



十速

# **TM52F5024**

## ***DATA SHEET***

### ***Rev 0.90***

**tenx** reserves the right to change or discontinue the manual and online documentation to this product herein to improve reliability, function or design without further notice. **tenx** does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. **tenx** products are not designed, intended, or authorized for use in life support appliances, devices, or systems. If Buyer purchases or uses **tenx** products for any such unintended or unauthorized application, Buyer shall indemnify and hold **tenx** and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that **tenx** was negligent regarding the design or manufacture of the part.

---

## AMENDMENT HISTORY

Version	Date	Description
V0.83	Jan, 2024	New release.
V0.84	Mar, 2024	Modify CRC16 storage location description Remove EAh.7 (MIIE)
V0.90	Apr, 2024	Modify EEPROM characteristic Modify Power On description Modify Pin Assignment/Ordering information

# CONTENTS

<b>AMENDMENT HISTORY</b> .....	<b>2</b>
<b>GENERAL DESCRIPTION</b> .....	<b>6</b>
<b>BLOCK DIAGRAM</b> .....	<b>6</b>
<b>FEATURES</b> .....	<b>7</b>
<b>PIN ASSIGNMENT</b> .....	<b>10</b>
<b>PIN DESCRIPTION</b> .....	<b>12</b>
<b>PIN SUMMERY</b> .....	<b>13</b>
<b>FUNCTIONAL DESCRIPTION</b> .....	<b>14</b>
1. CPU Core .....	14
1.1 Accumulator (ACC).....	14
1.2 B Register (B).....	14
1.3 Stack Pointer (SP).....	15
1.4 Dual Data Pointer (DPTRs).....	15
1.5 Program Status Word (PSW).....	16
2. Memory .....	18
2.1 Program Memory.....	19
2.1.1 Program Memory Functional Partition .....	19
2.1.2 Flash ICP Mode.....	20
2.1.3 Flash IAP Mode .....	20
2.1.4 IAP Mode Access Routines .....	20
2.1.5 Flash ISP Mode .....	22
2.2 Information Memory .....	23
2.3 EEPROM Memory .....	25
2.4 Data Memory .....	28
2.4.1 IRAM .....	28
2.4.2 XRAM.....	28
2.4.3 SFRs .....	28
3. LVR and LVD setting .....	30
4. Reset.....	33
4.1 Power on Reset.....	33
4.2 External Pin Reset .....	33
4.3 Software Command Reset.....	33
4.4 Watchdog Timer Reset.....	33
4.5 Low Voltage Reset .....	33

5. Clock Circuitry & Operation Mode .....	36
5.1 System Clock .....	36
5.2 Operation Modes .....	38
6. Interrupt & Wake-up .....	40
6.1 Interrupt Enable and Priority Control .....	41
6.2 Suggestions on interrupting subroutines.....	41
6.3 Pin Interrupt and LVD interrupt .....	44
6.4 Idle mode Wake up and Interrupt .....	48
6.5 Halt/Stop mode Wake up and Interrupt.....	48
7. I/O Ports .....	50
7.1 Port0~Port 3.....	50
8. Timers.....	57
8.1 Timer0 / Timer1.....	57
8.2 Timer2.....	60
8.3 Timer3.....	62
9. UARTs .....	63
10. PWMs.....	66
11. ADC .....	79
11.1 ADC Channels .....	80
11.2 ADC Conversion Time .....	80
12. S/W Controller LCD Driver .....	84
13. Cyclic Redundancy Check (CRC).....	86
14. Multiplier and divider.....	87
15. Master I <sup>2</sup> C Interface .....	89
16. In Circuit Emulation (ICE) Mode .....	93
<b>SFR &amp; CFGW MAP .....</b>	<b>94</b>
<b>SFR &amp; CFGW DESCRICPTION.....</b>	<b>97</b>
<b>INSTRUCTION SET .....</b>	<b>110</b>
<b>ELECTRICAL CHARACTERISTICS .....</b>	<b>113</b>
1. Absolute Maximum Ratings.....	113
2. DC Characteristics.....	113
3. Clock Timing.....	116
4. Reset Timing Characteristics .....	116
5. ADC Electrical Characteristics .....	116
6. EEPROM Characteristics.....	116



7. Characteristic Graphs ..... 117

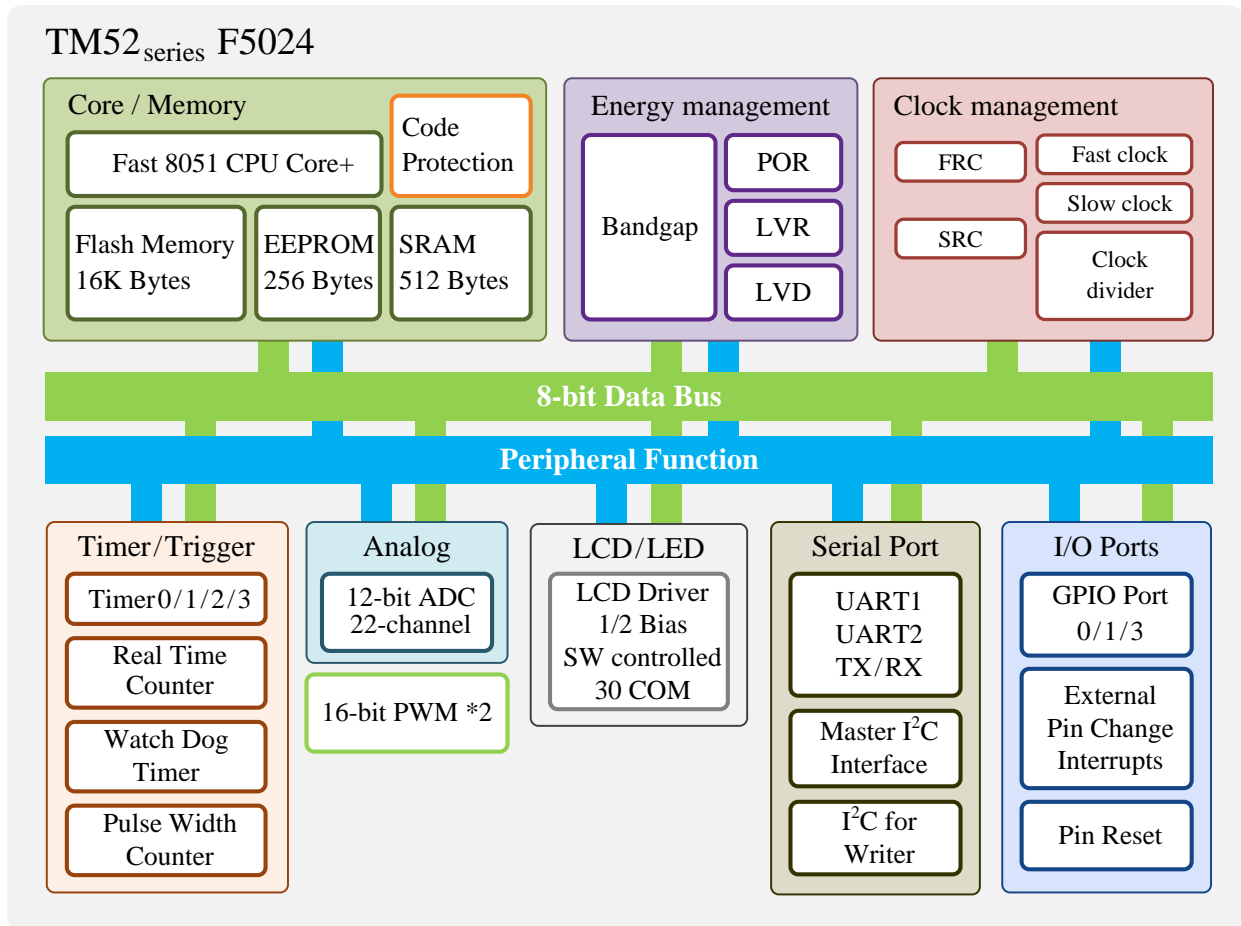
**Package and Dice Information..... 120**

## GENERAL DESCRIPTION

TM52<sub>series</sub> F5024 are versions of a new, fast 8051 architecture for an 8-bit microcontroller single chip with an instruction set fully compatible with industry standard 8051, and retains most 8051 peripheral's functional block. Typically, the TM52 executes instructions six times faster than the standard 8051 architecture.

The TM52-F5024 provides improved performance, lower cost and fast time-to-market by integrating features on the chip, including 16K Bytes Flash program memory, 512 Bytes SRAM, Low Voltage Reset (LVR), Low Voltage Detector (LVD), dual clock power saving operation mode, 8051 standard UART and Timer0/1/2, real time clock Timer3, LCD driver, 2 set 16-bit PWMs, 22 channels 12-bit A/D Convertor, I<sup>2</sup>C interface and Watch Dog Timer. It's a high reliability and low power consumption feature can be widely applied in consumer and home appliance products.

## BLOCK DIAGRAM



## FEATURES

- 1. Standard 8051 Instruction set, fast machine cycle**
  - Executes instructions six times faster than the standard 8051.
- 2. Flash Program Memory**
  - 16K Bytes (TM52F5024)
  - Support “In Circuit Programming” (ICP) or “In System Programming” (ISP) for the Flash code
  - Byte Write “In Application Programming” (IAP) mode is convenient as Data EEPROM access
  - Code Protection Capability
  - 100 erase times at least
  - 10 years data retention at least
- 3. 256 Bytes EEPROM Memory**
  - 50K erase times at least
  - 10 years data retention at least
- 4. Total 512 Bytes SRAM (IRAM + XRAM)**
  - 256 Bytes IRAM in the 8051 internal data memory area
  - 256 Bytes XRAM in the 8051 external data memory area (accessed by MOVX Instruction)
- 5. Two System Clock type selections**
  - Fast clock from Internal RC (FRC, 18.432 MHz)
  - Slow clock from Internal RC (SRC, 75 KHz)
  - System Clock can be divided by 1/2/4/16 option
- 6. 8051 Standard Timer – Timer0/1/2**
  - 16-bit Timer0, also supports T0O clock output for Buzzer application
  - 16-bit Timer1
  - 16-bit Timer2, also supports T2O clock output for Buzzer application
- 7. 15-bit Timer3**
  - Clock source is Slow clock or FRC/512
  - Interrupt period can be clock divided by  
262144/131072/65536/32768/16384/8192/4096/2048/1024/512/256/128/64/32/16/8 option
- 8. UARTs**
  - UART1, 8051 standard UART
  - UART2, the second UART
  - With UART pin select option

**9. Two independent 16 bits PWMs with period-adjustment**

- With PWM0/PWM1 Interrupt

**10. I<sup>2</sup>C interface (Master)****11. 12-bit ADC with 22 channels External Pin Input and 2 channels Internal Reference Voltage**

- Internal Reference Voltage:  $V_{BG} 2.5V \pm 1\%$  @  $V_{CC}=5V \sim 2.5V, 25^{\circ}C$
- Internal Reference Voltage:  $1/4V_{CC}, V_{CC}/201$
- ADC reference voltage = 1.18/2.0/2.5/4V

**12. LCD Driver**

- Software controlled COM00~05, COM10~17, COM30~37 (Max. 22 pins)
- 1/2 LCD Bias

**13. 14 Sources, 4-level priority Interrupt**

- Timer0/Timer1/Timer2/Timer3 Interrupt
- INT0/INT1 pin Falling-Edge/Low-Level Interrupt
- All Pin Change Wake up Interrupt from Halt/Stop mode
- UART1/UART2 TX/RX Interrupt
- LVD Interrupt
- ADC Interrupt
- I<sup>2</sup>C Interrupt
- EEP write Finish Interrupt
- PWM0/PWM1 Interrupt

**14. Pin Interrupt can Wake up CPU from Halt/Stop mode**

- P3.2/P3.3 (INT0/INT1) Interrupt & Wake up
- Each pin can be defined as Wake up interrupt pin (by pin change)

**15. Max. 22 Programmable I/O pins**

- CMOS Output
- Pseudo-Open-Drain, or Open-Drain Output
- Schmitt Trigger Input
- Pin Pull-up / Pull-down can be Enabled or Disabled
- All pin with high sink ( $70mA @ V_{CC}=5V, V_{OL}=0.1V_{CC}$ )

**16. Independent RC Oscillating Watch Dog Timer**

- 220ms/110ms/54ms/27ms selectable WDT timeout options



**17. Five types Reset**

- Power on Reset
- Selectable External Pin Reset
- Selectable Watch Dog Reset
- Software Command Reset
- Selectable Low Voltage Reset

**18. 16-level Low Voltage Detect**

- 4.12V/3.94V/3.0V/3.63V/3.50V/3.32V/3.18V/3.03V/  
2.87V/2.71V/2.56V/2.40V/2.26V/2.11V/1.95V/1.79V

**19. 16-level Low Voltage Reset**

- 4.12V/3.94V/3.0V/3.63V/3.50V/3.32V/3.18V/3.03V/  
2.87V/2.71V/2.56V/2.40V/2.26V/2.11V/1.95V/1.79V

**20. Five Power Operation Modes**

- Fast/ Slow/ Idle/ Halt/ Stop mode

**21. Integrated 16-bit Cyclic Redundancy Check function****22. Multiplication and division**

- 8-bit Multiplier & Divider (standard 8051)
- 16-bit Multiplier & Divider
- 32-bit ÷ 16-bit Divider

**23. On-chip Debug/ICE interface**

- Use P3.0/P3.1 pin or P0.0/P0.1 pin
- Share with ICP programming pin

**24. Operating Voltage and Current**

- $V_{CC} = 2.4V \sim 5.5V$  @ $F_{SYSCLK}=18.432MHz$  ( $-40^{\circ}C \sim +105^{\circ}C$ )
- $I_{CC} = 0.1\mu A$  @Stop mode,  $PWRS\Delta V=1$ ,  $V_{CC}=3V$
- $I_{CC} = 3\mu A$  @Halt mode,  $PWRS\Delta V=1$ ,  $V_{CC}=3V$
- $I_{CC} = 5\mu A$  @Idle mode,  $PWRS\Delta V=1$ ,  $V_{CC}=3V$

**25. Operating Temperature Range**

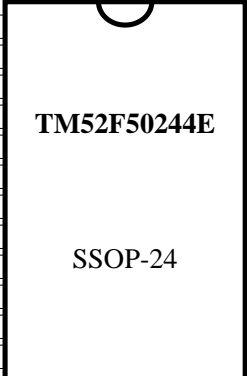
- $-40^{\circ}C \sim +105^{\circ}C$

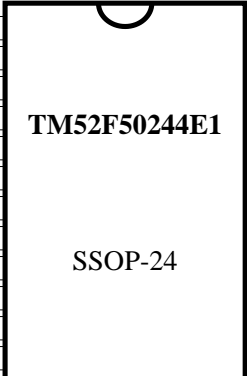
**26. Package Types**

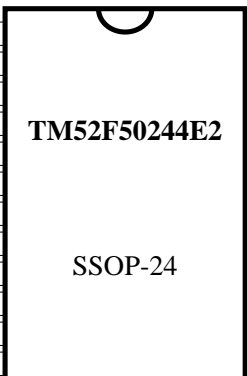
- 24-pin SSOP (150 mil)
- 20-pin SOP (300 mil)
- 20-pin TSSOP (173 mil)
- 16-pin SOP (150 mil)

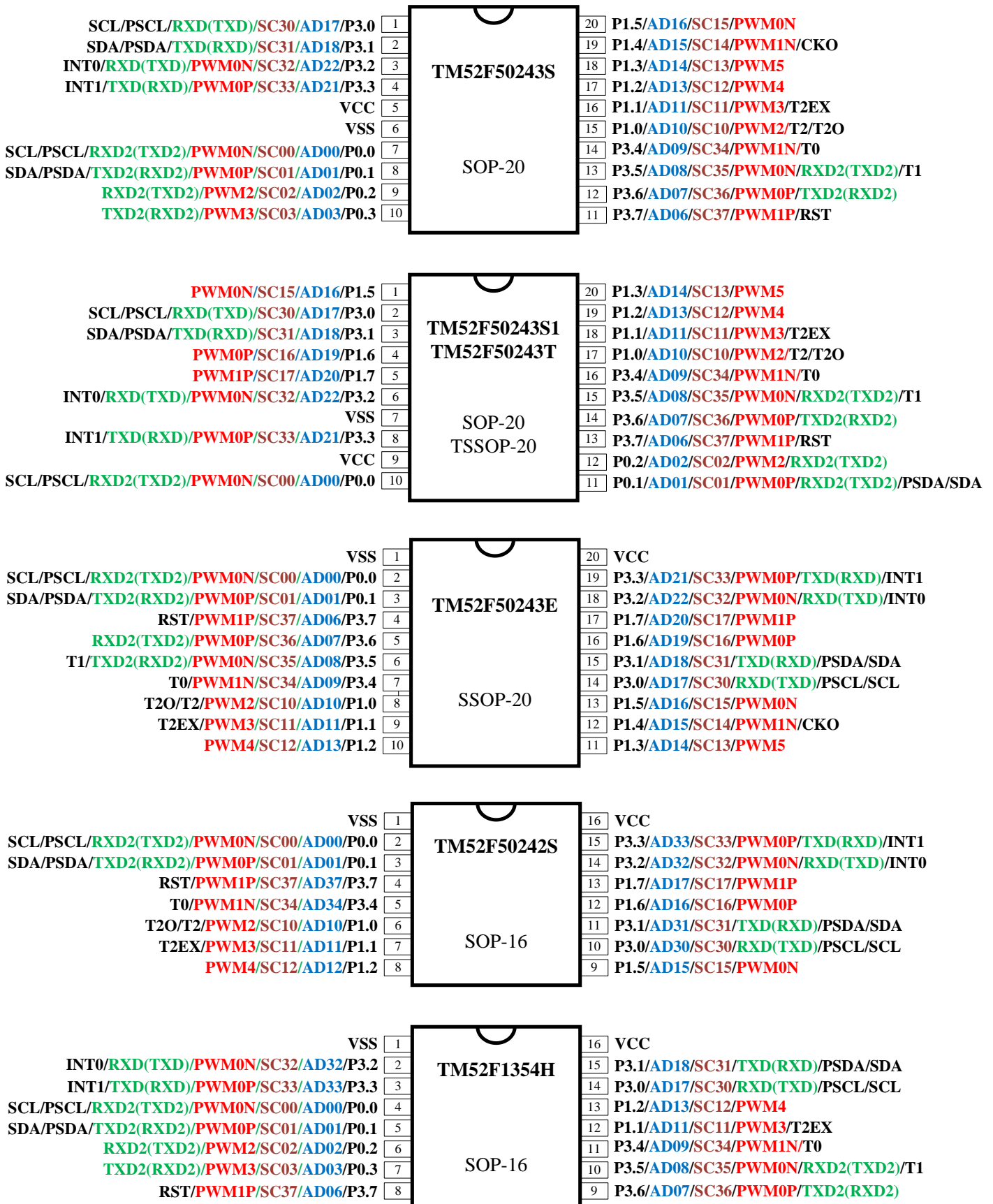
## PIN ASSIGNMENT

For low power applications, all digital I/Os (including unbonding or unused) should avoid high-impedance settings.

CKO/PWM1N/SC14/AD15/P1.4	1	 <p style="text-align: center;">TM52F50244E</p> <p style="text-align: center;">SSOP-24</p>	24	P1.3/AD14/SC13/PWM5
PWM0N/SC15/AD16/P1.5	2		23	P1.2/AD13/SC12/PWM4
SCL/PSCL/RXD(TXD)/SC30/AD17/P3.0	3		22	P1.1/AD11/SC11/PWM3/T2EX
SDA/PSDA/TXD(RXD)/SC31/AD18/P3.1	4		21	P1.0/AD10/SC10/PWM2/T2/T2O
PWM0P/SC16/AD19/P1.6	5		20	P3.4/AD09/SC34/PWM1N/T0
PWM1P/SC17/AD20/P1.7	6		19	P3.5/AD08/SC35/PWM0N/RXD2(TXD2)/T1
INT0/RXD(TXD)/PWM0N/SC32/AD22/P3.2	7		18	P3.6/AD07/SC36/PWM0P/TXD2(RXD2)
INT1/TXD(RXD)/PWM0P/SC33/AD21/P3.3	8		17	P3.7/AD06/SC37/PWM1P/RST
VSS	9		16	P0.5/AD05/SC05/PWM5
VCC	10		15	P0.4/AD04/SC04/PWM4
SCL/PSCL/RXD2(TXD2)/PWM0N/SC00/AD00/P0.0	11		14	P0.3/AD03/SC03/PWM3/TXD2(RXD2)
SDA/PSDA/TXD2(RXD2)/PWM0P/SC01/AD01/P0.1	12		13	P0.2/AD02/SC02/PWM2/RXD2(TXD2)

PWM0N/SC15/AD16/P1.5	1	 <p style="text-align: center;">TM52F50244E1</p> <p style="text-align: center;">SSOP-24</p>	24	P1.4/AD15/SC14/PWM1N/CKO
SCL/PSCL/RXD(TXD)/SC30/AD17/P3.0	2		23	P1.3/AD14/SC13/PWM5
SDA/PSDA/TXD(RXD)/SC31/AD18/P3.1	3		22	P1.2/AD13/SC12/PWM4
PWM0P/SC16/AD19/P1.6	4		21	P1.1/AD11/SC11/PWM3/T2EX
PWM1P/SC17/AD20/P1.7	5		20	P1.0/AD10/SC10/PWM2/T2/T2O
INT0/RXD(TXD)/PWM0N/SC32/AD22/P3.2	6		19	P3.4/AD09/SC34/PWM1N/T0
VSS	7		18	P3.5/AD08/SC35/PWM0N/RXD2(TXD2)/T1
INT1/TXD(RXD)/PWM0P/SC33/AD21/P3.3	8		17	P3.6/AD07/SC36/PWM0P/TXD2(RXD2)
VCC	9		16	P3.7/AD06/SC37/PWM1P/RST
SCL/PSCL/RXD2(TXD2)/PWM0N/SC00/AD00/P0.0	10		15	P0.5/AD05/SC05/PWM5
SDA/PSDA/TXD2(RXD2)/PWM0P/SC01/AD01/P0.1	11		14	P0.4/AD04/SC04/PWM4
RXD2(TXD2)/PWM2/SC02/AD02/P0.2	12		13	P0.3/AD03/SC03/PWM3/TXD2(RXD2)

VSS	1	 <p style="text-align: center;">TM52F50244E2</p> <p style="text-align: center;">SSOP-24</p>	24	VCC
SCL/PSCL/RXD2(TXD2)/PWM0N/SC00/AD00/P0.0	2		23	P3.3/AD21/SC33/PWM0P/TXD(RXD)/INT1
SDA/PSDA/TXD2(RXD2)/PWM0P/SC01/AD01/P0.1	3		22	P3.2/AD22/SC32/PWM0N/RXD(TXD)/INT0
RXD2(TXD2)/PWM2/SC02/AD02/P0.2	4		21	P1.7/AD20/SC17/PWM1P
TXD2(RXD2)/PWM3/SC03/AD03/P0.3	5		20	P1.6/AD19/SC16/PWM0P
PWM4/SC04/AD04/P0.4	6		19	P3.1/AD18/SC31/TXD(RXD)/PSDA/SDA
PWM5/SC05/AD05/P0.5	7		18	P3.0/AD17/SC30/RXD(TXD)/PSCL/SCL
RST/PWM1P/SC37/AD06/P3.7	8		17	P1.5/AD16/SC15/PWM0N
RXD2(TXD2)/PWM0P/SC36/AD07/P3.6	9		16	P1.4/AD15/SC14/PWM1N/CKO
T1/TXD2(RXD2)/PWM0N/SC35/AD08/P3.5	10		15	P1.3/AD14/SC13/PWM5
T0/PWM1N/SC34/AD09/P3.4	11		14	P1.2/AD13/SC12/PWM4
T2O/T2/PWM2/SC10/AD10/P1.0	12		13	P1.1/AD11/SC11/PWM3/T2EX



## PIN DESCRIPTION

Name	In/Out	Pin Description
P0.0~P0.5 P1.0~P1.7 P3.0~P3.7	I/O	Bit-programmable I/O port for Schmitt-trigger input, CMOS push-pull output or "open-drain" output. Pull-up and Pull-down resistors are assignable by software, so it can also be set to LCD 1/2 bias output. These pin's level change can interrupt/wake up CPU from Halt/Stop mode.
INT0, INT1	I	External low level or falling edge Interrupt input, Idle/Halt/Stop mode wake up input.
RXD	I/O	UART1 Mode0 transmit & receive data, Mode1/2/3 receive data
RXD2	I/O	UART2 Mode1/3 receive data
TXD	I/O	UART1 Mode0 transmit clock, Mode1/2/3 transmit data.
TXD2	I/O	UART2 Mode1/3 transmit data.
T0, T1, T2	I	Timer0, Timer1, Timer2 event count pin input.
T2EX	I	Timer2 external trigger input.
T0O	O	Timer0 overflow divided by 64 output
T2O	O	Timer2 overflow divided by 2 output
CKO	O	System Clock divided by 2 output
VBGO	O	Bandgap voltage output
PWM0P~PWM0N PWM1P~PWM1N PWM2~PWM5	O	16 bit PWM output
SC00~SC05 SC10~SC17 SC30~SC37		The pull-up resistor and pull-down resistor are turned on at the same time as LCD COM 1/2 bias output
AD00~AD22	I	ADC input
SCL	I/O	I <sup>2</sup> C SCL
SDA	I/O	I <sup>2</sup> C SDA
PSCL	I/O	I <sup>2</sup> C SCL for program
PSDA	I/O	I <sup>2</sup> C SDA for program
RSTn	I	External active low reset input, Pull-up resistor is fixed enable.
VCC, VSS	P	Power input pin and ground

**PIN SUMMERY**

Pin number								Pin Name	Type	Initial State	Input				Output		Alternate			other	
SSOP-24	SSOP-24	SSOP-24	SSOP-20	SOP/TSSOP-20	SOP-20	SOP-16	SOP-16				Pull-up	Pull_down	Wakeup	Ext. Interrupt	CMOS	Open-Drain	LCD	UART	PWM		IIC
3	18	2	1	2	14	10	14	SCL/PSCL/RXD(TXD)/SC30/AD17/P3.0	I/O	Hi-Z	●	●	●	●	●	●	●	●			
4	19	3	2	3	15	11	15	SDA/PSDA/TXD(RXD)/SC31/AD18/P3.1	I/O	Hi-Z	●	●	●	●	●	●	●	●			
7	22	6	3	6	18	14	2	INT0/RXD(TXD)/PWM0N/SC32/AD22/P3.2	I/O	Hi-Z	●	●	●	●	●	●	●	●	VBGO		
8	23	8	4	8	19	15	3	INT1/TXD(RXD)/PWM0P/SC33/AD21/P3.3	I/O	Hi-Z	●	●	●	●	●	●	●	●			
11	2	10	7	10	2	2	4	SCL/PSCL /RXD2(TXD2)/PWM0N/SC00/AD00/P0.0	I/O	Hi-Z	●	●	●	●	●	●	●	●			
12	3	11	8	11	3	3	5	SDA/PSDA /TXD2(RXD2)/PWM0P/SC01/AD01/P0.1	I/O	Hi-Z	●	●	●	●	●	●	●	●			
13	4	12	9	12	-	-	6	RXD2(TXD2)/PWM2/SC02/AD02/P0.2	I/O	Hi-Z	●	●	●	●	●	●	●	●			
14	5	13	10	-	-	-	7	TXD2(RXD2)/PWM3/SC03/AD03/P0.3	I/O	Hi-Z	●	●	●	●	●	●	●	●			
15	6	14	-	-	-	-	-	PWM4/SC04/AD04/P0.4	I/O	Hi-Z	●	●	●	●	●	●	●	●			
16	7	15	-	-	-	-	-	PWM5/SC05/AD05/P0.5	I/O	Hi-Z	●	●	●	●	●	●	●	●			
17	8	16	11	13	4	4	8	RST/PWM1P/SC37/AD06/P3.7	I/O	Hi-Z	●	●	●	●	●	●	●	●	Reset		
18	9	17	12	14	5	-	9	TXD2(RXD2)/PWM0P/SC36/AD07/P3.6	I/O	Hi-Z	●	●	●	●	●	●	●	●			
19	10	18	13	15	6	-	10	RXD2(TXD2)/PWM0N/SC35/AD08/P3.5	I/O	Hi-Z	●	●	●	●	●	●	●	●			
20	11	19	14	16	7	5	11	T0/PWM1N/SC34/AD09/P3.4	I/O	Hi-Z	●	●	●	●	●	●	●	●	T0		
21	12	20	15	17	8	6	-	T2O/T2/PWM2/SC10/AD10/P1.0	I/O	Hi-Z	●	●	●	●	●	●	●	●	T2O		
22	13	21	16	18	9	7	12	T2EX/PWM3/SC11/AD11/P1.1	I/O	Hi-Z	●	●	●	●	●	●	●	●			
23	14	22	17	19	10	8	13	PWM4/SC12/AD13/P1.2	I/O	Hi-Z	●	●	●	●	●	●	●	●			
24	15	23	18	20	11	-	-	PWM5/SC13/AD14/P1.3	I/O	Hi-Z	●	●	●	●	●	●	●	●			
1	16	24	19	-	12	-	-	CKO/PWM1N/SC14/AD15/P1.4	I/O	Hi-Z	●	●	●	●	●	●	●	●	CKO		
2	17	1	20	1	13	9	-	PWM0N/SC15/AD16/P1.5	I/O	Hi-Z	●	●	●	●	●	●	●	●			
5	20	4	-	4	16	12	-	PWM0P/SC16/AD19/P1.6	I/O	Hi-Z	●	●	●	●	●	●	●	●			
6	21	5	-	5	17	13	-	PWM1N/SC17/AD20/P1.7	I/O	Hi-Z	●	●	●	●	●	●	●	●			
9	1	7	6	7	1	1	1	VSS	P												
10	24	9	5	9	20	16	16	VCC	P												

## FUNCTIONAL DESCRIPTION

### 1. CPU Core

In the 8051 architecture, the C programming language is used as a development platform. The TM52 device features a fast 8051 core in a highly integrated microcontroller, allowing designers to be able to achieve improved performance compared to a classic 8051 device. TM52 series microcontrollers provide a complete binary code with standard 8051 instruction set compatibility, ensuring an easy migration path to accelerate the development speed of system products. The CPU core includes an ALU, a program status word (PSW), an accumulator (ACC), a B register, a stack point (SP), DPTRs, a program counter, an instruction decoder, and core special function registers (SFRs).

#### 1.1 Accumulator (ACC)

This register provides one of the operands for most ALU operations. Accumulators are generally referred to as A or Acc and sometimes referred to as Register A. In this document, the accumulator is represented as “A” or “ACC” including the instruction table. The accumulator, as its name suggests, is used as a general register to accumulate the intermediate results of a large number of instructions. The accumulator is the most important and frequently used register to complete arithmetic and logical operations. It holds the intermediate results of most arithmetic and logic operations and assists in data transportation.

SFR E0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>ACC</b>	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	R/W
Reset	0	0	0	0	0	0	0	0

E0h.7~0 **ACC**: Accumulator

#### 1.2 B Register (B)

The “B” register is very similar to the ACC and may hold a 1 Byte value. This register provides the second operand for multiply or divide instructions. Otherwise, it may be used as a scratch pad register. The B register is only used by two 8051 instructions, MUL and DIV. When A is to be multiplied or divided by another number, the other number is stored in B. For MUL and DIV instructions, it is necessary that the two operands are in A and B.

ex: DIV AB

When this instruction is executed, data inside A and B are divided, and the answer is stored in A.

SFR F0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>B</b>	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	R/W
Reset	0	0	0	0	0	0	0	0

F0h.7~0 **B**: B register

### 1.3 Stack Pointer (SP)

The SP register contains the Stack Pointer. The Stack Pointer is used to load the program counter into memory during LCALL and ACALL instructions and is used to retrieve the program counter from memory in RET and RETI instructions. The stack may also be saved or loaded using PUSH and POP instructions, which also increment and decrement the Stack Pointer.

SFR 81h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SP</b>	SP							
R/W	R/W							
Reset	0	0	0	0	0	1	1	1

81h.7~0 **SP:** Stack Point

### 1.4 Dual Data Pointer (DPTRs)

TM52 device has two DPTRs, which share the same SFR address. Each DPTR is 16 bits in size and consists of two registers: the DPTR high byte (DPH) and the DPTR low byte (DPL). The DPTR is used for 16-bit-address external memory accesses, for offset code byte fetches, and for offset program jumps. Setting the DPSEL control bit allows the program code to switch between the two physical DPTRs.

SFR 82h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>DPL</b>	DPL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

82h.7~0 **DPL:** Data Point low byte

SFR 83h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>DPH</b>	DPH							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

83h.7~0 **DPH:** Data Point high byte

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX1</b>	CLRWDT	CLRTM3	–	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	–	R/W	R/W	R/W	R/W	R/W
Reset	0	0	–	0	0	0	0	0

F8h.0 **DPSEL:** Active DPTR Select

### 1.5 Program Status Word (PSW)

This register contains status information resulting from CPU and ALU operations. The instructions that affect the PSW are listed below.

Instruction	Flag			Instruction	Flag		
	C	OV	AC		C	OV	AC
ADD	X	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	X			ORL C, /bit	X		
RRC	X			MOV C, bit	X		
RLC	X			CJNE	X		
SETB C	1						

A “0” means the flag is always cleared, a “1” means the flag is always set and an “X” means that the state of the flag depends on the result of the operation.

SFR D0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PSW</b>	CY	AC	F0	RS1	RS0	OV	F1	P
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

D0h.7 **CY**: ALU carry flag

D0h.6 **AC**: ALU auxiliary carry flag

D0h.5 **F0**: General purpose user-definable flag

D0h.4~3 **RS1, RS0**: The contents of (RS1, RS0) enable the working register banks as:

00: Bank 0 (00h~07h)

01: Bank 1 (08h~0Fh)

10: Bank 2 (10h~17h)

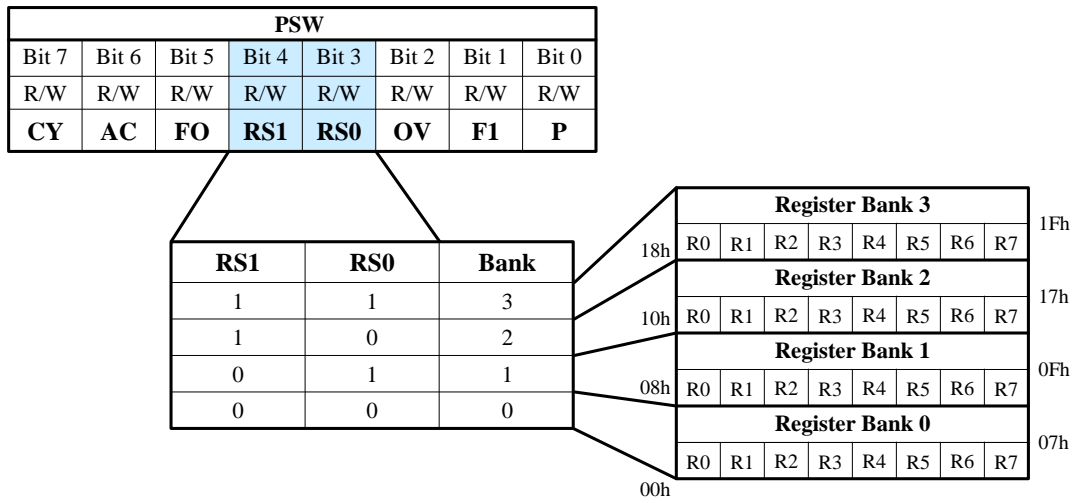
11: Bank 3 (18h~1Fh)

D0h.2 **OV**: ALU overflow flag

D0h.1 **F1**: General purpose user-definable flag

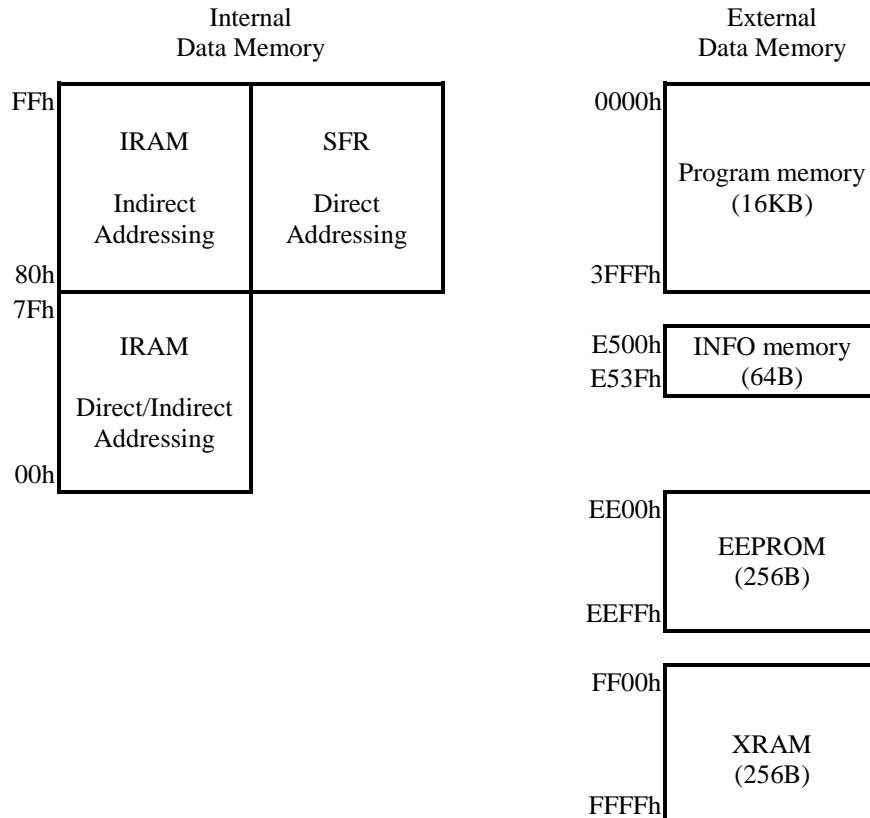
D0h.0 **P**: Parity flag. Set/cleared by hardware each instruction cycle to indicate odd/even number of “one” bits in the accumulator.





## 2. Memory

As the standard 8051, the Chip has Internal and External Data Memory space. The Internal Data Memory space consists of 256 bytes IRAM and SFRs, which are accessible through a rich instruction set. The External Data Memory space consists of 256 bytes XRAM, 256 bytes EEPROM, 64 bytes INFO memory and 16K bytes Program memory, which can be only accessed by MOVX instruction, Program memory also can be accessed by MOVC instruction.

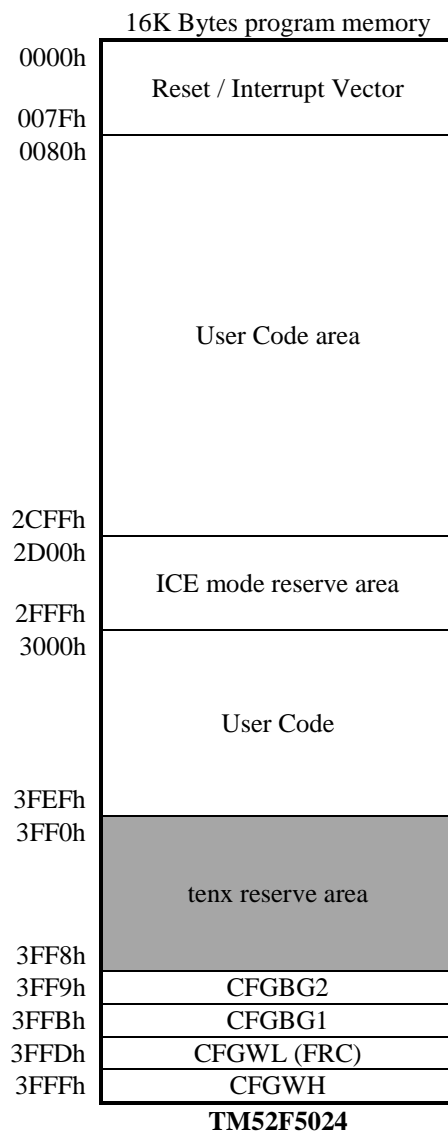


## 2.1 Program Memory

The Chip has a 16K Bytes Flash program memory which can support In Circuit Programming (ICP), In Application Programming (IAP) and In System Programming (ISP) function modes. The Flash write endurance is at least 100 cycles. The program memory address continuous space (0000h~3FFFh) is partitioned to several sectors for device operation.

### 2.1.1 Program Memory Functional Partition

The last 16 bytes (3FF0h~3FFFh) of program memory is defined as chip Configuration Word (CFGW), which is loaded into the device control registers upon power on reset (POR). The 0000h~007Fh is occupied by Reset/Interrupt vectors as standard 8051 definition. For **TM52F5024**, the address space 3000h~3FEFh is defined as the IAP area. In the in-circuit emulation (ICE) mode, user also needs to reserve the address space 2D00h~2FFFh for ICE System communication.



The **CFGW area** has 4 data bytes (CFGWH, CFGWL, CFGBG1 and CFGBG2), which is located at the last 16 addresses of Flash memory. CFGWL is copied to the SFR F6h, CFGBG1 is copied to the SFR

F5h and CFGBG2 is copied to the SFR E4h after power on reset, software then take over CFGWL's, CFGBG1's and CFGBG2's control capability by modifying the SFR F6h, F5h and E4h

### 2.1.2 Flash ICP Mode

The Flash memory can be programmed by the tenx proprietary writer (**TWR98/TWR99**), which needs at least four wires (VCC, VSS, P3.0 and P3.1) to connect to this chip. If user wants to program the Flash memory on the target circuit board (In Circuit Program, ICP), these pins must be reserved sufficient freedom to be connected to the Writer.

Writer wire number	Pin connection
4-Wire	VCC, VSS, P3.0, P3.1

### 2.1.3 Flash IAP Mode

The **TM52F5024** has “In Application Program” (IAP) capability, which allows software to read/write data from/to the Flash memory during CPU run time as conveniently as data EEPROM access. The IAP function is byte writable, meaning that the **TM52F5024** does not need to erase one Flash page before write.

Both write 47h and 74h to IAPWE\_SFR (C9h.7~0) can let IAPWE=1, the difference is when user write 47h to IAPWE\_SFR, user can write one byte at once, when user write 74h to IAPWE\_SFR, user can write two byte at once to save write time.

To use IAP function, user need to meet the following condition:

1. Only User Code area can be written by IAP
2. Set IAPALL=1 and IAPWE=1

The **TM52F5024** has a true EEPROM memory. It has the wider writing voltage range and the better write endurance than Flash memory. It is recommended to use EEPROM memory to store application data first.

The **CFGW area** has 3 data bytes (CFGWH, CFGWL and CFGBG), which is located at the last 16 addresses of Flash memory. The CFGWH is not accessible to IAP, while the CFGWL and CFGBG can be read or written by IAP in case the IAPALL flag is set. CFGWL is copied to the SFR F6h and CFGBG is copied to the SFR F5h after power on reset, software then take over CFGWL's and CFGBG's control capability by modifying the SFR F6h and F5h.

### 2.1.4 IAP Mode Access Routines

**Flash IAP Write** is simply achieved by a “MOVX @DPTR, A” instruction while the DPTR contains the target Flash address (0000h~3FFEh), and the ACC contains the data being written. The **TM52F5024** accepts IAP write command only when IAPWE=1. Flash IAP writing one byte requires approximately 1 ms @V<sub>CC</sub>=5V. Meanwhile, the CPU stays in a waiting state, but all peripheral modules (Timers and others) continue running during the writing time. The software must handle the pending interrupts after an IAP write. The **TM52F5024** has a build-in IAP Time-out function IAPTE (F7h.2~1) for escaping write fail state. Flash IAP writing needs higher V<sub>CC</sub> voltage, V<sub>CC</sub>>4.0V and WDT disabled to avoid false writing.

Because the Program memory and the IAP data space share the same entity, a **Flash IAP Read** can be performed by the “MOVX A, @DPTR” or “MOVC” instruction as long as the target address points to the 0000h~3FFEh area. A Flash IAP read does not require extra CPU wait time.

```

; IAP example code (ASM)
; need 4.0V < VCC < 5.5V
ANL    AUX2, #3Fh           ; WDT function disable
ORL    AUX2, #04h          ; IAP Time-Out function enable
MOV    DPTR, #3F00h        ; DPTR=3F00h=target IAP address
CLR    EA                  ; Disable Interrupt
ORL    LVRCON, #10h        ; Disable LVR
MOV    SWCMD, #65h         ; IAPALL flag =1
MOV    A, #5Ah              ; A=5Ah=target IAP write data
MOV    IAPWE_SFR, #47h     ; IAP write enable
MOVX   @DPTR, A            ; Flash[3F00h] =5Ah, after IAP write
                                ; 0.5ms~1ms H/W writing time, CPU wait
MOV    IAPWE_SFR, #00h     ; IAP write disable, immediately after IAP write
CLR    A                    ; A=0
MOVC   A, @A+DPTR          ; A=5Ah

```

```

; IAP example code (C)
; need 4.0V < VCC < 5.5V
unsigned char xdata *pMOVX;
unsigned char code *pMOVC;
EA = 0;           // Disable Interrupt
AUX2 = 0x04;      // WDT function disable & IAP Time-Out function enable
IAPALL = 0x65;
IAPWE_SFR = 0x47;
pMOVX = 0x2002;
*pMOVX = wData;   // write data into ROM (0x2002)
IAPWE_SFR = 0x00;
IAPALL = 0x00;
pMOVC = 0x2105;
rData = *pMOVC;  // read data from ROM (2105)

```

SFR 97h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
SWCMD	IAPALL/SWRST								
	–						WDTO	IAPALL	
R/W	W						R	R	
Reset	–						0	0	

97h.7~0 **IAPALL (W):** Write 65h to set IAPALL control flag; Write other value to clear IAPALL flag. It is recommended to clear it immediately after IAP access.

97h.0 **IAPALL (R):** Flag indicates Flash memory sectors can be accessed by IAP or not.

SFR C9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IAPWE_SFR</b>	IAPWE_SFR							
	IAPWE	IAPTO	EEPWE	INFOWE	EEPTO			
R/W	R	R	R	R	R	W		
Reset	0	0	0	0	0			

- C9h.7~0 **IAPWE (W):** Write 47h or 74h to set IAPWE flag; Write 47h can write 1 byte at once, write 74h can write 2 bytes at once. Write other value to clear IAPWE. It's recommended to clear it immediately after IAP write.  
Write A1h to set INFOWE flag; write other value to clear INFOWE flag. It's recommended to clear it immediately after IAP write  
Write E2h to set EEPWE flag; Write other value to clear EEPWE flag. It is recommended to clear it immediately after EEPROM write.
- C9h.7 **IAPWE (R):** Flag indicates Flash memory can be written by IAP or not, 1=IAP Write enable.
- C9h.6 **IAPTO (R):** IAP Time-Out flag, Set by H/W when IAP Time-out occurs. Cleared by H/W when IAPWE=0

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSV	VBGOUT	DIV32	IAPTE		MULDIV16
	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	1	1	0

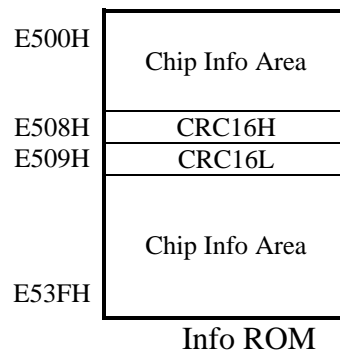
- F7h.7~6 **WDTE:** Watchdog Timer Reset control  
0x: Watchdog Timer Reset disable  
10: Watchdog Timer Reset enable in Fast/Slow mode, disable in Idle/Halt/Stop mode  
11: Watchdog Timer Reset always enable
- F7h.2~1 **IAPTE:** IAP (or EEPROM) write watchdog timer enable  
00: Disable  
01: wait 6.9 ms trigger watchdog time-out flag, and escape the write fail state  
10: wait 27 ms trigger watchdog time-out flag, and escape the write fail state  
11: wait 55 ms trigger watchdog time-out flag, and escape the write fail state

### 2.1.5 Flash ISP Mode

The “In System Programming” (ISP) usage is similar to IAP, except the purpose is to refresh the Program code. User can use UART/SPI or other method to get new Program code from external host, then writes code as the same way as IAP. ISP operation is complicated; basically it needs to assign a Boot code area to the Flash which does not change during the ISP process.

## 2.2 Information Memory

The Chip has a 64 words Information memory. The Information memory address continuous space (E500h~E53Fh) is partitioned to several sector for device operation. Chip Info area is tenx reserved defined as production information, such as ID, Special Regulations, Code Num, checksum. User can store new checksum code in this area after Flash IAP. CRC16H/L is the reserved area of the checksum. Tenx can provide a CRC verification subroutine. The user can calculate the checksum by the CRC verification subroutine to compare with CRC16H/L and check the validity of the ROM code.



**Info Rom IAP Write** is simply achieved by a “MOVX @DTPR, A” instruction while the DPTR contains the target Info ROM (E500h~E53Fh) address, and the ACC contains the data being written. Info ROM writing requires approximately 0.6 ms @V<sub>CC</sub>=4V~5.5V, VCC capacitance greater than 200uF. During the period of IAP, the CPU stays in waiting state, but all peripheral modules (Timers and others) continue running during the writing time. The software must handle the pending interrupts after an IAP write. The chip has a build-in write Time-out function selected by IAPTE (E7h.2~1) to escape write fail state. Besides, S/W must disable WDT before IAP write.

**Info Rom IAP Read** only can be performed by the “MOVX” instruction as long as the target address points to E500h~E53Fh area. A Info TOM IAP read does not require extra CPU wait time.

```

; Info ROM IAP example code
; need 4.0V < VCC < 5.5V
ANL    AUX2, #3Fh          ; WDT function disable
ORL    AUX2, #04h          ; IAP Time-Out function enable
MOV    DPTR, #E530h        ; DPTR=E530h=target IAP Info ROM address
CLR    EA                  ; Disable Interrupt
ORL    LVRCON, #10h        ; Disable LVR
MOV    SWCMD, #65h         ; IAPALL flag =1
MOV    A, #5Ah             ; A=5Ah=target IAP write data
MOV    IAPWE_SFR, #A1h     ; Info ROM IAP write enable
MOVX   @DPTR, A            ; Flash[E530h] =5Ah, after IAP write
                                ; 0.5ms~1ms H/W writing time, CPU wait
MOV    IAPWE_SFR, #00h     ; IAP write disable, immediately after IAP write
CLR    A                   ; A=0
MOVC   A, @A+DPTR          ; A=5Ah

```

SFR C9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IAPWE_SFR</b>	IAPWE_SFR							
	IAPWE	IAPTO	EEPWE	INFOWE	EPTO	–		
R/W	R	R	R	R	R	W		
Reset	0	0	0	0	0	–		

- C9h.7~0 **IAPWE (W):** Write 47h or 74h to set IAPWE flag; Write 47h can write 1 byte at once, write 74h can write 2 bytes at once. Write other value to clear IAPWE. It's recommended to clear it immediately after IAP write.  
Write A1h to set INFOWE flag; write other value to clear INFOWE flag. It's recommended to clear it immediately after IAP write  
Write E2h to set EEPWE flag; Write other value to clear EEPWE flag. It is recommended to clear it immediately after EEPROM write
- C9h.6 **IAPTO (R):** IAP Time-Out flag, Set by H/W when IAP Time-out occurs. Cleared by H/W when INFOWE=0.
- C9h.4 **INFOWE(R):** Flag indicates Info ROM can be written by IAP or not, 1=Write INFO enable.

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSVAV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	1	1	0

- F7h.7~6 **WDTE:** Watchdog Timer Reset control  
0x: Watchdog Timer Reset disable  
10: Watchdog Timer Reset enable in Fast/Slow mode, disable in Idle/Halt/Stop mode  
11: Watchdog Timer Reset always enable
- F7h.2~1 **IAPTE:** IAP (or EEPROM) write watchdog timer enable  
00: Disable  
01: wait 6.9 ms trigger watchdog time-out flag, and escape the write fail state  
10: wait 27 ms trigger watchdog time-out flag, and escape the write fail state  
11: wait 55 ms trigger watchdog time-out flag, and escape the write fail state



### 2.3 EEPROM Memory

The **TM52F5024** contains 256 bytes of data EEPROM memory. It is organized as a separate data space, in which single bytes can be read and written. The EEPROM has an endurance of at least 10K write/erase cycles.

EEPROM Memory	
<b>EE00h</b>	EEPROM[0]
<b>EE01h</b>	EEPROM[1]
<b>EE02h</b>	⋮
<b>EEFEh</b>	EEPROM[254]
<b>EEFFh</b>	EEPROM[255]

**The EEPROM Write** usage is similar to Flash IAP mode. It is simply achieved by a “MOVX @DPTR, A” instruction while the DPTR contains the target EEPROM address (EE00h~EEFEh), and the ACC contains the data being written. EEPROM writing requires approximately 6 ms @V<sub>CC</sub>=2.7V, 1.2 ms @V<sub>CC</sub>=5V. Meanwhile, the CPU and all peripheral modules (Timers and others) continue running during the writing time.

There is also a writing method that can write 1~4 bytes of data to a specified address at a time. S/W Write the data to be written to EEPROM into EEPWD0~EEPWD3 (SFR 9Ah~9Dh) in advance, and write the starting address to be written into EEPROM into EEPWADR (SFR 9Eh), then add the number of data to be written and H/W Write the control signal to start writing to EEPWCON[2:0](SFR AEh) H/W to write data into EEPROM. The CPU and all peripheral modules (timers, etc.) can continue to run during the write.

TM52F5024 has a built-in EEPROM watchdog timer (shared with the IAP watchdog timer) to exit the stuck state when writing fails. After the writing is completed or the writing time is too long (the EEPROM watchdog timer needs to be turned on), the interrupt flag EEPIF will be generated. The software can use this interrupt or poll EEPBUSY to know whether the EEPROM writing is completed. EEPIF is automatically cleared when the program executes the interrupt service routine. The interrupt service routine needs to check EEPTO to determine whether the writing is completed (EEPTO=0) or failed (EEPTO=1). Writing EEPROM data requires VCC > 3.0V and WDT is turned off to avoid accidental writing.

**The EEPROM Read** can be performed by the “MOVX A, @DPTR” instruction as long as the target address points to the EE00h~EEFEh area. The EEPROM read does require approximately 300ns.

```

; EEPROM example code
; need 3.0V < VCC < 5.5V
ANL    AUX2, #3Fh          ; WDT function disable
ORL    AUX2, #04h          ; EEPROM Time-Out function enable
MOV    DPTR, #0EE00h       ; DPTR=EE00h=target EEPROM[0] address
MOV    A, #0A5h            ; A=A5h=target EEPROM[0] write data
MOV    IAPWE_SFR, #0E2h    ; EEPROM write enable
MOVX   @DPTR, A           ; EEPROM[0]=A5h, after EEPROM write
                                ; 1ms~2ms H/W writing time, CPU wait
MOV    IAPWE_SFR, #000h    ; EEPROM write disable, immediately after EEPROM write
CLR    A                   ; A=0
MOVX   A, @DPTR           ; A=A5h

```

; **EEPROM H/W 写入 1~4 byte** example code  
 ; 需要 3.0V < V<sub>CC</sub> < 5.5V

```
ANL    AUX2, #3Fh          ; WDT function disable
ORL    AUX2, #04h         ; EEPROM Time-Out function enable
MOV    EEPWD0, #12h       ; 1st byte data
MOV    EEPWD1, #34h       ; 2nd byte data
MOV    EEPWD2, #56h       ; 3rd byte data
MOV    EEPWD3, #78h       ; 4th byte data
MOV    EEPWADR, #A0h      ; H/W write EEPROM start address
MOV    EEPWCON, #02h      ;
MOV    EEPWCON, #03h      ; set EEPWCON[2]=1, H/W start write data to EEPROM
                                ; set EEPWCON[1:0] write to EEPROM byte number
                                ; 1ms~6ms H/W writing time, CPU does not need wait
```

**Note: IAP write/read Flash/Info ROM or write/read EEPROM can only be used outside or inside the interrupt. If the IAP write/read Flash/Info ROM or write/read EEPROM are used both inside and outside the interrupt, it may cause errors.**

SFR C9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IAPWE_SFR</b>	IAPWE_SFR							
	IAPWE	IAPTO	EEPWE	INFOWE	EPTO	-		
R/W	R	R	R	R	R	W		
Reset	0	0	0	0	0	-		

C9h.7~0 **IAPWE (W):** Write 47h or 74h to set IAPWE flag; Write 47h can write 1 byte at once, write 74h can write 2 bytes at once. Write other value to clear IAPWE. It's recommended to clear it immediately after IAP write.  
 Write A1h to set INFOWE flag; write other value to clear INFOWE flag. It's recommended to clear it immediately after IAP write  
 Write E2h to set EEPWE flag; Write other value to clear EEPWE flag. It is recommended to clear it immediately after EEPROM write

C9h.5 **EEPWE(R):** Flag indicates EEPROM can be written by IAP or not, 1=Write EEPROM enable.

C9h.3 **EEPTO (R):** EEPROM write Time-Out flag, Set by H/W when EEPROM write Time-out occurs. Cleared by H/W when EEPWE=0.

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSVAV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	1	1	0

F7h.7~6 **WDTE:** Watchdog Timer Reset control  
 0x: Watchdog Timer Reset disable  
 10: Watchdog Timer Reset enable in Fast/Slow mode, disable in Idle/Halt/Stop mode  
 11: Watchdog Timer Reset always enable

F7h.2~1 **IAPTE:** IAP (or EEPROM) write watchdog timer enable  
 00: Disable  
 01: wait 6.9 ms trigger watchdog time-out flag, and escape the write fail state  
 10: wait 27 ms trigger watchdog time-out flag, and escape the write fail state  
 11: wait 55 ms trigger watchdog time-out flag, and escape the write fail state

SFR 9Ah	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EEPWD0</b>	EEPWD0							
	W							
Reset	-	-	-	-	-	-	-	-

9Ah.7~0 **EEPWD0:** The first byte data to EEPROM

SFR 9Bh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EEPWD1</b>	EEPWD1							
R/W	W							
Reset	-	-	-	-	-	-	-	-

9Bh.7~0 **EEPWD1:** The second byte data to EEPROM

SFR 9Ch	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EEPWD2</b>	EEPWD2							
R/W	W							
Reset	-	-	-	-	-	-	-	-

9Ch.7~0 **EEPWD2:** The third byte data to EEPROM

SFR 9Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EEPWD3</b>	EEPWD3							
R/W	W							
Reset	-	-	-	-	-	-	-	-

9Dh.7~0 **EEPWD3:** The fourth byte data to EEPROM

SFR 9Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EEPWADR</b>	EEPWADR							
R/W	W							
Reset	-	-	-	-	-	-	-	-

9Eh.7~0 **EEPWADR:** The beginning address of write to EEPROM

SFR AEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EEPWCON</b>	-	-	-	-	-	HWSTART	HWLEN	
R/W	-	-	-	-	-	W	W	W
Reset	-	-	-	-	-	0	0	0

AEh.2 **HWSTART:** H/W start write to EEPROM control signal

0: disable H/W write EEPROM

1: enable H/W write EEPROM, H/W clear it after write finish

AEh.1~0 **HWLEN:** the byte length write to EEPROM when HWSTART set to 1

00: 1 byte

01: 2 bytes

10: 3 bytes

11: 4 bytes

## 2.4 Data Memory

As the standard 8051, the Chip has both Internal and External Data Memory space. The Internal Data Memory space consists of 256 Bytes IRAM and SFRs, which are accessible through a rich instruction set. The External Data Memory space consists of 256 Bytes XRAM, 256 Bytes EEPROM and IAP Flash, which can be only accessed by MOVX instruction.

### 2.4.1 IRAM

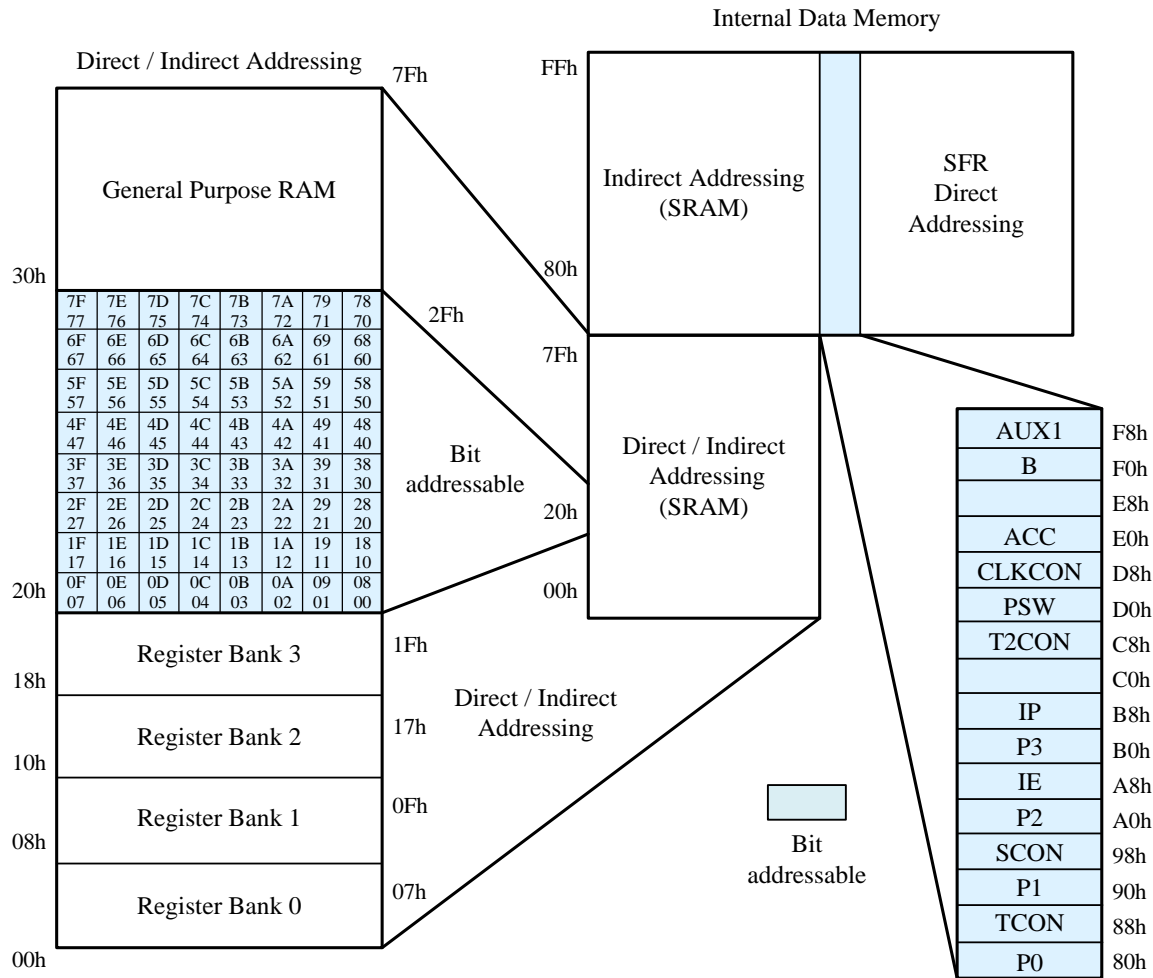
IRAM is located in the 8051 internal data memory space. The whole 256 Bytes IRAM are accessible using indirect addressing but only the lower 128 Bytes are accessible using direct addressing. There are four directly addressable register banks (switching by PSW), which occupy IRAM space from 00h to 1Fh. The address 20h to 2Fh 16 Bytes IRAM space is bit-addressable. IRAM can be used as scratch pad registers or program stack.

### 2.4.2 XRAM

XRAM is located in the 8051 external data memory space (address from FF00h to FFFFh). The 256 Bytes XRAM can be only accessed by “MOVX” instruction.

### 2.4.3 SFRs

All peripheral functional modules such as I/O ports, Timers and UART operations for the chip are accessed via Special Function Registers (SFRs). These registers occupy upper 128 Bytes of direct Data Memory space locations in the range 80h to FFh. There are 14 bit-addressable SFRs (which means that eight individual bits inside a single byte are addressable), such as ACC, B register, PSW, TCON, SCON, and others. The remaining SFRs are only byte addressable. SFRs provide control and data exchange with the resources and peripherals of the Chip. The TM52 series of microcontrollers provides complete binary code with standard 8051 instruction set compatibility. Beside the standard 8051 SFRs, the Chip implements additional SFRs used to configure and access subsystems such as the ADC/LCD, which are unique to the Chip.



	8/0	9/1	A/2	B/3	C/4	D/5	E/6	F/7
F8h	AUX1							
F0h	B	CRCDL	CRCDH	CRCIN	TESTMODE	CFGBG1	CFGWL	AUX2
E8h						PCL	PCH	AUX3
E0h	ACC	MICON	MIDAT	LVRCON	CFGBG2	EFTCON	EXA	EXB
D8h	CLKCON	PWM0PRDH	PWM0PRDL	PWM1PRDH	PWM1PRDL	PWM3DH	PWM3DL	RDCTL
D0h	PSW	PWM0DH	PWM0DL	PWM1DH	PWM1DL	PWM2DH	PWM2DL	
C8h	T2CON	IAPWE_SF R	RCP2L	RCP2H	TL2	TH2	EXA2	EXA3
C0h		PWM4DH	PWM4DL	PWM5DH	PWM5DL			
B8h	IP	IPH	IP1	IP1H				LVDSEL
B0h	P3						ADCHS	
A8h	IE	INTE	ADC DL	ADC DH			EERPWCN	PWMCON3
A0h	P2	PWMCON	PINMOD10	PINMOD32	PINMOD54	PINMOD76	PINMOD	PWMCON2
98h	SCON	SBUF	EERPWD0	EERPWD1	EERPWD2	EERPWD3	EERPWADR	UART2CON
90h	P1	PORTIDX		UARTCON	OPTION	INTFLG	INTPIN	SWCMD
88h	TCON	TMOD	TL0	TL1	TH0	TH1	SCON2	SBUF2
80h	P0	SP	DPL	DPH	INTE2	INTPORT	INTPWM	PCON

### 3. LVR and LVD setting

The Chip provides LVR and Low Voltage Detection (LVD) functions. There are 16-level LVD can be selected by CFGWH and 16-level LVR can be selected by SFR LVRSEL. The SFR PWRSAV/LVRPD bits also affect LVR function as tables below. After the program STARTUP, the first instruction in main requires a new LVR setting.

Operation Mode	SFR		CFGWH	LVR	Function	Note
	LVRPD	PWRSAV	LVRE			
Fast Slow	0	X	0000	ON	LV Reset 1.79V	
	0	X	0001	ON	LV Reset 1.95V	
	0	X	0010	ON	LV Reset 2.11V	
	0	X	0011	ON	LV Reset 2.26V	
	0	X	0100	ON	LV Reset 2.40V	
	0	X	0101	ON	LV Reset 2.56V	
	0	X	0110	ON	LV Reset 2.72V	
	0	X	0111	ON	LV Reset 2.87V	
	0	X	1000	ON	LV Reset 3.03V	
	0	X	1001	ON	LV Reset 3.18V	
	0	X	1010	ON	LV Reset 3.32V	
	0	X	1011	ON	LV Reset 3.50V	
	0	X	1100	ON	LV Reset 3.63V	
	0	X	1101	ON	LV Reset 3.80V	
	0	X	1110	ON	LV Reset 3.94V	
	0	X	1111	ON	LV Reset 4.12V	
Idle Stop Halt	0	0	0000	ON	LV Reset 1.79V	Current consumption about 70uA
	0	0	0001	ON	LV Reset 1.95V	
	0	0	0010	ON	LV Reset 2.11V	
	0	0	0011	ON	LV Reset 2.26V	
	0	0	0100	ON	LV Reset 2.40V	
	0	0	0101	ON	LV Reset 2.56V	
	0	0	0110	ON	LV Reset 2.72V	
	0	0	0111	ON	LV Reset 2.87V	
	0	0	1000	ON	LV Reset 3.03V	
	0	0	1001	ON	LV Reset 3.18V	
	0	0	1010	ON	LV Reset 3.32V	
	0	0	1011	ON	LV Reset 3.50V	
	0	0	1100	ON	LV Reset 3.63V	
	0	0	1101	ON	LV Reset 3.80V	
	0	0	1110	ON	LV Reset 3.94V	
	0	0	1111	ON	LV Reset 4.12V	
Idle	0	1	XXXX	ON	Disable LVR Enable POR 1.6V	Current consumption about 16uA
Stop Halt	0	1	XXXX	OFF	Disable	*Minimum Current consumption 0.1uA
Fast Slow Idle	1	X	XXXX	ON	Disable LVR Enable POR 1.6V	Current consumption about 16uA
Stop Halt	1	X	XXXX	OFF	Disable	*Minimum Current consumption 0.1uA

**Note:** The current consumption of Halt mode is more than Stop mode about 5.5~23uA, because SRC is enabled.

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSAB	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

F7h.5 **PWRSAB**: chip power-saving option  
Set 1 to reduce the chip's power consumption at Idle/Halt/Stop Mode

SFR BFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>LVDS</b>	LVDM	LVDO	–	LVDPD	LVDSSEL			
R/W	R/W	R	–	R/W	R/W	R/W	R/W	R/W
Reset	0	0	–	0	CFGWH[4:1]			

BFh.7 **LVDM**:  
0:  $V_{cc} < VLVD$  (LVDIF=1 while LVDO=1)  
1:  $V_{cc} > VLVD$  (LVDIF=1 while LVDO=0)

BFh.6 **LVDO**: Low Voltage Detect output

BFh.4 **LVDPD**: Low Voltage Detect function select (Auto disable in Idle/Halt/Stop mode)  
0: enable  
1: disable

BFh.4 **LVDSSEL**: Low voltage detect (LVD) select  
0000: set LVD at 1.79V  
0001: set LVD at 1.95V  
0010: set LVD at 2.11V  
0011: set LVD at 2.26V  
0100: set LVD at 2.40V  
0101: set LVD at 2.56V  
0110: set LVD at 2.71V  
0111: set LVD at 2.87V  
1000: set LVD at 3.03V  
1001: set LVD at 3.18V  
1010: set LVD at 3.32V  
1011: set LVD at 3.50V  
1100: set LVD at 3.63V  
1101: set LVD at 3.80V  
1110: set LVD at 3.94V  
1111: set LVD at 4.12V

SFR E3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>LVRCON</b>	–	PORPD_SAV	PORPD	LVRPD	LVRSEL			
R/W	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	1	0	0	0	0	0	0

E3h.6 **PORPD\_SAV**: POR disable at Stop mode

0: POR enable, 1: POR disable

E3h.5 **PORPD**: POR Power Down

0: POR enable, 1: POR disable

E3h.4 **LVRPD**: LVR Power Down

0: LVR enable, 1: LVR disable

E3h.3~0 **LVRSEL**: Low voltage Reset sleect

0000: set LVR at 1.79V

0001: set LVR at 1.95V

0010: set LVR at 2.11V

0011: set LVR at 2.26V

0100: set LVR at 2.40V

0101: set LVR at 2.56V

0110: set LVR at 2.71V

0111: set LVR at 2.87V

1000: set LVR at 3.03V

1001: set LVR at 3.18V

1010: set LVR at 3.32V

1011: set LVR at 3.50V

1100: set LVR at 3.63V

1101: set LVR at 3.80V

1110: set LVR at 3.94V

1111: set LVR at 4.12



## 4. Reset

The Chip has five types of reset methods. Resets can be caused by Power on Reset (POR), External Pin Reset (XRST), Software Command Reset (SWRST), Watchdog Timer Reset (WDTR), or Low Voltage Reset (LVR). The CFGWH controls the Reset functionality. The SFRs are returned to their default value after Reset.

### 4.1 Power on Reset

After Power on Reset, the device stays on Reset state for 40 ms as chip warm up time, then downloads the CFGW register from ROM's last eight bytes. The Power on Reset needs VCC pin's voltage first discharge to near  $V_{SS}$  level, then rise beyond 2.2V. Power-on reset can be enabled/disabled (non-Halt/Stop mode) by PORPD (SFR E3h.5). Under the condition of CFGWH [5] =0, when the chip enters Halt/Stop mode, power-on reset can also be enable/disable by setting PORPD\_SAV (SFR E3h.6).

### 4.2 External Pin Reset

External Pin Reset is active low. It needs to keep at least 2 SRC clock cycle long to be seen by the Chip. External Pin Reset can be disabled or enabled by CFGW.

### 4.3 Software Command Reset

Software Reset is activated by writing the SFR 97h with data 56h.

### 4.4 Watchdog Timer Reset

WDT overflow Reset is disabled or enabled by SFR F7h. The WDT uses SRC as its counting time base. It runs in Fast/Slow mode and runs or stops in Idle/Halt/Stop mode. WDT overflow speed can be defined by WDTOSC SFR. WDT is cleared by device Reset or CLRWDT SFR bit.

### 4.5 Low Voltage Reset

The Chip provides LVR and Low Voltage Detection (LVD) functions. There are 16-level LVR can be selected by LVRCON (E3h.3~0) and 16-level LVD can be selected by SFR LVDS. When PWRSV (SFR F7h.5) = 1, LVR will be disabled when enter IDLE/HALT/STOP mode. The first instruction is to set LVR level after STARTUP in main program.

SFR E3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>LVRCON</b>	–	PORPD_SAV	PORPD	LVRPD	LVRSEL			
R/W	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	1	0	0	0	0	0	0

E3h.6 **PORPD\_SAV**: POR disable at Stop mode

0: POR enable, 1: POR disable

E3h.5 **PORPD**: POR Power Down

0: POR enable, 1: POR disable

E3h.4 **LVRPD**: LVR Power Down

0: LVR enable, 1: LVR disable

E3h.3~0 **LVRSEL**: Low voltage Reset sleect

0000: set LVR at 1.79V

0001: set LVR at 1.95V

0010: set LVR at 2.11V

0011: set LVR at 2.26V

0100: set LVR at 2.40V

0101: set LVR at 2.56V

0110: set LVR at 2.71V

0111: set LVR at 2.87V

1000: set LVR at 3.03V

1001: set LVR at 3.18V

1010: set LVR at 3.32V

1011: set LVR at 3.50V

1100: set LVR at 3.63V

1101: set LVR at 3.80V

1110: set LVR at 3.94V

1111: set LVR at 4.12

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>OPTION</b>	–	TM3CKS	WDTPSC		ADCKS		–	–
R/W	–	R/W	R/W		R/W		–	–
Reset	–	0	0	0	0	0	–	–

94h.5~4 **WDTPSC**: Watchdog Timer pre-scalar time select

00: 220 ms WDT overflow rate

01: 110 ms WDT overflow rate

10: 55 ms WDT overflow rate

11: 27 ms WDT overflow rate

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTFLG</b>	LVDIF	EEPIF	EPPBUSY	ADIF	–	–	PCIF	TF3
R/W	R/W	R/W	R	R/W	–	–	R/W	R/W
Reset	0	0	0	0	–	–	0	0

95h.7 **LVDIF**: Low Voltage Detect interrupt flag

Set by H/W. S/W writes 7Fh to INTFLG to clear this flag.

*Note: S/W can write 0 to clear a flag in the INTFLG, but writing 1 has no effect.*

SFR 97h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SWCMD</b>	IAPEN/SWRST							
R/W	W						R/W	R/W
Reset	-						-	0

97h.7~0 **SWRST**: Write 56h to generate S/W Reset

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSAB	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

F7h.7~6 **WDTE**: Watchdog Timer Reset control

0x: Watchdog Timer Reset disable

10: Watchdog Timer Reset enable in Fast/Slow mode, disable in Idle/Halt/Stop mode

11: Watchdog Timer Reset always enable

F7h.5 **PWRSAB**: chip power-saving option

Set 1 to reduce the chip's power consumption at Idle/Halt/Stop Mode

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX1</b>	CLRWDT	CLRTM3	CLRPWM0	ADSOC	CLRPWM1	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	0	1	0	0	0

F8h.7 **CLRWDT**: Set to clear WDT, H/W auto clear it at next clock cycle

## 5. Clock Circuitry & Operation Mode

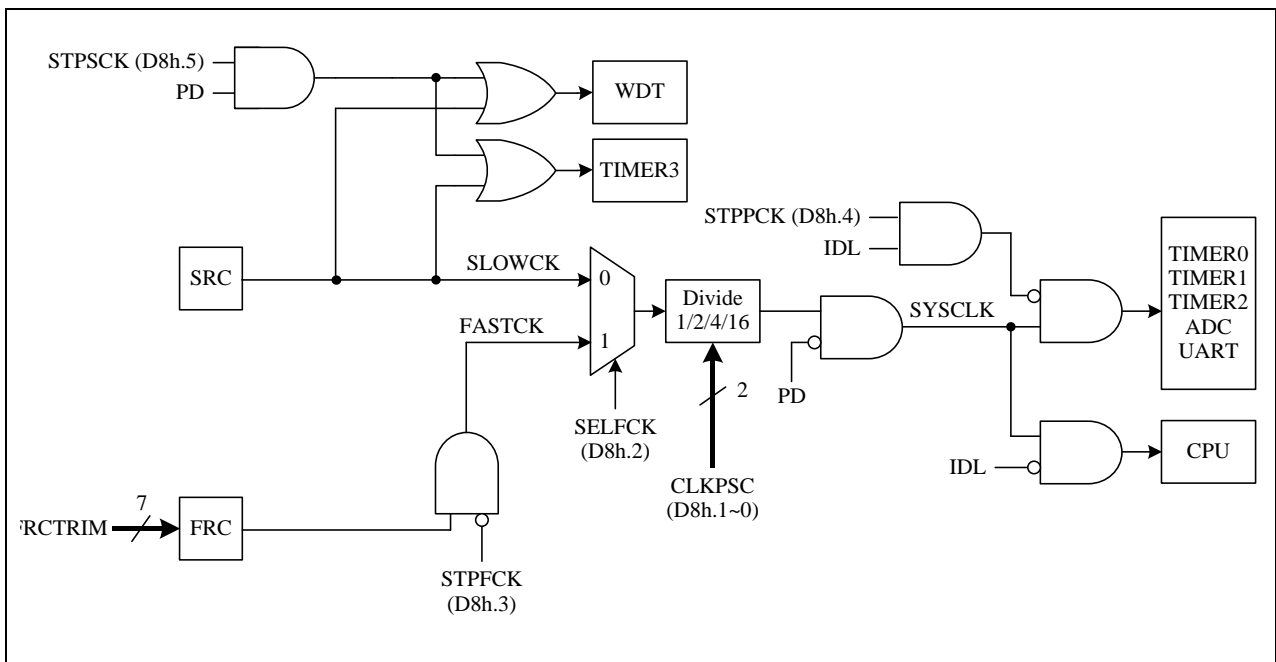
### 5.1 System Clock

The Chip is designed with dual-clock system. During runtime, user can directly switch the System clock from fast to slow or from slow to fast. It also can directly select a clock divider of 1, 2, 4 or 16. The Fast clock is FRC (Fast Internal RC, 18.432 MHz). The Slow clock is SRC (Slow Internal RC, 75 KHz). Fast mode and Slow mode are defined as the CPU running at Fast and Slow clock speeds.

After Reset, the device is running at Slow mode with 75 KHz SRC. S/W should select the proper clock rate for chip operation safety. The higher  $V_{CC}$  allows the chip to run at a higher System clock frequency. In a typical condition, an 18 MHz System clock rate requires  $V_{CC} > 2.4V$ .

The **CLKCON** SFR controls the System clock operating. H/W automatically blocks the S/W abnormally setting for this register. S/W can only change the Slow clock type in Fast mode and change the Fast clock type in Slow mode. Never to write both **STPFCK**=1 & **SELFCK**=1. It is recommended to write this SFR bit by bit.

The chip can also output the "System clock divided by 2" signal (CKO) to P1.4 pin. CKO pin's output setting is controlled by **PINMODE** SFR (*see section 7*).



**Clock Structure**

**Note:** Because of the **CLKPSC** delay, it needs to wait for 16 clock cycles (max.) before switching Slow clock to Fast clock. Also refer to *AP-TM52XXXXX\_01S* and *AP-TM52XXXXX\_02S* about System Clock Application Note.

SYSCLK	CLKCON (D8h)	
	bit3 STPFCK	bit2 SELFCK
Fast FRC	0	1
Slow SRC	0/1	0
Stop FRC	0 → 1	0
Switch to FRC	0	0 → 1
Switch to SRC	0	1 → 0

Flash 3FFDh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>CFGWL</b>	-	FRCTRIM						

**CFGWL FRCTRIM:** FRC frequency adjustment.

FRC is trimmed to 18.432 MHz in chip manufacturing. FRCF records the adjustment data. After Power on, CFGWL will be uploaded to SFR F6h

SFR F6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>FRCF</b>	-	FRCF						
R/W	-	R/W						
Reset	-	-	-	-	-	-	-	-

F6h.6~0 **FRCF:** FRC frequency adjustment

00h= lowest frequency, 7Fh=highest frequency.

SFR D8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>CLKCON</b>	-	-	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
R/W	-	-	R/W	R/W	R/W	R/W	R/W	
Reset	-	-	1	0	0	0	1	1

D8h.5 **STPSCK:** Set 1 to stop Slow clock in PDOWN mode

D8h.4 **STPPCK:** Set 1 to stop UARTs/Timer0/Timer1/Timer2/ADC clock in Idle mode for current reducing. If set, only Timer3 and pin interrupts are alive in Idle Mode.

D8h.3 **STPFCK:** Set 1 to stop Fast clock for power saving in Slow/Idle mode. This bit can be changed only in Slow mode.

D8h.2 **SELFCK:** System clock source selection. This bit can be changed only when STPFCK=0.

0: Slow clock

1: Fast clock

D8h.1~0 **CLKPSC:** System clock prescaler. Effective after 16 clock cycles (Max.) delay.

00: System clock is Fast/Slow clock divided by 16

01: System clock is Fast/Slow clock divided by 4

10: System clock is Fast/Slow clock divided by 2

11: System clock is Fast/Slow clock divided by 1

## 5.2 Operation Modes

There are five operation modes for this device. **Fast Mode** is defined as the CPU running at Fast clock speed. **Slow Mode** is defined as the CPU running at Slow clock speed. When the System clock speed is lower, the power consumption is lower.

**Idle Mode** is entered by setting the IDL bit in PCON SFR. Both Fast and Slow clock can be set as the System clock source in Idle Mode, but Slow clock is better for power saving. In Idle mode, the CPU puts itself to sleep while the on-chip peripherals stay active. The “STPPCK” bit in CLKCON SFR can be set to furthermore reduce Idle mode current. If STPPCK is set, only Timer3 and pin interrupts are alive in Idle Mode, others peripherals such as Timer0/1/2, UARTs and ADC are stop. The slower System clock rate also helps current saving. It can be achieved by setup the CLKPSC SFR to divide System clock frequency. Idle mode is terminated by Reset or enabled Interrupts wake up.

**Stop Mode** is entered by setting the PD bit in PCON SFR and STPSCK is set. This mode is the so-called “Power Down” mode in standard 8051. In Stop mode, all clocks stop except the WDT could be alive if it is enabled. Stop Mode is terminated by Reset or pin wake up.

**Halt Mode** is entered by setting the PD bit in PCON SFR and STPSCK is cleared. In Halt mode, all clocks stop except the Timer3 and WDT could be alive if they are enabled. Halt Mode is terminated by Reset, pin wake up or Timer3 interrupt. In this mode, Timer3 clock source can only choose Slow clock, not FRC/512.

*Note: Chip cannot enter Halt/Stop Mode if INTn pin is low and wakeup is enabled. (INTn=0 and EXn=1, n=0~1)*

*Note: FW must turn off Bandgap to obtain Tiny Current (VBGOUT=0)*

SFR 87h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PCON</b>	SMOD	–	–	–	GF1	GF0	PD	IDL
R/W	R/W	–	–	–	R/W	R/W	R/W	R/W
Reset	0	–	–	–	0	0	0	0

87h.1 **PD:** Power down control bit, set 1 to enter Halt/Stop mode.

87h.0 **IDL:** Idle mode control bit, set 1 to enter Idle mode.

SFR D8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>CLKCON</b>	-	-	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
R/W	-	-	R/W	R/W	R/W	R/W	R/W	
Reset	-	-	1	0	0	0	1	1

D8h.5 **STPSCK:** Set 1 to stop Slow clock in PDOWN mode

D8h.4 **STPPCK:** Set 1 to stop UART/Timer0/Timer1/Timer2/ADC clock in Idle mode for current reducing. If set, only Timer3 and pin interrupts are alive in Idle Mode.

D8h.3 **STPFCK:** Set 1 to stop Fast clock for power saving in Slow/Idle mode. This bit can be changed only in Slow mode.

D8h.2 **SELFCK:** System clock source selection. This bit can be changed only when STPFCK=0.  
 0: Slow clock  
 1: Fast clock

D8h.1~0 **CLKPSC:** System clock prescaler. Effective after 16 clock cycles (Max.) delay.

00: System clock is Fast/Slow clock divided by 16

01: System clock is Fast/Slow clock divided by 4

10: System clock is Fast/Slow clock divided by 2

11: System clock is Fast/Slow clock divided by 1

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>OPTION</b>	–	TM3CKS	WDTPSC		ADCKS		–	–
R/W	–	R/W	R/W		R/W		–	–
Reset	–	0	0	0	0	0	–	–

94h.6 **TM3CKS:** Timer3 Clock Source select  
 0: Slow clock (SRC)  
 1: FRC/512 (36KHz)

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSVAV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	1	1	0

F7h.4 **VBGOUT:** Bandgap voltage output to P3.2  
 0: Disable  
 1: Enable

## 6. Interrupt & Wake-up

This Chip has a 14-source four-level priority interrupt structure. Only the Pin Interrupts can wake up CPU from Halt/Stop mode. Each interrupt source has its own enable control bit. An interrupt event will set its individual Interrupt Flag, no matter whether its interrupt enable control bit is 0 or 1. The Interrupt vectors and flags are list below.

INT Vector	NT Num C51	Flag	Description
0003	0	IE0	INT0 external pin Interrupt (can wake up Halt/Stop mode)
000B	1	TF0	Timer0 Interrupt
0013	2	IE1	INT1 external pin Interrupt (can wake up Halt/Stop mode)
001B	3	TF1	Timer1 Interrupt
0023	4	RI+TI	Serial Port (UART1) Interrupt
002B	5	TF2+EXF2	Timer2 Interrupt
0033	6	–	Reserved for ICE mode use
003B	7	TF3	Timer3 Interrupt
0043	8	PCIF	Port0~Port3 external pin change Interrupt (can wake up Halt/Stop mode)
004B	9	LVDIF	LVD interrupt
0053	10	ADIF	ADC Interrupt
005B	11	EEPIF	EED Write Finish Interrupt
0063	12	RI2+TI2	Serial Port (UART2) Interrupt
006B	13	MIIF	Master I <sup>2</sup> C interrupt
0073	14	PWM0IF + PWM1IF	PWM0~ PWM1 Interrupt

**Interrupt Vector & Flag**

INT Vector	INT Num C51	Flag	INT Enable	Sub INT enable	INT Flag
0003	0	IE0	IE A8.0		TCON 88.1
000B	1	TF0	IE A8.1		TCON 88.5
0013	2	IE1	IE A8.2		TCON 88.3
001B	3	TF1	IE A8.3		TCON 88.7
0023	4	RI+TI	IE A8.4		SCON 98.1~0
002B	5	TF2+EXF2	IE A8.5		T2CON C8.7~6
0033	6	–			
003B	7	TF3	INTE1 A9.0		INTFLG 95.0
0043	8	PCIF	INTE1 A9.1		INTFLG 95.1
004B	9	LVDIF	INTE1 A9.2		INTFLG 95.7
0053	10	ADIF	INTE1 A9.3		INTFLG 95.4
005B	11	EEPIF	INTE1 A9.4		INTFLG 95.6
0063	12	RI2+TI2	INTE1 A9.5		SCON 8E.1~0
006B	13	MIIF	INTE1 A9.6		MICON E1.5
0073	14	PWM0IF+ PWM1IF	INTE1 A9.7	INTE2 84.0 INTE2 84.1	INTPWM 86.0 INTPWM 86.1

Interrupt related SFR



## 6.1 Interrupt Enable and Priority Control

The IE and INTE1 SFRs decide whether the pending interrupt is serviced by CPU. The IP, IPH, IP1 and IP1H SFRs decide the interrupt priority. An interrupt will be serviced as long as an interrupt of equal or higher priority is not already being serviced. If an interrupt of equal or higher level priority is being serviced, the new interrupt will wait until it is finished before being serviced. If a lower priority level interrupt is being serviced, it will be stopped and the new interrupt serviced. When the new interrupt is finished, the lower priority level interrupt that was stopped will be completed.

## 6.2 Suggestions on interrupting subroutines

When entering the interrupt program, in addition to the traditionally known SFR A or PSW that should be PUSH, POP, some SFRs used for indexing should also be added to the ranks of PUSH POP, such as PORTIDX. To avoid writing and reading these SFRs before and after the interruption may cause inconsistencies. In addition, PWMDH, PWMDL, PWMPRDH or PWMPRDL is a 16-bit operation, and the program should avoid interrupts when writing and reading the high byte and low byte. If you are reading and writing these 16-bit SFRs in the meantime an interrupt occurs. And these SFRs are read and written in the interrupt. It is easy to cause read and write errors. For the 16-bit PWM period and duty to read and write, it is recommended to update the data only in the main program, or update the data only in the interrupt to avoid possible errors.

SFR 84h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE2	-	-	-	-	-	-	PWM1IE	PWM0IE
R/W	-	-	-	-	-	-	R/W	R/W
Reset	-	-	-	-	-	-	0	0

84h.1 **PWM1IE:** PWM1 Interrupt Enable

0: disable

1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)

84h.0 **PWM0IE:** PWM0 Interrupt Enable

0: disable

1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)

SFR A8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IE</b>	EA	-	ET2	ES	ET1	EX1	ET0	EX0
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	-	0	0	0	0	0	0

A8h.7 **EA:** Global interrupt enable control.

0: Disable all Interrupts.

1: Each interrupt is enabled or disabled by its individual interrupt control bit

A8h.5 **ET2:** Timer2 interrupt enable

0: Disable Timer2 interrupt

1: Enable Timer2 interrupt

A8h.4 **ES:** Serial Port (UART1) interrupt enable

0: Disable Serial Port (UART1) interrupt

1: Enable Serial Port (UART1) interrupt

A8h.3 **ET1:** Timer1 interrupt enable

0: Disable Timer1 interrupt

1: Enable Timer1 interrupt

A8h.2 **EX1:** External INT1 pin Interrupt enable and Halt/Stop mode wake up enable

0: Disable INT1 pin Interrupt and Halt/Stop mode wake up

1: Enable INT1 pin Interrupt and Halt/Stop mode wake up, it can wake up CPU from Halt/Stop mode no matter EA is 0 or 1.

A8h.1 **ET0:** Timer0 interrupt enable

0: Disable Timer0 interrupt

1: Enable Timer0 interrupt

- A8h.0 **EX0:** External INT0 pin Interrupt enable and Halt/Stop mode wake up enable  
 0: Disable INT0 pin Interrupt and Halt/Stop mode wake up  
 1: Enable INT0 pin Interrupt and Halt/Stop mode wake up, it can wake up CPU from Halt/Stop mode no matter EA is 0 or 1.

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTE1</b>	PWMIE	I2CE	ES2	EEPIE	ADIE	LVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- A9h.7 **PWMIE:** PWM0~PWM2 interrupt enable  
 0: Disable PWM0~PWM2 interrupt  
 1: Enable PWM0~PWM2 interrupt
- A9h.6 **I2CE:** I<sup>2</sup>C (master/slave) interrupt enable  
 0: Disable I<sup>2</sup>C interrupt  
 1: Enable I<sup>2</sup>C interrupt
- A9h.5 **ES2:** Serial Port (UART2) interrupt enable  
 0: Disable Serial Port (UART2) interrupt  
 1: Enable Serial Port (UART2) interrupt
- A9h.4 **EEPIE:** EEP write finish interrupt enable  
 0: Disable EEP write finish interrupt  
 1: Enable EEP write finish interrupt
- A9h.3 **ADIE:** ADC interrupt enable  
 0: Disable ADC interrupt  
 1: Enable ADC interrupt
- A9h.2 **LVDIE:** LVD interrupt enable  
 0: Disable LVD interrupt  
 1: Enable LVD interrupt.
- A9h.1 **PCIE:** Port0~Port3 pin change interrupt enable. This bit does not affect Halt/Stop mode wake up capability.  
 0: Disable Port0~Port3 pin change interrupt  
 1: Enable Port0~Port3 pin change interrupt
- A9h.0 **TM3IE:** Timer3 interrupt enable  
 0: Disable Timer3 interrupt  
 1: Enable Timer3 interrupt

SFR B9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IPH</b>	–	–	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

SFR B8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IP</b>	–	–	PT2	PS	PT1	PX1	PT0	PX0
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

- B9h.5, B8h.5 **PT2H, PT2:** Timer2 Interrupt Priority control. (PT2H, PT2) =  
 11: Level 3 (highest priority)  
 10: Level 2  
 01: Level 1  
 00: Level 0 (lowest priority)
- B9h.4, B8h.4 **PSH, PS:** Serial Port (UART1) Interrupt Priority control. Definition as above.
- B9h.3, B8h.3 **PT1H, PT1:** Timer1 Interrupt Priority control. Definition as above.
- B9h.2, B8h.2 **PX1H, PX1:** External INT1 pin Interrupt Priority control. Definition as above.

B9h.1, B8h.1 **PT0H, PT0:** Timer0 Interrupt Priority control. Definition as above.

B9h.0, B8h.0 **PX0H, PX0:** External INT0 pin Interrupt Priority control. Definition as above.

SFR BbH	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IP1H</b>	PPWMH	PI2CH	PS2H	PEEPH	PADIH	PLVDH	PPCH	PT3H
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR BAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IP1</b>	PPWM	PI2C	PS2	PEEP	PADI	PLVD	PPC	PT3
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

BBh.7, BAh.7 **PPWMH, PPWM:** PWM0~PWM2 Interrupt Priority control. Definition as above.

BBh.6, BAh.6 **PI2CH, PI2C:** I<sup>2</sup>C (Master/Slave) Interrupt Priority control. Definition as above.

BBh.5, BAh.5 **PS2H, PS2:** Serial Port (UART2) Interrupt Priority control. Definition as above.

BBh.4, BAh.4 **PEEPH, PEEP:** EEP write finish Interrupt Priority control. Definition as above.

BBh.3, BAh.3 **PADIH, PADI:** ADC Interrupt Priority control. Definition as above.

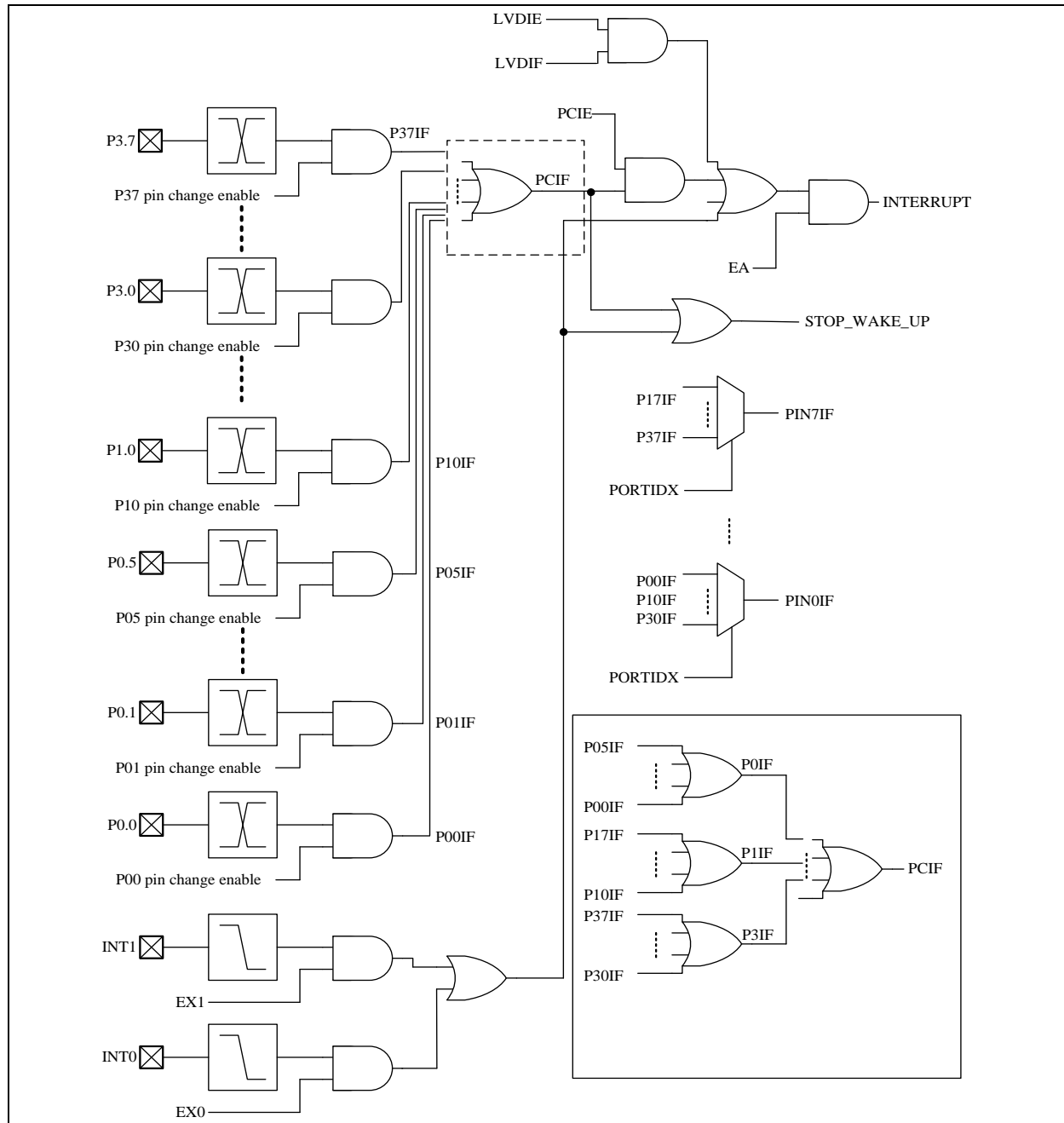
BBh.2, BAh.2 **PLVDH, PLVD:** LVD Interrupt Priority control. Definition as above.

BBh.1, BAh.1 **PPCH, PPC:** Port0~ Port 3 Pin Change Interrupt Priority control. Definition as above.

BBh.0, BAh.0 **PT3H, PT3:** Timer3 Interrupt Priority control. Definition as above.

### 6.3 Pin Interrupt and LVD interrupt

Pin Interrupts include INT0~INT1 and Port0~Port3 pin change interrupt. INT0~INT1 and Port0~Port3 pin change also have the Halt/Stop mode wake up capability. INT0 and INT1 are falling edge or low level triggered as the 8051 standard. Port0~Port3 Pin Change Interrupt is triggered by IO state change. Pin change enable are setting by PINMOD10/PINMOD32/PINMOD54/PINMOD76. For details, see Chapter 7. PINMODE and pin change enable settings. LVD interrupt can be used to detect the  $V_{CC}$  voltage level and generate an interrupt.



**Pin interrupt/Wake up & LVD interrupt**

**Note:** Chip cannot enter Halt/Stop Mode if  $INTn$  pin is low and wakeup is enabled. ( $INTn=0$  and  $EXn=1$ ,  $n=0\sim1$ )

SFR 85h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTPORT</b>	–	–	–	–	P3IF	–	P1IF	P0IF
R/W	–	–	–	–	R/W	–	R/W	R/W
Reset	–	–	–	–	0	–	0	0

96h.3 **P3IF**: P3.7~P3.0 pin change interrupt flag, Write 0 to clear P3.7~P3.0 pin change interrupt flag

96h.1 **P1IF**: P1.7~P1.0 pin change interrupt flag, Write 0 to clear P1.7~P1.0 pin change interrupt flag

96h.0 **P0IF**: P0.7~P0.0 pin change interrupt flag, Write 0 to clear P0.7~P0.0 pin change interrupt flag

SFR 91h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PORTIDX</b>	–	–	–	–	–	–	PORTIDX	
R/W	–	–	–	–	–	–	R/W	
Reset	–	–	–	–	–	–	0	0

91h.1~0 **PORTIDX**: Port index of INTPIN, PINMOD10, PINMOD32, PINMOD54, PINMOD76

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTFLG</b>	LVDIF	EEPIF	EPPBUSY	ADIF	–	–	PCIF	TF3
R/W	R	R/W	R	R/W	–	–	R/W	R/W
Reset	–	0	0	0	–	–	0	0

95h.7 **LVDIF**: Low Voltage Detect interrupt flag

Set by H/W. S/W writes 7Fh to INTFLG to clear this flag.

95h.1 **PCIF**: Port0~Port3 Pin change interrupt flag

Set by H/W when Port0~Port3 pin state change is detected and its interrupt enable bit is set.

S/W can write 0 to clear all pin change interrupt flags (Port0~Port3), it will also clear PIN0IF~PIN7IF and P0IF~P3IF.

*Note: S/W can write 0 to clear a flag in the INTFLG, but writing 1 has no effect.*

SFR 96h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTPIN</b>	PIN7IF	PIN6IF	PIN5IF	PIN4IF	PIN3IF	PIN2IF	PIN1IF	PIN0IF
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

96h.7 **PIN7IF**: Px.7 pin change interrupt flag, Write 0 to clear Px.7 pin change interrupt flag  
port number (x) define by PORTIDX

96h.6 **PIN6IF**: Px.6 pin change interrupt flag, Write 0 to clear Px.6 pin change interrupt flag  
port number (x) define by PORTIDX

96h.5 **PIN5IF**: Px.5 pin change interrupt flag, Write 0 to clear Px.5 pin change interrupt flag  
port number (x) define by PORTIDX

96h.4 **PIN4IF**: Px.4 pin change interrupt flag, Write 0 to clear Px.4 pin change interrupt flag  
port number (x) define by PORTIDX

96h.3 **PIN3IF**: Px.3 pin change interrupt flag, Write 0 to clear Px.3 pin change interrupt flag  
port number (x) define by PORTIDX

96h.2 **PIN2IF**: Px.2 pin change interrupt flag, Write 0 to clear Px.2 pin change interrupt flag  
port number (x) define by PORTIDX

96h.1 **PIN1IF**: Px.1 pin change interrupt flag, Write 0 to clear Px.1 pin change interrupt flag  
port number (x) define by PORTIDX

96h.0 **PIN0IF**: Px.0 pin change interrupt flag, Write 0 to clear Px.0 pin change interrupt flag  
port number (x) define by PORTIDX

SFR 88h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TCON</b>	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- 88h.3 **IE1:** External Interrupt 1 (INT1 pin) edge flag.  
Set by H/W when an INT1 pin falling edge is detected, no matter the EX1 is 0 or 1.  
It is cleared automatically when the program performs the interrupt service routine.
- 88h.2 **IT1:** External Interrupt 1 control bit  
0: Low level active (level triggered) for INT1 pin  
1: Falling edge active (edge triggered) for INT1 pin
- 88h.1 **IE0:** External Interrupt 0 (INT0 pin) edge flag  
Set by H/W when an INT0 pin falling edge is detected, no matter the EX0 is 0 or 1.  
It is cleared automatically when the program performs the interrupt service routine.
- 88h.0 **IT0:** External Interrupt 0 control bit  
0: Low level active (level triggered) for INT0 pin  
1: Falling edge active (edge triggered) for INT0 pin

SFR A8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IE</b>	EA	–	ET2	ES	ET1	EX1	ET0	EX0
R/W	R/W	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	–	0	0	0	0	0	0

- A8h.7 **EA:** Global interrupt enable control.  
0: Disable all Interrupts.  
1: Each interrupt is enabled or disabled by its individual interrupt control bit
- A8h.2 **EX1:** External INT1 pin Interrupt enable and Halt/Stop mode wake up enable  
0: Disable INT1 pin Interrupt and Halt/Stop mode wake up  
1: Enable INT1 pin Interrupt and Halt/Stop mode wake up, it can wake up CPU from Halt/Stop mode no matter EA is 0 or 1.
- A8h.0 **EX0:** External INT0 pin Interrupt enable and Halt/Stop mode wake up enable  
0: Disable INT0 pin Interrupt and Halt/Stop mode wake up  
1: Enable INT0 pin Interrupt and Halt/Stop mode wake up, it can wake up CPU from Halt/Stop mode no matter EA is 0 or 1.

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTE1</b>	PWMIE	I2CE	ES2	EEPIE	ADIE	LVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- A9h.2 **LVDIE:** LVD interrupt enable  
0: Disable LVD interrupt  
1: Enable LVD interrupt.
- A9h.1 **PCIE:** Port0~3 pin change interrupt enable. This bit does not affect Halt/Stop mode wake up capability.  
0: Disable Port0~3 pin change interrupt  
1: Enable Port0~3 pin change interrupt

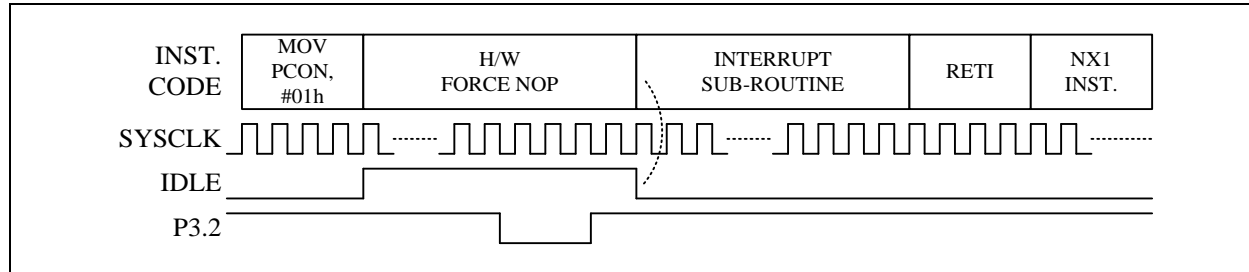
SFR BFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>LVDS</b>	LVDPD	LVDO	–	LVDPD	LVDSSEL			
R/W	R/W	R	–	R/W	R/W	R/W	R/W	R/W
Reset	0	0	–	0	0	0	0	0

BFh.3~0 **LVDSSEL:** Low Voltage Detect select

- 0000: Set LVD at 1.79V
- 0001: Set LVD at 1.95V
- 0010: Set LVD at 2.11V
- 0011: Set LVD at 2.26V
- 0100: Set LVD at 2.40V
- 0101: Set LVD at 2.56V
- 0110: Set LVD at 2.71V
- 0111: Set LVD at 2.87V
- 1000: Set LVD at 3.03V
- 1001: Set LVD at 3.18V
- 1010: Set LVD at 3.32V
- 1011: Set LVD at 3.50V
- 1100: Set LVD at 3.63V
- 1101: Set LVD at 3.80V
- 1110: Set LVD at 3.94V
- 1111: Set LVD at 4.12V

### 6.4 Idle mode Wake up and Interrupt

Idle mode is waked up by enabled Interrupts, which means individual interrupt enable bit (ex: EX0) and EA bit must be both set to 1 to establish Idle mode wake up capability. All enabled Interrupts change (INT0~INT1, Timers, PWM, ADC, and UARTs) can wake up CPU from Idle mode. Upon Idle wake-up, Interrupt service routine is entered immediately. “The first instruction behind IDL (PCON.0) setting” is executed after interrupt service routine return.



**EA=EX0=1, Idle mode wake-up and Interrupt by P3.2 (INT0)**

SFR 87h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PCON</b>	SMOD	–	–	–	GF1	GF0	PD	IDL
R/W	R/W	–	–	–	R/W	R/W	R/W	R/W
Reset	0	–	–	–	0	0	0	0

87h.1 **PD:** Power down control bit, set 1 to enter Halt/Stop mode.

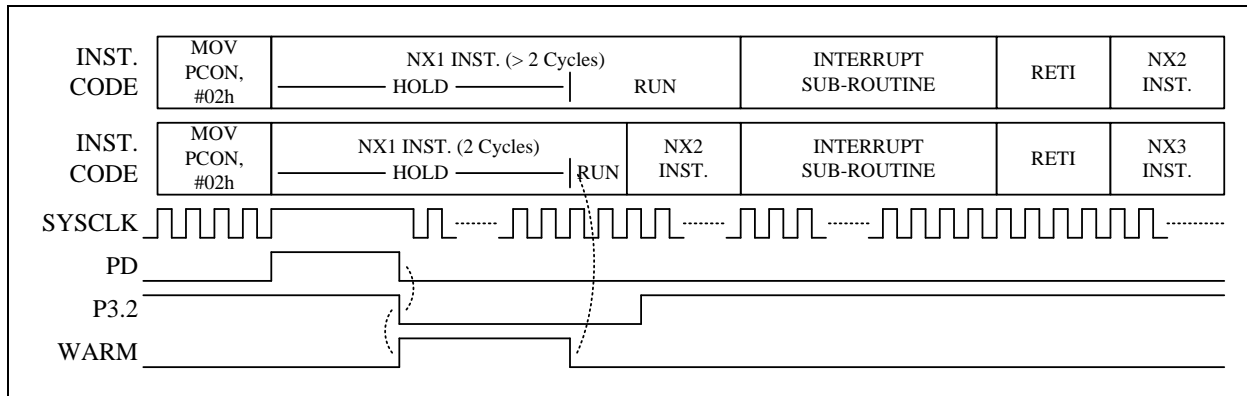
87h.0 **IDL:** Idle mode control bit, set 1 to enter Idle mode.

### 6.5 Halt/Stop mode Wake up and Interrupt

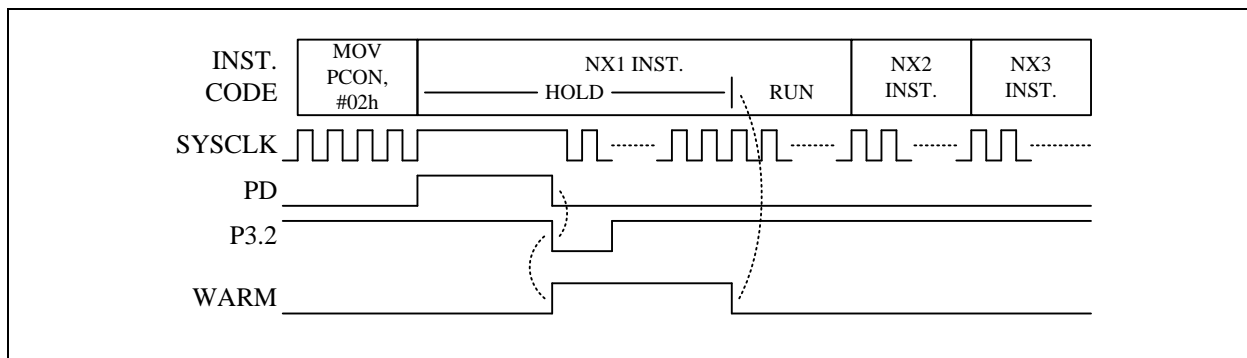
Halt/Stop mode wake up is simple, as long as the individual pin interrupt enable bit (ex: EX0) is set, the pin wake up capability is asserted. Set EX0/EX1 can enable INT0/INT1 pins’ Halt/Stop mode wake up capability. Set PINMOD10/PINMOD32/PINMOD54/PINMOD76 can enable Port0~Port3 Halt/Stop mode wake up capability. Upon Halt/Stop wake up, “the first instruction behind PD setting (PCON.1)” is executed immediately before Interrupt service. Interrupt entry requires EA=1 and trigger state of the pin staying sufficiently long to be observed by the System clock. This feature allows CPU to enter or not enter Interrupt sub-routine after Halt/Stop mode wake up.

*Note: It is recommended to place the NX1/NX2 with NOP Instruction in figures below.*

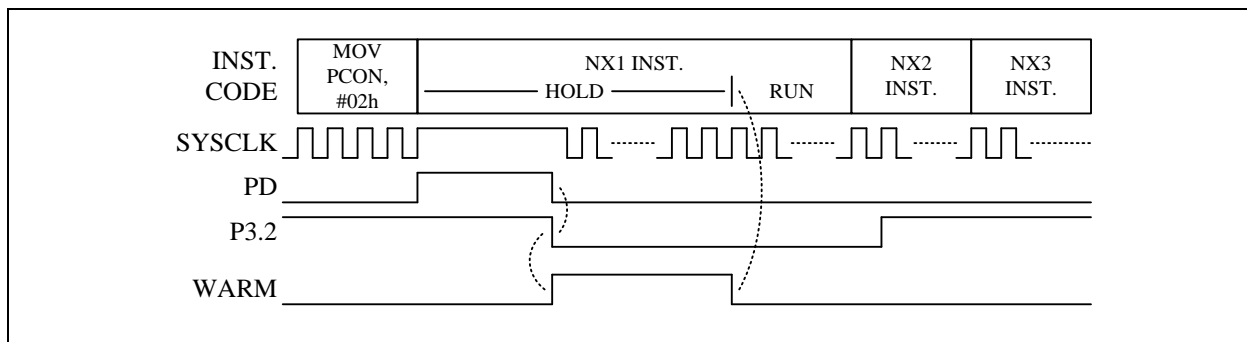




**EA=EX0=1, P3.2 (INT0) is sampled after warm-up, Halt/Stop mode wake-up and Interrupt**



**EA=EX0=1, Halt/Stop mode wake-up but not Interrupt. P3.2 (INT0) pulse too narrow**



**EX0= 1, EA=0, P3.2 (INT0) Halt/Stop mode wake-up but not Interrupt**

## 7. I/O Ports

The Chip has total 22 multi-function I/O pins. All I/O pins follow the standard 8051 “Read-Modify-Write” feature. The instructions that read the SFR rather than the Pin State are the ones that read a port or port bit value, possibly change it, and then rewrite it to the SFR (ex: ANL P1, A; INC P2; CPL P3.0).

When entering the interrupt program, in addition to the traditionally known SFR A or PSW that should be PUSH, POP, some SFRs used for indexing should also be added to the ranks of PUSH POP, such as PORTIDX. To avoid writing and reading these SFRs before and after the interruption may cause inconsistencies.

### 7.1 Port0~Port 3

These pins can operate in four different modes as below.

PINMOD76 PINMOD54 PINMOD32 PINMOD10					Pin State	Interrupt	Wake-up
MODE0	0	0	0	0	Open Drain with pull-up (for INT0/INT1)	Y	Y
MODE1	0	0	0	1	Open Drain ( <b>Default</b> ) (for INT0/INT1)	Y	Y
MODE2	0	0	1	0	CMOS Output	-	-
MODE3	0	0	1	1	ADC channel	-	-
MODE4	0	1	0	0	Open Drain with pull-down	Y	Y
MODE5	0	1	0	1	Open Drain	Y	Y
MODE6	0	1	1	0	CMOS Output	-	-
MODE7	0	1	1	1	-	-	-
MODE8	1	0	0	0	Open Drain with pull-up (for pin change from Halt/Stop)	Y	Y
MODE9	1	0	0	1	Open Drain (for pin change from Halt/Stop)	Y	Y
MODE10	1	0	1	0	CMOS Output	-	-
MODE11	1	0	1	1	PWMO	-	-
MODE12	1	1	0	0	Open Drain with pull-down (for pin change from Halt/Stop)	Y	Y
MODE13	1	1	0	1	Open Drain (for pin change from Halt/Stop)	Y	Y
MODE14	1	1	1	0	CMOS Output	-	-
MODE15	1	1	1	1	LCD 1/2 Vcc bias	-	-

**Table 7.1 Port0~Port3 I/O Pin Function Table**

PINMOD76/ PINMOD54/PINMOD32/PINMOD10 need PORTIDX to index the corresponding IO port.

The chip does not support Port 2, PORTIDX cannot be set to 2

For example:

If PORTIDX=0, PINMOD10 is set to P0.1 and P0.0, high 4 bits are set to P0.1, low 4 bits are set to P0.0

If PORTIDX=1, PINMOD10 is set to P1.1 and P1.0, high 4 bits are set to P1.1, low 4 bits are set to P1.0

If PORTIDX=3, PINMOD10 is set to P3.1 and P3.0, high 4 bits are set to P3.1, low 4 bits are set to P3.0

If PORTIDX=0, PINMOD32 is set to P0.3 and P0.2, high 4 bits are set to P0.3, low 4 bits are set to P0.2

...

If PORTIDX=3, PINMOD76 is set to P3.7 and P3.6, high 4 bits are set to P3.7, low 4 bits are set to P3.6

Mode	Port0~Port3 pin function	Px.n SFR data	Pin State	Resistor Pull-up	Resistor Pull-down	Digital Input
<b>MODE0</b> <b>MODE8</b>	Open Drain with pull-up	0	Drive Low	N	N	N
		1	Pull-up	Y	N	Y
<b>MODE4</b> <b>MODE12</b>	Open Drain with pull-down	0	Drive Low	N	N	N
		1	Pull-down	N	Y	Y
<b>MODE1</b> <b>MODE5</b> <b>MODE9</b> <b>MODE13</b>	Open Drain	0	Drive Low	N	N	N
		1	Hi-Z	N	N	Y
<b>MODE2</b> <b>MODE6</b> <b>MODE10</b> <b>MODE14</b>	CMOS Output	0	Drive Low	N	N	N
		1	Drive High	N	N	N
<b>MODE3</b>	ADC channel	X (don't care)	-	N	N	N
<b>MODE7</b>	-	-	-	N	N	N
<b>MODE11</b>	PWMO	X (don't care)	-	N	N	N
<b>MODE15</b>	LCD 1/2 Vcc bias output	X (don't care)	-	Y	Y	N

**I/O Pin Function Table**

If a Port0~Port3 pin is used for Schmitt-trigger input, S/W must set the I/O pin to MODE0, MODE1, MODE4, MODE5, MODE8, MODE9, MODE12 or MODE13 (Open Drain, Open Drain with pull-up or Open Drain with pull-down), and set the corresponding Port Data SFR to 1 to disable the pin's output driving circuitry.

Beside I/O port function, each Port0~Port3 has one or more alternative functions, such as ADC and LCD. Most of the functions are activated by setting the individual pin mode control SFR to MODE3, MODE7, MODE11 or MODE15. Port1/Port3 pins have standard 8051 auxiliary definition such as INT0/INT1, T0/T1/T2, or RXD/TXD. These pin functions need to set the pin mode SFR to MODE0, MODE1, MODE5, MODE8, MODE9 or MODE13 (Open Drain or Open Drain with pull-up), and keep the P1.n/P3.n SFR at 1.

Pin Name	Wake-up Interrupt	CKO	ADC	LCD	PWM	UART	I <sup>2</sup> C	others
P0.5	Y		AD05	Y	PWM5			
P0.4	Y		AD04	Y	PWM4			
P0.3	Y		AD03	Y	PWM3	TXD2 (RXD2)		
P0.2	Y		AD02	Y	PWM2	RXD2 (TXD2)		
P0.1	Y		AD01	Y	PWM0P	TXD2 (RXD2)	SDA	PSDA
P0.0	Y		AD00	Y	PWM0N	RXD2 (TXD2)	SCL	PSCL

**Port0 multi-function Table**

Pin Name	Wake-up Interrupt	CKO	ADC	LCD	PWM	UART	I <sup>2</sup> C	others
P1.7	Y		AD20	Y	PWM1P			
P1.6	Y		AD19	Y	PWM0P			
P1.5	Y		AD16	Y	PWM0N			
P1.4	Y	CKO	AD15	Y	PWM1N			
P1.3	Y		AD14	Y	PWM5			
P1.2	Y		AD13	Y	PWM4			
P1.1	Y		AD11	Y	PWM3			T2EX
P1.0	Y	T2O	AD10	Y	PWM2			T2

**Port1 multi-function Table**

Pin Name	Wake-up Interrupt	CKO	ADC	LCD	PWM	UART	I <sup>2</sup> C	others
P3.7	Y		AD06	Y	PWM1P			RSTn
P3.6	Y		AD07	Y	PWM0P	TXD2 (RXD2)		
P3.5	Y		AD08	Y	PWM0N	RXD2 (TXD2)		T1
P3.4	Y	T0O	AD09	Y	PWM1N			T0
P3.3	Y		AD21	Y	PWM0P	TXD (RXD)		INT1
P3.2	Y		AD22	Y	PWM0N	RXD (TXD)		INT0 VBGO
P3.1	Y		AD18	Y		TXD (RXD)	SDA	PSDA
P3.0	Y		AD17	Y		RXD (TXD)	SCL	PSCL

**Port3 multi-function Table**

The necessary SFR setting for Port0~Port3 pin's alternative function is list below.

Alternative Function	PINMOD <sub>xx</sub>	Px.n SFR data	Pin State	Other necessary SFR setting
INT0, INT1	<b>0000</b>	1	Input with Pull-up	
	<b>0001</b>	1	Input	
T0, T1, T2, T2EX	<b>x000</b>	1	Input with Pull-up	
	<b>xx01</b>	1	Input	
RXD RXD2	<b>x000</b>	1	UART RX (Input with Pull-up)	UARTCON
	<b>xx01</b>	1	UART RX (Input)	
TXD TXD2	<b>x000</b>	1	UART TX output (Open Drain Output, Pull-up)	
	<b>xx01</b>	1	UART TX output (Open Drain Output)	
VBGO	<b>0011</b>	X	Bandgap Voltage output	VBGOUT
AD00~ AD22	<b>0011</b>	X	ADC Channel	ADCHS
LCD	<b>1111</b>	X	LCD 1/2 Vcc bias Output	
T0O CKO T2O	<b>1011</b>	X	Clock Output (CMOS Push-Pull)	T0OE TCOE T2OE
PWM0P~PWM0N PWM1P~PWM1P PWM2~PWM5	<b>1011</b>	X	PWM Output (CMOS Push-Pull)	
I <sup>2</sup> C Master SCL	<b>0000</b>	X	I <sup>2</sup> C Clock Output (Open Drain Output, Pull-up)	I2CPS
	<b>xx10</b>	X	I <sup>2</sup> C Clock Output (CMOS Push-Pull)	
I <sup>2</sup> C Master SDA	<b>0000</b>	1	I <sup>2</sup> C DATA (Pull-up)	

**Mode Setting for Port0 ~ Port3 Alternative Function**

For tables above, a “**CMOS Output**” pin means it can sink and drive at least 4 mA current. It is not recommended to use such pin as input function.

An “**Open Drain**” pin means it can sink at least 4 mA current but only drive a small current (<20 μA). It can be used as input or output function and typically needs an external pull up resistor.

The chip also supports I/O High-sink function. It is an option. For efficient control, we divide the High-sink pins into three groups (Group 0: P0.0~P0.5; Group 1: P1.0~P1.7; Group 2: P3.0~P3.3). It is enabled by setting SFR HSNK0EN, HSNK1EN and HSNK2EN.

SFR 80h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>P0</b>	-	-	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	1	1	1	1	1	1

80h.7~0 **P0**: Port0 data

SFR 90h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>P1</b>	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

90h.7~0 **P1**: Port1 data

SFR B0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>P3</b>	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

B0h.7~0 **P3**: Port3 data

SFR 91h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PORTIDX</b>	–	–	–	–	–	–	PORTIDX	
R/W	–	–	–	–	–	–	R/W	
Reset	–	–	–	–	–	–	0	0

91h.1~0 **PORTIDX**: Port index of INTPIN, PINMOD10, PINMOD32, PINMOD54, PINMOD76

00: Port 0

01: Port 1

10: Reserved

11: Port 3

SFR 93h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>UARTCON</b>	UART2PS				–	UARTPS		
R/W	R/W				–	R/W		
Reset	0	0	0	1	–	0	0	1

93h.7~4 **UART2PS**: UART2 Pin select

0000: RXD2/TXD2 = P0.0/P0.1

0001: RXD2/TXD2 = P3.5/P3.6

0010: RXD2/TXD2 = P0.1/P0.0

0011: RXD2/TXD2 = P3.6/P3.5

0100: RXD2/TXD2 = P0.1/P0.1, 1-wire mode

0101: RXD2/TXD2 = P3.6/P3.6, 1-wire mode

0110: RXD2/TXD2 = P0.0/P0.0, 1-wire mode

0111: RXD2/TXD2 = P3.5/P3.5, 1-wire mode

1000: RXD2/TXD2 = P0.2/P0.3

1010: RXD2/TXD2 = P0.3/P0.2

1100: RXD2/TXD2 = P0.3/P0.3, 1-wire mode

1110: RXD2/TXD2 = P0.2/P0.2, 1-wire mode

93h.2~0 **UARTPS**: UART Pin select

0000: RXD/TXD = P3.0/P3.1

0001: RXD/TXD = P3.2/P3.3

0010: RXD/TXD = P3.1/P3.0

0011: RXD/TXD = P3.3/P3.2

0100: RXD/TXD = P3.1/P3.1, 1-wire mode

0101: RXD/TXD = P3.3/P3.3, 1-wire mode

0110: RXD/TXD = P3.0/P3.0, 1-wire mode

0111: RXD/TXD = P3.2/P3.2, 1-wire mode

SFR A2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD10</b>	PINMOD1				PINMOD0			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A2h.7~4 **PINMOD1**: Px.1 pin control, port index (x) is defined by PORTIDX

0000~1111: see table 7.1

A2h.3~0 **PINMOD0**: Px.0 pin control, port index (x) is defined by PORTIDX

0000~1111: see table 7.1

SFR A3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD32</b>	PINMOD3				PINMOD2			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A3h.7~4 **PINMOD3**: Px.3 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A3h.3~0 **PINMOD2**: Px.2 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR A4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD54</b>	PINMOD5				PINMOD4			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A4h.7~4 **PINMOD5**: Px.5 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A4h.3~0 **PINMOD4**: Px.4 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR A5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD76</b>	PINMOD7				PINMOD6			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A5h.7~4 **PINMOD7**: Px.7 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A5h.3~0 **PINMOD6**: Px.6 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD</b>	HSNK2EN	HSNK1EN	HSNK0EN	I2CPS	-	TCOE	T2OE	T0OE
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
Reset	0	0	0	0	-	0	0	0

- A6h.7 **HSNK2EN:** Pin High-sink enable (Group 2: P06, P07, P22~P25, P30~P33)  
 0: Group 2 High-sink disable  
 1: Group 2 High-sink enable
- A6h.6 **HSNK1EN:** Pin High-sink enable (Group 1: P04, P05, P10~P17)  
 0: Group 1 High-sink disable  
 1: Group 1 High-sink enable
- A6h.5 **HSNK0EN:** Pin High-sink enable (Group 0: P00~P03, P20, P21, P34~P37)  
 0: Group 0 High-sink disable  
 1: Group 0 High-sink enable
- A6h.4 **I2CPS:** I2C Pin select  
 0: SCL/SDA = P0.0/P0.1  
 1: SCL/SDA = P3.0/P3.1
- A6h.2 **TCOE:** System clock divided by 2 and output to P1.4 enable  
 0: disable System clock divided by 2 and output to P1.4  
 1: enable System clock divided by 2 and output to P1.4
- A6h.1 **T2OE:** Timer2 overflow divided by 2(T2O) and output to P1.0 enable  
 0: disable Timer2 overflow divided by 2 and output to P1.0  
 1: enable Timer2 overflow divided by 2 and output to P1.0
- A6h.0 **T0OE:** Timer0 overflow divided by 64(T0O) and output to P3.4 enable  
 0: disable Timer0 overflow divided by 64 and output to P3.4  
 1: enable Timer0 overflow divided by 64 and output to P3.4

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSAV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

- F7h.4 **VBGOUT:** Bandgap voltage output control  
 0: Disable  
 1: Bandgap voltage output to P3.2 pin



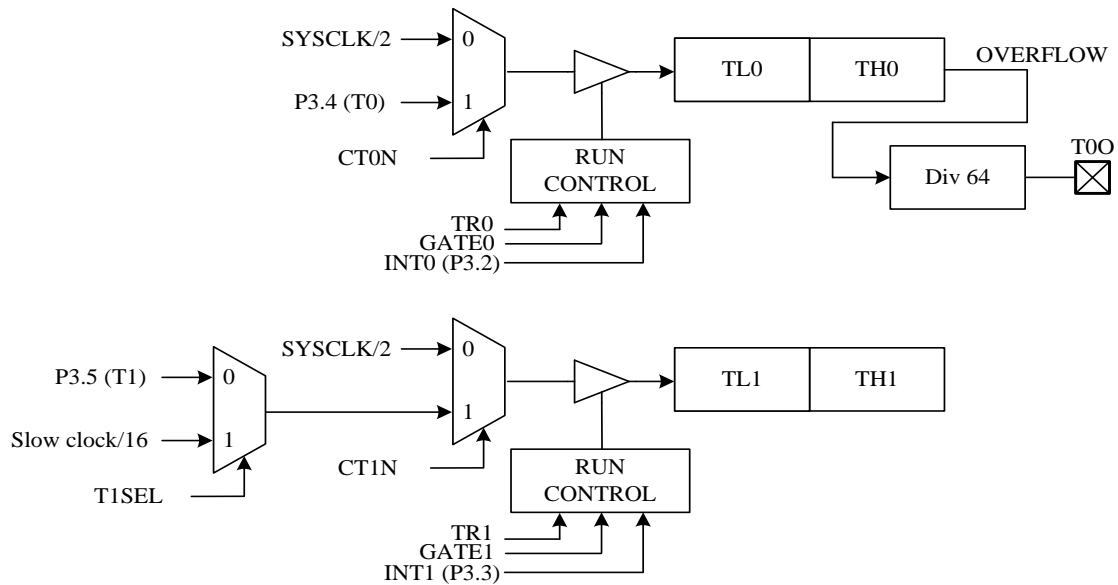
## 8. Timers

Timer0, Timer1 and Timer2 are provided as standard 8051 compatible timer/counter. Compare to the traditional 12T 8051, the Chip's Timer0/1/2 use 2 System clock cycle as the time base unit. That is, in timer mode, these timers increase at every “2 System clock” rate; in counter mode, T0/T1/T2 pin input pulse must be wider than 2 System clock to be seen by this device. In addition to the standard 8051 timers function.

The T0O pin can output the “Timer0 overflow divided by 64” signal, and the T2O pin can output the “Timer2 overflow divided by 2” signal. The CKO pin can output the system clock divided by 2 signal. By setting T0OE (A6h.0), T2OE (A6h.1) and TCOE (A6h.2) T0O, T2O and CKO can output to P3.4, P1.0 and P1.4 respectively.

### 8.1 Timer0 / Timer1

TCON and TMOD are used to set the mode of operation and to control the running and interrupt generation of the Timer0/1, with the timer/counter values stored in two pairs of 8-bit registers (TL0, TH0, and TL1, TH1).



**Timer0 and Timer1 Structure**

SFR 88h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TCON</b>	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- 88h.7 **TF1:** Timer1 overflow flag  
Set by H/W when Timer/Counter 1 overflows  
Cleared by H/W when CPU vectors into the interrupt service routine.
- 88h.6 **TR1:** Timer1 run control  
0: Timer1 stops  
1: Timer1 runs
- 88h.5 **TF0:** Timer0 overflow flag  
Set by H/W when Timer/Counter 0 overflows  
Cleared by H/W when CPU vectors into the interrupt service routine.
- 88h.4 **TR0:** Timer0 run control  
0: Timer0 stops  
1: Timer0 runs

SFR 89h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TMOD</b>	GATE1	CT1N	TMOD1		GATE0	CT0N	TMOD0	
R/W	R/W	R/W	R/W		R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	0

- 89h.7 **GATE1:** Timer1 gating control bit  
 0: Timer1 enable when TR1 bit is set  
 1: Timer1 enable only while the INT1 pin is high and TR1 bit is set
- 89h.6 **CT1N:** Timer1 Counter/Timer select bit  
 0: Timer mode, Timer1 data increases at 2 System clock cycle rate  
 1: Counter mode, Timer1 data increases at T1 pin's negative edge
- 89h.5~4 **TMOD1:** Timer1 mode select  
 00: 8-bit timer/counter (TH1) and 5-bit prescaler (TL1)  
 01: 16-bit timer/counter  
 10: 8-bit auto-reload timer/counter (TL1). Reloaded from TH1 at overflow.  
 11: Timer1 stops
- 89h.3 **GATE0:** Timer0 gating control bit  
 0: Timer0 enable when TR0 bit is set  
 1: Timer0 enable only while the INT0 pin is high and TR0 bit is set
- 89h.2 **CT0N:** Timer0 Counter/Timer select bit  
 0: Timer mode, Timer0 data increases at 2 System clock cycle rate  
 1: Counter mode, Timer0 data increases at T0 pin's negative edge
- 89h.1~0 **TMOD0:** Timer0 mode select  
 00: 8-bit timer/counter (TH0) and 5-bit prescaler (TL0)  
 01: 16-bit timer/counter  
 10: 8-bit auto-reload timer/counter (TL0). Reloaded from TH0 at overflow.  
 11: TL0 is an 8-bit timer/counter. TH0 is an 8-bit timer/counter using Timer1's TR1 and TF1 bits.

SFR 8Ah	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TL0</b>	TL0							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

8Ah.7~0 **TL0:** Timer0 data low byte

SFR 8Bh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TL1</b>	TL1							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

8Bh.7~0 **TL1:** Timer1 data low byte

SFR 8Ch	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TH0</b>	TH0							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

8Ch.7~0 **TH0:** Timer0 data high byte

SFR 8Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TH1</b>	TH1							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

8Dh.7~0 **TH1:** Timer1 data high byte

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD</b>	HSNK2EN	HSNK1EN	HSNK0EN	I2CPS	-	TCOE	T2OE	T0OE
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
Reset	0	0	0	0	-	0	0	0

A6h.2 **TCOE:** system clock divided by 2 and output to P1.4 enable  
 0: disable system clock divided by 2 and output to P1.4  
 1: enable system clock divided by 2 and output to P1.4

A6h.0 **T0OE:** Timer0 output signal(T0O) to P3.4enable  
 0: disable“Timer0 overflow flow divided by 64”output to P3.4  
 1: enable “Timer0 overflow flow divided by 64”output to P3.4

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX1</b>	CLRWDT	CLRTM3	CLRPWM0	ADSOC	CLRPWM1	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

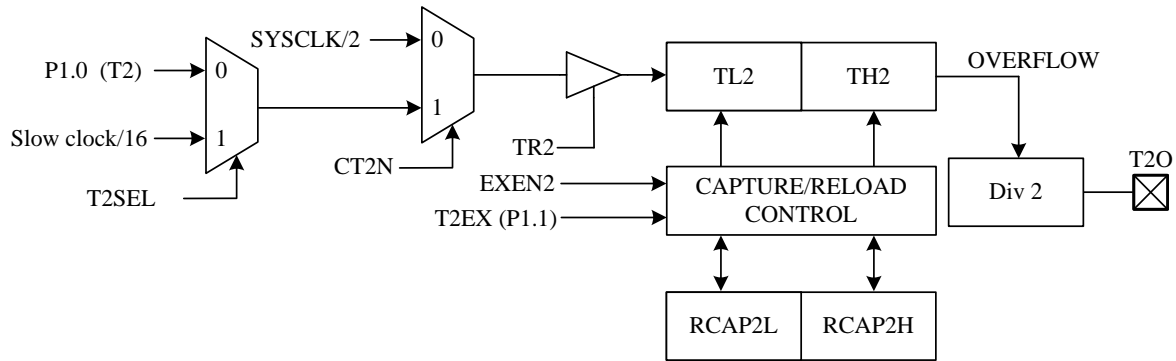
F8h.1 **T1SEL:** Timer1 counter mode (CT1N=1) input select  
 0: P3.5 (T1) pin (8051 standard)  
 1: Slow clock divide by 16 (SLOWCLK/16)

*Note:* See also Chapter 6 for more information on Timer0/1 interrupt enable and priority.

*Note:* See also Chapter 7 for details on T0O pin output settings.

## 8.2 Timer2

Timer2 is controlled through the TCON2 register with the low and high bytes of Timer/Counter2 stored in TL2 and TH2 and the low and high bytes of the Timer2 reload/capture registers stored in RCAP2L and RCAP2H.



**Timer2 Structure**

SFR C8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>T2CON</b>	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2N	CPRL2N
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- C8h.7 **TF2:** Timer2 overflow flag  
Set by H/W when Timer/Counter 2 overflows unless RCLK=1 or TCLK=1. This bit must be cleared by S/W.
- C8h.6 **EXF2:** T2EX interrupt pin falling edge flag  
Set when a capture or a reload is caused by a negative transition on T2EX pin if EXEN2=1. This bit must be cleared by S/W.
- C8h.5 **RCLK:** UART receive clock control bit  
0: Use Timer1 overflow as receive clock for serial port in mode 1 or 3  
1: Use Timer2 overflow as receive clock for serial port in mode 1 or 3
- C8h.4 **TCLK:** UART transmit clock control bit  
0: Use Timer1 overflow as transmit clock for serial port in mode 1 or 3  
1: Use Timer2 overflow as transmit clock for serial port in mode 1 or 3
- C8h.3 **EXEN2:** T2EX pin enable  
0: T2EX pin disable  
1: T2EX pin enable, it cause a capture or reload when a negative transition on T2EX pin is detected if RCLK=TCLK=0
- C8h.2 **TR2:** Timer2 run control  
0: Timer2 stops  
1: Timer2 runs
- C8h.1 **CT2N:** Timer2 Counter/Timer select bit  
0: Timer mode, Timer2 data increases at 2 System clock cycle rate  
1: Counter mode, Timer2 data increases at T2 pin's negative edge
- C8h.0 **CPRL2N:** Timer2 Capture/Reload control bit  
0: Reload mode, auto-reload on Timer2 overflows or negative transitions on T2EX pin if EXEN2=1.  
1: Capture mode, capture on negative transitions on T2EX pin if EXEN2=1.  
If RCLK=1 or TCLK=1, CPRL2N is ignored and timer is forced to auto-reload on Timer2 overflow.

SFR CAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>RCP2L</b>	RCP2L							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

CAh.7~0 **RCP2L**: Timer2 reload/capture data low byte

SFR CBh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>RCP2H</b>	RCP2H							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

CBh.7~0 **RCP2H**: Timer2 reload/capture data high byte

SFR CCh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TL2</b>	TL2							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

CCh.7~0 **TL2**: Timer2 data low byte

SFR CDh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TH2</b>	TH2							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

CDh.7~0 **TH2**: Timer2 data high byte

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD</b>	HSNK2EN	HSNK1EN	HSNK0EN	I2CPS	-	TCOE	T2OE	T0OE
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
Reset	0	0	0	0	-	0	0	0

A6h.1 **T2OE**: Timer2 output signal(T2O) to P1.0 enable  
 0: disable Timer2 overflow divided by 2 output to P1.0  
 1: enable Timer2 overflow divided by 2 output to P1.0

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX1</b>	CLRWDT	CLRTM3	CLRPWM0	ADSOC	CLRPWM1	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F8h.2 **T2SEL**: Timer2 counter mode (CT2N=1) input select  
 0: P1.0 (T2) pin (8051standard)  
 1: Slow clock divide by 16 (SLOWCLK/16)

*Note:* See also Chapter 6 for more information on Timer2 interrupt enable and priority.

*Note:* See also Chapter 7 for details on T2O pin output settings.

### 8.3 Timer3

Timer3 works as a time-base counter, which generates interrupts periodically. It generates an interrupt flag (TF3) with the clock divided by 262144, 131072, 65536, ..., 8 depending on the TM3PSC SFR. The Timer3 clock source is Slow clock SRC or FRC/512.

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>OPTION</b>	–	TM3CKS	WDTPSC		ADCKS		PWM1NMSK	PWM1PMSK
R/W	–	R/W	R/W		R/W		R/W	R/W
Reset	–	0	0	0	0	0	–	–

94h.6 **TM3CKS:** Timer3 Clock Source select

- 0: Slow clock (SRC)
- 1: FRC/512 (36KHz)

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTFLG</b>	LVDIF	EEPIF	EPPBUSY	ADIF	–	–	PCIF	TF3
R/W	R	R/W	R	R/W	–	–	R/W	R/W
Reset	0	0	0	0	–	–	0	0

95h.0 **TF3:** Timer3 Interrupt Flag

Set by H/W when Timer3 reaches TM3PSC setting cycles. Cleared automatically when the program performs the interrupt service routine. S/W can write FEh to INTFLG to clear this bit.

*Note: S/W can write 0 to clear a flag in the INTFLG, but writing 1 has no effect.*

SFR EFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX3</b>	Warmtime	TM3PSC				–	FJMPE	FJMPS
R/W	R/W	R/W				–	R/W	R/W
Reset	0	0	0	–	0	0	0	0

EFh.5~3 **TM3PSC:** Timer3 Interrupt rate

- 0000: Timer3 Interrupt rate is 262144 Timer3 clock cycle
- 0001: Timer3 Interrupt rate is 131072 Timer3 clock cycle
- 0010: Timer3 Interrupt rate is 65536 Timer3 clock cycle
- 0011: Timer3 Interrupt rate is 32768 Timer3 clock cycle
- 0100: Timer3 Interrupt rate is 16384 Timer3 clock cycle
- 0101: Timer3 Interrupt rate is 8192 Timer3 clock cycle
- 0110: Timer3 Interrupt rate is 4096 Timer3 clock cycle
- 1111: Timer3 Interrupt rate is 2048 Timer3 clock cycle
- 1000: Timer3 Interrupt rate is 1024 Timer3 clock cycle
- 1001: Timer3 Interrupt rate is 512 Timer3 clock cycle
- 1010: Timer3 Interrupt rate is 256 Timer3 clock cycle
- 1011: Timer3 Interrupt rate is 128 Timer3 clock cycle
- 1100: Timer3 Interrupt rate is 64 Timer3 clock cycle
- 1101: Timer3 Interrupt rate is 32 Timer3 clock cycle
- 1110: Timer3 Interrupt rate is 16 Timer3 clock cycle
- 1111: Timer3 Interrupt rate is 8 Timer3 clock cycle

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX1</b>	CLRWDT	CLRTM3	–	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	–	R/W	R/W	R/W	R/W	R/W
Reset	0	0	–	0	0	0	0	0

F8h.6 **CLRTM3:** Set 1 to clear Timer3, H/W auto clear it at next clock cycle.

*Note: also refer to Section 6 for more information about Timer3 Interrupt enable and priority.*

## 9. UARTs

This Chip has two UARTs, UART1 and UART2.

The **UART1** uses **SCON** and **SBUF** SFRs. **SCON** is the control register, **SBUF** is the data register. Data is written to **SBUF** for transmission and **SBUF** is read to obtain received data. The received data and transmitted data registers are completely independent.

The **UART2** uses **SCON2** and **SBUF2** SFRs. **SCON2** is the control register, **SBUF2** is the data register. Data is written to **SBUF2** for transmission and **SBUF2** is read to obtain received data. The received data and transmitted data registers are completely independent. The **UART2** supports most of the functions of **UART**, but it does not support **Mode0** and **Mode2**, it also does not support **Timer2** mode. On other hand, the option of **SMOD** is not use for **UART2**. **UART2** double baud rate is always enabled.

Both **UART1** and **UART2** provide two different **TXD** and **RXD** options. **TXD** and **RXD** can also be exchanged. In this way, there is more flexibility in application.

SFR 87h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PCON</b>	SMOD	–	–	–	GF1	GF0	PD	IDL
R/W	R/W	–	–	–	R/W	R/W	R/W	R/W
Reset	0	–	–	–	0	0	0	0

87h.7 **SMOD**: UART1 double baud rate control bit  
 0: Disable UART1 double baud rate  
 1: Enable UART1 double baud rate

SFR 98h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SCON</b>	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

98h.7~6 **SM0,SM1**: UART1 serial port mode select bit 0,1  
 00: Mode0: 8 bit shift register, Baud Rate= $F_{SYSCLK}/2$   
 01: Mode1: 8 bit UART1, Baud Rate is variable  
 10: Mode2: 9 bit UART1, Baud Rate= $F_{SYSCLK}/32$  or/64  
 11: Mode3: 9 bit UART1, Baud Rate is variable

98h.5 **SM2**: Serial port mode select bit 2  
**SM2** enables multiprocessor communication over a single serial line and modifies the above as follows. In Modes 2 & 3, if **SM2** is set then the received interrupt will not be generated if the received ninth data bit is 0. In Mode 1, the received interrupt will not be generated unless a valid stop bit is received. In Mode 0, **SM2** should be 0.

98h.4 **REN**: UART1 reception enable  
 0: Disable reception  
 1: Enable reception

98h.3 **TB8**: Transmit Bit 8, the ninth bit to be transmitted in Mode 2 and 3

98h.2 **RB8**: Receive Bit 8, contains the ninth bit that was received in Mode 2 and 3 or the stop bit is Mode 1 if **SM2**=0

98h.1 **TI**: Transmit interrupt flag  
 Set by H/W at the end of the eighth bit in Mode 0, or at the beginning of the stop bit in other modes. Must be cleared by S/W.

98h.0 **RI**: Receive interrupt flag  
 Set by H/W at the end of the eighth bit in Mode 0, or at the sampling point of the stop bit in other modes. Must be cleared by S/W.

SFR 99h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SBUF</b>	SBUF							
R/W	R/W							
Reset	-	-	-	-	-	-	-	-

99h.7~0 **SBUF**: UART1 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.

SFR 8Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SCON2</b>	SM	-	-	REN2	TB82	RB82	TI2	RI2
R/W	R/W	-	-	R/W	R/W	R/W	R/W	R/W
Reset	0	-	-	0	0	0	0	0

8Eh.7 **SM**: UART2 Serial port mode select bit  
 0: Mode1: 8 bit UART2, Baud Rate is variable  
 1: Mode3: 9 bit UART2, Baud Rate is variable  
**(UART2 does not support Mode0/Mode2)**

8Eh.4 **REN2**: UART2 reception enable  
 0: Disable reception  
 1: Enable reception

8Eh.3 **TB82**: Transmit Bit 8, the ninth bit to be transmitted in Mode 3

8Eh.2 **RB82**: Receive Bit 8, contains the ninth bit that was received in Mode3

8Eh.1 **TI2**: Transmit interrupt flag  
 Set by H/W at the beginning of the stop bit in Mode 1 & 3. Must be cleared by S/W.

8Eh.0 **RI2**: Receive interrupt flag  
 Set by H/W at the sampling point of the stop bit in Mode 1 & 3. Must be cleared by S/W.

SFR 8Fh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SBUF2</b>	SBUF2							
R/W	R/W							
Reset	-	-	-	-	-	-	-	-

8Fh.7~0 **SBUF2**: UART2 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.

SFR 9Fh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>UART2CON</b>	-	UART2BRP						
R/W	-	R/W						
Reset	-	0	0	0	0	0	0	0

9Fh.7~0 **UART2BRP**: define UART2 Baud rate.  
 $UART2 \text{ baud rate} = F_{sys}/32/UART2BRP$

SFR 93h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>UARTCON</b>	UART2PS				-	UARTPS		
R/W	R/W				-	R/W		
Reset	0	0	0	1	-	0	0	1

93h.7~4 **UART2PS**: UART2 pin select  
 0000: RXD2/TXD2 = P0.0/P0.1  
 0001: RXD2/TXD2 = P3.5/P3.6  
 0010: RXD2/TXD2 = P0.1/P0.0  
 0011: RXD2/TXD2 = P3.6/P3.5  
 0100: RXD2/TXD2 = P0.1/P0.1, 1-wire mode  
 0101: RXD2/TXD2 = P3.6/P3.6, 1-wire mode  
 0110: RXD2/TXD2 = P0.0/P0.0, 1-wire mode  
 0111: RXD2/TXD2 = P3.5/P3.5, 1-wire mode



- 1000: RXD2/TXD2 = P0.2/P0.3
- 1010: RXD2/TXD2 = P0.3/P0.2
- 1100: RXD2/TXD2 = P0.3/P0.3, 1-wire mode
- 1110: RXD2/TXD2 = P0.2/P0.2, 1-wire mode
- 93h.2~0 **UARTPS:** UART pin select
  - 0000: RXD/TXD = P3.0/P3.1
  - 0001: RXD/TXD = P3.2/P3.3
  - 0010: RXD/TXD = P3.1/P3.0
  - 0011: RXD/TXD = P3.3/P3.2
  - 0100: RXD/TXD = P3.1/P3.1, 1-wire mode
  - 0101: RXD/TXD = P3.3/P3.3, 1-wire mode
  - 0110: RXD/TXD = P3.0/P3.0, 1-wire mode
  - 0111: RXD/TXD = P3.2/P3.2, 1-wire mode

$F_{\text{SYSCLK}}$  denotes System clock frequency

**UART1 baud rate is calculated as below**

- Mode 0:  
Baud Rate= $F_{\text{SYSCLK}}/2$
- Mode 1, 3: if using Timer1 auto reload mode  
Baud Rate=  $(\text{SMOD} + 1) \times F_{\text{SYSCLK}} / (32 \times 2 \times (256 - \text{TH1}))$
- Mode 1, 3: if using Timer2  
Baud Rate=Timer2 overflow rate/16 =  $F_{\text{SYSCLK}} / (32 \times (65536 - (\text{RCP2H}, \text{RCP2L})))$
- Mode 2:  
Baud Rate=  $(\text{SMOD} + 1) \times F_{\text{SYSCLK}}/64$

**UART1 baud rate is calculated as below**

- Mode 0: Invalid
- Mode 1, 3: Baud Rate=  $F_{\text{SYSCLK}} / 32/\text{UART2BRP}$

*Note:* also refer to Section 6 for more information about UART Interrupt enable and priority.

*Note:* also refer to Section 8 for more information about how Timer2 controls UART clock.

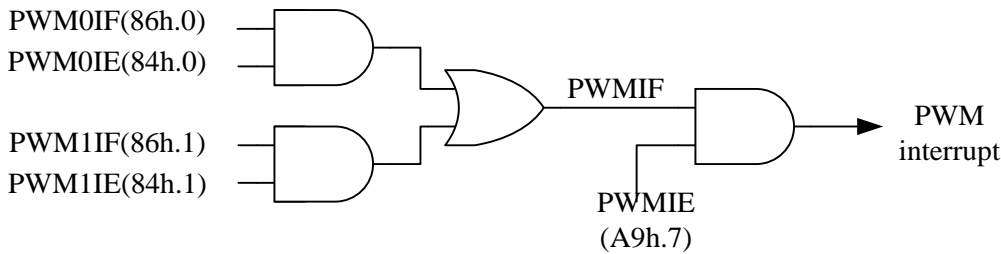
Fsys (Hz)	Expect Baud rate(bps)	UART2BRP	Generated Baud rate	Frequency offset (%)
18432000	19200	30	19200	0
18432000	28800	20	28800	0
18432000	38400	15	38400	0
18432000	57600	10	57600	0
18432000	115200	5	115200	0

## 10. PWMs

The Chip has three independent 16-bit PWM modules PWM0 and PWM1 with independent 16-bit period. The following takes PWM0 as an example for description. The PWM can generate varies frequency waveform with 65536 duty resolution on the basis of the PWM clock. The PWM clock can select FRC double frequency (FRC x 2), FRC, FRC/256 or  $F_{SYSCLK}$  as its clock source. PWM period must greater than duty.

PWM will be automatically enabled at power on. Set SFR PINMODx to control PWM output. If PINMODx is set to 1011b (relative), for example, PORTIDX = 1, PIMOD76 = BBh, then PWM0P and PWM1P will be output to P16 and P17. (see section 7)

The 16-bit period (PWM0PRD, PWM1PRD) and duty (PWM0D~PWM5D) registers all have a low byte and high byte structure. The high bytes can be directly accessed, but the low bytes can only be accessed via an internal 8-bit buffer, reading or writing to these register pairs must be carried out in a specific way. The important point to notes is that data transfer to and from the 8-bit buffer and its related low byte only takes place when write or read operation to its corresponding high bytes is executed. **Briefly speaking, write low byte first and then high byte; read high byte first and then low byte.**

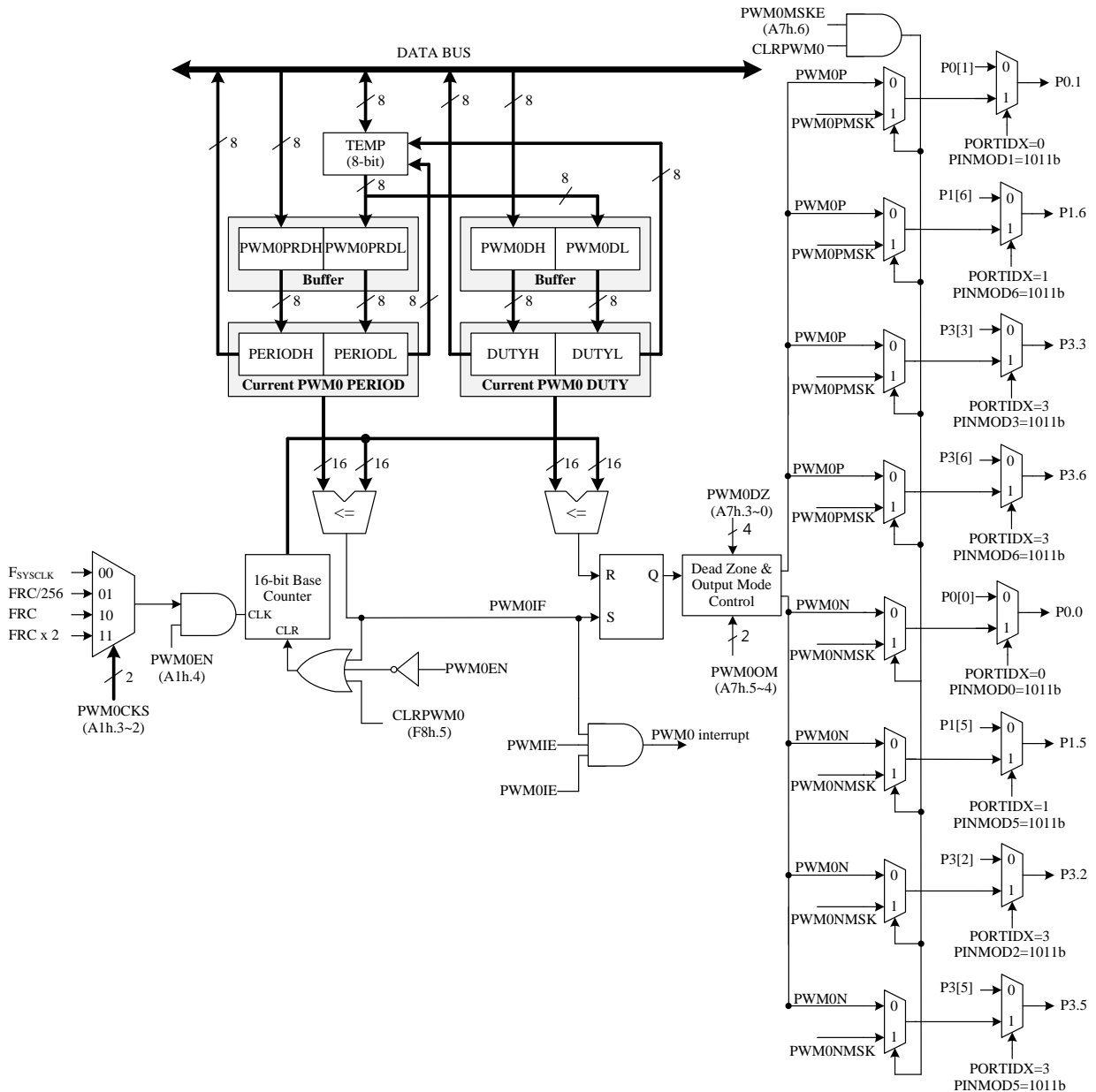


PWM Interrupt structure

### 10.1 PWM0 (PWM0P/PWM0N)

When PWM0CLR bit is set, the PWM0 will be cleared and held, otherwise the PWM0 is running. The PWM0 structure is shown as follow. The PWM0 duty cycle can be changed by writing to PWM0DH and PWM0DL. The PWM0 output signal resets to a low level whenever the 16-bit base counter matches the 16-bit PWM0 duty register {PWM0DH, PWM0DL}. The PWM0 period can be set by writing the period value to the PWM0PRDH and PWM0PRDL registers. After writing the PWM0D or PWM0PRD register, the new values will immediately save to their own buffer. H/W will update these values at the end of current period or while PWM0 is cleared. PWM0 has a corresponding interrupt flag, and an interrupt flag is generated at the end of the period.

PWMDH, PWMDL, PWM0PRDH or PWM0PRDL is a 16-bit operation, and the program should avoid interrupts when writing and reading the high byte and low byte. If you are reading and writing these 16-bit SFRs in the meantime an interrupt occurs. And these SFRs are read and written in the interrupt. It is easy to cause read and write errors. For the 16-bit PWM period and duty to read and write, it is recommended to update the data only in the main program, or update the data only in the interrupt to avoid possible errors. PWM0 structure is shown as follow.

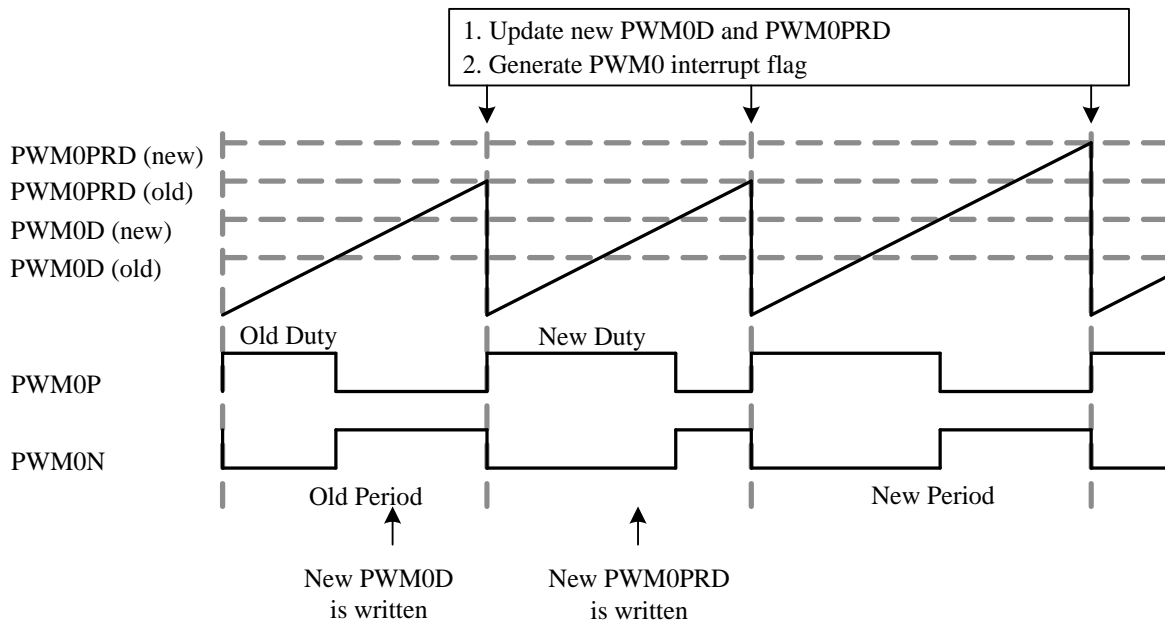


### PWM0 Structure

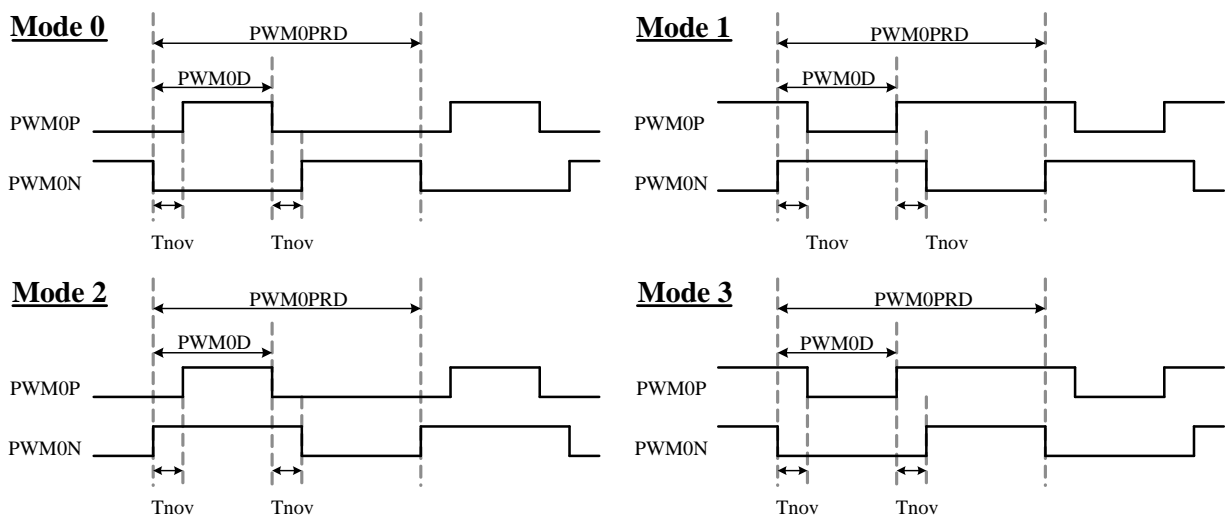
The PWM0 has two operation modes, normal mode and half-bridge mode. PWM0 output signal can be output via PWM0P and PWM0N with four different modes. These two outputs are non-overlapped with time interval  $T_{nov}$ . Non-overlapping time interval is also named as dead zone or dead band.  $T_{nov}$  is determined by setting PWM0DZ bits. The value 0~15 of PWM0DZ map onto 0~15, 16 PWM0CLK cycles respectively. If PWM0DZ=0, PWM0 outputs is directly passed to PWM0P and PWM0N so that waveforms of them have the same duty cycle. Note that, if high pulse width or low pulse width of PWM0 output is shorter than  $T_{nov}$ , the real waveforms of these two outputs will different from the expected waveforms. If the PWM0MSKE bit is set, the outputs can be masked to force output fix signal while S/W set the CLRPWM0 bit is set by H/W.

#### Normal Mode

The normal mode PWM0 is a simple structure, which switches its output high and low at uniform repeatable intervals. The PWM0D is the output duty cycle, and the output period is PWM0PRD+1. The output waveform of PWM0 is shown below.



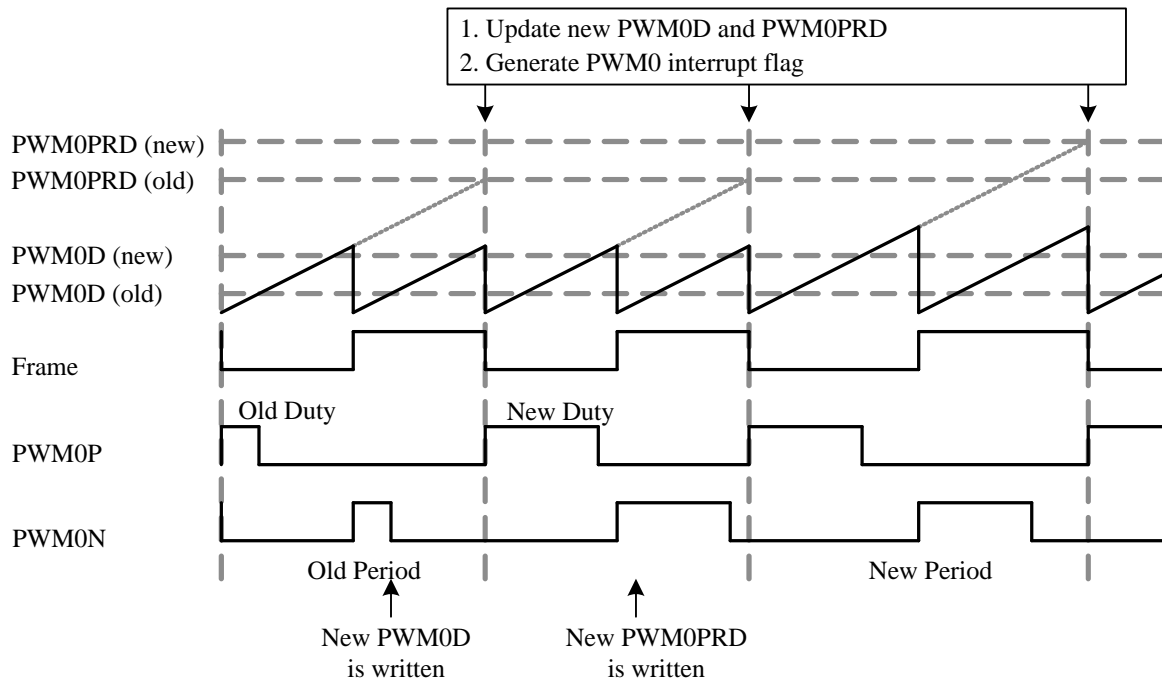
**PWM0 normal mode output waveform (PWM0OM=0, PWM0DZ=0)**



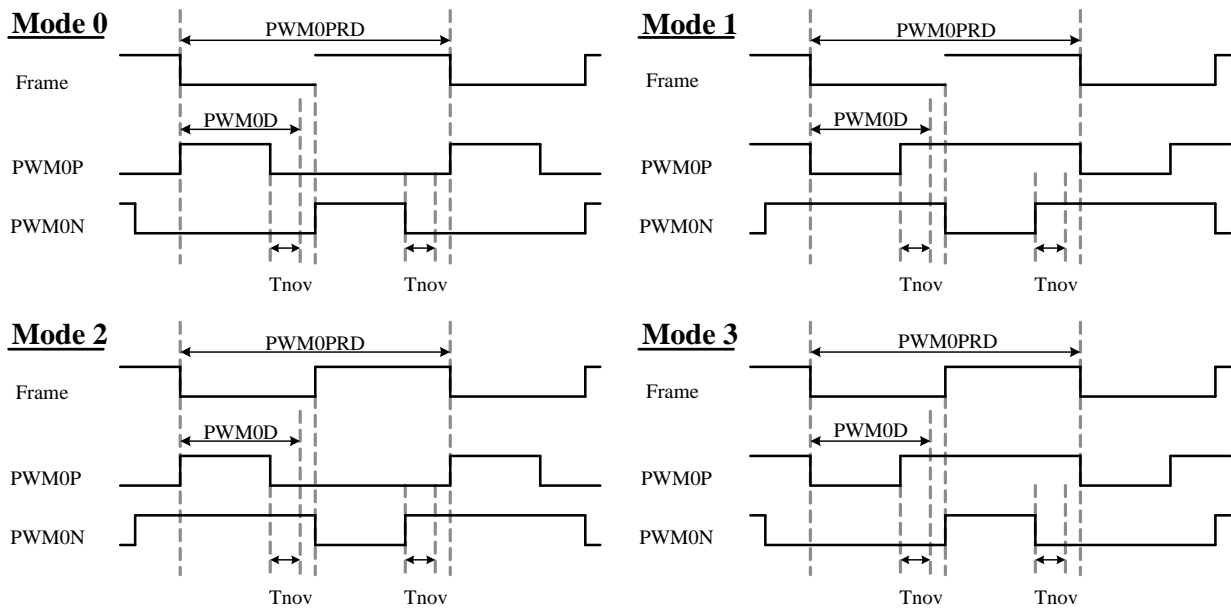
**PWM0 normal mode output modes**

### Half-Bridge Mode

The half-bridge mode PWM is similar to the normal mode but Dead zone is prohibited in half-bridge mode (SFR PWM0DZ must be 0). It has two frames in a period, PWM0P only output in the first frame, PWM0N only output in the second frame. The width of these two frames must be same, so their width is the integer part of  $PWM0PRD/2$ . Because each output channel only output in one frame, the maximum duty cycle is same as the width of a frame. If the PWM0D is larger than  $PWM0PRD/2$ , H/W will force set the duty cycle to  $PWM0PRD/2$ . Following figure shows the output waveform and the output modes.



**PWM0 half-bridge mode output waveform (PWM0OM=0, PWM0DZ=0)**



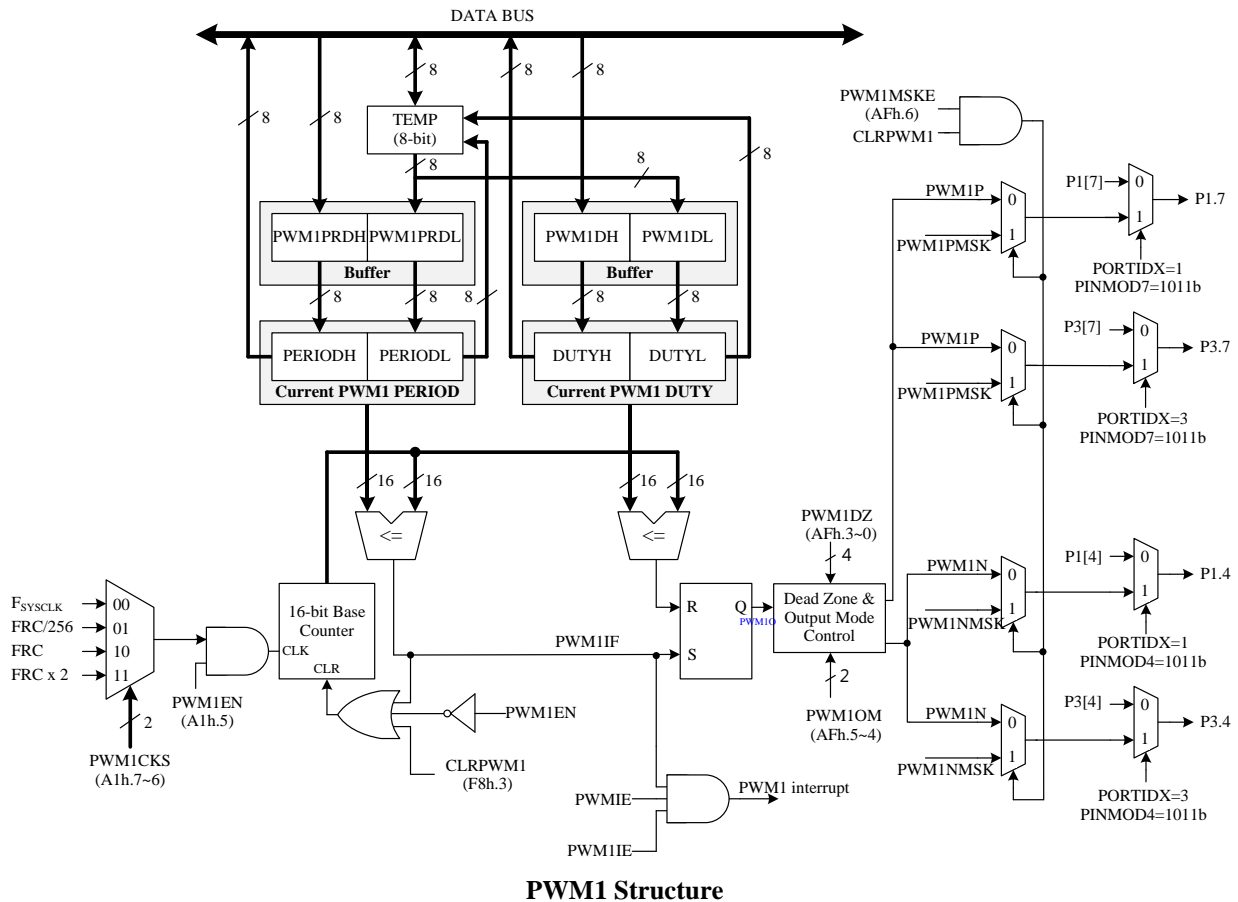
**PWM0 half-bridge mode output modes**

## 10.2 PWM1 (PWM1P/PWM1N)

PWM1 and PWM0 have similar architectures, and both have complementary PWM outputs PWM1P/PWM1N. When CLRPWM1 is set to 1, PWM1 is cleared and held, otherwise PWM1 remains running. The structure of PWM1 is shown below. The PWM1 duty cycle can be changed by writing to SFRs PWM1DH and PWM1DL. Whenever the 16-bit radix counter matches the 16-bit PWM1 duty cycle register {PWM1DH, PWM1DL}, the PWM1 output signal is reset to low level. The period of PWM1 can be set by writing to SFRs PWM1PRDH and PWM1PRDL. As soon as a PWM duty cycle or period register is written, the new value is saved to its own buffer. H/W will update these values at the

end of the current cycle or when PWM1 is cleared. PWM1 has a corresponding interrupt flag, which is generated at the end of the cycle.

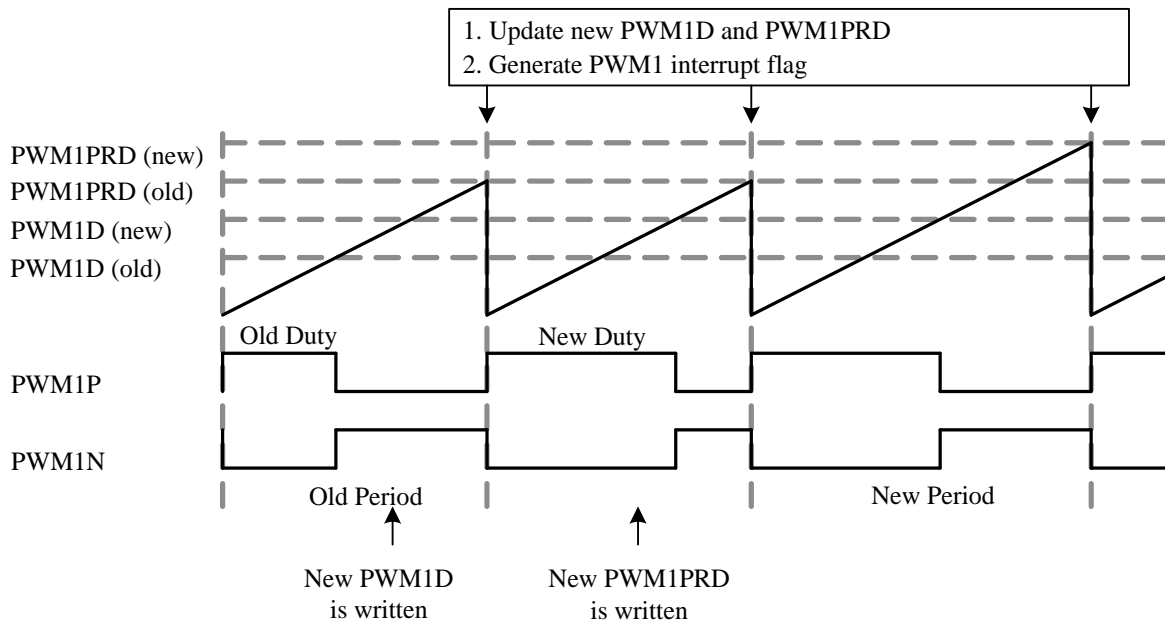
PWM1DH, PWM1DL, PWM1PRDH or PWM1PRDL are 16-bit operations, and the program should avoid interrupts when writing and reading the high byte and low byte. If an interrupt occurs during reading and writing these 16-bit registers and these registers are read and written within the interrupt, It is easy to cause reading and writing errors. For reading and writing of 16-bit PWM period and duty cycle, it is recommended to update data only in the main program or only in interrupts to avoid possible errors. The structure of PWM1 is as follows.



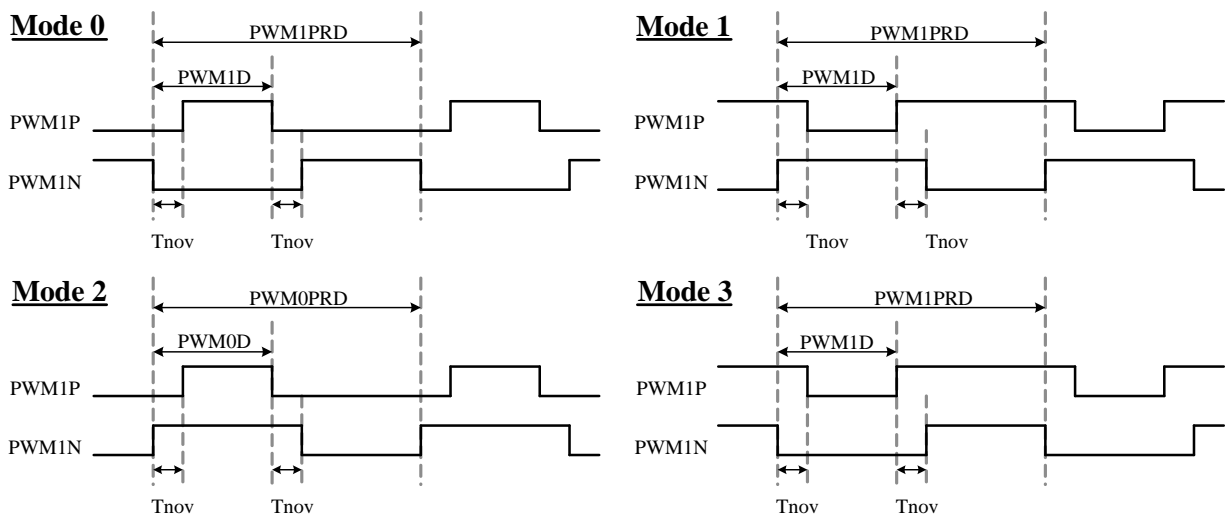
The PWM1 has two operation modes, normal mode and half-bridge mode. PWM1 output signal can be output via PWM1P and PWM1N with four different modes. These two outputs are non-overlapped with time interval  $T_{nov}$ . Non-overlapping time interval is also named as dead zone or dead band.  $T_{nov}$  is determined by setting PWM1DZ bits. The value 0~15 of PWM1DZ map onto 0~15, 16 PWM1CLK cycles respectively. If PWM1DZ=0, PWM1 outputs is directly passed to PWM1P and PWM1N so that waveforms of them have the same duty cycle. Note that, if high pulse width or low pulse width of PWM1 output is shorter than  $T_{nov}$ , the real waveforms of these two outputs will differ from the expected waveforms. If the PWM1MSKE bit is set, the outputs can be masked to force output fix signal while S/W set the CLR PWM1 bit is set by H/W.

### Normal Mode

The normal mode PWM1 is a simple structure, which switches its output high and low at uniform repeatable intervals. The PWM1D is the output duty cycle, and the output period is  $PWM1PRD+1$ . The output waveform of PWM1 is shown below.



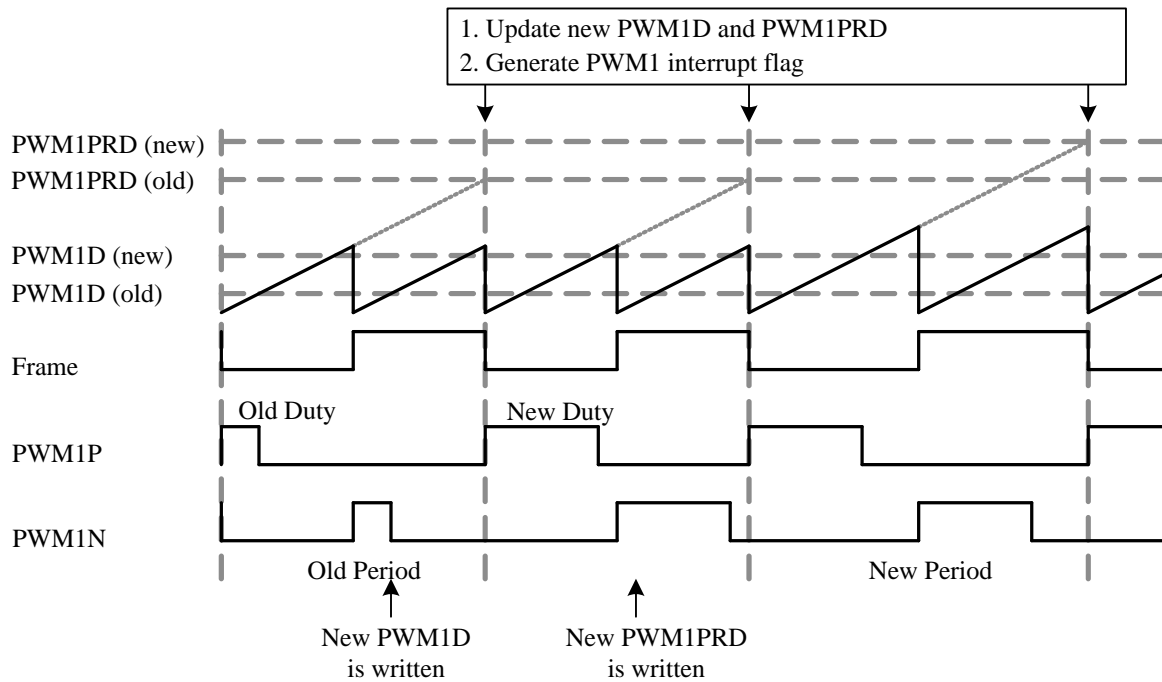
**PWM1 normal mode output waveform (PWM1OM=0, PWM1DZ=0)**



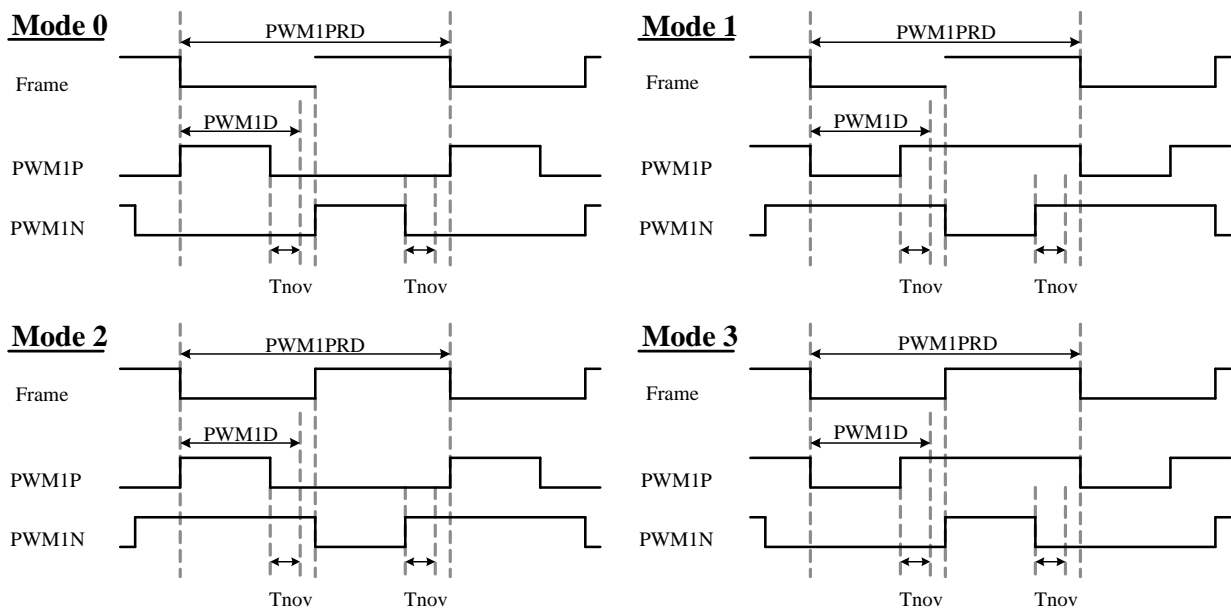
**PWM1 normal mode output modes**

**Half-Bridge Mode**

The half-bridge mode PWM1 is similar to the normal mode but Dead zone is prohibited in half-bridge mode (SFR PWM1DZ must be 0). It has two frames in a period, PWM1P only output in the first frame, PWM1N only output in the second frame. The width of these two frames must be same, so their width is the integer part of  $PWM1PRD/2$ . Because each output channel only output in one frame, the maximum duty cycle is same as the width of a frame. If the PWM1D is larger than  $PWM1PRD/2$ , H/W will force set the duty cycle to  $PWM1PRD/2$ . Following figure shows the output waveform and the output modes.



**PWM1 normal mode output waveform (PWM1OM=0, PWM1DZ=0)**

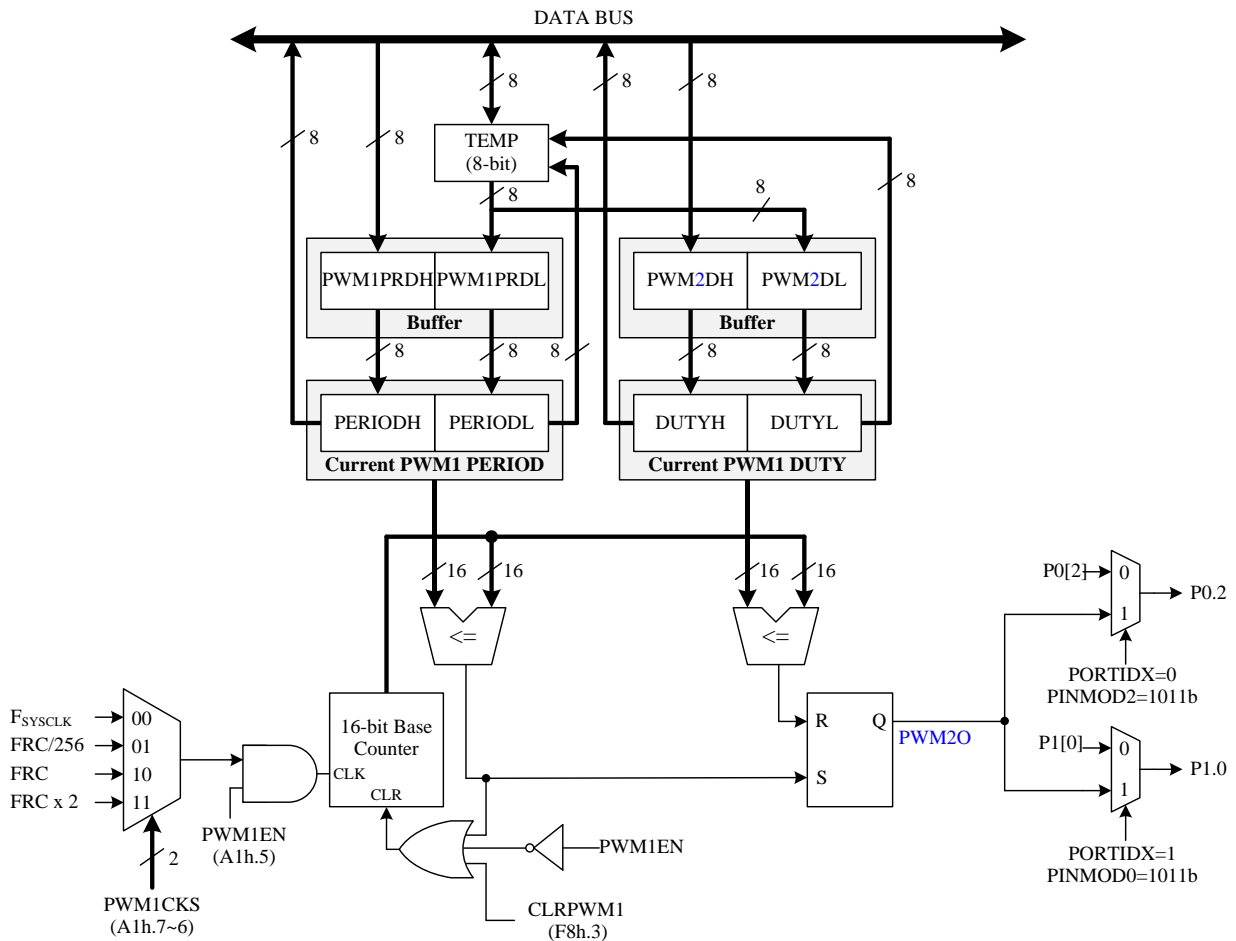


**PWM1 half-bridge mode output modes**

### 10.3 PWM2~PWM5

PWM2~5 has an architecture similar to PWM1, sharing PWM1 interrupts, clock sources and cycles, but has independently used duty cycles and no dead zone settings. The following uses PWM2 as an example for explanation. PWM2 can generate a changing frequency waveform with a duty cycle resolution of 65536 based on the PWM1 clock. The PWM1 clock can select dual frequency (FRC x 2), FRC, FRC/256 or FSYSCLK as its clock source. The PWM2 structure is as follows




**PWM2 structure**

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTE1</b>	PWMIE	I2CE	ES2	EEPIE	ADIE	LVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

A9h.7 **PWMIE:** PWM0~1 Interrupt Enable  
 0: disable PWM0~1 Interrupt  
 1: enable PWM0~1 Interrupt

SFR 84h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTE2</b>	—	—	—	—	—	—	PWM1IE	PWM0IE
R/W	—	—	—	—	—	—	R/W	R/W
Reset	—	—	—	—	—	—	0	0

84h.6 **PWM1IE:** PWM1 Interrupt Enable  
 0: disable  
 1: enable

84h.5 **PWM0IE:** PWM0 Interrupt Enable  
 0: disable  
 1: enable

SFR 86h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTPWM</b>	–	–	–	–	–	–	PWM1IF	PWM0IF
R/W	–	–	–	–	–	–	R/W	R/W
Reset	–	–	–	–	–	–	0	0

- 86h.1 **PWM1IF**: PWM1 interrupt flag  
 0: S/W write 0 to clear it  
 1: Set by H/W at the end of PWM1 period
- 86h.0 **PWM0IF**: PWM0 interrupt flag  
 0: S/W write 0 to clear it  
 1: Set by H/W at the end of PWM0 period,

SFR A1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMCON</b>	PWM1CKS		PWM1EN	PWM0EN	PWM0CKS		PWM0NMSK	PWM0PMSK
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- A1h.7~6 **PWM1CKS**: PWM1 clock source  
 00: F<sub>SYSCLK</sub>  
 01: FRC/256  
 10: FRC  
 11: FRC x 2 (V<sub>CC</sub> > 3.0V)
- A1h.5 **PWM1EN**: PWM1~5 enable  
 0: PWM1~5 disable  
 1: PWM1~5enable
- A1h.4 **PWM0EN**: PWM0 enable  
 0: PWM0 disable  
 1: PWM0 enable
- A1h.3~2 **PWM0CKS**: PWM0 clock source  
 00: F<sub>SYSCLK</sub>  
 01: FRC/256  
 10: FRC  
 11: FRC x 2 (V<sub>CC</sub> > 3.0V)
- A1h.1 **PWM0NMSK**: PWM0N mask data. If CLRPWM0=1 and PMW0MSKE=1, PWM0N will output this mask data .
- A1h.0 **PWM0PMSK**: PWM0P mask data. If CLRPWM0=1 and PMW0MSKE=1, PWM0N will output this mask data .

SFR A7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMCON2</b>	PWM0MOD	PWM0MSKE	PWM0OM		PWM0DZ			
R/W	R/W	R/W	R/W		R/W			
Reset	0	0	0	0	0	0	0	0

- A7h.7 **PWM0MOD**: PWM0 mode select  
 0: normal mode  
 1: half-bridge mode
- A7h.6 **PWM0MSKE**: mask output enable  
 0: disable  
 1: enable, PWM0P/PWM0N output data by PWM0PMSK/PWM0NMSK while CLRPWM0=1
- A7h.5~4 **PWM0OM**: PWM0 output mode select  
 00: Mode0  
 01: Mode1  
 10: Mode2  
 11: Mode3
- A7h.3~0 **PWM0DZ**: PWM0 dead zone (Dead zone is prohibited in half-bridge mode)  
 0000: 0 x T<sub>PWMCLK</sub> (disable)  
 0001: 1 x T<sub>PWMCLK</sub>  
 ...  
 1111: 15 x T<sub>PWMCLK</sub>

SFR AFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMCON3</b>	PWM1MOD	PWM1MSKE	PWM1OM		PWM1DZ			
R/W	R/W	R/W	R/W		R/W			
Reset	0	0	0	0	0	0	0	0

- AFh.7 **PWM1MOD:** PWM1mode select  
 0: normal mode  
 1: half-bridge mode
- AFh.6 **PWM1MSKE:** mask output enable  
 0: disable  
 1: enable, PWM1P/PWM1N output data by PWM1PMSK/PWM1NMSK while CLRPWM1=1
- AFh.5~4 **PWM1OM:** PWM1 output mode select  
 00: Mode0  
 01: Mode1  
 10: Mode2  
 11: Mode3
- AFh.3~0 **PWM1DZ:** PWM1 dead zone (Dead zone is prohibited in half-bridge mode)  
 0000: 0 x T<sub>PWMCLK</sub> (disable)  
 0001: 1 x T<sub>PWMCLK</sub>  
 ...  
 1111: 15 x T<sub>PWMCLK</sub>

SFR C1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM4DH</b>	PWM4DH							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C1h.7~0 **PWM4DH:** PWM4 duty high byte  
 write sequence: PW4xDL then PWM4DH  
 read sequence: PWM4DH then PWM4DL

SFR C2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM4DL</b>	PWM4DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C2h.7~0 **PWM4DL:** PWM4 duty low byte  
 write sequence: PWM4DL then PWM4DH  
 read sequence: PWM4DH then PWM4DL

SFR C3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM5DH</b>	PWM5DH							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C3h.7~0 **PWM5DH:** PWM5 duty high byte  
 write sequence: PWM5DL then PWM5DH  
 read sequence: PWM5DH then PWM5DL

SFR C4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM5DL</b>	PWM5DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C4h.7~0 **PWM5DL:** PWM5 duty low byte  
 write sequence: PWM5DL then PWM5DH  
 read sequence: PWM5DH then PWM5DL

SFR D1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM0DH</b>	PWM0DH							
R/W	R/W							
Reset	1	0	0	0	0	0	0	0

D1h.7~0 **PWM0DH**: PWM0 duty high byte  
 write sequence: PWM0DL then PWM0DH  
 read sequence: PWM0DH then PWM0DL

SFR D2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM0DL</b>	PWM0DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D2h.7~0 **PWM0DL**: PWM0 duty low byte  
 write sequence: PWM0DL then PWM0DH  
 read sequence: PWM0DH then PWM0DL

SFR D3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM1DH</b>	PWM1DH							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D3h.7~0 **PWM1DH**: PWM1 duty high byte  
 write sequence: PWM1DL then PWM1DH  
 read sequence: PWM1DH then PWM1DL

SFR D4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM1DL</b>	PWM1DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D4h.7~0 **PWM1DL**: PWM1 duty low byte  
 write sequence: PWM1DL then PWM1DH  
 read sequence: PWM1DH then PWM1DL

SFR D5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM2DH</b>	PWM2DH							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D5h.7~0 **PWM2DH**: PWM2 duty high byte  
 write sequence: PWM2DL then PWM2DH  
 read sequence: PWM2DH then PWM2DL

SFR D6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM2DL</b>	PWM2DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D6h.7~0 **PWM2DL**: PWM2 duty low byte  
 write sequence: PWM2DL then PWM2DH  
 read sequence: PWM2DH then PWM2DL

SFR D9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM0PRDH</b>	PWM0PRDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

D9h.7~0 **PWM0PRDH**: PWM0 period high byte  
 write sequence: PWM0PRDL then PWM0PRDH  
 read sequence: PWM0PRDH then PWM0PRDL

SFR DAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM0PRDL</b>	PWM0PRDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

DAh.7~0 **PWM0PRDL**: PWM0 period low byte  
 write sequence: PWM0PRDL then PWM0PRDH  
 read sequence: PWM0PRDH then PWM0PRDL

SFR DBh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM1PRDH</b>	PWM1PRDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

DBh.7~0 **PWM1PRDH**: PWM1 period high byte  
 write sequence: PWM1PRDL then PWM1PRDH  
 read sequence: PWM1PRDH then PWM1PRDL

SFR DCh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM1PRDL</b>	PWM1PRDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

DCh.7~0 **PWM1PRDL**: PWM1 period low byte  
 write sequence: PWM1PRDL then PWM1PRDH  
 read sequence: PWM1PRDH then PWM1PRDL

SFR DDh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM3DH</b>	PWM3DH							
R/W	R/W							
Reset	1	0	0	0	0	0	0	0

DDh.7~0 **PWM3DH**: PWM3 duty high byte  
 write sequence: PWM3DL then PWM3DH  
 read sequence: PWM3DH then PWM3DL

SFR DEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM3DL</b>	PWM3DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

DEh.7~0 **PWM3DL**: PWM3 duty low byte  
 write sequence: PWM3DL then PWM3DH  
 read sequence: PWM3DH then PWM3DL

SFR 91h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PORTIDX</b>	–	–	–	–	–	–	PORTIDX	
R/W	–	–	–	–	–	–	R/W	
Reset	–	–	–	–	–	–	0	0

91h.1~0 **PORTIDX**: Port index of INTPIN, PINMOD10, PINMOD32, PINMOD54, PINMOD76

SFR A2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD10</b>	PINMOD1				PINMOD0			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A2h.7~4 **PINMOD1**: Px.1 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A2h.3~0 **PINMOD0**: Px.0 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR A3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD32</b>	PINMOD3				PINMOD2			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A3h.7~4 **PINMOD3**: Px.3 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A3h.3~0 **PINMOD2**: Px.2 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR A4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD54</b>	PINMOD5				PINMOD4			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A4h.7~4 **PINMOD5**: Px.5 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A4h.3~0 **PINMOD4**: Px.4 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR A5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD76</b>	PINMOD7				PINMOD6			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A5h.7~4 **PINMOD7**: Px.7 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A5h.3~0 **PINMOD6**: Px.6 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX1</b>	CLRWDT	CLRTM3	CLRPWM0	ADSOC	CLRPWM1	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	0	1	0	0	0

F8h.5 **CLRPWM0**: PWM0 clear enable  
0: PWM0 is running  
1: PWM0 is cleared and held

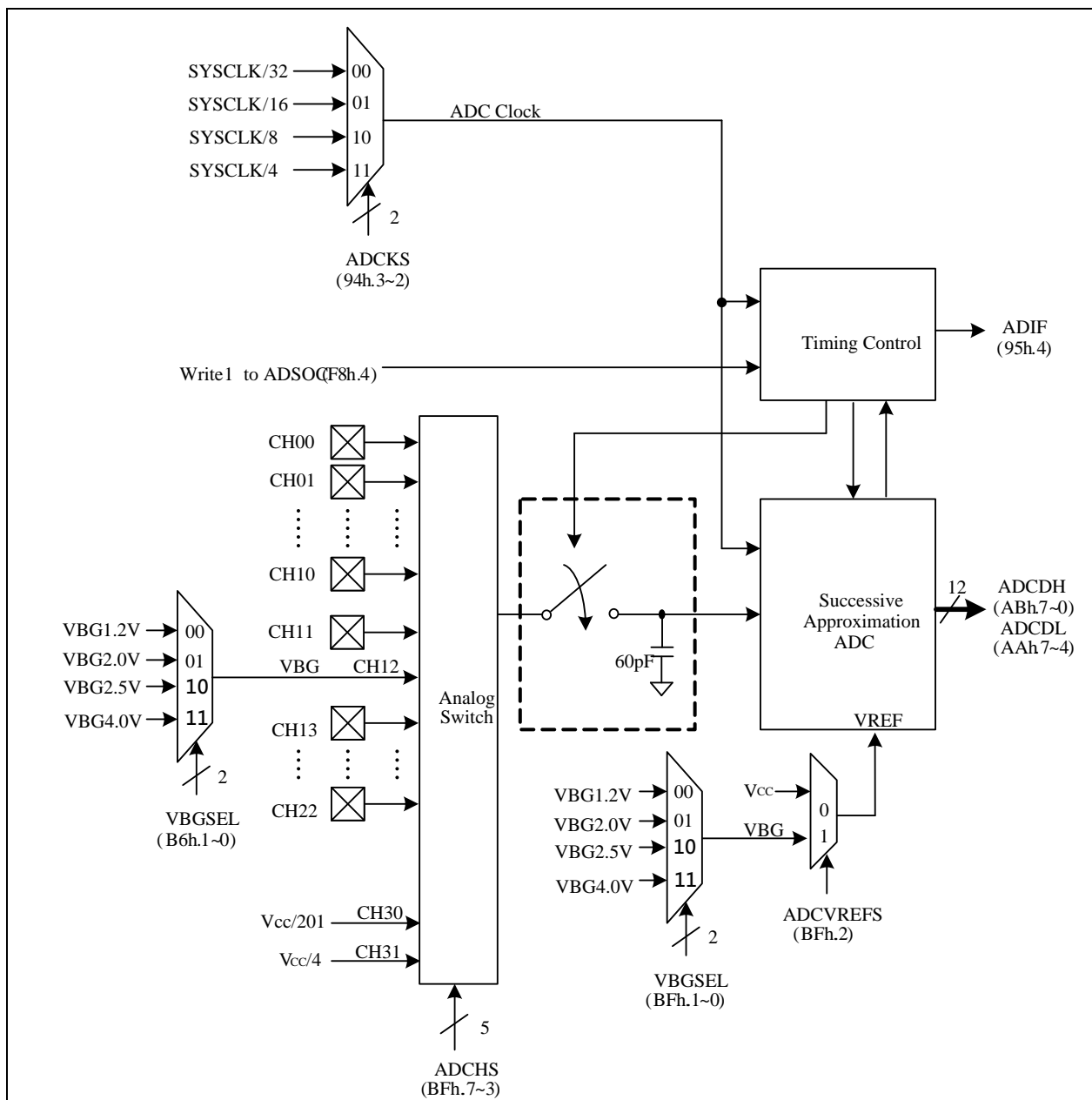
F8h.4 **CLRPWM1**: PWM1/PWM2/PWM3/PWM4/PWM5 clear enable  
0: PWM1/PWM2/PWM3/PWM4/PWM5 is running  
1: PWM1/PWM2/PWM3/PWM4/PWM5 is cleared and held

## 11. ADC

The Chip offers a 12-bit ADC consisting of a 24-channel analog input multiplexer, control register, clock generator, 12-bit successive approximation register, and output data register.

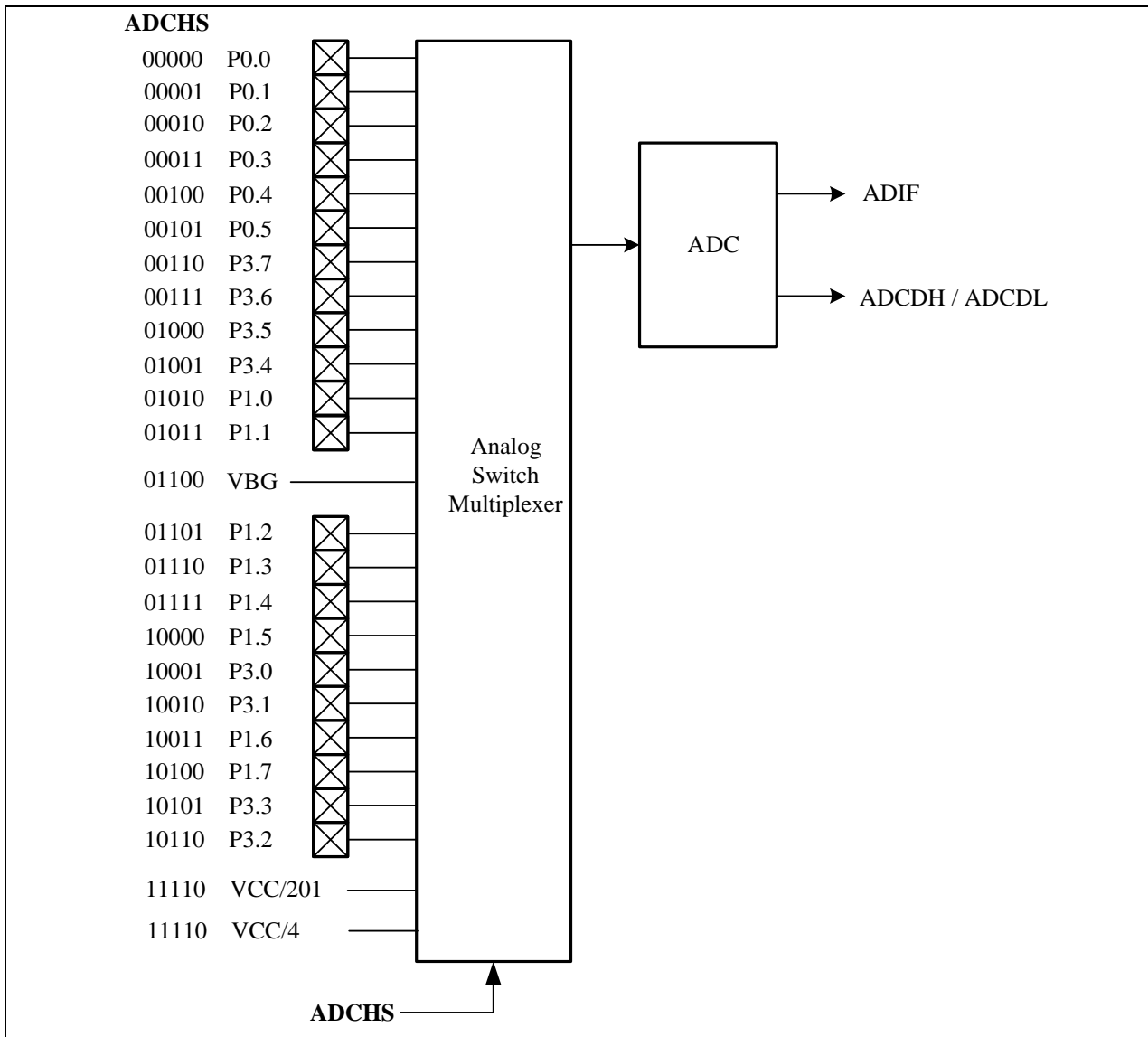
To use the ADC, set the ADCKS bit first to choose a proper ADC clock frequency, which must be less than 1 MHz. Then, launch the ADC conversion by setting the ADSOC bit, and H/W will automatic clear it at the end of the conversion. After the end of the conversion, H/W will set the ADIF bit and generate an interrupt if an ADC interrupt is enabled. The ADIF bit can be cleared by writing 0 to this bit or 1 to the ADSOC bit.

Using the ADCVREFS option, the ADC internal reference voltage source (VREF) can be selected to be VCC or VBG1.2V. When ADCVREFS=1, set VBGSEL to select VBG as 1.2V or 2.0V or 2.5V or 4.0V.



### 11.1 ADC Channels

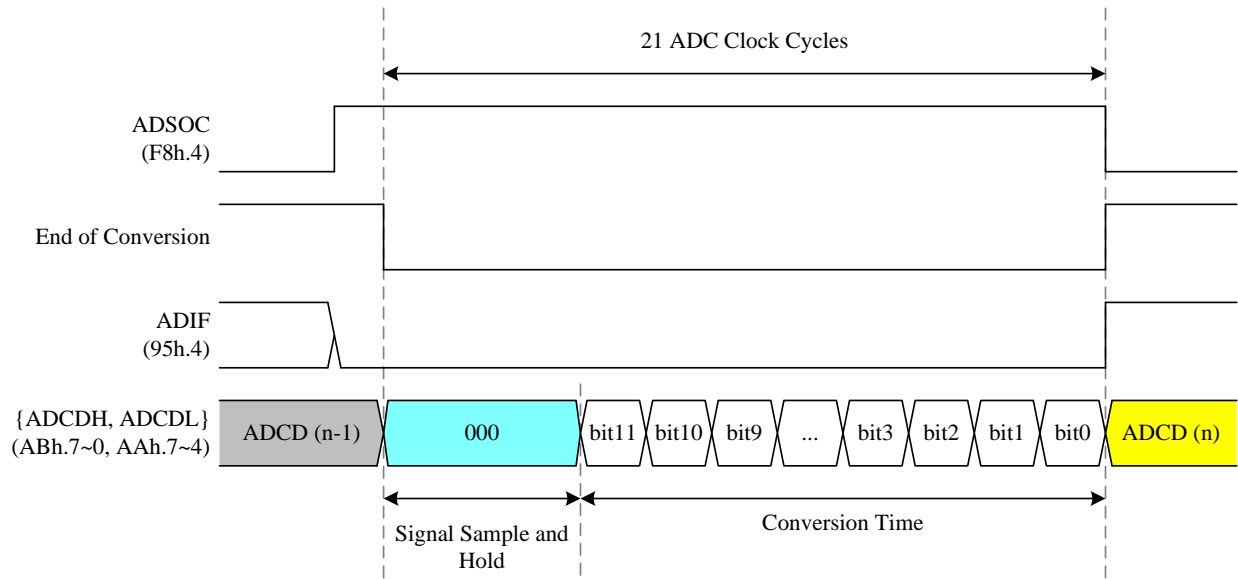
The ADC channels are connected to the analog input pins through analog switch multiplexers. The analog switch multiplexer is controlled by the ADCHS register. The chip provides up to 22 IO input pins. In addition, there are 4 internal reference voltages (VBG, VSS, VCC/4, VCC/201). When ADCHS is set to 01100b, the analog input will be connected to VBG. When ADCHS is set to 01101b, the analog input will be connected to the P1.2 input pin. At this time, P1.2 must also be set to ADC channel mode. For example, PORTIDX = 1, and the lower 4 bits of PINMOD32 are set to 0011.



### 11.2 ADC Conversion Time

The conversion time is the time required for the ADC to convert the voltage. The ADC requires two ADC clock cycles to convert each bit and several clock cycles to sample and hold the input voltage. A total of 21 ADC clock cycles are required to perform the complete conversion. When the conversion time is complete, the ADIF interrupt flag is set by H/W, and the result is loaded into the ADCDH and ADCDL registers of the 12-bit A/D result.





SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE1	PWMIE	I2CE	ES2	EEPIE	ADIE	LVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

A9h.3 **ADIE:** ADC Interrupt enable  
 0: disable ADC interrupt  
 1: enable ADC interrupt

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	–	TM3CKS	WDTPSC		ADCKS		PWM1NMSK	PWM1PMSK
R/W	–	R/W	R/W		R/W		R/W	R/W
Reset	–	0	0	0	0	0	0	0

94h.3~2 **ADCKS:** ADC clock rate select  
 00:  $F_{SYSCLK}/32$   
 01:  $F_{SYSCLK}/16$   
 10:  $F_{SYSCLK}/8$   
 11:  $F_{SYSCLK}/4$

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTFLG	LVDIF	–	–	ADIF	–	–	PCIF	TF3
R/W	R/W	–	–	R/W	–	–	R/W	R/W
Reset	0	–	–	0	–	–	0	0

95h.4 **ADIF:** ADC interrupt flag  
 Set by H/W at the end of ADC conversion. S/W writes EFh to INTFLG or sets the ADSOC bit to clear this flag.

**Note:** S/W can write 0 to clear a flag in the INTFLG, but writing 1 has no effect.

SFR AAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCDL	ADCDL				–	–	–	–
R/W	R				–	–	–	–
Reset	–	–	–	–	–	–	–	–

AAh.7~4 **ADCDL:** ADC data bit 3~0

SFR ABh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>ADCDH</b>	ADCDH							
R/W	R							
Reset	-	-	-	-	-	-	-	-

ABh.7~0 **ADCDH**: ADC data bit 11~4

SFR B6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>ADCHSEL</b>	ADCHS					ADCVERFS	VBGSEL	
R/W	R/W					R/W	R/W	
Reset	1	1	1	1	1	0	0	0

B6h.4~0 **ADCHS**: ADC channel select

00000: CH0 (P0.0)	01101: CH13 (P1.2)
00001: CH1 (P0.1)	01110: CH14 (P1.3)
00010: CH2 (P0.2)	01111: CH15 (P1.4)
00011: CH3 (P0.3)	10000: CH16 (P1.5)
00100: CH4 (P0.4)	10001: CH17 (P3.0)
00101: CH5 (P0.5)	10010: CH18 (P3.1)
00110: CH6 (P3.7)	10011: CH19 (P1.6)
00111: CH7 (P3.6)	10100: CH20 (P1.7)
01000: CH8 (P3.5)	10101: CH21 (P3.3)
01001: CH9 (P3.4)	10110: CH22 (P3.2)
01010: CH10 (P1.0)	others: reserved
01011: CH11 (P1.1)	11110: VCC/201
01100: CH12 VBG (bang gap voltage)	11111: VCC/4

B6h.2 **ADCVERFS**: ADC reference voltage select

0: VCC  
1: VBG

B6h.1 **VBGSEL**: VBG voltage select

00: 1.2V  
01: 2.0V  
10: 2.5V  
11: 4.0V

SFR 91h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PORTIDX</b>	-	-	-	-	-	-	PORTIDX	
R/W	-	-	-	-	-	-	R/W	
Reset	-	-	-	-	-	-	0	0

91h.1~0 **PORTIDX**: Port index of INTPIN, PINMOD10, PINMOD32, PINMOD54, PINMOD76

SFR A2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD10</b>	PINMOD1				PINMOD0			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A2h.7~4 **PINMOD1**: Px.1 pin control, port index (x) is defined by PORTIDX

0000~1111: see table 7.1

A2h.3~0 **PINMOD0**: Px.0 pin control, port index (x) is defined by PORTIDX

0000~1111: see table 7.1

SFR A3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD32</b>	PINMOD3				PINMOD2			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A3h.7~4 **PINMOD3**: Px.3 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A3h.3~0 **PINMOD2**: Px.2 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR A4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD54</b>	PINMOD5				PINMOD4			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A4h.7~4 **PINMOD5**: Px.5 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

A4h.3~0 **PINMOD4**: Px.4 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR A5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD76</b>	PINMOD7				PINMOD6			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A5h.7~4 **PINMOD7**: Px.7 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

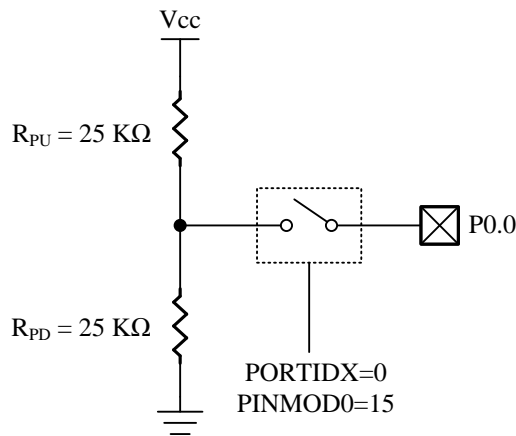
A5h.3~0 **PINMOD6**: Px.6 pin control, port index (x) is defined by PORTIDX  
0000~1111: see table 7.1

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX1</b>	CLRWDT	CLRTM3	CLRPWM0	ADSOC	CLRPWM1	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	0	1	0	0	0

F8h.4 **ADSOC**: Start ADC conversion  
Set the ADSOC bit to start ADC conversion, and the ADSOC bit will be cleared by H/W at the end of conversion. S/W can also write 0 to clear this flag. ◦

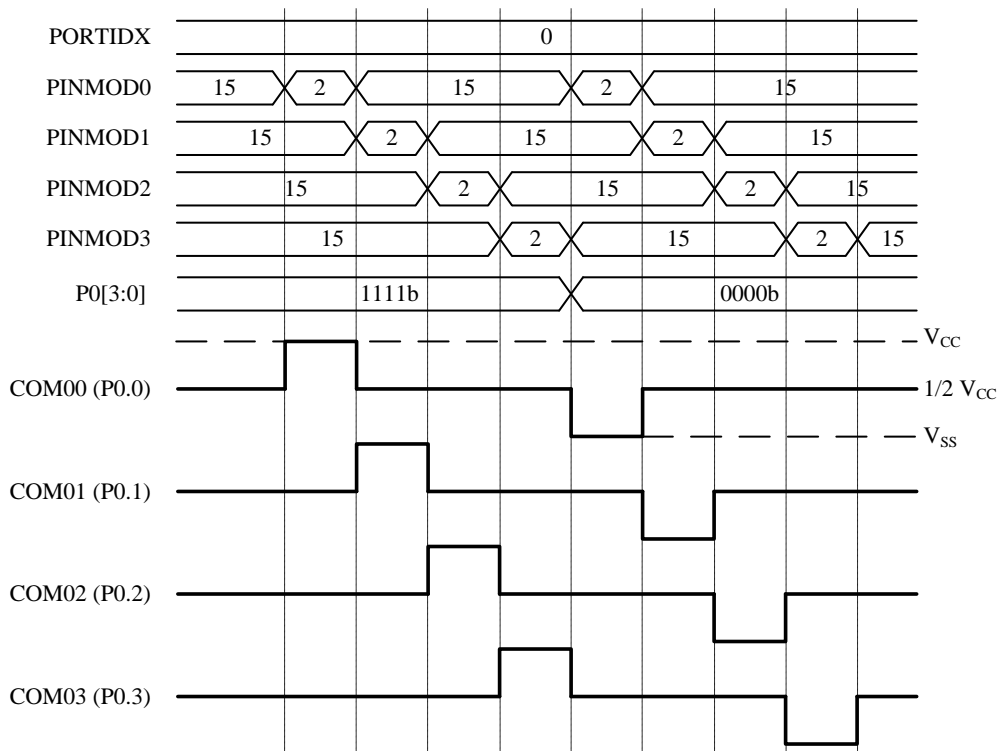
## 12. S/W Controller LCD Driver

The chip supports an S/W controlled method to driving LCD. All of the IO pins can be the Common pins. User can flexibly adjust the Common pins and Segment pins. It is capable of driving the LCD panel with 121 dots (Max.) by 11 Commons (COM) and 11 Segments (SEG). The P0.0~P0.5 are used for Common pins COM00~COM07. The P1.0~P1.7 are used for Common pins COM10~COM17. The P3.0~P3.7 are used for Common pins COM30~COM37. Common pins are capable of driving 1/2 bias by setting the corresponding PINMODE=15 (*see section 7*). Refer to the following figures.



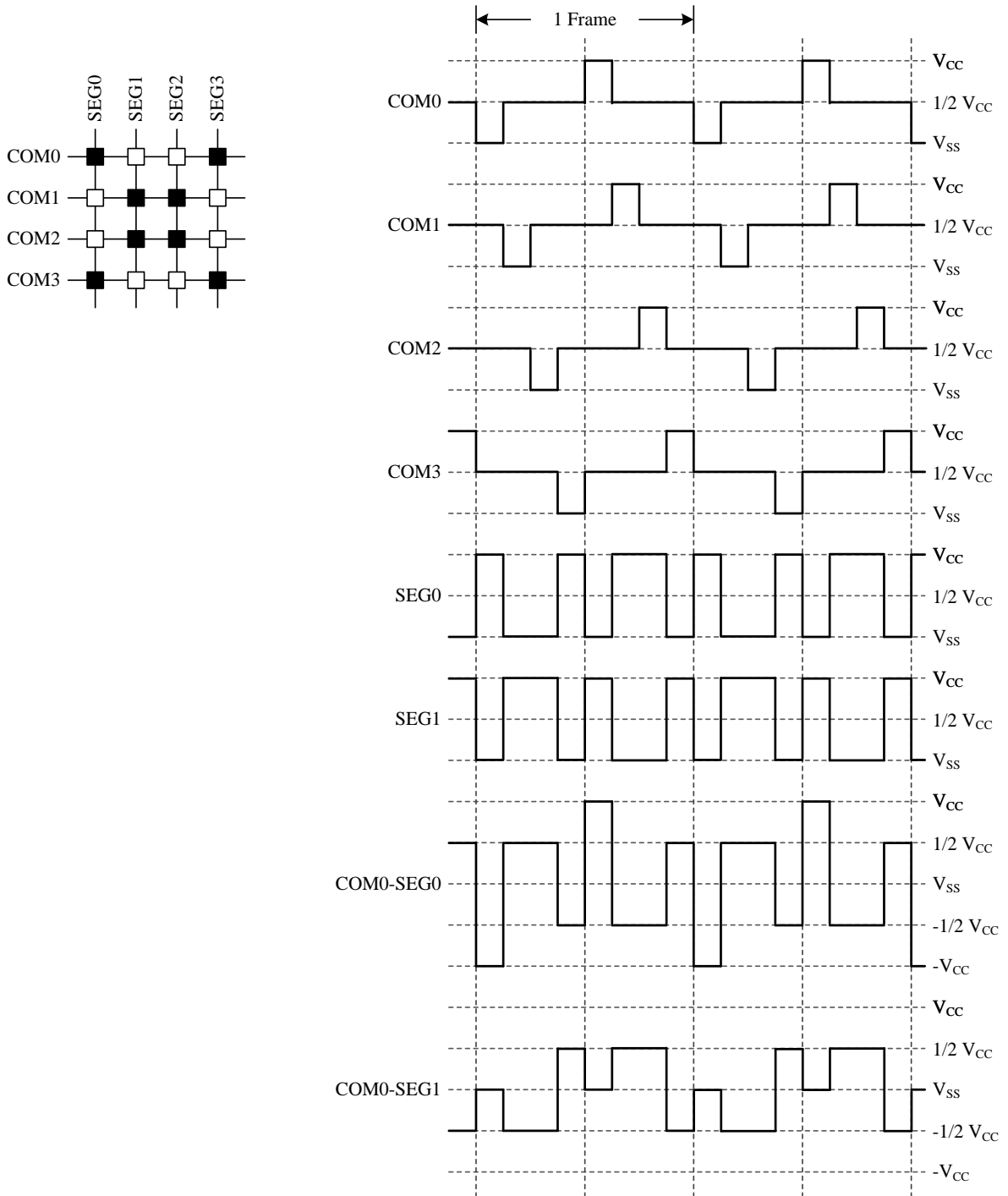
**LCD COM00 Circuit**

The frequency of any repeating waveform output on the COM pin can be used to represent the LCD frame rate. The figure below shows an LCD frame.



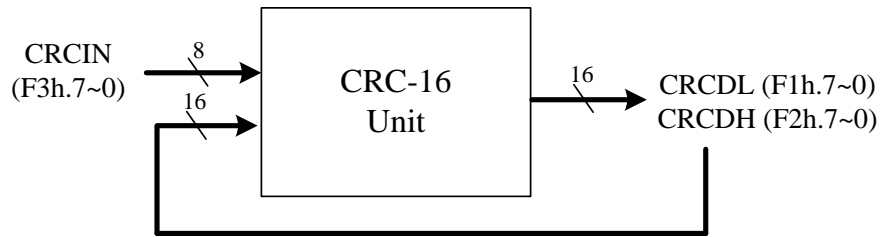
**S/W Controlled LCD COM00~03 Scanning**

1/4 Duty, 1/2 Bias Output Waveform



### 13. Cyclic Redundancy Check (CRC)

The chip supports an integrated 16-bit Cyclic Redundancy Check function. The Cyclic Redundancy Check (CRC) calculation unit is an error detection technique test algorithm and uses to verify data transmission or storage data correctness. The CRC calculation takes an 8-bit data stream or a block of data as input and generates a 16-bit output remainder. The data stream is calculated by the same generator polynomial.



**CRC Block Diagram**

The CRC generator provides the 16-bit CRC result calculation based on the CRC-16-IBM polynomial. In this CRC generator, there are only one polynomial available for the numeric values calculation. It can't support the 16-bit CRC calculations based on any other polynomials. Each write operation to the CRCIN register creates a combination of the previous CRC value stored in the CRCDH and CRCDL registers. It will take one MCU instruction cycle to calculate.

#### CRC-16-IBM (Modbus) Polynomial representation: $X^{16} + X^{15} + X^2 + 1$

SFR F1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>CRCDL</b>	CRCDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

F1h.7~0 **CRCDL**: 16-bit CRC checksum data bit 7~0

SFR F2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>CRCDH</b>	CRCDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

F2h.7~0 **CRCDL**: 16-bit CRC checksum data bit 15~8

SFR F3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>CRCIN</b>	CRCIN							
W	W							
Reset	-	-	-	-	-	-	-	-

F3h.7~0 **CRCIN**: CRC input data register

## 14. Multiplier and divider

The chip provide multiplier and divider have the following functions. The 8 bit operation is fully compatible with industry standard 8051.

- 8 bits × 8 bits = 16 bit (standard 8051)
- 8 bits ÷ 8 bits = 8 bits, 8 bits remainder (standard 8051)
- 16 bits × 16 bits = 32 bit
- 16 bits ÷ 16 bits = 16 bits, 16 bits remainder
- 32 bits ÷ 16 bits = 32 bits, 16 bits remainder

No matter 8bit / 16bit / 32bit operation, it's easy to execute by MUL AB and DIV AB instruction. There is extra SFR EXA/EXA2/EXA3/EXB for 16bit / 32bit multiply and divide operation.

For 8 bit multiplier/divider operation, be sure SFR bit muldiv16=0 and div32=0.

For 16 bit multiplier operation, multiplicand, multiplier and product as follows. 16 bit multiplier takes 16 System clock cycles to execute.

Condition	SFR bit muldiv16=1 and div32=0			
Multiplication	Byte3	Byte2	Byte1	Byte0
Multiplicand	-	-	EXA	A
Multiplier	-	-	EXB	B
Product	EXB	B	A	EXA
OV	Product (EXB or B) !=0			-

For 16 bit divider operation, dividend, divisor, quotient, remainder read as follows. 16 bit divider takes 16 System clock cycles to execute.

Condition	SFR bit muldiv16=1 and div32=0			
Division	Byte3	Byte2	Byte1	Byte0
Dividend	-	-	EXA	A
Divisor	-	-	EXB	B
Quotient	-	-	A	EXA
Remainder	-	-	B	EXB
OV	Divisor EXB = B =0			

For 32 bits ÷ 16 bits operation, dividend, divisor, quotient, remainder read as follows. 32 bit divider takes 32 System clock cycles to execute.

Condition	SFR bit muldiv16=1 and div32=1			
Division	Byte3	Byte2	Byte1	Byte0
Dividend	EXA3	EXA2	EXA	A
Divisor	-	-	EXB	B
Quotient	A	EXA	EXA2	EXA3
Remainder	-	-	B	EXB
OV	Divisor EXB=B =0			

SFR CEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EXA2</b>	EXA2							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

CEh.7~0 **EXA2**: Expansion accumulator 2

SFR CFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EXA3</b>	EXA3							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

CFh.7~0 **EXA3**: Expansion accumulator 3

SFR E6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EXA</b>	EXA							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

E6h.7~0 **EXA**: Expansion accumulator

SFR E7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EXB</b>	EXB							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

E7h.7~0 **EXB**: Expansion B register

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSVAV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	1	1	0

F7h.3 **DIV32**: (only active when MULDVI16=1)

0: instruction DIV as 16/16 bit division operation

1: instruction DIV as 32/16 bit division operation

F7h.0 **MULDIV16**:

0: instruction MUL/DIV as 8\*8, 8/8 operation

1: instruction MUL/DIV as 16\*16, 16/16 or 32/16 operation

ARITHMETIC				
Mnemonic	Description	byte	cycle	opcode
MUL AB	Multiply A by B	1	8/16	A4
DIV AB	Divide A by B	1	8/16/32	84

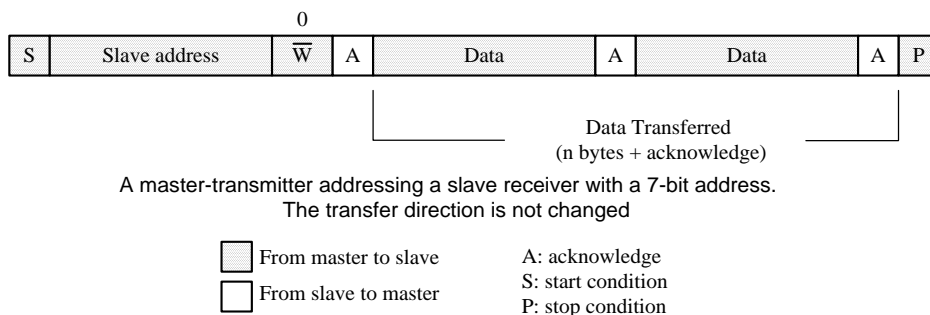


### 15. Master I<sup>2</sup>C Interface

#### Master I<sup>2</sup>C interface transmit mode:

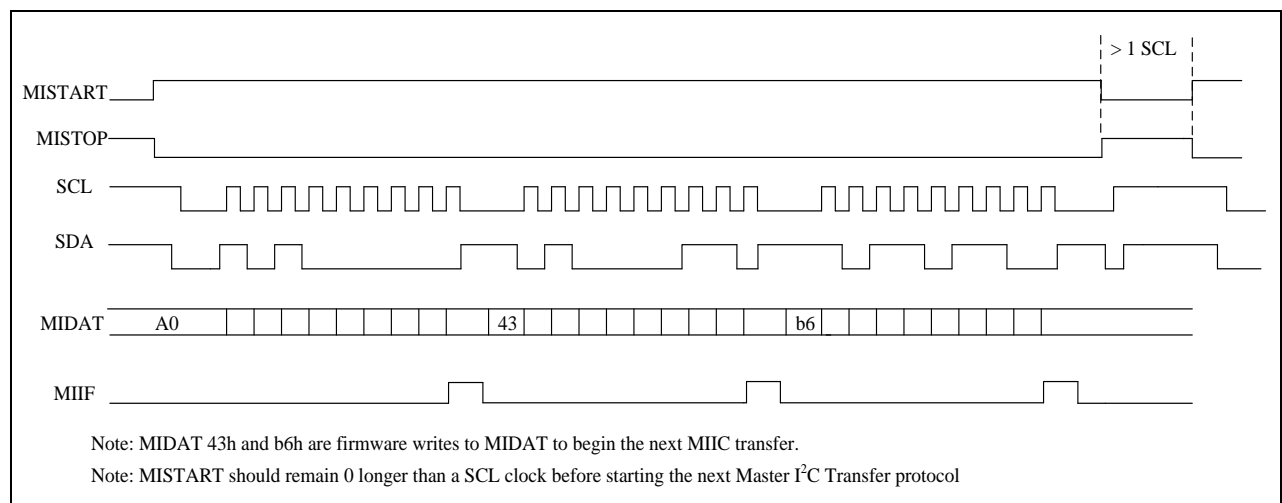
At the beginning write slave address and direction bit to MIDAT and set MISTART. After the START condition (MISTART), the 7 bits slave address and one bit direction bit are sent. When MIIF convert to 1, address and direction bit transmission was complete. After sending the address and direction bit, user should clear MIIF and write MIDAT to start first data transmission. When MIIF convert to 1, data transfer to slave was complete. User can write MIDAT again to transfer next data to slave. Set MISTOP to finish transmit mode. The main I<sup>2</sup>C clock frequency selection needs to be limited to below 260KHz

MISTART must remain at 1 for the next transfer. After the final data transmit/receive, set MISTOP to finish transmit/receive protocol. MISTART should remain 0 longer than a SCL clock before starting the next Master I<sup>2</sup>C protocol. SCL clock can be adjusted via MICR.



#### Master I<sup>2</sup>C Transmit flow:

- (1) Write slave address and direction bit to MIDAT
- (2) Clear MISTOP and set MISTART to start I<sup>2</sup>C transmission
- (3) Wait until MIIF convert to 1 (interrupt will be issued according to the user's request) and Clear MIIF
- (4) Write data to MIDAT to start next transfer (MISTART must remain at 1)
- (5) Wait until MIIF convert to 1 (interrupt will be issued according to the user's request) and Clear MIIF,  
Loop (4) ~ (5) for next transfer.
- (6) Clear MISTART and set MISTOP to stop the I<sup>2</sup>C transfer



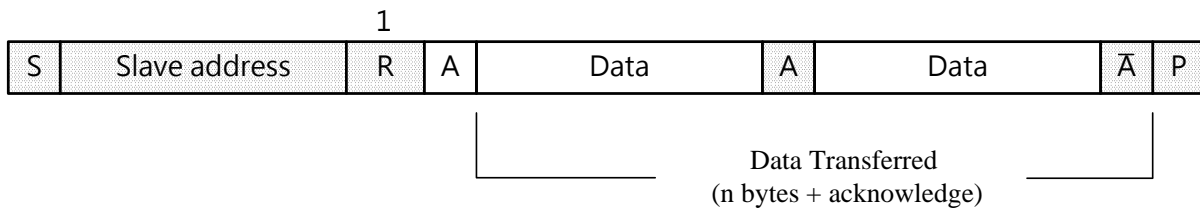
Master Transmit Timing

Note: MISTART should remain 0 longer than a SCL period before starting the next Master I<sup>2</sup>C protocol.

**Master I<sup>2</sup>C interface Receive mode:**

At the beginning write slave address and direction bit to MIDAT and set MISTART. After the START condition (MISTART), the 7 bits slave address and one bit direction bit are sent. When MIIF convert to 1, address and direction bit transmission was complete. After sending the address and direction bit, user should clear MIIF and read MIDAT to start first receive data (The first reading of MIDAT does not represent the data returned by the slave). When MIIF convert to 1, data receive from slave was complete. User can read MIDAT to get data from slave, and start next receive. Set MISTOP to finish receive mode.

MISTART must remain at 1 for the next transfer. After final data transmit/receive, set MISTOP to finish transmit/receive protocol. MISTART should remain 0 longer than a SCL clock before starting the next Master I<sup>2</sup>C protocol. SCL clock can be adjusted via MICR.

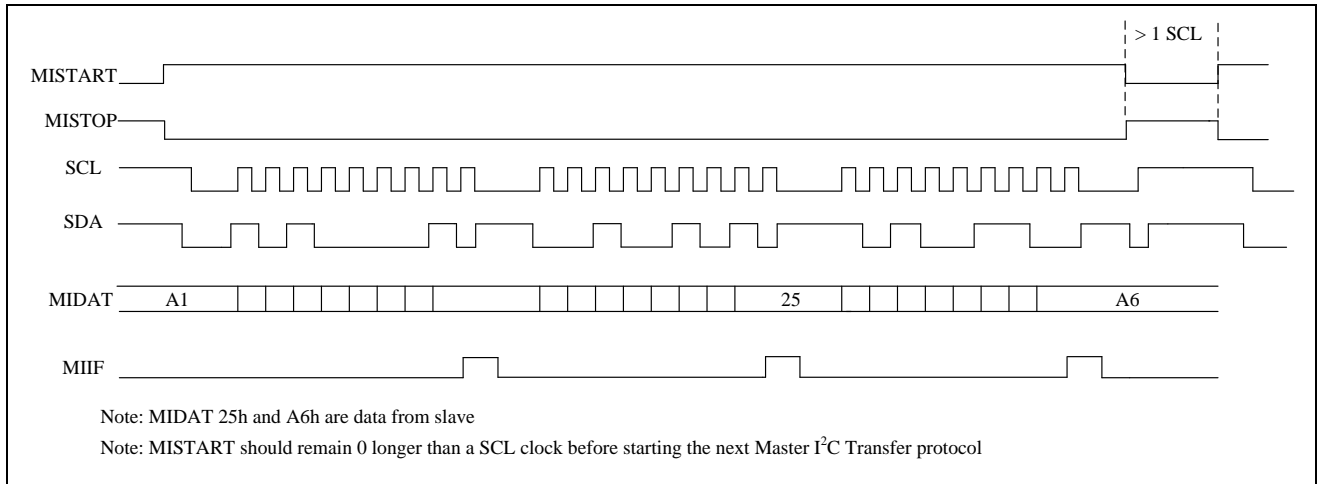


A master reads a slave immediately after the first byte

- From master to slave
- From slave to master
- A: acknowledge
- S: start condition
- P: stop condition
- $\bar{A}$ : Non acknowledge

**Master I<sup>2</sup>C Receive flow:**

- (1) Write the slave address and direction bit to MIDAT
  - (2) Clear MISTOP and set MISTART to start I<sup>2</sup>C transfer
  - (3) Wait until MIIF converts to 1 (issue an interrupt according to user requirements), clear MIIF
  - (4) Read MIDAT to start receiving data for the first time (receiving data has not been completed at this time, and the read MIDAT should be discarded)
  - (5) Wait until MIIF converts to 1 (issue an interrupt according to user requirements), clear MIIF
  - (6) Read MIDAT to get the received data, and loop (5)~(6) for the next reception
- If it is the last transaction, you need to set MIACKO and MISTOP first, and then read MIDAT.
- (7) Set MISTOP to stop I<sup>2</sup>C transmission


**Master Receive Timing**

I <sup>2</sup> C Function Pin	PINMOD <sub>xx</sub>	Px.n SFR data	Pin State
I <sup>2</sup> C Master SCL	<b>0000</b>	X	I <sup>2</sup> C Clock Output (Open Drain Output, Pull-up)
	<b>xx10</b>	X	I <sup>2</sup> C Clock Output (CMOS Push-Pull)
I <sup>2</sup> C Master SDA	<b>0000</b>	1	I <sup>2</sup> C DATA (Pull-up)

**Pin Mode Setting for Master I<sup>2</sup>C**

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD</b>	HSNK2EN	HSNK1EN	HSNK0EN	I2CPS	UART2PS		UART1PS	
R/W	R/W	R/W	R/W	R/W	R/W		R/W	
Reset	0	0	0	0	0	0	0	0

A6h.4 **I2CPS: I<sup>2</sup>C pin select**  
 0: SCL/SDA = P0.0/P0.1  
 1: SCL/SDA = P3.0/P3.1

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTE1</b>	PWMIE	I2CE	ES2	–	ADIE	LVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	–	R/W	R/W	R/W	R/W
Reset	0	0	0	–	0	0	0	0

A9h.6 **I2CE: I<sup>2</sup>C interrupt enable**  
 0: Disable I<sup>2</sup>C interrupt  
 1: Enable I<sup>2</sup>C interrupt

SFR E1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>MICON</b>	MIEN	MIACKO	MIIF	MIACKI	MISTART	MISTOP	MICR	
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	1	0	0

- E1h.7 **MIEN**: Master I<sup>2</sup>C enable  
 0: disable  
 1: enable
- E1h.6 **MIACKO**: When Master I<sup>2</sup>C receive data, send acknowledge to I<sup>2</sup>C Bus  
 0: ACK to slave device  
 1: NACK to slave device
- E1h.5 **MIIF**: Master I<sup>2</sup>C Interrupt flag  
 0: write 0 to clear it  
 1: Master I<sup>2</sup>C transfer one byte complete
- E1h.4 **MIACKI**: When Master I<sup>2</sup>C transfer, acknowledgement form I<sup>2</sup>C bus (read only)  
 0: ACK received  
 1: NACK received
- E1h.3 **MISTART**: Master I<sup>2</sup>C Start bit  
 1: start I<sup>2</sup>C bus transfer
- E1h.2 **MISTOP**: Master I<sup>2</sup>C Stop bit  
 1: send STOP signal to stop I<sup>2</sup>C bus
- E1h.1~0 **MICR**: Master I<sup>2</sup>C (SCL) clock frequency selection  
 00: F<sub>SYSCLK</sub>/4 (ex. If F<sub>SYSCLK</sub>=16MHz, I<sup>2</sup>C clock is 4 MHz)  
 01: F<sub>SYSCLK</sub>/16 (ex. If F<sub>SYSCLK</sub>=16MHz, I<sup>2</sup>C clock is 1 MHz)  
 10: F<sub>SYSCLK</sub>/64 (ex. If F<sub>SYSCLK</sub>=16MHz, I<sup>2</sup>C clock is 250 KHz)  
 11: F<sub>SYSCLK</sub>/256 (ex. If F<sub>SYSCLK</sub>=16MHz, I<sup>2</sup>C clock is 62.5 KHz)  
 Note: The main I2C clock frequency selection needs to be limited to below 260KHz.  
 That is, when F<sub>SYSCLK</sub>=16MHz, the options 00/01 (division by 4 / 16) cannot be selected.

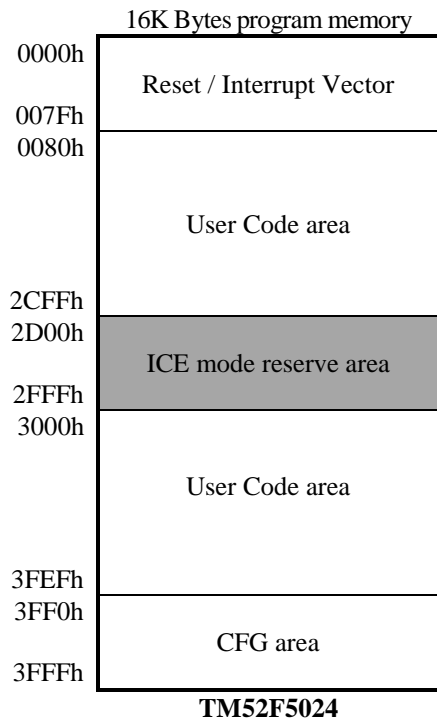
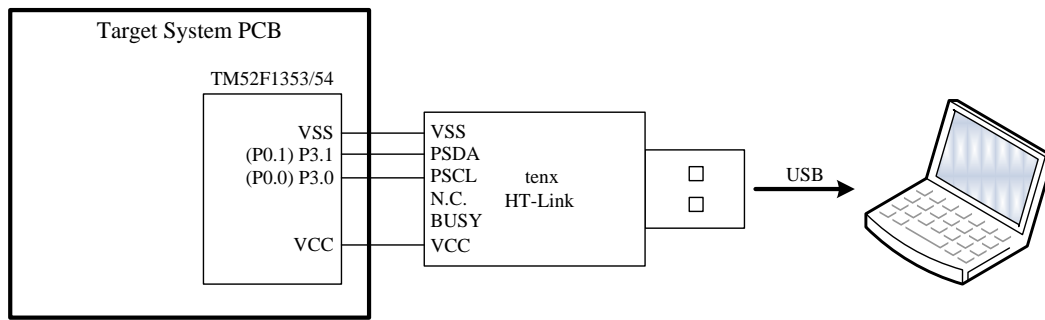
SFR E2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>MIDAT</b>	MIDAT							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- E2h.7~0 **MIDAT**: Master I<sup>2</sup>C data shift register  
 (W):After Start and before Stop condition, write this register will resume transmission to I<sup>2</sup>C bus  
 (R): After Start and before Stop condition, read this register will resume receiving from I<sup>2</sup>C bus

### 16. In Circuit Emulation (ICE) Mode

This device can support the In Circuit Emulation Mode. To use the ICE Mode, user just needs to connect P3.0 and P3.1 pin to the tenx proprietary EV Module. The benefit is that user can emulate the whole system without changing the on board target device. But there are some limits for the ICE mode as below.

1. The device must be un-protect.
2. The device's P3.0 and P3.1 pins must work in input Mode.
3. The Program Memory's addressing space 2D00h~2FFFh and 0033h~003Ah are occupied by tenx EV module. So user Program cannot access these spaces.
4. The HT-Link communication pin's function cannot be emulated.
5. The P3.0 and P3.1 pin's can be replaced by P0.0 and P0.1 (only in ICE Mode).
6. The V<sub>DD</sub> level is controlled by HT-Link module.



**SFR & CFGW MAP**

Adr	RST	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
80h	0000-0000	<b>P0</b>	–	–	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	
81h	0000-0111	<b>SP</b>	SP								
82h	0000-0000	<b>DPL</b>	DPL								
83h	0000-0000	<b>DPH</b>	DPH								
84h	xxxx-xx00	<b>INTE2</b>	–	–	–	–	–	–	PWM1IE	PWM0IE	
85h	xxxx-0x00	<b>INTPORT</b>	–	–	–	–	P3IF	–	P1IF	POIF	
86h	xxxx-x000	<b>INTPWM</b>	–	–	–	–	–	–	PWM1IF	PWM0IF	
87h	0xxx-0000	<b>PCON</b>	SMOD	–	–	–	GF1	GF0	PD	IDL	
88h	0000-0000	<b>TCON</b>	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
89h	0000-0000	<b>TMOD</b>	GATE1	CT1N	TMOD1		GATE0	CT0N	TMOD0		
8Ah	0000-0000	<b>TL0</b>	TL0								
8Bh	0000-0000	<b>TL1</b>	TL1								
8Ch	0000-0000	<b>TH0</b>	TH0								
8Dh	0000-0000	<b>TH1</b>	TH1								
8Eh	0100-0000	<b>SCON2</b>	SM	–	–	REN2	TB82	RB82	TI2	RI2	
8Fh	xxxx-xxxx	<b>SBUF2</b>	SBUF2								
90h	1111-1111	<b>P1</b>	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	
91h	xxxx-xx00	<b>PORTIDX</b>	–	–	–	–	–	–	PORTIDX		
93h	0000-0000	<b>UARTCON</b>	UART2PS				–	UARTPS			
94h	0000-0000	<b>OPTION</b>	–	TM3CKS	WDTPSC		ADCKS		PWM1NMSK	PWM1PMSK	
95h	0000-xx00	<b>INTFLG</b>	LVDIF	EEPIF	EEDBUSY	ADIF	–	–	PCIF	TF3	
96h	0000-0000	<b>INTPIN</b>	PIN7IF	PIN6IF	PIN5IF	PIN4IF	PIN3IF	PIN2IF	PIN1IF	PIN0IF	
97h	xxxx-xx00	<b>SWCMD</b>	SWRST / IAPALL / WDTO								
98h	0000-0000	<b>SCON</b>	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	
99h	xxxx-xxxx	<b>SBUF</b>	SBUF								
9Ah	xxxx-xxxx	<b>EEPWD0</b>	EEPWD0								
9Bh	xxxx-xxxx	<b>EEPWD1</b>	EEPWD1								
9Ch	xxxx-xxxx	<b>EEPWD2</b>	EEPWD2								
9Dh	xxxx-xxxx	<b>EEPWD3</b>	EEPWD3								
9Eh	0000-0000	<b>EEPWADR</b>	EEPWADR								
9Fh	0000-0000	<b>UART2CON</b>	–	UART2BRP							
A0h	1111-1111	<b>P2</b>	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	
A1h	xx10-1010	<b>PWMCON</b>	PWM1CKS		PWM1EN	PWM0EN	PWM0CKS		PWM0NMSK	PWM0PMSK	
A2h	0001-0001	<b>PINMOD10</b>	PINMOD1				PINMOD0				
A3h	0001-0001	<b>PINMOD32</b>	PINMOD3				PINMOD2				
A4h	0001-0001	<b>PINMOD54</b>	PINMOD5				PINMOD4				
A5h	0001-0001	<b>PINMOD76</b>	PINMOD7				PINMOD6				
A6h	0000-0000	<b>PINMOD</b>	HSNK2EN	HSNK1EN	HSNK0EN	I2CPS	–	TCOE	T2OE	T0OE	
A7h	0000-0000	<b>PWMCON2</b>	PWM0MOD	PWM0MSKE	PWM0OM		PWM0DZ				
A8h	0x00-0000	<b>IE</b>	EA	–	ET2	ES	ET1	EX1	ET0	EX0	
A9h	0000-0000	<b>INTE1</b>	PWMIE	I2CE	ES2	EEPIE	ADIE	LVDIE	PCIE	TM3IE	
AAh	xxxx-xxxx	<b>ADCDL</b>	ADCDL				–				
ABh	xxxx-xxxx	<b>ADCDH</b>	ADCDH								
ACh	xxxx-x000	<b>EEPWCON</b>	–	–	–	–	–	HWSTART	hw_len		
AFh	0000-0000	<b>PWMCON3</b>	PWM1MOD	PWM1MSKE	PWM1OM		PWM1DZ				
B0h	1111-1111	<b>P3</b>	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	

Adr	RST	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
B6h	xxx1-1111	ADCHSEL	ADCHS					ADCVREFS	VBGSEL		
B8h	xx00-0000	IP	-	-	PT2	PS	PT1	PX1	PT0	PX0	
B9h	xx00-0000	IPH	-	-	PT2H	PSH	PT1H	PX1H	PT0H	PX0H	
BAh	000x-0000	IP1	PPWM	PI2C	PS2	PEEP	PADI	PLVD	PPC	PT3	
BBh	000x-0000	IP1H	PPWMH	PI2CH	PS2H	PEEPH	PADIH	PLVDH	PPCH	PT3H	
BFh	000x-0000	LVDS	LVDM	LVDO	-	LVDPD	LVDSEL				
C1h	0000-0000	PWM4DH	PWM4DH								
C2h	0000-0000	PWM4DL	PWM4DL								
C3h	0000-0000	PWM5DH	PWM5DH								
C4h	0000-0000	PWM5DL	PWM5DL								
C8h	0000-0000	T2CON	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2N	CPRL2N	
C9h	0000-0000	IAPWE_SFR	IAPWE / IAPTO / EEPWE / INFOWE / EEPTO								
CAh	0000-0000	RCP2L	RCP2L								
CBh	0000-0000	RCP2H	RCP2H								
CCh	0000-0000	TL2	TL2								
CDh	0000-0000	TH2	TH2								
CEh	0000-0000	EXA2	EXA2								
CFh	0000-0000	EXA3	EXA3								
D0h	0000-0000	PSW	CY	AC	F0	RS1	RS0	OV	F1	P	
D1h	0000-0000	PWM0DH	PWM0DH								
D2h	0000-0000	PWM0DL	PWM0DL								
D3h	0000-0000	PWM1DH	PWM1DH								
D4h	0000-0000	PWM1DL	PWM1DL								
D5h	0000-0000	PWM2DH	PWM2DH								
D6h	0000-0000	PWM2DL	PWM2DL								
D8h	00x0-0011	CLKCON	-	-	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC		
D9h	1111-1111	PWM0PRDH	PWM0PRDH								
DAh	1111-1111	PWM0PRDL	PWM0PRDL								
DBh	1111-1111	PWM1PRDH	PWM1PRDH								
DCh	1111-1111	PWM1PRDL	PWM1PRDL								
DDh	0000-0000	PWM3DH	PWM3DH								
DEh	0000-0000	PWM3DL	PWM3DL								
DFh	0000-0011	RDCTLL	-	-	-	-	-	ATDEN	ATDT		
E0h	0000-0000	ACC	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	
E1h	000x-0100	MICON	MIEN	MIACKO	MIIF	MIACKI	MISTART	MISTOP	MICR		
E2h	0000-0000	MIDAT	MIDAT								
E3h	x100-0000	LVRCON	-	PORPD_SAV	PORPD	LVRPD	LVRSEL				
E4h	xxxx-xxxx	CFGBG2	-	-	-	BGTRIM2					
E5h	0000-0x00	EFTCON	EFT2CS	EFT1CS	EFT1S		EFTSLOW	-	EFTWOUT	CKHLDE	
E6h	0000-0000	EXA	EXA								
E7h	0000-0000	EXB	EXB								
EFh	xx00-0000	AUX3	warmtime	TM3PSC				-	FJMPE	FJMPS	
F0h	0000-0000	B	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	
F1h	1111-1111	CRCDL	CRCDL								
F2h	1111-1111	CRCDH	CRCDH								
F3h	0000-0000	CRCIN	CRCIN								
F5h	xxxx-xxxx	CFGBG	-	-	-	BGTRIM1					
F6h	xxxx-xxxx	CFGWL	-	FRCTRIM							
F7h	0000-0110	AUX2	WDTE		PWRSVAV	VBGOUT	DIV32	IAPTE		MULDIV16	
F8h	0010-1000	AUX1	CLRWDT	CLRTM3	CLRPWM0	ADSOC	CLRPWM1	T2SEL	T1SEL	DPSEL	

Flash Address	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
3FF9	<b>CFGBG2</b>	-	-	-	BGTRIM2				
3FFB	<b>CFGBG1</b>	-	-	-	BGTRIM1				
3FFD	<b>CFGWL</b>	-	FTCTRIM						
3FFFh	<b>CFGWH</b>	PROT	XRSTE	PORSEL					-



**SFR & CFGW DESCRIPTION**

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
80h	<b>P0</b>	7~0	P0	R/W	FFh	Port0 data
81h	<b>SP</b>	7~0	SP	R/W	07h	Stack Point
82h	<b>DPL</b>	7~0	DPL	R/W	00h	Data Point low byte
83h	<b>DPH</b>	7~0	DPH	R/W	00h	Data Point high byte
84h	<b>INTE2</b>	1	PWM1IE	R/W	0	PWM1 Interrupt Enable 0: disable PWM1 ~ PWM5 interrupt 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		0	PWM0IE	R/W	0	PWM0 Interrupt Enable 0: disable PWM0 interrupt 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
85h	<b>INTPORT</b>	3	P3IF	R/W	0	PORT3 Pin Change Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		1	P1IF	R/W	0	PORT1 Pin Change Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		0	P0IF	R/W	0	PORT0 Pin Change Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
86h	<b>INTPWM</b>	1	PWM1IF	R/W	0	PWM1~PWM5 interrupt Flag. Set by H/W at the end of PWM1 period, S/Wwrite 1h to clear int flag
		0	PWM0IF	R/W	0	PWM0 Interrupt Flag. Set by H/W at the end of PWM0 period, S/Wwrite 2h to clear int flag
87h	<b>PCON</b>	7	SMOD	R/W	0	Set 1 to enable UART1 double baud rate
		3	GF1	R/W	0	General purpose flag bit
		2	GF0	R/W	0	General purpose flag bit
		1	PD	R/W	0	Power down control bit, set 1 to enter Halt/Stop mode
		0	IDL	R/W	0	Idle control bit, set 1 to enter Idle mode
88h	<b>TCON</b>	7	TF1	R/W	0	Timer1 overflow flag Set by H/W when Timer/Counter 1 overflows. Cleared by H/W when CPU vectors into the interrupt service routine.
		6	TR1	R/W	0	Timer1 run control. 1: timer runs; 0: timer stops
		5	TF0	R/W	0	Timer0 overflow flag Set by H/W when Timer/Counter 0 overflows. Cleared by H/W when CPU vectors into the interrupt service routine.
		4	TR0	R/W	0	Timer0 run control. 1:timer runs; 0:timer stops
		3	IE1	R/W	0	External Interrupt 1 (INT1 pin) edge flag Set by H/W when an INT1 pin falling edge is detected. Cleared by H/W when CPU vectors into the interrupt service routine.
		2	IT1	R/W	0	External Interrupt 1 control bit 0: Low level active (level triggered) for INT1 pin 1: Falling edge active (edge triggered) for INT1 pin
		1	IE0	R/W	0	External Interrupt 0 (INT0 pin) edge flag Set by H/W when an INT0 pin falling edge is detected. Cleared by H/W when CPU vectors into the interrupt service routine.
		0	IT0	R/W	0	External Interrupt 0 control bit 0: Low level active (level triggered) for INT0 pin 1: Falling edge active (edge triggered) for INT0 pin

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
89h	<b>TMOD</b>	7	GATE1	R/W	0	Timer1 gating control bit 0: Timer1 enable when TR1 bit is set 1: Timer1 enable only while the INT1 pin is high and TR1 bit is set
		6	CT1N	R/W	0	Timer1 Counter/Timer select bit 0: Timer mode, Timer1 data increases at 2 System clock cycle rate 1: Counter mode, Timer1 data increases at T1 pin's negative edge
		5~4	TMOD1	R/W	00	Timer1 mode select 00: 8-bit timer/counter (TH1) and 5-bit prescaler (TL1) 01: 16-bit timer/counter 10: 8-bit auto-reload timer/counter (TL1). Reloaded from TH1 at overflow. 11: Timer1 stops
		3	GATE0	R/W	0	Timer0 gating control bit 0: Timer0 enable when TR0 bit is set 1: Timer0 enable only while the INTO pin is high and TR0 bit is set
		2	CT0N	R/W	0	Timer0 Counter/Timer select bit 0: Timer mode, Timer0 data increases at 2 System clock cycle rate 1: Counter mode, Timer0 data increases at T0 pin's negative edge
		1~0	TMOD0	R/W	00	Timer0 mode select 00: 8-bit timer/counter (TH0) and 5-bit prescaler (TL0) 01: 16-bit timer/counter 10: 8-bit auto-reload timer/counter (TL0). Reloaded from TH0 at overflow. 11: TL0 is an 8-bit timer/counter. TH0 is an 8-bit timer/counter using Timer1's TR1 and TF1 bits.
8Ah	<b>TL0</b>	7~0	TL0	R/W	00h	Timer0 data low byte
8Bh	<b>TL1</b>	7~0	TL1	R/W	00h	Timer1 data low byte
8Ch	<b>TH0</b>	7~0	TH0	R/W	00h	Timer0 data high byte
8Dh	<b>TH1</b>	7~0	TH1	R/W	00h	Timer1 data high byte
8Eh	<b>SCON2</b>	7	SM	R/W	0	UART2 Serial port mode select bit 0: Mode1: 8 bit UART2, Baud Rate is variable 1: Mode3: 9 bit UART2, Baud Rate is variable
		4	REN2	R/W	0	UART2 reception enable 0: Disable reception 1: Enable reception
		3	TB82	R/W	0	Transmit Bit 8, the ninth bit to be transmitted in Mode3
		2	RB82	R/W	0	Receive Bit 8, contains the ninth bit that was received in Mode3
		1	TI2	R/W	0	Transmit interrupt flag Set by H/W at the beginning of the stop bit in Mode 1 & 3. Must be cleared by S/W.
		0	RI2	R/W	0	Receive interrupt flag Set by H/W at the sampling point of the stop bit in Mode 1 & 3. Must be cleared by S/W.
8Fh	<b>SBUF2</b>	7~0	SBUF2	R/W	-	UART2 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.
90h	<b>P1</b>	7~0	P1	R/W	FFh	Port1 data
91h	<b>PORTIDX</b>	1~0	PORTIDX	R/W	00	Port index of INTPIN, PINMOD10, PINMOD32, PINMOD54, PINMOD76
93h	<b>UARTCON</b>	7~4	UART2PS	R/W	0000	UART2 Pin Select 0000: RXD2/TXD2 = P0.0/P0.1 0001: RXD2/TXD2 = P3.5/P3.6 0010: RXD2/TXD2 = P0.1/P0.0 0011: RXD2/TXD2 = P3.6/P3.5 0100: RXD2/TXD2 = P0.1/P0.1; 1-wire

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
						0101: RXD2/TXD2 = P3.6/P3.6; 1-wire 0110: RXD2/TXD2 = P0.0/P0.0; 1-wire 0111: RXD2/TXD2 = P3.5/P3.5; 1-wire 1000: RXD2/TXD2 = P0.2/P0.3 1010: RXD2/TXD2 = P0.3/P0.2 1100: RXD2/TXD2 = P0.3/P0.3; 1-wire 1110: RXD2/TXD2 = P0.2/P0.2; 1-wire
		2~0	UARTPS	R/W	000	UART1 Pin Select 000: RXD/TXD = P3.0/P3.1 001: RXD/TXD = P3.2/P3.3 010: RXD/TXD = P3.1/P3.0 011: RXD/TXD = P3.3/P3.2 100: RXD/TXD = P3.1/P3.1; 1-wire 101: RXD/TXD = P3.3/P3.3; 1-wire 110: RXD/TXD = P3.0/P3.0; 1-wire 111: RXD/TXD = P3.2/P3.2; 1-wire
94h	OPTION	6	TM3CKS	R/W	0	Timer3 Clock Source Select. 0: Slow clock (SRC) 1: FRC/512 (36KHz)
		5~4	WDTPSC	R/W	00	Watchdog Timer pre-scalar time select 00: 220 ms WDT overflow rate 01: 110 ms WDT overflow rate 10: 54 ms WDT overflow rate 11: 27 ms WDT overflow rate
		3~2	ADCKS	R/W	00	ADC clock rate select 00: F <sub>SYSCLK</sub> /32 01: F <sub>SYSCLK</sub> /16 10: F <sub>SYSCLK</sub> /8 11: F <sub>SYSCLK</sub> /4
		1	PWM1NMSK	R/W	0	PWM1N Mask Data, while CLRPWM1=1 and PWM1MSKE=1
		0	PWM1PMSK	R/W	0	PWM1P Mask Data, while CLRPWM1=1 and PWM1MSKE=1
95h	INTFLG	7	LVDIF	R/W	-	Low Voltage Detect Interrupt flag Set by H/W when a low voltage occurs. S/W can write 7Fh to INTFLG to clear this flag if VCC is not at Low Voltage.
		6	EEPIF	R/W	0	EEP write finish interrupt Set by H/W when write EEP finish, H/W auto clear when enter Interrupt subroutine
		5	EEPBUSY	R	0	EEP Busy Flag is set high when write EEP
		4	ADIF	R/W	0	ADC interrupt flag Set by H/W at the end of ADC conversion. S/W writes EFh to INTFLG or sets the ADSOC bit to clear this flag.
		1	PCIF	R/W	0	Port0~Port3 Pin change interrupt flag Set by H/W when Port0~Port3 pin state change is detected and its interrupt enable bit is set. S/W can write 0 to clear all pin interrupt flags (Port0~Port3), it will also clear PIN0IF~PIN7IF and P0IF~P3IF.
		0	TF3	R/W	0	Timer3 Interrupt Flag Set by H/W when Timer3 reaches TM3PSC setting cycles. It is cleared automatically when the program performs the interrupt service routine. S/W can write FEh to INTFLG to clear this bit.

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
96h	INTPIN	7	PIN7IF	R/W	0	Px.7 pin change interrupt flag, Write 0 to clear Px.7 pin change interrupt flag port number (x) define by PORTIDX
		6	PIN6IF	R/W	0	Px.6 pin change interrupt flag, Write 0 to clear Px.6 pin change interrupt flag port number (x) define by PORTIDX
		5	PIN5IF	R/W	0	Px.5 pin change interrupt flag, Write 0 to clear Px.5 pin change interrupt flag port number (x) define by PORTIDX
		4	PIN4IF	R/W	0	Px.4 pin change interrupt flag, Write 0 to clear Px.4 pin change interrupt flag port number (x) define by PORTIDX
		3	PIN3IF	R/W	0	Px.3 pin change interrupt flag, Write 0 to clear Px.3 pin change interrupt flag port number (x) define by PORTIDX
		2	PIN2IF	R/W	0	Px.2 pin change interrupt flag, Write 0 to clear Px.2 pin change interrupt flag port number (x) define by PORTIDX
		1	PIN1IF	R/W	0	Px.1 pin change interrupt flag, Write 0 to clear Px.1 pin change interrupt flag port number (x) define by PORTIDX
		0	PIN0IF	R/W	0	Px.0 pin change interrupt flag, Write 0 to clear Px.0 pin change interrupt flag port number (x) define by PORTIDX
97h	SWCMD	7~0	SWRST	W		Write 56h to generate S/W Reset
		7~0	IAPALL	W		Write 65h to set IAPEN control flag; Write other value to clear IAPALL flag. It is recommended to clear it immediately after IAP access.
		1	WDTO	R	0	WatchDog Time-Out flag
		0	IAPALL	R	0	Flag indicates Flash memory can be accessed by IAP or not. 0: Disable Flash IAP 1: Enable Flash IAP
98h	SCON	7	SM0	R/W	0	UART1 Serial port mode select bit 0, 1 (SM0, SM1) = 00: Mode0: 8 bit shift register, Baud Rate= $F_{SYSCLK}/2$ 01: Mode1: 8 bit UART1, Baud Rate is variable 10: Mode2: 9 bit UART1, Baud Rate= $F_{SYSCLK}/32$ or $/64$ 11: Mode3: 9 bit UART1, Baud Rate is variable
		6	SM1	R/W	0	
		5	SM2	R/W	0	Serial port mode select bit 2 SM2 enables multiprocessor communication over a single serial line and modifies the above as follows. In Modes 2 & 3, if SM2 is set then the received interrupt will not be generated if the received ninth data bit is 0. In Mode 1, the received interrupt will not be generated unless a valid stop bit is received. In Mode 0, SM2 should be 0.
		4	REN	R/W	0	Set 1 to enable UART1 Reception
		3	TB8	R/W	0	Transmitter bit 8, ninth bit to transmit in Modes 2 and 3
		2	RB8	R/W	0	Receive Bit 8, contains the ninth bit that was received in Mode 2 and 3 or the stop bit is Mode 1 if SM2=0
		1	TI	R/W	0	Transmit Interrupt flag Set by H/W at the end of the eighth bit in Mode 0, or at the beginning of the stop bit in other modes. Must be cleared by S/W
		0	RI	R/W	0	Receive Interrupt flag Set by H/W at the end of the eighth bit in Mode 0, or at the sampling point of the stop bit in other modes. Must be cleared by S/W.
99h	SBUF	7~0	SBUF	R/W	-	UART1 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.
9Ah	EEPWD0	7~0	EEPWD0	W	-	H/W Write EEP 1 <sup>st</sup> byte DATA0, when EEPWCON=04h~07h

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
9Bh	EEPWD1	7~0	EEPWD1	W	–	H/W Write EEP 2 <sup>nd</sup> byte DATA1, when EEPWCON=05h~07h
9Ch	EEPWD2	7~0	EEPWD2	W	–	H/W Write EEP 3 <sup>rd</sup> byte DATA2, when EEPWCON=06h~07h
9Dh	EEPWD3	7~0	EEPWD3	W	–	H/W Write EEP 4 <sup>th</sup> byte DATA3, when EEPWCON=07h
9Eh	EEPWADR	7~0	EEPWADR	W	0	H/W Write EEP start address (00h~FFh),
A0h	P2	7~0	P2	R/W	FFh	P2 data
A1h	PWMCON	7~6	PWM1CKS	R/W	00	PWM1~5 clock source 00: F <sub>SYSCLK</sub> 01: FRC/256 10: FRC 11: FRC x 2 (V <sub>CC</sub> > 3.0V)
		5	PWM1EN	R/W	0	PWM1~5 Enable. 0: PWM1~5 Disable 1: PWM1~5 enable
		4	PWM0EN	R/W	0	PWM0 Enable. 0: PWM0 Disable 1: PWM0 enable
		3~2	PWM0CKS	R/W	00	PWM0 clock source 00: F <sub>SYSCLK</sub> 01: FRC/256 10: FRC 11: FRC x 2 (V <sub>CC</sub> > 3.0V)
		1	PWM0NSK	R/W	0	PWM0N Mask Data. while CLRPWM0=1 and PMW0MSKE=1
		0	PWM0PSK	R/W	0	PWM0P Mask Data. while CLRPWM0=1 and PMW0MSKE=1
A2h	PINMOD10	7~4	PINMOD1	R/W	0001	Px.1 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD0	R/W	0001	Px.0 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A3h	PINMOD32	7~4	PINMOD3	R/W	0001	Px.3 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD2	R/W	0001	Px.2 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A4h	PINMOD54	7~4	PINMOD5	R/W	0001	Px.5 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD4	R/W	0001	Px.4 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A5h	PINMOD76	7~4	PINMOD7	R/W	0000	Px.7 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD6	R/W	0001	Px.6 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A6h	PINMOD	7	HSNK2EN	R/W	0	Pin H-sink enable (Group 2: P3.0~P3.7) 0: Group 2 High-sink disable 1: Group 2 High-sink enable
		6	HSNK1EN	R/W	0	Pin H-sink enable (Group 1: P1.0~P1.7) 0: Group 1 High-sink disable 1: Group 1 High-sink enable
		5	HSNK0EN	R/W	0	Pin H-sink enable (Group 0: P0.0~P0.5) 0: Group 0 High-sink disable 1: Group 0 High-sink enable
		4	I2CPS	R/W	0	I <sup>2</sup> C Pin Select 0: SCL/SDA = P0.0/P0.1 1: SCL/SDA = P3.0/P3.1
		2	TCOE	R/W	0	"Instruction Cycle Clock" signal output to P1.4 pin 0: disable; 1: enable
		1	T2OE	R/W	0	"Timer2 overflow divided by 2" signal output to P1.0 pin 0: disable; 1: enable
		0	T0OE	R/W	0	"Timer0 overflow divided by 64" signal output to P3.4 pin 0: disable; 1: enable
A7h	PWMCON2	7	PWM0MOD	R/W	0	PWM0 mode select 0: Normal mode 1: Half-bridge mode
		6	PWM0MSKE	R/W	0	PWM0 mask output enable

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
						0: Disable 1: Enable, PWM0P/PWM0N output data by PWM0PMSK/PWM0NMSK while CLRPWM0=1
		5~4	PWM0OM	R/W	00	PWM0 output mode select 00: Mode0 01: Mode1 10: Mode2 11: Mode3
		3~0	PWM0DZ	R/W	0000	PWM0 dead zone (Dead zone is prohibited in half-bridge mode) 0000: 0 x T <sub>PWMCLK</sub> 0001: 1 x T <sub>PWMCLK</sub> ... 1111: 15 x T <sub>PWMCLK</sub>
A8h	IE	7	EA	R/W	0	Global interrupt enable control. 0: Disable all Interrupts. 1: Each interrupt is enabled or disabled by its own interrupt control bit.
		5	ET2	R/W	0	Set 1 to enable Timer2 interrupt
		4	ES	R/W	0	Set 1 to enable Serial Port (UART1) Interrupt
		3	ET1	R/W	0	Set 1 to enable Timer1 Interrupt
		2	EX1	R/W	0	Set 1 to enable external INT1 pin Interrupt & Halt/Stop mode wake up capability
		1	ET0	R/W	0	Set 1 to enable Timer0 Interrupt
		0	EX0	R/W	0	Set 1 to enable external INTO pin Interrupt & Halt/Stop mode wake up capability
A9h	INTE1	7	PWMIE	R/W	0	Set 1 to enable PWM0~PWM1 interrupt
		6	I2CE	R/W	0	Set 1 to enable I <sup>2</sup> C interrupt
		5	ES2	R/W	0	Set 1 to enable Serial Port (UART2) interrupt
		4	EEPIE	R/W	0	Set 1 to enable EEP write finish interrupt
		3	ADIE	R/W	0	Set 1 to enable ADC Interrupt
		2	LVDIE	R/W	0	Set 1 to enable LVD interrupt
		1	PCIE	R/W	0	Set 1 to enable Port0~Port3 Pin Change Interrupt
		0	TM3IE	R/W	0	Set 1 to enable Timer3 Interrupt
AAh	ADC <sub>DL</sub>	7~4	ADC <sub>DL</sub>	R	–	ADC data bit 3~0
ABh	ADC <sub>DH</sub>	7~0	ADC <sub>DH</sub>	R	–	ADC data bit 11~4
AEh	EEPWCON	2	HWSTART	W	0	HW write EEP Start control signal; 0: disable HW write EEP 1: Start to write Data (stored in 9A~9D)to EEP (adr. stored in 9E), this bit is auto clear when write finish
		1~0	HWLEN	W	0	HW write EEP data length when HWSTART is set to high 0: 1 byte; 1: 2 bytes; 2: 3 bytes; 3: 4 bytes
AFh	PWMCON3	7	PWM1MOD	R/W	0	PWM0 mode select 0: Normal mode 1: Half-bridge mode
		6	PWM1MSKE	R/W	0	PWM1 mask output enable 0: Disable 1: Enable, PWM1P/PWM1N output data by PWM1PMSK/PWM1NMSK while CLRPWM1=1
		5~4	PWM1OM	R/W	00	PWM1 output mode select 00: Mode0; 01: Mode1; 10: Mode2; 11: Mode3
		3~0	PWM1DZ	R/W	0000	PWM1 dead zone (Dead zone is prohibited in half-bridge mode) 0000: 0 x T <sub>PWMCLK</sub> 0001: 1 x T <sub>PWMCLK</sub> ... 1111: 15 x T <sub>PWMCLK</sub>
B0h	P3	7~0	P3	R/W	FFh	Port3 data

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
B6h	ADCHS	7~3	ADCHS	R/W	1Fh	ADC Channel Select 00000: CH0 (P0.0) 00001: CH1 (P0.1) 00010: CH2 (P0.2) 00011: CH3 (P0.3) 00100: CH4 (P0.4) 00101: CH5 (P0.5) 00110: CH6 (P3.7) 00111: CH7 (P3.6) 01000: CH8 (P3.5) 01001: CH9 (P3.4) 01010: CH10 (P1.0) 01011: CH11(P1.1) 01100: CH12 V <sub>BG</sub> (Internal Bandgap Reference Voltage) 01101: CH13 (P1.2) 01110: CH14 (P1.3) 01111: CH15 (P1.4) 10000: CH16 (P1.5) 10001: CH17 (P3.0) 10010: CH18 (P3.1) 10011: CH19 (P1.6) 10100: CH20 (P1.7) 10101: CH21 (P3.3) 10110: CH22 (P3.2) Others: reserved 11110: CH30(VCC/201) 11111: CH31(VCC/4)
		2	ADCVREFS	R/W	0	ADC reference voltage 0: V <sub>CC</sub> 1: VBG
		1~0	VBGSEL	R/W	00	VBG voltage select, When ADCVREF is selected as VBG 00: 1.20V; 01: 2.0V; 10: 2.5V; 11: 4.0V
B8h	IP	5	PT2	R/W	0	Timer2 Interrupt Priority Low bit
		4	PS	R/W	0	Serial Port (UART1) Interrupt Priority Low bit
		3	PT1	R/W	0	Timer1 Interrupt Priority Low bit
		2	PX1	R/W	0	External INT1 Pin Interrupt Priority Low bit
		1	PT0	R/W	0	Timer0 Interrupt Priority Low bit
		0	PX0	R/W	0	External INT0 Pin Interrupt Priority Low bit
B9h	IPH	5	PT2H	R/W	0	Timer2 Interrupt Priority High bit
		4	PSH	R/W	0	Serial Port (UART1) Interrupt Priority High bit
		3	PT1H	R/W	0	Timer1 Interrupt Priority High bit
		2	PX1H	R/W	0	External INT1 Pin Interrupt Priority High bit
		1	PT0H	R/W	0	Timer0 Interrupt Priority High bit
		0	PX0H	R/W	0	External INT0 Pin Interrupt Priority High bit
BAh	IP1	7	PPWM	R/W	0	PWM Interrupt Priority Low bit
		6	PI2C	R/W	0	I <sup>2</sup> C Interrupt Priority Low bit
		5	PS2	R/W	0	Serial Port (UART2) interrupt priority low bit
		4	PEEP	R/W	0	EEP Interrupt Priority Low bit
		3	PADI	R/W	0	ADC Interrupt Priority Low bit
		2	PLVD	R/W	0	LVD Interrupt Priority Low bit
		1	PPC	R/W	0	Port0~Port3 pin change Interrupt Priority Low bit
		0	PT3	R/W	0	Timer3 Interrupt Priority Low bit
BBh	IP1H	7	PPWMH	R/W	0	PWM Interrupt Priority High bit
		6	PI2CH	R/W	0	I <sup>2</sup> C Interrupt Priority High bit
		5	PS2H	R/W	0	Serial Port (UART2) interrupt priority high bit
		4	PEEPH	R/W	0	EEP Interrupt Priority High bit
		3	PADIH	R/W	0	ADC Interrupt Priority High bit
		2	PLVDH	R/W	0	LVD Interrupt Priority High bit
		1	PPCH	R/W	0	Port0~Port3 pin change Interrupt Priority High bit

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
		0	PT3H	R/W	0	Timer3 Interrupt Priority High bit
BFh	LVDS	7	LVDM	R/W	0	0: VCC < VLVD (LVDIF = 1 while LVDO = 1) 1: VCC > VLVD (LVDIF = 1 while LVDO = 0)
		6	LVDO	R	0	LVD real-time Output
		4	LVDPD	R/W	0	LVD Power Down. 0: LVD Enable, 1: LVD Disable
		3~0	LVDSSEL	R/W	0h	Low Voltage Detect (LVD) select. (step=0.16V) 0000: Set LVD at 1.79V 0001: Set LVD at 1.95V 0010: Set LVD at 2.11V 0011: Set LVD at 2.26V 0100: Set LVD at 2.40V 0101: Set LVD at 2.56V 0110: Set LVD at 2.71V 0111: Set LVD at 2.87V 1000: Set LVD at 3.03V 1001: Set LVD at 3.18V 1010: Set LVD at 3.32V 1011: Set LVD at 3.50V 1100: Set LVD at 3.63V 1101: Set LVD at 3.80V 1110: Set LVD at 3.94V 1111: Set LVD at 4.12V
		C1h	PWM4DH	7~0	PWM4DH	R/W
C2h	PWM4DL	7~0	PWM4DL	R/W	00h	PWM4 duty low byte
C3h	PWM5DH	7~0	PWM5DH	R/W	00h	PWM5 duty high byte
C4h	PWM5DL	7~0	PWM5DL	R/W	00h	PWM5 duty low byte



Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
C8h	T2CON	7	TF2	R/W	0	Timer2 overflow flag Set by H/W when Timer/Counter 2 overflows unless RCLK=1 or TCLK=1. This bit must be cleared by S/W.
		6	EXF2	R/W	0	T2EX interrupt pin falling edge flag Set when a capture or a reload is caused by a negative transition on T2EX pin if EXEN2=1. This bit must be cleared by S/W.
		5	RCLK	R/W	0	UART receive clock control bit 0: Use Timer1 overflow as receive clock for serial port in mode 1 or 3 1: Use Timer2 overflow as receive clock for serial port in mode 1 or 3
		4	TCLK	R/W	0	UART transmit clock control bit 0: Use Timer1 overflow as transmit clock for serial port in mode 1 or 3 1: Use Timer2 overflow as transmit clock for serial port in mode 1 or 3
		3	EXEN2	R/W	0	T2EX pin enable 0: T2EX pin disable 1: T2EX pin enable, it cause a capture or reload when a negative transition on T2EX pin is detected if RCLK=TCLK=0
		2	TR2	R/W	0	Timer2 run control 0:timer stops 1:timer runs
		1	CT2N	R/W	0	Timer2 Counter/Timer select bit 0: Timer mode, Timer2 data increases at 2 System clock cycle rate 1: Counter mode, Timer2 data increases at T2 pin's negative edge
		0	CPRL2N	R/W	0	Timer2 Capture/Reload control bit 0: Reload mode, auto-reload on Timer2 overflows or negative transitions on T2EX pin if EXEN2=1. 1: Capture mode, capture on negative transitions on T2EX pin if EXEN2=1. If RCLK=1 or TCLK=1, CPRL2N is ignored and timer is forced to auto-reload on Timer2 overflow.
C9h	IAPWE_SFR	7~0	IAPWE_SFR	W	-	Write 47h or 74h to set IAPWE flag; Write 47h can write 1 byte at once, write 74h can write 2 bytes at once. Write other value to clear IAPWE flag. It is recommended to clear it immediately after IAP write. Write A1h to set INFOWE flag; write other value to clear INFOWE flag. It is recommended to clear it immediately after IAP write. Write E2h to set EEPWE flag; write other value to clear EEPWE flag. It is recommended to clear it immediately after EEPROM write.
		7	IAPWE	R	-	Flag indicates Flash memory can be written by IAP or not 0: IAP Write disable 1: IAP Write enable
		6	IAPTO	R	0	Time-Out flag of IAP write / INFO write. Set by H/W when IAP or INFO write Time-out occurs. Cleared this flag by H/W when IAPWE=0 or INFOWE=0.
		5	EEPWE	R	0	Flag indicates EEPROM memory can be written or not 0: EEPROM Write disable 1: EEPROM Write enable
		4	INFOWE	R	0	Flag indicates INFO memory can be written or not 0: INFO IAP Write disable 1: INFO IAP Write enable
		3	EEPTO	R	0	Time-Out flag of EEPROM write Set by H/W when EEPROM write Time-out occurs. Cleared this flag by H/W when EEPWE=0.
CAh	RCP2L	7~0	RCP2L	R/W	00h	Timer2 reload/capture data low byte
CBh	RCP2H	7~0	RCP2H	R/W	00h	Timer2 reload/capture data high byte
CCh	TL2	7~0	TL2	R/W	00h	Timer2 data low byte

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
CDh	TH2	7~0	TH2	R/W	00h	Timer2 data high byte
CEh	EXA2	7~0	EXA2	R/W	00h	Expansion accumulator 2
CFh	EXA3	7~0	EXA3	R/W	00h	Expansion accumulator 3
D0h	PSW	7	CY	R/W	0	ALU carry flag
		6	AC	R/W	0	ALU auxiliary carry flag
		5	F0	R/W	0	General purpose user-definable flag
		4	RS1	R/W	0	Register Bank Select bit 1
		3	RS0	R/W	0	Register Bank Select bit 0
		2	OV	R/W	0	ALU overflow flag
		1	F1	R/W	0	General purpose user-definable flag
		0	P	R/W	0	Parity flag
D1h	PWM0DH	7~0	PWM0DH	R/W	00h	PWM0 duty high byte
D2h	PWM0DL	7~0	PWM0DL	R/W	00h	PWM0 duty low byte
D3h	PWM1DH	7~0	PWM1DH	R/W	00h	PWM1 duty high byte
D4h	PWM1DL	7~0	PWM1DL	R/W	00h	PWM1 duty low byte
D5h	PWM2DH	7~0	PWM2DH	R/W	00h	PWM2 duty high byte
D6h	PWM2DL	7~0	PWM2DL	R/W	00h	PWM2 duty low byte
D8h	CLKCON	5	STPSCK	R/W	1	Set 1 to stop Slow clock in PDOWN mode
		4	STPPCK	R/W	0	Set 1 to stop UART/Timer0/1/2 clock in Idle mode for current reducing.
		3	STPFCK	R/W	0	Set 1 to stop Fast clock for power saving in Slow/Idle mode. This bit can be changed only in Slow mode.
		2	SELFCK	R/W	0	System clock select. This bit can be changed only when STPFCK=0. 0: Slow clock 1: Fast clock
		1~0	CLKPSC	R/W	11	System clock prescaler. Effective after 16 clock cycles (Max.) delay. 00: System clock is Fast/Slow clock divided by 16 01: System clock is Fast/Slow clock divided by 4 10: System clock is Fast/Slow clock divided by 2 11: System clock is Fast/Slow clock divided by 1
D9h	PWM0PRDH	7~0	PWM0PRDH	R/W	FFh	PWM0 period high byte
DAh	PWM0PRDL	7~0	PWM0PRDL	R/W	FFh	PWM0 period low byte
DBh	PWM1PRDH	7~0	PWM1PRDH	R/W	FFh	PWM1/2/3/4/5 period high byte
DCh	PWM1PRDL	7~0	PWM1PRDL	R/W	FFh	PWM1/2/3/4/5 period low byte
DDh	PWM3DH	7~0	PWM3DH	R/W	00h	PWM3 duty high byte
DEh	PWM3DL	7~0	PWM3DL	R/W	00h	PWM3 duty low byte
DFh	RDCTL	2	ATDEN	R/W	0	Flash ATD(Address Transition Detection) read control enable; 0: Flash Read always=1 1: Flash Read use ATD (for power saving at at slow clock)
		1~0	ATDT	R/W	3h	ATD timing controlwhen ATDEN=1 0: 5.4ns@5V or 9.0ns@3V 1: 6.8ns@5V or 10.7ns@3V 2: 9.7ns@5V or 15.1ns@3V 3: 12.3ns@5V or 19.4ns@3V
E0h	ACC	7~0	ACC	R/W	00h	Accumulator

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
E1h	MICON	7	MIEN	R/W	0	Master I <sup>2</sup> C enable 0: disable 1: enable
		6	MIACKO	R/W	0	When Master I <sup>2</sup> C receive data, send acknowledge to I <sup>2</sup> C Bus 0: ACK to slave device 1: NACK to slave device
		5	MIIF	R/W	0	Master I <sup>2</sup> C Interrupt flag 0: write 0 to clear it 1: Master I <sup>2</sup> C transfer one byte complete
		4	MIACKI	R	-	When Master I <sup>2</sup> C transfer, acknowledgement form I <sup>2</sup> C bus (read only) 0: ACK received 1: NACK received
		3	MISTART	R/W	0	Master I <sup>2</sup> C Start bit 1: start I <sup>2</sup> C bus transfer
		2	MISTOP	R/W	1	Master I <sup>2</sup> C Stop bit 1: send STOP signal to stop I <sup>2</sup> C bus
		1~0	MICR	R/W	00	Master I <sup>2</sup> C (SCL) clock frequency selection 00: F <sub>SYSCLK</sub> /4 (ex. If F <sub>SYSCLK</sub> =16MHz, I <sup>2</sup> C clock is 4M Hz) 01: F <sub>SYSCLK</sub> /16 (ex. If F <sub>SYSCLK</sub> =16MHz, I <sup>2</sup> C clock is 1M Hz) 10: F <sub>SYSCLK</sub> /64 (ex. If F <sub>SYSCLK</sub> =16MHz, I <sup>2</sup> C clock is 250K Hz) 11: F <sub>SYSCLK</sub> /256 (ex. If F <sub>SYSCLK</sub> =16MHz, I <sup>2</sup> C clock is 62.5K Hz)
E2h	MIDAT	7~0	MIDAT	R/W	00	Master I <sup>2</sup> C data shift register (W): After Start and before Stop condition, write this register will resume transmission to I <sup>2</sup> C bus (R): After Start and before Stop condition, read this register will resume receiving from I <sup>2</sup> C bus
E3h	LVRCON	6	PORPD_SAV	R/W	1	0: POR disable at PDOWN 1: POR ebanle at PDOWN
		5	PORPD	R/W	0	POR Power Down. 0: POR Enable, 1: POR Disable
		4	LVRPD	R/W	0	LVR Power Down. 0: LVR Enable, 1: LVR Disable
		3~0	LVRSEL	R/W	0	Low Voltage Reset (LVR) select. (step=0.16V) 0000: Set LVR at 1.79V 0001: Set LVR at 1.95V 0010: Set LVR at 2.11V 0011: Set LVR at 2.26V 0100: Set LVR at 2.40V 0101: Set LVR at 2.56V 0110: Set LVR at 2.71V 0111: Set LVR at 2.87V 1000: Set LVR at 3.03V 1001: Set LVR at 3.18V 1010: Set LVR at 3.32V 1011: Set LVR at 3.50V 1100: Set LVR at 3.63V 1101: Set LVR at 3.80V 1110: Set LVR at 3.94V 1111: Set LVR at 4.12V
E4h	CFGBG2	4~0	BGTRIM2	R/W	-	VBG trimming2 value
E5h	EFTCON	7	EFT2CS	R/W	0	EFT2 Detector enable 0: Disable EFT2 1: Enable EFT2
		6	EFT1CS	R/W	0	EFT1 Detector enable 0: Disable EFT1 1: Enable EFT1
		5~4	EFT1S	R/W	00	EFT1 Detector sensitivity adjustment
		3	EFTSLOW	R/W	0	Force SYSCLK to SLOWCLK while EFT detected 0: Disable 1: Enable

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
		1	EFTWOUT	R/W	0	EFTWAIT output to pin 0: P3.6 = normal I/O 1: P3.6 = EFTWAIT
		0	CKHLDE	R/W	0	clock hold enable 0: Disable 1: Enable
E6h	<b>EXA</b>	7~0	EXA	R/W	00h	Expansion accumulator
E7h	<b>EXB</b>	7~0	EXB	R/W	00h	Expansion B register
EFh	<b>AUX3</b>	6~3	TM3PSC	R/W	0000	Timer3 Interrupt rate 0000: Timer3 Interrupt rate is 262144 Timer3 clock cycle 0001: Timer3 Interrupt rate is 131072 Timer3 clock cycle 0010: Timer3 Interrupt rate is 65536 Timer3 clock cycle 0011: Timer3 Interrupt rate is 32768 Timer3 clock cycle 0100: Timer3 Interrupt rate is 16384 Timer3 clock cycle 0101: Timer3 Interrupt rate is 8192 Timer3 clock cycle 0110: Timer3 Interrupt rate is 4096 Timer3 clock cycle 0111: Timer3 Interrupt rate is 2048 Timer3 clock cycle 1000: Timer3 Interrupt rate is 1024 Timer3 clock cycle 1001: Timer3 Interrupt rate is 512 Timer3 clock cycle 1010: Timer3 Interrupt rate is 256 Timer3 clock cycle 1011: Timer3 Interrupt rate is 128 Timer3 clock cycle 1100: Timer3 Interrupt rate is 64 Timer3 clock cycle 1101: Timer3 Interrupt rate is 32 Timer3 clock cycle 1110: Timer3 Interrupt rate is 16 Timer3 clock cycle 1111: Timer3 Interrupt rate is 8 Timer3 clock cycle
		1	FJMPE	R/W	0	FRC frequency auto-change enable 0: FRC frequency define by CFGWL 1: FRC frequency auto-change enable
		0	FJMPS	R/W	0	RC frequency auto-change selection 0: (trim+0, +1, +2, +3, +0, -1, -2, -3; Exchange trim value every 10us) 1: (trim+0, +2, +4, +6, +0, -2, -4, -6; Exchange trim value every 10us)
F0h	<b>B</b>	7~0	B	R/W	00h	B register
F1h	<b>CRCDL</b>	7~0	CRCDL	R/W	FFh	16-bit CRC data bit 7~0
F2h	<b>CRCDH</b>	7~0	CRCDH	R/W	FFh	16-bit CRC data bit 15~8
F3h	<b>CRCIN</b>	7~0	CRCIN	W	-	CRC input data
F5h	<b>CFGBG1</b>	4~0	BGTRIM1	R/W	-	V <sub>BG</sub> trimming1 value (Chip Reserved)
F6h	<b>CFGWL</b>	6~0	FRCTRIM	R/W	-	FRC frequency adjustment 00h: lowest frequency 7Fh: highest frequency
F7h	<b>AUX2</b>	7~6	WDTE	R/W	00	Watchdog Timer Reset control 0x: WDT disable 10: WDT enable in Fast/Slow mode, disable in Idle/Halt/Stop mode 11: WDT always enable
		5	PWRSVAV	R/W	0	Set 1 to reduce the chip's power consumption at Idle/Halt/Stop Mode.
		4	VBGOUT	R/W	0	Bandgap voltage output control 0: P3.2 as normal I/O 1: Bandgap voltage output to P3.2 pin
		3	DIV32	R/W	0	only active when MULDIV16 =1 0: instruction DIV as 16/16 bit division operation 1: instruction DIV as 32/16 bit division operation
		2~1	IAPTE	R/W	00	IAP/EEP watchdog timer enable 00: Disable 01: wait 6.9 ms trigger watchdog time-out flag 10: wait 27.5ms trigger watchdog time-out flag 11: wait 55 ms trigger watchdog time-out flag
		0	MULDIV16	R/W	0	0: instruction MUL/DIV as 8*8, 8/8 operation 1: instruction MUL/DIV as 16*16, 16/16 or 32/16 operation
F8h	<b>AUX1</b>	7	CLRWDT	R/W	0	Set 1 to clear WDT, H/W auto clear it at next clock cycle

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
		6	CLRTM3	R/W	0	Set 1 to clear Timer3, HW auto clear it at next clock cycle.
		5	CLRPWM0	R/W	0	PWM0 clear enable 0: PWM0 is running 1: PWM0 is cleared and held
		4	ADSOC	R/W	0	ADC Start of Conversion Set 1 to start ADC conversion. Cleared by H/W at the end of conversion. S/W can also write 0 to clear this flag.
		3	CLRPWM1	R/W	0	PWM1 clear enable 0: PWM1 is running 1: PWM1 is cleared and held
		2	T2SEL	R/W	0	Timer2 counter mode (CT2N=1) input select 0: P1.0 (T2) pin (8051standard) 1: Slow clock divide by 16 (SLOWCLK/16)
		1	T1SEL	R/W	0	Timer1 counter mode (CT1N=1) input select 0: P3.5 (T1) pin (8051 standard) 1: Slow clock divide by 16 (SLOWCLK/16)
		0	DPSEL	R/W	0	Active DPTR Select

Adr	Flash	Bit#	Bit Name	Description
3FF9h	<b>CFGBG2</b>	4~0	BGTRIM2	VBG2 adjustment. VBG2 is trimmed to 1.20V in chip manufacturing.
3FFBh	<b>CFGBG1</b>	4~0	BGTRIM1	VBG adjustment. VBG is trimmed to 2.5V in chip manufacturing.
3FFDh	<b>CFGWL</b>	6~0	FRCTRIM	FRC frequency adjustment. FRC is trimmed to 18.432 MHz in chip manufacturing.
		7	PROT	Flash Code Protect, 1=Protect
		6	XRSTE	External Pin Reset enable, 1=enable.
3FFFh	<b>CFGWH</b>	5	PORSEL	0: POR always on (when PORPD=0) 1: POR turn on 2ms (1/8duty when PORPD=0)
		4~0		Reserved

## INSTRUCTION SET

Instructions are 1, 2 or 3 bytes long as listed in the ‘byte’ column below. Each instruction takes 1~8 System clock cycles to execute as listed in the ‘cycle’ column below.

ARITHMETIC				
Mnemonic	Description	byte	cycle	opcode
ADD A,Rn	Add register to A	1	2	28-2F
ADD A,dir	Add direct byte to A	2	2	25
ADD A,@Ri	Add indirect memory to A	1	2	26-27
ADD A,#data	Add immediate to A	2	2	24
ADDC A,Rn	Add register to A with carry	1	2	38-3F
ADDC A,dir	Add direct byte to A with carry	2	2	35
ADDC A,@Ri	Add indirect memory to A with carry	1	2	36-37
ADDC A,#data	Add immediate to A with carry	2	2	34
SUBB A,Rn	Subtract register from A with borrow	1	2	98-9F
SUBB A,dir	Subtract direct byte from A with borrow	2	2	95
SUBB A,@Ri	Subtract indirect memory from A with borrow	1	2	96-97
SUBB A,#data	Subtract immediate from A with borrow	2	2	94
INC A	Increment A	1	2	04
INC Rn	Increment register	1	2	08-0F
INC dir	Increment direct byte	2	2	05
INC @Ri	Increment indirect memory	1	2	06-07
DEC A	Decrement A	1	2	14
DEC Rn	Decrement register	1	2	18-1F
DEC dir	Decrement direct byte	2	2	15
DEC @Ri	Decrement indirect memory	1	2	16-17
INC DPTR	Increment data pointer	1	4	A3
MUL AB	Multiply A by B	1	8/16	A4
DIV AB	Divide A by B	1	8/16/32	84
DA A	Decimal Adjust A	1	2	D4

LOGICAL				
Mnemonic	Description	byte	cycle	opcode
ANL A,Rn	AND register to A	1	2	58-5F
ANL A,dir	AND direct byte to A	2	2	55
ANL A,@Ri	AND indirect memory to A	1	2	56-57
ANL A,#data	AND immediate to A	2	2	54
ANL dir,A	AND A to direct byte	2	2	52
ANL dir,#data	AND immediate to direct byte	3	4	53
ORL A,Rn	OR register to A	1	2	48-4F
ORL A,dir	OR direct byte to A	2	2	45
ORL A,@Ri	OR indirect memory to A	1	2	46-47
ORL A,#data	OR immediate to A	2	2	44
ORL dir,A	OR A to direct byte	2	2	42
ORL dir,#data	OR immediate to direct byte	3	4	43
XRL A,Rn	Exclusive-OR register to A	1	2	68-6F
XRL A,dir	Exclusive-OR direct byte to A	2	2	65
XRL A,@Ri	Exclusive-OR indirect memory to A	1	2	66-67
XRL A,#data	Exclusive-OR immediate to A	2	2	64
XRL dir,A	Exclusive-OR A to direct byte	2	2	62
XRL dir,#data	Exclusive-OR immediate to direct byte	3	4	63
CLR A	Clear A	1	2	E4
CPL A	Complement A	1	2	F4

LOGICAL				
Mnemonic	Description	byte	cycle	opcode
SWAP A	Swap Nibbles of A	1	2	C4
RL A	Rotate A left	1	2	23
RLC A	Rotate A left through carry	1	2	33
RR A	Rotate A right	1	2	03
RRC A	Rotate A right through carry	1	2	13

DATA TRANSFER				
Mnemonic	Description	byte	cycle	opcode
MOV A,Rn	Move register to A	1	2	E8-EF
MOV A,dir	Move direct byte to A	2	2	E5
MOV A,@Ri	Move indirect memory to A	1	2	E6-E7
MOV A,#data	Move immediate to A	2	2	74
MOV Rn,A	Move A to register	1	2	F8-FF
MOV Rn,dir	Move direct byte to register	2	4	A8-AF
MOV Rn,#data	Move immediate to register	2	2	78-7F
MOV dir,A	Move A to direct byte	2	2	F5
MOV dir,Rn	Move register to direct byte	2	4	88-8F
MOV dir,dir	Move direct byte to direct byte	3	4	85
MOV dir,@Ri	Move indirect memory to direct byte	2	4	86-87
MOV dir,#data	Move immediate to direct byte	3	4	75
MOV @Ri,A	Move A to indirect memory	1	2	F6-F7
MOV @Ri,dir	Move direct byte to indirect memory	2	4	A6-A7
MOV @Ri,#data	Move immediate to indirect memory	2	2	76-77
MOV DPTR,#data	Move immediate to data pointer	3	4	90
MOVC A,@A+DPTR	Move code byte relative DPTR to A	1	8	93
MOVC A,@A+PC	Move code byte relative PC to A	1	8	83
MOVX A,@Ri	Move external data(A8) to A	1	8	E2-E3
MOVX A,@DPTR	Move external data(A16) to A	1	8	E0
MOVX @Ri,A	Move A to external data(A8)	1	8	F2-F3
MOVX @DPTR,A	Move A to external data(A16)	1	8	F0
PUSH dir	Push direct byte onto stack	2	4	C0
POP dir	Pop direct byte from stack	2	4	D0
XCH A,Rn	Exchange A and register	1	2	C8-CF
XCH A,dir	Exchange A and direct byte	2	2	C5
XCH A,@Ri	Exchange A and indirect memory	1	2	C6-C7
XCHD A,@Ri	Exchange A and indirect memory nibble	1	2	D6-D7

BOOLEAN				
Mnemonic	Description	byte	cycle	opcode
CLR C	Clear carry	1	2	C3
CLR bit	Clear direct bit	2	2	C2
SETB C	Set carry	1	2	D3
SETB bit	Set direct bit	2	2	D2
CPL C	Complement carry	1	2	B3
CPL bit	Complement direct bit	2	2	B2
ANL C,bit	AND direct bit to carry	2	4	82
ANL C,/bit	AND direct bit inverse to carry	2	4	B0
ORL C,bit	OR direct bit to carry	2	4	72
ORL C,/bit	OR direct bit inverse to carry	2	4	A0
MOV C,bit	Move direct bit to carry	2	2	A2
MOV bit,C	Move carry to direct bit	2	4	92

<b>BRANCHING</b>				
<b>Mnemonic</b>	<b>Description</b>	<b>byte</b>	<b>cycle</b>	<b>opcode</b>
ACALL addr 11	Absolute jump to subroutine	2	6	11-F1
LCALL addr 16	Long jump to subroutine	3	6	12
RET	Return from subroutine	1	6	22
RETI	Return from interrupt	1	6	32
AJMP addr 11	Absolute jump unconditional	2	6	01-E1
LJMP addr 16	Long jump unconditional	3	6	02
SJMP rel	Short jump (relative address)	2	6	80
JC rel	Jump on carry = 1	2	4 (or 6)	40
JNC rel	Jump on carry = 0	2	4 (or 6)	50
JB bit,rel	Jump on direct bit = 1	3	4 (or 6)	20
JNB bit,rel	Jump on direct bit = 0	3	4 (or 6)	30
JBC bit,rel	Jump on direct bit = 1 and clear	3	4 (or 6)	10
JMP @A+DPTR	Jump indirect relative DPTR	1	6	73
JZ rel	Jump on accumulator = 0	2	4 (or 6)	60
JNZ rel	Jump on accumulator ≠ 0	2	4 (or 6)	70
CJNE A,dir,rel	Compare A,direct, jump not equal relative	3	4 (or 6)	B5
CJNE A,#data,rel	Compare A,immediate, jump not equal relative	3	4 (or 6)	B4
CJNE Rn,#data,rel	Compare register,immediate, jump not equal relative	3	4 (or 6)	B8-BF
CJNE @Ri,#data,rel	Compare indirect,immediate, jump not equal relative	3	4 (or 6)	B6-B7
DJNZ Rn,rel	Decrement register, jump not zero relative	2	4 (or 6)	D8-DF
DJNZ dir,rel	Decrement direct byte, jump not zero relative	3	4 (or 6)	D5

<b>MISCELLANEOUS</b>				
<b>Mnemonic</b>	<b>Description</b>	<b>byte</b>	<b>cycle</b>	<b>opcode</b>
NOP	No operation	1	2	00

In the above table, an entry such as E8-EF indicates a continuous block of hex opcodes used for 8 different registers, the register numbers of which are defined by the lowest three bits of the corresponding code. Non-continuous blocks of codes, shown as 11-F1 (for example), are used for absolute jumps and calls with the top 3 bits of the code being used to store the top three bits of the destination address.



## ELECTRICAL CHARACTERISTICS

### 1. Absolute Maximum Ratings ( $T_A=25^\circ\text{C}$ )

Parameter	Rating	Unit
Supply voltage	$V_{SS} - 0.3 \sim V_{SS} + 5.5$	V
Input voltage	$V_{SS} - 0.3 \sim V_{CC} + 0.3$	
Output voltage	$V_{SS} - 0.3 \sim V_{CC} + 0.3$	
All pins output current high	-80	mA
All pins output current low	+150	
Maximum Operating Voltage	5.5	V
Operating temperature	-40 ~ +105	°C
Storage temperature	-65 ~ +150	

### 2. DC Characteristics ( $T_A=25^\circ\text{C}$ , $V_{CC}=2.2\text{V} \sim 5.5\text{V}$ )

Parameter	Symbol	Conditions	Min	Typ	Max	Unit	
Operating Voltage	$V_{CC}$	$F_{SYSCLK}=18.432\text{ MHz}$	2.4	-	5.5	V	
Input High Voltage	$V_{IH}$	All Input	$V_{CC}=5\text{V}$	$0.6V_{CC}$	-	-	V
			$V_{CC}=3\text{V}$	$0.6V_{CC}$	-	-	V
Input Low Voltage	$V_{IL}$	All Input	$V_{CC}=5\text{V}$	-	-	$0.2V_{CC}$	V
			$V_{CC}=3\text{V}$	-	-	$0.2V_{CC}$	V
I/O Port Source Current	$I_{OH}$	All Output (P0.0~P0.5, P1.0~P1.7, P3.0~P3.7)	$V_{CC}=5\text{V}$ , $V_{OH}=0.9V_{CC}$	6	11	-	mA
			$V_{CC}=5\text{V}$ , $V_{OH}=0.6V_{CC}$	10	33	-	
			$V_{CC}=3\text{V}$ , $V_{OH}=0.9V_{CC}$	2.5	4.8	-	
			$V_{CC}=3\text{V}$ , $V_{OH}=0.66V_{CC}$	5	13	-	
I/O Port Sink Current	$I_{OL}$	All Output, (P0.0~P0.5, P1.0~P1.7, P3.0~P3.7)	$V_{CC}=5\text{V}$ , $V_{OL}=0.1V_{CC}$ HSNKxEN=1	56	70	-	mA
			$V_{CC}=5\text{V}$ , $V_{OL}=0.1V_{CC}$ HSNKxEN=0	32	40	-	
			$V_{CC}=3\text{V}$ , $V_{OL}=0.1V_{CC}$ HSNKxEN=1	24	32	-	
			$V_{CC}=3\text{V}$ , $V_{OL}=0.1V_{CC}$ HSNKxEN=0	9	18	-	
Power Supply Current	$I_{DD}$	Fast mode $V_{CC}=5\text{V}$	FRC=18.432 MHz	-	7	-	mA
			FRC=9.216 MHz	-	4.5	-	
		Fast mode $V_{CC}=3\text{V}$	FRC=18.432 MHz	-	3.7	-	
			FRC=9.216 MHz	-	2.5	-	

Parameter	Symbol	Conditions	Min	Typ	Max	Unit	
		Slow mode	$V_{CC}=5V$	-	45	-	$\mu A$
			$V_{CC}=3V$	-	22	-	
		Idle mode PWRSAV=0 No Load	SRC, $V_{CC}=5V$ POR ON	-	16	-	$\mu A$
			SRC, $V_{CC}=5V$ LVR ON		68		
			SRC, $V_{CC}=3V$ POR ON	-	8	-	
			SRC, $V_{CC}=3V$ LVR ON		44		
		Idle mode PWRSAV=1 No Load	SRC, $V_{CC}=5V$	-	16	-	$\mu A$
			SRC, $V_{CC}=3V$	-	8	-	
		Stop mode PWRSAV=1 No Load	$V_{CC}=5V$	0.3	-	-	$\mu A$
			$V_{CC}=3V$	0.1	-	-	
		Halt mode PWRSAV=1 No Load	$V_{CC}=5V$ (Timer3=0.5 sec)	8	-	-	$\mu A$
			$V_{CC}=3V$ (Timer3=0.5 sec)	3	-	-	
System Clock Frequency	$F_{SYSCLK}$	$V_{CC} > LVR_{TH}$	$V_{CC}=2.2V$	-	-	18.432	MHz
LVR Reference Voltage	$V_{LVR}$	$T_A=25^{\circ}C$	-	4.12	-	V	
			-	3.94	-		
			-	3.80	-		
			-	3.63	-		
			-	3.50	-		
			-	3.32	-		
			-	3.18	-		
			-	3.03	-		
			-	2.87	-		
			-	2.71	-		
			-	2.56	-		
			-	2.40	-		
			-	2.26	-		
			-	2.11	-		
			-	1.95	-		
-	1.79	-					

Parameter	Symbol	Conditions		Min	Typ	Max	Unit
LVD Reference Voltage	$V_{LVD}$	$T_A=25^\circ\text{C}$		–	4.12	–	V
				–	3.94	–	
				–	3.80	–	
				–	3.63	–	
				–	3.50	–	
				–	3.32	–	
				–	3.18	–	
				–	3.03	–	
				–	2.87	–	
				–	2.71	–	
				–	2.56	–	
				–	2.40	–	
				–	2.26	–	
				–	2.11	–	
–	1.95	–					
–	1.79	–					
LVR Hysteresis Voltage	$V_{HYST}$	$T_A=25^\circ\text{C}$		–	$\pm 0.1$	–	V
Low Voltage Detection time	$t_{LVR}$	$T_A=25^\circ\text{C}$		100	–	–	$\mu\text{s}$
Pull-Up Resistor	$R_{PU}$	$V_{IN}=0\text{V}$	$V_{CC}=5\text{V}$	–	25	–	K $\Omega$
			$V_{CC}=3\text{V}$	–	25	–	
Pull-Down Resistor	$R_{PD}$	$V_{IN}=V_{CC}$	$V_{CC}=5\text{V}$	–	25	–	
			$V_{CC}=3\text{V}$	–	25	–	

**3. Clock Timing** ( $T_A = -40^\circ\text{C} \sim +105^\circ\text{C}$ )

Parameter	Condition	Min	Typ	Max	Unit
FRC Frequency	25°C, $V_{CC}=5.0\text{V}$	-1%	18.432	+1%	MHz
	-40°C ~ 105°C, $V_{CC}=5.0\text{V}$	-1.5%	18.432	+1.5%	
	-40°C ~ 105°C, $V_{CC}=3.0 \sim 5.0\text{V}$	-2.5%	18.432	+2.5%	

**4. Reset Timing Characteristics** ( $T_A = -40^\circ\text{C} \sim +105^\circ\text{C}$ )

Parameter	Conditions	Min	Typ	Max	Unit
RESET Input Low width	Input $V_{CC}=5\text{V} \pm 10\%$	30	-	-	$\mu\text{s}$
WDT wake up time	$V_{CC}=5\text{V}$ , WDT $PSC=11$	-	30	-	ms
	$V_{CC}=3\text{V}$ , WDT $PSC=11$	-	32	-	
CPU start up time	$V_{CC} = 5\text{V}$	-	13.6	-	ms

**5. ADC Electrical Characteristics** ( $T_A = 25^\circ\text{C}$ ,  $V_{CC} = 3.0\text{V} \sim 5.5\text{V}$ ,  $V_{SS} = 0\text{V}$ )

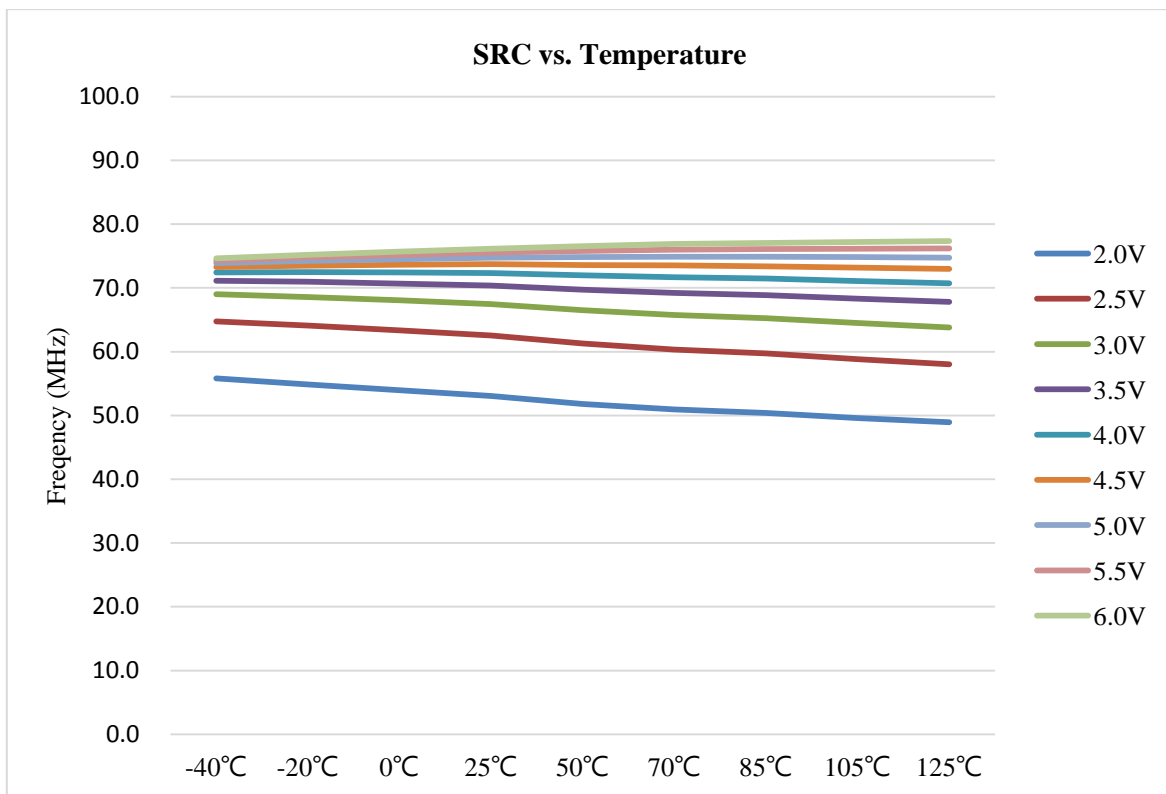
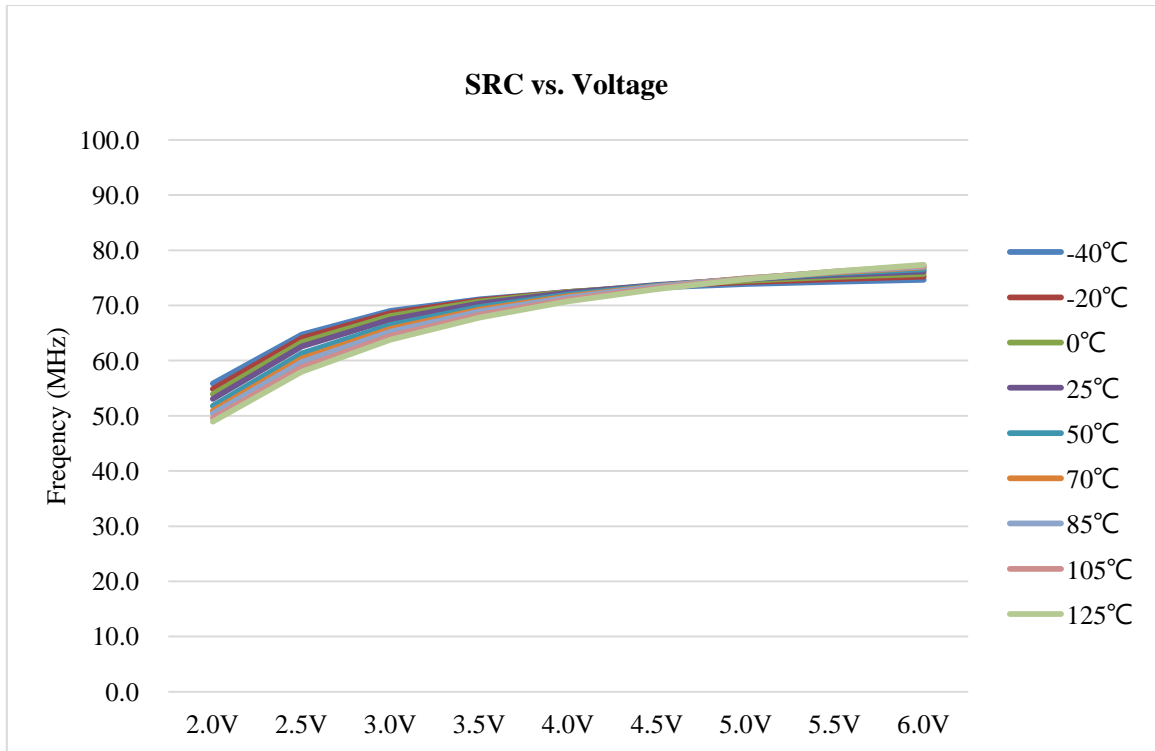
Parameter	Conditions	Min	Typ	Max	Unit	
Total Accuracy	$V_{CC}=5\text{V}$ , $V_{SS}=0\text{V}$	-	$\pm 2.5$	$\pm 4$	LSB	
Integral Non-Linearity		-	$\pm 3.2$	$\pm 5$		
Max Input Clock ( $f_{ADC}$ )	Source impedance ( $R_s < 10\text{K ohm}$ )	-	-	2	MHz	
	Source impedance ( $R_s < 20\text{K ohm}$ )	-	-	1		
	Source impedance ( $R_s < 50\text{K ohm}$ )	-	-	0.5		
	Source is $V_{BG}$ (ADCHS=01100b)	-	-	2.3		
Conversion Time	$F_{ADC} = 1\text{MHz}$	-	21	-	$\mu\text{s}$	
BandGap Voltage Reference ( $V_{BG}$ )	-	$V_{CC}=3\text{V} \sim 5.5\text{V}$ -40°C ~ 105°C	-1.5%	1.18	+1.5%	V
ADC Reference Voltage ( $V_{ADC}$ )	ADCVREFS=1	$V_{CC}=3\text{V} \sim 5.5\text{V}$ 40°C ~ 105°C	-1.5%	2.5	+1.5%	
$V_{CC}/4$ Reference Voltage ( $V_{1/4}$ )	-	$V_{CC}=5\text{V}$ , 25°C	-0.8%	1.26	+0.8%	
	-	$V_{CC}=3.6\text{V}$ , 25°C	-0.8%	0.907	+0.8%	
Input Voltage	-	$V_{SS}$	-	$V_{CC}$		

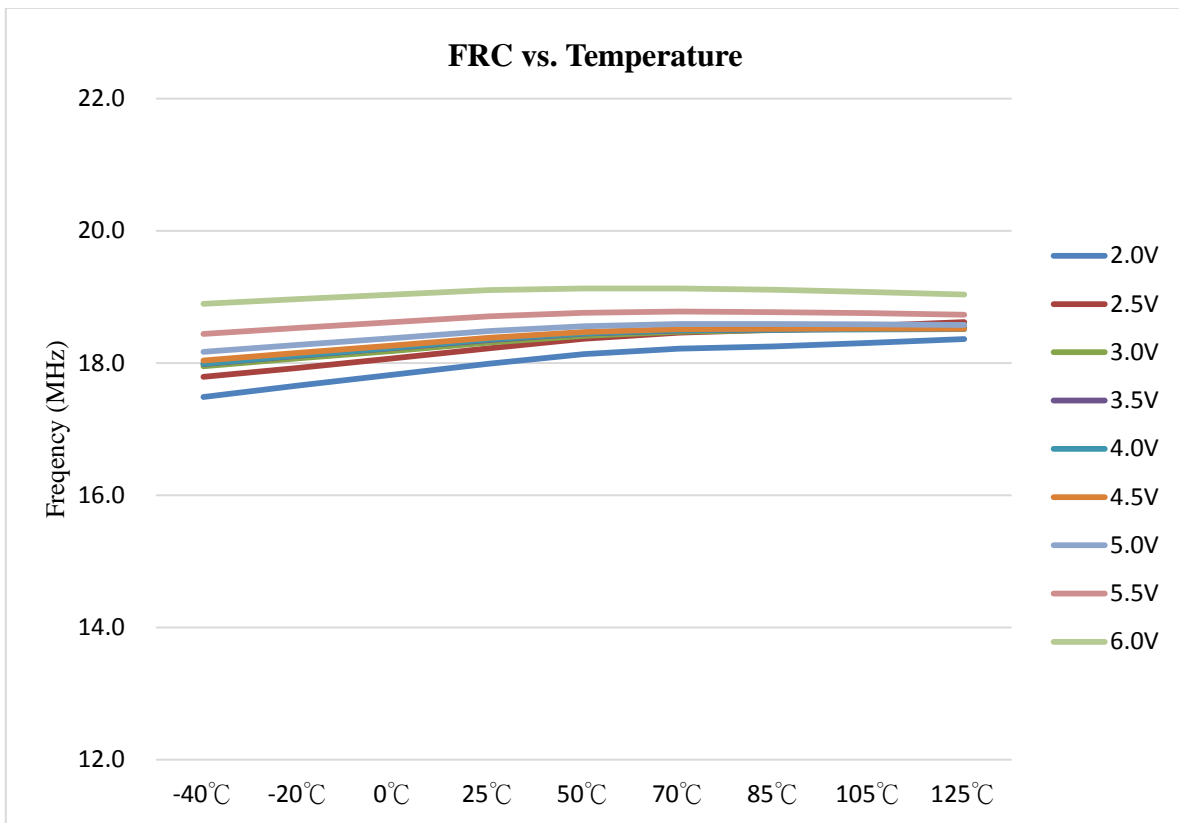
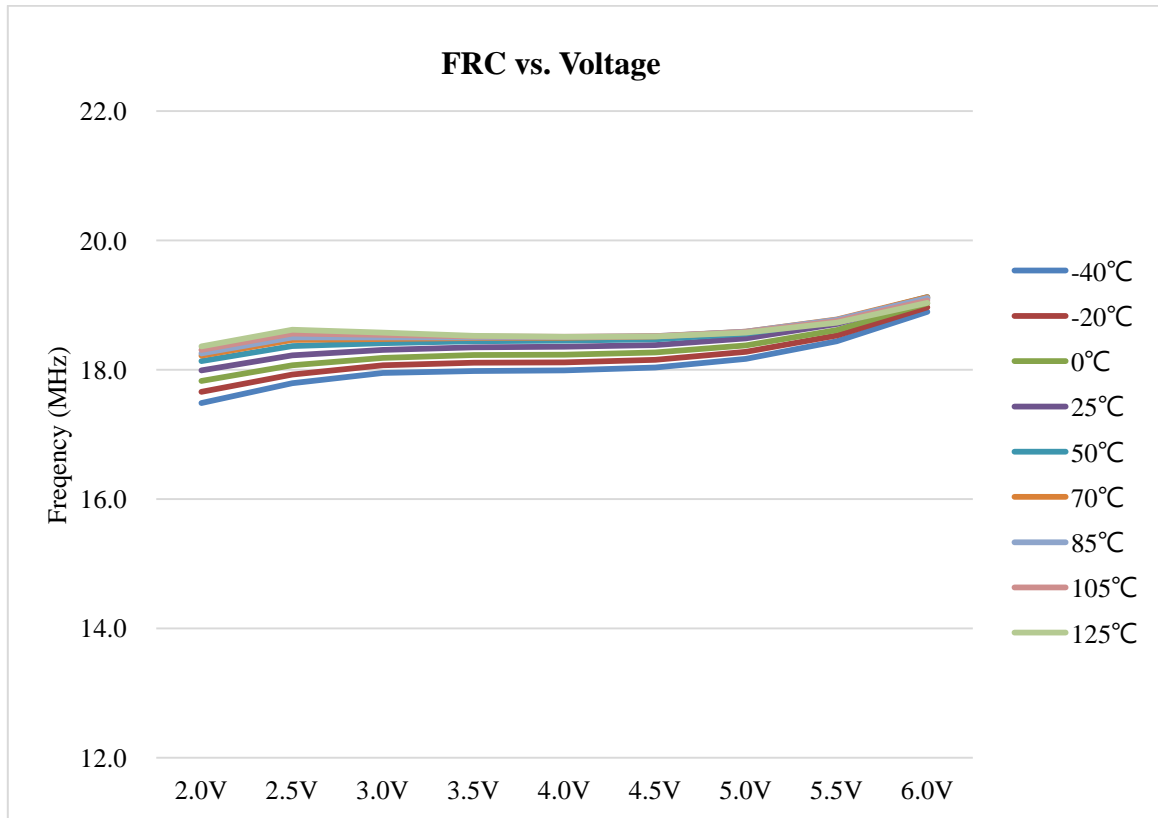
**6. EEPROM Characteristics**

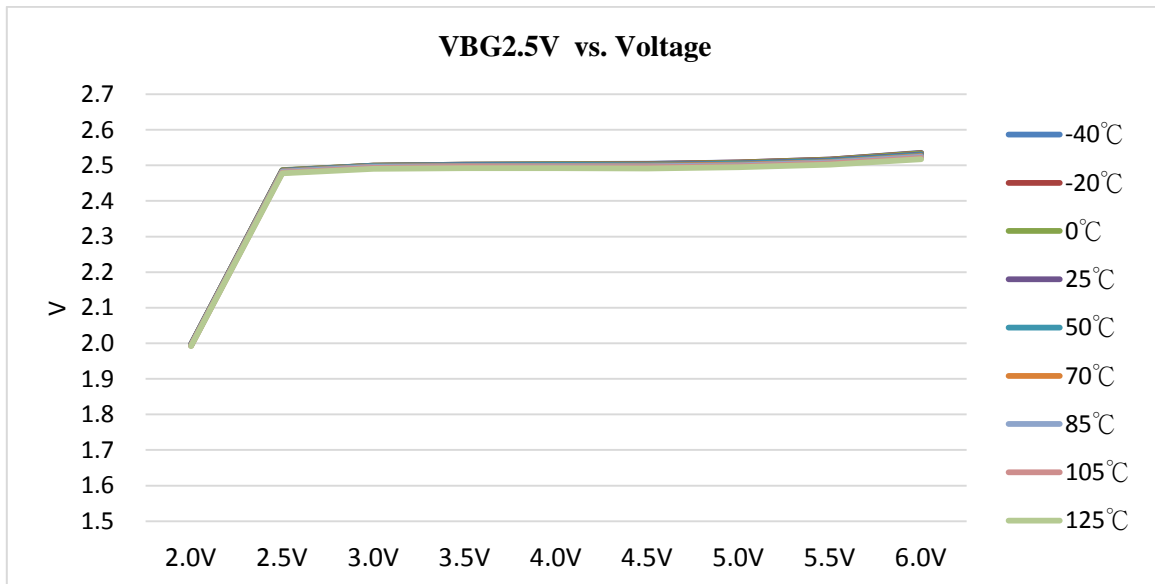
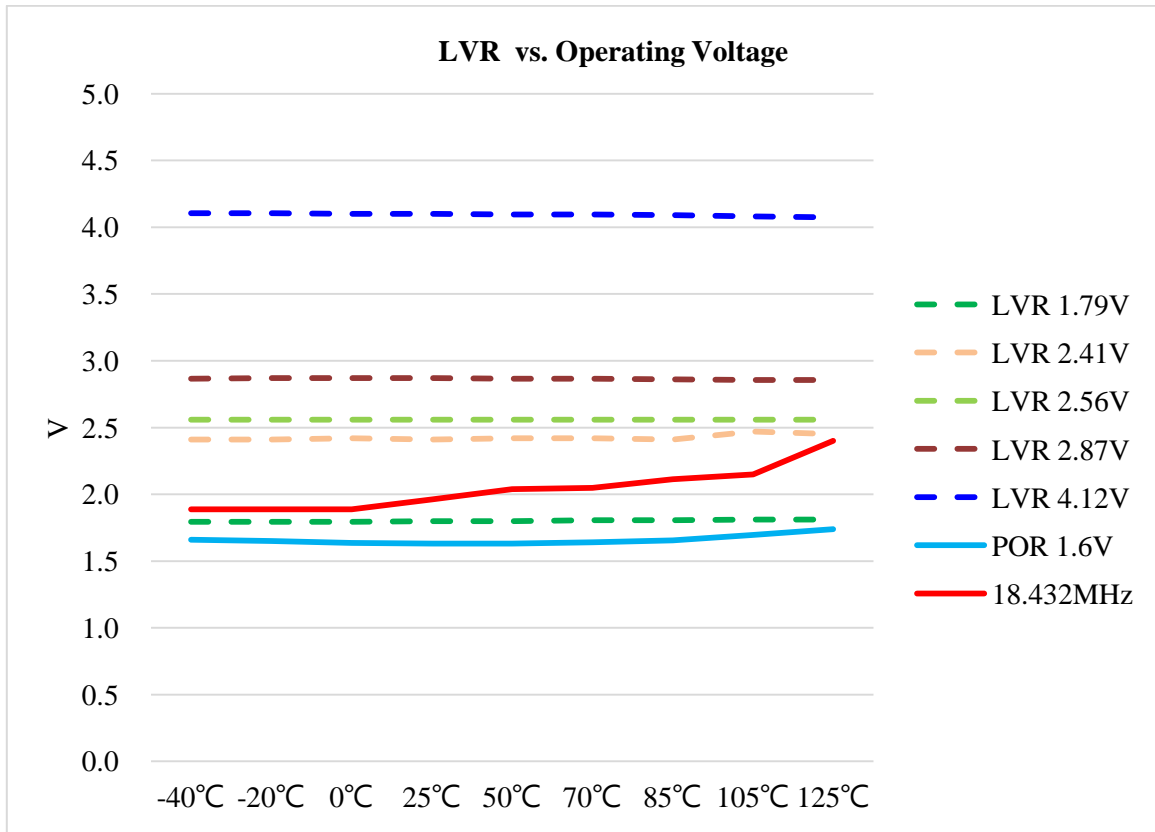
Parameter	Conditions	Min	Typ	Max	Unit
Write Voltage $V_{EEWR}$	-40°C ~ 105° $F_{sys}=FRC/1$ , $V_{CC}/47\mu\text{F}$	2.9		5.5	V
	-40°C ~ 105° $F_{sys}=FRC/2$ , $V_{CC}/47\mu\text{F}$	2.5		5.5	
Read Voltage $V_{EERD}$	-40°C ~ 105° $F_{sys}=FRC/1$ , $V_{CC}/47\mu\text{F}$	2.0		5.5	
*Write Endurance $N_{EE}$	$V_{CC}=2.5 \sim 5.5\text{V}$ , -40°C ~ 105°C	20K	-	-	cycles
	$V_{CC}=3.0 \sim 5.5\text{V}$ , -40°C ~ 105°C	30K	-	-	
	$V_{CC}=2.5 \sim 5.5\text{V}$ , -20°C ~ 85°C	30K			
Write Time $T_{EEWR}$	$V_{CC}=5.0\text{V}$ , 25°C, WDT disable		1.5		mS
	$V_{CC}=2.5\text{V}$ , 25°C, WDT disable		4		
	$V_{CC}=3.0\text{V}$ , 105°C, WDT disable		15		
Data Retention $Y_{RET}$		10			Year

Note: The value of this parameter is based on the characteristics of tested samples.

### 7. Characteristic Graphs







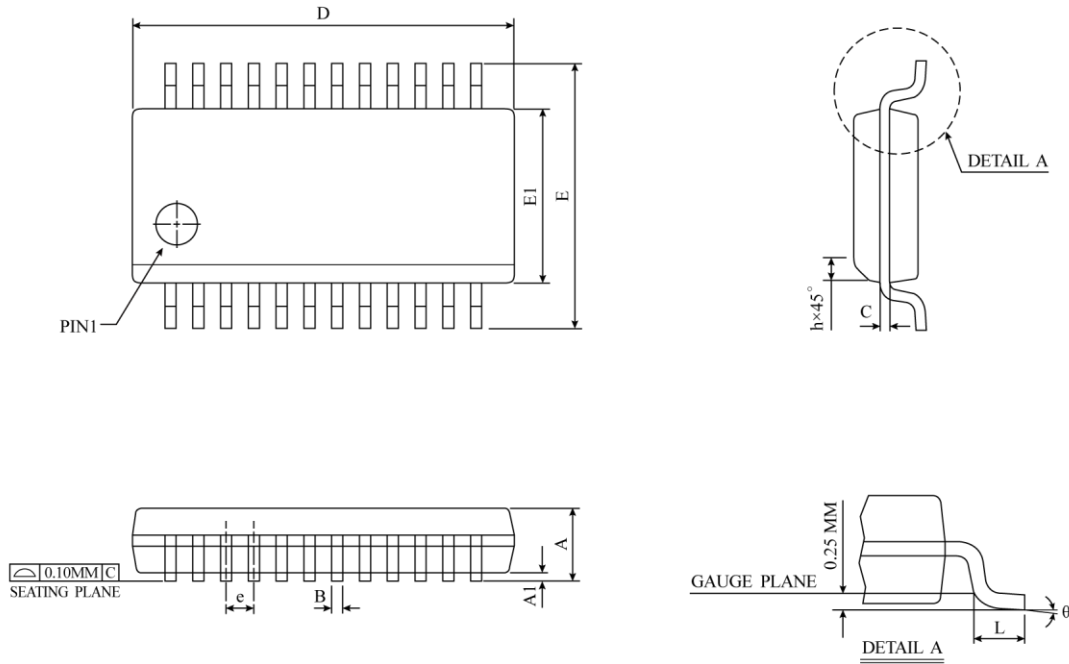
## Package and Dice Information

Please note that the package information provided is for reference only. Since this information is frequently updated, users can contact Sales to consult the latest package information and stocks.

### Ordering information

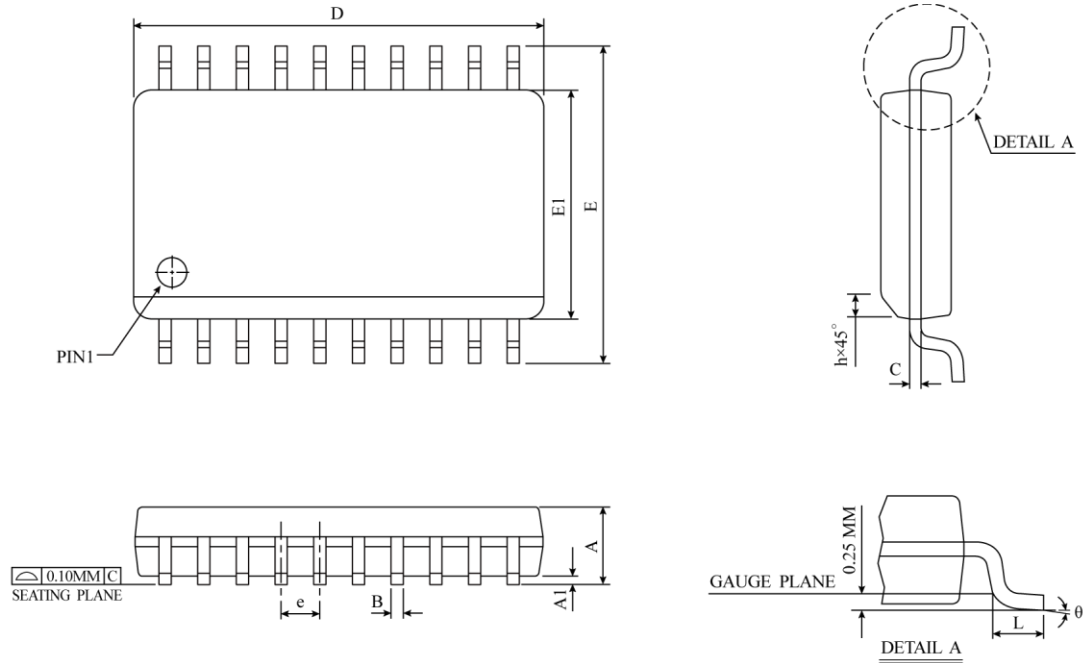
Ordering number	Package
TM52F5024-MTP	Wafer/Dice blank chip
TM52F5024-COD	Wafer/Dice with code
TM52F50243T	TSSOP20 (173 mil)
TM52F50244E	SSOP 24-pin (150 mil)
TM52F50244E1	
TM52F50244E2	
TM52F50243E	SSOP 20-pin (150 mil)
TM52F50243S	SOP 20-pin(300 mil)
TM52F50243S1	
TM52F50242S	SOP 16-pin (150 mil)
TM52F1354H	



**Package Information**
**SSOP-24 ( 150mil ) Package Dimension**


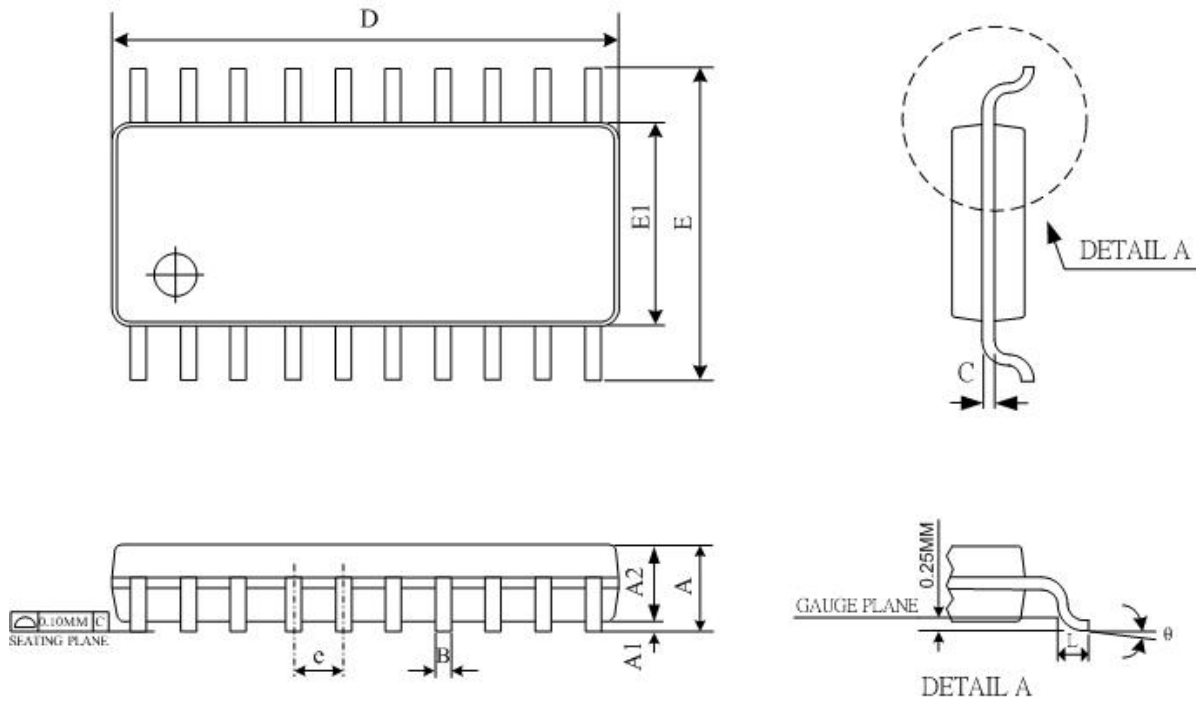
SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	1.35	1.55	1.75	0.053	0.061	0.069
A1	0.10	0.18	0.25	0.004	0.007	0.010
A2	-	-	1.50	-	-	0.059
B	0.20	0.25	0.30	0.008	0.010	0.012
C	0.18	0.22	0.25	0.007	0.009	0.010
D	8.56	8.65	8.74	0.337	0.341	0.344
E	5.79	6.00	6.20	0.228	0.236	0.244
E1	3.81	3.90	3.99	0.150	0.154	0.157
e	0.635 BSC			0.025 BSC		
L	0.41	0.84	1.27	0.016	0.033	0.050
θ	0°	4°	8°	0°	4°	8°
JEDEC	M0-137 (AE)					

$\Delta$  \* NOTES : DIMENSION " D " DOES NOT INCLUDE MOLD PROTRUSIONS OR GAT BURRS.  
 MOLD PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED 0.006 INCH PER SIDE.

**SOP-20 ( 300mil ) Package Dimension**


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.35	2.50	2.65	0.0926	0.0985	0.1043
A1	0.10	0.20	0.30	0.0040	0.0079	0.0118
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.23	0.28	0.32	0.0091	0.0108	0.0125
D	12.60	12.80	13.00	0.4961	0.5040	0.5118
E	10.00	10.33	10.65	0.3940	0.4425	0.4910
E1	7.40	7.50	7.60	0.2914	0.2953	0.2992
e	1.27 BSC			0.050 BSC		
h	0.25	0.50	0.75	0.0100	0.0195	0.0290
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
θ	0°	4°	8°	0°	4°	8°
JEDEC	MS-013 (AC)					

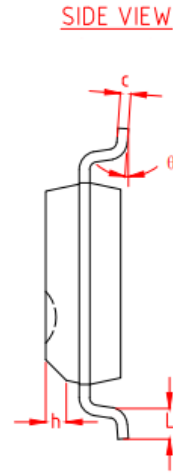
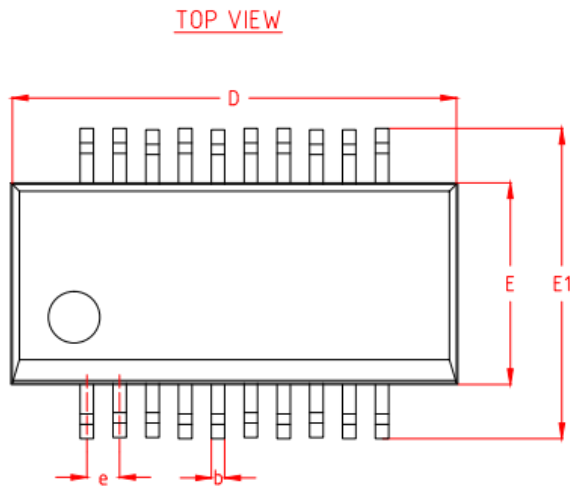
△ \*NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.  
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL  
NOT EXCEED 0.15 MM ( 0.006 INCH ) PER SIDE.

**TSSOP-20 ( 173mil ) Package Dimension**


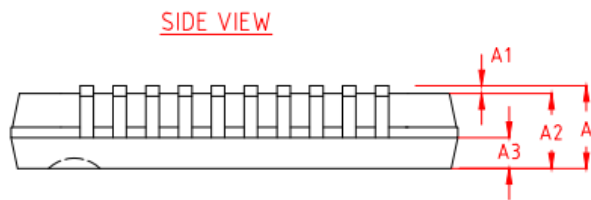
SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	-	-	1.2	-	-	0.047
A1	0.05	0.10	0.15	0.002	0.004	0.006
A2	0.8	0.93	1.05	0.031	0.036	0.041
B	0.19	-	0.3	0.007	-	0.012
D	6.4	6.5	6.6	0.252	0.256	0.260
E	6.25	6.4	6.55	0.246	0.252	0.258
E1	4.3	4.4	4.5	0.169	0.173	0.177
e	0.65 BSC			0.026 BSC		
L	0.45	0.60	0.75	0.018	0.024	0.030
θ	0 °		8 °	0 °		8 °
JEDEC	MO-153 AC REV.F					

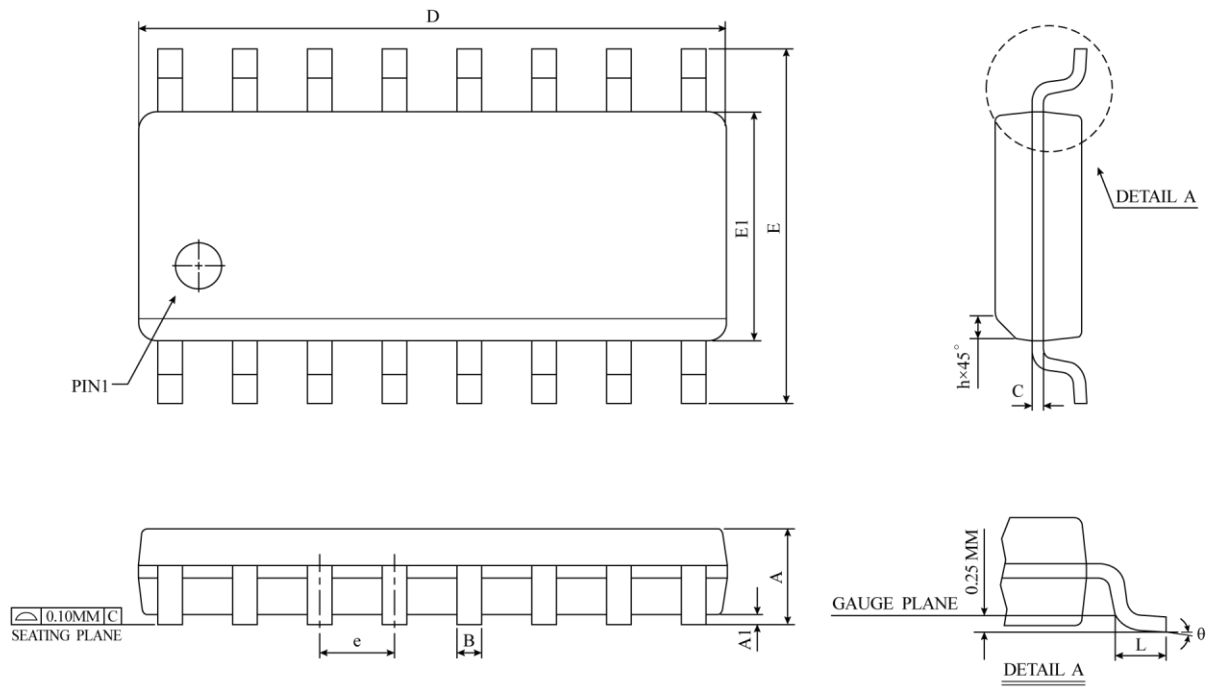
**Notes :**

- 1.DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 PER SIDE.
- 2.DIMENSION "E1" DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 PER SIDE.
- 3.DIMENSION "B" DOES NOT INCLUDE DAMBAR PROTRUSION.ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08MM TOTAL IN EXCESS OF THE "B" DIMENSION AT MAXIMUM METERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD IS 0.07MM.

**SSOP-20 ( 150mil ) Package Dimension**


Dimensions/mm			
字符 SYMBOL	最小值 MIN	典型值 NOMINAL	最大值 MAX
A	-	-	1.75
A1	0.05	-	0.08
A2	1.35	1.45	1.55
A3	0.60	0.65	0.70
b	0.23	-	0.31
c	0.19	-	0.25
D	8.50	8.60	8.70
E	3.80	3.90	4.00
E1	5.80	6.00	6.20
e	0.635 BSC		
h	0.30	-	0.50
L	0.40	-	0.80
$\theta$	0°	-	8°



**SOP-16 ( 150mil ) Package Dimension**


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	1.35	1.55	1.75	0.0532	0.0610	0.0688
A1	0.10	0.18	0.25	0.0040	0.0069	0.0098
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.19	0.22	0.25	0.0075	0.0087	0.0098
D	9.80	9.90	10.00	0.3859	0.3898	0.3937
E	5.80	6.00	6.20	0.2284	0.2362	0.2440
E1	3.80	3.90	4.00	0.1497	0.1536	0.1574
e	1.27 BSC			0.050 BSC		
h	0.25	0.38	0.50	0.0099	0.0148	0.0196
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
$\theta$	0°	4°	8°	0°	4°	8°
JEDEC	MS-012 (AC)					

△ \*NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.  
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED 0.15 MM ( 0.006 INCH ) PER SIDE.