FM8PU86LK

Full-Speed USB Controller

Datasheet V05

March 12 2010

Version Control

Version	Date.	Description
V01	2009-04-08	Release of prototype
V02	2009-10-14	Modify Register 15h,16h default value.
V03	2009-10-23	Modify Register 07h default value.
V05	2010-03-12	Updated Package type.

Contents

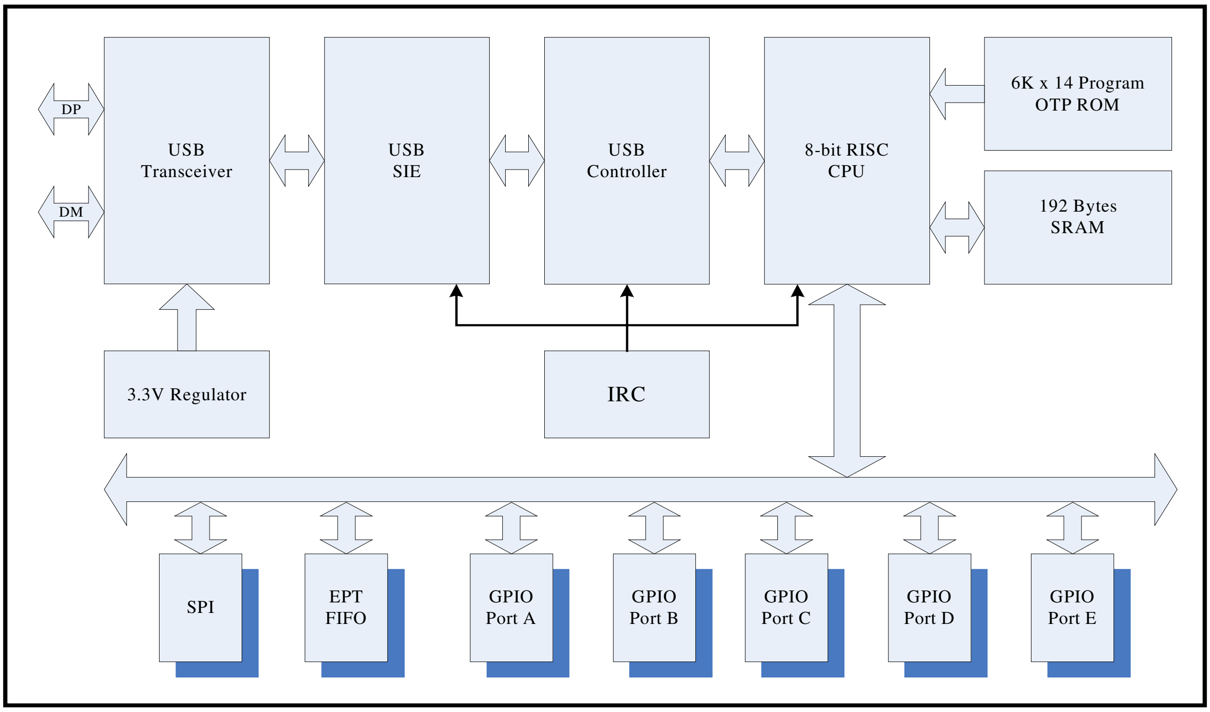
1.0 General Description

The FM8PU86LK is an 8-bit microprocessor embedded device tailored to the USB application. It includes an 8-bit RISC CPU core, 192 byte SRAM, Full Speed and Low Speed USB Interface and a 6K x 14 internal program OTP-ROM.

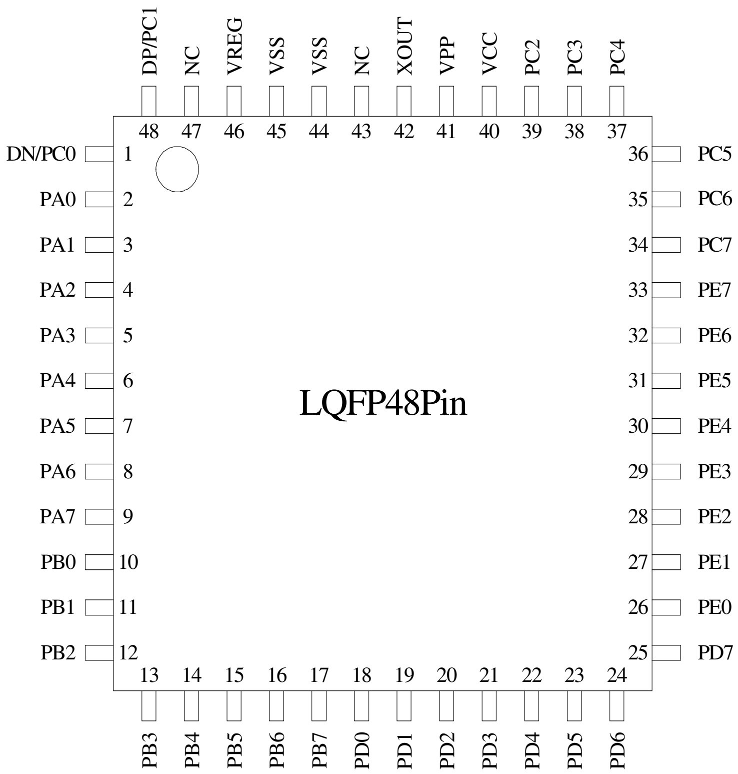
2.0 Features

- USB Specification Compliance
  - Support Full-Speed and Low-Speed USB 2.0 specification.
  - Built-in USB Transceiver.
  - Built-in 1.5k pull-high resistor.
  - Support one address and four 8-byte and two 32-byte data endpoints.
  - Support USB Suspend and Resume function.
  - One Control IN/OUT endpoints, four INT/BULK endpoints.
- 8-bit RISC microprocessor
  - 6K x 14 internal program OTP-ROM.
  - 256 bytes internal SRAM.
  - Optional 12MHz/24MHz internal MCU clock by configuration.
  - Optional Default/16MHz/12MHz/6MHz/58KHz internal MCU clock by firmware.
  - One Watch dog Timer ,Timer0,Timer1,and two captures Timers.
- System Clock
  - Clock Generator
    - 0.25% Accuracy after OTP calibration.
  - Internal slow oscillator 58KHz free run clock for suspend mode.
- 38-I/O ports
  - High current drive on any GPIO pin : 25mA/pin current sink.
  - Each GPIO pin supports high-impedance input, internal pull-ups, open drains output, or CMOS outputs.
- Power Manager and consumption
  - Power on reset (POR) is 2.0V
    - Chip can work on greater than 2.2V power.
  - Low Voltage Detect is 3.8V
  - Regulator 3.3V output, supply 45mA current.
- GPIO supply 1.8V~5V interface.
- Support SSC to reduce EMI.
- Package
  - LQFP-48PIN
    - Type is FM8PU86LKA3 for GPIO 3.3V interface.
    - Type is FM8PU86LKA5 for GPIO 5.0V interface.
  - QFP-44PIN
    - Type is FM8PU86LKB3 for GPIO 3.3V interface.
    - Type is FM8PU86LKB5 for GPIO 5.0V interface
  - DIP-40PIN
    - Type is FM8PU86LKC3 for GPIO 3.3V interface.
    - Type is FM8PU86LKC5 for GPIO 5.0V interface.

3.0 Block Diagram

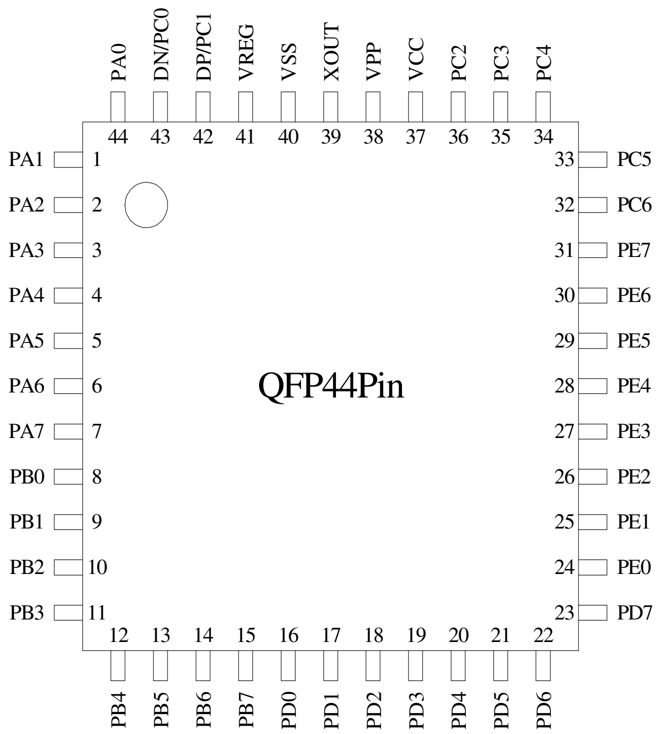


4.0 Pin Definitions



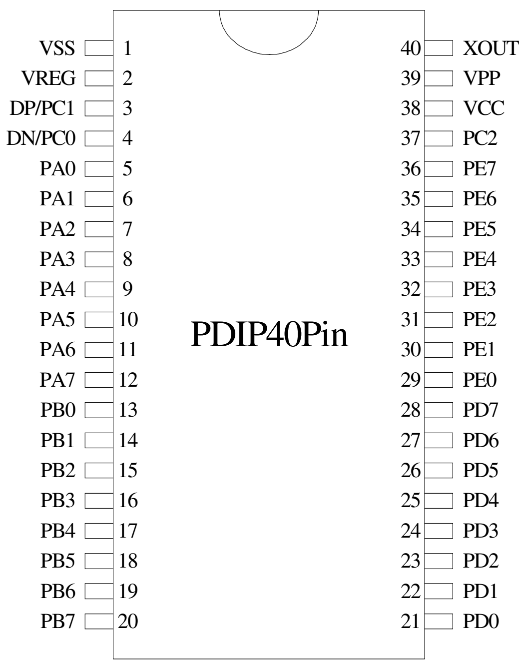
FM8PU86LKA3(48Pin)

FM8PU86LKA5(48Pin)



FM8PU86LKB3(44Pin)

FM8PU86LKB5(44Pin)



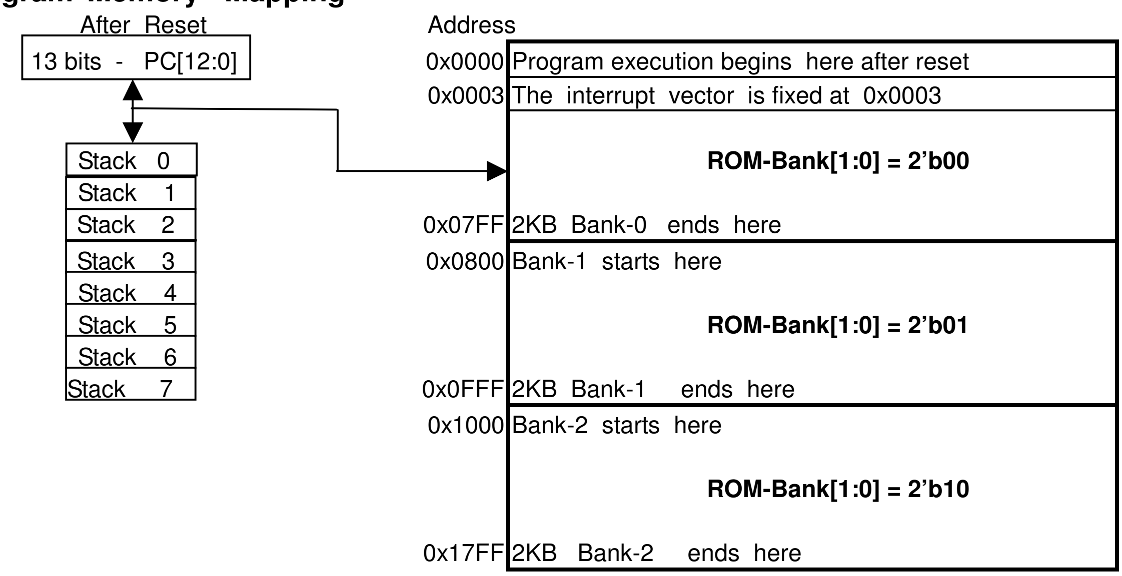
FM8PU86LKC3(40Pin)

FM8PU86LKC5(40Pin)

5.0 Pin Description

PIN Name	I/O	48-pin A	44-pin B	40-pin C	Description	Note
PA[7:0]	IO	2~9	44,1~7	5~12	GPIO Port A capable of sinking up to25 mA/pin, or sinking controlled low or high programmable current. Can also source 2 mA current, provide a resistive pull-up, or serve as a high-impedance input. PA[0] and PA[1] provide inputs to Capture Timers A and B, respectively. Use PA[4]~PA[6] in SPI mode.	
PB[7:0]	IO	10~17	8~15	13~20	GPIO Port B capable of sinking up to 25 mA/pin, or sinking controlled low or high programmable current. Can also source 2 mA current, provide a resistive pull-up, or serve as a high-impedance input.	
PC[7:2]	IO	32~36	32~26	37	GPIO Port C capable of sinking up to 25 mA/pin, or sinking controlled low or high programmable current. Can also source 2 mA current, provide a resistive pull-up, or serve as a high-impedance input.	
PD[7:0]	IO	18~25	16~23	21~28	GPIO Port D capable of sinking up to 25 mA/pin, or sinking controlled low or high programmable current. Can also source 2 mA current, provide a resistive pull-up, or serve as a high-impedance input.	
PE[7:0]	IO	24~31,	24~31	29~36	GPIO Port E capable of sinking up to 25 mA/pin, or sinking controlled low or high programmable current. Can also source 2 mA current, provide a resistive pull-up, or serve as a high-impedance input.	
VSS	G	45,46	40	1	Ground	
VPP OVDD	I P	38	38	39	Programming voltage supply, VCC for normal operation External reset pin , active low. I/O pad power Voltage supply 1.8 ~ 5 V	
VREG	O	46	41	2	3.3V Regulator output	
XOUT	O	10	39	40	24MHz,16MHz, or 12MHz internal oscillator output;	
NC	-	43,47	-	-	Non Connect	
VCC	P	11	37	38	Voltage supply(5V)	
DN SDATA	IO IO	12	43	4	USB differential data lines (D-) PS/2 data signal,PC[1]	
DP SCLK	IO IO	13	42	3	USB differential data lines (D+) PS/2 clock signal,PC[0[	

6.0 Program Memory Mapping



The FM8PU86LK has program memory size greater 2K works, but the CALL and GOTO instructions only have a 11-bits address range. This 11-bit address range allows a branch within a 2K program memory page size. To allow CALL and GOTO instructions to address the entire 6K program memory address range for FM8U86LK, there is another two bits to specify the program memory page. The paging bit comes from the ROMBANK[1:0] bits(PUMODE[6:5]). When doing a CALL or GOTO instruction, the user must ensure that page bit ROMBANK[1:0] are programmed so that the desired program memory page is addressed. When one of return instructions is executed, the entire 13-bit PC is POPed from the stack. Therefore, manipulation of the ROMBANK[1:0] is not required for return instructions.

7.0 Memory And Register Mapping

Table 7-1 I/O Register Mapping

RAMBNK[1:0] / Address	Description			
	0 0 Bank 0	0 1 Bank 1	1 0 Bank 2	1 1 Bank 3
00h	INDF			
01h	TMR0			
02h	PCL			
03h	STATUS			
04h	FSR			
05h	PORTA			
06h	PORTB			
07h	PORTC			
08h	PCON			
09h	PBIE			
0Ah	PBCON			
0Bh	TMCON			
0Ch	INTEN			
0Dh	INTFLAG			
0Eh	INTEN1			
0Fh	INTFLAG1			
10h	T0RLD			
11h	USBADDR			
12h	USBMDCFG			
13h	EP0RXST			
14h	EP0TXST			
15h	EP1TXST			
16h	EP2TXST			
17h	EP3TXST			
18h	EP4RXST			
19h	EPST1			
1Ah	TMR1			
1Bh	T1RLD			
1Ch	SPIRXB			
1Dh	SPITXB			
1Eh	SPISTAT			
1Fh	SPICON			
20H	PUMODE			
21H	PORTD			
22H	PORTE			
23H	CLKCFG			
24H	--			

08h	PAMODE0
09h	PAMODE1
0Ah	PBMODE0
0Bh	PBMODE1
0Ch	PCMODE0
0Dh	PCMODE1
0Eh	PDMODE0
0Fh	PDMODE1
10h	PEMODE0
11h	PEMODE1

<table>
<tr><td rowspan="2">RAMBNK[1:0]
Address</td><td colspan="4">Description</td></tr>
<tr><td>0 0
Bank 0</td><td>0 1
Bank 1</td><td>1 0
Bank 2</td><td>1 1
Bank 3</td></tr>
<tr><td>25H</td><td colspan="4">--</td></tr>
<tr><td>26H</td><td colspan="4">--</td></tr>
<tr><td>27H</td><td colspan="4">--</td></tr>
<tr><td>28H</td><td colspan="4">--</td></tr>
<tr><td>29H</td><td colspan="4">--</td></tr>
<tr><td>2AH</td><td colspan="4">--</td></tr>
<tr><td>2Bh
|
2Fh</td><td colspan="4">General
Purpose
Registers</td></tr>
<tr><td rowspan="2">30h
|
3Fh</td><td colspan="2">ENP0RX</td><td rowspan="2">ENP3TX0</td><td rowspan="2">ENP4RX0</td></tr>
<tr><td colspan="2">ENP0TX</td></tr>
<tr><td rowspan="2">40h
|
4Fh</td><td colspan="2">ENP1TRX</td><td rowspan="2">ENP3TX1</td><td rowspan="2">ENP4RX1</td></tr>
<tr><td colspan="2">ENP2TRX</td></tr>
<tr><td>50h
|
7Fh</td><td>General
Purpose
Registers
0</td><td>General
Purpose
Registers
1</td><td>General
Purpose
Registers
2</td><td>General
Purpose
Registers
3</td></tr>
</table>

TABLE 7-2 : The Registers Controlled IOST or IOSTR Instructions

<table>
<tr><td>Address</td><td>Name</td><td>B7</td><td>B6</td><td>B5</td><td>B4</td><td>B3</td><td>B2</td><td>B1</td><td>B0</td></tr>
<tr><td>08h (R/w)</td><td>PAMODE0</td><td colspan="8" rowspan="2">Port A Mode Control Register</td></tr>
<tr><td>09h (R/w)</td><td>PAMODE1</td></tr>
<tr><td>0Ah (R/w)</td><td>PBMODE0</td><td colspan="8" rowspan="2">Port B Mode Control Register</td></tr>
<tr><td>0Bh (R/w)</td><td>PBMODE1</td></tr>
<tr><td>0Ch (R/w)</td><td>PCMODE0</td><td colspan="8" rowspan="2">Port C Mode Control Register</td></tr>
<tr><td>0Dh (R/w)</td><td>PCMODE1</td></tr>
<tr><td>0Eh (R/w)</td><td>PDMODE0</td><td colspan="8" rowspan="2">Port D Mode Control Register</td></tr>
<tr><td>0Fh(R/w)</td><td>PDMODE1</td></tr>
<tr><td>10h (R/w)</td><td>PEMODE0</td><td colspan="8" rowspan="2">Port E Mode Control Register</td></tr>
<tr><td>11h(R/w)</td><td>PEMODE1</td></tr>
</table>

Accessed by IOST/IOSTR instruction
The port I/O Control Registers are loaded with the contents of the ACC register by executing the IOST R (05h~06h) instruction. By executing the IOSTR instruction, user read these registers into ACC.
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).
The IOST registers are set all '1's(output drivers disabled) upon POWER RESET.

TABLE 7-2.1 Port A and B Output control truth Table

Data register	Mode1	Mode0	Output Drive Strength
1	0	0	HI-z(Input mode)
0			
1	0	1	Normal Drive
0			Sink (8mA)
1	1	0	Resistive(14KΩ)
0			Sink (2mA)
1	1	1	Normal Drive
0			Sink (25mA)

TABLE 7-3: Operational Registers Map

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
00h (r/w)	INDF	Uses contents of FSR to address data memory (not a physical register)							
01h (r/w)	TMR0	8-bit real-time clock/counter							
02h (r/w)	PCL	Low order 8 bits of PC							
03h (r/w)	STATUS	--	--	SELPIN	/TO	/PD	Z	DC	C
04h (r/w)	FSR	--	Indirect data memory address pointer						
05h (r/w)	PORTA	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
06h (r/w)	PORTB	IOB7	IOB6	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0
07h (r)	PORTC	IOC7	IOC6	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0
08h (r/w)	PCON	WDTE	WDTSL	WDTPS2	WDTPS1	WDTPS0	PORE	PDCLK	SIRCEN
09h (r/w)	PBIE	PBIE7	PBIE6	PBIE5	PBIE4	PBIE3	PBIE2	PBIE1	PBIE0
0Ah (r/w)	PBCON	PBCON7	PBCON6	PBCON5	PBCON4	PBCON3	PBCON2	PBCON1	PBCON0
0BH(r/w)	TMCON	T1ON	T1PS2	T1PS1	T1PS0	T0ON	T0PS2	T0PS1	T0PS0
0Ch (r/w)	INTEN	GIE	--	--	RSTIE	SPITXIE	SPIRXIE	T1IE	T0IE
0Dh (r/w)	INTFLAG	PBIF	--	--	RSTIF	SPITXIF	SPIRXIF	T1IF	T0IF
0Eh (r/w)	INTEN1	RSMIE	SUSIE	RX0IE	TX0IE	TX1IE	TX2IE	TX3IE	RX4IE
0Fh (r/w)	INTFLAG1	RSMIF	SUSIF	RX0IF	TX0IF	TX1IF	TX2IF	TX3IF	RX4IF
10h (r/w)	T0RLD	T0RD7	T0RD6	T0RD5	T0RD4	T0RD3	T0RD2	T0RD1	T0RD0
11h(r/w)	USBADDR	EUSBAR	USBAR6	USBAR5	USBAR4	USBAR3	USBAR2	USBAR1	USBAR0
12h(r/w)	USBMDCFG	SUSPMD	RESMD	CTRRD	RX0RDY	EP1CFG	EP2CFG	EP3CFG	EP4CFG
13h(r)	EP0RXST	RX0TGL	RX0ERR	EP0DIR	EP0SET	RX0CN3	RX0CN2	RX0CN1	RX0CN0
14h(r/w)	EP0TXST	TX0RDY	TX0TGL	EP0STAL	--	TX0CN3	TX0CN2	TX0CN1	TX0CN0
15h(r/w)	EP1TXST	TX1RDY	TX1TGL	EP1STAL	EP1DIR	TX1CN3	TX1CN2	TX1CN1	TX1CN0
16h(r/w)	EP2TXST	TX2RDY	TX2TGL	EP2STAL	EP2DIR	TX2CN3	TX2CN2	TX2CN1	TX2CN0
17h(r/w)	EP3TXST	TX3RDY	TX3TGL	TX3CN5	TX3CN4	TX3CN3	TX3CN2	TX3CN1	TX3CN0
18h(r)	EP4RXST	RX4TGL	RX4ERR	RX4CN5	RX4CN4	RX4CN3	RX4CN2	RX4CN1	RX4CN0
19h(r/w)	EPST1	RX4RDY	EP4STAL	EP3STAL	LVDT	LVR	CLKSW	SELCLK1	SELCLK0
1Ah(r/w)	TMR1	TM17	TM16	TM15	TM14	TM13	TM12	TM11	TM10
1Bh(r/w)	T1RLD	T1RD7	T1RD6	T1RD5	T1RD4	T1RD3	T1RD2	T1RD1	T1RD0
1Ch(r)	SPIRXB	RX7	RX6	RX5	RX4	RX3	RX2	RX1	RX0
1Dh(r)	SPITXB	TX7	TX6	TX5	TX4	TX3	TX2	TX1	TX0
1Eh(r/w)	SPISTAT	--	--	TXBF	TM1IF	SDOOD	SCKOD	--	RXBF
1FH(r/w)	SPICON	CKEDG	SPION	RXOV	SSE	SSEMOD	SPIM2	SPIM1	SPIM0
20h(r/w)	PUMODE	USBMD	ROMBNK1	ROMBNK0	--	RAMBNK1	RAMBNK0	PS2EN	USBEN
21h(r/w)	PORTD	IOD7	IOD6	IOD5	IOD4	IOD3	IOD2	IOD1	IOD0
22h(r/w)	PORTE	IOE7	IOE6	IOE5	IOE4	IOE3	IOE2	IOE1	IOE0
23h(r/w)	CLKCFG	SELACK	DPMF1	DPMF0	SELINT2	SELINT1	SELINT0	INCODS	IRCEN

Legend: - = unimplemented, read as ‘0’,

Address 00H : Indirect Addressing Register (INDF)

Bits	Description	Read	Write	Default
7~0	Uses contents of FSR to address data memory	Yes	Yes	00h

Address 01H : Timer0(TMR0)

Bits	Description	Read	Write	Default
7~0	Timer 0	Yes	Yes	00h

Address 02H: Low bytes of Program Counter(PCL)

Bits	Description	Read	Write	Default
7~0	Low bytes of Program Counter(7~0)	Yes	Yes	00h

Address 03H : Status Register(STATUS)

Bits	Name	Description	Read	Write	Default
7~6	Revered	Revered	No	No	0
5	SELPIN	PortA,B,C,D,E register updated selection. 0 : updated port's register From port's register. 1 : From pin of portA,B,C,D,E to updated.	Yes	Yes	0
4	/TO	WDT overflow flag bit (0: active)	Yes	No	1
3	/PD	Power down flag bit (0: active)	Yes	No	1
2	Z	Zero flag	Yes	Yes	0
1	DC	Decimal carry flag or decimal/borrow flag	Yes	Yes	0
0	C	Carry flag or/borrow flag	Yes	Yes	0

Address 04H :File select Register(FSR)

Bits	Name	Description	Read	Write	Default
7	Revered	Revered	No	No	0
6~0	FSR	File select register to define address in indirect addressing mode	Yes	Yes	0

Address 05H : Port A (PORTA)

Bits	Name	Description	Read	Write	Default
7~0	PORTA	Port A data input/output	Yes	Yes	00h

Address 06H : Port A (PORTB)

Bits	Name	Description	Read	Write	Default
7~0	PORTB	Port B data input/output	Yes	Yes	00h

Address 07H :Port C (PORTC)

Bits	Name	Description	Read	Write	Default
7~2	PORTC[7:2]	IPortC data input/output	Yes	YES	00h
1	PORTC[1]	D+(SCLK).The state of the D+ pins can be read at Port C data register.	Yes	Yes	1
0	PORTC[0]	D-(SDATA).The state of the D- pins can be read at Port C data register.	Yes	Yes	1

Address 08H : Power Control Register (PCON)

Bits	Name	Description	Read	Write	Default
7	WDTE	Watch-dog timer enable (0:disable,1:enable)	Yes	Yes	1
6	WDTSL	Watch-dog timer out select . 1: If Watch-dog timer out ,the Device be reset.	Yes	Yes	1

		PC=0000h; 0: If Watch-dog timer out ,Device only SUSPMD(12.7H) be clear by H/w. PC= PC + 1;			
5~3	WDTPS[2:0]	WDT prescaler	Yes	Yes	0h
2	PORE	Power on reset detector 1: enable 0:disable	Yes	Yes	1h
1	PDCLK	SIE clock enable , 1: SIE clock disable 0: SIE clock enable	Yes	Yes	0h
0	SIRCEN	Slow IRC(58Khz) enable. 1: Enable SIRC 0:Disable SIRC	Yes	Yes	1h

WDTPS[2:0]	WDT time out
3'b000	18ms*1 = 18ms
3'b001	18ms*2 = 36ms
3'b010	18ms*4 = 72ms
3'b011	18ms*8 = 144ms
3'b100	18ms*16 = 288ms
3'b101	18ms*32 = 572ms
3'b110	18ms*64 = 1152ms
3'b111	18ms*128 = 2304ms

TABLE 7-4 WDT Prescaler

Address 09H : Port B Interrupt Control Register (PBIE)

Bits	Name	Description	Read	Write	Default
7~0	PBIE	Port B or Port A interrupt enable bits (0:disable ,1:enable)	Yes	Yes	00h

Note : The SELINT[2:0] of CLKCFG(28h) register bits (select PORTA,PORTB,PORTC,PORTD,PORTE intterrupt)

Address 0AH : Port B Wake-up Control Register (PBCON)

Bits	Name	Description	Read	Write	Default
7~0	PBCON	0 : falling edge(port B or port A), 1: Pin-changed (port B or port A)	Yes	Yes	00h

Address 0BH : Timer control Register(TMCON)

Bits	Name	Description	Read	Write	Default
7	T1ON	Timer1 module Enable bit	Yes	Yes	0
6~4	T1PS2~0	Timer1 Prescaler Rate 0:1/2 ,1:1/4 ,2:1/8 ,3:1/16,4:1/32,5:1/64,1:1/128,7:1/256	Yes	Yes	0h
3	T0ON	Timer0 module Enable bit	Yes	Yes	0
2~0	T0PS2~0	Timer0 Prescaler Rate 0:1/2 ,1:1/4 ,2:1/8 ,3:1/16,4:1/32,5:1/64,6:1/128,7:1/256	Yes	Yes	0h

T0/1 SP2,SP1,SP0	Prescaler Rate	Number of MCU clock
3'b000	1/2	4
3'b001	1/4	8
3'b010	1/8	16
3'b011	1/16	32
3'b100	1/32	64
3'b101	1/64	128
3'b110	1/128	256
3'b111	1/256	512

TABLE 7-5 Timer0/1 Prescaler

Address 0CH : Interrupt Mask Register(INTEN)

Bits	Name	Description	Read	Write	Default
7	GIE	Global Interrupt enable bit (0:disable,1:enable)	Yes	Yes	0
6	Revered	Revered	Yes	No	0
5	SOFIE	SOF interrupt enable bit.	Yes	Yes	0
4	RSTIE	USB bus reset interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
3	SPITXIE	SPI Transmit Interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
2	SPIRXIE	SPI Receive Interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
1	T1IE	Timer1 Interrupt enable bit. (0:disable,1:enable).	Yes	Yes	0
0	T0IE	Timer0 Interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0

Address 0DH : Interrupt Flag Register(INTFLAG)

Bits	Name	Description	Read	Write	Default
7	PBIF	Port B interrupt flag, write 0 clear flag	Yes	Yes	0
6	Revered	Revered	Yes	No	0
5	SOFIF	SOF packet is received flag, write 0 clear flag.	Yes	Yes	0
4	RSTIF	USB bus reset interrupt flag, write 0 clear flag	Yes	Yes	0
3	SPITXIF	SPI Transmit Interrupt flag ,write 0 clear flag	Yes	Yes	0
2	SPIRXIF	SPI Receive Interrupt flag, write 0 clear flag	Yes	Yes	0
1	T1IF	Timer1 Interrupt flag, write 0 clear flag	Yes	Yes	0
0	T0IF	Timer0 Interrupt flag, write 0 clear flag	Yes	Yes	0

Address 0EH : Interrupt Mask Register1(INTEN1)

Bits	Name	Description	Read	Write	Default
7	RSMIE	USB resume interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
6	SUSIE	USB suspend interrupt enable bit .(0:disable,1:enable)	Yes	Yes	0
5	RX0IE	Endpoint 0 received successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
4	TX0IE	Endpoint 0 transmit successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
3	TX1IE	Endpoint 1 transmit successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
2	TX2IE	Endpoint 2 transmit successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
1	TX3IE	Endpoint 3 transmit successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
0	RX4IE	Endpoint 4 received successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0

Address 0FH : Interrupt Flag Register1(INTFLAG1)

Bits	Name	Description	Read	Write	Default
7	RSMIF	USB resume interrupt flag, write 0 clear flag	Yes	Yes	0
6	SUSIF	USB suspend interrupt flag, write 0 clear flag	Yes	Yes	0
5	RX0IF	Endpoint 0 received successfully interrupt flag, write 0 clear flag	Yes	Yes	0
4	TX0IF	Endpoint 0 transmit successfully interrupt flag, write 0 clear flag	Yes	Yes	0
3	TX1IF	Endpoint 1 transmit successfully interrupt flag, write 0 clear flag	Yes	Yes	0
2	TX2IF	Endpoint 2 transmit successfully interrupt flag, write 0 clear flag	Yes	Yes	0
1	TX3IF	Endpoint 3 transmit successfully interrupt flag, write 0 clear flag	Yes	Yes	0
0	RX4IF	Endpoint 4 received successfully interrupt flag, write 0 clear flag	Yes	Yes	0

Note : The Endpoint 1,2,3,4 (ACK,NAK) interrupts are base on SELACK (28H Register)

Address 10H: Timer0 overflow reload value(T0RLD)

Bits	Name	Description	Read	Write	Default
7~0	T0RLD	Timer0 overflow reload value	Yes	Yes	00h

Address 11H : USB address(USBADDR)

Bits	Name	Description	Read	Write	Default
7	EUSBAR	Device Address Enable	Yes	Yes	0

		1: Enable Device Address 0: Disable Device Address			
6~0	USBAR	USB device address	Yes	Yes	00h

Address 12H: USB Mode and Endpoint configuration (USBMDCFG)

Bits	Name	Description	Read	Write	Default
7	SUSPMD	F/W force USB interface to go into suspend mode	Yes	Yes	0
6	RESMD	F/W force USB interface send Resume signal in suspend mode	Yes	Yes	0
5	CTRRD	H/W will stall an invalid OUT token during control read transfer	Yes	Yes	0
4	RX0RDY	Endpoint 0 ready for receive, clear by H/ W RX0IF occurs	Yes	Yes	0
3	EP1CFG	Set endpoint 1 configuration	Yes	Yes	0
2	EP2CFG	Set endpoint 2 configuration	Yes	Yes	0
1	EP3CFG	Set endpoint 3 configuration	Yes	Yes	0
0	EP4CFG	Set endpoint 4 configuration	Yes	Yes	0

Address 13H: Endpoint 0 received status (EP0RXST)

Bits	Name	Description	Read	Write	Default
7	RX0TGL	1:Received DATA1(PID) packet ; 0 : Received DATA0 PID Packet	Yes	No	X
6	RX0ERR	EP0 received data error	Yes	No	X
5	EP0DIR	1: IN transfer 0:OUT/SETUP transfer	Yes	No	X
4	EP0SET	SETUP token indicator	Yes	No	X
3~0	RX0CNT	EP0 received data byte count	Yes	No	0h

Address 14H: Endpoint 0 transmit status (EP0TXST)

Bits	Name	Description	Read	Write	Default
7	TX0RDY	EP0 ready for transmit ,clear by H/W while TX0IF occurs	Yes	Yes	0
6	TX0TGL	EP0 transmit DATA1/DATA0 PID packet	Yes	Yes	0
5	EP0STAL	EP0 will stall OUT/IN packet while this bit set 1	Yes	Yes	0
4	Revered	Revered	Yes	No	0
3~0	TX0CNT	EP0 transmit data byte count	Yes	Yes	0h

Address 15H: Endpoint 1 transmit status (EP1TXST)

Bits	Name	Description	Read	Write	Default
7	TX1RDY	EP1 ready for transmit ,clear by H/W while TX1IFoccurs	Yes	Yes	0
6	TX1TGL	When EP1DIR set 0, TX1TGL is EP1 transmit DATA1/DATA0 PID packet. When EP1DIR set 1, TX1TGL is received data toggle bit.(1:DATA1; 0:DATA0)	Yes	Yes	0
5	EP1STAL	EP1 will stall IN packet while this bit set 1	Yes	Yes	0
4	EP1DIR	EP1 direction. 0: IN Transfer 1: OUT Transfer	Yes	Yes	0
3~0	TX1CNT	EP1 transmit/received data byte count	Yes	Yes	0h

Address 16H: Endpoint 2 transmit status (EP2TXST)

Bits	Name	Description	Read	Write	Default
7	TX2RDY	EP2 ready for transmit ,clear by H/W while TX2IF occurs	Yes	Yes	0
6	TX2TGL	When EP2DIR set 0, TX2TGL is EP2 transmit DATA1/DATA0 PID packet. When EP2DIR set 1, TX2TGL is received data toggle bit.(1:DATA1; 0:DATA0)	Yes	Yes	0
5	EP2STAL	EP2 will stall IN packet while this bit set 1	Yes	Yes	0
4	EP2DIR	EP2 direction 0: IN Transfer 1: OUT Transfer	Yes	Yes	0
3~0	TX2CNT	EP2 transmit/received data byte count	Yes	Yes	0h

Address 17H: Endpoint 3 transmit status (EP3TXST)

Bits	Name	Description	Read	Write	Default
7	TX3RDY	EP3 ready for transmit ,clear by H/W while TX3IF occurs	Yes	Yes	0
6	TX3TGL	EP3 transmit DATA1/DATA0 PID packet	Yes	Yes	0
5~0	TX3CN	EP3 transmit data byte count	Yes	Yes	00h

Address 18H: Endpoint 4 received status (EP4RXST)

Bits	Name	Description	Read	Write	Default
7	RX4TGL	1:Received DATA1(PID) packet ; 0 : Received DATA0 PID Packet;	Yes	No	X
6	RX4ERR	EP4 received data error	Yes	No	X
5~0	RX4CNT	EP4 received data byte count	Yes	No	00h

Address 19H : Endpoint status register 1(EPST1)

Bits	Name	Description	Read	Write	Default
7	RX4RDY	EP4 ready for receive, clear by H/W while RX4IF occurs	Yes	Yes	0
6	EP4STAL	EP4 will stall OUT packet while this bit set 1	Yes	Yes	0
5	EP3STAL	EP3 will stall IN packet while this bit set 1	Yes	Yes	0
4	LVDT	0 : VBUS voltage less than 3.8 V 1 : VBUS voltage greater than 3.8V	Yes	No	X
3	LVR	0: When LVDT is high or low, Device still normal work. 1: When LVDT is low , Device will be cleared SUSPEN(12.7H) bit by H/W.	Yes	Yes	0h
2	CLKSW	0 : Don't switch clock . 1: Start to switch from default clock to select clock	Yes	Yes	0h
1:0	SELCLK[1:0]	2'b00 : Select MCU clock is 16MHz 2'b01 : Select MCU clock is 12MHz 2'b10 : Select MCU clock is 6MHz 2'b 11 : Select MCU clock is 58KHz	Yes	Yes	00h

CLKSW	SELCLK[1:0]	MCU Clock
0	2'bXX	6MHz or 12MHz or 24 MHz (base on clock of configuration)
0->1	2'b00	Switch MCU clock is from default clock to 16MHz clock
0->1	2'b01	Switch MCU clock is from default clock to 12MHz
0->1	2'b10	Switch MCU clock is from default clock to 6MHz
0->1	2'b11	Switch MCU clock is from default clock to 58KHz

TABLE 7-5 MCU clock select table

Address 1AH : Timer1(TMR1)

Bits	Name	Description	Read	Write	Default
7~0	TM1	8-bit counter	Yes	Yes	00h

Address 1BH: Timer1 overflow reload value(T1RLD)

Bits	Name	Description	Read	Write	Default
7~0	T1RLD	Timer1 overflow reload value	Yes	Yes	00h

Address 1CH: SPI Receive Buffer Register (SPIRXB)

Bits	Name	Description	Read	Write	Default
7~0	RX	Once the 8-bits data have been received, the data in SPI shift register (SPISR) will be moved to the SPIRXB register. The data must be read out before the next 8-bits data reception is completed	Yes	Yes	XXh

		if needed. The RXBF flag is set when the data in SPISR is moved to the SPIRXB register, and cleared as the SPIRXB register reads.			

Address 1DH: SPI transmit Buffer Register (SPITXB)

Bits	Name	Description	Read	Write	Default
7~0	TX	Once the first valid clock pulse appear on SCK pin, the data in SPITXB will be loaded into SPISR and start to shift in/out. The new data must be written to SPITXB before the 8-bits data transmission is completed if needed. The TXBF flag is set when the data in SPITXB is moved to the SPISR register, and cleared as the SPITXB register writes.	Yes	Yes	XXh

Address 1EH: SPI Status Register (SPISTAT)

Bits	Name	Description	Read	Write	Default
7~6	Revered	Revered	Yes	No	0
5	TXBF	SPI transmit buffer empty flag.	Yes	Yes	0
4	TM1IF	SPI receive complete interrupt flag in Timer 1 mode. Set when receiving complete, reset by software.	Yes	Yes	0
3	SDOOD	1:Open-drain 0: Normal-drive control bit for SDO pin output	Yes	Yes	0
2	SCKOD	1: Open-drain 0: Normal-drive control bit for SCK pin output	Yes	Yes	0
1	Revered	Revered	Yes	No	0
0	RXBF	SPI receive buffer full flag. Set when the data in SPISR is moved to the SPIRXB register, reset by software or reading SPIRXB register	Yes	Yes	0

Address 1FH: SPI Control Register (SPICON)

Bits	Name	Description	Read	Write	Default
7	CKEDG	Clock edge select bit	Yes	Yes	0
6	SPION	SPI module enable bit. (0:disable,1:enable)	Yes	Yes	0
5	RXOV	SPI receive buffer overflow bit (only in slave mode)	Yes	Yes	0
4	SSE	SPI shift register enable bit.(0:disable,1:enable) 0:Reset by hardware as soon as the shifting is complete 1:Start to transmit/receive, and keep on “1” while the current byte is still begin transmitted/received. Note: When SSEMOD is set, the SSE is a “don't care”.	Yes	Yes	0
3	SSEMOD	SSE bit control enable bit.(0:disable,1:enable) 0:Enable the SSE bit control. It means the SCK input/output will be inhibited of SSE=0 1:Disable the SSE bit control. It means the SCK input/output direct.	Yes	Yes	0
2~0	SPIM	SPI mode setting	Yes	Yes	0

SPIM[2:0]	SSP MODE
3'b000	SPI master mode, clock(SCK) = F_{cpuclk} /2
3'b001	SPI master mode, clock (SCK)= F_{cpuclk} /4
3'b010	SPI master mode, clock(SCK) = F_{cpuclk} /8
3'b011	SPI master mode, clock(SCK) = F_{cpuclk} /16
3'b100	SPI master mode, clock(SCK) = F_{cpuclk} /32
3'b101	SPI slave mode, clock =SCK pin, SSB pin control enabled
3'b110	SPI slave mode, clock =SCK pin, SSB pin control disabled
3'b111	SPI master mode, clock(SCK) = (Timer1 output)/2

TABLE 7-5 SPI Mode

Address 20H: PS2/USB Detect Mode (PUMODE)

Bits	Name	Description	Read	Write	Default
7	USBMD	USB mode select. 1 : Full_Speed mode 0:Low_speed mode	Yes	Yes	1
6~5	ROMBNK[1:0]	ROM bank select bits 2'b00 : Bank-0 ; 2'b01:Bank-1;2'b10:Bank-2:2'b11:Reversed	Yes	Yes	0h
4	Revered	Revered	Yes	No	0
3~2	RAMBNK[1:0]	RAM bank select bits 2'b00 : Bank-0 ; 2'b01:Bank-1;2'b10:Bank-2;2'b11:Bank-3	Yes	Yes	0h
1	PS2EN	Enable PS2 D+ ,D- pull-up 4.7kΩ resistors	Yes	Yes	0
0	USBEN	Enable USB D+ or D- pull-up 1.5KΩ resistor	Yes	Yes	0

{ PS2EN , USBEN } = 2'b 00: Detect Mode ; 2'b01: Set USB Mode ; 2'b10: Set PS2 Mode ; 2'b11: USB Test

1. After Power-on reset, The PUMODE[1:0] default value is 2'b 00. The device have 200KΩ pull-up resistors on D+/D- line.
2. If D+/CLK(portc[1]), D-/Data(portc[0]) status are 2'b00, microprocessor set PUMODE[0] = 1. Otherwise microprocessor set PUMODE[1] = 1.
3. When Device is defined USB mode , the D- or D+ line will be enable 1.5KΩ Pull-up resistor to 3.3V ,and disable 200KΩ pull-up Resistor.
4. When Device is defined PS/2 mode , the D- and D+ will be enable 4.7KΩ pull-up resistor to Vcc(5V).

Address 21H : Port D (PORTD)

Bits	Name	Description	Read	Write	Default
7~0	PORTD	Port D data input/output	Yes	Yes	00h

Address 22H : Port E (PORTE)

Bits	Name	Description	Read	Write	Default
7~0	PORTE	Port E data input/output	Yes	Yes	00h

Address 23H : CLKCFG

Bits	Name	Description	Read	Write	Default
7	SELACK	SELACK : 1: When handshaking is ACK packet, the S/W generates interrupt for ENP1,2,3,4. 0: When handshaking is any packet, he S/W generates interrupt for ENP1,2,3,4.	Yes	Yes	1'b1
6~5	DPMF[1:0]	DP,DM forcing bit. DPMF[1:0] DM, DP state 2'b11 Normal Drive SIE control 2'b10 DM Drive High, DP Drive Low 'K' State 2'b01 DM Drive Low, DP Drive High 'J' State 2'b00 DM Drive Low, DP drive Low SE0	Yes	Yes	2'b11
4~2	SELINT[2:0]	Select which port has interrupt function 3'b000: PORTB interrupt 3'b001: PORTA interrupt 3'b010: PORTC interrupt 3'b011: PORTD interrupt 3'b100: PORTE interrupt	Yes	Yes	3'b000
1	INCODS	Internal clock output disable. 1:Disable internal IRC clock output. XOUT pin will drive high.	Yes	Yes	0

Bits	Name	Description	Read	Write	Default
		0:Enable internal IRC clock output . The IRC clock is driven output to the XOUT pin.			
0	IRCEN	Internal IRC clock enable 1:Enable internal IRC clock. 0:Disable internal IRC clock.	Yes	Yes	1

Address 2B~2FH : General purpose Register

Bits	Name	Description	Read	Write	Default
7~0	Register	General purpose Register	Yes	Yes	00h

Address 30H~37H EP0 Received Buffer(EP0RX) **Rambnk[1:0] = 00 or 01**

Address	Description	Read	Write	Default
30H	Start address of EP0RX	Yes	Yes	XXh
30H~37H	Buffer length is 8 bytes for EP0RX	Yes	Yes	XXh

Address 38H~3FH EP0 Transmitter Buffer(EP0TX) **Rambnk[1:0] = 00 or 01**

Address	Description	Read	Write	Default
38H	Start address of EP0TX	Yes	Yes	XXh
38H~3FH	Buffer length is 8 bytes for EP0TX	Yes	Yes	XXh

Address 40H~47H EP1 Transmitter Buffer(EP1TX) **Rambnk[1:0] = 00 or 01**

Address	Description	Read	Write	Default
40H	Start address of EP1TX	Yes	Yes	XXh
40H~47H	Buffer length is 8 bytes for EP1TX	Yes	Yes	XXh

Address 48H~4FH EP2 Transmitter Buffer(EP2TX) **Rambnk[1:0] = 00 or 01**

Address	Description	Read	Write	Default
48H	Start address of EP2TX	Yes	Yes	XXh
48H~4FH	Buffer length is 8 bytes for EP2TX	Yes	Yes	XXh

Address 30H~3FH EP3 Transmitter Buffer0 (EP3TX0) **Rambnk[1:0] = 10**

Address	Description	Read	Write	Default
30H	Start address of EP3TX0	Yes	Yes	XXh
30H~3FH	Buffer length is 16 Bytes	Yes	Yes	XXh

Address 40H~4FH EP3 Transmitter Buffer1 (EP3TX1) **Rambnk[1:0] = 10**

Address	Description	Read	Write	Default
40H	Start address of EP3TX1	Yes	Yes	XXh
40H~4FH	Buffer length is 16 Bytes	Yes	Yes	XXh

Address 30H~3FH EP4 Received Buffer0 (EP4RX0) **Rambnk[1:0] = 11**

Address	Description	Read	Write	Default
30H	Start address of EP4RX0	Yes	Yes	XXh
30H~3FH	Buffer length is 16 Bytes	Yes	Yes	XXh

Address 40H~4FH EP4 Received buffer1 (EP4RX1) **Rambnk[1:0] = 11**

Address	Description	Read	Write	Default
40H	Start address of EP4RX1	Yes	Yes	XXh
40H~4FH	Buffer length is 16 Bytes	Yes	Yes	XXh

8.0 Instruction Set

Mnemonic, Operands	Description	Operation	Cycles	Status Affected
BCR R, bit	Clear bit in R	0 → R<b>	1	-
BSR R, bit	Set bit in R	1 → R<b>	1	-
BTRSC R, bit	Test bit in R, Skip if Clear	Skip if R<b> = 0	1/2 (1)	-
BTRSS R, bit	Test bit in R, Skip if Set	Skip if R<b> = 1	1/2 (1)	-
NOP	No Operation	No operation	1	-
CLRWDT	Clear Watchdog Timer	00h → WDT, 00h → WDT prescaler	1	$\overline{TO}$, $\overline{PD}$
SLEEP	Go into power-down mode	00h → WDT, 00h → WDT prescaler	1	$\overline{TO}$, $\overline{PD}$
RETURN	Return from subroutine	Top of Stack → PC	2	-
RETFIE	Return from interrupt, set GIE bit	Top of Stack → PC, 1 → GIE	2	-
CLRA	Clear ACC	00h → ACC	1	Z
IOST R	Load IOST register	ACC → IOST register	1	-
IOSTR R	Read IOST register	IOST register → ACC	1	-
CLRR R	Clear R	00h → R	1	Z
MOVAR R	Move ACC to R	ACC → R	1	-
MOVR R, d	Move R	R → dest	1	Z
DECR R, d	Decrement R	R - 1 → dest	1	Z
DECRSZ R, d	Decrement R, Skip if 0	R - 1 → dest, Skip if result = 0	1/2 (1)	-
INCR R, d	Increment R	R + 1 → dest	1	Z
INCRSZ R, d	Increment R, Skip if 0	R + 1 → dest, Skip if result = 0	1/2 (1)	-
ADDAR R, d	Add ACC and R	R + ACC → dest	1	C, DC, Z
SUBAR R, d	Subtract ACC from R	R - ACC → dest	1	C, DC, Z
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 f through Carry	R<7> → C, R<6:0> → dest<7:1>, C → dest<0>	1	C
RRR R, d	Rotate right f 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,	2	-

Mnemonic, Operands	Description	Operation	Cycles	Status Affected
		Top of Stack → PC		
CALL I	Call subroutine	PC + 1 → Top of Stack, I → PC	2	-
GOTO I	Unconditional branch	I → PC	2	-

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

2. bit : Bit address within an 8-bit register R

R : Register address (00h to 7Fh)

I : Immediate data

ACC : Accumulator

d : Destination select;

=0 (store result in ACC)

=1 (store result in file register R)

dest : Destination

PC : Program Counter

PCHBUF : High Byte Buffer of Program Counter

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 : Digital carry bit

Z : Zero bit

9.0 Clock Control

The chip can be generated clock from either IRC or from external crystal fed into PLL . The PLL clock can be programming 29h(MPLL), 2Ah(NPLL) register base on the external crystal frequency. For example code and steps as follows:

(1) Power on, chip begins operation using the internal clock.

(2) The MCU clock switched to select clockare as follow : (MCU is 24Mhz)

```
BCR  EPST1,CLKSW_B
BCR  EPST1,SELCLK0_B
BCR  EPST1,SELCLK1_B
BSR  EPST1,CLKSW_B
// MCU  clock switch to 16Mhz
BCR  EPST1,CLKSW_B
BSR  EPST1,SELCLK0_B
BCR  EPST1,SELCLK1_B
BSR  EPST1,CLKSW_B
// MCU  clock switch to 12Mhz
```

(3) After an addition delay the MCU is released to run. This delay depend on bit 3 of the CLKCFG(28h) register . The time is 128us if this bit is 0,or 4ms if this bit set 1. The timing(128us or 4ms) is internal clock switched to external clock.

10.0 Reset

There are three cases of reset on USB controller chip. These cases of reset occurrence are listed below.

1. Power On Reset Negative (PORN).
2. Watchdog Reset (WDR)
3. External Reset (EXR).

If the PORN,WDR, or EXR occurs, the chip will enter reset status. The following events take place on reset status.

(1) All registers are reset to their default values expect status registers.
(2) The status register (03h) is reset to their default value only for PORN .
(3) After reset status, program counter begins at address 0x0000.

PORN is asserted when VBUS(Vcc) voltage to the device is upper approximately 3.8V(see *Figure* 10-1).

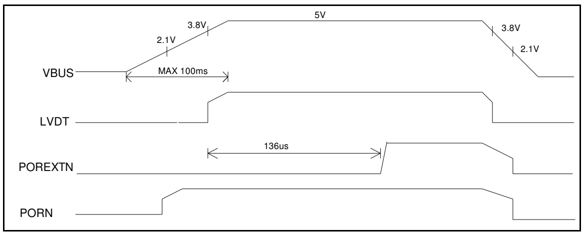


Figure 10-1 power on reset timing

11.0 Suspend Mode

The FM8PU86 chip has detection of resume and remote wake-up capability in suspend mode.

Normal code for entering suspend is shown below:

```
…                     Enable which PORTB wake-up
                      interrupt if desired for wake-up
bcr     12h,6         Clear resume bit
bsr     12h,7         Set suspend bit
sleep                 Chip go to sleep mode
```

More details on the various resume in suspend mode are given in the following sections.

11.1 Detection of Resume

The SIE uses the LineState(DP,DM) signals to determined when the USB transition from the 'J' to 'K' state in FS mode or from 'K' to 'J' state in LS mode. When a device is suspended, a 'J'(FS) or 'K'(LS) state is on bus and SIE should be looking for whether a 'K'(FS) or 'J'(LS) state be forced on bus by HOST . If event is happened, the device is resumed by HOST.

11.2 Remote Wake-up

A device with remote wake-up capability must set PBIE register (09h) before into suspend mode. When PORTB[7:0] or PORTA[7:0] are any transition base on PBCON(0Ah) set in suspend mode, the device will start to remote wake-up function. And then, the device need set resume (12h,6) bit to send 'K' state (if FS mode) or 'J' state (if LS mode) remote signal. The remote signaling must be asserted (FS 'K' or LS 'J') at least 1ms for USB specification. The device don't clear resume bit until delay time at least 1ms or enter next suspend mode.

(1)Remote Wake-up on PB/PA/PC/PD/PE, interrupt

```
PBITASK :
        bcr   USBMDCFG,SUSPMD_B
        bcr   INTFLAG,PBIF_B
//Device  remove wake up host to sent J or K state
        bsr   USBMDCFG,RESMD_B
        call  delay_1ms
        bcr   USBMDCFG,RESMD_B
```

(2).Remote Wake-up on Watch-dog timeout

(a) WakeUp for Full_speed :

```
        bcr   CLKCFG,DPMF[0]
        call  delay_1ms
        bsr   CLKCFG,DPMF[0]
//Device  remove wake up host to sent  K state
```

(b) WakeUp for Low_speed:

```
        bcr   CLKCFG,DPMF[1]
        call  delay_1ms
        bsr   CLKCFG,DPMF[1]
//Device  remove wake up host to sent  J state
```

Note : (1) "J" state is DP= high(3.3V),DN=Low(0V);
(2) "K" state is DP=Low(0V),DN=high(3.3V); on USB bus.

12.0 General Purpose I/O ports

Ports A are 8 bits I/O register. Ports B are 8 bits I/O register ,each bit can also selected as an external interrupt source for the microcontroller.

Figure 12-1 shows a diagram of a GPIO port pin.

Refer Table7-2.1 and Table 7-2.2 , each pin can be independently configured as high-impedance inputs, inputs with internal pull-ups, open drain outputs, or traditional output s with selectable drive strength.

When SPION(1fh,6) bit is set, the PA.4,PA.5,PA.6 and PA.7 pins are accessed by SPI hardware of chip.

After reset, all GPIO data and controlled mode register is cleared, so the GPIO pins are as input mode.

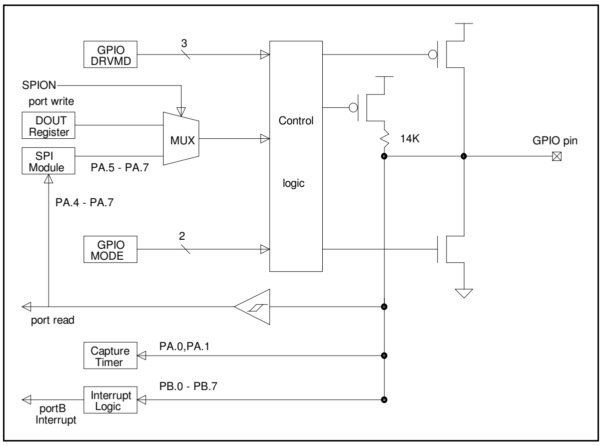


Figure 12-1 Block Diagram of GPIO port

13.0 USB and PS2 Mode Detection

The FM8PU86 are integrated USB and PS2 device on chip. When USB mode function is be enabled(set USBEN bit of PUMODE),the DP,DM can be read/write by SIE. The USB DP and DM pins can be used for PS2 SCLK and SDATA pins, respectively. When PS2 mode function is be enabled(set PS2EN bit of PUMODE), the SCLK and SDATA can be read/write at bits[1:0] of PORTC. The PS2 on chip support circuit is show ***Figure*13-1.**

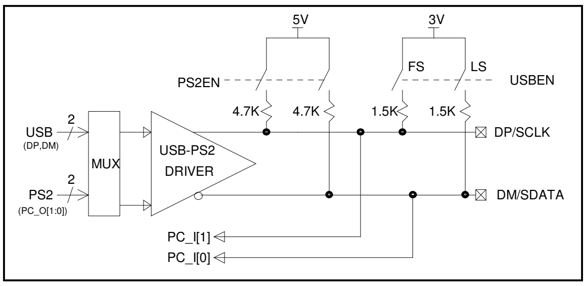


*Figure*13-1 Block diagram of USB/PS2 connections

After power on reset(VBUS asserted < 100ms),user firmware need to detect PORTC[1:0] bits. For example code as follows:

```
movr   PORTC,A
andia  03h              // Detect portc[1:0]
btrsc  STATUS,Z_B
goto   SETUSB           // USB mode
goto   SETPS2           // PS2 mode
```

14.0 USB Transceiver

This block handles the USB signaling. This include features such as below.

1. Complies with USB Rev2.0
2. Support 12Mbits/s "Full Speed" and 1.5Mbits/s "Low speed".
3. Data and clock recovery from serial stream on the USB.
4. Build in pull-up 1.5k resistor.
5. Build in power-on reset.

14.1 USB Regulator output

The VREG pin provide a regulated output for internal USB transceiver and external supply power 3.3V chip.

15.0 USB Serial Interface Engine(SIE)

The SIE will handle the USB packets from/to transceiver and communication with the USB Host.

Some key features of the USB SIE are :

1. For incoming packets, check CRC; For outgoing packets add CRC bytes.
2. Issue USB interrupt to the micro-controller only if IN/OUT token packets are successful transaction.
3. Automatic returned appropriate ACK/NAK/STALL handshake packets.
4. Automatic update the toggle bit (1/0) of data packet .
5. Handle the USB reset protocol; suspend condition detection; detection of resume; Handshake detection protocol.
6. SYNC/EOP generation and check.
7. Bit-stuffing/unstuffing.
8. Serializer/Deserializer.

15.1 Default Control Endpoint : EP0

EP0 is the control transfer endpoint where the transmission is defined as the direction from device to host and receiving is defined as the direction from host to device. The EP0 buffers will be mapped into the shared SRAM buffer rather than dedicated FIFO.

15.2 Interrupt Transfer Endpoints : EP1 and EP2

The endpoint 1,2 is used for interrupt transmission(INT IN) .The maximum allowable interrupt data payload size 8 bytes for full-speed or low-speed transaction. The micro-controller needs to program the maximun packet size registers (15h-TX1CNT,16h-TX2CNT) properly to avoid any size errors. The interrupt transfer type is designed to support those devices that need to sent data infrequently but with bounded service period.

15.3 Bulk Transfer Endpoints:EP3 and EP4

The endpoint 3 is used for bulk transmission (BULKI IN) and endpoint 4 is used for bulk receiving (BULK OUT). The maximum allowable bulk data payload size is 32 bytes for full-speed or low-speed transaction. The micro-controller needs to program the maximum packet size resisters (17h-TX3CNT,18h-RX4CNT) properly to avoid any size errors. The bulk transfer type is designed to support devices that need to communicate relatively large amounts of data at highly variable times where the transfer can use any available bandwidth. The EP3 or EP4 buffer is stand-alone SRAM(32 bytes) that can be as general purpose memory.

16.0 Micro-Controller

16.1 Timer0/Timer1

The Timer0/1 is a 8 bit clock counter with a programmable prescaler and a 8-bit overflow reload(T0RLD/T1RLD).The clock source of Timer0/1 comes from the internal clock(F_{MCUCLK}/4). The option of Timer0/1 precaler is defined by T0/1PS2,1,0(0Bh register) see Table 16-1.

T0/1 SP2, SP1, SP0	Prescaler Rate
3'b000	1*(4*clock)
3'b001	2*(4*clock)
3'b010	4 *(4*clock)
3'b011	8 * (4*clock)
3'b100	16 * (4*clock)
3'b101	32 *(4*clock)
3'b110	64 * (4*clock)
3'b111	128 * (4*clock)

Table 16-1 Timer0/1 Prescaler Rate

Timer1 also can be as a baud rate clock generator for the SPI module.

The Timer0/1 increment from 00h until it equals the T0RLD/T1RLD value. The timer interrupt flag(T0IF/T1IF) is asserted when the Timer0/1 rollover to 00h. The Timer0/1 also has corresponding interrupt enable bit(T0IE/T1IE). The Timer0/1 interrupt can be enabled/disabled by setting/clearing these bits. The Timer0/1 can be turned on and off by software control. When the Timer0/1 on control bit(0Bh,T0ON,T1ON) is set, the Timer0/1increments from the clock source. When T0ON/T1ON is cleared, the Timer0/1 is turned off and cannot cause the Timer0/1 interrupt flag to be set. The Timer0/1 block diagram is show ***Figure*16-1.**

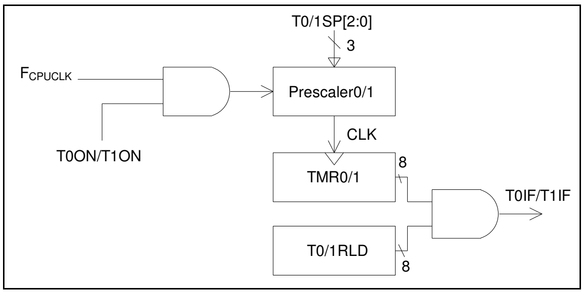


*Figure*16-1 Block diagram of Timer0/1

The Timer0/1 is calculated as follows:

MCU clock F_{cpuclk}= 6MHz

```
MOVIA    106D
MOVAR    T0RLD
MOVIA    0bH        (T0PS[2:0] = 011)
MOVAR    TMCON
```

T0IF = (106+1)*32*clock=570.667us

16.2 Watchdog Timer (WDT)

The Watchdog Timer (WDT) is a free running on chip RC oscillator which dose not require any external components. So the WDT will still run into SLEEP mode. During normal operation or in SLEEP mode, a WDT time-out will cause the device reset .

The CLRWDT instruction clears the WDT , if assigned to the WDT, and prevents it from timing out and generating a device reset.

The WDT can be disabled by clearing the control bit WDTE(08h-7). The WDT has a normal time-out period of 18ms(without precaler). When the SLEEP instruction executes , the WDT and the prescaler will be reset.

The prescaler of WDT refers Table7-4.

The block diagram of WDT shows in *Figure* 16-2.

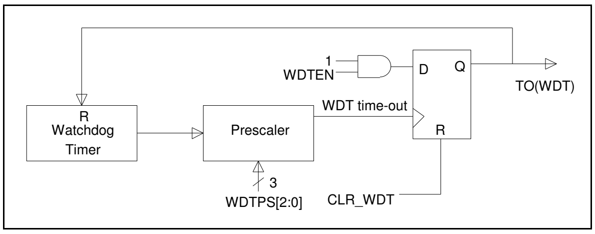


Figure 16-2 Block diagram of WDT

17.0 Serial Peripheral Interface (SPI)

The Serial Port Interface (SPI) module is a serial interface useful communication with other peripheral or micro-controller device.

The SPI module allows 8-bit of data to be synchronously transmitted and received simultaneously. To accomplish communication, typically three pins are used :

1. Serial Clock(SCK)
2. Serial Data IN (SDI)
3. Serial Data out (SDO)
4. Slave select(SSB)

The block diagram of SPI is show in *Figure* 17-1.

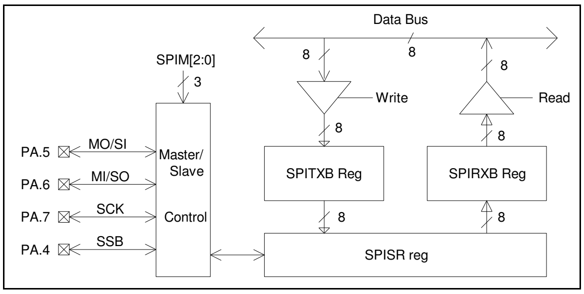


Figure 17-1 Block diagram of SPI

17.1 Master Mode

17.1.1 Master Mode with SSE Control(SSEMOD =0)

How to transmit/Receive data in this master mode , descript as below:

1. Enable SPI function by setting the SPION (1Fh-6) bit.
2. Decide the transmission rate and source by programming SPIM[2:0] bits. (Refer Table 7-5)
3. Write the data to SPITXB for transmitting if needed.
4. Set SSE (1Fh-4) bit to start transmit.
5. When the 8-bit data transmission starts, both of the SPITXIF and TXBFIF interrupt flags will set to 1. Beside, both of these are cleared by software. The TXBF flag also will be set to 1,cleared by software or by writing data to SPITXB register.
6. Write next byte data to SPITXB register before this byte transmission being finished if needed.
7. When the 8-bit data transmission is over, the SSE bit will be reset to "0" by hardware. Therefore, if user want to transmit/receive another 8-bit data user must set SSE bit to "1" again.
8. When the 8-bit data transmission is completed, the SPIRXIF interrupt flag will set to 1. Besides, the SPIRXIF is cleared by software. The RXBF flag also will be set to "1", cleared by software or by reading out SPRRXB register.
9. Read out the SPIRXB register before next byte transmission begin finished if needed.

The SPI mode timing is show in ***Figure* 17-1** for Master mode.

17.1.2 Master Mode without SSE Control(SSEMOD =1)

How to transmit/Receive data in this master mode , descript as below:

1. Enable SPI function by setting the SPION (1Fh-6) bit.
2. Decide the transmission rate and source by programming SPIM[2:0] bits.(Refer Table 7-5)
3. Write the data to SPITXB for transmitting if needed.
4. When the 8-bit data transmission starts, both of the SPITXIF and TXBFIF interrupt flags will set to 1. Beside, both of these are cleared by softwafe. The TXBF flag also will be set to 1,cleared by software or by writing data to SPITXB register.
5. Write next byte data to SPITXB register before this byte transmission being finished if next byte transmission is needed.
6. When the 8-bit data transmission is completed, the SPIRXIF interrupt flag will set to 1. Besides, the SPIRXIF is cleared by software. The RXBF flag also will be set to "1", cleared by software or by reading out SPRRXB register.
7. Read out the SPIRXB register before next byte transmission begin finished if needed.

17.2 Slave Mode

17.2.1 Slave Mode with SSE Control(SSEMOD =0)

How to transmit/Receive data in this master mode , descript as below:

Enable SPI function by setting the SPION (1Fh-6) bit.

1. Enable/Disable the SSB pin control by programming SPIM[2:0] bits. (Refer Table 7-5)
2. Write the data to SPITXB for transmitting if needed.
3. Set SSE (1Fh-4) bit and wait the external clock pulses appear on SCK pin to start transmit.
4. When the 8-bit data transmission starts, both of the SPITXIF and TXBF interrupt flags will set to 1. Beside, both of these are cleared by software. The TXBF flag also will be set to 1,cleared by software or by writing data to SPITXB register.
5. Write next new byte data to SPITXB register before this byte transmission being finished if next byte transmission is needed.
6. When the 8-bit data transmission is over, the SSE bit will be reset to "0" by hardware. Therefore, if user want to transmit/receive another 8-bit data user must set SSE bit to "1", again before next clock pulse appearing SCK pin.
7. When the 8-bit data transmission completed, both of the SPIRXIF and RCBF interrupt flags will set to "1". Besides, both of these bits are cleared by software. The RXBF flag also will be set to "1",

cleared by software or by reading out SPIRXB register.

8. Read out the SPIRXB register before next byte transmission begin finished if needed.

The SPI mode timing is show in ***Figure* 17-2** for Salve mode.

17.2.2 Slaver Mode with SSE Control(SSEMOD =1)

1. Enable SPI function by setting the SPION (1Fh-6) bit.
2. Enable/Disable the SSB pin control by programming SPIM[2:0] bits. (Refer Table 7-5)
3. Write the data to SPITXB for transmitting if needed.
4. Wait the external clock pulses appear on SCK pin to start transmit.
5. When the 8-bit data transmission starts, both of the SPITXIF and TXBFIF interrupt flags will set to 1. Beside, both of these are cleared by software. The TXBF flag also will be set to 1,cleared by software or by writing data to SPITXB register.
6. Write next new byte data to SPITXB register before this byte transmission being finished if next byte transmission is needed
7. When the 8-bit transmission is completed, both the SPIRXIF and RXBF interrupt flags will be set to 1. Besides, both of these bits are cleared by software. The RXBF flag also will be set to 1, cleared by software or by reading out SPITXB register.
8. Read out the SPIRXCB register before next byte transmission begin finished if needed.

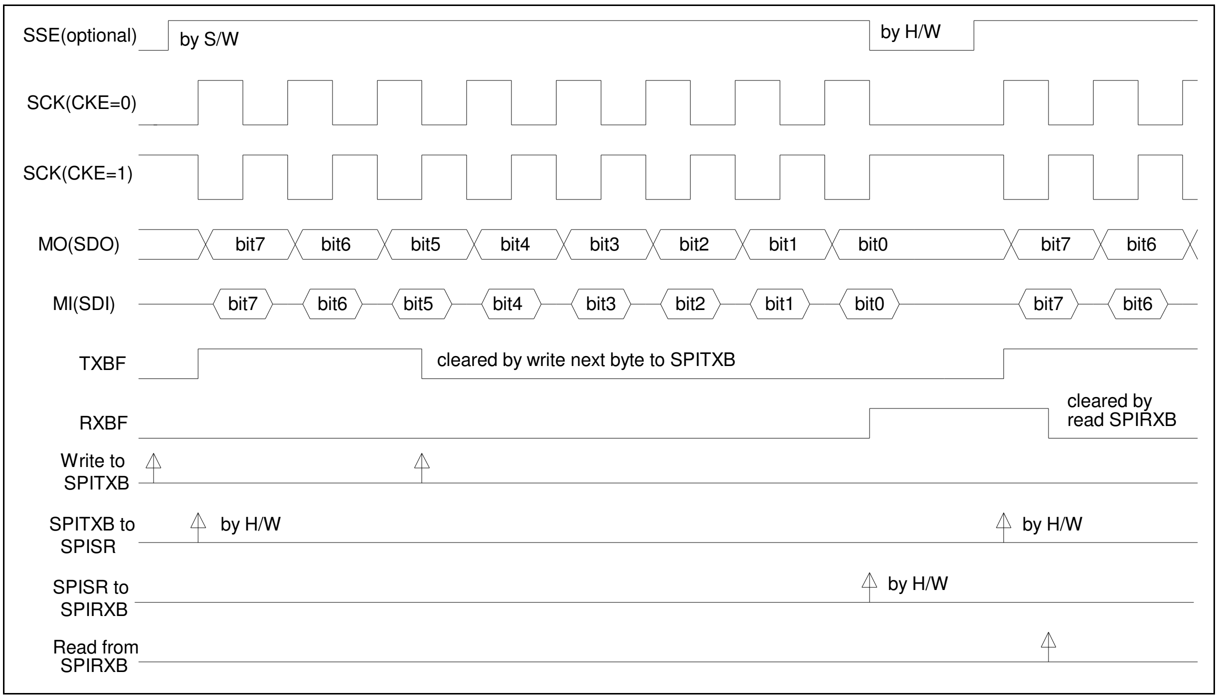


Figure 17-2 SPI Mode Timing (Master Mode)

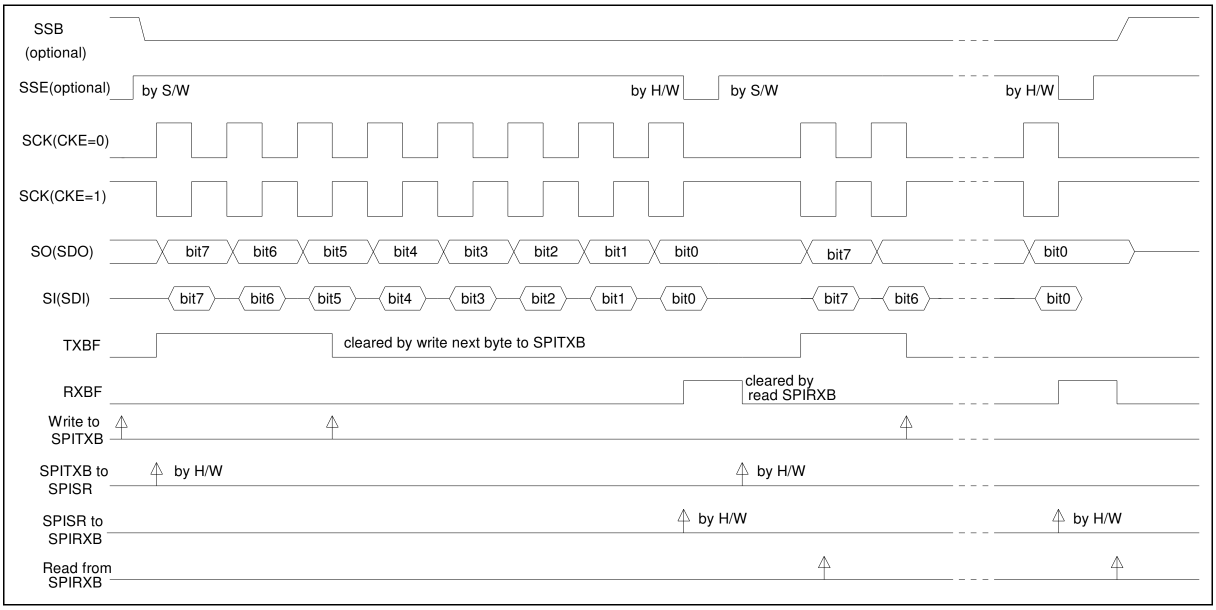


Figure 17-2 The SPI mode Timing (Slave Mode, with SSB control enable/disable)

18. SPI communication code

The following code can be used to implement the SPI data communication:

```
bcr   SPICON,SPIM0_B          // SPIM[2:0]=3'b011 , set SPI as master mode and clock=Fcpu/16
bsr   SPICON,SPIM1_B
bsr   SPICON,SPIM2_B
bsr   SPICON,SPION_B          //Enable SPI mode
bsr   SPICON,CKEDG_B          // SPI clock falling edge sent Data and raising edge sample data
```

Subroutine for SPI transmitted code

SPI_TX :

```
            bcr     SPISTAT,SDOOD_B   //  MO(PA.5) normal Drive
            movr    ADDRESS,A         //  Will sent data(address) to save ACC
            movar   SPITXB            //  Acc data move to SPITXB buffer
            bsr     SPICON,SSE_B      //  Asserted SSE  (start to transmitted)
WAITSSEN :  btrsc   SPICON,SSE_B      //  Wait transmitted to end
            goto    WAITSSEN
            return
```

Subroutine for SPI received code

SPI_RX:

```
            bsr     SPISTAT,SDOOD_B   //  SDO open drain
            movia   ffh               // When MO(PA.5) are connected to MI(PA.6), the SDO
                                      //  need pull-high in received .
            movar   SPITXB
            bcr     SPISTAT,RXBF_B
            bsr     SPICON,SSE_B      // Asserted SSE  (start to received)
WAITRXB:    btrsc   SPISTATSSE_B      // wait received interrupt
            goto    WAITRXB
            bcr     SPISTAT,RXBF_B    // Clear RXBF bit
            return
```

19. Interrupt

The FM8PU86LK has interrupt as following:

1. Timer 0 match interrupt.
2. Timer 1 match interrupt.
3. SPI transmit module interrupt.
4. SPI receive module interrupt.
5. GPIO(PA/PB/PC/PD/PE) external interrupt.
6. USB bus reset interrupt.
7. USB resume interrupt.
8. USB suspend interrupt.
9. USB endpoint0,1,2,3 and 4 interrupt.
10. USB SOF interrupt.

The FM8PU86LK reset vector is fixed at 0x0000h and
the interrupt vector is at 0x0003h.
A global interrupt enable bit, GIE(Reg.0Ch-7), enable(if set) all un-masked interrupts or disables(if cleared) all interrupts. Individual interrupts can be enable/disable through their corresponding enable bits in INTEN register regardless of the status of the GIE bit.
The interrupt priority is decided by customer firmware control.

20. Absolute Maximum Ratings

Parameter	Conditions	Values		Unit
		min.	max.	
Ambient Operating Temperature	-	-10	85	℃
Storage Temperature	-	-10	150	℃
DC Supply Voltage	-	2.4	5.5	V
Supply Current	-	-	-	mA

21. DC Characteristics (Operating Temperature = 0 to70 ℃)

21.1 General

Symbol	Parameter	Conditions	Values		Unit
			min.	max.	
V_{CC}	Operating Voltage	-	4.5	5.5	V
I_{CC1}	Operating Voltage Current Typical = mA[1]	Vcc=5.5V,No GPIO loading IRC Operating ,MCU=12MHz	-	13	mA
I_{CC2}	Operating Voltage Current Typical = mA[1]	Vcc=5.5V,No GPIO loading IRC Operating,MCU=24MHz	-	17	mA
I_{CC3}	Operating Voltage Current(EXOSC) Typical = mA[1] extern crystal =12MHz	Vcc=5.5V,No GPIO loading PLL Operating,MCU=12MHz	-	14	mA
I_{CC4}	Operating Voltage Current(EXOSC) Typical = [1] extern crystal =12MHz	Vcc=5.5V,No GPIO loading PLL Operating,MCU=24MHz	-	18	mA
I_{CC5}	Operating Voltage Current (Disable USB module)	Vcc=5V,No GPIO loading IRC Operating ,MCU=24MHz		12	mA
I_{CC6}	Operating Voltage Current (Disable USB module)	Vcc=5V,No GPIO loading IRC Operating ,MCU=12MHz		8	mA
I_{CC7}	Operating Voltage Current (Disable USB module)	Vcc=3.3V,No GPIO loading IRC Operating ,MCU=24MHz		6	mA
I_{CC8}	Operating Voltage Current (Disable USB module)	Vcc=3.3V,No GPIO loading IRC Operating ,MCU=12MHz		4	mA
I_{SPD}	Suspend Current(enable or disable USB)	With IRC(17us)	-	100	uA
V_{POR}	Power on Reset		2.0	-	V
V_{LVDT}	Low voltage detect		3.8		V
T_{VBUSST}	Vbus Power on Slew Time	Linear ramp:0 to 4V		100	ms
V_{REG}	VREG Regulator output	-	3.0	3.6	V
I_{REG}	VREG supply current			45	mA

21.2 GPIO Interface

Symbol	Parameter	Conditions	Values		Unit
			min.	max.	
RUP	Pull-up Resistor	-	11.6	13.7	KΩ
V_{OL_HD5v}	Output Low Voltage(High Sink)	OVDD=5V,I_{OL}=25mA	-	0.8	V
V_{OL_HD3v}	Output Low Voltage(High Sink)	OVDD=3.3V, I_{OL}=20mA		0.8	V
V_{OL_MD5v}	Output Low Voltage(Medium Sink)	OVDD=5V,I_{OL}=7mA	-	0.4	V
V_{OL_MD3v}	Output Low Voltage(Medium Sink)	OVDD=3.3V,I_{OL}=5mA	-	0.4	V
V_{OL_LD5v}	Output Low Voltage(Low Sink)	OVDD=5V,I_{OL}=2mA	-	0.4	V
V_{OL_LD3v}	Output Low Voltage(Low Sink)	OVDD=3.3VI_{OL}=1.5mA	-	0.4	V
V_{OH}	Output High Voltage	OVDD =5V,I_{OH}=2mA	OVDD-2	-	V
V_{OH}	Output High Voltage	I_{OH}=2mA	OVDD-1	-	V
V_{IL}	Input Threshold Voltage	High to Low edge, Port A,B	0.4*OVDD	0.6*OVDD	V
V_{IH}	Input Threshold Voltage	Low to High edge,PortA,B	0.4*OVDD	0.6*OVDD	V

Note:
[1]:Bench measurements on nominal operating conditions. V_{cc} = Vbus

22. AC Characteristics

22.1 Clock Period

Symbol	Parameter	Conditions	Values		Unit
			min.	max.	
$F_{IRC}1$	Internal Clock Frequency	Internal Clock Enable;	24-0.2%	24+0.2%	MHz

22.2 USB Timing

Symbol	Parameter	Conditions	Values		Unit
			min.	max.	
T_R	Transition Rise Time	C_{load}=200pF(10/%~90%)	-	-	ns
T_R	Transition Rise Time	C_{load}=600pF(10/%~90%)	-	-	ns
T_F	Transition Fall Time	C_{load}=200pF10/%~90%)	-	-	ns
T_F	Transition Fall Time	C_{load}=600pF(10/%~90%)	-	-	ns
V_{CRS}	Output Signals Crossover	C_{load}=200 to 600 pF	1.3	2.0	V

Note: See *Figure* 21-2

22.3 SPI Timing

Symbol	Parameter	Conditions	Values		Unit
			min.	max.	
T_{SMCK}	SPI Master Clock Rate	-	-	12	Mhz
T_{SSCK}	SPI Slave Clock Rate	-	-	12	Mhz
T_{SCKH}	SPI Clock High Time	-	41.65	-	ns
T_{SCKL}	SPI Clock Low Time	-	41.65	-	ns
T_{MDY}	Master Data out Delay Time	-	5	-	ns
T_{MSU}	Mater Input Data Set-up Time	-	10	-	ns
T_{MHD}	Master Input Data Hold Time	-	2	-	ns
T_{SDY}	Slave Data out Delay Time	-	5	-	ns
T_{SSU}	Slave Input Data Set-up Time	-	10	-	ns
T_{SHD}	Slave Input Data hold Time	-	2	-	ns
T_{SSBSU}	Slave Select Bar Set-up Time	-	10	-	ns
T_{SSBHD}	Slave Select Bar hold Time	-	10	-	ns

Note: See *Figure* 21-3,*Figure*21-4

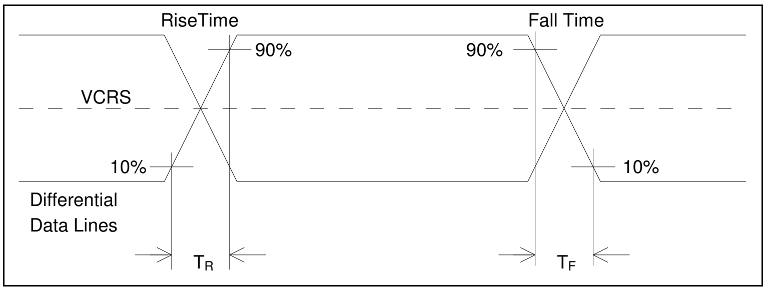


Figure 21-2 USB Data Signal Rise and Fall Time

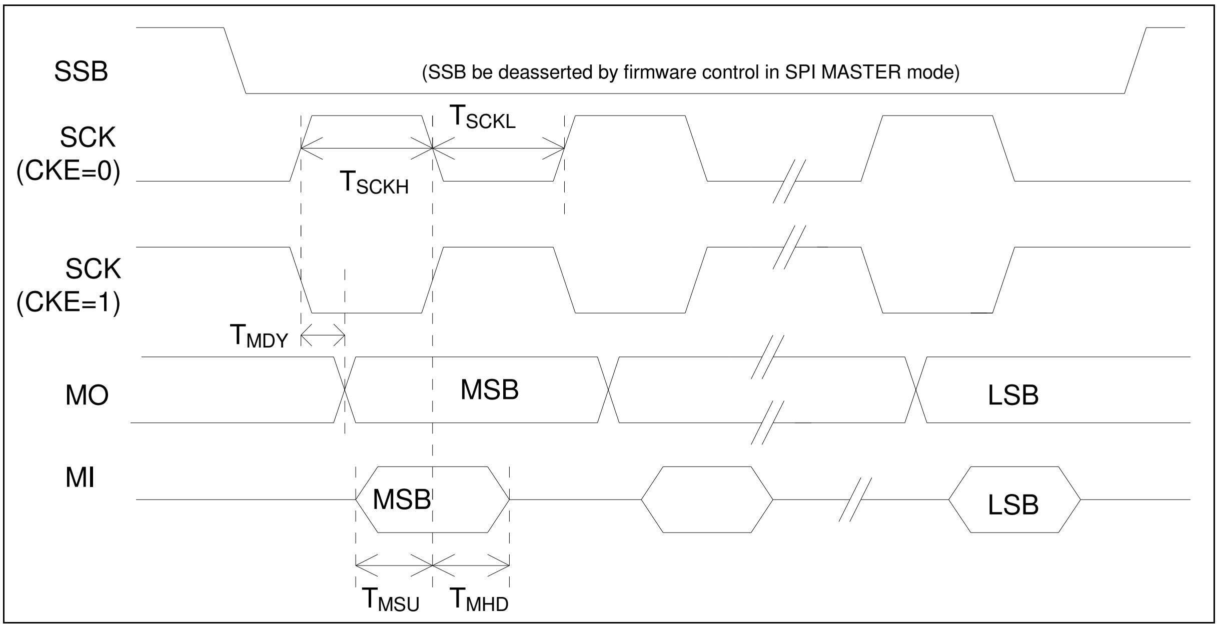


Figure 21-3 SPI Master Timing

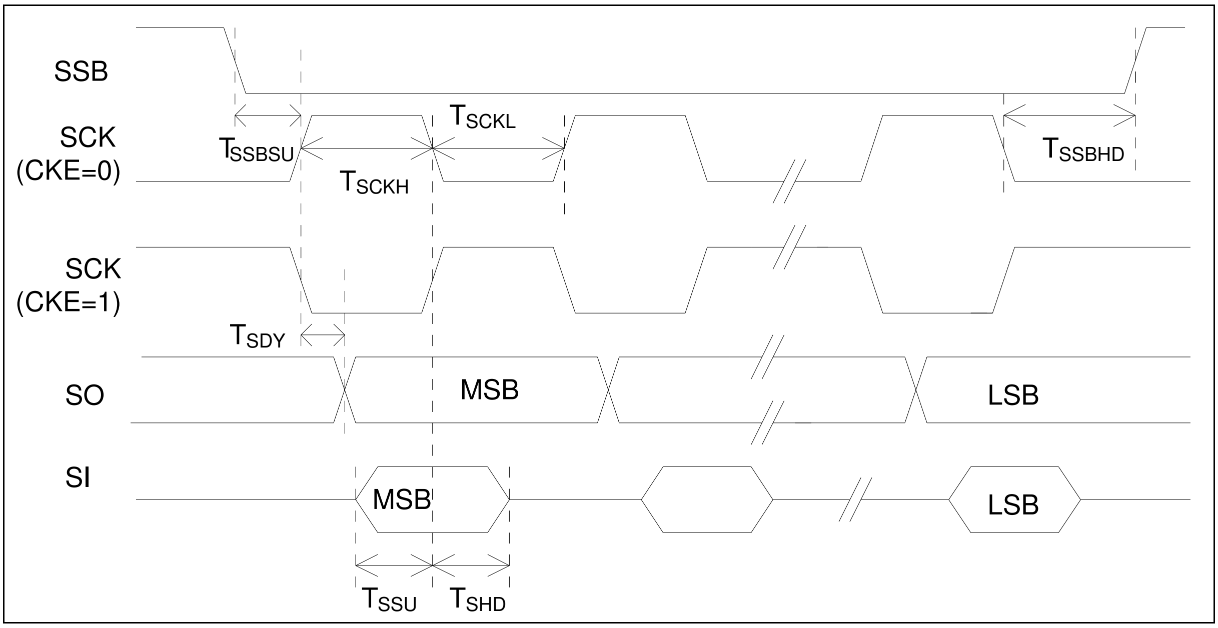
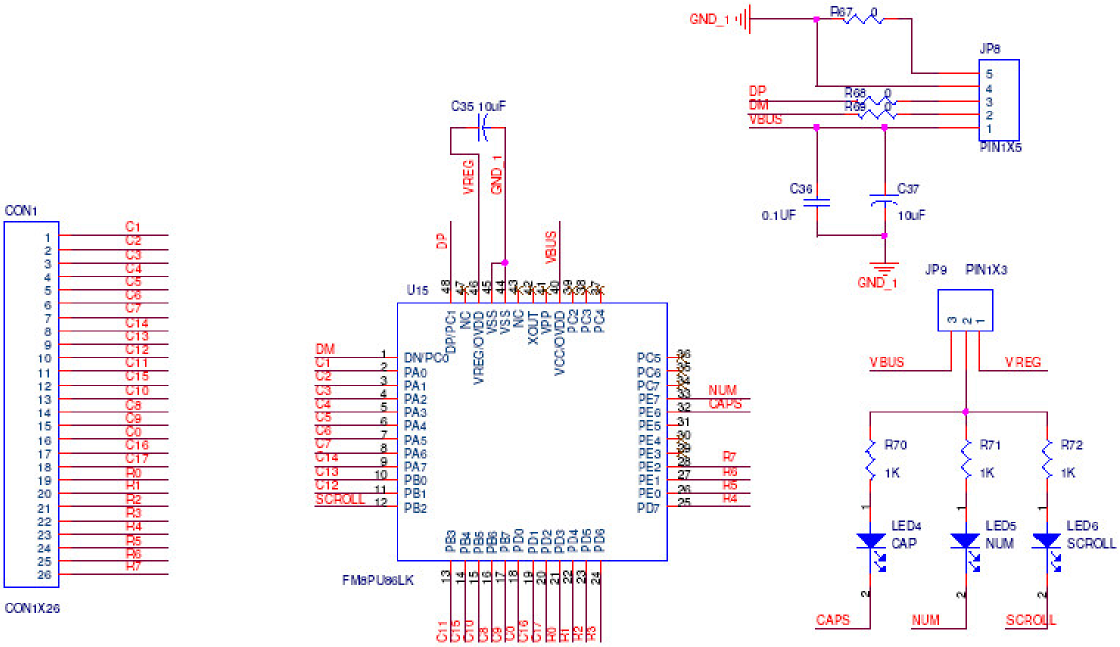


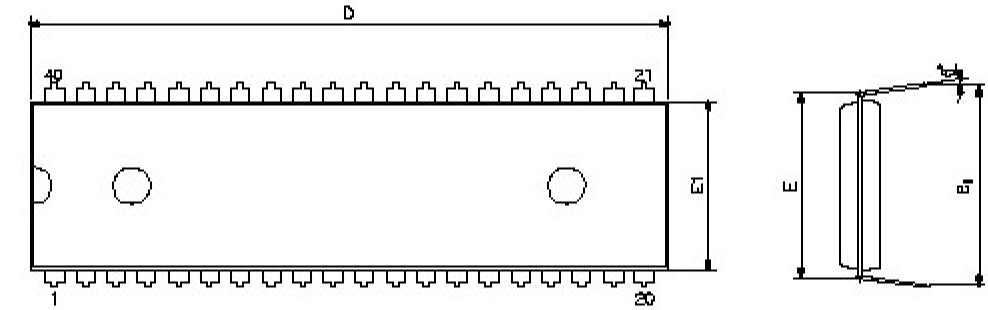
Figure 21-4 SPI Slave Timing

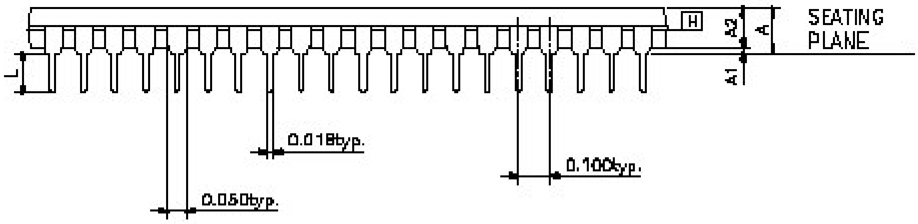
23.0 Application Circuit



24.0 Package Diagrams

■ **40- LEAD(600mil) DIP**



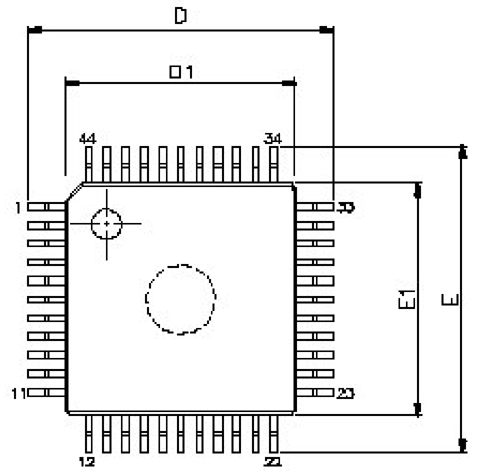


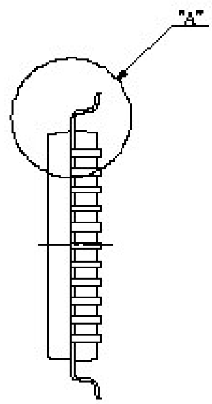
SYMBOLS	MIN.	NOR.	MAX.
A	–	–	0.220
A1	0.015	–	–
A2	0.150	0.155	0.160
D	2.055	2.060	2.070
E		0.600 BSC	
E1	0.540	0.545	0.550
L	0.115	0.130	0.150
e_B	0.630	0.650	0.670
θ	0	7	15

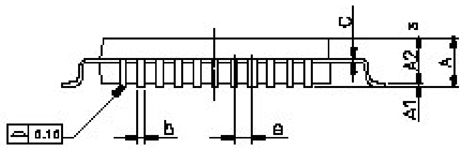
UNIT : INCH

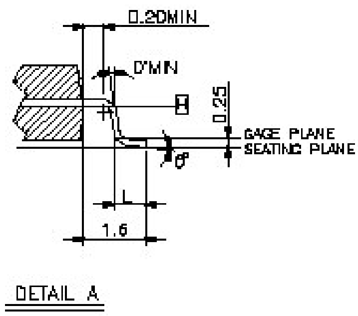
NOTE:
1.JEDEC OUTLINE : MS-011 AC

- **44- LEAD QFP**









SYMBOLS	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.)	
θ°	0	–	7
C	0.1	0.15	0.2

UNIT : mm

NOTES:

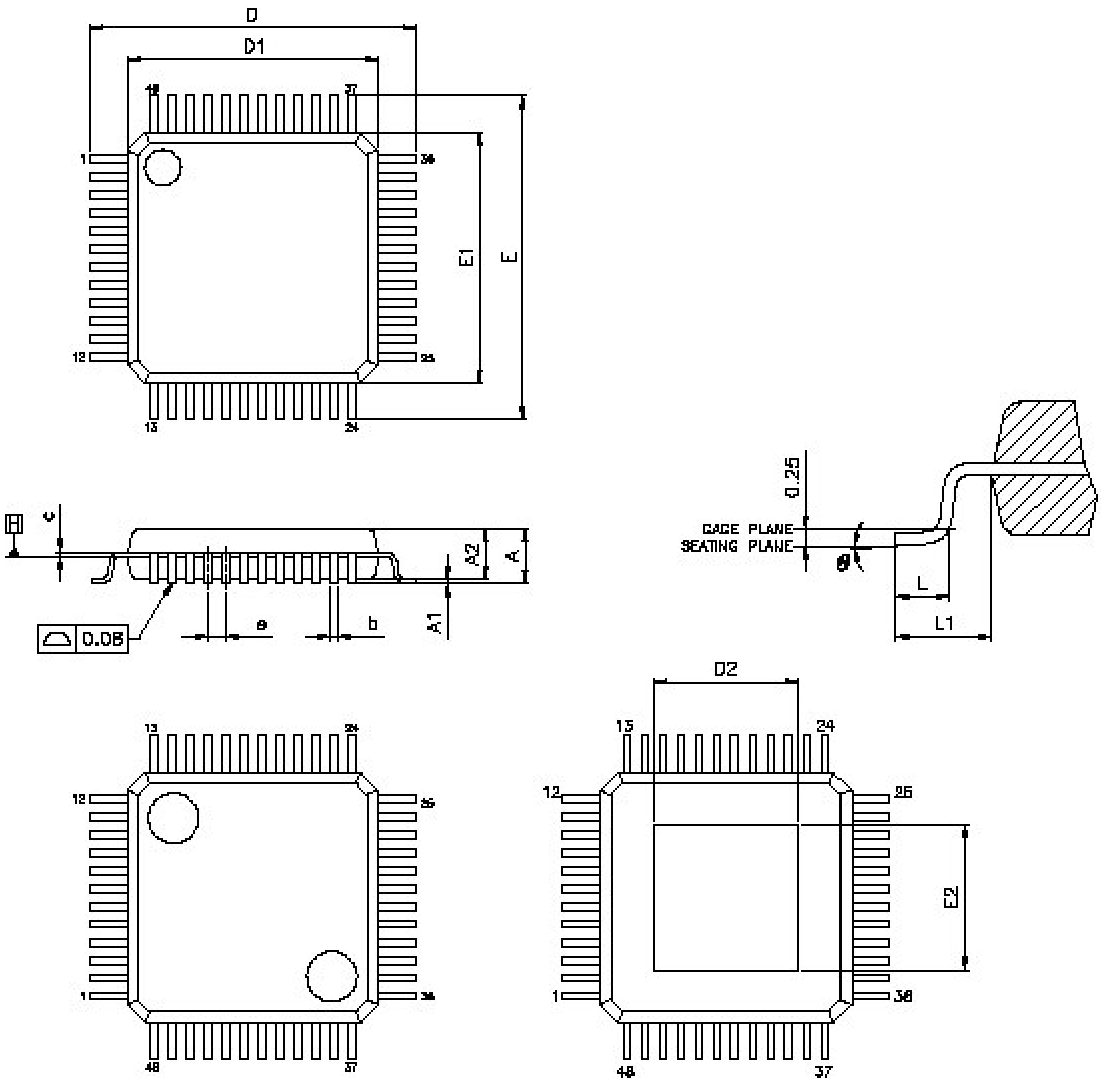
1.JEDEC OUTLINE:MO-108 AA-1

2.DATUM PLANE H IS LOCATED AT THE BOTTOM OF THE MOLD PARTING LINE COINCIDENT WITH WHERE THE LEAD EXITS THE BODY.

3.DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 mm PER SIDE. DIMENSIONS D1 AND E1 DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE H.

4.DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION.

- **LQFP – 48L(7X7mm)**



THERMALLY ENHANCED VARIATIONS ONLY

VARIATIONS (ALL DIMENSIONS SHOWN IN MM)

SYMBOLS	MIN.	NOM.	MAX.
A	--	--	1.60
A1	0.05	--	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
c	0.09	--	0.20
D		9.00 BSC	
D1		7.00 BSC	
E		9.00 BSC	
E1		7.00 BSC	
e		0.50 BSC	
L	0.45	0.60	0.75
L1		1.00 REF	
θ	0°	3.5°	7°

⚠ THERMALLY ENHANCED DIMENSIONS(SHOWN IN MM)

PAD SIZE	E2		D2	
	MIN.	MAX.	MIN.	MAX.
205X205	4.31	5.21	4.31	5.21

NOTES:
1.JEDEC OUTLINE :
MS-026 BBC
MS-026 BBC-HD(THERMALLY ENHANCED VARIATIONS ONLY)
2.DATUM PLANE [H] IS LOCATED AT THE BOTTOM OF THE MOLD PARTING LINE COINCIDENT WITH WHERE THE LEAD EXITS THE BODY.
3.DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 mm PER SIDE. DIMENSIONS D1 AND E1 DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE [H].
4.DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION.