# EM78P613

## Universal Serial Bus Series Microcontroller

# Product Specification

**DOC. VERSION 1.2**

**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.



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 1st Road
Hsinchu Science Park
Hsinchu, TAIWAN 30076
Tel: +886 3 563-9977
Fax: +886 3 563-9966
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

**Shanghai:**
**ELAN Microelectronics Shanghai, Ltd.**
Rm101, #3 Lane 289, Bisheng Rd.,
Zhangjiang Hi-Tech Park
Pudong New Area, Shanghai,
CHINA 201204
Tel: +86 21 5080-3866
Fax: +86 21 5080-0273
elan-sh@elanic.com.cn

# Contents

# APPENDIX



## Specification Revision History

| Doc. Version | Revision Description | Date |
|---|---|---|
| 1.0 | Initial release version | 2009/01/09 |
| 1.1 | Upgrade Operation frequency 16M Hz and Package type | 2009/11/11 |
| 1.2 | Reduced Package types (removed DIP/QFN) | 2011/12/22 |

# 1 General Description

The EM78P613 is a series of 8-bit Universal Serial Bus RISC architecture, One-Time Programmable (OTP) microcontrollers. It is specifically designed for USB full speed device application and supports standard devices such as PS/2 mouse. The EM78P613 also supports one device address and three endpoints. With no firmware involved, this series of microcontrollers can automatically identify and decode Standard USB Command to Endpoint Zero.

The EM78P613 has 8-level stacks and 6 interrupt sources. It has 208 bytes of general purpose SRAM and 6K bytes of OTP ROM.

# 2 Features

- Operating voltage**:** 4.0V ~ 5.5V
- All GPIO are 3.3V
- Low-cost solution for full-speed USB devices, such as mouse, joystick, and Gamepad
- Complies with USB Specifications
  - 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 three endpoints
- USB Applications Compliance
  - P75 (D+) has an internal pull-high resistor (1.5 KΩ)
  - USB protocol handling
  - USB device state handling
  - Identifies and decodes Standard USB commands to EndPoint Zero
- PS/2 Application Support
  - Built-in PS/2 port interface for mouse
- Built-in 8-bit RISC MCU
  - 8-level stacks for subroutine nesting and interrupt
  - 8-bit real time clock/counter (TCC) with overflow interrupt
  - Six available interrupts
  - Built-in free running RC oscillator for Watchdog Timer and Dual clock mode
  - Two independent programmable prescalers for WDT and TCC
  - Two power saving methods:
    a) Power-down mode (Sleep mode)
    b) Dual clock mode

- Two clocks per instruction cycle
- One-time programmable (OTP)

■ I/O Ports
- Up to 4 LED sink pins
- Each GPIO pin has an internal programmable pull-high resistor (25KΩ)
- Port 60~66, P74 ~ P77 and P92~ P94 can wake up the MCU from Sleep mode by input state change

■ Internal Memory
- Built-in 6K×13 bits Program ROM
- Built-in 208 bytes general purpose registers (SRAM)
- Built-in USB Application FIFOs

■ Operation Frequency
- Normal Mode**:** MCU runs with an external oscillator frequency of 6 , 12 or 16 MHz
- Dual Clock Mode**:** MCU runs at a frequency of 256kHz (or 32kHz, 4kHz, 500Hz), emitted by the internal oscillator with the external crystal turned off to save power.

■ Built-in 3.3V Voltage Regulator
- For MCU power supply
- Pull-up source for the external USB resistor on D+pin

■ Package Type**:**
- 18-pin SOP (300mil)**:** EM78P613AM
- 20-pin SOP (300mil)**:** EM78P613BM
- 24-pin SOP (300mil)**:** EM78P613CM

# 3 Application

■ USB and PS/2 both compatible with mouse

■ USB Mouse

■ USB Joystick

# 4 Pin Assignment

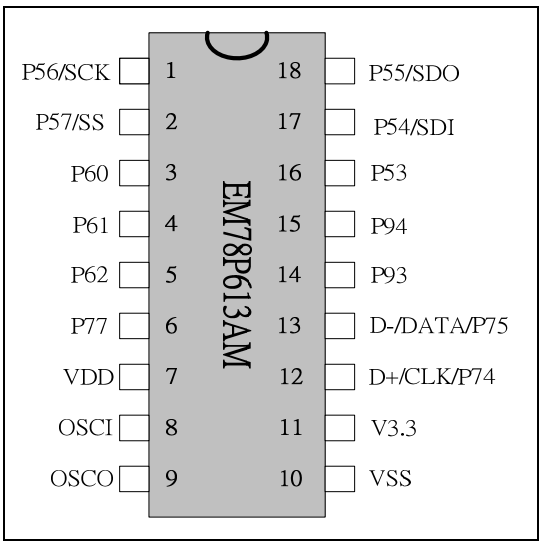



*Figure 4-1a EM78P613AM (18-Pin SOP)*

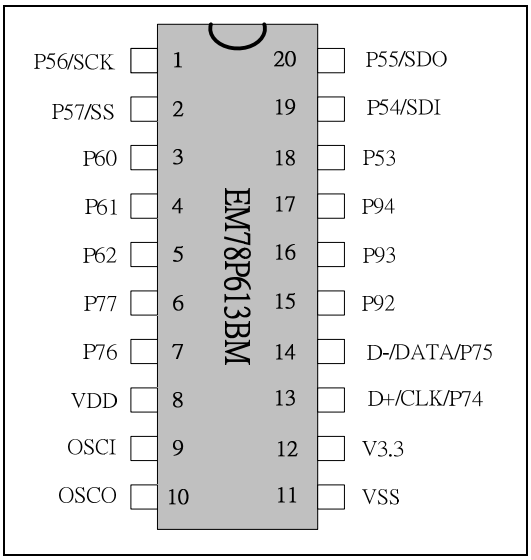



*Figure 4-1b EM78P613BM (20-Pin SOP)*

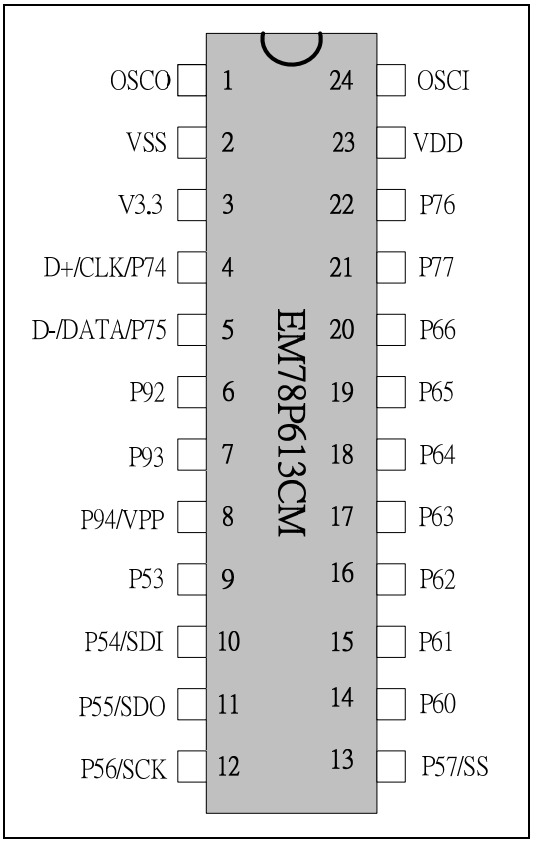



*Figure 4-1c EM78P613CM (24-Pin SOP)*

# 5 Pin Description

| Symbol | I/O | Function |
|---|---|---|
| P54 ~ P57 | I/O | Port 5 is a 4-bit 3.3V bi-directional I/O port. They can be pulled-high internally by software control.<br>For SPI function:<br>P54 SPI data Rx pin<br>P55 SPI data Tx pin<br>P56 SPI Clock<br>P57 SPI Slave Select Pin |
| P60 ~ P66 | I/O | Port 6 is a 7-bit 3.3V bi-directional I/O port. They can be pulled-high internally by software control. |
| P76 ~ P77 | I/O | LED sink pins<br>P76 ~ P77 are 3.3V bi-directional I/O ports. They can be pulled-high internally by software control. |
| P92 ~ P93 | I/O | P92~P93 are 3.3V bi-directional I/O ports. They can be pulled-high internally by software control. P92-P93 can be LED direct sink pins. |
| P94 / Vpp | I | Input only. OTP program pin. |
| D+/CLK/P74 | I/O | USB plus data line interface or CLK for PS/2 mouse.<br>When the EM78P613 is running in PS/2 mode, this pin will have an internally pulled high resistor (2.2KΩ), with VDD=5.0V.<br>When the EM78P613 is running in USB mode, this pin will have an internally pulled high resistor (1.5KΩ), with V33=3.3V. |
| D-/DATA/P75 | I/O | USB minus data line interface or DATA for PS/2 mouse.<br>When the EM78P613 is running in PS/2 mode, this pin will have an internally pulled high resistor (2.2KΩ), with VDD=5V. |
| OSCI | I | 6 MHz / 12/16 MHz Crystal input. |
| OSCO | O | Return path for 6 MHz / 12 MHz/16 MHz Crystal. |
| V3.3 | PWR | 3.3V regulator output. This pin has to be tied to a 4.7μF capacitor. |
| VDD | PWR | Power supply pin |
| GND | PWR | Ground pin |

# 6 Block Diagram

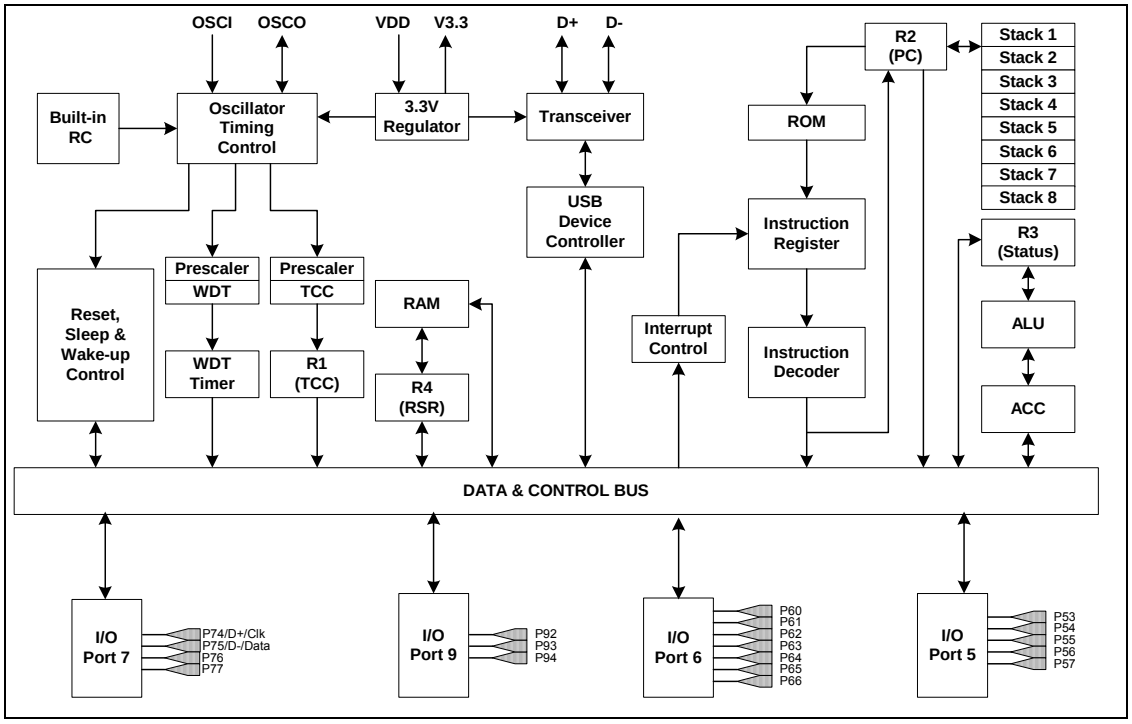



*Figure 6-1 EM78P613 Series Block Diagram*

# 7 Functional Description

The EM78P613 memory is organized into three spaces, namely**;**

- User Program memory in 6K×13 bits ROM space
- Data Memory in 208 bytes SRAM space
- USB Application FIFOs for EndPoint0, EndPoint1, and EndPoint2

## 7.1 Program Memory

The program space of the EM78P613 is 6K words, and is divided into six pages. Each page is 1K words long. After a reset, the 13-bit Program Counter (PC) points to location zero of the program space.

The Interrupt Vector is at 0x0001 and accommodates TCC interrupt, P74~P77 State Changed interrupt, EndPoint0 interrupt, USB Suspend interrupt, USB Reset interrupt, and USB Host Resume interrupt.

When interrupt occurs, the MCU will auto save to the Status Register (R3), RAM.

Select Register (R4), R8.7, Accumulator (A), then clear PS0~PS2 and fetch the next instruction from the corresponding address as illustrated in the following diagram.

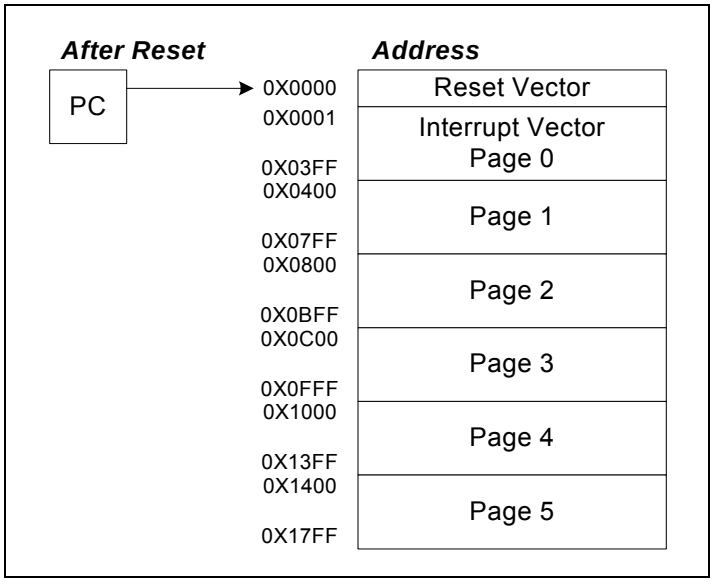



*Figure 7-1 Functional Map Diagram of EM78P613 Program Memory*

When executing “RETI” instruction, the MCU will pop A, R3, R4, R8.7and stack, and enable interrupt.

## 7.2 Data

The Data Memory has 208 bytes SRAM space. It also has an on-chip USB Application FIFO space for USB Application. Figure 7-1 below shows the organization of the Data Memory Space.

### *7.2.1 Special Purpose Register*

When the microcontroller executes instructions, specific registers are implemented to ensure proper operation of essential functions, such as Status Register which records the calculation status, Port I/O Control Registers which control the I/O pins' direction, etc. Lots of other special purpose registers are provided for various functions.

Note that Special Control Registers can only be read or written to by two instructions: IOR and IOW.

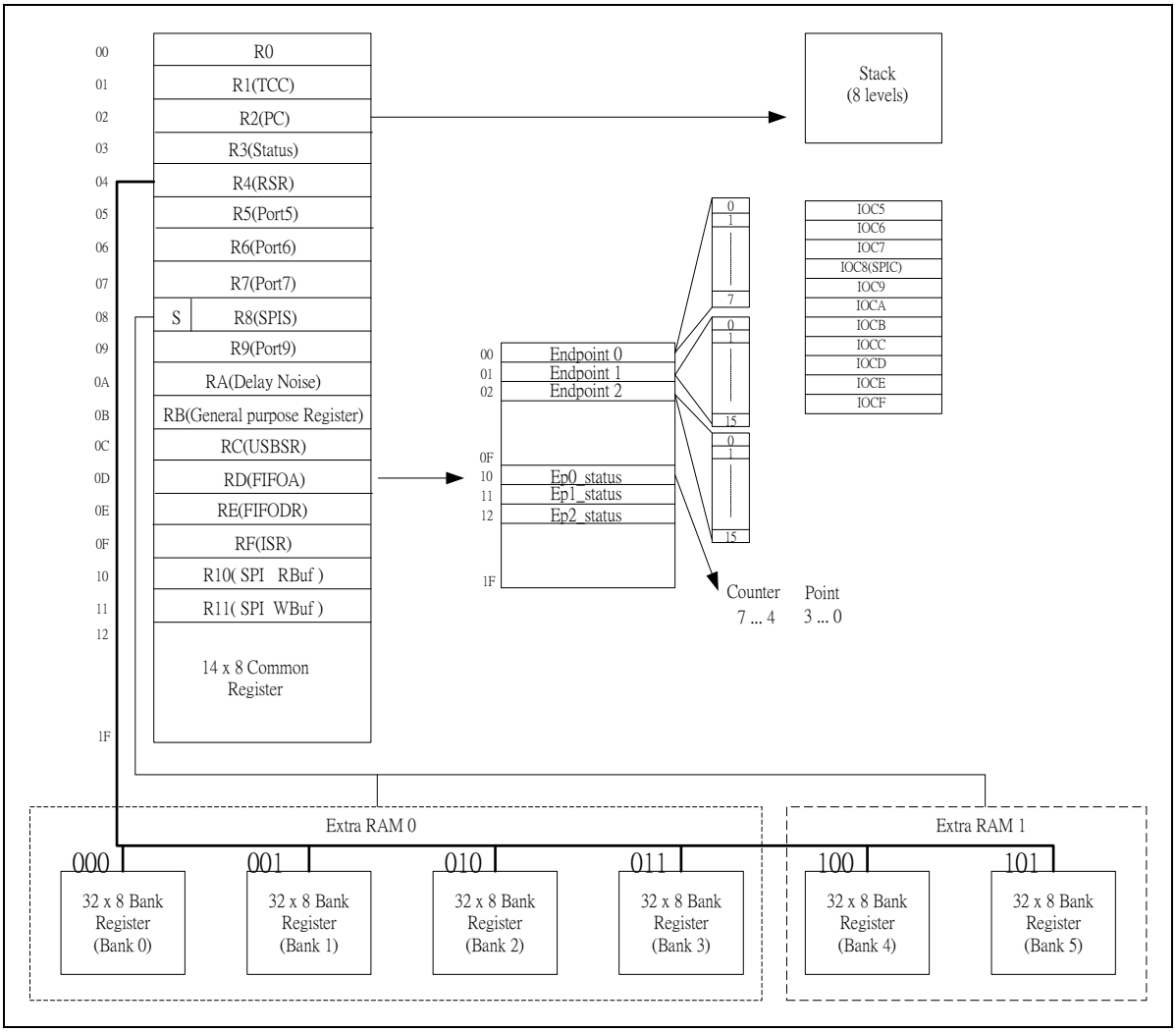



*Figure 7-2 EM78P613 Data RAM Organization*

### *7.2.2 Operation Registers*

The following subsections describe each of the Operation Registers of the Special Purpose Registers. The Operation Registers are arranged according to the order of the registers' address. Note that some registers are read only, while others are both readable and writable.

#### 7.2.2.1 R0 (Indirect Addressing Register) *Default Value: (0B_0000_0000)*

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

#### 7.2.2.2 R1 (Timer / Clock Counter) *Default Value: (0B_0000_0000)*

The TCC register is an 8-bit timer or counter. It is readable and writable as any other register. The Timer module will be incremented after execution of every instruction cycles. User can work around this by writing an adjusted value. The Timer interrupt is generated when the R1 register overflows from FFh to 00h. This overflow sets the TCIF (RF[0]) bit. The interrupt can be masked by clearing bit TCIE (IOCF[0]). After Power-on reset and Watchdog reset, the initial value of this register is 0x00.

#### 7.2.2.3 R2 (Program Counter and Stack) *Default Value: (0B_0000_0000)*

The EM78P613 Program Counter is a 13-bit register that allows accessing of the 6k words of the Program Memory with 8 level stacks. The eight LSB bits, A0~A7, are located at R2, while the three MSB bits, A12~A10, are located at R3. The Program Counter is cleared after Power-on reset or Watchdog reset. The first instruction that is executed after a reset is located at Address 00h.

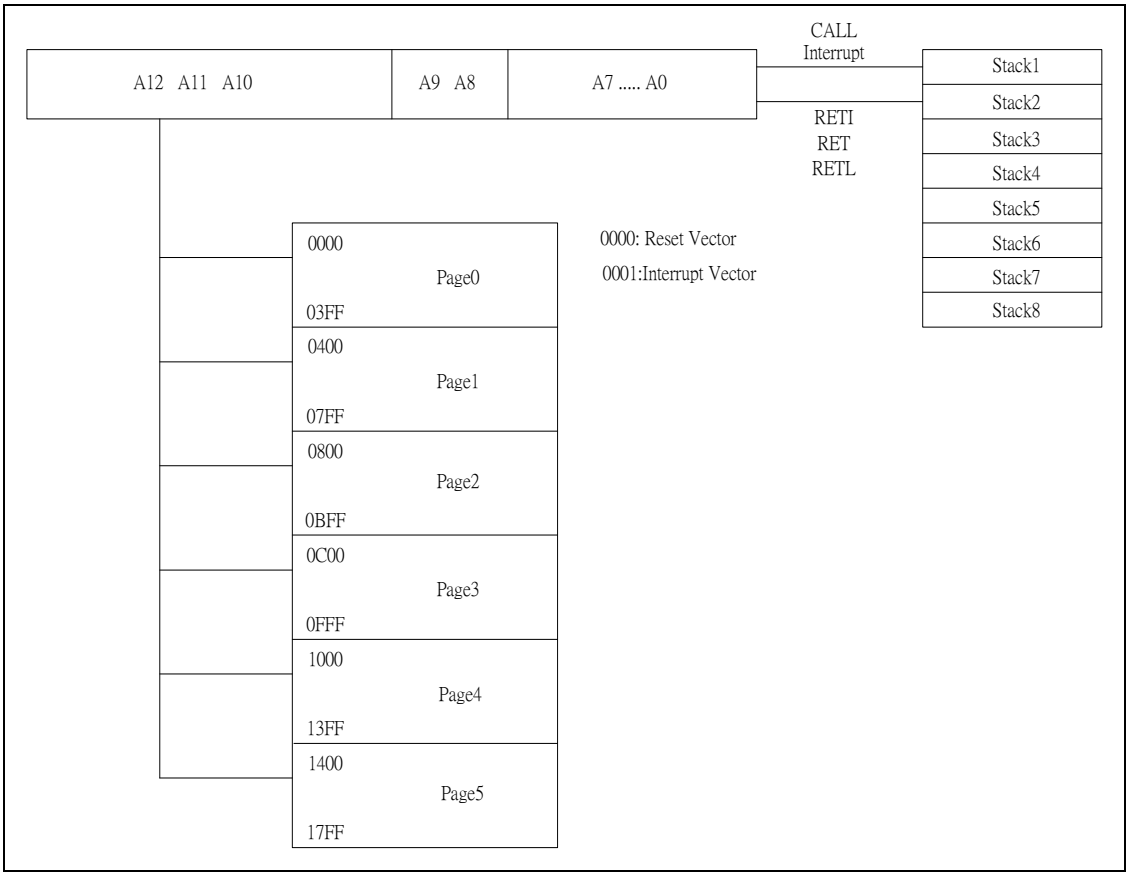



*Figure 7-3 Program Counter and Stack*

### 7.2.2.4 R3 (Status Register) *Default Value: (0B_0001_1XXX)*

| 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/Borrow Flag

**0 :** No carry-out from the result's Most Significant bit

**1 :** A carry-out from the result's Most Significant bit occurred

**NOTE**

*For Borrow, the polarity is reversed. For rotate (RRC, RLC) instructions, this bit is loaded with either high or low-order bit of the source register.*

**R3 [1] :** Auxiliary Carry/Borrow Flag. For ADD, SUB Instructions

**0 :** No carry-out from the 4th low-order bit of the result

**1 :** A carry-out from the 4th low-order bit of the result occurred

**NOTE**

*For Borrow, the polarity is reversed.*

**R3 [2] :** Zero flag. It will be set to **1** when the result of an arithmetic or logic operation is zero.

**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

**1 :** Power-on

**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 overflow occurs

**1 :** No Watchdog timer overflow

The various states of Power down flag and Time-out flag at different conditions are shown below:

| T | P | Condition |
|---|---|---|
| 1 | 1 | Power-on reset |
| 1 | 1 | WDTC instruction |
| 0 | *P | WDT time-out |
| 1 | 0 | Power down mode |
| 1 | 0 | Wake up caused by port change during Power down mode |

*P: *Previous status before WDT reset*

**R3 [5-7] :** Page selection bits. These three bits are used to select the program memory page.

| PS2 | PS1 | PS0 | Program Memory Page [Address] |
|---|---|---|---|
| 0 | 0 | 0 | Page 0 [0000-03FF] |
| 0 | 0 | 1 | Page 1 [0400-07FF] |
| 0 | 1 | 0 | Page 2 [0800-0BFF] |
| 0 | 1 | 1 | Page 3 [0C00-0FFF] |
| 1 | 0 | 0 | Page 4 [1000-13FF] |
| 1 | 0 | 1 | Page 5 [1400-17FF] |

#### 7.2.2.5 R4 (RAM Select Register) *Default Value: (0B_00XX_XXXX)*

| 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 (RAM select register) contains the address of the registers.

**R4 [0~5] :** Used to select registers in 0x00h~0x3Fh. The Address 0x00~0x1F is a common space. After 0x1Fh, SRAM is divided into four banks, using Bank Select Register.

**R4 [6, 7] :** Used to select the registers bank (refer to the table below). The following are two examples:

1) R4=00001100 and R4=10001100 point to the same Register 0x0Ch. Since 0x0Ch is in the common space, Bit 6 and Bit 7 are meaningless.
2) R4=10111100 points to Register 0x3C in Bank 2.

| R8[7]Extr_R | R4[7]Bk1 | R4[6]Bk0 | RAM Bank # |
|---|---|---|---|
| 0 | 0 | 0 | Bank 0 |
| 0 | 0 | 1 | Bank 1 |
| 0 | 1 | 0 | Bank 2 |
| 0 | 1 | 1 | Bank 3 |
| 1 | 0 | 0 | Bank 4 |
| 1 | 0 | 1 | Bank 5 |

#### 7.2.2.6 R5 (Port 5 I/O Register) *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 | – | – | – | – |

#### 7.2.2.7 R6 (Port 6 I/O Register) *Default Value: (0B_0000_0000)*

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

#### 7.2.2.8 R7 (Port 7 I/O Register) *Default Value: (0B_0000_X000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| P77 | P76 | D- / P75 / DATA | D+ / P74 / CLK | – | – | – | – |

#### 7.2.2.9 R8 SPIS (SPI Status Register) *Default Value:(0B_0000_1000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Extr_R | SRO | SSE | RBF | /P5 Wake-up | – | – | Set_Config |

**R8[0] :** Set configuration flag. Read only. Set when the value of “set configuration” command is a non-zero.

The hardware clears this bit either when the host “set configuration value” is zero or UDC receives a reset signal.

**R8[1,2] :** Reserved

**R8[3] :** Port 5 wake-up function

**0 :** Enable

**1 :** Disable

**NOTE**

*You must save the Port 5 input pin state before enabling the Port 5 wake-up function.*

**R8[4] :** RBF (Read Buffer Full flag)

**0 :** Receiving not completed yet, and the SPIRB is not fully updated.

**1 :** Receiving completed; SPIRB is fully updated.

**NOTE**

*This bit is valid in Master mode.*

**R8[5] :** SSE (SPI Shift Enable bit)

**0 :** Reset as soon as the shifting is completed, and the next transmission is ready to shift.

**1 :** Start to shift, and keep at **1** while the current transmission is still being transmitted.

**NOTE**

*This bit is reset to 0 by the hardware at every transmission*

**R8[6] :** SRO (SPI Read Overflow bit)

**0 :** No overflow

**1 :** A new data is received while the previous data is still being held in the SPIB register. In this state, the data in the SPIS register will be destroyed. To avoid setting this bit, you should read the SPIRB register even if only transmission is implemented.

**NOTE**

*This can only occur in Slave mode.*

**R8[7] :** Extr_R (Extra RAM block switch)

**0 :** Bank 0~Bank 3 (Extra RAM0)

**1 :** Bank 4~Bank 5 (Extra RAM1)

#### 7.2.2.10 R9 (Port 9 I/O Register) *Default Value: (0B_000X_0000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | P93 | P92 | – | – |

#### 7.2.2.11 RA (Reserved) *Default Value: (0B_0000 0000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | Delay noise | – | – | – | – |

**RA[0~3; 5~7]:** Reserved

**RA[4] :** Delay noise. Set by hardware and cleared by software.

**0 :** VDD noise never drop over 2V

**1 :** VDD noise drop over 2V

**NOTE**

*During MCU reset (including Power-on reset and Watchdog reset), RA[4] is reset to "0" .*

#### 7.2.2.12 RC (USB Application Status Register) *Default Value: (0B_0000_0000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| EP0_W | EP0_R | EP1_R | EP2_R | EP2_W | Host_Suspend | EP0_Busy | Stall |

**RC [0] :** Stall flag. While the MCU receives an unsupported command or invalid parameters from host, this bit is 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 automatically cleared. This bit is readable and writable.

**RC [1] :** EP0_Busy flag. When this bit is equal to **1**, it indicates that the UDC is in the process of writing data into the EP0's FIFO or reading data from it. During this time, the firmware will avoid accessing the FIFO until the UDC finishes writing or reading. This bit is readable only.

**RC [2] :** Host 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 there is a bus activity. This bit is readable only.

**RC [3] :** EP2_W flag. This bit is set when the UDC receives a successful data from the USB Host to EP2. Upon detecting that this bit is equal to **1**, the firmware will execute a read sequence to the EP2's FIFO, after which this bit is cleared. Otherwise, the subsequent data from the USB host will not be accepted by the UDC.

**RC [6, 5, 4]** EP0_R / EP1_R / EP2_R flag. These three bits inform the UDC to read the data from the FIFO. Then the UDC will automatically send the data to the Host. After the UDC finishes reading the data from the FIFO, these bits are cleared automatically.

Therefore, before writing data into the FIFO, the firmware will first check these bits to avoid overwriting the data. These three bits can only be set by firmware and cleared by hardware.

**RC [7] :** EP0_W flag. 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.

Therefore, before the firmware can write data into the FIFO, this bit must first be set by the firmware to prevent the UDC from writing data at the same time. This bit is both readable and writable.

#### 7.2.2.13 RD (USB Application FIFO Address Register)

*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 |

**RD [0~4]** USB Application FIFO address registers. These five bits are the address pointers of USB Application FIFO.

**RD [5~7]** Undefined registers. The default value is **0**.

#### 7.2.2.14 RE (USB Application FIFO Data Register)

*Default Value: (0B_0000_0000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| UD7 | UD6 | UD5 | UD4 | UD3 | UD2 | UD1 | UD0 |

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

**NOTE**

*For example, if you want to read the fourth byte of EndPoint Zero, you need to use the address of EP0 (0x00) and the Data Byte Pointer of EP0 (0x10) to access it.*

```
// Read the 4th byte of the EP0 FIFO
// First, assign the data byte pointer of the EP0 register (0X10) with 0X03.
   MOV A, @0X10
   MOV RD, a          // Move data in A to RD register
   MOV A, @0X03
   MOV RE , A         // Move data in A to RE register

// Then read the content from EP0 FIFO (0x00) 4th byte
   MOV A, @0X00
   MOV RD, A          // Assign an address point to EP0 FIFO
   MOV A, RE          // Read the fourth byte data (Byte 3) of the EP0 FIFO
   MOV A, 0X0E        // Read the fifth byte data (Byte 4) of the EP0 FIFO
```

#### 7.2.2.15 RF (Interrupt Status Register) *Default Value: (0B_0000_0000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| USB Host Resume_IF | – | SPIIF | Port7 state change_IF | USB Reset_IF | USB Suspend_IF | EP0_IF | TCC_IF |

**RF [0] :** TCC Overflow interrupt flag. It will be set while TCC overflows, and is cleared by firmware.

**RF [1] :** EndPoint Zero interrupt flag. It will be set when the EM78P613 receives the Vendor/Customer Command to EndPoint Zero. This bit is cleared by firmware.

**RF [2] :** USB Suspend interrupt flag. It will be set when the EM78P613 finds the USB Suspend Signal on the USB bus. This bit is cleared by firmware.

**RF [3] :** USB Reset interrupt flag. It will be set when the host issues a USB Reset signal.

**RF [4] :** P74/P75 (PS2 only) /P76/P77 (USB and PS2) Port state change interrupt flag.

In PS2 Mode, only pins configured as input can cause this interrupt to occur. These pins (P74, P75, P76 and P77) are compared with the value latched on the last read of Port 7.

In USB Mode, only P76 and P77 have this function.

**NOTE**

*RF[4]: Port State Change Interrupt Flag*

***EM78M611***

*USB Mode – P76 and P77 DO NOT have interrupt function.*

*PS2 Mode – P74/P75/P76/P77 have interrupt function.*

***EM78P613***

*P74/P75/P76/P77 all have interrupt function in both USB and PS2 modes*

**RF [5] :** (SPI_IF) SPI interrupt flag. Set when data transmission is completed. Cleared by software. The interrupt is created after the SPI data is saved to the SRAM.

**RF [6] :** Reserved

**RF [7] :** USB Host Resume interrupt flag. It will be set only in Dual Clock mode when the USB suspend signal becomes low.

**NOTE**

*RF [7]: USB Host Resume_IF*

*In **EM78M611**, this bit is always '**1**'. It is **not** available for use.*

*In **EM78P613**, this bit is available for use.*

#### 7.2.2.16 R10 SPIRB (SPI Read Register) *Default Value: (0b00000000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| SRB7 | SRB6 | SRB5 | SRB4 | SRB3 | SRB2 | SRB1 | SRB0 |

SRB0~SRB7 are the 8-bit data when there's complete transmission by SPI.

#### 7.2.2.17 R11 SPIWB (SPI Write Register) *Default Value: (0b0000_0000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| SWB7 | SWB6 | SWB5 | SRW4 | SWB3 | SWB2 | SWB1 | SWB0 |

SWB0~SWB7 are the 8-bit data that are waiting for transmission by SPI.

#### 7.2.2.18 RB, R12~R1F, and R20~R3F (including Banks 0~5) are General Purpose Registers

### *7.2.3 Control Registers*

Some special purpose registers are available for special control purposes. Except for the Accumulator (ACC), these registers must be read and written with special instructions. One of these registers, CONT, can only be read by the instruction "CONTR" and written by "CONTW" instruction. The other special control registers can be read by the instruction "IOR" and written by the instruction "IOW".

The following sections describe only the general functions of the control registers.

#### 7.2.3.1 A (Accumulator)

Internal Data Transfer operation, or Instruction Operand Holding usually involves the temporary storage function of the Accumulator. The Accumulator is an 8-bit register that holds operands and the results of arithmetic calculations. It is not addressable. After an interrupt occurs, the Accumulator is auto-saved by hardware.

#### 7.2.3.2 CONT (Control Register) *Default Value: (0B_0011_1111)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| LED | /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 by the instruction "CONTW".

**CONT [0~2] :** Watchdog Timer prescaler bits. These three bits are used as the Watchdog Timer prescaler (WDT base time = 8 ms).

**CONT [3~5] :** TCC Timer prescaler bits.

The relationship between the prescaler value and these bits are shown below:

| PSR2/TSR2 | PSR1/TSR1 | PSR0/TSR0 | TCC Rate | WDT Rate |
|---|---|---|---|---|
| 0 | 0 | 0 | 1: 2 | 1: 1 |
| 0 | 0 | 1 | 1: 4 | 1: 2 |
| 0 | 1 | 0 | 1: 8 | 1: 4 |
| 0 | 1 | 1 | 1: 16 | 1: 8 |
| 1 | 0 | 0 | 1: 32 | 1: 16 |
| 1 | 0 | 1 | 1: 64 | 1: 32 |
| 1 | 1 | 0 | 1: 128 | 1: 64 |
| 1 | 1 | 1 | 1: 256 | 1: 128 |

**NOTE**

*The WDT Timing base is 8 ms.*

*Ex. Prescaler = 1:128*

*WDT overflow time is: 8 ms $\times$ $2^7$ = 1024 ms*

**CONT [6] :** Interrupt enable control bit. This bit toggles the Interrupt function between enable and disable. It is set to **1** by the Interrupt Disable instruction "DISI" and reset by the Interrupt Enable instructions "ENI" or "RETI."

**0 :** Enable the Interrupt function

**1 :** Disable the Interrupt function

**CONT [7] :** LED bit. This bit is used to enable the P76 and P77 LED sink capacity.

**0 :** Disable the P76, P77 LED sink capacity

**1 :** Enable the P76, P77 LED sink capacity

#### 7.2.3.3 IOC5, IOC6, IOC7, & IOC9 I/O (Port Direction Control Registers)

*Default Value: (0B_1111_1111)*

These are I/O port (Port 5 ~ Port 9, excluding Port 8) direction control registers. Each bit controls the I/O direction of the 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 become outputs when the relative control bits are cleared.

**0 :** Output direction

**1 :** Input direction

#### 7.2.3.4 IOC8: (SPIC) SPI Control Register *(Default Value :0b00000000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| CES | SPIE | OD3 | OD4 | – | SRB2 | SRB1 | SRB0 |

**IOC8 [0~2] :** SBRS (SPI Baud Rate Select bits)

SPI baud rate table

| 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 | Master | Fosc/64 |
| 1 | 1 | 0 | Slave | /SS enable |
| 1 | 1 | 1 | Slave | /SS disable |

**IOC8 [3] :** Reserved

**IOC8 [4] :** OD4 (Open-drain Control bit)

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

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

**IOC8 [5] :** OD3 (Open-drain Control bit)

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

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

**IOC8 [6] :** SPIE (SPI Enable bit)

**0 :** Disable SPI mode

**1 :** Enable SPI mode

**IOC8 [7] :** CES (Clock Edge Select bit)

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

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

### 7.2.3.5 IOCA (Operation Mode Control Register) *Default Value: (0B_1110_0011)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Dual_Frq.1 | Dual_Frq.0 | /P76,/P77 Pull high | Remote_ Wake up | – | – | PS/2 | USB |

**IOCA [0, 1] :** Two bits used to select the operation mode.

| IOCA[1] | IOCA[0] | Operation Mode |
|---|---|---|
| 0 | 0 | Detect Mode |
| 0 | 1 | USB Mode |
| 1 | 0 | PS/2 Mode |
| 1 | 1 | USB Test Mode |

**IOCA[2~3] :** Reserved

**IOCA[4] :** Indicate whether the device is currently requested to support Remote Wake-up or not. The Remote Wake-up field can be modified by "SetFeature ()" and "ClearFeature ()" requests.

**0 :** Do Not support remote wake-up

**1 :** Supports remote wake-up

**NOTE**

*IOCA[4]: Remote_Wake up bit*

***EM78M611** does **NOT** support this function. Only **EM78P613**, **EM78P611F**, **EM78611E**, and **EM78M611E** do.*

**IOCA[5] :** Pull-high resistor of P77 and P76. This is applicable USB mode only.

**0 :** Pull-high is enabled

**1 :** Pull-high is disabled

**NOTE**

*IOCA[5]: /P76, /P77 Pull-high bit*

*The previous versions of EM78M611/EM78611 do **NOT** support this function. Only the **EM78P613, EM78611E, & EM78P611**F support this function in USB mode.*

**IOCA [6, 7] :** Select the operation frequency in Dual Clock Mode. Four frequencies are available and can be selected as Dual Clock Mode in running the MCU program.

| Dual_Frq.1 | Dual_Frq.0 | Frequency |
|---|---|---|
| 0 | 0 | 500Hz |
| 0 | 1 | 4kHz |
| 1 | 0 | 32kHz |
| 1 | 1 | 256kHz |

#### 7.2.3.6 IOCB (Port 9 Wake-up Pin Select Register) *Default Value: (0B_X111_1111)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | /P94 | /P93 | /P92 | – | – |

**IOCB [2~4] :** These bits are used to select which of the Port 9 pins is to be assigned to wake-up the MCU while in Power down mode.

**0 :** Enable the function

**1 :** Disable the function

**IOCB [5~7; 0~1] :** Reserved

#### 7.2.3.7 IOCC (Port 9 LED Sink Capacity Control Register)

*Default Value: (0B_000X_0000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | P93 | P92 | – | – |

**IOCC [2~3] :** LED sink control bit. These bits are used to enable the LED sink capacity of P92 ~ P93

**0 :** Disable the LED sink capacity of the respective pin

**1 :** Enable the LED sink capacity of the respective pin

**IOCC [7~4;1~0] :** Reserved bits

#### 7.2.3.8 IOCD (Port 9 Pull-high Control Register) *Default Value: (0B_111X_1111)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| – | – | – | – | /PH93 | /PH92 | – | – |

**IOCD [2~3] :** These bits control the 25KΩ pull-high resistor of individual pins in Port 9.

**0 :** Enable the pull-high function

**1 :** Disable the pull-high function

**IOCD [4~7; 0~1] :** Reserved bits

#### 7.2.3.9 IOCE (Special Function Control Register) *Default Value: (0B_1101_0111)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| /Dual clock | /WUE | WTE | RUN | Device_Resume | – | /PU6 | /PU5 |

**IOCE [0, 1] :** Port 5 and Port 6 pull-high control bits.

**0 :** Enable

**1 :** Disable

**IOCE [2] :** Reserved

**IOCE [3] :** Setting this bit will allow the UDC to execute Resume signaling. 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. It can only be used in Dual clock mode when the USB suspend signal becomes low.

**NOTE**

*IOCE[3]: Device_Resume bit*

*In **EM78M611**, this bit is always '**0**'. It's use is **not** available.*

*In **EM78P613**, this bit is available for use.*

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

**0 :** Sleep mode. The EM78P613 is in power down mode.

**1 :** Run mode. The EM78P613 is in normal working mode.

**IOCE [5] :** Watchdog Timer enable bit. The bit disables/enables the Watchdog Timer.

**0 :** Disable WDT

**1 :** Enable WDT

**IOCE [6] :** Enable the wake-up function as triggered by a port-changed status. This bit is set by UDC.

**0 :** Enable the wake-up function

**1 :** Disable the wake-up function

**IOCE [7] :** Dual clock Control bit. This bit is used to select the frequency of the system clock. When this bit is cleared, the MCU will run on very low frequency for power saving and the UDC will stop working.

**0 :** Selects the EM78P613 to run on slow frequency

**1 :** Selects the EM78P613 to run on normal frequency

#### 7.2.3.10 IOCF (Interrupt Mask Register) *Default Value: (0B_0000_0000)*

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| USB Host Resume_ IE | – | SPI_IE | Port 7 state change_1E | USB Reset_IE | USB Suspend_IE | EP0_IE | TCC_IE |

**IOCF [0~7]** TCC / EP0 / USB Suspend / USB Reset / Port 7 State Change /SPI /USB Host Resume interrupt enable bits. These eight bits control the TCC interrupt function, EP0 interrupt, USB Suspend interrupt, USB Reset interrupt, Port 7 State Change interrupt, SPI interrupt, and USB Host Resume interrupt respectively. Individual interrupt is enabled by setting its associated control bit in the IOCF to **1**.

**0** : Disable Interrupt

**1** : Enable Interrupt

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

The USB Host Resume Interrupt works only in Dual clock mode. This is because when the MCU is in Sleep mode, it will be automatically awaken by the UDC Resume signal.

## 7.3 USB Application FIFO

For USB Application, the EM78P613 provides an 8-byte First-In-First-Out (FIFO) buffer for endpoint0, and provides two 16-byte FIFO buffer for Endpoint1 and Endpoint2. The buffer cannot be accessed directly. However, a corresponding Data Byte Pointer register for each endpoint is made available to address the individual byte of the FIFO buffer. The content of the individual byte will map to a special register.

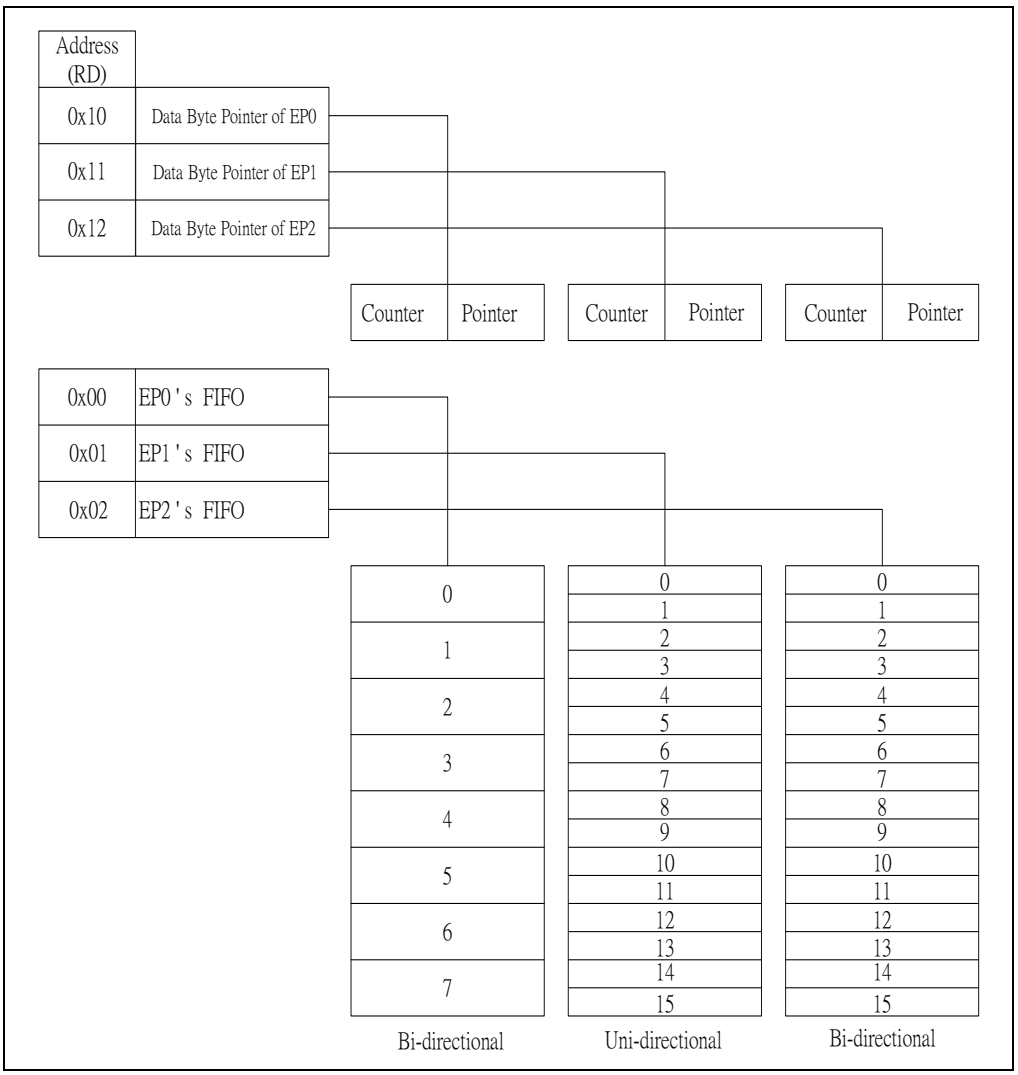



*Figure 7-4 Functional Map Diagram of USB Application FIFO*

## 7.4 USB Application

The EM78P613 is designed specially for USB device application and has many powerful functions that support the firmware to free itself from complex situation in various aspects of USB application.

### *7.4.1 USB Device Controller*

The EM78P613 has a built-in USB Device Controller (UDC) which 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 energy in dealing with USB transaction.

The UDC handles and decodes most Standard USB commands defined in the USB Specification Rev1.1. If the UDC receives an unsupported command, it will set a flag to notify the MCU of the receipt of such command. The Standard Commands that the EM78P613 supports includes; **Clear Feature, Get Configuration, Get Interface, Get Status, Set Address, Set Configuration, Set Feature, and Set Interface**.

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

Hence, the EM78P613 is very flexible under USB application since the developer can freely choose the appropriate method of decoding the USB command as dictated by varying conditons.

### *7.4.2 Device Address and Endpoints*

The EM78P613 supports one device address and three endpoints, i.e., EP0 for control endpoint, EP1 and EP2 for interrupt endpoints. Sending data to the USB host in EM78P613 is very easy. Just write data into the EP's FIFO, set the flag, and the UDC will handle the rest. It will then confirm that the USB host has received the correct data from EM78P613.

## 7.5 Reset

The EM78P613 provides three types of reset:

1) Power-on Reset

2) Watchdog Reset

3) USB Reset

### 7.5.1 Power-on Reset

Power-on Reset occurs when the device is attached to power and a reset signal is initiated. The signal will last until the MCU becomes stable. After a Power-on Reset, the MCU enters into the following predetermined states 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 values.

### 7.5.2 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.

### 7.5.3 USB Reset

When the UDC detects a USB Reset signal on the USB Bus, an MCU interrupt occurs, after which it 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.

## 7.6 Power Saving Mode

The EM78P613 provides two options of power-saving modes for energy conservation, i.e., Power Down mode and Dual Clock mode.

### 7.6.1 Power Down Mode

The EM78P613 enters into Power Down mode by clearing the RUN register (IOCE[4]). During this mode, the oscillator is turned off and the MCU enters Sleep mode. It will wake-up when a signal from the USB host is resumed, or when a Watchdog Reset occurs or when an input port state changes.

If the MCU wakes up when I/O port status changes, the direction of the I/O port should be set at input direction, and then read the state of port. For example**:**

```
:
// Set the Port 6 to input port
MOV      A, @0xFF
IOW      PORT 6
// Read the state of Port 6
MOV      PORT 6, PORT 6
// Clear the RUN bit
IOR      0xE
AND      A, 0B11101111
IOW      0xE
:
:
```

### 7.6.2 Dual Clock Mode

The EM78P613 has one internal oscillator for power saving application. Clearing the Bit IOCE [7] 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 frequency are available for selection by setting bits IOCA [6, 7] (see *Section 7.2.3.5* for details).

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 Host Resume signal is detected on USB Bus.

## 7.7 Interrupt

The EM78P613 has one interrupt vector in 0x0001. When an interrupt occurs during an MCU program run, it will jump to the interrupt vector (0x0001) and execute the instructions sequentially from the interrupt vector. RF is the interrupt status register, which records the interrupt status in the relative flags/bits.

The interrupt condition could be one of the following**:**

**TCC Overflow:** When the Timer Clock / Counter Register (R1) overflows, the status flag RF[0] will be set to **1**. Its Interrupt Vector is 0X0001.

**Port 7 State Change:** When the input signals in Port 7 changes, the status flag RF[4] will be set to **1**. Its Interrupt Vector is 0X0001.

**EP0 Interrupt:** When the UDC successfully accepts a setup transaction from host to EndPoint0, the status flag RF[1] is set to **1**. Its Interrupt Vector is 0X0001.

**USB Suspend:** When UDC detects a USB Suspend signal on the USB bus, the status flag RF[2] is set to **1**. Its Interrupt Vector is 0X0001.

**USB Reset:** When the UDC detects a USB Reset signal on the USB bus, the status flag R[3] is set to **1**. Its Interrupt Vector is 0X0001.

**USB Host Resume:** When the UDC detects that the USB bus is no longer in Suspend condition and without Device Resume signal, the status flag R[7] is set to **1**. Its Interrupt Vector is 0X0001.

IOCF is an interrupt mask register which can be set bit by bit. While their respective bit is written to **0**, the hardware interrupt will be inhibited, that is, the EM78P613 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 other interrupt.

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

## 7.8 Serial Peripheral Interface (SPI) Function

### *7.8.1 Overview*

Figures 7-5a and 7-5b below show how the EM78P613 communicates with other devices through an SPI module. If EM78P613 is a master controller, it sends clock through the SCK pin. A couple of 8-bit data are transmitted and received at the same time.

However, if the EM78P613 is defined as a slave, its SCK pin could be programmed as an input pin. Data will continue to be shifted based on both the clock rate and the selected edge.

### *7.8.2 Features*

- Operation in either Master mode or Slave mode
- 3-wire or 4-wire synchronous communication, that is, full duplex
- Programmable baud rates of communication
- Programming clock polarity
- Interrupt flag available for a read buffer full
- SPI transmission order
- Up to 6 MHz (maximum) bit frequency MSB

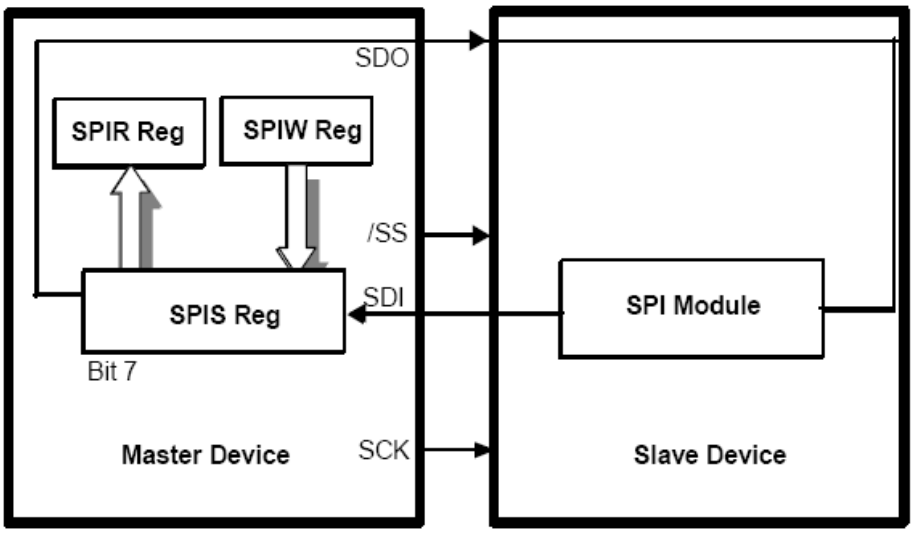



*Figure 7-5a SPI Master/Slave Communication*

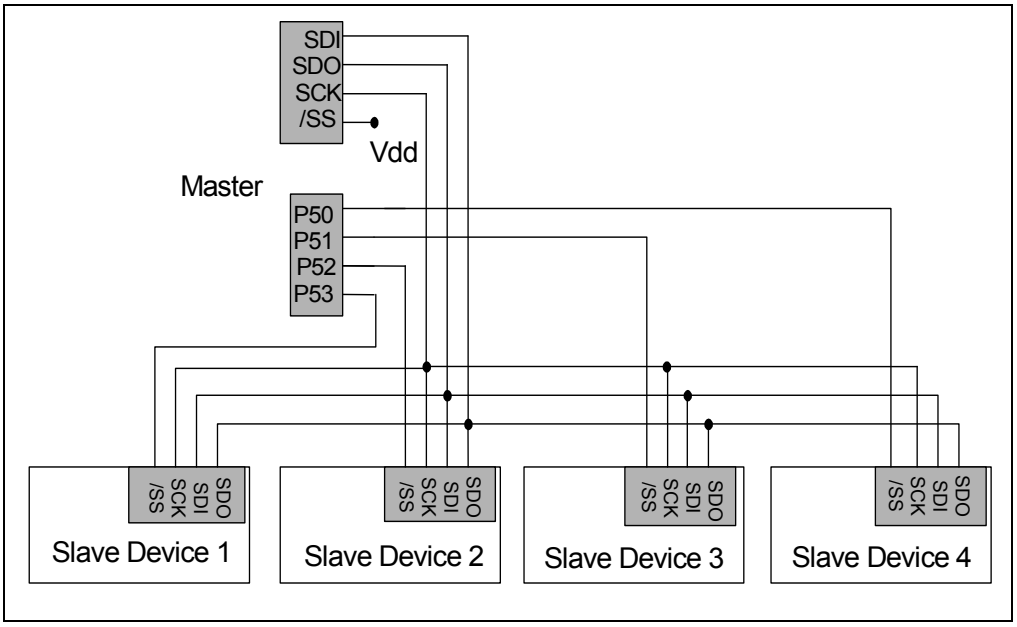



*Figure 7-5b SPI Configuration as Single-Master and Multi-Slave*

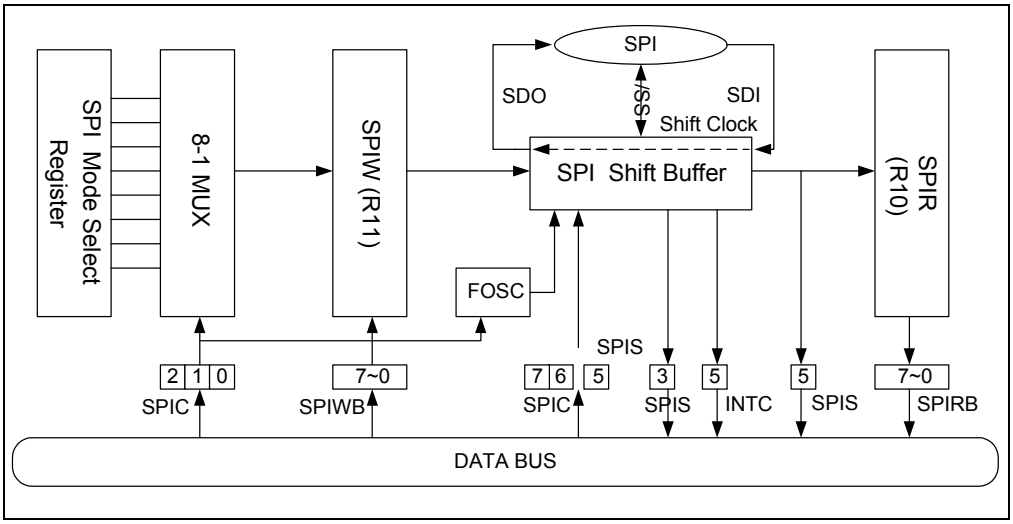



*Figure 7-5c Functional Block Diagram of SPI Transmission*

The following describes the function of each block and explains how to carry out the SPI communication with the signals depicted in Figure 7-5c above**:**

- **P54/SDI:** Serial Data In
- **P55/SDO:** Serial Data Out
- **P56/SCK:** Serial Clock
- **P57/SS:** /Slave Select (Option). This pin (/SS) may be required in a Slave mode.

- **RBF:** Set by Buffer Full Detector and reset by software.
- **Buffer Full Detector:** Set to **1** when an 8-bit data shifting is completed.
- **SSE:** Loads the data in SPIW register, and begin to shift.
- **SPIS reg:** Shift byte in and out. The MSB is shifted first. Both the SPI IN FIFO data and the SPI OUT FIFO data are loaded at the same time. Once data are written, the SPIS starts transmission/reception. The received data will be moved to the SPI IN FIFO register as soon as the shifting of the 8-bit data is completed. The RBF (Read Buffer Full) flag and the RBFIF (Read Buffer Full Interrupt) flag are then set.
- **SPIR reg:** Read buffer. The buffer will be updated as the 8-bit data shifting is completed. The data must be read before the next reception is completed. The RBF flag is cleared as soon as the SPI IN FIFO data is updated.
- **SPIW reg:** Write buffer. The buffer will ignore any attempt to write until the 8-bit data shifting is completed.

  The SSE bit will be kept at **1** if the communication is still in process. This flag must be cleared when shifting is completed. You can determine if the next write attempt is available.
- **SBRS2~SBRS0:** Program the clock frequency/rates and sources.
- **Edge Select:** Selects the appropriate clock edges by programming the CES bit.

### *7.8.2 SPI Mode Timing*

The SCK edge is selected by programming bit CES. The waveform shown in Figure 7-6 is applicable regardless whether the EM78P613 is in Master or Slave mode, with /SS disabled. However, the waveform in Figure 7-6b below can only be implemented in Slave mode with /SS enabled.

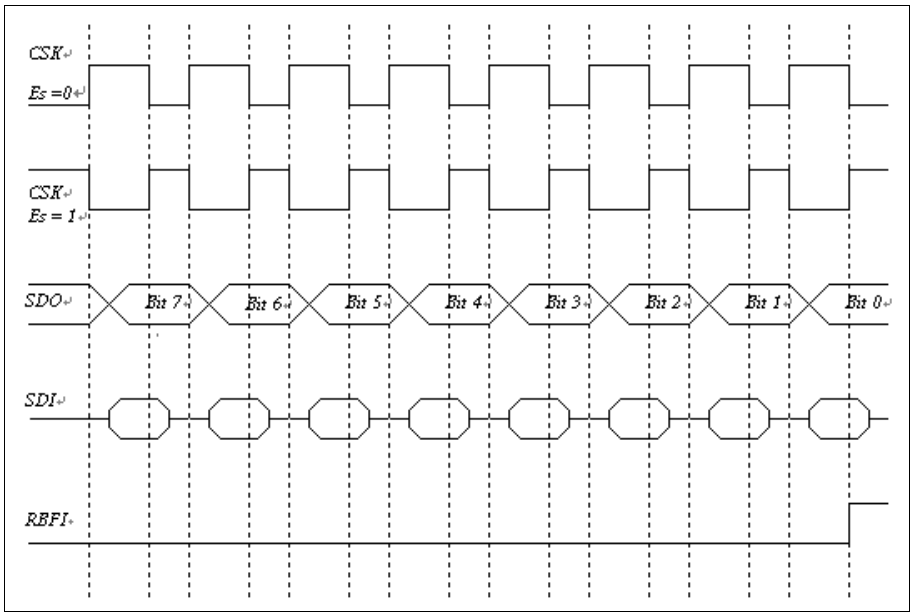



*Figure 7-6a SPI Mode with /SS Disabled*

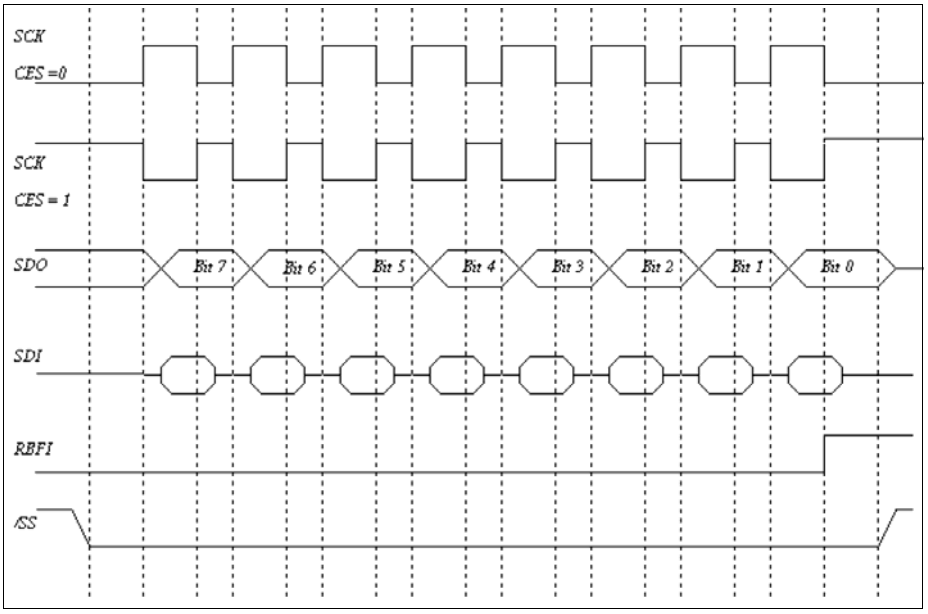



*Figure 7-6b SPI Mode with /SS Enabled*

# 8 Absolute Maximum Ratings

| Symbol | Min | Max | Unit |
|---|---|---|---|
| Temperature under bias | 0 | 70 | ºC |
| Storage temperature | -65 | 150 | ºC |
| Input voltage | -0.5 | 6.0 | V |
| Output voltage | -0.5 | 6.0 | V |

# 9 DC Electrical Characteristics

T = 25ºC, VDD=5V, VSS=0V

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| **3.3V Regulator** | | | | | | |
| $V_{Rag}$ | Output voltage of 3.3V Regulator | $V_{DD}$ = 5V | 3.0 | 3.3 | 3.6 | V |
| $V_{ResetL}$ | Low Power Reset detecting low Voltage | – | – | – | 2.2 | V |
| $V_{ResetH}$ | Low Power Reset detecting high Voltage | – | 2.3 | – | – | V |
| Ireg | 3.3V Regulator driving capacity | V3.3 = 3.3V | – | – | 100 | mA |
| **MCU Operation** | | | | | | |
| $I_{IL}$ | Input Leakage Current for input pins | VIN=VDD, VSS | – | – | ±1 | µA |
| $V_{IHX}$ | Clock Input High Voltage | OSCI | 2.5 | – | – | V |
| $V_{ILX}$ | Clock Input Low Voltage | OSCI | – | – | 1.0 | V |
| $I_{CC1}$ | VDD operating supply current – Normal frequency operation mode | Freq. = 6 MHz | – | – | 10 | mA |
| $I_{CC2}$ | VDD operating supply current – Normal frequency operation mode | Freq. = 12 MHz | – | – | 20 | mA |
| $I_{CC3}$ | VDD operating supply current – Dual clock mode | Freq. = 256kHz | – | – | 250 | µA |
| $I_{SB1}$ | Operating Supply Current 1 – Power down mode | WDT disabled | – | – | 120 | µA |
| **GPIO Pins** | | | | | | |
| VOH | Output High Voltage | (Ports 5, 6, 8, P7 (P74 75 in USB Mode) and P90~P93 P95~97) | – | V33 | – | V |
| VOL | Output Low Voltage | (Port 5, 6, 8, P7 and P90~P93 P95~97) | – | Vss | – | V |
| $V_{IH}$ | Input High Voltage | Port 5 ~ Port 9 | 2.0 | – | – | V |
| $V_{IL}$ | Input Low Voltage | Port 5 ~ Port 9 | – | – | 0.8 | V |
| IOH1 | Output High Voltage (Port 5, Port 6, P76~77 and P92~93) | $I_{Sink}$ = 10.0 mA Vreg = 3.3V | – | 10.0 | – | mA |

*(Continuation)*

| Symbol | Parameter | Condition | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| IOH2 | Output High Voltage (P74, P75 in PS2 mode) | $I_{Sink}$ = 5.0 mA $V_{DD}$ = 5V | – | 5.0 | – | mA |
| IOL1 | Output Low Voltage (Ports 5, 6; P92, P93, P76, P77 in normal mode) | $I_{Sink}$ = 10.0 mA Vreg = 3.3V | – | 10.0 | – | mA |
| IOL2 | Output Low Voltage (P74, P75 in PS2 Mode) | $I_{Sink}$ = 10.0 mA VDD = 5V | – | 10.0 | – | mA |
| IOL3 | Output Low Voltage (P92, P93, P76, P77 sink LED) | $I_{Sink}$ = 10.0 mA Vreg = 3.3V | – | 10.0 | – | mA |
| IOL4 | Output Low Voltage (P90 ~ P97 in normal mode) | $I_{Sink}$ = 10.0 mA Vreg = 3.3V | – | 10.0 | – | mA |
| IOL5 | Output Low Voltage (P90 ~P93, P95~ P97 sink LED) | $I_{Sink}$ = 10.0 mA VREG = 5V | – | 10.0 | – | mA |
| RPH1 | Pull-High Resistor (Ports 5, 6, 8, 9) | Input pin with pull-high resistor, VREG = 5V | – | 25.0 | – | KΩ |
| RPH2 | Pull-High Resistor ( P.74 ~ P.77), (P74/P75) in PS2 mode | Input pin with pull-high resistor, VDD = 5V | – | 2.20 | – | KΩ |
| **USB Interface** | | | | | | |
| $V_{OH}$ | Static Output High | USB Operation Mode | 2.8 | – | 3.6 | V |
| $V_{OL}$ | Static Output Low | | – | – | 0.3 | V |
| $V_{DI}$ | Differential Input Sensitivity | | 0.2 | – | – | V |
| $V_{CM}$ | Differential Input Command Mode Range | | 0.8 | – | 2.5 | V |
| $V_{SE}$ | Single Ended Receiver Threshold | | 0.8 | – | 2.0 | V |
| $C_{IN}$ | Transceiver Capacitance | | – | – | 20 | pF |
| $V_{RG}$ | Output Voltage of Internal Regulator | | 3.0 | – | 3.6 | V |
| $R_{PH3}$ | Pull-high Resistor (P.74 / D+) | | – | 1.5 | – | KΩ |

# 10 Application Circuits

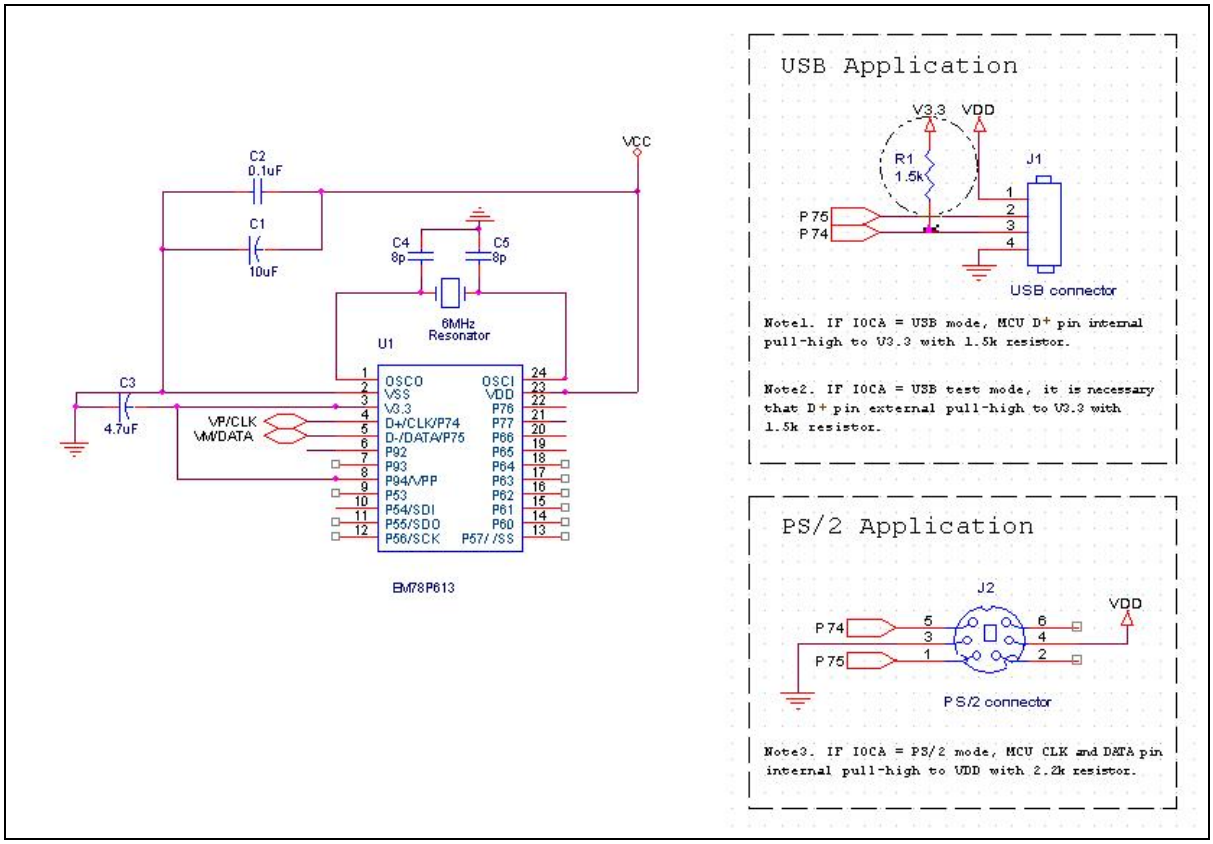



*Figure 10-1 EM78P613 Series USB & PS/2 Application Circuits*

Where**:**

**C1:** Bypass Capacitor. Placed adjacent to the $V_{DD}$ pin to minimize noise.

**C2, C3:** Power Capacitor. Placed adjacent to the power source to improve the transient response and ripple rejection.

**C4, C5:** Load Capacitor

# APPENDIX

# A Special Register Map

## A.1 Operation Registers

| Addr. | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit1 | Bit 0 | Default Value |
|---|---|---|---|---|---|---|---|---|---|---|
| 0×00 | R0 | Indirect Addressing Register | | | | | | | | 0B_0000_0000 |
| 0×01 | R1 (TCC) | Timer/Clock Counter | | | | | | | | 0B_0000_0000 |
| 0×02 | R2 (PC) | Program Counter | | | | | | | | 0B_0000_0000 |
| 0×03 | R3 (Status) | PS2 | PS1 | PS0 | T | P | Z | DC | C | 0B_0001_1xxx |
| 0×04 | R4 (RSR) | BK1 | BK0 | Select the register( Address: 00~3F) in indirect addressing mode | | | | | | |
| 0×05 | R5 (Port 5) | P57 | P56 | P55 | P54 | P53 | | | | 0B_uuuu_0000 |
| 0×06 | R6 (Port 6) | | P66 | P65 | P64 | P63 | P62 | P61 | P60 | 0B_0uuu_uuuu |
| 0×07 | R7 (Port 7) | P77 | P76 | P75/D-/DATA | P74/D+/CLK | – | | | | 0B_uuuu_0000 |
| 0×08 | R8 (SPIS) | Extr_R | SRO | SSE | SBF | /P5 Weakup | | | Set_Config | 0B_0000_1000 |
| 0×09 | R9 (Port 9) | – | – | – | – | P93 | P92 | – | – | 0B_000u_uu00 |
| 0×0A | RA | – | – | - | Delay Noise | – | – | – | – | 0B_00000000 |
| 0×0B | RB | | | | | | | | | 0B_0000_0000 |
| 0×0C | RC | EP0_W | EP0_R | EP1_R | EP2_R | EP2_W | UDC_ SUSPEND | UDC_ Writing | STALL | 0B_0000_0000 |
| 0×0D | RD | USB Application FIFO Address Register | | | | | | | | 0B_0000_0000 |
| 0×0E | RE | USB Application FIFO Data Register | | | | | | | | 0B_uuuu_uuuu |
| 0×0F | RF | USB Host Resume_IF | | SPIIF | Port 7 state change_1F | USB Reset_IF | USB Suspend_IF | EP0_IF | TCC_IF | 0B_0000_0000 |
| 0×10 | SPIRB | SRB7 | SRB6 | SRB5 | SRB4 | SRB3 | SRB2 | SRB1 | SRB0 | 0B_0000_0000 |
| 0×11 | SPIWB | SWB7 | SWB6 | SWB5 | SWB4 | SWB3 | SWB2 | SWB1 | SWB0 | 0B_0000_0000 |

## A.2 Control Registers

| Addr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Default Value |
|---|---|---|---|---|---|---|---|---|---|---|
| | CONT | LED | /INT | TSR2 | TSR1 | TSR0 | PSR2 | PSR1 | PSR0 | 0B_0011_1111 |
| 0×05 | IOC5 | Port 5 Direction Control Register | | | | | | | | 0B_1111_1111 |
| 0×06 | IOC6 | Port 6 Direction Control Register | | | | | | | | 0B_1111_1111 |
| 0×07 | IOC7 | Port 7 Direction Control Register | | | | | | | | 0B_1111_1111 |
| 0×08 | SPIC | SPI Control register | | | | | | | | 0B_0000_0000 |
| 0×09 | IOC9 | Port 9 Direction Control Register | | | | | | | | 0B_1111_1111 |
| 0×0A | IOCA | Dual_ Frq.1 | Dual_ Frq.0 | /P76_ 77 Pull -up | Remote _Wake Up | ExReg_ Sel | – | PS/2 | USB | 0B_1110_0011 |
| 0×0B | IOCB | | | | /P94 | /P93 | /P92 | – | – | 0B_xxx1_11xx |
| 0×0C | IOCC | | | | – | P93 | P92 | | | 0B_xxxx_00xx |
| 0×0D | IOCD | | | | – | /P93 | /P92 | | | 0B_xxxx_00xx |
| 0×0E | IOCE | /Dual clock | /WUE | WTE | RUN | Device_ Resume | /PU8 | /PU6 | /PU5 | 0B_1101_0111 |
| 0×0F | IOCF | USB Host Resume _IE | – | SPI_IE | Port 7 State Change _1F | USB Reset_IE | USB Suspend _IE | EP0_IE | TCC_IE | 0B_0000_0000 |

# 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 two 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) Executing 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.

Furthermore, 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 any 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[1] |
| 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 | A, R3, R4,R8.7 |
| 0 0000 0001 0100 | 0014 | CONTR | CONT → A | None |
| 0 0000 0001 rrrr | 001r | IOR R | IOCR → A | None[1] |
| 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 |
| 0 0001 00rr rrrr | 01rr | SUB A,R | R-A → A | Z, C, DC |
| 0 0001 01rr rrrr | 01rr | SUB R,A | R-A → R | Z, C, DC |

*(Continuation)*

| Binary Instruction | Hex | Mnemonic | Operation | Status Affected |
|---|---|---|---|---|
| 0 0001 10rr rrrr | 01rr | DECA R | R-1 → A | Z |
| 0 0001 11rr rrrr | 01rr | DEC R | R-1 → R | Z |
| 0 0010 00rr rrrr | 02rr | OR A,R | A ∨ VR → A | Z |
| 0 0010 01rr rrrr | 02rr | OR R,A | A ∨ VR → R | Z |
| 0 0010 10rr rrrr | 02rr | AND A,R | A & R → A | Z |
| 0 0010 11rr rrrr | 02rr | AND R,A | A & R → R | Z |
| 0 0011 00rr rrrr | 03rr | XOR A,R | A ⊕ R → A | Z |
| 0 0011 01rr rrrr | 03rr | XOR R,A | A ⊕ R → R | Z |
| 0 0011 10rr rrrr | 03rr | ADD A,R | A + R → A | Z, C, DC |
| 0 0011 11rr rrrr | 03rr | ADD R,A | A + R → R | Z, C, DC |
| 0 0100 00rr rrrr | 04rr | MOV A,R | R → A | Z |
| 0 0100 01rr rrrr | 04rr | MOV R,R | R → R | Z |
| 0 0100 10rr rrrr | 04rr | COMA R | /R → A | Z |
| 0 0100 11rr rrrr | 04rr | COM R | /R → R | Z |
| 0 0101 00rr rrrr | 05rr | INCA R | R+1 → A | Z |
| 0 0101 01rr rrrr | 05rr | INC R | R+1 → R | Z |
| 0 0101 10rr rrrr | 05rr | DJZA R | R-1 → A, skip if zero | None |
| 0 0101 11rr rrrr | 05rr | DJZ R | R-1 → R, skip if zero | None |
| 0 0110 00rr rrrr | 06rr | RRCA R | R(n) → A(n-1),<br>R(0) → C, C → A(7) | C |
| 0 0110 01rr rrrr | 06rr | RRC R | R(n) → R(n-1),<br>R(0) → C, C → R(7) | C |
| 0 0110 10rr rrrr | 06rr | RLCA R | R(n) → A(n+1),<br>R(7) → C, C → A(0) | C |
| 0 0110 11rr rrrr | 06rr | RLC R | R(n) → R(n+1),<br>R(7) → C, C → R(0) | C |
| 0 0111 00rr rrrr | 07rr | SWAPA R | R(0-3) → A(4-7),<br>R(4-7) → A(0-3) | None |
| 0 0111 01rr rrrr | 07rr | SWAP R | R(0-3) ↔ R(4-7) | None |
| 0 0111 10rr rrrr | 07rr | JZA R | R+1 → A, skip if zero | None |
| 0 0111 11rr rrrr | 07rr | JZ R | R+1 → R, skip if zero | None |
| 0 100b bbrr rrrr | 0xxx | BC R,b | 0 → R(b) | None[2] |
| 0 101b bbrr rrrr | 0xxx | BS R,b | 1 → R(b) | None |
| 0 110b bbrr rrrr | 0xxx | JBC R,b | if R(b)=0, skip | None |
| 0 111b bbrr rrrr | 0xxx | JBS R,b | if R(b)=1, skip | None |
| 1 00kk kkkk kkkk | 1kkk | CALL k | PC+1 → [SP],<br>(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 1111 kkkk kkkk | 1Fkk | ADD A,k | k+A → A | Z, C, DC |

**Note:** [1] This instruction is applicable to IOCx only.

[2] This instruction is not recommended for RE, RF operation.

# C Code Option Register

EM78P613 has two Code option registers, which are not part of the normal program memory. The option bits cannot be accessed during normal program execution.

## C.1 Address 000 Code Option Register

| Bit | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mnemonic | EP2 Ena | /R.S | Package_1 | Package_0 | ID_3 | ID_2 | ID_1 | ID_0 | OST_1 | OST_0 | Freq_1 | Freq_0 | /Protect |

## C.2 Address 001 Code Option Register

| Bit | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mnemonic | -- | -- | -- | -- | -- | -- | -- | EP2_ Maxsize_4 | EP2_ Maxsize_3 | EP2_ Maxsize_2 | EP2_ Maxsize_1 | EP2_ Maxsize_0 | EP2_ DIR |

## C.3 Address 000 Bit Description

**Bit 0:** (/Protect)

**0:** Enable Protect

**1:** Disable Protect

**Bit 2~1:** (Frequency): Frequency Selection

**00:** 6 MHz

**01:**12 MHz

**10:**16 MHz

**11:** Reserved

**Bits 4~3:** (OST_1 ~ OST_0): Oscillator start-up time

**00:** 500 μs

**01:** 2 ms

**10:** 8 ms

**11:** 16 ms

**Bits 8~5:** User ID

**Bits 10~9:** (Package_1 ~ Package_0): Package type selection

**00:** Not defined

**01:** 18 pins

**10:** 20 pins

**11:** 24 pins

**Bit 11:** (/R.S.): D+ Pull-up Resistance

**0:** Connect Resistor Switch

**1:** Disconnect Resistor Switch

**Bit 12:** (EP2_Enable): Endpoint 2 Enable

**0:** Disable

**1:** Enable

## C.4 Address 001 Bit Description

**Bit 0:** (EP2_Dir): Endpoint 2 Direction

**0:** OUT

**1:** IN

**Bits 5~1:** (EP2_Maxsize_4~0): Endpoint 2 maximum size

**00000:** reserved

**00001:** 1 Bytes

**00010:** 2 Bytes

**00011:** 3 Bytes

**:**

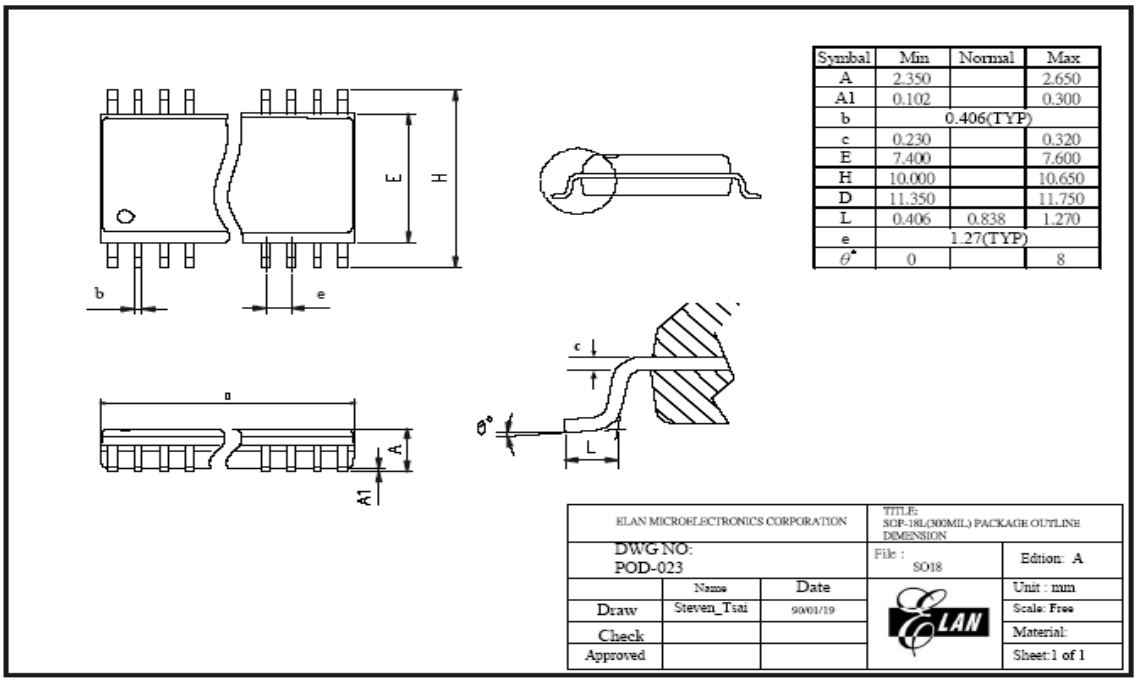
**:**

**10000:** 16 Bytes

**Bits 12~6:** Values are fixed

# D Package Out-line Dimension

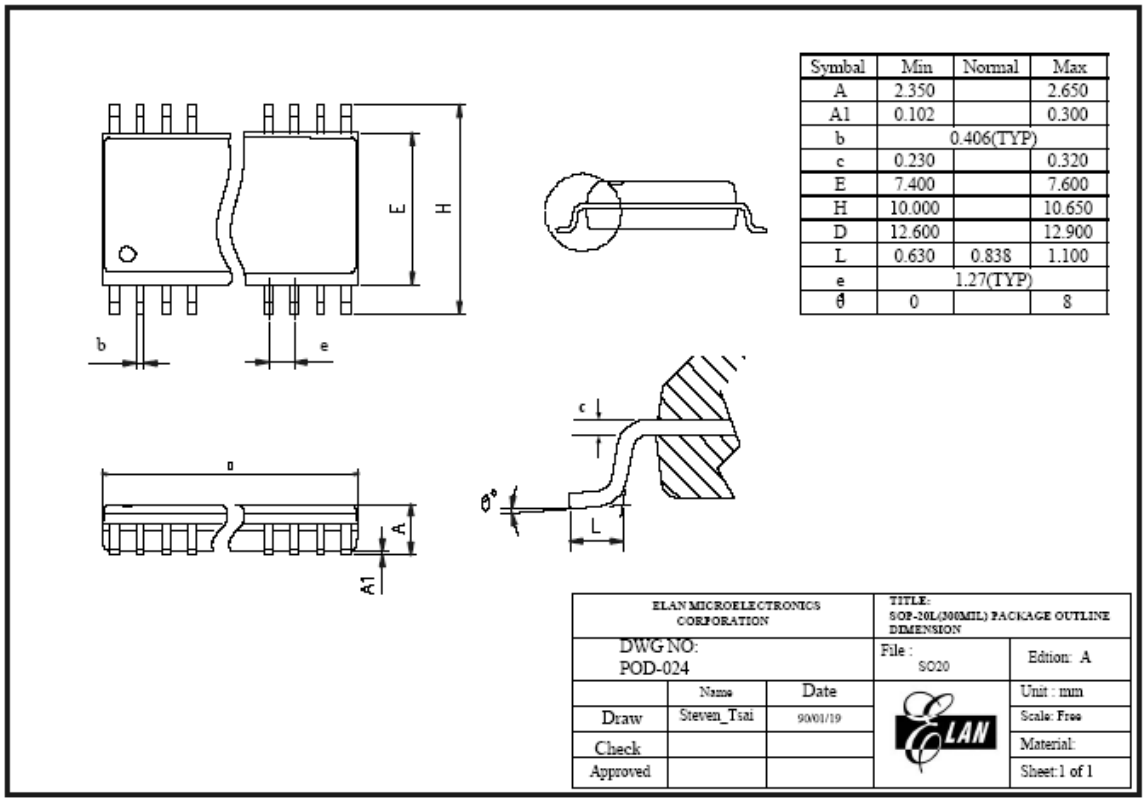
■ **EM78P613AM**




| Symbol | Min | Normal | Max |
|---|---|---|---|
| A | 2.350 | | 2.650 |
| A1 | 0.102 | | 0.300 |
| b | | 0.406(TYP) | |
| c | 0.230 | | 0.320 |
| E | 7.400 | | 7.600 |
| H | 10.000 | | 10.650 |
| D | 11.350 | | 11.750 |
| L | 0.406 | 0.838 | 1.270 |
| e | | 1.27(TYP) | |
| θ° | 0 | | 8 |

| ELAN MICROELECTRONICS CORPORATION | | | TITLE: SOP-18L(300MIL) PACKAGE OUTLINE DIMENSION |
|---|---|---|---|
| DWG NO: POD-023 | | | File : SO18 / Edition: A |
| | Name | Date | Unit : mm |
| Draw | Steven_Tsai | 90/01/19 | Scale: Free |
| Check | | | Material: |
| Approved | | | Sheet:1 of 1 |

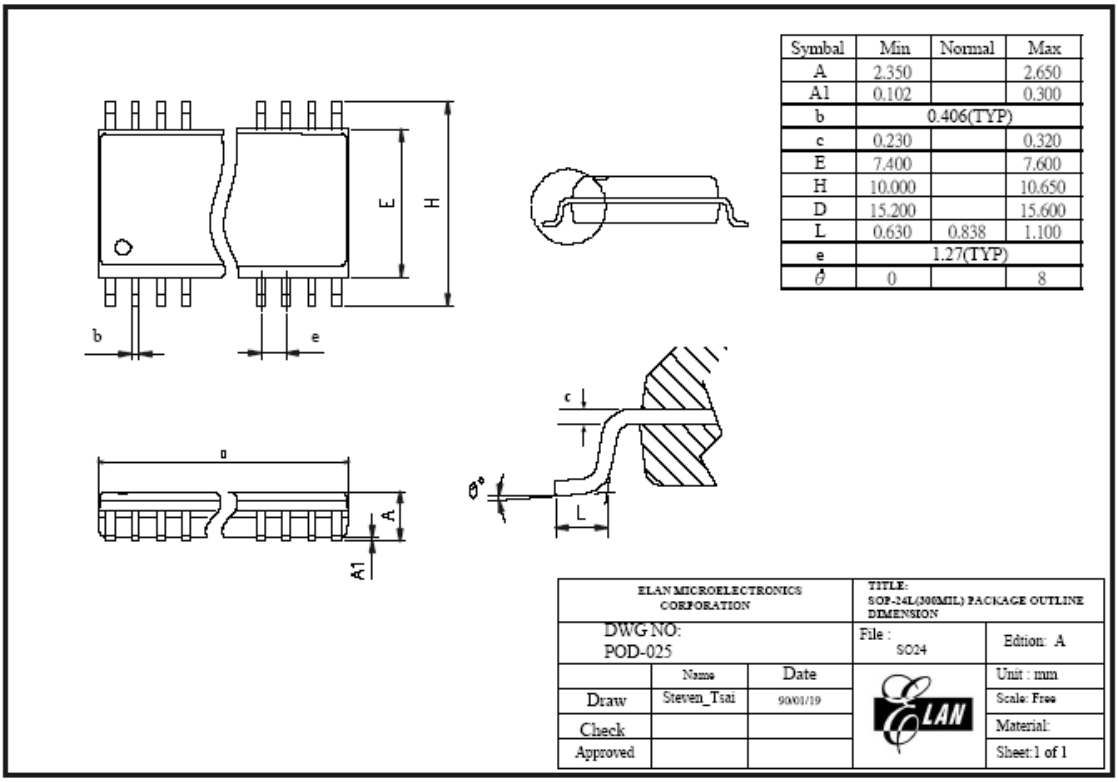
■ **EM78P613BM**




| Symbal | Min | Normal | Max |
|---|---|---|---|
| A | 2.350 | | 2.650 |
| A1 | 0.102 | | 0.300 |
| b | | 0.406(TYP) | |
| c | 0.230 | | 0.320 |
| E | 7.400 | | 7.600 |
| H | 10.000 | | 10.650 |
| D | 12.600 | | 12.900 |
| L | 0.630 | 0.838 | 1.100 |
| e | | 1.27(TYP) | |
| θ | 0 | | 8 |

| ELAN MICROELECTRONICS CORPORATION | | | TITLE: SOP-20L(300MIL) PACKAGE OUTLINE DIMENSION |
|---|---|---|---|
| DWG NO: POD-024 | | | File : SO20 / Edition: A |
| | Name | Date | Unit : mm |
| Draw | Steven_Tsai | 90/01/19 | Scale: Free |
| Check | | | Material: |
| Approved | | | Sheet:1 of 1 |

■ **EM78P613CM**




| Symbal | Min | Normal | Max |
|---|---|---|---|
| A | 2.350 | | 2.650 |
| A1 | 0.102 | | 0.300 |
| b | | 0.406(TYP) | |
| c | 0.230 | | 0.320 |
| E | 7.400 | | 7.600 |
| H | 10.000 | | 10.650 |
| D | 15.200 | | 15.600 |
| L | 0.630 | 0.838 | 1.100 |
| e | | 1.27(TYP) | |
| θ | 0 | | 8 |

| ELAN MICROELECTRONICS CORPORATION | | | TITLE: SOP-24L(300MIL) PACKAGE OUTLINE DIMENSION |
|---|---|---|---|
| DWG NO: POD-025 | | | File : SO24 / Edition: A |
| | Name | Date | Unit : mm |
| Draw | Steven_Tsai | 99/01/19 | Scale: Free |
| Check | | | Material: |
| Approved | | | Sheet:1 of 1 |

# E OTP Program Pin

| IC Pin Name | 18-Pin Package | 20-Pin Package | 24-Pin DIP/SOP |
|---|---|---|---|
| P94 | 15 | 17 | 8 |
| P57 | 2 | 2 | 13 |
| P56 | 1 | 1 | 12 |
| P55 | 18 | 20 | 11 |
| VSS | 10 | 11 | 2 |
| OSCI | 8 | 9 | 24 |
| VDD | 7 | 8 | 23 |
| P54 | 17 | 19 | 10 |