1T 8051

8-bit Microcontroller

NuMicro® Family
ML51 Series
Technical Reference Manual

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

TABLE OF CONTENTS

LIST OF FIGURES

List of Tables

1 GENERAL DESCRIPTION

The ML51 is a Flash embedded 1T 8051-based microcontroller. The instruction set of the ML51 is fully compatible with the standard 80C51 with performance enhanced.

The ML51 runs up to 24 MHz at a wide voltage range from 1.8V to 5.5V, and contains up to 64/32/16 Kbytes Flash called APROM for programming code. The ML51 Flash supports In-Application-Programming (IAP) function, which enables on-chip firmware updates. Partial Flash can be optionally configured as Data Flash programmed by IAP and read by IAP or MOVC instruction. The ML51 includes an additional configurable up to 4/3/2/1 Kbytes Flash area called LDROM, in which the Boot Code normally resides for carrying out the In-System-Programming (ISP). To facilitate mass production programming and verification, the Flash is allowed to be programmed and read electronically by parallel Writer/Programmer or In-Circuit-Programming (ICP) with Nu-Link. Once programmed and verified, the programmed code can be protected by the Flash lock mechanism for not being read out by any external programming tool.

The ML51 provides rich peripherals including 256 bytes of SRAM, 4/2/1 Kbytes of auxiliary RAM (XRAM), up to 43 general purpose I/O, two 16-bit Timers/Counters 0/1, one 16-bit Timer2 with three-channel input capture module, one Watchdog Timer (WDT), one Self Wake-up Timer (WKT), one 16-bit auto-reload Timer3 for general purpose or baud rate generator, two UARTs with frame error detection and automatic address recognition, two ISO7816 Smartcard interface, two SPI, two I^2C, 12 enhanced PWM output channels with dead zone control, two analog comparators, eight-channel shared pin interrupt for all I/O ports, and one 12-bit ADC at 500 ksps. There are a total of 30 sources with 4-level-priority interrupts capability.

The ML51 is equipped with four clock sources and supports on-the-fly clock switching via software control. The four clock sources include two sets of external crystal inputs (HXT, LXT), 38.4 kHz internal oscillator, and one 24 MHz internal high-precision ±5% oscillator. The ML51 provides additional power monitoring detection such as power-on reset and 7-level brown-out detection, which stabilizes the power-on/off sequence for a high reliability system design.

The ML51 microcontroller provides 3 power modes to reduce power consumption —Low power run mode, Low power Idle mode, and Power-down mode. In Low power run mode, the power consumption can be down to 15 uA at 38.4 kHz LIRC. In Low power idle mode, CPU processing is suspended by holding the Program Counter. No program code is fetched and run in low power idle mode if the power consumption does not exceed 13 uA. Power-down mode stops the whole system clock for minimum power consumption with the leakage current less than 1 uA. The system clock of the ML51 can also be slowed down by software clock divider, which allows for flexibility between execution performance and power consumption.

Through the high performance of 1T 8051 core, low power performance of ML51 and rich well-designed peripherals, the ML51 benefits for low-power, battery powered devices, general purpose, home appliances, or motor control system.

2 FEATURES

- CPU:
  - Fully static design 8-bit high performance 1T 8051-based CMOS microcontroller.
  - Instruction set fully compatible with MCS-51.
  - 4-priority-level interrupts capability.
  - Dual Data Pointers (DPTRs).
- Operating:
  - Wide supply voltage from 1.8 V to 5.5 V.
  - Wide operating frequency up to 24 MHz
  - Industrial temperature grade: -40 ℃ to +105 ℃.
- Low power features:
  - Normal run typical power consumption 80 μA/ MHz
  - Low power run mode typical power consumption 15 μA
  - Low power Idle mode power consumption does not exceed 13 μA
  - Power down mode typical power consumption less than 1 μA
  - Wake up time from power down mode less than 10 μs (run with HIRC).
- Memory:
  - Up to 64/32/16 Kbytes of APROM for User Code.
  - 4/3/2/1 Kbytes of Flash for loader (LDROM) configure from APROM for In-System-Programmable (ISP)
  - Flash Memory accumulated with pages of 128 Bytes from APROM by In-Application-Programmable (IAP).
  - Flash Memory 100,000 writing cycle endurance.
  - Code lock for security.
  - 256 Bytes on-chip RAM.
  - Additional 4/2/1 Kbytes on-chip auxiliary RAM (XRAM) accessed by MOVX instruction.
- PDMA:
  - Three modes: peripheral-to-memory, memory-to-peripheral, and memory-to-memory transfer.
  - Source address and destination address must be word alignment in all modes.
  - Memory-to-memory mode: transfer length must be word alignment.
  - Peripheral-to-memory and memory-to-peripheral mode: transfer length could be byte alignment.

- Peripheral-to-memory and memory-to-peripheral mode: transfer data width byte alignment.

- Clock sources:
  - 24 MHz high-speed internal oscillator (HIRC) trimmed to ±1% (accuracy at 25 ℃, 3.3 V), ±5% in all conditions.
  - 38.4 kHz low-speed internal oscillator (LIRC) calibrating to ±1% by software from high-speed internal oscillator (HIRC) or external crystal (HXT).
  - External 4~24 MHz crystal (HXT) input for precise timing operation.
  - External 32.768 kHz (LXT) crystal input.
  - On-the-fly clock source switch via software.
  - Programmable system clock divider from 1/2, 1/4, 1/6, 1/8…, up to 1/512.
- Peripherals:
  - Up to 56 general purpose I/O pins. All output pins have individual 2-level slew rate control.
  - 8 channels of GPIO interrupt with variable edge/level detection from all 56 GPIO configure as one of the input source.
  - Standard interrupt pins $\overline{INT0}$ and $\overline{INT1}$ compatible with standard 8051.
  - Two 16-bit Timers/Counters 0 and 1 compatible with standard 8051.
  - One 16-bit Timer 2 with three-channel input capture module.
  - One 16-bit auto-reload Timer 3, which can be the baud rate clock source of UARTs.
  - One programmable Watchdog Timer (WDT) clocked by dedicated 38.4 kHz LIRC.
  - One dedicated Self Wake-up Timer (WKT) for self-timed wake-up for power reduced modes by dedicated 38.4 kHz LIRC or 32.768 kHz LXT.
  - Two full-duplex UART ports with frame error detection and automatic address recognition. TXD and RXD pins of all UART exchangeable via software.
  - Two smart card port supports ISO7816-3 compliant T=0, T=1 and supports full-duplex UART mode .
  - Two SPI port with master and slave modes, up to 6 Mbps when system clock is 24 MHz
  - Two I^2C bus with master and slave modes, up to 400 kbps data rate.
  - Maxima 6 pairs, 12 channels of pulse width modulator (PWM) output, up to 16-bit resolution, with different modes and Fault Brake function for motor control. The 16-bit PWM counter individual used as timer with interrupt.
  - Two comparator supports hysteresis function.
  - One 12-bit ADC, up to 500 ksps (when V_{DD} over then 2.5 V) converting rate, hardware triggered and conversion result compare facilitating motor control.
- Power monitor:

  - Brown-out detection (BOD) with low power mode available, 7-level selection, interrupt or reset options.
  - Power-on reset (POR).
  - Low voltage reset (LVR).
- Strong ESD and EFT immunity.
  - ESD HBM pass 8 KV
  - EFT > ± 4.4 KV
  - Latch-up pass 150 mA
- Development Tools:
  - Nuvoton Nu-Link with KEIL™ and IAR development environment.
  - Nuvoton In-Circuit-Programmer (Nu-Link).
  - Nuvoton In-System-Programming (ISP) via UART.

3 PART INFORMATION

3.1 Package Type

	MSOP10	TSSOP14	TSSOP20	SOP20	QFN20	TSSOP28	SOP28	LQFP32	QFN33
Part No.	ML51BB9AE	ML51DB9AE	ML51FB9AE	ML51OB9AE	ML51XB9AE	ML51EB9AE ML51EC0AE	ML51UB9AE ML51UC0AE	ML51PB9AE ML51PC0AE	ML51TB9AE ML51TC0AE

3.2 ML51 Series Selection Guide

Part Number	Flash (KB)	SRAM (KB)	ISP ROM (KB)[1]	I/O	Timer/	PWM [3]	Analog Comparator	Internal Voltage Reference	PDMA	Connectivity				ADC(12-Bit)	Package
										ISO-7816[2]	UART	SPI	I^2C		
ML51BB9AE	16	1	4	7	4	5	-	-	-	-	2	-	2	2-ch	MSOP10
ML51DB9AE	16	1	4	11	4	6	-	-	-	1	2	1	2	3-ch	TSSOP14
ML51FB9AE	16	1	4	16	4	6	-	-	-	1	2	1	2	6-ch	TSSOP20
ML51OB9AE	16	1	4	16	4	6	-	-	-	1	2	1	2	6-ch	SOP20
ML51XB9AE	16	1	4	17	4	6	-	-	-	1	2	1	2	6-ch	QFN20
ML51EB9AE	16	1	4	24	4	6	-	-	-	1	2	1	2	8-ch	TSSOP28
ML51UB9AE	16	1	4	24	4	6	-	-	-	1	2	1	2	8-ch	SOP28
ML51PB9AE	16	1	4	28	4	6	2	Y	2	1	2	1	2	8-ch	LQFP32
ML51TB9AE	16	1	4	28	4	6	2	Y	2	1	2	1	2	8-ch	QFN33
ML51EC0AE	32	2	4	24	4	6	2	Y	2	1	2	2	2	8-ch	TSSOP28
ML51UC0AE	32	2	4	24	4	6	2	Y	2	1	2	2	2	8-ch	SOP28
ML51PC0AE	32	2	4	28	4	6	2	Y	2	1	2	2	2	8-ch	LQFP32
ML51TC0AE	32	2	4	28	4	6	2	Y	2	1	2	2	2	8-ch	QFN33

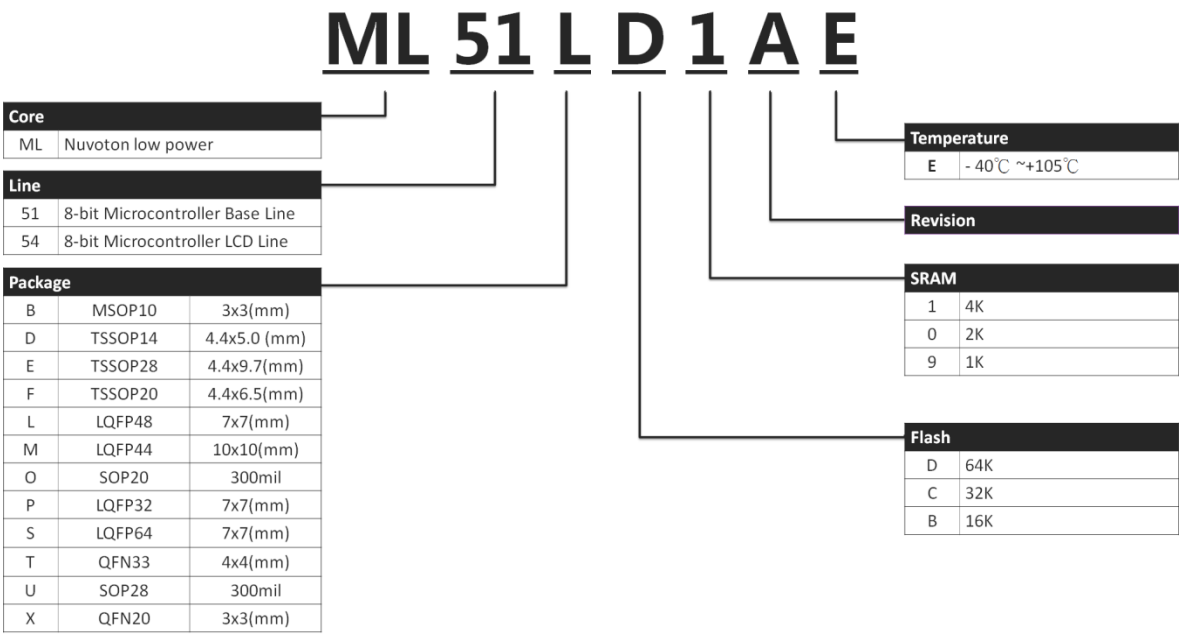
Note:

[1] ISP ROM programmable 1K/2K/3K/4KB Flash for user program loader (LDROM) share from ARPOM.

[2] ISO-7816 configurable as standard UART function.

[3] PWM 6 channel is based on PWM0. PWM1 function is reserved for ML51 64K Flash size product.

3.3 ML51 Series Selection Code



4 PIN CONFIGURATION

4.1 Pin Configuration

Users can find pin configuaration informations in chapter 3 or by using NuTool - PinConfigure. The NuTool - PinConfigure contains all Nuvoton NuMicro® Family chip series with all part number, and helps users configure GPIO multi-function correctly and handily.

4.1.1 ML51 Series Pin Diagram

4.1.1.1 QFN 33-pin Package Pin Diagram

Corresponding Part Number: ML51TC0AE / ML51TB9AE

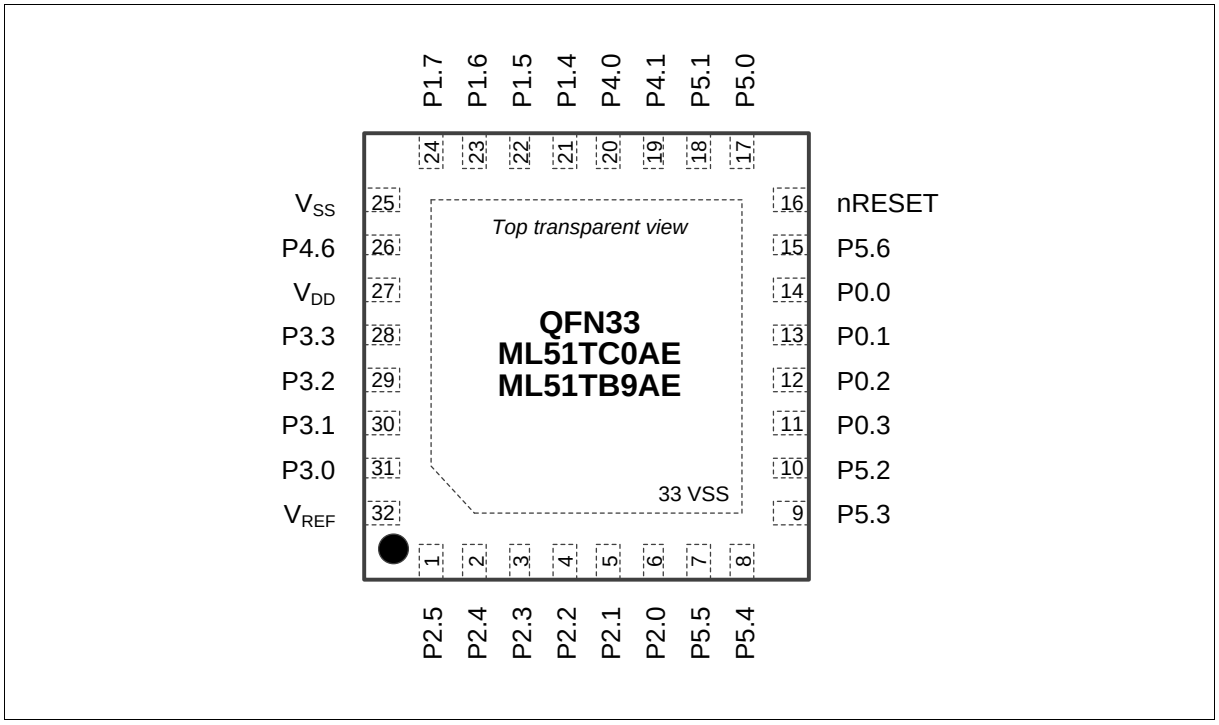


Figure 4.1-1 Pin Assignment of QFN-33 Package

4.1.1.2 LQFP 32-pin Package Pin Diagram

Corresponding Part Number: ML51PD1AE / ML51PC0AE / ML51PB9AE

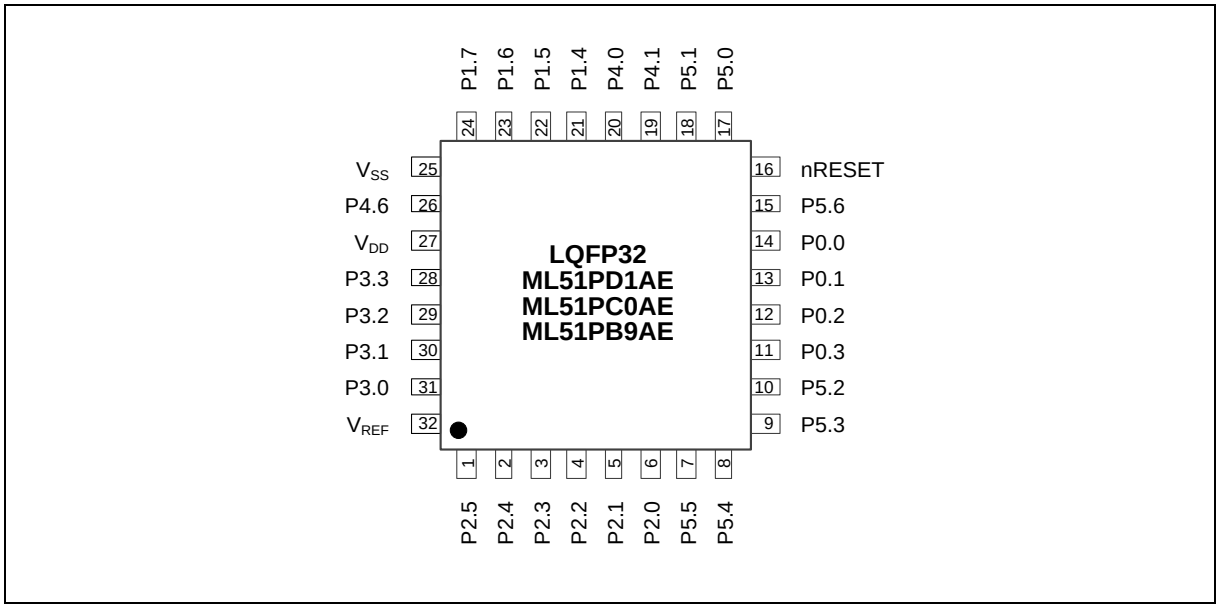


Figure 4.1-2 Pin Assignment of LQFP-32 Package

4.1.1.3 TSSOP 28-pin Package Pin Diagram

Corresponding Part Number: ML51EC0AE / ML51EB9AE

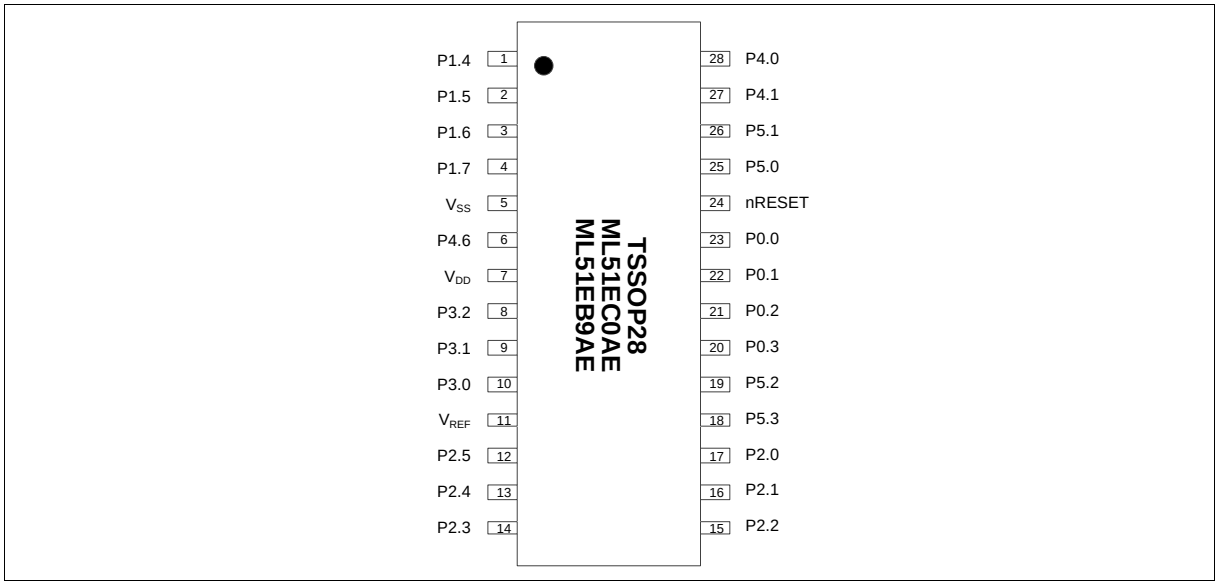


Figure 4.1-3 Pin Assignment of TSSOP-28 Package

4.1.1.4 SOP 28-pin Package Pin Diagram

Corresponding Part Number: ML51UC0AE / ML51UB9AE

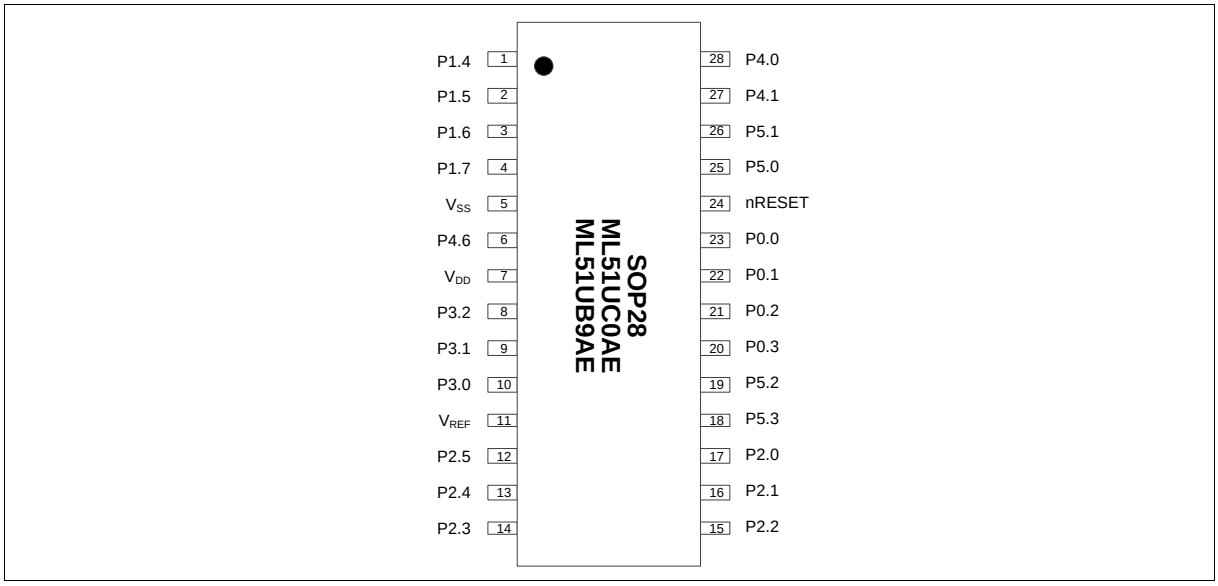


Figure 4.1-4 Pin Assignment of SOP-28 Package

4.1.1.5 TSSOP 20-pin Package Pin Diagram

Corresponding Part Number: ML51FB9AE

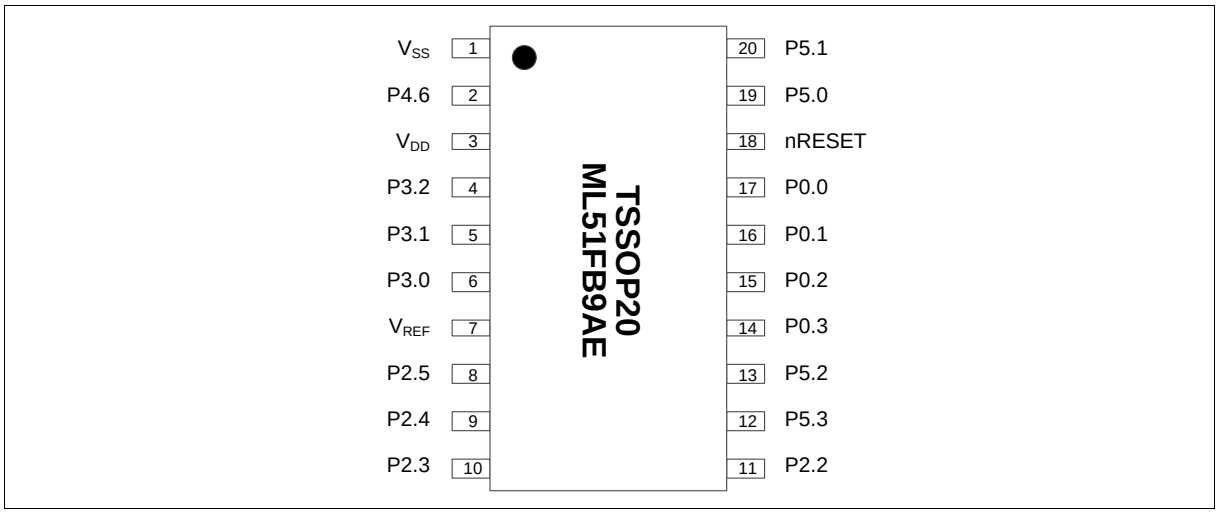


Figure 4.1-5 Pin Assignment of TSSOP-20 Package

4.1.1.6 SOP 20-pin Package Pin Diagram

Corresponding Part Number: ML51OB9AE

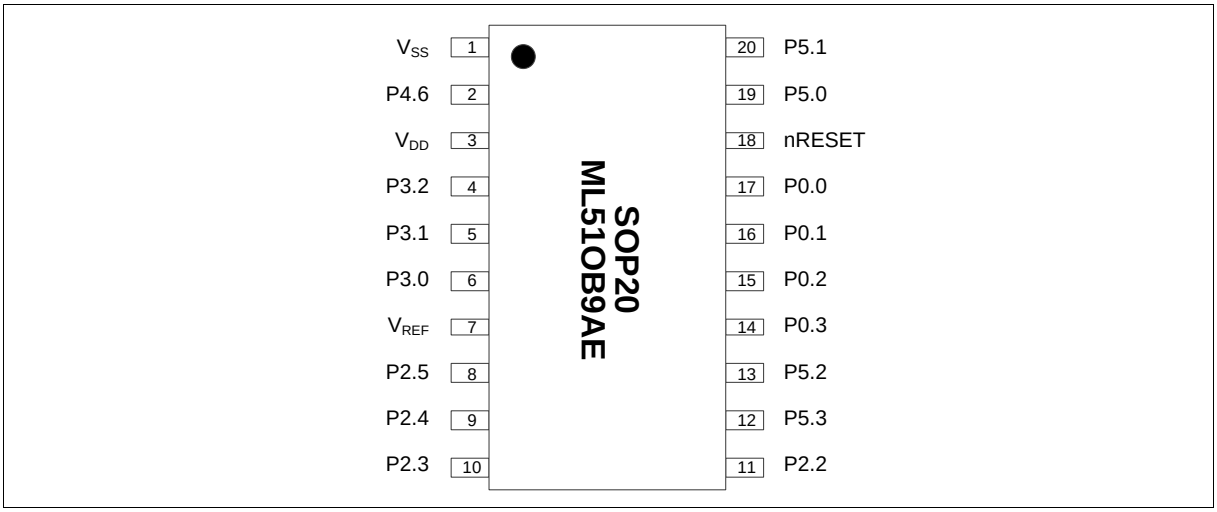


Figure 4.1-6 Pin Assignment of TSSOP-20 Package

4.1.1.7 QFN 20-pin Package Pin Diagram

Corresponding Part Number: ML51XB9AE

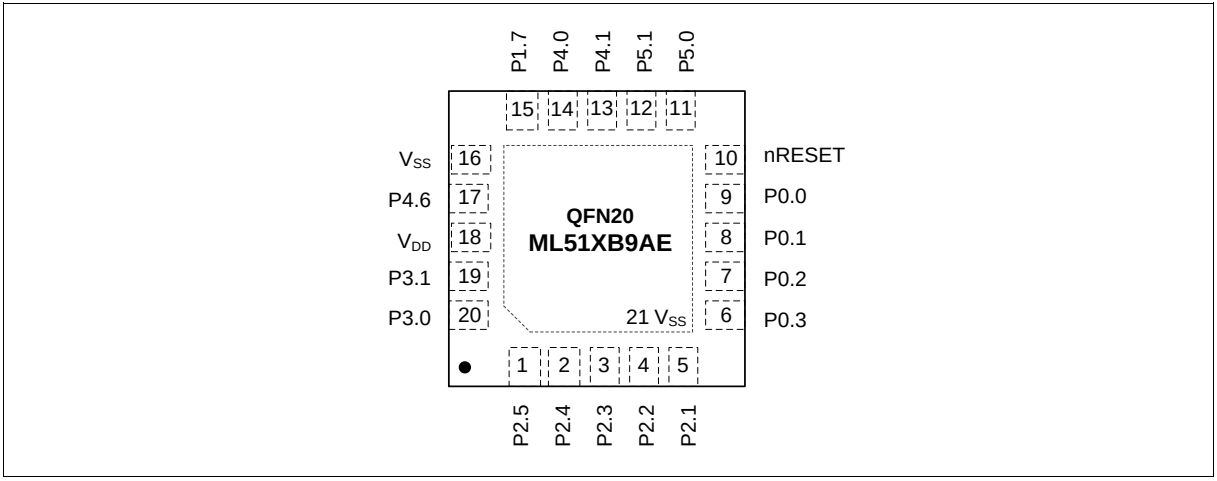


Figure 4.1-7 Pin Assignment of QFN-20 Package

4.1.1.8 TSSOP 14-pin Package Pin Diagram

Corresponding Part Number: ML51DB9AE

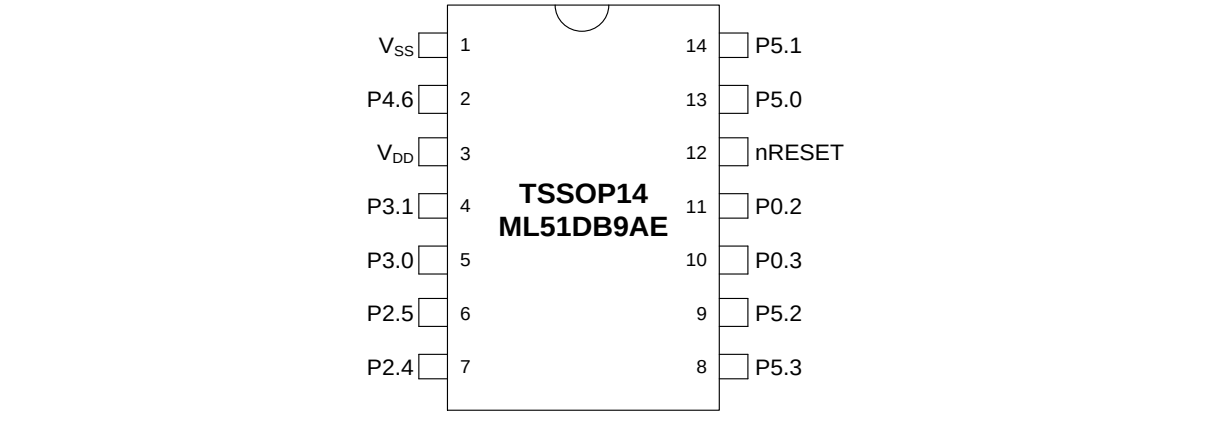


Figure 4.1-8 Pin Assignment of TSSOP-14 Package

4.1.1.9 MSOP 10-pin Package Pin Diagram

Corresponding Part Number: ML51BB9AE

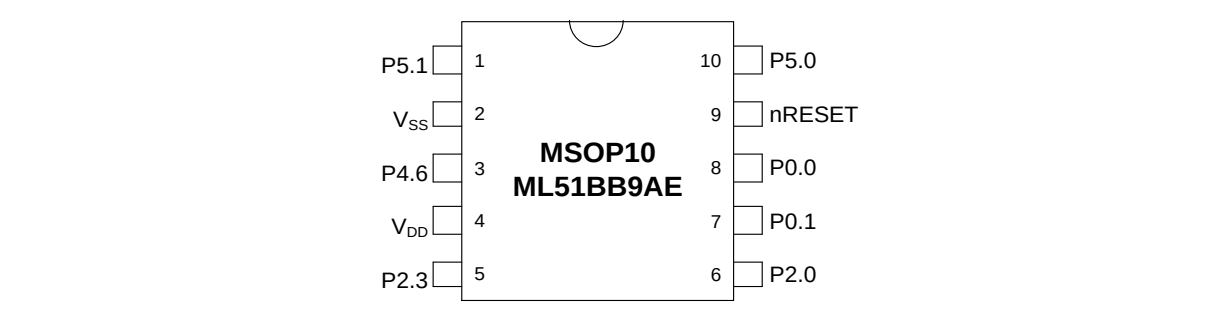


Figure 4.1-9 Pin Assignment of MSOP-10 Package

4.1.2 ML51 Series Function Pin Diagram

4.1.2.1 QFN 33-pin Package Pin Diagram

Corresponding Part Number: ML51TC0AE / ML51TB9AE

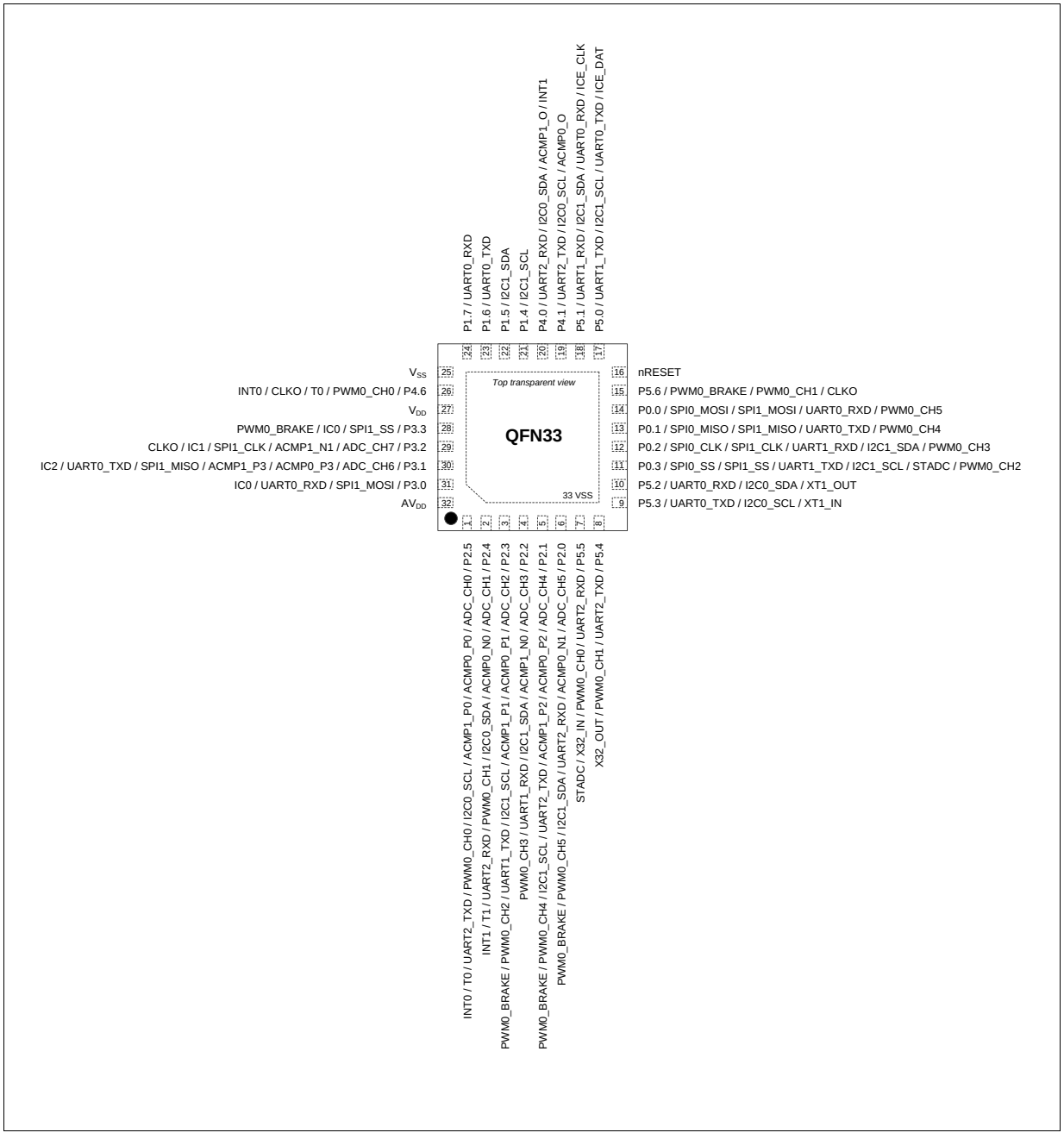


Figure 4.1-10 Multi Function Pin Assignment of QFN-33 Package

4.1.2.2 LQFP 32-pin Package Pin Diagram

Corresponding Part Number: ML51PD1AE / ML51PC0AE / ML51PB9AE

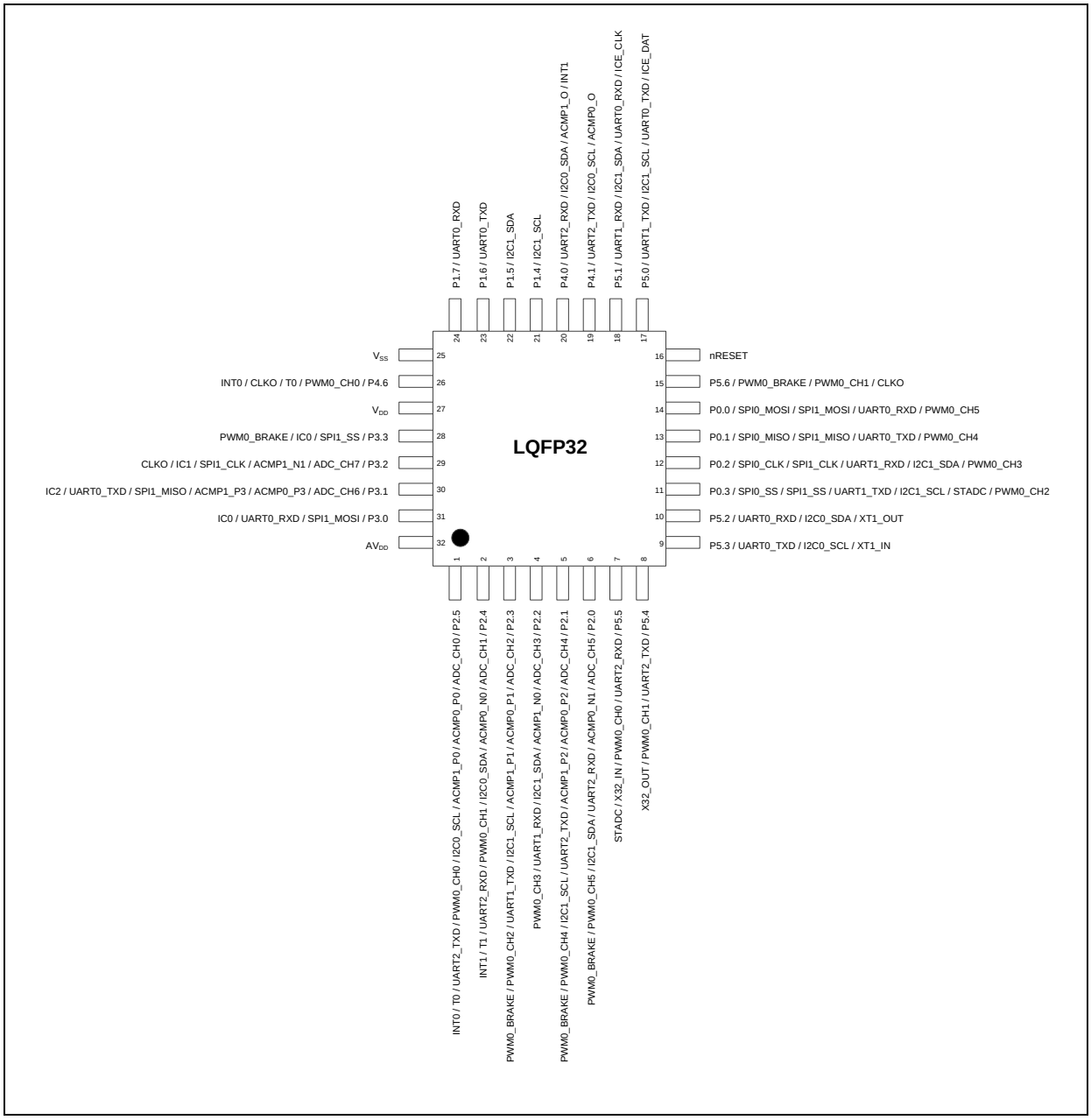


Figure 4.1-11 Multi Function Pin Assignment of LQFP-32 Package

4.1.2.3 TSSOP 28-pin Package Pin Diagram

Corresponding Part Number:ML51EC0AE / ML51EB9AE

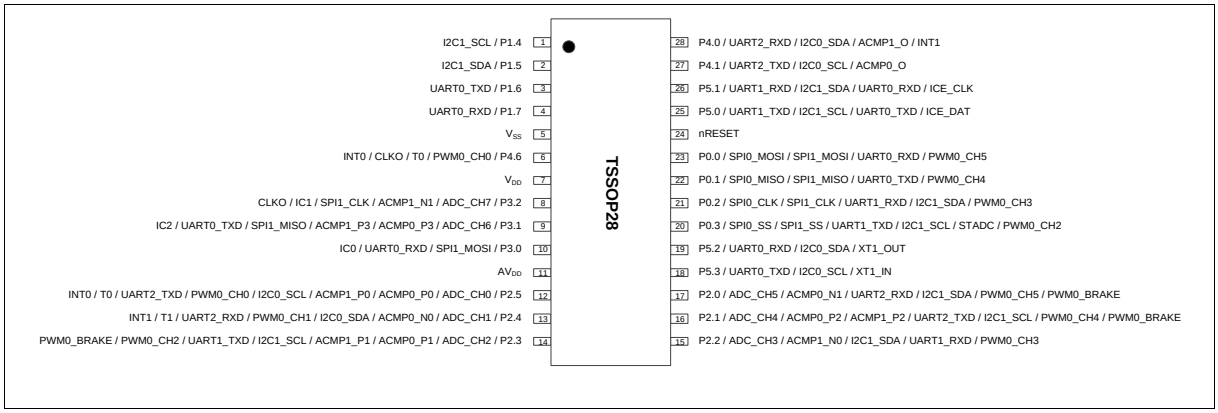


Figure 4.1-12 Multi Function Pin Assignment of TSSOP-28 Package

4.1.2.4 SOP 28-pin Package Pin Diagram

Corresponding Part Number:ML51UC0AE / ML51UB9AE

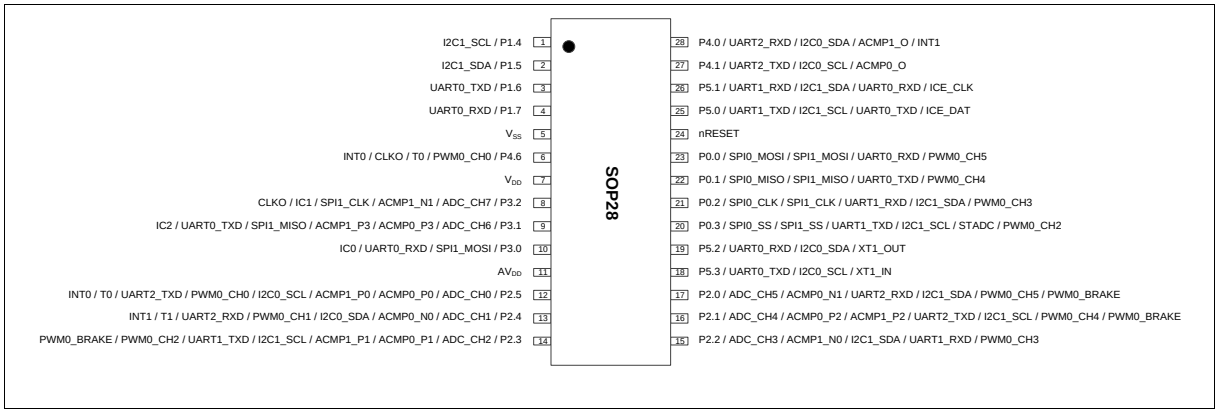


Figure 4.1-13 Multi Function Pin Assignment of SOP-28 Package

4.1.2.5 TSSOP 20-pin Package Pin Diagram

Corresponding Part Number: ML51FB9AE

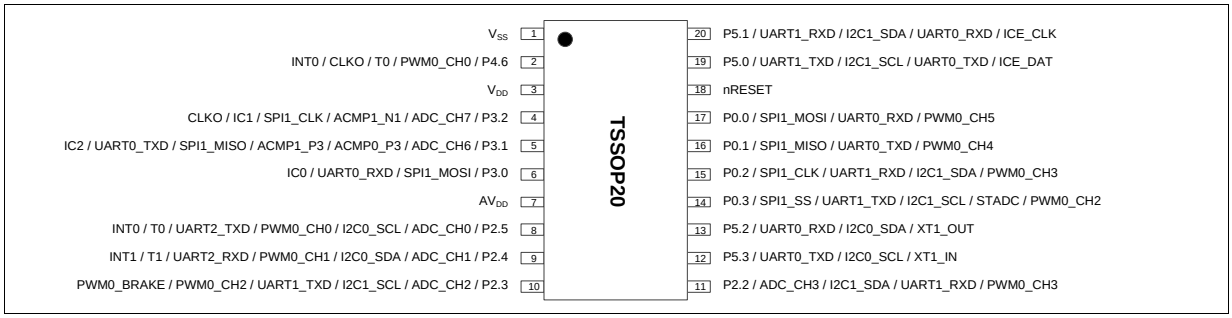


Figure 4.1-14 Multi Function Pin Assignment of TSSOP-20 Package

4.1.2.6 SOP 20-pin Package Pin Diagram

Corresponding Part Number: ML51OB9AE

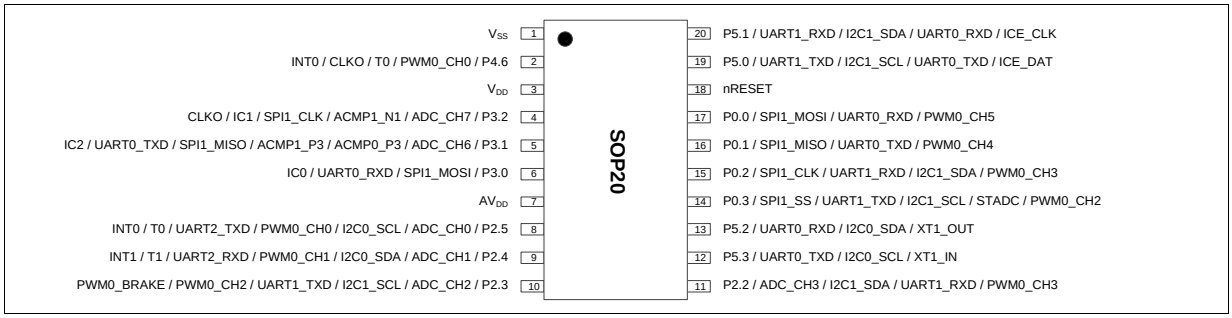


Figure 4.1-15 Multi Function Pin Assignment of SOP-20 Package

4.1.2.7 QFN20 Package

Corresponding Part Number:ML51XB9AE

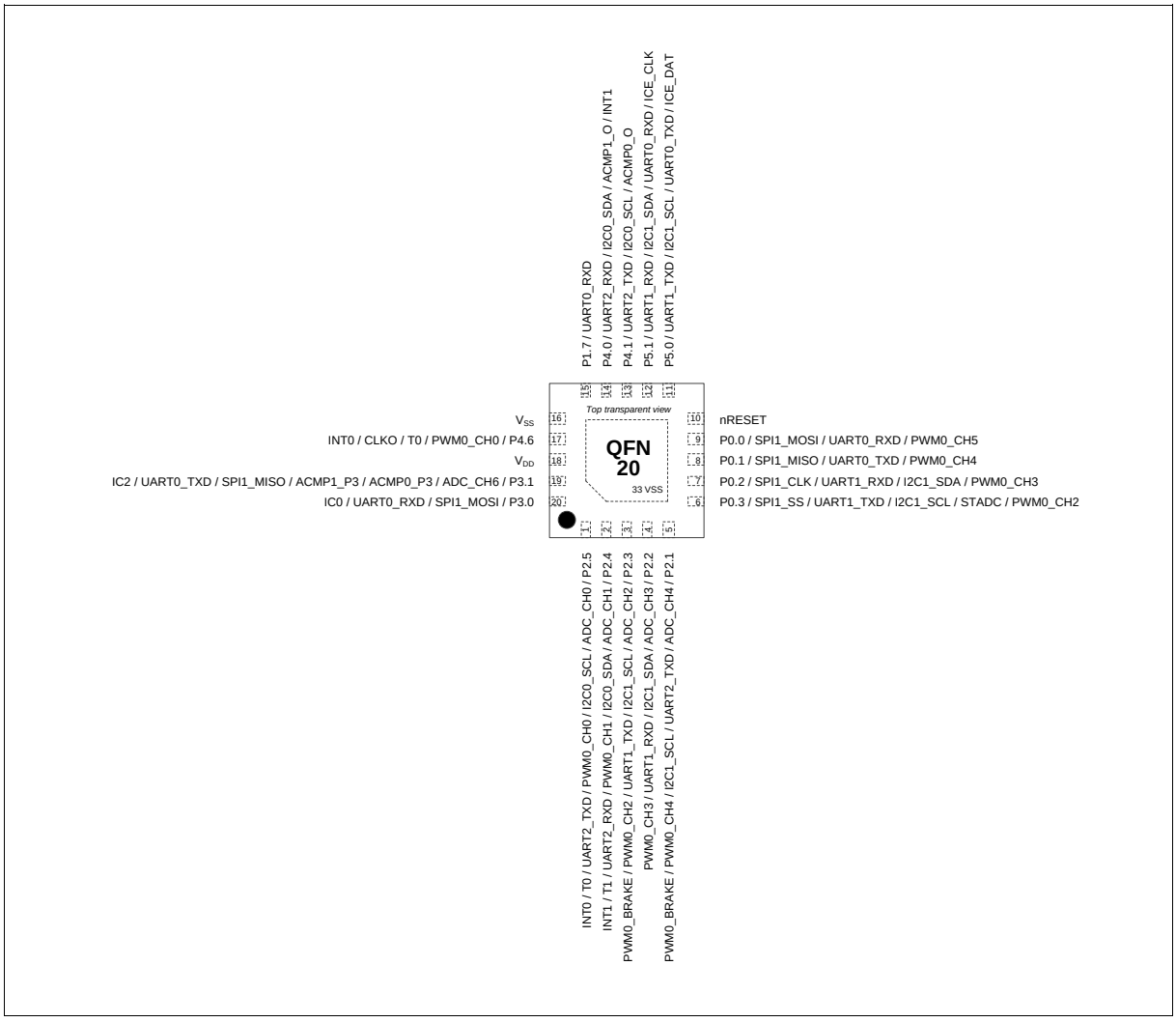


Figure 4.1-16 Multi Function Pin Assignment of QFN-20 Package

4.1.2.8 TSSOP14 Package

Corresponding Part Number: ML51DB9AE

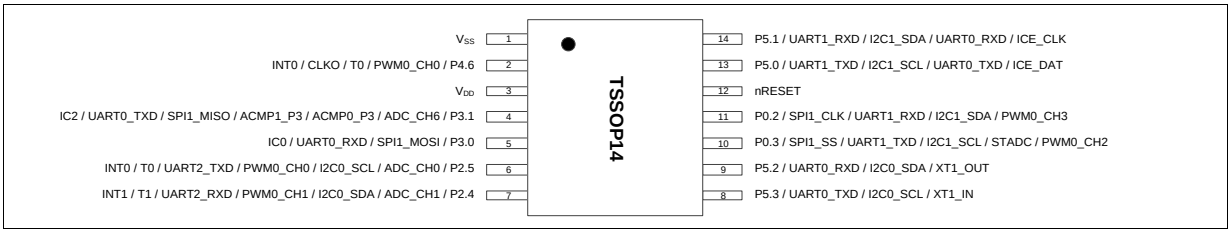


Figure 4.1-17 Multi Function Pin Assignment of Package

4.1.2.9 MSOP10 Package

Corresponding Part Number: ML51BB9AE

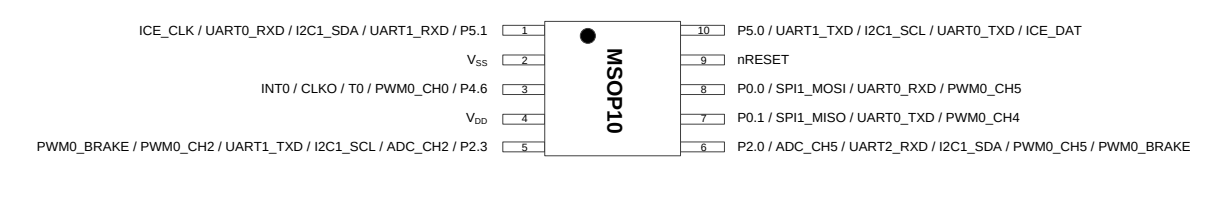


Figure 4.1-18 Pin Assignment of MSOP-10 Package

4.2 Pin Description

The default multi-function pin (MFPx = 0x00) is the GPIO pin. After setting MFPx as non-zero value, the GPIO pin will ANDed with a select function. A simple multi-function demo code for UART0 is given below:

```
SFRS = 0x02;              ;switch to SFR page 2
P3MF10 = 0x56;            ;set P30 as UART0_RXD and P31 as UART0_TXD
```

4.2.1 ML51 Series Pin Description

MSOP10	TSSOP14	QFN20	TSSOP20 SOP20	TSSOP28 SOP28	QFN33 LQFP32	Pin Name	MFP[1]	Description[2] [3] [4]	
-	6	1	8	12	1	P2.5	MFP0	General purpose digital I/O pin.	
						ADC_CH0	MFP1	GPIO	ADC channel 0 analog input.
						ACMP0_P0	MFP1	GPIO	Analog comparator 0 positive input 0 pin.
						ACMP1_P0	MFP1	GPIO	Analog comparator 1 positive input 0 pin.
						I^2C0_SCL	MFP6	GPIO	I^2C0 clock pin.
						PWM0_CH0	MFP11	GPIO	PWM0 channel 0 output.
						UART2_TXD	MFP13	GPIO	UART2 data transmitter output pin.
						SC0_CLK	MFP13	GPIO	Smart Card 0 clock pin.
						T0	MFP14	GPIO	Timer0 counter input/toggle output pin.
						INT0	MFP15	GPIO	External interrupt 0 input pin.
-	7	2	9	13	2	P2.4	MFP0	General purpose digital I/O pin.	
						ADC_CH1	MFP1	GPIO	ADC channel 1 analog input.
						ACMP0_N0	MFP1	GPIO	Analog comparator 0 negative input 0 pin.
						I^2C0_SDA	MFP6	GPIO	I^2C0 data input/output pin.
						PWM0_CH1	MFP11	GPIO	PWM0 channel 1 output.
						UART2_RXD	MFP13	GPIO	UART2 data receiver input pin.
						SC0_DAT	MFP13	GPIO	Smart Card 0 data pin.
						T1	MFP14	GPIO	Timer1 event counter input/toggle output pin.
						INT1	MFP15	GPIO	External interrupt 1 input pin.
5	-	3	10	14	3	P2.3	MFP0	General purpose digital I/O pin.	
						ADC_CH2	MFP1	GPIO	ADC channel 2 analog input.
						ACMP0_P1	MFP1	GPIO	Analog comparator 0 positive input 1 pin.
						ACMP1_P1	MFP1	GPIO	Analog comparator 1 positive input 1 pin.

MSOP10	TSSOP14	QFN20	TSSOP20 SOP20	TSSOP28 SOP28	QFN33 LQFP32	Pin Name	MFP[1]	Description[2] [3] [4]	
						I^2C1_SCL	MFP4	GPIO	I^2C1 clock pin.
						UART1_TXD	MFP6	GPIO	UART1 data transmitter output pin.
						PWM0_CH2	MFP11	GPIO	PWM0 channel 2 output.
						PWM0_BRAKE	MFP13	GPIO	PWM0 Brake input pin.
-	-	4	11	15	4	P2.2	MFP0	General purpose digital I/O pin.	
						ADC_CH3	MFP1	GPIO	ADC channel 3 analog input.
						ACMP1_N0	MFP1	GPIO	Analog comparator 1 negative input 0 pin.
						I^2C1_SDA	MFP4	GPIO	I^2C1 data input/output pin.
						UART1_RXD	MFP6	GPIO	UART1 data receiver input pin.
						PWM0_CH3	MFP11	GPIO	PWM0 channel 3 output.
-	-	5	-	16	5	P2.1	MFP0	General purpose digital I/O pin.	
						ADC_CH4	MFP1	GPIO	ADC channel 4 analog input.
						ACMP0_P2	MFP1	GPIO	Analog comparator 0 positive input 2 pin.
						ACMP1_P2	MFP1	GPIO	Analog comparator 1 positive input 2 pin.
						UART2_TXD	MFP7	GPIO	UART2 data transmitter output pin.
						SC0_CLK	MFP7	GPIO	Smart Card 0 clock pin.
						I^2C1_SCL	MFP9	GPIO	I^2C1 clock pin.
						PWM0_CH4	MFP11	GPIO	PWM0 channel 4 output.
						PWM0_BRAKE	MFP13	GPIO	PWM0 Brake input pin.
6	-	-	-	17	6	P2.0	MFP0	General purpose digital I/O pin.	
						ADC_CH5	MFP1	GPIO	ADC channel 5 analog input.
						ACMP0_N1	MFP1	GPIO	Analog comparator 0 negative input 1 pin.
						UART2_RXD	MFP7	GPIO	UART2 data receiver input pin.
						SC0_DAT	MFP7	GPIO	Smart Card 0 data pin.
						I^2C1_SDA	MFP9	GPIO	I^2C1 data input/output pin.
						PWM0_CH5	MFP11	GPIO	PWM0 channel 5 output.
-	-	-	-	-	7	P5.5	MFP0	General purpose digital I/O pin.	
						UART2_RXD	MFP2	GPIO	UART2 data receiver input pin.
						SC0_DAT	MFP2	GPIO	Smart Card 0 data pin.

MSOP10	TSSOP14	QFN20	TSSOP20 SOP20	TSSOP28 SOP28	QFN33 LQFP32	Pin Name	MFP[1]	Description[2] [3] [4]	
						PWM0_CH0	MFP7	GPIO	PWM0 channel 0 output.
						X32_IN	MFP10	External 32.768 kHz crystal input pin.	
						STADC	MFP11	GPIO	ADC external trigger input.
-	-	-	-	-	8	P5.4	MFP0	General purpose digital I/O pin.	
						UART2_TXD	MFP2	GPIO	UART2 data transmitter output pin.
						SC0_CLK	MFP2	GPIO	Smart Card 0 clock pin.
						PWM0_CH1	MFP7	GPIO	PWM0 channel 1 output.
						X32_OUT	MFP10	External 32.768 kHz crystal output pin.	
-	8	-	12	18	9	P5.3	MFP0	General purpose digital I/O pin.	
						UART0_TXD	MFP3	GPIO	UART0 data transmitter output pin.
						I^2C0_SCL	MFP4	GPIO	I^2C0 clock pin.
						XT1_IN	MFP10	External 4~24 MHz (high speed) crystal input pin.	
-	9	-	13	19	10	P5.2	MFP0	General purpose digital I/O pin.	
						UART0_RXD	MFP3	GPIO	UART0 data receiver input pin.
						I^2C0_SDA	MFP4	GPIO	I^2C0 data input/output pin.
						XT1_OUT	MFP10	External 4~24 MHz (high speed) crystal output pin.	
-	10	6	14	20	11	P0.3	MFP0	General purpose digital I/O pin.	
						SPI0_SS	MFP3	GPIO	SPI0 slave select pin.
						SPI1_SS	MFP4	GPIO	SPI1 slave select pin.
						UART1_TXD	MFP8	GPIO	UART1 data transmitter output pin.
						I^2C1_SCL	MFP9	GPIO	I^2C1 clock pin.
						STADC	MFP11	GPIO	ADC external trigger input.
						PWM0_CH2	MFP13	GPIO	PWM0 channel 2 output.
-	11	7	15	21	12	P0.2	MFP0	General purpose digital I/O pin.	
						SPI0_CLK	MFP3	GPIO	SPI0 serial clock pin.
						SPI1_CLK	MFP4	GPIO	SPI1 serial clock pin.
						UART1_RXD	MFP8	GPIO	UART1 data receiver input pin.
						I^2C1_SDA	MFP9	GPIO	I^2C1 data input/output pin.
						PWM0_CH3	MFP13	GPIO	PWM0 channel 3 output.
7	-	8	16	22	13	P0.1	MFP0	General purpose digital I/O pin.	

MSOP10	TSSOP14	QFN20	TSSOP20 SOP20	TSSOP28 SOP28	QFN33 LQFP32	Pin Name	MFP[1]	Description[2] [3] [4]	
						SPI0_MISO	MFP3	GPIO	SPI0 MISO (Master In, Slave Out) pin.
						SPI1_MISO	MFP4	GPIO	SPI1 MISO (Master In, Slave Out) pin.
						UART0_TXD	MFP7	GPIO	UART0 data transmitter output pin.
						PWM0_CH4	MFP13	GPIO	PWM0 channel 4 output.
8	-	9	17	23	14	P0.0	MFP0	General purpose digital I/O pin.	
						SPI0_MOSI	MFP3	GPIO	SPI0 MOSI (Master Out, Slave In) pin.
						SPI1_MOSI	MFP4	GPIO	SPI1 MOSI (Master Out, Slave In) pin.
						UART0_RXD	MFP7	GPIO	UART0 data receiver input pin.
						PWM0_CH5	MFP13	GPIO	PWM0 channel 5 output.
-	-	-	-	-	15	P5.6	MFP0	General purpose digital I/O pin.	
						PWM0_BRAKE	MFP11	GPIO	PWM0 Brake input pin.
						PWM0_CH1	MFP12	GPIO	PWM0 channel 1 output.
						CLKO	MFP14	GPIO	Clock Out
9	12	10	18	24	16	nRESET		External reset input: active LOW, with an internal pull-up. Set this pin low reset to initial state.	
10	13	11	19	25	17	P5.0	MFP0	General purpose digital I/O pin.	
						UART1_TXD	MFP2	GPIO	UART1 data transmitter output pin.
						I^2C1_SCL	MFP3	GPIO	I^2C1 clock pin.
						UART0_TXD	MFP4	GPIO	UART0 data transmitter output pin.
						ICE_DAT	MFP14	GPIO	Serial wired debugger data pin.
1	14	12	20	26	18	P5.1	MFP0	General purpose digital I/O pin.	
						UART1_RXD	MFP2	GPIO	UART1 data receiver input pin.
						I^2C1_SDA	MFP3	GPIO	I^2C1 data input/output pin.
						ICE_CLK	MFP14	GPIO	Serial wired debugger clock pin.
-	-	13	-	27	19	P4.1	MFP0	General purpose digital I/O pin.	
						UART2_TXD	MFP8	GPIO	UART2 data transmitter output pin.
						SC0_CLK	MFP8	GPIO	Smart Card 0 clock pin.
						I^2C0_SCL	MFP9	GPIO	I^2C0 clock pin.
						ACMP0_O	MFP14	GPIO	Analog comparator 0 output pin.

MSOP10	TSSOP14	QFN20	TSSOP20 SOP20	TSSOP28 SOP28	QFN33 LQFP32	Pin Name	MFP[1]	Description[2] [3] [4]	
-	-	14	-	28	20	P4.0	MFP0	General purpose digital I/O pin.	
						UART2_RXD	MFP8	GPIO	UART2 data receiver input pin.
						SC0_DAT	MFP8	GPIO	Smart Card 0 data pin.
						I^2C0_SDA	MFP9	GPIO	I^2C0 data input/output pin.
						ACMP1_O	MFP14	GPIO	Analog comparator 1 output pin.
						INT1	MFP15	GPIO	External interrupt 1 input pin.
-	-	-	-	1	21	P1.4	MFP0	General purpose digital I/O pin.	
						I^2C1_SCL	MFP4	GPIO	I^2C1 clock pin.
-	-	-	-	2	22	P1.5	MFP0	General purpose digital I/O pin.	
						I^2C1_SDA	MFP4	GPIO	I^2C1 data input/output pin.
-	-	-	-	3	23	P1.6	MFP0	General purpose digital I/O pin.	
						UART0_TXD	MFP3	GPIO	UART0 data transmitter output pin.
-	-	15	-	4	24	P1.7	MFP0	General purpose digital I/O pin.	
						UART0_RXD	MFP3	GPIO	UART0 data receiver input pin.
						SC2_CLK	MFP7	GPIO	Smart Card 2 clock pin.
2	1	16	1	5	25	VSS		Ground pin for digital circuit.	
3	2	17	2	6	26	P4.6	MFP0	General purpose digital I/O pin.	
						PWM0_CH0	MFP12	GPIO	PWM0 channel 0 output.
						T0	MFP13	GPIO	Timer0 counter input/toggle output pin.
						CLKO	MFP14	GPIO	Clock Out
						INT0	MFP15	GPIO	External interrupt 0 input pin.
4	3	18	3	7	27	V_{DD}		Power supply for I/O ports	
-	-	-	-	-	28	P3.3	MFP0	General purpose digital I/O pin.	
						SPI1_SS	MFP4	GPIO	SPI1 slave select pin.
						IC0	MFP13	GPIO	Input Capture channel 0
						PWM0_BRAKE	MFP15	GPIO	PWM0 Brake input pin.
-	-	-	4	8	29	P3.2	MFP0	General purpose digital I/O pin.	
						ADC_CH7	MFP1	GPIO	ADC channel 7 analog input.
						ACMP1_N1	MFP1	GPIO	Analog comparator 1 negative input 1 pin.
						SPI1_CLK	MFP4	GPIO	SPI1 serial clock pin.

MSOP10	TSSOP14	QFN20	TSSOP20 SOP20	TSSOP28 SOP28	QFN33 LQFP32	Pin Name	MFP[1]	Description[2] [3] [4]	
						UART3_RXD	MFP7	GPIO	UART3 data receiver input pin.
						SC1_DAT	MFP7	GPIO	Smart Card 1 data pin.
						IC1	MFP13	GPIO	Input Capture channel 1
						CLKO	MFP14	GPIO	Clock Out
-	4	19	5	9	30	P3.1	MFP0	General purpose digital I/O pin.	
						ADC_CH6	MFP1	GPIO	ADC channel 6 analog input.
						ACMP0_P3	MFP1	GPIO	Analog comparator 0 positive input 3 pin.
						ACMP1_P3	MFP1	GPIO	Analog comparator 1 positive input 3 pin.
						SPI1_MISO	MFP4	GPIO	SPI1 MISO (Master In, Slave Out) pin.
						UART3_TXD	MFP5	GPIO	UART3 data transmitter input pin.
						SC1_DAT	MFP5	GPIO	Smart Card 1 clock pin.
						UART0_TXD	MFP6	GPIO	UART0 data transmitter output pin.
						IC2	MFP13	GPIO	Input Capture channel 2
-	5	20	6	10	31	P3.0	MFP0	General purpose digital I/O pin.	
						SPI1_MOSI	MFP4	GPIO	SPI1 MOSI (Master Out, Slave In) pin.
						UART0_RXD	MFP6	GPIO	UART0 data receiver input pin.
						IC0	MFP13	GPIO	Input Capture channel 0
4	3	18	7	11	32	V_{REF}		ADC reference voltage input. **Note:** For internal V_{REF} appliction circuit this pin needs to be connected with a 0.1uF capacitor to V_{SS}.	
2	1	16	1	5	25	AVSS		Ground pin for analog circuit.	

Note:

[1] MFP* = Multi-function pin.
P0.0 MFP0 means P0MF10[3:0] = 0x0.
P0.1 MFP5 means P0MF10[7:4] = 0x5.

[2] Pin output mode be decided by PxMx, except for ACMP output, F_{SYS} clock output, SPI and Smart Card.

[3] All I/O pins can be configured as a interrupt pin. This feature is not listed in multi-function description. See Section 19“PIN INTERRUPT”.

[4] GPIO = General purpose digital I/O pin.

[5] PWM1 pins reserved for ML51 64K Flash product.

4.2.2 ML51 Series Multi-function Summary Table

Group	Pin Name	GPIO	MFP	Type	Description
ACMP0	ACMP0_N0	P2.4	MFP1	A	Analog comparator 0 negative input 0 pin.
	ACMP0_N1	P2.0	MFP1	A	Analog comparator 0 negative input 1 pin.
	ACMP0_O	P4.1	MFP14	O	Analog comparator 0 output pin.
	ACMP0_P0	P2.5	MFP1	A	Analog comparator 0 positive input 0 pin.
	ACMP0_P1	P2.3	MFP1	A	Analog comparator 0 positive input 1 pin.
	ACMP0_P2	P2.1	MFP1	A	Analog comparator 0 positive input 2 pin.
	ACMP0_P3	P3.1	MFP1	A	Analog comparator 0 positive input 3 pin.
ACMP1	ACMP1_N0	P2.2	MFP1	A	Analog comparator 1 negative input 0 pin.
	ACMP1_N1	P3.2	MFP1	A	Analog comparator 1 negative input 1 pin.
	ACMP1_O	P4.0	MFP14	O	Analog comparator 1 output pin.
	ACMP1_P0	P2.5	MFP1	A	Analog comparator 1 positive input 0 pin.
	ACMP1_P1	P2.3	MFP1	A	Analog comparator 1 positive input 1 pin.
	ACMP1_P2	P2.1	MFP1	A	Analog comparator 1 positive input 2 pin.
	ACMP1_P3	P3.1	MFP1	A	Analog comparator 1 positive input 3 pin.
ADC	ADC_CH0	P2.5	MFP1	A	ADC_ channel analog input.
	ADC_CH1	P2.4	MFP1	A	ADC_ channel analog input.
	ADC_CH2	P2.3	MFP1	A	ADC_ channel analog input.
	ADC_CH3	P2.2	MFP1	A	ADC_ channel analog input.
	ADC_CH4	P2.1	MFP1	A	ADC_ channel analog input.
	ADC_CH5	P2.0	MFP1	A	ADC_ channel analog input.
	ADC_CH6	P3.1	MFP1	A	ADC_ channel analog input.
	ADC_CH7	P3.2	MFP1	A	ADC_ channel analog input.
CLKO	CLKO	P5.7	MFP13	O	Clock Out
		P0.3	MFP14	O	
		P5.6	MFP14	O	
		P4.6	MFP14	O	
		P3.2	MFP14	O	
I2C0	I2C0_SCL	P2.5	MFP6	I/O	I2C0 clock pin.
		P5.3	MFP4	I/O	
		P0.5	MFP9	I/O	

Group	Pin Name	GPIO	MFP	Type	Description
		P4.1	MFP9	I/O	
	I2C0_SDA	P2.4	MFP6	I/O	I2C0 data input/output pin.
		P5.2	MFP4	I/O	
		P0.4	MFP9	I/O	
		P4.0	MFP9	I/O	
I2C1	I2C1_SCL	P2.3	MFP4	I/O	I2C1 clock pin.
		P2.1	MFP9	I/O	
		P0.7	MFP8	I/O	
		P0.3	MFP9	I/O	
		P5.0	MFP3	I/O	
		P4.5	MFP9	I/O	
		P1.4	MFP4	I/O	
	I2C1_SDA	P2.2	MFP4	I/O	I2C1 data input/output pin.
		P2.0	MFP9	I/O	
		P0.6	MFP8	I/O	
		P0.2	MFP9	I/O	
		P5.1	MFP3	I/O	
		P4.4	MFP9	I/O	
		P1.5	MFP4	I/O	
IC0	IC0	P1.3	MFP13	I/O	Input Capture channel 0
		P1.0	MFP13	I/O	
		P3.3	MFP13	I/O	
IC1	IC1	P1.2	MFP13	I/O	Input Capture channel 1
		P3.2	MFP13	I/O	
IC2	IC2	P1.1	MFP13	I/O	Input Capture channel 2
		P3.1	MFP13	I/O	
ICE	ICE_CLK	P5.1	MFP14	I	Serial wired debugger clock pin.
	ICE_DAT	P5.0	MFP14	O	Serial wired debugger data pin.
INT0	INT0	P2.5	MFP15	I	External interrupt 0 input pin.
		P0.6	MFP15	I	
		P4.6	MFP15	I	
INT1	INT1	P2.4	MFP15	I	External interrupt 1 input pin.

Group	Pin Name	GPIO	MFP	Type	Description
		P0.7	MFP15	I	
		P3.6	MFP15	I	
		P4.0	MFP15	I	
PWM0	PWM0_BRAKE	P2.3	MFP13	I	PWM0 Brake input pin.
		P2.1	MFP13	I	
		P2.0	MFP13	I	
		P5.7	MFP10	I	
		P5.6	MFP11	I	
		P3.3	MFP15	I	
	PWM0_CH0	P2.5	MFP11	I/O	PWM0 channel 0 output/capture input.
		P5.5	MFP7	I/O	
		P0.5	MFP13	I/O	
		P4.6	MFP12	I/O	
	PWM0_CH1	P2.4	MFP11	I/O	PWM0 channel 1 output/capture input.
		P5.4	MFP7	I/O	
		P0.4	MFP13	I/O	
		P5.6	MFP12	I/O	
	PWM0_CH2	P2.3	MFP11	I/O	PWM0 channel 2 output/capture input.
		P0.3	MFP13	I/O	
	PWM0_CH3	P2.2	MFP11	I/O	PWM0 channel 3 output/capture input.
		P0.2	MFP13	I/O	
	PWM0_CH4	P2.1	MFP11	I/O	PWM0 channel 4 output/capture input.
		P5.7	MFP12	I/O	
		P0.1	MFP13	I/O	
	PWM0_CH5	P2.0	MFP11	I/O	PWM0 channel 5 output/capture input.
		P3.6	MFP12	I/O	
		P0.0	MFP13	I/O	
SPI0	SPI0_CLK	P0.2	MFP3	I/O	SPI0 serial clock pin.
		P6.2	MFP4	I/O	
	SPI0_MISO	P0.1	MFP3	I/O	SPI0 MISO (Master In, Slave Out) pin.
		P6.1	MFP4	I/O	
	SPI0_MOSI	P0.0	MFP3	I/O	SPI0 MOSI (Master Out, Slave In) pin.

Group	Pin Name	GPIO	MFP	Type	Description
		P6.0	MFP4	I/O	
	SPI0_SS	P0.3	MFP3	I/O	SPI0 slave select pin.
		P6.3	MFP4	I/O	
SPI1	SPI1_CLK	P0.2	MFP4	I/O	SPI1 serial clock pin.
		P3.2	MFP4	I/O	
	SPI1_MISO	P0.1	MFP4	I/O	SPI1 MISO (Master In, Slave Out) pin.
		P3.1	MFP4	I/O	
	SPI1_MOSI	P3.7	MFP5	I/O	SPI1 MOSI (Master Out, Slave In) pin.
		P0.0	MFP4	I/O	
	SPI1_SS	P0.3	MFP4	I/O	SPI1 slave select pin.
		P3.3	MFP4	I/O	
STADC	STADC	P5.5	MFP11	I	ADC external trigger input.
		P0.3	MFP11	I	
T0	T0	P2.5	MFP14	I/O	External count input to Timer/Counter 0 or its toggle output.
		P3.5	MFP14	I/O	
		P4.6	MFP13	I/O	
T1	T1	P2.4	MFP14	I/O	External count input to Timer/Counter 1 or its toggle output.
		P3.4	MFP14	I/O	
		P4.7	MFP13	I/O	
UART0	UART0_RXD	P5.2	MFP3	I	UART0 data receiver input pin.
		P0.6	MFP7	I	
		P0.4	MFP8	I	
		P0.0	MFP7	I	
		P5.1	MFP4	I	
		P6.2	MFP9	I	
		P1.7	MFP3	I	
	UART0_TXD	P5.3	MFP3	O	UART0 data transmitter output pin.
		P0.7	MFP7	O	
		P0.5	MFP8	O	
		P0.1	MFP7	O	
		P5.0	MFP4	O	
		P6.3	MFP9	O	

Group	Pin Name	GPIO	MFP	Type	Description
		P1.6	MFP3	O	
		P3.1	MFP6	O	
UART1	UART1_RXD	P2.2	MFP6	I	UART1 data receiver input pin.
		P1.0	MFP7	I	
		P0.2	MFP8	I	
		P5.1	MFP2	I	
	UART1_TXD	P2.3	MFP6	O	UART1 data transmitter output pin.
		P1.1	MFP7	O	
		P0.3	MFP8	O	
		P5.0	MFP2	O	
UART2	UART2_RXD	P2.4	MFP13	I	UART2 data receiver input pin.
		P2.0	MFP7	I	
		P5.5	MFP2	I	
		P0.1	MFP6	I	
		P4.4	MFP8	I	
		P4.0	MFP8	I	
	UART2_TXD	P2.5	MFP13	O	UART2 data transmitter output pin.
		P2.1	MFP7	O	
		P5.4	MFP2	O	
		P0.0	MFP6	O	
		P4.5	MFP8	O	
		P4.1	MFP8	O	
X32	X32_IN	P5.5	MFP10	I	External 32.768 kHz crystal input pin.
	X32_OUT	P5.4	MFP10	O	External 32.768 kHz crystal output pin.
XT1	XT1_IN	P5.3	MFP10	I	External 4~24 MHz (high speed) crystal input pin.
	XT1_OUT	P5.2	MFP10	O	External 4~24 MHz (high speed) crystal output pin.

5 BLOCK DIAGRAM

5.1 ML51 FULL FUNCTION BLOCK

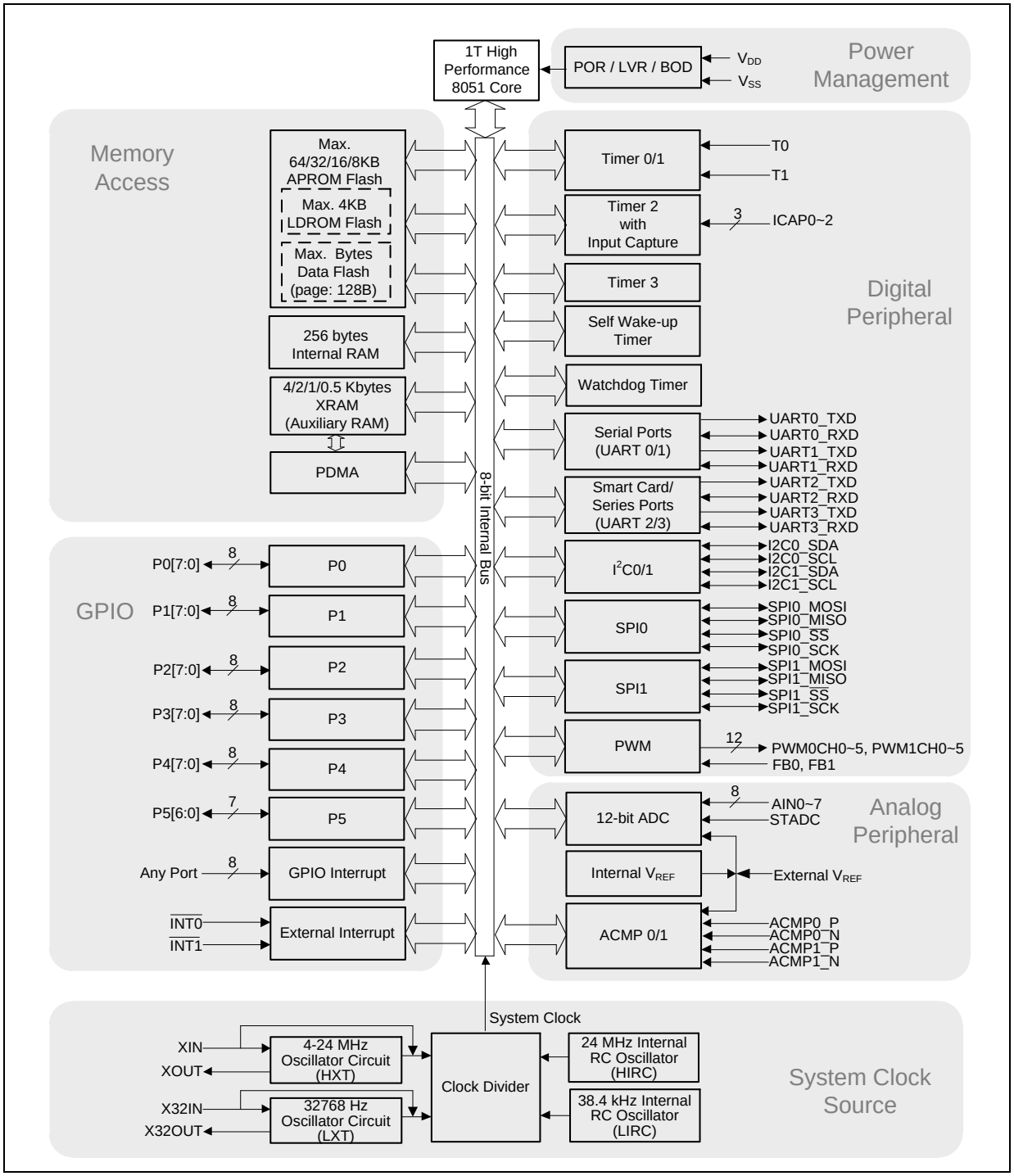


Figure 5.1-1 Functional Block Diagram

6 MEMORY ORGANIZATION

A standard 80C51 based microcontroller divides the memory into two different sections, Program Memory and Data Memory. The Program Memory is used to store the instruction codes, whereas the Data Memory is used to store data or variations during the program execution.

The Data Memory occupies a separate address space from Program Memory. In ML51, there are 256 bytes of internal scratch-pad RAM. For many applications those need more internal RAM, the ML51 provides another on-chip 4\2\1 Kbytes of RAM, which is called XRAM, accessed by MOVX instruction.

The whole embedded Flash, functioning as Program Memory, is divided into three blocks: Application ROM (APROM) normally for User Code, Loader ROM (LDROM) normally for Boot Code, and CONFIG bytes for hardware initialization. Actually, APROM and LDROM function in the same way but have different size. Each block is accumulated page by page and the page size is 128 bytes. The Flash control unit supports Erase, Program, and Read modes. The external writer tools though specific I/O pins, In-Application-Programming (IAP), or In-System-Programming (ISP) can both perform these modes.

6.1 Program Memory

The Program Memory stores the program codes to execute as shown in Figure 6.1-1 ML51 Program Memory Map. After any reset, the CPU begins execution from location 0000H.

To service the interrupts, the interrupt service locations (called interrupt vectors) should be located in the Program Memory. Each interrupt is assigned with a fixed location in the Program Memory. The interrupt causes the CPU to jump to that location with where it commences execution of the interrupt service routine (ISR). External Interrupt 0, for example, is assigned to location 0003H. If External Interrupt 0 is going to be used, its service routine should begin at location 0003H. If the interrupt is not going to be used, its service location is available as general purpose Program Memory.

The interrupt service locations are spaced at an interval of eight bytes: 0003H for External Interrupt 0, 000BH for Timer 0, 0013H for External Interrupt 1, 001BH for Timer 1, etc. If an interrupt service routine is short enough (as is often the case in control applications), it can reside entirely within the 8-Byte interval. However longer service routines should use a JMP instruction to skip over subsequent interrupt locations if other interrupts are in use.

The ML51 provides two internal Program Memory blocks APROM and LDROM. Although they both behave the same as the standard 8051 Program Memory, they play different rules according to their ROM size. The APROM on ML51 can be up to 64 Kbytes. User Code is normally put inside. CPU fetches instructions here for execution. The MOVC instruction can also read this region.

The other individual Program Memory block is called LDROM. The normal function of LDROM is to store the Boot Code for ISP. It can update APROM space and CONFIG bytes. The code in APROM can also re-program LDROM. For ISP details and configuration bit setting related with APROM and LDROM, see Section 27.5“In-System-Programming (ISP)”. Note that APROM and LDROM are hardware individual blocks, consequently if CPU re-boots from LDROM, CPU will automatically re-vector Program Counter 0000H to the LDROM start address. Therefore, CPU accounts the LDROM as an independent Program Memory and all interrupt vectors are independent from APROM.

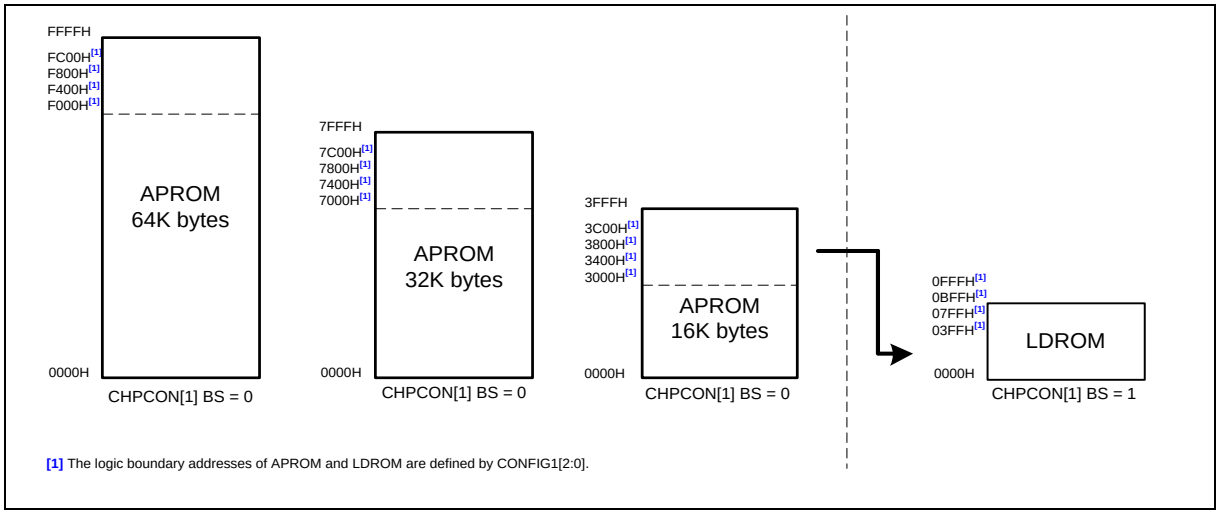


Figure 6.1-1 ML51 Program Memory Map

CONFIG1

7	6	5	4	3	2	1	0
-	-	-	-	-	LDSIZE[2:0]		
-	-	-	-	-	R/W		

Factory default value: 1111 1111b

Bit	Name	Description
2:0	LDSIZE[2:0]	LDROM size select Flash size is 64KB: 111 = No LDROM. APROM is 64 Kbytes. 110 = LDROM is 1 Kbytes. APROM is 63 Kbytes. 101 = LDROM is 2 Kbytes. APROM is 62 Kbytes. 100 = LDROM is 3 Kbytes. APROM is 61 Kbytes. 0xx = LDROM is 4 Kbytes. APROM is 60 Kbytes. Flash size is 32KB: 111 = No LDROM. APROM is 32 Kbytes. 110 = LDROM is 1 Kbytes. APROM is 31 Kbytes. 101 = LDROM is 2 Kbytes. APROM is 30 Kbytes. 100 = LDROM is 3 Kbytes. APROM is 29 Kbytes. 0xx = LDROM is 4 Kbytes. APROM is 28 Kbytes. Flash size is 16KB: 111 = No LDROM. APROM is 16 Kbytes. 110 = LDROM is 1 Kbytes. APROM is 15 Kbytes. 101 = LDROM is 2 Kbytes. APROM is 14 Kbytes. 100 = LDROM is 3 Kbytes. APROM is 13 Kbytes. 0xx = LDROM is 4 Kbytes. APROM is 12 Kbytes.

6.2 Data Memory

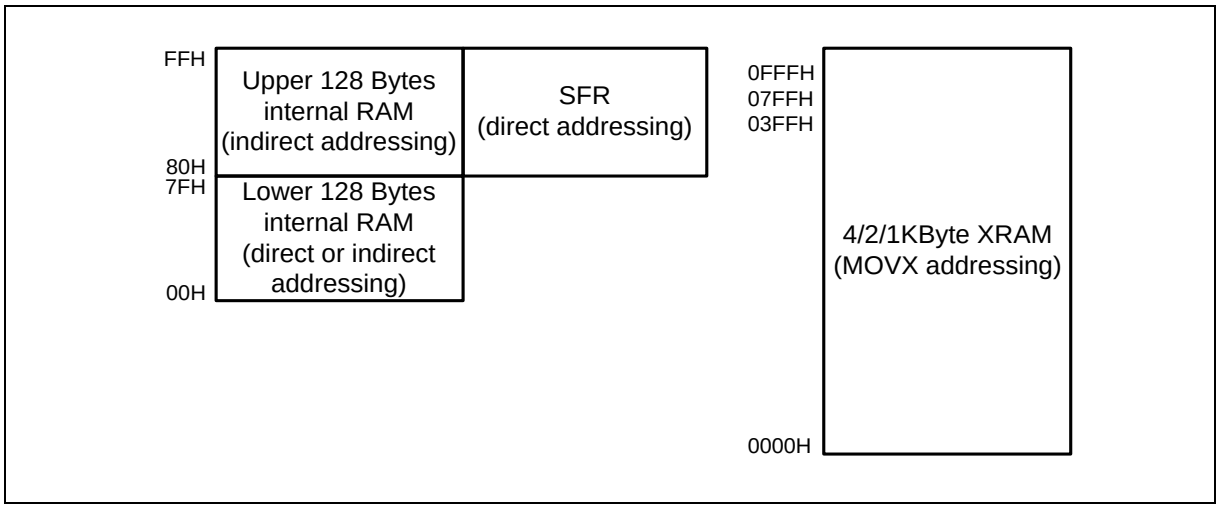


Figure 6.2-1 Data Memory Map

Figure 6.2-1 shows the internal Data Memory spaces available on ML51. Internal Data Memory occupies a separate address space from Program Memory. The internal Data Memory can be divided into three blocks. They are the lower 128 bytes of RAM, the upper 128 bytes of RAM, and the 128 bytes of SFR space. Internal Data Memory addresses are always 8-bit wide, which implies an address space of only 256 bytes. Direct addressing higher than 7FH will access the special function registers (SFR) space and indirect addressing higher than 7FH will access the upper 128 bytes of RAM. Although the SFR space and the upper 128 bytes of RAM share the same logic address, 80H through FFH, actually they are physically separate entities. Direct addressing to distinguish with the higher 128 bytes of RAM can only access these SFR. Sixteen addresses in SFR space are either byte-addressable or bit-addressable. The bit-addressable SFR are those whose addresses end in 0H or 8H.

The lower 128 bytes of internal RAM are present in all 80C51 devices. The lowest 32 bytes as general purpose registers are grouped into 4 banks of 8 registers. Program instructions call these registers as R0 to R7. Two bits RS0 and RS1 in the Program Status Word (PSW[3:4]) select which Register Bank is used. It benefits more efficiency of code space, since register instructions are shorter than instructions that use direct addressing. The next 16 bytes above the general purpose registers (byte-address 20H through 2FH) form a block of bit-addressable memory space (bit-address 00H through 7FH). The 80C51 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.

Either direct or indirect addressing can access the lower 128 bytes space. However, the upper 128 bytes can only be accessed by indirect addressing.

Another application implemented with the whole block of internal 256 bytes RAM is used for the stack. This area is selected by the Stack Pointer (SP), which stores the address of the top of the stack. Whenever a JMP, CALL or interrupt is invoked, the return address is placed on the stack. There is no restriction as to where the stack can begin in the RAM. By default however, the Stack Pointer contains 07H at reset. User can then change this to any value desired. The SP will point to the last used value. Therefore, the SP will be incremented and then address saved onto the stack. Conversely, while popping from the stack the contents will be read first, and then the SP is decreased.

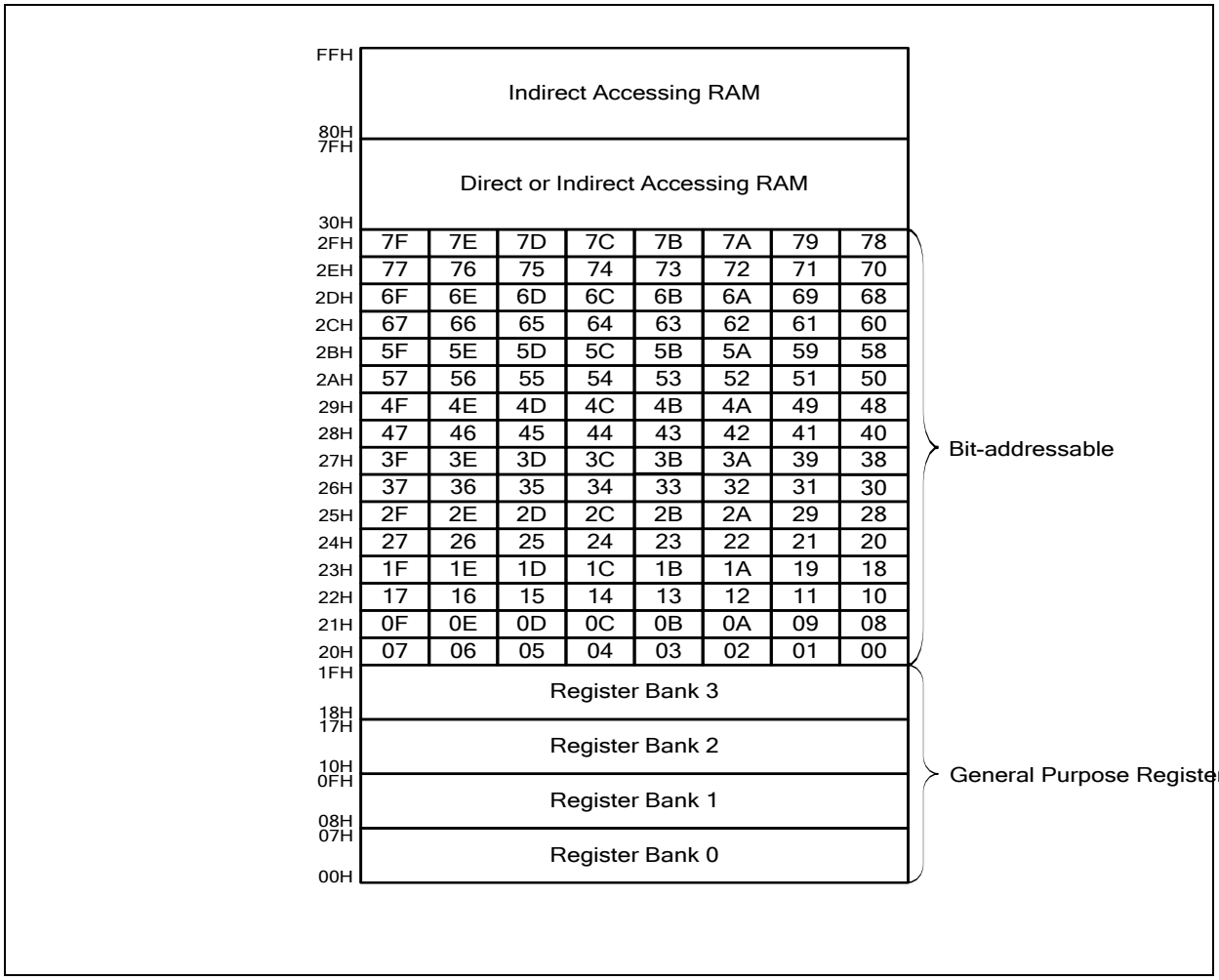


Figure 6.2-2 Internal 256 Bytes RAM Addressing

6.3 On-Chip XRAM

The ML51 provides additional on-chip 4/2/1 Kbytes auxiliary RAM called XRAM to enlarge the RAM space. It occupies the address space from 00H through FFH. The 4/2/1 Kbytes of XRAM are indirectly accessed by move external instruction MOVX @DPTR or MOVX @Ri. (See the demo code below.) Note that the stack pointer cannot be located in any part of XRAM.

XRAM demo code:

Assembler:

```
    MOV    R0,#23H          ;write #5AH to XRAM with address @23H
    MOV    A,#5AH
    MOVX   @R0,A
    MOV    R1,#23H          ;read from XRAM with address @23H
    MOVX   A,@R1
    MOV    DPTR,#0023H      ;write #5BH to XRAM with address @0023H
    MOV    A,#5BH
    MOVX   @DPTR,A
    MOV    DPTR,#0023H      ;read from XRAM with address @0023H
    MOVX   A,@DPTR
```

C51:

```
    unsigned char temp;                    //define data variable
    unsigned char xdata xtemp _at_ 0x23; //define variable at xdata 0x23;
    xtemp = 0x5B;                          // write #5BH to XRAM with address @0023H
    xtemp++;
    temp = xtemp;                          //read from XRAM with address @0023H
```

6.4 Data Flash

ML51 Series Data Flash is shared with APROM or LDROM. Any page of APROM or LDROM can be used as non-volatile data Flash storage and size no need special configuration. The base address of Data Flash is determined by applying IAP, For IAP details, please see Chapter 27 IN-APPLICATION-PROGRAMMING (IAP). All of embedded Flash memory is 128 bytes per page erased.

7 SPECIAL FUNCTION REGISTER (SFR)

The ML51 uses Special Function Registers (SFR) to control and monitor peripherals and their modes. The SFR reside in the register locations 80 to FFH and are accessed by direct addressing only. SFR those end their addresses as 0H or 8H are bit-addressable. It is very useful in cases where user would like to modify a particular bit directly without changing other bits via bit-field instructions. All other SFR are byte-addressable only. The ML51 contains all the SFR presenting in the standard 8051. However some additional SFR are built in. Therefore, some of unused bytes in the original 8051 have been given new functions. The SFR are listed below.

7.1 SFR Page Selection

To accommodate more than 128 SFR in the 0x80 to 0xFF address space, SFR paging has been implemented. By default, all SFR accesses target SFR Page 0. During device initialization, some SFR located on SFR Page 1 may need to be accessed. The register SFRS is used to switch SFR addressing page.

SFRS – SFR Page Selection

Register	SFR Address	Reset Value
SFRS	91H, All pages	0000_0000b

7	6	5	4	3	2	1	0
-	-	-	-	-	-	SFRPAGE	SFRPAGE
-	-	-	-	-	-	R/W	R/W

Bit	Name	Description
1:0	SFRPAGE	**SFR page select** 00 = Instructions access SFR Page 0. 01 = Instructions access SFR Page 1. 10 = Instructions access SFR page 2. 11 = Reserved

Switch SFR page demo code:

```
MOV     SFRS,#00H          ;switch to SFR Page 0
MOV     SFRS,#01H          ;switch to SFR Page 1
MOV     SFRS,#02H          ;switch to SFR page 2
```

Table 7.1-1 Special Function Register (SFR) Memory Map

	Addr	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
0 1 2	F8	S1CON	SPI1CR0 PWM0DTEN P0MF10	SPI1CR1 PWM0DTCNT P0MF32	SPI1SR PWM0MEN P0MF54	SPI1DR PWM0MD P0MF76	DMA1BAH LVRFLTEN P1MF10	EIP1 - P1MF32	EIPH1 LVRDIS P1MF54
0 1 2	F0	B	DMA1TSR - P1MF76	MTM1DA - P2MF10	SPI0CR0 SPI0CR1 P2MF32	SPI0SR - P2MF54	SPI0DR - P2MF76	DMA0BAH - P3MF10	EIPH0 -- P3MF32
0 1 2	E8	I2C1CON	DMA0TSR PICON P3MF54	MTM0DA PINEN P3MF76	DMA1CR PIPEN P4MF10	DMA1MA - P4MF32	DMA1CNT C2L P4MF54	DMA1CCNT C2H P4MF76	EIP0 - P5MF10
0 1 2	E0	ACC	ADCCON1 CAPCON0 P5MF32	ADCCON2 CAPCON1 P5MF54	ADCDLY CAPCON2 P5MF76	ADCBAH C0L -	ADCSN C0H -	ADCCN C1L SC1CR0	ADCSR C1H SC1CR1
0 1 2	D8	P4	SC0DR PWM0PL SC1DR	SC0EGT PWM0C0L SC1EGT	SC0ETURD0 PWM0C1L SC1ETURD0	SC0ETURD1 PWM0C2L SC1ETURD1	SC0IE PWM0C3L SC1IE	SC0IS - SC1IS	SC0TSR PWM0CON1 SC1TSR
0 1 2	D0	PSW	PWM0CON0 PWM0PH PWM1PL	ACMPCR0 PWM0C0H PWM1C0L	ACMPCR1 PWM0C1H PWM1C1L	ACMPSR PWM0C2H PWM1C2L	ACMPVREF PWM0C3H PWM1C3L	SC0CR0 PWM0NP PWM1C4L	SC0CR1 PWM0FBD PWM1C5L
0 1 2	C8	T2CON	T2MOD AUXR1 PWM1PH	PIF RCMP2L PWM1C0H	ADCBAL RCMP2H PWM1C1H	TL2 PWM0C4L PWM1C2H	TH2 PWM0C5L PWM1C3H	ADCMPL AINDIDS PWM1C4H	ADCMPH - PWM1C5H
0 1 2	C0	I2C0CON	I2C0ADDR CKDIV PWM1DTEN	ADCRL P3M1 PWM1DTCNT	ADCRH P3M2 PWM1MEN	T3CON PWM0C4H PWM1MD	RL3 PWM0C5H PWM1CON0	RH3 PORDIS PWM1CON1	TA
0 1 2	B8	IP	SADEN P4M1 -	SADEN1 P4M2 PWM1NP	SADDR1 P4S PWM1FBD	I2C0DAT P4SR PWM1INTC	I2C0STAT P5M1 -	I2C0CLK P5M2 -	I2C0TOC P5S -
0 1 2	B0	P3	P5 P0M1 DMA2TSR	I2C1ADDR0 P0M2 DMA2BAH	I2C1DAT P1M1 DMA2CR	I2STAT1 P1M2 DMA2MAL	I2C1CLK P2M1 DMA2CNT	I2C1TOC P2M2 DMA2CCNT	IPH PWM0INTC MTM2DA
0 1 2	A8	IE	SADDR VRFCON DMA3TSR	WDCON VRFTRIM DMA3BAH	BODCON1 ACMPCR2 DMA3CR	EIP2 P3S DMA3MAL	EIPH2 P3SR DMA3CNT	IAPFD P5SR DMA3CCNT	IAPCN PIPS7 MTM3DA
0 1 2	A0	P2	ADCCON0 PIPS0 I2C0ADDR1	AUXR0 PIPS1 I2C0ADDR2	BODCON0 PIPS2 I2C0ADDR3	IAPTRG PIPS3 I2C1ADDR1	IAPUEN PIPS4 I2C1ADDR2	IAPAL PIPS5 I2C1ADDR3	IAPAH PIPS6 -
0 1 2	98	SCON	SBUF P0S -	SBUF1 P0SR -	EIE0 P1S -	EIE1 P1SR -	RSR P2S -	P2SR - -	CHPCON
0 1 2	90	P1	SFRS	DMA0CR P0UP -	DMA0MA P1UP -	DMA0CNT P2UP -	DMA0CCNT P3UP -	CKSWT P4UP -	CKEN P5UP -
0 1 2	88	TCON	TMOD	TL0 P0DW -	TL1 P1DW -	TH0 P2DW -	TH1 P3DW -	CKCON P4DW -	WKCON P5DW -
0 1 2	80	P0	SP	DPL	DPH	RCTRIM0 LRCTRIM -	RCTRIM1 - -	RWK CWK -	PCON

Unoccupied addresses in the SFR space marked in "-" are reserved for future use. Accessing these areas will have an indeterminate effect and should be avoided.

Table 7.1-2 SFR Definitions And Reset Values

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
EIPH1	Extensive interrupt priority high 1	FFH (0)	-	-	-	-	-	PWKTH		PSH_1	0000 0000b
LVRDIS[4]	LVR Disable	FFH (1)	LVRDIS[7:0]								0000 0000b
P1MF54	P1.5 and P1.4 Multi-Function Select	FFH (2)	P1MF5				P1MF4				0000 0000b
EIP1	Extensive interrupt priority 1	FEH (0)	-	-	-	-	-	PWKT	PT3	PS_1	0000 0000b
P1MF32	P1.3 and P1.2 Multi-Function Select	FEH (2)	P1MF3				P1MF2				0000 0000b
DMA1BAH	PDMA1 Base Address High Byte	FDH (0)	MTMDA [7:4]				XRAMA[7:4]				0000 0000b
LVRFLTEN[4]	LVR Filter Enable	FDH (1)	LVRFLTEN[7:0]								0000 0000b
P1MF10	P1.1 and P1.0 Multi-Function Select	FDH (2)	P1MF1				P1MF0				0000 0000b
SPI1DR	Serial Pereral Data Register	FCH (0)	SPI1DR[7:0]								0000 0000b
PWM0MD	PWM Mask Data	FCH (1)	-	-	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0	0000 0000b
P0MF76	P0.7 and P0.6 Multi-Function Select	FCH (2)	P0MF7				P0MF6				0000 0000b
SPI1SR	Serial Peripheral Status Register	FBH (0)	SPIF	WCOL	SPIOVF	MODF	DISMODF	DISSPIF	TXBFF	-	0000 0000b
PWM0MEN	PWM Mask Enable	FBH (1)	-	-	PMEN5	PMEN4	PMEN3	PMEN2	PMEN1	PMEN0	0000 0000b
P0MF54	P0.5 and P0.4 Multi-Function Select	FBH (2)	P0MF5				P0MF4				0000 0000b
SPI1CR1	Serial Peripheral Control Register 1	FAH (0)	-	-	SPR3	SPR2	TXDMAEN	RXDMAEN	SPIS1	SPIS0	0000 0000b
PWM0DTCNT[4]	PWM Dead-time Counter	FAH (1)	PWM0DTCNT[7:0]								0000 0000b
P0MF32	P0.3 and P0.2 Multi-Function Select	FAH (2)	P0MF3				P0MF2				0000 0000b
SPI1CR0	Serial Peripheral Control Register 0	F9H (0)	SSOE	SPIEN	LSBFE	MSTR	CPOL	CPHA	SPR1	SPR0	0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
PWM0DTEN [4]	PWM Dead-time Enable	F9H (1)	-	-	-	PWMnDTCNT.8	-	PDT45EN	PDT23EN	PDT01EN	0000 0000b
P0MF10	P0.1 and P0.0 Multi-Function Select	F9H (2)	P0MF1				P0MF0				0000 0000b
S1CON	Serial port 1 control	F8H	SM0_1 / FE_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1	0000 0000b
EIPH0	Extensive interrupt priority high	F7H (0)	PT2H	PSPIH	PFBH	PWDTH	PPWMH	PCAPH	PPIH	PI^{2}CH	0000 0000b
P3MF32	P3.3 and P3.2 Multi-Function Select	F7H (2)	P3MF3				P3MF2				0000 0000b
DMA0BAH	PDMA1 Base Address High Byte	F6H (0)	MTMDA [7:4]				XRAMA[7:4]				0000 0000b
P3MF10	P3.1 and P3.0 Multi-Function Select	F6H (2)	P3MF1				P3MF0				0000 0000b
SPI0DR	SPI0 data	F5H (0)	SPDR[7:0]								0000 0000b
P2MF76	P2.7 and P2.6 Multi-Function Select	F5H (2)	P2MF7				P2MF6				0000 0000b
SPI0SR	SPI0 status	F4H (0)	SPIF	WCOL	SPIOVF	MODF	DISMODF	TXBUF	-	-	0000 0000b
P2MF54	P2.5 and P2.4 Multi-Function Select	F4H (2)	P2MF5				P2MF4				0000 0000b
SPI0CR0	SPI 0 control 0	F3H (0)	SSOE	SPIEN	LSBFE	MSTR	CPOL	CPHA	SPR[1:0]		0000 0000b
SPI0CR1	SPI 0 control 1	F3H (1)	-	-	SPR3	SPR2	TXDMAEN	RXDMAEN	SPIS1	SPIS0	0000 0000b
P2MF32	P2.3 and P2.2 Multi-Function Select	F3H (2)	P2MF3				P2MF2				0000 0000b
MTM1DA	Memory to Memory Destination Address Low Byte	F2H (0)	MDAL[7:0]								0000 0000b
P2MF10	P2.1 and P2.0 Multi-Function Select	F2H (2)	P2MF1				P2MF0				0000 0000b
DMA1TSR	PDMAn Transfer Status Register	F1H (0)	-	-	-	-	-	ACT	HDONE	FDONE	0000 0000b
P1MF76	P1.7 and P1.6 Multi-Function Select	F1H (2)	P1MF7				P1MF6				0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
B	B register	F0H	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	0000 0000b
EIP0	Extensive interrupt priority	EFH (0)	PT2	PSPI	PFB	PWDT	PPWM	PCAP	PPI	PI^2C	0000 0000b
P5MF10	P5.1 and P5.0 Multi-Function Select	EFH (2)	P5MF1				P5MF0				0000 0000b
DMA1CCNT	PDMA1 Current Transfer Count	EEH (0)	CCNT[7:0]								0000 0000b
C2H	Input capture 2 high byte	EEH (1)	C2H[7:0]								0000 0000b
P4MF76	P4.7 and P4.6 Multi-Function Select	EEH (2)	P4MF7				P4MF6				0000 0000b
DMA1CNT	PDMA1 Transfer Count	EDH (0)	DMA1CNT [7:0]								0000 0000b
C2L	Input capture 2 low byte	EDH (1)	C2L[7:0]								0000 0000b
P4MF54	P4.5 and P4.4 Multi-Function Select	EDH (2)	P4MF5				P4MF4				0000 0000b
DMA1MA	PDMA1 XRAM Base Address Low Byte	ECH (0)	DMA1MA [7:0]								0000 0000b
P4MF32	P4.3 and P4.2 Multi-Function Select	ECH (1)	P4MF3				P4MF2				0000 0000b
DMA1CR	PDMAn Control Register	EBH (1)	-	PSSEL[2:0]			HIE	FIE	RUN	EN	0000 0000b
PIPEN	Pin interrupt high level /rising edge enable	EBH (1)	PIPEN7	PIPEN6	PIPEN5	PIPEN4	PIPEN3	PIPEN2	PIPEN1	PIPEN0	0000 0000b
P4MF10	P4.5 and P4.4 Multi-Function Select	EBH (2)	P4MF1				P4MF0				0000 0000b
MTM0DA	Memory to Memory Destination Address Low Byte	EAH (0)	MDAL[7:0]								0000 0000b
PINEN	Pin interrupt low level /falling edge enable	EAH (1)	PINEN7	PINEN6	PINEN5	PINEN4	PINEN3	PINEN2	PINEN1	PINEN0	0000 0000b
P3MF76	P3.7 and P3.6Multi-Function Select	EAH (2)	P3MF7				P3MF6				0000 0000b
DMA0TSR	PDMA0 Transfer Status Register	E9H (0)	-	-	-	-	-	ACT	HDONE	FDONE	0000 0000b
PICON	Pin interrupt control	E9H (1)	PIT7	PIT6	PIT5	PIT4	PIT3	PIT2	PIT1	PIT0	0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
P3MF54	P3.5 and P3.4Multi-Function Select	E9H (2)	P3MF5				P3MF4				0000 0000b
I²C1CON	I²C1 control register	E8H	I	I²CEN	STA	STO	SI	AA	-	-	0000 0000b
ADCSR	ADC Status Register	E7H (0)	SLOW	ADCDIV[2:0]			-	CMPHIT	HDONE	FDONE	0000 0000b
C1H	Input capture 1 high byte	E7H (1)	C1H[7:0]								0000 0000b
SC1CR1	SC1 control register 1	E7H (2)	OPE	PBOFF	WLS		TXDMAEN	RXDMAEN	CLKKEEP	UARTEN	0000 0000b
ADCCN	ADC Current Sampling Number	E6H (0)	ADCCN[7:0]								0000 0000b
C1L	Input capture 1 low byte	E6H (1)	C1L[7:0]								0000 0000b
SC1CR0	SC1 Control Register0	E6H (2)	NSB	T	RXBGTEN	CONSEL	AUTOCEN	TXOFF	RXOFF	SCEN	0000 0000b
ADCSN	ADC Sampling Number	E5H (0)	ADCSN[7:0]								0000 0000b
C0H	Input capture 0 high byte	E5H (1)	C0H[7:0]								0000 0000b
ADCBAH	ADC RAM Base Address High byte	E4H (0)	-				ADCBAH[3:0]				0000 0000b
C0L	Input capture 0 low byte	E4H (1)	C0L[7:0]								0000 0000b
ADCDLY	ADC trigger delay	E3H (0)	ADCDLY[7:0]								0000 0000b
CAPCON2	Input Capture Control 2	E3H (1)	-	ENF2	ENF1	ENF0	-	-	-	-	0000 0000b
P5MF76	P5.7 and P5.6Multi-Function Select	E3H (2)	P5MF7				P5MF6				0000 0000b
ADCCON2	ADC control 2	E2H (0)	ADFBEN	ADCMPOP	ADCMPEN	ADCMPO	-	-	-	ADCDLY.8	0000 0000b
CAPCON1	Input Capture Control 1	E2H (1)			CAP2LS[1:0]		CAP1LS[1:0]		CAP0LS[1:0]		0000 0000b
P5MF54	P5.5 and P5.4Multi-Function Select	E2H (2)	P5MF5				P5MF4				0000 0000b
ADCCON1	ADC control 1	E1H (0)	-	STADCPX	-	-	ETGTYP[1:0]		ADCEX	ADCEN	0000 0000b
CAPCON0	Input Capture Control 0	E1H (1)	-	CAPEN2	CAPEN1	CAPEN0	-		CAPF2	CAPF1	0000 0000b
P5MF32	P5.3 and P5.2Multi-Function Select	E1H (2)	P5MF3				P5MF2				0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
ACC	Accumulator	E0H	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	0000 0000b
SC0TSR	SC0 Transfer Status Register	DFH (0)	ACT	BEF	FEF	PEF	TXEMPTY	TXOV	RXEMPTY	RXOV	0000 1010b
PWM0CON1	PWM control 1	DFH (1)	PWMMOD[1:0]		GP	PWMTYP	FBINEN	PWMDIV[2:0]			0000 0000b
SC1TSR	SC Transfer Status Register	DFH (2)	ACT	BEF	FEF	PEF	TXEMPTY	TXOV	RXEMPTY	RXOV	0000 1010b
SC0IS	SC0 Interrupt Status Register	DEH 0)	-	-	-	ACERRIF	BGTIF	TERRIF	TBEIF	RDAIF	0000 0010b
SC1IS	SC1 Control Register 0	DEH (1)	-	-	-	ACERRIF	BGTIF	TERRIF	TBEIF	RDAIF	0000 0010b
SC0IE	SC0 Interrupt Enable Control Register	DDH (0)	-	-	-	ACERRIEN	BGTIEN	TERRIEN	TBEIEN	RDAIEN	0000 0000b
PWM0C3L	PWM0 channel3 duty low byte	DDH (1)	PWM0C3 [7:0]								0000 0000b
SC1IE	SC1 Interrupt Enable Control Register	DDH (2)	-	-	-	ACERRIEN	BGTIEN	TERRIEN	TBEIEN	RDAIEN	0000 0000b
SC0ETURD1	SC0 ETU Rate Divider Register1	DCH (0)	-	SCDIV[2:0]			ETURDIV[11:8]				0011 0001b
PWM0C2L	PWM0 channel2 duty low byte	DCH (1)	PWM0C2 [7:0]								0000 0000b
SC1ETURD1	SC1 ETU Rate Divider Register1	DCH (2)	-	SCDIV[2:0]			ETURDIV[11:8]				0011 0001b
SC0ETURD0	SC0 ETU Rate Divider Register0	DBH (0)	ETURDIV[7:0]								0111 0001b
PWM0C1L	PWM0 channel1 duty low byte	DBH (1)	PWM0C1 [7:0]								0000 0000b
SC1ETURD0	SC1 ETU Rate Divider Register0	DBH (2)	ETURDIV[7:0]								0111 0001b
SC0EGT	SC Extra Guard Time Register	DAH (0)	SCnEGT[7:0]								0000 0000b
PWM0C0L	PWM0 channel0 duty low byte	DAH (1)	PWM0C0 [7:0]								0000 0000b
SC1EGT	SC Extra Guard Time Register	DAH (2)	SCnEGT[7:0]								0000 0000b
SC0DR	SC Data Register	D9H (0)	SC0DR[7:0]								0000 0000b
PWM1PL	PWM period low byte	D9H (1)	PWMP[7:0]								0000 0000b
SC1DR	SC Data Register	D9H (2)	SC1DR[7:0]								0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
P4	Port 4	D8H	P4.7	P4.6	P4.5	P4.4	P4.3	P4.2	P4.1	P4.0	Output latch, 0000 0000b Input, [3] XXXX XXXXb
SC0CR1	SC0 Control Register1	D7H (0)	OPE	PBOFF	WLS		TXDMAEN	RXDMAEN	CLKKEEP	UARTEN	0000 0000b
PWM0FBD	Brake data	D7H (1)	FBF	FBINLS	FBD5	FBD4	FBD3	FBD2	FBD1	FBD0	0000 0000b
PWM1C5L	PWM1 channel5 duty low byte	D7H (2)	PWM1C5 [7:0]								0000 0000b
SC0CR0	SC0 Control Register0	D6H (0)	NSB	T	RXBGTEN	CONSEL	AUTOCEN	TXOFF	RXOFF	SCEN	0000 0000b
PWM0NP	PWM Negative Polarity	D6H (1)	-	-	PNP5	PNP4	PNP3	PNP2	PNP1	PNP0	0000 0000b
PWM1C4L	PWM1 channel4 duty low byte	D6H (2)	PWM1C4 [7:0]								0000 0000b
ACMPVREF	ACMP Reference Voltage Control	D5H (0)	AO1PIV	CRV1CTL[2:0]			AO0PIV	CRV0CTL[2:0]			0000 0000b
PWM0C3H	PWM0 Channel 3 Duty High Byte	D5H (1)	PWM0C3[15:8]								0000 0000b
PWM1C3L	PWM1 Channel 3 Duty Low High Byte	D5H (2)	PWM1C3[7:0]								0000 0000b
ACMPSR	Analog Comparator Status Register	D4H (0)	-	-	-	-	ACMP1O	ACMP1IF	ACMP0O	ACMP0IF	0000 0000b
PWM0C2H	PWM0 Channel 2 Duty High Byte	D4H (1)	PWM0C3[15:8]								0000 0000b
PWM1C2L	PWM1 Channel 2 Duty Low Byte	D4H (2)	PWM1C2[7:0]								0000 0000b
ACMPCR1	Analog Comparator Control Register 1	D3H (0)	POSSEL		NEGSEL		WKEN	HYSEN	ACMPIE	ACMPEN	0000 0000b
PWM0C1H	PWM0 Channel1 Duty High Byte	D3H (1)	PWM0C1[15:8]								0000 0000b
PWM1C1L	PWM1 Channel 1Duty Low Byte	D3H (2)	PWM1C1[7:0]								0000 0000b
ACMPCR0	Analog Comparator Control Register 0	D2H (0)	POSSEL		NEGSEL		WKEN	HYSEN	ACMPIE	ACMPEN	0000 0000b
PWM0C0H	PWM0 Channel0 Duty High Byte	D2H (1)	PWM0C1[15:8]								0000 0000b
PWM1C0L	PWM1 Channel0Duty Low Byte	D2H (2)	PWM1C1[7:0]								0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
PWM0CON0	PWM0 Control register0	D1H (0)	PWMRUN	LOAD	PWMF	CLRPWM	-	-	-	-	0000 0000b
PWM0PH	PWM0 Period High Byte	D1H (1)	PWM0P[15:8]								0000 0000b
PWM1PL	PWM1 Period Low Byte	D1H (2)	PWM1P[7:0]								0000 0000b
PSW	Program status word	D0H	CY	AC	F0	RS1	RS0	OV	-	P	0000 0000b
ADCMPH	ADC Compare High Byte	CFH (0)	ADCMP[15:8]								0000 0000b
PWM1C5H	PWM1 Channel 5 Duty High Byte	CFH (1)	PWM1C5[15:8]								0000 0000b
ADCMPL	ADC Compare Low' Byte	CEH (0)	ADCMP[7:0]								0000 0000b
AINDIDS	ADC Channel Digital Input Disconnect	CEH (1)	AIN7DIDS	AIN6DIDS	AIN5DIDS	AIN4DIDS	AIN3DIDS	AIN2DIDS	AIN1DIDS	AIN0DIDS	0000 0000b
PWM1C4H	PWM1 Channel 4 Duty High Byte	CEH (2)	PWM1C4[15:8]								0000 0000b
TH2	Timer 2 High Byte	CDH (0)	T2[15:8]								0000 0000b
PWM0C5L	PWM0 Channel 5 Duty Low Byte	CDH (1)	PWM0C5[7:0]								0000 0000b
PWM1CON0	PWM1 Control register0	CDH (2)	PWMRUN	LOAD	PWMF	CLRPWM	-	-	-	-	0000 0000b
TL2	Timer 2 Low Byte	CCH (0)	T2[7:0]								0000 0000b
PWM0C4L	PWM0 Channel 4 Duty Low Byte	CCH (1)	PWM0C4[7:0]								0000 0000b
PWM1C2H	PWM1 Channel 2 Duty High Byte	CCH (2)	PWM1C2[15:8]								0000 0000b
ADCBAL	ADC RAM Base Address Low Byte	CBH (0)	ADCBA[7:0]								0000 0000b
RCMP2H	Timer 2 Reload /Compare High Byte	CBH (1)	RCMP2[15:8]								0000 0000b
PWM1C1H	PWM1 Channel 1 Duty High Byte	CBH (2)	PWM1C1[15:8]								0000 0000b
PIF	Pin Interrupt Flags	CAH (0)	PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0	0000 0000b
RCMP2L	Timer 2 compare low byte	CAH (1)	RCMP2[7:0]								0000 0000b
PWM1C0H	PWM1 Channel 0 Duty High Byte	CAH (2)	PWM1C0[15:8]								0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
T2MOD	Timer 2 Mode	C9H (0)	LDEN	T2DIV[2:0]			CAPCR	CMPCR	LDTS[1:0]		0000 0000b
AUXR1	Auxiliary Register 1	C9H (1)	-	-	-	-	-	UART2PX	UART1PX	UART0PX	0000 0000b
PWM1PH	PWM1 Period high Byte	C9H (2)	PWM1P[7:0]								0000 0000b
T2CON	Timer 2 control	C8H	TF2	-	-	-	-	TR2	-	CM/$\overline{RL2}$	0000 0000b
TA	Timed access protection	C7H	TA[7:0]								0000 0000b
RH3	Timer 3 reload high byte	C6H (0)	R3[15:8]								0000 0000b
PORDIS	POR Disable	C6H (1)	PORDIS[7:0]								0000 0000b
PWM1CON1	PWM1 Control 1	C6H (2)	PWMMOD[1:0]		GP	PWMTYP	FBINEN	PWMDIV[2:0]			0000 0000b
RL3	Timer 3 reload low byte	C5H (0)	R3[7:0]								0000 0000b
PWM0C5H	PWM0 Channel 5 Duty High Byte	C5H (1)	PWM0C5[15:8]								0000 0000b
PWM1CON0	PWM1 Control register 0	C5H (2)	PWMRUN	LOAD	PWMF	CLRPWM	-	-	-	-	0000 0000b
T3CON	Timer 3 Control	C4H (0)	SMOD_1	SMOD0_1	BRCK	TF3	TR3	T3PS[2:0]			0000 0000b
PWM0C4H	PWM0 Channel 4 Duty High Byte	C4H (1)	PWM0C4[15:8]								0000 0000b
PWM1MD	PWM 1 Mask Data	C4H (2)	-	-	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0	0000 0000b
ADCRH	ADC result high byte	C3H (0)	ADCR[11:4]								0000 0000b
P3M2	Port 3 Mode Select 2	C3H (1)	P3M2.7	P3M2.6	P3M2.5	P3M2.4	P3M2.3	P3M2.2	P3M2.1	P3M2.0	0000 0000b
PWM1MEN	PWM1 mask enable	C3H (2)	-	-	PMEN5	PMEN4	PMEN3	PMEN2	PMEN1	PMEN0	0000 0000b
ADCRL	ADC result low byte	C2H (0)	-	-	-	-	ADCR[3:0]				0000 0000b
P3M1	Port 3 Mode Select 1	C2H (1)	P3M1.7	P3M1.6	P3M1.5	P3M1.4	P3M1.3	P3M1.2	P3M1.1	P3M1.0	1111 1111B
PWM1DTCNT[4]	PWM Dead-time Counter	C2H (2)	PWM0DTCNT[7:0]								0000 0000b
I2ADDR	I^2C own slave address	C1H (0)	I2ADDR[7:1]							GC	0000 0000b
CKDIV	Clock Divider	C1H (1)	CKDIV[7:0]								0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
PWM1DTEN [4]	PWM Dead-time Enable	C1H (2)	-	-	-	PWM1DTCNT.8	-	PDT45EN	PDT23EN	PDT01EN	0000 0000b
I²CON	I²C0 C control	C0H	-	I²CEN	STA	STO	SI	AA	-	I²CPX	0000 0000b
I²C0TOC	I²C0 time-out	BFH (0)	-	-	-	-	-	I2TOCEN	DIV	I2TOF	0000 0000b
P5S	Port 5 Schmitt Triggered Input	BFH (1)	P5S.7	P5S.6	P5S.5	P5S.4	P5S.3	P5S.2	P5S.1	P5S.0	0000 0000b
I²C0CLK	I²C0 Clock	BEH (0)	I²C0CLK[7:0]								0000 0000b
P5M2	Port 5 Mode Select 2	BEH (1)	P5M2.7	P5M2.6	P5M2.5	P5M2.4	P5M2.3	P5M2.2	P5M2.1	P5M2.0	0000 0000b
I²C0STAT	I²C0 status	BDH (0)	I2STAT[7:3]					0	0	0	1111 1000b
P5M1	Port 5 Mode Select 1	BDH (1)	P5M1.7	P5M1.6	P5M1.5	P5M1.4	P5M1.3	P5M1.2	P5M1.1	P5M1.0	1111 1111b
I²C0DAT	I²C0 data	BCH (0)	I²C0DAT[7:0]								0000 0000b
P4SR	Port 4 Slew Rate Control	BCH (1)	P4SR.7	P4SR.6	P4SR.5	P4SR.4	P4SR.3	P4SR.2	P4SR.1	P4SR.0	0000 0000b
PMW1INTC	PWM1 interrupt control	BCH (12	-	-	INTTYP1	INTTYP0	-	INTSEL2	INTSEL1	INTSEL0	0000 0000b
SADDR1	Slave 1 Address	BBH (0)	SADDR1[7:0]								0000 0000b
P4S	Port 4 Schmitt Triggered Input	BBH (1)	P4S.7	P54S.6	P4S.5	P4S.4	P4S.3	P4S.2	P4S.1	P4S.0	0000 0000b
PWM1FBD	PWM Fault Brake Data	BBH (2)	FBF	FBINLS	FBD5	FBD4	FBD3	FBD2	FBD1	FBD0	0000 0000b
SADEN1	Slave 1 address mask	BAH (0)	SADEN_1[7:0]								0000 0000b
P4M2	Port 4 Mode Select 2	BAH (1)	P4M2.7	P4M2.6	P4M2.5	P4M2.4	P4M2.3	P4M2.2	P4M2.1	P4M2.0	0000 0000b
PWM1NP	PWM Negative Polarity	BAH (2)	-	-	PNP5	PNP4	PNP3	PNP2	PNP1	PNP0	0000 0000b
SADEN	Slave 0 address mask	B9H (0)	SADEN[7:0]								0000 0000b
P4M1	Port 4 Mode Select 1	B9H (1)	P4M1.7	P4M1.6	P4M1.5	P4M1.4	P4M1.3	P4M1.2	P4M1.1	P4M1.0	1111 1111b
IP	Interrupt priority	B8H	-	PADC	PBOD	PS	PT1	PX1	PT0	PX0	0000 0000b
IPH	Interrupt priority high	B7H (0)	-	PADCH	PBODH	PSH	PT1H	PX1H	PT0H	PX0H	0000 0000b
PWM0INTC	PWM0 Interrupt Control	B7H (1)	-	-	INTTYP1	INTTYP0	-	INTSEL2	INTSEL1	INTSEL0	0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
MTM3DA	Memory to Memory 3 Destination Address Low Byte	B7H (2)	MDAL[7:0]								0000 0000b
I²C1TOC	I²C1 Time-out Counter	B6H (0)	-	-	-	-	-	I2TOCEN	DIV	I2TOF	0000 0000b
P2M2	Port 5 Mode Select 2	B6H (1)	P2M2.7	P2M2.6	P2M2.5	P2M2.4	P2M2.3	P2M2.2	P5M2.1	P2M2.0	0000 0000b
DMA2CCNT	PDMA 2 Current Transfer Count	B6H (2)	DMA2CCNT[7:0]								0000 0000b
I²C1CLK	I²C1 Clock	B5H (0)	I²C1CLK[7:0]								0000 0000b
P2M1	Port 2 Mode Select 1	B5H (1)	P2M1.7	P2M1.6	P2M1.5	P2M1.4	P2M1.3	P2M1.2	P2M1.1	P2M1.0	1111 1111b
DMA2CNT	PDMA2 Transfer Count	B5H (2)	DMA2CNT[7:0]								0000 0000b
I2STAT1	I²C1 Status	B4H (0)	I2STAT1[7:3]					0	0	0	1111 1000b
P1M2	P1 mode select 2	B4H (1)	P1M2.7	P1M2.6	P1M2.5	P1M2.4	P1M2.3	P1M2.2	P1M2.1	P1M2.0	0000 0000b
DMA2MA	PDMA2 XRAM Base Address Low Byte	B4H (2)	MAL[7:0]								0000 0000b
I²C1DAT	I²C1 data	B3H (0)	I²C1DAT[7:0]								0000 0000b
P1M1	P1 mode select 1	B3H (1)	P1M1.7	P1M1.6	P1M1.5	P1M1.4	P1M1.3	P1M1.2	P1M1.1	P1M1.0	1111 1111b
DMA2CR	PDMA 2 Control Register	B3H (2)	-	PSSEL[2:0]			HIE	FIE	RUN	EN	0000 0000b
I²C1ADDR	I²C1 Own Slave Address 0	B2H (0)	I²C1ADDR0[7:1]							GC	0000 0000b
P0M2	P0 mode select 2	B2H (1)	P0M2.7	P0M2.6	P0M2.5	P0M2.4	P0M2.3	P0M2.2	P0M2.1	P0M2.0	0000 0000b
DMA2BAH	PDMA2 Base Address High Byte	B2H (2)	MDAH[3:0]				MAH[3:0]				0000 0000b
P5	Port 5	B1H (0)	P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0	Output latch, 0000 0000b Input, [3] XXXX XXXXb
P0M1	P0 mode select 1	B1H (1)	P0M1.7	P0M1.6	P0M1.5	P0M1.4	P0M1.3	P0M1.2	P0M1.1	P0M1.0	1111 1111b
DMA2TSR	PDMAn Transfer Status Register	B1H (2)	-	-	-	-	-	ACT	HDONE	FDONE	0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
P3	Port 3	B0H	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	Output latch, 0000 0000b Input, [3] XXXX XXXXb
IAPCN	IAP control	AFH (0)	IAPA[17:16]		FOEN	FCEN	FCTRL[3:0]				0011 0000b
PIPS7	Pin Interrupt Control register 7	AFH (1)	-	PSEL[2:0]			-	BSEL[2:0]			0000 0000b
MTM3DA	Memory to Memory 3 Destination Address Low Byte	AFH (2)	MTM3DA[7:0]								0000 0000b
IAPFD	IAP Flash data	AEH (0)	IAPFD[7:0]								0000 0000b
P5SR	P5 slew rate	AEH (1)	P5SR.7	P5SR.6	P5SR.5	P5SR.4	P5SR.3	P5SR.2	P5SR.1	P5SR.0	0000 0000b
DMA3CCNT	PDMA 2 Current Transfer Count	AEH (2)	DMA3CCNT[7:0]								0000 0000b
EIPH2	Extensive Interrupt Priority High 2	ADH (0)	-	PDMA3H	PDMA2H	SMC1H	PFB1H	PPWM1H	PI^2C1H	PACMPH	0000 0000b
P3SR	P3 slew rate	ADH (1)	P3SR.7	P3SR.6	P3SR.5	P3SR.4	P3SR.3	P3SR.2	P3SR.1	P3SR.0	0000 0000b
DMA3CNT	PDMA3 Transfer Count	ADH (2)	DMA2CNT[7:0]								0000 0000b
EIP2	Extensive Interrupt Priority 2	ACH (0)	-	PDMA3	PDMA2	SMC1	PFB1	PPWM1	PI^2C1	PACMP	0000 0000b
P3S	Port 3 Schmitt Triggered Input	ACH (1)	P3S.7	P3S.6	P3S.5	P3S.4	P3S.3	P3S.2	P3S.1	P3S.0	0000 0000b
DMA3MA	PDMA3 XRAM Base Address Low Byte	ACH (2)	MAL[7:0]								0000 0000b
BODCON1 [4]	Brown-out Detection Control 1	ABH (0)	-	-	-	-	-	LPBOD[1:0]		BODFLT	POR: 0000 0001b Others: 0000 0UUUb
ACMPCR2	Analog Comparator Control Register 2	ABH (1)	-	-	AO1PEN	AO0PEN	-	-	CRVSSEL	CRVEN	0000 0000b
DMA3CR	PDMA3 Control Register	ABH (2)	-	PSSEL[2:0]			HIE	FIE	RUN	EN	0000 0000b
WDCON[4]	Watchdog Timer control	AAH (0)	WDTR	WDCLR	WDTF	WIDPD	WDTRF	WDPS[2:0]			POR: 0000 0111b WDT: 0000 1UUUb Others: 0000 UUUUb

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
DMA2BAH	PDMA2 XRAM Base Address High Byte	AAH (2)	MTMDA [7:4]				XRAMA[7:4]				0000 0000b
SADDR	Slave 0 address	A9H (0)	SADDR[7:0]								0000 0000b
VRFCON	Internal V_{REF} Control	A9H (1)	-	VRFSEL[2:0]			-	-	ENLOAD	ENVRF	0000 0000b
IE	Interrupt enable	A8H	EA	EADC	EBOD	ES	ET1	EX1	ET0	EX0	0000 0000b
IAPAH	IAP address high byte	A7H (0)	IAPA[15:8]								0000 0000b
PIPS6	Pin Interrupt Control 6	A7H (1)	-	PSEL[2:0]			-	BSEL[2:0]			0000 0000b
IAPAL	IAP address low byte	A6H (0)	IAPA[7:0]								0000 0000b
PIPS5	Pin Interrupt Control 5	A6H (1)	-	PSEL[2:0]			-	BSEL[2:0]			0000 0000b
I²C1ADDR3	I²C1 Own Slave Address 3	A6H (2)	I²C1ADDR3[7:1]							GC	0000 0000b
IAPUEN[4]	IAP update enable	A5H (0)	-	-	-	-	-	CFUEN	LDUEN	APUEN	0000 0000b
PIPS4	Pin Interrupt Control 4	A5H (1)	-	PSEL[2:0]			-	BSEL[2:0]			0000 0000b
I²C1ADDR2	I²C1 Own Slave Address 2	A5H (2)	I²C1ADDR2[7:1]							GC	0000 0000b
IAPTRG[4]	IAP trigger	A4H (0)	-	-	-	-	-	-	-	IAPGO	0000 0000b
PIPS3	Pin Interrupt Control 3	A4H (1)	-	PSEL[2:0]			-	BSEL[2:0]			0000 0000b
I²C1ADDR1	I²C1 Own Slave Address 1	A4H (2)	I²C1ADDR1[7:1]							GC	0000 0000b
BODCON0 [4]	Brown-out Detection Control 0	A3H (0)	BODEN	BOV[2:0]			BOF	BORST	BORF	BOS[6]	POR: CCCC XC0Xb BOD: UUUU XU1Xb Others: UUUU XUUXb
PIPS2	Pin Interrupt Control 2	A3H (1)	-	PSEL[2:0]-			-	BSEL[2:0]			0000 0000b
I²C0ADDR3	I²C0 Own Slave Address 3	A3H (2)	I²C0ADDR3[7:1]							GC	0000 0000b
AUXR0	Auxiliary register 0	A2H (0)	SWRF	RSTPINF	HFRF	HFIF	GF2	-	0	DPS	0000 0000b
PIPS1	Pin Interrupt Control 1	A2H (1)	-	PSEL[2:0]			-	BSEL[2:0]			0000 0000b
I²C0ADDR2	I²C0 Own Slave Address 2	A2H (2)	I²C0ADDR2[7:1]							GC	0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
ADCCON0	ADC Control register 0	A0H (0)	ADCF	ADCS	ETGSEL1	ETGSEL0	ADCHS3	ADCHS2	ADCHS1	ADCHS0	0000 0000b
PIPS0	Pin Interrupt Control 1	A1H (1)	-	PSEL[2:0]			-	BSEL[2:0]			0000 0000b
I²C0ADDR1	I²C0 Own Slave Address 1	A1H (2)	I²C0ADDR1[7:1]							GC	0000 0000b
P2	Port 2	A0H	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	Output latch, 0000 0000b Input,[3] XXXX XXXXb
CHPCON[4]	Chip control	9FH	SWRST	IAPFF	-	-	-	-	BS[5]	IAPEN	Software, 0000 00U0b Others, 0000 00C0b
P2SR	P2 slew rate	9EH (1)	P2SR.7	P2SR.6	P2SR.5	P2SR.4	P2SR.3	P2SR.2	P2SR.1	P2SR.0	0000 0000b
RSR	Reset Flag Register	9DH (0)	LVRF	PORF	HFRF	POF	RSTPINF	BOR	WDTRF	SWRF	1101 0000b
P2S	Port 2 Schmitt Triggered Input	9DH (1)	P2S.7	P2S.6	P2S.5	P2S.4	P2S.3	P2S.2	P2S.1	P2S.0	0000 0000b
EIE1	Extensive interrupt enable 1	9CH (0)	EFB1	EPWM1	EI²C1	ESPI1	EHFI	EWKT	ET3	ES1	0000 0000b
P1SR	P2 slew rate	9CH (1)	P2SR.7	P2SR.6	P2SR.5	P2SR.4	P2SR.3	P2SR.2	P2SR.1	P2SR.0	0000 0000b
EIE0	Extensive interrupt enable 1	9BH (0)	ET2	ESPI0	EFB0	EWDT	EPWM0	ECAP	EPI	EI²C0	0000 0000b
P1S	Port 1 Schmitt Triggered Input	9BH (1)	P1SR.7	P1SR.6	P1SR.5	P1SR.4	P1SR.3	P1SR.2	P1SR.1	P1SR.0	0000 0000b
SBUF1	Serial Port 1 Data Buffer	9AH (0)	SBUF1[7:0]								0000 0000b
P0UP	Port 0 Pull-Up resister control	9AH (1)	P0UP.7	P0UP.6	P0UP.5	P0UP.4	P0UP.3	P0UP.2	P0UP.1	P0UP.0	0000 0000b
SBUF	Serial Port 0Data Buffer	99H (0)	SBUF[7:0]								0000 0000b
P0S	Port 0 Schmitt Triggered Input	99H (1)	P0SR.7	P0SR.6	P0SR.5	P0SR.4	P0SR.3	P0SR.2	P0SR.1	P0SR.0	0000 0000b
SCON	Serial port 0 control	98H	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI	0000 0000b
CKEN[4]	Clock enable	97H (0)	EHXTEN	ELXTEN	HIRCEN	LIRCEN	ECLKEN	-	-	CKSWTF	0011 0000b
P5UP	Port5 Pull-Up resister control	97H (1)	P5UP.7	P5UP.6	P5UP.5	P5UP.4	P4UP.3	P5UP.2	P5UP.1	P5UP.0	0000 0000b
CKSWT[4]	Clock Switch	96H (0)	HXTST	LXTST	HIRCST	LIRCST	ECLKST	OSC[2:0]			0011 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
P4UP	Port4 Pull-Up resister control	96H (1)	P4UP.7	P4UP.6	P4UP.5	P4UP.4	P4UP.3	P4UP.2	P4UP.1	P4UP.0	0000 0000b
DMA0CCNT	PDMA1 Current Transfer Count	95H (0)	DMA0CCNT [7:0]								0000 0000b
P3UP	Port 3 Pull-Up resister control	95H (1)	P3UP.7	P3UP.6	P3UP.5	P3UP.4	P3UP.3	P3UP.2	P3UP.1	P3UP.0	0000 0000b
DMA0CNT	PDMA0 Transfer Count	94H (0)	DMA1CNT [7:0]								0000 0000b
P2UP	Port 2 Pull-Up resister control	94H (1)	P2UP.7	P2UP.6	P2UP.5	P2UP.4	P2UP.3	P2UP.2	P2UP.1	P2UP.0	0000 0000b
DMA0MA	PDMA0 XRAM Base Address Low Byte	93H (0)	MAL[7:0]								0000 0000b
P1UP	Port1 Pull-Up resister control	93H (1)	P1UP.7	P1UP.6	P1UP.5	P1UP.4	P1UP.3	P1UP.2	P1UP.1	P1UP.0	0000 0000b
DMA0CR	PDMA 0 Control Register	92H (0)	-	PSSEL[2:0]			HIE	FIE	RUN	EN	0000 0000b
P0SR	P2 slew rate	92H (1)	P0SR.7	P0SR.6	P0SR.5	P0SR.4	P0SR.3	P0SR.2	P0SR.1	P0SR.0	0000 0000b
SFRS	SFR page selection	91H	-	-	-	-	-	-	SFRPAGE[1:0]		0000 0000b
P1	Port 1	90H	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	Output latch, 0000 0000b Input, [3] XXXX XXXXb
WKCON	Self Wake-up Timer control	8FH (0)	-	-	-	WKTF	WKTR	WKPS[2:0]			0000 0000b
P5DW	Port 5 Pull-Down resister control	8FH (1)	P5DW.7	P5DW.6	P5DW.5	P5DW.4	P5DW.3	P5DW.2	P5DW.1	P5DW.0	0000 0000b
CKCON	Clock Control	8EH (0)	FASTWK	PWMCKS	T1OE	T1M	T0M	T0OE	CLOEN	-	1000 0000b
P4DW	Port4 Pull-Down resister control	8EH (1)	P4DW.7	P4DW.6	P4DW.5	P4DW.4	P4DW.3	P4DW.2	P4DW.1	P4DW.0	0000 0000b
TH1	Timer 1 high byte	8DH (0)	TH1[7:0]								0000 0000b
P3DW	Port3 Pull-Down resister control	8DH (1)	P3DW.7	P3DW.6	P3DW.5	P3DW.4	P3DW.3	P3DW.2	P3DW.1	P3DW.0	0000 0000b
TH0	Timer 0 high byte	8CH (0)	TH0[7:0]								0000 0000b
P2DW	Port2 Pull-Down resister control	8CH (1)	P2DW.7	P2DW.6	P2DW.5	P2DW.4	P2DW.3	P2DW.2	P2DW.1	P2DW.0	0000 0000b
TL1	Timer 1 low byte	8BH (0)	TL1[7:0]								0000 0000b
P1DW	Port1 Pull-Down resister control	8BH (1)	P1DW.7	P1DW.6	P1DW.5	P1DW.4	P1DW.3	P1DW.2	P1DW.1	P1DW.0	0000 0000b

Sym.	Definition	Addr (Page)	MSB							LSB[1]	Reset Value[2]
TL0	Timer 0 low byte	8AH (0)	TL0[7:0]								0000 0000b
P0DW	Port0 Pull-Down resister control	BAH (1)	P0DW.7	P0DW.6	P0DW.5	P0DW.4	P0DW.3	P0DW.2	P0DW.1	P0DW.0	0000 0000b
TMOD	Timer 0 and 1 mode	89H	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0	0000 0000b
TCON	Timer 0 and 1control	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	0000 0000b
PCON	Power control	87H	SMOD	SMOD0	LPR	POF	GF1	GF0	PD	IDL	POR, 0001 0000b Others, 000U 0000b
RWK	Self Wake-up Timer reload byte	86H (0)	RWK[7:0]								0000 0000b
CWK	Self Wake-up Timer Current count value	86H (1)	CWK[7:0]								0000 0000b
RCTRIM1	Internal RC trim value low byte	85H (0)	-	-	-	-	-	-	-	HIRCTRIM[0]	0000 000Xb
RCTRIM0	Internal RC trim value high byte	84H (0)	HIRCTRIM[8:1]								XXXX XXXXb
LIRCTRIM[4]	Low Speed Internal Oscillator Trim	84H (1)	LIRCTRIM[7:0]								XXXX XXXXb
DPH	Data pointer high byte	83H	DPTR[15:8]								0000 0000b
DPL	Data pointer low byte	82H	DPTR[7:0]								0000 0000b
SP	Stack pointer	81H	SP[7:0]								0000 0111b
P0	Port 0	80H	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	Output latch, 0000 0000b Input,[3] XXXX XXXXb

[1] ( ) item means the page of SFRs. Without special define means all pages accessable.
[2] Reset value symbol description. 0: logic 0; 1: logic 1; U: unchanged; C: see [5]; X: see [3], [6], and [7].
[3] All I/O pins are default input-only mode (floating) after reset. After reset ICE_DAT and ICE_CLK pin will keep quasi mode with pull high resister 600 LIRC clock before change to input mode.
[4] These SFRs have TA protected writing.
[5] These SFRs have bits those are initialized according to CONFIG values after specified resets.
[6] BOF reset value depends on different setting of CONFIG2 and V_{DD} voltage level.
[7] BOS is a read-only flag decided by V_{DD} level while brown-out detection is enabled.
[8] All PWM1 register is reserved for ML51 64K Flash body.

Note: Bits marked in "-" are reserved for future use. They must be kept in their own initial states. Accessing these bits may cause an unpredictable effect.

7.2 All SFR Description

Note:

1. All SFRs reset value show as following means U-unchanged; C-initialized by CONFIG; X- base on real chip status.

2. All PWM1 register is reserved for ML51 64K flash product. All ACMP / internal V_{REF} register is reserved for 64K and 32K flash product.

Pn – Port

Register	SFR Address	Reset Value
P0	80H, All pages, Bit-addressable	1111_1111 b
P1	90H, All pages, Bit-addressable	1111_1111 b
P2	A0H, All pages, Bit-addressable	1111_1111 b
P3	B0H, All pages, Bit-addressable	1111_1111 b
P4	D8H, All pages, Bit-addressable	1111_1111 b
P5	B1H, Page 0	0111_1111 b

7	6	5	4	3	2	1	0
Pn.7	Pn.6	Pn.5	Pn.4	Pn.3	Pn.2	Pn.1	Pn.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	Pn[7:0]	Port n Port n is an maximum 8-bit general purpose I/O port.

SP – Stack Pointer

Register	SFR Address	Reset Value
SP	81H, All pages	0000_0111b

7	6	5	4	3	2	1	0
SP[7:0]							
R/W							

Bit	Name	Description
7:0	SP[7:0]	Stack pointer The Stack Pointer stores the scratch-pad RAM address where the stack begins. It is incremented before data is stored during PUSH or CALL instructions. Note that the default value of SP is 07H. This causes the stack to begin at location 08H.

DPL – Data Pointer Low Byte

Register	SFR Address	Reset Value
DPL	82H, All pages	0000_0000b

7	6	5	4	3	2	1	0
DPL[7:0]							
R/W							

Bit	Name	Description
7:0	DPL[7:0]	Data pointer low byte This is the low byte of 16-bit data pointer. DPL combined with DPH serve as a 16-bit data pointer DPTR to access indirect addressed RAM or Program Memory. DPS (AUXR0.0) bit decides which data pointer, DPTR or DPTR1, is activated.

DPH – Data Pointer High Byte

Register	SFR Address	Reset Value
DPH	83H, All pages	0000_0000b

7	6	5	4	3	2	1	0
DPH[7:0]							
R/W							

Bit	Name	Description
7:0	DPH[7:0]	Data pointer high byte This is the high byte of 16-bit data pointer. DPH combined with DPL serve as a 16-bit data pointer DPTR to access indirect addressed RAM or Program Memory. DPS (AUXR0.0) bit decides which data pointer, DPTR or DPTR1, is activated.

RCTRIM0 – High Speed Internal Oscillator Trim 0

Register	SFR Address	Reset Value
RCTRIM0	84H, Page 0, TA protected	XXXX _ XXXXb

7	6	5	4	3	2	1	0
HIRCTRIM[8:1]							
R/W							

LRCTRIM – Low Speed Internal Oscillator Trim

Register	SFR Address	Reset Value
LRCTRIM	84H, Page 1, TA protected	XXXX _ XXXXb

7	6	5	4	3	2	1	0
LIRCTRIM[7:0]							
R/W							

RCTRIM1 – High Speed Internal Oscillator Trim 1

Register	SFR Address	Reset Value
RCTRIM1	85H, Page 0, TA protected	XXXX _ XXXXb

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	HIRCTRIM.0
-	-	-	-	-	-	-	R/W

XLTCON – XLT Clock Control

Register	SFR Address	Reset Value
XLTCON	85H, Page 1, TA protected	0111 _ 0111b

7	6	5	4	3	2	1	0
HSCH	HXSG			-	-	LXSG	
R/W	R/W			-	-	R/W	

Bit	Name	Description
7	HSCH	HXT Schmitt trigger select 0 = disable 1 = enable
6:4	HXSG	HXT gain value select 000 = L0 mode (smallest value) 001 = L1 mode 010 = L2 mode 011 = L3 mode 100 = L4 mode 101 = L5 mode 110 = L6 mode 111 = L7 mode (largest value)
3:2	-	Reserved
1:0	LXSG	LXT gain value select 00 = L0 mode (smallest value) 01 = L1 mode 10 = L2 mode 11 = L3 mode (largest value)

RWK – Self Wake-up Timer Reload Byte

Register	SFR Address	Reset Value
RWK	86H, Page 0	0000 _ 0000b

7	6	5	4	3	2	1	0
RWK[7:0]							
R/W							

Bit	Name	Description
7:0	RWK[7:0]	**WKT reload byte** It holds the 8-bit reload value of WKT. Note that RWK should not be FFH if the pre-scale is 1/1 for implement limitation.

CWK – Self Wake-up Timer current count value

Register	SFR Address	Reset Value
CWK	86H, Page 1	0000 _ 0000b

7	6	5	4	3	2	1	0
CWK[7:0]							
R/W							

Bit	Name	Description
7:0	CWK[7:0]	**WKT current count value** It is store value of WKT current count.

PCON – Power Control

Register	SFR Address	Reset Value
PCON	87H, All pagess	POR: 0001_000b, other: 000U_0000b

7	6	5	4	3	2	1	0
SMOD	SMOD0	LPR	POF	GF1	GF0	PD	IDL
R/W	R/W	RW	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	SMOD	**Serial port 0 double baud rate enable** Setting this bit doubles the serial port baud rate when UART0 is in Mode 2 or when Timer 1 overflow is used as the baud rate source of UART0 Mode 1 or 3. See Table 15.1-1 Serial Port 0 Mode / baudrate Description for details.

Bit	Name	Description
6	SMOD0	**Serial port 0 framing error flag access enable** 0 = SCON.7 accesses to SM0 bit. 1 = SCON.7 accesses to FE bit.
5	LPR	**Low power run mode** 0 = disable 1 = enable Note**:** If PD = 1 and LPR = 1 at the same time, LPR is invalid, CPU will enter power down mode.
4	POF	**Power-on reset flag** This bit will be set as 1 after a power-on reset. It indicates a cold reset, a power-on reset complete. This bit remains its value after any other resets. This flag is recommended to be cleared via software.
3	GF1	**General purpose flag 1** The general purpose flag that can be set or cleared by user via software.
2	GF0	**General purpose flag 0** The general purpose flag that can be set or cleared by user via software.
1	PD	**Power-down mode** Setting this bit puts CPU into Power-down mode. Under this mode, both CPU and peripheral clocks stop and Program Counter (PC) suspends. It provides the lowest power consumption. After CPU is woken up from Power-down, this bit will be automatically cleared via hardware and the program continue executing the interrupt service routine (ISR) of the very interrupt source that woke the system up before. After return from the ISR, the device continues execution at the instruction, which follows the instruction that put the system into Power-down mode. Note that If IDL bit and PD bit are set simultaneously, CPU will enter Power-down mode. Then it does not go to Idle mode after exiting Power-down.
0	IDL	**Idle mode** Setting this bit puts CPU into Idle mode. Under this mode, Program Counter (PC) suspends but the CPU clock keep running and all peripherals keep activated. After CPU is woken up from Idle, this bit will be automatically cleared via hardware and the program continue executing the ISR of the very interrupt source that woke the system up before. After return from the ISR, the device continues execution at the instruction which follows the instruction that put the system into Idle mode.

TCON – Timer 0 and 1 Control

Register	SFR Address	Reset Value
TCON	88H, All pages, Bit-addressable	0000_0000b

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R (level) R/W (edge)	R/W	R (level) R/W (edge)	R/W

Bit	Name	Description

Bit	Name	Description
7	TF1	**Timer 1 overflow flag** This bit is set when Timer 1 overflows. It is automatically cleared by hardware when the program executes the Timer 1 interrupt service routine. This bit can be set or cleared by software.
6	TR1	**Timer 1 run control** 0 = Timer 1 Disabled. Clearing this bit will halt Timer 1 and the current count will be preserved in TH1 and TL1. 1 = Timer 1 Enabled.
5	TF0	**Timer 0 overflow flag** This bit is set when Timer 0 overflows. It is automatically cleared via hardware when the program executes the Timer 0 interrupt service routine. This bit can be set or cleared by software.
4	TR0	**Timer 0 run control** 0 = Timer 0 Disabled. Clearing this bit will halt Timer 0 and the current count will be preserved in TH0 and TL0. 1 = Timer 0 Enabled.
3	IE1	**External interrupt 1 edge flag** If IT1 = 1 (falling edge trigger), this flag will be set by hardware when a falling edge is detected. It remain set until cleared via software or cleared by hardware in the beginning of its interrupt service routine. If IT1 = 0 (low level trigger), this flag follows the inverse of the $\overline{INT1}$ input signal's logic level. Software cannot control it.
2	IT1	**External interrupt 1 type select** This bit selects by which type that $\overline{INT1}$ is triggered. 0 = $\overline{INT1}$ is low level triggered. 1 = $\overline{INT1}$ is falling edge triggered.
1	IE0	**External interrupt 0 edge flag** If IT0 = 1 (falling edge trigger), this flag will be set by hardware when a falling edge is detected. It remain set until cleared via software or cleared by hardware in the beginning of its interrupt service routine. If IT0 = 0 (low level trigger), this flag follows the inverse of the $\overline{INT0}$ input signal's logic level. Software cannot control it.
0	IT0	**External interrupt 0 type select** This bit selects by which type that $\overline{INT0}$ is triggered. 0 = $\overline{INT0}$ is low level triggered. 1 = $\overline{INT0}$ is falling edge triggered.

TMOD – Timer 0 and 1 Mode

Register	SFR Address	Reset Value
TMOD	89H All pages	0000_0000b

7	6	5	4	3	2	1	0
GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	GATE	**Timer 1 gate control** 0 = Timer 1 will clock when TR1 is 1 regardless of $\overline{INT1}$ logic level. 1 = Timer 1 will clock only when TR1 is 1 and $\overline{INT1}$ is logic 1.
6	C/$\overline{T}$	**Timer 1 Counter/Timer select** 0 = Timer 1 is incremented by internal system clock. 1 = Timer 1 is incremented by the falling edge of the external pin T1.
5	M1	**Timer 1 mode select** M1 M0 Timer 1 Mode 0 0 Mode 0: 13-bit Timer/Counter 0 1 Mode 1: 16-bit Timer/Counter 1 0 Mode 2: 8-bit Timer/Counter with auto-reload from TH1 1 1 Mode 3: Timer 1 halted
4	M0	
3	GATE	**Timer 0 gate control** 0 = Timer 0 will clock when TR0 is 1 regardless of $\overline{INT0}$ logic level. 1 = Timer 0 will clock only when TR0 is 1 and $\overline{INT0}$ is logic 1.
2	C/$\overline{T}$	**Timer 0 Counter/Timer select** 0 = Timer 0 is incremented by internal system clock. 1 = Timer 0 is incremented by the falling edge of the external pin T0.
1	M1	**Timer 0 mode select** M1 M0 Timer 0 Mode 0 0 Mode 0: 13-bit Timer/Counter 0 1 Mode 1: 16-bit Timer/Counter 1 0 Mode 2: 8-bit Timer/Counter with auto-reload from TH0 1 1 Mode 3: TL0 as a 8-bit Timer/Counter and TH0 as a 8-bit Timer
0	M0	

TL0 – Timer 0 Low Byte

Register	SFR Address	Reset Value
TL0	8AH All pages	0000_0000b

7	6	5	4	3	2	1	0
TL0[7:0]							
R/W							

Bit	Name	Description
7:0	TL0[7:0]	**Timer 0 low byte** The TL0 register is the low byte of the 16-bit counting register of Timer 0.

PnDW – Port n Pull-Down resister control

Register	SFR Address	Reset Value
P0DW	8AH, Page 1	0000_0000 b
P1DW	8BH, Page 1	0000_0000 b
P2DW	8CH, Page 1	0000_0000 b
P3DW	8DH, Page 1	0000_0000 b
P4DW	8EH, Page 1	0000_0000 b
P5DW	8FH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnDW.7	PnDW.6	PnDW.5	PnDW.4	PnDW.3	PnDW.2	PnDW.1	PnDW.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnDW[7:0]	**P0.n pull-down enable** 0 = Pn.x pull-down Disabled. 1 = Pn.x pull-down Enabled.

TL1 – Timer 1 Low Byte

Register	SFR Address	Reset Value
TL1	8BH, Page 0	0000_0000b

7	6	5	4	3	2	1	0
TL1[7:0]							
R/W							

Bit	Name	Description
7:0	TL1[7:0]	**Timer 1 low byte** The TL1 register is the low byte of the 16-bit counting register of Timer 1.

TH0 – Timer 0 High Byte

Register	SFR Address	Reset Value
TH0	8CH, Page 0	0000_0000b

7	6	5	4	3	2	1	0
TH0[7:0]							
R/W							

Bit	Name	Description
7:0	TH0[7:0]	**Timer 0 high byte** The TH0 register is the high byte of the 16-bit counting register of Timer 0.

TH1 – Timer 1 High Byte

Register	SFR Address	Reset Value
TH1	8DH, Page 0	0000_0000b

7	6	5	4	3	2	1	0
TH1[7:0]							
R/W							

Bit	Name	Description
7:0	TH1[7:0]	**Timer 1 high byte** The TH1 register is the high byte of the 16-bit counting register of Timer 1.

CKCON – Clock Control

Register	SFR Address	Reset Value
CKCON	8EH, Page 0	1000_0000b

7	6	5	4	3	2	1	0
FASTWK	PWMCKS	T1OE	T1M	T0M	T0OE	CLOEN	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Bit	Name	Description
7	FASTWK	**Fast Wakeup enable** 0 = Faster Wakeup Disabled, when system wakeup from power down mode, HIRC clock stable time is about 10us. 1 = Faster Wakeup Enabled, when system wakeup from power down mode, HIRC clock stable time is about 3us.
6	PWMCKS	**PWM clock source select** 0 = The clock source of PWM is the system clock FSYS. 1 = The clock source of PWM is the overflow of Timer 1.
5	T1OE	**Timer 1 output enable** 0 = Timer 1 output Disabled. 1 = Timer 1 output Enabled from T1 pin. Note that Timer 1 output should be enabled only when operating in its “Timer” mode.

Bit	Name	Description
4	T1M	**Timer 1 clock mode select** 0 = The clock source of Timer 1 is the system clock divided by 12. It maintains standard 8051 compatibility. 1 = The clock source of Timer 1 is direct the system clock.
3	T0M	**Timer 0 clock mode select** 0 = The clock source of Timer 0 is the system clock divided by 12. It maintains standard 8051 compatibility. 1 = The clock source of Timer 0 is direct the system clock.
2	T0OE	**Timer 0 output enable** 0 = Timer 0 output Disabled. 1 = Timer 0 output Enabled from T0 pin. Note that Timer 0 output should be enabled only when operating in its “Timer” mode.
1	CLOEN	**System clock output enable** 0 = System clock output Disabled. 1 = System clock output Enabled from CLO pin.
0	-	Reserved

WKCON – Self Wake-up Timer Control

Register	SFR Address	Reset Value
WKCON	8FH, PAGE 0	0000_0000b

7	6	5	4	3	2	1	0
-	-	WKTCK	WKTF	WKTR	WKPS[2:0]		
-	-	R/W	R/W	R/W	R/W		

Bit	Name	Description
7:6	-	Reserved
	WKTCK	**WKT clock source** This bit is set WKT clock source select bit. 0 = LIRC 1 = LXT
4	WKTF	**WKT overflow flag** This bit is set when WKT overflows. If the WKT interrupt and the global interrupt are enabled, setting this bit will make CPU execute WKT interrupt service routine. This bit is not automatically cleared via hardware and should be cleared via software.

Bit	Name	Description
3	WKTR	**WKT run control** 0 = WKT is halted. 1 = WKT starts running. Note that the reload register RWK can only be written when WKT is halted (WKTR bit is 0). If WKT is written while WKTR is 1, result is unpredictable.
2:0	WKPS[2:0]	**WKT pre-scalar** These bits determine the pre-scale of WKT clock. 000 = 1/1. 001 = 1/4. 010 = 1/16. 011 = 1/64. 100 = 1/256. 101 = 1/512. 110 = 1/1024. 111 = 1/2048.

SFRS – SFR Page Selection

Register	SFR Address	Reset Value
SFRS	91H, All pages	0000_0000b

7	6	5	4	3	2	1	0
-	-	-	-	-	-	SFRPAGE	SFRPAGE
-	-	-	-	-	-	R/W	R/W

Bit	Name	Description
1:0	SFRPAGE	**SFR page select** 00 = Instructions access SFR Page 0. 01 = Instructions access SFR Page 1. 10 = Instructions access SFR page 2. 11 = Reserved

DMAnCR – PDMAn Control Register

Register	SFR Address	Reset Value
DMA0CR	92H, Page 0	0000_0000 b
DMA1CR	EBH, Page 0	0000_0000 b
DMA2CR	B3H, Page 2	0000_0000 b
DMA3CR	ABH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	PSSEL[2:0]			HIE	FIE	RUN	EN
-	R/W			R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6:4	PSSEL[2:0]	**Peripheral Source Select** 000 = XRAM to XRAM 001 = SPI0 RX 010 = SMC/UART RX. 011 = SPI1 RX 100 = Reserved, No peripheral source select 101 = SPI0 TX 110 = SMC/UART TX. 111 = SPI1 TX **Note:** 001~011 : peripheral devices to XRAM memory 101~111 : XRAM memory to peripheral devices
3	HIE	**PDMA HALFTransfer Done Interrupt Enable Bit** 0 = Interrupt Disabled when PDMA half transfer is done. 1 = Interrupt Enabled when PDMA half transfer is done.
2	FIE	**PDMA Full Transfer Done Interrupt Enable Bit** 0 = Interrupt Disabled when PDMA full transfer is done. 1 = Interrupt Enabled when PDMA full transfer is done.
1	RUN	**Trigger Enable Bit** 0 = No effect. 1 = PDMA data transfer Enabled. **Note1:** When PDMA transfer completed, this bit will be cleared automatically.
0	EN	**PDMA Enable Bit** Setting this bit to 1 enables PDMA operation. If this bit is cleared, PDMA will ignore all PDMA request and Reset the internal state machine, pointers and internal buffer. The contents of all control registers will not be cleared.

PnUP – Port n Pull-Up resister control

Register	SFR Address	Reset Value
P0UP	92H, Page 1	0000_0000 b
P1UP	93H, Page 1	0000_0000 b
P2UP	94H, Page 1	0000_0000 b
P3UP	95H, Page 1	0000_0000 b
P4UP	96H, Page 1	0000_0000 b
P5UP	97H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnUP.7	PnUP.6	PnUP.5	PnUP.4	PnUP.3	PnUP.2	PnUP.1	PnUP.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnUP[7:0]	**P0.n pull-up enable** 0 = Pn.x pull-up Disabled. 1 = Pn.x pull-up Enabled.

DMAnMA – PDMA XRAM Base Address Low Byte

Register	SFR Address	Reset Value
DMA0MAL	93H, Page 0	0000_0000 b
DMA1MAL	ECH, Page 0	0000_0000 b
DMA2MAL	B4H, Page 2	0000_0000 b
DMA3MAL	ACH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
MAL[7:0]							
R/W							

Bit	Name	Description
7:0	MAL[7:0]	**PDMA XRAM Base Address (Low Byte)** The least significant 8 bits of XRAM address to store or read for the peripheral source data; in memory to memory transfer, this register is the source address. XRAM address = {MAH[3:0],MAL[7:0]}

DMAnCNT – PDMA Transfer Count

Register	SFR Address	Reset Value
DMA0CNT	94H, Page 0	0000_0000 b
DMA1CNT	EDH, Page 0	0000_0000 b
DMA2CNT	B5H, Page 2	0000_0000 b
DMA3CNT	ADH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
CNT[7:0]							
R/W							

Bit	Name	Description

Bit	Name	Description
7:0	CNT[7:0]	**PDMA Transfer Count** The total transfer count for PDMA request operation. Total transfer count = CNT[7:0] + 1

DMAnCCNT – PDMA Current Transfer Count

Register	SFR Address	Reset Value
DMA0CCNT	95H, Page 0	0000_0000 b
DMA1CCNT	EEH, Page 0	0000_0000 b
DMA2CCNT	B6H, Page 2	0000_0000 b
DMA3CCNT	AEH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
DMAnCCNT[7:0]							
R/W							

Bit	Name	Description
7:0	CNT[7:0]	**PDMA Current Transfer Count** The current transfer count for PDMA request operation. Current transfer count = CCNT[7:0] **Note:** while DMAnCNT=0xFF (total transfer count = 256) and DMAnCCNT = 0x00 , If PDMA FDONE flag (DMAnTSR[0])=0, that means, 1'st byte data is not complete.If PDMA FDONE flag (DMAnTSR[0])=1, that means, all of data are transferred..

CKSWT – Clock Switch

Register	SFR Address	Reset Value
CKSWT	96H, PAGE 0, TA protected	0011 _0000 b

7	6	5	4	3	2	1	0
HXTST	LXTST	HIRCST	LIRCST	ECLKST	OSC[2:0]		
R	R	R	R	R	W		

Bit	Name	Description
7	HXTST	**High speed external crystal/resonator 4 MHz to 24 MHz status** 0 = High speed external crystal/resonator is not stable or is disabled. 1 = High speed external crystal/resonator is enabled and stable.
6	LXTST	**Low speed external crystal/resonator 32.768 kHz status** 0 = Low speed external crystal/resonator is not stable or is disabled. 1 = Low speed external crystal/resonator is enabled and stable.

Bit	Name	Description
5	HIRCST	**High-speed internal oscillator 24 MHz status** 0 = High-speed internal oscillator is not stable or disabled. 1 = High-speed internal oscillator is enabled and stable.
4	LIRCST	**Low speed internal oscillator 38.4 kHz status** 0 = Low speed internal oscillator is not stable or is disabled. 1 = Low speed internal oscillator is enabled and stable.
3	ECLKST	**External clock input status** 0 = External clock input is not stable or disabled. 1 = External clock input is enabled and stable.
2:0	OSC[2:0]	**Oscillator selection bits** This field selects the system clock source. 00x = Internal 24 MHz oscillator. Defaul value accoding to HIRCEN(CKEN.5) enabled. 01x = External oscillator clock source accoding to ECLKEN(CKEN.3) enabled. 10x = Internal 38.4 kHz oscillator according to LIRCEN(CKEN.4) enabled. 110 = External High speed crystal/resonator clock source (4 MHz ~ 24 MHz) accoding to EHXTEN(CKEN.7) enabled. 111 = External Low speed crystal/resonator clock source (32.768 kHz) accoding to ELXTEN(CKEN.6) enabled. Note that this field is write only. The read back value of this field may not correspond to the present system clock source.

CKEN – Clock Enable

Register	SFR Address	Reset Value
CKEN	97H, PAGE 0, TA protected	0011 _0100 b

7	6	5	4	3	2	1	0
EHXTEN	ELXTEN	HIRCEN	LIRCEN	ECLKEN	-	-	CKSWTF
R/W	R/W	R/W	R/W	R/W	-	-	R

Bit	Name	Description
7	EHXTEN	**External High-speed crystal/resonator enable** 1 = High-speed external crystal/resonator 4 MHz to 24 MHz Enabled. 0 = High-speed external crystal/resonator 4 MHz to 24 MHz Disabled, P5.2 and P5.3 work as general purpose I/O or other functions if ECLKEN set to 0.
6	ELXTEN	**External Low-speed crystal/resonator enable** 1 = Low-speed external crystal/resonator 32.768 kHz Enabled. 0 = Low-speed external crystal/resonator 32.768 kHz Disabled, P5.4 and P5.5 work as general purpose I/O or other functions.

Bit	Name	Description
5	HIRCEN	**High-speed internal oscillator 24 MHz enable** 0 = The high-speed internal oscillator Disabled. 1 = The high-speed internal oscillator Enabled. Note that once IAP is enabled by setting IAPEN (CHPCON.0), the high-speed internal 24 MHz oscillator will be enabled automatically. The hardware will also set HIRCEN and HIRCST bits. After IAPEN is cleared, HIRCEN and EHRCST resume the original values.
4	LIRCEN	**Low speed internal oscillator 38.4 kHz enable** 0 = The low speed internal oscillator Disabled. 1 = The low speed internal oscillator Enabled. Note that when (1)WDT is enabled, (2)WKT is running by the clock source of the internal 38.4 kHz oscillator ,(3) BOD is enabled, or (4)LVR filter is enabled, a write 0 to LIRCEN will be ignored. LIRCEN is always 1 and the internal 38.4 kHz oscillator is always enabled.
3	ECLKEN	**External Clock Input enable** 1 = External clock input (XIN , P5.3) Enabled. 0 = External clock input (XIN, P5.3) Disabled, P5.2 and P5.3 work as general purpose I/O or other functions if EHXTEN set to 0.
1	-	Reserved
0	CKSWTF	**Clock switch fault flag** 0 = The previous system clock source switch was successful. 1 = User tried to switch to an instable or disabled clock source at the previous system clock source switch. If switching to an instable clock source, this bit remains 1 until the clock source is stable and switching is successful.

SCON – Serial Port Control

Register	SFR Address	Reset Value
SCON	98H, All pages, Bit-addressable	0000 _0000 b

7	6	5	4	3	2	1	0
SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	SM0/FE	**Serial port mode select** SMOD0 (PCON.6) = 0: See Table 15.1-1 Serial Port 0 Mode / baudrate Description for details. SMOD0 (PCON.6) = 1: SM0/FE bit is used as frame error (FE) status flag. It is cleared by software. 0 = Frame error (FE) did not occur. 1 = Frame error (FE) occurred and detected.
6	SM1	

Bit	Name	Description
5	SM2	**Multiprocessor communication mode enable** The function of this bit is dependent on the serial port 0 mode. Mode 0: This bit select the baud rate between FSYS/12 and FSYS/2. 0 = The clock runs at FSYS/12 baud rate. It maintains standard 8051compatibility. 1 = The clock runs at FSYS/2 baud rate for faster serial communication. Mode 1: This bit checks valid stop bit. 0 = Reception is always valid no matter the logic level of stop bit. 1 = Reception is valid only when the received stop bit is logic 1 and the received data matches “Given” or “Broadcast” address. Mode 2 or 3: For multiprocessor communication. 0 = Reception is always valid no matter the logic level of the 9th bit. 1 = Reception is valid only when the received 9th bit is logic 1 and the received data matches “Given” or “Broadcast” address.
4	REN	**Receiving enable** 0 = Serial port 0 reception Disabled. 1 = Serial port 0 reception Enabled in Mode 1,2, or 3. In Mode 0, reception is initiated by the condition REN = 1 and RI = 0.
3	TB8	**9th transmitted bit** This bit defines the state of the 9th transmission bit in serial port 0 Mode 2 or 3. It is not used in Mode 0 or 1.
2	RB8	**9th received bit** The bit identifies the logic level of the 9th received bit in serial port 0 Mode 2 or 3. In Mode 1, RB8 is the logic level of the received stop bit. SM2 bit as logic 1 has restriction for exception. RB8 is not used in Mode 0.
1	TI	**Transmission interrupt flag** This flag is set by hardware when a data frame has been transmitted by the serial port 0 after the 8th bit in Mode 0 or the last data bit in other modes. When the serial port 0 interrupt is enabled, setting this bit causes the CPU to execute the serial port 0 interrupt service routine. This bit should be cleared manually via software.
0	RI	**Receiving interrupt flag** This flag is set via hardware when a data frame has been received by the serial port 0 after the 8th bit in Mode 0 or after sampling the stop bit in Mode 1, 2, or 3. SM2 bit as logic 1 has restriction for exception. When the serial port 0 interrupt is enabled, setting this bit causes the CPU to execute to the serial port 0 interrupt service routine. This bit should be cleared manually via software.

SBUF – Serial Port Data Buffer

Register	SFR Address	Reset Value
SBUF	99H, Page 0	0000 _0000 b
SBUF1	9AH, Page 0	0000 _0000 b

7	6	5	4	3	2	1	0
SBUFn[7:0]							
R/W							

Bit	Name	Description
7:0	SBUF[7:0]	**Serial port data buffer** This byte actually consists two separate registers. One is the receiving resister, and the other is the transmitting buffer. When data is moved to SBUF, it goes to the transmitting buffer and is shifted for serial transmission. When data is moved from SBUF, it comes from the receiving register. The transmission is initiated through giving data to SBUF.

PnS – Port n Schmitt Triggered Input

Register	SFR Address	Reset Value
P0S	99H, Page 1	0000_0000 b
P1S	9BH, Page 1	0000_0000 b
P2S	9DH, Page 1	0000_0000 b
P3S	ACH, Page 1	0000_0000 b
P4S	BBH, Page 1	0000_0000 b
P5S	BFH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnS.7	PnS.6	PnS.5	PnS.4	PnS.3	PnS.2	PnS.1	PnS.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnS[7:0]	**P0 Schmitt triggered input** 0 = TTL level input of Pn.x. 1 = Schmitt triggered input of Pn.x.

PnSR –Port n Slew Rate Control

Register	SFR Address	Reset Value
P0SR	9AH, Page 1	0000_0000 b
P1SR	9CH, Page 1	0000_0000 b
P2SR	9EH, Page 1	0000_0000 b
P3SR	ADH, Page 1	0000_0000 b
P4SR	BCH, Page 1	0000_0000 b
P5SR	AEH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnSR.7	PnSR.6	PnSR.5	PnSR.4	PnSR.3	PnSR.2	PnSR.1	PnSR.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnSR[7:0]	**P0.n slew rate** 0 = Pn.x normal output slew rate. 1 = Pn.x high-speed output slew rate.

EIE0 – Extensive Interrupt Enable

Register	SFR Address	Reset Value
EIE0	9BH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ET2	ESPI0	EFB0	EWDT	EPWM0	ECAP	EPI	EI^{2}C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	ET2	**Enable Timer 2 interrupt** 0 = Timer 2 interrupt Disabled. 1 = Timer 2 interrupt Enable. When interrupt generated, TF2 (T2CON.7) set 1
6	ESPI0	**Enable SPI interrupt** 0 = SPI interrupt Disabled. 1 = SPI interrupt Enable. When interrupt generated SPIF (SPInSR.7), SPIOVF (SPInSR.5), or MODF (SPInSR.4) set 1 .
5	EFB0	**Enable Fault Brake interrupt** 0 = Fault Brake interrupt Disabled. 1 = Fault Brake interrupt Enable. When interrupt generated FBF (PWM0FBD.7) set 1.

Bit	Name	Description
4	EWDT	**Enable WDT interrupt** 0 = WDT interrupt Disabled. 1 = WDT interrupt Enable. When interrupt generated WDTF (WDCON.5) set 1.
3	EPWM0	**Enable PWM0 interrupt** 0 = PWM interrupt Disabled. 1 = PWM interrupt Enable. When interrupt generated PWMF (PWMnCON0.5) set 1.
2	ECAP	**Enable input capture interrupt** 0 = Input capture interrupt Disabled. 1 = Input capture interrupt Enable. When interrupt generated CAPF[2:0] (CAPCON0[2:0]) set 1.
1	EPI	**Enable pin interrupt** 0 = Pin interrupt Disabled. 1 = Pin interrupt Enable. When interrupt generated PIF related bit set 1.
0	EI^{2}C0	**Enable I^{2}C0 interrupt** 0 = I^{2}C interrupt Disabled. 1 = I^{2}C interrupt Enable. When interrupt generated SI (I^{2}C0CON.3) or I2TOF (I^{2}C0TOC.0) set 1.

EIE1 – Extensive Interrupt Enable 1

Register	SFR Address	Reset Value
EIE1	9CH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
EFB1	EPWM1	EI^{2}C1	ESPI1	EHFI	EWKT	ET3	ES1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	EFB1	**Enable Fault Brake 1 interrupt** 0 = Fault Brake interrupt Disabled. 1 = Fault Brake interrupt Enable. When interrupt generated FBF (PWM1FBD.7) Enabled.
6	EPWM1	**Enable PWM1 interrupt** 0 = PWM1 interrupt Disabled. 1 = PWM1 interrupt Enable. When interrupt generated PWMF (PWM1CON0.5) set 1.
5	EI^{2}C1	**Enable I^{2}C1 interrupt** 0 = I^{2}C1 interrupt Disabled. 1 = I^{2}C1 interrupt Enable. When interrupt generated SI (I^{2}C1CON.3) or I2TOF (I^{2}C1TOC.0) set 1.

Bit	Name	Description
4	ESPI1	**Enable SPI1 interrupt** 0 = SPI1 interrupt Disabled. 1 = SPI1 interrupt Enable. When interrupt generated SPIF (SP2SR.7), MODF (SP2SR.4) or SPIOVF (SP2SR.5) set 1
3	EHFI	**Enable hard fault interrupt** 0 = hard fault interrupt Disabled and hard fault reset is Enabled 1 = hard fault interrupt Enable. When interrupt generated HFIF (AUXR0.4) set 1.
2	EWKT	**Enable WKT interrupt** 0 = WKT interrupt Disabled. 1 = WKT interrupt Enable. When interrupt generated WKTF (WKCON.4) set 1.
1	ET3	**Enable Timer 3 interrupt** 0 = Timer 3 interrupt Disabled. 1 = Timer 3interrupt Enable. When interrupt generated TF3 (T3CON.4) set 1.
0	ES1	**Enable serial port 1 interrupt** 0 = Serial port 1 interrupt Disabled. 1 = Serial port 1 interrupt Enable. When interrupt generated TI_1 (S1CON.1) or RI_1 (S1CON.0) set 1.

RSR – Reset Flag Register

Register	SFR Address	Reset Value
RSR	9DH, Page 0	1101_0000 b

7	6	5	4	3	2	1	0
LVRF	PORF	HFRF	POF	RSTPINF	BOR	WDTRF	SWRF
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	LVRF	**LVR Reset Flag** 1: LVR Reset Flag is active 0: LVR Reset Flag is inactive Write 0 to clear this bit
6	PORF	**POR Reset Flag** 1: POR15 Reset Flag is active 0: POR15 Reset Flag is inactive Write 0 to clear this bit
5	HFRF	**mirrored from AUXR0.5** Clear this bit by write AUXR0.5=0 or RSR.5=0

Bit	Name	Description
4	POF	**mirrored from PCON.4** Clear this bit by write PCON.4=0 or RSR.4=0
3	RSTPINF	**mirrored from AUXR0.6** Clear this bit by write AUXR0.6=0 or RSR.3=0
2	BORF	**mirrored from BODCON0.1** Clear this bit by write BODCON0.1=0 or RSR.2=0
1	WDTRF	**mirrored from WDCON.3** Clear this bit by write WDCON.3=0 or RSR.1=0
0	SWRF	**mirrored from AUXR0.7** Clear this bit by write AUXR0.7=0 or RSR.0=0

CHPCON – Chip Control

Register	SFR Address	Reset Value
CHPCON	9FH, All pages, TA protected	Software: 0000_00U0 b others: 0000_00C0 b

7	6	5	4	3	2	1	0
SWRST	IAPFF	-	-	-	-	BS	IAPEN
W	R/W	-	-	-	-	R/W	R/W

Bit	Name	Description
7	SWRST	Software reset To set this bit as logic 1 will cause a software reset. It will automatically be cleared via hardware after reset is finished.
6	IAPFF	IAP fault flag The hardware will set this bit after IAPGO (IAPTRG.0) is set if any of the following condition is met: (1) The accessing address is oversize. (2) IAPCN command is invalid. (3) IAP erases or programs updating un-enabled block. (4) IAP erasing or programming operates under VBOD while BOIAP (CONFIG2.5) remains un-programmed 1 with BODEN (BODCON0.7) as 1 and BORST (BODCON0.2) as 0. This bit should be cleared via software.
5:2	-	Reserved
1	BS	**Boot select** This bit defines from which block that MCU re-boots after all resets. 0 = MCU will re-boot from APROM after all resets. 1 = MCU will re-boot from LDROM after all resets.

Bit	Name	Description
0	IAPEN	**IAP enable** 0 = IAP function Disabled. 1 = IAP function Enabled. Once enabling IAP function, the HIRC will be turned on for timing control. To clear IAPEN should always be the last instruction after IAP operation to stop internal oscillator if reducing power consumption is concerned.

ADCCON0 – ADC Control register 0

Register	SFR Address	Reset Value
ADCCON0	A1H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCF	ADCS	ETGSEL1	ETGSEL0	ADCHS3	ADCHS2	ADCHS1	ADCHS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	ADCF	**ADC flag** This flag is set when an A/D conversion is completed in single sampling mode, final sampling complete in continue sampling mode or comparing hit if result comparator is enabled. The ADC result can be read. While this flag is 1, ADC cannot start a new converting. This bit is cleared by software.
6	ADCS	**A/D converting software start trigger** Setting this bit 1 triggers an A/D conversion. This bit remains logic 1 during A/D converting time and is automatically cleared via hardware right after conversion complete. The meaning of writing and reading ADCS bit is different. Writing: 0 = No effect. 1 = Start an A/D converting. Reading: 0 = ADC is in idle state. 1 = ADC is busy in converting.
5:4	ETGSEL[1:0]	**External trigger source select** When ADCEX (ADCCON1.1) is set, these bits select which pin output triggers ADC conversion. 00 = PWM0CH0. 01 = PWM0CH2. 10 = PWM0CH4. 11 = STADC pin.

Bit	Name	Description
3:0	ADCHS[3:0]	**A/D converting channel select** This filed selects the activating analog input source of ADC. If ADCEN is 0, all inputs are disconnected. 0000 = ADC_CH0. 0001 = ADC_CH1. 0010 = ADC_CH2. 0011 = ADC_CH3. 0100 = ADC_CH4. 0101 = ADC_CH5. 0110 = ADC_CH6. 0111 = ADC_CH7. 1000 = VBG (Internal band-gap voltage 1.22V). 1001 = VTEMP. (Temperature Sensor). Others = Reserved.

PIPSn – Pin Interrupt Control

Register	SFR Address	Reset Value
PIPS0	A1H, Page 1	0000_0000 b
PIPS1	A2H, Page 1	0000_0000 b
PIPS2	A3H, Page 1	0000_0000 b
PIPS3	A4H, Page 1	0000_0000 b
PIPS4	A5H, Page 1	0000_0000 b
PIPS5	A6H, Page 1	0000_0000 b
PIPS6	A7H, Page 1	0000_0000 b
PIPS7	AFH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	PSEL[2:0]			-	BSEL[2:0]		
-	R/W			-	R/W		

Bit	Name	Description
7	-	Reserved

Bit	Name	Description
6:4	PSEL[2:0]	**Pin interrupt channel Port select** 000 = P0 Port. 001 = P1 Port. 010 = P2 Port. 011 = P3 Port. 100 = P4 Port. 101 = P5 Port. 110 = Reserved. 111 = Reserved.
3	-	Reserved
2:0	BSEL[2:0]	**Pin interrupt channel bit select** 000 = Pn.0. 001 = Pn.1 010 =.Pn.2 011 = Pn.3. 100 = Pn.4. 101 = Pn.5. 110 = Pn.6. 111 = Pn.7. n is the PORT number, which is selected by PSEL[2:0].

AUXR0 – Auxiliary Register 0

Register	SFR Address	Reset Value
AUXR0	A2H, Page:0	POR: 0000_0000b Software: 1UU0_0000b Reset pin: U1U0_0000b Hard fault: UU10_0000b Others: UUU0_0000b

7	6	5	4	3	2	1	0
SWRF	RSTPINF	HFRF	HFIF	GF2	-	0	DPS
R/W	R/W	R/W	R/W	R/W	-	R	R/W

Bit	Name	Description
7	SWRF	**Software reset flag** When the MCU is reset via software reset, this bit will be set via hardware. It is recommended that the flag be cleared via software.
6	RSTPINF	**External reset flag** When the MCU is reset by the external reset, this bit will be set via hardware. It is recommended that the flag be cleared via software.

Bit	Name	Description
5	HFRF	**Hard Fault reset flag** Once CPU fetches instruction address over Flash size while EHFI (EIE1.4)=0, MCU will reset and this bit will be set via hardware. It is recommended that the flag be cleared via software. **Note:** If MCU run in OCD debug mode and OCDEN = 0, Hard fault reset will disable. Only HFRF flag be asserted.
4	HFIF	**Hard Fault Interrupt flag** Once CPU fetches instruction address over Flash size while EHFI (EIE1.4)=1. MCU will be interrupt and this bit will be set via hardware. It is recommended that the flag be cleared via software.
3	GF2	**General purpose flag 2** The general purpose flag that can be set or cleared by the user via software.
1	-	Reserved
1	0	Reserved This bit is always read as 0.
0	DPS	**Data pointer select** 0 = Data pointer 0 (DPTR) is active by default. 1 = Data pointer 1 (DPTR1) is active. After DPS switches the activated data pointer, the previous inactivated data pointer remains its original value unchanged.

BODCON0 – Brown-out Detection Control 0

Register	SFR Address	Reset Value
BODCON0	A3H, Page 0, TA protected	POR: CCCC_XC0Xb BOD: UUUU_XU1Xb Others: UUUU_XUUXb

7	6	5	4	3	2	1	0
BODEN	BOV[2:0]			BOF	BORST	BORF	BOS
R/W	R/W			R/W	R/W	R/W	R

Bit	Name	Description
7	BODEN	**Brown-out detection enable** 0 = Brown-out detection circuit off. 1 = Brown-out detection circuit on. Note that BOD output is not available until 2~3 LIRC clocks after enabling.

Bit	Name	Description
6:4	BOV[2:0]	**CONFIG brown-out voltage select** 111 = VBOD is 1.8V. 110 = VBOD is 1.8V. 101 = VBOD is 2.0V. 100 = VBOD is 2.4V. 011 = VBOD is 2.7V. 010 = VBOD is 3.0V. 001 = VBOD is 3.7V. 000 = VBOD is 4.4V.
3	BOF	**Brown-out interrupt flag** This flag will be set as logic 1 via hardware after a V_{DD} dropping below or rising above VBOD event occurs. If both EBOD (IE.5) and EA (IE.7) are set, a brown-out interrupt requirement will be generated. This bit should be cleared via software.
2	BORST	**Brown-out reset enable** This bit decides whether a brown-out reset is caused by a power drop below VBOD. 0 = Brown-out reset when V_{DD} drops below VBOD Disabled. 1 = Brown-out reset when V_{DD} drops below VBOD Enabled.
1	BORF	**Brown-out reset flag** When the MCU is reset by brown-out event, this bit will be set via hardware. This flag is recommended to be cleared via software.
0	BOS	**Brown-out status** This bit indicates the V_{DD} voltage level comparing with VBOD while BOD circuit is enabled. It keeps 0 if BOD is not enabled. 0 = V_{DD} voltage level is higher than VBOD or BOD is disabled. 1 = V_{DD} voltage level is lower than VBOD. Note that this bit is read-only.

IAPTRG – IAP Trigger

Register	SFR Address	Reset Value
IAPTRG	A4H, Page 0, TA protected	0000 _0000 b

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	IAPGO
-	-	-	-	-	-	-	W

Bit	Name	Description
7:1	-	Reserved

Bit	Name	Description

0	IAPGO	**IAP go**
		IAP begins by setting this bit as logic 1. After this instruction, the CPU holds the Program Counter (PC) and the IAP hardware automation takes over to control the progress. After IAP action completed, the Program Counter continues to run the following instruction. The IAPGO bit will be automatically cleared and always read as logic 0.
		Before triggering an IAP action, interrupts (if enabled) should be temporary disabled for hardware limitation. The program process should follows below.
		CLR EA
		MOV TA,#0AAH
		MOV TA,#55H
		ORL IAPTRG,#01H
		(SETB EA)

IAPUEN – IAP Updating Enable

Register	SFR Address	Reset Value
IAPUEN	A5H, Page 0, TA protected	0000 _0000 b

7	6	5	4	3	2	1	0
-	-	-	SPMEN	SPUEN	CFUEN	LDUEN	APUEN
-	-	-	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:5	-	Reserved
4	SPMEN	**SPROM Memory space mapping enable** 0 = CPU memory address 0xff80~0xffff is mapping to APROM memory 1 = CPU memory address 0xff80~0xffff is mapping to SPROM memory
3	SPUEN	**SPROM Memory space updated enable(TA protected)** 0 = Inhibit erasing or programming SPROM bytes by IAP 1 = Allow erasing or programming SPROM bytes by IAP.
2	CFUEN	**CONFIG bytes updated enable** 0 = Inhibit erasing or programming CONFIG bytes by IAP. 1 = Allow erasing or programming CONFIG bytes by IAP.
1	LDUEN	**LDROM updated enable** 0 = Inhibit erasing or programming LDROM by IAP. 1 = Allow erasing or programming LDROM by IAP.
0	APUEN	**APROM updated enable** 0 = Inhibit erasing or programming APROM by IAP. 1 = Allow erasing or programming APROM by IAP.

IAPAL – IAP Address Low Byte

Register	SFR Address	Reset Value
IAPAL	A6H, Page 0	0000 _0000 b

7	6	5	4	3	2	1	0
IAPA[7:0]							
R/W							

Bit	Name	Description
7:0	IAPA[7:0]	**IAP address low byte** IAPAL contains address IAPA[7:0] for IAP operations.

IAPAH – IAP Address High Byte

Register	SFR Address	Reset Value
IAPAH	A7H, Page 0	0000 _0000 b

7	6	5	4	3	2	1	0
IAPA[15:8]							
R/W							

Bit	Name	Description
7:0	IAPA[15:8]	**IAP address high byte** IAPAH contains address IAPA[15:8] for IAP operations.

IE – Interrupt Enable

Register	SFR Address	Reset Value
IE	A8H, All pages, Bit addressable	0000 _0000 b

7	6	5	4	3	2	1	0
EA	EADC	EBOD	ES	ET1	EX1	ET0	EX0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	EA	**Enable all interrupt** This bit globally enables/disables all interrupts that are individually enabled. 0 = All interrupt sources Disabled. 1 = Each interrupt Enabled depending on its individual mask setting. Individual interrupts will occur if enabled.

Bit	Name	Description
6	EADC	**Enable ADC interrupt** 0 = ADC interrupt Disabled. 1 = ADC interrupt Enable. When interrupt generated ADCF (ADCCON0.7) set 1.
5	EBOD	**Enable brown-out interrupt** 0 = Brown-out detection interrupt Disabled. 1 = Brown-out detection interrupt Enable. When interrupt generated BOF (BODCON0.3) set 1.
4	ES	**Enable serial port 0 interrupt** 0 = Serial port 0 interrupt Disabled. 1 = Serial port 0 interrupt Enable. When interrupt generated TI (SCON.1) or RI (SCON.0) set 1.
3	ET1	**Enable Timer 1 interrupt** 0 = Timer 1 interrupt Disabled. 1 = Timer 1 interrupt Enable. When interrupt generated TF1 (TCON.7) set 1.
2	EX1	**Enable external interrupt 1** 0 = External interrupt 1 Disabled. 1 = External interrupt 1 interrupt Enable. When interrupt generated $\overline{INT1}$ pin set 1.
1	ET0	**Enable Timer 0 interrupt** 0 = Timer 0 interrupt Disabled. 1 = Timer 0 interrupt Enable. When interrupt generated TF0 (TCON.5) set 1.
0	EX0	**Enable external interrupt 0** 0 = External interrupt 0 Disabled. 1 = External interrupt 0 interrupt Enable. When interrupt generated $\overline{INT0}$ pin set 1.

SADDR – Slave 0 Address

Register	SFR Address	Reset Value
SADDR	A9H, Page 0	0000 _0000 b

7	6	5	4	3	2	1	0
SADDR[7:0]							
R/W							

Bit	Name	Description
7:0	SADDR[7:0]	**Slave 0 address** This byte specifies the microcontroller's own slave address for UATR0 multi-processor communication.

VRFCON – Internal V_{REF} Control

Register	SFR Address	Reset Value
VRFCON	A9H, Page 1, TA protected	0000 _0000 b

7	6	5	4	3	2	1	0
-	VRFSEL[2:0]			-	-	ENLOAD	ENVRF
-	R/W			-	-	R/W	R/W

Bit	Name	Description
7	-	Reserved
6:4	VRFSEL	**Internal V_{REF} Output Voltage Select** This field selects V_{REF} output voltage. 000 = 1.538V , when V_{DD} > 2.0V 001 = 2.048V , when V_{DD} > 2.4V 010 = 2.560V , when V_{DD} > 2.9V 011 = 3.072V , when V_{DD} > 3.4V 100 = 4.096V , when V_{DD} > 4.5V 101 = reserved 110 = reserved 111 = reserved
3:2	-	Reserved
1	ENLOAD	**Internal V_{REF} Pre-Load enable** 1 = Internal V_{REF} Pre-load Enabled. 0 = Internal V_{REF} Pre-load Disabled
0	ENVRF	**Internal V_{REF} enable** 1 = Internal V_{REF} Enabled, 0 = Internal V_{REF} Disabled Note that a 1 uF has to add on V_{REF} pin while internal V_{REF} is enabled.

WDCON – Watchdog Timer Control

Register	SFR Address	Reset Value
WDCON	AAH, Page 0, TA protected	POR: 0000 0111b WDT: 0000 1UUUb Others: 0000 UUUUb

7	6	5	4	3	2	1	0
WDTR	WDCLR	WDTF	WIDPD	WDTRF	WDPS[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W		

Bit	Name	Description

Bit	Name	Description
7	WDTR	**WDT run** This bit is valid only when control bits in WDTEN[3:0] (CONFIG4[7:4]) are all 1. At this time, WDT works as a general purpose timer. 0 = WDT Disabled. 1 = WDT Enabled. The WDT counter starts running.
6	WDCLR	**WDT clear** Setting this bit will reset the WDT count to 00H. It puts the counter in a known state and prohibit the system from unpredictable reset. The meaning of writing and reading WDCLR bit is different. Writing: 0 = No effect. 1 = Clearing WDT counter. Reading: 0 = WDT counter is completely cleared. 1 = WDT counter is not yet cleared.
5	WDTF	**WDT time-out flag** This bit indicates an overflow of WDT counter. This flag should be cleared by software.
4	WIDPD	**WDT running in Idle or Power-down mode** This bit is valid only when control bits in WDTEN[3:0] (CONFIG4[7:4]) are all 1. It decides whether WDT runs in Idle or Power-down mode when WDT works as a general purpose timer. 0 = WDT stops running during Idle or Power-down mode. 1 = WDT keeps running during Idle or Power-down mode.
3	WDTRF	**WDT reset flag** When the CPU is reset by WDT time-out event, this bit will be set via hardware. This flag is recommended to be cleared via software after reset.
2:0	WDPS[2:0]	**WDT clock pre-scalar select** These bits determine the pre-scale of WDT clock from 1/1 through 1/256. See Table 12.2-1 The default is the maximum pre-scale value.

VRFTRIM – Internal V_{REF} Trim Select

Register	SFR Address	Reset Value
VRFTRIM	AAH, Page 1, TA protected	0100_0000 b

7	6	5	4	3	2	1	0
-	VRFTRIM[6:0]						
-	R/W						

Bit	Name	Description
7	-	Reserved

Bit	Name	Description
6:0	VRFTIRM[6:0]	**Internal V_{REF} Trim Select** default=7'b1000000 Output MAX=7'b1111110; Output MIN=7'b0000000 1111111 = Untrimmed nature voltage = Medium voltage (1000000)

BODCON1 – Brown-out Detection Control 1

Register	SFR Address	Reset Value
BODCON1	ABH, Page 0, TA protected	POR: 0000_0001b Others:0000_0UUUb

7	6	5	4	3	2	1	0
-	-	-	-	-	LPBOD[1:0]		BODFLT
-	-	-	-	-	R/W		R/W

Bit	Name	Description
7:3	-	Reserved
2:1	LPBOD[1:0]	**Low power BOD enable** 00 = BOD normal mode. BOD circuit is always enabled. 01 = BOD low power mode 1 by turning on BOD circuit every 1.6 ms periodically. 10 = BOD low power mode 2 by turning on BOD circuit every 6.4 ms periodically. 11 = BOD low power mode 3 by turning on BOD circuit every 25.6 ms periodically.
0	BODFLT	**BOD filter control** BOD has a filter which counts 32 clocks of F_{SYS} to filter the power noise when MCU runs with HIRC, or ECLK as the system clock and BOD does not operates in its low power mode (LPBOD[1:0] = [0, 0]). In other conditions, the filter counts 2 clocks of LIRC. Note that when CPU is halted in Power-down mode. The BOD output is permanently filtered by 2 clocks of LIRC. The BOD filter avoids the power noise to trigger BOD event. This bit controls BOD filter enabled or disabled. 0 = BOD filter Disabled. 1 = BOD filter Enabled. (Power-on reset default value.)

ACMPCR2 – Analog Comparator Control Register 2

Register	SFR Address	Reset Value
ACMPCR2	ABH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	-	AO1PEN	AO0PEN	-	-	CRVSSEL	CRVEN
-	-	R/W	R/W	-	-	R/W	R/W

Bit	Name	Description
7:6	-	Reserved
5	AO1PEN	**Analog comparator 1 output to pin enable control** 0 = pin output disabled 1 = pin output to P4.0 Enabled
4	AO0PEN	**Analog comparator 0 output to pin enable control** 0 = pin output disabled 1= output to P4.1 Enabled
3:2	-	Reserved
1	CRVSSEL	**CRV Source Voltage Selection** 0 = V_{DD} is selected as CRV source voltage. 1 = The reference voltage (V_{REF}) is selected as CRV source voltage.
0	CRVEN	**CRV Enable Bit** 0 = CRV Disabled. 1 = CRV Enabled.

EIP2 – Extensive Interrupt Priority 2

Register	SFR Address	Reset Value
EIP2	ACH, Page 0	0000_0000 b

Note: EIP2 is used in combination with the EIPH2 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
-	PDMA3	PDMA2	SMC1	PFB1	PPWM1	PI^{2}C1	PACMP
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	PDMA3	PDMA3 interrupt priority low bit
5	PDMA2	PDMA2 interrupt priority low bit
4	SMC1	SMC1 interrupt priority low bit
3	PFB1	Fault Brake1 interrupt priority low bit
2	PPWM1	PPWM1 interrupt priority low bit
1	PI^{2}C1	I^{2}C interrupt priority low bit
0	PACMP	ACMP interrupt priority low bit

EIPH2 – Extensive Interrupt Priority High 2

Register	SFR Address	Reset Value
EIPH2	ADH, Page 0	0000_0000 b

Note: EIPH2 is used in combination with the EIP2 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
-	PDMA3H	PDMA2H	SMC1H	PFB1H	PPWM1H	PI^{2}C1H	PACMPH
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	PDMA3H	PDMA3H interrupt priority high bit
5	PDMA2H	PDMA2H interrupt priority high bit
4	SMC1H	SMC1H interrupt priority high bit
3	PFB1H	Fault Brake1 interrupt priority high bit
2	PPWM1H	PPWM1H interrupt priority high bit
1	PI^{2}C1H	I^{2}C interrupt priority high bit
0	PACMPH	ACMP interrupt priority high bit

IAPFD – IAP Flash Data

Register	SFR Address	Reset Value
IAPFD	AEH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
IAPFD[7:0]							
R/W							

Bit	Name	Description
7:0	IAPFD[7:0]	**IAP Flash data** This byte contains Flash data, which is read from or is going to be written to the Flash Memory. User should write data into IAPFD for program mode before triggering IAP processing and read data from IAPFD for read/verify mode after IAP processing is finished.

IAPCN – IAP Control

Register	SFR Address	Reset Value
IAPCN	AFH, Page 0	0011_0000 b

7	6	5	4	3	2	1	0
IAPB[1:0]		FOEN	FCEN	FCTRL[3:0]			
R/W		R/W	R/W	R/W			

Note: This byte is used for IAP command. For details, see Table 27.1-1 IAP Modes and Command Codes.

PnM1 – Port 0 Mode Select 1

Register	SFR Address	Reset Value
P0M1	B1H, Page 1	1111_1111 b
P1M1	B3H, Page 1	1111_1111 b
P2M1	B5H, Page 1	1111_1111 b
P3M1	C2H, Page 1	1111_1111 b
P4M1	B9H, Page 1	1111_1111 b
P5M1	BDH, Page 1	1111_1111 b

Note: PnM1 and PnM2 [n:0~5] are used in combination to determine the I/O mode of each pin of P0. See Table 7.2-1 Configuration for Different I/O Modes

7	6	5	4	3	2	1	0
PnM1.7	PnM1.6	PnM1.5	PnM1.4	PnM1.3	PnM1.2	PnM1.1	PnM1.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnM1[7:0]	Port 0 mode select 1

PnM2 – Port 0 Mode Select 2

Register	SFR Address	Reset Value
P0M2	B2H, Page 1	0000_0000 b
P1M2	B4H, Page 1	0000_0000 b
P2M2	B6H, Page 1	0000_0000 b
P3M2	C3H, Page 1	0000_0000 b
P4M2	BAH, Page 1	0000_0000 b
P5M2	BEH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnM2.7	PnM2.6	PnM2.5	PnM2.4	PnM2.3	PnM2.2	PnM2.1	PnM2.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnM2[7:0]	Port 0 mode select 2

I2C1DAT – I²C1 Data

Register	SFR Address	Reset Value
I2C1DAT	B3H, Page 0	0000 _0000 b

7	6	5	4	3	2	1	0
I²C1DAT[7:0]							
R/W							

Bit	Name	Description
7:0	I2C1DAT[7:0]	**I²C1 data** I²CnDAT contains a byte of the I²C data to be transmitted or a byte, which has just received. Data in I²CnDAT remains as long as SI is logic 1. The result of reading or writing I²CnDAT during I²C transceiver progress is unpredicted. While data in I²CnDAT is shifted out, data on the bus is simultaneously being shifted in to update I²CnDAT. I²CnDAT always shows the last byte that presented on the I²C bus. Thus the event of lost arbitration, the original value of I²CnDAT changes after the transaction.

I2STAT1 – I²C1 Status

Register	SFR Address	Reset Value
I2C1DAT	B4H, Page 0	1111_1000 b

7	6	5	4	3	2	1	0
I2STAT1[7:3]					0	0	0
R					R	R	R

Bit	Name	Description
7:3	I2STAT1[7:3]	**I²C1 status code** The MSB five bits of I²CnSTAT contains the status code. There are 27 possible status codes. When I²CnSTAT is F8H, no relevant state information is available and SI flag keeps 0. All other 26 status codes correspond to the I²C states. When each of these status is entered, SI will be set as logic 1 and a interrupt is requested.
2:0	0	Reserved The least significant three bits of I²CnSTAT are always read as 0.

I2C1TOC – I²C1 Time-out Counter

Register	SFR Address	Reset Value
I2C1TOC	B6H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	-	I2TOCEN	DIV	I2TOF
-	-	-	-	-	R/W	R/W	R/W

Bit	Name	Description
7:3	-	Reserved
2	I2TOCEN	**I²C1 time-out counter enable** 0 = I²C time-out counter Disabled. 1 = I²C time-out counter Enabled.
1	DIV	**I²C1 time-out counter clock divider** 0 = The clock of I²C time-out counter is FSYS/1. 1 = The clock of I²C time-out counter is FSYS/4.
0	I2TOF	**I²C1 time-out flag** This flag is set by hardware if 14-bit I²C time-out counter overflows. It is cleared by software.

IPH – Interrupt Priority High

Register	SFR Address	Reset Value
IPH	B7H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	PADCH	PBODH	PSH	PT1H	PX1H	PT0H	PX0H
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	PADC	ADC interrupt priority high bit
5	PBOD	Brown-out detection interrupt priority high bit
4	PSH	Serial port 0 interrupt priority high bit
3	PT1H	Timer 1 interrupt priority high bit
2	PX1H	External interrupt 1 priority high bit
1	PT0H	Timer 0 interrupt priority high bit

Bit	Name	Description
0	PX0H	External interrupt 0 priority high bit

PWMnINTC – PWM Interrupt Control

Register	SFR Address	Reset Value
PWM0INTC	B7H, Page 1	0000_0000 b
PWM1INTC	BCH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	INTTYP1	INTTYP0	-	INTSEL2	INTSEL1	INTSEL0
-	-	R/W	R/W	-	R/W	R/W	R/W

Bit	Name	Description
5:4	INTTYP[1:0]	**PWM interrupt type select** These bit select PWM interrupt type. 00 = Falling edge on PWM0/1/2/3/4/5 pin. 01 = Rising edge on PWM0/1/2/3/4/5 pin. 10 = Central point of a PWM period. 11 = End point of a PWM period. Note that the central point interrupt or the end point interrupt is only available while PWM operates in center-aligned type.
2:0	INTSEL[2:0]	**PWM interrupt pair select** These bits select which PWM channel asserts PWM interrupt when PWM interrupt type is selected as falling or rising edge on PWM0/1/2/3/4/5 pin.. 000 = PWM0. 001 = PWM1. 010 = PWM2. 011 = PWM3. 100 = PWM4. 101 = PWM5. Others = PWM0.

IP – Interrupt Priority

Register	SFR Address	Reset Value
IP	B8H, All pages, Bit-addressable	0000_0000 b

Note: IP is used in combination with the IPH to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
-	PADC	PBOD	PS	PT1	PX1	PT0	PX0
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	PADC	ADC interrupt priority low bit
5	PBOD	Brown-out detection interrupt priority low bit
4	PS	Serial port 0 interrupt priority low bit
3	PT1	Timer 1 interrupt priority low bit
2	PX1	External interrupt 1 priority low bit
1	PT0	Timer 0 interrupt priority low bit
0	PX0	External interrupt 0 priority low bit

SADEN – Slave 0 Address Mask

Register	SFR Address	Reset Value
SADEN	B9H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SADEN[7:0]							
R/W							

Bit	Name	Description
7:0	SADEN[7:0]	**Slave 0 address mask** This byte is a mask byte of UART0 that contains “don’t-care” bits (defined by zeros) to form the device’s “Given” address. The don’t-care bits provide the flexibility to address one or more slaves at a time.

SADEN1 – Slave 1 Address Mask

Register	SFR Address	Reset Value
SADEN1	BAH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SADEN1[7:0]							
R/W							

Bit	Name	Description

Bit	Name	Description
7:0	SADEN1[7:0]	**Slave 1 address mask** This byte is a mask byte of UART1 that contains “don’t-care” bits (defined by zeros) to form the device’s “Given” address. The don’t-care bits provide the flexibility to address one or more slaves at a time.

SADDR1 – Slave 1 Address

Register	SFR Address	Reset Value
SADDR1	BBH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SADDR1[7:0]							
R/W							

Bit	Name	Description
7:0	SADDR1[7:0]	**Slave 1 address** This byte specifies the microcontroller’s own slave address for UART1 multi-processor communication.

I2CnDAT – I^2C Data

Register	SFR Address	Reset Value
I2C0DAT	BCH, Page 0	0000_0000 b
I2C1DAT	B3H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
I^2C0DAT[7:0]							
R/W							

Bit	Name	Description
7:0	I2CnDAT[7:0]	**I^2C0 data** I^2CnDAT contains a byte of the I^2C data to be transmitted or a byte, which has just received. Data in I^2CnDAT remains as long as SI is logic 1. The result of reading or writing I^2CnDAT during I^2C transceiver progress is unpredicted. While data in I^2CnDAT is shifted out, data on the bus is simultaneously being shifted in to update I^2CnDAT. I^2CnDAT always shows the last byte that presented on the I^2C bus. Thus the event of lost arbitration, the original value of I^2CnDAT changes after the transaction.

I2CnSTAT – I²C Status

Register	SFR Address	Reset Value
I2C0STAT	BDH, Page 0	1111_1000 b
I2C1STAT	B4H, Page 0	1111_1000 b

7	6	5	4	3	2	1	0
I²C0STAT[7:3]					0	0	0
R					R	R	R

Bit	Name	Description
7:3	I²C0STAT7:3]	**I²C0 status code** The MSB five bits of I²CnSTAT contains the status code. There are 27 possible status codes. When I²CnSTAT is F8H, no relevant state information is available and SI flag keeps 0. All other 26 status codes correspond to the I²C states. When each of these status is entered, SI will be set as logic 1 and a interrupt is requested.
2:0	0	Reserved The least significant three bits of I²CnSTAT are always read as 0.

I2CnCLK – I²C Clock

Register	SFR Address	Reset Value
I2C0CLK	BEH, Page 0	0000_1001 b
I2C1CLK	B5H, Page 0	0000_1001 b

7	6	5	4	3	2	1	0
I2C0CLK[7:0]							
R/W							

Bit	Name	Description
7:0	I2CnCLK[7:0]	**I²C0 clock setting** In master mode: This register determines the clock rate of I²C bus when the device is in a master mode. The clock rate follows the equation, $\frac{F_{SYS}}{4\times(I2CLK+1)}$. Note that the I²CnCLK value of 00H and 01H are not valid. This is an implement limitation. In slave mode: This byte has no effect. In slave mode, the I²C device will automatically synchronize with any given clock rate up to 400k bps.

I2C0TOC – I²C0 Time-out Counter

Register	SFR Address	Reset Value
I2C0TOC	BFH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	-	I2TOCEN	DIV	I2TOF
-	-	-	-	-	R/W	R/W	R/W

Bit	Name	Description
7:3	-	Reserved
2	I2TOCEN	**I²C0 time-out counter enable** 0 = I²C time-out counter Disabled. 1 = I²C time-out counter Enabled.
1	DIV	**I²C0 time-out counter clock divider** 0 = The clock of I²C time-out counter is F_{SYS}/1. 1 = The clock of I²C time-out counter is F_{SYS}/4.
0	I2TOF	**I²C0 time-out flag** This flag is set by hardware if 14-bit I²C time-out counter overflows. It is cleared by software.

I2CnCON – I²C Control

Register	SFR Address	Reset Value
I2C0CON	C0H, Page 0, Bit-addressable	0000_0000 b
I2C1CON	E8H, Page 0, Bit-addressable	0000_0000 b

7	6	5	4	3	2	1	0
I	I²CEN	STA	STO	SI	AA	-	-
R/W	R/W	R/W	R/W	R/W	R/W	-	-

Bit	Name	Description
7	I	**I²C0 hold time extend enable** 0 = I²C DATA to SCL hold time extend disabled 1 = I²C DATA to SCL hold time extend enabled, extend 8 system clock
6	I²CEN	**I²C0 bus enable** 0 = I²C bus Disabled. 1 = I²C bus Enabled. Before enabling the I²C, SCL and SDA port latches should be set to logic 1.

Bit	Name	Description
5	STA	**START flag** When STA is set, the I²C generates a START condition if the bus is free. If the bus is busy, the I²C waits for a STOP condition and generates a START condition following. If STA is set while the I²C is already in the master mode and one or more bytes have been transmitted or received, the I²C generates a repeated START condition. Note that STA can be set anytime even in a slave mode, but STA is not hardware automatically cleared after START or repeated START condition has been detected. User should take care of it by clearing STA manually.
4	STO	**STOP flag** When STO is set if the I²C is in the master mode, a STOP condition is transmitted to the bus. STO is automatically cleared by hardware once the STOP condition has been detected on the bus. The STO flag setting is also used to recover the I²C device from the bus error state (I²CnSTAT as 00H). In this case, no STOP condition is transmitted to the I²C bus. If the STA and STO bits are both set and the device is original in the master mode, the I²C bus will generate a STOP condition and immediately follow a START condition. If the device is in slave mode, STA and STO simultaneous setting should be avoid from issuing illegal I²C frames.
3	SI	**I²C0 interrupt flag** SI flag is set by hardware when one of 26 possible I²C status (besides F8H status) is entered. After SI is set, the software should read I²CnSTAT register to determine which step has been passed and take actions for next step. SI is cleared by software. Before the SI is cleared, the low period of SCL line is stretched. The transaction is suspended. It is useful for the slave device to deal with previous data bytes until ready for receiving the next byte. The serial transaction is suspended until SI is cleared by software. After SI is cleared, I²C bus will continue to generate START or repeated START condition, STOP condition, 8-bit data, or so on depending on the software configuration of controlling byte or bits. Therefore, user should take care of it by preparing suitable setting of registers before SI is software cleared.
2	AA	**Acknowledge assert flag** If the AA flag is set, an ACK (low level on SDA) will be returned during the acknowledge clock pulse of the SCL line while the I²C device is a receiver or an own-address-matching slave. If the AA flag is cleared, a NACK (high level on SDA) will be returned during the acknowledge clock pulse of the SCL line while the I²C device is a receiver or an own-address-matching slave. A device with its own AA flag cleared will ignore its own salve address and the General Call. Consequently, SI will note be asserted and no interrupt is requested. Note that if an addressed slave does not return an ACK under slave receiver mode or not receive an ACK under slave transmitter mode, the slave device will become a not addressed slave. It cannot receive any data until its AA flag is set and a master addresses it again. There is a special case of I²CnSTAT value C8H occurs under slave transmitter mode. Before the slave device transmit the last data byte to the master, AA flag can be cleared as 0. Then after the last data byte transmitted, the slave device will actively switch to not addressed slave mode of disconnecting with the master. The further reading by the master will be all FFH.
1:0	-	Reserved

I2CnADDRx – I²Cn Own Slave Address

Register	SFR Address	Reset Value
I²C0ADDR0	C1H, Page 0	0000_0000 b
I²C0ADDR1	A1H, Page 2	0000_0000 b
I²C0ADDR2	A2H, Page 2	0000_0000 b
I²C0ADDR3	A3H, Page 2	0000_0000 b
I²C1ADDR0	B2H, Page 0	0000_0000 b
I²C1ADDR1	A4H, Page 2	0000_0000 b
I²C1ADDR2	A5H, Page 2	0000_0000 b
I²C1ADDR3	A6H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
I²CnADDRx[7:1]							GC
R/W							R/W

Bit	Name	Description
7:1	I²C0ADDRx [7:1]	**I²C0 device's own slave address** In master mode: These bits have no effect. In slave mode: These 7 bits define the slave address of this I²C device by user. The master should address I²C device by sending the same address in the first byte data after a START or a repeated START condition. If the AA flag is set, this I²C device will acknowledge the master after receiving its own address and become an addressed slave. Otherwise, the addressing from the master will be ignored. Note that I²CnADDRx[7:1] should not remain its default value of all 0, because address 0x00 is reserved for General Call.
6	GC	**General Call bit** In master mode: This bit has no effect. In slave mode: 0 = The General Call is always ignored. 1 = The General Call is recognized if AA flag is 1; otherwise, it is ignored if AA is 0.

CKDIV – Clock Divider

Register	SFR Address	Reset Value
CKDIV	C1H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
CKDIV[7:0]							
R/W							

Bit	Name	Description
7:0	CKDIV[7:0]	**Clock divider** The system clock frequency F_{SYS} follows the equation below according to CKDIV value. $F_{SYS} = F_{OSC}$, while CKDIV = 00H, $F_{SYS} = \frac{F_{OSC}}{2 \times CKDIV}$, while CKDIV = 01H to FFH.

ADCRL – ADC Result Low Byte

Register	SFR Address	Reset Value
ADCRL	C2H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	ADCR[3:0]			
-	-	-	-	R			

Bit	Name	Description
7:4	-	Reserved
3:0	ADCR[3:0]	**ADC result low byte** The least significant 4 bits of the ADC result stored in this register.

ADCRH – ADC Result High Byte

Register	SFR Address	Reset Value
ADCRH	C3H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCR[11:4]							
R							

Bit	Name	Description
7:0	ADCR[11:4]	ADC result high byte The most significant 8 bits of the ADC result stored in this register.

T3CON – Timer 3 Control

Register	SFR Address	Reset Value
T3CON	C4H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SMOD_1	SMOD0_1	BRCK	TF3	TR3	T3PS[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W		

Bit	Name	Description
7	SMOD_1	**Serial port 1 double baud rate enable** Setting this bit doubles the serial port baud rate when UART1 is in Mode 2.see Table 15.1-2 Serial Port 1 Mode / baudrate Description for details.
6	SMOD0_1	**Serial port 1 framing error access enable** 0 = S1CON.7 accesses to SM0_1 bit. 1 = S1CON.7 accesses to FE_1 bit.
5	BRCK	**Serial port 0 baud rate clock source** This bit selects which Timer is used as the baud rate clock source when serial port 0 is in Mode 1 or 3. 0 = Timer 1. 1 = Timer 3.
4	TF3	**Timer 3 overflow flag** This bit is set when Timer 3 overflows. It is automatically cleared by hardware when the program executes the Timer 3 interrupt service routine. This bit can be set or cleared by software.
3	TR3	**Timer 3 run control** 0 = Timer 3 is halted. 1 = Timer 3 starts running. Note that the reload registers RH3 and RL3 can only be written when Timer 3 is halted (TR3 bit is 0). If any of RH3 or RL3 is written if TR3 is 1, result is unpredictable.
2:0	T3PS[2:0]	**Timer 3 pre-scalar** These bits determine the scale of the clock divider for Timer 3. 000 = 1/1. 001 = 1/2. 010 = 1/4. 011 = 1/8. 100 = 1/16. 101 = 1/32. 110 = 1/64. 111 = 1/128.

RL3 – Timer 3 Reload Low Byte

Register	SFR Address	Reset Value
RL3	C5H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
RL3[7:0]							
R/W							

Bit	Name	Description
7:0	RL3[7:0]	**Timer 3 reload low byte** It holds the low byte of the reload value of Timer 3.

RH3 – Timer 3 Reload High Byte

Register	SFR Address	Reset Value
RH3	C6H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
RH3[7:0]							
R/W							

Bit	Name	Description
7:0	RH3[7:0]	**Timer 3 reload high byte** It holds the high byte of the reload value of Time 3.

PORDIS – POR Disable (TA protected)

Register	SFR Address	Reset Value
PORDIS	C6H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PORDIS[7:0]							
W							

Bit	Name	Description
7:0	PORDIS[7:0]	**POR Disable** To first writing 5AH to the PORDIS and immediately followed by a writing of A5H will disable all of PORs (POR50 and POR15).

TA – Timed Access

Register	SFR Address	Reset Value
TA	C7H, All pages	0000_0000 b

7	6	5	4	3	2	1	0
TA[7:0]							
W							

Bit	Name	Description
7:0	TA[7:0]	**Timed access** The timed access register controls the access to protected SFR. To access protected bits, user should first write AAH to the TA and immediately followed by a write of 55H to TA. After these two steps, a writing permission window is opened for 4 clock cycles during this period that user may write to protected SFR.

T2CON – Timer 2 Control

Register	SFR Address	Reset Value
T2CON	C8H, All pages	0000_0000 b

7	6	5	4	3	2	1	0
TF2	-	-	-	-	TR2	-	CM/$\overline{RL2}$
R/W	-	-	-	-	R/W	-	R/W

Bit	Name	Description
7	TF2	**Timer 2 overflow flag** This bit is set when Timer 2 overflows or a compare match occurs. If the Timer 2 interrupt and the global interrupt are enable, setting this bit will make CPU execute Timer 2 interrupt service routine. This bit is not automatically cleared via hardware and should be cleared via software.
5:3	-	Reserved
2	TR2	**Timer 2 run control** 0 = Timer 2 Disabled. Clearing this bit will halt Timer 2 and the current count will be preserved in TH2 and TL2. 1 = Timer 2 Enabled.
1	-	Reserved
0	CM/$\overline{RL2}$	**Timer 2 compare or auto-reload mode select** This bit selects Timer 2 functioning mode. 0 = Auto-reload mode. 1 = Compare mode.

T2MOD – Timer 2 Mode

Register	SFR Address	Reset Value
T2MOD	C9H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
LDEN	T2DIV[2:0]			CAPCR	CMPCR	LDTS[1:0]	
R/W	R/W			R/W	R/W	R/W	

Bit	Name	Description
7	LDEN	**Enable auto-reload** 0 = Reloading RCMP2H and RCMP2L to TH2 and TL2 Disabled. 1 = Reloading RCMP2H and RCMP2L to TH2 and TL2 Enabled.
6:4	T2DIV[2:0]	**Timer 2 clock divider** 000 = Timer 2 clock divider is 1/1. 001 = Timer 2 clock divider is 1/4. 010 = Timer 2 clock divider is 1/16. 011 = Timer 2 clock divider is 1/32. 100 = Timer 2 clock divider is 1/64. 101 = Timer 2 clock divider is 1/128. 110 = Timer 2 clock divider is 1/256. 111 = Timer 2 clock divider is 1/512.
3	CAPCR	**Capture auto-clear** This bit is valid only under Timer 2 auto-reload mode. It enables hardware auto-clearing TH2 and TL2 counter registers after they have been transferred in to RCMP2H and RCMP2L while a capture event occurs. 0 = Timer 2 continues counting when a capture event occurs. 1 = Timer 2 value is auto-cleared as 0000H when a capture event occurs.
2	CMPCR	**Compare match auto-clear** This bit is valid only under Timer 2 compare mode. It enables hardware auto-clearing TH2 and TL2 counter registers after a compare match occurs. 0 = Timer 2 continues counting when a compare match occurs. 1 = Timer 2 value is auto-cleared as 0000H when a compare match occurs.
1:0	LDTS[1:0]	**Auto-reload trigger select** These bits select the reload trigger event. 00 = Reload when Timer 2 overflows. 01 = Reload when input capture 0 event occurs. 10 = Reload when input capture 1 event occurs. 11 = Reload when input capture 2 event occurs.

AUXR1 – Auxiliary Register 1

Register	SFR Address	Reset Value
T2MOD	C9H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	-	UART2PX	UART1PX	UART0PX
-	-	-	-	-	R/W	R/W	R/W

Bit	Name	Description
7:3	-	Reserved
2	UART2PX	**Serial port 2 RX (SMC0 DATA) /TX (SMC0 CLK) pin exchange** 0 = Assign UART2 RXD (SMC0 DATA) to multiple I/O pin RXD UART2 TXD (SMC CLK) to multiple I/O pin TXD 1 = Assign UART2 RXD (SMC0 DATA) to multiple I/O pin TXD UART2 TXD (SMC CLK) to multiple I/O pin RXD Note : that Pin direction is controlled by I/O type of relative pin. RXD/TXD will exchange immediately once setting or clearing this bit. User should take care of not exchanging pins during transmission or receiving. Or it may cause unpredictable situation and no warning alarms.
1	UART1PX	**Serial port 1 RX/TX pin exchange** 0 = Assign UART1 RXD to multiple I/O pin RXD UART1 TXD to multiple I/O pin TXD 1 = Assign UART1 RXD to multiple I/O pin TXD UART1 TXD to multiple I/O pin RXD **Note:** that Pin direction is controlled by I/O type of relative pin. RXD/TXD will exchange immediately once setting or clearing this bit. User should take care of not exchanging pins during transmission or receiving. Or it may cause unpredictable situation and no warning alarms.
0	UART0PX	**Serial port 0 RX/TX pin exchange** 0 = Assign UART0 RXD to multiple I/O pin RXD UART0 TXD to multiple I/O pin TXD 1 = Assign UART0 RXD to multiple I/O pin TXD UART0 TXD to multiple I/O pin RXD **Note:** that Pin direction is controlled by I/O type of relative pin. RXD/TXD will exchange immediately once setting or clearing this bit. User should take care of not exchanging pins during transmission or receiving. Or it may cause unpredictable situation and no warning alarms.

PIF – Pin Interrupt Flags

Register	SFR Address	Reset Value
PIF	CAH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0

R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)

Bit	Name	Description
7:0	PIFn	**Pin interrupt channel n flag** If the edge trigger is selected, this flag will be set by hardware if the channel n of pin interrupt detects an enabled edge trigger. This flag should be cleared by software. If the level trigger is selected, this flag follows the inverse of the input signal's logic level on the channel n of pin interrupt. Software cannot control it.

RCMP2L– Timer 2 Reload/Compare Low Byte

Register	SFR Address	Reset Value
RCMP2L	CAH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
RCMP2L[7:0]							
R/W							

Bit	Name	Description
7:0	RCMP2L[7:0]	**Timer 2 reload/compare low byte** This register stores the low byte of compare value when Timer 2 is configured in compare mode. Also it holds the low byte of the reload value in auto-reload mode.

ADCBAL – ADC RAM Base Address Low Byte

Register	SFR Address	Reset Value
ADCBAL	CBH, Page:0	0000_0000 b

7	6	5	4	3	2	1	0
ADCBAL[7:0]							
R/W							

Bit	Name	Description
7:0	ADCBAL[7:0]	**ADC RAM base address (Low byte)** The least significant 8 bits of RAM base address to store ADC continue sampling data. RAM base address ADCBA[11:0] = { ADCBAH[3:0], ADCBAL[7:0]}

RCMP2H – Timer 2 Reload/Compare High Byte

Register	SFR Address	Reset Value
RCMP2H	CBH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
RCMP2H[7:0]							
R/W							

Bit	Name	Description
7:0	RCMP2H[7:0]	**Timer 2 reload/compare high byte** This register stores the high byte of compare value when Timer 2 is configured in compare mode. Also it holds the high byte of the reload value in auto-reload mode.

TL2 – Timer 2 Low Byte

Register	SFR Address	Reset Value
TL2	CCH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
TL2[7:0]							
R/W							

Bit	Name	Description
7:0	TL2[7:0]	**Timer 2 low byte** The TL2 register is the low byte of the 16-bit counting register of Timer 2.

TH2 – Timer 2 High Byte

Register	SFR Address	Reset Value
TH2	CDH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
TH2[7:0]							
R/W							

Bit	Name	Description
7:0	TH2[7:0]	**Timer 2 high byte** The TH2 register is the high byte of the 16-bit counting register of Timer 2.

ADCMPL – ADC Compare Low Byte

Register	SFR Address	Reset Value
ADCMPL	CEH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	ADCMP[3:0]			

-	-	-	-	W/R

Bit	Name	Description
7:4	-	Reserved
3:0	ADCMP[3:0]	**ADC compare low byte** The least significant 4 bits of the ADC compare value stores in this register.

AINDIDS – ADC Channel Digital Input Disconnect

Register	SFR Address	Reset Value
AINDIDS	CEH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
AIN7DIDS	AIN6DIDS	AIN5DIDS	AIN4DIDS	AIN3DIDS	AIN2DIDS	AIN1DIDS	AIN0DIDS
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	AINnDIDS	**ADC Channel digital input disable** 0 = Enabled digital input at ADC channel n. 1 = Disabled digital input at ADC channel n . ADC channel n is read always 0.

ADCMPH – ADC Compare High Byte

Register	SFR Address	Reset Value
AINDIDS	CFH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCMP[11:4]							
W/R							

Bit	Name	Description
7:0	ADCMP[11:4]	**ADC compare high byte** The most significant 8 bits of the ADC compare value stores in this register.

PSW – Program Status Word

Register	SFR Address	Reset Value
PSW	D0H, All pages, Bit-addressable	0000_0000 b

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

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Name	Description
7	CY	Carry flag For a adding or subtracting operation, CY will be set when the previous operation resulted in a carry-out from or a borrow-in to the Most Significant bit, otherwise cleared. If the previous operation is MUL or DIV, CY is always 0. CY is affected by DA A instruction, which indicates that if the original BCD sum is greater than 100. For a CJNE branch, CY will be set if the first unsigned integer value is less than the second one. Otherwise, CY will be cleared.
6	AC	Auxiliary carry Set when the previous operation resulted in a carry-out from or a borrow-in to the 4th bit of the low order nibble, otherwise cleared.
5	F0	User flag 0 The general purpose flag that can be set or cleared by user.
4	RS1	Register bank selection bits These two bits select one of four banks in which R0 to R7 locate.
3	RS0	
2	OV	Overflow flag OV is used for a signed character operands. For a ADD or ADDC instruction, OV will be set if there is a carry out of bit 6 but not out of bit 7, or a carry out of bit 7 but not bit 6. Otherwise, OV is cleared. OV indicates a negative number produced as the sum of two positive operands or a positive sum from two negative operands. For a SUBB, OV is set if a borrow is needed into bit6 but not into bit 7, or into bit7 but not bit 6. Otherwise, OV is cleared. OV indicates a negative number produced when a negative value is subtracted from a positive value, or a positive result when a positive number is subtracted from a negative number. For a MUL, if the product is greater than 255 (00FFH), OV will be set. Otherwise, it is cleared. For a DIV, it is normally 0. However, if B had originally contained 00H, the values returned in A and B will be undefined. Meanwhile, the OV will be set.
1	F1	User flag 1 The general purpose flag that can be set or cleared by user via software.
0	P	Parity flag Set to 1 to indicate an odd number of ones in the accumulator. Cleared for an even number of ones. It performs even parity check.

RS1	RS0	Register Bank	RAM Address
0	0	0	00H to 07H
0	1	1	08H to 0FH
1	0	2	10H to 17H
1	1	3	18H to 1FH

PWMnCON0 – PWM Control register0

Register	SFR Address	Reset Value
PWM0CON0	D1H, Page 0	0000_0000 b
PWM1CON0	C5H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PWMRUN	LOAD	PWMF	CLRPWM	-	-	-	-
R/W	R/W	R/W	R/W	-	-	-	-

Bit	Name	Description
7	PWMRUN	PWM run enable 0 = PWM stays in idle. 1 = PWM starts running.
6	LOAD	PWM new period and duty load This bit is used to load period and duty control registers in their buffer if new period or duty value needs to be updated. The loading will act while a PWM period is completed. The new period and duty affected on the next PWM cycle. After the loading is complete, LOAD will be automatically cleared via hardware. The meaning of writing and reading LOAD bit is different. Writing: 0 = No effect. 1 = Load new period and duty in their buffers while a PWM period is completed. Reading: 0 = A loading of new period and duty is finished. 1 = A loading of new period and duty is not yet finished.
5	PWMF	PWM flag This flag is set according to definitions of INTSEL[2:0] and INTTYP[1:0] in PWMnINTC. This bit is cleared by software.
4	CLRPWM	Clear PWM counter Setting this bit clears the value of PWM 16-bit counter for resetting to 0000H. After the counter value is cleared, CLRPWM will be automatically cleared via hardware. The meaning of writing and reading CLRPWM bit is different. Writing: 0 = No effect. 1 = Clearing PWM 16-bit counter. Reading: 0 = PWM 16-bit counter is completely cleared. 1 = PWM 16-bit counter is not yet cleared.
3:0	-	Reserved

PWMnPH – PWM Period High Byte

Register	SFR Address	Reset Value
PWM0PH	D1H, Page 1	0000_0000 b
PWM1PH	C9H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PWMP[15:8]							

R/W

Bit	Name	Description
7:0	PWMP[15:8]	PWM period high byte This byte with PWMnPL controls the period of the PWM generator signal.

ACMPCR0 – Analog Comparator Control Register 0

Register	SFR Address	Reset Value
ACMPCR0	D2H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
POSSEL		NEGSEL		WKEN	HYSEN	ACMPIE	ACMPEN
R/W		R/W		R/W	R/W	R/W	R/W

Bit	Name	Description
7:6	POSSEL	Comparator 0 positive Input Selection 00 = ACMP0_P0 (P2.5) pin. 01 = ACMP0_P1 (P2.3) pin. 10 = ACMP0_P2 (P2.1) pin. 11 = ACMP0_P3 (P3.0) pin.
5:4	NEGSEL	Comparator 0 Negative Input Selection 00 = ACMP0_N0 (P2.4) pin. 01 = Internal comparator reference voltage (CRV). 10 = VBG (Bandgap). 11 = ACMP0_N1 (P2.0)pin.
3	WKEN	Comparator 0 Power-down Wake-up Enable Bit 0 = Comparator 0 Wake-up function Disabled. 1 = Comparator 0 Wake-up function Enabled.
2	HYSEN	Comparator 0 Hysteresis Enable Bit 0 = Comparator 0 hysteresis Disabled. 1 = Comparator 0 hysteresis Enabled.
1	ACMPIE	Comparator 0 Interrupt Enable Bit 0 = Comparator 0 interrupt Disabled. 1 = Comparator 0 interrupt Enabled. If WKEN (ACMPCR1[3]) is set to 1, the wake-up interrupt function will be enabled as well.
0	ACMPEN	Comparator 0 Enable Bit 0 = Comparator 0 Disabled. 1 = Comparator 0 Enabled.

PWMnCxH – PWM0/1 Channel 0~5 Duty High Byte n=0,1; x=0,1,2,3,4,5

Register	SFR Address	Description	Reset Value
PWM0C0H	D2H, Page 1	PWM0 Channel 0 Duty High Byte	0000_0000 b
PWM0C1H	D3H, Page 1	PWM0 Channel 1 Duty High Byte	0000_0000 b
PWM0C2H	D4H, Page 1	PWM0 Channel 2 Duty High Byte	0000_0000 b
PWM0C3H	D5H, Page 1	PWM0 Channel 3 Duty High Byte	0000_0000 b
PWM0C4H	C4H, Page 1	PWM0 Channel 4 Duty High Byte	0000_0000 b
PWM0C5H	C5H, Page 1	PWM0 Channel 5 Duty High Byte	0000_0000 b
PWM1C0H	CAH, Page 2	PWM1 Channel 0 Duty High Byte	0000_0000 b
PWM1C1H	CBH, Page 2	PWM1 Channel 1 Duty High Byte	0000_0000 b
PWM1C2H	CCH, Page 2	PWM1 Channel 2 Duty High Byte	0000_0000 b
PWM1C3H	CDH, Page 2	PWM1 Channel 3 Duty High Byte	0000_0000 b
PWM1C4H	CEH, Page 2	PWM1 Channel 4 Duty High Byte	0000_0000 b
PWM1C5H	CFH, Page 2	PWM1 Channel 5 Duty High Byte	0000_0000 b

7	6	5	4	3	2	1	0
PWMnCx [15:8], n=0,1; x=0,1,2,3,4,5							
R/W							

Bit	Name	Description
7:0	PWMnCx[15:8] n=0,1 x=0,1,2,3,4,5	PWMnCx duty high byte This byte with PWMnC0x controls the duty of the output signal PGx from PWM generator.

ACMPCR1 – Analog Comparator Control Register 1

Register	SFR Address	Reset Value
ACMPCR1	D3H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
POSSEL		NEGSEL		WKEN	HYSEN	ACMPIE	ACMPEN
R/W		R/W		R/W	R/W	R/W	R/W

Bit	Name	Description
7:6	POSSEL	Comparator 1 positive Input Selection 00 = ACMP1_P0 (P2.5) pin. 01 = ACMP1_P1 (P2.3) pin. 10 = ACMP1_P2 (P2.1) pin. 11 = ACMP1_P3 (P3.0) pin.

Bit	Name	Description
5:4	NEGSEL	Comparator 1 Negative Input Selection 00 = ACMP1_N0 (P2.2) pin. 01 = Internal comparator reference voltage (CRV). 10 = VBG (Bandgap). 11 = ACMP1_N1 (P3.1)pin.
3	WKEN	Comparator 1 Power-down Wake-up Enable Bit 0 = Comparator 1 Wake-up function Disabled. 1 = Comparator 1 Wake-up function Enabled.
2	HYSEN	Comparator 1 Hysteresis Enable Bit 0 = Comparator 1 hysteresis Disabled. 1 = Comparator 1 hysteresis Enabled.
1	ACMPIE	Comparator 1 Interrupt Enable Bit 0 = Comparator 1 interrupt Disabled. 1 = Comparator 1 interrupt Enabled. If WKEN (ACMPCR2[3]) is set to 1, the wake-up interrupt function will be enabled as well.
0	ACMPEN	Comparator 1 Enable Bit 0 = Comparator 1 Disabled. 1 = Comparator 1 Enabled.

ACMPSR – Analog Comparator Status Register

Register	SFR Address	Reset Value
ACMPSR	D4H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	ACMP1O	ACMP1IF	ACMP0O	ACMP0IF
-	-	-	-	R	R/W	R	R/W

Bit	Name	Description
7:4	-	Reserved
3	ACMP1O	Comparator 1 Output Synchronized to the PCLK to allow reading by software. Cleared when the comparator 1 is disabled, i.e. ACMPEN (ACMPCR1[0]) is cleared to 0. **Note:** This bit is read only.
2	ACMP1IF	Comparator 1 Interrupt Flag This bit is set by hardware whenever the comparator 1 output changes state. This will generate an interrupt if ACMPIE (ACMPCR1[1]) is set to 1 **Note:** Write “0” to clear this bit to 0.

Bit	Name	Description
1	ACMP0O	Comparator 0 Output Synchronized to the PCLK to allow reading by software. Cleared when the comparator 0 is disabled, i.e. ACMPEN (ACMPCR0[0]) is cleared to 0. **Note:** This bit is read only.
0	ACMP0IF	Comparator 0 Interrupt Flag This bit is set by hardware whenever the comparator 0 output changes state. This will generate an interrupt if ACMPIE (ACMPCR0[1]) is set to 1 **Note:** Write "0" to clear this bit to 0.

ACMPVREF – ACMP Reference Voltage Control Register

Register	SFR Address	Reset Value
ACMPVREF	D5H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	CRV1CTL[2:0]			-	CRV0CTL[2:0]		
-	R/W			-	R/W		

Bit	Name	Description
7	-	Reserved
6:4	CRV1CTL[2:0]	Comparator 1 Reference Voltage Setting CRV1 = CRV source voltage * (2/12+CRV1CTL/12).
3	-	Reserved
2:0	CRV0CTL[2:0]	Comparator 0 Reference Voltage Setting CRV0 = CRV source voltage * (2/12+CRV0CTL/12).

SCnCR0 – SC Control Register

Register	SFR Address	Reset Value
SC0CR0	D6H, Page 1	0000_0000 b
SC1CR0	DEH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
NSB	T	RXBGTEN	CONSEL	AUTOCEN	TXOFF	RXOFF	SCEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description

Bit	Name	Description
7	NSB	Stop Bit Length This field indicates the length of stop bit. 0 = The stop bit length is 2 ETU. 1= The stop bit length is 1 ETU. **Note:** The default stop bit length is 2. SC and UART adopt NSB to program the stop bit length.
6	T	T Mode 0 = T0 (ISO7816-3 T = 0 mode). 1 = T1 (ISO7816-3 T = 1 mode). The T mode controls the BGT (Block Guard Time). Block guard time means the minimum bit length between the leading edges of two consecutive characters between different transfer directions. This field indicates the counter for the bit length of block guard time. According to ISO7816-3, in T = 0 mode, the software must clear T bit to 0 for real block guard time = 16.5. In T = 1 mode, the software must set T bit to 1 for real block guard time = 22.5. **Note:** In T = 0 mode, the receiver will generate the parity flag on PEF(SC0TSR[4]), if parity error is detected and also drive the parity error signal to transceiver. In T = 1 mode, the receiver will generate the parity flag on PEF(SC0TSR[4]), if parity error detected, but doesn't drive the parity error signal to transceiver. **Note:** The description please see section 16.2.4 Error Signal and Character Repetition
5	RXBGTEN	Receiver Block Guard Time Function Enable Bit 0 = Receiver block guard time function Disabled. 1 = Receiver block guard time function Enabled.
4	CONSEL	Convention Selection 0 = Direct convention. 1 = Inverse convention. **Note1:** This bit is auto clear to "0", if AUTOCEN(SCnCR0[3]) is writing "1" **Note2:** If AUTOCEN(SCnCR0[3]) is enabled, hardware will decide the convention and change the CONSEL (SCnCR0[4]) bits automatically after SCEN (SCnCR0[0]) ="1".
3	AUTOCEN	Auto Convention Enable Bit 0 = Auto-convention Disabled. 1 = Auto-convention Enabled. When hardware receives TS in answer to reset state and the TS is direct convention, CONSEL(SCnCR0[4]) will be set to 0 automatically, otherwise if the TS is inverse convention, and CONSEL (SCnCR0[4]) will be set to 1. **Note:** If software enables auto convention function, the setting step must be done before Answer to Reset state and the first data must be 0x3B or 0x3F. After hardware received first data and stored it at buffer, hardware will decided the convention and change the CONSEL (SCnCR0[4]) bits automatically.
2	TXOFF	TX Transition Disable Bit 0 = The transceiver Enabled. 1 = The transceiver Disabled.
1	RXOFF	RX Transition Disable Bit 0 = The receiver Enabled. 1 = The receiver Disabled. **Note:** If AUTOCEN (SCnCR0[3])is enabled, these fields must be ignored.

Bit	Name	Description
0	SCEN	SC Engine Enable Bit Set this bit to 1 to enable SC operation. If this bit is cleared, SC will force all transition to IDLE state **Note:** SCEN must be set to 1 before filling in other registers, or smart card will not work properly.

PWMnNP – PWM Negative Polarity

Register	SFR Address	Reset Value
PWM0NP	D6H, Page 1	0000_0000 b
PWM1NP	BAH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	PNP5	PNP4	PNP3	PNP2	PNP1	PNP0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
5:0	PNPn	PWMn negative polarity output enable 0 = PWMn signal outputs directly on PWMn pin. 1 = PWMn signal outputs inversely on PWMn pin.

SCnCR1 – SC Control Register

Register	SFR Address	Reset Value
SC0CR1	D7H, Page 0	0000_0000 b
SC1CR1	E7H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
OPE	PBOFF	WLS		TXDMAEN	RXDMAEN	CLKKEEP	UARTEN
R/W	R/W	R/W		R/W	R/W	R/W	R/W

Bit	Name	Description
7	OPE	Odd Parity Enable Bit 0 = Even number of logic 1's are transmitted or check the data word and parity bits in receiving mode. 1 = Odd number of logic 1's are transmitted or check the data word and parity bits in receiving mode. **Note:** This bit has effect only when PBOFF bit is '0'.

Bit	Name	Description
6	PBOFF	Parity Bit Disable Control 0 = Parity bit is generated or checked between the “last data word bit” and “stop bit” of the serial data. 1 = Parity bit is not generated (transmitting data) or checked (receiving data) during transfer. **Note:** In smart card mode, this field must be ‘0’ (default setting is with parity bit)
5:4	WLS	Word Length Selection 00 = Word length is 8 bits. 01 = Word length is 7 bits. 10 = Word length is 6 bits. 11 = Word length is 5 bits. **Note:** In smart card mode, this WLS must be ‘00’
3	TXDMAEN	SC/UART TX DMA enable This bit enables the SC/UART TX operating by through PDMA transfer, TX data needs to be ready in XRAM before SC/UART TX starting. 0 = SPI TX DMA Disabled 1 = SPI TX DMA Enabled
2	RXDMAEN	SC/UART RX DMA enable This bit enables the SC/UART RX operating by through PDMA transfer, RX data are saved in XRAM after SC/UART RX operation. 0 = SC/UART RX DMA Disabled 1 = SC/UART RX DMA Enabled
1	CLKKEEP	SC Clock Enable Bit 0 = SC clock generation Disabled. 1 = SC clock always keeps free running.
0	UARTEN	UART Mode Enable Bit 0 = Smart Card mode. 1 = UART mode. **Note1:**When operating in UART mode, user must set CONSEL (SCnCR0[4]) = 0 and AUTOCEN(SCnCR0[3]) = 0. **Note2:**When operating in Smart Card mode, user must set UARTEN(SCnCR1 [0]) = 0. **Note3:**When UART is enabled, hardware will generate a reset to reset FIFO and internal state machine.

PWMnFBD – PWM Fault Brake Data

Register	SFR Address	Reset Value
PWM0FBD	D7H, Page 1	0000_0000 b
PWM1FBD	BBH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
FBF	FBINLS	FBD5	FBD4	FBD3	FBD2	FBD1	FBD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	FBF	Fault Brake flag This flag is set when FBINEN is set as 1 and FB pin detects an edge, which matches FBINLS (PWMnFBD.6) selection. This bit is cleared by software. After FBF is cleared, Fault Brake data output will not be released until PWMRUN (PWMnCON0.7) is set.
6	FBINLS	FB pin input level selection 0 = Falling edge. 1 = Rising edge.
5:0	FBDx	PWMnCHx Fault Brake data 0 = PWMn channel x signal is overwritten by 0 once Fault Brake asserted. 1 = PWMn channel x signal is overwritten by 1 once Fault Brake asserted. **Note:** n = 0,1; x = 0,1,2,3,4,5

SCnDR – SC Data Register

Register	SFR Address	Reset Value
SC0DR	D9H, Page 1	0000_0000 b
SC1DR	D9H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
SCnDR[7:0]							
R/W							

Bit	Name	Description
7:0	SCnDR[7:0]	SC / UART buffer data This byte is used for transmitting or receiving data on SC / UART bus. A write of this byte is a write to the shift register. A read of this byte is actually a read of the read data buffer. **Note:** If SCEN(SCnCR0[0]) is not enabled, SCnDR cannot be programmed.

PWMnPL – PWM Period Low Byte

Register	SFR Address	Reset Value
PWM0PL	D9H, Page 1	0000_0000 b
PWM1PL	D1H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PWMnP[7:0]							
R/W							

Bit	Name	Description

Bit	Name	Description
7:0	PWMnP[7:0]	PWMn period low byte This byte with PWMnPH controls the period of the PWM generator signal.

SCnEGT – SC Extra Guard Time Register

Register	SFR Address	Reset Value
SC0EGT	DAH, Page 0	0000_0000 b
SC1EGT	DAH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
SCnEGT[7:0]							
R/W							

Bit	Name	Description
7:0	SCnEGT [7:0]	SC Extra Guard Time This field indicates the extra guard timer value. **Note:** The counter is ETU base .

PWMnCxL – PWM0/1 Channel 0~5 Duty Low Byte n=0,1; x=0,1,2,3,4,5

Register	SFR Address	Description	Reset Value
PWM0C0L	D2H, Page 1	PWM0 Channel 0 Duty Low Byte	0000_0000 b
PWM0C1L	D3H, Page 1	PWM0 Channel 1 Duty Low Byte	0000_0000 b
PWM0C2L	D4H, Page 1	PWM0 Channel 2 Duty Low Byte	0000_0000 b
PWM0C3L	D5H, Page 1	PWM0 Channel 3 Duty Low Byte	0000_0000 b
PWM0C4L	C4H, Page 1	PWM0 Channel 4 Duty Low Byte	0000_0000 b
PWM0C5L	C5H, Page 1	PWM0 Channel 5 Duty Low Byte	0000_0000 b
PWM1C0L	CAH, Page 2	PWM1 Channel 0 Duty Low Byte	0000_0000 b
PWM1C1L	CBH, Page 2	PWM1 Channel 1 Duty Low Byte	0000_0000 b
PWM1C2L	CCH, Page 2	PWM1 Channel 2 Duty Low Byte	0000_0000 b
PWM1C3L	CDH, Page 2	PWM1 Channel 3 Duty Low Byte	0000_0000 b
PWM1C4L	CEH, Page 2	PWM1 Channel 4 Duty Low Byte	0000_0000 b
PWM1C5L	CFH, Page 2	PWM1 Channel 5 Duty Low Byte	0000_0000 b

7	6	5	4	3	2	1	0
PWMnCx [7:0], n=0,1; x=0,1,2,3,4,5							
R/W							

Bit	Name	Description
7:0	PWMnCx[7:0] n=0,1 x=0,1,2,3,4,5	PWMnCx duty low byte This byte with PWMnCxH controls the duty of the output signal PGx from PWM generator.

SCnETURD0 – SCn ETU Rate Divider Register

Register	SFR Address	Reset Value
SC0ETURD0	DBH, Page 1	0111_0011 b
SC1ETURD0	DBH, Page 2	0111_0011 b

7	6	5	4	3	2	1	0
ETURDIV[7:0]							
R/W							

Bit	Name	Description
7:0	ETURDIV[7:0]	LSB bits of ETU Rate Divider The field indicates the LSB of clock rate divider. The real ETU is ETURDIV[11:0] + 1. **Note1:** ETURDIV[11:0] must be greater than 0x004. **Note2:** SCnETURD0 has to program first, then SCnETUDR2.

SCnETURD1 –SC ETU Rate Divider Register

Register	SFR Address	Reset Value
SC0ETURD1	DCH, Page 0	0011_0001 b
SC1ETURD1	DCH, Page 2	0011_0001 b

7	6	5	4	3	2	1	0
-	SCDIV[2:0]			ETURDIV[11:8]			
-	R/W			R/W			

Bit	Name	Description
7	-	Reserved

Bit	Name	Description
6:4	SCDIV [2:0]	SC clock divider 000 = F_{SC} is F_{SYS}/1. 001 = F_{SC} is F_{SYS}/2. 010 = F_{SC} is F_{SYS}/4. 011 = F_{SC} is F_{SYS}/8. (By default.) 100 = F_{SC} is F_{SYS}/16. 101 = F_{SC} is F_{SYS}/16. 110 = F_{SC} is F_{SYS}/16. 111 = F_{SC} is F_{SYS}/16. **Note:** that the F_{SC} clock should be 1Mhz ~ 5Mhz for ISO/IEC 7816-3 standard
3:0	ETURDIV [11:8]	MSB bits of ETU Rate Divider The field indicates the MSB of clock rate divider. The real ETU is ETURDIV[11:0] + 1. **Note1:** ETURDIV[11:0] must be greater than 0x004. **Note2:** SCnETURD0 has to program first, then SCnETUDR1 .

SCnIE – SC Interrupt Enable Control Register

Register	SFR Address	Reset Value
SC0IE	DDH, Page 0	0000_0000 b
SC1IE	DCH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	ACERRIEN	BGTIEN	TERRIEN	TBEIEN	RDAIEN
-	-	-	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:4	-	Reserved
4	ACERRIEN	Auto Convention Error Interrupt Enable Bit This field is used to enable auto-convention error interrupt. 0 = Auto-convention error interrupt Disabled. 1 = Auto-convention error interrupt Enabled.
3	BGTIEN	Block Guard Time Interrupt Enable Bit This field is used to enable block guard time interrupt. 0 = Block guard time interrupt Disabled. 1 = Block guard time interrupt Enabled.

Bit	Name	Description
2	TERRIEN	Transfer Error Interrupt Enable Bit This field is used to enable transfer error interrupt. The transfer error states is at SC0TSR register which includes receiver break error BEF(SC0TSR[6]), frame error FEF(SC0TSR[5]), parity error PEF(SC0TSR[4]), receiver buffer overflow error RXOV(SC0TSR[0]) and transmit buffer overflow error TXOV(SC0TSR[2]). 0 = Transfer error interrupt Disabled. 1 = Transfer error interrupt Enabled.
1	TBEIEN	Transmit Buffer Empty Interrupt Enable Bit This field is used to enable transmit buffer empty interrupt. 0 = Transmit buffer empty interrupt Disabled. 1 = Transmit buffer empty interrupt Enabled.
0	RDAIEN	Receive Data Reach Interrupt Enable Bit This field is used to enable received data interrupt. 0 = Receive data interrupt Disabled. 1 = Receive data interrupt Enabled.

SCnIS – SC Interrupt Status Register

Register	SFR Address	Reset Value
SC0IS	DEH, Page 0	0000_0010 b
SC1IS	DEH, Page 2	0000_0010 b

7	6	5	4	3	2	1	0
-	-	-	ACERRIF	BGTIF	TERRIF	TBEIF	RDAIF
-	-	-	R/W	R/W	R	R	R

Bit	Name	Description
7:5	-	Reserved
4	ACERRIF	Auto Convention Error Interrupt Status Flag (Read Only) This field indicates auto convention sequence error. If the received TS at ATR state is neither 0x3B nor 0x3F, this bit will be set. **Note:** This bit is read only, but it can be cleared by writing “0” to it.
3	BGTIF	Block Guard Time Interrupt Status Flag (Read Only) This field is used for block guard time interrupt status flag. **Note1:** This bit is valid when RXBGTEN (SCnCR0[5]) is enabled. **Note2:** This bit is read only, but it can be cleared by writing “0” to it.

Bit	Name	Description
2	TERRIF	Transfer Error Interrupt Status Flag (Read Only) This field is used for transfer error interrupt status flag. The transfer error states is at SC0TSR register which includes receiver break error BEF(SC0TSR[6]), frame error FEF(SC0TSR[5]), parity error PEF(SC0TSR[4]) and receiver buffer overflow error RXOV(SC0TSR[0]) and transmit buffer overflow error TXOV(SC0TSR[2]). **Note:** This field is the status flag of BEF(SC0TSR[6]), FEF(SC0TSR[5]), PEF(SC0TSR[4]), RXOV(SC0TSR[0]) and TXOV(SC0TSR[2]). So, if software wants to clear this bit, software must write "0" to each field.
1	TBEIF	Transmit Buffer Empty Interrupt Status Flag (Read Only) This field is used for transmit buffer empty interrupt status flag. **Note:** This field is the status flag of transmit buffer empty state. If software wants to clear this bit, software must write data to DAT(SCnDR[7:0]) buffer and then this bit will be cleared automatically.
0	RDAIF	Receive Data Reach Interrupt Status Flag (Read Only) This field is used for received data interrupt status flag. **Note:** This field is the status flag of received data. If software reads data from SC_DAT pin, this bit will be cleared automatically.

SCnTSR – SC Transfer Status Register

Register	SFR Address	Reset Value
SC0TSR	DFH, Page 0	0000_1010 b
SC1TSR	DFH, Page 2	0000_1010 b

7	6	5	4	3	2	1	0
ACT	BEF	FEF	PEF	TXEMPTY	TXOV	RXEMPTY	RXOV
R	R/W	R/W	R/W	R	R/W	R	R/W

Bit	Name	Description
7	ACT	Transmit /Receive in Active Status Flag (Read Only) 0 = This bit is cleared automatically when TX/RX transfer is finished 1 = This bit is set by hardware when TX/RX transfer is in active.
6	BEF	Receiver Break Error Status Flag (Read Only) This bit is set to logic 1 whenever the received data input (RX) held in the "spacing state" (logic 0) is longer than a full word transmission time (that is, the total time of "start bit" + data bits + parity + stop bits). . **Note:** This bit is read only, but it can be cleared by writing 0 to it.
5	FET	Receiver Frame Error Status Flag (Read Only) This bit is set to logic 1 whenever the received character does not have a valid "stop bit" (that is, the stop bit following the last data bit or parity bit is detected as logic 0). **Note:** This bit is read only, but it can be cleared by writing 0 to it.

Bit	Name	Description
4	PEF	Receiver Parity Error Status Flag (Read Only) This bit is set to logic 1 whenever the received character does not have a valid "parity bit". **Note:** This bit is read only, but it can be cleared by writing 0 to it.
3	TXEMPTY	Transmit Buffer Empty Status Flag (Read Only) This bit indicates TX buffer empty or not. **Note:** When TX buffer has been transferred to Transmitter Shift Register, hardware sets this bit high. It will be cleared when writing data into DAT(SCnDR[7:0]) (TX buffer not empty).
2	TXOVF	TX Overflow Error Interrupt Status Flag (Read Only) If TX buffer is full, an additional write to DAT(SCnDR[7:0]) will cause this bit be set to "1" by hardware. **Note:** This bit is read only, but it can be cleared by writing 0 to it.
1	RXEMPTY	Receiver Buffer Empty Status Flag(Read Only) This bit indicates RX buffer empty or not. **Note:** When Rx buffer has been read by CPU, hardware sets this bit high. It will be cleared when SC receives any new data.
0	RXOV	RX Overflow Error Status Flag (Read Only) This bit is set when RX buffer overflow. **Note:** This bit is read only, but it can be cleared by writing 0 to it.

PWMnCON1 – PWM Control 1

Register	SFR Address	Reset Value
PWM0CON1	DFH, Page 1	0000_0000 b
PWM1CON1	C6H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PWMMOD[1:0]		GP	PWMTYP	FBINEN	PWMDIV[2:0]		
R/W		R/W	R/W	R/W	R/W		

Bit	Name	Description
7:6	PWMMOD [1:0]	PWM mode select 00 = Independent mode. 01 = Complementary mode. 10 = Synchronized mode. 11 = Reserved.
5	GP	Group mode enable This bit enables the group mode. If enabled, the duty of first three pairs of PWM are decided by PWM01H and PWM01L rather than their original duty control registers. 0 = Group mode Disabled. 1 = Group mode Enabled.

Bit	Name	Description
4	PWMTYP	PWM type select 0 = Edge-aligned PWM. 1 = Center-aligned PWM.
3	FBINEN	FB pin input enable 0 = PWM output Fault Braked by FB pin input Disabled. 1 = PWM output Fault Braked by FB pin input Enabled. Once an edge, which matches FBINLS (PWMnFBD.6) selection, occurs on FB pin, PWM0~5 output Fault Brake data in PWMnFBD register and PWM6/7 remains their states. PWMRUN (PWMnCON0.7) will also be automatically cleared by hardware. The PWM output resumes when PWMRUN is set again.
2:0	PWMDIV[2:0]	PWM clock divider This field decides the pre-scale of PWM clock source. 000 = 1/1. 001 = 1/2 010 = 1/4. 011 = 1/8. 100 = 1/16. 101 = 1/32. 110 = 1/64. 111 = 1/128.

A or ACC – Accumulator

Register	SFR Address	Reset Value
ACC	E0H, All pages, Bit-addressable	0000_0000 b

7	6	5	4	3	2	1	0
ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	ACC[7:0]	Accumulator The A or ACC register is the standard 80C51 accumulator for arithmetic operation.

ADCCON1 – ADC Control 1

Register	SFR Address	Reset Value
ADCCON1	E1H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	HIE	CONT	ETGTYP[1:0]		ADCEX	ADCEN
-	-	R/W	R/W	R/W		R/W	R/W

Bit	Name	Description
7:6	-	Reserved
5	HIE	**ADC Half Done Interrupt Enable** 0 = ADC interrupt is not set while half of A/D conversions are complete in continue mode 1 = ADC interrupt is set while half of A/D conversions are complete in continue mode
4	CONT	**ADC Continue Sampling select** 0 = ADC single sampling, ADC interrupt is set while an A/D conversion is completed 1 = ADC continue sampling. ADC interrupt is set while total A/D conversions are completed
3:2	ETGTYP[1:0]	**External trigger type select** When ADCEX (ADCCON1.1) is set, these bits select which condition triggers ADC conversion. 00 = Falling edge on PWM0/2/4 or STADC pin. 01 = Rising edge on PWM0/2/4 or STADC pin. 10 = Central point of a PWM period. 11 = End point of a PWM period. Note that the central point interrupt or the period point interrupt is only available for PWM center-aligned type.
1	ADCEX	**ADC external conversion trigger select** This bit to select the methods of triggering an A/D conversion. 0 = A/D conversion is started only via setting ADCS bit. 1 = A/D conversion is started via setting ADCS bit or by external trigger source depending on ETGSEL[1:0] and ETGTYP[1:0]. Note that while ADCS is 1 (busy in converting), the ADC will ignore the following external trigger until ADCS is hardware cleared.
0	ADCEN	**ADC enable** 0 = ADC circuit off. 1 = ADC circuit on.

CAPCON0 – Input Capture Control 0

Register	SFR Address	Reset Value
CAPCON0	E1H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	CAPEN2	CAPEN1	CAPEN0	-	CAPF2	CAPF1	CAPF0
-	R/W	R/W	R/W	-	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	CAPEN2	Input capture 2 enable 0 = Input capture channel 2 Disabled. 1 = Input capture channel 2 Enabled.

Bit	Name	Description
5	CAPEN1	Input capture 1 enable 0 = Input capture channel 1 Disabled. 1 = Input capture channel 1 Enabled.
4	CAPEN0	Input capture 0 enable 0 = Input capture channel 0 Disabled. 1 = Input capture channel 0 Enabled.
3	-	Reserved
2	CAPF2	Input capture 2 flag This bit is set by hardware if the determined edge of input capture 2 occurs. This bit should cleared by software.
1	CAPF1	Input capture 1 flag This bit is set by hardware if the determined edge of input capture 1 occurs. This bit should cleared by software.
0	CAPF0	Input capture 0 flag This bit is set by hardware if the determined edge of input capture 0 occurs. This bit should cleared by software.

ADCCON2 – ADC Control 2

Register	SFR Address	Reset Value
ADCCON2	E2H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADFBEN	ADCMPOP	ADCMPEN	ADCMPO	ADCAQT[2:0]			ADCDLY.8
R/W	R/W	R/W	R	R/W			R/W

Bit	Name	Description
7	ADFBEN	ADC compare result asserting Fault Brake enable 0 = ADC asserting Fault Brake Disabled. 1 = ADC asserting Fault Brake Enabled. Fault Brake is asserted once its compare result ADCMPO is 1. Meanwhile, PWM channels output Fault Brake data. PWMRUN (PWMnCON0.7) will also be automatically cleared by hardware. The PWM output resumes when PWMRUN is set again.
6	ADCMPOP	ADC comparator output polarity 0 = ADCMPO is 1 if ADCR[11:0] is greater than or equal to ADCMP[11:0]. 1 = ADCMPO is 1 if ADCR[11:0] is less than ADCMP[11:0].
5	ADCMPEN	ADC result comparator enable 0 = ADC result comparator Disabled. 1 = ADC result comparator Enabled.

Bit	Name	Description
4	ADCMPO	ADC comparator output value This bit is the output value of ADC result comparator based on the setting of ACMPOP. This bit updates after every A/D conversion complete.
3:1	ADCAQT	ADC acquisition time This 3-bit field decides the acquisition time for ADC sampling, following by equation below: ADC acquisition time = $\frac{4 * ADCAQT + 10}{F_{ADC}}$. The default and minimum acquisition time is 10 ADC clock cycles. Note that this field should not be changed when ADC is in converting.
0	ADCDLY.8	ADC external trigger delay counter bit 8 See ADCDLY register.

CAPCON1 – Input Capture Control 1

Register	SFR Address	Reset Value
CAPCON1	E2H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	-	CAP2LS[1:0]		CAP1LS[1:0]		CAP0LS[1:0]	
-	-	R/W		R/W		R/W	

Bit	Name	Description
7:6	-	Reserved
5:4	CAP2LS[1:0]	Input capture 2 level select 00 = Falling edge. 01 = Rising edge. 10 = Either rising or falling edge. 11 = Reserved.
3:2	CAP1LS[1:0]	Input capture 1 level select 00 = Falling edge. 01 = Rising edge. 10 = Either rising or falling edge. 11 = Reserved.
1:0	CAP0LS[1:0]	Input capture 0 level select 00 = Falling edge. 01 = Rising edge. 10 = Either rising or falling edge. 11 = Reserved.

ADCDLY – ADC Trigger Delay Counter

Register	SFR Address	Reset Value
ADCDLY	E3H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCDLY[7:0]							
R/W							

Bit	Name	Description
7:0	ADCDLY[7:0]	ADC external trigger delay counter low byte This 8-bit field combined with ADCCON2.0 forms a 9-bit counter. This counter inserts a delay after detecting the external trigger. An A/D converting starts after this period of delay. External trigger delay time = $\frac{ADCDLY}{F_{ADC}}$. Note that this field is valid only when ADCEX (ADCCON1.1) is set. User should not modify ADCDLY during PWM run time if selecting PWM output as the external ADC trigger source.

CAPCON2 – Input Capture Control 2

Register	SFR Address	Reset Value
ADCDLY	E3H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	ENF2	ENF1	ENF0	-	-	-	-
-	R/W	R/W	R/W	-	-	-	-

Bit	Name	Description
7	-	Reserved
6	ENF2	Enable noise filer on input capture 2 0 = Noise filter on input capture channel 2 Disabled. 1 = Noise filter on input capture channel 2 Enabled.
5	ENF1	Enable noise filer on input capture 1 0 = Noise filter on input capture channel 1 Disabled. 1 = Noise filter on input capture channel 1 Enabled.
4	ENF0	Enable noise filer on input capture 0 0 = Noise filter on input capture channel 0 Disabled. 1 = Noise filter on input capture channel 0 Enabled.
3:0	-	Reserved

ADCBAH – ADC RAM Base Address High byte

Register	SFR Address	Reset Value
ADCBAH	E4H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-				ADCBA[11:8]			
-				R/W			

Bit	Name	Description
7:4	-	Reserved
3:0	ADCBA[11:8]	ADC RAM base address (High byte) The most significant 4 bits of RAM base address to store ADC continue sampling data. RAM base address ADCBA[11:0] = {ADCBAH[3:0], ADCBAL[7:0]}

C0L – Capture 0 Low Byte

Register	SFR Address	Reset Value
C0L	E4H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C0L[7:0]							
R/W							

Bit	Name	Description
7:0	C0L[7:0]	Input capture 0 result low byte The C0L register is the low byte of the 16-bit result captured by input capture 0.

ADCSN – ADC Sampling Number

Register	SFR Address	Reset Value
ADCSN	E5H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCSN[7:0]							
R/W							

Bit	Name	Description
7:0	ADCSN[7:0]	ADC Sampling Number The total sampling numbers for ADC continue sampling select. Total sampling number= ADCSN[7:0] + 1

C0H – Capture 0 High Byte

Register	SFR Address	Reset Value
C0H	E5H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C0H[7:0]							
R/W							

Bit	Name	Description
7:0	C0H[7:0]	Input capture 0 result high byte The C0H register is the high byte of the 16-bit result captured by input capture 0.

ADCCN – ADC Current Sampling Number

Register	SFR Address	Reset Value
ADCCN	E6H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCCN[7:0]							
R/W							

Bit	Name	Description
7:0	ADCCN[7:0]	ADC Current Sampling Number The current sampling numbers for ADC continue sampling select. The current sampling number= ADCCN[7:0] + 1

C1L – Capture 1 Low Byte

Register	SFR Address	Reset Value
C1L	E6H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C1L[7:0]							
R/W							

Bit	Name	Description
7:0	C1L[7:0]	Input capture 1 result low byte The C1L register is the low byte of the 16-bit result captured by input capture 1.

ADCSR – ADC Status Register

Register	SFR Address	Reset Value
ADCSR	E7H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SLOW	ADCDIV[2:0]			-	CMPHIT	HDONE	FDONE
R/W	R/W			-	R/W	R/W	R/W

Bit	Name	Description
7	SLOW	ADC Slow Speed Selection This bit is used to select ADC low speed. 0 = high speed 500 ksps 1 = low speed 200 ksps
6:4	ADCDIV	ADC clock divider 000 = FADC is FSYS/1. 001 = FADC is FSYS/2. 010 = FADC is FSYS/4. 011 = FADC is FSYS/8. 100 = FADC is FSYS/16. 101 = FADC is FSYS/32. 110 = FADC is FSYS/64. 111 = FADC is FSYS/128.
3	-	Reserved
2	CMPHIT	ADC comparator Hit Flag This bit is set by hardware when ADCMPO (ADCCON2.4) flag rising **Note:** This bit can be cleared by writing 0 to it.
1	HDONE	A/D Conversion Half Done Flag This bit is set by hardware when half of ADCSN A/D conversions are complete in continue mode. **Note:** This bit can be cleared by writing 0 to it
0	FDONE	A/D Conversion Full Done Flag This bit is set by hardware when all of ADCSN A/D conversions are complete in continue mode or single conversion in single mode. **Note:** This bit can be cleared by writing 0 to it..

C1H – Capture 1 High Byte

Register	SFR Address	Reset Value
C1H	E7H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C1H[7:0]							
R/W							

Bit	Name	Description
7:0	C1H[7:0]	Input capture 1 result high byte The C1H register is the high byte of the 16-bit result captured by input capture 1.

DMAnTSR – PDMAn Transfer Status Register

Register	SFR Address	Reset Value
DMA0TSR	E9H, Page 0	0000_0000 b
DMA1TSR	F1H, Page 0	0000_0000 b
DMA2TSR	B1H, Page 2	0000_0000 b
DMA3TSR	A9H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	-	ACT	HDONE	FDONE
-	-	-	-	-	R	R/W	R/W

Bit	Name	Description
7:3	-	Reserved
2	ACT	PDMA in Active Status Flag (Read Only) 0 = This bit is cleared automatically when PDMA transfer is done or disabled. 1 = This bit is set by hardware when PDMA transfer is in active.
1	HDONE	PDMA Half Transfer Done Flag This bit is set by hardware when PDMA half transfer is done. **Note:** This bit can be cleared by writing 0 to it.
0	FDONE	PDMA Full Transfer Done Flag This bit is set by hardware when PDMA full transfer is done. **Note:** This bit can be cleared by writing 0 to it.

PICON – Pin Interrupt Control

Register	SFR Address	Reset Value
PICON	E9H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PIT7	PIT6	PIT5	PIT4	PIT3	PIT2	PIT1	PIT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PIT7	Pin interrupt channel 7 type select This bit selects which type that pin interrupt channel 7 is triggered. 0 = Level triggered. 1 = Edge triggered.
6	PIT6	Pin interrupt channel 6 type select This bit selects which type that pin interrupt channel 6 is triggered. 0 = Level triggered. 1 = Edge triggered.
5	PIT5	Pin interrupt channel 5 type select This bit selects which type that pin interrupt channel 5 is triggered. 0 = Level triggered. 1 = Edge triggered.
4	PIT4	Pin interrupt channel 4 type select This bit selects which type that pin interrupt channel 4 is triggered. 0 = Level triggered. 1 = Edge triggered.
3	PIT3	Pin interrupt channel 3 type select This bit selects which type that pin interrupt channel 3 is triggered. 0 = Level triggered. 1 = Edge triggered.
2	PIT2	Pin interrupt channel 2 type select This bit selects which type that pin interrupt channel 2 is triggered. 0 = Level triggered. 1 = Edge triggered.
1	PIT1	Pin interrupt channel 1 type select This bit selects which type that pin interrupt channel 1 is triggered. 0 = Level triggered. 1 = Edge triggered.

Bit	Name	Description
0	PIT0	Pin interrupt channel 0 type select This bit selects which type that pin interrupt channel 0 is triggered. 0 = Level triggered. 1 = Edge triggered.

MTMnDA – Memory to Memory Destination Address Low Byte

Register	SFR Address	Reset Value
MTM0DA	EAH, Page 0	0000_0000 b
MTM1DA	F2H, Page 0	0000_0000 b
MTM2DA	B7H, Page 2	0000_0000 b
MTM3DA	AFH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
MDAL[7:0]							
R/W							

Bit	Name	Description
7:0	MDAL[7:0]	Memory to Memory Destination Address (Low Byte) The least significant 8 bits of XRAM address are used for memory to memory destination address. XRAM destination address = {MDAH[3:0], MDAL[7:0]}

PINEN – Pin Interrupt Negative Polarity Enable.

Register	SFR Address	Reset Value
PINEN	EAH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PINEN7	PINEN6	PINEN5	PINEN4	PINEN3	PINEN2	PINEN1	PINEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PINENn	Pin interrupt channel n negative polarity enable This bit enables low-level/falling edge triggering pin interrupt channel n. The level or edge triggered selection depends on each control bit PITn in PICON. 0 = Low-level/falling edge detect Disabled. 1 = Low-level/falling edge detect Enabled.

PIPEN – Pin Interrupt Positive Polarity Enable.

Register	SFR Address	Reset Value
PIPEN	EBH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PIPEN7	PIPEN6	PIPEN5	PIPEN4	PIPEN3	PIPEN2	PIPEN1	PIPEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PIPENn	Pin interrupt channel n positive polarity enable This bit enables high-level/rising edge triggering pin interrupt channel n. The level or edge triggered selection depends on each control bit PITn in PICON. 0 = High-level/rising edge detect Disabled. 1 = High-level/rising edge detect Enabled.

C2L – Capture 2 Low Byte

Register	SFR Address	Reset Value
C2L	EDH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C2L[7:0]							
R/W							

Bit	Name	Description
7:0	C2L[7:0]	Input capture 2 result low byte The C2L register is the low byte of the 16-bit result captured by input capture 2.

C2H – Capture 2 High Byte

Register	SFR Address	Reset Value
C2H	EEH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C2H[7:0]							
R/W							

Bit	Name	Description
7:0	C2H[7:0]	Input capture 2 result high byte The C2H register is the high byte of the 16-bit result captured by input capture 2.

EIP0 – Extensive Interrupt Priority

Register	SFR Address	Reset Value
EIP0	EFH, Page 0	0000_0000 b

Note: EIP0 is used in combination with the EIPH0 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
PT2	PSPI0	PFB	PWDT	PPWM0	PCAP	PPI	PI^{2}C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PT2	Timer 2 interrupt priority low bit
6	PSPI0	SPI0 interrupt priority low bit
5	PFB	Fault Brake interrupt priority low bit
4	PWDT	WDT interrupt priority low bit
3	PPWM0	PWM interrupt priority low bit
2	PCAP	Input capture interrupt priority low bit
1	PPI	Pin interrupt priority low bit
0	PI^{2}C0	I^{2}C interrupt priority low bit

B – B Register

Register	SFR Address	Reset Value
B	F0H, All pages, Bit-addressable	0000_0000 b

7	6	5	4	3	2	1	0
B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	B[7:0]	B register The B register is the other accumulator of the standard 80C51 .It is used mainly for MUL and DIV instructions.

SPInCR0 – Serial Peripheral Control Register0

Register	SFR Address	Reset Value
SPI0CR0	F3H, Page 0	0000_0000 b
SPI1CR0	F9H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SSOE	SPIEN	LSBFE	MSTR	CPOL	CPHA	SPR1	SPR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	SSOE	Slave select output enable This bit is used in combination with the DISMODF (SPInSR.3) bit to determine the feature of $\overline{SS}$ pin as shown in Table 17–1. Slave Select Pin Configurations. This bit takes effect only under MSTR = 1 and DISMODF = 1 condition. 0 = $\overline{SS}$ functions as a general purpose I/O pin. 1 = $\overline{SS}$ automatically goes low for each transmission when selecting external Slave device and goes high during each idle state to de-select the Slave device.
6	SPIEN	SPI enable 0 = SPI function Disabled. 1 = SPI function Enabled.
5	LSBFE	LSB first enable 0 = The SPI data is transferred MSB first. 1 = The SPI data is transferred LSB first.
4	MSTR	Master mode enable This bit switches the SPI operating between Master and Slave modes. 0 = The SPI is configured as Slave mode. 1 = The SPI is configured as Master mode.
3	CPOL	SPI clock polarity select CPOL bit determines the idle state level of the SPI clock. See Figure 17.2-3 SPI Clock Formats. 0 = The SPI clock is low in idle state. 1 = The SPI clock is high in idle state.
2	CPHA	SPI clock phase select CPHA bit determines the data sampling edge of the SPI clock. See Figure 17.2-3 SPI Clock Formats. 0 = The data is sampled on the first edge of the SPI clock. 1 = The data is sampled on the second edge of the SPI clock.

Bit	Name	Description
1:0	SPR[1:0]	SPI clock rate select These two bits select four grades of SPI clock divider. The clock rates below are illustrated under F_{SYS} = 24 MHz condition. SPR3 SPR2 SPR1 SPR0 Divider SPI clock rate 0 0 0 0 2 12M bit/s 0 0 0 1 4 6M bit/s 0 0 1 0 8 3M bit/s 0 0 1 1 16 1.5M bit/s 0 1 0 0 32 750k bit/s 0 1 0 1 64 375k bit/s 0 1 1 0 128 187k bit/s 0 1 1 1 256 93.7k bit/s 1 0 0 0 3 8M bit/s 1 0 0 1 6 4M bit/s 1 0 1 0 12 2M bit/s 1 0 1 1 24 1M bit/s 1 1 0 0 48 500k bit/s 1 1 0 1 96 250k bit/s 1 1 1 0 192 125k bit/s 1 1 1 1 384 62.5k bit/s SPR[3:0] are valid only under Master mode (MSTR = 1). If under Slave mode, the clock will automatically synchronize with the external clock on SPICLK pin from Master device up to F_{SYS}/4 communication speed.

SPInCR1 – Serial Peripheral Control Register1

Register	SFR Address	Reset Value
SPI0CR1	F3H, Page 1	0000_0000 b
SPI1CR1	FAH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	SPR3	SPR2	TXDMAEN	RXDMAEN	SPIS1	SPIS0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:6	-	Reserved

<table>
<tr><th>Bit</th><th>Name</th><th>Description</th></tr>
<tr><td>5:4</td><td>SPR[3:2]</td><td>SPI clock rate select
These two bits select four grades of SPI clock divider. The clock rates below are illustrated under F_{SYS} = 24 MHz condition.
<table>
<tr><th>SPR3</th><th>SPR2</th><th>SPR1</th><th>SPR0</th><th>Divider</th><th>SPI clock rate</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>2</td><td>12M bit/s</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>4</td><td>6M bit/s</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>8</td><td>3M bit/s</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>16</td><td>1.5M bit/s</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>32</td><td>750k bit/s</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>64</td><td>375k bit/s</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>128</td><td>187k bit/s</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>256</td><td>93.7k bit/s</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>3</td><td>8M bit/s</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>6</td><td>4M bit/s</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>12</td><td>2M bit/s</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>24</td><td>1M bit/s</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>48</td><td>500k bit/s</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>96</td><td>250k bit/s</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>192</td><td>125k bit/s</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>384</td><td>62.5k bit/s</td></tr>
</table>
SPR[3:0] are valid only under Master mode (MSTR = 1). If under Slave mode, the clock will automatically synchronize with the external clock on SPICLK pin from Master device up to F_{SYS}/4 communication speed.</td></tr>
<tr><td>3</td><td>TXDMAEN</td><td>SPI TX DMA enable
This bit enables the SPI TX operating by through PDMA transfer, TX data needs to be ready in XRAM before SPI TX starting.
0 = SPI TX DMA Disabled
1 = SPI TX DMA Enabled</td></tr>
<tr><td>2</td><td>RXDMAEN</td><td>SPI RX DMA enable
This bit enables the SPI RX operating by through PDMA transfer, RX data are saved in XRAM after SPI RX operation.
0 = SPI RX DMA Disabled
1 = SPI RX DMA Enabled</td></tr>
<tr><td>1:0</td><td>SPIS[1:0]</td><td>SPI Interval time selection between adjacent bytes
SPIS[1:0] and CPHA select eight grades of SPI interval time selection between adjacent bytes. As below table:
<table>
<tr><th>CPHA</th><th>SPIS1</th><th>SPIS0</th><th>SPI clock</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0.0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0.5</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1.5</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>2.0</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0.0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1.0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>2.0</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>2.5</td></tr>
</table>
SPIS[1:0] are valid only under Master mode (MSTR = 1).</td></tr>
</table>

SPInSR – Serial Peripheral Status Register

Register	SFR Address	Reset Value
SPI0SR	F4H, Page 0	0000_0000 b
SPI1SR	FBH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SPIF	WCOL	SPIOVF	MODF	DISMODF	DISSPIF	TXBFF	-
R/W	R/W	R/W	R/W	R/W	R/W	R	-

Bit	Name	Description
7	SPIF	SPI complete flag This bit is set to logic 1 via hardware while an SPI data transfer is complete or an receiving data has been moved into the SPI read buffer. If ESPI and EA are enabled, an SPI interrupt will be required. This bit should be cleared via software. Attempting to write to SPInDR is inhibited if SPIF is set.
6	WCOL	Write collision error flag This bit indicates a write collision event. Once a write collision event occurs, this bit will be set. It should be cleared via software.
5	SPIOVF	SPI overrun error flag This bit indicates an overrun event. Once an overrun event occurs, this bit will be set. If ESPI and EA are enabled, an SPI interrupt will be required. This bit should be cleared via software.
4	MODF	Mode Fault error flag This bit indicates a Mode Fault error event. If $\overline{SS}$ pin is configured as Mode Fault input (MSTR = 1 and DISMODF = 0) and $\overline{SS}$ is pulled low by external devices, a Mode Fault error occurs. Instantly MODF will be set as logic 1. If ESPI and EA are enabled, an SPI interrupt will be required. This bit should be cleared via software.
3	DISMODF	Disable Mode Fault error detection This bit is used in combination with the SSOE (SPInCR.7) bit to determine the feature of $\overline{SS}$ pin as shown in Table 17–1. Slave Select Pin Configurations. DISMODF is valid only in Master mode (MSTR = 1). 0 = Mode Fault detection Enabled. $\overline{SS}$ serves as input pin for Mode Fault detection disregard of SSOE. 1 = Mode Fault detection Disabled. The feature of $\overline{SS}$ follows SSOE bit.
2	DISSPIF	Disable SPI complete interrupt This bit is used to disable SPI complete interrupt while an SPI data transfer is complete or an receiving data has been moved into the SPI read buffer. Especially in SPI DMA operation. 0 = SPI Complete Interrupt Enabled while ESPI and EA are enabled, 1 = SPI Complete Interrupt Disabled
1	TXBFF	SPI TX Buffer Full Flag 0 = SPI TX buffer is empty 1 = SPI TX buffer is full

SPInDR – Serial Peripheral Data Register

Register	SFR Address	Reset Value
SPI0DR	F5H, Page 0	0000_0000 b
SPI1DR	FCH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SPInDR[7:0]							
R/W							

Bit	Name	Description
7:0	SPInDR[7:0]	Serial peripheral data This byte is used for transmitting or receiving data on SPI bus. A write of this byte is a write to the shift register. A read of this byte is actually a read of the read data buffer. In Master mode, a write to this register initiates transmission and reception of a byte simultaneously.

DMAnBAH – PDMAn XRAM Base Address High Byte

Register	SFR Address	Reset Value
DMA0BAH	F6H, Page 0	0000_0000 b
DMA1BAH	FDH, Page 0	0000_0000 b
DMA2BAH	B2H, Page 2	0000_0000 b
DMA3BAH	AAH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
MTMDA [7:4]				XRAMA[7:4]			
R/W				R/W			

Bit	Name	Description
7:4	MDAH[7:0]	Memory to Memory Destination Address (High Byte) The most significant 4 bits of XRAM address are used for memory to memory destination address. XRAM destination address = {MDAH[3:0], MDAL[7:0]}
3:0	MAH[3:0]	PDMA XRAM Base Address (High Byte) The most significant 4 bits of XRAM address to store or read for the peripheral source data; in memory to memory transfer, this register is the destination address. XRAM address = {MAH[3:0], MAL[7:0]}

EIPH0 – Extensive Interrupt Priority High

Register	SFR Address	Reset Value
EIPH0	F7H, All pages	0000_0000 b

Note: EIPH0 is used in combination with the EIP0 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
PT2H	PSPI0H	PFBH	PWDTH	PPWM0H	PCAPH	PPIH	PI^{2}C0H
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PT2H	Timer 2 interrupt priority high bit
6	PSPI0H	SPI0 interrupt priority high bit
5	PFBH	Fault Brake interrupt priority high bit
4	PWDTH	WDT interrupt priority high bit
3	PPWM0H	PWM0 interrupt priority high bit
2	PCAPH	Input capture interrupt priority high bit
1	PPIH	Pin interrupt priority high bit
0	PI^{2}C0H	I^{2}C interrupt priority high bit

S1CON – Serial Port 1 Control

Register	SFR Address	Reset Value
S1CON	F8H, All pages, Bit-addressable	0000_0000 b

7	6	5	4	3	2	1	0
SM0_1/FE_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	SM0_1/FE_1	Serial port 1 mode select
6	SM1_1	SMOD0_1 (T3CON.6) = 0: See Table 15.1-2 Serial Port 1 Mode / baudrate Description for details. SMOD0_1 (T3CON.6) = 1: SM0_1/FE_1 bit is used as frame error (FE) status flag. It is cleared by software. 0 = Frame error (FE) did not occur. 1 = Frame error (FE) occurred and detected.

Bit	Name	Description
5	SM2_1	Multiprocessor communication mode enable The function of this bit is dependent on the serial port 1 mode. Mode 0: No effect. Mode 1: This bit checks valid stop bit. 0 = Reception is always valid no matter the logic level of stop bit. 1 = Reception is valid only when the received stop bit is logic 1 and the received data matches “Given” or “Broadcast” address. Mode 2 or 3: For multiprocessor communication. 0 = Reception is always valid no matter the logic level of the 9th bit. 1 = Reception is valid only when the received 9th bit is logic 1 and the received data matches “Given” or “Broadcast” address.
4	REN_1	Receiving enable 0 = Serial port 1 reception Disabled. 1 = Serial port 1 reception Enabled in Mode 1,2, or 3. In Mode 0, reception is initiated by the condition REN_1 = 1 and RI_1 = 0.
3	TB8_1	9th transmitted bit This bit defines the state of the 9th transmission bit in serial port 1 Mode 2 or 3. It is not used in Mode 0 or 1.
2	RB8_1	9th received bit The bit identifies the logic level of the 9th received bit in serial port 1 Mode 2 or 3. In Mode 1, RB8_1 is the logic level of the received stop bit. SM2_1 bit as logic 1 has restriction for exception. RB8_1 is not used in Mode 0.
1	TI_1	Transmission interrupt flag This flag is set by hardware when a data frame has been transmitted by the serial port 1 after the 8th bit in Mode 0 or the last data bit in other modes. When the serial port 1 interrupt is enabled, setting this bit causes the CPU to execute the serial port 1 interrupt service routine. This bit must be cleared manually via software.
0	RI_1	Receiving interrupt flag This flag is set via hardware when a data frame has been received by the serial port 1 after the 8th bit in Mode 0 or after sampling the stop bit in Mode 1, 2, or 3. SM2_1 bit as logic 1 has restriction for exception. When the serial port 1 interrupt is enabled, setting this bit causes the CPU to execute to the serial port 1 interrupt service routine. This bit must be cleared manually via software.

PWMnDTEN – PWM Dead-time Enable

Register	SFR Address	Reset Value
PWM0DTEN	F9H, Page 1, TA protected	0000_0000 b
PWM1DTEN	C1H, Page 2, TA protected	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	PWMnDTCNT.8	-	PDT45EN	PDT23EN	PDT01EN
-	-	-	R/W	-	R/W	R/W	R/W

Bit	Name	Description
7:5	0	Reserved
4	PWMnDTCNT.8	PWM dead-time counter bit 8 See PWMnDTCNT register.
2	PDT45EN	PWM4/5 pair dead-time insertion enable This bit is valid only when PWM4/5 is under complementary mode. 0 = No delay on GP4/GP5 pair signals. 1 = Insert dead-time delay on the rising edge of GP4/GP5 pair signals.
1	PDT23EN	PWM2/3 pair dead-time insertion enable This bit is valid only when PWM2/3 is under complementary mode. 0 = No delay on GP2/GP3 pair signals. 1 = Insert dead-time delay on the rising edge of GP2/GP3 pair signals.
0	PDT01EN	PWM0/1 pair dead-time insertion enable This bit is valid only when PWM0/1 is under complementary mode. 0 = No delay on GP0/GP1 pair signals. 1 = Insert dead-time delay on the rising edge of GP0/GP1 pair signals.

PWMnDTCNT – PWM Dead-time Counter

Register	SFR Address	Reset Value
PWM0DTCNT	FAH, Page 1, TA protected	0000_0000 b
PWM1DTCNT	C2H, Page 2, TA protected	0000_0000 b

7	6	5	4	3	2	1	0
PWM0DTCNT[7:0]							
R/W							

Bit	Name	Description

Bit	Name	Description
7:0	PWM0DTCNT [7:0]	PWM dead-time counter low byte This 8-bit field combined with PWMnDTEN .4 forms a 9-bit PWM dead-time counter PWM0DTCNT. This counter is valid only when PWM is under complementary mode and the correspond PWMnDTEN bit for PWM pair is set. PWM dead-time = $\frac{PDTCNT+1}{F_{SYS}}$. Note that user should not modify PWM0DTCNT during PWM run time.

PWMnMEN – PWM Mask Enable

Register	SFR Address	Reset Value
PWM0MEN	FBH, Page 1	0000_0000 b
PWM1MEN	C3H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	PMEN5	PMEN4	PMEN3	PMEN2	PMEN1	PMEN0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
5:0	PMENn	PWMn mask enable 0 = PWMn signal outputs from its PWM generator. 1 = PWMn signal is masked by PMDn.

PWMnMD – PWM Mask Data

Register	SFR Address	Reset Value
PWM0MD	FCH, Page 1	0000_0000 b
PWM1MD	C4H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:6	0	Reserved The bits are always read as 0.
5:0	PMDn	PWMn mask data The PWMn signal outputs mask data once its corresponding PMENn is set. 0 = PWMn signal is masked by 0. 1 = PWMn signal is masked by 1.

DMA1BAH – PDMA XRAM Base Address High Byte

Register	SFR Address	Reset Value
DMA1BAH	FDH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
MDAH[3:0]				MAH[3:0]			
R/W				R/W			

Bit	Name	Description
7:4	MDAH[7:0]	Memory to Memory Destination Address (High Byte) The most significant 4 bits of XRAM address are used for memory to memory destination address. XRAM destination address = {MDAH[3:0], MDAL[7:0]}
3:0	MAH[3:0]	PDMA XRAM Base Address (High Byte) The most significant 4 bits of XRAM address to store or read for the peripheral source data; in memory to memory transfer, this register is the destination address. XRAM address = {MAH[3:0], MAL[7:0]}

LVRFLTEN – LVR Filter Enable

Register	SFR Address	Reset Value
LVRFLTEN	FDH, Page 1, TA protected	0000_0000 b

7	6	5	4	3	2	1	0
LVRFLTEN[7:0]							
W							

Bit	Name	Description
7:0	LVRFLTEN[7:0]	LVR18 filter enable To first writing 5AH to the LVRFLTEN and immediately followed by a writing of A5H. Others = Disabled.

EIP1 – Extensive Interrupt Priority 1

Register	SFR Address	Reset Value
EIP1	FEH, Page 0	0000_0000 b

Note: EIP1 is used in combination with the EIPH1 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
PSPI1	PDMA1	PDMA0	PSMC	PHF	PWKT	PT3	PS1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PSPI1	SPI1 interrupt priority low bit
6	PDMA1	PDMA1 interrupt priority low bit
5	PDMA0	PDMA0 interrupt priority low bit
4	PSMC	SMC interrupt priority low bit
3	PHF	Hard fault interrupt priority low bit
2	PWKT	WKT interrupt priority low bit
1	PT3	Timer 3 interrupt priority low bit
0	PS1	Serial port 1 interrupt priority low bit

EIPH1 – Extensive Interrupt Priority High 1

Register	SFR Address	Reset Value
EIPH1	FFH, Page 0	0000_0000 b

Note: EIPH1 is used in combination with the EIP1 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
PSPI1H	PDMA1H	PDMA0H	PSMCH	PHFH	PWKTH	PT3H	PS1H
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PSPI1H	SPI1 interrupt priority high bit
6	PDMA1H	PDMA1 interrupt priority high bit
5	PDMA0H	PDMA0 interrupt priority high bit
4	PSMCH	SMC interrupt priority high bit
3	PHFH	Hard fault interrupt priority high bit
2	PWKTH	WKT interrupt priority high bit
1	PT3H	Timer 3 interrupt priority high bit
0	PS1H	Serial port 1 interrupt priority high bit

LVRDIS – LVR Disable

Register	SFR Address	Reset Value
LVRDIS	FFH, Page 1, TA protected	0000_0000 b

7	6	5	4	3	2	1	0
LVRDIS[7:0]							
W							

Bit	Name	Description
7:0	LVRDIS[7:0]	LVR disable To first writing 5AH to the LVRDIS and immediately followed by a writing of A5H will disable LVR.

8 GENERAL 80C51 SYSTEM CONTROL

A or ACC – Accumulator

Register	SFR Address	Reset Value
ACC	E0H, All pages, Bit-addressable	0000_0000 b

7	6	5	4	3	2	1	0
ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	ACC[7:0]	**Accumulator** The A or ACC register is the standard 80C51 accumulator for arithmetic operation.

B – B Register

Register	SFR Address	Reset Value
B	F0H, All pages	0000_0000 b

7	6	5	4	3	2	1	0
B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	B[7:0]	**B register** The B register is the other accumulator of the standard 80C51 .It is used mainly for MUL and DIV instructions.

SP – Stack Pointer

Register	SFR Address	Reset Value
SP	81H, All pages	0000_0111b

7	6	5	4	3	2	1	0
SP[7:0]							
R/W							

Bit	Name	Description

7:0	SP[7:0]	**Stack pointer** The Stack Pointer stores the scratch-pad RAM address where the stack begins. It is incremented before data is stored during PUSH or CALL instructions. Note that the default value of SP is 07H. This causes the stack to begin at location 08H.

DPL – Data Pointer Low Byte

Register	SFR Address	Reset Value
DPL	82H, All pages	0000_0000b

7	6	5	4	3	2	1	0
DPL[7:0]							
R/W							

Bit	Name	Description
7:0	DPL[7:0]	**Data pointer low byte** This is the low byte of 16-bit data pointer. DPL combined with DPH serve as a 16-bit data pointer DPTR to access indirect addressed RAM or Program Memory. DPS (AUXR0.0) bit decides which data pointer, DPTR or DPTR1, is activated.

DPH – Data Pointer High Byte

Register	SFR Address	Reset Value
DPH	83H, All pages	0000_0000b

7	6	5	4	3	2	1	0
DPH[7:0]							
R/W							

Bit	Name	Description
7:0	DPH[7:0]	**Data pointer high byte** This is the high byte of 16-bit data pointer. DPH combined with DPL serve as a 16-bit data pointer DPTR to access indirect addressed RAM or Program Memory. DPS (AUXR0.0) bit decides which data pointer, DPTR or DPTR1, is activated.

PSW – Program Status Word

Register	SFR Address	Reset Value
PSW	D0H, All pages, Bit-addressable	0000_0000 b

7	6	5	4	3	2	1	0
CY	AC	F0	RS1	RS0	OV	F1	P
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

<table>
<tr><th>Bit</th><th>Name</th><th>Description</th></tr>
<tr><td>7</td><td>CY</td><td>Carry flag
For a adding or subtracting operation, CY will be set when the previous operation resulted in a carry-out from or a borrow-in to the Most Significant bit, otherwise cleared.
If the previous operation is MUL or DIV, CY is always 0.
CY is affected by DA A instruction, which indicates that if the original BCD sum is greater than 100.
For a CJNE branch, CY will be set if the first unsigned integer value is less than the second one. Otherwise, CY will be cleared.</td></tr>
<tr><td>6</td><td>AC</td><td>Auxiliary carry
Set when the previous operation resulted in a carry-out from or a borrow-in to the 4th bit of the low order nibble, otherwise cleared.</td></tr>
<tr><td>5</td><td>F0</td><td>User flag 0
The general purpose flag that can be set or cleared by user.</td></tr>
<tr><td>4</td><td>RS1</td><td rowspan="2">Register bank selection bits
These two bits select one of four banks in which R0 to R7 locate.
<table><tr><th>RS1</th><th>RS0</th><th>Register Bank</th><th>RAM Address</th></tr><tr><td>0</td><td>0</td><td>0</td><td>00H to 07H</td></tr><tr><td>0</td><td>1</td><td>1</td><td>08H to 0FH</td></tr><tr><td>1</td><td>0</td><td>2</td><td>10H to 17H</td></tr><tr><td>1</td><td>1</td><td>3</td><td>18H to 1FH</td></tr></table></td></tr>
<tr><td>3</td><td>RS0</td></tr>
<tr><td>2</td><td>OV</td><td>Overflow flag
OV is used for a signed character operands. For a ADD or ADDC instruction, OV will be set if there is a carry out of bit 6 but not out of bit 7, or a carry out of bit 7 but not bit 6. Otherwise, OV is cleared. OV indicates a negative number produced as the sum of two positive operands or a positive sum from two negative operands. For a SUBB, OV is set if a borrow is needed into bit6 but not into bit 7, or into bit7 but not bit 6. Otherwise, OV is cleared. OV indicates a negative number produced when a negative value is subtracted from a positive value, or a positive result when a positive number is subtracted from a negative number.
For a MUL, if the product is greater than 255 (00FFH), OV will be set. Otherwise, it is cleared.
For a DIV, it is normally 0. However, if B had originally contained 00H, the values returned in A and B will be undefined. Meanwhile, the OV will be set.</td></tr>
<tr><td>1</td><td>F1</td><td>User flag 1
The general purpose flag that can be set or cleared by user via software.</td></tr>
<tr><td>0</td><td>P</td><td>Parity flag
Set to 1 to indicate an odd number of ones in the accumulator. Cleared for an even number of ones. It performs even parity check.</td></tr>
</table>

PCON – Power Control

Register	SFR Address	Reset Value
CWK	87H, All pagess	POR: 0001_000b, other: 000U_0000b

7	6	5	4	3	2	1	0
SMOD	SMOD0	LPR	POF	GF1	GF0	PD	IDL
R/W	R/W	RW	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
3	GF1	**General purpose flag 1** The general purpose flag that can be set or cleared by user via software.
2	GF0	**General purpose flag 0** The general purpose flag that can be set or cleared by user via software.

Table 7.2-1 Instructions That Affect Flag Settings

Instruction	CY	OV	AC	Instruction	CY	OV	AC
ADD	X[1]	X	X	CLR C	0		
ADDC	X	X	X	CPL C	X		
SUBB	X	X	X	ANL C, bit	X		
MUL	0	X		ANL C, /bit	X		
DIV	0	X		ORL C, bit	X		
DA A	X			ORL C, /bit	X		
RRC A	X			MOV C, bit	X		
RLC A	X			CJNE	X		
SETB C	1						

Note: X indicates the modification depends on the result of the instruction.

9 I/O PORT STRUCTURE AND OPERATION

The ML51 has a maximum of 43 general purpose I/O pins which 40 bit-addressable general I/O pins grouped as 5 ports, P0 to P4, and 7 general I/O pins grouped as P5. Each port has its port control register (Px register). The writing and reading of a port control register have different meanings. A write to port control register sets the port output latch logic value, whereas a read gets the port pin logic state. These four modes are quasi-bidirectional (standard 8051 port structure), push-pull, input-only, and open-drain modes. Each port spends two special function registers PxM1 and PxM2 to select the I/O mode of port Px. The list below illustrates how to select the I/O mode of Px.n. Note that the default configuration of is input-only (high-impedance) after any reset.

Table 7.2-1 Configuration for Different I/O Modes

PnM1.X[1]	PnM2.X[1]	I/O Type
0	0	Quasi-bidirectional
0	1	Push-pull
1	0	Input-only (high-impedance)
1	1	Open-drain

[1] N = 0~5, x = 0~7

All I/O pins can be selected as TTL level inputs or Schmitt triggered inputs by selecting corresponding bit in PxS register. Schmitt triggered input has better glitch suppression capability. All I/O pins also have bit-controllable, slew rate select ability via software. The control registers are PxSR. By default, the slew rate is slow. If user would like to increase the I/O output speed, setting the corresponding bit in PxSR, the slew rate is selected in a faster level.

For example:

```
    P0M1 |= 0x40;
    P0M2 &= 0xBF;           //Set P0.6 as input only mode
```

9.1 Quasi-Bidirectional Mode

The quasi-bidirectional mode, as the standard 8051 I/O structure, can rule as both input and output. When the port outputs a logic high, it is weakly driven, allowing an external device to pull the pin low. When the pin is pulled low, it is driven strongly and able to sink a large current. In the quasi-bidirectional I/O structure, there are two pull-high transistors. Each of them serves different purposes. One of these pull-highs, called the “very weak” pull-high, is turned on whenever the port latch contains logic 1. The “very weak” pull-high sources a very small current that will pull the pin high if it is left floating.

The second pull-high is the “strong” pull-high. This pull-high is used to speed up 0-to-1 transitions on a quasi-bidirectional port pin when the port latch changes from logic 0 to logic 1. When this occurs, the strong pull-high turns on for two-CPU-clock time to pull the port pin high quickly. Then it turns off “very weak” pull-highs continue remaining the port pin high. The quasi-bidirectional port structure is shown below.

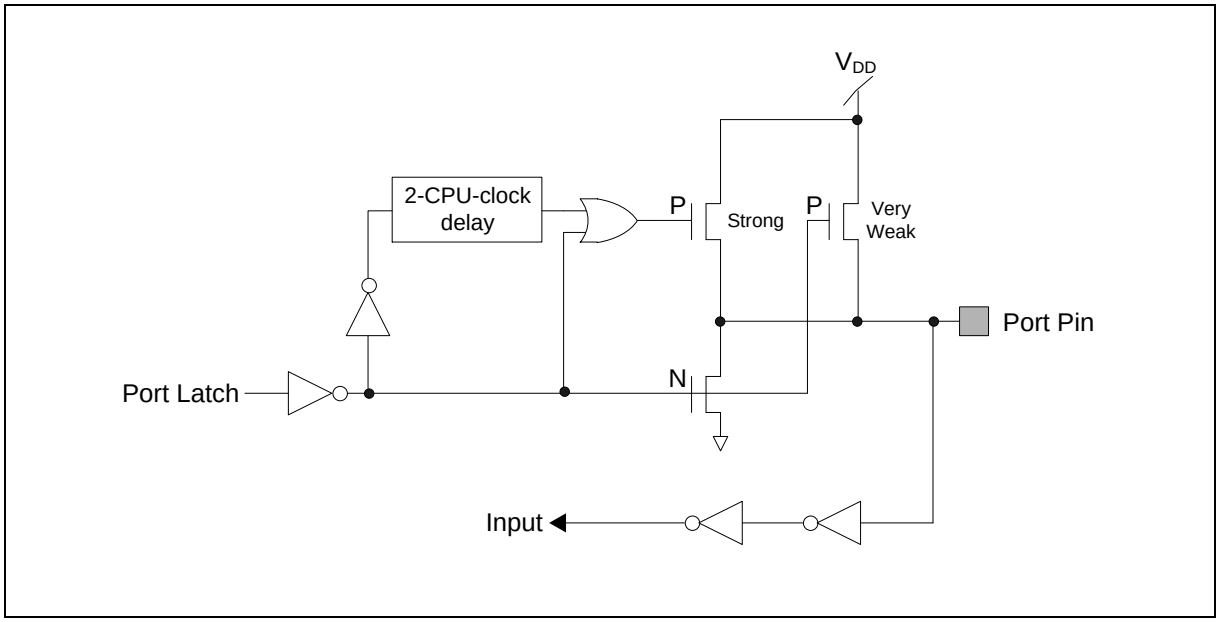


Figure 9.1-1 Quasi-Bidirectional Mode Structure

9.2 Push-Pull Mode

The push-pull mode has the same pull-low structure as the quasi-bidirectional mode, but provides a continuous strong pull-high when the port latch is written by logic 1. The push-pull mode is generally used as output pin when more source current is needed for an output driving.

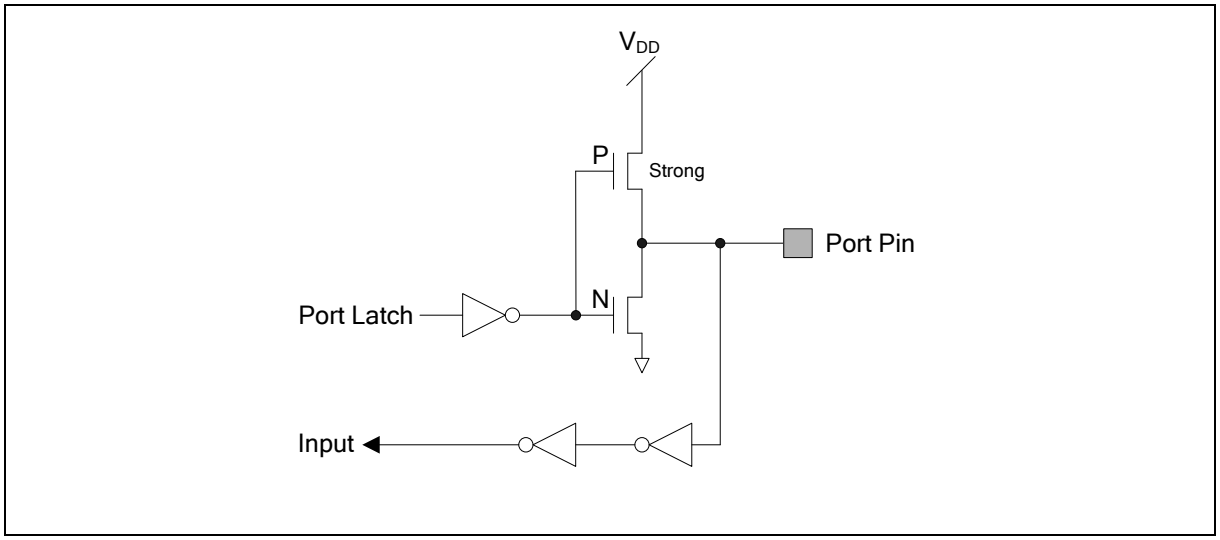


Figure 9.2-1 Push-Pull Mode Structure

9.3 Input-Only Mode

Input-only mode provides true high-impedance input path. Although a quasi-bidirectional mode I/O can also be an input pin, but it requires relative strong input source. Input-only mode also benefits to power consumption reduction for logic 0 input always consumes current from VDD if in quasi-bidirectional mode. User needs to take care that an input-only mode pin should be given with a determined voltage level by external devices or resistors. A floating pin will induce leakage current especially in Power-down mode.

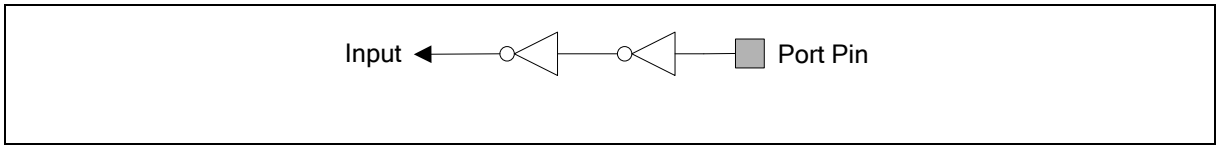


Figure 9.3-1 Input-Only Mode Structure

9.4 Open-Drain Mode

The open-drain mode turns off all pull-high transistors and only drives the pull-low of the port pin when the port latch is given by logic 0. If the port latch is logic 1, it behaves as if in input-only mode. To be used as an output pin generally as I2C lines, an open-drain pin should add an external pull-high, typically a resistor tied to VDD. User needs to take care that an open-drain pin with its port latch as logic 1 should be given with a determined voltage level by external devices or resistors. A floating pin will induce leakage current especially in Power-down mode.

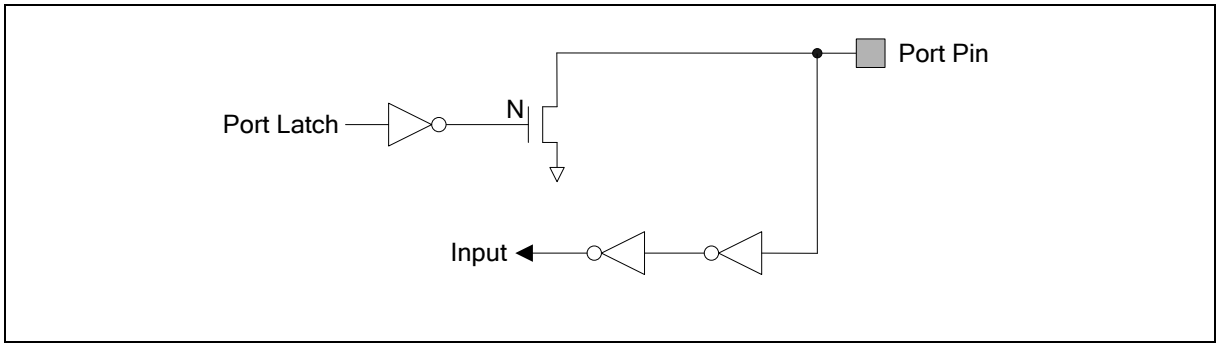


Figure 9.4-1 Open-Drain Mode Structure

9.5 Read-Modify-Write Instructions

Instructions that read a byte from SFR or internal RAM, modify it, and rewrite it back, are called “Read-Modify-Write” instructions. When the destination is an I/O port or a port bit, these instructions read the internal output latch rather than the external pin state. This kind of instructions read the port SFR value, modify it and write back to the port SFR. All “Read-Modify-Write” instructions are listed as follows.

Instruction	Description
ANL	Logical AND. (ANL direct, A and ANL direct, #data)
ORL	Logical OR. (ORL direct, A and ORL direct, #data)
XRL	Logical exclusive OR. (XRL direct, A and XRL direct, #data)
JBC	Jump if bit = 1 and clear it. (JBC bit, rel)
CPL	Complement bit. (CPL bit)
INC	Increment. (INC direct)
DEC	Decrement. (DEC direct)
DJNZ	Decrement and jump if not zero. (DJNZ direct, rel)
MOV	bit, CMove carry to bit. (MOV bit, C)
CLR	bit Clear bit. (CLR bit)
SETB	bit Set bit. (SETB bit)

The last three seem not obviously “Read-Modify-Write” instructions but actually they are. They read the entire port latch value, modify the changed bit, and then write the new value back to the port latch.

9.6 Control Registers of I/O Ports

The ML51 has a lot of I/O control registers to provide flexibility in all kinds of applications. The SFR related with I/O ports can be categorized into four groups: input and output control, output mode control, input type and sink current control, and output slew rate control. All of SFR are listed as follows.

9.6.1 Input and Output Data Control

These registers are I/O input and output data buffers. Reading gets the I/O input data. Writing forces the data output. All of these registers are bit-addressable.

Pn – Port

Register	SFR Address	Reset Value
P0	80H, All pages, Bit-addressable	1111_1111 b
P1	90H, All pages, Bit-addressable	1111_1111 b
P2	A0H, All pages, Bit-addressable	1111_1111 b
P3	B0H, All pages, Bit-addressable	1111_1111 b
P4	D8H, All pages, Bit-addressable	1111_1111 b
P5	B1H, Page 0	0111_1111 b

7	6	5	4	3	2	1	0
Pn.7	Pn.6	Pn.5	Pn.4	Pn.3	Pn.2	Pn.1	Pn.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	Pn[7:0]	Port n Port n is an maximum 8-bit general purpose I/O port.

9.6.2 GPIO Mode Control

These registers control GPIO mode, which is configurable among four modes: input-only, quasi-bidirectional, push-pull, or open-drain. Each pin can be configured individually.

Note[PnM1 and PnM2 [n:0~5] are used in combination to determine the I/O mode of each pin of P0. See Table 7.2-1 Configuration for Different I/O Modes. As default after reset all GPIO setting as input only mode.

PnM1 – Port Mode Select 1 [1]

Register	SFR Address	Reset Value
P0M1	B1H, Page 1	1111_1111 b
P1M1	B3H, Page 1	1111_1111 b
P2M1	B5H, Page 1	1111_1111 b
P3M1	C2H, Page 1	1111_1111 b
P4M1	B9H, Page 1	1111_1111 b
P5M1	BDH, Page 1	1111_1111 b

PnM2 – Port Mode Select 2 [1]

Register	SFR Address	Reset Value
P0M2	B2H, Page 1	0000_0000 b
P1M2	B4H, Page 1	0000_0000 b
P2M2	B6H, Page 1	0000_0000 b
P3M2	C3H, Page 1	0000_0000 b
P4M2	BAH, Page 1	0000_0000 b
P5M2	BEH, Page 1	0000_0000 b

PnM1 – Port n Mode Select 1 [1]

7	6	5	4	3	2	1	0
PnM1.7	PnM1.6	PnM1.5	PnM1.4	PnM1.3	PnM1.2	PnM1.1	PnM1.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnM1[7:0]	Port 0 mode select 1

PnM2 – Port n Mode Select 2 [1]

7	6	5	4	3	2	1	0
PnM2.7	PnM2.6	PnM2.5	PnM2.4	PnM2.3	PnM2.2	PnM2.1	PnM2.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description

Bit	Name	Description
7:0	PnM2[7:0]	Port 0 mode select 2

PnM1.X	PnM2.X	I/O Type
0	0	Quasi-bidirectional
0	1	Push-pull
1	0	Input-only (high-impedance)
1	1	Open-drain

9.6.3 GPIO Multi-Function Select

PnMF10 – Pn.1 and Pn.0 Multi-Function Select

Register	SFR Address	Reset Value
P0MF10	F9H, Page 2	0000_0000 b
P1MF10	FDH, Page 2	0000_0000 b
P2MF10	F2H, Page 2	0000_0000 b
P3MF10	F6H, Page 2	0000_0000 b
P4MF10	EBH, Page 2	0000_0000 b
P5MF10	EFH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PnMF1				PnMF0			
R/W				R/W			

Bit	Name	Description
7:4	PnMF1[7:4]	Pn.1 multi-function select
3:0	PnMF0[3:0]	Pn.0 multi-function select

PnMF32 – Pn.3 and Pn.2 Multi-Function Select

Register	SFR Address	Reset Value
P0MF32	FAH, Page 2	0000_0000 b
P1MF32	FEH, Page 2	0000_0000 b
P2MF32	F3H, Page 2	0000_0000 b
P3MF32	F7H, Page 2	0000_0000 b
P4MF32	ECH, Page 2	0000_0000 b

P5MF32	E1H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PnMF3				PnMF2			
R/W				R/W			

Bit	Name	Description
7:4	PnMF3[7:4]	Pn.3 multi-function select
3:0	PnMF2[3:0]	Pn.2 multi-function select

PnMF54 – Pn.5 and Pn.4 Multi-Function Select

Register	SFR Address	Reset Value
P0MF54	FBH, Page 2	0000_0000 b
P1MF54	FFH, Page 2	0000_0000 b
P2MF54	F4H, Page 2	0000_0000 b
P3MF54	E9H, Page 2	0000_0000 b
P4MF54	EDH, Page 2	0000_0000 b
P5MF54	E2H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PnMF5				PnMF4			
R/W				R/W			

Bit	Name	Description
7:4	PnMF5[7:4]	Pn.5 multi-function select
3:0	PnMF4[3:0]	Pn.4 multi-function select

PnMF76 – Pn.7 and Pn.6 Multi-Function Select

Register	SFR Address	Reset Value
P0MF76	FCH, Page 2	0000_0000 b
P1MF76	F1H, Page 2	0000_0000 b
P2MF76	F5H, Page 2	0000_0000 b
P3MF76	EAH, Page 2	0000_0000 b
P4MF76	EEH, Page 2	0000_0000 b
P5MF76	E3H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PnMF7				PnMF6			
R/W				R/W			

Bit	Name	Description
7:4	PnMF7[7:4]	Pn.7 multi-function select
3:0	PnMF6[3:0]	Pn.6 multi-function select

For example: As list P2.5 can define as following as anyone of the list function. If want to define as PWM0_CH0 output, please setting P2MF54 |= 0xB0.

Pin Name	MFP	Description	
P2.5	MFP0	General purpose digital I/O pin.	
ADC_CH0	MFP1	GPIO	ADC channel 0 analog input.
ACMP0_P0	MFP1	GPIO	Analog comparator 0 positive input 0 pin.
ACMP1_P0	MFP1	GPIO	Analog comparator 1 positive input 0 pin.
I^2C0_SCL	MFP6	GPIO	I^2C0 clock pin.
PWM0_CH0	MFP11	GPIO	PWM0 channel 0 output.
UART2_TXD	MFP13	GPIO	UART2 data transmitter output pin.
SC0_CLK	MFP13	GPIO	Smart Card 0 clock pin.
T0	MFP14	GPIO	Timer0 counter input/toggle output pin.
INT0	MFP15	GPIO	External interrupt 0 input pin.

9.6.4 Input Type

Each I/O pin can be configured individually as TTL input or Schmitt triggered input. Note that all of PxS registers are accessible by switching SFR page to Page 1.

PnS – Port n Schmitt Triggered Input

Register	SFR Address	Reset Value
P0S	99H, Page 1	0000_0000 b
P1S	9BH, Page 1	0000_0000 b
P2S	9DH, Page 1	0000_0000 b
P3S	ACH, Page 1	0000_0000 b
P4S	BBH, Page 1	0000_0000 b
P5S	BFH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnS.7	PnS.6	PnS.5	PnS.4	PnS.3	PnS.2	PnS.1	PnS.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnS[7:0]	P0 Schmitt triggered input 0 = TTL level input of Pn.x. 1 = Schmitt triggered input of Pn.x.

9.6.5 Output Slew Rate Control

Slew rate for each I/O pin is configurable individually. By default, each pin is in normal slew rate mode. User can set each control register bit to enable high-speed slew rate for the corresponding I/O pin. Note that all PxSR registers are accessible by switching SFR page to Page 1.

PnSR –Port n Slew Rate Control

Register	SFR Address	Reset Value
P0SR	9AH, Page 1	0000_0000 b
P1SR	9CH, Page 1	0000_0000 b
P2SR	9EH, Page 1	0000_0000 b
P3SR	ADH, Page 1	0000_0000 b
P4SR	BCH, Page 1	0000_0000 b
P5SR	AEH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnSR.7	PnSR.6	PnSR.5	PnSR.4	PnSR.3	PnSR.2	PnSR.1	PnSR.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnSR[7:0]	P0.n slew rate 0 = Pn.x normal output slew rate. 1 = Pn.x high-speed output slew rate.

9.6.6 Pull-Up Resister Control

Pull up resister for each I/O pin is configurable individually. But even enabled the pull up resister only effect when GPIO setting as input mode. By default, after reset each pin pull high resister is disabled.

PnUP – Port n Pull-Up resister control

Register	SFR Address	Reset Value
P0UP	92H, Page 1	0000_0000 b

P1UP	93H, Page 1	0000_0000 b
P2UP	94H, Page 1	0000_0000 b
P3UP	95H, Page 1	0000_0000 b
P4UP	96H, Page 1	0000_0000 b
P5UP	97H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnUP.7	PnUP.6	PnUP.5	PnUP.4	PnUP.3	PnUP.2	PnUP.1	PnUP.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnUP[7:0]	P0.n pull-up enable 0 = Pn.x pull-up Disabled. 1 = Pn.x pull-up Enabled.

9.6.7 Pull-Down Resister Control

Pull down resister for each I/O pin is configurable individually. Even enabled the pull down resister only effect when GPIO setting as input mode. By default, after reset each pin pull high resister is disabled.

PnDW – Port n Pull-Down resister control

Register	SFR Address	Reset Value
P0DW	8AH, Page 1	0000_0000 b
P1DW	8BH, Page 1	0000_0000 b
P2DW	8CH, Page 1	0000_0000 b
P3DW	8DH, Page 1	0000_0000 b
P4DW	8EH, Page 1	0000_0000 b
P5DW	8FH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PnDW.7	PnDW.6	PnDW.5	PnDW.4	PnDW.3	PnDW.2	PnDW.1	PnDW.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	PnDW[7:0]	P0.n pull-down enable 0 = Pn.x pull-down Disabled. 1 = Pn.x pull-down Enabled.

10 TIMER/COUNTER 0 AND 1

Timer/Counter 0 and 1 on ML51 are two 16-bit Timers/Counters. Each of them has two 8-bit registers those form the 16-bit counting register. For Timer/Counter 0 they are TH0, the upper 8-bit register, and TL0, the lower 8-bit register. Similarly Timer/Counter 1 has two 8-bit registers, TH1 and TL1. TCON and TMOD can configure modes of Timer/Counter 0 and 1.

The Timer or Counter function is selected by the C/$\overline{T}$ bit in TMOD. Each Timer/Counter has its own selection bit. TMOD.2 selects the function for Timer/Counter 0 and TMOD.6 selects the function for Timer/Counter 1

When configured as a “Timer”, the timer counts the system clock cycles. The timer clock is 1/12 of the system clock (FSYS) for standard 8051 capability or direct the system clock for enhancement, which is selected by T0M (CKCON.3) bit for Timer 0 and T1M (CKCON.4) bit for Timer 1. In the “Counter” mode, the countering register increases on the falling edge of the external input pin T0. If the sampled value is high in one clock cycle and low in the next, a valid 1-to-0 transition is recognized on T0 or T1 pin.

The Timers 0 and 1 can be configured to automatically to toggle output whenever a timer overflow occurs. The same device pins that are used for the T0 and T1 count inputs are also used for the timer toggle outputs. This function is enabled by control bits T0OE and T1OE in the CKCON register, and apply to Timer 0 and Timer 1 respectively. The port outputs will be logic 1 prior to the first timer overflow when this mode is turned on. In order for this mode to function, the C/$\overline{T}$ bit should be cleared selecting the system clock as the clock source for the timer.

Note that the TH0 (TH1) and TL0 (TL1) are accessed separately. It is strongly recommended that in mode 0 or 1, user should stop Timer temporally by clearing TR0 (TR1) bit before reading from or writing to TH0 (TH1) and TL0 (TL1). The free-running reading or writing may cause unpredictable result.

10.1 Mode 0 (13-Bit Timer)

In Mode 0, the Timer/Counter is a 13-bit counter. The 13-bit counter consists of TH0 (TH1) and the five lower bits of TL0 (TL1). The upper three bits of TL0 (TL1) are ignored. The Timer/Counter is enabled when TR0 (TR1) is set and either GATE is 0 or $\overline{INT0}$ ($\overline{INT1}$) is 1. Gate setting as 1 allows the Timer to calculate the pulse width on external input pin $\overline{INT0}$ ($\overline{INT1}$). When the 13-bit value moves from 1FFFH to 0000H, the Timer overflow flag TF0 (TF1) is set and an interrupt occurs if enabled.

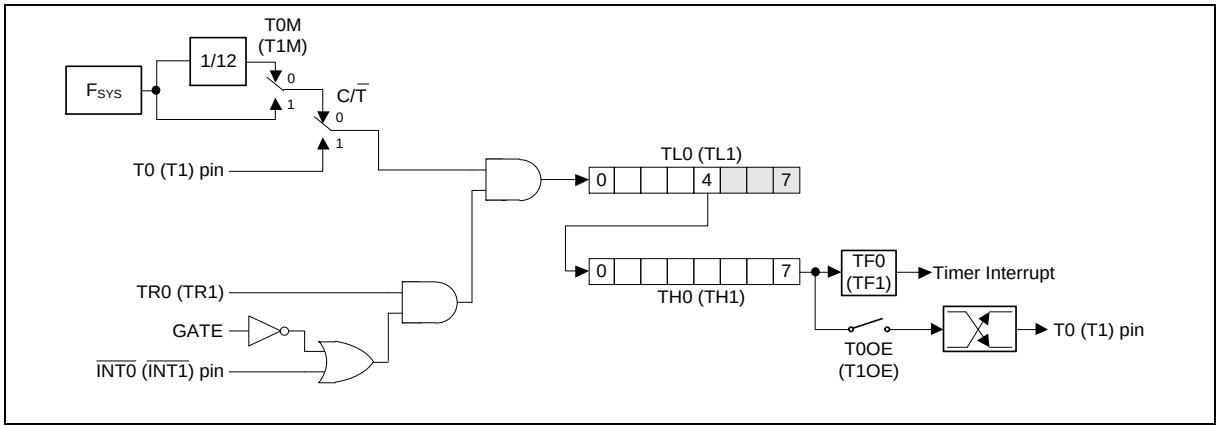


Figure 10.1-1 Timer/Counters 0 and 1 in Mode 0

10.2 Mode 1 (16-Bit Timer)

Mode 1 is similar to Mode 0 except that the counting registers are fully used as a 16-bit counter. Roll-over occurs when a count moves FFFFH to 0000H. The Timer overflow flag TF0 (TF1) of the relevant Timer/Counter is set and an interrupt will occurs if enabled.

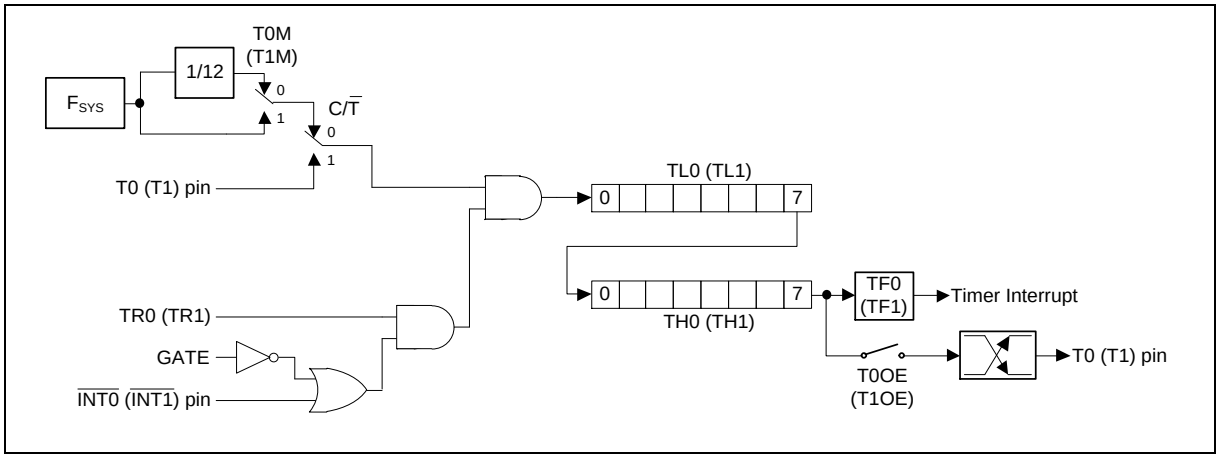


Figure 10.2-1 Timer/Counters 0 and 1 in Mode 1

10.3 Mode 2 (8-Bit Auto-Reload Timer)

In Mode 2, the Timer/Counter is in auto-reload mode. In this mode, TL0 (TL1) acts as an 8-bit count register whereas TH0 (TH1) holds the reload value. When the TL0 (TL1) register overflow, the TF0 (TF1) bit in TCON is set and TL0 (TL1) is reloaded with the contents of TH0 (TH1) and the counting process continues from here. The reload operation leaves the contents of the TH0 (TH1) register unchanged. This feature is best suitable for UART baud rate generator for it runs without continuous software intervention. Note that only Timer1 can be the baud rate source for UART. Counting is enabled by setting the TR0 (TR1) bit as 1 and proper setting of GATE and INT0 (INT1) pins. The functions of GATE and INT0 (INT1) pins are just the same as Mode 0 and 1.

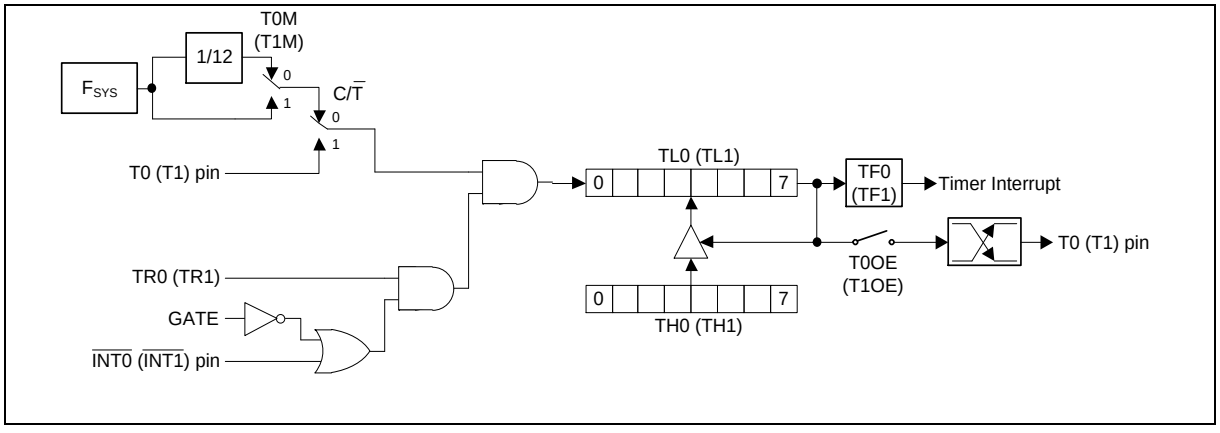


Figure 10.3-1 Timer/Counters 0 and 1 in Mode 2

10.4 Mode 3 (Two Separate 8-Bit Timers)

Mode 3 has different operating methods for Timer 0 and Timer 1. For Timer/Counter 1, Mode 3 simply freezes the counter. Timer/Counter 0, however, configures TL0 and TH0 as two separate 8 bit count registers in this mode. TL0 uses the Timer/Counter 0 control bits C/$\overline{T}$, GATE, TR0, $\overline{INT0}$, and TF0. The TL0 also can be used as a 1-to-0 transition counter on pin T0 as determined by C/$\overline{T}$ (TMOD.2). TH0 is forced as a clock cycle counter and takes over the usage of TR1 and TF1 from Timer/Counter 1. Mode 3 is used in case that an extra 8 bit timer is needed. If Timer/Counter 0 is configured in Mode 3, Timer/Counter 1 can be turned on or off by switching it out of or into its own Mode 3. It can still be used in Modes 0, 1 and 2 although its flexibility is restricted. It no longer has control over its overflow flag TF1 and the enable bit TR1. However Timer 1 can still be used as a Timer/Counter and retains the use of GATE, $\overline{INT1}$ pin and T1M. It can be used as a baud rate generator for the serial port or other application not requiring an interrupt.

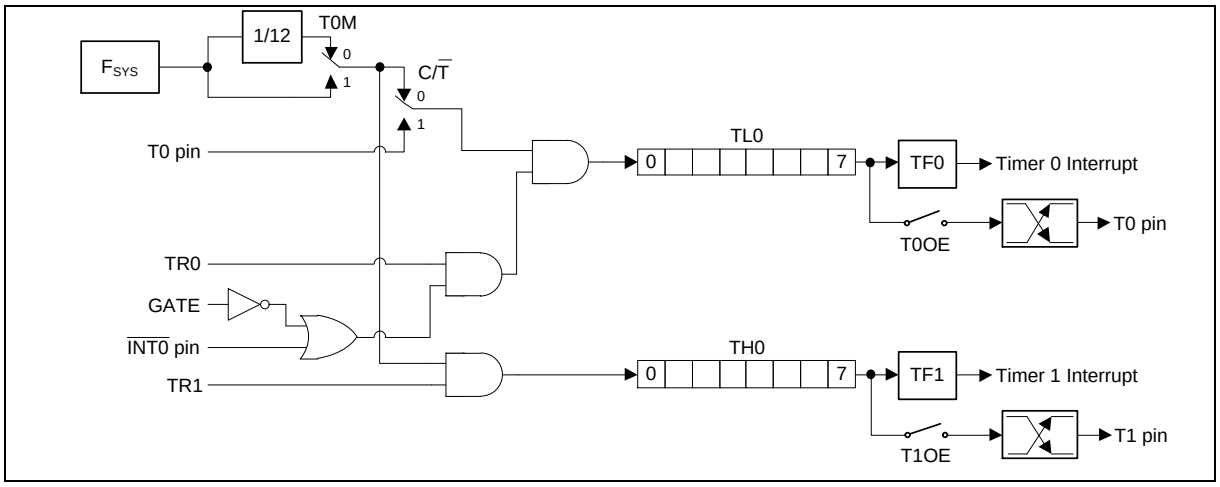


Figure 10.4-1 Timer/Counter 0 in Mode 3

10.5 Control Register of Timer/Counter 0 and 1

TMOD – Timer 0 and 1 Mode

Register	SFR Address	Reset Value
TMOD	89H All pages	0000_0000b

7	6	5	4	3	2	1	0
GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	GATE	**Timer 1 gate control** 0 = Timer 1 will clock when TR1 is 1 regardless of $\overline{INT1}$ logic level. 1 = Timer 1 will clock only when TR1 is 1 and $\overline{INT1}$ is logic 1.
6	C/$\overline{T}$	**Timer 1 Counter/Timer select** 0 = Timer 1 is incremented by internal system clock. 1 = Timer 1 is incremented by the falling edge of the external pin T1.

Bit	Name	Description
5	M1	**Timer 1 mode select** M1 M0 Timer 1 Mode 0 0 Mode 0: 13-bit Timer/Counter 0 1 Mode 1: 16-bit Timer/Counter 1 0 Mode 2: 8-bit Timer/Counter with auto-reload from TH1 1 1 Mode 3: Timer 1 halted
4	M0	
3	GATE	**Timer 0 gate control** 0 = Timer 0 will clock when TR0 is 1 regardless of $\overline{INT0}$ logic level. 1 = Timer 0 will clock only when TR0 is 1 and $\overline{INT0}$ is logic 1.
2	C/$\overline{T}$	**Timer 0 Counter/Timer select** 0 = Timer 0 is incremented by internal system clock. 1 = Timer 0 is incremented by the falling edge of the external pin T0.
1	M1	**Timer 0 mode select** M1 M0 Timer 0 Mode 0 0 Mode 0: 13-bit Timer/Counter 0 1 Mode 1: 16-bit Timer/Counter 1 0 Mode 2: 8-bit Timer/Counter with auto-reload from TH0 1 1 Mode 3: TL0 as a 8-bit Timer/Counter and TH0 as a 8-bit Timer
0	M0	

TCON – Timer 0 and 1 Control

Register	SFR Address	Reset Value
TCON	88H, All pages, Bit-addressable	0000_0000b

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R (level) R/W (edge)	R/W	R (level) R/W (edge)	R/W

Bit	Name	Description
7	TF1	**Timer 1 overflow flag** This bit is set when Timer 1 overflows. It is automatically cleared by hardware when the program executes the Timer 1 interrupt service routine. This bit can be set or cleared by software.
6	TR1	**Timer 1 run control** 0 = Timer 1 Disabled. Clearing this bit will halt Timer 1 and the current count will be preserved in TH1 and TL1. 1 = Timer 1 Enabled.
5	TF0	**Timer 0 overflow flag** This bit is set when Timer 0 overflows. It is automatically cleared via hardware when the program executes the Timer 0 interrupt service routine. This bit can be set or cleared by software.

Bit	Name	Description
4	TR0	**Timer 0 run control** 0 = Timer 0 Disabled. Clearing this bit will halt Timer 0 and the current count will be preserved in TH0 and TL0. 1 = Timer 0 Enabled.
3	IE1	**External interrupt 1 edge flag** If IT1 = 1 (falling edge trigger), this flag will be set by hardware when a falling edge is detected. It remain set until cleared via software or cleared by hardware in the beginning of its interrupt service routine. If IT1 = 0 (low level trigger), this flag follows the inverse of the $\overline{INT1}$ input signal's logic level. Software cannot control it.
2	IT1	**External interrupt 1 type select** This bit selects by which type that $\overline{INT1}$ is triggered. 0 = $\overline{INT1}$ is low level triggered. 1 = $\overline{INT1}$ is falling edge triggered.
1	IE0	**External interrupt 0 edge flag** If IT0 = 1 (falling edge trigger), this flag will be set by hardware when a falling edge is detected. It remain set until cleared via software or cleared by hardware in the beginning of its interrupt service routine. If IT0 = 0 (low level trigger), this flag follows the inverse of the $\overline{INT0}$ input signal's logic level. Software cannot control it.
0	IT0	**External interrupt 0 type select** This bit selects by which type that $\overline{INT0}$ is triggered. 0 = $\overline{INT0}$ is low level triggered. 1 = $\overline{INT0}$ is falling edge triggered.

TL0 – Timer 0 Low Byte

Register	SFR Address	Reset Value
TMOD	8AH All pages	0000_0000b

7	6	5	4	3	2	1	0
TL0[7:0]							
R/W							

Bit	Name	Description
7:0	TL0[7:0]	**Timer 0 low byte** The TL0 register is the low byte of the 16-bit counting register of Timer 0.

TH0 – Timer 0 High Byte

Register	SFR Address	Reset Value
TH0	8CH, Page 0	0000_0000b

7	6	5	4	3	2	1	0
TH0[7:0]							
R/W							

Bit	Name	Description
7:0	TH0[7:0]	**Timer 0 high byte** The TH0 register is the high byte of the 16-bit counting register of Timer 0.

TL1 – Timer 1 Low Byte

Register	SFR Address	Reset Value
TL1	8BH, Page 0	0000_0000b

7	6	5	4	3	2	1	0
TL1[7:0]							
R/W							

Bit	Name	Description
7:0	TL1[7:0]	**Timer 1 low byte** The TL1 register is the low byte of the 16-bit counting register of Timer 1.

TH1 – Timer 1 High Byte

Register	SFR Address	Reset Value
TH1	8DH, Page 0	0000_0000b

7	6	5	4	3	2	1	0
TH1[7:0]							
R/W							

Bit	Name	Description
7:0	TH1[7:0]	**Timer 1 high byte** The TH1 register is the high byte of the 16-bit counting register of Timer 1.

CKCON – Clock Control

Register	SFR Address	Reset Value
TH1	8EH, Page 0	1000_0000b

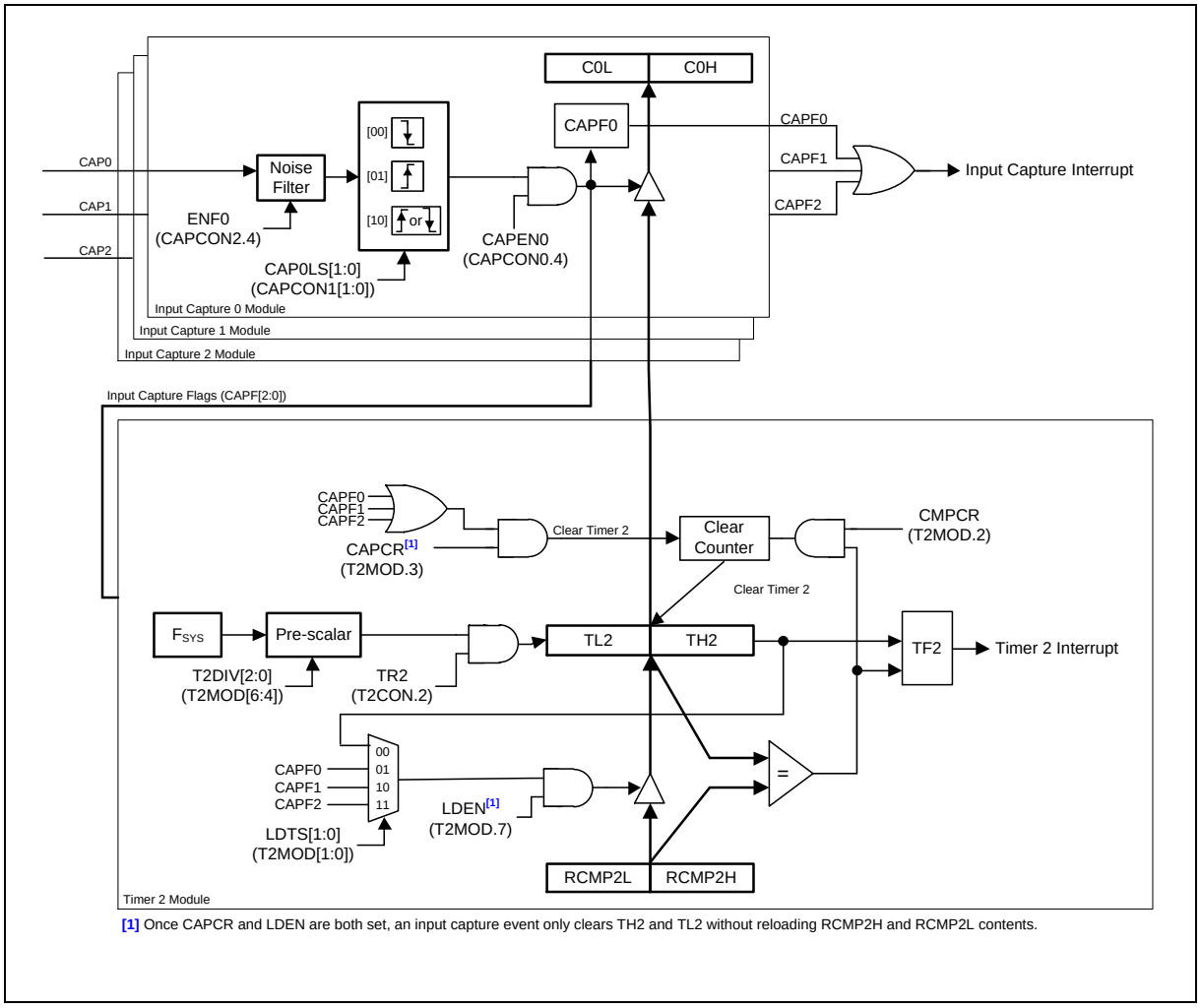
7	6	5	4	3	2	1	0
FASTWK	PWMCKS	T1OE	T1M	T0M	T0OE	CLOEN	-

R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Bit	Name	Description
5	T1OE	**Timer 1 output enable** 0 = Timer 1 output Disabled. 1 = Timer 1 output Enabled from T1 pin. Note that Timer 1 output should be enabled only when operating in its “Timer” mode.
4	T1M	**Timer 1 clock mode select** 0 = The clock source of Timer 1 is the system clock divided by 12. It maintains standard 8051 compatibility. 1 = The clock source of Timer 1 is direct the system clock.
3	T0M	**Timer 0 clock mode select** 0 = The clock source of Timer 0 is the system clock divided by 12. It maintains standard 8051 compatibility. 1 = The clock source of Timer 0 is direct the system clock.
2	T0OE	**Timer 0 output enable** 0 = Timer 0 output Disabled. 1 = Timer 0 output Enabled from T0 pin. Note that Timer 0 output should be enabled only when operating in its “Timer” mode.

11 TIMER 2 AND INPUT CAPTURE

Timer 2 is a 16-bit up counter cascaded with TH2, the upper 8 bits register, and TL2, the lower 8 bit register. Equipped with RCMP2H and RCMP2L, Timer 2 can operate under compare mode and auto-reload mode selected by CM/$\overline{RL2}$ (T2CON.0). An 3-channel input capture module makes Timer 2 detect and measure the width or period of input pulses. The results of 3 input captures are stores in C0H and C0L, C1H and C1L, C2H and C2L individually. The clock source of Timer 2 is from the system clock pre-scaled by a clock divider with 8 different scales for wide field application. The clock is enabled when TR2 (T2CON.2) is 1, and disabled when TR2 is 0. The following registers are related to Timer 2 function.

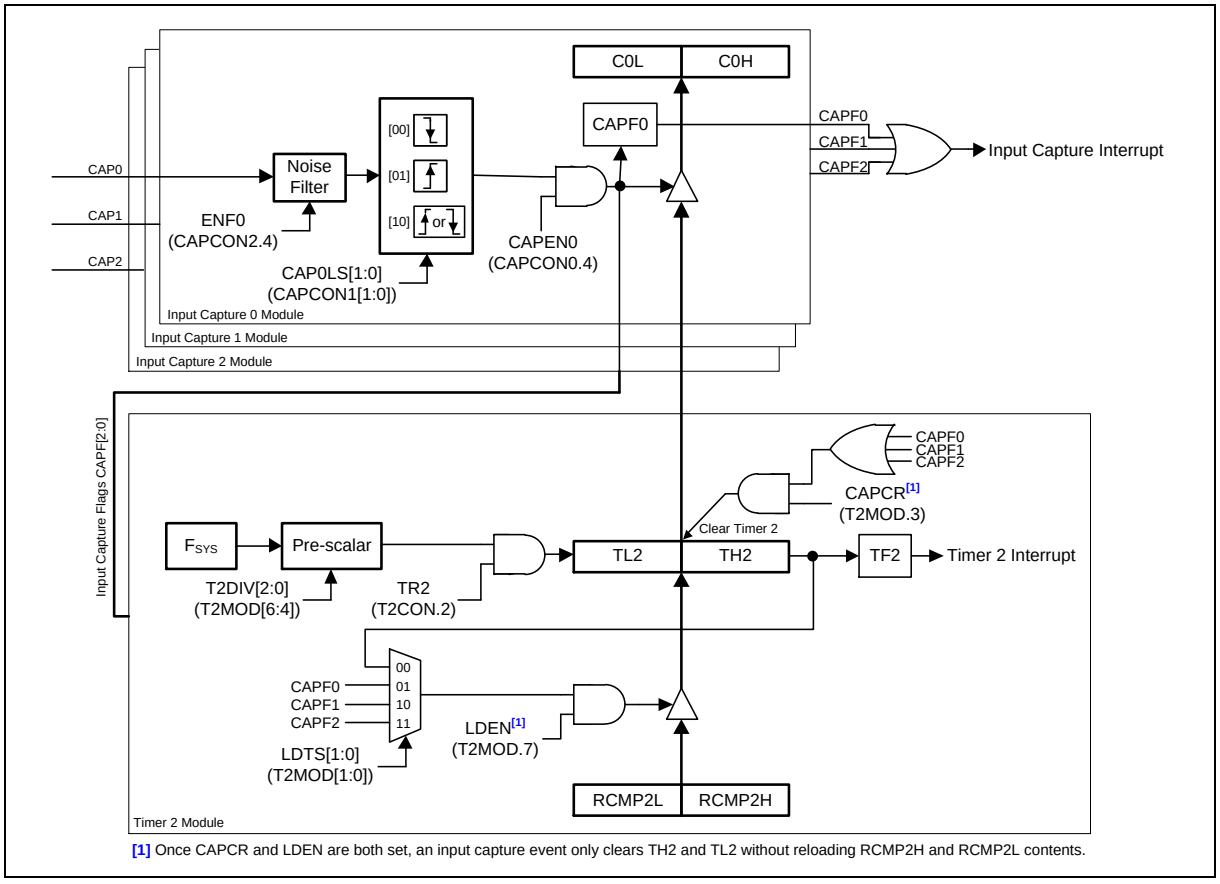


[1] Once CAPCR and LDEN are both set, an input capture event only clears TH2 and TL2 without reloading RCMP2H and RCMP2L contents.

Figure 10.5-1 Timer 2 Block Diagram

11.1 Auto-Reload Mode

The Timer 2 is configured as auto-reload mode by clearing CM/$\overline{RL2}$. In this mode RCMP2H and RCMP2L registers store the reload value. The contents in RCMP2H and RCMP2L transfer into TH2 and TL2 once the auto-reload event occurs if setting LDEN bit. The event can be the Timer 2 overflow or one of the triggering event on any of enabled input capture channel depending on the LDTS[1:0] (T2MOD[1:0]) selection. Note that once CAPCR (T2MOD.3) is set, an input capture event only clears TH2 and TL2 without reloading RCMP2H and RCMP2L contents.



[1] Once CAPCR and LDEN are both set, an input capture event only clears TH2 and TL2 without reloading RCMP2H and RCMP2L contents.

Figure 11.1-1 Timer 2 Auto-Reload Mode and Input Capture Module Functional Block Diagram

11.2 Compare Mode

Timer 2 can also be configured as the compare mode by setting CM/$\overline{RL2}$. In this mode RCMP2H and RCMP2L registers serve as the compare value registers. As Timer 2 up counting, TH2 and TL2 match RCMP2H and RCMP2L, TF2 (T2CON.7) will be set by hardware to indicate a compare match event.

Setting CMPCR (T2MOD.2) makes the hardware to clear Timer 2 counter as 0000H automatically after a compare match has occurred.

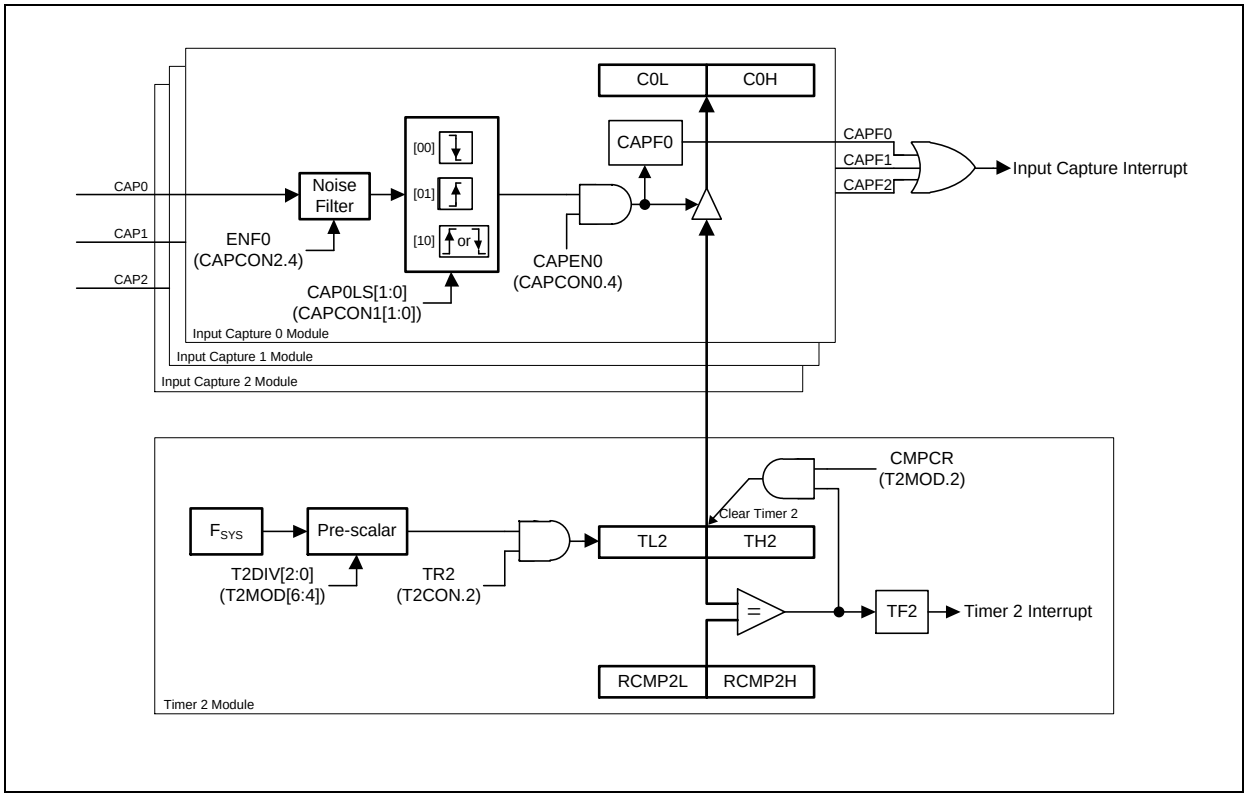


Figure 11.2-1 Timer 2 Compare Mode and Input Capture Module Functional Block Diagram

11.3 Input Capture Module

The input capture module along with Timer 2 implements the input capture function. The input capture module is configured through CAPCON0~2 registers. The input capture module supports 3-channel inputs (CAP0, CAP1, and CAP2). Each input channel consists its own noise filter, which is enabled via setting ENF0~2 (CAPCON2[6:4]). It filters input glitches smaller than four system clock cycles. Input capture channels has their own independent edge detector but share the unique Timer 2. Each trigger edge detector is selected individually by setting corresponding bits in CAPCON1. It supports positive edge capture, negative edge capture, or any edge capture. Each input capture channel has to set its own enabling bit CAPEN0~2 (CAPCON0[6:4]) before use.

While input capture channel is enabled and the selected edge trigger occurs, the content of the free running Timer 2 counter, TH2 and TL2, will be captured, transferred, and stored into the capture registers CnH and CnL. The edge triggering also causes CAPFn (CAPCON0.n) set by hardware. The interrupt will also generate if the ECAP (EIE0.2) and EA bit are both set. For three input capture flags share the same interrupt vector, user should check CAPFn to confirm which channel comes the input capture edge. These flags should be cleared by software.

The bit CAPCR (CAPCON2.3) benefits the implement of period calculation. Setting CAPCR makes the hardware clear Timer 2 as 0000H automatically after the value of TH2 and TL2 have been captured after an input capture edge event occurs. It eliminates the routine software overhead of writing 16-bit counter or an arithmetic subtraction.

Following is the Timer2 input capture multi function pin define.

Group	Pin Name	GPIO	MFP	Type	Description
IC0	IC0	P1.3	MFP13	I/O	Input Capture channel 0
		P1.0	MFP13	I/O	
		P3.3	MFP13	I/O	
IC1	IC1	P1.2	MFP13	I/O	Input Capture channel 1
		P3.2	MFP13	I/O	
IC2	IC2	P1.1	MFP13	I/O	Input Capture channel 2
		P3.1	MFP13	I/O	

11.4 Control Registers of Timer2

T2CON – Timer 2 Control

Register	SFR Address	Reset Value
T2CON	C8H, All pages	0000_0000 b

7	6	5	4	3	2	1	0
TF2	-	-	-	-	TR2	-	CM/$\overline{RL2}$
R/W	-	-	-	-	R/W	-	R/W

Bit	Name	Description
7	TF2	**Timer 2 overflow flag** This bit is set when Timer 2 overflows or a compare match occurs. If the Timer 2 interrupt and the global interrupt are enable, setting this bit will make CPU execute Timer 2 interrupt service routine. This bit is not automatically cleared via hardware and should be cleared via software.
5:3	-	Reserved
2	TR2	**Timer 2 run control** 0 = Timer 2 Disabled. Clearing this bit will halt Timer 2 and the current count will be preserved in TH2 and TL2. 1 = Timer 2 Enabled.
1	-	Reserved
0	CM/$\overline{RL2}$	**Timer 2 compare or auto-reload mode select** This bit selects Timer 2 functioning mode. 0 = Auto-reload mode. 1 = Compare mode.

T2MOD – Timer 2 Mode

Register	SFR Address	Reset Value
T2MOD	C9H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
LDEN	T2DIV[2:0]			CAPCR	CMPCR	LDTS[1:0]	
R/W	R/W			R/W	R/W	R/W	

Bit	Name	Description
7	LDEN	**Enable auto-reload** 0 = Reloading RCMP2H and RCMP2L to TH2 and TL2 Disabled. 1 = Reloading RCMP2H and RCMP2L to TH2 and TL2 Enabled.

Bit	Name	Description
6:4	T2DIV[2:0]	**Timer 2 clock divider** 000 = Timer 2 clock divider is 1/1. 001 = Timer 2 clock divider is 1/4. 010 = Timer 2 clock divider is 1/16. 011 = Timer 2 clock divider is 1/32. 100 = Timer 2 clock divider is 1/64. 101 = Timer 2 clock divider is 1/128. 110 = Timer 2 clock divider is 1/256. 111 = Timer 2 clock divider is 1/512.
3	CAPCR	**Capture auto-clear** This bit is valid only under Timer 2 auto-reload mode. It enables hardware auto-clearing TH2 and TL2 counter registers after they have been transferred in to RCMP2H and RCMP2L while a capture event occurs. 0 = Timer 2 continues counting when a capture event occurs. 1 = Timer 2 value is auto-cleared as 0000H when a capture event occurs.
2	CMPCR	**Compare match auto-clear** This bit is valid only under Timer 2 compare mode. It enables hardware auto-clearing TH2 and TL2 counter registers after a compare match occurs. 0 = Timer 2 continues counting when a compare match occurs. 1 = Timer 2 value is auto-cleared as 0000H when a compare match occurs.
1:0	LDTS[1:0]	**Auto-reload trigger select** These bits select the reload trigger event. 00 = Reload when Timer 2 overflows. 01 = Reload when input capture 0 event occurs. 10 = Reload when input capture 1 event occurs. 11 = Reload when input capture 2 event occurs.

RCMP2L– Timer 2 Reload/Compare Low Byte

Register	SFR Address	Reset Value
RCMP2L	CAH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
RCMP2L[7:0]							
R/W							

Bit	Name	Description
7:0	RCMP2L[7:0]	**Timer 2 reload/compare low byte** This register stores the low byte of compare value when Timer 2 is configured in compare mode. Also it holds the low byte of the reload value in auto-reload mode.

RCMP2H – Timer 2 Reload/Compare High Byte

Register	SFR Address	Reset Value
RCMP2H	CBH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
RCMP2H[7:0]							
R/W							

Bit	Name	Description
7:0	RCMP2H[7:0]	**Timer 2 reload/compare high byte** This register stores the high byte of compare value when Timer 2 is configured in compare mode. Also it holds the high byte of the reload value in auto-reload mode.

TL2 – Timer 2 Low Byte

Register	SFR Address	Reset Value
TL2	CCH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
TL2[7:0]							
R/W							

Bit	Name	Description
7:0	TL2[7:0]	**Timer 2 low byte** The TL2 register is the low byte of the 16-bit counting register of Timer 2.

TH2 – Timer 2 High Byte

Register	SFR Address	Reset Value
TH2	CDH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
TH2[7:0]							
R/W							

Bit	Name	Description
7:0	TH2[7:0]	**Timer 2 high byte** The TH2 register is the high byte of the 16-bit counting register of Timer 2.

Note: that the TH2 and TL2 are accessed separately. It is strongly recommended that user stops Timer 2 temporally by clearing TR2 bit before reading from or writing to TH2 and TL2. The free-running reading or writing may cause unpredictable result.

CAPCON0 – Input Capture Control 0

Register	SFR Address	Reset Value
CAPCON0	E1H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	CAPEN2	CAPEN1	CAPEN0	-	CAPF2	CAPF1	CAPF0
-	R/W	R/W	R/W	-	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	CAPEN2	Input capture 2 enable 0 = Input capture channel 2 Disabled. 1 = Input capture channel 2 Enabled.
5	CAPEN1	Input capture 1 enable 0 = Input capture channel 1 Disabled. 1 = Input capture channel 1 Enabled.
4	CAPEN0	Input capture 0 enable 0 = Input capture channel 0 Disabled. 1 = Input capture channel 0 Enabled.
3	-	Reserved
2	CAPF2	Input capture 2 flag This bit is set by hardware if the determined edge of input capture 2 occurs. This bit should cleared by software.
1	CAPF1	Input capture 1 flag This bit is set by hardware if the determined edge of input capture 1 occurs. This bit should cleared by software.
0	CAPF0	Input capture 0 flag This bit is set by hardware if the determined edge of input capture 0 occurs. This bit should cleared by software.

CAPCON1 – Input Capture Control 1

Register	SFR Address	Reset Value
CAPCON1	E2H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	-	CAP2LS[1:0]		CAP1LS[1:0]		CAP0LS[1:0]	
-	-	R/W		R/W		R/W	

Bit	Name	Description
7:6	-	Reserved
5:4	CAP2LS[1:0]	Input capture 2 level select 00 = Falling edge. 01 = Rising edge. 10 = Either rising or falling edge. 11 = Reserved.
3:2	CAP1LS[1:0]	Input capture 1 level select 00 = Falling edge. 01 = Rising edge. 10 = Either rising or falling edge. 11 = Reserved.
1:0	CAP0LS[1:0]	Input capture 0 level select 00 = Falling edge. 01 = Rising edge. 10 = Either rising or falling edge. 11 = Reserved.

CAPCON2 – Input Capture Control 2

Register	SFR Address	Reset Value
ADCDLY	E3H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	ENF2	ENF1	ENF0	-	-	-	-
-	R/W	R/W	R/W	-	-	-	-

Bit	Name	Description
7	-	Reserved
6	ENF2	Enable noise filer on input capture 2 0 = Noise filter on input capture channel 2 Disabled. 1 = Noise filter on input capture channel 2 Enabled.
5	ENF1	Enable noise filer on input capture 1 0 = Noise filter on input capture channel 1 Disabled. 1 = Noise filter on input capture channel 1 Enabled.
4	ENF0	Enable noise filer on input capture 0 0 = Noise filter on input capture channel 0 Disabled. 1 = Noise filter on input capture channel 0 Enabled.
3:0	-	Reserved

C0L – Capture 0 Low Byte

Register	SFR Address	Reset Value
C0L	E4H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C0L[7:0]							
R/W							

Bit	Name	Description
7:0	C0L[7:0]	Input capture 0 result low byte The C0L register is the low byte of the 16-bit result captured by input capture 0.

C0H – Capture 0 High Byte

Register	SFR Address	Reset Value
C0H	E5H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C0H[7:0]							
R/W							

Bit	Name	Description
7:0	C0H[7:0]	Input capture 0 result high byte The C0H register is the high byte of the 16-bit result captured by input capture 0.

C1L – Capture 1 Low Byte

Register	SFR Address	Reset Value
C1L	E6H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C1L[7:0]							
R/W							

Bit	Name	Description
7:0	C1L[7:0]	Input capture 1 result low byte The C1L register is the low byte of the 16-bit result captured by input capture 1.

C1H – Capture 1 High Byte

Register	SFR Address	Reset Value
C1H	E7H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C1H[7:0]							
R/W							

Bit	Name	Description
7:0	C1H[7:0]	Input capture 1 result high byte The C1H register is the high byte of the 16-bit result captured by input capture 1.

C2L – Capture 2 Low Byte

Register	SFR Address	Reset Value
C2L	EDH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C2L[7:0]							
R/W							

Bit	Name	Description
7:0	C2L[7:0]	Input capture 2 result low byte The C2L register is the low byte of the 16-bit result captured by input capture 2.

C2H – Capture 2 High Byte

Register	SFR Address	Reset Value
C2H	EEH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
C2H[7:0]							
R/W							

Bit	Name	Description
7:0	C2H[7:0]	Input capture 2 result high byte The C2H register is the high byte of the 16-bit result captured by input capture 2.

12 TIMER 3

Timer 3 is implemented simply as a 16-bit auto-reload, up-counting timer. The user can select the pre-scale with T3PS[2:0] (T3CON[2:0]) and fill the reload value into RH3 and RL3 registers to determine its overflow rate. User then can set TR3 (T3CON.3) to start counting. When the counter rolls over FFFFH, TF3 (T3CON.4) is set as 1 and a reload is generated and causes the contents of the RH3 and RL3 registers to be reloaded into the internal 16-bit counter. If ET3 (EIE1.1) is set as 1, Timer 3 interrupt service routine will be served. TF3 is auto-cleared by hardware after entering its interrupt service routine.

Timer 3 can also be the baud rate clock source of both UARTs. For details, please see Section 15.1.2“Baud Rate”.

12.1 Timer 3 Block Diagram

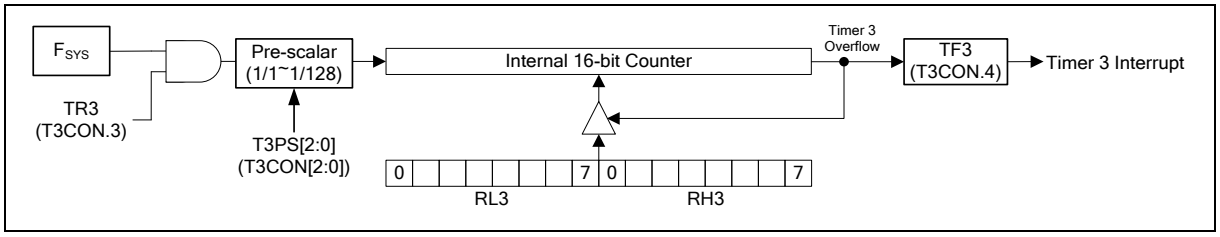


Figure 12.1-1 Timer 3 Block Diagram

12.2 Control Register Of Timer 3

T3CON – Timer 3 Control

Register	SFR Address	Reset Value
T3CON	C4H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SMOD_1	SMOD0_1	BRCK	TF3	TR3	T3PS[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W		

Bit	Name	Description
4	TF3	Timer 3 overflow flag This bit is set when Timer 3 overflows. It is automatically cleared by hardware when the program executes the Timer 3 interrupt service routine. This bit can be set or cleared by software.
3	TR3	Timer 3 run control 0 = Timer 3 is halted. 1 = Timer 3 starts running. Note that the reload registers RH3 and RL3 can only be written when Timer 3 is halted (TR3 bit is 0). If any of RH3 or RL3 is written if TR3 is 1, result is unpredictable.

Bit	Name	Description
2:0	T3PS[2:0]	Timer 3 pre-scalar These bits determine the scale of the clock divider for Timer 3. 000 = 1/1. 001 = 1/2. 010 = 1/4. 011 = 1/8. 100 = 1/16. 101 = 1/32. 110 = 1/64. 111 = 1/128.

RL3 – Timer 3 Reload Low Byte

Register	SFR Address	Reset Value
RL3	C5H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
RL3[7:0]							
R/W							

Bit	Name	Description
7:0	RL3[7:0]	**Timer 3 reload low byte** It holds the low byte of the reload value of Timer 3.

RH3 – Timer 3 Reload High Byte

Register	SFR Address	Reset Value
RH3	C6H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
RH3[7:0]							
R/W							

Bit	Name	Description
7:0	RH3[7:0]	**Timer 3 reload high byte** It holds the high byte of the reload value of Time 3.

13 WATCHDOG TIMER (WDT)

The ML51 provides one Watchdog Timer (WDT). It can be configured as a time-out reset timer to reset whole device. Once the device runs in an abnormal status or hangs up by outward interference, a WDT reset recover the system. It provides a system monitor, which improves the reliability of the system. Therefore, WDT is especially useful for system that is susceptible to noise, power glitches, or electrostatic discharge. The WDT also can be configured as a general purpose timer, of which the periodic interrupt serves as an event timer or a durational system supervisor in a monitoring system, which is able to operate during Idle or Power-down mode. WDTEN[3:0] (CONFIG4[7:4]) initialize the WDT to operate as a time-out reset timer or a general purpose timer.

The Watchdog time-out interval is determined by the formula $\frac{1}{F_{LIRC} \times clockdividerscalar} \times 64$, where FLIRC is the frequency of internal 38.4 kHz oscillator. The following table shows an example of the Watchdog time-out interval with different pre-scales.

Table 12.2-1 Watchdog Timer-out Interval Under Different Pre-scalars

WDPS.2	WDPS.1	WDPS.0	Clock Divider Scale	WDT Time-Out Timing[1]
0	0	0	1/1	1.66 ms
0	0	1	1/4	6.64 ms
0	1	0	1/8	13.31 ms
0	1	1	1/16	26.62 ms
1	0	0	1/32	53.25 ms
1	0	1	1/64	106.66 ms
1	1	0	1/128	213.12 ms
1	1	1	1/256	426.64 ms

Note: This is an approximate value since the deviation of LIRC.

Since the limitation of the maxima vaule of WDT timer delay. To up ML51 from idle mode or power down mode suggest use WKT function see Chapter14 SELF WAKE-UP TIMER (WKT).

13.1 Time-Out Reset Timer

When the CONFIG bits WDTEN[3:0] (CONFIG4[7:4]) is not FH, the WDT is initialized as a time-out reset timer. If WDTEN[3:0] is not 5H, the WDT is allowed to continue running after the system enters Idle or Power-down mode. Note that when WDT is initialized as a time-out reset timer, WDTR and WIDPD has no function.

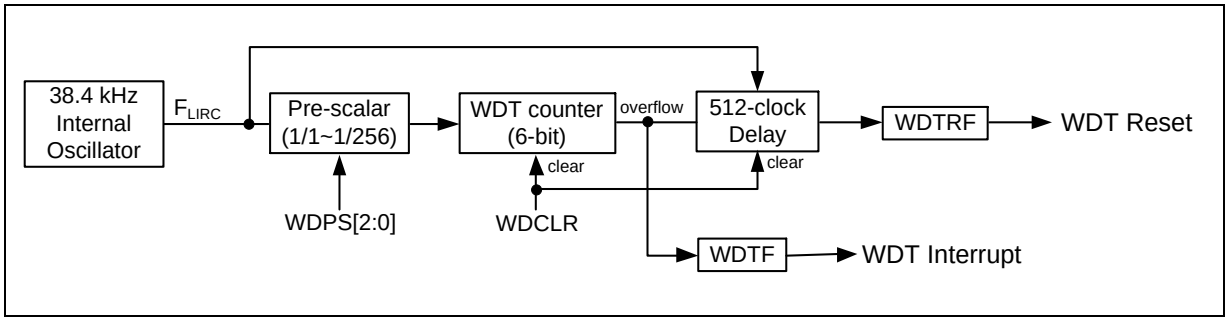


Figure 13.1-1 WDT as A Time-Out Reset Timer

After the device is powered and it starts to execute software code, the WDT starts counting simultaneously. The time-out interval is selected by the three bits WDPS[2:0] (WDCON[2:0]). When the selected time-out occurs, the WDT will set the interrupt flag WDTF (WDCON.5). If the WDT interrupt enable bit EWDT (EIE0.4) and global interrupt enable EA are both set, the WDT interrupt routine will be executed. Meanwhile, an additional 512 clocks of the low-speed internal oscillator delays to expect a counter clearing by setting WDCLR to avoid the system reset by WDT if the device operates normally. If no counter reset by writing 1 to WDCLR during this 512-clock period, a WDT reset will happen. Setting WDCLR bit is used to clear the counter of the WDT. This bit is self-cleared for user monitoring it. Once a reset due to WDT occurs, the WDT reset flag WDTRF (WDCON.3) will be set. This bit keeps unchanged after any reset other than a power-on reset. User may clear WDTRF via software. Note that all bits in WDCON require timed access writing.

The main application of the WDT with time-out reset enabling is for the system monitor. This is important in real-time control applications. In case of some power glitches or electro-magnetic interference, CPU may begin to execute erroneous codes and operate in an unpredictable state. If this is left unchecked the entire system may crash. Using the WDT during software development requires user to select proper “Feeding Dog” time by clearing the WDT counter. By inserting the instruction of setting WDCLR, it allows the code to run without any WDT reset. However If any erroneous code executes by any interference, the instructions to clear the WDT counter will not be executed at the required instants. Thus the WDT reset will occur to reset the system state from an erroneously executing condition and recover the system.

13.2 General Purpose Timer

There is another application of the WDT, which is used as a simple, long period timer. When the CONFIG bits WDTEN[3:0] (CONFIG4[7:4]) is FH, the WDT is initialized as a general purpose timer. In this mode, WDTR and WIDPD are fully accessed via software.

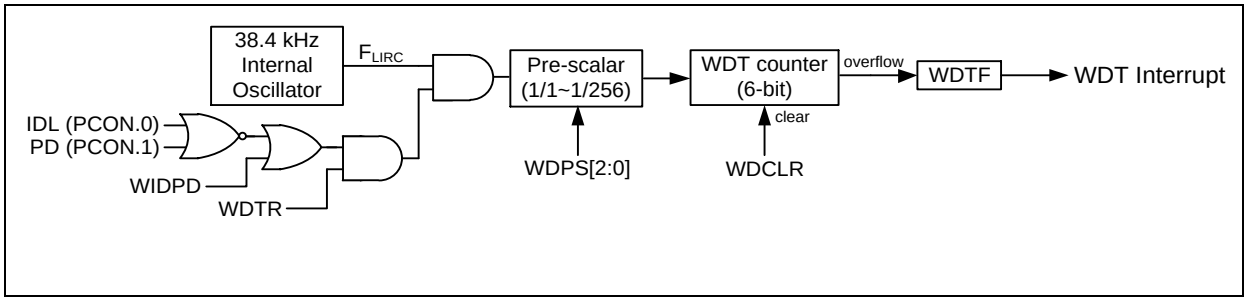


Figure 13.2-1 Watchdog Timer Block Diagram

The WDT starts running by setting WDTR as 1 and halts by clearing WDTR as 0. The WDTF flag will be set while the WDT completes the selected time interval. The software polls the WDTF flag to detect a time-out. An interrupt will occur if the individual interrupt EWDT (EIE0.4) and global interrupt enable EA is set. WDT will continue counting. User should clear WDTF and wait for the next overflow by polling WDTF flag or waiting for the interrupt occurrence.

In some application of low power consumption, the CPU usually stays in Idle mode when nothing needs to be served to save power consumption. After a while the CPU will be woken up to check if anything needs to be served at an interval of programmed period implemented by Timer 0~3. However, the current consumption of Idle mode still keeps at a “mA” level. To further reducing the current consumption to “uA” level, the CPU should stay in Power-down mode when nothing needs to be served, and has the ability of waking up at a programmable interval. The ML51 is equipped with this useful function by WDT waking up. It provides a very low power internal oscillator 38.4 kHz as the clock source of the WDT. It is also able to count under Power-down mode and wake CPU up. The demo code to accomplish this feature is shown below.

13.3 Control Register Of WDT

CONFIG4

7	6	5	4	3	2	1	0
WDTEN[3:0]				-	-	-	-
R/W				-	-	-	-

Factory default value: 1111 1111b

Bit	Name	Description
7:4	WDTEN[3:0]	WDT enable This field configures the WDT behavior after MCU execution. 1111 = WDT is Disabled. WDT can be used as a general purpose timer via software control. 0101 = WDT is Enabled as a time-out reset timer and it stops running during Idle or Power-down mode. Others = WDT is Enabled as a time-out reset timer and it keeps running during Idle or Power-down mode.

The WDT is implemented with a set of divider that divides the low-speed internal oscillator clock nominal 38.4 kHz. The divider output is selectable and determines the time-out interval. When the time-out interval is fulfilled, it will wake the system up from Idle or Power-down mode and an interrupt event will occur if WDT interrupt is enabled. If WDT is initialized as a time-out reset timer, a system reset will occur after a period of delay if without any software action.

WDCON – Watchdog Timer Control

Register	SFR Address	Reset Value
WDCON	AAH, Page 0, TA protected	POR: 0000 0111b WDT: 0000 1UUUb Others: 0000 UUUUb

7	6	5	4	3	2	1	0
WDTR	WDCLR	WDTF	WIDPD	WDTRF	WDPS[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W		

Bit	Name	Description

Bit	Name	Description
7	WDTR	**WDT run** This bit is valid only when control bits in WDTEN[3:0] (CONFIG4[7:4]) are all 1. At this time, WDT works as a general purpose timer. 0 = WDT Disabled. 1 = WDT Enabled. The WDT counter starts running.
6	WDCLR	**WDT clear** Setting this bit will reset the WDT count to 00H. It puts the counter in a known state and prohibit the system from unpredictable reset. The meaning of writing and reading WDCLR bit is different. Writing: 0 = No effect. 1 = Clearing WDT counter. Reading: 0 = WDT counter is completely cleared. 1 = WDT counter is not yet cleared.
5	WDTF	**WDT time-out flag** This bit indicates an overflow of WDT counter. This flag should be cleared by software.
4	WIDPD	**WDT running in Idle or Power-down mode** This bit is valid only when control bits in WDTEN[3:0] (CONFIG4[7:4]) are all 1. It decides whether WDT runs in Idle or Power-down mode when WDT works as a general purpose timer. 0 = WDT stops running during Idle or Power-down mode. 1 = WDT keeps running during Idle or Power-down mode.
3	WDTRF	**WDT reset flag** When the CPU is reset by WDT time-out event, this bit will be set via hardware. This flag is recommended to be cleared via software after reset.
2:0	WDPS[2:0]	**WDT clock pre-scalar select** These bits determine the pre-scale of WDT clock from 1/1 through 1/256. See Table 12.2-1 The default is the maximum pre-scale value.

Note:
[1] WDTRF will be cleared after power-on reset, be set after WDT reset, and remains unchanged after any other resets.
[2] WDPS[2:0] are all set after power-on reset and keep unchanged after any reset other than power-on reset.

```
      ORG   0000H
      LJMP  START

      ORG   0053H
      LJMP  WDT_ISR

      ORG   0100H
     ;*********************************************************************
     ;WDT interrupt service routine
     ;*********************************************************************
     WDT_ISR:
```

```
    CLR   EA
    MOV   TA,#0AAH
    MOV   TA,#55H
    ANL   WDCON,#11011111B ;clear WDT interrupt flag
    SETB  EA
    RETI
;*******************************************************************
;Start here
;*******************************************************************
START:
    MOV   TA,#0AAH
    MOV   TA,#55H
    ORL   WDCON,#00010111B ;choose interval length and enable WDT
                           ;running during Power-down
    SETB  EWDT             ;enable WDT interrupt
    SETB  EA

    MOV   TA,#0AAH
    MOV   TA,#55H
    ORL   WDCON,#10000000B ; WDT run
;*******************************************************************
;Enter Power-down mode
;*******************************************************************
LOOP:
    ORL   PCON,#02H
    LJMP  LOOP
```

14 SELF WAKE-UP TIMER (WKT)

The ML51 has a dedicated Self Wake-up Timer (WKT), which serves for a periodic wake-up timer in low power mode or for general purpose timer. WKT remains counting in Idle or Power-down mode. When WKT is being used as a wake-up timer, a start of WKT can occur just prior to entering a power management mode. WKT has two clock source, internal LIRC 38.4 kHz or LXT 32.768 kHz. Note that the system clock frequency must be twice over WKT clock. If WKT starts counting, the selected clock source will remain active once the device enters Idle or Power-down mode. Note that the selected clock source of WKT will not automatically enabled along with WKT configuration. User should manually enable the selected clock source and waiting for stability to ensure a proper operation.

The WKT is implemented simply as a 8-bit auto-reload, up-counting timer with pre-scale 1/1 to 1/2048 selected by WKPS[2:0] (WKCON[2:0]). User fills the reload value into RWK register to determine its overflow rate. The RWK can reloadable when counter is count to overflow. The CWK can read current count value. The WKTR (WKCON.3) can be set to start counting. When the counter rolls over FFH, WKTF (WKCON.4) is set as 1 and a reload is generated and causes the contents of the RWK register to be reloaded into the internal 8-bit counter. If EWKT (EIE1.2) is set as 1, WKT interrupt service routine will be served.

14.1 WKT Block Diagram

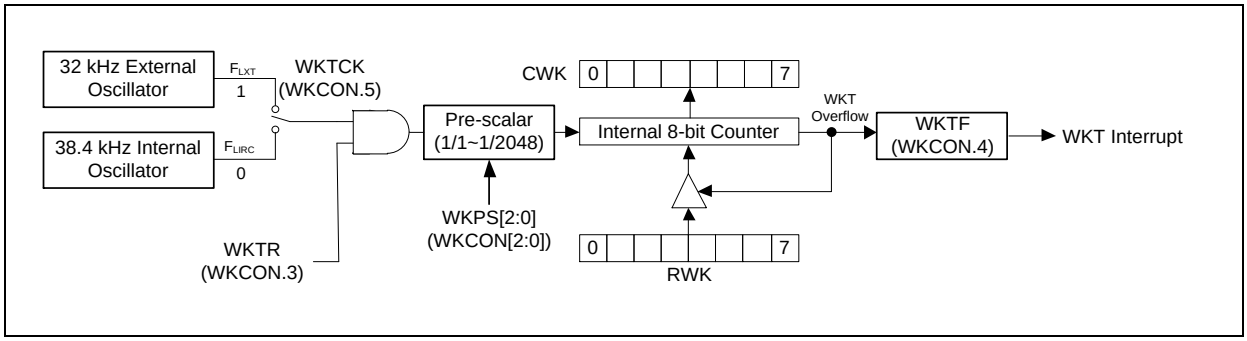


Figure 14.1-1 Self Wake-Up Timer Block Diagram

14.2 Control Register Of WKT

WKCON – Self Wake-up Timer Control

Register	SFR Address	Reset Value
WKCON	8FH, PAGE 0	0000_0000b

7	6	5	4	3	2	1	0
-	-	WKTCK	WKTF	WKTR	WKPS[2:0]		
-	-	R/W	R/W	R/W	R/W		

Bit	Name	Description
7:6	-	Reserved
	WKTCK	**WKT clock source** This bit is set WKT clock source select bit. 0 = LIRC 1 = LXT

Bit	Name	Description
4	WKTF	**WKT overflow flag** This bit is set when WKT overflows. If the WKT interrupt and the global interrupt are enabled, setting this bit will make CPU execute WKT interrupt service routine. This bit is not automatically cleared via hardware and should be cleared via software.
3	WKTR	**WKT run control** 0 = WKT is halted. 1 = WKT starts running. Note that the reload register RWK can only be written when WKT is halted (WKTR bit is 0). If WKT is written while WKTR is 1, result is unpredictable.
2:0	WKPS[2:0]	**WKT pre-scalar** These bits determine the pre-scale of WKT clock. 000 = 1/1. 001 = 1/4. 010 = 1/16. 011 = 1/64. 100 = 1/256. 101 = 1/512. 110 = 1/1024. 111 = 1/2048.

RWK – Self Wake-up Timer Reload Byte

Register	SFR Address	Reset Value
RWK	86H, Page 0	0000 _ 0000b

7	6	5	4	3	2	1	0
RWK[7:0]							
R/W							

Bit	Name	Description
7:0	RWK[7:0]	**WKT reload byte** It holds the 8-bit reload value of WKT. Note that RWK should not be FFH if the pre-scale is 1/1 for implement limitation.

CWK – Self Wake-up Timer current count value

Register	SFR Address	Reset Value
CWK	86H, Page 1	0000 _ 0000b

7	6	5	4	3	2	1	0
CWK[7:0]							
R/W							

Bit	Name	Description
7:0	CWK[7:0]	**WKT current count value** It is store value of WKT current count.

15 SERIAL PORT (UART0 & UART1)

The ML51 includes two enhanced full duplex serial ports enhanced with automatic address recognition and framing error detection. As control bits of these two serial ports are implemented the same. Generally speaking, in the following contents, there will not be any reference to serial port 1, but only to serial port 0.

Each serial port supports one synchronous communication mode, Mode 0, and three modes of full duplex UART (Universal Asynchronous Receiver and Transmitter), Mode 1, 2, and 3. This means it can transmit and receive simultaneously. The serial port is also receiving-buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the register. The receiving and transmitting registers are both accessed at SBUF. Writing to SBUF loads the transmitting register, and reading SBUF accesses a physically separate receiving register. There are four operation modes in serial port. In all four modes, transmission initiates by any instruction that uses SBUF as a destination register.

The multi function pin define for UARTs

Group	Pin Name	GPIO	MFP	Type	Description
UART0	UART0_RXD	P5.2	MFP3	I	UART0 data receiver input pin.
		P0.6	MFP7	I	
		P0.4	MFP8	I	
		P0.0	MFP7	I	
		P5.1	MFP4	I	
		P6.2	MFP9	I	
		P1.7	MFP3	I	
	UART0_TXD	P5.3	MFP3	O	UART0 data transmitter output pin.
		P0.7	MFP7	O	
		P0.5	MFP8	O	
		P0.1	MFP7	O	
		P5.0	MFP4	O	
		P6.3	MFP9	O	
		P1.6	MFP3	O	
		P3.1	MFP6	O	
UART1	UART1_RXD	P2.2	MFP6	I	UART1 data receiver input pin.
		P1.0	MFP7	I	
		P0.2	MFP8	I	
		P5.1	MFP2	I	
	UART1_TXD	P2.3	MFP6	O	UART1 data transmitter output pin.
		P1.1	MFP7	O	
		P0.3	MFP8	O	
		P5.0	MFP2	O	

15.1 Function Description

15.1.1 Operating mode

15.1.1.1 Mode 0

Mode 0 provides synchronous communication with external devices. Serial data centers and exits through RXD pin. TXD outputs the shift clocks. 8-bit frame of data are transmitted or received. Mode 0 therefore provides half-duplex communication because the transmitting or receiving data is via the same data line RXD. The baud rate is enhanced to be selected as FSYS/12 if SM2 (SCON.5) is 0 or as FSYS/2 if SM2 is 1. Note that whenever transmitting or receiving, the serial clock is always generated by the MCU. Thus any device on the serial port in Mode 0 should accept the MCU as the master. Figure 15.1-1 shows the associated timing of the serial port in Mode 0.

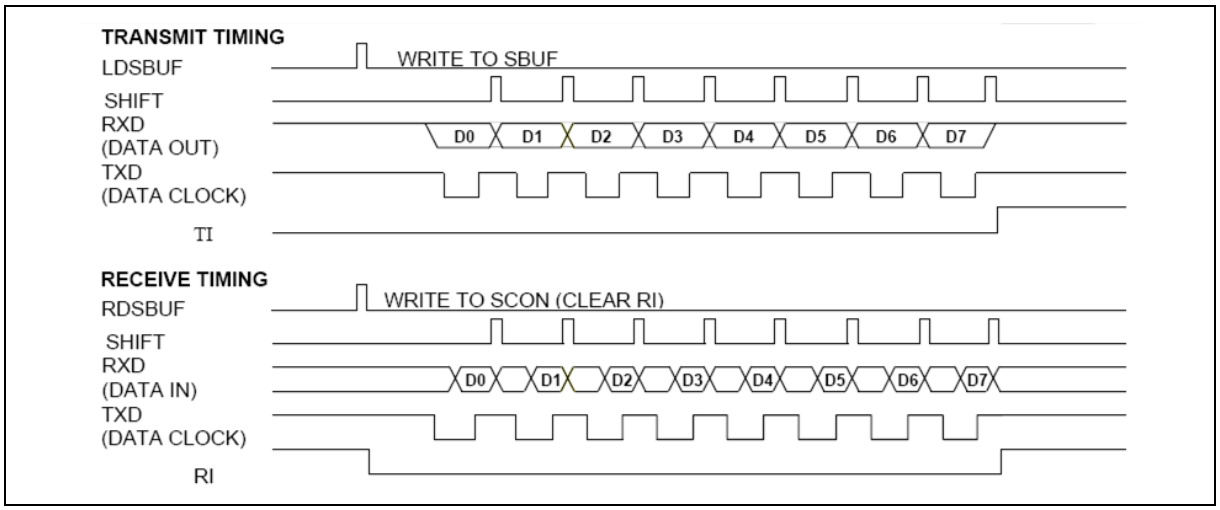


Figure 15.1-1 Serial Port Mode 0 Timing Diagram

As shown there is one bi-directional data line (RXD) and one shift clock line (TXD). The shift clocks are used to shift data in or out of the serial port controller bit by bit for a serial communication. Data bits enter or emit LSB first. The band rate is equal to the shift clock frequency.

Transmission is initiated by any instruction writes to SBUF. The control block will then shift out the clocks and begin to transfer data until all 8 bits are complete. Then the transmitted flag TI (SCON.1) will be set 1 to indicate one byte transmitting complete.

Reception is initiated by the condition REN (SCON.4) = 1 and RI (SCON.0) = 0. This condition tells the serial port controller that there is data to be shifted in. This process will continue until 8 bits have been received. Then the received flag RI will be set as 1. User can clear RI to triggering the next byte reception.

15.1.1.2 Mode 1

Mode 1 supports asynchronous, full duplex serial communication. The asynchronous mode is commonly used for communication with PCs, modems or other similar interfaces. In Mode 1, 10 bits are transmitted through TXD or received through RXD including a start bit (logic 0), 8 data bits (LSB first) and a stop bit (logic 1). The baud rate is determined by the Timer 1. SMOD (PCON.7) setting 1 makes the baud rate double. Figure 15.1-2 shows the associated timings of the serial port in Mode 1 for transmitting and receiving.

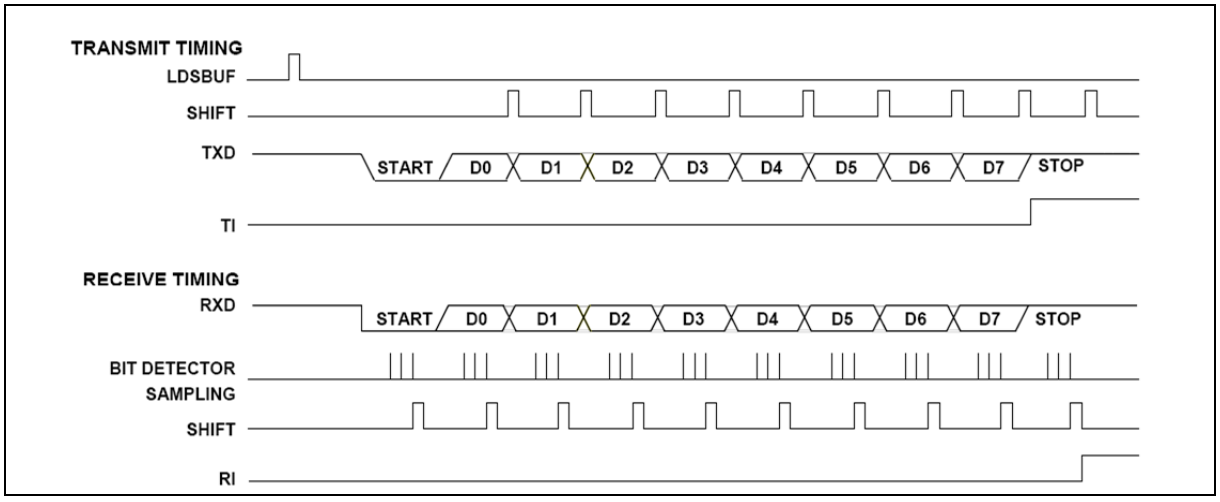


Figure 15.1-2 Serial Port Mode 1 Timing Diagram

Transmission is initiated by any writing instructions to SBUF. Transmission takes place on TXD pin. First the start bit comes out, the 8-bit data follows to be shifted out and then ends with a stop bit. After the stop bit appears, TI (SCON.1) will be set to indicate one byte transmission complete. All bits are shifted out depending on the rate determined by the baud rate generator.

Once the baud rate generator is activated and REN (SCON.4) is 1, the reception can begin at any time. Reception is initiated by a detected 1-to-0 transition at RXD. Data will be sampled and shifted in at the selected baud rate. In the midst of the stop bit, certain conditions should be met to load SBUF with the received data:

1. RI (SCON.0) = 0, and

2. Either SM2 (SCON.5) = 0, or the received stop bit = 1 while SM2 = 1 and the received data matches “Given” or “Broadcast” address. (For enhancement function, see Section 15.1.4“Multiprocessor Communication” and Section 15.1.5“Automatic Address Recognition”.)

If these conditions are met, then the SBUF will be loaded with the received data, the RB8 (SCON.2) with stop bit, and RI will be set. If these conditions fail, there will be no data loaded and RI will remain 0. After above receiving progress, the serial control will look forward another 1-to-0 transition on RXD pin to start next data reception.

15.1.1.3 Mode 2

Mode 2 supports asynchronous, full duplex serial communication. Different from Mode1, there are 11 bits to be transmitted or received. They are a start bit (logic 0), 8 data bits (LSB first), a programmable 9th bit TB8 or RB8 bit and a stop bit (logic 1). The most common use of 9th bit is to put the parity bit in it or to label address or data frame for multiprocessor communication. The baud rate is fixed as 1/32 or 1/64 the system clock frequency depending on SMOD (PCON.7) bit. Figure 15.1-3 shows the associated timings of the serial port in Mode 2 for transmitting and receiving.

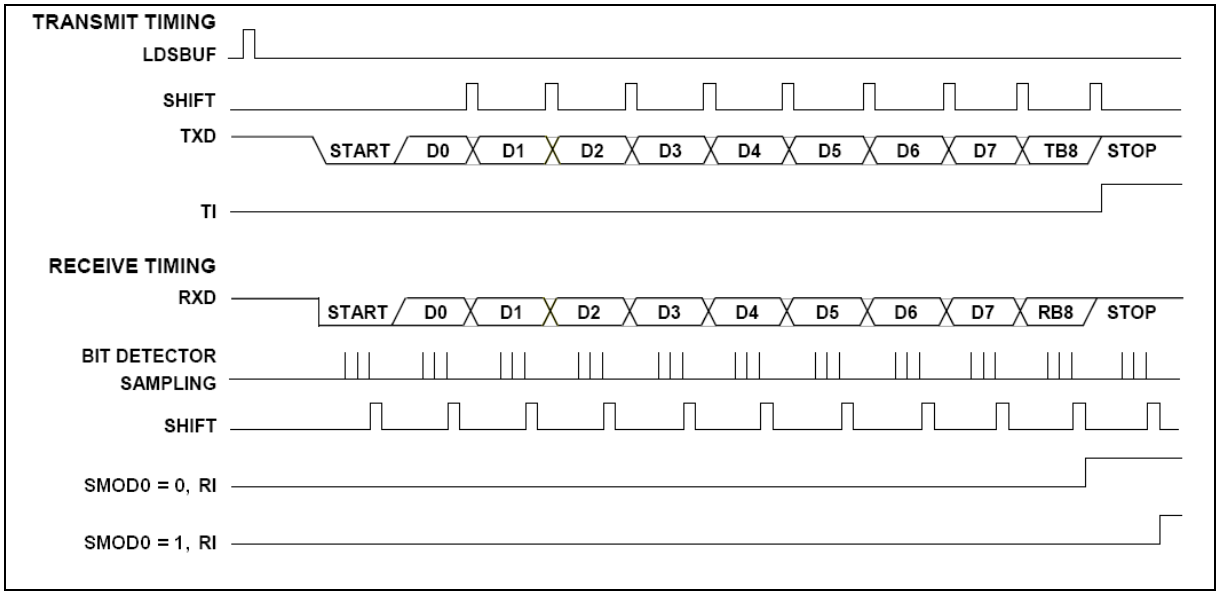


Figure 15.1-3 Serial Port Mode 2 and 3 Timing Diagram

Transmission is initiated by any writing instructions to SBUF. Transmission takes place on TXD pin. First the start bit comes out, the 8-bit data and bit TB8 (SCON.3) follows to be shifted out and then ends with a stop bit. After the stop bit appears, TI will be set to indicate the transmission complete.

While REN is set, the reception is allowed at any time. A falling edge of a start bit on RXD will initiate the reception progress. Data will be sampled and shifted in at the selected baud rate. In the midst of the stop bit, certain conditions should be met to load SBUF with the received data:

1. RI (SCON.0) = 0, and

2. Either SM2 (SCON.5) = 0, or the received 9th bit = 1 while SM2 = 1 and the received data matches “Given” or “Broadcast” address. (For enhancement function, see Section 15.1.4“Multiprocessor Communication” and Section 15.1.5“Automatic Address Recognition”.)

If these conditions are met, the SBUF will be loaded with the received data, the RB8(SCON.2) with the received 9th bit and RI will be set. If these conditions fail, there will be no data loaded and RI will remain 0. After above receiving progress, the serial control will look forward another 1-to-0 transition on RXD pin to start next data reception.

15.1.1.4 Mode 3

Mode 3 has the same operation as Mode 2, except its baud rate clock source uses Timer 1 overflows as its baud rate clocks. See Figure 15.1-3 for timing diagram of Mode 3. It has no difference from Mode 2.

15.1.2 Baud Rate

The baud rate source and speed for different modes of serial port is quite different from one another. All cases are listed in Table 15.1-1 Serial Port 0 Mode / baudrate Description. The user should calculate the baud rate according to their system configuration.

In Mode 1 or 3, the baud rate clock source of UART0 can be selected from Timer 1 or Timer 3. User can select the baud rate clock source by BRCK (T3CON.5). For UART1, its baud rate clock comes only from Timer 3 as its unique clock source.

When using Timer 1 as the baud rate clock source, note that the Timer 1 interrupt should be disabled. Timer 1 itself can be configured for either “Timer” or “Counter” operation. It can be in any of its three running modes. However, in the most typical applications, it is configured for “Timer” operation, in the auto-reload mode (Mode 2). If using Timer 3 as the baud rate generator, its interrupt should also be

disabled.

Table 15.1-1 Serial Port 0 Mode / baudrate Description

Mode	SM0 / SM1 (SCON[7:6])	SM2 (SCON[5])	SMOD (PCON[7])	Frame Bits	Baud Rate	
0	00	0	-	8	F_{SYS} divided by 12	
		1			F_{SYS} divided by 2	
1	01	-	0	10	**Time1 TM1 (CKCON[3]) = 0**	$\frac{1}{32}\times\frac{F_{SYS}}{12\times(256-TH1)}$
					Time1 TM1 (CKCON[3]) = 1	$\frac{1}{32}\times\frac{F_{SYS}}{(256-TH1)}$
					Timer 3	$\frac{1}{32}\times\frac{F_{SYS}}{Pre\text{-}scale\times(65536-(256\times RH3+RL3))}$
			1		**Time1 TM1 (CKCON[3]) = 0**	$\frac{1}{16}\times\frac{F_{SYS}}{12\times(256-TH1)}$
					Time1 TM1 (CKCON[3]) = 1	$\frac{1}{16}\times\frac{F_{SYS}}{(256-TH1)}$
					Timer 3	$\frac{1}{16}\times\frac{F_{SYS}}{Pre\text{-}scale\times(65536-(256\times RH3+RL3))}$
2	10	-	0	11	F_{SYS} divided by 64	
			1		F_{SYS} divided by 32	
3	11	-	0	11	**Time 1[1] TM1 (CKCON[3]) = 0**	$\frac{1}{32}\times\frac{F_{SYS}}{12\times(256-TH1)}$
					Time 1 [1] TM1 (CKCON[3]) = 1	$\frac{1}{32}\times\frac{F_{SYS}}{(256-TH1)}$
					Timer 3	$\frac{1}{32}\times\frac{F_{SYS}}{Pre\text{-}scale\times(65536-(256\times RH3+RL3))}$
			1		**Time1 [1] TM1 (CKCON[3]) = 0**	$\frac{1}{16}\times\frac{F_{SYS}}{12\times(256-TH1)}$

					Time1[1] **TM1 (CKCON[3]) = 1**	$\frac{1}{16}\times\frac{F_{SYS}}{(256\text{-}TH1)}$
					Timer 3	$\frac{1}{16}\times\frac{F_{SYS}}{Pre\text{-}scale\times(65536-(256\times RH3+RL3))}$

Note: Timer 1 should configured as a timer in auto-reload mode (Mode 2).

Table 15.1-2 Serial Port 1 Mode / baudrate Description

Mode	SM0_1 / SM1_1 (S1CON[7:6])	SMOD_1 (T3CON[7])	Frame Bits	Baud Rate	
0	00	-	8	F_{SYS} divided by 12	
1	01	0	10	**Timer 3**	$\frac{1}{32}\times\frac{F_{SYS}}{Pre\text{-}scale\times(65536-(256\times RH3+RL3))}$
		1		**Timer 3**	$\frac{1}{16}\times\frac{F_{SYS}}{Pre\text{-}scale\times(65536-(256\times RH3+RL3))}$
2	10	0	11	F_{SYS} divided by 64	
		1		F_{SYS} divided by 32	
3	11	0	11	**Timer 3**	$\frac{1}{32}\times\frac{F_{SYS}}{Pre\text{-}scale\times(65536-(256\times RH3+RL3))}$
		1		**Timer 3**	$\frac{1}{16}\times\frac{F_{SYS}}{Pre\text{-}scale\times(65536-(256\times RH3+RL3))}$

Sample code: we list the most popular UART setting Mode 1 initial step as following：

Serial port 0 (**UART0**) use **timer 1** as baudrate generator: Formula is $\frac{1}{16}\times\frac{F_{SYS}}{(256\text{-}TH1)}$

```
SCON = 0x50;        //UART0 Mode1,REN=1,TI=1
TMOD |= 0x20;       //Timer1 set to Mode2 auto reload mode (must)
PCON |= 0x80;       //UART0 Double Rate Enable
CKCON |= 0x10;      //Timer 1 as clock source
T3CON &= 0xDF;      //Timer1 as UART0 clock source
TH1 = value;
TR1=1;
```

Serial port 0 (**UART0**) use **timer 3** as baudrate generator: Formula is

$$\frac{1}{16}\times\frac{F_{SYS}}{\text{Pre-scale}\times(65536-(256\times RH3+RL3))}$$

```
SCON = 0x50;      //UART0 Mode1,REN=1,TI=1
PCON |= 0x80;     //UART0 Double Rate Enable
T3CON &= 0xF8;    //(Prescale=1)
T3CON |= 0x20;    //UART0 baud rate clock source = Timer3
RH3 = value high byte
RL3 = value low byte
T3CON|= 0x08;          //Trigger Timer3
```

Serial port 1 (UART1) use Timer 3 as baudrate generator: Fomula is

$$\frac{1}{16}\times\frac{F_{SYS}}{\text{Pre-scale}\times(65536-(256\times RH3+RL3))}$$

```
SCON_1 = 0x52;     //UART1 Mode1,REN_1=1,TI_1=1
T3CON = 0xF8;      //T3PS2=0,T3PS1=0,T3PS0=0(Prescale=1),
RH3 = value high byte
RL3 = value low byte
T3CON|= 0x08;
```

Following list some popular baudrate value base on different Fsys and the deviation value:

Fsys Value	Baud Rate	TH1 Value (Hex)	RH3,RL3 Value (Hex)	Baudrate Deviation
24000000	4800	64 (SMOD=0)	FEC8	0.160256%
	9600	64	FF64	0.160256%
	19200	B2	FFB2	0.160256%
	38400	D9	FFD9	0.160256%
	57600	E6	FFE6	0.160256%
	115200	F3	FFF3	0.160256%
	150000	F6	FFF6	0.000000%
	166666	F7	FFF7	0.000400%
	187500	F8	FFF8	0.000000%
	214285	F9	FFF9	0.000333%
	250000	FA	FFFA	0.000000%
	300000	FB	FFFB	0.000000%
	375000	FC	FFFC	0.000000%
	500000	FD	FFFD	0.000000%
	750000	FE	FFFE	0.000000%
	1500000	FF	FFFF	0.000000%

Fsys Value	Baud Rate	TH1 Value (Hex)	RH3,RL3 Value (Hex)	Baudrate Deviation
22118400	4800	70 (SMOD=0)	FEE0	0.000000%
	9600	70	FF70	0.000000%
	19200	B8	FFB8	0.000000%
	38400	DC	FFDC	0.000000%
	57600	E8	FFE8	0.000000%
	115200	F4	FFF4	0.000000%
	230400	FA	FFFA	0.000000%
	276480	FB	FFFB	0.000000%
	345600	FC	FFFC	0.000000%
	460800	FD	FFFD	0.000000%
	691200	FE	FFFE	0.000000%
	1382400	FF	FFFF	0.000000%
16600000	4800	28	FF28	0.067515%
	9600	94	FF94	0.067515%
	19200	CA	FFCA	0.067515%
	38400	E5	FFE5	0.067515%
	57600	EE	FFEE	0.067515%
	115200	F7	FFF7	0.067515%
16000000	4800	30	FF30	0.160256%
	9600	98	FF98	0.160256%
	19200	CC	FFCC	0.160256%
	38400	E6	FFE6	0.160256%
	57600	EF	FFEF	2.124183%
	115200	F7	FFF7	-3.549383%
	200000	FB	FFFB	0.000000%
	250000	FC	FFFC	0.000000%
	333333	FD	FFFD	0.000100%
	500000	FE	FFFE	0.000000%
	1000000	FF	FFFF	0.000000%
11059200	4800	70	FF70	0.000000%
	9600	B8	FFB8	0.000000%
	19200	DC	FFDC	0.000000%
	38400	EE	FFEE	0.000000%
	57600	F4	FFF4	0.000000%

Fsys Value	Baud Rate	TH1 Value (Hex)	RH3,RL3 Value (Hex)	Baudrate Deviation
	115200	FA	FFFA	0.000000%
	230400	FD	FFFD	0.000000%
	345600	FE	FFFE	0.000000%
	691200	FF	FFFF	0.000000%

15.1.3 Framing Error Detection

Framing error detection is provided for asynchronous modes. (Mode 1, 2, or 3.) The framing error occurs when a valid stop bit is not detected due to the bus noise or contention. The UART can detect a framing error and notify the software.

The framing error bit, FE, is located in SCON.7. This bit normally serves as SM0. While the framing error accessing enable bit SMOD0 (PCON.6) is set 1, it serves as FE flag. Actually, SM0 and FE locate in different registers.

The FE bit will be set 1 via hardware while a framing error occurs. FE can be checked in UART interrupt service routine if necessary. Note that SMOD0 should be 1 while reading or writing to FE. If FE is set, any following frames received without frame error will not clear the FE flag. The clearing has to be done via software.

15.1.4 Multiprocessor Communication

The ML51 multiprocessor communication feature lets a master device send a multiple frame serial message to a slave device in a multi-slave configuration. It does this without interrupting other slave devices that may be on the same serial line. This feature can be used only in UART Mode 2 or 3. User can enable this function by setting SM2 (SCON.5) as logic 1 so that when a byte of frame is received, the serial interrupt will be generated only if the 9th bit is 1. (For Mode 2, the 9th bit is the stop bit.) When the SM2 bit is 1, serial data frames that are received with the 9th bit as 0 do not generate an interrupt. In this case, the 9th bit simply separates the slave address from the serial data.

When the master device wants to transmit a block of data to one of several slaves on a serial line, it first sends out an address byte to identify the target slave. Note that in this case, an address byte differs from a data byte. In an address byte, the 9th bit is 1 and in a data byte, it is 0. The address byte interrupts all slaves so that each slave can examine the received byte and see if it is addressed by its own slave address. The addressed slave then clears its SM2 bit and prepares to receive incoming data bytes. The SM2 bits of slaves that were not addressed remain set, and they continue operating normally while ignoring the incoming data bytes.

Follow the steps below to configure multiprocessor communications:

1. Set all devices (masters and slaves) to UART Mode 2 or 3.

2. Write the SM2 bit of all the slave devices to 1.

3. The master device's transmission protocol is:

First byte: the address, identifying the target slave device, (9th bit = 1).

Next bytes: data, (9th bit = 0).

4. When the target slave receives the first byte, all of the slaves are interrupted because the 9th data bit is 1. The targeted slave compares the address byte to its own address and then clears its SM2 bit to receiving incoming data. The other slaves continue operating normally.

5. After all data bytes have been received, set SM2 back to 1 to wait for next address.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit. For

Mode 1 reception, if SM2 is 1, the receiving interrupt will not be issue unless a valid stop bit is received.

15.1.5 Automatic Address Recognition

The automatic address recognition is a feature, which enhances the multiprocessor communication feature by allowing the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address, which passes by the serial port. Only when the serial port recognizes its own address, the receiver sets RI bit to request an interrupt. The automatic address recognition feature is enabled when the multiprocessor communication feature is enabled, SM2 is set.

If desired, user may enable the automatic address recognition feature in Mode 1. In this configuration, the stop bit takes the place of the ninth data bit. RI is set only when the received command frame address matches the device's address and is terminated by a valid stop bit.

Using the automatic address recognition feature allows a master to selectively communicate with one or more slaves by invoking the "Given" slave address or addresses. All of the slaves may be contacted by using the "Broadcast" address. Two SFR are used to define the slave address, SADDR, and the slave address mask, SADEN. SADEN is used to define which bits in the SADDR are to be used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "Given" address, which the master will use for addressing each of the slaves. Use of the "Given" address allows multiple slaves to be recognized while excluding others.

The following examples will help to show the versatility of this scheme.

Example 1, slave 0:

SADDR = 11000000b

SADEN = 11111101b

Given = 110000X0b

Example 2, slave 1:

SADDR = 11000000b

SADEN = 11111110b

Given = 1100000Xb

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires 0 in bit 0 and it ignores bit 1. Slave 1 requires 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires 0 in bit 1. A unique address for slave 1 would be 11000001b since 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address, which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 11000000b as their "Broadcast" address.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

Example 1, slave 0:

SADDR = 11000000b

SADEN = 11111001b

Given = 11000XX0b

Example 2, slave 1:

SADDR = 11100000b

SADEN = 11111010b

Given = 11100X0Xb

Example 3, slave 2:

SADDR = 11000000b

SADEN = 11111100b

Given = 110000XXb

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 11100110b. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 11100101b. Slave 2 requires that bit 2 = 0 and its unique address is 11100011b. To select Slaves 0 and 1 and exclude Slave 2 use address 11100100b, since it is necessary to make bit 2 = 1 to exclude slave 2.

The “Broadcast” address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are treated as “don’t-cares”, e.g.:

SADDR = 01010110b

SADEN = 11111100b

Broadcast = 1111111Xb

The use of don’t-care bits provides flexibility in defining the Broadcast address, however in most applications, interpreting the “don’t-cares” as all ones, the broadcast address will be FFH.

On reset, SADDR and SADEN are initialized to 00H. This produces a “Given” address of all “don’t cares” as well as a “Broadcast” address of all XXXXXXXXb (all “don’t care” bits). This ensures that the serial port will reply to any address.

15.2 Control register of Serial Port Register Description

SCON – Serial Port Control

Register	SFR Address	Reset Value
SCON	98H, All pages, Bit-addressable	0000 _0000 b

7	6	5	4	3	2	1	0
SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	SM0/FE	**Serial port mode select** SMOD0 (PCON.6) = 0: See Table 15.1-1 Serial Port 0 Mode / baudrate Description for details. SMOD0 (PCON.6) = 1: SM0/FE bit is used as frame error (FE) status flag. It is cleared by software. 0 = Frame error (FE) did not occur. 1 = Frame error (FE) occurred and detected.
6	SM1	
5	SM2	**Multiprocessor communication mode enable** The function of this bit is dependent on the serial port 0 mode. Mode 0: This bit select the baud rate between FSYS/12 and FSYS/2. 0 = The clock runs at FSYS/12 baud rate. It maintains standard 8051compatibility. 1 = The clock runs at FSYS/2 baud rate for faster serial communication. Mode 1: This bit checks valid stop bit. 0 = Reception is always valid no matter the logic level of stop bit. 1 = Reception is valid only when the received stop bit is logic 1 and the received data matches “Given” or “Broadcast” address. Mode 2 or 3: For multiprocessor communication. 0 = Reception is always valid no matter the logic level of the 9th bit. 1 = Reception is valid only when the received 9th bit is logic 1 and the received data matches “Given” or “Broadcast” address.
4	REN	**Receiving enable** 0 = Serial port 0 reception Disabled. 1 = Serial port 0 reception Enabled in Mode 1,2, or 3. In Mode 0, reception is initiated by the condition REN = 1 and RI = 0.
3	TB8	**9th transmitted bit** This bit defines the state of the 9th transmission bit in serial port 0 Mode 2 or 3. It is not used in Mode 0 or 1.

Bit	Name	Description
2	RB8	**9th received bit** The bit identifies the logic level of the 9th received bit in serial port 0 Mode 2 or 3. In Mode 1, RB8 is the logic level of the received stop bit. SM2 bit as logic 1 has restriction for exception. RB8 is not used in Mode 0.
1	TI	**Transmission interrupt flag** This flag is set by hardware when a data frame has been transmitted by the serial port 0 after the 8th bit in Mode 0 or the last data bit in other modes. When the serial port 0 interrupt is enabled, setting this bit causes the CPU to execute the serial port 0 interrupt service routine. This bit should be cleared manually via software.
0	RI	**Receiving interrupt flag** This flag is set via hardware when a data frame has been received by the serial port 0 after the 8th bit in Mode 0 or after sampling the stop bit in Mode 1, 2, or 3. SM2 bit as logic 1 has restriction for exception. When the serial port 0 interrupt is enabled, setting this bit causes the CPU to execute to the serial port 0 interrupt service routine. This bit should be cleared manually via software.

S1CON – Serial Port 1 Control

Register	SFR Address	Reset Value
S1CON	F8H, All pages, Bit-addressable	0000_0000 b

7	6	5	4	3	2	1	0
SM0_1/FE_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	SM0_1/FE_1	Serial port 1 mode select SMOD0_1 (T3CON.6) = 0: See Table 15.1-2 Serial Port 1 Mode / baudrate Description for details. SMOD0_1 (T3CON.6) = 1: SM0_1/FE_1 bit is used as frame error (FE) status flag. It is cleared by software. 0 = Frame error (FE) did not occur. 1 = Frame error (FE) occurred and detected.
6	SM1_1	

Bit	Name	Description
5	SM2_1	Multiprocessor communication mode enable The function of this bit is dependent on the serial port 1 mode. Mode 0: No effect. Mode 1: This bit checks valid stop bit. 0 = Reception is always valid no matter the logic level of stop bit. 1 = Reception is valid only when the received stop bit is logic 1 and the received data matches “Given” or “Broadcast” address. Mode 2 or 3: For multiprocessor communication. 0 = Reception is always valid no matter the logic level of the 9th bit. 1 = Reception is valid only when the received 9th bit is logic 1 and the received data matches “Given” or “Broadcast” address.
4	REN_1	Receiving enable 0 = Serial port 1 reception Disabled. 1 = Serial port 1 reception Enabled in Mode 1,2, or 3. In Mode 0, reception is initiated by the condition REN_1 = 1 and RI_1 = 0.
3	TB8_1	9th transmitted bit This bit defines the state of the 9th transmission bit in serial port 1 Mode 2 or 3. It is not used in Mode 0 or 1.
2	RB8_1	9th received bit The bit identifies the logic level of the 9th received bit in serial port 1 Mode 2 or 3. In Mode 1, RB8_1 is the logic level of the received stop bit. SM2_1 bit as logic 1 has restriction for exception. RB8_1 is not used in Mode 0.
1	TI_1	Transmission interrupt flag This flag is set by hardware when a data frame has been transmitted by the serial port 1 after the 8th bit in Mode 0 or the last data bit in other modes. When the serial port 1 interrupt is enabled, setting this bit causes the CPU to execute the serial port 1 interrupt service routine. This bit must be cleared manually via software.
0	RI_1	Receiving interrupt flag This flag is set via hardware when a data frame has been received by the serial port 1 after the 8th bit in Mode 0 or after sampling the stop bit in Mode 1, 2, or 3. SM2_1 bit as logic 1 has restriction for exception. When the serial port 1 interrupt is enabled, setting this bit causes the CPU to execute to the serial port 1 interrupt service routine. This bit must be cleared manually via software.

PCON – Power Control

Register	SFR Address	Reset Value
CWK	87H, All pagess	POR: 0001_000b, other: 000U_0000b

7	6	5	4	3	2	1	0
SMOD	SMOD0	LPR	POF	GF1	GF0	PD	IDL
R/W	R/W	RW	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	SMOD	Serial port 0 double baud rate enable Setting this bit doubles the serial port baud rate when UART0 is in Mode 2 or when Timer 1 overflow is used as the baud rate source of UART0 Mode 1 or 3. See Table 15.1-1 Serial Port 0 Mode / baudrate Description for details.
6	SMOD0	Serial port 0 framing error flag access enable 0 = SCON.7 accesses to SM0 bit. 1 = SCON.7 accesses to FE bit.

T3CON – Timer 3 Control

Register	SFR Address	Reset Value
T3CON	C4H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SMOD_1	SMOD0_1	BRCK	TF3	TR3	T3PS[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W		

Bit	Name	Description
7	SMOD_1	Serial port 1 double baud rate enable Setting this bit doubles the serial port baud rate when UART1 is in Mode 2. See Table 15.1-2 Serial Port 1 Mode / baudrate Description for details.
6	SMOD0_1	Serial port 1 framing error access enable 0 = S1CON.7 accesses to SM0_1 bit. 1 = S1CON.7 accesses to FE_1 bit.

SBUF – Serial Port Data Buffer

Register	SFR Address	Reset Value
SBUF	99H, Page 0	0000 _0000 b
SBUF1	9AH, Page 0	0000 _0000 b

7	6	5	4	3	2	1	0
SBUFn[7:0]							
R/W							

Bit	Name	Description
7:0	SBUF[7:0]	**Serial port data buffer** This byte actually consists two separate registers. One is the receiving resister, and the other is the transmitting buffer. When data is moved to SBUF, it goes to the transmitting buffer and is shifted for serial transmission. When data is moved from SBUF, it comes from the receiving register. The transmission is initiated through giving data to SBUF.

IE – Interrupt Enable

Register	SFR Address	Reset Value
IE	A8H, All pages, Bit addressable	0000 _0000 b

7	6	5	4	3	2	1	0
EA	EADC	EBOD	ES	ET1	EX1	ET0	EX0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
4	ES	Enable serial port 0 interrupt 0 = Serial port 0 interrupt Disabled. 1 = Interrupt generated by TI (SCON.1) or RI (SCON.0) Enabled.

EIE1 – Extensive Interrupt Enable 1

Register	SFR Address	Reset Value
EIE0	9CH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
EFB1	EPWM1	EI^{2}C1	ESPI1	EHFI	EWKT	ET3	ES1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description

Bit	Name	Description
0	ES1	Enable serial port 1 interrupt 0 = Serial port 1 interrupt Disabled. 1 = Serial port 1Interrupt Enable. When interrupt generated TI_1 (S1CON.1) or RI_1 (S1CON.0) set 1.

SADDR – Slave 0 Address

Register	SFR Address	Reset Value
SADDR	A9H, Page 0	0000 _0000 b

7	6	5	4	3	2	1	0
SADDR[7:0]							
R/W							

Bit	Name	Description
7:0	SADDR[7:0]	Slave 0 address This byte specifies the microcontroller's own slave address for UATR0 multi-processor communication.

SADEN – Slave 0 Address Mask

Register	SFR Address	Reset Value
SADEN	B9H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SADEN[7:0]							
R/W							

Bit	Name	Description
7:0	SADEN[7:0]	**Slave 0 address mask** This byte is a mask byte of UART0 that contains "don't-care" bits (defined by zeros) to form the device's "Given" address. The don't-care bits provide the flexibility to address one or more slaves at a time.

SADDR1 – Slave 1 Address

Register	SFR Address	Reset Value
SADDR1	BBH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SADDR1[7:0]							
R/W							

Bit	Name	Description
7:0	SADDR1[7:0]	**Slave 1 address** This byte specifies the microcontroller's own slave address for UART1 multi-processor communication.

SADEN1 – Slave 1 Address Mask

Register	SFR Address	Reset Value
SADEN1	BAH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SADEN1[7:0]							
R/W							

Bit	Name	Description
7:0	SADEN1[7:0]	**Slave 1 address mask** This byte is a mask byte of UART1 that contains "don't-care" bits (defined by zeros) to form the device's "Given" address. The don't-care bits provide the flexibility to address one or more slaves at a time.

16 SMART CARD INTERFACE (SC)

The ML51 provides Smart Card Interface controller (SC controller) with asynchronous protocal based on ISO/IEC 7816-3 standard. Software controls GPIO pins as the smartcard reset function and card detection function. This controller also provides UART emulation for high precision baud rate communication.

- ISO-7816-3 T = 0, T = 1 compliant
- Programmable transmission clock frequency
- Programmable extra guard time selection
- Supports auto inverse convention function
- Supports UART mode
  - Full duplex, asynchronous communications
  - Supports programmable baud rate generator for each channel
  - Programmable transmitting data delay time between the last stop bit leaving the TX-FIFO and the de-assertion by setting SCnEGT register
  - Programmable even, odd or no parity bit generation and detection
  - Programmable stop bit, 1 or 2 stop bit generation

Following is the Smart Card multi function pin define

Group	Pin Name	GPIO	MFP	Type	Description
Smart Card UART2	SC_DAT / UART2_RXD	P2.4	MFP13	I	UART2 data receiver input pin.
		P2.0	MFP7	I	
		P5.5	MFP2	I	
		P0.1	MFP6	I	
		P4.4	MFP8	I	
		P4.0	MFP8	I	
	SC_CLK / UART2_TXD	P2.5	MFP13	O	UART2 data transmitter output pin.
		P2.1	MFP7	O	
		P5.4	MFP2	O	
		P0.0	MFP6	O	
		P4.5	MFP8	O	
		P4.1	MFP8	O	

16.1 Smart card Module Block Diagram

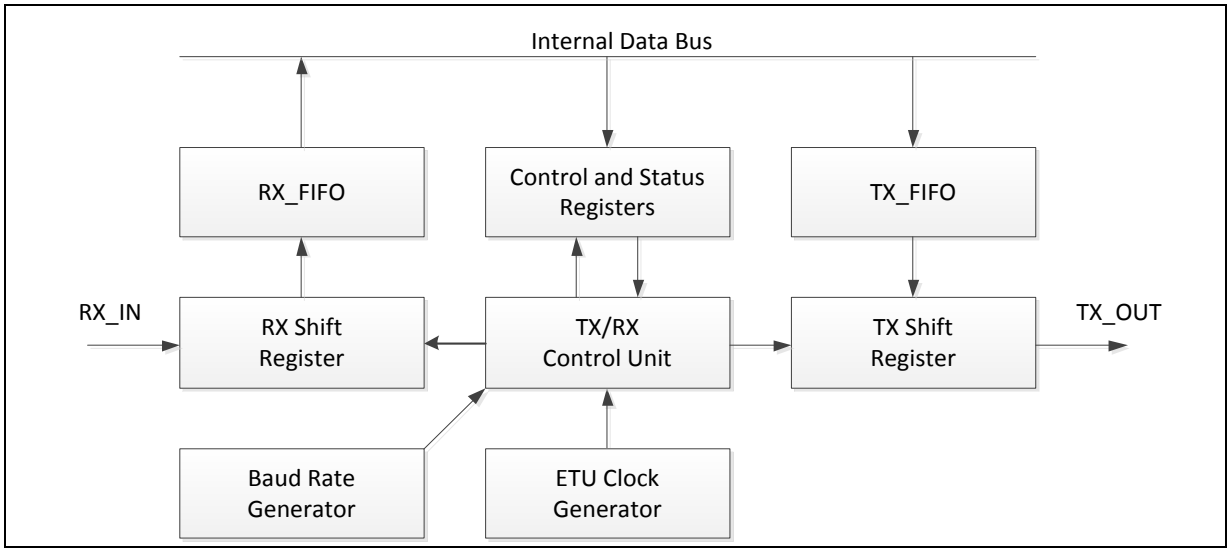


Figure 16.1-1 SC Controller Block Diagram

16.2 Functional Description

16.2.1 Operating Modes

16.2.1.1 Smart Card Mode

The Smart Card Interface controller supports activation, cold reset, warm reset and deactivation sequence by software control. The activation, cold reset, warm reset and deactivation and sequence are shown as follows.

SC Interface Connection

The SC interface connection is shown in Figure 15.3-1

1. SC_CLK / UART2_TXD : SC clock pin (output from MCU)
2. SC_DAT / UART2_RXD : SC data pin (bi-directional)
3. SC_RST: SC reset pin (output from MCU, firmware assigned GPIO)
4. SC_PWR: SC power pin (output from MCU, firmware assigned GPIO)
5. *: SC_PWR is used for power control function to turn ON/OFF the power for Smart Card. **Do not** use SC_PWR as the direct power supply for Smart Card.
6. SC_CD: SC card detect pin (input to MCU, detect card by a card insert mechanism)

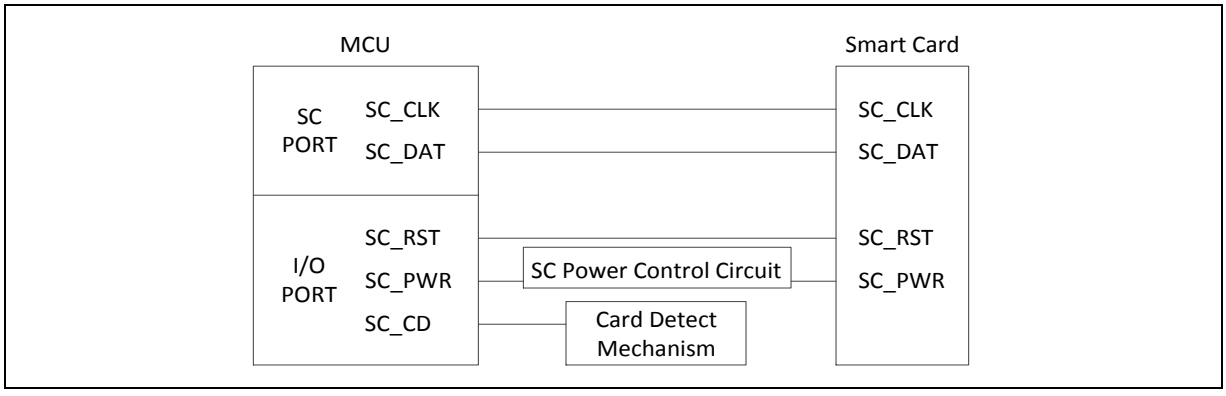
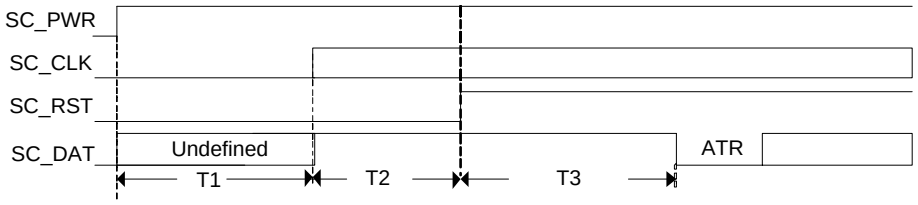


Figure 16.2 SC Interface Connection

Activation and Cold Reset

The activation and cold reset sequence is shown in Figure 15.3-2

1. Set SC_RST to low by software programming to ‘0’
2. Set SC_PWR at high level by software programming to ‘1’ before timing T1 and SC_DAT at high level (reception mode) by software programming to ‘1’ period of timing T1.
3. Enable SC_CLK clock by programming CLKKEEP (SCCR2[1]) to ‘1’ after timing T1.
4. De-assert SC_RST to high by software programming to ‘1’ after timing T2.
5. Smart Card host controller read the card ATR period of timing T3.



Time	Comment
T1	SC_PWR to SC_CLK Start
T2	SC_CLK Start to SC_RST Assert
T3	SC_RST Start to ATR Appear

Suggestion timing (Unit SC Clock)

T1	T2
83.5	491
133	537
165	569
165	42060

400 <= T3 <= 40000

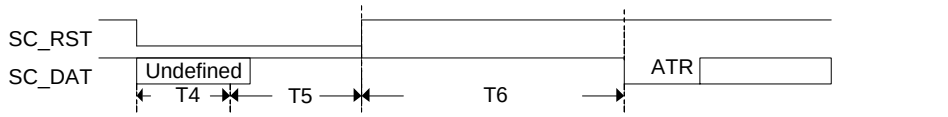
Note : The values are measured by chip I / O pin and the real value will depend on system design

Figure 16.2 SC Activation and Cold Reset Sequence

Warm Reset

The warm reset sequence is showed in Figure 15.3-3

1. Set SC_RST to low by software programming to ‘0’ before timing T4.
2. Set SC_DAT to high by software programming to ‘1’ period of timing T4.
3. Set SC_RST to high by software programming to ‘1’ after timing T5.
4. Smart Card host controller read the card ATR period of timing T6.



Time	Comment
T4	SC_RST to SC_DAT Reception Mode
T5	SC_DAT Reception Mode to SC_RST Assert
T6	SC_RST start to ATR Appear

Suggesting timing (Unit SC Clock)

T4	T5
81	483
129	531
161	563
161	42106

400 <= T6 <= 40000

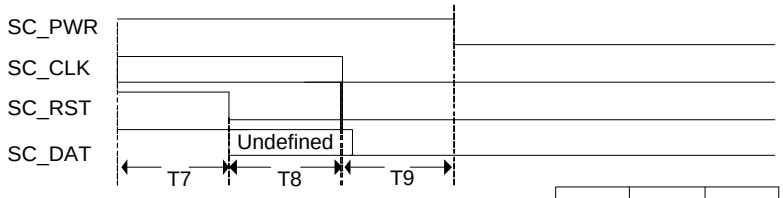
Note: This value is measured by chip IO pin, the real value will depended on system design

Figure 15.3-3 SC Warm Reset Sequence

Deactivation

The deactivation sequence is showedin Figure 15.3-4

1. Set SC_RST to low by software programming to '0' period of timing T7.
2. Stop SC_CLK by programming CLKKEEP (SCCR2[1]) to '0' period of timing T8.
3. Set SC_DAT to low by software programming to '0' period of timing T8.
4. Deactivate SC_PWR by software programming to '0' period of timing T9.



Time	Comment
T7	Deactivation Trigger to SC_RST Low
T8	SMC_ RST Low to Stop SC _CLK
T9	Stop SC_CLK to Stop SC_PWR

Unit SC Clock

T7	T8	T9
97	83	87
145	131	135
177	163	167
177	163	167

Figure 16.2 SC Deactivation Sequence

16.2.1.2 UART Mode

When the UARTEN (SCCR2[0]) bit is set, the Smart Card Interface controller can also be used as basic UART function. The following is the program example for UART mode.

Programming example:

1. Set UARTEN (SCCR2[0]) bit to enter UART mode.
2. Fill "0" to CONSEL (SCnCR1[4]) and AUTOCEN (SCnCR1[3]) field. (In UART mode, those fields must be "0")
3. Select the UART baud rate by setting ETURDIV[11:0] ({SCnETURD1[3:0]:SCnETURD0[7:0]}) fields. For example, if smartcard module clock is 12 MHz and target baud rate is 115200bps, ETURDIV should fill with (12000000 / 115200 – 1).

4. Select the data format include data length (by setting WLS (SCCR2[5:4]), parity format (by setting OPE (SCCR2[7]) and PBOFF (SCCR2[6])) and stop bit length (by setting NSB (SCnCR1[7])).
5. Write the SCnDR (SCnDR[7:0]) (TX) register or read the SCnDR (SCnDR[7:0]) (RX) register can perform UART function.

16.2.2 Smart Card Data Transfer

Basically, the smart card interface acts as a half-duplex asynchronous communication port and its data format is composed of ten consecutive bits which is showman.

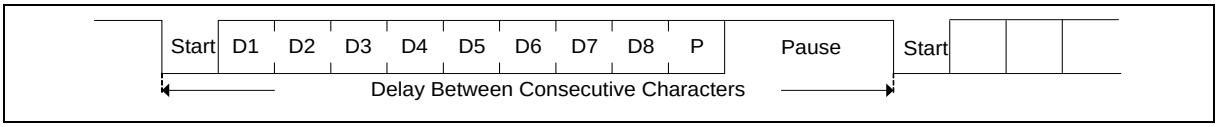
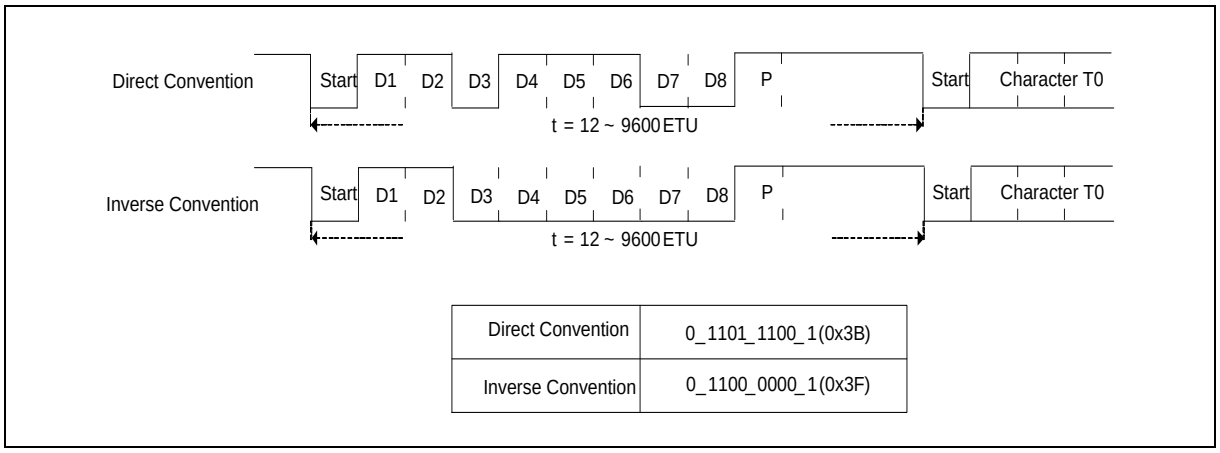


Figure 16.2-1 SC Data Character

16.2.3 Initial Character TS

According to 7816-3, the initial character TS has two possible patterns shown in Figure 16.2-2 Initial Character TS. If the TS pattern is 1100_0000, it is inverse convention. When decoded by inverse convention, the conveyed byte is equal to 0x3F. If the TS pattern is 1101_1100, it is direct convention. When decoded by direct convention, the conveyed byte is equal to 0x3B.Software can set AUTOCEN (SCnCR1[3]) and then the operating convention will be decided by hardware. Software can also set the CONSEL (SCnCR1[4]) register (set to '0' or '1') to change the operating convention after SC received TS of answer to request (ATR).

If auto convention function is enabled by setting AUTOCEN (SCnCR1[3]) register, the setting step must be done before Answer to Request state and the first data must be 0x3B or 0x3F. After hardware received first data and stored it at buffer, the hardware will decided the convention and change the CONSEL (SCnCR1[4]) register automatically. If the first data is neither 0x3B nor 0x3F, the hardware will generate an interrupt (if ACERRIEN (ScnIE[4] = '1') to CPU.



Direct Convention	0_1101_1100_1(0x3B)
Inverse Convention	0_1100_0000_1(0x3F)

Figure 16.2-2 Initial Character TS

16.2.4 Error Signal and Character Repetition

According to ISO7816-3 T=0 mode description, as shown in Figure 16.2-3,if the receiver receives a wrong parity bit, it will pull the SC_DAT to low by 1.5 bit period to inform the transmitter parity error. Then the transmitter will retransmit the character. The SC interface controller supports hardware error detection function(SC0TSR[4]) in receiver, SC controller will generate a transfer error interrupt(if TERRIEN(ScnIE[2] = '1') to CPU.

When in T=1 mode, the receiver will not pull the SC_DAT to low by 1.5 bit period to inform the transmitter parity error.

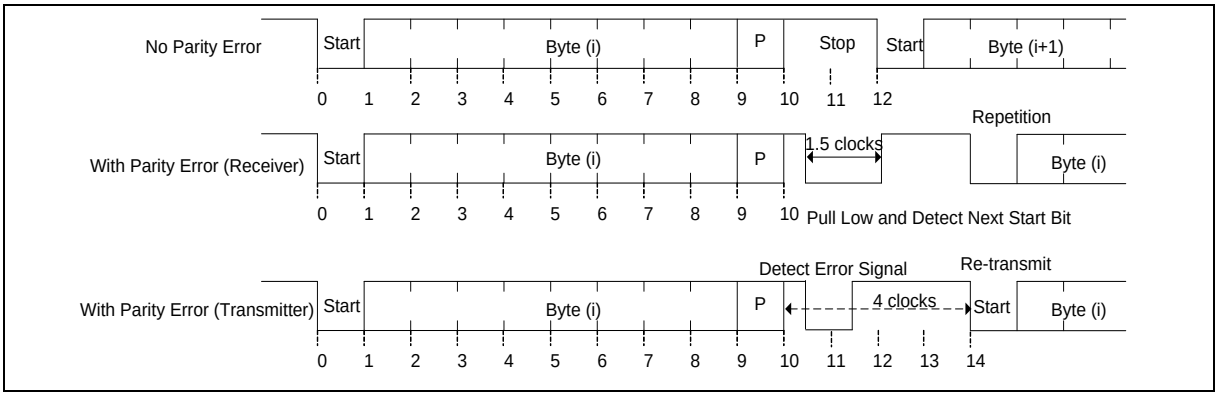


Figure 16.2-3 SC Error Signal

16.2.5 Block Guard Time and Extra Guard Time

Block guard time means the minimum bit length between the leading edges of two consecutive characters between different transfer directions. This field indicates the counter for the bit length of block guard time.

According to ISO7816-3, in T = 0 mode, software must fill T bit = 0 (real block guard time = 16.5) to this field; in T = 1 mode, software must fill T bit = 1 (real block guard time = 22.5) to it.

In transmit direction, the smart card sends data to smart card host controller, first. After the period is greater than (16.5 or 22.5, by T bit setting), the smart card host controller begin to send the data.

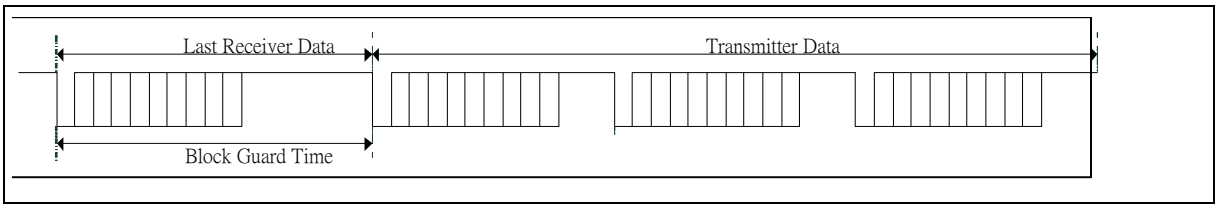


Figure 16.2-4 Transmit Direction Block Guard Time Operation

In receive direction, the smart card host controller sends data to smart card, first. If the smart card sends data to smart card host controller at the time which is less than (16.5 or 22.5, by T bit setting),the block guard time interrupt BGTIF (ScnIS[3]) is generated when RXBGTEN (SCnCR1[5]) and BGTIEN (ScnIE[3]) are enabled.

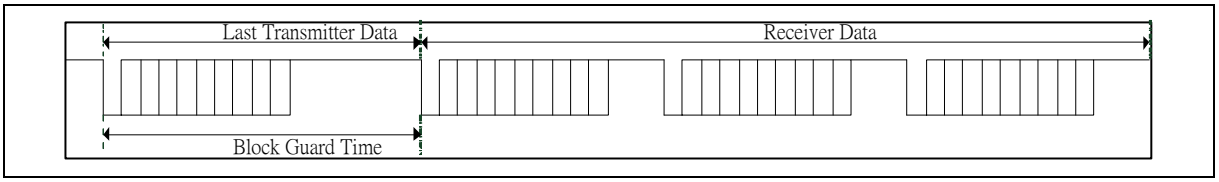


Figure 16.2-5 Receive Direction Block Guard Time Operation

Extra Guard Time is EGT (SCnEGT[7:0]),it only affects the data transmitted by smart card interface, the format is shown as Figure 16.2-6.

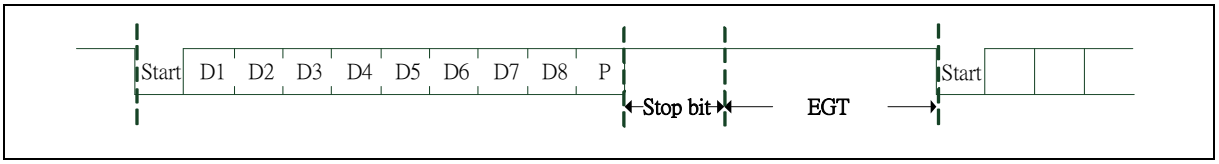


Figure 16.2-6 Extra Guard Time Operation

16.3 Control Registers of SC Controller

SCnCR0 – SC Control Register 0

Register	SFR Address	Reset Value
SC0CR0	D6H, Page 1	0000_0000 b
SC1CR0	DEH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
NSB	T	RXBGTEN	CONSEL	AUTOCEN	TXOFF	RXOFF	SCEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	NSB	Stop Bit Length This field indicates the length of stop bit. 0 = The stop bit length is 2 ETU. 1= The stop bit length is 1 ETU. **Note:** The default stop bit length is 2. SC and UART adopt NSB to program the stop bit length.
6	T	T Mode 0 = T0 (ISO7816-3 T = 0 mode). 1 = T1 (ISO7816-3 T = 1 mode). The T mode controls the BGT (Block Guard Time). Block guard time means the minimum bit length between the leading edges of two consecutive characters between different transfer directions. This field indicates the counter for the bit length of block guard time. According to ISO7816-3, in T = 0 mode, the software must clear T bit to 0 for real block guard time = 16.5. In T = 1 mode, the software must set T bit to 1 for real block guard time = 22.5. **Note:** In T = 0 mode, the receiver will generate the parity flag on PEF(SC0TSR[4]), if parity error is detected and also drive the parity error signal to transceiver. In T = 1 mode, the receiver will generate the parity flag on PEF(SC0TSR[4]), if parity error detected, but doesn't drive the parity error signal to transceiver. **Note:** The description please see section 16.2.4 Error Signal and Character Repetition
5	RXBGTEN	Receiver Block Guard Time Function Enable Bit 0 = Receiver block guard time function Disabled. 1 = Receiver block guard time function Enabled.
4	CONSEL	Convention Selection 0 = Direct convention. 1 = Inverse convention. **Note1:** This bit is auto clear to "0", if AUTOCEN(SCnCR0[3]) is writing "1" **Note2:** If AUTOCEN(SCnCR0[3]) is enabled, hardware will decide the convention and change the CONSEL (SCnCR0[4]) bits automatically after SCEN (SCnCR0[0]) ="1".

Bit	Name	Description
3	AUTOCEN	Auto Convention Enable Bit 0 = Auto-convention Disabled. 1 = Auto-convention Enabled. When hardware receives TS in answer to reset state and the TS is direct convention, CONSEL(SCnCR0[4]) will be set to 0 automatically, otherwise if the TS is inverse convention, and CONSEL (SCnCR0[4]) will be set to 1. **Note:** If software enables auto convention function, the setting step must be done before Answer to Reset state and the first data must be 0x3B or 0x3F. After hardware received first data and stored it at buffer, hardware will decided the convention and change the CONSEL (SCnCR0[4]) bits automatically.
2	TXOFF	TX Transition Disable Bit 0 = The transceiver Enabled. 1 = The transceiver Disabled.
1	RXOFF	RX Transition Disable Bit 0 = The receiver Enabled. 1 = The receiver Disabled. **Note:** If AUTOCEN (SCnCR0[3])is enabled, these fields must be ignored.
0	SCEN	SC Engine Enable Bit Set this bit to 1 to enable SC operation. If this bit is cleared, SC will force all transition to IDLE state **Note:** SCEN must be set to 1 before filling in other registers, or smart card will not work properly.

SCnCR1 – SC Control Register

Register	SFR Address	Reset Value
SC0CR1	D7H, Page 0	0000_0000 b
SC1CR1	E7H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
OPE	PBOFF	WLS		TXDMAEN	RXDMAEN	CLKKEEP	UARTEN
R/W	R/W	R/W		R/W	R/W	R/W	R/W

Bit	Name	Description
7	OPE	Odd Parity Enable Bit 0 = Even number of logic 1's are transmitted or check the data word and parity bits in receiving mode. 1 = Odd number of logic 1's are transmitted or check the data word and parity bits in receiving mode. **Note:** This bit has effect only when PBOFF bit is '0'.

Bit	Name	Description
6	PBOFF	Parity Bit Disable Control 0 = Parity bit is generated or checked between the “last data word bit” and “stop bit” of the serial data. 1 = Parity bit is not generated (transmitting data) or checked (receiving data) during transfer. **Note:** In smart card mode, this field must be ‘0’ (default setting is with parity bit)
5:4	WLS	Word Length Selection 00 = Word length is 8 bits. 01 = Word length is 7 bits. 10 = Word length is 6 bits. 11 = Word length is 5 bits. **Note:** In smart card mode, this WLS must be ‘00’
3	TXDMAEN	SC/UART TX DMA enable This bit enables the SC/UART TX operating by through PDMA transfer, TX data needs to be ready in XRAM before SC/UART TX starting. 0 = SPI TX DMA Disabled 1 = SPI TX DMA Enabled
2	RXDMAEN	SC/UART RX DMA enable This bit enables the SC/UART RX operating by through PDMA transfer, RX data are saved in XRAM after SC/UART RX operation. 0 = SC/UART RX DMA Disabled 1 = SC/UART RX DMA Enabled
1	CLKKEEP	SC Clock Enable Bit 0 = SC clock generation Disabled. 1 = SC clock always keeps free running.
0	UARTEN	UART Mode Enable Bit 0 = Smart Card mode. 1 = UART mode. **Note1:**When operating in UART mode, user must set CONSEL (SCnCR0[4]) = 0 and AUTOCEN(SCnCR0[3]) = 0. **Note2:**When operating in Smart Card mode, user must set UARTEN(SCnCR1 [0]) = 0. **Note3:**When UART is enabled, hardware will generate a reset to reset FIFO and internal state machine.

SCnDR – SC Data Register

Register	SFR Address	Reset Value
SC0DR	D9H, Page 1	0000_0000 b
SC1DR	D9H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
SCnDR[7:0]							
R/W							

Bit	Name	Description
7:0	SCnDR[7:0]	SC / UART buffer data This byte is used for transmitting or receiving data on SC / UART bus. A write of this byte is a write to the shift register. A read of this byte is actually a read of the read data buffer. **Note:** If SCEN(SCnCR0[0]) is not enabled, SCnDR cannot be programmed.

SCnEGT – SC0~1 Extra Guard Time Register

Register	SFR Address	Reset Value
SC0EGT	DAH, Page 0	0000_0000 b
SC1EGT	DAH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
SCnEGT[7:0]							
R/W							

Bit	Name	Description
7:0	SCnEGT[7:0]	SC Extra Guard Time This field indicates the extra guard timer value. **Note:** The counter is ETU base .

SCnETURD0 – SCn ETU Rate Divider Register

Register	SFR Address	Reset Value
SC0ETURD0	DBH, Page 1	0111_0011 b
SC1ETURD0	DBH, Page 2	0111_0011 b

7	6	5	4	3	2	1	0
ETURDIV[7:0]							
R/W							

Bit	Name	Description
7:0	ETURDIV[7:0]	LSB bits of ETU Rate Divider The field indicates the LSB of clock rate divider. The real ETU is ETURDIV[11:0] + 1. **Note1:** ETURDIV[11:0] must be greater than 0x004. **Note2:** SCnETURD0 has to program first, then SCnETUDR2.

SCnETURD1 –SC ETU Rate Divider Register

Register	SFR Address	Reset Value

SC0ETURD1	DCH, Page 0	0011_0001 b
SC1ETURD1	DCH, Page 2	0011_0001 b

7	6	5	4	3	2	1	0
-	SCDIV[2:0]			ETURDIV[11:8]			
-	R/W			R/W			

Bit	Name	Description
7	-	Reserved
6:4	SCDIV [2:0]	SC clock divider 000 = F_{SC} is F_{SYS}/1. 001 = F_{SC} is F_{SYS}/2. 010 = F_{SC} is F_{SYS}/4. 011 = F_{SC} is F_{SYS}/8. (By default.) 100 = F_{SC} is F_{SYS}/16. 101 = F_{SC} is F_{SYS}/16. 110 = F_{SC} is F_{SYS}/16. 111 = F_{SC} is F_{SYS}/16. **Note:** that the F_{SC} clock should be 1Mhz ~ 5Mhz for ISO/IEC 7816-3 standard
3:0	ETURDIV [11:8]	MSB bits of ETU Rate Divider The field indicates the MSB of clock rate divider. The real ETU is ETURDIV[11:0] + 1. **Note1:** ETURDIV[11:0] must be greater than 0x004. **Note2:** SCnETURD0 has to program first, then SCnETUDR1 .

ScnIE – SC Interrupt Enable Control Register

Register	SFR Address	Reset Value
SC0IE	DDH, Page 0	0000_0000 b
SC1IE	DCH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	ACERRIEN	BGTIEN	TERRIEN	TBEIEN	RDAIEN
-	-	-	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:4	-	Reserved

Bit	Name	Description
4	ACERRIEN	Auto Convention Error Interrupt Enable Bit This field is used to enable auto-convention error interrupt. 0 = Auto-convention error interrupt Disabled. 1 = Auto-convention error interrupt Enabled.
3	BGTIEN	Block Guard Time Interrupt Enable Bit This field is used to enable block guard time interrupt. 0 = Block guard time interrupt Disabled. 1 = Block guard time interrupt Enabled.
2	TERRIEN	Transfer Error Interrupt Enable Bit This field is used to enable transfer error interrupt. The transfer error states is at SC0TSR register which includes receiver break error BEF(SC0TSR[6]), frame error FEF(SC0TSR[5]), parity error PEF(SC0TSR[4]), receiver buffer overflow error RXOV(SC0TSR[0]) and transmit buffer overflow error TXOV(SC0TSR[2]). 0 = Transfer error interrupt Disabled. 1 = Transfer error interrupt Enabled.
1	TBEIEN	Transmit Buffer Empty Interrupt Enable Bit This field is used to enable transmit buffer empty interrupt. 0 = Transmit buffer empty interrupt Disabled. 1 = Transmit buffer empty interrupt Enabled.
0	RDAIEN	Receive Data Reach Interrupt Enable Bit This field is used to enable received data interrupt. 0 = Receive data interrupt Disabled. 1 = Receive data interrupt Enabled.

ScnIS – SC Interrupt Status Register

Register	SFR Address	Reset Value
SC0IS	DEH, Page 0	0000_0010 b
SC1IS	DEH, Page 2	0000_0010 b

7	6	5	4	3	2	1	0
-	-	-	ACERRIF	BGTIF	TERRIF	TBEIF	RDAIF
-	-	-	R/W	R/W	R	R	R

Bit	Name	Description
7:5	-	Reserved
4	ACERRIF	Auto Convention Error Interrupt Status Flag (Read Only) This field indicates auto convention sequence error. If the received TS at ATR state is neither 0x3B nor 0x3F, this bit will be set. **Note:** This bit is read only, but it can be cleared by writing “0” to it.

Bit	Name	Description
3	BGTIF	Block Guard Time Interrupt Status Flag (Read Only) This field is used for block guard time interrupt status flag. **Note1:** This bit is valid when RXBGTEN (SCnCR0[5]) is enabled. **Note2:** This bit is read only, but it can be cleared by writing “0” to it.
2	TERRIF	Transfer Error Interrupt Status Flag (Read Only) This field is used for transfer error interrupt status flag. The transfer error states is at SC0TSR register which includes receiver break error BEF(SC0TSR[6]), frame error FEF(SC0TSR[5]), parity error PEF(SC0TSR[4]) and receiver buffer overflow error RXOV(SC0TSR[0]) and transmit buffer overflow error TXOV(SC0TSR[2]). **Note:** This field is the status flag of BEF(SC0TSR[6]), FEF(SC0TSR[5]), PEF(SC0TSR[4]), RXOV(SC0TSR[0]) and TXOV(SC0TSR[2]). So, if software wants to clear this bit, software must write “0” to each field.
1	TBEIF	Transmit Buffer Empty Interrupt Status Flag (Read Only) This field is used for transmit buffer empty interrupt status flag. **Note:** This field is the status flag of transmit buffer empty state. If software wants to clear this bit, software must write data to DAT(SCnDR[7:0]) buffer and then this bit will be cleared automatically.
0	RDAIF	Receive Data Reach Interrupt Status Flag (Read Only) This field is used for received data interrupt status flag. **Note:** This field is the status flag of received data. If software reads data from SC_DAT pin, this bit will be cleared automatically.

SCnTSR – SC Transfer Status Register

Register	SFR Address	Reset Value
SC0TSR	DFH, Page 0	0000_1010 b
SC1TSR	DFH, Page 2	0000_1010 b

7	6	5	4	3	2	1	0
ACT	BEF	FEF	PEF	TXEMPTY	TXOV	RXEMPTY	RXOV
R	R/W	R/W	R/W	R	R/W	R	R/W

Bit	Name	Description
7	ACT	Transmit /Receive in Active Status Flag (Read Only) 0 = This bit is cleared automatically when TX/RX transfer is finished 1 = This bit is set by hardware when TX/RX transfer is in active.
6	BEF	Receiver Break Error Status Flag (Read Only) This bit is set to logic 1 whenever the received data input (RX) held in the “spacing state” (logic 0) is longer than a full word transmission time (that is, the total time of “start bit” + data bits + parity + stop bits). . **Note:** This bit is read only, but it can be cleared by writing 0 to it.

Bit	Name	Description
5	FET	Receiver Frame Error Status Flag (Read Only) This bit is set to logic 1 whenever the received character does not have a valid “stop bit” (that is, the stop bit following the last data bit or parity bit is detected as logic 0). **Note:** This bit is read only, but it can be cleared by writing 0 to it.
4	PEF	Receiver Parity Error Status Flag (Read Only) This bit is set to logic 1 whenever the received character does not have a valid “parity bit”. **Note:** This bit is read only, but it can be cleared by writing 0 to it.
3	TXEMPTY	Transmit Buffer Empty Status Flag (Read Only) This bit indicates TX buffer empty or not. **Note:** When TX buffer has been transferred to Transmitter Shift Register, hardware sets this bit high. It will be cleared when writing data into DAT(SCnDR[7:0]) (TX buffer not empty).
2	TXOVF	TX Overflow Error Interrupt Status Flag (Read Only) If TX buffer is full, an additional write to DAT(SCnDR[7:0]) will cause this bit be set to “1” by hardware. **Note:** This bit is read only, but it can be cleared by writing 0 to it.
1	RXEMPTY	Receiver Buffer Empty Status Flag(Read Only) This bit indicates RX buffer empty or not. **Note:** When Rx buffer has been read by CPU, hardware sets this bit high. It will be cleared when SC receives any new data.
0	RXOV	RX Overflow Error Status Flag (Read Only) This bit is set when RX buffer overflow. **Note:** This bit is read only, but it can be cleared by writing 0 to it.

17 SERIAL PERIPHERAL INTERFACE (SPI)

The ML51 provides two Serial Peripheral Interface (SPI) block to support high-speed serial communication. SPI is a full-duplex, high-speed, synchronous communication bus between microcontrollers or other peripheral devices such as serial EEPROM, LCD driver, or D/A converter. It provides either Master or Slave mode, high-speed rate up to $F_{SYS}/4$, transfer complete and write collision flag. For a multi-master system, SPI supports Master Mode Fault to protect a multi-master conflict.

Following is the SPI multi function pin define list:

Group	Pin Name	GPIO	MFP	Type	Description
SPI0	SPI0_CLK	P0.2	MFP3	I/O	SPI0 serial clock pin.
		P6.2	MFP4	I/O	
	SPI0_MISO	P0.1	MFP3	I/O	SPI0 MISO (Master In, Slave Out) pin.
		P6.1	MFP4	I/O	
	SPI0_MOSI	P0.0	MFP3	I/O	SPI0 MOSI (Master Out, Slave In) pin.
		P6.0	MFP4	I/O	
	SPI0_SS	P0.3	MFP3	I/O	SPI0 slave select pin.
		P6.3	MFP4	I/O	
SPI1	SPI1_CLK	P0.2	MFP4	I/O	SPI1 serial clock pin.
		P3.2	MFP4	I/O	
	SPI1_MISO	P0.1	MFP4	I/O	SPI1 MISO (Master In, Slave Out) pin.
		P3.1	MFP4	I/O	
	SPI1_MOSI	P3.7	MFP5	I/O	SPI1 MOSI (Master Out, Slave In) pin.
		P0.0	MFP4	I/O	
	SPI1_SS	P0.3	MFP4	I/O	SPI1 slave select pin.
		P3.3	MFP4	I/O	

17.1 SPI Block Diagram

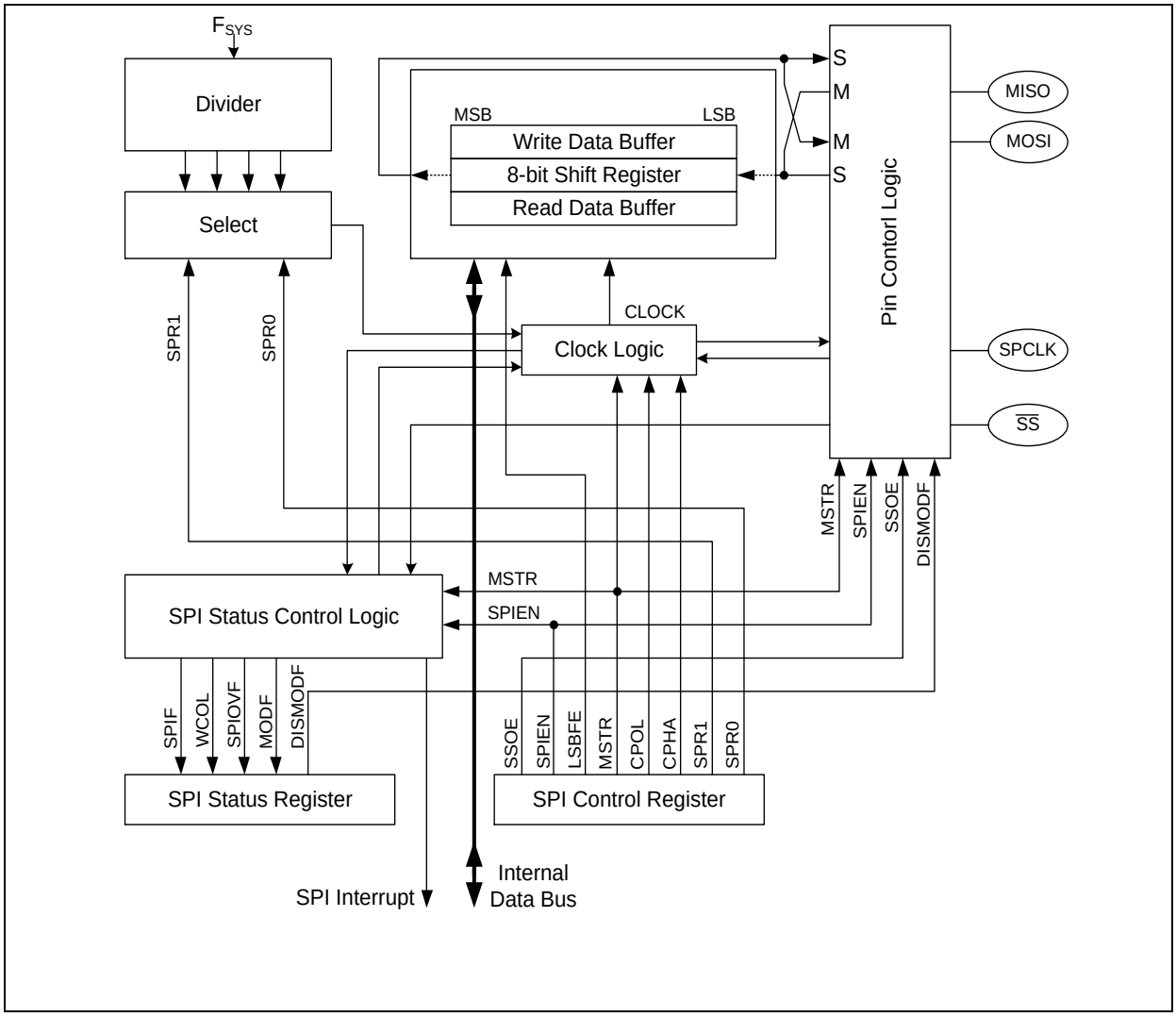


Figure 17.1-1 SPI Block Diagram

17.2 Functional Description

SPI block diagram provides an overview of SPI architecture in this device. The main blocks of SPI are the SPI control register logic, SPI status logic, clock rate control logic, and pin control logic. For a serial data transfer or receiving, The SPI block exists a shift register and a read data buffer. It is double buffered in the receiving and transmit directions. Transmit data cannot be written to the shifter until the previous transfer is complete. Receiving logic consists of parallel read data buffer so the shift register is free to accept a second data, as the first received data will be transferred to the read data buffer.

The 2 set, 16 pins of SPI interface and 4 pins of SPI0 interface which are Master-In/Slave-Out (MISO), Master-Out/Slave-In (MOSI), Shift Clock (SPCLK), and Slave Select ($\overline{SS}$). The MOSI pin is used to transfer a 8-bit data in series from the Master to the Slave. Therefore, MOSI is an output pin for Master device and an input for Slave. Respectively, the MISO is used to receive a serial data from the Slave to the Master.

The SPCLK pin is the clock output in Master mode, but is the clock input in Slave mode. The shift clock is used to synchronize the data movement both in and out of the devices through their MOSI and MISO pins. The shift clock is driven by the Master mode device for eight clock cycles. Eight clocks exchange one byte data on the serial lines. For the shift clock is always produced out of the Master device, the system should never exist more than one device in Master mode for avoiding device conflict.

Each Slave peripheral is selected by one Slave Select pin ($\overline{SS}$). The signal should stay low for any Slave access. When $\overline{SS}$ is driven high, the Slave device will be inactivated. If the system is multi-slave, there should be only one Slave device selected at the same time. In the Master mode MCU, the $\overline{SS}$ pin does not function and it can be configured as a general purpose I/O. However, $\overline{SS}$ can be used as Master Mode Fault detection (see Section 17.2.4"Mode Fault Detection") via software setting if multi-master environment exists. The ML51 also provides auto-activating function to toggle $\overline{SS}$ between each byte-transfer.

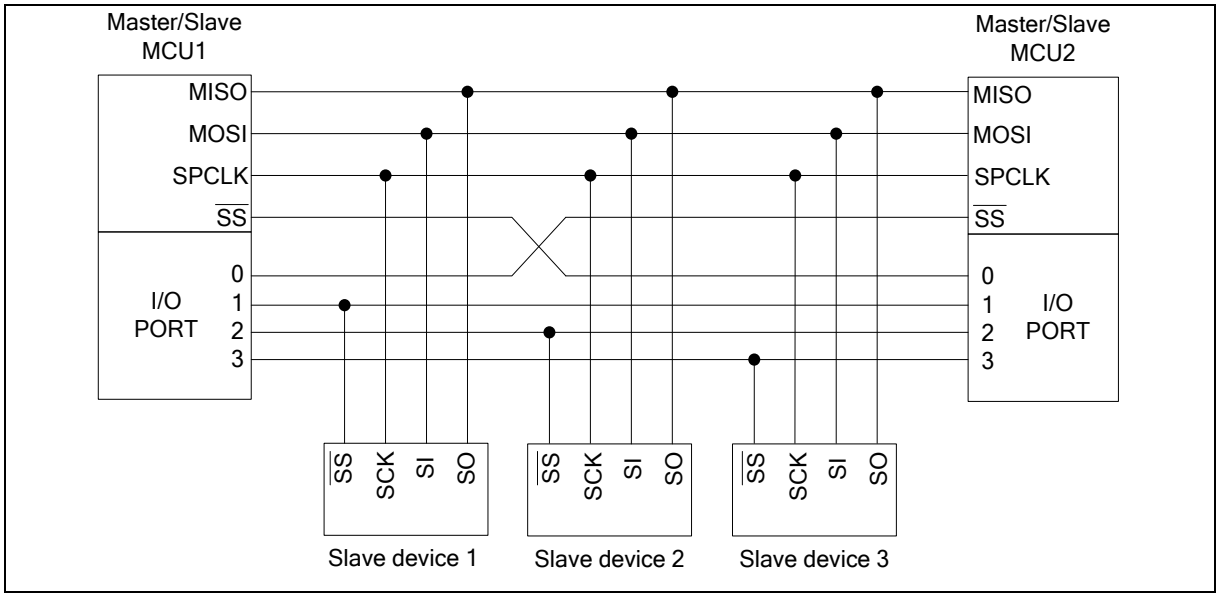


Figure 17.2-1 SPI Multi-Master, Multi-Slave Interconnection

Figure 17.2-1 shows a typical interconnection of SPI devices. The bus generally connects devices together through three signal wires, MOSI to MOSI, MISO to MISO, and SPCLK to SPCLK. The Master devices select the individual Slave devices by using four pins of a parallel port to control the four $\overline{SS}$ pins. MCU1 and MCU2 play either Master or Slave mode. The $\overline{SS}$ should be configured as Master Mode Fault detection to avoid multi-master conflict.

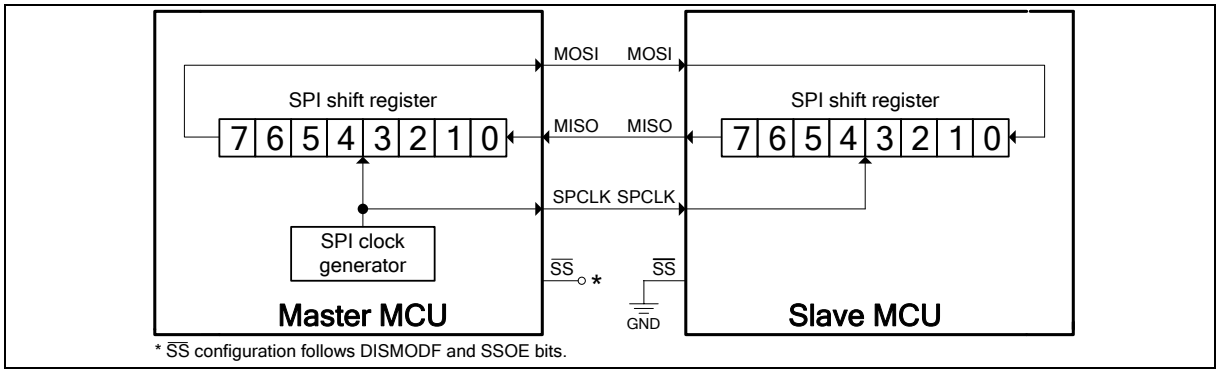


Figure 17.2-2 SPI Single-Master, Single-Slave Interconnection

Figure 17.2-2 shows the simplest SPI system interconnection, single-master and signal-slave. During a transfer, the Master shifts data out to the Slave via MOSI line. While simultaneously, the Master shifts data in from the Slave via MISO line. The two shift registers in the Master MCU and the Slave MCU can be considered as one 16-bit circular shift register. Therefore, while a transfer data pushed from Master into Slave, the data in Slave will also be pulled in Master device respectively. The transfer effectively exchanges the data, which was in the SPI shift registers of the two MCUs.

By default, SPI data is transferred MSB first. If the LSBFE (SPInCR.5) is set, SPI data shifts LSB first. This bit does not affect the position of the MSB and LSB in the data register. Note that all the following description and figures are under the condition of LSBFE logic 0. MSB is transmitted and received first.

There are three SPI registers to support its operations, including SPI control register (SPInCR), SPI status register (SPInSR), and SPI data register (SPInDR). These registers provide control, status, data storage functions, and clock rate selection. The following registers relate to SPI function.

17.2.1 Operating Modes

17.2.1.1 Master Mode

The SPI can operate in Master mode while MSTR (SPInCR.4) is set as 1. Only one Master SPI device can initiate transmissions. A transmission always begins by Master through writing to SPInDR. The byte written to SPInDR begins shifting out on MOSI pin under the control of SPCLK. Simultaneously, another byte shifts in from the Slave on the MISO pin. After 8-bit data transfer complete, SPIF (SPInSR.7) will automatically set via hardware to indicate one byte data transfer complete. At the same time, the data received from the Slave is also transferred in SPInDR. User can clear SPIF and read data out of SPInDR.

17.2.1.2 Slave Mode

When MSTR is 0, the SPI operates in Slave mode. The SPCLK pin becomes input and it will be clocked by another Master SPI device. The $\overline{SS}$ pin also becomes input. The Master device cannot exchange data with the Slave device until the $\overline{SS}$ pin of the Slave device is externally pulled low. Before data transmissions occurs, the $\overline{SS}$ of the Slave device should be pulled and remain low until the transmission is complete. If $\overline{SS}$ goes high, the SPI is forced into idle state. If the $\overline{SS}$ is forced to high at the middle of transmission, the transmission will be aborted and the rest bits of the receiving shifter buffer will be high and goes into idle state.

In Slave mode, data flows from the Master to the Slave on MOSI pin and flows from the Slave to the Master on MISO pin. The data enters the shift register under the control of the SPCLK from the Master device. After one byte is received in the shift register, it is immediately moved into the read data buffer and the SPIF bit is set. A read of the SPInDR is actually a read of the read data buffer. To prevent an overrun and the loss of the byte that caused by the overrun, the Slave should read SPInDR out and the first SPIF should be cleared before a second transfer of data from the Master device comes in the

read data buffer.

17.2.2 Clock Formats and Data Transfer

To accommodate a wide variety of synchronous serial peripherals, the SPI has a clock polarity bit CPOL (SPInCR.3) and a clock phase bit CPHA (SPInCR.2). Figure 17.2-3 SPI Clock Formats shows that CPOL and CPHA compose four different clock formats. The CPOL bit denotes the SPCLK line level in its idle state. The CPHA bit defines the edge on which the MOSI and MISO lines are sampled. The CPOL and CPHA should be identical for the Master and Slave devices on the same system. To Communicate in different data formats with one another will result undetermined result.

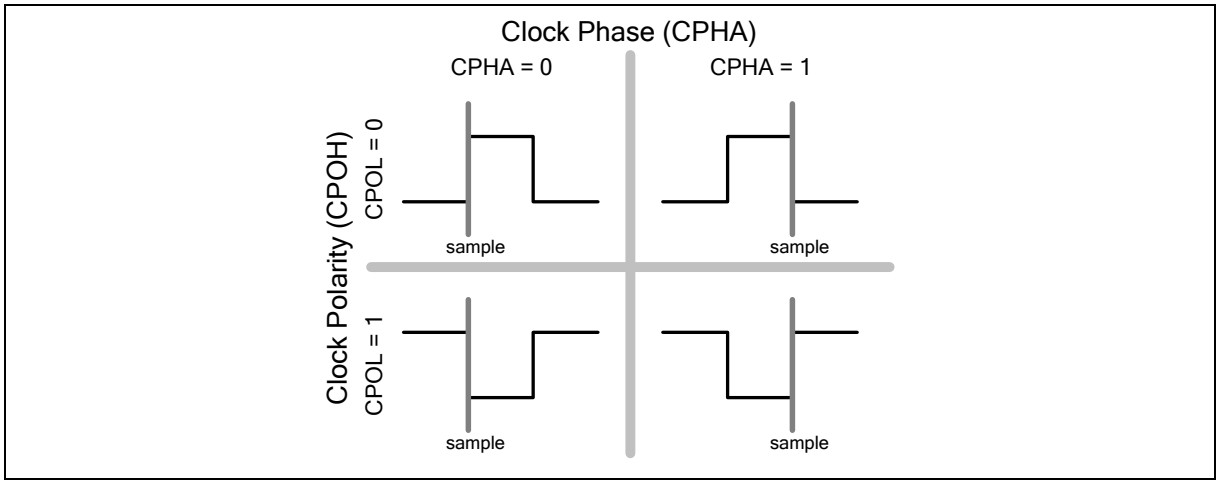


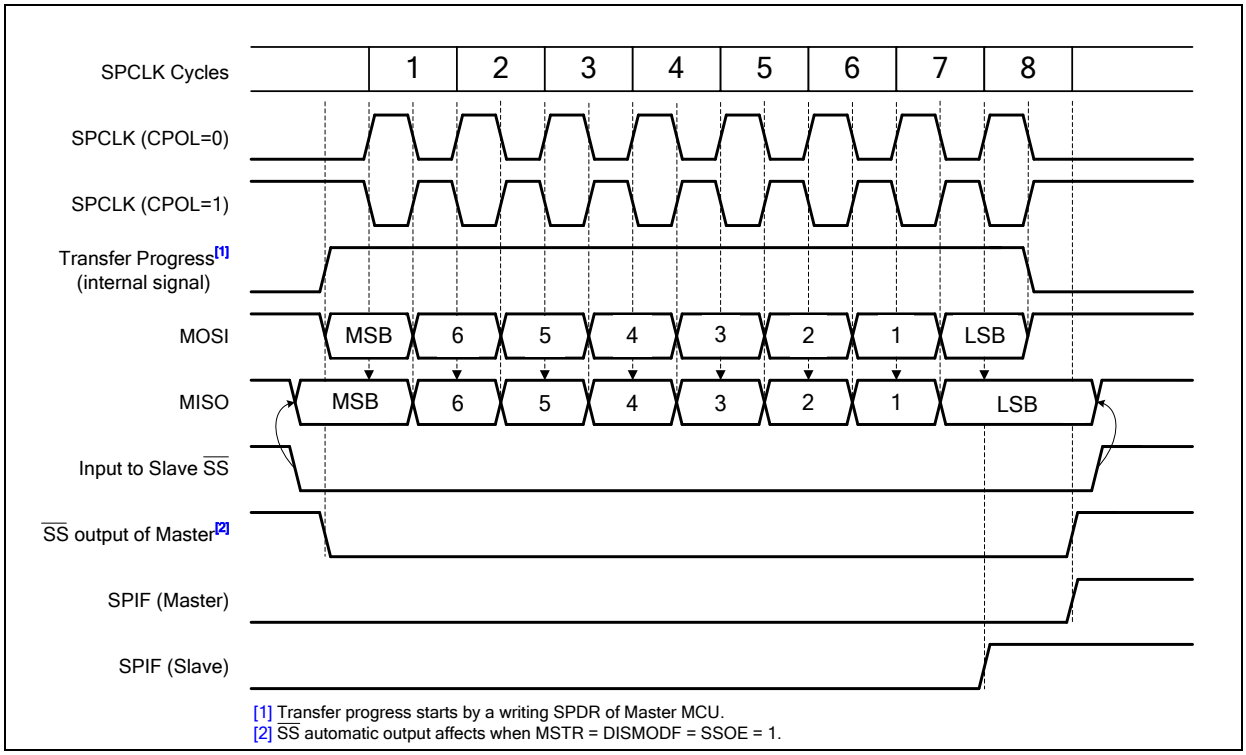
Figure 17.2-3 SPI Clock Formats

In SPI, a Master device always initiates the transfer. If SPI is selected as Master mode (MSTR = 1) and enabled (SPIEN = 1), writing to the SPI data register (SPInDR) by the Master device starts the SPI clock and data transfer. After shifting one byte out and receiving one byte in, the SPI clock stops and SPIF (SPInSR.7) is set in both Master and Slave. If SPI interrupt enable bit is set 1 and global interrupt is enabled (EA = 1), the interrupt service routine (ISR) of SPI will be executed.

Concerning the Slave mode, the $\overline{SS}$ signal needs to be taken care. As shown in Figure 17.2-3 SPI Clock Formats, when CPHA = 0, the first SPCLK edge is the sampling strobe of MSB (for an example of LSBFE = 0, MSB first). Therefore, the Slave should shift its MSB data before the first SPCLK edge. The falling edge of $\overline{SS}$ is used for preparing the MSB on MISO line. The $\overline{SS}$ pin therefore should toggle high and then low between each successive serial byte. Furthermore, if the slave writes data to the SPI data register (SPInDR) while $\overline{SS}$ is low, a write collision error occurs.

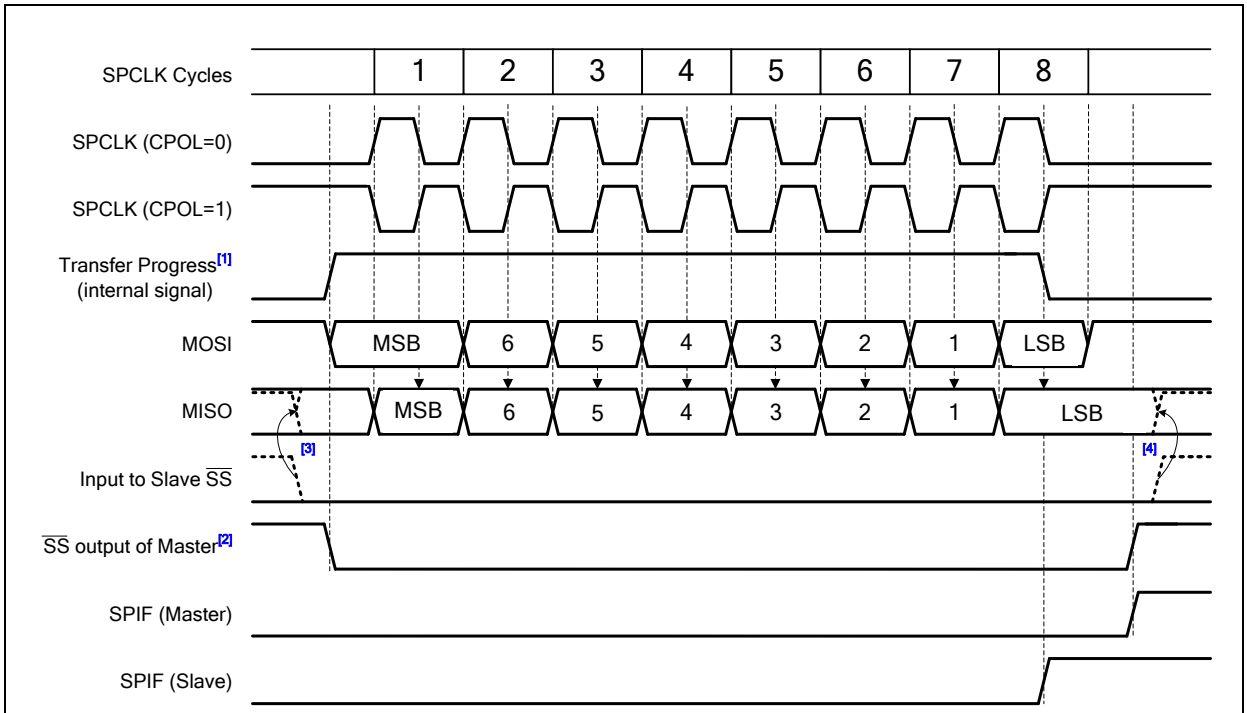
When CPHA = 1, the sampling edge thus locates on the second edge of SPCLK clock. The Slave uses the first SPCLK clock to shift MSB out rather than the $\overline{SS}$ falling edge. Therefore, the $\overline{SS}$ line can remain low between successive transfers. This format may be preferred in systems having single fixed Master and single fixed Slave. The $\overline{SS}$ line of the unique Slave device can be tied to GND as long as only CPHA = 1 clock mode is used.

Note: The SPI should be configured before it is enabled (SPIEN = 1), or a change of LSBFE, MSTR, CPOL, CPHA and SPR[1:0] will abort a transmission in progress and force the SPI system into idle state. Prior to any configuration bit changed, SPIEN must be disabled first.



[1] Transfer progress starts by a writing SPDR of Master MCU.
[2] SS automatic output affects when MSTR = DISMODF = SSOE = 1.

Figure 17.2-4 SPI Clock and Data Format with CPHA = 0



[1] Transfer progress starts by a writing SPDR of Master MCU.
[2] SS automatic output affects when DISMODF = SSOE = MSTR = 1.
[3] If SS of Slave is low, the MISO will be the LSB of previous data. Otherwise, MISO will be high.
[4] While SS stays low, the LSB will last its state. Once SS is released to high, MISO will switch to high level.

Figure 17.2-5 SPI Clock and Data Format with CPHA = 1

17.2.3 Slave Select Pin Configuration

The ML51 SPI gives a flexible $\overline{SS}$ pin feature for different system requirements. When the SPI operates as a Slave, $\overline{SS}$ pin always rules as Slave select input. When the Master mode is enabled, $\overline{SS}$ has three different functions according to DISMODF (SPInSR.3) and SSOE (SPInCR.7). By default, DISMODF is 0. It means that the Mode Fault detection activates. $\overline{SS}$ is configured as a input pin to check if the Mode Fault appears. On the contrary, if DISMODF is 1, Mode Fault is inactivated and the SSOE bit takes over to control the function of the $\overline{SS}$ pin. While SSOE is 1, it means the Slave select signal will generate automatically to select a Slave device. The $\overline{SS}$ as output pin of the Master usually connects with the $\overline{SS}$ input pin of the Slave device. The $\overline{SS}$ output automatically goes low for each transmission when selecting external Slave device and goes high during each idle state to de-select the Slave device. While SSOE is 0 and DISMODF is 1, $\overline{SS}$ is no more used by the SPI and reverts to be a general purpose I/O pin.

17.2.4 Mode Fault Detection

The Mode Fault detection is useful in a system where more than one SPI devices might become Masters at the same time. It may induce data contention. When the SPI device is configured as a Master and the $\overline{SS}$ input line is configured for Mode Fault input depending on SPInCR0, a Mode Fault error occurs once the $\overline{SS}$ is pulled low by others. It indicates that some other SPI device is trying to address this Master as if it is a Slave. Instantly the MSTR and SPIEN control bits in the SPInCR are cleared via hardware to disable SPI, Mode Fault flag MODF (SPInSR.4) is set and an interrupt is generated if ESPI and EA are enabled.

17.2.5 Write Collision Error

The SPI is signal buffered in the transfer direction and double buffered in the receiving and transmit direction. New data for transmission cannot be written to the shift register until the previous transaction is complete. Write collision occurs while SPInDR be written more than once while a transfer was in progress. SPInDR is double buffered in the transmit direction. Any writing to SPInDR cause data to be written directly into the SPI shift register. Once a write collision error is generated, WCOL (SPInSR.6) will be set as 1 via hardware to indicate a write collision. In this case, the current transferring data continues its transmission. However the new data that caused the collision will be lost. Although the SPI logic can detect write collisions in both Master and Slave modes, a write collision is normally a Slave error because a Slave has no indicator when a Master initiates a transfer. During the receiving of Slave, a write to SPInDR causes a write collision in Slave mode. WCOL flag needs to be cleared via software.

17.2.6 Overrun Error

For receiving data, the SPI is double buffered in the receiving direction. The received data is transferred into a parallel read data buffer so the shifter is free to accept a second serial byte. However, the received data should be read from SPInDR before the next data has been completely shifted in. As long as the first byte is read out of the read data buffer and SPIF is cleared before the next byte is ready to be transferred, no overrun error condition occurs. Otherwise the overrun error occurs. In this condition, the second byte data will not be successfully received into the read data register and the previous data will remains. If overrun occur, SPIOVF (SPInSR.5) will be set via hardware. An SPIOVF setting will also require an interrupt if enabled. Figure 17.2-6 SPI Overrun Waveform shows the relationship between the data receiving and the overrun error.

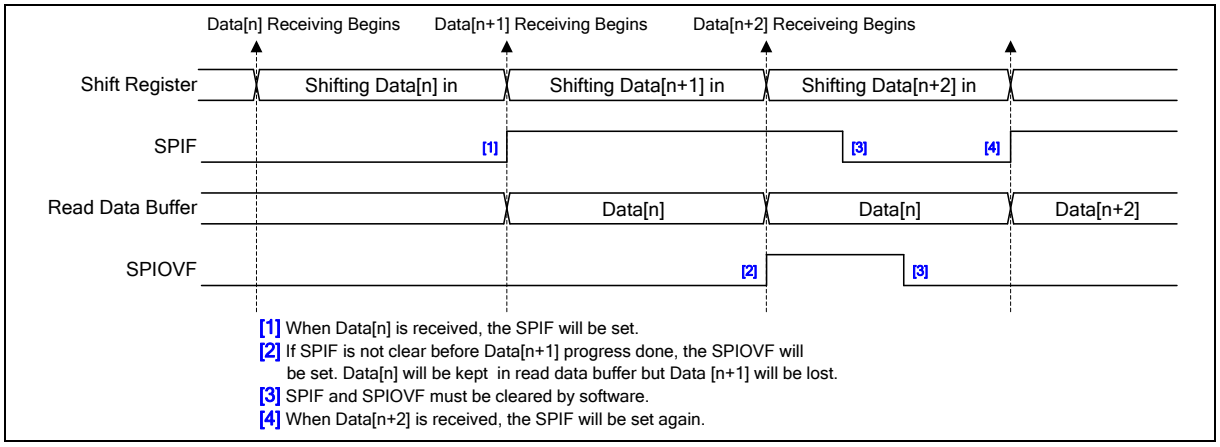


Figure 17.2-6 SPI Overrun Waveform

17.2.7 SPI Interrupt

Three SPI status flags, SPIF, MODF, and SPIOVF, can generate an SPI event interrupt requests. All of them locate in SPInSR. SPIF will be set after completion of data transfer with external device or a new data have been received and copied to SPInDR. MODF becomes set to indicate a low level on $\overline{SS}$ causing the Mode Fault state. SPIOVF denotes a receiving overrun error. If SPI interrupt mask is enabled via setting ESPI and EA is 1, CPU will executes the SPI interrupt service routine once any of these three flags is set. User needs to check flags to determine what event caused the interrupt. These three flags are software cleared.

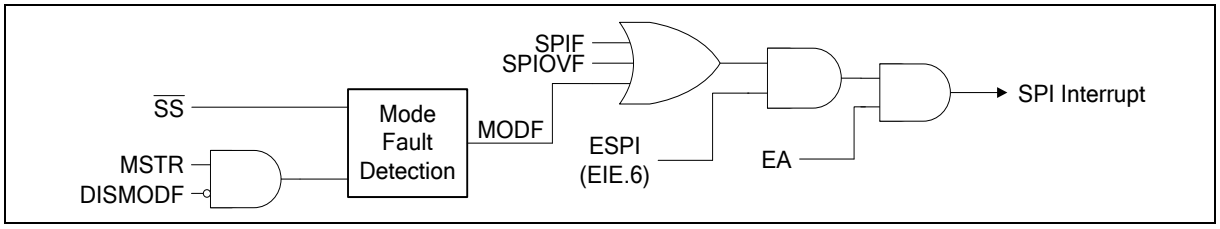


Figure 17.2-7 SPI Interrupt Request

17.3 Control Register Of SPI

Table 17.3-1 Slave Select Pin Configurations

DISMODF	SSOE	Master Mode (MSTR = 1)	Slave Mode (MSTR = 0)
0	X	$\overline{SS}$ input for Mode Fault	$\overline{SS}$ Input for Slave select
1	0	General purpose I/O	
1	1	Automatic $\overline{SS}$ output	

SPInCR0 – Serial Peripheral Control Register0

Register	SFR Address	Reset Value
SPI0CR0	F3H, Page 0	0000_0000 b
SPI1CR0	F9H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SSOE	SPIEN	LSBFE	MSTR	CPOL	CPHA	SPR1	SPR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	SSOE	**Slave select output enable** This bit is used in combination with the DISMODF (SPInSR.3) bit to determine the feature of $\overline{SS}$ pin as shown in Table 17–1. Slave Select Pin Configurations. This bit takes effect only under MSTR = 1 and DISMODF = 1 condition. 0 = $\overline{SS}$ functions as a general purpose I/O pin. 1 = $\overline{SS}$ automatically goes low for each transmission when selecting external Slave device and goes high during each idle state to de-select the Slave device.
6	SPIEN	**SPI enable** 0 = SPI function Disabled. 1 = SPI function Enabled.
5	LSBFE	**LSB first enable** 0 = The SPI data is transferred MSB first. 1 = The SPI data is transferred LSB first.
4	MSTR	**Master mode enable** This bit switches the SPI operating between Master and Slave modes. 0 = The SPI is configured as Slave mode. 1 = The SPI is configured as Master mode.
3	CPOL	**SPI clock polarity select** CPOL bit determines the idle state level of the SPI clock. See Figure 17.2-3 SPI Clock Formats. 0 = The SPI clock is low in idle state. 1 = The SPI clock is high in idle state.
2	CPHA	**SPI clock phase select** CPHA bit determines the data sampling edge of the SPI clock. See Figure 17.2-3 SPI Clock Formats. 0 = The data is sampled on the first edge of the SPI clock. 1 = The data is sampled on the second edge of the SPI clock.

Bit	Name	Description
1:0	SPR[1:0]	**SPI clock rate select** These two bits select four grades of SPI clock divider. The clock rates below are illustrated under F_{SYS} = 24 MHz condition. SPR3 SPR2 SPR1 SPR0 Divider SPI clock rate 0 0 0 0 2 12M bit/s 0 0 0 1 4 6M bit/s 0 0 1 0 8 3M bit/s 0 0 1 1 16 1.5M bit/s 0 1 0 0 32 750k bit/s 0 1 0 1 64 375k bit/s 0 1 1 0 128 187k bit/s 0 1 1 1 256 93.7k bit/s 1 0 0 0 3 8M bit/s 1 0 0 1 6 4M bit/s 1 0 1 0 12 2M bit/s 1 0 1 1 24 1M bit/s 1 1 0 0 48 500k bit/s 1 1 0 1 96 250k bit/s 1 1 1 0 192 125k bit/s 1 1 1 1 384 62.5k bit/s SPR[3:0] are valid only under Master mode (MSTR = 1). If under Slave mode, the clock will automatically synchronize with the external clock on SPICLK pin from Master device up to F_{SYS}/4 communication speed.

SPInCR1 – Serial Peripheral Control Register1

Register	SFR Address	Reset Value
SPI0CR1	F3H, Page 1	0000_0000 b
SPI1CR1	FAH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	SPR3	SPR2	TXDMAEN	RXDMAEN	SPIS1	SPIS0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:6	-	Reserved

<table>
<tr><th>Bit</th><th>Name</th><th>Description</th></tr>
<tr><td>5:4</td><td>SPR[3:2]</td><td>SPI clock rate select
These two bits select four grades of SPI clock divider. The clock rates below are illustrated under F_{SYS} = 24 MHz condition.
<table>
<tr><th>SPR3</th><th>SPR2</th><th>SPR1</th><th>SPR0</th><th>Divider</th><th>SPI clock rate</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>2</td><td>12M bit/s</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>4</td><td>6M bit/s</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>8</td><td>3M bit/s</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>16</td><td>1.5M bit/s</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>32</td><td>750k bit/s</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>64</td><td>375k bit/s</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>128</td><td>187k bit/s</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>256</td><td>93.7k bit/s</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>3</td><td>8M bit/s</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>6</td><td>4M bit/s</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>12</td><td>2M bit/s</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>24</td><td>1M bit/s</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>48</td><td>500k bit/s</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>96</td><td>250k bit/s</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>192</td><td>125k bit/s</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>384</td><td>62.5k bit/s</td></tr>
</table>
SPR[3:0] are valid only under Master mode (MSTR = 1). If under Slave mode, the clock will automatically synchronize with the external clock on SPICLK pin from Master device up to F_{SYS}/4 communication speed.</td></tr>
<tr><td>3</td><td>TXDMAEN</td><td>SPI TX DMA enable
This bit enables the SPI TX operating by through PDMA transfer, TX data needs to be ready in XRAM before SPI TX starting.
0 = SPI TX DMA Disabled
1 = SPI TX DMA Enabled</td></tr>
<tr><td>2</td><td>RXDMAEN</td><td>SPI RX DMA enable
This bit enables the SPI RX operating by through PDMA transfer, RX data are saved in XRAM after SPI RX operation.
0 = SPI RX DMA Disabled
1 = SPI RX DMA Enabled</td></tr>
<tr><td>1:0</td><td>SPIS[1:0]</td><td>SPI Interval time selection between adjacent bytes
SPIS[1:0] and CPHA select eight grades of SPI interval time selection between adjacent bytes. As below table:
<table>
<tr><th>CPHA</th><th>SPIS1</th><th>SPIS0</th><th>SPI clock</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0.0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0.5</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1.5</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>2.0</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0.0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1.0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>2.0</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>2.5</td></tr>
</table>
SPIS[1:0] are valid only under Master mode (MSTR = 1).</td></tr>
</table>

SPInSR – Serial Peripheral Status Register

Register	SFR Address	Reset Value
SPI0SR	F4H, Page 0	0000_0000 b
SPI1SR	FBH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SPIF	WCOL	SPIOVF	MODF	DISMODF	DISSPIF	TXBFF	-
R/W	R/W	R/W	R/W	R/W	R/W	R	-

Bit	Name	Description
7	SPIF	SPI complete flag This bit is set to logic 1 via hardware while an SPI data transfer is complete or an receiving data has been moved into the SPI read buffer. If ESPI and EA are enabled, an SPI interrupt will be required. This bit should be cleared via software. Attempting to write to SPInDR is inhibited if SPIF is set.
6	WCOL	Write collision error flag This bit indicates a write collision event. Once a write collision event occurs, this bit will be set. It should be cleared via software.
5	SPIOVF	SPI overrun error flag This bit indicates an overrun event. Once an overrun event occurs, this bit will be set. If ESPI and EA are enabled, an SPI interrupt will be required. This bit should be cleared via software.
4	MODF	Mode Fault error flag This bit indicates a Mode Fault error event. If $\overline{SS}$ pin is configured as Mode Fault input (MSTR = 1 and DISMODF = 0) and $\overline{SS}$ is pulled low by external devices, a Mode Fault error occurs. Instantly MODF will be set as logic 1. If ESPI and EA are enabled, an SPI interrupt will be required. This bit should be cleared via software.
3	DISMODF	Disable Mode Fault error detection This bit is used in combination with the SSOE (SPInCR.7) bit to determine the feature of $\overline{SS}$ pin as shown in Table 17–1. Slave Select Pin Configurations. DISMODF is valid only in Master mode (MSTR = 1). 0 = Mode Fault detection Enabled. $\overline{SS}$ serves as input pin for Mode Fault detection disregard of SSOE. 1 = Mode Fault detection Disabled. The feature of $\overline{SS}$ follows SSOE bit.
2	DISSPIF	Disable SPI complete interrupt This bit is used to disable SPI complete interrupt while an SPI data transfer is complete or an receiving data has been moved into the SPI read buffer. Especially in SPI DMA operation. 0 = SPI Complete Interrupt Enabled while ESPI and EA are enabled, 1 = SPI Complete Interrupt Disabled
1	TXBFF	SPI TX Buffer Full Flag 0 = SPI TX buffer is empty 1 = SPI TX buffer is full

SPInDR – Serial Peripheral Data Register

Register	SFR Address	Reset Value
SPI0DR	F5H, Page 0	0000_0000 b
SPI1DR	FCH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
SPInDR[7:0]							
R/W							

Bit	Name	Description
7:0	SPInDR[7:0]	Serial peripheral data This byte is used for transmitting or receiving data on SPI bus. A write of this byte is a write to the shift register. A read of this byte is actually a read of the read data buffer. In Master mode, a write to this register initiates transmission and reception of a byte simultaneously.

18 INTER-INTEGRATED CIRCUIT (I^2C)

The ML51 provides two Inter-Integrated Circuit (I^2C) bus to serves as an serial interface between the microcontrollers and the I^2C devices such as EEPROM, LCD module, temperature sensor, and so on. The I^2C bus used two wires design (a serial data line SDA and a serial clock line SCL) to transfer information between devices.

The I^2C bus uses bi-directional data transfer between masters and slaves. There is no central master and the multi-master system is allowed by arbitration between simultaneously transmitting masters. The serial clock synchronization allows devices with different bit rates to communicate via one serial bus. The I^2C bus supports four transfer modes including master transmitter, master receiver, slave receiver, and slave transmitter. The I^2C interface only supports 7-bit addressing mode. A special mode General Call is also available. The I^2C can meet both standard (up to 100kbps) and fast (up to 400k bps) speeds.

Folloing is the I^2C multi function pin define list:

Group	Pin Name	GPIO	MFP	Type	Description
I2C0	I2C0_SCL	P2.5	MFP6	I/O	I2C0 clock pin.
		P5.3	MFP4	I/O	
		P0.5	MFP9	I/O	
		P4.1	MFP9	I/O	
	I2C0_SDA	P2.4	MFP6	I/O	I2C0 data input/output pin.
		P5.2	MFP4	I/O	
		P0.4	MFP9	I/O	
		P4.0	MFP9	I/O	
I2C1	I2C1_SCL	P2.3	MFP4	I/O	I2C1 clock pin.
		P2.1	MFP9	I/O	
		P0.7	MFP8	I/O	
		P0.3	MFP9	I/O	
		P5.0	MFP3	I/O	
		P4.5	MFP9	I/O	
		P1.4	MFP4	I/O	
	I2C1_SDA	P2.2	MFP4	I/O	I2C1 data input/output pin.
		P2.0	MFP9	I/O	
		P0.6	MFP8	I/O	
		P0.2	MFP9	I/O	
		P5.1	MFP3	I/O	
		P4.4	MFP9	I/O	
		P1.5	MFP4	I/O	

18.1 Functional Description

For a bi-directional transfer operation, the SDA and SCL pins should be open-drain pads. This implements a wired-AND function, which is essential to the operation of the interface. A low level on a I²C bus line is generated when one or more I²C devices output a “0”. A high level is generated when all I²C devices output “1”, allowing the pull-up resistors to pull the line high. In ML51, user should set output latches of SCL and SDA. As logic 1 before enabling the I²C function by setting I²CEN.

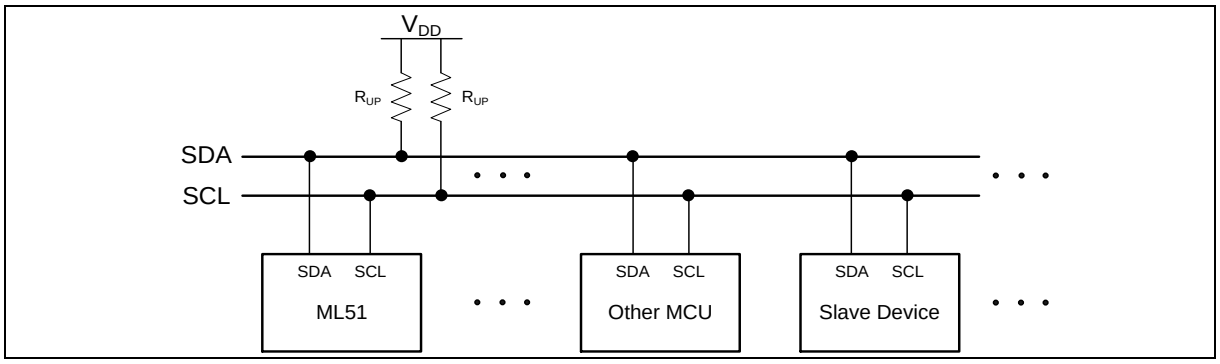


Figure 18.1-1 I²C Bus Interconnection

The I²C is considered free when both lines are high. Meanwhile, any device, which can operate as a master can occupy the bus and generate one transfer after generating a START condition. The bus now is considered busy before the transfer ends by sending a STOP condition. The master generates all of the serial clock pulses and the START and STOP condition. However if there is no START condition on the bus, all devices serve as not addressed slave. The hardware looks for its own slave address or a General Call address. (The General Call address detection may be enabled or disabled by GC (I²CnADDRx.0).) If the matched address is received, an interrupt is requested.

Every transaction on the I²C bus is 9 bits long, consisting of 8 data bits (MSB first) and a single acknowledge bit. The number of bytes per transfer (defined as the time between a valid START and STOP condition) is unrestricted but each byte has to be followed by an acknowledge bit. The master device generates 8 clock pulse to send the 8-bit data. After the 8th falling edge of the SCL line, the device outputting data on the SDA changes that pin to an input and reads in an acknowledge value on the 9th clock pulse. After 9th clock pulse, the data receiving device can hold SCL line stretched low if next receiving is not prepared ready. It forces the next byte transaction suspended. The data transaction continues when the receiver releases the SCL line.

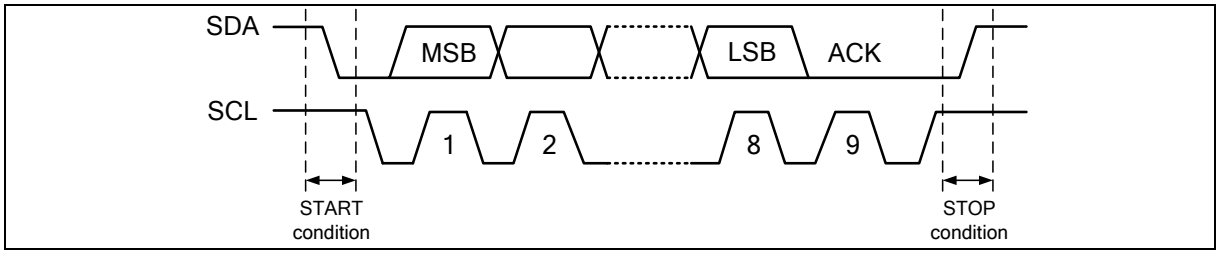


Figure 18.1-2 I²C Bus Protocol

18.1.1 START and STOP Condition

The protocol of the I²C bus defines two states to begin and end a transfer, START (S) and STOP (P) conditions. A START condition is defined as a high-to-low transition on the SDA line while SCL line is high. The STOP condition is defined as a low-to-high transition on the SDA line while SCL line is high. A START or a STOP condition is always generated by the master and I²C bus is considered busy after a START condition and free after a STOP condition. After issuing the STOP condition successful, the

original master device will release the control authority and turn back as a not addressed slave. Consequently, the original addressed slave will become a not addressed slave. The I^{2}C bus is free and listens to next START condition of next transfer.

A data transfer is always terminated by a STOP condition generated by the master. However, if a master still wishes to communicate on the bus, it can generate a repeated START (Sr) condition and address the pervious or another slave without first generating a STOP condition. Various combinations of read/write formats are then possible within such a transfer.

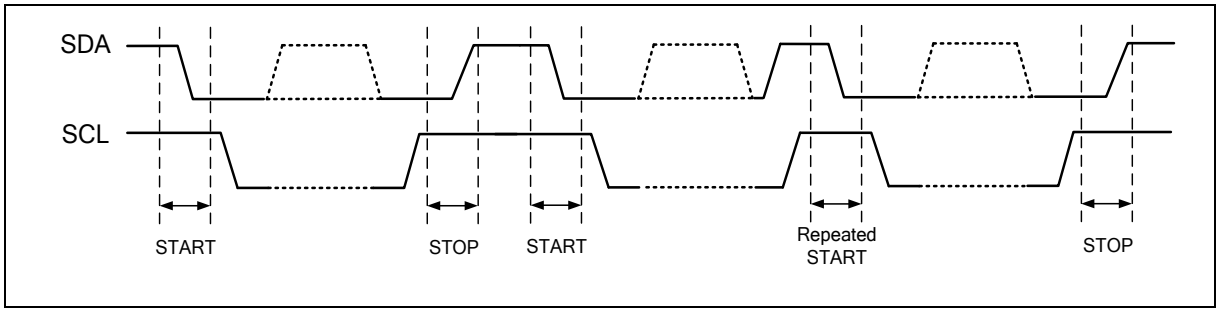


Figure 18.1-3 START, Repeated START, and STOP Conditions

18.1.2 7-Bit Address with Data Format

Following the START condition is generated, one byte of special data should be transmitted by the master. It includes a 7-bit long slave address (SLA) following by an 8th bit, which is a data direction bit (R/W), to address the target slave device and determine the direction of data flow. If R/W bit is 0, it indicates that the master will write information to a selected slave. Also, if R/W bit is 1, it indicates that the master will read information from the addressed slave. An address packet consisting of a slave address and a read I or a write (W) bit is called SLA+R or SLA+W, respectively. A transmission basically consists of a START condition, a SLA+W/R, one or more data packets and a STOP condition. After the specified slave is addressed by SLA+W/R, the second and following 8-bit data bytes issue by the master or the slave devices according to the R/W bit configuration.

Figure 18.1-4 shows a master transmits data to slave by 7-bit. A master addresses a slave with a 7-bit address and 1-bit write index to denote that the master wants to transmit data to the slave. The master keeps transmitting data after the slave returns acknowledge to the master.

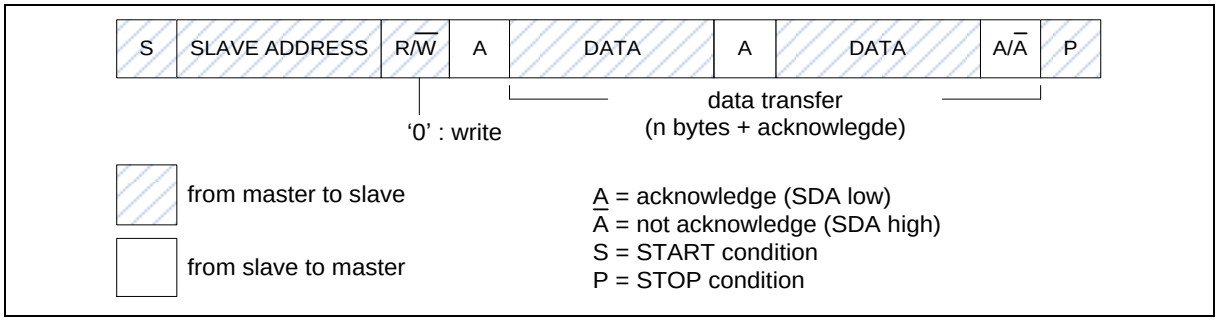


Figure 18.1-4 Master Transmits Data to Slave by 7-bit

Figure 18.1-5 shows a master read data from slave by 7-bit. A master addresses a slave with a 7-bit address and 1-bit read index to denote that the master wants to read data from the slave. The slave will start transmitting data after the slave returns acknowledge to the master.

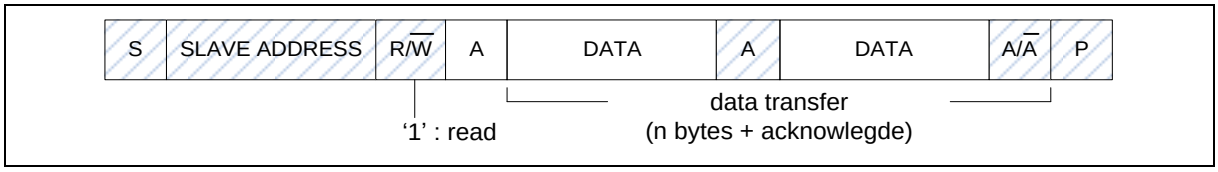


Figure 18.1-5 Master Reads Data from Slave by 7-bit

There is an exception called "General Call" address, which can address all devices by giving the first byte of data all 0. A General Call is used when a master wishes to transmit the same message to several slaves in the system. When this address is used, other devices may respond with an acknowledge or ignore it according to individual software configuration. If a device response the General Call, it operates as like in the slave-receiver mode. Note that the address 0x00 is reserved for General Call and cannot be used as a slave address, therefore, in theory, a 7-bit addressing I^2C bus accepts 127 devices with their slave addresses 1 to 127.

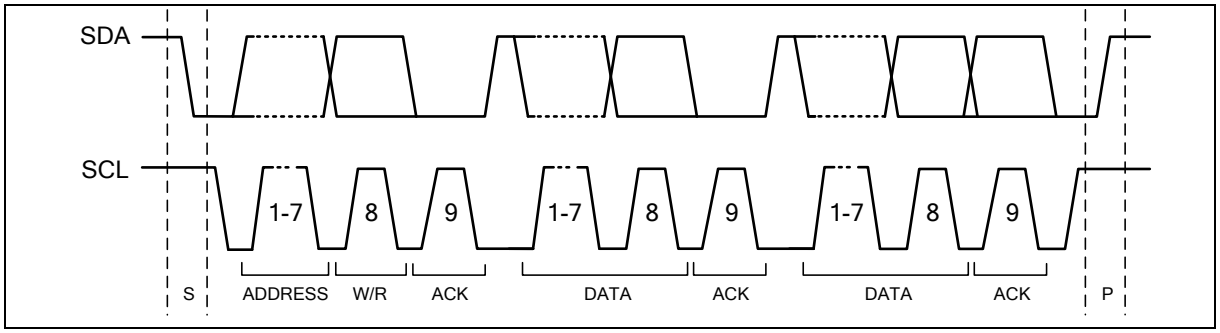


Figure 18.1-6 Data Format of One I^2C Transfer

During the data transaction period, the data on the SDA line should be stable during the high period of the clock, and the data line can only change when SCL is low.

18.1.3 Acknowledge

The 9th SCL pulse for any transferred byte is dedicated as an Acknowledge (ACK). It allows receiving devices (which can be the master or slave) to respond back to the transmitter (which also can be the master or slave) by pulling the SDA line low. The acknowledge-related clock pulse is generated by the master. The transmitter should release control of SDA line during the acknowledge clock pulse. The ACK is an active-low signal, pulling the SDA line low during the clock pulse high duty, indicates to the transmitter that the device has received the transmitted data. Commonly, a receiver, which has been addressed is requested to generate an ACK after each byte has been received. When a slave receiver does not acknowledge (NACK) the slave address, the SDA line should be left high by the slave so that the mater can generate a STOP or a repeated START condition.

If a slave-receiver does acknowledge the slave address, it switches itself to not addressed slave mode and cannot receive any more data bytes. This slave leaves the SDA line high. The master should generate a STOP or a repeated START condition.

If a master-receiver is involved in a transfer, because the master controls the number of bytes in the transfer, it should signal the end of data to the slave-transmitter by not generating an acknowledge on the last byte. The slave-transmitter then switches to not addressed mode and releases the SDA line to allow the master to generate a STOP or a repeated START condition.

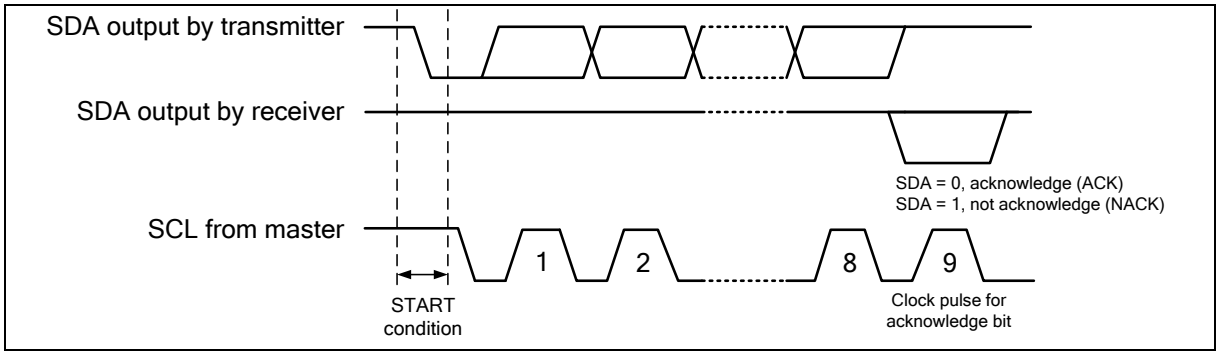


Figure 18.1-7 Acknowledge Bit

18.1.4 Arbitration

A master may start a transfer only if the bus is free. It is possible for two or more masters to generate a START condition. In these situations, an arbitration scheme takes place on the SDA line, while SCL is high. During arbitration, the first of the competing master devices to place‘a’’1’ (high) on SDA while another master transmits‘a’’0’ (low) switches off its data output stage because the level on the bus does not match its own level. The arbitration lost master switches to the not addressed slave immediately to detect its own slave address in the same serial transfer whether it is being addressed by the winning master. It also releases SDA line to high level for not affecting the data transfer continued by the winning master. However, the arbitration lost master continues generating clock pulses on SCL line until the end of the byte in which it loses the arbitration.

Arbitration is carried out by all masters continuously monitoring the SDA line after outputting data. If the value read from the SDA line does not match the value that the master has to output, it has lost the arbitration. Note that a master can only lose arbitration when it outputs a high SDA value while another master outputs a low value. Arbitration will continue until only one master remains, and this may take many bits. Its first stage is a comparison of address bits, and if both masters are trying to address the same device, arbitration continues on to the comparison of data bits or acknowledge bit.

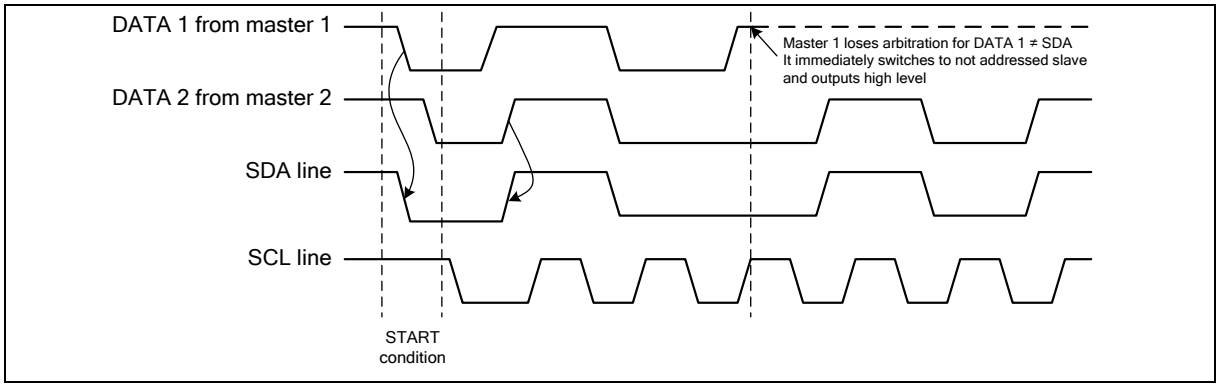


Figure 18.1-8 Arbitration Procedure of Two Masters

Since control of the I^2C bus is decided solely on the address or master code and data sent by competing masters, there is no central master, nor any order of priority on the bus. Slaves are not involved in the arbitration procedure.

18.1.5 Operation Modes

The on-chip I^2C ports support three operation modes, Master, Slave, and General Call Mode.

In a given application, I^2C port may operate as a master or as a slave. In Slave mode, the I^2C port hardware looks for its own slave address and the general call address. If one of these addresses is detected, and if the slave is willing to receive or transmit data from/to master(by setting the AA bit),

acknowledge pulse will be transmitted out on the 9th clock, hence an interrupt is requested on both master and slave devices if interrupt is enabled. When the microcontroller wishes to become the bus master, hardware waits until the bus is free before entering Master mode so that a possible slave action is not be interrupted. If bus arbitration is lost in Master mode, I²C port switches to Slave mode immediately and can detect its own slave address in the same serial transfer.

To control the I²C bus transfer in each mode, user needs to set I²C_CTL0, I²C_DAT registers according to current status code of I²C_STATUS0 register. In other words, for each I²C bus action, user needs to check current status by I²C_STATUS0 register, and then set I²C_CTL0, I²C_DAT registers to take bus action. Finally, check the response status by I²C_STATUS0.

The bits, STA, STO and AA in I²C_CTL0 register are used to control the next state of the I²C hardware after SI flag of I²C_CTL0 [3] register is cleared. Upon completion of the new action, a new status code will be updated in I²C_STATUS0 register and the SI flag of I²C_CTL0 register will be set. But the SI flag will not be set when I²C STOP. If the I²C interrupt control bit INTEN (I²C_CTL0 [7]) is set, appropriate action or software branch of the new status code can be performed in the Interrupt service routine.

Figure 18.1-9 Control I2C Bus according to the Current I2C Status shows the current I²C status code is 0x08, and then set I²C_DATA=SLA+W and (STA,STO,SI,AA) = (0,0,1,x) to send the address to I²C bus. If a slave on the bus matches the address and response ACK, the I²C_STATUS0 will be updated by status code 0x18.

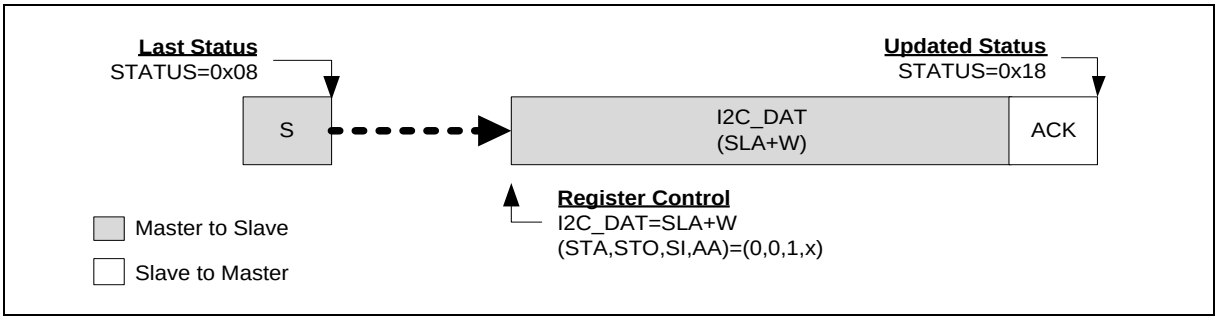


Figure 18.1-9 Control I²C Bus according to the Current I²C Status

Master Transmitter Mode

In the master transmitter mode, several bytes of data are transmitted to a slave receiver. The master should prepare by setting desired clock rate in I²CnCLK. The master transmitter mode may now be entered by setting STA (I²CnCON.5) bit as 1. The hardware will test the bus and generate a START condition as soon as the bus becomes free. After a START condition is successfully produced, the SI flag (I²CnCON.3) will be set and the status code in I²CnSTAT show 08H. The progress is continued by loading I²CnDAT with the target slave address and the data direction bit “write” (SLA+W). The SI bit should then be cleared to commence SLA+W transaction.

After the SLA+W byte has been transmitted and an acknowledge (ACK) has been returned by the addressed slave device, the SI flag is set again and I²CnSTAT is read as 18H. The appropriate action to be taken follows user defined communication protocol by sending data continuously. After all data is transmitted, the master can send a STOP condition by setting STO (I²CnCON.4) and then clearing SI to terminate the transmission. A repeated START condition can also be generated without sending STOP condition to immediately initial another transmission.

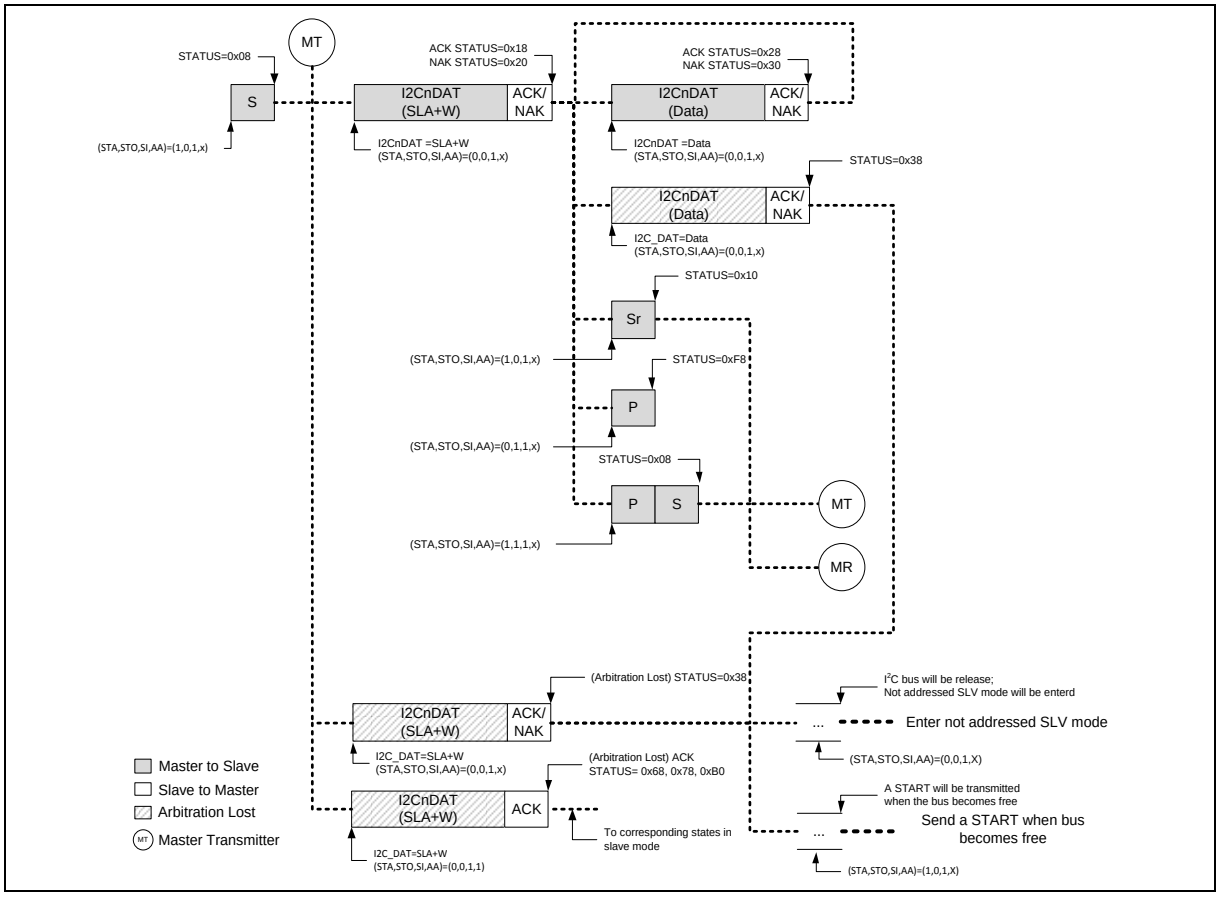


Figure 18.1-10 Flow and Status of Master Transmitter Mode

Master Receiver Mode

In the master receiver mode, several bytes of data are received from a slave transmitter. The transaction is initialized just as the master transmitter mode. Following the START condition, I^2CnDAT should be loaded with the target slave address and the data direction bit “read” (SLA+R). After the SLA+R byte is transmitted and an acknowledge bit has been returned, the SI flag is set again and $I^2CnSTAT$ is read as 40H. SI flag then should be cleared to receive data from the slave transmitter. If AA flag ($I^2CnCON.2$) is set, the master receiver will acknowledge the slave transmitter. If AA is cleared, the master receiver will not acknowledge the slave and release the slave transmitter as a not addressed slave. After that, the master can generate a STOP condition or a repeated START condition to terminate the transmission or initial another one.

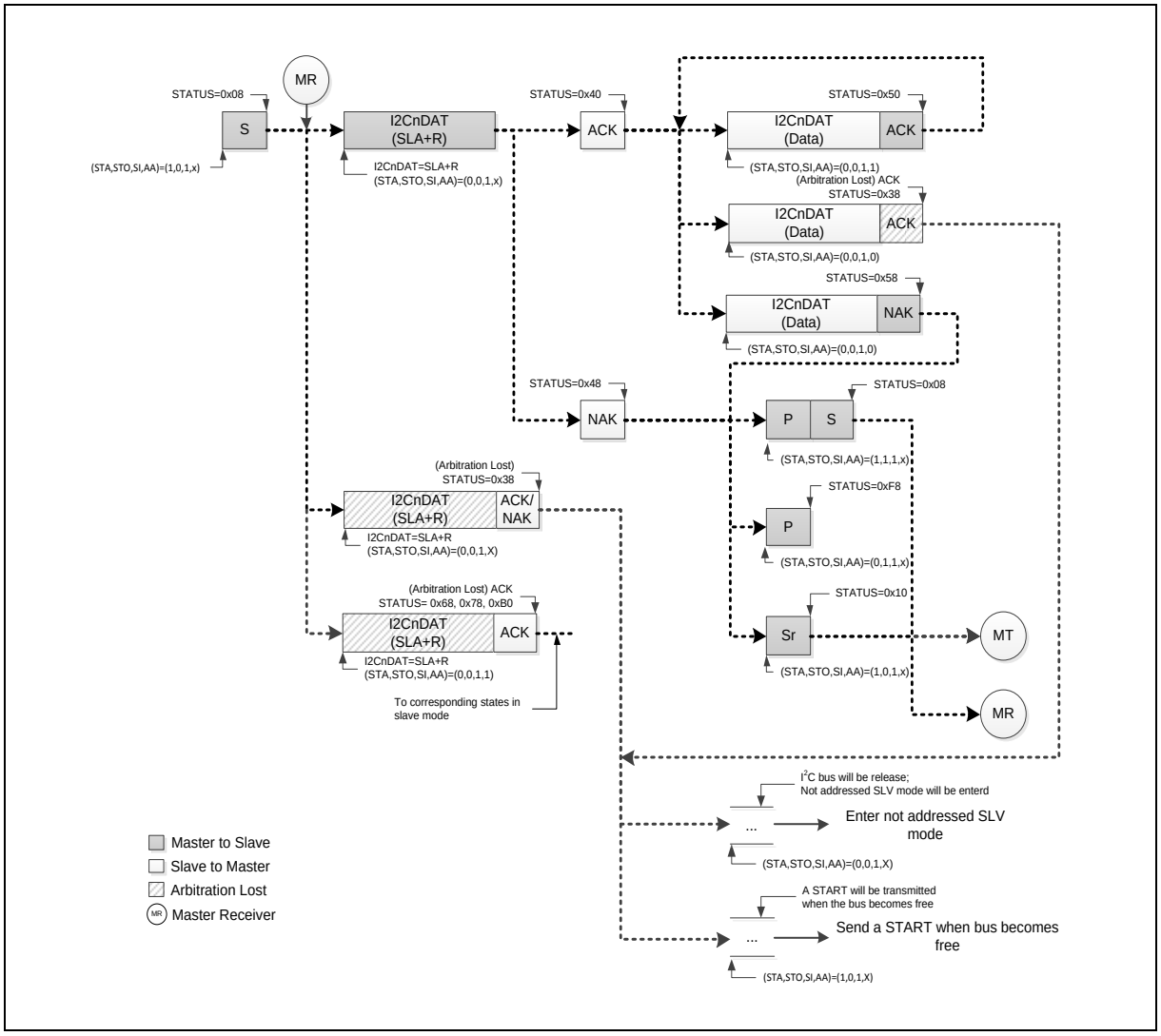


Figure 18.1-11 Flow and Status of Master Receiver Mode

18.1.6 Slave mode

Slave Receiver

In the slave receiver mode, several bytes of data are received form a master transmitter. Before a transmission is commenced, I^{2}CnADDRx should be loaded with the address to which the device will respond when addressed by a master. I^{2}CnCLK does not affect in slave mode. The AA bit should be set to enable acknowledging its own slave address. After the initialization above, the I^{2}C idles until it is addressed by its own address with the data direction bit “write” (SLA+W). The slave receiver mode may also be entered if arbitration is lost.

After the slave is addressed by SLA+W, it should clear its SI flag to receive the data from the master transmitter. If the AA bit is 0 during a transaction, the slave will return a non-acknowledge after the next received data byte. The slave will also become not addressed and isolate with the master. It cannot receive any byte of data with I^{2}CnDAT remaining the previous byte of data, which is just received.

Slave Transmitter

The I^{2}C port is equipped with four slave address registers, I^{2}CnADDRx (x=0~3). The contents of the

register are irrelevant when I²C is in Master mode. In the slave transmitter mode, several bytes of data are transmitted to a master receiver. After I²CnADDRx and I²CnCON values are given, the I²C wait until it is addressed by its own address with the data direction bit “read” (SLA+R). The slave transmitter mode may also be entered if arbitration is lost.

After the slave is addressed by SLA+R, it should clear its SI flag to transmit the data to the master receiver. Normally the master receiver will return an acknowledge after every bytes of data is transmitted by the slave. If the acknowledge is not received, it will transmit all “1” data if it continues the transaction. It becomes a not addressed slave. If the AA flag is cleared during a transaction, the slave transmits the last byte of data. The next transmitting data will be all “1” and the slave becomes not addressed.

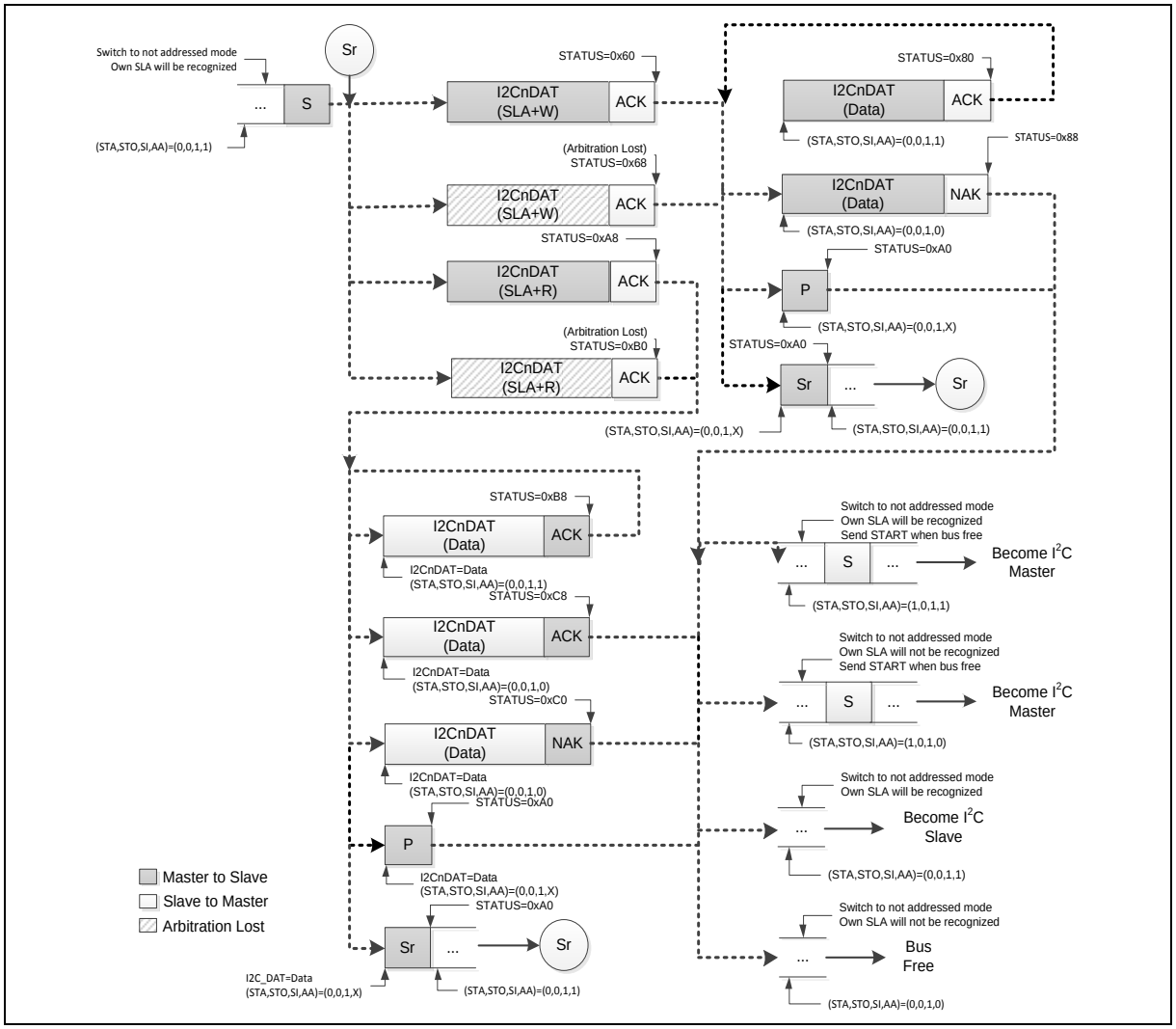


Figure 18.1-12 Flow and Status of Slave Receiver Mode

18.1.7 General Call

The General Call is a special condition of slave receiver mode by been addressed with all “0” data in slave address with data direction bit. Both GC (I²CnADDRx.0) bit and AA bit should be set as 1 to enable acknowledging General Calls. The slave addressed by a General Call has different status code in I²CnSTAT with normal slave receiver mode. The General Call may also be produced if arbitration is lost.

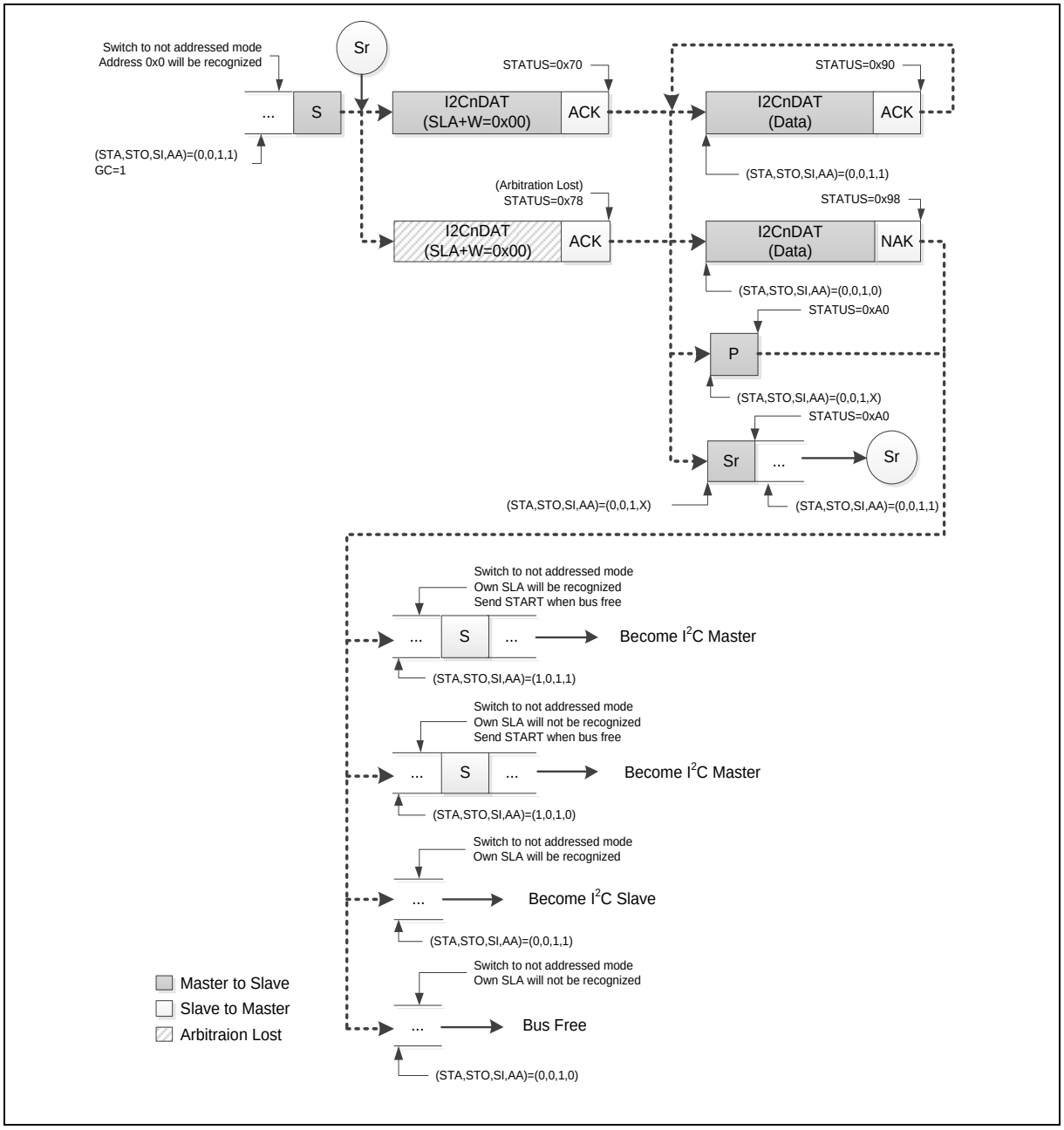


Figure 18.1-13 Flow and Status of General Call Mode

18.1.8 Miscellaneous States

There are two I²CnSTAT status codes that do not correspond to the 25 defined states, The first status code F8H indicates that no relevant information is available during each transaction. Meanwhile, the SI flag is 0 and no I²C interrupt is required. The other status code 00H means a bus error has occurred during a transaction. A bus error is caused by a START or STOP condition appearing temporally at an illegal position such as the second through eighth bits of an address or a data byte, and the acknowledge bit. When a bus error occurs, the SI flag is set immediately. When a bus error is detected on the I²C bus, the operating device immediately switches to the not addressed salve mode, releases SDA and SCL lines, sets the SI flag, and loads I²CnSTAT as 00H. To recover from a bus error, the STO bit should be set and then SI should be cleared. After that, STO is cleared by hardware and release the I²C bus without issuing a real STOP condition waveform on I²C bus.

There is a special case if a START or a repeated START condition is not successfully generated for I²C bus is obstructed by a low level on SDA line e.g. a slave device out of bit synchronization, the problem can be solved by transmitting additional clock pulses on the SCL line. The I²C hardware transmits additional clock pulses when the STA bit is set, but no START condition can be generated because the SDA line is pulled low. When the SDA line is eventually released, a normal START condition is transmitted, state 08H is entered, and the serial transaction continues. If a repeated START condition is transmitted while SDA is obstructed low, the I²C hardware also performs the same action as above. In this case, state 08H is entered instead of 10H after a successful START condition is transmitted. Note that the software is not involved in solving these bus problems.

The following table is show the status display in I2STAT register of I²C number and description:

Master Mode		**Slave Mode**	
STATUS	Description	STATUS	Description
0x08	Start	0Xa0	Slave Transmit Repeat Start or Stop
0x10	Master Repeat Start	0Xa8	Slave Transmit Address ACK
0x18	Master Transmit Address ACK	0Xb0	Slave Transmit Arbitration Lost
0x20	Master Transmit Address NACK	0Xb8	Slave Transmit Data ACK
0x28	Master Transmit Data ACK	0Xc0	Slave Transmit Data NACK
0x30	Master Transmit Data NACK	0Xc8	Slave Transmit Last Data ACK
0x38	Master Arbitration Lost	0x60	Slave Receive Address ACK
0x40	Master Receive Address ACK	0x68	Slave Receive Arbitration Lost
0x48	Master Receive Address NACK	0x80	Slave Receive Data ACK
0x50	Master Receive Data ACK	0x88	Slave Receive Data NACK
0x58	Master Receive Data NACK	0x70	GC mode Address ACK
0x00	Bus error	0x78	GC mode Arbitration Lost
		0x90	GC mode Data ACK
		0x98	GC mode Data NACK
0xF8	Bus Released **Note:** Status “0xF8” exists in both master/slave modes, and it won’t raise interrupt.		

18.1.9 I²C Time-Out

There is a 14-bit time-out counter, which can be used to deal with the I²C bus hang-up. If the time-out counter is enabled, the counter starts up counting until it overflows. Meanwhile I2TOF will be set by hardware and requests I²C interrupt. When time-out counter is enabled, setting flag SI to high will reset counter and restart counting up after SI is cleared. If the I²C bus hangs up, it causes the SI flag not set for a period. The 14-bit time-out counter will overflow and require the interrupt service.

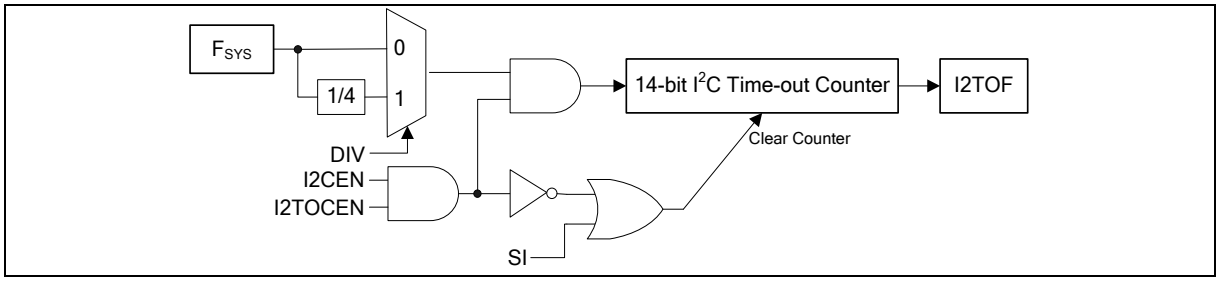


Figure 18.1-14 I²C Time-Out Counter

18.1.10 I²C Interrupt

There are two I²C flags, SI and I2TOF. Both of them can generate an I²C event interrupt requests. If I²C interrupt mask is enabled via setting EI²C and EA as 1, CPU will execute the I²C interrupt service routine once any of these two flags is set. User needs to check flags to determine what event caused the interrupt. Both of I²C flags are cleared by software.

18.2 Control Registers of I²C

There are five control registers to interface the I²C bus including I²CnCON, I²CnSTAT, I²CnDAT, I²CnADDRx, and I²CnCLK. These registers provide protocol control, status, data transmitting and receiving functions, and clock rate configuration. For application flexibility. The following registers relate to I²C function.

I²CnCON – I²C Control

Register	SFR Address	Reset Value
I2C0CON	C0H, Page 0, Bit-addressable	0000_0000 b
I2C1CON	E8H, Page 0, Bit-addressable	0000_0000 b

7	6	5	4	3	2	1	0
I	I²CEN	STA	STO	SI	AA	-	-
R/W	R/W	R/W	R/W	R/W	R/W	-	-

Bit	Name	Description
7	I	I²C0 hold time extend enable 0 = I²C DATA to SCL hold time extend disabled 1 = I²C DATA to SCL hold time extend enabled, extend 8 system clock
6	I²CEN	I²C0 bus enable 0 = I²C bus Disabled. 1 = I²C bus Enabled. Before enabling the I²C, SCL and SDA port latches should be set to logic 1.
5	STA	START flag When STA is set, the I²C generates a START condition if the bus is free. If the bus is busy, the I²C waits for a STOP condition and generates a START condition following. If STA is set while the I²C is already in the master mode and one or more bytes have been transmitted or received, the I²C generates a repeated START condition. Note that STA can be set anytime even in a slave mode, but STA is not hardware automatically cleared after START or repeated START condition has been detected. User should take care of it by clearing STA manually.
4	STO	STOP flag When STO is set if the I²C is in the master mode, a STOP condition is transmitted to the bus. STO is automatically cleared by hardware once the STOP condition has been detected on the bus. The STO flag setting is also used to recover the I²C device from the bus error state (I²CnSTAT as 00H). In this case, no STOP condition is transmitted to the I²C bus. If the STA and STO bits are both set and the device is original in the master mode, the I²C bus will generate a STOP condition and immediately follow a START condition. If the device is in slave mode, STA and STO simultaneous setting should be avoid from issuing illegal I²C frames.

Bit	Name	Description
3	SI	I²C0 interrupt flag SI flag is set by hardware when one of 26 possible I²C status (besides F8H status) is entered. After SI is set, the software should read I²CnSTAT register to determine which step has been passed and take actions for next step. SI is cleared by software. Before the SI is cleared, the low period of SCL line is stretched. The transaction is suspended. It is useful for the slave device to deal with previous data bytes until ready for receiving the next byte. The serial transaction is suspended until SI is cleared by software. After SI is cleared, I²C bus will continue to generate START or repeated START condition, STOP condition, 8-bit data, or so on depending on the software configuration of controlling byte or bits. Therefore, user should take care of it by preparing suitable setting of registers before SI is software cleared.
2	AA	Acknowledge assert flag If the AA flag is set, an ACK (low level on SDA) will be returned during the acknowledge clock pulse of the SCL line while the I²C device is a receiver or an own-address-matching slave. If the AA flag is cleared, a NACK (high level on SDA) will be returned during the acknowledge clock pulse of the SCL line while the I²C device is a receiver or an own-address-matching slave. A device with its own AA flag cleared will ignore its own salve address and the General Call. Consequently, SI will note be asserted and no interrupt is requested. Note that if an addressed slave does not return an ACK under slave receiver mode or not receive an ACK under slave transmitter mode, the slave device will become a not addressed slave. It cannot receive any data until its AA flag is set and a master addresses it again. There is a special case of I²CnSTAT value C8H occurs under slave transmitter mode. Before the slave device transmit the last data byte to the master, AA flag can be cleared as 0. Then after the last data byte transmitted, the slave device will actively switch to not addressed slave mode of disconnecting with the master. The further reading by the master will be all FFH.
1:0	-	Reserved

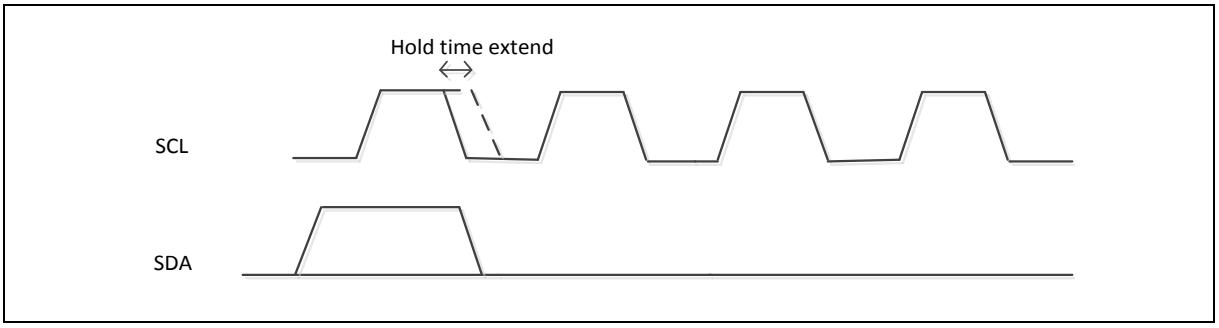


Figure 18.2 Hold Time extend enable

I²CnSTAT – I²C Status

Register	SFR Address	Reset Value
I²C0STAT	BDH, Page 0	1111_1000 b
I²C1STAT	B4H, Page 0	1111_1000 b

7	6	5	4	3	2	1	0
I²C0STAT[7:3]					0	0	0
R					R	R	R

Bit	Name	Description
7:3	I²C0STAT7:3]	I²C0 status code The MSB five bits of I²CnSTAT contains the status code. There are 27 possible status codes. When I²CnSTAT is F8H, no relevant state information is available and SI flag keeps 0. All other 26 status codes correspond to the I²C states. When each of these status is entered, SI will be set as logic 1 and a interrupt is requested.
2:0	0	Reserved The least significant three bits of I²CnSTAT are always read as 0.

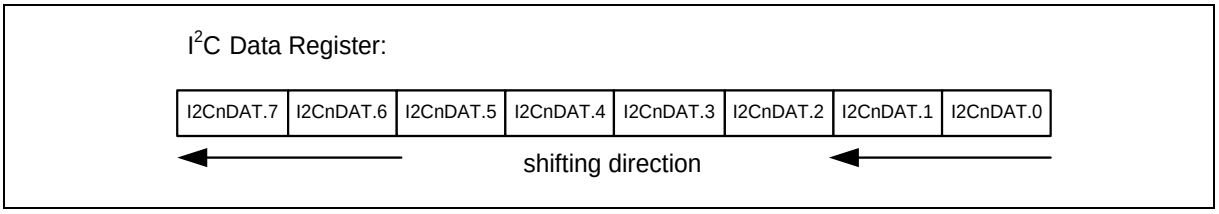
I²CnDAT – I²C Data

Register	SFR Address	Reset Value
I²C0DAT	BCH, Page 0	0000_0000 b
I²C1DAT	B3H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
I²C0DAT[7:0]							
R/W							

Bit	Name	Description
7:0	I²CnDAT[7:0]	I²C0 data I²CnDAT contains a byte of the I²C data to be transmitted or a byte, which has just received. Data in I²CnDAT remains as long as SI is logic 1. The result of reading or writing I²CnDAT during I²C transceiver progress is unpredicted. While data in I²CnDAT is shifted out, data on the bus is simultaneously being shifted in to update I²CnDAT. I²CnDAT always shows the last byte that presented on the I²C bus. Thus the event of lost arbitration, the original value of I²CnDAT changes after the transaction.

I²C Data Shifting Direction.



I²CnADDRx – I²Cn Own Slave Address

Register	SFR Address	Reset Value
I²C0ADDR0	C1H, Page 0	0000_0000 b
I²C0ADDR1	A1H, Page 2	0000_0000 b
I²C0ADDR2	A2H, Page 2	0000_0000 b
I²C0ADDR3	A3H, Page 2	0000_0000 b

I²C1ADDR0	B2H, Page 0	0000_0000 b
I²C1ADDR1	A4H, Page 2	0000_0000 b
I²C1ADDR2	A5H, Page 2	0000_0000 b
I²C1ADDR3	A6H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
I²CnADDRx[7:1]							GC
R/W							R/W

Bit	Name	Description
7:1	I²C0ADDRx[7:1]	I²C0 device's own slave address In master mode: These bits have no effect. In slave mode: These 7 bits define the slave address of this I²C device by user. The master should address I²C device by sending the same address in the first byte data after a START or a repeated START condition. If the AA flag is set, this I²C device will acknowledge the master after receiving its own address and become an addressed slave. Otherwise, the addressing from the master will be ignored. Note that I²CnADDRx[7:1] should not remain its default value of all 0, because address 0x00 is reserved for General Call.
6	GC	General Call bit In master mode: This bit has no effect. In slave mode: 0 = The General Call is always ignored. 1 = The General Call is recognized if AA flag is 1; otherwise, it is ignored if AA is 0.

I²CnCLK – I²C Clock

Register	SFR Address	Reset Value
I²C0CLK	BEH, Page 0	0000_1001 b
I²C1CLK	B5H, Page 0	0000_1001 b

7	6	5	4	3	2	1	0
I²CnCLK[7:0]							
R/W							

Bit	Name	Description

Bit	Name	Description
7:0	I^2CnCLK[7:0]	I^2C1 clock setting In master mode: This register determines the clock rate of I^2C bus when the device is in a master mode. The clock rate follows the equation, $\frac{F_{SYS}}{4 \times (I2CLK + 1)}$. Note that the I^2CnCLK value of 00H and 01H are not valid. This is an implement limitation. In slave mode: This byte has no effect. In slave mode, the I^2C device will automatically synchronize with any given clock rate up to 400k bps.

I2CnTOC – I^2C Time-out Counter

Register	SFR Address	Reset Value
I^2C0TOC	BFH, Page 0	0000_0000 b
I^2C1TOC	B6H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	-	I2TOCEN	DIV	I2TOF
-	-	-	-	-	R/W	R/W	R/W

Bit	Name	Description
7:3	-	Reserved
2	I2TOCEN	I^2C0 time-out counter enable 0 = I^2C time-out counter Disabled. 1 = I^2C time-out counter Enabled.
1	DIV	I^2C0 time-out counter clock divider 0 = The clock of I^2C time-out counter is F_{SYS}/1. 1 = The clock of I^2C time-out counter is F_{SYS}/4.

18.3 Typical Structure of I²C Interrupt Service Routine

The following software example in C language for KEIL™ C51 compiler shows the typical structure of the I²C interrupt service routine including the 26 state service routines and may be used as a base for user applications. User can follow or modify it for their own application. If one or more of the five modes are not used, the associated state service routines may be removed, but care should be taken that a deleted routine can never be invoked.

```
Void I2C_ISR (void) interrupt 6
{
      switch (I2STAT)
      {
            //============================================
            //Bus Error, always put in ISR for noise handling
            //============================================
      case 0x00:                    /*00H, bus error occurs*/
                  STO = 1;                //recover from bus error
                  break;
            //===========
            //Master Mode
            //===========
      case 0x08:                    /*08H, a START transmitted*/
                  STA = 0;                //STA bit should be cleared by software
                  I2DAT = SLA_ADDR1;            //load SLA+W/R
                  break;
      case 0x10:                    /*10H, a repeated START transmitted*/
                  STA = 0;
                  I2DAT = SLA_ADDR2;
                  break;
            //=======================
            //Master Transmitter Mode
            //=======================
      case 0x18:                    /*18H, SLA+W transmitted, ACK received*/
                  I2DAT = NEXT_SEND_DATA1;    //load DATA
                  break;
      case 0x20:                    /*20H, SLA+W transmitted, NACK received*/
                  STO = 1;                //transmit STOP
                  AA = 1;                 //ready for ACK own SLA+W/R or General Call
                  break;
      case 0x28:                    /*28H, DATA transmitted, ACK received*/
                  if (Conti_TX_Data)            //if continuing to send DATA
                  I2DAT = NEXT_SEND_DATA2;
                  else                    //if no DATA to be sent
                  {
                        STO = 1;
                        AA = 1;
                  }
                  break;
      case 0x30:                    /*30H, DATA transmitted, NACK received*/
                  STO = 1;
                  AA = 1;
                  break;
            //===========
            //Master Mode
            //===========
      case 0x38:                    /*38H, arbitration lost*/
                  STA = 1;                //retry to transmit START if bus free
                  break;
            //====================
            //Master Receiver Mode
            //====================
      case 0x40:                    /*40H, SLA+R transmitted, ACK received*/
```

```
                AA = 1;                 //ACK next received DATA
                break;
        case 0x48:                      /*48H, SLA+R transmitted, NACK received*/
                STO = 1;
                AA = 1;
                break;
        case 0x50:                      /*50H, DATA received, ACK transmitted*/
                DATA_RECEIVED1 = I2DAT;     //store received DATA
                if (To_RX_Last_Data1)   //if last DATA will be received
                      AA = 0;           //not ACK next received DATA
                else                    //if continuing receiving DATA
                      AA = 1;
                break;
        case 0x58:                      /*58H, DATA received, NACK transmitted*/
                DATA_RECEIVED_LAST1 = I2DAT;
                STO = 1;
                AA = 1;
                break;
             //====================================
             //Slave Receiver and General Call Mode
             //====================================
        case 0x60:                      /*60H, own SLA+W received, ACK returned*/
                AA = 1;
                break;
        case 0x68:                      /*68H, arbitration lost in SLA+W/R
                                               own SLA+W received, ACK returned */
                AA = 0;                 //not ACK next received DATA after
                                        //arbitration lost
                STA = 1;                //retry to transmit START if bus free
                break;
         case 0x70:                             //*70H, General Call received, ACK
                                                returned
                AA = 1;
                break;
        case 0x78:                      /*78H, arbitration lost in SLA+W/R
                                               General Call received, ACK returned*/
                AA = 0;
                STA = 1;
                break;
        case 0x80:                      /*80H, previous own SLA+W, DATA received,
                                               ACK returned*/
                DATA_RECEIVED2 = I2DAT;
                if (To_RX_Last_Data2)
                      AA = 0;
                else
                      AA = 1;
                break;
        case 0x88:                      /*88H, previous own SLA+W, DATA received,
                                               NACK returned, not addressed SLAVE mode
                                               entered*/
                DATA_RECEIVED_LAST2 = I2DAT;
                AA = 1;                 //wait for ACK next Master addressing
                break;
        case 0x90:                      /*90H, previous General Call, DATA received,
                                               ACK returned*/
                DATA_RECEIVED3 = I2DAT;
                if (To_RX_Last_Data3)
                      AA = 0;
                else
                      AA = 1;
                break;
        case 0x98:                      /*98H, previous General Call, DATA received,
                                               NACK returned, not addressed SLAVE mode
                                               entered*/
```

```
                DATA_RECEIVED_LAST3 = I2DAT;
                AA = 1;
                break;
          //==========
          //Slave Mode
          //==========
    case 0Xa0:                  /*A0H, STOP or repeated START received while
                                  still addressed SLAVE mode*/
                AA = 1;
                break;
          //======================
          //Slave Transmitter Mode
          //======================
    case 0Xa8:                  /*A8H, own SLA+R received, ACK returned*/
                I2DAT = NEXT_SEND_DATA3;
                AA = 1;               //when AA is "1", not last data to be
                                      //transmitted
                break;
    case 0Xb0:                  /*B0H, arbitration lost in SLA+W/R
                                  own SLA+R received, ACK returned */
                I2DAT = DUMMY_DATA;
                AA = 0;               //when AA is "0", last data to be
                                      //transmitted
                STA = 1;              //retry to transmit START if bus free
                break;
    case 0Xb8:                  /*B8H, previous own SLA+R, DATA transmitted,
                                  ACK received*/
                I2DAT = NEXT_SEND_DATA4;
                if (To_TX_Last_Data)  //if last DATA will be transmitted
                      AA = 0;
                else
                      AA = 1;
                break;
    case 0Xc0:                  /*C0H, previous own SLA+R, DATA transmitted,
                                  NACK received, not addressed SLAVE mode
                                  entered*/
                AA = 1;
                break;
    case 0Xc8:                  /*C8H, previous own SLA+R, last DATA trans-
                                  mitted, ACK received, not addressed SLAVE
                AA = 1;               mode entered*/
                break;
    }//end of switch (I2STAT)

    SI = 0;                           //SI should be the last command of I2C ISR
    while(STO);                       //wait for STOP transmitted or bus error
                                      //free, STO is cleared by hardware
}//end of I2C_ISR
```

19 PIN INTERRUPT

The ML51 provides pin interrupt input for each I/O pin to detect pin state if button or keypad set is used. A maximum 8-channel pin interrupt detection can be assigned by I/O port sharing. The pin interrupt is generated when any key is pressed on a keyboard or keypad, which produces an edge or level triggering event. Pin interrupt may be used to wake the CPU up from Idle or Power-down mode.

Each channel of pin interrupt can be enabled and polarity controlled independently by PIPEN and PINEN register. PICON selects which port that the pin interrupt is active. It also defines which type of pin interrupt is used – level detect or edge detect. Each channel also has its own interrupt flag. There are total eight pin interrupt flags located in PIF register. The respective flags for each pin interrupt channel allow the interrupt service routine to poll on which channel on which the interrupt event occurs. All flags in PIF register are set by hardware and should be cleared by software.

Pin interrupt is generally used to detect an edge transient from peripheral devices like keyboard or keypad. During idle state, the system prefers to enter Power-down mode to minimize power consumption and waits for event trigger. Pin interrupt can wake up the device from Power-down mode.

19.1 Pin Interrupt Block Diagram

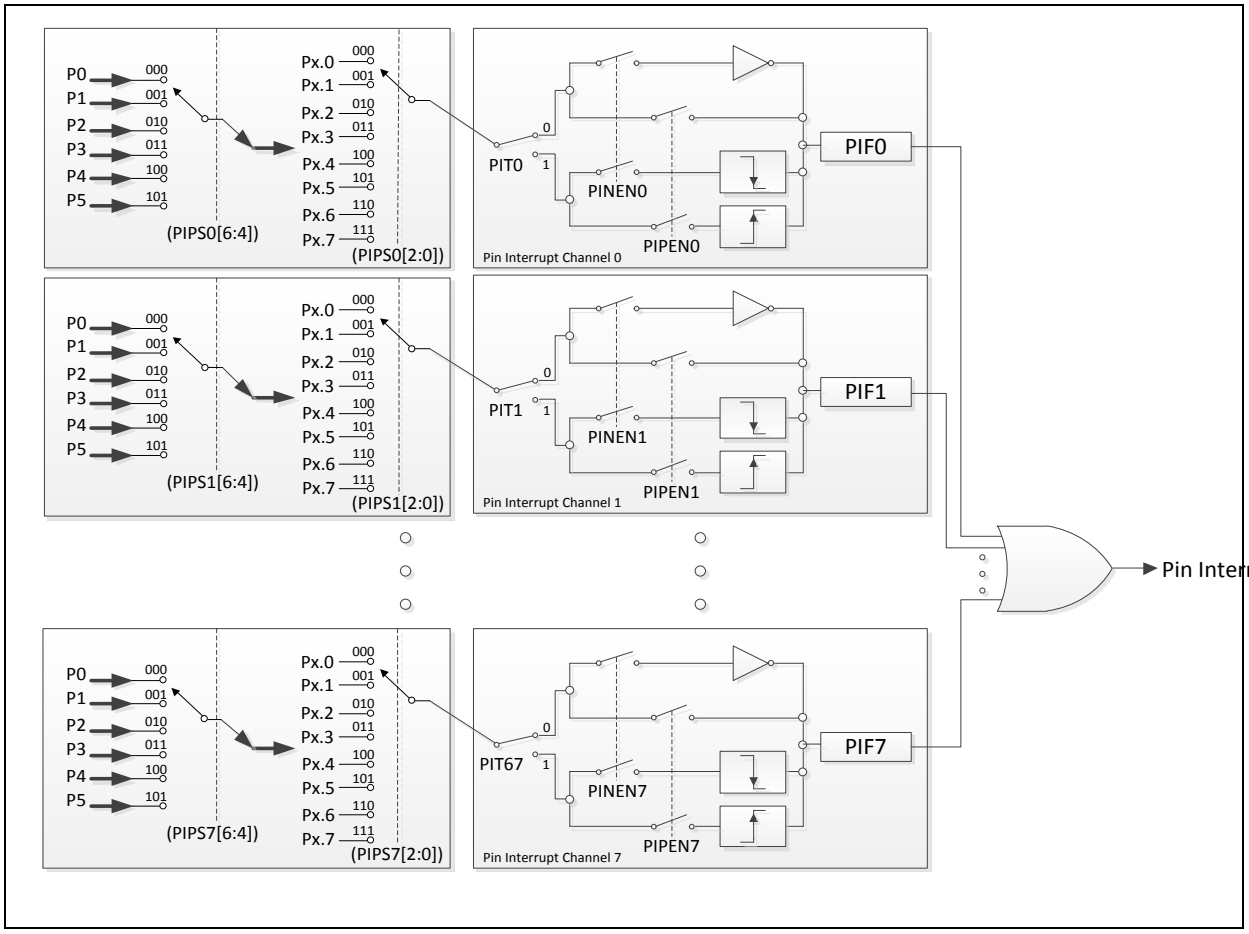


Figure 19.1-1 Pin Interface Block Diagram

19.2 Control Register of Pin Interrupt

PICON – Pin Interrupt Control

Register	SFR Address	Reset Value
PICON	E9H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
PIT7	PIT6	PIT5	PIT4	PIT3	PIT2	PIT1	PIT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PIT7	Pin interrupt channel 7 type select This bit selects which type that pin interrupt channel 7 is triggered. 0 = Level triggered. 1 = Edge triggered.
6	PIT6	Pin interrupt channel 6 type select This bit selects which type that pin interrupt channel 6 is triggered. 0 = Level triggered. 1 = Edge triggered.
5	PIT5	Pin interrupt channel 5 type select This bit selects which type that pin interrupt channel 5 is triggered. 0 = Level triggered. 1 = Edge triggered.
4	PIT4	Pin interrupt channel 4 type select This bit selects which type that pin interrupt channel 4 is triggered. 0 = Level triggered. 1 = Edge triggered.
3	PIT3	Pin interrupt channel 3 type select This bit selects which type that pin interrupt channel 3 is triggered. 0 = Level triggered. 1 = Edge triggered.
2	PIT2	Pin interrupt channel 2 type select This bit selects which type that pin interrupt channel 2 is triggered. 0 = Level triggered. 1 = Edge triggered.
1	PIT1	Pin interrupt channel 1 type select This bit selects which type that pin interrupt channel 1 is triggered. 0 = Level triggered. 1 = Edge triggered.

Bit	Name	Description
0	PIT0	Pin interrupt channel 0 type select This bit selects which type that pin interrupt channel 0 is triggered. 0 = Level triggered. 1 = Edge triggered.

PINEN – Pin Interrupt Negative Polarity Enable.

7	6	5	4	3	2	1	0
PINEN7	PINEN6	PINEN5	PINEN4	PINEN3	PINEN2	PINEN1	PINEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: EAH, Page 1 Reset value: 0000 0000b

Bit	Name	Description
7:0	PINENn	Pin interrupt channel n negative polarity enable This bit enables low-level/falling edge triggering pin interrupt channel n. The level or edge triggered selection depends on each control bit PITn in PICON. 0 = Low-level/falling edge detect Disabled. 1 = Low-level/falling edge detect Enabled.

PIPEN – Pin Interrupt Positive Polarity Enable.

7	6	5	4	3	2	1	0
PIPEN7	PIPEN6	PIPEN5	PIPEN4	PIPEN3	PIPEN2	PIPEN1	PIPEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Address: EBH, Page 1 Reset value: 0000 0000b

Bit	Name	Description
7:0	PIPENn	Pin interrupt channel n positive polarity enable This bit enables high-level/rising edge triggering pin interrupt channel n. The level or edge triggered selection depends on each control bit PITn in PICON. 0 = High-level/rising edge detect Disabled. 1 = High-level/rising edge detect Enabled.

PIF – Pin Interrupt Flags

7	6	5	4	3	2	1	0
PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0
R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)	R (level) R/W (edge)

Address: CAH, Page 0 Reset value: 0000 0000b

Bit	Name	Description

Bit	Name	Description
7:0	PIFn	Pin interrupt channel n flag If the edge trigger is selected, this flag will be set by hardware if the channel n of pin interrupt detects an enabled edge trigger. This flag should be cleared by software. If the level trigger is selected, this flag follows the inverse of the input signal's logic level on the channel n of pin interrupt. Software cannot control it.

PIPSn – Pin Interrupt Control

Register	SFR Address	Reset Value
PIPS0	A1H, Page 1	0000_0000 b
PIPS1	A2H, Page 1	0000_0000 b
PIPS2	A3H, Page 1	0000_0000 b
PIPS3	A4H, Page 1	0000_0000 b
PIPS4	A5H, Page 1	0000_0000 b
PIPS5	A6H, Page 1	0000_0000 b
PIPS6	A7H, Page 1	0000_0000 b
PIPS7	AFH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
-	PSEL[2:0]			-	BSEL[2:0]		
-	R/W			-	R/W		

Bit	Name	Description
7	-	Reserved
6:4	PSEL[2:0]	Pin interrupt channel Port select 000 = P0 PORT. 001 = P1 PORT. 010 = P2 PORT. 011 = P3 PORT. 100 = P4 PORT. 101 = P5 PORT. 110 = Reserved. 111 = Reserved.
3	-	Reserved

Bit	Name	Description
2:0	BSEL[2:0]	Pin interrupt channel bit select 000 = Pn.0. 001 = Pn.1 010 =.Pn.2 011 = Pn.3. 100 = Pn.4. 101 = Pn.5. 110 = Pn.6. 111 = Pn.7. n is the PORT number, which is selected by PSEL[2:0].

20 PULSE WIDTH MODULATED (PWM)

The PWM (Pulse Width Modulation) signal is a useful control solution in wide application field. It can used on motor driving, fan control, backlight brightness tuning, LED light dimming, or simulating as a simple digital to analog converter output through a low pass filter circuit.

The ML51 PWM is especially designed for motor control by providing three pairs, maximum 16-bit resolution of PWM output with programmable period and duty. The architecture makes user easy to drive the one-phase or three-phase brushless DC motor (BLDC), or three-phase AC induction motor. Each of six PWM can be configured as one of independent mode, complementary mode, or synchronous mode. If the complementary mode is used, a programmable dead-time insertion is available to protect MOS turn-on simultaneously. The PWM waveform can be edge-aligned or center-aligned with variable interrupt points.

Following is the PWM multi function pin define list.

Group	Pin Name	GPIO	MFP	Type	Description
PWM0	PWM0_BRAKE	P2.3	MFP13	I	PWM0 Brake input pin.
		P2.1	MFP13	I	
		P2.0	MFP13	I	
		P5.7	MFP10	I	
		P5.6	MFP11	I	
		P3.3	MFP15	I	
	PWM0_CH0	P2.5	MFP11	I/O	PWM0 channel 0 output/capture input.
		P5.5	MFP7	I/O	
		P0.5	MFP13	I/O	
		P4.6	MFP12	I/O	
	PWM0_CH1	P2.4	MFP11	I/O	PWM0 channel 1 output/capture input.
		P5.4	MFP7	I/O	
		P0.4	MFP13	I/O	
		P5.6	MFP12	I/O	
	PWM0_CH2	P2.3	MFP11	I/O	PWM0 channel 2 output/capture input.
		P0.3	MFP13	I/O	
	PWM0_CH3	P2.2	MFP11	I/O	PWM0 channel 3 output/capture input.
		P0.2	MFP13	I/O	
	PWM0_CH4	P2.1	MFP11	I/O	PWM0 channel 4 output/capture input.
		P5.7	MFP12	I/O	
		P0.1	MFP13	I/O	
	PWM0_CH5	P2.0	MFP11	I/O	PWM0 channel 5 output/capture input.
		P3.6	MFP12	I/O	
		P0.0	MFP13	I/O	

20.1 Functional Description

20.1.1 PWM block diagram

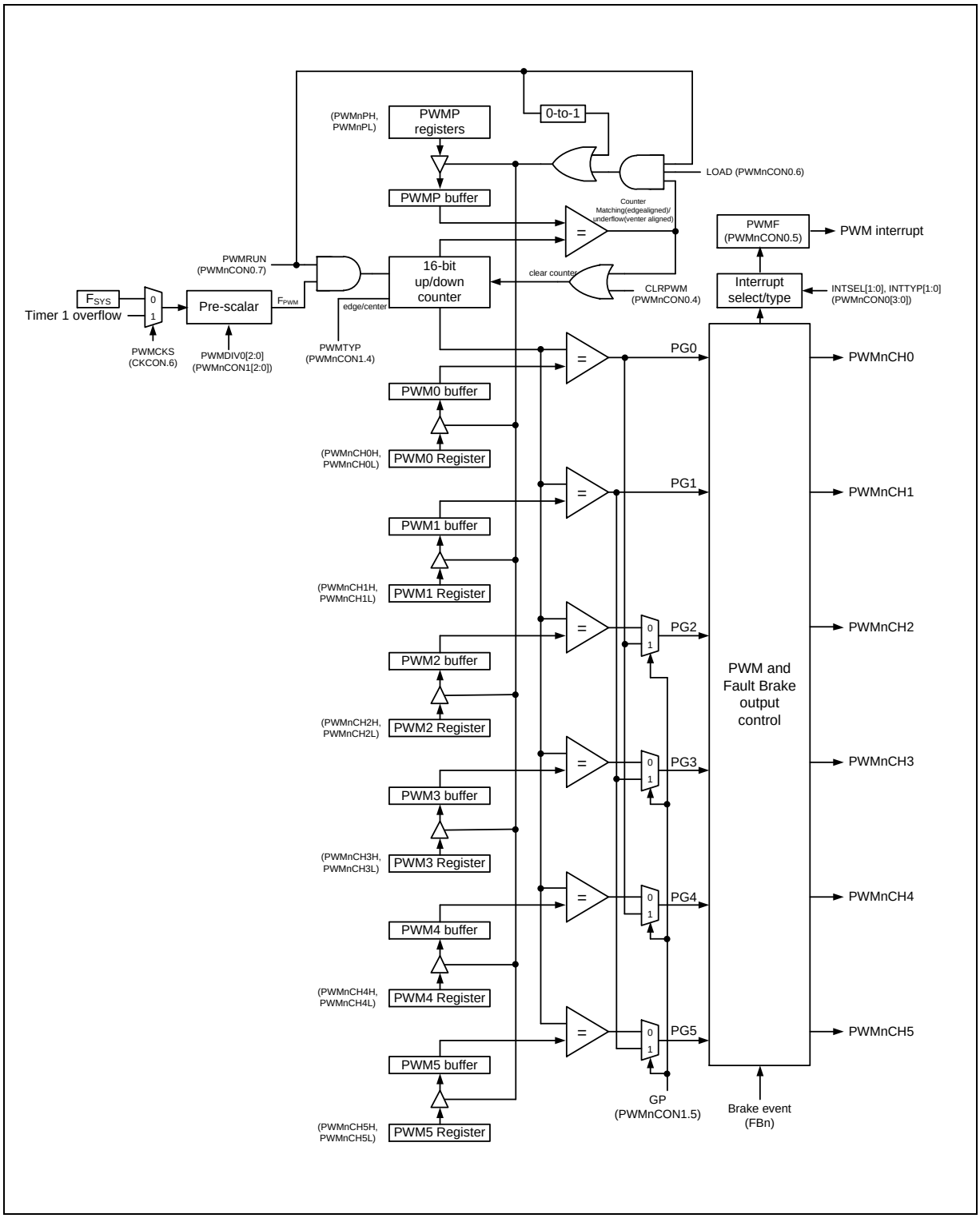


Figure 20.1-1 PWM Block Diagram

20.1.2 PWM Generator

The PWM generator is clocked by the system clock or Timer 1 overflow divided by a PWM clock pre-scalar selectable from 1/1~1/128. The PWM period is defined by effective 16-bit period registers, {PWMnPH, PWMnPL}. The period is the same for all PWM channels for they share the same 16-bit period counter. The duty of each PWM is determined independently by the value of duty registers {PWMnC0H, PWMnC0L}, {PWMnC1H, PWMnC1L}, {PWMnC2H, PWMnC2L}, {PWMnC3H, PWMnC3L}, {PWMnC4H, PWMnC4L}, and {PWMnC5H, PWMnC5L}. With six duty registers, six PWM output can be generated independently with different duty cycles. The interval and duty of PWM signal is generated by a 16-bit counter comparing with the period and duty registers.

To facilitate the three-phase motor control, a group mode can be used by setting GP (PWMnCON1.5), which makes {PWMnC0H, PWMnC0L} and {PWMnC1H, PWMnC1L} duty register decide duties of the PWM outputs. In a three-phase motor control application, two-group PWM outputs generally are given the same duty cycle. When the group mode is enabled, {PWMnC2H, PWMnC2L}, {PWMnC3H, PWMnC3L}, {PWMnC4H, PWMnC4L} and {PWMnC5H, PWMnC5L} registers have no effect. This mean is {PWMnC2H, PWMnC2L} and {PWMnC4H, PWMnC4L} both as same as {PWMnC0H, PWMnC0L}. Also {PWMnC3H, PWMnC3L} and {PWMnC5H, PWMnC5L} are same as {PWMnC1H, PWMnC1L}.Note that enabling PWM does not configure the I/O pins into their output mode automatically. User should configure I/O output mode via software manually.

The PWM counter generates six PWM signals called PG0, PG1, PG2, PG3, PG4, and PG5. These signals will go through the PWM and Fault Brake output control circuit. It generates real PWM outputs on I/O pins. The output control circuit determines the PWM mode, dead-time insertion, mask output, Fault Brake control, and PWM polarity. The last stage is a multiplexer of PWM output or I/O function.

User should follow the initialization steps below to start generating the PWM signal output. In the first step by setting CLRPWM (PWMnCON0.4), it ensures the 16-bit up counter reset for the accuracy of the first duration. After initialization and setting {PWMnPH, PWMnPL} and all {PWMnH, PWMnL} registers, PWMRUN (PWMnCON0.7) can be set as logic 1 to trigger the 16-bit counter running. PWM starts to generate waveform on its output pins. The hardware for all period and duty control registers are double buffered designed. Therefore, {PWMnPH, PWMnPL} and all {PWMnH, PWMnL} registers can be written to at any time, but the period and duty cycle of PWM will not be updated immediately until the LOAD (PWMnCON0.6) is set and previous period is complete. This prevents glitches when updating the PWM period or duty.

NOTE: A loading of new period and duty by setting LOAD should be ensured complete by monitoring it and waiting for a hardware automatic clearing LOAD bit. Any updating of PWM control registers during LOAD bit as logic 1 will cause unpredictable output

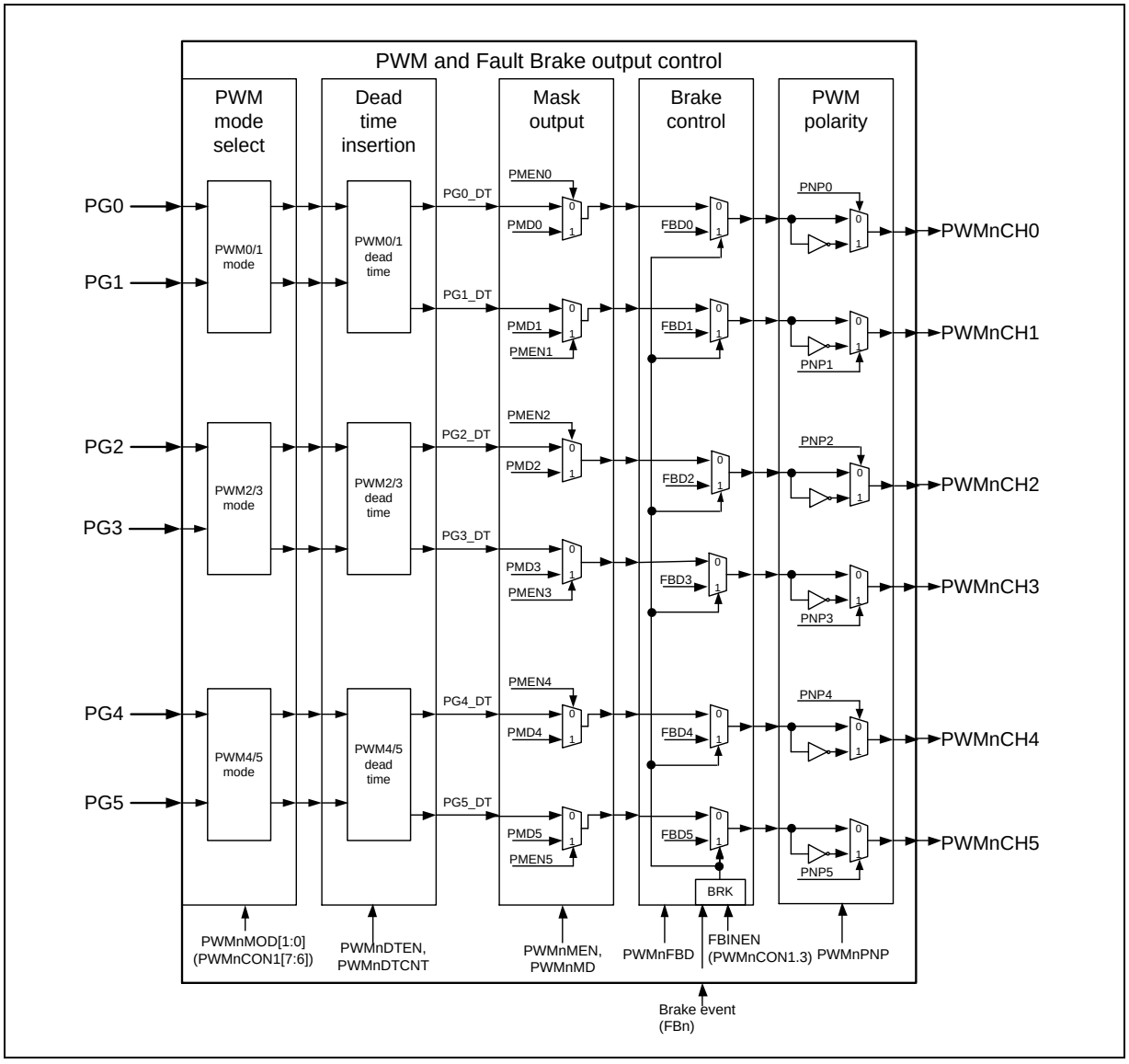


Figure 20.1-2 PWM and Fault Brake Output Control Block Diagram

20.1.3 PWM Types

The PWM generator provides two PWM types: edge-aligned or center-aligned. PWM type is selected by PWMTYP (PWMnCON1.4).

20.1.3.1 Edge-Aligned Type

In edge-aligned mode, the 16-bit counter uses single slop operation by counting up from 0000H to {PWMnPH, PWMnPL} and then starting from 0000H. The PWM generator signal (PGn before PWM and Fault Brake output control) is cleared on the compare match of 16-bit counter and the duty register {PWMnH, PWMnL} and set at the 16-bit counter is 0000H. The result PWM output waveform is left-edge aligned.

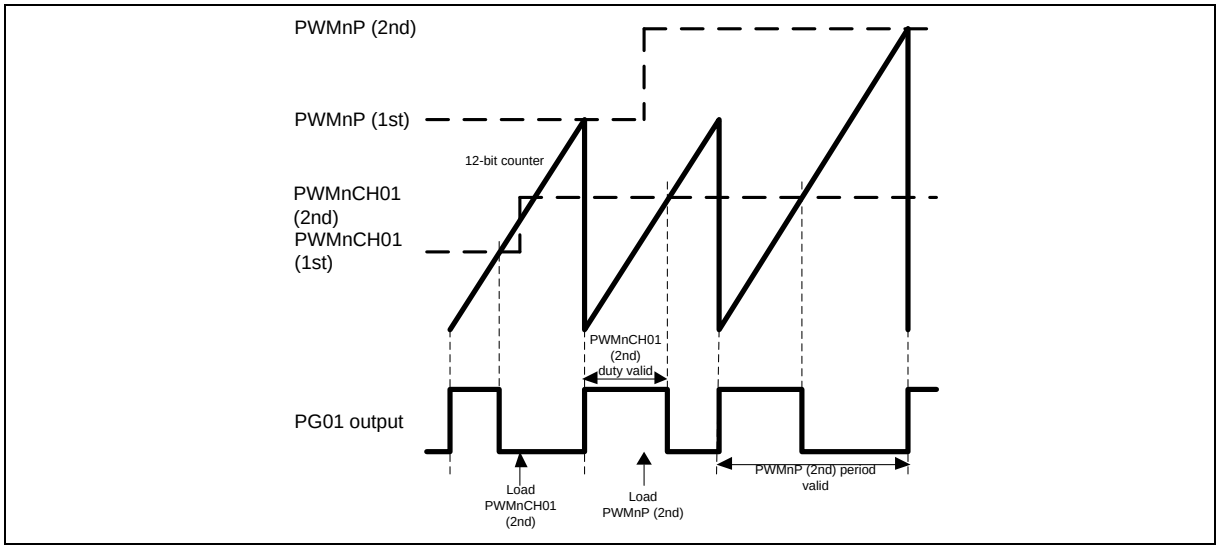


Figure 20.1-3 PWM Edge-aligned Type Waveform

The output frequency and duty cycle for edge-aligned PWM are given by following equations:

PWM frequency = $\frac{F_{PWM}}{\{PWMnPH, PWMnPL\}+1}$ (F_{PWM} is the PWM clock source frequency divided by PWMDIV).

PWM high level duty = $\frac{\{PWMnCHxH, PWMnCHxL\}}{\{PWMnPH, PWMnPL\}+1}$.

20.1.3.2 Center-Aligned Type

In center-aligned mode, the 16-bit counter use dual slop operation by counting up from 0000H to {PWMnPH, PWMnPL} and then counting down from {PWMnPH, PWMnPL} to 0000H. The PGn signal is cleared on the up-count compare match of 16-bit counter and the duty register {PWMnH, PWMnL} and set on the down-count compare match. Center-aligned PWM may be used to generate non-overlapping waveforms.

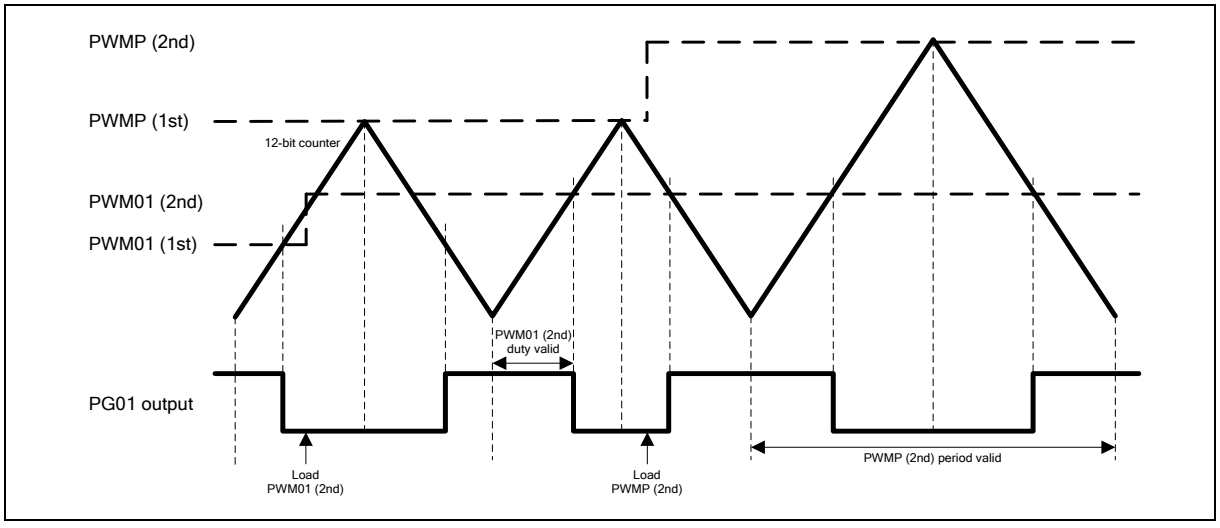


Figure 20.1-4 PWM Center-aligned Type Waveform

The output frequency and duty cycle for center-aligned PWM are given by following equations:

PWM frequency = $\frac{F_{PWM}}{2 \times \{PWMnPH, PWMnPL\}}$ (F_{PWM} is the PWM clock source frequency divided by PWMDIV).

PWM high level duty = $\frac{\{PWMnCHxH, PWMnCHxL\}}{\{PWMnPH, PWMnPL\}}$.

20.1.4 Operation Modes

After PGn signals pass through the first stage of the PWM and Fault Brake output control circuit. The PWM mode selection circuit generates different kind of PWM output modes with six-channel, three-pair signal PG0~PG5 . It supports independent mode, complementary mode, and synchronous mode.

20.1.4.1 Independent Mode

Independent mode is enabled when PWMMOD[1:0] (PWMnCON1[7:6]) is [0:0]. It is the default mode of PWM. PG0, PG1, PG2, PG3, PG4 and PG5 output PWM signals independently.

20.1.4.2 Complementary Mode with Dead-Time Insertion

Complementary mode is enabled when PWMMOD[1:0] = [0:1]. In this mode, PG0/2/4 output PWM signals the same as the independent mode. However, PG1/3/5 output the out-phase PWM signals of PG0/2/4 correspondingly, and ignore PG1/3/5 Duty register {PWMnH, PWMnL} (n:1/3/5). This mode makes PG0/PG1 a PWM complementary pair and so on PG2/PG3 and PG4/PG5.

In a real motor application, a complementary PWM output always has a need of “dead-time” insertion to prevent damage of the power switching device like GPIBs due to being active on simultaneously of the upper and lower switches of the half bridge, even in a “μs” duration. For a power switch device physically cannot switch on/off instantly. For the ML51 PWM, each PWM pair share a 9-bit dead-time down-counter PWM0DTCNT used to produce the off time between two PWM signals in the same pair. On implementation, a 0-to-1 signal edge delays after PWM0DTCNT timer underflows. The timing diagram illustrates the complementary mode with dead-time insertion of PG0/PG1 pair. Pairs of PG2/PG3 and PG4/PG5 have the same dead-time circuit. Each pair has its own dead-time enabling bit in the field of PWMnDTEN [3:0].

Note that the PWM0DTCNT and PWMnDTEN registers are all TA write protection. The dead-time control are also valid only when the PWM is configured in its complementary mode.

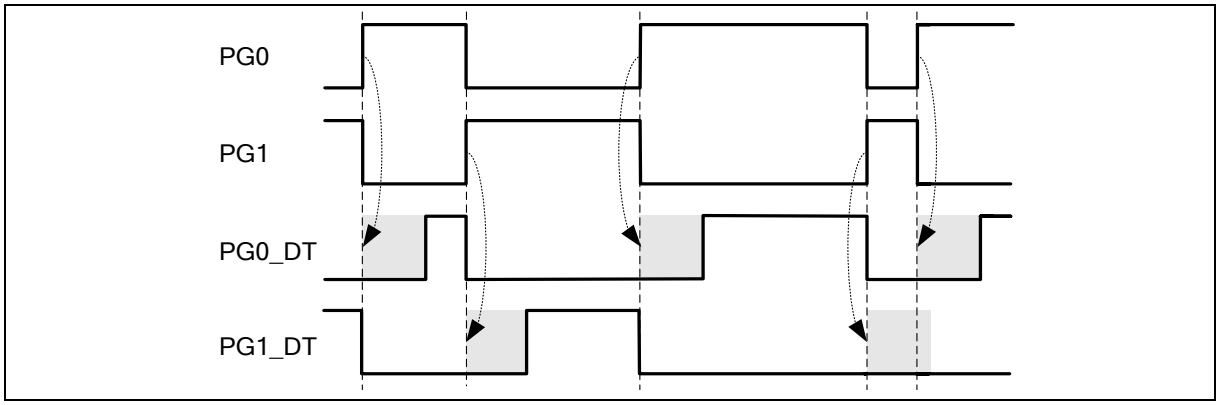


Figure 20.1-5 PWM Complementary Mode with Dead-time Insertion

20.1.4.3 Synchronous Mode

Synchronous mode is enabled when PWMMOD[1:0] = [1:0]. In this mode, PG0/2/4 output PWM signals the same as the independent mode. PG1/3/5 output just the same in-phase PWM signals of PG02/4 correspondingly.

20.1.5 Mask Output Control

Each PWM signal can be software masked by driving a specified level of PWM signal. The PWM mask output function is quite useful when controlling Electrical Commutation Motor like a BLDC. PWMnMEN contains six bits, those determine which channel of PWM signal will be masked. PWMnMD set the individual mask level of each PWM channel. The default value of PWMnMEN is 00H, which makes all outputs of PWM channels follow signals from PWM generator. Note that the masked level is reversed or not by PWM0NP setting on PWM output pins.

20.1.6 Fault Brake

The Fault Brake function is usually implemented in conjunction with an enhanced PWM circuit. It rules as a fault detection input to protect the motor system from damage. Fault Brake pin input (FB) is valid when FBINEN (PWMnCON1.3) is set. When Fault Brake is asserted PWM signals will be individually overwritten by PWMnFBD corresponding bits. PWMRUN (PWMnCON0.7) will also be automatically cleared by hardware to stop PWM generating. The PWM 16-bit counter will also be reset as 0000H. A indicating flag FBF will be set by hardware to assert a Fault Brake interrupt if enabled. PWMnFBD data output remains even after the FBF is cleared by software. User should resume the PWM output only by setting PWMRUN again. Meanwhile the Fault Brake state will be released and PWM waveform outputs on pins as usual. Fault Brake input has a polarity selection by FBINLS (PWMnFBD.6) bit. Note that the Fault Brake signal feed in FB pin should be longer than eight-system-clock time for FB pin input has a permanent $8/F_{SYS}$ de-bouncing, which avoids fake Fault Brake event by input noise. The other path to trigger a Fault Brake event is the ADC compare event. It asserts the Fault Brake behavior just the same as FB pin input. See Section 21.2.3 ADC Conversion Result Comparator21.2.3

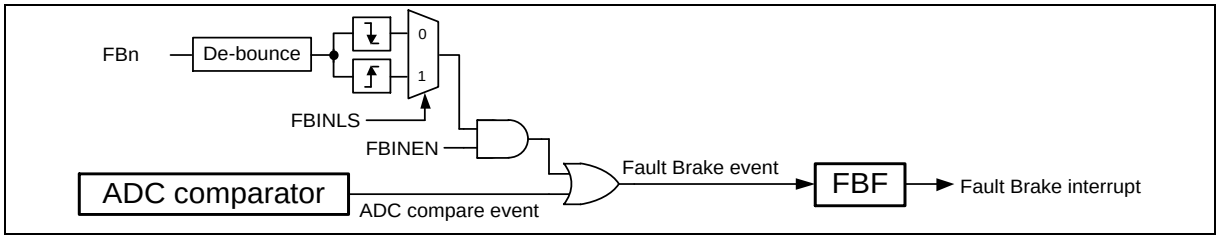


Figure 20.1-6 Fault Brake Function Block Diagram

20.1.7 Polarity Control

Each PWM output channel has its independent polarity control bit, PNP0~PNP5. The default is high active level on all control fields implemented with positive logic. It means the power switch is ON when PWM outputs high level and OFF when low level. User can easily configure all setting with positive logic and then set PWMnNP bit to make PWM actually outputs according to the negative logic.

20.1.8 PWM Interrupt

The PWM module has a flag PWMF (PWMnCON0.5) to indicate certain point of each complete PWM period. The indicating PWM channel and point can be selected by INTSEL[2:0] and INTTYP[1:0] (PWMnINTC[2:0] and [5:4]). Note that the center point and the end point interrupts are only available when PWM operates in its center-aligned type. PWMF is cleared by software.

The PWM interrupt related with PWM waveform is shown as figure below.

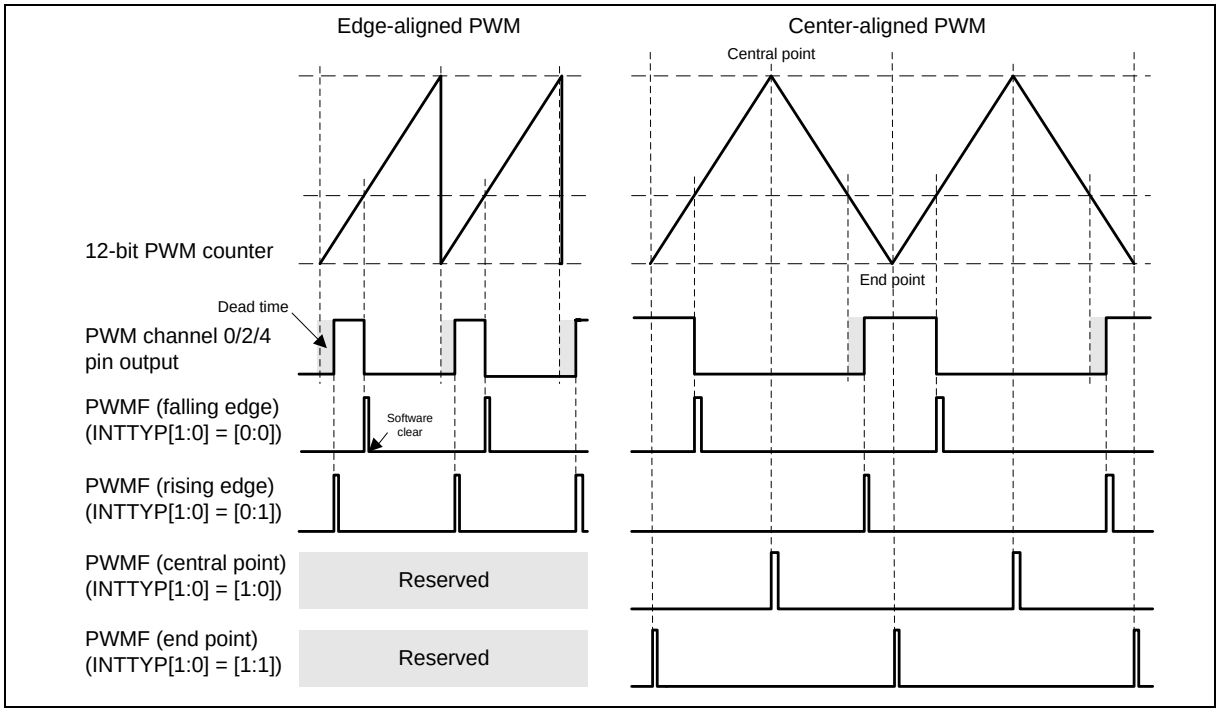


Figure 20.1-7 PWM Interrupt Type

Fault Brake event requests another interrupt, Fault Brake interrupt. It has different interrupt vector from PWM interrupt. When either Fault Brake pin input event or ADC compare event occurs, FBF (PWMnFBD.7) will be set by hardware. It generates Fault Brake interrupt if enabled. The Fault Brake interrupt enable bit is EFB0 (EIE0.5). FBF Is cleared via software.

20.2 Control Register Of PWM

Note: this chapter register table include PWM0 & PWM1 introduction. But following list products only support PWM0, include ML51BB9AE, ML51DB9AE, ML51FB9AE, ML51OB9AE, ML51XB9AE, ML51EB9AE, ML51UB9AE, ML51PB9AE, ML51TB9AE, ML51EC0AE, ML51UC0AE, ML51PC0AE, ML51TC0AE.

CKCON – Clock Control

Register	SFR Address	Reset Value
TH1	8EH, Page 0	1000_0000b

7	6	5	4	3	2	1	0
FASTWK	PWMCKS	T1OE	T1M	T0M	T0OE	CLOEN	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Bit	Name	Description
6	PWMCKS	PWM clock source select 0 = The clock source of PWM is the system clock FSYS. 1 = The clock source of PWM is the overflow of Timer 1.

PWMnCON0 – PWM Control register 0

Register	SFR Address	Reset Value
PWM0CON0	D1H, Page 0	0000_0000 b
PWM1CON0	C5H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PWMRUN	LOAD	PWMF	CLRPWM	-	-	-	-
R/W	R/W	R/W	R/W	-	-	-	-

Bit	Name	Description
7	PWMRUN	PWM run enable 0 = PWM stays in idle. 1 = PWM starts running.
6	LOAD	PWM new period and duty load This bit is used to load period and duty control registers in their buffer if new period or duty value needs to be updated. The loading will act while a PWM period is completed. The new period and duty affected on the next PWM cycle. After the loading is complete, LOAD will be automatically cleared via hardware. The meaning of writing and reading LOAD bit is different. Writing: 0 = No effect. 1 = Load new period and duty in their buffers while a PWM period is completed. Reading: 0 = A loading of new period and duty is finished. 1 = A loading of new period and duty is not yet finished.
5	PWMF	PWM flag This flag is set according to definitions of INTSEL[2:0] and INTTYP[1:0] in PWMnINTC. This bit is cleared by software.
4	CLRPWM	Clear PWM counter Setting this bit clears the value of PWM 16-bit counter for resetting to 0000H. After the counter value is cleared, CLRPWM will be automatically cleared via hardware. The meaning of writing and reading CLRPWM bit is different. Writing: 0 = No effect. 1 = Clearing PWM 16-bit counter. Reading: 0 = PWM 16-bit counter is completely cleared. 1 = PWM 16-bit counter is not yet cleared.

PWMnCON1 – PWM Control 1

Register	SFR Address	Reset Value
PWM0CON1	DFH, Page 1	0000_0000 b
PWM1CON1	C6H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PWMMOD[1:0]		GP	PWMTYP	FBINEN	PWMDIV[2:0]		
R/W		R/W	R/W	R/W	R/W		

Bit	Name	Description
2:0	PWMDIV[2:0]	PWM clock divider This field decides the pre-scale of PWM clock source. 000 = 1/1. 001 = 1/2 010 = 1/4. 011 = 1/8. 100 = 1/16. 101 = 1/32. 110 = 1/64. 111 = 1/128.

PWMnPL – PWM Period Low Byte

Register	SFR Address	Reset Value
PWM0PL	D9H, Page 1	0000_0000 b
PWM1PL	D1H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PWMnP[7:0]							
R/W							

Bit	Name	Description
7:0	PWMnP[7:0]	PWMn period low byte This byte with PWMnPH controls the period of the PWM generator signal.

PWMnPH – PWM Period High Byte

Register	SFR Address	Reset Value
PWM0PH	D1H, Page 1	0000_0000 b
PWM1PH	C9H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
PWMP[15:8]							
R/W							

Bit	Name	Description
7:0	PWMP[15:8]	PWM period high byte This byte with PWMnPL controls the period of the PWM generator signal.

PWMnCxH – PWM0/1 Channel 0~5 Duty High Byte n=0,1; x=0,1,2,3,4,5

Register	SFR Address	Description	Reset Value
PWM0C0H	D2H, Page 1	PWM0 Channel 0 Duty High Byte	0000_0000 b
PWM0C1H	D3H, Page 1	PWM0 Channel 1 Duty High Byte	0000_0000 b
PWM0C2H	D4H, Page 1	PWM0 Channel 2 Duty High Byte	0000_0000 b
PWM0C3H	D5H, Page 1	PWM0 Channel 3 Duty High Byte	0000_0000 b
PWM0C4H	C4H, Page 1	PWM0 Channel 4 Duty High Byte	0000_0000 b
PWM0C5H	C5H, Page 1	PWM0 Channel 5 Duty High Byte	0000_0000 b
PWM1C0H	CAH, Page 2	PWM1 Channel 0 Duty High Byte	0000_0000 b
PWM1C1H	CBH, Page 2	PWM1 Channel 1 Duty High Byte	0000_0000 b
PWM1C2H	CCH, Page 2	PWM1 Channel 2 Duty High Byte	0000_0000 b
PWM1C3H	CDH, Page 2	PWM1 Channel 3 Duty High Byte	0000_0000 b
PWM1C4H	CEH, Page 2	PWM1 Channel 4 Duty High Byte	0000_0000 b
PWM1C5H	CFH, Page 2	PWM1 Channel 5 Duty High Byte	0000_0000 b

7	6	5	4	3	2	1	0
PWMnCx [15:8], n=0,1; x=0,1,2,3,4,5							
R/W							

Bit	Name	Description
7:0	PWMnCx[15:8] n=0,1 x=0,1,2,3,4,5	PWMnCx duty high byte This byte with PWMnCxL controls the duty of the output signal PGx from PWM generator.

PWMnCxL – PWM0/1 Channel 0~5 Duty Low Byte n=0,1; x=0,1,2,3,4,5

Register	SFR Address	Description	Reset Value
PWM0C0L	DAH, Page 1	PWM0 Channel 0 Duty Low Byte	0000_0000 b
PWM0C1L	DBH, Page 1	PWM0 Channel 1 Duty Low Byte	0000_0000 b
PWM0C2L	DCH, Page 1	PWM0 Channel 2 Duty Low Byte	0000_0000 b
PWM0C3L	DDH, Page 1	PWM0 Channel 3 Duty Low Byte	0000_0000 b

PWM0C4L	CCH, Page 1	PWM0 Channel 4 Duty Low Byte	0000_0000 b
PWM0C5L	CDH, Page 1	PWM0 Channel 5 Duty Low Byte	0000_0000 b
PWM1C0L	D2H, Page 2	PWM1 Channel 0 Duty Low Byte	0000_0000 b
PWM1C1L	D3H, Page 2	PWM1 Channel 1 Duty Low Byte	0000_0000 b
PWM1C2L	D4H, Page 2	PWM1 Channel 2 Duty Low Byte	0000_0000 b
PWM1C3L	D5H, Page 2	PWM1 Channel 3 Duty Low Byte	0000_0000 b
PWM1C4L	D6H, Page 2	PWM1 Channel 4 Duty Low Byte	0000_0000 b
PWM1C5L	D7H, Page 2	PWM1 Channel 5 Duty Low Byte	0000_0000 b

7	6	5	4	3	2	1	0
PWMnCx [7:0], n=0,1; x=0,1,2,3,4,5							
R/W							

Bit	Name	Description
7:0	PWMnCx[7:0] n=0,1 x=0,1,2,3,4,5	PWMnCx duty Low byte This byte with PWMnCxH controls the duty of the output signal PGx from PWM generator.

PWMnDTEN – PWM Dead-time Enable

Register	SFR Address	Reset Value
PWM0DTEN	F9H, Page 1, TA protected	0000_0000 b
PWM1DTEN	C1H, Page 2, TA protected	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	PWMnDTCNT.8	-	PDT45EN	PDT23EN	PDT01EN
-	-	-	R/W	-	R/W	R/W	R/W

Bit	Name	Description
7:5	0	Reserved
4	PWMnDTCNT.8	PWM dead-time counter bit 8 See PWMnDTCNT register.
2	PDT45EN	PWM4/5 pair dead-time insertion enable This bit is valid only when PWM4/5 is under complementary mode. 0 = No delay on GP4/GP5 pair signals. 1 = Insert dead-time delay on the rising edge of GP4/GP5 pair signals.

Bit	Name	Description
1	PDT23EN	PWM2/3 pair dead-time insertion enable This bit is valid only when PWM2/3 is under complementary mode. 0 = No delay on GP2/GP3 pair signals. 1 = Insert dead-time delay on the rising edge of GP2/GP3 pair signals.
0	PDT01EN	PWM0/1 pair dead-time insertion enable This bit is valid only when PWM0/1 is under complementary mode. 0 = No delay on GP0/GP1 pair signals. 1 = Insert dead-time delay on the rising edge of GP0/GP1 pair signals.

PWMnDTCNT – PWM Dead-time Counter

Register	SFR Address	Reset Value
PWM0DTCNT	FAH, Page 1, TA protected	0000_0000 b
PWM1DTCNT	C2H, Page 2, TA protected	0000_0000 b

7	6	5	4	3	2	1	0
PWM0DTCNT[7:0]							
R/W							

Bit	Name	Description
7:0	PWM0DTCNT [7:0]	PWM dead-time counter low byte This 8-bit field combined with PWMnDTEN .4 forms a 9-bit PWM dead-time counter PWM0DTCNT. This counter is valid only when PWM is under complementary mode and the correspond PWMnDTEN bit for PWM pair is set. PWM dead-time = $\frac{PDTCNT+1}{F_{SYS}}$. Note that user should not modify PWM0DTCNT during PWM run time.

PWMnMEN – PWM Mask Enable

Register	SFR Address	Reset Value
PWM0MEN	FBH, Page 1	0000_0000 b
PWM1MEN	C3H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	PMEN5	PMEN4	PMEN3	PMEN2	PMEN1	PMEN0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description

Bit	Name	Description
5:0	PMENn	PWMn mask enable 0 = PWMn signal outputs from its PWM generator. 1 = PWMn signal is masked by PMDn.

PWMnMD – PWM Mask Data

Register	SFR Address	Reset Value
PWM0MD	FCH, Page 1	0000_0000 b
PWM1MD	C4H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:6	0	Reserved The bits are always read as 0.
5:0	PMDn	PWMn mask data The PWMn signal outputs mask data once its corresponding PMENn is set. 0 = PWMn signal is masked by 0. 1 = PWMn signal is masked by 1.

PWMnFBD – PWM Fault Brake Data

Register	SFR Address	Reset Value
PWM0FBD	D7H, Page 1	0000_0000 b
PWM1FBD	BBH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
FBF	FBINLS	FBD5	FBD4	FBD3	FBD2	FBD1	FBD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	FBF	Fault Brake flag This flag is set when FBINEN is set as 1 and FB pin detects an edge, which matches FBINLS (PWMnFBD.6) selection. This bit is cleared by software. After FBF is cleared, Fault Brake data output will not be released until PWMRUN (PWMnCON0.7) is set.

Bit	Name	Description
6	FBINLS	FB pin input level selection 0 = Falling edge. 1 = Rising edge.
5:0	FBDx	PWMnCHx Fault Brake data 0 = PWMn channel x signal is overwritten by 0 once Fault Brake asserted. 1 = PWMn channel x signal is overwritten by 1 once Fault Brake asserted. **Note:** n = 0,1; x = 0,1,2,3,4,5

PWMnNP – PWM Negative Polarity

Register	SFR Address	Reset Value
PWM0NP	D6H, Page 1	0000_0000 b
PWM1NP	BAH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	PNP5	PNP4	PNP3	PNP2	PNP1	PNP0
-	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
5:0	PNPn	PWMn negative polarity output enable 0 = PWMn signal outputs directly on PWMn pin. 1 = PWMn signal outputs inversely on PWMn pin.

PWMnINTC – PWM Interrupt Control

Register	SFR Address	Reset Value
PWM0INTC	B7H, Page 1	0000_0000 b
PWM1INTC	BCH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	INTTYP1	INTTYP0	-	INTSEL2	INTSEL1	INTSEL0
-	-	R/W	R/W	-	R/W	R/W	R/W

Bit	Name	Description

Bit	Name	Description
5:4	INTTYP[1:0]	PWM interrupt type select These bit select PWM interrupt type. 00 = Falling edge on PWMn channel 0/1/2/3/4/5 pin. 01 = Rising edge on PWMn channel 0/1/2/3/4/5 pin. 10 = Central point of a PWM period. 11 = End point of a PWM period. Note that the central point interrupt or the end point interrupt is only available while PWM operates in center-aligned type.
2:0	INTSEL[2:0]	PWM interrupt pair select These bits select which PWM channel asserts PWM interrupt when PWM interrupt type is selected as falling or rising edge on PWM0/1/2/3/4/5 pin.. 000 = PWMn channel 0. 001 = PWMn channel1. 010 = PWMn channel2. 011 = PWMn channel3. 100 = PWMn channel4. 101 = PWMn channel5. Others = PWMn channel 0.

21 12-BIT ANALOG-TO-DIGITAL CONVERTER (ADC)

The ML51 is embedded with a 12-bit SAR ADC. The ADC (analog-to-digital converter) allows conversion of an analog input signal to a 12-bit binary representation of that signal. The ML51 is selected as 8-channel inputs in single end mode. The internal band-gap voltage 0.814 V also can be the internal ADC input. The analog input, multiplexed into one sample and hold circuit, charges a sample and hold capacitor. The output of the sample and hold capacitor is the input into the converter. The converter then generates a digital result of this analog level via successive approximation and stores the result in the result registers. The ADC controller also supports DMA (direct memory access) function for ADC continuous conversion and storage result data into XRAM no need special enable PDMA module.

Following shows the multi function define of ADC.

Group	Pin Name	GPIO	MFP	Type	Description
ADC	ADC_CH0	P2.5	MFP1	A	ADC_ channel analog input.
	ADC_CH1	P2.4	MFP1	A	ADC_ channel analog input.
	ADC_CH2	P2.3	MFP1	A	ADC_ channel analog input.
	ADC_CH3	P2.2	MFP1	A	ADC_ channel analog input.
	ADC_CH4	P2.1	MFP1	A	ADC_ channel analog input.
	ADC_CH5	P2.0	MFP1	A	ADC_ channel analog input.
	ADC_CH6	P3.1	MFP1	A	ADC_ channel analog input.
	ADC_CH7	P3.2	MFP1	A	ADC_ channel analog input.

21.1 ADC Block Diagram

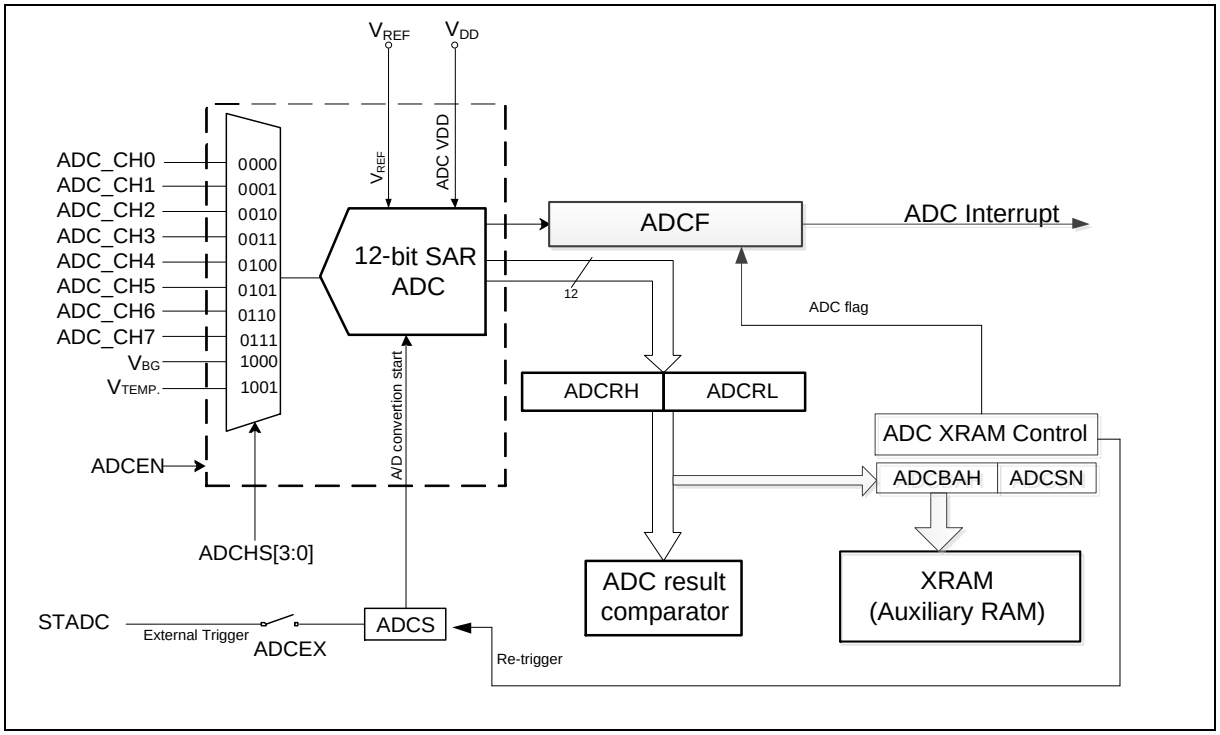


Figure 21.1-1 12-bit ADC Block Diagram

21.2 Functional Description

21.2.1 ADC Operation

Before ADC operation, the ADC circuit should be enabled by setting ADCEN (ADCCON1.0). This makes ADC circuit active. It consume extra power. Once ADC is not used, clearing ADCEN to turn off ADC circuit saves power.

The ADC analog input pin should be specially considered. ADCHS[2:0] are channel selection bits that control which channel is connected to the sample and hold circuit. User needs to configure selected ADC input pins as input-only (high impedance) mode via respective bits in PxMn registers. This configuration disconnects the digital output circuit of each selected ADC input pin. But the digital input circuit still works. Digital input may cause the input buffer to induce leakage current. To disable the digital input buffer, the respective bits in AINDIDS should be set. Configuration above makes selected ADC analog input pins pure analog inputs to allow external feeding of the analog voltage signals. Also, the ADC clock rate needs to be considered carefully. The ADC maximum clock frequency is listed in ADC Analog Electrical Characteristics. Clock above the maximum clock frequency degrades ADC performance unpredictably.

An A/D conversion is initiated by setting the ADCS bit (ADCCON0.6). When the conversion is complete, the hardware will clear ADCS automatically, set ADCF (ADCCON0.7) and generate an interrupt if enabled. The new conversion result will also be stored in ADCRH (most significant 8 bits) and ADCRL (least significant 4 bits). The 12-bit ADC result value is $4095 \times \frac{V_{AIN}}{V_{REF}}$.

By the way, digital circuitry inside and outside the device generates noise which might affect the accuracy of ADC measurements. If conversion accuracy is critical, the noise level can be reduced by applying the following techniques:

1. Keep analog signal paths as short as possible. Make sure to run analog signals tracks well away from high-speed digital tracks.

2. Place the device in Idle mode during a conversion.

3. If any ADC_CHn pins are used as digital outputs, it is essential that these do not switch while a conversion is in progress.

21.2.2 ADC Conversion Triggered by External Source

Besides setting ADCS via software, the ML51 is enhanced by supporting hardware triggering method to start an A/D conversion. If ADCEX (ADCCON1.1) is set, edges or period points on selected PWM channel or edges of STADC pin will automatically trigger an A/D conversion. (The hardware trigger also sets ADCS by hardware.)

The effective condition is selected by ETGSEL (ADCCON0[5:4]) and ETGTYP (ADCCON1[3:2]). A trigger delay can also be inserted between external trigger point and A/D conversion. The external trigging ADC hardware with controllable trigger delay makes the ML51 feasible for high performance motor control. Note that during ADC is busy in converting (ADCS = 1), any conversion triggered by software or hardware will be ignored and there is no warning presented.

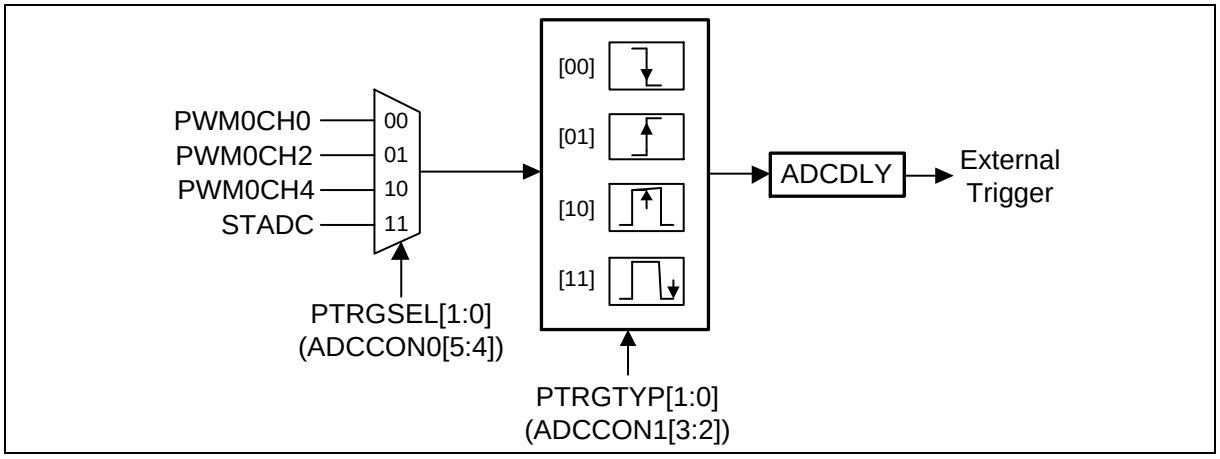


Figure 21.2-1 External Triggering ADC Circuit

21.2.3 ADC Conversion Result Comparator

The ML51 ADC has a digital comparator, which compares the A/D conversion result with a 12-bit constant value given in ACMPH and ACMPL registers. The ADC comparator is enabled by setting ADCMPEN (ADCCON2.5) and each compare will be done on every A/D conversion complete moment. ADCMPO (ADCCON2.4) shows the compare result according to its output polarity setting bit ADCMPOP (ADCCON2.6). The ADC comparing result can trigger a PWM Fault Brake output directly. This function is enabled when ADFBEN (ADCCON2.7). When ADCMPO is set, it generates a ADC compare event and asserts Fault Brake. Please also see Sector 18.1.5“Fault Brake”.

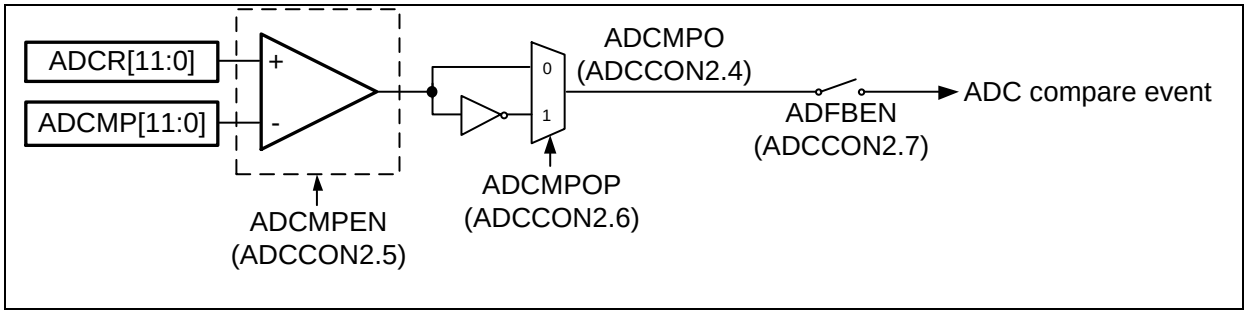


Figure 21.2-2 ADC Result Comparator

21.2.4 ADC Continues Conversion

The ADC controller supports DMA function, which auto store continues the A/D conversion result. The ADC DMA mode can store 12-bit ADC result into XRAM buffer, and 12-bit ADC data will auto-divide 8-bit high byte and 4-bit low nibble data two part. For reduce XRAM memory size, two 4-bit nibble data (continuing ADC conversion results) are automatically combine into one byte size and stored in XRAM.

The store method as below illustrate. It will store 8-bit of conversion high byte data first, then store combine data, the split point is ADC continue conversion length which define by ADCCN.

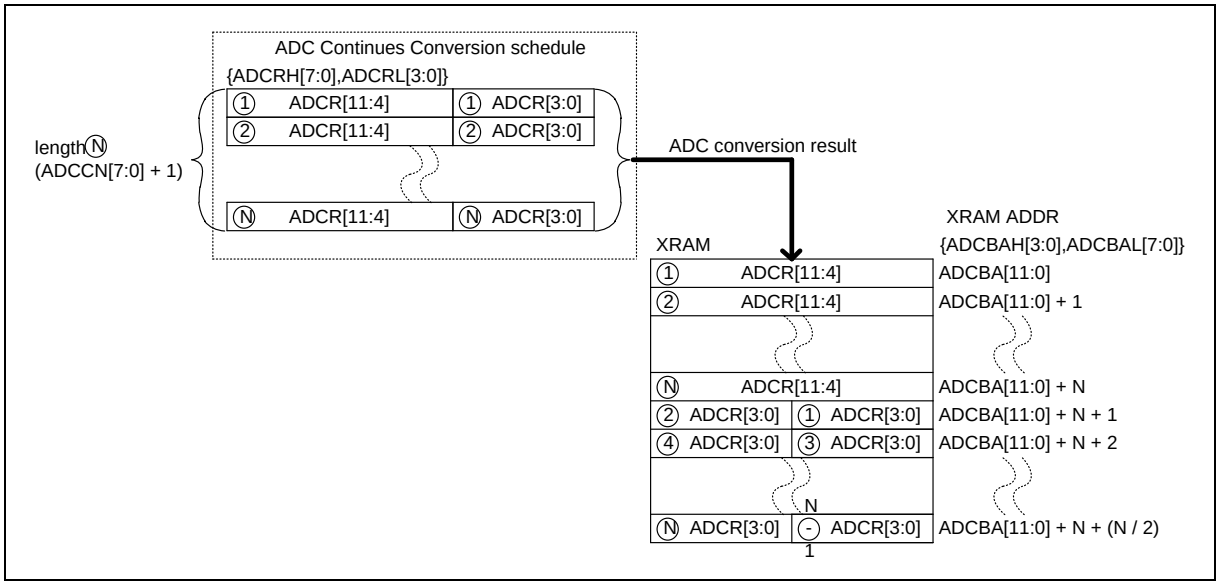


Figure 21.2-3 ADC Continues mode with DMA

A programing sequence is described below.

1 Set ADC channel and enable ADC as same as normal ADC setting method.

2 Set CONT (ADCCON1.4)to one for set ADC into continues conversion mode.

3 Set ADCBAH and ADCBAL registers to configure store address of conversion result.

4 Set ADCCN register to configure ADC conversion count.

5 Set HIE/FIE (ADCCON1[5]) to enable ADC conversion half done interrupt. (optional)

6 Start ADC RUN by software trigger (ADCS=1) or external trigger (ADCEX=1).

21.3 Control Registers of ADC

ADCCON0 – ADC Control register 0

Register	SFR Address	Reset Value
ADCCON0	A1H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCF	ADCS	ETGSEL1	ETGSEL0	ADCHS3	ADCHS2	ADCHS1	ADCHS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	ADCF	**ADC flag** This flag is set when an A/D conversion is completed in single sampling mode, final sampling complete in continue sampling mode or comparing hit if result comparator is enabled. The ADC result can be read. While this flag is 1, ADC cannot start a new converting. This bit is cleared by software.

Bit	Name	Description
6	ADCS	**A/D converting software start trigger** Setting this bit 1 triggers an A/D conversion. This bit remains logic 1 during A/D converting time and is automatically cleared via hardware right after conversion complete. The meaning of writing and reading ADCS bit is different. Writing: 0 = No effect. 1 = Start an A/D converting. Reading: 0 = ADC is in idle state. 1 = ADC is busy in converting.
5:4	ETGSEL[1:0]	**External trigger source select** When ADCEX (ADCCON1.1) is set, these bits select which pin output triggers ADC conversion. 00 = PWM0CH0. 01 = PWM0CH2. 10 = PWM0CH4. 11 = STADC pin.
3:0	ADCHS[3:0]	**A/D converting channel select** This filed selects the activating analog input source of ADC. If ADCEN is 0, all inputs are disconnected. 0000 = ADC_CH0. 0001 = ADC_CH1. 0010 = ADC_CH2. 0011 = ADC_CH3. 0100 = ADC_CH4. 0101 = ADC_CH5. 0110 = ADC_CH6. 0111 = ADC_CH7. 1000 = VBG (Internal band-gap voltage 1.22V). 1001 = VTEMP. (Temperature Sensor). Others = Reserved.

ADCCON1 – ADC Control 1

Register	SFR Address	Reset Value
ADCCON1	E1H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	HIE	CONT	ETGTYP[1:0]		ADCEX	ADCEN
-	-	R/W	R/W	R/W		R/W	R/W

Bit	Name	Description
7:6	-	Reserved

Bit	Name	Description
5	HIE	**ADC Half Done Interrupt Enable** 0 = ADC interrupt is not set while half of A/D conversions are complete in continue mode 1 = ADC interrupt is set while half of A/D conversions are complete in continue mode
4	CONT	**ADC Continue Sampling select** 0 = ADC single sampling, ADC interrupt is set while an A/D conversion is completed 1 = ADC continue sampling. ADC interrupt is set while total A/D conversions are completed
3:2	ETGTYP[1:0]	**External trigger type select** When ADCEX (ADCCON1.1) is set, these bits select which condition triggers ADC conversion. 00 = Falling edge on PWM0/2/4 or STADC pin. 01 = Rising edge on PWM0/2/4 or STADC pin. 10 = Central point of a PWM period. 11 = End point of a PWM period. Note that the central point interrupt or the period point interrupt is only available for PWM center-aligned type.
1	ADCEX	**ADC external conversion trigger select** This bit to select the methods of triggering an A/D conversion. 0 = A/D conversion is started only via setting ADCS bit. 1 = A/D conversion is started via setting ADCS bit or by external trigger source depending on ETGSEL[1:0] and ETGTYP[1:0]. Note that while ADCS is 1 (busy in converting), the ADC will ignore the following external trigger until ADCS is hardware cleared.
0	ADCEN	**ADC enable** 0 = ADC circuit off. 1 = ADC circuit on.

ADCCON2 – ADC Control 2

Register	SFR Address	Reset Value
ADCCON2	E2H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADFBEN	ADCMPOP	ADCMPEN	ADCMPO	ADCAQT[2:0]			ADCDLY.8
R/W	R/W	R/W	R	R/W			R/W

Bit	Name	Description
7	ADFBEN	ADC compare result asserting Fault Brake enable 0 = ADC asserting Fault Brake Disabled. 1 = ADC asserting Fault Brake Enabled. Fault Brake is asserted once its compare result ADCMPO is 1. Meanwhile, PWM channels output Fault Brake data. PWMRUN (PWMnCON0.7) will also be automatically cleared by hardware. The PWM output resumes when PWMRUN is set again.

Bit	Name	Description
6	ADCMPOP	ADC comparator output polarity 0 = ADCMPO is 1 if ADCR[11:0] is greater than or equal to ADCMP[11:0]. 1 = ADCMPO is 1 if ADCR[11:0] is less than ADCMP[11:0].
5	ADCMPEN	ADC result comparator enable 0 = ADC result comparator Disabled. 1 = ADC result comparator Enabled.
4	ADCMPO	ADC comparator output value This bit is the output value of ADC result comparator based on the setting of ACMPOP. This bit updates after every A/D conversion complete.
3:1	ADCAQT	ADC acquisition time This 3-bit field decides the acquisition time for ADC sampling, following by equation below: ADC acquisition time = $\frac{4 * ADCAQT + 10}{F_{ADC}}$. The default and minimum acquisition time is 10 ADC clock cycles. Note that this field should not be changed when ADC is in converting.
0	ADCDLY.8	ADC external trigger delay counter bit 8 See ADCDLY register.

ADCDLY – ADC Trigger Delay Counter

Register	SFR Address	Reset Value
ADCDLY	E3H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCDLY[7:0]							
R/W							

Bit	Name	Description
7:0	ADCDLY[7:0]	ADC external trigger delay counter low byte This 8-bit field combined with ADCCON2.0 forms a 9-bit counter. This counter inserts a delay after detecting the external trigger. An A/D converting starts after this period of delay. External trigger delay time = $\frac{ADCDLY}{F_{ADC}}$. Note that this field is valid only when ADCEX (ADCCON1.1) is set. User should not modify ADCDLY during PWM run time if selecting PWM output as the external ADC trigger source.

AINDIDS – ADC Channel Digital Input Disconnect

Register	SFR Address	Reset Value
AINDIDS	CEH, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
AIN7DIDS	AIN6DIDS	AIN5DIDS	AIN4DIDS	AIN3DIDS	AIN2DIDS	AIN1DIDS	AIN0DIDS
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:0	AINnDIDS	**ADC Channel digital input disable** 0 = Enabled digital input at ADC channel n. 1 = Disabled digital input at ADC channel n . ADC channel n is read always 0.

ADCRH – ADC Result High Byte

Register	SFR Address	Reset Value
ADCRH	C3H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCR[11:4]							
R							

Bit	Name	Description
7:0	ADCR[11:4]	ADC result high byte The most significant 8 bits of the ADC result stored in this register.

ADCRL – ADC Result Low Byte

Register	SFR Address	Reset Value
ADCRL	C2H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	ADCR[3:0]			
-	-	-	-	R			

Bit	Name	Description
7:4	-	Reserved
3:0	ADCR[3:0]	**ADC result low byte** The least significant 4 bits of the ADC result stored in this register.

ADCMPH – ADC Compare High Byte

Register	SFR Address	Reset Value
AINDIDS	CFH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ADCMP[11:4]							
W/R							

Bit	Name	Description
7:0	ADCMP[11:4]	**ADC compare high byte** The most significant 8 bits of the ADC compare value stores in this register.

ADCMPL – ADC Compare Low Byte

Register	SFR Address	Reset Value
ADCMPL	CEH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	ADCMP[3:0]			
-	-	-	-	W/R			

Bit	Name	Description
7:4	-	Reserved
3:0	ADCMP[3:0]	**ADC compare low byte** The least significant 4 bits of the ADC compare value stores in this register.

ADCBAH – ADC RAM Base Address High byte

7	6	5	4	3	2	1	0
-				ADCBAH[3:0]			
-				R/W			

Address: E4H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:4	-	Reserved
3:0	ADCBAH[3:0]	ADC RAM base address (High byte) The most significant 4 bits of RAM base address to store ADC continue sampling data. RAM base address ADCBA[11:0] = {ADCBAH[3:0], ADCBAL[7:0]}

ADCBAL – ADC RAM Base Address Low Byte

7	6	5	4	3	2	1	0
ADCBAL[7:0]							
R/W							

Address: CBH, Page:0 Reset value: 0000 0000b

Bit	Name	Description
7:0	ADCBAL[7:0]	ADC RAM base address (Low byte) The least significant 8 bits of RAM base address to store ADC continue sampling data. RAM base address ADCBA[11:0] = { ADCBAH[3:0], ADCBAL[7:0]}

ADCSN – ADC Sampling Number

7	6	5	4	3	2	1	0
ADCSN[7:0]							
R/W							

Address: E5H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:0	ADCSN[7:0]	ADC Sampling Number The total sampling numbers for ADC continue sampling select. Total sampling number= ADCSN[7:0] + 1

ADCCN – ADC Current Sampling Number

7	6	5	4	3	2	1	0
ADCCN[7:0]							
R/W							

Address: E6H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:0	ADCCN[7:0]	ADC Current Sampling Number The current sampling numbers for ADC continue sampling select. The current sampling number= ADCCN[7:0] + 1

ADCSR – ADC Status Register

7	6	5	4	3	2	1	0
SLOW	ADCDIV[2:0]			-	CMPHIT	HDONE	FDONE
R/W	R/W			-	R/W	R/W	R/W

Address: E7H, Page 0 Reset value: 0000 0000b

Bit	Name	Description

Bit	Name	Description
7	SLOW	ADC Slow Speed Selection This bit is used to select ADC low speed. 0 = high speed 500 ksps 1 = low speed 200 ksps
6:4	ADCDIV	ADC clock divider 000 = FADC is FSYS/1. 001 = FADC is FSYS/2. 010 = FADC is FSYS/4. 011 = FADC is FSYS/8. 100 = FADC is FSYS/16. 101 = FADC is FSYS/32. 110 = FADC is FSYS/64. 111 = FADC is FSYS/128.
3	-	Reserved
2	CMPHIT	ADC comparator Hit Flag This bit is set by hardware when ADCMPO (ADCCON2.4) flag rising **Note:** This bit can be cleared by writing 0 to it.
1	HDONE	A/D Conversion Half Done Flag This bit is set by hardware when half of ADCSN A/D conversions are complete in continue mode. **Note:** This bit can be cleared by writing 0 to it
0	FDONE	A/D Conversion Full Done Flag This bit is set by hardware when all of ADCSN A/D conversions are complete in continue mode or single conversion in single mode. **Note:** This bit can be cleared by writing 0 to it..

22 VOLTAGE REFERENCE

The V_{REF} pin is for analog multiplexer, such as ADC, ACMP. It default be used as an external source(set ENVRF = 0). It also could be configurable as on-chip reference voltage generator (V_{REF_IN}) by setting ENVRF = 1 (seeFigure 22.1-1 VREF Block Diagram). The output voltage is selectable 1.538 V, 2.048 V, 20560 V, 3.072 V and 4.096 V by setting VRFSEL[2:0]. The maximum load of the V_{REF_IN} must be less than 200 uA to AV_{SS}. Set pre-load is to reduce stable time of V_{REF_IN}. At first enable V_{REF_IN} and turn on pre-load at the same time, the minimum stable time of pre-load on the V_{REF_IN} must be greater than 3 ms. After the V_{REF_IN} stable, user should be turn off pre-load to avoid any interference on analog multiplexer. Pre-load is only for internal V_{REF} use

22.1 Voltage Reference Block Diagram

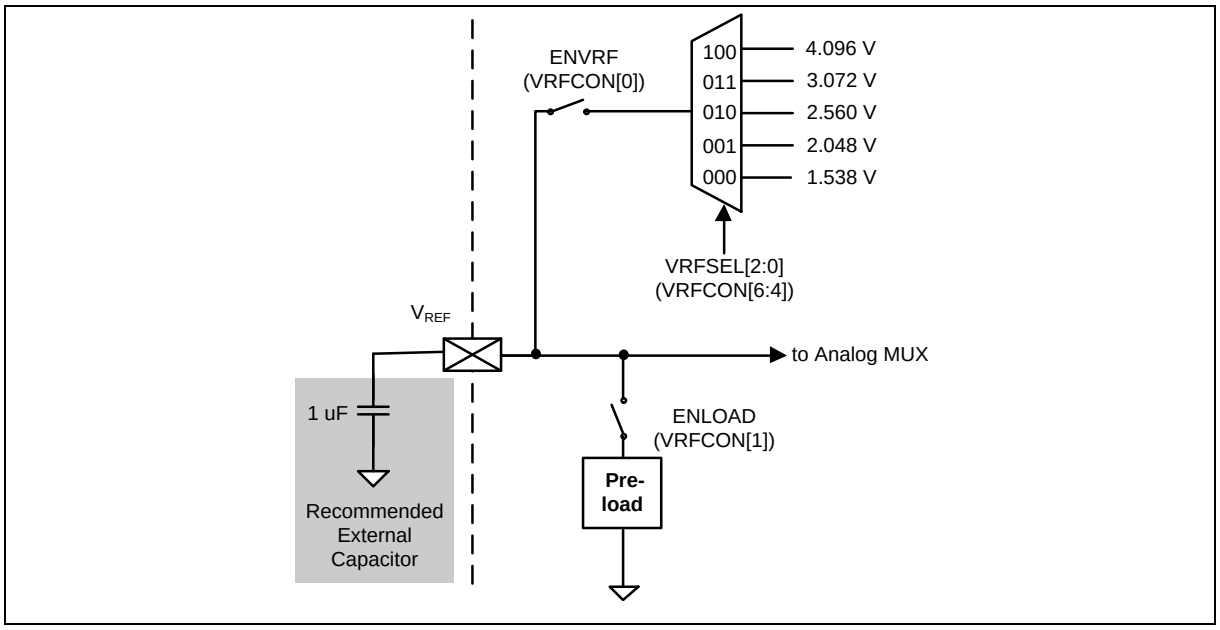


Figure 22.1-1 V_{REF} Block Diagram

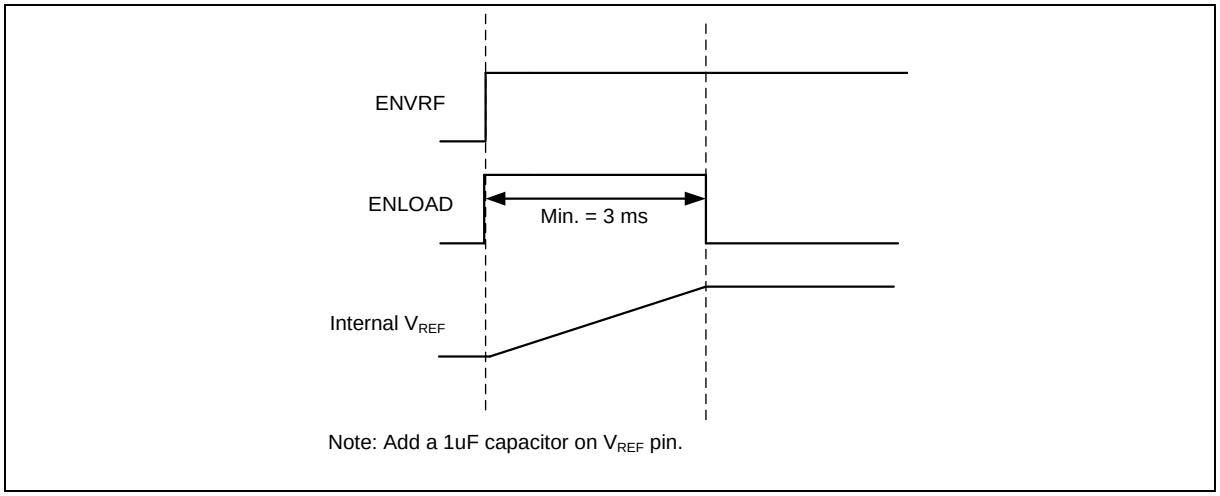


Figure 22.1-2 Pre-load Timing

22.2 Control Register of Voltage Reference

VRFCON – Internal V_{REF} Control

Register	SFR Address	Reset Value
VRFCON	A9H, Page 1, TA protected	0000 _0000 b

7	6	5	4	3	2	1	0
-	VRFSEL[2:0]			-	-	ENLOAD	ENVRF
-	R/W			-	-	R/W	R/W

Bit	Name	Description
7	-	Reserved
6:4	VRFSEL	**Internal V_{REF} Output Voltage Select** This field selects V_{REF} output voltage. 000 = 1.538V , when V_{DD} > 2.0V 001 = 2.048V , when V_{DD} > 2.4V 010 = 2.560V , when V_{DD} > 2.9V 011 = 3.072V , when V_{DD} > 3.4V 100 = 4.096V , when V_{DD} > 4.5V 101 = reserved 110 = reserved 111 = reserved
3:2	-	Reserved
1	ENLOAD	**Internal V_{REF} Pre-Load enable** 1 = Internal V_{REF} Pre-load Enabled. 0 = Internal V_{REF} Pre-load Disabled
0	ENVRF	**Internal V_{REF} enable** 1 = Internal V_{REF} Enabled, 0 = Internal V_{REF} Disabled Note that a 1 uF has to add on V_{REF} pin while internal V_{REF} is enabled.

23 ANALOG COMPARATOR CONTROLLER (ACMP)

The ML51 series contains two comparators. The comparator output is logic 1 when positive input is greater than negative input; otherwise, the output is 0. The comparator can be configured to generate an interrupt when the comparator output value changes.

- Analog input voltage range: 0 ~ AV_{DD}(voltage of AV_{DD} pin)
- Supports hysteresis function
- Supports wake-up function
- Selectable input sources of negative input
- Comparator ACMP0 supports
  - 4 positive source
    - P2.5 (ACMPn_P0)
    - P2.3 (ACMPn_P1)
    - P2.1 (ACMPn_P2)
    - P3.1 (ACMPn_P3)
  - 4 negative sources
    - P2.4 (ACMP0_N0)
    - Comparator Reference Voltage (CRV)
    - VBG (BANDGAP voltage)
    - P2.0 (ACMP0_N1)
- Comparator ACMP1 supports
  - 4 positive source
    - P2.5 (ACMPn_P0)
    - P2.3 (ACMPn_P1)
    - P2.1 (ACMPn_P2)
    - P3.1 (ACMPn_P3)
  - 3 negative sources
    - P2.2 (ACMP1_N0)
    - Comparator Reference Voltage (CRV)
    - VBG (BANDGAP voltage)
    - P3.2 (ACMP1_N1)

23.1 ACMP block Diagram

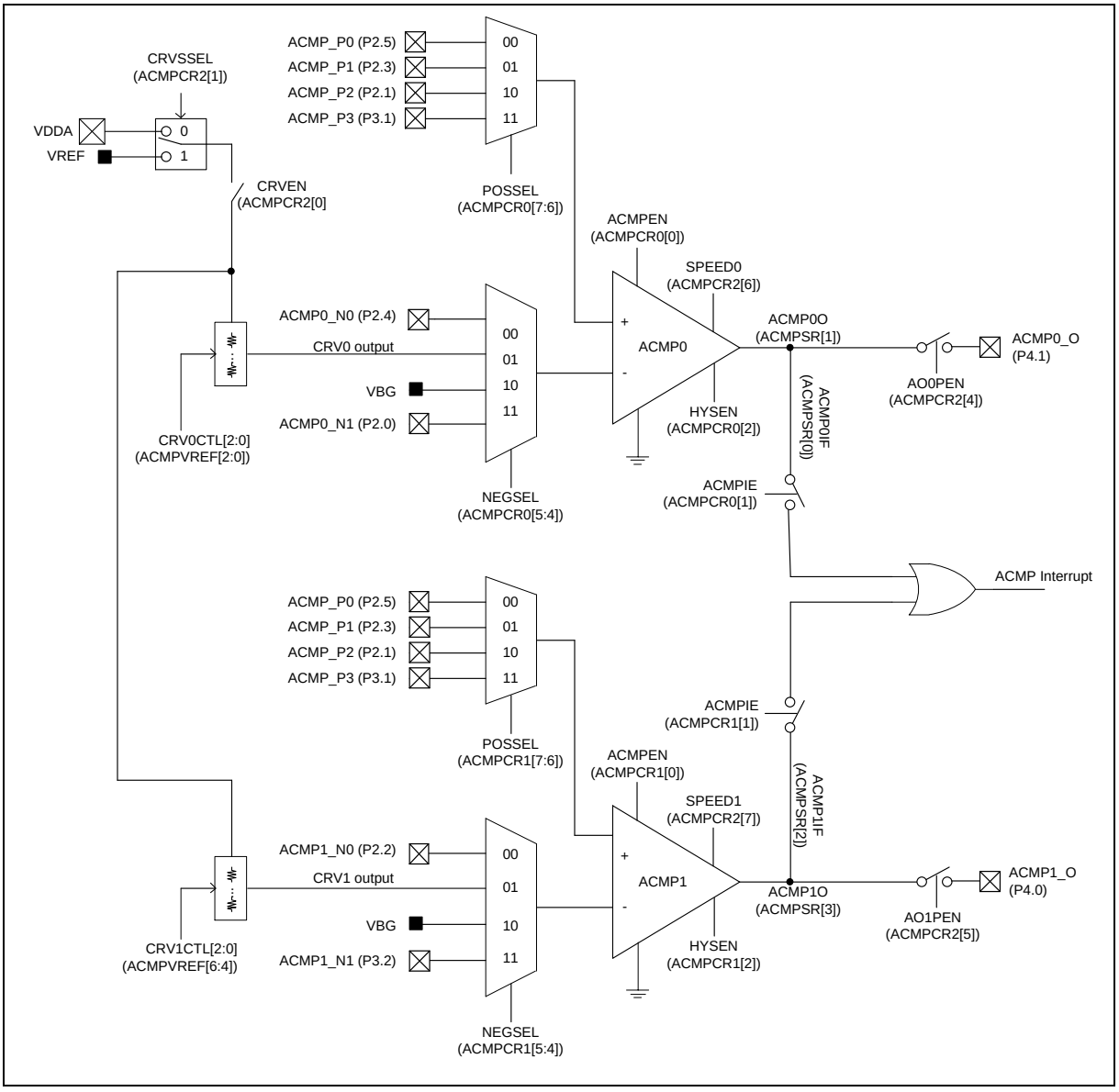


Figure 23.1-1 Analog Comparator Block Diagram

23.2 Functional Description

23.2.1 Hysteresis Function

The analog comparator provides the hysteresis function to make the comparator to have a stable output transition. If comparator output is 0, it will not be changed to 1 until the positive input voltage exceeds the negative input voltage by a high threshold voltage. Similarly, if comparator output is 1, it will not be changed to 0 until the positive input voltage drops below the negative input voltage by a low threshold voltage.

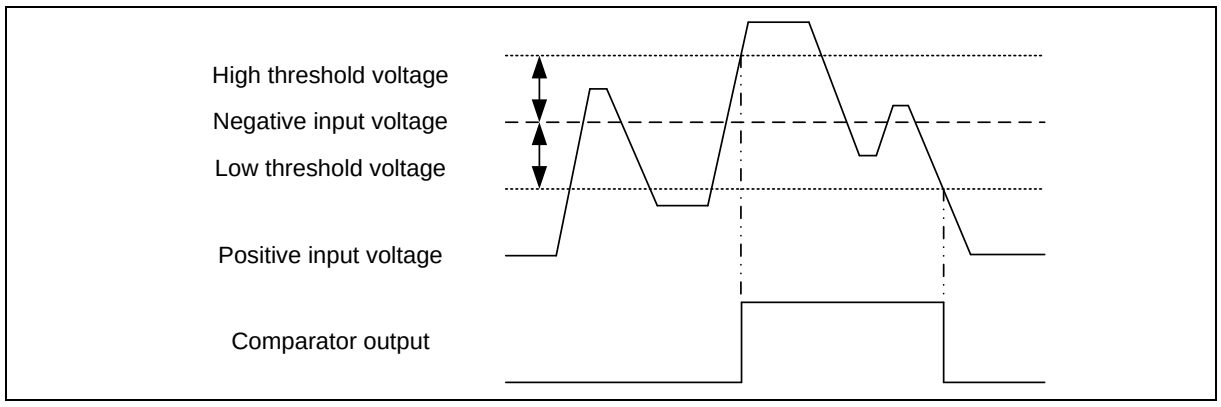


Figure 23.2-1 Comparator Hysteresis Function

23.2.2 Comparator Reference Voltage (CRV)

The comparator reference voltage (CRV) module is responsible for generating reference voltage for comparators. The CRV module consists of resisters ladder and analog switch. User can set the CRV output voltage by setting the CRVnCTL(ACMPVREF). The CRV output voltage can be selected as the negative input of comparator by setting NEGSEL (ACMPCR0[5:4]).

Features:

1. User selectable references voltage source by setting the CRVSSEL(ACMPCR2[1]) register.
2. User selectable references voltage by setting the CRVnCTL(ACMPVREF) register.

Comparator reference voltage = V_{IN} * (1/6+CRVnCTL/12); V_{IN} = AV_{DD} or $V_{REF.}$

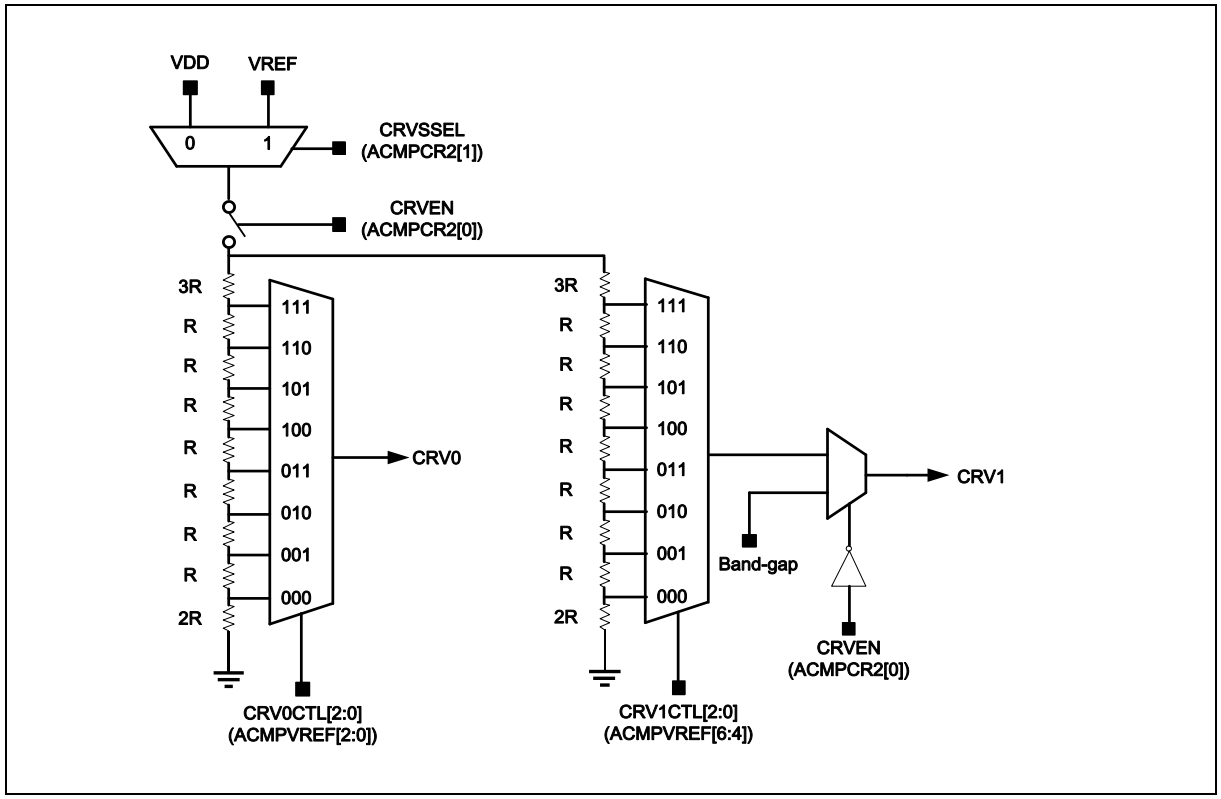


Figure 23.2-2 Comparator Reference Voltage Block Diagram

Note that If CRVEN = 0, CRV0 is equal to 0 and CRV1 is equal to Band-gap.

23.2.3 Interrupt Sources

The comparator generates an output ACMPnO (ACMPSR). If the ACMPIE (ACMPCR0[1]) bit in ACMPCR0 is set, a state change on the comparator output ACMPnO (ACMPSR) will cause comparator flag ACMPnIF (ACMPSR) be set and the comparator interrupt requested. User can write 1 to ACMPnIF (ACMPSR) through software to stop interrupt request.

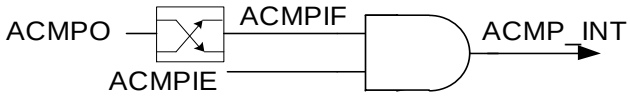


Figure 23.2-3 Analog Comparator Interrupt Sources

23.3 Control Registers of ACMP Controller

ACMPCR0 – Analog Comparator Control Register 0

7	6	5	4	3	2	1	0
POSSEL		NEGSEL		WKEN	HYSEN	ACMPIE	ACMPEN
R/W		R/W		R/W	R/W	R/W	R/W

Address: D2H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:6	POSSEL	Comparator 0 positive Input Selection 00 = ACMP0_P0 (P2.5) pin. 01 = ACMP0_P1 (P2.3) pin. 10 = ACMP0_P2 (P2.1) pin. 11 = ACMP0_P3 (P3.1) pin.
5:4	NEGSEL	Comparator 0 Negative Input Selection 00 = ACMP0_N0 (P2.4) pin. 01 = Internal comparator reference voltage (CRV). 10 = VBG (Bandgap). 11 = ACMP0_N1 (P2.0)pin.
3	WKEN	Comparator 0 Power-down Wake-up Enable Bit 0 = Comparator 0 Wake-up function Disabled. 1 = Comparator 0 Wake-up function Enabled.
2	HYSEN	Comparator 0 Hysteresis Enable Bit 0 = Comparator 0 hysteresis Disabled. 1 = Comparator 0 hysteresis Enabled.
1	ACMPIE	Comparator 0 Interrupt Enable Bit 0 = Comparator 0 interrupt Disabled. 1 = Comparator 0 interrupt Enabled. If WKEN (ACMPCR1[3]) is set to 1, the wake-up interrupt function will be enabled as well.
0	ACMPEN	Comparator 0 Enable Bit 0 = Comparator 0 Disabled. 1 = Comparator 0 Enabled.

ACMPCR1 – Analog Comparator Control Register 1

7	6	5	4	3	2	1	0
POSSEL		NEGSEL		WKEN	HYSEN	ACMPIE	ACMPEN
R/W		R/W		R/W	R/W	R/W	R/W

Address: D3H, Page 0 Reset value: 0000 0000b

Bit	Name	Description

Bit	Name	Description
7:6	POSSEL	Comparator 1 positive Input Selection 00 = ACMP1_P0 (P2.5) pin. 01 = ACMP1_P1 (P2.3) pin. 10 = ACMP1_P2 (P2.1) pin. 11 = ACMP1_P3 (P3.1) pin.
5:4	NEGSEL	Comparator 1 Negative Input Selection 00 = ACMP1_N0 (P2.2) pin. 01 = Internal comparator reference voltage (CRV). 10 = VBG (Bandgap). 11 = ACMP1_N1 (P3.2)pin.
3	WKEN	Comparator 1 Power-down Wake-up Enable Bit 0 = Comparator 1 Wake-up function Disabled. 1 = Comparator 1 Wake-up function Enabled.
2	HYSEN	Comparator 1 Hysteresis Enable Bit 0 = Comparator 1 hysteresis Disabled. 1 = Comparator 1 hysteresis Enabled.
1	ACMPIE	Comparator 1 Interrupt Enable Bit 0 = Comparator 1 interrupt Disabled. 1 = Comparator 1 interrupt Enabled. If WKEN (ACMPCR2[3]) is set to 1, the wake-up interrupt function will be enabled as well.
0	ACMPEN	Comparator 1 Enable Bit 0 = Comparator 1 Disabled. 1 = Comparator 1 Enabled.

ACMPCR2 – Analog Comparator Control Register 2

7	6	5	4	3	2	1	0
-	-	AO1PEN	AO0PEN	-	-	CRVSSEL	CRVEN
-	-	R/W	R/W	-	-	R/W	R/W

Address: ABH, Page 1 Reset value: 0000 0000b

Bit	Name	Description
7:6	-	Reserved
5	AO1PEN	Analog comparator 1 output to pin enable control 0 = pin output disabled 1 = pin output to P4.0 Enabled
4	AO0PEN	Analog comparator 0 output to pin enable control 0 = pin output disabled 1= output to P4.1 Enabled

Bit	Name	Description
3:2	-	Reserved
1	CRVSSEL	CRV Source Voltage Selection 0 = V_{DD} is selected as CRV source voltage. 1 = The reference voltage (V_{REF}) is selected as CRV source voltage.
0	CRVEN	CRV Enable Bit 0 = CRV Disabled. 1 = CRV Enabled.

ACMPSR – Analog Comparator Status Register

7	6	5	4	3	2	1	0
-	-	-	-	ACMP1O	ACMP1IF	ACMP0O	ACMP0IF
-	-	-	-	R	R/W	R	R/W

Address: D4H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:4	-	Reserved
3	ACMP1O	Comparator 1 Output Synchronized to the PCLK to allow reading by software. Cleared when the comparator 1 is disabled, i.e. ACMPEN (ACMPCR1[0]) is cleared to 0. **Note:** This bit is read only.
2	ACMP1IF	Comparator 1 Interrupt Flag This bit is set by hardware whenever the comparator 1 output changes state. This will generate an interrupt if ACMPIE (ACMPCR1[1]) is set to 1 **Note:** Write “0” to clear this bit to 0.
1	ACMP0O	Comparator 0 Output Synchronized to the PCLK to allow reading by software. Cleared when the comparator 0 is disabled, i.e. ACMPEN (ACMPCR0[0]) is cleared to 0. **Note:** This bit is read only.
0	ACMP0IF	Comparator 0 Interrupt Flag This bit is set by hardware whenever the comparator 0 output changes state. This will generate an interrupt if ACMPIE (ACMPCR0[1]) is set to 1 **Note:** Write “0” to clear this bit to 0.

ACMPV_{REF} – ACMP Reference Voltage Control Register

7	6	5	4	3	2	1	0
AO1PIV	CRV1CTL[2:0]			AO0PIV	CRV0CTL[2:0]		
R/W	R/W			R/W	R/W		

Address: D5H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7	AO1PIV	Comparator 1 Output Invert Function 0 = ACMP1 output signal to P4.0 as normal. 1 = ACMP1 output signal to P4.0 value invert.
6:4	CRV1CTL[2:0]	Comparator 1 Reference Voltage Setting CRV1 = CRV source voltage * (2/12+CRV1CTL/12).
3	AO0PIV	Comparator 0 Output Invert Function 0 = ACMP0 output signal to P4.1 as normal. 1 = ACMP0 output signal to P4.1 value invert.
2:0	CRV0CTL[2:0]	Comparator 0 Reference Voltage Setting CRV0 = CRV source voltage * (2/12+CRV0CTL/12).

24 PDMA CONTROLLER (PDMA)

The ML51 provides peripheral direct memory access (PDMA) controller. The PDMA controller is used to provide high-speed data transfer between memory and peripherals or between memory and memory. The PDMA controller can transfer data from one address to another without CPU intervention. This has the benefit of reducing the workload of CPU and keeps CPU resources free for other applications.

- Supports transfer data width of 8 bits
- Supports software and SPI and SMC/UART request
- Supports source and destination address increment size can be byte
- Supports transfer done and half done interrupt

24.1 PDMA Block Diagram

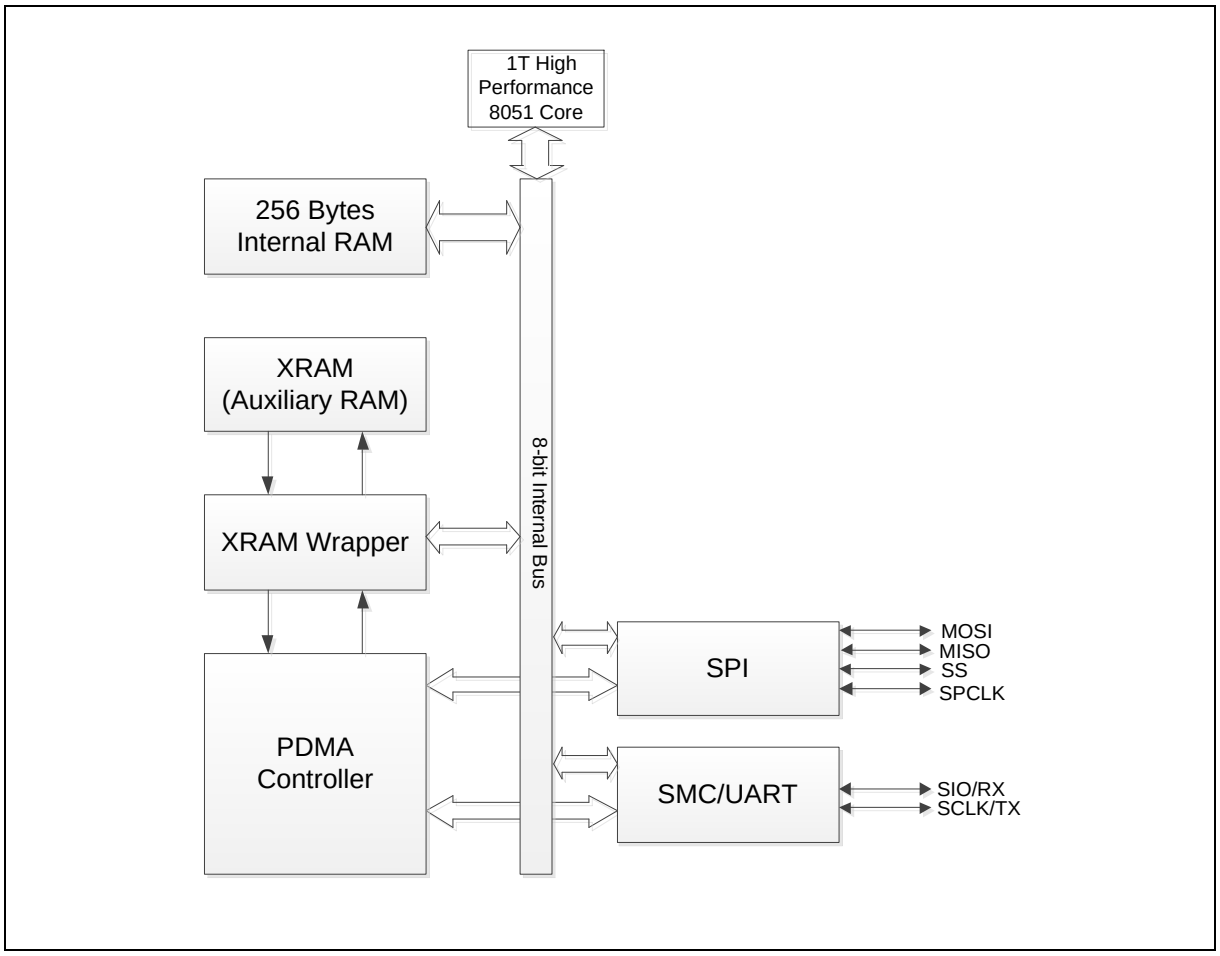


Figure 24.1-1 PDMA Interface Diagram

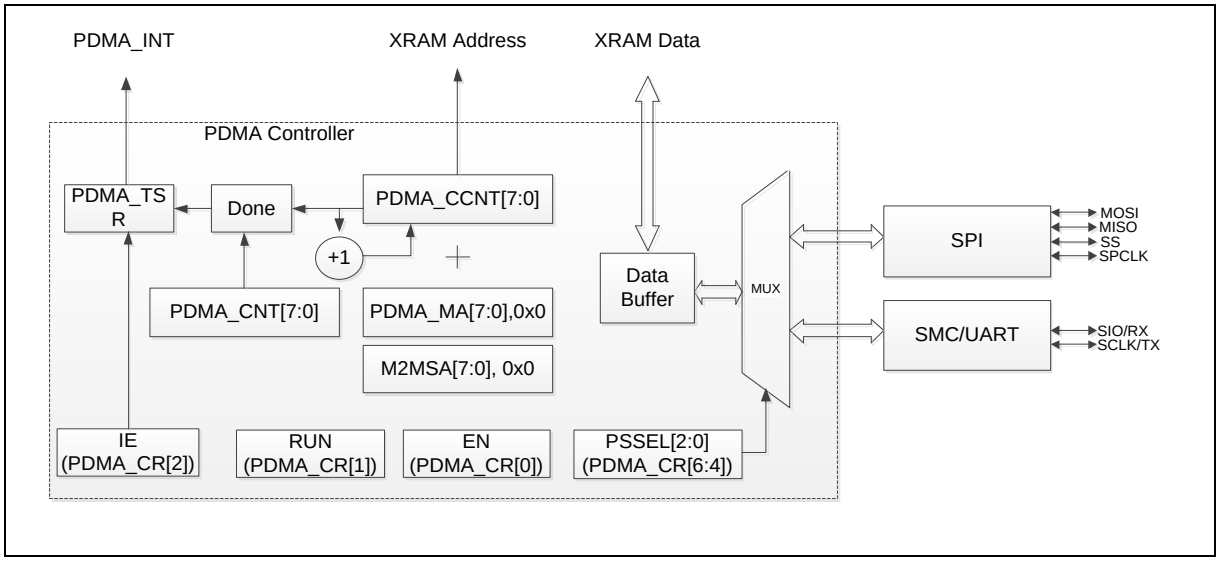


Figure 24.1-2 PDMA Controller Block Diagram

24.2 Functional Description

24.2.1 Operating Modes

Each PDMA channel behavior is not pre-defined, user must configure the channel service settings of PSSEL[2:0] registers before starting the related PDMA channel operation.

User must set EN DMAnCR[0] bit to enable PDMA channel. Then write a valid source address to the DMAnMA and DMAnBAH[3:0] register, a destination address to the MTMnDA and DMAnBAH[7:4]register if use memory to memory, and a transfer count to the DMAnCNT register. Next, trigger the RUN DMAnCR[1]. If the source address and destination are not in wrap around mode, the PDMA will continue the transfer until DMAnCCNT counts down to 0. In wrap around mode, when DMAnCCNT counts down to 0, the PDMA will reload DMAnCCNT and work around until user clears EN DMAnCR[0] bit to disable PDMA channel.

A programing sequence example is described below.

24.2.2 SPI peripheral to XRAM memory

1 Configure DMAnCR register to set EN DMAnCR[0] bit to enable PDMA channel.

2 Set PSSEL[2:0] = 001 SPI0 RX (, 011 SPI1 RX, 101 SPI0 TX or 111 SPI1 TX) (DMAnCR (n-1~2)) register to configure the channel service setting.

3 Set DMAnMA/DMAnBAH[3:0] registers to configure destination address.

4 Set DMAnCNT register to configure PDMA transfer count.

5 Set HIE/FIE DMAnCR[3]/[2] to enable PDMA transfer done interrupt (optional) then enable global interrupt.

6 Set RUN DMAnCR[0] bit to enable PDMA transfer.

7 Write “0” to HDONE and FDONE DMAnTSR[1][0] bit to clear interrupt flag if PDMA transfer done interrupt is generated.

8 Set RUN DMAnCR[0] bit to enable next PDMA transfer.

If an error occurs during the PDMA operation, the channel operation stops until user clears the error condition, , and then clears the EN DMAnCR[0] bit to disable the PDMA channel, then sets EN DMAnCR[0] bit and RUN DMAnCR[1] bit to start operation again.

24.2.3 SMC/UART peripheral to XRAM memory

1 Configure DMAnCR register to set EN DMAnCR[0] bit to enable PDMA channel.

2 Set PSSEL[2:0] = 010 SMC/UART RX (or 110 SMC/UART TX) (DMAnCR (n-1~2)) register to configure the channel service setting.

3 Set DMAnMA/DMAnBAH[3:0] registers to configure destination address.

4 Set DMAnCNT register to configure PDMA transfer count.

5 Set HIE/FIE DMAnCR[3]/[2] to enable PDMA transfer done interrupt (optional) then enable global interrupt.

6 Set RUN DMAnCR[0] bit to enable PDMA transfer.

7 Write “0” to HDONE and FDONE DMAnTSR[1][0] bit to clear interrupt flag if PDMA transfer done interrupt is generated.

8 Set RUN DMAnCR[0] bit to enable next PDMA transfer.

If an error occurs during the PDMA operation, the channel operation stops until user clears the error condition, , and then clears the EN DMAnCR[0] bit to disable the PDMA channel, then sets EN DMAnCR[0] bit and RUN DMAnCR[1] bit to start operation again.

24.2.4 Memory to Memory (XRAM) Transfer

1 Configure DMAnCR register to set EN DMAnCR[0] bit to enable PDMA channel.

2 Set PSSEL[2:0] = 000 (XRAM to XRAM) (DMAnCR (n-1~2)) register to configure the channel service setting.

3 Set DMAnMA/DMAnBAH[3:0] registers to configure source address.

4 Set DMAnDA/DMAnBAH[7:4] registers to configure destination address.

5 Set DMAnCNT register to configure PDMA transfer count.

6 Set HIE/FIE DMAnCR[3]/[2] to enable PDMA transfer done interrupt (optional) then enable global interrupt.

7 Set RUN DMAnCR[0] bit to enable PDMA transfer.

8 Write “0” to HDONE and FDONE DMAnTSR[1][0] bit to clear interrupt flag if PDMA transfer done interrupt is generated.

9 Set RUN DMAnCR[0] bit to enable next PDMA transfer.

If an error occurs during the PDMA operation, the channel operation stops until user clears the error condition and then clears the EN DMAnCR[0] bit to disable the PDMA channel, then sets EN DMAnCR[0] bit and RUN DMAnCR[1] bit to start operation again.

24.3 PDMA Control Registers

Note: this chapter register table include PDMA0~3 introduction. But following list products only support PDMA0~1, include ML51EC0AE, ML51UC0AE, ML51PC0AE, ML51TC0AE, ML51PB9AE, ML51TB9AE. And following product not support PDMA function include ML51BB9AE, ML51DB9AE, ML51FB9AE, ML51OB9AE, ML51XB9AE, ML51EB9AE, ML51UB9AE.

DMAnCR – PDMAn Control Register

Register	SFR Address	Reset Value
DMA0CR	92H, Page 0	0000_0000 b
DMA1CR	EBH, Page 0	0000_0000 b
DMA2CR	B3H, Page 2	0000_0000 b
DMA3CR	ABH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	PSSEL[2:0]			HIE	FIE	RUN	EN
-	R/W			R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6:4	PSSEL[2:0]	Peripheral Source Select 000 = XRAM to XRAM 001 = SPI0 RX 010 = SMC/UART RX. 011 = SPI1 RX 100 = Reserved, No peripheral source select 101 = SPI0 TX 110 = SMC/UART TX. 111 = SPI1 TX **Note:** 001~011 : peripheral devices to XRAM memory 101~111 : XRAM memory to peripheral devices
3	HIE	PDMA HALFTransfer Done Interrupt Enable Bit 0 = Interrupt Disabled when PDMA half transfer is done. 1 = Interrupt Enabled when PDMA half transfer is done.
2	FIE	PDMA Full Transfer Done Interrupt Enable Bit 0 = Interrupt Disabled when PDMA full transfer is done. 1 = Interrupt Enabled when PDMA full transfer is done.
1	RUN	Trigger Enable Bit 0 = No effect. 1 = PDMA data transfer Enabled. **Note1:** When PDMA transfer completed, this bit will be cleared automatically.

Bit	Name	Description
0	EN	PDMA Enable Bit Setting this bit to 1 enables PDMA operation. If this bit is cleared, PDMA will ignore all PDMA request and Reset the internal state machine, pointers and internal buffer. The contents of all control registers will not be cleared.

DMAnMAL – PDMA XRAM Base Address Low Byte

Register	SFR Address	Reset Value
DMA0MAL	93H, Page 0	0000_0000 b
DMA1MAL	ECH, Page 0	0000_0000 b
DMA2MAL	B4H, Page 2	0000_0000 b
DMA3MAL	ACH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
DMAnMAL[7:0]							
R/W							

Bit	Name	Description
7:0	MAL[7:0]	PDMA XRAM Base Address (Low Byte) The least significant 8 bits of XRAM address to store or read for the peripheral source data; in memory to memory transfer, this register is the source address. XRAM address = {MAH[3:0],MAL[7:0]}

DMAnBAH – PDMAn XRAM Base Address and Memory to Memory Destination Address High Byte

Register	SFR Address	Reset Value
DMA0BAH	F6H, Page 0	0000_0000 b
DMA1BAH	FDH, Page 0	0000_0000 b
DMA2BAH	B2H, Page 2	0000_0000 b
DMA3BAH	AAH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
MDAH[3:0]				MAH[3:0]			
R/W				R/W			

Bit	Name	Description

Bit	Name	Description
7:4	MDAH[7:0]	Memory to Memory Destination Address (High Byte) The most significant 4 bits of XRAM address are used for memory to memory destination address. XRAM destination address = {MDAH[3:0], MDAL[7:0]}
3:0	MAH[3:0]	PDMA XRAM Base Address (High Byte) The most significant 4 bits of XRAM address to store or read for the peripheral source data; in memory to memory transfer, this register is the destination address. XRAM address = {MAH[3:0], MAL[7:0]}

DMAnCNT – PDMA Transfer Count

Register	SFR Address	Reset Value
DMA0CNT	94H, Page 0	0000_0000 b
DMA1CNT	EDH, Page 0	0000_0000 b
DMA2CNT	B5H, Page 2	0000_0000 b
DMA3CNT	ADH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
CNT[7:0]							
R/W							

Bit	Name	Description
7:0	CNT[7:0]	PDMA Transfer Count The total transfer count for PDMA request operation. Total transfer count = CNT[7:0] + 1

DMAnCCNT – PDMA Current Transfer Count

Register	SFR Address	Reset Value
DMA0CCNT	95H, Page 0	0000_0000 b
DMA1CCNT	EEH, Page 0	0000_0000 b
DMA2TSR	B6H, Page 2	0000_0000 b
DMA3TSR	AEH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
CCNT[7:0]							
R/W							

Bit	Name	Description
7:0	CNT[7:0]	PDMA Current Transfer Count The current transfer count for PDMA request operation. Current transfer count = CCNT[7:0] **Note:** while DMAnCNT=0xFF (total transfer count = 256) and DMAnCCNT = 0x00 , If PDMA FDONE flag (DMAnTSR[0])=0, that means, 1'st byte data is not complete.If PDMA FDONE flag (DMAnTSR[0])=1, that means, all of data are transferred..

DMAnTSR – PDMAn Transfer Status Register

Register	SFR Address	Reset Value
DMA0TSR	E9H, Page 0	0000_0000 b
DMA1TSR	F1H, Page 0	0000_0000 b
DMA2TSR	B1H, Page 2	0000_0000 b
DMA3TSR	A9H, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	-	-	ACT	HDONE	FDONE
-	-	-	-	-	R	R/W	R/W

Bit	Name	Description
7:3	-	Reserved
2	ACT	PDMA in Active Status Flag (Read Only) 0 = This bit is cleared automatically when PDMA transfer is done or disabled. 1 = This bit is set by hardware when PDMA transfer is in active.
1	HDONE	PDMA Half Transfer Done Flag This bit is set by hardware when PDMA half transfer is done. **Note:** This bit can be cleared by writing 0 to it.
0	FDONE	PDMA Full Transfer Done Flag This bit is set by hardware when PDMA full transfer is done. **Note:** This bit can be cleared by writing 0 to it.

MTMnDA – Memory to Memory Destination Address Low Byte

Register	SFR Address	Reset Value
MTM0DA	EAH, Page 0	0000_0000 b
MTM1DA	F2H, Page 0	0000_0000 b
MTM2DA	B7H, Page 2	0000_0000 b
MTM3DA	AFH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
MDAL[7:0]							
R/W							

Bit	Name	Description
7:0	MDAL[7:0]	Memory to Memory Destination Address (Low Byte) The least significant 8 bits of XRAM address are used for memory to memory destination address. XRAM destination address = {MDAH[3:0], MDAL[7:0]}

25 TIMED ACCESS PROTECTION (TA)

The ML51 has several features such as WDT and Brown-out detection that are crucial to proper operation of the system. If leaving these control registers unprotected, errant code may write undetermined value into them and results in incorrect operation and loss of control. To prevent this risk, the ML51 has a protection scheme, which limits the write access to critical SFR. This protection scheme is implemented using a timed access (TA). The following registers are related to the TA process.

TA – Timed Access

Register	SFR Address	Reset Value
TA	C7H, All pages	0000_0000 b

7	6	5	4	3	2	1	0
TA[7:0]							
W							

Bit	Name	Description
7:0	TA[7:0]	**Timed access** The timed access register controls the access to protected SFR. To access protected bits, user should first write AAH to the TA and immediately followed by a write of 55H to TA. After these two steps, a writing permission window is opened for 4 clock cycles during this period that user may write to protected SFR.

In timed access method, the bits, which are protected, have a timed write enable window. A write is successful only if this window is active, otherwise the write will be discarded. When the software writes AAH to TA, a counter is started. This counter waits for 3 clock cycles looking for a write of 55H to TA. If the second write of 55H occurs within 3 clock cycles of the first write of AAH, then the timed access window is opened. It remains open for 4 clock cycles during which user may write to the protected bits. After 4 clock cycles, this window automatically closes. Once the window closes, the procedure should be repeated to write another protected bits. Not that the TA protected SFR are required timed access for writing but reading is not protected. User may read TA protected SFR without giving AAH and 55H to TA register. The suggestion code for opening the timed access window is shown below.

```
(CLR EA)              ;if any interrupt is enabled, disable temporally
MOV  TA,#0AAH
MOV  TA,#55H
(Instruction that writes a TA protected register)
(SETBEA)              ;resume interrupts enabled
```

Any enabled interrupt should be disabled during this procedure to avoid delay between these three writings. If there is no interrupt enabled, the CLR EA and SETB EA instructions can be left out.

Examples of timed assess are shown to illustrate correct or incorrect writing process.

Example 1,

```
MOV  TA,#0AAH         ;3 clock cycles
MOV  TA,#55H          ;3 clock cycles
ORL  WDCON,#data           ;4 clock cycles
```

Example 2,

```
MOV   TA,#0AAH        ;3 clock cycles
MOV   TA,#55H         ;3 clock cycles
NOP                   ;1 clock cycle
ANL   BODCON0,#data   ;4 clock cycles
```

Example 3,

```
MOV   TA,#0AAH        ;3 clock cycles
MOV   TA,#55H         ;3 clock cycles
MOV   WDCON,#data1    ;3 clock cycles
ORL   BODCON0,#data2  ;4 clock cycles
```

Example 4,

```
MOV   TA,#0AAH        ;3 clock cycles
NOP                   ;1 clock cycle
MOV   TA,#55H         ;3 clock cycles
ANL   BODCON0,#data   ;4 clock cycles
```

In the first example, the writing to the protected bits is done before the 3-clock-cycle window closes. In example 2, however, the writing to BODCON0 does not complete during the window opening, there will be no change of the value of BODCON0. In example 3, the WDCON is successful written but the BODCON0 write is out of the 3-clock-cycle window. Therefore, the BODCON0 value will not change either. In Example 4, the second write 55H to TA completes after 3 clock cycles of the first write TA of AAH, and thus the timed access window is not opened at all, and the write to the protected byte affects nothing.

26 INTERRUPT SYSTEM

26.1 Interrupt Overview

The purpose of the interrupt is to make the software deal with unscheduled or asynchronous events. The ML51 has a four-priority-level interrupt structure with 30 interrupt sources. Each of the interrupt sources has an individual priority setting bits, interrupt vector and enable bit. In addition, the interrupts can be globally enabled or disabled. When an interrupt occurs, the CPU is expected to service the interrupt. This service is specified as an Interrupt Service Routine (ISR). The ISR resides at a predetermined address as shown in Table 26.1-1 Interrupt Vectors. When the interrupt occurs if enabled, the CPU will vector to the respective location depending on interrupt source, execute the code at this location, stay in an interrupt service state until the ISR is done. Once an ISR has begun, it can be interrupted only by a higher priority interrupt. The ISR should be terminated by a return from interrupt instruction RETI. This instruction will force the CPU return to the instruction that would have been next when the interrupt occurred.

Table 26.1-1 Interrupt Vectors

Source	Vector Addess	Vector Number	Source	Vector Address	Vector Number
Reset	0000H	-	Serial port 1 interrupt	007BH	15
External interrupt 0	0003H	0	Timer 3 overflow	0083H	16
Timer 0 overflow	000BH	1	Self Wake-up Timer interrupt	008BH	17
External interrupt 1	0013H	2	CPU Hard Fault Interrupt	0093H	18
Timer 1 overflow	001BH	3	SMC0 Interrupt	009BH	19
Serial port 0 interrupt	0023H	4	PDMA0 Interrupt	00A3H	20
Timer 2 event	002BH	5	PDMA1 Interrupt	00ABH	21
I^2C0 status/timer-out interrupt	0033H	6	SPI1 Interrupt	00B3H	22
Pin interrupt	003BH	7	ACMP Interrupt	00BBH	23
Brown-out detection interrupt	0043H	8	I^2C1 status/timer-out interrupt	00C3H	24
SPI0 interrupt	004BH	9	PWM1 Interrupt	00CBH	25
WDT interrupt	0053H	10	Fault Brake1 interrupt	00D3H	26
ADC interrupt	005BH	11	SMC1 Interrupt	00DBH	27
Input capture interrupt	0063H	12	PDMA2 Interrupt	00E3H	28
PWM0 interrupt	006BH	13	PDMA3 Interrupt	00EBH	29
Fault Brake0 interrupt	0073H	14			

26.2 Enabling Interrupts

Each of individual interrupt sources can be enabled or disabled through the use of an associated interrupt enable bit in the IE and EIE0 SFR. There is also a global enable bit EA bit (IE.7), which can be cleared to disable all the interrupts at once. It is set to enable all individually enabled interrupts. Setting the EA bit to logic 0 disables all interrupt sources regardless of the individual interrupt-enable settings. Note that interrupts which occur when the EA bit is set to logic 0 will be held in a pending state, and will not be serviced until the EA bit is set back to logic 1. All interrupt flags that generate interrupts can also be set via software. Thereby software initiated interrupts can be generated.

Note that every interrupts, if enabled, is generated by a setting as logic 1 of its interrupt flag no matter by hardware or software. User should take care of each interrupt flag in its own interrupt service routine (ISR). Most of interrupt flags should be cleared by writing it as logic 0 via software to avoid recursive interrupt requests.

26.2.1 Control Register of Enabling Interrupts

IE – Interrupt Enable

Register	SFR Address	Reset Value
IE	A8H, All pages, Bit addressable	0000 _0000 b

7	6	5	4	3	2	1	0
EA	EADC	EBOD	ES	ET1	EX1	ET0	EX0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	EA	**Enable all interrupt** This bit globally enables/disables all interrupts that are individually enabled. 0 = All interrupt sources Disabled. 1 = Each interrupt Enabled depending on its individual mask setting. Individual interrupts will occur if enabled.
6	EADC	**Enable ADC interrupt** 0 = ADC interrupt Disabled. 1 = ADC interrupt Enable. When interrupt generated ADCF (ADCCON0.7) set 1.
5	EBOD	**Enable brown-out interrupt** 0 = Brown-out detection interrupt Disabled. 1 = Brown-out detection interrupt Enable. When interrupt generated BOF (BODCON0.3) set 1.
4	ES	**Enable serial port 0 interrupt** 0 = Serial port 0 interrupt Disabled. 1 = Serial port 0 interrupt Enable. When interrupt generated TI (SCON.1) or RI (SCON.0) set 1.
3	ET1	**Enable Timer 1 interrupt** 0 = Timer 1 interrupt Disabled. 1 = Timer 1 interrupt Enable. When interrupt generated TF1 (TCON.7) set 1.

Bit	Name	Description
2	EX1	**Enable external interrupt 1** 0 = External interrupt 1 Disabled. 1 = External interrupt 1 interrupt Enable. When interrupt generated $\overline{INT1}$ pin set 1.
1	ET0	**Enable Timer 0 interrupt** 0 = Timer 0 interrupt Disabled. 1 = Timer 0 interrupt Enable. When interrupt generated TF0 (TCON.5) set 1.
0	EX0	**Enable external interrupt 0** 0 = External interrupt 0 Disabled. 1 = External interrupt 0 interrupt Enable. When interrupt generated $\overline{INT0}$ pin set 1.

EIE0 – Extensive Interrupt Enable

Register	SFR Address	Reset Value
EIE0	9BH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
ET2	ESPI0	EFB0	EWDT	EPWM0	ECAP	EPI	EI^{2}C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	ET2	**Enable Timer 2 interrupt** 0 = Timer 2 interrupt Disabled. 1 = Timer 2 interrupt Enable. When interrupt generated, TF2 (T2CON.7) set 1
6	ESPI0	**Enable SPI interrupt** 0 = SPI interrupt Disabled. 1 = SPI interrupt Enable. When interrupt generated SPIF (SPInSR.7), SPIOVF (SPInSR.5), or MODF (SPInSR.4) set 1 .
5	EFB0	**Enable Fault Brake interrupt** 0 = Fault Brake interrupt Disabled. 1 = Fault Brake interrupt Enable. When interrupt generated FBF (PWM0FBD.7) set 1.
4	EWDT	**Enable WDT interrupt** 0 = WDT interrupt Disabled. 1 = WDT interrupt Enable. When interrupt generated WDTF (WDCON.5) set 1.
3	EPWM0	**Enable PWM0 interrupt** 0 = PWM interrupt Disabled. 1 = PWM interrupt Enable. When interrupt generated PWMF (PWMnCON0.5) set 1.
2	ECAP	**Enable input capture interrupt** 0 = Input capture interrupt Disabled. 1 = Input capture interrupt Enable. When interrupt generated CAPF[2:0] (CAPCON0[2:0]) set 1.

Bit	Name	Description
1	EPI	**Enable pin interrupt** 0 = Pin interrupt Disabled. 1 = Pin interrupt Enable. When interrupt generated PIF related bit set 1.
0	EI^{2}C0	**Enable I^{2}C0 interrupt** 0 = I^{2}C interrupt Disabled. 1 = I^{2}C interrupt Enable. When interrupt generated SI (I^{2}C0CON.3) or I2TOF (I^{2}C0TOC.0) set 1.

EIE1 – Extensive Interrupt Enable 1

Register	SFR Address	Reset Value
EIE0	9CH, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
EFB1	EPWM1	EI^{2}C1	ESPI1	EHFI	EWKT	ET3	ES1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	EFB1	**Enable Fault Brake 1 interrupt** 0 = Fault Brake interrupt Disabled. 1 = Fault Brake interrupt Enable. When interrupt generated FBF (PWM1FBD.7) Enabled.
6	EPWM1	**Enable PWM1 interrupt** 0 = PWM1 interrupt Disabled. 1 = PWM1 interrupt Enable. When interrupt generated PWMF (PWM1CON0.5) set 1.
5	EI^{2}C1	**Enable I^{2}C1 interrupt** 0 = I^{2}C1 interrupt Disabled. 1 = I^{2}C1 interrupt Enable. When interrupt generated SI (I^{2}C1CON.3) or I2TOF (I^{2}C1TOC.0) set 1.
4	ESPI1	**Enable SPI1 interrupt** 0 = SPI1 interrupt Disabled. 1 = SPI1 interrupt Enable. When interrupt generated SPIF (SP2SR.7), MODF (SP2SR.4) or SPIOVF (SP2SR.5) set 1
3	EHFI	**Enable hard fault interrupt** 0 = hard fault interrupt Disabled and hard fault reset is Enabled 1 = hard fault interrupt Enable. When interrupt generated HFIF (AUXR0.4) set 1.
2	EWKT	**Enable WKT interrupt** 0 = WKT interrupt Disabled. 1 = WKT interrupt Enable. When interrupt generated WKTF (WKCON.4) set 1.

Bit	Name	Description
1	ET3	**Enable Timer 3 interrupt** 0 = Timer 3 interrupt Disabled. 1 = Timer 3interrupt Enable. When interrupt generated TF3 (T3CON.4) set 1.
0	ES1	**Enable serial port 1 interrupt** 0 = Serial port 1 interrupt Disabled. 1 = Serial port 1 interrupt Enable. When interrupt generated TI_1 (S1CON.1) or RI_1 (S1CON.0) set 1.

SCnIE – SC Interrupt Enable Control Register

Register	SFR Address	Reset Value
SC0IE	DDH, Page 0	0000_0000 b
SC1IE	DCH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	-	-	ACERRIEN	BGTIEN	TERRIEN	TBEIEN	RDAIEN
-	-	-	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7:4	-	Reserved
4	ACERRIEN	Auto Convention Error Interrupt Enable Bit This field is used to enable auto-convention error interrupt. 0 = Auto-convention error interrupt Disabled. 1 = Auto-convention error interrupt Enabled.
3	BGTIEN	Block Guard Time Interrupt Enable Bit This field is used to enable block guard time interrupt. 0 = Block guard time interrupt Disabled. 1 = Block guard time interrupt Enabled.
2	TERRIEN	Transfer Error Interrupt Enable Bit This field is used to enable transfer error interrupt. The transfer error states is at SC0TSR register which includes receiver break error BEF(SC0TSR[6]), frame error FEF(SC0TSR[5]), parity error PEF(SC0TSR[4]), receiver buffer overflow error RXOV(SC0TSR[0]) and transmit buffer overflow error TXOV(SC0TSR[2]). 0 = Transfer error interrupt Disabled. 1 = Transfer error interrupt Enabled.
1	TBEIEN	Transmit Buffer Empty Interrupt Enable Bit This field is used to enable transmit buffer empty interrupt. 0 = Transmit buffer empty interrupt Disabled. 1 = Transmit buffer empty interrupt Enabled.

Bit	Name	Description
0	RDAIEN	Receive Data Reach Interrupt Enable Bit This field is used to enable received data interrupt. 0 = Receive data interrupt Disabled. 1 = Receive data interrupt Enabled.

DMAnCR – PDMAn Control Register

Register	SFR Address	Reset Value
DMA0CR	92H, Page 0	0000_0000 b
DMA1CR	EBH, Page 0	0000_0000 b
DMA2CR	B3H, Page 2	0000_0000 b
DMA3CR	ABH, Page 2	0000_0000 b

7	6	5	4	3	2	1	0
-	PSSEL[2:0]			HIE	FIE	RUN	EN
-	R/W			R/W	R/W	R/W	R/W

Bit	Name	Description
3	HIE	PDMA HALFTransfer Done Interrupt Enable Bit 0 = Interrupt Disabled when PDMA half transfer is done. 1 = Interrupt Enabled when PDMA half transfer is done.
2	FIE	PDMA Full Transfer Done Interrupt Enable Bit 0 = Interrupt Disabled when PDMA full transfer is done. 1 = Interrupt Enabled when PDMA full transfer is done.

26.3 Interrupt Priorities

There are four priority levels for all interrupts. They are level highest, high, low, and lowest; and they are represented by level 3, level 2, level 1, and level 0. The interrupt sources can be individually set to one of four priority levels by setting their own priority bits. Table 26.3-1 lists four priority setting. Naturally, a low level priority interrupt can itself be interrupted by a high level priority interrupt, but not by any same level interrupt or lower level. In addition, there exists a pre-defined natural priority among the interrupts themselves. The natural priority comes into play when the interrupt controller has to resolve simultaneous requests having the same priority level.

In case of multiple interrupts, the following rules apply:

1. While a low priority interrupt handler is running, if a high priority interrupt arrives, the handler will be interrupted and the high priority handler will run. When the high priority handler does “RETI”, the low priority handler will resume. When this handler does “RETI”, control is passed back to the main program.

2. If a high priority interrupt is running, it cannot be interrupted by any other source – even if it is a high priority interrupt which is higher in natural priority.

3. A low-priority interrupt handler will be invoked only if no other interrupt is already executing. Again, the low priority interrupt cannot preempt another low priority interrupt, even if the later one is higher in natural priority.

4. If two interrupts occur at the same time, the interrupt with higher priority will execute first. If both interrupts are of the same priority, the interrupt which is higher in natural priority will be executed first. This is the only context in which the natural priority matters.

This natural priority is defined as shown on Table 26.3-2 It also summarizes the interrupt sources, flag bits, vector addresses, enable bits, priority bits, natural priority and the permission to wake up the CPU from Power-down mode. For details of waking CPU up from Power-down mode, please see Section 28.4 “Power-Down Mode”.

Table 26.3-1 Interrupt Priority Level Setting

Interrupt Priority Control Bits		Interrupt Priority Level
IPH/EIPH0/EIPH1/EIPH2	IP/EIP0/EIP1/EIP2	
0	0	Level 0 (lowest)
0	1	Level 1
1	0	Level 2
1	1	Level 3 (highest)

Table 26.3-2 Characteristics of Each Interrupt Source

Interrupt Source	Vector Address	Interrupt Flag(S)	Enable Bit	Natural Priority	Priority Control Bits	Power-Down Wake-Up
Reset	0000H	-	Always Enabled	Highest	-	Yes
CPU Hard Fault	0093H	HFIF (RSR.5)	EHFI	1	PHF, PHFH	No

Interrupt Source	Vector Address	Interrupt Flag(S)	Enable Bit	Natural Priority	Priority Control Bits	Power-Down Wake-Up
External interrupt 0	0003H	IE0[1] (TCON.1)	EX0	2	PX0, PX0H	Yes
Brown-out	0043H	BOF (BODCON0.3)	EBOD	3	PBOD, PBODH	Yes
Watchdog Timer	0053H	WDTF (WDCON.5)	EWDT	4	PWDT, PWDTH	Yes
Timer 0	000BH	TF0[2] (TCON.5)	ET0	5	PT0, PT0H	No
I^2C0 status/time-out	0033h	SI + I2TOF (I2C0TOC.0)	EI^2C0	6	PI2C0, PI2C0H	No
ADC	005Bh	ADCF (ADCCON0.7)	EADC	7	PADC, PADCH	No
External interrupt 1	0013H	IE1[1] (TCON.3)	EX1	8	PX1, PX1H	Yes
Pin interrupt	003BH	PIF0 to PIF7 (PIF)[3]	EPI	9	PPI, PPIH	Yes
Timer 1	001BH	TF1[2] (TCON.7)	ET1	10	PT1, PT1H	No
Serial port 0	0023H	RI + TI (SCON[1:0])	ES	11	PS, PSH	No
PWM0 Fault Brake event	0073h	FBF (PWM0FBD.7)	EFB0	12	PFB, PFBH	No
SPI0	004Bh	SPIF (SPI0SR.7) + MODF (SPI0SR.4) + SPIOVF (SPI0SR.5)	ESPI0	13	PSPI, PSPIH	No
Timer 2	002BH	TF2[2] (T2CON.7)	ET2	14	PT2, PT2H	No
Input capture	0063H	CAPF[2:0] (CAPCON0[2:0])	ECAP	15	PCAP, PCAPH	No
PWM0 interrupt	006BH	PWMF(PWM0CON0.5)	EPWM0	16	PPWM0, PPWM0H	No
Serial port 1	007BH	RI_1 + TI_1(S1CON[1:0])	ES1	17	PS1, PS1H	No
Timer 3	0083H	TF3[2] (T3CON.4)	ET3	18	PT3, PT3H	No
Self Wake-up Timer	008BH	WKTF (WKCON.4)	EWKT	19	PWKT, PWKTH	Yes
SC0	009BH	ACERR+BGT+TERR+ TBE+RDA (SC0IS[4:0])	SC0IE	20	SMC0, SMC0H	No
PDMA0	00A3H	HDONE+FDONE (DMA0TSR[1:0])	DMA0CR (HIE, FIE)	21	PDMA0, PDMA0H	No
PDMA1	00ABH	HDONE+FDONE (DMA1TSR[1:0])	DMA1CR (HIE, FIE)	22	PDMA1, PDMA1H	No
SPI1	00B3h	SPIF (SPI1SR.7) + MODF (SPI1SR.4) + SPIOVF (SPI1SR.5)	ESPI1	23	PSPI, PSPIH	No

Interrupt Source	Vector Address	Interrupt Flag(S)	Enable Bit	Natural Priority	Priority Control Bits	Power-Down Wake-Up
ACMP	00BBh	ACMP1IF (ACMPSR.0) + ACMP2IF (ACMPSR.2) +	CMPIE (ACMPCR1.1 ACMPCR2.1)	24	PACMP, PACMPH	Yes
I^2C1 status/time-out	00C3h	SI + I2TOF (I^2C1TOC.0)	EI^{2}C1	25	PI^{2}C1, PI^{2}C1H	No
PWM1 interrupt	00CBH	PWMF(PWM1CON0.5)	EPWM1	26	PPWM1, PPWM1H	No
SC1	00D3H	ACERR+BGT+TERR+ TBE+RDA (SC1IS[4:0])	SC1IE	27	SMC1, SMC1H	No
PWM1 Fault Brake event	00DBH	FBF (PWM1FBD.7)	EFB1	28	PFB, PFBH	No
PDMA2	00E3H	HDONE+FDONE DMA2TSR[1:0])	DMA2CR (HIE, FIE)	29	PDMA2, PDMA2H	No
PDMA3	00EBH	HDONE+FDONE DMA3TSR[1:0])	DMA3CR (HIE, FIE)	30	PDMA3, PDMA3H	No

Note:
[1] While the external interrupt pin is set as edge triggered (Itx = 1), its own flag Iex will be automatically cleared if the interrupt service routine (ISR) is executed. While as level triggered (Itx = 0), Iex follows the inverse of respective pin state. It is not controlled via software.
[2] TF0, TF1, or TF3 is automatically cleared if the interrupt service routine (ISR) is executed. On the contrary, be aware that TF2 is not.
[3] If level triggered is selected for pin interrupt channel n, PIFn flag reflects the respective channel state. It is not controlled via software.

26.3.1 Control Register of Interrupt Priorities

IP – Interrupt Priority

Register	SFR Address	Reset Value
IP	B8H, All pages, Bit-addressable	0000_0000 b

Note: IP is used in combination with the IPH to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
-	PADC	PBOD	PS	PT1	PX1	PT0	PX0
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	PADC	ADC interrupt priority low bit
5	PBOD	Brown-out detection interrupt priority low bit
4	PS	Serial port 0 interrupt priority low bit

Bit	Name	Description
3	PT1	Timer 1 interrupt priority low bit
2	PX1	External interrupt 1 priority low bit
1	PT0	Timer 0 interrupt priority low bit
0	PX0	External interrupt 0 priority low bit

IPH – Interrupt Priority High

Register	SFR Address	Reset Value
IPH	B7H, Page 0	0000_0000 b

7	6	5	4	3	2	1	0
-	PADCH	PBODH	PSH	PT1H	PX1H	PT0H	PX0H
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	PADC	ADC interrupt priority high bit
5	PBOD	Brown-out detection interrupt priority high bit
4	PSH	Serial port 0 interrupt priority high bit
3	PT1H	Timer 1 interrupt priority high bit
2	PX1H	External interrupt 1 priority high bit
1	PT0H	Timer 0 interrupt priority high bit
0	PX0H	External interrupt 0 priority high bit

EIP0 – Extensive Interrupt Priority

Register	SFR Address	Reset Value
EIP0	EFH, Page 0	0000_0000 b

Note: EIP0 is used in combination with the EIPH0 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
PT2	PSPI0	PFB	PWDT	PPWM0	PCAP	PPI	PI^{2}C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PT2	Timer 2 interrupt priority low bit
6	PSPI0	SPI0 interrupt priority low bit
5	PFB	Fault Brake interrupt priority low bit
4	PWDT	WDT interrupt priority low bit
3	PPWM0	PWM interrupt priority low bit
2	PCAP	Input capture interrupt priority low bit
1	PPI	Pin interrupt priority low bit
0	PI^{2}C0	I^{2}C interrupt priority low bit

EIPH0 – Extensive Interrupt Priority High

Register	SFR Address	Reset Value
EIPH0	F7H, All pages	0000_0000 b

Note: EIPH0 is used in combination with the EIP0 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
PT2H	PSPI0H	PFBH	PWDTH	PPWM0H	PCAPH	PPIH	PI^{2}C0H
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PT2H	Timer 2 interrupt priority high bit
6	PSPI0H	SPI0 interrupt priority high bit
5	PFBH	Fault Brake interrupt priority high bit
4	PWDTH	WDT interrupt priority high bit
3	PPWM0H	PWM0 interrupt priority high bit
2	PCAPH	Input capture interrupt priority high bit
1	PPIH	Pin interrupt priority high bit
0	PI^{2}C0H	I^{2}C interrupt priority high bit

EIP1 – Extensive Interrupt Priority 1

Register	SFR Address	Reset Value
EIP1	FEH, Page 0	0000_0000 b

Note: EIP1 is used in combination with the EIPH1 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
PSPI1	PDMA1	PDMA0	PSMC	PHF	PWKT	PT3	PS1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PSPI1	SPI1 interrupt priority low bit
6	PDMA1	PDMA1 interrupt priority low bit
5	PDMA0	PDMA0 interrupt priority low bit
4	PSMC	SMC interrupt priority low bit
3	PHF	Hard fault interrupt priority low bit
2	PWKT	WKT interrupt priority low bit
1	PT3	Timer 3 interrupt priority low bit
0	PS1	Serial port 1 interrupt priority low bit

EIPH1 – Extensive Interrupt Priority High 1

Register	SFR Address	Reset Value
EIPH1	FFH, Page 0	0000_0000 b

Note: EIPH1 is used in combination with the EIP1 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
PSPI1H	PDMA1H	PDMA0H	PSMCH	PHFH	PWKTH	PT3H	PS1H
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	PSPI1H	SPI1 interrupt priority high bit
6	PDMA1H	PDMA1 interrupt priority high bit
5	PDMA0H	PDMA0 interrupt priority high bit
4	PSMCH	SMC interrupt priority high bit

Bit	Name	Description
3	PHFH	Hard fault interrupt priority high bit
2	PWKTH	WKT interrupt priority high bit
1	PT3H	Timer 3 interrupt priority high bit
0	PS1H	Serial port 1 interrupt priority high bit

EIP2 – Extensive Interrupt Priority 2

Register	SFR Address	Reset Value
EIP2	ACH, Page 0	0000_0000 b

Note: EIP2 is used in combination with the EIPH2 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
-	PDMA3	PDMA2	SMC1	PFB1	PPWM1	PI^{2}C1	PACMP
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	PDMA3	PDMA3 interrupt priority low bit
5	PDMA2	PDMA2 interrupt priority low bit
4	SMC1	SMC1 interrupt priority low bit
3	PFB1	Fault Brake1 interrupt priority low bit
2	PPWM1	PPWM1 interrupt priority low bit
1	PI^{2}C1	I^{2}C interrupt priority low bit
0	PACMP	ACMP interrupt priority low bit

EIPH2 – Extensive Interrupt Priority High 2

Register	SFR Address	Reset Value
EIPH2	ADH, Page 0	0000_0000 b

Note: EIPH2 is used in combination with the EIP2 to determine the priority of each interrupt source. See Table 26.3-1 Interrupt Priority Level Setting for correct interrupt priority configuration.

7	6	5	4	3	2	1	0
-	PDMA3H	PDMA2H	SMC1H	PFB1H	PPWM1H	PI^{2}C1H	PACMPH
-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
7	-	Reserved
6	PDMA3H	PDMA3H interrupt priority high bit
5	PDMA2H	PDMA2H interrupt priority high bit
4	SMC1H	SMC1H interrupt priority high bit
3	PFB1H	Fault Brake1 interrupt priority high bit
2	PPWM1H	PPWM1H interrupt priority high bit
1	PI^{2}C1H	I^{2}C interrupt priority high bit
0	PACMPH	ACMP interrupt priority high bit

AUXR0 – Auxiliary Register 0

Register	SFR Address	Reset Value
AUXR0	A2H, Page:0	POR: 0000_0000b Software: 1UU0_0000b Reset pin: U1U0_0000b Hard fault: UU10_0000b Others: UUU0_0000b

7	6	5	4	3	2	1	0
SWRF	RSTPINF	HFRF	HFIF	GF2	-	0	DPS
R/W	R/W	R/W	R/W	R/W	-	R	R/W

Bit	Name	Description
7	SWRF	Software reset flag When the MCU is reset via software reset, this bit will be set via hardware. It is recommended that the flag be cleared via software.
6	RSTPINF	External reset flag When the MCU is reset by the external reset, this bit will be set via hardware. It is recommended that the flag be cleared via software.
5	HFRF	Hard Fault reset flag Once CPU fetches instruction address over Flash size while EHFI (EIE1.4)=0, MCU will reset and this bit will be set via hardware. It is recommended that the flag be cleared via software. **Note:** If MCU run in OCD debug mode and OCDEN = 0, Hard fault reset will disable. Only HFIF flag be asserted.
4	HFIF	Hard Fault Interrupt flag Once CPU fetches instruction address over Flash size while EHFI (EIE1.4)=1. MCU will be interrupt and this bit will be set via hardware. It is recommended that the flag be cleared via software.

26.4 Interrupt Service

The interrupt flags are sampled every system clock cycle. In the same cycle, the sampled interrupts are polled and their priority is resolved. If certain conditions are met then the hardware will execute an internally generated LCALL instruction, which will vector the process to the appropriate interrupt vector address. The conditions for generating the LCALL are,

1. An interrupt of equal or higher priority is not currently being serviced.

2. The current polling cycle is the last cycle of the instruction currently being executed.

3. The current instruction does not involve a write to any enabling or priority setting bits and is not a RETI.

If any of these conditions are not met, then the LCALL will not be generated. The polling cycle is repeated every system clock cycle. If an interrupt flag is active in one cycle but not responded to for the above conditions are not met, if the flag is not still active when the blocking condition is removed, the denied interrupt will not be serviced. This means that the interrupt flag, which was once active but not serviced is not remembered. Every polling cycle is new.

The processor responds to a valid interrupt by executing an LCALL instruction to the appropriate service routine. This action may or may not clear the flag, which caused the interrupt according to different interrupt source. The hardware LCALL behaves exactly like the software LCALL instruction. This instruction saves the Program Counter contents onto the Stack RAM but does not save the Program Status Word (PSW). The PC is reloaded with the vector address of that interrupt, which caused the LCALL. Execution continues from the vectored address until an RETI instruction is executed. On execution of the RETI instruction, the processor pops the Stack and loads the PC with the contents at the top of the stack. User should take care that the status of the stack. The processor does not notice anything if the stack contents are modified and will proceed with execution from the address put back into PC. Note that a simple RET instruction would perform exactly the same process as a RETI instruction, but it would not inform the Interrupt controller that the interrupt service routine is completed. RET would leave the controller still thinking that the service routine is underway, making future interrupts impossible.

26.5 Interrupt Latency

The response time for each interrupt source depends on several factors, such as the nature of the interrupt and the instruction underway. Each interrupt flags are polled and priority decoded each system clock cycle. If a request is active and all three previous conditions are met, then the hardware generated LCALL is executed. This LCALL itself takes 4 clock cycles to be completed. Thus, there is a minimum reaction time of 5 clock cycles between the interrupt flag being set and the interrupt service routine being executed.

A longer response time should be anticipated if any of the three conditions are not met. If a higher or equal priority is being serviced, then the interrupt latency time obviously depends on the nature of the service routine currently being executed. If the polling cycle is not the last clock cycle of the instruction being executed, then an additional delay is introduced. The maximum response time (if no other interrupt is currently being serviced or the new interrupt is of greater priority) occurs if the device is performing a RETI, and then executes a longest 6-clock-cycle instruction as the next instruction. From the time an interrupt source is activated (not detected), the longest reaction time is 16 clock cycles. This period includes 5 clock cycles to complete RETI, 6 clock cycles to complete the longest instruction, 1 clock cycle to detect the interrupt, and 4 clock cycles to complete the hardware LCALL to the interrupt vector location.

Thus in a single-interrupt system the interrupt response time will always be more than 5 clock cycles and not more than 16 clock cycles.

26.6 External Interrupt Pins

The external interrupt $\overline{INT0}$ and $\overline{INT1}$ can be used as interrupt sources. They are selectable to be either edge or level triggered depending on bits IT0 (TCON.0) and IT1 (TCON.2). The bits IE0 (TCON.1) and IE1 (TCON.3) are the flags those are checked to generate the interrupt. In the edge triggered mode, the $\overline{INT0}$ or $\overline{INT1}$ inputs are sampled every system clock cycle. If the sample is high in one cycle and low in the next, then a high to low transition is detected and the interrupts request flag IE0 or IE1 will be set. Since the external interrupts are sampled every system clock, they have to be held high or low for at least one system clock cycle. The IE0 and IE1 are automatically cleared when the interrupt service routine is called. If the level triggered mode is selected, then the requesting source has to hold the pin low till the interrupt is serviced. The IE0 and IE1 will not be cleared by the hardware on entering the service routine. In the level triggered mode, IE0 and IE1 follows the inverse value of $\overline{INT0}$ and $\overline{INT1}$ pins. If interrupt pins continue to be held low even after the service routine is completed, the processor will acknowledge another interrupt request from the same source. Both $\overline{INT0}$ and $\overline{INT1}$ can wake up the device from the Power-down mode.

TCON – Timer 0 and 1 Control

Register	SFR Address	Reset Value
TCON	88H, All pages, Bit-addressable	0000_0000b

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R (level) R/W (edge)	R/W	R (level) R/W (edge)	R/W

Bit	Name	Description
3	IE1	External interrupt 1 edge flag If IT1 = 1 (falling edge trigger), this flag will be set by hardware when a falling edge is detected. It remain set until cleared via software or cleared by hardware in the beginning of its interrupt service routine. If IT1 = 0 (low level trigger), this flag follows the inverse of the $\overline{INT1}$ input signal's logic level. Software cannot control it.
2	IT1	External interrupt 1 type select This bit selects by which type that $\overline{INT1}$ is triggered. 0 = $\overline{INT1}$ is low level triggered. 1 = $\overline{INT1}$ is falling edge triggered.
1	IE0	External interrupt 0 edge flag If IT0 = 1 (falling edge trigger), this flag will be set by hardware when a falling edge is detected. It remain set until cleared via software or cleared by hardware in the beginning of its interrupt service routine. If IT0 = 0 (low level trigger), this flag follows the inverse of the $\overline{INT0}$ input signal's logic level. Software cannot control it.
0	IT0	External interrupt 0 type select This bit selects by which type that $\overline{INT0}$ is triggered. 0 = $\overline{INT0}$ is low level triggered. 1 = $\overline{INT0}$ is falling edge triggered.

27 IN-APPLICATION-PROGRAMMING (IAP)

Unlike RAM's real-time operation, to update Flash data often takes long time. Furthermore, it is a quite complex timing procedure to erase, program, or read Flash data. The ML51 carried out the Flash operation with convenient mechanism to help user re-programming the Flash content by In-Application-Programming (IAP). IAP is an in-circuit electrical erasure and programming method through software.

After IAP enabling by setting IAPEN (CHPCON.0 with TA protected) and setting the enable bit in IAPUEN that allows the target block to be updated, user can easily fill the 16-bit target address in IAPAH and IAPAL, data in IAPFD, and command in IAPCN. Then the IAP is ready to begin by setting a triggering bit IAPGO (IAPTRG.0). Note that IAPTRG is also TA protected. At this moment, the CPU holds the Program Counter and the built-in IAP automation takes over to control the internal charge-pump for high voltage and the detail signal timing. The erase and program time is internally controlled disregard of the operating voltage and frequency. Nominally, a page-erase time is 5 ms and a byte-program time is 23.5 μs. After IAP action completed, the Program Counter continues to run the following instructions. The IAPGO bit will be automatically cleared. An IAP failure flag, IAPFF (CHPCON.6), can be check whether the previous IAP operation was successful or not. Through this progress, user can easily erase, program, and verify the Flash Memory by just taking care of pure software.

The following registers are related to IAP processing.

CONFIG2

7	6	5	4	3	2	1	0
CBODEN	CBOV[2:0]			BOIAP	CBORST	-	-
R/W	R/W			R/W	R/W	-	-

Factory default value: 1111 1111b

Bit	Name	Description
3	BOIAP	Brown-out inhibiting IAP This bit decide whether IAP erasing or programming is inhibited by brown-out status. This bit is valid only when brown-out detection is enabled. 1 = IAP erasing or programming is inhibited if V_{DD} is lower than V_{BOD}. 0 = IAP erasing or programming is allowed under any workable V_{DD}.

27.1 Commands of IAP

The ML51 provides a wide range of applications to perform IAP to APROM, LDROM, or CONFIG bytes. The IAP action mode and the destination of the Flash block are defined by IAP control register IAPCN.

Table 27.1-1 IAP Modes and Command Codes

IAP Mode	IAPCN				IAPA[15:0] {IAPAH, IAPAL}	IAPFD[7:0]
	IAPB [1:0]	FOEN	FCEN	FCTRL [3:0]		
Company ID read	XX[1]	0	0	1011	X	DAH
Device ID read	XX	0	0	1100	Low-byte DID: 0000H High-byte DID: 0001H	Low-byte DID High-byte DID
96-bit Unique Code read	XX	0	0	0100	0000H to 000BH	Data out
APROM page-erase	00	1	0	0010	Address in[2]	FFH
LDROM page-erase	01	1	0	0010	Address in[2]	FFH
APROM byte-program	00	1	0	0001	Address in	Data in
LDROM byte-program	01	1	0	0001	Address in	Data in
APROM byte-read	00	0	0	0000	Address in	Data out
LDROM byte-read	01	0	0	0000	Address in	Data out
All CONFIG bytes erase	11	1	0	0010	0000H	FFH
CONFIG byte-program	11	1	0	0001	CONFIG0: 0000H CONFIG1: 0001H CONFIG2: 0002H CONFIG4: 0004H CONFIG6: 0005H	Data in
CONFIG byte-read	11	0	0	0000	CONFIG0: 0000H CONFIG1: 0001H CONFIG2: 0002H CONFIG4: 0004H CONFIG6: 0005H	Data out

Note:
[1] "X" means "don't care".
[2] Each page is 128 bytes size. Therefore, the address should be the address pointed to the target page.

27.2 Control register of IAP

CHPCON – Chip Control (TA protected)

7	6	5	4	3	2	1	0
SWRST	IAPFF	-	-	-	-	BS	IAPEN
W	R/W	-	-	-	-	R/W	R/W

Address: 9FH, All pages Reset value: Software: 0000 00U0b / others: 0000 00C0b

Bit	Name	Description
6	IAPFF	IAP fault flag The hardware will set this bit after IAPGO (IAPTRG.0) is set if any of the following condition is met: (1) The accessing address is oversize. (2) IAPCN command is invalid. (3) IAP erases or programs updating un-enabled block. (4) IAP erasing or programming operates under VBOD while BOIAP (CONFIG2.5) remains un-programmed 1 with BODEN (BODCON0.7) as 1 and BORST (BODCON0.2) as 0. This bit should be cleared via software.
0	IAPEN	IAP enable 0 = IAP function Disabled. 1 = IAP function Enabled. Once enabling IAP function, the HIRC will be turned on for timing control. To clear IAPEN should always be the last instruction after IAP operation to stop internal oscillator if reducing power consumption is concerned.

IAPUEN – IAP Updating Enable (TA protected)

7	6	5	4	3	2	1	0
-	-	-	SPMEN	SPUEN	CFUEN	LDUEN	APUEN
-	-	-	R/W	R/W	R/W	R/W	R/W

Address: A5H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:5	-	Reserved
4	SPMEN	SPROM Memory space mapping enable 0 = CPU memory address 0xff80~0xffff is mapping to APROM memory 1 = CPU memory address 0xff80~0xffff is mapping to SPROM memory
3	SPUEN	SPROM Memory space updated enable(TA protected) 0 = Inhibit erasing or programming SPROM bytes by IAP 1 = Allow erasing or programming SPROM bytes by IAP.
2	CFUEN	CONFIG bytes updated enable 0 = Inhibit erasing or programming CONFIG bytes by IAP. 1 = Allow erasing or programming CONFIG bytes by IAP.

Bit	Name	Description
1	LDUEN	LDROM updated enable 0 = Inhibit erasing or programming LDROM by IAP. 1 = Allow erasing or programming LDROM by IAP.
0	APUEN	APROM updated enable 0 = Inhibit erasing or programming APROM by IAP. 1 = Allow erasing or programming APROM by IAP.

IAPCN – IAP Control

7	6	5	4	3	2	1	0
IAPB[1:0]		FOEN	FCEN	FCTRL[3:0]			
R/W		R/W	R/W	R/W			

Address: AFH, Page 0 Reset value: 0011 0000b

Bit	Name	Description
7:6	IAPB[1:0]	IAP control This byte is used for IAP command. For details, see Table 27.1-1 IAP Modes and Command Codes.
5	FOEN	
4	FCEN	
3:0	FCTRL[3:0]	

IAPAH – IAP Address High Byte

7	6	5	4	3	2	1	0
IAPA[15:8]							
R/W							

Address: A7H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:0	IAPA[15:8]	IAP address high byte IAPAH contains address IAPA[15:8] for IAP operations.

IAPAL – IAP Address Low Byte

7	6	5	4	3	2	1	0
IAPA[7:0]							
R/W							

Address: A6H, Page 0 Reset value: 0000 0000b

Bit	Name	Description

Bit	Name	Description
7:0	IAPA[7:0]	IAP address low byte IAPAL contains address IAPA[7:0] for IAP operations.

IAPFD – IAP Flash Data

7	6	5	4	3	2	1	0
IAPFD[7:0]							
R/W							

Address: AEH, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:0	IAPFD[7:0]	IAP Flash data This byte contains Flash data, which is read from or is going to be written to the Flash Memory. User should write data into IAPFD for program mode before triggering IAP processing and read data from IAPFD for read/verify mode after IAP processing is finished.

IAPTRG – IAP Trigger (TA protected)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	IAPGO
-	-	-	-	-	-	-	W

Address: A4H, Page 0 Reset value: 0000 0000b

Bit	Name	Description
7:1	-	Reserved

0	IAPGO	IAP go
		IAP begins by setting this bit as logic 1. After this instruction, the CPU holds the Program Counter (PC) and the IAP hardware automation takes over to control the progress. After IAP action completed, the Program Counter continues to run the following instruction. The IAPGO bit will be automatically cleared and always read as logic 0.
		Before triggering an IAP action, interrupts (if enabled) should be temporary disabled for hardware limitation.
		The program process should follows below.
		CLR EA
		MOV TA,#0AAH
		MOV TA,#55H
		ORL IAPTRG,#01H
		(SETB EA)

27.3 IAP User Guide

IAP facilitates the updating Flash contents in a convenient way; however, user should follow some restricted laws in order that the IAP operates correctly. Without noticing warnings will possible cause undetermined results even serious damages of devices. Furthermore, this paragraph will also support useful suggestions during IAP procedures.

(1) If no more IAP operation is needed, user should clear IAPEN (CHPCON.0). It will make the system void to trigger IAP unaware. Furthermore, IAP requires the HIRC running. If the external clock source is selected, disabling IAP will stop the HIRC for saving power consumption. Note that a write to IAPEN is TA protected.

(2) When the LOCK bit (CONFIG0.1) is activated, IAP reading, writing, or erasing can still be valid.

During IAP progress, interrupts (if enabled) should be disabled temporally by clearing EA bit for implement limitation.

Do not attempt to erase or program to a page that the code is currently executing. This will cause unpredictable program behavior and may corrupt program data.

27.4 Using Flash Memory as Data Storage

In general application, there is a need of data storage, which is non-volatile so that it remains its content even after the power is off. Therefore, in general application user can read back or update the data, which rules as parameters or constants for system control. The Flash Memory array of the ML51 supports IAP function and any byte in the Flash Memory array may be read using the MOVC instruction and thus is suitable for use as non-volatile data storage. IAP provides erase and program function that makes it easy for one or more bytes within a page to be erased and programmed in a routine. IAP performs in the application under the control of the microcontroller's firmware. Be aware of Flash Memory writing endurance of 100,000 cycles. A demo is illustrated as follows.

Assembly demo code:

```
;*****************************************************************************
    ;This code illustrates how to use IAP to make APROM 201h as a byte of
    ;Data Flash when user code is executed in APROM.
;*****************************************************************************
    PAGE_ERASE_AP             EQU       00100010b
    BYTE_PROGRAM_AP      EQU       00100001b

     ORG   0000h

     MOV   TA,#0Aah            ;CHPCON is TA protected
     MOV   TA,#55h
     ORL   CHPCON,#00000001b        ;IAPEN = 1, enable IAP mode

     MOV   TA,#0Aah            ;IAPUEN is TA protected
     MOV   TA,#55h
     ORL   IAPUEN,#00000001b        ;APUEN = 1, enable APROM update

     MOV   IAPCN,#PAGE_ERASE_AP;Erase page 200h~27Fh
     MOV   IAPAH,#02h
     MOV   IAPAL,#00h
     MOV   IAPFD,#0FFh
```

```
     MOV  TA,#0Aah            ;IAPTRG is TA protected
     MOV  TA,#55h
     ORL  IAPTRG,#00000001b        ;write '1' to IAPGO to trigger IAP
process
     MOV  IAPCN,#BYTE_PROGRAM_AP   ;Program 201h with 55h
     MOV  IAPAH,#02h
     MOV  IAPAL,#01h
     MOV  IAPFD,#55h
     MOV  TA,#0Aah
     MOV  TA,#55h
     ORL  IAPTRG,#00000001b

     MOV  TA,#0Aah
     MOV  TA,#55h
     ANL  IAPUEN,#11111110b        ;APUEN = 0, disable APROM update

     MOV  TA,#0Aah
     MOV  TA,#55h
     ANL  CHPCON,#11111110b        ;IAPEN = 0, disable IAP mode

     MOV  DPTR,#201h
     CLR  A
     MOVC A,@A+DPTR           ;Read content of address 201h
     MOV  P0,A

     SJMP $
```

C language demo code:

```
//***********************************************************************
    //    This code illustrates how to use IAP to make APROM 201h as a byte
of
    //    Data Flash when user code is executed in APROM.
//***********************************************************************
    #define    PAGE_ERASE_AP       0x22
    #define    BYTE_PROGRAM_AP     0x21

  /*Data Flash, as part of APROM, is read by MOVC. Data Flash can be
defined as128-element array in "code" area from absolute address 0x0200
*/

    volatile unsigned char code Data_Flash[128] _at_ 0x0200;

    Main (void)
    {
     TA = 0Xaa;               //CHPCON is TA protected
     TA = 0x55;
     CHPCON |= 0x01;               //IAPEN = 1, enable IAP mode
```

```
    TA = 0Xaa;                //IAPUEN is TA protected
    TA = 0x55;
    IAPUEN |= 0x01;                 //APUEN = 1, enable APROM update

    IAPCN = PAGE_ERASE_AP;          //Erase page 200h~27Fh
    IAPAH = 0x02;
    IAPAL = 0x00;
    IAPFD = 0Xff;
    TA = 0Xaa;                //IAPTRG is TA protected
    TA = 0x55;
    IAPTRG |= 0x01;           //write '1' to IAPGO to trigger IAP process

    IAPCN = BYTE_PROGRAM_AP; // Program 201h with 55h
    IAPAH = 0x02;
    IAPAL = 0x01;
    IAPFD = 0x55;
    TA = 0Xaa;
    TA = 0x55;
    IAPTRG |= 0x01;           //write '1' to IAPGO to trigger IAP process

    TA = 0Xaa;                //IAPUEN is TA protected
    TA = 0x55;
    IAPUEN &= ~0x01;                //APUEN = 0, disable APROM update

    TA = 0Xaa;                //CHPCON is TA protected
    TA = 0x55;
    CHPCON &= ~0x01;                //IAPEN = 0, disable IAP mode

    P0 = Data_Flash[1];       //Read content of address 200h+1

    while(1);
   }
```

27.5 In-System-Programming (ISP)

The Flash Memory supports both hardware programming and In-Application-Programming (IAP). If the product is just under development or the end product needs firmware updating in the hand of an end user, the hardware programming mode will make repeated programming difficult and inconvenient. In-System-Programming (ISP) makes it easy and possible. ISP performs Flash Memory updating without removing the microcontroller from the system. It allows a device to be re-programmed under software control. Furthermore, the capability to update the application firmware makes wide range of applications possible.

User can develop a custom Boot Code that resides in LDROM. The maximum size of LDROM is 4K Byte. User developed Boot Code can be re-programmed by parallel writer or In-Circuit-Programming (ICP) tool.

General speaking, an ISP is carried out by a communication between PC and MCU. PC transfers the new User Code to MCU through serial port. Then Boot Code receives it and re-programs into User Code through IAP commands. Nuvoton provides ISP firmware and PC application for ML51. It makes

user quite easy perform ISP through UART port. Please visit Nuvoton 8-bit Microcontroller website: Nuvoton 80C51 Microcontroller Technical Support. A simple ISP demo code is given below.

Assembly demo code:

```
;*****************************************************************************
    ;This code illustrates how to do APROM and CONFIG IAP from LDROM.
    ;APROM are re-programmed by the code to output P1 as 55h and P2 as aah.
    ;The CONFIG2 is also updated to disable BOD reset.
    ;User needs to configure CONFIG0 = 0x7F, CONFIG1 = 0Xfe, CONFIG2 =
0xFF.
;*****************************************************************************
    PAGE_ERASE_AP             EQU       00100010b
    BYTE_PROGRAM_AP      EQU       00100001b
    BYTE_READ_AP              EQU       00000000b
    ALL_ERASE_CONFIG     EQU       11100010b
    BYTE_PROGRAM_CONFIG       EQU       11100001b
    BYTE_READ_CONFIG     EQU       11000000b

     ORG  0000h

     CLR  EA                   ;disable all interrupts
     CALL Enable_IAP

     CALL Enable_AP_Update
     CALL Erase_AP             ;erase AP data
     CALL Program_AP           ;programming AP data
     CALL Disable_AP_Update
     CALL Program_AP_Verify         ;verify Programmed AP data

     CALL Read_CONFIG               ;read back CONFIG2
     CALL Enable_CONFIG_Update
     CALL Erase_CONFIG         ;erase CONFIG bytes
     CALL Program_CONFIG       ;programming CONFIG2 with new data
     CALL Disable_CONFIG_Update
     CALL Program_CONFIG_Verify     ;verify Programmed CONFIG2

     CALL Disable_IAP
     MOV  TA,#0Aah             ;TA protection
     MOV  TA,#55h              ;
     ANL  CHPCON,#11111101b         ;BS = 0, reset to APROM
     MOV  TA,#0Aah
     MOV  TA,#55h
     ORL  CHPCON,#80h               ;software reset and reboot from APROM

     SJMP $

    ;*****************************************************************
    ;     IAP Subroutine
```

```
    ;*****************************************************************
    Enable_IAP:
     MOV  TA,#0Aah            ;CHPCON is TA protected
     MOV  TA,#55h
     ORL  CHPCON,#00000001b        ;IAPEN = 1, enable IAP mode
     RET

    Disable_IAP:
     MOV  TA,#0Aah
     MOV  TA,#55h
     ANL  CHPCON,#11111110b        ;IAPEN = 0, disable IAP mode
     RET

    Enable_AP_Update:
     MOV  TA,#0Aah            ;IAPUEN is TA protected
     MOV  TA,#55h
     ORL  IAPUEN,#00000001b        ;APUEN = 1, enable APROM update
     RET

    Disable_AP_Update:
     MOV  TA,#0Aah
     MOV  TA,#55h
     ANL  IAPUEN,#11111110b        ;APUEN = 0, disable APROM update
     RET

    Enable_CONFIG_Update:
     MOV  TA,#0Aah
     MOV  TA,#55h
     ORL  IAPUEN,#00000100b        ;CFUEN = 1, enable CONFIG update
     RET

    Disable_CONFIG_Update:
     MOV  TA,#0Aah
     MOV  TA,#55h
     ANL  IAPUEN,#11111011b        ;CFUEN = 0, disable CONFIG update
     RET

    Trigger_IAP:
     MOV  TA,#0Aah            ;IAPTRG is TA protected
     MOV  TA,#55h
     ORL  IAPTRG,#00000001b        ;write '1' to IAPGO to trigger IAP
process
     RET

    ;*****************************************************************
    ;     IAP APROM Function
    ;*****************************************************************
```

```
Erase_AP:
 MOV  IAPCN,#PAGE_ERASE_AP
 MOV  IAPFD,#0FFh
 MOV  R0,#00h
Erase_AP_Loop:
 MOV  IAPAH,R0
 MOV  IAPAL,#00h
 CALL Trigger_IAP
 MOV  IAPAL,#80h
 CALL Trigger_IAP
 INC  R0
 CJNE R0,#44h,Erase_AP_Loop
 RET

Program_AP:
 MOV  IAPCN,#BYTE_PROGRAM_AP
 MOV  IAPAH,#00h
 MOV  IAPAL,#00h
 MOV  DPTR,#AP_code
Program_AP_Loop:
 CLR  A
 MOVC A,@A+DPTR
 MOV  IAPFD,A
 CALL Trigger_IAP
 INC  DPTR
 INC  IAPAL
 MOV  A,IAPAL
 CJNE A,#14,Program_AP_Loop
 RET

Program_AP_Verify:
 MOV  IAPCN,#BYTE_READ_AP
 MOV  IAPAH,#00h
 MOV  IAPAL,#00h
 MOV  DPTR,#AP_code
Program_AP_Verify_Loop:
 CALL Trigger_IAP
 CLR  A
 MOVC A,@A+DPTR
 MOV  B,A
 MOV  A,IAPFD
 CJNE A,B,Program_AP_Verify_Error
 INC  DPTR
 INC  IAPAL
 MOV  A,IAPAL
 CJNE A,#14,Program_AP_Verify_Loop
 RET
```

```
Program_AP_Verify_Error:
 CALL Disable_IAP
 MOV  P0,#00h
 SJMP $

;*****************************************************************
;      IAP CONFIG Function
;*****************************************************************
Erase_CONFIG:
 MOV  IAPCN,#ALL_ERASE_CONFIG
 MOV  IAPAH,#00h
 MOV  IAPAL,#00h
 MOV  IAPFD,#0FFh
 CALL Trigger_IAP
 RET

Read_CONFIG:
 MOV  IAPCN,#BYTE_READ_CONFIG
 MOV  IAPAH,#00h
 MOV  IAPAL,#02h
 CALL Trigger_IAP
 MOV  R7,IAPFD
 RET

Program_CONFIG:
 MOV  IAPCN,#BYTE_PROGRAM_CONFIG
 MOV  IAPAH,#00h
 MOV  IAPAL,#02h
 MOV  A,R7
 ANL  A,#11111011b
 MOV  IAPFD,A                    ;disable BOD reset
 MOV  R6,A                       ;temp data
 CALL Trigger_IAP
 RET

Program_CONFIG_Verify:
 MOV  IAPCN,#BYTE_READ_CONFIG
 MOV  IAPAH,#00h
 MOV  IAPAL,#02h
 CALL Trigger_IAP
 MOV  B,R6
 MOV  A,IAPFD
 CJNE A,B,Program_CONFIG_Verify_Error
 RET

Program_CONFIG_Verify_Error:
```

```
    CALL Disable_IAP
    MOV  P0,#00h
    SJMP $

;*********************************************************************
;     APROM code
;*********************************************************************
AP_code:
    DB   75h,0B1h, 00h            ;OPCODEs of "MOV    P0M1,#0"
    DB   75h,0Ach, 00h            ;OPCODEs of "MOV    P3M1,#0"
    DB   75h, 90h, 55h            ;OPCODEs of "MOV    P1,#55h"
    DB   75h,0A0h,0Aah            ;OPCODEs of "MOV    P2,#0Aah"
    DB   80h,0Feh                 ;OPCODEs of "SJMP   $"

    END
```

28 POWER MANAGEMENT

The ML51 has several features that help user to control the power consumption of the device. Table 27.5-1 Table 27.5-1 Power Mode Tablelists all power mode at ML51 series to save the power consumption. For a stable current consumption, the state and mode of each pin should be taken care of. The minimum power consumption can be attained by giving the pin state just the same as the external pulls for example output 1 if pull-high is used or output 0 if pull-low. If the I/O pin is floating, user is recommended to leave it as quasi-bidirectional mode.

Table 27.5-1 Power Mode Table

Mode	Clock Source	Comment
Normal mode	Any clock source	-
Idle mode	Any clock source	Only CPU clock is stoped.
Low power run mode	Only for LIRC or LXT	-
Low power idle mode	Only for LIRC or LXT	Only CPU clock is stoped.
Power-down mode (PD)	Any clock source	1. CPU enters deep sleep mode 2. Most clocks are disabled except ACMP/LIRC/LXT, and only WDT/WKT peripheral clocks still enable if their clock sources are selected as LIRC/LXT.

For each power mode, they have different entry setting and leaving condition. The Table 27.5-2 shows the entry setting for each power mode. When chip power-on, chip is running in normal mode. User can enter each mode by setting IDL (PCON.0), LPR (PCON.5) and PD (PCON.1).

Table 27.5-2 Entry setting of power down mode

Register/Instruction Mode	LPR (PCON.5)	PD (PCON.1)	IDL (PCON.0)
Normal mode	0	0	0
Idle mode	0	0	1
Low power run mode	1	0	0
Low power idle mode	1	0	1
Power-down mode	X	1	X

PCON – Power Control

7	6	5	4	3	2	1	0
SMOD	SMOD0	LPR	POF	GF1	GF0	PD	IDL
R/W	R/W	RW	R/W	R/W	R/W	R/W	R/W

Address: 87H, All pagess POR reset value: 0001 000b, other reset value: 000U 0000b

Bit	Name	Description
5	LPR	Low power run mode 0 = disable 1 = enable **Note:** If PD = 1 and LPR = 1 at the same time, LPR is invalid, CPU will enter power down mode.

Bit	Name	Description
1	PD	Power-down mode Setting this bit puts CPU into Power-down mode. Under this mode, both CPU and peripheral clocks stop and Program Counter (PC) suspends. It provides the lowest power consumption. After CPU is woken up from Power-down, this bit will be automatically cleared via hardware and the program continue executing the interrupt service routine (ISR) of the very interrupt source that woke the system up before. After return from the ISR, the device continues execution at the instruction, which follows the instruction that put the system into Power-down mode. Note that If IDL bit and PD bit are set simultaneously, CPU will enter Power-down mode. Then it does not go to Idle mode after exiting Power-down.
0	IDL	Idle mode Setting this bit puts CPU into Idle mode. Under this mode, Program Counter (PC) suspends but the CPU clock keep running and all peripherals keep activated. After CPU is woken up from Idle, this bit will be automatically cleared via hardware and the program continue executing the ISR of the very interrupt source that woke the system up before. After return from the ISR, the device continues execution at the instruction which follows the instruction that put the system into Idle mode.

28.1 Idle Mode

Idle mode suspends CPU processing by holding the Program Counter. No program code are fetched and run in Idle mode. It forces the CPU state to be frozen. The Program Counter (PC), Stack Pointer (SP), Program Status Word (PSW), Accumulator (ACC), and the other registers hold their contents during Idle mode. The port pins hold the logical states they had at the time Idle was activated. Generally, it saves considerable power of typical half of the full operating power.

Since the clock provided for peripheral function logic circuit like timer or serial port still remain in Idle mode, the CPU can be released from the Idle mode with any of enabled interrupt sources. User can put the device into Idle mode by writing 1 to the bit IDL (PCON.0). The instruction that sets the IDL bit is the last instruction that will be executed before the device enters Idle mode.

The Idle mode can be terminated in two ways. First, as mentioned, any enabled interrupt will cause an exit. It will automatically clear the IDL bit, terminate Idle mode, and the interrupt service routine (ISR) will be executed. After using the RETI instruction to jump out of the ISR, execution of the program will be the one following the instruction, which put the CPU into Idle mode. The second way to terminate Idle mode is with any reset other than software reset. Remember that if Watchdog reset is used to exit Idle mode, the WIDPD (WDCON.4) needs to be set 1 to let WDT keep running in Idle mode.

28.2 Low Power Run Mode

The CPU and the selected peripherals are running with a low speed oscillator (LXT or LIRC). At first system clock should be switch to LXT or LIRC. And then put the device into Low power run mode by writing 1 to the bit LPR (PCON.5) .

The voltage regulator is configured in ultra-low-power mode. The microcontroller enters Low power run mode by software and can exit from this mode by software or by a reset.

28.3 Low Power Idle Mode

Only the device is in Low power run mode, user can put into Low power idle mode by writing 1 to bit IDL (PCON.0).

28.4 Power-Down Mode

Power-down mode is the lowest power state that the ML51 can enter. It remains the power consumption as A ”uA” level by stopping the system clock source. Both of CPU and peripheral functions like Timers or UART are frozen. Flash memory is put into its stop mode. All activity is completely stopped and the power consumption is reduced to the lowest possible value. The device

can be put into Power-down mode by writing 1 to bit PD (PCON.1). The instruction that does this action will be the last instruction to be executed before the device enters Power-down mode. In the Power-down mode, RAM maintains its content. The port pins output the values held by their own state before Power-down respectively.

There are several ways to exit the ML51 from the Power-down mode. The first is with all resets except software reset. Brown-out reset will also wake up CPU from Power-down mode. Be sure that brown-out detection is enabled before the system enters Power-down. However, for least power consumption, it is recommended to enable low power BOD in Power-down mode. Of course the external pin reset and power-on reset will remove the Power-down status. After the external reset or power-on reset. The CPU is initialized and start executing program code from the beginning.

The second way to wake the ML51 up from the Power-down mode is by an enabled external interrupt. The trigger on the external pin will asynchronously restart the system clock. After oscillator is stable, the device executes the interrupt service routine (ISR) for the corresponding external interrupt. After the ISR is completed, the program execution returns to the instruction after the one, which puts the device into Power-down mode and continues. Interrupts that allows to wake up CPU from Power-down mode includes external interrupt $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$, pin interrupt, WDT interrupt, WKT interrupt, and brown-out interrupt.

29 CLOCK SYSTEM

The ML51 has a wide variety of clock sources and selection features that allow it to be used in a wide range of applications while maximizing performance and minimizing power consumption. The ML51 provides five options of the system clock sources including internal oscillator, crystal/resonator, or external clock from XIN pin via software. The ML51 is embedded with two internal oscillators: one 38.4 kHz low-speed and one 24 MHz high-speed, which is factory trimmed to ±2% under all conditions. A clock divider CKDIV is also available on ML51 for adjustment of the flexibility between power consumption and operating performance.

29.1 System Clock Block Diagram

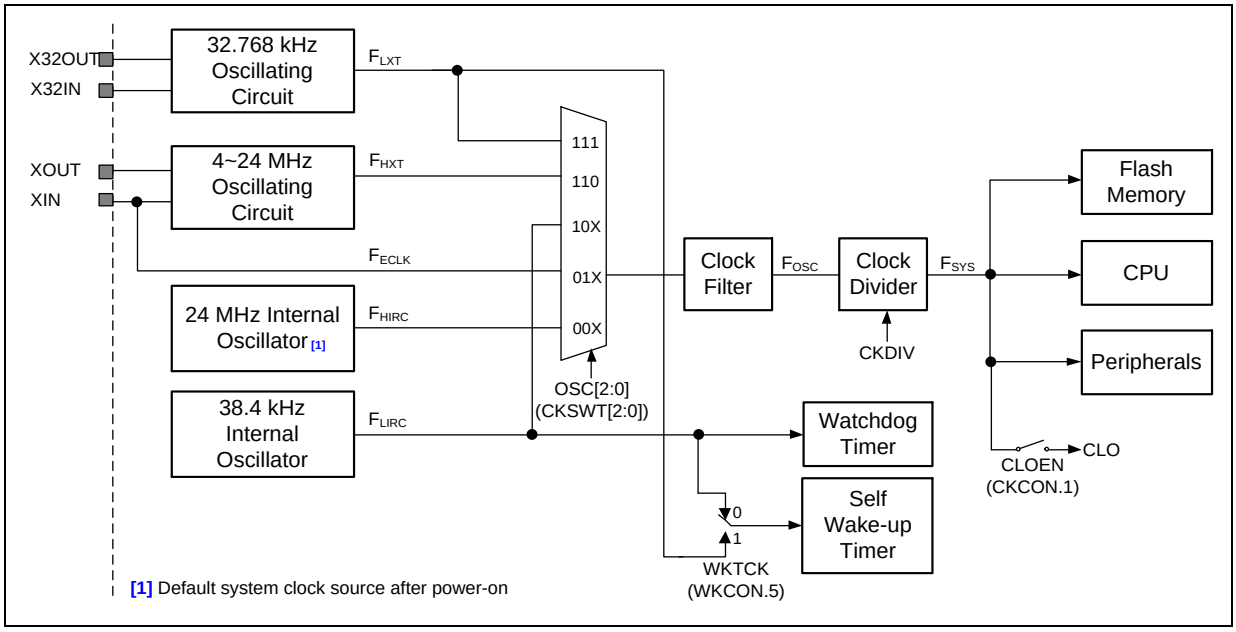


Figure 29.1-1 Clock System Block Diagram

29.2 System Clock Sources

There are a total of three system clock sources selectable in the ML51 including high-speed internal oscillator, low-speed internal oscillator and external clock input. Each of them can be the system clock source in the ML51. Different active system clock sources also affect multi-function of P5.3/XIN, P5.2/XOUT and P5.5/X32IN, P5.4/X32OUT pins.

29.2.1 Internal Oscillators

There are two internal oscillators in the ML51 – one 24 MHz high-speed internal oscillator (HIRC) and one 38.4 kHz low-speed (LIRC). Both of them can be selected as the system clock. HIRC can be enabled by setting HIRCEN (CKEN.5). LIRC is enabled after device is powered up. User can set OSC[2:0] (CKSWT [2:0]) as [0,0,x] to select the HIRC as the system clock. By setting OSC[2:0] as [1,0,x], LIRC will be selected as the system clock. Note that after the ML51 is powered, HIRC and LIRC will be both enabled and HIRC is default selected as the system clock source. While using internal oscillators, XIN, XOUT, X32I and X32O automatically switch as one general purpose I/O to expend the number of general purpose I/O.

29.2.2 External Crystal/Resonator or Clock Input

There are three possible clock source options of external clock sources – 4 MHz to 24 MHz high-speed crystal/resonator (HXT), 32.768 kHz low-speed crystal/resonator (LXT), and the external clock input (ECLK) through XIN pin. User can set OSC[2:0] as [0,1,x] to select ECLK as the system clock. By setting OSC[2:0] as [1,1,0], HXT will be selected as the system clock. By setting OSC[2:0] as [1,1,1],

LXT will be selected as the system clock. When HXT or LXT is used as the system clock, X_{IN}/ $X32_{IN}$ and X_{OUT}/ $X32_{OUT}$ are the input and output, respectively, of an internal inverting amplifier. A crystal or resonator should be connected between XIN/ $X32_{IN}$ and XOUT/$X32O_{UT}$ pins. When enabling and selecting ECLK as the system clock source, the system clock is supplied via the X_{IN} pin. The common application is to drive X_{IN} with an active oscillator or clocks from another host device. Be aware that user should never feed any clock signal larger than voltage 1.8V to X_{IN} and give a DC voltage to X_{OUT} pin which value is half of X_{IN}, when ECLK mode is selected. Otherwise, it may break the device.

XLTCON – XLT Clock Control

Register	SFR Address	Reset Value
XLTCON	85H, Page 1, TA protected	0111 _ 0111b

7	6	5	4	3	2	1	0
HSCH	HXSG			-	-	LXSG	
R/W	R/W			-	-	R/W	

Bit	Name	Description
7	HSCH	HXT Schmitt trigger select 0 = disable 1 = enable
6:4	HXSG	HXT gain value select 000 = L0 mode (smallest value) 001 = L1 mode 010 = L2 mode 011 = L3 mode 100 = L4 mode 101 = L5 mode 110 = L6 mode 111 = L7 mode (largest value)
3:2	-	Reserved
1:0	LXSG	LXT gain value select 00 = L0 mode (smallest value) 01 = L1 mode 10 = L2 mode 11 = L3 mode (largest value)

29.3 System Clock Switching

The ML51 supports clock source switching on-the-fly by controlling CKSWT and CKEN registers via software. It provides a wide flexibility in application. Note that these SFR are writing TA protected for precaution. With this clock source control, the clock source can be switched between the external clock source and the internal oscillator, even between the high and low-speed internal oscillator. However, during clock source switching, the device requires some amount of warm-up period for an original disabled clock source. Therefore, use should follow steps below to ensure a complete clock source switching. User can enable the target clock source by writing proper value into CKEN register, wait for the clock source stable by polling its status bit in CKSWT register, and switch to the target clock source by changing OSC[2:0] (CKSWT[2:0]). After these step, the clock source switching is successful and then user can also disable the original clock source if power consumption is concerned. Note that if not following the steps above, the hardware will take certain actions to deal with such illegal operations as follows.

1. If user tries to disable the current clock source by changing CKEN value, the device will ignore this action. The system clock will remain the original one and CKEN will remain the original value.

2. If user tries to switch the system clock source to a disabled one by changing OSC[2:0] value, OSC[2:0] value will be updated right away. But the system clock will remain the original one and CKSWTF flag will be set by hardware.

3. Once user switches the system clock source to an enabled but still instable one, the hardware will wait for stabilization of the target clock source and then switch to it in the background. During this waiting period, the device will continue executing the program with the original clock source and CKSWTF will be set as 1. After the stable flag of the target clock source (see CKSWT[7:3]) is set and the clock source switches successfully, CKSWTF will be cleared as 0 automatically by hardware.

CKSWT – Clock Switch

Register	SFR Address	Reset Value
CKSWT	96H, PAGE 0, TA protected	0011 _0000 b

7	6	5	4	3	2	1	0
HXTST	LXTST	HIRCST	LIRCST	ECLKST	OSC[2:0]		
R	R	R	R	R	W		

Bit	Name	Description
7	HXTST	**High speed external crystal/resonator 4 MHz to 24 MHz status** 0 = High speed external crystal/resonator is not stable or is disabled. 1 = High speed external crystal/resonator is enabled and stable.
6	LXTST	**Low speed external crystal/resonator 32.768 kHz status** 0 = Low speed external crystal/resonator is not stable or is disabled. 1 = Low speed external crystal/resonator is enabled and stable.
5	HIRCST	**High-speed internal oscillator 24 MHz status** 0 = High-speed internal oscillator is not stable or disabled. 1 = High-speed internal oscillator is enabled and stable.

Bit	Name	Description
4	LIRCST	**Low speed internal oscillator 38.4 kHz status** 0 = Low speed internal oscillator is not stable or is disabled. 1 = Low speed internal oscillator is enabled and stable.
3	ECLKST	**External clock input status** 0 = External clock input is not stable or disabled. 1 = External clock input is enabled and stable.
2:0	OSC[2:0]	**Oscillator selection bits** This field selects the system clock source. 00x = Internal 24 MHz oscillator. Defaul value accoding to HIRCEN(CKEN.5) enabled. 01x = External oscillator clock source accoding to ECLKEN(CKEN.3) enabled. 10x = Internal 38.4 kHz oscillator according to LIRCEN(CKEN.4) enabled. 110 = External High speed crystal/resonator clock source (4 MHz ~ 24 MHz) accoding to EHXTEN(CKEN.7) enabled. 111 = External Low speed crystal/resonator clock source (32.768 kHz) accoding to ELXTEN(CKEN.6) enabled. Note that this field is write only. The read back value of this field may not correspond to the present system clock source.

CKEN – Clock Enable

Register	SFR Address	Reset Value
CKEN	97H, PAGE 0, TA protected	0011 _0100 b

7	6	5	4	3	2	1	0
EHXTEN	ELXTEN	HIRCEN	LIRCEN	ECLKEN	-	-	CKSWTF
R/W	R/W	R/W	R/W	R/W	-	-	R

Bit	Name	Description
7	EHXTEN	**External High-speed crystal/resonator enable** 1 = High-speed external crystal/resonator 4 MHz to 24 MHz Enabled. 0 = High-speed external crystal/resonator 4 MHz to 24 MHz Disabled, P5.2 and P5.3 work as general purpose I/O or other functions if ECLKEN set to 0.
6	ELXTEN	**External Low-speed crystal/resonator enable** 1 = Low-speed external crystal/resonator 32.768 kHz Enabled. 0 = Low-speed external crystal/resonator 32.768 kHz Disabled, P5.4 and P5.5 work as general purpose I/O or other functions.
5	HIRCEN	**High-speed internal oscillator 24 MHz enable** 0 = The high-speed internal oscillator Disabled. 1 = The high-speed internal oscillator Enabled. Note that once IAP is enabled by setting IAPEN (CHPCON.0), the high-speed internal 24 MHz oscillator will be enabled automatically. The hardware will also set HIRCEN and HIRCST bits. After IAPEN is cleared, HIRCEN and EHRCST resume the original values.

Bit	Name	Description
4	LIRCEN	**Low speed internal oscillator 38.4 kHz enable** 0 = The low speed internal oscillator Disabled. 1 = The low speed internal oscillator Enabled. Note that when (1)WDT is enabled, (2)WKT is running by the clock source of the internal 38.4 kHz oscillator ,(3) BOD is enabled, or (4)LVR filter is enabled, a write 0 to LIRCEN will be ignored. LIRCEN is always 1 and the internal 38.4 kHz oscillator is always enabled.
3	ECLKEN	**External Clock Input enable** 1 = External clock input (XIN , P5.3) Enabled. 0 = External clock input (XIN, P5.3) Disabled, P5.2 and P5.3 work as general purpose I/O or other functions if EHXTEN set to 0.
1	-	Reserved
0	CKSWTF	**Clock switch fault flag** 0 = The previous system clock source switch was successful. 1 = User tried to switch to an instable or disabled clock source at the previous system clock source switch. If switching to an instable clock source, this bit remains 1 until the clock source is stable and switching is successful.

29.4 System Clock Divider

The oscillator frequency (F_{OSC}) can be divided down, by an integer, up to 1/510 by configuring a dividing register, CKDIV, to provide the system clock (F_{SYS}). This feature makes it possible to temporarily run the MCU at a lower rate, reducing power consumption. By dividing the clock, the MCU can retain the ability to respond to events other than those that can cause interrupts (i.e. events that allow exiting the Idle mode) by executing its normal program at a lower rate. This can often result in lower power consumption than in Idle mode. This can allow bypassing the oscillator start-up time in cases where Power-down mode would otherwise be used. The value of CKDIV may be changed by the program at any time without interrupting code execution.

CKDIV – Clock Divider

Register	SFR Address	Reset Value
CKDIV	C1H, Page 1	0000_0000 b

7	6	5	4	3	2	1	0
CKDIV[7:0]							
R/W							

Bit	Name	Description
7:0	CKDIV[7:0]	**Clock divider** The system clock frequency F_{SYS} follows the equation below according to CKDIV value. $F_{SYS} = F_{OSC}$, while CKDIV = 00H, $F_{SYS} = \frac{F_{OSC}}{2 \times CKDIV}$, while CKDIV = 01H to FFH.

29.5 System Clock Output

The ML51 provides a CLO pin that outputs the system clock. Its frequency is the same as F_{SYS}. The output enable bit is CLOEN (CKCON.1). CLO output stops when device is put in its Power-down mode because the system clock is turned off. Note that when noise problem or power consumption is important issue, user had better not enable CLO output.

CKCON – Clock Control

Register	SFR Address	Reset Value
CKCON	8EH, Page 0	1000_0000b

7	6	5	4	3	2	1	0
FASTWK	PWMCKS	T1OE	T1M	T0M	T0OE	CLOEN	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	-

Bit	Name	Description
1	CLOEN	System clock output enable 0 = System clock output Disabled. 1 = System clock output Enabled from CLO pin.

30 POWER MONITORING

To prevent incorrect execution during power up and power drop, The ML51 provide three power monitor functions, power-on detection and brown-out detection.

30.1 Power-On Detection

The power-on detection function is designed for detecting power up after power voltage reaches to a level where system can work. After power-on detected, the POF (PCON.4) will be set 1 to indicate a cold reset, a power-on reset complete. The POF flag can be cleared via software.

PCON – Power Control

7	6	5	4	3	2	1	0
SMOD	SMOD0	LPR	POF	GF1	GF0	PD	IDL
R/W	R/W	RW	R/W	R/W	R/W	R/W	R/W

Address: 87H, All pagess POR reset value: 0001 000b, other reset value: 000U 0000b

Bit	Name	Description
4	POF	Power-on reset flag This bit will be set as 1 after a power-on reset. It indicates a cold reset, a power-on reset complete. This bit remains its value after any other resets. This flag is recommended to be cleared via software.

30.2 Brown-Out Detection (BOD)

The other power monitoring function brown-out detection (BOD) circuit is used for monitoring the V_{DD} level during execution. There are eight CONFIG selectable brown-out trigger levels available for wide voltage applications. These eight nominal levels are 1.8V, 2.0V, 2.4V, 2.7V, 3.0V, 3.7V and 4.4V selected via setting CBOV[2:0] (CONFIG2[6:4]). BOD level can also be changed by setting BOV[2:0] (BODCON0[6:4]) after power-on. When V_{DD} drops to the selected brown-out trigger level (V_{BOD}), the BOD logic will either reset the MCU or request a brown-out interrupt. User may decide to being reset or generating a brown-out interrupt according to different applications. V_{BOD} also can be set by software after power-on. Note that BOD output is not available until 2~3 LIRC clocks after software enabling.

The BOD will request the interrupt while V_{DD} drops below V_{BOD} while BORST (BODCON0.2) is 0. In this case, BOF (BODCON0.3) will be set as 1. After user cleared this flag whereas V_{DD} remains below V_{BOD}, BOF will not set again. BOF just acknowledge user a power drop occurs. The BOF will also be set as 1 after V_{DD} goes higher than V_{BOD} to indicate a power resuming. The BOD circuit provides an useful status indicator BOS (BODCON0.0), which is helpful to tell a brown-out event or power resuming event occurrence. If the BORST bit is set as 1, this will enable brown-out reset function. After a brown-out reset, BORF (BODCON0.1) will be set as 1 via hardware. It will not be altered by reset other than power-on. This bit can be cleared by software. Note that all bits in BODCON0 is writing protected by timed access (TA).

The ML51 provides low power BOD mode for saving current consumption and remaining BOD functionality with limited detection response. By setting LPBOD[1:0] (BODCON1[2:1]), the BOD circuit can be periodically enabled to sense the power voltage nominally every 1.6 ms, 6.4 ms, or 25.6 ms. It saves much power but also provides low-speed power voltage sensing. Note that the hysteresis feature will disappear in low power BOD mode.

For a noise sensitive system, the ML51 has a BOD filter which filters the power noise to avoid BOD event triggering unconsciously. The BOD filter is enabled by default and can be disabled by setting BODFLT (BODCON1.0) as 0 if user requires a rapid BOD response. The minimum brown-out detect pulse width is listed in LPBOD[1:0] (BODCON1)

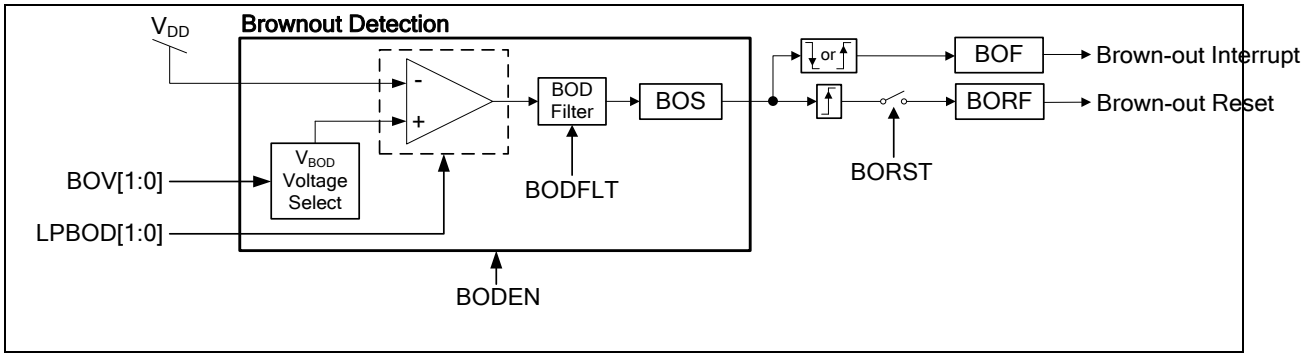


Figure 30.2-1 Brown-out Detection Block Diagram

CONFIG2

7	6	5	4	3	2	1	0
CBODEN	CBOV[2:0]			BOIAP	CBORST	-	-
R/W	R/W			R/W	R/W	-	-

Factory default value: 1111 1111b

Bit	Name	Description
7	CBODEN	CONFIG brown-out detect enable 1 = Brown-out detection circuit on. 0 = Brown-out detection circuit off.
6:4	CBOV[2:0]	CONFIG brown-out voltage select 111 = V_{BOD} is 1.8V. 110 = V_{BOD} is 1.8V. 101 = V_{BOD} is 2.0V. 100 = V_{BOD} is 2.4V. 011 = V_{BOD} is 2.7V. 010 = V_{BOD} is 3.0V. 001 = V_{BOD} is 3.7V. 000 = V_{BOD} is 4.4V.
3	BOIAP	Brown-out inhibiting IAP This bit decides whether IAP erasing or programming is inhibited by brown-out status. This bit is valid only when brown-out detection is enabled. 1 = IAP erasing or programming is inhibited if V_{DD} is lower than V_{BOD}. 0 = IAP erasing or programming is allowed under any workable V_{DD}.
2	CBORST	CONFIG brown-out reset enable This bit decides whether a brown-out reset is caused by a power drop below V_{BOD}. 1 = Brown-out reset Enabled. 0 = Brown-out reset Disabled.

BODCON0 – Brown-out Detection Control 0 (TA protected)

7	6	5	4	3	2	1	0
BODEN[1]	BOV[2:0] [1]			BOF[2]	BORST[1]	BORF	BOS
R/W	R/W			R/W	R/W	R/W	R

Address: A3H, Page 0 Reset value: POR: CCCC XC0Xb / BOD: UUUU XU1Xb / Others: UUUU XUUXb

Bit	Name	Description
7	BODEN	Brown-out detection enable 0 = Brown-out detection circuit off. 1 = Brown-out detection circuit on. Note that BOD output is not available until 2~3 LIRC clocks after enabling.
6:4	BOV[2:0]	CONFIG brown-out voltage select 111 = VBOD is 1.8V. 110 = VBOD is 1.8V. 101 = VBOD is 2.0V. 100 = VBOD is 2.4V. 011 = VBOD is 2.7V. 010 = VBOD is 3.0V. 001 = VBOD is 3.7V. 000 = VBOD is 4.4V.
3	BOF	Brown-out interrupt flag This flag will be set as logic 1 via hardware after a V_{DD} dropping below or rising above VBOD event occurs. If both EBOD (IE.5) and EA (IE.7) are set, a brown-out interrupt requirement will be generated. This bit should be cleared via software.
2	BORST	Brown-out reset enable This bit decides whether a brown-out reset is caused by a power drop below VBOD. 0 = Brown-out reset when V_{DD} drops below VBOD Disabled. 1 = Brown-out reset when V_{DD} drops below VBOD Enabled.
1	BORF	Brown-out reset flag When the MCU is reset by brown-out event, this bit will be set via hardware. This flag is recommended to be cleared via software.
0	BOS	Brown-out status This bit indicates the V_{DD} voltage level comparing with VBOD while BOD circuit is enabled. It keeps 0 if BOD is not enabled. 0 = V_{DD} voltage level is higher than VBOD or BOD is disabled. 1 = V_{DD} voltage level is lower than VBOD. Note that this bit is read-only.

Note:
[1] BODEN, BOV[2:0], and BORST are initialized by being directly loaded from CONFIG2 bit 7, [6:4], and 2 after all resets.
[2] BOF reset value depends on different setting of CONFIG2 and V_{DD} voltage level. Please checkTable 30.2-1

Table 30.2-1 BOF Reset Value

CBODEN (CONFIG2.7)	CBORST (CONFIG2.2)	V_{DD} Level	BOF
1	1	> VBOD always	0
1	0	< VBOD	1
1	0	> VBOD	0
0	X	X	0

BODCON0 – Brown-out Detection Control 0

Register	SFR Address	Reset Value
BODCON0	A3H, Page 0, TA protected	POR: CCCC_XC0Xb BOD: UUUU_XU1Xb Others: UUUU_XUUXb

7	6	5	4	3	2	1	0
-	-	-	-	-	LPBOD[1:0]		BODFLT
-	-	-	-	-	R/W		R/W

Address: ABH, Page 0 Reset value: POR: 0000 0001b / Others:0000 0UUUb

Bit	Name	Description
7:3	-	Reserved
2:1	LPBOD[1:0]	Low power BOD enable 00 = BOD normal mode. BOD circuit is always enabled. 01 = BOD low power mode 1 by turning on BOD circuit every 1.6 ms periodically. 10 = BOD low power mode 2 by turning on BOD circuit every 6.4 ms periodically. 11 = BOD low power mode 3 by turning on BOD circuit every 25.6 ms periodically.
0	BODFLT	BOD filter control BOD has a filter which counts 32 clocks of F_{SYS} to filter the power noise when MCU runs with HIRC, or ECLK as the system clock and BOD does not operates in its low power mode (LPBOD[1:0] = [0, 0]). In other conditions, the filter counts 2 clocks of LIRC. Note that when CPU is halted in Power-down mode. The BOD output is permanently filtered by 2 clocks of LIRC. The BOD filter avoids the power noise to trigger BOD event. This bit controls BOD filter enabled or disabled. 0 = BOD filter Disabled. 1 = BOD filter Enabled. (Power-on reset default value.)

Table 30.2-2 Minimum Brown-out Detect Pulse Width

BODFLT (BODCON1.1)	BOD Operation Mode	System Clock Source	Minimum Brown-Out Detect Pulse Width
0	Normal mode (LPBOD[1:0] = [0,0])	Any clock source	Typ. 1μs
	Low power mode 1 (LPBOD[1:0] = [0,1])	Any clock source	16 ($1/F_{LIRC}$)
	Low power mode 2 (LPBOD[1:0] = [1,0])	Any clock source	64 ($1/F_{LIRC}$)
	Low power mode 3 (LPBOD[1:0] = [1,1])	Any clock source	256 ($1/F_{LIRC}$)
1	Normal mode (LPBOD[1:0] = [0,0])	HIRC/ECLK	Normal operation: 32 ($1/F_{SYS}$) Idle mode: 32 ($1/F_{SYS}$) Power-down mode: 2 ($1/F_{LIRC}$)
		LIRC	2 ($1/F_{LIRC}$)
	Low power mode 1 (LPBOD[1:0] = [0,1])	Any clock source	18 ($1/F_{LIRC}$)
	Low power mode 2 (LPBOD[1:0] = [1,0])	Any clock source	66 ($1/F_{LIRC}$)
	Low power mode 3 (LPBOD[1:0] = [1,1])	Any clock source	258 ($1/F_{LIRC}$)

31 RESET

The ML51 has several options to place device in reset condition. It also offers the software flags to indicate the source, which causes a reset. In general, most SFR go to their Reset value irrespective of the reset condition, but there are several reset source indicating flags whose state depends on the source of reset. User can read back these flags to determine the cause of reset using software. There are five ways of putting the device into reset state. They are power-on reset, brown-out reset, external reset, WDT reset, and software reset.

31.1 Power-On Reset and Low Voltage Reset

The ML51 incorporates an internal power-on reset (POR) and a low voltage reset (LVR). During a power-on process of rising power supply voltage V_{DD}, the POR or LVR will hold the MCU in reset mode when V_{DD} is lower than the voltage reference thresholds. This design makes CPU not access program Flash while the V_{DD} is not adequate performing the Flash reading. If an undetermined operating code is read from the program Flash and executed, this will put CPU and even the whole system in to an erroneous state. After a while, V_{DD} rises above the threshold where the system can work, the selected oscillator will start and then program code will execute from 0000H. At the same time, a power-on flag POF (PCON.4) will be set 1 to indicate a cold reset, a power-on process complete. Note that the contents of internal RAM will be undetermined after a power-on. It is recommended that user gives initial values for the RAM block.

The POF is recommended to be cleared to 0 via software to check if a cold reset or warm reset performed after the next reset occurs. If a cold reset caused by power off and on, POF will be set 1 again. If the reset is a warm reset caused by other reset sources, POF will remain 0. User may take a different course to check other reset flags and deal with the warm reset event. For detailed electrical characteristics.

PCON – Power Control

Register	SFR Address	Reset Value
PCON	87H, All pagess	POR: 0001_000b, other: 000U_0000b

7	6	5	4	3	2	1	0
SMOD	SMOD0	LPR	POF	GF1	GF0	PD	IDL
R/W	R/W	RW	R/W	R/W	R/W	R/W	R/W

Bit	Name	Description
4	POF	Power-on reset flag This bit will be set as 1 after a power-on reset. It indicates a cold reset, a power-on reset complete. This bit remains its value after any other resets. This flag is recommended to be cleared via software.

LVRDIS – LVR Disable

Register	SFR Address	Reset Value
LVRDIS	FFH, Page 1, TA protected	0000_0000 b

7	6	5	4	3	2	1	0
LVRDIS[7:0]							
W							

Bit	Name	Description
7:0	LVRDIS[7:0]	LVR disable To first writing 5AH to the LVRDIS and immediately followed by a writing of A5H will disable LVR.

31.2 Brown-Out Reset

The brown-out detection circuit is used for monitoring the V_{DD} level during execution. When V_{DD} drops to the selected brown-out trigger level (V_{BOD}), the brown-out detection logic will reset the MCU if BORST (BODCON0.2) setting 1. After a brown-out reset, BORF (BODCON0.1) will be set as 1 via hardware. BORF will not be altered by any reset other than a power-on reset or brown-out reset itself. This bit can be set or cleared by software.

BODCON0 – Brown-out Detection Control 0

Register	SFR Address	Reset Value
BODCON0	A3H, Page 0, TA protected	POR: CCCC_XC0Xb BOD: UUUU_XU1Xb Others: UUUU_XUUXb

7	6	5	4	3	2	1	0
BODEN	BOV[2:0]			BOF	BORST	BORF	BOS
R/W	R/W			R/W	R/W	R/W	R

Bit	Name	Description
1	BORF	Brown-out reset flag When the MCU is reset by brown-out event, this bit will be set via hardware. This flag is recommended to be cleared via software.

31.3 External Reset and Hard Fault Reset

The external reset pin nRESET is an input with a Schmitt trigger. An external reset is accomplished by holding the nRESET pin low for at least 24 system clock cycles to ensure detection of a valid hardware reset signal. The reset circuitry then synchronously applies the internal reset signal. Thus, the reset is a synchronous operation and requires the clock to be running to cause an external reset.

Once the device is in reset condition, it will remain as long as nRESET pin is low. After the nRESET high is removed, the MCU will exit the reset state and begin code executing from address 0000H. If an external reset applies while CPU is in Power-down mode, the way to trigger a hardware reset is slightly different. Since the Power-down mode stops system clock, the reset signal will asynchronously cause the system clock resuming. After the system clock is stable, MCU will enter the reset state.

There is a RSTPINF (AUXR0.6) flag, which indicates an external reset took place. After the external reset, this bit will be set as 1 via hardware. RSTPINF will not change after any reset other than a power-on reset or the external reset itself. This bit can be cleared via software.

Hard Fault reset will occur if CPU fetches instruction address over Flash size, HardF (AUXR0.5) flag will be set via hardware. HardF will not change after any reset other than a power-on reset or the external reset itself. This bit can be cleared via software. If MCU run in OCD debug mode and OCDEN = 0, hard fault reset will be disabled. Only HardF flag be asserted.

AUXR0 – Auxiliary Register 0

Register	SFR Address	Reset Value
AUXR0	A2H, Page:0	POR: 0000_0000b Software: 1UU0_0000b Reset pin: U1U0_0000b Hard fault: UU10_0000b Others: UUU0_0000b

7	6	5	4	3	2	1	0
SWRF	RSTPINF	HardF	HardFInt	GF2	-	0	DPS
R/W	R/W	R/W	R/W	R/W	-	R	R/W

Address: A2H, Page:0
Reset value: POR: 0000 0000b / Software: 1UU0 0000b / Reset pin: U1U0 0000b / Hard fault: UU10 0000b / Others: UUU0 0000b

Bit	Name	Description
6	RSTPINF	External reset flag When the MCU is reset by the external reset, this bit will be set via hardware. It is recommended that the flag be cleared via software.
5	HardF	Hard Fault reset flag Once CPU fetches instruction address over Flash size while EHFI (EIE1.4)=0, MCU will reset and this bit will be set via hardware. It is recommended that the flag be cleared via software. **Note:** If MCU run in OCD debug mode and OCDEN = 0, Hard fault reset will disable. Only HardF flag be asserted.

31.4 Watchdog Timer Reset

The WDT is a free running timer with programmable time-out intervals and a dedicated internal clock source. User can clear the WDT at any time, causing it to restart the counter. When the selected time-out occurs but no software response taking place for a while, the WDT will reset the system directly and CPU will begin execution from 0000H.

Once a reset due to WDT occurs, the WDT reset flag WDTRF (WDCON.3) will be set. This bit keeps unchanged after any reset other than a power-on reset or WDT reset itself. User can clear WDTRF via software.

WDCON – Watchdog Timer Control

Register	SFR Address	Reset Value
WDCON	AAH, Page 0, TA protected	POR: 0000 0111b WDT: 0000 1UUUb Others: 0000 UUUUb

7	6	5	4	3	2	1	0
WDTR	WDCLR	WDTF	WIDPD	WDTRF	WDPS[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W		

Bit	Name	Description

Bit	Name	Description
3	WDTRF	WDT reset flag When the CPU is reset by WDT time-out event, this bit will be set via hardware. This flag is recommended to be cleared via software after reset.

31.5 Software Reset

The ML51 provides a software reset, which allows the software to reset the whole system just similar to an external reset, initializing the MCU as it reset state. The software reset is quite useful in the end of an ISP progress. For example, if an ISP of Boot Code updating User Code finishes, a software reset can be asserted to re-boot CPU to execute new User Code immediately. Writing 1 to SWRST (CHPCON.7) will trigger a software reset. Note that this bit is writing TA protection. The instruction that sets the SWRST bit is the last instruction that will be executed before the device reset. See demo code below.

If a software reset occurs, SWRF (AUXR0.7) will be automatically set by hardware. User can check it as the reset source indicator. SWRF keeps unchanged after any reset other than a power-on reset or software reset itself. SWRF can be cleared via software.

CHPCON – Chip Control

Register	SFR Address	Reset Value
CHPCON	9FH, All pages, TA protected	Software: 0000_00U0 b others: 0000_00C0 b

7	6	5	4	3	2	1	0
SWRST	IAPFF	-	-	-	-	BS	IAPEN
W	R/W	-	-	-	-	R/W	R/W

Bit	Name	Description
7	SWRST	Software reset To set this bit as logic 1 will cause a software reset. It will automatically be cleared via hardware after reset is finished.

AUXR0 – Auxiliary Register 0

Register	SFR Address	Reset Value
AUXR0	A2H, Page:0	POR: 0000_0000b Software: 1UU0_0000b Reset pin: U1U0_0000b Hard fault: UU10_0000b Others: UUU0_0000b

7	6	5	4	3	2	1	0
SWRF	RSTPINF	HardF	HardFInt	GF2	-	0	DPS
R/W	R/W	R/W	R/W	R/W	-	R	R/W

Bit	Name	Description

Bit	Name	Description
7	SWRF	Software reset flag When the MCU is reset via software reset, this bit will be set via hardware. It is recommended that the flag be cleared via software.

The software demo code is listed below.

```
ANL   AUXR0,#01111111b    ;software reset flag clear
CLR   EA
MOV   TA,#0Aah
MOV   TA,#55h
ORL   CHPCON,#10000000b   ;software reset
```

31.6 Boot Select

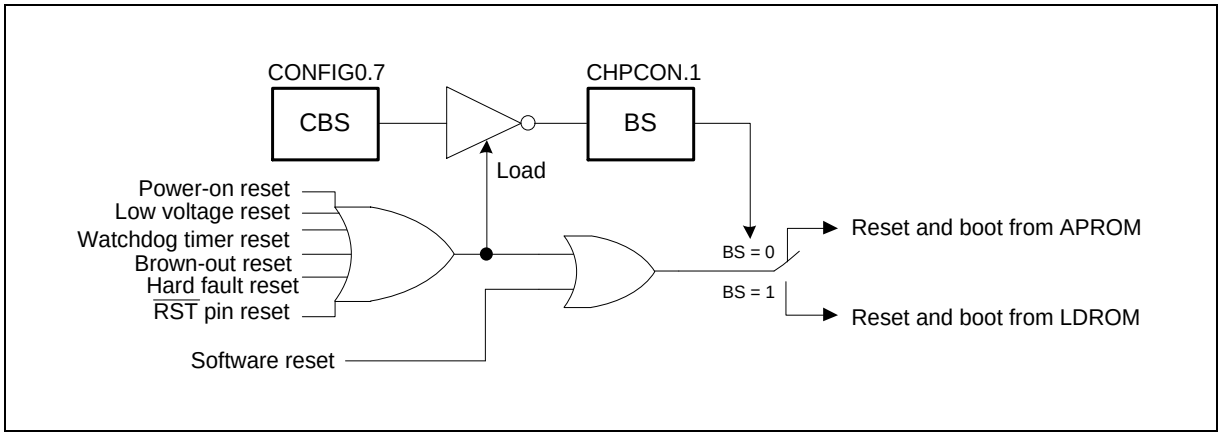


Figure 31.6-1 Boot Selecting Diagram

The ML51 provides user a flexible boot selection for variant application. The SFR bit BS in CHPCON.1 determines MCU booting from APROM or LDROM after any source of reset. If reset occurs and BS is 0, MCU will reboot from address 0000H of APROM. Else, the CPU will reboot from address 0000H of LDROM. Note that BS is loaded from the inverted value of CBS bit in CONFIG0.7 after all resets except software reset.

Note: After the MCU is released from reset state, the hardware will always check the BS bit instead of the CBS bit to determine from which block that the device reboots.

CONFIG0

7	6	5	4	3	2	1	0
CBS	-	OCDPWM	OCDEN	-	-	LOCK	-
R/W	-	R/W	R/W	-	-	R/W	-

Factory default value: 1111 1111b

Bit	Name	Description

Bit	Name	Description
7	CBS	CONFIG boot select This bit defines from which block that MCU re-boots after resets except software reset. 1 = MCU will re-boot from APROM after resets except software reset. 0 = MCU will re-boot from LDROM after resets except software reset.

CHPCON – Chip Control

Register	SFR Address	Reset Value
CHPCON	9FH, All pages, TA protected	Software: 0000_00U0 b others: 0000_00C0 b

7	6	5	4	3	2	1	0
SWRST	IAPFF	-	-	-	-	BS[1]	IAPEN
W	R/W	-	-	-	-	R/W	R/W

Bit	Name	Description
1	BS	Boot select This bit defines from which block that MCU re-boots after all resets. 0 = MCU will re-boot from APROM after all resets. 1 = MCU will re-boot from LDROM after all resets.

Note: BS is initialized by being loaded from the inverted value of CBS bit in CONFIG0.7 after resets except software reset. It keeps unchanged after software reset.

31.7 Reset State

The reset state besides power-on reset does not affect the on-chip RAM. The data in the RAM will be preserved during the reset. After the power-on reset the RAM contents will be indeterminate.

After a reset, most of SFR go to their initial values except bits, which are affected by different reset events.. The Program Counter is forced to 0000H and held as long as the reset condition is applied. Note that the Stack Pointer is also reset to 07H and thus the stack contents may be effectively lost during the reset event even though the RAM contents are not altered.

After a reset, all peripherals and interrupts are disabled. The I/O port latches resumes FFH and I/O mode input-only.

32 AUXILIARY FEATURES

32.1 Dual DPTRs

The original 8051 contains one DPTR (data pointer) only. With single DPTR, it is difficult to move data form one address to another with wasting code size and low performance. The ML51 provides two data pointers. Thus, software can load both a source and a destination address when doing a block move. Once loading, the software simply switches between DPTR and DPTR1 by the active data pointer selection DPS (AUXR0.0) bit.

An example of 64 bytes block move with dual DPTRs is illustrated below. By giving source and destination addresses in data pointers and activating cyclic makes block RAM data move more simple and efficient than only one DPTR. The INC AUXR0 instruction is the shortest (2 bytes) instruction to accomplish DPTR toggling rather than ORL or ANL. For AUXR0.1 contains a hard-wired 0, it allows toggling of the DPS bit by incrementing AUXR0 without interfering with other bits in the register.

```
      MOV   R0,#64                ;number of bytes to move
      MOV   DPTR,#D_Addr          ;load destination address
      INC   AUXR0             ;change active DPTR
      MOV   DPTR,#S_Addr          ;load source address
    LOOP:
      MOVX  A,@DPTR           ;read source data byte
      INC   AUXR0             ;change DPTR to destination
      MOVX  @DPTR,A           ;write data to destination
      INC   DPTR              ;next destination address
      INC   AUXR0             ;change DPTR to source
      INC   DPTR              ;next source address
      DJNZ  R0,LOOP
      INC   AUXR0             ;(optional) restore DPS
```

AUXR0 also contains a general purpose flag GF2 in its bit 3 that can be set or cleared by the user via software.

DPL – Data Pointer Low Byte

Register	SFR Address	Reset Value
DPL	82H, All pages	0000_0000b

7	6	5	4	3	2	1	0
DPL[7:0]							
R/W							

Bit	Name	Description
7:0	DPL[7:0]	Data pointer low byte This is the low byte of 16-bit data pointer. DPL combined with DPH serve as a 16-bit data pointer DPTR to access indirect addressed RAM or Program Memory. DPS (AUXR0.0) bit decides which data pointer, DPTR or DPTR1, is activated.

DPH – Data Pointer High Byte

Register	SFR Address	Reset Value
DPH	83H, All pages	0000_0000b

7	6	5	4	3	2	1	0
DPH[7:0]							
R/W							

Bit	Name	Description
7:0	DPH[7:0]	Data pointer high byte This is the high byte of 16-bit data pointer. DPH combined with DPL serve as a 16-bit data pointer DPTR to access indirect addressed RAM or Program Memory. DPS (AUXR0.0) bit decides which data pointer, DPTR or DPTR1, is activated.

AUXR0 – Auxiliary Register 0

Register	SFR Address	Reset Value
AUXR0	A2H, Page:0	POR: 0000_0000b Software: 1UU0_0000b Reset pin: U1U0_0000b Hard fault: UU10_0000b Others: UUU0_0000b

7	6	5	4	3	2	1	0
SWRF	RSTPINF	HardF	HardFInt	GF2	-	0	DPS
R/W	R/W	R/W	R/W	R/W	-	R	R/W

Bit	Name	Description
3	GF2	General purpose flag 2 The general purpose flag that can be set or cleared by the user via software.
2	-	Reserved
1	0	Reserved This bit is always read as 0.
0	DPS	Data pointer select 0 = Data pointer 0 (DPTR) is active by default. 1 = Data pointer 1 (DPTR1) is active. After DPS switches the activated data pointer, the previous inactivated data pointer remains its original value unchanged.

32.2 96-Bit Unique Code

Before shipping out, each ML51 chip was factory pre-programmed with a 96-bit width serial number, which is guaranteed to be unique for each piece of ML51. The serial number is called Unique Code or UID. The user can read the Unique Code only by IAP command. More details please see Chapter 27 IN-APPLICATION-PROGRAMMING (IAP).

IAP Mode	IAPCN				IAPA[15:0] {IAPAH, IAPAL}	IAPFD[7:0]
	IAPB [1:0]	FOEN	FCEN	FCTRL [3:0]		
96-bit Unique Code read	XX	0	0	0100	0000H to 000BH	Data out

33 ON-CHIP-DEBUGGER (OCD)

33.1 Functional Description

The ML51 is embedded in an on-chip-debugger (OCD) providing developers with a low cost method for debugging user code, which is available on each package. The OCD gives debug capability of complete program flow control with eight hardware address breakpoints, single step, free running, and non-intrusive commands for memory access. The OCD system does not occupy any locations in the memory map and does not share any on-chip peripherals.

When the OCDEN (CONFIG0.4) is programmed as 0 and LOCK (CONFIG0.1) remains un-programmed as 1, the OCD is activated. The OCD cannot operate if chip is locked. The OCD system uses a two-wire serial interface, ICE_DAT and ICE_CLK, to establish communication between the target device and the controlling debugger host. ICE_DAT is an input/output pin for debug data transfer and ICE_CLK is an input pin for synchronization with ICE_DAT data. The nRESET pin is also necessary for OCD mode entry and exit. The ML51 supports OCD with Flash Memory control path by ICP writer mode, which shares the same three pins of OCD interface.

The ML51 uses ICE_DAT, ICE_CLK, and nRESET pins to interface with the OCD system. When designing a system where OCD will be used, the following restrictions must be considered for correct operation:

1 nRESET. cannot be connected directly to V_{DD} and any external capacitors connected must be removed.

2. All external reset sources must be disconnected.

3. Any external component connected on ICE_DAT and ICE_CLK must be isolated.

33.2 Limitation of OCD

The ML51 is a fully-featured microcontroller that multiplexes several functions on its limited I/O pins. Some device functionality must be sacrificed to provide resources for OCD system. The OCD has the following limitations:

1. The nRESET pin needs to be used for OCD mode selection.

2. The ICE_DAT pin is physically located on the same pin P5.0. Therefore, neither its I/O function nor shared multi-functions can be emulated.

3. The ICE_CLK pin is physically located on the same pin as P5.1. Therefore, neither its I/O function nor shared multi-functions can be emulated.

4. When the system is in Idle or Power-down mode, it is invalid to perform any accesses because parts of the device may not be clocked. A read access could return garbage or a write access might not succeed.

5. HIRC cannot be turned off because OCD uses this clock to monitor its internal status. The instruction that turns off HIRC affects nothing if executing under debug mode. When CPU enters its Power-down mode under debug mode, HIRC keeps turning on.

The ML51 OCD system has another limitation that non-intrusive commands cannot be executed at any time while the user's program is running. Non-intrusive commands allow a user to read or write MCU memory locations or access status and control registers with the debug controller. A reading or writing memory or control register space is allowed only when MCU is under halt condition after a matching of the hardware address breakpoint or a single step running.

CONFIG0

7	6	5	4	3	2	1	0
CBS	-	OCDPWM	OCDEN	-	-	LOCK	-

R/W	-	R/W	R/W	-	-	R/W	-

Factory default value: 1111 1111b

Bit	Name	Description
5	OCDPWM	PWM output state under OCD halt This bit decides the output state of PWM when OCD halts CPU. 1 = Tri-state pins those are used as PWM outputs. 0 = PWM continues.
4	OCDEN	OCD enable 1 = OCD Disabled. 0 = OCD Enabled. **Note:** If MCU run in OCD debug mode and OCDEN = 0, hard fault reset will disable. Only HardF flag be asserted.

Following is the OCD Relation multi function pin define list. As default when CONFIG define OCD enabled, P5.1 and P5.0 pin auto setting as ICE_CLK and ICE_DAT function.

Group	Pin Name	GPIO	MFP	Type	Description
ICE	ICE_CLK	P5.1	MFP14	I	Serial wired debugger clock pin.
	ICE_DAT	P5.0	MFP14	O	Serial wired debugger data pin.

34 IN-CIRCUIT-PROGRAMMING (ICP)

The Flash Memory can be programmed by “In-Circuit-Programming” (ICP). If the product is just under development or the end product needs firmware updating in the hand of an end customer, the hardware programming mode will make repeated programming difficult and inconvenient. ICP method makes it easy and possible without removing the microcontroller from the system. ICP mode also allows customers to manufacture circuit boards with un-programmed devices. Programming can be done after the assembly process allowing the device to be programmed with the most recent firmware or a customized firmware.

There are three signal pins, nRESET , ICE_DAT, and ICE_CLK, involved in ICP function. nRESET is used to enter or exit ICP mode. ICE_DAT is the data input and output pin. ICE_CLK is the clock input pin, which synchronizes the data shifted in to or out from MCU under programming. User should leave these three pins plus V_{DD} and GND pins on the circuit board to make ICP possible.

Nuvoton provides ICP tool for ML51, which enables user to easily perform ICP through Nuvoton ICP programmer. The ICP programmer developed by Nuvoton has been optimized according to the electric characteristics of MCU. It also satisfies the stability and efficiency during production progress. For more details, please visit Nuvoton 8-bit Microcontroller website: Nuvoton 80C51 Microcontroller Technical Support.

35 INSTRUCTION SET

The ML51 executes all the instructions of the standard 80C51 family fully compatible with MCS-51. However, the timing of each instruction is different for it uses high performance 1T 8051 core. The architecture eliminates redundant bus states and implements parallel execution of fetching, decode, and execution phases. The ML51 uses one clock per machine-cycle. It leads to performance improvement of rate 8.1 (in terms of MIPS) with respect to traditional 12T 80C51 device working at the same clock frequency. However, the real speed improvement seen in any system will depend on the instruction mix.

All instructions are coded within an 8-bit field called an OPCODE. This single byte should be fetched from Program Memory. The OPCODE is decoded by the CPU. It determines what action the CPU will take and whether more operation data is needed from memory. If no other data is needed, then only one byte was required. Thus the instruction is called a one byte instruction. In some cases, more data is needed, which is two or three byte instructions.

Following lists all instructions for details. The note of the instruction set and addressing modes are shown below.

Rn (N = 0~7)	Register R0 To R7 Of The Currently Selected Register Bank.
Direct	8-bit internal data location's address. It could be an internal data RAM location (00H to 7FH) or an SFR (80H to FFH).
@RI (I = 0, 1)	8-bit internal data RAM location (00H to FFH) addressed indirectly through register R0 or R1.
#data	8-bit constant included in the instruction.
#data16	16-bit constant included in the instruction.
Addr16	16-bit destination address. Used by LCALL and LJMP. A branch can be any-where within the Program Memory address space.
Addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2K-Byte page of Program Memory as the first byte of the following instruction.
Rel	Signed (2's complement) 8-bit offset Byte. Used by SJMP and all conditional branches. The range is -128 to +127 bytes relative to first byte of the following instruction.
Bit	Direct addressed bit in internal data RAM or SFR.

Table 33.2-1 Instruction Set

Instruction	OPCODE	Bytes	Clock Cycles	ML51 V.S. Tradition 80C51 Speed Ratio
NOP	00	1	1	12
ADD A, Rn	28~2F	1	2	6
ADD A, direct	25	2	3	4
ADD A, @Ri	26, 27	1	4	3
ADD A, #data	24	2	2	6
ADDC A, Rn	38~3F	1	2	6
ADDC A, direct	35	2	3	4
ADDC A, @Ri	36, 37	1	4	3
ADDC A, #data	34	2	2	6
SUBB A, Rn	98~9F	1	2	6
SUBB A, direct	95	2	3	4
SUBB A, @Ri	96, 97	1	4	3
SUBB A, #data	94	2	2	6
INC A	04	1	1	12
INC Rn	08~0F	1	3	4
INC direct	05	2	4	3
INC @Ri	06, 07	1	5	2.4
INC DPTR	A3	1	1	24
DEC A	14	1	1	12
DEC Rn	18~1F	1	3	4
DEC direct	15	2	4	3
DEC @Ri	16, 17	1	5	2.4
MUL AB	A4	1	4	12
DIV AB	84	1	4	12
DA A	D4	1	1	12
ANL A, Rn	58~5F	1	2	6
ANL A, direct	55	2	3	4
ANL A, @Ri	56, 57	1	4	3

Instruction	OPCODE	Bytes	Clock Cycles	ML51 V.S. Tradition 80C51 Speed Ratio
ANL A, #data	54	2	2	6
ANL direct, A	52	2	4	3
ANL direct, #data	53	3	4	6
ORL A, Rn	48~4F	1	2	6
ORL A, direct	45	2	3	4
ORL A, @Ri	46, 47	1	4	3
ORL A, #data	44	2	2	6
ORL direct, A	42	2	4	3
ORL direct, #data	43	3	4	6
XRL A, Rn	68~6F	1	2	6
XRL A, direct	65	2	3	4
XRL A, @Ri	66, 67	1	4	3
XRL A, #data	64	2	2	6
XRL direct, A	62	2	4	3
XRL direct, #data	63	3	4	6
CLR A	E4	1	1	12
CPL A	F4	1	1	12
RL A	23	1	1	12
RLC A	33	1	1	12
RR A	03	1	1	12
RRC A	13	1	1	12
SWAP A	C4	1	1	12
MOV A, Rn	E8~EF	1	1	12
MOV A, direct	E5	2	3	4
MOV A, @Ri	E6, E7	1	4	3
MOV A, #data	74	2	2	6
MOV Rn, A	F8~FF	1	1	12
MOV Rn, direct	A8~AF	2	4	6
MOV Rn, #data	78~7F	2	2	6

Instruction	OPCODE	Bytes	Clock Cycles	ML51 V.S. Tradition 80C51 Speed Ratio
MOV direct, A	F5	2	2	6
MOV direct, Rn	88~8F	2	3	8
MOV direct, direct	85	3	4	6
MOV direct, @Ri	86, 87	2	5	4.8
MOV direct, #data	75	3	3	8
MOV @Ri, A	F6, F7	1	3	4
MOV @Ri, direct	A6, A7	2	4	6
MOV @Ri, #data	76, 77	2	3	6
MOV DPTR, #data16	90	3	3	8
MOVC A, @A+DPTR	93	1	4	6
MOVC A, @A+PC	83	1	4	6
MOVX A, @Ri[1]	E2, E3	1	5	4.8
MOVX A, @DPTR[1]	E0	1	4	6
MOVX @Ri, A[1]	F2, F3	1	6	4
MOVX @DPTR, A[1]	F0	1	5	4.8
PUSH direct	C0	2	4	6
POP direct	D0	2	3	8
XCH A, Rn	C8~CF	1	2	6
XCH A, direct	C5	2	3	4
XCH A, @Ri	C6, C7	1	4	3
XCHD A, @Ri	D6, D7	1	5	2.4
CLR C	C3	1	1	12
CLR bit	C2	2	4	3
SETB C	D3	1	1	12
SETB bit	D2	2	4	3
CPL C	B3	1	1	12
CPL bit	B2	2	4	3
ANL C, bit	82	2	3	8
ANL C, /bit	B0	2	3	8

Instruction	OPCODE	Bytes	Clock Cycles	ML51 V.S. Tradition 80C51 Speed Ratio
ORL C, bit	72	2	3	8
ORL C, /bit	A0	2	3	8
MOV C, bit	A2	2	3	4
MOV bit, C	92	2	4	6
ACALL addr11	11, 31, 51, 71, 91, B1, D1, F1[2]	2	4	6
LCALL addr16	12	3	4	6
RET	22	1	5	4.8
RETI	32	1	5	4.8
AJMP addr11	01, 21, 41, 61, 81, A1, C1, E1[3]	2	3	8
LJMP addr16	02	3	4	6
SJMP rel	80	2	3	8
JMP @A+DPTR	73	1	3	8
JZ rel	60	2	3	8
JNZ rel	70	2	3	8
JC rel	40	2	3	8
JNC rel	50	2	3	8
JB bit, rel	20	3	5	4.8
JNB bit, rel	30	3	5	4.8
JBC bit, rel	10	3	5	4.8
CJNE A, direct, rel	B5	3	5	4.8
CJNE A, #data, rel	B4	3	4	6
CJNE Rn, #data, rel	B8~BF	3	4	6
CJNE @Ri, #data, rel	B6, B7	3	6	4
DJNZ Rn, rel	D8~DF	2	4	6
DJNZ direct, rel	D5	3	5	4.8

[1] The ML51 does not have external memory bus. MOVX instructions are used to access internal XRAM.
[2] The most three significant bits in the 11-bit address [A10:A8] decide the ACALL hex code. The code will be [A10,A9,A8,1,0,0,0,1].
[3] The most three significant bits in the 11-bit address [A10:A8] decide the AJMP hex code. The code will be [A10,A9,A8,0,0,0,0,1].

36 CONFIG BYTES

The ML51 has several hardware configuration bytes, called CONFIG, those are used to configure the hardware options such as the security bits, system clock source, and so on. These hardware options can be re-configured through the parallel Writer, In-Circuit-Programming (ICP), or In-Application-Programming (IAP). Several functions, which are defined by certain CONFIG bits are also available to be re-configured by SFR. Therefore, there is a need to load such CONFIG bits into respective SFR bits. Such loading will occur after resets. These SFR bits can be continuously controlled via user's software.

CONFIG bits marked as "-"should always keep un-programmed.

CONFIG0

7	6	5	4	3	2	1	0
CBS	-	OCDPWM	OCDEN	-	-	LOCK	-
R/W	-	R/W	R/W	-	-	R/W	-

Factory default value: 1111 1111b

Bit	Name	Description
7	CBS	CONFIG boot select This bit defines from which block that MCU re-boots after resets except software reset. 1 = MCU will re-boot from APROM after resets except software reset. 0 = MCU will re-boot from LDROM after resets except software reset.
5	OCDPWM	PWM output state under OCD halt This bit decides the output state of PWM when OCD halts CPU. 1 = Tri-state pins those are used as PWM outputs. 0 = PWM continues.
4	OCDEN	OCD enable 1 = OCD Disabled. 0 = OCD Enabled. **Note:** If MCU run in OCD debug mode and OCDEN = 0, hard fault reset will disable. Only HFIF flag be asserted.
3:2	-	Reserved
1	LOCK	Chip lock enable 1 = Chip is unlocked. Flash Memory is not locked. Their contents can be read out through a parallel Writer/ICP programmer. 0 = Chip is locked. Whole Flash Memory is locked. Their contents read through a parallel Writer or ICP programmer will be all blank (FFH). Programming to Flash Memory is invalid. Note that CONFIG bytes are always unlocked and can be read. Hence, once the chip is locked, the CONFIG bytes cannot be erased or programmed individually. The only way to disable chip lock is execute "whole chip erase". However, all data within the Flash Memory and CONFIG bits will be erased when this procedure is executed. If the chip is locked, it does not alter the IAP function.

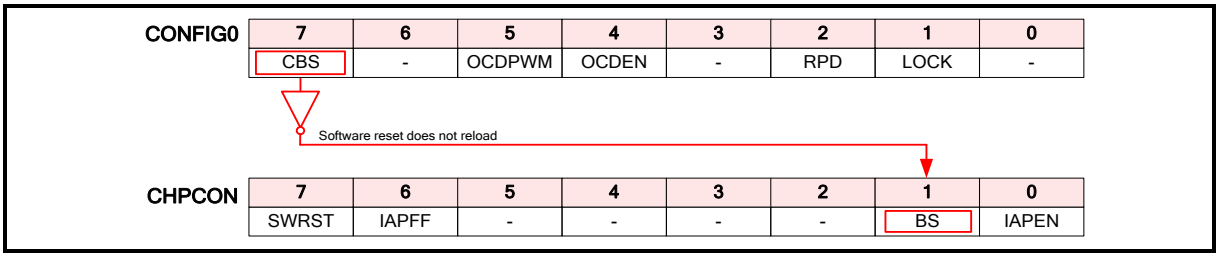


Figure 33.2-1 CONFIG0 Any Reset Reloading

CONFIG1

7	6	5	4	3	2	1	0
-	-	-	-	-	LDSIZE[2:0]		
-	-	-	-	-	R/W		

Factory default value: 1111 1111b

Bit	Name	Description
2:0	LDSIZE[2:0]	LDROM size select Flash size is 64KB: 111 = No LDROM. APROM is 64 Kbytes. 110 = LDROM is 1 Kbytes. APROM is 63 Kbytes. 101 = LDROM is 2 Kbytes. APROM is 62 Kbytes. 100 = LDROM is 3 Kbytes. APROM is 61 Kbytes. 0xx = LDROM is 4 Kbytes. APROM is 60 Kbytes. Flash size is 32KB: 111 = No LDROM. APROM is 32 Kbytes. 110 = LDROM is 1 Kbytes. APROM is 31 Kbytes. 101 = LDROM is 2 Kbytes. APROM is 30 Kbytes. 100 = LDROM is 3 Kbytes. APROM is 29 Kbytes. 0xx = LDROM is 4 Kbytes. APROM is 28 Kbytes. Flash size is 16KB: 111 = No LDROM. APROM is 16 Kbytes. 110 = LDROM is 1 Kbytes. APROM is 15 Kbytes. 101 = LDROM is 2 Kbytes. APROM is 14 Kbytes. 100 = LDROM is 3 Kbytes. APROM is 13 Kbytes. 0xx = LDROM is 4 Kbytes. APROM is 12 Kbytes.

CONFIG2

7	6	5	4	3	2	1	0
CBODEN	CBOV[2:0]			BOIAP	CBORST	-	-
R/W	R/W			R/W	R/W	-	-

Factory default value: 1111 1111b

Bit	Name	Description
7	CBODEN	CONFIG brown-out detect enable 1 = Brown-out detection circuit on. 0 = Brown-out detection circuit off.
6:4	CBOV[2:0]	CONFIG brown-out voltage select 111 = V_{BOD} is 1.8V. 110 = V_{BOD} is 1.8V. 101 = V_{BOD} is 2.0V. 100 = V_{BOD} is 2.4V. 011 = V_{BOD} is 2.7V. 010 = V_{BOD} is 3.0V. 001 = V_{BOD} is 3.7V. 000 = V_{BOD} is 4.4V.
3	BOIAP	Brown-out inhibiting IAP This bit decides whether IAP erasing or programming is inhibited by brown-out status. This bit is valid only when brown-out detection is enabled. 1 = IAP erasing or programming is inhibited if V_{DD} is lower than V_{BOD}. 0 = IAP erasing or programming is allowed under any workable V_{DD}.
2	CBORST	CONFIG brown-out reset enable This bit decides whether a brown-out reset is caused by a power drop below V_{BOD}. 1 = Brown-out reset Enabled. 0 = Brown-out reset Disabled.

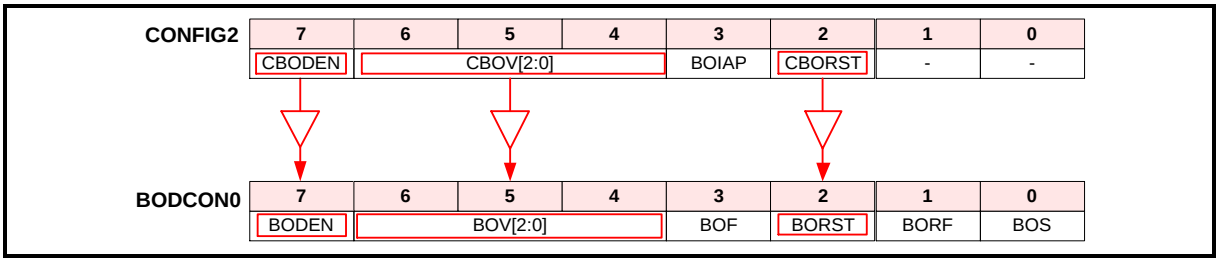


Figure 33.2-2 CONFIG2 Power-On Reset Reloading

CONFIG4

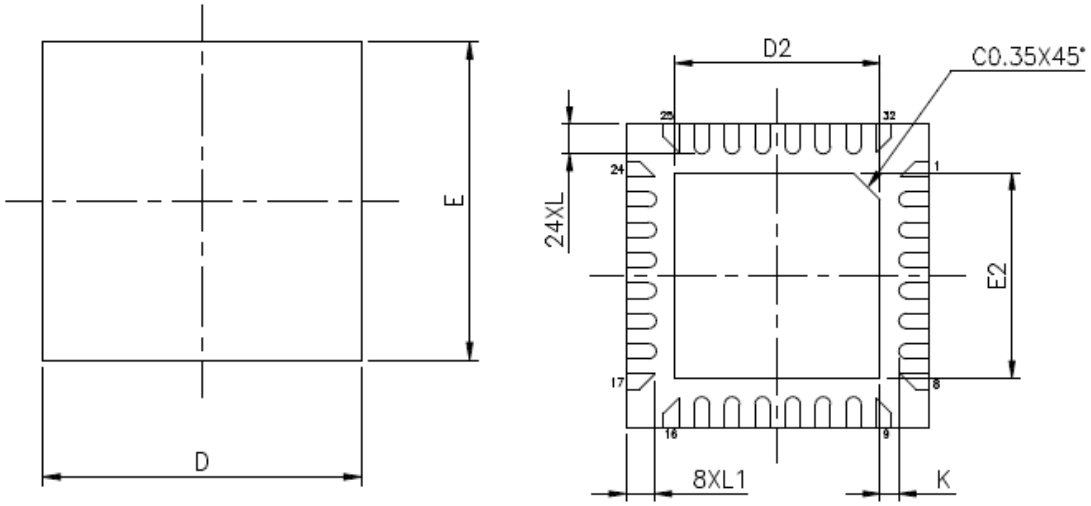
7	6	5	4	3	2	1	0
WDTEN[3:0]				-	-	-	-
R/W				-	-	-	-

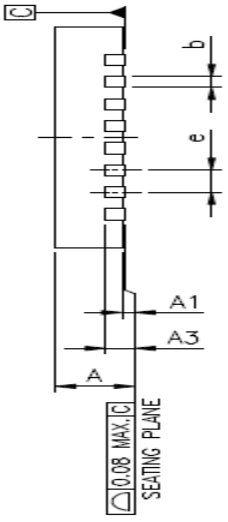
Factory default value: 1111 1111b

Bit	Name	Description
7:4	WDTEN[3:0]	WDT enable This field configures the WDT behavior after MCU execution. 1111 = WDT is Disabled. WDT can be used as a general purpose timer via software control. 0101 = WDT is Enabled as a time-out reset timer and it stops running during Idle or Power-down mode. Others = WDT is Enabled as a time-out reset timer and it keeps running during Idle or Power-down mode.
3:0	-	Reserved

37 PACKAGE DIMENSIONS

37.1 QFN 33-pin (4.0 x 4.0 x 0.8 mm)



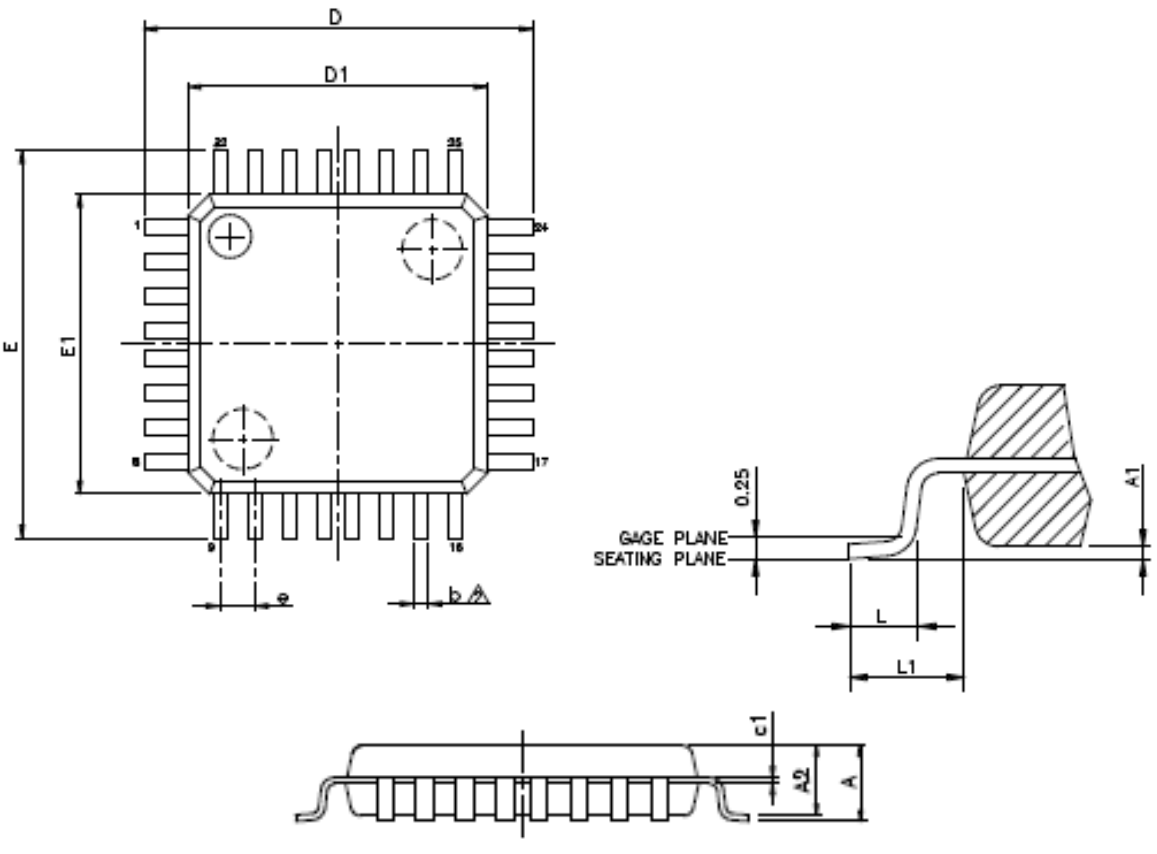


	Symbol	Min	Nom	Max
Total Thickness	A	0.7	0.75	0.8
Stand Off	A1	0.0	0.02	0.05
L/F Thickness	A3		0.2 REF.	
Lead Width	b	0.15	0.20	0.25
Body Size	D		4.00 BSC	
	E		4.00 BSC	
Lead Pitch	e		0.40 BSC	
Lead Length	L	0.35	0.4	0.45
	K	0.2	--	--

Symbol	L/F SIZE	Dimension in mm		
		MIN.	NOR.	MAX.
D2	114X114 mm	2.6	2.7	2.75
E2		2.6	2.7	2.75

Figure 37.1-1 QFN-33 Package Dimension

37.2 LQFP 32-pin (7.0 x 7.0 x 1.4 mm)

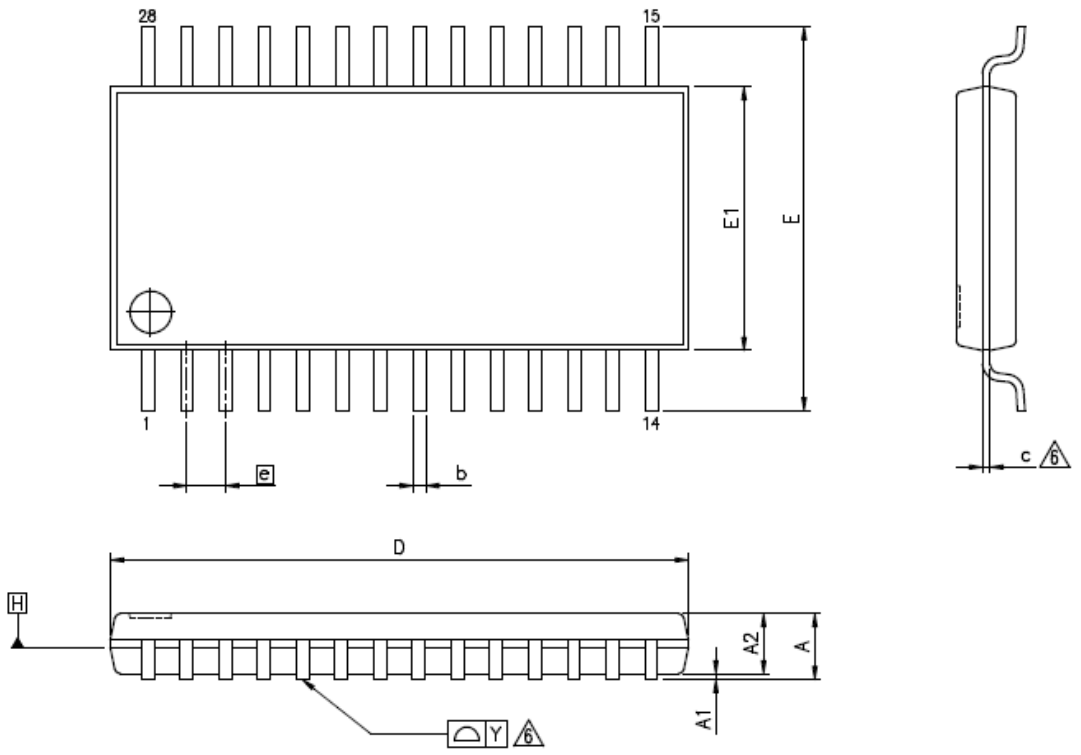


VARIATIONS (ALL DIMENSIONS SHOWN IN MM)

SYMBOLS	MIN.	MAX.
A	--	1.6
A1	0.05	0.15
A2	1.35	1.45
c1	0.09	0.16
D	9.00 BSC	
D1	7.00 BSC	
E	9.00 BSC	
E1	7.00 BSC	
e	0.8 BSC	
b	0.30	0.45
L	0.45	0.75
L1	1 REF	

Figure 37.2-1 LQFP-32 Package Dimension

37.3 TSSOP 28-pin (4.4 x 9.7 x 1.0 mm)

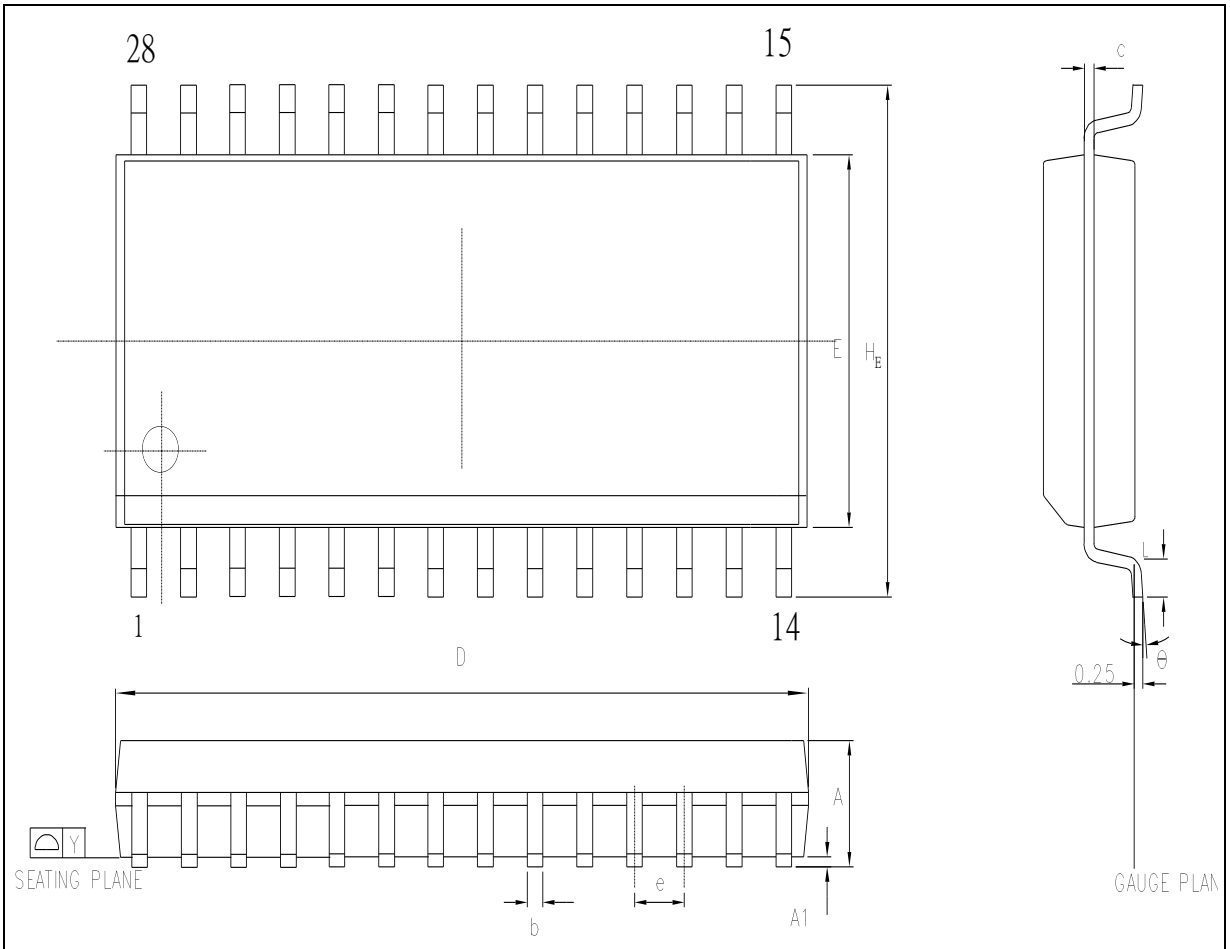


VARIATIONS (ALL DIMENSIONS SHOWN IN MM)

SYMBOLS	MIN.	NOM.	MAX.
A	–	–	1.20
A1	0.00	–	0.15
A2	0.80	1.00	1.05
b	0.19	–	0.30
c	0.09	–	0.20
D	9.60	9.70	9.80
E1	4.30	4.40	4.50
E	6.40 BSC		
e	0.65 BSC		
L1	1.00 REF		
L	0.45	0.60	0.75
S	0.20	–	–
θ	0°	–	8°
Y	0.10		

Figure 37.3-1 TSSOP-28 Package Dimension

37.4 SOP 28-pin (300mil)

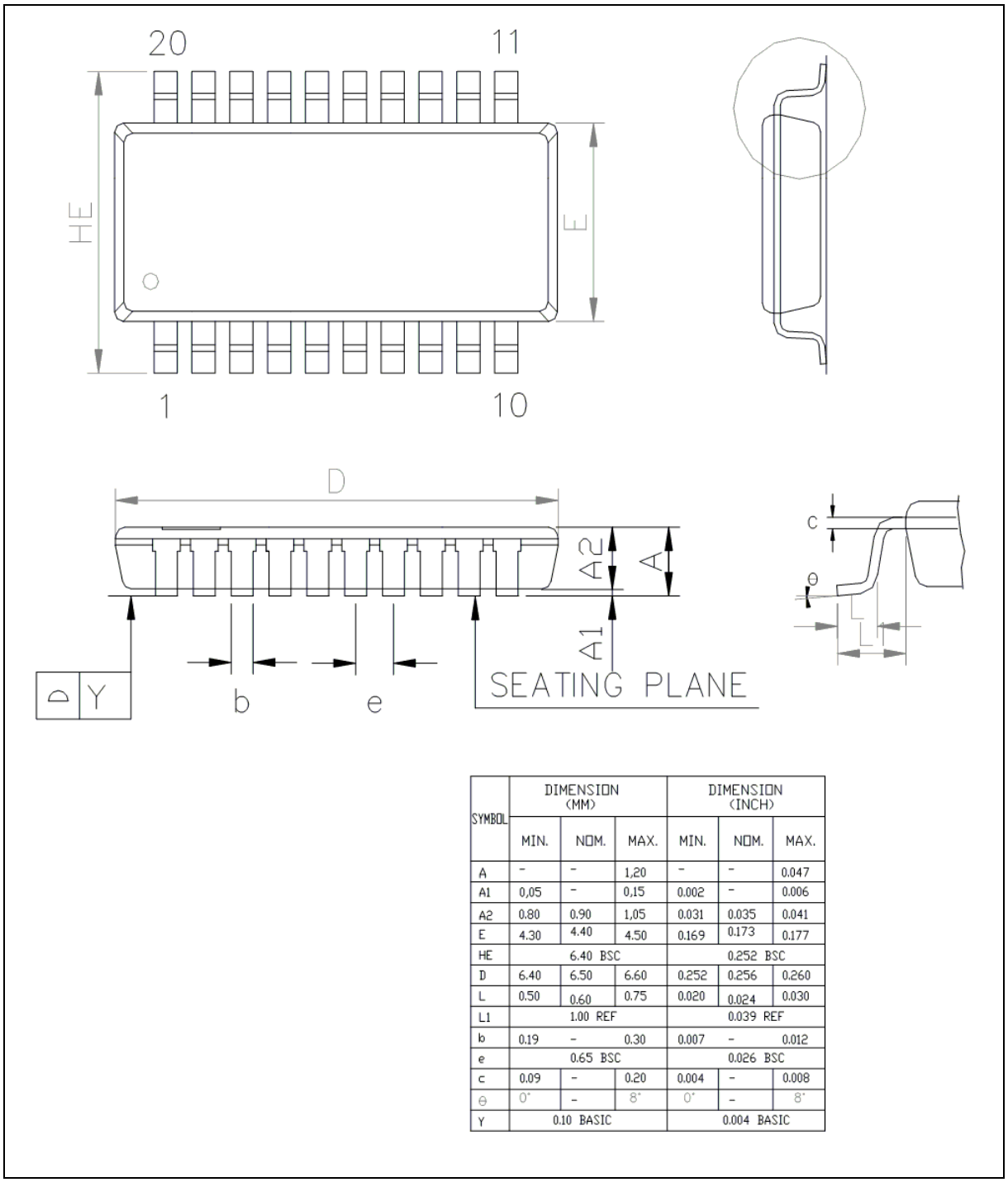


Control demensions are in milmeters .

SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN.	MAX.	MIN.	MAX.
A	2.35	2.65	0.093	0.104
A1	0.10	0.30	0.004	0.012
b	0.33	0.51	0.013	0.020
c	0.23	0.32	0.009	0.013
E	7.40	7.60	0.291	0.299
D	17.70	18.10	0.697	0.713
e	1.27 BSC		0.050 BSC	
H_E	10.00	10.65	0.394	0.419
Y	—	0.10	—	0.004
L	0.40	1.27	0.016	0.050
θ	0	8	0	8

Figure 37.4-1 SOP-28 Package Dimension

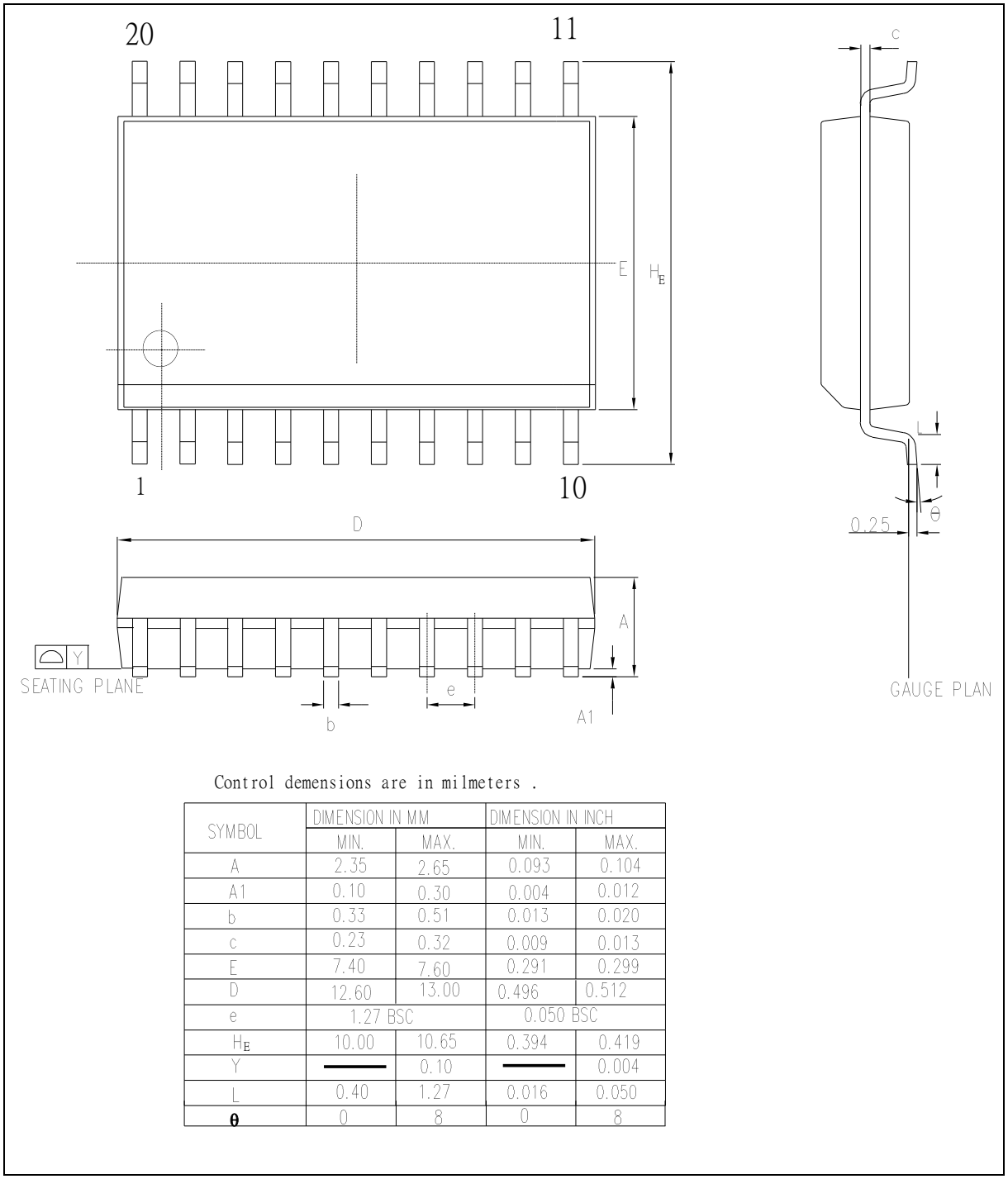
37.5 TSSOP 20-pin (4.4 x 6.5 x 0.9 mm)



SYMBOL	DIMENSION (MM)			DIMENSION (INCH)		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	-	-	1.20	-	-	0.047
A1	0.05	-	0.15	0.002	-	0.006
A2	0.80	0.90	1.05	0.031	0.035	0.041
E	4.30	4.40	4.50	0.169	0.173	0.177
HE		6.40 BSC			0.252 BSC	
D	6.40	6.50	6.60	0.252	0.256	0.260
L	0.50	0.60	0.75	0.020	0.024	0.030
L1		1.00 REF			0.039 REF	
b	0.19	-	0.30	0.007	-	0.012
e		0.65 BSC			0.026 BSC	
c	0.09	-	0.20	0.004	-	0.008
θ	0°	-	8°	0°	-	8°
Y		0.10 BASIC			0.004 BASIC	

Figure 37.5-1 TSSOP-20 Package Dimension

37.6 SOP 20-pin (300 mil)

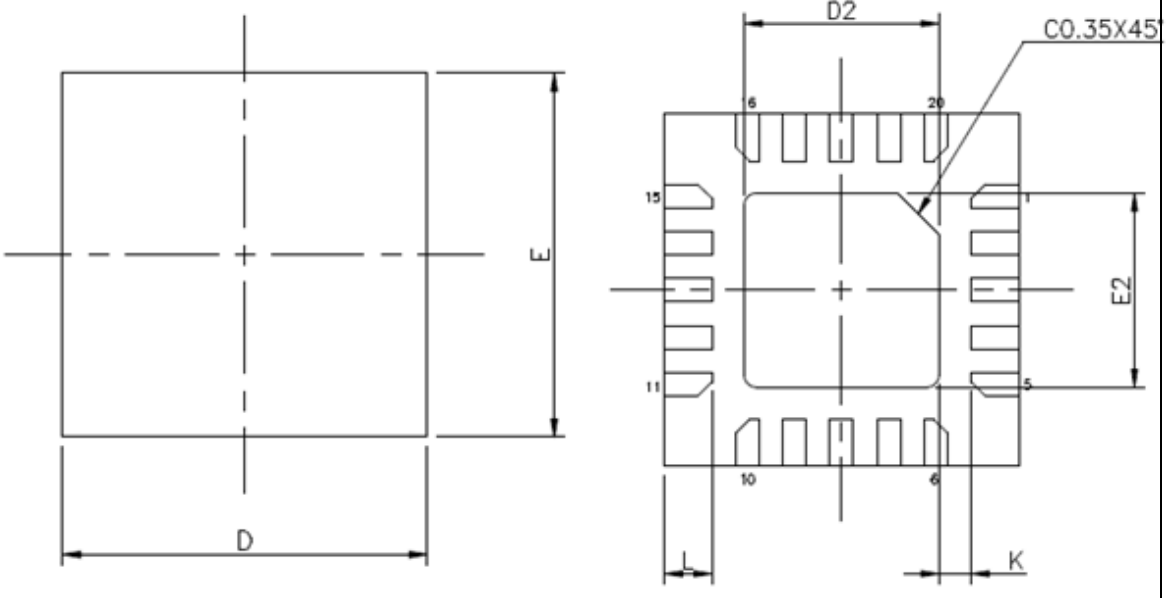


Control demensions are in milmeters .

SYMBOL	DIMENSION IN MM		DIMENSION IN INCH	
	MIN.	MAX.	MIN.	MAX.
A	2.35	2.65	0.093	0.104
A1	0.10	0.30	0.004	0.012
b	0.33	0.51	0.013	0.020
c	0.23	0.32	0.009	0.013
E	7.40	7.60	0.291	0.299
D	12.60	13.00	0.496	0.512
e	1.27 BSC		0.050 BSC	
H_E	10.00	10.65	0.394	0.419
Y	—	0.10	—	0.004
L	0.40	1.27	0.016	0.050
θ	0	8	0	8

Figure 37.6-1 SOP-20 Package Dimension

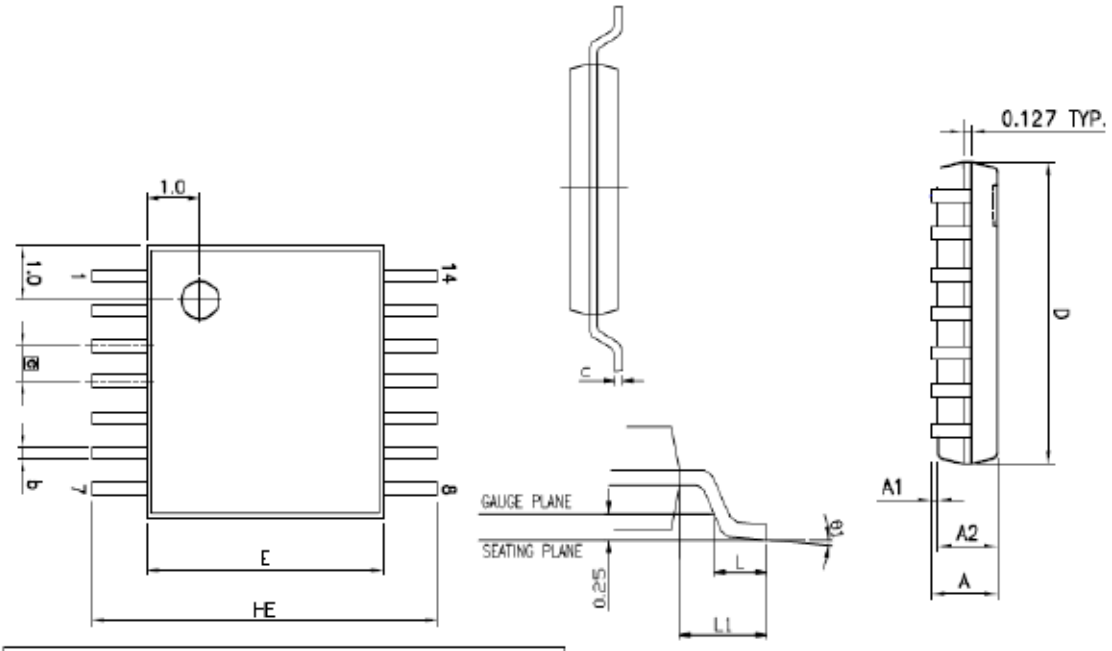
37.7 QFN 20-pin (3.0 x 3.0 x 0.8 mm)



SYMBOLS	MIN.	NOM.	MAX.
A	0.70	0.75	0.80
A1	0.00	0.02	0.05
A3		0.203 REF.	
b	0.15	0.20	0.25
D		3.00 BSC	
E		3.00 BSC	
e		0.40 BSC	
K	0.20	—	—
E2	1.60	1.65	1.70
D2	1.60	1.65	1.70
L	0.30	0.40	0.50

Figure 37.7-1 QFN-20 Package Dimension

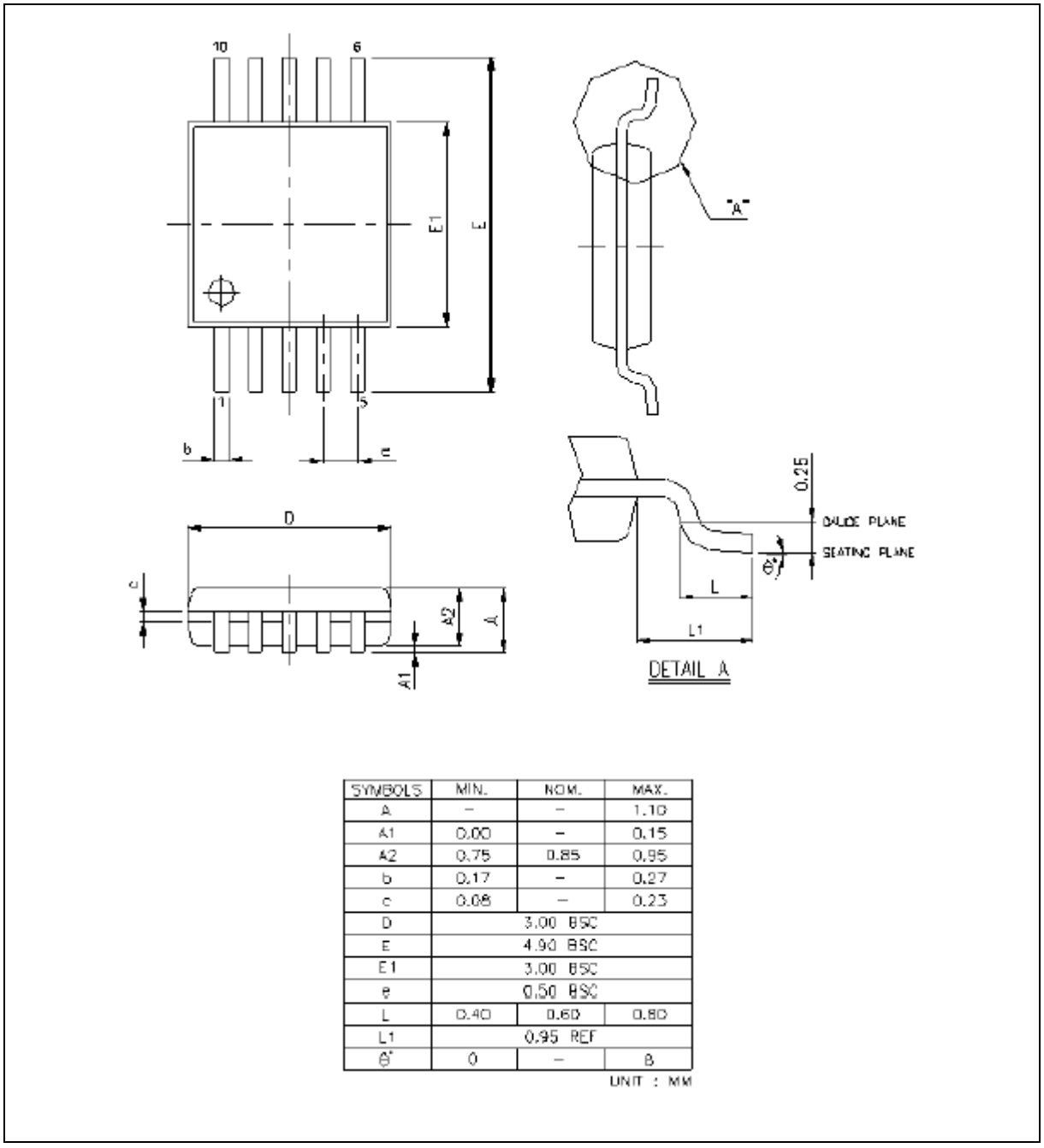
37.8 TSSOP 14-pin (4.4 x 5.0 x 0.9 mm)



CONTROLLING DIMENSION: MILLMETERS						
SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN.	NOM	MAX.	MIN.	NOM	MAX.
A	---	---	1.20	---	---	0.043
A1	0.05	---	0.15	0.02	---	0.006
A2	0.80	0.90	1.05	0.031	0.035	0.041
L	0.50	0.60	0.75	0.020	0.024	0.030
HE	6.40 BSC.			0.252 BSC.		
E	4.30	4.40	4.50	0.169	0.173	0.177
D	4.90	5.00	5.10	0.193	0.197	0.201
b	0.19	---	0.30	0.007	---	0.012
c	0.09	---	0.20	0.004	---	0.008
L1	1.0 REF.			0.039 REF.		
e	0.65 BSC.			0.026 BSC.		
θ1	0	---	8	0	---	8

Figure 37.8-1 TSSOP-14 Package Dimension

37.9 MSOP 10-pin (3.0 x 3.0 x 0.85 mm)



SYMBOLS	MIN.	NOM.	MAX.
A	–	–	1.10
A1	0.00	–	0.15
A2	0.75	0.85	0.95
b	0.17	–	0.27
c	0.08	–	0.23
D	3.00 BSC		
E	4.90 BSC		
E1	3.00 BSC		
e	0.50 BSC		
L	0.40	0.60	0.80
L1	0.95 REF		
θ°	0	–	8

UNIT : MM

Figure 37.9-1 MSOP-10 Package Dimension

38 REVISION HISTORY

Date	Revision	Description
2018.12.05	1.00	Initial release.
2019.3.18	1.01	Section 3.1 Added package type table. Section 4.2.2 Added Multi-function summary table Section 7.2 Added description that all about PWM1 register is only for 64K flash body product. Section 24.3 Added PDMA support in different part number. Section 37.6 Modified TSSOP20 package value.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.