FEELING TECHNOLOGY

FM8PE68B

OTP-Based 8-Bit Microcontroller with LCD Driver

Devices Included in this Data Sheet:

- FM8PE68BA/BB: 64-pin OTP device
- FM8PE68BC: 44-pin OTP device

FEATURES

- Only 49 single word instructions.
- 4K Word on chip OTP.
- All OTP area GOTO/FGOTO instruction.
- All OTP area subroutine CALL/FCALL instruction.
- Totally 272 x 8 bits on chip general purpose registers (SRAM):
  - 144 bytes general purpose register.
  - 128 bytes on-chip data RAM.
- 8-level deep hardware stack.
- Direct, indirect addressing modes for data accessing.
- One 8-bit real time clock/counter (Timer0) with 8-bit programmable pre-scaler.
- Four sets of 8-bit auto reload counter/timer can be used as IROUT/PWM generator or interrupt sources:
  - Counter 1: independent counter.
  - Counter 2: High Pulse Width Timer, and Low Pulse Width Timer shared with IR function.
- One IROUT/PWM generator.
- LCD driver with 4(common) x 32(segment) pixels; 1/3, 1/2 bias and 1/4, 1/3, 1/2 duty selection.
- 3 channels of 15-bit resolution Resistor to Frequency Converter (RFC) output.
- 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 and soft-ware watch-dog enable/disable control.
- Four I/O ports PORTA, PORTB, PORTC and PORTD with independent direction control.
- Two output only ports PORTE and PORTF.
- Soft-ware I/O pull-high/pull-down or open-drain control.
- Seven internal interrupt source: Timer0, Counter1, Counter2, High-pulse width timer, Low-pulse width timer, RFC and Fs divider; Three external interrupt source: INT0 pin, INT1 pin and Port B / Port D input status change.
- Wake-up from SLEEP/IDLE by Port B/Port D input change.
- Operation modes:
  - Normal mode: CPU operate on high frequency main-oscillator.
  - Green mode: CPU operate on low frequency sub-oscillator.
  - Idle mode: CPU idle, LCD display remains working.
  - Sleep mode: whole chip stop working.
- Dual clock Operation: main-oscillator and sub-oscillator.
- Selectable main-oscillator options:
  - ERIC: External Resistor/Internal Capacitor Oscillator.
  - XT: Crystal/Resonator Oscillator.
  - LF: Low Frequency Crystal/Resonator Oscillator.
  - PLL: Phase lock loop.
- Selectable sub-oscillator options:
  - ERIC: External Resistor/Internal Capacitor Oscillator.
  - LF: Low Frequency Crystal Oscillator.
- Wide-operating voltage range:
  - OTP: 2.3V to 5.5V.

This datasheet contains new product information. Feeling Technology reserves the rights to modify the product specification without notice. No liability is assumed as a result of the use of this product. No rights under any patent accompany the sales of the product.

GENERAL DESCRIPTION

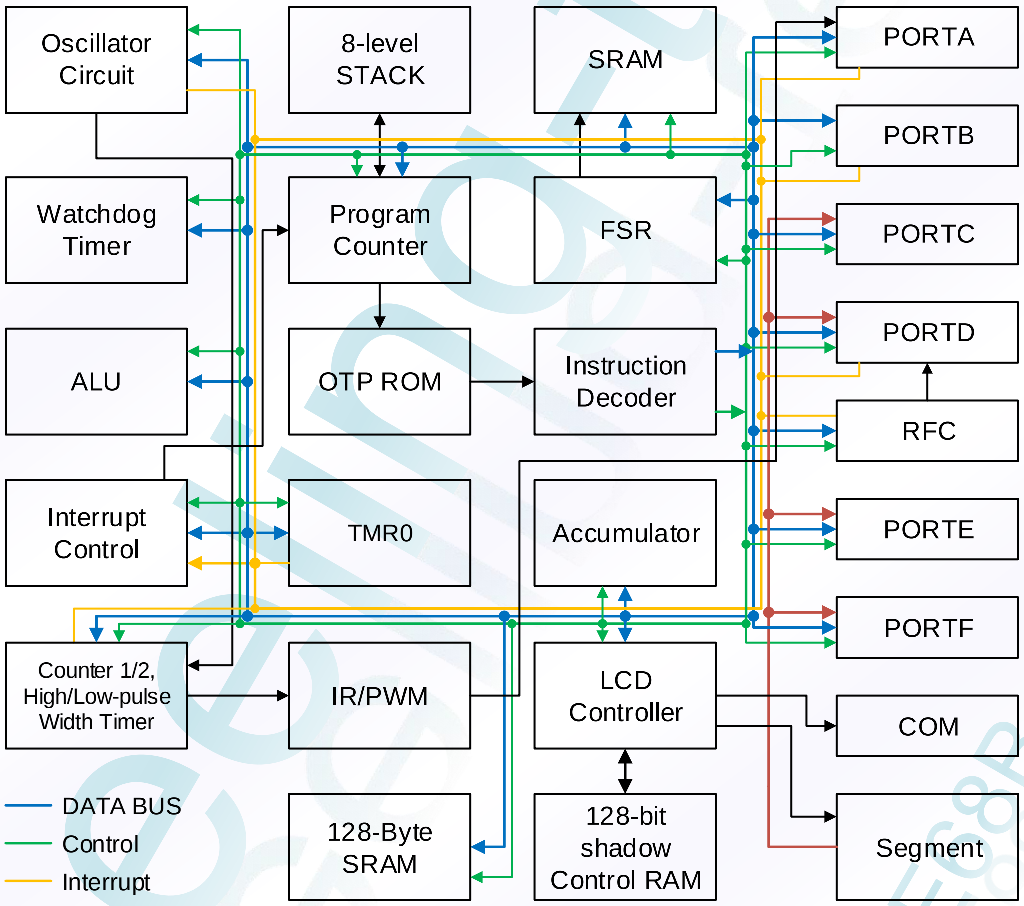
The FM8PE68B is a family of low-cost, high speed, OTP-based 8-bit CMOS microcontrollers. It employs a RISC architecture with only 49 instructions. All instructions are single cycle except for program branches which take two cycles. The easy to use and easy to remember instruction set reduces development time significantly.

The FM8PE68B consists of Power-on Reset (POR), Brown-out Reset (BOR), Power-up Reset Timer (PWRT), Oscillator Start-up Timer(OST), Watchdog Timer, Data RAM, OTP/ROM, SRAM, LCD driver, IROUT function, tristate I/O port, I/O pull-high/open-drain/pull-down control, Power saving SLEEP mode, real time programmable clock/counter, Interrupt, Wake-up from SLEEP mode, RFC, and Code Protection for OTP products. There are three oscillator configurations to choose from, including the power-saving LP (Low Power) oscillator and cost saving RC oscillator.

The FM8PE68B address 4K of program memory.

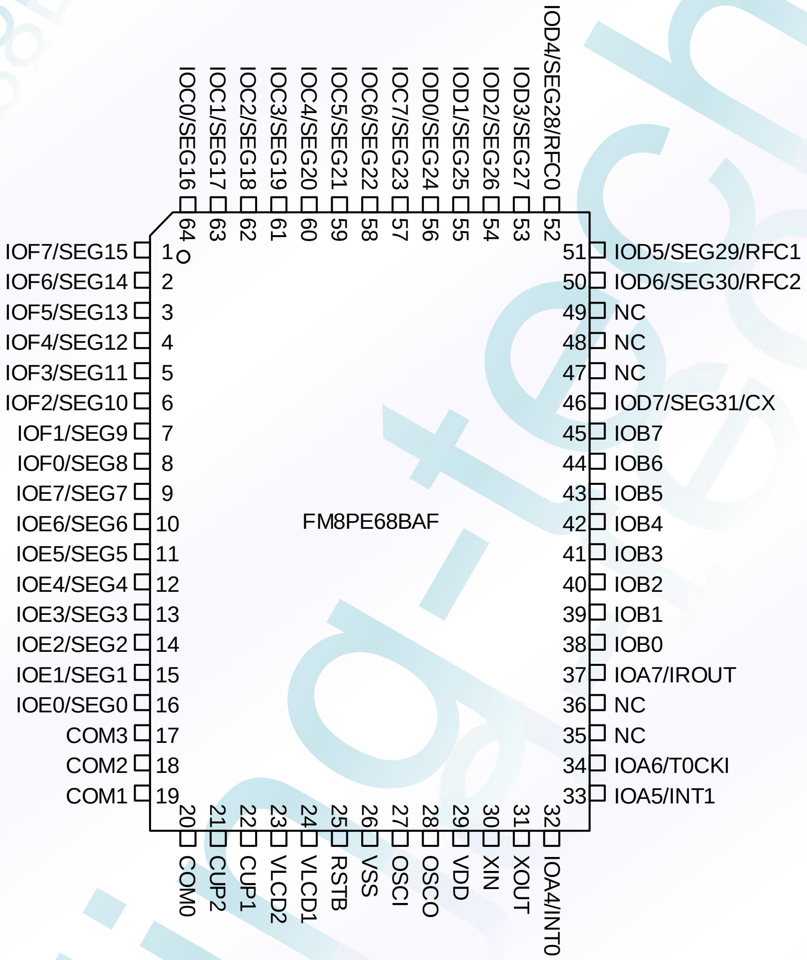
The FM8PE68B can directly or indirectly address its register files and data memory. All special function registers including the program counter are mapped in the data memory.

BLOCK DIAGRAM

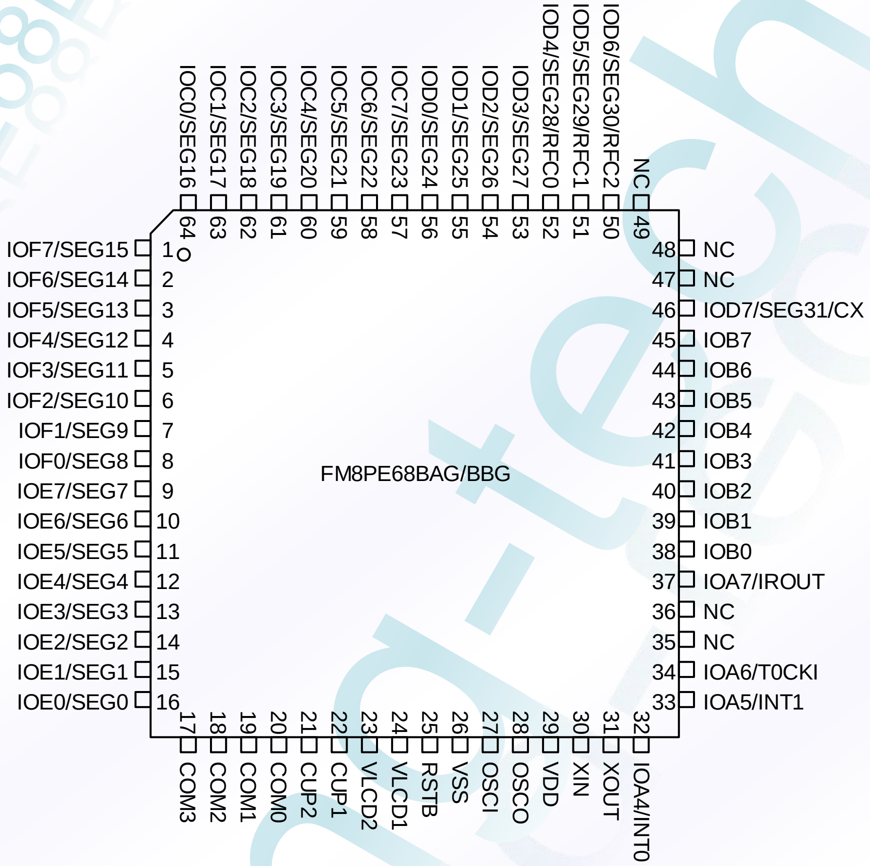


PIN CONNECTION

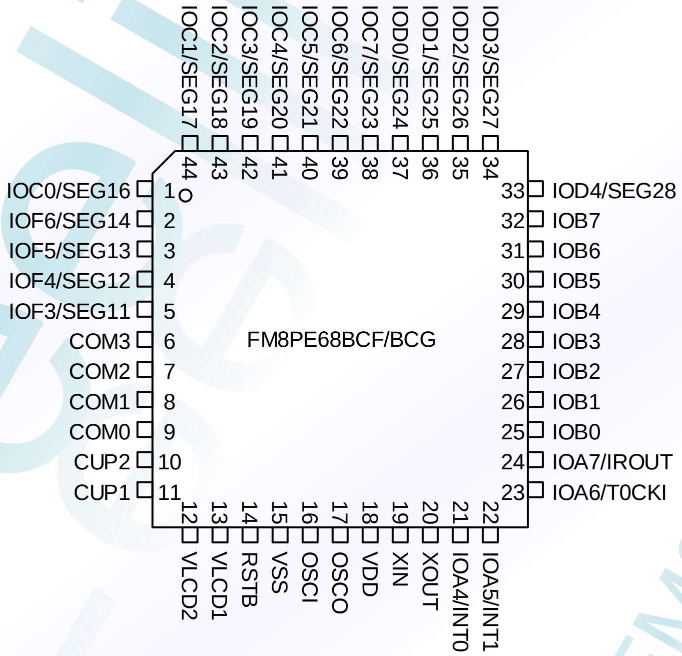
QFP64 (14x20)



LQFP64 (10x10) / LQFP64 (7x7)



QFP44 (10x10) / LQFP44 (10x10)



PIN DESCRIPTIONS

Name	I/O	Description
IOA4/INT0	I/O	• Bi-direction I/O pin. • External interrupt input 0, the trigger edge is controlled by INT0EDG bit.
IOA5/INT1	I/O	• Bi-direction I/O pin. • External interrupt input 1 with falling edge trigger.
IOA6/T0CKI	I/O	• Bi-direction I/O pin. • Clock input to Timer0.
IOA7/IROUT	I/O	• Bi-direction I/O pin. • IR mode output pin.
IOB0 ~ IOB7	I/O	• Bi-direction I/O port with system wake-up function.
IOC0 ~ IOC7	I/O	• Bi-direction I/O port.
IOD0 ~ IOD7	I/O	• Bi-direction I/O port with system wake-up function.
IOE0 ~ IOE7	O	• Output only pins.
IOF0 ~ IOF7	O	• Output only pins.
COM0 ~ COM3	O	• LCD common output pins.
SEG0 ~ SEG31	O	• LCD segment output pins.
VLCD1	-	• One of LCD bias voltage.
VLCD2	-	• One of LCD bias voltage.
CUP1	-	• Connect capacitors for LCD bias voltage.
CUP2	-	• Connect capacitors for LCD bias voltage.
RFC0 ~ RFC2	O	• The RC oscillator network output of RFC module
CX	I	• The RC oscillator network input of RFC module
RSTB	I	• System clear (RESET) input. This pin is an active low RESET to the device.
OSCI	I	• Main Oscillator: - X'tal type: Oscillator crystal input. - ERIC type: Clock input of RC oscillator. - PLL type: Connect 0.01uF capacitor to V_{SS}.
OSCO	O	• Main Oscillator: - X'tal type: Oscillator crystal output. -ERIC and PLL type: Instruction clock output.
XIN	I	• Sub-Oscillator: - X'tal type: 32.768K_{HZ} Oscillator crystal input. - ERIC type: Clock input of RC oscillator.
XOUT	O	• Sub-Oscillator: - X'tal type: 32.768K_{HZ} Oscillator crystal output. - ERIC type: Instruction clock output.
V_{DD}	-	Positive supply
V_{SS}	-	Ground

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

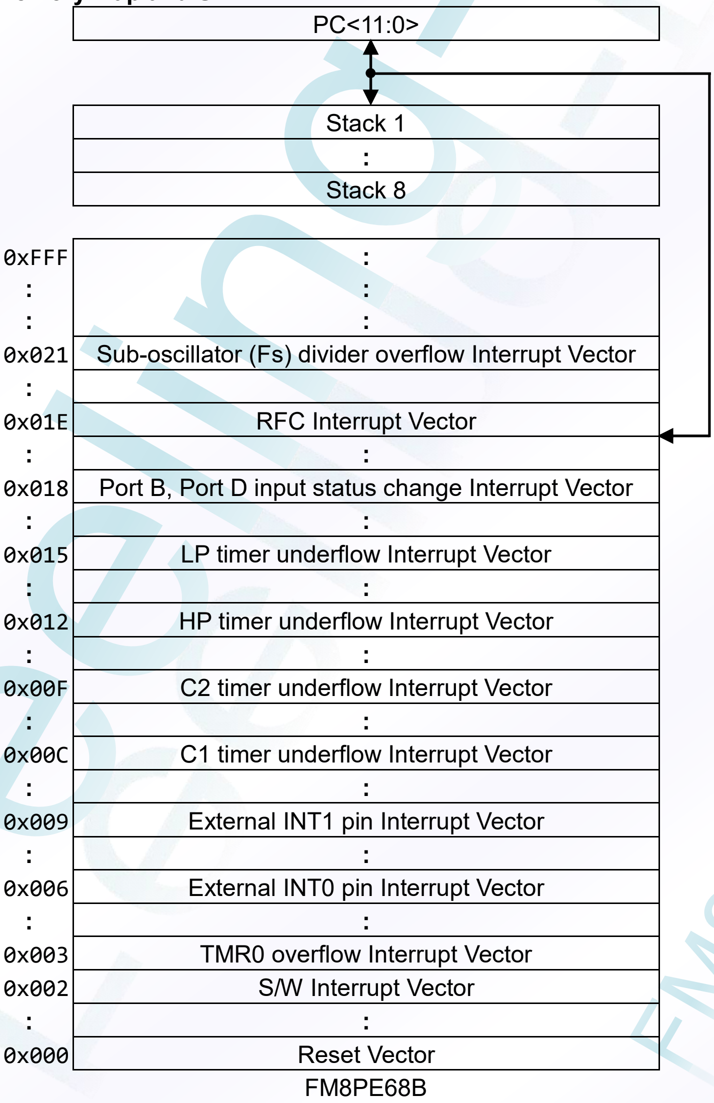
1.0 MEMORY ORGANIZATION

FM8PE68B memory is organized into program memory and data memory.

1.1 Program Memory Organization

The FM8PE68 have an 12-bit Program Counter capable of addressing a 4K program memory space.
The RESET vector for the FM8PE68B is at 0x000.
The H/W interrupt vector is at 0x003/0x006/0x009/0x00C/0x00F/0x012/0x015/0x018/0x01E/0x021 based on different H/W interrupt event. And the S/W interrupt vector is at 0x002.
FM8PE68B 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 FM8PE68B, there is another two bits to specify the program memory page. This paging bit comes from the STATUS<6:5> bits. When doing a CALL or GOTO instruction, the user must ensure that page bit STATUS<6:5> 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 STATUS<6:5> is not required for the return instructions.
User can use "**PAGE**" instruction to change memory page directly and maintains the program memory page. Otherwise, user can use "**FCALL**(far call)/**FGOTO**(far goto)" instructions to program user's code directly.

Figure 1.1: Program Memory Map and STACK


PC<11:0>
Stack 1
:
Stack 8
0xFFF
:
:
:
:
:
0x021 Sub-oscillator (Fs) divider overflow Interrupt Vector
:
0x01E RFC Interrupt Vector
:
:
0x018 Port B, Port D input status change Interrupt Vector
:
:
0x015 LP timer underflow Interrupt Vector
:
:
0x012 HP timer underflow Interrupt Vector
:
:
0x00F C2 timer underflow Interrupt Vector
:
:
0x00C C1 timer underflow Interrupt Vector
:
:
0x009 External INT1 pin Interrupt Vector
:
:
0x006 External INT0 pin Interrupt Vector
:
:
0x003 TMR0 overflow Interrupt Vector
0x002 S/W Interrupt Vector
:
:
0x000 Reset Vector
FM8PE68B

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 FM8PE68B, 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 FM8PE68B

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			
0x03	STATUS			
0x04	FSR			
0x05	PORTA			
0x06	PORTB			
0x07	PORTC			
0x08	PORTD			
0x09	LCDCON	PORTE	LCDCON	PORTE
0x0A	LCDA	PORTF	LCDA	PORTF
0x0B	LCDD	RFCCON	LCDD	RFCCON
0x0C	CNTCON	RFCDL	CNTCON	RFCDL
0x0D	SYSCON	RFCDH	SYSCON	RFCDH
0x0E	IRCON	DIVCON	IRCON	DIVCON
0x0F	INTFLAG	Memory back to address in Bank 0		
0x10 | 0x1F	General Purpose Registers			
0x20 | 0x3F	General Purpose Registers	General Purpose Registers	General Purpose Registers	General Purpose Registers

0x05	IOSTA
0x06	IOSTB
0x07	IOSTC
0x08	IOSTD
0x09	DRAMA
0x0A	DRAMD
0x0B	C1PR
0x0C	C2PR
0x0D	HPPR
0x0E	LPPR
0x0F	INTEN
0x15	SEGCON
0x16	WUCON
0x17	T0CON
0x18	WDTCON
0x19	C12CON
0x1A	HLPCON
0x1B	BPHCON
0x1C	BODCON
0x1D	DPHCON
0x1E	BPDCON
0x1F	INTEN1

Table 1.2: 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	*	PG1	PG0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
0x04 (r/w)	FSR	RP1	RP0	Indirect data memory address pointer					
0x05 (r/w)	PORTA	IOA7	IOA6	IOA5	IOA4	-	-	*	-
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)	PORTD	IOD7	IOD6	IOD5	IOD4	IOD3	IOD2	IOD1	IOD0
Bank 0, Bank 2									
0x09 (r/w)	LCDCON	BIAS	DUTY1	DUTY0	LCDEN	-	TYPE	LCDF1	LCDF0
0x0A (r/w)	LCDA	0	0	0	LCDA4	LCDA3	LCDA2	LCDA1	LCDA0
0x0B (r/w)	LCDD	0	0	0	0	LCDD3	LCDD2	LCDD1	LCDD0
0x0C (r/w)	CNTCON	0	0	0	0	LPEN	HPEN	C2EN	C1EN
0x0D (r/w)	SYSCON	0	PLLCK2	PLLCK1	PLLCK0	IDLE	LCDBF1	LCDBF0	CPUS
0x0E (r/w)	IRCON	IRE	HF	LGP	-	EIROUT	ET0CKI	EINT1	EINT0
Bank 1, Bank 3									
0x09 (r/w)	PORTE	IOE7	IOE6	IOE5	IOE4	IOE3	IOE2	IOE1	IOE0
0x0A (r/w)	PORTF	IOF7	IOF6	IOF5	IOF4	IOF3	IOF2	IOF1	IOF0
0x0B (r/w)	RFCCON	RFCON	START	RFCIF	RFCMOD	-	-	RFCS1	RFCS0
0x0C (r)	RFCDL	RFCD7	RFCD6	RFCD5	RFCD4	RFCD3	RFCD2	RFCD1	RFCD0
0x0D (r)	RFCDH	RFCOV	RFCD14	RFCD13	RFCD12	RFCD11	RFCD10	RFCD9	RFCD8
0x0E (r/w)	DIVCON	DIVON	DIVRST	DIVIF	-	-	-	-	-
Unbanked									
0x0F (r/w)	INTFLAG	PBDIF	LPIF	HPIF	C2IF	C1IF	INT1IF	INT0IF	T0IF

Legend: - = unimplemented, read as ‘0’, * = unimplemented, read as ‘1’, 0 = Not used, must fix to ‘0’.

Table 1.3: The Registers Controlled by IOST / IOSTR Instructions

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x05 (r/w)	IOSTA	IOSTA7	IOSTA6	IOSTA5	IOSTA4	-	-	-	-
0x06 (r/w)	IOSTB	IOSTB7	IOSTB6	IOSTB5	IOSTB4	IOSTB3	IOSTB2	IOSTB1	IOSTB0
0x07 (r/w)	IOSTC	IOSTC7	IOSTC6	IOSTC5	IOSTC4	IOSTC3	IOSTC2	IOSTC1	IOSTC0
0x08 (r/w)	IOSTD	IOSTD7	IOSTD6	IOSTD5	IOSTD4	IOSTD3	IOSTD2	IOSTD1	IOSTD0
0x09 (r/w)	DRAMA	0	RAMA6	RAMA5	RAMA4	RAMA3	RAMA2	RAMA1	RAMA0
0x0A (r/w)	DRAMD	RAMD7	RAMD6	RAMD5	RAMD4	RAMD3	RAMD2	RAMD1	RAMD0
0x0B (r/w)	C1PR	C1PR7	C1PR6	C1PR5	C1PR4	C1PR3	C1PR2	C1PR1	C1PR0
0x0C (r/w)	C2PR	C2PR7	C2PR6	C2PR5	C2PR4	C2PR3	C2PR2	C2PR1	C2PR0
0x0D (r/w)	HPPR	HPPR7	HPPR6	HPPR5	HPPR4	HPPR3	HPPR2	HPPR1	HPPR0
0x0E (r/w)	LPPR	LPPR7	LPPR6	LPPR5	LPPR4	LPPR3	LPPR2	LPPR1	LPPR0
0x0F (r/w)	INTEN	PBDIE	LPIE	HPIE	C2IE	C1IE	INT1IE	INT0IE	T0IE
0x15 (r/w)	SEGCON	IOFHS	IOFLS	IOEHS	IOELS	IODHS	IODLS	IOCHS	IOCLS
0x16 (r/w)	WUCON	IRSC	*	*	*	/WUEDH	/WUEDL	/WUEBH	/WUEBL
0x17 (r/w)	T0CON	INT0EDG	GIE	T0CS	T0SE	T0PS3	T0PS2	T0PS1	T0PS0
0x18 (r/w)	WDTCON	*	*	*	*	WDTEN	WDTPS2	WDTPS1	WDTPS0
0x19 (r/w)	C12CON	C2CS	C2PS2	C2PS1	C2PS0	C1CS	C1PS2	C1PS1	C1PS0
0x1A (r/w)	HLPCON	LPCS	LPPS2	LPPS1	LPPS0	HPCS	HPPS2	HPPS1	HPPS0
0x1B (r/w)	BPHCON	PHB7	PHB6	PHB5	PHB4	PHB3	PHB2	PHB1	PHB0
0x1C (r/w)	BODCON	ODB7	ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0
0x1D (r/w)	DPHCON	PHD7	PHD6	PHD5	PHD4	PHD3	PHD2	PHD1	PHD0
0x1E (r/w)	BPDCON	PDB7	PDB6	PDB5	PDB4	PDB3	PDB2	PDB1	PDB0
0x1F (r/w)	INTEN1	-	-	-	-	-	DIVIE	RFCIE	-

Legend: - = unimplemented, read as ‘0’, * = unimplemented, read as ‘1’, 0 = Not used, must fix to ‘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 2.9.

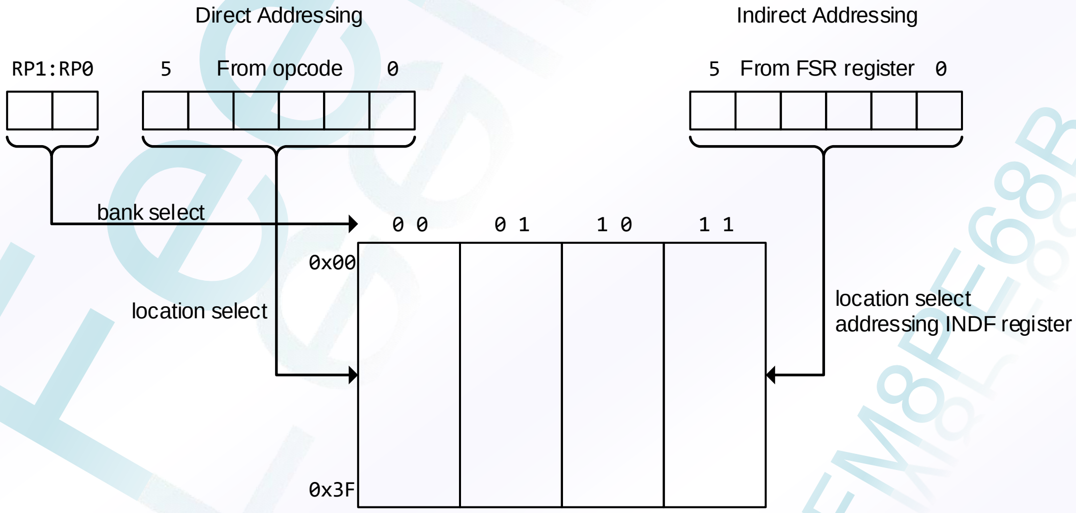
The INDF Register is not a physical register. Any instruction accessing the INDF register can actually access the register pointed by FSR Register. Reading the INDF register itself indirectly (FSR=”0”) will read 00h. Writing to the INDF register indirectly results in a no-operation (although status bits may be affected).
The bits 5-0 of FSR register are used to select up to 64 registers (address: 0x00 ~ 0x3F).
In FM8PE68B, 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.

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

Example 2.1: INDIRECT ADDRESSING

- Register file 38 contains the value 0x10
- Register file 39 contains the value 0x0A
- Load the value 38 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 FM8PE68B



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

Note: more bits default state, please refer to Table 2.9.

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 (T0CON<5>). If T0CKI pin is selected, the Timer0 is increased by T0CKI signal rising/falling edge (selected by T0SE bit (T0CON<4>)).
Please note, the pre-scaler will be cleared when TMR0 register is written with a value.

2.1.3 PCL (Low Bytes of Program Counter) & Stack

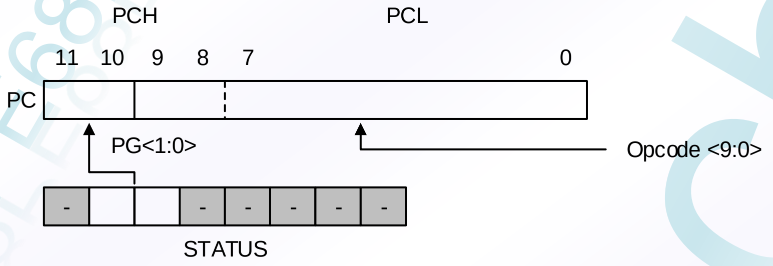
Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x02	PCL	Low order 8 bits of PC							

Note: more bits default state, please refer to Table 2.9.

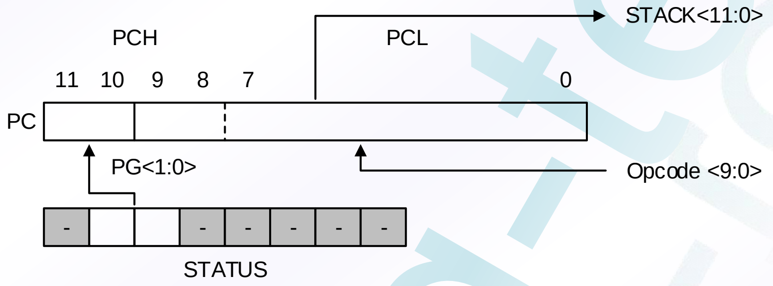
FM8PE68B devices have a 12-bit wide Program Counter (PC) and eight-level deep 12-bit hardware push/pop stack. The low byte of PC is called the PCL register. This register is readable and writable. The high byte of PC is called the PCH register. This register contains the PC<11:8> bits and is not directly readable or writable. All updates to the PCH register go through the PG<1:0> bits (STATUS<6:5>). 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 PG<1:0> bits (STATUS<6:5>). The PCL register is mapped to PC<7:0>.
For a CALL instruction, the PC<9:0> is provided by the CALL instruction word. The PC<11:10> is updated from the PG<1:0> bits (STATUS<6:5>). The next PC will be loaded (PUSHed) into the top of STACK. The PCL register is mapped to PC<7:0>.
For a FGOTO instruction, the PC<11:0> is provided by the FGOTO instruction word. The PCL register is mapped to PC<7:0>, and the PG<1:0> bits is also updated from the FGOTO instruction word.
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>, and the PG<1:0> bits is also updated from the FCALL instruction word.
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>.
For any instruction where the PCL is the destination, the PC<7:0> is provided by the instruction word or ALU result, and the PC<9:8> will be not changed. The PG<1:0> bits whether to update to the PC<11:10>, It can be decision by configure-word PCHS bit.

Figure 2.2: Loading of PC in Different Situations

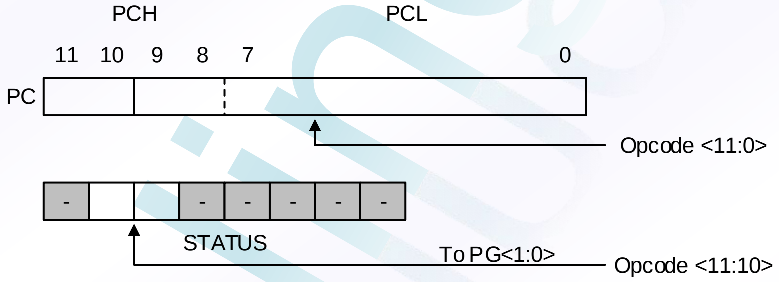
Situation 1: **GOTO** Instruction



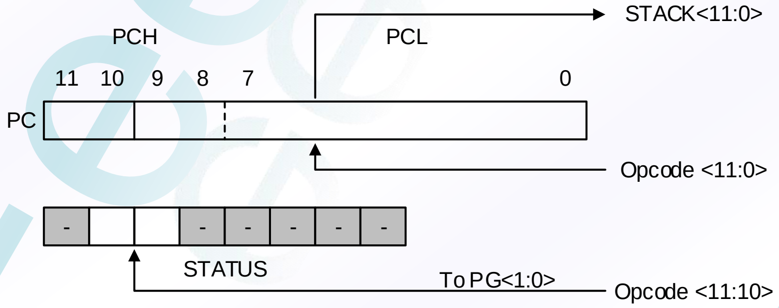
Situation 2: **CALL** Instruction



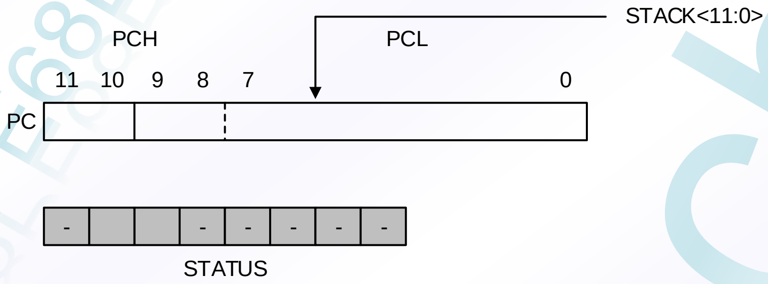
Situation 3: **FGOTO** Instruction



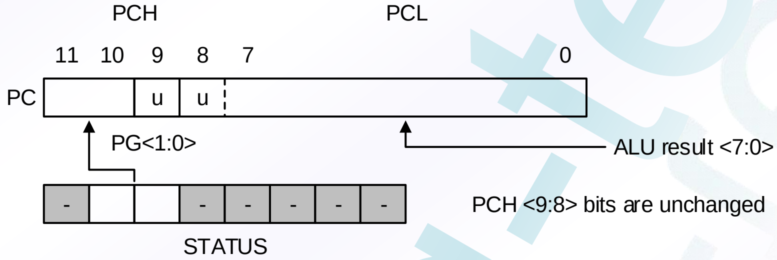
Situation 4: **FCALL** Instruction



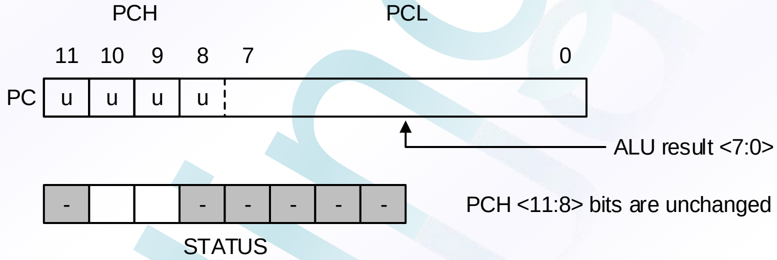
Situation 5: **RETIA**, **RETFIE**, or **RETURN** Instruction



Situation 6: Instruction with PCL as destination (Configuration bit **PCHS** is select to **PC<11:10>=PG<1:0>**)



Situation 7: Instruction with PCL as destination (Configuration bit **PCHS** is select to **Unchanged**)



2.1.4 STATUS (Status Register)

Read/Write-POR		*	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	*	PG1	PG0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C

Legend: * = unimplemented, read as ‘1’, x = unknown, # = refer Table 2.10 for detail description, more bits default state, please refer to Table 2.9.

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 $\overline{\text{TO}}$ and $\overline{\text{PD}}$ 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 100u u1uu (where u = unchanged).

C: Carry/borrow bit.
ADDAR, ADDIA:
= 0, No Carry occurred.
= 1, Carry occurred.
SUBAR, SUBIA:
= 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:
= 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:
= 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.

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

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

PG1:PG0: Program memory page select bits. Used for GOTO, CALL, or any instruction with PCL as destination.

PG1:PG0		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 page and maintains the program page. Otherwise, user can use "FGOTO" (far goto), or "FCALL" (far call) instructions to program user's code. It changes the user's program by inserting instructions within the program.

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 2.9.

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. See 2.1.1 for detail description.

2.1.6 PORTA, PORTB, PORTC & PORTD (Port Data Registers)

Read/Write-POR		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	-	-	*	-

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

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
0x08	PORTD	IOD7	IOD6	IOD5	IOD4	IOD3	IOD2	IOD1	IOD0

Legend: - = unimplemented, read as '0', * = unimplemented, read as '1', more bits default state, please refer to Table 2.9.

Reading the port (PORTA, PORTB, PORTC and PORTD 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.
For FM8PE68B devices, PORTA is a 4-bit port data Register. Only the high order 4 bits are used (PORTA<7:4>) and bits 3-0 are unimplemented and read as '0's.
All of PORTB, PORTC and PORTD are 8-bit port data registers.

2.1.7 LCDCON (LCD Control Register) (Bank 0, 2)

Read/Write-POR		R/W-1	R/W-1	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	LCDCON	BIAS	DUTY1	DUTY0	LCDEN	-	TYPE	LCDF1	LCDF0

Legend: - = unimplemented, read as '0', more bits default state, please refer to Table 2.9.

LCDF1:LCDF0: LCD frame frequency select bits

LCDF1	LCDF0	LCD frame frequency (F_S=32.768KHZ)		
		1/2 duty	1/3 duty	1/4 duty
0	0	F_S/(256*2)=64.0	F_S/(172*3)=63.5	F_S/(128*4) =64.0
0	1	F_S/(280*2)=58.5	F_S/(188*3)=58.0	F_S/(140*4) =58.5
1	0	F_S/(304*2)=53.9	F_S/(204*3)=53.5	F_S/(152*4) =53.9
1	1	F_S/(232*2)=70.6	F_S/(156*3)=70.0	F_S/(116*4) =70.6

F_S: sub-oscillator frequency

TYPE: LCD drive waveform type select bit.
= 0, A type waveform.
= 1, B type waveform.

LCDEN: LCD enable bit. When LCD function is disabled, all common/segment outputs are set to ground level.
= 0, LCD circuit disable.
= 1, LCD circuit enable.

DUTY1:DUTY0: LCD duty select bits.
= 0, 0 → 1/2 duty.
= 0, 1 → 1/3 duty.
= 1, 0 → 1/4 duty.

BIAS: LCD bias select bit.
= 0, 1/2 bias.
= 1, 1/3 bias.

2.1.8 LCDA (LCD Address Register) (Bank 0, 2)

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
0x0A	LCDA	0	0	0	LCDA4	LCDA3	LCDA2	LCDA1	LCDA0

Legend: 0 = Not used, must fixed to "0", more bits default state, please refer to Table 2.9.

LCDA4:LCDA0: LCD RAM address.

LCD Address	LCDD (LCD data buffer)								Segment
	Bit7	Bit6	Bit5	Bit4	Bit3 LCDD3	Bit2 LCDD2	Bit1 LCDD1	Bit0 LCDD0	
00h	-	-	-	-	C3S0	C2S0	C1S0	C0S0	SEG0
01h	-	-	-	-	C3S1	C2S1	C1S1	C0S1	SEG1
02h	-	-	-	-	C3S2	C2S2	C1S2	C0S2	SEG2
|	|								|
1Dh	-	-	-	-	C3S29	C2S29	C1S29	C0S29	SEG29
1Eh	-	-	-	-	C3S30	C2S30	C1S30	C0S30	SEG30
1Fh	-	-	-	-	C3S31	C2S31	C1S31	C0S31	SEG31
Common	-	-	-	-	COM3	COM2	COM1	COM0	

2.1.9 LCDD (LCD Data Buffer) (Bank 0, 2)

Read/Write-POR		x	x	x	x	R/W-x	R/W-x	R/W-x	R/W-x
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0B	LCDD	0	0	0	0	LCDD3	LCDD2	LCDD1	LCDD0

Legend: 0 = Not used, must fixed to “0”, x = unknown, more bits default state, please refer to Table 2.9.

LCDD4:LCDD0: LCD RAM data transfer buffer.

2.1.10 CNTCON (Counter Control Register) (Bank 0, 2)

Read/Write-POR		x	x	x	x	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0C	CNTCON	0	0	0	0	LPEN	HPEN	C2EN	C1EN

Legend: 0 = Not used, must fixed to “0”, x = unknown, more bits default state, please refer to Table 2.9.

C1EN: Counter 1 enable bit.
= 0, Disable.
= 1, Enable.

C2EN: Counter 2 enable bit.
= 0, Disable.
= 1, Enable.

HPEN: High pulse width timer enable bit.
= 0, Disable.
= 1, Enable.

LPEN: Low pulse width timer enable bit.
= 0, Disable.
= 1, Enable.

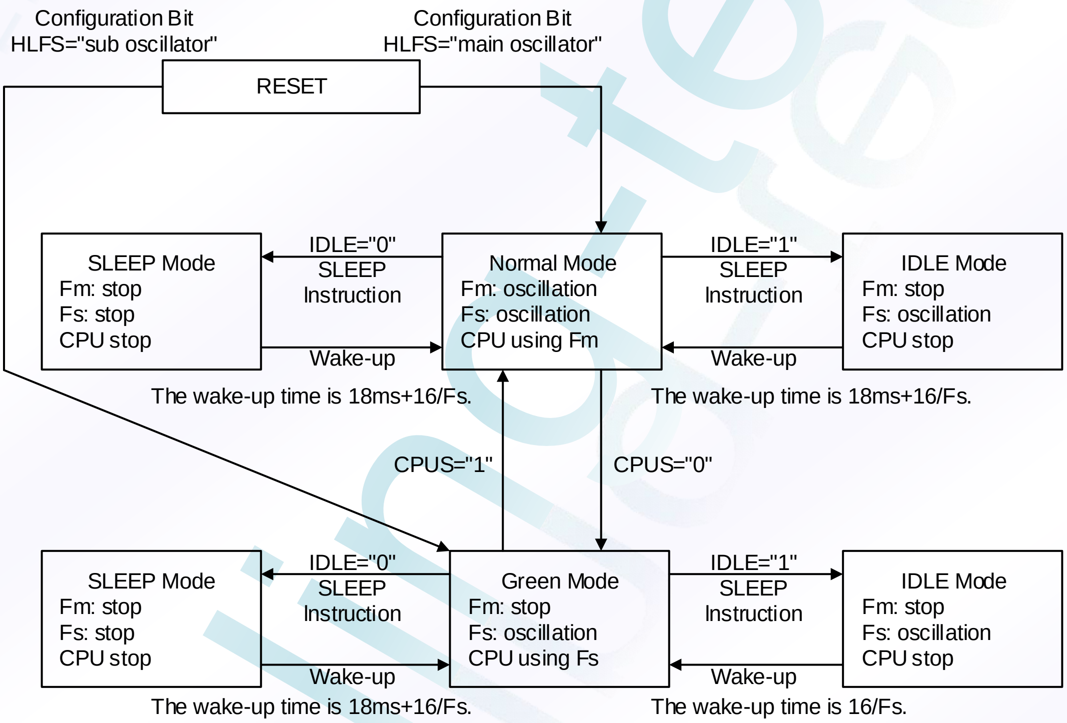
2.1.11 SYSCON (System Control Register) (Bank 0, 2)

Read/Write-POR		x	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-%
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Dh	SYSCON	0	PLLCK2	PLLCK1	PLLCK0	IDLE	LCDBF1	LCDBF0	CPUS

Legend: 0 = Not used, must fixed to “0”, x = unknown, % = refer to the configuration bit “HLFS”, more bits default state, please refer to Table 2.9.

CPUS: CPU oscillator source select bit.
= 0, Sub-oscillator (F_S) is selected, and the main oscillator is stopped.
= 1, Main oscillator (F_M) is selected.

Figure 2.3: CPU Operation Mode



LCDBF1:LCDBF0: LCD booster frequency select bits.
= 0, 0 → F_S
= 0, 1 → $F_S/4$
= 1, 0 → $F_S/8$
= 1, 1 → $F_S/16$

IDLE: Idle / sleep mode select bit of the SLEEP instruction.
= 0, Sleep mode after SLEEP instruction.
= 1, Idle mode after SLEEP instruction.

PLLCK2:PLLCK0: Main clock select bit for PLL mode (code option select)

PLLCK2:PLLCK0			Main clock (F$_M$) Frequency
0	0	0	32.768K*130 = 4.26M$_{HZ}$
0	0	1	32.768K*65 = 2.13M$_{HZ}$
0	1	0	32.768K*65/2 = 1.065M$_{HZ}$
0	1	1	32.768K*65/4 = 532.5K$_{HZ}$
1	x	x	32.768K*244 = 8M$_{HZ}$

2.1.12 IRCON (IR Control Register) (Bank 0, 2)

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	IRCON	IRE	HF	LGP	-	EIROUT	ET0CKI	EINT1	EINT0

Legend: - = unimplemented, read as ‘0’, more bits default state, please refer to Table 2.9.

EINT0: Define the function of IOA4/INT0 pin.
= 0, IOA4, bi-directional I/O pin.
= 1, INT0, external interrupt pin. The I/O control bit of IOA4 (bit 4 of IOSTA) must be set to “1”.

EINT1: Define the function of IOA5/INT1 pin.
= 0, IOA5, bi-directional I/O pin.
= 1, INT1, external interrupt pin. The I/O control bit of IOA5 (bit 5 of IOSTA) must be set to “1”.

ET0CKI: Define the function of IOA6/T0CKI pin.
= 0, IOA6, bi-directional I/O pin.
= 1, T0CKI, external input pin of Timer0. The I/O control bit of IOA6 (bit 6 of IOSTA) must be set to “1”.

EIROUT: Define the function of IOA7/IROUT pin.
= 0, IOA7, bi-directional I/O pin.
= 1, IROUT, The I/O control bit of IOA7 (bit 7 of IOSTA) must be set to “0”.

LGP: Long pulse.
=0, The high-pulse timer register and low-pulse width timer is valid.
=1, The high-pulse width timer register is ignored. So the IROUT waveform is dependent on low-pulse width timer register only

HF: High frequency.
=0, For PWM application, IROUT waveform is created according to high-pulse and low-pulse width time as determined by the high pulse and low pulse width timers respectively.
=1, For IR application mode, the low time sections of the generated pulse is modulated with the frequency F$_{CARRIER}$.

IRE: Infrared Remote Enable bit.
= 0, Disable IR H/W Modulator Function. IROUT pin fixed to high level.
= 1, Enable IR H/W Modulator Function.

2.1.13 PORTE & PORTF (Port Data Register) (Bank 1, 3)

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
0x09	PORTE	IOE7	IOE6	IOE5	IOE4	IOE3	IOE2	IOE1	IOE0

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
0x0A	PORTF	IOF7	IOF6	IOF5	IOF4	IOF3	IOF2	IOF1	IOF0

Note: more bits default state, please refer to Table 2.9.

Data latch of PORTE and PORTF. These registers are readable and writable.

2.1.14 RFCCON (RFC Control Register) (Bank 1, 3)

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
0x0B	RFCCON	RFCON	START	RFCIF	RFCMOD	-	-	RFCS1	RFCS0

Legend: - = unimplemented, read as ‘0’, more bits default state, please refer to Table 2.9.

RFCS1:RFCS0: Select one the RFC oscillation network of RFCx (x = 0 to 2). The selected RFCx pin will be configured as output pin if RFCON = 1. Other RFCx pins will behave as tristate input pins. If RFCON = 0, all RFCx pins will behave as tristate input pins.

RFCS1:RFCS0	RFC channel
0, 0	RFC0 pin is selected.
0, 1	RFC1 pin is selected.
1, 0	RFC2 pin is selected.
1, 1	No function, don’t use.

RFCMOD: RFC mode selection bit.
= 0, Enable/disable the counter by CX signal, and the clock source of the counter is the internal system clock (F_{OSC}).
= 1, Enable/disable the counter by START bit, and the clock source of the counter is the CX signal.

RFCIF: RFC module interrupt flag. Set when RFC conversion is completed if RFCMOD = 0, reset by software.

START: RFC counter enable bit
= 0, Stop the RFC conversion, reset by hardware when conversion is finished or by software.
= 1, RFC counter start to convert.

RFCON: RFC module enable bit.
= 0, Disable RFC module, all the RFCx and CX pins will behave as tristate input pins.
= 1, Enable RFC module.

2.1.15 RFCDL (RFC Data Register Low Byte) (Bank 1, 3)

Read/Write-POR		R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0C	RFCDL	RFCD7	RFCD6	RFCD5	RFCD4	RFCD3	RFCD2	RFCD1	RFCD0

Note: more bits default state, please refer to Table 2.9.

RFCD7:RFCD0: The low byte of RFC conversion result.

2.1.16 RFCDH (RFC Data Register High Byte) (Bank 1, 3)

Read/Write-POR		R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0D	RFCDH	RFCOV	RFCD14	RFCD13	RFCD12	RFCD11	RFCD10	RFCD9	RFCD8

Note: more bits default state, please refer to Table 2.9.

RFCD14:RFCD8: The high byte of RFC conversion result.

RFCOV: RFC counter overflow flag. Set when RFC counter overflow, reset by RFC counter reset.
= 0, Not overflow.
= 1, Overflow.

2.1.17 DIVCON (Divider Control Register) (Bank 1, 3)

Read/Write-POR		R/W-0	R/W-1	R/W-0	-	-	-	-	-
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0E	DIVCON	DIVON	DIVRST	DIVIF	-	-	-	-	-

Legend: - = unimplemented, read as ‘0’, more bits default state, please refer to Table 2.9.

DIVIF: Sub-oscillator (F_S) divider overflow (0.5 sec) interrupt flag. Set when F_S divider overflows, reset by software.

DIVRST: Sub-oscillator (F_S) divider RESET bit.
= 0, Reset the sub-oscillator divider. Set to “1” by hardware after the Sub-oscillator (F_S) divider is reset.
= 1, No action.

DIVON: Sub-oscillator (F_S) divider PAUSE bit.
= 0, Pause.
= 1, Continue.

2.1.18 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	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x0F	INTFLAG	PBDIF	LPIF	HPIF	C2IF	C1IF	INT1IF	INT0IF	T0IF

Note: more bits default state, please refer to Table 2.9.

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

INT0IF: External INT0 pin interrupt flag. Set by rising/falling (selected by INTEDG bit (T0CON<7>)) edge on INT0 pin, reset by software.

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

C1IF: Counter 1 underflow interrupt flag. Set when counter 1 underflows, reset by software.

C2IF: Counter 2 underflow interrupt flag. Set when counter 2 underflows, reset by software.

HPIF: High-pulse width timer underflow interrupt flag. Set when high-pulse width timer underflows, reset by software.

LPIF: Low-pulse width timer underflow interrupt flag. Set when low-pulse width timer underflows, reset by software.

PBDIF: Port B / Port D input change interrupt flag. Set when Port B / Port D input changes, reset by software.

2.1.19 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							

Note: more bits default state, please refer to Table 2.9.

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

2.1.20 IOSTA, IOSTB, IOSTC & IOSTD (Port I/O Control Registers)

Read/Write-POR		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	-	-	-	-

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

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
0x08	IOSTD	IOSTD7	IOSTD6	IOSTD5	IOSTD4	IOSTD3	IOSTD2	IOSTD1	IOSTD0

Accessed by IOST / IOSTR instruction.
Legend: - = unimplemented, read as ‘0’, more bits default state, please refer to Table 2.9.

The Port I/O Control Registers are loaded with the contents of the ACC Register by executing the IOST R (0x05~0x08) instruction. A ‘1’ from a IOST Register bit puts the corresponding output driver in hi-impedance state (input mode). A ‘0’ enables the output buffer and puts the contents of the output data latch on the selected pins (output mode).
The IOST Registers are set (output drivers disabled) upon RESET.

IOSTA7:IOSTA4: PORTA I/O direction control register.
= 0, set the relative I/O pins as output.
= 1, set the relative I/O pin into high impedance (input pin).

IOSTB7:IOSTB0: PORTB I/O direction control register.
= 0, set the relative I/O pins as output.
= 1, set the relative I/O pin into high impedance (input pin).

IOSTC7:IOSTC0: PORTC I/O direction control register.
= 0, set the relative I/O pins as output.
= 1, set the relative I/O pin into high impedance (input pin).

IOSTD7:IOSTD0: PORTD I/O direction control register.
= 0, set the relative I/O pins as output.
= 1, set the relative I/O pin into high impedance (input pin).

2.1.21 DRAMA (128 Bytes Data RAM Address 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
0x09	DRAMA	0	RAMA6	RAMA5	RAMA4	RAMA3	RAMA2	RAMA1	RAMA0

Accessed by IOST / IOSTR instruction.
Legend: 0 = Not used, must fixed to "0", - = unimplemented, read as '0', more bits default state, please refer to Table 2.9.

RAMA6:RAMA0: 128 bytes data RAM address.

2.1.22 DRAMD (128 Bytes Data RAM Data Buffer)

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
0x0A	DRAMD	RAMD7	RAMD6	RAMD5	RAMD4	RAMD3	RAMD2	RAMD1	RAMD0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

RAMD7:RAMD0: 128 bytes data RAM data transfer buffer.

2.1.23 C1PR (Counter 1 Preset 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
0x0B	C1PR	C1PR7	C1PR6	C1PR5	C1PR4	C1PR3	C1PR2	C1PR1	C1PR0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

C1PR7:C1PR0: All are Counter 1 buffer which is readable and writable. Counter 1 is an 8-bit down-counter with 8-bit pre-scaler. User can preset the counter and read preset value through C1PR register. After interrupt, it will auto reload the preset value. (The pre-scaler value is controlled by C12CON register)

2.1.24 C2PR (Counter 2 Preset 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
0x0C	C2PR	C2PR7	C2PR6	C2PR5	C2PR4	C2PR3	C2PR2	C2PR1	C2PR0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

C2PR7:C2PR0: All are Counter 2 buffer which is readable and writable. Counter 2 is an 8-bit down-counter with 8-bit pre-scaler. User can preset the counter and read preset value through C2PR register. After interrupt, it will auto reload the preset value.
When IR output is enabled, this control register can obtain carrier frequency output.
If the Counter 2 clock source is equal to FT (F_M or F_S, select by C12CON register):
Carrier frequency ($F_{CARRIER}$) = FT/[2*(preset value+1)*pre-scaler]
(The pre-scaler value is controlled by C12CON register)

2.1.25 HPPR (High-Pulse Width Timer Preset 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
0x0D	HPPR	HPPR7	HPPR6	HPPR5	HPPR4	HPPR3	HPPR2	HPPR1	HPPR0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

HPPR7:HPPR0: All are high-pulse width timer buffer which is readable and writable. High-pulse width timer is an 8-bit down-counter with 8-bit pre-scaler. User can preset the counter and read preset value through HPPR register. After interrupt, it will auto reload the preset value.
When PWM or IR output is enabled, this control register is set as high-pulse width.
If the high-pulse width timer clock source is equal to F_T (F_M or F_S, select by HLPCON register):
The high-pulse width = [(preset value+1)*pre-scaler] / F_T
(The pre-scaler value is controlled by HLPCON register)

2.1.26 LPPR (Low-Pulse Width Timer Preset 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
0Eh	LPPR	LPPR7	LPPR6	LPPR5	LPPR4	LPPR3	LPPR2	LPPR1	LPPR0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

LPPR7:LPPR0: All are low-pulse width timer buffer which is readable and writable. Low-pulse width timer is an 8-bit down-counter with 8-bit pre-scaler. User can preset the counter and read preset value through LPPR register. After interrupt, it will auto reload the preset value.
When PWM or IR output is enabled, this control register is set as low-pulse width.
If the low-pulse width timer clock source is equal to F_T (F_M or F_S, select by HLPCON register):
The low-pulse width = [(preset value+1)*pre-scaler]/ F_T
(The pre-scaler value is controlled by HLPCON register)

2.1.27 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
0Fh	INTEN	PBDIE	LPIE	HPIE	C2IE	C1IE	INT1IE	INT0IE	T0IE

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

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

C1IE: Counter 1 underflow interrupt enable bit.
= 0, Disable the counter 1 underflow interrupt.
= 1, Enable the counter 1 underflow interrupt.

C2IE: Counter 2 underflow interrupt enable bit.
= 0, Disable the counter 2 underflow interrupt.
= 1, Enable the counter 2 underflow interrupt.

HPIE: High-pulse width timer underflow interrupt enable bit.
= 0, Disable the high-pulse width timer underflow interrupt.
= 1, Enable the high-pulse width timer underflow interrupt.

LPIE: Low-pulse width timer underflow interrupt enable bit.
= 0, Disable the low-pulse width timer underflow interrupt.
= 1, Enable the low-pulse width timer underflow interrupt.

PBDIE: Port B / Port D input change interrupt enable bit.
= 0, Disable the Port B / Port D input change interrupt.
= 1, Enable Port B / Port D input change interrupt.

2.1.28 SEGCON (Segment Control Register)

Read/Write-POR		R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x15	SEGCON	IOFHS	IOFLS	IOEHS	IOELS	IODHS	IODLS	IOCHS	IOCLS

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

IOCLS: Select IOC0~IOC3 or SEG16~SEG19 output for SEGxx/IOCx pins.
= 0, Bi-directional I/O pins as normal IOC0~IOC3 are selected.
= 1, LCD segment SEG16~SEG19 output are selected.

IOCHS: Select IOC4~IOC7 or SEG20~SEG23 output for SEGxx/IOCx pins.
= 0, Bi-directional I/O pins as normal IOC4~IOC7 are selected.
= 1, LCD segment SEG20~SEG23 output are selected.

IODLS: Select IOD0~IOD3 or SEG24~SEG27output for SEGxx/IODx pins.
= 0, Bi-directional I/O pins as normal IOD0~IOD3 are selected.
= 1, LCD segment SEG24~SEG27 output are selected.

IODHS: Select IOD4~IOD7 (or RFC pins) or SEG28~SEG31 output for SEGxx/IODx/RFCx pins.
= 0, Bi-directional I/O pins as normal IOD4~IOD7 (or RFC pins) are selected.
= 1, LCD segment SEG28~SEG31 output are selected.

IOELS: Select IOE0~IOE3 or SEG0~SEG3 output for SEGxx/IOCx pins.
= 0, Output only pins as normal IOE0~IOE3 are selected.
= 1, LCD segment SEG0~SEG3 output are selected.

IOEHS: Select IOE4~IOE7 or SEG4~SEG7 output for SEGxx/IOCx pins.
= 0, Output only pins as normal IOE4~IOE7 are selected.
= 1, LCD segment SEG4~SEG7 output are selected.

IOFLS: Select IOF0~IOF3 or SEG8~SEG11output for SEGxx/IODx pins.
= 0, Output only I/O pins as normal IOF0~IOF3 are selected.
= 1, LCD segment SEG8~SEG11 output are selected.

IOFHS: Select IOF4~IOF7 or SEG12~SEG15 output for SEGxx/IODx pins.
= 0, Output only pins as normal IOF4~IOF7 are selected.
= 1, LCD segment SEG12~SEG15 output are selected.

2.1.29 WUCON (Wake-up Control Register)

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
0x16	WUCON	IRSC	*	*	*	/WUEDH	/WUEDL	/WUEBH	/WUEBL

Accessed by IOST / IOSTR instruction.
Legend: * = unimplemented, read as ‘1’, more bits default state, please refer to Table 2.9.

/WUEBL: Pin change wake up enable bit of IOB0~IOB3 pins.
= 0, enable IOB0~IOB3 pin change wake up function.
= 1, disable IOB0~IOB3 pin change wake up function.

/WUEBH: Pin change wake up enable bit of IOB4~IOB7 pins.
= 0, enable IOB4~IOB7 pin change wake up function.
= 1, disable IOB4~IOB7 pin change wake up function.

/WUEDL: Pin change wake up enable bit of IOD0~IOD3 pins.
= 0, enable IOD0~IOD3 pin change wake up function.
= 1, disable IOD0~IOD3 pin change wake up function.

/WUEDH: Pin change wake up enable bit of IOD4~IOD7 pins.
= 0, enable IOD4~IOD7 pin change wake up function.
= 1, disable IOD4~IOD7 pin change wake up function.

IRSC: IOA7/IROUT output Drive / Sink current select bit.

IRSC	IOA7/IROUT Sink current		IOA7/IROUT Drive current	
	V_{DD}=3V	V_{DD}=5V	V_{DD}=3V	V_{DD}=5V
0	7mA	10mA	1mA	3mA
1	14mA	20mA	2mA	6mA

2.1.30 T0CON (Timer0 Control Register)

Read/Write-POR		R/W-1	R-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
0x17	T0CON	INT0EDG	GIE	T0CS	T0SE	T0PS3	T0PS2	T0PS1	T0PS0

Accessed by OPTION / OPTIONR instruction. The T0CON Register are set all “1”s except GIE bit.
Note: more bits default state, please refer to Table 2.9.

T0PS3:T0PS0: Timer0 Pre-scaler rate select bits.

T0PS3:T0PS0				Timer0 Pre-Scaler Rate
0	x	x	x	1:1
1	0	0	0	1:2
1	0	0	1	1:4
1	0	1	0	1:8
1	0	1	1	1:16
1	1	0	0	1:32
1	1	0	1	1:64
1	1	1	0	1:128
1	1	1	1	1:256

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.

GIE: Global interrupt enable bit. Set by “ENI” or “RETFIE” instructions. Cleared by “DISI” instruction or entering into interrupt subroutine.
= 0, Disable all interrupts. For wake-up from SLEEP mode through an interrupt event, the device will continue execution at the instruction after the SLEEP instruction.
= 1, Enable all un-masked interrupts. For wake-up from SLEEP mode through an interrupt event, the device will branch to the interrupt address (0x003~0x021, based on different interrupt event).
Note: 1. The GIE bit is not writable bit. This bit is only set by “ENI” or “RETFIE” instructions, and cleared by “DISI” instruction or entering into interrupt subroutine.
2. When an interrupt event occur with the GIE bit and its corresponding interrupt enable bit are all set, the GIE bit will be cleared by hardware to disable any further interrupts. The RETFIE instruction will exit the interrupt routine and set the GIE bit to re-enable interrupt.

INT0EDG: INT0 pin interrupt edge select bit.
= 1, interrupt on falling edge of INT0 pin.
= 0, interrupt on rising edge of INT0 pin.

2.1.31 WDTCON (Watch-Dog Timer Control Register)

Read/Write-POR		*	*	*	*	R/W-0	R/W-1	R/W-1	R/W-1
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x18	WDTCON	*	*	*	*	WDTEN	WDTPS2	WDTPS1	WDTPS0

Accessed by IOST / IOSTR instruction.
Legend: * = unimplemented, read as ‘1’, more bits default state, please refer to Table 2.9.

WDTPS2:WDTPS0: Watch-Dog timer Pre-scaler rate select bits.

WDTPS2:WDTPS0			WDT Pre-Scaler Rate
0	0	0	1:1
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

WDTEN: Watchdog timer enable bit.
= 1, Enable WDT function.
= 0, Disable WDT function.

2.1.32 C12CON (Counter 1 & 2 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
0x19	C12CON	C2CS	C2PS2	C2PS1	C2PS0	C1CS	C1PS2	C1PS1	C1PS0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

C1S2:C1PS0: Counter1 pre-scaler select bits.

C1PS2:C1PS0			Counter 1 Pre-scaler Rate
0	0	0	1:2
0	0	1	1:4
0	1	0	1:8
0	1	1	1:16
1	0	0	1:32
1	0	1	1:64
1	1	0	1:128
1	1	1	1:256

C1CS: Counter 1 clock source select bit.
= 0, Sub-oscillator clock (F_S) is selected.
= 1, Main-oscillator clock (F_M) is selected.

C2S2:C2PS0: Counter 1 pre-scaler select bits.

C2PS2:C2PS0			Counter 2 Pre-scaler Rate
0	0	0	1:2
0	0	1	1:4
0	1	0	1:8
0	1	1	1:16
1	0	0	1:32
1	0	1	1:64
1	1	0	1:128
1	1	1	1:256

C2CS: Counter 2 clock source select bit.
= 0, Sub-oscillator clock (F_S) is selected.
= 1, Main-oscillator clock (F_M) is selected.

2.1.33 HLPCON (High-pulse / Low-pulse width timer 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
0x1A	HLPCON	LPCS	LPPS2	LPPS1	LPPS0	HPCS	HPPS2	HPPS1	HPPS0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

HPPS2:HPPS0: High-pulse width timer pre-scaler select bits.

HPPS2:HPPS0			High-pulse width timer Pre-Scaler Rate
0	0	0	1:2
0	0	1	1:4
0	1	0	1:8
0	1	1	1:16
1	0	0	1:32
1	0	1	1:64
1	1	0	1:128
1	1	1	1:256

HPCS: High-pulse width timer clock source select bit.
= 0, Sub-oscillator clock (F_S) is selected.
= 1, Main-oscillator clock (F_M) is selected.

LPPS2:LPPS0: Low-pulse width timer pre-scaler select bits.

LPPS2:LPPS0			Low-pulse width timer Pre-Scaler Rate
0	0	0	1:2
0	0	1	1:4
0	1	0	1:8
0	1	1	1:16
1	0	0	1:32
1	0	1	1:64
1	1	0	1:128
1	1	1	1:256

LPCS: Low-pulse width timer clock source select bit.
= 0, Sub-oscillator clock (F_S) is selected.
= 1, Main-oscillator clock (F_M) is selected.

2.1.34 BPHCON (PORTB Pull-high Control Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x1B	BPHCON	PHB7	PHB6	PHB5	PHB4	PHB3	PHB2	PHB1	PHB0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

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

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

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

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

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

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

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

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

2.1.35 BODCON (PORTB Open-drain 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
0x1C	BODCON	ODB7	ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

ODB0: = 0, Disable the internal open-drain of IOB0 pin.
= 1, Enable the internal open-drain of IOB0 pin.

ODB1: = 0, Disable the internal open-drain of IOB1 pin.
= 1, Enable the internal open-drain of IOB1 pin.

ODB2: = 0, Disable the internal open-drain of IOB2 pin.
= 1, Enable the internal open-drain of IOB2 pin.

ODB3: = 0, Disable the internal open-drain of IOB3 pin.
= 1, Enable the internal open-drain of IOB3 pin.

ODB4: = 0, Disable the internal open-drain of IOB4 pin.
= 1, Enable the internal open-drain of IOB4 pin.

ODB5: = 0, Disable the internal open-drain of IOB5 pin.
= 1, Enable the internal open-drain of IOB5 pin.

ODB6: = 0, Disable the internal open-drain of IOB6 pin.
= 1, Enable the internal open-drain of IOB6 pin.

ODB7: = 0, Disable the internal open-drain of IOB7 pin.
= 1, Enable the internal open-drain of IOB7 pin.

2.1.36 DPHCON (PORTD Pull-high Control Register)

Read/Write-POR		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x1D	DPHCON	PHD7	PHD6	PHD5	PHD4	PHD3	PHD2	PHD1	PHD0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

PHD0: = 0, Disable the internal pull-high of IOD0 pin.
= 1, Enable the internal pull-high of IOD0 pin.

PHD1: = 0, Disable the internal pull-high of IOD1 pin.
= 1, Enable the internal pull-high of IOD1 pin.

PHD2: = 0, Disable the internal pull-high of IOD2 pin.
= 1, Enable the internal pull-high of IOD2 pin.

PHD3: = 0, Disable the internal pull-high of IOD3 pin.
= 1, Enable the internal pull-high of IOD3 pin.

PHD4: = 0, Disable the internal pull-high of IOD4 pin.
= 1, Enable the internal pull-high of IOD4 pin.

PHD5: = 0, Disable the internal pull-high of IOD5 pin.
= 1, Enable the internal pull-high of IOD5 pin.

PHD6: = 0, Disable the internal pull-high of IOD6 pin.
= 1, Enable the internal pull-high of IOD6 pin.

PHD7: = 0, Disable the internal pull-high of IOD7 pin.
= 1, Enable the internal pull-high of IOD7 pin.

2.1.37 BPDCON (PORTB Pull-down 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
0x1E	BPDCON	PDB7	PDB6	PDB5	PDB4	PDB3	PDB2	PDB1	PDB0

Accessed by IOST / IOSTR instruction.
Note: more bits default state, please refer to Table 2.9.

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

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

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

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

PDB4: = 0, Disable the internal pull-down of IOB4 pin.
= 1, Enable the internal pull-down of IOB4 pin.

PDB5: = 0, Disable the internal pull-down of IOB5 pin.
= 1, Enable the internal pull-down of IOB5 pin.

PDB6: = 0, Disable the internal pull-down of IOB6 pin.
= 1, Enable the internal pull-down of IOB6 pin.

PDB7: = 0, Disable the internal pull-down of IOB7 pin.
= 1, Enable the internal pull-down of IOB7 pin.

2.1.38 INTEN1 (Interrupt Mask Register 1)

Read/Write-POR		-	-	-	-	-	R/W-0	R/W-0	-
Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x1F	INTEN1	-	-	-	-	-	DIVIE	RFCIE	-

Accessed by IOST / IOSTR instruction.
Legend: - = unimplemented, read as ‘0’, more bits default state, please refer to Table 2.9.

RFCIE: RFC module interrupt enable bit.
= 0, Disable the RFC module interrupt.
= 1, Enable the RFC module interrupt.

DIVIE: Sub-oscillator (F_S) divider overflow (0.5 sec) interrupt enable bit.
= 0, Disable the sub-oscillator divider interrupt. And the divider will be cleared.
= 1, Enable the sub-oscillator divider interrupt.

2.2 I/O Ports

Port A, port B, port C and port D are bi-directional tri-state I/O ports, and Port E and port F are output only ports. Port A is a 4-pin I/O port. Port B, port C and port D are 8-pin I/O ports. Port E and port F are 8-pin output only ports. All I/O pins (IOA<7:4>, IOB<7:0>, IOC<7:0> and IOD<7:0>) have data direction control registers (IOSTA, IOSTB, IOSTC and IOSTD) which can configure these pins as output or input.

IOB<7:0> and IOD<7:0> have its corresponding pull-high control bits (BPHCON and DPHCON 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.

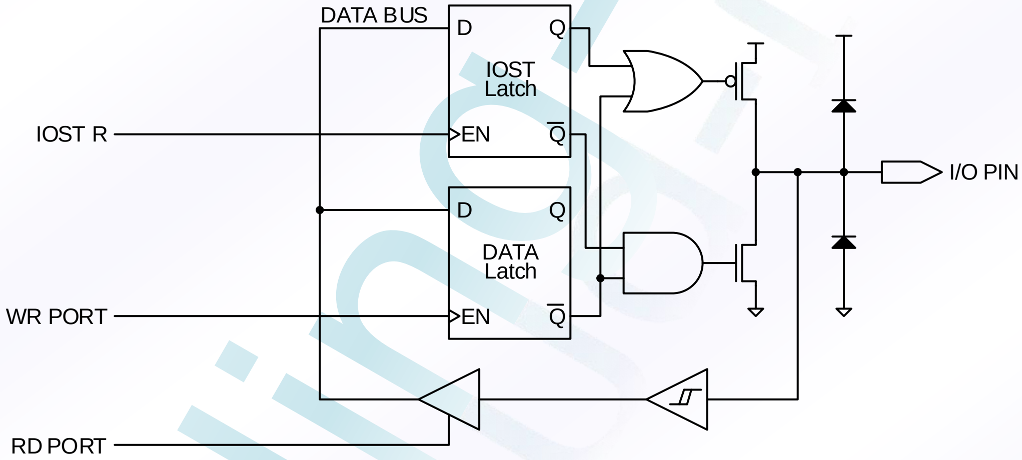
IOB<7:0> have its corresponding pull-down control bits (BPDCON register) to enable the weak internal pull-down. The weak pull-down is automatically turned off when the pin is configured as an output pin.

IOB<7:0> have its corresponding open-drain control bits (BODCON register) to enable the open-drain output when these pins are configured to be an output pin.

IOB<7:0> and IOD<7:0> also provide the input status change interrupt/wake-up function. Each pin has its corresponding input change interrupt/wake-up enable bits (WUCON register) to select the input change interrupt/wake-up source.

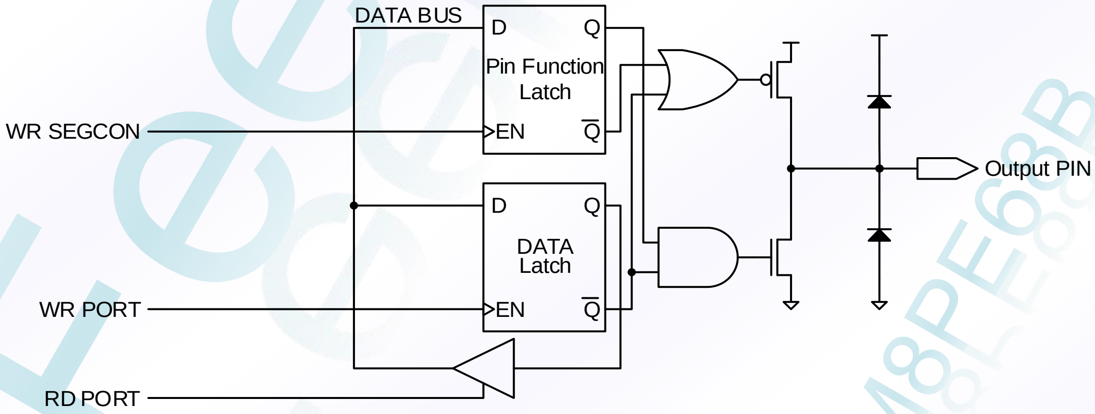
Figure 2.4: Block Diagram of I/O Pins

IOA7 ~ IOA4, IOB7 ~ IOB0, IOC7 ~ IOC0, IOD7 ~ IOD0:



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

IOE7 ~ IOE0, IOF7 ~ IOF0:



2.3 Timer0/WDT & Pre-scaler

2.3.1 Timer0

The Timer0 is a 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 (T0CON<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 (T0CON<5>). In this mode, Timer0 will increment either on every rising or falling edge, setting by T0SE bit (T0CON<4>), of pin T0CKI. The incrementing edge is determined by the T0SE bit (T0CON<4>).

2.3.1.3 Timer0 Pre-scaler

An 8-bit counter (down counter) is available as a pre-scaler for the Timer0. And the T0PS<2:0> bits (T0CON<2:0>) determine pre-scaler ratio.

The pre-scaler is neither readable nor writable. All instructions writing to the TMR0 register will clear the pre-scaler. On a RESET, the pre-scaler contains all ‘1’s.

2.3.2 Watchdog Timer (WDT)

The Watchdog Timer (WDT) is a free running on sub-oscillator on operation mode, and running on sub-oscillator or on-chip RC oscillator selected by WDTS configuration bit on SLEEP mode. So the WDT can still run even if the oscillator driver has been turned off if WDTS = Internal RC. During Normal mode, Green mode, or Idle mode operation, a WDT time-out will cause the device to reset and the $\overline{\text{TO}}$ bit (STATUS<4>) will be cleared.

The WDT can be disabled by clearing the control bit WDTEN (WDTCON<3>) to “0”.

The WDT time-out period is equal to (pre-scaler * (512/F_S))

The “CLRWDT” instruction clears the WDT and the pre-scaler, and prevents it from timing out and generating a device reset.

The “SLEEP” instruction resets the WDT and the pre-scaler. This gives the maximum SLEEP time before a WDT Wake-up Reset.

2.3.2.1 Watchdog Pre-scaler

An 8-bit counter (down counter) is available as a pre-scaler for the Watchdog Timer (WDT). And the WDTPS<2:0> bits (WDTCON<2:0>) determine pre-scaler ratio.

The pre-scaler is neither readable nor writable. A “CLRWDT” or “SLEEP” instruction will clear the pre-scaler. On a RESET, the pre-scaler contains all ‘1’s.

Figure 2.5: Block Diagram of the Timer0/WDT Pre-scaler

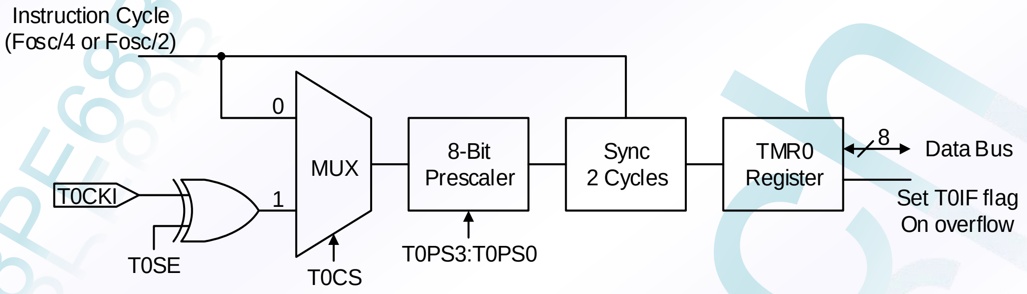
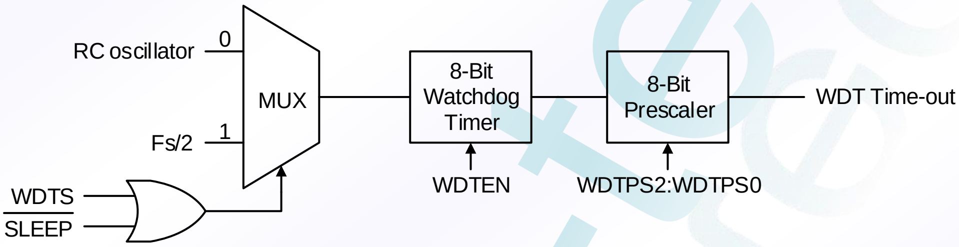


Figure 2.6: Block Diagram of the Timer0/WDT Pre-scaler



2.4 Interrupts

The FM8PE68B has up to ten sources of interrupt:

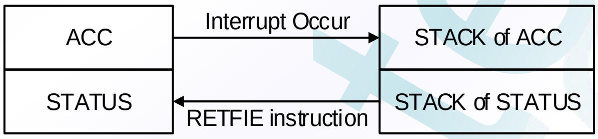
1. TMR0 overflow.
2. External interrupt INT0 pin.
3. External interrupt INT1 pin.
4. Counter 1 underflow.
5. Counter 2 underflow.
6. High-pulse width timer underflow.
7. Low-pulse width timer underflow.
8. Port B, Port D input status change.
9. RFC module interrupt.
10. Sub-oscillator (Fs) divider overflow (0.5 sec).

INTFLAG is the interrupt flag register that recodes the interrupt requests in the relative flags.
A global interrupt enable bit, GIE (T0CON<6>), 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 and INTEN1 registers regardless of the status of the GIE bit.
When an interrupt event occur with the GIE bit and its corresponding interrupt enable bit are all set, the GIE bit will be cleared by hardware to disable any further interrupts, and the next instruction will be fetched from address 003h~0021h based on the interrupt source. The interrupt flag bits must be cleared by software before re-enabling GIE bit to avoid recursive interrupts.
The existing interrupt service routine does not allow other interrupt service routine to be executed. So if other interrupts occur while the existing interrupt service routine is being executed, the hardware will save the later interrupts. Only after the existing interrupt service routine is completed that the next interrupt service routine is executed.
Executing the “ENI” instruction will set the GIE bit, and executing the “DISI” instruction will clear the GIE bit.
The RETFIE instruction exits the interrupt routine and set the GIE bit to re-enable interrupt.
When an interrupt is generated by the INT instruction, the next instruction will be fetched from address 0x002.
Each individual interrupt source has its own interrupt vector as the table list below. Before the interrupt subroutine is executed, the contents of ACC and the STATUS register are initially saved by hardware. After the interrupt service routine is completed, ACC and STATUS register are restored.

Table 2.1: Interrupt Vector

Interrupt Vector	Interrupt Source
0x003	TMR0 overflow
0x006	External interrupt INT0 pin
0x009	External interrupt INT1 pin
0x00C	Counter 2 underflow
0x00F	Counter 1 underflow
0x012	High-pulse width timer underflow
0x015	Low-pulse width timer underflow
0x018	Port B, Port D input status change wake-up
0x01E	RFC module interrupt
0x021	F_S divider overflow (0.5 sec)

Figure 2.7: Interrupt Backup Diagram



2.4.1 Timer0 Overflow 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>).

2.4.2 External INT0 Pin Interrupt

External interrupt on INT0 pin is rising or falling edge triggered selected by INT0EDG (T0CON<7>).
When a valid edge appears on the INT0 pin the flag bit INT0IF (INTFLAG<1>) is set. This interrupt can be disabled by clearing INT0IE bit (INTEN<1>).

2.4.3 External INT1 Pin Interrupt

External interrupt on INT1 pin is falling edge triggered.
When a falling edge appears on the INT1 pin the flag bit INT1IF (INTFLAG<2>) is set. This interrupt can be disabled by clearing INT1IE bit (INTEN<2>).

2.4.4 Counter 1 Underflow Interrupt

An underflow (0x00 → 0xFF) in the counter 1 timer will set the flag bit C1IF (INTFLAG<3>). This interrupt can be disabled by clearing C1IE bit (INTEN<3>).

2.4.5 Counter 2 Underflow Interrupt

An underflow (0x00 → 0xFF) in the counter 2 timer will set the flag bit C2IF (INTFLAG<4>). This interrupt can be disabled by clearing C2IE bit (INTEN<4>).

2.4.6 High-pulse Width Timer Underflow Interrupt

An underflow (0x00 → 0xFF) in the high-pulse width timer will set the flag bit HPIF (INTFLAG<5>). This interrupt can be disabled by clearing HPIE bit (INTEN<5>).

2.4.7 Low-pulse Width Timer Underflow Interrupt

An underflow (0x00 → 0xFF) in the low-pulse width timer will set the flag bit LPIF (INTFLAG<6>). This interrupt can be disabled by clearing LPIE bit (INTEN<6>).

2.4.8 Port B / Port D Input Status Change Interrupt

An input status change on IOB<7:0> or IOD<7:0> will set flag bit PBDIF (INTFLAG<7>). This interrupt can be disabled by clearing PBDIE bit (INTEN<7>).
Before the port B / port D input change interrupt is enabled, reading PORTB and/or PORTD (any instruction accessed to PORTB and/or PORTD, including read/write instructions) is necessary. Any pin which corresponding WUEn bit (WUCON<3:0>) is cleared to "0" or configured as output will be excluded from this function.

2.4.9 RFC Module Interrupt

After RFC conversion is finished, the RFCIF flag (RFCCON<5>) will be set. This interrupt can be disabled by clearing RFCIE bit (INTEN1<1>).

2.4.10 Sub-oscillator (Fs) divider overflow (0.5 sec) interrupt

An overflow (0x3FFF → 0x0000) in the Sub-oscillator (F_S) divider will set the flag bit DIVIF (DIVCON<5>). This interrupt can be disabled by clearing DIVIE bit (INTEN1<2>).

2.5 Power-down (SLEEP) and IDLE Mode

Power-down (SLEEP) or IDLE mode is entered by executing a SLEEP instruction. The "IDLE" bit (SYSCON<3>) decides the intended mode of the SLEEP instruction.

Table 2.2: SLEEP or IDLE Mode after SLEEP instruction

IDLE	SLEEP / IDLE Mode Selection
0	SLEEP Mode
1	IDLE Mode

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.

2.5.1 Wake-up from SLEEP Mode

The device can wake-up from SLEEP mode and IDLE mode through one of the following events:
1. RSTB reset.
2. WDT time-out reset (if enabled).
3. External interrupt INT0 pin.
4. External interrupt INT1 pin.
5. Port B, Port D input status change.

The device can wake-up from IDLE mode through one of the additional events:
6. Counter 1 underflow.
7. Counter 2 underflow.
8. High-pulse width timer underflow.
9. Low-pulse width timer underflow.
10. Sub-oscillator (F_S) divider overflow.

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/PORTD input status change, and the program will execute interrupt service routine or next PC after wake-up. Any pin which corresponding /WUEmn bit (WUCON<3:0>) is set to "1" or configured as output will be excluded from this function.

And GIE bit also decides whether or not the processor branches to the interrupt vector following wake-up. If GIE bit was set, the program will execute interrupt service routine after wake-up; or if GIE bit was cleared, the program will execute next PC after wake-up.

The system wake-up delay time is 18ms plus 16 sub-oscillator cycle time.

Table 2.3: Wake-up from SLEEP or IDLE Mode

Wake-up signal	Sleep mode	Idle mode	Green mode	Normal mode
Timer0 overflow (T0IE = 1)	X	X	Interrupt (0x003)	Interrupt (0x003)
INT0 pin (INT0IE = 1)	Wake-up + interrupt (0x006) + next instruction	Wake-up + interrupt (0x006) + next instruction	Interrupt (0x006)	Interrupt (0x006)
INT1 pin (INT1IE = 1)	Wake-up + interrupt (0x009) + next instruction	Wake-up + interrupt (0x009) + next instruction	Interrupt (0x009)	Interrupt (0x009)
Counter 1 underflow (C1IE = 1)	X	Wake-up + interrupt (0x00C) + next instruction	Interrupt (0x00C)	Interrupt (0x00C)
Counter 2 underflow (C2IE = 1)	X	Wake-up + interrupt (0x00F) + next instruction	Interrupt (0x00F)	Interrupt (0x00F)
High-pulse timer underflow (HPIE = 1)	X	Wake-up + interrupt (0x012) + next instruction	Interrupt (0x012)	Interrupt (0x012)
Low-pulse timer underflow (LPIE = 1)	X	Wake-up + interrupt (0x015) + next instruction	Interrupt (0x015)	Interrupt (0x015)
Port B, Port D input status change (PBDIE = 0)	Wake-up + next instruction	Wake-up + next instruction	X	X
Port B, Port D input status change (PBDIE = 1)	Wake-up + interrupt (0x018) + next instruction	Wake-up + interrupt (0x018) + next instruction	X	X
RFC conversion finished (RFCIE = 1)	X	X	Interrupt (0x01E)	Interrupt (0x01E)
Sub-oscillator (Fs) divider overflow (DIVIE = 1)	X	Wake-up + interrupt (0x021) + next instruction	Interrupt (0x021)	Interrupt (0x021)
WDT time out (WDTS = F_S)	X	Reset	Reset	Reset
WDT time out (WDTS = Internal RC)	Reset	Reset	Reset	Reset
RSTB pin	Reset	Reset	Reset	Reset

2.6 Infrared Remote Output (IROUT) / PWM Generator

The FM8PE68B devices can output infrared carrier in a friendly manner or in PWM standard waveform. The IR and PWM waveform generated functions include an 8-bit down count timer, high-pulse width timer, low-pulse width timer, and IR control register. The IROUT waveform is determined by IR control register (IRCON), Counter 1, 2 control register (C12CON), high-pulse width timer, low-pulse width timer control register (HLPCON), Counter 2 preset register (C2PR), high-pulse width timer preset register (HPPR), and low-pulse width timer preset register (LPPR). Details on IR carrier, high-pulse time, and low pulse time are explained as follows:

If Counter 2 source clock is F_T (F_M or F_S, select by C12CON register);

$$\text{IR carrier} = \frac{F_T}{2 * [\text{Counter 2 preset value (C2PR)}+1] * \text{pre-scaler}}$$

(The pre-scaler value is controlled by C12CON register)

If high-pulse width timer source clock is F_T (F_M or F_S, select by HLPCON register);

$$\text{High-pulse time} = \frac{\text{pre-scaler} * [\text{ high-pulse width timer value (HPPR)}+1]}{F_T}$$

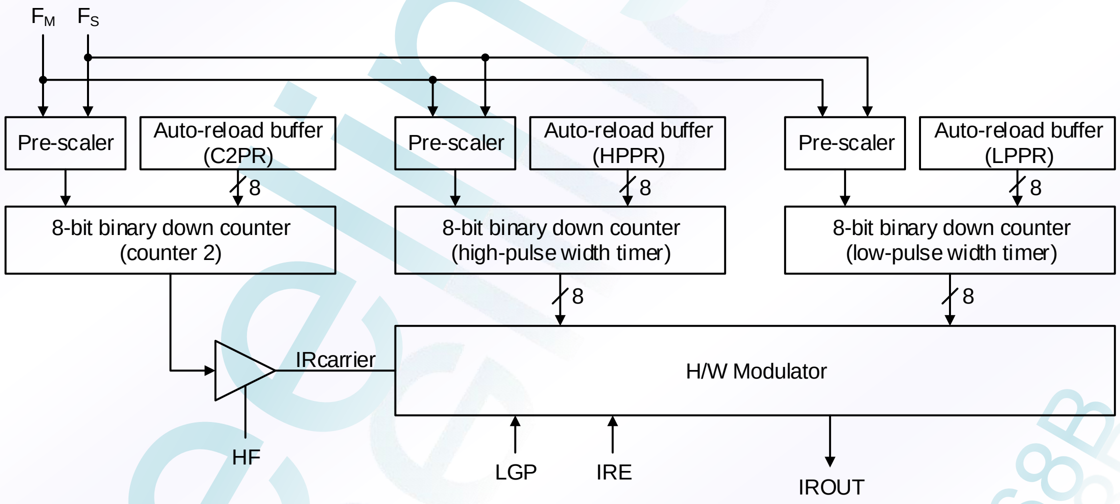
(The pre-scaler value is controlled by HLPCON register)

If low-pulse width timer source clock is F_T (F_M or F_S, select by HLPCON register);

$$\text{Low-pulse time} = \frac{\text{pre-scaler} * [\text{ low-pulse width timer value (LPPR)}+1]}{F_T}$$

(The pre-scaler value is controlled by HLPCON register)

Figure 2.8: IROUT / PWM System Block Diagram


FM
FS
Pre-scaler
Auto-reload buffer (C2PR)
8
8-bit binary down counter (counter 2)
Pre-scaler
Auto-reload buffer (HPPR)
8
8-bit binary down counter (high-pulse width timer)
Pre-scaler
Auto-reload buffer (LPPR)
8
8-bit binary down counter (low-pulse width timer)
8
8
IRcarrier
H/W Modulator
HF
LGP
IRE
IROUT

F_M: Main-oscillator frequency; F_S: Sub-oscillator frequency

Figure 2.9: IROUT Pin Output Waveform (HF=1, LGP=0)

The IROUT waveform can modulate IR carrier waveform when in low-pulse width time.

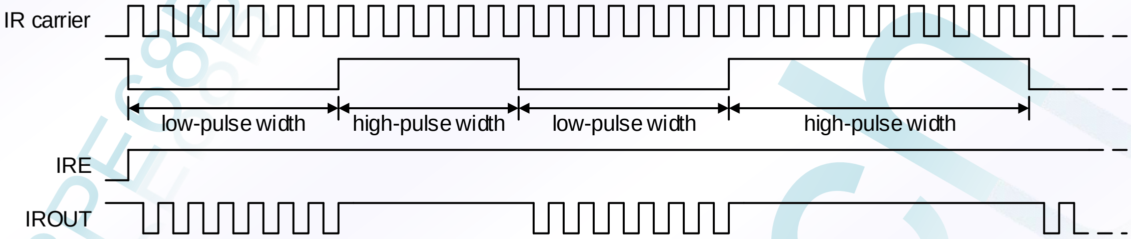


Figure 2.10: IROUT Pin Output Waveform (HF=1, LGP=0)

The IROUT waveform can modulate IR carrier waveform when in low-pulse width time. When IRE goes from high to low, the output waveform of IROUT will keep on transmitting till high-pulse width timer interrupt occurs.

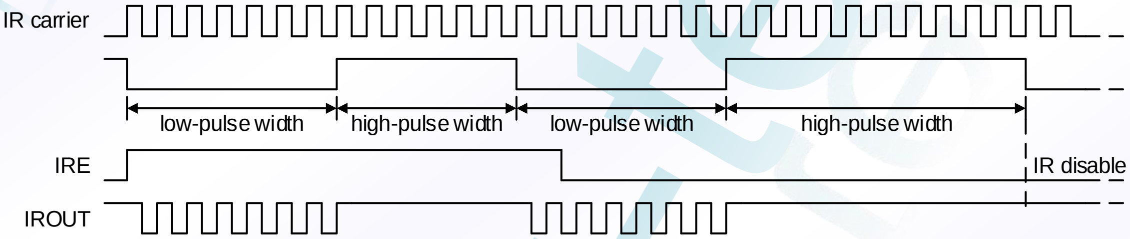


Figure 2.11: IROUT Pin Output Waveform (HF=0, LGP=0)

The IROUT waveform cannot modulate IR carrier waveform when in low-pulse width time. So IROUT waveform is determined by high-pulse time and low-pulse time. This mode can produce standard PWM waveform

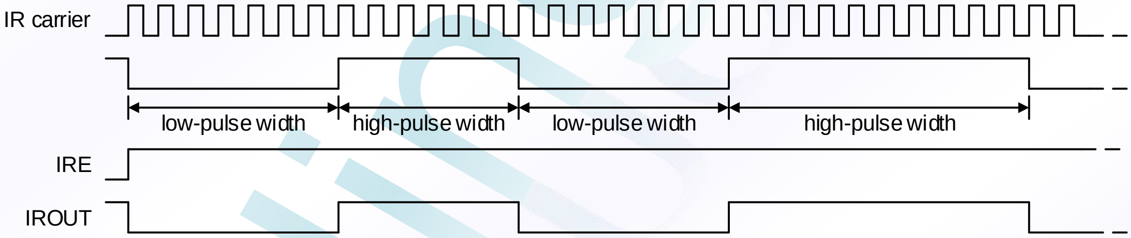


Figure 2.12: IROUT Pin Output Waveform (HF=0, LGP=0)

The IROUT waveform cannot modulate IR carrier waveform when in low-pulse width time. So IROUT waveform is determined by high-pulse time and low-pulse time. This mode can produce standard PWM waveform. When IRE goes from high to low, the output waveform of IROUT will keep on transmitting till high-pulse width timer interrupt occurs.

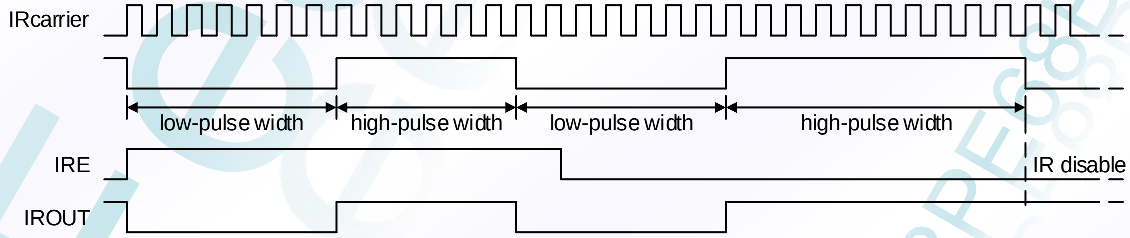


Figure 2.13: IROUT Pin Output Waveform (HF=1, LGP=1)

When LGP bit is set to high level, the high-pulse width timer is ignored. So IROUT waveform output from low-pulse width timer is established.

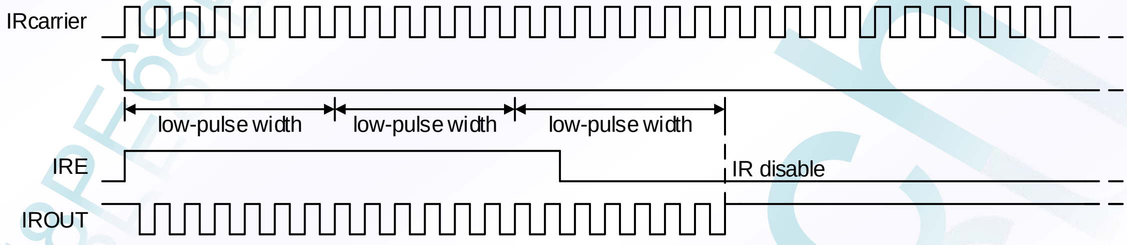
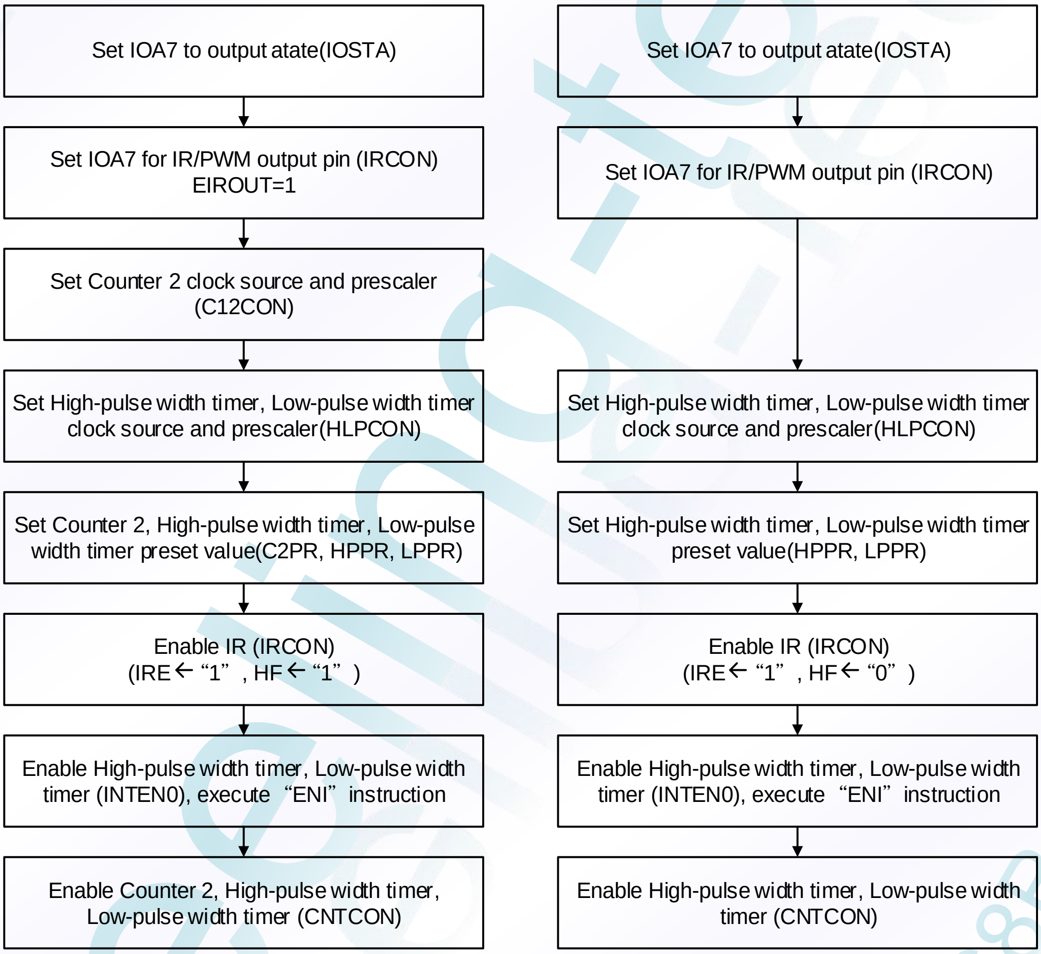


Figure 2.14: IR / PWM function enable flowchart


Set IOA7 to output atate(IOSTA)
Set IOA7 for IR/PWM output pin (IRCON) EIROUT=1
Set Counter 2 clock source and prescaler (C12CON)
Set High-pulse width timer, Low-pulse width timer clock source and prescaler(HLPCON)
Set Counter 2, High-pulse width timer, Low-pulse width timer preset value(C2PR, HPPR, LPPR)
Enable IR (IRCON) (IRE← “1” , HF← “1”)
Enable High-pulse width timer, Low-pulse width timer (INTEN0), execute “ENI” instruction
Enable Counter 2, High-pulse width timer, Low-pulse width timer (CNTCON)
Set IOA7 to output atate(IOSTA)
Set IOA7 for IR/PWM output pin (IRCON)
Set High-pulse width timer, Low-pulse width timer clock source and prescaler(HLPCON)
Set High-pulse width timer, Low-pulse width timer preset value(HPPR, LPPR)
Enable IR (IRCON) (IRE← “1” , HF← “0”)
Enable High-pulse width timer, Low-pulse width timer (INTEN0), execute “ENI” instruction
Enable High-pulse width timer, Low-pulse width timer (CNTCON)

(a) IR application

(b) PWM application

2.7 LCD

The FM8PE68B devices can drive LCD of up to 32 segments and 4 commons that can drive a total of 4*32 dots. LCD block is made up of LCD driver, display RAM, segment output pins, common output pins, and LCD operating power supply pins. This circuit works on normal mode, green mode, and idle mode. The LCD duty, bias, the number of segment, the number of common, and frame frequency are determined by the LCD controller register (LCDCON). The basic structure contains a timing control that uses a subsystem clock to generate the proper timing for different duty and display accesses. The LCDCON register is a command register for LCD driver which includes LCD enable/disable, bias (1/2 and 1/3), duty (1/2, 1/3, 1/4), and LCD frame frequency control. The register LCDA is an LCD RAM address control register. The register LCDD is an LCD RAM data buffer. LCD booster circuit can change operation frequency to improve VLCD1 and VLCD2 drive capability.

Figure 2.15: LCD Bias Selection

BIAS	LCD bias
0	1/2 bias
1	1/3 bias

Figure 2.16: LCD Duty Selection

DUTY1	DUTY0	LCD duty
0	0	1/2 duty
0	1	1/3 duty
1	X	1/4 duty

Table 2.4: LCD Enable / Disable

LCDEN	LCD ON / OFF
0	Disable
1	Enable

Table 2.5: LCD Display Type Selection

TYPE	LCD display type
0	A type
1	B type

Table 2.6: LCD Frame Frequency Selection

LCDF1	LCDF0	LCD frame frequency (Fs=32.768KHZ)		
		1/2 duty	1/3 duty	1/4 duty
0	0	Fs/(256*2)=64.0	Fs/(172*3)=63.5	Fs/(128*4) =64.0
0	1	Fs/(280*2)=58.5	Fs/(188*3)=58.0	Fs/(140*4) =58.5
1	0	Fs/(304*2)=53.9	Fs/(204*3)=53.5	Fs/(152*4) =53.9
1	1	Fs/(232*2)=70.6	Fs/(156*3)=70.0	Fs/(116*4) =70.6

Fs: sub-oscillator frequency

Table 2.7: LCD Duty Selection

LCDBF1	LCDBF0	Booster frequency
0	0	Fs
0	1	Fs/4
1	0	Fs/8
1	1	Fs/16

Fs: sub-oscillator frequency

Figure 2.17: The Connection of Charge Pump Circuit

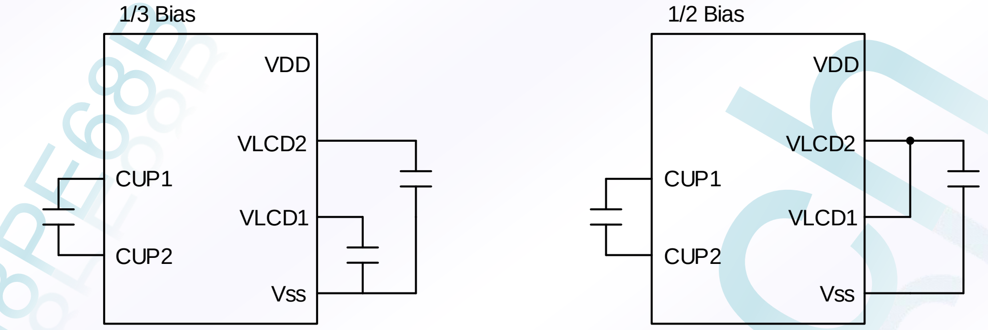
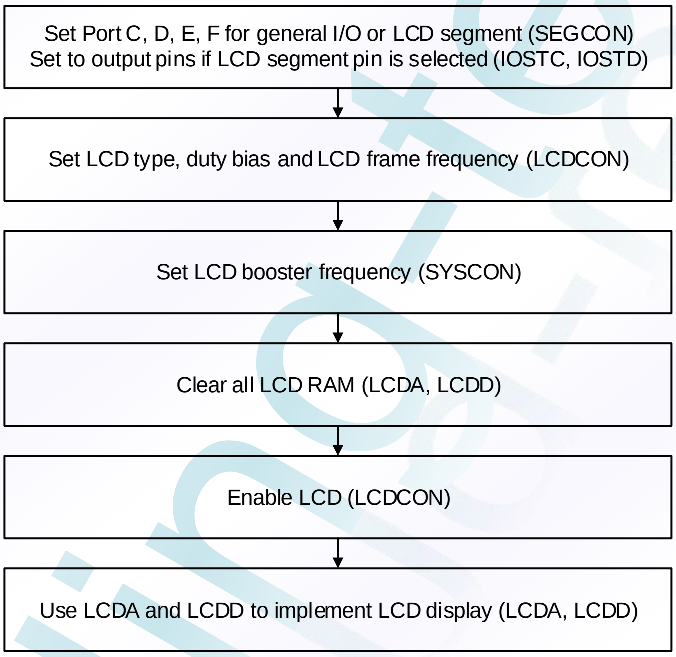


Figure 2.18: The Initial Setting Flowchart for LCD Function



Set Port C, D, E, F for general I/O or LCD segment (SEGCON)
Set to output pins if LCD segment pin is selected (IOSTC, IOSTD)

Set LCD type, duty bias and LCD frame frequency (LCDCON)

Set LCD booster frequency (SYSCON)

Clear all LCD RAM (LCDA, LCDD)

Enable LCD (LCDCON)

Use LCDA and LCDD to implement LCD display (LCDA, LCDD)

Figure 2.19: LCD Waveform for 1/2 Bias, 1/2 Duty

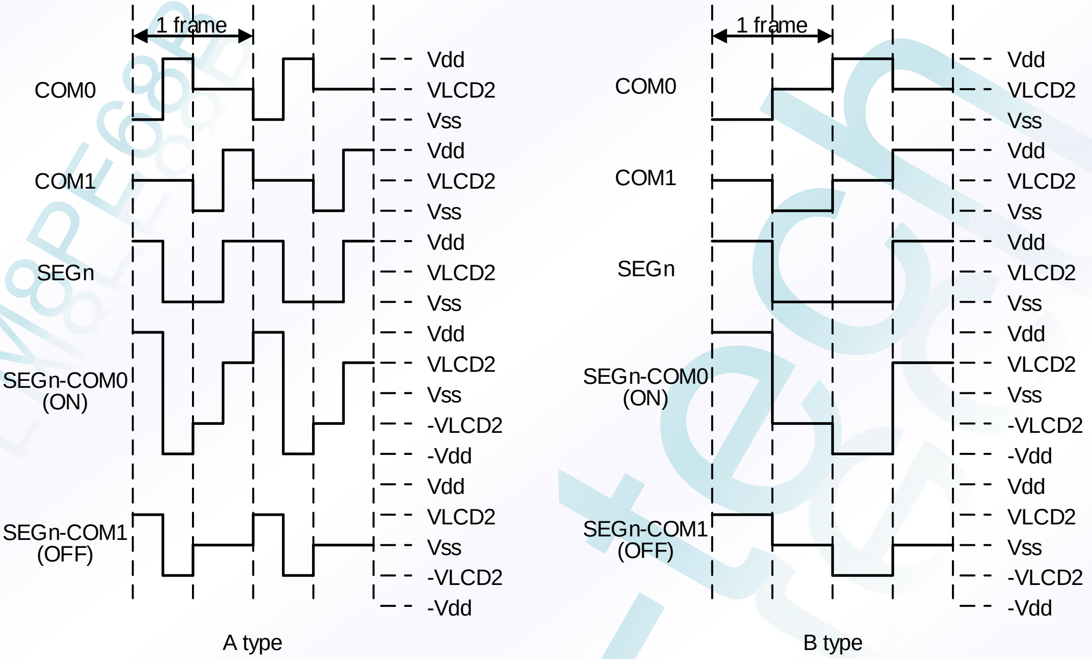


Figure 2.20: LCD Waveform for 1/2 Bias, 1/3 Duty

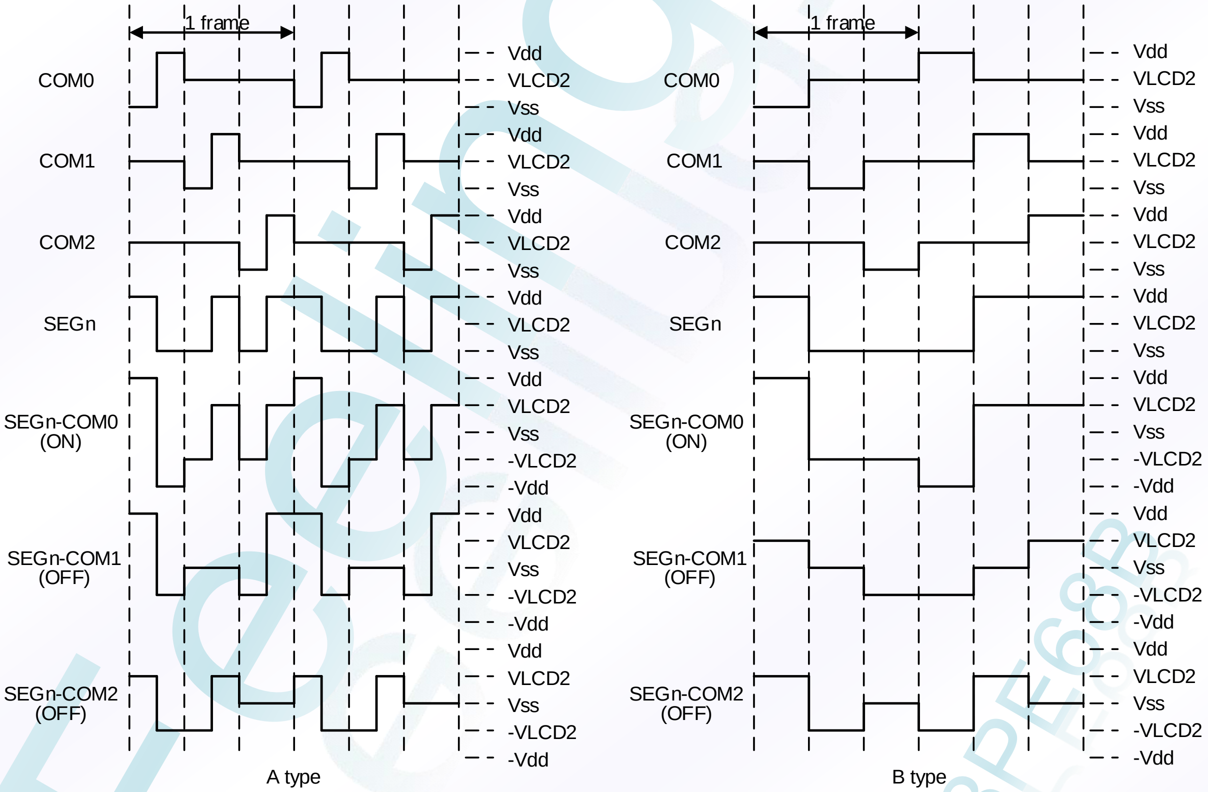


Figure 2.21: LCD Waveform for 1/3 Bias, 1/3 Duty

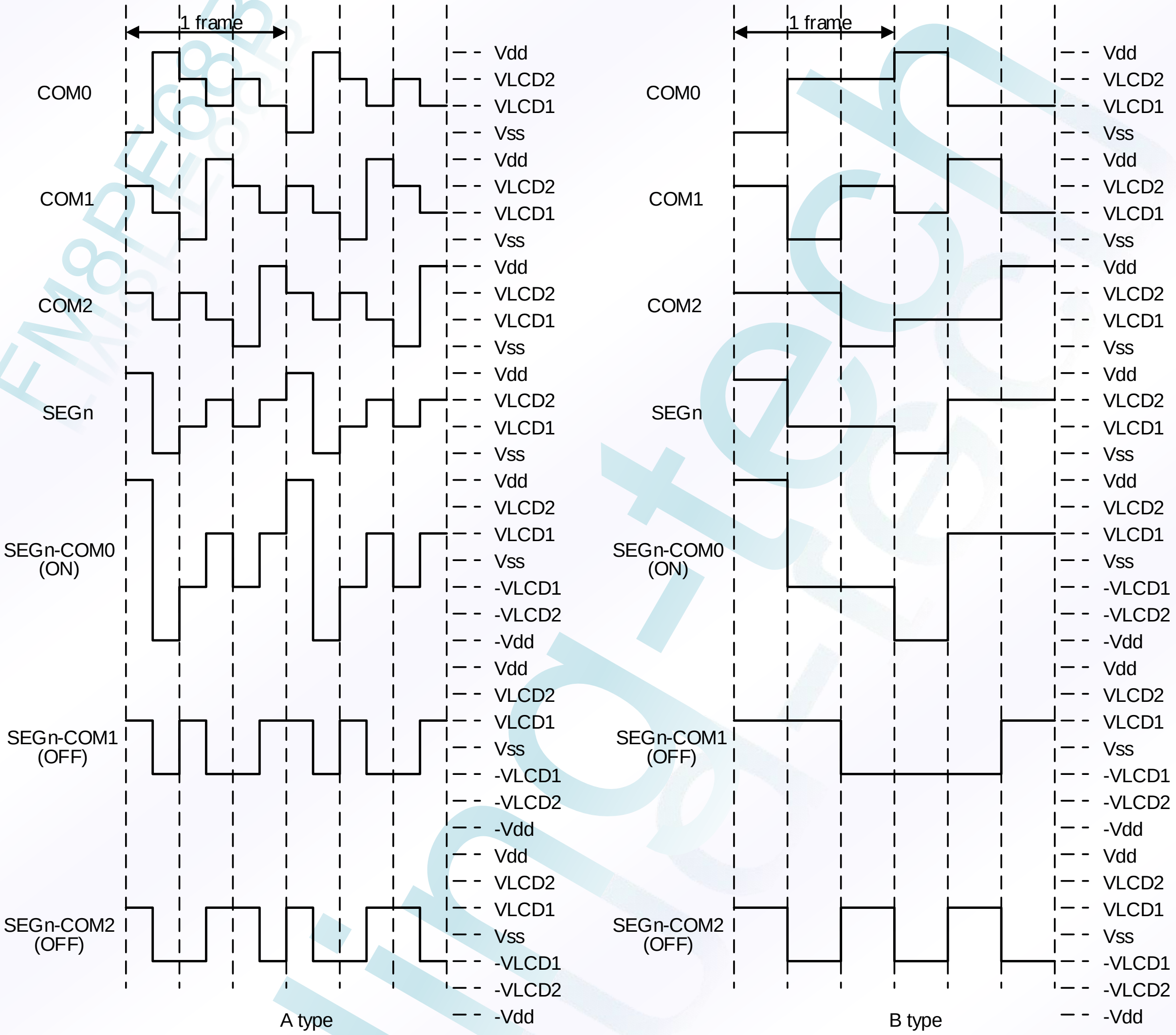
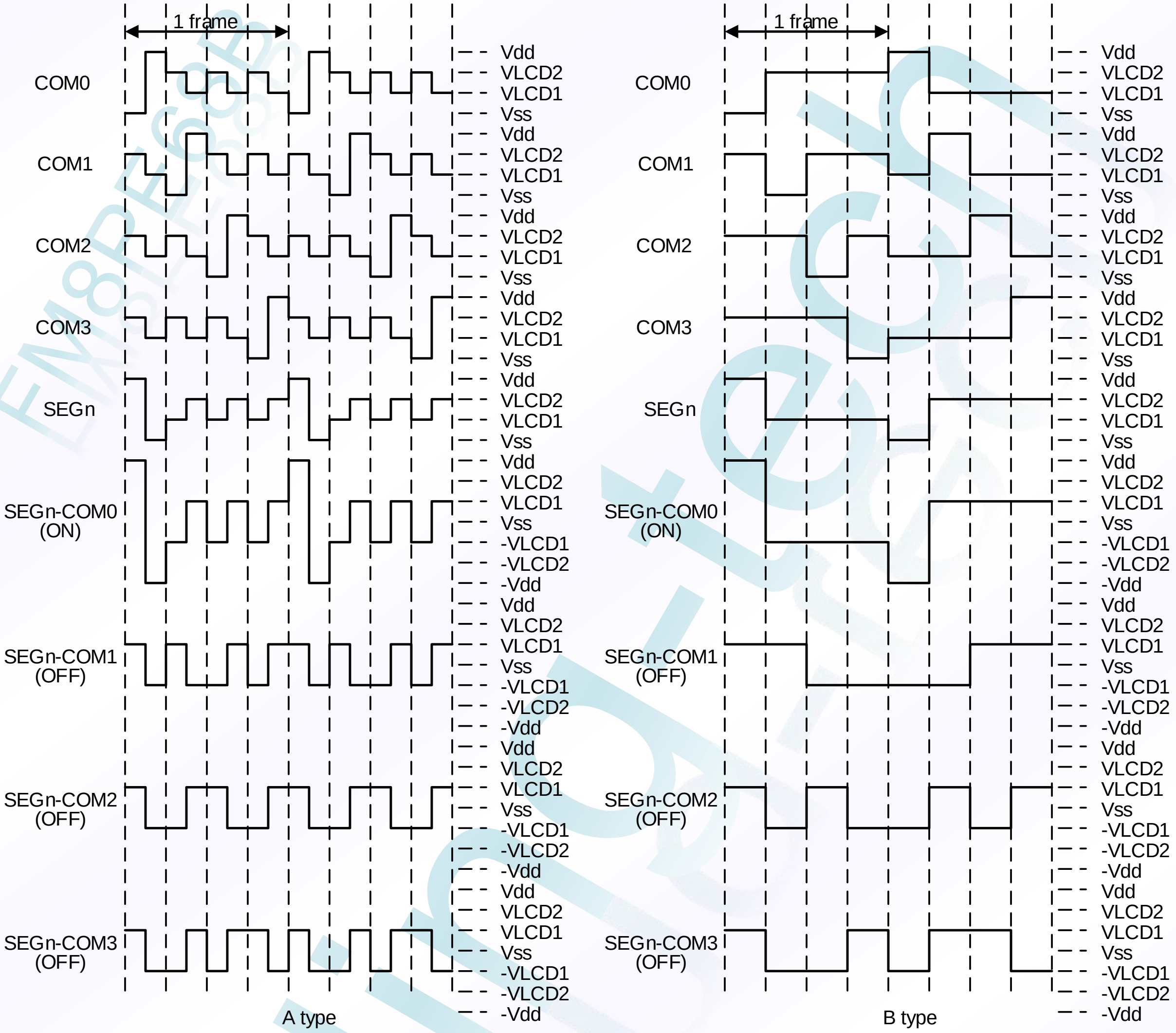


Figure 2.22: LCD Waveform for 1/3 Bias, 1/4 Duty



2.8 Resistor to Frequency Converter (RFC)

The Resistor to Frequency Converter (RFC) can compare nineteen different sensors with the reference resistor separately.
This RFC contains eighteen external pins:
CX: the oscillation Schmitt trigger input (IOA1/CX pin).
RFC0 ~ RFC2: the resistor/sensor output pin 0 ~ 2 (RFC0, RFC1, and RFC2 pins)

Figure 2.23: The Block Diagram of RFC

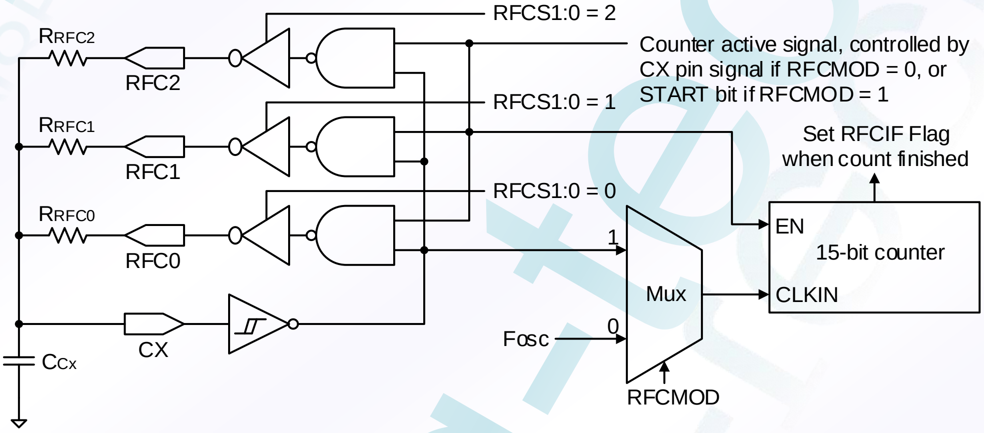

RFCS1:0 = 2
RRFC2
RFC2
Counter active signal, controlled by CX pin signal if RFCMOD = 0, or START bit if RFCMOD = 1
RFCS1:0 = 1
RRFC1
RFC1
Set RFCIF Flag when count finished
RFCS1:0 = 0
RRFC0
RFC0
1
EN
15-bit counter
Mux
CLKIN
CX
CCx
Fosc
0
RFCMOD

Table 2.8: The Description of RFC Control Bits

RFCS1:RFCS0	Select one the RFC oscillation network of RFCx (x = 0 to 2). The selected RFCx pin will be configured as RFCx output pin if RFCON = 1. Other RFCx pins will still behave as tristate input pins. If RFCON = 0, all RFCx pins will behave as tristate input pins.
RFCMOD	= 0, Enable/disable the counter by CX signal, and the clock source of the counter is the internal system clock (F_{OSC}). = 1, Enable/disable the counter by START bit, and the clock source of the counter is the CX signal.
START	= 0, Stop the RFC conversion = 1, RFC counter start to convert. Reset by hardware after conversion is finished. Note: Don't clear START bit by software during the RFC conversion.
RFCON	= 0, Disable RFC module, all the RFCx and CX pins will behave as tristate input pins. = 1, Enable RFC module.

2.8.1 RC Oscillator Network

The RFC circuitry may build up 3 RC oscillation networks through RFC0 to RFC2 and CX pins with external resistors. Only one RC oscillation network may be active at a time. When the oscillation network is built up, the count active pulse will be generated by the oscillation network and transferred to the 15-bit counter through the CX pin. It will then enable or disable the 15-bit counter in order to count the oscillation clock. **The 15-bit RFC counter is cleared when a value is written to RFCCON register, RFCON bit is cleared, and during any kind of reset as well.**

How to build the RC oscillation network:

1. Connect the resistor and capacitor on RFCx (x = 0 to 2, if needed) and CX pins.
2. Switch all of the needed RFCx and CX pins to input mode.
3. Enable the RFC module by set the RFCON bit.
4. Select one of RFCx pins by RFCS1:RFCS0 bits to enable the output pin for RC networks respectively. The selected RFCx will output low at this time. Other RFCx pins will become of a tristate type.

5. Set START bit to enable the RC oscillation network and 15-bit counter. The RC oscillation network will not operate if this bit has not been set. Clear the START bit by H/W or S/W will finish the conversion, and the RFCIF flag will be set if RFCMOD = 0 (if enable).

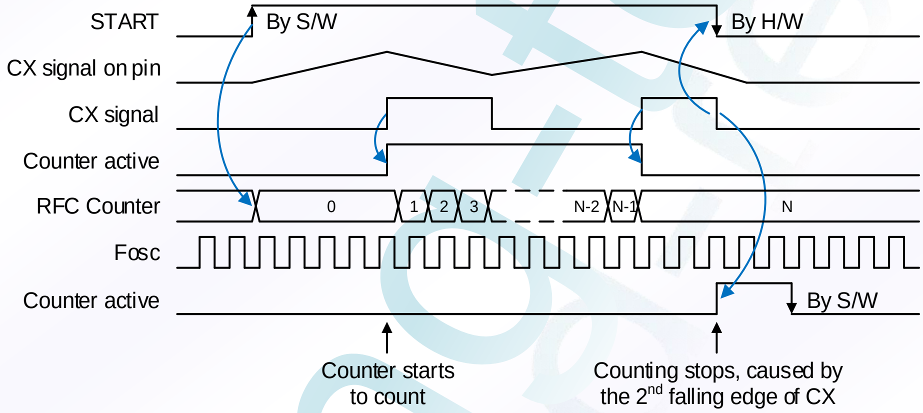
2.8.2 Enable/Disable the Counter by CX Signal

In this mode, CX pin is the signal to control the counter period and the clock source of the counter comes from the internal system clock (FOSC).

The counter will start to count after the first rising edge signal applied on the CX pin after the RFCON bit (RFCCON<7>) is set. Once the second rising edge is applied to the CX pin after the counter is enabled, the counter will stop counting. And after the second falling edge is applied to the CX pin, the RFC block will clear the START bit and set the RFC interrupt flag RFCIF bit (RFCCON<5>) if RFCIE bit (INTEN1<1>) is set.

User also can be polling the RFCON or RFCIF bit to check if the conversion is finished.

Figure 2.24: The Sample of the RFC Counter Controlled by the CX Pin (RFCMOD = 0)

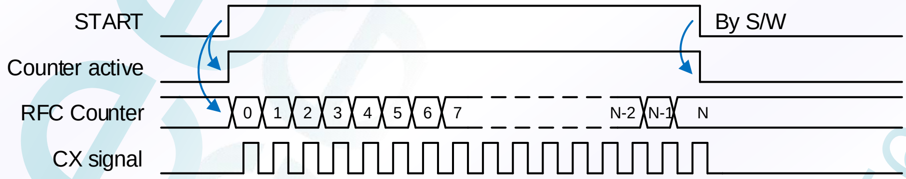


2.8.3 Enable/Disable the Counter by START Bit

In this mode, START bit is the signal to control the counter period and the clock source of the counter comes from the CX pin.

The counter will start to count after the START bit (RFCCON<6>) is set. Once the START bit is cleared by S/W, the counter will stop counting. And after the second falling edge is applied to the CX pin, the RFC block will clear the START bit and set the RFC interrupt flag RFCIF bit (RFCCON<5>) is not needed.

Figure 2.25: The Sample of the RFC Counter Controlled by the START Bit (RFCMOD = 1)



2.9 Reset

FM8PE68B 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 $\overline{\text{TO}}$ and $\overline{\text{PD}}$ 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 16 sub-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 16 sub-oscillator cycle time.

Figure 2.26: Simplified Block Diagram of on-chip Reset Circuit

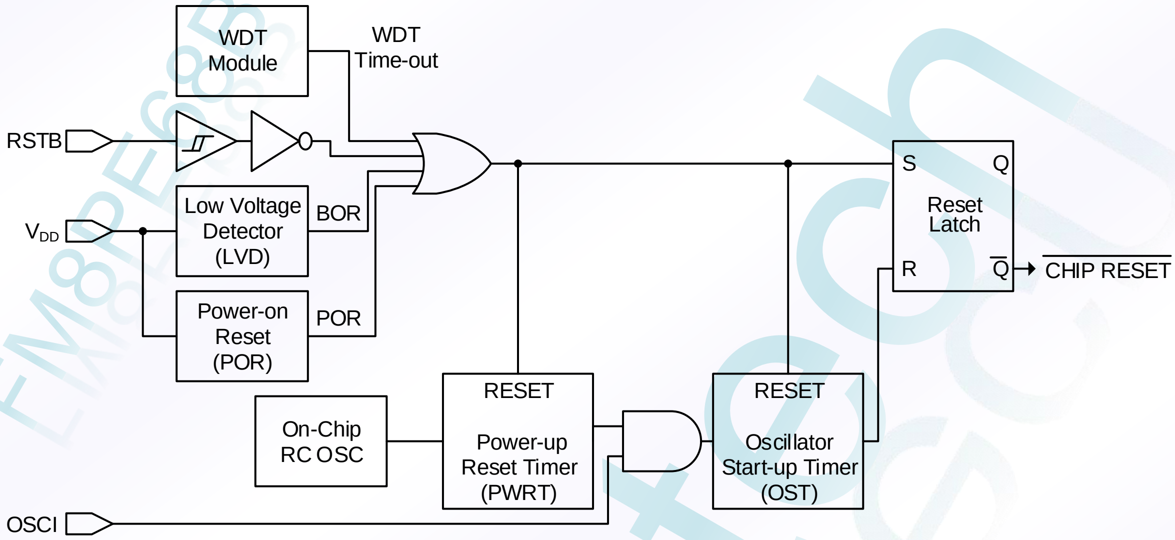
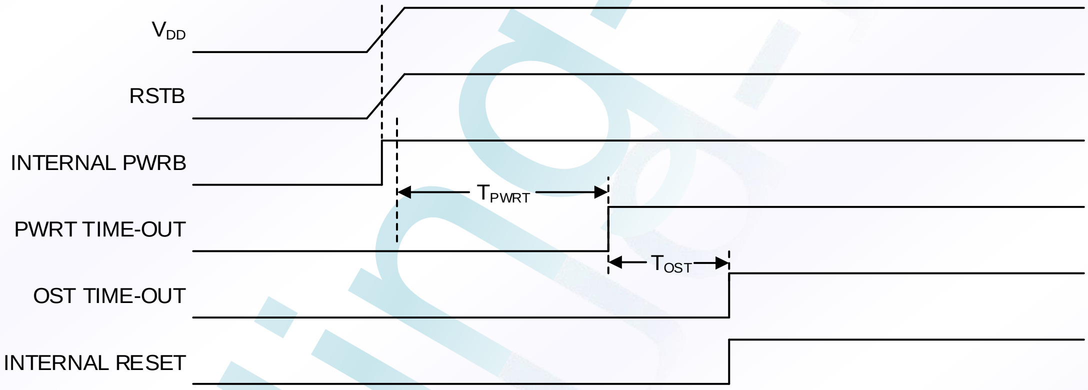
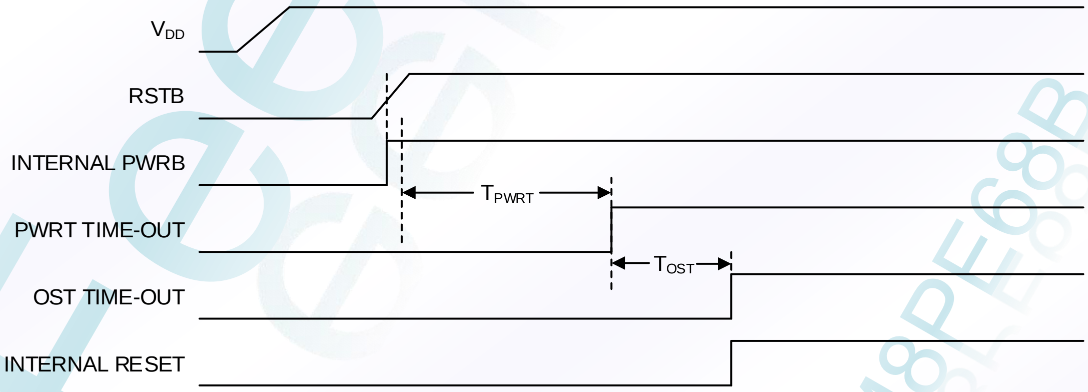


Figure 2.27: Time-out Sequence on Power-up (RSTB Pin Tied to V_{DD})



Note: T_{PWRT} = 18ms; T_{OST} = 16 sub-oscillator cycle time

Figure 2.28: Time-out Sequence on Power-up (RSTB Pin Not Tied to V_{DD})



Note: T_{PWRT} = 18ms; T_{OST} = 16 sub-oscillator cycle time

Table 2.9: Reset Conditions for All Registers

Register	Address	Power-on Reset Brown-out Reset	RSTB Reset WDT Reset	Wake-up from pin change
ACC	N/A	xxxx xxxx	uuuu uuuu	uuuu uuuu
IOSTA	0x05	1111 ----	1111 ----	uuuu ----
IOSTB	0x06	1111 1111	1111 1111	uuuu uuuu
IOSTC	0x07	1111 1111	1111 1111	uuuu uuuu
IOSTD	0x08	1111 1111	1111 1111	uuuu uuuu
DRAMA	0x09	-000 0000	-000 0000	-uuu uuuu
DRAMD	0x0A	xxxx xxxx	uuuu uuuu	uuuu uuuu
C1PR	0x0B	0000 0000	0000 0000	uuuu uuuu
C2PR	0x0C	0000 0000	0000 0000	uuuu uuuu
HPPR	0x0D	0000 0000	0000 0000	uuuu uuuu
LPPR	0x0E	0000 0000	0000 0000	uuuu uuuu
INTEN	0x0F	0000 0000	0000 0000	uuuu uuuu
SEGCON	0x15	1111 0000	1111 0000	uuuu uuuu
WUCON	0x16	0*** 0000	0*** 0000	u*** uuuu
T0CON	0x17	1011 1111	1011 1111	uuuu uuuu
WDTCON	0x18	**** 0111	**** 0111	**** uuuu
C12CON	0x19	0000 0000	0000 0000	uuuu uuuu
HLPCON	0x1A	0000 0000	0000 0000	uuuu uuuu
BPHCON	0x1B	0000 0000	0000 0000	uuuu uuuu
BODCON	0x1C	0000 0000	0000 0000	uuuu uuuu
DPHCON	0x1D	0000 0000	0000 0000	uuuu uuuu
BPDCON	0x1E	0000 0000	0000 0000	uuuu uuuu
INTEN1	0x1F	---- -00-	---- -00-	---- -uu-
INDF	0x00, unbanked	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR0	0x01, unbanked	0000 0000	0000 0000	uuuu uuuu
PCL	0x02, unbanked	0000 0000	0000 0000	See Table 2.3
STATUS	0x03, unbanked	*001 1xxx	*00# #uuu	*uu# #uuu
FSR	0x04, unbanked	00xx xxxx	00uu uuuu	uuuu uuuu
PORTA	0x05, unbanked	xxxx --*-	xxxx --*-	uuuu --*-
PORTB	0x06, unbanked	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTC	0x07, unbanked	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTD	0x08, unbanked	xxxx xxxx	xxxx xxxx	uuuu uuuu
LCDCON	0x09, bank 0 & 2	1100 -000	1100 -000	uuuu -uuu
LCDA	0x0A, bank 0 & 2	---0 0000	---0 0000	---u uuuu
LCDD	0x0B, bank 0 & 2	xxxx xxxx	xxxx uuuu	xxxx uuuu
CNTCON	0x0C, bank 0 & 2	xxxx 0000	xxxx 0000	xxxx uuuu
SYSCON	0x0D, bank 0 & 2	x000 100%	x000 100%	xuuu uuuu
IRCON	0x0E, bank 0 & 2	000- 0000	000- 0000	uuu- uuuu
PORTE	0x09, bank 1 & 3	1111 1111	1111 1111	uuuu uuuu
PORTF	0x0A, bank 1 & 3	1111 1111	1111 1111	uuuu uuuu
RFCCON	0x0B, bank 1 & 3	0000 --00	0000 --00	uuuu --uu
RFCDL	0x0C, bank 1 & 3	0000 0000	0000 0000	uuuu uuuu

Register	Address	Power-on Reset Brown-out Reset	RSTB Reset WDT Reset	Wake-up from pin change
RFCDH	0x0D, bank 1 & 3	0000 0000	0000 0000	uuuu uuuu
DIVCON	0x0E, bank 1 & 3	010- ----	010- ----	uuu- ----
INTFLAG	0x0F, unbanked	0000 0000	0000 0000	uuuu uuuu
General Purpose Registers	0x10 ~ 0x3F	xxxx xxxx	uuuu uuuu	uuuu uuuu

Legend: u = unchanged, x = unknown, - and * = unimplemented, # = refer to the following table for possible values.
% = refer to the configuration bit “HLFS”.

Table 2.10: $\overline{TO}$ / $\overline{PD}$ 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 2.11: 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 FM8PE68B. 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  <8PE68B.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     0x30,A   ;Convert the content of ACC to decimal format, and restored to ACC
                           ;The result in the ACC is "00" and the carry bit C is "1". This represents
                           ;the decimal number "100"
          …
```

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  <8PE68B.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     0x30,A   ;Convert the content of ACC to decimal format, and restored to ACC
                           ;The result in the ACC is "90" and the carry bit C is "0". This represents
                           ;the decimal number " -10"
          …
```

2.11 Oscillator Configurations

FM8PE68B can be operated in four different oscillator modes, and two different sub-oscillator modes. Users can program F_{OSC} configuration bit to select the appropriate modes:

- Selectable main-oscillator options:
  - ERIC: External Resistor/Internal Capacitor Oscillator
  - XT: Crystal/Resonator Oscillator
  - LF: Low Frequency Crystal Oscillator
  - PLL: Phase lock loop

- Selectable sub-oscillator options:
  - ERIC: External Resistor/Internal Capacitor Oscillator
  - LF: Low Frequency Crystal Oscillator

In LF, or XT modes, a crystal or ceramic resonator in connected to the OSCI/XIN and OSCO/XOUT pins to establish oscillation. When in LF or XT modes, the devices can have an external clock source drive the OSCI pin.
The ERIC device option offers additional cost savings for timing insensitive applications. The RC oscillator frequency is a function of the resistor (Rext), and the process parameter.
In PLL mode, connect 0.01uF capacitor to the OSCI pin and V_{SS}.

Figure 2.29: HF, XT or LF Oscillator Modes (Crystal Operation or Ceramic Resonator)

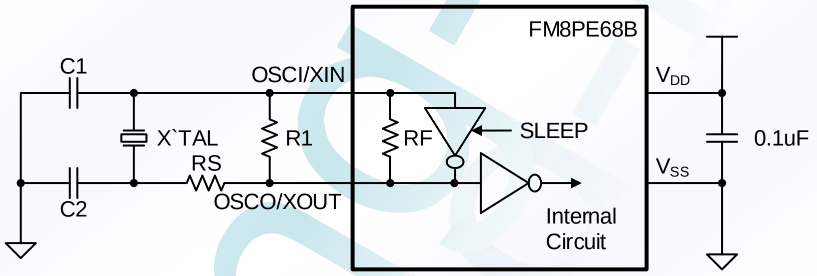


Figure 2.30: HF, XT or LF Oscillator Modes (External Clock Input Operation)

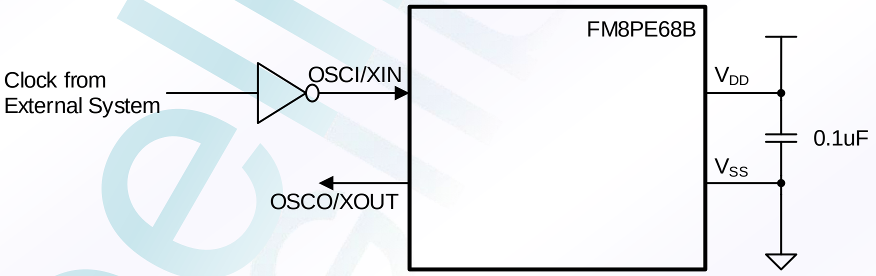
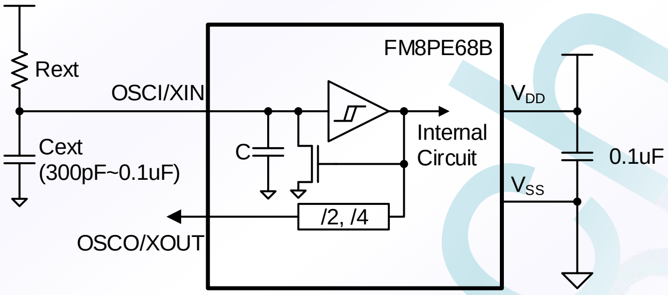


Figure 2.31: ERIC Oscillator Mode (External R, Internal C Oscillator)

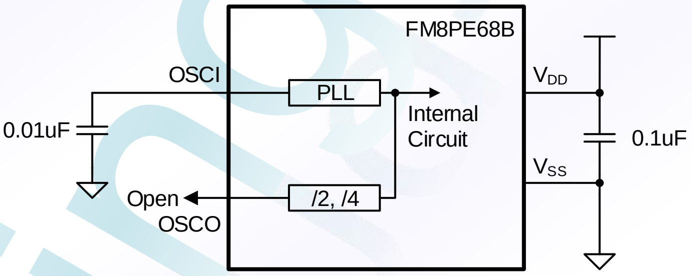


The typical oscillator frequency vs. external resistor is as following table
When Cext = 0.01uf (103)

Frequency	Rext @ 3V	Rext @ 5V
455KHZ	392.3K	447.4K
1MHZ	201.4K	216.8K
4MHZ	52.1K	54.7K
8MHZ	25.4K	26.6K
16MHZ	NA	12.4K

Note: Values are provided for design reference only.

Figure 2.32: PLL Oscillator Mode



2.12 Configuration Words

Table 2.12: Configuration Words

Name	Description
Main_Fosc	Main-Oscillator Selection Bit → ERIC mode (external R & internal C) (default) → XT mode → LF mode → PLL mode
Sub_Fosc	Sub-Oscillator Selection Bit → ERIC mode (external R & internal C) → LF mode
WDTEN	Watchdog Timer Enable Bit → WDT enabled (default) → WDT disabled
PROTECT	Code Protection Bit → OTP code protection off (default) → OTP code protection on
LVDT	Low Voltage Detector Selection Bit → Enable, LVDT voltage = 2.4V (default) → Enable, LVDT voltage = 2.2V, controlled by SLEEP → Enable, LVDT voltage = 2.2V → Enable, LVDT voltage = 2.0V, controlled by SLEEP → Enable, LVDT voltage = 2.0V → Enable, LVDT voltage = 1.8V, controlled by SLEEP → Enable, LVDT voltage = 1.8V
OSCD	Instruction Period Selection Bits → Four oscillator periods (default) → Two oscillator periods
HLFS	Main or Sub-oscillator Selection Bit → CPU is set to select main-oscillator when reset occurred (default) → CPU is set to select sub-oscillator when reset occurred
CYES	Cycle Selection for CALL and GOTO instruction → 2 instruction cycles (default) → 1 instruction cycles
TYPE	Type Selection Bit → 64-pins (A/B Type) is selected (default) → 44-pins (C Type) is selected
RDPORT	Read Port Control Bit for Output Pins → From registers (default) → From pins
SCHMITT	I/O Pin Input Buffer Control Bit → With Schmitt-trigger (default) → Without Schmitt-trigger
WDTS	Watchdog Timer Clock Source in Sleep Mode → Fs (default) → Internal RC
PCHS	Program Counter High Bits Operation Selection Bit for Instruction with PCL as Destination → PC<11:10> = PG<1:0>; PC<9:8> is unchanged. (default) → PC<11: 8> is unchanged.
WUTRIG	Wake-up Trigger Control Bit → Falling edge trigger (default) → Low level trigger

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	$\overline{TO}$,$\overline{PD}$
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	-
INT		S/W interrupt	PC + 1 → Top of Stack 0x002 → PC	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 PG<1:0> → 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
MOV2	**R, d**	Move R	R → dest	1	-
DECR	**R, d**	Decrement R	R - 1 → dest	1	Z
DECRSZ	**R, d**	Decrement R, Skip if 0	R - 1 → dest, Skip if result = 0	1/2/3$^{(1)}$	-
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 → PG<1:0>	1	-
CALL I	Call subroutine	PC + 1 → Top of Stack, I → PC<9:0> PG<1:0> → PC<11:10>	2	-
GOTO I	Unconditional branch	I → PC<9:0> PG<1:0> → PC<11:10>	2	-
FCALL I	Call subroutine	PC + 1 → Top of Stack, I → PC<11:0> I<11:10> → PG<1:0>	3	-
FGOTO I	Unconditional branch	I → PC<11:0> I<11:10> → PG<1:0>	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
PG: Program Memory Page Select Bits
WDT: Watchdog Timer Counter
GIE: Global interrupt enable bit
$\overline{\text{TO}}$: Time-out bit
$\overline{\text{PD}}$: Power-down bit
C: Carry bit
DC: Half 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≤0x3FF
Operation:	PC + 1 → Top of Stack, I → PC<9:0> PG<1:0> → 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)

DISI	**Clear GIE bit**
Syntax:	DISI
Operands:	None
Operation:	0 → GIE
Status Affected:	None
Description:	Disable interrupt
Cycles:	1

ENI	Set GIE bit
Syntax:	ENI
Operands:	None
Operation:	1 → GIE
Status Affected:	None
Description:	Enable interrupt
Cycles:	1

FCALL	Subroutine Call
Syntax:	FCALL I
Operands:	0x000≤I≤0xFFF
Operation:	PC +1 → Top of Stack; I → PC<11:0> I<11:10> → PG<1:0>
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> → PG<1:0>
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> PG<1:0> → 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 <9:0>.
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 = 0x05~0x0F or 0x15~0x1F
Operation:	ACC → IOST register R
Status Affected:	None
Description:	IOST register ‘R’ (R= 0x05~0x0F or 0x15~0x1F) is loaded with the contents of the ACC register.
Cycles:	1

IOSTR	Read IOST Register
Syntax:	IOST R
Operands:	R = 0x05~0x0F or 0x15~0x1F
Operation:	IOST register R → ACC
Status Affected:	None
Description:	The ACC register is loaded with the contents of IOST register ‘R’ (0x05~0x0F or 0x15~0x1F).
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

PAGE	Move Immediate to program page bits
Syntax:	PAGE I
Operands:	0x0≤I≤0x3
Operation:	I → PG<1:0>
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 two-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 and its pre-scaler 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 PG<1:0> → 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

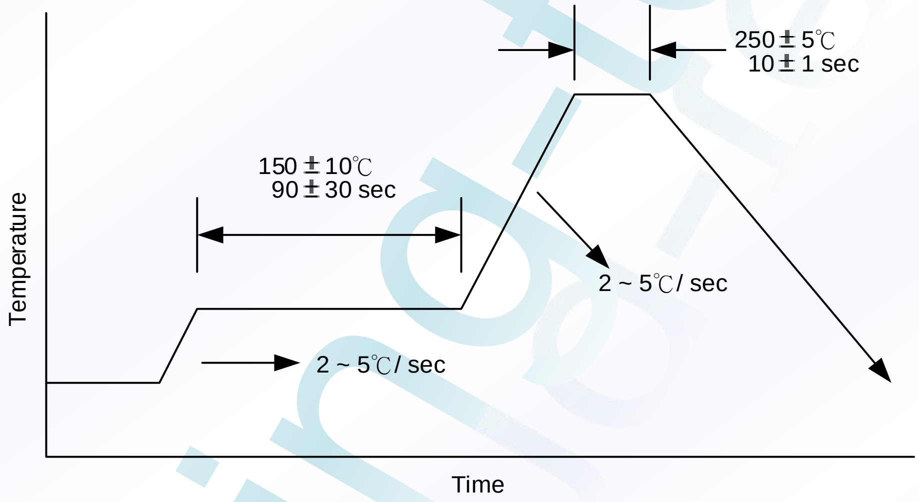
XORIA	Exclusive OR Immediate with ACC
Syntax:	XORIA I
Operands:	0x00≤I≤0xFF
Operation:	ACC xor I → ACC
Status Affected:	Z
Description:	The contents of the ACC register are XOR'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

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	-	KV
		MM (Machine Mode)	-	200	-	V
	Lead Temperature	Soldering, 10 Sec	-	-	250	°C

This table need update

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				
$F_{M\ ERIC}$	Main ERIC Oscillation range	3V	ERIC mode	DC	-	8	M_{HZ}
		5V		DC	-	16	
$F_{M\ XT}$	Main X'tal Oscillation range	3V	XT mode	0.4	-	8	M_{HZ}
		5V		0.4	-	16	
$F_{M\ LF}$	Main X'tal Oscillation range	3V	LF mode	32	-	455	K_{HZ}
		5V		32	-	455	
$F_{M\ PLL}$	Main PLL Oscillation range	3V	PLL mode, F_S=32K_{HZ}	0.5	-	8	M_{HZ}
		5V		0.5	-	8	
$F_{S\ ERIC}$	Sub ERIC Oscillation range	3V	ERIC mode	DC	-	455	K_{HZ}
		5V		DC	-	455	
$F_{S\ LF}$	Sub X'tal Oscillation range	3V	LF mode	32	-	32	K_{HZ}
		5V		32	-	455	
T_{WDT}	WDT period time	3V	Pre-scaler rate=1:1	-	20	-	mS
		4V		-	16.7	-	
		5V		-	14.8	-	

Note: 1. In the PLL mode, to maintain the accuracy of the internal PLL oscillator frequency, a 300pF ~ 0.01μF 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

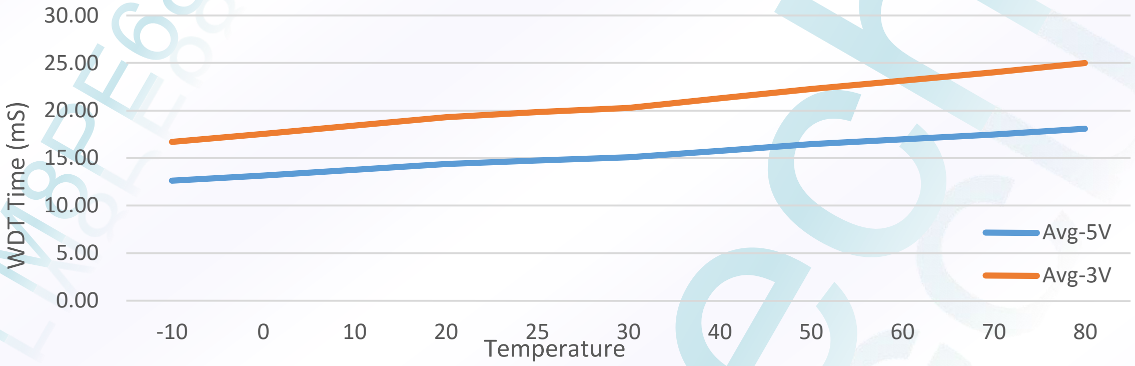
Under Operating Conditions, at two clock instruction cycles and WDT & LVDT are disable, I/O output float, no LCD load.

Symbol	Description	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
V_{IH1}	Input high voltage with Schmitt-trigger, I/O Ports	3V	-	-	1.36	V_{DD}	V
		5V		2.1	-	V_{DD}	
	Input high voltage with Schmitt-trigger, RSTB Pin	3V	-	-	1.73	V_{DD}	
		5V		3.5	-	V_{DD}	
V_{IH2}	Input high voltage without Schmitt-trigger, I/O Ports	3V	-	-	1.13	V_{DD}	V
		5V		-	1.6	V_{DD}	
V_{IL1}	Input low voltage with Schmitt-trigger, I/O Ports	3V	-	V_{SS}	0.99	-	V
		5V		V_{SS}		0.9	
	Input low voltage with Schmitt-trigger, RSTB Pin	3V	-	V_{SS}	1.21	-	
		5V		V_{SS}	1.35	-	
V_{IL2}	Input low voltage without Schmitt-trigger, I/O Ports	3V	-	V_{SS}	1.06	-	V
		5V		V_{SS}	1.48	-	
I_{OH}	I/O Ports Drive current, I/O Ports, IOA7 Pin (IRSC=0)	3V	V_{OH}=0.9V_{DD}	-	1.44	-	mA
		5V		1.0	3.69	-	
	Only IOA7 Pin (IRSC=1)	3V	V_{OH}=0.9V_{DD}	-	2.9	-	
		5V		-	7.6	-	
I_{OL}	I/O Ports Sink current, I/O Ports, IOA7 Pin (IRSC=0)	3V	V_{OL}=0.1V_{DD}	-	7.7	-	mA
		5V		15	18.1	-	

Symbol	Description	Test Conditions		Min.	Typ.	Max.	Unit
		V_{DD}	Conditions				
I_{OL}	Only IOA7 Pin (IRSC=1)	3V	V_{OL}=0.1V_{DD}	-	14.4	-	mA
		5V		-	32.4	-	
I_{PH}	Pull-high current	3V	Input pin at V_{SS}	-	22	-	uA
		5V		65	80	95	
I_{PL}	Pull-low current	3V	Input pin at V_{DD}	-	21.5	-	uA
		5V		55	70	85	
V_{LVDT}	Low Voltage Detector Voltage	-	LVDT=2.4V	2.04	2.4	2.76	V
		-	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	
	Low Voltage Reset Voltage	-	LVDT=2.4V	1.87	2.2	2.53	
		-	LVDT=2.2V	1.7	2.0	2.3	
		-	LVDT=2.0V	1.6	1.8	2.07	
		-	LVDT=1.8V	1.6	1.7	1.84	
I_{LVDT}	LVDT current	3V	LVDT=2.4V	-	0.13	-	uA
		5V		-	0.38	-	
		3V	LVDT=2.2V	-	0.13	-	
		5V		-	0.42	-	
		3V	LVDT=2.0V	-	0.14	-	
		5V		-	0.37	-	
		3V	LVDT=1.8V	-	0.12	-	
		5V		-	0.38	-	
I_{WDT}	WDT current	3V	Sleep mode, Pre-scaler rate=1:256	-	0.6	-	uA
		5V		-	3.7	-	
I_{SB}	Sleep mode (Power down) current	3V	-	-	<1	-	uA
		5V		-	<1	1	
I_{DD1}	IDLE mode current	3V	F_S=32K_{HZ}, LCD=Enable	-	5.8	-	uA
		5V		-	15.8	-	
I_{DD2}	Green mode current	3V	F_S=32K_{HZ}, LCD=Enable	-	90.4	-	uA
		5V		-	240.1	-	
I_{DD3}	Normal mode current	3V	F_S=32K_{HZ}, F_M=8M_{HZ}, LCD=Enable	-	2.2	-	mA
		5V	F_S=32K_{HZ}, F_M=16M_{HZ}, LCD=Enable	-	7.6	-	

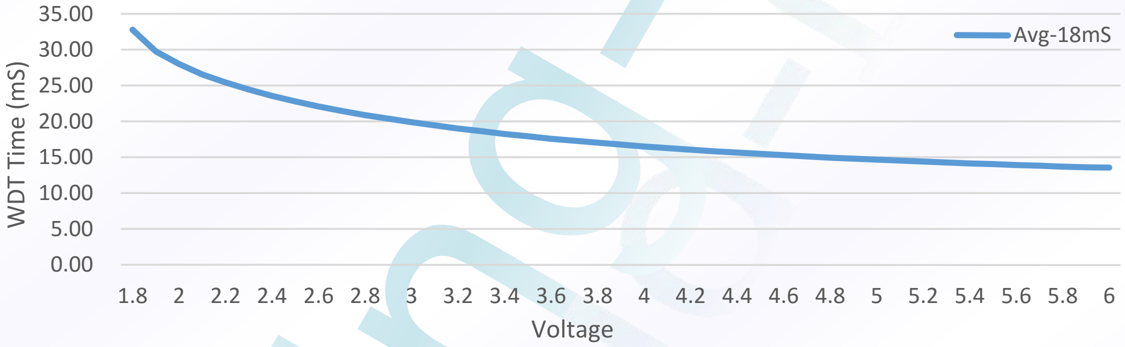
6.3 ELECTRICAL CHARACTERISTICS Charts of FM8PE68B

6.3.1 WDT 18mS Reset time vs. Temperature



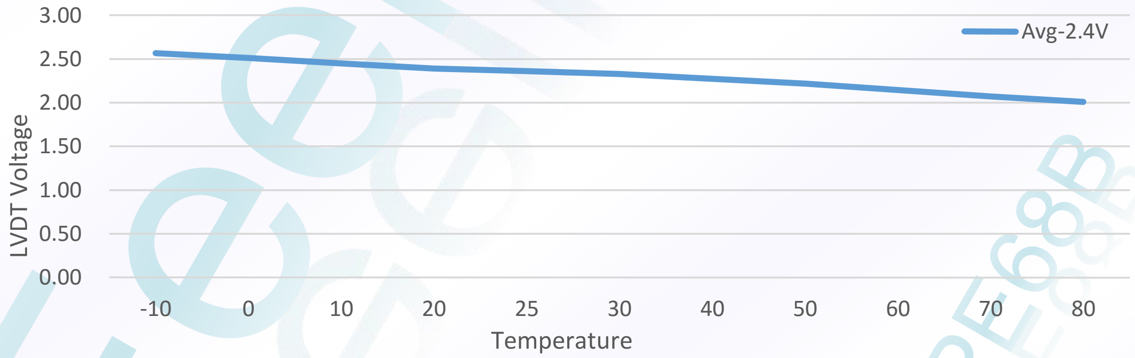
Note: Curves are for design reference only.

6.3.2 WDT 18mS Reset time vs. Supply Voltage (Ta=25℃)



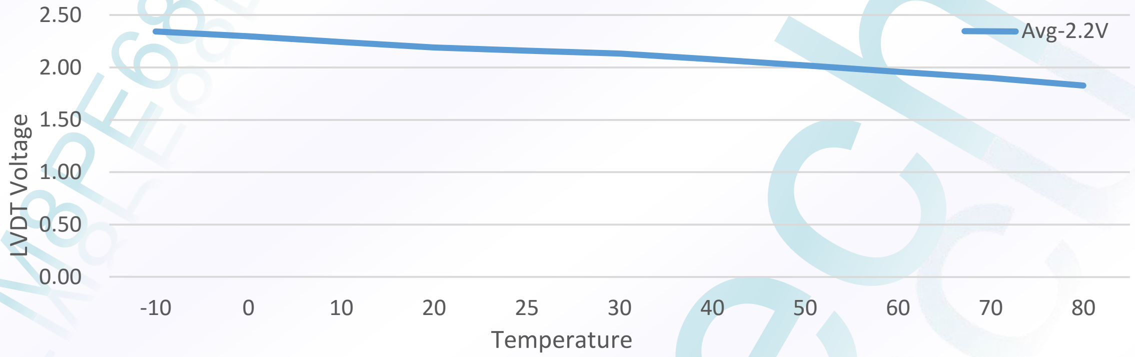
Note: Curves are for design reference only.

6.3.3 LVDT 2.4V vs. Temperature



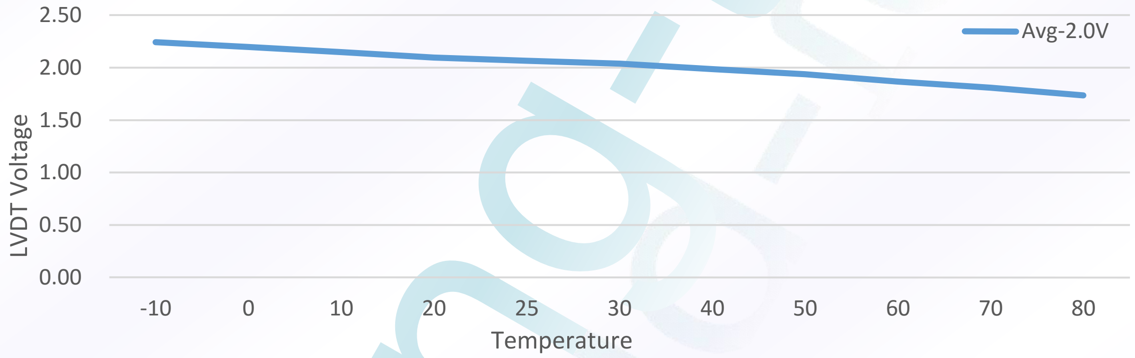
Note: Curves are for design reference only.

6.3.4 LVDT 2.2V vs. Temperature



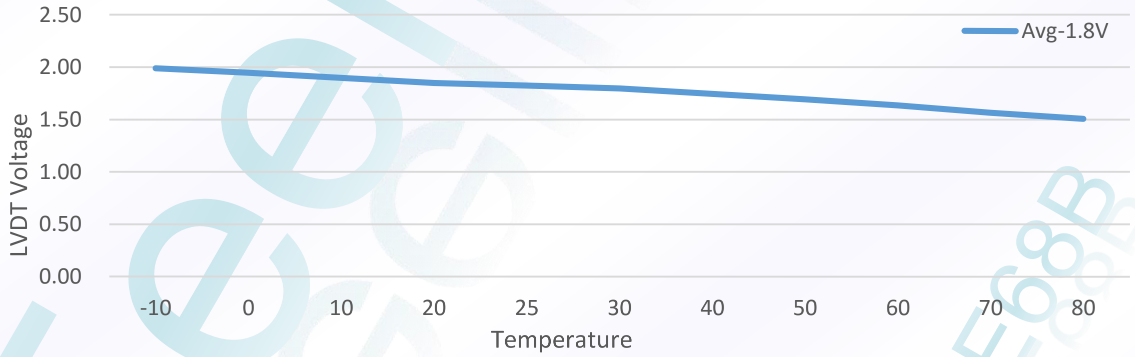
Note: Curves are for design reference only.

6.3.5 LVDT 2.0V vs. Temperature



Note: Curves are for design reference only.

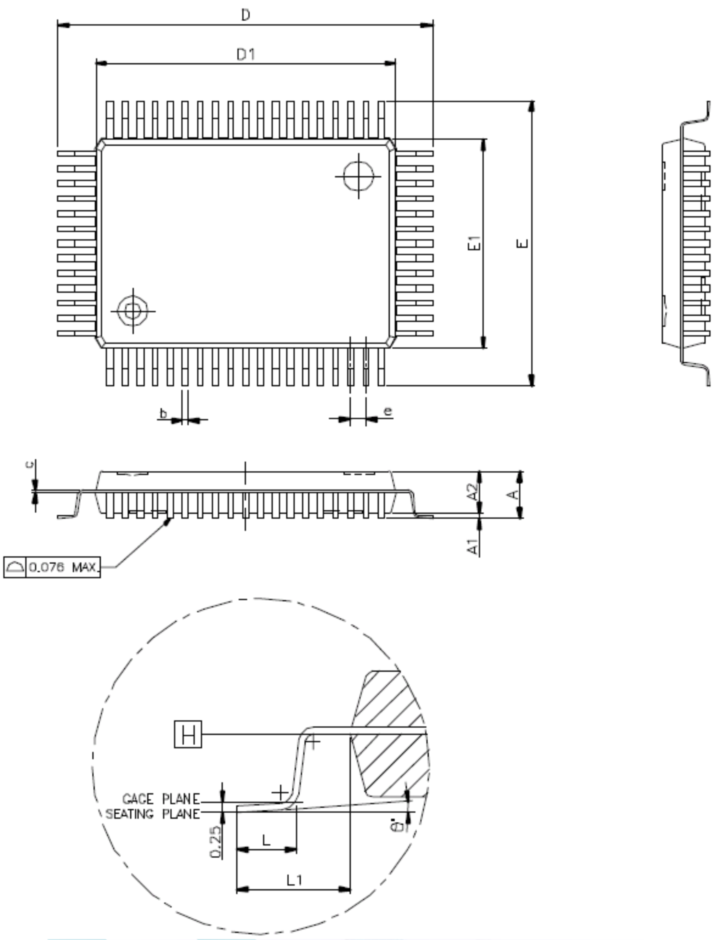
6.3.6 LVDT 1.8V vs. Temperature



Note: Curves are for design reference only.

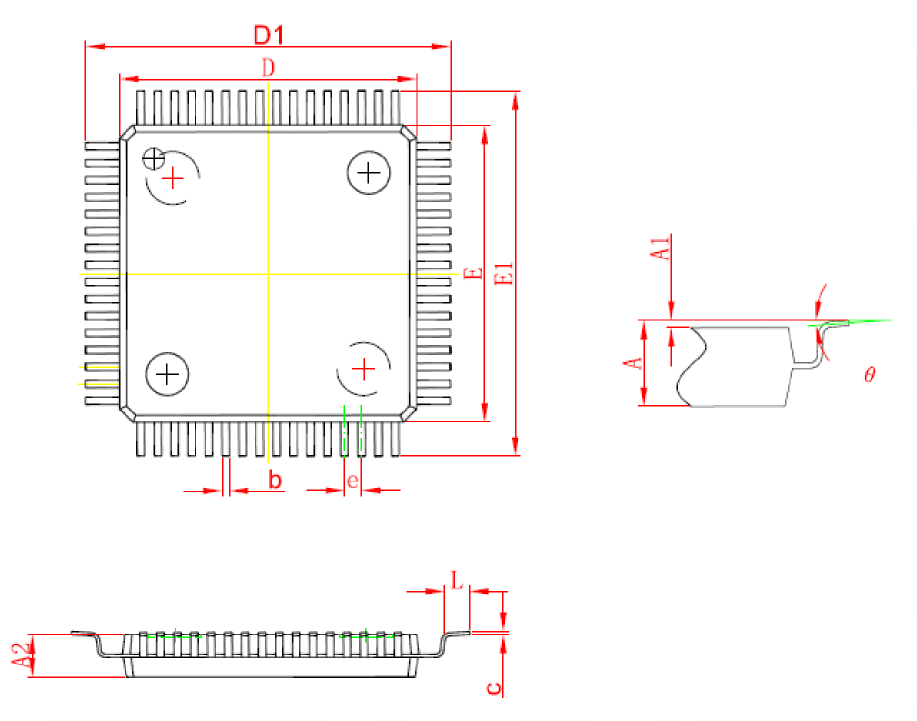
7.0 PACKAGE DIMENSION

7.1 64-PIN QFp(14x20)



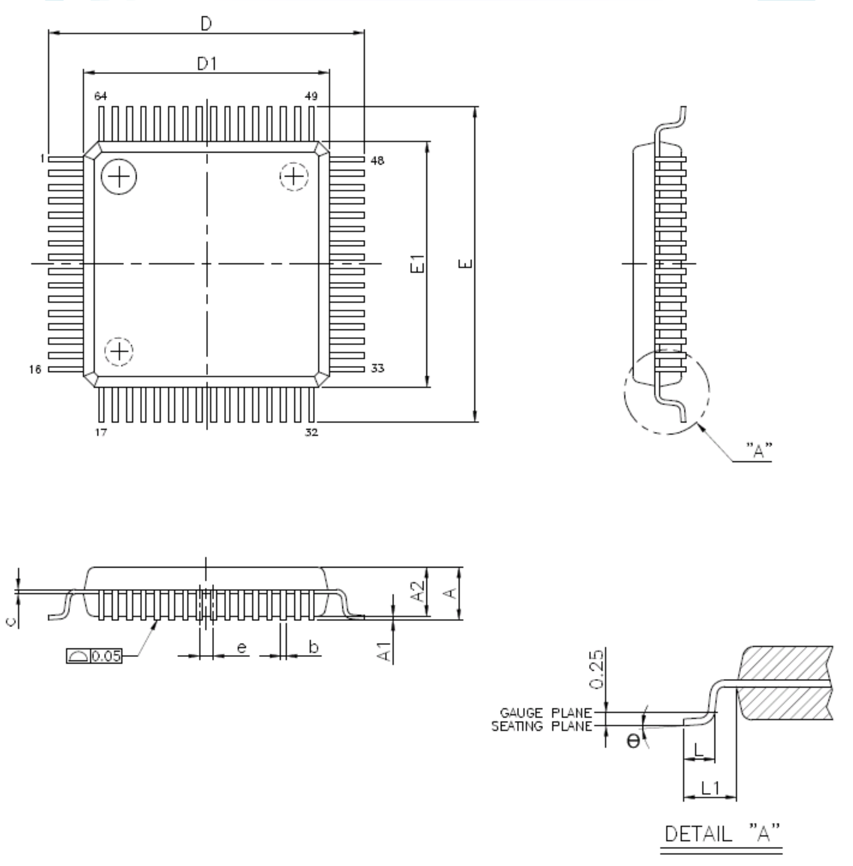
Symbols	Dimension In MM		
	Min	Nom	Max
A	-	-	3.40
A1	0.25	-	-
A2	2.55	2.72	3.05
b	0.35	0.40	0.50
c	0.11	0.15	0.23
D	25.00 BASIC		
D1	20.00 BASIC		
e	1.00 BASIC		
E	19.00 BASIC		
E1	14.00 BASIC		
L	1.15	1.30	1.45
L1	2.50 REF		
θ°	0°	3.5°	7°

7.2 64-PIN QFP(14x14)



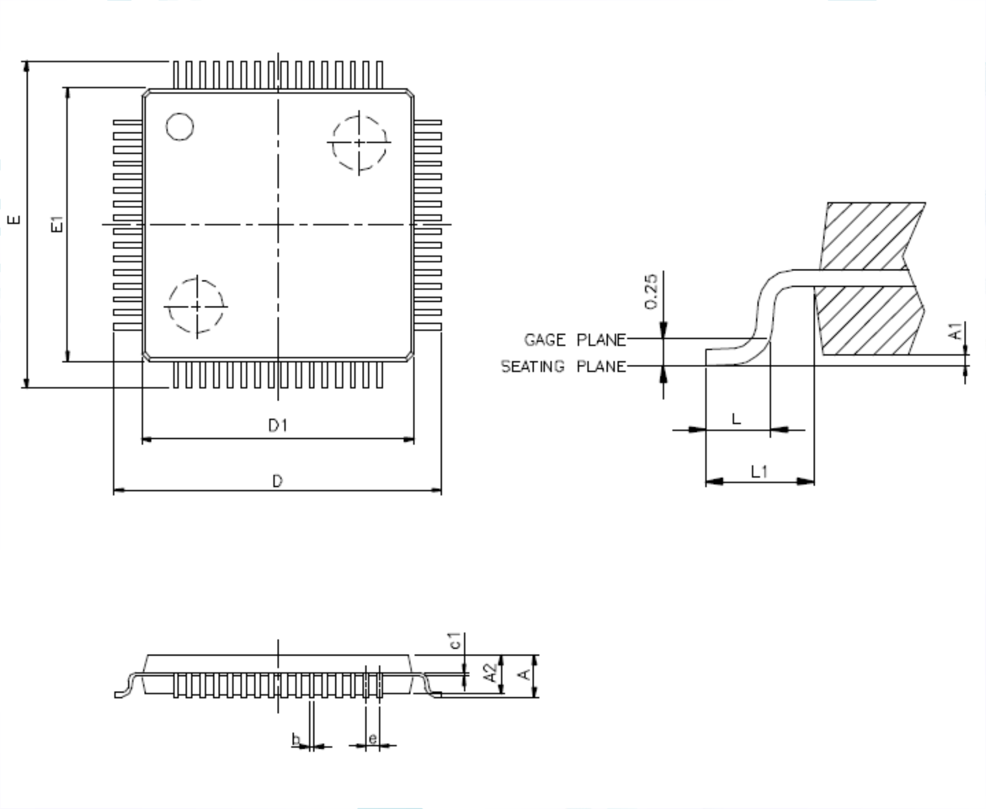
Symbols	Dimension In MM		
	Min	Nom	Max
A	-	-	2.45
A1	0.05	-	0.25
A2	1.8	-	2.2
b	0.3	-	0.45
c	0.11	-	0.23
D	14.00 BASIC		
D1	17.00 BASIC		
e	0.800 BASIC		
E	14.00 BASIC		
E1	17.00 BASIC		
L	0.73	-	1.03
θ°	0°	-	7°

7.3 64-PIN QFP(7x7)



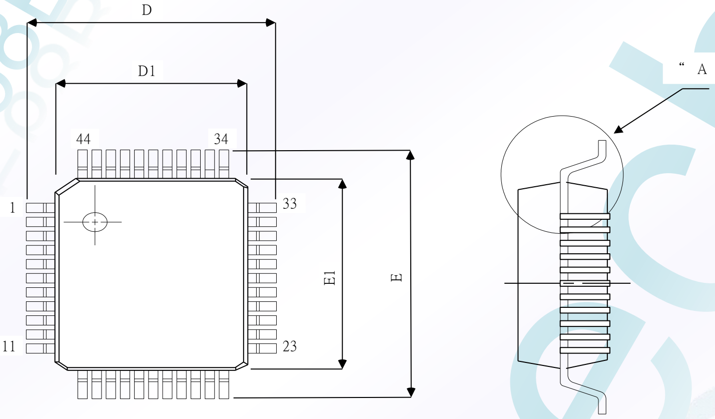
Symbols	Dimension In MM		
	Min	Nom	Max
A	-	-	1.6
A1	0.05	-	0.15
A2	1.35	1.40	1.45
b	0.13	0.18	0.23
c	0.09	-	0.20
D	9.00 BASIC		
D1	7.00 BASIC		
e	0.40 BASIC		
E	9.00 BASIC		
E1	7.00 BASIC		
L	0.45	0.60	0.75
L1	1.00 REF		
θ°	0°	3.5°	7°

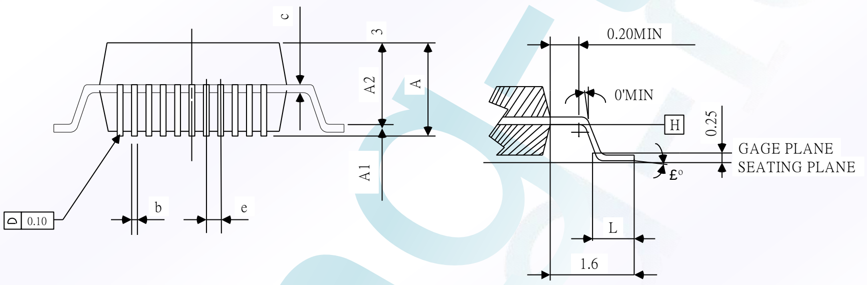
7.4 64-PIN QFP(10x10)



Symbols	Dimension In MM		
	Min	Nom	Max
A	-	-	1.6
A1	0.05	-	0.15
A2	1.35	1.40	1.45
b	0.17	0.18	0.27
c1	0.09	-	0.16
D	12.00 BASIC		
D1	10.00 BASIC		
e	0.50 BASIC		
E	12.00 BASIC		
E1	10.00 BASIC		
L	0.45	-	0.75
L1	1.00 REF		

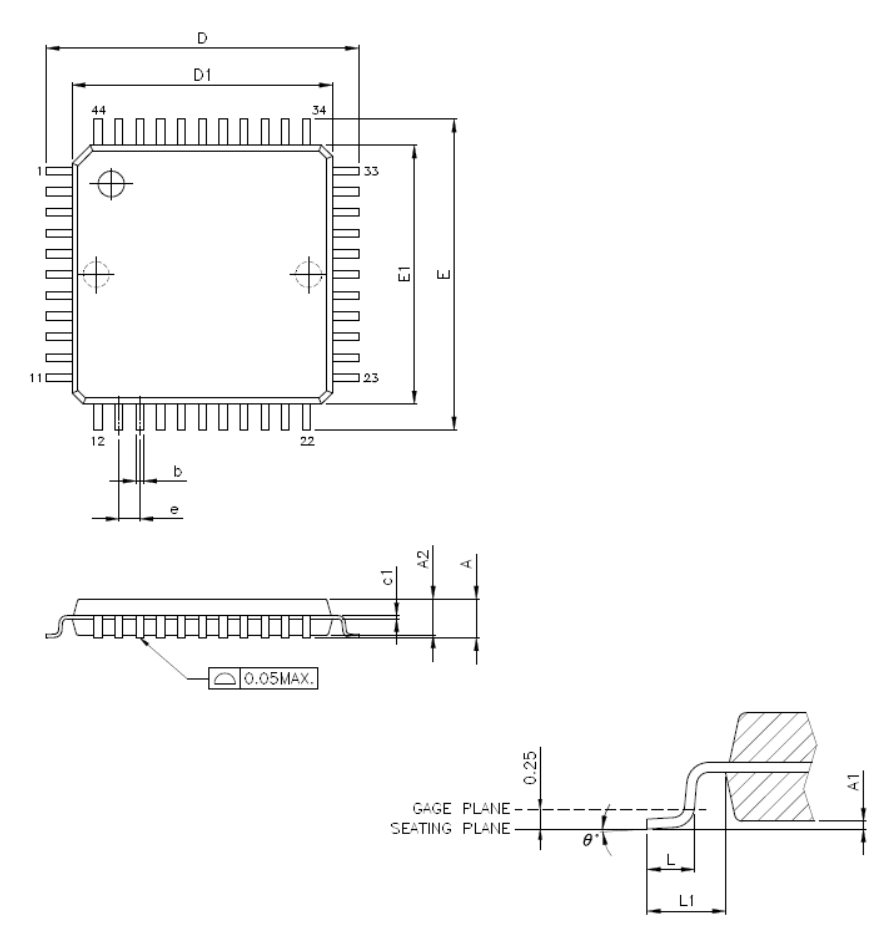
7.5 44-PIN QFP(10x10)





Symbols	Dimension In Millimeters		
	Min	Nom	Max
A	-	-	2.7
A1	0.25	-	0.50
A2	1.9	2.0	2.2
b	0.3 (TYP)		
D	13.00	13.20	13.40
D1	9.9	10.00	10.10
E	13.00	13.20	13.40
E1	9.9	10.00	10.10
L	0.73	0.88	0.93
e	0.80 (TYP)		
θº	0º	-	7º
C	0.1	0.15	0.2

7.6 44-PIN LQFP(10x10)



Symbols	Dimension In Millimeters		
	Min	Nom	Max
A	-	-	1.60
A1	0.05	-	0.15
A2	1.35	1.40	1.45
c1	0.09	-	0.16
D	12.00 BSC		
D1	10.00 BSC		
E	12.00 BSC		
E1	10.00 BSC		
e	0.80 BSC		
b	0.30	0.37	0.45
L	0.45	0.60	0.75
L1	1.00 REF		
θ°	0°	3.5°	7°

8.0 ORDERING INFORMATION

OTP Type MCU	Package Type	Pin Count	Package Size	SAMPLE Stock
FM8PE68BAF	QFP	64	14mm x 20mm	Available
FM8PE68BAG	LQFP	64	10mm x 10mm	No stock
FM8PE68BBG	LQFP	64	7mm x 7mm	No stock
FM8PE68BCF	QFP	44	10mm x 10mm	No stock
FM8PE68BCG	LQFP	44	10mm x 10mm	No stock