

---

**EM78M680**

**USB Full Speed  
Microcontroller**

**Product  
Specification**

**DOC. VERSION 1.6**

**ELAN MICROELECTRONICS CORP.**

December 2011

---



---

**Trademark Acknowledgments:**

IBM is a registered trademark and PS/2 is a trademark of IBM.

Windows is a trademark of Microsoft Corporation.

ELAN and ELAN logo  are trademarks of ELAN Microelectronics Corporation.

Copyright © 2006~2011 by ELAN Microelectronics Corporation

**All Rights Reserved**

Printed in Taiwan

The contents of this specification are subject to change without further notice. ELAN Microelectronics assumes no responsibility concerning the accuracy, adequacy, or completeness of this specification. ELAN Microelectronics makes no commitment to update, or to keep current the information and material contained in this specification. Such information and material may change to conform to each confirmed order.

In no event shall ELAN Microelectronics be made responsible for any claims attributed to errors, omissions, or other inaccuracies in the information or material contained in this specification. ELAN Microelectronics shall not be liable for direct, indirect, special incidental, or consequential damages arising from the use of such information or material.

The software (if any) described in this specification is furnished under a license or nondisclosure agreement, and may be used or copied only in accordance with the terms of such agreement.

ELAN Microelectronics products are not intended for use in life support appliances, devices, or systems. Use of ELAN Microelectronics product in such applications is not supported and is prohibited.

NO PART OF THIS SPECIFICATION MAY BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS WITHOUT THE EXPRESSED WRITTEN PERMISSION OF ELAN MICROELECTRONICS.



---

**ELAN MICROELECTRONICS CORPORATION**

---

**Headquarters:**

No. 12, Innovation 1<sup>st</sup> Road  
Hsinchu Science Park  
Hsinchu, TAIWAN 30076  
Tel: +886 3 563-9977  
Fax: +886 3 563-9966  
[webmaster@emc.com.tw](mailto:webmaster@emc.com.tw)  
<http://www.emc.com.tw>

**Hong Kong:**

**Elan (HK) Microelectronics Corporation, Ltd.**  
Flat A, 19F., World Tech Centre 95  
How Ming Street, Kwun Tong  
Kowloon, HONG KONG  
Tel: +852 2723-3376  
Fax: +852 2723-7780

**USA:**

**Elan Information Technology Group (U.S.A.)**  
PO Box 601  
Cupertino, CA 95015  
U.S.A.  
Tel: +1 408 366-8225  
Fax: +1 408 366-8225

**Korea:**

**Elan Korea Electronics Company, Ltd.**  
301 Dong-A Building  
632 Kojan-Dong, Namdong-ku  
Incheon City, KOREA  
Tel: +82 32 814-7730  
Fax: +82 32 813-7730

**Shenzhen:**

**Elan Microelectronics Shenzhen, Ltd.**  
3F, SSMEC Bldg., Gaoxin S. Ave. I  
Shenzhen Hi-tech Industrial Park (South Area), Shenzhen  
CHINA 518057  
Tel: +86 755 2601-0565  
Fax: +86 755 2601-0500  
[elan-sz@elanic.com.cn](mailto:elan-sz@elanic.com.cn)

**Shanghai:**

**Elan Microelectronics Shanghai, Ltd.**  
3F, Building #13  
No. 116, Lane 572, Bibo Road  
Zhangjiang Hi-Tech Park  
Shanghai, CHINA 201203  
Tel: +86 21 5080-3866  
Fax: +86 21 5080-4600  
[elan-sh@elanic.com.cn](mailto:elan-sh@elanic.com.cn)

---

# Contents

<b>1</b>	<b>General Description .....</b>	<b>1</b>
<b>2</b>	<b>Features .....</b>	<b>1</b>
<b>3</b>	<b>Pin Assignment .....</b>	<b>4</b>
3.1	Pin Configuration .....	4
<b>4</b>	<b>Pin Description .....</b>	<b>5</b>
<b>5</b>	<b>Block Diagram .....</b>	<b>6</b>
<b>6</b>	<b>Functional Description .....</b>	<b>7</b>
6.1	Program Memory .....	7
6.2	Data Memory .....	8
6.2.1	Operational Registers .....	9
6.3	Special Function Registers .....	20
6.3.1	A (Accumulator) .....	20
6.3.2	CONT (Control Register) .....	20
6.3.3	IOC5 ~ IOC9 Port Direction Control Registers .....	20
6.3.4	IOCA (RFCNT) .....	21
6.3.5	IOCB (PWM_CNT) .....	21
6.3.6	IOCC (SPIT1CON:SPI Timer 1 Control Register) .....	22
6.3.7	IOCD (Port 9 Pull-high Control Register) .....	22
6.3.8	IOCE (MCU Control Register) .....	22
6.3.9	IOCF (Interrupt Mask Register) .....	23
6.4	USB Device Controller .....	24
6.5	Device Address and Endpoints .....	25
6.6	Reset .....	25
6.6.1	Power-on Reset .....	25
6.6.2	Watchdog Reset .....	25
6.6.3	USB Reset .....	25
6.7	Saving Power Mode .....	26
6.7.1	Power-down Mode .....	26
6.7.2	Dual Clock Mode .....	26
6.8	Interrupt .....	27
6.9	Pattern Detect Application (PDA) .....	28
6.9.1	Functional Description .....	28
6.9.2	Control Register .....	28
6.9.3	Sampling Rate and Debounce Length .....	29



6.10	Pulse Width Modulation (PWM) .....	30
6.10.1	Functional Description.....	30
6.11	Analog-to-Digital Converter (ADC) .....	31
6.11.1	Functional Description.....	31
6.11.2	Control Register .....	31
6.12	Serial Peripheral Interface Mode .....	31
6.12.1	Overview .....	31
6.12.2	Features .....	31
6.12.3	SPI Functional Description.....	33
6.12.4	SPI Signal and Pin Description .....	35
6.12.5	SPI Control Registers.....	36
6.12.6	SPI Mode Timing.....	36
6.12.7	SPI Timer 1 Mode.....	38
<b>7</b>	<b>Absolute Maximum Ratings .....</b>	<b>39</b>
<b>8</b>	<b>DC Electrical Characteristics .....</b>	<b>40</b>

## APPENDIX

<b>A</b>	<b>Special Register Map .....</b>	<b>41</b>
<b>B</b>	<b>Instruction Set .....</b>	<b>47</b>
<b>C</b>	<b>Code Option.....</b>	<b>49</b>
<b>D</b>	<b>Application Circuits .....</b>	<b>52</b>

### Specification Revision History

Doc. Version	Revision Description	Date
0.9	Preliminary version	2006/08/29
1.0	Initial released version	2006/12/12
1.1	Updated the code option bit and definition.	2007/12/28
1.2	Revised as per customer request (not officially released)	–
1.3	Modified the application circuits and regulator output current limit..	2008/03/04
1.4	Added EM78M680 SPI function.	2009/03/23
1.5	Updated the SPI description and corrected the typographical error.	2011/01/20
1.6	Reduced the number of package types	2011/12/08

---

## 1 General Description

The EM78M680 is a series of 8-bit Universal Serial Bus, RISC architecture, Multi-Time Programming (MTP) microcontrollers. It is specifically designed for USB full speed device application. The EM78M680 also supports one device address and six endpoints.

The EM78M680 has 8-level stacks and four sets of interrupt sources. It has a maximum of 36 General Input/Output pins with the capacity of sinking large current. Each device has 271 bytes of General Purpose SRAM, 6K bytes of program ROM, and is embedded with 32 bytes of EEPROM.

These series of ICs have special features that meet user's requirements. Such features are:

- Dual Clock mode which allows the device to run on very low power.
- Pattern Detect Application function which is used in serial transmission to count waveform width.
- Pulse Width Modulation that can generate a duty-cycle-programmable signal.
- 24-channel AD converter with up to 10 bits resolution
- Serial Peripheral Interface

---

## 2 Features

- Operating voltage: 4.4V ~ 5.25V
- USB Specification Compliance
  - Universal Serial Bus Specification Version 1.1
  - USB Device Class Definition for Human Interface Device (HID), Firmware Specification Version 1.1
  - Supports one device address and six endpoints
- USB Application
  - P74 (D+) has an internal pull-high resistor (1.5K $\Omega$ )
  - Firmware programmable D+ pull-high resistor
  - USB protocol handling
  - USB device state handling
  - Identifies and decodes Standard USB commands to Endpoint Zero
- Built-in 8-bit RISC MCU
  - 8-level stacks for subroutine nesting and interrupt
  - Auto saving R3 and R4 when interrupt occurs
  - 8-bit real time clock/counter (TCC) with overflow interrupt

- Built-in RC oscillator free running for Watchdog Timer and Dual clock mode
- Two independent programmable prescalers for WDT and TCC
- Two methods of power saving:
  - Power-down mode (Sleep mode)
  - Dual clock mode
- Two clocks per instruction cycle
- Multi-time programmable
- Four sets of interrupts
  - Set 1 INT : (Jump to 0x08)
    - ◆ TCC overflow interrupt
    - ◆ EP0 command in interrupt
    - ◆ USB suspend interrupt
    - ◆ USB reset interrupt
    - ◆ USB Host resume interrupt
  - Set 2 INT : (Jump to 0x10)
    - ◆ RF1 low pattern interrupt
    - ◆ RF1 high pattern interrupt
    - ◆ RF2 low pattern interrupt
    - ◆ RF2 high pattern interrupt
  - Set 3 INT : (Jump to 0x18)
    - ◆ P77 port change interrupt
    - ◆ P76 port change interrupt
    - ◆ SPI Timer 1 comparator completed interrupt
    - ◆ SPI transmission completed interrupt
  - Set 4 INT : (Jump to 0x20)
    - ◆ EP1~5 output Endpoint received OK interrupt
- I/O Ports
  - 3 LED sink pins
  - Each GPIO pin in Ports 5, 6, 8, P90~P93, P95, P96, P70~P72 and P76~P77, has an internal programmable pull-high resistor (25 K $\Omega$ )
  - Each GPIO pin of Port 6, Port 90~93 can wakeup the MCU from sleep mode by input state change
- Internal Memory
  - Built-in 6K $\times$ 13 bits Program ROM
  - Built-in 271 bytes general purpose registers (SRAM)
  - Built-in USB Application FIFOs
  - Built-in 32 bytes E2PROM



- Operation Frequency
- Normal Mode: the MCU runs on an external oscillator frequency of 4 MHz, Internal system frequency of 8 MHz, 16 MHz or 24 MHz
  - Dual Clock Mode: the MCU runs at a frequency of 256kHz (or 32kHz, 4kHz, 500Hz), using an internal oscillator with an external crystal resonator turned off to save power
- Built-in Pattern Detecting Application for serial signal transmission
- Built-in Pulse Width Modulation (PWM)
  - 2 channels PWM function on Pin 92 (PWM1) and Pin 93 (PWM2)
  - 8-bit resolution of PWM output
  - 8 selections of duty cycles
- Built-in 24-Channel Analog-to-Digital Converter (ADC)
  - Built-in AD Converter with 10-bit resolution
  - 4 types of ADC clock source selection: 256K/128K/64K/32K
- Built-in Serial Peripheral Interface (SPI)
  - Built-in two FIFO for SPI read/write data
  - Operation in either Master mode or Slave mode
  - Programmable baud rates of communication
  - Up to 8 MHz bit frequency
- Built-in 3.3V Voltage Regulator
  - For UDC power supply
  - Pull-up source for the external USB resistor on D+ pin
- Package Type
  - 44-pin QFP : EM78M680D AQ
  - 24-pin SOP 300 mil : EM78M680D CM

### 3 Pin Assignment

#### 3.1 Pin Configuration

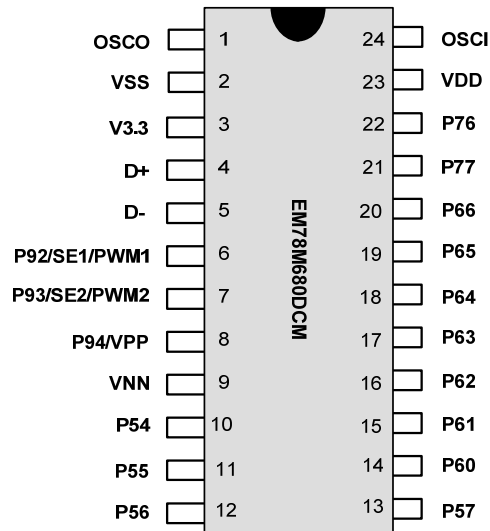


Figure 3-1 EM78M680DCM (24-pin SOP)

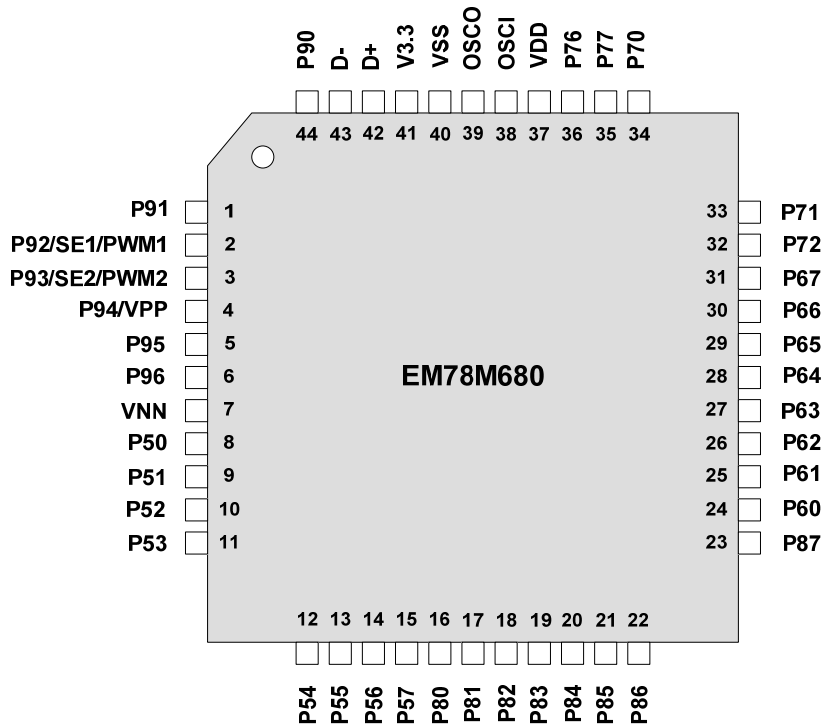


Figure 3-2 EM78M680DAQ (44-Pin QFP)

## 4 Pin Description

Symbol	I/O	Function
P50 ~ P57	I/O	General 8-bit bidirectional input/output port. All pins on this port can be internally pulled-high by register IOCE Bit 0. All Port 5 input/output pins can be used for ADC function. P57 : SPI/SS bit P56 : SCK P55 : SDO P54 : SDI
P60 ~ P67	I/O	General 8-bit bidirectional input/output port. All pins on this port can be internally pulled-high by register IOCE Bit 1. All Port 6 input/output pins can be used for ADC function.
P70 ~ P72 P76 ~ P77	I/O	General 8-bit bidirectional input/output port. All pins on this port can be internally pulled-high by register IOCE Bit 3. The sink current of P70 ~ P72 are used for driving LED.
P80 ~ P87	I/O	General 8-bit bidirectional input/output port. All pins on this port can be internally pulled-high by register IOCE Bit 2. All Port 8 input/output pins can be used for ADC function.
P94 / VPP	I	MTP program pin. Used in programming the on-chip ROM. P94 functions as an input pin only (without an internally pulled-high resistor).
P90 ~ P93 P95 ~ P96	I/O	General bidirectional input/output port. Each pin can be internally pulled-high by register IOCD. P92 ~ P93 can be used for PWM (pulse width modulation) or PDA (serial signal transmission application) function.
USB D+	I/O	USB D+ pin. Built-in internal 1.5K pulled-high resistor to V3.3
USB D-	I/O	USB D- pin
OSCI	I	4 MHz crystal input
OSCO	I/O	Return path for 4 MHz crystal resonator.
VNN	-	MTP program pin. Used in programming the on-chip ROM. During normal operation, this pin is connected to Ground.
V3.3	O	3.3V DC voltage output from an internal regulator. This pin has to be tied to a 4.7 $\mu$ F capacitor.
VDD	-	Connect to the USB power source or to a nominal 5V-power supply. Actual VDD range can vary between 4.4V and 5.25V.
VSS	-	Connect to ground



## 5 Block Diagram

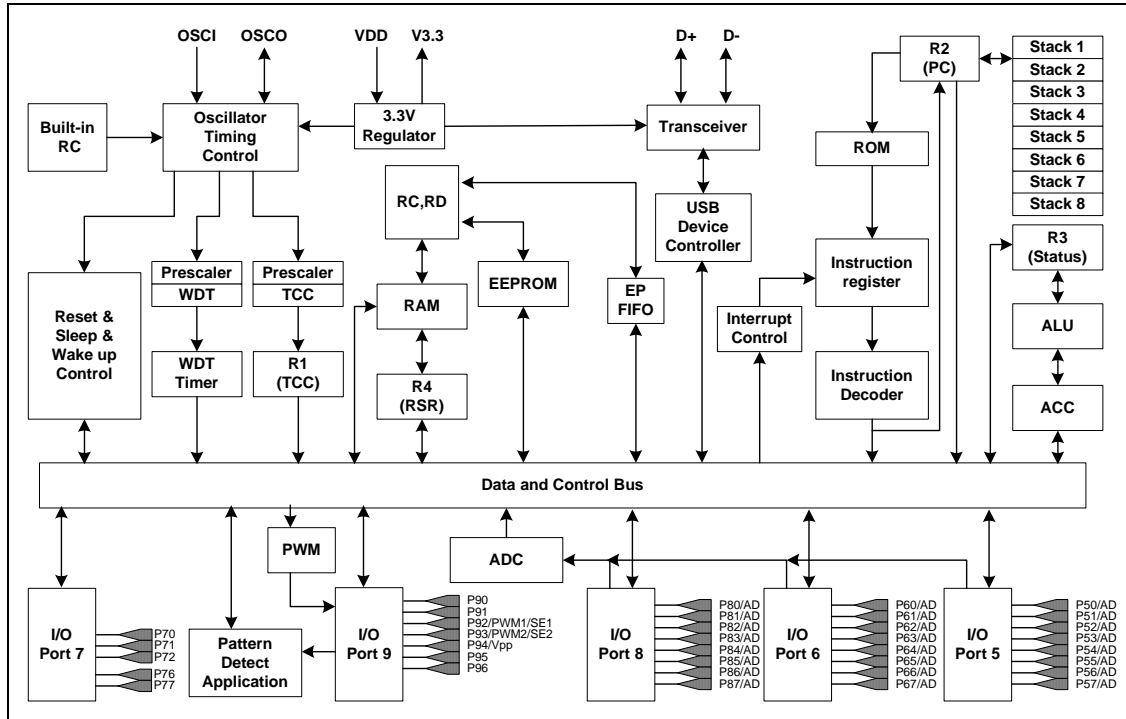


Figure 5-1 Functional Block Diagram

## 6 Functional Description

The EM78M680 memory is organized into four spaces, namely; User Program memory in 6K×13 bits ROM space, Data Memory in 271 bytes SRAM space, EEPROM space, and USB Application FIFOs for Endpoint 0, Endpoint 1, Endpoint 2, Endpoint 3, Endpoint 4, and Endpoint 5. Furthermore, several registers are used for special purposes.

### 6.1 Program Memory

The program space of the EM78M680 is 6K bytes, and is divided into six pages. Each page is 1K byte long. After a Reset, the 13-bit Program Counter (PC) points to location zero of the program space.

The Interrupt Vectors are at 0x0008 (USB and TCC interrupt), 0x0010 (RF function interrupt), 0x0018 (P76 P77 port change and SPI interrupt) and 0x0020 (EP1~5 output endpoint interrupt).

When interrupt occurs, the MCU will auto save to the Status Register (R3), RAM Select Register (R4 Bits 6, 7), Accumulator (A) then clear PS0~PS2 and fetch the instruction from the interrupt vector address. The interrupt vector address is illustrated in the following diagram.

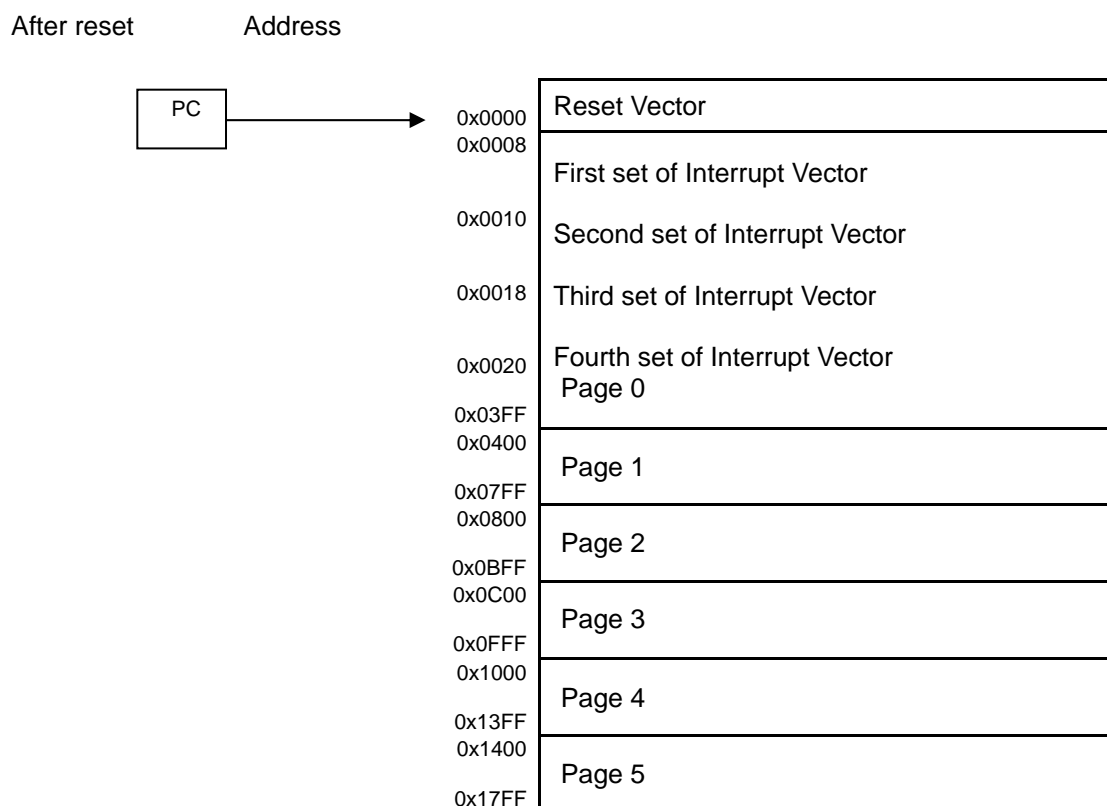


Figure 6-1 EM78M680 Data RAM Organization



## 6.2 Data Memory

The Data Memory has 271 bytes SRAM space. It is also equipped with FIFO space for USB, EEPROM and SPI application. Figure 6-2 shows the organization of the Data Memory Space.

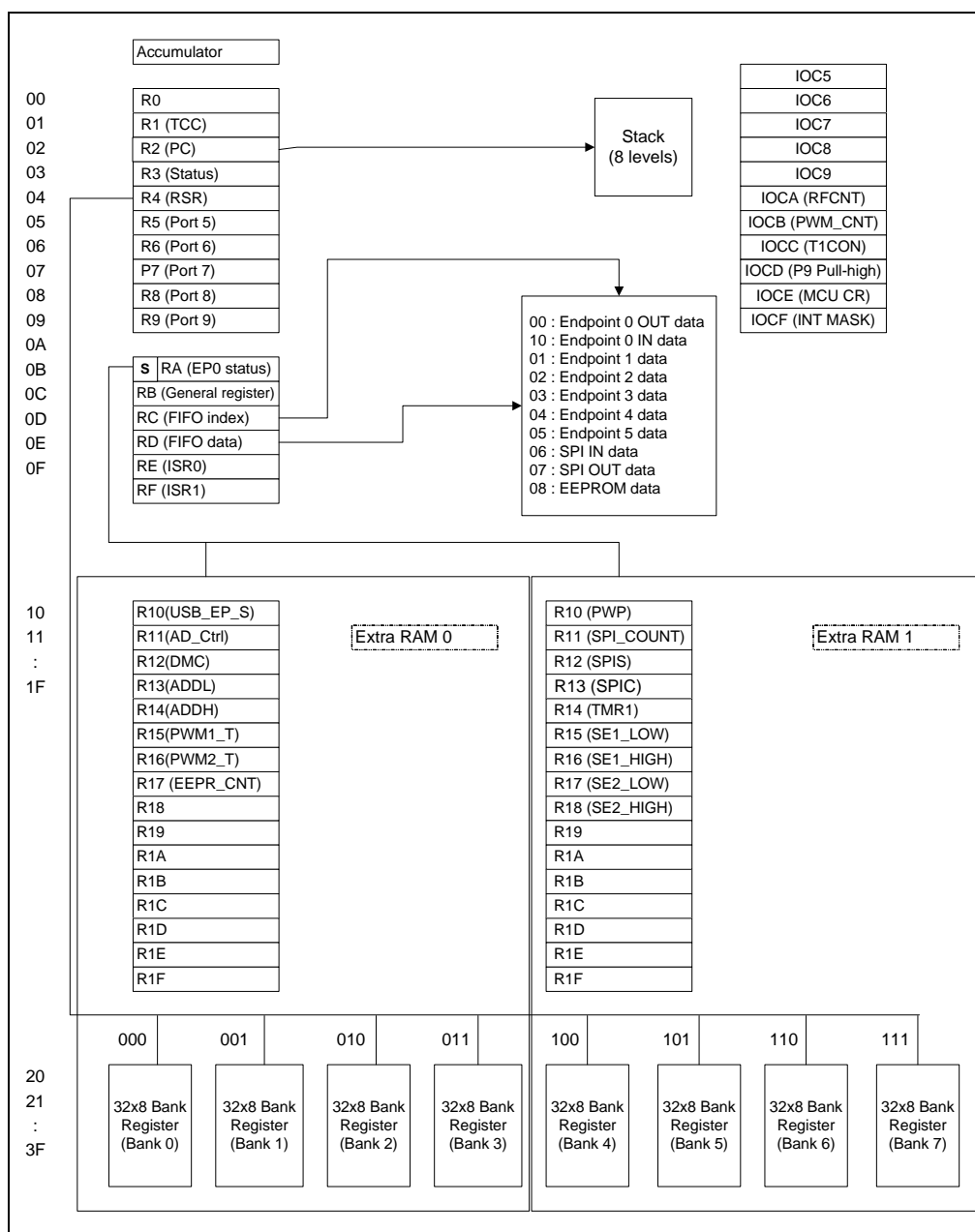


Figure 6-2 Data Memory Configuration

## 6.2.1 Operational Registers

### 6.2.1.1 R0 (Indirect Addressing Register)

R0 is not a physically implemented register. It is used as an indirect addressing pointer. Any instruction using R0 as register actually accesses data pointed by the RAM Select Register (R4).

### 6.2.1.2 R1 (Timer/Clock Counter, TCC)

This TCC register is an 8-bit timer or counter. It is incremented by the instruction cycle clock, and is readable and writable as any other register.

### 6.2.1.3 R2 (Program Counter and Stack)

- R2 and the hardware stacks are 13 bits wide.
- The structure is depicted in Figure 3.
- Generates 6K×13 bit on-chip ROM addresses to the relative programming instruction codes. One program page is 1024 words long.
- All the R2 bits are set to "0"s when a reset condition occurs.
- "JMP" instruction allows direct loading of the lower 10 program counter bits. Thus, "JMP" allows jump to any location on one page.
- "CALL" instruction loads the lower 10 bits of the PC, and then PC+1 is pushed onto the stack. Thus, the subroutine entry address can be located anywhere within a page
- "RET" ("RETL k", "RETI") instruction loads the program counter with the contents at the top of stack.
- "MOV R2, A" allows the loading of an address from the "A" register to the lower 8 bits of PC, and the ninth and tenth bits (A8~A9) of PC are cleared.
- "ADD R2, A" allows a relative address to be added to the current PC, and the ninth and tenth bits of PC are cleared.
- Any instruction that is written to R2 (e.g. "ADD R2, A", "MOV R2, A", "BC R2,6", etc.), except "TBL" will cause the ninth and tenth bits (A8~A9) of the PC to be cleared. Thus, the computed jump is limited to the first 256 locations of any program page.
- "TBL" allows a relative address to be added to the current PC ( $R2+A \rightarrow R2$ ), and contents of the ninth and tenth bits (A8~A9) of the PC are not changed. Thus, the computed jump can be on the second (third, or 4th) 256 locations on one program page.
- For the EM78M680, the most significant bits (A10~A12) will be loaded with the contents of bits PS0~PS2 in the status register (R3) upon the execution of a "JMP", "CALL", or any other instructions which writes to R2.
- All instructions are single instruction cycle except for the instruction that will change the contents of R2. Such instruction will need one more instruction cycle.

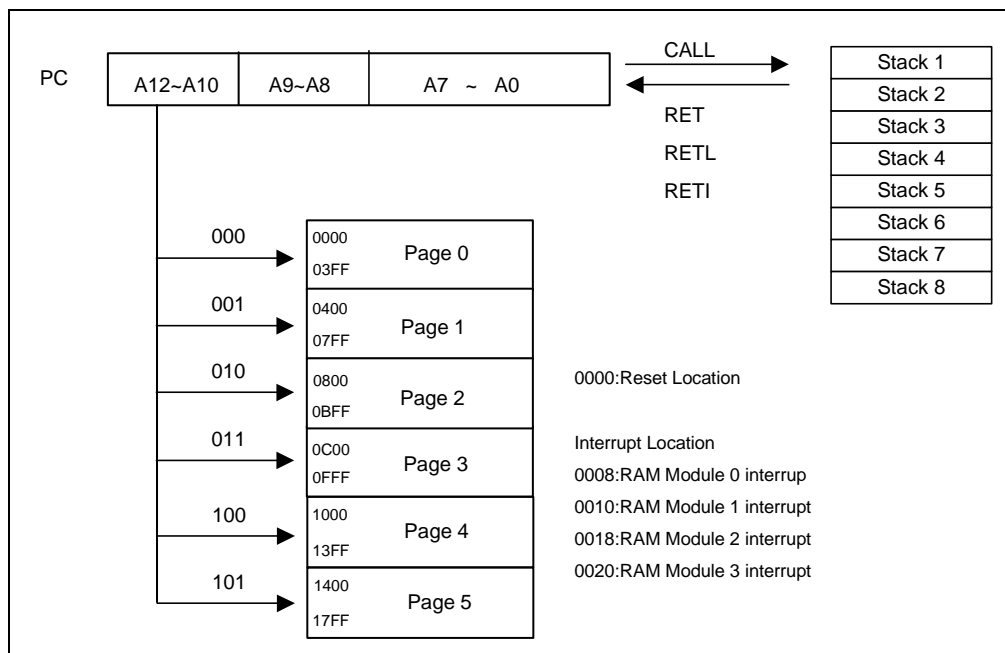


Figure 6-3 Program Counter and Stack

**6.2.1.4 R3 (SR)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PS2	PS1	PS0	T	P	Z	DC	C

- R3 [0] Carry flag
- R3 [1] Auxiliary carry flag
- R3 [2] Zero flag. It will be set to 1 when the result of an arithmetic or logic operation is 0.
- R3 [3] Power down flag. It will be set to “1” during Power-on phase or by “WDTC” command and cleared when the MCU enters into Power-down mode. It remains in its previous state after a Watchdog Reset.  
 “0”: Power-down mode  
 “1”: Power on

**Status of T and P Being Affected by Events**

Reset Type	T	P
Power on	1	1
WDT during Operation mode	0	P
WDT Wake-up during Sleep 1 mode	0	0
WDT Wake-up during Sleep 2 mode	0	P
Wake-up on pin change during Sleep 2 mode	P	P
WDTC instruction	1	1
WDT time-out	0	*P
SLEP instruction	1	0

\*P: Previous value before reset

R3 [4] Time-out Flag. It will be set to “1” during Power-on phase or by “WDTC” command. It is reset to “0” by WDT time-out.

“0”: Watchdog timer with overflow

“1”: Watchdog timer without overflow

R3 [5~7] Page Select Bits. These three bits are used to select the page of the program memory.

PS2	PS1	PS0	Program Memory Page [Address]
0	0	0	Page 0 [000-3FF]
0	0	1	Page 1 [400-7FF]
0	1	0	Page 2 [800-BFF]
0	1	1	Page 3 [C00-FFF]
1	0	0	Page 4 [1000-13FF]
1	0	1	Page 5 [1400-17FF]

#### 6.2.1.5 R4 (RSR)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BK1	BK0	Ad5	Ad4	Ad3	Ad2	Ad1	Ad0

R4 [0~5] used to select registers (Address: 0x00h~0x3Fh) in indirect addressing mode.

R4 [6~7] used to determine which bank is activated among the 8 banks. To select a register bank, refer to the table below:

R4[7]BK1	R4[6]BK0	RAM Bank #
0	0	Bank 0
0	1	Bank 1
1	0	Bank 2
1	1	Bank 3

#### 6.2.1.6 R5 (P5) Default Value: (0B\_0000\_0000)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P57	P56	P55	P54	P53	P52	P51	P50

#### 6.2.1.7 R6 (P6) Default Value: (0B\_0000\_0000)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P67	P66	P65	P64	P63	P62	P61	P60

#### 6.2.1.8 R7 (P7) Default Value: (0B\_0000\_0000)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P77	P76	D-	D+	-	P72	P71	P70

**6.2.1.9 R8 (P8) Default Value: (0B\_0000\_0000)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P87	P86	P85	P84	P83	P82	P81	P80

**6.2.1.10 R9 (P9) Default Value: (0B\_0000\_0000)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	P96	P95	P94	P93	P92	P91	P90

**6.2.1.11 RA (USBES): Default Value: (0B0000\_0000)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Extr_R	Remote Status	EP0_W	EP0_R	Dev_Resume	UDC_Suspend	UDC_Writing	STALL

- RA [0] Stall Flag. When the MCU receives an unsupported command or invalid parameters from host, this bit will be set to 1 by the firmware to notify the UDC to return a Stall handshake. When a successful setup transaction is received, this bit is cleared automatically. This bit is both readable and writable.
- RA [1] UDC Writing Flag. Read only. When this bit is equal to “1,” it indicates that the UDC is writing data into the EP0’s FIFO or reading data from it. During this time, user should avoid accessing the FIFO until the UDC finishes writing or reading. This bit is readable only.
- 1: EP0’s FIFO is busy  
0: EP0’s FIFO is free for data transition. ACK, NAK are reset.
- RA [2] UDC Suspend Flag. If this bit is equal to “1”, it indicates that the USB bus has no traffic for a specified period of 3.0 ms. This bit will also be cleared automatically when a bus activity takes place. This bit is readable only.
- RA [3] Device Resume Flag. This bit is set by firmware to generate a signal to wake up the USB host and is cleared as soon as the USB Suspend signal becomes low. This bit can only be set by firmware and cleared by hardware. It can only be used under dual mode. This bit is both readable and writable.
- RA [4] EP0\_R flag. This bit informs the UDC to read the data written by the firmware from the FIFO. Then the UDC will automatically send the data to the Host. After the UDC finishes reading the data from the FIFO, this bit will be cleared automatically.
- Before writing data into FIFO, the firmware will first check this bit to avoid overwriting the data. This bit can only be set by the firmware and cleared by the hardware.
- RA [5] EP0\_W. After the UDC completes writing data to the FIFO, this bit will be set automatically. The firmware will clear it as soon as it gets the data from EP0’s FIFO. Only when this bit is cleared will the UDC be able to write a new data into the FIFO.

RA [6] Remote Wake-up status. Pass device remote wake-up setting from the PC.

RA [7] Extra RAM Switch. RAM block switch

**0:** Switch to Bank 0~Bank 3 and external RAM 0

**1:** Switch to Bank 4~Bank 7 and external RAM 1

**6.2.1.12 RC (FIFO\_Index) Default Value: (0B\_0000\_0000)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	UAD4	UAD3	UAD2	UAD1	UAD0

RC [0~4] Application FIFO Address Registers. These five bits are the address pointers of Application FIFO.

RC [5~7] Undefined registers.

**6.2.1.13 RD (FIFO\_Data) Default Value: (0B\_XXXX\_XXXX)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
UAD7	UAD6	UAD5	UAD4	UAD3	UAD2	UAD1	UAD0

RD (Application FIFO data register) contains the data in the register of which address is pointed by RC.

**6.2.1.14 RE (ISR0) Default Value: (0B\_0000\_0000)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P77_IF	P76_IF	TM1IF	SPIIF	RF2_High	RF2_Low	RF1_High	RF1_Low

RE [0] RF1\_Low flag: Pattern Detect Interrupt Flag. RF1 low pattern compare flag.

RE [1] RF1\_High flag: Pattern Detect Interrupt Flag. RF1 high pattern compare flag.

RE [2] RF2\_Low flag: Pattern Detect Interrupt Flag. RF2 low pattern compare flag.

RE [3] RF2\_High flag: Pattern Detect Interrupt Flag. RF2 high pattern compare flag.

RE [4] SPI interrupt flag. Set by data transmission complete, flag is cleared by software.

RE [5] SPI Timer 1 Interrupt Flag. Set by the comparator at Timer 1 application, flag is cleared by software.

RE [6] P76\_IF: P76 State Change Interrupt Flag.

RE [7] P77\_IF: P77 State Change Interrupt Flag. The interrupt vector is in 0x0018 addresses.

**6.2.1.15 RF (ISR1) Default Value: (0B\_0000\_0000)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OUT EP_IF	-	-	H_Resume_IF	USBReset_IF	Suspend_IF	EP0_IF	TCC_IF

RF [0] TCC Timer Overflow Interrupt Flag. It will be set while the TCC timer overflows, and is cleared by the firmware.



- RF [1] EndPoint Zero Interrupt Flag. It will be set when the EM78M680 receives Vendor/Customer Command to Endpoint Zero. This bit is cleared by the firmware.
- RF [2] USB Suspend Interrupt Flag: It will be set when the EM78M680 finds the USB Suspend Signal on the USB bus. This bit is cleared by the firmware.
- RF [3] USB Reset Interrupt Flag. It will be set when the Host issues the USB Reset signal.
- RF [4] USB Host Resume Interrupt Flag. It is set only when MCU under Dual clock mode and USB suspend signal becomes low.
- RF [5~6] Not used and read as "0".
- RF [7] OUT Endpoint Interrupt Flag. It will be set when the FIFO of Out Endpoint has received data from host.

**Extra RAM 0:**

**6.2.1.16 R10 (USB\_EP\_S): Default (0b0000\_0000)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	EP5_ST	EP4_ST	EP3_ST	EP2_ST	EP1_ST

R10 [0~4] EPx\_ST: End point state flag.

For IN Endpoint: Set to "1" by firmware when buffer is Ok. These bits are reset after the UDC sends data to Host and gets ACK from Host. Before writing data into FIFO, user must check these bits to prevent overwriting the data. These are readable and writable.

For Out Endpoint: Set to 1 by UDC when buffer is Ok. Set to "0" by firmware when buffer is read.

After an out token finish, and the UDC completes writing data to the FIFO, this bit will be set automatically, and run into Interrupt Vector 0x0020. The firmware should clear it as soon as it gets the data from OUT Endpoint's FIFO. Only when this bit is cleared, will the UDC be able to write a new data into the FIFO.

**6.2.1.17 R11 (AD\_Ctrl): Default (0b0001\_1111)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AD_start	AD_R1	AD_R0	AD_A4	AD_A3	AD_A2	AD_A1	AD_A0

R11 [0~4] AD Channel Selector: If the AD number is from 0 to 0x17, the AD converter will be powered on. Otherwise, it will be powered off.



AD4	AD3	AD2	AD1	AD0	Channel	I/O Port
0	0	0	0	0	0	P50
0	0	0	0	1	1	P51
0	0	0	1	0	2	P52
0	0	0	1	1	3	P53
0	0	1	0	0	4	P54
0	0	1	0	1	5	P55
0	0	1	1	0	6	P56
0	0	1	1	1	7	P57
0	1	0	0	0	8	P80
0	1	0	0	1	9	P81
0	1	0	1	0	10	P82
0	1	0	1	1	11	P83
0	1	1	0	0	12	P84
0	1	1	0	1	13	P85
0	1	1	1	0	14	P86
0	1	1	1	1	15	P87
1	0	0	0	0	16	P60
1	0	0	0	1	17	P61
1	0	0	1	0	18	P62
1	0	0	1	1	19	P63
1	0	1	0	0	20	P64
1	0	1	0	1	21	P65
1	0	1	1	0	22	P66
1	0	1	1	1	23	P67

R11 [5~6] AD conversion clock source

**00:** 256K

**01:** 128K

**10:** 64K

**11:** 32K

R11 [7] AD Converter ready flag

**1:** Start ADC Conversion. This bit can be set by software.

**0:** Reset on completion of the conversion. This bit cannot be reset by software.

**NOTE**

*The hardware can enable this function only at the AD Channel Selector of the functional I/O port. After Power-on reset, the initial value of this register is 0b0001 1111.*

**6.2.1.18 R12 (DMC) : Default (0b0000\_1000)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	USB_Token	LOW FR1	LOW FR0	/LOW FREQ	–	–	–

R12 [3] /LOW FREQ: Dual Clock Control bit. This bit is used to select the frequency of the system clock. When this bit is set to “0”, the MCU will run on very slow frequency for power saving purposes and the UDC will stop working.

**0:** Slow frequency (500Hz~256kHz)

**1:** Normal frequency

R12 [4~5] LOW FR0 ~ LOW FR1: Low Frequency Switches. These bits select the operation frequency in Dual Clock Mode. Four frequencies are available and can be chosen as Dual Clock Mode for running the MCU program.

Low FR1	Low FR0	Frequency
0	0	500Hz
0	1	4kHz
1	0	32kHz
1	1	256kHz

**Bit 6 (USB\_Token):** Set when USB Token from Host. Reset when end of the Token.

**6.2.1.19 R13 (ADDL): Default (0b0000\_0000). Read Only.**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADB1	ADB0	0	0	0	0	0	0

R14 (ADDH): Default (0b0000\_0000). Read Only.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADB9	ADB8	ADB7	ADB6	ADB5	ADB4	ADB3	ADB2

R15(PWM1\_T): The high level time of the 1st PWM module that outputs to P92.

R16 (PWM2\_T): The high level time of the 2nd PWM module that outputs to P93.

R15~R16 can be general purpose registers if the PWM function is disabled.

**6.2.1.20 R17 (EECNT): Default (0b0000\_0011)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	EE_Act	EE_Reset	EE_A4	EE_A3	EE_O.K	EE_C1	EE_C0

R17 [0~1] EE\_C0 ~ EE\_C1: EEPROM control bits

**00:** Read data from EEPROM to EEPROM FIFO.

**01:** Write data from EEPROM FIFO to EEPROM

**10:** Erase EEPROM

**11:** Disable EEPROM function

R17 [2] EE\_O.K: EEPROM activated OK bits

0: not OK

1: OK

R17 [3~4] EE\_A4 ~ EE\_A3: Bank selector. The EEPROM is divided into four banks, these two bits can select which bank of the EEPROM to read, write or erase.

00: Byte 0 ~ Byte 7

01: Byte 8 ~ Byte 15

10: Byte 16 ~ Byte 23

11: Byte 24 ~ Byte 31

R17 [5] EE\_Reset: EEPROM FIFO Address Reset flag.

0: Default. EE\_Reset is set to "0" after FIFO address is reset.

1: Reset EEPROM FIFO address by firmware

R17 [6] EE\_Act: EEPROM activated mode switch.

0: Activate all EEPROM

1: Activate partial EEPROM

#### 6.2.1.21 R18~R1F (General Purpose Registers)

R17~R1F are general-purpose registers.

#### 6.2.1.22 R20~R3F (General Purpose Registers)

R20~R3F (including Banks 0~3) are general-purpose registers.

**Extra RAM1:**

#### 6.2.1.23 R10 (PWP)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWP7	PWP6	PWP5	PWP4	PWP3	PWP2	PWP1	PWP0

R10 [0~7] PWP7~PWP0 bits are set of pulse widths preset in advance for the desired width of baud clock.

$$Pulse\ Width = (PWP + 1) \times TMR$$

#### 6.2.1.24 R11 (SPIN)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	SPIC7	SPIC6	SPIC5	SPIC4	-	-	-

R11 [0~2]: Reserved

R11 [3~6] SPI byte number: For setting SPI byte number. The maximum number is 8 bytes

R11 [7]: Reserved

### 6.2.1.25 R12 (SPIS)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Lstatus1	Lstatus0	M/LSB	--	OD3	OD4	-	RBF

R12[0] RBF: Read Buffer Full flag

**0:** Receiving not completed yet, and SPIRB (SPI Read Buffer) has not fully exchanged

**1:** Receiving completed; SPIRB (SPI Read Buffer) is fully exchanged.

R12 [1]: Reserved

R12 [2] OD4: Open-drain Control bit:

**0:** Open-drain disable for SCK

**1:** Open-drain enable for SCK

R12 [3] OD3: Open-drain Control bit:

**0:** Open-drain disable for SDO

**1:** Open-drain enable for SDO

R12 [4]: Reserved

R12 [5] M/LSB: First bit of SPI Shift

**0:** LSB

**1:** MSB

R12 [6~7] Lstatus0~ Lstatus1:

**00:** The SDO will keep the status after transfer is over.

**01:** The SDO will pull-down after transfer is over.

**10:** The SDO will pull-high after transfer is over.

**11:** Not defined

### 6.2.1.26 R13 (SPIC)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CES	SPIE	SRO	SSE	--	SBR2	SBR1	SBR0

R13 [0~2] SBR0~2: SPI Baud Rate Select bits

SBRS2 (Bit 2)	SBRS1 (Bit 1)	SBRS0 (Bit 0)	Mode	Baud Rate
0	0	0	Master	Fosc/2
0	0	1	Master	Fosc/4
0	1	0	Master	Fosc/8
0	1	1	Master	Fosc/16
1	0	0	Master	Fosc/32
1	0	1	Slave	/SS enable
1	1	0	Slave	/SS disable
1	1	1	Master	TMR1/TMR2

R13 [3]: Reserved

R13 [4] SSE: SPI Shift Enable bit

**0:** Reset as soon as shifting is completed

**1:** Start to shift, and remain at 1 while the current transmissions are still being transmitted.

R13 [5] SRO: SPI Read Overflow bit

**0: No overflow**

**1:** A new data is received while the previous data is still being held in the SPI FIFO. In this situation, the data in SPI FIFO will be destroyed. To avoid setting this bit, user is required to read the SPIR register even though only transmission is implemented. This can only occur in slave mode.

R13 [6] SPIE: SPI Enable bit

**0:** Disable SPI mode

**1:** Enable SPI mode

R13 [7] CES : Clock Edge Select bit

**0:** Data shifts out on a rising edge, and shifts in on a falling edge. Data is on hold during a low level.

**1:** Data shifts out on a falling edge, and shifts in on a rising edge. Data is on hold during a high level.

**6.2.1.27 R14 (SPITMR1)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMR17	TMR16	TMR15	TMR14	TMR13	TMR12	TMR11	TMR10

R14 [0~7] TMR: TMR bits are set of Timer 1 registers which increases until the value matches PWP and then resets to 0.

**R15 (SE1\_LOW ):** Low signal counter of the 1st RF module that is inputted from P92.

**R16 (SE1\_HIGH ):** High signal counter of the 1st RF module that is inputted from P92.

**R17 (SE2\_LOW ):** Low signal counter of the 2nd RF module that is inputted from P93.

**R18 (SE2\_HIGH ):** High signal counter of the 2nd RF module that is inputted from P93.

R15 ~ R18 are RF Timing counter registers if RF function is enabled by setting Bit 2 or Bit 3 of IOCF. Otherwise, they are general-purpose registers.

**R19~R1F (General Purpose Register)**

R19~R1F are general-purpose registers.

### R20~R3F (General Purpose Register)

R20~R3F (including Banks 4~7) are general-purpose registers.

## 6.3 Special Function Registers

### 6.3.1 A (Accumulator)

The accumulator is an 8-bit register that holds operands and results of arithmetic calculations. It is not addressable. After an interrupt occurs, the Accumulator is auto-saved by hardware.

### 6.3.2 CONT (Control Register)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	INT	TSR2	TSR1	TSR0	PSR2	PSR1	PSR0

Except for Bit 6 (Interrupt enable control bit), the CONT register can be read by the instruction "CONTR" and written to by the instruction "CONTW".

CONT [6] INT: An interrupt enable flag cannot be written by the CONTW instruction.

CONT [3~5] TSR0 ~ TSR2: TCC prescaler bits

CONT [0~2] PSR0 ~ PSR2: WDT prescaler bits

PSR2	PSR1	PSR0	TCC Rate (Base Freq: Fosc/2)	WDT Rate
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

### 6.3.3 IOC5 ~ IOC9 Port Direction Control Registers

These are I/O port (Port 5 ~ Port 7) direction control registers. Each bit controls the I/O direction of three I/O ports respectively. When these bits are set to "1", the relative I/O pins become input pins. Similarly, the I/O pins becomes outputs when the relative control bits are cleared.

0: Output direction

1: Input direction

### 6.3.4 IOCA (RFCNT)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	RF2	RF1	RF0	RF_DBN2	RF_DBN1	RF_DBN0

IOCA [0~2] RF\_DBN0 ~ RF\_DBN2: These are used for defining debounce times in RF pattern detecting application.

IOCA [3~5] RF0 ~ RF2: RF Timing prescaler bits. These are base on MCU frequency.

RF2	RF1	RF0	Timing Rate	8 MHz System Clock (Time(Cnt.))	256kHz RC Mode (Time (Cnt.))
0	0	0	1:1	0.125 $\mu$ s (1), 31.875 $\mu$ s (255)	3.91 $\mu$ s (1), 996.1 $\mu$ s (255)
0	0	1	1:2	0.25 $\mu$ s (1), 63.75 $\mu$ s (255)	7.81 $\mu$ s (1), 1992 $\mu$ s (255)
0	1	0	1:4	0.5 $\mu$ s (1), 127.5 $\mu$ s (255)	15.625 $\mu$ s (1), 3984 $\mu$ s (255)
0	1	1	1:8	1 $\mu$ s (1), 255 $\mu$ s (255)	31.25 $\mu$ s (1), 7969 $\mu$ s (255)
1	0	0	1:16	2 $\mu$ s (1), 510 $\mu$ s (255)	62.5 $\mu$ s (1), 15.938 ms (255)
1	0	1	1:32	4 $\mu$ s (1), 1020 $\mu$ s (255)	125 $\mu$ s (1), 31.875 ms (255)
1	1	0	1:64	8 $\mu$ s (1), 2040 $\mu$ s (255)	250 $\mu$ s (1), 63.75 ms (255)
1	1	1	1:128	16 $\mu$ s (1), 4080 $\mu$ s (255)	500 $\mu$ s (1), 127.5 ms (255)

### 6.3.5 IOCB (PWM\_CNT) : Default (0b0000\_0001)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2_E	PWM1_E	-	-	D+_PH	PWM_SR2	PWM_SR1	PWM_SR0

IOCB [0~2] PWM\_SR0 ~ PWM\_SR2: PWM clock prescaler

Base on MCU frequency

PWM_SR2	PWM_SR1	PWM_SR0	Clock (Hz)
0	0	0	Fosc/2
0	0	1	Fosc/4
0	1	0	Fosc/8
0	1	1	Fosc/16
1	0	0	Fosc/32
1	0	1	Fosc/64
1	1	0	Fosc/128
1	1	1	Fosc/256

IOCB [3] ( D+\_PH): Programmable D+ pull-high resistor.

**0**: Enable D+ internal pull-high resistor

**1**: Disable D+ internal pull-high resistor

IOCB [6] (PWM1\_E): PWM1 Enable. The 1st PWM (P92) module enable bit.

**0**: Disable the PWM function of the 1st module

**1**: Enable the PWM function of the 1st module



IOCB [7] (PWM2\_E): PWM2 Enable: The 2nd PWM (P93) module enable bit.

**0:** Disable the PWM function of the 2nd module

**1:** Enable the PWM function of the 2nd module

**6.3.6 IOCC (SPIT1CON:SPI Timer 1 Control Register)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	TM1E	TM1P1	TM1P0

IOCC [0~1]: Timer 1 Prescaler bit

TM1P1	TM1P0	Prescaler Rate
0	0	1:4
0	1	1:16
1	0	1:32
1	1	1:64

IOCC [2]: Timer 1 Function Enable bit

**0:** Disable Timer 1 function as default.

**1:** Enable Timer 1 function.

**6.3.7 IOCD (Port 9 Pull-high Control Register):**

*Default Value: (0B\_1111\_1111)*

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	/PH96	/PH95	1	/PH93	/PH92	/PH91	/PH90

IOCD [0~6] /PH90 ~ /PH96: These bits control the 25KΩ pull-high resistor of individual pins in Port 9. If the I/O port is set as output, the pull-high function is disabled.

**0:** Enable pull-high function

**1:** Disable pull-high function

**6.3.8 IOCE (MCU Control Register): Default (0b1101\_1111)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
S7	/WUE	WTE	MCU_Run	/PU7	/PU8	/PU6	/PU5

IOCE [0~3] /PU5~/PU8: Pull-High Control register. Default=1, Disable pull high function. If the I/O port is set as output, the pull-high function is disabled.

**0:** Enable pull-high function

**1:** Disable pull-high function

IOCE [4] MCU Run bit: This bit can be cleared by the firmware and set during power-on, or by the hardware at a falling edge of wake-up signal. When this bit is cleared, the clock system is disabled and the MCU enters into Power down mode. At the transition of wake-up signal from high to low, this bit is set to enable the clock system.

**0:** Sleep mode. The device is in power down mode.

**1:** Run mode. The device is working normally.

IOCE [5] WTE: Watchdog timer enable bit. WDT is disabled / enabled by the WTE bit.

**0:** Disable WDT

**1:** Enable WDT

IOCE [6] /WUE: Enable the weak-up function as triggered by port- changed.

**0:** Enable wake-up function

**1:** Disable wake-up function

IOCE [7] S7 bit: S7 defines the driving ability of P70-P72

**0:** Normal output

**1:** Enhance the driving ability of LED

### 6.3.9 IOCF (Interrupt Mask Register)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OUTEP_IE	P76/P77_IE	SPI_IE	Resume_IE	RF2_IE	RF1_IE	USB_IE	TCC_IE

IOCF [0] TCC\_IE: TCIF interrupt enable bit

**0:** Disable TCIF interrupt

**1:** Enable TCIF interrupt

IOCF [1] USB\_IE: USB interrupt enable bit. Bits 1, 2, 3 of RF interrupt will be enabled while this bit is set.

**0:** Disable USB\_IE interrupt

**1:** Enable USB\_IE interrupt

IOCF [2] RF1\_IE: RF1 pattern compare interrupt enable bit. Bits 0, 1 of RE interrupt will be enabled while this bit is set.

**0:** Disable RF1\_IE interrupt

**1:** Enable RF1\_IE interrupt



IOCF [3] RF2\_IE: RF2 pattern compare interrupt enable bit. Bits 2, 3 of RE interrupt will be enabled while this bit is set.

**0:** Disable RF2\_IE interrupt

**1:** Enable RF2\_IE interrupt

IOCF [4] Resume\_IE: USB Resume interrupt enable bit

**0:** Disable Resume\_IE interrupt

**1:** Enable Resume\_IE interrupt

IOCF [5] SPI and TM1 interrupt enable bit. Bits 4, 5 of RE interrupt will be enabled while this bit is set.

**0:** Disable SPI interrupt

**1:** Enable SPI interrupt

IOCF [6] P76/P77\_IE: P76/P77 port change interrupt enable bit. Bits 6, 7 of RE interrupt will be enabled while this bit is set.

**0:** Disable P76/P77\_IE interrupt

**1:** Enable P76/P77\_IE interrupt

IOCF [7] OUTEP\_IE: Output Endpoint interrupt enable bit

**0:** Disable OUTEP\_IE interrupt

**1:** Enable OUTEP\_IE interrupt

Only when the global interrupt is enabled by the ENI instruction will the individual interrupt work. After DISI instruction, any interrupt will not work even if the respective control bits of IOCF are set to "1".

## **6.4 USB Device Controller**

The EM78M680 built-in USB Device Controller (UDC) can interpret the USB Standard Command and respond automatically without involving firmware. The embedded Series Interface Engine (SIE) handles the serialization and de-serialization of actual USB transmission. Thus, a developer can concentrate his efforts more in perfecting the device' actual functions and spend less effort in dealing with USB transaction.

The UDC handles and decodes most Standard USB commands defined in the USB Specification Rev 1.1. If the UDC receives an unsupported command, it will set a flag to notify the MCU of the receipt of such command.

Each time the UDC receives a USB command, it writes the command into EP0's FIFO. Only when it receives unsupported command will the UDC notify the MCU through interrupt.

Therefore, the EM78M680 is very flexible for USB applications since developers can freely choose the method of decoding the USB command as called for by different situations.

## 6.5 Device Address and Endpoints

The EM78M680 supports one device address and six endpoints, EP0 for control endpoint, EP1 ~ EP5 for interrupt/bulk endpoints. Sending data to USB host in EM78M680 is very easy. Just write the data into the EP's FIFO then set the flag, and the UDC will handle the rest. It will then confirm that the USB host has received the correct data from the EM78M680.

## 6.6 Reset

The EM78M680 provides three types of reset:

- (1) Power-on Reset
- (2) Watchdog Reset
- (3) USB Reset

### 6.6.1 Power-on Reset

Power-on reset occurs when the device is attached to a power source and a reset signal is initiated. The signal will last until the MCU becomes stable. After a Power-on reset, the MCU enters the following predetermined states (see below), and then, it is ready to execute the program.

- a. The program counter is cleared.
- b. The TCC timer and Watchdog timer are cleared.
- c. Special registers and Special Control registers are all set to their initial value.

### 6.6.2 Watchdog Reset

When the Watchdog Timer overflows, it causes the Watchdog to reset. After it resets, the program is executed from the beginning and some registers will be reset. The UDC however, remains unaffected.

### 6.6.3 USB Reset

When the UDC detects a USB Reset signal on the USB Bus, it interrupts the MCU, and proceeds to perform the specified process that follows. After a USB device is attached to the USB port, it cannot respond to any bus transactions until it receives a USB reset signal from the bus.

## 6.7 Saving Power Mode

The EM78M680 provides two options of power-saving modes for energy conservation, i.e., Power-down mode and Dual clock mode.

### 6.7.1 Power-down Mode

The EM78M680 enters into Power-down mode by clearing the SLPC register (IOCE[4]). During this mode, the oscillator is turned off and the MCU goes to sleep. It will wake up when signal from the USB host is resumed, or when the Watchdog resets or the input port state changes.

If the MCU wakes up when the I/O port status changes, the direction of the I/O port should be set at input direction, then read the port status.

For example:

```
:  
// Set the Port 6 to input port  
MOV      A , @0XFF  
IOW      PORT6  
// Read the status of Port 6  
MOV      PORT6, PORT6  
// Clear the RUN bit  
IOR      0XE  
AND      A , 0B11101111  
IOW      0XE  
:  
:
```

If the MCU is awoken by a USB Resume signal, the next instruction will be executed and one flag, IOCE [4] will be set to “1”.

### 6.7.2 Dual Clock Mode

The EM78M680 has one internal oscillator for power saving application. Clearing the Bit R12 [3] of ExtraRAM 0 will enable the low frequency oscillator. At the same time, the external oscillator will be turned off. Then the MCU will run under very low frequency to conserve power. Four types of frequencies are available for selection in setting Bits R12 [4, 5].

The USB Host Resume Interrupt can only be used in this mode. If this interrupt is enabled, the MCU will be interrupted when the USB Resume signal is detected on the USB Bus.

## 6.8 Interrupt

The EM78M680 has four Interrupt Vectors 0x0008, 0x0010, 0x0018, and 0x0020. When an interrupt occurs during an MCU running program, it will jump to the interrupt vector and execute the instructions sequentially from the interrupt vector. RE and RF are the interrupt status registers, which record the interrupt status in the relative flags/bits.

The interrupt condition could be one of the following:

- **Set 1 INT: (Jump to 0x08)**
  - TCC overflow interrupt
  - EP0 command interrupt
  - USB suspend interrupt
  - USB reset interrupt
  - USB HOST resume interrupt
- **Set 2 INT: (Jump to 0x10)**
  - RF1 low pattern interrupt
  - RF1 high pattern interrupt
  - RF2 low pattern interrupt
  - RF2 high pattern interrupt
- **Set 3 INT: (Jump to 0x18)**
  - P77 port change interrupt
  - P76 port change interrupt
  - SPI Timer 1 comparator completed interrupt (Note 1)
  - SPI transmission completed interrupt (Note 1).
- **Set 4 INT: (Jump to 0x20)**
  - EP5~8 output Endpoint received OK interrupt

IOCF is an interrupt mask register which can be set bit by bit. While their respective bit is written with a 0, the hardware interrupt will be inhibited, that is, the EM78M680 will not jump to the interrupt vector to execute instructions. But the interrupt status flags still records the conditions no matter whether the interrupt is masked or not. The interrupt status flags must be cleared by firmware before leaving the interrupt service routine and enabling interrupt.

The global interrupt is enabled by the ENI (RETI) instruction and is disabled by the DISI instruction.



## 6.9 Pattern Detect Application (PDA)

### 6.9.1 Functional Description

This function is designed for serial signal transmission, e.g., the transmission between a wireless device and its receiver box. The EM78M680 has two sets of built-in Pattern Detect Application block that ensures the EM78M680 is equipped with a compound device, such as the receiver box controller for a wireless keyboard paired with a wireless mouse.

Pattern Detect Application (PDA) can calculate the length of one pattern and interrupt the MCU while the serial signal is transiting from high to low (or vise-versa). Then the MCU reads the length value from a specified register.

### 6.9.2 Control Register

The PDA includes two enable control bits, one control register and four length counter registers in 0x15 ~0x18 in ExtraRAM1.

#### 6.9.2.1 IOCF [2~3] PDA Enable Control Bit

When this bit is set, the PDA function starts and the P92 and P93 automatically become input pins to sample the serial signal. It should be noted that enabling these two bits will also enable the interrupt mask of PDA.

0: disable PDA function

1: enable PDA function

#### 6.9.2.2 IOCA (PDA Control Register) Default Value: (0B\_0000\_0000)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		RF.2	RF.1	RF.0	DB2	DB1	DB0

This register is used to define two parameters of PDA function; signal sampling rate and debounce length. When a pattern ends, the value in the counter is loaded into its respective register and the RE[0~4] is set to indicate which channel and which type of pattern (high or low) is at its end or which type of pattern counter overflows.

0: low pattern

1: high pattern

#### 6.9.2.3 R15 ExtraRAM1 (P.92 Low Pattern Counter)

This register records the length of P.92 in low status.

#### 6.9.2.4 R16 ExtraRAM1 (P.92 High Pattern Counter)

This register records the length of P.92 in high status.

#### 6.9.2.5 R17 ExtraRAM1 (P.93 Low Pattern Counter)

This register records the length of P.93 in low status

### 6.9.2.6 R18 ExtraRAM1 (P.93 High Pattern Counter)

This register records the length of P.93 in high status.

R15~R18 function as general registers if this function is not enabled. Once the enabled bit is set, these four registers will be loaded with the value of the pattern counter.

### 6.9.3 Sampling Rate and Debounce Length

The two pattern detect pins are separate, and each pin has its own pattern counter. Both pins use the same Sampling Rate and Debounce Length parameters.

The PDA samples the serial signal for every fixed interval. The pattern counter will be incremented by one at sampling time if the signal remains unchanged. If the signal is at high state, then the “high pattern counter” is incremented, otherwise the “low pattern counter” is incremented. As long as the signal state changes, the PDA will debounce signal and load the value of the pattern counter into the respective register for the firmware to read. For example, if the signal in P.92 is in “low” state, the low counter of P.92 will count continuously until the state of the input signal in P.92 changes. When a state change occurs (in this case, the signal changes from “low” to “high” state), the PDA will take a time break (which is equal to the result of sampling interval multiplied by the debounce length), to avoid possible noise. After the debounce length time, if the signal remains in high state, the high pattern counter will start to count and load the low pattern counter’s value into R15 ExtraRAM1. At the same time, RE [0] is set to indicate that the low pattern is over.

The correlation between the control register value and debounce time are as follows:

DB.2	DB.1	DB.0	Debounce Time
0	0	0	No Sampling clock
0	0	1	1 Sampling clock
0	1	0	2 Sampling clocks
0	1	1	3 Sampling clocks
1	0	0	4 Sampling clocks
1	0	1	5 Sampling clocks
1	1	0	6 Sampling clocks
1	1	1	7 Sampling clocks

Now consider another situation of this case, where the signal of P92 always stays “low”. The low pattern counter of P92 will eventually overflow. Once the counter overflows, the content of the counter will also be loaded into R15, that is, the register is written to 0xFF, and the counter is reset to count from zero again.

If the hardware interrupt of PDA function is enabled, (IOCF[2] is equal to “1”), then the program will go to 0x0010 to execute interrupt routine while the content of a pattern counter is loaded into the register.

The correlation between the value of control register and actual sampling rate are shown below:

RF2	RF1	RF0	Timing Rate	8 MHz System Clock (Time(Cnt.))	256kHz RC Mode (Time (Cnt.))
0	0	0	1:1	0.125 $\mu$ s (1), 31.875 $\mu$ s (255)	3.91 $\mu$ s (1), 996.1 $\mu$ s (255)
0	0	1	1:2	0.25 $\mu$ s (1), 63.75 $\mu$ s (255)	7.81 $\mu$ s (1), 1992 $\mu$ s (255)
0	1	0	1:4	0.5 $\mu$ s (1), 127.5 $\mu$ s (255)	15.625 $\mu$ s (1), 3984 $\mu$ s (255)
0	1	1	1:8	1 $\mu$ s (1), 255 $\mu$ s (255)	31.25 $\mu$ s (1), 7969 $\mu$ s (255)
1	0	0	1:16	2 $\mu$ s (1), 510 $\mu$ s (255)	62.5 $\mu$ s (1), 15.938 ms (255)
1	0	1	1:32	4 $\mu$ s (1), 1020 $\mu$ s (255)	125 $\mu$ s (1), 31.875 ms (255)
1	1	0	1:64	8 $\mu$ s (1), 2040 $\mu$ s (255)	250 $\mu$ s (1), 63.75 ms (255)
1	1	1	1:128	16 $\mu$ s (1), 4080 $\mu$ s (255)	500 $\mu$ s (1), 127.5 ms (255)

User can write a default value to the High Pattern counter register and Low Pattern counter register. Then set the corresponding interrupt enable bit (IOCF [2, 3]). When the counting value of one “H” pattern is bigger than the default value of R15\_ExtraRAM1, a Pattern Detecting interrupt will be generated. Similarly, if the counting value of one “L” pattern is bigger than the default value of R16\_ExtraRAM1, a Low Pattern Detecting interrupt will occur. Thus, the EM78M680 is notified and aware that one effective pattern is received from P.92.

## 6.10 Pulse Width Modulation (PWM)

### 6.10.1 Functional Description

In PWM mode, both of PWM1 (P.92) and PWM2 (P.93) produce plus programmable signal of up to 8 bits resolution.

The PWM Period is defined as  $0xFF \times \text{Timer Counter Clock}$ . The Timer Counter clock source is controlled by the control register IOCB. For example; if the clock source is 1 MHz, then the Period will be 255  $\mu$ s.  $\text{Period} = 255 \times (1/\text{Timer Counter Clock})$ . The PWM duty cycle is defined by writing to the R15/R16 Register of ExtraRAM0 for PWM1/PWM2.

Duty Cycle =  $(\text{R15 of ExtraRAM0} / 255) \times 100\%$  for PWM1

$(\text{R16 of ExtraRAM0} / 255) \times 100\%$  for PWM2

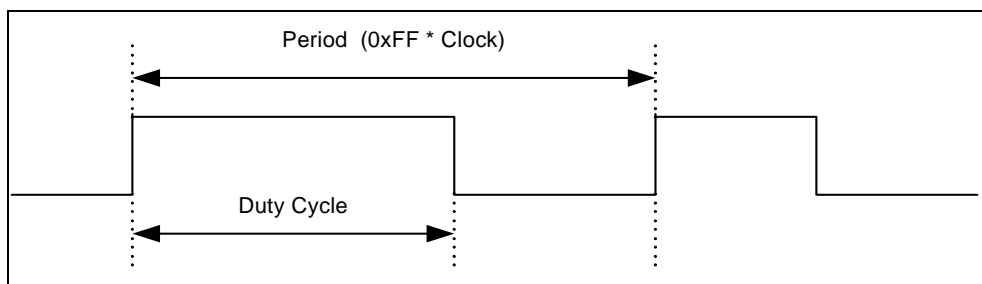


Figure 6-3 PWM Output Timing Diagram

## 6.11 Analog-to-Digital Converter (ADC)

### 6.11.1 Functional Description

The Analog to Digital converter consists of a 5-bit analog multiplexer, one Control Register (R11\_ExtraRAM0), and two data registers (R13\_ExtraRAM0 ~ R14\_ExtraRAM0) for a 10-bit resolution.

The ADC module utilizes successive approximation to convert the unknown analog signal to a digital value. The result is fed to the AD DATA. Input channels are selected by the analog input multiplexer via the ADCR/AD\_Sel bits AD0~AD4.

- 10-bit resolution: 0x00-00~0xC0-FF (0b11000000-11111111)
- Start (0x00-00):  $0 \text{ Vref} \sim (1/1024) \times \text{Vref}$
- Full (0xC0-FF):  $(1023/1024) \times \text{Vref} \sim \text{Vref}$
- Conversion Time: 12 clock time of internal clock source

### 6.11.2 Control Register

RAM	Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
External Ram0	0X11	AD/Control	AD_start	AD_R1	AD_R0	AD4	AD3	AD2	AD1	AD0
External Ram0	0x13	ADLoData	ADD1	ADD0	-	-	-	-	-	-
External Ram0	0X14	ADHiData	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2

## 6.12 Serial Peripheral Interface Mode

### 6.12.1 Overview

Figures 6-4, 6-5, and 6-6 show how the EM78M680 communicates with other devices through an SPI module. If the EM78M680 is a master controller, it sends clock through the SCK pin. A couple of 8 bits multiple data are transmitted and received at the same time. However, if EM78M680 is defined as a slave, its SCK pin could be programmed as an input pin.

### 6.12.2 Features

- Two 8 bytes FIFO for SPI shift in/out data
- Operation in either Master mode or Slave mode
- Three-wire or four-wire synchronous communication; that is, full duplex
- Programmable baud rates of communication
- Programming clock polarity, (ExtraRAM1 R13 Bit 7)
- Interrupt flag available for the read buffer full
- Up to 8 Mbps (maximum baud rates)
- SPI transmission order
- SPI handshake pin

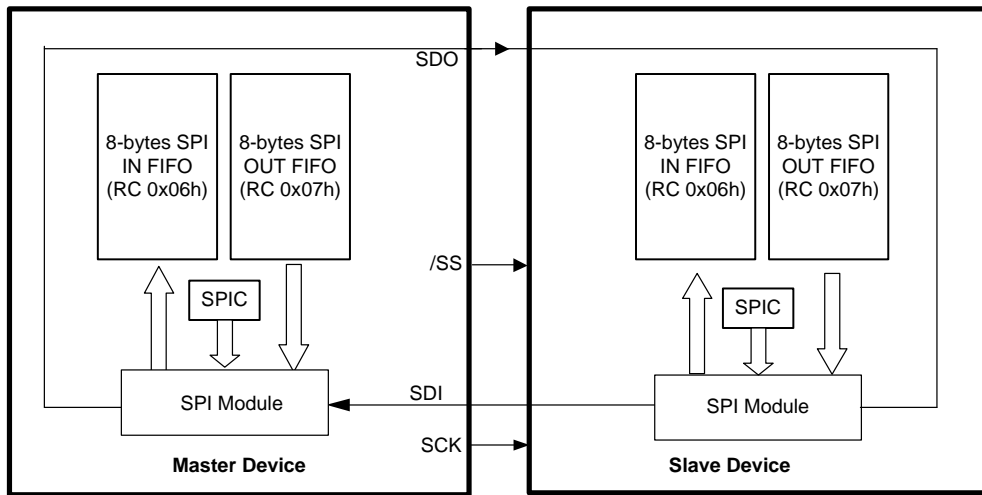


Figure 6-4 SPI Master/Slave Communication

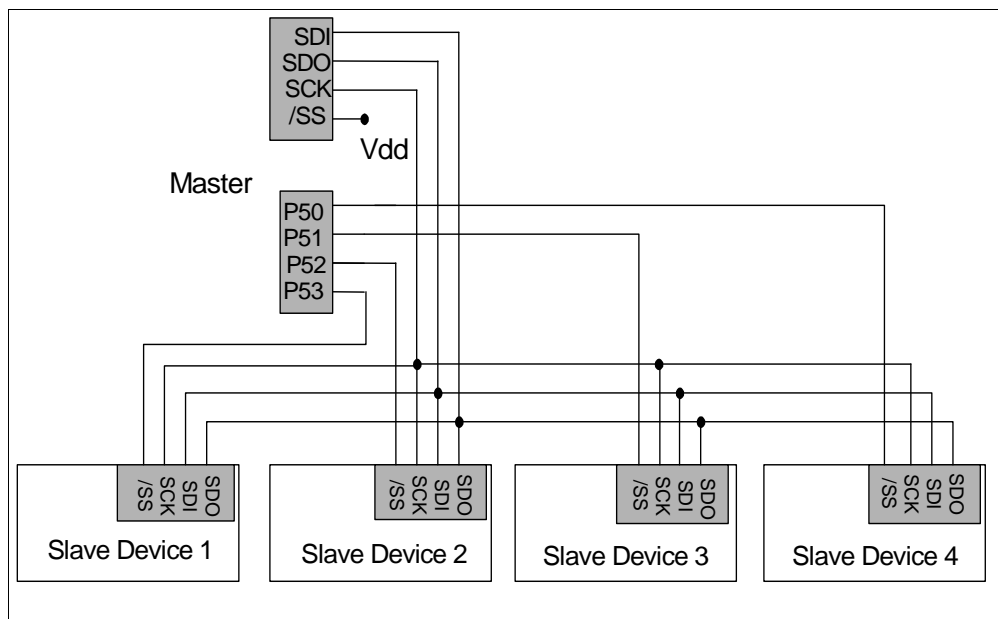


Figure 6-5 SPI Configuration of Single-Master and Multi-Slave

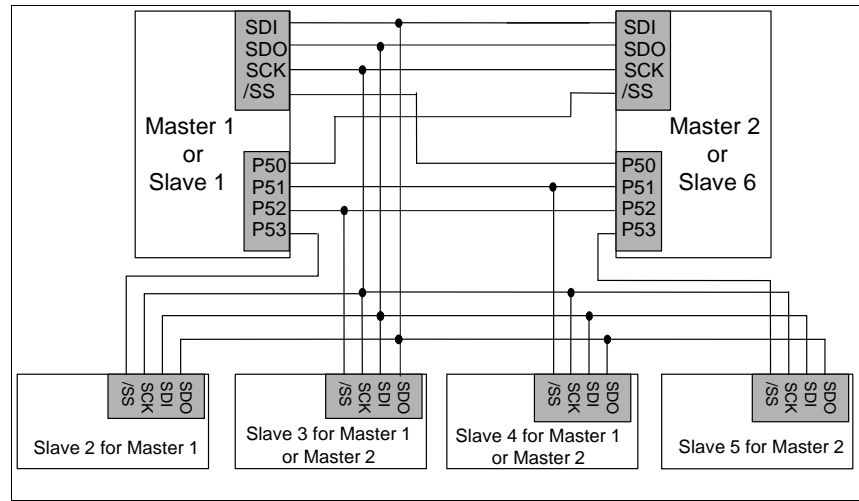


Figure 6-6 SPI Configuration of Single-Master and Multi-Slave

### 6.12.3 SPI Functional Description

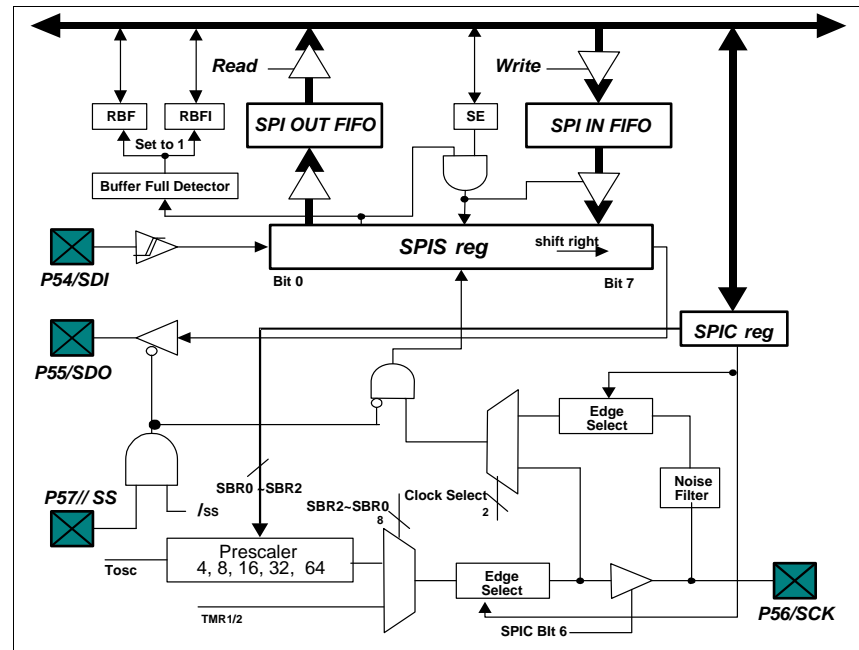


Figure 6-7 SPI Block Diagram



### **6.12.4 SPI Signal and Pin Description**

The detailed functions of the four pins, SDI, SDO, SCK, and /SS, which are shown in Figure 6-5, are as follows:

#### **SDI/P54 :**

- Serial Data In
- Receive serially, the Most Significant Bit (MSB) first, Least Significant Bit (LSB) last.
- Defined as high-impedance, if not selected.
- Program the same clock rate and clock edge to latch on both the master and slave devices.
- The received byte will update the transmitted byte.
- Both the RBF and RBFIF bits (ExtraRAM1 Register 0x12h) will be set as the SPI operation is completed.
- Timing is shown in Figure 6-9 and 6-10.

#### **SDO/P55 :**

- Serial Data Out
- Transmit serially; the Most Significant Bit (MSB) first, Least Significant Bit (LSB) last
- Program the same clock rate and clock edge to latch on both the master and slave devices.
- The received byte will update the transmitted byte.
- The CES (ExtraRAM1 Register 0x13h) bit will be reset, as the SPI operation is completed.
- Timing is shown in Figure 6-9 and 6-10.

#### **SCK/P56 :**

- Serial Clock
- Generated by a master device
- Synchronize the data communication on both the SDI and SDO pins
- The CES (ExtraRAM1 Register 0x13h) is used to select the edge to communicate.
- The SBR0~SBR2 (ExtraRAM1 Register 0x13h) is used to determine the baud rate of communication.
- The CES, SBR0, SBR1, and SBR2 bits have no effect in slave mode.
- Timing is shown in Figure 6-9 and 6-10.

#### **/SS/P57 :**

- Slave Select; negative logic.
- Generated by a master device to signify the slave(s) to receive data.
- Goes low before the first cycle of SCK appears, and remains low until the last (8th) cycle is completed,
- Ignores the data on the SDI and SDO pins while /SS is high, because the SDO is no longer driven.
- Timing is shown in Figures 6-9 and 6-10.

### 6.12.5 SPI Control Registers

As the SPI mode is defined, the related registers of this operation are shown in Table 6-1 and Table 6-2.

**Table 6-1 Related Control Registers of the SPI Mode**

RAM	Addr.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
External Ram1	0x11	SPIS	-	SPIB3	SPIB2	SPIB1	SPIB0	-	-	-
External Ram1	0x12	SPIS	Lstatus1	Lstatus0	M/LSB	-	OD3	OD4	-	RBF
External Ram1	0x13	SPIC	CES	SPIE	SRO	SSE	-	SBR2	SBR1	SBR0
External Ram1	0x0F	IOCF	-	-	SPI_IE	-	-	-	-	-

### 6.12.6 SPI Mode Timing

The edge of SCK is selected by programming bit CES. The waveform shown in Figure 6-9 is applicable regardless whether the EM78M680 is in master or slave mode with /SS disabled. However, the waveform in Figure 6-10 can only be implemented in slave mode with /SS enabled.

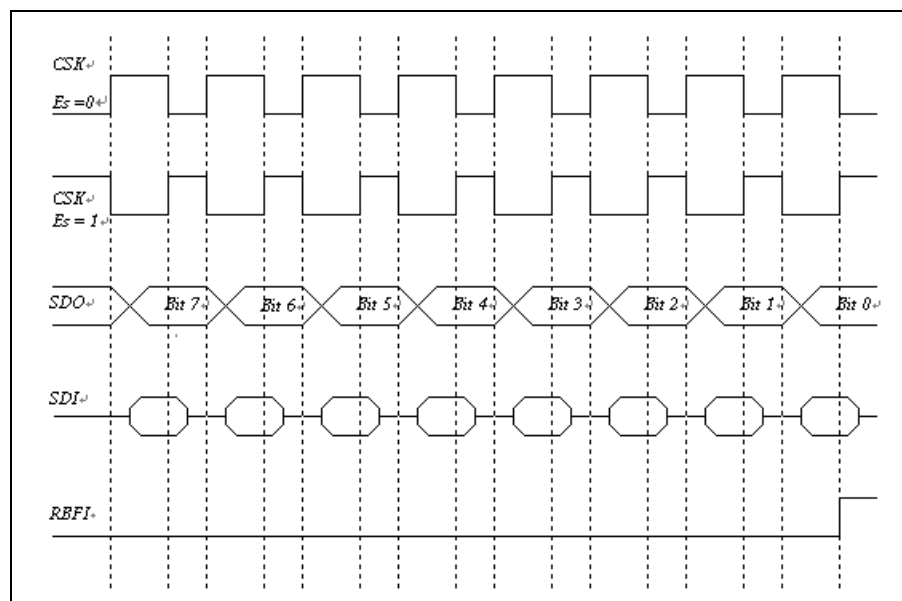


Figure 6-9 SPI Mode with /SS Disabled

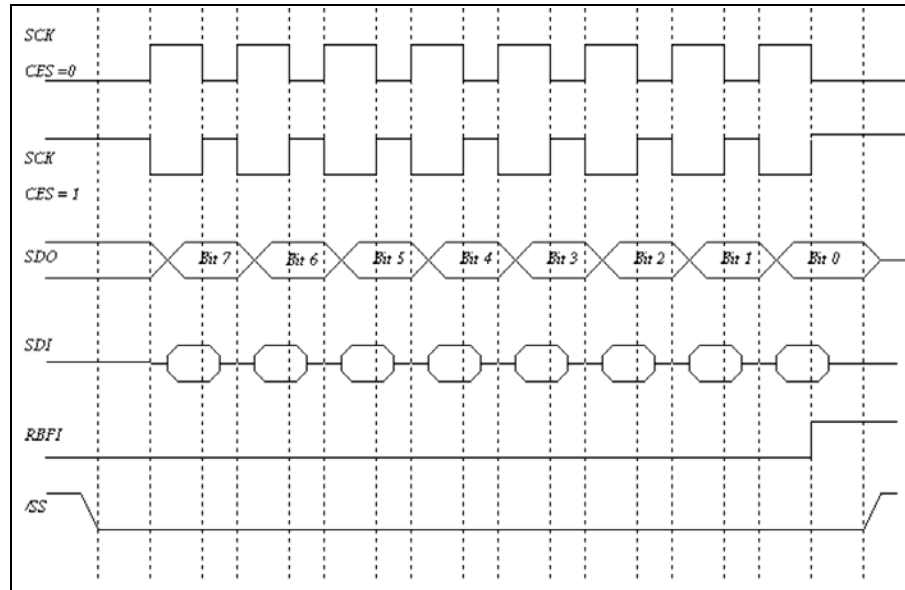


Figure 6-10 SPI Mode with /SS Enabled

Example for SPI:

**For Master**

```

BS 0XA,7                ; Switch to external RAM1
MOV A,@0X18
MOV 0X11, A             ; SET SPI THREE BYTES
MOV A,@0X07             ; POINT TO SPI WRITE FIFO
MOV 0XC,A
MOV A,@0XAA             ; FILL THREE BYTES SPI WRITE BUFFER
MOV 0XD,A
MOV 0XD,A
MOV 0XD,A

MOV A,@0X00             ; CLEAR RBF AND RBFIF FLAG
                        ; SDO KEEP STATUS, MSB FIRST

MOV 0X12,A

MOV A,@0B11000010      ; SELECT CLOCK EDGE AND ENABLE SPI
                        ; FUNCTION

MOV 0X13,A
BS 0X13,4               ; START TO SHIFT SPI DATA
POLLING_LOOP:
  JBC 0X13,4            ; POLLING LOOP FOR CHECKING SPI
                        ; TRANSMISSION COMPLETED

  JMP POLLING_LOOP

```

**For Slaver example:**

```

BS RA,7                ; Switch to external RAM1
MOV A,@0x18
MOV 0x11,A             ; SET SPI THREE BYTES
MOV A,@0x00            ; SPIS control register setting
MOV 0x12,A

```

```

MOV A,@0B11000110
MOV 0x13,A                ; SET CES=1 ,SPI ENABLE, SLAVE MODE
                           ; WITH /SS DISABLE
BS 0X13,4                 ; START TO SHIFT SPI DATA
S_POLLING_LOOP:
    JBC 0X13,4            ; POLLING LOOP FOR CHECKING SPI
                           ; TRANSMISSION COMPLETED

    JMP S_POLLING_LOOP
    MOV A,@0X06           ; POINT TO SPI READ FIFO
    MOV 0XC,A
    MOV A,0X0D            ; FILL THREE BYTES SPI WRITE BUFFER
    MOV A,0X0D
    MOV A,0X0D

```

### 6.12.7 SPI Timer 1 Mode

#### 6.12.7.1 Overview

Timer 1 (TMR1) is an 8-bit clock counter with programmable prescaler. It is designed for the SPI module as a baud rate clock generator. TMR1 can be read, written to and cleared on any reset conditions. If employed, it can be powered down for power saving purposes by setting TMR1EN Bit [T1CON<2>] to 0.

#### 6.12.7.2 Functional Description

Figure 6-11 shows the Timer 1 block diagram. Each signal and block is described as follows:

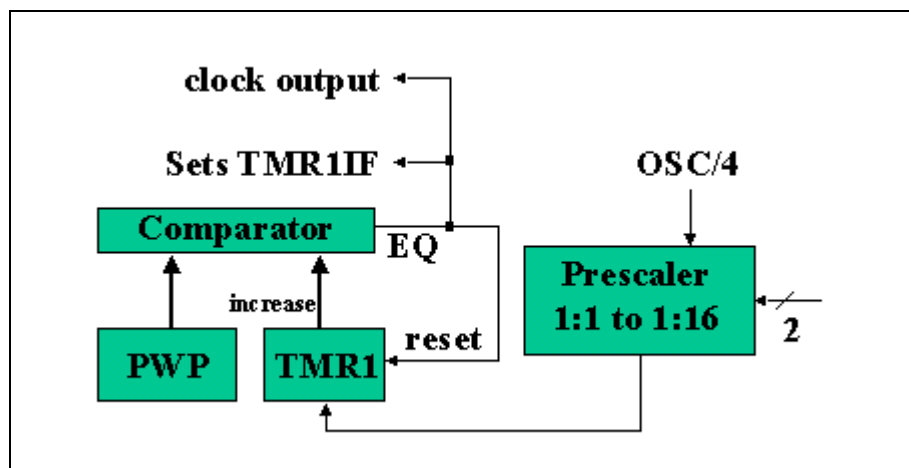


Figure 6-11 Timer 1 Block Diagram

**OSC/4:** Input clock

**Prescaler:** Option of 1:4, 1:16, 1:32, and 1:64 defined by T1P1 and T1P02 (T1CON<1, 0>). It is cleared when a value is written to TMR1 or T1CON, and during any kind of reset as well.



**PWP:** Pulse width preset register. The desired width of baud clock is written in advance.

**TMR1:** Timer 1 Register. TMR1 is incremented until it matches with PWP, and then resets to 0. If it is optionally chosen in SPI mode, its output is fed as a shifting clock.

**Comparator:** To change the output status while a match occurs. The TMR1IF flag will be set at the same time.

### 6.12.7.3 Programming the Related Registers

The related registers of the defining TMR1 operation are shown in Table 6-4 and Table 6-5.

**Table 6-4 Related Control Registers of TMR1**

RAM	Addr.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
External RAM1	0x0C	T1CON	0	0	0	0	0	TM1E	TM1P1	TM1P0
External RAM1	0x0E	ISR	P77_IF	P76_IF	TM1IF	SPIIF	RF_2High	RF_2Low	RF_1High	RF_1Low
External RAM1	0x10	PWP	PWP7	PWP6	PWP5	PWP4	PWP3	PWP2	PWP1	PWP0
External RAM1	0x12	SPIS	Lstatus1	Lstatus0	M/LSB	-	OD3	OD4	-	RBF
External RAM1	0x14	TMR1	TMR17	TMR16	TMR15	TMR14	TMR13	TMR12	TMR11	TMR10
	0x0F	IOCF	-	-	TMR1E/SPIE	-	-	-	-	-

## 7 Absolute Maximum Ratings

Items	Min.	Max.	Unit
Temperature under bias	0	70	°C
Storage temperature	-65	150	°C
Input voltage	-0.5	6.0	V
Output voltage	-0.5	6.0	V

## 8 DC Electrical Characteristics

Ta=25°C, VDD=5V, VSS=0V

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
VDD	Operating voltage	–	4.4	5.0	5.25	V
IIL	Input Leakage Current	VIN = VDD, VSS	–	–	±1	μA
VIH	Input High Voltage	–	2.0	–	–	V
VIL	Input Low Voltage	–	–	–	0.8	V
VIHX	Clock Input High Voltage	OSCI	2.5	–	–	V
VILX	Clock Input Low Voltage	OSCI	–	–	1.0	V
VOH1	Output High Voltage (P70~P72, P76~P77)	IOH = 7.0 mA VDD = 5V	2.4	–	–	V
VOH2	Output High Voltage (Ports 5, 6, 8, P90~P93, P95~P96)	IOH = 7.0 mA Vreg = 3.3V	2.4	–	–	V
VOL1	Output Low Voltage (P70~P72, P76~77)	IOL = -8.0 mA VDD = 5V	–	–	0.4	V
VOL2	Output Low Voltage (P70~P72 : LED drive mode)	IOL = -8.0 mA VDD = 5V	–	–	3.0	V
VOL3	Output Low Voltage (Ports 5,6,8, P90~P93, P95~P96)	IOL = -8.0 mA Vreg = 3.3V	–	–	0.4	V
IPH1	Pull-high current (Ports 5, 6, 7, 8, P90~P93, P95~P96)	Pull-high active, input pin at VSS Vreg=3.3V	-20%	132	+20%	μA
IPH2	Pull-high current (P70~P72, P76~77)	Pull-high active, input pin at VSS VDD = 5V	-20%	132	+20%	μA
IPH3	Pull-high current (USB D+)	Pull-high active, input pin at VSS Vreg=3.3V	–	2.2	–	mA
ICC1	Operating supply current Normal operation	Fosc= 8 MHz , no GPIO loading	–	–	10	mA
ICC2	Operating supply current Sleep mode	All input and I/O pin at VDD, Output pin floating, WDT disabled	–	–	100	μA
ICC3	Operating supply current Dual clock mode – 256kHz	All input and I/O pin at VDD, Output pin floating, WDT disabled	–	–	250	μA
Vreg	Output voltage of 3.3V regulator	VDD = 4.4V ~ 5.25V	3.0	3.3	3.6	V

## APPENDIX

### A Special Register Map

Addr.	Name	Reset Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
N/A	IOC5	Bit Name	C57	C56	C55	C54	C53	C52	C51	C50
		Power-on	1	1	1	1	1	1	1	1
		/RESET and WDT	1	1	1	1	1	1	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
N/A	IOC6	Bit Name	C67	C66	C65	C64	C63	C62	C61	C60
		Power-on	1	1	1	1	1	1	1	1
		/RESET and WDT	1	1	1	1	1	1	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
N/A	IOC7	Bit Name	C77	C76	-	-	-	C72	C71	C70
		Power-on	1	1	1	1	1	1	1	1
		/RESET and WDT	1	1	1	1	1	1	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
N/A	IOC8	Bit Name	C87	C86	C85	C84	C83	C82	C81	C80
		Power-on	1	1	1	1	1	1	1	1
		/RESET and WDT	1	1	1	1	1	1	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
N/A	IOC9	Bit Name	—	C96	C95	C94	C93	C92	C91	C90
		Power-on	1	1	1	1	1	1	1	1
		/RESET and WDT	1	1	1	1	1	1	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0A	IOCA (RFCNT)	Bit Name	-	-	RF2	RF1	RF0	RF_DB2	RF_DB1	RF_DB0
		Power-un	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0B	IOCB (PWM_CNT)	Bit Name	PW2_E	PW1_E	-	-	D+_PH	PWM_2	PWM_1	PWM_0
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P



(Continuation)

Addr.	Name	Reset Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0C	IOCC (T1CON)	Bit Name	GPB	GPB	GPB	GPB	GPB	T1E	T1P1	T1P0
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0D	IOCD (P9_PH)	Bit Name	-	/PH96	/PH95	-	/PH93	/PH92	/PH91	/PH90
		Power-on	1	1	1	1	1	1	1	1
		/RESET and WDT	1	1	1	1	1	1	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0E	IOCE (MCU Cnt)	Bit Name	S7	/WUE	WTE	SLPC	-	/PU8	/PU6	/PU5
		Power-on	1	1	0	1	U	1	1	1
		/RESET and WDT	1	1	0	1	U	1	1	1
		Wake-up from Pin Change	P	P	P	1	U	P	P	P
0x0F	IOCF	Bit Name	OUTEP	P7IE	SPIIE	ResulE	RF2IE	RF1IE	USBIE	TCIE
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
N/A	CONT	Bit Name	-	INT	TSR2	TSR1	TSR0	PSR2	PSR1	PSR0
		Power-on	U	0	1	1	1	1	1	1
		/RESET and WDT	U	P	1	1	1	1	1	1
		Wake-up from Pin Change	U	P	P	P	P	P	P	P
0x00	R0 (IAR)	Bit Name	-	-	-	-	-	-	-	-
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x01	R1 (TCC)	Bit Name	-	-	-	-	-	-	-	-
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x02	R2 (PC)	Bit Name	-	-	-	-	-	-	-	-
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	**P	**P	**P	**P	**P	**P	**P	**P



(Continuation)

Addr.	Name	Reset Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x03	R3 (SR)	Bit Name	PS2	PS1	PS0	T	P	Z	DC	C
		Power-on	0	0	0	t	t	U	U	U
		/RESET and WDT	0	0	0	t	t	P	P	P
		Wake-up from Pin Change	P	P	P	t	t	P	P	P
0x04	R4 (RSR)	Bit Name	RSR.1	RSR.0	-	-	-	-	-	-
		Power-on	0	0	U	U	U	U	U	U
		/RESET and WDT	0	0	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x05	R5 (P5)	Bit Name	P57	P56	P55	P54	P53	P52	P51	P50
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x06	R6 (P6)	Bit Name	P67	P66	P65	P64	P63	P62	P61	P60
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x07	R7 (P7)	Bit Name	P77	P76	-	-	-	P72	P71	P70
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x08	R8 (P8)	Bit Name	P87	P86	P85	P84	P83	P82	P81	P80
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x09	R9 (P9)	Bit Name	—	P96	P95	P94	P93	P92	P91	P90
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0A	RA (USBES)	Bit Name	Ext_R	Remote	EP0_W	EP0_R	D_Resu	UDC_Su	UDC_w	STALL
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P



(Continuation)

Addr.	Name	Reset Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0B	GPR	Bit Name	GPB	GPB	GPB	GPB	GPB	GPB	GPB	GPB
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0C	RC (FIFO_Index)	Bit Name	Index7	Index6	Index5	Index4	Index3	Index2	Index1	Index0
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0D	RD (FIFO_Data)	Bit Name	FIFO7	FIFO6	FIFO5	FIFO4	FIFO3	FIFO2	FIFO1	FIFO0
		Power-on	--	--	--	--	--	--	--	--
		/RESET and WDT	--	--	--	--	--	--	--	--
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0E	RE (ISR0)	Bit Name	P77_IF	P76_IF	TM1IF	SPIIF-	RF2_H	RF2_L	RF1_H	RF1_L
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x0F	RF (ISR1)	Bit Name	Out_EP	GPB	GPB	ResumIF	USBres	Suspend	EP0_IF	TCCIF
		Power-on	0	U	U	0	0	0	0	0
		/RESET and WDT	0	P	P	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x10	R10 (USB_EP_S)	Bit Name	GPB	GPB	GPB	EP5_ST	EP4_ST	EP3_ST	EP2_ST	EP1_ST
		Power-on	U	U	U	0	0	0	0	0
		/RESET and WDT	P	P	P	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x11	R11 (AD_Ctrl)	Bit Name	ADstart	AD_R1	AD_R0	AD4	AD3	AD2	AD1	AD0
		Power-on	0	0	0	1	1	1	1	1
		/RESET and WDT	0	0	0	1	1	1	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x12	R12 (DMC)	Bit Name	GPB	USB_To	Low_F1	Low_F0	/LowFre	GPB	GPB	GPB
		Power-on	U	U	0	0	1	U	U	U
		/RESET and WDT	P	U	0	0	1	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P



(Continuation)

Addr.	Name	Reset Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x13	R13 (ADDL)	Bit Name	ADD1	ADD0	-	-	-	-	-	-
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x14	R14 (ADDH)	Bit Name	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x15	R15 (PWM1_T)	Bit Name	PWM17	PWM16	PWM15	PWM14	PWM13	PWM12	PWM11	PWM10
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x16	R16 (PWM2_T)	Bit Name	PWM27	PWM26	PWM25	PWM24	PWM23	PWM22	PWM21	PWM20
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x17	R17 (EECNT)	Bit Name	GPB	EE_Act	EE_rest	EEA4	EEA3	EE_OK	EE_C1	EE_C0
		Power-on	U	0	0	0	0	1	1	1
		/RESET and WDT	P	0	0	0	0	0	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x18~ 0x3F	GPR	Bit Name	-	-	-	-	-	-	-	-
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x10	R10 (PWP)	Bit Name	PWP7	PWP6	PWP5	PWP4	PWP3	PWP2	PWP1	PWP0
		Power-on	1	1	1	1	1	1	1	1
		/RESET and WDT	1	1	1	1	1	1	1	1
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x11	R11 (SPIN)	Bit Name	GPB	SPI7	SPI6	SPI5	SPI4	GPB	GPB	GPB
		Power-on	U	0	0	0	0	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P

(Continuation)

Addr.	Name	Reset Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x12	R12 (SPIS)	Bit Name	Lstatus1	Lstatus0	M/LSB	GPB	OD3	OD4	GPB	RBF
		Power-on	0	0	0	U	0	0	U	0
		/RESET and WDT	0	0	0	P	0	0	P	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x13	R13 (SPIC)	Bit Name	CES	SPIE	SRO	SPISE	GPB	SBRS2	SBRS1	SBRS0
		Power-on	0	0	0	0	U	0	0	0
		/RESET and WDT	0	0	0	0	P	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x14	R14 (SPITMR1)	Bit Name	TMR17	TRM16	TRM15	TRM14	TRM13	TRM12	TRM11	TRM10
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x15	R15 (SE1_LOW)	Bit Name	SE1_L7	SE1_L6	SE1_L5	SE1_L4	SE1_L3	SE1_L2	SE1_L1	SE1_L0
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x16	R16 (SE1_HIGH)	Bit Name	SE1_H7	SE1_H6	SE1_H5	SE1_H4	SE1_H3	SE1_H2	SE1_H1	SE1_H0
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x17	R17 (SE2_LOW)	Bit Name	SE2_L7	SE2_L6	SE2_L5	SE2_L4	SE2_L3	SE2_L2	SE2_L1	SE2_L0
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x18	R18 (SE2_HIGH)	Bit Name	SE2_H7	SE2_H6	SE2_H5	SE2_H4	SE2_H3	SE2_H2	SE2_H1	SE2_H0
		Power-on	0	0	0	0	0	0	0	0
		/RESET and WDT	0	0	0	0	0	0	0	0
		Wake-up from Pin Change	P	P	P	P	P	P	P	P
0x19~ 0x3F	GPR	Bit Name	-	-	-	-	-	-	-	-
		Power-on	U	U	U	U	U	U	U	U
		/RESET and WDT	P	P	P	P	P	P	P	P
		Wake-up from Pin Change	P	P	P	P	P	P	P	P

\*\* Execute the next instruction after the "SLPC" bit status of IOCE register has been on high-to-low transition.

X: Not for use      U: Unknown or don't care      P: Previous value before reset

## B Instruction Set

Each instruction in the instruction set is a 13-bit word divided into an OP code and one or more operands. All instructions are executed within one single instruction cycle (consisting of 2 oscillator periods), unless the program counter is changed by-

- (a) Executing the instruction "MOV R2,A", "ADD R2,A", "TBL", or any other instructions that write to R2 (e.g. "SUB R2,A", "BS R2,6", "CLR R2", etc.).
- (b) Execute CALL, RET, RETI, RETL, JMP, Conditional skip (JBS, JBC, JZ, JZA, DJZ, DJZA) which were tested to be true.

Under these cases, the execution takes two instruction cycles.

In addition, the instruction set has the following features:

- (1). Every bit of any register can be set, cleared, or tested directly.
- (2). The I/O register can be regarded as general register. That is, the same instruction can operate on the I/O register.

### Legend:

**R** = Register designator that specifies which one of the 64 registers (including operation and general purpose registers) is to be utilized by the instruction.

Bits 6 and 7 in R4 determine the selected register bank.

**b** = Bit field designator that selects the value for the bit located in the register R and which affects the operation.

**k** = 8 or 10-bit constant or literal value

Binary Instruction	Hex	Mnemonic	Operation	Status Affected
0 0000 0000 0000	0000	NOP	No Operation	None
0 0000 0000 0001	0001	DAA	Decimal Adjust A	C
0 0000 0000 0010	0002	CONTW	A → CONT	None
0 0000 0000 0011	0003	SLEP	0 → WDT, Stop oscillator	T, P
0 0000 0000 0100	0004	WDTC	0 → WDT	T, P
0 0000 0000 rrrr	000r	IOW R	A → IOCR	None <Note <sup>1</sup> >
0 0000 0001 0000	0010	ENI	Enable Interrupt	None
0 0000 0001 0001	0011	DISI	Disable Interrupt	None
0 0000 0001 0010	0012	RET	[Top of Stack] → PC	None
0 0000 0001 0011	0013	RETI	[Top of Stack] → PC, Enable Interrupt	None
0 0000 0001 0100	0014	CONTR	CONT → A	None
0 0000 0001 rrrr	001r	IOR R	IOCR → A	None <Note <sup>1</sup> >
0 0000 0010 0000	0020	TBL	R2+A → R2, Bits 8-9 of R2 unchanged	Z, C, DC
0 0000 01rr rrrr	00rr	MOV R,A	A → R	None
0 0000 1000 0000	0080	CLRA	0 → A	Z
0 0000 11rr rrrr	00rr	CLR R	0 → R	Z

(Continuation)

Binary Instruction	Hex	Mnemonic	Operation	Status Affected
0 0001 00rr rrrr	01rr	SUB A,R	$R-A \rightarrow A$	Z, C, DC
0 0001 01rr rrrr	01rr	SUB R,A	$R-A \rightarrow R$	Z, C, DC
0 0001 10rr rrrr	01rr	DECA R	$R-1 \rightarrow A$	Z
0 0001 11rr rrrr	01rr	DEC R	$R-1 \rightarrow R$	Z
0 0010 00rr rrrr	02rr	OR A,R	$A \vee VR \rightarrow A$	Z
0 0010 01rr rrrr	02rr	OR R,A	$A \vee VR \rightarrow R$	Z
0 0010 10rr rrrr	02rr	AND A,R	$A \& R \rightarrow A$	Z
0 0010 11rr rrrr	02rr	AND R,A	$A \& R \rightarrow R$	Z
0 0011 00rr rrrr	03rr	XOR A,R	$A \oplus R \rightarrow A$	Z
0 0011 01rr rrrr	03rr	XOR R,A	$A \oplus R \rightarrow R$	Z
0 0011 10rr rrrr	03rr	ADD A,R	$A + R \rightarrow A$	Z, C, DC
0 0011 11rr rrrr	03rr	ADD R,A	$A + R \rightarrow R$	Z, C, DC
0 0100 00rr rrrr	04rr	MOV A,R	$R \rightarrow A$	Z
0 0100 01rr rrrr	04rr	MOV R,R	$R \rightarrow R$	Z
0 0100 10rr rrrr	04rr	COMA R	$\neg R \rightarrow A$	Z
0 0100 11rr rrrr	04rr	COM R	$\neg R \rightarrow R$	Z
0 0101 00rr rrrr	05rr	INCA R	$R+1 \rightarrow A$	Z
0 0101 01rr rrrr	05rr	INC R	$R+1 \rightarrow R$	Z
0 0101 10rr rrrr	05rr	DJZA R	$R-1 \rightarrow A$ , skip if zero	None
0 0101 11rr rrrr	05rr	DJZ R	$R-1 \rightarrow R$ , skip if zero	None
0 0110 00rr rrrr	06rr	RRCA R	$R(n) \rightarrow A(n-1)$ , $R(0) \rightarrow C$ , $C \rightarrow A(7)$	C
0 0110 01rr rrrr	06rr	RRC R	$R(n) \rightarrow R(n-1)$ , $R(0) \rightarrow C$ , $C \rightarrow R(7)$	C
0 0110 10rr rrrr	06rr	RLCA R	$R(n) \rightarrow A(n+1)$ , $R(7) \rightarrow C$ , $C \rightarrow A(0)$	C
0 0110 11rr rrrr	06rr	RLC R	$R(n) \rightarrow R(n+1)$ , $R(7) \rightarrow C$ , $C \rightarrow R(0)$	C
0 0111 00rr rrrr	07rr	SWAPA R	$R(0-3) \rightarrow A(4-7)$ , $R(4-7) \rightarrow A(0-3)$	None
0 0111 01rr rrrr	07rr	SWAP R	$R(0-3) \leftrightarrow R(4-7)$	None
0 0111 10rr rrrr	07rr	JZA R	$R+1 \rightarrow A$ , skip if zero	None
0 0111 11rr rrrr	07rr	JZ R	$R+1 \rightarrow R$ , skip if zero	None
0 100b brrr rrrr	0xxx	BC R,b	$0 \rightarrow R(b)$	None <Note <sup>2</sup> >
0 101b brrr rrrr	0xxx	BS R,b	$1 \rightarrow R(b)$	None
0 110b brrr rrrr	0xxx	JBC R,b	if $R(b)=0$ , skip	None
0 111b brrr rrrr	0xxx	JBS R,b	if $R(b)=1$ , skip	None



(Continuation)

Binary Instruction	Hex	Mnemonic	Operation	Status Affected
1 00kk kkkk kkkk	1kkk	CALL k	PC+1 → [SP], (Page, k) → PC	None
1 01kk kkkk kkkk	1kkk	JMP k	(Page, k) → PC	None
1 1000 kkkk kkkk	18kk	MOV A,k	k → A	None
1 1001 kkkk kkkk	19kk	OR A,k	A ∨ k → A	Z
1 1010 kkkk kkkk	1Akk	AND A,k	A & k → A	Z
1 1011 kkkk kkkk	1Bkk	XOR A,k	A ⊕ k → A	Z
1 1100 kkkk kkkk	1Ckk	RETL k	k → A, [Top of Stack] → PC	None
1 1101 kkkk kkkk	1Dkk	SUB A,k	k-A → A	Z, C, DC
1 1110 0000 0kkk	1E0k			None
1 1111 kkkk kkkk	1Fkk	ADD A,k	k+A → A	Z, C, DC

**Note:** <sup>1</sup> This instruction is applicable to IOCx only.

<sup>2</sup> This instruction is not recommended for RE and RF operation.

## C Code Option

Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8	Bit 9	Bit 10
/Protect	USB	/D+ resistor	OST0	OST1	FRQ0	FRQ1	PKG0	PKG1	Reversed	/AD hold

Bit 11	Bit 12	Bits 13~15	Bits 16~17	Bit 18	Bits 19~20	Bits 22~22	Bit 23
Reserved	Reserved	EPX_SEL	EP1 Type	EP1 Direction	EP1 Max Size	EP2 Type	EP2 Direction

Bits 24~25	Bits 26~27	Bit 28	Bits 29~30	Bits 31~32	Bit 33	Bits 34~35
EP2 Max Size	EP3 Type	EP3 Direction	EP3 Max Size	EP4 Type	EP4 Direction	EP4 Max Size

Bits 36~37	Bit 38	Bit 39~40	Bits 41~ 42	Bits 43~59	Bits 60~64
EP5 Type	EP5 Direction	EP5 Max Size	reserved	User ID	Reserved

Bit 65	Bit 66	Bit 67	Bit 68		
Reserved	Reserved	Reserved	D+ pull-high		

**Bit 0 (/Protect):** Protect bit

0: Enable

1: Disable

**Bit 1 (USB):** Operation mode

0: Non-USB mode. Close the UDC and Transceiver function.

1: USB mode



**Bit 2 (/D+ resistor):** D+ Resistor pull-high switch.

**0:** Enable internal USB D+ pull-high resistor.

**1:** Disable internal USB D+ pull-high resistor.

**Bits 4~3 (OST1~OST0):** Oscillator start time. WDT time-out period

**00:** 500  $\mu$ s

**01:** 2 ms

**10:** 4 ms

**11:** 8 ms

**Bits 6~5 (FRQ1~FRQ0):** System clock frequency switch.

**00:** 8 MHz, External Crystal  $\times$  2

**01:** 16 MHz, External Crystal  $\times$  4

**10:** 24 MHz, External Crystal  $\times$  6

**11:** Not Defined

**Bits 8~7 (PKG1~PKG0):** Package switch.

**00:** Reserved

**01:** Reserved

**10:** Reserved

**11:** 44 pins

**Bit 9 (Reverse):** Set to "0" as default value

**Bit 10 (/AD hold):** MCU Setting during AD conversion

**0:** Hold MCU during AD conversion.

**1:** Keep the MCU running during AD conversion.

**Bit 11 ~ Bit 12: Reverse**

**Bits 15 ~ 13 (EPX\_SEL2 ~ EPX\_SEL0):** Endpoint function select

EPX_SEL[2:0] (USB Mode)	EPX Status / EPX FIFO Max Size				
	EP1	EP2	EP3	EP4	EP5
000	Enable/64 byte	Disable/NA	Disable/NA	Disable/NA	Disable/NA
001	Enable/64 byte	Enable/64 byte	Disable/NA	Disable/NA	Disable/NA
010	Enable/64 byte	Enable/32 byte	Enable/32 byte	Disable/NA	Disable/NA
011	Enable/64 byte	Enable/32 byte	Enable/16 byte	Enable/16 byte	Disable/NA
100	Enable/64 byte	Enable/32 byte	Enable/16 byte	Enable/8 byte	Enable/8 byte
101	Enable/64 byte	Enable/16 byte	Enable/16 byte	Enable/16 byte	Enable/16 byte
110	Enable/32 byte	Enable/32 byte	Enable/32 byte	Enable/32 byte	Disable/NA
111	Enable/32 byte	Enable/32 byte	Enable/32 byte	Enable/16 byte	Enable/16 byte



**Bits 16 ~ 40:**

EPx Type:

- 00:** Not defined
- 01:** Reserved
- 10:** Bulk mode transfer
- 11:** Interrupt mode transfer

EPx Direction:

- 0:** Output way
- 1:** Input way

EPx Max Size:

- 00:** 8 bytes
- 01:** 16 bytes
- 10:** 32 bytes
- 11:** 64 bytes

If EPx Max Size select is larger than Endpoint function select, EPX FIFO size will depend on the size of the Endpoint function selector (EPX\_Sel[0~2]).

**Bits 41 ~ 42 (Reserved):** reserved bit

Default: 1

**Bits 43~ Bit 59 (USER ID):** Define by user.

**Bits 60~67 (Reserved):** reserved bit

**Bit 68 (Programmable D+ pull high resistor):**

- 0:** Enable
- 1:** Disable

## D Application Circuits

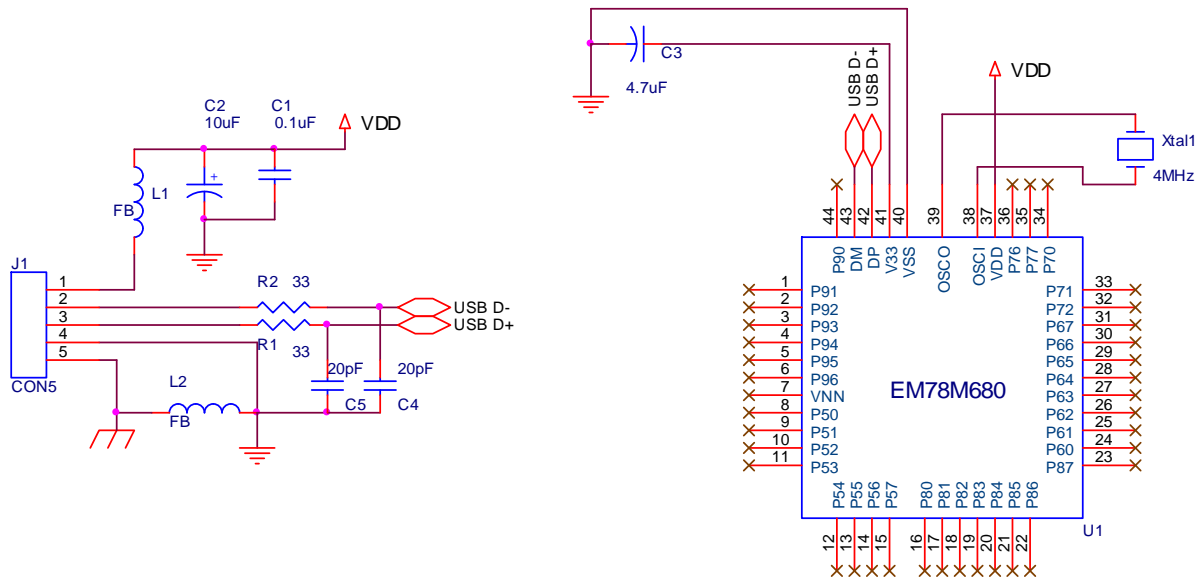


Figure D-1 EM78M680 Application Circuits