9:0> is provided by the GOTO instruction word. The PC<11:10> is updated from the PCHBUF<3:2>. The PCL register is mapped to PC<7:0>, and the PCHBUF register is not updated.
For a CALL instruction, the PC<9:0> is provided by the CALL instruction word. The PC<11:10> is updated from the PCHBUF<3:2>. The next PC will be loaded (PUSHed) onto the top of STACK. The PCL register is mapped to PC<7:0>, and the PCHBUF register is not updated.
For a FGOTO instruction, the PC<11:0> is provided by the FGOTO instruction word. The PCL register is mapped to PC<7:0>, the PCHBUF<3:2> bits is also updated from the FGOTO instruction word, and the PCHBUF<1:0> bits are not updated.
For a FCALL instruction, the PC<11:0> is provided by the FCALL instruction word. The next PC will be loaded (PUSHed) onto the top of STACK. The PCL register is mapped to PC<7:0>, the PCHBUF<3:2> bits is also updated from the FCALL instruction word, and the PCHBUF<1:0> bits are not updated.
For a RETIA, RETFIE, or RETURN 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 any instruction where the PCL is the destination (excluding TBL instruction), the PC<7:0> is provided by the instruction word or ALU result. However, the PC<11:8> will come from the PCHBUF<3:0> bits (PCHBUF → PCH).
For TBL instruction, the PC<7:0> is provided by the ALU result, and the PC<9:8> are not changed. The PC<11:10> will come from the PCH<3:2> bits.
PCHBUF register is never updated with the contents of PCH.

Figure 2.2: Loading of PC in Different Situations

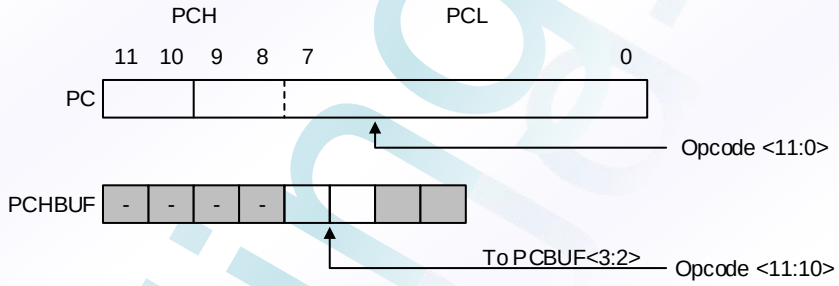
Situation 1: **GOTO** Instruction



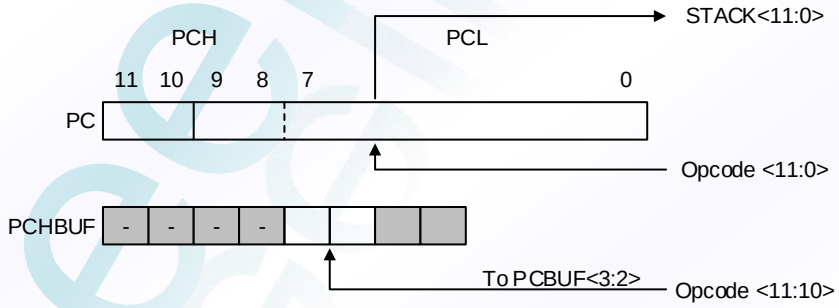
Situation 2: **CALL** Instruction



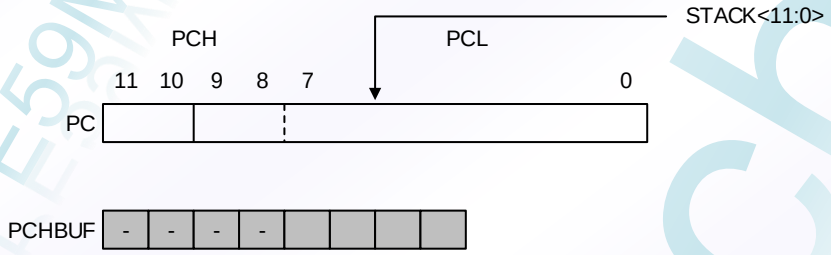
Situation 3: **FGOTO** Instruction



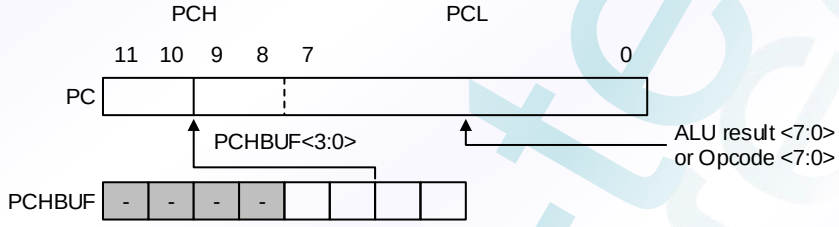
Situation 4: **FCALL** Instruction



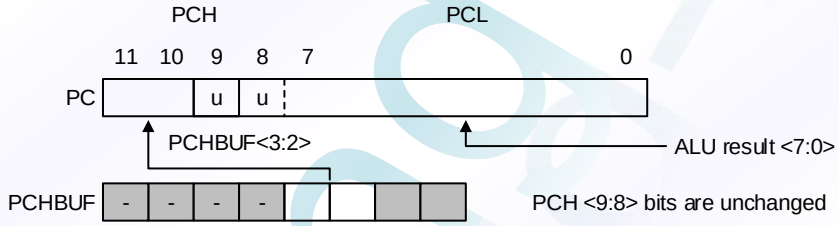
Situation 5: **RETIA**, **RETFIE**, or **RETURN** Instruction



Situation 6: Instruction with PCL as destination (except TBL instruction)



Situation 7: **TBL** instruction



Note: PCHBUF is used only for instruction with PCL as destination, GOTO and CALL instructions.

2.1.4 STATUS (Status Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R-#	R-#	R/W-x	R/W-x	R/W-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x03	STATUS	GP2	GP1	GP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C

Note: # = refer Table 2.7 for detail description, more bits' default state, please refer to Table 2.6.

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, ADDIA, ADCAR:
= 0, No Carry occurred.
= 1, Carry occurred.
SUBAR, SUBIA, SBCAR:
= 0, Borrow occurred.
= 1, No 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, ADDIA, ADCAR:
= 0, No Carry from the 4th low order bit of the result occurred.
= 1, Carry from the 4th low order bit of the result occurred.
SUBAR, SUBIA, SBCAR:
= 0, Borrow from the 4th low order bit of the result occurred.
= 1, No Borrow from the 4th low order bit of the result occurred.

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

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

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

GP2:GP0: General purpose read/write bits.

2.1.5 FSR (Indirect Data Memory Address Pointer)

Read/Write-POR		R/W-0	R/W-0	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
0x04	FSR	RP1	RP0	Indirect data memory address pointer					

Legend: x = unknown, more bits' default state, please refer to Table 2.6.

Bit5:Bit0: Select registers address in the indirect addressing mode. See 2.1.1 for detail description.

RP1:RP0: These bits are used to switching the bank of four data memory banks. User can use "BANK" instruction to change bank. See 2.1.1 for detail description.

2.1.6 PORTA, PORTB, PORTC (Port Data Registers)

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
0x05	PORTA	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0

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
0x06	PORTB	IOB7	IOB6	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0

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
0x07	PORTC	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0

Legend: x = unknown, more bits' default state, please refer to Table 2.6.

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.
All of PORTA, PORTB and PORTC are 8-bit port data registers.

2.1.7 PCON (Power Control Register)

Read/Write-POR		R/W-1	R/W-0	R/W-1	R/W-0	-	-	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x08	PCON	WDTE	EIS	LVDTE	ROC	-	-	ODC67	/WUC45

Legend: - = unimplemented, read as '0'; more bits' default state, please refer to Table 2.6.

/WUC45: = 0, Enable the input falling wake-up function of IOC4 and IOC5 pins.
= 1, Disable the input falling wake-up function of IOC4 and IOC5 pins.

ODC67: = 0, Disable the internal open-drain of IOC6 and IOC7 pins.
= 1, Enable the internal open-drain of IOC6 and IOC7 pins.

ROC: R-option function of IOC0 and IOC1 pins enable bit.
=0, Disable the R-option function.
=1, Enable the R-option function. In this case, if a 430KΩ external resister is connected/disconnected to V_{SS}, the status of IOC0 (IOC1) is read as "0"/"1".

LVDTE: LVDT (low voltage detector) enable bit.
= 0, Disable LVDT.
= 1, Enable LVDT.

EIS: Define the function of IOB0/INT pin.
= 0, IOB0 (bi-directional I/O pin) is selected. The path of INT0 is masked.
= 1, INT0 (external interrupt pin) is selected. In this case, the I/O control bit of IOB0 must be set to "1". The path of Port B input change of IOB0 pin is masked by hardware, the status of INT0 pin can also be read by way of reading PORTB.

WDTE: WDT (watch-dog timer) enable bit.
= 0, Disable WDT.
= 1, Enable WDT.

2.1.8 WUCON (Port B Input Falling Wake-up 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
0x09	WUCON	/WUB7	/WUB6	/WUB5	/WUB4	/WUB3	/WUB2	/WUB1	/WUB0

Note: more bits' default state, please refer to Table 2.6.

/WUB0: = 0, Enable the input falling wake-up function of IOB0 pin.
= 1, Disable the input falling wake-up function of IOB0 pin.

/WUB1: = 0, Enable the input falling wake-up function of IOB1 pin.
= 1, Disable the input falling wake-up function of IOB1 pin.

/WUB2: = 0, Enable the input falling wake-up function of IOB2 pin.
= 1, Disable the input falling wake-up function of IOB2 pin.

/WUB3: = 0, Enable the input falling wake-up function of IOB3 pin.
= 1, Disable the input falling wake-up function of IOB3 pin.

/WUB4: = 0, Enable the input falling wake-up function of IOB4 pin.
= 1, Disable the input falling wake-up function of IOB4 pin.

/WUB5: = 0, Enable the input falling wake-up function of IOB5 pin.
= 1, Disable the input falling wake-up function of IOB5 pin.

/WUB6: = 0, Enable the input falling wake-up function of IOB6 pin.
= 1, Disable the input falling wake-up function of IOB6 pin.

/WUB7: = 0, Enable the input falling wake-up function of IOB7 pin.
= 1, Disable the input falling wake-up function of IOB7 pin.

2.1.9 PCHBUF (High Byte Buffer of Program Counter)

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
0x0A	PCHBUF	-	-	-	-	Upper 4 MSBs Buffer of PC			

Legend: - = unimplemented, read as '0'; more bits' default state, please refer to Table 2.6.

PCHBUF<3:2>: Program memory page select bits.

PCHBUF<3:2>		Program Memory Page [Address]
0	0	Page 0 [0x000~0x3FF]
0	1	Page 1 [0x400~0x7FF]
1	0	Page 2 [0x800~0xBFF]
1	1	Page 3 [0xC00~0xFFF]

User can use "PAGE" instruction to change memory page and maintains the program memory page. Otherwise, user can use "FGOTO" (far goto), or "FCALL" (far call) instructions to program user's code.
See 2.1.3 for detail description.

2.1.10 PDCON (Pull-down Control Register)

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
0x0B	PDCON	/PDB3	/PDB2	/PDB1	/PDB0	/PDA3	/PDA2	/PDA1	/PDA0

Note: more bits' default state, please refer to Table 2.6.

/PDA0: = 0, Enable the internal pull-down of IOA0 pin.
= 1, Disable the internal pull-down of IOA0 pin.

/PDA1: = 0, Enable the internal pull-down of IOA1 pin.
= 1, Disable the internal pull-down of IOA1 pin.

/PDA2: = 0, Enable the internal pull-down of IOA2 pin.
= 1, Disable the internal pull-down of IOA2 pin.

/PDA3: = 0, Enable the internal pull-down of IOA3 pin.
= 1, Disable the internal pull-down of IOA3 pin.

/PDB0: = 0, Enable the internal pull-down of IOB0 pin.
= 1, Disable the internal pull-down of IOB0 pin.

/PDB1: = 0, Enable the internal pull-down of IOB1 pin.
= 1, Disable the internal pull-down of IOB1 pin.

/PDB2: = 0, Enable the internal pull-down of IOB2 pin.
= 1, Disable the internal pull-down of IOB2 pin.

/PDB3: = 0, Enable the internal pull-down of IOB3 pin.
= 1, Disable the internal pull-down of IOB3 pin.

2.1.11 BPHCON (Port B Pull-high Control Register) (Bank 0, 2)

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
0x0C	BPHCON	/PHB7	/PHB6	/PHB5	/PHB4	/PHB3	/PHB2	/PHB1	/PHB0

Note: more bits' default state, please refer to Table 2.6.

/PHB0: = 0, Enable the internal pull-high of IOB0 pin.
= 1, Disable the internal pull-high of IOB0 pin.

/PHB1: = 0, Enable the internal pull-high of IOB1 pin.
= 1, Disable the internal pull-high of IOB1 pin.

/PHB2: = 0, Enable the internal pull-high of IOB2 pin.
= 1, Disable the internal pull-high of IOB2 pin.

/PHB3: = 0, Enable the internal pull-high of IOB3 pin.
= 1, Disable the internal pull-high of IOB3 pin.

/PHB4: = 0, Enable the internal pull-high of IOB4 pin.
= 1, Disable the internal pull-high of IOB4 pin.

/PHB5: = 0, Enable the internal pull-high of IOB5 pin.
= 1, Disable the internal pull-high of IOB5 pin.

/PHB6: = 0, Enable the internal pull-high of IOB6 pin.
= 1, Disable the internal pull-high of IOB6 pin.

/PHB7: = 0, Enable the internal pull-high of IOB7 pin.
= 1, Disable the internal pull-high of IOB7 pin.

2.1.12 CPHCON (Port C Pull-high Control Register) (Bank 0, 2)

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
0x0D	CPHCON	/PHC7	/PHC6	/PHC5	/PHC4	/PHC3	/PHC2	/PHC1	/PHC0

Note: more bits' default state, please refer to Table 2.6.

/PHC0: = 0, Enable the internal pull-high of IOC0 pin.
= 1, Disable the internal pull-high of IOC0 pin.

/PHC1: = 0, Enable the internal pull-high of IOC1 pin.
= 1, Disable the internal pull-high of IOC1 pin.

/PHC2: = 0, Enable the internal pull-high of IOC2 pin.
= 1, Disable the internal pull-high of IOC2 pin.

/PHC3: = 0, Enable the internal pull-high of IOC3 pin.
= 1, Disable the internal pull-high of IOC3 pin.

/PHC4: = 0, Enable the internal pull-high of IOC4 pin.
= 1, Disable the internal pull-high of IOC4 pin.

/PHC5: = 0, Enable the internal pull-high of IOC5 pin.
= 1, Disable the internal pull-high of IOC5 pin.

/PHC6: = 0, Enable the internal pull-high of IOC6 pin.
= 1, Disable the internal pull-high of IOC6 pin.

/PHC7: = 0, Enable the internal pull-high of IOC7 pin.
= 1, Disable the internal pull-high of IOC7 pin.

2.1.13 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
0x0E	INTEN	GIE	SPIIE	IRIE	LVDTIE	INT1IE	INT0IE	T1IE	T0IE

Note: more bits' default state, please refer to Table 2.6.

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

T1IE: Timer1 match interrupt enable bit.
= 0, Disable the Timer1 match interrupt.
= 1, Enable the Timer1 match interrupt.

INT0IE: External INT0 pin interrupt enable bit.
= 0, Disable the External INT0 pin interrupt.
= 1, Enable the External INT0 pin interrupt.

INT1IE: External INT1 pin interrupt enable bit.
= 0, Disable the External INT1 pin interrupt.
= 1, Enable the External INT1 pin interrupt.

LVDTIE: Low-Voltage detector interrupt enable bit.
= 0, Disable the Low-Voltage detector interrupt, LVDT will reset MCU.
= 1, Enable the Low-Voltage detector interrupt, LVDT will not reset MCU.
Note: 1. ***The LVDT interrupt function will be fixed to "Disable" by H/W if the configuration bit IOA5OD = Disable, even if bit LVDTIE = 1 and LVDTE = 1.***
2. The Detector voltage selected by configuration LVDT bit.
For detail description of the LVDT interrupt, see 2.7.7 section.

IRIE: IROUT counter match interrupt enable bit.
= 0, Disable the IROUT counter match interrupt.
= 1, Enable the IROUT counter match interrupt.

SPIIE: SPI module interrupt enable bit.
= 0, Disable the SPI module interrupt.
= 1, Enable the SPI module interrupt.

GIE: Global interrupt enable bit.
= 0, Disable all interrupts.
= 1, Enable all un-masked interrupts.
Note: When an interrupt event occurs 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. The RETFIE instruction will exit the interrupt routine and set the GIE bit to re-enable interrupt.

2.1.14 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
0x0F	INTFLAG	-	SPIIF	IRIF	LVDTIF	INT1IF	INT0IF	T1IF	T0IF

Legend: - = unimplemented, read as '0'; more bits' default state, please refer to Table 2.6.

T0IF: Timer0 overflow interrupt flag. Set when Timer0 overflows, reset by software.

T1IF: Timer1 match interrupt flag. Set when TMR1 register matches to PR1 register, reset by software.

INT0IF: External INT0 pin interrupt flag. Set by rising/falling (selected by INTEDG bit (OPTION<6>)) edge on INT0 pin, reset by software.

INT1IF: External INT1 pin interrupt flag. Set by falling edge on INT1 pin, reset by software.

LVDTIF: Low-voltage detector interrupt flag. Set when Low-Voltage was detected, reset by software.

IRIF: IR counter match interrupt flag. Set when IROUT counter matches to IRCPR register, reset by software.

SPIIF: SPI module interrupt flag. Set after one byte of SPI transmission is completed, reset by software.

2.1.15 T1CON (Timer 1 Control Register) (Bank 1)

Read/Write-POR		-	-	-	-	-	R/W-1	R/W-1	R/W-1
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0B	T1CON	-	-	-	-	-	T1ON	T1P1	T1P0

Legend: - = unimplemented, read as '0'; more bits' default state, please refer to Table 2.6.

T1P1:T1P0: Timer 1 pre-scaler select bits.

T1P1:T1P0		Pre-scaler Rate
0	0	1 : 1
0	1	1 : 4
1	0	1 : 8
1	1	1 : 16

T1ON: Timer 1 module enable bit.
= 0, Disable the Timer 1 module.
= 1, Enable the Timer 1 module.

2.1.16 TMR1 (Timer 1 Register) (Bank 1)

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
0x0C	TMR1	TMR17	TMR16	TMR15	TMR14	TMR13	TMR12	TMR11	TMR10

Note: more bits' default state, please refer to Table 2.6.

TMR17:TMR10: Timer 1 register and increase until the value matches to PR1 register, and then reset to "0".

2.1.17 PR1 (Timer 1 Pulse-width Register) (Bank 1)

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
0x0D	PR1	PR17	PR16	PR15	PR14	PR13	PR12	PR11	PR10

Note: more bits' default state, please refer to Table 2.6.

PR17:PR10: Timer 1 period register.

2.1.18 SPIRXB (SPI Receive Buffer Register) (Bank 3)

Read/Write-POR		R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0B	SPIRXB	SPI Receive buffer							

Legend: x = unknown, more bits' default state, please refer to Table 2.6.

SPI receives data buffer. Once the 8-bits data have been received, the data in SPI shift register (SPISR) will be moved to the SPIRXB register.
The data must be read out before the next 8-bits data reception is completed if needed.
The RXBF flag is set when the data in SPISR is moved to the SPIRXB register, and cleared as the SPIRXB register reads.

2.1.19 SPITXB (SPI Transmit Buffer Register) (Bank 3)

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
0x0C	SPITXB	SPI Transmitter buffer							

Legend: x = unknown, more bits' default state, please refer to Table 2.6.

SPI transmits data buffer. Once the first valid clock pulse appears on SCK pin, the data in SPITXB will be loaded into SPISR and start to shift in/out.
The new data must be written to SPITXB before the 8-bits data transmission is completed if needed.

2.1.20 SPISTAT (SPI Status Register) (Bank 3)

Read/Write-POR		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
0x0D	SPISTAT	DORD	SDOS	-	-	SDOOD	SCKOD	-	RXBF

Legend: - = unimplemented, read as '0'; more bits' default state, please refer to Table 2.6.

RXBF: SPI receive buffer full flag. Set when the data in SPISR is moved to the SPIRXB register, reset by software or by reading SPIRXB register.
= 0, Receive not complete, SPIRXB is empty.
= 1, Receive complete, SPIRXB is full.

SCKOD: Open-drain control bit for SCK pin output
= 0, Open-drain disable.
= 1, Open-drain enable.

SDOOD: Open-drain control bit for SDO pin output
= 0, Open-drain disable.
= 1, Open-drain enable.

SDOS: SDO output status control bit while SSB = 1 for slave mode with SSB control enabled.
= 0, Disable, the SDO will be floating.
= 1, Enable, the SDO will remain low.

DORD: SPI data transmission order.
= 0, Data shift out MSB first.
= 1, Data shift out LSB first.

2.1.21 SPICON (SPI Control Register) (Bank 3)

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
0x0E	SPICON	CKEDG	SPION	RXOV	SSE	-	SPIM2	SPIM1	SPIM0

Legend: - = unimplemented, read as '0'; more bits' default state, please refer to Table 2.6.

SPIM2:SPIM0: SPI mode select bits.

SPIM2:SPIM0			SPI Mode
0	0	0	SPI master mode, clock = $F_{OSC}/2$
0	0	1	SPI master mode, clock = $F_{OSC}/4$
0	1	0	SPI master mode, clock = $F_{OSC}/8$
0	1	1	SPI master mode, clock = $F_{OSC}/16$
1	0	0	SPI master mode, clock = $F_{OSC}/32$
1	0	1	SPI slave mode, clock = SCK pin, SSB pin control enabled
1	1	0	SPI slave mode, clock = SCK pin, SSB pin control disabled
1	1	1	SPI master mode, clock = Timer1 output/2

SSE: SPI shift register enable bit
= 0, Reset by hardware as soon as the shifting is complete.
= 1, Start to transmit/receive, and keep on "1" while the current byte is still being transmitted/received.

RXOV: SPI receive buffer overflow bit (only in slave mode)
=0, Not overflow.
=1, A new byte is received while the SPIRXB register is still holding the previous data. In this case, the data in SPISR register will be ignored and lost.

SPION: SPI module enable bit
= 0, Disable SPI module.
= 1, Enable SPI module.

CKEDG: Clock edge select bit
= 0, Data shifts out on rising edge of SCK, and shifts in on falling edge of SCK.
= 1, Data shifts out on falling edge of SCK, and shifts in on rising edge of SCK.

2.1.22 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 2.6.

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

2.1.23 OPTION Register

Read/Write-POR		*	R/W-0	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
N/A	OPTION	*	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

Accessed by OPTION / OPTIONR instruction.
Legend: * = unimplemented, read as '1'; more bits' default state, please refer to Table 2.6.

By executing the OPTION instruction, the contents of the ACC Register will be transferred to the OPTION Register.
By executing the OPTIONR instruction, user can read this register into ACC.
The OPTION Register is a 7-bit wide register which contains various control bits to configure the Timer0/WDT pre-scaler, Timer0, and the external INT interrupt.
The OPTION Register are set all "1"s except INTEDG bit.

PS2:PS0: Pre-scaler rate select bits.

PS2:PS0			Timer0 Rate	WDT Rate
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

PSA: Pre-scaler assign bit.
= 0, TMR0 (Timer0).
= 1, WDT (watch-dog timer).

T0SE: TMR0 source edge select bit.
= 0, Rising edge on T0CKI pin.
= 1, Falling edge on T0CKI pin.

T0CS: TMR0 clock source select bit.
= 0, internal instruction clock cycle.
= 1, External T0CKI pin.

INTEDG: INT0 pin interrupt edge select bit.
= 0, interrupt on falling edge of INT0 pin.
= 1, interrupt on rising edge of INT0 pin.

2.1.24 IOSTA, IOSTB & IOSTC (Port I/O 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
0x05	IOSTA	IOSTA7	IOSTA6	IOSTA5	IOSTA4	IOSTA3	IOSTA2	IOSTA1	IOSTA0

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
0x06	IOSTB	IOSTB7	IOSTB6	IOSTB5	IOSTB4	IOSTB3	IOSTB2	IOSTB1	IOSTB0

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
0x07	IOSTC	IOSTC7	IOSTC6	IOSTC5	IOSTC4	IOSTC3	IOSTC2	IOSTC1	IOSTC0

Accessed by IOST / IOSTR instruction.
Note: more bits' default state, please refer to Table 2.6.

The Port I/O Control Registers are loaded with the contents of the ACC Register by executing the IOST R (0x05~0x07) instruction. By executing the IOSTR instruction, user can read these registers into ACC.
The IOST Registers are set (output drivers disabled) upon RESET.

IOSTA7:IOSTA0: PORTA I/O control bit.
= 0, PORTA pin configured as an output.
= 1, PORTA pin configured as an input (tristate).
Note: 1. IOA5 is open-drain output only if IOSTA5 = 0.
2. The IOA5 open-drain function will be fixed to "Disable" by H/W if the configuration bit IOA5OD= Disable, even if bit IOSTA5 = 0.
3. The IOA5 open-drain function only for A-type.

IOSTB7:IOSTB0: PORTB I/O control bit.
= 0, PORTB pin configured as an output.
= 1, PORTB pin configured as an input (tristate).

IOSTC7:IOSTC0: PORTC I/O control bit.
= 0, PORTC pin configured as an output.
= 1, PORTC pin configured as an input (tristate).

2.1.25 IRCON (IROUT Control Register)

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
0x0C	IRCON	IREN	IROEN	IRCEN	IRSC	-	-	IRPS1	IRPS0

Accessed by IOST / IOSTR instruction.
Legend: - = unimplemented, read as '0'; more bits' default state, please refer to Table 2.6.

IREN: IOA4/IROUT pin select bit.
= 0, IOA4 is selected and IR module is disabled.
= 1, IROUT is selected and IR module is enabled.

IROEN: IROUT output enable bit.
= 0, IROUT is disabled.
= 1, IROUT is enabled.

IRCEN: IROUT counter enable bit.
= 0, IROUT counter is disabled and be reset to "0".
= 1, IROUT counter is enabled and start to count.

IRSC: IROUT pin drive/sink current select bit.
= 0, Normal.
= 1, Heavy.

IRPS1:IRPS0: IR module clock source pre-scaler select bits.

IRPS1:IRPS0		IR Module Clock Source Frequency
0	0	Oscillator Frequency / 1
0	1	Oscillator Frequency / 2
1	0	Oscillator Frequency / 4
1	1	Oscillator Frequency / 8

2.1.26 IRCYCLE (IROUT Cycle Control Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0D	IRCYCLE	IRC7	IRC6	IRC5	IRC4	IRC3	IRC2	IRC1	IRC0

Accessed by IOST / IOSTR instruction.
Note: more bits' default state, please refer to Table 2.6.

IRC7:IRC0: IROUT (IR Carrier output) frequency = (IR clock source frequency) / (IRC7:IRC0).

2.1.27 IRDUTY (IROUT Duty Control Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0E	IRDUTY	IRD7	IRD6	IRD5	IRD4	IRD3	IRD2	IRD1	IRD0

Accessed by IOST / IOSTR instruction.
Note: more bits' default state, please refer to Table 2.6.

IRD7:IRD0: IROUT (IR Carrier output) duty cycle = (IRD7:IRD0) / (IRC7:IRC0).
(IRD7:IRD0) must be less than (IRC7:IRC0).

2.1.28 IRCPR (IROUT Counter Pre-set 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
0x0F	IRCPR	IRCPR7	IRCPR6	IRCPR5	IRCPR4	IRCPR3	IRCPR2	IRCPR1	IRCPR0

Accessed by IOST / IOSTR instruction.
Note: more bits' default state, please refer to Table 2.6.

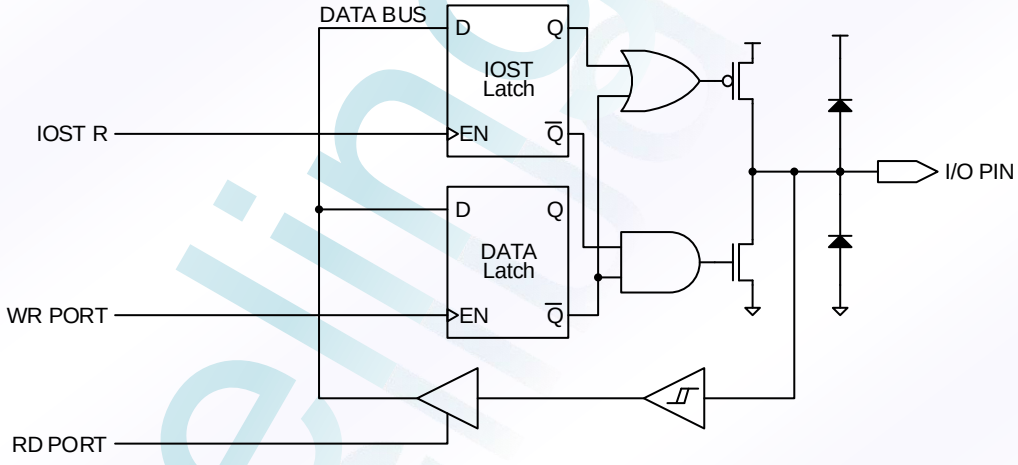
IRCPR7:IRCPR0: IROUT counter pre-set bits. IROUT counter increase on every leading edge of internal IR pulse until the value of IR counter matches to IRCPR register, and then the IR counter will be reset to "0", set the IRIF interrupt flag, and increase again.
Note: IROUT counter period = ((IRCPR7:IRCPR0) + 1) x (IR Carrier output frequency)

2.2 I/O Ports

Port A, port B and port C are bi-directional tristate I/O ports. All of Port A, Port B and port C are 8-pin I/O ports.
All I/O pins (IOA<7:0>, IOB<7:0> and IOC<7:0>) have data direction control registers (IOSTA, IOSTB, IOSTC) which can configure these pins as output or input.
IOB<7:0> and IOC<7:0> have its corresponding pull-high control bits (BPHCON and CPHCON registers) to enable the weak internal pull-high. The weak pull-high is automatically turned off when the pin is configured as an output pin.
IOA<3:0> and IOB<3:0> have its corresponding pull-down control bits (PDCON register) to enable the weak internal pull-down. The weak pull-down is automatically turned off when the pin is configured as an output pin.
IOC<7:6> have its corresponding open-drain control bit (ODC67 bit (PCON<1>)) to enable the open-drain output when these pins are configured to be an output pin.
IOC0 and IOC1 are the R-option pins enabled by setting the ROC bit (PCON<4>). When the R-option function is used, it is recommended that IOA0 and IOA1 are used as output pins, and read the status of IOA0 and IOA1 before these pins are configured to be an output pin.
IOB<7:0> and IOC<5:4> also provide the input falling or low level wake-up function. Each pin has its corresponding input wake-up enable bits (WUCON register and /WUC45 bit (PCON<0>)) to select the input falling or low level wake-up source. Falling or low level wake-up function can be selected by WUOPT bit of Configuration word.
The IOB0 is also an external interrupt input signal by setting the EIS bit (PCON<6>). In this case, IOB0 input falling wake-up function will be disabled by hardware even if it is enabled by software.

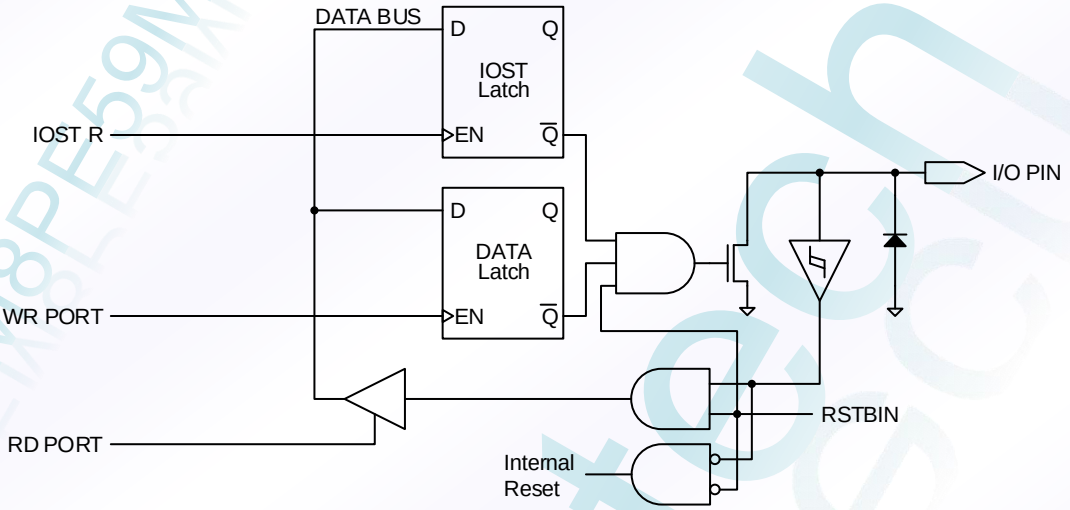
Figure 2.3: Block Diagram of I/O Pins

IOA7 ~ IOA6, IOA4 ~ IOA0, IOC7 ~ IOC6, IOC3 ~ IOC0:
IOA5 (for FM8PE59MB):



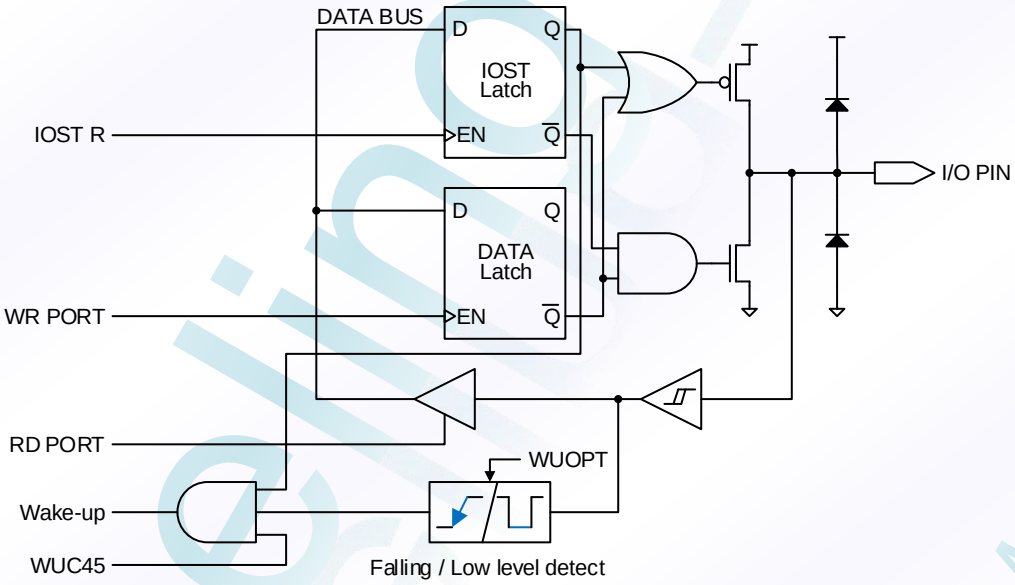
Pull-down and open-drain are not shown in the figure

IOA5 (for FM8PE59MA):



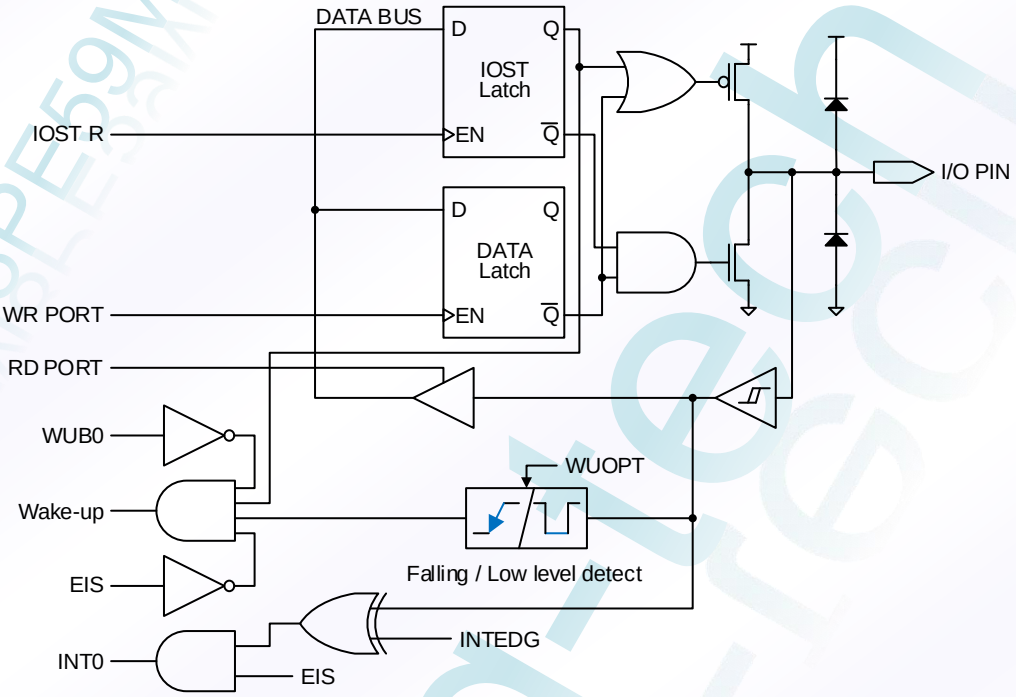
Voltage on this pin must not exceed V_{DD}.

IOC5 ~ IOC4:



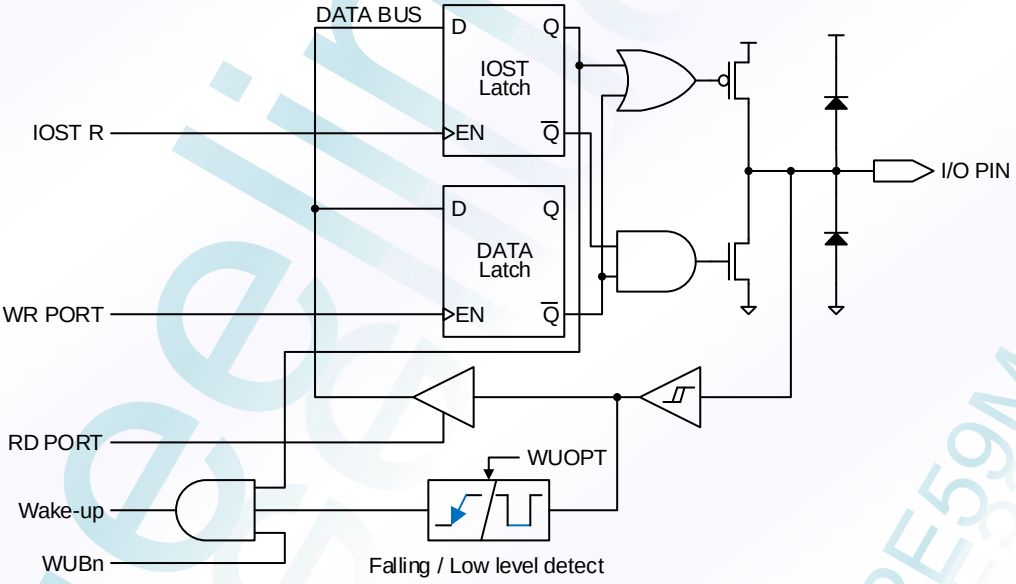
Pull-high are not shown in the figure

IOB0:



Pull-high/pull-down are not shown in the figure

IOB7 ~ IOB1:



Pull-high/pull-down are not shown in the figure

2.3 Timer0/WDT & Pre-scaler

2.3.1 Timer0

The Timer0 is an 8-bit timer/counter. The clock source of Timer0 can come from the internal clock or by an external clock source (T0CKI pin).

2.3.1.1 Using Timer0 with an Internal Clock: Timer mode

Timer mode is selected by clearing the T0CS bit (OPTION<5>). In timer mode, the timer0 register (TMR0) will increment every instruction cycle (without pre-scaler). If TMR0 register is written, the increment is inhibited for the following two cycles.

2.3.1.2 Using Timer0 with an External Clock: Counter mode

Counter mode is selected by setting the T0CS bit (OPTION<5>). In this mode, Timer0 will increment either on every rising or falling edge of pin T0CKI. The incrementing edge is determined by the source edge select bit T0SE (OPTION<4>).
The external clock requirement is due to internal phase clock (T_{OSC}) synchronization. Also, there is a delay in the actual incrementing of Timer0 after synchronization.
When no pre-scaler is used, the external clock input is the same as the pre-scaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the pre-scaler output on the T2 and T4 cycles of the internal phase clocks. Therefore, it is necessary for T0CKI to be high for at least 2 T_{OSC} and low for at least 2 T_{OSC}.
When a pre-scaler is used, the external clock input is divided by the asynchronous pre-scaler. For the external clock to meet the sampling requirement, the ripple counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4Tosc divided by the pre-scaler value.

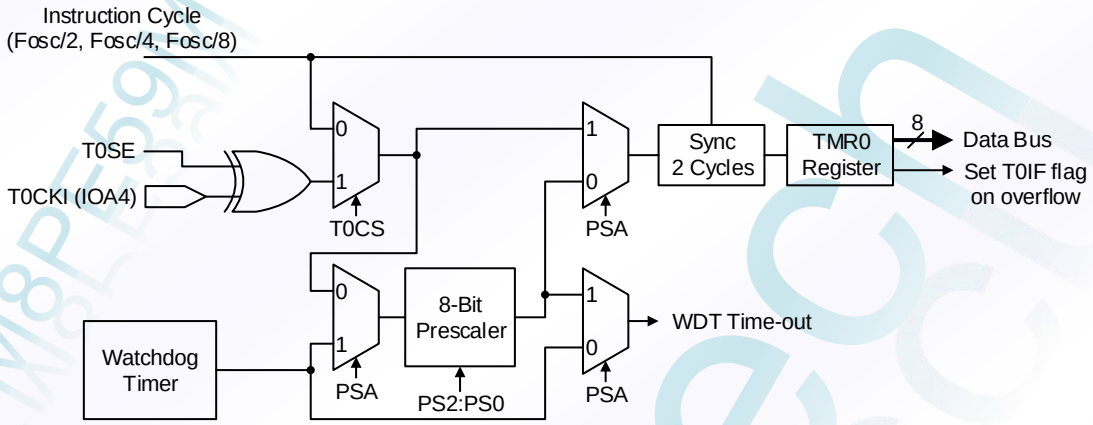
2.3.2 Watchdog Timer (WDT)

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. 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.
The WDT can be disabled by clearing the control bit WDTE (PCON<7>) to “0”.
The WDT has a nominal time-out period of 18ms (without pre-scaler). If a longer time-out period is desired, a pre-scaler with a division ratio of up to 1:128 can be assigned to the WDT controlled by the OPTION register. Thus, the longest time-out period is approximately 2.3 seconds.
The CLRWDT instruction clears the WDT and the pre-scaler, if assigned to the WDT, and prevents it from timing out and generating a device reset.
The SLEEP instruction resets the WDT and the pre-scaler, if assigned to the WDT. This gives the maximum SLEEP time before a WDT Wake-up Reset.

2.3.3 Pre-scaler

An 8-bit counter (down counter) is available as a pre-scaler for the Timer0, or as a post-scaler for the Watchdog Timer (WDT). Note that the pre-scaler may be used by either the Timer0 module or the WDT, but not both. Thus, a pre-scaler assignment for the Timer0 means that there is no pre-scaler for the WDT, and vice-versa.
The PSA bit (OPTION<3>) determines pre-scaler assignment. The PS<2:0> bits (OPTION<2:0>) determine pre-scaler ratio.
When the pre-scaler is assigned to the Timer0 module, all instructions writing to the TMR0 register will clear the pre-scaler. When it is assigned to WDT, a CLRWDT instruction will clear the pre-scaler along with the WDT.
The pre-scaler is neither readable nor writable. On a RESET, the pre-scaler contains all ‘1’s.
To avoid an unintended device reset, CLRWDT or CLRR TMR0 instructions must be executed when changing the pre-scaler assignment from Timer0 to the WDT, and vice-versa.

Figure 2.4: Block Diagram of the Timer0/WDT Pre-scaler



2.4 Timer1

The Timer1 is a 8-bit clock counter with a programmable pre-scaler and a 8-bit period register (PR1). It also can be as a baud rate clock generator for the SPI module. The clock source of Timer1 comes from the internal clock (F_{OSC}/4). The option of Timer1 pre-scaler (1:1, 1:4, 1:8, and 1:16) is defined by T1P1:T1P0 (T1CON<1:0>) bits. **The pre-scaler is cleared when a value is written to TMR1 or T1CON register, and during any kind of reset as well.**

The timer increments from 0x00 until it equals the period register (PR1). It then resets to 00h at the next increment cycle. The timer interrupt flag (T1IF) is set when the timer rollover to 0x00.

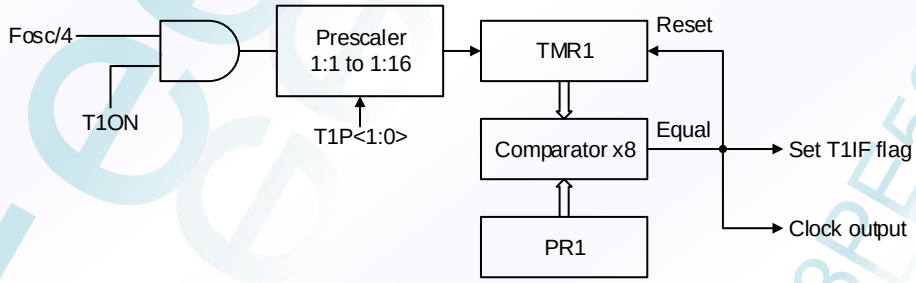
The timer also has a corresponding interrupt enable bit (T1IE). The timer interrupt can be enabled/disabled by setting/clearing this bit.

The timer s can be turned on and off under software control. When the timer on control bit (T1ON, T1CON<2>) is set, the timer increments from the clock source. When T1ON is cleared, the timer is turned off and cannot cause the timer interrupt flag to be set.

Table 2.1: Timer 1 Pre-scaler Rate

T1P1:T1P0		Pre-scaler Rate
0	0	1 : 1
0	1	1 : 4
1	0	1 : 8
1	1	1 : 16

Figure 2.5: Block Diagram of the Timer1



2.5 IR Carrier Output (IROUT)

FM8PE59M is build-in an IR carrier output generator. The output is controlled by IREN (IRCON<7>), IROEN (IRCON<6>), IRCEN (IRCON<5>), IRSC (IRCON<4>), IRPS1:IRPS0 (IRCON<1:0>) bits and IRCYCLE, IRDUTY, IRCPR registers.

Table 2.2: IR Module Clock Source Pre-scaler Bits.

IRPS1:IRPS0		IR Module Clock Source Frequency
0	0	Oscillator Frequency / 1
0	1	Oscillator Frequency / 2
1	0	Oscillator Frequency / 4
1	1	Oscillator Frequency / 8

The IROUT frequency and duty cycle are following the equations below:

IROUT frequency = (IR Module Clock Source Frequency) / IRCYCLE<7:0>
IROUT duty cycle = IRDUTY<7:0> / IRCYCLE<7:0>

For example, if oscillator frequency is equal to 455K$_{HZ}$, and the IRPS1:IRPS0 = (0, 0), IRCYCLE = 12, and IRDUTY = 6, then

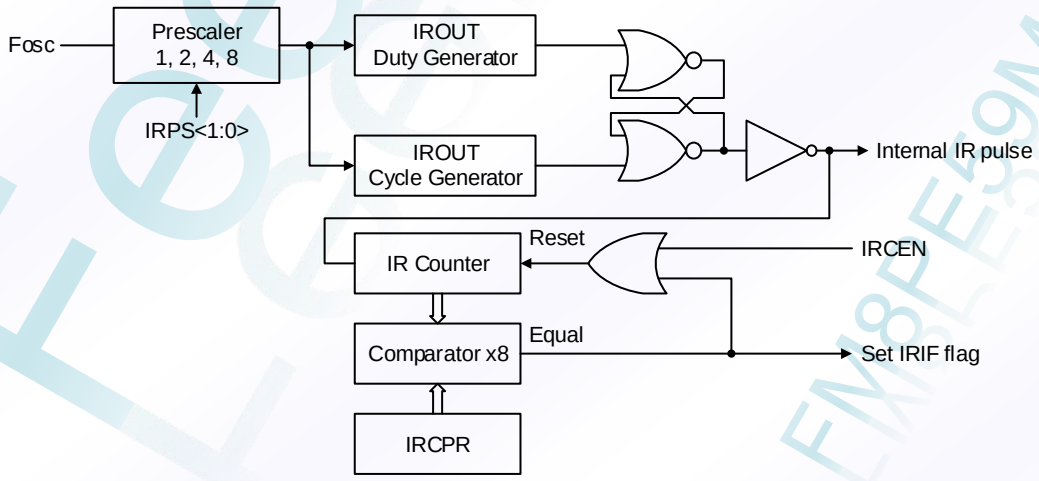
IR Module Clock Source Frequency = 455 K$_{HZ}$ / 1 = 455 K$_{HZ}$;
IROUT frequency = 455K$_{HZ}$ / 12 = 38K$_{HZ}$, and
IROUT duty cycle = 6 / 12 = 50%

Note: 1. Before enabling the IROUT (set IREN = "1"), set the IOB1 (A type) / IOA4 (B type) pin to be an output pin and output "high" for negative pulse or "low" for positive pulse is needed.
2. The value of IRDUTY<7:0> must be less than IRCYCLE<7:0>.

The IR module is also build-in an IROUT counter which increase on every leading edge of internal IR pulse until the value of IR counter matches to IRCPR register, and then the IR counter will be reset to "0", set IRIF interrupt flag, and increase again.

Note: 1. IROUT counter period = ((IRCPP7:IRCPR0) + 1) x (IR Carrier output frequency)
2. The first period of IRIF interrupt may be not equal to ((IRCPR7:IRCPR0) + 1) x (IR Carrier output frequency), which is based-on the timing of enabling IROEN and IROCEN bits.
3. **The IR counter is also cleared when IROCEN (IRCON<5>) bit is cleared, and during any kind of reset as well.**

Figure 2.6: Block Diagram of the IROUT



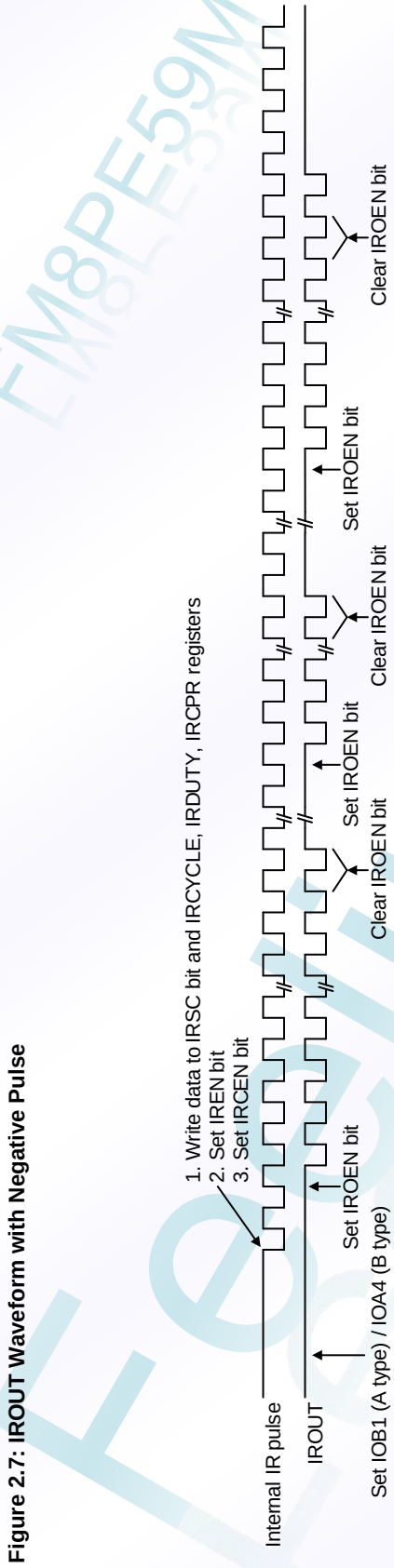


Figure 2.7: IROUT Waveform with Negative Pulse

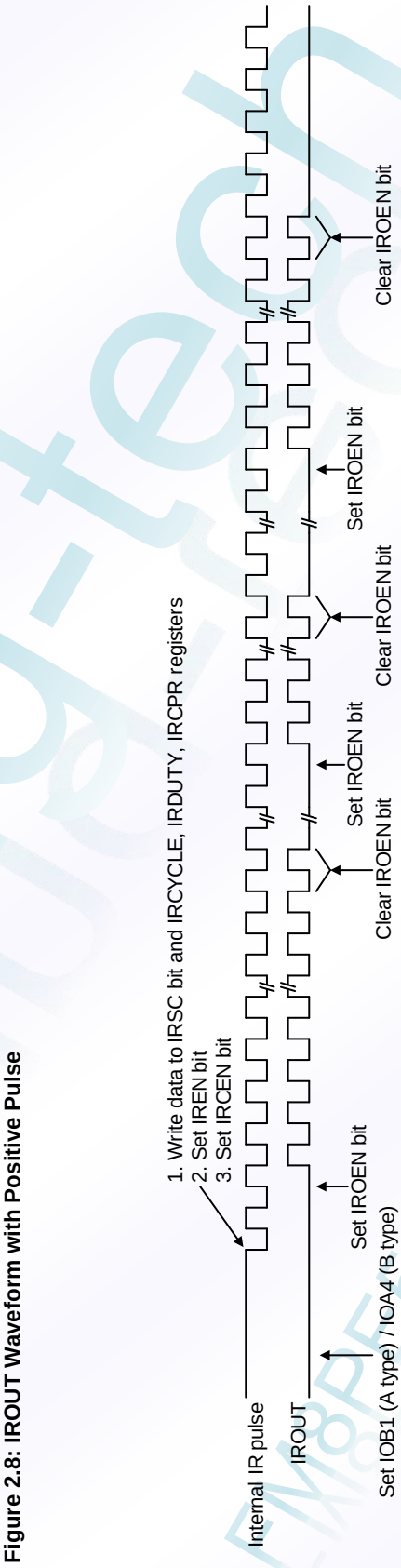


Figure 2.8: IROUT Waveform with Positive Pulse

2.6 SPI (Serial Peripheral Interface) Module

The Serial Port Interface (SPI) Module is a serial interface useful communicating with other peripheral or microcontroller device.
The SPI mode allows 8-bit of data to be synchronously transmitted and received simultaneously. To accomplish communication, typically three pins are used:

1. Serial Clock (SCK)
2. Serial Data In (SDI)
3. Serial Data Output (SDO)

Additionally, a fourth pin may be used when in a slave mode of operation:

4. Slave Select (SSB)

The SPI consists of a transmit/receive shift register (SPISR), a receive buffer register (SPIRXB), and a transmit buffer register (SPITXB). The SPISR shifts the data in and out of the device, MSB first. Once the first valid clock pulse appears on SCK pin (controlled by SSE (SPICON<4>) bit), data in SPITXB will be loaded into SPISR and start to shift in/out. Once the 8-bits of data have been received, the data in SPISR will be moved to the SPIRXB register, then receive buffer full detect bit RXBF (SPISTAT<0>), and interrupt flag bits SPIIF (INTFLAG<6>) are set.
If FM8PE59M is a master controller, it sends clock through the SCK pin. A couple of 8-bit data are transmitted and received at the same time. And if FM8PE59M is defined as a slave, its SCK pin could be programmed as an input pin. Data will continue to be shifted based on both the clock rate and the selected edge.
When the application S/W is expecting to transmit valid data, the SPITXB should be written before the SSE bit is set.
Also, when the application S/W is expecting to receive valid data, the SPIRXB should be read before the next byte of data have been received completely. Buffer full bit RXBF indicates when SPIRXB has been loaded with the received data (reception/transmission is complete). The RXBF bit is cleared by software or by reading SPIRXB register. And the RXBF bit may be ignored if the SPI is only a transmitter.
Generally, the SPI interrupt is used to determine when the transmission/reception has completed, the SPIRXB/SPITXB must be read and/or written. If the interrupt method is not going to be used, then S/W polling RXBF bit is needed.

Figure 2.9: SPI Block Diagram

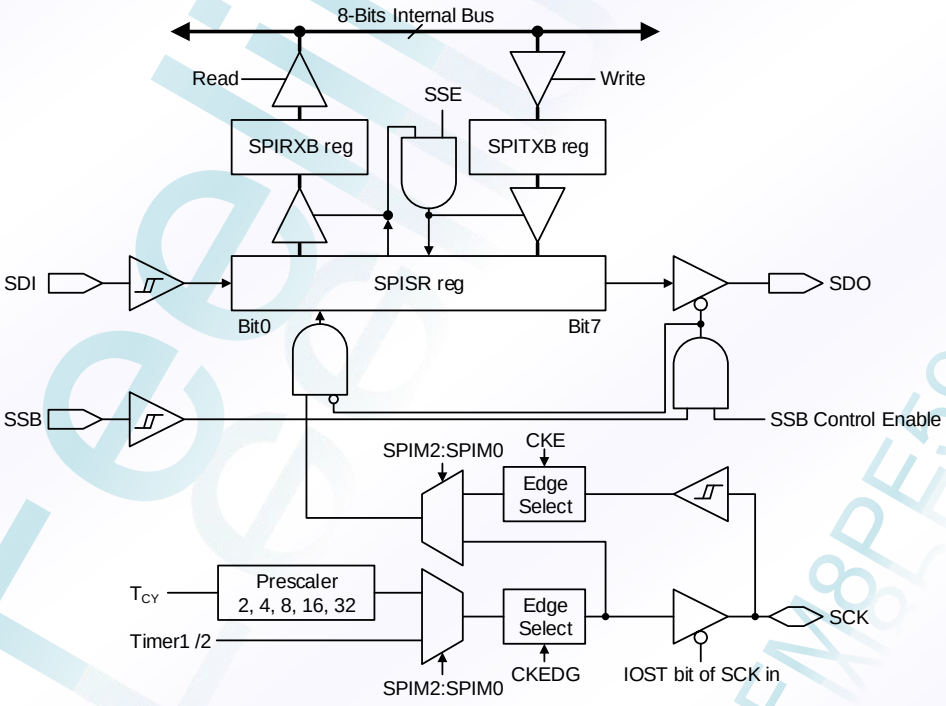


Table 2.3: SPI Mode Setting

SPIM2:SPIM0			SPI Mode
0	0	0	SPI master mode, clock = Fosc/2
0	0	1	SPI master mode, clock = Fosc/4
0	1	0	SPI master mode, clock = Fosc/8
0	1	1	SPI master mode, clock = Fosc/16
1	0	0	SPI master mode, clock = Fosc/32
1	0	1	SPI slave mode, clock = SCK pin, SSB pin control enabled
1	1	0	SPI slave mode, clock = SCK pin, SSB pin control disabled
1	1	1	SPI master mode, clock = Timer1 output/2

Table 2.4: The Description SPI SCK Control Bit

CKEDG:	= 0, Data shifts out on rising edge of SCK, and shifts in on falling edge of SCK. = 1, Data shifts out on falling edge of SCK, and shifts in on rising edge of SCK.

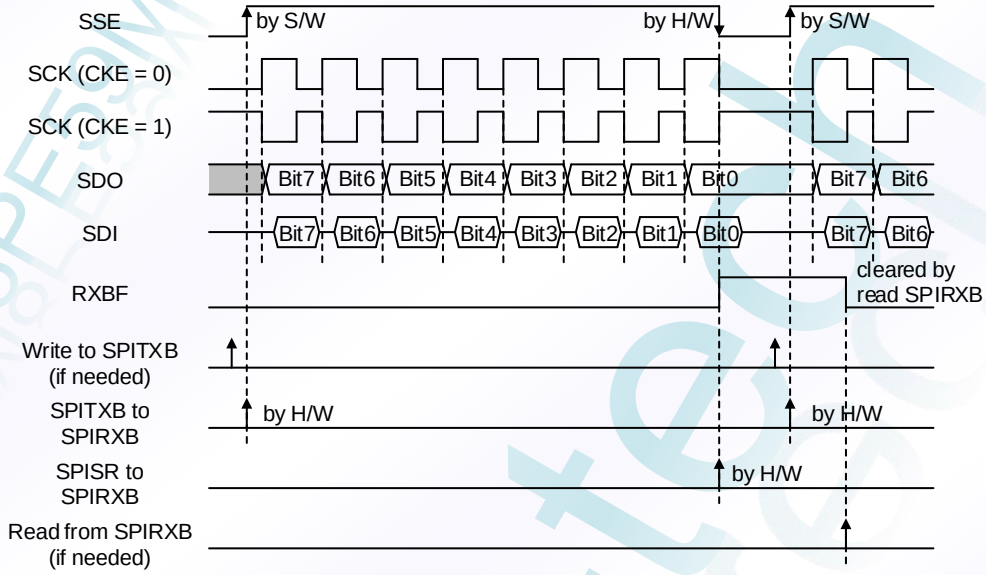
2.6.1 Master Mode

In master mode, the data is transmitted / received as soon as the SPI shift register enable bit SSE (SPICON<4>) bit is setting to “1” by S/W. The data in SPITXB will be loaded into SPISR at the same time and start to shift in/out. The SSE bit will be kept in “1” if the communication is still undergoing, and the SSE bit will be cleared by hardware while the shifting is completed. Once the 8-bits of data have been received, the data in SPISR will be moved to the SPIRXB register, then buffer full detect bit (RXBF), interrupt flag bit (SPIIF) are set. And then user could read out the SPIRXB register before next 8-bit data transmission is completed if needed.

How to transmit/receive data in this master mode:

1. Enable SPI function by setting the SPION (SPICON<6>) bit.
2. Decide the transmission rate and source by programming SPIM2:SPIM0 (SPICON<2:0>) bits.
3. Write the data that you want to transmit to SPIRXB register if needed.
4. Set SSE (SPICON<4>) bit to start transmit.
5. When the 8-bit data transmission is completed, the SSE bit will be reset to “0” by hardware. Therefore, if user wants to transmit/receive another 8-bit data, write next byte data to SPITXB register and set SSE bit to “1” again.
6. When the 8-bit data transmission is completed, the SPIIF interrupt flag will set to 1. Besides, the bit is cleared by software. The RXBF flag also will be set to “1”, cleared by software or by reading out SPIRXB register.
7. Read out the SPIRXB register before next byte transmission being finished if needed.

Figure 2.10: SPI Mode Timing (Master Mode)



2.6.2 Slave Mode

In slave mode, the data is transmitted and received as the external clock pulses appear on SCK pin. Once the SPI shift register enable bit SSE (SPICON<4>) has been set to "1", data in SPITXB will be loaded into SPISR and start to shift in/out. The SSE bit will be kept in "1" if the communication is still undergoing, and the SSE bit will be cleared by hardware while the shifting is completed. Once the 8-bits of data have been received, the data in SPISR will be moved to the SPIRXB register, then buffer full detect bit (RXBF), interrupt flag bit (SPIIF) are set. And then user could read out the SPIRXB register before next 8-bit data transmission is completed if needed.
The SSB pin allows a synchronous slave mode. The SPI must be in slave mode with SSB pin control enabled (SPICON<2:0> = 101). When the SSB pin is low, transmission and reception are enabled and the SDO pin is driven. When the SSB pin goes high, the SDO pin is no longer driven, even if in the middle of transmitted byte, and becomes a floating output. External pull-up/pull-down resistors may be desirable, depending on the application.

How to transmit/receive data in this slave mode:

1. Enable SPI function by setting the SPION (SPICON<6>) bit.
2. Enable/disable the SSB pin control by programming SPIM2:SPIM0 (SPICON<2:0>) bits.
3. Write the data that you want to transmit to SPITXB register if needed.
4. Set SSE (SPICON<4>) bit and wait the external clock pulses appear on SCK pin to start transmit.
5. Write next new data to SPITXB register before this byte transmission being finished if needed.
6. When the 8-bit data transmission is completed, the SSE bit will be reset to "0" by hardware. Therefore, if user wants to transmit/receive another 8-bit data, user must write next byte data to SPITXB register (if needed) and set SSE bit to "1" again before next clock pulse appearing SCK pin.
7. When the 8-bit data transmission is completed, the SPIIF interrupt flag will set to 1. Besides, the bit is cleared by software. The RXBF flag also will be set to "1", cleared by software or by reading out SPIRXB register.
8. Read out the SPIRXB register before next byte transmission being finished if needed.

Figure 2.11: SPI Mode Timing (Slave Mode, with SSB control enabled)

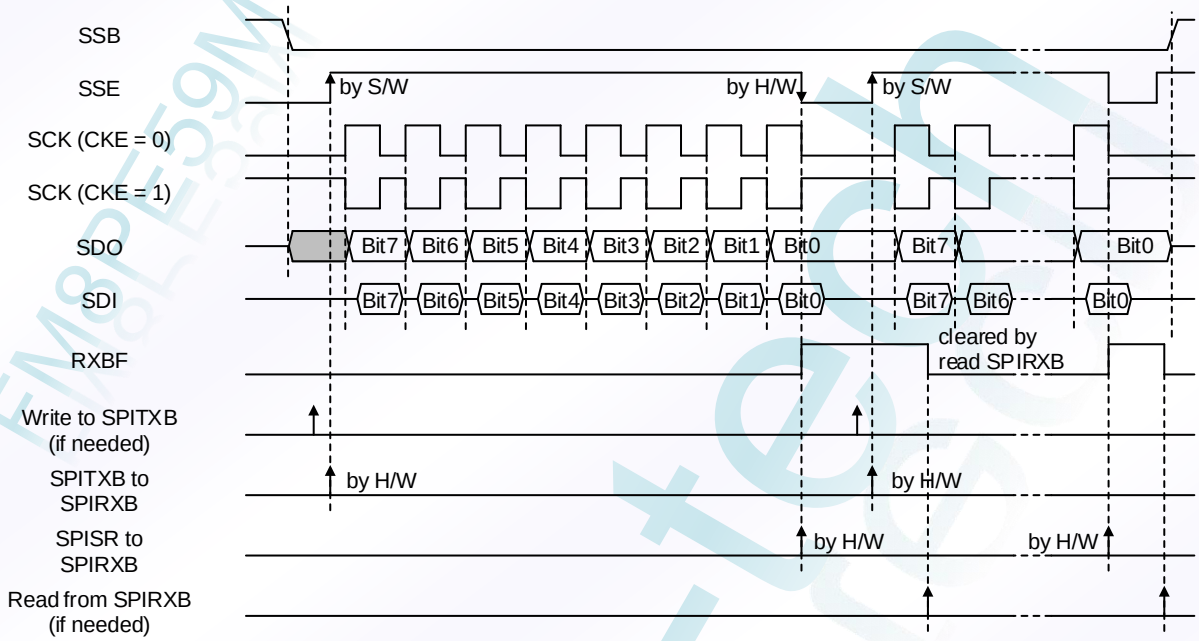
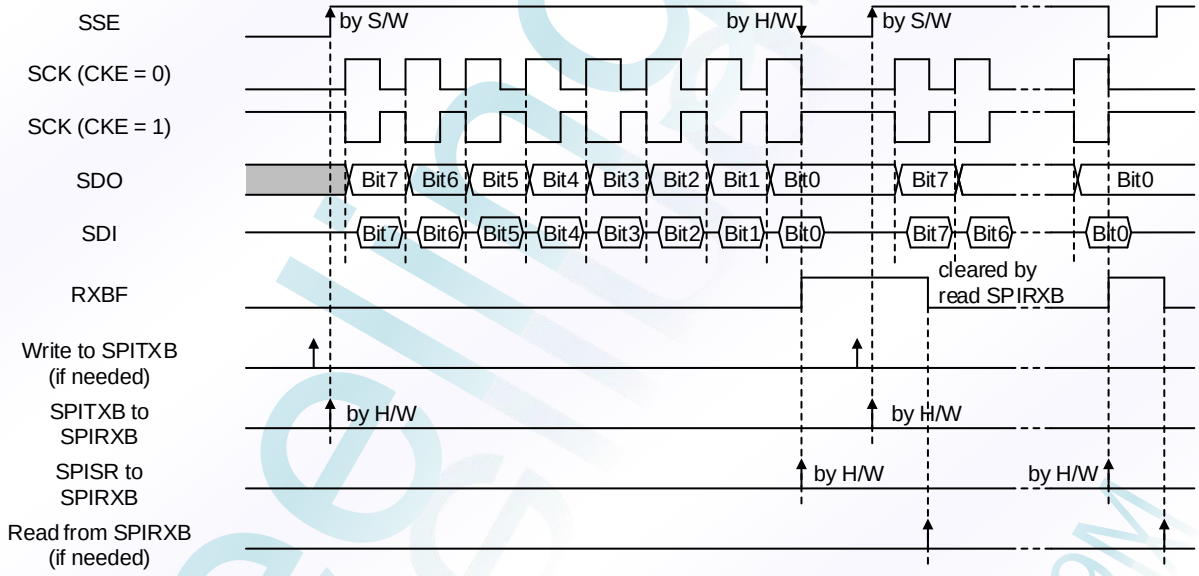


Figure 2.12: SPI Mode Timing (Slave Mode, with SSB control disabled)



2.7 Interrupts

The FM8PE59M has up to seven sources of interrupt:
1. TMR0 overflow interrupt.
2. TMR1 match interrupt.
3. External interrupt INT0 pin.
4. External interrupt INT1 pin.
5. IROUT interrupt.
6. SPI module interrupt.
7. Low-Voltage detector interrupt.

INTFLAG is the interrupt flag register that recodes the interrupt requests in 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 occurs 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 0x008.
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 flag bit in INTFLAG register is set by interrupt event regardless of the status of its mask bit. Reading the INTFLAG register will be the logic AND of INTFLAG and INTEN.
When an interrupt is generated by the INT instruction, the next instruction will be fetched from address 0x002.

2.7.1 Timer0 Interrupt

An overflow (0xFF → 0x00) in the TMR0 register will set the flag bit T0IF (INTFLAG<0>). This interrupt can be disabled by clearing T0IE bit (INTEN<0>).

2.7.2 Timer1 Interrupt

A match condition (TMR1 = PR1) in the TMR1 register will set the flag bits T1IF (INTFLAG<1>).
This interrupt can be disabled by clearing T1IE bit (INTEN<1>).

2.7.3 External INT0 Interrupt

External interrupt on INT0 pin is rising or falling edge triggered selected by INTEDG (OPTION<6>).
When a valid edge appears on the INT0 pin the flag bit INT0IF (INTFLAG<2>) is set. This interrupt can be disabled by clearing INT0IE bit (INTEN<2>).

2.7.4 External INT1 Interrupt

External interrupt on INT1 pin is falling edge triggered.
When a falling edge appears on the INT1 pin the flag bit INT1IF (INTFLAG<3>) is set. This interrupt can be disabled by clearing INT1IE bit (INTEN<3>).

2.7.5 IROUT Interrupt

The IROUT interrupt flag bit IRIF (INTFLAG<5>) is set whenever the value of IR counter matches to IRCPR register. This interrupt can be disabled by clearing IRIE bit (INTEN<5>).

2.7.6 SPI Module Interrupt

After one byte of SPI transmission is completed, the flag bit SPIIF (INTFLAG<6>) will be set.
This interrupt can be disabled by clearing SPIIE bit (INTEN<6>).

2.7.7 Low-Voltage Detector Interrupt

When a low-voltage condition was detected, the flag bit LVDTIF (INTFLAG<4>) is set. This interrupt can be disabled by clearing LVDTIE bit (INTEN<4>).
The Low-Voltage Detector Interrupt function will be fixed to “Disable” by H/W if the configuration bit IOA5OD= Disable, even if bit LVDTIE = 1.
To enable the Low-Voltage Detector Interrupt, clear the bit LVDTE (PCON<5>) to "0" by S/W is needed.

Table 2.5: LVDT Operation mode (V_{DD} < LVDT detect voltage)

Configuration LVDT bit	LVDTE bit (PCON<5>)	Configuration IOA5OD bit	LVDTIE bit (INTEN<4>)	LVDT action
Disable	x	x	x	Disable
Enable, Voltage = 3.6V to 1.8V	0	Disable	x	Disable
	0	Enable	1	Interrupt, LVDTIF = 1
	0	Enable	0	Disable
	1	Disable	x	Reset
	1	Enable	x	Reset

Legend: x = don’t care (Register = 0 or 1, Configuration = Enable or Disable).

2.8 Power-down Mode (SLEEP)

Power-down mode is entered by executing a SLEEP instruction.
When SLEEP instruction is executed, the $\overline{PD}$ bit (STATUS<3>) is cleared, the $\overline{TO}$ bit is set, the watchdog timer will be cleared and keeps running, and the oscillator driver is turned off.
All I/O pins maintain the status they had before the SLEEP instruction was executed.

2.8.1 Wake-up from SLEEP Mode

The device can wake-up from SLEEP mode through one of the following events:
1. RSTB reset.
2. WDT time-out reset (if enabled).
3. PORTB/IOC4/IOC5 input falling.

External RSTB reset and WDT time-out reset will cause a device reset. The $\overline{PD}$ and $\overline{TO}$ bits can be used to determine the cause of device reset. The $\overline{PD}$ bit is set on power-up and is cleared when SLEEP instruction is executed. The $\overline{TO}$ bit is cleared if a WDT time-out occurred.
For the device to wake-up through an PORTB/IOC4/IOC5 input falling event, and the program will execute next PC after wake-up. Any pin which corresponding /WUBn bit (WUCON<7:0>) or /WUC45 bit (PCON<0>) is set to “1” or configured as output will be excluded from this function.
The system wake-up delay time is 18ms plus 128 oscillator cycle time.

2.9 Reset

FM8PE59M devices 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 V_{DD} rise is detected. To use this feature, the user merely ties the RSTB pin to V_{DD}.

On-chip Low Voltage Detector (LVD) places the device into reset when V_{DD} is below a fixed voltage. This ensures that the device does not continue program execution outside the valid operation V_{DD} 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.

2.9.1 Power-up Reset Timer (PWRT)

The Power-up Reset Timer provides a nominal 18ms 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 V_{DD}, temperature, and process variation.

2.9.2 Oscillator Start-up Timer (OST)

The OST timer provides a 128 oscillator cycle delay (from OSCI input) after the PWRT delay (18ms) is over. This delay ensures that the X’tal oscillator or resonator has started and stabilized. The device is kept in reset state as long as the OST is active.

This counter only starts incrementing after the amplitude of the OSCI signal reaches the oscillator input thresholds.

2.9.3 Reset Sequence

When Power-on Reset (POR), Brown-out Reset (BOR), RSTB Reset or WDT time-out Reset is detected, the reset sequence is as follows:

1. The reset latch is set and the PWRT & OST are cleared.
2. When the internal POR, BOR, RSTB Reset or WDT time-out Reset pulse is finished, then the PWRT begins counting.
3. After the PWRT time-out, the OST is activated.
4. And after the OST delay is over, the reset latch will be cleared and thus end the on-chip reset signal.

The totally system reset delay time is 18ms plus 128 oscillator cycle time.

Figure 2.13: Simplified Block Diagram of on-chip Reset Circuit

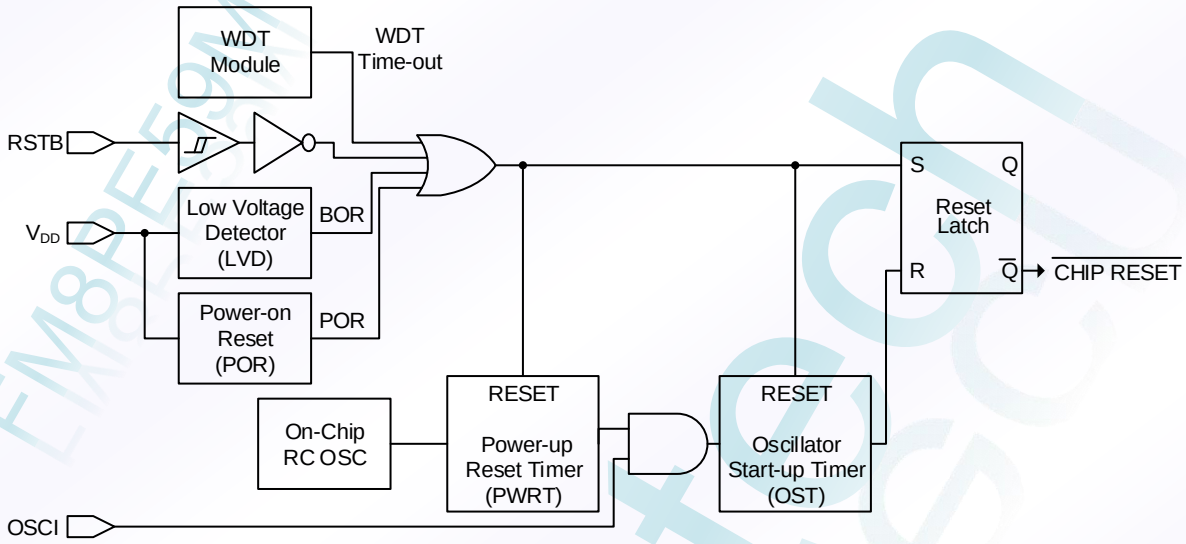
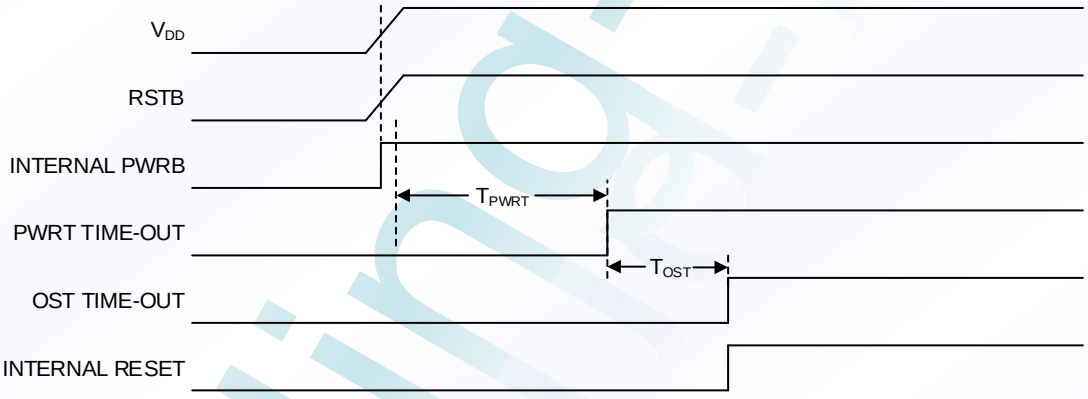
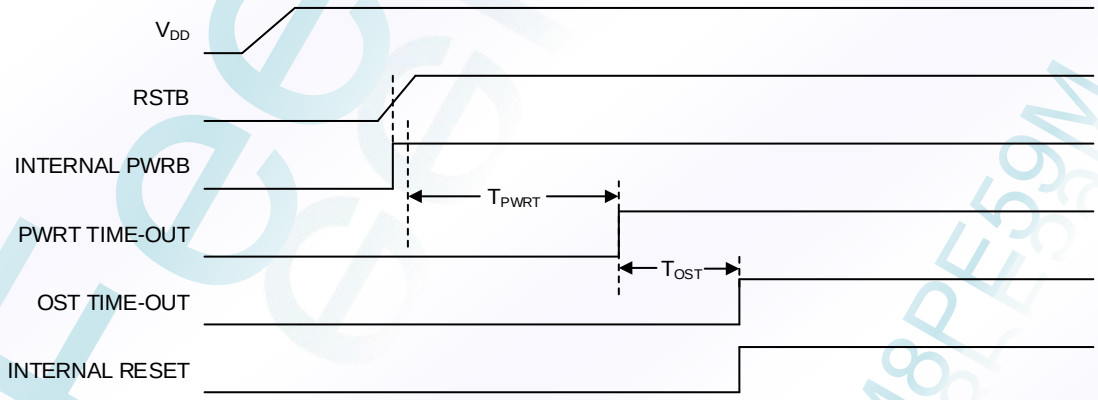


Figure 2.14: Time-out Sequence on Power-up (RSTB Pin Tied to V_{DD})



Note: T_{PWRT} = 18ms; T_{OST} = 128 oscillator cycle time

Figure 2.15: Time-out Sequence on Power-up (RSTB Pin Not Tied to V_{DD})



Note: T_{PWRT} = 18ms; T_{OST} = 128 oscillator cycle time

Table 2.6: Reset Conditions for All Registers

Register	Address	Power-on Reset Brown-out Reset	RSTB Reset WDT Reset
ACC	N/A	xxxx xxxx	uuuu uuuu
OPTION	N/A	-011 1111	-011 1111
IOSTA	0x05	1111 1111	1111 1111
IOSTB	0x06	1111 1111	1111 1111
IOSTC	0x07	1111 1111	1111 1111
IRCON	0x0C	0000 --00	0000 --00
IRCYCLE	0x0D	0000 1100	0000 1100
IRDUTY	0x0E	0000 0110	0000 0110
IRCPR	0x0F	0000 0000	0000 0000
INDF	0x00, unbanked	xxxx xxxx	uuuu uuuu
TMR0	0x01, unbanked	xxxx xxxx	uuuu uuuu
PCL	0x02, unbanked	1111 1111	1111 1111
STATUS	0x03, unbanked	0001 1xxx	000# #uuu
FSR	0x04, unbanked	xxxx xxxx	uuuu uuuu
PORTA	0x05, unbanked	xxxx xxxx	uuuu uuuu
PORTB	0x06, unbanked	xxxx xxxx	uuuu uuuu
PORTC	0x07, unbanked	xxxx xxxx	uuuu uuuu
PCON	0x08, unbanked	1010 --00	1010 --00
WUCON	0x09, unbanked	0000 0000	0000 0000
PCHBUF	0x0A, unbanked	---- 0000	---- 0000
PDCON	0x0B, unbanked	1111 1111	1111 1111
BPHCON	0x0C, unbanked	1111 1111	1111 1111
CPHCON	0x0D, unbanked	1111 1111	1111 1111
INTEN	0x0E, unbanked	0000 0000	0000 0000
T1CON	0x0B, bank 1	---- -111	---- -111
TMR1	0x0C, bank 1	1111 1111	1111 1111
PR1	0x0D, bank 1	1111 1111	1111 1111
SPIRXB	0x0B, bank 3	xxxx xxxx	uuuu uuuu
SPITXB	0x0C, bank 3	xxxx xxxx	uuuu uuuu
SPISTAT	0x0D, bank 3	00-- 00-0	00-- 00-0
SPICON	0x0E, bank 3	0000 -000	0000 -000
INTFLAG	0x0F, unbanked	-000 0000	-000 0000
General Purpose Registers	0x10 ~ 0x3F	xxxx xxxx	uuuu uuuu

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

Table 2.7: $\overline{TO}$ / $\overline{PD}$ Status after Reset or Wake-up

$\overline{TO}$	$\overline{PD}$	RESET was caused by
1	1	Power-on Reset
1	1	Brown-out reset
u	u	RSTB Reset during normal operation
1	0	RSTB Reset during SLEEP
0	1	WDT Reset during normal operation
0	0	WDT Wake-up during SLEEP

Legend: u = unchanged

Table 2.8: Events Affecting $\overline{TO}$ / $\overline{PD}$ Status Bits

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

2.10 Hexadecimal Convert to Decimal (HCD)

Decimal format is another number format for FM8PE59M series. 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 2.2.

Example 2.2: DAA CONVERSION

```
Code
#include  <8PE59M.ASH>
          …
          MOVIA   0x90      ;Set immediate data = decimal format number “90” (ACC ← 0x90)
          MOVAR   0x30      ;Load immediate data “90” to data memory address 0x30
          MOVIA   0x10      ;Set immediate data = decimal format number “10” (ACC ← 0x10)
          ADDAR   0x30,A    ;Contents of the data memory address 0x30 and ACC are binary-added
                            ;the result loads to the ACC (ACC ← 0xA0, C ← 0)
          DAA               ;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”
          …
```

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 2.3.

Example 2.3: DAS CONVERSION

```
Code
#include  <8PE59M.ASH>
          …
          MOVIA   0x10    ;Set immediate data = decimal format number “10” (ACC ← 0x10)
          MOVAR   0x30    ;Load immediate data “90” to data memory address 0x30
          MOVIA   0x20    ;Set immediate data = decimal format number “20” (ACC ← 0x20)
          SUBAR   0x30,A  ;Contents of the data memory address 0x30 and ACC are binary-subtracted
                          ;the result loads to the ACC (ACC ← 0xF0, C ← 0)
          DAS             ;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”
          …
```

2.11 Oscillator Configurations

FM8PE59M can be operated in six different oscillator modes. Users can program F_{OSC} configuration bit to select the appropriate modes:

- ERC: External Resistor/Capacitor Oscillator
- HF: High Frequency Crystal/Resonator Oscillator
- XT: Crystal/Resonator Oscillator
- LF: Low Frequency Crystal Oscillator
- IRC: Internal Resistor/Capacitor Oscillator
- ERIC: External Resistor/Internal 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/ERIC device option offers largest cost savings for timing insensitive applications. These devices offer 4 different internal RC oscillator frequency, 8 M_{HZ}, 4 M_{HZ}, 1 M_{HZ}, and 455 K_{HZ}, which is selected by configuration bits (F_{OSC}). Or user can change the oscillator frequency with external resistor. The ERIC oscillator frequency is a function of the resistor (Rext), the operating temperature, and the process parameter.

Figure 2.16: HF, XT or LF Oscillator Modes (Crystal Operation or Ceramic Resonator)

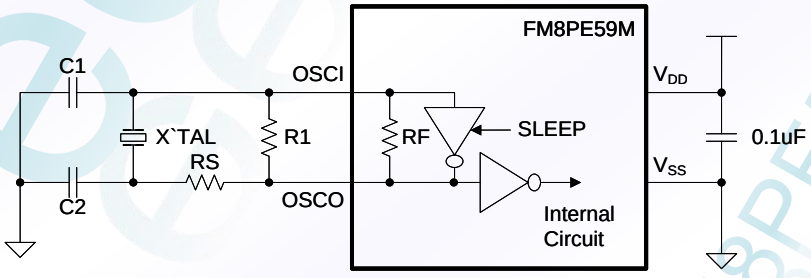


Figure 2.17: HF, XT or LF Oscillator Modes (External Clock Input Operation)

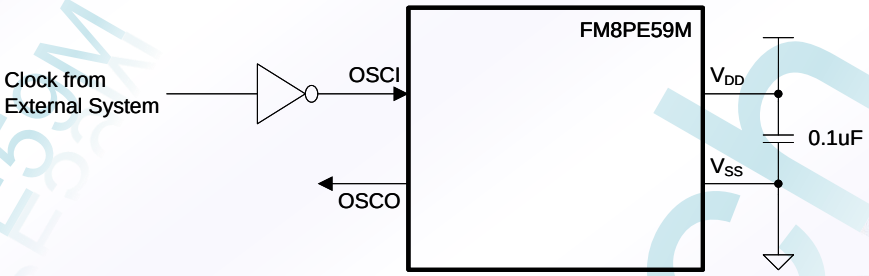


Figure 2.18: ERC Oscillator Mode (External RC Oscillator)

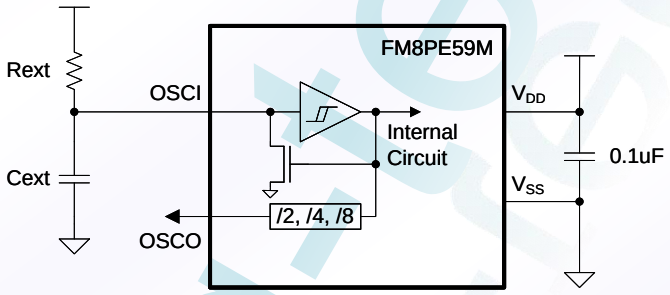
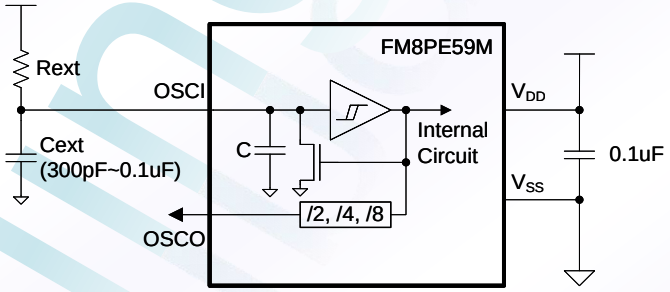


Figure 2.19: ERIC Oscillator Mode (External R, Internal C Oscillator)

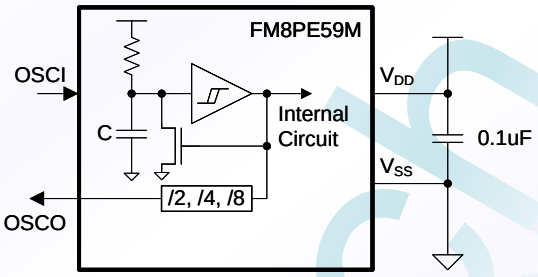


The typical oscillator frequency vs. external resistor is as following table

Frequency	Rext @ 3V	Rext @ 5V
455K$_{HZ}$	998.9K	1478.9K
1M$_{HZ}$	709.7K	949.2K
4M$_{HZ}$	282K	322K
8M$_{HZ}$	158.6K	166.5K
16M$_{HZ}$	N/A	84.6K

Note: Values are provided for design reference only.

Figure 2.20: IRC Oscillator Mode (Internal R, Internal C Oscillator)



2.12 Configuration Words

Table 2.9: Configuration Words

Name	Description
FOSC	Oscillator Selection Bit → ERC mode (external R & C) (default) → HF mode → XT mode → LF mode → 4MHZ IRC mode (internal R & C) → 8MHZ IRC mode (internal R & C) → 1MHZ IRC mode (internal R & C) → 455KHZ IRC mode (internal R & C) → ERIC mode (external R & internal C) Note: See Table 2.10 for detail description.
WDTEN	Watchdog Timer Enable Bit → WDT enabled (default) → WDT disabled
LVDT	Low Voltage Detector Selection Bit → Disable (default) → Enable, LVDT voltage = 3.6V → Enable, LVDT voltage = 2.6V → Enable, LVDT voltage = 2.4V → Enable, LVDT voltage = 2.2V → Enable, LVDT voltage = 2.0V → Enable, LVDT voltage = 2.0V, controlled by SLEEP → Enable, LVDT voltage = 1.8V
T0CKIN	IOA4/T0CKI Pin Selection Bit (Only for A type, force to T0CKI for B type) → T0CKI pin is selected (default) → Both IOA4 and T0CKI pin is selected
RSTBIN	IOA5/RSTB Pin Selection Bit (Only for A type, force to RSTB for B type) → IOA5 pin is selected (default) → RSTB pin is selected
OSCOUT	IOA6/OSCO Pin Selection Bit for ERC/IRC/ERIC Mode (Only for A type, force to OSCO for B type) → OSCO pin is selected; Instruction clock will be output (default) → IOA6 pin is selected
OSCIN	IOA7/OSCI Pin Selection Bit for IRC Mode (Only for A type, force to OSCI for B type) → OSCI pin is selected (default) → IOA7 pin is selected
TYPE	Type Selection Bit → A type (28-pin) is selected (default) → B type (32-pin) is selected

Name	Description
PROTECT	Code Protection Bit → OTP code protection off (default) → OTP code protection on
OSCD	Instruction Period Selection Bits → Four oscillator periods (default) → Two oscillator periods → Eight oscillator periods
PMOD	Power Mode Selection Bit → Non-power saving (default) → Power saving
RDPORT	Read Port Control Bit for Output Pins → From registers (default) → From pins
COUT	Instruction clock Output Enable Bit for OSCO Pin (Only for ERC/IRC/ERIC Mode) → Instruction clock will be output (default) → Instruction clock will be not output
SCHMITT	I/O Pin Input Buffer Control Bit → With Schmitt-trigger (default) → Without Schmitt-trigger
RBANK	Operational Registers Bank Enable Bit → Disable register (0x0B ~ 0x0E) banks; These registers are all memory map back to address in BANK 0. (default) → Enable register (0x0B ~ 0x0E) banks.
DEL	SDI Input Delay Time Selection Bit → 0ns (default) → 50ns → 100ns
WUOPT	Wake up Trigger Source Control Bit → Falling Edge Trigger (default) → Low Level Trigger
IOA5OD	IOA5 Pin Open-Drain Output and LVDT interrupt Enable Bit → Enable IOA5/RSTB (A-type) pin open-drain output and LVDT interrupt function (default) → Disable IOA5/RSTB (A-type) pin open-drain output and LVDT interrupt function

Table 2.10: Selection of IOA7/OSCI and IOA6/OSCO Pin for A Type (28 pin)

Mode of oscillation	IOA7/OSCI	IOA6/OSCO
IRC	IOA7	IOA6/OSCO selected by OSCOUT bit
	OSCI (No function)	
ERC, ERIC	OSCI	IOA6/OSCO selected by OSCOUT bit
HF, XT, LF	OSCI	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	1/2/3(1)	-
BTRSS	**R, bit**	Test bit in R, Skip if Set	Skip if R<b> = 1	1/2/3(1)	-
NOP		No Operation	No operation	1	-
CLRWDT		Clear Watchdog Timer	0x00 → WDT, 0x00 → WDT pre-scaler	1	$\overline{TO}$, $\overline{PD}$
SLEEP		Go into power-down mode	0x00 → WDT, 0x00 → WDT pre-scaler	1	$\overline{TO}$, $\overline{PD}$
OPTION		Load OPTION register	ACC → OPTION	1	-
OPTIONR		Read OPTION register	OPTION → ACC	1	-
DAA		Adjust ACC's data format from HEX to DEC after any addition operation	ACC(hex) → ACC (dec)	1	C
DAS		Adjust ACC's data format from HEX to DEC after any subtraction operation	ACC(hex) → ACC (dec)	1	-
RETURN		Return from subroutine	Top of Stack → PC	2	-
RETFIE		Return from interrupt, set GIE bit	Top of Stack → PC, 1 → GIE	2	-
IOST	**R**	Load IOST register	ACC → IOST register	1	-
IOSTR	**R**	Read IOST register	IOST register → ACC	1	-
TBL		Table look-up	PC<7:0> + ACC → PC<7:0> PC<9:8> unchanged PCHBUF<3:2> → PC<11:10>	1	C, DC, Z
CLRA		Clear ACC	0x00 → ACC	1	Z
CLRR	**R**	Clear R	0x00 → R	1	Z
MOVAR	**R**	Move ACC to R	ACC → R	1	-
MOVR	**R, d**	Move R	R → dest	1	Z
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	1/2/3(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	1/2/3(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
RLR	**R, d**	Rotate left R through Carry	R<7> → C, R<6:0> → dest<7:1>, C → dest<0>	1	C

Mnemonic, Operands		Description	Operation	Cycles	Status Affected
RRR	**R, d**	Rotate right R through Carry	C → dest<7>, R<7:1> → dest<6:0>, R<0> → C	1	C
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	-
BANK	**I**	Move Immediate to memory bank bits	I → RP<1:0>	1	-
PAGE	**I**	Move Immediate to program page bits	I → PCHBUF<3:2>	1	-
CALL	**I**	Call subroutine	PC + 1 → Top of Stack, I → PC<9:0> PCHBUF<3:2> → PC<11:10>	2	-
GOTO	**I**	Unconditional branch	I → PC<9:0> PCHBUF<3:2> → PC<11:10>	2	-
FCALL	**I**	Call subroutine	PC + 1 → Top of Stack, I → PC<11:0> I<11:10> → PCHBUF<3:2>	3	-
FGOTO	**I**	Unconditional branch	I → PC<11:0> I<11:10> → PCHBUF<3:2>	3	-

Note: 1.2 cycles for skip, else 1 cycle. (3 cycles if skip and followed by a 2-word instruction FCALL/FGOTO)

2.
- bit: Bit address within an 8-bit register R
- R: Register address (0x00 to 0x3F)
- 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
- RP: RAM Page(Bank) Select Bits
- PCHBUF: Program Counter High-byte buffer
- 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:	0x00≤R≤0x3F 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:	0x00≤R≤0x3F 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:	0x00≤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:	0x00≤R≤0x3F 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:	0x00≤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

BANK	Move Immediate to memory bank bits
Syntax:	BANK I
Operands:	0x0≤I≤0x3
Operation:	I → RP<1:0>
Status Affected:	None
Description:	The memory bank bits are loaded with the 2-bit immediate 'I'.
Cycles:	1

BCR	Clear Bit in R
Syntax:	BCR R, b
Operands:	0x00≤R≤0x3F 0x0≤b≤0x7
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:	0x00≤R≤0x3F 0x0≤b≤0x7
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:	0x00≤R≤0x3F 0x0≤b≤0x7
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 (3 cycles if skip and followed by a 2-word instruction FCALL/FGOTO)

BTRSS	Test Bit in R, Skip if Set
Syntax:	BTRSS R, b
Operands:	0x00≤R≤0x3F 0x0≤b≤0x7
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 (3 cycles if skip and followed by a 2-word instruction FCALL/FGOTO)

CALL	Subroutine Call
Syntax:	CALL I
Operands:	0x000≤I≤0xFFF
Operation:	PC + 1 → Top of Stack, I → PC<9:0> PCHBUF<3:2> → PC<11:10>
Status Affected:	None
Description:	Subroutine call. First, return address (PC+1) is pushed onto the stack. The 10-bit immediate address is loaded into PC bits <9:0>.
Cycles:	2

CLRA	Clear ACC
Syntax:	CLRA
Operands:	None
Operation:	0x00 → 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:	0x00≤R≤0x3F
Operation:	0x00 → 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:	0x00 → WDT; 0x00 → WDT pre-scaler (if assigned); 1 → $\overline{TO}$; 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	The CLRWDT instruction resets the WDT. It also resets the pre-scaler, if the pre-scaler is assigned to the WDT and not Timer0. Status bits $\overline{TO}$ and $\overline{PD}$ are set.
Cycles:	1

COMR	Complement R
Syntax:	COMR R, d
Operands:	0x00≤R≤0x3F 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
Operands:	None
Operation:	ACC(hex) → ACC(dec)
Status Affected:	C
Description:	Convert the ACC data from hexadecimal to decimal format after any addition operation and restored to ACC.
Cycles:	1

DAS	Adjust ACC's data format from HEX to DEC
Syntax:	DAS
Operands:	None
Operation:	ACC(hex) → ACC(dec)
Status Affected:	None
Description:	Convert the ACC data from hexadecimal to decimal format after any subtraction operation and restored to ACC.
Cycles:	1

DECR	Decrement R
Syntax:	DECR R, d
Operands:	0x00≤R≤0x3F 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:	0x00≤R≤0x3F 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 2-cycle instruction.
Cycles:	1/2 (3 cycles if skip and followed by a 2-word instruction FCALL/FGOTO)

FCALL	Subroutine Call
Syntax:	FCALL I
Operands:	0x000≤I≤0xFFF
Operation:	PC +1 → Top of Stack; I → PC<11:0> I<11:10> → PCHBUF<3:2>
Status Affected:	None
Description:	Subroutine call. First, return address (PC+1) is pushed onto the stack. The 12-bit immediate address is loaded into PC bits <11:0>. FCALL is a two-word (3-cycle) instruction.
Cycles:	3

FGOTO	Unconditional Branch
Syntax:	FGOTO I
Operands:	0x000≤I≤0xFFF
Operation:	I → PC<11:0> I<11:10> → PCHBUF<3:2>
Status Affected:	None
Description:	FGOTO is an unconditional branch. The 12-bit immediate value is loaded into PC bits <11:0>. FGOTO is a two-word (3-cycle) instruction.
Cycles:	3

GOTO	Unconditional Branch
Syntax:	GOTO I
Operands:	0x000≤I≤0x3FF
Operation:	I → PC<9:0> PCHBUF<3:2> → PC<11:10>
Status Affected:	None
Description:	GOTO is an unconditional branch. The 10-bit immediate value is loaded into PC bits <9:0>.
Cycles:	2

INCR	Increment R
Syntax:	INCR R, d
Operands:	0x00≤R≤0x3F 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:	0x00≤R≤0x3F 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 2-cycle instruction.
Cycles:	1/2 (3 cycles if skip and followed by a 2-word instruction FCALL/FGOTO)

INT	S/W Interrupt
Syntax:	INT
Operands:	None
Operation:	PC + 1 → Top of Stack, 0x002 → PC
Status Affected:	None
Description:	Interrupt subroutine call. First, return address (PC+1) is pushed onto the stack. The address 0x002 is loaded into PC bits <11:0>.
Cycles:	2

IORAR	OR ACC with R
Syntax:	IORAR R, d
Operands:	0x00≤R≤0x3F 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:	0x00≤I≤0x3F
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

IOST	Load IOST Register
Syntax:	IOST R
Operands:	R = 0x05~0x06 or 0x0C~0x0E
Operation:	ACC → IOST register R
Status Affected:	None
Description:	IOST register 'R' (R= 0x05~0x06 or 0x0C~0x0E) is loaded with the contents of the ACC register.
Cycles:	1

IOSTR	Read IOST Register
Syntax:	IOST R
Operands:	R = 0x05~0x06 or 0x0C~0x0E
Operation:	IOST register R → ACC
Status Affected:	None
Description:	The ACC register is loaded with the contents of IOST register 'R' (0x05~0x06 or 0x0C~0x0E).
Cycles:	1

MOVAR	Move ACC to R
Syntax:	MOVAR R
Operands:	0x00≤R≤0x3F
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:	0x00≤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:	0x00≤R≤0x3F 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

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

OPTION	Load OPTION Register
Syntax:	OPTION
Operands:	None
Operation:	ACC → OPTION
Status Affected:	None
Description:	The content of the ACC register is loaded into the OPTION register.
Cycles:	1

OPTIONR	Read OPTION Register
Syntax:	OPTION
Operands:	None
Operation:	OPTION → ACC
Status Affected:	None
Description:	The content of the OPTION register is loaded into the ACC register.
Cycles:	1

PAGE	Move Immediate to program page bits
Syntax:	PAGE I
Operands:	0x0≤I≤0x3
Operation:	I → PCHBUF<3:2>
Status Affected:	None
Description:	The program page bits are loaded with the 2-bit immediate 'I'.
Cycles:	1

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 2-cycle instruction.
Cycles:	2

RETIA	Return with Immediate in ACC
Syntax:	RETIA I
Operands:	0x00≤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 2-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 2-cycle instruction.
Cycles:	2

RLR	Rotate Left R through Carry
Syntax:	RLR R, d
Operands:	0x00≤R≤0x3F 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

RRR	Rotate Right R through Carry
Syntax:	RRR R, d
Operands:	0x00≤R≤0x3F 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:	0x00 → WDT; 0x00 → WDT pre-scaler; 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:	0x00≤R≤0x3F 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:	0x00≤R≤0x3F 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:	0x00≤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:	0x00≤R≤0x3F 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

TBL	**Table Look-up**
Syntax:	TBL
Operands:	None
Operation:	PC<7:0> + ACC → PC<7:0> PC<9:8> unchanged PCHBUF<3:2> → PC<11:10>
Status Affected:	C, DC, Z
Description:	Operate with RETIA to look-up table
Cycles:	1

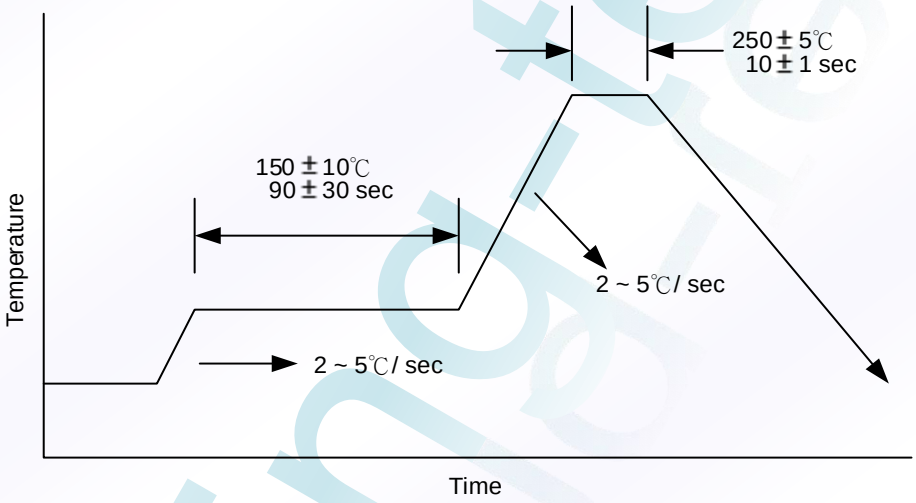
XORAR	Exclusive OR ACC with R
Syntax:	XORAR R, d
Operands:	0x00≤R≤0x3F d∈[0,1]
Operation:	ACC xor R → dest
Status Affected:	Z
Description:	Exclusive OR the contents of the ACC register with register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

XORIA	**Exclusive OR Immediate with ACC**
Syntax:	XORIA I
Operands:	0x00≤I≤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

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
	Ambient Operating Temperature	-	0	-	70	°C
	Store Temperature	-	-65	-	150	°C
V_{DD}	DC Supply Voltage	-	0	-	6	V
	Input Voltage with respect to Ground	-	-0.3	-	V_{DD}+0.3	V
	ESD Susceptibility	HBM (Human Body Mode)	-	2.0	-	KV
		MM (Machine Mode)	-	200	-	V
	Lead Temperature	Soldering, 10 Sec	-	-	250	°C

4.1 PACKAGE IR Re-flow Soldering Curve



5.0 RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
V_{DD}	DC Supply Voltage	-	2.3	-	5.5	V
	Operating Temperature	-	0	-	70	°C

6.0 ELECTRICAL CHARACTERISTICS

6.1 AC Characteristics

Ta=25°C

Symbol	Description	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
F_{HF}	HF Oscillation range	3V	HF mode	4	-	20	MHZ
		5V		4	-	20	
F_{XT}	XT Oscillation range	3V	XT mode	0.455	-	16	MHZ
		5V		0.455	-	20	
F_{LF}	LF oscillation range	3V	LF mode	32	-	32	KHZ
		5V		32	-	32	
F_{ERC}	ERC Oscillation range	3V	ERC mode	DC	-	16	MHZ
		5V		DC	-	16	
F_{ERIC}	ERIC Oscillation range	3V	ERIC mode	DC	-	8	MHZ
		5V		DC	-	16	
F_{IRC}	Internal RC Oscillation range	3V	IRC mode	0.455	-	8	MHZ
		5V		0.455	-	8	
T_{WDT}	WDT period time	3V	Pre-scaler rate=1:1	-	23.6	-	mS
		4V		-	19.4	-	
		5V		-	17.8	-	

Note: 1. In the ERIC mode, to maintain the accuracy of the internal RC oscillator frequency, a 300pF ~ 0.1uF decoupling capacitor should be connected between OSCI and V_{SS} and located as close to the device as possible.

2. At any time, a 0.1μF decoupling capacitor should be connected between V_{DD} and V_{SS} and device as close as possible.

6.2 DC Characteristics

Ta=25°C

Under Operating Conditions, at two clock instruction cycles and WDT & LVDT are disable, I/O output float.

Symbol	Description	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
V_{IH1}	Input high voltage, I/O Ports	3V	With Schmitt-trigger	-	1.28	V_{DD}	V
		5V		1.9	-	V_{DD}	
	Input high voltage, RSTB, T0CKI Pins	3V	With Schmitt-trigger	-	1.79	V_{DD}	
		5V		-	3.31	V_{DD}	
V_{IH2}	Input high voltage, I/O Ports	3V	Without Schmitt-trigger	-	1.19	V_{DD}	V
		5V		-	1.58	V_{DD}	
	Input high voltage, RSTB, T0CKI Pins	3V	Without Schmitt-trigger	-	1.79	V_{DD}	
		5V		-	3.31	V_{DD}	
V_{IL1}	Input low voltage, I/O Ports	3V	With Schmitt-trigger	V_{SS}	1.0	-	V
		5V		V_{SS}	-	1.0	
	Input low voltage, RSTB, T0CKI Pins	3V	With Schmitt-trigger	V_{SS}	1.15	-	
		5V		V_{SS}	1.49	-	
V_{IL2}	Input low voltage, I/O Ports	3V	Without Schmitt-trigger	V_{SS}	1.09	-	V
		5V		V_{SS}	1.5	-	
	Input low voltage, RSTB, T0CKI Pins	3V	Without Schmitt-trigger	V_{SS}	1.15	-	
		5V		V_{SS}	1.5	-	

Symbol	Description	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
V_{LVDT}	LVDT voltage	-	LVDT=3.6V	3.06	3.6	4.14	V
		-	LVDT=2.6V	2.21	2.6	2.99	
		-	LVDT=2.4V	2.04	2.4	2.76	
		-	LVDT=2.2V	1.87	2.2	2.53	
		-	LVDT=2.0V	1.7	2.0	2.3	
		-	LVDT=1.8V	1.6	1.8	2.07	
I_{OH}	I/O Ports Drive current (without A type IOA5)	3V	V_{OH}=0.9V_{DD}	-	1.74	-	mA
		5V		1.5	4.54	-	
	A type IOB0, B type IOA4 IR-out mode Drive current	3V	V_{OH}=0.9V_{DD}	-	3.38	-	
		5V		-	8.55	-	
I_{OL}	I/O Ports Sink current	3V	V_{OL}=0.1V_{DD}	-	9.69	-	mA
		5V		10	22.74	-	
	A type IOB0, B type IOA4 IR-out mode Sink current	3V	V_{OL}=0.1V_{DD}	-	18.67	-	
		5V		-	42.74	-	
I_{PH}	I/O Ports Pull-high current	3V	Input pin at V_{SS}	-	19.46	-	uA
		5V		55	65.03	85	
I_{PL}	I/O Ports Pull-low current	3V	Input pin at V_{DD}	-	12.64	-	uA
		5V		30	41.33	60	
I_{ROC}	IOC0 & IOC1 ROC mode Pull-high current	3V	Input pin at V_{SS}	-	0.31	-	uA
		5V		-	1.05	-	
I_{LVDT}	LVDT current	5V	LVDT=3.6V	-	1.09	-	uA
		3V	LVDT=2.6V	-	0.4	-	
		5V		-	1.4	-	
		3V	LVDT=2.4V	-	0.5	-	
		5V		-	1.5	-	
		3V	LVDT=2.2V	-	0.5	-	
		5V		-	1.6	-	
		3V	LVDT=2.0V	-	0.6	-	
		5V		-	1.7	-	
		3V	LVDT=1.8V	-	0.6	-	
		5V		-	1.9	-	
I_{WDT}	WDT current	3V	Sleep mode, Pre-scaler rate=1:256	-	0.5	-	uA
		5V		-	3.8	8	
I_{SB}	Sleep mode (Power down) current	3V	-	-	<1	-	uA
		5V		-	<1	1	
I_{DD1}	HF Operating current	3V	Freq=20M_{HZ}, 2T	-	1.78	-	mA
		5V		-	4.1	-	
I_{DD2}	XT Operating current	3V	Freq=16M_{HZ}, 2T	-	0.96	-	mA
		5V		-	2.27	-	
		5V	Freq=20M_{HZ}, 2T	-	3.04	-	
I_{DD3}	LF Operating current	3V	Freq=32K_{HZ}, 2T	-	45.82	-	uA
		5V		-	129.83	-	
I_{DD4}	ERC Operating current Rext=3.3K, Cext=3pF	3V	Freq=13.05M_{HZ}, 2T	-	1.8	-	mA
		5V	Freq=15.31M_{HZ}, 2T	-	3.5	-	
I_{DD5}	ERIC Operating current	3V	Freq=8M_{HZ}, 2T	-	0.79	-	mA
		5V		-	1.51	-	
		5V	Freq=16M_{HZ}, 2T	-	2.87	-	
I_{DD6}	Operating current	3V	IRC 8M_{HZ}, 2T	-	0.79	-	mA
		5V		-	1.52	-	
I_{DD7}	Operating current	3V	IRC 4M_{HZ}, 2T	-	0.42	-	mA
		5V		-	0.8	-	

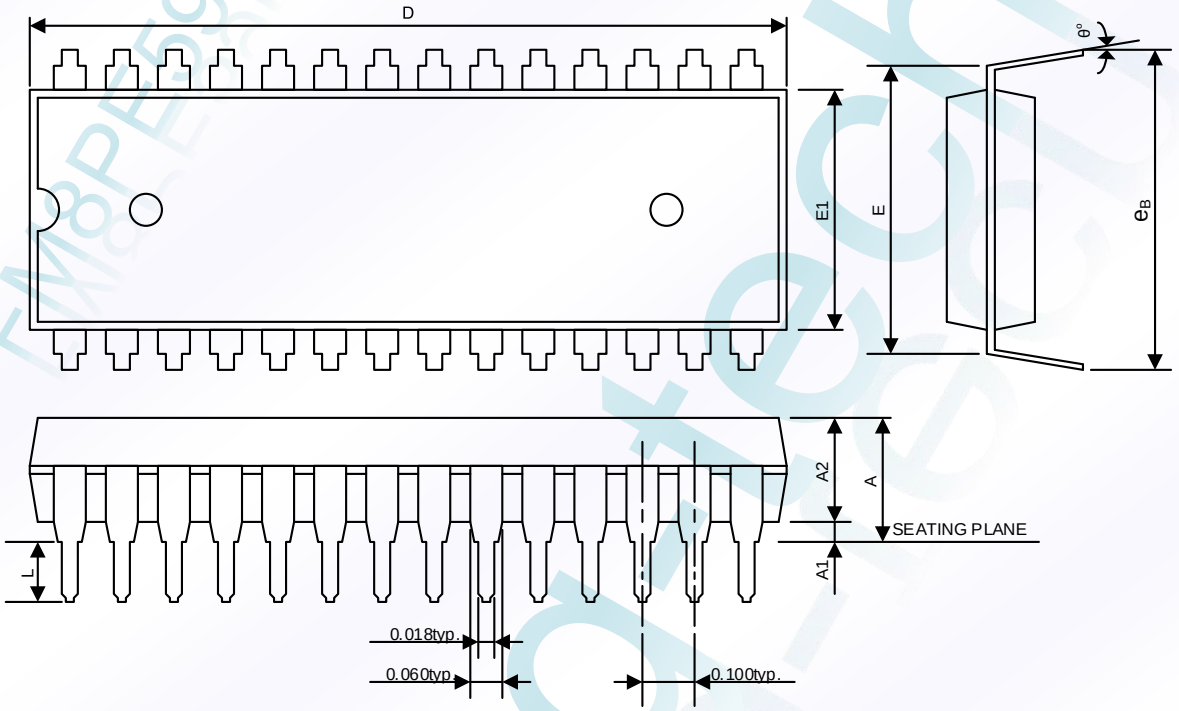
Symbol	Description	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
I_{DD8}	Operating current	3V	IRC 1M$_{HZ}$, 2T	-	141	-	uA
		5V		-	297.2	-	
I_{DD9}	Operating current	3V	IRC 455K$_{HZ}$, 2T	-	88.6	-	uA
		5V		-	202.8	-	

6.3 ELECTRICAL CHARACTERISTICS Charts of FM8PE59M

To be define…

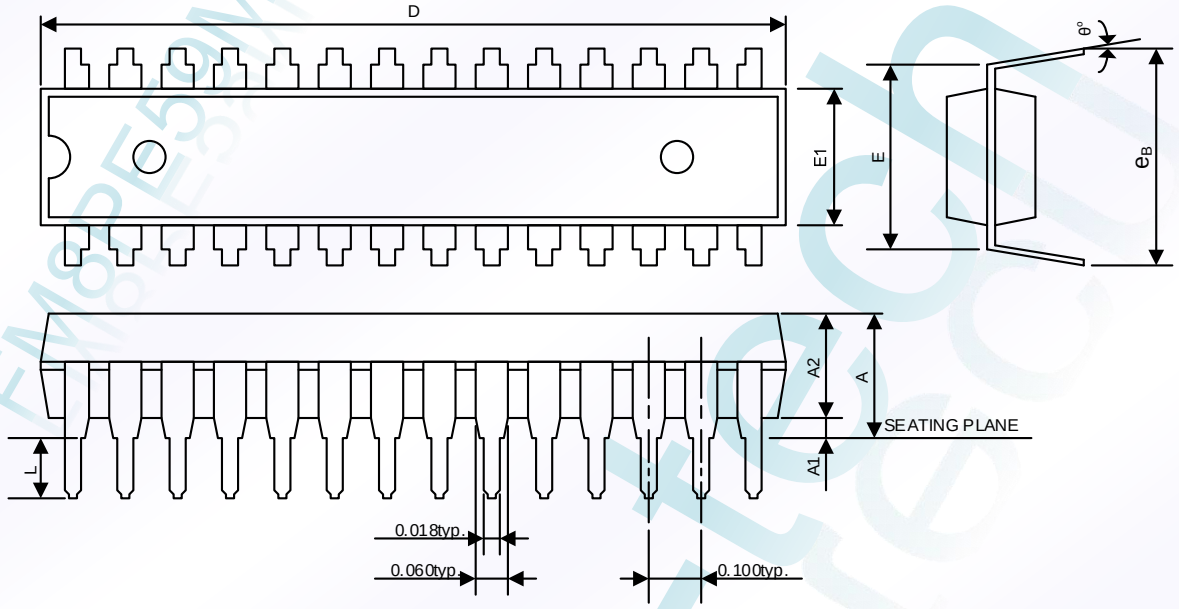
7.0 PACKAGE DIMENSION

7.1 28-PIN PDIP 600mil



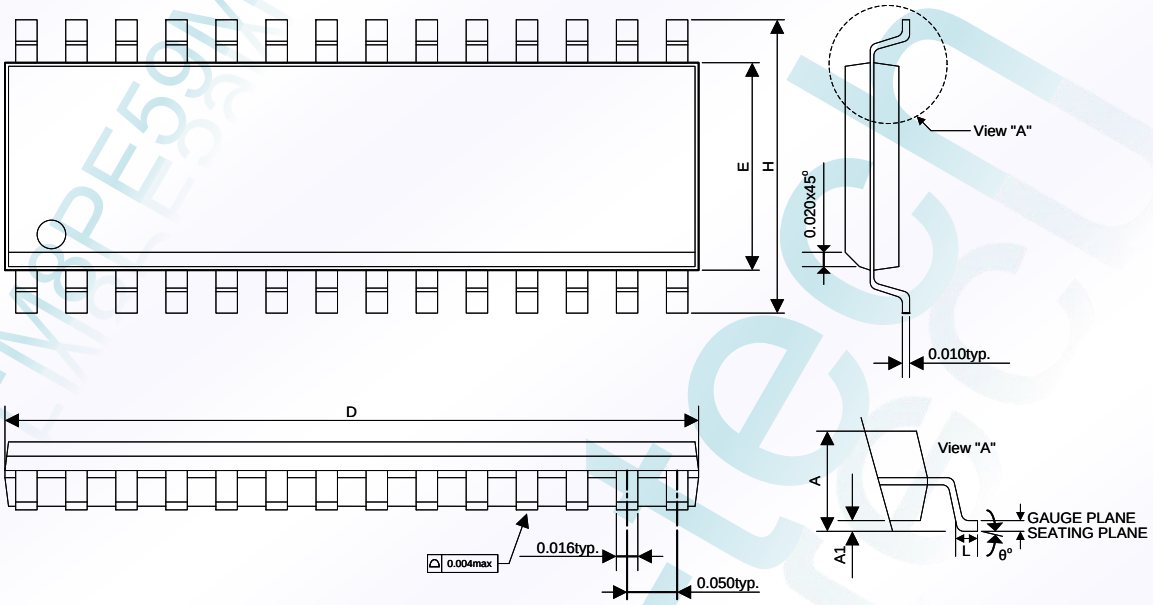
Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.220
A1	0.015	-	-
A2	0.150	0.155	0.160
D	1.455	1.460	1.470
E		0.600 BSC.	
E1	0.540	0.545	0.550
L	0.115	0.158	0.200
eB	0.630	0.650	0.670
θ°	0°	7°	15°

7.2 28-PIN Skinny-PDIP 300mil



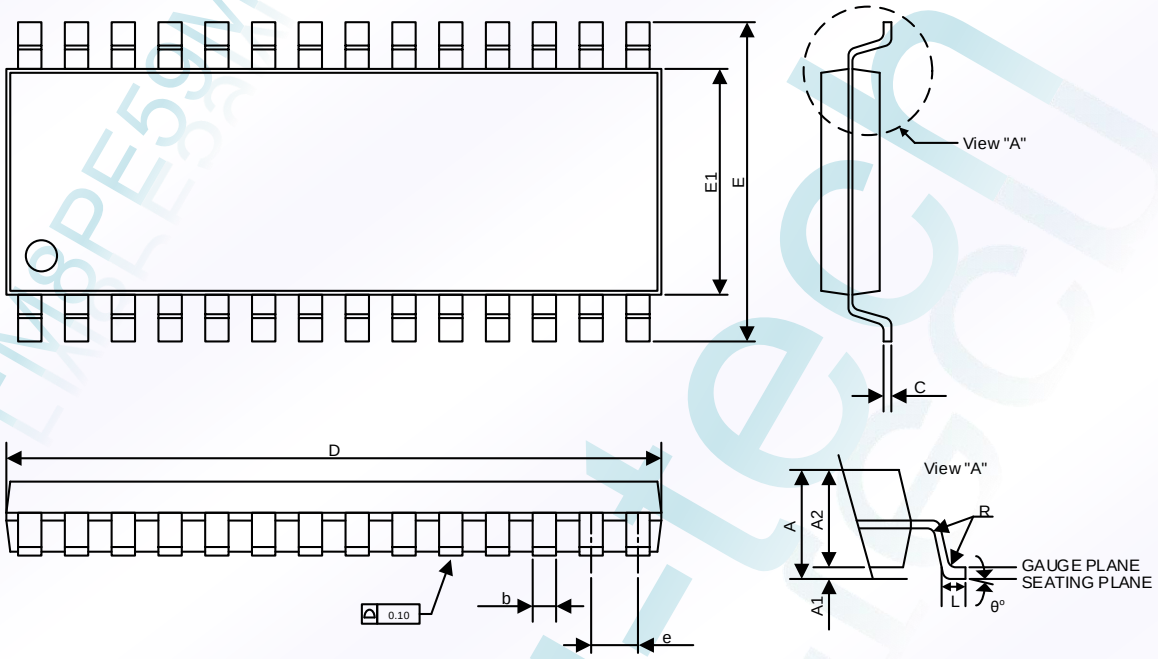
Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.175
A1	0.015	-	-
A2	0.125	0.130	0.135
D	1.385	1.390	1.400
E		0.310 BSC.	
E1	0.283	0.288	0.293
L	0.120	0.130	0.140
eB	0.330	0.350	0.370
θ°	0°	7°	15°

7.3 28-PIN SOP 300mil



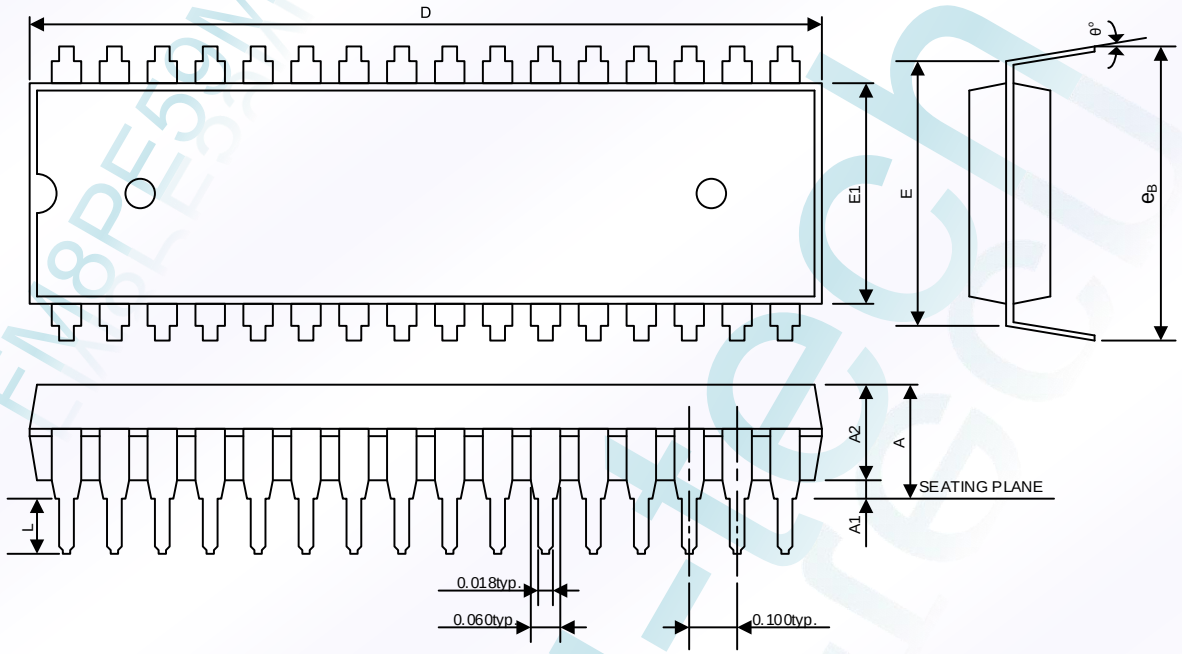
Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.104
A1	0.004	-	-
D	0.697	0.718	0.724
E	0.291	0.295	0.299
H	0.394	0.406	0.419
L	0.016	0.035	0.050
θ	0°	4°	8°

7.4 28-PIN SSOP 209mil



Symbols	Dimension In MM		
	Min	Nom	Max
A	-	-	2.0
A1	0.05	-	-
A2	1.65	1.75	1.85
b	0.22	-	0.38
c	0.09	-	0.25
D	10.05	10.20	10.50
E	7.65	7.80	7.90
E1	5.00	5.30	5.60
e	0.65 BSC		
L	0.55	0.75	0.95
R	0.09	-	-
θ	0°	4°	8°

7.5 32-PIN PDIP 600mil



Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.220
A1	0.015	-	-
A2	0.150	0.155	0.160
D	1.645	1.650	1.660
E		0.600 BSC.	
E1	0.540	0.545	0.550
L	0.115	0.158	0.200
eB	0.630	0.650	0.670
θ°	0°	7°	15°

8.0 ORDERING INFORMATION

OTP Type MCU	Package Type	Pin Count	Package Size	MOQ	MSL	Sample Stock
FM8PE59MAP	PDIP	28	600mil	3,000EA/Tube	3	Call sales
FM8PE59MAM	SKINNY-PDIP	28	300mil	3,000EA/Tube	3	Available
FM8PE59MAD	SOP	28	300mil	3,000EA/Tube 1,000EA/Reel*3	3	Available
FM8PE59MAR	SSOP	28	209mil	3,000EA/Tube	3	Available
FM8PE59MBP	PDIP	32	600mil	3,000EA/Tube	3	Call sales