

ABOV SEMICONDUCTOR Co., Ltd.  
8-BIT MICROCONTROLLERS

# MC96F8316

*User's Manual (Ver. 1.3)*



## REVISION HISTORY

### VERSION 0.0 (November 10, 2011)

### VERSION 1.0 (April 12, 2012)

Change '1.3/2.6mA (Typ/Max)' to "IDD2 @X-tal 12MHz" in DC Electrical characteristics.

Change '0.7/1.4mA (Typ/Max)' to "IDD2 @X-tal 10MHz" in DC Electrical characteristics.

Change '0.8/1.6mA (Typ/Max)' to "IDD2 @IRC 16MHz" in DC Electrical characteristics.

Change '60.0/90.0uA (Typ/Max)' to "IDD3" in DC Electrical characteristics.

Change '6.0/12.0uA (Typ/Max)' to "IDD4" in DC Electrical characteristics.

Change ' $\pm 1.0\%(T_A = 0^\circ\text{C} \sim +50^\circ\text{C})/\pm 2.0\%(T_A = -20^\circ\text{C} \sim +85^\circ\text{C})/\pm 3.0\%(T_A = -40^\circ\text{C} \sim +85^\circ\text{C})$ ' @2.0V – 5.5V to "IRC's Tolerance" in Internal RC Oscillator characteristics.

Change ' $\pm 4$  (Max)' to "ILE" in ADC electrical characteristics.

Change '50/150mV (Typ/Max)' to "LVR/LVI Hysteresis" in LVR/LVI electrical characteristics.

Change '8.0/12.0 uA (Typ/Max)' to "Current consumption(one of two)" in LVR/LVI characteristics.

Change '10.0/15.0 uA (Typ/Max)' to "Current consumption(both)" in LVR/LVI characteristics.

Add 'Instructions on how to use the input port' descriptions.

Add description about clock source(T1 A Match) of Timer 2.

Add 'Application circuit for ADC Key Input by resistor string'.

Retype a typo at 'EO Register description'.

Retype a typo on I2C Frequency formula.

Retype some typo errors.

### VERSION 1.1 (April 20, 2012)

Change '30V/mS (Max)' to "VDD Voltage Rising Time" in Power-on Reset Electrical characteristics.

### VERSION 1.2 (July 9, 2012)

Change '-10mA' to "IOH of VOH1" in DC Electrical characteristics.

Change 'VDD-1.5V (Min)' to "VOH1" in DC Electrical characteristics.

### VERSION 1.3 (January 8, 2013) This book

Add a package type, "MC96F8316U (32QFN)".

---

Version 1.3

Published by FAE Team

©2013 ABOV Semiconductor Co.,Ltd. All rights reserved.

---

Additional information of this manual may be served by ABOV Semiconductor offices in Korea or Distributors.

ABOV Semiconductor reserves the right to make changes to any information here in at any time without notice.

The information, diagrams and other data in this manual are correct and reliable; however, ABOV Semiconductor is in no way responsible for any violations of patents or other rights of the third party generated by the use of this manual.

# Table of Contents

|                                                                       |    |
|-----------------------------------------------------------------------|----|
| 1. Overview .....                                                     | 9  |
| 1.1 Description .....                                                 | 9  |
| 1.2 Features .....                                                    | 10 |
| 1.3 Ordering Information .....                                        | 11 |
| 1.4 Development Tools .....                                           | 12 |
| 2. Block Diagram.....                                                 | 14 |
| 3. Pin Assignment .....                                               | 15 |
| 4. Package Diagram.....                                               | 18 |
| 5. Pin Description .....                                              | 25 |
| 6. Port Structures .....                                              | 28 |
| 6.1 General Purpose I/O Port.....                                     | 28 |
| 6.2 External Interrupt I/O Port.....                                  | 29 |
| 7. Electrical Characteristics.....                                    | 30 |
| 7.1 Absolute Maximum Ratings.....                                     | 30 |
| 7.2 Recommended Operating Conditions .....                            | 30 |
| 7.3 A/D Converter Characteristics .....                               | 31 |
| 7.4 Power-On Reset Characteristics .....                              | 31 |
| 7.5 Low Voltage Reset and Low Voltage Indicator Characteristics ..... | 32 |
| 7.6 Internal RC Oscillator Characteristics.....                       | 33 |
| 7.7 Internal Watch-Dog Timer RC Oscillator Characteristics.....       | 33 |
| 7.8 DC Characteristics .....                                          | 34 |
| 7.9 AC Characteristics.....                                           | 36 |
| 7.10 SPI Characteristics.....                                         | 37 |
| 7.11 UART Characteristics .....                                       | 38 |
| 7.12 I2C Characteristics .....                                        | 39 |
| 7.13 Data Retention Voltage in Stop Mode .....                        | 40 |
| 7.14 Internal Flash Rom Characteristics .....                         | 41 |
| 7.15 Input/Output Capacitance.....                                    | 41 |
| 7.16 Main Clock Oscillator Characteristics .....                      | 42 |
| 7.17 Sub Clock Oscillator Characteristics .....                       | 43 |
| 7.18 Main Oscillation Stabilization Characteristics .....             | 44 |
| 7.19 Sub Oscillation Characteristics.....                             | 44 |
| 7.20 Operating Voltage Range.....                                     | 45 |
| 7.21 Recommended Circuit and Layout .....                             | 46 |
| 7.22 Typical Characteristics .....                                    | 47 |
| 8. Memory .....                                                       | 50 |
| 8.1 Program Memory.....                                               | 50 |
| 8.2 Data Memory .....                                                 | 52 |
| 8.3 XRAM Memory .....                                                 | 54 |
| 8.4 SFR Map .....                                                     | 55 |
| 9. I/O Ports.....                                                     | 62 |
| 9.1 I/O Ports .....                                                   | 62 |
| 9.2 Port Register .....                                               | 62 |
| 9.3 P0 Port .....                                                     | 64 |
| 9.4 P1 Port .....                                                     | 67 |
| 9.5 P2 Port .....                                                     | 71 |
| 9.6 P3 Port .....                                                     | 73 |
| 10. Interrupt Controller .....                                        | 75 |
| 10.1 Overview .....                                                   | 75 |
| 10.2 External Interrupt.....                                          | 76 |

|                                                             |     |
|-------------------------------------------------------------|-----|
| 10.3 Block Diagram .....                                    | 77  |
| 10.4 Interrupt Vector Table .....                           | 78  |
| 10.5 Interrupt Sequence .....                               | 78  |
| 10.6 Effective Timing after Controlling Interrupt Bit ..... | 80  |
| 10.7 Multi Interrupt .....                                  | 81  |
| 10.8 Interrupt Enable Accept Timing .....                   | 82  |
| 10.9 Interrupt Service Routine Address .....                | 82  |
| 10.10 Saving/Restore General-Purpose Registers .....        | 82  |
| 10.11 Interrupt Timing .....                                | 83  |
| 10.12 Interrupt Register Overview .....                     | 83  |
| 10.13 Interrupt Register Description .....                  | 85  |
| 11. Peripheral Hardware .....                               | 93  |
| 11.1 Clock Generator .....                                  | 93  |
| 11.2 Basic Interval Timer .....                             | 96  |
| 11.3 Watch Dog Timer .....                                  | 99  |
| 11.4 Watch Timer .....                                      | 102 |
| 11.5 Timer 0 .....                                          | 105 |
| 11.6 Timer 1 .....                                          | 114 |
| 11.7 Timer 2 .....                                          | 124 |
| 11.8 Buzzer Driver .....                                    | 134 |
| 11.9 12-Bit A/D Converter .....                             | 136 |
| 11.10 SPI .....                                             | 149 |
| 11.11 UART .....                                            | 155 |
| 11.12 I2C .....                                             | 169 |
| 12. Power Down Operation .....                              | 187 |
| 12.1 Overview .....                                         | 187 |
| 12.2 Peripheral Operation in IDLE/STOP Mode .....           | 187 |
| 12.3 IDLE Mode .....                                        | 188 |
| 12.4 STOP Mode .....                                        | 189 |
| 12.5 Release Operation of STOP Mode .....                   | 190 |
| 13. RESET .....                                             | 192 |
| 13.1 Overview .....                                         | 192 |
| 13.2 Reset Source .....                                     | 192 |
| 13.3 RESET Block Diagram .....                              | 192 |
| 13.4 RESET Noise Canceller .....                            | 193 |
| 13.5 Power on RESET .....                                   | 193 |
| 13.6 External RESETB Input .....                            | 196 |
| 13.7 Brown Out Detector Processor .....                     | 197 |
| 13.8 LVI Block Diagram .....                                | 198 |
| 14. On-chip Debug System .....                              | 202 |
| 14.1 Overview .....                                         | 202 |
| 14.2 Two-Pin External Interface .....                       | 203 |
| 15. Flash Memory .....                                      | 208 |
| 15.1 Overview .....                                         | 208 |
| 16. Configure Option .....                                  | 219 |
| 16.1 Configure Option Control .....                         | 219 |
| 17. APPENDIX .....                                          | 220 |

# List of Figures

|                                                                          |    |
|--------------------------------------------------------------------------|----|
| Figure 1.1 OCD Debugger and Pin Description .....                        | 12 |
| Figure 1.2 PGMplusUSB (Single Writer) .....                              | 13 |
| Figure 1.3 StandAlone PGMplus (Single Writer) .....                      | 13 |
| Figure 1.4 StandAlone Gang8 (for Mass Production) .....                  | 13 |
| Figure 2.1 Block Diagram .....                                           | 14 |
| Figure 3.1 MC96F8316D 32SOP Pin Assignment .....                         | 15 |
| Figure 3.2 MC96F8316U 32QFN Pin Assignment .....                         | 15 |
| Figure 3.3 MC96F8316M/G 28SOP/28SKDIP Pin Assignment .....               | 16 |
| Figure 3.4 MC96F8216U 24QFN Pin Assignment .....                         | 16 |
| Figure 3.5 MC96F8216D/B 20SOP/20PDIP Pin Assignment .....                | 17 |
| Figure 4.1 32-Pin SOP Package .....                                      | 18 |
| Figure 4.2 32-Pin QFN Package .....                                      | 19 |
| Figure 4.3 28-Pin SOP Package .....                                      | 20 |
| Figure 4.4 28-Pin SKDIP Package .....                                    | 21 |
| Figure 4.5 24-Pin QFN Package .....                                      | 22 |
| Figure 4.6 20-Pin SOP Package .....                                      | 23 |
| Figure 4.7 20-Pin PDIP Package .....                                     | 24 |
| Figure 6.1 General Purpose I/O Port .....                                | 28 |
| Figure 6.2 External Interrupt I/O Port .....                             | 29 |
| Figure 7.1 Input Timing for RESETB .....                                 | 36 |
| Figure 7.2 Input Timing for External Interrupts .....                    | 36 |
| Figure 7.3 Input Timing for EC0, EC1, EC2 .....                          | 36 |
| Figure 7.4 SPI Timing .....                                              | 37 |
| Figure 7.5 Waveform for UART Timing Characteristics .....                | 38 |
| Figure 7.6 Timing Waveform for the UART Module .....                     | 38 |
| Figure 7.7 I2C Timing .....                                              | 39 |
| Figure 7.8 Stop Mode Release Timing when Initiated by an Interrupt ..... | 40 |
| Figure 7.9 Stop Mode Release Timing when Initiated by RESETB .....       | 40 |
| Figure 7.10 Crystal/Ceramic Oscillator .....                             | 42 |
| Figure 7.11 External Clock .....                                         | 42 |
| Figure 7.12 Crystal Oscillator .....                                     | 43 |
| Figure 7.13 External Clock .....                                         | 43 |
| Figure 7.14 Clock Timing Measurement at XIN .....                        | 44 |
| Figure 7.15 Clock Timing Measurement at SXIN .....                       | 44 |
| Figure 7.16 Operating Voltage Range .....                                | 45 |
| Figure 7.17 Recommended Circuit and Layout .....                         | 46 |
| Figure 7.18 RUN (IDD1) Current .....                                     | 47 |
| Figure 7.19 IDLE (IDD2) Current .....                                    | 47 |
| Figure 7.20 SUB RUN (IDD3) Current .....                                 | 48 |
| Figure 7.21 SUB IDLE (IDD4) Current .....                                | 48 |
| Figure 7.22 STOP (IDD5) Current .....                                    | 49 |
| Figure 8.1 Program Memory .....                                          | 51 |
| Figure 8.2 Data Memory Map .....                                         | 52 |
| Figure 8.3 Lower 128 Bytes RAM .....                                     | 53 |
| Figure 8.4 XDATA Memory Area .....                                       | 54 |
| Figure 10.1 External Interrupt Description .....                         | 76 |

|                                                                                            |     |
|--------------------------------------------------------------------------------------------|-----|
| Figure 10.2 Block Diagram of Interrupt .....                                               | 77  |
| Figure 10.3 Interrupt Vector Address Table .....                                           | 79  |
| Figure 10.4 Effective Timing of Interrupt Enable Register .....                            | 80  |
| Figure 10.5 Effective Timing of Interrupt Flag Register .....                              | 80  |
| Figure 10.6 Effective Timing of Interrupt .....                                            | 81  |
| Figure 10.7 Interrupt Response Timing Diagram .....                                        | 82  |
| Figure 10.8 Correspondence between Vector Table Address and the Entry Address of ISP ..... | 82  |
| Figure 10.9 Saving/Restore Process Diagram and Sample Source .....                         | 82  |
| Figure 10.10 Timing Chart of Interrupt Acceptance and Interrupt Return Instruction .....   | 83  |
| Figure 11.1 Clock Generator Block Diagram .....                                            | 93  |
| Figure 11.2 Basic Interval Timer Block Diagram .....                                       | 96  |
| Figure 11.3 Watch Dog Timer Interrupt Timing Waveform .....                                | 99  |
| Figure 11.4 Watch Dog Timer Block Diagram .....                                            | 100 |
| Figure 11.5 Watch Timer Block Diagram .....                                                | 102 |
| Figure 11.6 8-Bit Timer/Counter Mode for Timer 0 .....                                     | 106 |
| Figure 11.7 8-Bit Timer/Counter 0 Example .....                                            | 106 |
| Figure 11.8 8-Bit PWM Mode for Timer 0 .....                                               | 107 |
| Figure 11.9 PWM Output Waveforms in PWM Mode for Timer 0 .....                             | 108 |
| Figure 11.10 8-Bit Capture Mode for Timer 0 .....                                          | 109 |
| Figure 11.11 Input Capture Mode Operation for Timer 0 .....                                | 110 |
| Figure 11.12 Express Timer Overflow in Capture Mode .....                                  | 110 |
| Figure 11.13 8-Bit Timer 0 Block Diagram .....                                             | 111 |
| Figure 11.14 16-Bit Timer/Counter Mode for Timer 1 .....                                   | 115 |
| Figure 11.15 16-Bit Timer/Counter 1 Example .....                                          | 115 |
| Figure 11.16 16-Bit Capture Mode for Timer 1 .....                                         | 116 |
| Figure 11.17 Input Capture Mode Operation for Timer 1 .....                                | 117 |
| Figure 11.18 Express Timer Overflow in Capture Mode .....                                  | 117 |
| Figure 11.19 16-Bit PPG Mode for Timer 1 .....                                             | 118 |
| Figure 11.20 16-Bit PPG Mode Timing chart for Timer 1 .....                                | 119 |
| Figure 11.21 16-Bit Timer 1 Block Diagram .....                                            | 120 |
| Figure 11.22 16-Bit Timer/Counter Mode for Timer 2 .....                                   | 125 |
| Figure 11.23 16-Bit Timer/Counter 2 Example .....                                          | 125 |
| Figure 11.24 16-Bit Capture Mode for Timer 2 .....                                         | 126 |
| Figure 11.25 Input Capture Mode Operation for Timer 2 .....                                | 127 |
| Figure 11.26 Express Timer Overflow in Capture Mode .....                                  | 127 |
| Figure 11.27 16-Bit PPG Mode for Timer 2 .....                                             | 128 |
| Figure 11.28 16-Bit PPG Mode Timing chart for Timer 2 .....                                | 129 |
| Figure 11.29 16-Bit Timer 2 Block Diagram .....                                            | 130 |
| Figure 11.30 Buzzer Driver Block Diagram .....                                             | 134 |
| Figure 11.31 12-bit ADC Block Diagram .....                                                | 137 |
| Figure 11.32 A/D Analog Input Pin with Capacitor .....                                     | 137 |
| Figure 11.33 A/D Power (AVREF) Pin with Capacitor .....                                    | 137 |
| Figure 11.34 ADC Operation for Align Bit .....                                             | 138 |
| Figure 11.35 A/D Converter Operation Flow .....                                            | 139 |
| Figure 11.36 ADC Power-down Wake-up Function Block Diagram .....                           | 140 |
| Figure 11.37 Application circuit for ADC Key Input by resistor string .....                | 141 |
| Figure 11.38 SPI Block Diagram .....                                                       | 149 |
| Figure 11.39 SPI Transmit/Receive Timing Diagram at CPHA = 0 .....                         | 151 |
| Figure 11.40 SPI Transmit/Receive Timing Diagram at CPHA = 1 .....                         | 151 |

|                                                                      |     |
|----------------------------------------------------------------------|-----|
| Figure 11.41 UART Block Diagram.....                                 | 156 |
| Figure 11.42 Clock Generation Block Diagram.....                     | 157 |
| Figure 11.43 Frame Format.....                                       | 158 |
| Figure 11.44 Start Bit Sampling.....                                 | 162 |
| Figure 11.45 Sampling of Data and Parity Bit.....                    | 162 |
| Figure 11.46 Stop Bit Sampling and Next Start Bit Sampling.....      | 163 |
| Figure 11.47 I2C Block Diagram.....                                  | 169 |
| Figure 11.48 Bit Transfer on the I2C-Bus.....                        | 170 |
| Figure 11.49 START and STOP Condition.....                           | 171 |
| Figure 11.50 Data Transfer on the I2C-Bus.....                       | 171 |
| Figure 11.51 Acknowledge on the I2C-Bus.....                         | 172 |
| Figure 11.52 Clock Synchronization during Arbitration Procedure..... | 173 |
| Figure 11.53 Arbitration Procedure of Two Masters.....               | 173 |
| Figure 11.54 Formats and States in the Master Transmitter Mode.....  | 175 |
| Figure 11.55 Formats and States in the Master Receiver Mode.....     | 177 |
| Figure 11.56 Formats and States in the Slave Transmitter Mode.....   | 179 |
| Figure 11.57 Formats and States in the Slave Receiver Mode.....      | 181 |
| Figure 12.1 IDLE Mode Release Timing by External Interrupt.....      | 188 |
| Figure 12.2 STOP Mode Release Timing by External Interrupt.....      | 189 |
| Figure 12.3 STOP Mode Release Flow.....                              | 190 |
| Figure 13.1 RESET Block Diagram.....                                 | 192 |
| Figure 13.2 Reset noise canceller timer diagram.....                 | 193 |
| Figure 13.3 Fast VDD Rising Time.....                                | 193 |
| Figure 13.4 Internal RESET Release Timing On Power-Up.....           | 193 |
| Figure 13.5 Configuration Timing when Power-on.....                  | 194 |
| Figure 13.6 Boot Process WaveForm.....                               | 194 |
| Figure 13.7 Timing Diagram after RESET.....                          | 196 |
| Figure 13.8 Oscillator generating waveform example.....              | 196 |
| Figure 13.9 Block Diagram of BOD.....                                | 197 |
| Figure 13.10 Internal Reset at the power fail situation.....         | 197 |
| Figure 13.11 Configuration timing when BOD RESET.....                | 198 |
| Figure 13.12 LVI Diagram.....                                        | 198 |
| Figure 14.1 Block Diagram of On-Chip Debug System.....               | 202 |
| Figure 14.2 10-bit Transmission Packet.....                          | 203 |
| Figure 14.3 Data Transfer on the Twin Bus.....                       | 204 |
| Figure 14.4 Bit Transfer on the Serial Bus.....                      | 204 |
| Figure 14.5 Start and Stop Condition.....                            | 205 |
| Figure 14.6 Acknowledge on the Serial Bus.....                       | 205 |
| Figure 14.7 Clock Synchronization during Wait Procedure.....         | 206 |
| Figure 14.8 Connection of Transmission.....                          | 207 |
| Figure 15.1 Flash Program ROM Structure.....                         | 209 |

## List of Tables

|                                                  |    |
|--------------------------------------------------|----|
| Table 1-1 Ordering Information of MC96F8316..... | 11 |
| Table 5-1 Normal Pin Description.....            | 25 |
| Table 7-1 Absolute Maximum Ratings.....          | 30 |
| Table 7-2 Recommended Operating Conditions.....  | 30 |

|                                                                                        |     |
|----------------------------------------------------------------------------------------|-----|
| Table 7-3 A/D Converter Characteristics .....                                          | 31  |
| Table 7-4 Power-on Reset Characteristics .....                                         | 31  |
| Table 7-5 LVR and LVI Characteristics .....                                            | 32  |
| Table 7-6 Internal RC Oscillator Characteristics .....                                 | 33  |
| Table 7-7 Internal WDTRC Oscillator Characteristics .....                              | 33  |
| Table 7-8 DC Characteristics .....                                                     | 34  |
| Table 7-9 AC Characteristics .....                                                     | 36  |
| Table 7-10 SPI Characteristics .....                                                   | 37  |
| Table 7-11 UART Characteristics .....                                                  | 38  |
| Table 7-12 I2C Characteristics .....                                                   | 39  |
| Table 7-13 Data Retention Voltage in Stop Mode .....                                   | 40  |
| Table 7-14 Internal Flash Rom Characteristics .....                                    | 41  |
| Table 7-15 Input/Output Capacitance .....                                              | 41  |
| Table 7-16 Main Clock Oscillator Characteristics .....                                 | 42  |
| Table 7-17 Sub Clock Oscillator Characteristics .....                                  | 43  |
| Table 7-18 Main Oscillation Stabilization Characteristics .....                        | 44  |
| Table 7-19 Sub Oscillation Stabilization Characteristics .....                         | 44  |
| Table 8-1 SFR Map Summary .....                                                        | 55  |
| Table 8-2 SFR Map .....                                                                | 56  |
| Table 9-1 Port Register Map .....                                                      | 63  |
| Table 10-1 Interrupt Group Priority Level .....                                        | 75  |
| Table 10-2 Interrupt Vector Address Table .....                                        | 78  |
| Table 10-3 Interrupt Register Map .....                                                | 85  |
| Table 11-1 Clock Generator Register Map .....                                          | 94  |
| Table 11-2 Basic Interval Timer Register Map .....                                     | 97  |
| Table 11-3 Watch Dog Timer Register Map .....                                          | 100 |
| Table 11-4 Watch Timer Register Map .....                                              | 103 |
| Table 11-5 Timer 0 Operating Modes .....                                               | 105 |
| Table 11-6 Timer 0 Register Map .....                                                  | 112 |
| Table 11-7 Timer 1 Operating Modes .....                                               | 114 |
| Table 11-8 Timer 2 Register Map .....                                                  | 120 |
| Table 11-9 Timer 2 Operating Modes .....                                               | 124 |
| Table 11-10 Timer 3 Register Map .....                                                 | 130 |
| Table 11-11 Buzzer Frequency at 8 MHz .....                                            | 134 |
| Table 11-12 Buzzer Driver Register Map .....                                           | 135 |
| Table 11-13 ADC Register Map .....                                                     | 142 |
| Table 11-14 SPI Register Map .....                                                     | 152 |
| Table 11-15 Equations for Calculating Baud Rate Register Setting .....                 | 157 |
| Table 11-16 UART Register Map .....                                                    | 163 |
| Table 11-17 Examples of UARTBD Settings for Commonly Used Oscillator Frequencies ..... | 168 |
| Table 11-18 I2C Register Map .....                                                     | 182 |
| Table 12-1 Peripheral Operation during Power Down Mode .....                           | 187 |
| Table 12-2 Power Down Operation Register Map .....                                     | 191 |
| Table 13-1 Reset State .....                                                           | 192 |
| Table 13-2 Boot Process Description .....                                              | 195 |
| Table 13-3 Reset Operation Register Map .....                                          | 199 |
| Table 15-1 Flash Memory Register Map .....                                             | 210 |



# MC96F8316

## CMOS SINGLE-CHIP 8-BIT MICROCONTROLLER WITH 12-BIT A/D CONVERTER

### 1. Overview

#### 1.1 Description

The MC96F8316 is advanced CMOS 8-bit microcontroller with 16k bytes of FLASH. This is powerful microcontroller which provides a highly flexible and cost effective solution to many embedded control applications. This provides the following features : 16k bytes of FLASH, 256 bytes of IRAM, 512 bytes of XRAM , general purpose I/O, basic interval timer, watchdog timer, 8/16-bit timer/counter, 16-bit PPG output, 8-bit PWM output, watch timer, buzzer driving port, SPI, UART, I2C, 12-bit A/D converter, on-chip POR, LVR, LVI, on-chip oscillator and clock circuitry. The MC96F8316 also supports power saving modes to reduce power consumption.

| Device Name | FLASH     | XRAM      | IRAM      | ADC        | I/O PORT | Package  |
|-------------|-----------|-----------|-----------|------------|----------|----------|
| MC96F8316D  | 16k bytes | 512 bytes | 256 bytes | 15 channel | 30       | 32 SOP   |
| MC96F8316U  |           |           |           | 15 channel | 30       | 32 QFN   |
| MC96F8316M  |           |           |           | 12 channel | 26       | 28 SOP   |
| MC96F8316G  |           |           |           | 12 channel | 26       | 28 SKDIP |
| MC96F8216U  |           |           |           | 11 channel | 22       | 24 QFN   |
| MC96F8216D  |           |           |           | 8 channel  | 18       | 20 SOP   |
| MC96F8216B  |           |           |           | 8 channel  | 18       | 20 PDIP  |

## 1.2 Features

- **CPU**
  - 8 Bit CISC Core (8051 Compatible)
- **ROM (FLASH) Capacity**
  - 16k Bytes
  - Flash with self read/write capability
  - On chip debug and In-system programming (ISP)
  - Endurance : 100,000 times
- **256 Bytes IRAM**
- **512 Bytes XRAM**
- **General Purpose I/O (GPIO)**
  - Normal I/O : 30 Ports  
(P0[6:0], P1[7:0], P2[6:0], P3[7:0])
  - LED display drive capability pins : 14 Ports  
(P1[7:0], P2[5:0])
- **Basic Interval Timer (BIT)**
  - 8Bit × 1ch
- **Watch Dog Timer (WDT)**
  - 8Bit × 1ch
  - 5kHz internal RC oscillator
- **Timer/ Counter**
  - 8Bit × 1ch (T0), 16Bit × 2ch (T1/T2)
- **Programmable Pulse Generation**
  - Pulse generation (by T1/T2)
  - 8Bit PWM (by T0)
- **Watch Timer (WT)**
  - 3.91mS/0.25S/0.5S/1S/1M interval at 32.768kHz
- **Buzzer**
  - 8Bit × 1ch
- **SPI**
  - 8Bit × 1ch
- **UART**
  - 8Bit × 1ch
- **I2C**
  - 8Bit × 1ch
- **12 Bit A/D Converter**
  - 15 Input channels
  - Power down wake-up function
- **Power On Reset**
  - Reset release level (1.4V)
- **Low Voltage Reset**
  - 14 level detect (1.60V/ 2.00V/ 2.10V/ 2.20V/ 2.32V/ 2.44V/ 2.59V/ 2.75V/ 2.93V/ 3.14V/ 3.38V/ 3.67V/ 4.00V/ 4.40V)
- **Low Voltage Indicator**
  - 13 level detect (2.00V/ 2.10V/ 2.20V/ 2.32V/ 2.44V/ 2.59V/ 2.75V/ 2.93V/ 3.14V/ 3.38V/ 3.67V/ 4.00V/ 4.40V)
- **Interrupt Sources**
  - External Interrupts  
(EINT0~4, EINT5, EINT6, EINT7~A, EINT10, EINT11, EINT12) (7)
  - Timer(0/1/2) (4)
  - WDT (1)
  - BIT (1)
  - WT (1)
  - SPI (1)
  - UART (2)
  - I2C (1)
  - ADC (1)
  - ADC Wake-up (1)
- **Internal RC Oscillator**
  - Internal RC frequency:  
16MHz ±1.0% (T<sub>A</sub>= 0 ~ +50°C)
- **Power Down Mode**
  - STOP, IDLE mode
- **Operating Voltage and Frequency**
  - 1.8V ~ 5.5V (@ 32 ~ 38kHz with X-tal)
  - 1.8V ~ 5.5V (@ 0.4 ~ 4.2MHz with X-tal)
  - 2.7V ~ 5.5V (@ 0.4 ~ 10.0MHz with X-tal)
  - 3.0V ~ 5.5V (@ 0.4 ~ 12.0MHz with X-tal)
  - 1.8V ~ 5.5V (@ 0.5 ~ 8.0MHz with Internal RC)
  - 2.0V ~ 5.5V (@ 0.5 ~ 16.0MHz with Internal RC)
  - Voltage dropout converter included for core
- **Minimum Instruction Execution Time**
  - 125nS (@ 16MHz main clock)
  - 61μS (@ 32.768kHz sub clock)
- **Operating Temperature:** – 40 ~ + 85°C
- **Oscillator Type**
  - 0.4-12MHz Crystal or Ceramic for main clock
  - 32.768kHz Crystal for sub clock
- **Package Type**
  - 32 SOP/QFN
  - 28 SOP/SKDIP
  - 24 QFN
  - 20 SOP/PDIP
  - Pb-free package

### 1.3 Ordering Information

**Table 1-1 Ordering Information of MC96F8316**

| Device name | ROM size        | IRAM size | XRAM size | Package  |
|-------------|-----------------|-----------|-----------|----------|
| MC96F8316D  | 16k bytes FLASH | 256 bytes | 512 bytes | 32 SOP   |
| MC96F8316U  |                 |           |           | 32 QFN   |
| MC96F8316M  |                 |           |           | 28 SOP   |
| MC96F8316G  |                 |           |           | 28 SKDIP |
| MC96F8216U  |                 |           |           | 24 QFN   |
| MC96F8216D  |                 |           |           | 20 SOP   |
| MC96F8216B  |                 |           |           | 20 PDIP  |

## 1.4 Development Tools

### 1.4.1 Compiler

We do not provide the compiler. Please contact the third parties.

The core of MC96F8316 is Mentor 8051. And, device ROM size is smaller than 64k bytes. Developer can use all kinds of third party's standard 8051 compiler.

### 1.4.2 OCD emulator and debugger

The OCD (On Chip Debug) emulator supports ABOV Semiconductor's 8051 series MCU emulation.

The OCD interface uses two-wire interfacing between PC and MCU which is attached to user's system. The OCD can read or change the value of MCU internal memory and I/O peripherals. And the OCD also controls MCU internal debugging logic, it means OCD controls emulation, step run, monitoring, etc.

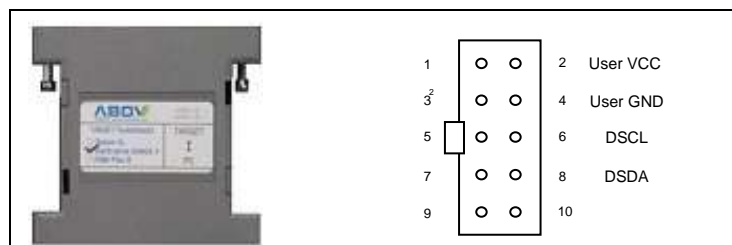
The OCD Debugger program works on Microsoft-Windows NT, 2000, XP, Vista (32bit) operating system.

If you want to see more details, please refer to OCD debugger manual. You can download debugger S/W and manual from our web-site.

Connection:

- DSCL (MC96F8316 P01 port)
- DSDA (MC96F8316 P00 port)

OCD connector diagram: Connect OCD with user system



**Figure 1.1 OCD Debugger and Pin Description**

### 1.4.3 Programmer

Single programmer:

PGMplus USB: It programs MCU device directly.



Figure 1.2 PGMplusUSB (Single Writer)

StandAlone PGMplus: It programs MCU device directly.



Figure 1.3 StandAlone PGMplus (Single Writer)

OCD emulator: It can write code in MCU device too, because OCD debugging supports ISP (In System Programming).

It does not require additional H/W, except developer's target system.

Gang programmer:

It programs 8 MCU devices at once.

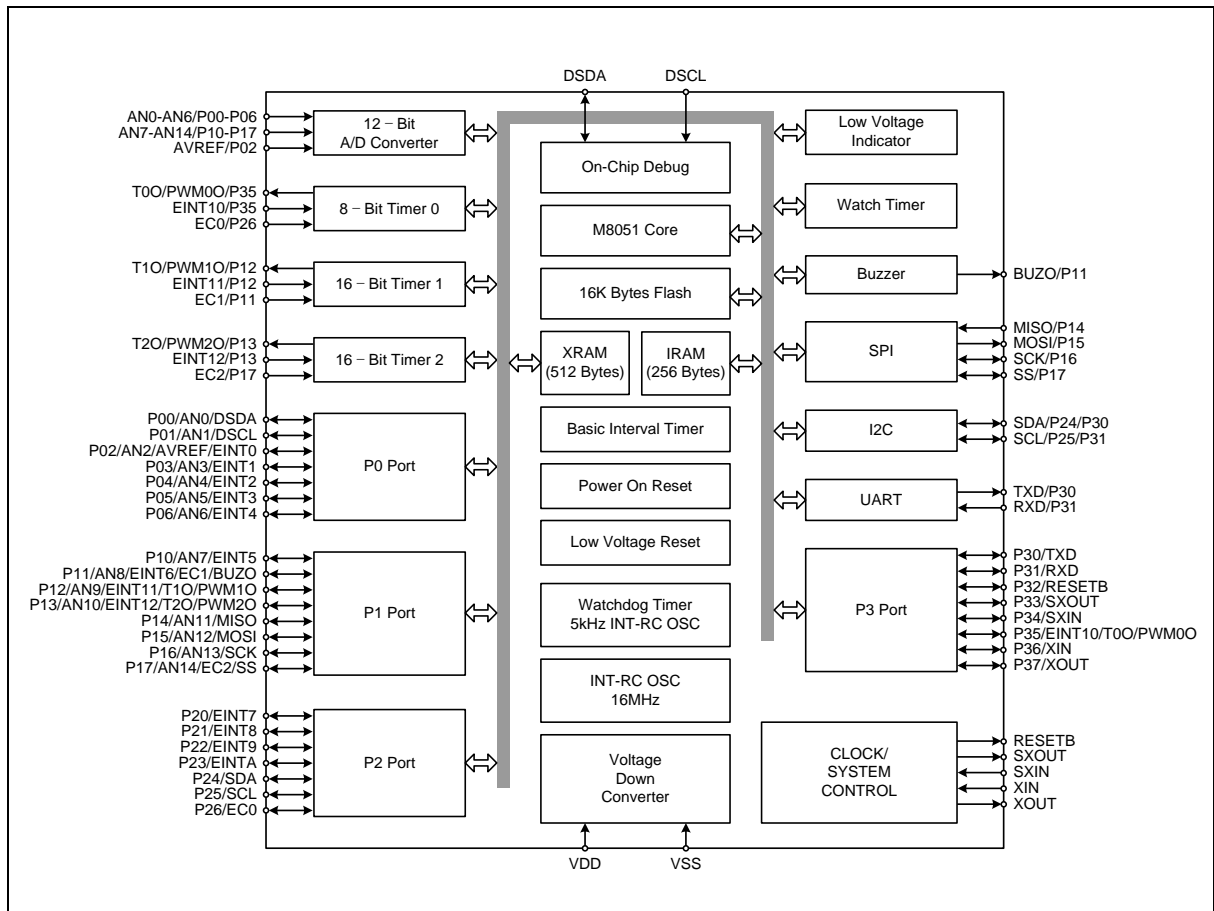
So, it is mainly used in mass production line.

Gang programmer is standalone type, it means it does not require host PC, after a program is downloaded from host PC to Gang programmer.



Figure 1.4 StandAlone Gang8 (for Mass Production)

## 2. Block Diagram



**Figure 2.1 Block Diagram**

- NOTES) 1. The P04–P06, and P26 are not in the 28-pin package.  
 2. The P04–P06, P17, P20–P22, and P26 are not in the 24-pin package.  
 3. The P04–P06, P14–P17, P20, P23, P26, P33–P34 and are not in the 20-pin package.

### 3. Pin Assignment

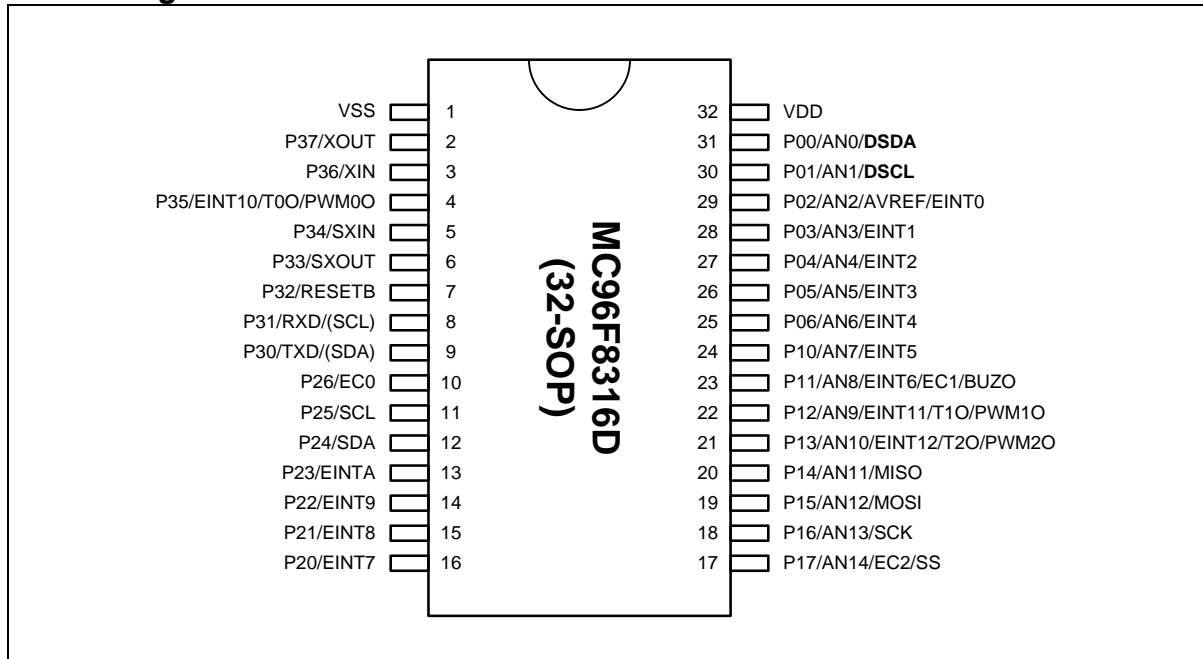


Figure 3.1 MC96F8316D 32SOP Pin Assignment

- NOTES) 1. On On-Chip Debugging, ISP uses P0[1:0] pin as DSCL, DSDA.  
2. The SDA/SCL lines for I2C can be configured on the P30/P31 pins by software control.

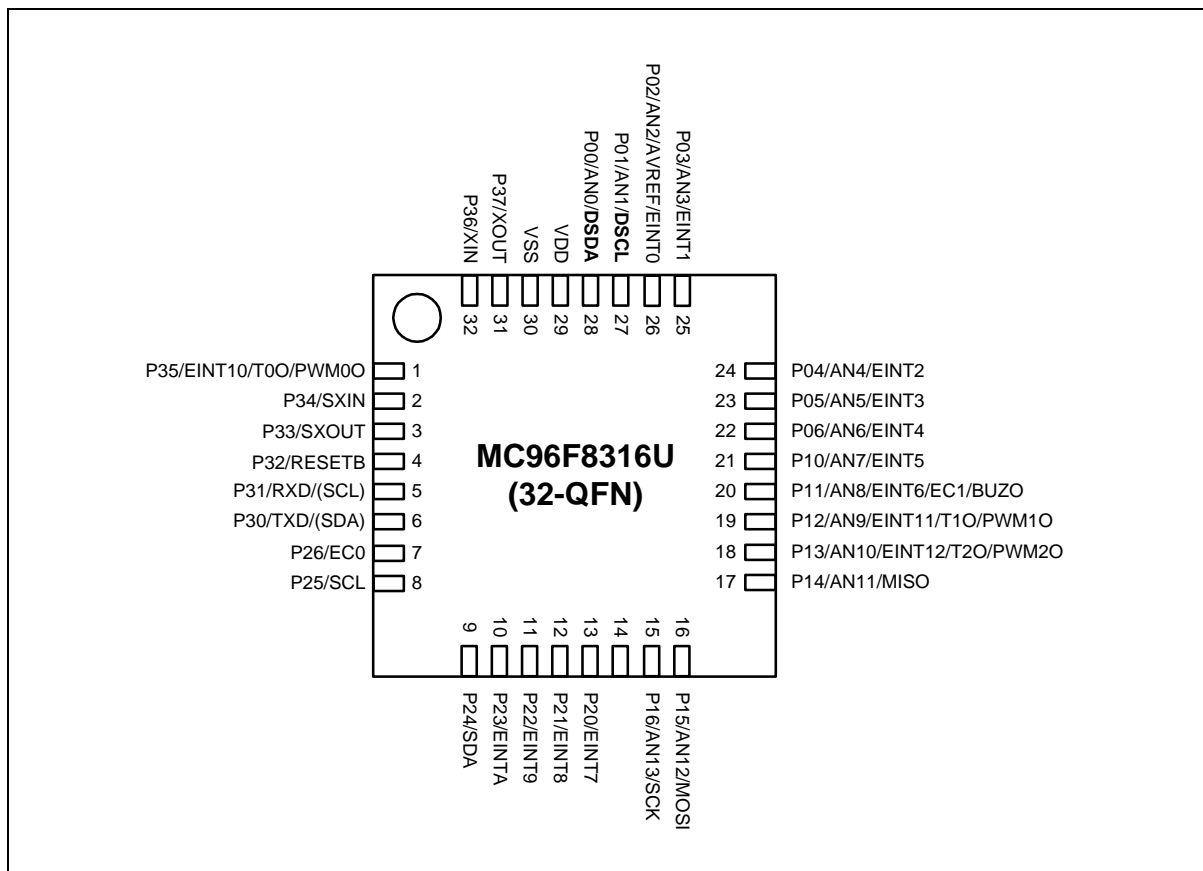
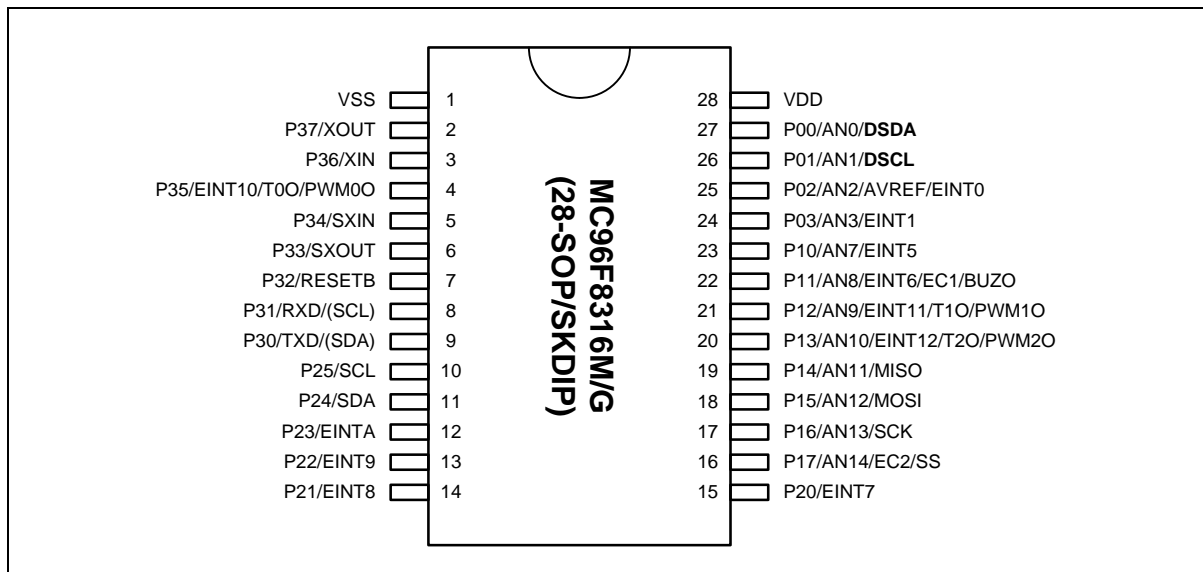


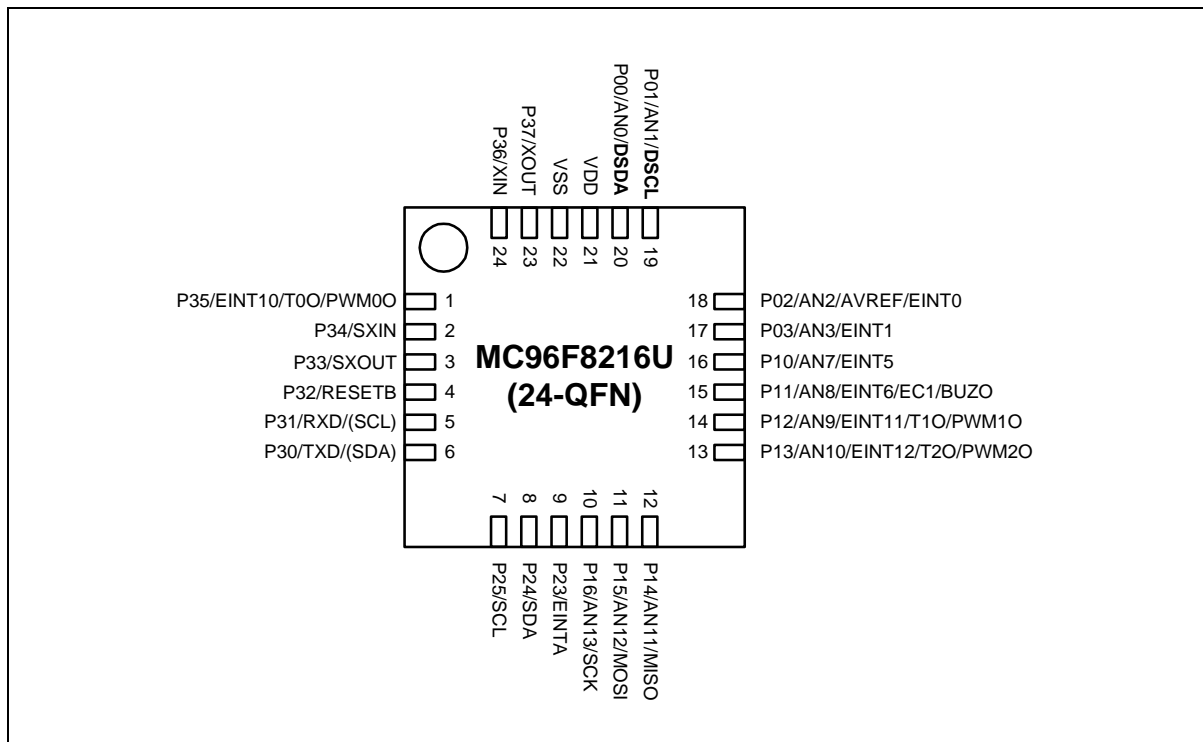
Figure 3.2 MC96F8316U 32QFN Pin Assignment

- NOTES) 1. On On-Chip Debugging, ISP uses P0[1:0] pin as DSCL, DSDA.  
2. The SDA/SCL lines for I2C can be configured on the P30/P31 pins by software control.



**Figure 3.3 MC96F8316M/G 28SOP/28SKDIP Pin Assignment**

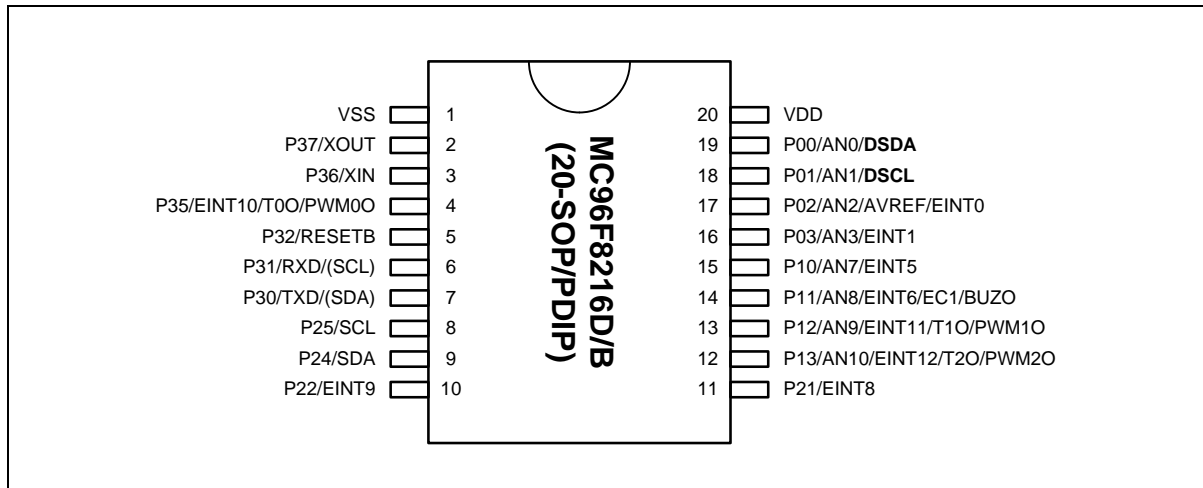
- NOTES) 1. On On-Chip Debugging, ISP uses P0[1:0] pin as DSCL, DSDA.  
 2. The P04-P06 and P26 pins should be selected as a push-pull output or an input with pull-up resistor by software control when the 28-pin package is used.  
 3. The SDA/SCL lines for I2C can be configured on the P30/P31 pins by software control.



**Figure 3.4 MC96F8216U 24QFN Pin Assignment**

- NOTES) 1. On On-Chip Debugging, ISP uses P0[1:0] pin as DSCL, DSDA.  
 2. The P04-P06, P17, P20-P22 and P26 pins should be selected as a push-pull output or an input with pull-up resistor by software control when the 24-pin package is used.  
 3. The SDA/SCL lines for I2C can be configured on the P30/P31 pins by software control.





**Figure 3.5 MC96F8216D/B 20SOP/20PDIP Pin Assignment**

- NOTES) 1. On On-Chip Debugging, ISP uses P0[1:0] pin as DSCL, DSDA.  
 2. The P04-P06, P14-P17, P20, P23, P26 and P33-P34 pins should be selected as a push-pull output or an input with pull-up resistor by software control when the 20-pin package is used.  
 3. The SDA/SCL lines for I2C can be configured on the P30/P31 pins by software control.

#### 4. Package Diagram

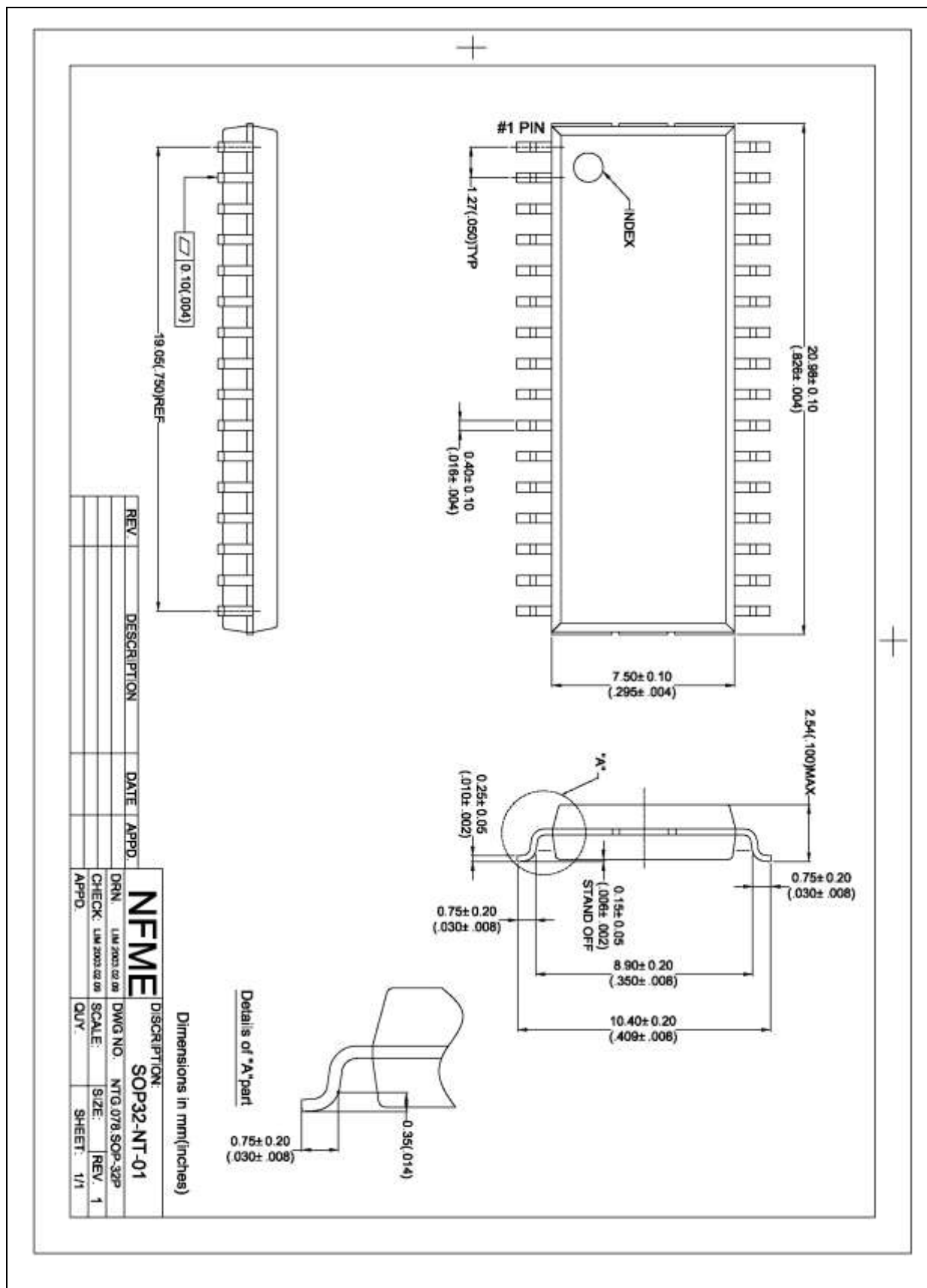


Figure 4.1 32-Pin SOP Package

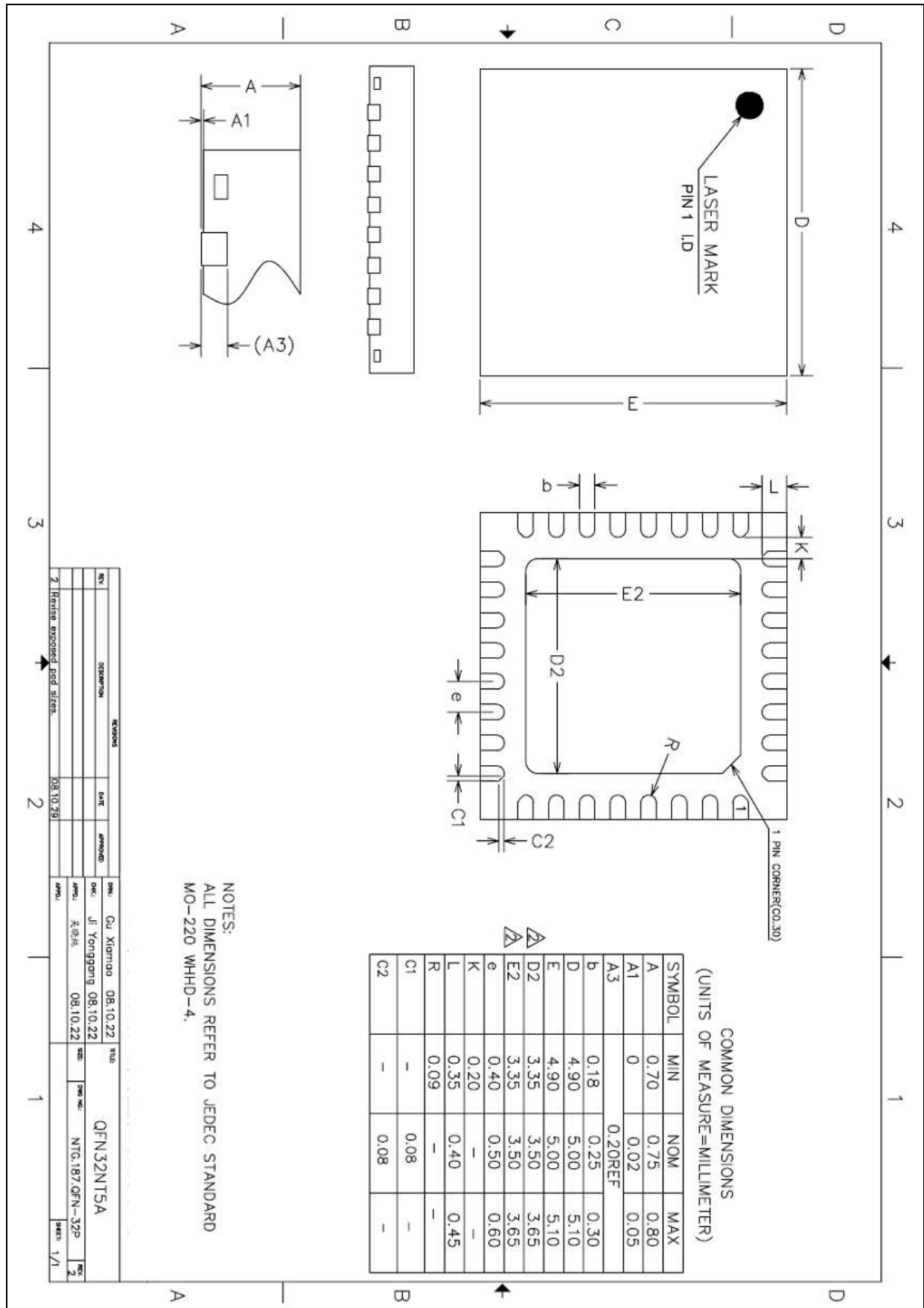


Figure 4.2 32-Pin QFN Package

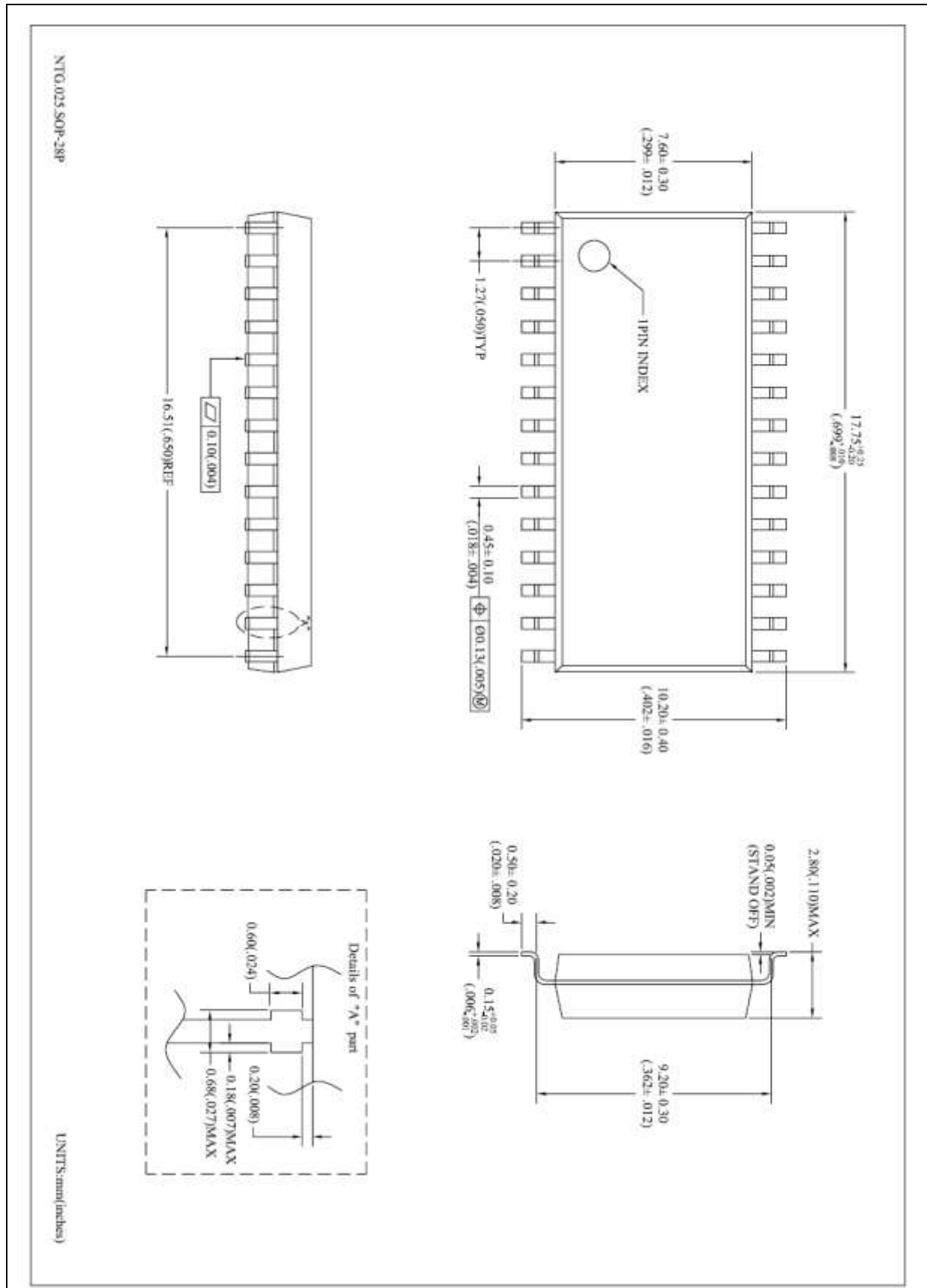


Figure 4.3 28-Pin SOP Package

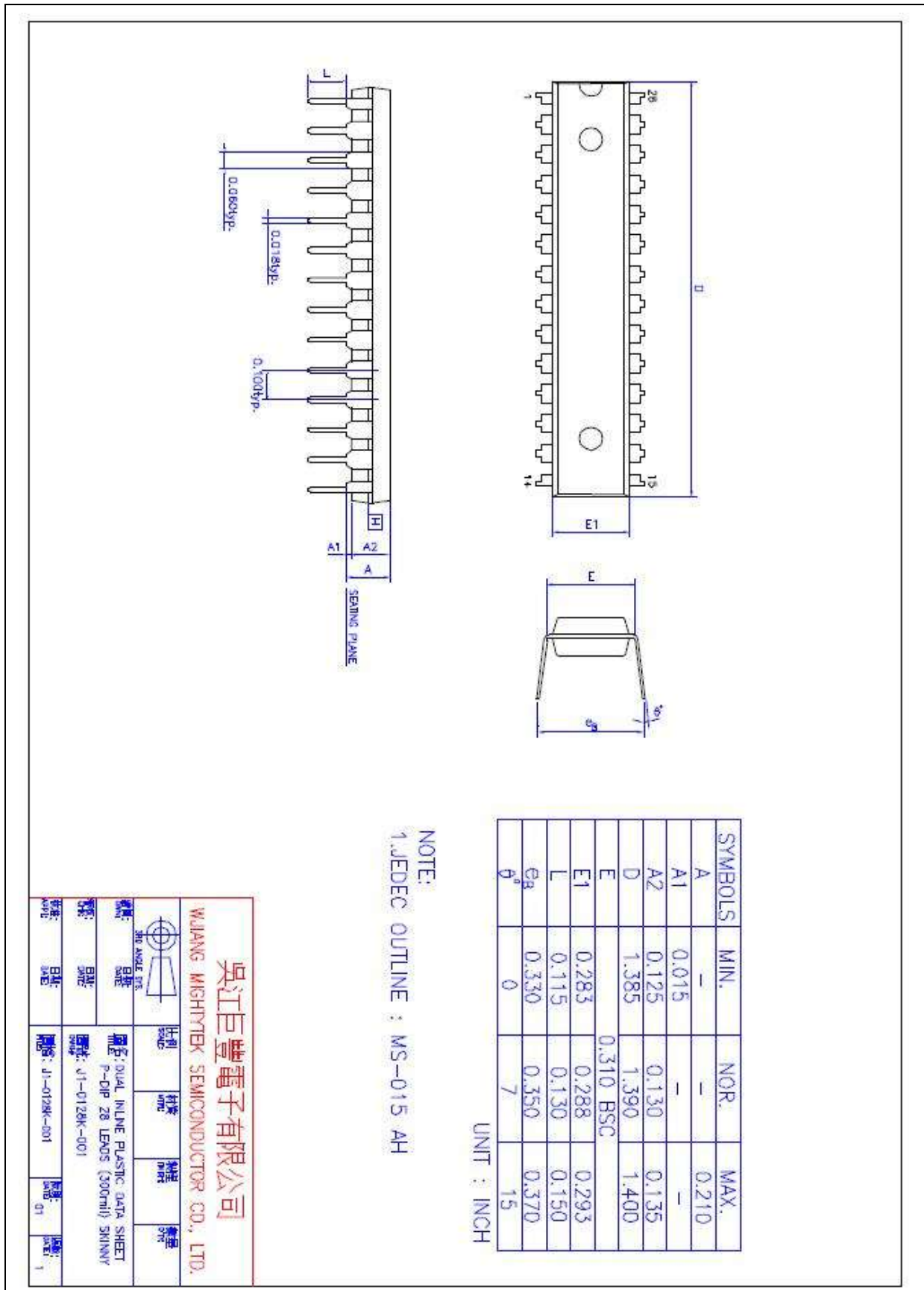


Figure 4.4 28-Pin SKDIP Package

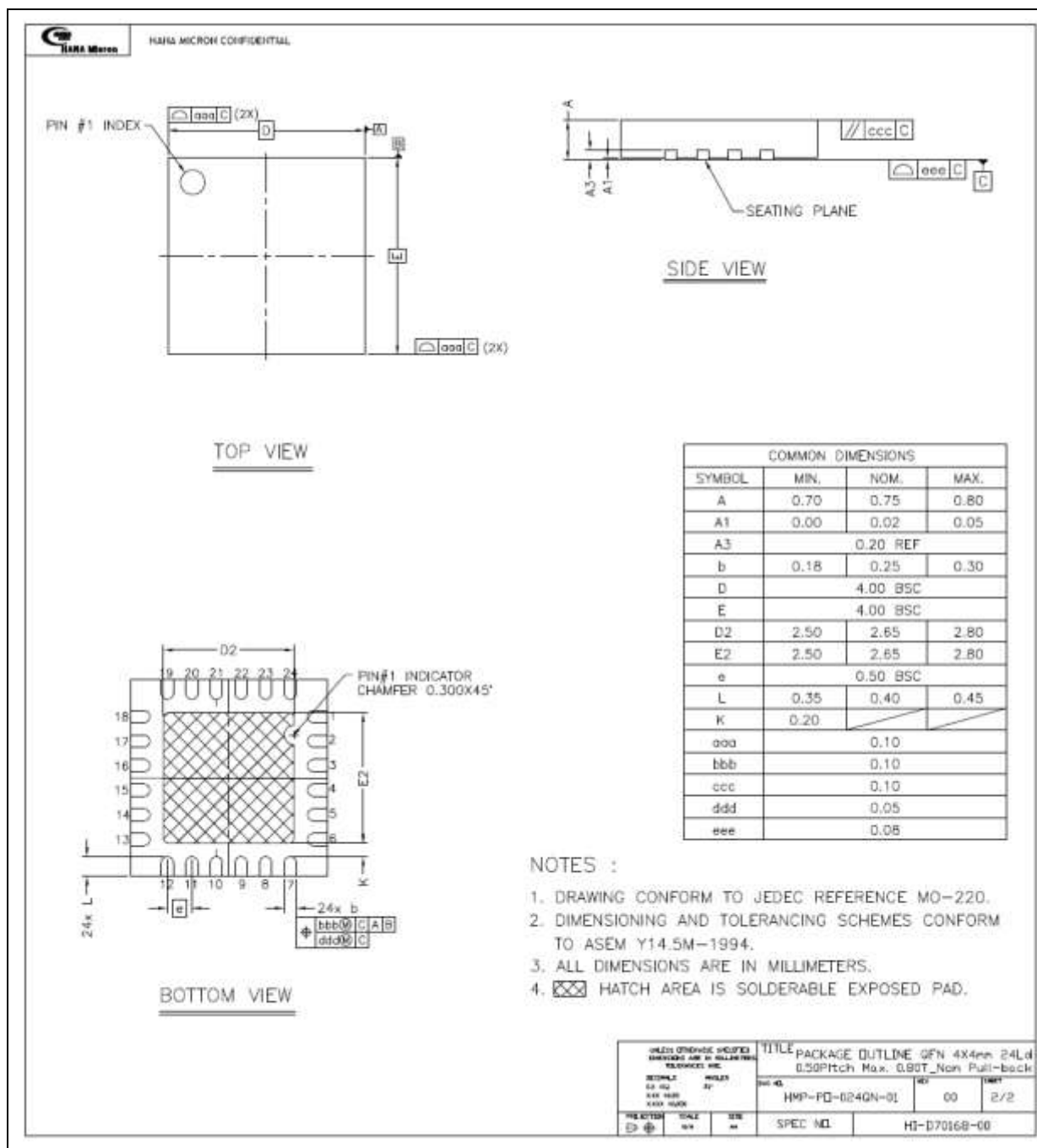


Figure 4.5 24-Pin QFN Package

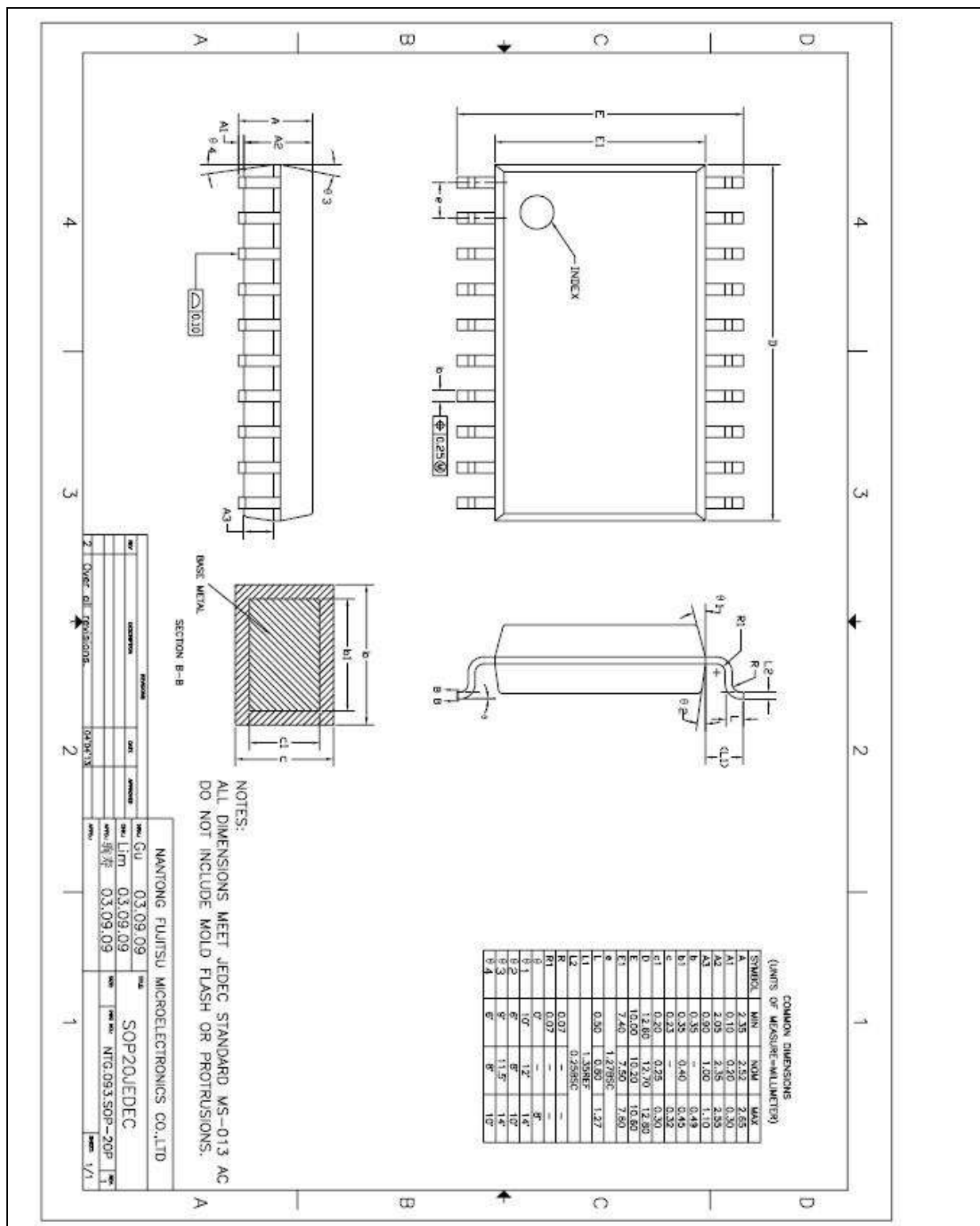


Figure 4.6 20-Pin SOP Package





## 5. Pin Description

**Table 5-1 Normal Pin Description**

| PIN Name | I/O | Function                                                                                                                                                                                                                                                                                                                     | @RESET | Shared with           |
|----------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|-----------------------|
| P00      | I/O | Port 0 is a bit-programmable I/O port which can be configured as a schmitt-trigger input, a push-pull output, or an open-drain output. A pull-up resistor can be specified in 1-bit unit. The P04 – P06 are only in the 32-Pin package.                                                                                      | Input  | AN0/DSDA              |
| P01      |     |                                                                                                                                                                                                                                                                                                                              |        | AN1/DSCL              |
| P02      |     |                                                                                                                                                                                                                                                                                                                              |        | AN2/AVREF/EINT0       |
| P03      |     |                                                                                                                                                                                                                                                                                                                              |        | AN3/EINT1             |
| P04      |     |                                                                                                                                                                                                                                                                                                                              |        | AN4/EINT2             |
| P05      |     |                                                                                                                                                                                                                                                                                                                              |        | AN5/EINT3             |
| P06      |     |                                                                                                                                                                                                                                                                                                                              |        | AN6/EINT4             |
| P10      | I/O | Port 1 is a bit-programmable I/O port which can be configured as a schmitt-trigger input, a push-pull output, or an open-drain output. A pull-up resistor can be specified in 1-bit unit. The P17 is not in the 24-Pin package. The P14 – P17 are not in the 20-Pin package.                                                 | Input  | AN7/EINT5             |
| P11      |     |                                                                                                                                                                                                                                                                                                                              |        | AN8/EINT6/EC1/BUZO    |
| P12      |     |                                                                                                                                                                                                                                                                                                                              |        | AN9/EINT11/T1O/PWM1O  |
| P13      |     |                                                                                                                                                                                                                                                                                                                              |        | AN10/EINT12/T2O/PWM2O |
| P14      |     |                                                                                                                                                                                                                                                                                                                              |        | AN11/MISO             |
| P15      |     |                                                                                                                                                                                                                                                                                                                              |        | AN12/MOSI             |
| P16      |     |                                                                                                                                                                                                                                                                                                                              |        | AN13/SCK              |
| P17      |     |                                                                                                                                                                                                                                                                                                                              |        | AN14/EC2/SS           |
| P20      | I/O | Port 2 is a bit-programmable I/O port which can be configured as a schmitt-trigger input, a push-pull output, or an open-drain output. A pull-up resistor can be specified in 1-bit unit. The P20 – P22 are not in the 24-Pin package. The P20 and P23 are not in the 20-Pin package. The P26 is only in the 32-Pin package. | Input  | EINT7                 |
| P21      |     |                                                                                                                                                                                                                                                                                                                              |        | EINT8                 |
| P22      |     |                                                                                                                                                                                                                                                                                                                              |        | EINT9                 |
| P23      |     |                                                                                                                                                                                                                                                                                                                              |        | EINTA                 |
| P24      |     |                                                                                                                                                                                                                                                                                                                              |        | SDA                   |
| P25      |     |                                                                                                                                                                                                                                                                                                                              |        | SCL                   |
| P26      |     |                                                                                                                                                                                                                                                                                                                              |        | EC0                   |
| P30      | I/O | Port 3 is a bit-programmable I/O port which can be configured as a schmitt-trigger input, a push-pull output, or an open-drain output. A pull-up resistor can be specified in 1-bit unit. The P33 – P34 are not in the 20-Pin package.                                                                                       | Input  | TXD                   |
| P31      |     |                                                                                                                                                                                                                                                                                                                              |        | RXD                   |
| P32      |     |                                                                                                                                                                                                                                                                                                                              |        | RESETB                |
| P33      |     |                                                                                                                                                                                                                                                                                                                              |        | SXOUT                 |
| P34      |     |                                                                                                                                                                                                                                                                                                                              |        | SXIN                  |
| P35      |     |                                                                                                                                                                                                                                                                                                                              |        | EINT10/T0O/PWM0O      |
| P36      |     |                                                                                                                                                                                                                                                                                                                              |        | XIN                   |
| P37      |     |                                                                                                                                                                                                                                                                                                                              |        | XOUT                  |

Table 5-1 Normal Pin Description (Continued)

| PIN Name | I/O | Function                                           | @RESET | Shared with           |
|----------|-----|----------------------------------------------------|--------|-----------------------|
| EINT0    | I/O | External interrupt inputs                          | Input  | P02/AN2/AVREF         |
| EINT1    |     |                                                    |        | P03/AN3               |
| EINT2    |     |                                                    |        | P04/AN4               |
| EINT3    |     |                                                    |        | P05/AN5               |
| EINT4    |     |                                                    |        | P06/AN6               |
| EINT5    |     |                                                    |        | P10/AN7               |
| EINT6    |     |                                                    |        | P11/AN8/EC1/BUZO      |
| EINT7    |     |                                                    |        | P20                   |
| EINT8    |     |                                                    |        | P21                   |
| EINT9    |     |                                                    |        | P22                   |
| EINTA    |     |                                                    |        | P23                   |
| EINT10   | I/O | External interrupt input and Timer 0 capture input | Input  | P35/T00/PWM00         |
| EINT11   | I/O | External interrupt input and Timer 1 capture input | Input  | P12/AN9/T10/PWM10     |
| EINT12   | I/O | External interrupt input and Timer 2 capture input | Input  | P13/AN10/T20/PWM20    |
| T00      | I/O | Timer 0 interval output                            | Input  | P35/EINT10/PWM00      |
| T10      | I/O | Timer 1 interval output                            | Input  | P12/AN9/EINT11/PWM10  |
| T20      | I/O | Timer 2 interval output                            | Input  | P13/AN10/EINT12/PWM20 |
| PWM00    | I/O | Timer 0 PWM output                                 | Input  | P35/EINT10/T00        |
| PWM10    | I/O | Timer 1 PWM output                                 | Input  | P12/AN9/EINT11/T10    |
| PWM20    | I/O | Timer 2 PWM output                                 | Input  | P13/AN10/EINT12/T20   |
| EC0      | I/O | Timer 0 event count input                          | Input  | P26                   |
| EC1      | I/O | Timer 1 event count input                          | Input  | P11/AN8/EINT6/BUZO    |
| EC2      | I/O | Timer 2 event count input                          | Input  | P17/AN14/SS           |
| BUZO     | I/O | Buzzer signal output                               | Input  | P11/AN8/EINT6/EC1     |
| SCK      | I/O | Serial clock input/output                          | Input  | P16/AN13              |
| MISO     | I/O | Serial data input/output                           | Input  | P14/AN11              |
| MOSI     | I/O | Serial data input/output                           | Input  | P15/AN12              |
| SS       | I/O | Slave select input                                 | Input  | P17/AN14/EC2          |
| TXD      | I/O | UART data output                                   | Input  | P30                   |
| RXD      | I/O | UART data input                                    | Input  | P31                   |
| SCL      | I/O | I2C clock input/output                             | Input  | P25                   |
| SDA      | I/O | I2C data input/output                              | Input  | P24                   |

Table 5-1 Normal Pin Description (Continued)

| PIN Name | I/O | Function                                                                                       | @RESET | Shared with          |
|----------|-----|------------------------------------------------------------------------------------------------|--------|----------------------|
| AVREF    | I/O | A/D converter reference voltage                                                                | Input  | P02/AN2/EINT0        |
| AN0      | I/O | A/D converter analog input channels                                                            | Input  | P00/DSDA             |
| AN1      |     |                                                                                                |        | P01/DSCL             |
| AN2      |     |                                                                                                |        | P02/AVREF/EINT0      |
| AN3      |     |                                                                                                |        | P03/EINT1            |
| AN4      |     |                                                                                                |        | P04/EINT2            |
| AN5      |     |                                                                                                |        | P05/EINT3            |
| AN6      |     |                                                                                                |        | P06/EINT4            |
| AN7      |     |                                                                                                |        | P10/EINT5            |
| AN8      |     |                                                                                                |        | P11/EINT6/EC1/BUZO   |
| AN9      |     |                                                                                                |        | P12/EINT11/T1O/PWM1O |
| AN10     |     |                                                                                                |        | P13/EINT12/T2O/PWM2O |
| AN11     |     |                                                                                                |        | P14/MISO             |
| AN12     |     |                                                                                                |        | P15/MOSI             |
| AN13     |     |                                                                                                |        | P16/SCK              |
| AN14     |     |                                                                                                |        | P17/EC2/SS           |
| RESETB   | I/O | System reset pin with a pull-up resistor when it is selected as the RESETB by CONFIGURE OPTION | Input  | P32                  |
| DSDA     | I/O | On chip debugger data input/output <sup>(NOTE5,6)</sup>                                        | Input  | P00                  |
| DSCL     | I/O | On chip debugger clock input <sup>(NOTE5,6)</sup>                                              | Input  | P01                  |
| XIN      | I/O | Main oscillator pins                                                                           | Input  | P36                  |
| XOUT     |     |                                                                                                |        | P37                  |
| SXIN     | I/O | Sub oscillator pins                                                                            | Input  | P34                  |
| SXOUT    |     |                                                                                                |        | P33                  |
| VDD, VSS | —   | Power input pins                                                                               | —      | —                    |

- NOTES) 1. The P04-P06 and P26 are not in the 28-Pin package.  
2. The P04-P06, P17, P20-P22, and P26 are not in the 24-Pin package.  
3. The P04-P06, P14-P17, P20, P23, P26 and P33-P34 are not in the 20-Pin package.  
4. The P32/RESETB pin is configured as one of the P32 and the RESETB pin by the "CONFIGURE OPTION".  
5. If the P00/DSDA and P01/DSCL pins are connected to an emulator during power-on reset, the pins are automatically configured as the debugger pins.  
6. The P00/DSDA and P01/DSCL pins are configured as inputs with internal pull-up resistor only during the reset or power-on reset.  
7. The P37/XOUT, P36/XIN, P34/SXIN, and P33/SXOUT pins are configured as a function pin by software control.

## 6. Port Structures

### 6.1 General Purpose I/O Port

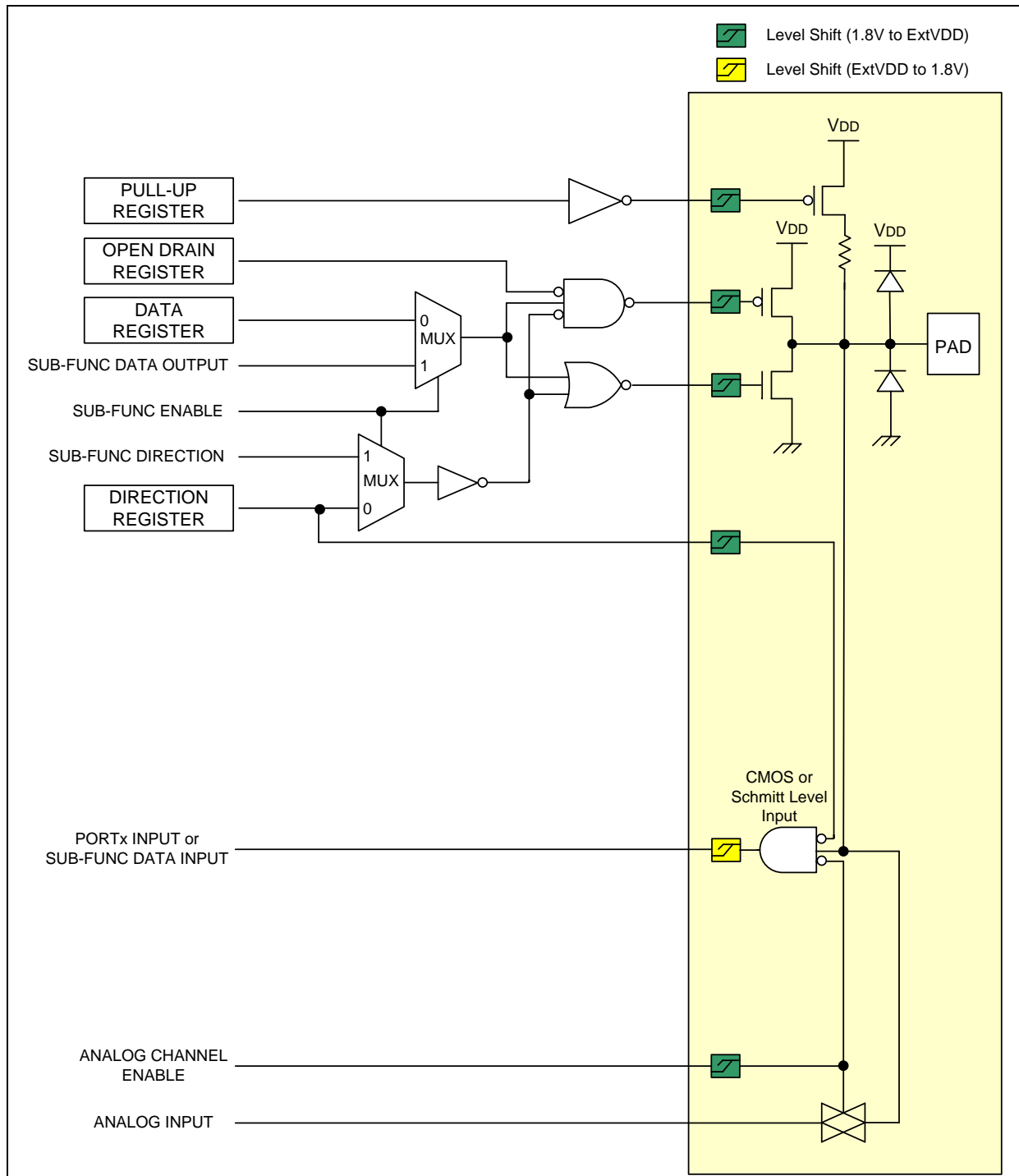
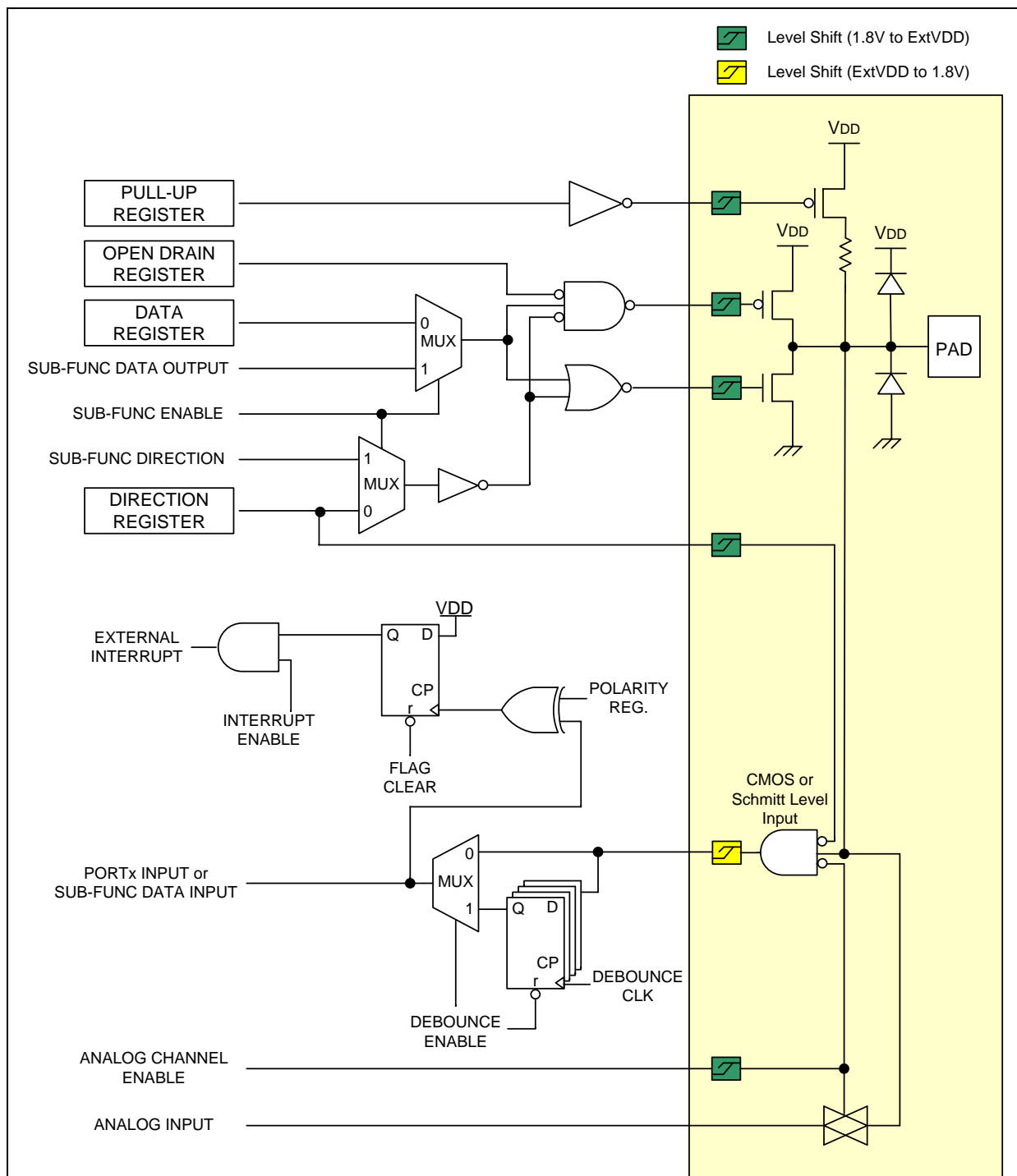


Figure 6.1 General Purpose I/O Port

## 6.2 External Interrupt I/O Port



### Figure 6.2 External Interrupt I/O Port

## 7. Electrical Characteristics

### 7.1 Absolute Maximum Ratings

**Table 7-1 Absolute Maximum Ratings**

| Parameter               | Symbol           | Rating         | Unit | Note                                                            |
|-------------------------|------------------|----------------|------|-----------------------------------------------------------------|
| Supply Voltage          | VDD              | -0.3 ~ +6.5    | V    | —                                                               |
| Normal Voltage Pin      | V <sub>I</sub>   | -0.3 ~ VDD+0.3 | V    | Voltage on any pin with respect to VSS                          |
|                         | V <sub>O</sub>   | -0.3 ~ VDD+0.3 | V    |                                                                 |
|                         | I <sub>OH</sub>  | -25            | mA   | Maximum current output sourced by (I <sub>OH</sub> per I/O pin) |
|                         | ΣI <sub>OH</sub> | -200           | mA   | Maximum current (ΣI <sub>OH</sub> )                             |
|                         | I <sub>OL</sub>  | 180            | mA   | Maximum current sunk by (I <sub>OL</sub> per I/O pin)           |
|                         | ΣI <sub>OL</sub> | 200            | mA   | Maximum current (ΣI <sub>OL</sub> )                             |
| Total Power Dissipation | P <sub>T</sub>   | 600            | mW   | —                                                               |
| Storage Temperature     | T <sub>STG</sub> | -65 ~ +150     | °C   | —                                                               |

NOTE) Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

### 7.2 Recommended Operating Conditions

**Table 7-2 Recommended Operating Conditions**

(T<sub>A</sub> = -40°C ~ +85°C)

| Parameter             | Symbol           | Conditions                     |             | MIN | TYP | MAX | Unit |
|-----------------------|------------------|--------------------------------|-------------|-----|-----|-----|------|
| Operating Voltage     | VDD              | f <sub>X</sub> = 32 ~ 38kHz    | SX-tal      | 1.8 | —   | 5.5 | V    |
|                       |                  | f <sub>X</sub> = 0.4 ~ 4.2MHz  | X-tal       | 1.8 | —   | 5.5 |      |
|                       |                  | f <sub>X</sub> = 0.4 ~ 10.0MHz |             | 2.7 | —   | 5.5 |      |
|                       |                  | f <sub>X</sub> = 0.4 ~ 12.0MHz |             | 3.0 | —   | 5.5 |      |
|                       |                  | f <sub>X</sub> = 0.5 ~ 8.0MHz  | Internal RC | 1.8 | —   | 5.5 |      |
|                       |                  | f <sub>X</sub> = 0.5 ~ 16.0MHz |             | 2.0 | —   | 5.5 |      |
| Operating Temperature | T <sub>OPR</sub> | VDD= 1.8 ~ 5.5V                |             | -40 | —   | 85  | °C   |

## 7.3 A/D Converter Characteristics

**Table 7-3 A/D Converter Characteristics**
 $(T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}, V_{DD} = 1.8\text{V} \sim 5.5\text{V}, V_{SS} = 0\text{V})$ 

| Parameter                    | Symbol           | Conditions                     | MIN             | TYP | MAX             | Unit |
|------------------------------|------------------|--------------------------------|-----------------|-----|-----------------|------|
| Resolution                   | —                | —                              | —               | 12  | —               | bit  |
| Integral Linear Error        | ILE              | AVREF= 2.7V – 5.5V<br>fx= 8MHz | —               | —   | ±4              | LSB  |
| Differential Linearity Error | DLE              |                                | —               | —   | ±1              |      |
| Zero Offset Error            | ZOE              |                                | —               | —   | ±3              |      |
| Full Scale Error             | FSE              |                                | —               | —   | ±3              |      |
| Conversion Time              | t <sub>CON</sub> | 12bit resolution, 8MHz         | 20              | —   | —               | μs   |
| Analog Input Voltage         | V <sub>AN</sub>  | —                              | V <sub>SS</sub> | —   | AVREF           | V    |
| Analog Reference Voltage     | AVREF            | —                              | 1.8             | —   | V <sub>DD</sub> |      |
| VDD18                        | —                | —                              | —               | 1.8 | —               | V    |
| Analog Input Leakage Current | I <sub>AN</sub>  | AVREF= 5.12V                   | —               | —   | 2               | μA   |
| ADC Operating Current        | I <sub>ADC</sub> | Enable                         | —               | 1   | 2               | mA   |
|                              |                  | Disable                        | —               | —   | 0.1             | μA   |

NOTES) 1. Zero offset error is the difference between 000000000000 and the converted output for zero input voltage (V<sub>SS</sub>).

2. Full scale error is the difference between 111111111111 and the converted output for full-scale input voltage (AVREF).

## 7.4 Power-On Reset Characteristics

**Table 7-4 Power-on Reset Characteristics**
 $(T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}, V_{DD} = 1.8\text{V} \sim 5.5\text{V}, V_{SS} = 0\text{V})$ 

| Parameter               | Symbol           | Conditions | MIN  | TYP | MAX  | Unit |
|-------------------------|------------------|------------|------|-----|------|------|
| RESET Release Level     | V <sub>POR</sub> | —          | —    | 1.4 | —    | V    |
| VDD Voltage Rising Time | t <sub>R</sub>   | —          | 0.05 | —   | 30.0 | V/ms |
| POR Current             | I <sub>POR</sub> | —          | —    | 0.2 | —    | μA   |

## 7.5 Low Voltage Reset and Low Voltage Indicator Characteristics

Table 7-5 LVR and LVI Characteristics

(T<sub>A</sub>= -40°C ~ +85°C, VDD= 1.8V ~ 5.5V, VSS= 0V)

| Parameter           | Symbol                               | Conditions                                                                  |         | MIN  | TYP  | MAX  | Unit |
|---------------------|--------------------------------------|-----------------------------------------------------------------------------|---------|------|------|------|------|
| Detection Level     | V <sub>LVR</sub><br>V <sub>LVI</sub> | The LVR can select all levels but LVI can select other levels except 1.60V. |         | –    | 1.60 | 1.75 | V    |
|                     |                                      |                                                                             |         | 1.85 | 2.00 | 2.15 |      |
|                     |                                      |                                                                             |         | 1.95 | 2.10 | 2.25 |      |
|                     |                                      |                                                                             |         | 2.05 | 2.20 | 2.35 |      |
|                     |                                      |                                                                             |         | 2.17 | 2.32 | 2.47 |      |
|                     |                                      |                                                                             |         | 2.29 | 2.44 | 2.59 |      |
|                     |                                      |                                                                             |         | 2.39 | 2.59 | 2.79 |      |
|                     |                                      |                                                                             |         | 2.55 | 2.75 | 2.95 |      |
|                     |                                      |                                                                             |         | 2.73 | 2.93 | 3.13 |      |
|                     |                                      |                                                                             |         | 2.94 | 3.14 | 3.34 |      |
|                     |                                      |                                                                             |         | 3.18 | 3.38 | 3.58 |      |
|                     |                                      |                                                                             |         | 3.37 | 3.67 | 3.97 |      |
|                     |                                      |                                                                             |         | 3.70 | 4.00 | 4.30 |      |
|                     |                                      |                                                                             |         | 4.10 | 4.40 | 4.70 |      |
| Hysteresis          | ΔV                                   | –                                                                           |         | –    | 50   | 150  | mV   |
| Minimum Pulse Width | t <sub>LW</sub>                      | –                                                                           |         | 100  | –    | –    | μS   |
| LVR and LVI Current | I <sub>BL</sub>                      | Enable (Both)                                                               | VDD= 3V | –    | 10.0 | 15.0 | μA   |
|                     |                                      | Enable (One of two)                                                         |         | –    | 8.0  | 12.0 |      |
|                     |                                      | Disable (Both)                                                              |         | –    | –    | 0.1  |      |



## 7.6 Internal RC Oscillator Characteristics

**Table 7-6 Internal RC Oscillator Characteristics**

( $T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ ,  $V_{SS} = 0\text{V}$ )

| Parameter          | Symbol    | Conditions                                                  | MIN | TYP | MAX     | Unit          |
|--------------------|-----------|-------------------------------------------------------------|-----|-----|---------|---------------|
| Frequency          | $f_{IRC}$ | $V_{DD} = 2.0 - 5.5\text{ V}$                               | –   | 16  | –       | MHz           |
| Tolerance          | –         | $T_A = 0^{\circ}\text{C} \text{ to } +50^{\circ}\text{C}$   | –   | –   | $\pm 1$ | %             |
|                    |           | $T_A = -20^{\circ}\text{C} \text{ to } +85^{\circ}\text{C}$ |     |     | $\pm 2$ |               |
|                    |           | $T_A = -40^{\circ}\text{C} \text{ to } +85^{\circ}\text{C}$ |     |     | $\pm 3$ |               |
| Clock Duty Ratio   | TOD       | –                                                           | 40  | 50  | 60      | %             |
| Stabilization Time | $T_{HFS}$ | –                                                           | –   | –   | 100     | $\mu\text{S}$ |
| IRC Current        | $I_{IRC}$ | Enable                                                      | –   | 0.2 | –       | mA            |
|                    |           | Disable                                                     | –   | –   | 0.1     | $\mu\text{A}$ |

## 7.7 Internal Watch-Dog Timer RC Oscillator Characteristics

**Table 7-7 Internal WDTRC Oscillator Characteristics**

( $T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ ,  $V_{SS} = 0\text{V}$ )

| Parameter          | Symbol      | Conditions | MIN | TYP | MAX | Unit          |
|--------------------|-------------|------------|-----|-----|-----|---------------|
| Frequency          | $f_{WDTRC}$ | –          | 2   | 5   | 10  | kHz           |
| Stabilization Time | $t_{WDTS}$  | –          | –   | –   | 1   | mS            |
| WDTRC Current      | $I_{WDTRC}$ | Enable     | –   | 1   | –   | $\mu\text{A}$ |
|                    |             | Disable    | –   | –   | 0.1 |               |

## 7.8 DC Characteristics

**Table 7-8 DC Characteristics**

( $T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ ,  $V_{SS} = 0\text{V}$ ,  $f_{XIN} = 12\text{MHz}$ )

| Parameter                    | Symbol      | Conditions                                                                                             |                        | MIN          | TYP  | MAX         | Unit             |
|------------------------------|-------------|--------------------------------------------------------------------------------------------------------|------------------------|--------------|------|-------------|------------------|
| Input High Voltage           | $V_{IH}$    | All input pins, RESETB                                                                                 |                        | $0.8V_{DD}$  | —    | $V_{DD}$    | V                |
| Input Low Voltage            | $V_{IL}$    | All input pins, RESETB                                                                                 |                        | —            | —    | $0.2V_{DD}$ | V                |
| Output High Voltage          | $V_{OH1}$   | $V_{DD} = 4.5\text{V}$ , $I_{OH} = -10\text{mA}$ ,<br>All output ports except $V_{OH2}$ ,<br>$V_{OH3}$ |                        | $V_{DD}-1.5$ | —    | —           | V                |
|                              | $V_{OH2}$   | $V_{DD} = 4.5\text{V}$ , $I_{OH} = -10\text{mA}$ ;<br>P20-P25                                          |                        | $V_{DD}-2.0$ | —    | —           |                  |
|                              | $V_{OH3}$   | $V_{DD} = 4.5\text{V}$ , $I_{OH} = -20\text{mA}$ ;<br>P1                                               |                        | $V_{DD}-2.0$ | —    | —           |                  |
| Output Low Voltage           | $V_{OL1}$   | $V_{DD} = 4.5\text{V}$ , $I_{OL} = 15\text{mA}$ ;<br>All output ports except $V_{OL2}$                 |                        | —            | —    | 1.0         | V                |
|                              | $V_{OL2}$   | $V_{DD} = 4.5\text{V}$ , $I_{OL} = 160\text{mA}$ ;<br>P20-P25                                          |                        | —            | 1.5  | 3.0         |                  |
| Input High Leakage Current   | $I_{IH}$    | All input ports                                                                                        |                        | —            | —    | 1           | $\mu\text{A}$    |
| Input Low Leakage Current    | $I_{IL}$    | All input ports                                                                                        |                        | -1           | —    | —           | $\mu\text{A}$    |
| Pull-Up Resistor             | $R_{PU1}$   | $V_I = 0\text{V}$ ,<br>$T_A = 25^{\circ}\text{C}$<br>All Input ports                                   | $V_{DD} = 5.0\text{V}$ | 25           | 50   | 100         | $\text{k}\Omega$ |
|                              |             |                                                                                                        | $V_{DD} = 3.0\text{V}$ | 50           | 100  | 200         |                  |
|                              | $R_{PU2}$   | $V_I = 0\text{V}$ ,<br>$T_A = 25^{\circ}\text{C}$<br>RESETB                                            | $V_{DD} = 5.0\text{V}$ | 150          | 250  | 400         | $\text{k}\Omega$ |
|                              |             |                                                                                                        | $V_{DD} = 3.0\text{V}$ | 300          | 500  | 700         |                  |
| ADC wake-up pull-up resistor | $R_{AWPU1}$ | $T_A = 25^{\circ}\text{C}$                                                                             |                        | 100          | 150  | 200         | $\text{k}\Omega$ |
|                              | $R_{AWPU2}$ |                                                                                                        |                        | 200          | 300  | 400         |                  |
| OSC feedback resistor        | $R_{X1}$    | $XIN = V_{DD}$ , $XOUT = V_{SS}$<br>$T_A = 25^{\circ}\text{C}$ , $V_{DD} = 5\text{V}$                  |                        | 600          | 1200 | 2000        | $\text{k}\Omega$ |
|                              | $R_{X2}$    | $SXIN = V_{DD}$ , $SXOUT = V_{SS}$<br>$T_A = 25^{\circ}\text{C}$ , $V_{DD} = 5\text{V}$                |                        | 2500         | 5000 | 10000       |                  |

**Table 7-9 DC Characteristics (Continued)**
 $(T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}, V_{DD} = 1.8\text{V} \sim 5.5\text{V}, V_{SS} = 0\text{V}, f_{XIN} = 12\text{MHz})$ 

| Parameter      | Symbol                     | Condition                                   |          | MIN | TYP  | MAX  | Unit |
|----------------|----------------------------|---------------------------------------------|----------|-----|------|------|------|
| Supply Current | I <sub>DD1</sub><br>(RUN)  | f <sub>XIN</sub> = 12MHz, VDD= 5V±10%       |          | –   | 3.0  | 6.0  | mA   |
|                |                            | f <sub>XIN</sub> = 10MHz, VDD= 3V±10%       |          | –   | 2.2  | 4.4  |      |
|                |                            | f <sub>IRC</sub> = 16MHz, VDD= 5V±10%       |          | –   | 3.0  | 6.0  |      |
|                | I <sub>DD2</sub><br>(IDLE) | f <sub>XIN</sub> = 12MHz, VDD= 5V±10%       |          | –   | 1.3  | 2.6  | mA   |
|                |                            | f <sub>XIN</sub> = 10MHz, VDD= 3V±10%       |          | –   | 0.7  | 1.4  |      |
|                |                            | f <sub>IRC</sub> = 16MHz, VDD= 5V±10%       |          | –   | 0.8  | 1.6  |      |
|                | I <sub>DD3</sub>           | f <sub>XIN</sub> = 32.768kHz<br>VDD= 3V±10% | Sub RUN  | –   | 60.0 | 90.0 | µA   |
|                | I <sub>DD4</sub>           | T <sub>A</sub> = 25°C                       | Sub IDLE | –   | 6.0  | 12.0 | µA   |
|                | I <sub>DD5</sub>           | STOP, VDD= 5V±10%, T <sub>A</sub> = 25°C    |          | –   | 0.5  | 3.0  | µA   |

- NOTES) 1. Where the  $f_{XIN}$  is an external main oscillator,  $f_{SUB}$  is an external sub oscillator, the  $f_{IRC}$  is an internal RC oscillator, and the  $f_x$  is the selected system clock.
2. All supply current items don't include the current of an internal Watch-dog timer RC (WDTRC) oscillator and a peripheral block.
3. All supply current items include the current of the power-on reset (POR) block.

## 7.9 AC Characteristics

Table 7-9 AC Characteristics

( $T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ )

| Parameter                                    | Symbol                     | Conditions                                               | MIN | TYP | MAX | Unit          |
|----------------------------------------------|----------------------------|----------------------------------------------------------|-----|-----|-----|---------------|
| RESETB input low width                       | $t_{RST}$                  | Input, $V_{DD} = 5\text{V}$                              | 10  | —   | —   | $\mu\text{S}$ |
| Interrupt input high, low width              | $t_{IWH}$ ,<br>$t_{IWL}$   | All interrupt, $V_{DD} = 5\text{V}$                      | 200 | —   | —   | nS            |
| External Counter Input High, Low Pulse Width | $t_{ECWH}$ ,<br>$t_{ECWL}$ | $\text{ECn}$ , $V_{DD} = 5\text{V}$<br>( $n = 0, 1, 2$ ) | 200 | —   | —   |               |
| External Counter Transition Time             | $t_{REC}$ ,<br>$t_{FEC}$   | $\text{ECn}$ , $V_{DD} = 5\text{V}$<br>( $n = 0, 1, 2$ ) | 20  | —   | —   |               |

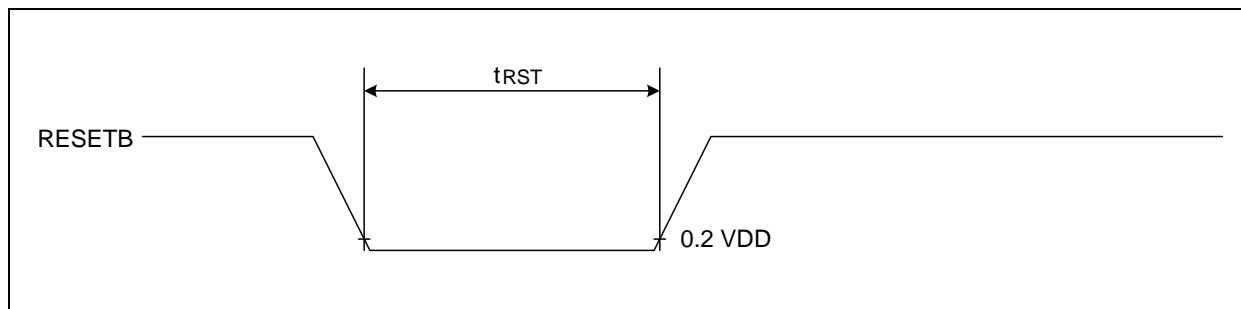


Figure 7.1 Input Timing for RESETB

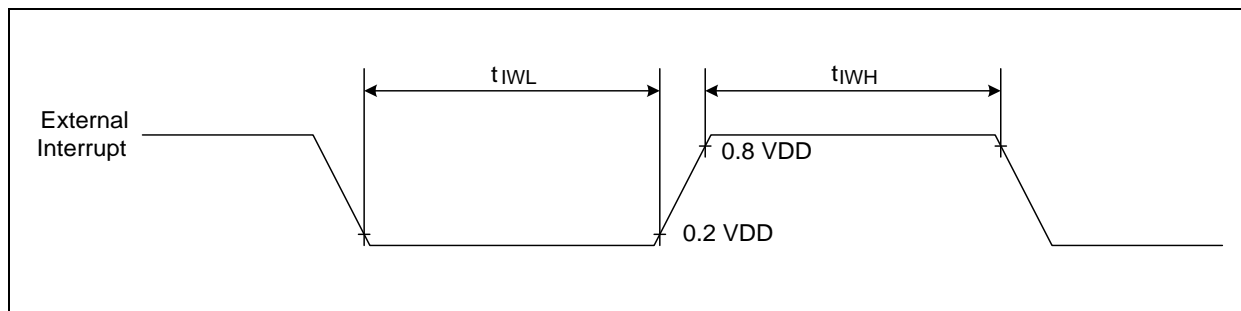


Figure 7.2 Input Timing for External Interrupts

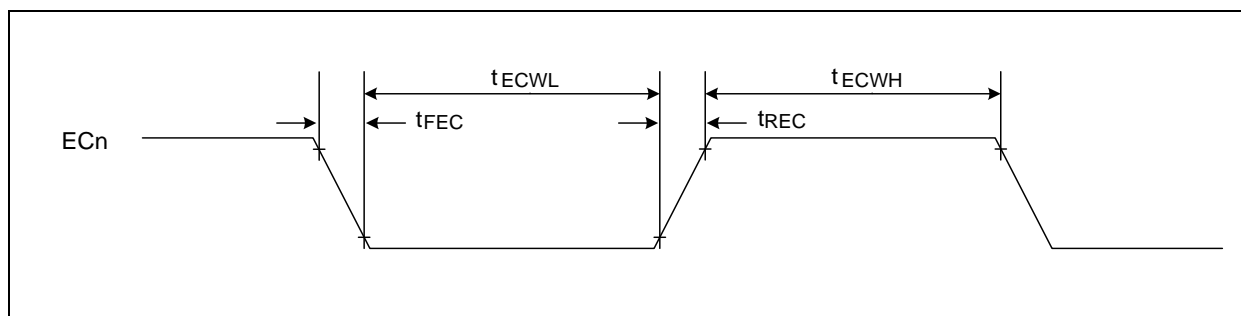


Figure 7.3 Input Timing for EC0, EC1, EC2

## 7.10 SPI Characteristics

Table 7-10 SPI Characteristics

( $T_A = -40^{\circ}\text{C} - +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} - 5.5\text{V}$ )

| Parameter                          | Symbol                             | Conditions                   | MIN | TYP | MAX | Unit |
|------------------------------------|------------------------------------|------------------------------|-----|-----|-----|------|
| Output Clock Pulse Period          | $t_{\text{SCK}}$                   | Internal SCK source          | 200 | —   | —   | nS   |
| Input Clock Pulse Period           |                                    | External SCK source          | 200 | —   | —   |      |
| Output Clock High, Low Pulse Width | $t_{\text{SCKH}}, t_{\text{SCKL}}$ | Internal SCK source          | 70  | —   | —   |      |
| Input Clock High, Low Pulse Width  |                                    | External SCK source          | 70  | —   | —   |      |
| First Output Clock Delay Time      | $t_{\text{FOD}}$                   | Internal/External SCK source | 100 | —   | —   |      |
| Output Clock Delay Time            | $t_{\text{DS}}$                    | —                            | —   | —   | 50  |      |
| Input Setup Time                   | $t_{\text{DIS}}$                   | —                            | 100 | —   | —   |      |
| Input Hold Time                    | $t_{\text{DIH}}$                   | —                            | 150 | —   | —   |      |

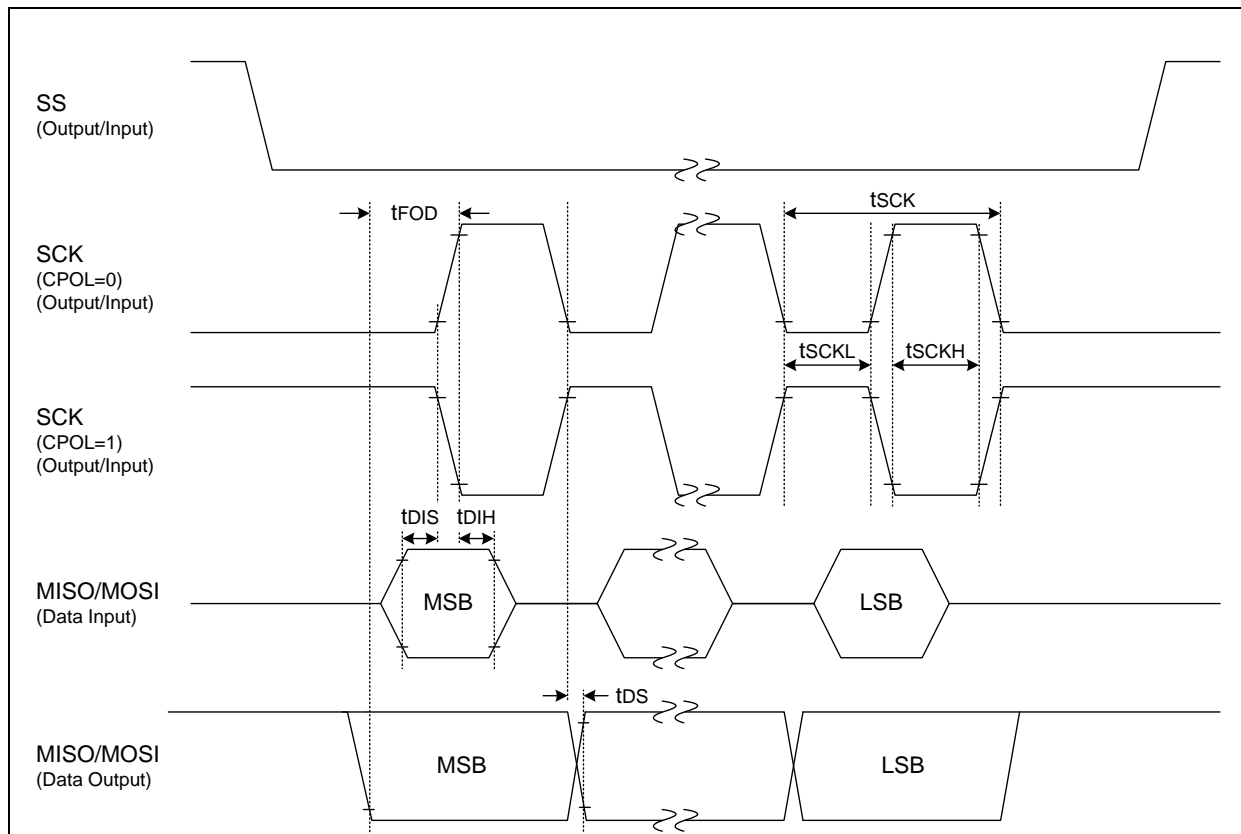


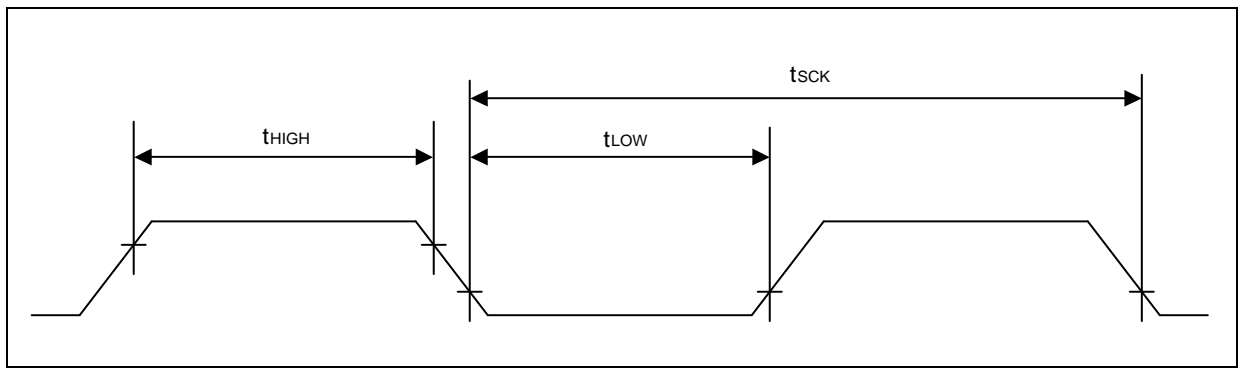
Figure 7.4 SPI Timing

## 7.11 UART Characteristics

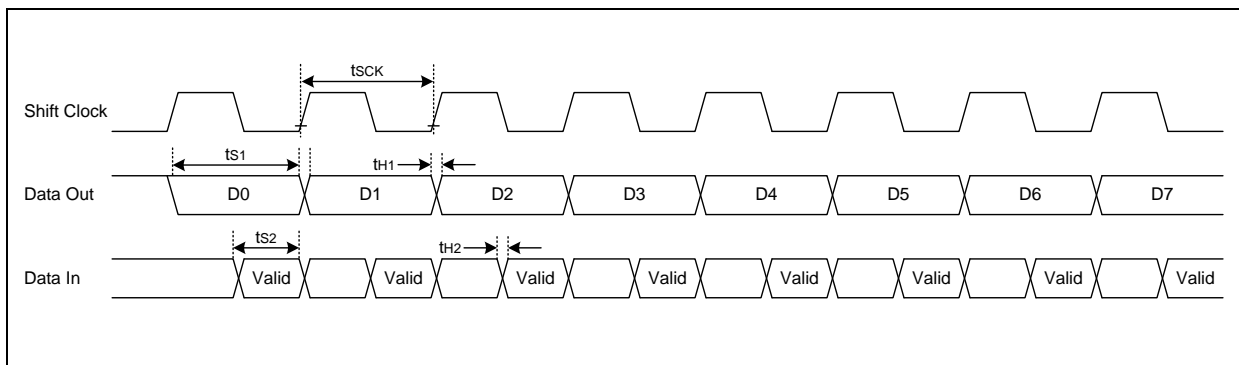
**Table 7-11 UART Characteristics**

( $T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ ,  $f_{XIN} = 11.1\text{MHz}$ )

| Parameter                                | Symbol              | MIN            | TYP                 | MAX  | Unit |
|------------------------------------------|---------------------|----------------|---------------------|------|------|
| Serial port clock cycle time             | $t_{SCK}$           | 1250           | $t_{CPU} \times 16$ | 1650 | nS   |
| Output data setup to clock rising edge   | $t_{S1}$            | 590            | $t_{CPU} \times 13$ | —    | nS   |
| Clock rising edge to input data valid    | $t_{S2}$            | —              | —                   | 590  | nS   |
| Output data hold after clock rising edge | $t_{H1}$            | $t_{CPU} - 50$ | $t_{CPU}$           | —    | nS   |
| Input data hold after clock rising edge  | $t_{H2}$            | 0              | —                   | —    | nS   |
| Serial port clock High, Low level width  | $t_{HIGH}, t_{LOW}$ | 470            | $t_{CPU} \times 8$  | 970  | nS   |



**Figure 7.5 Waveform for UART Timing Characteristics**



**Figure 7.6 Timing Waveform for the UART Module**

## 7.12 I2C Characteristics

Table 7-12 I2C Characteristics

( $T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ )

| Parameter                  | Symbol     | Standard Mode |     | High-Speed Mode |     | Unit          |
|----------------------------|------------|---------------|-----|-----------------|-----|---------------|
|                            |            | MIN           | MAX | MIN             | MAX |               |
| Clock frequency            | $t_{SCL}$  | 0             | 100 | 0               | 400 | kHz           |
| Clock High Pulse Width     | $t_{SCLH}$ | 4.0           | —   | 0.6             | —   | $\mu\text{S}$ |
| Clock Low Pulse Width      | $t_{SCLL}$ | 4.7           | —   | 1.3             | —   |               |
| Bus Free Time              | $t_{BF}$   | 4.7           | —   | 1.3             | —   |               |
| Start Condition Setup Time | $t_{STSU}$ | 4.7           | —   | 0.6             | —   |               |
| Start Condition Hold Time  | $t_{STHD}$ | 4.0           | —   | 0.6             | —   |               |
| Stop Condition Setup Time  | $t_{SPSU}$ | 4.0           | —   | 0.6             | —   |               |
| Stop Condition Hold Time   | $t_{SPHD}$ | 4.0           | —   | 0.6             | —   |               |
| Output Valid from Clock    | $t_{VD}$   | 0             | —   | 0               | —   |               |
| Data Input Hold Time       | $t_{DIH}$  | 0             | —   | 0               | 1.0 |               |
| Data Input Setup Time      | $t_{DIS}$  | 250           | —   | 100             | —   | nS            |

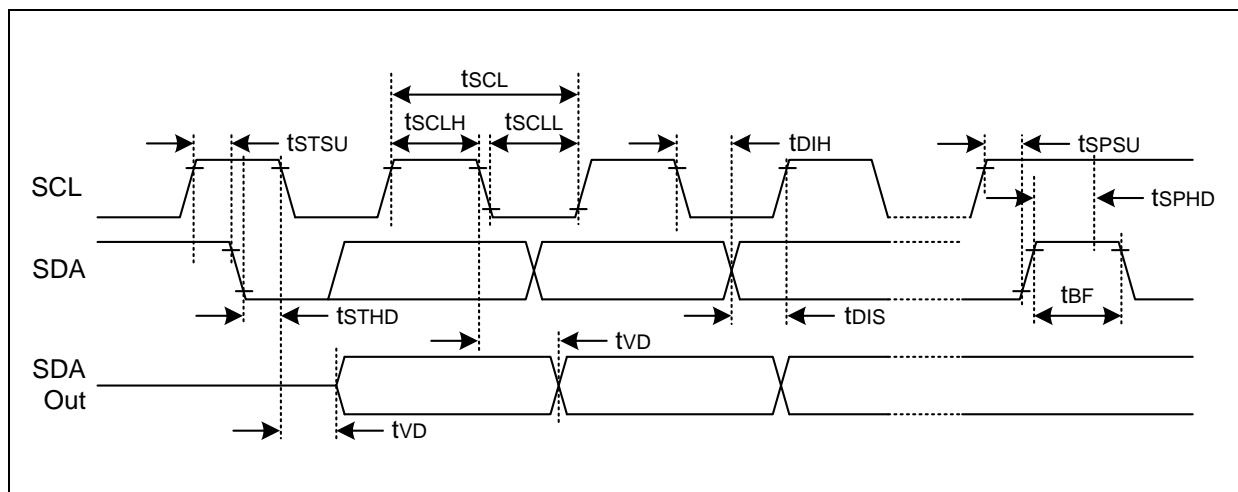


Figure 7.7 I2C Timing

### 7.13 Data Retention Voltage in Stop Mode

Table 7-13 Data Retention Voltage in Stop Mode

( $T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ )

| Parameter                     | Symbol     | Conditions                                                              | MIN | TYP | MAX | Unit          |
|-------------------------------|------------|-------------------------------------------------------------------------|-----|-----|-----|---------------|
| Data retention supply voltage | $V_{DDDR}$ | —                                                                       | 1.8 | —   | 5.5 | V             |
| Data retention supply current | $I_{DDDR}$ | $V_{DDDR} = 1.8\text{V}$ ,<br>( $T_A = 25^{\circ}\text{C}$ ), Stop mode | —   | —   | 1   | $\mu\text{A}$ |

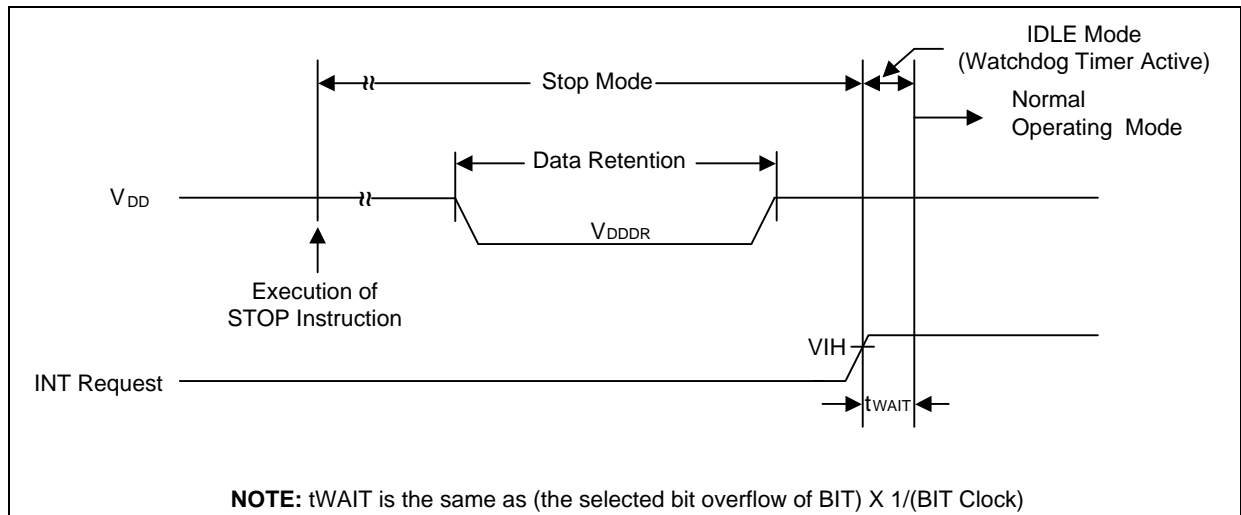


Figure 7.8 Stop Mode Release Timing when Initiated by an Interrupt

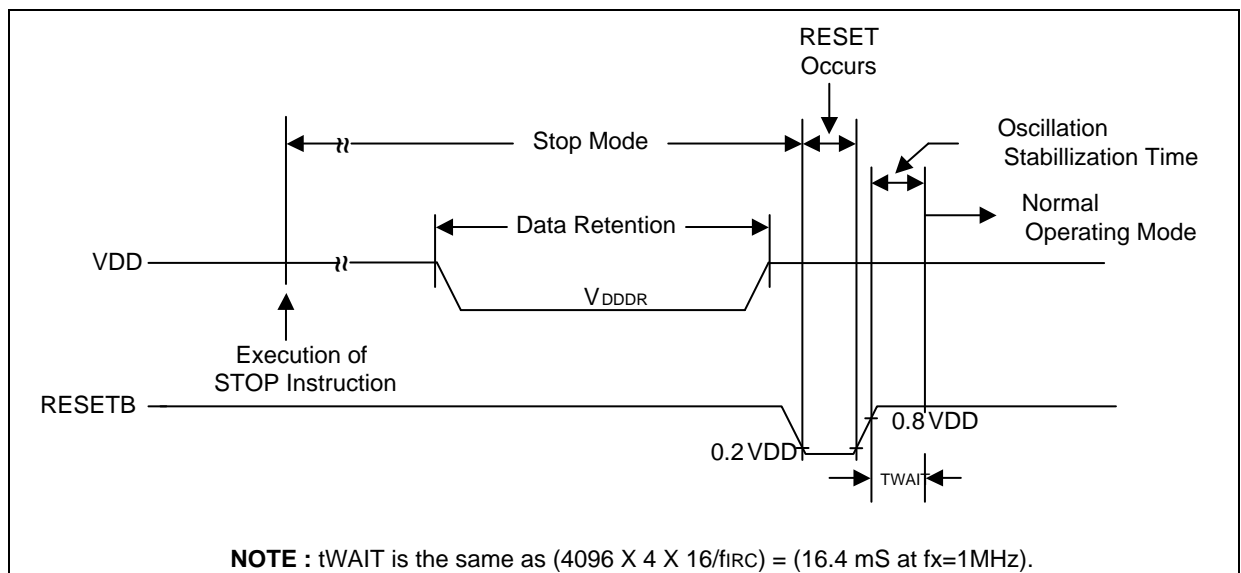


Figure 7.9 Stop Mode Release Timing when Initiated by RESETB



## 7.14 Internal Flash Rom Characteristics

**Table 7-14 Internal Flash Rom Characteristics**

(T<sub>A</sub>= -40°C ~ +85°C, VDD= 1.8V ~ 5.5V, VSS= 0V)

| Parameter                   | Symbol           | Condition | MIN | TYP | MAX     | Unit  |
|-----------------------------|------------------|-----------|-----|-----|---------|-------|
| Sector Write Time           | t <sub>FSW</sub> | —         | —   | 2.5 | 2.7     | mS    |
| Sector Erase Time           | t <sub>FSE</sub> | —         | —   | 2.5 | 2.7     |       |
| Hard-Lock Time              | t <sub>FHL</sub> | —         | —   | 2.5 | 2.7     |       |
| Page Buffer Reset Time      | t <sub>FBR</sub> | —         | —   | —   | 5       | μS    |
| Flash Programming Frequency | f <sub>PGM</sub> | —         | 0.4 | —   | —       | MHz   |
| Endurance of Write/Erase    | N <sub>FWE</sub> | —         | —   | —   | 100,000 | Times |

NOTE) During a flash operation, SCLK[1:0] of SCCR must be set to “00” or “01” (INT-RC OSC or Main X-TAL for system clock).

## 7.15 Input/Output Capacitance

**Table 7-15 Input/Output Capacitance**

(T<sub>A</sub>= -40°C ~ +85°C, VDD= 0V)

| Parameter          | Symbol           | Condition                                           | MIN | TYP | MAX | Unit |
|--------------------|------------------|-----------------------------------------------------|-----|-----|-----|------|
| Input Capacitance  | C <sub>IN</sub>  | fx= 1MHz<br>Unmeasured pins are<br>connected to VSS | —   | —   | 10  | pF   |
| Output Capacitance | C <sub>OUT</sub> |                                                     |     |     |     |      |
| I/O Capacitance    | C <sub>IO</sub>  |                                                     |     |     |     |      |

## 7.16 Main Clock Oscillator Characteristics

Table 7-16 Main Clock Oscillator Characteristics

( $T_A = -40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ )

| Oscillator         | Parameter                  | Condition   | MIN | TYP | MAX  | Unit |
|--------------------|----------------------------|-------------|-----|-----|------|------|
| Crystal            | Main oscillation frequency | 1.8V – 5.5V | 0.4 | –   | 4.2  | MHz  |
|                    |                            | 2.7V – 5.5V | 0.4 | –   | 10.0 |      |
|                    |                            | 3.0V – 5.5V | 0.4 | –   | 12.0 |      |
| Ceramic Oscillator | Main oscillation frequency | 1.8V – 5.5V | 0.4 | –   | 4.2  | MHz  |
|                    |                            | 2.7V – 5.5V | 0.4 | –   | 10.0 |      |
|                    |                            | 3.0V – 5.5V | 0.4 | –   | 12.0 |      |
| External Clock     | XIN input frequency        | 1.8V – 5.5V | 0.4 | –   | 4.2  | MHz  |
|                    |                            | 2.7V – 5.5V | 0.4 | –   | 10.0 |      |
|                    |                            | 3.0V – 5.5V | 0.4 | –   | 12.0 |      |

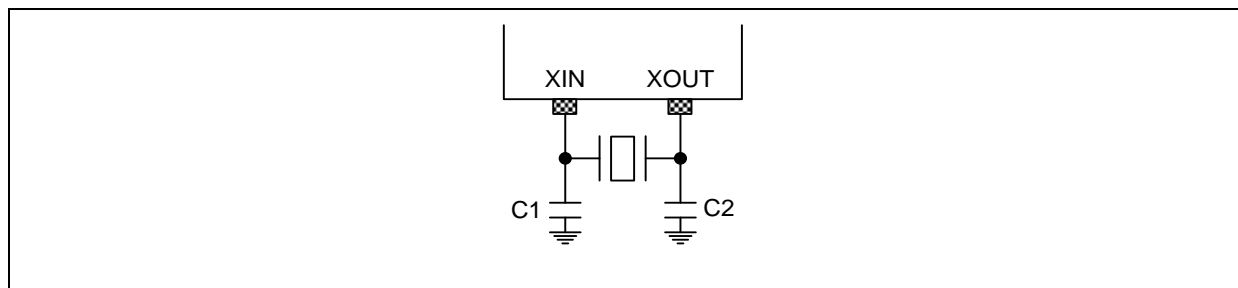


Figure 7.10 Crystal/Ceramic Oscillator

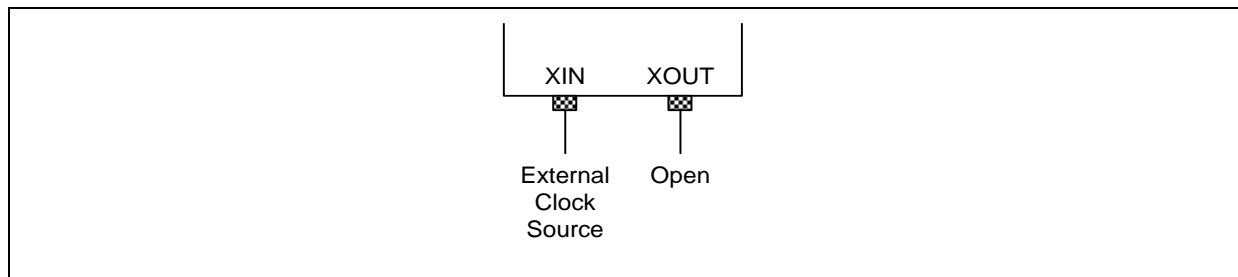


Figure 7.11 External Clock

## 7.17 Sub Clock Oscillator Characteristics

Table 7-17 Sub Clock Oscillator Characteristics

(T<sub>A</sub>= -40°C ~ +85°C, VDD= 1.8V ~ 5.5V)

| Oscillator     | Parameter                 | Condition   | MIN | TYP    | MAX | Unit |
|----------------|---------------------------|-------------|-----|--------|-----|------|
| Crystal        | Sub oscillation frequency | 1.8V – 5.5V | 32  | 32.768 | 38  | kHz  |
| External Clock | SXIN input frequency      |             | 32  | –      | 100 | kHz  |

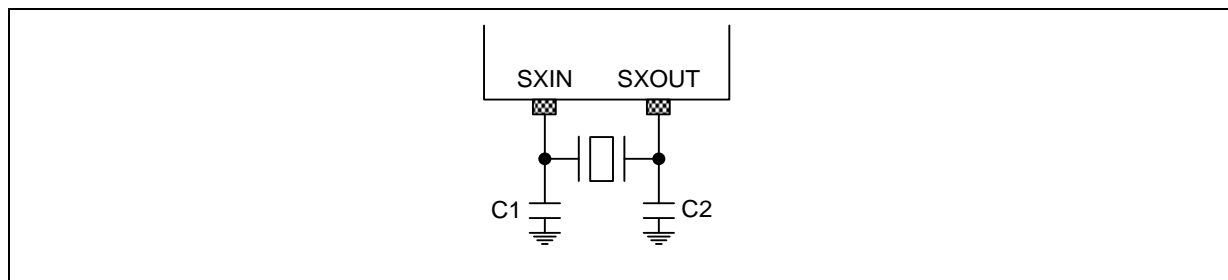


Figure 7.12 Crystal Oscillator

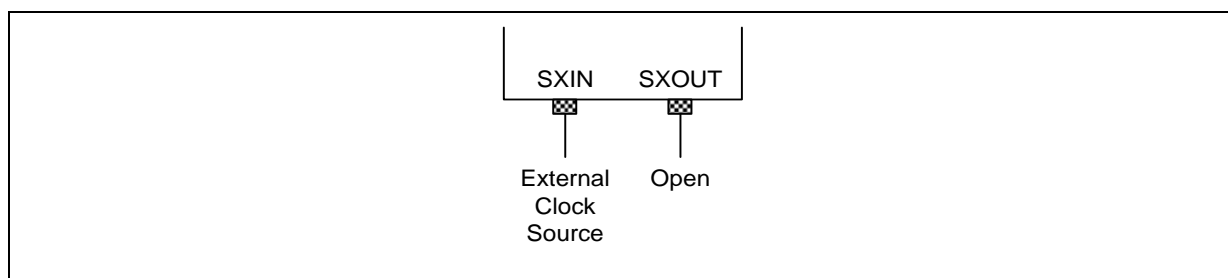


Figure 7.13 External Clock

## 7.18 Main Oscillation Stabilization Characteristics

Table 7-18 Main Oscillation Stabilization Characteristics

( $T_A = -40^\circ\text{C} \sim +85^\circ\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ )

| Oscillator     | Parameter                                                                                                               | MIN | TYP | MAX  | Unit |
|----------------|-------------------------------------------------------------------------------------------------------------------------|-----|-----|------|------|
| Crystal        | $f_x > 1\text{MHz}$<br>Oscillation stabilization occurs when $V_{DD}$ is equal to the minimum oscillator voltage range. | –   | –   | 60   | mS   |
| Ceramic        |                                                                                                                         | –   | –   | 10   | mS   |
| External Clock | $f_{XIN} = 0.4 \text{ to } 12\text{MHz}$<br>$XIN$ input high and low width ( $t_{XH}$ , $t_{XL}$ )                      | 42  | –   | 1250 | nS   |

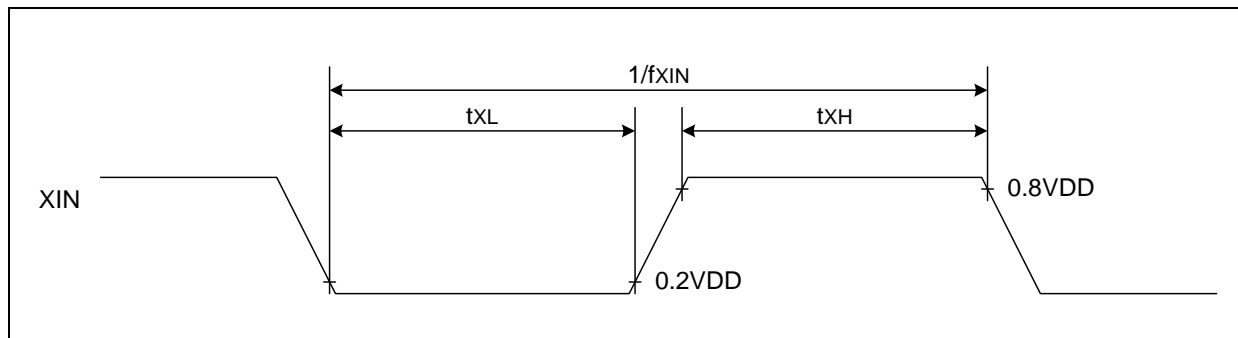


Figure 7.14 Clock Timing Measurement at XIN

## 7.19 Sub Oscillation Characteristics

Table 7-19 Sub Oscillation Stabilization Characteristics

( $T_A = -40^\circ\text{C} \sim +85^\circ\text{C}$ ,  $V_{DD} = 1.8\text{V} \sim 5.5\text{V}$ )

| Oscillator     | Parameter                                               | MIN | TYP | MAX | Unit          |
|----------------|---------------------------------------------------------|-----|-----|-----|---------------|
| Crystal        | –                                                       | –   | –   | 10  | S             |
| External Clock | $SXIN$ input high and low width ( $t_{XH}$ , $t_{XL}$ ) | 5   | –   | 15  | $\mu\text{S}$ |

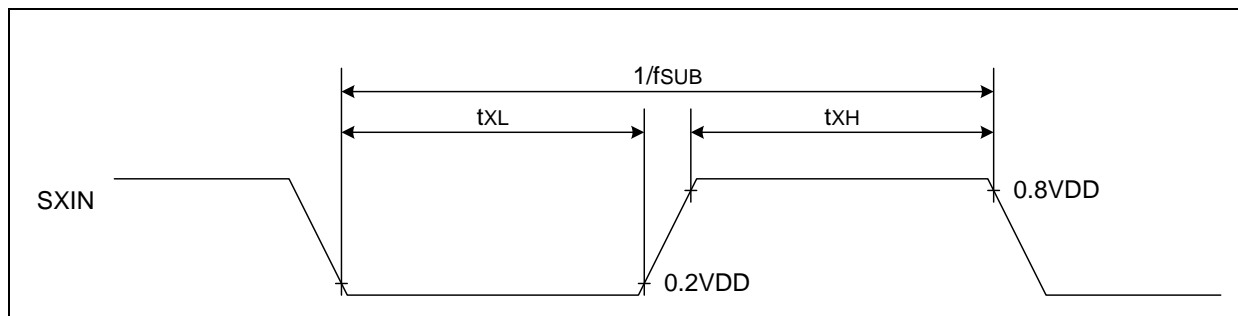


Figure 7.15 Clock Timing Measurement at SXIN

## 7.20 Operating Voltage Range

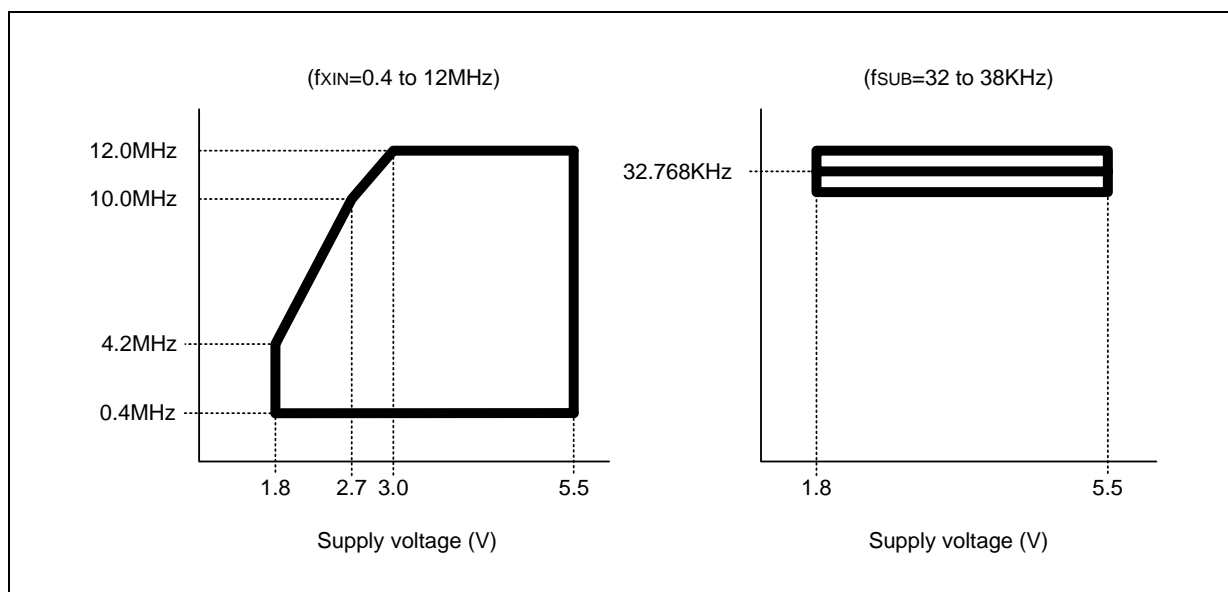


Figure 7.16 Operating Voltage Range

## 7.21 Recommended Circuit and Layout

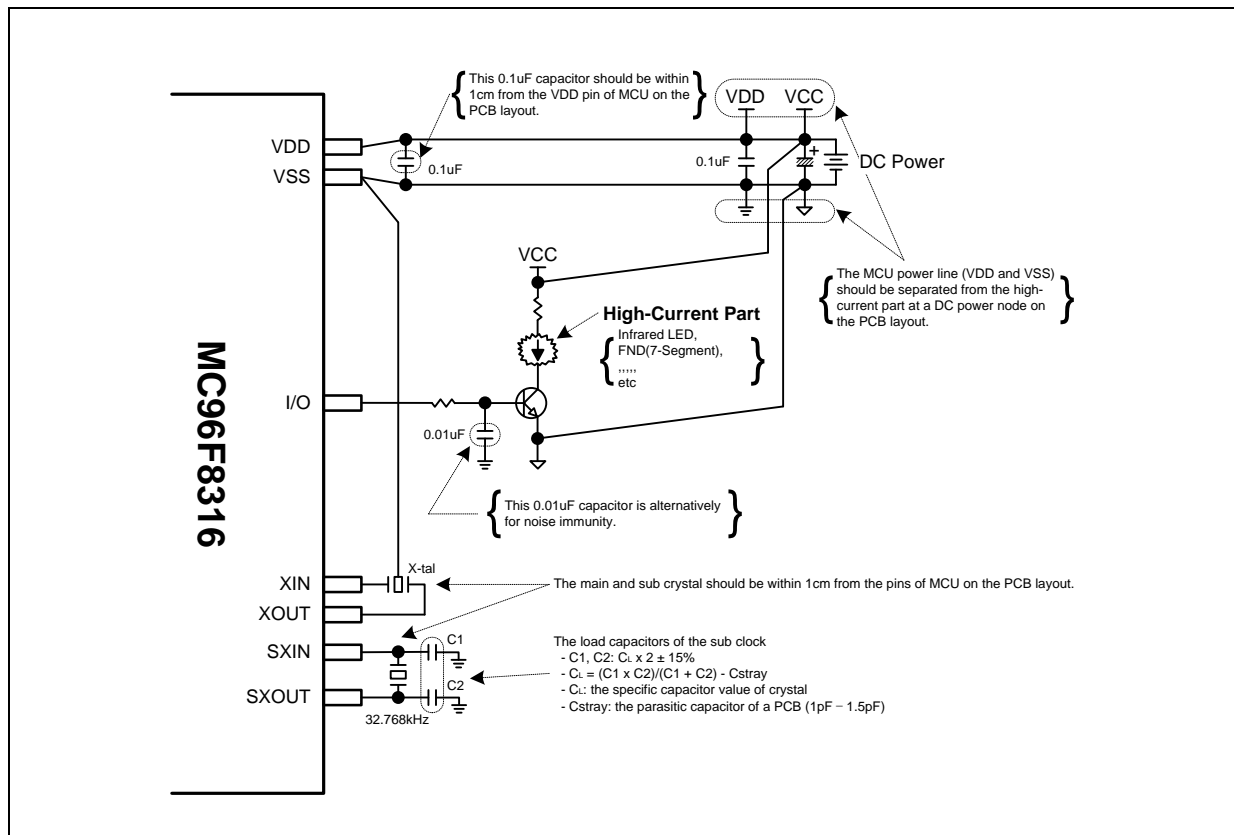


Figure 7.17 Recommended Circuit and Layout

## 7.22 Typical Characteristics

These graphs and tables provided in this section are only for design guidance and are not tested or guaranteed. In graphs or tables some data are out of specified operating range (e.g. out of specified VDD range). This is only for information and devices are guaranteed to operate properly only within the specified range.

The data presented in this section is a statistical summary of data collected on units from different lots over a period of time. "Typical" represents the mean of the distribution while "max" or "min" represents (mean +  $3\sigma$ ) and (mean -  $3\sigma$ ) respectively where  $\sigma$  is standard deviation.

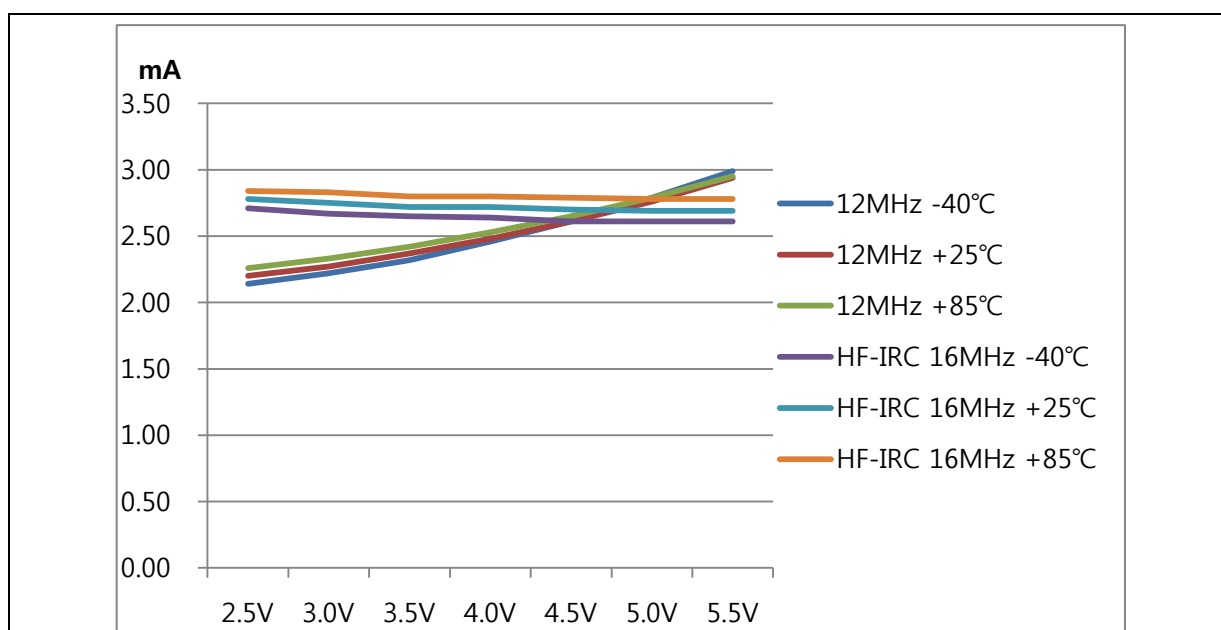


Figure 7.18 RUN (IDD1) Current

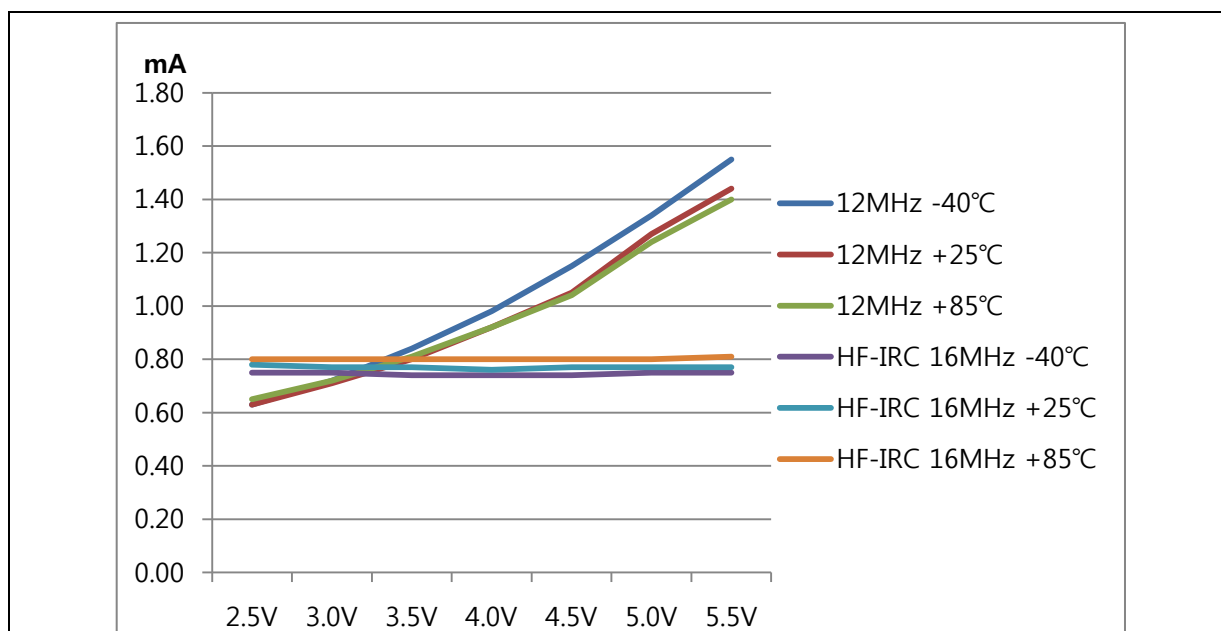


Figure 7.19 IDLE (IDD2) Current

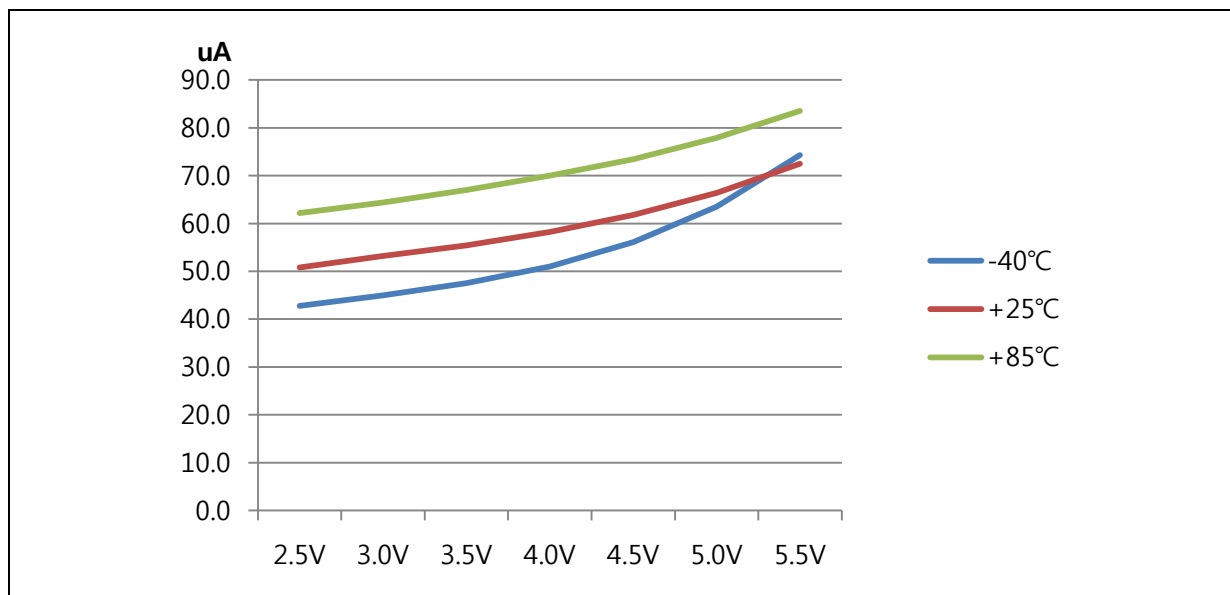


Figure 7.20 SUB RUN (IDD3) Current

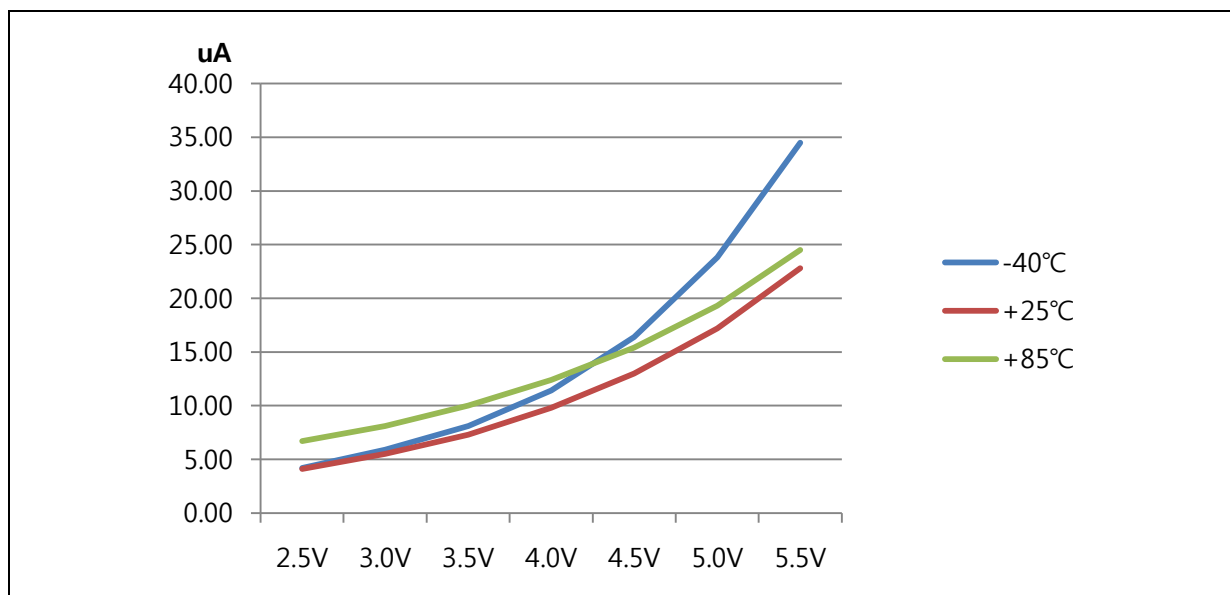


Figure 7.21 SUB IDLE (IDD4) Current



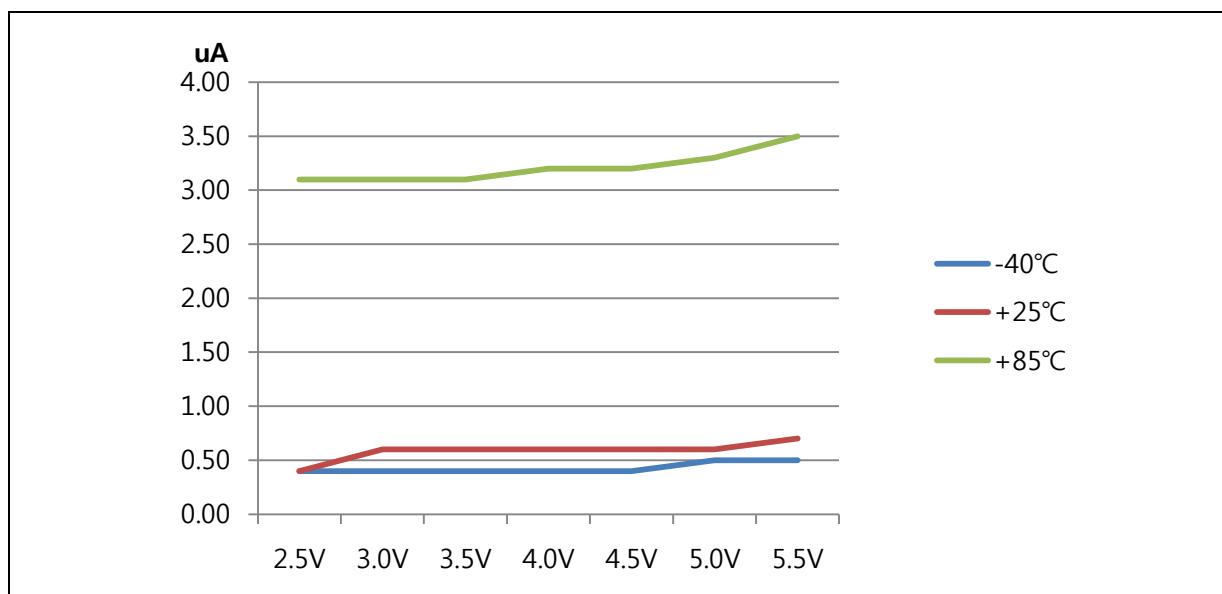


Figure 7.22 STOP (IDD5) Current

## 8. Memory

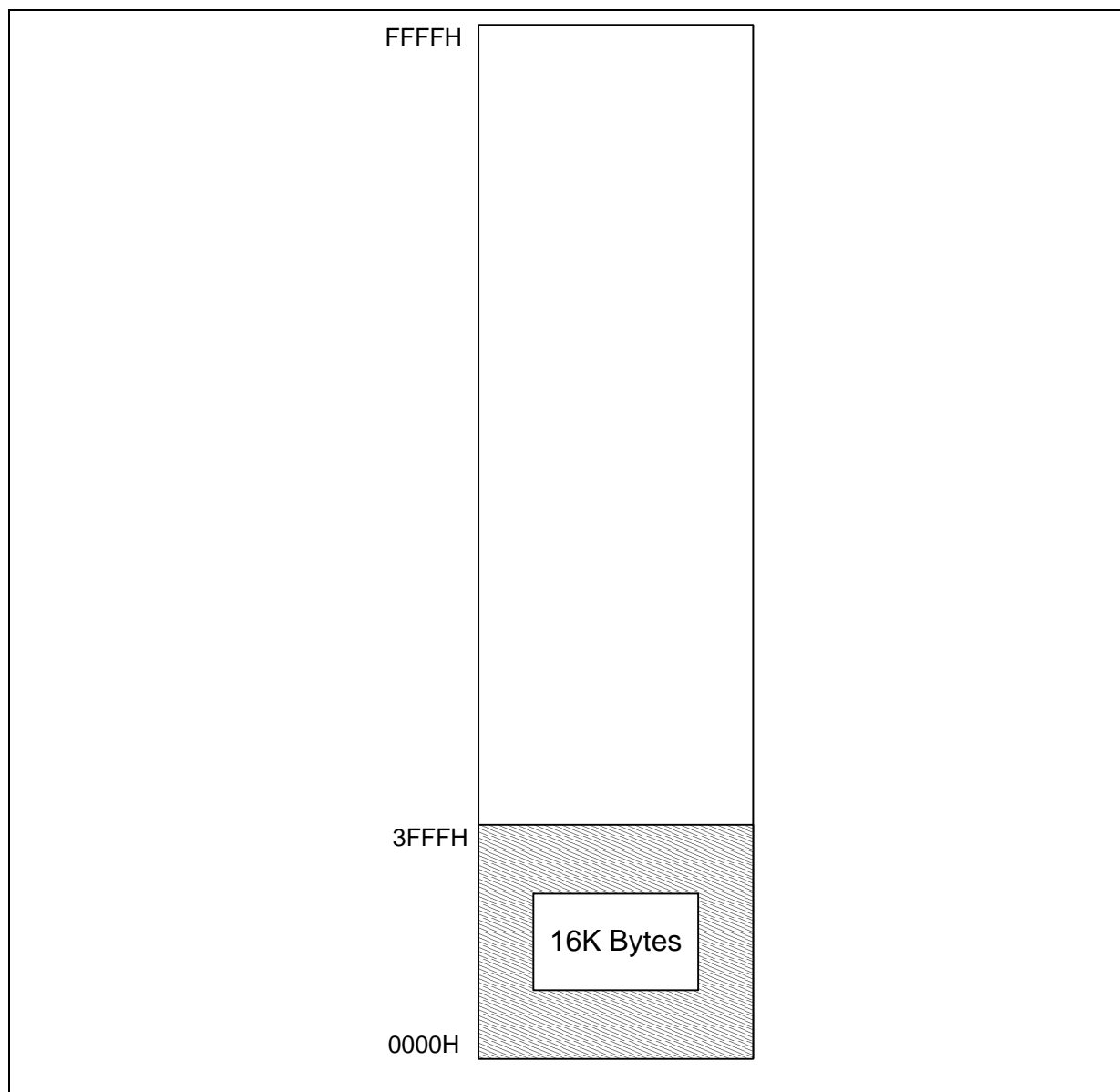
The MC96F8316 addresses two separate address memory stores: Program memory and Data memory. The logical separation of Program and Data memory allows Data memory to be accessed by 8-bit addresses, which makes the 8-bit CPU access the data memory more rapidly. Nevertheless, 16-bit Data memory addresses can also be generated through the DPTR register.

MC96F8316 provides on-chip 16k bytes of the ISP type flash program memory, which can be read and written to. Internal data memory (IRAM) is 256 bytes and it includes the stack area. External data memory (XRAM) is 512 bytes.

### 8.1 Program Memory

A 16-bit program counter is capable of addressing up to 64k bytes, but this device has just 16k bytes program memory space.

Figure 8-1 shows the map of the lower part of the program memory. After reset, the CPU begins execution from location 0000H. Each interrupt is assigned a fixed location in program memory. The interrupt causes the CPU to jump to that location, where it commences execution of the service routine. External interrupt 11, for example, is assigned to location 000BH. If external interrupt 11 is going to be used, its service routine must begin at location 000BH. If the interrupt is not going to be used, its service location is available as general purpose program memory. If an interrupt service routine is short enough (as is often the case in control applications), it can reside entirely within that 8 byte interval. Longer service routines can use a jump instruction to skip over subsequent interrupt locations, if other interrupts are in use.

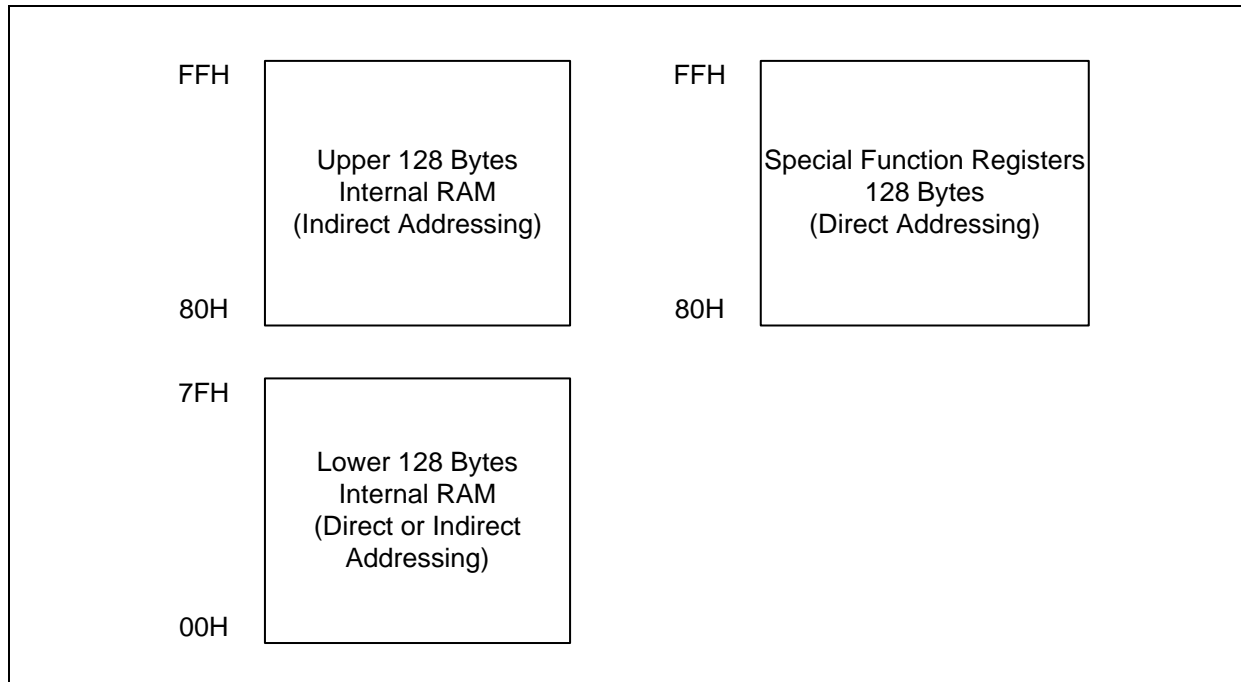


**Figure 8.1 Program Memory**

- 16k Bytes Including Interrupt Vector Region

## 8.2 Data Memory

Figure 8-2 shows the internal data memory space available.



**Figure 8.2 Data Memory Map**

The internal data memory space is divided into three blocks, which are generally referred to as the lower 128 bytes, upper 128 bytes, and SFR space.

Internal data memory addresses are always one byte wide, which implies an address space of only 256 bytes. However, in fact the addressing modes for internal RAM can accommodate up to 384 bytes by using a simple trick. Direct addresses higher than 7FH access one memory space and indirect addresses higher than 7FH access a different memory space. Thus Figure 8-2 shows the upper 128 bytes and SFR space occupying the same block of addresses, 80H through FFH, although they are physically separate entities.

The lower 128 bytes of RAM are present in all 8051 devices as mapped in Figure 8-3. The lowest 32 bytes are grouped into 4 banks of 8 registers. Program instructions call out these registers as R0 through R7. Two bits in the Program Status Word select which register bank is in use. This allows more efficient use of code space, since register instructions are shorter than instructions that use direct addressing.

The next 16 bytes above the register banks form a block of bit-addressable memory space. The 8051 instruction set includes a wide selection of single-bit instructions, and the 128 bits in this area can be directly addressed by these instructions. The bit addresses in this area are 00H through 7FH.

All of the bytes in the lower 128 bytes can be accessed by either direct or indirect addressing. The upper 128 bytes RAM can only be accessed by indirect addressing. These spaces are used for data RAM and stack.

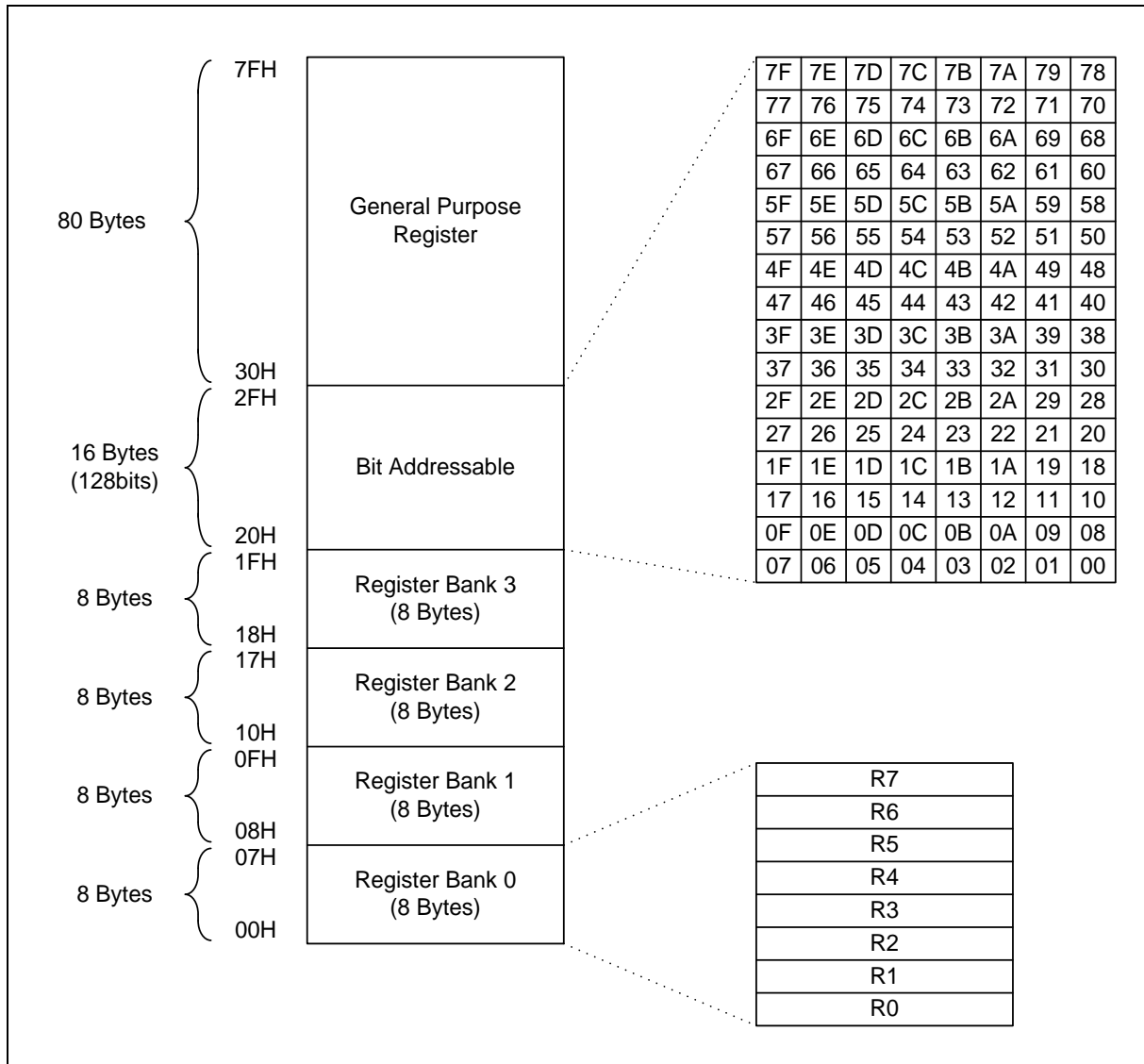


Figure 8.3 Lower 128 Bytes RAM

### 8.3 XRAM Memory

MC96F8316 has 512 bytes XRAM. This area has no relation with RAM/FLASH. It can be read and written to through SFR with 8-bit unit.

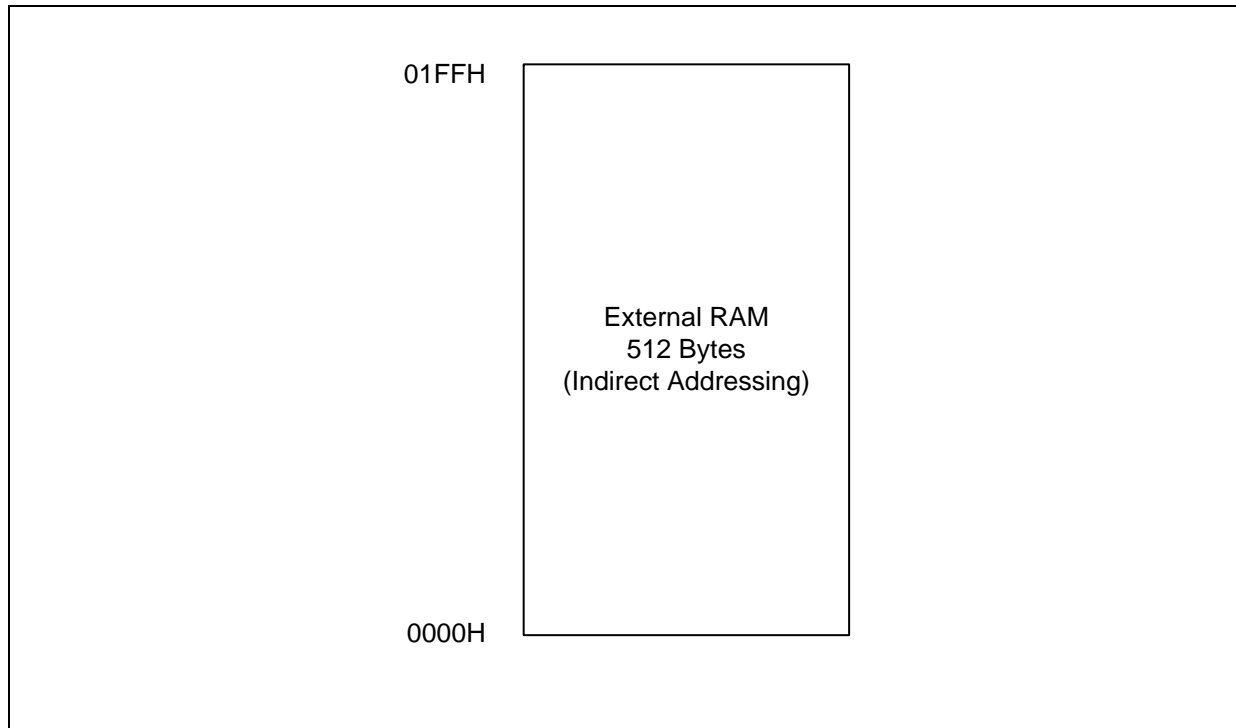


Figure 8.4 XDATA Memory Area

## 8.4 SFR Map

### 8.4.1 SFR Map Summary

Table 8-1 SFR Map Summary

|   |                  |
|---|------------------|
| — | Reserved         |
|   | M8051 compatible |

|      | 00H/8H <sup>(1)</sup> | 01H/9H         | 02H/0AH | 03H/0BH | 04H/0CH        | 05H/0DH | 06H/0EH          | 07H/0FH |
|------|-----------------------|----------------|---------|---------|----------------|---------|------------------|---------|
| 0F8H | IP1                   | —              | FSADRH  | FSADRM  | FSADRL         | FIDR    | FMCR             | —       |
| 0F0H | B                     | I2CSAR1        | ADWRCR0 | ADWRCR1 | ADWRCR2        | ADWRCR3 | ADWCRL           | ADWCRH  |
| 0E8H | RSTFR                 | I2CCR          | I2CSR   | I2CSAR0 | I2CDR          | I2CSDHR | I2CSCLR          | I2CSCHR |
| 0E0H | ACC                   | —              | UARTCR1 | UARTCR2 | UARTCR3        | UARTST  | UARTBD           | UARTDR  |
| 0D8H | LVRCR                 | —              | —       | —       | ADWIFRL        | ADWIFRH | P03DB            | P12DB   |
| 0D0H | PSW                   | —              | —       | P0FSR   | P1FSRL         | P1FSRH  | P2FSR            | P3FSR   |
| 0C8H | OSCCR                 | —              | —       | —       | —              | —       | —                | —       |
| 0C0H | EIFLAG0               | P3IO           | T2CRL   | T2CRH   | T2ADRL         | T2ADRH  | T2BDRL           | T2BDRH  |
| 0B8H | IP                    | P2IO           | T1CRL   | T1CRH   | T1ADRL         | T1ADRH  | T1BDRL           | T1BDRH  |
| 0B0H | EIFLAG1               | P1IO           | T0CR    | T0CNT   | T0DR/<br>T0CDR | SPICR   | SPIDR            | SPISR   |
| 0A8H | IE                    | IE1            | IE2     | IE3     | P0PU           | P1PU    | P2PU             | P3PU    |
| 0A0H | IIFLAG                | P0IO           | EO      | —       | EIPOL0L        | EIPOL0H | EIPOL1           | EIPOL2  |
| 98H  | P3                    | —              | —       | —       | ADCCR1         | ADCCR2  | ADCDRL           | ADCDRH  |
| 90H  | P2                    | P0OD           | P1OD    | P2OD    | P3OD           | —       | WTCR             | BUZCR   |
| 88H  | P1                    | WTDR/<br>WTCNT | SCCR    | BITCR   | BITCNT         | WDTCR   | WDTDR/<br>WDTCNT | BUZDR   |
| 80H  | P0                    | SP             | DPL     | DPH     | DPL1           | DPH1    | LVICR            | PCON    |

NOTE) These registers are bit-addressable.

## 8.4.2 SFR Map

Table 8-2 SFR Map

| Address | Function                               | Symbol | R/W | @Reset |   |   |   |   |   |   |   |
|---------|----------------------------------------|--------|-----|--------|---|---|---|---|---|---|---|
|         |                                        |        |     | 7      | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 80H     | P0 Data Register                       | P0     | R/W | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 81H     | Stack Pointer                          | SP     | R/W | 0      | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 82H     | Data Pointer Register Low              | DPL    | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 83H     | Data Pointer Register High             | DPH    | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 84H     | Data Pointer Register Low 1            | DPL1   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 85H     | Data Pointer Register High 1           | DPH1   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 86H     | Low Voltage Indicator Control Register | LVICR  | R/W | –      | – | 0 | 0 | 0 | 0 | 0 | 0 |
| 87H     | Power Control Register                 | PCON   | R/W | 0      | – | – | – | 0 | 0 | 0 | 0 |
| 88H     | P1 Data Register                       | P1     | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 89H     | Watch Timer Data Register              | WTDR   | W   | 0      | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
|         | Watch Timer Counter Register           | WTCNT  | R   | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8AH     | System and Clock Control Register      | SCCR   | R/W | –      | – | – | – | – | – | 0 | 0 |
| 8BH     | Basic Interval Timer Control Register  | BITCR  | R/W | 0      | 0 | 0 | – | 0 | 0 | 0 | 1 |
| 8CH     | Basic Interval Timer Counter Register  | BITCNT | R   | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8DH     | Watch Dog Timer Control Register       | WDTCR  | R/W | 0      | 0 | 0 | – | – | – | 0 | 0 |
| 8EH     | Watch Dog Timer Data Register          | WDTDR  | W   | 1      | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
|         | Watch Dog Timer Counter Register       | WDTCNT | R   | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8FH     | BUZZER Data Register                   | BUZDR  | R/W | 1      | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 90H     | P2 Data Register                       | P2     | R/W | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 91H     | P0 Open-drain Selection Register       | P0OD   | R/W | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 92H     | P1 Open-drain Selection Register       | P1OD   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 93H     | P2 Open-drain Selection Register       | P2OD   | R/W | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 94H     | P3 Open-drain Selection Register       | P3OD   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 95H     | Reserved                               | –      | –   | –      |   |   |   |   |   |   |   |
| 96H     | Watch Timer Control Register           | WTCR   | R/W | 0      | – | – | 0 | 0 | 0 | 0 | 0 |
| 97H     | BUZZER Control Register                | BUZCR  | R/W | –      | – | – | – | 0 | 0 | 0 | 0 |
| 98H     | P3 Data Register                       | P3     | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 99H     | Reserved                               | –      | –   | –      |   |   |   |   |   |   |   |
| 9AH     | Reserved                               | –      | –   | –      |   |   |   |   |   |   |   |
| 9BH     | Reserved                               | –      | –   | –      |   |   |   |   |   |   |   |
| 9CH     | A/D Converter Control Low Register     | ADCCRL | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9DH     | A/D Converter Control High Register    | ADCCRH | R/W | 0      | – | – | 0 | 0 | 0 | 0 | 0 |
| 9EH     | A/D Converter Data Low Register        | ADCDRH | R   | x      | x | x | x | x | x | x | x |
| 9FH     | A/D Converter Data High Register       | ADCDRH | R   | x      | x | x | x | x | x | x | x |



Table 8-2 SFR Map (Continued)

| Address | Function                                    | Symbol  | R/W | @ Reset |   |   |   |   |   |   |   |
|---------|---------------------------------------------|---------|-----|---------|---|---|---|---|---|---|---|
|         |                                             |         |     | 7       | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| A0H     | Internal Interrupt Flag Register            | IIFLAG  | R/W | –       | – | – | – | – | 0 | 0 | 0 |
| A1H     | P0 Direction Register                       | P0IO    | R/W | –       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| A2H     | Extended Operation Register                 | EO      | R/W | –       | – | – | 0 | – | 0 | 0 | 0 |
| A3H     | Reserved                                    | –       | –   | –       |   |   |   |   |   |   |   |
| A4H     | External Interrupt Polarity 0 Low Register  | EIPOL0L | R/W | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| A5H     | External Interrupt Polarity 0 High Register | EIPOL0H | R/W | –       | – | 0 | 0 | 0 | 0 | 0 | 0 |
| A6H     | External Interrupt Polarity 1 Register      | EIPOL1  | R/W | –       | – | 0 | 0 | 0 | 0 | 0 | 0 |
| A7H     | External Interrupt Polarity 2 Register      | EIPOL2  | R/W | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| A8H     | Interrupt Enable Register                   | IE      | R/W | 0       | – | 0 | – | – | 0 | 0 | 0 |
| A9H     | Interrupt Enable Register 1                 | IE1     | R/W | –       | – | 0 | 0 | 0 | 0 | 0 | 0 |
| AAH     | Interrupt Enable Register 2                 | IE2     | R/W | –       | – | 0 | – | 0 | 0 | 0 | 0 |
| ABH     | Interrupt Enable Register 3                 | IE3     | R/W | –       | – | – | 0 | 0 | 0 | 0 | 0 |
| ACH     | P0 Pull-up Resistor Selection Register      | P0PU    | R/W | –       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ADH     | P1 Pull-up Resistor Selection Register      | P1PU    | R/W | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| AEH     | P2 Pull-up Resistor Selection Register      | P2PU    | R/W | –       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| AFH     | P3 Pull-up Resistor Selection Register      | P3PU    | R/W | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B0H     | External Interrupt Flag 1 Register          | EIFLAG1 | R/W | –       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B1H     | P1 Direction Register                       | P1IO    | R/W | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B2H     | Timer 0 Control Register                    | T0CR    | R/W | 0       | – | 0 | 0 | 0 | 0 | 0 | 0 |
| B3H     | Timer 0 Counter Register                    | T0CNT   | R   | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B4H     | Timer 0 Data Register                       | T0DR    | R/W | 1       | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
|         | Timer 0 Capture Data Register               | T0CDR   | R   | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B5H     | SPI Control Register                        | SPICR   | R/W | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B6H     | SPI Data Register                           | SPIDR   | R/W | 0       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B7H     | SPI Status Register                         | SPISR   | R/W | 0       | 0 | 0 | – | 0 | 0 | – | – |
| B8H     | Interrupt Priority Register                 | IP      | R/W | –       | – | 0 | 0 | 0 | 0 | 0 | 0 |
| B9H     | P2 Direction Register                       | P2IO    | R/W | –       | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| BAH     | Timer 1 Control Low Register                | T1CRL   | R/W | 0       | 0 | 0 | 0 | – | 0 | 0 | 0 |
| BBH     | Timer 1 Control High Register               | T1CRH   | R/W | 0       | – | 0 | 0 | – | – | – | 0 |
| BCH     | Timer 1 A Data Low Register                 | T1ADRL  | R/W | 1       | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| BDH     | Timer 1 A Data High Register                | T1ADRH  | R/W | 1       | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| BEH     | Timer 1 B Data Low Register                 | T1BDRL  | R/W | 1       | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| BFH     | Timer 1 B Data High Register                | T1BDRH  | R/W | 1       | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Table 8-2 SFR Map (Continued)

| Address | Function                                 | Symbol  | R/W | @Reset |   |   |   |   |   |   |   |
|---------|------------------------------------------|---------|-----|--------|---|---|---|---|---|---|---|
|         |                                          |         |     | 7      | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| C0H     | External Interrupt Flag 0 Register       | EIFLAG0 | R/W | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| C1H     | P3 Direction Register                    | P3IO    | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| C2H     | Timer 2 Control Low Register             | T2CRL   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| C3H     | Timer 2 Control High Register            | T2CRH   | R/W | 0      | – | 0 | 0 | – | – | – | 0 |
| C4H     | Timer 2 A Data Low Register              | T2ADRL  | R/W | 1      | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| C5H     | Timer 2 A Data High Register             | T2ADRH  | R/W | 1      | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| C6H     | Timer 2 B Data Low Register              | T2BDRL  | R/W | 1      | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| C7H     | Timer 2 B Data High Register             | T2BDRH  | R/W | 1      | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| C8H     | Oscillator Control Register              | OSCCR   | R/W | –      | – | 0 | 0 | 1 | 0 | 0 | 0 |
| C9H     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| CAH     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| CBH     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| CCH     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| CDH     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| CEH     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| CFH     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| D0H     | Program Status Word Register             | PSW     | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D1H     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| D2H     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| D3H     | P0 Function Selection Register           | P0FSR   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D4H     | P1 Function Selection Low Register       | P1FSRL  | R/W | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D5H     | P1 Function Selection High Register      | P1FSRH  | R/W | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D6H     | P2 Function Selection Register           | P2FSR   | R/W | –      | – | – | – | – | – | 0 | 0 |
| D7H     | P3 Function Selection Register           | P3FSR   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D8H     | Low Voltage Reset Control Register       | LVRCCR  | R/W | 0      | – | – | 0 | 0 | 0 | 0 | 0 |
| D9H     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| DAH     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| DBH     | Reserved                                 | –       | –   | –      |   |   |   |   |   |   |   |
| DCH     | ADC Wake-up Interrupt Flag Low Register  | ADWIFRL | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DDH     | ADC Wake-up Interrupt Flag High Register | ADWIFRH | R/W | –      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DEH     | P03 Debounce Enable Register             | P03DB   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DFH     | P12 Debounce Enable Register             | P12DB   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table 8-2 SFR Map (Continued)

| Address | Function                               | Symbol  | R/W | @Reset |   |   |   |   |   |   |   |
|---------|----------------------------------------|---------|-----|--------|---|---|---|---|---|---|---|
|         |                                        |         |     | 7      | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| E0H     | Accumulator A Register                 | ACC     | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| E1H     | Reserved                               | —       | —   | —      |   |   |   |   |   |   |   |
| E2H     | UART Control Register 1                | UARTCR1 | R/W | —      | — | 0 | 0 | 0 | 0 | 0 | — |
| E3H     | UART Control Register 2                | UARTCR2 | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| E4H     | UART Control Register 3                | UARTCR3 | R/W | —      | 0 | — | — | — | 0 | 0 | 0 |
| E5H     | UART Status Register                   | UARTST  | R/W | 1      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| E6H     | UART Baud Rate Generation Register     | UARTBD  | R/W | 1      | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| E7H     | UART Data Register                     | UARTDR  | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| E8H     | Reset Flag Register                    | RSTFR   | R/W | 1      | x | 0 | 0 | x | — | — | — |
| E9H     | I2C Control Register                   | I2CCR   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| EAH     | I2C Status Register                    | I2CSR   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| EBH     | I2C Slave Address 0 Register           | I2CSAR0 | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ECH     | I2C Data Register                      | I2CDR   | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| EDH     | I2C SDA Hold Time Register             | I2CSDHR | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| EEH     | I2C SCL Low Period Register            | I2CSCLR | R/W | 0      | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| EFH     | I2C SCL High Period Register           | I2CSCHR | R/W | 0      | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| F0H     | B Register                             | B       | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F1H     | I2C Slave Address 1 Register           | I2CSAR1 | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F2H     | ADC Wake-up Resistor Control Register0 | ADWRCR0 | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F3H     | ADC Wake-up Resistor Control Register1 | ADWRCR1 | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F4H     | ADC Wake-up Resistor Control Register2 | ADWRCR2 | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F5H     | ADC Wake-up Resistor Control Register3 | ADWRCR3 | R/W | —      | — | 0 | 0 | 0 | 0 | 0 | 0 |
| F6H     | ADC Wake-up Control Low Register       | ADWCRL  | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F7H     | ADC Wake-up Control High Register      | ADWCRH  | R/W | —      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F8H     | Interrupt Priority Register 1          | IP1     | R/W | —      | — | 0 | 0 | 0 | 0 | 0 | 0 |
| F9H     | Reserved                               | —       | —   | —      |   |   |   |   |   |   |   |
| FAH     | Flash Sector Address High Register     | FSADRH  | R/W | —      | — | — | — | 0 | 0 | 0 | 0 |
| FBH     | Flash Sector Address Middle Register   | FSADRM  | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| FCH     | Flash Sector Address Low Register      | FSADRL  | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| FDH     | Flash Identification Register          | FIDR    | R/W | 0      | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| FEH     | Flash Mode Control Register            | FMCR    | R/W | 0      | — | — | — | — | 0 | 0 | 0 |
| FFH     | Reserved                               | —       | —   | —      |   |   |   |   |   |   |   |

### 8.4.3 Compiler Compatible SFR

**ACC (Accumulator Register) : E0H**

| 7   | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|-----|----|----|----|----|----|----|----|
| ACC |    |    |    |    |    |    |    |
| RW  | RW | RW | RW | RW | RW | RW | RW |

Initial value : 00H

ACC                      Accumulator

**B (B Register) : F0H**

| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|----|----|----|----|----|----|----|----|
| B  |    |    |    |    |    |    |    |
| RW | RW | RW | RW | RW | RW | RW | RW |

Initial value : 00H

B                      B Register

**SP (Stack Pointer) : 81H**

| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|----|----|----|----|----|----|----|----|
| SP |    |    |    |    |    |    |    |
| RW | RW | RW | RW | RW | RW | RW | RW |

Initial value : 07H

SP                      Stack Pointer

**DPL (Data Pointer Register Low) : 82H**

| 7   | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|-----|----|----|----|----|----|----|----|
| DPL |    |    |    |    |    |    |    |
| RW  | RW | RW | RW | RW | RW | RW | RW |

Initial value : 00H

DPL                      Data Pointer Low

**DPH (Data Pointer Register High) : 83H**

| 7   | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|-----|----|----|----|----|----|----|----|
| DPH |    |    |    |    |    |    |    |
| RW  | RW | RW | RW | RW | RW | RW | RW |

Initial value : 00H

DPH                      Data Pointer High

**DPL1 (Data Pointer Register Low 1) : 84H**

| 7    | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|------|----|----|----|----|----|----|----|
| DPL1 |    |    |    |    |    |    |    |
| RW   | RW | RW | RW | RW | RW | RW | RW |

Initial value : 00H

DPL1      Data Pointer Low 1

**DPH1 (Data Pointer Register High 1) : 85H**

| 7    | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|------|----|----|----|----|----|----|----|
| DPH1 |    |    |    |    |    |    |    |
| RW   | RW | RW | RW | RW | RW | RW | RW |

Initial value : 00H

DPH1      Data Pointer High 1

**PSW (Program Status Word Register) : D0H**

| 7  | 6  | 5  | 4   | 3   | 2  | 1  | 0  |
|----|----|----|-----|-----|----|----|----|
| CY | AC | F0 | RS1 | RS0 | OV | F1 | P  |
| RW | RW | RW | RW  | RW  | RW | RW | RW |

Initial value : 00H

CY      Carry Flag  
 AC      Auxiliary Carry Flag  
 F0      General Purpose User-Definable Flag  
 RS1      Register Bank Select bit 1  
 RS0      Register Bank Select bit 0  
 OV      Overflow Flag  
 F1      User-Definable Flag  
 P      Parity Flag. Set/Cleared by hardware each instruction cycle to indicate an odd/even number of '1' bits in the accumulator

**EO (Extended Operation Register) : A2H**

| 7 | 6 | 5 | 4       | 3 | 2      | 1      | 0      |
|---|---|---|---------|---|--------|--------|--------|
| – | – | – | TRAP_EN | – | DPSEL2 | DPSEL1 | DPSEL0 |
| – | – | – | RW      | – | RW     | RW     | RW     |

Initial value : 00H

TRAP\_EN      Select the Instruction (**Keep always '0'**).  
                  0      Select MOVC @(DPTR++), A  
                  1      Select Software TRAP Instruction  
 DPSEL[2:0]      Select Banked Data Pointer Register  
                  DPSEL2   DPSEL1   DPSEL0   Description  
                  0      0      0      DPTR0  
                  0      0      1      DPTR1  
                  Reserved

## 9. I/O Ports

### 9.1 I/O Ports

The MC96F8316 has four groups of I/O ports (P0 ~ P3). Each port can be easily configured by software as I/O pin, internal pull up and open-drain pin to meet various system configurations and design requirements. Also P0, P1, P2, and P3 include function that can generate interrupt according to change of state of the pin.

### 9.2 Port Register

#### 9.2.1 Data Register (Px)

Data Register is a bidirectional I/O port. If ports are configured as output ports, data can be written to the corresponding bit of the Px. If ports are configured as input ports, the data can be read from the corresponding bit of the Px.

#### 9.2.2 Direction Register (PxIO)

Each I/O pin can be independently used as an input or an output through the PxIO register. Bits cleared in this register will make the corresponding pin of Px to input mode. Set bits of this register will make the pin to output mode. Almost bits are cleared by a system reset, but some bits are set by a system reset.

#### 9.2.3 Pull-up Resistor Selection Register (PxPU)

The on-chip pull-up resistor can be connected to I/O ports individually with a pull-up resistor selection register (PxPU). The pull-up register selection controls the pull-up resistor enable/disable of each port. When the corresponding bit is 1, the pull-up resistor of the pin is enabled. When 0, the pull-up resistor is disabled. All bits are cleared by a system reset.

#### 9.2.4 Open-drain Selection Register (PxOD)

There are internally open-drain selection registers (PxOD) for P0 ~ P3. The open-drain selection register controls the open-drain enable/disable of each port. Almost ports become push-pull by a system reset, but some ports become open-drain by a system reset.

#### 9.2.5 Debounce Enable Register (PxDB)

P0[6:2], P1[3:0], P2[3:0] and P35 support debounce function. Debounce clocks of each ports are  $f_x/1$ ,  $f_x/4$ , and  $f_x/4096$ .

#### 9.2.6 Port Function Selection Register (PxFSR)

These registers define alternative functions of ports. Please remember that these registers should be set properly for alternative port function. A reset clears the PxFSR register to '00H', which makes all pins to normal I/O ports.

## 9.2.7 Register Map

Table 9-1 Port Register Map

| Name   | Address | Dir | Default | Description                            |
|--------|---------|-----|---------|----------------------------------------|
| P0     | 80H     | R/W | 00H     | P0 Data Register                       |
| P0IO   | A1H     | R/W | 00H     | P0 Direction Register                  |
| P0OD   | 91H     | R/W | 00H     | P0 Open-drain Selection Register       |
| P0PU   | ACH     | R/W | 00H     | P0 Pull-up Resistor Selection Register |
| P03DB  | DEH     | R/W | 00H     | P0/P3 Debounce Enable Register         |
| P0FSR  | D3H     | R/W | 00H     | P0 Function Selection Register         |
| P1     | 88H     | R/W | 00H     | P1 Data Register                       |
| P1IO   | B1H     | R/W | 00H     | P1 Direction Register                  |
| P1OD   | 92H     | R/W | 00H     | P1 Open-drain Selection Register       |
| P1PU   | ADH     | R/W | 00H     | P1 Pull-up Resistor Selection Register |
| P12DB  | DFH     | R/W | 00H     | P1/P2 Debounce Enable Register         |
| P1FSRH | D5H     | R/W | 00H     | P1 Function Selection High Register    |
| P1FSRL | D4H     | R/W | 00H     | P1 Function Selection Low Register     |
| P2     | 90H     | R/W | 00H     | P2 Data Register                       |
| P2IO   | B9H     | R/W | 00H     | P2 Direction Register                  |
| P2OD   | 93H     | R/W | 00H     | P2 Open-drain Selection Register       |
| P2PU   | AEH     | R/W | 00H     | P2 Pull-up Resistor Selection Register |
| P2FSR  | D6H     | R/W | 00H     | P2 Function Selection Register         |
| P3     | 98H     | R/W | 00H     | P3 Data Register                       |
| P3IO   | C1H     | R/W | 00H     | P3 Direction Register                  |
| P3OD   | 94H     | R/W | 00H     | P3 Open-drain Selection Register       |
| P3PU   | AFH     | R/W | 00H     | P3 Pull-up Resistor Selection Register |
| P3FSR  | D7H     | R/W | 00H     | P3 Function Selection Register         |

## 9.3 P0 Port

### 9.3.1 P0 Port Description

P0 is 7-bit I/O port. P0 control registers consist of P0 data register (P0), P0 direction register (P0IO), debounce enable register (P03DB), P0 pull-up resistor selection register (P0PU), and P0 open-drain selection register (P0OD). Refer to the port function selection registers for the P0 function selection.

### 9.3.2 Register description for P0

#### P0 (P0 Data Register) : 80H

| 7 | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
|---|-----|-----|-----|-----|-----|-----|-----|
| – | P06 | P05 | P04 | P03 | P02 | P01 | P00 |
| – | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Initial value : 00H

P0[6:0] I/O Data

#### P0IO (P0 Direction Register) : A1H

| 7 | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|---|-------|-------|-------|-------|-------|-------|-------|
| – | P06IO | P05IO | P04IO | P03IO | P02IO | P01IO | P00IO |
| – | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P0IO[6:0] P0 Data I/O Direction.  
 0 Input  
 1 Output

NOTE: EINT0 ~ EINT4 function possible when input

#### P0PU (P0 Pull-up Resistor Selection Register) : ACH

| 7 | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|---|-------|-------|-------|-------|-------|-------|-------|
| – | P06PU | P05PU | P04PU | P03PU | P02PU | P01PU | P00PU |
| – | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P0PU[6:0] Configure Pull-up Resistor of P0 Port  
 0 Disable  
 1 Enable

#### P0OD (P0 Open-drain Selection Register) : 91H

| 7 | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|---|-------|-------|-------|-------|-------|-------|-------|
| – | P06OD | P05OD | P04OD | P03OD | P02OD | P01OD | P00OD |
| – | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P0OD[6:0] Configure Open-drain of P0 Port  
 0 Push-pull output  
 1 Open-drain output



**P03DB (P0/P3 Debounce Enable Register) : DEH**

| 7      | 6      | 5     | 4     | 3     | 2     | 1     | 0     |
|--------|--------|-------|-------|-------|-------|-------|-------|
| DBCLK1 | DBCLK0 | P35DB | P06DB | P05DB | P04DB | P03DB | P02DB |
| RW     | RW     | RW    | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

DBCLK[1:0] Configure Debounce Clock of Port

DBCLK1 DBCLK0 Description

0 0 fx/1

0 1 fx/4

1 0 fx/4096

1 1 Reserved

P35DB Configure Debounce of P35 Port

0 Disable

1 Enable

P06DB Configure Debounce of P06 Port

0 Disable

1 Enable

P05DB Configure Debounce of P05 Port

0 Disable

1 Enable

P04DB Configure Debounce of P04 Port

0 Disable

1 Enable

P03DB Configure Debounce of P03Port

0 Disable

1 Enable

P02DB Configure Debounce of P02 Port

0 Disable

1 Enable

NOTES) 1. If the same level is not detected on enabled pin three or four times in a row at the sampling clock, the signal is eliminated as noise.

2. A pulse level should be input for the duration of 3 clock or more to be actually detected as a valid edge.

3. The port debounce is automatically disabled at stop mode and recovered after stop mode release.

**P0FSR (Port 0 Function Selection Register) : D3H**

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| P0FSR7 | P0FSR6 | P0FSR5 | P0FSR4 | P0FSR3 | P0FSR2 | P0FSR1 | P0FSR0 |
| R/W    | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    |

Initial value : 00H

|            |                     |                                                    |
|------------|---------------------|----------------------------------------------------|
| P0FSR7     | P06 Function select |                                                    |
|            | 0                   | I/O Port (EINT4 function possible when input)      |
|            | 1                   | AN6 Function                                       |
| P0FSR6     | P05 Function select |                                                    |
|            | 0                   | I/O Port (EINT3 function possible when input)      |
|            | 1                   | AN5 Function                                       |
| P0FSR5     | P04 Function select |                                                    |
|            | 0                   | I/O Port (EINT2 function possible when input)      |
|            | 1                   | AN4 Function                                       |
| P0FSR4     | P03 Function select |                                                    |
|            | 0                   | I/O Port (EINT1 function possible when input)      |
|            | 1                   | AN3 Function                                       |
| P0FSR[3:2] | P02 Function Select |                                                    |
|            | P0FSR3              | P0FSR2    Description                              |
|            | 0                   | 0    I/O Port (EINT0 function possible when input) |
|            | 0                   | 1    AVREF Function                                |
|            | 1                   | 0    AN2 Function                                  |
|            | 1                   | 1    Not used                                      |
| P0FSR1     | P01 Function select |                                                    |
|            | 0                   | I/O Port                                           |
|            | 1                   | AN1 Function                                       |
| P0FSR0     | P00 Function select |                                                    |
|            | 0                   | I/O Port                                           |
|            | 1                   | AN0 Function                                       |

## 9.4 P1 Port

### 9.4.1 P1 Port Description

P1 is 8-bit I/O port. P1 control registers consist of P1 data register (P1), P1 direction register (P1IO), debounce enable register (P12DB), P1 pull-up resistor selection register (P1PU), and P1 open-drain selection register (P1OD) . Refer to the port function selection registers for the P1 function selection.

### 9.4.2 Register description for P1

#### P1 (P1 Data Register) : 88H

| 7   | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
|-----|-----|-----|-----|-----|-----|-----|-----|
| P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Initial value : 00H

P1[7:0] I/O Data

#### P1IO (P1 Direction Register) : B1H

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| P17IO | P16IO | P15IO | P14IO | P13IO | P12IO | P11IO | P10IO |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P1IO[7:0] P1 Data I/O Direction  
 0 Input  
 1 Output

NOTE: EINT5/ENINT6/EINT11/EINT12/EC1/EC2/SS function possible when input

#### P1PU (P1 Pull-up Resistor Selection Register) : ADH

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| P17PU | P16PU | P15PU | P14PU | P13PU | P12PU | P11PU | P10PU |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P1PU[7:0] Configure Pull-up Resistor of P1 Port  
 0 Disable  
 1 Enable

#### P1OD (P1 Open-drain Selection Register) : 92H

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| P17OD | P16OD | P15OD | P14OD | P13OD | P12OD | P11OD | P10OD |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P1OD[7:0] Configure Open-drain of P1 Port  
 0 Push-pull output  
 1 Open-drain output

**P12DB (P1/P2 Debounce Enable Register) : DFH**

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| P23DB | P22DB | P21DB | P20DB | P13DB | P12DB | P11DB | P10DB |
| RW    | RW    | RW    | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

|       |                                |
|-------|--------------------------------|
| P23DB | Configure Debounce of P23 Port |
| 0     | Disable                        |
| 1     | Enable                         |
| P22DB | Configure Debounce of P22 Port |
| 0     | Disable                        |
| 1     | Enable                         |
| P21DB | Configure Debounce of P21 Port |
| 0     | Disable                        |
| 1     | Enable                         |
| P20DB | Configure Debounce of P20 Port |
| 0     | Disable                        |
| 1     | Enable                         |
| P13DB | Configure Debounce of P13 Port |
| 0     | Disable                        |
| 1     | Enable                         |
| P12DB | Configure Debounce of P12 Port |
| 0     | Disable                        |
| 1     | Enable                         |
| P11DB | Configure Debounce of P11 Port |
| 0     | Disable                        |
| 1     | Enable                         |
| P10DB | Configure Debounce of P10 Port |
| 0     | Disable                        |
| 1     | Enable                         |

NOTES) 1. If the same level is not detected on enabled pin three or four times in a row at the sampling clock, the signal is eliminated as noise.

2. A pulse level should be input for the duration of 3 clock or more to be actually detected as a valid edge.

3. The port debounce is automatically disabled at stop mode and recovered after stop mode release.

4. Refer to the port 0/3 debounce enable register (P03DB) for the debounce clock of port 1 and port 2.

**P1FSRH (Port 1 Function Selection High Register) : D5H**

| 7 | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---|---------|---------|---------|---------|---------|---------|---------|
| – | P1FSRH6 | P1FSRH5 | P1FSRH4 | P1FSRH3 | P1FSRH2 | P1FSRH1 | P1FSRH0 |
| – | RW      | RW      | RW      | RW      | RW      | RW      | RW      |

Initial value : 00H

|             |                                                |               |
|-------------|------------------------------------------------|---------------|
| P1FSRH6     | P17 Function Select                            |               |
| 0           | I/O Port (EC2/SS function possible when input) |               |
| 1           | AN14 Function                                  |               |
| P1FSRH[5:4] | P16 Function Select                            |               |
| P1FSRH5     | P1FSRH4                                        | Description   |
| 0           | 0                                              | I/O Port      |
| 0           | 1                                              | SCK Function  |
| 1           | 0                                              | AN13 Function |
| 1           | 1                                              | Not used      |
| P1FSRH[3:2] | P15 Function Select                            |               |
| P1FSRH3     | P1FSRH2                                        | Description   |
| 0           | 0                                              | I/O Port      |
| 0           | 1                                              | MOSI Function |
| 1           | 0                                              | AN12 Function |
| 1           | 1                                              | Not used      |
| P1FSRH[1:0] | P14 Function Select                            |               |
| P1FSRH1     | P0FSRH0                                        | Description   |
| 0           | 0                                              | I/O Port      |
| 0           | 1                                              | MISO Function |
| 1           | 0                                              | AN11 Function |
| 1           | 1                                              | Not used      |

**P1FSRL (Port 1 Function Selection Low Register) : D4H**

| 7 | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---|---------|---------|---------|---------|---------|---------|---------|
| – | P1FSRL6 | P1FSRL5 | P1FSRL4 | P1FSRL3 | P1FSRL2 | P1FSRL1 | P1FSRL0 |
| – | RW      | RW      | RW      | RW      | RW      | RW      | RW      |

Initial value : 00H

|             |                     |         |                                                   |
|-------------|---------------------|---------|---------------------------------------------------|
| P1FSRL[6:5] | P13 Function Select |         |                                                   |
|             | P1FSRL6             | P1FSRL5 | Description                                       |
|             | 0                   | 0       | I/O Port (EINT12 function possible when input)    |
|             | 0                   | 1       | T2O/PWM2O Function                                |
|             | 1                   | 0       | AN10 Function                                     |
|             | 1                   | 1       | Not used                                          |
| P1FSRL[4:3] | P12 Function Select |         |                                                   |
|             | P1FSRL4             | P1FSRL3 | Description                                       |
|             | 0                   | 0       | I/O Port (EINT11 function possible when input)    |
|             | 0                   | 1       | T1O/PWM1O Function                                |
|             | 1                   | 0       | AN9 Function                                      |
|             | 1                   | 1       | Not used                                          |
| P1FSRL[2:1] | P11 Function Select |         |                                                   |
|             | P1FSRL2             | P1FSRL1 | Description                                       |
|             | 0                   | 0       | I/O Port (EINT6/EC1 function possible when input) |
|             | 0                   | 1       | BUZO Function                                     |
|             | 1                   | 0       | AN8 Function                                      |
|             | 1                   | 1       | Not used                                          |
| P1FSRL0     | P10 Function Select |         |                                                   |
|             | 0                   |         | I/O Port (EINT5 function possible when input)     |
|             | 1                   |         | AN7 Function                                      |

## 9.5 P2 Port

### 9.5.1 P2 Port Description

P2 is 7-bit I/O port. P2 control registers consist of P2 data register (P2), P2 direction register (P2IO), P2 pull-up resistor selection register (P2PU) and P2 open-drain selection register (P2OD). Refer to the port function selection registers for the P2 function selection.

### 9.5.2 Register description for P2

#### P2 (P2 Data Register) : 90H

| 7 | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
|---|-----|-----|-----|-----|-----|-----|-----|
| – | P26 | P25 | P24 | P23 | P22 | P21 | P20 |
| – | RW  | RW  | RW  | RW  | RW  | RW  | RW  |

Initial value : 00H

P2[6:0]      I/O Data

#### P2IO (P2 Direction Register) : B9H

| 7 | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|---|-------|-------|-------|-------|-------|-------|-------|
| – | P26IO | P25IO | P24IO | P23IO | P22IO | P21IO | P20IO |
| – | RW    | RW    | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

P2IO[6:0]      P2 Data I/O Direction

0      Input

1      Output

NOTE: EINT7 – EINTA, EC0 function possible when input

#### P2PU (P2 Pull-up Resistor Selection Register) : AEH

| 7 | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|---|-------|-------|-------|-------|-------|-------|-------|
| – | P26PU | P25PU | P24PU | P23PU | P22PU | P21PU | P20PU |
| – | RW    | RW    | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

P2PU[6:0]      Configure Pull-up Resistor of P2 Port

0      Disable

1      Enable

#### P2OD (P2 Open-drain Selection Register) : 93H

| 7 | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|---|-------|-------|-------|-------|-------|-------|-------|
| – | P26OD | P25OD | P24OD | P23OD | P22OD | P21OD | P20OD |
| – | RW    | RW    | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

P2OD[6:0]      Configure Open-drain of P2 Port

0      Push-pull output

1      Open-drain output

**P2FSR (Port 2 Function Selection Register) : D6H**

| 7 | 6 | 5 | 4 | 3 | 2 | 1      | 0      |
|---|---|---|---|---|---|--------|--------|
| – | – | – | – | – | – | P2FSR1 | P2FSR0 |
| – | – | – | – | – | – | RW     | RW     |

Initial value : 00H

|        |                     |
|--------|---------------------|
| P2FSR1 | P25 Function select |
| 0      | I/O Port            |
| 1      | SCL Function        |
| P2FSR0 | P24 Function Select |
| 0      | I/O Port            |
| 1      | SDA Function        |



## 9.6 P3 Port

### 9.6.1 P3 Port Description

P3 is 8-bit I/O port. P3 control registers consist of P3 data register (P3), P3 direction register (P3IO), P3 pull-up resistor selection register (P3PU) and P3 open-drain selection register (P3OD). Refer to the port function selection registers for the P3 function selection.

### 9.6.2 Register description for P3

#### P3 (P3 Data Register) : 98H

| 7   | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
|-----|-----|-----|-----|-----|-----|-----|-----|
| P37 | P36 | P35 | P34 | P33 | P32 | P31 | P30 |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Initial value : 00H

P3[7:0] I/O Data

#### P3IO (P3 Direction Register) : C1H

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| P37IO | P36IO | P35IO | P34IO | P33IO | P32IO | P31IO | P30IO |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P3IO[7:0] P3 Data I/O Direction  
0 Input  
1 Output

NOTE: RXD, EINT10 function possible when input

#### P3PU (P3 Pull-up Resistor Selection Register) : AFH

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| P37PU | P36PU | P35PU | P34PU | P33PU | P32PU | P31PU | P30PU |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P3PU[7:0] Configure Pull-up Resistor of P3 Port  
0 Disable  
1 Enable

#### P3OD (P3 Open-drain Selection Register) : 94H

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| P37OD | P36OD | P35OD | P34OD | P33OD | P32OD | P31OD | P30OD |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

P3OD[7:0] Configure Open-drain of P3 Port  
0 Push-pull output  
1 Open-drain output

**P3FSR (Port 3 Function Selection Register) : D7H**

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| P3FSR7 | P3FSR6 | P3FSR5 | P3FSR4 | P3FSR3 | P3FSR2 | P3FSR1 | P3FSR0 |
| RW     | RW     | RW     | RW     | RW     | RW     | RW     | RW     |

Initial value : 00H

|            |                     |                                             |              |
|------------|---------------------|---------------------------------------------|--------------|
| P3FSR7     | P37 Function select |                                             |              |
|            | 0                   | I/O Port                                    |              |
|            | 1                   | XOUT Function                               |              |
| P3FSR6     | P36 Function Select |                                             |              |
|            | 0                   | I/O Port                                    |              |
|            | 1                   | XIN Function                                |              |
| P3FSR5     | P35 Function select |                                             |              |
|            | 0                   | I/O Port                                    |              |
|            | 1                   | T0O/PWM0O Function                          |              |
| P3FSR4     | P34 Function Select |                                             |              |
|            | 0                   | I/O Port                                    |              |
|            | 1                   | SXIN Function                               |              |
| P3FSR3     | P33 Function select |                                             |              |
|            | 0                   | I/O Port                                    |              |
|            | 1                   | SXOUT Function                              |              |
| P3FSR2     | P31 Function Select |                                             |              |
|            | 0                   | I/O Port (RXD function possible when input) |              |
|            | 1                   | SCL Function                                |              |
| P3FSR[1:0] | P30 Function select |                                             |              |
|            | P3FSR1              | P3FSR0                                      | Description  |
|            | 0                   | 0                                           | I/O Port     |
|            | 0                   | 1                                           | TXD Function |
|            | 1                   | 0                                           | SDA Function |
|            | 1                   | 1                                           | Not used     |

NOTE) Refer to the configure option for the P32/RESETB

## 10. Interrupt Controller

### 10.1 Overview

The MC96F8316 supports up to 20 interrupt sources. The interrupts have separate enable register bits associated with them, allowing software control. They can also have four levels of priority assigned to them. The non-maskable interrupt source is always enabled with a higher priority than any other interrupt source, and is not controllable by software. The interrupt controller has following features:

- Receive the request from 20 interrupt source
- 6 group priority
- 4 priority levels
- Multi Interrupt possibility
- If the requests of different priority levels are received simultaneously, the request of higher priority level is served first.
- Each interrupt source can be controlled by EA bit and each IEx bit
- Interrupt latency: 3~9 machine cycles in single interrupt system

The non-maskable interrupt is always enabled. The maskable interrupts are enabled through four pair of interrupt enable registers (IE, IE1, IE2, IE3). Each bit of IE, IE1, IE2, IE3 register individually enables/disables the corresponding interrupt source. Overall control is provided by bit 7 of IE (EA). When EA is set to '0', all interrupts are disabled: when EA is set to '1', interrupts are individually enabled or disabled through the other bits of the interrupt enable registers. The EA bit is always cleared to '0' jumping to an interrupt service vector and set to '1' executing the [RETI] instruction. The MC96F8316 supports a four-level priority scheme. Each maskable interrupt is individually assigned to one of four priority levels according to IP and IP1.

Default interrupt mode is level-trigger mode basically, but if needed, it is possible to change to edge-trigger mode. Table 10-1 shows the Interrupt Group Priority Level that is available for sharing interrupt priority. Priority of a group is set by two bits of interrupt priority registers (one bit from IP, another one from IP1). Interrupt service routine serves higher priority interrupt first. If two requests of different priority levels are received simultaneously, the request of higher priority level is served prior to the lower one.

**Table 10-1 Interrupt Group Priority Level**

| Interrupt Group | Highest <span style="float: right;">Lowest</span> |              |              |              |                                       |
|-----------------|---------------------------------------------------|--------------|--------------|--------------|---------------------------------------|
|                 |                                                   |              |              |              |                                       |
| 0 (Bit0)        | Interrupt 0                                       | Interrupt 6  | Interrupt 12 | Interrupt 18 | Highest<br><br><br><br><br><br>Lowest |
| 1 (Bit1)        | Interrupt 1                                       | Interrupt 7  | Interrupt 13 | Interrupt 19 |                                       |
| 2 (Bit2)        | Interrupt 2                                       | Interrupt 8  | Interrupt 14 | Interrupt 20 |                                       |
| 3 (Bit3)        | Interrupt 3                                       | Interrupt 9  | Interrupt 15 | Interrupt 21 |                                       |
| 4 (Bit4)        | Interrupt 4                                       | Interrupt 10 | Interrupt 16 | Interrupt 22 |                                       |
| 5 (Bit5)        | Interrupt 5                                       | Interrupt 11 | Interrupt 17 | Interrupt 23 |                                       |

## 10.2 External Interrupt

The external interrupt on INT0, INT1, INT2, INT3, INT4, INT5, INT6, INT7, INT8, INT9, INTA, INT10, INT11 and INT12 pins receive various interrupt request depending on the external interrupt polarity 0 high/low register (EIPOL0H/L), external interrupt polarity 1 register (EIPOL1) and external interrupt polarity 2 register (EIPOL2) as shown in Figure 10.1. Also each external interrupt source has enable/disable bits. The external interrupt flag 0 register (EIFLAG0) and external interrupt flag 1 register (EIFLAG1) provides the status of external interrupts.

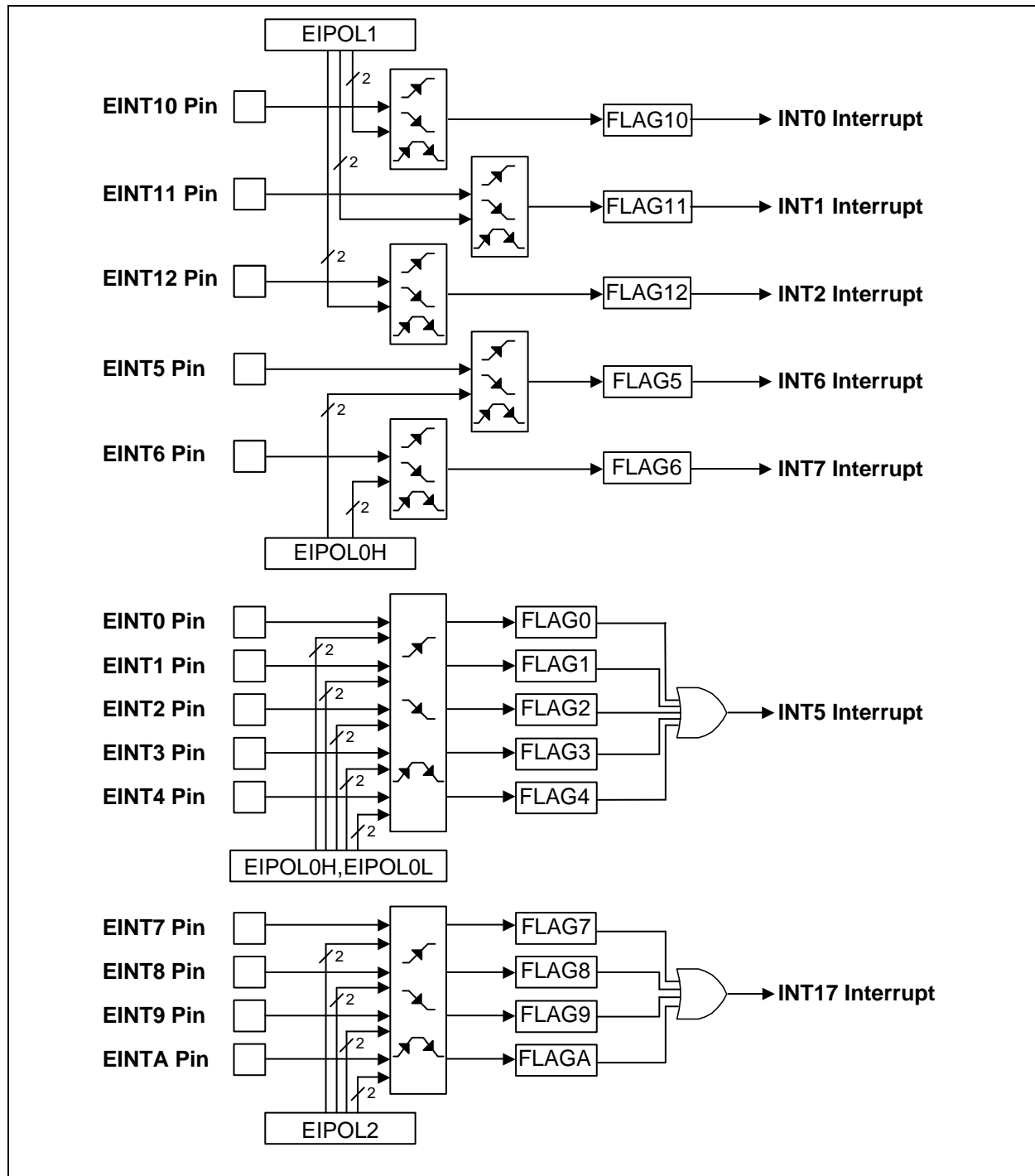


Figure 10.1 External Interrupt Description

### 10.3 Block Diagram

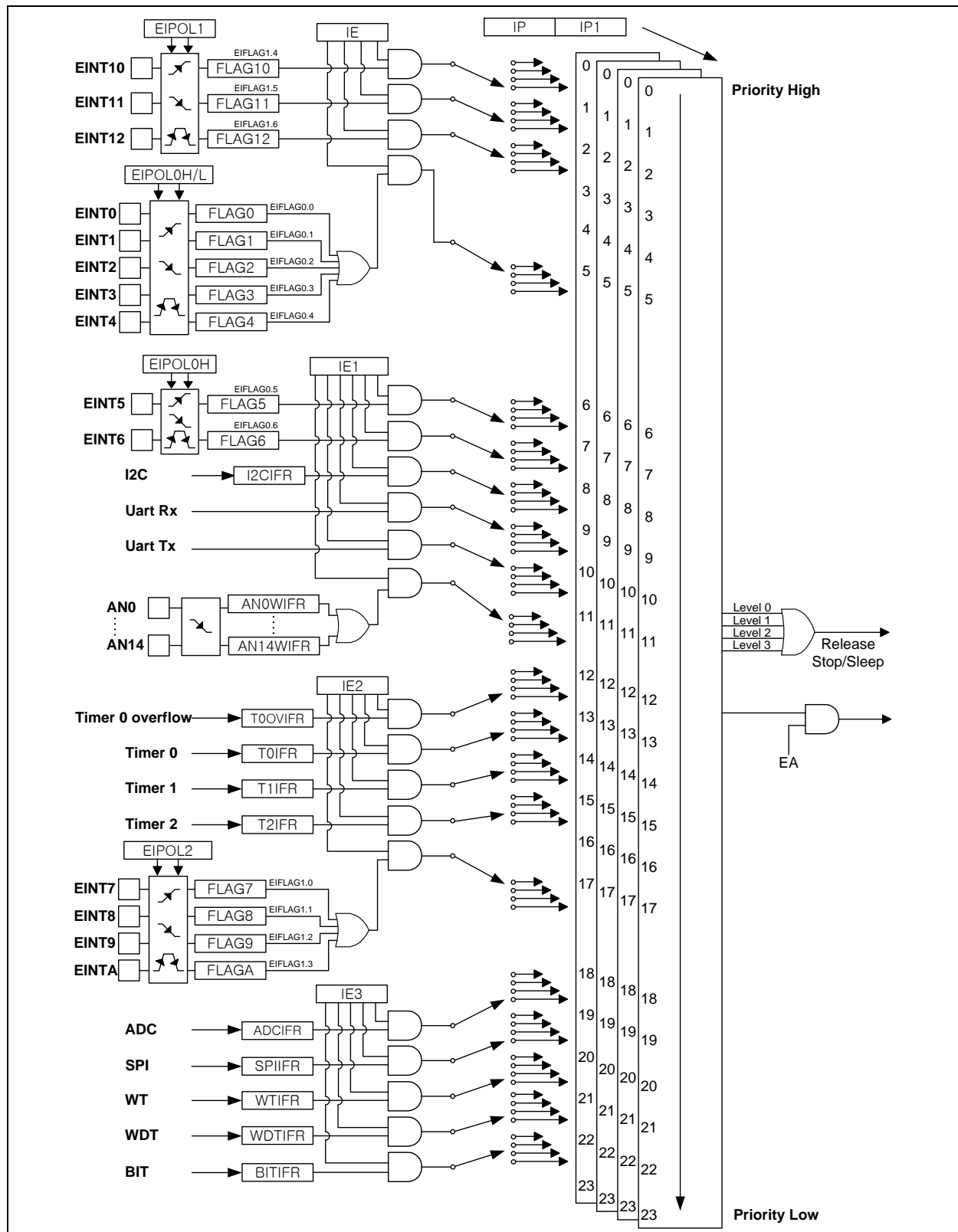


Figure 10.2 Block Diagram of Interrupt

- NOTES) 1. The release signal for stop/idle mode may be generated by all interrupt sources which are enabled without reference to the priority level.
2. An interrupt request is delayed while data are written to IE, IE1, IE2, IE3, IP, IP1, and PCON register.

## 10.4 Interrupt Vector Table

The interrupt controller supports 24 interrupt sources as shown in the Table 10-2. When interrupt is served, long call instruction (LCALL) is executed and program counter jumps to the vector address. All interrupt requests have their own priority order.

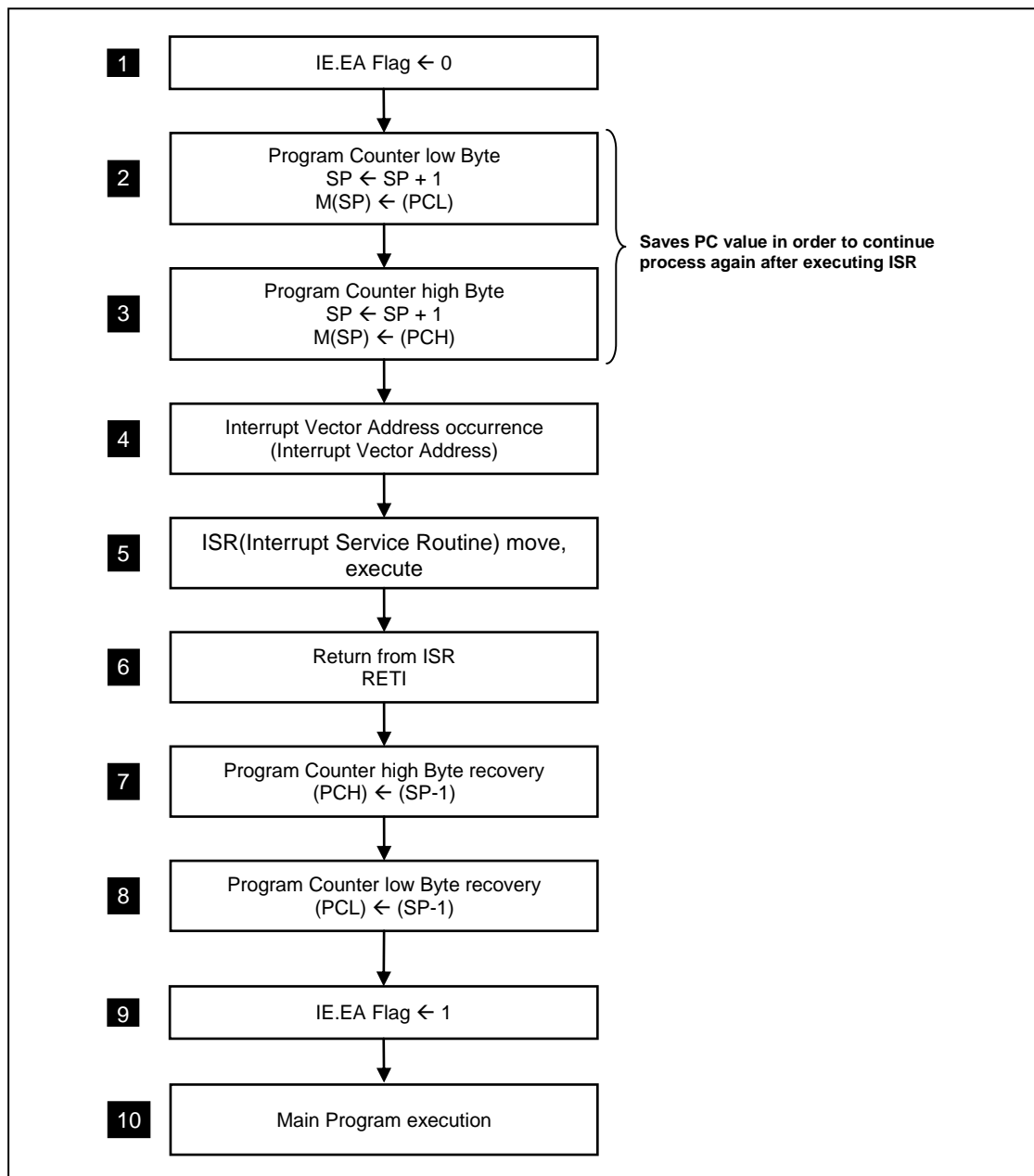
Table 10-2 Interrupt Vector Address Table

| Interrupt Source         | Symbol | Interrupt Enable Bit | Priority | Mask         | Vector Address |
|--------------------------|--------|----------------------|----------|--------------|----------------|
| Hardware Reset           | RESETB | 0 0                  | 0        | Non-Maskable | 0000H          |
| External Interrupt 10    | INT0   | IE.0                 | 1        | Maskable     | 0003H          |
| External Interrupt 11    | INT1   | IE.1                 | 2        | Maskable     | 000BH          |
| External Interrupt 12    | INT2   | IE.2                 | 3        | Maskable     | 0013H          |
| -                        | INT3   | IE.3                 | 4        | Maskable     | 001BH          |
| -                        | INT4   | IE.4                 | 5        | Maskable     | 0023H          |
| External Interrupt 0 – 4 | INT5   | IE.5                 | 6        | Maskable     | 002BH          |
| External Interrupt 5     | INT6   | IE1.0                | 7        | Maskable     | 0033H          |
| External Interrupt 6     | INT7   | IE1.1                | 8        | Maskable     | 003BH          |
| I2C Interrupt            | INT8   | IE1.2                | 9        | Maskable     | 0043H          |
| UART Rx Interrupt        | INT9   | IE1.3                | 10       | Maskable     | 004BH          |
| UART Tx Interrupt        | INT10  | IE1.4                | 11       | Maskable     | 0053H          |
| ADC Wake-up Interrupt    | INT11  | IE1.5                | 12       | Maskable     | 005BH          |
| T0 Overflow Interrupt    | INT12  | IE2.0                | 13       | Maskable     | 0063H          |
| T0 Match Interrupt       | INT13  | IE2.1                | 14       | Maskable     | 006BH          |
| T1 Match Interrupt       | INT14  | IE2.2                | 15       | Maskable     | 0073H          |
| T2 Match Interrupt       | INT15  | IE2.3                | 16       | Maskable     | 007BH          |
| -                        | INT16  | IE2.4                | 17       | Maskable     | 0083H          |
| External Interrupt 7 – A | INT17  | IE2.5                | 18       | Maskable     | 008BH          |
| ADC Interrupt            | INT18  | IE3.0                | 19       | Maskable     | 0093H          |
| SPI Interrupt            | INT19  | IE3.1                | 20       | Maskable     | 009BH          |
| WT Interrupt             | INT20  | IE3.2                | 21       | Maskable     | 00A3H          |
| WDT Interrupt            | INT21  | IE3.3                | 22       | Maskable     | 00ABH          |
| BIT Interrupt            | INT22  | IE3.4                | 23       | Maskable     | 00B3H          |
| -                        | INT23  | IE3.5                | 24       | Maskable     | 00BBH          |

For maskable interrupt execution, EA bit must set '1' and specific interrupt must be enabled by writing '1' to associated bit in the IEx. If an interrupt request is received, the specific interrupt request flag is set to '1'. And it remains '1' until CPU accepts interrupt. If the interrupt is served, the interrupt request flag will be cleared automatically.

## 10.5 Interrupt Sequence

An interrupt request is held until the interrupt is accepted or the interrupt latch is cleared to '0' by a reset or an instruction. Interrupt acceptance always generates at last cycle of the instruction. So instead of fetching the current instruction, CPU executes internally LCALL instruction and saves the PC at stack. For the interrupt service routine, the interrupt controller gives the address of LJMP instruction to CPU. Since the end of the execution of current instruction, it needs 3~9 machine cycles to go to the interrupt service routine. The interrupt service task is terminated by the interrupt return instruction [RETI]. Once an interrupt request is generated, the following process is performed.



**Figure 10.3 Interrupt Vector Address Table**

## 10.6 Effective Timing after Controlling Interrupt Bit

Case a) Control Interrupt Enable Register (IE, IE1, IE2, IE3)

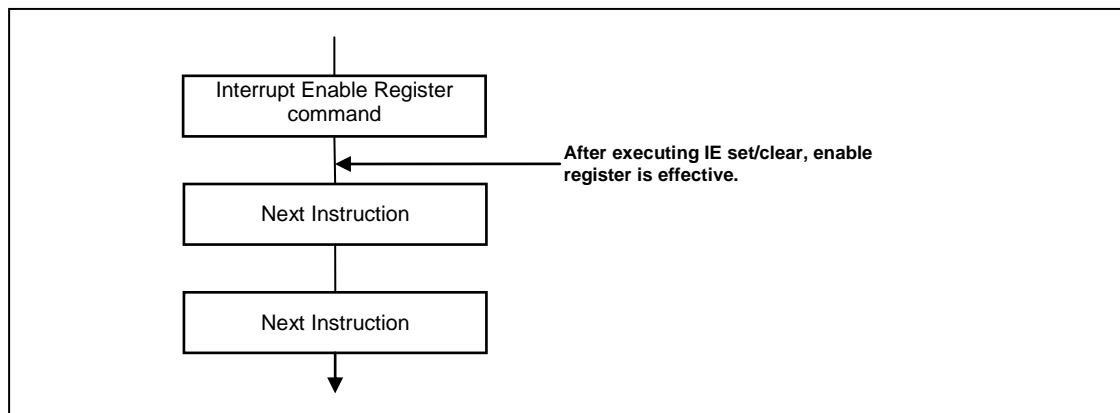


Figure 10.4 Effective Timing of Interrupt Enable Register

Case b) Interrupt flag Register

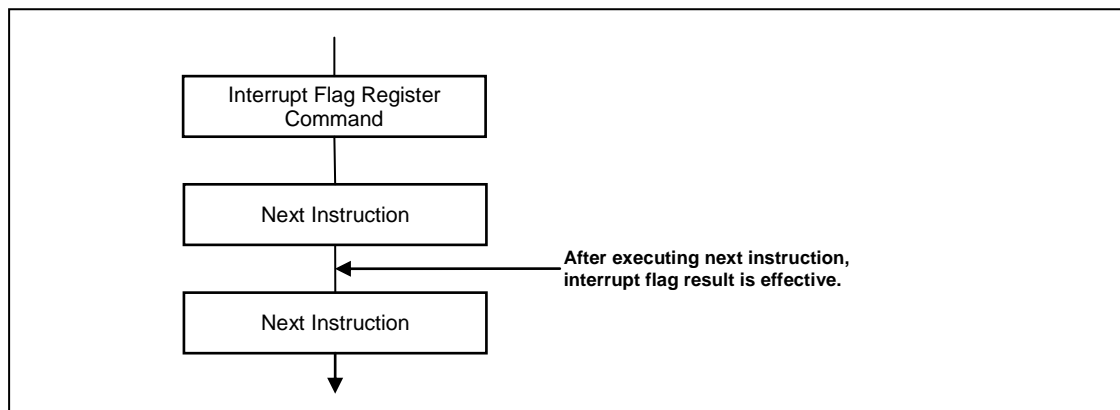
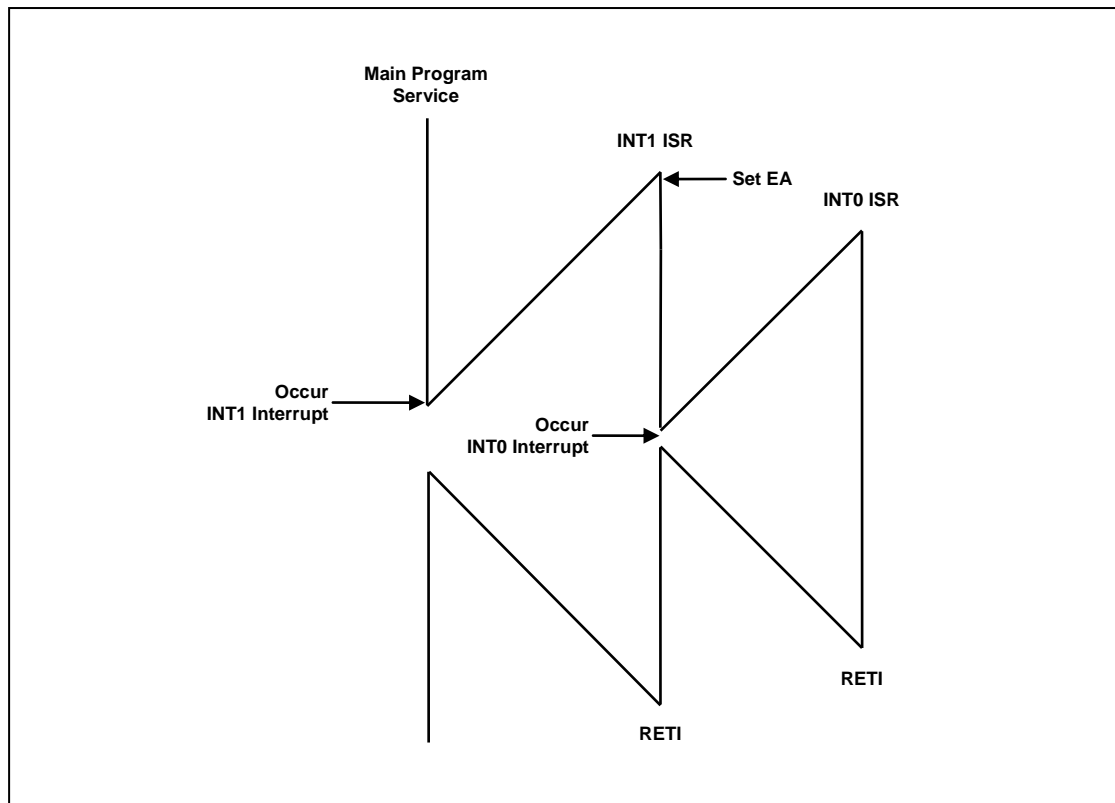


Figure 10.5 Effective Timing of Interrupt Flag Register



## 10.7 Multi Interrupt

If two requests of different priority levels are received simultaneously, the request of higher priority level is served first. If more than one interrupt request are received, the interrupt polling sequence determines which request is served first by hardware. However, for special features, multi-interrupt processing can be executed by software.



**Figure 10.6 Effective Timing of Interrupt**

Figure 10.6 shows an example of multi-interrupt processing. While INT1 is served, INT0 which has higher priority than INT1 is occurred. Then INT0 is served immediately and then the remain part of INT1 service routine is executed. If the priority level of INT0 is same or lower than INT1, INT0 will be served after the INT1 service has completed.

An interrupt service routine may be only interrupted by an interrupt of higher priority and, if two interrupts of different priority occur at the same time, the higher level interrupt will be served first. An interrupt cannot be interrupted by another interrupt of the same or a lower priority level. If two interrupts of the same priority level occur simultaneously, the service order for those interrupts is determined by the scan order.

## 10.8 Interrupt Enable Accept Timing

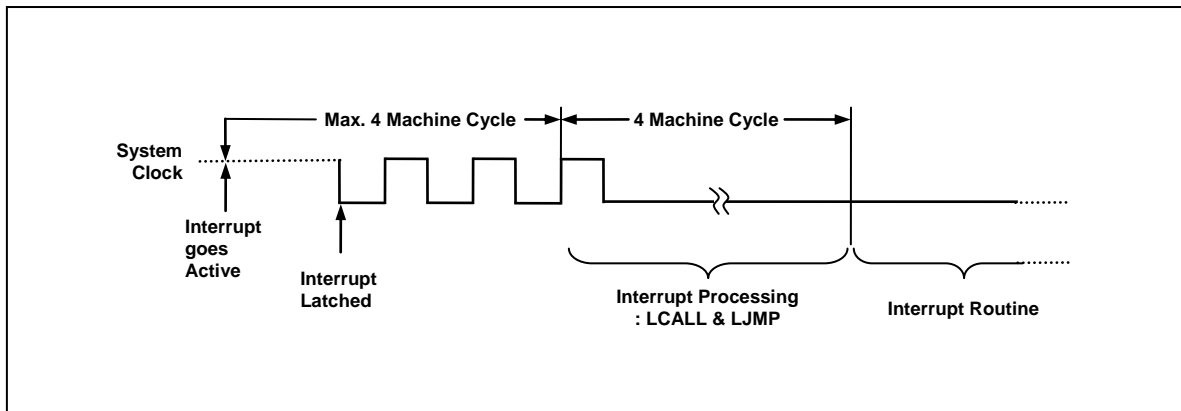


Figure 10.7 Interrupt Response Timing Diagram

## 10.9 Interrupt Service Routine Address

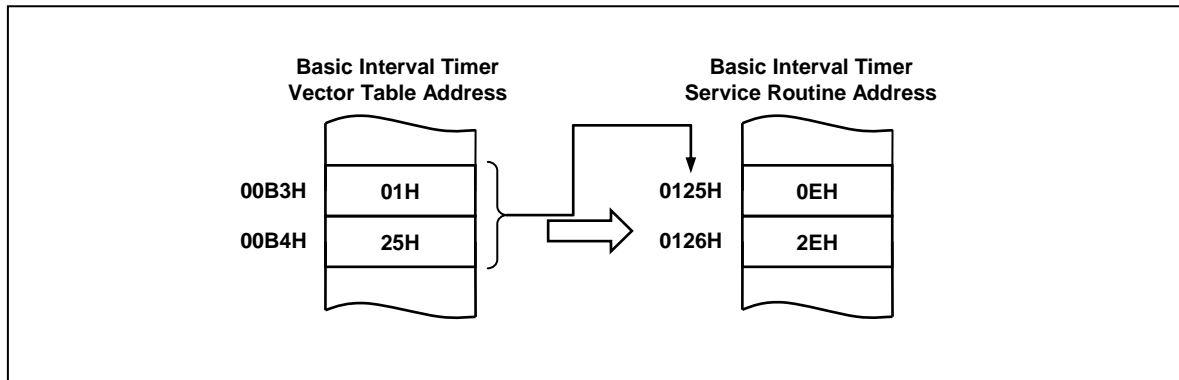


Figure 10.8 Correspondence between Vector Table Address and the Entry Address of ISP

## 10.10 Saving/Restore General-Purpose Registers

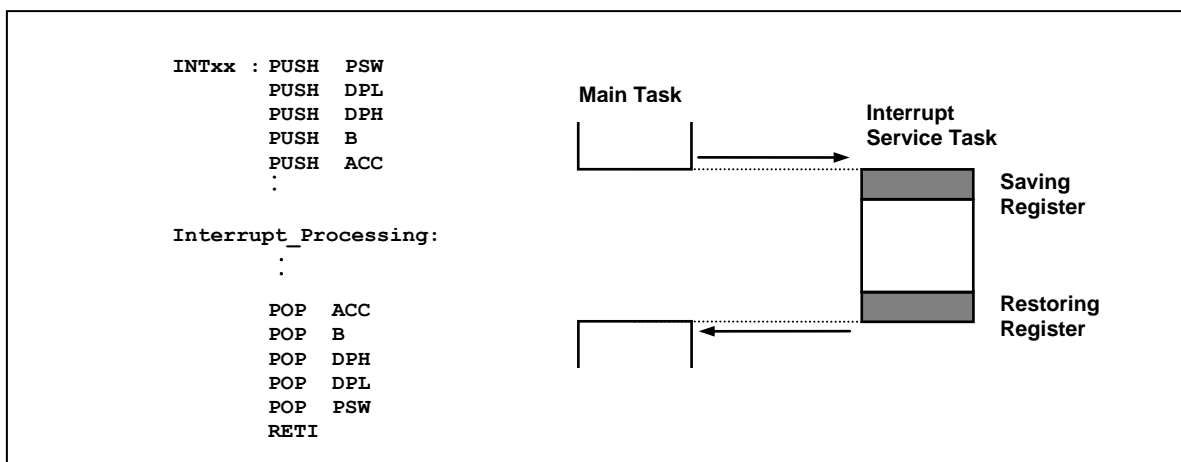
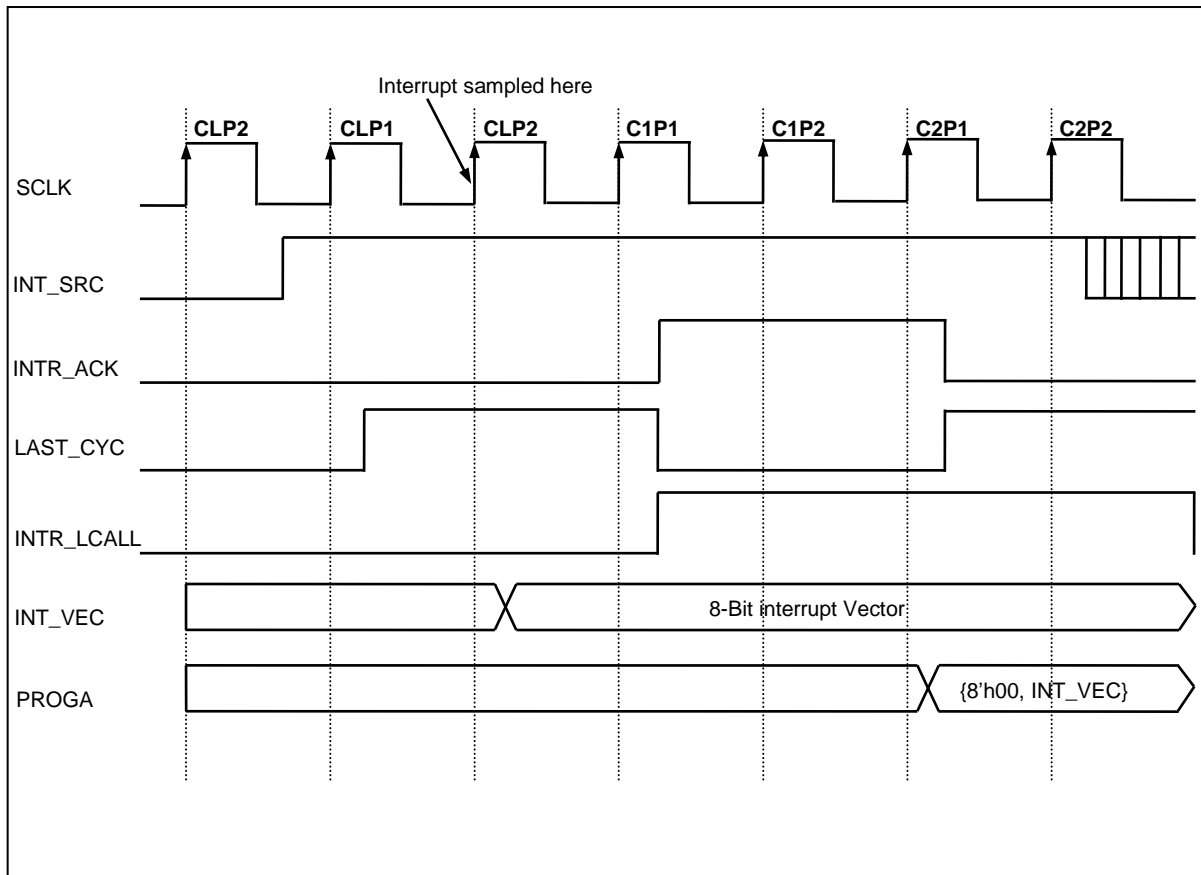


Figure 10.9 Saving/Restore Process Diagram and Sample Source

## 10.11 Interrupt Timing



**Figure 10.10 Timing Chart of Interrupt Acceptance and Interrupt Return Instruction**

Interrupt sources are sampled at the last cycle of a command. If an interrupt source is detected the lower 8-bit of interrupt vector (INT\_VEC) is decided. M8051W core makes interrupt acknowledge at the first cycle of a command, and executes long call to jump to interrupt service routine.

NOTE) command cycle CLPx: L=Last cycle, 1=1<sup>st</sup> cycle or 1<sup>st</sup> phase, 2=2<sup>nd</sup> cycle or 2<sup>nd</sup> phase

## 10.12 Interrupt Register Overview

### 10.12.1 Interrupt Enable Register (IE, IE1, IE2, IE3)

Interrupt enable register consists of global interrupt control bit (EA) and peripheral interrupt control bits. Total 24 peripherals are able to control interrupt.

### 10.12.2 Interrupt Priority Register (IP, IP1)

The 24 interrupts are divided into 6 groups which have each 4 interrupt sources. A group can be assigned 4 levels interrupt priority using interrupt priority register. Level 3 is the highest priority, while level 0 is the lowest priority. After a reset IP and IP1 are cleared to '00H'. If interrupts have the same priority level, lower number interrupt is served first.

### **10.12.3 External Interrupt Flag Register (EIFLAG0, EIFLAG1)**

The external interrupt flag 0 register (EIFLAG0) and external interrupt flag 1 register (EIFLAG1) are set to '1' when the external interrupt generating condition is satisfied. The flag is cleared when the interrupt service routine is executed. Alternatively, the flag can be cleared by writing '0' to it.

### **10.12.4 External Interrupt Polarity Register (EIPOL0H, EIPOL0L, EIPOL1, EIPOL2)**

The external interrupt polarity 0 high/low register (EIPOL0H/L), external interrupt polarity 1 register (EIPOL1) and external interrupt polarity 2 register (EIPOL2) determines which type of rising/falling/both edge interrupt. Initially, default value is no interrupt at any edge.

## 10.12.5 Register Map

**Table 10-3 Interrupt Register Map**

| Name    | Address | Dir | Default | Description                                 |
|---------|---------|-----|---------|---------------------------------------------|
| IE      | A8H     | R/W | 00H     | Interrupt Enable Register                   |
| IE1     | A9H     | R/W | 00H     | Interrupt Enable Register 1                 |
| IE2     | AAH     | R/W | 00H     | Interrupt Enable Register 2                 |
| IE3     | ABH     | R/W | 00H     | Interrupt Enable Register 3                 |
| IP      | B8H     | R/W | 00H     | Interrupt Priority Register                 |
| IP1     | F8H     | R/W | 00H     | Interrupt Priority Register 1               |
| IIFLAG  | A0H     | R/W | 00H     | Internal Interrupt Flag Register            |
| EIFLAG0 | C0H     | R/W | 00H     | External Interrupt Flag 0 Register          |
| EIPOL0L | A4H     | R/W | 00H     | External Interrupt Polarity 0 Low Register  |
| EIPOL0H | A5H     | R/W | 00H     | External Interrupt Polarity 0 High Register |
| EIFLAG1 | B0H     | R/W | 00H     | External Interrupt Flag 1 Register          |
| EIPOL1  | A6H     | R/W | 00H     | External Interrupt Polarity 1 Register      |
| EIPOL2  | A7H     | R/W | 00H     | External Interrupt Polarity 2 Register      |

## 10.13 Interrupt Register Description

The interrupt register is used for controlling interrupt functions. Also it has external interrupt control registers. The interrupt register consists of interrupt enable register (IE), interrupt enable register 1 (IE1), interrupt enable register 2 (IE2) and interrupt enable register 3 (IE3). For external interrupt, it consists of external interrupt flag 0 register (EIFLAG0), external interrupt polarity 0 high/low register (EIPOL0H/L), external interrupt flag 1 register (EIFLAG1), external interrupt polarity 1 register (EIPOL1) and external interrupt polarity 2 register (EIPOL2).

### 10.13.1 Register Description for Interrupt

#### IE (Interrupt Enable Register) : A8H

| 7  | 6 | 5     | 4 | 3 | 2     | 1     | 0     |
|----|---|-------|---|---|-------|-------|-------|
| EA | – | INT5E | – | – | INT2E | INT1E | INT0E |
| RW | – | RW    | – | – | RW    | RW    | RW    |

Initial value : 00H

|       |                                                            |
|-------|------------------------------------------------------------|
| EA    | Enable or Disable All Interrupt bits                       |
| 0     | All Interrupt disable                                      |
| 1     | All Interrupt enable                                       |
| INT5E | Enable or Disable External Interrupt 0 ~ 4 (EINT0 ~ EINT4) |
| 0     | Disable                                                    |
| 1     | Enable                                                     |
| INT2E | Enable or Disable External Interrupt 12(EINT12)            |
| 0     | Disable                                                    |
| 1     | Enable                                                     |
| INT1E | Enable or Disable External Interrupt 11(EINT11)            |
| 0     | Disable                                                    |
| 1     | Enable                                                     |
| INT0E | Enable or Disable External Interrupt 10 (EINT10)           |
| 0     | Disable                                                    |
| 1     | Enable                                                     |

**IE1 (Interrupt Enable Register 1): A9H**

| 7 | 6 | 5      | 4      | 3     | 2     | 1     | 0     |
|---|---|--------|--------|-------|-------|-------|-------|
| – | – | INT11E | INT10E | INT9E | INT8E | INT7E | INT6E |
| – | – | RW     | RW     | RW    | RW    | RW    | RW    |

Initial value: 00H

|        |                                                |
|--------|------------------------------------------------|
| INT11E | Enable or Disable ADC Wake-up Interrupt        |
| 0      | Disable                                        |
| 1      | Enable                                         |
| INT10E | Enable or Disable UART Tx Interrupt            |
| 0      | Disable                                        |
| 1      | Enable                                         |
| INT9E  | Enable or Disable UART Rx Interrupt            |
| 0      | Disable                                        |
| 1      | Enable                                         |
| INT8E  | Enable or Disable I2C Interrupt                |
| 0      | Disable                                        |
| 1      | Enable                                         |
| INT7E  | Enable or Disable External Interrupt 6 (EINT6) |
| 0      | Disable                                        |
| 1      | Enable                                         |
| INT6E  | Enable or Disable External Interrupt 5 (EINT5) |
| 0      | Disable                                        |
| 1      | Enable                                         |

**IE2 (Interrupt Enable Register 2) : AAH**

| 7 | 6 | 5      | 4 | 3      | 2      | 1      | 0      |
|---|---|--------|---|--------|--------|--------|--------|
| — | — | INT17E | — | INT15E | INT14E | INT13E | INT12E |
| — | — | RW     | — | RW     | RW     | RW     | RW     |

Initial value : 00H

INT17E      Enable or Disable External Interrupt 7 ~ A (EINT7 ~ EINTA)

0      Disable

1      Enable

INT15E      Enable or Disable Timer 2 Match Interrupt

0      Disable

1      Enable

INT14E      Enable or Disable Timer 1 Match Interrupt

0      Disable

1      Enable

INT13E      Enable or Disable Timer 0 Match Interrupt

0      Disable

1      Enable

INT12E      Enable or Disable Timer 0 Overflow Interrupt

0      Disable

1      Enable



**IE3 (Interrupt Enable Register 3) : ABH**

| 7 | 6 | 5 | 4      | 3      | 2      | 1      | 0      |
|---|---|---|--------|--------|--------|--------|--------|
| – | – | – | INT22E | INT21E | INT20E | INT19E | INT18E |
| – | – | – | RW     | RW     | RW     | RW     | RW     |

Initial value : 00H

|        |                                 |
|--------|---------------------------------|
| INT22E | Enable or Disable BIT Interrupt |
| 0      | Disable                         |
| 1      | Enable                          |
| INT21E | Enable or Disable WDT Interrupt |
| 0      | Disable                         |
| 1      | Enable                          |
| INT20E | Enable or Disable WT Interrupt  |
| 0      | Disable                         |
| 1      | Enable                          |
| INT19E | Enable or Disable SPI Interrupt |
| 0      | Disable                         |
| 1      | Enable                          |
| INT18E | Enable or Disable ADC Interrupt |
| 0      | Disable                         |
| 1      | Enable                          |

**IP (Interrupt Priority Register) : B8H**

| 7 | 6 | 5   | 4   | 3   | 2   | 1   | 0   |
|---|---|-----|-----|-----|-----|-----|-----|
| – | – | IP5 | IP4 | IP3 | IP2 | IP1 | IP0 |
| – | – | RW  | RW  | RW  | RW  | RW  | RW  |

Initial value : 00H

**IP1 (Interrupt Priority Register 1) : F8H**

| 7 | 6 | 5    | 4    | 3    | 2    | 1    | 0    |
|---|---|------|------|------|------|------|------|
| – | – | IP15 | IP14 | IP13 | IP12 | IP11 | IP10 |
| – | – | RW   | RW   | RW   | RW   | RW   | RW   |

Initial value : 00H

| IP[5:0], IP1[5:0] Select Interrupt Group Priority |     |                   |
|---------------------------------------------------|-----|-------------------|
| IP1x                                              | IPx | Description       |
| 0                                                 | 0   | level 0 (lowest)  |
| 0                                                 | 1   | level 1           |
| 1                                                 | 0   | level 2           |
| 1                                                 | 1   | level 3 (highest) |

**IIFLAG (Internal Interrupt Flag Register) : A0H**

| 7 | 6 | 5 | 4 | 3 | 2      | 1       | 0     |
|---|---|---|---|---|--------|---------|-------|
| – | – | – | – | – | IICIFR | T0OVIFR | T0IFR |
| – | – | – | – | – | R      | RW      | RW    |

Initial value : 00H

|         |                                                                                                                                                 |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| IICIFR  | This is an interrupt flag bit. When an interrupt occurs, this bit becomes '1'. This bit is cleared when write any values in the I2CSR register. |
| 0       | I2C interrupt no generation                                                                                                                     |
| 1       | I2C interrupt generation                                                                                                                        |
| T0OVIFR | When T0 overflow interrupt occurs, this bit becomes '1'. For clearing bit, write '0' to this bit or auto clear by INT_ACK signal.               |
| 0       | T0 overflow interrupt no generation                                                                                                             |
| 1       | T0 overflow interrupt generation                                                                                                                |
| T0IFR   | When T0 interrupt occurs, this bit becomes '1'. For clearing bit, write '0' to this bit or auto clear by INT_ACK signal.                        |
| 0       | T0 interrupt no generation                                                                                                                      |
| 1       | T0 interrupt generation                                                                                                                         |

**EIFLAG0 (External Interrupt Flag 0 Register) : C0H**

| 7 | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|---|-------|-------|-------|-------|-------|-------|-------|
| – | FLAG6 | FLAG5 | FLAG4 | FLAG3 | FLAG2 | FLAG1 | FLAG0 |
| – | RW    | RW    | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

EIFLAG0[6:5] When an External Interrupt 5 ~ 6 is occurred, the flag becomes '1'.  
The flag is cleared by writing '0' to the bit or automatically cleared by INT\_ACK signal.

0 External Interrupt 5 ~ 6 not occurred

1 External Interrupt 5 ~ 6 occurred

EIFLAG0[4:0] When an External Interrupt 0 ~ 4 is occurred, the flag becomes '1'.  
The flag is cleared only by writing '0' to the bit. So, the flag should be cleared by software.

0 External Interrupt 0 ~ 4 not occurred

1 External Interrupt 0 ~ 4 occurred

**EIPOL0H (External Interrupt Polarity 0 High Register): A5H**

| 7 | 6 | 5    | 4  | 3    | 2  | 1    | 0  |
|---|---|------|----|------|----|------|----|
| – | – | POL6 |    | POL5 |    | POL4 |    |
| – | – | RW   | RW | RW   | RW | RW   | RW |

Initial value: 00H

EIPOL0H[5:0] External interrupt (EINT6, EINT5, EINT4) polarity selection

POLn[1:0] Description

0 0 No interrupt at any edge

0 1 Interrupt on rising edge

1 0 Interrupt on falling edge

1 1 Interrupt on both of rising and falling edge

Where n =4, 5 and 6

**EIPOL0L (External Interrupt Polarity 0 Low Register): A4H**

| 7    | 6  | 5    | 4  | 3    | 2  | 1    | 0  |
|------|----|------|----|------|----|------|----|
| POL3 |    | POL2 |    | POL1 |    | POL0 |    |
| RW   | RW | RW   | RW | RW   | RW | RW   | RW |

Initial value: 00H

EIPOL0L[7:0] External interrupt (EINT3, EINT2, EINT1, EINT0) polarity selection

POLn[1:0] Description

0 0 No interrupt at any edge

0 1 Interrupt on rising edge

1 0 Interrupt on falling edge

1 1 Interrupt on both of rising and falling edge

Where n =0, 1, 2 and 3

**EIFLAG1 (External Interrupt Flag 1 Register) : B0H**

| 7 | 6      | 5      | 4      | 3     | 2     | 1     | 0     |
|---|--------|--------|--------|-------|-------|-------|-------|
| – | FLAG12 | FLAG11 | FLAG10 | FLAGA | FLAG9 | FLAG8 | FLAG7 |
| – | RW     | RW     | RW     | RW    | RW    | RW    | RW    |

Initial value : 00H

EIFLAG1[6:4] When an External Interrupt 10 ~ 12 is occurred, the flag becomes '1'. The flag is cleared by writing '0' to the bit or automatically cleared by INT\_ACK signal.

0 External Interrupt 10 ~ 12 not occurred

1 External Interrupt 10 ~ 12 occurred

EIFLAG1[3:0] When an External Interrupt 7 ~ A is occurred, the flag becomes '1'. The flag is cleared only by writing '0' to the bit. So, the flag should be cleared by software.

0 External Interrupt 7 ~ A not occurred

1 External Interrupt 7 ~ A occurred

**EIPOL1 (External Interrupt Polarity 1 Register): A6H**

| 7 | 6 | 5     | 4  | 3     | 2  | 1     | 0  |
|---|---|-------|----|-------|----|-------|----|
| – | – | POL12 |    | POL11 |    | POL10 |    |
| – | – | RW    | RW | RW    | RW | RW    | RW |

Initial value: 00H

EIPOL1[5:0] External interrupt (EINT12, EINT11, EINT10) polarity selection

POLn[1:0] Description

0 0 No interrupt at any edge

0 1 Interrupt on rising edge

1 0 Interrupt on falling edge

1 1 Interrupt on both of rising and falling edge

Where n =10, 11 and 12

**EIPOL2 (External Interrupt Polarity 2 Register): A7H**

| 7    | 6  | 5    | 4  | 3    | 2  | 1    | 0  |
|------|----|------|----|------|----|------|----|
| POLA |    | POL9 |    | POL8 |    | POL7 |    |
| RW   | RW | RW   | RW | RW   | RW | RW   | RW |

Initial value: 00H

EIPOL2[7:0] External interrupt (EINTA, EINT9, EINT8, EINT7) polarity selection

POLn[1:0] Description

0 0 No interrupt at any edge

0 1 Interrupt on rising edge

1 0 Interrupt on falling edge

1 1 Interrupt on both of rising and falling edge

Where n = 7, 8, 9 and A

## 11. Peripheral Hardware

### 11.1 Clock Generator

#### 11.1.1 Overview

As shown in Figure 11.1, the clock generator produces the basic clock pulses which provide the system clock to be supplied to the CPU and the peripheral hardware. It contains main/sub-frequency clock oscillator. The main/sub clock operation can be easily obtained by attaching a crystal between the XIN/SXIN and XOUT/SXOUT pin, respectively. The main/sub clock can be also obtained from the external oscillator. In this case, it is necessary to put the external clock signal into the XIN/SXIN pin and open the XOUT/SXOUT pin. The default system clock is 1MHz INT-RC Oscillator and the default division rate is sixteen. In order to stabilize system internally, it is used 1MHz INT-RC oscillator on POR.

- Calibrated Internal RC Oscillator (16 MHz)
  - . INT-RC OSC/1 (16 MHz)
  - . INT-RC OSC/2 (8 MHz)
  - . INT-RC OSC/4 (4 MHz)
  - . INT-RC OSC/8 (2 MHz)
  - . INT-RC OSC/16 (1 MHz, Default system clock)
  - . INT-RC OSC/32 (0.5 MHz)
- Main Crystal Oscillator (0.4~12 MHz)
- Sub Crystal Oscillator (32.768 kHz)
- Internal WDTRC Oscillator (5 kHz)

#### 11.1.2 Block Diagram

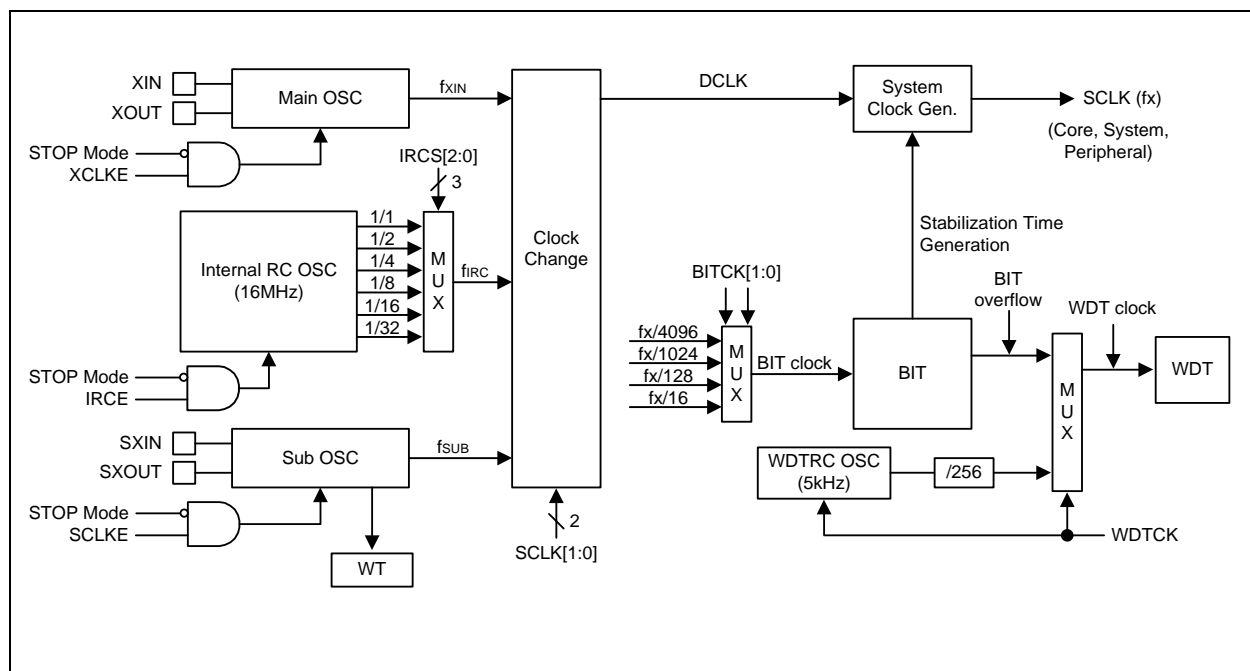


Figure 11.1 Clock Generator Block Diagram

### 11.1.3 Register Map

Table 11-1 Clock Generator Register Map

| Name  | Address | Dir | Default | Description                       |
|-------|---------|-----|---------|-----------------------------------|
| SCCR  | 8AH     | R/W | 00H     | System and Clock Control Register |
| OSCCR | C8H     | R/W | 08H     | Oscillator Control Register       |

### 11.1.4 Clock Generator Register Description

The clock generator register uses clock control for system operation. The clock generation consists of System and clock control register and oscillator control register.

### 11.1.5 Register Description for Clock Generator

#### SCCR (System and Clock Control Register) : 8AH

| 7 | 6 | 5 | 4 | 3 | 2 | 1     | 0     |
|---|---|---|---|---|---|-------|-------|
| – | – | – | – | – | – | SCLK1 | SCLK0 |
| – | – | – | – | – | – | RW    | RW    |

Initial value : 00H

SCLK [1:0]

System Clock Selection Bit

SCLK1 SCLK0 Description

0 0 INT RC OSC ( $f_{IRC}$ ) for system clock

0 1 External Main OSC ( $f_{XIN}$ ) for system clock

1 0 External Sub OSC ( $f_{SUB}$ ) for system clock

1 1 Not used

**OSCCR (Oscillator Control Register) : C8H**

| 7 | 6 | 5     | 4     | 3     | 2    | 1     | 0     |
|---|---|-------|-------|-------|------|-------|-------|
| – | – | IRCS2 | IRCS1 | IRCS0 | IRCE | XCLKE | SCLKE |
| – | – | RW    | RW    | RW    | RW   | RW    | RW    |

Initial value : 08H

|           |                                                       |                                 |       |                    |
|-----------|-------------------------------------------------------|---------------------------------|-------|--------------------|
| IRCS[2:0] | Internal RC Oscillator Post-divider Selection         |                                 |       |                    |
|           | IRCS2                                                 | IRCS1                           | IRCS0 | Description        |
|           | 0                                                     | 0                               | 0     | INT-RC/32 (0.5MHz) |
|           | 0                                                     | 0                               | 1     | INT-RC/16 (1MHz)   |
|           | 0                                                     | 1                               | 0     | INT-RC/8 (2MHz)    |
|           | 0                                                     | 1                               | 1     | INT-RC/4 (4MHz)    |
|           | 1                                                     | 0                               | 0     | INT-RC/2 (8MHz)    |
|           | 1                                                     | 0                               | 1     | INT-RC/1 (16MHz)   |
|           | Other values                                          |                                 |       | Not used           |
| IRCE      | Control the Operation of the Internal RC Oscillator   |                                 |       |                    |
|           | 0                                                     | Enable operation of INT-RC OSC  |       |                    |
|           | 1                                                     | Disable operation of INT-RC OSC |       |                    |
| XCLKE     | Control the Operation of the External Main Oscillator |                                 |       |                    |
|           | 0                                                     | Disable operation of X-TAL      |       |                    |
|           | 1                                                     | Enable operation of X-TAL       |       |                    |
| SCLKE     | Control the Operation of the External Sub Oscillator  |                                 |       |                    |
|           | 0                                                     | Disable operation of SX-TAL     |       |                    |
|           | 1                                                     | Enable operation of SX-TAL      |       |                    |

## 11.2 Basic Interval Timer

### 11.2.1 Overview

The MC96F8316 has one 8-bit basic interval timer that is free-run and can't stop. Block diagram is shown in Figure 11.2. In addition, the basic interval timer generates the time base for watchdog timer counting. It also provides a basic interval timer interrupt (BITIFR).

The MC96F8316 has these basic interval timer (BIT) features:

- During Power On, BIT gives a stable clock generation time
- On exiting Stop mode, BIT gives a stable clock generation time
- As timer function, timer interrupt occurrence

### 11.2.2 Block Diagram

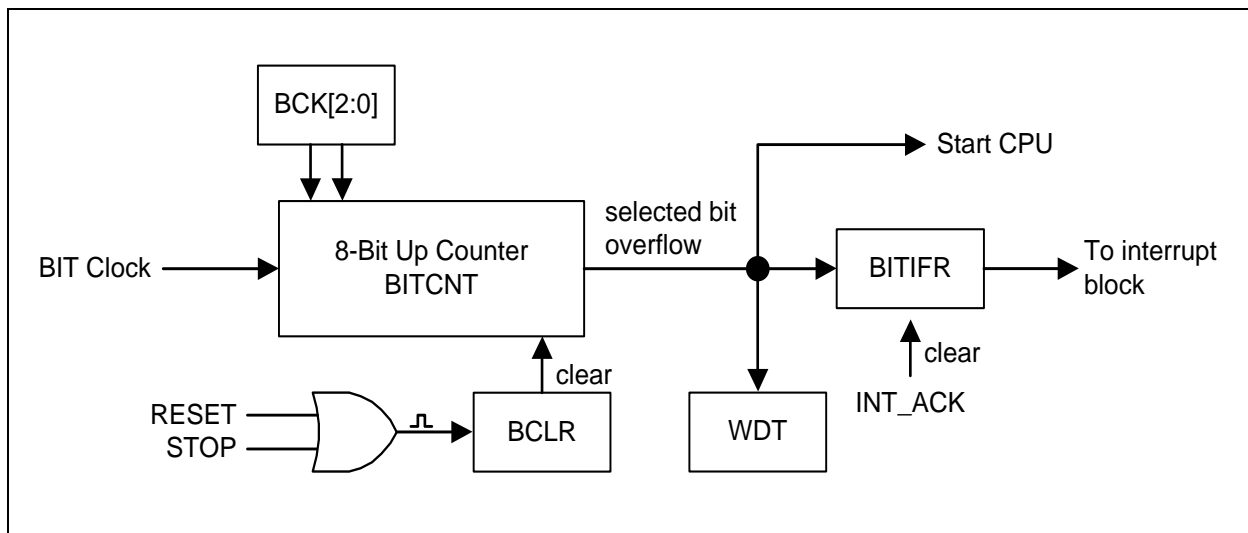


Figure 11.2 Basic Interval Timer Block Diagram



### 11.2.3 Register Map

Table 11-2 Basic Interval Timer Register Map

| Name   | Address | Dir | Default | Description                           |
|--------|---------|-----|---------|---------------------------------------|
| BITCNT | 8CH     | R   | 00H     | Basic Interval Timer Counter Register |
| BITCR  | 8BH     | R/W | 01H     | Basic Interval Timer Control Register |

### 11.2.4 Basic Interval Timer Register Description

The basic interval timer register consists of basic interval timer counter register (BITCNT) and basic interval timer control register (BITCR). If BCLR bit is set to '1', BITCNT becomes '0' and then counts up. After 1 machine cycle, BCLR bit is cleared to '0' automatically.

### 11.2.5 Register Description for Basic Interval Timer

#### BITCNT (Basic Interval Timer Counter Register) : 8CH

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| BITCNT7 | BITCNT6 | BITCNT5 | BITCNT4 | BITCNT3 | BITCNT2 | BITCNT1 | BITCNT0 |
| R       | R       | R       | R       | R       | R       | R       | R       |

Initial value : 00H

BITCNT[7:0]      BIT Counter

**BITCR (Basic Interval Timer Control Register) : 8BH**

| 7      | 6      | 5      | 4 | 3    | 2    | 1    | 0    |
|--------|--------|--------|---|------|------|------|------|
| BITIFR | BITCK1 | BITCK0 | – | BCLR | BCK2 | BCK1 | BCK0 |
| RW     | RW     | RW     | – | RW   | RW   | RW   | RW   |

Initial value : 01H

BITIFR When BIT Interrupt occurs, this bit becomes '1'. For clearing bit, write '0' to this bit or auto clear by INT\_ACK signal.

0 BIT interrupt no generation

1 BIT interrupt generation

BITCK[1:0] Select BIT clock source

| BITCK1 | BITCK0 | Description |
|--------|--------|-------------|
| 0      | 0      | fx/4096     |
| 0      | 1      | fx/1024     |
| 1      | 0      | fx/128      |
| 1      | 1      | fx/16       |

BCLR If this bit is written to '1', BIT Counter is cleared to '0'

0 Free Running

1 Clear Counter

BCK[2:0] Select BIT overflow period

| BCK2 | BCK1 | BCK0 | Description                              |
|------|------|------|------------------------------------------|
| 0    | 0    | 0    | Bit 0 overflow (BIT Clock * 2)           |
| 0    | 0    | 1    | Bit 1 overflow (BIT Clock * 4) (default) |
| 0    | 1    | 0    | Bit 2 overflow (BIT Clock * 8)           |
| 0    | 1    | 1    | Bit 3 overflow (BIT Clock * 16)          |
| 1    | 0    | 0    | Bit 4 overflow (BIT Clock * 32)          |
| 1    | 0    | 1    | Bit 5 overflow (BIT Clock * 64)          |
| 1    | 1    | 0    | Bit 6 overflow (BIT Clock * 128)         |
| 1    | 1    | 1    | Bit 7 overflow (BIT Clock * 256)         |

## 11.3 Watch Dog Timer

### 11.3.1 Overview

The watchdog timer rapidly detects the CPU malfunction such as endless looping caused by noise or something like that, and resumes the CPU to the normal state. The watchdog timer signal for malfunction detection can be used as either a CPU reset or an interrupt request. When the watchdog timer is not being used for malfunction detection, it can be used as a timer to generate an interrupt at fixed intervals. It is possible to use free running 8-bit timer mode (WDTRSON='0') or watch dog timer mode (WDTRSON='1') as setting WDTCR[6] bit. If WDTCR[5] is written to '1', WDT counter value is cleared and counts up. After 1 machine cycle, this bit is cleared to '0' automatically. The watchdog timer consists of 8-bit binary counter and the watchdog timer data register. When the value of 8-bit binary counter is equal to the 8 bits of WDTCNT, the interrupt request flag is generated. This can be used as Watchdog timer interrupt or reset of CPU in accordance with the bit WDTRSON.

The input clock source of watch dog timer is the BIT overflow. The interval of watchdog timer interrupt is decided by BIT overflow period and WDTDR set value. The equation can be described as

$$\text{WDT Interrupt Interval} = (\text{BIT Interrupt Interval}) \times (\text{WDTDR Value} + 1)$$

### 11.3.2 WDT Interrupt Timing Waveform

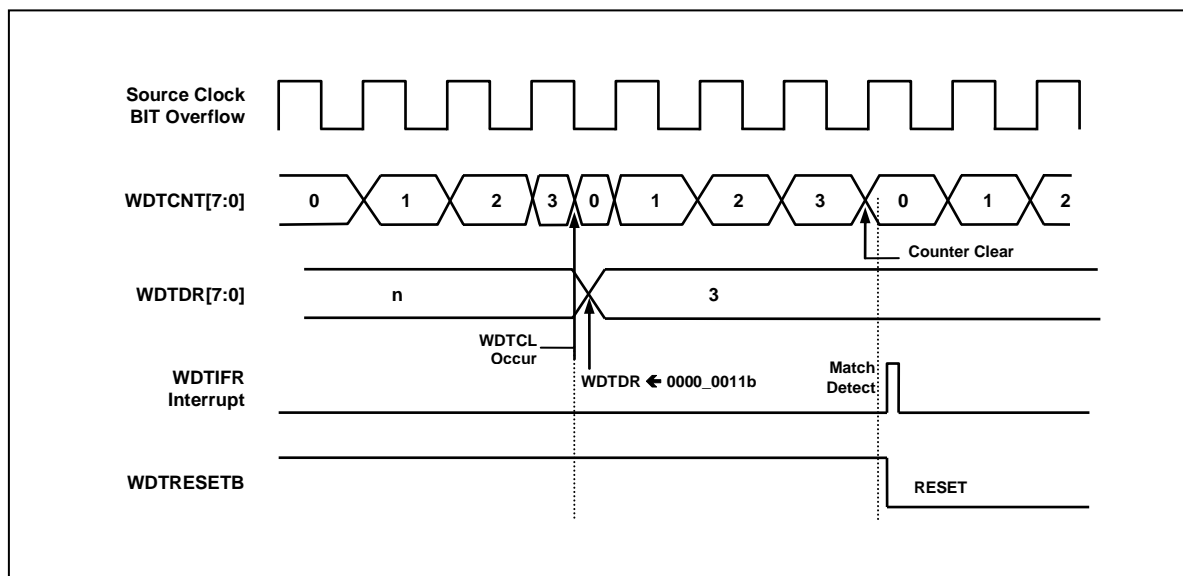
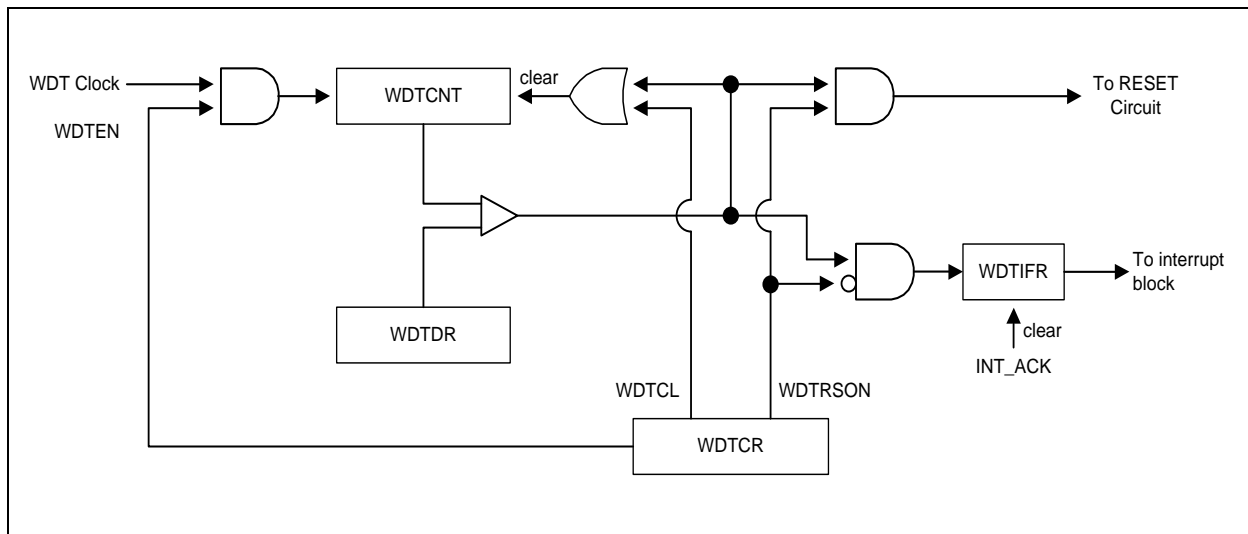


Figure 11.3 Watch Dog Timer Interrupt Timing Waveform

### 11.3.3 Block Diagram



### Figure 11.4 Watch Dog Timer Block Diagram

### 11.3.4 Register Map

### Table 11-3 Watch Dog Timer Register Map

| Name   | Address | Dir | Default | Description                      |
|--------|---------|-----|---------|----------------------------------|
| WDTCNT | 8EH     | R   | 00H     | Watch Dog Timer Counter Register |
| WDTDR  | 8EH     | W   | FFH     | Watch Dog Timer Data Register    |
| WDTCR  | 8DH     | R/W | 00H     | Watch Dog Timer Control Register |

### 11.3.5 Watch Dog Timer Register Description

The watch dog timer register consists of watch dog timer counter register (WDTCNT), watch dog timer data register (WDTDR) and watch dog timer control register (WDTCR).

### 11.3.6 Register Description for Watch Dog Timer

#### WDCNT (Watch Dog Timer Counter Register: Read Case) : 8EH

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| WDCNT7 | WDCNT6 | WDCNT5 | WDCNT4 | WDCNT3 | WDCNT2 | WDCNT1 | WDCNT0 |
| R      | R      | R      | R      | R      | R      | R      | R      |

Initial value : 00H

WDCNT[7:0] WDT Counter

#### WDTDR (Watch Dog Timer Data Register: Write Case) : 8EH

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| WDTDR7 | WDTDR6 | WDTDR5 | WDTDR4 | WDTDR3 | WDTDR2 | WDTDR1 | WDTDR0 |
| W      | W      | W      | W      | W      | W      | W      | W      |

Initial value : FFH

WDTDR[7:0] Set a period  
WDT Interrupt Interval=(BIT Interrupt Interval) x(WDTDR Value+1)  
NOTE) Do not write "0" in the WDTDR register.

#### WDTCR (Watch Dog Timer Control Register) : 8DH

| 7     | 6        | 5     | 4 | 3 | 2 | 1     | 0      |
|-------|----------|-------|---|---|---|-------|--------|
| WDTEN | WDTTRSON | WDTCL | — | — | — | WDTCK | WDTIFR |
| R/W   | R/W      | R/W   | — | — | — | R/W   | R/W    |

Initial value : 00H

WDTEN Control WDT Operation  
0 Disable  
1 Enable

WDTTRSON Control WDT RESET Operation  
0 Free Running 8-bit timer  
1 Watch Dog Timer RESET ON

WDTCL Clear WDT Counter  
0 Free Run  
1 Clear WDT Counter (auto clear after 1 Cycle)

WDTCK Control WDT Clock Selection Bit  
0 BIT overflow for WDT clock (WDTTRC disable)  
1 WDTTRC for WDT clock (WDTTRC enable)

WDTIFR When WDT Interrupt occurs, this bit becomes '1'. For clearing bit, write '0' to this bit or auto clear by INT\_ACK signal.  
0 WDT Interrupt no generation  
1 WDT Interrupt generation

## 11.4 Watch Timer

### 11.4.1 Overview

The watch timer has the function for RTC (Real Time Clock) operation. It is generally used for RTC design. The internal structure of the watch timer consists of the clock source select circuit, timer counter circuit, output select circuit, and watch timer control register. To operate the watch timer, determine the input clock source, output interval, and set WTEN to '1' in watch timer control register (WTCR). It is able to execute simultaneously or individually. To stop or reset WT, clear the WTEN bit in WTCR register. Even if CPU is STOP mode, sub clock is able to be so alive that WT can continue the operation. The watch timer counter circuits may be composed of 21-bit counter which contains low 14-bit with binary counter and high 7-bit counter in order to raise resolution. In WTDR, it can control WT clear and set interval value at write time, and it can read 7-bit WT counter value at read time.

### 11.4.2 Block Diagram

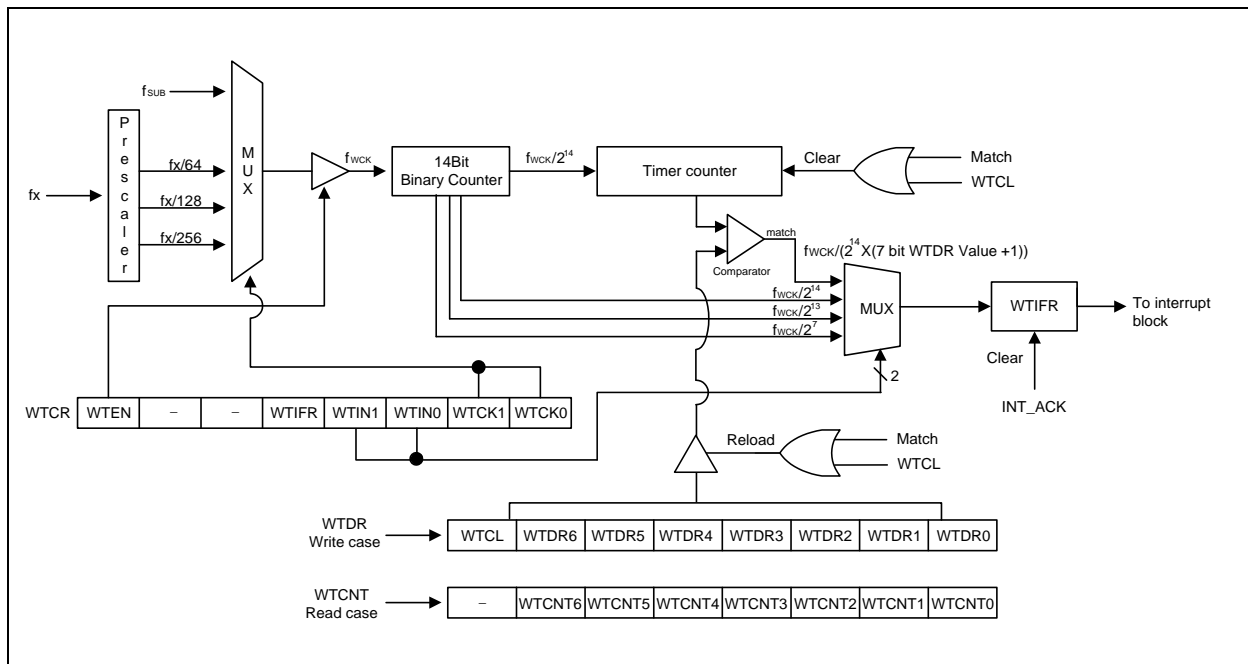


Figure 11.5 Watch Timer Block Diagram

### 11.4.3 Register Map

Table 11-4 Watch Timer Register Map

| Name  | Address | Dir | Default | Description                  |
|-------|---------|-----|---------|------------------------------|
| WTCNT | 89H     | R   | 00H     | Watch Timer Counter Register |
| WTDR  | 89H     | W   | 7FH     | Watch Timer Data Register    |
| WTCR  | 96H     | R/W | 00H     | Watch Timer Control Register |

### 11.4.4 Watch Timer Register Description

The watch timer register consists of watch timer counter register (WTCNT), watch timer data register (WTDR), and watch timer control register (WTCR). As WTCR is 6-bit writable/readable register, WTCR can control the clock source (WTCK[1:0]), interrupt interval (WTIN[1:0]), and function enable/disable (WTEN). Also there is WT interrupt flag bit (WTIFR).

### 11.4.5 Register Description for Watch Timer

#### WTCNT (Watch Timer Counter Register: Read Case) : 89H

| 7 | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|---|--------|--------|--------|--------|--------|--------|--------|
| – | WTCNT6 | WTCNT5 | WTCNT4 | WTCNT3 | WTCNT2 | WTCNT1 | WTCNT0 |
| – | R      | R      | R      | R      | R      | R      | R      |

Initial value : 00H

WTCNT[6:0] WT Counter

#### WTDR (Watch Timer Data Register: Write Case) : 89H

| 7    | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|------|-------|-------|-------|-------|-------|-------|-------|
| WTCL | WTDR6 | WTDR5 | WTDR4 | WTDR3 | WTDR2 | WTDR1 | WTDR0 |
| RW   | W     | W     | W     | W     | W     | W     | W     |

Initial value : 7FH

WTCL Clear WT Counter  
 0 Free Run  
 1 Clear WT Counter (auto clear after 1 Cycle)

WTDR[6:0] Set WT period  
 $WT\ Interrupt\ Interval = fwck / (2^{14} \times (7bit\ WTDR\ Value + 1))$   
 NOTE) Do not write "0" in the WTDR register.

**WTCR (Watch Timer Control Register) : 96H**

| 7    | 6 | 5 | 4     | 3     | 2     | 1     | 0     |
|------|---|---|-------|-------|-------|-------|-------|
| WTEN | – | – | WTIFR | WTIN1 | WTIN0 | WTCK1 | WTCK0 |
| RW   | – | – | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

WTEN Control Watch Timer

0 Disable

1 Enable

WTIFR When WT Interrupt occurs, this bit becomes '1'. For clearing bit, write '0' to this bit or automatically clear by INT\_ACK signal.

0 WT Interrupt no generation

1 WT Interrupt generation

WTIN[1:0] Determine interrupt interval

| WTIN1 | WTIN0 | Description                                          |
|-------|-------|------------------------------------------------------|
| 0     | 0     | $f_{WCK}/2^7$                                        |
| 0     | 1     | $f_{WCK}/2^{13}$                                     |
| 1     | 0     | $f_{WCK}/2^{14}$                                     |
| 1     | 1     | $f_{WCK}/(2^{14} \times (7\text{bit WTDR Value}+1))$ |

WTCK[1:0] Determine Source Clock

| WTCK1 | WTCK0 | Description |
|-------|-------|-------------|
| 0     | 0     | $f_{SUB}$   |
| 0     | 1     | $f_X/256$   |
| 1     | 0     | $f_X/128$   |
| 1     | 1     | $f_X/64$    |

NOTE)  $f_X$  – System clock frequency (Where  $f_X = 4.19\text{MHz}$ ) $f_{SUB}$  – Sub clock oscillator frequency (32.768kHz) $f_{WCK}$  – Selected Watch timer clock



## 11.5 Timer 0

### 11.5.1 Overview

The 8-bit timer 0 consists of multiplexer, timer 0 counter register, timer 0 data register, timer 0 capture data register and timer 0 control register (T0CNT, T0DR, T0CDR, T0CR).

It has three operating modes:

- 8-bit timer/counter mode
- 8-bit PWM output mode
- 8-bit capture mode

The timer/counter 0 can be clocked by an internal or an external clock source (EC0). The clock source is selected by clock selection logic which is controlled by the clock selection bits (T0CK[2:0]).

- TIMER 0 clock source:  $f_x/2$ , 4, 8, 32, 128, 512, 2048 and EC0

In the capture mode, by EINT10, the data is captured into input capture data register (T0CDR). In timer/counter mode, whenever counter value is equal to T0DR, T0O port toggles. Also the timer 0 outputs PWM waveform through PWM0O port in the PWM mode.

**Table 11-5 Timer 0 Operating Modes**

| T0EN | T0MS[1:0] | T0CK[2:0] | Timer 0                  |
|------|-----------|-----------|--------------------------|
| 1    | 00        | XXX       | 8 Bit Timer/Counter Mode |
| 1    | 01        | XXX       | 8 Bit PWM Mode           |
| 1    | 1X        | XXX       | 8 Bit Capture Mode       |

### 11.5.2 8-Bit Timer/Counter Mode

The 8-bit timer/counter mode is selected by control register as shown in Figure 11.6.

The 8-bit timer have counter and data register. The counter register is increased by internal or external clock input. Timer 0 can use the input clock with one of 2, 4, 8, 32, 128, 512 and 2048 prescaler division rates (T0CK[2:0]). When the value of T0CNT and T0DR is identical in timer 0, a match signal is generated and the interrupt of Timer 0 occurs. T0CNT value is automatically cleared by match signal. It can be also cleared by software (T0CC).

The external clock (EC0) counts up the timer at the rising edge. If the EC0 is selected as a clock source by T0CK[2:0], EC0 port should be set to the input port by P26IO bit.

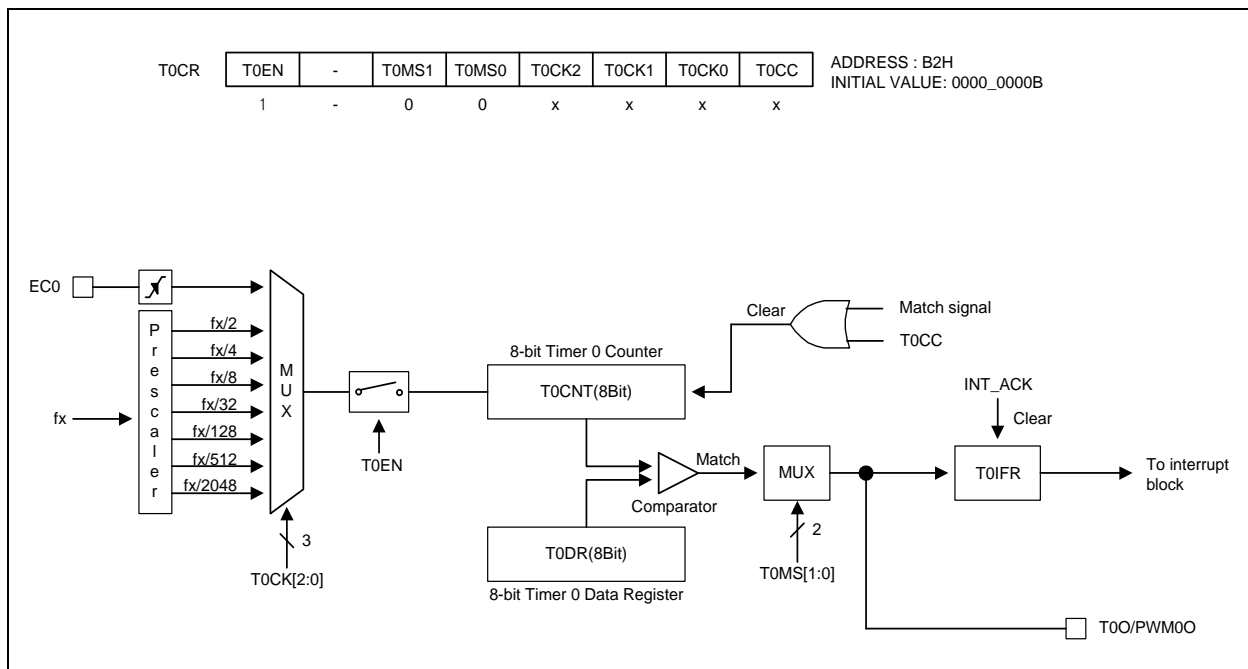


Figure 11.6 8-Bit Timer/Counter Mode for Timer 0

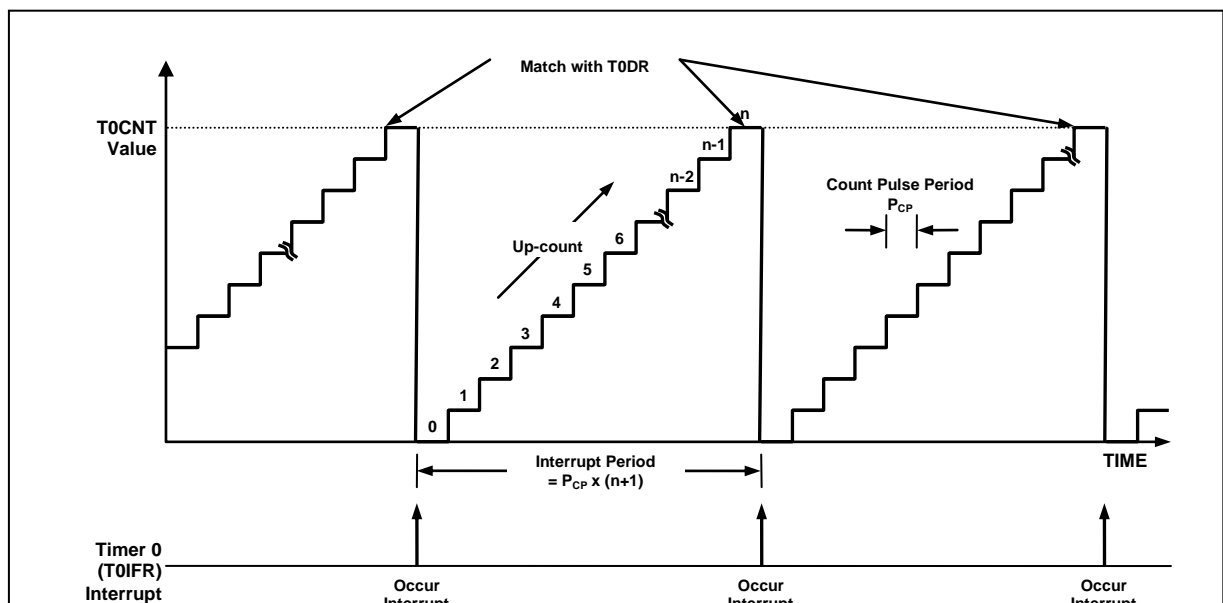


Figure 11.7 8-Bit Timer/Counter 0 Example

### 11.5.3 8-Bit PWM Mode

The timer 0 has a high speed PWM (Pulse Width Modulation) function. In PWM mode, T0O/PWM0O pin outputs up to 8-bit resolution PWM output. This pin should be configured as a PWM output by setting the T0O/PWM0O function by P3FSR[5] bit. In the 8-bit timer/counter mode, a match signal is generated when the counter value is identical to the value of T0DR. When the value of T0CNT and T0DR is identical in timer 0, a match signal is generated and the interrupt of timer 0 occurs. In PWM mode, the match signal does not clear the counter. Instead, it runs continuously, overflowing at "FFH", and then continues incrementing from "00H". The timer 0 overflow interrupt is generated whenever a counter overflow occurs. T0CNT value is cleared by software (T0CC) bit.

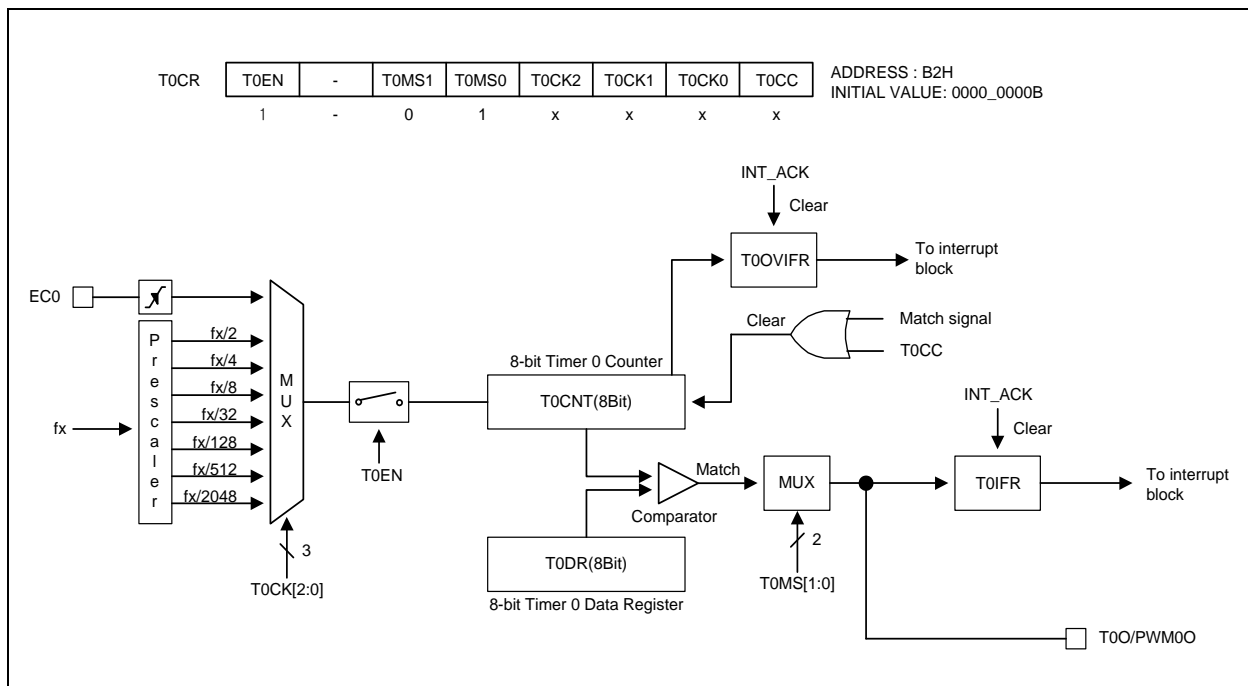


Figure 11.8 8-Bit PWM Mode for Timer 0

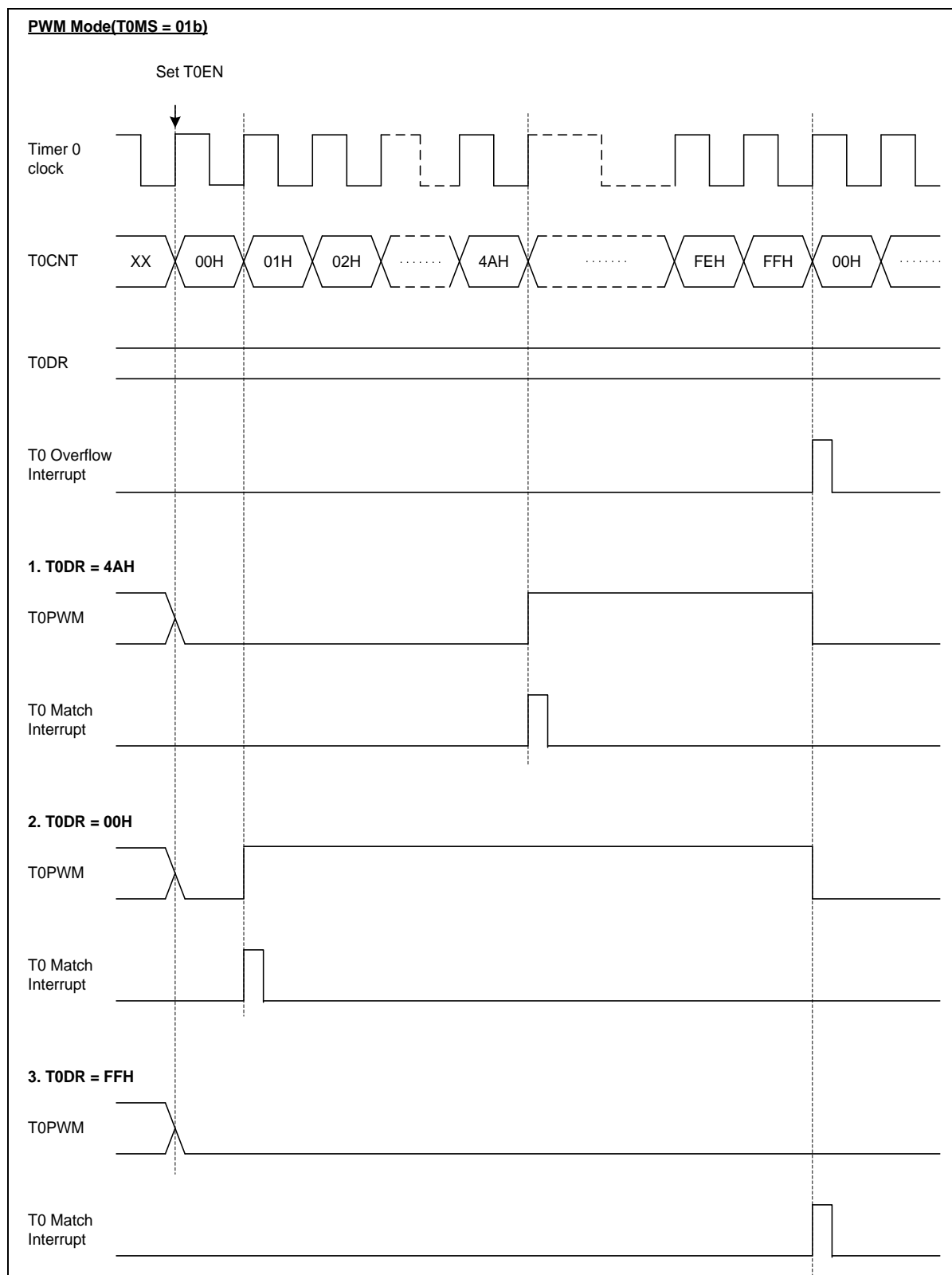


Figure 11.9 PWM Output Waveforms in PWM Mode for Timer 0

### 11.5.4 8-Bit Capture Mode

The timer 0 capture mode is set by T0MS[1:0] as '1x'. The clock source can use the internal/external clock. Basically, it has the same function as the 8-bit timer/counter mode and the interrupt occurs when T0CNT is equal to T0DR. T0CNT value is automatically cleared by match signal and it can be also cleared by software (T0CC).

This timer interrupt in capture mode is very useful when the pulse width of captured signal is wider than the maximum period of timer.

The capture result is loaded into T0CDR. In the timer 0 capture mode, timer 0 output (T0O) waveform is not available.

According to EIPOL1 registers setting, the external interrupt EINT10 function is chosen. Of course, the EINT10 pin must be set to an input port.

T0CDR and T0DR are in the same address. In the capture mode, reading operation reads T0CDR, not T0DR and writing operation will update T0DR.

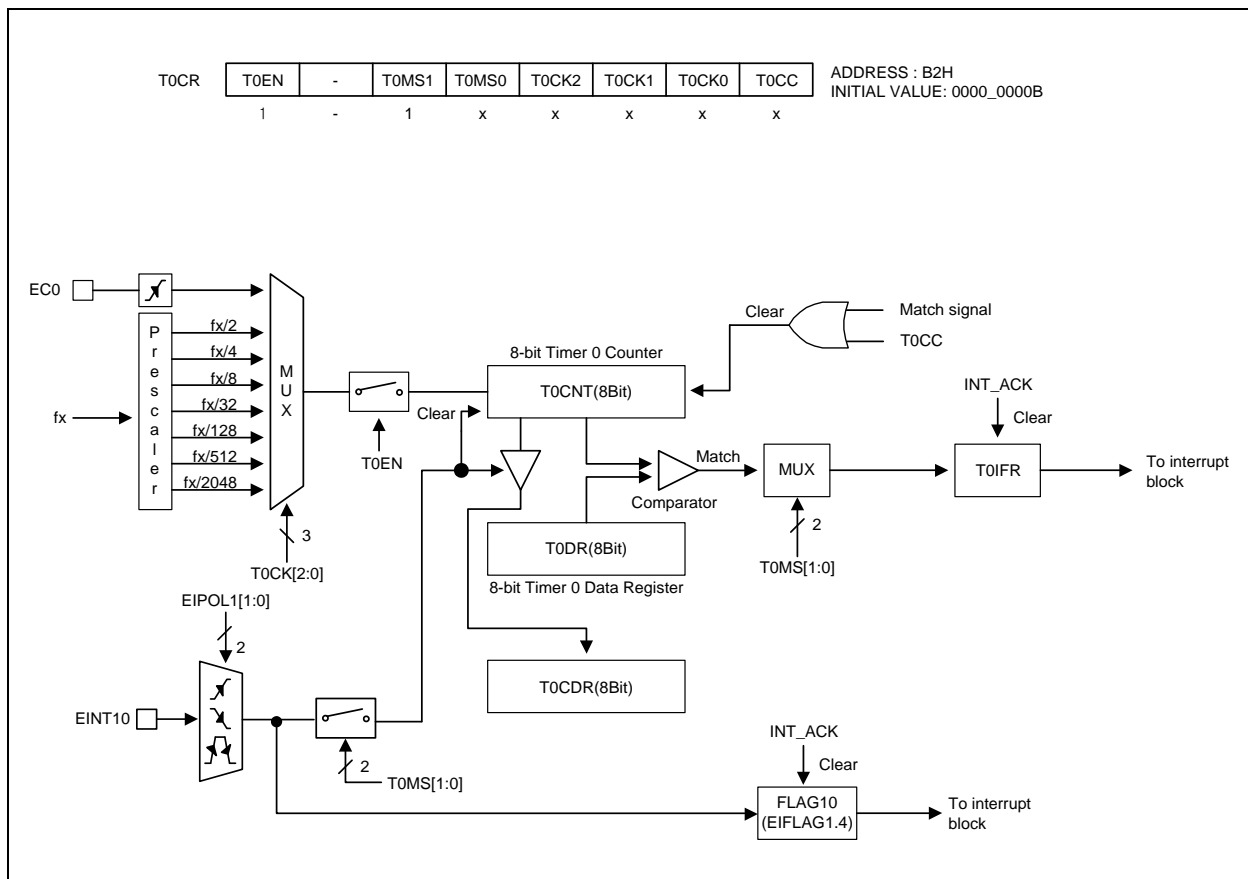


Figure 11.10 8-Bit Capture Mode for Timer 0

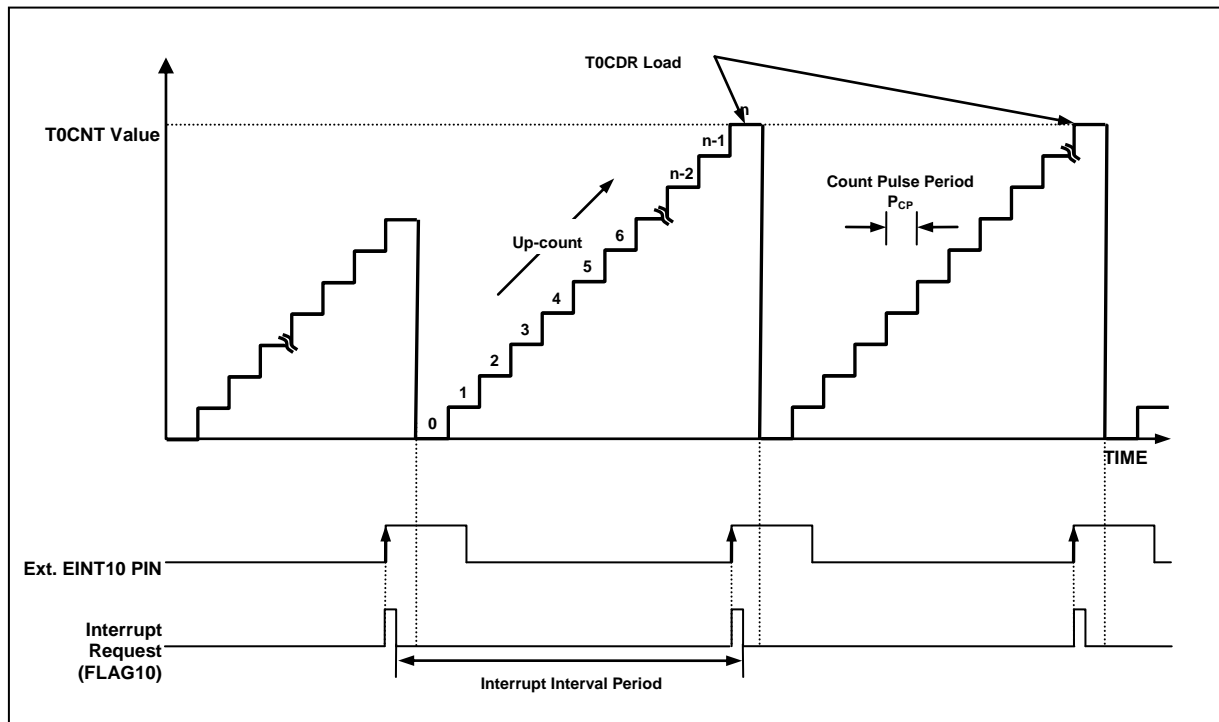


Figure 11.11 Input Capture Mode Operation for Timer 0

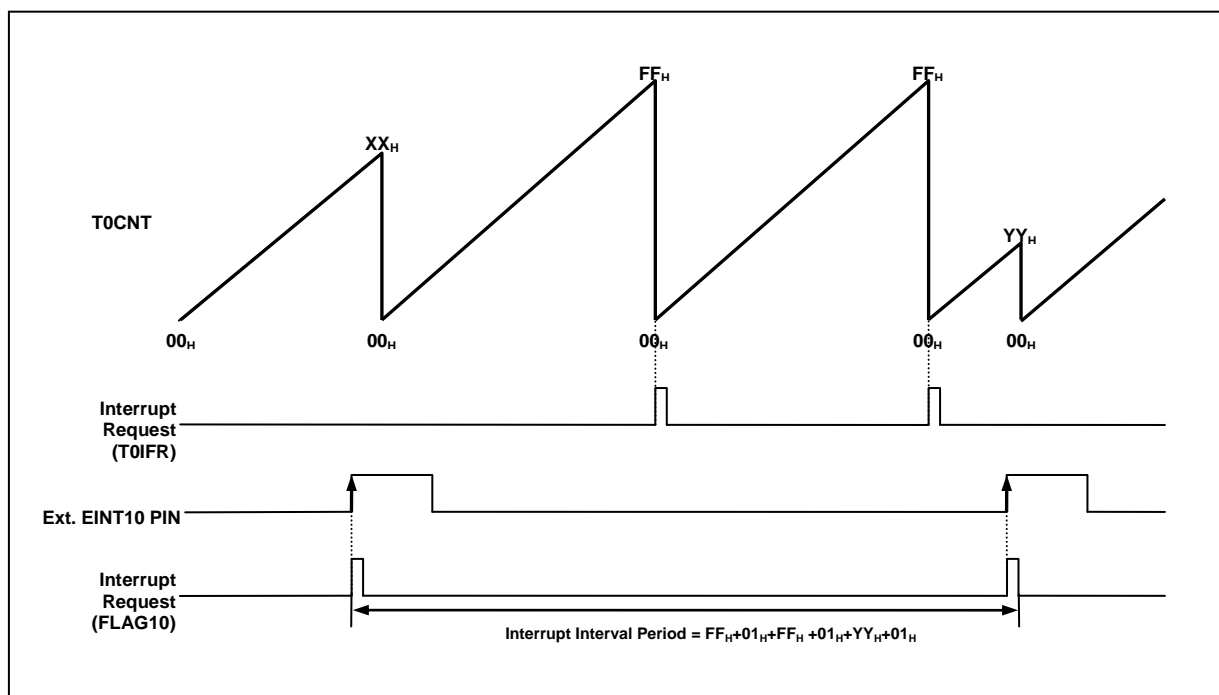


Figure 11.12 Express Timer Overflow in Capture Mode

### 11.5.5 Block Diagram

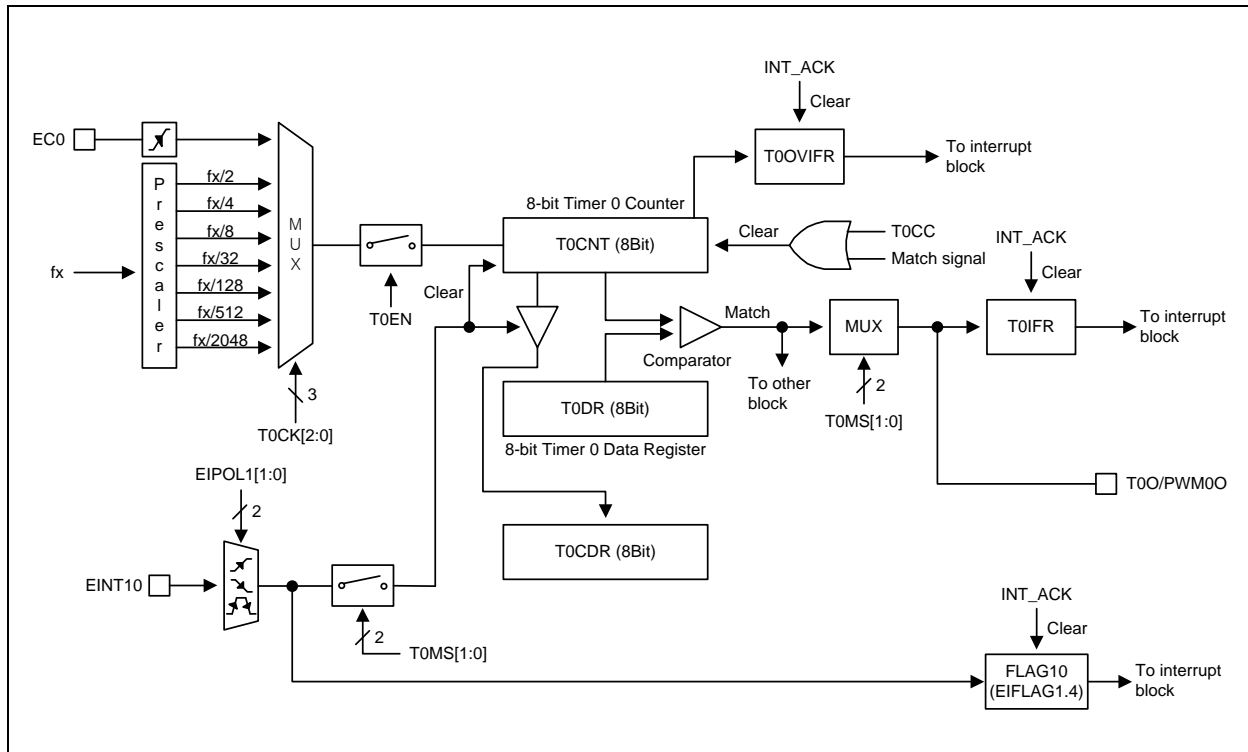


Figure 11.13 8-Bit Timer 0 Block Diagram

## 11.5.6 Register Map

Table 11-6 Timer 0 Register Map

| Name  | Address | Dir | Default | Description                   |
|-------|---------|-----|---------|-------------------------------|
| T0CNT | B3H     | R   | 00H     | Timer 0 Counter Register      |
| T0DR  | B4H     | R/W | FFH     | Timer 0 Data Register         |
| T0CDR | B4H     | R   | 00H     | Timer 0 Capture Data Register |
| T0CR  | B2H     | R/W | 00H     | Timer 0 Control Register      |

### 11.5.6.1 Timer/Counter 0 Register Description

The timer/counter 0 register consists of timer 0 counter register (T0CNT), timer 0 data register (T0DR), timer 0 capture data register (T0CDR), and timer 0 control register (T0CR). T0IFR and T0OVIFR bits are in the internal interrupt flag register (IIFLAG).

### 11.5.6.2 Register Description for Timer/Counter 0

#### T0CNT (Timer 0 Counter Register) : B3H

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| T0CNT7 | T0CNT6 | T0CNT5 | T0CNT4 | T0CNT3 | T0CNT2 | T0CNT1 | T0CNT0 |
| R      | R      | R      | R      | R      | R      | R      | R      |

Initial value : 00H

T0CNT[7:0] T0 Counter

#### T0DR (Timer 0 Data Register) : B4H

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| T0DR7 | T0DR6 | T0DR5 | T0DR4 | T0DR3 | T0DR2 | T0DR1 | T0DR0 |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |

Initial value : FFH

T0DR[7:0] T0 Data

#### T0CDR (Timer 0 Capture Data Register: Read Case, Capture mode only) : B4H

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| T0CDR7 | T0CDR6 | T0CDR5 | T0CDR4 | T0CDR3 | T0CDR2 | T0CDR1 | T0CDR0 |
| R      | R      | R      | R      | R      | R      | R      | R      |

Initial value : 00H

T0CDR[7:0] T0 Capture Data



**T0CR (Timer 0 Control Register) : B2H**

| 7    | 6 | 5     | 4     | 3     | 2     | 1     | 0    |
|------|---|-------|-------|-------|-------|-------|------|
| T0EN | – | T0MS1 | T0MS0 | T0CK2 | T0CK1 | T0CK0 | T0CC |
| RW   | – | RW    | RW    | RW    | RW    | RW    | RW   |

Initial value : 00H

|           |                                                             |                                                                                               |                                              |                      |
|-----------|-------------------------------------------------------------|-----------------------------------------------------------------------------------------------|----------------------------------------------|----------------------|
| T0EN      | Control Timer 0                                             |                                                                                               |                                              |                      |
|           | 0                                                           | Timer 0 disable                                                                               |                                              |                      |
|           | 1                                                           | Timer 0 enable                                                                                |                                              |                      |
| T0MS[1:0] | Control Timer 0 Operation Mode                              |                                                                                               |                                              |                      |
|           | T0MS1                                                       | T0MS0                                                                                         | Description                                  |                      |
|           | 0                                                           | 0                                                                                             | Timer/counter mode (T0O: toggle at match)    |                      |
|           | 0                                                           | 1                                                                                             | PWM mode (The overflow interrupt can occur)  |                      |
|           | 1                                                           | x                                                                                             | Capture mode (The match interrupt can occur) |                      |
| T0CK[2:0] | Select Timer 0 clock source. fx is a system clock frequency |                                                                                               |                                              |                      |
|           | T0CK2                                                       | T0CK1                                                                                         | T0CK0                                        | Description          |
|           | 0                                                           | 0                                                                                             | 0                                            | fx/2                 |
|           | 0                                                           | 0                                                                                             | 1                                            | fx/4                 |
|           | 0                                                           | 1                                                                                             | 0                                            | fx/8                 |
|           | 0                                                           | 1                                                                                             | 1                                            | fx/32                |
|           | 1                                                           | 0                                                                                             | 0                                            | fx/128               |
|           | 1                                                           | 0                                                                                             | 1                                            | fx/512               |
|           | 1                                                           | 1                                                                                             | 0                                            | fx/2048              |
|           | 1                                                           | 1                                                                                             | 1                                            | External Clock (EC0) |
| T0CC      | Clear timer 0 Counter                                       |                                                                                               |                                              |                      |
|           | 0                                                           | No effect                                                                                     |                                              |                      |
|           | 1                                                           | Clear the Timer 0 counter (When write, automatically cleared "0" after being cleared counter) |                                              |                      |

NOTE) Refer to the internal interrupt flag register (IIFLAG) for the T0 interrupt flags.

## 11.6 Timer 1

### 11.6.1.1 Overview

The 16-bit timer 1 consists of multiplexer, timer 1 A data register high/low, timer 1 B data register high/low and timer 1 control register high/low (T1ADRH, T1ADRL, T1BDRH, T1BDRL, T1CRH, T1CRL).

It has four operating modes:

- 16-bit timer/counter mode
- 16-bit capture mode
- 16-bit PPG output mode (one-shot mode)
- 16-bit PPG output mode (repeat mode)

The timer/counter 1 can be clocked by an internal or an external clock source (EC1). The clock source is selected by clock selection logic which is controlled by the clock selection bits (T1CK[2:0]).

- TIMER 1 clock source:  $f_x/1, 2, 4, 8, 64, 512, 2048$  and EC1

In the capture mode, by EINT11, the data is captured into input capture data register (T1BDRH/T1BDRL). Timer 1 outputs the comparison result between counter and data register through T1O port in timer/counter mode. Also Timer 1 outputs PWM wave form through PWM1O port in the PPG mode.

**Table 11-7 Timer 1 Operating Modes**

| T1EN | P1FSRL[4:3] | T1MS[1:0] | T1CK[2:0] | Timer 1                            |
|------|-------------|-----------|-----------|------------------------------------|
| 1    | 01          | 00        | XXX       | 16 Bit Timer/Counter Mode          |
| 1    | 00          | 01        | XXX       | 16 Bit Capture Mode                |
| 1    | 01          | 10        | XXX       | 16 Bit PPG Mode<br>(one-shot mode) |
| 1    | 01          | 11        | XXX       | 16 Bit PPG Mode<br>(repeat mode)   |

### 11.6.2 16-Bit Timer/Counter Mode

The 16-bit timer/counter mode is selected by control register as shown in Figure 11.14.

The 16-bit timer have counter and data register. The counter register is increased by internal or external clock input. Timer 1 can use the input clock with one of 1, 2, 4, 8, 64, 512 and 2048 prescaler division rates (T1CK[2:0]). When the value of T1CNTH, T1CNTL and the value of T1ADRH, T1ADRL are identical in Timer 1 respectively, a match signal is generated and the interrupt of Timer 1 occurs. The T1CNTH, T1CNTL value is automatically cleared by match signal. It can be also cleared by software (T1CC).

The external clock (EC1) counts up the timer at the rising edge. If the EC1 is selected as a clock source by T1CK[2:0], EC1 port should be set to the input port by P11IO bit.

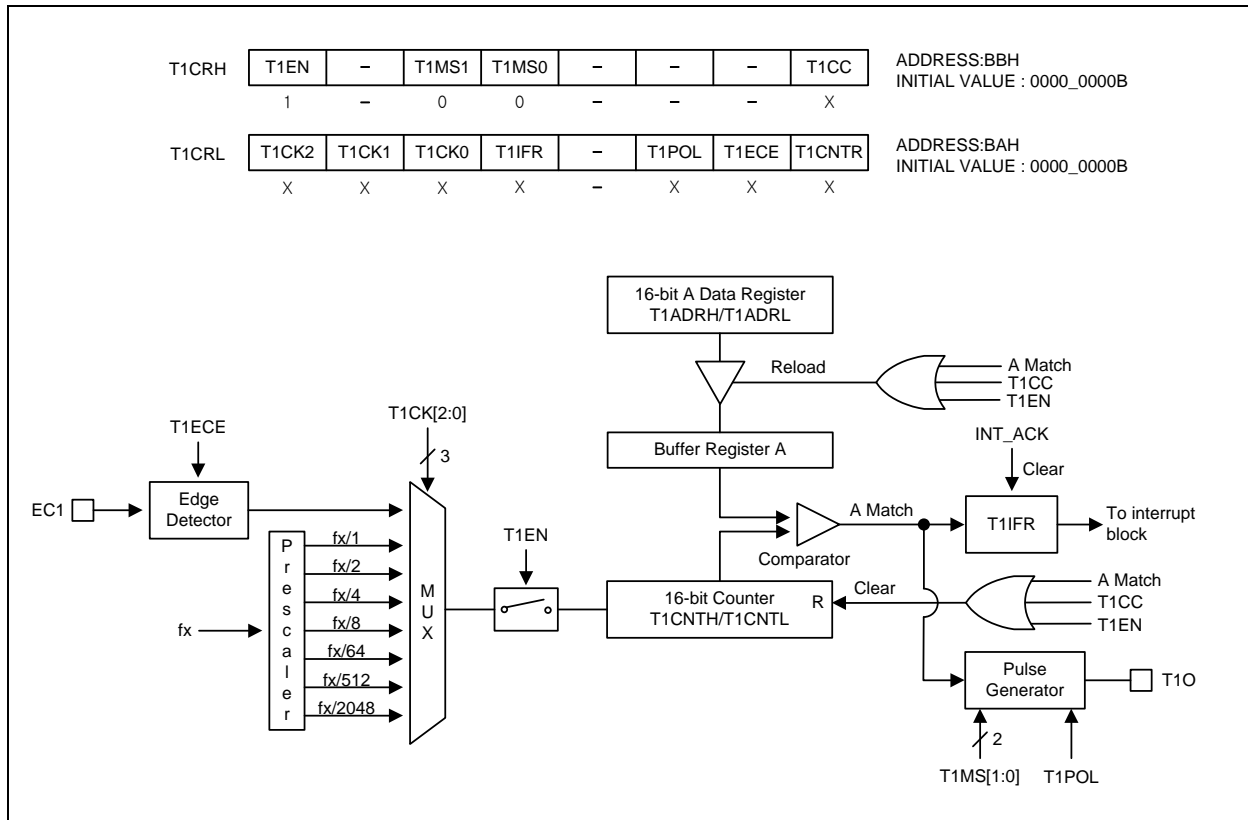


Figure 11.14 16-Bit Timer/Counter Mode for Timer 1

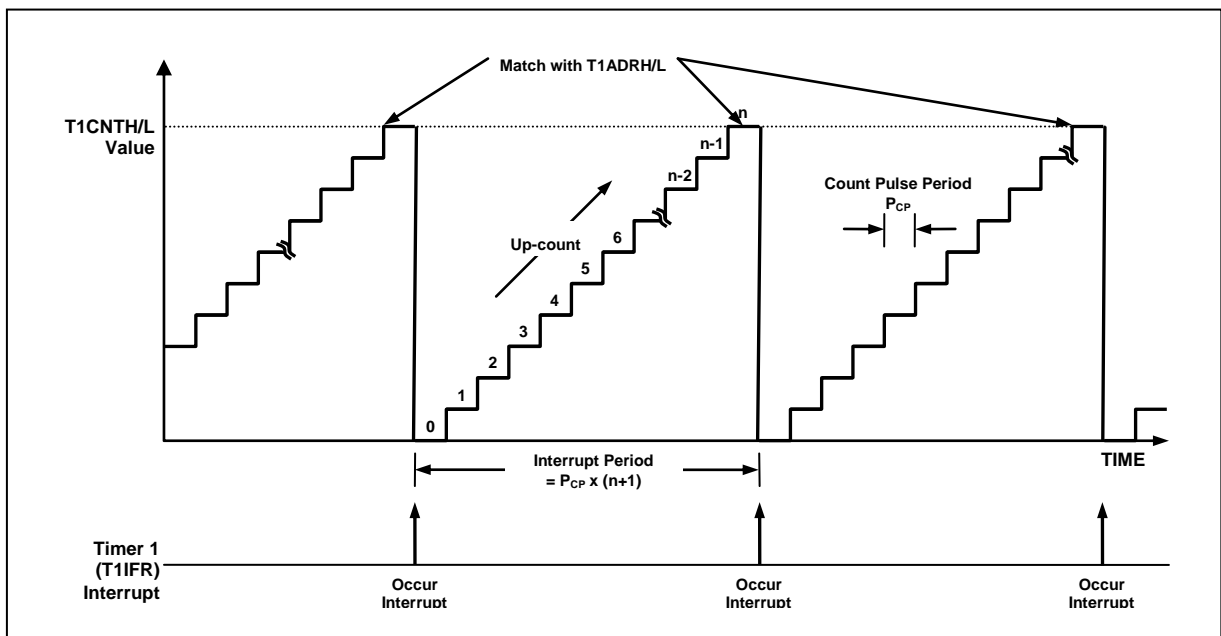


Figure 11.15 16-Bit Timer/Counter 1 Example

### 11.6.3 16-Bit Capture Mode

The 16-bit timer 1 capture mode is set by T1MS[1:0] as '01'. The clock source can use the internal/external clock. Basically, it has the same function as the 16-bit timer/counter mode and the interrupt occurs when T1CNTH/T1CNTL is equal to T1ADRH/T1ADRL. The T1CNTH, T1CNTL values are automatically cleared by match signal. It can be also cleared by software (T1CC).

This timer interrupt in capture mode is very useful when the pulse width of captured signal is wider than the maximum period of timer.

The capture result is loaded into T1BDRH/T1BDRL.

According to EIPOL1 registers setting, the external interrupt EINT11 function is chosen. Of course, the EINT11 pin must be set as an input port.

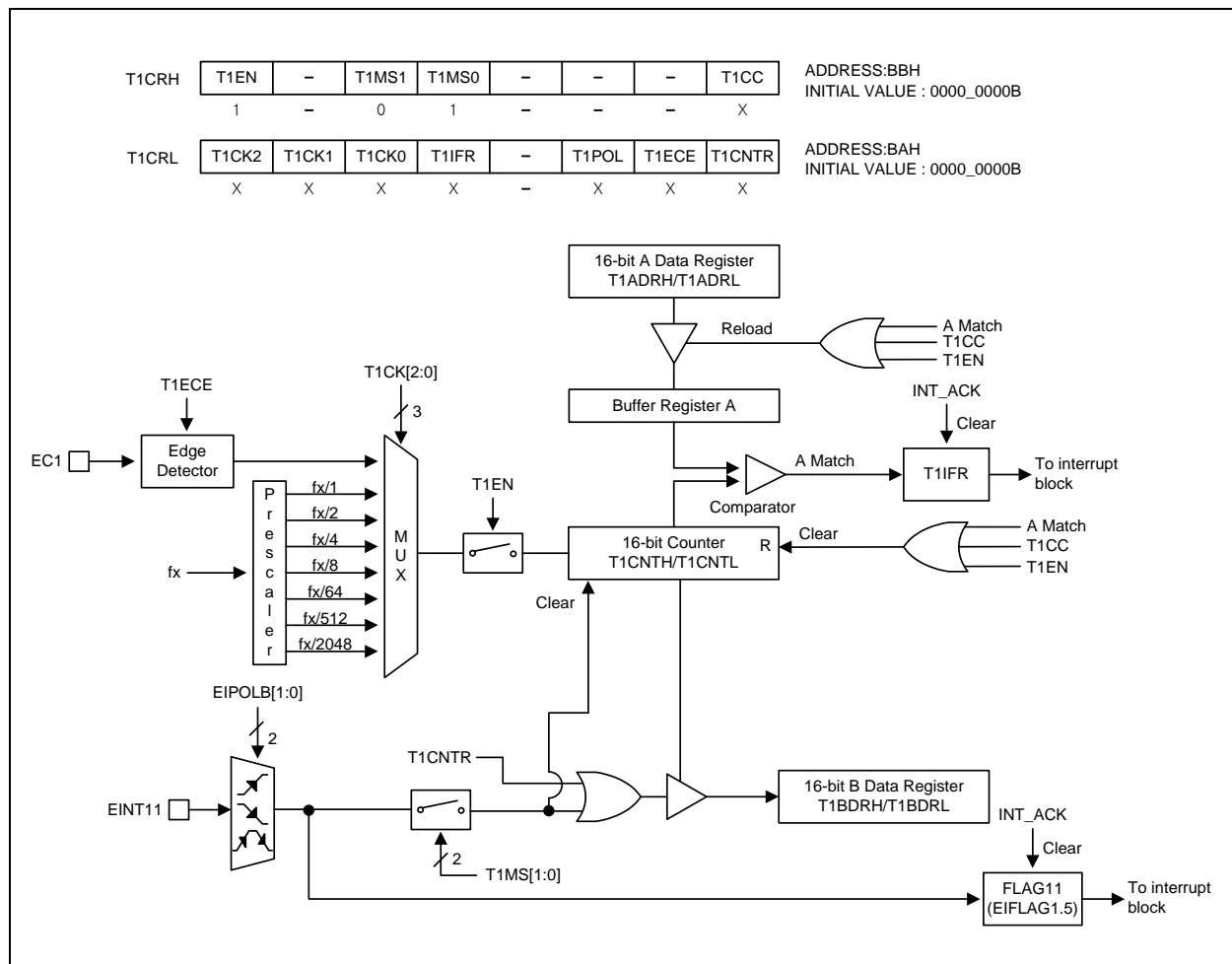


Figure 11.16 16-Bit Capture Mode for Timer 1

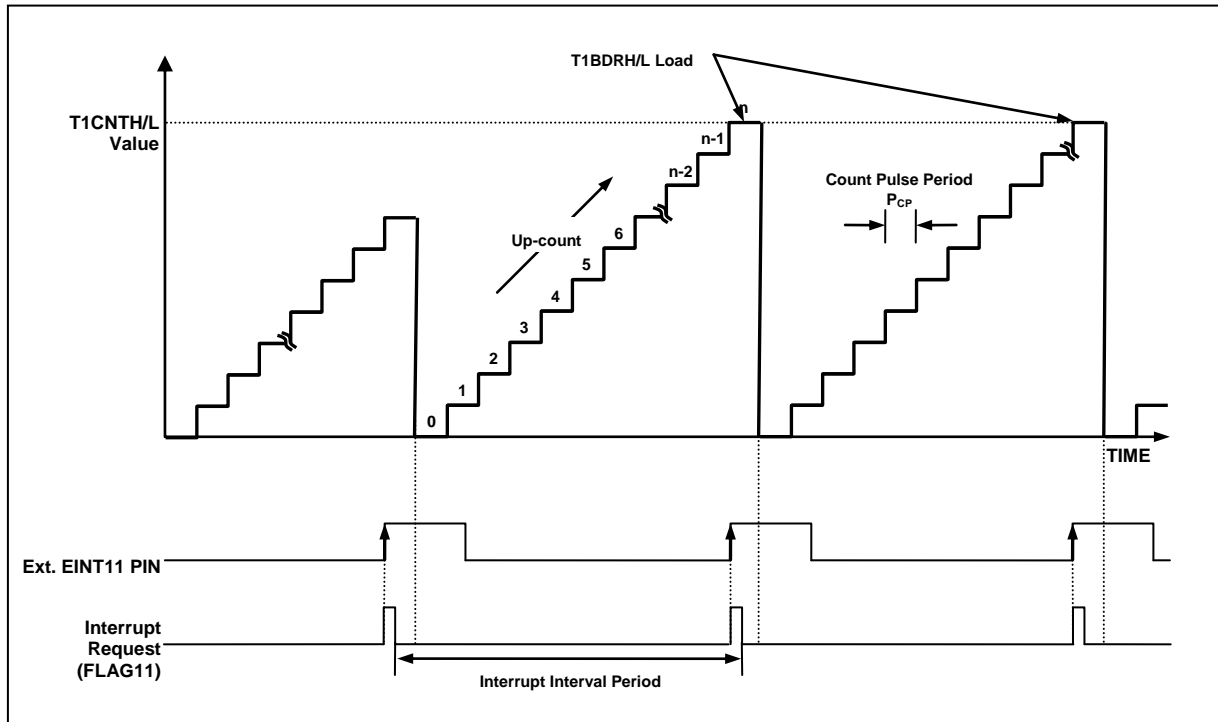


Figure 11.17 Input Capture Mode Operation for Timer 1

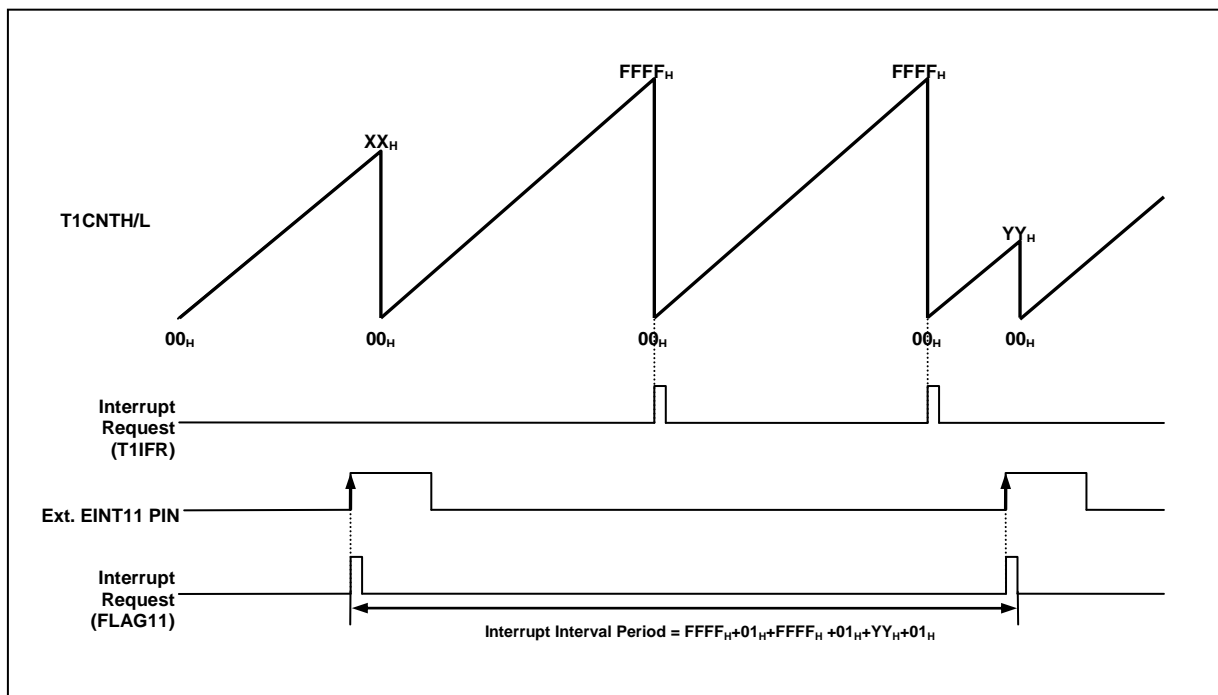


Figure 11.18 Express Timer Overflow in Capture Mode

### 11.6.4 16-Bit PPG Mode

The timer 1 has a PPG (Programmable Pulse Generation) function. In PPG mode, T1O/PWM1O pin outputs up to 16-bit resolution PWM output. This pin should be configured as a PWM output by setting P1FSRL[4:3] to '01'. The period of the PWM output is determined by the T1ADRH/T1ADRL. And the duty of the PWM output is determined by the T1BDRH/T1BDRL.

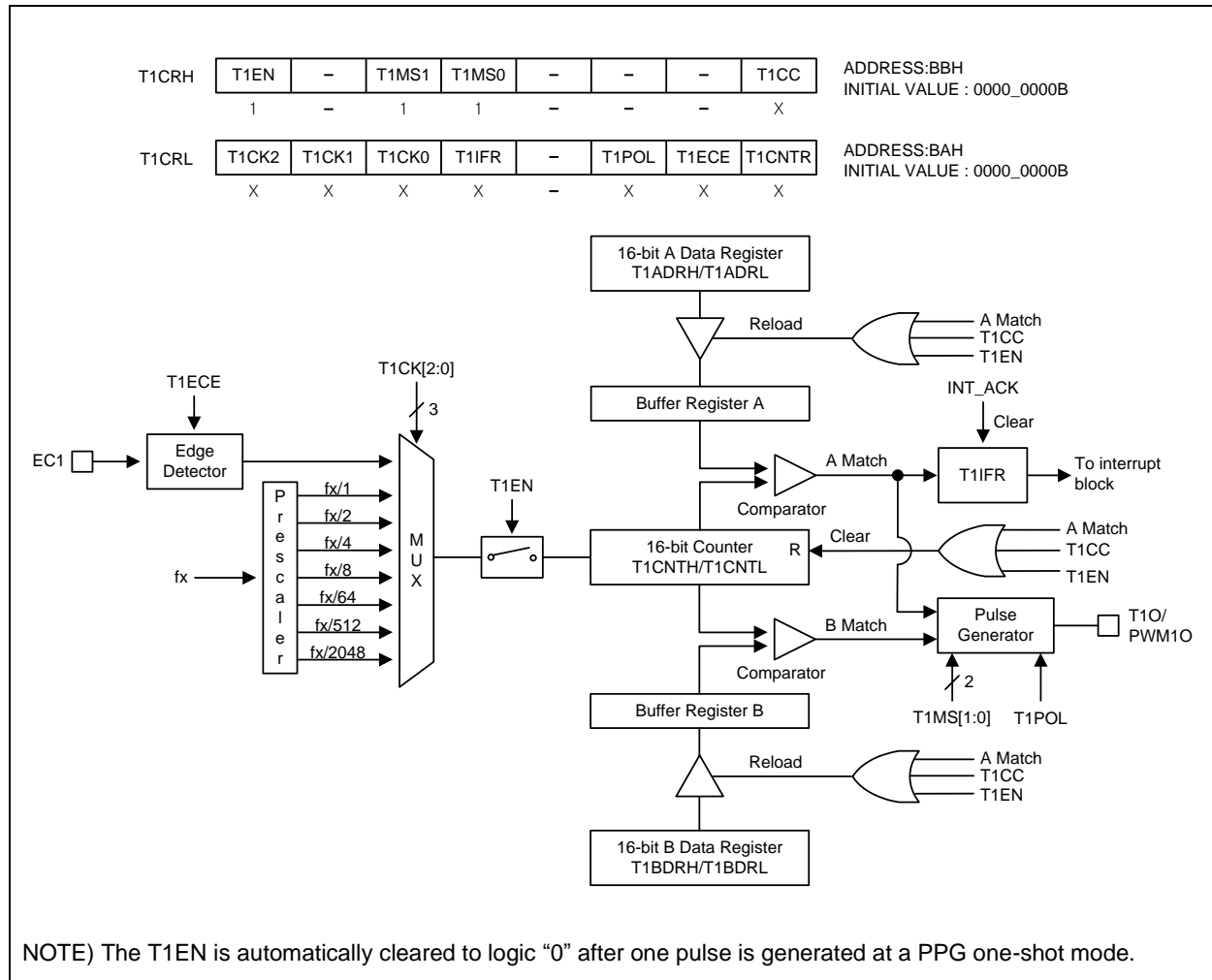


Figure 11.19 16-Bit PPG Mode for Timer 1

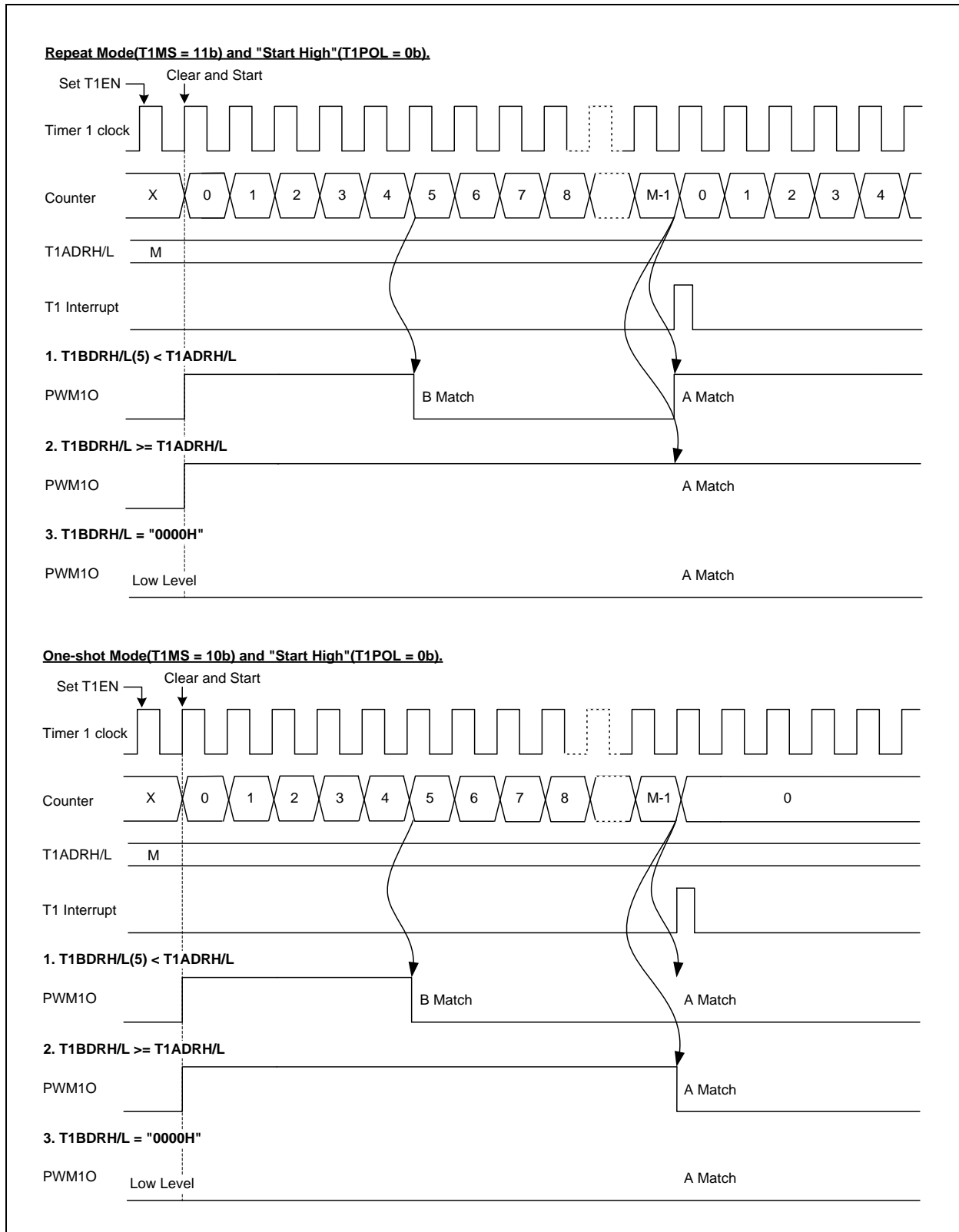


Figure 11.20 16-Bit PPG Mode Timming chart for Timer 1

### 11.6.5 Block Diagram

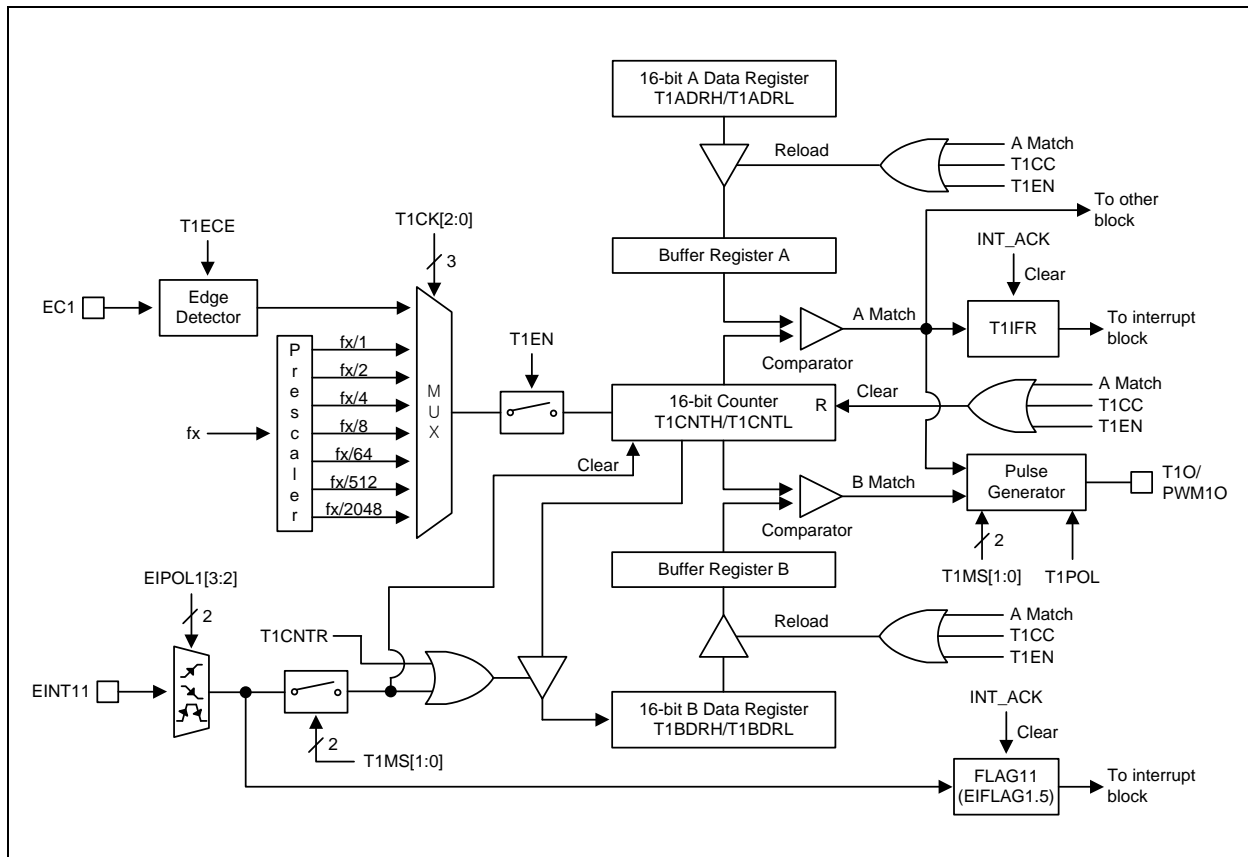


Figure 11.21 16-Bit Timer 1 Block Diagram

### 11.6.6 Register Map

Table 11-8 Timer 2 Register Map

| Name   | Address | Dir | Default | Description                   |
|--------|---------|-----|---------|-------------------------------|
| T1ADRH | BDH     | R/W | FFH     | Timer 1 A Data High Register  |
| T1ADRL | BCH     | R/W | FFH     | Timer 1 A Data Low Register   |
| T1BDRH | BFH     | R/W | FFH     | Timer 1 B Data High Register  |
| T1BDRL | BEH     | R/W | FFH     | Timer 1 B Data Low Register   |
| T1CRH  | BBH     | R/W | 00H     | Timer 1 Control High Register |
| T1CRL  | BAH     | R/W | 00H     | Timer 1 Control Low Register  |



### 11.6.6.1 Timer/Counter 1 Register Description

The timer/counter 1 register consists of timer 1 A data high register (T1ADRH), timer 1 A data low register (T1ADRL), timer 1 B data high register (T1BDRH), timer 1 B data low register (T1BDRL), timer 1 control high register (T1CRH) and timer 1 control low register (T1CRL).

### 11.6.6.2 Register Description for Timer/Counter 1

#### T1ADRH (Timer 1 A data High Register) : BDH

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| T1ADRH7 | T1ADRH6 | T1ADRH5 | T1ADRH4 | T1ADRH3 | T1ADRH2 | T1ADRH1 | T1ADRH0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

T1ADRH[7:0] T1 A Data High Byte

#### T1ADRL (Timer 1 A Data Low Register) : BCH

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| T1ADRL7 | T1ADRL6 | T1ADRL5 | T1ADRL4 | T1ADRL3 | T1ADRL2 | T1ADRL1 | T1ADRL0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

T1ADRL[7:0] T1 A Data Low Byte

NOTE) Do not write "0000H" in the T1ADRH/T1ADRL register when PPG mode

#### T1BDRH (Timer 1 B Data High Register) : BFH

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| T1BDRH7 | T1BDRH6 | T1BDRH5 | T1BDRH4 | T1BDRH3 | T1BDRH2 | T1BDRH1 | T1BDRH0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

T1BDRH[7:0] T1 B Data High Byte

#### T1BDRL (Timer 1 B Data Low Register) : BEH

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| T1BDRL7 | T1BDRL6 | T1BDRL5 | T1BDRL4 | T1BDRL3 | T1BDRL2 | T1BDRL1 | T1BDRL0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

T1BDRL[7:0] T1 B Data Low Byte

**T1CRH (Timer 1 Control High Register) : BBH**

| 7    | 6 | 5     | 4     | 3 | 2 | 1 | 0    |
|------|---|-------|-------|---|---|---|------|
| T1EN | – | T1MS1 | T1MS0 | – | – | – | T1CC |
| R/W  | – | R/W   | R/W   | – | – | – | R/W  |

Initial value : 00H

|           |                                                                                               |                                                |
|-----------|-----------------------------------------------------------------------------------------------|------------------------------------------------|
| T1EN      | Control Timer 1                                                                               |                                                |
| 0         | Timer 1 disable                                                                               |                                                |
| 1         | Timer 1 enable (Counter clear and start)                                                      |                                                |
| T1MS[1:0] | Control Timer 1 Operation Mode                                                                |                                                |
| T1MS1     | T1MS0                                                                                         | Description                                    |
| 0         | 0                                                                                             | Timer/counter mode (T1O: toggle at A match)    |
| 0         | 1                                                                                             | Capture mode (The A match interrupt can occur) |
| 1         | 0                                                                                             | PPG one-shot mode (PWM1O)                      |
| 1         | 1                                                                                             | PPG repeat mode (PWM1O)                        |
| T1CC      | Clear Timer 1 Counter                                                                         |                                                |
| 0         | No effect                                                                                     |                                                |
| 1         | Clear the Timer 1 counter (When write, automatically cleared "0" after being cleared counter) |                                                |

**T1CRL (Timer 1 Control Low Register) : BAH**

| 7     | 6     | 5     | 4     | 3 | 2     | 1     | 0      |
|-------|-------|-------|-------|---|-------|-------|--------|
| T1CK2 | T1CK1 | T1CK0 | T1IFR | – | T1POL | T1ECE | T1CNTR |
| R/W   | R/W   | R/W   | R/W   | – | R/W   | R/W   | R/W    |

Initial value : 00H

|           |                                                                                                                          |                                                                                                          |       |                      |
|-----------|--------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|-------|----------------------|
| T1CK[2:0] | Select Timer 1 clock source. fx is main system clock frequency                                                           |                                                                                                          |       |                      |
|           | T1CK2                                                                                                                    | T1CK1                                                                                                    | T1CK0 | Description          |
|           | 0                                                                                                                        | 0                                                                                                        | 0     | fx/2048              |
|           | 0                                                                                                                        | 0                                                                                                        | 1     | fx/512               |
|           | 0                                                                                                                        | 1                                                                                                        | 0     | fx/64                |
|           | 0                                                                                                                        | 1                                                                                                        | 1     | fx/8                 |
|           | 1                                                                                                                        | 0                                                                                                        | 0     | fx/4                 |
|           | 1                                                                                                                        | 0                                                                                                        | 1     | fx/2                 |
|           | 1                                                                                                                        | 1                                                                                                        | 0     | fx/1                 |
|           | 1                                                                                                                        | 1                                                                                                        | 1     | External clock (EC1) |
| T1IFR     | When T1 Interrupt occurs, this bit becomes '1'. For clearing bit, write '0' to this bit or auto clear by INT_ACK signal. |                                                                                                          |       |                      |
|           | 0                                                                                                                        | T1 Interrupt no generation                                                                               |       |                      |
|           | 1                                                                                                                        | T1 Interrupt generation                                                                                  |       |                      |
| T1POL     | T1O/PWM1O Polarity Selection                                                                                             |                                                                                                          |       |                      |
|           | 0                                                                                                                        | Start High (T1O/PWM1O is low level at disable)                                                           |       |                      |
|           | 1                                                                                                                        | Start Low (T1O/PWM1O is high level at disable)                                                           |       |                      |
| T1ECE     | Timer 1 External Clock Edge Selection                                                                                    |                                                                                                          |       |                      |
|           | 0                                                                                                                        | External clock falling edge                                                                              |       |                      |
|           | 1                                                                                                                        | External clock rising edge                                                                               |       |                      |
| T1CNTR    | Timer 1 Counter Read Control                                                                                             |                                                                                                          |       |                      |
|           | 0                                                                                                                        | No effect                                                                                                |       |                      |
|           | 1                                                                                                                        | Load the counter value to the B data register (When write, automatically cleared "0" after being loaded) |       |                      |

## 11.7 Timer 2

### 11.7.1.1 Overview

The 16-bit timer 2 consists of multiplexer, timer 2 A data high/low register, timer 2 B data high/low register and timer 2 control high/low register (T2ADRH, T2ADRL, T2BDRH, T2BDRL, T2CRH, T2CRL).

It has four operating modes:

- 16-bit timer/counter mode
- 16-bit capture mode
- 16-bit PPG output mode (one-shot mode)
- 16-bit PPG output mode (repeat mode)

The timer/counter 2 can be clocked by an internal or an external clock source (EC2) or T1 A Match (timer 1 A match signal). The clock source is selected by clock selection logic which is controlled by the clock selection bits (T2CK[2:0]).

- TIMER 2 clock source:  $f_x/1, 2, 4, 8, 64, 512, 2048$ , EC2 and T1 A Match

In the capture mode, by EINT12, the data is captured into input capture data register (T2BDRH/T2BDRL). In timer/counter mode, whenever counter value is equal to T2ADRH/L, T2O port toggles. Also the timer 2 outputs PWM wave form to PWM2O port in the PPG mode.

**Table 11-9 Timer 2 Operating Modes**

| T2EN | P1FSRL[6:5] | T2MS[1:0] | T2CK[2:0] | Timer 2                            |
|------|-------------|-----------|-----------|------------------------------------|
| 1    | 01          | 00        | XXX       | 16 Bit Timer/Counter Mode          |
| 1    | 00          | 01        | XXX       | 16 Bit Capture Mode                |
| 1    | 01          | 10        | XXX       | 16 Bit PPG Mode<br>(one-shot mode) |
| 1    | 01          | 11        | XXX       | 16 Bit PPG Mode<br>(repeat mode)   |

### 11.7.2 16-Bit Timer/Counter Mode

The 16-bit timer/counter mode is selected by control register as shown in Figure 11.22.

The 16-bit timer have counter and data register. The counter register is increased by internal or external clock(EC2) or timer 1 A match clock input. Timer 2 can use the input clock with one of 1, 2, 4, 8, 64, 512 and 2048 prescaler division rates, EC2 and T1 A Match (T2CK[2:0]). When the values of T2CNTH/T2CNTL and T2ADRH/T2ADRL are identical in timer 2, a match signal is generated and the interrupt of Timer 2 occurs. The T2CNTH/T2CNTL values are automatically cleared by match signal. It can be also cleared by software (T2CC).

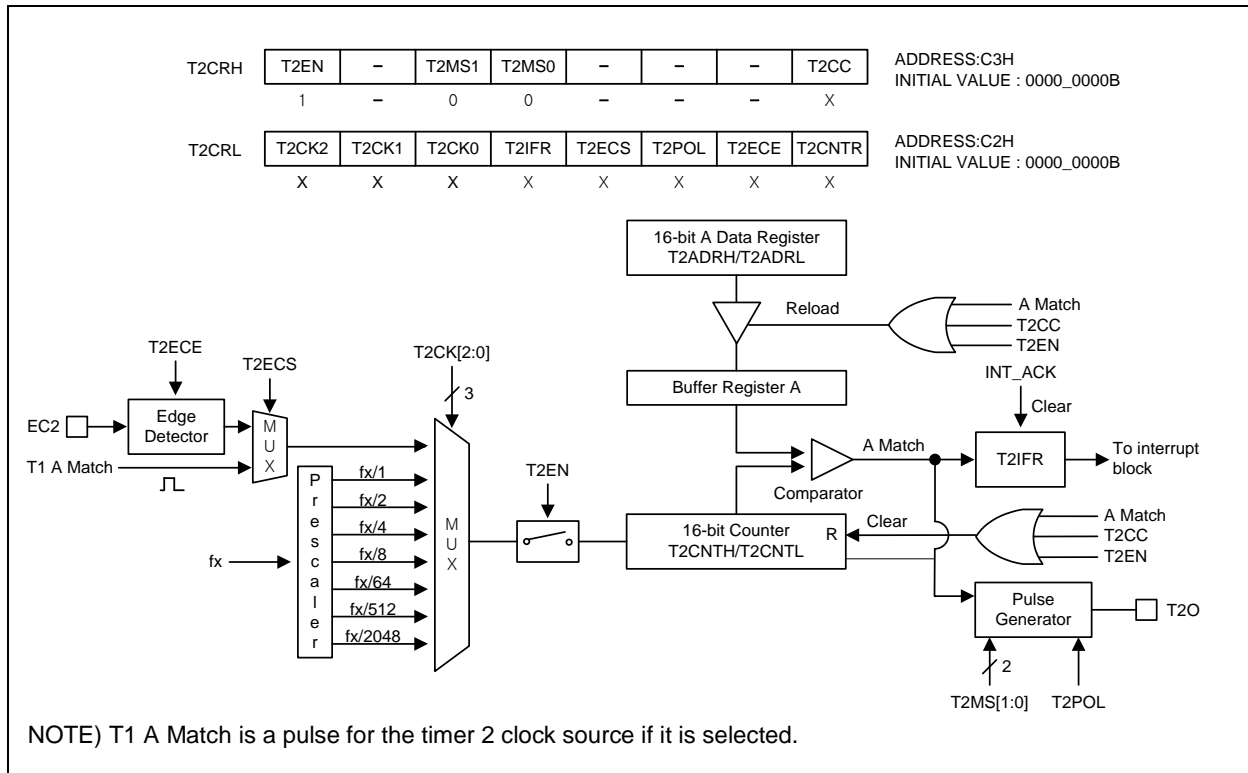


Figure 11.22 16-Bit Timer/Counter Mode for Timer 2

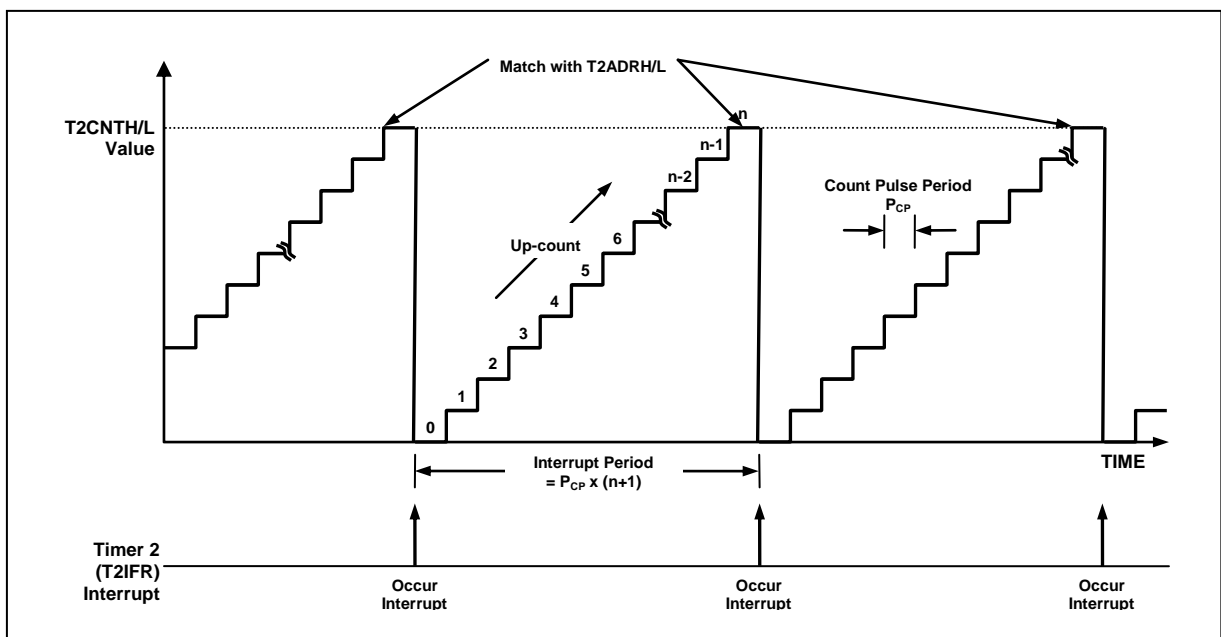


Figure 11.23 16-Bit Timer/Counter 2 Example

### 11.7.3 16-Bit Capture Mode

The timer 2 capture mode is set by T2MS[1:0] as '01'. The clock source can use the internal clock. Basically, it has the same function as the 16-bit timer/counter mode and the interrupt occurs when T2CNTH/T2CNTL is equal to T2ADRH/T2ADRL. T2CNTH/T2CNTL values are automatically cleared by match signal and it can be also cleared by software (T2CC).

This timer interrupt in capture mode is very useful when the pulse width of captured signal is wider than the maximum period of timer.

The capture result is loaded into T2BDRH/T2BDRL. In the timer 2 capture mode, timer 2 output(T2O) waveform is not available.

According to EIPOL1 registers setting, the external interrupt EINT12 function is chosen. Of course, the EINT12 pin must be set to an input port.

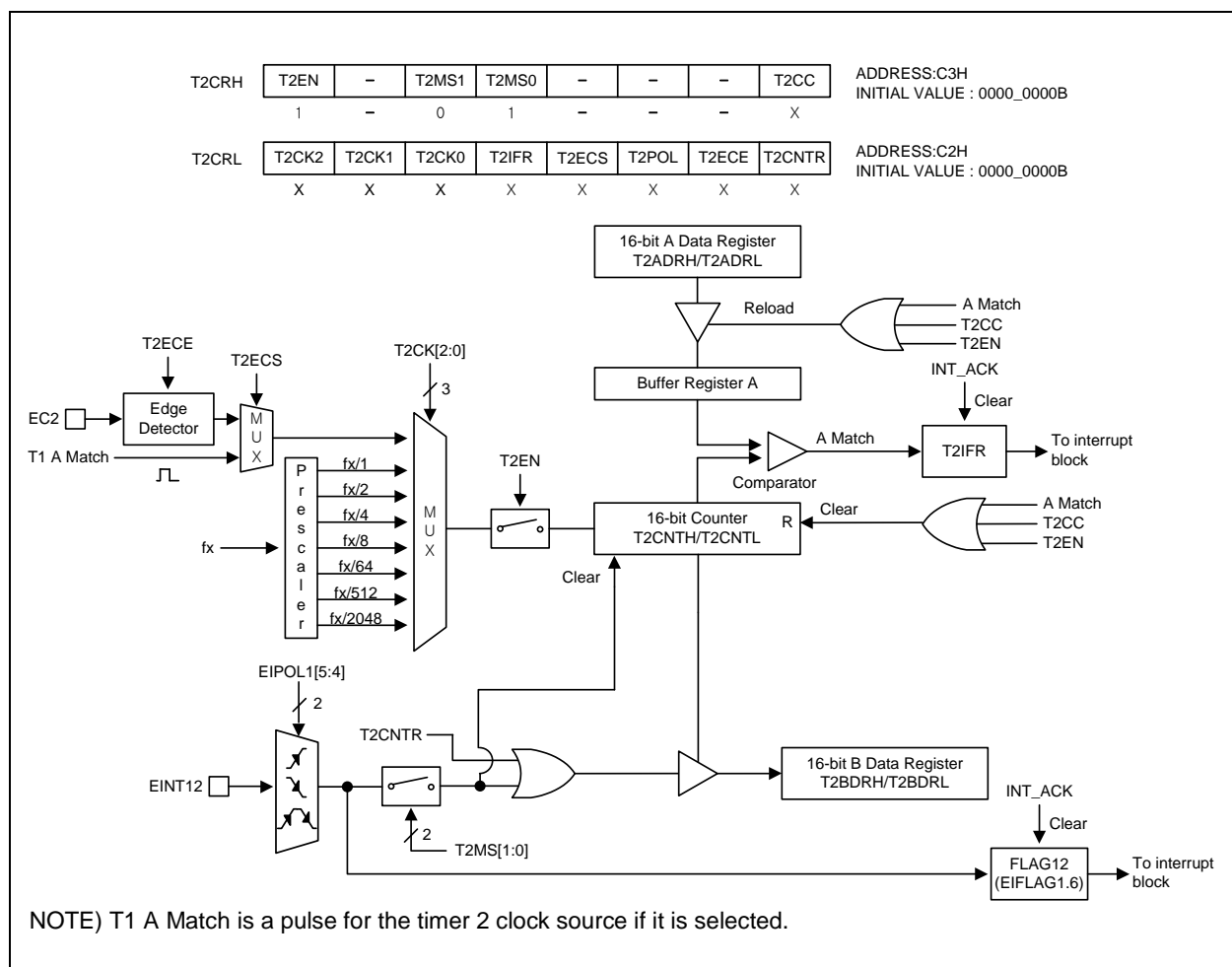


Figure 11.24 16-Bit Capture Mode for Timer 2

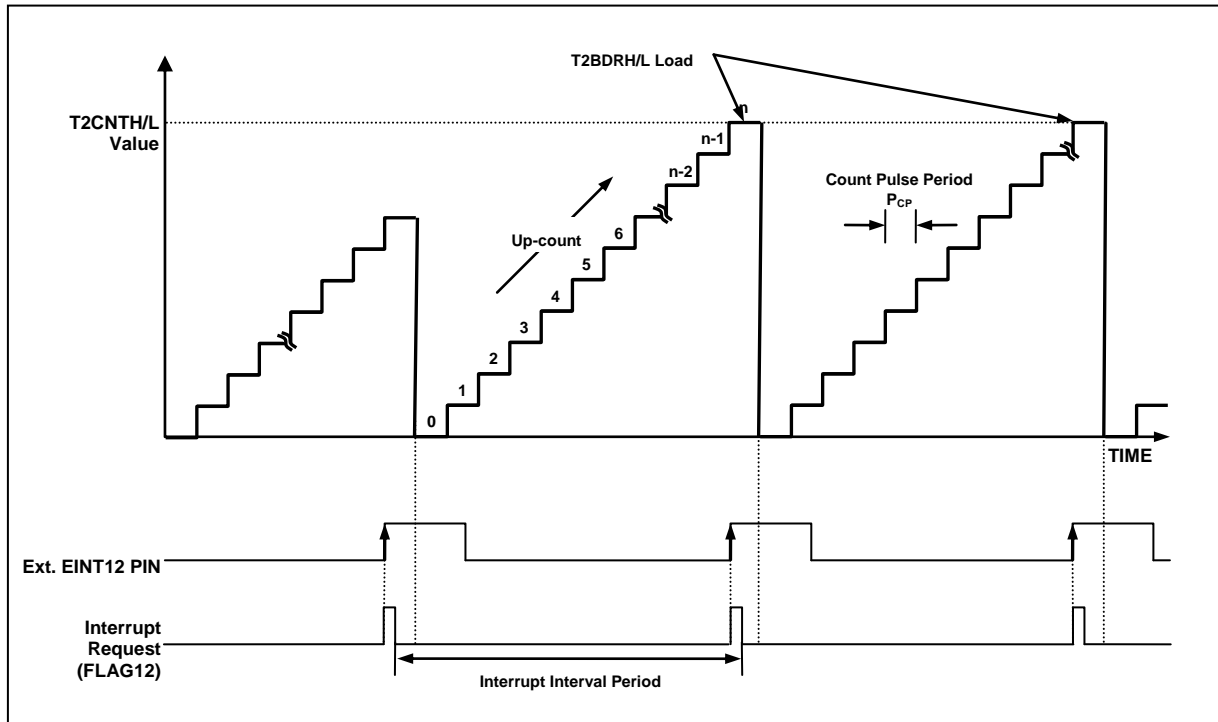


Figure 11.25 Input Capture Mode Operation for Timer 2

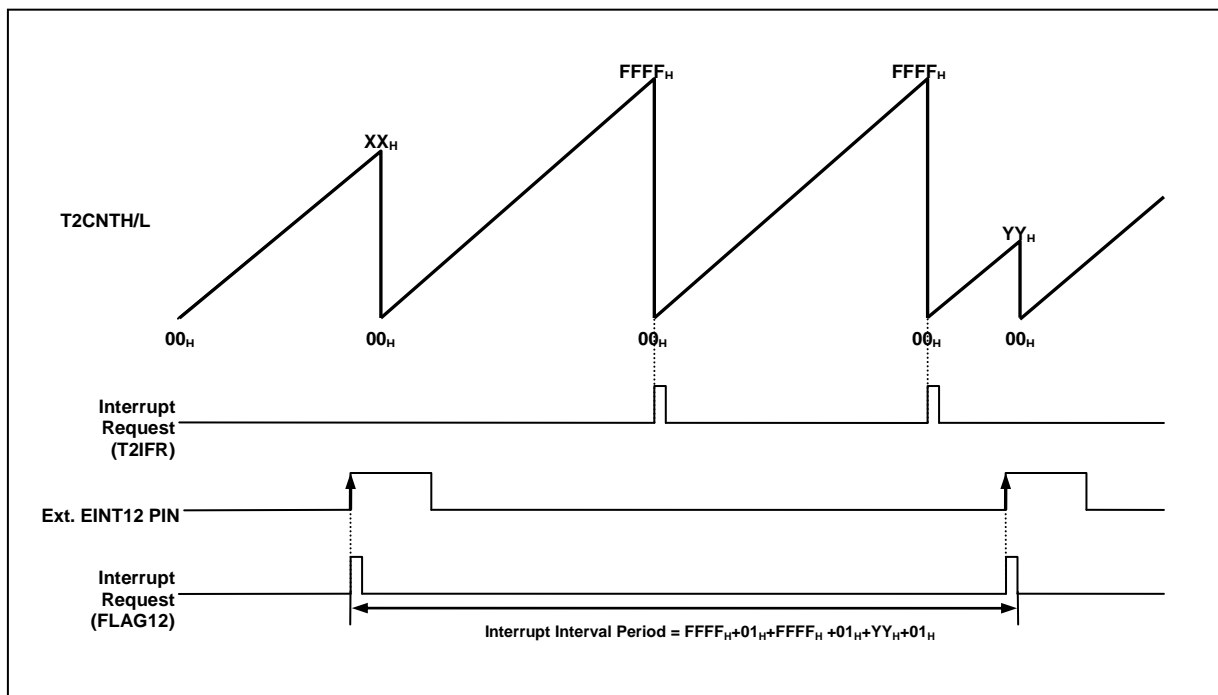
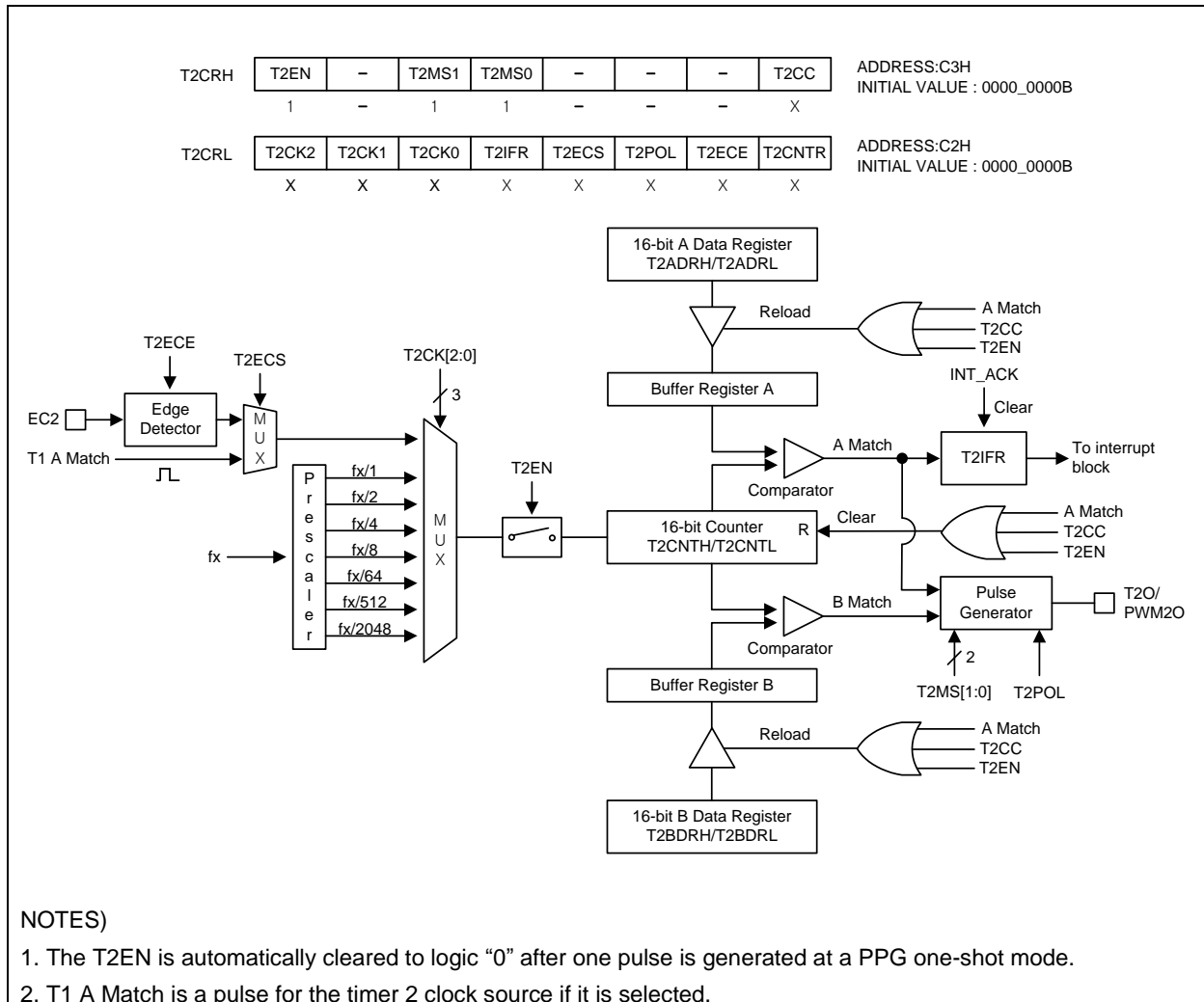


Figure 11.26 Express Timer Overflow in Capture Mode

### 11.7.4 16-Bit PPG Mode

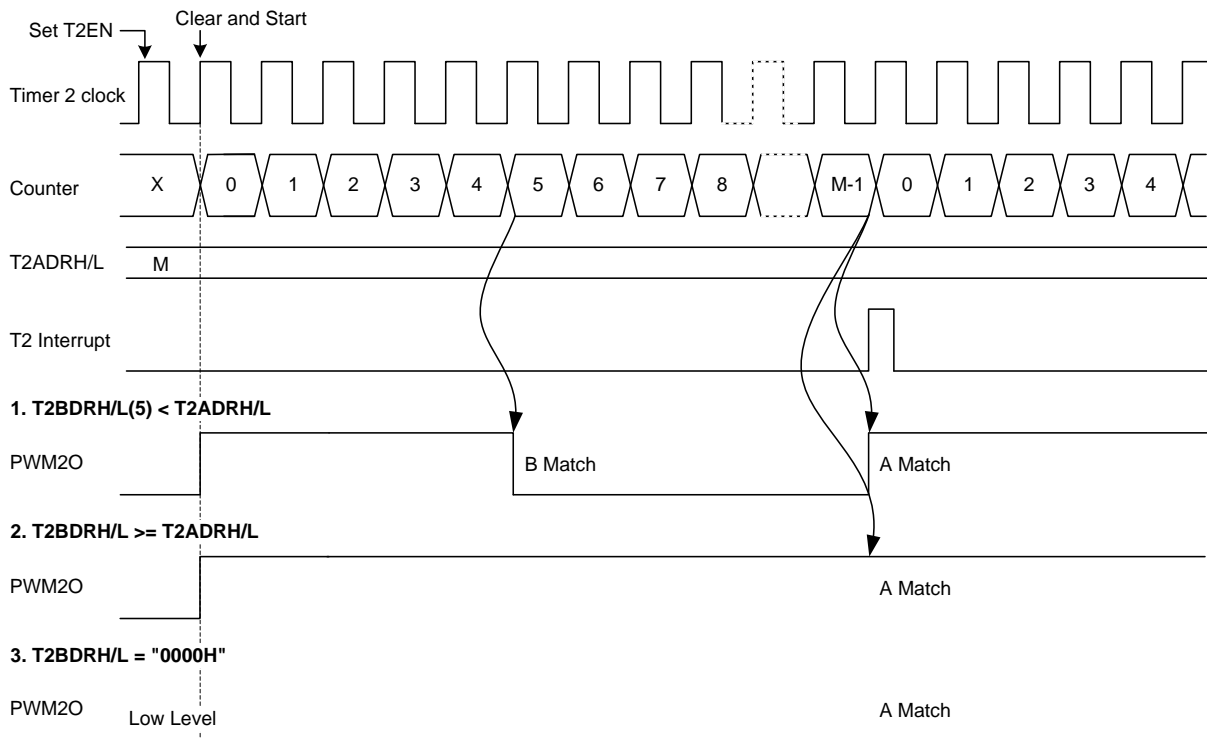
The timer 2 has a PPG (Programmable Pulse Generation) function. In PPG mode, the T2O/PWM2O pin outputs up to 16-bit resolution PWM output. This pin should be configured as a PWM output by set P1FSRL[6:5] to '01'. The period of the PWM output is determined by the T2ADRH/T2ADRL. And the duty of the PWM output is determined by the T2BDRH/T2BDRL.



**Figure 11.27 16-Bit PPG Mode for Timer 2**



**Repeat Mode(T2MS = 11b) and "Start High"(T2POL = 0b).**



**One-shot Mode(T2MS = 10b) and "Start High"(T2POL = 0b).**

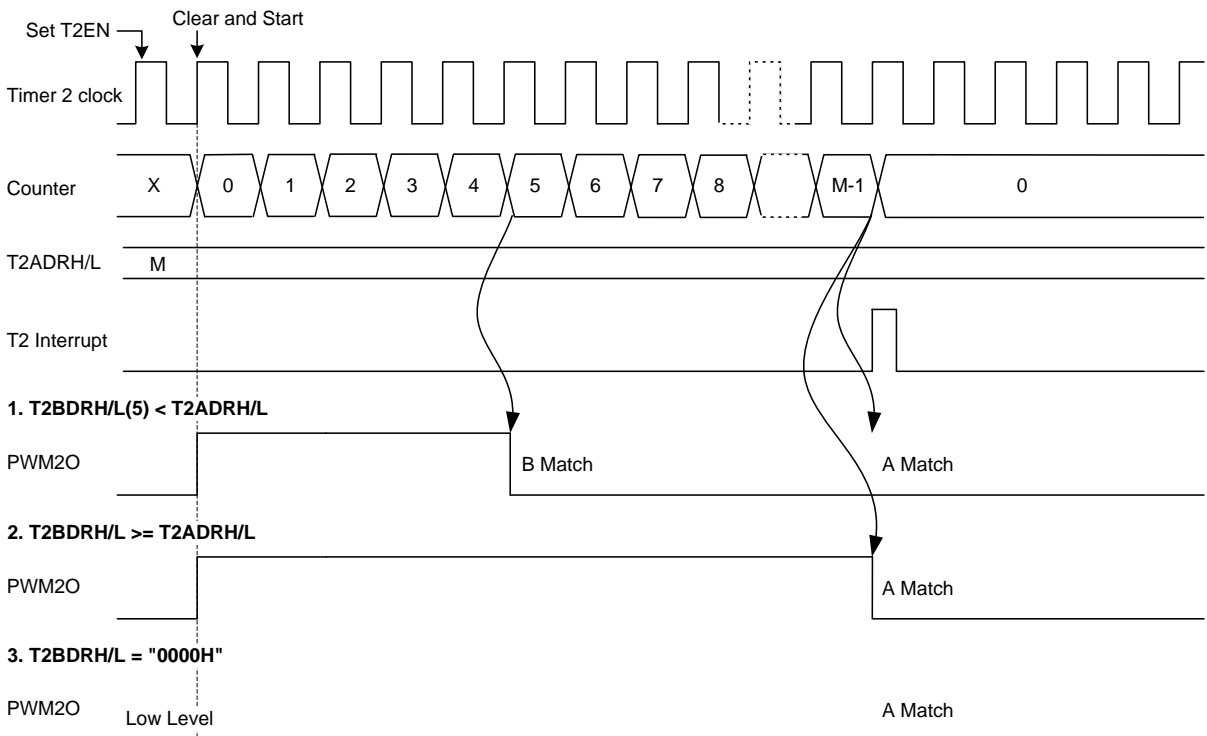


Figure 11.28 16-Bit PPG Mode Timing chart for Timer 2

### 11.7.5 Block Diagram

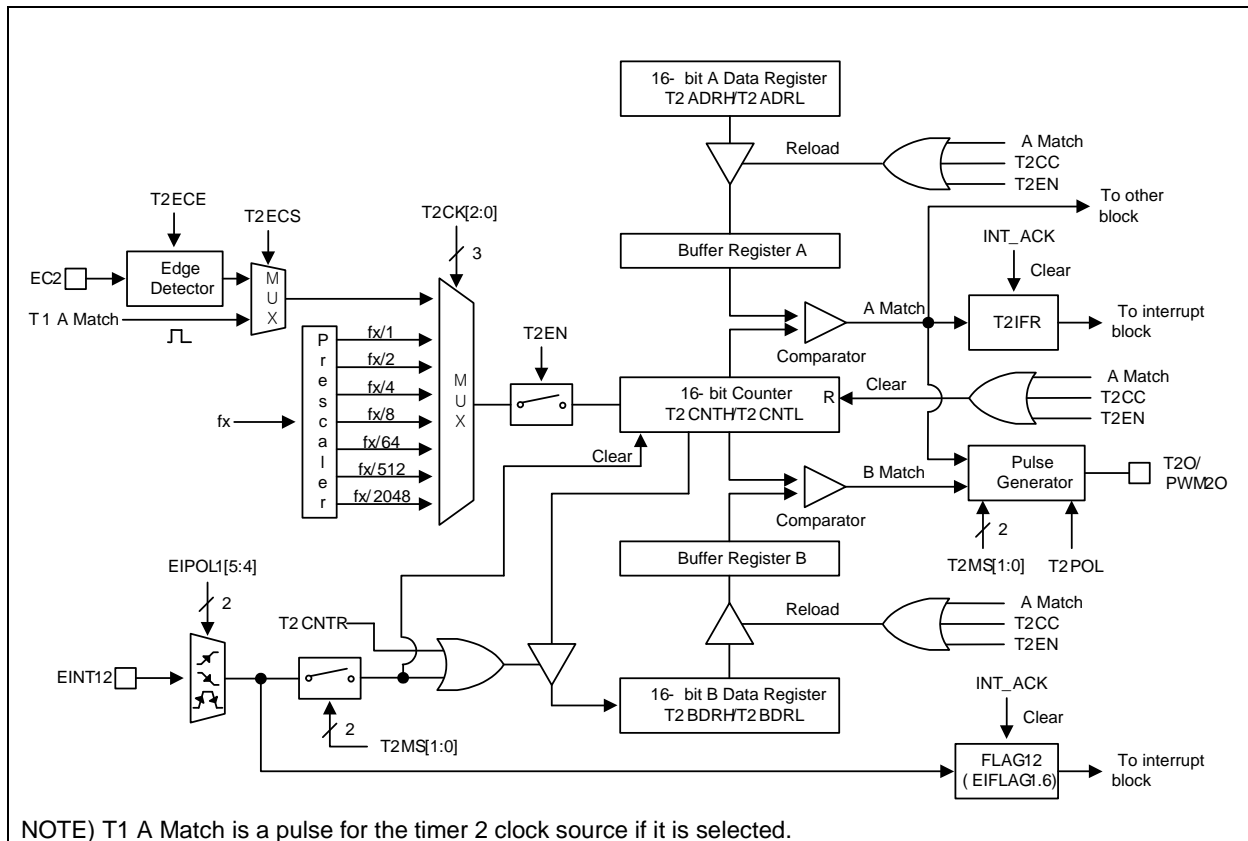


Figure 11.29 16-Bit Timer 2 Block Diagram

### 11.7.6 Register Map

Table 11-10 Timer 3 Register Map

| Name   | Address | Dir | Default | Description                   |
|--------|---------|-----|---------|-------------------------------|
| T2ADRH | C5H     | R/W | FFH     | Timer 2 A Data High Register  |
| T2ADRL | C4H     | R/W | FFH     | Timer 2 A Data Low Register   |
| T2BDRH | C7H     | R/W | FFH     | Timer 2 B Data High Register  |
| T2BDRL | C6H     | R/W | FFH     | Timer 2 B Data Low Register   |
| T2CRH  | C3H     | R/W | 00H     | Timer 2 Control High Register |
| T2CRL  | C2H     | R/W | 00H     | Timer 2 Control Low Register  |

### 11.7.6.1 Timer/Counter 2 Register Description

The timer/counter 2 register consists of timer 2 A data high register (T2ADRH), timer 2 A data low register (T2ADRL), timer 2 B data high register (T2BDRH), timer 2 B data low register (T2BDRL), timer 2 control high register (T2CRH) and timer 2 control low register (T2CRL).

### 11.7.6.2 Register Description for Timer/Counter 2

#### T2ADRH (Timer 2 A data High Register) : C5H

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| T2ADRH7 | T2ADRH6 | T2ADRH5 | T2ADRH4 | T2ADRH3 | T2ADRH2 | T2ADRH1 | T2ADRH0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

T2ADRH[7:0] T2 A Data High Byte

#### T2ADRL (Timer 2 A Data Low Register) : C4H

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| T2ADRL7 | T2ADRL6 | T2ADRL5 | T2ADRL4 | T2ADRL3 | T2ADRL2 | T2ADRL1 | T2ADRL0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

T2ADRL[7:0] T2 A Data Low Byte

NOTE) Do not write "0000H" in the T2ADRH/T2ADRL register when PPG mode.

#### T2BDRH (Timer 2 B Data High Register) : C7H

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| T2BDRH7 | T2BDRH6 | T2BDRH5 | T2BDRH4 | T2BDRH3 | T2BDRH2 | T2BDRH1 | T2BDRH0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

T2BDRH[7:0] T2 B Data High Byte

#### T2BDRL (Timer 2 B Data Low Register) : C6H

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| T2BDRL7 | T2BDRL6 | T2BDRL5 | T2BDRL4 | T2BDRL3 | T2BDRL2 | T2BDRL1 | T2BDRL0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

T2BDRL[7:0] T2 B Data Low

**T2CRH (Timer 2 Control High Register) : C3H**

| 7    | 6 | 5     | 4     | 3 | 2 | 1 | 0    |
|------|---|-------|-------|---|---|---|------|
| T2EN | – | T2MS1 | T2MS0 | – | – | – | T2CC |
| RW   | – | R/W   | RW    | – | – | – | RW   |

Initial value : 00H

|           |                                                                                               |                                                |
|-----------|-----------------------------------------------------------------------------------------------|------------------------------------------------|
| T2EN      | Control Timer 2                                                                               |                                                |
| 0         | Timer 2 disable                                                                               |                                                |
| 1         | Timer 2 enable (Counter clear and start)                                                      |                                                |
| T2MS[1:0] | Control Timer 2 Operation Mode                                                                |                                                |
| T2MS1     | T2MS0                                                                                         | Description                                    |
| 0         | 0                                                                                             | Timer/counter mode (T2O: toggle at A match)    |
| 0         | 1                                                                                             | Capture mode (The A match interrupt can occur) |
| 1         | 0                                                                                             | PPG one-shot mode (PWM2O)                      |
| 1         | 1                                                                                             | PPG repeat mode (PWM2O)                        |
| T2CC      | Clear Timer 2 Counter                                                                         |                                                |
| 0         | No effect                                                                                     |                                                |
| 1         | Clear the Timer 2 counter (When write, automatically cleared "0" after being cleared counter) |                                                |

**T2CRL (Timer 2 Control Low Register) : C2H**

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0      |
|-------|-------|-------|-------|-------|-------|-------|--------|
| T2CK2 | T2CK1 | T2CK0 | T2IFR | T2ECS | T2POL | T2ECE | T2CNTR |
| R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W    |

Initial value : 00H

|           |                                                                                                                                |                                                                                                          |       |                             |
|-----------|--------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|-------|-----------------------------|
| T2CK[2:0] | Select Timer 2 clock source. fx is main system clock frequency                                                                 |                                                                                                          |       |                             |
|           | T2CK2                                                                                                                          | T2CK1                                                                                                    | T2CK0 | Description                 |
|           | 0                                                                                                                              | 0                                                                                                        | 0     | fx/2048                     |
|           | 0                                                                                                                              | 0                                                                                                        | 1     | fx/512                      |
|           | 0                                                                                                                              | 1                                                                                                        | 0     | fx/64                       |
|           | 0                                                                                                                              | 1                                                                                                        | 1     | fx/8                        |
|           | 1                                                                                                                              | 0                                                                                                        | 0     | fx/4                        |
|           | 1                                                                                                                              | 0                                                                                                        | 1     | fx/2                        |
|           | 1                                                                                                                              | 1                                                                                                        | 0     | fx/1                        |
|           | 1                                                                                                                              | 1                                                                                                        | 1     | Selected clock by T2ECS bit |
| T2IFR     | When T2 Match Interrupt occurs, this bit becomes '1'. For clearing bit, write '0' to this bit or auto clear by INT_ACK signal. |                                                                                                          |       |                             |
|           | 0                                                                                                                              | T2 interrupt no generation                                                                               |       |                             |
|           | 1                                                                                                                              | T2 interrupt generation                                                                                  |       |                             |
| T2ECS     | Timer 2 External Clock Selection                                                                                               |                                                                                                          |       |                             |
|           | 0                                                                                                                              | Select external clock(EC2)                                                                               |       |                             |
|           | 1                                                                                                                              | Select Timer 1 A match                                                                                   |       |                             |
| T2POL     | T2O/PWM2O Polarity Selection                                                                                                   |                                                                                                          |       |                             |
|           | 0                                                                                                                              | Start High (T2O/PWM2O is low level at disable)                                                           |       |                             |
|           | 1                                                                                                                              | Start Low (T2O/PWM2O is high level at disable)                                                           |       |                             |
| T2ECE     | Timer 2 External Clock Edge Selection                                                                                          |                                                                                                          |       |                             |
|           | 0                                                                                                                              | External clock falling edge                                                                              |       |                             |
|           | 1                                                                                                                              | External clock rising edge                                                                               |       |                             |
| T2CNTR    | Timer 2 Counter Read Control                                                                                                   |                                                                                                          |       |                             |
|           | 0                                                                                                                              | No effect                                                                                                |       |                             |
|           | 1                                                                                                                              | Load the counter value to the B data register (When write, automatically cleared "0" after being loaded) |       |                             |

## 11.8 Buzzer Driver

### 11.8.1 Overview

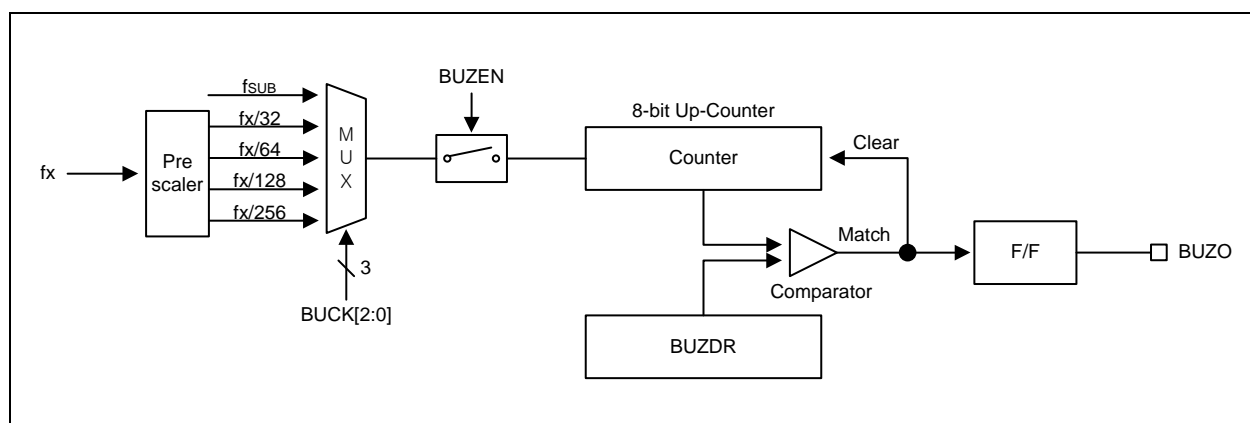
The Buzzer consists of 8 bit counter, buzzer data register (BUZDR), and buzzer control register (BUZCR). The Square Wave (61.035Hz~125.0 kHz @8MHz) is outputted through P11/BUZO pin. The buzzer data register (BUZDR) controls the bsuzzer frequency (look at the following expression). In buzzer control register (BUZCR), BUCK[2:0] selects source clock divided by prescaler.

$$f_{\text{BUZ}}(\text{Hz}) = \frac{\text{OscillatorFrequency}}{2 \times \text{PrescalerRatio} \times (\text{BUZDR} + 1)}$$

**Table 11-11 Buzzer Frequency at 8 MHz**

| BUZDR[7:0] | Buzzer Frequency (kHz) |                |                |                |
|------------|------------------------|----------------|----------------|----------------|
|            | BUZCR[3:1]=000         | BUZCR[3:1]=001 | BUZCR[3:1]=010 | BUZCR[3:1]=011 |
| 0000_0000  | 125kHz                 | 62.5kHz        | 31.25kHz       | 15.625kHz      |
| 0000_0001  | 62.5kHz                | 31.25kHz       | 15.625kHz      | 7.812kHz       |
| ...        | ...                    | ...            | ...            | ...            |
| 1111_1101  | 492.126Hz              | 246.063Hz      | 123.031Hz      | 61.515Hz       |
| 1111_1110  | 490.196Hz              | 245.098Hz      | 122.549Hz      | 61.274Hz       |
| 1111_1111  | 488.281Hz              | 244.141Hz      | 122.07Hz       | 61.035Hz       |

### 11.8.2 Block Diagram



**Figure 11.30 Buzzer Driver Block Diagram**

### 11.8.3 Register Map

Table 11-12 Buzzer Driver Register Map

| Name  | Address | Dir | Default | Description             |
|-------|---------|-----|---------|-------------------------|
| BUZDR | 8FH     | R/W | FFH     | Buzzer Data Register    |
| BUZCR | 97H     | R/W | 00H     | Buzzer Control Register |

### 11.8.4 Buzzer Driver Register Description

Buzzer driver consists of buzzer data register (BUZDR) and buzzer control register (BUZCR).

### 11.8.5 Register Description for Buzzer Driver

#### BUZDR (Buzzer Data Register) : 8FH

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| BUZDR7 | BUZDR6 | BUZDR5 | BUZDR4 | BUZDR3 | BUZDR2 | BUZDR1 | BUZDR0 |
| R/W    | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    |

Initial value : FFH

BUZDR[7:0] This bits control the Buzzer frequency  
Its resolution is 00H ~ FFH

#### BUZCR (Buzzer Control Register) : 97H

| 7 | 6 | 5 | 4 | 3     | 2     | 1     | 0     |
|---|---|---|---|-------|-------|-------|-------|
| – | – | – | – | BUCK2 | BUCK1 | BUCK0 | BUZEN |
| – | – | – | – | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

BUCK[2:0] Buzzer Driver Source Clock Selection

| BUCK2 | BUCK1 | BUCK0 | Description                         |
|-------|-------|-------|-------------------------------------|
| 0     | 0     | 0     | fx/32                               |
| 0     | 0     | 1     | fx/64                               |
| 0     | 1     | 0     | fx/128                              |
| 0     | 1     | 1     | fx/256                              |
| 1     | x     | x     | f <sub>SUB</sub> (External Sub OSC) |

BUZEN Buzzer Driver Operation Control

| BUZEN | Description           |
|-------|-----------------------|
| 0     | Buzzer Driver disable |
| 1     | Buzzer Driver enable  |

NOTE) fx: System clock oscillation frequency.

## 11.9 12-Bit A/D Converter

### 11.9.1 Overview

The analog-to-digital converter (A/D) allows conversion of an analog input signal to corresponding 12-bit digital value. The A/D module has fifteen analog inputs. The output of the multiplexer is the input into the converter which generates the result through successive approximation. The A/D module has four registers which are the A/D converter control high register (ADCCRH), A/D converter control low register (ADCCRL), A/D converter data high register (ADCDRH), and A/D converter data low register (ADCDDL). The channels to be converted are selected by setting ADSEL[3:0]. To execute A/D conversion, TRIG[1:0] bits should be set to 'xx'. The register ADCDRH and ADCDDL contains the results of the A/D conversion. When the conversion is completed, the result is loaded into the ADCDRH and ADCDDL, the A/D conversion status bit AFLAG is set to '1', and the A/D interrupt is set. During A/D conversion, AFLAG bit is read as '0'.

### 11.9.2 Conversion Timing

The A/D conversion process requires 4 steps (4 clock edges) to convert each bit and 10 clocks to set up A/D conversion. Therefore, total of 58 clocks are required to complete a 12-bit conversion: When fxx/8 is selected for conversion clock with a 12MHz fxx clock frequency, one clock cycle is 0.66  $\mu$ s. Each bit conversion requires 4 clocks, the conversion rate is calculated as follows:

4 clocks/bit  $\times$  12 bits + set-up time = 58 clocks,

58 clock  $\times$  0.66  $\mu$ s = 38.28  $\mu$ s at 1.5 MHz (12 MHz/8)

NOTE) The A/D converter needs at least 20  $\mu$ s for conversion time. So you must set the conversion time more than 20  $\mu$ s.



### 11.9.3 Block Diagram

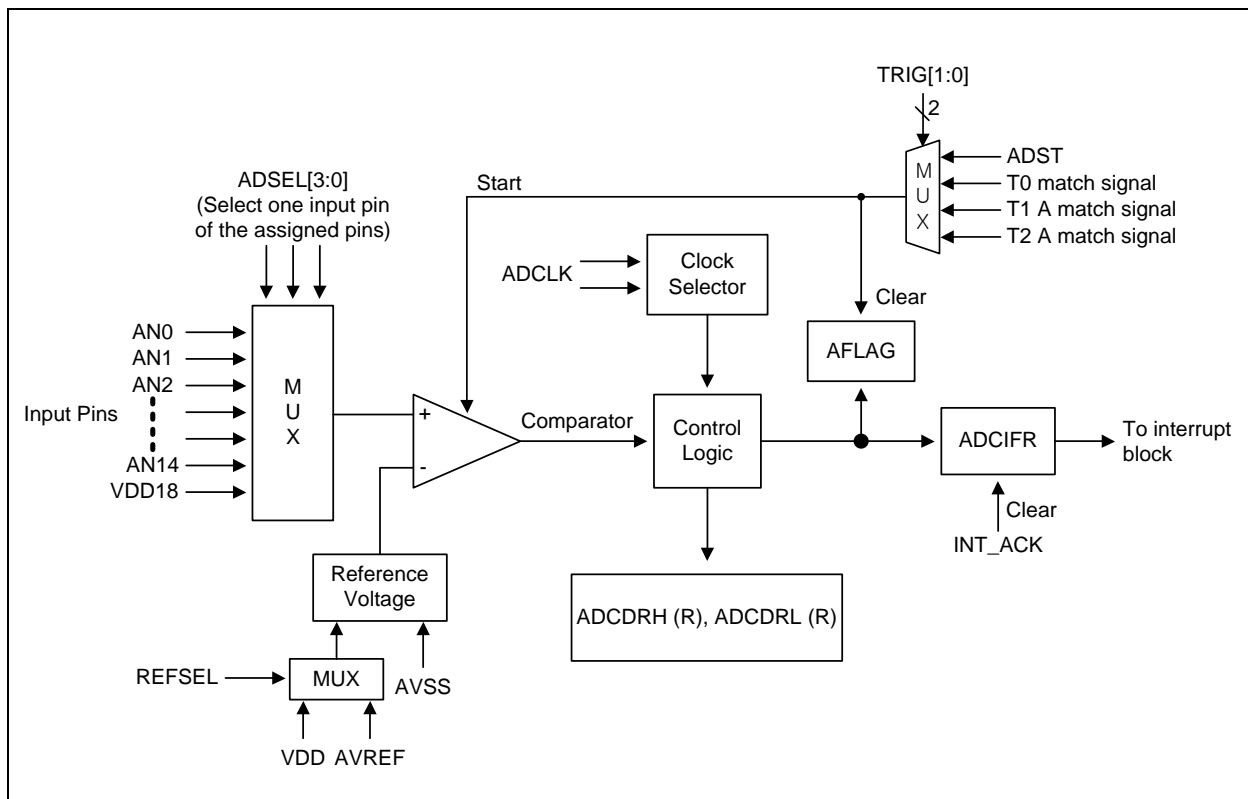


Figure 11.31 12-bit ADC Block Diagram

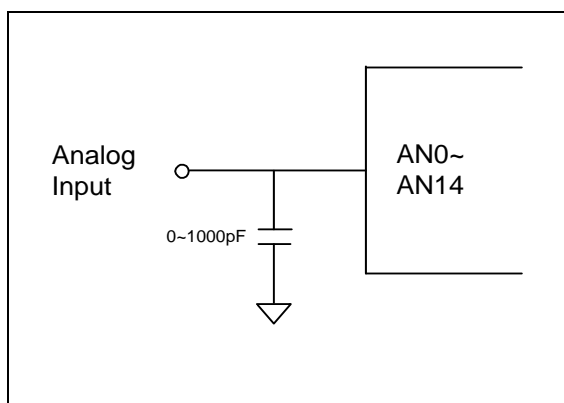


Figure 11.32 A/D Analog Input Pin with Capacitor

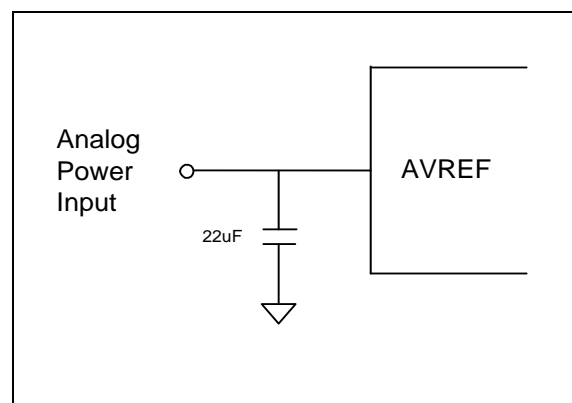
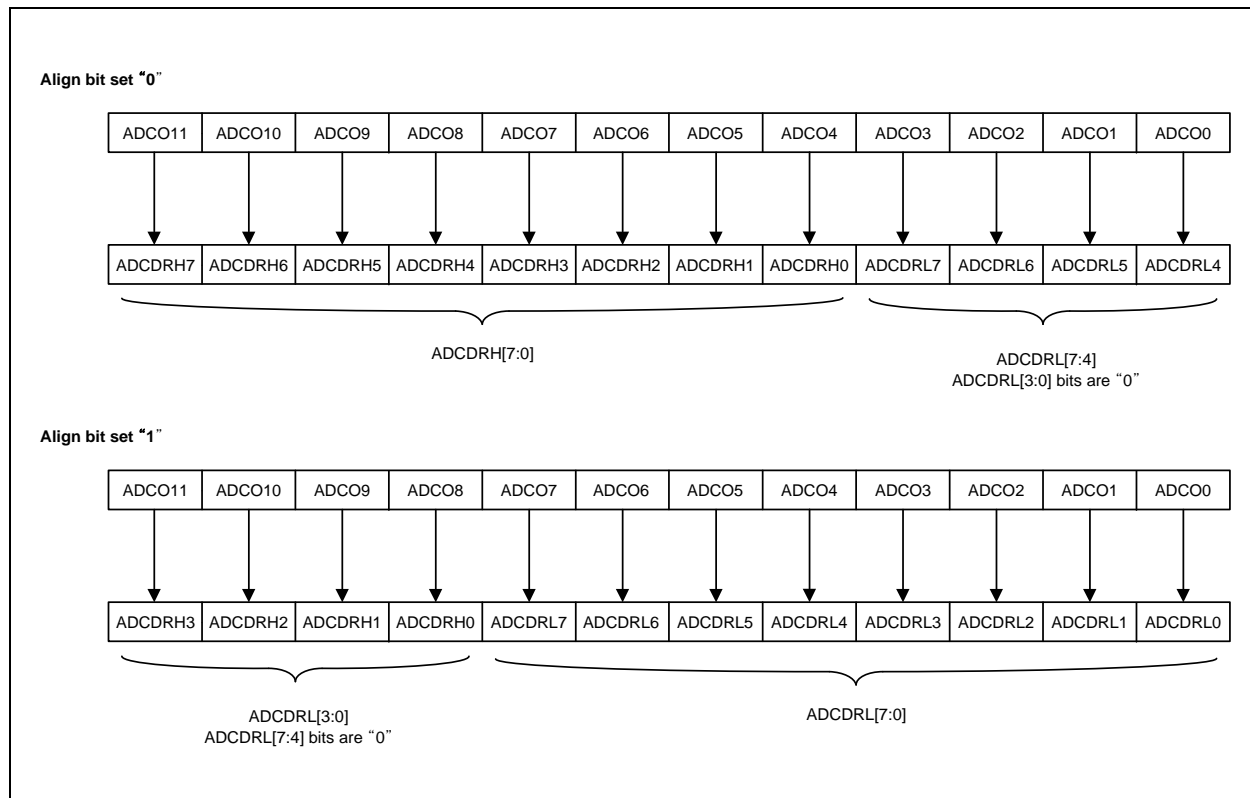


Figure 11.33 A/D Power (AVREF) Pin with Capacitor

### 11.9.4 ADC Operation



**Figure 11.34 ADC Operation for Align Bit**

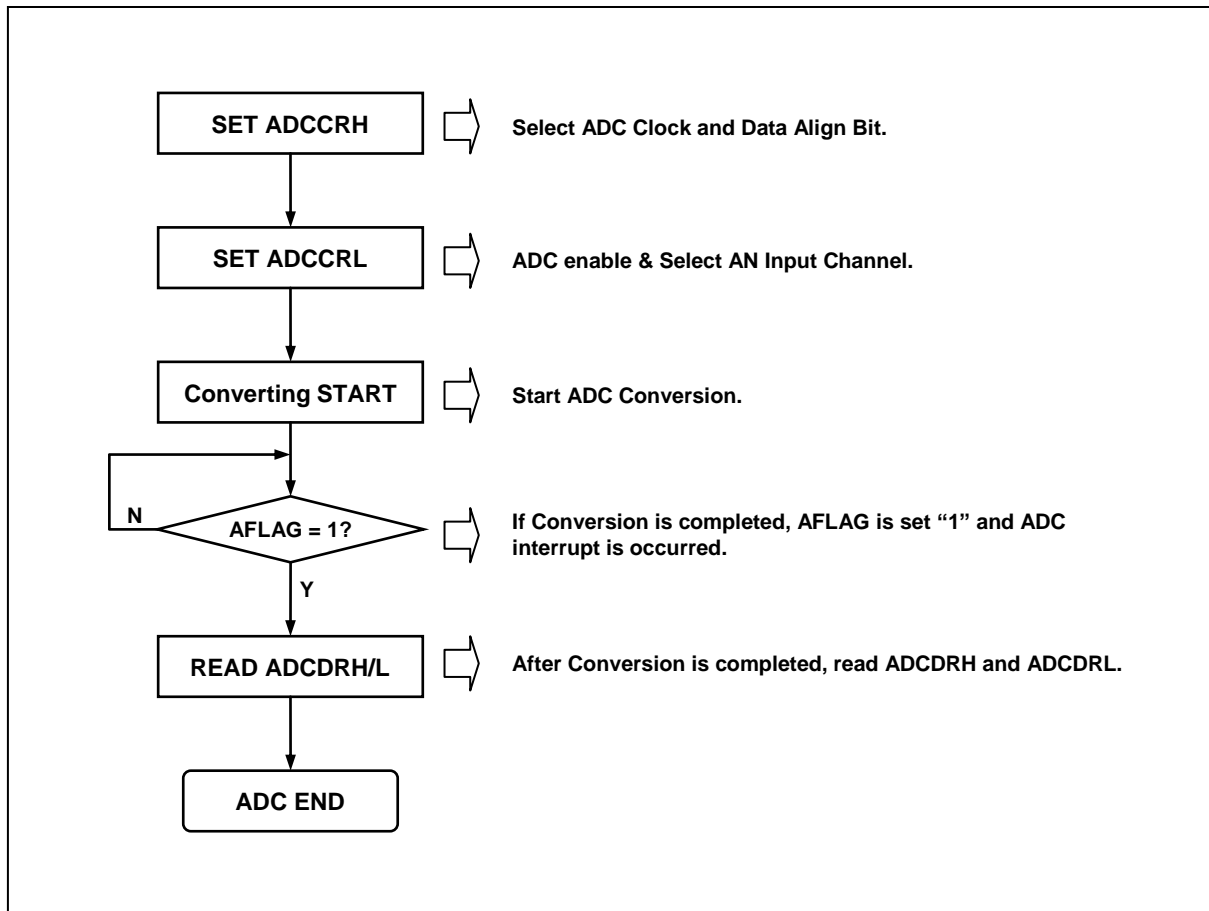
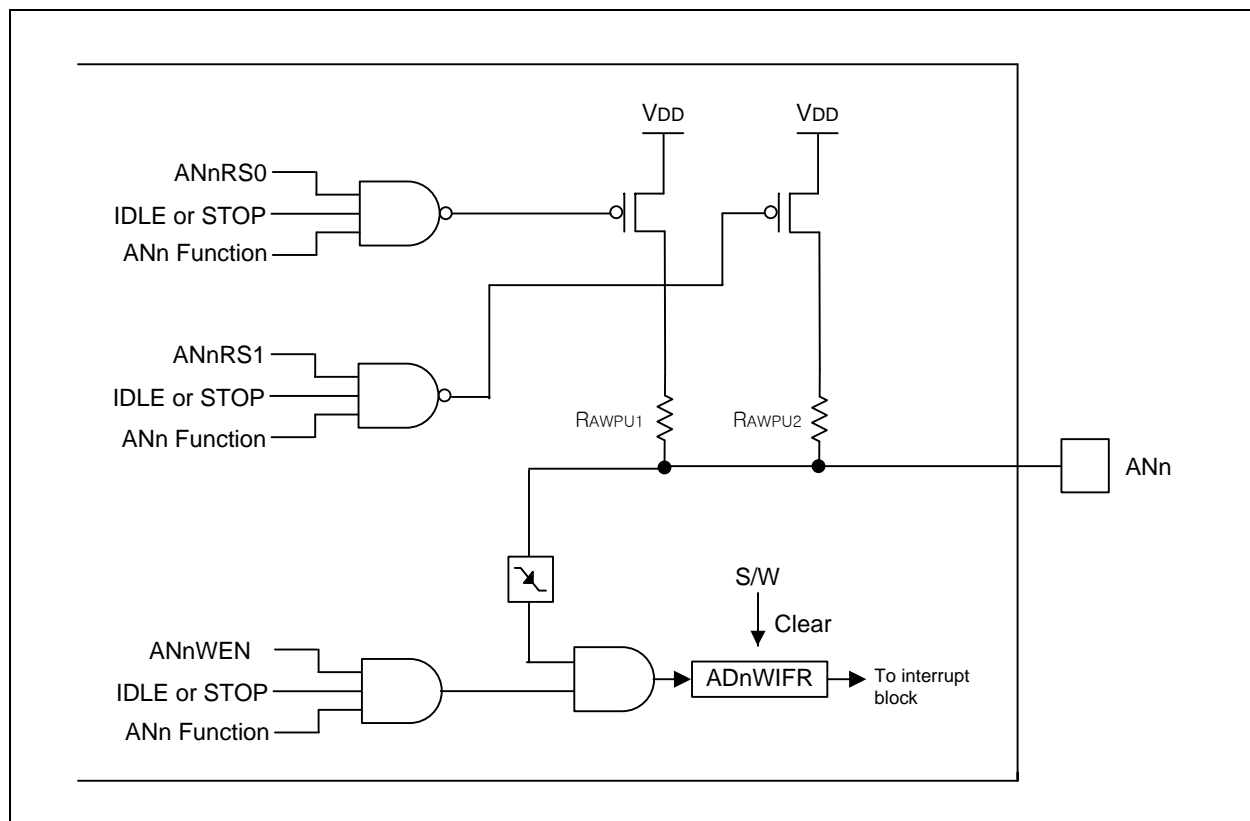


Figure 11.35 A/D Converter Operation Flow

### 11.9.5 ADC Power-down Wake-up Function

The A/D converter has ADC power-down wake-up function. The function includes two pull-up resistors for wake-up from power-down mode. The corresponding pull-up resistor which is selected as an ADC input function by P0FSR/P1FSRL/P1FSRH register is enabled during power-down mode (IDLE, STOP) if ANnRS[1:0] is not "00b". An ADC wake-up interrupt can occur by a falling edge (VIL) of key inputs during power-down mode (IDLE, STOP) if ANnWEN bit is '1'. Where n = 0, 1, 2, ..., and 14.

### 11.9.6 ADC Power-down Wake-up Function Block Diagram



**Figure 11.36 ADC Power-down Wake-up Function Block Diagram**

- NOTES)
1. AN0~AN14 Function can be controlled by P0FSR, P1FSRL, and P1FSRH.
  2. The pull-up resistor of P0/P1 can be enabled by P0PU/P1PU register. So, Be careful of each P0/P1 pull-up resistor. If a pull-up resistor of P0/P1 is enabled, the corresponding pin will be changed the equivalent resistor value by it.
  3. ADC wake-up interrupt can occur by a falling edge (VIL) of selected ANn pins.
  4. ADC path is off when it is power-down mode and ADC wake-up interrupt path is on during power-down mode when ANnWEN bit is '1'.
  5. Where n = 0, 1, 2, ..., and 14.

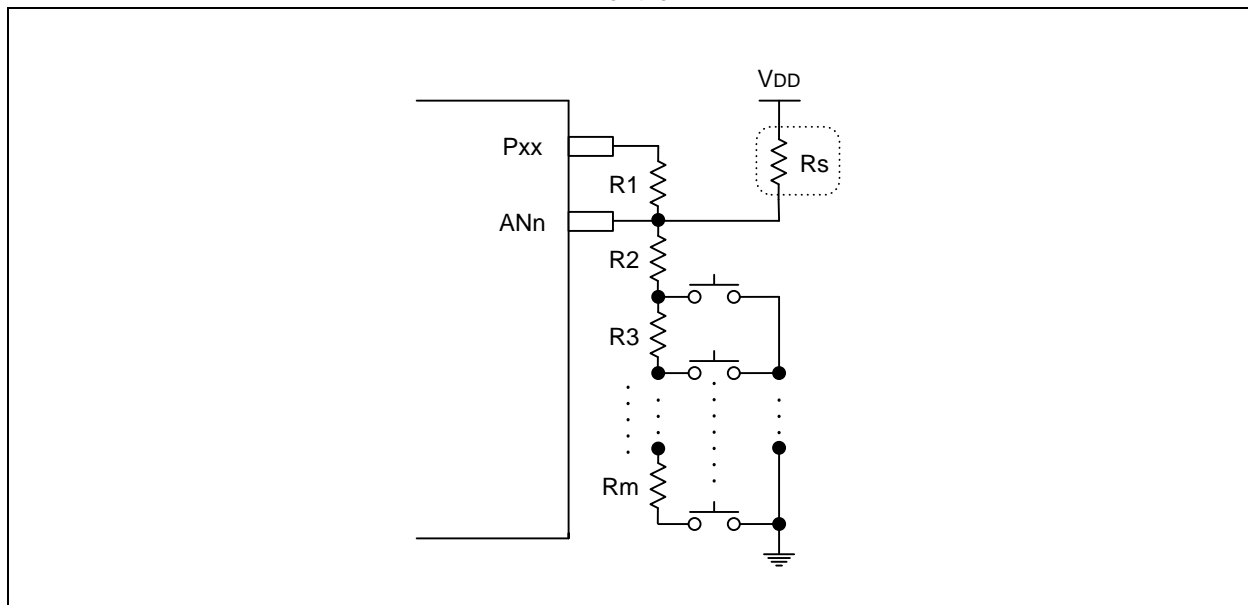
### 11.9.7 ADC Power-down Wake-up Function Operation

To use ADC power-down wake-up function in case of AN0 and a wake-up pull-up resistor 150kΩ, follow the recommended steps below.

1. Select AN0 function by P0FSR0 bit set to '1' in P0FSR register. Disable P00's pull-up resistor (P00PU).
2. Enable pull-up resistor 150kΩ by AN0RS[1:0] bits of ADWRCR0 register set to '01b' for AN0.
3. Enable ADC wake-up interrupt by AN0WEN bit of ADWCRL register set to '1' for AN0.
4. Enter the power-down(idle or stop) mode.
5. If it occurs the falling edge(VIL) of key input through AN0, an ADC wake-up interrupt will be requested and released from power-down mode.
6. At this time, ADC wake-up interrupt flag for AN0, AN0WIFR bit of ADWIFRL register becomes '1'. The flag is cleared only by writing a '0' to the bit. So, the flag should be cleared by software.

### 11.9.8 Application circuit for ADC Key Input by resistor string.

The resistor  $R_s$  needs to use the "ADC power-down wake-up function". The resistor prevents the ADC input pin from floating when the CPU goes into power-down mode. If  $R_s \gg R_1$ , the equivalent resistor of  $R_s/R_1$  depends on the resistor  $R_1$ . So, the resistor  $R_s$  had better use highly greater value than the  $R_1$  resistor.



**Figure 11.37 Application circuit for ADC Key Input by resistor string**

- NOTES)
1. The  $R_s$  about 1MΩ recommended.
  2. Pxx is normal I/O pin.
  3. Where  $n = 0, 1, 2, \dots, 14$ .

### 11.9.9 Register Map

**Table 11-13 ADC Register Map**

| Name    | Address | Dir | Default | Description                              |
|---------|---------|-----|---------|------------------------------------------|
| ADCDRH  | 9FH     | R   | xxH     | A/D Converter Data High Register         |
| ADCDDL  | 9EH     | R   | xxH     | A/D Converter Data Low Register          |
| ADCCRH  | 9DH     | R/W | 00H     | A/D Converter Control High Register      |
| ADCCRL  | 9CH     | R/W | 00H     | A/D Converter Control Low Register       |
| ADWRCR0 | F2H     | R/W | 00H     | ADC Wake-up Resistor Control Register 0  |
| ADWRCR1 | F3H     | R/W | 00H     | ADC Wake-up Resistor Control Register 1  |
| ADWRCR2 | F4H     | R/W | 00H     | ADC Wake-up Resistor Control Register 2  |
| ADWRCR3 | F5H     | R/W | 00H     | ADC Wake-up Resistor Control Register 3  |
| ADWCRH  | F7H     | R/W | 00H     | ADC Wake-up Control High Register        |
| ADWCRL  | F6H     | R/W | 00H     | ADC Wake-up Control Low Register         |
| ADWIFRH | DDH     | R/W | 00H     | ADC Wake-up Interrupt Flag High Register |
| ADWIFRL | DCH     | R/W | 00H     | ADC Wake-up Interrupt Flag Low Register  |

#### 11.9.10 ADC Register Description

The ADC register consists of A/D converter data high register (ADCDRH), A/D converter data low register (ADCDDL), A/D converter control high register (ADCCRH), A/D converter control low register (ADCCRL), ADC wake-up resistor control register 0 (ADWRCR0), ADC wake-up resistor control register 1 (ADWRCR1), ADC wake-up resistor control register 2 (ADWRCR2), ADC wake-up resistor control register 3 (ADWRCR3), ADC wake-up control high register (ADWCRH), ADC wake-up control low register (ADWCRL), ADC wake-up interrupt flag high register (ADWIFRH), and ADC wake-up interrupt flag low register (ADWIFRL).

### 11.9.11 Register Description for ADC

#### ADCDRH (A/D Converter Data High Register) : 9FH

| 7      | 6      | 5     | 4     | 3               | 2               | 1              | 0              |
|--------|--------|-------|-------|-----------------|-----------------|----------------|----------------|
| ADDM11 | ADDM10 | ADDM9 | ADDM8 | ADDM7<br>ADDL11 | ADDM6<br>ADDL10 | ADDM5<br>ADDL9 | ADDM4<br>ADDL8 |
| R      | R      | R     | R     | R               | R               | R              | R              |

Initial value : xxH

ADDM[11:4] MSB align, A/D Converter High Data (8-bit)

ADDL[11:8] LSB align, A/D Converter High Data (4-bit)

#### ADCDRL (A/D Converter Data Low Register) : 9EH

| 7              | 6              | 5              | 4              | 3     | 2     | 1     | 0     |
|----------------|----------------|----------------|----------------|-------|-------|-------|-------|
| ADDM3<br>ADDL7 | ADDM2<br>ADDL6 | ADDM1<br>ADDL5 | ADDM0<br>ADDL4 | ADDL3 | ADDL2 | ADDL1 | ADDL0 |
| R              | R              | R              | R              | R-    | R     | R     | R     |

Initial value : xxH

ADDM[3:0] MSB align, A/D Converter Low Data (4-bit)

ADDL[7:0] LSB align, A/D Converter Low Data (8-bit)

**ADCCRH (A/D Converter High Register) : 9DH**

| 7      | 6 | 5 | 4     | 3     | 2     | 1      | 0      |
|--------|---|---|-------|-------|-------|--------|--------|
| ADCIFR | – | – | TRIG1 | TRIG0 | ALIGN | CKSEL1 | CKSEL0 |
| RW     | – | – | RW    | RW    | RW    | RW     | RW     |

Initial value : 00H

ADCIFR When ADC interrupt occurs, this bit becomes '1'. For clearing bit, write '0' to this bit or auto clear by INT\_ACK signal.

0 ADC Interrupt no generation

1 ADC Interrupt generation

TRIG[1:0] A/D Trigger Signal Selection(The ADC module is automatically disabled at stop mode)

| TRIG1 | TRIG0 | Description            |
|-------|-------|------------------------|
| 0     | 0     | ADST                   |
| 0     | 1     | Timer 0 match signal   |
| 1     | 0     | Timer 1 A match signal |
| 1     | 1     | Timer 2 A match signal |

ALIGN A/D Converter data align selection.

0 MSB align (ADCDRH[7:0], ADCDRL[7:4])

1 LSB align (ADCDRH[3:0], ADCDRL[7:0])

CKSEL[1:0] A/D Converter Clock selection

| CKSEL1 | CKSEL0 | Description |
|--------|--------|-------------|
| 0      | 0      | fx/1        |
| 0      | 1      | fx/2        |
| 1      | 0      | fx/4        |
| 1      | 1      | fx/8        |



**ADCCRL (A/D Converter Counter Low Register) : 9CH**

| 7    | 6    | 5      | 4     | 3      | 2      | 1      | 0      |
|------|------|--------|-------|--------|--------|--------|--------|
| STBY | ADST | REFSEL | AFLAG | ADSEL3 | ADSEL2 | ADSEL1 | ADSEL0 |
| RW   | RW   | RW     | R     | RW     | RW     | RW     | RW     |

Initial value : 00H

|            |                                                                                                                            |                                                |        |        |             |
|------------|----------------------------------------------------------------------------------------------------------------------------|------------------------------------------------|--------|--------|-------------|
| STBY       | Control Operation of A/D<br>(The ADC module is automatically disabled at stop mode)                                        |                                                |        |        |             |
|            | 0                                                                                                                          | ADC module disable                             |        |        |             |
|            | 1                                                                                                                          | ADC module enable                              |        |        |             |
| ADST       | Control A/D Conversion start.                                                                                              |                                                |        |        |             |
|            | 0                                                                                                                          | No effect                                      |        |        |             |
|            | 1                                                                                                                          | Trigger signal generation for conversion start |        |        |             |
| REFSEL     | A/D Converter Reference Selection                                                                                          |                                                |        |        |             |
|            | 0                                                                                                                          | Internal Reference (VDD)                       |        |        |             |
|            | 1                                                                                                                          | External Reference (AVREF)                     |        |        |             |
| AFLAG      | A/D Converter Operation State (This bit is cleared to '0' when the STBY bit is set to '0' or when the CPU is at STOP mode) |                                                |        |        |             |
|            | 0                                                                                                                          | During A/D Conversion                          |        |        |             |
|            | 1                                                                                                                          | A/D Conversion finished                        |        |        |             |
| ADSEL[3:0] | A/D Converter input selection                                                                                              |                                                |        |        |             |
|            | ADSEL3                                                                                                                     | ADSEL2                                         | ADSEL1 | ADSEL0 | Description |
|            | 0                                                                                                                          | 0                                              | 0      | 0      | AN0         |
|            | 0                                                                                                                          | 0                                              | 0      | 1      | AN1         |
|            | 0                                                                                                                          | 0                                              | 1      | 0      | AN2         |
|            | 0                                                                                                                          | 0                                              | 1      | 1      | AN3         |
|            | 0                                                                                                                          | 1                                              | 0      | 0      | AN4         |
|            | 0                                                                                                                          | 1                                              | 0      | 1      | AN5         |
|            | 0                                                                                                                          | 1                                              | 1      | 0      | AN6         |
|            | 0                                                                                                                          | 1                                              | 1      | 1      | AN7         |
|            | 1                                                                                                                          | 0                                              | 0      | 0      | AN8         |
|            | 1                                                                                                                          | 0                                              | 0      | 1      | AN9         |
|            | 1                                                                                                                          | 0                                              | 1      | 0      | AN10        |
|            | 1                                                                                                                          | 0                                              | 1      | 1      | AN11        |
|            | 1                                                                                                                          | 1                                              | 0      | 0      | AN12        |
|            | 1                                                                                                                          | 1                                              | 0      | 1      | AN13        |
|            | 1                                                                                                                          | 1                                              | 1      | 0      | AN14        |
|            | 1                                                                                                                          | 1                                              | 1      | 1      | VDD18       |

**ADWRCR0 (ADC Wake-up Resistor Control Register 0): F2H**

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| AN3RS1 | AN3RS0 | AN2RS1 | AN2RS0 | AN1RS1 | AN1RS0 | AN0RS1 | AN0RS0 |
| RW     | RW     | RW     | RW     | RW     | RW     | RW     | RW     |

Initial value: 00H

ADWRCR0[7:0] ADC Wake-up Resistor selection for ANn input

| ANnRS[1:0] | 300k $\Omega$ Resistor | 150k $\Omega$ Resistor |
|------------|------------------------|------------------------|
| 0 0        | Disable                | Disable                |
| 0 1        | Disable                | Enable                 |
| 1 0        | Enable                 | Disable                |
| 1 1        | Enable                 | Enable                 |

Where n = 0, 1, 2, and 3

**ADWRCR1 (ADC Wake-up Resistor Control Register 1): F3H**

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| AN7RS1 | AN7RS0 | AN6RS1 | AN6RS0 | AN5RS1 | AN5RS0 | AN4RS1 | AN4RS0 |
| RW     | RW     | RW     | RW     | RW     | RW     | RW     | RW     |

Initial value: 00H

ADWRCR1[7:0] ADC Wake-up Resistor selection for ANn input

| ANnRS[1:0] | 300k $\Omega$ Resistor | 150k $\Omega$ Resistor |
|------------|------------------------|------------------------|
| 0 0        | Disable                | Disable                |
| 0 1        | Disable                | Enable                 |
| 1 0        | Enable                 | Disable                |
| 1 1        | Enable                 | Enable                 |

Where n = 4, 5, 6, and 7

**ADWRCR2 (ADC Wake-up Resistor Control Register 2): F4H**

| 7       | 6       | 5       | 4       | 3      | 2      | 1      | 0      |
|---------|---------|---------|---------|--------|--------|--------|--------|
| AN11RS1 | AN11RS0 | AN10RS1 | AN10RS0 | AN9RS1 | AN9RS0 | AN8RS1 | AN8RS0 |
| RW      | RW      | RW      | RW      | RW     | RW     | RW     | RW     |

Initial value: 00H

ADWRCR2[7:0] ADC Wake-up Resistor selection for ANn input

| ANnRS[1:0] | 300k $\Omega$ Resistor | 150k $\Omega$ Resistor |
|------------|------------------------|------------------------|
| 0 0        | Disable                | Disable                |
| 0 1        | Disable                | Enable                 |
| 1 0        | Enable                 | Disable                |
| 1 1        | Enable                 | Enable                 |

Where n = 8, 9, 10, and 11

**ADWRCR3 (ADC Wake-up Resistor Control Register 3): F5H**

| 7 | 6 | 5       | 4       | 3       | 2       | 1       | 0       |
|---|---|---------|---------|---------|---------|---------|---------|
| – | – | AN14RS1 | AN14RS0 | AN13RS1 | AN13RS0 | AN12RS1 | AN12RS0 |
| – | – | RW      | RW      | RW      | RW      | RW      | RW      |

Initial value: 00H

ADWRCR3[5:0] ADC Wake-up Resistor selection for ANn input

ANnRS[1:0]      300k $\Omega$  Resistor      150k $\Omega$  Resistor

0      0      Disable      Disable

0      1      Disable      Enable

1      0      Enable      Disable

1      1      Enable      Enable

Where n = 12, 13, and 14

**ADWCRH (ADC Wake-up Control High Register): F7H**

| 7 | 6       | 5       | 4       | 3       | 2       | 1      | 0      |
|---|---------|---------|---------|---------|---------|--------|--------|
| – | AN14WEN | AN13WEN | AN12WEN | AN11WEN | AN10WEN | AN9WEN | AN8WEN |
| – | RW      | RW      | RW      | RW      | RW      | RW     | RW     |

Initial value: 00H

ADWCRH[6:0] Enable or Disable ADC Wake-up Function for ANn input

ANnWEN      Description

0      Disable

1      Enable

Where n = 8, 9, 10, 11, 12, 13, and 14

**ADWCRL (ADC Wake-up Control Low Register): F6H**

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| AN7WEN | AN6WEN | AN5WEN | AN4WEN | AN3WEN | AN2WEN | AN1WEN | AN0WEN |
| RW     | RW     | RW     | RW     | RW     | RW     | RW     | RW     |

Initial value: 00H

ADWCRL[7:0] Enable or Disable ADC Wake-up Function for ANn input

ANnWEN      Description

0      Disable

1      Enable

Where n = 0, 1, 2, 3, 4, 5, 6, and 7

**ADWIFRH (ADC Wake-up Interrupt Flag High Register): DDH**

| 7 | 6        | 5        | 4        | 3        | 2        | 1       | 0       |
|---|----------|----------|----------|----------|----------|---------|---------|
| – | AN14WIFR | AN13WIFR | AN12WIFR | AN11WIFR | AN10WIFR | AN9WIFR | AN8WIFR |
| – | RW       | RW       | RW       | RW       | RW       | RW      | RW      |

Initial value: 00H

ADWIFRH[6:0] When a ADC wake-up interrupt AN14 ~ AN8 is occurred, the flag becomes '1'. The flag is cleared only by writing a '0' to the bit. So, the flag should be cleared by software.

0 ADC wake-up interrupt AN14 ~ AN8 not occurred

1 ADC wake-up interrupt AN14 ~ AN8 occurred

**ADWIFRL (ADC Wake-up Interrupt Flag Low Register): DCH**

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| AN7WIFR | AN6WIFR | AN5WIFR | AN4WIFR | AN3WIFR | AN2WIFR | AN1WIFR | AN0WIFR |
| RW      | RW      | RW      | RW      | RW      | RW      | RW      | RW      |

Initial value: 00H

ADWIFRL[7:0] When a ADC wake-up interrupt AN7 ~ AN0 is occurred, the flag becomes '1'. The flag is cleared only by writing a '0' to the bit. So, the flag should be cleared by software.

0 ADC wake-up interrupt AN7 ~ AN0 not occurred

1 ADC wake-up interrupt AN7 ~ AN0 occurred



### 11.10.3 Data Transmit / Receive Operation

User can use SPI for serial data communication by following step

1. Select SPI operation mode(master/slave, polarity, phase) by control register SPICR.
2. When the SPI is configured as a Master, it selects a Slave by SS signal (active low).  
When the SPI is configured as a Slave, it is selected by SS signal incoming from Master
3. When the user writes a byte to the data register SPIDR, SPI will start an operation.
4. In this time, if the SPI is configured as a Master, serial clock will come out of SCK pin. And Master shifts the eight bits into the Slave (transmit), Slave shifts the eight bits into the Master at the same time (receive). If the SPI is configured as a Slave, serial clock will come into SCK pin. And Slave shifts the eight bits into the Master (transmit), Master shifts the eight bits into the Slave at the same time (receive).
5. When transmit/receive is done, SPIIFR bit will be set. If the SPI interrupt is enabled, an interrupt is requested. And SPIIFR bit is cleared by hardware when executing the corresponding interrupt. If SPI interrupt is disable, SPIIFR bit is cleared when user read the status register SPISR, and then access (read/write) the data register SPIDR.

### 11.10.4 SS pin function

1. When the SPI is configured as a Slave, the SS pin is always input. If LOW signal come into SS pin, the SPI logic is active. And if 'HIGH' signal come into SS pin, the SPI logic is stop. In this time, SPI logic will be reset, and invalidated any received data.
2. When the SPI is configured as a Master, the user can select the direction of the SS pin by port direction register (P17IO). If the SS pin is configured as an output, user can use general P17IO output mode. If the SS pin is configured as an input, 'HIGH' signal must come into SS pin to guarantee Master operation. If 'LOW' signal come into SS pin, the SPI logic interprets this as another master selecting the SPI as a slave and starting to send data to it. To avoid bus contention, MSB bit of SPICR will be cleared and the SPI becomes a Slave and then, SPIIFR bit of SPISR will be set, and if the SPI interrupt is enabled, an interrupt is requested.

#### NOTES)

- When the SS pin is configured as an output at Master mode, SS pin's output value is defined by user's software (P17IO). Before SPICR setting, the direction of SS pin must be defined
- If you don't need to use SS pin, clear the SENA bit of SPISR. So, you can use disabled pin by P17IO freely. In this case, SS signal is driven by 'HIGH' or 'LOW' internally. In other words, master is 'HIGH', slave is 'LOW'
- When SS pin is configured as input, if 'HIGH' signal come into SS pin, SS\_HIGH flag bit will be set. And you can clear it by writing '0'.

### 11.10.5 SPI Timing Diagram

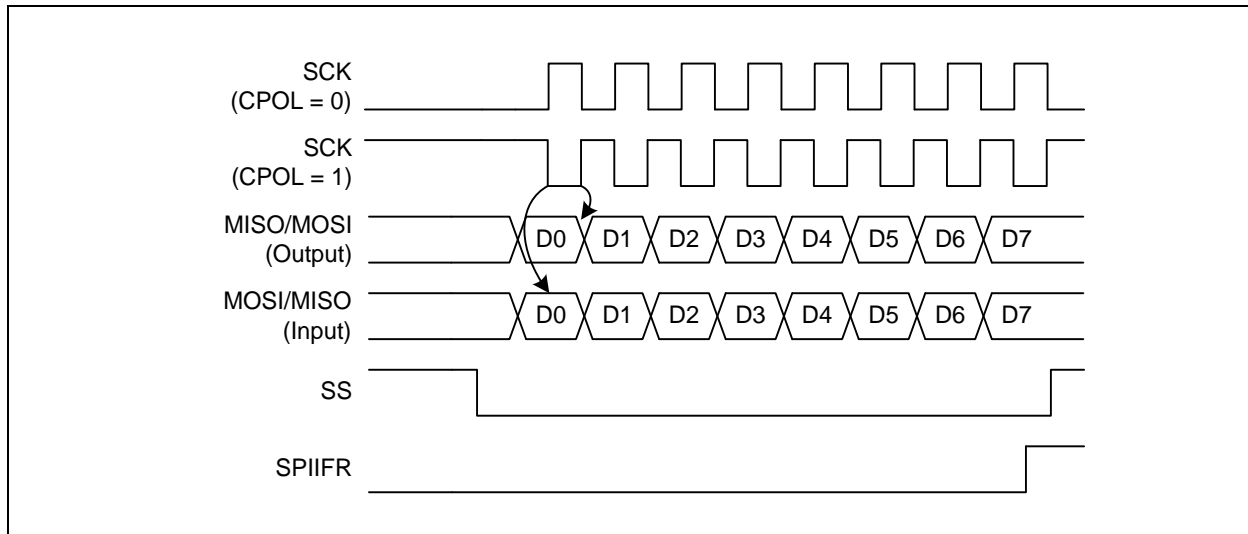


Figure 11.39 SPI Transmit/Receive Timing Diagram at CPHA = 0

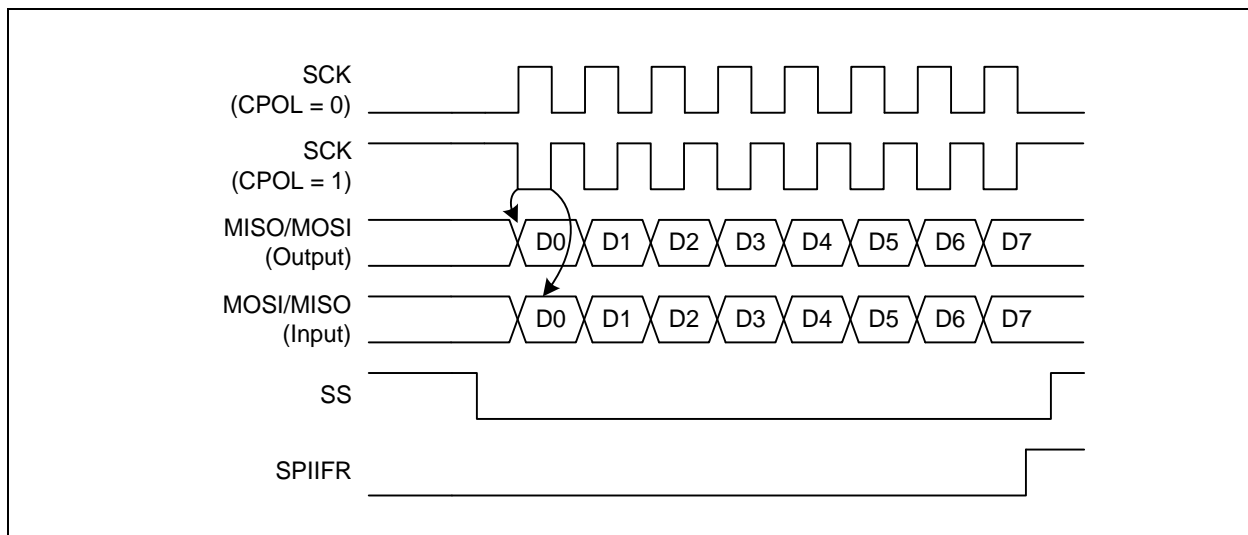


Figure 11.40 SPI Transmit/Receive Timing Diagram at CPHA = 1

### 11.10.6 Register Map

Table 11-14 SPI Register Map

| Name  | Address | Dir | Default | Description          |
|-------|---------|-----|---------|----------------------|
| SPISR | B7H     | R/W | 00H     | SPI Status Register  |
| SPIDR | B6H     | R/W | 00H     | SPI Data Register    |
| SPICR | B5H     | R/W | 00H     | SPI Control Register |

### 11.10.7 SPI Register Description

The SPI register consists of SPI control register (SPICR), SPI status register (SPISR) and SPI data register (SPIDR)

### 11.10.8 Register Description for SPI

#### SPIDR (SPI Data Register) : B6H

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| SPIDR7 | SPIDR6 | SPIDR5 | SPIDR4 | SPIDR3 | SPIDR2 | SPIDR1 | SPIDR0 |
| R/W    | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    |

Initial value : 00H

SPIDR [7:0]

SPI Data

When it is written a byte to this data register, the SPI will start an operation.



**SPISR (SPI Status Register) : B7H**

| 7      | 6    | 5       | 4 | 3    | 2     | 1 | 0 |
|--------|------|---------|---|------|-------|---|---|
| SPIIFR | WCOL | SS_HIGH | – | FXCH | SSENA | – | – |
| RW     | R    | RW      | – | RW   | RW    | – | – |

Initial value : 00H

|         |                                                                                                                                                                                                                                                                            |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SPIIFR  | When SPI Interrupt occurs, this bit becomes '1'. IF SPI interrupt is enable, this bit is auto cleared by INT_ACK signal. And if SPI Interrupt is disable, this bit is cleared when the status register SPISR is read, and then access (read/write) the data register SPIDR |
| 0       | SPI Interrupt no generation                                                                                                                                                                                                                                                |
| 1       | SPI Interrupt generation                                                                                                                                                                                                                                                   |
| WCOL    | This bit is set if any data are written to the data register SPIDR during transfer. This bit is cleared when the status register SPISR is read, and then access (read/write) the data register SPIDR                                                                       |
| 0       | No collision                                                                                                                                                                                                                                                               |
| 1       | Collision                                                                                                                                                                                                                                                                  |
| SS_HIGH | When the SS pin is configured as input, if "HIGH" signal comes into the pin, this flag bit will be set.                                                                                                                                                                    |
| 0       | Cleared when '0' is written                                                                                                                                                                                                                                                |
| 1       | No effect when '1' is written                                                                                                                                                                                                                                              |
| FXCH    | SPI port function exchange control bit.                                                                                                                                                                                                                                    |
| 0       | No effect                                                                                                                                                                                                                                                                  |
| 1       | Exchange MOSI and MISO function                                                                                                                                                                                                                                            |
| SSENA   | This bit controls the SS pin operation                                                                                                                                                                                                                                     |
| 0       | Disable                                                                                                                                                                                                                                                                    |
| 1       | Enable (The P17 should be a normal input)                                                                                                                                                                                                                                  |

**SPICR (SPI Control Register) : B5H**

| 7     | 6    | 5   | 4    | 3    | 2    | 1    | 0    |
|-------|------|-----|------|------|------|------|------|
| SPIEN | FLSB | MS  | CPOL | CPHA | DSCR | SCR1 | SCR0 |
| R/W   | R/W  | R/W | R/W  | R/W  | R/W  | R/W  | R/W  |

Initial value : 00H

|          |                                                                                                                                              |                       |                  |                  |
|----------|----------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|------------------|------------------|
| SPIEN    | This bit controls the SPI operation                                                                                                          |                       |                  |                  |
|          | 0                                                                                                                                            | Disable SPI operation |                  |                  |
|          | 1                                                                                                                                            | Enable SPI operation  |                  |                  |
| FLSB     | This bit selects the data transmission sequence                                                                                              |                       |                  |                  |
|          | 0                                                                                                                                            | MSB first             |                  |                  |
|          | 1                                                                                                                                            | LSB first             |                  |                  |
| MS       | This bit selects whether Master or Slave mode                                                                                                |                       |                  |                  |
|          | 0                                                                                                                                            | Slave mode            |                  |                  |
|          | 1                                                                                                                                            | Master mode           |                  |                  |
| CPOL     | This two bits control the serial clock (SCK) mode.                                                                                           |                       |                  |                  |
| CPHA     |                                                                                                                                              |                       |                  |                  |
|          | Clock polarity(CPOL) bit determine SCK's value at idle mode.                                                                                 |                       |                  |                  |
|          | Clock phase (CPHA) bit determine if data are sampled on the leading or trailing edge of SCK.                                                 |                       |                  |                  |
|          | CPOL                                                                                                                                         | CPHA                  | Leading edge     | Trailing edge    |
|          | 0                                                                                                                                            | 0                     | Sample (Rising)  | Setup (Falling)  |
|          | 0                                                                                                                                            | 1                     | Setup (Rising)   | Sample (Falling) |
|          | 1                                                                                                                                            | 0                     | Sample (Falling) | Setup (Rising)   |
|          | 1                                                                                                                                            | 1                     | Setup (Falling)  | Sample (Rising)  |
| DSCR     | These three bits select the SCK rate of the device configured as a master. When DSCR bit is written one, SCK will be doubled in master mode. |                       |                  |                  |
| SCR[2:0] |                                                                                                                                              |                       |                  |                  |
|          | DSCR                                                                                                                                         | SCR1                  | SCR0             | SCK frequency    |
|          | 0                                                                                                                                            | 0                     | 0                | fx/4             |
|          | 0                                                                                                                                            | 0                     | 1                | fx/16            |
|          | 0                                                                                                                                            | 1                     | 0                | fx/64            |
|          | 0                                                                                                                                            | 1                     | 1                | fx/128           |
|          | 1                                                                                                                                            | 0                     | 0                | fx/2             |
|          | 1                                                                                                                                            | 0                     | 1                | fx/8             |
|          | 1                                                                                                                                            | 1                     | 0                | fx/32            |
|          | 1                                                                                                                                            | 1                     | 1                | fx/64            |

## 11.11 UART

### 11.11.1 Overview

The universal asynchronous serial receiver and transmitter (UART) is a highly flexible serial communication device. The main features are listed below.

- Full Duplex Operation (Independent Serial Receive and Transmit Registers)
- Baud Rate Generator
- Supports Serial Frames with 5, 6, 7, 8, or 9 Data Bits and 1 or 2 Stop Bits
- Odd or Even Parity Generation and Parity Check Supported by Hardware
- Data OverRun Detection
- Framing Error Detection
- Three Separate Interrupts on TX Complete, TX Data Register Empty and RX Complete

UART has baud rate generator, transmitter and receiver. The baud rate generator for asynchronous operation. The Transmitter consists of a single write buffer, a serial shift register, parity generator and control logic for handling different serial frame formats. The write buffer allows continuous transfer of data without any delay between frames. The receiver is the most complex part of the UART module due to its clock and data recovery units. The recovery unit is used for asynchronous data reception. In addition to the recovery unit, the receiver includes a parity checker, a shift register, a two-level receive FIFO (UARTDR) and control logic. The receiver supports the same frame formats as the transmitter and can detect frame error, data overrun and parity errors.

## 11.11.2 Block Diagram

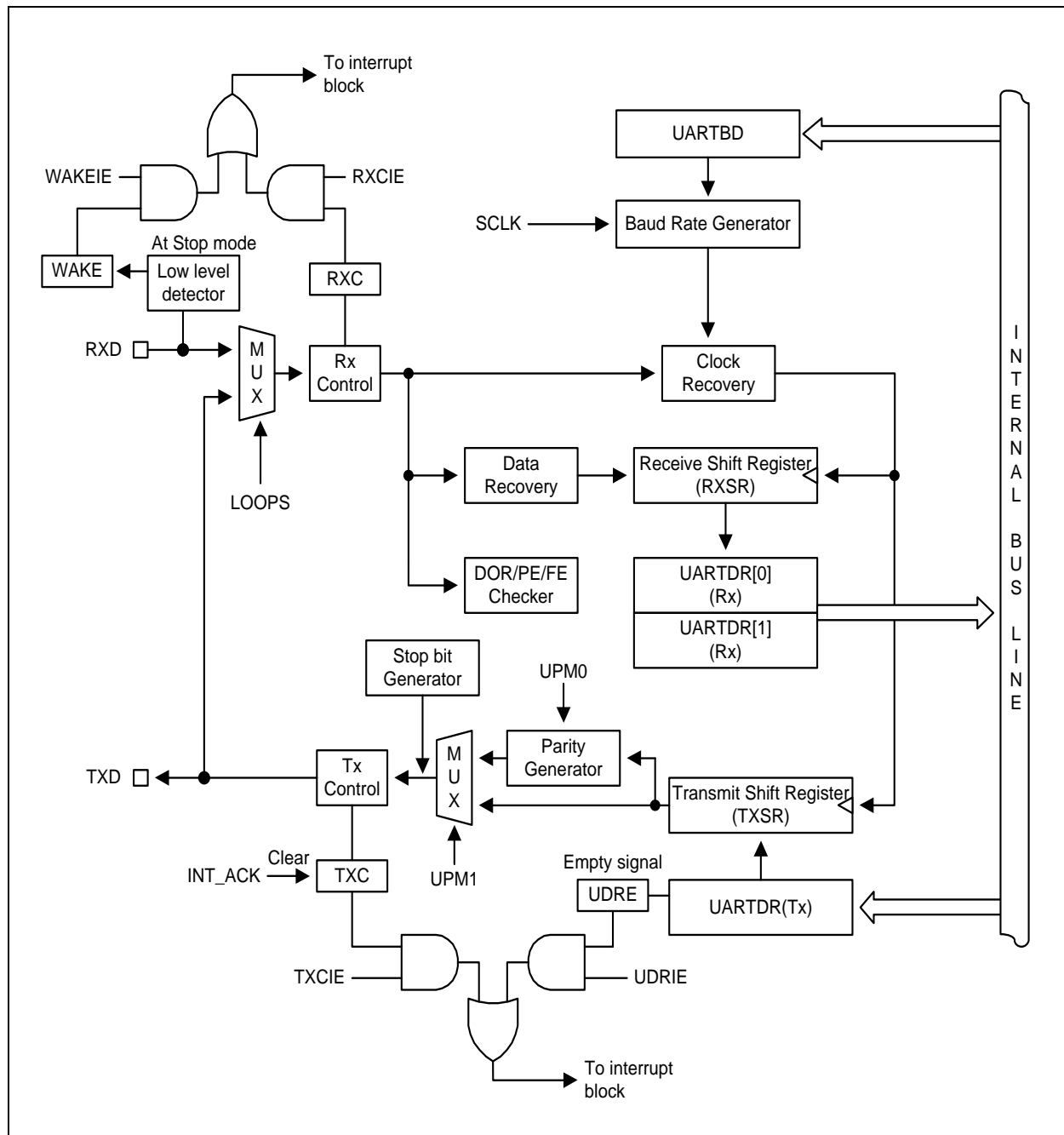
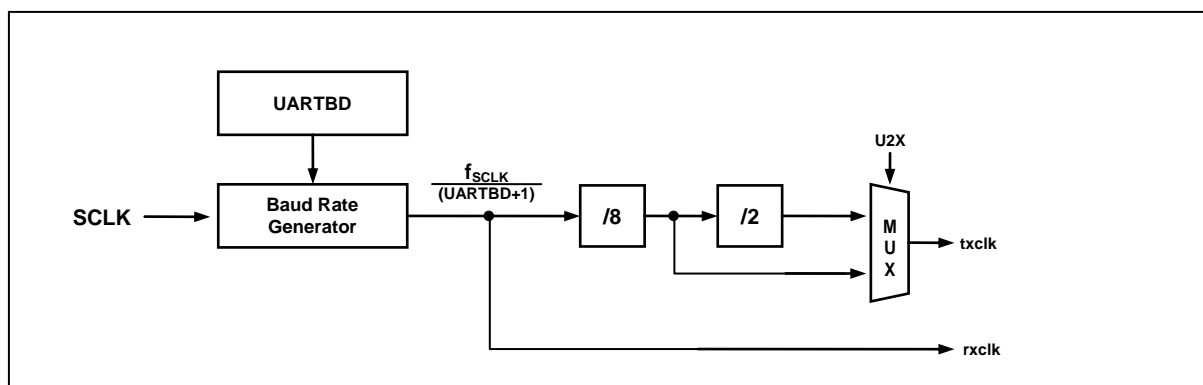


Figure 11.41 UART Block Diagram

### 11.11.3 Clock Generation



**Figure 11.42 Clock Generation Block Diagram**

The clock generation logic generates the base clock for the transmitter and receiver.

Following table shows equations for calculating the baud rate (in bps).

**Table 11-15 Equations for Calculating Baud Rate Register Setting**

| Operating Mode           | Equation for Calculating Baud Rate                     |
|--------------------------|--------------------------------------------------------|
| Normal Mode(U2X=0)       | $\text{Baud Rate} = \frac{f_x}{16(\text{UARTBD} + 1)}$ |
| Double Speed Mode(U2X=1) | $\text{Baud Rate} = \frac{f_x}{8(\text{UARTBD} + 1)}$  |

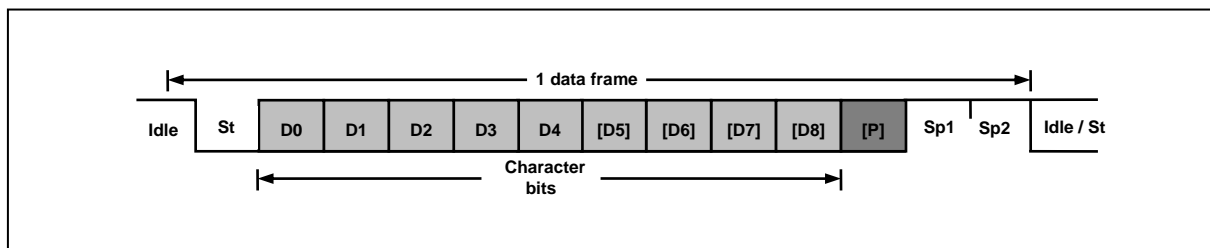
### 11.11.4 Data format

A serial frame is defined to be one character of data bits with synchronization bits (start and stop bits), and optionally a parity bit for error detection.

The UART supports all 30 combinations of the following as valid frame formats.

- 1 start bit
- 5, 6, 7, 8 or 9 data bits
- no, even or odd parity bit
- 1 or 2 stop bits

A frame starts with the start bit followed by the least significant data bit (LSB). Then the next data bits, up to nine, are succeeding, ending with the most significant bit (MSB). If parity function is enabled, the parity bit is inserted between the last data bit and the stop bit. A high-to-low transition on data pin is considered as start bit. When a complete frame is transmitted, it can be directly followed by a new frame, or the communication line can be set to an idle state. The idle means high state of data pin. The following figure shows the possible combinations of the frame formats. Bits inside brackets are optional.



**Figure 11.43 Frame Format**

1 data frame consists of the following bits

- Idle No communication on communication line (TxD/RxD)
- St Start bit (Low)
- Dn Data bits (0~8)
- Parity bit ----- Even parity, Odd parity, No parity
- Stop bit(s) ----- 1 bit or 2 bits

The frame format used by the UART is set by the USIZE[2:0], UPM[1:0] and USBS bits in UARTCR1 and UARTCR3 register. The Transmitter and Receiver use the same setting.

### 11.11.5 Parity bit

The parity bit is calculated by doing an exclusive-OR of all the data bits. If odd parity is used, the result of the exclusive-or is inverted. The parity bit is located between the MSB and first stop bit of a serial frame.

$$P_{\text{even}} = D_{n-1} \wedge \dots \wedge D_3 \wedge D_2 \wedge D_1 \wedge D_0 \wedge 0$$

$$P_{\text{odd}} = D_{n-1} \wedge \dots \wedge D_3 \wedge D_2 \wedge D_1 \wedge D_0 \wedge 1$$

$P_{\text{even}}$  : Parity bit using even parity

$P_{\text{odd}}$  : Parity bit using odd parity

$D_n$  : Data bit n of the character

### 11.11.6 UART Transmitter

The UART transmitter is enabled by setting the TXE bit in UARTCR2 register. When the Transmitter is enabled, the TXD pin should be set to TXD function for the serial output pin of UART by the P3FSR[1:0]. The baud-rate, operation mode and frame format must be setup once before doing any transmission.

#### 11.11.6.1 Sending Tx data

A data transmission is initiated by loading the transmit buffer (UARTDR register I/O location) with the data to be transmitted. The data written in transmit buffer is moved to the shift register when the shift register is ready to send a new frame. The shift register is loaded with the new data if it is in idle state or immediately after the last stop bit of the previous frame is transmitted. When the shift register is loaded with new data, it will transfer one complete frame according to the settings of control registers. If the 9-bit characters are used, the ninth bit must be written to the TX8 bit in UARTCR3 register before it is loaded to the transmit buffer (UARTDR register).

#### 11.11.6.2 Transmitter flag and interrupt

The UART transmitter has 2 flags which indicate its state. One is UART data register empty flag (UDRE) and the other is transmit complete flag (TxC). Both flags can be interrupt sources.

UDRE flag indicates whether the transmit buffer is ready to receive new data. This bit is set when the transmit buffer is empty and cleared when the transmit buffer contains data to be transmitted but has not yet been moved into the shift register. And also this flag can be cleared by writing '0' to this bit position. Writing '1' to this bit position is prevented.

When the data register empty interrupt enable (UDRIE) bit in UARTCR2 register is set and the global interrupt is enabled, UART data register empty interrupt is generated while UDRE flag is set.

The transmit complete (TxC) flag bit is set when the entire frame in the transmit shift register has been shifted out and there is no more data in the transmit buffer. The TxC flag is automatically cleared when the transmit complete interrupt service routine is executed, or it can be cleared by writing '0' to TxC bit in UARTST register.

When the transmit complete interrupt enable (TxCIE) bit in UARTCR2 register is set and the global interrupt is enabled, UART transmit complete interrupt is generated while TxC flag is set.

### 11.11.6.3 Parity Generator

The parity generator calculates the parity bit for the serial frame data to be sent. When parity bit is enabled (UPM[1]=1), the transmitter control logic inserts the parity bit between the MSB and the first stop bit of the frame to be sent.

### 11.11.6.4 Disabling Transmitter

Disabling the transmitter by clearing the TXE bit will not become effective until ongoing transmission is completed. When the Transmitter is disabled, the TXD pin can be used as a normal general purpose I/O (GPIO).

### 11.11.7 UART Receiver

The UART receiver is enabled by setting the RXE bit in the UARTCR2 register. When the receiver is enabled, the RXD pin should be set to the input port for the serial input pin of UART by P31IO bit. The baud-rate, mode of operation and frame format must be set before serial reception.

#### 11.11.7.1 Receiving Rx data

The receiver starts data reception when it detects a valid start bit (LOW) on RXD pin. Each bit after start bit is sampled at pre-defined baud-rate (asynchronous) and shifted into the receive shift register until the first stop bit of a frame is received. Even if there's 2<sup>nd</sup> stop bit in the frame, the 2<sup>nd</sup> stop bit is ignored by the receiver. That is, receiving the first stop bit means that a complete serial frame is present in the receiver shift register and contents of the shift register are to be moved into the receive buffer. The receive buffer is read by reading the UARTDR register.

If 9-bit characters are used (USIZE[2:0] = "111"), the ninth bit is stored in the RX8 bit position in the UARTCR3 register. The 9<sup>th</sup> bit must be read from the RX8 bit before reading the low 8 bits from the UARTDR register. Likewise, the error flags FE, DOR, PE must be read before reading the data from UARTDR register. It's because the error flags are stored in the same FIFO position of the receive buffer.



### 11.11.7.2 Receiver Flag and Interrupt

The UART receiver has one flag that indicates the receiver state.

The receive complete (RXC) flag indicates whether there are unread data in the receive buffer. This flag is set when there are unread data in the receive buffer and cleared when the receive buffer is empty. If the receiver is disabled (RXE=0), the receiver buffer is flushed and the RXC flag is cleared.

When the receive complete interrupt enable (RXCIE) bit in the UARTCR2 register is set and global interrupt is enabled, the UART receiver complete interrupt is generated while RXC flag is set.

The UART receiver has three error flags which are frame error (FE), data overrun (DOR) and parity error (PE). These error flags can be read from the UARTST register. As received data are stored in the 2-level receive buffer, these error flags are also stored in the same position of receive buffer. So, before reading received data from UARTDR register, read the UARTST register first which contains error flags.

The frame error (FE) flag indicates the state of the first stop bit. The FE flag is '0' when the stop bit was correctly detected as '1', and the FE flag is '1' when the stop bit was incorrect, i.e. detected as '0'. This flag can be used for detecting out-of-sync conditions between data frames.

The data overrun (DOR) flag indicates data loss due to a receive buffer full condition. DOR occurs when the receive buffer is full, and another new data is present in the receive shift register which are to be stored into the receive buffer. After the DOR flag is set, all the incoming data are lost. To prevent data loss or clear this flag, read the receive buffer.

The parity error (PE) flag indicates that the frame in the receive buffer had a parity error when received. If parity check function is not enabled (UPM[1]=0), the PE bit is always read '0'.

### 11.11.7.3 Parity Checker

If parity bit is enabled (UPM[1]=1), the Parity Checker calculates the parity of the data bits in incoming frame and compares the result with the parity bit from the received serial frame.

### 11.11.7.4 Disabling Receiver

In contrast to transmitter, disabling the Receiver by clearing RXE bit makes the Receiver inactive immediately. When the receiver is disabled, the receiver flushes the receive buffer, the remaining data in the buffer is all reset, and the RXD pin can be used as a normal general purpose I/O (GPIO).

### 11.11.7.5 Asynchronous Data Reception

To receive asynchronous data frame, the UART includes a clock and data recovery unit. The clock recovery logic is used for synchronizing the internally generated baud-rate clock to the incoming asynchronous serial frame on the RXD pin.

The data recovery logic samples and low pass filters the incoming bits, and this removes the noise of RXD pin.

The next figure illustrates the sampling process of the start bit of an incoming frame. The sampling rate is 16 times the baud-rate for normal mode ( $U2X=0$ ) and 8 times the baud-rate for double speed mode ( $U2X=1$ ). The horizontal arrows show the synchronization variation due to the asynchronous sampling process. Note that larger time variation is shown when using the double speed mode.

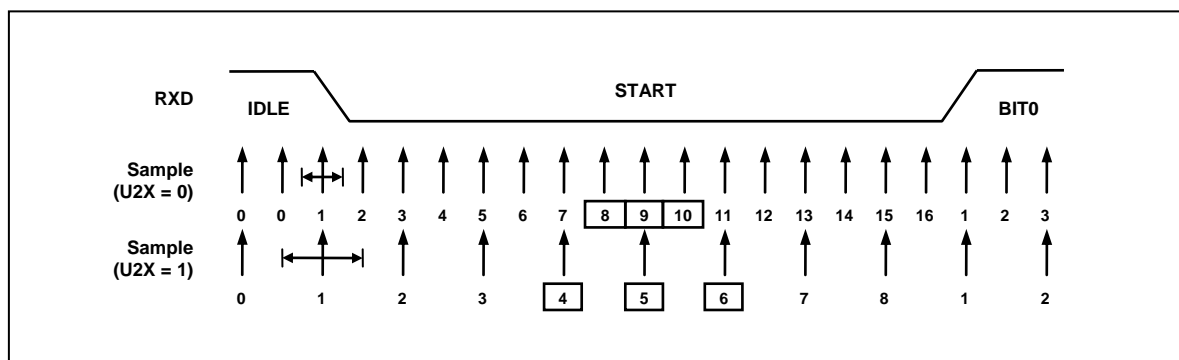


Figure 11.44 Start Bit Sampling

When the receiver is enabled ( $RXE=1$ ), the clock recovery logic tries to find a high-to-low transition on the RXD line, the start bit condition. After detecting high to low transition on RXD line, the clock recovery logic uses samples 8, 9, and 10 for normal mode, and samples 4, 5, and 6 for double speed mode to decide if a valid start bit is received. If more than 2 samples have logical low level, it is considered that a valid start bit is detected and the internally generated clock is synchronized to the incoming data frame. And the data recovery can begin. The synchronization process is repeated for each start bit.

As described above, when the receiver clock is synchronized to the start bit, the data recovery can begin. Data recovery process is almost similar to the clock recovery process. The data recovery logic samples 16 times for each incoming bits for normal mode and 8 times for double speed mode. And uses sample 8, 9, and 10 to decide data value for normal mode, and samples 4, 5, and 6 for double speed mode. If more than 2 samples have low levels, the received bit is considered to a logic '0' and if more than 2 samples have high levels, the received bit is considered to a logic '1'. The data recovery process is then repeated until a complete frame is received including the first stop bit. The decided bit value is stored in the receive shift register in order. Note that the Receiver only uses the first stop bit of a frame. Internally, after receiving the first stop bit, the Receiver is in idle state and waiting to find start bit.

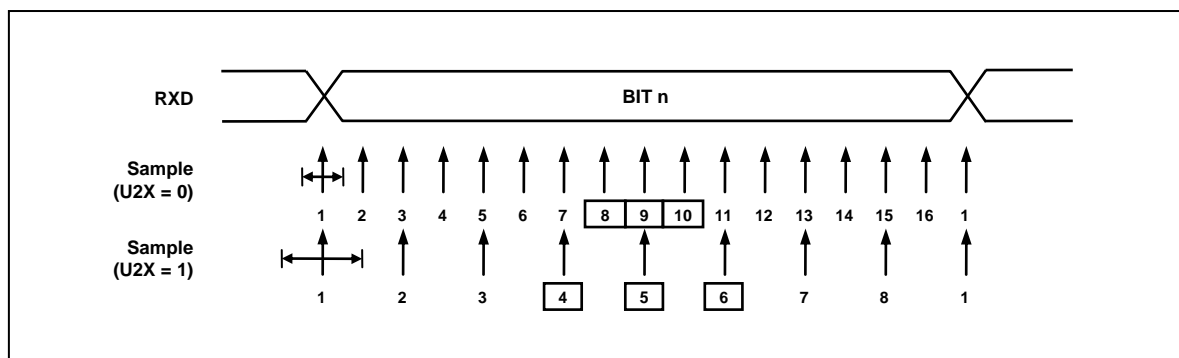


Figure 11.45 Sampling of Data and Parity Bit

The process for detecting stop bit is like clock and data recovery process. That is, if 2 or more samples of 3 center values have high level, correct stop bit is detected, else a frame error (FE) flag is set. After deciding whether the first stop bit is valid or not, the Receiver goes to idle state and monitors the RXD line to check a valid high to low transition is detected (start bit detection).

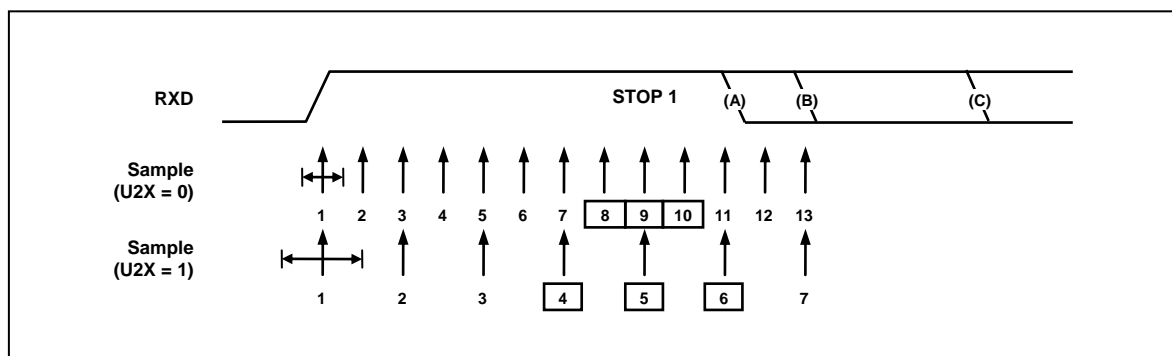


Figure 11.46 Stop Bit Sampling and Next Start Bit Sampling

### 11.11.8 Register Map

Table 11-16 UART Register Map

| Name    | Address | Dir | Default | Description                        |
|---------|---------|-----|---------|------------------------------------|
| UARTBD  | E6H     | R/W | FFH     | UART Baud Rate Generation Register |
| UARTDR  | E7H     | R/W | 00H     | UART Data Register                 |
| UARTCR1 | E2H     | R/W | 00H     | UART Control Register 1            |
| UARTCR2 | E3H     | R/W | 00H     | UART Control Register 2            |
| UARTCR3 | E4H     | R/W | 00H     | UART Control Register 3            |
| UARTST  | E5H     | R/W | 80H     | UART Status Register               |

### 11.11.9 UART Register Description

UART module consists of UART baud rate generation register (UARTBD), UART data register (UARTDR), UART control register 1 (UARTCR1), UART control register 2 (UARTCR2), UART control register 3 (UARTCR3), and UART status register (UARTST).

### 11.11.10 Register Description for UART

#### UARTBD (UART Baud Rate Generation Register) : E6H

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| UARTBD7 | UARTBD6 | UARTBD5 | UARTBD4 | UARTBD3 | UARTBD2 | UARTBD1 | UARTBD0 |
| R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |

Initial value : FFH

UARTBD [7:0] The value in this register is used to generate internal baud rate. To prevent malfunction, do not write '0'.

**UARTDR (UART Data Register) : E7H**

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| UARTDR7 | UARTDR6 | UARTDR5 | UARTDR4 | UARTDR3 | UARTDR2 | UARTDR1 | UARTDR0 |
| RW      | RW      | RW      | RW      | RW      | RW      | RW      | RW      |

Initial value : 00H

UARTDR [7:0] The UART Transmit Buffer and Receive Buffer share the same I/O address with this DATA register. The Transmit Data Buffer is the destination for data written to the UARTDR register. Reading the UARTDR register returns the contents of the Receive Buffer.  
Write this register only when the UDRE flag is set.

**UARTCR1 (UART Control Register 1) : E2H**

| 7 | 6 | 5    | 4    | 3      | 2      | 1      | 0 |
|---|---|------|------|--------|--------|--------|---|
| - | - | UPM1 | UPM0 | USIZE2 | USIZE1 | USIZE0 | - |
| - | - | RW   | RW   | RW     | RW     | RW     | - |

Initial value : 00H

UPM[1:0] Selects Parity Generation and Check methods

| UPM1 | UPM0 | Parity      |
|------|------|-------------|
| 0    | 0    | No Parity   |
| 0    | 1    | Reserved    |
| 1    | 0    | Even Parity |
| 1    | 1    | Odd Parity  |

USIZE[2:0] Selects the Length of Data Bits in Frame

| USIZE2       | USIZE1 | USIZE0 | Data Length |
|--------------|--------|--------|-------------|
| 0            | 0      | 0      | 5 bit       |
| 0            | 0      | 1      | 6 bit       |
| 0            | 1      | 0      | 7 bit       |
| 0            | 1      | 1      | 8 bit       |
| 1            | 1      | 1      | 9 bit       |
| Other values |        |        | Reserved    |

**UARTCR2 (UART Control Register 2) : E3H**

| 7     | 6     | 5     | 4      | 3   | 2   | 1      | 0   |
|-------|-------|-------|--------|-----|-----|--------|-----|
| UDRIE | TXCIE | RXCIE | WAKEIE | TXE | RXE | UARTEN | U2X |
| RW    | RW    | RW    | RW     | RW  | RW  | RW     | RW  |

Initial value : 00H

|        |                                                                                                                                                                                                                                                         |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UDRIE  | Interrupt enable bit for UART Data Register Empty                                                                                                                                                                                                       |
| 0      | Interrupt from UDRE is inhibited (use polling)                                                                                                                                                                                                          |
| 1      | When UDRE is set, request an interrupt                                                                                                                                                                                                                  |
| TXCIE  | Interrupt enable bit for Transmit Complete                                                                                                                                                                                                              |
| 0      | Interrupt from TXC is inhibited (use polling)                                                                                                                                                                                                           |
| 1      | When TXC is set, request an interrupt                                                                                                                                                                                                                   |
| RXCIE  | Interrupt enable bit for Receive Complete                                                                                                                                                                                                               |
| 0      | Interrupt from RXC is inhibited (use polling)                                                                                                                                                                                                           |
| 1      | When RXC is set, request an interrupt                                                                                                                                                                                                                   |
| WAKEIE | Interrupt enable bit for Wake in STOP mode. When device is in stop mode, if RXD goes to LOW level, an interrupt can be requested to wake-up system. At that time the UDRIE bit and UARTST register value should be set to '0b' and "00H", respectively. |
| 0      | Interrupt from Wake is inhibited                                                                                                                                                                                                                        |
| 1      | When WAKE is set, request an interrupt                                                                                                                                                                                                                  |
| TXE    | Enables the transmitter unit                                                                                                                                                                                                                            |
| 0      | Transmitter is disabled                                                                                                                                                                                                                                 |
| 1      | Transmitter is enabled                                                                                                                                                                                                                                  |
| RXE    | Enables the receiver unit                                                                                                                                                                                                                               |
| 0      | Receiver is disabled                                                                                                                                                                                                                                    |
| 1      | Receiver is enabled                                                                                                                                                                                                                                     |
| UARTEN | Activate UART module by supplying clock. When one of TXE and RXE values is "1", the UARTEN bit always set to "1".                                                                                                                                       |
| 0      | UART is disabled (clock is halted)                                                                                                                                                                                                                      |
| 1      | UART is enabled                                                                                                                                                                                                                                         |
| U2X    | This bit selects receiver sampling rate.                                                                                                                                                                                                                |
| 0      | Normal Asynchronous operation                                                                                                                                                                                                                           |
| 1      | Double Speed Asynchronous operation                                                                                                                                                                                                                     |

**UARTCR3 (UART Control Register 3) : E4H**

| 7 | 6     | 5 | 4 | 3 | 2    | 1   | 0   |
|---|-------|---|---|---|------|-----|-----|
| - | LOOPS | - | - | - | USBS | TX8 | RX8 |
| - | RW    | - | - | - | RW   | RW  | R   |

Initial value : 00H

|       |                                                                                              |
|-------|----------------------------------------------------------------------------------------------|
| LOOPS | Controls the Loop Back Mode of UART, for test mode                                           |
| 0     | Normal operation                                                                             |
| 1     | Loop Back mode                                                                               |
| USBS  | Selects the length of stop bit.                                                              |
| 0     | 1 Stop Bit                                                                                   |
| 1     | 2 Stop Bit                                                                                   |
| TX8   | The ninth bit of data frame in UART. Write this bit first before loading the UARTDR register |
| 0     | MSB (9 <sup>th</sup> bit) to be transmitted is '0'                                           |
| 1     | MSB (9 <sup>th</sup> bit) to be transmitted is '1'                                           |
| RX8   | The ninth bit of data frame in UART. Read this bit first before reading the receive buffer   |
| 0     | MSB (9 <sup>th</sup> bit) received is '0'                                                    |
| 1     | MSB (9 <sup>th</sup> bit) received is '1'                                                    |

**UARTST (UART Status Register) : E5H**

| 7    | 6   | 5   | 4    | 3        | 2   | 1  | 0  |
|------|-----|-----|------|----------|-----|----|----|
| UDRE | TXC | RXC | WAKE | SOFTTRST | DOR | FE | PE |
| RW   | RW  | R   | RW   | RW       | R   | RW | RW |

Initial value : 80H

|          |                                                                                                                                                                                                                                                                                                             |
|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UDRE     | The UDRE flag indicates if the transmit buffer (UARTDR) is ready to receive new data. If UDRE is '1', the buffer is empty and ready to be written. This flag can generate a UDRE interrupt.                                                                                                                 |
| 0        | Transmit buffer is not empty.                                                                                                                                                                                                                                                                               |
| 1        | Transmit buffer is empty.                                                                                                                                                                                                                                                                                   |
| TXC      | This flag is set when the entire frame in the transmit shift register has been shifted out and there is no new data currently present in the transmit buffer. This flag is automatically cleared when the interrupt service routine of a TXC interrupt is executed. This flag can generate a TXC interrupt. |
| 0        | Transmission is ongoing.                                                                                                                                                                                                                                                                                    |
| 1        | Transmit buffer is empty and the data in transmit shift register are shifted out completely.                                                                                                                                                                                                                |
| RXC      | This flag is set when there are unread data in the receive buffer and cleared when all the data in the receive buffer are read. The RXC flag can be used to generate a RXC interrupt.                                                                                                                       |
| 0        | There is no data unread in the receive buffer                                                                                                                                                                                                                                                               |
| 1        | There are more than 1 data in the receive buffer                                                                                                                                                                                                                                                            |
| WAKE     | This flag is set when the RXD pin is detected low while the CPU is in stop mode. This flag can be used to generate a WAKE interrupt. This bit should be cleared by program software.                                                                                                                        |
| 0        | No WAKE interrupt is generated.                                                                                                                                                                                                                                                                             |
| 1        | WAKE interrupt is generated.                                                                                                                                                                                                                                                                                |
| SOFTTRST | This is an internal reset and only has effect on UART. Writing '1' to this bit initializes the internal logic of UART and this bit is automatically cleared.                                                                                                                                                |
| 0        | No operation                                                                                                                                                                                                                                                                                                |
| 1        | Reset UART                                                                                                                                                                                                                                                                                                  |
| DOR      | This bit is set if a Data OverRun occurs. While this bit is set, the incoming data frame is ignored. This flag is valid until the receive buffer is read.                                                                                                                                                   |
| 0        | No Data OverRun                                                                                                                                                                                                                                                                                             |
| 1        | Data OverRun detected                                                                                                                                                                                                                                                                                       |
| FE       | This bit is set if the first stop bit of next character in the receive buffer is detected as '0'. This bit is valid until the receive buffer is read.                                                                                                                                                       |
| 0        | No Frame Error                                                                                                                                                                                                                                                                                              |
| 1        | Frame Error detected                                                                                                                                                                                                                                                                                        |
| PE       | This bit is set if the next character in the receive buffer has a Parity Error to be received while Parity Checking is enabled. This bit is valid until the receive buffer is read.                                                                                                                         |
| 0        | No Parity Error                                                                                                                                                                                                                                                                                             |
| 1        | Parity Error detected                                                                                                                                                                                                                                                                                       |

### 11.11.11 Baud Rate setting (example)

Table 11-17 Examples of UARTBD Settings for Commonly Used Oscillator Frequencies

| Baud Rate | fx=1.00MHz |        | fx=1.8432MHz |        | fx=2.00MHz |        |
|-----------|------------|--------|--------------|--------|------------|--------|
|           | UARTBD     | ERROR  | UARTBD       | ERROR  | UARTBD     | ERROR  |
| 2400      | 25         | 0.2%   | 47           | 0.0%   | 51         | 0.2%   |
| 4800      | 12         | 0.2%   | 23           | 0.0%   | 25         | 0.2%   |
| 9600      | 6          | -7.0%  | 11           | 0.0%   | 12         | 0.2%   |
| 14.4k     | 3          | 8.5%   | 7            | 0.0%   | 8          | -3.5%  |
| 19.2k     | 2          | 8.5%   | 5            | 0.0%   | 6          | -7.0%  |
| 28.8k     | 1          | 8.5%   | 3            | 0.0%   | 3          | 8.5%   |
| 38.4k     | 1          | -18.6% | 2            | 0.0%   | 2          | 8.5%   |
| 57.6k     | -          | -      | 1            | -25.0% | 1          | 8.5%   |
| 76.8k     | -          | -      | 1            | 0.0%   | 1          | -18.6% |
| 115.2k    | -          | -      | -            | -      | -          | -      |
| 230.4k    | -          | -      | -            | -      | -          | -      |

(continued)

| Baud Rate | fx=3.6864MHz |       | fx=4.00MHz |       | fx=7.3728MHz |       |
|-----------|--------------|-------|------------|-------|--------------|-------|
|           | UARTBD       | ERROR | UARTBD     | ERROR | UARTBD       | ERROR |
| 2400      | 95           | 0.0%  | 103        | 0.2%  | 191          | 0.0%  |
| 4800      | 47           | 0.0%  | 51         | 0.2%  | 95           | 0.0%  |
| 9600      | 23           | 0.0%  | 25         | 0.2%  | 47           | 0.0%  |
| 14.4k     | 15           | 0.0%  | 16         | 2.1%  | 31           | 0.0%  |
| 19.2k     | 11           | 0.0%  | 12         | 0.2%  | 23           | 0.0%  |
| 28.8k     | 7            | 0.0%  | 8          | -3.5% | 15           | 0.0%  |
| 38.4k     | 5            | 0.0%  | 6          | -7.0% | 11           | 0.0%  |
| 57.6k     | 3            | 0.0%  | 3          | 8.5%  | 7            | 0.0%  |
| 76.8k     | 2            | 0.0%  | 2          | 8.5%  | 5            | 0.0%  |
| 115.2k    | 1            | 0.0%  | 1          | 8.5%  | 3            | 0.0%  |
| 230.4k    | -            | -     | -          | -     | 1            | 0.0%  |
| 250k      | -            | -     | -          | -     | 1            | -7.8% |
| 0.5M      | -            | -     | -          | -     | -            | -     |

(continued)

| Baud Rate | fx=8.00MHz |       | fx=11.0592MHz |       |
|-----------|------------|-------|---------------|-------|
|           | UARTBD     | ERROR | UARTBD        | ERROR |
| 2400      | 207        | 0.2%  | -             | -     |
| 4800      | 103        | 0.2%  | 143           | 0.0%  |
| 9600      | 51         | 0.2%  | 71            | 0.0%  |
| 14.4k     | 34         | -0.8% | 47            | 0.0%  |
| 19.2k     | 25         | 0.2%  | 35            | 0.0%  |
| 28.8k     | 16         | 2.1%  | 23            | 0.0%  |
| 38.4k     | 12         | 0.2%  | 17            | 0.0%  |
| 57.6k     | 8          | -3.5% | 11            | 0.0%  |
| 76.8k     | 6          | -7.0% | 8             | 0.0%  |
| 115.2k    | 3          | 8.5%  | 5             | 0.0%  |
| 230.4k    | 1          | 8.5%  | 2             | 0.0%  |
| 250k      | 1          | 0.0%  | 2             | -7.8% |
| 0.5M      | -          | -     | -             | -     |
| 1M        | -          | -     | -             | -     |



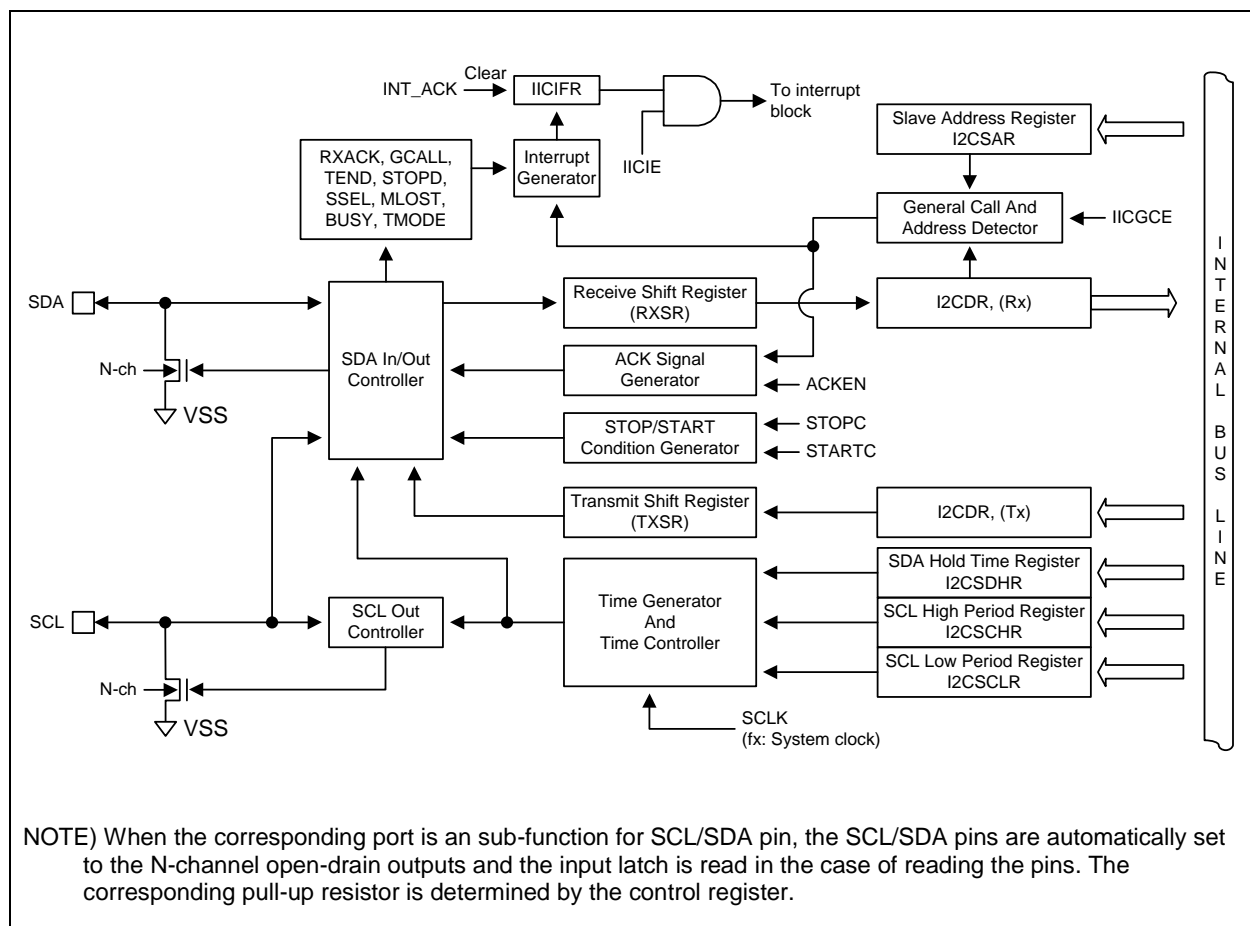
## 11.12 I2C

### 11.12.1 Overview

The I2C is one of industrial standard serial communication protocols, and which uses 2 bus lines Serial Data Line (SDA) and Serial Clock Line (SCL) to exchange data. Because both SDA and SCL lines are open-drain output, each line needs pull-up resistor. The features are as shown below.

- Compatible with I2C bus standard
- Multi-master operation
- Up to 400kHz data transfer read speed
- 7 bit address
- Support two slave address
- Both master and slave operation
- Bus busy detection

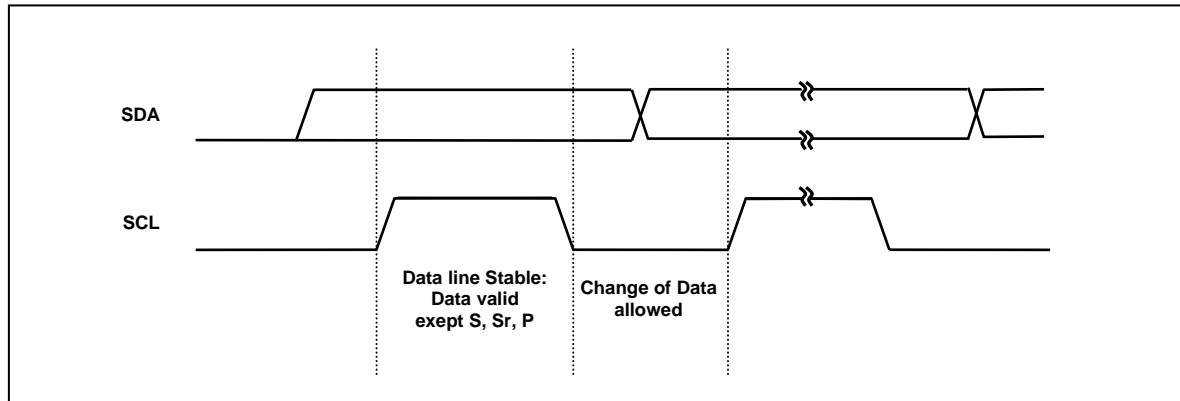
### 11.12.2 I2C Block Diagram



### Figure 11.47 I2C Block Diagram

### 11.12.3 I2C Bit Transfer

The data on the SDA line must be stable during HIGH period of the clock, SCL. The HIGH or LOW state of the data line can only change when the clock signal on the SCL line is LOW. The exceptions are START(S), repeated START(Sr) and STOP(P) condition where data line changes when clock line is high.



### Figure 11.48 Bit Transfer on the I2C-Bus

### 11.12.4 Start / Repeated Start / Stop

One master can issue a START (S) condition to notice other devices connected to the SCL, SDA lines that it will use the bus. A STOP (P) condition is generated by the master to release the bus lines so that other devices can use it.

A high to low transition on the SDA line while SCL is high defines a START (S) condition.

A low to high transition on the SDA line while SCL is high defines a STOP (P) condition.

START and STOP conditions are always generated by the master. The bus is considered to be busy after START condition. The bus is considered to be free again after STOP condition, ie, the bus is busy between START and STOP condition. If a repeated START condition (Sr) is generated instead of STOP condition, the bus stays busy. So, the START and repeated START conditions are functionally identical.

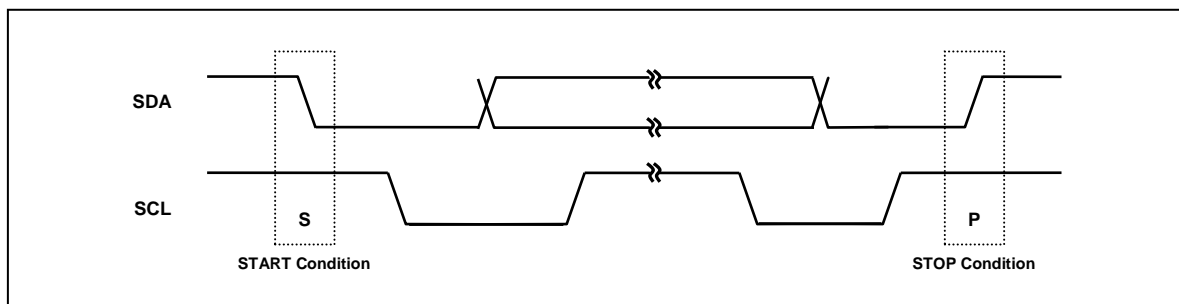


Figure 11.49 START and STOP Condition

### 11.12.5 Data Transfer

Every byte put on the SDA line must be 8-bits long. The number of bytes that can be transmitted per transfer is unlimited. Each byte has to be followed by an acknowledge bit. Data is transferred with the most significant bit (MSB) first. If a slave can't receive or transmit another complete byte of data until it has performed some other function, it can hold the clock line SCL LOW to force the master into a wait state. Data transfer then continues when the slave is ready for another byte of data and releases clock line SCL.

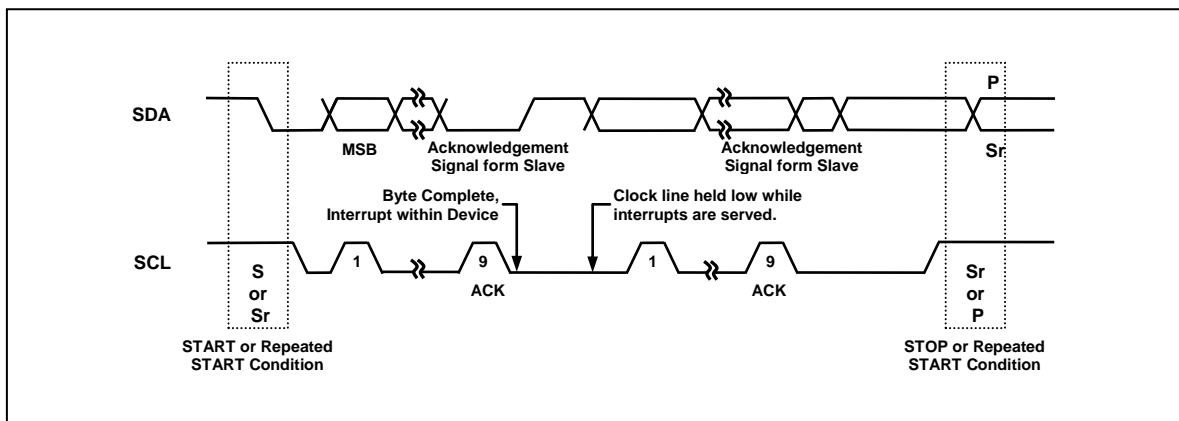


Figure 11.50 Data Transfer on the I2C-Bus

### 11.12.6 I2C Acknowledge

The acknowledge related clock pulse is generated by the master. The transmitter releases the SDA line (HIGH) during the acknowledge clock pulse. The receiver must pull down the SDA line during the acknowledge clock pulse so that it remains stable LOW during the HIGH period of this clock pulse. When a slave is addressed by a master (Address Packet), and if it is unable to receive or transmit because it's performing some real time function, the data line must be left HIGH by the slave. And also, when a slave addressed by a master is unable to receive more data bits, the slave receiver must release the SDA line (Data Packet). The master can then generate either a STOP condition to abort the transfer, or a repeated START condition to start a new transfer.

If a master receiver is involved in a transfer, it must signal the end of data to the slave transmitter by not generating an acknowledge on the last byte that was clocked out of the slave. The slave transmitter must release the data line to allow the master to generate a STOP or repeated START condition.

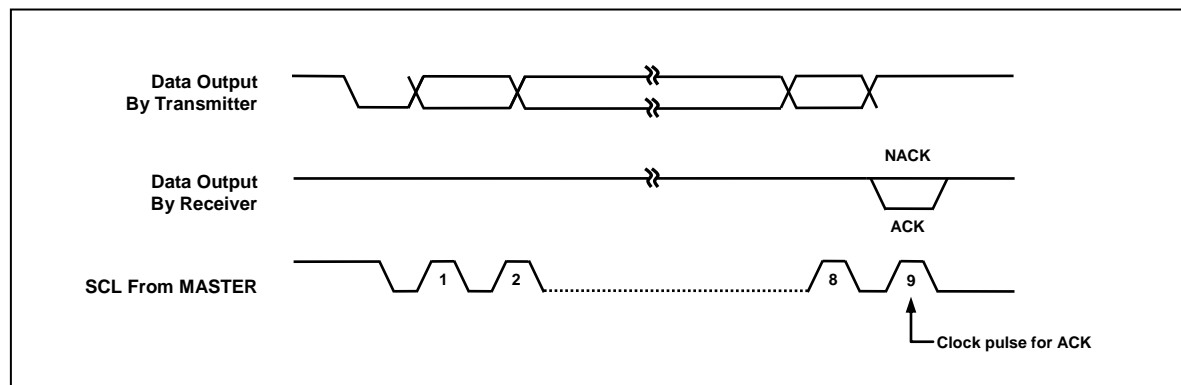


Figure 11.51 Acknowledge on the I2C-Bus

### 11.12.7 Synchronization / Arbitration

Clock synchronization is performed using the wired-AND connection of I2C interfaces to the SCL line. This means that a HIGH to LOW transition on the SCL line will cause the devices concerned to start counting off their LOW period and it will hold the SCL line in that state until the clock HIGH state is reached. However the LOW to HIGH transition of this clock may not change the state of the SCL line if another clock is still within its LOW period. In this way, a synchronized SCL clock is generated with its LOW period determined by the device with the longest clock LOW period, and its HIGH period determined by the one with the shortest clock HIGH period.

A master may start a transfer only if the bus is free. Two or more masters may generate a START condition. Arbitration takes place on the SDA line, while the SCL line is at the HIGH level, in such a way that the master which transmits a HIGH level, while another master is transmitting a LOW level will switch off its DATA output state because the level on the bus doesn't correspond to its own level. Arbitration continues for many bits until a winning master gets the ownership of I2C bus. Its first stage is comparison of the address bits.

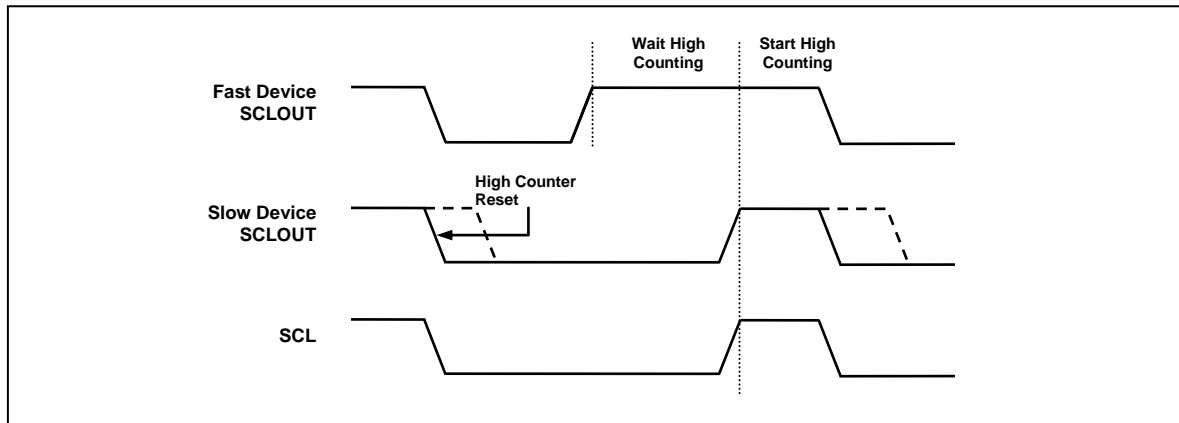


Figure 11.52 Clock Synchronization during Arbitration Procedure

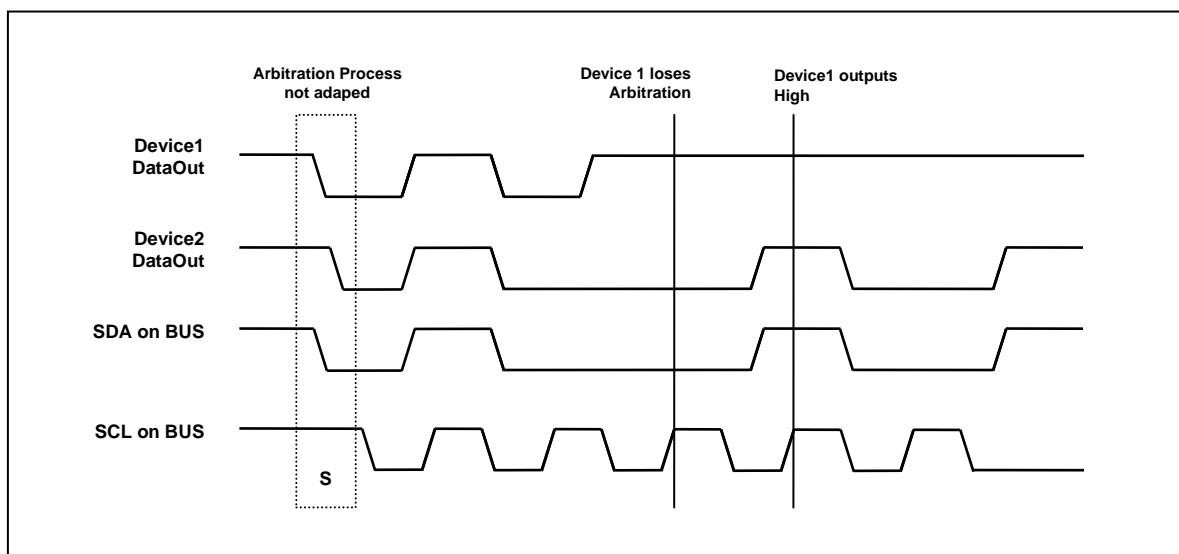


Figure 11.53 Arbitration Procedure of Two Masters

### 11.12.8 Operation

The I2C is byte-oriented and interrupt based. Interrupts are issued after all bus events except for a transmission of a START condition. Because the I2C is interrupt based, the application software is free to carry on other operations during a I2C byte transfer.

Note that when a I2C interrupt is generated, IICIFR flag in IIFLAG register is set, it is cleared by writing an any value to I2CSR. When I2C interrupt occurs, the SCL line is hold LOW until writing any value to I2CSR. When the IICIFR flag is set, the I2CSR contains a value indicating the current state of the I2C bus. According to the value in I2CSR, software can decide what to do next.

I2C can operate in 4 modes by configuring master/slave, transmitter/receiver. The operating mode is configured by a winning master. A more detailed explanation follows below.

### 11.12.8.1 Master Transmitter

To operate I2C in master transmitter, follow the recommended steps below.

7. Enable I2C by setting IICEN bit in I2CCR. This provides main clock to the peripheral.
8. Load SLA+W into the I2CDR where SLA is address of slave device and W is transfer direction from the viewpoint of the master. For master transmitter, W is '0'. Note that I2CDR is used for both address and data.
9. Configure baud rate by writing desired value to both I2CSCLR and I2CSCHR for the Low and High period of SCL line.
10. Configure the I2CSDHR to decide when SDA changes value from falling edge of SCL. If SDA should change in the middle of SCL LOW period, load half the value of I2CSCLR to the I2CSDHR.
11. Set the STARTC bit in I2CCR. This transmits a START condition. And also configure how to handle interrupt and ACK signal. When the STARTC bit is set, 8-bit data in I2CDR is transmitted out according to the baud-rate.
12. This is ACK signal processing stage for address packet transmitted by master. When 7-bit address and 1-bit transfer direction is transmitted to target slave device, the master can know whether the slave acknowledged or not in the 9<sup>th</sup> high period of SCL. If the master gains bus mastership, I2C generates GCALL interrupt regardless of the reception of ACK from the slave device. When I2C loses bus mastership during arbitration process, the MLOST bit in I2CSR is set, and I2C waits in idle state or can be operate as an addressed slave. To operate as a slave when the MLOST bit in I2CSR is set, the ACKEN bit in I2CCR must be set and the received 7-bit address must equal to the SLA bits in I2CSAR. In this case I2C operates as a slave transmitter or a slave receiver (go to appropriate section). In this stage, I2C holds the SCL LOW. This is because to decide whether I2C continues serial transfer or stops communication. The following steps continue assuming that I2C does not lose mastership during first data transfer.

I2C (Master) can choose one of the following cases regardless of the reception of ACK signal from slave.

- 1) Master receives ACK signal from slave, so continues data transfer because slave can receive more data from master. In this case, load data to transmit to I2CDR.
- 2) Master stops data transfer even if it receives ACK signal from slave. In this case, set the STOPC bit in I2CCR.
- 3) Master transmits repeated START condition with not checking ACK signal. In this case, load SLA+R/W into the I2CDR and set STARTC bit in I2CCR.

After doing one of the actions above, write any arbitrary to I2CSR to release SCL line. In case of 1), move to step 7. In case of 2), move to step 9 to handle STOP interrupt. In case of 3), move to step 6 after transmitting the data in I2CDR and if transfer direction bit is '1' go to master receiver section.

13. 1-Byte of data is being transmitted. During data transfer, bus arbitration continues.
14. This is ACK signal processing stage for data packet transmitted by master. I2C holds the SCL LOW. When I2C loses bus mastership while transmitting data arbitrating other masters, the MLOST bit in I2CSR is set. If then, I2C waits in idle state. When the data in I2CDR is transmitted completely, I2C generates TEND interrupt.

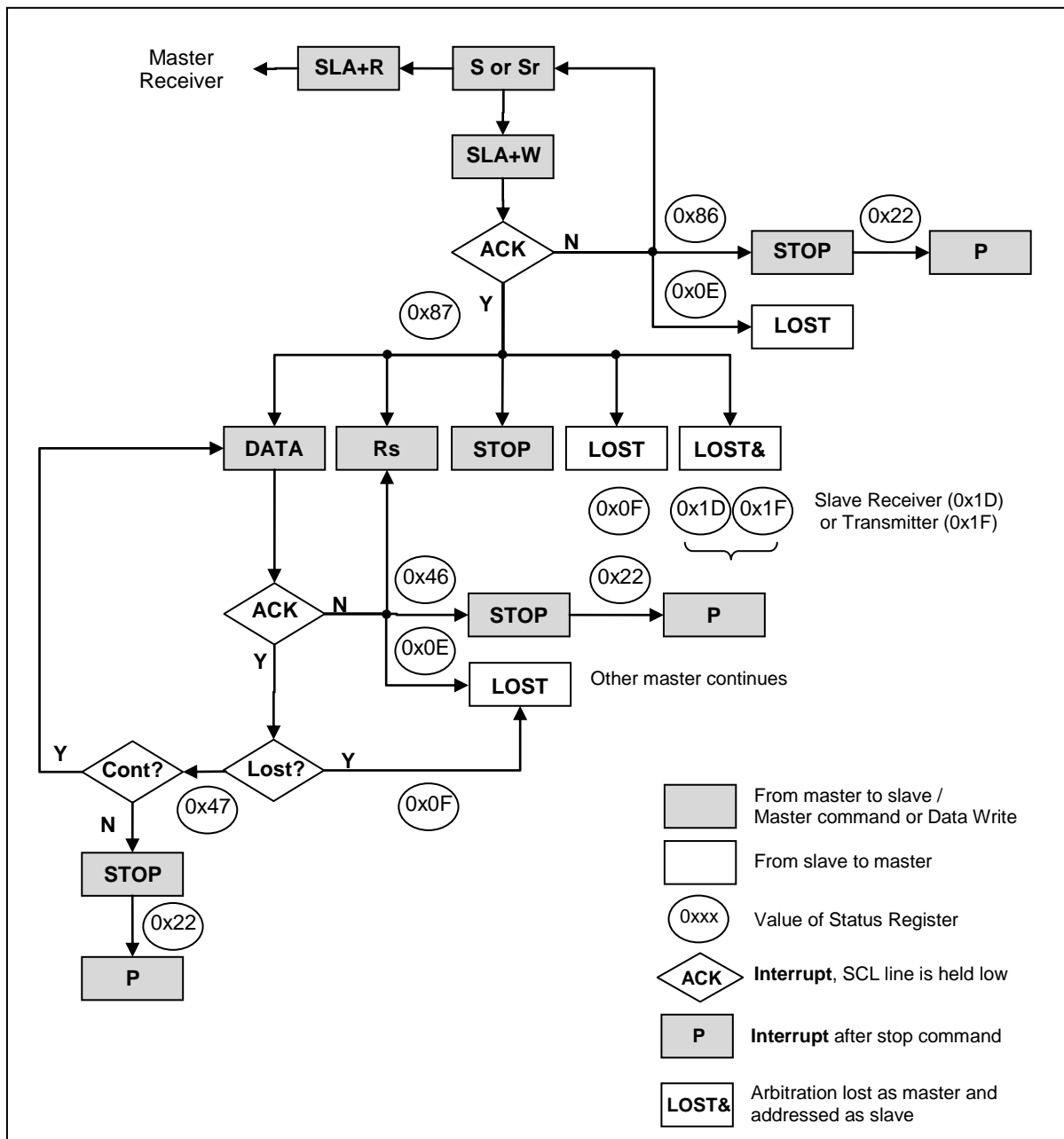
I2C can choose one of the following cases regardless of the reception of ACK signal from slave.

- 1) Master receives ACK signal from slave, so continues data transfer because slave can receive more data from master. In this case, load data to transmit to I2CDR.
- 2) Master stops data transfer even if it receives ACK signal from slave. In this case, set the STOPC bit in I2CCR.
- 3) Master transmits repeated START condition with not checking ACK signal. In this case, load SLA+R/W into the I2CDR and set the STARTC bit in I2CCR.

After doing one of the actions above, write any arbitrary to I2CSR to release SCL line. In case of 1), move to step 7. In case of 2), move to step 9 to handle STOP interrupt. In case of 3), move to step 6 after transmitting the data in I2CDR, and if transfer direction bit is '1' go to master receiver section.

15. This is the final step for master transmitter function of I2C, handling STOP interrupt. The STOP bit indicates that data transfer between master and slave is over. To clear I2CSR, write any value to I2CSR. After this, I2C enters idle state.

The next figure depicts above process for master transmitter operation of I2C.



### Figure 11.54 Formats and States in the Master Transmitter Mode

### 11.12.8.2 Master Receiver

To operate I2C in master receiver, follow the recommended steps below.

1. Enable I2C by setting IICEN bit in I2CCR. This provides main clock to the peripheral.
2. Load SLA+R into the I2CDR where SLA is address of slave device and R is transfer direction from the viewpoint of the master. For master receiver, R is '1'. Note that I2CDR is used for both address and data.
3. Configure baud rate by writing desired value to both I2CSCLR and I2CSCHR for the Low and High period of SCL line.
4. Configure the I2CSDHR to decide when SDA changes value from falling edge of SCL. If SDA should change in the middle of SCL LOW period, load half the value of I2CSCLR to the I2CSDHR.
5. Set the STARTC bit in I2CCR. This transmits a START condition. And also configure how to handle interrupt and ACK signal. When the STARTC bit is set, 8-bit data in I2CDR is transmitted out according to the baud-rate.
6. This is ACK signal processing stage for address packet transmitted by master. When 7-bit address and 1-bit transfer direction is transmitted to target slave device, the master can know whether the slave acknowledged or not in the 9<sup>th</sup> high period of SCL. If the master gains bus mastership, I2C generates GCALL interrupt regardless of the reception of ACK from the slave device. When I2C loses bus mastership during arbitration process, the MLOST bit in I2CSR is set, and I2C waits in idle state or can be operate as an addressed slave. To operate as a slave when the MLOST bit in I2CSR is set, the ACKEN bit in I2CCR must be set and the received 7-bit address must equal to the SLA bits in I2CSAR. In this case I2C operates as a slave transmitter or a slave receiver (go to appropriate section). In this stage, I2C holds the SCL LOW. This is because to decide whether I2C continues serial transfer or stops communication. The following steps continue assuming that I2C does not lose mastership during first data transfer.

I2C (Master) can choose one of the following cases according to the reception of ACK signal from slave.

- 1) Master receives ACK signal from slave, so continues data transfer because slave can prepare and transmit more data to master. Configure ACKEN bit in I2CCR to decide whether I2C ACKnowledges the next data to be received or not.
- 2) Master stops data transfer because it receives no ACK signal from slave. In this case, set the STOPC bit in I2CCR.
- 3) Master transmits repeated START condition due to no ACK signal from slave. In this case, load SLA+R/W into the I2CDR and set STARTC bit in I2CCR.

After doing one of the actions above, write arbitrary value to I2CSR to release SCL line. In case of 1), move to step 7. In case of 2), move to step 9 to handle STOP interrupt. In case of 3), move to step 6 after transmitting the data in I2CDR and if transfer direction bit is '0' go to master transmitter section.

7. 1-Byte of data is being received.
8. This is ACK signal processing stage for data packet transmitted by slave. I2C holds the SCL LOW. When 1-Byte of data is received completely, I2C generates TEND interrupt.

I2C0 can choose one of the following cases according to the RXACK flag in I2CSR.

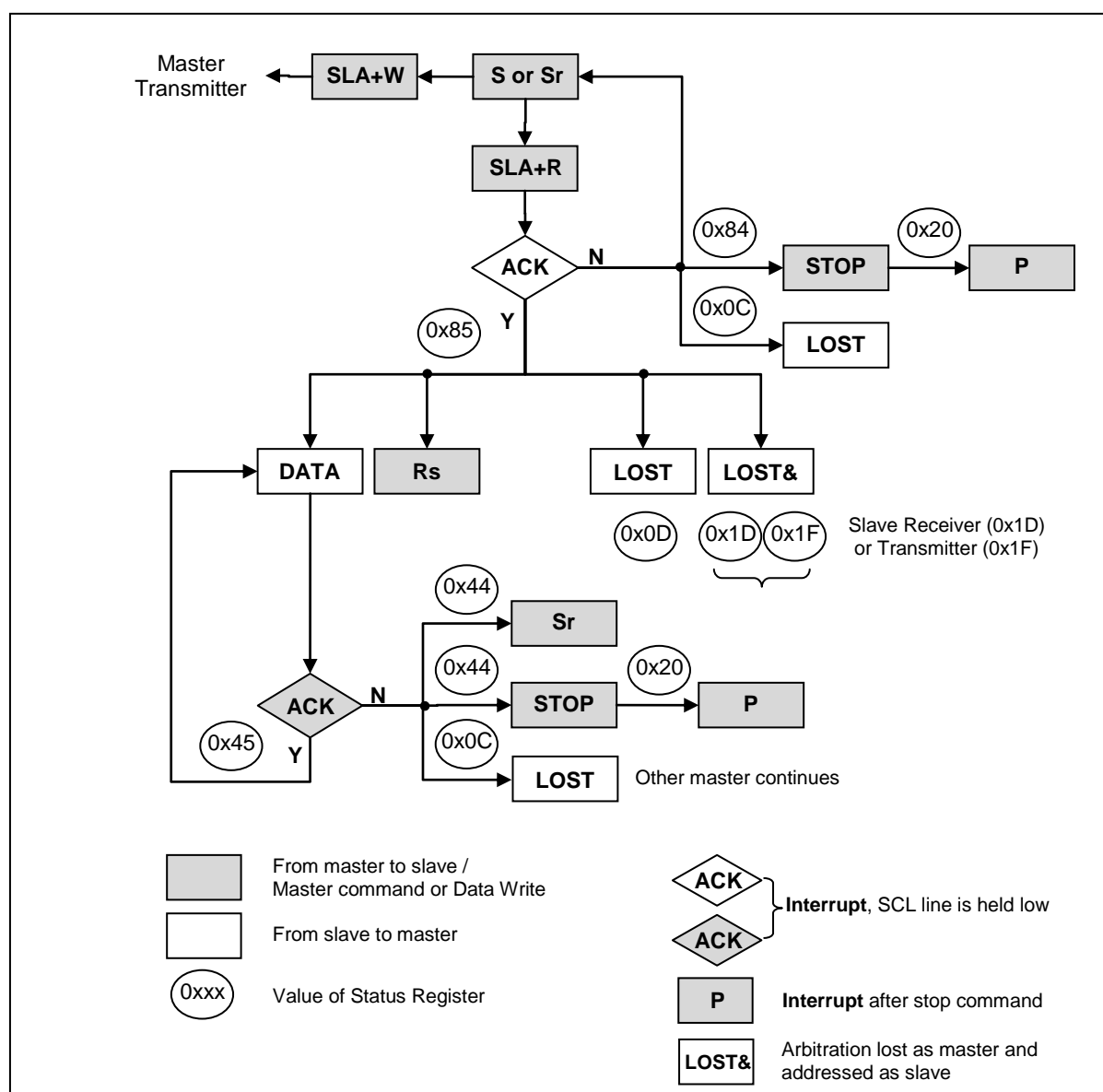
- 1) Master continues receiving data from slave. To do this, set ACKEN bit in I2CCR to ACKnowledge the next data to be received.
- 2) Master wants to terminate data transfer when it receives next data by not generating ACK signal. This can be done by clearing ACKEN bit in I2CCR.
- 3) Because no ACK signal is detected, master terminates data transfer. In this case, set the STOPC bit in I2CCR.
- 4) No ACK signal is detected, and master transmits repeated START condition. In this case, load SLA+R/W into the I2CDR and set the STARTC bit in I2CCR.

After doing one of the actions above, write arbitrary value to I2CSR to release SCL line. In case of 1) and 2), move to step 7. In case of 3), move to step 9 to handle STOP interrupt. In case of 4), move to step 6 after transmitting the data in I2CDR, and if transfer direction bit is '0' go to master transmitter section.

9. This is the final step for master receiver function of I2C, handling STOP interrupt. The STOP bit indicates that data transfer between master and slave is over. To clear I2CSR, write any value to I2CSR. After this, I2C enters idle state.



The processes described above for master receiver operation of I2C can be depicted as the following figure.



**Figure 11.55 Formats and States in the Master Receiver Mode**

### 11.12.8.3 I2C Slave Transmitter

To operate I2C in slave transmitter, follow the recommended steps below.

1. If the main operating clock (SCLK) of the system is slower than that of SCL, load value 0x00 into I2CSDHR to make SDA change within one system clock period from the falling edge of SCL. Note that the hold time of SDA is calculated by SDAH x period of SCLK where SDAH is multiple of number of SCLK coming from I2CSDHR. When the hold time of SDA is longer than the period of SCLK, I2C (slave) cannot transmit serial data properly.
2. Enable I2C by setting IICIE bit and IICEN bit in I2CCR. This provides main clock to the peripheral.
3. When a START condition is detected, I2C receives one byte of data and compares it with SLA bits in I2CSAR. If the GCALLEN bit in I2CSAR is enabled, I2C compares the received data with value 0x00, the general call address.
4. If the received address does not equal to SLA bits in I2CSAR, I2C enters idle state ie, waits for another START condition. Else if the address equals to SLA bits and the ACKEN bit is enabled, I2C generates SSEL interrupt and the SCL line is held LOW. Note that even if the address equals to SLA bits, when the ACKEN bit is disabled, I2C enters idle state. When SSEL interrupt occurs, load transmit data to I2CDR and write arbitrary value to I2CSR to release SCL line.
5. 1-Byte of data is being transmitted.
6. In this step, I2C generates TEND interrupt and holds the SCL line LOW regardless of the reception of ACK signal from master. Slave can select one of the following cases.
  - 1) No ACK signal is detected and I2C waits STOP or repeated START condition.
  - 2) ACK signal from master is detected. Load data to transmit into I2CDR.

After doing one of the actions above, write arbitrary value to I2CSR to release SCL line. In case of 1) move to step 7 to terminate communication. In case of 2) move to step 5. In either case, a repeated START condition can be detected. For that case, move step 4.

7. This is the final step for slave transmitter function of I2C, handling STOP interrupt. The STOPC bit indicates that data transfer between master and slave is over. To clear I2CSR, write any value to I2CSR. After this, I2C enters idle state.

The next figure shows flow chart for handling slave transmitter function of I2C.

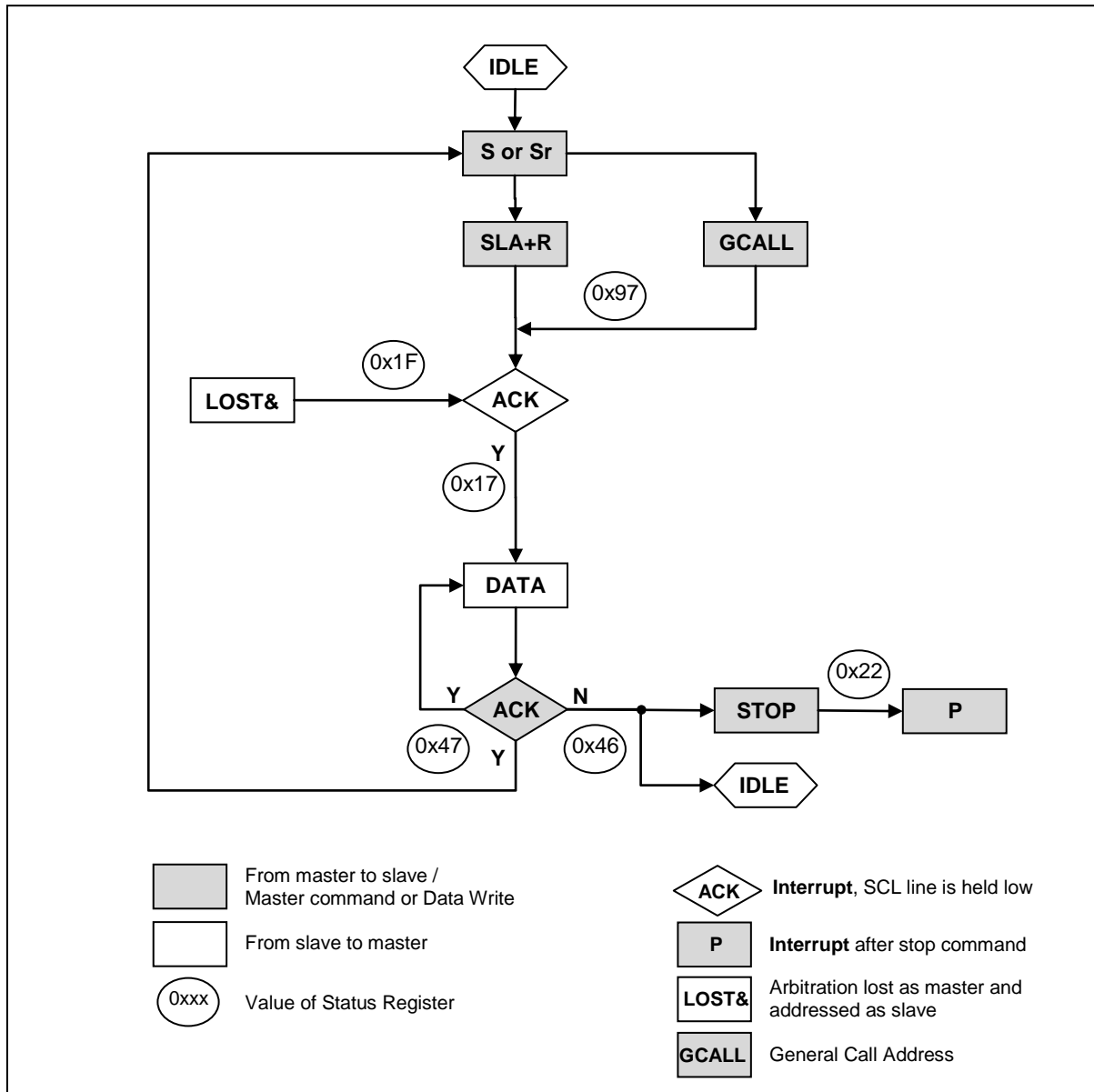


Figure 11.56 Formats and States in the Slave Transmitter Mode

#### 11.12.8.4 Slave Receiver

To operate I2C in slave receiver, follow the recommended steps below.

1. If the main operating clock (SCLK) of the system is slower than that of SCL, load value 0x00 into I2CSDHR to make SDA change within one system clock period from the falling edge of SCL. Note that the hold time of SDA is calculated by SDAH x period of SCLK where SDAH is multiple of number of SCLK coming from I2CSDHR. When the hold time of SDA is longer than the period of SCLK, I2C (slave) cannot transmit serial data properly.
2. Enable I2C by setting IICIE bit and USIEN bit in I2CCR. This provides main clock to the peripheral.
3. When a START condition is detected, I2C receives one byte of data and compares it with SLA bits in I2CSAR. If the GCALLEN bit in I2CSAR is enabled, I2C compares the received data with value 0x00, the general call address.
4. If the received address does not equal to SLA bits in I2CSAR, I2C enters idle state ie, waits for another START condition. Else if the address equals to SLA bits and the ACKEN bit is enabled, I2C generates SSEL interrupt and the SCL line is held LOW. Note that even if the address equals to SLA bits, when the ACKEN bit is disabled, I2C enters idle state. When SSEL interrupt occurs and I2C is ready to receive data, write arbitrary value to I2CSR to release SCL line.
5. 1-Byte of data is being received.
6. In this step, I2C generates TEND interrupt and holds the SCL line LOW regardless of the reception of ACK signal from master. Slave can select one of the following cases.
  - 1) No ACK signal is detected (ACKEN=0) and I2C waits STOP or repeated START condition.
  - 2) ACK signal is detected (ACKEN=1) and I2C can continue to receive data from master.

After doing one of the actions above, write arbitrary value to I2CSR to release SCL line. In case of 1) move to step 7 to terminate communication. In case of 2) move to step 5. In either case, a repeated START condition can be detected. For that case, move step 4.

7. This is the final step for slave receiver function of I2C, handling STOP interrupt. The STOPC bit indicates that data transfer between master and slave is over. To clear I2CSR, write any value to I2CSR. After this, I2C enters idle state.

The process can be depicted as following figure when I2C operates in slave receiver mode.

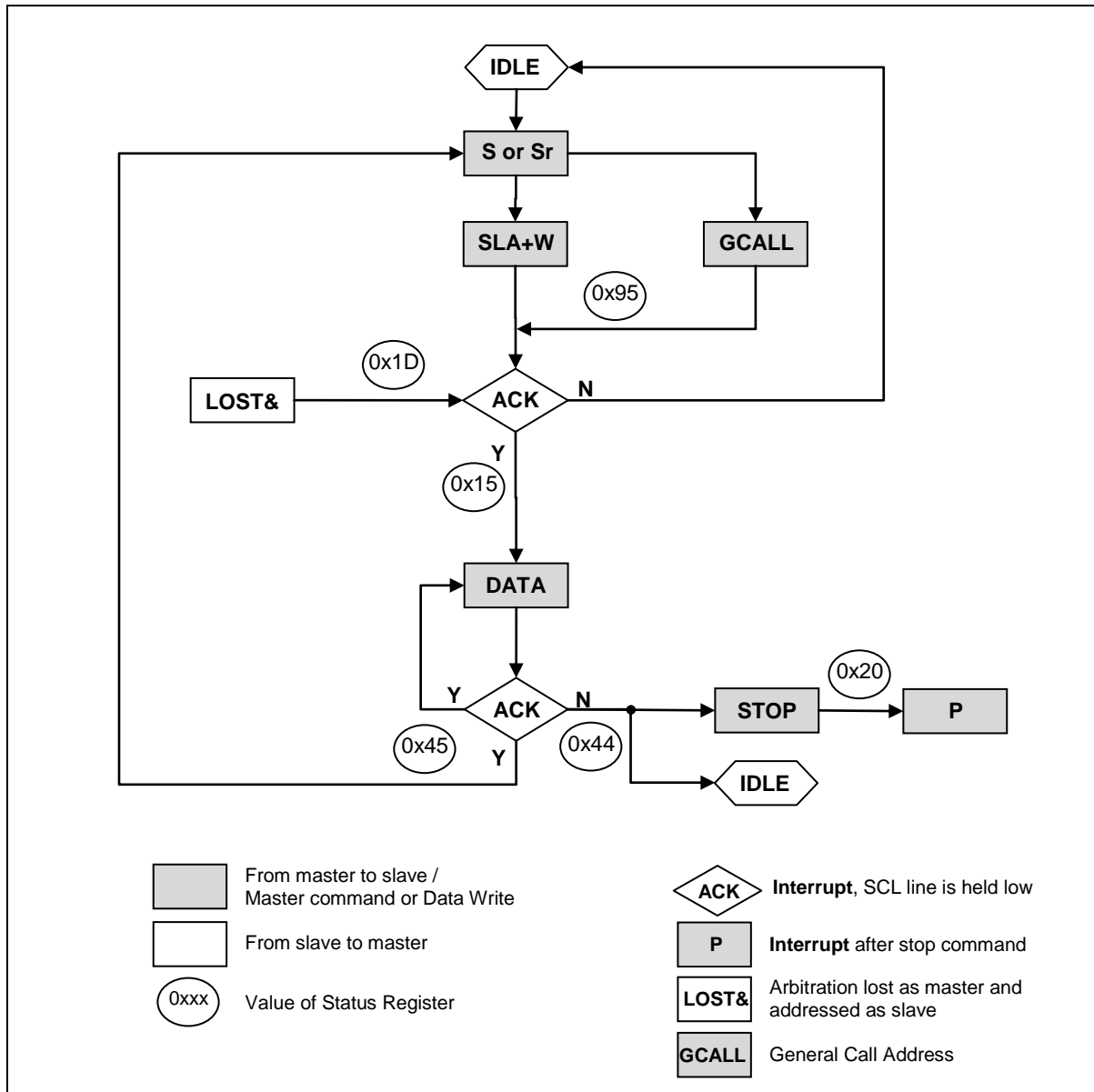


Figure 11.57 Formats and States in the Slave Receiver Mode

### 11.12.10 Register Map

Table 11-18 I2C Register Map

| Name    | Address | Dir | Default | Description                  |
|---------|---------|-----|---------|------------------------------|
| I2CCR   | E9H     | R/W | 00H     | I2C Control Register         |
| I2CSR   | EAH     | R/W | 00H     | I2C Status Register          |
| I2CSAR0 | EBH     | R/W | 00H     | I2C Slave Address 0 Register |
| I2CSAR1 | F1H     | R/W | 00H     | I2C Slave Address 1 Register |
| I2CDR   | ECH     | R/W | 00H     | I2C Data Register            |
| I2CSDHR | EDH     | R/W | 01H     | I2C SDA Hold Time Register   |
| I2CSCLR | EEH     | R/W | 3FH     | I2C SCL Low Period Register  |
| I2CSCHR | EFH     | R/W | 3FH     | I2C SCL High Period Register |

### 11.12.11 I2C Register Description

I2C module consists of I2C control register (I2CCR), I2C status register (I2CSR), I2C slave address 0/1 register (I2CSAR0/I2CSAR1), I2C data register (I2CDR), I2C SDA hold time register (I2CSDHR), I2C SCL low period register (I2CSCLR), and I2C SCL high period Register (I2CSCHR).

### 11.12.12 Register Description for I2C

#### I2CDR (I2C Data Register) : ECH

| 7      | 6      | 5      | 4      | 3      | 2      | 1      | 0      |
|--------|--------|--------|--------|--------|--------|--------|--------|
| I2CDR7 | I2CDR6 | I2CDR5 | I2CDR4 | I2CDR3 | I2CDR2 | I2CDR1 | I2CDR0 |
| R/W    | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    |

Initial value : 00H

I2CDR[7:0]

The I2CDR transmit buffer and receive buffer share the same I/O address with this DATA register. The transmit data buffer is the destination for data written to the I2CDR register. Reading the I2CDR register returns the contents of the receive buffer.

**I2CSDHR (I2C SDA Hold Time Register) : EDH**

| 7        | 6        | 5        | 4        | 3        | 2        | 1        | 0        |
|----------|----------|----------|----------|----------|----------|----------|----------|
| I2CSDHR7 | I2CSDHR6 | I2CSDHR5 | I2CSDHR4 | I2CSDHR3 | I2CSDHR2 | I2CSDHR1 | I2CSDHR0 |
| R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      |

Initial value : 01H

I2CSDHR[7:0]

The register is used to control SDA output timing from the falling edge of SCL.

Note that SDA is changed after  $t_{SCLK} \times (I2CSDHR+2)$ , in master mode, load half the value of I2CSCLR to this register to make SDA change in the middle of SCL.

In slave mode, configure this register regarding the frequency of SCL from master.

The SDA is changed after  $t_{sclk} \times (I2CSDHR+2)$  in master mode. So, to insure operation in slave mode, the value

$t_{SCLK} \times (I2CSDHR + 2)$  must be smaller than the period of SCL.

**I2CSCHR (I2C SCL High Period Register) : EFH**

| 7        | 6        | 5        | 4        | 3        | 2        | 1        | 0        |
|----------|----------|----------|----------|----------|----------|----------|----------|
| I2CSCHR7 | I2CSCHR6 | I2CSCHR5 | I2CSCHR4 | I2CSCHR3 | I2CSCHR2 | I2CSCHR1 | I2CSCHR0 |
| R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      |

Initial value : 3FH

I2CSCHR[7:0]

This register defines the high period of SCL in master mode.

The base clock is SCLK, the system clock, and the period is calculated by the formula:  $t_{SCLK} \times (4 \times I2CSCHR + 2)$  where

$t_{SCLK}$  is the period of SCLK.

**So, the operating frequency of I2C master mode is calculated by the following equation.**

$$f_{I2C} = \frac{1}{t_{SCLK} \times (4 \times (I2CSCLR + I2CSCHR) + 4)}$$

**I2CSCLR (I2C SCL Low Period Register) : EEH**

| 7        | 6        | 5        | 4        | 3        | 2        | 1        | 0        |
|----------|----------|----------|----------|----------|----------|----------|----------|
| I2CSCLR7 | I2CSCLR6 | I2CSCLR5 | I2CSCLR4 | I2CSCLR3 | I2CSCLR2 | I2CSCLR1 | I2CSCLR0 |
| R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      |

Initial value : 3FH

I2CSCLR[7:0]

This register defines the low period of SCL in master mode.

The base clock is SCLK, the system clock, and the period is calculated by the formula:  $t_{SCLK} \times (4 \times I2CSCLR + 2)$  where

$t_{SCLK}$  is the period of SCLK.

**I2CSAR0 (I2C Slave Address 0 Register) : EBH**

| 7        | 6        | 5        | 4        | 3        | 2        | 1        | 0        |
|----------|----------|----------|----------|----------|----------|----------|----------|
| I2CSLA06 | I2CSLA05 | I2CSLA04 | I2CSLA03 | I2CSLA02 | I2CSLA01 | I2CSLA00 | GCALL0EN |
| R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      |

Initial value : 00H

I2CSLA0[6:0]

These bits configure the slave address 0 in slave mode.

GCALL0EN

This bit decides whether I2C allows general call address or not in I2C slave mode.

0 Ignore general call address

1 Allow general call address

**I2CSAR1 (I2C Slave Address 1 Register) : F1H**

| 7        | 6        | 5        | 4        | 3        | 2        | 1        | 0        |
|----------|----------|----------|----------|----------|----------|----------|----------|
| I2CSLA16 | I2CSLA15 | I2CSLA14 | I2CSLA13 | I2CSLA12 | I2CSLA11 | I2CSLA10 | GCALL1EN |
| R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      | R/W      |

Initial value : 00H

I2CSLA1[6:0]

These bits configure the slave address 1 in slave mode.

GCALL1EN

This bit decides whether I2C allows general call address or not in I2C slave mode.

0 Ignore general call address

1 Allow general call address



**I2CCR (I2C Control Register) : E9H**

| 7      | 6     | 5        | 4     | 3     | 2       | 1     | 0      |
|--------|-------|----------|-------|-------|---------|-------|--------|
| IICRST | IICEN | TXDLYENB | IICIE | ACKEN | IMASTER | STOPC | STARTC |
| RW     | RW    | RW       | RW    | RW    | R       | RW    | RW     |

Initial value : 00H

|          |                                                                          |
|----------|--------------------------------------------------------------------------|
| IICRST   | Initialize Internal Registers of I2C.                                    |
| 0        | No effect                                                                |
| 1        | Initialize I2C, auto cleared                                             |
| IICEN    | Activate I2C Function Block by Supplying.                                |
| 0        | I2C is disabled                                                          |
| 1        | I2C is enabled                                                           |
| TXDLYENB | I2CSDHR register control bit                                             |
| 0        | Enable I2CSDHR register                                                  |
| 1        | Disable I2CSDHR register                                                 |
| IICIE    | Interrupt Enable bit                                                     |
| 0        | Interrupt from I2C is inhibited (use polling)                            |
| 1        | Enable interrupt for I2C                                                 |
| ACKEN    | Controls ACK signal Generation at ninth SCL period.                      |
| 0        | No ACK signal is generated (SDA = 1)                                     |
| 1        | ACK signal is generated (SDA = 0)                                        |
|          | NOTES) ACK signal is output (SDA =0) for the following 3 cases.          |
|          | 1. When received address packet equals to I2CSLA bits in I2CSAR.         |
|          | 2. When received address packet equals to value 0x00 with GCALL enabled. |
|          | 3. When I2C operates as a receiver (master or slave)                     |
| IMASTER  | Represent operating mode of I2C                                          |
| 0        | I2C is in slave mode                                                     |
| 1        | I2C is in master mode                                                    |
| STOPC    | When I2C is master, STOP condition generation                            |
| 0        | No effect                                                                |
| 1        | STOP condition is to be generated                                        |
| STARTC   | When I2C is master, START condition generation                           |
| 0        | No effect                                                                |
| 1        | START or repeated START condition is to be generated                     |

NOTE) Refer to the internal interrupt flag register (IIFLAG) for the I2C interrupt flags.

**I2CSR (I2C Status Register) : EAH**

| 7     | 6    | 5     | 4    | 3     | 2    | 1     | 0     |
|-------|------|-------|------|-------|------|-------|-------|
| GCALL | TEND | STOPD | SSEL | MLOST | BUSY | TMODE | RXACK |
| R     | RW   | RW    | RW   | RW    | RW   | RW    | RW    |

Initial value : 00H

|                         |                                                                                                                                                                          |                                                          |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------|
| GCALL <sup>(NOTE)</sup> | This bit has different meaning depending on whether I2C is master or slave. When I2C is a master, this bit represents whether it received AACK (address ACK) from slave. |                                                          |
|                         | 0                                                                                                                                                                        | No AACK is received (Master mode)                        |
|                         | 1                                                                                                                                                                        | AACK is received (Master mode)                           |
|                         | When I2C is a slave, this bit is used to indicated general call.                                                                                                         |                                                          |
|                         | 0                                                                                                                                                                        | General call address is not detected (Slave mode)        |
|                         | 1                                                                                                                                                                        | General call address is detected (Slave mode)            |
| TEND <sup>(NOTE)</sup>  | This bit is set when 1-byte of data is transferred completely                                                                                                            |                                                          |
|                         | 0                                                                                                                                                                        | 1 byte of data is not completely transferred             |
|                         | 1                                                                                                                                                                        | 1 byte of data is completely transferred                 |
| STOPD <sup>(NOTE)</sup> | This bit is set when a STOP condition is detected.                                                                                                                       |                                                          |
|                         | 0                                                                                                                                                                        | No STOP condition is detected                            |
|                         | 1                                                                                                                                                                        | STOP condition is detected                               |
| SSEL <sup>(NOTE)</sup>  | This bit is set when I2C is addressed by other master.                                                                                                                   |                                                          |
|                         | 0                                                                                                                                                                        | I2C is not selected as a slave                           |
|                         | 1                                                                                                                                                                        | I2C is addressed by other master and acts as a slave     |
| MLOST <sup>(NOTE)</sup> | This bit represents the result of bus arbitration in master mode.                                                                                                        |                                                          |
|                         | 0                                                                                                                                                                        | I2C maintains bus mastership                             |
|                         | 1                                                                                                                                                                        | I2C has lost bus mastership during arbitration process   |
| BUSY                    | This bit reflects bus status.                                                                                                                                            |                                                          |
|                         | 0                                                                                                                                                                        | I2C bus is idle, so a master can issue a START condition |
|                         | 1                                                                                                                                                                        | I2C bus is busy                                          |
| TMODE                   | This bit is used to indicate whether I2C is transmitter or receiver.                                                                                                     |                                                          |
|                         | 0                                                                                                                                                                        | I2C is a receiver                                        |
|                         | 1                                                                                                                                                                        | I2C is a transmitter                                     |
| RXACK                   | This bit shows the state of ACK signal                                                                                                                                   |                                                          |
|                         | 0                                                                                                                                                                        | No ACK is received                                       |
|                         | 1                                                                                                                                                                        | ACK is received at ninth SCL period                      |

NOTE) These bits can be source of interrupt.

When an I2C interrupt occurs except for STOP mode, the SCL line is hold LOW. To release SCL, write arbitrary value to I2CSR. When I2CSR is written, the TEND, STOPD, SSEL, MLOST, and RXACK bits are cleared.

## 12. Power Down Operation

### 12.1 Overview

The MC96F8316 has two power-down modes to minimize the power consumption of the device. In power down mode, power consumption is reduced considerably. The device provides three kinds of power saving functions, Main-IDLE, Sub-IDLE and STOP mode. In three modes, program is stopped.

### 12.2 Peripheral Operation in IDLE/STOP Mode

**Table 12-1 Peripheral Operation during Power Down Mode**

| Peripheral           | IDLE Mode                          | STOP Mode                                                                                                            |
|----------------------|------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| CPU                  | ALL CPU Operation are Disable      | ALL CPU Operation are Disable                                                                                        |
| RAM                  | Retain                             | Retain                                                                                                               |
| Basic Interval Timer | Operates Continuously              | Stop                                                                                                                 |
| Watch Dog Timer      | Operates Continuously              | Stop (Can be operated with WDTRC OSC)                                                                                |
| Watch Timer          | Operates Continuously              | Stop (Can be operated with sub clock)                                                                                |
| Timer0~2             | Operates Continuously              | Halted (Only when the Event Counter Mode is Enabled, Timer operates Normally)                                        |
| ADC                  | Operates Continuously              | Stop                                                                                                                 |
| BUZ                  | Operates Continuously              | Stop                                                                                                                 |
| SPI                  | Operates Continuously              | Only operate with external clock                                                                                     |
| UART                 | Operates Continuously              | Stop                                                                                                                 |
| I2C                  | Operates Continuously              | Only operate with external clock                                                                                     |
| Internal OSC (16MHz) | Oscillation                        | Stop when the system clock (fx) is f <sub>IRC</sub>                                                                  |
| WDTRC OSC (5kHz)     | Can be operated with setting value | Can be operated with setting value                                                                                   |
| Main OSC (0.4~12MHz) | Oscillation                        | Stop when fx = fx <sub>IN</sub>                                                                                      |
| Sub OSC (32.768kHz)  | Oscillation                        | Stop when fx = f <sub>SUB</sub>                                                                                      |
| I/O Port             | Retain                             | Retain                                                                                                               |
| Control Register     | Retain                             | Retain                                                                                                               |
| Address Data Bus     | Retain                             | Retain                                                                                                               |
| Release Method       | By RESET, all Interrupts           | By RESET, Timer Interrupt (EC0, EC1, EC2), SPI (External clock), External Interrupt, UART by RX, WT (sub clock), WDT |

### 12.3 IDLE Mode

The power control register is set to '01h' to enter the IDLE Mode. In this mode, the internal oscillation circuits remain active. Oscillation continues and peripherals are operated normally but CPU stops. It is released by reset or interrupt. To be released by interrupt, interrupt should be enabled before IDLE mode. If using reset, because the device becomes initialized state, the registers have reset value.

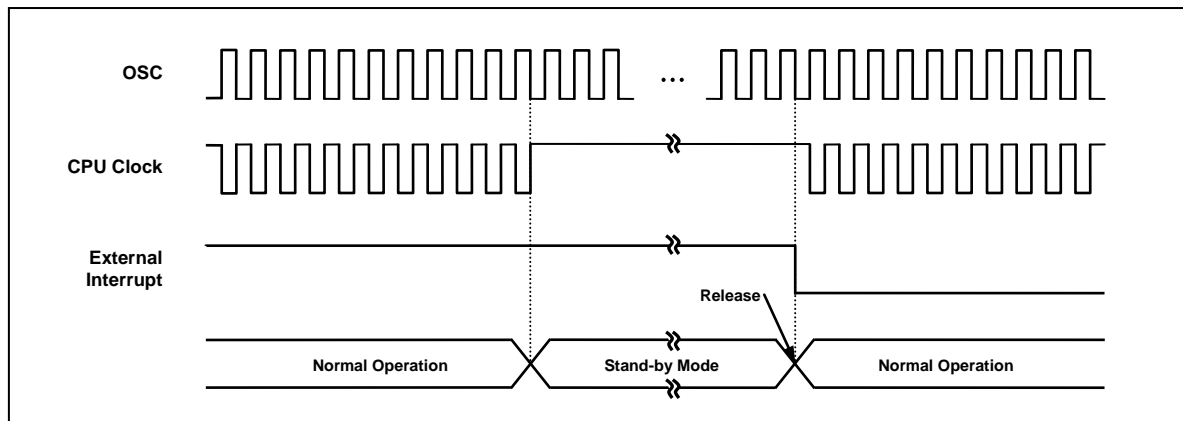


Figure 12.1 IDLE Mode Release Timing by External Interrupt

## 12.4 STOP Mode

The power control register is set to '03H' to enter the STOP Mode. In the stop mode, the selected oscillator, system clock and peripheral clock is stopped, but watch timer can be continued to operate with sub clock. With the clock frozen, all functions are stopped, but the on-chip RAM and control registers are held. For example, If the internal RC oscillator (f<sub>IRC</sub>) is selected for the system clock and the sub clock (f<sub>SUB</sub>) is oscillated, the internal RC oscillator stops oscillation and the sub clock is continuously oscillated in stop mode. At that time, the watch timer can be operated with the sub clock.

The source for exit from STOP mode is hardware reset and interrupts. The reset re-defines all the control registers.

When exit from STOP mode, enough oscillation stabilization time is required to normal operation. Figure 12.2 shows the timing diagram. When released from STOP mode, the Basic interval timer is activated on wake-up. Therefore, before STOP instruction, user must be set its relevant prescale divide ratio to have long enough time. This guarantees that oscillator has started and stabilized.

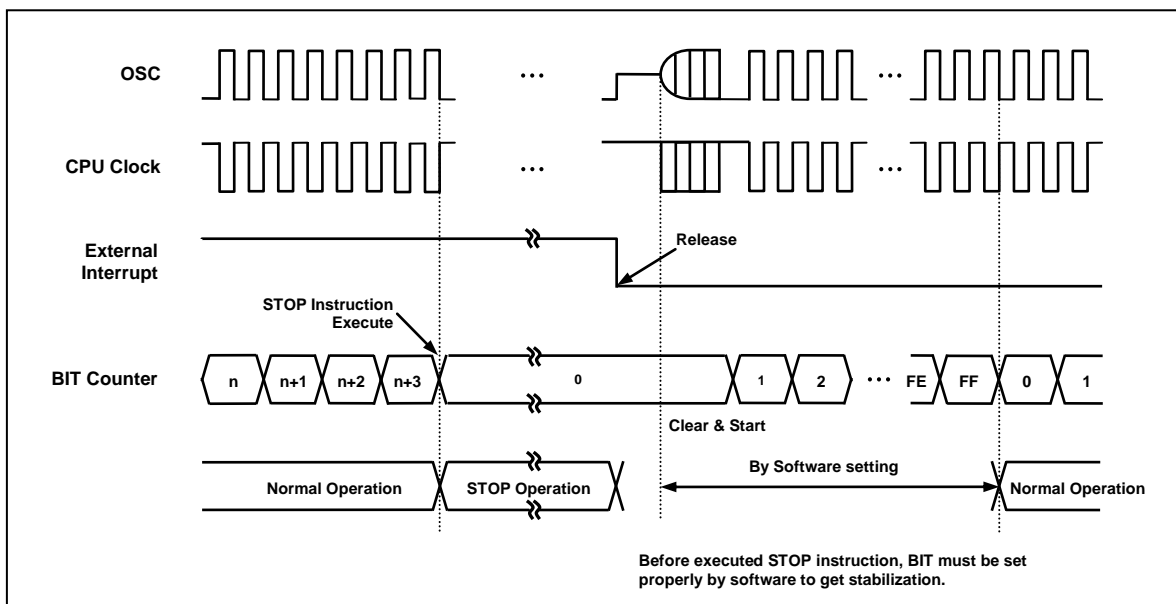


Figure 12.2 STOP Mode Release Timing by External Interrupt

## 12.5 Release Operation of STOP Mode

After STOP mode is released, the operation begins according to content of related interrupt register just before STOP mode start (Figure 12.3). If the global interrupt Enable Flag (IE.EA) is set to '1', the STOP mode is released by the interrupt which each interrupt enable flag = '1' and the CPU jumps to the relevant interrupt service routine. Even if the IE.EA bit is cleared to '0', the STOP mode is released by the interrupt of which the interrupt enable flag is set to '1'.

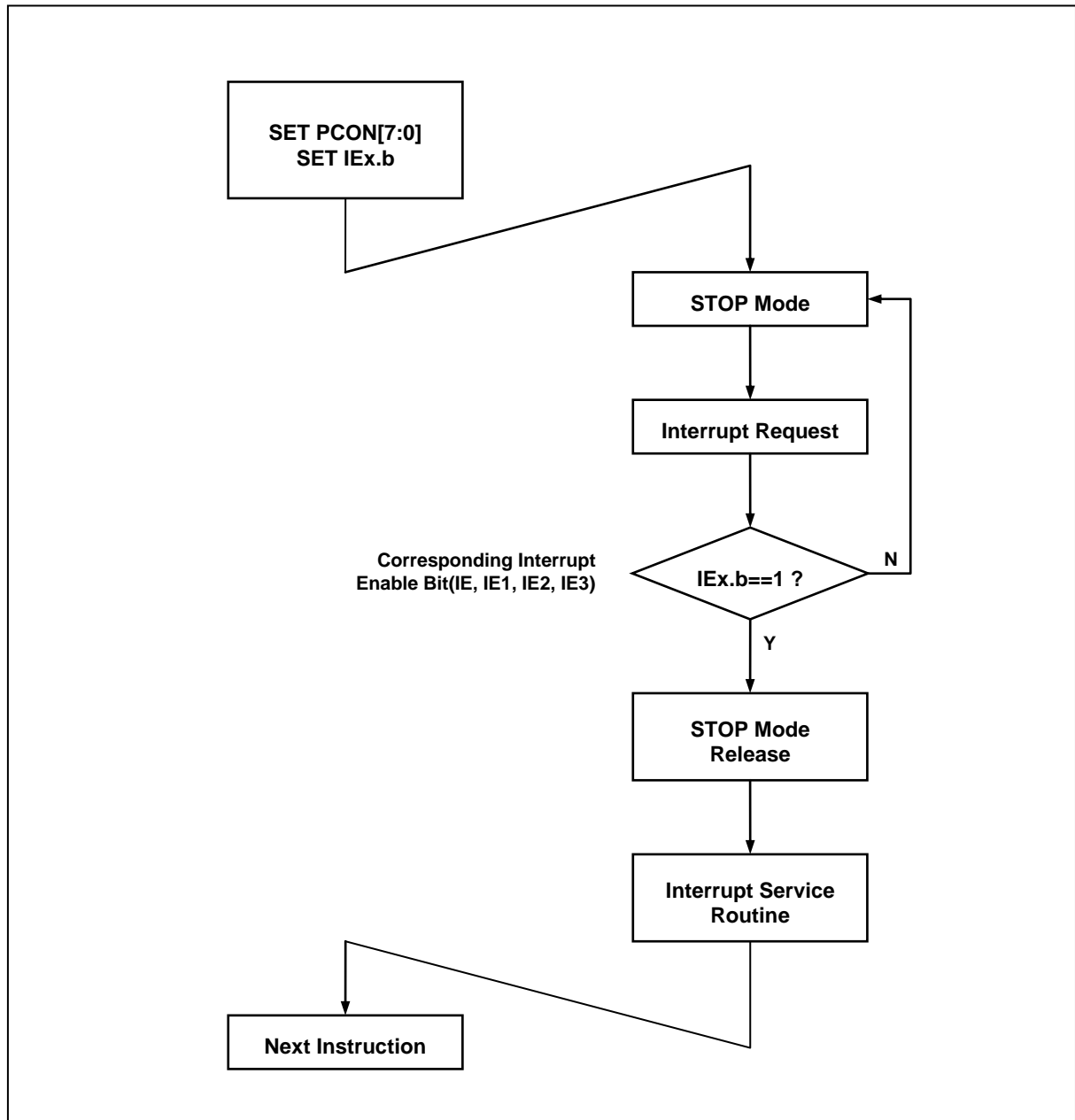


Figure 12.3 STOP Mode Release Flow

### 12.5.1 Register Map

Table 12-2 Power Down Operation Register Map

| Name | Address | Dir | Default | Description            |
|------|---------|-----|---------|------------------------|
| PCON | 87H     | R/W | 00H     | Power Control Register |

### 12.5.2 Power Down Operation Register Description

The power down operation register consists of the power control register (PCON).

### 12.5.3 Register Description for Power Down Operation

PCON (Power Control Register) : 87H

| 7     | 6 | 5 | 4 | 3     | 2     | 1     | 0     |
|-------|---|---|---|-------|-------|-------|-------|
| PCON7 | – | – | – | PCON3 | PCON2 | PCON1 | PCON0 |
| R/W   | – | – | – | R/W   | R/W   | R/W   | R/W   |

Initial value : 00H

PCON[7:0]      Power Control  
 01H            IDLE mode enable  
 03H            STOP mode enable  
 Other Values   Normal operation

NOTES) 1. To enter IDLE mode, PCON must be set to '01H'.

2. To enter STOP mode, PCON must be set to '03H'.

3. The PCON register is automatically cleared by a release signal in STOP/IDLE mode.

4. Three or more NOP instructions must immediately follow STOP/IDLE mode like the below examples.

Ex1)    MOV    PCON, #01H    ; IDLE mode  
       NOP  
       NOP  
       NOP  
       .  
       .  
       .

Ex2)    MOV    PCON, #03H    ; STOP mode  
       NOP  
       NOP  
       NOP  
       .  
       .  
       .

## 13. RESET

### 13.1 Overview

The following is the hardware setting value.

**Table 13-1 Reset State**

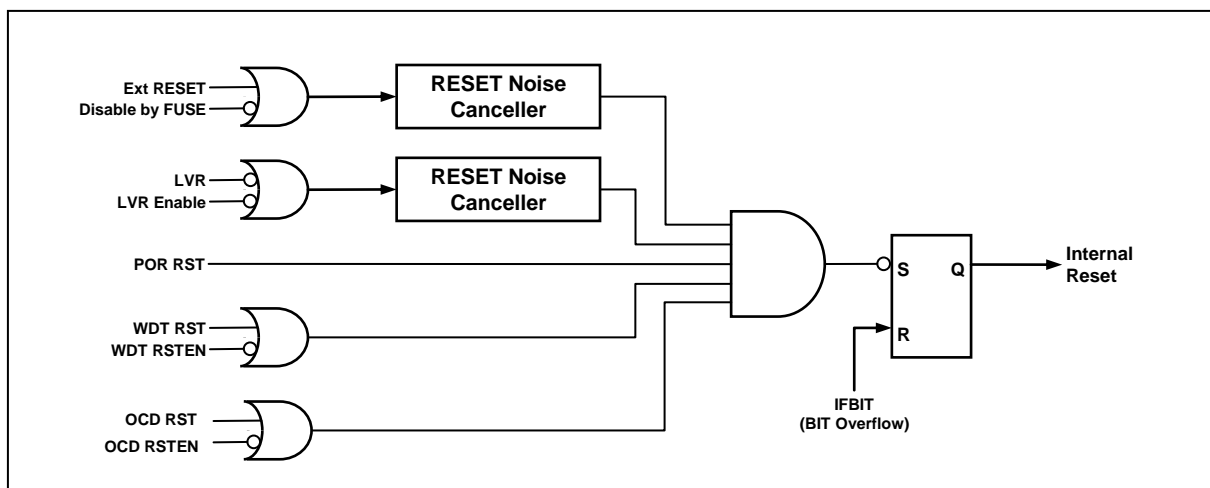
| On Chip Hardware     | Initial Value                     |
|----------------------|-----------------------------------|
| Program Counter (PC) | 0000h                             |
| Accumulator          | 00h                               |
| Stack Pointer (SP)   | 07h                               |
| Peripheral Clock     | On                                |
| Control Register     | Refer to the Peripheral Registers |

### 13.2 Reset Source

The MC96F8316 has five types of reset sources. The following is the reset sources.

- External RESETB
- Power ON RESET (POR)
- WDT Overflow Reset (In the case of WD TEN = `1`)
- Low Voltage Reset (In the case of LVREN = `0`)
- OCD Reset

### 13.3 RESET Block Diagram



**Figure 13.1 RESET Block Diagram**



### 13.4 RESET Noise Canceller

The Figure 13.2 is the noise canceller diagram for noise cancellation of RESET. It has the noise cancellation value of about 2 $\mu$ s (@V<sub>DD</sub>=5V) to the low input of system reset.

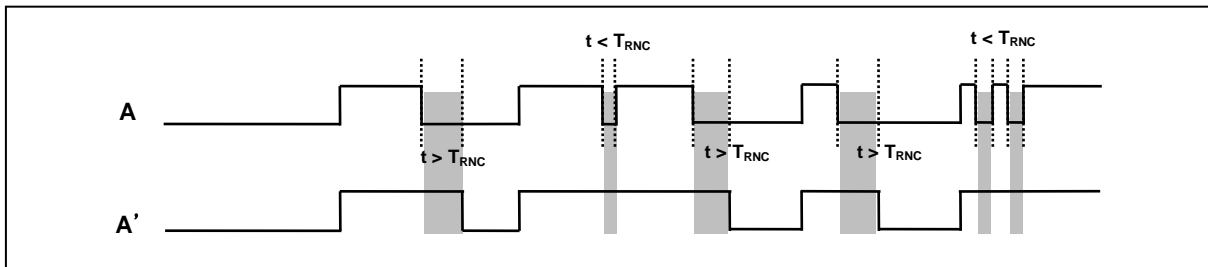


Figure 13.2 Reset noise canceller timer diagram

### 13.5 Power on RESET

When rising device power, the POR (Power On Reset) has a function to reset the device. If POR is used, it executes the device RESET function instead of the RESET IC or the RESET circuits.

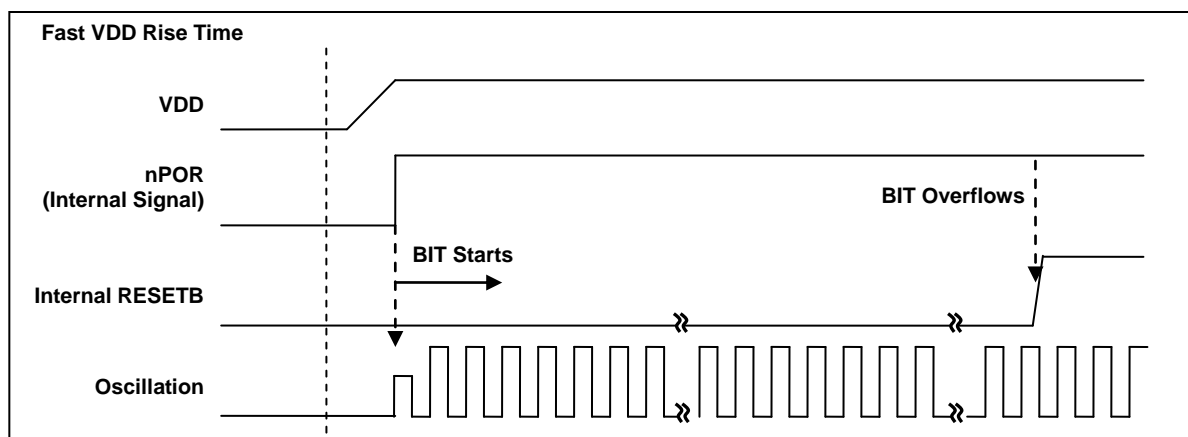


Figure 13.3 Fast VDD Rising Time

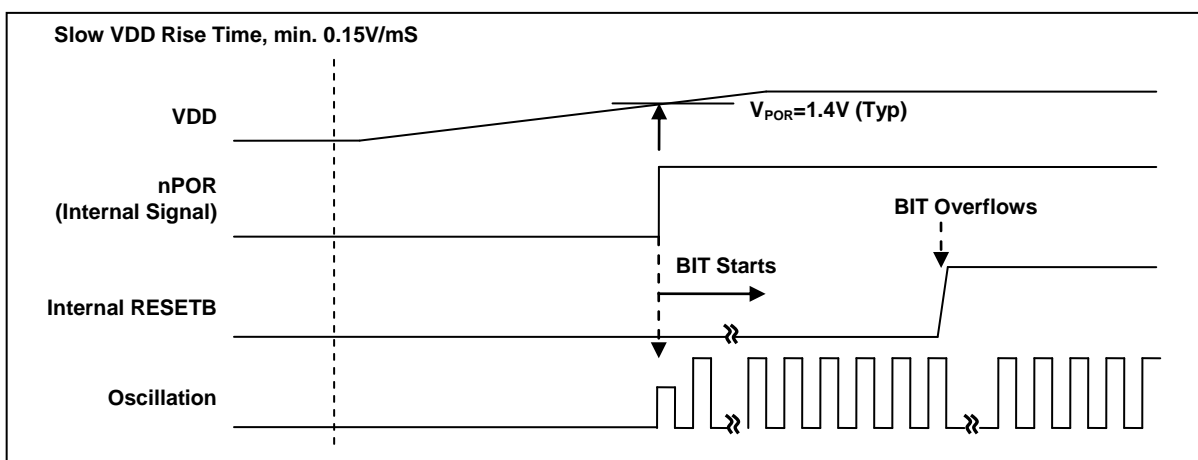


Figure 13.4 Internal RESET Release Timing On Power-Up

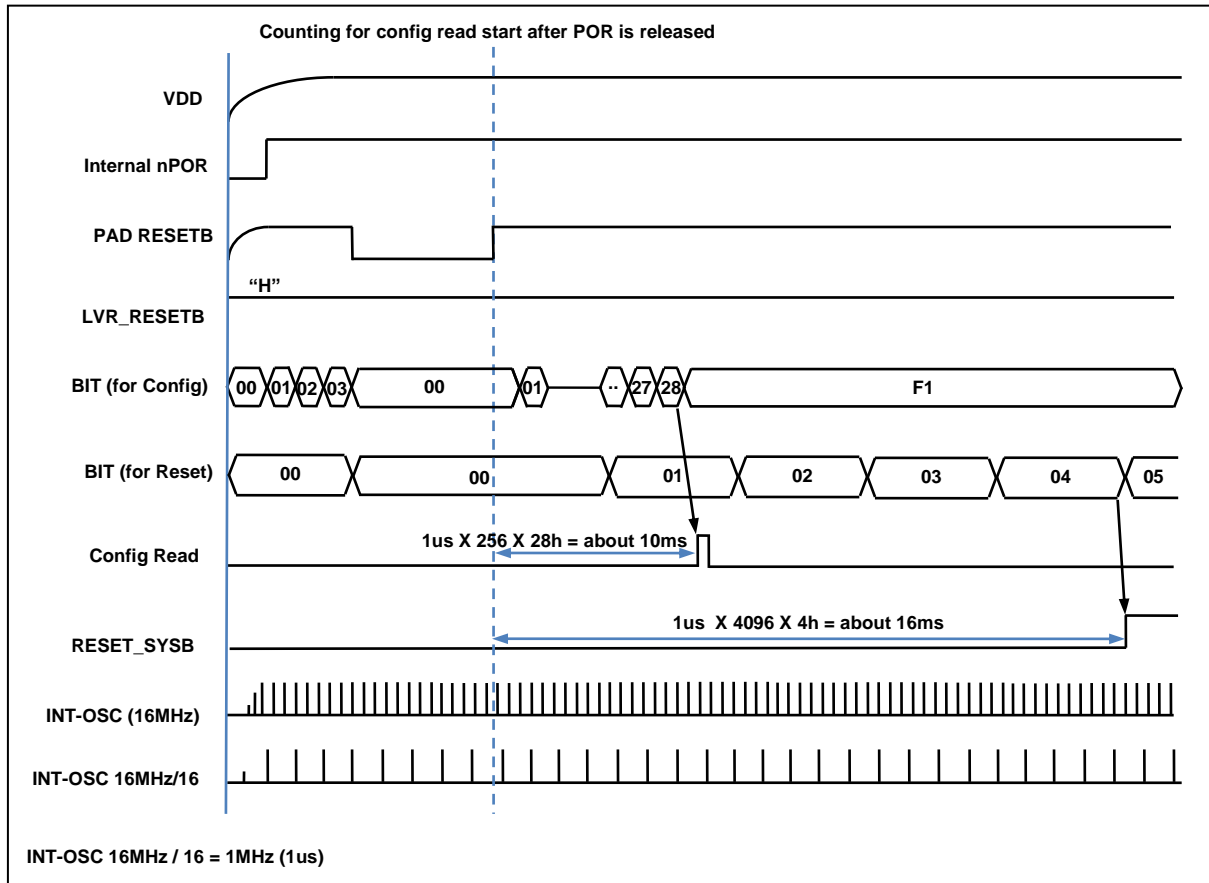


Figure 13.5 Configuration Timing when Power-on

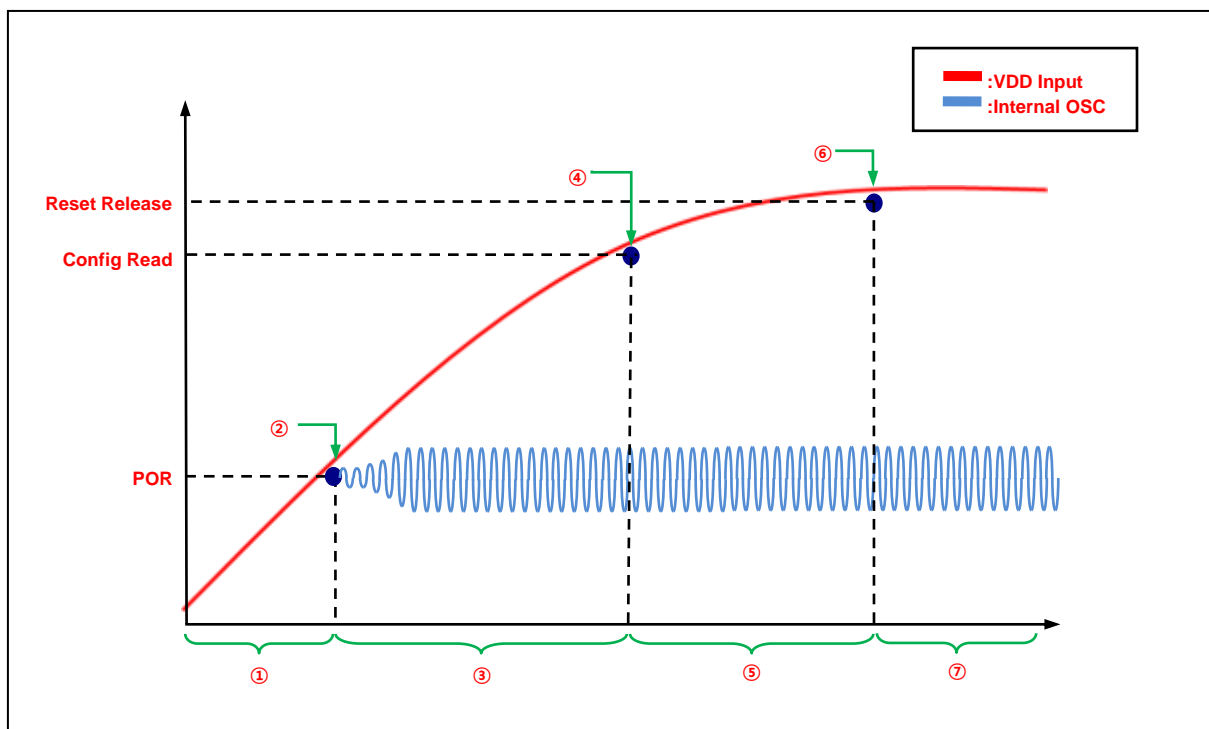


Figure 13.6 Boot Process WaveForm

**Table 13-2 Boot Process Description**

| Process | Description                                                                                                                                  | Remarks                                                             |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|
| ①       | -No Operation                                                                                                                                |                                                                     |
| ②       | -1st POR level Detection                                                                                                                     | -about 1.4V                                                         |
| ③       | - (INT-OSC 16MHz/16)x256x28h Delay section (=10ms)<br>-VDD input voltage must rise over than flash operating voltage for Config read         | -Slew Rate $\geq 0.15\text{V/ms}$                                   |
| ④       | - Config read point                                                                                                                          | -about 1.5V ~ 1.6V<br>-Config Value is determined by Writing Option |
| ⑤       | - Rising section to Reset Release Level                                                                                                      | -16ms point after POR or Ext_reset release                          |
| ⑥       | - Reset Release section (BIT overflow)<br>i) after16ms, after External Reset Release (External reset)<br>ii) 16ms point after POR (POR only) | - BIT is used for Peripheral stability                              |
| ⑦       | -Normal operation                                                                                                                            |                                                                     |

### 13.6 External RESETB Input

The External RESETB is the input to a Schmitt trigger. If RESETB pin is held with low for at least 10us over within the operating voltage range and stable oscillation, it is applied and the internal state is initialized. After reset state becomes '1', it needs the stabilization time with 16ms and after the stable state, the internal RESET becomes '1'. The Reset process step needs 5 oscillator clocks. And the program execution starts at the vector address stored at address 0000H.

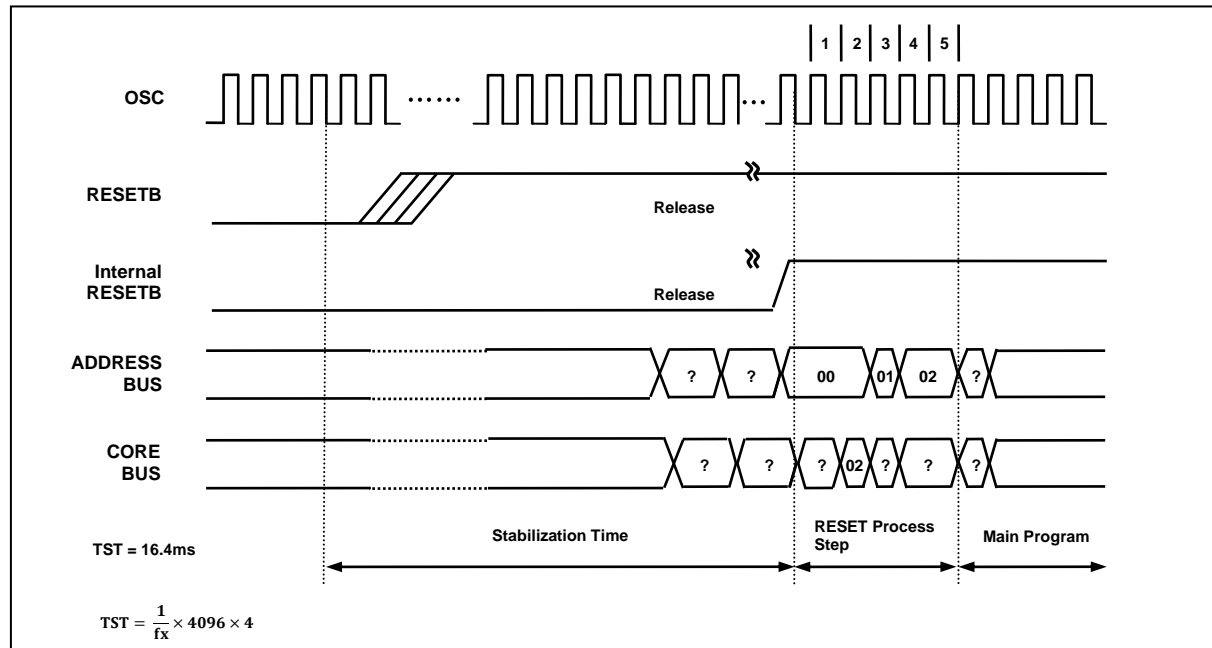


Figure 13.7 Timing Diagram after RESET

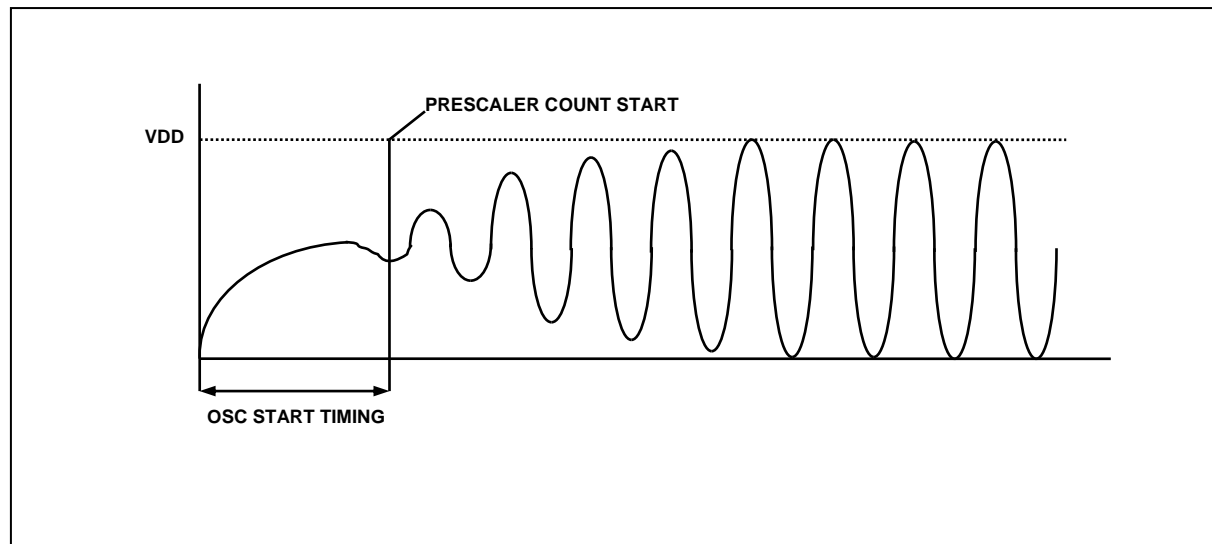


Figure 13.8 Oscillator generating waveform example

NOTE) As shown Figure 13.8, the stable generating time is not included in the start-up time.

The RESETB pin has a internal pull-up resistor by H/W.

### 13.7 Brown Out Detector Processor

The MC96F8316 has an On-chip brown-out detection circuit (BOD) for monitoring the VDD level during operation by comparing it to a fixed trigger level. The trigger level for the BOD can be selected by LVRVS[3:0] bit to be 1.60V, 2.00V, 2.10V, 2.20V, 2.32V, 2.44V, 2.59V, 2.75V, 2.93V, 3.14V, 3.38V, 3.67V, 4.00V, 4.40V. In the STOP mode, this will contribute significantly to the total current consumption. So to minimize the current consumption, the LVREN bit is set to off by software.

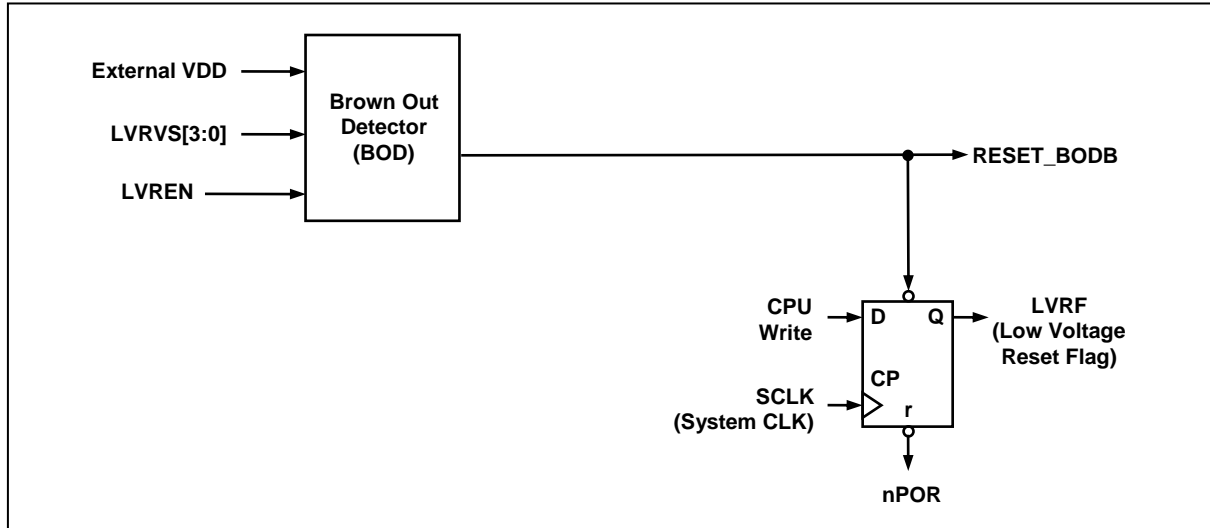


Figure 13.9 Block Diagram of BOD

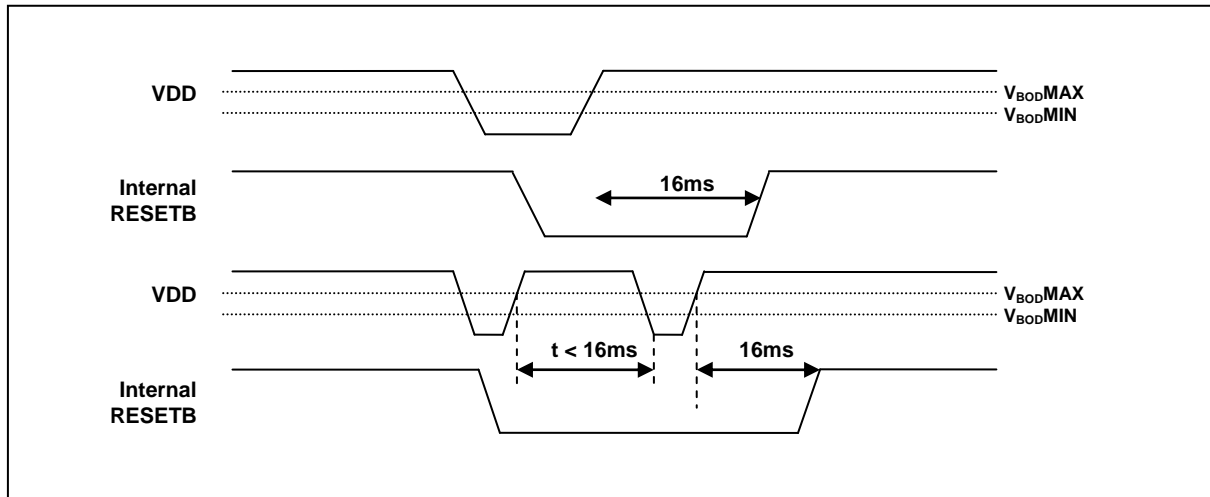


Figure 13.10 Internal Reset at the power fail situation

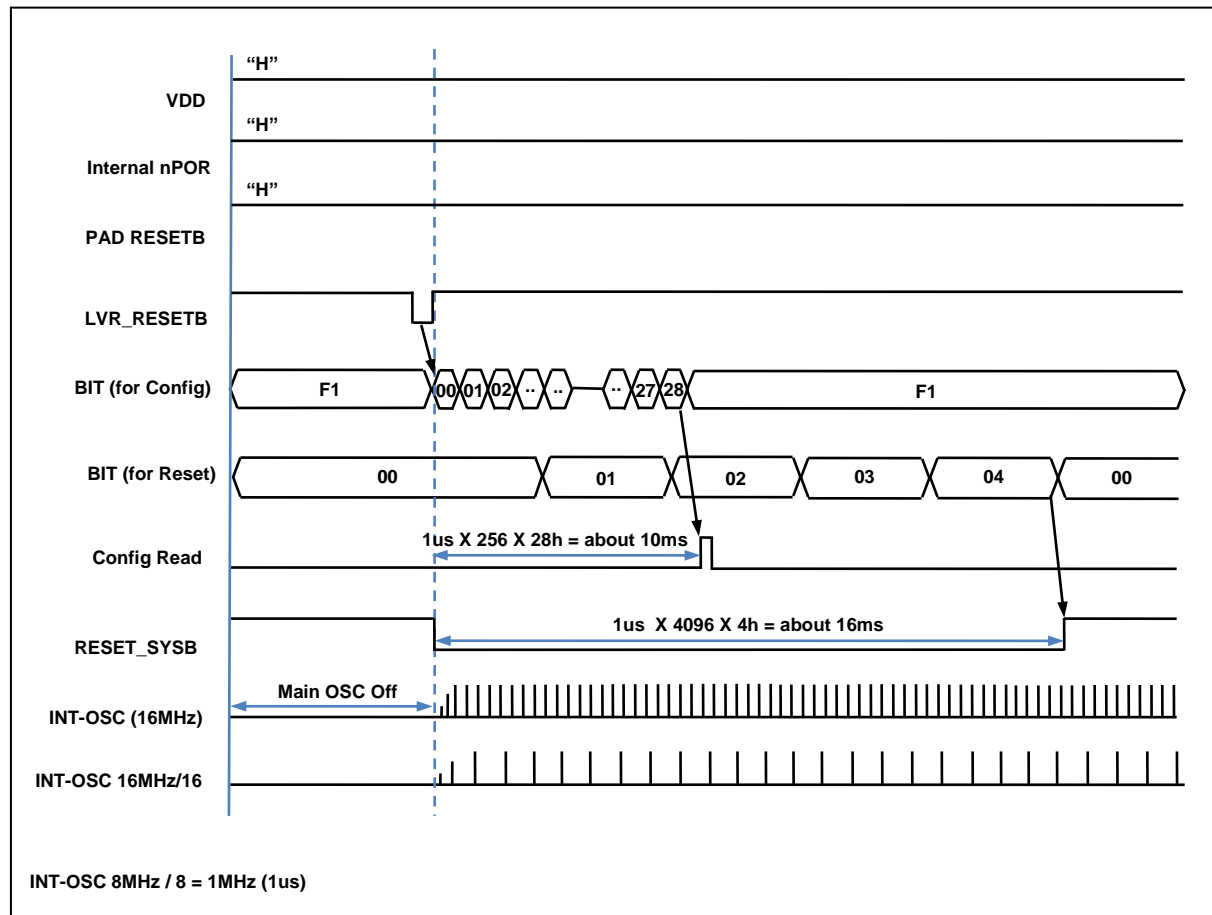


Figure 13.11 Configuration timing when BOD RESET

### 13.8 LVI Block Diagram

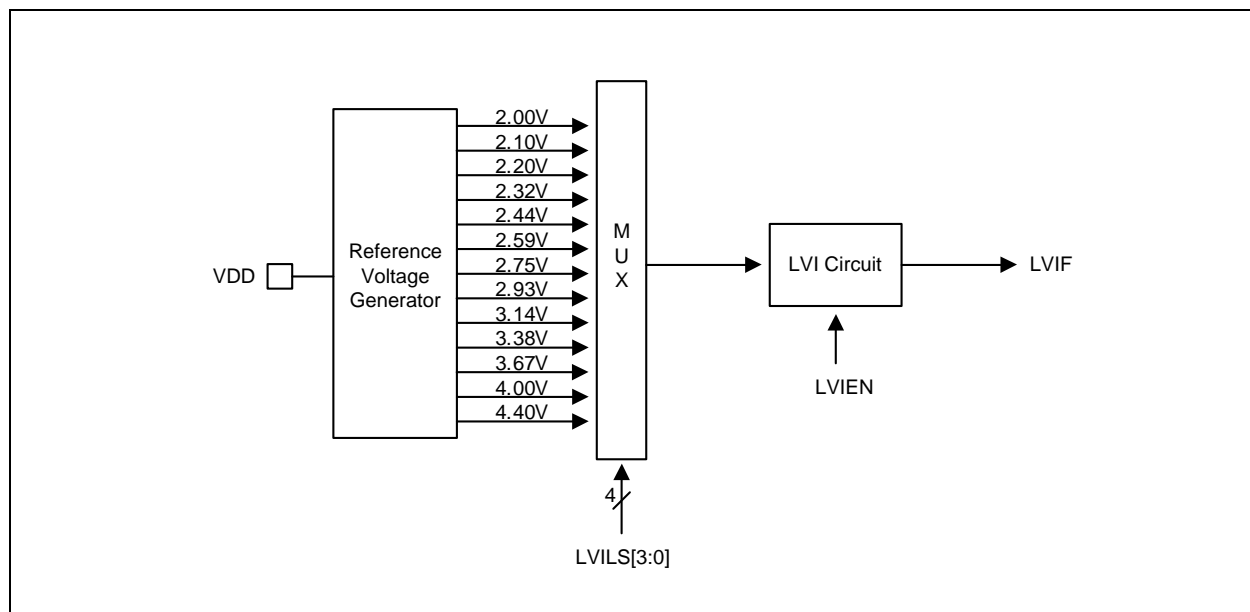


Figure 13.12 LVI Diagram

### 13.8.1 Register Map

Table 13-3 Reset Operation Register Map

| Name   | Address | Dir | Default | Description                            |
|--------|---------|-----|---------|----------------------------------------|
| RSTFR  | E8H     | R/W | 80H     | Reset Flag Register                    |
| LVRCCR | D8H     | R/W | 00H     | Low Voltage Reset Control Register     |
| LVICR  | 86H     | R/W | 00H     | Low Voltage Indicator Control Register |

### 13.8.2 Reset Operation Register Description

The reset control register consists of the reset flag register (RSTFR), low voltage reset control register (LVRCCR), and low voltage indicator control register (LVICR).

### 13.8.3 Register Description for Reset Operation

#### RSTFR (Reset Flag Register) : E8H

| 7    | 6     | 5     | 4     | 3    | 2 | 1 | 0 |
|------|-------|-------|-------|------|---|---|---|
| PORF | EXTRF | WDTRF | OCDRF | LVRF | — | — | — |
| R/W  | R/W   | R/W   | R/W   | R/W  | — | — | — |

Initial value : 80H

|       |                                                                                                     |
|-------|-----------------------------------------------------------------------------------------------------|
| PORF  | Power-On Reset flag bit. The bit is reset by writing '0' to this bit.                               |
| 0     | No detection                                                                                        |
| 1     | Detection                                                                                           |
| EXTRF | External Reset (RESETB) flag bit. The bit is reset by writing '0' to this bit or by Power-On Reset. |
| 0     | No detection                                                                                        |
| 1     | Detection                                                                                           |
| WDTRF | Watch Dog Reset flag bit. The bit is reset by writing '0' to this bit or by Power-On Reset.         |
| 0     | No detection                                                                                        |
| 1     | Detection                                                                                           |
| OCDRF | On-Chip Debug Reset flag bit. The bit is reset by writing '0' to this bit or by Power-On Reset.     |
| 0     | No detection                                                                                        |
| 1     | Detection                                                                                           |
| LVRF  | Low Voltage Reset flag bit. The bit is reset by writing '0' to this bit or by Power-On Reset.       |
| 0     | No detection                                                                                        |
| 1     | Detection                                                                                           |

NOTES) 1. When the Power-On Reset occurs, the PORF bit is only set to "1", the other flag (WDTRF and OCDRF) bits are all cleared to "0".

2. When the Power-On Reset occurs, the EXTRF bit is unknown, At that time, the EXTRF bit can be set to "1" when External Reset (RESETB) occurs.

3. When the Power-On Reset occurs, the LVRF bit is unknown, At that time, the LVRF bit can be set to "1" when LVR Reset occurs.

4. When a reset except the POR occurs, the corresponding flag bit is only set to "1", the other flag bits are kept in the previous values.

**LVR CR (Low Voltage Reset Control Register) : D8H**

| 7     | 6 | 5 | 4      | 3      | 2      | 1      | 0     |
|-------|---|---|--------|--------|--------|--------|-------|
| LVRST | – | – | LVRVS3 | LVRVS2 | LVRVS1 | LVRVS0 | LVREN |
| R/W   | – | – | R/W    | R/W    | R/W    | R/W    | R/W   |

Initial value : 00H

**LVRST** LVR Enable when Stop Release

0 Not effect at stop release

1 LVR enable at stop release

**NOTES)**

When this bit is '1', the LVREN bit is cleared to '0' by stop mode release. (LVR enable)

When this bit is '0', the LVREN bit is not effect by stop mode release.

**LVRVS[3:0]** LVR Voltage Select

| LVRVS3       | LVRVS2 | LVRVS1 | LVRVS0 | Description   |
|--------------|--------|--------|--------|---------------|
| 0            | 0      | 0      | 0      | 1.60V         |
| 0            | 0      | 0      | 1      | 2.00V         |
| 0            | 0      | 1      | 0      | 2.10V         |
| 0            | 0      | 1      | 1      | 2.20V         |
| 0            | 1      | 0      | 0      | 2.32V         |
| 0            | 1      | 0      | 1      | 2.44V         |
| 0            | 1      | 1      | 0      | 2.59V         |
| 0            | 1      | 1      | 1      | 2.75V         |
| 1            | 0      | 0      | 0      | 2.93V         |
| 1            | 0      | 0      | 1      | 3.14V         |
| 1            | 0      | 1      | 0      | 3.38V         |
| 1            | 0      | 1      | 1      | 3.67V         |
| 1            | 1      | 0      | 0      | 4.00V         |
| 1            | 1      | 0      | 1      | 4.40V         |
| Other Values |        |        |        | Not available |

**LVREN** LVR Operation

0 LVR Enable

1 LVR Disable

NOTES) 1. The LVRVS[3:0] bits are cleared by a power-on reset but are retained by other reset signals.

2. The LVRVS[3:0] bits should be set to '0000b' while LVREN bit is "1".



**LVICR (Low Voltage Indicator Control Register) : 86H**

| 7 | 6 | 5    | 4     | 3     | 2     | 1     | 0     |
|---|---|------|-------|-------|-------|-------|-------|
| – | – | LVIF | LVLEN | LVLS3 | LVLS2 | LVLS1 | LVLS0 |
| – | – | RW   | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

LVIF Low Voltage Indicator Flag Bit

0 No detection

1 Detection

LVLEN LVI Enable/Disable

0 Disable

1 Enable

LVLS[3:0] LVI Level Select

| LVLS3 | LVLS2 | LVLS1 | LVLS0 | Description |
|-------|-------|-------|-------|-------------|
|-------|-------|-------|-------|-------------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 0 | 0 | 0 | 0 | 2.00V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 0 | 0 | 0 | 1 | 2.10V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 0 | 0 | 1 | 0 | 2.20V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 0 | 0 | 1 | 1 | 2.32V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 0 | 1 | 0 | 0 | 2.44V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 0 | 1 | 0 | 1 | 2.59V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 0 | 1 | 1 | 0 | 2.75V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 0 | 1 | 1 | 1 | 2.93V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 1 | 0 | 0 | 0 | 3.14V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 1 | 0 | 0 | 1 | 3.38V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 1 | 0 | 1 | 0 | 3.67V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 1 | 0 | 1 | 1 | 4.00V |
|---|---|---|---|-------|

|   |   |   |   |       |
|---|---|---|---|-------|
| 1 | 1 | 0 | 0 | 4.40V |
|---|---|---|---|-------|

Other Values Not available

## 14. On-chip Debug System

### 14.1 Overview

#### 14.1.1 Description

On-chip debug system (OCD) of MC96F8316 can be used for programming the non-volatile memories and on-chip debugging. Detail descriptions for programming via the OCD interface can be found in the following chapter.

Figure 14.1 shows a block diagram of the OCD interface and the On-chip Debug system.

#### 14.1.2 Feature

- Two-wire external interface: 1-wire serial clock input, 1-wire bi-directional serial data bus
- Debugger Access to:
  - All Internal Peripheral Units
  - Internal data RAM
  - Program Counter
  - Flash and Data EEPROM Memories
- Extensive On-chip Debug Support for Break Conditions, Including
  - Break Instruction
  - Single Step Break
  - Program Memory Break Points on Single Address
  - Programming of Flash, EEPROM, Fuses, and Lock Bits through the two-wire Interface
  - On-chip Debugging Supported by Dr.Choice<sup>®</sup>
- Operating frequency
  - Supports the maximum frequency of the target MCU

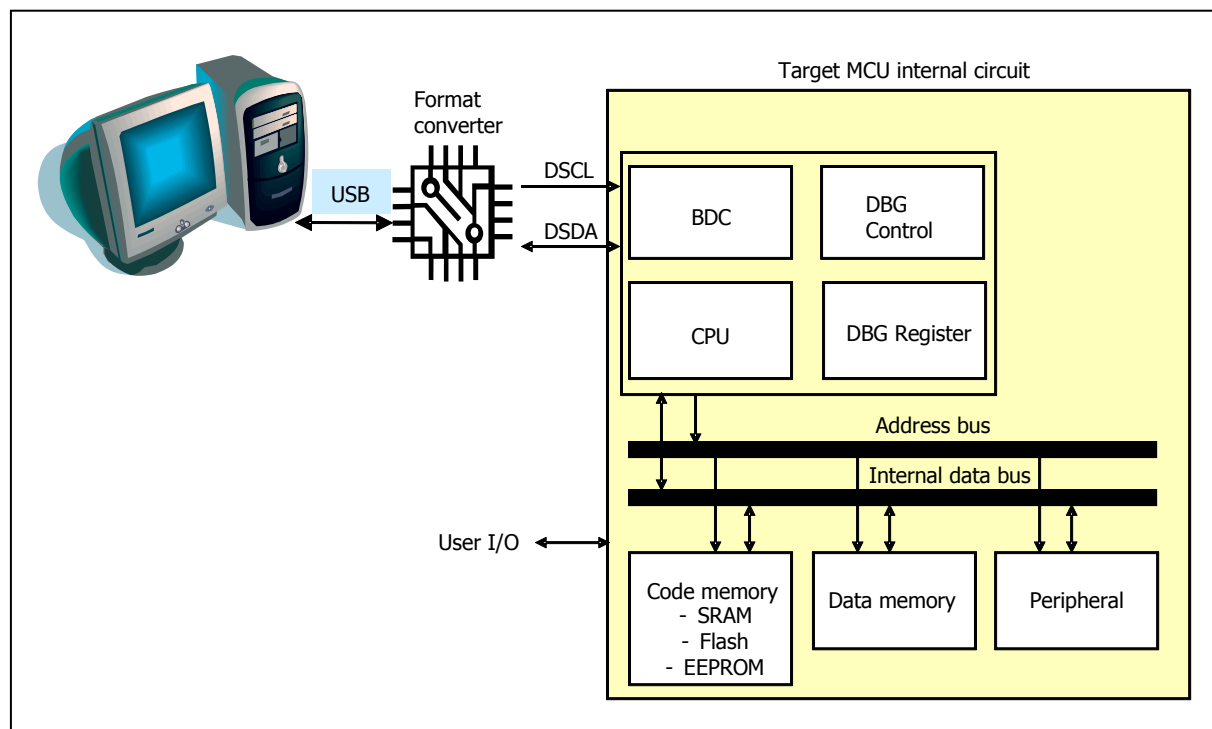


Figure 14.1 Block Diagram of On-Chip Debug System

## 14.2 Two-Pin External Interface

### 14.2.1 Basic Transmission Packet

- 10-bit packet transmission using two-pin interface.
- 1-packet consists of 8-bit data, 1-bit parity and 1-bit acknowledge.
- Parity is even of '1' for 8-bit data in transmitter.
- Receiver generates acknowledge bit as '0' when transmission for 8-bit data and its parity has no error.
- When transmitter has no acknowledge (Acknowledge bit is '1' at tenth clock), error process is executed in transmitter.
- When acknowledge error is generated, host PC makes stop condition and transmits command which has error again.
- Background debugger command is composed of a bundle of packet.
- Start condition and stop condition notify the start and the stop of background debugger command respectively.

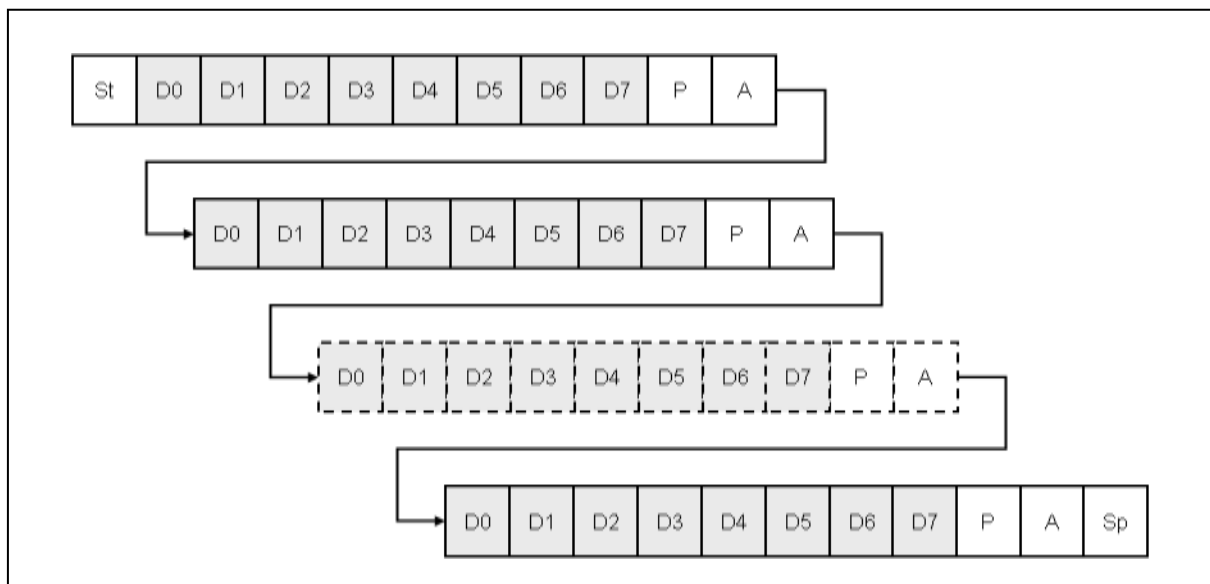


Figure 14.2 10-bit Transmission Packet

## 14.2.2 Packet Transmission Timing

### 14.2.2.1 Data Transfer

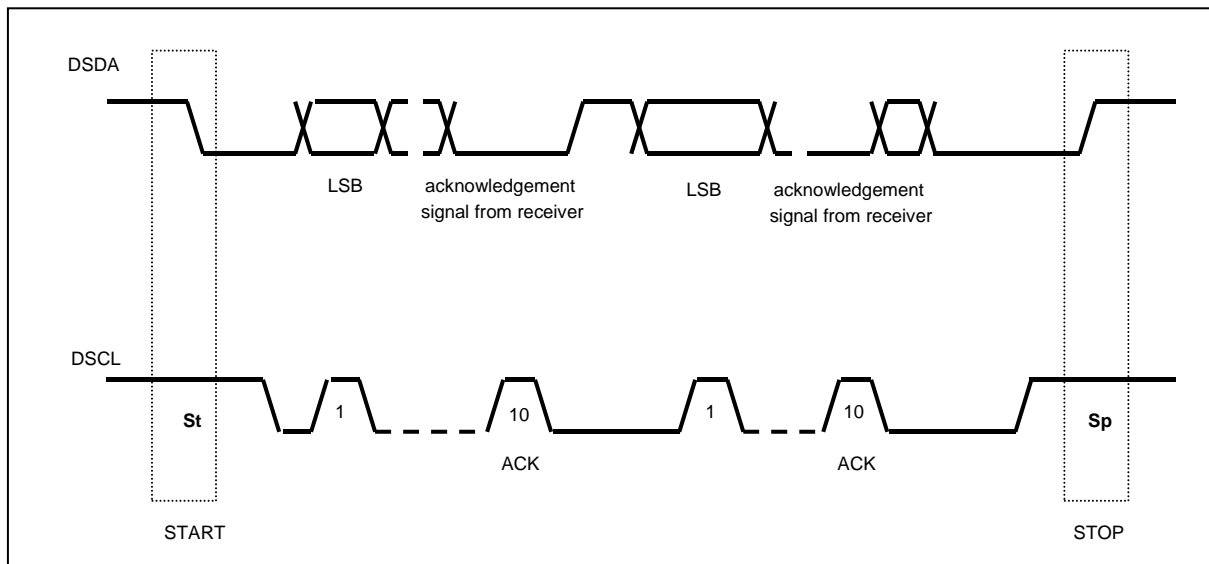


Figure 14.3 Data Transfer on the Twin Bus

### 14.2.2.2 Bit Transfer

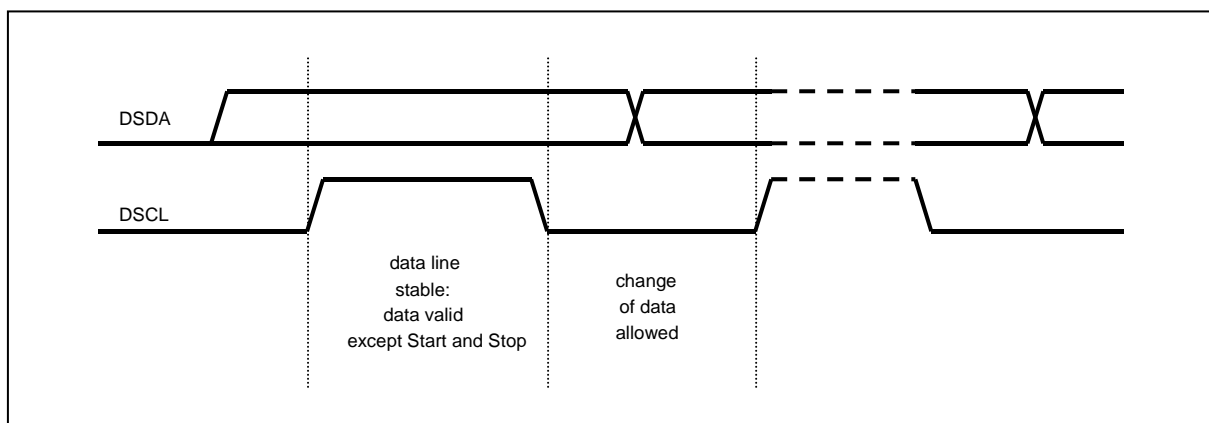


Figure 14.4 Bit Transfer on the Serial Bus

### 14.2.2.3 Start and Stop Condition

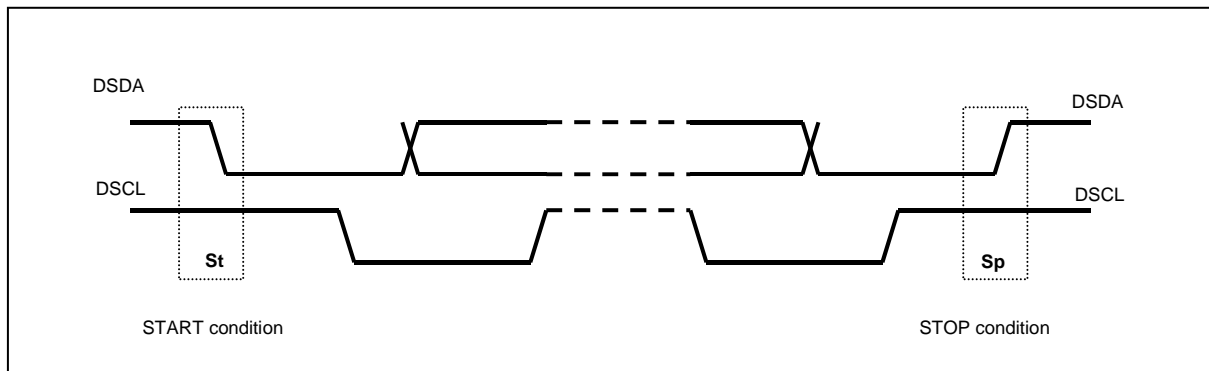


Figure 14.5 Start and Stop Condition

### 14.2.2.4 Acknowledge Bit

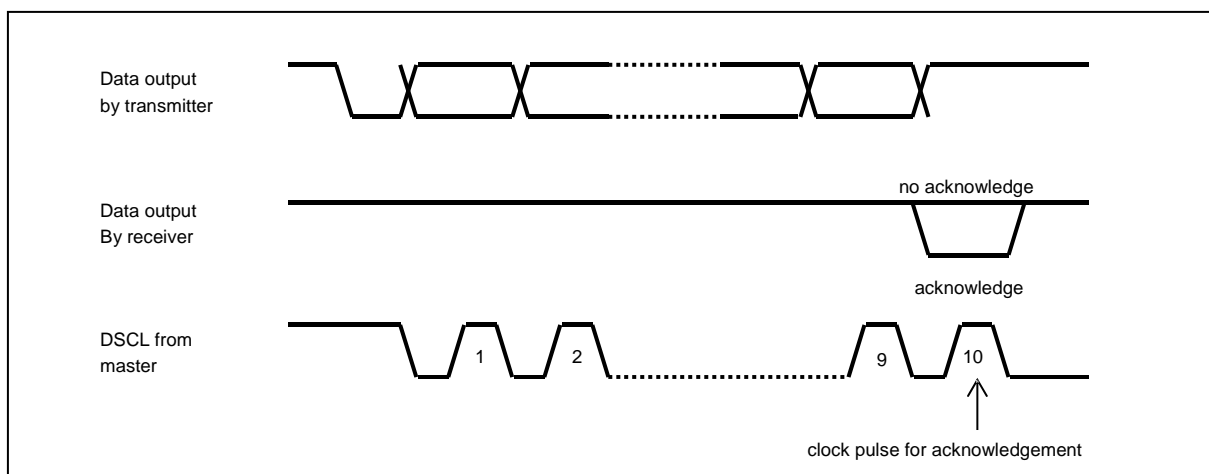
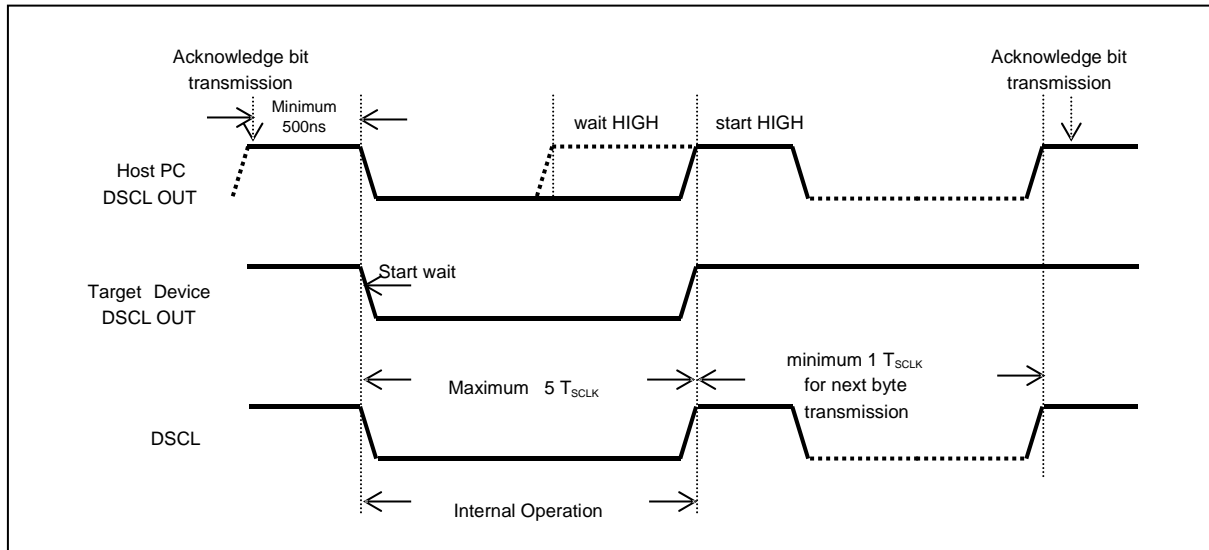
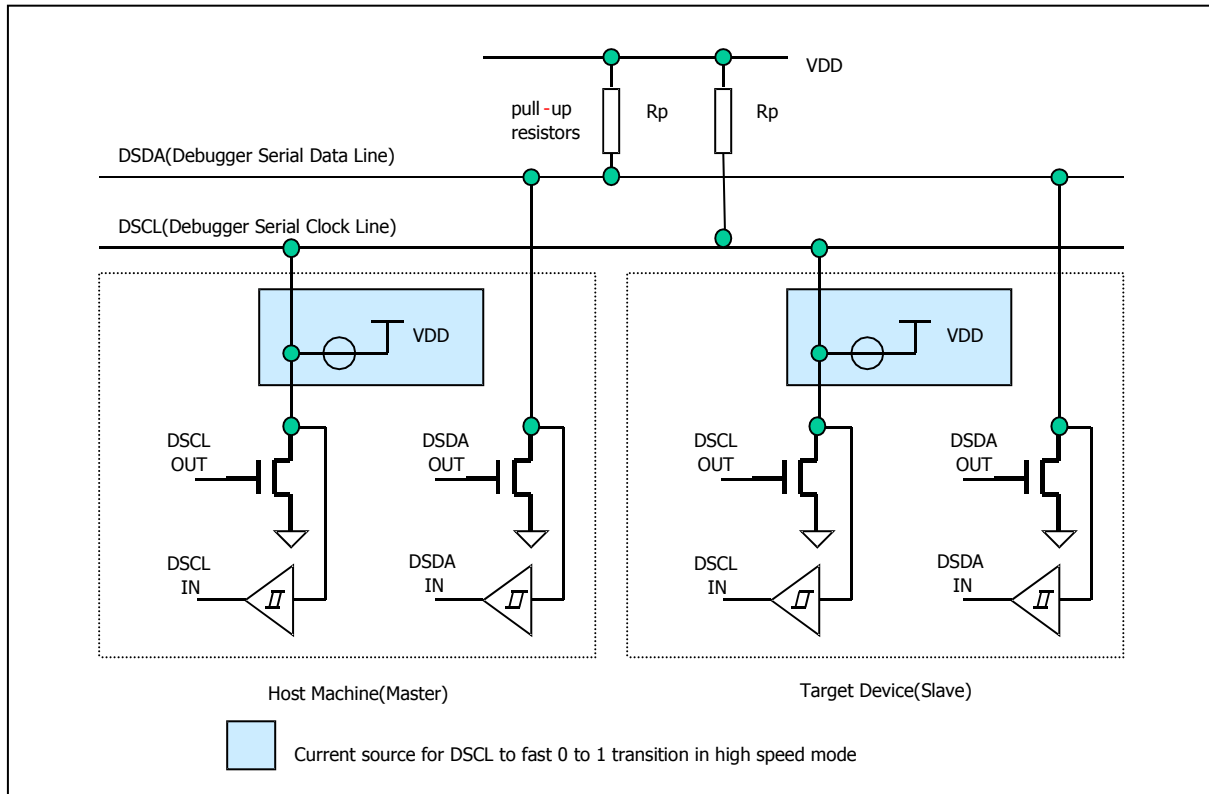


Figure 14.6 Acknowledge on the Serial Bus

**Figure 14.7 Clock Synchronization during Wait Procedure**

### 14.2.3 Connection of Transmission

Two-pin interface connection uses open-drain (wire-AND bidirectional I/O).



**Figure 14.8 Connection of Transmission**

## 15. Flash Memory

### 15.1 Overview

#### 15.1.1 Description

MC96F8316 incorporates flash memory to which a program can be written, erased, and overwritten while mounted on the board. The flash memory can be read by 'MOVC' instruction and it can be programmed in OCD, serial ISP mode or user program mode.

- Flash Size : 16k bytes
- Single power supply program and erase
- Command interface for fast program and erase operation
- Up to 100,000 program/erase cycles at typical voltage and temperature for flash memory



### 15.1.2 Flash Program ROM Structure

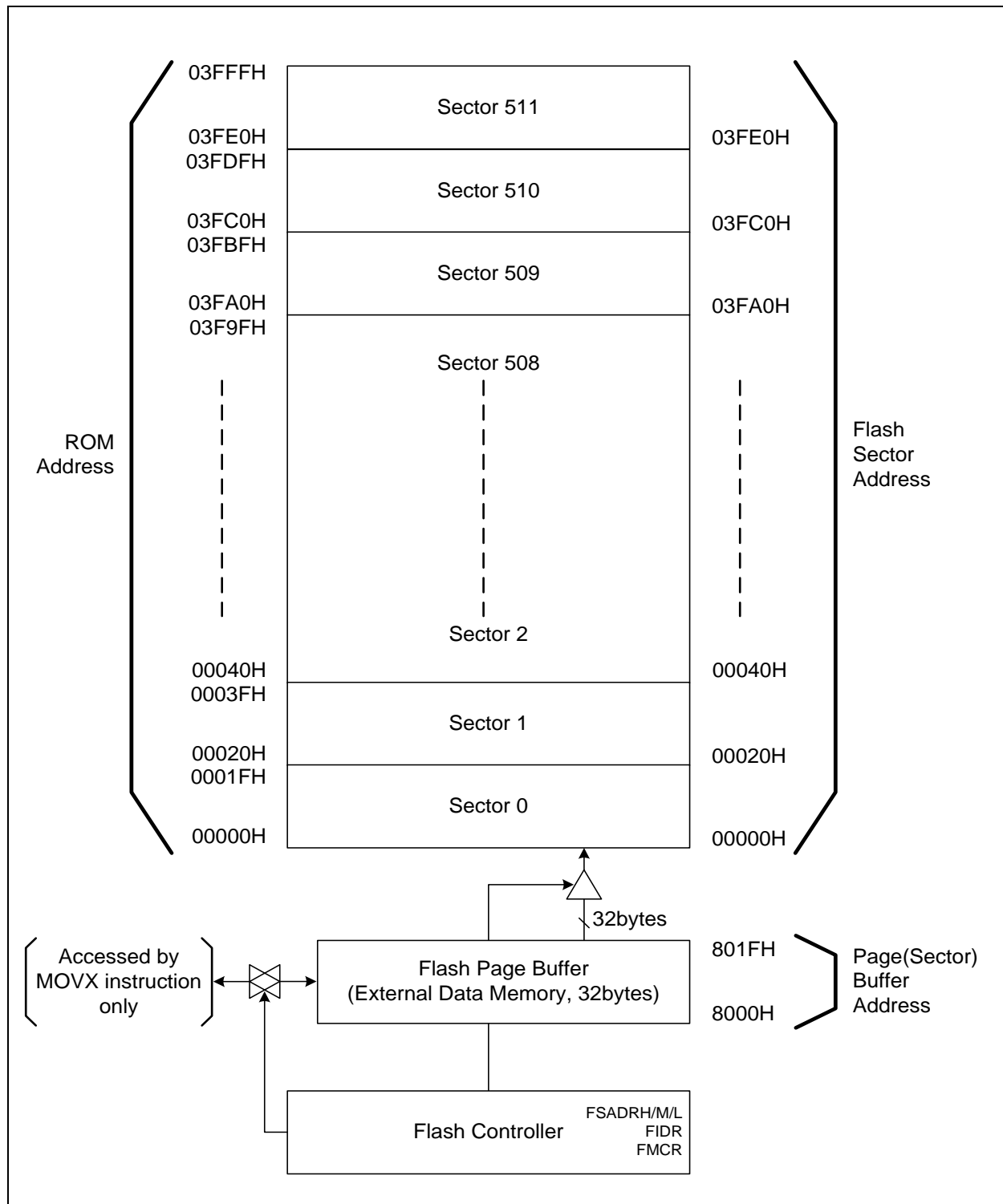


Figure 15.1 Flash Program ROM Structure

### 15.1.3 Register Map

**Table 15-1 Flash Memory Register Map**

| Name   | Address | Dir | Default | Description                          |
|--------|---------|-----|---------|--------------------------------------|
| FSADRH | FAH     | R/W | 00H     | Flash Sector Address High Register   |
| FSADRM | FBH     | R/W | 00H     | Flash Sector Address Middle Register |
| FSADRL | FCH     | R/W | 00H     | Flash Sector Address Low Register    |
| FIDR   | FDH     | R/W | 00H     | Flash Identification Register        |
| FMCR   | FEH     | R/W | 00H     | Flash Mode Control Register          |

### 15.1.4 Register Description for Flash Memory Control and Status

Flash control register consists of the flash sector address high register (FSADRH), flash sector address middle register (FSADRM), flash sector address low register (FSADRL), flash identification register (FIDR), and flash mode control register (FMCR). They are mapped to SFR area and can be accessed only in programming mode.

### 15.1.5 Register Description for Flash

#### FSADRH (Flash Sector Address High Register) : FAH

| 7 | 6 | 5 | 4 | 3       | 2       | 1       | 0       |
|---|---|---|---|---------|---------|---------|---------|
| – | – | – | – | FSADRH3 | FSADRH2 | FSADRH1 | FSADRH0 |
| – | – | – | – | RW      | RW      | RW      | RW      |

Initial value : 00H

FSADRH[3:0] Flash Sector Address High

#### FSADRM (Flash Sector Address Middle Register) : FBH

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| FSADRM7 | FSADRM6 | FSADRM5 | FSADRM4 | FSADRM3 | FSADRM2 | FSADRM1 | FSADRM0 |
| RW      | RW      | RW      | RW      | RW      | RW      | RW      | RW      |

Initial value : 00H

FSADRM[7:0] Flash Sector Address Middle

#### FSADRL (Flash Sector Address Low Register) : FCH

| 7       | 6       | 5       | 4       | 3       | 2       | 1       | 0       |
|---------|---------|---------|---------|---------|---------|---------|---------|
| FSADRL7 | FSADRL6 | FSADRL5 | FSADRL4 | FSADRL3 | FSADRL2 | FSADRL1 | FSADRL0 |
| RW      | RW      | RW      | RW      | RW      | RW      | RW      | RW      |

Initial value : 00H

FSADRL[7:0] Flash Sector Address Low

#### FIDR (Flash Identification Register) : FDH

| 7     | 6     | 5     | 4     | 3     | 2     | 1     | 0     |
|-------|-------|-------|-------|-------|-------|-------|-------|
| FIDR7 | FIDR6 | FIDR5 | FIDR4 | FIDR3 | FIDR2 | FIDR1 | FIDR0 |
| RW    | RW    | RW    | RW    | RW    | RW    | RW    | RW    |

Initial value : 00H

FIDR[7:0] Flash Identification

Others No identification value

10100101 Identification value for a flash mode

(These bits are automatically cleared to logic '00H' immediately after one time operation except "flash page buffer reset mode")

**FMCR (Flash Mode Control Register) : FEH**

| 7      | 6 | 5 | 4 | 3 | 2     | 1     | 0     |
|--------|---|---|---|---|-------|-------|-------|
| FMBUSY | – | – | – | – | FMCR2 | FMCR1 | FMCR0 |
| R      | – | – | – | – | RW    | RW    | RW    |

Initial value : 00H

FMBUSY Flash Mode Busy Bit. This bit will be used for only debugger.

0 No effect when “1” is written

1 Busy

FMCR[2:0] Flash Mode Control Bits. During a flash mode operation, the CPU is hold and the global interrupt is on disable state regardless of the IE.7 (EA) bit.

| FMCR2 | FMCR1 | FMCR0 | Description                                                                                           |
|-------|-------|-------|-------------------------------------------------------------------------------------------------------|
| 0     | 0     | 1     | Select flash page buffer reset mode and start regardless of the FIDR value (Clear all 32bytes to '0') |
| 0     | 1     | 0     | Select flash sector erase mode and start operation when the FIDR="10100101b'                          |
| 0     | 1     | 1     | Select flash sector write mode and start operation when the FIDR="10100101b'                          |
| 1     | 0     | 0     | Select flash sector hard lock and start operation when the FIDR="10100101b'                           |

Others Values: No operation

(These bits are automatically cleared to logic '00H' immediately after one time operation)

### 15.1.6 Serial In-System Program (ISP) Mode

Serial in-system program uses the interface of debugger which uses two wires. Refer to chapter 14 in details about debugger

### 15.1.7 Protection Area (User program mode)

MC96F8316 can program its own flash memory (protection area). The protection area can not be erased or programmed. The protection areas are available only when the PAEN bit is cleared to '0', that is, enable protection area at the configure option 2 if it is needed. If the protection area isn't enabled (PAEN = '1'), this area can be used as a normal program memory.

The size of protection area can be varied by setting of configure option 2.

**Table 15-2 Protection Area size**

| Protection Area Size Select |       | Size of Protection Area | Address of Protection Area |
|-----------------------------|-------|-------------------------|----------------------------|
| PASS1                       | PASS0 |                         |                            |
| 0                           | 0     | 3.8k Bytes              | 0100H – 0FFFH              |
| 0                           | 1     | 1.7k Bytes              | 0100H – 07FFH              |
| 1                           | 0     | 768 Bytes               | 0100H – 03FFH              |
| 1                           | 1     | 256 Bytes               | 0100H – 01FFH              |

NOTE) Refer to chapter 16 in configure option control.

### 15.1.8 Erase Mode

#### The sector erase program procedure in user program mode

1. Page buffer clear (FMCR=0x01)
2. Write '0' to page buffer
3. Set flash sector address register (FSADRH/FSADRM/FSADRL).
4. Set flash identification register (FIDR).
5. Set flash mode control register (FMCR).
6. Erase verify

#### Program Tip – sector erase

```

MOV    FMCR,#0x01           ;page buffer clear
NOP                    ;Dummy instruction, This instruction must be needed.
NOP                    ;Dummy instruction, This instruction must be needed.
NOP                    ;Dummy instruction, This instruction must be needed.

MOV    A,#0
MOV    R0,#32               ;Sector size is 32bytes
MOV    DPH,#0x80
MOV    DPL,#0

Pgbuf_clr: MOVX   @DPTR,A
INC     DPTR
DJNZ    R0, Pgbuf_clr       ;Write '0' to all page buffer

MOV    FSADRH,#0x00
MOV    FSADRM,#0x3F
MOV    FSADRL,#0xA0        ;Select sector 509
MOV    FIDR,#0xA5          ;Identification value
MOV    FMCR,#0x02          ;Start flash erase mode
NOP                    ;Dummy instruction, This instruction must be needed.
NOP                    ;Dummy instruction, This instruction must be needed.
NOP                    ;Dummy instruction, This instruction must be needed.

MOV    A,#0                 ;erase verify
MOV    R0,#32               ;Sector size is 32bytes
MOV    R1,#0
MOV    DPH,#0x3F
MOV    DPL,#0xA0

Erase_verify:
MOV    A,@A+DPTR
SUBB   A,R1
JNZ    Verify_error
INC     DPTR
DJNZ    R0, Erase_verify

Verify_error:

```

## The Byte erase program procedure in user program mode

1. Page buffer clear (FMCR=0x01)
2. Write '0' to page buffer
3. Set flash sector address register (FSADRH/FSADRM/FSADRL).
4. Set flash identification register (FIDR).
5. Set flash mode control register (FMCR).
6. Erase verify

### Program Tip – byte erase

```

MOV    FMCR,#0x01          ;page buffer clear
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.

MOV     A,#0
MOV     DPH,#0x80
MOV     DPL,#0
MOVX    @DPTR,A

MOV     DPH,#0x80
MOV     DPL,#0x05
MOVX    @DPTR,A            ;Write '0' to page buffer

MOV     FSADRH,#0x00
MOV     FSADRM,#0x3F
MOV     FSADRL,#0xA0       ;Select sector 509
MOV     FIDR,#0xA5         ;Identification value
MOV     FMCR,#0x02         ;Start flash erase mode
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.

MOV     A,#0                ;erase verify
MOV     R1,#0
MOV     DPH,#0x3F
MOV     DPL,#0xA0
MOVC    A,@A+DPTR
SUBB    A,R1                ;0x3FA0 = 0 ?
JNZ     Verify_error

MOV     A,#0
MOV     R1,#0
MOV     DPH,#0x3F
MOV     DPL,#0xA5
MOVC    A,@A+DPTR
SUBB    A,R1                ;0x3FA5 = 0 ?
JNZ     Verify_error

```

Verify\_error:

### 15.1.9 Write Mode

#### The sector Write program procedure in user program mode

1. Page buffer clear (FMCR=0x01)
2. Write data to page buffer
3. Set flash sector address register (FSADRH/FSADRM/FSADRL).
4. Set flash identification register (FIDR).
5. Set flash mode control register (FMCR).
6. Erase verify

#### Program Tip – sector write

```

MOV    FMCR,#0x01           ;page buffer clear
NOP                               ;Dummy instruction, This instruction must be needed.
NOP                               ;Dummy instruction, This instruction must be needed.
NOP                               ;Dummy instruction, This instruction must be needed.

MOV    A,#0
MOV    R0,#32                ;Sector size is 32bytes
MOV    DPH,#0x80
MOV    DPL,#0

Pgbuf_WR: MOVX   @DPTR,A
INC     A
INC     DPTR
DJNZ    R0, Pgbuf_WR         ;Write data to all page buffer

MOV     FSADRH,#0x00
MOV     FSADRM,#0x3F
MOV     FSADRL,#0xA0         ;Select sector 509
MOV     FIDR,#0xA5           ;Identification value
MOV     FMCR,#0x03           ;Start flash write mode
NOP                               ;Dummy instruction, This instruction must be needed.
NOP                               ;Dummy instruction, This instruction must be needed.
NOP                               ;Dummy instruction, This instruction must be needed.

MOV     A,#0                 ;write verify
MOV     R0,#32               ;Sector size is 32bytes
MOV     R1,#0
MOV     DPH,#0x3F
MOV     DPL,#0xA0

Write_verify:
MOVC    A,@A+DPTR
SUBB    A,R1
JNZ     Verify_error
INC     R1
INC     DPTR
DJNZ    R0, Write_verify

Verify_error:

```



## The Byte Write program procedure in user program mode

1. Page buffer clear (FMCR=0x01)
2. Write data to page buffer
3. Set flash sector address register (FSADRH/FSADRM/FSADRL).
4. Set flash identification register (FIDR).
5. Set flash mode control register (FMCR).
6. Erase verify

### Program Tip – byte write

```

MOV    FMCR,#0x01          ;page buffer clear
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.

MOV    A,#5
MOV    DPH,#0x80
MOV    DPL,#0
MOVX   @DPTR,A              ;Write data to page buffer

MOV    A,#6
MOV    DPH,#0x80
MOV    DPL,#0x05
MOVX   @DPTR,A              ;Write data to page buffer

MOV    FSADRH,#0x00
MOV    FSADRM,#0x3F
MOV    FSADRL,#0xA0        ;Select sector 509
MOV    FIDR,#0xA5          ;Identification value
MOV    FMCR,#0x03          ;Start flash write mode
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.

MOV    A,#0                  ;write verify
MOV    R1,#5
MOV    DPH,#0x3F
MOV    DPL,#0xA0
MOVC   A,@A+DPTR
SUBB   A,R1                  ;0x3FA0 = 5 ?
JNZ    Verify_error

MOV    A,#0
MOV    R1,#6
MOV    DPH,#0x3F
MOV    DPL,#0xA5
MOVC   A,@A+DPTR
SUBB   A,R1                  ;0x3FA5 = 6 ?
JNZ    Verify_error

```

Verify\_error:

### 15.1.10 Read Mode

#### The Reading program procedure in user program mode

1. Load receive data from flash memory on MOVC instruction by indirectly addressing mode.

##### Program Tip – reading

```

MOV    A,#0
MOV    DPH,#0x3F
MOV    DPL,#0xA0           ;flash memory address

MOVC   A,@A+DPTR           ;read data from flash memory

```

### 15.1.11 Hard Lock Mode

#### The Reading program procedure in user program mode

1. Set flash identification register (FIDR).
2. Set flash mode control register (FMCR).

##### Program Tip – reading

```

MOV    FIDR,#0xA5           ;Identification value
MOV    FMCR,#0x04           ;Start flash hard lock mode
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.
NOP                                ;Dummy instruction, This instruction must be needed.

```

## 16. Configure Option

### 16.1 Configure Option Control

The data for configure option should be written in the configure option area (001EH – 001FH) by programmer (Writer tools).

#### CONFIGURE OPTION 1 : ROM Address 001FH

| 7   | 6  | 5 | 4 | 3 | 2 | 1 | 0    |
|-----|----|---|---|---|---|---|------|
| R_P | HL | – | – | – | – | – | RSTS |

Initial value : 00H

|      |                                     |
|------|-------------------------------------|
| R_P  | Read Protection                     |
| 0    | Disable "Read protection"           |
| 1    | Enable "Read protection"            |
| HL   | Hard-Lock                           |
| 0    | Disable "Hard-lock"                 |
| 1    | Enable "Hard-lock"                  |
| RSTS | RESETB Select                       |
| 0    | P32 port                            |
| 1    | RESETB port with a pull-up resistor |

#### CONFIGURE OPTION 2: ROM Address 001EH

| 7 | 6 | 5 | 4 | 3 | 2    | 1     | 0     |
|---|---|---|---|---|------|-------|-------|
| – | – | – | – | – | PAEN | PASS1 | PASS0 |

Initial value : 00H

|            |                                |                                                 |                                     |
|------------|--------------------------------|-------------------------------------------------|-------------------------------------|
| PAEN       | Protection Area Enable/Disable |                                                 |                                     |
|            | 0                              | Disable Protection (Erasable by instruction)    |                                     |
|            | 1                              | Enable Protection (Not erasable by instruction) |                                     |
| PASS [1:0] | Protection Area Size Select    |                                                 |                                     |
|            | PASS1                          | PASS0                                           | Description                         |
|            | 0                              | 0                                               | 3.8k Bytes (Address 0100H – 0FFFFH) |
|            | 0                              | 1                                               | 1.7k Bytes (Address 0100H – 07FFFH) |
|            | 1                              | 0                                               | 768 Bytes (Address 0100H – 03FFFH)  |
|            | 1                              | 1                                               | 256 Bytes (Address 0100H – 01FFFH)  |

## 17. APPENDIX

### A. Instruction Table

Instructions are either 1, 2 or 3 bytes long as listed in the 'Bytes' column below.

Each instruction takes either 1, 2 or 4 machine cycles to execute as listed in the following table. 1 machine cycle comprises 2 system clock cycles.

| ARITHMETIC   |                                             |       |        |          |
|--------------|---------------------------------------------|-------|--------|----------|
| Mnemonic     | Description                                 | Bytes | Cycles | Hex code |
| ADD A,Rn     | Add register to A                           | 1     | 1      | 28-2F    |
| ADD A,dir    | Add direct byte to A                        | 2     | 1      | 25       |
| ADD A,@Ri    | Add indirect memory to A                    | 1     | 1      | 26-27    |
| ADD A,#data  | Add immediate to A                          | 2     | 1      | 24       |
| ADDC A,Rn    | Add register to A with carry                | 1     | 1      | 38-3F    |
| ADDC A,dir   | Add direct byte to A with carry             | 2     | 1      | 35       |
| ADDC A,@Ri   | Add indirect memory to A with carry         | 1     | 1      | 36-37    |
| ADDC A,#data | Add immediate to A with carry               | 2     | 1      | 34       |
| SUBB A,Rn    | Subtract register from A with borrow        | 1     | 1      | 98-9F    |
| SUBB A,dir   | Subtract direct byte from A with borrow     | 2     | 1      | 95       |
| SUBB A,@Ri   | Subtract indirect memory from A with borrow | 1     | 1      | 96-97    |
| SUBB A,#data | Subtract immediate from A with borrow       | 2     | 1      | 94       |
| INC A        | Increment A                                 | 1     | 1      | 04       |
| INC Rn       | Increment register                          | 1     | 1      | 08-0F    |
| INC dir      | Increment direct byte                       | 2     | 1      | 05       |
| INC @Ri      | Increment indirect memory                   | 1     | 1      | 06-07    |
| DEC A        | Decrement A                                 | 1     | 1      | 14       |
| DEC Rn       | Decrement register                          | 1     | 1      | 18-1F    |
| DEC dir      | Decrement direct byte                       | 2     | 1      | 15       |
| DEC @Ri      | Decrement indirect memory                   | 1     | 1      | 16-17    |
| INC DPTR     | Increment data pointer                      | 1     | 2      | A3       |
| MUL AB       | Multiply A by B                             | 1     | 4      | A4       |
| DIV AB       | Divide A by B                               | 1     | 4      | 84       |
| DA A         | Decimal Adjust A                            | 1     | 1      | D4       |

| LOGICAL       |                                   |       |        |          |
|---------------|-----------------------------------|-------|--------|----------|
| Mnemonic      | Description                       | Bytes | Cycles | Hex code |
| ANL A,Rn      | AND register to A                 | 1     | 1      | 58-5F    |
| ANL A,dir     | AND direct byte to A              | 2     | 1      | 55       |
| ANL A,@Ri     | AND indirect memory to A          | 1     | 1      | 56-57    |
| ANL A,#data   | AND immediate to A                | 2     | 1      | 54       |
| ANL dir,A     | AND A to direct byte              | 2     | 1      | 52       |
| ANL dir,#data | AND immediate to direct byte      | 3     | 2      | 53       |
| ORL A,Rn      | OR register to A                  | 1     | 1      | 48-4F    |
| ORL A,dir     | OR direct byte to A               | 2     | 1      | 45       |
| ORL A,@Ri     | OR indirect memory to A           | 1     | 1      | 46-47    |
| ORL A,#data   | OR immediate to A                 | 2     | 1      | 44       |
| ORL dir,A     | OR A to direct byte               | 2     | 1      | 42       |
| ORL dir,#data | OR immediate to direct byte       | 3     | 2      | 43       |
| XRL A,Rn      | Exclusive-OR register to A        | 1     | 1      | 68-6F    |
| XRL A,dir     | Exclusive-OR direct byte to A     | 2     | 1      | 65       |
| XRL A,@Ri     | Exclusive-OR indirect memory to A | 1     | 1      | 66-67    |

|               |                                       |   |   |    |
|---------------|---------------------------------------|---|---|----|
| XRL A,#data   | Exclusive-OR immediate to A           | 2 | 1 | 64 |
| XRL dir,A     | Exclusive-OR A to direct byte         | 2 | 1 | 62 |
| XRL dir,#data | Exclusive-OR immediate to direct byte | 3 | 2 | 63 |
| CLR A         | Clear A                               | 1 | 1 | E4 |
| CPL A         | Complement A                          | 1 | 1 | F4 |
| SWAP A        | Swap Nibbles of A                     | 1 | 1 | C4 |
| RL A          | Rotate A left                         | 1 | 1 | 23 |
| RLC A         | Rotate A left through carry           | 1 | 1 | 33 |
| RR A          | Rotate A right                        | 1 | 1 | 03 |
| RRC A         | Rotate A right through carry          | 1 | 1 | 13 |

| DATA TRANSFER  |                                       |       |        |          |
|----------------|---------------------------------------|-------|--------|----------|
| Mnemonic       | Description                           | Bytes | Cycles | Hex code |
| MOV A,Rn       | Move register to A                    | 1     | 1      | E8-EF    |
| MOV A,dir      | Move direct byte to A                 | 2     | 1      | E5       |
| MOV A,@Ri      | Move indirect memory to A             | 1     | 1      | E6-E7    |
| MOV A,#data    | Move immediate to A                   | 2     | 1      | 74       |
| MOV Rn,A       | Move A to register                    | 1     | 1      | F8-FF    |
| MOV Rn,dir     | Move direct byte to register          | 2     | 2      | A8-AF    |
| MOV Rn,#data   | Move immediate to register            | 2     | 1      | 78-7F    |
| MOV dir,A      | Move A to direct byte                 | 2     | 1      | F5       |
| MOV dir,Rn     | Move register to direct byte          | 2     | 2      | 88-8F    |
| MOV dir,dir    | Move direct byte to direct byte       | 3     | 2      | 85       |
| MOV dir,@Ri    | Move indirect memory to direct byte   | 2     | 2      | 86-87    |
| MOV dir,#data  | Move immediate to direct byte         | 3     | 2      | 75       |
| MOV @Ri,A      | Move A to indirect memory             | 1     | 1      | F6-F7    |
| MOV @Ri,dir    | Move direct byte to indirect memory   | 2     | 2      | A6-A7    |
| MOV @Ri,#data  | Move immediate to indirect memory     | 2     | 1      | 76-77    |
| MOV DPTR,#data | Move immediate to data pointer        | 3     | 2      | 90       |
| MOVC A,@A+DPTR | Move code byte relative DPTR to A     | 1     | 2      | 93       |
| MOVC A,@A+PC   | Move code byte relative PC to A       | 1     | 2      | 83       |
| MOVX A,@Ri     | Move external data(A8) to A           | 1     | 2      | E2-E3    |
| MOVX A,@DPTR   | Move external data(A16) to A          | 1     | 2      | E0       |
| MOVX @Ri,A     | Move A to external data(A8)           | 1     | 2      | F2-F3    |
| MOVX @DPTR,A   | Move A to external data(A16)          | 1     | 2      | F0       |
| PUSH dir       | Push direct byte onto stack           | 2     | 2      | C0       |
| POP dir        | Pop direct byte from stack            | 2     | 2      | D0       |
| XCH A,Rn       | Exchange A and register               | 1     | 1      | C8-CF    |
| XCH A,dir      | Exchange A and direct byte            | 2     | 1      | C5       |
| XCH A,@Ri      | Exchange A and indirect memory        | 1     | 1      | C6-C7    |
| XCHD A,@Ri     | Exchange A and indirect memory nibble | 1     | 1      | D6-D7    |

| BOOLEAN    |                                 |       |        |          |
|------------|---------------------------------|-------|--------|----------|
| Mnemonic   | Description                     | Bytes | Cycles | Hex code |
| CLR C      | Clear carry                     | 1     | 1      | C3       |
| CLR bit    | Clear direct bit                | 2     | 1      | C2       |
| SETB C     | Set carry                       | 1     | 1      | D3       |
| SETB bit   | Set direct bit                  | 2     | 1      | D2       |
| CPL C      | Complement carry                | 1     | 1      | B3       |
| CPL bit    | Complement direct bit           | 2     | 1      | B2       |
| ANL C,bit  | AND direct bit to carry         | 2     | 2      | 82       |
| ANL C,/bit | AND direct bit inverse to carry | 2     | 2      | B0       |

|            |                                |   |   |    |
|------------|--------------------------------|---|---|----|
| ORL C,bit  | OR direct bit to carry         | 2 | 2 | 72 |
| ORL C,/bit | OR direct bit inverse to carry | 2 | 2 | A0 |
| MOV C,bit  | Move direct bit to carry       | 2 | 1 | A2 |
| MOV bit,C  | Move carry to direct bit       | 2 | 2 | 92 |

| BRANCHING       |                                          |       |        |          |
|-----------------|------------------------------------------|-------|--------|----------|
| Mnemonic        | Description                              | Bytes | Cycles | Hex code |
| ACALL addr 11   | Absolute jump to subroutine              | 2     | 2      | 11→F1    |
| LCALL addr 16   | Long jump to subroutine                  | 3     | 2      | 12       |
| RET             | Return from subroutine                   | 1     | 2      | 22       |
| RETI            | Return from interrupt                    | 1     | 2      | 32       |
| AJMP addr 11    | Absolute jump unconditional              | 2     | 2      | 01→E1    |
| LJMP addr 16    | Long jump unconditional                  | 3     | 2      | 02       |
| SJMP rel        | Short jump (relative address)            | 2     | 2      | 80       |
| JC rel          | Jump on carry = 1                        | 2     | 2      | 40       |
| JNC rel         | Jump on carry = 0                        | 2     | 2      | 50       |
| JB bit,rel      | Jump on direct bit = 1                   | 3     | 2      | 20       |
| JNB bit,rel     | Jump on direct bit = 0                   | 3     | 2      | 30       |
| JBC bit,rel     | Jump on direct bit = 1 and clear         | 3     | 2      | 10       |
| JMP @A+DPTR     | Jump indirect relative DPTR              | 1     | 2      | 73       |
| JZ rel          | Jump on accumulator = 0                  | 2     | 2      | 60       |
| JNZ rel         | Jump on accumulator ≠ 0                  | 2     | 2      | 70       |
| CJNE A,dir,rel  | Compare A,direct jne relative            | 3     | 2      | B5       |
| CJNE A,#d,rel   | Compare A,immediate jne relative         | 3     | 2      | B4       |
| CJNE Rn,#d,rel  | Compare register, immediate jne relative | 3     | 2      | B8-BF    |
| CJNE @Ri,#d,rel | Compare indirect, immediate jne relative | 3     | 2      | B6-B7    |
| DJNZ Rn,rel     | Decrement register, jnz relative         | 3     | 2      | D8-DF    |
| DJNZ dir,rel    | Decrement direct byte, jnz relative      | 3     | 2      | D5       |

| MISCELLANEOUS |              |       |        |          |
|---------------|--------------|-------|--------|----------|
| Mnemonic      | Description  | Bytes | Cycles | Hex code |
| NOP           | No operation | 1     | 1      | 00       |

| ADDITIONAL INSTRUCTIONS (selected through EO[7:4]) |                                                                                      |       |        |          |
|----------------------------------------------------|--------------------------------------------------------------------------------------|-------|--------|----------|
| Mnemonic                                           | Description                                                                          | Bytes | Cycles | Hex code |
| MOVC @(DPTR++),A                                   | M8051W/M8051EW-specific instruction supporting software download into program memory | 1     | 2      | A5       |
| TRAP                                               | Software break command                                                               | 1     | 1      | A5       |

In the above table, an entry such as E8-EF indicates a continuous block of hex opcodes used for 8 different registers, the register numbers of which are defined by the lowest three bits of the corresponding code. Non-continuous blocks of codes, shown as 11→F1 (for example), are used for absolute jumps and calls, with the top 3 bits of the code being used to store the top three bits of the destination address.

The CJNE instructions use the abbreviation #d for immediate data; other instructions use #data.

## B. Instructions on how to use the input port.

- Error occur status
  - Using compare jump instructions with input port, it could cause error due to the timing conflict inside the MCU.
  - Compare jump Instructions which cause potential error used with input port condition:

```
JB    bit, rel    ; jump on direct bit=1
JNB   bit, rel    ; jump on direct bit=0
JBC   bit, rel    ; jump on direct bit=1 and clear
CJNE  A, dir, rel ; compare A, direct jne relative
DJNZ  dir, rel    ; decrement direct byte, jnz relative
```

- It is only related with Input port. Internal parameters, SFRs and output bit ports don't cause any error by using compare jump instructions.
- If input signal is fixed, there is no error in using compare jump instructions.

- Error status example

```
while(1){
    if (P00==1){ P10=1; }
    else { P10=0; }
    P11^=1;
}
```

```
zzz:  JNB    080.0, xxx ; it possible to be error
      SETB   088.0
      SJMP   yyy
xxx:   CLR    088.0
yyy:   MOV    C,088.1
      CPL    C
      MOV    088.1,C
      SJMP   zzz
```

```
unsigned char ret_bit_err(void)
{
    return !P00;
}
```

```
MOV    R7, #000
JB      080.0, xxx ; it possible to be error
MOV    R7, #001
xxx:    RET
```

- Preventative measures (2 cases)
  - Do not use input bit port for bit operation but for byte operation. Using byte operation instead of bit operation will not cause any error in using compare jump instructions for input port.

```
while(1){
    if ((P0&0x01)==0x01){ P10=1; }
    else { P10=0; }
    P11^=1;
}
```

```
zzz:  MOV    A, 080      ; read as byte
      JNB    0E0.0, xxx  ; compare
      SETB   088.0
      SJMP   yyy
xxx:   CLR    088.0
yyy:   MOV    C,088.1
      CPL    C
      MOV    088.1,C
      SJMP   zzz
```

- If you use input bit port for compare jump instruction, you have to copy the input port as internal parameter or carry bit and then use compare jump instruction.

```
bit tt;  
while(1){  
    tt=P00;  
    if (tt==0){ P10=1;}  
    else {P10=0;}  
    P11^=1;  
}
```

```
zzz:  MOV    C,080.0    ; input port use internal parameter  
      MOV    020.0, C    ; move  
      JB     020.0, xxx    ; compare  
      SETB   088.0  
      SJMP   yyy  
xxx:  CLR     088.0  
yyy:  MOV     C,088.1  
      CPL     C  
      MOV     088.1,C  
      SJMP   zzz
```