十速

# TM52F5864

## *DATA SHEET Rev 1.3*

***(Please read the precautions on the second page before use)***

**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.

# PRECAUTIONS

1. Chip cannot enter Halt/Stop Mode if INTn pin is low and wakeup is enabled. (INTn=0 and EXn=1, n=0~1)
2. SFR.LVRCON (E3h) need to be set first when power on.

# AMENDMENT HISTORY

| Version | Date | Description |
|---|---|---|
| V1.0 | Feb, 2025 | New Release |
| V1.1 | Mar, 2025 | Some error correction. |
| V1.2 | Jun, 2025 | 1. Correct the reverse marking of AD23 and AD24 in QFN-32 package type.<br>2. Modify SSOP-28 package type.<br>3. Add LQFP-32、SSOP-24 package type.<br>4. Remove TSSOP-20 package type.<br>5. Modify ADC Electrical Characteristics.<br>6. Some error correction. |
| V1.3 | Mar, 2026 | 1. Add TSSOP-20 package type.<br>2. Some error correction. |

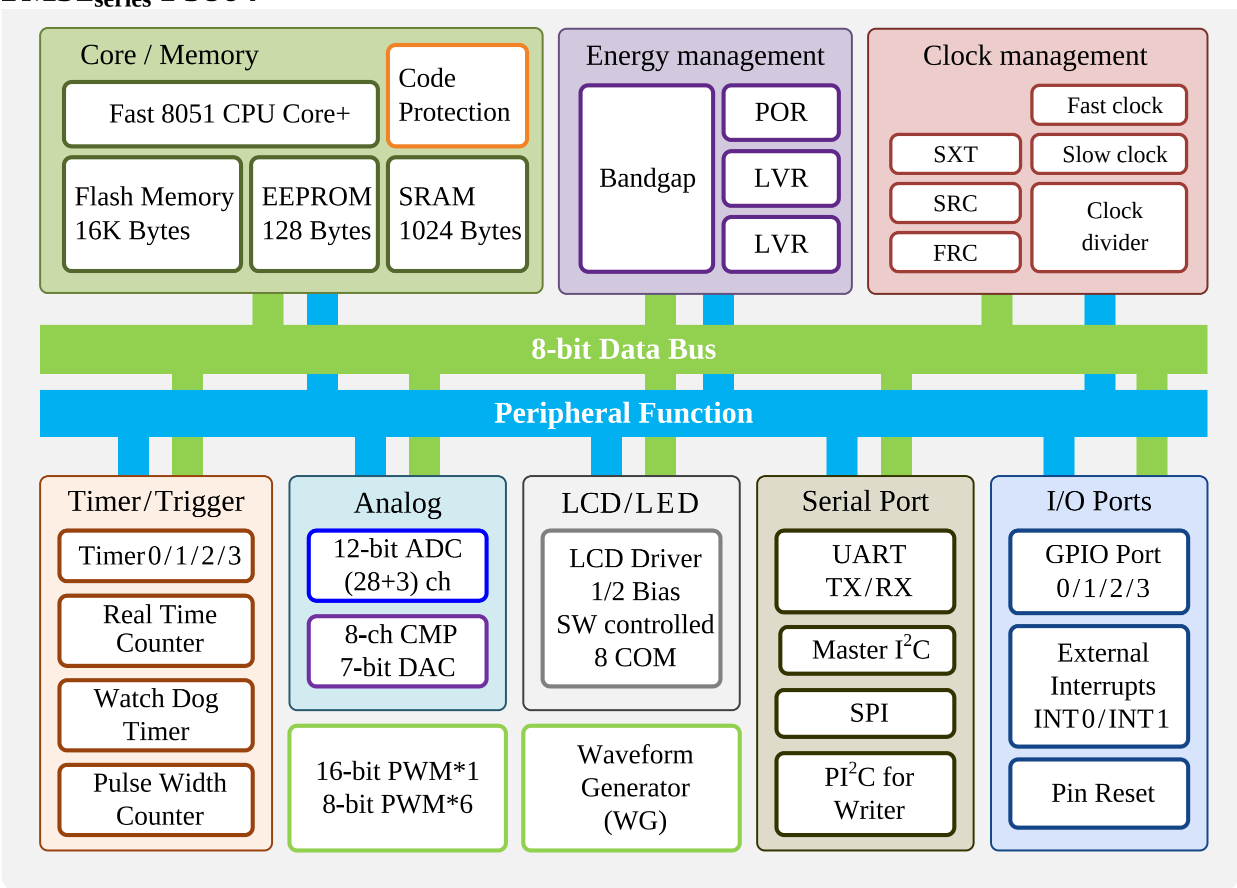
# CONTENTS

# GENERAL DESCRIPTION

**TM52** series **F5864** 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 **TM52-5864** provides improved performance, lower cost and fast time-to-market by integrating features on the chip, including 16K Bytes Flash program memory, 128 Bytes EEPROM, 1024 Bytes SRAM, Low Voltage Reset (LVR), Low Voltage Detector (LVD), dual clock power saving operation mode, 8051 standard Timer0/1/2, real time clock Timer3, 28+3 channels 12-bit A/D Convertor, 1 set 16-bit PWM0 and 6 sets 8-bit PWM1~6, LED communication waveform generator, S/W control 1/2 bias LCD COM, Comparator with 7-bit D/A converter, UART interface, I²C interface, SPI 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.

## TM52series F5864



# FEATURES

1. **Standard 8051 Instruction set, fast machine cycle**
   - Executes instructions six times faster than the standard 8051.
2. **Flash Program Memory**
   - 16K Bytes Flash program memory
   - 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
   - 10K erase times at least
   - 10 years data retention at least
3. **128 Bytes EEPROM Memory**
   - 50K erase times at least
   - 10 years data retention at least
4. **Total 1024 Bytes SRAM (IRAM + XRAM)**
   - 256 Bytes IRAM in the 8051 internal data memory area
   - 768 Bytes XRAM in the 8051 external data memory area (accessed by MOVX Instruction)
5. **Three System Clock type selections**
   - Fast clock from Internal RC (FRC，16.5888 MHz) (trim 7-bit)
   - Slow clock from 32768Hz Crystal (SXT)
   - Slow clock from Internal RC (SRC, 41 KHz)
   - System Clock can be divided by 1/2/4/16 option
6. **8051 Standard Timer – Timer0/1/2**
   - 16-bit Timer0
   - 16-bit Timer1
   - 16-bit Timer2
7. **16-bit Timer3**
   - Clock source is SXTor SRC62K clock or FRC/512
   - with reload function
   - with clear and hold function
8. **UART**
   - supports only Mode1 and Mode3
   - Additional Baud Rate generator option
   - With UART pin select option

**9. Master $I^2C$ interface**

- with $I^2C$ pin select option

**10. SPI interface**

- Master or Slave mode selectable
- Programmable transmit bit rate
- Serial clock phase and polarity options
- MSB-first or LSB-first selectable
- with SPI pin select option

**11. 1 set 16-bit PWM and 6 sets 8-bit PWM**

【16-bit PWM0 P+N】

- with period-adjustment/buffer-reload/clear and hold function
- Non-overlap durations adjustable
- Half-bridge phase control output
- FRC*2 (33.1776MHz)，FRC (16.5888MHz) or system clock source selectable

【8-bit PWM1~6】

- share period
- with period-adjustment/buffer-reload/clear and hold function
- FRC*2 (33.1776MHz)，FRC (16.5888MHz) or system clock source selectable
- with clock prescaler 1/2/4/8/16/32/64/128 option

**12. LCD drives**

- Software control LCDC0~ LCDC7 (up to 8 pins).
- 1/2 LCD Bias

**13. 12-bit ADC with 28 channels External Pin Input and 3 channels Internal Reference Voltage**

- Internal Reference Voltage: $V_{BG}$ / DAC / $V_{CC}/4$
- ADC reference voltage: $V_{CC}$ / $V_{REF}$ (1.18V/2.0V/3.0V/4.0V)

**14. Comparator (CMP)**

- 4 sets of positive and negative input selection
- with 7-bit DAC for the negative input of CMP
- DAC reference voltage: $V_{CC}/2$ or $V_{BG}$ 1.18V/2

**15. Waveform Generator (WG)**

- 1 set of 1 byte communication format waveform generator
- with WG pin select option

**16. 14 Sources, 4-level priority Interrupt**

- Timer0/Timer1/Timer2/Timer3 Interrupt
- INT0/INT1 pin Falling-Edge/Low-Level Interrupt
- Port0/1/2/3 Pin Change Interrupt
- UART TX/RX Interrupt
- ADC Interrupt
- Master $I^2C$/SPI interrupt
- LVD Interrupt
- WG Interrupt
- CMP Interrupt
- PWM0/PWM1 Interrupt

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

- P3.2/P3.3 (INT0/INT1) Interrupt & Wake-up
- Each Port0/1/2/3 pin can be defined as Interrupt & Wake-up pin (by pin change)

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

**18. Max. 30 Programmable I/O pins**

- CMOS Output
- Open-Drain Output
- Schmitt Trigger Input
- Pin Pull-up can be Enabled or Disabled

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

- 528ms/264ms/132ms/66ms selectable WDT timeout options

**20. Five types Reset**

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

**21. 16-level Low Voltage Reset (LVR)**

- 1.70V / 1.85V / 2.00V / 2.15V / 2.30V / 2.45V / 2.60V / 2.75V / 2.90V / 3.05V / 3.20V / 3.35V / 3.50V / 3.65V / 3.80V / 3.95V (step=0.15V)

**22. 16-level Low Voltage Detect (LVD)**

- 1.85V / 2.00V / 2.15V / 2.30V / 2.45V / 2.60V / 2.75V / 2.90V / 3.05V / 3.20V / 3.35V / 3.50V / 3.65V / 3.80V / 3.95V / 4.15V (step =0.15V)

## 23. Five Power Operation Modes

- Fast/Slow/Idle/Halt/Stop mode

## 24. Operating Voltage and Current

- $V_{CC}$ = 1.9V ~ 5.5V
- $I_{CC}$ = 3.7mA @Fast mode, FRC=8.2944MHz, $V_{CC}$=5V
- $I_{CC}$ = 5.4mA @Fast mode, FRC=16.5888MHz, $V_{CC}$=5V
- $I_{CC}$ = 0.6μA @Stop mode, PWRSAV=1, $V_{CC}$=5V (*POR 1.60V always on)
- $I_{CC}$ = 7.3μA @Halt mode, PWRSAV=1, $V_{CC}$=5V
- $I_{CC}$ = 11μA @Idle mode, PWRSAV=1, $V_{CC}$=5V
- Operating temperature range –40°C ~ +105°C

## 25. On-chip Debug/ICE interface

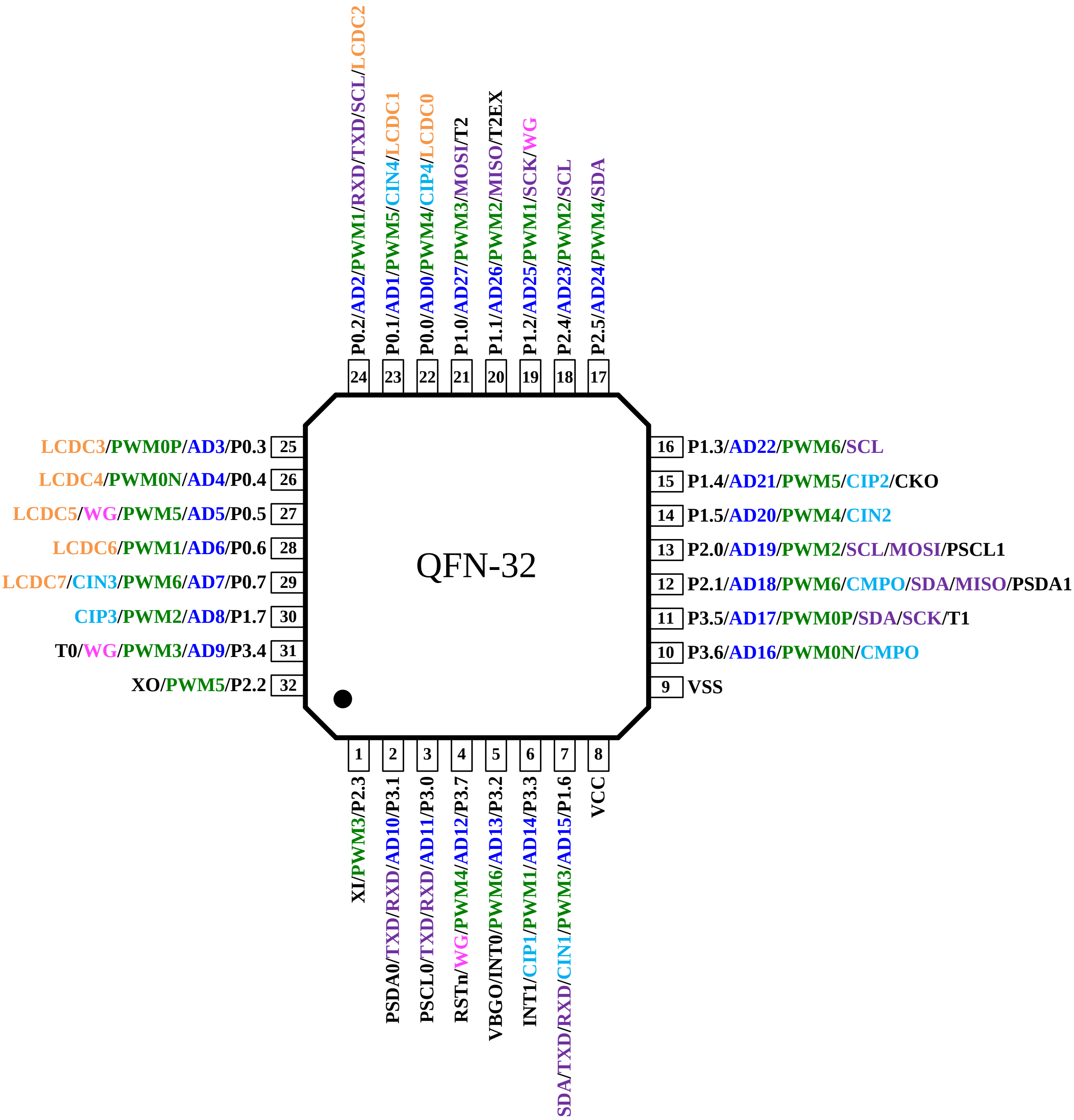
- Use P3.0/P3.1 pin or P2.0/P2.1 pin
- Share with ICP programming pin
- Mass production writer only supports P3.0/P3.1

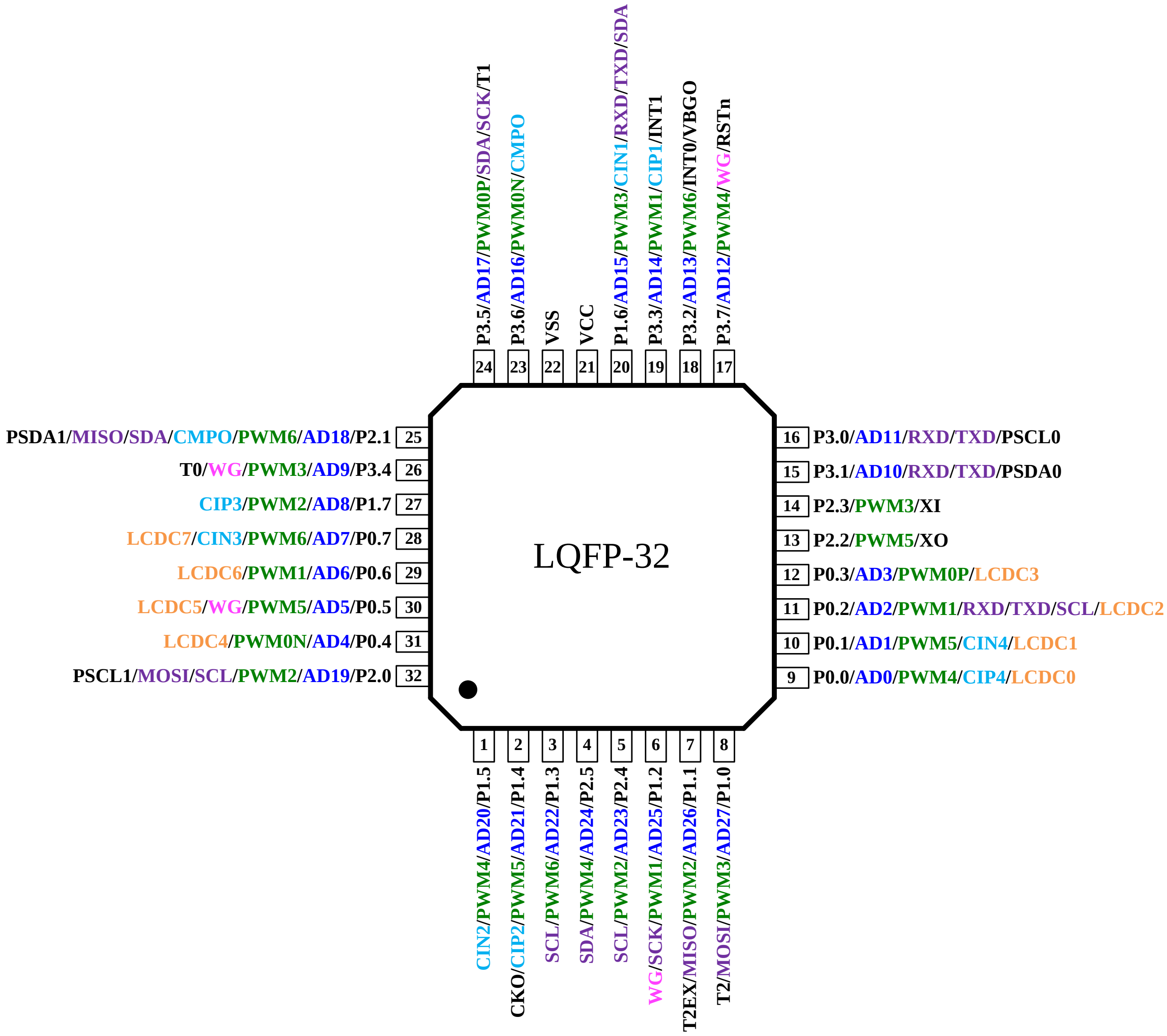
## 26. Package Types

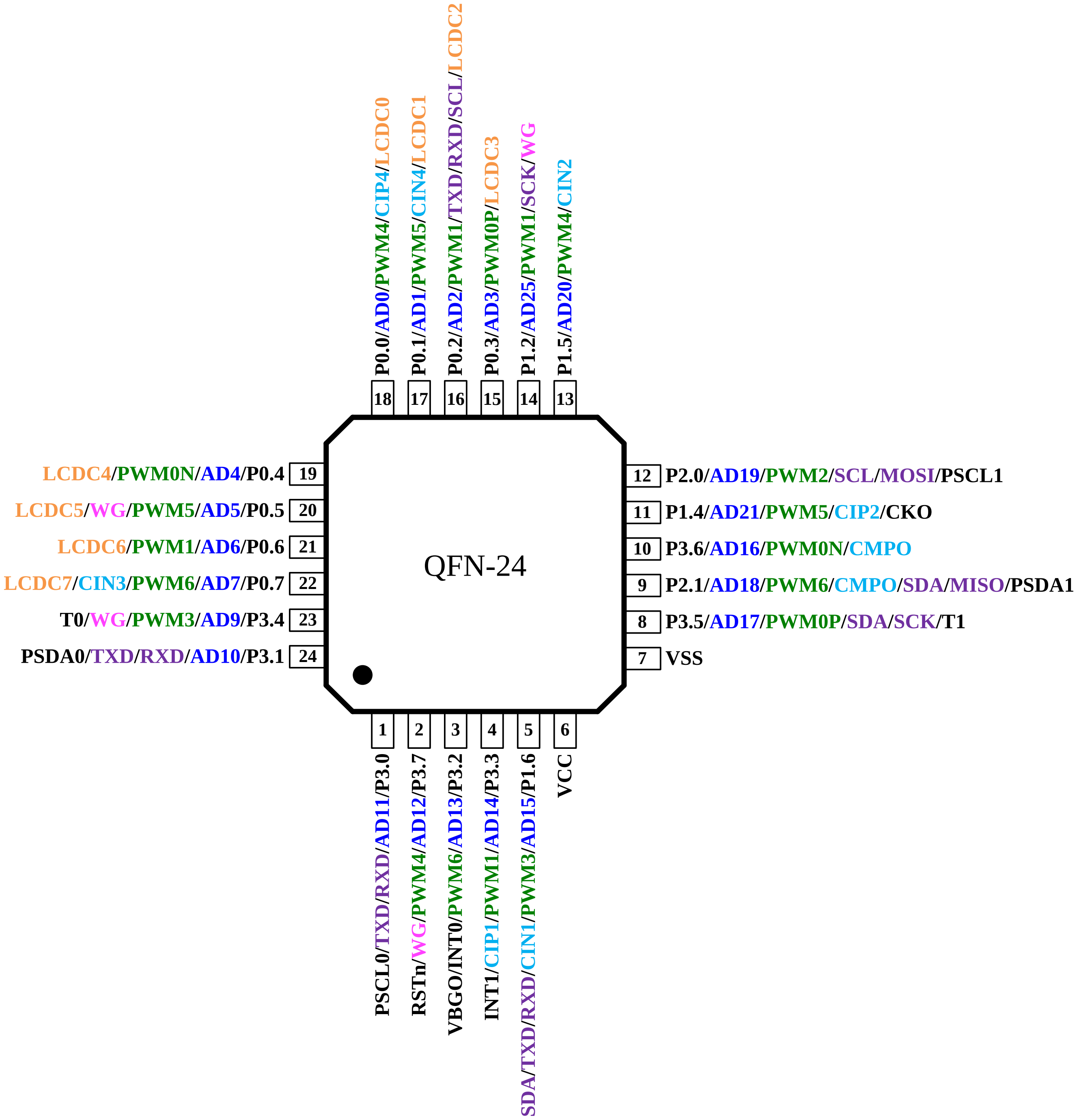
- 32-pin QFN (4x4x0.75-0.4 mm)
- 32-pin LQFP (7x7x1.4 mm)
- 24-pin QFN (3x3x0.75-0.35 mm)
- 28-pin SSOP (150mil)
- 24-pin SSOP (150 mil)
- 20-pin TSSOP（173 mil）

# PIN ASSIGNMENT

*For low-power applications, all digital I/O (including unpinned or unused) avoids setting high resistance states.




LQFP-32
1 CIN2/PWM4/AD20/P1.5
2 CKO/CIP2/PWM5/AD21/P1.4
3 SCL/PWM6/AD22/P1.3
4 SDA/PWM4/AD24/P2.5
5 SCL/PWM2/AD23/P2.4
6 WG/SCK/PWM1/AD25/P1.2
7 T2EX/MISO/PWM2/AD26/P1.1
8 T2/MOSI/PWM3/AD27/P1.0
9 P0.0/AD0/PWM4/CIP4/LCDC0
10 P0.1/AD1/PWM5/CIN4/LCDC1
11 P0.2/AD2/PWM1/RXD/TXD/SCL/LCDC2
12 P0.3/AD3/PWM0P/LCDC3
13 P2.2/PWM5/XO
14 P2.3/PWM3/XI
15 P3.1/AD10/RXD/TXD/PSDA0
16 P3.0/AD11/RXD/TXD/PSCL0
17 P3.7/AD12/PWM4/WG/RSTn
18 P3.2/AD13/PWM6/INT0/VBGO
19 P3.3/AD14/PWM1/CIP1/INT1
20 P1.6/AD15/PWM3/CIN1/RXD/TXD/SDA
21 VCC
22 VSS
23 P3.6/AD16/PWM0N/CMPO
24 P3.5/AD17/PWM0P/SDA/SCK/T1
25 PSDA1/MISO/SDA/CMPO/PWM6/AD18/P2.1
26 T0/WG/PWM3/AD9/P3.4
27 CIP3/PWM2/AD8/P1.7
28 LCDC7/CIN3/PWM6/AD7/P0.7
29 LCDC6/PWM1/AD6/P0.6
30 LCDC5/WG/PWM5/AD5/P0.5
31 LCDC4/PWM0N/AD4/P0.4
32 PSCL1/MOSI/SCL/PWM2/AD19/P2.0


P0.0/AD0/PWM4/CIP4/LCDC0
P0.1/AD1/PWM5/CIN4/LCDC1
P0.2/AD2/PWM1/TXD/RXD/SCL/LCDC2
P0.3/AD3/PWM0P/LCDC3
P1.2/AD25/PWM1/SCK/WG
P1.5/AD20/PWM4/CIN2
18
17
16
15
14
13
LCDC4/PWM0N/AD4/P0.4
LCDC5/WG/PWM5/AD5/P0.5
LCDC6/PWM1/AD6/P0.6
LCDC7/CIN3/PWM6/AD7/P0.7
T0/WG/PWM3/AD9/P3.4
PSDA0/TXD/RXD/AD10/P3.1
19
20
21
22
23
24
QFN-24
12
11
10
9
8
7
P2.0/AD19/PWM2/SCL/MOSI/PSCL1
P1.4/AD21/PWM5/CIP2/CKO
P3.6/AD16/PWM0N/CMPO
P2.1/AD18/PWM6/CMPO/SDA/MISO/PSDA1
P3.5/AD17/PWM0P/SDA/SCK/T1
VSS
1
2
3
4
5
6
PSCL0/TXD/RXD/AD11/P3.0
RSTn/WG/PWM4/AD12/P3.7
VBGO/INT0/PWM6/AD13/P3.2
INT1/CIP1/PWM1/AD14/P3.3
SDA/TXD/RXD/CIN1/PWM3/AD15/P1.6
VCC

**TM52F5864**

**SSOP-28**

| | | | |
|---|---|---|---|
| VCC | 1 | 28 | P1.6/AD15/PWM3/CIN1/RXD/TXD/SDA |
| VSS | 2 | 27 | P3.3/AD14/PWM1/CIP1/INT1 |
| CMPO/PWM0N/AD16/P3.6 | 3 | 26 | P3.2/AD13/PWM6/INT0/VBGO |
| T1/SCK/SDA/PWM0P/AD17/P3.5 | 4 | 25 | P3.7/AD12/PWM4/WG/RSTn |
| PSDA1/MISO/SDA/CMPO/PWM6/AD18/P2.1 | 5 | 24 | P3.0/AD11/RXD/TXD/PSCL0 |
| T0/WG/PWM3/AD9/P3.4 | 6 | 23 | P3.1/AD10/RXD/TXD/PSDA0 |
| WG/SCK/PWM1/AD25/P1.2 | 7 | 22 | P2.3/PWM3/XI |
| PSCL1/MOSI/SCL/PWM2/AD19/P2.0 | 8 | 21 | P2.2/PWM5/XO |
| SCL/PWM6/AD22/P1.3 | 9 | 20 | P1.7/AD8/PWM2/CIP3 |
| SDA/PWM4/AD24/P2.5 | 10 | 19 | P0.7/AD7/PWM6/CIN3/LCDC7 |
| SCL/PWM2/AD23/P2.4 | 11 | 18 | P0.6/AD6/PWM1/LCDC6 |
| T2EX/MISO/PWM2/AD26/P1.1 | 12 | 17 | P0.4/AD4/PWM0N/LCDC4 |
| T2/MOSI/PWM3/AD27/P1.0 | 13 | 16 | P0.3/AD3/PWM0P/LCDC3 |
| LCDC0/CIP4/PWM4/AD0/P0.0 | 14 | 15 | P0.1/AD1/PWM5/CIN4/LCDC1 |

**TM52F5864**

**SSOP-24**

| | | | |
|---|---|---|---|
| LCDC7/CIN3/PWM6/AD7/P0.7 | 1 | 24 | P0.6/AD6/PWM1/LCDC6 |
| T0/WG/PWM3/AD9/P3.4 | 2 | 23 | P0.4/AD4/PWM0N/LCDC4 |
| PSDA0/TXD/RXD/AD10/P3.1 | 3 | 22 | P0.3/AD3/PWM0P/LCDC3 |
| PSCL0/TXD/RXD/AD11/P3.0 | 4 | 21 | P0.1/AD1/PWM5/CIN4/LCDC1 |
| RSTn/WG/PWM4/AD12/P3.7 | 5 | 20 | P0.0/AD0/PWM4/CIP4/LCDC0 |
| VBGO/INT0/PWM6/AD13/P3.2 | 6 | 19 | P1.0/AD27/PWM3/MOSI/T2 |
| INT1/CIP1/PWM1/AD14/P3.3 | 7 | 18 | P1.1/AD26/PWM2/MISO/T2EX |
| SDA/TXD/RXD/CIN1/PWM3/AD15/P1.6 | 8 | 17 | P1.2/AD25/PWM1/SCK/WG |
| VSS | 9 | 16 | P2.0/AD19/PWM2/SCL/MOSI/PSCL1 |
| VCC | 10 | 15 | P2.2/PWM5/XO |
| CMPO/PWM0N/AD16/P3.6 | 11 | 14 | P2.3/PWM3/XI |
| T1/SCK/SDA/PWM0P/AD17/P3.5 | 12 | 13 | P2.1/AD18/PWM6/CMPO/SDA/MISO/PSDA1 |

**TM52F5864**

**TSSOP-20**

| | | | |
|---|---|---|---|
| T0/WG/PWM3/AD9/P3.4 | 1 | 20 | P1.7/AD8/PWM2/CIP3 |
| PSDA0/TXD/RXD/AD10/P3.1 | 2 | 19 | P0.7/AD7/PWM6/CIN3/LCDC7 |
| PSCL0/TXD/RXD/AD11/P3.0 | 3 | 18 | P0.4/AD4/PWM0N/LCDC4 |
| RSTn/WG/PWM4/AD12/P3.7 | 4 | 17 | P0.3/AD3/PWM0P/LCDC3 |
| VBGO/INT0/PWM6/AD13/P3.2 | 5 | 16 | P0.2/AD2/PWM1/RXD/TXD/SCL/LCDC2 |
| INT1/CIP1/PWM1/AD14/P3.3 | 6 | 15 | P0.1/AD1/PWM5/CIN4/LCDC1 |
| VSS | 7 | 14 | P0.0/AD0/PWM4/CIP4/LCDC0 |
| SDA/TXD/RXD/CIN1/PWM3/AD15/P1.6 | 8 | 13 | P1.2/AD25/PWM1/WG |
| VCC | 9 | 12 | P2.0/AD19/PWM2/SCL/MOSI/PSCL1 |
| T1/SCK/SDA/PWM0P/AD17/P3.5 | 10 | 11 | P2.1/AD18/PWM6/CMPO/SDA/MISO/PSDA1 |

# PIN DESCRIPTION

| Name | In/Out | Pin Description |
|---|---|---|
| P3.0~P3.7<br>P2.0~P2.5<br>P1.0~P1.7<br>P0.0~P0.7 | I/O | Bit-programmable I/O port for Schmitt-trigger input, CMOS push-pull output or "open-drain" output. Pull-up resistors are assignable by software. These pin's level change can interrupt/wake up CPU from Idle/Halt/Stop mode. |
| INT0，INT1 | I | External low level or falling edge Interrupt input, Idle/Halt/Stop mode wake up input. |
| T0，T1，T2 | I | Timer0, Timer1, Timer2 event count pin input |
| T2EX | I | Timer2 external trigger input |
| CKO | O | System Clock divided by 2 output |
| VBGO | O | Bandgap voltage output |
| PWM0N, PWM0P | O | 16-bit PWM0 dead-zone complementary output |
| PWM1~PWM6 | O | 8-bit PWM1~6 output shares period with PWM1 |
| AD0~AD27 | O | ADC input |
| LCDC0~LCDC7 | O | LCD 1/2 bias output |
| CIN1~4 | I | Comparator negative input |
| CIP1~4 | I | Comparator positive input |
| CMPO | O | Comparator output |
| WG | O | RGB LED waveform generator output |
| RXD | I | UART Mode1/3 receive data |
| TXD | O | UART Mode1/3 transmit data |
| SCK | I/O | SPI clock output for master or clock input for slave mode |
| MISO | I/O | SPI data input for master mode, data output for slave mode |
| MOSI | I/O | SPI data output for master mode, data input for slave mode |
| SCL | I/O | Master $I^2C$ SCL |
| SDA | I/O | Master $I^2C$ SDA |
| PSCL0，PSCL1 | I/O | $I^2C$ SCL for for debug mode and memory programming |
| PSDA0，PSDA1 | I/O | $I^2C$ SDA for for debug mode and memory programming |
| RSTn | I | External active low reset input with pull-up resistor |
| XI，XO | I | Crystal/Resonator oscillator input for System clock |
| VCC，VSS | P | Power input pin and ground |

# PIN SUMMERY

| Pin number | | | | | | Pin Name | Type | Initial State | Input | | | Output | | Alternative Function | | | | | | | | | MISC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| QFN-32 | LQFP-32 | QFN-24 | SSOP-28 | SSOP-24 | TSSOP-20 | | | | Pull-up Control | Wake up | Ext. Interrupt | CMOS Push-Pull | Open Drain | ADC | PWM | LCD | CMP | WG | UART | Master $I^2C$ | SPI | Timer | |
| 1 | 14 | – | 22 | 14 | - | **XI/PWM3/P2.3** | I/O | Hi-Z | ● | ● | ● | ● | ● | | ● | | | | | | | | Crystal |
| 2 | 15 | 24 | 23 | 3 | 2 | **PSDA0/TXD/RXD/AD10/P3.1** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | | | | | ● | | | | PSDA |
| 3 | 16 | 1 | 24 | 4 | 3 | **PSCL0/TXD/RXD/AD11/P3.0** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | | | | | ● | | | | PSCL |
| 4 | 17 | 2 | 25 | 5 | 4 | **RSTn/WG/PWM4/AD12/P3.7** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | ● | | | | | Reset |
| 5 | 18 | 3 | 26 | 6 | 5 | **VBGO/INT0/PWM6/AD13/P3.2** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | | | | | | VBGO |
| 6 | 19 | 4 | 27 | 7 | 6 | **INT1/CIP1/PWM1/AD14/P3.3** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | ● | | | | | | |
| 7 | 20 | 5 | 28 | 8 | 8 | **SDA/TXD/RXD/CIN1/PWM3/AD15/P1.6** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | ● | | ● | ● | | | |
| 8 | 21 | 6 | 1 | 10 | 9 | **VCC** | P | | | | | | | | | | | | | | | | |
| 9 | 22 | 7 | 2 | 9 | 7 | **VSS** | P | | | | | | | | | | | | | | | | |
| 10 | 23 | 10 | 3 | 11 | - | **CMPO/PWM0N/AD16/P3.6** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | ● | | | | | | |
| 11 | 24 | 8 | 4 | 12 | 10 | **T1/SCK/SDA/PWM0P/AD17/P3.5** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | | | ● | ● | ● | |
| 12 | 25 | 9 | 5 | 13 | 11 | **PSDA1/MISO/SDA/CMPO/PWM6/AD18/P2.1** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | ● | | | ● | ● | | PSDA |
| 13 | 32 | 12 | 8 | 16 | 12 | **PSCL1/MOSI/SCL/PWM2/AD19/P2.0** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | | | ● | ● | | PSCL |
| 14 | 1 | 13 | – | – | - | **CIN2/PWM4/AD20/P1.5** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | ● | | | | | | |
| 15 | 2 | – | – | – | - | **CKO/CIP2/PWM5/AD21/P1.4** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | ● | | | | | | CKO |
| 16 | 3 | – | 9 | – | - | **SCL/PWM6/AD22/P1.3** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | | | ● | | | |
| 17 | 4 | – | 10 | – | - | **SDA/PWM4/AD24/P2.5** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | | | ● | | | |
| 18 | 5 | – | 11 | – | - | **SCL/PWM2/AD23/P2.4** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | | | ● | | | |
| 19 | 6 | 14 | 7 | 17 | 13 | **WG/SCK/PWM1/AD25/P1.2** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | ● | | | ● | | |
| 20 | 7 | – | 12 | 18 | - | **T2EX/MISO/PWM2/AD26/P1.1** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | | | | ● | ● | |
| 21 | 8 | – | 13 | 19 | - | **T2/MOSI/PWM3/AD27/P1.0** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | | | | ● | ● | |
| 22 | 9 | 18 | 14 | 20 | 14 | **LCDC0/CIP4/PWM4/AD0/P0.0** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | |
| 23 | 10 | 17 | 15 | 21 | 15 | **LCDC1/CIN4/PWM5/AD1/P0.1** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | |
| 24 | 11 | 16 | – | – | 16 | **LCDC2/SCL/TXD/RXD/PWM1/AD2/P0.2** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | ● | | | ● | ● | | | |
| 25 | 12 | 15 | 16 | 22 | 17 | **LCDC3/PWM0P/AD3/P0.3** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | |
| 26 | 31 | 19 | 17 | 23 | 18 | **LCDC4/PWM0N/AD4/P0.4** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | |
| 27 | 30 | 20 | – | – | - | **LCDC5/WG/PWM5/AD5/P0.5** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | ● | | ● | | | | | |
| 28 | 29 | 21 | 18 | 24 | - | **LCDC6/PWM1/AD6/P0.6** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | | |
| 29 | 28 | 22 | 19 | 1 | 19 | **LCDC7/CIN3/PWM6/AD7/P0.7** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | ● | ● | | | | | | |
| 30 | 27 | – | 20 | – | 20 | **CIP3/PWM2/AD8/P1.7** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | ● | | | | | | |
| 31 | 26 | 23 | 6 | 2 | 1 | **T0/WG/PWM3/AD9/P3.4** | I/O | Hi-Z | ● | ● | ● | ● | ● | ● | ● | | | ● | | | | ● | |
| 32 | 13 | – | 21 | 15 | - | **XO/PWM5/P2.2** | I/O | Hi-Z | ● | ● | ● | ● | ● | | ● | | | | | | | | Crystal |

# 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 |
|---|---|---|---|---|---|---|---|---|
| **ACC** | 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.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 |
|---|---|---|---|---|---|---|---|---|
| **SP** | 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 |
|---|---|---|---|---|---|---|---|---|
| **DPL** | 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 |
|---|---|---|---|---|---|---|---|---|
| **DPH** | 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 |
|---|---|---|---|---|---|---|---|---|
| **AUX1** | CLRWDT | CLRTM3 | – | ADSOC | CLRPWM0 | CLRPWM1 | WGEN | DPSEL |
| R/W | R/W | R/W | – | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | – | 0 | 1 | 1 | 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 C | Flag OV | Flag AC |
|---|---|---|---|
| ADD | X | X | X |
| ADDC | X | X | X |
| SUBB | X | X | X |
| MUL | 0 | X | |
| DIV | 0 | X | |
| DA | X | | |
| RRC | X | | |
| RLC | X | | |
| SETB C | 1 | | |

| Instruction | Flag C | Flag OV | Flag AC |
|---|---|---|---|
| CLR C | 0 | | |
| CPL C | X | | |
| ANL C, bit | X | | |
| ANL C, /bit | X | | |
| ORL C, bit | X | | |
| ORL C, /bit | X | | |
| MOV C, bit | X | | |
| CJNE | X | | |
| | | | |

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 |
|---|---|---|---|---|---|---|---|---|
| **PSW** | 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)

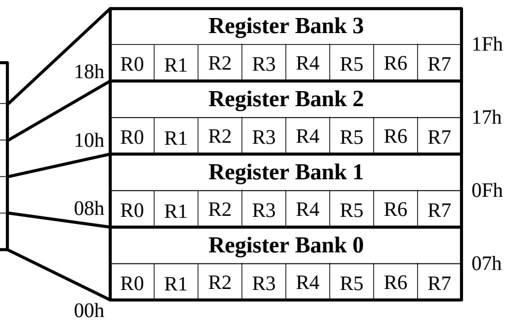
D0h.2 **OV:** ALU overflow flag

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.

| PSW | | | | | | | |
|---|---|---|---|---|---|---|---|
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **CY** | **AC** | **FO** | **RS1** | **RS0** | **OV** | **F1** | **P** |

| RS1 | RS0 | Bank |
|---|---|---|
| 1 | 1 | 3 |
| 1 | 0 | 2 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |



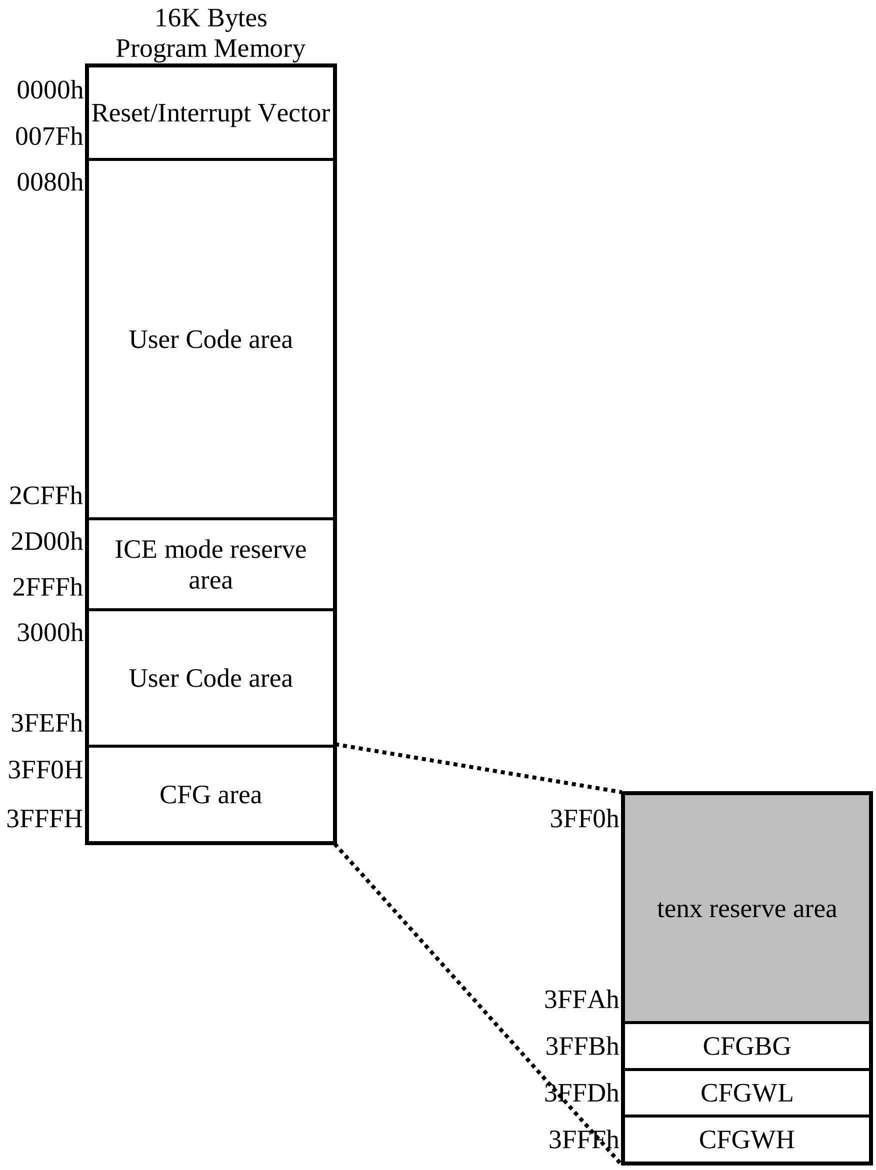
# 2. Memory

## 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 10K 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. In the in-circuit emulation (ICE) mode, user also needs to reserve the address space 2D00h~2FFFh for ICE System communication.




**Program Memory Partition**

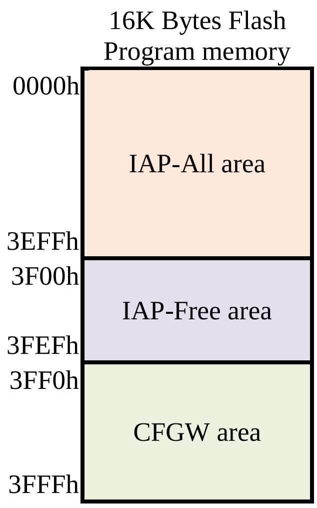
### 2.1.2 Flash ICP Mode

The Flash memory can be programmed by the tenx proprietary writer, 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

This chip 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 chip does not need to erase one Flash page before write. The available IAP data space is 240 Bytes after chip reset, and can be re-defined by the "IAPALL" control register as shown below.




| Flash memory | IAPALL | MOVC Accessible | MOVX (IAP) Accessible |
|---|---|---|---|
| 0000h~3EFFh | 0 | Yes | No |
| | 1 | Yes | Yes |
| 3F00h~3FEFh | X | Yes | Yes |
| 3FF0h~3FF7h | X | Yes | Yes |
| 3FF8h~3FFEh | 0 | Yes | No |
| | 1 | Yes | Yes |
| 3FFFh | X | Yes | No |

In IAP mode, the program Flash memory is separated into three sectors: IAP-All area, IAP-Free area, and CFGW area. These three sectors are regulated differently.

The **IAP-All area** is protected by the IAPALL register to prevent IAP mode from writing application data to the program area, resulting in a program code error that cannot be repaired. The size of this area is 16128 Bytes. Enabling IAPALL requires writing 65h to SFR SWCMD 97h to set the IAPALL control flag. Then, software can use MOVX instructions to write application data to flash memory from 0000h to 3EFFh. If user wants to disable IAPALL function, user can write other values to SFR SWCMD 97h to clear the IAPALL control flag. User must be careful not to overwrite program code which is already resided on the same Flash memory area.

The **IAP-Free area** has no control bit to protect. It can be used to reliably store system application data that needs to be programmed once or periodically during system operation. Other areas of Flash memory can be used to store data, but this area is usually better. The size of this area is 240 Bytes, equivalent to an EEPROM, and Flash memory can provide byte access to read and write commands. The chip has a physical 128 byte 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 chip accepts IAP write command only when IAPWE=1. Flash IAP writing one byte requires approximately 1 ms @$V_{CC}$=5.0V. Meanwhile, the CPU stays in a waiting state, but all peripheral modules (Timers, LED, 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 IAP Time-out function for escaping write fail state. Flash IAP writing needs setting the system clock to FRC/2 (or slower) and $V_{CC}$ >3.0V.

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 3.0V < VCC < 5.5V
ANL     AUX2, #3Fh         ; Disable WDT
MOV     DPTR, #3F00h       ; DPTR=3F00h=target IAP address
MOV     A, #5Ah            ; A=5Ah=target IAP write data
MOV     IAPWE, #47h        ; IAP write enable
ORL     AUX2, #02h         ; IAP Time-Out function enable
MOVX    @DPTR, A           ; Flash[3F00h] =5Ah, after IAP write
                           ; 1ms~2ms H/W writing time, CPU wait
MOV     IAPWE, #00h        ; IAP write disable, immediately after IAP write
CLR     A                  ; A=0
MOVX    A, @DPTR           ; A=5Ah
CLR     A                  ; A=0
MOVC    A, @A+DPTR         ; A=5Ah
```

```
 ; IAP example code (C)
 ; need 3.0V < VCC < 5.5V
unsigned char xdata PROM[4096] _at_ 0x2000    // 0x2000 = start address
unsigned char code CODE[4096] _at_ 0x2000     // 0x2000 = start address

IAPALL = 0x65;
IAPWE = 0x47;
PROM[0x02] = wdata;     // write data into ROM[0x2002]
IAPWE = 0x00;
IAPALL = 0x00;

rdata = CODE[0x105];    // read data from ROM[0x2105]
```

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

97h.7~0 **IAPALL (W):**
Write 65h to set IAPALL flag. Write other value to clear IAPALL flag.

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

97h.0 **IAPALL (R):** Flag indicates Flash can be written by IAP or not
0: Flash IAP disable
1: Flash IAP enable

| SFR C9h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IAPCON** | IAPCON | | | | | | | |
| R/W | W | | | | | | | |
| Reset | – | – | – | – | – | – | – | – |

C9h.7~0 **IAPCON (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.

| SFR C9h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IAPCON** | IAPWE | IAPTO | EEPWE | INFOWE | – | – | – | – |
| R/W | R | R | R | R | – | – | – | – |
| Reset | 0 | 0 | 0 | 0 | – | – | – | – |

C9h.7 **IAPWE (R):** Flag indicates Flash memory can be written by IAP or not
0: IAP Write disable
1: IAP Write enable

C9h.6 **IAPTO (R):** Time-Out flag of IAP write/EEPROM write/INFO write. Set by H/W when IAP or EEPROM write or INFO write Time-out occurs. Cleared this flag by H/W when IAPWE=0 or EEPWE=0 or INFOWE=0.

| SFR F7h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX2** | WDTE | | PWRSAV | VBGOUT | – | IAPTE | | – |
| R/W | R/W | R/W | R/W | R/W | – | R/W | | – |
| Reset | 0 | 0 | 0 | 0 | – | 1 | 1 | – |

F7h.2~1 **IAPTE:**IAP write / EEPROM write / INFO write watchdog timer enable
00: Disable
01: wait 2ms trigger watchdog time-out flag, and escape the write fail state
10: wait 4ms trigger watchdog time-out flag, and escape the write fail state
11: wait 16ms 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 EEPROM Memory

This chip contains 128 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 50K write/erase cycles.

**EEPROM Memory**

| Address | EEPROM Memory |
|---|---|
| **EE00h** | EEPROM[0] |
| **EE02h** | EEPROM[1] |
| **EE04h** | . . . |
| **EEFCh** | EEPROM[126] |
| **EEFEh** | EEPROM[127] |

**(Only even addresses can be used, odd addresses are invalid)**

**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, ADDR=ADDR+2), and the ACC contains the data being written. EEPROM writing requires approximately 2 ms @$V_{CC}$=3.0V, 1 ms @$V_{CC}$=5.0V. Meanwhile, the CPU stays in a waiting state, but all peripheral modules (Timers, LED, and others) continue running during the writing time. The software must handle the pending interrupts after an EEPROM write. The chip has a build-in EEPROM Time-out function shared with Flash IAP for escaping write fail state. EEPROM writing needs $V_{CC}$>2.7V.

**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
 ; neeed 2.7V < VCC < 5.5V
ANL     AUX2, #3Fh         ; Disable WDT
MOV     DPTR, #0EE00h      ; DPTR=EE00h=target EEPROM[0] address
MOV     A, #0A5h           ; A=A5h=target EEPROM[0] write data
MOV     EEPWE, #0E2h       ; EEPROM write enable
ORL     AUX2, #04h         ; EEPROM Time-Out function select
MOVX    @DPTR, A           ; EEPROM[0]=A5h, after EEPROM write
                           ; 1ms~2ms H/W writing time, CPU wait
MOV     EEPWE, #000h       ; EEPROM write disable, immediately after EEPROM write
CLR     A                  ; A=0
MOVX    A, @DPTR           ; A=A5h
```

## 2.3 Precautions for using EEPROM

### 2.3.1 About the writing characteristics of EEPROM

(1) The writing time of EEPROM is not fixed. It takes different time to write different data.

(2) The writing time is affected by voltage, temperature, and data conversion conditions. Higher voltage makes the writing time shorter. When the temperature is high or there are more data 0, the writing time is longer.

(3) The CPU is in a waiting state during the EEPROM writing process, but all peripheral modules (timers, etc.) continue to run. The software must handle the interrupt generated during the process after the EEPROM data is written.

(4) This chip has a built-in timeout watchdog timer to protect the write timeout, ensuring that the system can execute the program normally.

### 2.3.2 About the write time of EEPROM

The write time of EEPROM is related to voltage, temperature, and the number of writes.

At least 50,000 erase cycles ($F_{SYS}$=FRC/2, 2.7V<$V_{CC}$<5.5V, -20℃~105℃)

### 2.3.3 Write verification

Depending on the specific application, it is generally required to read back the value written to the program EEPROM for comparison and verification.

### 2.3.4 Protection against erroneous writes

When starting the write operation, the following operations can prevent erroneous writes:

(1) Under-voltage detection. When writing EEPROM, the voltage must be >2.7V. The LVD function can be used to monitor the voltage.

(LVD monitoring voltage is recommended to be greater than 3.2V to prevent power failure and leave enough time for writing EEPROM)

(2) Clear the watchdog (WDT) every time a byte is written. Prevent the watchdog from resetting when multiple bytes are written continuously.

(3) When writing data, it is necessary to temporarily disable the interrupt and enable the interrupt after the writing is completed.

(4) In case of software failure, add an EEPROM read-back mechanism to the program to ensure that the data is written correctly.

(5) Timeout protection: Enable the write timeout watchdog (IAPTE) in the program to prevent the system from freezing due to write timeout.

(6) Power glitch: Connecting capacitors in parallel to the VCC and VSS pins according to the waveform can stabilize the system power supply.

| SFR **C9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IAPCON** | IAPCON | | | | | | | |
| R/W | W | | | | | | | |
| Reset | – | – | – | – | – | – | – | – |

C9h.7~0 **IAPCON (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.

| SFR **C9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IAPCON** | IAPWE | IAPTO | EEPWE | INFOWE | – | – | – | – |
| R/W | R | R | R | R | – | – | – | – |
| Reset | 0 | 0 | 0 | 0 | – | – | – | – |

C9h.6 **IAPTO (R):** Time-Out flag of IAP write/EEPROM write/INFO write. Set by H/W when IAP or EEPROM write or INFO write Time-out occurs. Cleared this flag by H/W when IAPWE=0 or EEPWE=0 or INFOWE=0.

C9h.5 **EEPWE (R):** Flag indicates EEPROM memory can be written or not
0: EEPROM Write disable
1: EEPROM Write enable

| SFR **F7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX2** | WDTE | | PWRSAV | VBGOUT | – | IAPTE | | – |
| R/W | R/W | R/W | R/W | R/W | – | R/W | | – |
| Reset | 0 | 0 | 0 | 0 | – | 1 | 1 | – |

F7h.2~1 **IAPTE:**IAP write / EEPROM write / INFO write watchdog timer enable
00: Disable
01: wait 2ms trigger watchdog time-out flag, and escape the write fail state
10: wait 4ms trigger watchdog time-out flag, and escape the write fail state
11: wait 16ms trigger watchdog time-out flag, and escape the write fail state

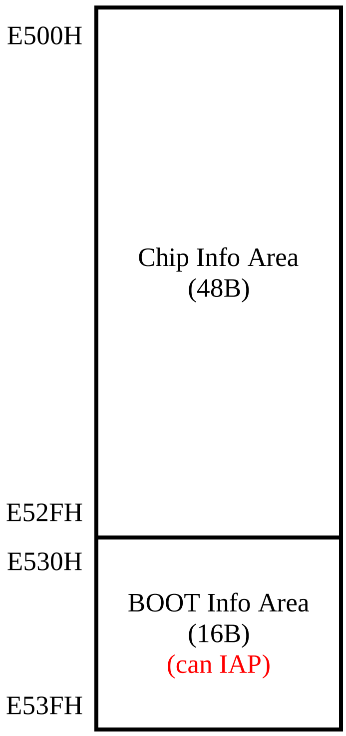
## 2.4 Information Memory

The Chip has a 64 bytes Information memory. The Information memory address continuous space (E500h~E53Fh) is partitioned to several sectors for device operation.

Chip Info area is tenx reserved defined as production information, such as ID, Special Regulations, Code Num, checksum. BOOT Info area allow IAP write, user can store new checksum code in this area after Flash IAP.

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

1. Only BOOT Info Area can be written by IAP.
2. Set INFOWE=1.




**Info ROM partition**

**Info ROM IAP Write** is simply achieved by a "MOVX @DPTR, A" instruction while the DPTR contains the target Flash address, and the ACC contains the data being written. Flash writing requires approximately 0.6 ms @$V_{CC}$=3.0V~5.5V, VCC capacitance greater than 220uF. During the period of IAP, the CPU stays in a waiting state, but all peripheral modules 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 (F7h.2~1) to escape write fail state.

**Info ROM IAP Read** only can be performed by the "MOVX" instruction as long as the target address points to the E500h~E53Fh area. An Info ROM IAP read does not require extra CPU wait time.

Info ROM IAP Example:

```
; need 3.0V < VCC < 5.5V
ANL     AUX2, #3Fh        ; Disable WDT
MOV     DPTR, #E530h      ; DPTR=E530h=target IAP address
MOV     A, #5Ah           ; A=5Ah=target IAP write data
ORL     AUX2,  #04h       ; IAP Time-Out function select
MOV     IAPCON, #A1h      ; Info ROM IAP write enable.
MOVX    @DPTR, A          ; IAP Write Info ROM
                          ; Info ROM[E530h]=5Ah after IAP write
MOV     IAPCON, #00h      ; IAP write disable, immediately after IAP write
ANL     PWRCON, #7Fh      ; IVC Enable
MOVX    A, @DPTR          ; Read Info ROM.  A=5Ah
```

| SFR **C9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IAPCON** | IAPCON | | | | | | | |
| R/W | W | | | | | | | |
| Reset | – | – | – | – | – | – | – | – |

C9h.7~0 **IAPCON (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.

| SFR **C9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IAPCON** | IAPWE | IAPTO | EEPWE | INFOWE | – | – | – | – |
| R/W | R | R | R | R | – | – | – | – |
| Reset | 0 | 0 | 0 | 0 | – | – | – | – |

C9h.6 **IAPTO (R):** Time-Out flag of IAP write/EEPROM write/INFO write. Set by H/W when IAP or EEPROM write or INFO write Time-out occurs. Cleared this flag by H/W when IAPWE=0 or EEPWE=0 or INFOWE=0.

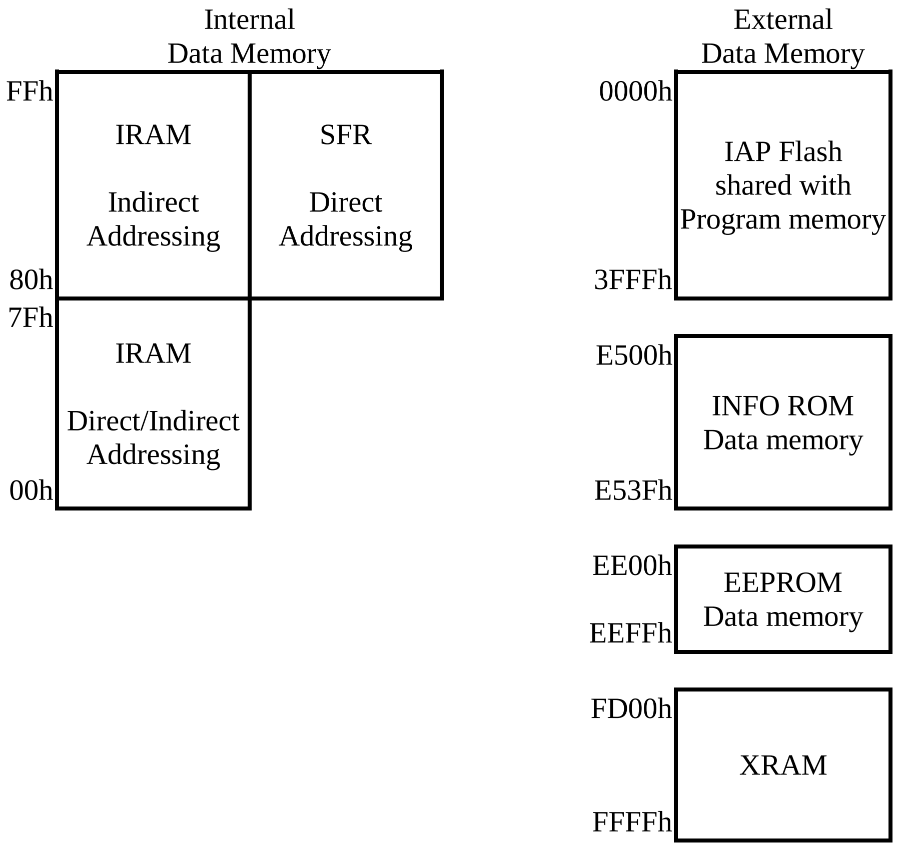
C9h.4 **INFOWE (R):** Flag indicates INFO memory can be written by IAP or not
0: INFO IAP Write disable
1: INFO IAP Write enable

| SFR **F7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX2** | WDTE | | PWRSAV | VBGOUT | – | IAPTE | | – |
| R/W | R/W | R/W | R/W | R/W | – | R/W | | – |
| Reset | 0 | 0 | 0 | 0 | – | 1 | 1 | – |

F7h.2~1 **IAPTE:**IAP write / EEPROM write / INFO write watchdog timer enable
00: Disable
01: wait 2ms trigger watchdog time-out flag, and escape the write fail state
10: wait 4ms trigger watchdog time-out flag, and escape the write fail state
11: wait 16ms trigger watchdog time-out flag, and escape the write fail state

## 2.5 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 768 Bytes XRAM, 128 Bytes EEPROM, 64 Bytes INFO ROM and IAP Flash, which can be only accessed by MOVX instruction.



### 2.5.1 IRAM

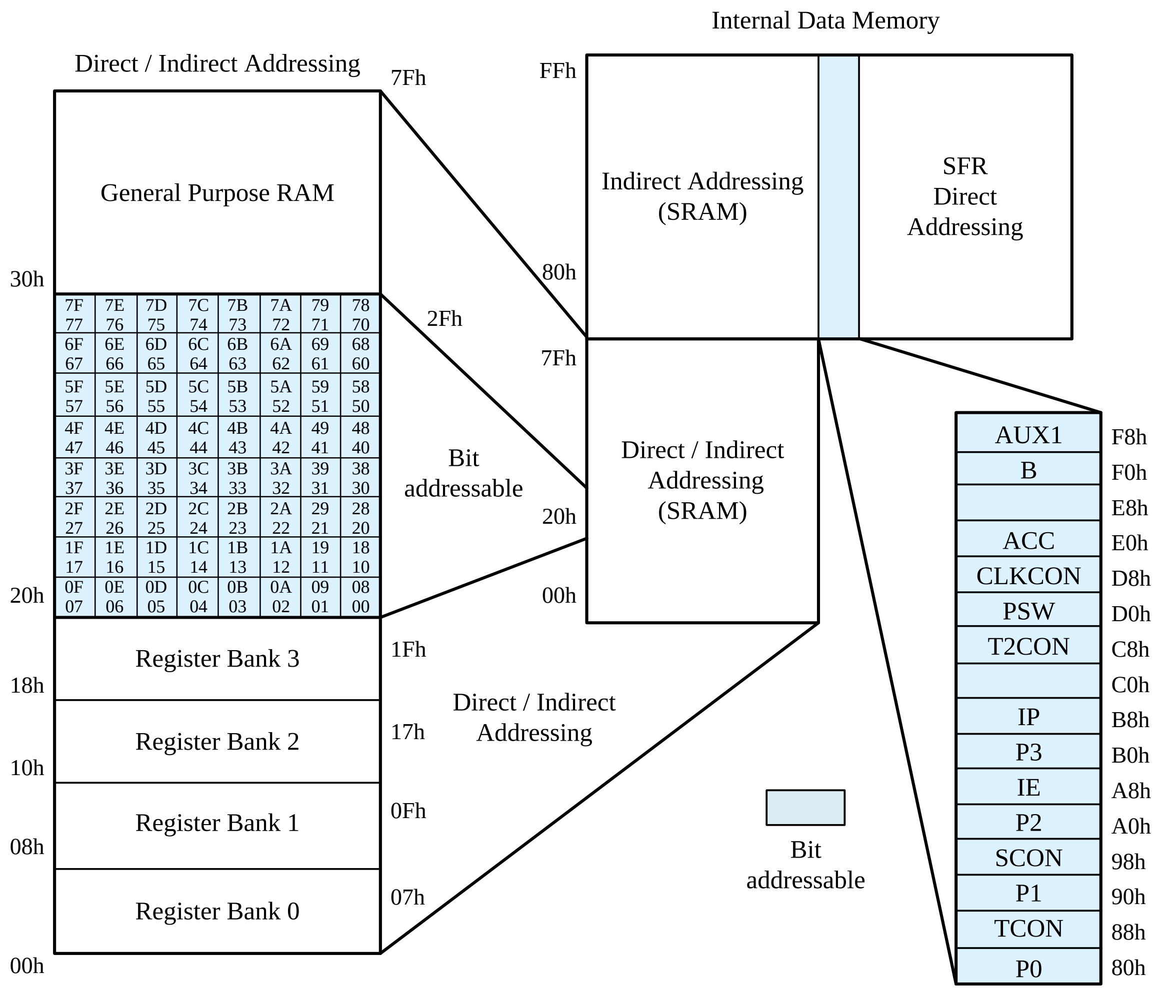
IRAM is located in the 8051 internal data memory space. The whole 256 Bytes IRAM are accessible using indirect addressing but only the lower 128 Bytes are accessible using direct addressing. There are four directly addressable register banks (switching by PSW), which occupy IRAM space from 00h to 1Fh.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

### 2.5.2 XRAM

XRAM is located in the 8051 external data memory space (address from FD00h to FFFFh). The 768 Bytes XRAM can be only accessed by "MOVX" instruction.

### 2.5.3 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/PWM/CMP…, 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 | | | | | CFGBG | CFGWL | AUX2 |
| E8h | | | PWM4D | | PWM5D | | PWM6D | AUX3 |
| E0h | ACC | MICON | MIDAT | LVRCON | LVDCON | | | |
| D8h | CLKCON | PWM0PRDH | PWM0PRDL | | PWM1PRD | | PWM3D | UARTBRP |
| D0h | PSW | PWM0DH | PWM0DL | | PWM1D | | PWM2D | TM3RLD |
| C8h | T2CON | IAPCON | RCP2L | RCP2H | TL2 | TH2 | | |
| C0h | | | | | LVDDT | DACDT | CMPCON | CMPPNS |
| B8h | IP | IPH | IP1 | IP1H | SPCON | SPSTA | SPDAT | |
| B0h | P3 | | | | | | WGCON | WGBUF |
| A8h | IE | INTE1 | ADCDL | ADCDH | | | ADCHSEL | PWMCON2 |
| A0h | P2 | PWMCON | P3MOD10 | P3MOD32 | P3MOD54 | P3MOD76 | PINMOD | P2MOD54 |
| 98h | SCON | SBUF | P1MOD10 | P1MOD32 | P1MOD54 | P1MOD76 | P2MOD10 | P2MOD32 |
| 90h | P1 | P0MOD10 | P0MOD32 | P0MOD54 | OPTION | INTFLG | P0MOD76 | SWCMD |
| 88h | TCON | TMOD | TL0 | TL1 | TH0 | TH1 | | |
| 80h | P0 | SP | DPL | DPH | INTE2 | INTFLG2 | | PCON |

**SFR table**

### 3. LVR and LVD setting

The Chip provides LVR function. There are 16-level LVR can be selected by LVRCON The SFR PWRSAV and LVRPD bits also affect LVR function as tables below.

| Operation Mode | SFR | | | LVR | Function | Note |
|---|---|---|---|---|---|---|
| | LVRPD | PWRSAV | LVRSEL | | | |
| Fast<br>Slow | 0 | X | 0000 | ON | LV Reset 1.70V | |
| | 0 | X | 0001 | ON | LV Reset 1.85V | |
| | 0 | X | 0010 | ON | LV Reset 2.00V | |
| | 0 | X | 0011 | ON | LV Reset 2.15V | |
| | 0 | X | 0100 | ON | LV Reset 2.30V | |
| | 0 | X | 0101 | ON | LV Reset 2.45V | |
| | 0 | X | 0110 | ON | LV Reset 2.60V | |
| | 0 | X | 0111 | ON | LV Reset 2.75V | |
| | 0 | X | 1000 | ON | LV Reset 2.90V | |
| | 0 | X | 1001 | ON | LV Reset 3.05V | |
| | 0 | X | 1010 | ON | LV Reset 3.20V | |
| | 0 | X | 1011 | ON | LV Reset 3.35V | |
| | 0 | X | 1100 | ON | LV Reset 3.50V | |
| | 0 | X | 1101 | ON | LV Reset 3.65V | |
| | 0 | X | 1110 | ON | LV Reset 3.80V | |
| | 0 | X | 1111 | ON | LV Reset 3.95V | |
| Idle<br>Stop<br>Halt | 0 | 0 | 0000 | ON | LV Reset 1.70V | Current consumption about 60uA |
| | 0 | 0 | 0001 | ON | LV Reset 1.85V | |
| | 0 | 0 | 0010 | ON | LV Reset 2.00V | |
| | 0 | 0 | 0011 | ON | LV Reset 2.15V | |
| | 0 | 0 | 0100 | ON | LV Reset 2.30V | |
| | 0 | 0 | 0101 | ON | LV Reset 2.45V | |
| | 0 | 0 | 0110 | ON | LV Reset 2.60V | |
| | 0 | 0 | 0111 | ON | LV Reset 2.75V | |
| | 0 | 0 | 1000 | ON | LV Reset 2.90V | |
| | 0 | 0 | 1001 | ON | LV Reset 3.05V | |
| | 0 | 0 | 1010 | ON | LV Reset 3.20V | |
| | 0 | 0 | 1011 | ON | LV Reset 3.35V | |
| | 0 | 0 | 1100 | ON | LV Reset 3.50V | |
| | 0 | 0 | 1101 | ON | LV Reset 3.65V | |
| | 0 | 0 | 1110 | ON | LV Reset 3.80V | |
| | 0 | 0 | 1111 | ON | LV Reset 3.95V | |
| Idle | 0 | 1 | XXXX | ON | Disable LVR<br>Enable POR 1.60V | Current consumption about 11uA |
| Stop<br>Halt | 0 | 1 | XXXX | OFF | Disable | *Minimum Current consumption about 0.6uA |
| Idle | 1 | X | XXXX | ON | Disable LVR<br>Enable POR 1.60V | Current consumption about 16uA |
| Stop<br>Halt | 1 | X | XXXX | OFF | Disable | *Minimum Current consumption about 0.6uA |

***Note:*** *The current consumption of Halt mode is more than Stop mode about 2~7uA, because SRC is enabled.*

***Note:*** *POR is always on and will consume about 0.6uA*

The Chip provides Low Voltage Detection (LVD) function. There are 16-level LVD can be selected by LVDCON. The SFR PWRSAV and LVDPD bits also affect LVD function as tables below. The LVD function supports two detection modes, below $V_{CC}$ voltage or above $V_{CC}$ voltage detection. In order to avoid false detection, the chip can choose whether the LVD hysteresis is on or off, and also provides a debounce function, and the debounce time can be selected from 0.96 us~60.76 us.

| Operation Mode | SFR | | | LVD | Function | Note |
|---|---|---|---|---|---|---|
| | LVDPD | PWRSAV | LVDSEL | | | |
| Fast<br>Slow | 0 | X | 0000 | ON | LV Detection 1.85V | |
| | 0 | X | 0001 | ON | LV Detection 2.00V | |
| | 0 | X | 0010 | ON | LV Detection 2.15V | |
| | 0 | X | 0011 | ON | LV Detection 2.30V | |
| | 0 | X | 0100 | ON | LV Detection 2.45V | |
| | 0 | X | 0101 | ON | LV Detection 2.60V | |
| | 0 | X | 0110 | ON | LV Detection 2.75V | |
| | 0 | X | 0111 | ON | LV Detection 2.90V | |
| | 0 | X | 1000 | ON | LV Detection 3.05V | |
| | 0 | X | 1001 | ON | LV Detection 3.20V | |
| | 0 | X | 1010 | ON | LV Detection 3.35V | |
| | 0 | X | 1011 | ON | LV Detection 3.50V | |
| | 0 | X | 1100 | ON | LV Detection 3.65V | |
| | 0 | X | 1101 | ON | LV Detection 3.80V | |
| | 0 | X | 1110 | ON | LV Detection 3.95V | |
| | 0 | X | 1111 | ON | LV Detection 4.10V | |
| Idle<br>Stop<br>Halt | 0 | 0 | 0000 | ON | LV Detection 1.85V | Current consumption about 60uA |
| | 0 | 0 | 0001 | ON | LV Detection 2.00V | |
| | 0 | 0 | 0010 | ON | LV Detection 2.15V | |
| | 0 | 0 | 0011 | ON | LV Detection 2.30V | |
| | 0 | 0 | 0100 | ON | LV Detection 2.45V | |
| | 0 | 0 | 0101 | ON | LV Detection 2.60V | |
| | 0 | 0 | 0110 | ON | LV Detection 2.75V | |
| | 0 | 0 | 0111 | ON | LV Detection 2.90V | |
| | 0 | 0 | 1000 | ON | LV Detection 3.05V | |
| | 0 | 0 | 1001 | ON | LV Detection 3.20V | |
| | 0 | 0 | 1010 | ON | LV Detection 3.35V | |
| | 0 | 0 | 1011 | ON | LV Detection 3.50V | |
| | 0 | 0 | 1100 | ON | LV Detection 3.65V | |
| | 0 | 0 | 1101 | ON | LV Detection 3.80V | |
| | 0 | 0 | 1110 | ON | LV Detection 3.95V | |
| | 0 | 0 | 1111 | ON | LV Detection 4.10V | |
| Idle | 0 | 1 | XXXX | ON | Disable LVD<br>Enable POR 1.60V | Current consumption about 11uA |
| Stop<br>Halt | 0 | 1 | XXXX | OFF | Disable | *Minimum Current consumption about 0.6uA |
| Idle | 1 | X | XXXX | ON | Disable LVD<br>Enable POR 1.60V | Current consumption about 16uA |
| Stop<br>Halt | 1 | X | XXXX | OFF | Disable | *Minimum Current consumption about 0.6uA |

| SFR E3h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **LVRCON** | – | – | – | LVRPD | LVRSEL | | | |
| R/W | – | – | – | R/W | R/W | | | |
| Reset | – | – | – | 0 | 0 | 0 | 0 | 0 |

E3h.4 **LVRPD:** Low Voltage Reset function select
0: LVR is enable
1: LVR is disable

E3h.3~0 **LVRSEL:** Low Voltage Reset select (step=0.15V)
0000:Set LVR at 1.70V
0001:Set LVR at 1.85V
0010:Set LVR at 2.00V
0011:Set LVR at 2.15V
0100:Set LVR at 2.30V
0101:Set LVR at 2.45V
0110:Set LVR at 2.60V
0111:Set LVR at 2.75V
1000:Set LVR at 2.90V
1001:Set LVR at 3.05V
1010:Set LVR at 3.20V
1011:Set LVR at 3.35V
1100:Set LVR at 3.50V
1101:Set LVR at 3.65V
1110:Set LVR at 3.80V
1111:Set LVR at 3.95V

| SFR **E4h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **LVDCON** | LVDM | LVDO | LVDHYS | LVDPD | LVDSEL | | | |
| R/W | R/W | R | R/W | R/W | R/W | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

E4h.7 **LVDM:** Low Voltage Detect function mode
0: $V_{CC} < V_{LVD}$ (LVDIF = 1 while LVDO = 1)
1: $V_{CC} > V_{LVD}$ (LVDIF = 1 while LVDO = 0)

E4h.6 **LVDO:** Low Voltage Detect real time output

E4h.5 **LVDHYS:** LVD Hysteresis Enable
0: LVD Hysteresis disable
1: LVD Hysteresis enable

E4h.4 **LVDPD:** Low Voltage Detect function select (Auto disable in Idle/Halt/Stop mode)
0: LVD enable
1: LVD disable

E4h.3~0 **LVDSEL:** Low Voltage Detect function select (step=0.15V)
0000: Set LVD at 1.85V
0001: Set LVD at 2.00V
0010: Set LVD at 2.15V
0011: Set LVD at 2.30V
0100: Set LVD at 2.45V
0101: Set LVD at 2.60V
0110: Set LVD at 2.75V
0111: Set LVD at 2.90V
1000: Set LVD at 3.05V
1001: Set LVD at 3.20V
1010: Set LVD at 3.35V
1011: Set LVD at 3.50V
1100: Set LVD at 3.65V
1101: Set LVD at 3.80V
1110: Set LVD at 3.95V
1111: Set LVD at 4.10V

| SFR **C4h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **LVDDT** | – | – | LVDDT | | | | | |
| R/W | – | – | R/W | | | | | |
| Reset | – | – | 0 | 0 | 0 | 0 | 0 | 0 |

C4h.5~0 **LVDDT:**LVD debounce time select (step=FRC16.588M/16，$T_{LVD}$=0.96 us)
00_0000: disable LVD debounce mode
00_0001: enable LVD debounce mode and the debounce time is 0.96 us ($T_{LVD}$*1)
...
11_1111: enable LVD debounce mode and the debounce time is 60.76 us ($T_{LVD}$*63)

| SFR **F7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX2** | WDTE | | PWRSAV | VBGOUT | – | IAPTE | | – |
| R/W | R/W | R/W | R/W | R/W | – | R/W | | – |
| Reset | 0 | 0 | 0 | 0 | – | 1 | 1 | – |

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

# 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 (POR)

After power-on reset, the device stays in the reset state and the preheating time of this chip is about 40 ms. A power-on reset requires the voltage on the VCC pin to discharge to near the VSS level before rising above 2.2V. POR is automatically turned off when the chip enters Halt/Stop mode and can be enabled or disabled by PORPD (E3h.5) when the chip enters Halt/Stop mode.

## 4.2 External Pin Reset (XRST)

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 (SWRST)

Software Reset is activated by writing data 56h to SWCMD (97h).

## 4.4 Watchdog Timer Reset (WDTR)

WDT overflow Reset is disabled or enabled by WDTE (F7h.7~6). The WDT uses SRC as its counting time base. It runs in Fast/Slow mode and runs or stops in Idle/Halt/Stop mode. The watchdog timer overflow speed can be defined by WDTPSC (94h.5~4). WDT is cleared by CLRWDT (F8h.7) or reset.

## 4.5 Low Voltage Reset (LVR)

Low voltage reset (LVR) can select 16 different voltage thresholds through LVRCON (E3h.3~0). When PWRSAV (F7h.5) =1, the LVR will automatically turn off when the chip enters Idle/Halt/Stop mode. It can be enabled or disabled by LVRPD (E3h.4).

***Note:*** *refer to AP-TM52XXXXX_02S for LVR setting information*

| Flash **3FFFh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **CFGWH** | PROT | XRSTE | – | – | – | – | – | – |

3FFFh.6 **XRSTE:** External Pin Reset control
0: Disable External Pin Reset
1: Enable External Pin Reset

| SFR **94h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **OPTION** | – | TM3CKS | WDTPSC | | ADCKS | | SXTGAIN | |
| R/W | – | R/W | R/W | | R/W | | R/W | |
| Reset | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

94h.5~4 **WDTPSC:** Watchdog Timer prescaler time select
00: 528ms WDT overflow rate
01: 264ms WDT overflow rate
10: 132ms WDT overflow rate
11: 66ms WDT overflow rate

| SFR **97h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **SWCMD** | IAPEN/SWRST | | | | | | | |
| R/W | W | | | | | | | |
| Reset | – | | | | | | | |

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

| SFR E3h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **LVRCON** | – | – | – | LVRPD | LVRSEL | | | |
| R/W | – | – | – | R/W | R/W | | | |
| Reset | – | – | – | 0 | 0 | 0 | 0 | 0 |

E3h.4 **LVRPD:** Low Voltage Reset function select
0: LVR is enable
1: LVR is disable

E3h.3~0 **LVRSEL:** Low Voltage Reset select (step=0.15V)
0000: Set LVR at 1.70V
0001: Set LVR at 1.85V
0010: Set LVR at 2.00V
0011: Set LVR at 2.15V
0100: Set LVR at 2.30V
0101: Set LVR at 2.45V
0110: Set LVR at 2.60V
0111: Set LVR at 2.75V
1000: Set LVR at 2.90V
1001: Set LVR at 3.05V
1010: Set LVR at 3.20V
1011: Set LVR at 3.35V
1100: Set LVR at 3.50V
1101: Set LVR at 3.65V
1110: Set LVR at 3.80V
1111: Set LVR at 3.95V

| SFR E4h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **LVDCON** | LVDM | LVDO | LVDHYS | LVDPD | LVDSEL | | | |
| R/W | R/W | R | R/W | R/W | R/W | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

E4h.7 **LVDM:** Low Voltage Detect function mode
0: $V_{CC} < V_{LVD}$ (LVDIF = 1 while LVDO = 1)
1: $V_{CC} > V_{LVD}$ (LVDIF = 1 while LVDO = 0)

E4h.6 **LVDO:** Low Voltage Detect real time output

E4h.5 **LVDHYS:** LVD Hysteresis Enable
0: LVD Hysteresis disable
1: LVD Hysteresis enable

E4h.4 **LVDPD:** Low Voltage Detect function select (Auto disable in Idle/Halt/Stop mode)
0: LVD enable
1: LVD disable

E4h.3~0 **LVDSEL:** Low Voltage Detect select (step=0.15V)
0000: Set LVD at 1.85V
0001: Set LVD at 2.00V
0010: Set LVD at 2.15V
0011: Set LVD at 2.30V
0100: Set LVD at 2.45V
0101: Set LVD at 2.60V
0110: Set LVD at 2.75V
0111: Set LVD at 2.90V
1000: Set LVD at 3.05V
1001: Set LVD at 3.20V
1010: Set LVD at 3.35V
1011: Set LVD at 3.50V
1100: Set LVD at 3.65V
1101: Set LVD at 3.80V
1110: Set LVD at 3.95V
1111: Set LVD at 4.10V

| SFR C4h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **LVDDT** | – | – | LVDDT | | | | | |
| R/W | – | – | R/W | | | | | |
| Reset | – | – | 0 | 0 | 0 | 0 | 0 | 0 |

C4h.5~0 **LVDDT:**LVD debounce time select (step=FRC16.588M/16， $T_{LVD}$=0.96 us)
00_0000: disable LVD debounce mode
00_0001: enable LVD debounce mode and the debounce time is 0.96 us ($T_{LVD}$*1)
...
11_1111: enable LVD debounce mode and the debounce time is 60.76 us ($T_{LVD}$*63)

| SFR F7h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX2** | WDTE | | PWRSAV | VBGOUT | – | IAPTE | | – |
| R/W | R/W | R/W | R/W | R/W | – | R/W | | – |
| Reset | 0 | 0 | 0 | 0 | – | 1 | 1 | – |

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 **PWRSAV:** 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 |
|---|---|---|---|---|---|---|---|---|
| **AUX1** | CLRWDT | CLRTM3 | – | ADSOC | CLRPWM0 | CLRPWM1 | WGEN | DPSEL |
| R/W | R/W | R/W | – | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | – | 0 | 1 | 1 | 0 | 0 |

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

# 5. Clock Circuitry & Operation Mode

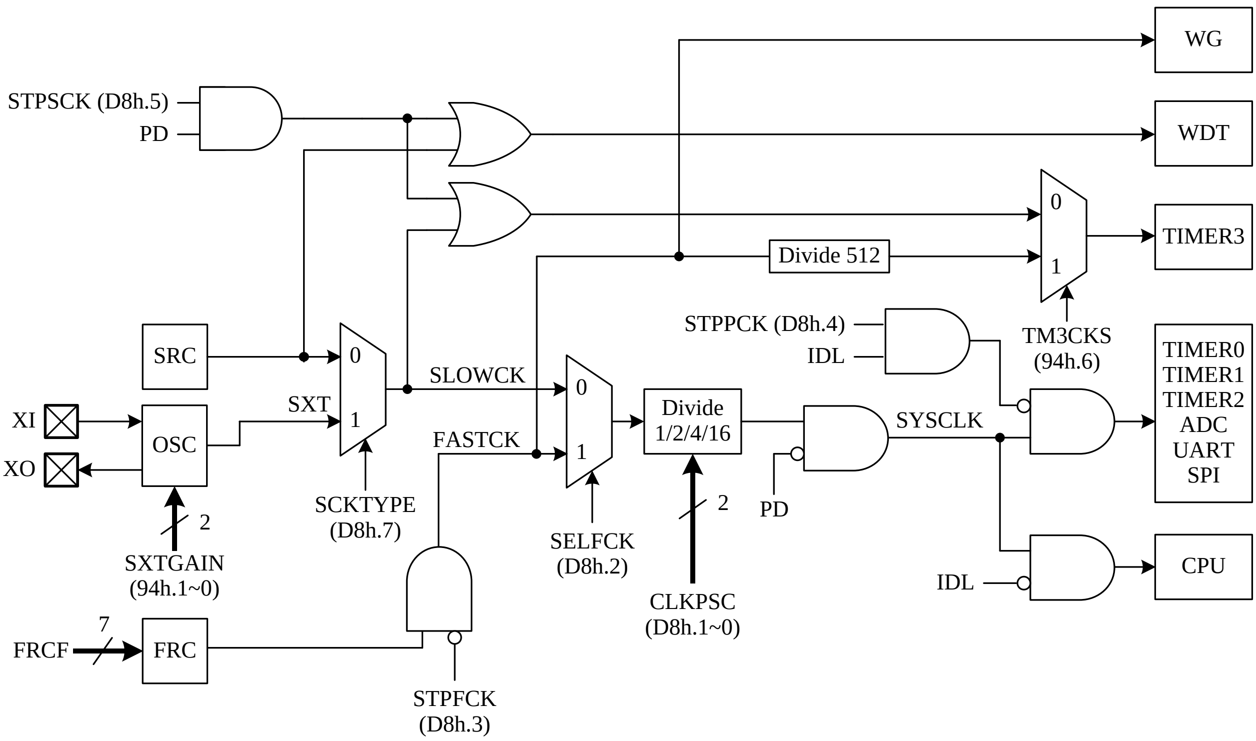
## 5.1 System Clock

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 can be selected as FRC (Fast Internal RC, 16.588 MHz). The Slow clock can be selected as SXT (Slow Crystal, 32768 Hz) or SRC (Slow Internal RC, 62 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 62 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, a 16.588MHz System clock rate requires $V_{CC}$> 1.9V.

The Chip has an external oscillators connected to the XI/XO pins. It relies on external circuitry for the clock signal and frequency stabilization, such as a stand-alone oscillator, quartz crystal, or ceramic resonator. In Slow mode, the slow oscillator can only use a clock frequency of 32.768 KHz. By setting the gain of the oscillator (SXTGAIN), the start time of the crystal oscillation can be shortened, and the consumption of oscillation current can also be reduced (Max gain is 3, Min gain is 0).

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.




***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) | | |
|---|---|---|---|
| | bit7 SCKTYPE | bit3 STPFCK | bit2 SELFCK |
| Fast FRC | 0/1 | 0 | 1 |
| Slow SXT | 1 | 0/1 | 0 |
| Slow SRC | 0 | 0/1 | 0 |
| Slow type change | 0 ← → 1 | 0 | 1 |
| Stop FRC | 0/1 | 0 → 1 | 0 |
| Switch to FRC | 0/1 | 0 | 0 → 1 |
| Switch to SRC/SXT | 0/1 | 0 | 1 → 0 |

| Flash **3FFDh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **CFGWL** | – | FRCF | | | | | | |

3FFDh.6~0 **FRCF:** FRC frequency adjustment.
FRC is trimmed to 16.588 MHz in chip manufacturing. FRCF records the adjustment data.

| SFR **F6h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **CFGWL** | – | 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 |
|---|---|---|---|---|---|---|---|---|
| **CLKCON** | SCKTYPE | – | STPSCK | STPPCK | STPFCK | SELFCK | CLKPSC | |
| R/W | R/W | – | R/W | R/W | R/W | R/W | R/W | |
| Reset | 0 | – | 1 | 0 | 0 | 0 | 1 | 1 |

D8h.7 **SCKTYPE:** Slow clock type. This bit can be changed only in Fast mode (SELFCK=1).
0: SRC
1: SXT, P2.0 and P2.1 are crystal pins

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 |
|---|---|---|---|---|---|---|---|---|
| **PCON** | 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 |
|---|---|---|---|---|---|---|---|---|
| **CLKCON** | SCKTYPE | – | STPSCK | STPPCK | STPFCK | SELFCK | CLKPSC | |
| R/W | R/W | – | R/W | R/W | R/W | R/W | R/W | |
| Reset | 0 | – | 1 | 0 | 0 | 0 | 1 | 1 |

D8h.7 **SCKTYPE:** Slow clock type. This bit can be changed only in Fast mode (SELFCK=1).
0: SRC 1: SXT, P2.0 and P2.1 are crystal pins

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 **F7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX2** | WDTE | | PWRSAV | VBGOUT | – | IAPTE | | – |
| R/W | R/W | R/W | R/W | R/W | – | R/W | | – |
| Reset | 0 | 0 | 0 | 0 | – | 1 | 1 | – |

F7h.4 **VBGOUT:** $V_{BG}$ 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.

| Vector | Flag | Description |
|---|---|---|
| 0003 | IE0 | INT0 external pin Interrupt (can wake up Halt/Stop mode) |
| 000B | TF0 | Timer0 Interrupt |
| 0013 | IE1 | INT1 external pin Interrupt (can wake up Halt/Stop mode) |
| 001B | TF1 | Timer1 Interrupt |
| 0023 | RI+TI | Serial Port (UART1) Interrupt |
| 002B | TF2+EXF2 | Timer2 Interrupt |
| 0033 | – | Reserved for ICE mode use |
| 003B | TF3 | Timer3 Interrupt |
| 0043 | PCIF | Port0~Port3 external pin change Interrupt (can wake up Halt/Stop mode) |
| 004B | WGIF | WG Interrupt |
| 0053 | ADIF | ADC Interrupt |
| 005B | SPIF<br>MIIF | SPI/Master I$^2$C Interrupt |
| 0063 | LVDIF | LVD Interrupt |
| 006B | CMPIF | CMP Interrupt |
| 0073 | PWM0IF<br>PWM1IF | PWM0~1 Interrupt |

**Interrupt Vector & Flag**

| Vector | Item | Interrupt enable | Sub-interrupt enable | Interrupt flag |
|---|---|---|---|---|
| 0003 | IE0 | IE A8.0 | | TCON 88.1 |
| 000B | TF0 | IE A8.1 | | TCON 88.5 |
| 0013 | IE1 | IE A8.2 | | TCON 88.3 |
| 001B | TF1 | IE A8.3 | | TCON 88.7 |
| 0023 | RI+TI | IE A8.4 | | SCON 98.1~0 |
| 002B | TF2+EXF2 | IE A8.5 | | T2CON C8.7~6 |
| 0033 | – | | | |
| 003B | TF3 | INTE1 A9.0 | | INTFLG 95.0 |
| 0043 | PCIF | INTE1 A9.1 | | INTFLG 95.1 |
| 004B | WGIF | INTE1 A9.2 | | INTFLG 95.3 |
| 0053 | ADIF | INTE1 A9.3 | | INTFLG 95.4 |
| 005B | SPIF<br>MIIF | INTE1 A9.4 | | SPSTA BD.7<br>MICON E1.5 |
| 0063 | LVDIF | INTE1 A9.5 | | INTFLG 95.7 |
| 006B | CMPIF | INTE1 A9.6 | | INTFLG 95.6 |
| 0073 | PWM0IF<br>PWM1IF | INTE1 A9.7 | INTE2 84.6<br>INTE2 84.5 | INTFLG2 85.6<br>INTFLG2 85.5 |

**Interrupt related SFRs**

## 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. To avoid writing and reading these SFRs before and after the interruption may cause inconsistencies. In addition, PWM0DH, PWM0DL, 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.

| SFR **A8h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IE** | 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 (UART) interrupt enable
0: Disable Serial Port (UART) interrupt
1: Enable Serial Port (UART) 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 |
|---|---|---|---|---|---|---|---|---|
| **INTE1** | PWMIE | CMPIE | LVDIE | SPI2CE | ADIE | WGIE | 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~PWM1 interrupt enable
0: Disable PWM0~PWM1 interrupt
1: Enable PWM0~PWM1 interrupt

A9h.6 **CMPIE:** CMP interrupt enable
0: Disable CMP interrupt
1: Enable CMP interrupt

A9h.5 **LVDIE:** LVD interrupt enable
0: Disable LVD interrupt
1: Enable LVD interrupt

A9h.4 **SPI2CE:** SPI/Master $I^2C$ interrupt enable
0: Disable SPI/ Master $I^2C$ interrupt
1: Enable SPI/ Master $I^2C$ interrupt

A9h.3 **ADIE:** ADC interrupt enable
0: Disable ADC interrupt
1: Enable ADC interrupt

A9h.2 **WGIE:** WG interrupt enable
0: Disable WG interrupt
1: Enable WG 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 **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.6 **PWM1IE:** PWM1 Interrupt Enable
0: disable
1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)

84h.5 **PWM0IE:** PWM0 Interrupt Enable
0: disable
1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)

| SFR **B9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IPH** | – | – | 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 |
|---|---|---|---|---|---|---|---|---|
| **IP** | – | – | 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 (UART) 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 |
|---|---|---|---|---|---|---|---|---|
| **IP1H** | PPWMH | PCMPH | PLVDH | PSPI2CH | PADIH | PWGH | 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 |
|---|---|---|---|---|---|---|---|---|
| **IP1** | PPWM | PCMP | PLVD | PSPI2C | PADI | PWG | 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~PWM1 Interrupt Priority control. (PPWMH，PPWM)=
11: Level 3 (highest priority)
10: Level 2
01: Level 1
00: Level 0 (lowest priority)

BBh.6, BAh.6 **PCMPH, PCMP:**CMP Interrupt Priority control. Definition as above.

BBh.5, BAh.5 **PLVDH, PLVD:**LVD Interrupt Priority control. Definition as above.

BBh.4, BAh.4 **PSPI2CH, PSPI2C:**SPI/Master $I^2C$ Interrupt Priority control. Definition as above.

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

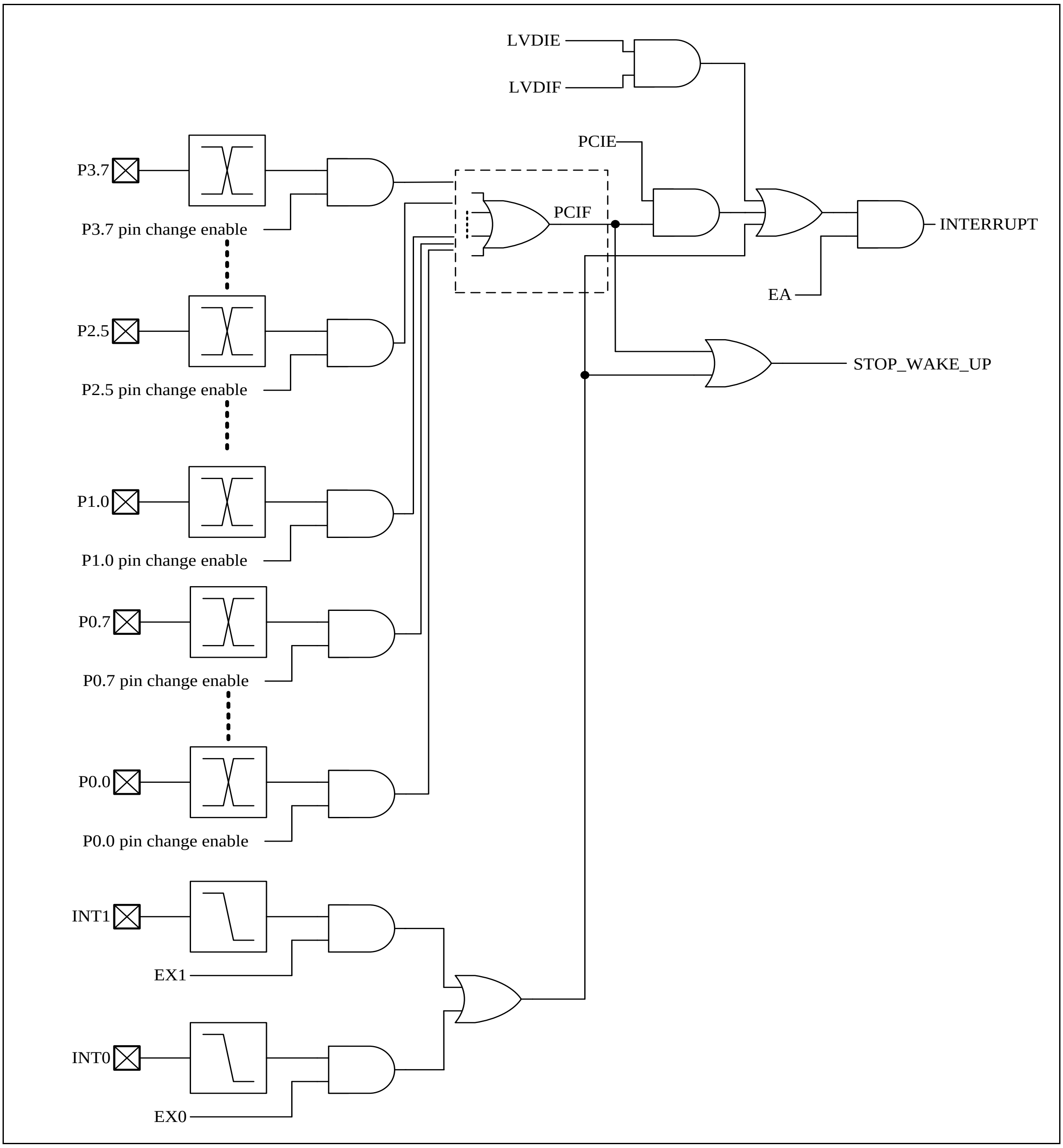
BBh.2, BAh.2 **PWGH, PWG:**WG Interrupt Priority control. Definition as above.

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 (P3.2), INT1 (P3.3) and Port0~Port3 pin change interrupt. These pins 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 I/O state change. For details, see Chapter 7. Pin Mode 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=0~1)*

| SFR **88h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **TCON** | 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 **95h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **INTFLG** | LVDIF | CMPIF | – | ADIF | WGIF | – | PCIF | TF3 |
| R/W | R/W | R/W | – | R/W | 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.

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).

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

| SFR **A8h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IE** | 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 |
|---|---|---|---|---|---|---|---|---|
| **INTE1** | PWMIE | CMPIE | LVDIE | SPI2CE | ADIE | WGIE | 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.5 **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

| SFR **E4h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **LVDCON** | LVDM | LVDO | LVDHYS | LVDPD | LVDSEL | | | |
| R/W | R/W | R | R/W | R/W | R/W | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

E4h.7 **LVDM:** Low Voltage Detect function mode
0: $V_{CC} < V_{LVD}$ (LVDIF = 1 while LVDO = 1)
1: $V_{CC} > V_{LVD}$ (LVDIF = 1 while LVDO = 0)

E4h.6 **LVDO:** Low Voltage Detect real time output

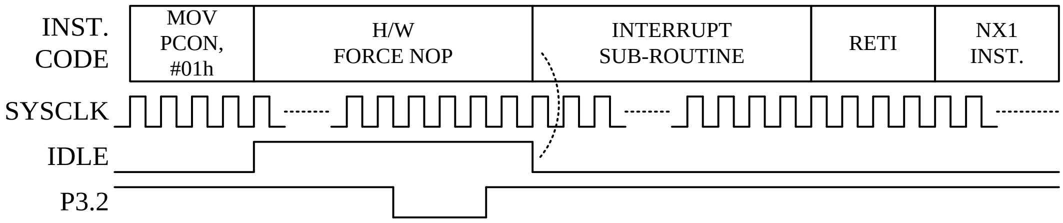
E4h.5 **LVDHYS:** LVD Hysteresis Enable
0: LVD Hysteresis disable
1: LVD Hysteresis enable

E4h.4 **LVDPD:** Low Voltage Detect function select (Auto disable in Idle/Halt/Stop mode)
0: LVD enable
1: LVD disable

E4h.3~0 **LVDSEL:** Low Voltage Detect select (step=0.15V)
0000: Set LVD at 1.85V
0001: Set LVD at 2.00V
0010: Set LVD at 2.15V
0011: Set LVD at 2.30V
0100: Set LVD at 2.45V
0101: Set LVD at 2.60V
0110: Set LVD at 2.75V
0111: Set LVD at 2.90V
1000: Set LVD at 3.05V
1001: Set LVD at 3.20V
1010: Set LVD at 3.35V
1011: Set LVD at 3.50V
1100: Set LVD at 3.65V
1101: Set LVD at 3.80V
1110: Set LVD at 3.95V
1111: Set LVD at 4.10V

## 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 UART) 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. When the EA is not set to 1 or the pin trigger state does not stay long enough, it will not wake up and will not generate an interrupt subroutine.




**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 |
|---|---|---|---|---|---|---|---|---|
| **PCON** | – | – | – | – | GF1 | GF0 | PD | IDL |
| R/W | – | – | – | – | R/W | R/W | R/W | R/W |
| Reset | – | – | – | – | 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

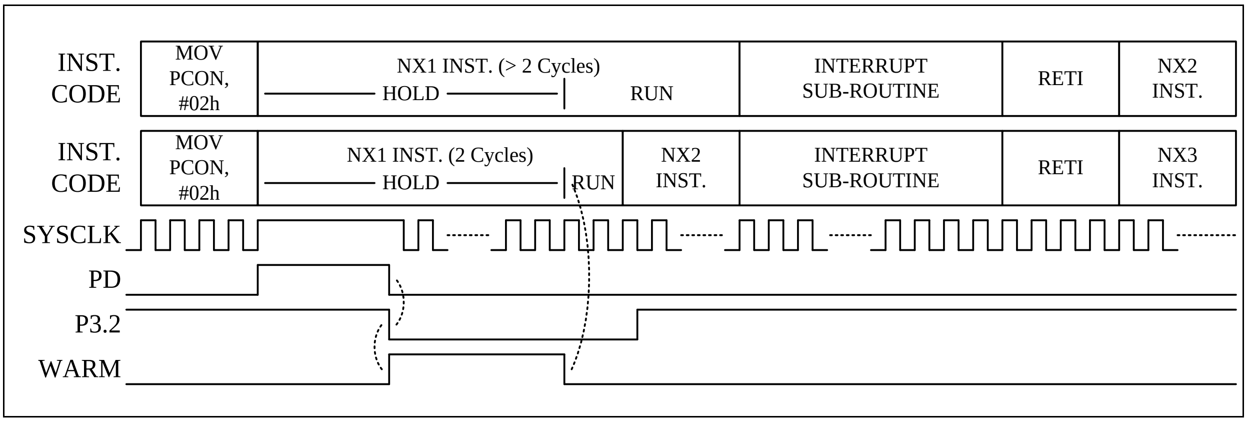
Each interrupt enable bit (e.g. TM3IE, EX0) and the EA bit must be set to 1 to establish the Halt/Stop mode interrupt function. All enabled interrupts (pins, Timer3) can wake up the CPU from Halt/Stop mode. Once Halt/Stop is woken up, if "the first instruction after PD (PCON.1) is set" is a two-cycle instruction, it will execute immediately before the interrupt is serviced, if "the first instruction after PD (PCON.1) is set" is a four-cycle or more long instruction, it will execute after the interrupt is serviced.

In addition to setting EX0/EX1, the INT0~1 pin interrupt needs to set EA=1 and the pin trigger state stays long enough (greater than 128 system clocks) to be sampled by the system clock, that is to say, when EA is not set to 1 or if the pin trigger state does not stay long enough, the CPU will only wake up without entering the interrupt subroutine.

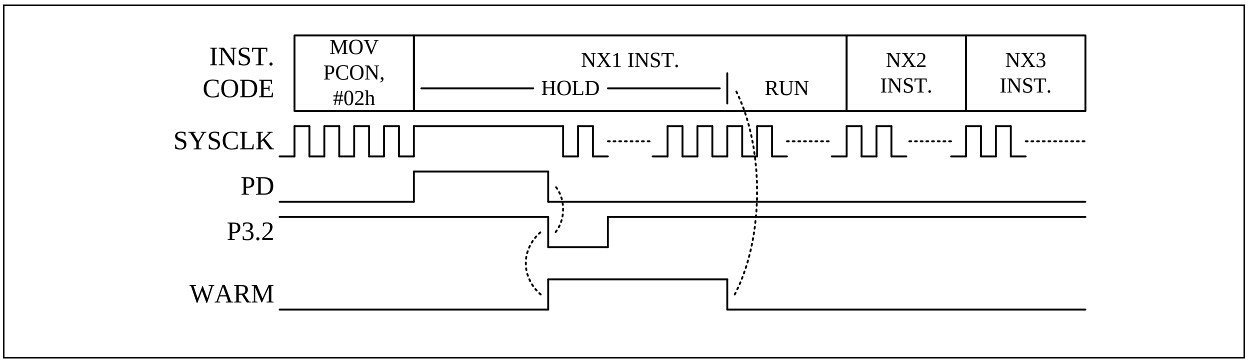
The same applies to the Port0~3 WKUP pin interrupt. In addition to setting pin mode, EA=1 is also required. When EA is not set to 1, the CPU will only wake up without entering the interrupt subroutine.

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

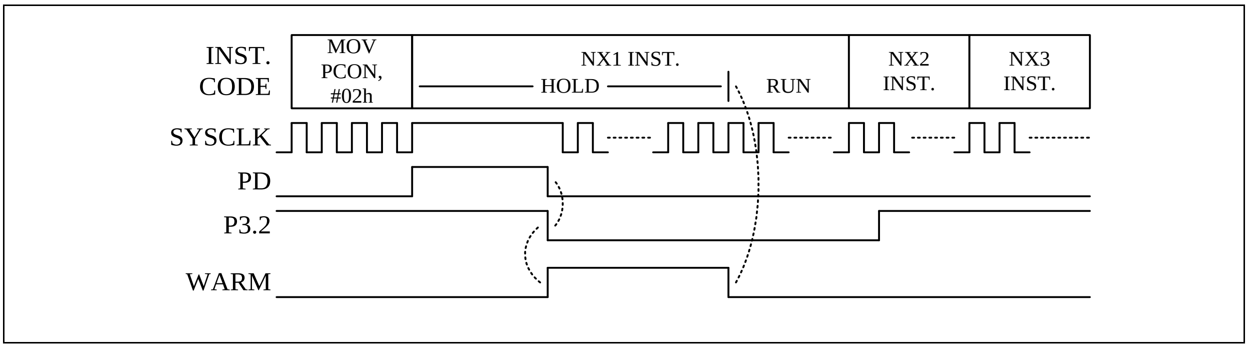
***Note:*** *The chip cannot enter Halt/Stop mode if the INTn pin is low and the INTn wake-up function is enabled. (INTn=0 and EXn=1, n=0~1)*




**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 30 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).

## 7.1 Port0~Port 3

These pins can operate in four different modes as below.

| PxMODn (x=0~3，n=0~7) | | | | | Pin State | Interrupt | Wake up | High Sink |
|---|---|---|---|---|---|---|---|---|
| **Mode0** | 0 | 0 | 0 | 0 | Open Drain with pull-up | Support* | Support* | - |
| **Mode1** | 0 | 0 | 0 | 1 | Open Drain **(default)** | Support* | Support* | - |
| **Mode2** | 0 | 0 | 1 | 0 | CMOS output | - | - | - |
| **Mode3** | 0 | 0 | 1 | 1 | ADC/CMP input | - | - | - |
| **Mode4** | 0 | 1 | 0 | 0 | Open Drain with pull-up | Support* | Support* | - |
| **Mode5** | 0 | 1 | 0 | 1 | Open Drain | Support* | Support* | - |
| **Mode6** | 0 | 1 | 1 | 0 | CMOS output | - | - | - |
| **Mode7** | 0 | 1 | 1 | 1 | LCD 1/2 $V_{CC}$ Bias output | - | - | - |
| **Mode8** | 1 | 0 | 0 | 0 | Open Drain with pull-up (for Halt/Stop mode pin change used) | Support | Support | - |
| **Mode9** | 1 | 0 | 0 | 1 | Open Drain (for Halt/Stop mode pin change used) | Support | Support | - |
| **Mode10** | 1 | 0 | 1 | 0 | CKO/WG/CMPO/TXD (CMOS output) | - | - | - |
| **Mode11** | 1 | 0 | 1 | 1 | PWMO (CMOS output) | - | - | - |
| **Mode12** | 1 | 1 | 0 | 0 | Open Drain with pull-up (for Halt/Stop mode pin change used) | Support | Support | Support |
| **Mode13** | 1 | 1 | 0 | 1 | Open Drain (for Halt/Stop mode pin change used) | Support | Support | Support |
| **Mode14** | 1 | 1 | 1 | 0 | CKO/WG/CMPO/TXD (CMOS output) | - | - | Support |
| **Mode15** | 1 | 1 | 1 | 1 | PWMO (CMOS output) | - | - | Support |

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

***Note****: *Only for INT0/INT1 pin*

<table>
<tr><th>Pin Mode</th><th>Port0~Port3 pin function</th><th>Px.n SFR data</th><th>Pin State</th><th>Resistor Pull-up</th><th>Digital Input</th></tr>
<tr><td rowspan="2">Mode0<br>Mode4<br>Mode8<br>Mode12</td><td rowspan="2">Open Drain with pull-up</td><td>0</td><td>Drive Low</td><td>N</td><td>N</td></tr>
<tr><td>1</td><td>Pull-up</td><td>Y</td><td>Y</td></tr>
<tr><td rowspan="2">Mode1<br>Mode5<br>Mode9<br>Mode13</td><td rowspan="2">Open Drain</td><td>0</td><td>Drive Low</td><td>N</td><td>N</td></tr>
<tr><td>1</td><td>Hi-Z</td><td>N</td><td>Y</td></tr>
<tr><td rowspan="2">Mode2<br>Mode6</td><td rowspan="2">CMOS Output</td><td>0</td><td>Drive Low</td><td>N</td><td>N</td></tr>
<tr><td>1</td><td>Drive High</td><td>N</td><td>N</td></tr>
<tr><td>Mode3</td><td>ADC/CMPchannel</td><td>X<br>(don’t care)</td><td>–</td><td>N</td><td>N</td></tr>
<tr><td>Mode7</td><td>LCD 1/2 $V_{CC}$ Bias output</td><td>X<br>(don’t care)</td><td>–</td><td>N</td><td>N</td></tr>
<tr><td>Mode10<br>Mode14</td><td>CKO/WG/CMPO/TXD<br>(CMOS output)</td><td>X<br>(don’t care)</td><td>–</td><td>N</td><td>N</td></tr>
<tr><td>Mode11<br>Mode15</td><td>PWMO<br>(CMOS output)</td><td>X<br>(don’t care)</td><td>–</td><td>N</td><td>N</td></tr>
</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 set the corresponding Port Data SFR to 1 to disable the pin’s output driving circuitry.

When user selects Mode2 or Mode6, the function is CMOS output, user can choose output low or high by port data value. When user selects Mode3, the function is for analog signal, such as ADC or CMP pin, the port type is Hi-Z and the digital input Schmitt-trigger is disabled in this mode.

Beside I/O port function, each Port0~Port3 pin has one or more alternative functions, such as ADC，CMP，LCD，WG，TXD and PWM. Most of the functions are activated by setting the individual pin mode control SFR to Mode3，Mode7，Mode10，Mode11，Mode14 or Mode 5. Port1/Port3 pins have standard 8051 auxiliary definition such as INT0/INT1. These pin functions need to set the pin mode SFR to Mode4，Mode5，Mode8，Mode9，Mode12 or Mode13 and keep the Px.n (x=0~3，n=0~7) SFR at 1.

| Pin Name | Wake up Interrupt | ADC | PWM | LCD | UART | MI$^2$C | SPI | CMP | WG | others |
|---|---|---|---|---|---|---|---|---|---|---|
| P0.7 | Y | AD7 | PWM6 | LCDC7 | | | | CIN3 | | |
| P0.6 | Y | AD6 | PWM1 | LCDC6 | | | | | | |
| P0.5 | Y | AD5 | PWM5 | LCDC5 | | | | | WG$_{(m)}$ | |
| P0.4 | Y | AD4 | PWM0N | LCDC4 | | | | | | |
| P0.3 | Y | AD3 | PWM0P | LCDC3 | | | | | | |
| P0.2 | Y | AD2 | PWM1 | LCDC2 | RXD$_{(1)}$<br>TXD$_{(m)}$ | SCL$_{(1)}$ | | | | |
| P0.1 | Y | AD1 | PWM5 | LCDC1 | | | | CIN4 | | |
| P0.0 | Y | AD0 | PWM4 | LCDC0 | | | | CIP4 | | |
| P1.7 | Y | AD8 | PWM2 | | | | | CIP3 | | |
| P1.6 | Y | AD15 | PWM3 | | RXD$_{(3)}$<br>TXD$_{(m)}$ | SDA$_{(1)}$ | | CIN1 | | |
| P1.5 | Y | AD20 | PWM4 | | | | | CIN2 | | |
| P1.4 | Y | AD21 | PWM5 | | | | | CIP2 | | CKO |
| P1.3 | Y | AD22 | PWM6 | | | SCL$_{(0)}$ | | | | |
| P1.2 | Y | AD25 | PWM1 | | | | SCK$_{(1)}$ | | WG$_{(m)}$ | |
| P1.1 | Y | AD26 | PWM2 | | | | MISO$_{(1)}$ | | | T2EX |
| P1.0 | Y | AD27 | PWM3 | | | | MOSI$_{(1)}$ | | | T2 |
| P2.5 | Y | AD24 | PWM4 | | | SDA$_{(3)}$ | | | | |
| P2.4 | Y | AD23 | PWM2 | | | SCL$_{(3)}$ | | | | |
| P2.3 | Y | | PWM3 | | | | | | | XI |
| P2.2 | Y | | PWM5 | | | | | | | XO |
| P2.1 | Y | AD18 | PWM6 | | | SDA$_{(2)}$ | MISO$_{(0)}$ | CMPO$_{(m)}$ | | PSDA1 |
| P2.0 | Y | AD19 | PWM2 | | | SCL$_{(2)}$ | MOSI$_{(0)}$ | | | PSCL1 |
| P3.7 | Y | AD12 | PWM4 | | | | | | WG$_{(m)}$ | RSTn |
| P3.6 | Y | AD16 | PWM0N | | | | | CMPO$_{(m)}$ | | |
| P3.5 | Y | AD17 | PWM0P | | | SDA$_{(0)}$ | SCK$_{(0)}$ | | | T1 |
| P3.4 | Y | AD9 | PWM3 | | | | | | WG$_{(m)}$ | T0 |
| P3.3 | Y | AD14 | PWM1 | | | | | CIP1 | | INT1 |
| P3.2 | Y | AD13 | PWM6 | | | | | | | INT0<br>VBGO |
| P3.1 | Y | AD10 | | | RXD$_{(2)}$<br>TXD$_{(m)}$ | | | | | PSDA0 |
| P3.0 | Y | AD11 | | | RXD$_{(0)}$<br>TXD$_{(m)}$ | | | | | PSCL0 |

**Port0~Port3 multi-function Table**

*注:(m)＝need to set Mode 10 or Mode 14*

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

| Alternative Function | PxMODn (x=0~3，n=0~7) | Px.n SFR data | Pin State | Other necessary SFR setting |
|---|---|---|---|---|
| INT0，INT1 | **0000** | 1 | Input with Pull-up | |
| | **0001** | 1 | Input | |
| T0，T1，T2，T2EX | **0000** | 1 | Input with Pull-up | |
| | **0001** | 1 | Input | |
| XI，XO | **0000** | 1 | Crystal oscillation | CLKCON |
| VBGO | **0011** | X | Bandgap Voltage output | AUX2 |
| AD0~AD27 | **0011** | X | ADC Channel | ADCHSEL |
| CIN1~CIN4<br>CIP1~CIP4 | **0011** | X | CMP Channel | CMPPNS |
| LCDC0~LCDC7 | **0111** | X | LCD 1/2 $V_{CC}$ Bias output | |
| CKO | **1x10** | X | Clock Output (CMOS Push-Pull) | |
| CMPO | **1x10** | X | CMP Output (CMOS Push-Pull) | |
| WG | **1x10** | X | WG Output (CMOS Push-Pull) | |
| RXD | **0000** | 1 | UART RX (Input with Pull-up) | PINMOD |
| | **0001** | 1 | UART RX (Input) | |
| TXD | **1x10** | | UART TX Output (CMOS Push-Pull) | |
| SPI Master Mode MISO | **0001** | 1 | SPI Data Input | SPCON<br>PINMOD |
| SPI Master Mode SCK，MOSI | **0010** | X | SPI Clock/Data Output (CMOS Push-Pull) | |
| SPI Slave Mode MISO | **0010** | X | SPI Data Output (CMOS Push-Pull) | |
| SPI Slave Mode SCK，MOSI | **0001** | 1 | SPI Clock/Data Input | |
| Master $I^2C$ SCL | **0000** | X | $I^2C$ Clock Output (Open Drain Output, Pull-up) | MICON<br>PINMOD |
| | **0010** | X | $I^2C$ Clock Output (CMOS Push-Pull) | |
| Master $I^2C$ SDA | **0000** | 1 | $I^2C$ Data (Pull-up) | |
| PWM0N/P<br>PWM1~PWM6 | **1x11** | X | PWM Output (CMOS Push-Pull) | |

For tables above, a "**CMOS Push-Pull 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.

This chip supports I/O High-sink function. This is an option by setting the pin mode to Mode12, Mode13, Mode14 or Mode15.

| SFR 80h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P0** | P0.7 | P0.6 | 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 | R/W | R/W |
| Reset | 1 | 1 | 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 |
|---|---|---|---|---|---|---|---|---|
| **P1** | 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 A0h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P2** | P2.7 | P2.6 | P2.5 | P2.4 | P2.3 | P2.2 | P2.1 | P2.0 |
| 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 |

A0h.7~0 **P2:** Port2 data

| SFR B0h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P3** | 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 |
|---|---|---|---|---|---|---|---|---|
| **P0MOD10** | P0MOD1 | | | | P0MOD0 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

91h.7~4 **P0MOD1:** P0.1 pin control
**0000~1111:** see Table 7.1
91h.3~0 **P0MOD0:** P0.0 pin control
**0000~1111:** see Table 7.1

| SFR 92h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P0MOD32** | P0MOD3 | | | | P0MOD2 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

92h.7~4 **P0MOD3:** P0.3 pin control
**0000~1111:** see Table 7.1
92h.3~0 **P0MOD2:** P0.2 pin control
**0000~1111:** see Table 7.1

| SFR 93h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P0MOD54** | P0MOD5 | | | | P0MOD4 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

93h.7~4 **P0MOD5:** P0.5 pin control
**0000~1111:** see Table 7.1
93h.3~0 **P0MOD4:** P0.4 pin control
**0000~1111:** see Table 7.1

| SFR **96h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P0MOD76** | P0MOD7 | | | | P0MOD6 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

96h.7~4 **P0MOD7:** P0.7 pin control
0000~1111**:** see Table 7.1
96h.3~0 **P0MOD6:** P0.6 pin control
0000~1111**:** see Table 7.1

| SFR **9Ah** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P1MOD10** | P1MOD1 | | | | P1MOD0 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

9Ah.7~4 **P1MOD1:** P1.1 pin control
0000~1111**:** see Table 7.1
9Ah.3~0 **P1MOD0:** P1.0 pin control
0000~1111**:** see Table 7.1

| SFR **9Bh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P1MOD32** | P1MOD3 | | | | P1MOD2 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

9Bh.7~4 **P1MOD3:** P1.3 pin control
0000~1111**:** see Table 7.1
9Bh.3~0 **P1MOD2:** P1.2 pin control
0000~1111**:** see Table 7.1

| SFR **9Ch** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P1MOD54** | P1MOD5 | | | | P1MOD4 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

9Ch.7~4 **P1MOD5:** P1.5 pin control
0000~1111**:** see Table 7.1
9Ch.3~0 **P1MOD4:** P1.4 pin control
0000~1111**:** see Table 7.1

| SFR **9Dh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P1MOD76** | P1MOD7 | | | | P1MOD6 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

9Dh.7~4 **P1MOD7:** P1.7 pin control
0000~1111**:** see Table 7.1
9Dh.3~0 **P1MOD6:** P1.6 pin control
0000~1111**:** see Table 7.1

| SFR **9Eh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P2MOD10** | P2MOD1 | | | | P2MOD0 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

9Eh.7~4 **P2MOD1:** P2.1 pin control
0000~1111**:** see Table 7.1
9Eh.3~0 **P2MOD0:** P2.0 pin control
0000~1111**:** see Table 7.1

| SFR **9Fh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P2MOD32** | P2MOD3 | | | | P2MOD2 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

9Fh.7~4 **P2MOD3:** P2.3 pin control
0000~1111**:** see Table 7.1
9Fh.3~0 **P2MOD2:** P2.2 pin control
0000~1111**:** see Table 7.1

| SFR **A7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P2MOD54** | P2MOD5 | | | | P2MOD4 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

A7h.7~4 **P2MOD5:** P2.5 pin control
0000~1111**:** see Table 7.1
A7h.3~0 **P2MOD4:** P2.4 pin control
0000~1111**:** see Table 7.1

| SFR **A2h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P3MOD10** | P3MOD1 | | | | P3MOD0 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

A2h.7~4 **P3MOD1:** P3.1 pin control
0000~1111**:** see Table 7.1
A2h.3~0 **P3MOD0:** P3.0 pin control
0000~1111**:** see Table 7.1

| SFR **A3h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P3MOD32** | P3MOD3 | | | | P3MOD2 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

A3h.7~4 **P3MOD3:** P3.3 pin control
0000~1111**:** see Table 7.1
A3h.3~0 **P3MOD2:** P3.2 pin control
0000~1111**:** see Table 7.1

| SFR **A4h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P3MOD54** | P3MOD5 | | | | P3MOD4 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

A4h.7~4 **P3MOD5:** P3.5 pin control
0000~1111**:** see Table 7.1
A4h.3~0 **P3MOD4:** P3.4 pin control
0000~1111**:** see Table 7.1

| SFR **A5h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **P3MOD76** | P3MOD7 | | | | P3MOD6 | | | |
| R/W | R/W | | | | R/W | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

A5h.7~4 **P1MOD7:** P3.7 pin control
0000~1111**:** see Table 7.1
A5h.3~0 **P1MOD6:** P3.6 pin control
0000~1111**:** see Table 7.1

| SFR **A6h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PINMOD** | – | SPIPS | UARTRXS | | MSDAPS | | MSCLPS | |
| R/W | – | R/W | R/W | | R/W | | R/W | |
| Reset | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

A6h.6 **SPIPS:** SPI pin select
0: SCK/MOSI/MISO = P3.5/P2.0/P2.1
1: SCK/MOSI/MISO = P1.2/P1.0/P1.1

A6h.5~4 **UARTRXS:** UART RXD pin select (TXD pin select by PINMOD)
00: RXD = P3.0
01: RXD = P0.2
10: RXD = P3.1
11: RXD = P1.6

A6h.3~2 **MSDAPS:** Master I$^2$C SDA pin select
00: P3.5
01: P1.6
10: P2.1
11: P2.5

A6h.1~0 **MSCLPS:** Master I$^2$C SCL pin select
00: P1.3
01: P0.2
10: P2.0
11: P2.4

| SFR **BCh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **SPCON** | SPEN | MSTR | CPOL | CPHA | – | LSBF | SPCR | |
| R/W | R/W | R/W | R/W | R/W | – | R/W | R/W | |
| Reset | 0 | 0 | 0 | 0 | – | 0 | 0 | 0 |

BCh.7 **SPEN:** SPI enable
0: SPI disable
1: SPI enable

| SFR **C7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **CMPPNS** | SCMPN | SCIN | | | – | SCIP | | |
| R/W | R/W | R/W | | | – | R/W | | |
| Reset | 1 | 1 | 1 | 1 | – | 1 | 1 | 1 |

C7h.7 **SCMPN:** Comparator CMPN source select
0: Comparator CMPN source is external pin (CINx)
1: Comparator CMPN source is DAC output

C7h.6~4 **SCIN:** Comparator CMPN external pin select
000: Comparator CMPN external input is CIN1 (P1.6)
001: Comparator CMPN external input is CIN2 (P1.5)
010: Comparator CMPN external input is CIN3 (P0.7)
011: Comparator CMPN external input is CIN4 (P0.1)
100: Comparator CMPN external input is CIN5 ($V_{SS}$)
other: No connection

C7h.2~0 **SCIP:** Comparator CMPP external pin select
000: Comparator CMPP external input is CIP1 (P3.3)
001: Comparator CMPP external input is CIP2 (P1.4)
010: Comparator CMPP external input is CIP3 (P1.7)
011: Comparator CMPP external input is CIP4 (P0.0)
100: Comparator CMPP external input is CIP5 ($V_{SS}$)
other: No connection

| SFR AEh | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| ADCHSEL | ADCHS | | | | | ADCVREFS | ADCVBGS | |
| R/W | R/W | | | | | R/W | R/W | |
| Reset | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

AEh.7~3 **ADCHS:** ADC channel select

| | |
|---|---|
| 00000: AD0 (P0.0) | 10000: AD16 (P3.6) |
| 00001: AD1 (P0.1) | 10001: AD17 (P3.5) |
| 00010: AD2 (P0.2) | 10010: AD18 (P2.1) |
| 00011: AD3 (P0.3) | 10011: AD19 (P2.0) |
| 00100: AD4 (P0.4) | 10100: AD20 (P1.5) |
| 00101: AD5 (P0.5) | 10101: AD21 (P1.4) |
| 00110: AD6 (P0.6) | 10110: AD22 (P1.3) |
| 00111: AD7 (P0.7) | 10111: AD23 (P2.4) |
| 01000: AD8 (P1.7) | 11000: AD24 (P2.5) |
| 01001: AD9 (P3.4) | 11001: AD25 (P1.2) |
| 01010: AD10 (P3.1) | 11010: AD26 (P1.1) |
| 01011: AD11 (P3.0) | 11011: AD27 (P1.0) |
| 01100: AD12 (P3.7) | 11100: $V_{BG}$ |
| 01101: AD13 (P3.2) | 11101: DAC |
| 01110: AD14 (P3.3) | 11110: Reserved |
| 01111: AD15 (P1.6) | 11111: $V_{CC}/4$ |

| SFR D8h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| CLKCON | SCKTYPE | – | STPSCK | STPPCK | STPFCK | SELFCK | CLKPSC | |
| R/W | R/W | – | R/W | R/W | R/W | R/W | R/W | |
| Reset | 0 | – | 1 | 0 | 0 | 0 | 1 | 1 |

D8h.7 **SCKTYPE:** Slow clock type. This bit can be changed only in Fast mode (SELFCK=1).
0: SRC
1: SXT, P2.0 and P2.1 are crystal pins

| SFR E1h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| MICON | 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 $I^2C$ enable
0: disable
1: enable

| SFR F7h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| AUX2 | WDTE | | PWRSAV | VBGOUT | – | IAPTE | | – |
| R/W | R/W | R/W | R/W | R/W | – | R/W | | – |
| Reset | 0 | 0 | 0 | 0 | – | 1 | 1 | – |

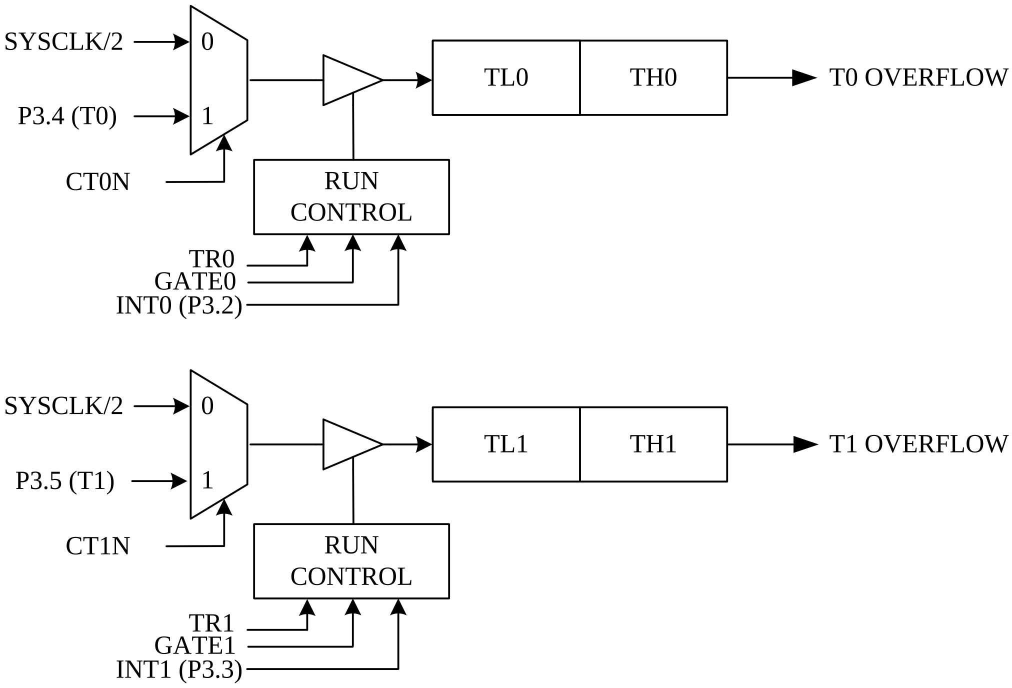
F7h.4 **VBGOUT:** $V_{BG}$ voltage output to P3.2
0: Disable
1: Enable

# 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. Timer3 is provided for a real-time clock count, when its time base is SXT.

## 8.1 Timer0/1

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 |
|---|---|---|---|---|---|---|---|---|
| **TCON** | 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 |
|---|---|---|---|---|---|---|---|---|
| **TMOD** | 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 |
|---|---|---|---|---|---|---|---|---|
| **TL0** | 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 |
|---|---|---|---|---|---|---|---|---|
| **TL1** | 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 |
|---|---|---|---|---|---|---|---|---|
| **TH0** | TH0 | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

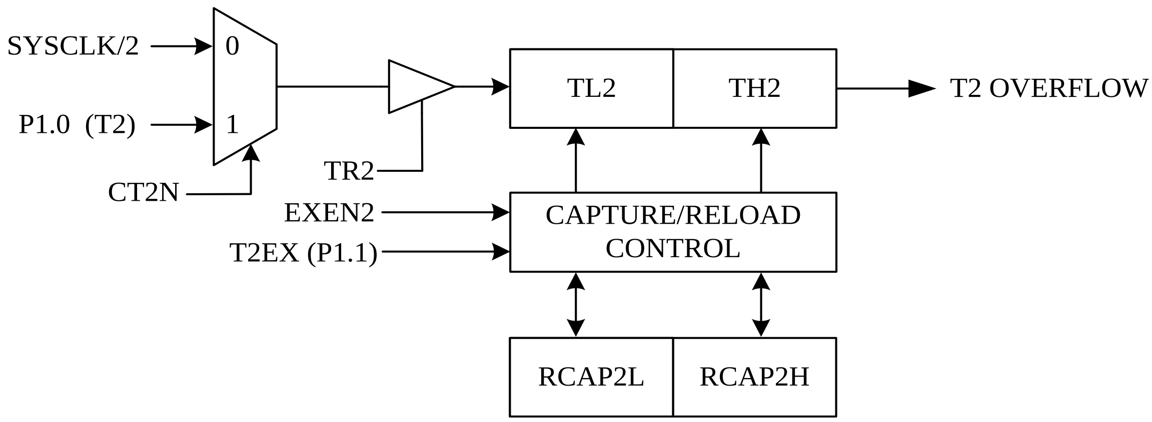
| SFR **8Dh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **TH1** | TH1 | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

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

## 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 |
|---|---|---|---|---|---|---|---|---|
| **T2CON** | 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:** keep this force 0

C8h.4 **TCLK:** keep this force 0

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 |
|---|---|---|---|---|---|---|---|---|
| **RCP2L** | 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 |
|---|---|---|---|---|---|---|---|---|
| **RCP2H** | 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 |
|---|---|---|---|---|---|---|---|---|
| **TL2** | TL2 | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

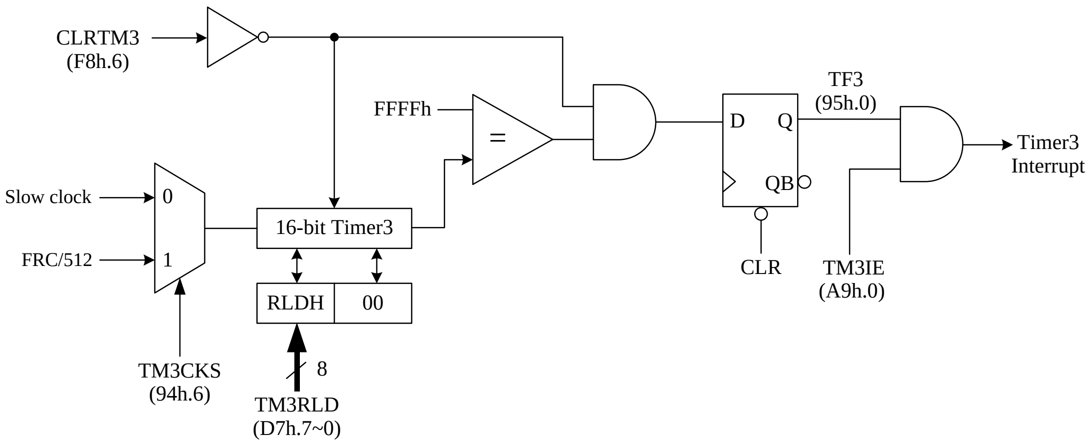
| SFR **CDh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **TH2** | TH2 | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

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

## 8.3 Timer3

Timer3 works as a 16-bit time-base counter, which generates interrupts periodically. Besides, Timer3 increases itself periodically and automatically reloads a new "offset value" (TM3RLD) from SFR TM3RLD into MSB 8-bit data of Timer3. The counting range is from [TM3RLD, 00h] to FFFFh. Timer3 can be stopped counting if the CLRTM3 bit is set. The Timer3 clock source is Slow clock (SRC or SXT) or FRC/512. This is ideal for real-time-clock (RTC) functionality when the clock source is SXT.




**Timer3 Structure**

| SFR 94h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **OPTION** | – | TM3CKS | WDTPSC | | ADCKS | | SXTGAIN | |
| R/W | – | R/W | R/W | | R/W | | R/W | |
| Reset | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

94h.6 **TM3CKS:** Timer3 clock source select.
0: Slow Clock (SRC/SXT)
1: FRC/512 (32.4KHz)

| SFR **D7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **TM3RLD** | TM3RLD | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

D7h.7~0 **TM3RLD:** 16-bit TM3 MSB 8-bit reload data
count range: [TM3RLD,00h]~FFFF

| SFR 95h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **INTFLG** | LVDIF | CMPIF | – | ADIF | WGIF | – | PCIF | TF3 |
| R/W | R/W | R/W | – | R/W | 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 counts to FFFFh. 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 **F8h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX1** | CLRWDT | CLRTM3 | – | ADSOC | CLRPWM0 | CLRPWM1 | WGEN | DPSEL |
| R/W | R/W | R/W | – | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | – | 0 | 1 | 1 | 0 | 0 |

F8h.6 **CLRTM3:** Set 1 to clear and hold Timer3, need S/W clear.

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

## 9. UART

The UART uses SCON, SBUF_TX and SBUF_RX SFRs. SCON is the control register, SBUF_TX and SBUF_RX are the data register. Data is written to SBUF_TX for transmission, while when SBUF_RX is read, received data is obtained. The SBUF_TX and SBUF_RX are completely independent. Transmit and receive data are pinned in P0.2, P1.6, P3.0 or P3.1, the transmit pin is selected through pin mode, and the receive pin is selected via SFR PINMOD.

$F_{SYSCLK}$ denotes System clock frequency, the UART Baud Rate is calculated as below.

**UART** Baud Rate setting:

- **Mode 1，3:** Baud Rate= $F_{SYSCLK}$ /16/UARTBRP

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

***Note:*** *also refer to Chapter 7 for more details on UART pin mode settings.*

| SFR **A6h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PINMOD** | – | SPIPS | UARTRXS | | MSDAPS | | MSCLPS | |
| R/W | – | R/W | R/W | | R/W | | R/W | |
| Reset | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

A6h.5~4 **UARTRXS:** UART RXD pin select (TXD pin select by Pin Control)
00: RXD = P3.0
01: RXD = P0.2
10: RXD = P3.1
11: RXD = P1.6

| SFR **98h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **SCON** | 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:** UART serial port mode select bit 0,1
00: Reserved
01: Mode1: 8 bit UART, Baud Rate is variable.
10: Reserved
11: Mode3: 9 bit UART, 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 |
|---|---|---|---|---|---|---|---|---|
| **SBUF** | SBUF_RX | | | | | | | |
| R/W | R | | | | | | | |
| Reset | – | – | – | – | – | – | – | – |

99h.7~0 **SBUF_RX:** UART receive data.
(Transmit data is written to this location and receive data is read from this location, but the paths are independent.)

| SFR **99h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **SBUF** | SBUF_TX | | | | | | | |
| R/W | W | | | | | | | |
| Reset | – | – | – | – | – | – | – | – |

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

| SFR **DFh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **UARTBRP** | UARTBRP | | | | | | | |
| 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 |

DFh.7~0 **UARTBRP:** Define UART Baud Rate prescaler.
UART Baud Rate = $F_{SYSCLK}$/16/UARTBRP

| SFR **A8h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **IE** | 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 |

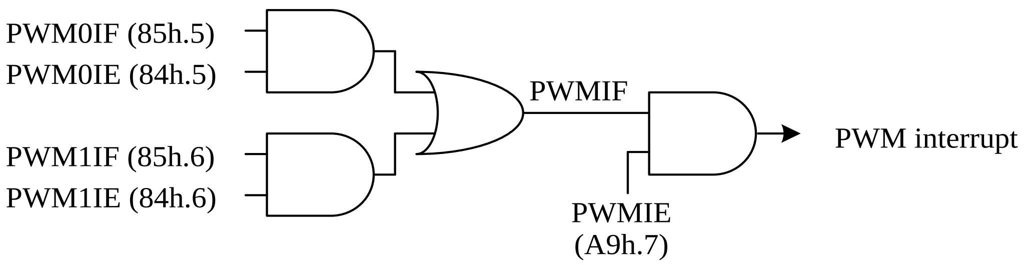
A8h.4 **ES:** Serial Port (UART) interrupt enable
0: Disable Serial Port (UART) interrupt
1: Enable Serial Port (UART) interrupt

# 10. PWMs

This Chip has one 16-bit PWM0 and six 8-bit PWM1~PWM6 modules, The PWM0 has independent 16-bit duty cycle control registers and 16-bit period registers. PWM1~PWM6 has independent 8-bit duty cycle control registers and shares a set of 8-bit period registers. PWM0 can generate a variable frequency waveform with 65536 duty cycle resolution based on the PWM clock. Although PWM1~PWM6 can only produce a resolution of 256 duty cycle, PWM1~PWM6 also has a PWM clock pre-divider, which can be used to add more variable frequency waveforms.The PWM clock can select FRC double frequency (FRC x 2), FRC or $F_{SYSCLK}$ as its clock source. Users should pay attention to the setting; the period of PWM must be greater than duty.

The 16-bit PWM0PRD and PWM0D registers all have a low and high byte structure. The high bytes can be directly accessed, but as 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.**

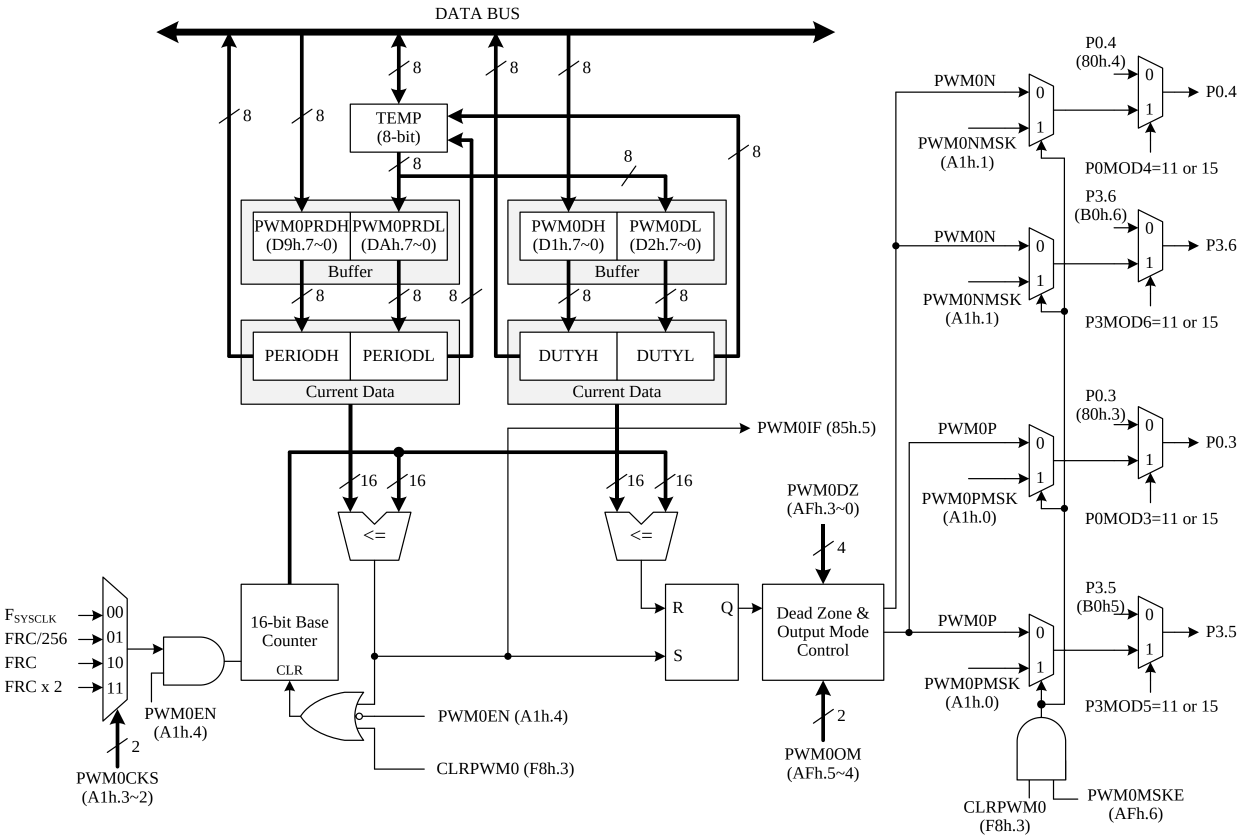
Only PWM0 has dead zone control, which can be divided into PWM0P/PWM0N outputs, and the rest of PWM1~PWM6 have non-overlapping control. The user can set the pin mode to Mode11 (PWM CMOS Push-Pull Output, no High-sink Current) or Mode15 (PWM CMOS Push-Pull Output with High-sink Current) to the corresponding I/O pins. (*see Chapter 7 for detail*)




**PWM interrupt Structure**

## 10.1 PWM0

PWM0EN serves as the control enabling bit of PWM0. If PWM0EN is cleared, PWM0 will be cleared and stopped, otherwise PWM0 is running. CLRPWM0 bit has the same functionality. When setting the CLRPWM0 bit, clear PWM0 and keep it on, otherwise PWM0 is running. The PWM0 structure is shown in the figure below.



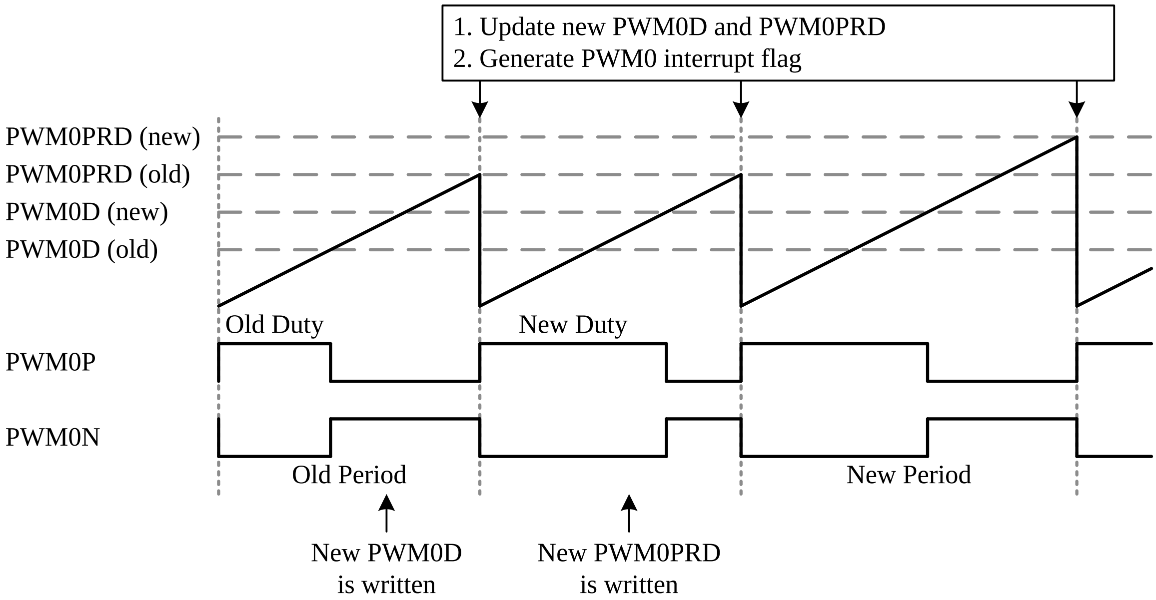

**PWM0 Structure**

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. At the end of current period, H/W will set the PWM0IF bit and generate an interrupt if a PWM0 interrupt is enabled.

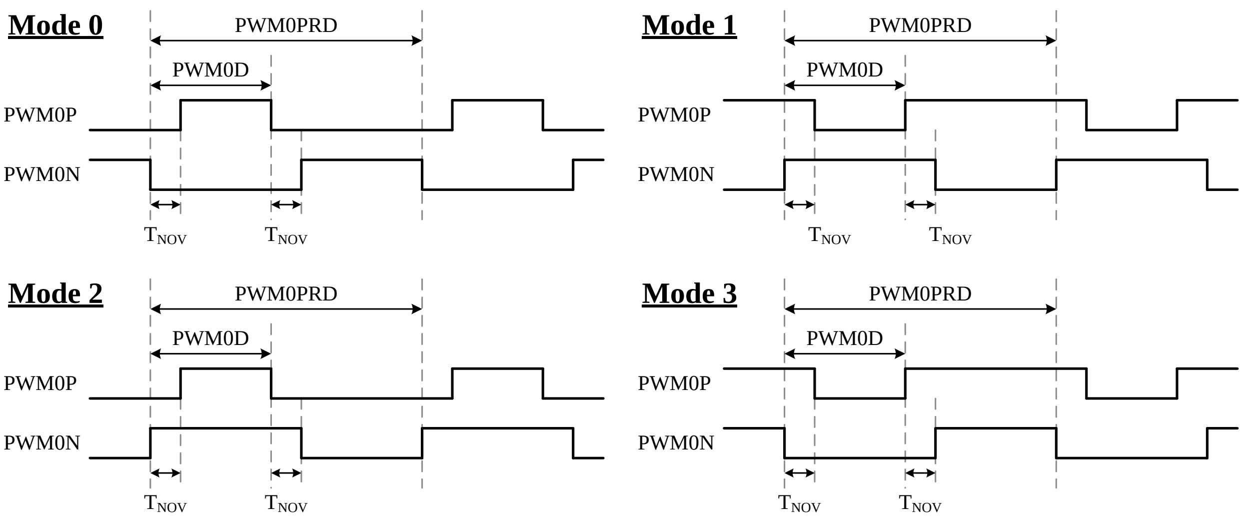
The PWM0 has two operation modes, normal mode and half-bridge mode. PWM0 output signal can be output via PWM0P and PWM0N with four different modes. These two outputs are non-overlapped with time interval $T_{NOV}$. Non-overlapping time interval is also named as dead zone or dead band. $T_{NOV}$ is determined 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.

### 10.1.1 Normal Mode

The normal mode PWM 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.



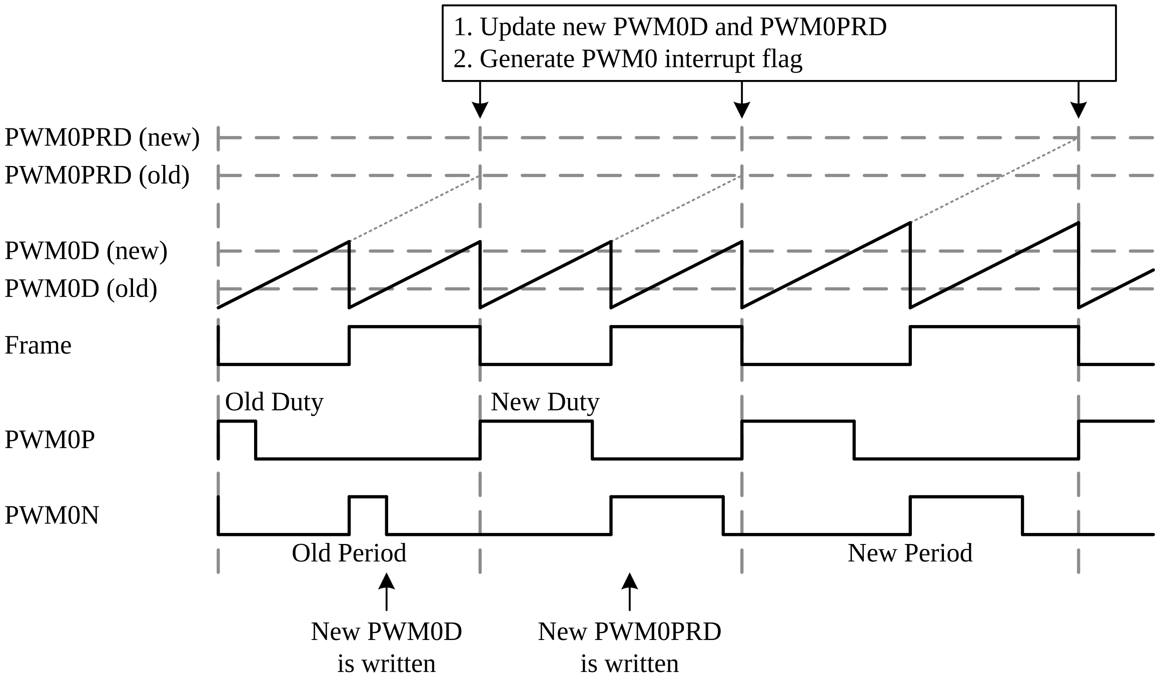

**PWM0 normal mode output waveform (PWM0OM=0, PWM0DZ=0)**



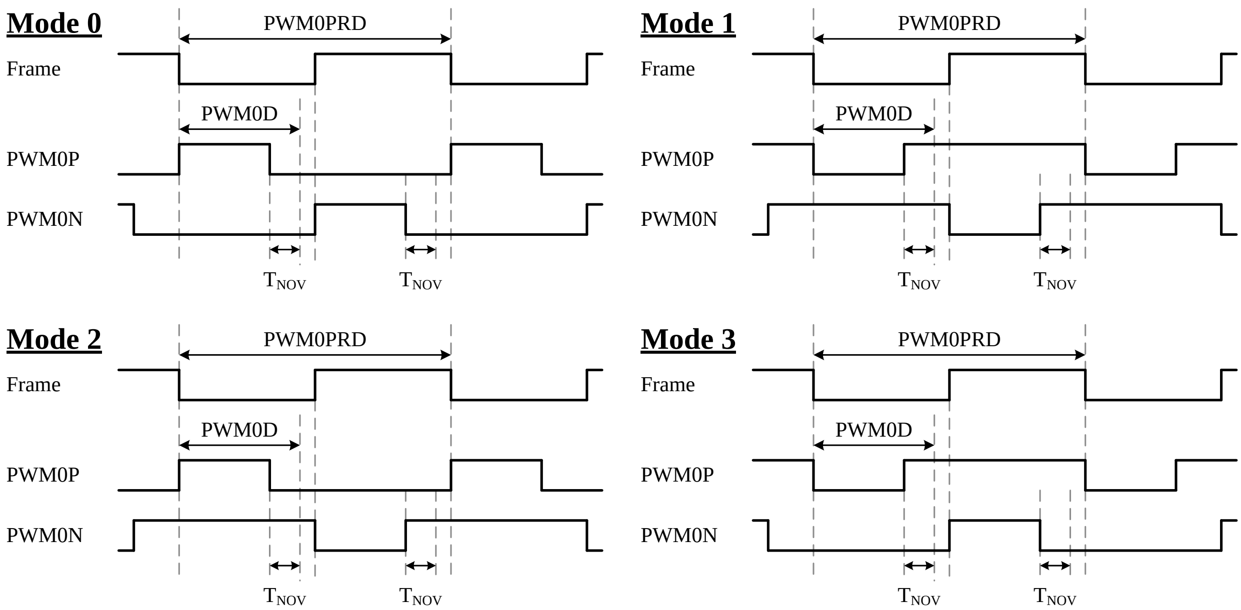

**PWM0 normal mode output modes**

### 10.1.2 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.



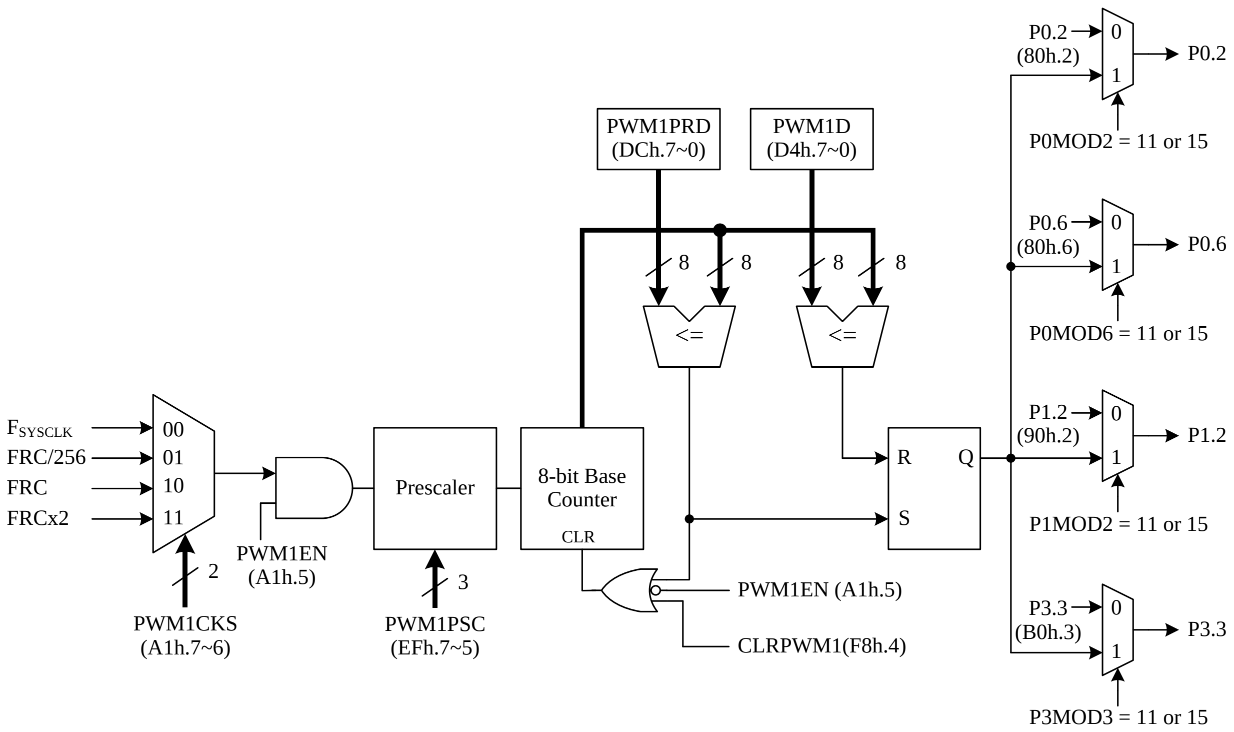

**PWM0 half-bridge mode output waveform (PWM0OM=0, PWM0DZ=0)**




**PWM0 half-bridge mode output modes**

### 10.2 PWM1~PWM6

The Chip has six 8-bit PWM modules PWM1~PWM6. PWM1~6 are sharing period, clock source and interrupt (PWM1IF). The following takes PWM1 as an example for description. The PWM can generate varies frequency waveform with 256 duty resolution on the basis of the PWM clock. The PWM clock has a prescaler that can generate a PWM clock that is divided by 1 to 128. The PWM clock can select double frequency (FRC x 2), FRC or $F_{SYSCLK}$ as its clock source.




**PWM1~6 Structure**

| 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.6 **PWM1IE:** PWM1~PWM6 interrupt enable
0: Disable PWM1~PWM6 interrupt
1: Enable PWM1~PWM6 interrupt

84h.5 **PWM0IE:** PWM0 interrupt enable
0: Disable PWM0 interrupt
1: Enable PWM0 interrupt

| SFR **85h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **INTFLG2** | – | PWM1IF | PWM0IF | – | – | – | – | – |
| R/W | – | R/W | R/W | – | – | – | – | – |
| Reset | – | 0 | 0 | – | – | – | – | – |

85h.6 **PWM1IF:** PWM1~PWM6 interrupt flag
Set by H/W at the end of PWM1 period, S/W writes BFh to INTFLG2 to clear this flag.

85h.5 **PWM0IF:** PWM0 interrupt enable
Set by H/W at the end of PWM0 period, S/W writes DFh to INTFLG2 to clear this flag.

| SFR **A9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **INTE1** | PWMIE | CMPIE | LVDIE | SPI2CE | ADIE | WGIE | 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/PWM1~PWM6 interrupt enable
0: Disable PWM0/PWM1~PWM6 interrupt
1: Enable PWM0/PWM1~PWM6 interrupt

| SFR **A1h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWMCON** | PWM1CKS | | PWM1EN | PWM0EN | PWM0CKS | | PWM0NMSK | PWM0PMSK |
| 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~PWM6 clock source select
00: $F_{SYSCLK}$
01: FRC/256
10: FRC
11: FRCx2 ($V_{CC}$>2.5V)

A1h.5 **PWM1EN:** PWM1~PWM6 enable
0: PWM1~PWM6 disable
1: PWM1~PWM6 enable

A1h.4 **PWM0EN:** PWM0 enable
0: PWM0 disable
1: PWM0 enable

A1h.3~2 **PWM0CKS:** PWM0 clock source select
00: $F_{SYSCLK}$
01: FRC/256
10: FRC
11: FRCx2 ($V_{CC}$>2.5V)

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, PWM0P will output this mask data.

| SFR **AFh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWMCON2** | PWM0MOD | PWM0MSKE | PWM0OM | | PWM0DZ | | | |
| R/W | R/W | R/W | R/W | | R/W | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

AFh.7 **PWM0MOD:** PWM0 mode select
0: Normal mode
1: Half-bridge mode

AFh.6 **PWM0MSKE:** PWM0 mask output enable
0: Disable
1: Enable, PWM0P/PWM0N output data by PWM0PMSK/PWM0NMSK while CLRPWM0=1

AFh.5~4 **PWM0OM:** PWM0 output mode select
00: Mode0
01: Mode1
10: Mode2
11: Mode3

AFh.3~0 **PWM0DZ:** PWM0 dead zone (Dead zone is prohibited in half-bridge mode)
0000: 0 x $T_{PWMCLK}$
0001: 1 x $T_{PWMCLK}$
...
1111: 15 x $T_{PWMCLK}$

| SFR **D1h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM0DH** | PWM0DH | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

| SFR **D2h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM0DL** | PWM0DL | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

| SFR **D4h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM1D** | PWM1D | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

D4h.7~0 **PWM1D:** PWM1 duty

| SFR **D6h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM2D** | PWM2D | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

D6h.7~0 **PWM2D:** PWM2 duty

| SFR D9h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM0PRDH** | PWM0PRDH | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

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

| SFR DAh | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM0PRDL** | PWM0PRDL | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

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

| SFR DCh | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM1PRD** | PWM1PRD | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

DCh.7~0 **PWM1PRD:** PWM1~PWM6 period

| SFR DEh | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM3D** | PWM3D | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

DEh.7~0 **PWM3D:** PWM3 duty

| SFR EAh | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM4D** | PWM4D | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

EAh.7~0 **PWM4D:** PWM4 duty

| SFR ECh | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM5D** | PWM5D | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

ECh.7~0 **PWM5D:** PWM5 duty

| SFR EEh | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PWM6D** | PWM6D | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

EEh.7~0 **PWM6D:** PWM6 duty

| SFR **EFh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX3** | PWM1PSC | | | PRGD | WARMTIME | DACVREFS | – | – |
| R/W | R/W | | | R/W | R/W | R/W | – | – |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | – | – |

EFh.7~5 **PWM1PSC:** PWM1~PWM6 clock pre-scaler select
0: div 1
1: div 2
2: div 4
3: div 8
4: div 16
5: div 32
6: div 64
7: div 128

| SFR **F8h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX1** | CLRWDT | CLRTM3 | – | ADSOC | CLRPWM0 | CLRPWM1 | WGEN | DPSEL |
| R/W | R/W | R/W | – | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | – | 0 | 1 | 1 | 0 | 0 |

F8h.3 **CLRPWM0:** PWM0 clear enable
0: PWM0 is running
1: PWM0 is cleared and held

F8h.2 **CLRPWM1:** PWM1/PWM2/PWM3/PWM4/PWM5/PWM6 clear enable
0: PWM1/PWM2/PWM3/PWM4/PWM5/PWM6 is running
1: PWM1/PWM2/PWM3/PWM4/PWM5/PWM6 is cleared and held

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

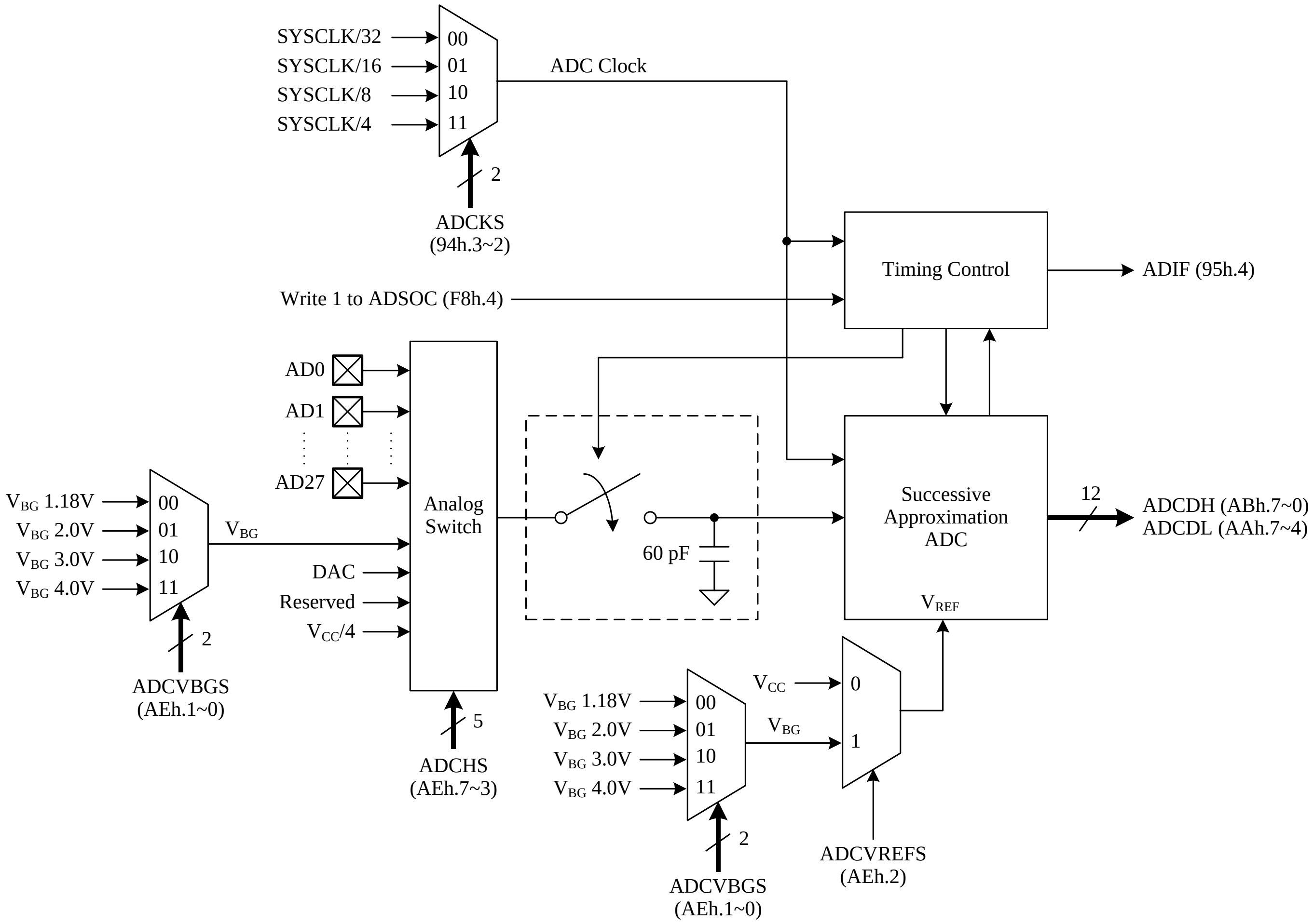
***Note:*** *also refer to Chapter 7 for more details on PWM pin mode settings.*

## 11. ADC

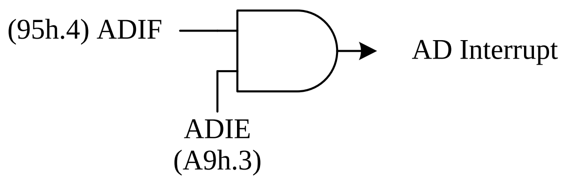
The Chip offers a 12-bit ADC consisting of a 28-channel analog input multiplexer, control register, clock generator, 12-bit successive approximation register, and output data register. Generally, ADC clock frequency is less than 4 MHz, user can refer to Electrical Characteristics Chapter for detail.

To use the ADC, set the ADCKS bits first to choose a proper ADC clock frequency. Then, user 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 set ADSOC bit. The analog input level must remain within the range from $V_{SS}$ to $V_{CC}$.

Using the ADCVREFS option, the ADC internal reference voltage source ($V_{REF}$) can be selected as $V_{CC}$ or $V_{BG}$.



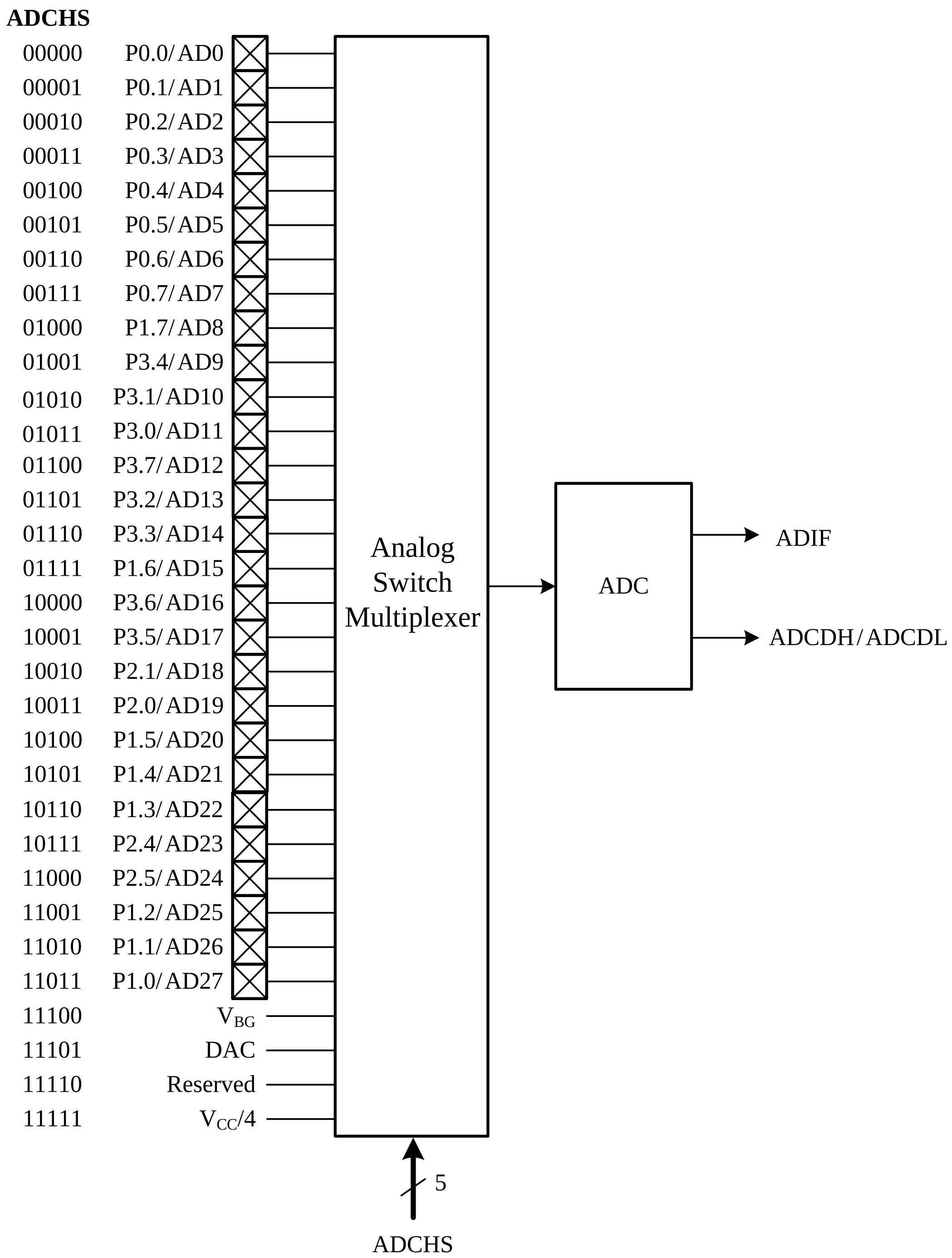

**ADC Structure**




**ADC Interrupt Structure**

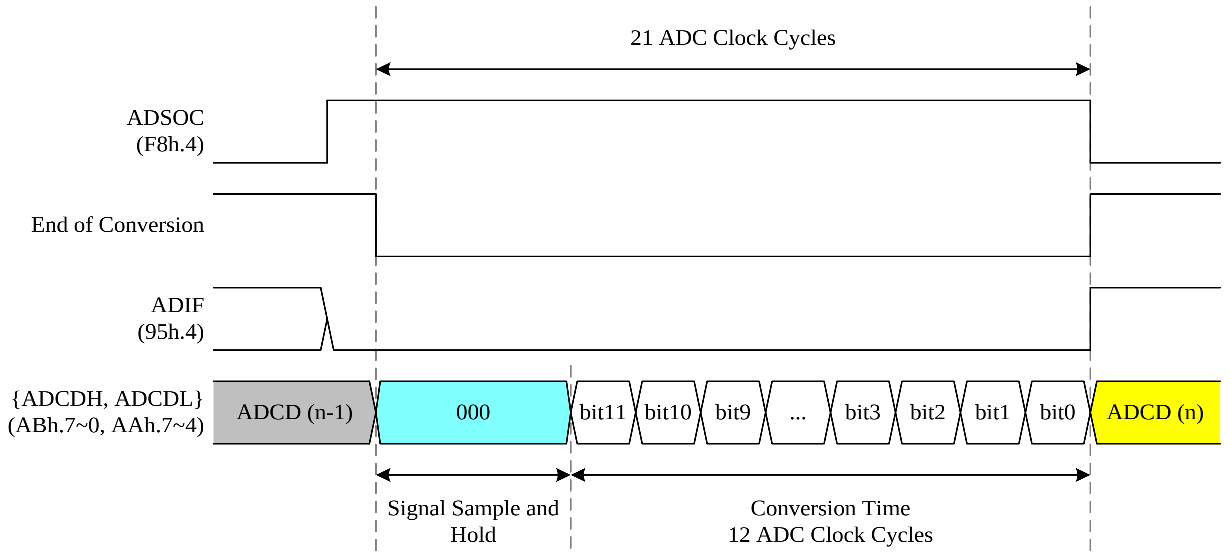
## 11.1 ADC Channels

The ADC channels are connected to the analog input pins via the analog switch multiplexer. The analog switch multiplexer is controlled by ADCHS register. The Chip offers up to 28 I/O input pins, designated AD0~AD27. In addition, there are 3 internal reference voltages ($V_{BG}$, DAC and $V_{CC}/4$). When ADCHS is set to 11100b, the analog input will connect to $V_{BG}$, when ADCHS is set to 11101b, the analog input will connect to DAC and when ADCHS is set to 11111b, the analog input will connect to $V_{CC}/4$.



### 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 94h | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **OPTION** | – | TM3CKS | WDTPSC | | ADCKS | | SXTGAIN | |
| 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 | CMPIF | – | ADIF | WGIF | – | PCIF | TF3 |
| R/W | R/W | R/W | – | R/W | R/W | – | R/W | R/W |
| Reset | 0 | 0 | – | 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.

| 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 |
|---|---|---|---|---|---|---|---|---|
| **ADCDH** | ADCDH | | | | | | | |
| R/W | R | | | | | | | |
| Reset | – | – | – | – | – | – | – | – |

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

| SFR **AEh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **ADCHSEL** | ADCHS | | | | | ADCVREFS | ADCVBGS | |
| R/W | R/W | | | | | R/W | R/W | |
| Reset | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

AEh.7~3 **ADCHS:** ADC channel select

| | |
|---|---|
| 00000: AD0 (P0.0) | 10000: AD16 (P3.6) |
| 00001: AD1 (P0.1) | 10001: AD17 (P3.5) |
| 00010: AD2 (P0.2) | 10010: AD18 (P2.1) |
| 00011: AD3 (P0.3) | 10011: AD19 (P2.0) |
| 00100: AD4 (P0.4) | 10100: AD20 (P1.5) |
| 00101: AD5 (P0.5) | 10101: AD21 (P1.4) |
| 00110: AD6 (P0.6) | 10110: AD22 (P1.3) |
| 00111: AD7 (P0.7) | 10111: AD23 (P2.4) |
| 01000: AD8 (P1.7) | 11000: AD24 (P2.5) |
| 01001: AD9 (P3.4) | 11001: AD25 (P1.2) |
| 01010: AD10 (P3.1) | 11010: AD26 (P1.1) |
| 01011: AD11 (P3.0) | 11011: AD27 (P1.0) |
| 01100: AD12 (P3.7) | 11100: $V_{BG}$ |
| 01101: AD13 (P3.2) | 11101: DAC |
| 01110: AD14 (P3.3) | 11110: Reserved |
| 01111: AD15 (P1.6) | 11111: $V_{CC}/4$ |

AEh.2 **ADCVREFS:** ADC reference voltage select
0: $V_{CC}$
1: $V_{BG}$

AEh.1~0 **ADCVBGS:** $V_{BG}$ voltage select (for ADC)
00: 1.18V
01: 2.0V
10: 3.0V
11: 4.0V

| SFR **F8h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX1** | CLRWDT | CLRTM3 | – | ADSOC | CLRPWM0 | CLRPWM1 | WGEN | DPSEL |
| R/W | R/W | R/W | – | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | – | 0 | 1 | 1 | 0 | 0 |

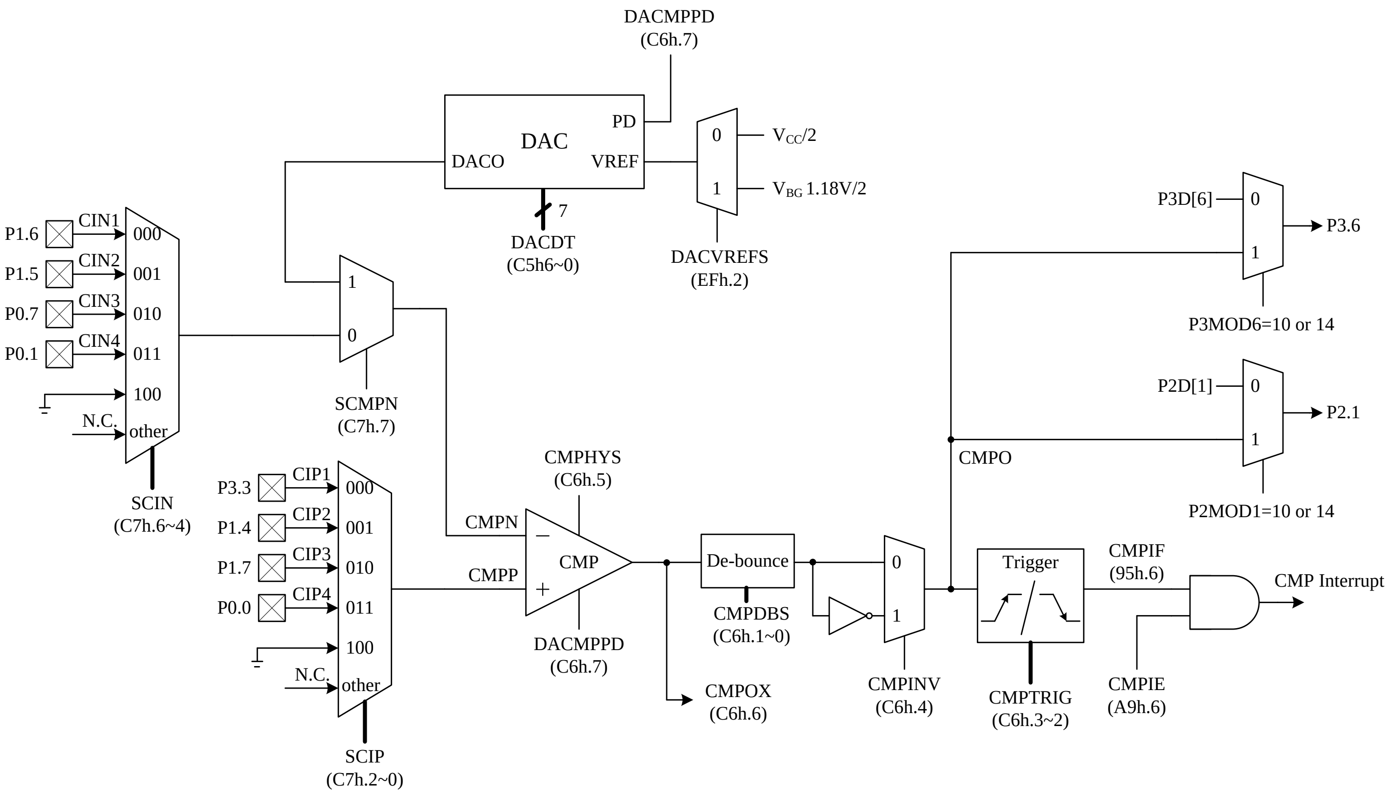
F8h.4 **ADSOC:** Start ADC conversion
Set the ADSOC bit to start ADC conversion, and the ADSOC bit will be cleared by H/W at the end of conversion. S/W can also write 0 to clear this flag.

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

***Note:*** *Also refer to Chapter 7 for more details on ADC pin input settings.*

## 12. Comparator (CMP)

The chip has a Comparator (CMP) that has a 7-bit DAC module built into the CMP with outputs that access the CMP's negative input port. The DAC's reference voltage can be selected as $V_{CC}/2$ or $V_{BG}$ 1.18V/2 by setting DACVREFS (EFh.2). By setting the DACDT (C5h.6~0), the appropriate voltage level can be selected to make the user application run normally, which will change the resistance to convert the voltage value. Setting DACMPPD=1 (C6h.7) will put the DAC and CMP into power-down mode. By configuring SCMPN (C7h.7), the negative port input source will be an external pin input or DAC output or a ground $V_{SS}$. The positive port input source can be either an external pin input or a ground $V_{SS}$, by defining SCIN (C7h.6~4) and SCIP (C7h.2~0). The registers determine the negative port and positive port external input source, respectively. Since the input module of a CMP consists of a PMOS, the input voltage range will be affected by 1/V of the PMOS. Therefore, the Max input voltage of the CMP will be ($V_{CC}$-0.5). Meanwhile, the hysteresis voltage of the comparator is about 10mV, which can be turned on or off by setting CMPHYS (C6h.5). Comparator raw output (CMPOX) can be read via CMPOX (C6h.6) bit. The chip provides a debounce module to debounce the CMPOX signal, and users can select the debounce time through CMPDBS (C6h.1~0). The debounced output signal can be selected by CMPINV (C6h.4) to select inverting or non-inverting to generate a CMPO signal. CMPO can be set to pin (P2.1) or P3MOD6=10 or 14 output to pin (P3.6) by setting P2MOD1=10 or 14 output to pin (P3.6). The CMPO is also the trigger source for the interrupt trigger module and is used to generate the interrupt flag CMPIF (95h.6). The trigger mode is selected by CMPTRIG (C6h.3~2). When the comparator is powered off, an interrupt flag is still generated. Therefore, it is necessary to clear the interrupt flag after opening the CMP module every time to prevent the use of virtual flags.




**Comparator Structure**

| SFR **A9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **INTE1** | PWMIE | CMPIE | LVDIE | SPI2CE | ADIE | WGIE | 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.6 **CMPIE:** CMP interrupt enable
0: Disable CMP interrupt
1: Enable CMP interrupt

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

95h.6 **CMPIF:** CMP interrupt flag
Set by H/W while CMPO match trigger condition. It is cleared automatically when the program performs the interrupt service routine. S/W writes BFh to INTFLG to clear this flag.

| SFR **C6h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| CMPCON | DACMPPD | CMPOX | CMPHYS | CMPINV | CMPTRIG | | CMPDBS | |
| R/W | R/W | R | R/W | R/W | R/W | | R/W | |
| Reset | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

C6h.7 **DACMPPD:** Comparator & DAC power down control
0: enable Comparator & DAC
1: disable Comparator & DAC

C6h.6 **CMPOX:** Comparator original output (CMPOX) status
0: $V_{CMPP} < V_{CMPN}$
1: $V_{CMPP} > V_{CMPN}$ or CMPPD =1

C6h.5 **CMPHYS:** Comparator Hysteresis Control
0: disable
1: enable (the hysteresis voltage is about 10mV)

C6h.4 **CMPINV:** Comparator de-bounce output invert select
0: no invert
1: invert

C6h.3~2 **CMPTRIG:** Comparator interrupt trigger mode
00: Rising edge
01: Falling edge
10: Both edge
11: High level

C6h.1~0 **CMPDBS:** Comparator original output (CMPOX) de-bounce time
00: none
01: 4 $F_{SYSCLK}$
10: 8 $F_{SYSCLK}$
11: 16 $F_{SYSCLK}$

| SFR **C7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **CMPPNS** | SCMPN | SCIN | | | – | SCIP | | |
| R/W | R/W | R/W | | | – | R/W | | |
| Reset | 1 | 1 | 1 | 1 | – | 1 | 1 | 1 |

C7h.7 **SCMPN:** Comparator CMPN source select
0: Comparator CMPN source is external pin (CINx)
1: Comparator CMPN source is DAC output

C7h.6~4 **SCIN:** Comparator CMPN external pin select
000: Comparator CMPN external input is CIN1 (P1.6)
001: Comparator CMPN external input is CIN2 (P1.5)
010: Comparator CMPN external input is CIN3 (P0.7)
011: Comparator CMPN external input is CIN4 (P0.1)
100: Comparator CMPN external input is CIN5 ($V_{SS}$)
other: No connection

C7h.2~0 **SCIP:** Comparator CMPP external pin select
000: Comparator CMPP external input is CIP1 (P3.3)
001: Comparator CMPP external input is CIP2 (P1.4)
010: Comparator CMPP external input is CIP3 (P1.7)
011: Comparator CMPP external input is CIP4 (P0.0)
100: Comparator CMPP external input is CIP5 ($V_{SS}$)
other: No connection

| SFR **EFh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX3** | PWM1PSC | | | PRGD | WARMTIME | DACVREFS | – | – |
| R/W | R/W | | | R/W | R/W | R/W | – | – |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | – | – |

EFh.2 **DACVREFS:** DAC reference voltage select
0: $V_{CC}/2$
1: $V_{BG}$ 1.18V/2

| SFR **C5h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **DACDT** | – | DACDT | | | | | | |
| R/W | – | R/W | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

C5h.6~0 **DACDT:** select DAC output voltage (Reference source can be selected as $V_{CC}/2$ or $V_{BG}$1.18V/2)
000_0000: 0/128 *Reference source
000_0001: 1/128 *Reference source
000_0010: 2/128 *Reference source
…
111_1101:125/128 *Reference source
111_1110:126/128 *Reference source
111_1111:127/128 *Reference source

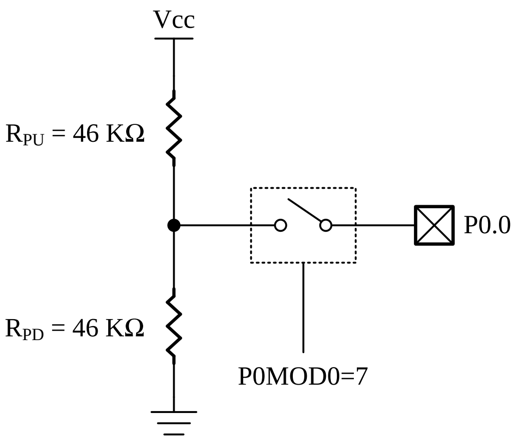
***Note: The highest bit must force 0***

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

***Note:*** *Also refer to Chapter 7 for more details on CMP pin input settings.*

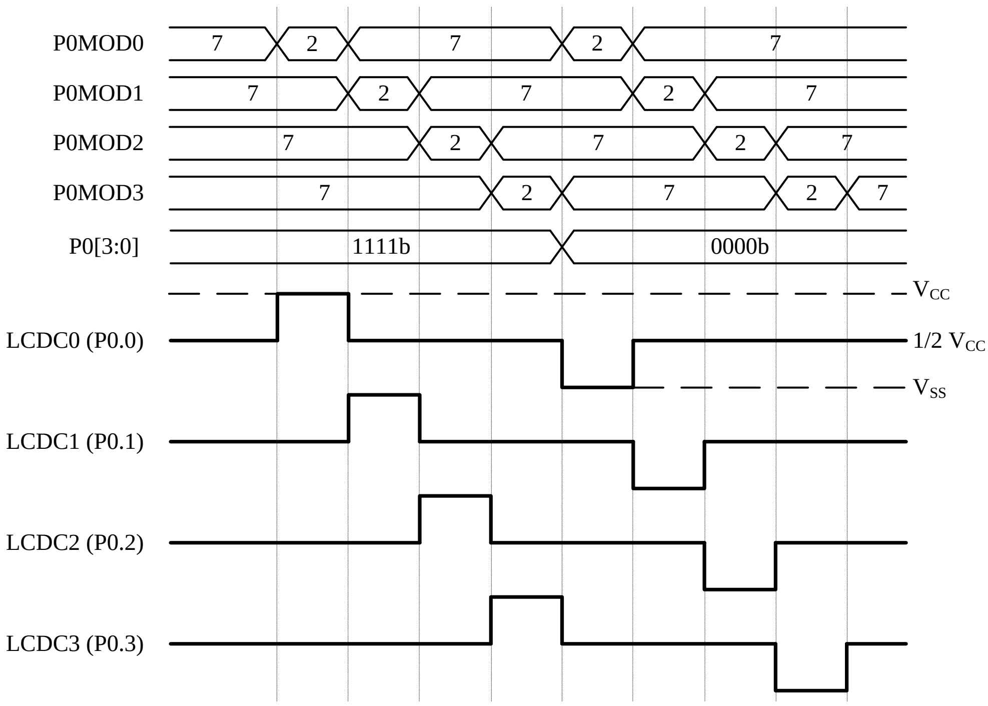
## 13. S/W Controller LCD Driver

The chip supports LCD drivers controlled by S/W. P0.0~P0.7 can be set as Common pins (COM) LCDC0~LCDC7, and the remaining pins are used as Segment pins (SEG). User can flexibly adjust the Common pins and Segment pins. It is capable of driving the LCD panel with 176 dots (Max.) by with 8 common pins and 22 segmented pins . The LCD driver of the chip can only drive 1/2 bias, which is achieved by setting the corresponding pin mode to Mode7 (see Table 7.1 in Chapter 7). Refer to the following figures.



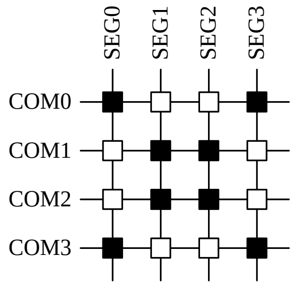

**LCDC0 bias circuit**

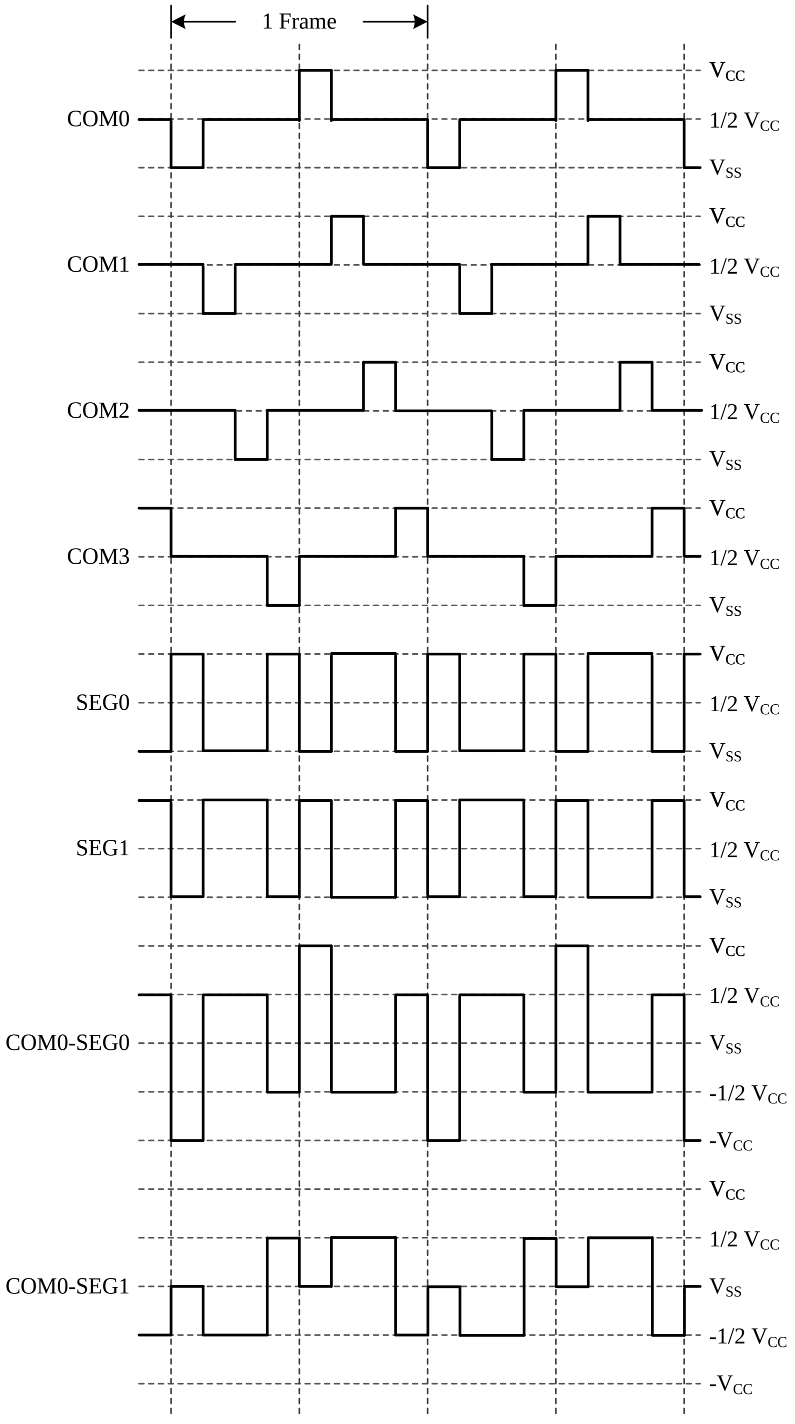
The frequency of any repeating waveform output on the COM pin can be used to represent the LCD frame rate. The figure below shows an LCD frame.




**S/W Controlled LCDC0~ LCDC3 Scanning**

### 1/4 Duty, 1/2 Bias Output Waveform



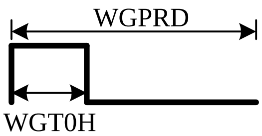



## 14. Communication Format Waveform Generator (WG)

The input information of the Waveform Generator is stored in the SFR WGBUF. The waveform generator will output its serial output to P0 according to the following encoding rules5, P1.2, P3.4 or P3.7. (See Table 7.1 in Chapter 7).

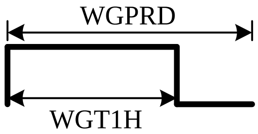
If the data value is 0, the output waveform will be encoded as shown in the figure below, and the duty cycle and period can be adjusted by WGT0H and WGPRD.

**code 0**

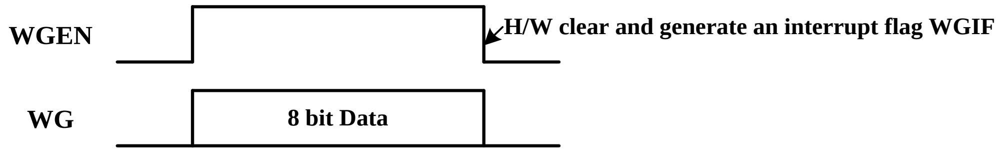



If the data value is 1, the output waveform will be encoded as shown in the figure below, and the duty cycle and period can be adjusted by WGT1H and WGPRD.

**code 1**




When the user sets WGEN = 1, the hardware will start sending the data code, and after sending it, the hardware will automatically clear the WGEN bit and generate an interrupt flag WGIF.




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

95h.3 **WGIF:** Wave Generator interrupt flag
Set by H/W at the end of Wave generate. S/W writes F7h 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 **A9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **INTE1** | PWMIE | CMPIE | LVDIE | SPI2CE | ADIE | WGIE | 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 **WGIE:** WG interrupt enable
0: Disable WG interrupt
1: Enable WG interrupt

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

***Note:*** *also refer to Chapter 7 for more details on WG pin mode settings.*

| SFR **B6h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **WGCON** | WGPRD | | | WGT1H | | | WGT0H | |
| R/W | R/W | | | R/W | | | R/W | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

B6h.7~5 **WGPRD:** WG period of each bit select ($T_{FRC}$ = 60.28 ns)
- 0: 15*$T_{FRC}$ = 904 ns
- 1: 16*$T_{FRC}$ = 965 ns
- 2: 17*$T_{FRC}$ = 1025 ns
- 3: 18*$T_{FRC}$ = 1085 ns
- 4: 19*$T_{FRC}$ = 1145 ns
- 5: 20*$T_{FRC}$ = 1206 ns
- 6: 21*$T_{FRC}$ = 1266 ns
- 7: 22*$T_{FRC}$ = 1326 ns

B6h.4~2 **WGT1H:** WG code 1 high level time select ($T_{FRC}$ = 60.28 ns)
- 0: 10*$T_{FRC}$ = 603 ns
- 1: 11*$T_{FRC}$ = 663 ns
- 2: 12*$T_{FRC}$ = 723 ns
- 3: 13*$T_{FRC}$ = 784 ns
- 4: 14*$T_{FRC}$ = 844 ns
- 5: 15*$T_{FRC}$ = 904 ns
- 6: 16*$T_{FRC}$ = 965 ns
- 7: 17*$T_{FRC}$ = 1025 ns

B6h.1~0 **WGT0H:** WG code 0 high level time select ($T_{FRC}$ = 60.28 ns)
- 0: 4*$T_{FRC}$ = 241 ns
- 1: 5*$T_{FRC}$ = 301 ns
- 2: 6*$T_{FRC}$ = 362 ns
- 3: 7*$T_{FRC}$ = 422 ns

| SFR **B7h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **WGBUF** | WGBUF | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

B7h.7~0 **WGBUF:** Wave Generator code buffer

| SFR **F8h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **AUX1** | CLRWDT | CLRTM3 | – | ADSOC | CLRPWM0 | CLRPWM1 | WGEN | DPSEL |
| R/W | R/W | R/W | – | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | – | 0 | 1 | 1 | 0 | 0 |

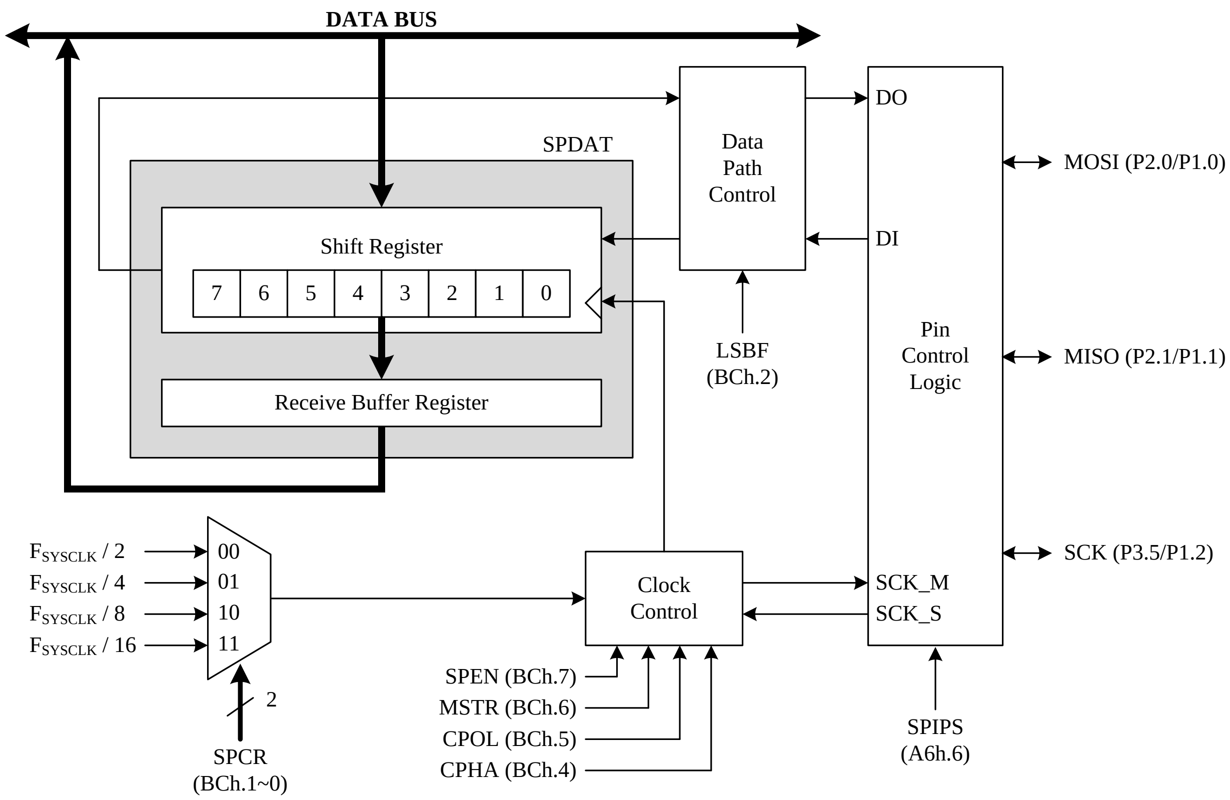
F8h.1 **WGEN:** Wave Generator enable, H/W will automatically clear WGEN after one cycle
- 0: WG disable
- 1: WG enable

## 15. Serial Peripheral Interface (SPI)

The Serial Peripheral Interface (SPI) module is capable of full-duplex, synchronous, serial communication between the MCU and peripheral devices. The peripheral devices can be other MCUs, A/D converter, sensors, or flash memory, etc. The SPI runs at a clock rate up to the system clock divided by two. Firmware can read the status flags, or the operation can be interrupt driven. Following figure shows the SPI system block diagram.

The features of the SPI module include:

- Master or Slave mode operation
- 3-wire mode operation
- Full-duplex operation
- Programmable transmit bit rate
- Single buffer receive
- Serial clock phase and polarity options
- MSB-first or LSB-first shifting selectable




| SPI Function Pin | Pin Mode | SFR data |
|---|---|---|
| Master Mode, MISO | **Mode1** | 1 |
| Master Mode, SCK, MOSI | **Mode2** | X |
| Slave Mode, MISO | **Mode2** | X |
| Slave Mode, SCK, MOSI | **Mode1** | 1 |

**Pin Mode Setting for SPI**

The three signals used by SPI are described below. The MOSI (P2.0/P1.0) signal is an output from a Master Device and an input to Slave Devices. The signal is an output when SPI is operating in Master mode and an input when SPI is operating in Slave mode. The MISO (P2.1/P1.1) signal is an output from a Slave Device and an input to a Master Device. The signal is an input when SPI is operating in Master mode and an output when SPI is operating in Slave mode. Data is transferred most-significant bit (MSB) or least-significant bit (LSB) first by setting the LSBF bit. The SCK (P3.5/P1.2) signal is an output from a Master Device and an input to Slave Devices. It is used to synchronize the data on the MOSI and MISO lines of Master and Slave. SPI generates the signal with four programmable clock rates in Master mode.

### Master Mode

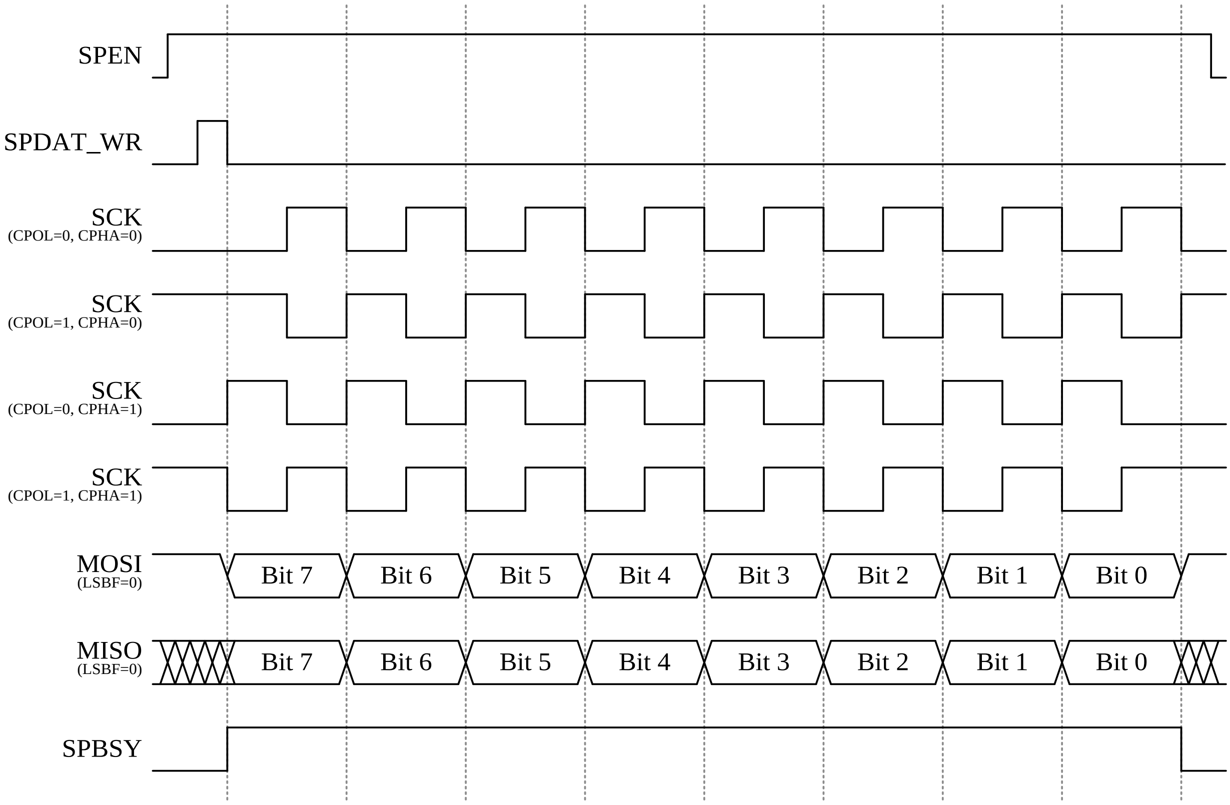
The SPI operates in Master mode by setting the MSTR bit in the SPCON. To start transmit, writing a data to the SPDAT. If SPBSY=0, the data will be transferred to the shift register and starts shift out on the MOSI line. The data of the Slave shift in from the MISO line at the same time. When the SPIF bit becomes set at the end of transfer, the receive data is written to receiver buffer and the RCVBF bit in the SPSTA is set. To prevent an overrun condition, software must read the SPDAT before next byte enters the shift register. The SPBSY bit will be set when writing a data to SPDAT to start transmit, and be cleared at the end of the eighth SCK period in Master mode.

### Slave Mode

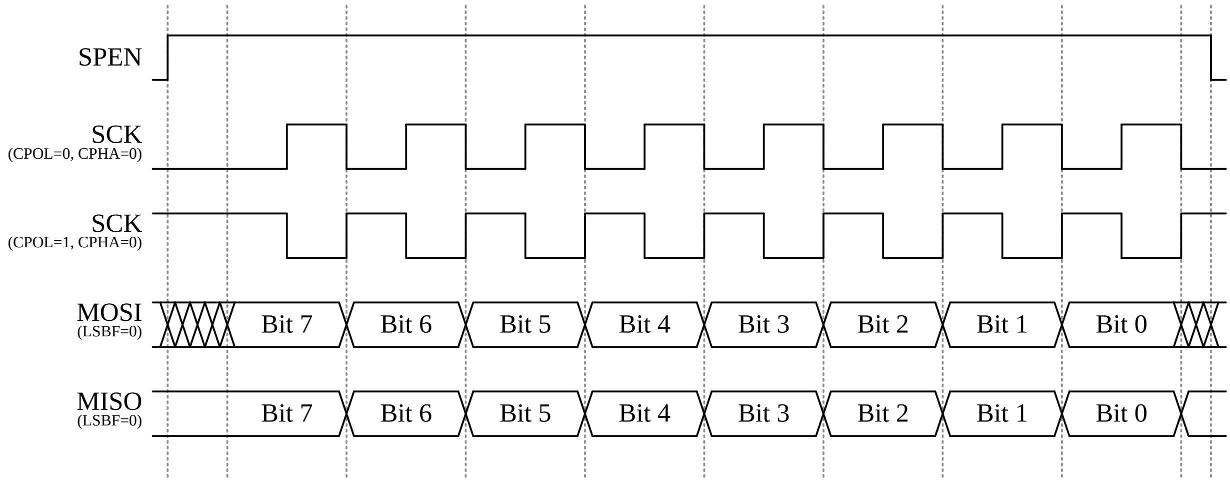
The SPI operates in Slave mode by clearing the MSTR bit in the SPCON. The transmission will start when the SPEN bit in the SPCON is set. The data from a Master will shift into the shift register through the MOSI line, and shift out from the shift register on the MISO line. When a byte enters the shift register, the data will be transferred to receiver buffer if RCVBF=0. If RCVBF=1, the newer received data will not be transferred to receiver buffer and the RCVOVF bit is set. After a byte enters the shift register, the SPIF and RCVBF bits are set. To prevent an overrun condition, software must read the SPDAT or write 0 to RCVBF before next byte enters the shift register. **The maximum SCK frequency allowed in Slave mode is $F_{SYSCLK}$/4**.

### Serial Clock

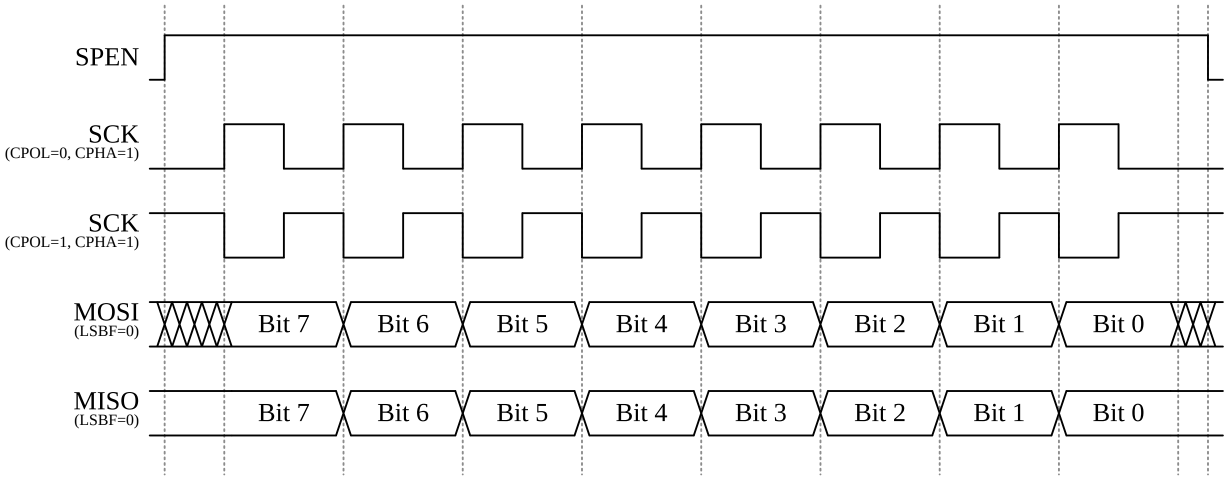
The SPI has four clock types by setting the CPOL and CPHA bits in the SPCON register. The CPOL bit defines the level of the SCK in SPI idle state. The level of the SCK in idle state is low when CPOL=0, and is high when CPOL=1. The CPHA bit defines the edges used to sample and shift data. The SPI sample data on the first edge of SCK period and shift data on the second edge of SCK period when CPHA=0. The SPI sample data on the second edge of SCK period and shift data on first edge of SCK period when CPHA=1. Figures below show the detail timing in Master and Slave modes. Both Master and Slave devices must be configured to use the same clock type before the SPEN bit is set. The SPCR controls the Master mode serial clock frequency. This register is ignored when operating in Slave mode. The SPI clock can select System clock divided by 2, 4, 8, or 16 in Master mode.




**Master Mode Timing**




**Slave Mode Timing (CPHA=0)**




**Slave Mode Timing (CPHA=1)**

In both Master and Slave modes, the SPIF interrupt flag is set by H/W at the end of a data transfer. If write data to SPDAT when SPBSY=1, the WCOL interrupt flag will be set by H/W. When this occurs, the data write to SPDAT will be ignored, and shift register will not be written.

| SFR **BCh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **SPCON** | SPEN | MSTR | CPOL | CPHA | – | LSBF | SPCR | |
| R/W | R/W | R/W | R/W | R/W | – | R/W | R/W | |
| Reset | 0 | 0 | 0 | 0 | – | 0 | 0 | 0 |

BCh.7 **SPEN:** SPI enable
- 0: SPI disable
- 1: SPI enable

BCh.6 **MSTR:** Master mode enable
- 0: Slave mode
- 1: Master mode

BCh.5 **CPOL:** SPI clock polarity
- 0: SCK is low in idle state
- 1: SCK is high in idle state

BCh.4 **CPHA:** SPI clock phase
- 0: Data sample on first edge of SCK period
- 1: Data sample on second edge of SCK period

BCh.2 **LSBF:** LSB first
- 0: MSB first
- 1: LSB first

BCh.1~0 **SPCR:** SPI clock rate
- 00: $F_{SYSCLK}/2$
- 01: $F_{SYSCLK}/4$
- 10: $F_{SYSCLK}/8$
- 11: $F_{SYSCLK}/16$

| SFR **BDh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **SPSTA** | SPIF | WCOL | – | RCVOVF | RCVBF | SPBSY | – | – |
| R/W | R/W | R/W | – | R/W | R/W | R | – | – |
| Reset | 0 | 0 | – | 0 | 0 | 0 | – | – |

BDh.7 **SPIF:** SPI interrupt flag

This is set by H/W at the end of a data transfer. Cleared by H/W when an interrupt is vectored into. Writing 0 to this bit will clear this flag.

BDh.6 **WCOL:** Write collision interrupt flag

Set by H/W if write data to SPDAT when SPBSY is set. Write 0 to this bit or rewrite data to SPDAT when SPBSY is cleared will clear this flag.

BDh.4 **RCVOVF:** Received buffer overrun flag

Set by H/W at the end of a data transfer and RCVBF is set. Write 0 to this bit or read SPDAT register will clear this flag.

BDh.3 **RCVBF:** Receive buffer full flag

Set by H/W at the end of a data transfer. Write 0 to this bit or read SPDAT register will clear this flag.

BDh.2 **SPBSY:** SPI busy flag

Set by H/W when a SPI transfer is in progress.

| SFR **BEh** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **SPDAT** | SPDAT | | | | | | | |
| R/W | R/W | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

BEh.7~0 **SPDAT:** SPI transmit and receive data

The SPDAT register is used to transmit and receive data. Writing data to SPDAT place the data into shift register and start a transfer when in master mode. Reading SPDAT returns the contents of the receive buffer.

| SFR **A9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **INTE1** | PWMIE | CMPIE | LVDIE | SPI2CE | ADIE | WGIE | 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.4 **SPI2CE:** SPI/Master I²C interrupt enable

0: Disable SPI/Master I²C interrupt

1: Enable SPI/Master I²C interrupt

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

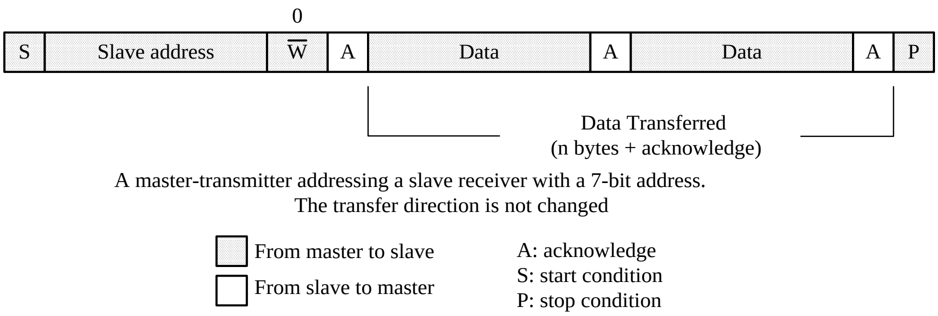
***Note:*** *also refer to Chapter 7 for more details on SPI pin mode settings.*

## 16. Master I²C Interface

### Master I²C 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.

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 $I^2C$ protocol. SCL clock can be adjusted via MICR




### Master I²C Transmit flow:

(1) Write slave address and direction bit to MIDAT

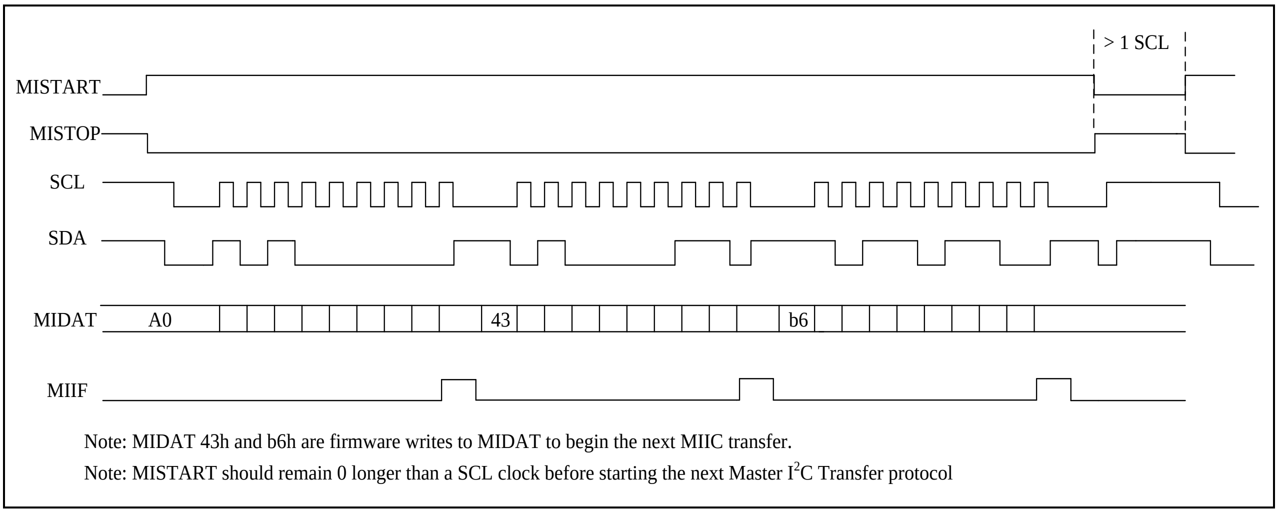
(2) Clear MISTOP and set MISTART to start I²C transmission

(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 I²C transfer



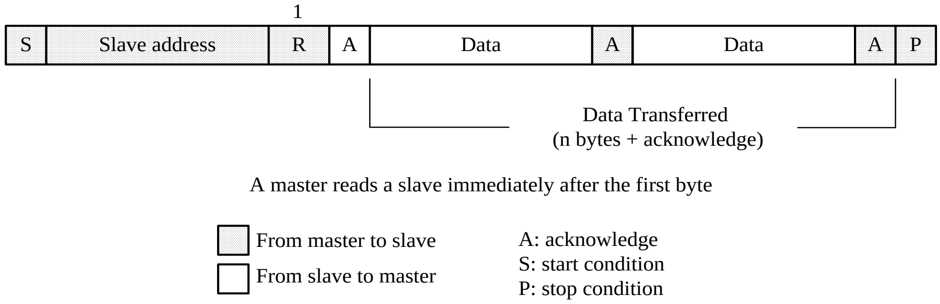

**Master Transmit Timing**

***Note:*** *MISTART should remain 0 longer than a SCL period before starting the next Master $I^2C$ protocol.*

**Master I²C 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 I²C protocol. SCL clock can be adjusted via MICR.




A master reads a slave immediately after the first byte

**Master I²C Receive flow:**

(1) Write slave address and direction bit to MIDAT

(2) Clear MISTOP and set MISTART to start I²C transmission

(3) Wait until MIIF convert to 1 (interrupt will be issued according to the user's request)

(4) Clear MIIF

(5) Read data from MIDAT to start first receive data

(The first reading of MIDAT does not represent the data returned by the slave)

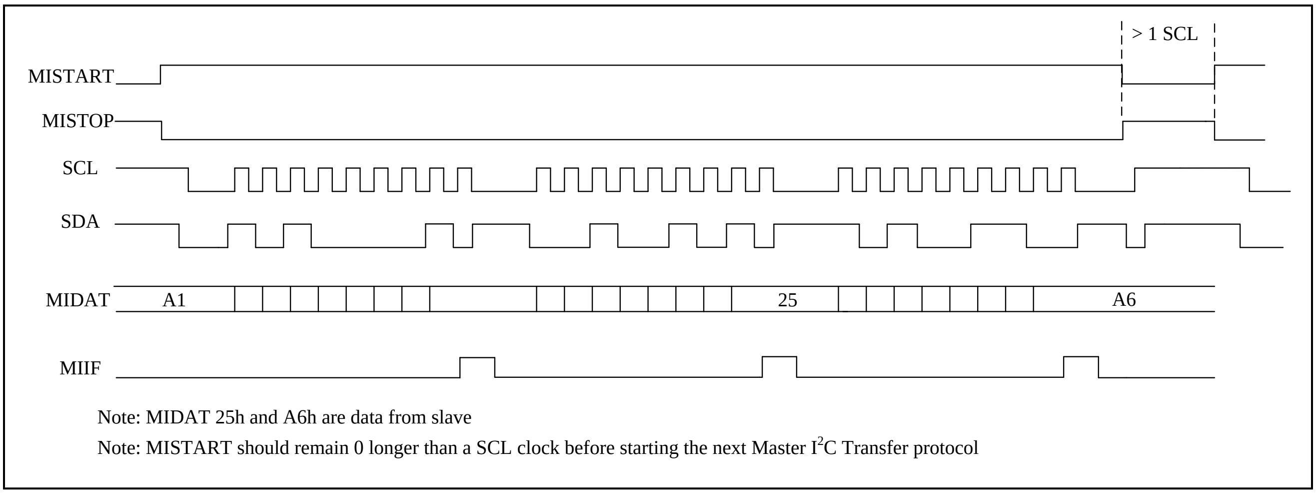
(6) Wait until MIIF convert to 1

(7) Clear MIIF

(8) Read slave data from MIDAT and receive next data

(9) Loop (6) ~(8)

(10) Set MISTOP to stop the I²C transfer




**Master Receive Timing**

| $I^2C$ Function Pin | PINMODxx | Px.n SFR data | Pin State |
|---|---|---|---|
| $I^2C$ Master SCL | **Mode0** | X | $I^2C$ Clock Output (Open Drain Output, Pull-up) |
| | **Mode2** | X | $I^2C$ Clock Output (CMOS Push-Pull) |
| $I^2C$ Master SDA | **Mode0** | 1 | $I^2C$ DATA (Pull-up) |

**Pin Mode Setting for Master $I^2C$**

| SFR **E1h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **MICON** | 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 $I^2C$ enable
0: disable
1: enable

E1h.6 **MIACKO**: When Master $I^2C$ receive data, send acknowledge to $I^2C$ Bus
0: ACK to slave device
1: NACK to slave device

E1h.5 **MIIF**: Master $I^2C$ Interrupt flag
0: write 0 to clear it
1: Master $I^2C$ transfer one byte complete

E1h.4 **MIACKI**: When Master $I^2C$ transfer, acknowledgement form $I^2C$ bus (read only)
0: ACK received
1: NACK received

E1h.3 **MISTART**: Master $I^2C$ Start bit
1: start $I^2C$ bus transfer

E1h.2 **MISTOP**: Master $I^2C$ Stop bit
1: send STOP signal to stop $I^2C$ bus

E1h.1~0 **MICR:** Master $I^2C$ (SCL) clock frequency selection
00: $F_{SYSCLK}/4$ (ex. If $F_{SYSCLK}$=16MHz, $I^2C$ clock is 4 MHz)
01: $F_{SYSCLK}/16$ (ex. If $F_{SYSCLK}$=16MHz, $I^2C$ clock is 1 MHz)
10: $F_{SYSCLK}/64$ (ex. If $F_{SYSCLK}$=16MHz, $I^2C$ clock is 250 KHz)
11: $F_{SYSCLK}/256$ (ex. If $F_{SYSCLK}$=16MHz, $I^2C$ clock is 62.5 KHz)

| SFR **E2h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **MIDAT** | 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 $I^2C$ data shift register
(W):After Start and before Stop condition, write this register will resume transmission to $I^2C$ bus
(R): After Start and before Stop condition, read this register will resume receiving from $I^2C$ bus

| SFR **A9h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **INTE1** | PWMIE | CMPIE | LVDIE | SPI2CE | ADIE | WGIE | 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.4 **SPI2CE:** SPI/Master $I^2C$ interrupt enable
0: Disable SPI/Master $I^2C$ interrupt
1: Enable SPI/Master $I^2C$ interrupt

| SFR **A6h** | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| **PINMOD** | – | SPIPS | UARTRXS | | MSDAPS | | MSCLPS | |
| R/W | – | R/W | R/W | | R/W | | R/W | |
| Reset | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

A6h.3~2 **MSDAPS:** Master $I^2C$ SDA pin select
00: P3.5
01: P1.6
10: P2.1
11: P2.5

A6h.1~0 **MSCLPS:** Master $I^2C$ SCL pin select
00: P1.3
01: P0.2
10: P2.0
11: P2.4

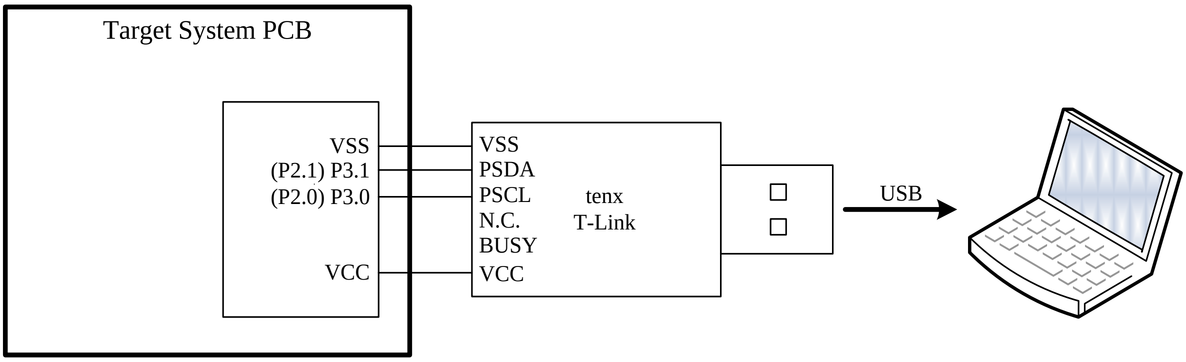
***Note:*** *also refer to Chapter 6 for more information about $I^2C$ Interrupt enable and priority.*

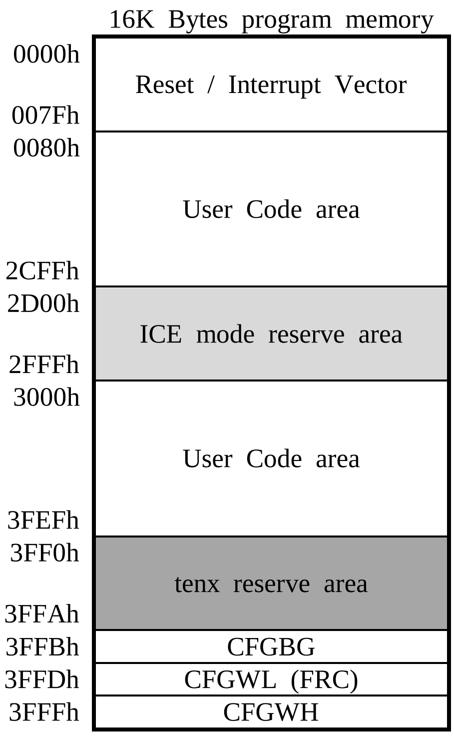
***Note:*** *also refer to Chapter 7 for more details on $I^2C$ pin mode settings.*

## 17. 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 (P3MOD0 = 0/1 and P3MOD1=0/1).
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 T-Link communication pin's function cannot be emulated.
5. The P3.0 and P3.1 pin's can be replaced by P2.0 and P2.1. (Only emulation can be replaced, mass production writer only supports P3.0/P3.1)
6. The VDD level is controlled by T-Link module.




**TM52F5864**

# 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 | **P0** | P0.7 | P0.6 | P0.5 | P0.4 | P0.3 | P0.2 | P0.1 | P0.0 |
| 81h | 0000-0111 | **SP** | SP | | | | | | | |
| 82h | 0000-0000 | **DPL** | DPL | | | | | | | |
| 83h | 0000-0000 | **DPH** | DPH | | | | | | | |
| 84h | x00x-xxxx | **INTE2** | — | PWM1IE | PWM0IE | — | — | — | — | — |
| 85h | x00x-xxxx | **INTFLG2** | — | PWM1IF | PWM0IF | — | — | — | — | — |
| 86h | xxxx-xxxx | – | – | | | | | | | |
| 87h | xxxx-0000 | **PCON** | – | – | – | – | GF1 | GF0 | PD | IDL |
| 88h | 0000-0000 | **TCON** | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |
| 89h | 0000-0000 | **TMOD** | GATE1 | CT1N | TMOD1 | | GATE0 | CT0N | TMOD0 | |
| 8Ah | 0000-0000 | **TL0** | TL0 | | | | | | | |
| 8Bh | 0000-0000 | **TL1** | TL1 | | | | | | | |
| 8Ch | 0000-0000 | **TH0** | TH0 | | | | | | | |
| 8Dh | 0000-0000 | **TH1** | TH1 | | | | | | | |
| 8Eh | xxxx-xxxx | – | – | | | | | | | |
| 8Fh | xxxx-xxxx | – | – | | | | | | | |
| 90h | 1111-1111 | **P1** | P1.7 | P1.6 | P1.5 | P1.4 | P1.3 | P1.2 | P1.1 | P1.0 |
| 91h | 0001-0001 | **P0MOD10** | P0MOD1 | | | | P0MOD0 | | | |
| 92h | 0001-0001 | **P0MOD32** | P0MOD3 | | | | P0MOD2 | | | |
| 93h | 0001-0001 | **P0MOD54** | P0MOD5 | | | | P0MOD4 | | | |
| 94h | x000-0000 | **OPTION** | – | TM3CKS | WDTPSC | | ADCKS | | SXTGAIN | |
| 95h | 00x0-0x00 | **INTFLG** | LVDIF | CMPIF | – | ADIF | WGIF | – | PCIF | TF3 |
| 96h | 0001-0001 | **P0MOD76** | P0MOD7 | | | | P0MOD6 | | | |
| 97h | xxxx-xx0x | **SWCMD** | SWRST / WDTO / IAPALL | | | | | | | |
| 98h | 0100-0000 | **SCON** | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
| 99h | xxxx-xxxx | **SBUF** | SBUF_RX / SBUF_TX | | | | | | | |
| 9Ah | 0001-0001 | **P1MOD10** | P1MOD1 | | | | P1MOD0 | | | |
| 9Bh | 0001-0001 | **P1MOD32** | P1MOD3 | | | | P1MOD2 | | | |
| 9Ch | 0001-0001 | **P1MOD54** | P1MOD5 | | | | P1MOD4 | | | |
| 9Dh | 0001-0001 | **P1MOD76** | P1MOD7 | | | | P1MOD6 | | | |
| 9Eh | 0001-0001 | **P2MOD10** | P2MOD1 | | | | P2MOD0 | | | |
| 9Fh | 0001-0001 | **P2MOD32** | P2MOD3 | | | | P2MOD2 | | | |
| A0h | 1111-1111 | **P2** | P2.7 | P2.6 | P2.5 | P2.4 | P2.3 | P2.2 | P2.1 | P2.0 |
| A1h | 0000-0000 | **PWMCON** | PWM1CKS | | PWM1EN | PWM0EN | PWM0CKS | | PWM0NMSK | PWM0PMSK |
| A2h | 0001-0001 | **P3MOD10** | P3MOD1 | | | | P3MOD0 | | | |
| A3h | 0001-0001 | **P3MOD32** | P3MOD3 | | | | P3MOD2 | | | |
| A4h | 0001-0001 | **P3MOD54** | P3MOD5 | | | | P3MOD4 | | | |
| A5h | 0001-0001 | **P3MOD76** | P3MOD7 | | | | P3MOD6 | | | |
| A6h | x000-0000 | **PINMOD** | – | SPIPS | UARTRXS | | MSDAPS | | MSCLPS | |
| A7h | 0001-0001 | **P2MOD54** | P2MOD5 | | | | P2MOD4 | | | |
| A8h | 0x00-0000 | **IE** | EA | – | ET2 | ES | ET1 | EX1 | ET0 | EX0 |
| A9h | 0000-0000 | **INTE1** | PWMIE | CMPIE | LVDIE | SPI2CE | ADIE | WGIE | PCIE | TM3IE |
| AAh | xxxx-xxxx | **ADCDL** | ADCDL | | | | – | | | |
| ABh | xxxx-xxxx | **ADCDH** | ADCDH | | | | | | | |
| ACh | xxxx-xxxx | – | – | | | | | | | |
| ADh | xxxx-xxxx | – | – | | | | | | | |
| AEh | 1111-1000 | **ADCHSEL** | ADCHS | | | | | ADCVREFS | ADCVBGS | |

| Adr | RST | NAME | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| AFh | 0000-0000 | **PWMCON2** | PWM0MOD | PWM0MSKE | PWM0OM | | PWM0DZ | | | |
| B0h | 1111-1111 | **P3** | P3.7 | P3.6 | P3.5 | P3.4 | P3.3 | P3.2 | P3.1 | P3.0 |
| B1h | xxxx-xxxx | – | – | | | | | | | |
| B2h | xxxx-xxxx | – | – | | | | | | | |
| B3h | xxxx-xxxx | – | – | | | | | | | |
| B4h | xxxx-xxxx | – | – | | | | | | | |
| B5h | xxxx-xxxx | – | – | | | | | | | |
| B6h | 0000-0000 | **WGCON** | WGPRD | | | WGT1H | | | WGT0H | |
| B7h | 0000-0000 | **WGBUF** | WGBUF | | | | | | | |
| B8h | xx00-0000 | **IP** | – | – | PT2 | PS | PT1 | PX1 | PT0 | PX0 |
| B9h | xx00-0000 | **IPH** | – | – | PT2H | PSH | PT1H | PX1H | PT0H | PX0H |
| BAh | 0000-0000 | **IP1** | PPWM | PCMP | PLVD | PSPI2C | PADI | PWG | PPC | PT3 |
| BBh | 0000-0000 | **IP1H** | PPWMH | PCMPH | PLVDH | PSPI2CH | PADIH | PWGH | PPCH | PT3H |
| BCh | 0000-x000 | **SPCON** | SPEN | MSTR | CPOL | CPHA | – | LSBF | SPCR | |
| BDh | 00x0-00xx | **SPSTA** | SPIF | WCOL | – | RCVOVF | RCVBF | SPBSY | – | – |
| BEh | 0000-0000 | **SPDAT** | SPDAT | | | | | | | |
| BFh | xxxx-xxxx | – | – | | | | | | | |
| C0h | xxxx-xxxx | – | – | | | | | | | |
| C1h | xxxx-xxxx | – | – | | | | | | | |
| C2h | xxxx-xxxx | – | – | | | | | | | |
| C3h | xxxx-xxxx | – | – | | | | | | | |
| C4h | xx00-0000 | **LVDDT** | – | – | LVDDT | | | | | |
| C5h | 0000-0000 | **DACDT** | – | DACDT | | | | | | |
| C6h | 1110-0000 | **CMPCON** | DACMPPD | CMPOX | CMPHYS | CMPINV | CMPTRIG | | CMPDBS | |
| C7h | 1111-x111 | **CMPPNS** | SCMPN | SCIN | | | – | SCIP | | |
| C8h | 0000-0000 | **T2CON** | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | CT2N | CPRL2N |
| C9h | 0000-xxxx | **IAPCON** | IAPCON / IAPWE / EEPWE / INFOWE / IAPTO | | | | | | | |
| CAh | 0000-0000 | **RCP2L** | RCP2L | | | | | | | |
| CBh | 0000-0000 | **RCP2H** | RCP2H | | | | | | | |
| CCh | 0000-0000 | **TL2** | TL2 | | | | | | | |
| CDh | 0000-0000 | **TH2** | TH2 | | | | | | | |
| CEh | xxxx-xxxx | – | – | | | | | | | |
| CFh | xxxx-xxxx | – | – | | | | | | | |
| D0h | 0000-0000 | **PSW** | CY | AC | F0 | RS1 | RS0 | OV | F1 | P |
| D1h | 1000-0000 | **PWM0DH** | PWM0DH | | | | | | | |
| D2h | 1000-0000 | **PWM0DL** | PWM0DL | | | | | | | |
| D3h | xxxx-xxxx | – | – | | | | | | | |
| D4h | 1000-0000 | **PWM1D** | PWM1D | | | | | | | |
| D5h | xxxx-xxxx | – | – | | | | | | | |
| D6h | 1000-0000 | **PWM2D** | PWM2D | | | | | | | |
| D7h | 0000-0000 | **TM3RLD** | TM3RLD | | | | | | | |
| D8h | 0x10-0011 | **CLKCON** | SCKTYPE | – | STPSCK | STPPCK | STPFCK | SELFCK | CLKPSC | |
| D9h | 1111-1111 | **PWM0PRDH** | PWM0PRDH | | | | | | | |
| DAh | 1111-1111 | **PWM0PRDL** | PWM0PRDL | | | | | | | |
| DBh | xxxx-xxxx | – | – | | | | | | | |
| DCh | 1111-1111 | **PWM1PRD** | PWM1PRD | | | | | | | |
| DDh | xxxx-xxxx | **–** | – | | | | | | | |
| DEh | 1000-0000 | **PWM3D** | PWM3D | | | | | | | |
| DFh | 0000-0000 | **UARTBRP** | UARTBRP | | | | | | | |

| Adr | RST | NAME | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 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 | xxx0-0000 | **LVRCON** | – | – | – | LVRPD | LVRSEL | | | |
| E4h | 0000-0000 | **LVDCON** | LVDM | LVDO | LVDHYS | LVDPD | LVDSEL | | | |
| E5h | xxxx-xxxx | – | – | | | | | | | |
| E6h | xxxx-xxxx | – | – | | | | | | | |
| E7h | xxxx-xxxx | – | – | | | | | | | |
| E8h | xxxx-xxxx | – | – | | | | | | | |
| E9h | xxxx-xxxx | – | – | | | | | | | |
| EAh | 1000-0000 | **PWM4D** | PWM4D | | | | | | | |
| EBh | xxxx-xxxx | – | – | | | | | | | |
| ECh | 1000-0000 | **PWM5D** | PWM5D | | | | | | | |
| EDh | xxxx-xxxx | – | – | | | | | | | |
| EEh | 1000-0000 | **PWM6D** | PWM6D | | | | | | | |
| EFh | 0000-00xx | **AUX3** | *Chapter 7* | | | PRGD | WARMTIME | DACVREFS | – | – |
| F0h | 0000-0000 | **B** | B.7 | B.6 | B.5 | B.4 | B.3 | B.2 | B.1 | B.0 |
| F1h | xxxx-xxxx | – | – | | | | | | | |
| F2h | xxxx-xxxx | – | – | | | | | | | |
| F3h | xxxx-xxxx | – | – | | | | | | | |
| F4h | xxxx-xxxx | – | – | | | | | | | |
| F5h | xxxx-xxxx | **CFGBG** | – | – | – | BGTRIM | | | | |
| F6h | xxxx-xxxx | **CFGWL** | – | FRCF | | | | | | |
| F7h | 0000-x11x | **AUX2** | WDTE | | PWRSAV | VBGOUT | – | IAPTE | | – |
| F8h | 00x0-1100 | **AUX1** | CLRWDT | CLRTM3 | – | ADSOC | CLRPWM0 | CLRPWM1 | WGEN | DPSEL |

| Flash Address | NAME | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|
| 3FFFh | **CFGWH** | PROT | XRSTE | – | – | – | – | – | – |

# SFR & CFGW DESCRIPTION

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| 80h | **P0** | 7~0 | P0 | R/W | FFh | Port0 data |
| 81h | **SP** | 7~0 | SP | R/W | 07h | Stack Point |
| 82h | **DPL** | 7~0 | DPL | R/W | 00h | Data Point low byte |
| 83h | **DPH** | 7~0 | DPH | R/W | 00h | Data Point high byte |
| 84h | **INTE2** | 6 | PWM1IE | R/W | 0 | PWM1 interrupt enable<br>0: Disable PWM1 interrupt<br>1: Enable PWM1 interrupt |
| | | 5 | PWM0IE | R/W | 0 | PWM0 interrupt enable<br>0: Disable PWM0 interrupt<br>1: Enable PWM0 interrupt |
| 85h | **INTFLG2** | 6 | PWM1IF | R/W | 0 | PWM1 interrupt flag<br>Set by H/W at the end of PWM1 period, S/W writes BFh to INTFLG2 to clear this flag. |
| | | 5 | PWM0IF | R/W | 0 | PWM0 interrupt enable<br>Set by H/W at the end of PWM0 period, S/W writes DFh to INTFLG2 to clear this flag. |
| 87h | **PCON** | 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 | **TCON** | 7 | TF1 | R/W | 0 | Timer1 overflow flag<br>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<br>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<br>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<br>0: Low level active (level triggered) for INT1 pin<br>1: Falling edge active (edge triggered) for INT1 pin |
| | | 1 | IE0 | R/W | 0 | External Interrupt 0 (INT0 pin) edge flag<br>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<br>0: Low level active (level triggered) for INT0 pin<br>1: Falling edge active (edge triggered) for INT0 pin |
| 89h | **TMOD** | 7 | GATE1 | R/W | 0 | Timer1 gating control bit<br>0: Timer1 enable when TR1 bit is set<br>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<br>0: Timer mode, Timer1 data increases at 2 System clock cycle rate<br>1: Counter mode, Timer1 data increases at T1 pin's negative edge |
| | | 5~4 | TMOD1 | R/W | 00 | Timer1 mode select<br>00: 8-bit timer/counter (TH1) and 5-bit prescaler (TL1)<br>01: 16-bit timer/counter<br>10: 8-bit auto-reload timer/counter (TL1). Reloaded from TH1 at overflow.<br>11: Timer1 stops |
| | | 3 | GATE0 | R/W | 0 | Timer0 gating control bit<br>0: Timer0 enable when TR0 bit is set<br>1: Timer0 enable only while the INT0 pin is high and TR0 bit is set |
| | | 2 | CT0N | R/W | 0 | Timer0 Counter/Timer select bit<br>0: Timer mode, Timer0 data increases at 2 System clock cycle rate<br>1: Counter mode, Timer0 data increases at T0 pin's negative edge |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| | | 1~0 | TMOD0 | R/W | 00 | Timer0 mode select<br>00: 8-bit timer/counter (TH0) and 5-bit prescaler (TL0)<br>01: 16-bit timer/counter<br>10: 8-bit auto-reload timer/counter (TL0). Reloaded from TH0 at overflow.<br>11: TL0 is an 8-bit timer/counter.<br>TH0 is an 8-bit timer/counter using Timer1's TR1 and TF1 bits. |
| 8Ah | **TL0** | 7~0 | TL0 | R/W | 00h | Timer0 data low byte |
| 8Bh | **TL1** | 7~0 | TL1 | R/W | 00h | Timer1 data low byte |
| 8Ch | **TH0** | 7~0 | TH0 | R/W | 00h | Timer0 data high byte |
| 8Dh | **TH1** | 7~0 | TH1 | R/W | 00h | Timer1 data high byte |
| 90h | **P1** | 7~0 | P1 | R/W | FFh | Port1 data |
| 91h | **P0MOD10** | 7~4 | P0MOD1 | R/W | 0001 | P0.1 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P0MOD0 | R/W | 0001 | P0.0 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 92h | **P0MOD32** | 7~4 | P0MOD3 | R/W | 0001 | P0.3 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P0MOD2 | R/W | 0001 | P0.2 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 93h | **P0MOD54** | 7~4 | P0MOD5 | R/W | 0001 | P0.5 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P0MOD4 | R/W | 0001 | P0.4 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 94h | **OPTION** | 6 | TM3CKS | R/W | 0 | Timer3 clock source select.<br>0: Slow Clock (SRC/SXT) 1: FRC/512 (32.4KHz) |
| | | 5~4 | WDTPSC | R/W | 00 | Watchdog Timer pre-scalar time select<br>00: 528ms WDT overflow rate<br>01: 264ms WDT overflow rate<br>10: 132ms WDT overflow rate<br>11: 66ms WDT overflow rate |
| | | 3~2 | ADCKS | R/W | 00 | ADC clock rate select<br>00: $F_{SYSCLK}/32$<br>01: $F_{SYSCLK}/16$<br>10: $F_{SYSCLK}/8$<br>11: $F_{SYSCLK}/4$ |
| | | 1~0 | SXTGAIN | R/W | 00 | SXT oscillator gain<br>00=Lowest gain, 11=Highest Gain |
| 95h | **INTFLG** | 7 | LVDIF | R/W | 0 | LVD interrupt flag<br>Set by H/W when $V_{CC}$ less than the LVD voltage. S/W writes 7Fh to INTFLG to clear this flag. |
| | | 6 | CMPIF | R/W | 0 | CMP interrupt flag<br>Set by H/W while CMPO match trigger condition. It is cleared automatically when the program performs the interrupt service routine. S/W writes BFh to INTFLG to clear this flag. |
| | | 4 | ADIF | R/W | 0 | ADC interrupt flag<br>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. |
| | | 3 | WGIF | R/W | 0 | Wave Generator interrupt flag<br>Set by H/W at the end of Wave generate. S/W writes F7h to INTFLG to clear this flag. |
| | | 1 | PCIF | R/W | 0 | Port0~3 pin change Interrupt flag<br>Set by H/W when a Port0~3 pin state change is detected and its interrupt enable bit is set. PCIE does not affect this flag's setting. It is cleared automatically when the program performs the interrupt service routine. S/W can write FDh to INTFLG to clear this bit. |
| | | 0 | TF3 | R/W | 0 | Timer3 interrupt flag.<br>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 | **P0MOD76** | 7~4 | P0MOD7 | R/W | 0001 | P0.7 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P0MOD6 | R/W | 0001 | P0.6 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 97h | **SWCMD** | 7~0 | SWRST | W | | Write 56h to generate S/W Reset |
| | | 7~0 | IAPALL | W | | Write 65h to set IAPALL flag. Write other value to clear IAPALL flag. |
| | | 1 | WDTO | R | 0 | Watchdog Time-Out flag |
| | | 0 | IAPALL | R | 0 | Flag indicates Flash can be written by IAP or not<br>0: Flash IAP only can write IAP-free area.<br>1: Flash IAP can write IAP-all area. |
| 98h | **SCON** | 7 | SM0 | R/W | 0 | UART Serial port mode select bit 0, 1 (SM0, SM1) =<br>00: Reserved<br>01: Mode1: 8 bit UART, Baud Rate is variable.<br>10: Reserved<br>11: Mode3: 9 bit UART, Baud Rate is variable. |
| | | 6 | SM1 | R/W | 1 | |
| | | 5 | SM2 | R/W | 0 | UART Serial port mode select bit 2<br>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 UART 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<br>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<br>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_TX | W | – | UART transmit data. Transmit data is written to this locationt. |
| | | | SBUF_RX | R | – | UART receive data. Receive data is read from this location. |
| 9Ah | **P1MOD10** | 7~4 | P1MOD1 | R/W | 0001 | P1.1 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P1MOD0 | R/W | 0001 | P1.0 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 9Bh | **P1MOD32** | 7~4 | P1MOD3 | R/W | 0001 | P1.3 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P1MOD2 | R/W | 0001 | P1.2 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 9Ch | **P1MOD54** | 7~4 | P1MOD5 | R/W | 0001 | P1.5 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P1MOD4 | R/W | 0001 | P1.4 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 9Dh | **P1MOD76** | 7~4 | P1MOD7 | R/W | 0001 | P1.7 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P1MOD6 | R/W | 0001 | P1.6 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 9Eh | **P2MOD10** | 7~4 | P2MOD1 | R/W | 0001 | P2.1 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P2MOD0 | R/W | 0001 | P2.0 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| 9Fh | **P2MOD32** | 7~4 | P2MOD3 | R/W | 0001 | P2.3 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P2MOD2 | R/W | 0001 | P2.2 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| A0h | **P2** | 7~6 | P2.7~P2.6 | R/W | FFh | P2.7~P2.6 have no pin out, so these bits are used as general purpose register |
| | | 5~0 | P2.5~P2.0 | R/W | 11 | P2.5~P2.0 data |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| A1h | **PWMCON** | 7~6 | PWM1CKS | R/W | 00 | PWM1 clock source<br>00: $F_{SYSCLK}$<br>01: FRC/256<br>10: FRC<br>11: FRCx2 (Vcc>2.5V) |
| | | 5 | PWM1EN | R/W | 0 | PWM1 Enable. 0: PWM1 Disable, 1: PWM1 Enable |
| | | 4 | PWM0EN | R/W | 0 | PWM0 Enable. 0: PWM0 Disable, 1: PWM0 Enable |
| | | 3~2 | PWM0CKS | R/W | 00 | PWM0 clock source<br>00: $F_{SYSCLK}$<br>01: FRC/256<br>10: FRC<br>11: FRCx2 (Vcc>2.5V) |
| | | 1 | PWM0NMSK | R/W | 0 | PWM0N mask data. If CLRPWM0=1 and PMW0MSKE=1, PWM0N will output this mask data. |
| | | 0 | PWM0PMSK | R/W | 0 | PWM0P mask data. If CLRPWM0=1 and PMW0MSKE=1, PWM0P will output this mask data. |
| A2h | **P3MOD10** | 7~4 | P3MOD1 | R/W | 0001 | P3.1 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P3MOD0 | R/W | 0001 | P3.0 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| A3h | **P3MOD32** | 7~4 | P3MOD3 | R/W | 0001 | P3.3 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P3MOD2 | R/W | 0001 | P3.2 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| A4h | **P3MOD54** | 7~4 | P3MOD5 | R/W | 0001 | P3.5 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P3MOD4 | R/W | 0001 | P3.4 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| A5h | **P3MOD76** | 7~4 | P3MOD7 | R/W | 0001 | P3.7 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P3MOD6 | R/W | 0001 | P3.6 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| A6h | **PINMOD** | 6 | SPIPS | R/W | 0 | SPI pin select<br>0: SCK/MOSI/MISO = P3.5/P2.0/P2.1<br>1: SCK/MOSI/MISO = P1.2/P1.0/P1.1 |
| | | 5~4 | UARTRXS | R/W | 0 | UART RXD pin select (TXD pin select by Pin Control)<br>00: RXD = P3.0<br>01: RXD = P0.2<br>10: RXD = P3.1<br>11: RXD = P1.6 |
| | | 3~2 | MSDAPS | R/W | 0 | Master $I^2C$ SDA pin select<br>00: P3.5<br>01: P1.6<br>10: P2.1<br>11: P2.5 |
| | | 1~0 | MSCLPS | R/W | 0 | Master $I^2C$ SCL pin select<br>00: P1.3<br>01: P0.2<br>10: P2.0<br>11: P2.4 |
| A7h | **P2MOD54** | 7~4 | P2MOD5 | R/W | 0001 | P2.5 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| | | 3~0 | P2MOD4 | R/W | 0001 | P2.4 Pin Control<br>0000~1111: Mode0~Mode15, see PINMODE table 7.1 |
| A8h | **IE** | 7 | EA | R/W | 0 | Global interrupt enable control.<br>0: Disable all Interrupts.<br>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 (UART) 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 |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| | | | | | | up capability. |
| | | 1 | ET0 | R/W | 0 | Set 1 to enable Timer0 Interrupt |
| | | 0 | EX0 | R/W | 0 | Set 1 to enable external INT0 pin Interrupt & Halt/Stop mode wake up capability. |
| A9h | **INTE1** | 7 | PWMIE | R/W | 0 | Set 1 to enable PWM0/PWM1 Interrupt |
| | | 6 | CMPIE | R/W | 0 | Set 1 to enable CMP Interrupt |
| | | 5 | LVDIE | R/W | 0 | Set 1 to enable LVD Interrupt |
| | | 4 | SPI2CE | R/W | 0 | Set 1 to enable SPI/Master $I^2C$ Interrupt |
| | | 3 | ADIE | R/W | 0 | Set 1 to enable ADC Interrupt |
| | | 2 | WGIE | R/W | 0 | Set 1 to enable Wave Generator Interrupt |
| | | 1 | PCIE | R/W | 0 | Set 1 to enable Port0/Port1/Port2/Port3 Pin Change Interrupt |
| | | 0 | TM3IE | R/W | 0 | Set 1 to enable Timer3 Interrupt |
| AAh | **ADCDL** | 7~4 | ADCDL | R | – | ADC data bit 3~0 |
| ABh | **ADCDH** | 7~0 | ADCDH | R | – | ADC data bit 11~4 |
| AEh | **ADCHSEL** | 7~3 | ADCHS | R/W | 11111 | ADC channel select<br>00000: AD0 (P0.0)<br>00001: AD1 (P0.1)<br>00010: AD2 (P0.2)<br>00011: AD3 (P0.3)<br>00100: AD4 (P0.4)<br>00101: AD5 (P0.5)<br>00110: AD6 (P0.6)<br>00111: AD7 (P0.7)<br>01000: AD8 (P1.7)<br>01001: AD9 (P3.4)<br>01010: AD10 (P3.1)<br>01011: AD11 (P3.0)<br>01100: AD12 (P3.7)<br>01101: AD13 (P3.2)<br>01110: AD14 (P3.3)<br>01111: AD15 (P1.6)<br>10000: AD16 (P3.6)<br>10001: AD17 (P3.5)<br>10010: AD18 (P2.1)<br>10011: AD19 (P2.0)<br>10100: AD20 (P1.5)<br>10101: AD21 (P1.4)<br>10110: AD22 (P1.3)<br>10111: AD23 (P2.4)<br>11000: AD24 (P2.5)<br>11001: AD25 (P1.2)<br>11010: AD26 (P1.1)<br>11011: AD27 (P1.0)<br>11100: $V_{BG}$<br>11101: DAC<br>11110: Reserved<br>11111: $V_{CC}/4$ |
| | | 2 | ADCVREFS | R/W | 0 | ADC reference voltage select<br>0: $V_{CC}$<br>1: $V_{BG}$ |
| | | 1~0 | ADCVBGS | R/W | 00 | $V_{BG}$ voltage select for ADC.<br>00: 1.18V<br>01: 2.0V (need $V_{CC}$>2.8V)<br>10: 3.0V (need $V_{CC}$>3.3V)<br>11: 4.0V (need $V_{CC}$>4.3V) |
| AFh | **PWMCON2** | 7 | PWM0MOD | R/W | 0 | PWM0 mode select<br>0: Normal mode<br>1: Half-bridge mode |
| | | 6 | PWM0MSKE | R/W | 0 | PWM0 mask output enable<br>0: Disable<br>1: Enable, PWM0P/PWM0N output data by |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| | | | | | | PWM0PMSK/PWM0NMSK while CLRPWM0=1 |
| | | 5~4 | PWM0OM | R/W | 00 | PWM0 output mode select<br>00: Mode0<br>01: Mode1<br>10: Mode2<br>11: Mode3 |
| | | 3~0 | PWM0DZ | R/W | 0000 | PWM0 dead zone<br>0000: 0 x $T_{PWMCLK}$<br>0001: 1 x $T_{PWMCLK}$<br>…<br>1111: 15 x $T_{PWMCLK}$ |
| B0h | **P3** | 7~0 | P3 | R/W | FFh | Port3 data |
| B6h | **WGCON** | 7~5 | WGPRD | R/W | 000 | WG period of each bit select ($T_{FRC}$ = 60.28 ns)<br>0: 15*$T_{FRC}$ = 904 ns<br>1: 16*$T_{FRC}$ = 965 ns<br>2: 17*$T_{FRC}$ = 1025 ns<br>3: 18*$T_{FRC}$ = 1085 ns<br>4: 19*$T_{FRC}$ = 1145 ns<br>5: 20*$T_{FRC}$ = 1206 ns<br>6: 21*$T_{FRC}$ = 1266 ns<br>7: 22*$T_{FRC}$ = 1326 ns |
| | | 4~2 | WGT1H | R/W | 000 | WG code 1 high level time select ($T_{FRC}$ = 60.28 ns)<br>0: 10*$T_{FRC}$ = 603 ns<br>1: 11*$T_{FRC}$ = 663 ns<br>2: 12*$T_{FRC}$ = 723 ns<br>3: 13*$T_{FRC}$ = 784 ns<br>4: 14*$T_{FRC}$ = 844 ns<br>5: 15*$T_{FRC}$ = 904 ns<br>6: 16*$T_{FRC}$ = 965 ns<br>7: 17*$T_{FRC}$ = 1025 ns |
| | | 1~0 | WGT0H | R/W | 00 | WG code 0 high level time select ($T_{FRC}$ = 60.28 ns)<br>0: 4*$T_{FRC}$ = 241 ns<br>1: 5*$T_{FRC}$ = 301 ns<br>2: 6*$T_{FRC}$ = 362 ns<br>3: 7*$T_{FRC}$ = 422 ns |
| B7h | **WGBUF** | 7~0 | WGBUF | R/W | 00h | Wave Generator code buffer |
| B8h | **IP** | 5 | PT2 | R/W | 0 | Timer2 Interrupt Priority Low bit |
| | | 4 | PS | R/W | 0 | Serial Port (UART) 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 (UART) 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 | PWM0/PWM1 Interrupt Priority Low bit |
| | | 6 | PCMP | R/W | 0 | CMP Interrupt Priority Low bit |
| | | 5 | PLVD | R/W | 0 | LVD Interrupt Priority Low bit |
| | | 4 | PSPI2C | R/W | 0 | SPI/Master $I^2C$ Interrupt Priority Low bit |
| | | 3 | PADI | R/W | 0 | ADC Interrupt Priority Low bit |
| | | 2 | PWG | R/W | 0 | Wave Generator 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 |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| BBh | **IP1H** | 7 | PPWMH | R/W | 0 | PWM0/PWM1 Interrupt Priority High bit |
| | | 6 | PCMPH | R/W | 0 | CMP Interrupt Priority High bit |
| | | 5 | PLVDH | R/W | 0 | LVD Interrupt Priority High bit |
| | | 4 | PSPI2CH | R/W | 0 | SPI/Master $I^2C$ Interrupt Priority High bit |
| | | 3 | PADIH | R/W | 0 | ADC Interrupt Priority High bit |
| | | 2 | PWGH | R/W | 0 | Wave Generator Interrupt Priority High bit |
| | | 1 | PPCH | R/W | 0 | Port0~Port3 Interrupt Priority High bit |
| | | 0 | PT3H | R/W | 0 | Timer3 Interrupt Priority High bit |
| BCh | **SPCON** | 7 | SPEN | R/W | 0 | SPI enable<br>0: SPI disable<br>1: SPI enable |
| | | 6 | MSTR | R/W | 0 | Master mode enable<br>0: Slave mode<br>1: Master mode |
| | | 5 | CPOL | R/W | 0 | SPI clock polarity<br>0: SCK is low in idle state<br>1: SCK is high in idle state |
| | | 4 | CPHA | R/W | 0 | SPI clock phase<br>0: Data sample on first edge of SCK period<br>1: Data sample on second edge of SCK period |
| | | 2 | LSBF | R/W | 0 | LSB first<br>0: MSB first<br>1: LSB first |
| | | 1~0 | SPCR | R/W | 00 | SPI clock rate<br>00: $F_{SYSCLK}/2$<br>01: $F_{SYSCLK}/4$<br>10: $F_{SYSCLK}/8$<br>11: $F_{SYSCLK}/16$ |
| BDh | **SPSTA** | 7 | SPIF | R/W | 0 | SPI interrupt flag<br>This is set by H/W at the end of a data transfer. Cleared by H/W when an interrupt is vectored into. Writing 0 to this bit will clear this flag. |
| | | 6 | WCOL | R/W | 0 | Write collision interrupt flag<br>Set by H/W if write data to SPDAT when SPBSY is set. Write 0 to this bit or rewrite data to SPDAT when SPBSY is cleared will clear this flag. |
| | | 4 | RCVOVF | R/W | 0 | Received buffer overrun flag<br>Set by H/W at the end of a data transfer and RCVBF is set. Write 0 to this bit or read SPDAT register will clear this flag. |
| | | 3 | RCVBF | R/W | 0 | Receive buffer full flag<br>Set by H/W at the end of a data transfer. Write 0 to this bit or read SPDAT register will clear this flag. |
| | | 2 | SPBSY | R | 0 | SPI busy flag<br>Set by H/W when a SPI transfer is in progress. |
| BEh | **SPDAT** | 7~0 | SPDAT | R/W | 00h | SPI transmit and receive data<br>The SPDAT register is used to transmit and receive data. Writing data to SPDAT place the data into shift register and start a transfer when in master mode. Reading SPDAT returns the contents of the receive buffer. |
| C4h | **LVDDT** | 5~0 | LVDDT | R/W | 00 | LVD delay time select (step=FRC16.5888M/16，$T_{LVD}$=0.96us)<br>00_0000: disable<br>00_0001: $T_{LVD}$*1=0.96us<br>...<br>11_1111: $T_{LVD}$*63=60.76us |
| C5h | **DACDT** | 6~0 | DACDT | R/W | 00h | Select DAC output voltage, reference source can be selected as $V_{CC}/2$ or $V_{BG}$1.18V/2<br>000_0000: 0/128 * reference source<br>…<br>111_1111: 127/128* reference source |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| C6h | **CMPCON** | 7 | DACMPPD | R/W | 1 | Comparator & DAC power down control<br>0: enable Comparator & DAC<br>1: disable Comparator & DAC |
| | | 6 | CMPOX | R/W | 1 | Comparator original output (CMPOX) status<br>0: $V_{CMPP} < V_{CMPN}$<br>1: $V_{CMPP} > V_{CMPN}$ or CMPPD =1 |
| | | 5 | CMPHYS | R/W | 1 | Comparator Hysteresis Control<br>0: disable<br>1: enable |
| | | 4 | CMPINV | R/W | 0 | Comparator de-bounce output invert select<br>0: no invert<br>1: invert |
| | | 3~2 | CMPTRIG | R/W | 00 | Comparator interrupt trigger mode<br>00: Rising edge<br>01: Falling edge<br>10: Both edge<br>11: High level |
| | | 1`0 | CMPDBS | R/W | 00 | Comparator original output (CMPOX) de-bounce time<br>00: none<br>01: 4 $F_{SYSCLK}$<br>10: 8 $F_{SYSCLK}$<br>11: 16 $F_{SYSCLK}$ |
| C7h | **CMPPNS** | 7 | SCMPN | R/W | 1 | Comparator CMPN source select<br>0: Comparator CMPN source is external input (CINx)<br>1: Comparator CMPN source is DAC output |
| | | 6~4 | SCIN | R/W | 111 | Comparator CMPN external input select<br>000: Comparator CMPN external input is CIN1 (P1.6)<br>001: Comparator CMPN external input is CIN2 (P1.5)<br>010: Comparator CMPN external input is CIN3 (P0.7)<br>011: Comparator CMPN external input is CIN4 (P0.1)<br>100: Comparator CMPN external input is CIN5 ($V_{SS}$)<br>1xx: No connect |
| | | 2~0 | SCIP | R/W | 111 | Comparator CMPP external input select<br>000: Comparator CMPP external input is CIP1 (P3.3)<br>001: Comparator CMPP external input is CIP2 (P1.4)<br>010: Comparator CMPP external input is CIP3 (P1.7)<br>011: Comparator CMPP external input is CIP4 (P0.0)<br>100: Comparator CMPP external input is CIP5 ($V_{SS}$)<br>1xx: No connect |
| C8h | **T2CON** | 7 | TF2 | R/W | 0 | Timer2 overflow flag<br>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<br>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 | Force 0 |
| | | 4 | TCLK | R/W | 0 | Force 0 |
| | | 3 | EXEN2 | R/W | 0 | T2EX pin enable<br>0: T2EX pin disable<br>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<br>0:timer stops<br>1:timer runs |
| | | 1 | CT2N | R/W | 0 | Timer2 Counter/Timer select bit<br>0: Timer mode, Timer2 data increases at 2 System clock cycle rate<br>1: Counter mode, Timer2 data increases at T2 pin's negative edge |
| | | 0 | CPRL2N | R/W | 0 | Timer2 Capture/Reload control bit<br>0: Reload mode, auto-reload on Timer2 overflows or negative transitions on T2EX pin if EXEN2=1.<br>1: Capture mode, capture on negative transitions on T2EX pin if EXEN2=1.<br>If RCLK=1 or TCLK=1, CPRL2N is ignored and timer is forced to |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| | | | | | | auto-reload on Timer2 overflow. |
| C9h | **IAPCON** | 7~0 | IAPCON | 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.<br>Write A1h to set INFOWE flag; write other value to clear INFOWE flag. It is recommended to clear it immediately after IAP write.<br>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 | 0 | Flag indicates Flash memory can be written by IAP or not<br>0: IAP Write disable<br>1: IAP Write enable |
| | | 6 | IAPTO | R | 0 | Time-Out flag of IAP write/EEPROM write/INFO write.<br>Set by H/W when IAP or EEPROM or INFO write Time-out occurs.<br>Cleared this flag by H/W when IAPWE=0 or EEPWE=0 or INFOWE=0. |
| | | 5 | EEPWE | R | 0 | Flag indicates EEPROM memory can be written or not<br>0: EEPROM Write disable<br>1: EEPROM Write enable |
| | | 4 | INFOWE | R | 0 | Flag indicates INFO memory can be written or not<br>0: INFO IAP Write disable<br>1: INFO IAP Write enable |
| 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 |
| CDh | **TH2** | 7~0 | TH2 | R/W | 00h | Timer2 data high byte |
| 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 | 80h | PWM0 duty high byte<br>**write sequence: PWMxDL then PWMxDH**<br>**read sequence: PWMxDH then PWMxDL** |
| D2h | **PWM0DL** | 7~0 | PWM0DL | R/W | 00h | PWM0 duty low byte |
| D4h | **PWM1D** | 7~0 | PWM1D | R/W | 80h | PWM1 duty |
| D6h | **PWM2D** | 7~0 | PWM2D | R/W | 80h | PWM2 duty |
| D7h | **TM3RLD** | 7~0 | TM3RLD | R/W | 00 | 16-bit TM3 MSB 8-bit reload data<br>count range: [TM3RLD,00h]~FFFF |
| D8h | **CLKCON** | 7 | SCKTYPE | R/W | 0 | Slow clock Type. This bit can be changed only in Fast mode (SELFCK=1)<br>0: SRC<br>1: SXT, P2.2 and P2.3 are crystal pins |
| | | 5 | STPSCK | R/W | 1 | Set 1 to stop Slow clock in Stop Mode. |
| | | 4 | STPPCK | R/W | 0 | Set 1 to stop 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.<br>0: Slow clock<br>1: Fast clock |
| | | 1~0 | CLKPSC | R/W | 11 | System clock prescaler. Effective after 16 clock cycles (Max.) delay.<br>00: System clock is Fast/Slow clock divided by 16<br>01: System clock is Fast/Slow clock divided by 4<br>10: System clock is Fast/Slow clock divided by 2<br>11: System clock is Fast/Slow clock divided by 1 |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| D9h | **PWM0PRDH** | 7~0 | PWM0PRDH | R/W | FFh | PWM0 period high byte<br>**write sequence: PWM0PRDL then PWM0PRDH**<br>**read sequence: PWM0PRDH then PWM0PRDL** |
| DAh | **PWM0PRDL** | 7~0 | PWM0PRDL | R/W | FFh | PWM0 period low byte |
| DCh | **PWM1PRD** | 7~0 | PWM1PRD | R/W | FFh | PWM1~PWM6 shared period |
| DEh | **PWM3D** | 7~0 | PWM3D | R/W | 80h | PWM3 duty |
| DFh | **UARTBRP** | 7~0 | UARTBRP | R/W | 0 | Define UART Baud Rate prescaler<br>UART Baud Rate = $F_{SYSCLK}$/16/UARTBRP |
| E0h | **ACC** | 7~0 | ACC | R/W | 00h | Accumulator |
| E1h | **MICON** | 7 | MIEN | R/W | 0 | Master $I^2C$ enable<br>0: disable<br>1: enable |
| | | 6 | MIACKO | R/W | 0 | When Master $I^2C$ receive data, send acknowledge to $I^2C$ bus<br>0: ACK to slave device<br>1: NACK to slave device |
| | | 5 | MIIF | R/W | 0 | Master $I^2C$ Interrupt flag<br>0: write 0 to clear it<br>1: Master $I^2C$ transfer one byte complete |
| | | 4 | MIACKI | R | – | When Master $I^2C$ transfer, acknowledgement form $I^2C$ bus (read only)<br>0: ACK received<br>1: NACK received |
| | | 3 | MISTART | R/W | 0 | Master $I^2C$ Start bit<br>1: start $I^2C$ bus transfer |
| | | 2 | MISTOP | R/W | 1 | Master $I^2C$ Stop bit<br>1: send STOP signal to stop $I^2C$ bus |
| | | 1~0 | MICR | R/W | 00 | Master $I^2C$ (SCL) clock frequency selection<br>00: $F_{SYSCLK}$/4<br>01: $F_{SYSCLK}$/16<br>10: $F_{SYSCLK}$/64<br>11: $F_{SYSCLK}$/256 |
| E2h | **MIDAT** | 7~0 | MIDAT | R/W | 00h | Master $I^2C$ data shift register<br>(W): After Start and before Stop condition, write this register will resume transmission to $I^2C$ bus<br>(R): After Start and before Stop condition, read this register will resume receiving from $I^2C$ bus |
| E3h | **LVRCON** | 4 | LVRPD | R/W | 0 | LVR Power Down.<br>0: LVR Enable<br>1: LVR Disable |
| | | 3~0 | LVRSEL | R/W | 0000 | Low Voltage Reset (LVR) select. (step=0.15V)<br>0000: Set LVR at 1.70V<br>0001: Set LVR at 1.85V<br>0010: Set LVR at 2.00V<br>0011: Set LVR at 2.15V<br>0100: Set LVR at 2.30V<br>0101: Set LVR at 2.45V<br>0110: Set LVR at 2.60V<br>0111: Set LVR at 2.75V<br>1000: Set LVR at 2.90V<br>1001: Set LVR at 3.05V<br>1010: Set LVR at 3.20V<br>1011: Set LVR at 3.35V<br>1100: Set LVR at 3.50V<br>1101: Set LVR at 3.65V<br>1110: Set LVR at 3.80V<br>1111: Set LVR at 3.95V |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| E4h | **LVDCON** | 7 | LVDM | R/W | 0 | Low Voltage Detect interrupt enable<br>0: LVDIF =1 and LVDO =1 while $V_{CC} < V_{LVD}$<br>1: LVDIF =1 and LVDO =0 while $V_{CC} > V_{LVD}$ |
| | | 6 | LVDO | R | 0 | Low Voltage Detect output |
| | | 5 | LVDHYS | R/W | 0 | LVD Hysteresis Enable<br>0: LVD Hysteresis disable<br>1: LVD Hysteresis enable |
| | | 4 | LVDPD | R/W | 0 | LVD power down<br>0: LVD enable<br>1: LVD disable |
| | | 3~0 | LVDSEL | R/W | 0000 | Low Voltage Detect (LVD) select. (step=0.15V)<br>0000: Set LVD at 1.85V<br>0001: Set LVD at 2.00V<br>0010: Set LVD at 2.15V<br>0011: Set LVD at 2.30V<br>0100: Set LVD at 2.45V<br>0101: Set LVD at 2.60V<br>0110: Set LVD at 2.75V<br>0111: Set LVD at 2.90V<br>1000: Set LVD at 3.05V<br>1001: Set LVD at 3.20V<br>1010: Set LVD at 3.35V<br>1011: Set LVD at 3.50V<br>1100: Set LVD at 3.65V<br>1101: Set LVD at 3.80V<br>1110: Set LVD at 3.95V<br>1111: Set LVD at 4.10V |
| EAh | **PWM4D** | 7~0 | PWM4D | R/W | 80h | PWM4 duty |
| ECh | **PWM5D** | 7~0 | PWM5D | R/W | 80h | PWM5 duty |
| EEh | **PWM6D** | 7~0 | PWM6D | R/W | 80h | PWM6 duty |
| EFh | **AUX3** | 7~5 | PWM1PSC | R/W | 0 | PWM1~PWM6 clock pre-scaler select<br>0: div 1<br>1: div 2<br>2: div 4<br>3: div 8<br>4: div 16<br>5: div 32<br>6: div 64<br>7: div 128 |
| | | 4 | PRGD | R/W | 0 | Disable P21/P20 to enter PRG/ICE mode control<br>0: Enable P21/P20 to enter PRG/ICE mode<br>1: Disable P21/P20 to enter PRG/ICE mode |
| | | 3 | WARMTIME | R/W | 0 | Warm-up time for wake-up from Power Down mode<br>0: 128 Clock 1: 64 Clock |
| | | 2 | DACVREFS | R/W | 0 | DAC reference voltage select<br>0: $V_{CC}/2$<br>1: $V_{BG}$ 1.18V/2 |
| F0h | **B** | 7~0 | B | R/W | 00h | B register |
| F5h | **CFGBG** | 4~0 | BGTRIM | R/W | – | VBG trimming value |
| F6h | **CFGWL** | 6~0 | FRCTRIM | R/W | – | FRC frequency adjustment<br>00h: lowest frequency<br>7Fh: highest frequency |

| Adr | SFR | Bit# | Bit Name | R/W | Rst | Description |
|---|---|---|---|---|---|---|
| F7h | **AUX2** | 7~6 | WDTE | R/W | – | Watchdog Timer Reset control<br>0x: WDT disable<br>10: WDT enable in Fast/Slow mode, disable in Idle/Halt/Stop mode<br>11: WDT always enable |
| | | 5 | PWRSAV | R/W | – | Set 1 to reduce the chip's power consumption at Idle/Halt/Stop Mode. |
| | | 4 | VBGOUT | R/W | 0 | Bandgap voltage output control<br>0: P3.2 as normal I/O<br>1: Bandgap voltage output to P3.2 pin |
| | | 2~1 | IAPTE | R/W | 11 | IAP write/EEPROM write/INFO write watchdog timer enable<br>00: Disable<br>01: wait 2ms trigger watchdog time-out flag<br>10: wait 4ms trigger watchdog time-out flag<br>11: wait 16ms trigger watchdog time-out flag |
| F8h | **AUX1** | 7 | CLRWDT | R/W | 0 | Set 1 to clear WDT, H/W auto clear it at next clock cycle |
| | | 6 | CLRTM3 | R/W | 0 | Set 1 to clear and hold Timer3, need S/W clear. |
| | | 4 | ADSOC | R/W | 0 | ADC Start of Conversion<br>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 | CLRPWM0 | R/W | 1 | PWM0 clear enable<br>0: PWM0 is running<br>1: PWM0 is cleared and held or set PWM0 stop status by PWM0PMSK/PWM0NMSK & PWM0MSK=1 |
| | | 2 | CLRPWM1 | R/W | 1 | PWM1~PWM6 clear enable<br>0: PWM1~PWM6 is running<br>1: PWM1~PWM6 is cleared and held |
| | | 1 | WGEN | R/W | 0 | Wave Generator enable, H/W will automatically clear WGEN after one cycle<br>0: WG disable<br>1: WG enable |
| | | 0 | DPSEL | R/W | 0 | Active DPTR Select |

| Adr | Flash | Bit# | Bit Name | Description |
|---|---|---|---|---|
| 3FFBh | **CFGBG** | 4~0 | BGTRIM | VBG adjustment. VBG is trimmed to 1.18V in chip manufacturing. |
| 3FFDh | **CFGWL** | 6~0 | FRCTRIM | FRC frequency adjustment. FRC is trimmed to 16.5888 MHz in chip manufacturing. |
| 3FFFh | **CFGWH** | 7 | PROT | Flash Code Protect, 1=Protect |
| | | 6 | XRSTE | External Pin Reset Enable, 1=Enable. |
| | | 5~0 | - | 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 | A4 |
| DIV AB | Divide A by B | 1 | 8 | 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 |
| SWAP A | Swap Nibbles of A | 1 | 2 | C4 |

| LOGICAL | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **byte** | **cycle** | **opcode** |
| 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 |

| BRANCHING | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **byte** | **cycle** | **Opcode** |
| 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 |

| MISCELLANEOUS | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **byte** | **cycle** | **opcode** |
| 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 ℃)

| Parameter | Rating | Unit |
|---|---|---|
| Supply voltage | $V_{SS}$ –0.3 ~ $V_{SS}$ +5.5 | V |
| Input voltage | $V_{SS}$ –0.3 ~ $V_{CC}$ +0.3 | |
| Output voltage | $V_{SS}$ –0.3 ~ $V_{CC}$ +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 | ℃ |
| Storage temperature | –65 ~ +150 | |

## 2. DC Characteristics ($T_A$=25 ℃，$V_{CC}$=1.9V ~ 5.5V)

| Parameter | Symbol | Conditions | | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|---|
| Operating Voltage | $V_{CC}$ | Fast mode，$F_{SYSCLK}$= 16.5888 MHz | | | 1.9 | – | 5.5 | V |
| Input High Voltage | $V_{IH}$ | All Input | | $V_{CC}$=5V | $0.7V_{CC}$ | – | – | V |
| Input Low Voltage | $V_{IL}$ | All Input | | $V_{CC}$=5V | – | – | $0.3V_{CC}$ | V |
| I/O Port Source Current | $I_{OH}$ | All Output | | $V_{CC}$=5V，$V_{OH}=0.9V_{CC}$ | 6 | 12 | – | mA |
| | | | | $V_{CC}$=3V，$V_{OH}=0.9V_{CC}$ | 2.5 | 5 | – | |
| I/O Port Sink Current | $I_{OL}$ | All Output | With High Sink | $V_{CC}$=5V，$V_{OL}=0.1V_{CC}$ | 40 | 80 | – | |
| | | | | $V_{CC}$=3V，$V_{OL}=0.1V_{CC}$ | 19 | 38 | – | |
| | | | Without High Sink | $V_{CC}$=5V，$V_{OL}=0.1V_{CC}$ | 21 | 42 | – | |
| | | | | $V_{CC}$=3V，$V_{OL}=0.1V_{CC}$ | 9 | 18 | – | |
| Supply Current | $I_{DD}$ | Fast mode $V_{CC}$=5V | | FRC= 16.5888 MHz | – | 5.4 | – | mA |
| | | | | FRC= 8.2994 MHz | – | 3.7 | – | |
| | | Fast mode $V_{CC}$=3V | | FRC= 16.5888 MHz | – | 3.0 | – | |
| | | | | FRC= 8.2994 MHz | – | 2.1 | – | |
| | | Slow mode | | SRC，$V_{CC}$=5V | – | 1.1 | – | |
| | | | | SRC，$V_{CC}$=3V | – | 0.75 | – | |
| | | Idle mode (PWRSAV=0) | | SRC，$V_{CC}$=5V | – | 64 | – | μA |
| | | | | SRC，$V_{CC}$=3V | – | 42 | – | |
| | | Idle mode (PWRSAV=1) | | SRC，$V_{CC}$=5V | – | 11 | – | |
| | | | | SRC，$V_{CC}$=3V | – | 4 | – | |
| | | Halt mode (PWRSAV=1) | | $V_{CC}$=5V | – | 7.3 | – | |
| | | | | $V_{CC}$=3V | – | 2.6 | – | |
| | | Stop mode | | $V_{CC}$=5V | – | 0.6 | – | |
| | | | | $V_{CC}$=3V | – | 0.47 | – | |
| Pull-Up Resistor | $R_{PU}$ | $V_{IN}=V_{CC}$ | | $V_{CC}$=5V | – | 33 | – | KΩ |
| | | | | $V_{CC}$=3V | – | 33 | – | |

## 3. Clock Timing

| Parameter | Symbol | Conditions | Min | Typ | Max |
|---|---|---|---|---|---|
| FRC Frequency | 25℃，$V_{CC}$=4.5V | –1% | 16.5888 | +1% | MHz |
| | 0℃ ~ 105℃，$V_{CC}$=4.5V | –1.5% | 16.5888 | +1.5% | |
| | 0℃ ~ 105℃，$V_{CC}$=3.0 ~ 5.5V | –3.5% | 16.5888 | +3.5% | |
| SRC Frequency | $V_{CC}$=5V | – | 62 | – | KHz |
| | $V_{CC}$=3V | – | 56 | – | |

## 4. Reset Timing Characteristics ($T_A$=-40 ℃ ~ 105 ℃)

| Parameter | Symbol | Conditions | Min | Typ | Max |
|---|---|---|---|---|---|
| RESET Input Low width | Input $V_{CC}$=5V ± 10 % | 30 | – | – | μs |
| WDT wake up time | $V_{CC}$=5V，WDTPSC=11 | – | 66 | – | ms |
| | $V_{CC}$=3V，WDTPSC=11 | – | 73 | – | |
| CPU start up time | $V_{CC}$ = 5 V | – | 30 | – | ms |

## 5. LVR Electrical Characteristics ($T_A$= 25 ℃)

| Parameter | Symbol | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| LVR Reference Voltage | $V_{LVR}$ | $T_A$=25°C | – | 3.95 | – | V |
| | | | – | 3.80 | – | |
| | | | – | 3.65 | – | |
| | | | – | 3.50 | – | |
| | | | – | 3.35 | – | |
| | | | – | 3.20 | – | |
| | | | – | 3.05 | – | |
| | | | – | 2.90 | – | |
| | | | – | 2.75 | – | |
| | | | – | 2.60 | – | |
| | | | – | 2.45 | – | |
| | | | – | 2.30 | – | |
| | | | – | 2.15 | – | |
| | | | – | 2.00 | – | |
| | | | – | 1.85 | – | |
| | | | – | 1.70 | – | |
| LVR Hysteresis Window | $V_{HYS\_LVR}$ | $T_A$ = 25℃ | – | 20 | – | mV |
| Low Voltage Detection time | $t_{LVR}$ | $T_A$=25°C | 100 | – | – | μs |

## 6. LVD Electrical Characteristics ($T_A$= 25 ℃)

| Parameter | Symbol | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| LVD Reference Voltage | $V_{LVD}$ | $T_A$=25°C | – | 4.10 | – | V |
| | | | – | 3.95 | – | |
| | | | – | 3.80 | – | |
| | | | – | 3.65 | – | |
| | | | – | 3.50 | – | |
| | | | – | 3.35 | – | |
| | | | – | 3.20 | – | |
| | | | – | 3.05 | – | |
| | | | – | 2.90 | – | |
| | | | – | 2.75 | – | |
| | | | – | 2.60 | – | |
| | | | – | 2.45 | – | |
| | | | – | 2.30 | – | |
| | | | – | 2.15 | – | |
| | | | – | 2.00 | – | |
| | | | – | 1.85 | – | |
| LVD Hysteresis Window | $V_{HYS\_LVD}$ | LVDHYS = 0 | – | 0 | – | mV |
| | | LVDHYS = 1 | – | 50 | – | |
| Low Voltage Detection time | $t_{LVR}$ | $T_A$=25°C | 100 | – | – | μs |

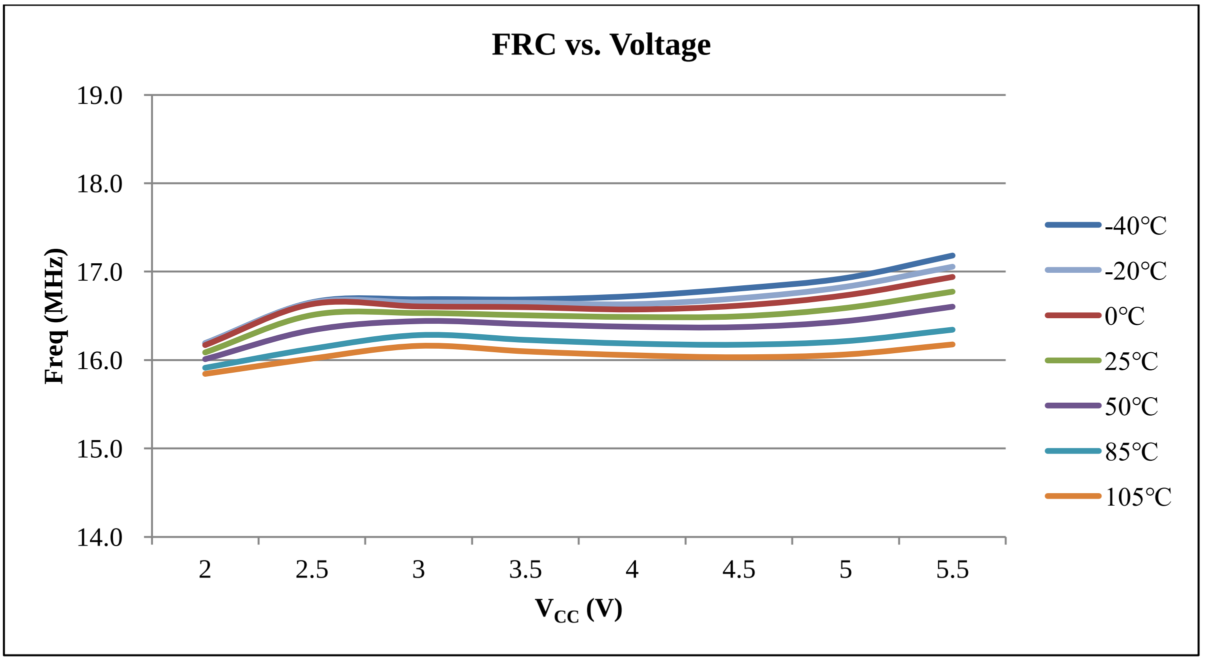
## 7. ADC Electrical Characteristics ($T_A$=25 ℃，$V_{CC}$=3.0V ~ 5.5V，$V_{SS}$= 0V)

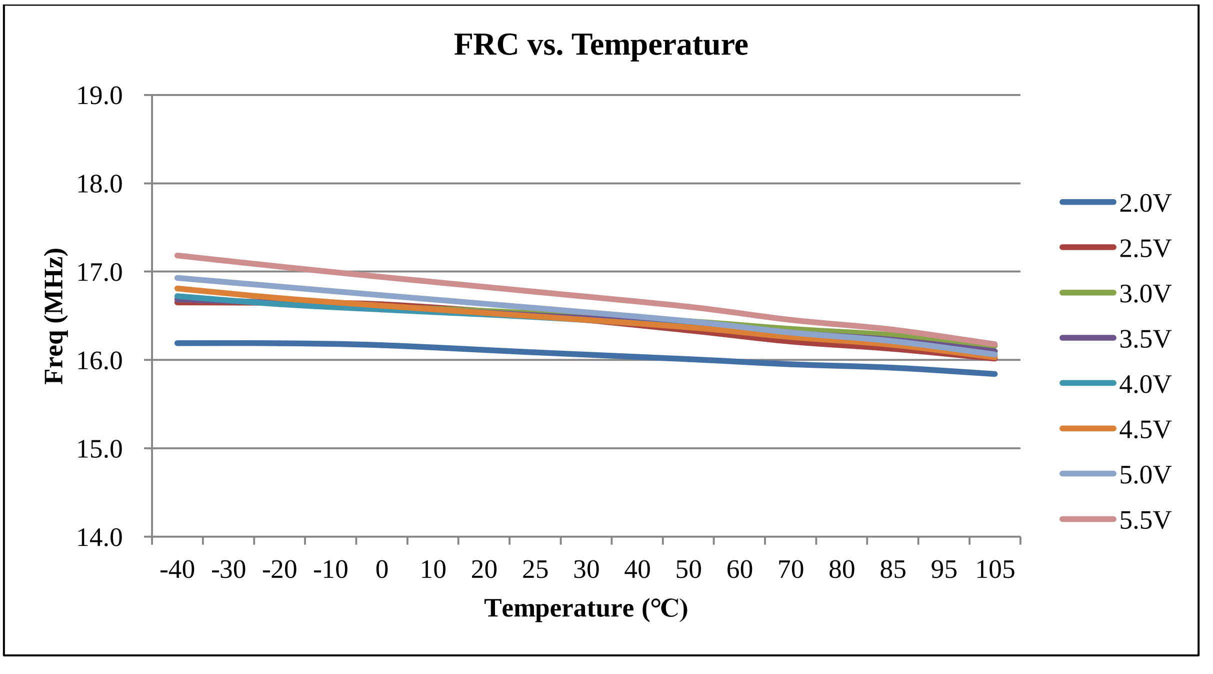
| Parameter | Conditions | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Total Accuracy | $V_{CC}$=5.12 V，$V_{SS}$=0V | | – | ±2.5 | ±4 | LSB |
| Integral Non-Linearity | | | – | ±3.2 | ±5 | |
| Max Input Clock ($f_{ADC}$) | Source impedance (Rs < 5KΩ) | | – | – | 4 | MHz |
| | Source impedance (Rs < 10KΩ) | | – | – | 2 | |
| | Source impedance (Rs < 25KΩ) | | – | – | 1 | |
| | Source is $V_{BG}$ (ADCHS=11100b) | | – | – | $F_{SYSCLK}$/4 | |
| Conversion Time | $F_{ADC}$ = 1MHz | | – | 21 | – | μs |
| BandGap Voltage Reference ($V_{BG}$) | – | $V_{CC}$=2.5V~5.5V 25 ℃ | -1.5% | 1.18 | +1.5% | V |
| | | $V_{CC}$=2.5V~5.5V -40 ℃~105 ℃ | -1.8% | 1.18 | +1.8% | |
| ADC Reference Voltage ($V_{REF}$) | ADCVREFS=1 ADCVBGS=1 | $V_{CC}$=3V~5.5V 25 ℃ | -1.7% | 2.0 | +1.7% | |
| | | $V_{CC}$=2.8V~5.5V -40 ℃~105 ℃ | -2.3% | 2.0 | +2.3% | |
| Vcc/4 Reference Voltage ($V_{1/4}$) | – | $V_{CC}$=5V，25 ℃ | -0.8% | 1.252 | +0.8% | |
| | | $V_{CC}$=3.6V，25 ℃ | -0.8% | 0.902 | +0.8% | |
| Input Voltage | – | | $V_{SS}$ | – | $V_{CC}$ | |

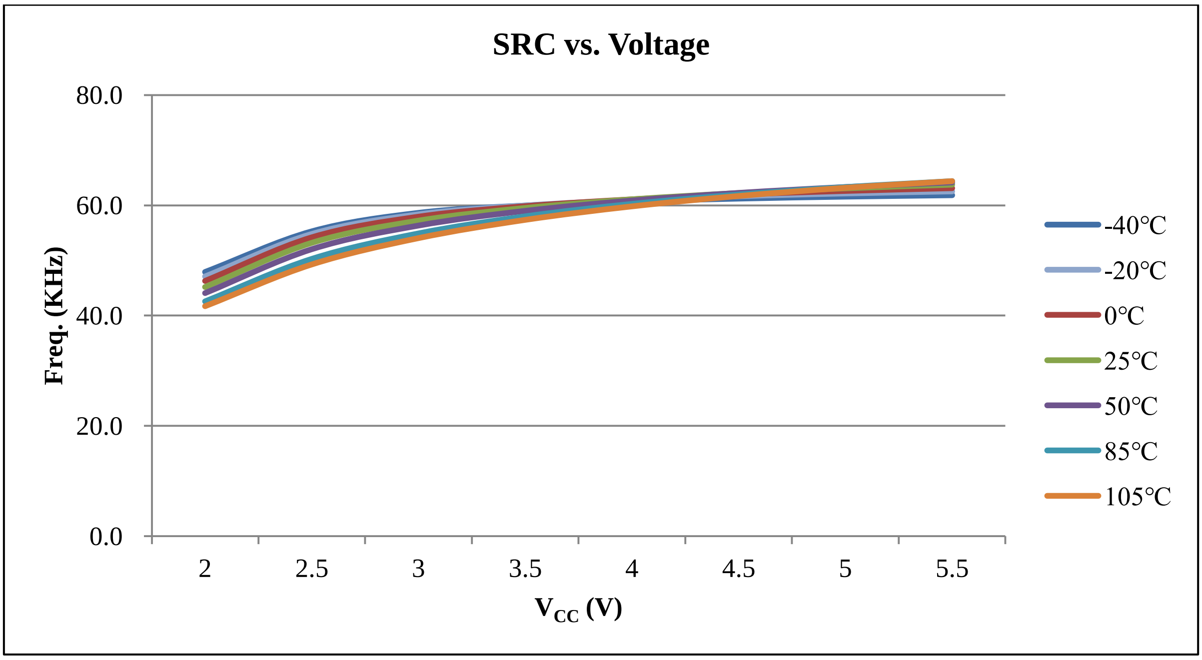
## 8. CMP Characteristics ($T_A$=25 ℃，$V_{CC}$=3.0V ~ 5.5V，$V_{SS}$= 0V)

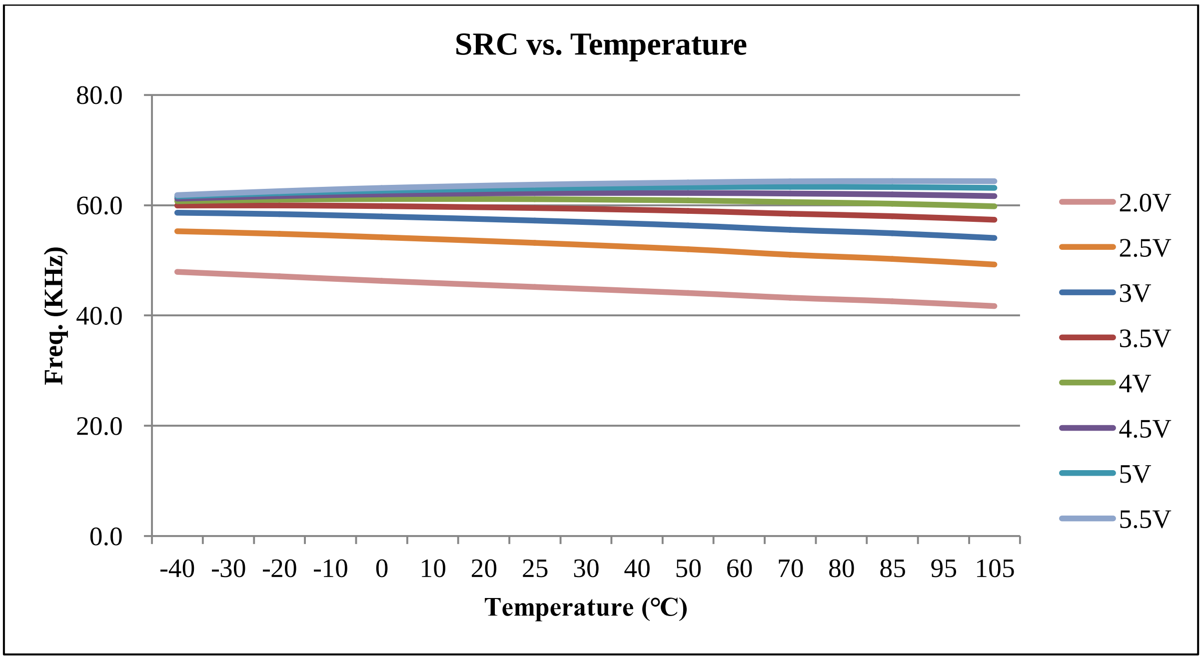
| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Power supply $V_{CC}$ | – | 2.2 | – | 5.5 | V |
| Quiescent current | $V_{CC}$ = 5.0V | – | 100 | – | µA |
| DAC current | $V_{CC}$ = 5.0V | 60 | – | 220 | µA |
| $V_{OS\_CMP}$ | $V_{CC}$ = 5.0V | -15 | – | 15 | mV |
| $V_{CM\_CMP}$ | $V_{CC}$ = 5.0V | 0 | – | $V_{CC}$-0.5 | V |
| $V_{HYS\_CMP}$ | $V_{CC}$ = 5.0V | 20 | 30 | 40 | mV |

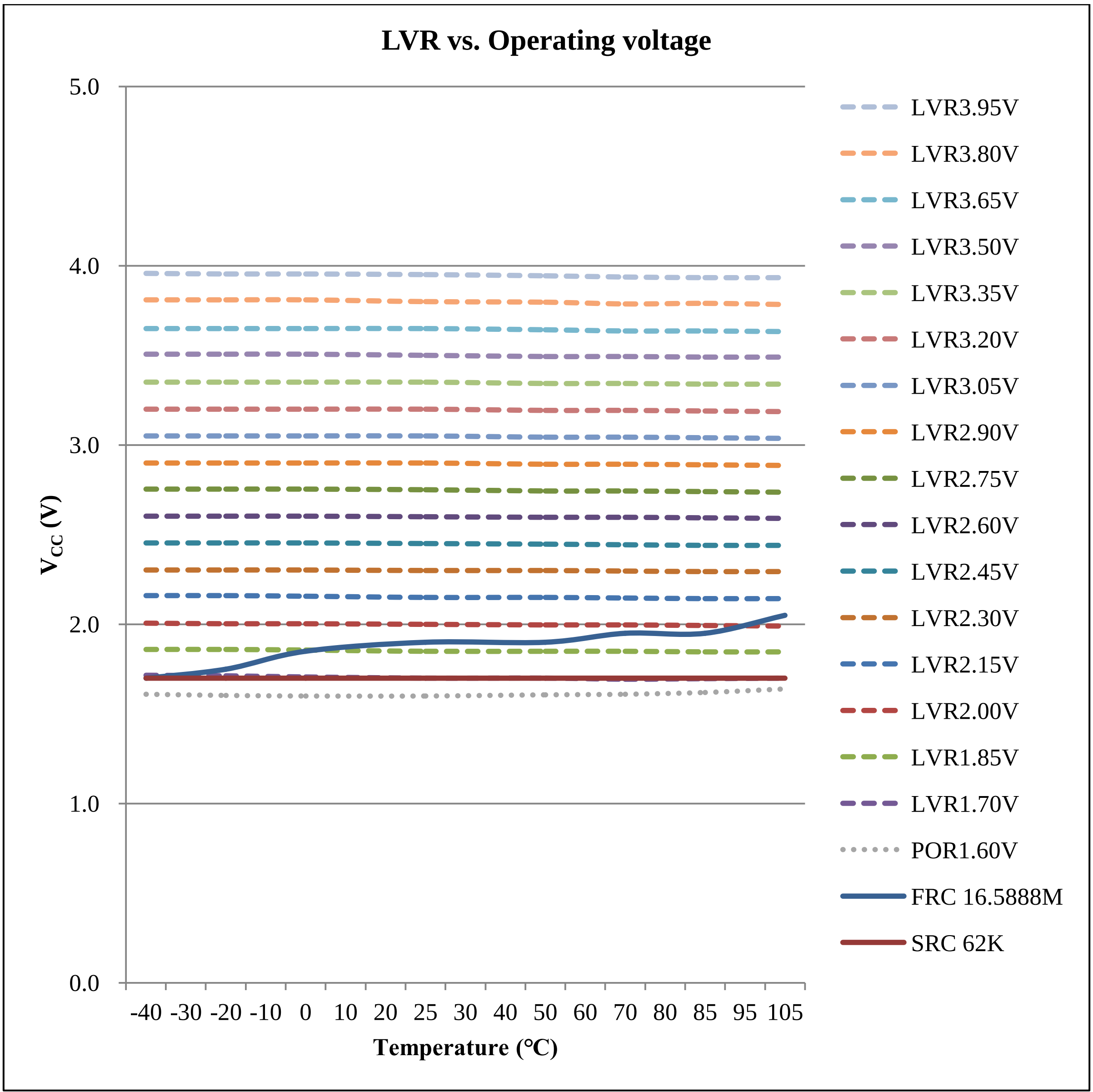
## 9. Characteristic Graphs



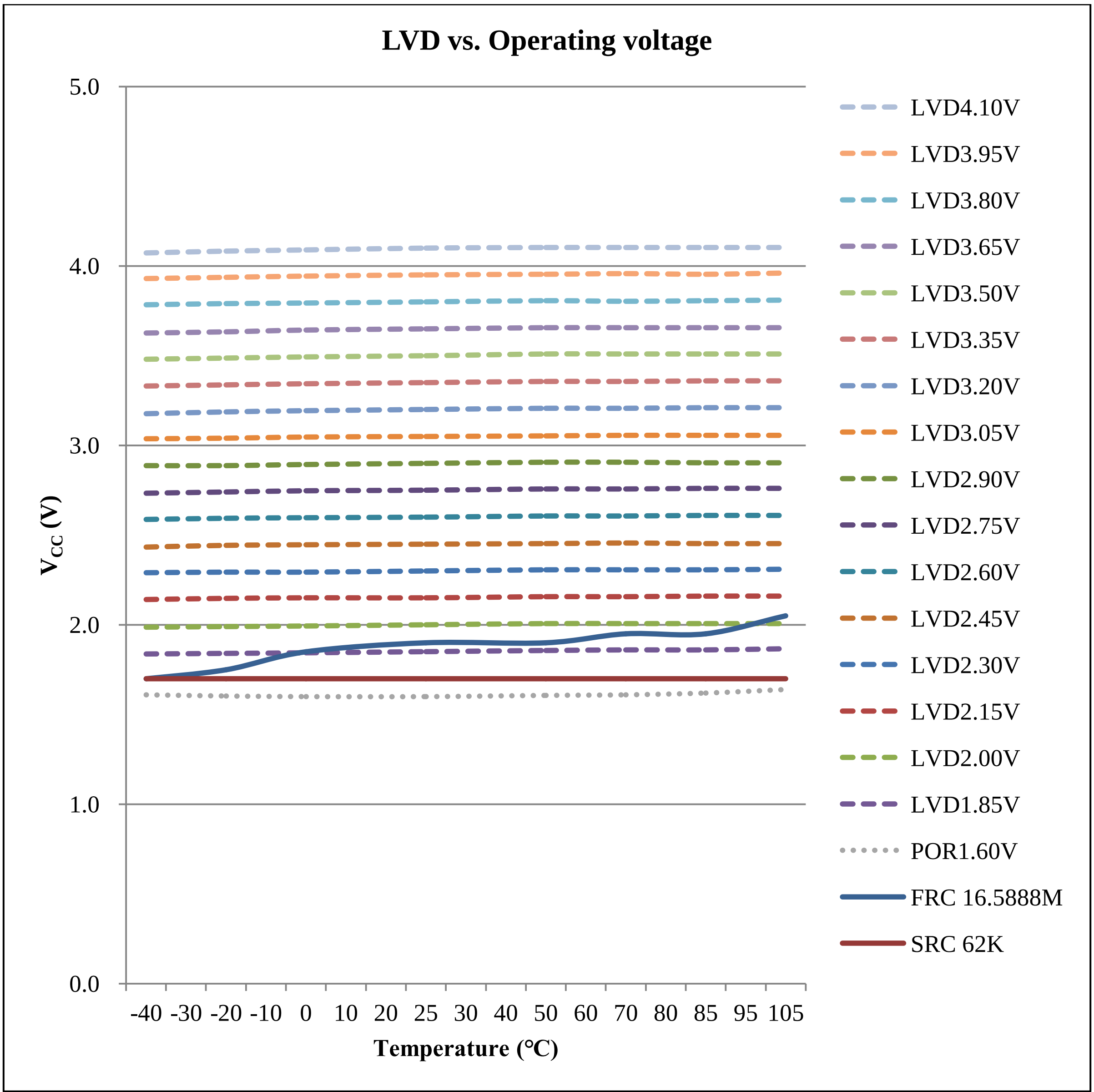




SRC vs. Voltage
80.0
60.0
40.0
20.0
0.0
Freq. (KHz)
2
2.5
3
3.5
4
4.5
5
5.5
VCC (V)
-40℃
-20℃
0℃
25℃
50℃
85℃
105℃

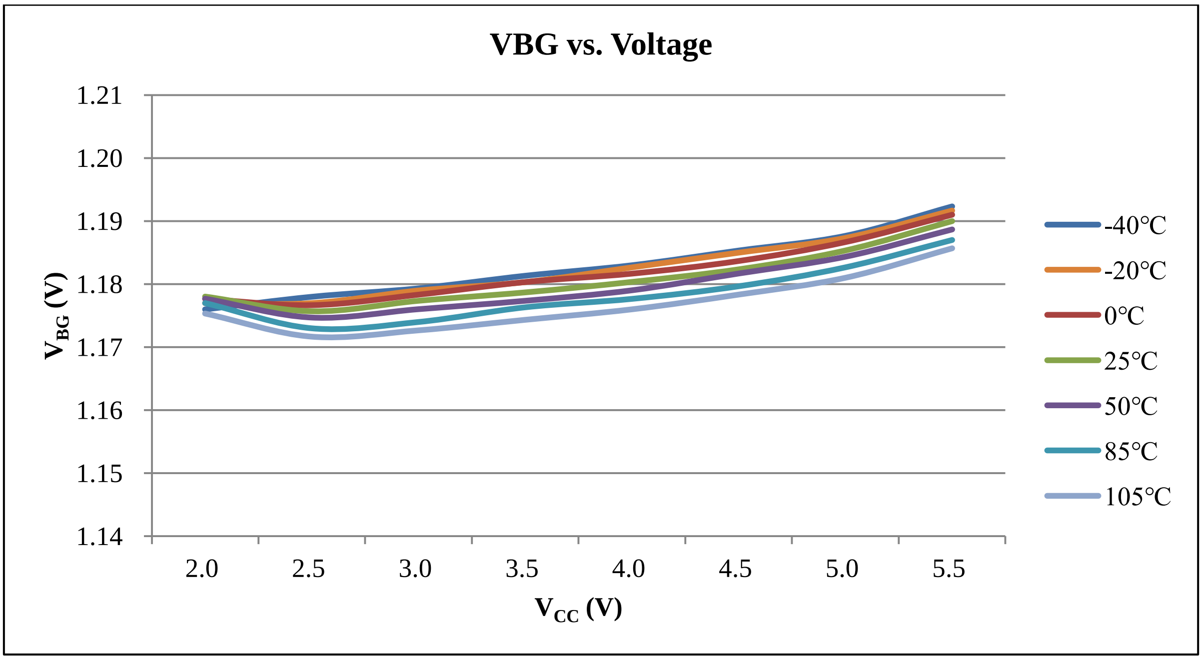

SRC vs. Temperature
80.0
60.0
40.0
20.0
0.0
Freq. (KHz)
-40
-30
-20
-10
0
10
20
25
30
40
50
60
70
80
85
95
105
Temperature (℃)
2.0V
2.5V
3V
3.5V
4V
4.5V
5V
5.5V




***Note:*** *POR: Power on reset. VCC should greater than POR when power on. Due to the variation of the manufacturing process, the POR value will be slightly different between different chips.*




***Note:*** *POR: Power on reset. VCC should greater than POR when power on. Due to the variation of the manufacturing process, the POR value will be slightly different between different chips.*


VBG vs. Voltage
1.21
1.20
1.19
1.18
1.17
1.16
1.15
1.14
V_BG (V)
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
V_CC (V)
-40°C
-20°C
0°C
25°C
50°C
85°C
105°C

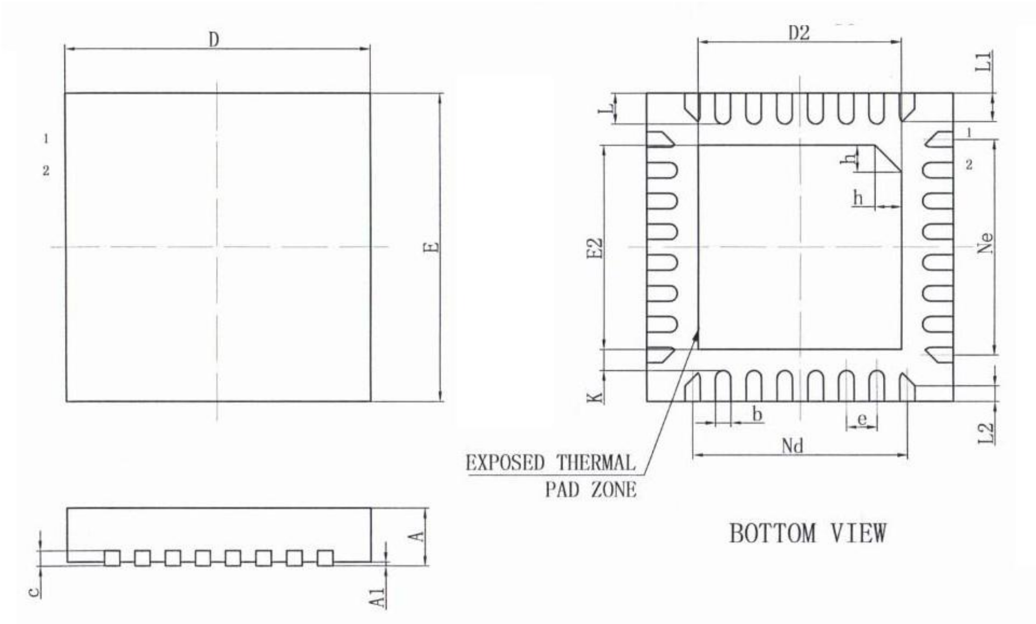
# Package and Dice Information

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

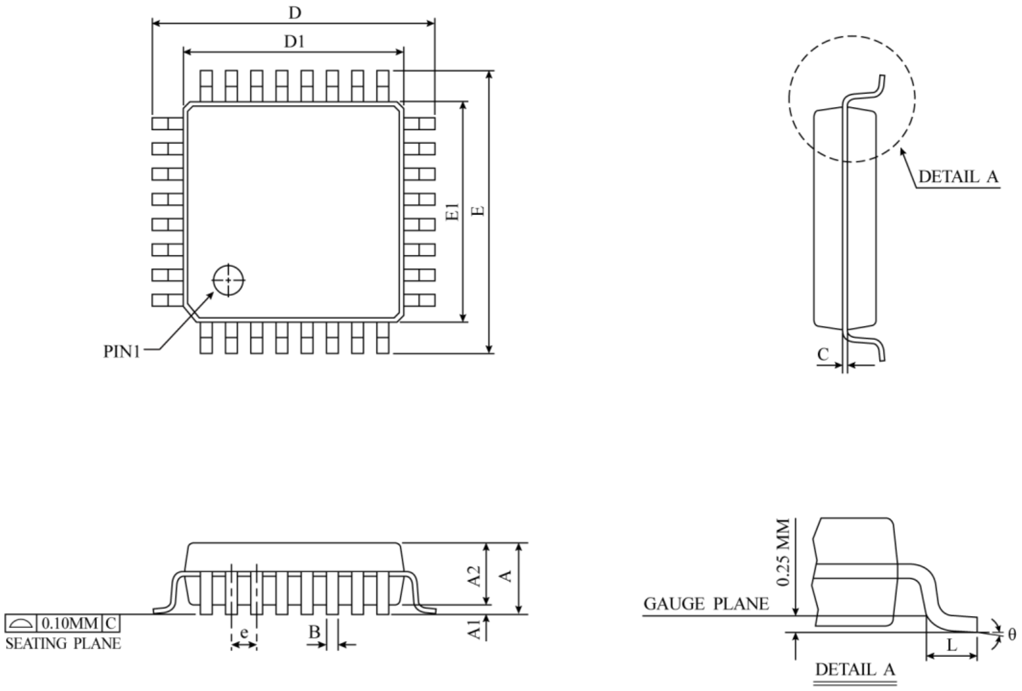
| Ordering number | Package |
|---|---|
| TM52F58646P | QFN 32-pin (4x4x0.75-0.4 mm) |
| TM52F58646L1 | LQFP 32-pin (7x7x1.4 mm) |
| TM52F58644Q | QFN 24-pin (3x3x0.75-0.35 mm) |
| TM52F58645E1 | SSOP 28-pin (150 mil) |
| TM52F58644E1 | SSOP 24-pin (150 mil) |
| TM52F58643T1 | TSSOP 20-pin (173 mil) |

**QFN-32 ( 4x4x0.75-0.4mm ) Package Dimension**




| SYMBOL | DIMENSION IN MM | | | DIMENSION IN INCH | | |
|---|---|---|---|---|---|---|
| | MIN | NOM | MAX | MIN | NOM | MAX |
| A | 0.7 | 0.75 | 0.8 | 0.028 | 0.030 | 0.031 |
| A1 | 0 | 0.02 | 0.05 | 0.000 | 0.001 | 0.002 |
| b | 0.15 | 0.20 | 0.25 | 0.006 | 0.008 | 0.010 |
| c | 0.18 | 0.20 | 0.25 | 0.007 | 0.008 | 0.010 |
| D | 3.90 | 4.00 | 4.10 | 0.154 | 0.157 | 0.161 |
| D2 | 2.60 | 2.65 | 2.70 | 0.102 | 0.104 | 0.106 |
| e | 0.40 BSC | | | 0.016 BSC | | |
| Nd | 2.80 BSC | | | 0.110 BSC | | |
| E | 3.90 | 4.00 | 4.10 | 0.154 | 0.157 | 0.161 |
| E2 | 2.60 | 2.65 | 2.70 | 0.102 | 0.104 | 0.106 |
| Ne | 2.80 BSC | | | 0.110 BSC | | |
| K | 0.20 | - | - | 0.008 | - | - |
| L | 0.35 | 0.40 | 0.45 | 0.014 | 0.016 | 0.018 |
| L1 | 0.30 | 0.35 | 0.40 | 0.012 | 0.014 | 0.016 |
| L2 | 0.15 | 0.20 | 0.25 | 0.006 | 0.008 | 0.010 |
| h | 0.30 | 0.35 | 0.40 | 0.012 | 0.014 | 0.016 |
| JEDEC | M0-220 | | | | | |

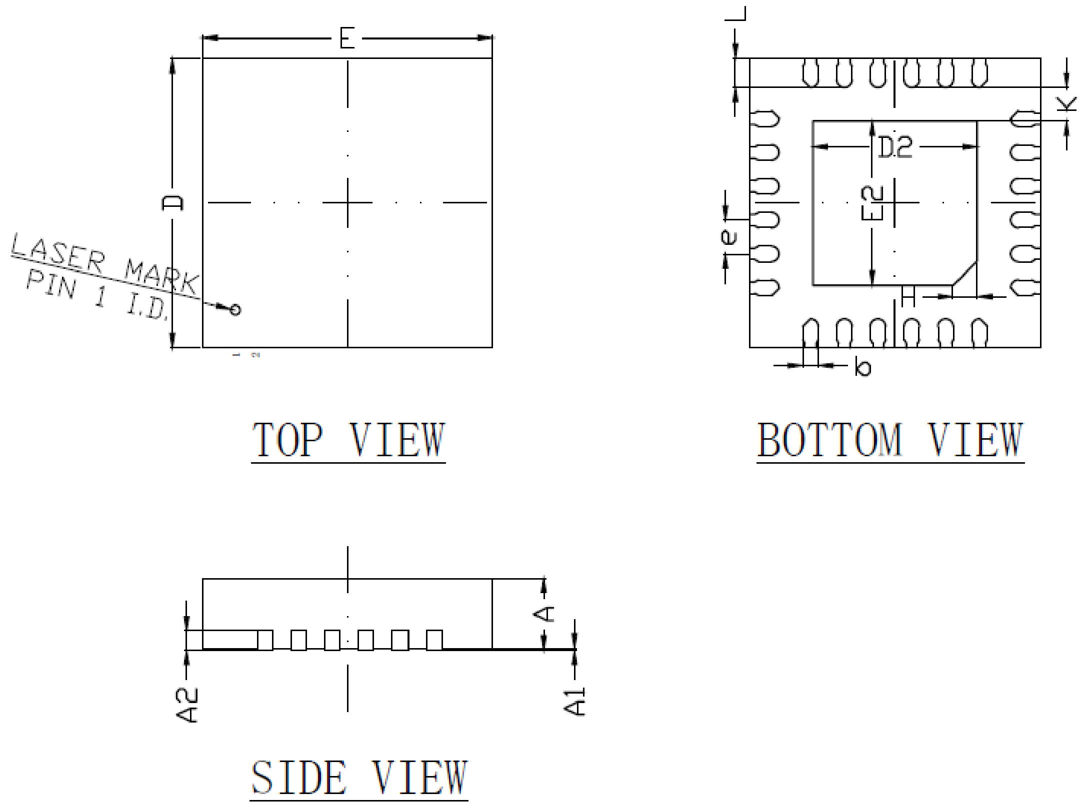
## LQFP-32 ( 7×7x1.4mm ) Package Dimension




| SYMBOL | DIMENSION IN MM | | | DIMENSION IN INCH | | |
|---|---|---|---|---|---|---|
| | MIN | NOM | MAX | MIN | NOM | MAX |
| A | - | - | 1.60 | - | - | 0.063 |
| A1 | 0.05 | 0.10 | 0.15 | 0.001 | 0.004 | 0.006 |
| A2 | 1.35 | 1.40 | 1.45 | 0.053 | 0.055 | 0.057 |
| B | 0.30 | 0.38 | 0.45 | 0.012 | 0.015 | 0.018 |
| C | 0.09 | 0.09 | 0.20 | 0.004 | 0.006 | 0.008 |
| D | 9.00 BSC | | | 0.354 BSC | | |
| D1 | 7.00 BSC | | | 0.276 BSC | | |
| E | 9.00 BSC | | | 0.354 BSC | | |
| E1 | 7.00 BSC | | | 0.276 BSC | | |
| e | 0.80 BSC | | | 0.031 BSC | | |
| L | 0.45 | 0.60 | 0.75 | 0.018 | 0.027 | 0.035 |
| θ | 0° | 3.5° | 7° | 0° | 3.5° | 7° |
| JEDEC | MS-026 (BBA) | | | | | |

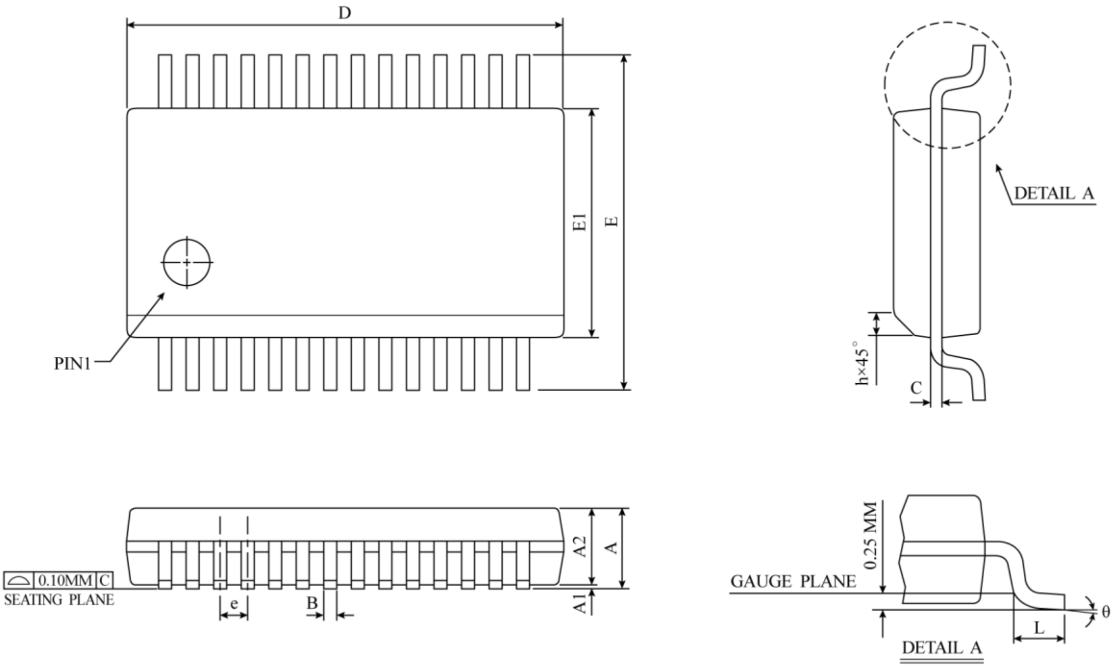
△A * NOTES : DIMENSION " D1 " AND " E1 " DO NOT INCLUDE MOLD PROTRUSIONS. ALLOWABLE PROTRUSIONS IS 0.25 mm PER SIDE. " D1 " AND " E1 " ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMACH.

**QFN 24 (3*3*0.75-0.35mm) Package Dimension**




| SYMBOL | MILLIMETER | | |
|---|---|---|---|
| | MIN | NOM | MAX |
| A | 0.70 | 0.75 | 0.80 |
| A1 | -- | 0.02 | 0.05 |
| A2 | 0.20REF | | |
| b | 0.11 | 0.16 | 0.21 |
| D | 2.90 | 3.00 | 3.10 |
| E | 2.90 | 3.00 | 3.10 |
| D2 | 1.60 | 1.70 | 1.80 |
| E2 | 1.60 | 1.70 | 1.80 |
| e | 0.35BSC | | |
| H | 0.20 | 0.25 | 0.30 |
| K | 0.35REF | | |
| L | 0.25 | 0.30 | 0.35 |

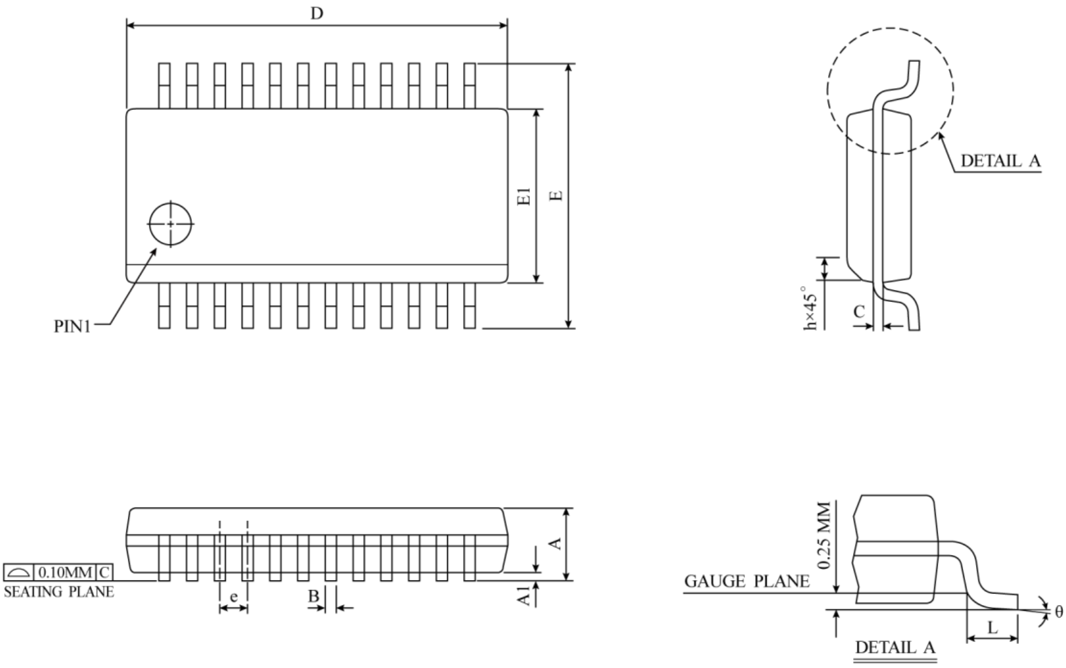
## SSOP-28 ( 150mil ) Package Dimension




| SYMBOL | DIMENSION IN MM | | | DIMENSION IN INCH | | |
|---|---|---|---|---|---|---|
| | MIN | NOM | MAX | MIN | NOM | MAX |
| A | 1.50 | 1.65 | 1.80 | 0.06 | 0.06 | 0.07 |
| A1 | 0.102 | 0.176 | 0.249 | 0.004 | 0.007 | 0.010 |
| A2 | 1.40 | 1.475 | 1.55 | 0.06 | 0.06 | 0.06 |
| B | 0.20 | 0.25 | 0.30 | 0.01 | 0.01 | 0.01 |
| C | 0.2TYP | | | 0.008TYP | | |
| e | 0.635TYP | | | 0.025TYP | | |
| D | 9.804 | 9.881 | 9.957 | 0.386 | 0.389 | 0.392 |
| E | 5.842 | 6.020 | 6.198 | 0.230 | 0.237 | 0.244 |
| E1 | 3.86 | 3.929 | 3.998 | 0.152 | 0.155 | 0.157 |
| L | 0.406 | 0.648 | 0.889 | 0.016 | 0.026 | 0.035 |
| θ | 0° | 4° | 8° | 0° | 4° | 8° |
| JEDEC | M0-137(AF) | | | | | |

B *NOTES：DIMENSION “D” DOES NOT INCLUDE MOLD PROTRUSIONS OR GATE BURRS.
MOLD PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED 0.006 INCH PER SIDE.

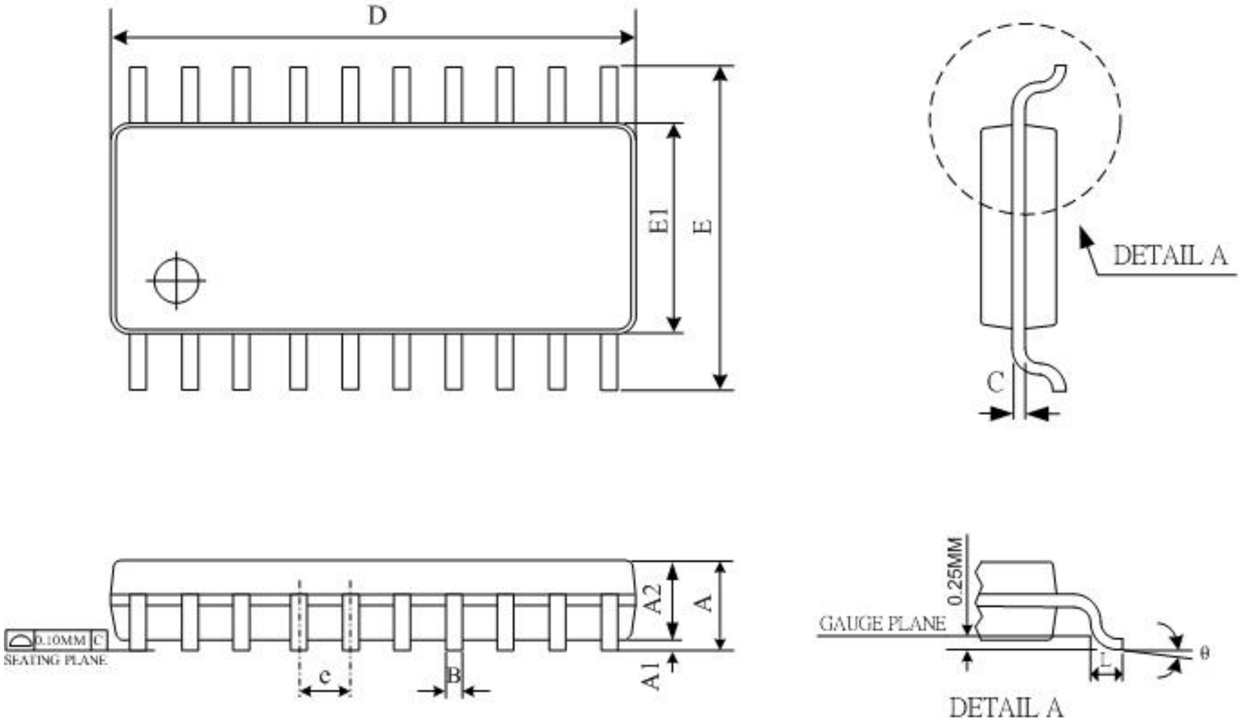
## SSOP-24 ( 150mil ) Package Dimension




| 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) | | | | | |

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

## TSSOP-20 引脚（173mil）包装尺寸




| 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.