FM8PU83M

Full-Speed USB Controller

Datasheet R1_03.004B

March 24 2015

The datasheet contains new product information. Feeling Technology reveres 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.

Version Control

Version	Date.	Description
R1_0	2014-01-07	Release of prototype
R1_1	2014-02-21	Add application circuit
R1_2	2014-07-01	Updated IO specification
R1_03.004B	2015-03-24	Modify pin standard name, SPI description, Configuration definition, SFR's name, Wake-up example. Update DC Characteristics and modify bit name.

1.0 FM8PU83M Overview

The FM8PU83M is optimum from FM8PU83S product.

- FM8PU83M enhance PS2 mode and I/O voltage support to 1.8V~5.5V
- FM8PU83M functional and coding is compatible to FM8PU83S.
- FM8PU83M configuration option table, please reference page-6.

Compared **FM8PU83M** with FM8PU83S is as follows Table1-1:

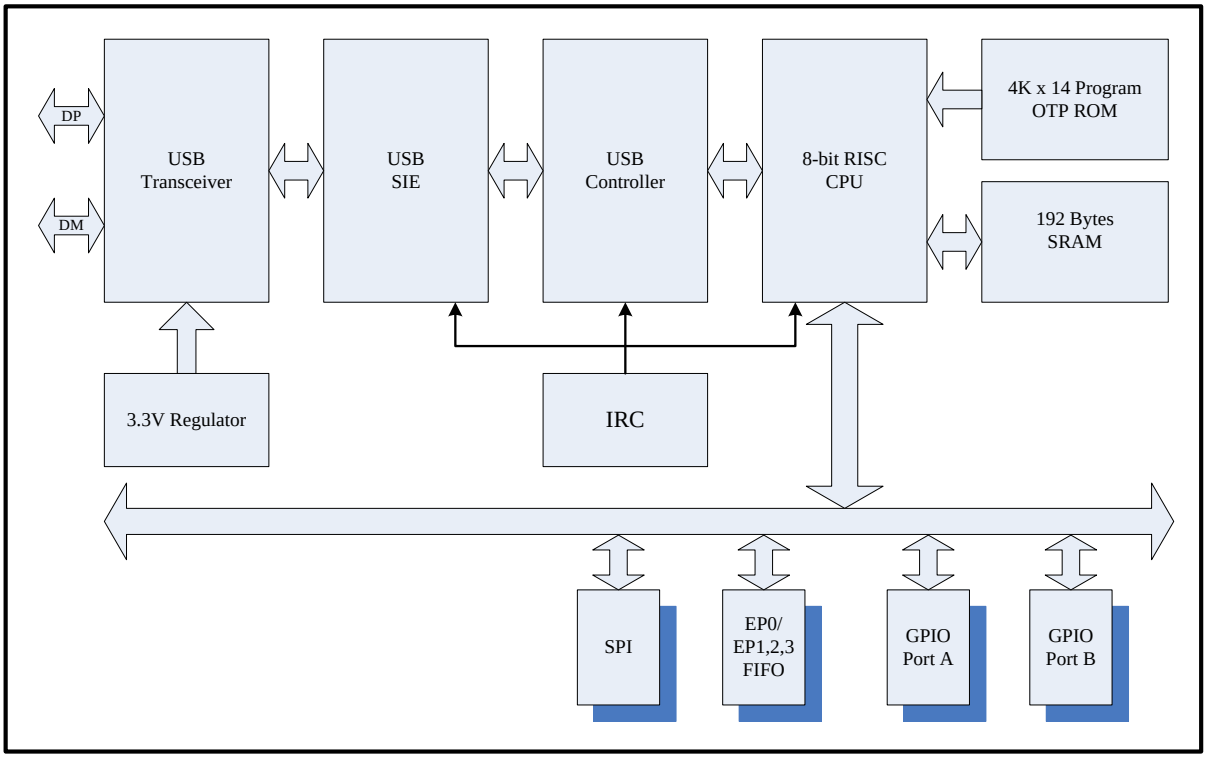
CHIP	**ROM**	RAM	**ENP**	Stack	I/O	**I/O tolerance**	Timer			USB1.1
							WDT	T0	T1	PS2
FM8PU83M	**4kX14 -OTP**	192 bytes	**EP0(TX)-64B EP0(RX) -64B EP1(IN/OUT)-64B EP2(IN/OUT) -64B EP3(IN/OUT) -64B**	8	16+1	**1.8~5.5V**	V	V	V	V V
FM8PU83S	**4kX14 -OTP**	192 bytes	**EP0(TX)-64B EP0(RX) -64B EP1(IN/OUT)-64B EP2(IN/OUT) -64B EP3(IN/OUT) -64B**	**8**	16+1	**1.8~3.6V**	V	V	V	V X

Table 1-1

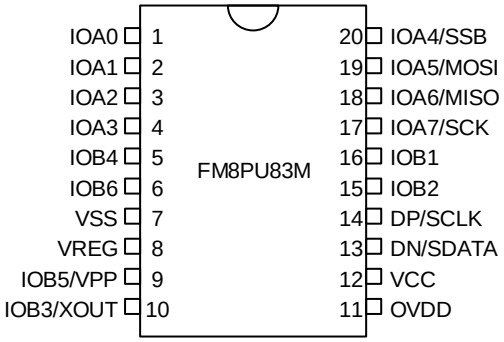
2.0 Features

- Support Full-Speed and Low-Speed USB 1.1 specification.
- Built-in USB Transceiver and 3.3V regulator.
- Support USB Suspend and Resume function.
- One Control IN/OUT, three INT (IN/OUT) /Bulk (IN/OUT) endpoints.
- Support PS2 compatible interface share with USB interface.
- 192 bytes internal SRAM.
- 4K x 14 internal program OTP-ROM
- 8-bit RISC CPU core.
- Support Master/Slave SPI serial Communication Interface.
- Support up to three user configured endpoints.
  - Up to three 64-byte data endpoints (EP0, EP1, EP2, EP3).
- Internal Clock Generator
  - 0.25% Accuracy.
  - Supply 24MHz or 16MHz or 12MHz clock output.
- General-purpose programmable-level IO interface.
- 5V power supply only; GPIO support 1.8V~5.5V interface.
- Supply 3.3V voltage.
- Support SSC to reduce EMI.

3.0 Block Diagram



4.0 Pin Definitions

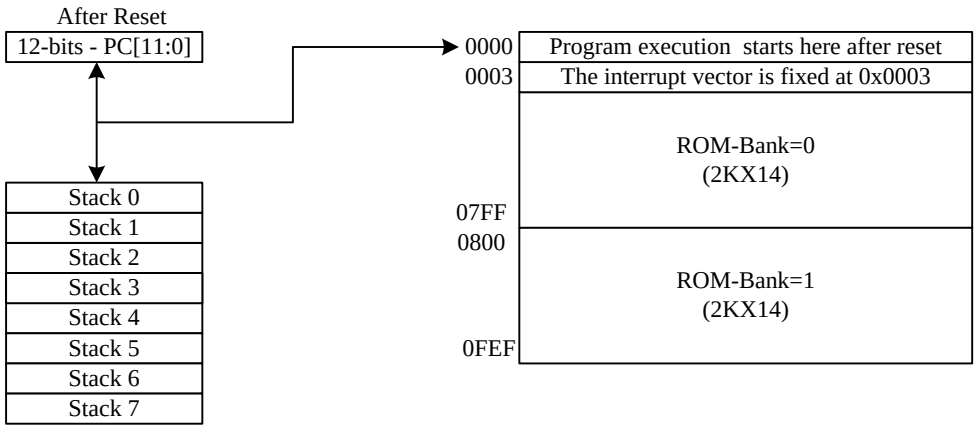


20-pin
FM8PU83M D-Type
PDIP/SOP

5.0 Pin Description

PIN Name	I/O	20-pin D	Description	Note
IOA[7:0]	IO	1,2, 3,4 17, 18, 19, 20,	GPIO Port A capable of sinking up to 50 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. Use IOA7~4 in SPI mode.	
IOB4 IOB6 IOB1 IOB2	IO	5, 6, 16, 15,	GPIO Port B capable of sinking up to 50 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	7	Ground	
VREG	O	8	3.3V Regulator output	
VPP IOB5	I I	9	Programming voltage supply, VCC for normal operation or IOB5 input (pull_high)	
XOUT IOB3	O I/O	10	24MHz or 16MHz or 12MHz internal oscillator output; Or GPIO (Default is pull-high input)	
VCC	P	12	Voltage supply	
DN	IO IO	13	USB differential data lines (D-) IOC0	
DP	IO IO	14	USB differential data lines (D+) IOC1	
OVDD	P	11	I/O pad power Voltage supply 1.8 ~ 5.5V	

6.0 Program Memory Mapping



The FM8PU83M 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 4K program memory address range for FM8U83M, there is another two bits to specify the program memory page. The paging bit comes from the ROMBANK bit (PUMODE[5]). When doing a CALL or GOTO instruction, the user must ensure that page bit ROMBANK are programmed so that the desired program memory page is addressed. When one of return instructions is executed, the entire 12-bit PC is POPed from the stack. Therefore, manipulation of the ROMBANK is not required for return instructions.

6.1 Configuration Option Table

Name	Description
FSEN	→ Full Speed (default) → Low Speed
ENWDT	→ Enable Watch dog (default) → Disable Watch dog
SELCPUCLK	CPU Operating Frequency Selection Bit If FSEN=Full Speed: → Microprocessor clock is 24MHz (default) → Microprocessor clock is 12MHz If FSEN=Low Speed: → Microprocessor clock is 12MHz → Microprocessor clock is 6MHz
WDTSEL	Watch dog time-out Selection Bit → 72mS → 288mS → 4.5mS → 18mS (default)
SELPOR	Power-on reset extend timing Selection Bit → 482uS → 4.462mS → 8.814mS → 17.518mS (default)
SELXOUT	Internal clock Output Frequency Selection Bit → 24MHz (default) → 16MHz → 12MHz
PROTECT	Code Protection Bit → NO, OTP code protection off (default) → YES, OTP code protection on

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			

08h	PAMODE0
09h	PAMODE1
0Ah	PBMODE0
0Bh	PBMODE1

RAMBNK[1:0] / Address	Description			
	0 0 Bank 0	0 1 Bank 1	1 0 Bank 2	1 1 Bank 3
0Fh	INTFLAG1			
10h	T0RLD			
11h	USBADDR			
12h	USBMDCFG			
13h	EP0RXST			
14h	EP0TXST			
15h	EP1TRXST			
16h	EP2TRXST			
17h	PAIE			
18h	PACON			
19h	USBST			
1Ah	TMR1			
1Bh	T1RLD			
1Ch	SPIRXB			
1Dh	SPITXB			
1Eh	SPISTAT			
1Fh	SPICON			
20h	PUMODE			
21h	CLKCFG			
22h~26h	General Purpose Register			
27h	EP0TXHBK			
28h	EP0RXHBK			
29h	EP1TRXHCN			
2Ah	EP2TRXHCN			
2Bh	EP3CN			
2C	EP3TRXST			
2D	EP3AR			
2E	EP3DAP			
2F	General Purpose Register			
30h | 37h	ENP0RX (8Bytes* 8-Bank = 64Bytes)			
38h | 3Fh	ENP0TX (8Bytes* 8-Bank = 64Bytes)			
40h | 47h	ENP1TRX (8Bytes* 8-Bank = 64Bytes)			
48h | 4Fh	ENP2TRX (8Bytes* 8-Bank = 64Bytes)			
50h | 7Fh	General Purpose Registers 0	General Purpose Registers 1	General Purpose Registers 2	General Purpose Registers 3

ENDPOINT-64 bytes mapping

EP0-RX:

Address	RX0BK=0	RX0BK=1	RX0BK=2	RX0BK=3	RX0BK=4	RX0BK=5	RX0BK=6	RX0BK=7
30h 37h	BK-0 (8Bytes)	BK-1 (8Bytes)	BK-2 (8Bytes)	BK-3 (8Bytes)	BK-4 (8Bytes)	BK-5 (8Bytes)	BK-6 (8Bytes)	BK-7 (8Bytes)

EP0-TX:

Address	TX0BK=0	TX0BK=1	TX0BK=2	TX0BK=3	TX0BK=4	TX0BK=5	TX0BK=6	TX0BK=7
38h 3Fh	BK-0 (8Bytes)	BK-1 (8Bytes)	BK-2 (8Bytes)	BK-3 (8Bytes)	BK-4 (8Bytes)	BK-5 (8Bytes)	BK-6 (8Bytes)	BK-7 (8Bytes)

EP1TRX:

Address	EP1BK=0	EP1BK=1	EP1BK=2	EP1BK=3	EP1BK=4	EP1BK=5	EP1BK=6	EP1BK=7
40h 47h	BK-0 (8Bytes)	BK-1 (8Bytes)	BK-2 (8Bytes)	BK-3 (8Bytes)	BK-4 (8Bytes)	BK-5 (8Bytes)	BK-6 (8Bytes)	BK-7 (8Bytes)

EP2TRX:

Address	EP2BK=0	EP2BK=1	EP2BK=2	EP2BK=3	EP2BK=4	EP2BK=5	EP2BK=6	EP2BK=7
48h 4Fh	BK-0 (8Bytes)	BK-1 (8Bytes)	BK-2 (8Bytes)	BK-3 (8Bytes)	BK-4 (8Bytes)	BK-5 (8Bytes)	BK-6 (8Bytes)	BK-7 (8Bytes)

EP3TRX:

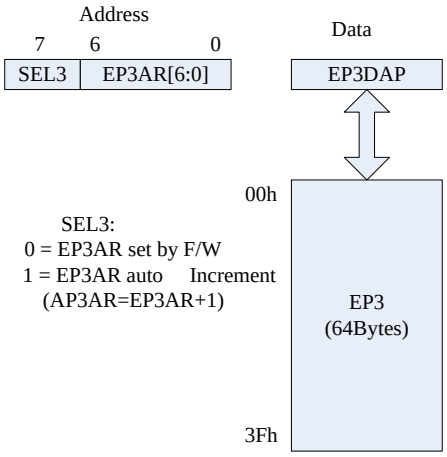


TABLE 7-2: The Registers Controlled IOST or IOSTR Instructions

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
08h (r/w)	PAMODE0	Port A Mode Control Register							
09h (r/w)	PAMODE1								
0Ah (r/w)	PBMODE0	Port B Mode Control Register							
0Bh (r/w)	PBMODE1								

Note: Power on reset PBMODE1=8'h00, PBMODE0=8'h00
PAMODE1=8'h00, PAMODE1=8'h00

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 (08h~0Bh) 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 '0'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(2mA)
0			Sink (8mA)
1	1	0	Resistive(14KΩ)
0			Sink (2mA)
1	1	1	Normal Drive(2mA)
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 0							
02h (r/w)	PCL	Low order 8 bits of PC							
03h (r/w)	STATUS	--	--	--	/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/w)	PORTC	--	--	--	--	--	--	IOC1	IOC0
08h (r/w)	PCON	WDTE	WDTSL	WDTPS2	WDTPS1	WDTPS0	LVR	--	--
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	--	SOFIE	RSTIE	SPITXIE	SPIRXIE	T1IE	T0IE
0Dh (r/w)	INTFLAG	PBIF	PAIF	SOFIF	RSTIF	SPITXIF	SPIRXIF	T1IF	T0IF
0Eh (r/w)	INTEN1	RSMIE	SUSIE	RX0IE	TX0IE	TRX1IE	TRX2IE	TRX3IE	--
0Fh (r/w)	INTFLAG1	RSMIF	SUSIF	RX0IF	TX0IF	TRX1IF	TRX2IF	TRX3IF	--
10h (r/w)	T0RLD	8-bit real-time TMR0 overflow reload							
11h(r/w)	USBADDR	EUSBAR	USBAR6	USBAR5	USBAR4	USBAR3	USBAR2	USBAR1	USBAR0
12h(r/w)	USBMDCFG	SUSPMD	RESMD	CTRRD	RX0RDY	EP1CFG	EP2CFG	EP3CFG	--
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)	EP1TRXST	TRX1RDY	TRX1TGL	EP1STAL	EP1DIR	TRX1CN[3:0]			

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
16h(r/w)	EP2TRXST	TRX2RDY	TRX2TGL	EP2STAL	EP2DIR	TRX2CN[3:0]			
17h(r/w)	PAIE	PAIE7	PAIE6	PAIE5	PAIE4	PAIE3	PAIE2	PAIE1	PAIE0
18h(r/w)	PACON	PACON7	PACON6	PACON5	PACON4	PACON3	PACON2	PACON1	PACON0
19h(r/w)	USBST	DPMF1	DPMF0	--	SELINT	LVDT	--	--	--
1Ah(r/w)	TMR1	8-bit real-time clock/counter 1							
1Bh(r/w)	T1RLD	8-bit real-time TMR1 overflow reload							
1Ch(r)	SPIRXB	RX7	RX6	RX5	RX4	RX3	RX2	RX1	RX0
1Dh(r/w)	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	USBMOD	--	ROMBNK	--	RAMBNK1	RAMBNK0	PS2EN	USBEN
21h(r/w)	CLKCFG	--	--	--	--	--	INCODS	--	--
27h(r/w)	EP0TXHBK	--	TX0CN[6:4]			--	TX0BK[2:0]		
28h(r/w)	EP0RXHBK	--	RX0CN[6:4]			--	RX0BK[2:0]		
29h(r/w)	EP1TRXHCN	RX1ERR	TRX1CN[6:4]			--	EP1BK[2:0]		
2A(r/w)	EP2TRXHCN	RX2ERR	TRX2CN[6:4]			--	EP2BK[2:0]		
2B(r/w)	EP3CN	EP3TRX	EP3CN[6:0]						
2C(r/w)	EP3TRXST	TRX3RDY	TX3TGL	EP3STAL	EP3DIR	RX3ERR	--		
2D(r/w)	EP3AR	SEL3	EP3AR[6:0]						
2E(r/w)	EP3DAP	EP3DAP[7:0]							

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
7~5	--	Unimplemented, read as '0'.	Yes	No	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	--	Unimplemented, read as '0'.	Yes	No	0
6~0	FSR[6:0]	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	IOA[7:0]	Port A data input/output	Yes	Yes	00h

Address 06H: Port B (PORTB)

Bits	Name	Description	Read	Write	Default
7~0	IOB[7:0]	Port B data input/output	Yes	Yes	00h

Address 07H: Port C (PORTC)

Bits	Name	Description	Read	Write	Default
7~2	--	Unimplemented, read as '0'.	Yes	No	00h
1	IOC1	D+ The state of the D+ pins can be read at Port C data register.	Yes	Yes	1
0	IOC0	D- 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. 0: If Watch-dog timer out, Device only SUSPMD (12.7H) be clear.	Yes	Yes	1
5~3	WDTPS[2:0]	WDT prescaler	Yes	Yes	0h
2	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	1

Bits	Name	Description	Read	Write	Default
1~0	--	Unimplemented, read as '0'.	Yes	No	0h

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[7:0]	Port B interrupt enable bits (0:disable ,1:enable)	Yes	Yes	00h

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

Bits	Name	Description	Read	Write	Default
7~0	PBCON[7:0]	0 : falling edge(port B), 1: Pin-changed (port B)	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	T1PS[2:0]	Timer1 Prescaler Rate 0:1/4 ,1:1/8 ,2:1/16 ,3:1/32,4:1/64,5:1/128,6:1/256,7:1/512	Yes	Yes	0h
3	T0ON	Timer0 module Enable bit	Yes	Yes	0
2~0	T0PS[2:0]	Timer0 Prescaler Rate 0:1/4 ,1:1/8 ,2:1/16 ,3:1/32,4:1/64,5:1/128,6:1/256,7:1/512	Yes	Yes	0h

T0/1 PS2, PS1, PS0	Prescaler Rate	Number of MCU clock
3'b000	1/4	4
3'b001	1/8	8
3'b010	1/16	16
3'b011	1/32	32
3'b100	1/64	64
3'b101	1/128	128
3'b110	1/256	256
3'b111	1/512	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	--	Unimplemented, read as '0'.	Yes	No	0
5	SOFIE	SOF interrupt enable bit.(0:disable,1:enable)	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	PAIF	Port A interrupt flag, write 0 clear flag	Yes	Yes	0
5	SOFIF	SOF interrupt 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	TRX1IE	Endpoint 1 transmit/Received successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
2	TRX2IE	Endpoint 2 transmit/Received successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
1	TRX3IE	Endpoint 3 transmit/Received successfully interrupt enable bit. (0:disable,1:enable)	Yes	Yes	0
0	--	Unimplemented, read as '0'.	Yes	No	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	TRX1IF	Endpoint 1 Transmit /Received successfully interrupt flag, write 0 clear flag	Yes	Yes	0
2	TRX2IF	Endpoint 2 Transmit /Received successfully interrupt flag, write 0 clear flag	Yes	Yes	0
1	TRX3IF	Endpoint 3 Transmit /Received successfully interrupt flag, write 0 clear flag	Yes	Yes	0
0	--	Unimplemented, read as '0'.	Yes	No	0

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 1: Enable Device Address 0: Disable Device Address	Yes	Yes	0
6~0	USBAR[6:0]	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	--	Unimplemented, read as '0'.	Yes	No	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	RX0CN[3:0]	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	--	Unimplemented, read as '0'.	Yes	No	0
3~0	TX0CN[3:0]	EP0 transmit data byte count	Yes	Yes	0h

Address 15H: Endpoint 1 transmit/Received status (EP1TRXST)

Bits	Name	Description	Read	Write	Default
7	TRX1RDY	EP1 ready for transmit/Received ,clear by H/W while TRX1IFoccurs	Yes	Yes	0
6	TRX1TGL	When EP1DIR set 0, TRX1TGL is EP1 transmit DATA1/DATA0 PID packet. When EP1DIR set 1, TRX1TGL is received data toggle bit.(1:DATA1; 0:DATA0)	Yes	Yes	0
5	EP1STAL	EP1 will stall IN/OUT packet while this bit set 1	Yes	Yes	0
4	EP1DIR	0: IN transfer 1:out transfer	Yes	Yes	0
3~0	TRX1CN[3:0]	EP1 transmit data byte count when EP1DIR=0 EP1 Received data byte count when EP1DIR=1	Yes	Yes	0h

Address 16H: Endpoint 2 transmit/Received status (EP2TRXST)

Bits	Name	Description	Read	Write	Default
7	TRX2RDY	EP2 ready for transmit /Received, clear by H/W while TRX2IF occurs	Yes	Yes	0
6	TRX2TGL	When EP2DIR set 0, TRX2TGL is EP2 transmit DATA1/DATA0 PID packet. When EP2DIR set 1, TRX2TGL is received data toggle bit.(1:DATA1; 0:DATA0)	Yes	Yes	0
5	EP2STAL	EP2 will stall IN/OUT packet while this bit set 1	Yes	Yes	0
4	EP2DIR	0: IN transfer 1:out transfer	Yes	Yes	0
3~0	TRX2CN[3:0]	EP2 transmit data byte count when EP2DIR=0 EP2 Received data byte count when EP2DIR=1	Yes	Yes	0h

Address 17H: Port A Interrupt Control Register (PAIE)

Bits	Name	Description	Read	Write	Default
7~0	PAIE[7:0]	Port A interrupt enable bits (0:disable ,1:enable)	Yes	Yes	00h

Address 18H: Port B Wake-up Control Register (PACON)

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

Address 19H: USB status register 1(USBST)

Bits	Name	Description	Read	Write	Default
7~6	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
5	--	Unimplemented, read as '0'.	Yes	No	0
4	SELINT	SELINT : 0: When handshaking is ACK packet, the S/W generates interrupt for ENP1,2.,3 1: When handshaking is any packet, he S/W generates interrupt for ENP1,2,3	Yes	Yes	1'b0
3	LVDT	When VBUS voltage less than 3.8 V, the LVDT be set 0. When VBUS voltage greater than 3.8V the LVDT be set 1.	Yes	No	X
2~0	--	Unimplemented, read as '0'.	Yes	No	00h

Address 1AH: Timer1 (TMR1)

Bits	Name	Description	Read	Write	Default
7~0	TMR1	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[7:0]	Once the 8-bits data have been received, the data in SPI shift register (SPISR) will be moved to the SPIRXB register. The data must be read out before the next 8-bits data reception is completed if needed. The RXBF flag is set when the data in SPISR is moved to the SPIRXB register, and cleared as the SPIRXB register reads.	Yes	No	XXh

Address 1DH: SPI transmit Buffer Register (SPITXB)

Bits	Name	Description	Read	Write	Default
7~0	TX[7:0]	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	--	Unimplemented, read as '0'.	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	--	Unimplemented, read as '0'.	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[2:0]	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	--	Unimplemented, read as '0'.	Yes	No	0
5	ROMBNK	ROM bank select bit. 0 : bank-0 1:bank-1	Yes	Yes	0h
4	--	Unimplemented, read as '0'.	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 (IOC1), D-/Data (IOC0) 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: CLKCFG

Bits	Name	Description	Read	Write	Default
7~4	--	Unimplemented, read as '0'.	Yes	No	4'b0000
2	INCODS	Internal clock output disable 1: Disable internal IRC clock output. XOUT pin will pull high and as input pin on IOB3. 0: Enable internal IRC clock output. The IRC clock is driven output to the XOUT pin.	Yes	Yes	0
1~0	--	Unimplemented, read as '0'.	Yes	No	2'b00

Address 22~26H: General purpose Register

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

Address 27H: Endpoint 0 TX High byte Count and Bank (EP0TXHBK)

Bits	Name	Description	Read	Write	Default
7	--	Unimplemented, read as '0'.	Yes	No	0
6~4	TX0CN[6:4]	EP0 Transmit data High byte count	Yes	Yes	0h
3	--	Unimplemented, read as '0'.	Yes	No	0
2~0	TX0BK[2:0]	EP0 Transmit data Bank	Yes	Yes	0h

Address 28H: Endpoint 0 RX High byte Count and Bank (EP0RXHBK)

Bits	Name	Description	Read	Write	Default
7	--	Unimplemented, read as '0'.	Yes	No	0
6~4	RX0CN[6:4]	EP0 Received data High byte count	Yes	No	0h
3	--	Unimplemented, read as '0'.	Yes	No	0
2~0	RX0BK[2:0]	EP0 Received data Bank	Yes	Yes	0h

Address 29H: Endpoint 1 TX/RX High byte Count (EP1TRXHCN)

Bits	Name	Description	Read	Write	Default
7	RX1ERR	EP1 received data error on OUT transfer	Yes	No	0
6~4	TRX1CNT[6:4]	EP1 transmit data High byte count when EP1DIR=0 (write only) EP1 Received data High byte count when EP1DIR=1 (read only)	Yes	Yes	0h
3	--	Unimplemented, read as '0'.	Yes	No	0
2~0	EP1BK[2:0]	EP1 Transmit/ received data Bank	Yes	Yes	0h

Address 2AH: Endpoint 2 TX/RX High byte Count (EP2TRXHCN)

Bits	Name	Description	Read	Write	Default
7	RX2ERR	EP2 received data error on OUT transfer	Yes	No	0
6~4	TRX2CNT[6:4]	EP2 transmit data High byte count when EP2DIR=0 (write only) EP2 Received data High byte count when EP2DIR=1 (read only)	Yes	Yes	0h
3	--	Unimplemented, read as '0'.	Yes	No	0
2~0	EP2BK[2:0]	EP2 Transmit/ received data Bank	Yes	Yes	0h

Address 2BH: Endpoint 3 TX/RX byte Count (EP3CN)

Bits	Name	Description	Read	Write	Default
7	EP3TRX	When EP3 of Device is receiving from Host, the EP3TRX High.(EP3DIR = 1 (OUT)) When EP3 of Device is transmitting to Host, the EP3TRX is high.(EP3DIR = 0 (IN))	Yes	No	0
6~0	EP3CN[6:0]	EP3 transmit data High byte count when EP3DIR=0 (write only) EP3 Received data High byte count when EP3DIR=1 (read only)	Yes	Yes	0h

Address 2CH: Endpoint 3 transmit/Received status (EP3TRXST)

Bits	Name	Description	Read	Write	Default
7	TRX3RDY	EP3 ready for transmit /Received, clear by H/W while TRX3IF occurs	Yes	Yes	0
6	TRX3TGL	When EP3DIR set 0, TRX3TGL is EP3 transmit DATA1/DATA0 PID packet. When EP3DIR set 1, TRX3TGL is received data toggle bit.(1:DATA1; 0:DATA0)	Yes	Yes	0
5	EP3STAL	EP3 will stall IN/OUT packet while this bit set 1	Yes	Yes	0
4	EP3DIR	0: IN transfer 1:out transfer	Yes	Yes	0
3	RX3ERR	EP3 received data error on OUT transfer	Yes	No	0
2~0	--	Unimplemented, read as '0'.	Yes	No	0h

Address 2DH: Address of EP3-FIFO register (EP3AR)

Bits	Name	Description	Read	Write	Default
7	SEL3	Select EP3-FIFO Addressing. 0 = EP3AR set by F/W for indirect addressing when MCU Read/Write EP3-FIFO by F/W. 1 = EP3AR auto increment from EP3AR[6:0] to EP3CN when MCU Read/Write EP3-FIFO by F/W	Yes	Yes	0
6~0	EP3AR[6:0]	Address of EP3-FIFO	Yes	Yes	0h

Address 2EH: Endpoint 3 Transmit/Received data pointer register (EP3DAP)

Bits	Name	Description	Read	Write	Default
7~0	EP3DAP[7:0]	AS IN token: when Device transmit data from EP3-FIFO, the F/W need write data to EP3-FIFO through EP3DAP register. AS OUT token: when Device received data to EP3-FIFO, the F/W need read data from EP3-FIFO through EP3DAP register.	Yes	Yes	00h

Address 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)

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

Address 38H~3FH: EP0 Transmitter Buffer (EP0TX)

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

Address 40H~47H: EP1 TX/RX Buffer (EP1TRX)

Address	Description	Read	Write	Default
40H	Start address of EP1TRX	Yes	Yes	XXh
40H~47H	Buffer length is 8 bytes for EP1TRX (8 Banks)	Yes	Yes	XXh

Address 48H~4FH: EP2 TX/RX Buffer (EP2TRX)

Address	Description	Read	Write	Default
48H	Start address of EP2TRX	Yes	Yes	XXh
48H~4FH	Buffer length is 8 bytes for EP2TRX (8 Banks)	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 R-plane register	ACC → R-plane register	1	-
IOSTR	**R**	Read R-plane register	R-plane 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

Mnemonic, Operands	Description	Operation	Cycles	Status Affected
RETIA I	Return, place Immediate in ACC	I → ACC, Top of Stack → PC	2	-
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
WDT :Watchdog Timer Counter
GIE :Global interrupt enable bit
$\overline{TO}$:Time-out bit
$\overline{PD}$:Power-down bit
C :Carry bit
DC :Digital carry bit
Z :Zero bit

9.0 Internal Oscillator

The internal oscillator can provide an operating clock and sensor clock. The XOUT pin output 24MHz or 16MHz or 12MHz.The selected clock used configured bit.

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).

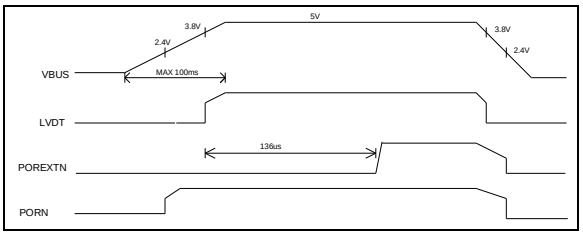


Figure 10-1 power on reset timing

11.0 Suspend Mode

The FM8PU83M 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 Line-state (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 or PA interrupt

```
PBITASK:
        bcr   INTFLAG,PBIF_B
        bcr   USBMDCFG,SUSPMD_B
//Device remove wake up host to set 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 mode:

```
        bcr   USBST,DPMF0_B
        call  delay_1ms
        bsr   USBST,DPMF0_B
//Device remove wake up host to send K state
```

(b) WakeUp for Low_speed mode:

```
        bcr   USBST,DPMF1_B
        call  delay_1ms
        bsr   USBST,DPMF1_B
//Device remove wake up host to send 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 IOA4, IOA5, IOA6 and IOA7 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.

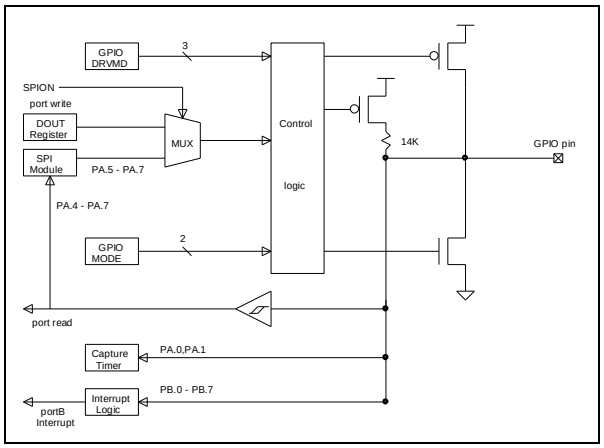


Figure 12-1 Block Diagram of GPIO port

13.0 USB and PS2 Mode Detection

The FM8PU83M 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.

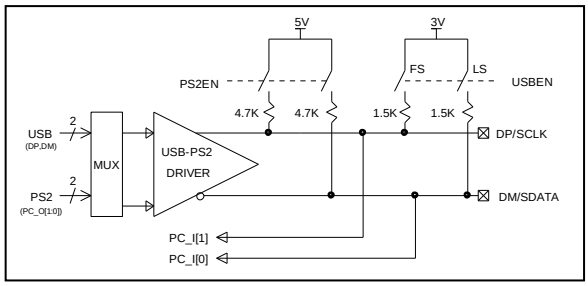


*Figure*13-1 Block diagram of USB/PS2 connections

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 EP2 and EP3

The endpoint 1,2,3 is used for interrupt /Bulk TX/RX(IN/OUT) .The maximum allowable interrupt data payload size 64 bytes for full-speed or low-speed transaction.
The micro-controller needs to program the maximum packet size registers ({TRX1CNT, TRX2CNT, TRX3CNT) properly to avoid any size errors. The interrupt transfer type is designed to support those devices that need to send data infrequently but with bounded service period.

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 15-1.

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 15-1 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*15-1.

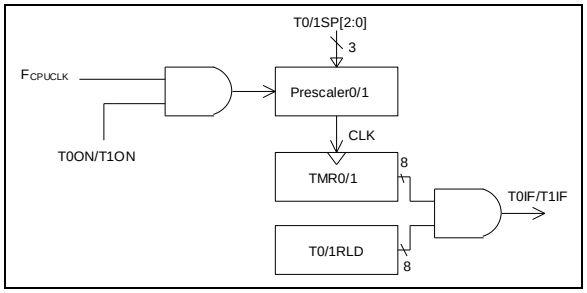


*Figure*15-1 Block diagram of Timer0/1

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 does 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* 15-2.

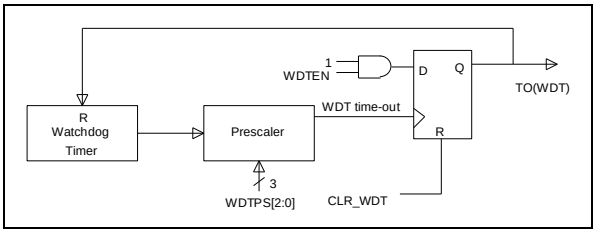


Figure 15-2 Block diagram of WDT

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. Slave Select (SSB): IOA4
2. Master Output Slave Input (MOSI): IOA5
3. Master Input Slave Output (MISO): IOA6
4. Serial Clock (SCK): IOA7

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

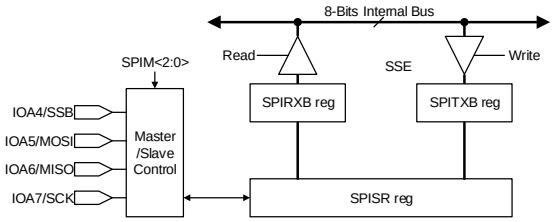


Figure 17-1 Block diagram of SPI

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

The SPI mode timing is show in *Figure* 18-1 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 software. 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 SPIRXB 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 RXBF 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 SPIRXB register before next byte transmission begin finished if needed.

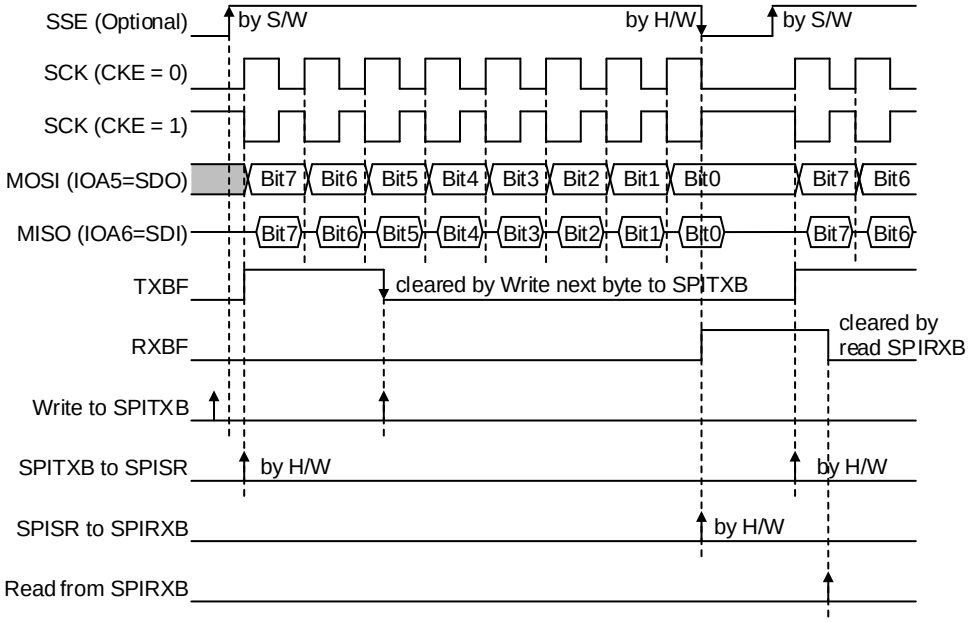


Figure 17-2 SPI Mode Timing (Master Mode)

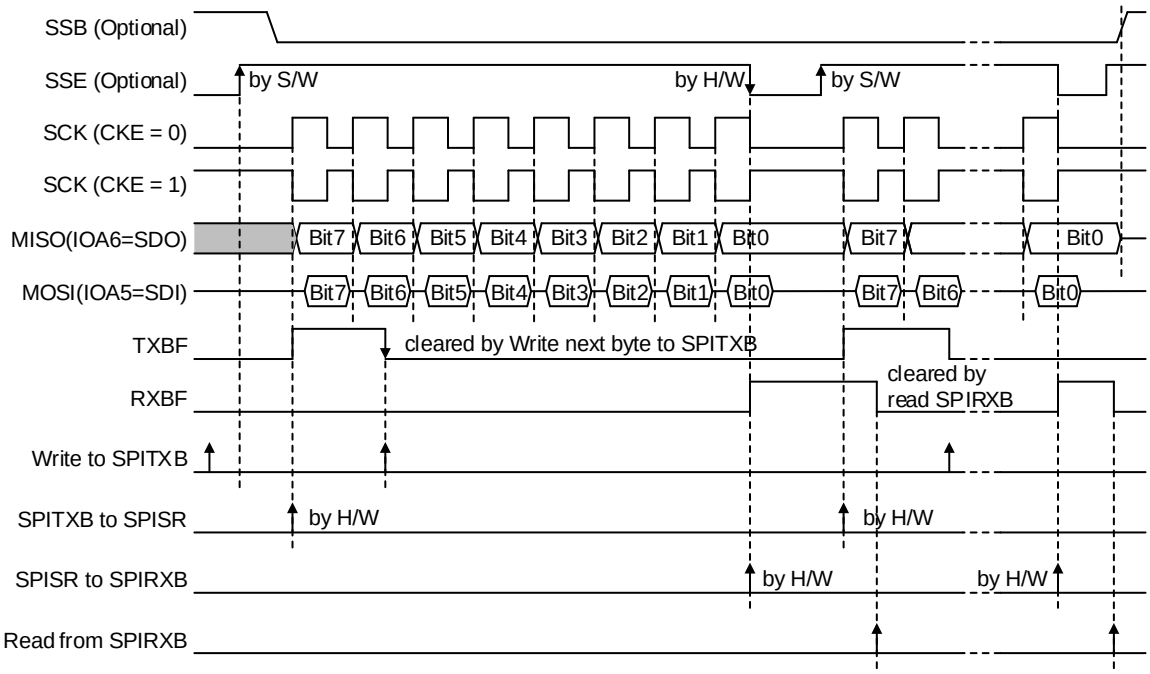


Figure 17-2 The SPI mode Timing (Slave Mode, with SSB control enable/disable)

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   //  MOSI(IOA5) 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 MOSI(IOA5) are connected to MISO(IOA6), the SDO
                                        // need pull-high in received.
            movar   SPITXB
            bcr     SPISTAT,RXBF_B
            bsr     SPICON,SSE_B        // Asserted SSE (start to received)
WAITRXB:    btrss   SPISTAT,RXBF_B      // wait received interrupt
            goto    WAITRXB
            bcr     SPISTAT,RXBF_B      // Clear RXBF bit
            return
```

19. Interrupt

The FM8PU83M 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 (PORTB0~7) and GPIO (PORTA0~7) external interrupt.
6. USB bus reset interrupt.
7. USB resume interrupt.
8. USB suspend interrupt.
9. USB endpoint0, 1, 2 and 3 interrupt.

The FM8PU83M 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
Voltage on all GPIO pin	Respect to VSS	-0.3	VCC+0.3V	V

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 = 3mA[1]	Vcc=5.5V,No GPIO loading IRC Operating ,MCU=12MHz	-	3	mA
I_{CC2}	Operating Voltage Current Typical = 4.5mA[1]	Vcc=5.5V,No GPIO loading IRC Operating, MCU=24MHz	-	4.5	mA
I_{SPD}	Suspend Current	With IRC(17us)	-	100	uA
V_{POR}	Power on Reset		2.4	-	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	-	15	17	KΩ
V_{OVDD}	GPIO support voltage	-	1.8	5.5	V
V_{OL}	Output Low Voltage(High drive)	OVDD=3.3/5V, I_{OL}=16mA		0.8	V
V_{OL}	Output Low Voltage(Medium drive)	OVDD=3.3/5V,I_{OL}=8mA	-	0.4	V
V_{OL}	Output Low Voltage(Low drive)	OVDD=3.3/5VI_{OL}=2mA	-	0.4	V
V_{OH}	Output High Voltage	I_{OH}=2mA	OVDD-1	-	V
V_{IL1}	Input Low Voltage	OVDD=1.8V~5.5V		0.3OVDD	V
V_{IH1}	Input High Voltage	OVDD=1.8V~5.5V	0.7OVDD		V

Note:
[1]: Bench measurements on nominal operating conditions. Vcc = Vbus=5V

22. AC Characteristics

22.1 Clock Period

Symbol	Parameter	Conditions	Values		Unit
			min.	max.	
$F_{IRC}1$	Internal Clock Frequency	Internal Clock Enable; After CP calibration	24-0.25%	24+0.25%	MHz
F_{EXO}	External Clock Frequency	External clock enable; PLL operating	48-0.25%	48+0.25%	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}=200pF (10/%~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

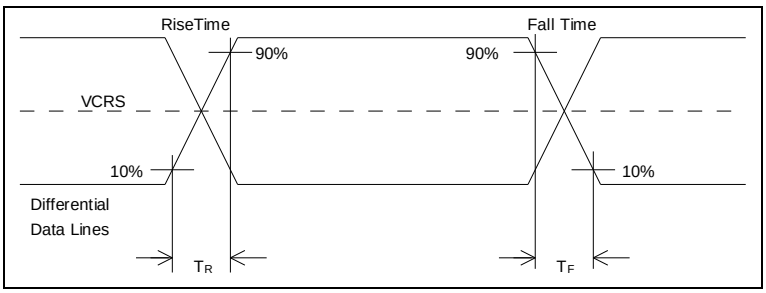


Figure 21-2 USB Data Signal Rise and Fall Time

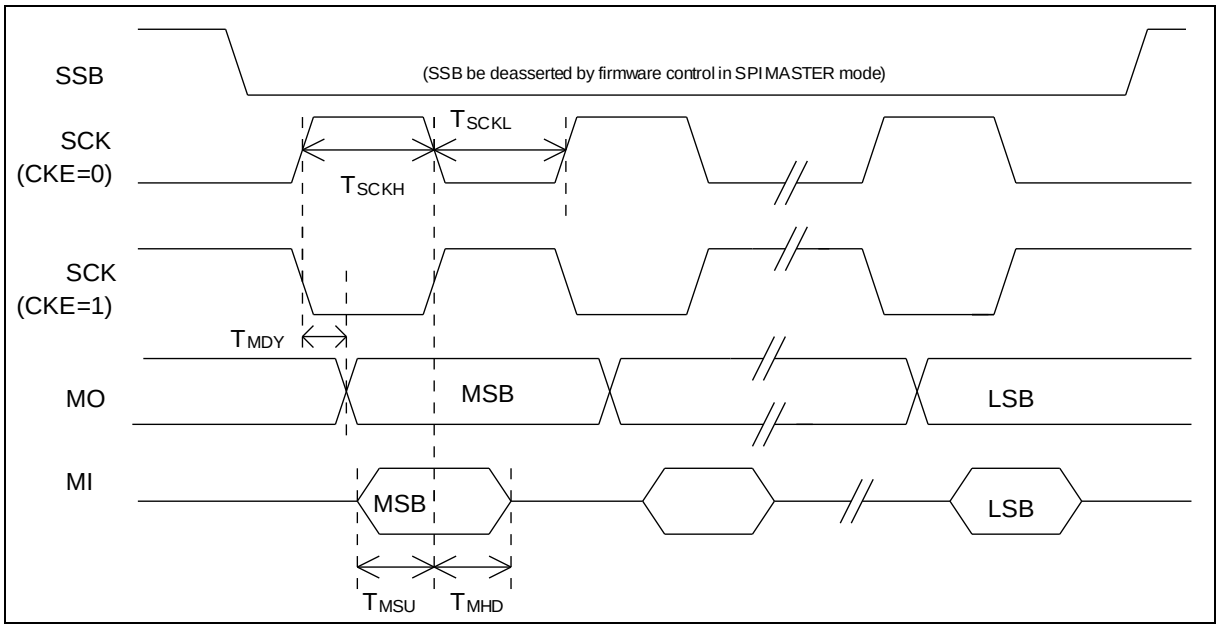


Figure 21-3 SPI Master Timing

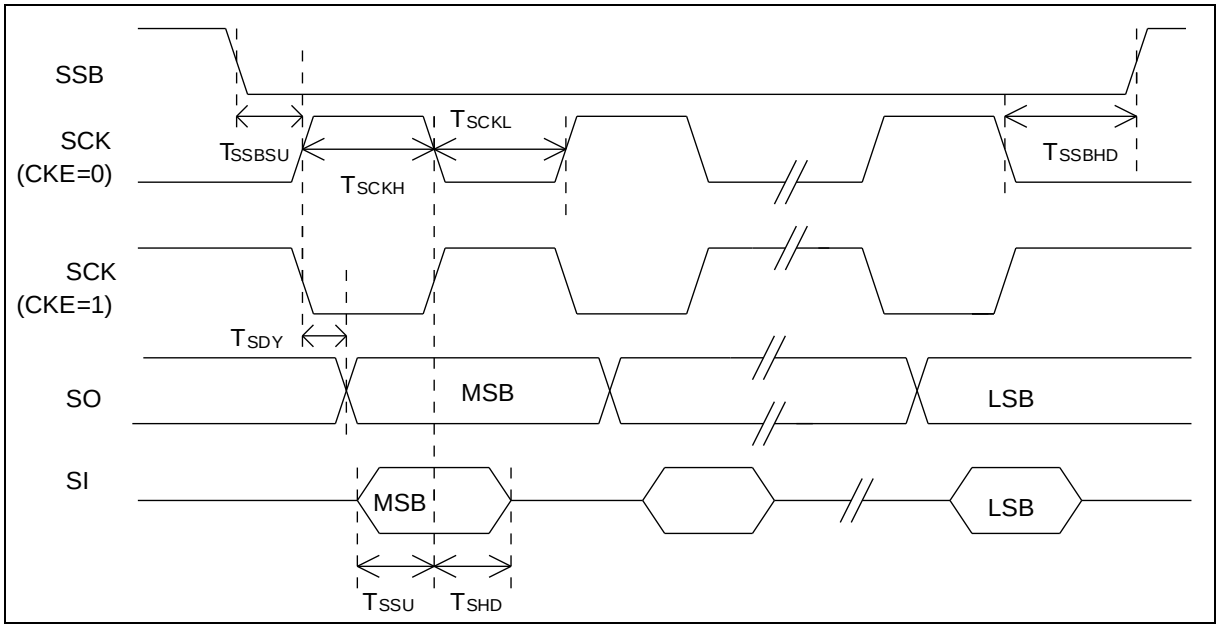
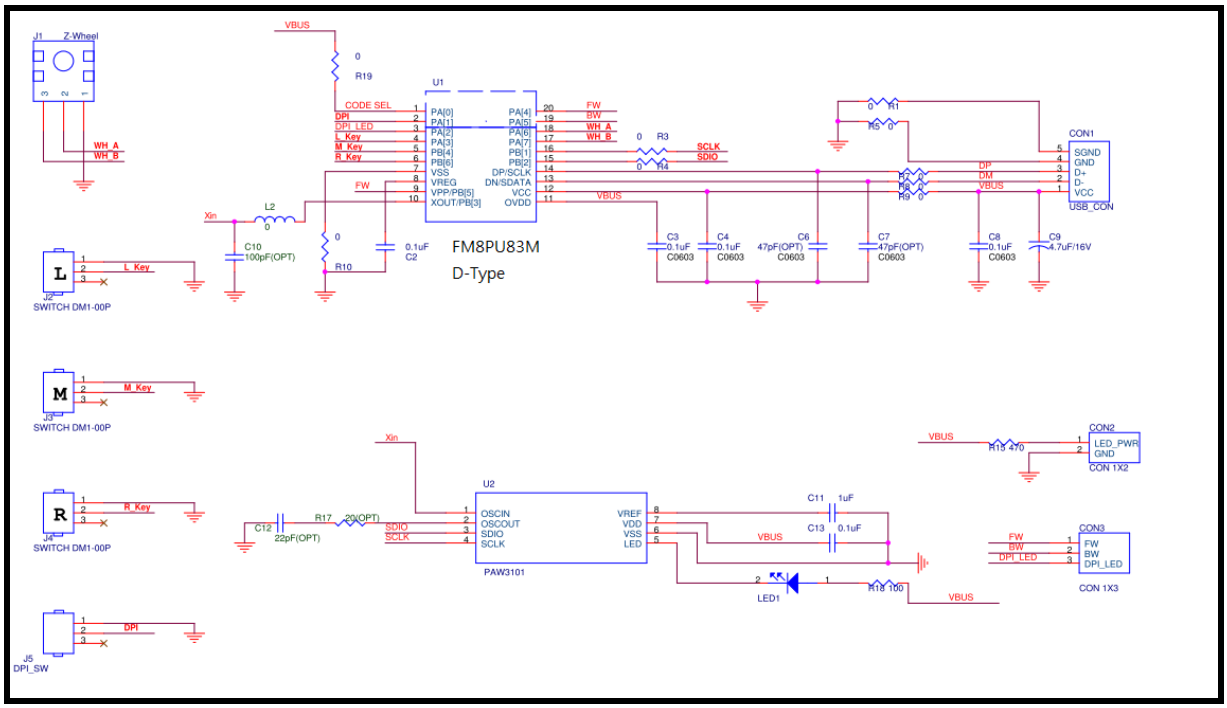


Figure 21-4 SPI Slave Timing

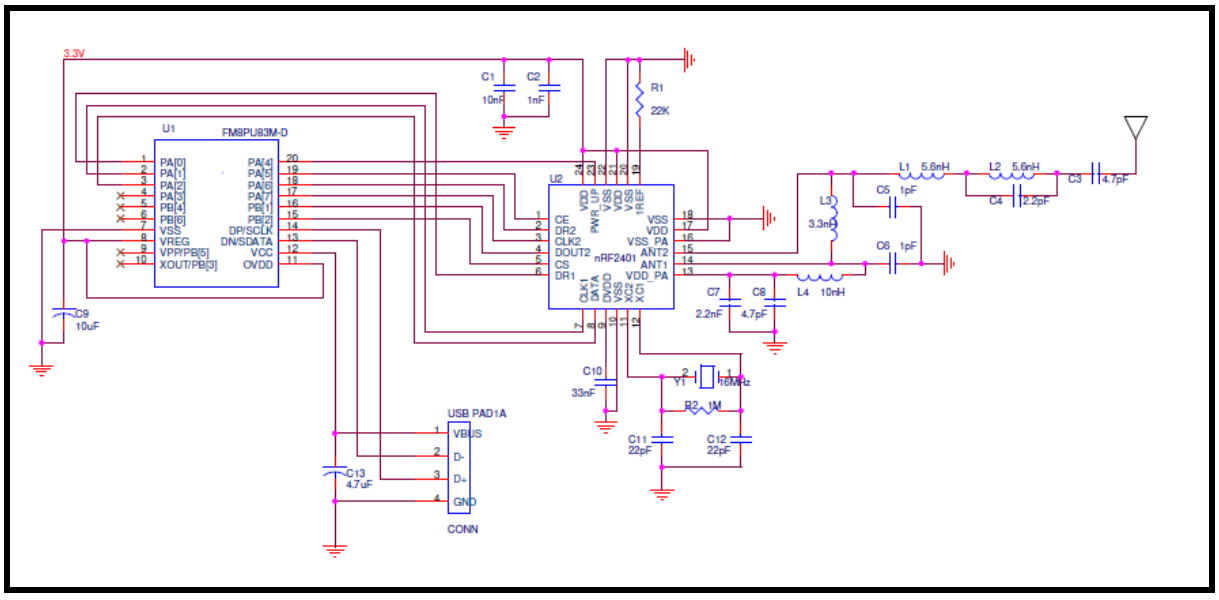
23.0 Application Circuit

23.1 FM8PU83M OVDD=5V Mechanical Z/2 3D3Key Application, with supply clock (12MHz).

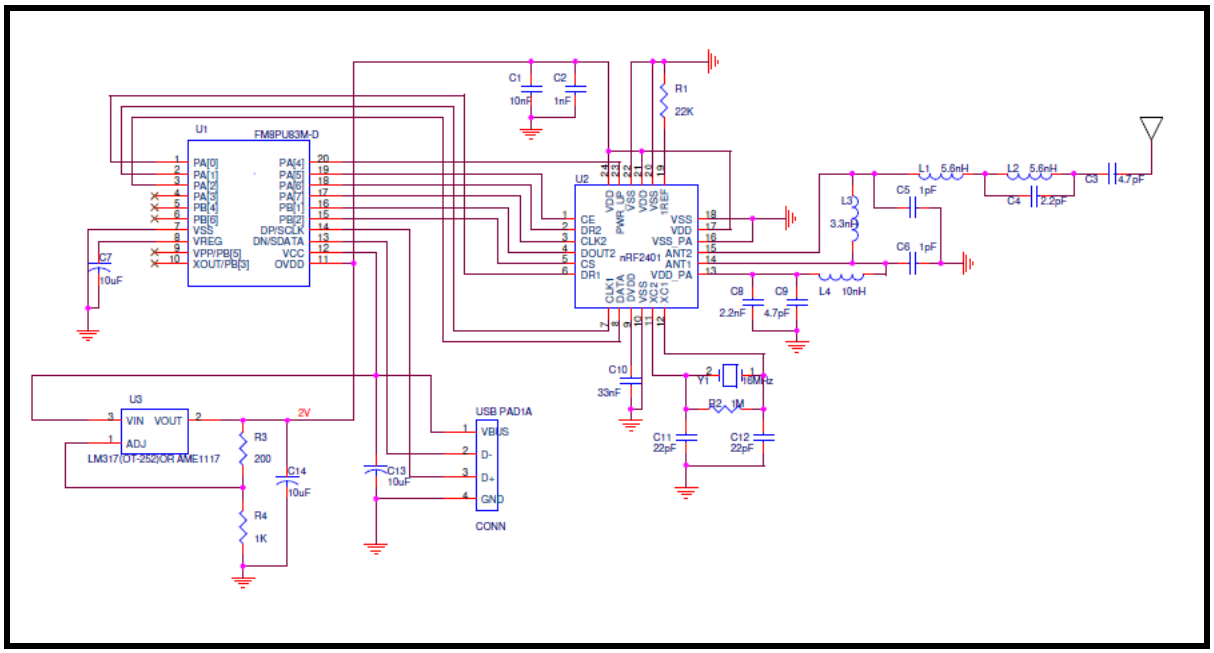


Note: Suggest VREG Capacitance used 1~10 uF.

23.2 FM8PU83M OVDD=VREG (3.3V) and 83M VREG supply RF Power

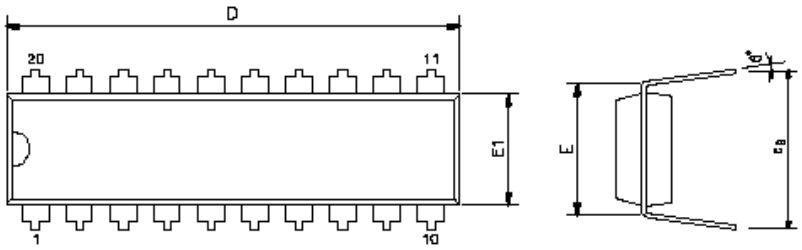


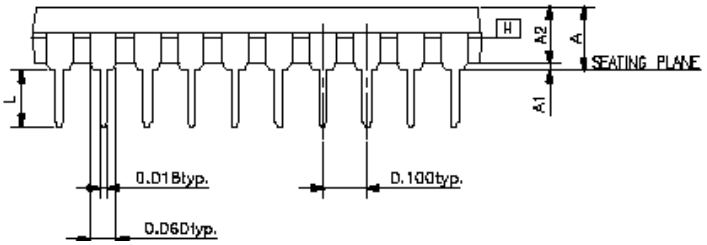
23.3 FM8PU83M used external Regulator to OVDD=2V.



24.0 Package Diagram

24.1 20- LEAD (300mil) DIP



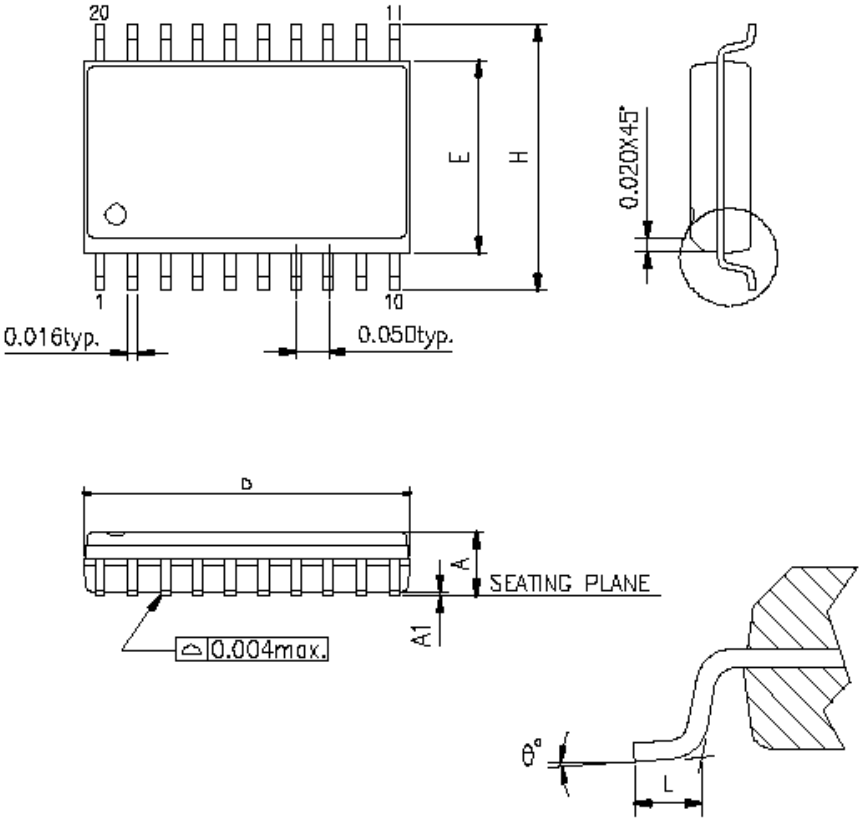


SYMBOLS	MIN.	NOR.	MAX.
A	–	–	0.210
A1	0.015	–	–
A2	0.125	0.130	0.135
D	0.98	1.030	1.060
E		0.300 BSC.	
E1	0.245	0.250	0.255
L	0.115	0.130	0.150
e_B	0.335	0.355	0.375
θ°	0	7	15

UNIT : INCH

NOTES:
1.JEDEC OUTLINE : MS-001 AD
2."D","E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS.MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 INCH.
3.eB IS MEASURED AT THE LEAD TIPS WITH THE LEADS UNCONSTRAINED.
4.POINTED OR ROUNDED LEAD TIPS ARE PREFERRED TO EASE INSERTION.
5.DISTANCE BETWEEN LEADS INCLUDING DAM BAR PROTRUSIONS TO BE .005 INCH MININUM.
6.DATUM PLANE [H] COINCIDENT WITH THE BOTTOM OF LEAD, WHERE LEAD EXITS BODY.

24.2 20- LEAD (300mil) SOP



SYMBOLS	MIN.	MAX.
A	0.093	0.104
A1	0.004	0.012
D	0.496	0.508
E	0.291	0.299
H	0.394	0.419
L	0.016	0.050
θ°	0	8

UNIT : INCH

NOTES:
1.JEDEC OUTLINE : MS-013 AC
2.DIMENSIONS "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED .15mm (.006in) PER SIDE.
3.DIMENSIONS "E" DOES NOT INCLUDE INTER-LEAD FLASH, OR PROTRUSIONS. INTER-LEAD FLASH AND PROTRUSIONS SHALL NOT EXCEED .25mm (.010in) PER SIDE.

Contents