十速

TM52eF0C86/85

DATA SHEET

Rev 0.91

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

PRECAUTIONS

1. Chip cannot enter Stop/Halt Mode if INTn pin is low and wakeup is enabled. (INTn=0 and EXn=1, n=0~1)
2. If PCIF (P0IF~P5IF) is high, the chip cannot enter stop/Halt mode.
3. Before IAP Write, the user should disable the LVR first and turn on LVR after IAP writing is completed.

AMENDMENT HISTORY

Version	Date	Description
V0.90	Oct, 2022	New release.
V0.91	Oct, 2022	1. Operating temperature. 2. IAP write times. 3. UART description.

CONTENTS

TM52 eF0xxx FAMILY

Common Feature

CPU	MTP/Flash Program memory	RAM bytes	Dual Clock	Operation Mode	Timer0 Timer1 Timer2	UART	Real-time Timer3	LVD	LVR
Fast 8051 (2T)	4K~64K with IAP, ISP, ICP	256 ~ 4096	SXT SRC FXT FRC	Fast Slow Idle Stop Halt	8051 Standard		15-bit	16 level	8 level

Note: *IAP, ISP only for Flash type program memory*

Family Members Features

P/N	Program Memory	RAM Bytes	IO Pin	PWM	SAR ADC	Touch Key	LCD	LED	Interface
TM52-eF1716 TM52-eF1732	Flash 16KB 32KB	1280	30	16-bit x3 8-bit x3	12-bit 16-ch	20-ch	8com	BiD 4Cx6S	SPI UARTx2 I^2C
TM52-eF1374 TM52-eF1375	Flash 16KB 32KB	1280	26	16-bit x3	12-bit 16-ch	20-ch	8com	BiD 4Cx6S DMX 8x8	SPI UARTx2 I^2C
TM52-eF0C85 TM52-eF0C86	Flash 32KB 64KB	4352	42	16-bit x9	12-bit 44-ch	21-ch x 2	4Cx20S ~ 8Cx16S	MX 8x8 DMX 7x8	SPI UARTx3 I^2C

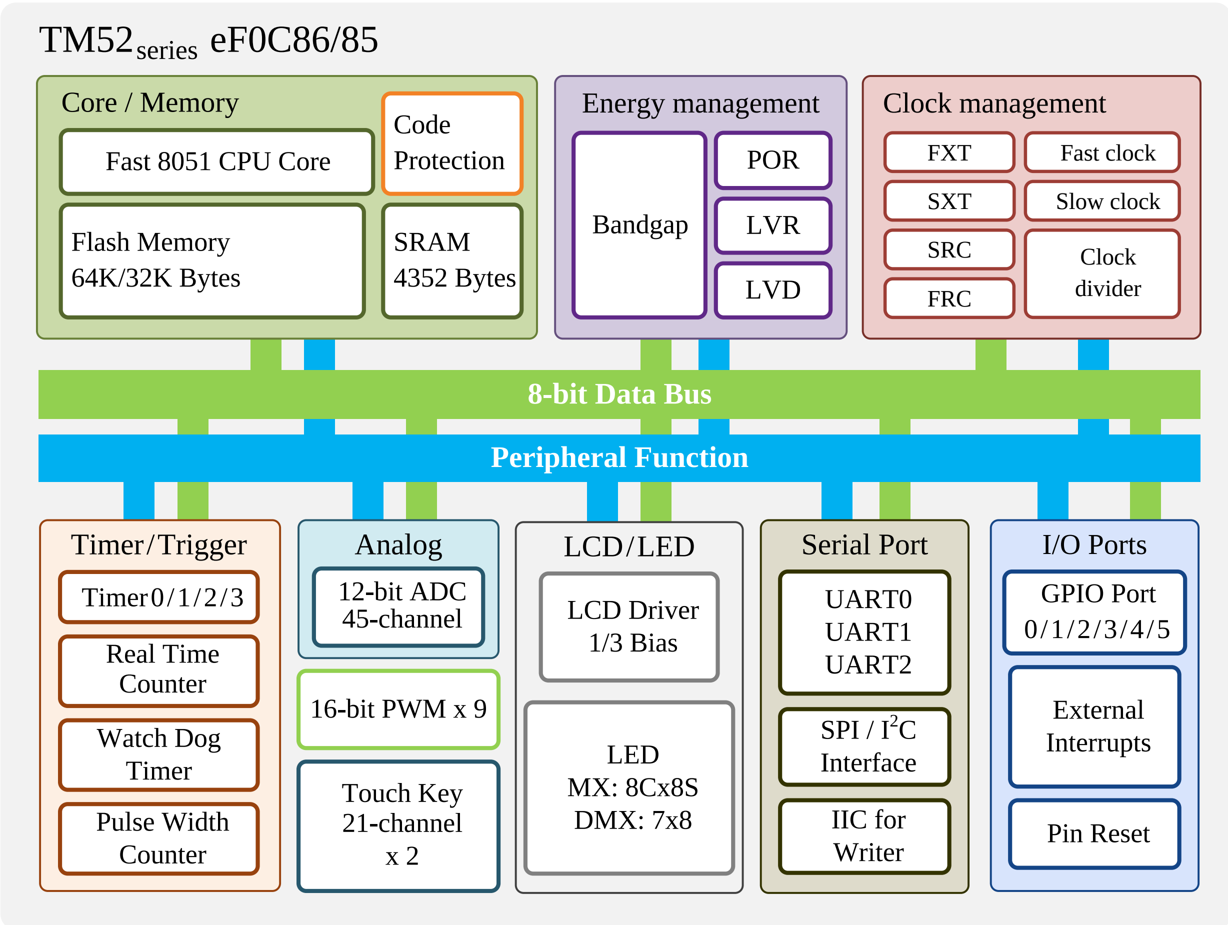
P/N	Operation Voltage	Operation Current					Max. System Clock (Hz)			
		Fast FRC	Slow SRC	Idle SRC	Stop	Halt	SXT	SRC	FXT	FRC
TM52-F1716 TM52-F1732	2.5~5.5V	3.5mA	0.18mA	0.15 mA	7uA@5V 1.4uA@3V	11uA@5V 4uA@3V	32K	80K	16M	14.7456M
TM52-eF1374 TM52-eF1375	2.3~5.5V	4mA	0.22mA	0.2mA	10uA@5V 4uA@3V	13uA@5V 6uA@3V	32K	80K	18M	18.432M
TM52-eF0C85 TM52-eF0C86	2.3~5.5V	3.5mA	0.2mA	0.18mA	11uA@5V 4uA@3V	14uA@5V 6uA@3V	32K	80K	16M	18.432M

GENERAL DESCRIPTION

TM52 **series** **eF0C86/85** 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-eF0C86/85** provides improved performance, lower cost and fast time-to-market by integrating features on the chip, including 64K/32K Bytes Flash program memory, 4352 Bytes SRAM, Low Voltage Reset (LVR), Low Voltage Detector (LVD), dual clock power saving operation mode, 8051 standard UART and Timer0/1/2, real time clock Timer3, LCD/LED driver, 9 set 16-bit PWMs, 45 channels 12-bit A/D Convertor, 2 group of 21 channels Touch Key, I^2C/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.

BLOCK DIAGRAM



FEATURES

1. **Standard 8051 Instruction set, fast machine cycle**
   - Executes instructions six times faster than the standard 8051.
2. **Flash Program Memory**
   - 64K Bytes (TM52eF0C86)
   - 32K Bytes (TM52eF0C85)
   - Support IAP "In Application Programming"
   - Code Protection Capability
   - 100K erase times at least
   - 10 years data retention at least
3. **Total 4352 Bytes SRAM (IRAM + XRAM)**
   - 256 Bytes IRAM in the 8051 internal data memory area
   - 4096 Bytes XRAM in the 8051 external data memory area (accessed by MOVX Instruction)
4. **Four System Clock type selections**
   - Fast clock from 1~16MHz Crystal (FXT)
   - Fast clock from Internal RC (FRC, 18.432 MHz)
   - Slow clock from 32768Hz Crystal (SXT)
   - Slow clock from Internal RC (SRC,80 KHz)
   - System Clock can be divided by 1/2/4/16 option
5. **8051 Standard Timer – Timer0/1/2**
   - 16-bit Timer0, also supports T0O clock output for Buzzer application
   - 16-bit Timer1
   - 16-bit Timer2, also supports T2O clock output for Buzzer application
6. **15-bit Timer3**
   - Clock source can be selected as slow clock, FRC/512, slow clock/2 or FRC/1024
   - Interrupt period can be clock divided by 65536/16384/4096/1024 option
7. **UARTs**
   - UART0, 8051 standard UART
   - UART1, supports only mode1 and mode3
   - UART2, supports only mode1 and mode3

8. Nine 16 bits PWMs with period-adjustment

- Three 16 bits PWMs with independent period-adjustment (PWM0~PWM2)
- Six 16 bits PWMs with sharing period-adjustment (PWM30~PWM35)
- PWM2 with dead zone control

9. SPI Interface

- Master or Slave mode selectable
- Programmable transmit bit rate
- Serial clock phase and polarity options
- MSB-first or LSB-first selectable

10. I^2C interface (Master / Slave)

11. 2 group of 21-Channel Touch Key (FTK)

12. 12-bit ADC with 42 channels External Pin Input and 2 channels Internal Reference Voltage

- Internal Reference Voltage: VBG 1.22V @V_{CC}=5V~3V, 25℃
- Internal Reference Voltage: 1/4V_{CC}

13. LCD Driver

- 4x20 ~ 8x16 LCD driver
- 1/3 LCD Bias

14. LED Controller/Driver

- Matrix mode (MX): 8*8, 16 pins up to 64 dots
- Dot matrix mode (DMX): 7*8、6*7、5*6、4*5, 8 pins up to 56 dots

15. 14 Sources, 4-level priority Interrupt

- Timer0/Timer1/Timer2/Timer3 Interrupt
- INT0~INT1 pin low level or falling edge Interrupt
- Port0/1/2/3/4/5 Pin Change Interrupt
- UART0/UART1/UART2 TX/RX Interrupt
- ADC/Touch Key Interrupt
- SPI Interrupt
- I^2C interrupt
- PWM0/PWM1/PWM2/PWM3 interrupt

16. Pin Interrupt can Wake up CPU from Power-Down (Stop/Halt) mode

- Each pin can be defined as Interrupt & Wake-up pin (by pin change)

17. Max. 42 Programmable I/O pins

- CMOS Output
- Pseudo-Open-Drain, or Open-Drain Output
- Schmitt Trigger Input
- Pin Pull-up can be Enabled or Disabled
- All pin with High sink (60mA@V_{CC}=5V，V_{OL}=0.1V_{CC})
- COM pins have high sink current in LED mode (120mA@V_{CC}=5V，V_{OL}=0.1V_{CC})

18. Independent RC Oscillating Watch Dog Timer

- 400ms/200ms/100ms/50ms selectable WDT timeout options

19. Five types Reset

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

20. 16-level Low Voltage Detect

- 4.38V/4.3V/4.14V/4.06V/3.9V/3.82V/3.66V/3.58V/
  3.42V/3.34V/3.18V/3.1V/2.94V/2.86V/2.7V/2.62V

21. 8-level Low Voltage Reset

- 3.92V/3.68V/3.44V/3.20V/2.96V/2.72V/2.48V/2.24V

22. Five Power Operation Modes

- Fast/Slow/Idle/Stop/Halt mode

23. Integrated 16-bit Cyclic Redundancy Check function

24. Multiplication and division

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

25. On-chip Debug/ICE interface

- Use P3.0/P3.1, P3.4/P3.5, or P0.2/P0.3 pin
- Share with ICP programming pin

26. Writer interface

- Use P3.0/P3.1

27. Operating Voltage and Current

- V_{CC} =2.3V ~ 5.5V @F_{SYSCLK}=18.432MHz
- Icc<10uA @Vcc=3V, Halt mode, LCD On, 0.5S wakeup (timepiece current)
- I_{CC} =11μA @Stop mode, V_{CC}=5V
- I_{CC} =4μA @Stop mode, V_{CC}=3V
- I_{CC} =180μA @Idle mode, V_{CC}=5V

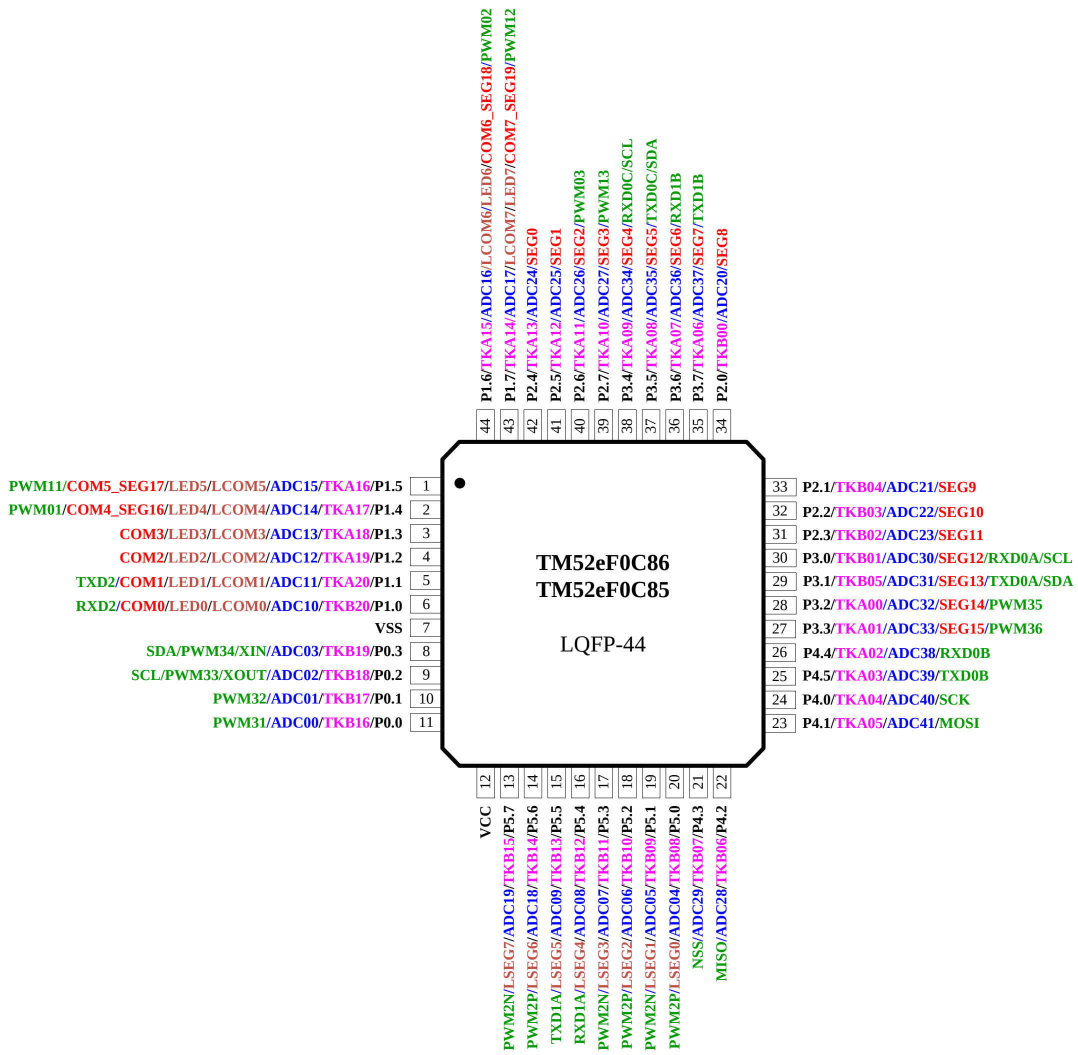
28. Operating Temperature Range

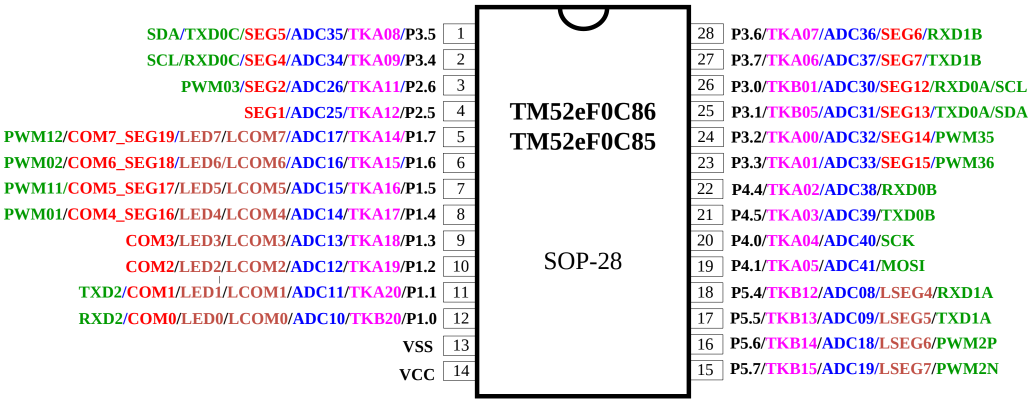
- –40℃ ~ +105℃

29. Package Types

- 44-pin LQFP
- SOP-28

PIN ASSIGNMENT




TM52eF0C86
TM52eF0C85
SOP-28
SDA/TXD0C/SEG5/ADC35/TKA08/P3.5 1
SCL/RXD0C/SEG4/ADC34/TKA09/P3.4 2
PWM03/SEG2/ADC26/TKA11/P2.6 3
SEG1/ADC25/TKA12/P2.5 4
PWM12/COM7_SEG19/LED7/LCOM7/ADC17/TKA14/P1.7 5
PWM02/COM6_SEG18/LED6/LCOM6/ADC16/TKA15/P1.6 6
PWM11/COM5_SEG17/LED5/LCOM5/ADC15/TKA16/P1.5 7
PWM01/COM4_SEG16/LED4/LCOM4/ADC14/TKA17/P1.4 8
COM3/LED3/LCOM3/ADC13/TKA18/P1.3 9
COM2/LED2/LCOM2/ADC12/TKA19/P1.2 10
TXD2/COM1/LED1/LCOM1/ADC11/TKA20/P1.1 11
RXD2/COM0/LED0/LCOM0/ADC10/TKB20/P1.0 12
VSS 13
VCC 14
28 P3.6/TKA07/ADC36/SEG6/RXD1B
27 P3.7/TKA06/ADC37/SEG7/TXD1B
26 P3.0/TKB01/ADC30/SEG12/RXD0A/SCL
25 P3.1/TKB05/ADC31/SEG13/TXD0A/SDA
24 P3.2/TKA00/ADC32/SEG14/PWM35
23 P3.3/TKA01/ADC33/SEG15/PWM36
22 P4.4/TKA02/ADC38/RXD0B
21 P4.5/TKA03/ADC39/TXD0B
20 P4.0/TKA04/ADC40/SCK
19 P4.1/TKA05/ADC41/MOSI
18 P5.4/TKB12/ADC08/LSEG4/RXD1A
17 P5.5/TKB13/ADC09/LSEG5/TXD1A
16 P5.6/TKB14/ADC18/LSEG6/PWM2P
15 P5.7/TKB15/ADC19/LSEG7/PWM2N

PIN DESCRIPTION

Name	In/ Out	Pin Description
P0/P1/P2/P3/P4/P5	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.
INT0, INT1	I	External low level or falling edge Interrupt input, Idle/Stop/Halt mode wake up input.
RXD0A, TXD0A RXD0B, TXD0B RXD0C, TXD0C	I/O	UART0 receive and transmits data pin
RXD1A, TXD1A RXD1B, TXD1B	I/O	UART1 receive and transmits data pin
RXD2, TXD2	I/O	UART2 receive and transmits data pin
T0, T1, T2	I	Timer0, Timer1, Timer2 event count pin input.
T2EX	I	Timer2 external trigger input.
T0O	O	Timer0 overflow divided by 64 output
T2O	O	Timer2 overflow divided by 2 output
VBGO	O	Bandgap voltage output
PWM0 PWM1 PWM2P, PWN2N PWM30~PWM35	O	16 bit PWM output
AD0~AD41	I	ADC input
TKA00~TKA20 TKB00~TKB20	I	Touch Key module A and Touch Key module B input
XCAPA, XCAPB	I	Touch Key module A/B charge collection capacitor connection pin
COM0~COM7	O	LCD COM output
SEG0~SEG19	O	LCD segment output
LCOM0~LCOM7	O	LED matrix mode COM output
LSEG0~LSEG7	O	LED matrix mode segment output
LED0~LED7	O	LED dot matrix mode output
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
SS	I	SPI active low slave select input for slave mode
SCK	I/O	SPI clock output for master or clock input for slave mode
SCL	I/O	I^2C SCL
SDA	I/O	I^2C SDA
RSTn	I	External active low reset input, Pull-up resistor is fixed enable.
XI, XO	–	Crystal/Resonator oscillator connection for System clock (FXT or SXT)
VCC, VSS	P	Power input pin and ground

PIN SUMMERY

Pin #	Pin Name	Type	Initial State	Wake up	Ext. Interrupt	LCD	LED matrix	LED dot matrix	ADC	Touch Key	UART	PWM	SPI	I^2C	Other
LQFP-44															
1	P15	I/O	Hi-Z	●	●	●	●	●	●	●		●			
2	P14	I/O	Hi-Z	●	●	●	●	●	●	●		●			
3	P13	I/O	Hi-Z	●	●	●	●	●	●	●					
4	P13	I/O	Hi-Z	●	●	●	●	●	●	●					
5	P11	I/O	Hi-Z	●	●	●	●	●	●	●	●				
6	P10	I/O	Hi-Z	●	●	●	●	●	●	●	●				T2O
7	VSS	P													
8	P03	I/O	Hi-Z	●	●				●	●		●		●	Crystal
9	P02	I/O	Hi-Z	●	●				●	●		●		●	Crystal
10	P01	I/O	Hi-Z	●	●				●	●		●			
11	P00	I/O	Hi-Z	●	●				●	●		●			
12	VCC	P													
13	P57	I/O	Hi-Z	●	●		●		●	●		●			Reset
14	P56	I/O	Hi-Z	●	●		●		●	●		●			
15	P55	I/O	Hi-Z	●	●		●		●	●	●				
16	P54	I/O	Hi-Z	●	●		●		●	●	●				
17	P53	I/O	Hi-Z	●	●		●		●	●		●			
18	P52	I/O	Hi-Z	●	●		●		●	●		●			
19	P51	I/O	Hi-Z	●	●		●		●	●		●			
20	P50	I/O	Hi-Z	●	●		●		●	●		●			
21	P43	I/O	Hi-Z	●	●				●	●			●		
22	P42	I/O	Hi-Z	●	●				●	●			●		
23	P41	I/O	Hi-Z	●	●				●	●			●		
24	P40	I/O	Hi-Z	●	●				●	●			●		
25	P45	I/O	Hi-Z	●	●				●	●	●				
26	P44	I/O	Hi-Z	●	●				●	●	●				
27	P33	I/O	Hi-Z	●	●	●			●	●		●			INT1
28	P32	I/O	Hi-Z	●	●	●			●	●		●			INT0
29	P31	I/O	Hi-Z	●	●	●			●	●	●			●	
30	P30	I/O	Hi-Z	●	●	●			●	●	●			●	
31	P23	I/O	Hi-Z	●	●	●			●	●					
32	P22	I/O	Hi-Z	●	●	●			●	●					
33	P21	I/O	Hi-Z	●	●	●			●	●					
34	P20	I/O	Hi-Z	●	●	●			●	●					
35	P37	I/O	Hi-Z	●	●	●			●	●	●				
36	P36	I/O	Hi-Z	●	●	●			●	●	●				
37	P35	I/O	Hi-Z	●	●	●			●	●	●			●	
38	P34	I/O	Hi-Z	●	●	●			●	●	●			●	T0O
39	P27	I/O	Hi-Z	●	●	●			●	●		●			
40	P26	I/O	Hi-Z	●	●	●			●	●		●			
41	P25	I/O	Hi-Z	●	●	●			●	●					
42	P24	I/O	Hi-Z	●	●	●			●	●					
43	P17	I/O	Hi-Z	●	●	●	●	●	●	●		●			
44	P16	I/O	Hi-Z	●	●	●	●	●	●	●		●			

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	TKSOCA	ADSOC	–	TKSOCB	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Reset	0	0	0	0	–	0	0	0

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

1.5 Program Status Word (PSW)

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

Instruction	Flag 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

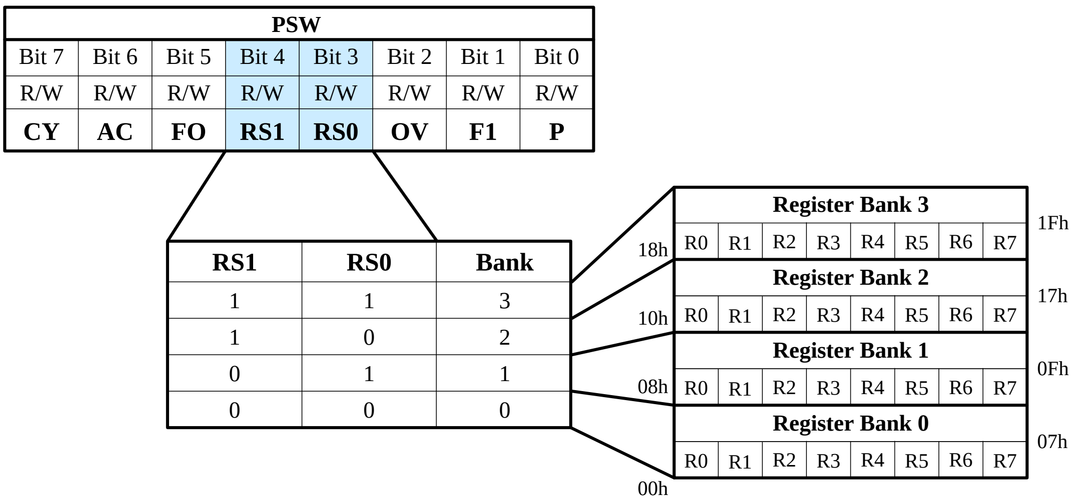
D0h.5 **F0:** General purpose user-definable flag

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

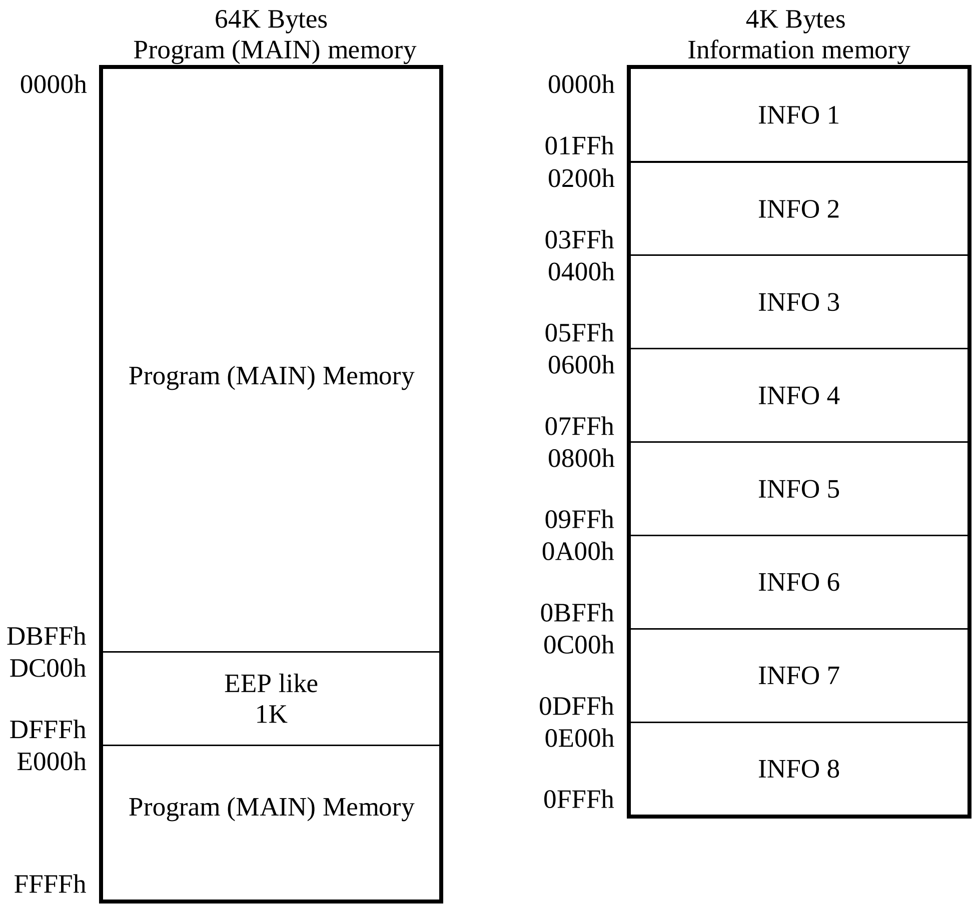
D0h.0 **P:** Parity flag. Set/cleared by hardware each instruction cycle to indicate odd/even number of “one” bits in the accumulator.



2. Memory

2.1 Program and Information Memory

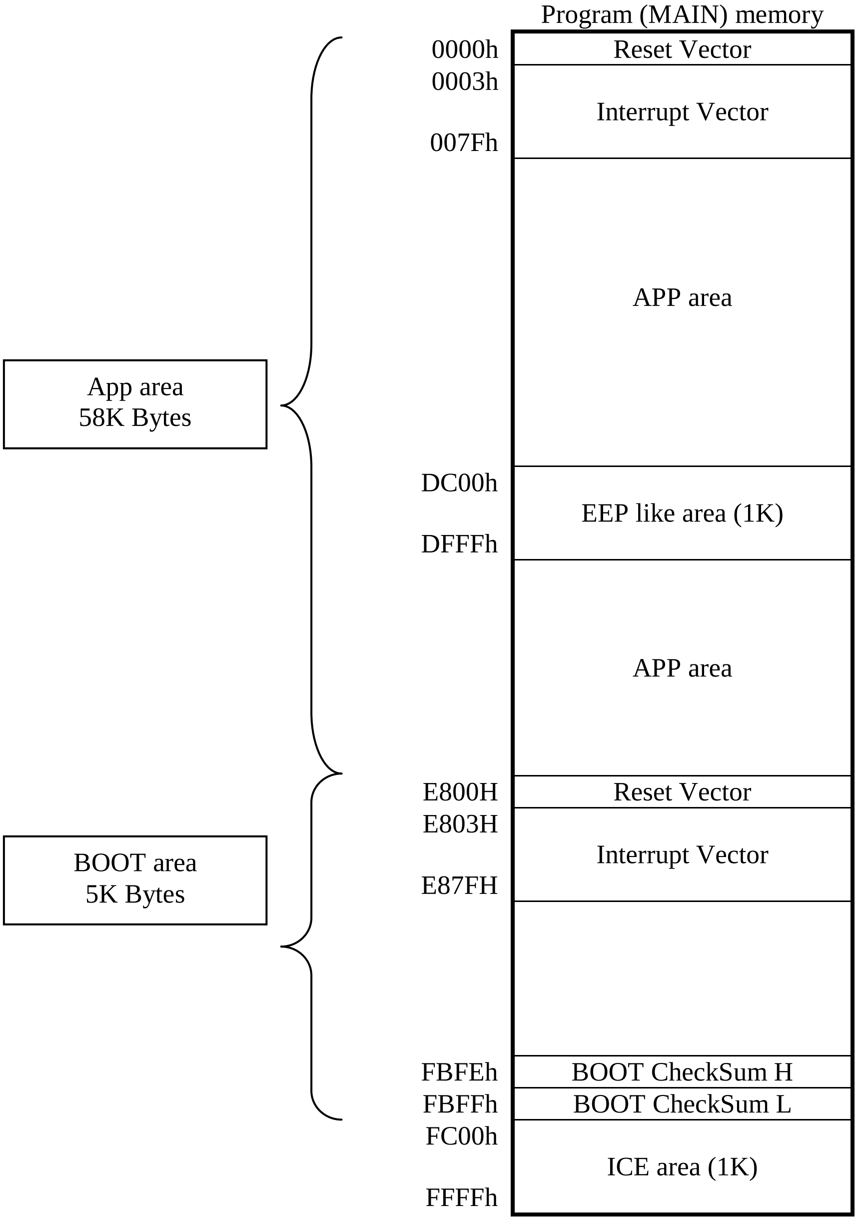
The Chip has a 64K Bytes program (MAIN) memory which can be divided into 128 pages and 4K Byte information memory which can be divided into 8 pages. EEPROM-like area (DC00h~DFFFh) are 1024 bytes. For program (MAIN) memory and information memory, one page is 512Bytes. Flash only provides page erase and byte write functions. If there are functions that use ICE, FC00h~FFFFh are reserved for ICE. Information memory INFO1~4 are reserved for the system, and INFO5~8 are for user to erase and write. Program (MAIN) memory area READ is simply achieved by a "MOVC" instruction and information memory (INFO) area READ is simply achieved by a "MOVX A, @DPTR" instruction.



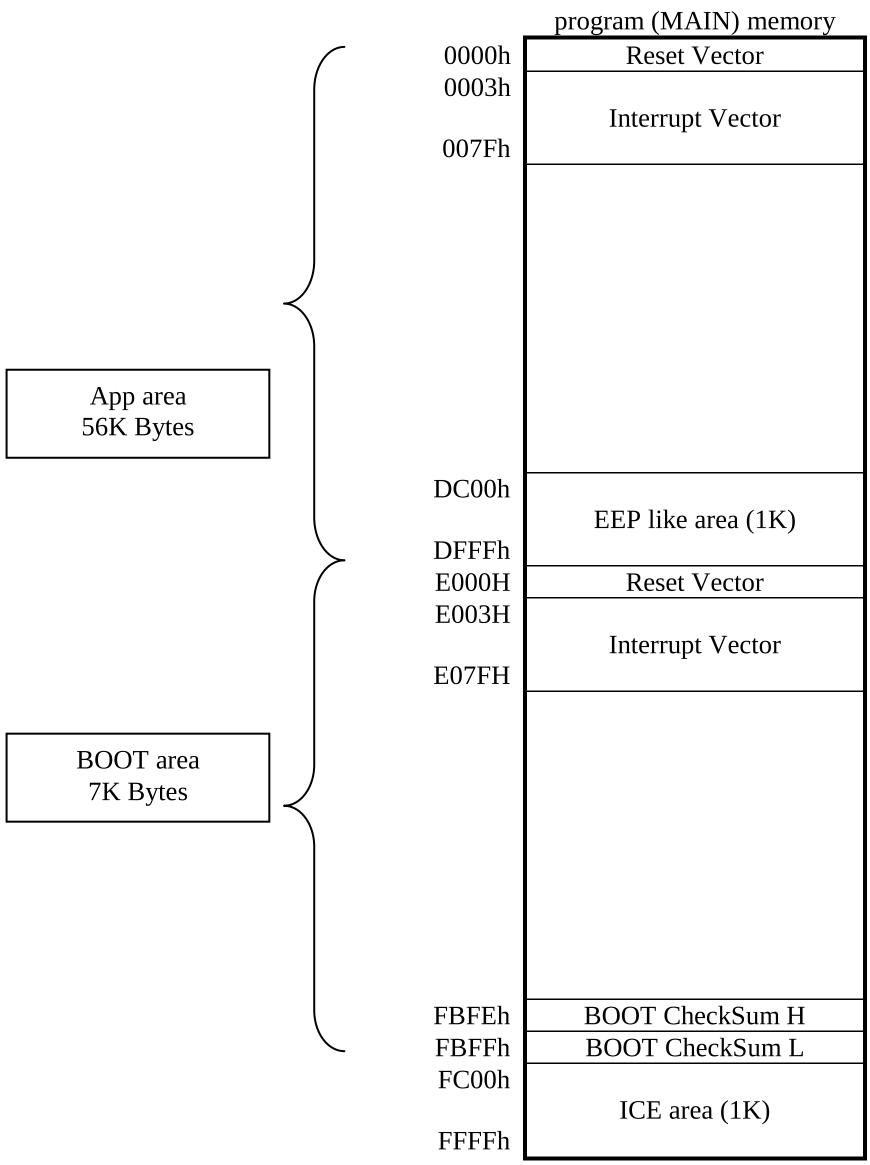
2.2 IAP BOOT UPGRADE

Flash supports the IAP BOOT upgrade function. At this time, the program memory can be divided into BOOT area and APP area. The BOOT area stores the protected update program code, and the APP area stores the program code that can be rewritten. Set SFR RSTV to decide Reset/Interrupt Vector after reset (etc: software reset, WDT reset or pin reset). BOOT area has its own storage write protection, and the size of the BOOT area is configured by set CFGWH[1:0] to select 5K, 7K or no BOOT area.

If CFGWH[1:0]=00b, sets no BOOT area, the power on reset/interrupt vector is 0000h.
If CFGWH[1:0]=10b, sets the 7K BOOT area, the power on reset/interrupt vector is E000h.
If CFGWH[1:0]=11b, sets the 5K BOOT area, the power on reset/interrupt vector is E800h.



64K Bytes Program (MAIN) memory with 5K boot area



64K Bytes Program (MAIN) memory with 7K boot area

2.3 IAP OPERATION

The **eF1386** has "In Application Program" (IAP) capability, which allows software to erase/write data to the Flash memory during CPU run time. When using IAP to write, user needs to erase first and then write bytes. After erasing, each address can only be written once. Flash only provides page erase and byte write functions. Before IAP Write, the user should disable the LVR first.

Flash IAP Write is simply achieved by a "MOVX @DPTR, A" instruction. Flash IAP writing needs higher V_{CC} voltage, V_{CC}>2.5V.

When IAP write/erase, SFR IAPWE/SWCMD needs to follow the following settings
SFR IAPWE
Write E2h and 4Ch to enable MAIN (APP/EEP-like) area byte write
Write E2h and BAh to enable MAIN (APP/EEP-like) area page erase
Write A1h and 4Ch to enable INFO5~8 area byte write
Write A1h and BAh to enable INFO5~8 area page erase

SFR SWCMD
Write 65h and A7h to enable IAP MAIN (APP) area write/erase

Write/Erase ADR range	EEP like area (MAIN)		APP area (when Boot area 7K)		APP area (when Boot area 5K)		INFO5~8	
	Write	Erase	Write	Erase	Write	Erase	Write	Erase
	DC00h-DFFFh		0000h-DBFFh		0000h-DBFFh E000h-E7FFh		0800h-0FFFh	
SFR SWCMD	-		65, A7 Only takes effect at boot area (E000~FFFFh)		65, A7 Only takes effect at boot area (E000~FFFFh)		-	
SFR IAPWE	E2, 4C	E2, BA	E2, 4C	E2, BA	E2, 4C	E2, BA	A1, 4C	A1, BA

IAP Write/Erase Enable Condition

Erase EEP-like area DC00h~DDFFh

```
 ; IAP example code
 ; need 2.5V < VCC < 5.5V
ORL     LVRCON, #10h      ; Disable LVR
MOV     DPTR, #DC00h      ; DPTR=DC00h=target IAP address
MOV     IAPWE, #E2h       ;
MOV     IAPWE, #BAh       ;
MOVX    @DPTR, A          ; write any data to DC00h~DDFFh
                          ; to page erase EEP-like area from DC00h to DDFFh
                          ; 2ms H/W writing time, CPU wait
MOV     IAPWE, #00h       ; IAP write disable, immediately after IAP write
ANL     LVRCON, #0EFh     ; Enable LVR
```

Erase APP area 0000h~01FFh

```
 ; IAP example code
 ; need 2.5V < VCC < 5.5V
ORL     LVRCON, #10h      ; Disable LVR
MOV     DPTR, #0000h      ; DPTR=0000h=target IAP address
MOV     IAPWE, #E2h       ;
MOV     IAPWE, #BAh       ;
MOV     SWCMD, #65h       ; Only takes effect at boot area
MOV     SWCMD, #A7h       ; Only takes effect at boot area
MOVX    @DPTR, A          ; write any data to 0000h~01FFh
                          ; to page erase APP area from 0000h~01FFh
                          ; 2ms H/W writing time, CPU wait
MOV     IAPWE, #00h       ; IAP write disable, immediately after IAP write
MOV     SWCMD, #00h       ; IAP write disable, immediately after IAP write
ANL     LVRCON, #0EFh     ; Enable LVR
```

Erase INFO5 area 0800h~09FFh

```
 ; IAP example code
 ; need 2.5V < VCC < 5.5V
ORL     LVRCON, #10h      ; Disable LVR
MOV     DPTR, #0800h      ; DPTR=0800h=target IAP address
MOV     IAPWE, #A1h       ;
MOV     IAPWE, #BAh       ;
MOVX    @DPTR, A          ; write any data to 0800h~09FF
                          ; to page erase INFO5 area from 0800h~09FFh
                          ; 2ms H/W writing time, CPU wait
MOV     IAPWE, #00h       ; IAP write disable, immediately after IAP write
ANL     LVRCON, #0EFh     ; Enable LVR
```

Write EEP-like area DC00h

```
 ; IAP example code
 ; need 2.5V < VCC < 5.5V
ORL     LVRCON, #10h       ; Disable LVR
MOV     DPTR, #DC00h       ; DPTR=DC00h=target IAP address
MOV     IAPWE, #E2h        ;
MOV     IAPWE, #4Ch        ;
MOV     A, #55h
MOVX    @DPTR, A           ; write 55h to EEP-like area DC00h
MOV     IAPWE, #00h        ; IAP write disable, immediately after IAP write
ANL     LVRCON, #0EFh      ; Enable LVR
```

Write APP area 0000h

```
 ; IAP example code
 ; need 2.5V < VCC < 5.5V
ORL     LVRCON, #10h       ; Disable LVR
MOV     DPTR, #0000h       ; DPTR=0000h=target IAP address
MOV     IAPWE, #E2h        ;
MOV     IAPWE, #4Ch        ;
MOV     SWCMD, #65h        ; Only takes effect at boot area
MOV     SWCMD, #A7h        ; Only takes effect at boot area
MOV     A, #55h
MOVX    @DPTR, A           ; write 55h to APP area 0000h
MOV     IAPWE, #00h        ; IAP write disable, immediately after IAP write
MOV     SWCMD, #00h        ; IAP write disable, immediately after IAP write
ANL     LVRCON, #0EFh      ; Enable LVR
```

Write INFO5 area 0800h

```
 ; IAP example code
 ; need 2.5V < VCC < 5.5V
ORL     LVRCON, #10h       ; Disable LVR
MOV     DPTR, #0800h       ; DPTR=0800h=target IAP address
MOV     IAPWE, #A1h        ;
MOV     IAPWE, #4Ch        ;
MOV     A, #55h
MOVX    @DPTR, A           ; write 55h to INFO5 area 0800h
MOV     IAPWE, #00h        ; IAP write disable, immediately after IAP write
ANL     LVRCON, #0EFh      ; Enable LVR
```

Read INFO5 area 0800h

```
 ; IAP example code
 ; need 2.5V < VCC < 5.5V
MOV     DPTR, #0800h       ; DPTR=0800h=target IAP address
MOVX    A, @DPTR           ; Read  data of INFO5 area 0800h
```

Read Main area 0800h

```
 ; IAP example code
 ; need 2.5V < VCC < 5.5V
MOV     DPTR, #0800h       ; DPTR=0800h=target IAP address
CLR     A
MOVC    A, @A+DPTR         ; Read  data of Main area 0800h
```

2.4 Flash ICP Mode

The Flash memory can be programmed by the tenx proprietary writer **(TWR98/TWR99)**, which needs at least four wires 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

INFO2 **0200h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CFGWH	PROTN	XRSTEN	–			–	BOOTV	

0200h.1~0 **BOOTV:** Power on reset vector select
00: 0x0000
01: 0x0000
10: 0xE000 (BOOT area 7K bytes)
11: 0xE800 (BOOT area 5K bytes, default)

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

97h.7~0 **IAPEN (W):**
Write 65h and A7h to enable APP area IAP write/erase (only activate at BOOTV area)
Write other value to disable IAP write/erase. It is recommended to clear it immediately after IAP access.

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

SFR **C9h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPWE	IAPWE							
	IAPWE	IAPTO	–					
R/W	R	R	W					
Reset	0	0	–					

C9h.7~0 **IAPWE (W):**
Write E2h and 4Ch to enable IAP APP area write
Write E2h and BAh to enable IAP APP area erase
Write A1h and 4Ch to enable IAP INFO5~8 area write
Write A1h and BAh to enable IAP INFO5~8 area erase
Write other value to disable IAP write/erase. It is recommended to clear it immediately after IAP access

C9h.7 **IAPWE (R):**
0: IAP write/page erase disable
1: IAP write/page erase enable

C9h.6 **IAPTO (R):**
IAP Time-Out flag, Set by H/W when IAP Time-out occurs. Cleared by H/W when IAPWE=0.

SFR **EEh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BOOTV	–	–	–	–	–	RSTV	BOOTVR	
R/W	–	–	–	–	–	R/W	R	R
Reset	–	–	–	–	–	1	0	0

EEh.2 **RSTV:** Change the reset vector. Default 1 at power-on reset, other resets will not change user settings
0: Reset vector = 0x0000
1: Reset vector = 0xE800 or 0xE000 (determined by BOOTV)

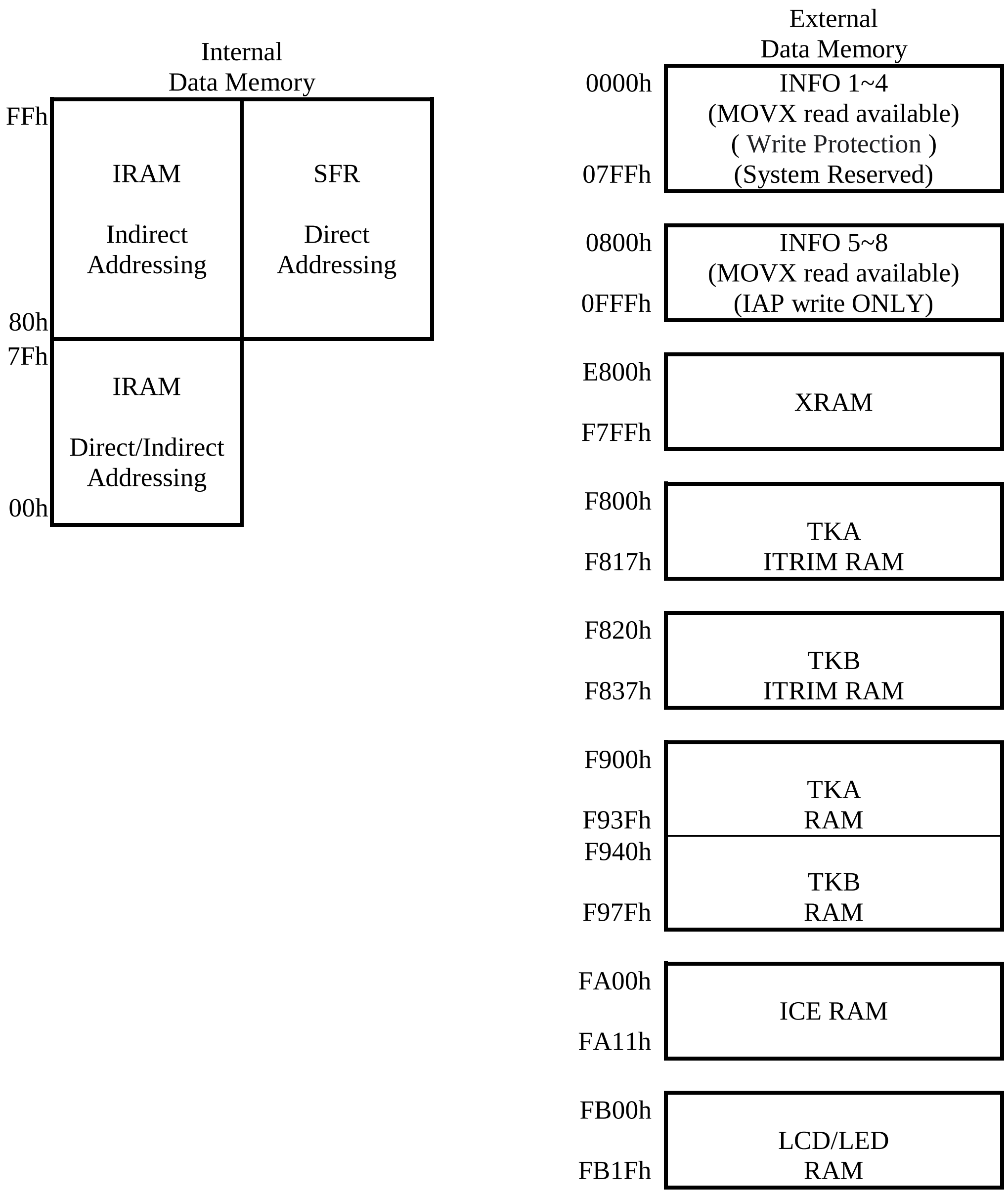
EEh.1~0 **BOOTVR:** Power on reset vector select. Read only. Load from CFGWH.BOOTV
00: 0x0000
01: 0x0000
10: 0xE000 (BOOT area 7K bytes)
11: 0xE800 (BOOT area 5K bytes, default)

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

F7h.2~1 **IAPTE:** IAP write watchdog timer enable
00: Disable
01: wait 0.8mS trigger watchdog time-out flag, and escape the write fail state
10: wait 3.2mS trigger watchdog time-out flag, and escape the write fail state
11: wait 6.4mS 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 4096 Bytes XRAM, 32 Bytes LCD/LED RAM, 10 Bytes ICE RAM and TK RAM, which can be only accessed by MOVX instruction. INFO1~8 is flash information memory but is treated as external memory in use.



2.5.1 IRAM

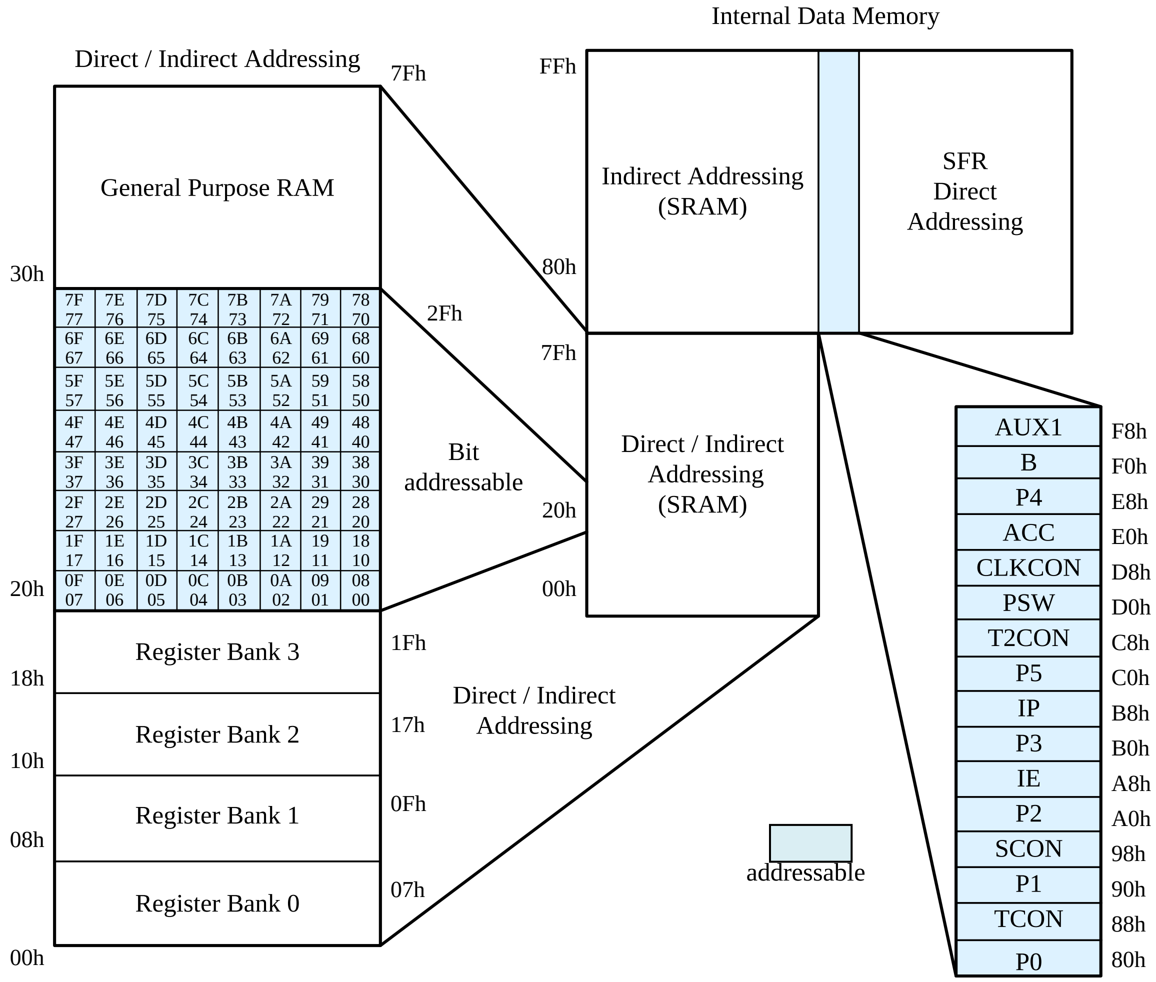
IRAM is located in the 8051 internal data memory space. The whole 256 Bytes IRAM are accessible using indirect addressing but only the lower 128 Bytes are accessible using direct addressing. There are four directly addressable register banks (switching by PSW), which occupy IRAM space from 00h to 1Fh. 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 E800h to F7FFh). The 4096 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/LCD, which are unique to the Chip.



	8/0	9/1	A/2	B/3	C/4	D/5	E/6	F/7
F8h	AUX1							
F0h	B	CRCDL	CRCDH	CRCIN		CFGBG	CFGWL	AUX2
E8h	P4	SIADR	SICON	SIRCD1	SITXRCD2		BOOTV	PWRCON
E0h	ACC	MICON	MIDAT			EFTCON	EXA	EXB
D8h	CLKCON	PWMPRDH	PWMPRDL			UART1CON	UART0CON	TKPINSB2
D0h	PSW	PWMDH	PWMDL			UART2CON	LVRCON	TKPINSB1
C8h	T2CON	IAPWE	RCP2L	RCP2H	TL2	TH2	EXA2	EXA3
C0h	P5	TKPINSA0	TKPINSA1	TKPINSA2	TKPINSB0	ATKCHA0	ATKCHA1	ATKCHA2
B8h	IP	IPH	IP1	IP1H	SPCON	SPSTA	SPDAT	LVDCON
B0h	P3	LXDCON	LXDCON2		TKTMRL	TKCON2	ATKCHB1	ATKCHB0
A8h	IE	INTE1	ADCDL	ADCDH	TKCHSB	TKCON	CHSEL	ATKCHB2
A0h	P2	PWMCON	PINMOD10	PINMOD32	PINMOD54	PINMOD76	PINMODE	TKCHSA
98h	SCON	SBUF	SCON1	SBUF1	TKCON3	PWM2CON	PWMIDX	PWMEN
90h	P1	PORTIDX			OPTION	INTFLG	INTPIN	SWCMD
88h	TCON	TMOD	TL0	TL1	TH0	TH1	SCON2	SBUF2
80h	P0	SP	DPL	DPH		INTPORT	INTPWM	PCON

3. LVR and LVD setting

The Chip provides LVR and Low Voltage Detection (LVD) functions. There are 8-level LVR can be selected by LVRCON and 16-level LVD can be selected by SFR LVDCON. The SFR PWRSAV bits also affect LVR function as tables below.

Operation Mode	SFR			LVR	Function	Current consumption
	LVRPD	PWRSAV	LVRS			
Fast Slow	0	X	000	ON	LV Reset 2.24V	
	0	X	001	ON	LV Reset 2.48V	
	0	X	010	ON	LV Reset 2.72V	
	0	X	011	ON	LV Reset 2.96V	
	0	X	100	ON	LV Reset 3.2V	
	0	X	101	ON	LV Reset 3.44V	
	0	X	110	ON	LV Reset 3.68V	
	0	X	111	ON	LV Reset 3.92V	
Idle Halt Stop	0	0	000	ON	LV Reset 2.24V	Idle:180uA@5V Halt: 58uA@5V Stop: 55uA@5V
	0	0	001	ON	LV Reset 2.48V	
	0	0	010	ON	LV Reset 2.72V	
	0	0	011	ON	LV Reset 2.96V	
	0	0	100	ON	LV Reset 3.2V	
	0	0	101	ON	LV Reset 3.44V	
	0	0	110	ON	LV Reset 3.68V	
	0	0	111	ON	LV Reset 3.92V	
Idle	0	1	XXX	ON	POR 2.3V	165uA@5V
	1	0	XXX			
Halt Stop	0	1	XXX	OFF	-	Halt: 14uA@5V Stop: 11uA@5V
	1	0	XXX			

LVR and LVD function

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

F7h.5 **PWRSAV:** Power saving mode control
0: No power saving
1: Power saving, disable LVR in IDLE/HALT/STOP mode

SFR **BFh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVDCON	LVDM	LVDO	LVDDBS	LVDPD	LVDS			
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

BFh.7 **LVDM:** Low Voltage Detect MODE
0: LVDIF =1 and LVDO =1 while VCC < LVDS
1: LVDIF =1 and LVDO =1 while VCC > LVDS

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

BFh.5 **LVDDBS:** Low Voltage Detect debounce select
0: Disable
1: Enable

BFh.4 **LVDPD:** Low Voltage Detect select
0: Enable LVD
1: Disable LVD

BFh.3~0 **LVDS:** Low Voltage Detect select
0000: Set LVD at 2.52V
0001: Set LVD at 2.62V
0010: Set LVD at 2.74V
0011: Set LVD at 2.86V
0100: Set LVD at 2.99V
0101: Set LVD at 3.1V
0110: Set LVD at 3.23V
0111: Set LVD at 3.35V
1000: Set LVD at 3.48V
1001: Set LVD at 3.6V
1010: Set LVD at 3.72V
1011: Set LVD at 3.84V
1100: Set LVD at 3.96V
1101: Set LVD at 4.08V
1110: Set LVD at 4.2V
1111: Set LVD at 4.32V

SFR D6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVRCON	SXTGAIN		–	LVRPD	–	LVRS		
R/W	R/W	R/W	–	R/W	–	R/W	R/W	R/W
Reset	0	0	–	0	–	0	0	0

D6h.4 **LVRPD:** Low Voltage Reset function select
0: Disable
1: Enable

D6h.2~0 **LVRS:** Low Voltage Reset function select
000: Set LVR at 2.24V
001: Set LVR at 2.48V
010: Set LVR at 2.72V
011: Set LVR at 2.96V
100: Set LVR at 3.20V
101: Set LVR at 3.44V
110: Set LVR at 3.68V
111: Set LVR at 3.92V

4. Reset

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

4.1 Power on Reset

After Power on Reset, the device stays on Reset state for 40 ms as chip warm up time, then downloads the CFGW register from ROM's last six bytes. The Power on Reset needs VCC pin's voltage first discharge to near VSS level, then rise beyond 2.3V.

4.2 External Pin Reset

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

4.3 Software Command Reset

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

4.4 Watchdog Timer Reset

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

4.5 Low Voltage Reset

The Chip provides LVR functions. There are 8-level LVR can be selected by CFGWH.

SFR D6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVRCON	SXTGAIN		–	LVRPD	–	LVRS		
R/W	R/W	R/W	–	R/W	–	R/W	R/W	R/W
Reset	0	0	–	0	–	0	0	0

D6h.4 **LVRPD:** Low Voltage Reset function select
0: Disable
1: Enable

D6h.2~0 **LVRS:** Low Voltage Reset function select
000: Set LVR at 2.24V
001: Set LVR at 2.48V
010: Set LVR at 2.72V
011: Set LVR at 2.96V
100: Set LVR at 3.20V
101: Set LVR at 3.44V
110: Set LVR at 3.68V
111: Set LVR at 3.92V

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	TM3CKS		WDTPSC		ADCKS		TM3PSC	
R/W	R/W	R/W	R/W		R/W		R/W	
Reset	0	0	0	0	0	0	0	0

94h.5~4 **WDTPSC:** Watchdog Timer pre-scalar time select
00: 400ms WDT overflow rate
01: 200ms WDT overflow rate
10: 100ms WDT overflow rate
11: 50ms WDT overflow rate

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

95h.7 **LVDIF:** Low Voltage Detect interrupt flag
Set by H/W. S/W writes 7Fh to INTFLG to clear this flag.

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

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

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

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

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Reset	0	0	0	0	–	0	0	0

F8h.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 FXT (Fast Crystal, 1~16 MHz) or FRC (Fast Internal RC, 18.432 MHz). The Slow clock can be selected as SXT (Slow Crystal, 32 KHz) or SRC (Slow Internal RC, 80 KHz). Fast mode and Slow mode are defined as the CPU running at Fast and Slow clock speeds.

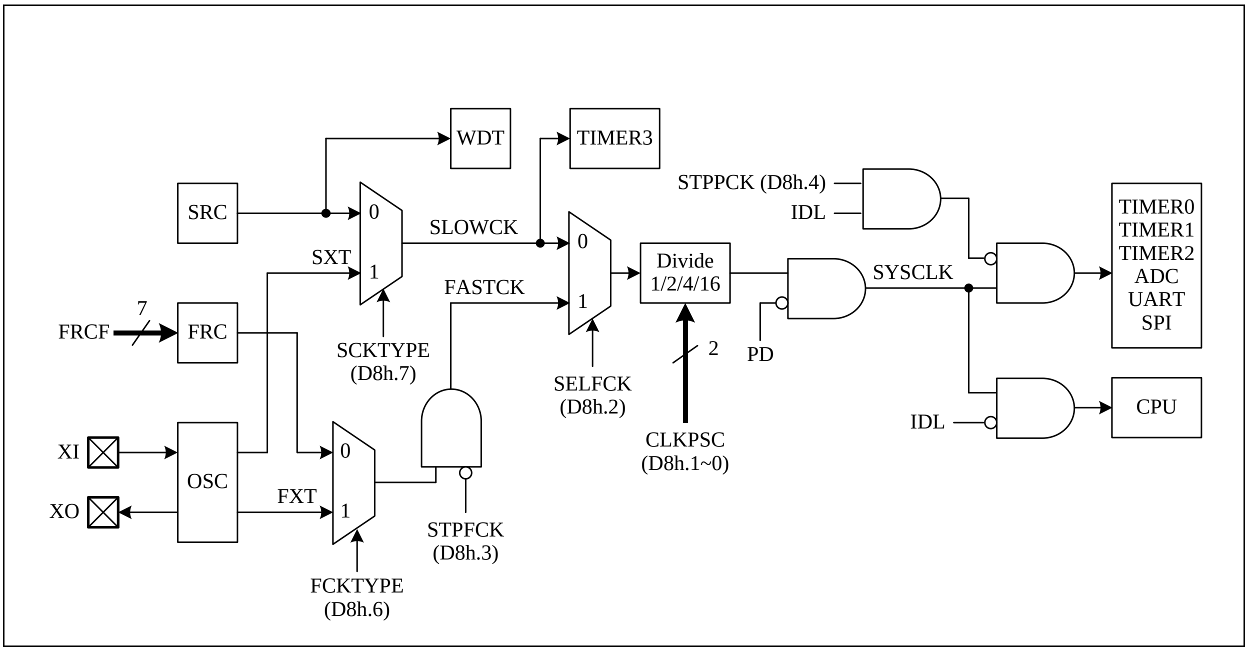
After Reset, the device is running at Slow mode with 80 KHz SRC. S/W should select the proper clock rate for chip operation safety. The higher V_{CC} allows the chip to run at a higher System clock frequency. In a typical condition, an 18 MHz System clock rate requires V_{CC}> 2.3V.

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 Fast mode, the fast oscillator can be used in the range from 1~16 MHz. In Slow mode, the slow oscillator can only use a clock frequency of 32.768 KHz.

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.

If user wants to switch Fsys from Slow clock to FXT, user should be following the step below

1. Set FCKTYPE (D8h.6)
2. Wait 2ms until FXT oscillation stable
3. Set SELFCK (D8h.2)



Clock Structure

Note: *Because of the CLKPSC delay, it needs to wait for 16 clock cycles (max.) before switching Slow clock to Fast clock. Also refer to AP-TM52XXXXX_01S and AP-TM52XXXXX_02S about System Clock Application Note.*

SYSCLK	CLKCON (D8h)			
	bit7 SCKTYPE	bit6 FCKTYPE	bit3 STPFCK	bit2 SELFCK
Fast FXT	0/1	1	0	1
Fast FRC	0/1	0	0	1
Slow SXT	1	0/1	0/1	0
Slow SRC	0	0/1	0/1	0
Fast type change	0/1	0 ←→ 1	0/1	0
Slow type change	0 ←→ 1	0/1	0	1
Stop FRC/FXT	0/1	0/1	0 → 1	0
Switch to FRC/FXT	0/1	0/1	0	0 → 1
Switch to SRC/SXT	0/1	0/1	0	1 → 0

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	FCKTYPE	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	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

D8h.6 **FCKTYPE:** Fast clock type. This bit can be changed only in Slow mode (SELFCK=0).
0: FRC
1: FXT

D8h.5 **STPSCK:** Set 1 to stop Slow clock after PD=1 (Halt / Stop mode control)

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 four 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. 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. Must be set to slow clock mode (SELFCK=0) before entering Stop mode (PDOWN).

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.

Note: *Chip cannot enter Stop/Halt Mode if INTn pin is low and wakeup is enable. (INTn=0 and EXn=1, n=0~9)*

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 STOP/Halt mode

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

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

F7h.4 **VBGOUT:** VBG voltage output to P3.2
0: Disable
1: Enable, The additional condition VBGEN=1 (AEh.1) should be set.

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

D8h.6 **FCKTYPE:** Fast clock type. This bit can be changed only in Slow mode (SELFCK=0).
0: FRC 1: FXT

D8h.5 **STPSCK:** Set 1 to stop Slow clock after PD=1 (Halt / Stop mode control)

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

6. Interrupt & Wake-up

This Chip has a 14-source four-level priority interrupt structure. All enabled Interrupts can wake up CPU from Idle mode, but only the Pin Interrupts can wake up CPU from Stop mode. The Halt mode can be waked up by Time3 and Pin Interrupts. 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 Stop/Halt mode)
000B	TF0	Timer0 Interrupt
0013	IE1	INT1 external pin Interrupt (can wake up Stop/Halt mode)
001B	TF1	Timer1 Interrupt
0023	RI+TI	Serial Port (UART0) Interrupt
002B	TF2+EXF2	Timer2 Interrupt
0033	–	Reserved for ICE mode use
003B	TF3	Timer3 Interrupt
0043	PCIF P0IF~P5IF PIN0IF~PIN7IF	Pin change Interrupt (can wake up Stop/Halt mode)
004B	LVDIF	LVD interrupt
0053	ADIF TKIFA TKIFB	ADC/Touch Key A/B Interrupt
005B	SPIF+WCOL+MODF	SPI Interrupt
0063	RI1+TI1 RI2+TI2	Serial Port (UART1/UART2) Interrupt
006B	MIIF TXDF, RCD2F, RCD1F	I^2C interrupt Vector
0073	PWM0IF, PWM1IF PWM2IF, PWM3IF	PWM0~3 Interrupt Vector

Interrupt Vector & Flag

6.1 Interrupt Enable and Priority Control

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

6.2 Suggestions on interrupting subroutines

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

SFR **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 (UART0) interrupt enable
0: Disable Serial Port (UART0) interrupt
1: Enable Serial Port (UART0) interrupt

A8h.3 **ET1:** Timer1 interrupt enable
0: Disable Timer1 interrupt
1: Enable Timer1 interrupt

A8h.2 **EX1:** External INT1 pin Interrupt enable and Stop/Halt mode wake up enable
0: Disable INT1 pin Interrupt and Stop/Halt mode wake up
1: Enable INT1 pin Interrupt and Stop/Halt mode wake up, it can wake up CPU from Stop/Halt 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 Stop/Halt mode wake up enable
0: Disable INT0 pin Interrupt and Stop/Halt mode wake up
1: Enable INT0 pin Interrupt and Stop/Halt mode wake up, it can wake up CPU from Stop/Halt 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	I2CE	ES2	SPIE	ADTKIE	LVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

A9h.7 **PWMIE:** PWM0~PWM2 interrupt enable
0: Disable PWM0~PWM2 interrupt
1: Enable PWM0~PWM2 interrupt

A9h.6 **I2CE:** I^2C (master/slave) interrupt enable
0: Disable I^2C interrupt
1: Enable I^2C interrupt

A9h.5 **ES2:** Serial Port (UART1/UART2) interrupt enable
0: Disable Serial Port (UART1/UART2) interrupt
1: Enable Serial Port (UART1/UART2) interrupt

A9h.4 **SPIE:** SPI interrupt enable
0: Disable SPI interrupt
1: Enable SPI interrupt

A9h.3 **ADTKIE:** ADC/Touch Key interrupt enable
0: Disable ADC/Touch Key interrupt
1: Enable ADC/Touch Key interrupt

A9h.2 **LVDIE:** LVD interrupt enable
0: Disable LVD interrupt
1: Enable LVD interrupt.

A9h.1 **PCIE:** Port0~Port5 pin change interrupt enable. This bit does not affect Stop/Halt mode wake up capability.
0: Disable Port0~Port5 pin change interrupt
1: Enable Port0~Port5 pin change interrupt

A9h.0 **TM3IE:** Timer3 interrupt enable and Halt mode wake up enable
0: Disable Timer3 interrupt t and Halt mode wake up
1: Enable Timer3 interrupt t and Halt mode wake up, it can wake up CPU from Halt mode no matter EA is 0 or 1.

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 (UART1) Interrupt Priority control. Definition as above.

B9h.3, B8h.3 **PT1H, PT1 :** Timer1 Interrupt Priority control. Definition as above.

B9h.2, B8h.2 **PX1H, PX1 :** External INT1 pin Interrupt Priority control. Definition as above.

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

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

SFR **BBh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IP1H	PPWMH	PI2CH	PS2H	PSPIH	PADTKIH	PLVDH	PPCH	PT3H
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR **BAh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IP1	PPWM	PI2C	PS2	PSPI	PADTKI	PLVD	PPC	PT3
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

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

BBh.6, BAh.6 **PI2CH, PI2C:** I2C (Master/Slave) Interrupt Priority control. Definition as above.

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

BBh.4, BAh.4 **PSPIH, PSPI:** SPI Interrupt Priority control. Definition as above.

BBh.3, BAh.3 **PADTKIH, PADTKI:** ADC/Touch Key Interrupt Priority control. Definition as above.

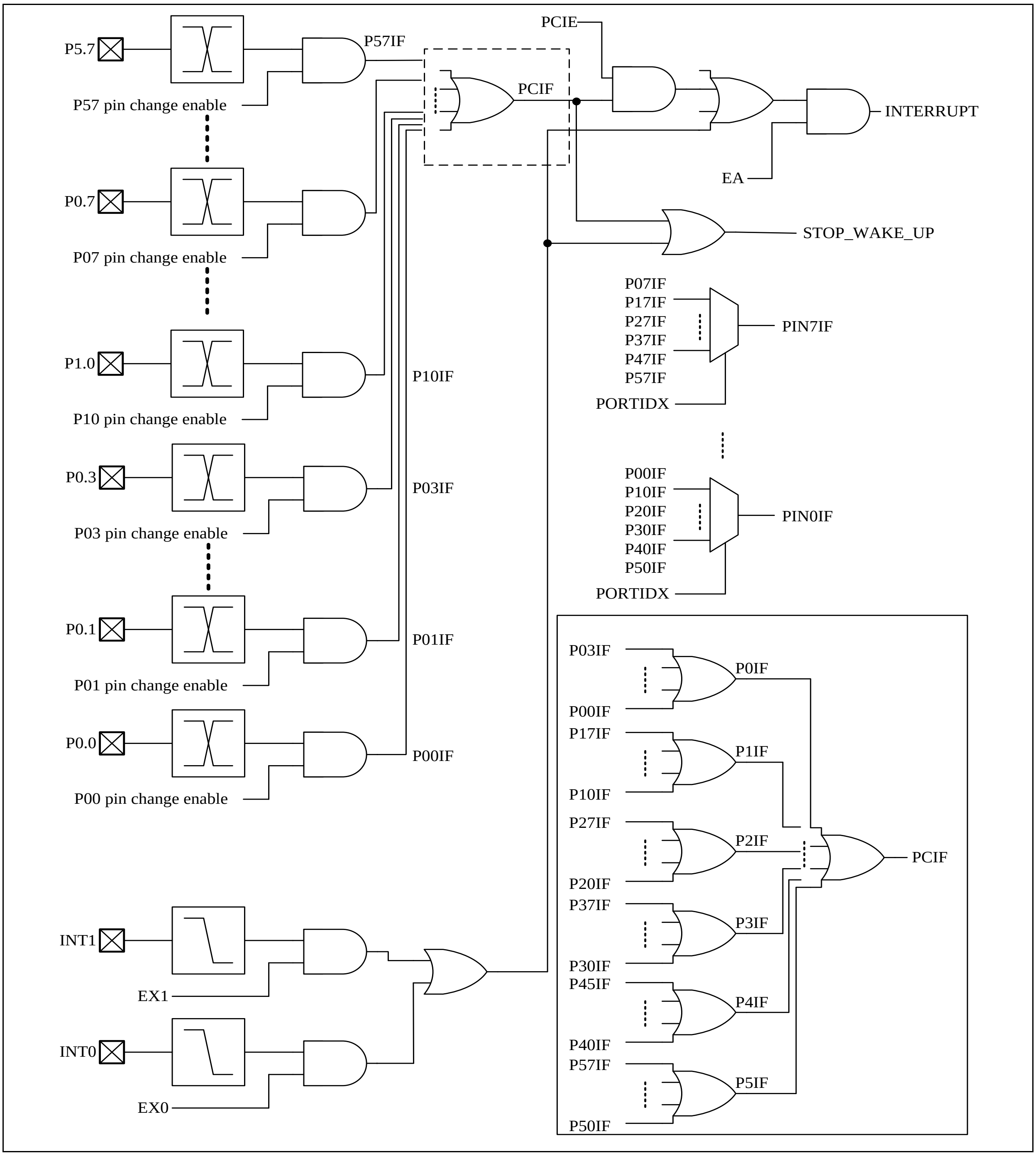
BBh.2, BAh.2 **PLVDH, PLVD:** LVD Interrupt Priority control. Definition as above.

BBh.1, BAh.1 **PPCH, PPC:** Port0~Port5 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

Pin Interrupts include INT0~INT1 and Port0~Port5 pin change interrupt. INT0~INT1 and Port0~Port5 pin change also have the Stop/Halt mode wake up capability. INT0 and INT1 are falling edge or low level triggered as the 8051 standard. Port0~Port5 Pin Change Interrupt is triggered by IO state change. Pin change enable are setting by PINMOD10/PINMOD32/PINMOD54/PINMOD76. For details, see Chapter 7. PINMODE and pin change enable settings.



Pin interrup/Wake up

Note: Chip cannot enter Stop/Halt Mode if INTn is low and wakeup is enabled. (INTn=0 and EXn=1, n=0~1)
Chip cannot enter Stop/Halt Mode if PCIF=1. User should clear PCIF before enter Stop/Halt mode.

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 85h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTPORT	–	–	P5IF	P4IF	P3IF	P2IF	P1IF	P0IF
R/W	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	0	0	0	0	0	0

96h.5 **P5IF:** P5.7~P5.0 pin change interrupt flag, Write 0 to clear P5.7~P5.0 pin change interrupt flag

96h.4 **P4IF:** P4.7~P4.0 pin change interrupt flag, Write 0 to clear P4.7~P4.0 pin change interrupt flag

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

96h.2 **P2IF:** P2.7~P2.0 pin change interrupt flag, Write 0 to clear P2.7~P2.0 pin change interrupt flag

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

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

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

95h.1 **PCIF:** Port0~Port5 Pin change interrupt flag
Set by H/W when Port0~Port5 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~Port5), it will also clear PIN0IF~PIN7F and P0IF~P5IF.

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

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

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

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

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

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

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

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

96h.0 **PIN0IF:** Px.0 pin change interrupt flag, Write 0 to clear Px.1 pin change interrupt flag
port number (x) define by PORTIDX

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

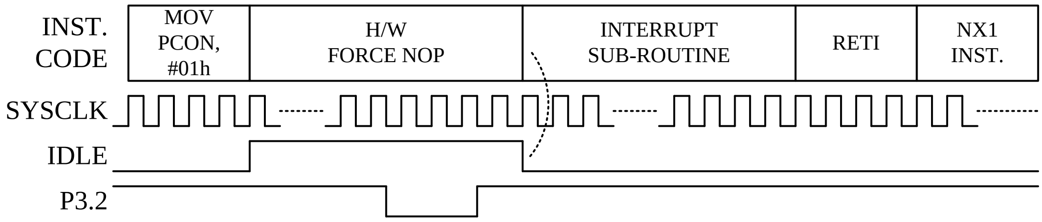
A8h.7 **EA:** Global interrupt enable control.
0: Disable all Interrupts.
1: Each interrupt is enabled or disabled by its individual interrupt control bit

A8h.2 **EX1:** External INT1 pin Interrupt enable and Stop/Halt mode wake up enable
0: Disable INT1 pin Interrupt and Stop/Halt mode wake up
1: Enable INT1 pin Interrupt and Stop/Halt mode wake up, it can wake up CPU from Stop/Halt mode no matter EA is 0 or 1.

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

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 (Pins, Timers, TK, SPI and UARTs) can wake up CPU from Idle mode. Upon Idle wake-up, Interrupt service routine is entered immediately. "The first instruction behind IDL (PCON.0) setting" is executed after interrupt service routine return.



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

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

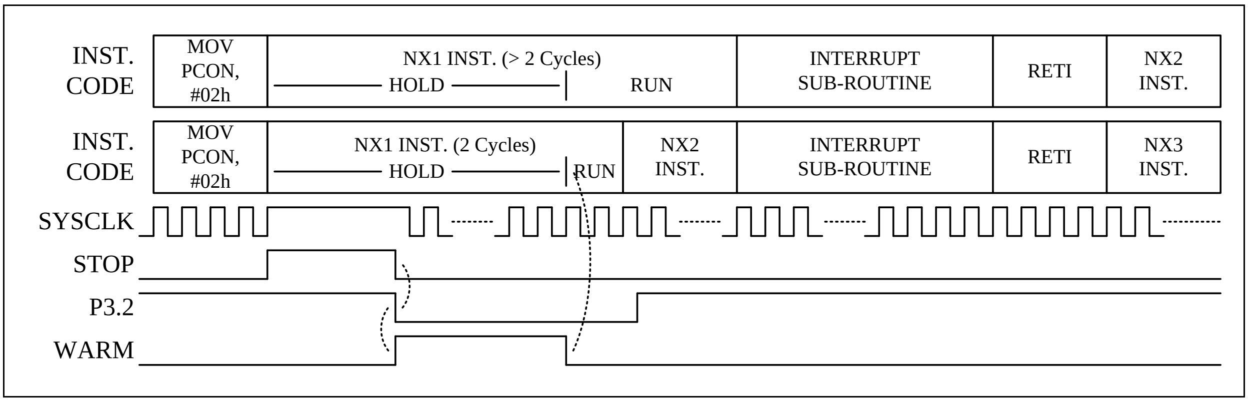
87h.1 **PD:** Power down control bit, set 1 to enter STOP/HALT mode.

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

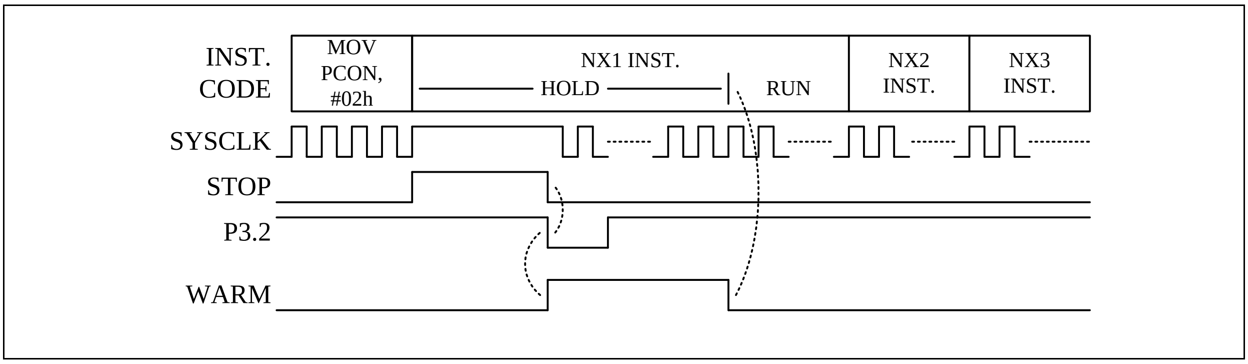
6.5 Stop/Halt mode Wake up and Interrupt

Stop/Halt mode wake up is simple, as long as the individual pin interrupt enable bit (ex: EX0) is set, the pin wake up capability is asserted. Set EX0/EX1 can enable INT0/INT1 pins' Stop/Halt mode wake up capability. Set PINMOD10/PINMOD32/PINMOD54/PINMOD76 can enable Port0~Port5 Stop/Halt mode wake up capability. Upon Stop/Halt wake up, "the first instruction behind PD setting (PCON.1)" is executed immediately before Interrupt service. Interrupt entry requires EA=1 and trigger state of the pin staying sufficiently long to be observed by the System clock. This feature allows CPU to enter or not enter Interrupt sub-routine after Stop/Halt mode wake up.

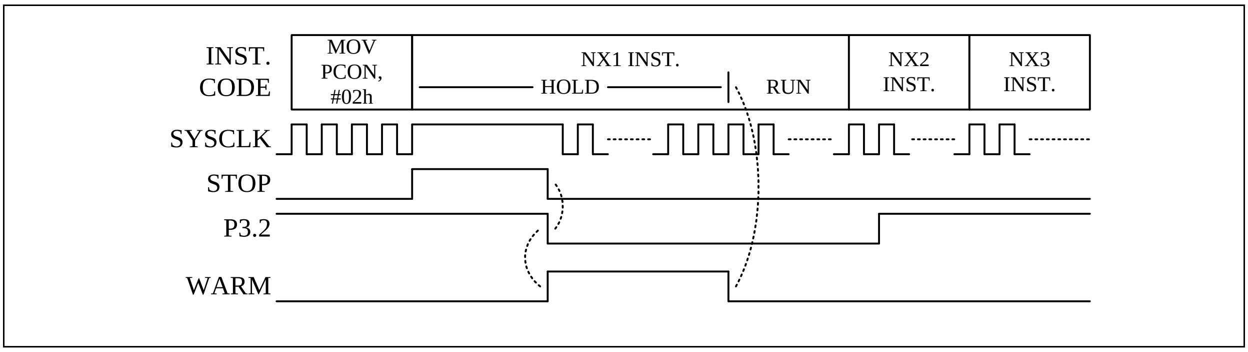
Note: Chip cannot enter Stop/Halt Mode if INTn is low and wakeup is enabled. (INTn=0 and EXn=1, n=0~1)
Chip cannot enter Stop/Halt Mode if PCIF=1. User should clear PCIF before enter Stop/Halt mode.



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



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



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

7. I/O Ports

The Chip has total 42 multi-function I/O pins. All I/O pins follow the standard 8051 "Read-Modify-Write" feature. The instructions that read the SFR rather than the Pin State are the ones that read a port or port bit value, possibly change it, and then rewrite it to the SFR (ex: ANL P1, A; INC P2; CPL P3.0).

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

7.1 Port0~Port5

IO pins can be defined in different modes as below.

P INMOD76 PINMOD54 PINMOD32 PINMOD10					Function	Interrupt	Wake-up
MODE00	0	0	0	0	Open Drain with pull-up	-	-
MODE01	0	0	0	1	Open Drain (Default)	-	-
MODE02	0	0	1	0	CMOS Output	-	-
MODE03	0	0	1	1	ADC/TK channel, XI/XO	-	-
MODE10	0	1	0	0	Open Drain with pull-up	-	-
MODE11	0	1	0	1	Open Drain	-	-
MODE12	0	1	1	0	CMOS Output	-	-
MODE13	0	1	1	1	LCD / LED output	-	-
MODE20	1	0	0	0	Open Drain with pull-up	Y	Y
MODE21	1	0	0	1	Open Drain	Y	Y
MODE22	1	0	1	0	CMOS Output	-	-
MODE23	1	0	1	1	PWMO, TxO output	-	-
MODE30	1	1	0	0	Open Drain with pull-up	Y	Y
MODE31	1	1	0	1	Open Drain	Y	Y
MODE32	1	1	1	0	CMOS Output	-	-
MODE33	1	1	1	1	Reserved	-	-

Table 7.1 Port0~Port5 I/O Pin Function Table

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

For example:
If PORTIDX=0, PINMOD10 is set to P0.1 and P0.0, high 4 bits are set to P0.1, low 4bits are set to P0.0
If PORTIDX=1, PINMOD10 is set to P1.1 and P1.0, high 4 bits are set to P1.1, low 4bits are set to P1.0
…
If PORTIDX=5, PINMOD10 is set to P5.1 and P5.0, high 4 bits are set to P5.1, low 4bits are set to P5.0
If PORTIDX=0, PINMOD32 is set to P0.3 and P0.2, high 4 bits are set to P0.3, low 4bits are set to P0.2
…
If PORTIDX=5, PINMOD76 is set to P5.7 and P5.6, high 4 bits are set to P5.7, low 4bits are set to P5.6

Mode	Port0~Port5 pin function	Px.n SFR data	Pin State	Resistor Pull-up	Digital Input
MODEx0	Open Drain with pull-up	0	Drive Low	N	N
		1	Pull-up	Y	Y
MODEx1	Open Drain	0	Drive Low	N	N
		1	Hi-Z	N	Y
MODEx2	CMOS Output	0	Drive Low	N	N
		1	Drive High	N	N
MODEx3	Alternative function ADC/TK/LCD/LED/PWM/TxO	X (don't care)	–	N	N

I/O Pin Function Table

If a Port0~Port5 pin is used for Schmitt-trigger input, S/W must set the I/O pin to MODEx0 or MODEx1 and set the corresponding Port Data SFR to 1 to disable the pin's output driving circuitry.

Beside I/O port function, each Port0~Port5 has one or more alternative functions, such as LED, ADC and Touch Key. Most of the functions are activated by setting the individual pin mode control SFR to MODEx3. Port1/Port3 pins have standard 8051 auxiliary definition such as INT0/1, T0/1/2, or RXD/TXD. These pin functions need to set the pin mode SFR to MODEx0 or MODE1 and keep the P1.n/P3.n SFR at 1.

Pin Name	Wakeup Interrup	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P0.3	Y	AD03	TKB19			PWM33		SDA
P0.2	Y	AD02	TKB18			PWM32		SCL
P0.1	Y	AD01	TKB17			PWM31		
P0.0	Y	AD00	TKB16			PWM30		

Port0 multi-function Table

Pin Name	Wakeup Interrup	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P1.7	Y	AD17	TKA14	LCOM7/LED7	COM7/SEG19	PWM1		
P1.6	Y	AD16	TKA15	LCOM6/LED6	COM6/SEG18	PWM0		
P1.5	Y	AD15	TKA16	LCOM5/LED5	COM5/SEG17	PWM1		
P1.4	Y	AD14	TKA17	LCOM4/LED4	COM4/SEG16	PWM0		
P1.3	Y	AD13	TKA18	LCOM3/LED3	COM3			
P1.2	Y	AD12	TKA19	LCOM2/LED2	COM2			
P1.1	Y	AD11	TKA20	LCOM1/LED1	COM1		TXD2	
P1.0	Y	AD10	TKB20	LCOM0/LED0	COM0		RXD2	

Port1 multi-function Table

Pin Name	Wakeup Interrup	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P2.7	Y	AD27	TKA10		SEG3	PWM1		
P2.6	Y	AD26	TKA11		SEG2	PWM0		
P2.5	Y	AD25	TAK12		SEG1			
P2.4	Y	AD24	TKA13		SEG0			
P2.3	Y	AD23	TKB02		SEG11			
P2.2	Y	AD22	TKB03		SEG10			
P2.1	Y	AD21	TKB04		SEG9			
P2.0	Y	AD20	TKB00		SEG8			

Port2 multi-function Table

Pin Name	Wakeup Interrup	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P3.7	Y	AD37	TKA06		SEG7		TXD1A	
P3.6	Y	AD36	TKA07		SEG6		RXD1A	
P3.5	Y	AD35	TKA08		SEG5		TXD0B	SDA
P3.4	Y	AD34	TKA09		SEG4		RXD0B	SCL
P3.3	Y	AD33	TKA01		SEG15	PWM35		
P3.2	Y	AD32	TKA00		SEG14	PWM34		
P3.1	Y	AD31	TKB05		SEG13		TXD0A	SDA
P3.0	Y	AD30	TKB01		SEG12		RXD0A	SCL

Port3 multi-function Table

Pin Name	Wakeup Interrup	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P4.5	Y	AD39	TKA03				TXD0C	
P4.4	Y	AD38	TKA02				RXD0C	
P4.3	Y	AD29	TKB07					NSS
P4.2	Y	AD28	TKB06					MISO
P4.1	Y	AD41	TKA05					MOSI
P4.0	Y	AD40	TKA04					SCK

Port4 multi-function Table

Pin Name	Wakeup Interrup	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P5.7	Y	AD19	TKB15	LSEG7		PWM2N		
P5.6	Y	AD18	TKB14	LSEG6		PWM2P		
P5.5	Y	AD09	TKB13	LSEG5			TXD1B	
P5.4	Y	AD08	TKB12	LSEG4			RXD1B	
P5.3	Y	AD07	TKB11	LSEG3		PWM2N		
P5.2	Y	AD06	TKB10	LSEG2		PWM2P		
P5.1	Y	AD05	TKB09	LSEG1		PWM2N		
P5.0	Y	AD04	TKB08	LSEG0		PWM2P		

Port5 multi-function Table

The necessary SFR setting for Port0~Port5 pin’s alternative function is list below.

Alternative Function	PINMODxx	Px.n SFR data	Pin State	Other necessary SFR setting
T0, T1, T2, T2EX, INT0, INT1	**00x0**	1	Input with Pull-up	
	00x1	1	Input	
RXD0x RXD1x RXD2	**xxx0**	1	UART RX (Input with Pull-up)	PINMODE
	xxx1	1	UART RX (Input)	
TXD0x TXD1x TXD2	**xx10**	1	UART TX output (CMOS Push-Pull)	
XI, XO	**0011**	1	Crystal oscillation	CLKCON
SPI Master Mode MISO	**xx01**	1	SPI Data Input	SPCON
SPI Master Mode SCK, MOSI	**xx10**	X	SPI Clock/Data Output (CMOS Push-Pull)	
SPI Slave Mode MISO	**xx10**	X	SPI Data Output (CMOS Push-Pull)	
SPI Slave Mode SCK, MOSI	**xx01**	1	SPI Clock/Data Input	
NSS	**xx01**	1	SPI Chip Selection	
I^2C Master SCL	**0x00**	X	I^2C Clock Output (Open Drain Output, Pull-up)	PINMODE
	xx10	X	I^2C Clock Output (CMOS Push-Pull)	
I^2C Slave SCL	**xx01**	1	I^2C Clock Input (Hi-Z)	
I^2C Master/Slaver SDA	**xx00**	1	I^2C DATA (Pull-up)	
AD00~AD41	**0011**	X	ADC Channel	ADCHS
TKA00~TKA20 TKB00~TKB20			Touch Key Channel	TKCHSA TKCHSB ATKCHA0/1/2 ATKCHB0/1/2
LCOM0~ LCOM7 LSEG0~ LSEG7	**0111**	X	LED Output (matrix mode)	LXDCON LXDCON2
LED0~ LED7			LED Output (dot matrix mode)	
COM0~COM7 SEG0~SEG19			LCD output	
T0O, T2O	**1011**	X	Clock Output (CMOS Push-Pull)	
PWMx			PWM Output (CMOS Push-Pull)	PWMEN

Mode Setting for Port0~Port5 Alternative Function

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

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

An 8051 standard pin is a “**Pseudo Open Drain**” pin. It can sink at least 4 mA current when output is at low level, and drives at least 4 mA current for 1~2 clock cycle when output transits from low to high, then keeps driving a small current (<20 μA) to maintain the pin at high level. It can be used as input or output function.

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 **E8h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P4	P4.7	P4.6	P4.5	P4.4	P4.3	P4.2	P4.1	P4.0
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

E8h.7~0 **P4:** Port4 data

SFR **C0h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P5	P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0
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

C0h.7~0 **P5:** Port5 data

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

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

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

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

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

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

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

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

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

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

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

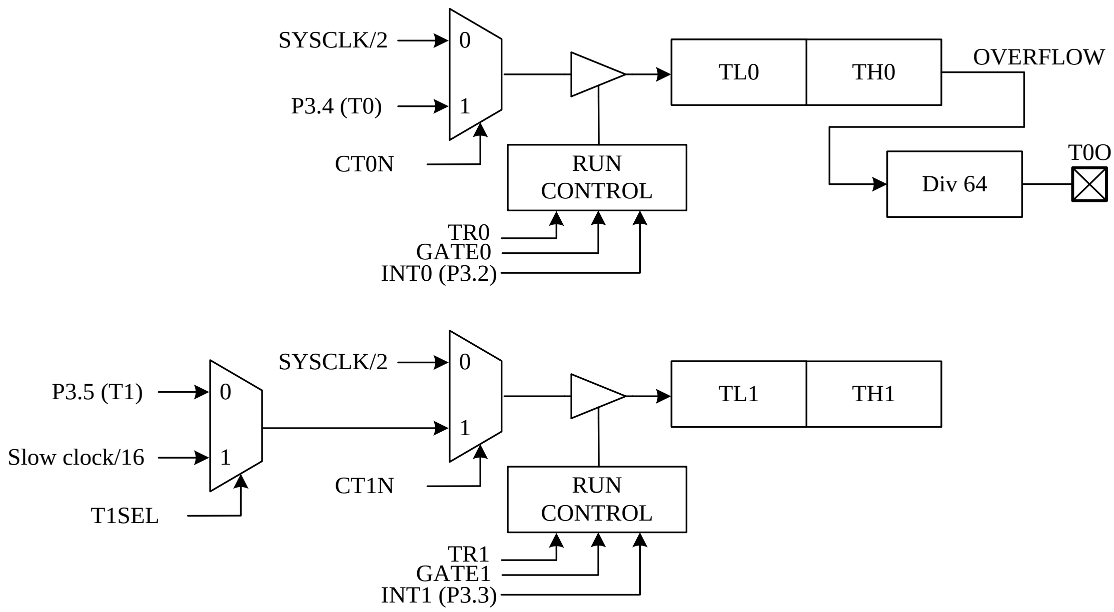
A5h.3~0 **PINMOD6:** Px.6 pin control, port index (x) is defined by PORTIDX
0000~1111: see table 7.1

8. Timers

Timer0, Timer1 and Timer2 are provided as standard 8051 compatible timer/counter. Compare to the traditional 12T 8051, the Chip's Timer0/1/2 use 2 System clock cycle as the time base unit. That is, in timer mode, these timers increase at every “2 System clock” rate; in counter mode, T0/T1/T2 pin input pulse must be wider than 2 System clock to be seen by this device. In addition to the standard 8051 timers function. The T0O pin can output the “Timer0 overflow divided by 64” signal, and the T2O pin can output the “Timer2 overflow divided by 2” signal. Timer3 is provided for a real-time clock count, when its time base is SXT.

8.1 Timer0 / Timer1

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



Timer0 and Timer1 Structure

SFR **88h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
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

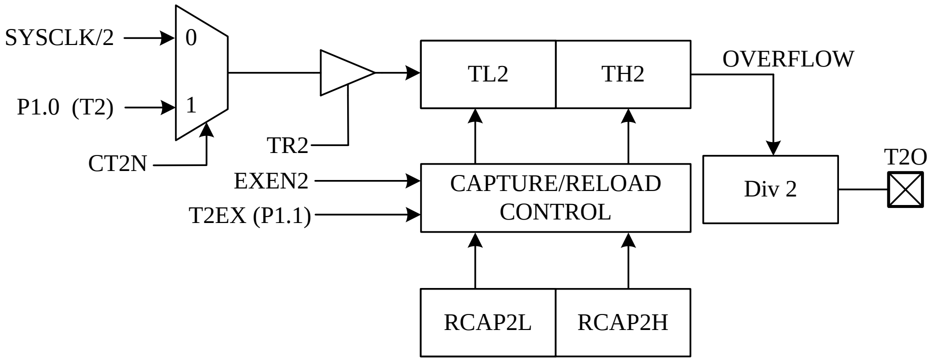
SFR **8Dh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TH1	TH1							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

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

Note: *See also Chapter 6 for more information on Timer0/1 interrupt enable and priority.*
Note: *See also Chapter 7 for details on T0O pin output settings.*

8.2 Timer2

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



Timer2 Structure

SFR **C8h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
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:** UART receive clock control bit
0: Use Timer1 overflow as receive clock for serial port in mode 1 or 3
1: Use Timer2 overflow as receive clock for serial port in mode 1 or 3

C8h.4 **TCLK:** UART transmit clock control bit
0: Use Timer1 overflow as transmit clock for serial port in mode 1 or 3
1: Use Timer2 overflow as transmit clock for serial port in mode 1 or 3

C8h.3 **EXEN2:** T2EX pin enable
0: T2EX pin disable
1: T2EX pin enable, it cause a capture or reload when a negative transition on T2EX pin is detected if RCLK=TCLK=0

C8h.2 **TR2:** Timer2 run control
0: Timer2 stops
1: Timer2 runs

C8h.1 **CT2N:** Timer2 Counter/Timer select bit
0: Timer mode, Timer2 data increases at 2 System clock cycle rate
1: Counter mode, Timer2 data increases at T2 pin’s negative edge

C8h.0 **CPRL2N:** Timer2 Capture/Reload control bit
0: Reload mode, auto-reload on Timer2 overflows or negative transitions on T2EX pin if EXEN2=1.
1: Capture mode, capture on negative transitions on T2EX pin if EXEN2=1.
If RCLK=1 or TCLK=1, CPRL2N is ignored and timer is forced to auto-reload on Timer2 overflow.

SFR **CAh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
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

SFR **F8h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Reset	0	0	0	0	–	0	0	0

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

Note: *See also Chapter 6 for more information on Timer2 interrupt enable and priority.*
Note: *See also Chapter 7 for details on T2O pin output settings.*

8.3 Timer3

Timer3 works as a time-base counter, which generates interrupts periodically. It generates an interrupt flag (TF3) with the clock divided by 65536, 16384, 4096, or 1024 depending on the TM3PSC SFR. The Timer3 clock source can be selected as SLOW clock (SRC or SXT) or FRC/512. This is ideal for real-time-clock (RTC) functionality when the clock source is SXT.

SFR **94h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	TM3CKS		WDTPSC		ADCKS		TM3PSC	
R/W	R/W	R/W	R/W		R/W		R/W	
Reset	0	0	0	0	0	0	0	0

94h.6 **TM3CKS:** Timer3 clock source select
00: SLOW clock (SXT/SRC)
01: FRC/512
10: SLOW clock (SXT/SRC) / 2
11: FRC/1024

94h.1~0 **TM3PSC:** Timer3 Interrupt rate
00: Timer3 Interrupt rate is 65536 Timer3 clock source cycle
01: Timer3 Interrupt rate is 16384 Timer3 clock source cycle
10: Timer3 Interrupt rate is 4096 Timer3 clock source cycle
11: Timer3 Interrupt rate is 1024 Timer3 clock source cycle

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

95h.0 **TF3:** Timer3 Interrupt Flag
Set by H/W when Timer3 reaches TM3PSC setting cycles. Cleared automatically when the program performs the interrupt service routine. S/W can write FEh to INTFLG to clear this bit. (***Note1***)

SFR **F8h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Reset	0	0	0	0	–	0	0	0

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

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

8.4 T0O and T2O Output Control

This device can generate various frequency waveform pin output (in CMOS or Open-Drain format) for Buzzer. The T0O and T2O waveform is divided by Timer0/Timer2 overflow signal. The T0O waveform is Timer0 overflow divided by 64, and T2O waveform is Timer2 overflow divided by 2. User can control their frequency by Timers auto reload speed. Set the MODE of P3.4 or P1.0 to 1011b to output T0O and T2O. See table 7.1 for more detail.

9. UARTs

This Chip has three UARTs, UART0, UART1 and UART2.

The **UART0** is a standard 8051's full duplex UART. The UART0 use SCON and SBUF SFRs. SCON is the control register, SBUF is the data register. Data is written to SBUF for transmission and SBUF is read to obtain received data. The received data and transmitted data registers are completely independent.

The **UART1** uses SCON1 and SBUF1 SFRs. SCON1 is the control register, SBUF1 is the data register. Data is written to SBUF1 for transmission and SBUF1 is read to obtain received data. The received data and transmitted data registers are completely independent. The UART1 supports most of the functions of UART, but it does not support Mode0 and Mode2.

The **UART2** uses SCON2 and SBUF2 SFRs. SCON2 is the control register, SBUF2 is the data register. Data is written to SBUF2 for transmission and SBUF2 is read to obtain received data. The received data and transmitted data registers are completely independent. The UART2 supports most of the functions of UART, but it does not support Mode0 and Mode2.

UART0 BAUD rate setting: while SFR **UART0BRS=0**
UART0 BAUD rate set as standard 8051 as following.

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

UART0 BAUD rate setting: while SFR **UART0BRS=1**

- Mode 0: Baud Rate= $F_{SYSCLK}/2$
- Mode 1: Baud Rate= Fsys/32/UART0BRP
- Mode 2: Baud Rate= (SMOD + 1) x $F_{SYSCLK}/64$
- Mode 3: Baud Rate= Fsys/32/UART0BRP

UART1 BAUD rate setting:

- Mode 0, 2: invalid
- Mode 1, 3: Baud Rate= Fsys/32/UART1BRP

UART2 BAUD rate setting:

- Mode 0, 2: invalid
- Mode 1, 3: Baud Rate= Fsys/32/UART2BRP

Note: *also refer to Section 6 for more information about UART Interrupt enable and priority.*
Note: *also refer to Section 8 for more information about how Timer2 controls UART clock.*

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.7 **SMOD:** UART0 double baud rate control bit
0: Disable UART0 double baud rate
1: Enable UART0 double baud rate

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:** UART0 serial port mode select bit 0,1
00: Mode0: 8 bit shift register, Baud Rate=F_{SYSCLK}/2
01: Mode1: 8 bit UART0, Baud Rate is variable
10: Mode2: 9 bit UART0, Baud Rate=F_{SYSCLK}/32 or/64
11: Mode3: 9 bit UART0, 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:** UART0 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							
R/W	R/W							
Reset	–	–	–	–	–	–	–	–

99h.7~0 **SBUF:** UART0 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.

SFR **8Eh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCON2	SM2S	–	–	REN2	TB82	RB82	TI2	RI2
R/W	R/W	–	–	R/W	R/W	R/W	R/W	R/W
Reset	0	–	–	0	0	0	0	0

8Eh.7 **SM2S:** UART2 Serial port mode select bit
0: Mode1: 8 bit UART2, Baud Rate is variable
1: Mode3: 9 bit UART2, Baud Rate is variable
(UART2 does not support Mode0/Mode2)

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

8Eh.3 **TB82:** UART2 Transmit Bit 8, the ninth bit to be transmitted in Mode 3

8Eh.2 **RB82:** UART2 Receive Bit 8, contains the ninth bit that was received in Mode3

8Eh.1 **TI2:** UART2 Transmit interrupt flag
Set by H/W at the beginning of the stop bit in Mode 1 & 3. Must be cleared by S/W.

8Eh.0 **RI2:** UART2 Receive interrupt flag
Set by H/W at the sampling point of the stop bit in Mode 1 & 3. Must be cleared by S/W.

SFR **8Fh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SBUF2	SBUF2							
R/W	R/W							
Reset	–	–	–	–	–	–	–	–

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

SFR **9Ah**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCON1	SM1S	–	–	REN1	TB81	RB81	TI1	RI1
R/W	R/W	–	–	R/W	R/W	R/W	R/W	R/W
Reset	0	–	–	0	0	0	0	0

9Ah.7 **SM1S:** UART1 Serial port mode select bit
0: Mode1: 8 bit UART1, Baud Rate is variable
1: Mode3: 9 bit UART1, Baud Rate is variable
(UART1 does not support Mode0/Mode2)

9Ah.4 **REN1:** UART1 reception enable
0: Disable reception
1: Enable reception

9Ah.3 **TB81:** UART1 Transmit Bit 8, the ninth bit to be transmitted in Mode 3

9Ah.2 **RB81:** UART1 Receive Bit 8, contains the ninth bit that was received in Mode3

9Ah.1 **TI1:** UART1 Transmit interrupt flag
Set by H/W at the beginning of the stop bit in Mode 1 & 3. Must be cleared by S/W.

9Ah.0 **RI1:** UART1 Receive interrupt flag
Set by H/W at the sampling point of the stop bit in Mode 1 & 3. Must be cleared by S/W.

SFR **9Bh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SBUF1	SBUF1							
R/W	R/W							
Reset	–	–	–	–	–	–	–	–

9Bh.7~0 **SBUF1:** UART1 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.

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

A9h.5 **ES2:** Serial Port (UART1/UART2) interrupt enable
0: Disable Serial Port (UART1/UART2) interrupt
1: Enable Serial Port (UART1/UART2) interrupt

SFR **D5h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
UART2CON	–	UART2BRP						
R/W	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	0	0	0	0	0	0	0

D5h.6~0 **UART2BRP:** UART2 baud rate pre-scaler

SFR **DDh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
UART1CON	–	UART1BRP						
R/W	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	0	0	0	0	0	0	0

DDh.6~0 **UART1BRP:** UART1 baud rate pre-scaler

SFR **DEh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
UART0CON	UART0BRS	UART0BRP						
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

DEh.7 **UART0BRS:** UART0 baud rate source select
DEh.6~0 **UART0BRP:** UART0 baud rate pre-scaler

10. PWMs

The Chip has nine 16-bit PWM modules PWM0, PWM1, PWM2 and PWM30~P35. PWM0~2 have independent 16 bit period. PWM30~P35 share a group of 16 bit period. The following takes PWM0 as an example for description. The PWM can generate varies frequency waveform with 65536 duty resolution on the basis of the PWM clock. The PWM clock can select FRC double frequency (FRC x 2), FRC or F_{SYSCLK} as its clock source.

Using the SFR PINMODx to controls the PWM output to IO and set PWMxEN to enable PWM function. For example, PORTIDX=1, PIMOD76=BBh and PWMxEN=1, then PWM1 and PWM0 will output to IO. (***see section 7***)

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

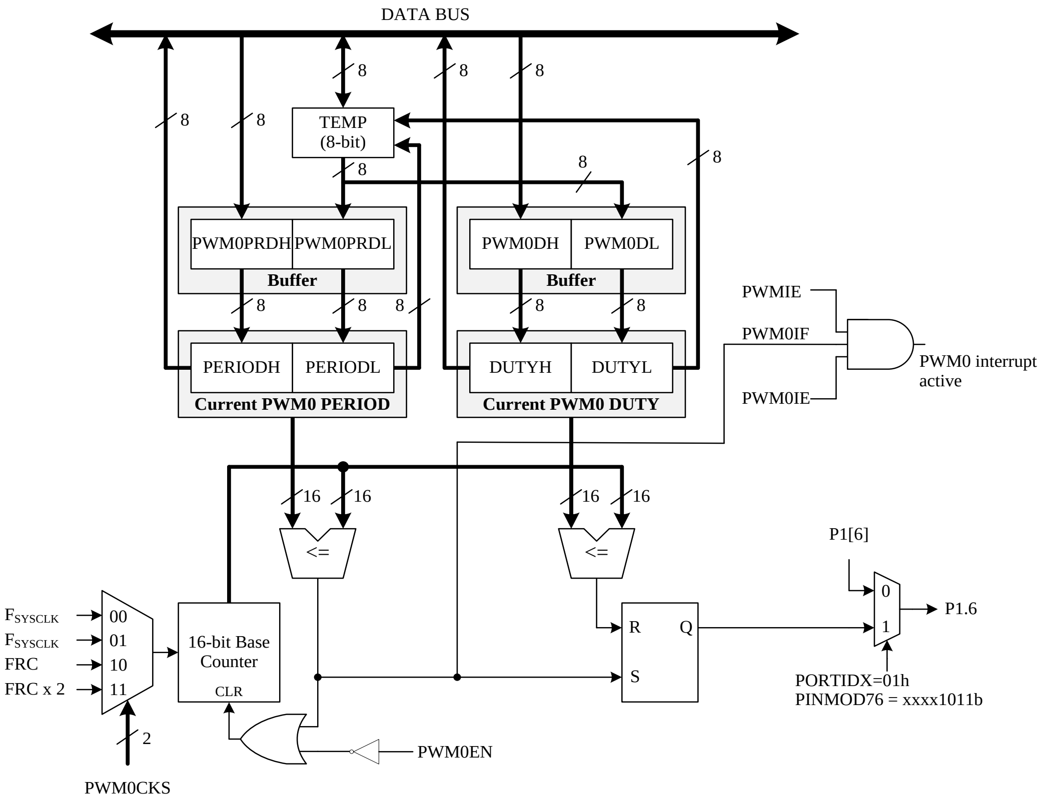
When PWM0EN bit is set, the PWM0 will be running, otherwise the PWM0 is stop. The PWM0 structure is shown as follow. The PWM0 duty cycle can be changed by writing to PWMDH and PWMDL while PWMIDX=0. The PWM0 output signal resets to a low level whenever the 16-bit base counter matches the 16-bit PWM0 duty register {PWM0DH, PWM0DL}. The PWM0 period can be set by writing the period value to the PWMPRDH and PWMPRDL registers while PWMIDX=0. After writing the PWM duty or period register, the new values will immediately save to their own buffer. H/W will update these values at the end of current period or while PWM0 is cleared. PWM0~3 has a corresponding interrupt flag, and an interrupt flag is generated at the end of the period.

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

As shown in the table below, using PWMIDX to set the 16 bit period and duty cycle of PWM0~3 is as follows

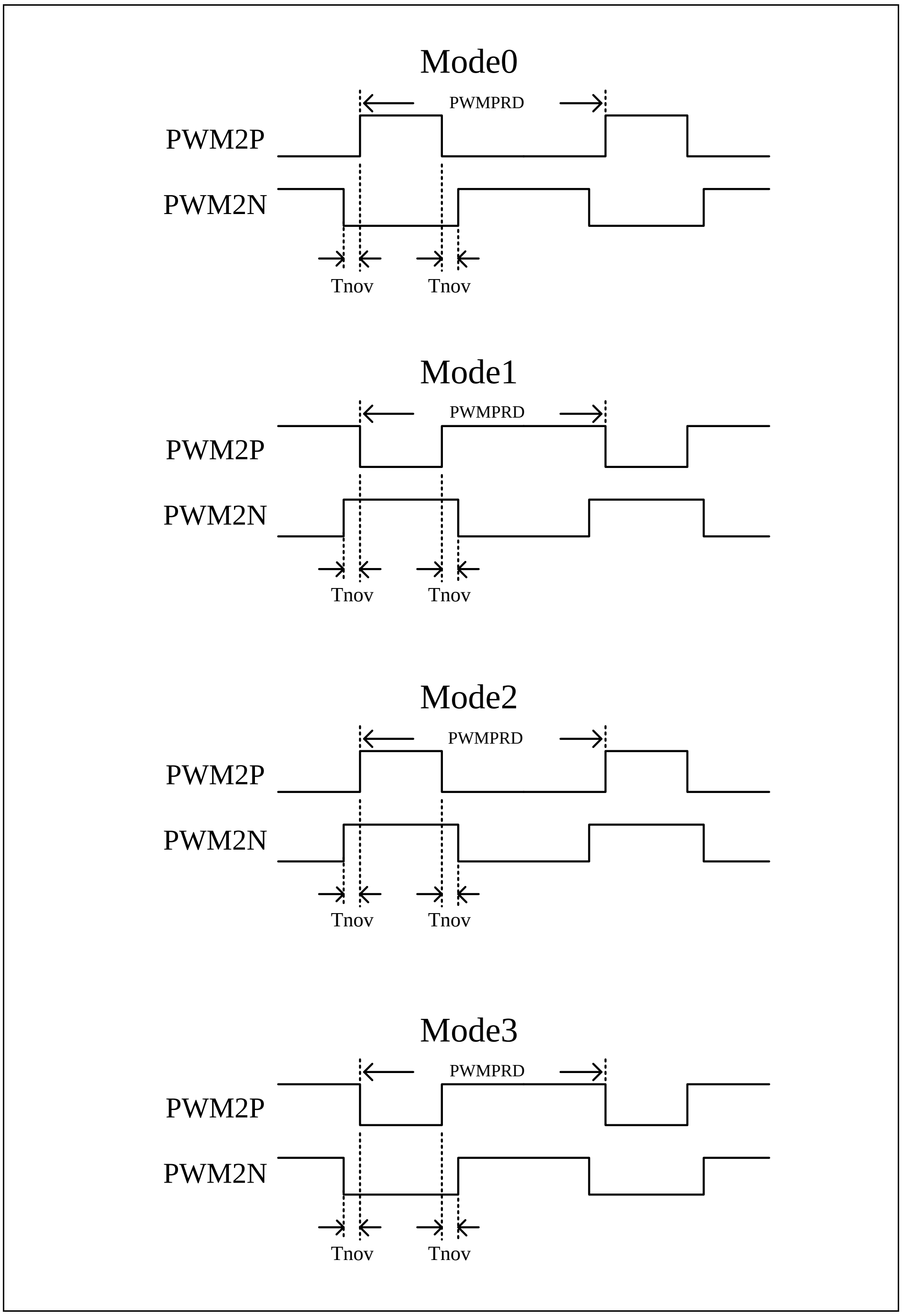
<table>
<tr><th>PWMIDX
(SFR 9Eh)</th><th>PWMPRDH
(SFR D9h)</th><th>PWMPRDL
(SFR DAh)</th><th>PWMDH
(SFR D1h)</th><th>PWMDL
(SFR D2h)</th></tr>
<tr><td>0xh</td><td>PWM0PRDH</td><td>PWM0PRDL</td><td>PWM0DH</td><td>PWM0DL</td></tr>
<tr><td>1xh</td><td>PWM1PRDH</td><td>PWM1PRDL</td><td>PWM1DH</td><td>PWM1DL</td></tr>
<tr><td>2xh</td><td>PWM2PRDH</td><td>PWM2PRDL</td><td>PWM2DH</td><td>PWM2DL</td></tr>
<tr><td>30h</td><td rowspan="6">PWM3PRDH</td><td rowspan="6">PWM3PRDL</td><td>PWM30DH</td><td>PWM30DL</td></tr>
<tr><td>31h</td><td>PWM31DH</td><td>PWM31DL</td></tr>
<tr><td>32h</td><td>PWM32DH</td><td>PWM32DL</td></tr>
<tr><td>33h</td><td>PWM33DH</td><td>PWM33DL</td></tr>
<tr><td>34h</td><td>PWM34DH</td><td>PWM34DL</td></tr>
<tr><td>35h</td><td>PWM35DH</td><td>PWM35DL</td></tr>
</table>

Table 10.1 PWM0~2, PWM30~PWM35 period and duty index table



PWM0 Structure

Only PWM2 can be output via PWM2P and PWM2N with four different modes. The edges of the PWM pulse can be separated with 16 different time non-overlap clocks intervals (Tnov). The width of Tnov can be selected by PWM2DZ within 0~15 pwm clock. The default output form is Mode2. The waveforms of the four output modes are shown below.



PWM2 Waveform Modes

SFR **86h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTPWM	–	–	–	–	PWM3IF	PWM2IF	PWM1IF	PWM0IF
R/W	–	–	–	–	R/W	R/W	R/W	R/W
Reset	–	–	–	–	0	0	0	0

86h.3 **PWM3IF:**
0: S/W write 0 to clear it
1: Set by H/W at the end of the period

86h.2 **PWM2IF:**
0: S/W write 0 to clear it
1: Set by H/W at the end of the period

86h.1 **PWM1IF:**
0: S/W write 0 to clear it
1: Set by H/W at the end of the period

86h.0 **PWM0IF:**
0: S/W write 0 to clear it
1: Set by H/W at the end of the period

SFR **9Dh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2CON	PWM2OM		PWM2DZ					
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0

9Dh.7~6 **PWM2OM:** output mode
00: mode 0
01: mode 1
10: mode 2
11: mode 3

9Dh.5~0 **PWM2DZ:** PWM2 Dead zone Control
0000: dead zone disabled
0001: Dead zone width 1*Tpwmclk
0010: Dead zone width 2*Tpwmclk
…
1111: Dead zone width 15*Tpwmclk

SFR **9Eh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMIDX	PWMIDX							
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

9Eh.7~0 **PWMIDX:** PWM period and duty index. See table 10.1 for more detail
0xh: PWM0 Period/Duty access
1xh: PWM1 Period/Duty access
2xh: PWM2 Period/Duty access
3xh: PWM30~PWM35 Period/Duty access
30h: PWM30 Period/Duty access
31h: PWM31 Period/Duty access
32h: PWM32 Period/Duty access
33h: PWM33 Period/Duty access
34h: PWM34 Period/Duty access
35h: PWM35 Period/Duty access

SFR **9Fh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMEN	PWM3IE	PWM2IE	PWM1IE	PWM0IE	PWM3EN	PWM2EN	PWM1EN	PWM0EN
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

9Fh.7 **PWM3IE:** PWM3 Interrupt Enable
0: disable
1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)

9Fh.6 **PWM2IE:** PWM2 Interrupt Enable
0: disable
1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)

9Fh.5 **PWM1IE:** PWM1 Interrupt Enable
0: disable
1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)

9Fh.4 **PWM0IE:** PWM0 Interrupt Enable
0: disable
1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)

9Fh.3 **PWM3EN:**
0: PWM3 is cleared and held 1: PWM3 is running

9Fh.2 **PWM2EN:**
0: PWM2 is cleared and held 1: PWM2 is running

9Fh.1 **PWM1EN:**
0: PWM1 is cleared and held 1: PWM1 is running

9Fh.0 **PWM0EN:**
0: PWM0 is cleared and held 1: PWM0 is running

SFR **A1h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCON	PWM3CKS		PWM2CKS		PWM1CKS		PWM0CKS	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	1	0	1	0	1	0

A1h.7~6 **PWM3CKS:** PWM3 Clock source
00: F_{SYSCLK}
01: F_{SYSCLK}
10: FRC
11: FRC x 2

A1h.5~4 **PWM2CKS:** PWM2 Clock source
00: F_{SYSCLK}
01: F_{SYSCLK}
10: FRC
11: FRC x 2

A1h.3~2 **PWM1CKS:** PWM1 Clock source
00: F_{SYSCLK}
01: F_{SYSCLK}
10: FRC
11: FRC x 2

A1h.1~0 **PWM0CKS:** PWM0 Clock source
00: F_{SYSCLK}
01: F_{SYSCLK}
10: FRC
11: FRC x 2

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

A9h.6 **PWMIE:** PWM0~3 interrupt enable
0: Disable PWM0~3 interrupt
1: Enable PWM0~3 interrupt

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

D1h.7~0 **PWMDH:** PWM duty high byte, index by PWMIDX
See table 10.1 for more detail
PWMIDX = 0xh: PWM0DH access
PWMIDX = 1xh: PWM1DH access
PWMIDX = 2xh: PWM2DH access
PWMIDX = 30h: PWM30DH access
PWMIDX = 31h: PWM31DH access
PWMIDX = 32h: PWM32DH access
PWMIDX = 33h: PWM33DH access
PWMIDX = 34h: PWM34DH access
PWMIDX = 35h: PWM35DH access

Note :
write sequence: PWMDL then PWMDH
read sequence: PWMDH then PWMDL

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

D2h.7~0 **PWMDL:** PWM duty low byte, index by PWMIDX
See table 10.1 for more detail
PWMIDX = 0xh: PWM0DH access
PWMIDX = 1xh: PWM1DH access
PWMIDX = 2xh: PWM2DH access
PWMIDX = 30h: PWM30DH access
PWMIDX = 31h: PWM31DH access
PWMIDX = 32h: PWM32DH access
PWMIDX = 33h: PWM33DH access
PWMIDX = 34h: PWM34DH access
PWMIDX = 35h: PWM35DH access

Note :
write sequence: PWMDL then PWMDH
read sequence: PWMDH then PWMDL

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

D9h.7~0 **PWMPRDH:** PWM period high byte, index by PWMIDX
See table 10.1 for more detail
PWMIDX = 0xh: PWM0PRDH access
PWMIDX = 1xh: PWM1PRDH access
PWMIDX = 2xh: PWM2PRDH access
PWMIDX = 3xh: PWM3PRDH access

Note :
write sequence: PWMPRDL then PWMPRDH
read sequence: PWMPRDH then PWMPRDL

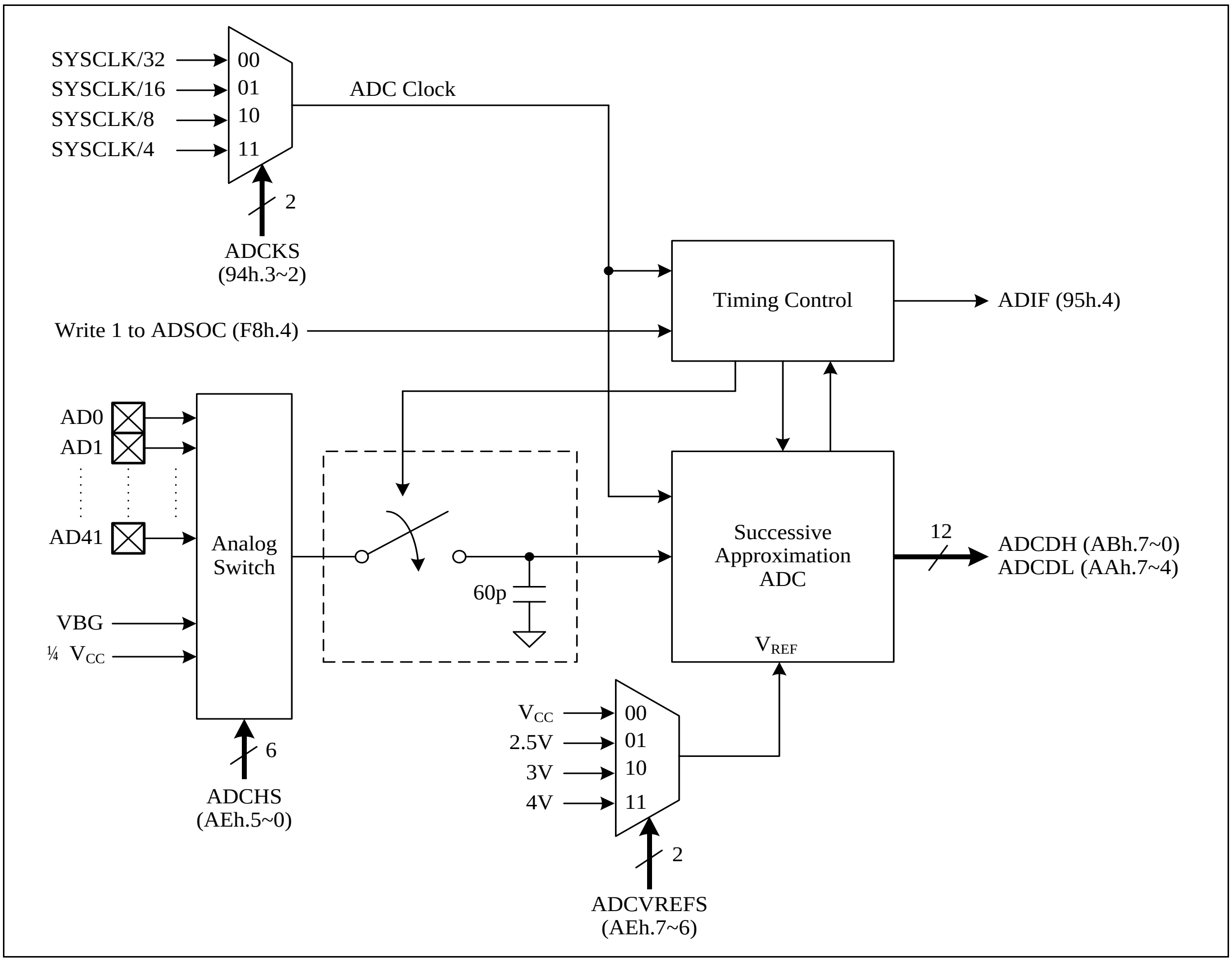
SFR **DAh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMPRDL	PWMPRDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

Dah.7~0 **PWMPRDL:** PWM period low byte, index by PWMIDX
See table 10.1 for more detail
PWMIDX = 0xh: PWM0PRDH access
PWMIDX = 1xh: PWM1PRDH access
PWMIDX = 2xh: PWM2PRDH access
PWMIDX = 3xh: PWM3PRDH access

Note :
write sequence: PWMPRDL then PWMPRDH
read sequence: PWMPRDH then PWMPRDL

11. ADC

The Chip offers a 12-bit ADC consisting of a 44-channel analog input multiplexer, control register, clock generator, 12-bit successive approximation register, and output data register. To use the ADC, set the ADCKS bit first to choose a proper ADC clock frequency, which must be less than 1 MHz. Then, launch the ADC conversion by setting the ADSOC bit, and H/W will automatic clear it at the end of the conversion. After the end of the conversion, H/W will set the ADIF bit and generate an interrupt if an ADC interrupt is enabled. The ADIF bit can be cleared by writing 0 to this bit or 1 to the ADSOC bit. Because certain channels are shared with the Touch Key, the ADC channel must be configured differently from the Touch Key channel to avoid affecting the channel input sensitivity. The VREF of the ADC can be selected from the following four voltages: V_{CC}, 2.5V, 3V and 4V. When ADCHS is selected to VBG, ADCVREFS must be set to V_{CC}, otherwise ADC conversion will be invalid.

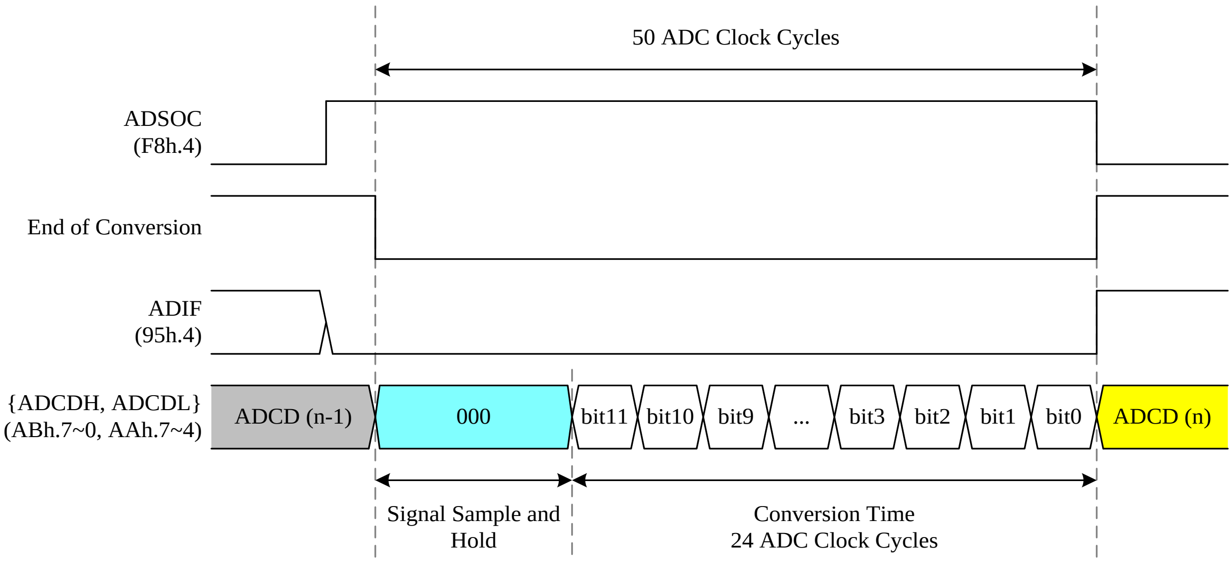


11.1 ADC Channels

The 12-bit ADC has a total of 44 channels, designated AD0~AD41, VBG and 1/4V_{CC}. The ADC channels are connected to the analog input pins via the analog switch multiplexer. The analog switch multiplexer is controlled by the ADCHS register. VBG is an internal voltage reference at 1.22V. When ADC channel select to VBG, VBG generator will enable automatically. User can get more stable VBG voltage by setting SFR VBGEN=1 to always enable VBG generator. When ADCHS is selected to VBG, ADCVREFS must be set to V_{CC}, otherwise ADC conversion will be invalid.

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 50 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		TM3PSC	
R/W	R/W	R/W	R/W		R/W		R/W	
Reset	0	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	–	TKIFA	ADIF	–	–	PCIF	TF3
R/W	R	–	R/W	R/W	–	–	R/W	R/W
Reset	–	–	0	0	–	–	0	0

95h.4 **ADIF:** ADC interrupt flag
Set by H/W at the end of ADC conversion. S/W writes EFh to INTFLG or sets the ADSOC bit to clear this flag. (***Note1***)

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

PINMODE	VBGEN	–	UART1PS	PSEUDOEN	I2CPS		UART0PS	
R/W	R/W	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	–	0	0	0	0	0	0

A6h.7 **VBGEN:** force VBG generator enable
0: VBG generator is automatically enable and disable
1: Force VBG generator enable included in IDLE mode but disabled in Stop/Halt mode

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
CHSEL	ADCVREFS		ADCHS					
R/W	R/W		R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	1	1	1	1	1

AEh.7~6 **ADCVREFS:** ADC reference voltage. When ADCHS is selected to VBG, ADCVREFS must be set to VCC, otherwise ADC conversion will be invalid
00: VCC
01: 2.5V
10: 3V
11: 4V

AEh.5~0 **ADCHS:** ADC channel select
000000: AD00
000001: AD01
…
101001: AD41
101011: V_{BG} (Internal Bandgap Reference Voltage)
101100: 1/4V_{CC} (Internal Reference Voltage)

SFR **F8h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Reset	0	0	0	0	–	0	0	0

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

12. Touch Key (FTK)

The Touch Key offers an easy simple and reliable method to implement finger touch detection. During the key scan operation, the device support 2 groups of 21 channels touch key detection.

To use the Touch Key, user should setup correctly. There are two ways to set IO as TK channel. Set SFR PINMODx to 0011b or set SFR TKPINSA/B 0~2 to force IO as TK channel. If TKPINSA/B 0~2 are set, the corresponding IO pins will be fixed as TK channels and will no longer be affected by PINMODx.

TKPINSA	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSA0	TKA07	TKA06	TKA05	TKA04	TKA03	TKA02	TKA01	TKA00
TKPINSA1	TKA15	TKA14	TKA13	TKA12	TKA11	TKA10	TKA09	TKA08
TKPINSA2				TKA20	TKA19	TKA18	TKA17	TKA16

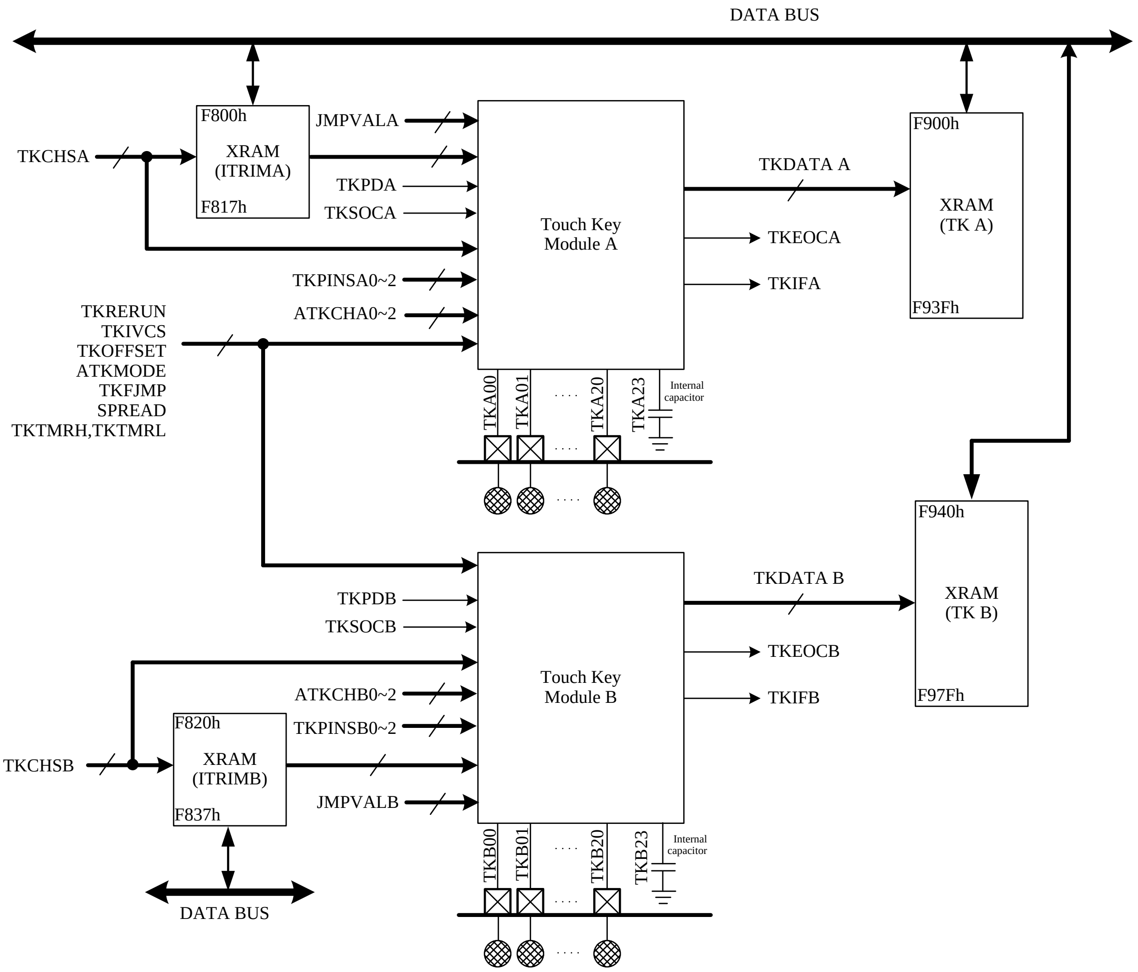
Set TKPINSA0~2 to fix IO as TKA channel

TKPINSA	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSB0	TKB07	TKB06	TKB05	TKB04	TKB03	TKB02	TKB01	TKB00
TKPINSB1	TKB15	TKB14	TKB13	TKB12	TKB11	TKB10	TKB09	TKB08
TKPINSB2				TKB20	TKB19	TKB18	TKB17	TKB16

Set TKPINSB0~2 to fix IO as TKB channel

In the TK Mode, user assigns TKPD=0 to turn on the TK module, then set the TKSOC bit to start touch key conversion, the TKSOC bit can be automatically cleared while end of conversion. However, if the F_{SYSCLK} is too slow, H/W might fail to clear TKSOC due to clock sampling rate. TKEOC=0 means conversion is in process. TKEOC=1 means the conversion is finish, and the touch key counting result is stored into the XRAM. After TKEOC=1, user must wait at least 50 µs for next conversion. But if TKRERUN = 1, TK will always be converted, and there is no need to set TKSOC for each conversion. Reducing/increasing TKTMR can reduce/increase the TKDATA to accommodate the condition of the system.

The FTK has an internal built-in reference capacitor to simulate the KEY behavior. Set TKCHS=17h and start the scanning can get the TK Data Count of internal reference capacitor (TKCAP). Since the internal capacitor would not be affected by water or mobile phone, it is useful for comparing the environment background noise. Setting the TKFJMP, the frequency of Touch Key clock can be change automatically by H/W controlled. It may help to improve the ability to resist noise.



FTK Structure

SFR ATKCHA/B0~2 are used to specify scan TK channel, and each bit is mapped to TK pin. TK scan will scan from low bit to high bit. If ATKMODE = 0, TK can scan up to 22 channels, TK00~TK20 and TKCAP (TK23), each channel is scanned once. If ATKMODE = 1, TK can scan up to 16 channels, each channel is scanned twice. If ATKMODE = 2, TK can scan up to 8 channels, each channel is scanned 4 times. If ATKMODE = 3, TK can scan up to 4 channels, each channel is scanned 8 times. TKCHSA and TKCHSB is used to specify the first channel for TK to start scanning.

For example:

Condition ATKMODE=0, scan TKA16/TKA14/TKA08/TKA07/TKA06/TKA02

⇨ TKPINSA2=0000_0001, TKPINSA1=0100_0001, TKPINSA0=1100_0100
⇨ ATKCHA2=0000_0001, ATKCHA1=0100_0001, ATKCHA0=1100_0100
⇨ TKCHSA=0x02 (Specify the first scan channel)

The arrangement of TK data stored in XRAM is as follows.

	XRAM
F900h	TKA00 DATAL
F901h	TKA00 DATAH
F902h	TKA01 DATAL
F903h	TKA01 DATAH
	…
F928h	TKA20 DATAL
F929h	TKA20 DATAH
	…
F92Eh	TKA23 DATAL
F92Fh	TKA23 DATAH

Condition ATKMODE=1, scan TKA16/TKA14/TKA08/TKA07/TKA06/TKA02

⇨ TKPINSA2=0000_0001, TKPINSA1=0100_0001, TKPINSA0=1100_0100
⇨ ATKCHA2=0000_0001, ATKCHA1=0100_0001, ATKCHA0=1100_0100
⇨ TKCHSA=0x02 (Specify the first scan channel)

The arrangement of TK data stored in XRAM is as follows.

	XRAM
F900h	TKA02 1st DATAL
F901h	TKA02 1st DATAH
F902h	TKA02 2nd DATAL
F903h	TKA02 2nd DATAH
F904h	TKA06 1st DATAL
F905h	TKA06 1st DATAH
F906h	TKA06 2nd DATAL
F907h	TKA06 2nd DATAH
	…
F914h	TKA16 1st DATAL
F915h	TKA16 1st DATAH
F916h	TKA16 2nd DATAL
F917h	TKA16 2nd DATAH

The TK scan result is 14-bit data, which are DATAH 6-bit and DATAL 8-bit. DATAH/L must be read in order to get the correct 14-bit data: first read the low byte (DATAL), then read the high word byte (DATAH)

Condition ATKMODE=2, scan TKA16/TKA14/TKA08/TKA07/TKA06/TKA02

⇨ TKPINSA2=0000_0001, TKPINSA1=0100_0001, TKPINSA0=1100_0100
⇨ ATKCHA2=0000_0001, ATKCHA1=0100_0001, ATKCHA0=1100_0100
⇨ TKCHSA=0x02 (Specify the first scan channel)

The arrangement of TK data stored in XRAM is as follows.

	XRAM
F900h	TKA02 1st DATAL
F901h	TKA02 1st DATAH
F902h	TKA02 2nd DATAL
F903h	TKA02 2nd DATAH
F904h	TKA02 3rd DATAL
F905h	TKA02 3rd DATAH
F906h	TKA02 4th DATAL
F907h	TKA02 4th DATAH
F908h	TKA06 1st DATAL
F909h	TKA06 1st DATAH
F90Ah	TKA06 2nd DATAL
F90Bh	TKA06 2nd DATAH
F90Ch	TKA06 3rd DATAL
F90Dh	TKA06 3rd DATAH
F90Eh	TKA06 4th DATAL
F90Fh	TKA06 4th DATAH
	…
F928h	TKA16 1st DATAL
F929h	TKA16 1st DATAH
F92Ah	TKA16 2nd DATAL
F92Bh	TKA16 2nd DATAH
F92Ch	TKA16 3rd DATAL
F92Dh	TKA16 3rd DATAH
F92Eh	TKA16 4th DATAL
F92Fh	TKA16 4th DATAH

The TK scan result is 14-bit data, which are DATAH 6-bit and DATAL 8-bit. DATAH/L must be read in order to get the correct 14-bit data: first read the low byte (DATAL), then read the high word byte (DATAH)

Condition ATKMODE=3, scan TKA08/TKA07/TKA06/TKA02

- ⇨ TKPINSA2=0000_0000, TKPINSA1=0000_0001, TKPINSA0=1100_0100
- ⇨ ATKCHA2=0000_0000, ATKCHA1=0000_0001, ATKCHA0=1100_0100
- ⇨ TKCHSA=0x02 (Specify the first scan channel)

The arrangement of TK data stored in XRAM is as follows.

	XRAM
F900h	TKA02 1st DATAL
F901h	TKA02 1st DATAH
F902h	TKA02 2nd DATAL
F903h	TKA02 2nd DATAH
F904h	TKA02 3rd DATAL
F905h	TKA02 3rd DATAH
F906h	TKA02 4th DATAL
F907h	TKA02 4th DATAH
F908h	TKA02 5th DATAL
F909h	TKA02 5th DATAH
F90Ah	TKA02 6th DATAL
F90Bh	TKA02 6th DATAH
F90Ch	TKA02 7th DATAL
F90Dh	TKA02 7th DATAH
F90Eh	TKA02 8th DATAL
F90Fh	TKA02 8th DATAH
	…
F930h	TKA08 1st DATAL
F931h	TKA08 1st DATAH
F932h	TKA08 2nd DATAL
F933h	TKA08 2nd DATAH
F934h	TKA08 3rd DATAL
F935h	TKA08 3rd DATAH
F936h	TKA08 4th DATAL
F937h	TKA08 4th DATAH
F938h	TKA08 5th DATAL
F939h	TKA08 5th DATAH
F93Ah	TKA08 6th DATAL
F93Bh	TKA08 6th DATAH
F93Ch	TKA08 7th DATAL
F93Dh	TKA08 7th DATAH
F93Eh	TKA08 8th DATAL
F93Fh	TKA08 8th DATAH

The TK scan result is 14-bit data, which are DATAH 6-bit and DATAL 8-bit. DATAH/L must be read in order to get the correct 14-bit data: first read the low byte (DATAL), then read the high word byte (DATAH)

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

95h.5 **TKIFA:** Touch Key A Interrupt Flag

Set by H/W at the end of Touch Key A conversion if F_{SYSCLK} is fast enough. S/W writes DFh to INTFLG or sets the TKSOCA bit to clear this flag.

SFR **9Ch**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON3	TKPDB	TKEOCB	TKIFB	TKXCAPB	JMPVALB			SPREAD
R/W	R/W	R	R/W	R/W	R/W			R/W
Reset	1	1	0	0	0	0	0	0

9Ch.7 **TKPDB:** Touch Key B power down

0: Touch Key B enable

1: Touch Key B disable

9C h.6 **TKEOCB:** Touch Key end of conversion flag, TKEOCB may have 3uS delay after TKSOCB=1, so F/W must wait enough time before polling this Flag.

0: Indicates conversion is in progress

1: Indicates conversion is finished

9C h.5 **TKIFB:** Touch Key B Interrupt Flag

Set by H/W at the end of Touch Key B conversion if SYSCLK is fast enough. S/W clear TKIFB or sets the TKSOCB bit to clear this flag.

9C h.4 **TKXCAPB:** Touch Key B external capacitor select

0: Keep 0, disable Touch Key B external capacitor

1: reserved (Do not set to 1)

9C h.3~1 **JMPVALB :** Touch Key Clock frequency fine tune , only available in TKFJMP=0

000=frequency slowest, 111=frequency fastest

9C h.3~0 **SPREAD:** TK spread spectrum

0: Disable

1: Enable

SFR **ADh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON	TKPDA	TKEOCA	TKRERUN	TKIVCS	TKXCAPA	TKOFFSET	ATKMODE	
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Reset	1	1	0	0	0	0	0	0

ADh.7 **TKPDA:** Touch Key A power down
0: Touch Key A enable
1: Touch Key A disable

ADh.6 **TKEOCA:** Touch Key end of conversion flag, TKEOCA may have 3uS delay after TKSOCA=1, so F/W must wait enough time before polling this Flag.
0: Indicates conversion is in progress
1: Indicates conversion is finished

ADh.5 **TKRERUN:** TK A/B Auto re-start, doesn’t need to set TKSOCA/B again to restart TK A/B converter.
0: Auto re-start disable. TKSOCA/B needs to be executed once for each TK A/B conversion
1: Auto re-start enable. After TKSOCA/B is executed once, TK A/B will be converted continuously without re-executing TKSOCA/B

ADh.4 **TKIVCS:** Touch Key internal voltage control select
0: VCHG=2.8V; VINT=1.4V
1: VCHG=3.6V; VINT=1.8V

ADh.3 **TKXCAPA:** Touch Key A external capacitor select
0: Keep 0, disable Touch Key A external capacitor
1: reserved (Do not set to 1)

ADh.2 **TKOFFSET:** status of non-scan TK
0: connect to VSS
1: connect to AC shielding , connect to VSS@EOC

ADh.1~0 **ATKMODE:** Touch Key Scan Mode
00: TKA and TKB scan method, each channel scan 1 time, max 22 TK channels
01: TKA and TKB scan method, each channel scan 2 times, max 16 TK channels
10: TKA and TKB scan method, each channel scan 4 times, max 8 TK channels
11: TKA and TKB scan method, each channel scan 8 times, max 4 TK channels

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

SFR **B4h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKTMRL	TKTMRL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

B4h.7~0 **TKTMRL:** Touch Key A/B Scan length bit 7~0 adjustment.
00: shortest, FF: longest

SFR **B5h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON2	TKFJMP	JMPVALA			TKTMRH			
R/W	R/W	R/W			R/W			
Reset	0	0	0	0	0	0	0	0

B5h.7 **TKFJMP:** Internal Touch Key clock frequency auto adjust option
0: Disable
1: Enable

B5h.6~5 **JMPVALA :** Touch Key A Clock frequency fine tune , only available in TKFJMP=0
000=frequency slowest, 111=frequency fastest

B5h.3~0 **TKTMRH:** Touch Key A/B Scan length 11~8 adjustment.
0000: shortest, 1111: longest

SFR **F8h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Reset	0	0	0	0	–	0	0	0

F8h.5 **TKSOCA:** Touch Key A Start of Conversion
Set 1 to start Touch Key A conversion. If SYSCLK is fast enough, this bit will be cleared by H/W at the end of conversion while TKRERUN=0. S/W can also write 0 to clear this flag.

F8h.2 **TKSOCB:** Touch Key B Start of Conversion
Set 1 to start Touch Key B conversion. If SYSCLK is fast enough, this bit will be cleared by H/W at the end of conversion while TKRERUN=0. S/W can also write 0 to clear this flag.

SFR **A7h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCHSA	–	–	–	TKCHSA				
R/W	–	–	–	R/W				
Reset	–	–	–	1	1	1	1	1

A7h.4~0 **TKCHSA:** Specify the first Touch Key A scan channel
00000: TKA00
00001: TKA01
00010: TKA02
00011: TKA03
00100: TKA04
00101: TKA05
00110: TKA06
00111: TKA07
01000: TKA08
01001: TKA09
01010: TKA10
01011: TKA11
01100: TKA12
01101: TKA13
01110: TKA14
01111: TKA15
10000: TKA16
10001: TKA17
10010: TKA18
10011: TKA19
10100: TKA20
10111: TKACAP internal reference capacitor channel

SFR **ACh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCHSB	–	–	–	TKCHSB				
R/W	–	–	–	R/W				
Reset	–	–	–	1	1	1	1	1

Ach.4~0 **TKCHSB:** Specify the first Touch Key B scan channel
00000: TKB00
00001: TKB01
00010: TKB02
00011: TKB03
00100: TKB04
00101: TKB05
00110: TKB06
00111: TKB07
01000: TKB08
01001: TKB09
01010: TKB10
01011: TKB11
01100: TKB12
01101: TKB13
01110: TKB14
01111: TKB15
10000: TKB16
10001: TKB17
10010: TKB18
10011: TKB19
10100: TKB20
10111: TKBCAP internal reference capacitor channel

SFR **C1h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSA0	**TKPINSA0**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C1h.7 TKA07 Pin fix as TK channel: 0: disable 1: enable
C1h.6 TKA06 Pin fix as TK channel: 0: disable 1: enable
C1h.5 TKA05 Pin fix as TK channel: 0: disable 1: enable
C1h.4 TKA04 Pin fix as TK channel: 0: disable 1: enable
C1h.3 TKA03 Pin fix as TK channel: 0: disable 1: enable
C1h.2 TKA02 Pin fix as TK channel: 0: disable 1: enable
C1h.1 TKA01 Pin fix as TK channel: 0: disable 1: enable
C1h.0 TKA00 Pin fix as TK channel: 0: disable 1: enable

SFR **C2h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSA1	**TKPINSA1**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C2h.7 TKA15 Pin fix as TK channel: 0: disable 1: enable
C2h.6 TKA14 Pin fix as TK channel: 0: disable 1: enable
C2h.5 TKA13 Pin fix as TK channel: 0: disable 1: enable
C2h.4 TKA12 Pin fix as TK channel: 0: disable 1: enable
C2h.3 TKA11 Pin fix as TK channel: 0: disable 1: enable
C2h.2 TKA10 Pin fix as TK channel: 0: disable 1: enable
C2h.1 TKA09 Pin fix as TK channel: 0: disable 1: enable
C2h.0 TKA08 Pin fix as TK channel: 0: disable 1: enable

SFR **C3h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSA2	**TKPINSA2**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C3h.7~5 Reservd
C3h.4 TKA20 Pin fix as TK channel: 0: disable 1: enable
C3h.3 TKA19 Pin fix as TK channel: 0: disable 1: enable
C3h.2 TKA18 Pin fix as TK channel: 0: disable 1: enable
C3h.1 TKA17 Pin fix as TK channel: 0: disable 1: enable
C3h.0 TKA16 Pin fix as TK channel: 0: disable 1: enable

SFR **C4h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSB0	**TKPINSB0**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C4h.7 TKB07 Pin fix as TK channel: 0: disable 1: enable
C4h.6 TKB06 Pin fix as TK channel: 0: disable 1: enable
C4h.5 TKB05 Pin fix as TK channel: 0: disable 1: enable
C4h.4 TKB04 Pin fix as TK channel: 0: disable 1: enable
C4h.3 TKB03 Pin fix as TK channel: 0: disable 1: enable
C4h.2 TKB02 Pin fix as TK channel: 0: disable 1: enable
C4h.1 TKB01 Pin fix as TK channel: 0: disable 1: enable
C4h.0 TKB00 Pin fix as TK channel: 0: disable 1: enable

SFR **D7h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSB1	**TKPINSB1**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D7h.7 TKB15 Pin fix as TK channel: 0: disable 1: enable
D7h.6 TKB14 Pin fix as TK channel: 0: disable 1: enable
D7h.5 TKB13 Pin fix as TK channel: 0: disable 1: enable
D7h.4 TKB12 Pin fix as TK channel: 0: disable 1: enable
D7h.3 TKB11 Pin fix as TK channel: 0: disable 1: enable
D7h.2 TKB10 Pin fix as TK channel: 0: disable 1: enable
D7h.1 TKB09 Pin fix as TK channel: 0: disable 1: enable
D7h.0 TKB08 Pin fix as TK channel: 0: disable 1: enable

SFR **DFh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSB2	**TKPINSB2**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

DFh.7~5 Reservd
DFh.4 TKB20 Pin fix as TK channel: 0: disable 1: enable
DFh.3 TKB19 Pin fix as TK channel: 0: disable 1: enable
DFh.2 TKB18 Pin fix as TK channel: 0: disable 1: enable
DFh.1 TKB17 Pin fix as TK channel: 0: disable 1: enable
DFh.0 TKB16 Pin fix as TK channel: 0: disable 1: enable

SFR **C5h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCHA0	**ATKCHA0**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C5h.7 TKA07 scan enable: 0: disable 1: enable
C5h.6 TKA06 scan enable: 0: disable 1: enable
C5h.5 TKA05 scan enable: 0: disable 1: enable
C5h.4 TKA04 scan enable: 0: disable 1: enable
C5h.3 TKA03 scan enable: 0: disable 1: enable
C5h.2 TKA02 scan enable: 0: disable 1: enable
C5h.1 TKA01 scan enable: 0: disable 1: enable
C5h.0 TKA00 scan enable: 0: disable 1: enable

SFR **C6h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCHA1	**ATKCHA1**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C6h.7 TKA15 scan enable: 0: disable 1: enable
C6h.6 TKA14 scan enable: 0: disable 1: enable
C6h.5 TKA13 scan enable: 0: disable 1: enable
C6h.4 TKA12 scan enable: 0: disable 1: enable
C6h.3 TKA11 scan enable: 0: disable 1: enable
C6h.2 TKA10 scan enable: 0: disable 1: enable
C6h.1 TKA09 scan enable: 0: disable 1: enable
C6h.0 TKA08 scan enable: 0: disable 1: enable

SFR **C7h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCHA2	**ATKCHA2**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C7h.7 TKACAP (TKA23) internal reference capacitor channel scan enable: 0: disable 1: enable
C7h.6~5 Reservd
C7h.4 TKA20 scan enable: 0: disable 1: enable
C7h.3 TKA19 scan enable: 0: disable 1: enable
C7h.2 TKA18 scan enable: 0: disable 1: enable
C7h.1 TKA17 scan enable: 0: disable 1: enable
C7h.0 TKA16 scan enable: 0: disable 1: enable

SFR **B7h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCHB0	**ATKCHB0**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

B7h.7 TKB07 scan enable: 0: disable 1: enable
B7h.6 TKB06 scan enable: 0: disable 1: enable
B7h.5 TKB05 scan enable: 0: disable 1: enable
B7h.4 TKB04 scan enable: 0: disable 1: enable
B7h.3 TKB03 scan enable: 0: disable 1: enable
B7h.2 TKB02 scan enable: 0: disable 1: enable
B7h.1 TKB01 scan enable: 0: disable 1: enable
B7h.0 TKB00 scan enable: 0: disable 1: enable

SFR **B6h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCHB1	**ATKCHB1**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

B6h.7 TKB15 scan enable: 0: disable 1: enable
B6h.6 TKB14 scan enable: 0: disable 1: enable
B6h.5 TKB13 scan enable: 0: disable 1: enable
B6h.4 TKB12 scan enable: 0: disable 1: enable
B6h.3 TKB11 scan enable: 0: disable 1: enable
B6h.2 TKB10 scan enable: 0: disable 1: enable
B6h.1 TKB09 scan enable: 0: disable 1: enable
B6h.0 TKB08 scan enable: 0: disable 1: enable

SFR **AFh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCHB2	**ATKCHB2**							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

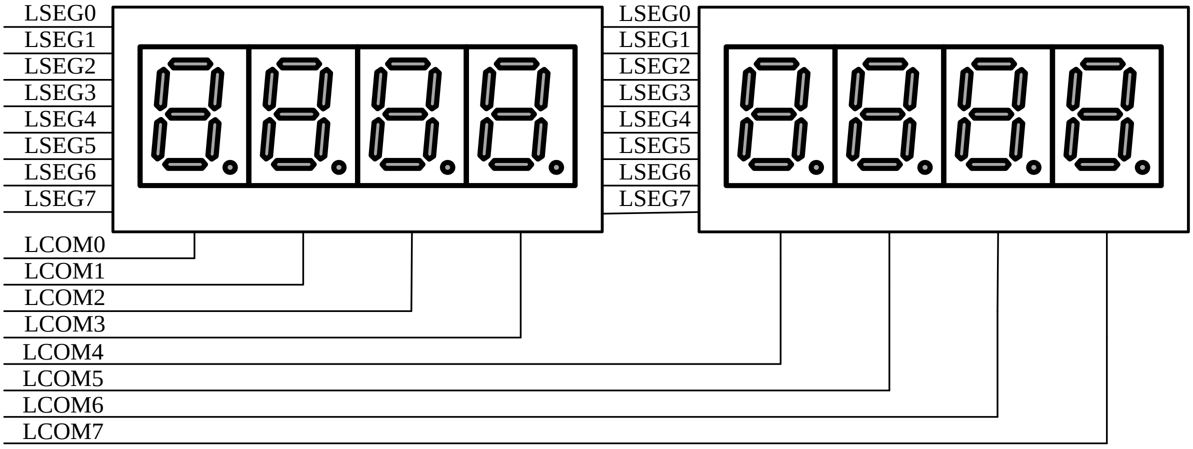
Afh.7 TKBCAP (TKB23) internal reference capacitor channel scan enable: 0: disable 1: enable
Afh.6~5 Reservd
Afh.4 TKB20 scan enable: 0: disable 1: enable
Afh.3 TKB19 scan enable: 0: disable 1: enable
Afh.2 TKB18 scan enable: 0: disable 1: enable
Afh.1 TKB17 scan enable: 0: disable 1: enable
Afh.0 TKB16 scan enable: 0: disable 1: enable

13. LCD/LED Controller/Driver

The module can be configured with three drive modes: LED matrix mode, LED dot matrix mode, LCD drive mode. By register configuration, it only supports one mode of operation at the same time.

13.1 LED Matrix (MX) Mode

The Chip supports an LED controller and driver at matrix mode. If LEDMODE=00b, LXDON=SELLED=1. The LED matrix mode will enable. It provides 8 Segment pins and 8 Common pins to drive an LED module with 64 pixels. The COM pins have a high sink current. The brightness of the LED can be set by LXDBRIT. When it is set to 1111b, it is the highest brightness. In addition, LEDBRITM is used to set the brightness and uniformity bit. When LEDBRITM=0, better display uniformity can be obtained. When LEDBRITM= 1, better display brightness can be obtained.



The display configuration in XRAM corresponds to the lighting status of the corresponding address (1 means lighting, 0 means not lighting).

XRAM Addr.	COM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FB00h	LCOM0	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB01h	LCOM1	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB02h	LCOM2	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB03h	LCOM3	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB04h	LCOM4	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB05h	LCOM5	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB06h	LCOM6	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB07h	LCOM7	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0

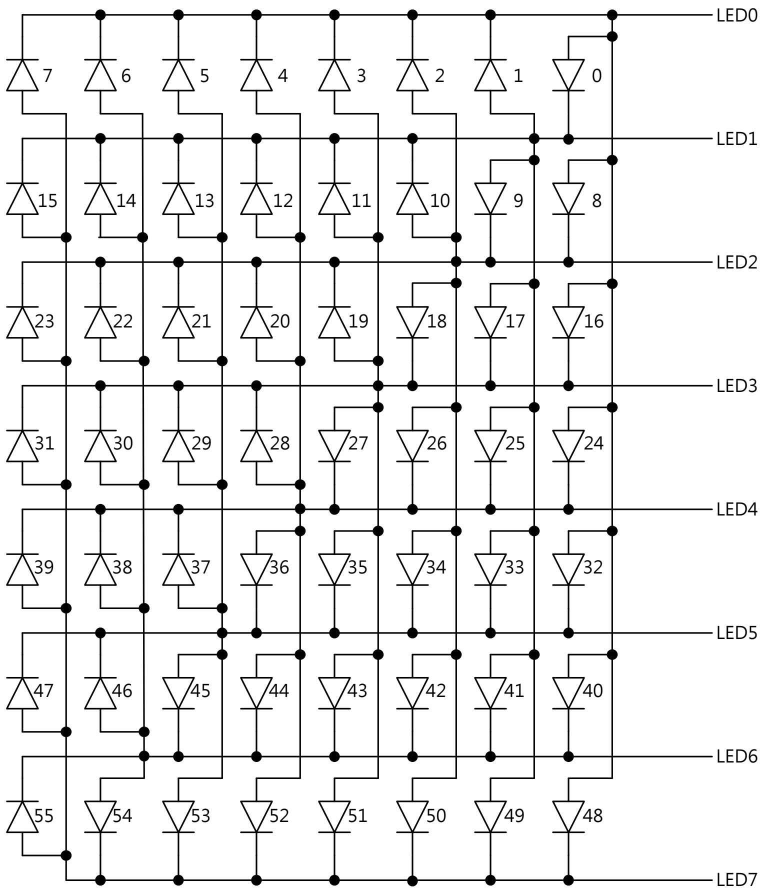
LED matrix drive mode corresponding display configuration table

13.2 LED Dot Matrix (DMX) Mode

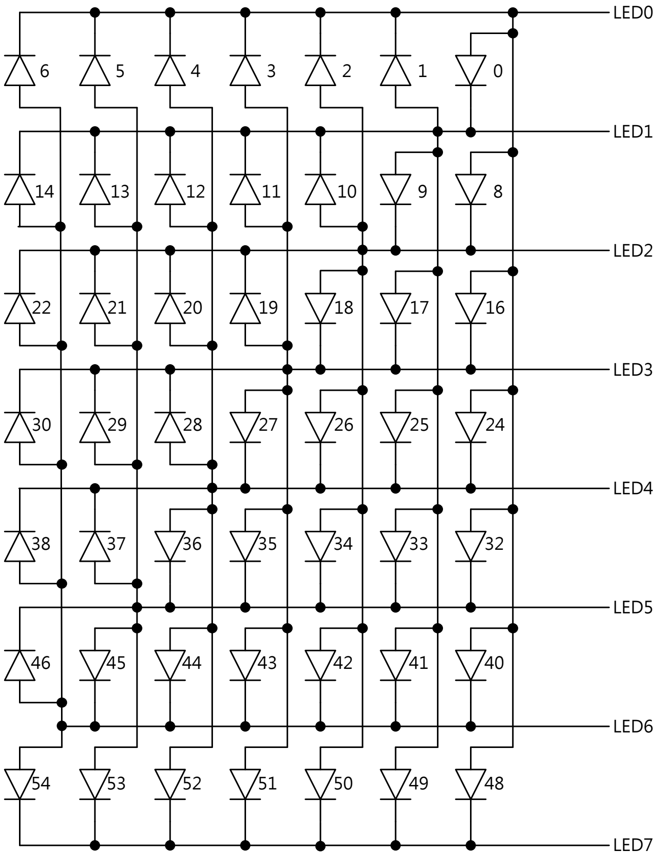
If LEDMODE=10b, LXDON=SELLED=1. The LED dot matrix mode will enable. The LED dot matrix is a universal 7*8 dot matrix. Corresponding to LED0~LED7 ports, up to 7x8=56 LED dots can be configured to drive, the corresponding position of the LED is marked in the 7*8 dot matrix in the figure below Address, the display configuration in XRAM corresponds to the lighting status of the corresponding address (1 means lighting, 0 means not lighting). Support up to 56 lights LED drive. Using LXDDUTY to choose dot matrix 4*4, 5*5, 6*6, 6*7, 7*7 and 7*8, the corresponding LED address remains unchanged. The brightness of the LED can be set by LXDBRIT. When it is set to 1111b, it is the highest brightness. In addition, LEDBRITM is used to set the brightness and uniformity bit. When LEDBRITM=0, better display uniformity can be obtained. When LEDBRITM= 1, better display brightness can be obtained.

XRAM Addr.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FB00h	7	6	5	4	3	2	1	0
FB01h	15	14	13	12	11	10	9	8
FB02h	23	22	21	20	19	18	17	16
FB03h	31	30	29	28	27	26	25	24
FB04h	39	38	37	36	35	34	33	32
FB05h	47	46	45	44	43	42	41	40
FB06h	55	54	53	52	51	50	49	48

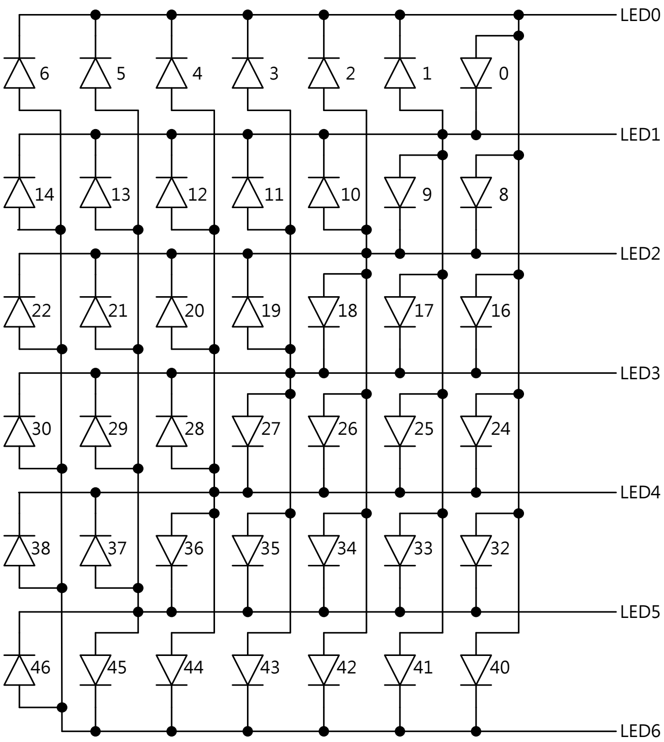
LED dot matrix drive mode corresponding display configuration table



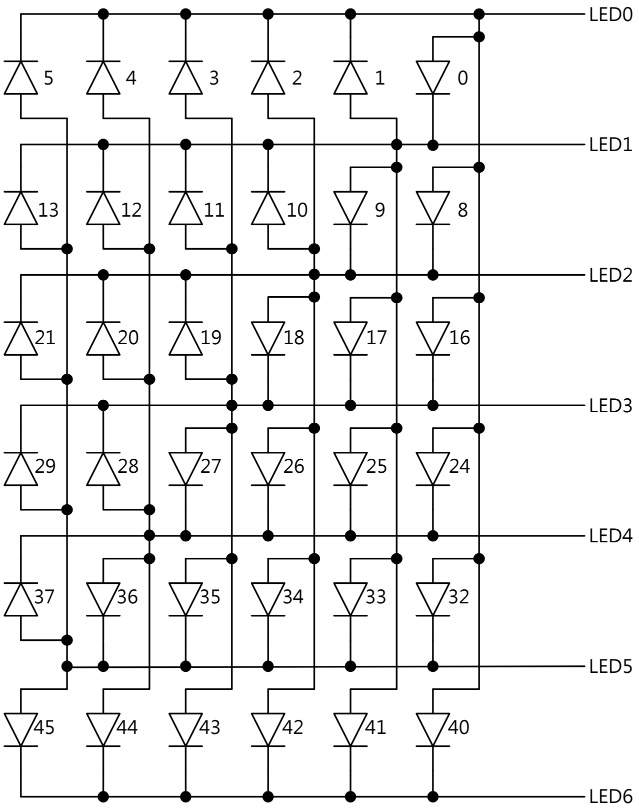
LED 7*8 dot matrix



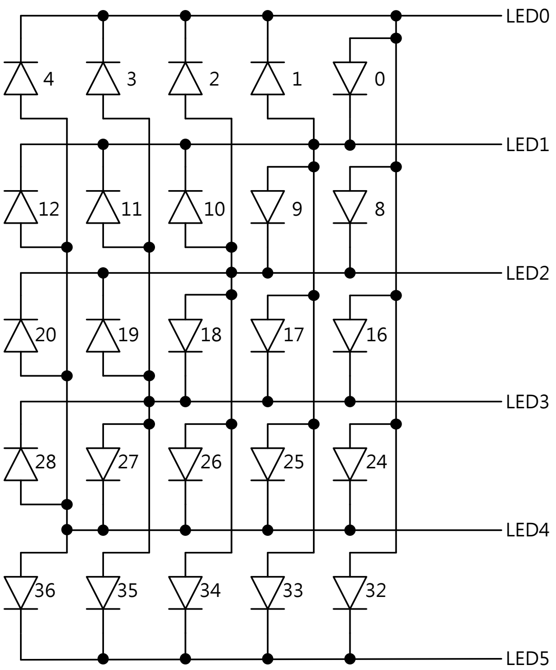
LED 7*7 dot matrix



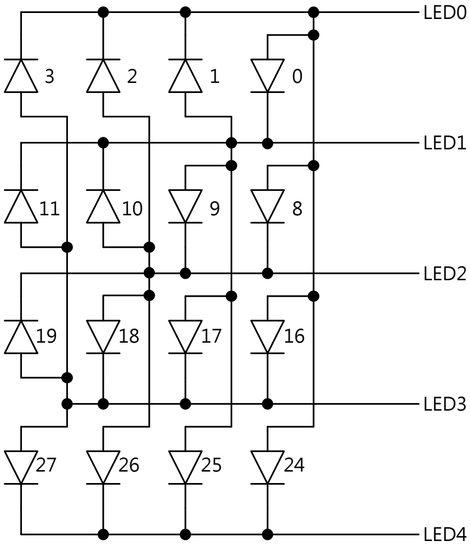
LED 6*7 dot matrix



LED 6*6 dot matrix



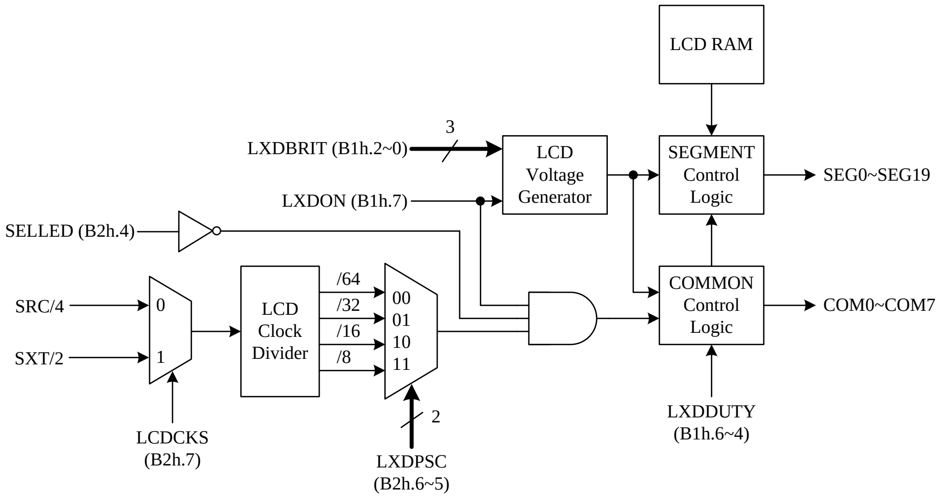
LED 5*5 dot matrix



LED 4*4 dot matrix

13.3 LCD Mode

The Chip supports an LCD controller and driver. The LCD driver is capable of driving the LCD panel with 80 dots by 4 Commons and 20 Segments or 128 dots by 8 Commons and 16 Segments. It is capable of driving 1/3 bias. The LCD clock source is generated from SRC/4 or SXT/2 depends on SFR LCDCKS. The clock rate can be divided by 8, 16, 32, and 64 by the LXDPSC bits. If SRC/4 is the LCD clock source, the V_{CC} voltage level would affect the SRC frequency and LCD frame rate. The LCDRAM is located in the 8051's External Data Memory space, addressing from FB00h to FB1Fh.

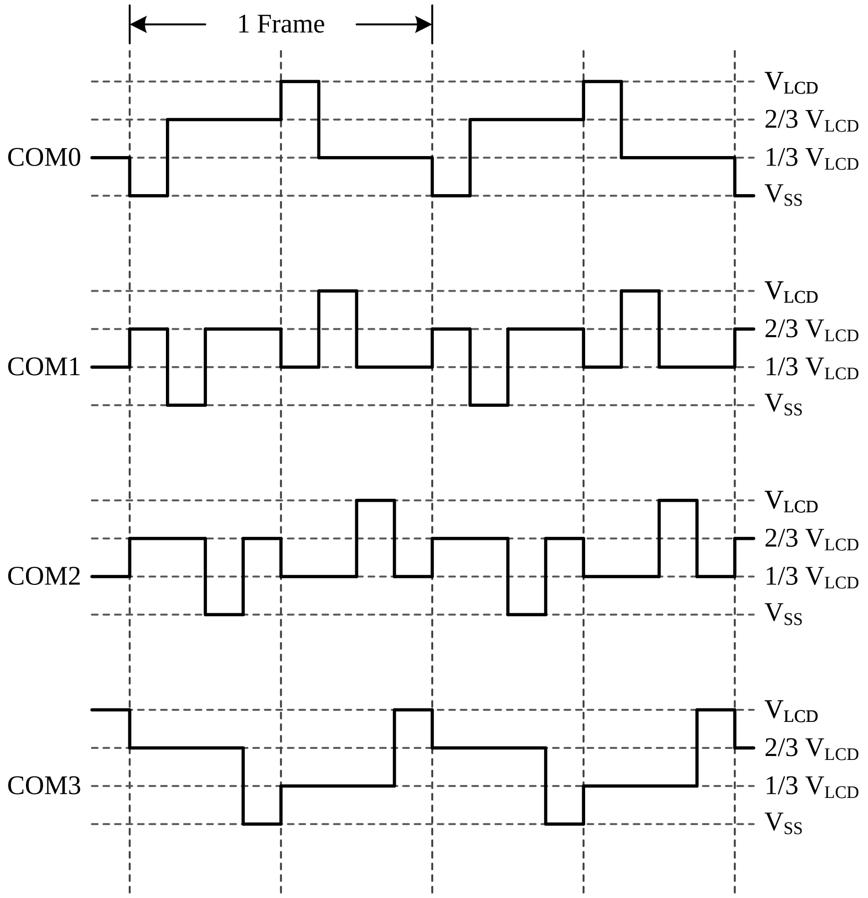


LCD COM0~7 Circuit

Addr.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	COM
FB00h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM0
FB01h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM1
FB02h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM2
FB03h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM3
FB04h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM4
FB05h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM5
FB06h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM6
FB07h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM7
FB08h	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM0
FB09h	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM1
FB0Ah	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM2
FB0Bh	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM3
FB0Ch	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM4
FB0Dh	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM5
FB0Eh	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM6
FB0Fh	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM7
FB10h					SEG19	SEG18	SEG17	SEG16	COM0
FB11h					SEG19	SEG18	SEG17	SEG16	COM1
FB12h					SEG19	SEG18	SEG17	SEG16	COM2
FB13h					SEG19	SEG18	SEG17	SEG16	COM3
FB14h					SEG19	SEG18	SEG17		COM4
FB15h					SEG19	SEG18			COM5
FB16h					SEG19				COM6
FB17h									COM7

LCD corresponding display configuration table

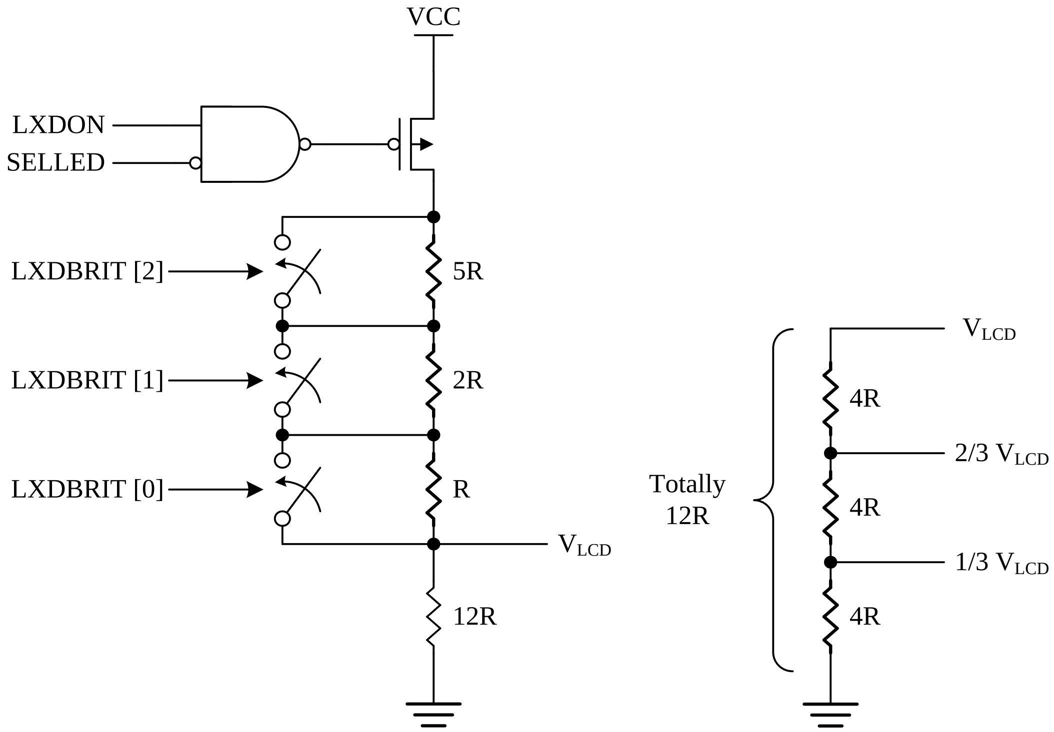
The frequency of any repeating waveform output on the COM pin can be used to represent the LCD frame rate. The figure below shows an LCD frame.



The frame rate table for each lighting system is shown below.

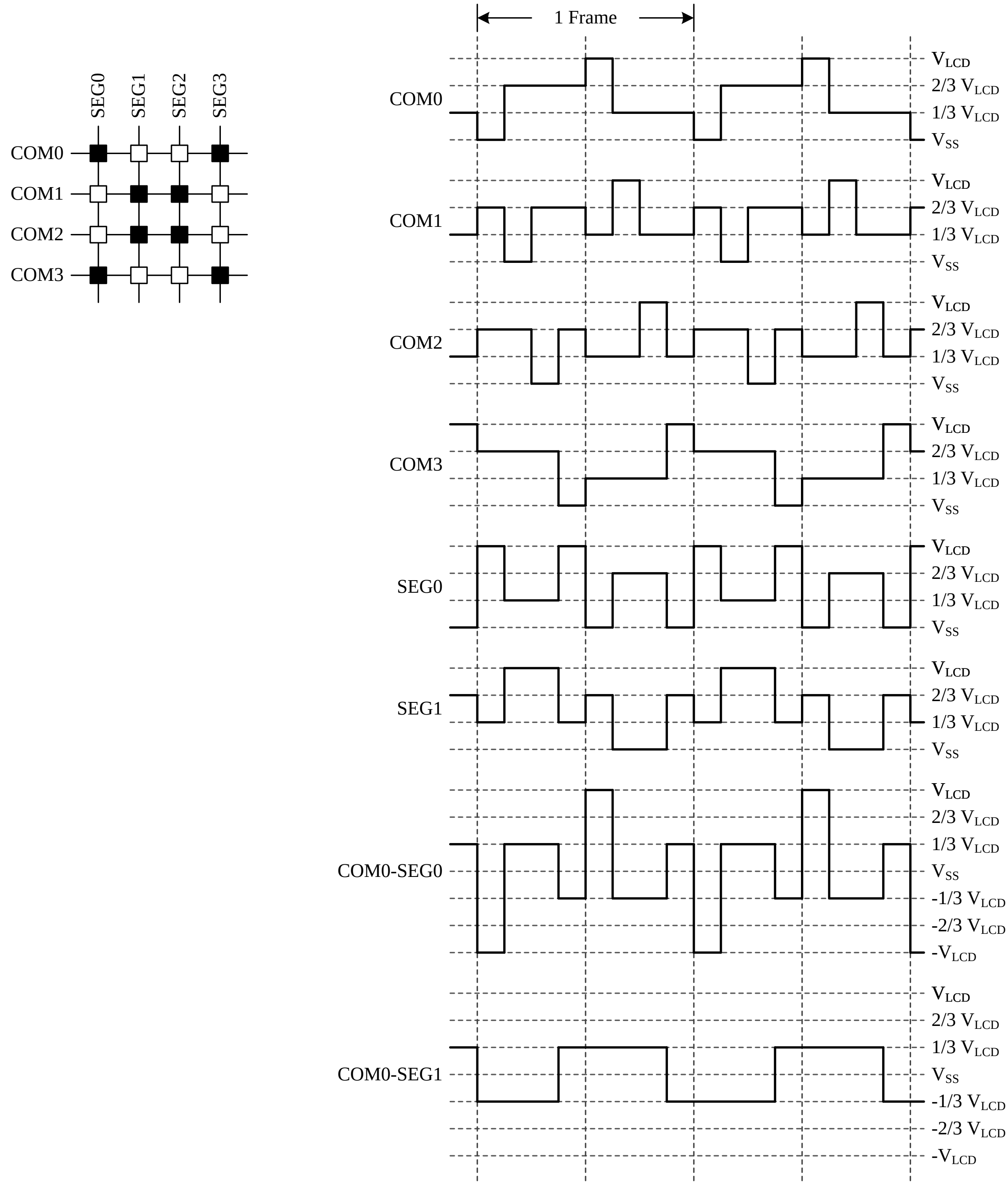
LCDCKS (B2h.7)	LXDPSC (B2h.6~5)	LCD Frame Rate (Hz)			
		LXDDUTY (B1h.6~4)			
		1/4 duty	1/5 duty	1/6 duty	1/8 duty
SXT/2 16384 Hz	/8	256	205	171	128
	/16	128	102	85	64
	/32	64	51	43	32
	/64	32	26	21	16
SRC/4 20000 Hz	/8	313	250	208	156
	/16	156	125	104	78
	/32	78	62	52	39
	/64	39	31	26	20

The following figure of the LCD voltage generator shows the internal voltage generator composed by resistors. LXDON and SELLED control the current flows from V_{CC} to ground. If LXDON=0 or SELLED=1, the PMOS will turn off the path so that all LCD voltages will be 0 V. If LXDON=1 and SELLED=0, the resistor divider will work to generate multi voltages to provide the LCD control module for generating the desired waveforms. The LXDBRIT control bits will open/short the switches to determine V_{LCD}. The table below shows V_{LCD} corresponding to LXDBRIT. The voltage divider circuit will consume current because the DC path is always on when LXDON=1 and SELLED=0.



LXDBRIT	V_{LCD}
000	(12/20) x V_{CC}
001	(12/19) x V_{CC}
010	(12/18) x V_{CC}
011	(12/17) x V_{CC}
100	(12/15) x V_{CC}
101	(12/14) x V_{CCc}
110	(12/13) x V_{CC}
111	V_{CC}

1/4 Duty, 1/3 Bias Output Waveform



SFR **B1h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LXDCON	LXDON	LXDDUTY			LEDBRITM	LXDBRIT		
R/W	R/W	R/W			R/W	R/W		
Reset	0	0	0	0	0	1	1	1

B1h.7 **LXDON:** LCD/LED enable
0: LCD/LED disable
1: LCD/LED enable

B1h.6~4 **LXDDUTY:** LCD/LED duty select
LCD select (if SELLED=0):
000: 1/4 Duty, COM 0~3
001: 1/4 Duty, COM 0~3
010: 1/5 Duty, COM 0~4
011: 1/6 Duty, COM 0~5
100: 1/6 Duty, COM 0~5
101: 1/8 Duty, COM 0~7
110: 1/8 Duty, COM 0~7
111: 1/8 Duty, COM 0~7

LED select: Matrix mode (if SELLED=1, LEDMODE=00b)
000: 1/2 Duty, LCOM 0~1
001: 1/3 Duty, LCOM 0~2
010: 1/4 Duty, LCOM 0~3
011: 1/5 Duty, LCOM 0~4
100: 1/6 Duty, LCOM 0~5
101: 1/7 Duty, LCOM 0~6
110: 1/8 Duty, LCOM 0~7
111: 1/8 Duty, LCOM 0~7

LED select: Dot Matrix mode (if SELLED=1, LEDMODE=10b)
000: 4x4, LED 0~4
001: 5x5, LED 0~5
010: 6x6, LED 0~6
011: 6x7, LED 0~6
100: 7x7, LED 0~7
101: 7x8, LED 0~7
110: Reserved
111: Reserved

B1h.3 **LEDBRITM:** LED Brightness Mode
0: Uniform brightness mode
1: Brightness enhancement mode

B1h.2~0 **LXDBRIT:** LCD/LED Brightness control
000: Level 0 (Darkest)
…
111: Level 7 (Brightest)

SFR **B2h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LXDCON2	LCDCKS	LXDPSC		SELLED	LEDHOLD	–	LEDMODE	
R/W	R/W	R/W		R/W	R/W	–	R/W	
Reset	0	0	0	0	0	–	0	0

B2h.7 **LCDCKS:** LCD clock source select (note LED Clock Source fixed as FRC)
0: SRC/4
1: SXT/2

B2h.6~5 **LXDPSC:** LCD/LED clock prescaler select
00: LCD/LED clock is FRC divided by 64
01: LCD/LED clock is FRC divided by 32
10: LCD/LED clock is FRC divided by 16
11: LCD/LED clock is FRC divided by 8

B2h.4 **SELLED:** LCD/LED function select
0: LCD
1: LED

B2h.3 **LEDHOLD:** Keep at 0, cannot be set to 1

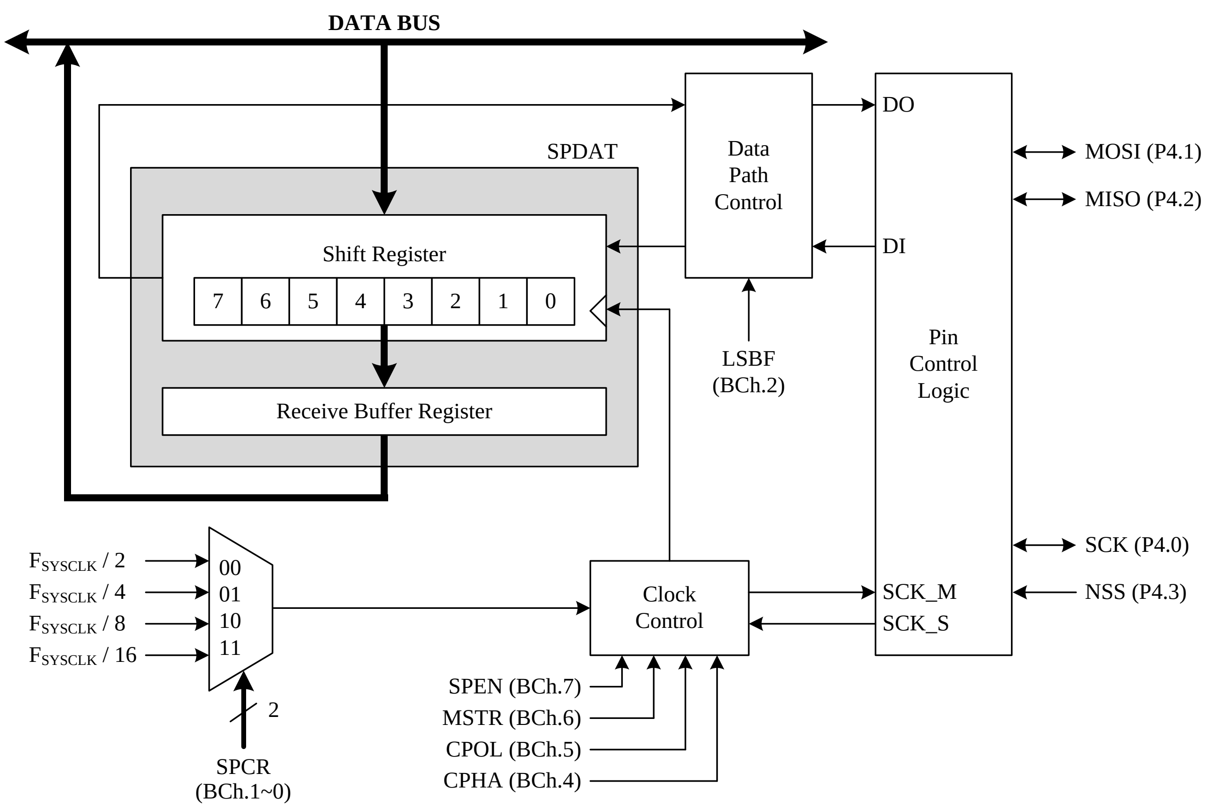
B2h.1~0 **LEDMODE:** LED Mode select
00: Matrix scan mode
01: Reserved
10: Dot Matrix scan mode
11: Reserved

14. 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 or 4-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	PINMODx	P4.n SFR data	Pin State
Master Mode MISO	**xx01**	1	SPI Data Input
Master Mode SCK, MOSI	**xx10**	X	SPI Clock/Data Output (CMOS Push-Pull)
Slave Mode MISO	**xx10**	X	SPI Data Output (CMOS Push-Pull)
Slave Mode SCK, MOSI	**xx01**	1	SPI Clock/Data Input
SS	**xx01**	1	SPI Chip Selection

Pin Mode Setting for SPI

The four signals used by SPI are described below. The MOSI 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 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 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 eight programmable clock rates in Master mode. The SS signal is a low active slave select pin. In 4-wire Slave mode, the signal is ignored when the Slave is not selected (SS=1). The SS is ignored when the SSDIS in SPCON is set in both Master and Slave modes. In Slave mode and the SSDIS is clear, the SPI active when SS stay low. For multiple-slave mode, only one slave device is selected at a time to avoid bus collision on the MISO line. In Master mode and the SSDIS is cleared, the MODF in SPSTA is set when this signal is low. For multiple-master mode, enable SS line to avoid multiple driving on MOSI and SCK lines from multiple masters.

Master Mode

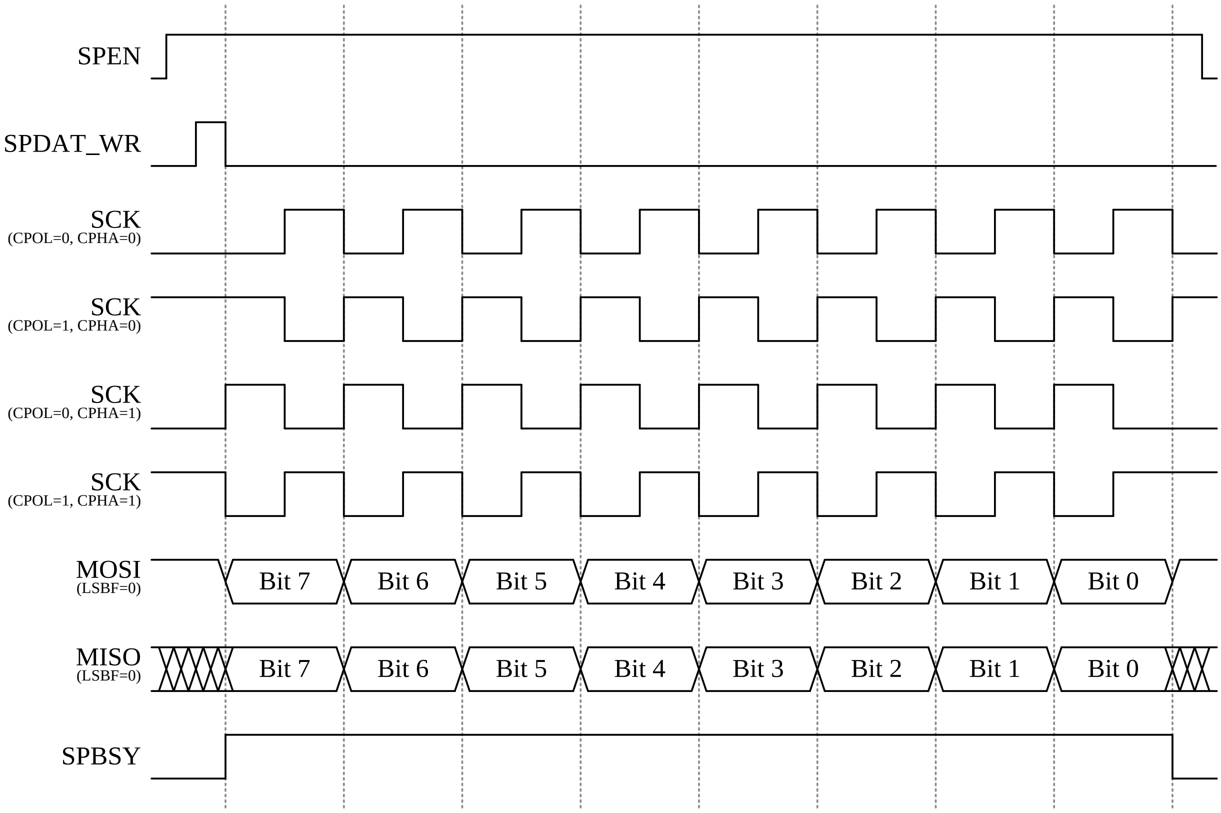
The SPI operates in Master mode by setting the MSTR bit in the SPCON. To start transmit, writing a data to the SPDAT. If the SPBSY bit is cleared, the data will be transferred to the shift register and starts shift out on the MOSI line. The data of the slave shift in from the MISO line at the same time. When the SPIF bit in the SPSTA becomes set at the end of the 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

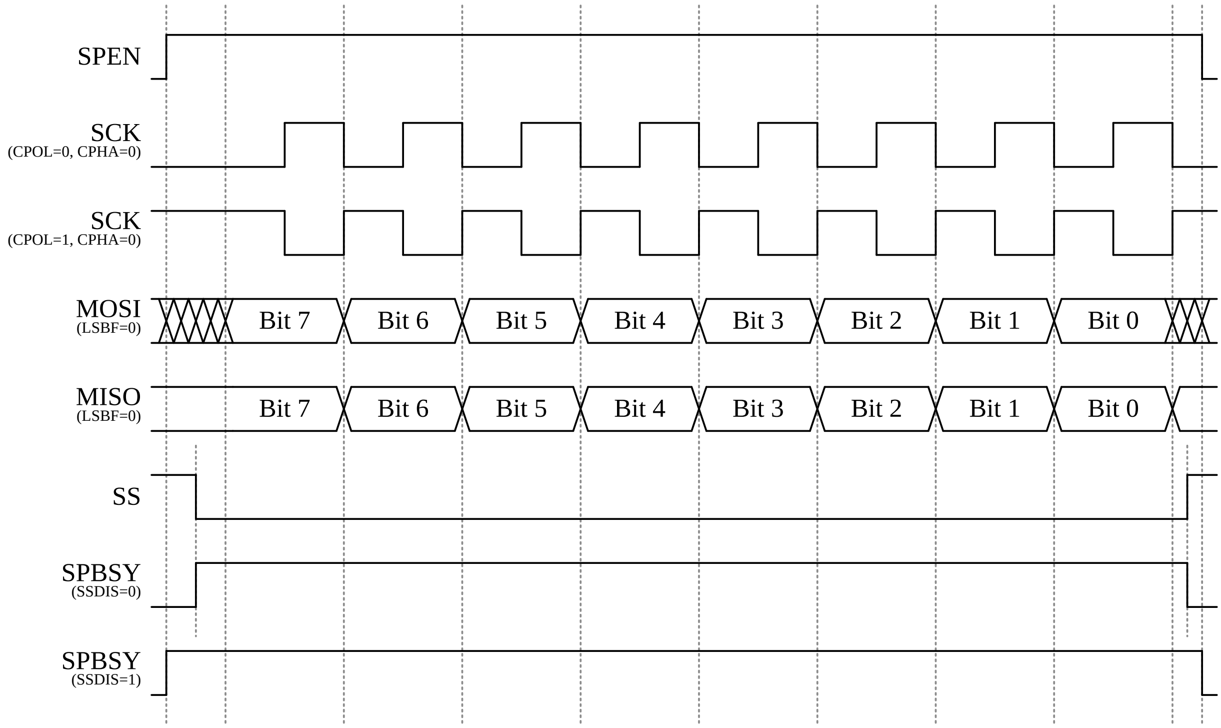
The SPI operates in Slave mode by clearing the MSTR bit in the SPCON. If the SSDIS is cleared, the transmission will start when the SS become low and remain low until the end of a data transfer. If the SSDIS is set, the transmission will start when the SPEN bit in the SPCON is set, and don't care the SS. 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 the RCVBF is cleared. If the RCVBF is set, the newer receive 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 $F_{SYSCLK}/4$. In Slave mode, the SPBSY bit refers to the SS pin when the SSDIS bit is cleared, and refer to the SPEN bit when SSDIS bit is set.

Serial Clock

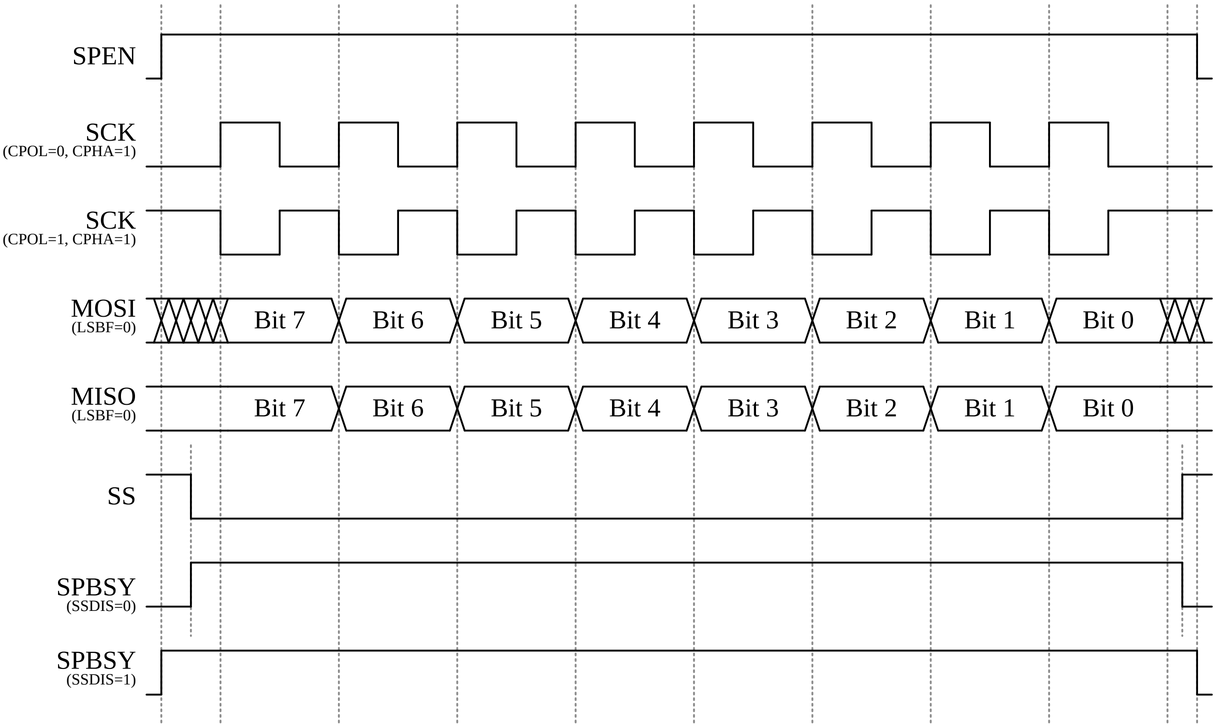
The SPI has four clock types by setting the CPOL and CPHA bits in the SPCON register. The CPOL bit defines the level of the SCK in SPI idle state. The level of the SCK in idle state is low when the CPOL bit is cleared, and is high when the CPOL bit is set. The CPHA bit defines the edges used to sample 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 the CPHA bit is cleared. The SPI sample data on the second edge of SCK period and shift data on first edge of SCK period when the CPHA bit is set. The 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 bit is set by H/W at the end of a data transfer and generates an interrupt if SPI interrupt is enabled. The SPIF bit is cleared automatically when the program performs the interrupt service routines. S/W can also write 0 to clear this flag. If write data to SPDAT when the SPBSY is set, the WCOL bit will be set by H/W and generates an interrupt if SPI interrupt is enabled. When this occurs, the data write to SPDAT will be ignored, and shift register will not be written. Write 0 to this bit or when SPBSY is cleared and rewrite data to SPDAT will clear this flag. The MODF bit is set when SSDIS is cleared and SS pin is pulled low in Master mode. If SPI interrupt is enabled, an interrupt will be generated. When this bit is set, the SPEN and MSTR in SPCON will be cleared by H/W. Write 0 to this bit will clear this flag.

SFR **BCh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPCON	SPEN	MSTR	CPOL	CPHA	SSDIS	LSBF	SPCR	
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

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.3 **SSDIS:** SS pin disable
0: Enable SS pin 1: Disable SS pin

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	MODF	RCVOVF	RCVBF	SPBSY	–	–
R/W	R/W	R/W	R/W	R/W	R/W	R	–	–
Reset	0	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.5 **MODF:** Mode fault interrupt flag
Set by H/W when SSDIS is cleared and SS pin is pulled low in Master mode. Write 0 to this bit will clear this flag. When this bit is set, the SPEN and MSTR in SPCON will be cleared by H/W.

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.

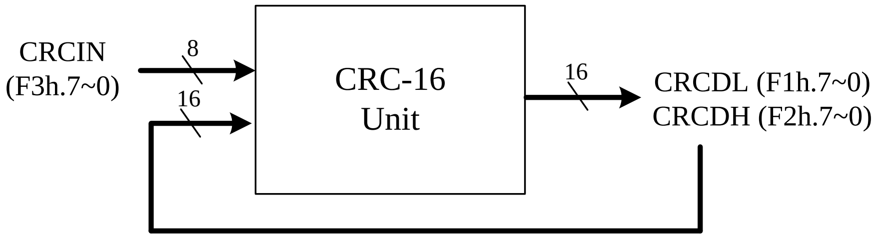
BDh.2 **SPBSY:** SPI busy flag
Set by H/W when a SPI transfer is in progress.

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.

15. Cyclic Redundancy Check (CRC)

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



CRC Block Diagram

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

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

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

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

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

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

SFR **F3h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCIN	CRCIN							
W	W							
Reset	–	–	–	–	–	–	–	–

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

16. Multiplier and divider

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

F7h.3 **DIV32:**
only active when MULDVI16 =1
0: instruction DIV as 16/16 bit division operation
1: instruction DIV as 32/16 bit division operation

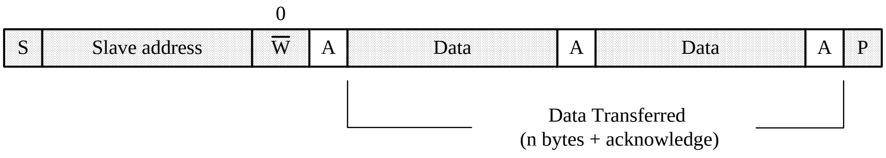
F7h.0 **MULDIV16:**
0: instruction MUL/DIV as 8*8, 8/8 operation
1: instruction MUL/DIV as 16*16, 16/16 or 32/16 operation

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

17. Master I²C Interface

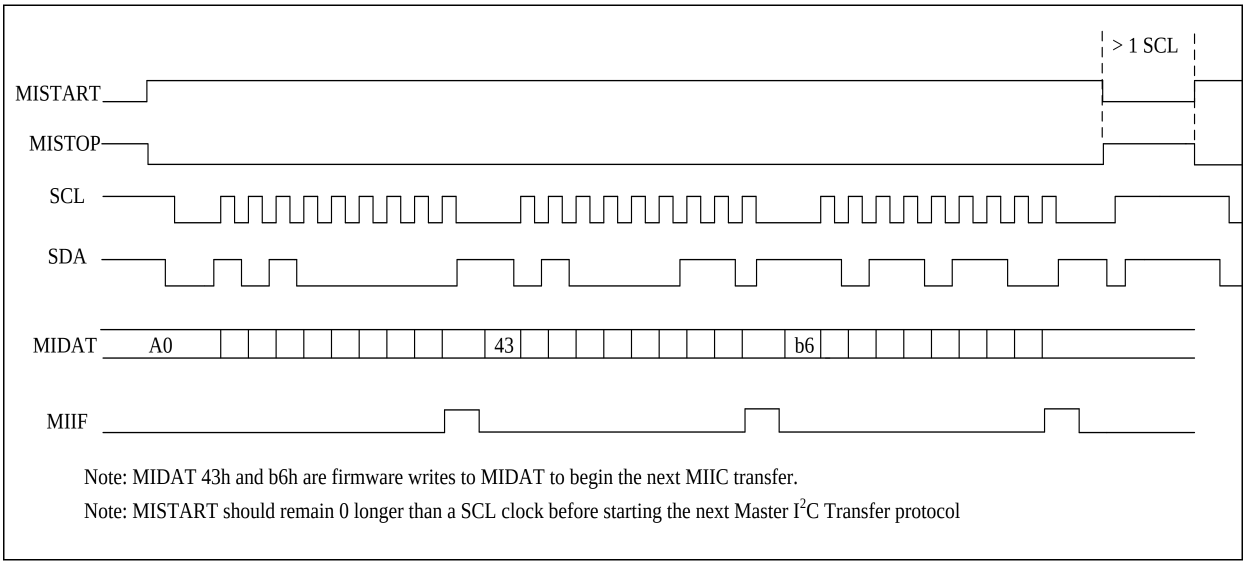
Master I²C interface transmit mode:
At the beginning write slave address and direction bit to MIDAT and set MISTART. After the START condition (MISTART), the 7 bits slave address and one bit direction bit are sent. When MIIF convert to 1, address and direction bit transmission was complete. After sending 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 I²C protocol. SCL clock can be adjusted via MICR.



A master-transmitter addressing a slave receiver with a 7-bit address.
The transfer direction is not changed

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

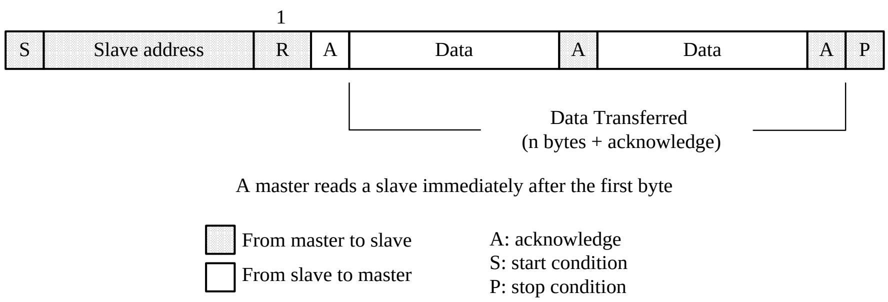


Master Transmit Timing

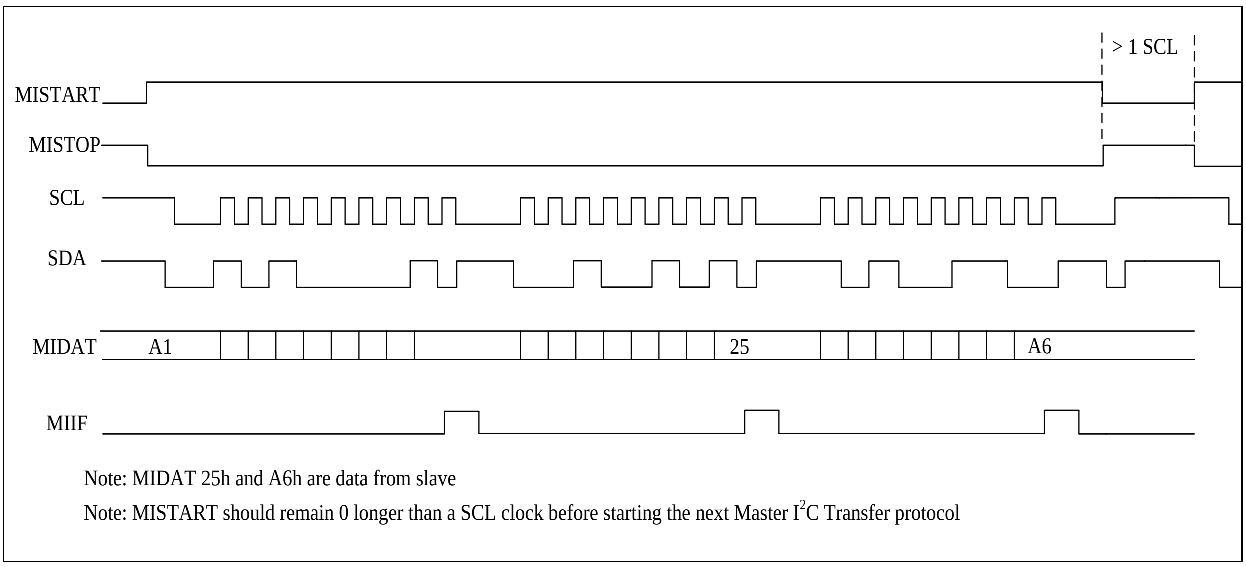
Note: MISTART should remain 0 longer than a SCL period before starting the next Master I²C protocol.

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.



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), Clear MIIF
(4) Read data from MIDAT to start first receive data
(receiving data has not been completed at this time, and the read MIDAT should be discarded)
(5) Wait until MIIF convert to 1, Clear MIIF
(6) Read slave data from MIDAT and Loop (5) ~(6) to receive next data
(7) Set MISTOP to stop the I²C transfer



Master Receive Timing

I²C Function Pin	PINMODx	Px.n SFR data	Pin State
I²C Master SCL	**0000**	X	Clock Output (Open Drain Output)
	0010	X	Clock Output (CMOS Push-Pull)
I²C Master/Slaver SDA	**0000**	1	DATA (Pull-up)

Pin Mode Setting for Master I²C

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

A9h.6 **I2CE:** I²C interrupt enable
0: Disable I²C interrupt
1: Enable I²C interrupt

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 I²C enable
0: disable 1: enable

E1h.6 **MIACKO**: When Master I²C receive data, send acknowledge to I²C Bus
0: ACK to slave device 1: NACK to slave device

E1h.5 **MIIF**: Master I²C Interrupt flag
When the master I²C sends or receives a byte, it is set by H/W. Writing "0" to this bit will clear the flag

E1h.4 **MIACKI**: When Master I²C transfer, acknowledgement form I²C bus (read only)
0: ACK received 1: NACK received

E1h.3 **MISTART**: Master I²C Start bit
1: start I²C bus transfer

E1h.2 **MISTOP**: Master I²C Stop bit
1: send STOP signal to stop I²C bus

E1h.1~0 **MICR:** Master I²C (SCL) clock frequency selection
00: Fsys/4 (ex. If Fsys=16MHz, I²C clock is 4M Hz)
01: Fsys/16 (ex. If Fsys=16MHz, I²C clock is 1M Hz)
10: Fsys/64 (ex. If Fsys=16MHz, I²C clock is 250K Hz)
11: Fsys/256 (ex. If Fsys=16MHz, I²C clock is 62.5K Hz)

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 I²C data shift register
(W):After Start and before Stop condition, write this register will resume transmission to I²C bus
I: After Start and before Stop condition, read this register will resume receiving from I²C bus

SFR **Eah**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SICON	MIIE	TXDIE	RCD2IE	RCD1IE	–	TXDF	RCD2F	RCD1F
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Reset	0	0	0	0	–	1	0	0

Eah.7 **MIIE:** I²C Master interrupt enable
0: disable 1: enable

18. Slave I²C Interface

The chip provides Slave I²C interface receive protocol as following. Slave I²C module only allow to receive one or two byte data each time after start condition. Before receiving DATA1, be aware that RCD1F must be 0. After DATA1 reception is completed, RCD1F will be converted to 1 and an interrupt will be issued according to the user's request. User can use firmware to clear RCD1F before receiving next DATA1 again. User can write RCD1F to 0 to clear RCD1F. DATA2 and RCD2F operate in the same way as DATA1 and RCD1. After DATA1 or DATA2 reception is completed, the Master side should restart the transfer protocol to transmit the next DATA1 and DATA2.

S	Slave address	$\overline{W}$	A	Data 1	A	P

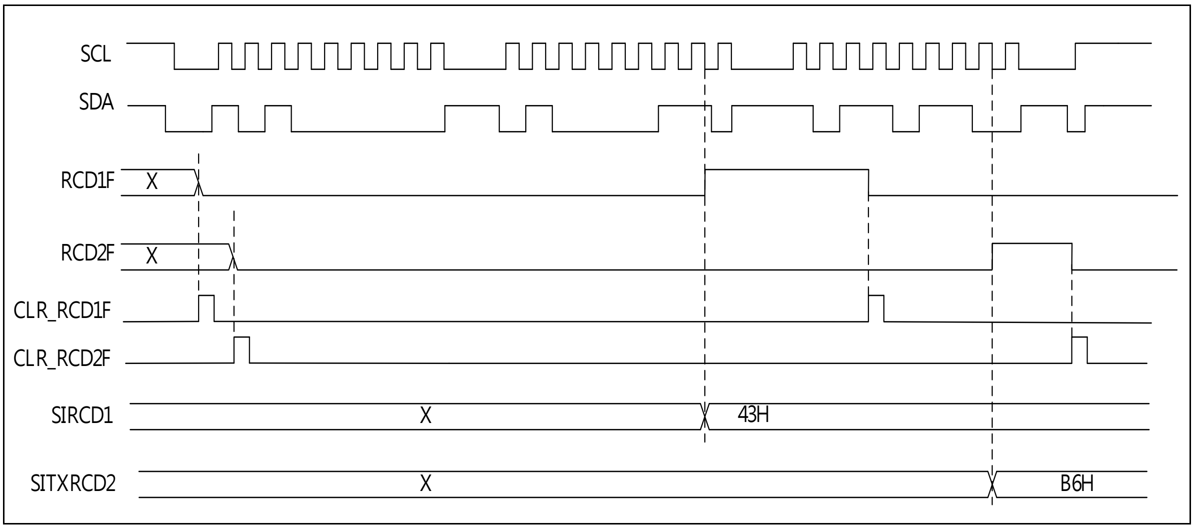
Slave I²C Receive Byte protocol

S	Slave address	$\overline{W}$	A	Data 1	A	Data 2	A	P

Slave I²C Receive Two Byte protocol

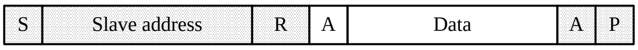
From master to slave
From slave to master

A: acknowledge
S: start condition
P: stop condition



Slave Receive Timing

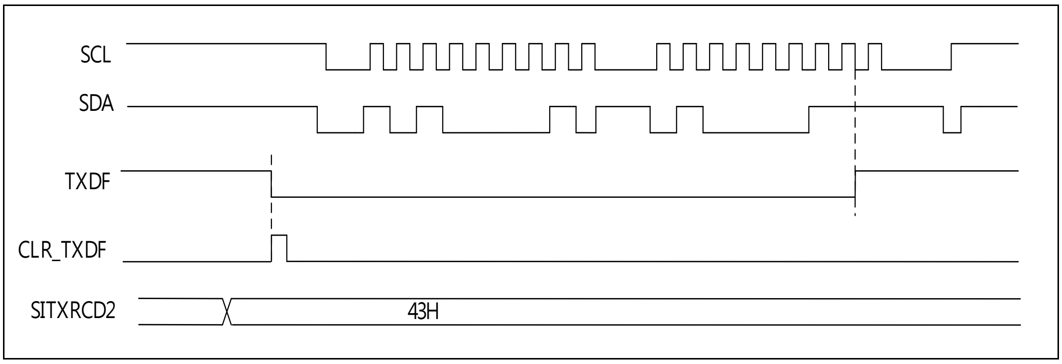
The chip provides Slave I²C interface transmission protocol as following. Slave I²C module allow to transmit one byte data each time after start condition. Before data transmitting, be aware that TXDF must be 0. After data transmission is completed, TXDF will be converted to 1 and an interrupt will be issued according to the user's request. User can use firmware to clear TXDF before transmitting next data again. User can write TXDF to 0 to clear TXDF. After each transmission is completed, the host should restart the transmission protocol to transmit the next data.



Slave I²C Transmit protocol

From master to slave
From slave to master

A: acknowledge
S: start condition
P: stop condition



Slave Transmit Timing

I²C Function Pin	PINMODx	P2.n SFR data	Pin State
I²C Slave SCL	**xx01**	1	Clock input (Hi-Z)
I²C Master/Slaver SDA	**xx00**	1	DATA (Pull-up)

Pin Mode Setting for Slave I²C

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

A9h.6 **I2CE:** I²C interrupt enable
0: Disable I²C interrupt
1: Enable I²C interrupt

SFR **E9h**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SIADR	SA							SIEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	1	0	0	1	0	0

E9h.7~1 **SA:** Slave I²C address assigned

E9h.0 **SIEN:** Slave I²C enable
0: disable
1: enable

SFR **Eah**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SICON	MIIE	TXDIE	RCD2IE	RCD1IE	–	TXDF	RCD2F	RCD1F
R/W	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Reset	0	0	0	0	–	1	0	0

Eah.6 **TXDIE:** Slave I²C transmission completed interrupt enable
0: disable
1: enable

Eah.5 **RCD2IE:** Slave I²C DATA2(SITXRCD2) reception completed interrupt enable
0: disable
1: enable

Eah.4 **RCD1IE:** Slave I²C DATA1(SIRCD1) reception completed interrupt enable
0: disable
1: enable

Eah.2 **TXDF:** Slave I²C transmission completed interrupt flag
Set by H/W when Slave I²C transmission complete, write 0 to clear it

Eah.1 **RCD2F:** Slave I²C DATA2(SITXRCD2) reception completed interrupt flag
Set by H/W when Slave I²C DATA2(SITXRCD2) reception complete, write 0 to clear it

Eah.0 **RCD1F:** Slave I²C DATA1(SIRCD1) reception completed interrupt flag
Set by H/W when Slave I²C DATA1(SIRCD1) reception complete, write 0 to clear it

SFR **Ebh**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SIRCD1	SIRCD1							
R/W	R	R	R	R	R	R	R	R
Reset	–	–	–	–	–	–	–	–

Ebh.7~0 **SIRCD1:** Slave I²C data receive register1 (DATA1)

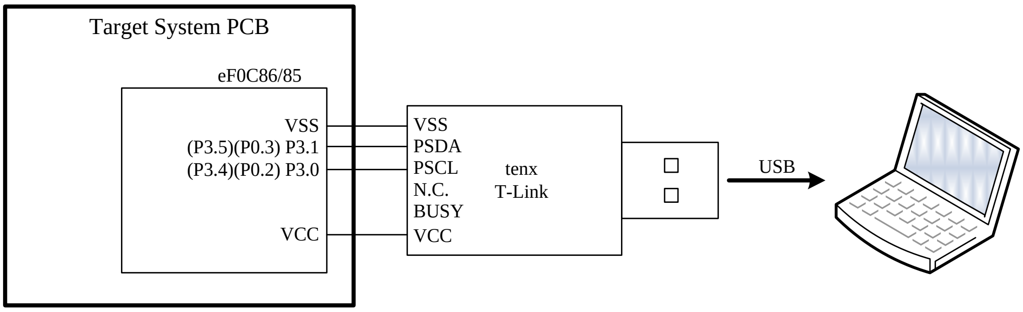
SFR **Ech**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SITXRCD2	SITXRCD2							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	–	–	–	–	–	–

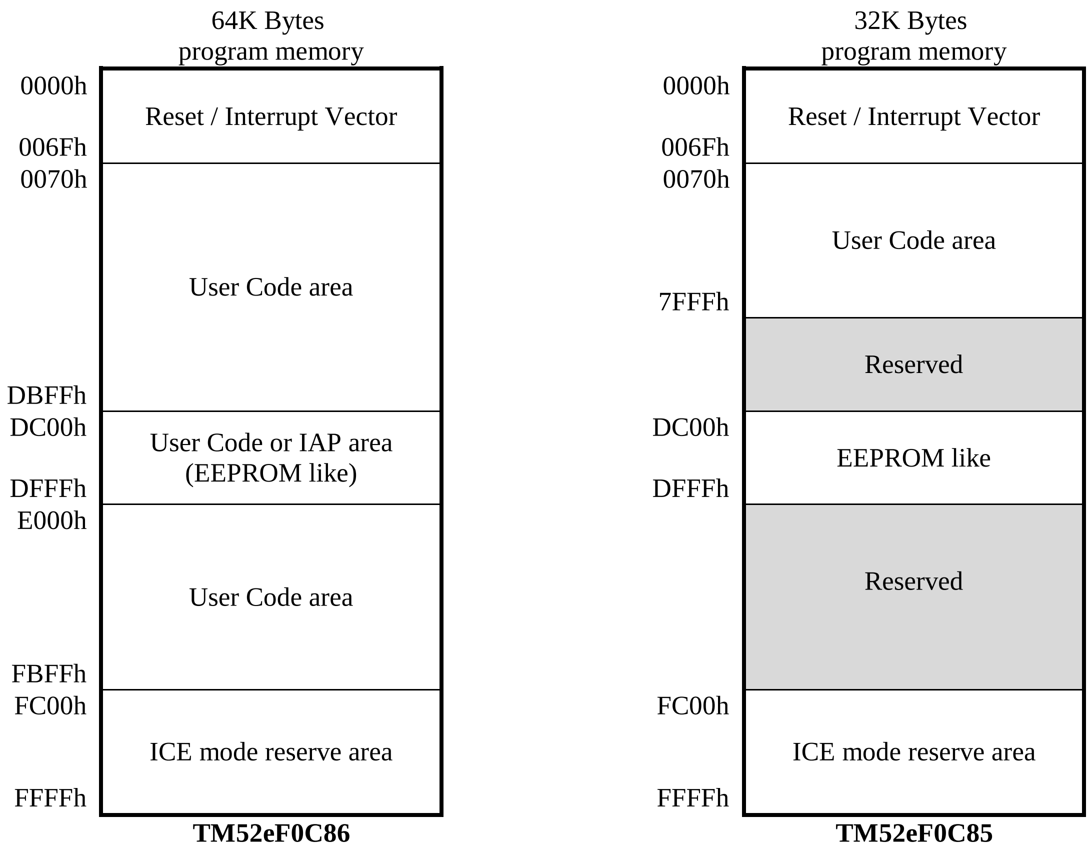
Ech.7~0 **SITXRCD2:** Slave I²C transmit and receive data register
Read: Slave I²C data receive register2 (DATA2)
Write: Slave I²C data transmission register (TXD)

19. 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/P3.1, P02/P30 or P34/P35 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/P3.1, P02/P03 or P34/P35 pins must work in input Mode (PINMODx=xx00b, or xx01b).
3. The Program Memory’s addressing space FC00h~FFFFh 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/P3.1 pin’s can be replaced by P0.2/P0.3 or P34/P35.
   (P0.2/P0.3 or P34/P35 can only support ICE function, not for Writer)
6. SFR PWRSAV (F7h.5) will be cleared when use T-Link module.





SFR & CFGW MAP

Adr	RST	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
80h	0000-0000	**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							
85h	xx00-0000	**INTPORT**	–	–	P5IF	P4IF	P3IF	P2IF	P1IF	P0IF
86h	xxxx-0000	**INTPWM**	–	–	–	–	PWM3IF	PWM2IF	PWM1IF	PWM0IF
87h	0xxx-0000	**PCON**	SMOD	–	–	–	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	0100-0000	**SCON2**	SM2S	–	–	REN2	TB82	RB82	TI2	RI2
8Fh	xxxx-xxxx	**SBUF2**	SBUF2							
90h	1111-1111	**P1**	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
91h	xxxx-x000	**PORTIDX**	–	–	–	–	–	PORTIDX		
94h	0000-0000	**OPTION**	TM3CKS		WDTPSC		ADCKS		TM3PSC	
95h	xx00-x000	**INTFLG**	LVDIF	–	TKIFA	ADIF	–	–	PCIF	TF3
96h	0000-0000	**INTPIN**	PIN7IF	PIN6IF	PIN5IF	PIN4IF	PIN3IF	PIN2IF	PIN1IF	PIN0IF
97h	xxxx-xx00	**SWCMD**	IAPEN / SWRST / WDTO							
98h	0000-0000	**SCON**	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
99h	xxxx-xxxx	**SBUF**	SBUF							
9Ah	0100-0000	**SCON1**	SM1S	–	–	REN1	TB81	RB81	TI1	RI1
9Bh	xxxx-xxxx	**SBUF1**	SBUF1							
9Ch	1100-xxxx	**TKCON3**	TKPDB	TKEOCB	TKIFB	TKXCAPB	JMPVALB			SPREAD
9Dh	1000-0000	**PWM2CON**	PWM2OM		PWM2DZ					
9Eh	0000-0000	**PWMIDX**	PWMIDX							
9Fh	0000-0000	**PWMEN**	PWM3IE	PWM2IE	PWM1IE	PWM0IE	PWM3EN	PWM2EN	PWM1EN	PWM0EN
A0h	1111-1111	**P2**	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
A1h	1010-1010	**PWMCON**	PWM3CKS		PWM2CKS		PWM1CKS		PWM0CKS	
A2h	0001-0001	**PINMOD10**	PINMOD1				PINMOD0			
A3h	0001-0001	**PINMOD32**	PINMOD3				PINMOD2			
A4h	0001-0001	**PINMOD54**	PINMOD5				PINMOD4			
A5h	0001-0001	**PINMOD76**	PINMOD7				PINMOD6			
A6h	0000-0000	**PINMODE**	VBGEN	–	UART1PS	PSEUDOEN	I2CPS		UART0PS	
A7h	xxx1-1111	**TKCHSA**	–	–	–	TKCHSA				
A8h	0x00-0000	**IE**	EA	–	ET2	ES	ET1	EX1	ET0	EX0
A9h	xx00-0000	**INTE1**	PWMIE	I2CE	ES2	SPIE	ADTKIE	LVDIE	PCIE	TM3IE
AAh	xxxx-xxxx	**ADCDL**	ADCDL				–			
ABh	xxxx-xxxx	**ADCDH**	ADCDH							
ACh	xxx1-1111	**TKCHSB**	–	–	–	TKCHSB				
ADh	1100-0000	**TKCON**	TKPDA	TKEOCA	TKRERUN	TKIVCS	TKXCAPA	TKOFFSET	ATKMODE	
AEh	0011-1111	**CHSEL**	ADCVREFS		ADCHS					
AFh	0000-0000	**ATKCHB2**	ATKCHB2							
B0h	1111-1111	**P3**	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
B1h	0000-0111	**LXDCON**	LXDON	LXDDUTY			LEDBRITM	LXDBRIT		
B2h	0000-0000	**LXDCON2**	LCDCKS	LXDPSC		SELLED	LEDHOLD		LEDMODE	
B4h	1111-1111	**TKTMRL**	TKTMRL							
B5h	0000-0000	**TKCON2**	TKFJMP	JMPVALA			TKTMRH			
B6h	0000-0000	**ATKCHB1**	ATKCHB1							
B7h	0000-0000	**ATKCHB0**	ATKCHB0							
B8h	xx00-0000	**IP**	–	–	PT2	PS	PT1	PX1	PT0	PX0
B9h	xx00-0000	**IPH**	–	–	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
Bah	xx00-0000	**IP1**	PPWM	PI2C	PS2	PSPI	PADTKI	PLVD	PPC	PT3
BBh	xx00-0000	**IP1H**	PPWMH	PI2CH	PS2H	PSPIH	PADTKIH	PLVDH	PPCH	PT3H
BCh	0000-0000	**SPCON**	SPEN	MSTR	CPOL	CPHA	SSDIS	LSBF	SPCR	
BDh	0000-0xxx	**SPSTA**	SPIF	WCOL	MODF	RCVOVF	RCVBF	SPBSY	–	–

Adr	RST	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BEh	0000-0000	**SPDAT**	SPDAT							
BFh	0x00-0000	**LVDCON**	LVDM	LVDO	LVDDBS	LVDPD	LVDS			
C0h	1111-1111	**P5**	P5							
C1h	0000-0000	**TKPINSA0**	TKPINSA0							
C2h	0000-0000	**TKPINSA1**	TKPINSA1							
C3h	0000-0000	**TKPINSA2**	TKPINSA2							
C4h	0000-0000	**TKPINSB0**	TKPINSB0							
C5h	0000-0000	**ATKCHA0**	ATKCHA0							
C6h	0000-0000	**ATKCHA1**	ATKCHA1							
C7h	0000-0000	**ATKCHA2**	ATKCHA2							
C8h	0000-0000	**T2CON**	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2N	CPRL2N
C9h	00xx-xxxx	**IAPWE**	IAPWE / IAPTO							
CAh	0000-0000	**RCP2L**	RCP2L							
CBh	0000-0000	**RCP2H**	RCP2H							
CCh	0000-0000	**TL2**	TL2							
CDh	0000-0000	**TH2**	TH2							
CEh	0000-0000	**EXA2**	EXA2							
CFh	0000-0000	**EXA3**	EXA3							
D0h	0000-0000	**PSW**	CY	AC	F0	RS1	RS0	OV	F1	P
D1h	0000-0000	**PWMDH**	PWMDH							
D2h	0000-0000	**PWMDL**	PWMDL							
D5h	x000-0000	**UART2CON**	–	UART2BRP						
D6h	00x0-0011	**LVRCON**	SXTGAIN		–	LVRPD	–	LVRS		
D7h	0000-0000	**TKPINSB1**	TKPINSB1							
D8h	00x0-0011	**CLKCON**	SCKTYPE	FCKTYPE	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
D9h	1111-1111	**PWMPRDH**	PWMPRDH							
DAh	1111-1111	**PWMPRDL**	PWMPRDL							
DDh	x000-0000	**UART1CON**	–	UART1BRP						
DEh	0000-0000	**UART0CON**	UART0BRS	UART0BRP						
DFh	0000-0000	**TKPINSB2**	TKPINSB2							
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							
E5h	0000-0000	**EFTCON**	EFT2CS	EFT1CS	EFT1S		EFTSLOW	EFTWCPU	EFTWOUT	CKHLDE
E6h	0000-0000	**EXA**	EXA							
E7h	0000-0000	**EXB**	EXB							
E8h	1111-1111	**P4**	P4							
E9h	0110-1000	**SIADR**	SA							SIEN
EAh	0000-x100	**SICON**	MIIE	TXDIE	RCD2IE	RCD1IE	–	TXDF	RCD2F	RCD1F
EBh	xxxx-xxxx	**SIRCD1**	SIRCD1							
ECh	xxxx-xxxx	**SITXRCD2**	SITXRCD2							
EEh	xxxx-x1xx	**BOOTV**	–	–	–	–	–	RSTV	BOOTVR	
EFh	xxx0-0000	**PWRCON**	–	–	–	AVPULL	WARMTIME	ENVPULL	PWRIDLE	PWRSLOW
F0h	0000-0000	**B**	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
F1h	1111-1111	**CRCDL**	CRCDL							
F2h	1111-1111	**CRCDH**	CRCDH							
F3h	0000-0000	**CRCIN**	CRCIN							
F5h	xxxx-xxxx	**CFGBG**	–	–	–	BGTRIM				
F6h	xxxx-xxxx	**CFGWL**	–	FRCF						
F7h	0000-1110	**AUX2**	WDTE		PWRSAV	VBGOUT	DIV32	IAPTE		MULDIV16
F8h	0000-0000	**AUX1**	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	T1SEL	DPSEL

Flash Address	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INFO2 0200H	**CFGWH**	PROTN	XRSTEN	–	–	–	–	BOOTV	

SFR & CFGW DESCRIPTION

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
80h	**P0**	7~0	P0	R/W	00h	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
85h	**INTPORT**	5	P5IF	R/W	0h	P5.7~P5.0 pin change interrupt flag, Write 0 to clear P5.7~P5.0 pin change interrupt flag
		4	P4IF	R/W	0h	P4.7~P4.0 pin change interrupt flag, Write 0 to clear P4.7~P4.0 pin change interrupt flag
		3	P3IF	R/W	0h	P3.7~P3.0 pin change interrupt flag, Write 0 to clear P3.7~P3.0 pin change interrupt flag
		2	P2IF	R/W	0h	P2.7~P2.0 pin change interrupt flag, Write 0 to clear P2.7~P2.0 pin change interrupt flag
		1	P1IF	R/W	0h	P1.7~P1.0 pin change interrupt flag, Write 0 to clear P1.7~P1.0 pin change interrupt flag
		0	P0IF	R/W	0h	P0.7~P0.0 pin change interrupt flag, Write 0 to clear P0.7~P0.0 pin change interrupt flag
86h	**INTPWM**	3	PWM3IF	R/W	0h	PWM3 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		2	PWM2IF	R/W	0h	PWM2 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		1	PWM1IF	R/W	0h	PWM1 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		0	PWM0IF	R/W	0h	PWM0 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
87h	**PCON**	7	SMOD	R/W	0	Set 1 to enable UART0 double baud rate
		3	GF1	R/W	0	General purpose flag bit
		2	GF0	R/W	0	General purpose flag bit
		1	PD	R/W	0	Power down control bit, set 1 to enter STOP/Halt 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 Set by H/W when Timer/Counter 1 overflows. Cleared by H/W when CPU vectors into the interrupt service routine.
		6	TR1	R/W	0	Timer1 run control. 1: timer runs; 0: timer stops
		5	TF0	R/W	0	Timer0 overflow flag Set by H/W when Timer/Counter 0 overflows. Cleared by H/W when CPU vectors into the interrupt service routine.
		4	TR0	R/W	0	Timer0 run control. 1:timer runs; 0:timer stops
		3	IE1	R/W	0	External Interrupt 1 (INT1 pin) edge flag Set by H/W when an INT1 pin falling edge is detected. Cleared by H/W when CPU vectors into the interrupt service routine.
		2	IT1	R/W	0	External Interrupt 1 control bit 0: Low level active (level triggered) for INT1 pin 1: Falling edge active (edge triggered) for INT1 pin
		1	IE0	R/W	0	External Interrupt 0 (INT0 pin) edge flag Set by H/W when an INT0 pin falling edge is detected. Cleared by H/W when CPU vectors into the interrupt service routine.
		0	IT0	R/W	0	External Interrupt 0 control bit 0: Low level active (level triggered) for INT0 pin 1: Falling edge active (edge triggered) for INT0 pin

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
89h	**TMOD**	7	GATE1	R/W	0	Timer1 gating control bit 0: Timer1 enable when TR1 bit is set 1: Timer1 enable only while the INT1 pin is high and TR1 bit is set
		6	CT1N	R/W	0	Timer1 Counter/Timer select bit 0: Timer mode, Timer1 data increases at 2 System clock cycle rate 1: Counter mode, Timer1 data increases at T1 pin's negative edge
		5~4	TMOD1	R/W	00	Timer1 mode select 00: 8-bit timer/counter (TH1) and 5-bit prescaler (TL1) 01: 16-bit timer/counter 10: 8-bit auto-reload timer/counter (TL1). Reloaded from TH1 at overflow. 11: Timer1 stops
		3	GATE0	R/W	0	Timer0 gating control bit 0: Timer0 enable when TR0 bit is set 1: Timer0 enable only while the INT0 pin is high and TR0 bit is set
		2	CT0N	R/W	0	Timer0 Counter/Timer select bit 0: Timer mode, Timer0 data increases at 2 System clock cycle rate 1: Counter mode, Timer0 data increases at T0 pin's negative edge
		1~0	TMOD0	R/W	00	Timer0 mode select 00: 8-bit timer/counter (TH0) and 5-bit prescaler (TL0) 01: 16-bit timer/counter 10: 8-bit auto-reload timer/counter (TL0). Reloaded from TH0 at overflow. 11: TL0 is an 8-bit timer/counter. TH0 is an 8-bit timer/counter using Timer1's TR1 and TF1 bits.
8Ah	**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
8Eh	**SCON2**	7	SM2S	R/W	0	UART2 Serial port mode select bit 0: Mode1: 8 bit UART2, Baud Rate is variable 1: Mode3: 9 bit UART2, Baud Rate is variable
		4	REN2	R/W	0	UART2 reception enable 0: Disable reception 1: Enable reception
		3	TB82	R/W	0	Transmit Bit 8, the ninth bit to be transmitted in Mode3
		2	RB82	R/W	0	Receive Bit 8, contains the ninth bit that was received in Mode3
		1	TI2	R/W	0	Transmit interrupt flag Set by H/W at the beginning of the stop bit in Mode 1 & 3. Must be cleared by S/W.
		0	RI2	R/W	0	Receive interrupt flag Set by H/W at the sampling point of the stop bit in Mode 1 & 3. Must be cleared by S/W.
8Fh	**SBUF2**	7~0	SBUF2	R/W	–	UART2 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.
90h	**P1**	7~0	P1	R/W	FFh	Port1 data
91h	**PORTIDX**	2~0	PORTIDX	R/W	00h	PORT index of INTPIN, PINMOD10, PINMOD32, PINMOD54, PINMOD76

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
94h	**OPTION**	7~6	TM3CKS	R/W	0	Timer3 Clock Source Select. 00: SLOW clock (SXT/SRC) 01: FRC/512 10: SLOW clock (SXT/SRC) /2 11: FRC/1024
		5~4	WDTPSC	R/W	00	Watchdog Timer pre-scalar time select 00: 480ms WDT overflow rate 01: 240ms WDT overflow rate 10: 120ms WDT overflow rate 11: 60ms WDT overflow rate
		3~2	ADCKS	R/W	00	ADC clock rate select 00: F_{SYSCLK} /32 01: F_{SYSCLK} /16 10: F_{SYSCLK} /8 11: F_{SYSCLK} /4
		1~0	TM3PSC	R/W	00	Timer3 Interrupt rate 00: Timer3 Interrupt rate is 65536 Timer3 Clock Source cycle 01: Timer3 Interrupt rate is 16384 Timer3 Clock Source cycle 10: Timer3 Interrupt rate is 4096 Timer3 Clock Source cycle 11: Timer3 Interrupt rate is 1024 Timer3 Clock Source cycle
95h	**INTFLG**	7	LVDIF	R	–	Low Voltage Detect flag Set by H/W when a low voltage occurs.
		5	TKIFA	R/W	0	Touch Key A Interrupt Flag Set by H/W at the end of TK conversion if SYSCLK is fast enough. S/W writes DFh to INTFLG or sets the TKSOCA bit to clear this flag.
		4	ADIF	R/W	0	ADC interrupt flag Set by H/W at the end of ADC conversion. S/W writes EFh to INTFLG or sets the ADSOC bit to clear this flag.
		1	PCIF	R/W	0	Port0~Port5 Pin change interrupt flag Set by H/W when Port0~Port5 pin state change is detected and its interrupt enable bit is set. S/W can write 0 to clear all pin interrupt flags (Port0~Port5), it will also clear PIN0IF~PIN7F and P0IF~P5IF.
		0	TF3	R/W	0	Timer3 Interrupt Flag Set by H/W when Timer3 reaches TM3PSC setting cycles. It is cleared automatically when the program performs the interrupt service routine. S/W can write FEh to INTFLG to clear this bit.

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
96h	INTPIN	7	PIN7IF	R/W	0	Px.7 pin change interrupt flag, Write 0 to clear Px.7 pin change interrupt flag port number (x) define by PORTIDX
		6	PIN6IF	R/W	0	Px.6 pin change interrupt flag, Write 0 to clear Px.6 pin change interrupt flag port number (x) define by PORTIDX
		5	PIN5IF	R/W	0	Px.5 pin change interrupt flag, Write 0 to clear Px.5 pin change interrupt flag port number (x) define by PORTIDX
		4	PIN4IF	R/W	0	Px.4 pin change interrupt flag, Write 0 to clear Px.4 pin change interrupt flag port number (x) define by PORTIDX
		3	PIN3IF	R/W	0	Px.3 pin change interrupt flag, Write 0 to clear Px.3 pin change interrupt flag port number (x) define by PORTIDX
		2	PIN2IF	R/W	0	Px.2 pin change interrupt flag, Write 0 to clear Px.2 pin change interrupt flag port number (x) define by PORTIDX
		1	PIN1IF	R/W	0	Px.1 pin change interrupt flag, Write 0 to clear Px.1 pin change interrupt flag port number (x) define by PORTIDX
		0	PIN0IF	R/W	0	Px.0 pin change interrupt flag, Write 0 to clear Px.0 pin change interrupt flag port number (x) define by PORTIDX
97h	SWCMD	7~0	SWRST	W		Write 56h to generate S/W Reset
		7~0	IAPEN	W		Write 65h to set IAPEN control flag; Write other value to clear IAPEN flag. It is recommended to clear it immediately after IAP access.
		1	WDTO	R	0	WatchDog Time-Out flag
		0	IAPEN	R	0	Flag indicates Flash memory sectors can be accessed by IAP or not. This bit combines with MVCLOCK to define the accessible IAP area.
98h	SCON	7	SM0	R/W	0	UART0 Serial port mode select bit 0, 1 (SM0, SM1) = 00: Mode0: 8 bit shift register, Baud Rate=F_{SYSCLK}/2 01: Mode1: 8 bit UART0, Baud Rate is variable 10: Mode2: 9 bit UART0, Baud Rate=F_{SYSCLK}/32 or /64 11: Mode3: 9 bit UART0, Baud Rate is variable
		6	SM1	R/W	0	
		5	SM2	R/W	0	Serial port mode select bit 2 SM2 enables multiprocessor communication over a single serial line and modifies the above as follows. In Modes 2 & 3, if SM2 is set then the received interrupt will not be generated if the received ninth data bit is 0. In Mode 1, the received interrupt will not be generated unless a valid stop bit is received. In Mode 0, SM2 should be 0.
		4	REN	R/W	0	Set 1 to enable UART0 Reception
		3	TB8	R/W	0	Transmitter bit 8, ninth bit to transmit in Modes 2 and 3
		2	RB8	R/W	0	Receive Bit 8, contains the ninth bit that was received in Mode 2 and 3 or the stop bit is Mode 1 if SM2=0
		1	TI	R/W	0	Transmit Interrupt flag Set by H/W at the end of the eighth bit in Mode 0, or at the beginning of the stop bit in other modes. Must be cleared by S/W
		0	RI	R/W	0	Receive Interrupt flag Set by H/W at the end of the eighth bit in Mode 0, or at the sampling point of the stop bit in other modes. Must be cleared by S/W.
99h	SBUF	7~0	SBUF	R/W	–	UART0 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
9Ah	**SCON1**	7	SM1S	R/W	0	UART1 Serial port mode select bit 0: Mode1: 8 bit UART1, Baud Rate is variable 1: Mode3: 9 bit UART1, Baud Rate is variable
		4	REN1	R/W	0	UART1 reception enable 0: Disable reception 1: Enable reception
		3	TB81	R/W	0	Transmit Bit 8, the ninth bit to be transmitted in Mode3
		2	RB81	R/W	0	Receive Bit 8, contains the ninth bit that was received in Mode3
		1	TI1	R/W	0	Transmit interrupt flag Set by H/W at the beginning of the stop bit in Mode 1 & 3. Must be cleared by S/W.
		0	RI1	R/W	0	Receive interrupt flag Set by H/W at the sampling point of the stop bit in Mode 1 & 3. Must be cleared by S/W.
9Bh	**SBUF1**	7~0	SBUF1	R/W	–	UART1 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.
9Ch	**TKCON3**	7	TKPDB	R/W	1	Touch Key B Power Down 0: Touch Key B enable; 1: Touch Key B disable
		6	TKEOCB	R	1	Touch Key B end of conversion flag 0: Indicates conversion is in progress 1: Indicates conversion is finished
		5	TKIFB	R/W	0	Touch Key B Interrupt Flag Set by H/W at the end of TK conversion if SYSCLK is fast enough. S/W clear TKIFB or sets the TKSOCB bit to clear this flag.
		4	TKXCAPB	R/W	0	Touch Key B external capacitor select 0: Keep 0, disable Touch Key B external capacitor 1: reserved (Do not set to 1)
		3~1	JMPVALB	R/W	0	Touch Key Clock frequency fine tune , only available in TKFJMP=0 000=frequency slowest, 111=frequency fastest
		0	SPREAD	R/W	0	TK spread spectrum 0: Disable 1: Enable
9Dh	**PWM2CON**	7~6	PWM2OM	R/W	10	output mode 00: mode 0 01: mode 1 10: mode 2 11: mode 3
		5~0	PWM2DZ	R/W	0	PWM2 Dead zone Control 0000: dead zone disabled 0001: Dead zone width 1*Tpwmclk 0010: Dead zone width 2*Tpwmclk … 1111: Dead zone width 15*Tpwmclk
9Eh	**PWMIDX**	7~0	PWMIDX	R/W	FFh	PWM period and duty index. See table 10.1 for more detail 0xh: PWM0 Period/Duty access 1xh: PWM1 Period/Duty access 2xh: PWM2 Period/Duty access 3xh: PWM30~PWM35 Period/Duty access 30h: PWM30 Period/Duty access 31h: PWM31 Period/Duty access 32h: PWM32 Period/Duty access 33h: PWM33 Period/Duty access 34h: PWM34 Period/Duty access 35h: PWM35 Period/Duty access

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
9Fh	**PWMEN**	7	PWM3IE	R/W	0	PWM3 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		6	PWM2IE	R/W	0	PWM2 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		5	PWM1IE	R/W	0	PWM1 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		4	PWM0IE	R/W	0	PWM0 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		3	PWM3EN	R/W	0	PWM3 enable 0: PWM3 is cleared and held 1: PWM3 is running
		2	PWM2EN	R/W	0	PWM2 enable 0: PWM2 is cleared and held 1: PWM2 is running
		1	PWM1EN	R/W	0	PWM1 enable 0: PWM1 is cleared and held 1: PWM1 is running
		0	PWM0EN	R/W	0	PWM0 enable 0: PWM0 is cleared and held 1: PWM0 is running
A0h	**P2**	7~0	P2	R/W	FF	P2.7~P2.0 data
A1h	**PWMCON**	7~6	PWM3CKS	R/W	10	PWM3 clock source 00: F_{SYSCLK} 01: F_{SYSCLK} 10: FRC 11: FRC x 2
		5~4	PWM2CKS	R/W	10	PWM2 clock source 00: F_{SYSCLK} 01: F_{SYSCLK} 10: FRC 11: FRC x 2
		3~2	PWM1CKS	R/W	10	PWM1 clock source 00: F_{SYSCLK} 01: F_{SYSCLK} 10: FRC 11: FRC x 2
		1~0	PWM0CKS	R/W	10	PWM0 clock source 00: F_{SYSCLK} 01: F_{SYSCLK} 10: FRC 11: FRC x 2
A2h	**PINMOD10**	7~4	PINMOD1	R/W	0001	Px.1 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD0	R/W	0001	Px.0 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A3h	**PINMOD32**	7~4	PINMOD3	R/W	0001	Px.3 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD2	R/W	0001	Px.2 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A4h	**PINMOD54**	7~4	PINMOD5	R/W	0001	Px.5 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD4	R/W	0001	Px.4 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
A5h	**PINMOD76**	7~4	PINMOD7	R/W	0001	Px.7 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD6	R/W	0001	Px.6 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A6h	**PINMOD**	7	VBGEN	R/W	0	Force VBG generator enable 0: VBG generator is automatically enable and disable 1: Force VBG generator enable included in IDLE mode but disabled in Stop/Halt mode
		5	UART1PS	R/W	0	UART1 Pin Select 0: RXD1/TXD1 = P3.6/P3.7 1: RXD1/TXD1 = P5.4/P5.5
		4	PSEDOEN	R/W	0	P30~P32 pseudo open-drain 0: disable 1: enable
		3~2	I2CPS	R/W	0	I2C Pin Select 00: SCL/SDA = P3.4/P3.5 01: SCL/SDA = P3.0/P3.1 1x: SCL/SDA = P0.2/P0.3
		1~0	UART0PS	R/W	0	UART0 Pin Select 00: RXD0/TXD0 = P3.0/P3.1 01: RXD0/TXD0 = P3.4/P3.5 10: RXD0/TXD0 = P4.4/P4.5 11: Reservd
A7h	**TKCHSA**	4~0	TKCHS	R/W	1Fh	Specify the first Touch Key A scan channel 00000: TKA00 00001: TKA01 00010: TKA02 00011: TKA03 00100: TKA04 00101: TKA05 00110: TKA06 00111: TKA07 01000: TKA08 01001: TKA09 01010: TKA10 01011: TKA11 01100: TKA12 01101: TKA13 01110: TKA14 01111: TKA15 10000: TKA16 10001: TKA17 10010: TKA18 10011: TKA19 10100: TKA20 10111: TKACAP: internal reference capacitor channel
A8h	**IE**	7	EA	R/W	0	Global interrupt enable control. 0: Disable all Interrupts. 1: Each interrupt is enabled or disabled by its own interrupt control bit.
		5	ET2	R/W	0	Set 1 to enable Timer2 interrupt
		4	ES	R/W	0	Set 1 to enable Serial Port (UART0) 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 & Stop/Halt mode wake up capability
		1	ET0	R/W	0	Set 1 to enable Timer0 Interrupt
		0	EX0	R/W	0	Set 1 to enable external INT0 pin Interrupt & Stop/Halt mode wake up capability

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
A9h	**INTE1**	7	PWMIE	R/W	0	Set 1 to enable PWM0~PWM2 interrupt
		6	I2CE	R/W	0	Set 1 to enable I^2C (master/slave) interrupt
		5	ES2	R/W	0	Set 1 to enable Serial Port (UART1/UART2) interrupt
		4	SPIE	R/W	0	Set 1 to enable SPI interrupt
		3	ADTKIE	R/W	0	Set 1 to enable ADC/Touch Key Interrupt
		2	LVDIE	R/W	0	Set 1 to enable LVD interrupt.
		1	PCIE	R/W	0	Port0~Port5 pin change interrupt enable. This bit does not affect Stop/Halt mode wake up capability. 0: Disable Port0~Port5 pin change interrupt 1: Enable Port0~Port5 pin change interrupt
		0	TM3IE	R/W	0	Set 1 to enable Timer3 Interrupt and Halt mode wake up
AAh	**ADCDL**	7~4	ADCDL	R	–	ADC data bit 3~0
ABh	**ADCDH**	7~0	ADCDH	R	–	ADC data bit 11~4
ACh	**TKCHSB**	4~0	TKCHS	R/W	1Fh	Specify the first Touch Key B scan channel 00000: TKB00 00001: TKB01 00010: TKB02 00011: TKB03 00100: TKB04 00101: TKB05 00110: TKB06 00111: TKB07 01000: TKB08 01001: TKB09 01010: TKB10 01011: TKB11 01100: TKB12 01101: TKB13 01110: TKB14 01111: TKB15 10000: TKB16 10001: TKB17 10010: TKB18 10011: TKB19 10100: TKB20 10111: TKBCAP: internal reference capacitor channel

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
ADh	TKCON	7	TKPDA	R/W	1	Touch Key A Power Down 0: Touch Key A enable; 1: Touch Key A disable
		6	TKEOCA	R	1	Touch Key A end of conversion flag 0: Indicates conversion is in progress 1: Indicates conversion is finished
		5	TKRERUN	R/W	0	TK A/B Auto re-start 0: Auto re-start disable. TKSOCA/B needs to be executed once for each TK A/B conversion 1: Auto re-start enable. After TKSOCA/B is executed once, TK A/B will be converted continuously without re-executing TKSOCA/B
		4	TKIVCS	R/W	0	Touch Key internal voltage control select 0: VCHG=2.8V; VINT=1.4V 1: VCHG=3.6V; VINT=1.8V
		3	TKXCAPA	R/W	0	Touch Key A external capacitor select 0: Keep 0, disable Touch Key A external capacitor 1: reserved (Do not set to 1)
		2	TKOFFSET	R/W	0	status of non-scan TK 0: connect to VSS 1: connect to AC shielding , connect to VSS@EOC
		1~0	ATKMODE	R/W	00	Touch Key Scan Mode 00: TKA and TKB scan method, each channel scan 1 time, max 22 TK channels 01: TKA and TKB scan method, each channel scan 2 times, max 16 TK channels 10: TKA and TKB scan method, each channel scan 4 times, max 8 TK channels 11: TKA and TKB scan method, each channel scan 8 times, max 4 TK channels
AEh	CHSEL	7~6	ADCVREFS	R/W	00	ADC reference voltage. When ADCHS is selected to VBG, ADCVREFS must be set to VCC, otherwise ADC conversion will be invalid 00: VCC 01: 2.5V 10: 3V 11: 4V
		5~0	ADCHS	R/W	1111	ADC channel select 000000: AD00 000001: AD01 … 101001: AD41 101010: Reserved 101011: V_{BG} (Internal Bandgap Reference Voltage) 101100: 1/4V_{CC} (Internal Reference Voltage)
AFh	ATKCHB2	7	ATKCHB2	R/W	0	TKBCAP (TKB23) internal reference capacitor channel scan enable: 0: disable 1: enable
		4		R/W	0	TKB20 scan enable: 0: disable 1: enable
		3		R/W	0	TKB19 scan enable: 0: disable 1: enable
		2		R/W	0	TKB18 scan enable: 0: disable 1: enable
		1		R/W	0	TKB17 scan enable: 0: disable 1: enable
		0		R/W	0	TKB16 scan enable: 0: disable 1: enable
B0h	P3	7~0	P3	R/W	FFh	Port3 data

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
B1h	**LXDCON**	7	LXDON	R/W	0	LCD/LED enable 0: LCD/LED disable 1: LCD/LED enable
		4~4	LXDDUTY	R/W	00	LCD/LED duty select LCD select (SELLED=0): 000: 1/4 Duty, COM 0~3 001: 1/4 Duty, COM 0~3 010: 1/5 Duty, COM 0~4 011: 1/6 Duty, COM 0~5 100: 1/6 Duty, COM 0~5 101: 1/8 Duty, COM 0~7 110: 1/8 Duty, COM 0~7 111: 1/8 Duty, COM 0~7 LED select: Matrix mode (SELLED=1, LEDMODE=00b) 000: 1/2 Duty, LCOM 0~1 001: 1/3 Duty, LCOM 0~2 010: 1/4 Duty, LCOM 0~3 011: 1/5 Duty, LCOM 0~4 100: 1/6 Duty, LCOM 0~5 101: 1/7 Duty, LCOM 0~6 110: 1/8 Duty, LCOM 0~7 111: 1/8 Duty, LCOM 0~7 LED select: Dot Matrix mode (SELLED=1, LEDMODE=10b) 000: 4x4, LED 0~4 001: 5x5, LED 0~5 010: 6x6, LED 0~6 011: 6x7, LED 0~6 100: 7x7, LED 0~7 101: 7x8, LED 0~7 110: Reserved 111: Reserved
		3	LEDBRITM	R/W	0	LED Brightness Mode 0: Uniform brightness mode 1: Brightness enhancement mode
		2~0	LXDBRIT	R/W	111	LCD/LED Brightness control 000: Level 0 (Darkest) … 111: Level 7 (Brightest)

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
B2h	**LCNCON2**	7	LCDCKS	R/W	0	LCD clock source select and LED Clock Source fixed as FRC 0: SRC/4 1: SXT/2
		6~5	LXDPSC	R/W	00	LCD/LED clock prescaler select 00: LCD/LED clock is FRC divided by 64 01: LCD/LED clock is FRC divided by 32 10: LCD/LED clock is FRC divided by 16 11: LCD/LED clock is FRC divided by 8
		4	SELLED	R/W	0	LCD/LED function select 0: LCD 1: LED
		3	LEDHOLD	R/W	0	Keep at 0, cannot be set to 1
		1~0	LEDMODE	R/W	00	LED Mode select 00: Matrix scan mode 01: Reserved 10: Dot Matrix scan mode 11: Reserved
B4h	**TKTMRL**	7~0	TKTMRL	R/W	FFh	Touch Key A/B Scan length bit 7~0 adjustment 00: shortest, FF: longest
B5h	**TKCON2**	7	TKFJMP	R/W	0	Internal Touch Key clock frequency auto adjust option 0: Disable 1: Enable
		6~4	JMPVAL	R/W	000	Touch Key Clock frequency fine tune , only available in TKFJMP=0 00: frequency slowest … 11: frequency fastest
		1~0	TKTMRH	R/W	00	Touch Key A/B Scan length 9~8 adjustment 00: shortest … 11: longest
B6h	**ATKCHB1**	7	ATKCHB1	R/W	0	TKB15 scan enable: 0: disable 1: enable
		6		R/W	0	TKB14 scan enable: 0: disable 1: enable
		5		R/W	0	TKB13 scan enable: 0: disable 1: enable
		4		R/W	0	TKB12 scan enable: 0: disable 1: enable
		3		R/W	0	TKB11 scan enable: 0: disable 1: enable
		2		R/W	0	TKB10 scan enable: 0: disable 1: enable
		1		R/W	0	TKB09 scan enable: 0: disable 1: enable
		0		R/W	0	TKB08 scan enable: 0: disable 1: enable
B7h	**ATKCHB0**	7	ATKCHB0	R/W	0	TKB07 scan enable: 0: disable 1: enable
		6		R/W	0	TKB06 scan enable: 0: disable 1: enable
		5		R/W	0	TKB05 scan enable: 0: disable 1: enable
		4		R/W	0	TKB04 scan enable: 0: disable 1: enable
		3		R/W	0	TKB03 scan enable: 0: disable 1: enable
		2		R/W	0	TKB02 scan enable: 0: disable 1: enable
		1		R/W	0	TKB01 scan enable: 0: disable 1: enable
		0		R/W	0	TKB00 scan enable: 0: disable 1: enable

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
B8h	**IP**	5	PT2	R/W	0	Timer2 Interrupt Priority Low bit
		4	PS	R/W	0	Serial Port (UART1) Interrupt Priority Low bit
		3	PT1	R/W	0	Timer1 Interrupt Priority Low bit
		2	PX1	R/W	0	External INT1 Pin Interrupt Priority Low bit
		1	PT0	R/W	0	Timer0 Interrupt Priority Low bit
		0	PX0	R/W	0	External INT0 Pin Interrupt Priority Low bit
B9h	**IPH**	5	PT2H	R/W	0	Timer2 Interrupt Priority High bit
		4	PSH	R/W	0	Serial Port (UART1) Interrupt Priority High bit
		3	PT1H	R/W	0	Timer1 Interrupt Priority High bit
		2	PX1H	R/W	0	External INT1 Pin Interrupt Priority High bit
		1	PT0H	R/W	0	Timer0 Interrupt Priority High bit
		0	PX0H	R/W	0	External INT0 Pin Interrupt Priority High bit
BAh	**IP1**	7	PPWM	R/W	0	PWM Interrupt Priority Low bit
		6	PI2C	R/W	0	I2C Interrupt Priority Low bit
		5	PS2	R/W	0	Serial Port (UART2) interrupt priority low bit
		4	PSPI	R/W	0	SPI interrupt priority low bit
		3	PADTKI	R/W	0	ADC/Touch Key Interrupt Priority Low bit
		2	PLVD	R/W	0	External INT2~INT9 Pin Interrupt Priority Low bit
		1	PPC	R/W	0	Port0~Port5 pin change Interrupt Priority Low bit
		0	PT3	R/W	0	Timer3 Interrupt Priority Low bit
BBh	**IP1H**	7	PPWMH	R/W	0	PWM Interrupt Priority High bit
		6	PI2CH	R/W	0	I2C Interrupt Priority High bit
		5	PS2H	R/W	0	Serial Port (UART2) interrupt priority high bit
		4	PSPIH	R/W	0	SPI interrupt priority high bit
		3	PADTKIH	R/W	0	ADC/Touch Key Interrupt Priority High bit
		2	PLVDH	R/W	0	External INT2~INT9 Pin Interrupt Priority High bit
		1	PPCH	R/W	0	Port0~Port5 Interrupt Priority High bit
		0	PT3H	R/W	0	Timer3 Interrupt Priority High bit
BCh	**SPCON**	7	SPEN	R/W	0	SPI enable 0: SPI disable 1: SPI enable
		6	MSTR	R/W	0	Master mode enable 0: Slave mode 1: Master mode
		5	CPOL	R/W	0	SPI clock polarity 0: SCK is low in idle state 1: SCK is high in idle state
		4	CPHA	R/W	0	SPI clock phase 0: Data sample on first edge of SCK period 1: Data sample on second edge of SCK period
		3	SSDIS	R/W	0	SS pin disable 0: Enable SS pin 1: Disable SS pin
		2	LSBF	R/W	0	LSB first 0: MSB first 1: LSB first
		1~0	SPCR	R/W	00	SPI clock rate 00: FSYSCLK/2 01: FSYSCLK/4 10: FSYSCLK/8 11: FSYSCLK/16

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
BDh	**SPSTA**	7	SPIF	R/W	0	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.
		6	WCOL	R/W	0	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.
		5	MODF	R/W	0	Mode fault interrupt flag Set by H/W when SSDIS is cleared and SS pin is pulled low in Master mode. Write 0 to this bit will clear this flag. When this bit is set, the SPEN and MSTR in SPCON will be cleared by H/W.
		4	RCVOVF	R/W	0	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.
		3	RCVBF	R/W	0	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.
		2	SPBSY	R	0	SPI busy flag Set by H/W when a SPI transfer is in progress.
BEh	**SPDAT**	7~0	SPDAT	R/W	0	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.
BFh	**LVDCON**	7	LVDM	R/W	0	Low Voltage Detect interrupt enable 0: LVDIF =1 and LVDO =1 while VCC < LVDS 1: LVDIF =1 and LVDO =1 while VCC > LVDS
		6	LVDO	R	-	Low Voltage Detect output
		5	LVDDBS	R/W	0	Low Voltage Detect debounce select 0: Disable 1: Enable
		4	LVDPD	R/W	0	Low Voltage Detect select 0: Enable LVD 1: Disable LVD
		3~0	LVDS	R/W	0	Low Voltage Detect select 0000: Set LVD at 2.52V 0001: Set LVD at 2.62V 0010: Set LVD at 2.74V 0011: Set LVD at 2.86V 0100: Set LVD at 2.99V 0101: Set LVD at 3.1V 0110: Set LVD at 3.23V 0111: Set LVD at 3.35V 1000: Set LVD at 3.48V 1001: Set LVD at 3.6V 1010: Set LVD at 3.72V 1011: Set LVD at 3.84V 1100: Set LVD at 3.96V 1101: Set LVD at 4.08V 1110: Set LVD at 4.2V 1111: Set LVD at 4.32V
C0h	**P5**	7~0	P5	R/W	FFh	Port5 data

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
C1h	**TKPINSA0**	7	TKPINSA0	R/W	0	TKA07 Pin fix as TK channel: 0: disable 1: enable
		6			0	TKA06 Pin fix as TK channel: 0: disable 1: enable
		5			0	TKA05 Pin fix as TK channel: 0: disable 1: enable
		4			0	TKA04 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKA03 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKA02 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKA01 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKA00 Pin fix as TK channel: 0: disable 1: enable
C2h	**TKPINSA1**	7	TKPINSA1	R/W	0	TKA15 Pin fix as TK channel: 0: disable 1: enable
		6			0	TKA14 Pin fix as TK channel: 0: disable 1: enable
		5			0	TKA13 Pin fix as TK channel: 0: disable 1: enable
		4			0	TKA12 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKA11 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKA10 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKA09 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKA08 Pin fix as TK channel: 0: disable 1: enable
C3h	**TKPINSA2**	4	TKPINSA2	R/W	0	TKA20 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKA19 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKA18 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKA17 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKA16 Pin fix as TK channel: 0: disable 1: enable
C4h	**TKPINSB0**	7	TKPINSB0	R/W	0	TKB07 Pin fix as TK channel: 0: disable 1: enable
		6			0	TKB06 Pin fix as TK channel: 0: disable 1: enable
		5			0	TKB05 Pin fix as TK channel: 0: disable 1: enable
		4			0	TKB04 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKB03 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKB02 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKB01 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKB00 Pin fix as TK channel: 0: disable 1: enable
C5h	**ATKCHA0**	7	ATKCHB0	R/W	0	TKB07 scan enable: 0: disable 1: enable
		6		R/W	0	TKB06 scan enable: 0: disable 1: enable
		5		R/W	0	TKB05 scan enable: 0: disable 1: enable
		4		R/W	0	TKB04 scan enable: 0: disable 1: enable
		3		R/W	0	TKB03 scan enable: 0: disable 1: enable
		2		R/W	0	TKB02 scan enable: 0: disable 1: enable
		1		R/W	0	TKB01 scan enable: 0: disable 1: enable
		0		R/W	0	TKB00 scan enable: 0: disable 1: enable
C6h	**ATKCHA1**	7	ATKCHB1	R/W	0	TKB15 scan enable: 0: disable 1: enable
		6		R/W	0	TKB14 scan enable: 0: disable 1: enable
		5		R/W	0	TKB13 scan enable: 0: disable 1: enable
		4		R/W	0	TKB12 scan enable: 0: disable 1: enable
		3		R/W	0	TKB11 scan enable: 0: disable 1: enable
		2		R/W	0	TKB10 scan enable: 0: disable 1: enable
		1		R/W	0	TKB09 scan enable: 0: disable 1: enable
		0		R/W	0	TKB08 scan enable: 0: disable 1: enable
C7h	**ATKCHA2**	7	ATKCHA2	R/W	0	TKACAP (TKA23) internal reference capacitor channel scan enable: 0: disable 1: enable
		4		R/W	0	TKB20 scan enable: 0: disable 1: enable
		3		R/W	0	TKB19 scan enable: 0: disable 1: enable
		2		R/W	0	TKB18 scan enable: 0: disable 1: enable
		1		R/W	0	TKB17 scan enable: 0: disable 1: enable
		0		R/W	0	TKB16 scan enable: 0: disable 1: enable

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
C8h	**T2CON**	7	TF2	R/W	0	Timer2 overflow flag Set by H/W when Timer/Counter 2 overflows unless RCLK=1 or TCLK=1. This bit must be cleared by S/W.
		6	EXF2	R/W	0	T2EX interrupt pin falling edge flag Set when a capture or a reload is caused by a negative transition on T2EX pin if EXEN2=1. This bit must be cleared by S/W.
		5	RCLK	R/W	0	UART receive clock control bit 0: Use Timer1 overflow as receive clock for serial port in mode 1 or 3 1: Use Timer2 overflow as receive clock for serial port in mode 1 or 3
		4	TCLK	R/W	0	UART transmit clock control bit 0: Use Timer1 overflow as transmit clock for serial port in mode 1 or 3 1: Use Timer2 overflow as transmit clock for serial port in mode 1 or 3
		3	EXEN2	R/W	0	T2EX pin enable 0: T2EX pin disable 1: T2EX pin enable, it cause a capture or reload when a negative transition on T2EX pin is detected if RCLK=TCLK=0
		2	TR2	R/W	0	Timer2 run control 0:timer stops 1:timer runs
		1	CT2N	R/W	0	Timer2 Counter/Timer select bit 0: Timer mode, Timer2 data increases at 2 System clock cycle rate 1: Counter mode, Timer2 data increases at T2 pin's negative edge
		0	CPRL2N	R/W	0	Timer2 Capture/Reload control bit 0: Reload mode, auto-reload on Timer2 overflows or negative transitions on T2EX pin if EXEN2=1. 1: Capture mode, capture on negative transitions on T2EX pin if EXEN2=1. If RCLK=1 or TCLK=1, CPRL2N is ignored and timer is forced to auto-reload on Timer2 overflow.
C9h	**IAPWE**	7~0	IAPWE	W	–	Write 4Ah to enable one byte IAP write to ROM[FA00~FBFF] Write 4Ch to enable one byte IAP write to ROM[FC00~FDFF] Write BAh to enable ERASE 512 byte of ROM[FA00~FBFF] Write BCh to enable ERASE 512 byte of ROM[FC00~FDFF] Write other value to disable IAP write
		7	IAPWE	R	0	Flag indicates Flash memory can be written by IAP or not 0: IAP Write/Erase disable 1: IAP Write/Erase enable
		6	IAPTO	R	0	IAP Time-Out flag Set by H/W when IAP Time-out occurs. Cleared by H/W when IAPWE=0.
CAh	**RCP2L**	7~0	RCP2L	R/W	00h	Timer2 reload/capture data low byte
CBh	**RCP2H**	7~0	RCP2H	R/W	00h	Timer2 reload/capture data high byte
CCh	**TL2**	7~0	TL2	R/W	00h	Timer2 data low byte
CDh	**TH2**	7~0	TH2	R/W	00h	Timer2 data high byte
CEh	**EXA2**	7~0	EXA2	R/W	00h	Expansion accumulator 2
CFh	**EXA3**	7~0	EXA3	R/W	00h	Expansion accumulator 3
D0h	**PSW**	7	CY	R/W	0	ALU carry flag
		6	AC	R/W	0	ALU auxiliary carry flag
		5	F0	R/W	0	General purpose user-definable flag
		4	RS1	R/W	0	Register Bank Select bit 1
		3	RS0	R/W	0	Register Bank Select bit 0
		2	OV	R/W	0	ALU overflow flag
		1	F1	R/W	0	General purpose user-definable flag
		0	P	R/W	0	Parity flag

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
D1h	**PWMDH**	7~0	PWM0DH	R/W	80H	PWM duty high byte, index by PWMIDX See table 10.1 for more detail PWMIDX = 0xh: PWM0DH access PWMIDX = 1xh: PWM1DH access PWMIDX = 2xh: PWM2DH access PWMIDX = 30h: PWM30DH access PWMIDX = 31h: PWM31DH access PWMIDX = 32h: PWM32DH access PWMIDX = 33h: PWM33DH access PWMIDX = 34h: PWM34DH access PWMIDX = 35h: PWM35DH access write sequence: PWMDL then PWMDH read sequence: PWMDH then PWMDL
D2h	**PWMDL**	7~0	PWM0DL	R/W	00H	PWM duty low byte, index by PWMIDX See table 10.1 for more detail PWMIDX = 0xh: PWM0DH access PWMIDX = 1xh: PWM1DH access PWMIDX = 2xh: PWM2DH access PWMIDX = 30h: PWM30DH access PWMIDX = 31h: PWM31DH access PWMIDX = 32h: PWM32DH access PWMIDX = 33h: PWM33DH access PWMIDX = 34h: PWM34DH access PWMIDX = 35h: PWM35DH access write sequence: PWMDL then PWMDH read sequence: PWMDH then PWMDL
D5h	**UART2CON**	6~0	UART2BRP	R/W	00H	UART2 baud rate pre-scaler
D6h	**LVRCON**	7~6	SXTGAIN	R/W	00	SXT GAIN select 00: lowest ~ 11: highest
		4	LVRPD	R/W	0	Low Voltage Reset function select 0: Disable 1: Enable
		2~0	LVRS	R/W	0000	Low Voltage Reset function select 000: Set LVR at 2.52V 001: Set LVR at 2.74V 010: Set LVR at 2.99V 011: Set LVR at 3.23V 100: Set LVR at 3.48V 101: Set LVR at 3.72V 110: Set LVR at 3.96V 111: Set LVR at 4.2V
D7h	**TKPINSB1**	7	TKPINSB1	R/W	0	TKB15 Pin fix as TK channel: 0: disable 1: enable
		6			0	TKB14 Pin fix as TK channel: 0: disable 1: enable
		5			0	TKB13 Pin fix as TK channel: 0: disable 1: enable
		4			0	TKB12 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKB11 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKB10 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKB09 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKB08 Pin fix as TK channel: 0: disable 1: enable
D8h	**CLKCON**	7	SCKTYPE	R/W	0	Slow clock Type. This bit can be changed only in Fast mode (SELFCK=1) 0: SRC 1: SXT
		6	FCKTYPE	R/W	0	Fast clock type. This bit can be changed only in Slow mode (SELFCK=0). 0: FRC

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
						1: FXT
		5	STPSCK	R/W	1	Set 1 to stop Slow clock after PD=1 (Halt / Stop mode control)
		4	STPPCK	R/W	0	Set 1 to stop UART/Timer0/1/2 clock in Idle mode for current reducing.
		3	STPFCK	R/W	0	Set 1 to stop Fast clock for power saving in Slow/Idle mode. This bit can be changed only in Slow mode.
		2	SELFCK	R/W	0	System clock select. This bit can be changed only when STPFCK=0. 0: Slow clock 1: Fast clock
		1~0	CLKPSC	R/W	11	System clock prescaler. Effective after 16 clock cycles (Max.) delay. 00: System clock is Fast/Slow clock divided by 16 01: System clock is Fast/Slow clock divided by 4 10: System clock is Fast/Slow clock divided by 2 11: System clock is Fast/Slow clock divided by 1
D9h	**PWMPRDH**	7~0	PWMPRDH	R/W	FFH	PWM period high byte, index by PWMIDX See table 10.1 for more detail PWMIDX = 0xh: PWM0PRDH access PWMIDX = 1xh: PWM1PRDH access PWMIDX = 2xh: PWM2PRDH access PWMIDX = 3xh: PWM3PRDH access write sequence: PWMPRDL then PWMPRDH read sequence: PWMPRDH then PWMPRDL
DAh	**PWMPRDL**	7~0	PWMPRDL	R/W	FFH	PWM period low byte, index by PWMIDX See table 10.1 for more detail PWMIDX = 0xh: PWM0PRDH access PWMIDX = 1xh: PWM1PRDH access PWMIDX = 2xh: PWM2PRDH access PWMIDX = 3xh: PWM3PRDH access write sequence: PWMPRDL then PWMPRDH read sequence: PWMPRDH then PWMPRDL
DDh	**UART1CON**	6~0	UART1BRP	R/W	00H	UART1 baud rate pre-scaler
DEh	**UART0CON**	7	UART0BRS	R/W	00H	UART0 baud baud rate source select
		6~0	UART0BRP	R/W	00H	UART0 baud rate pre-scaler
DFh	**TKPINSB2**	4	TKPINSB2	R/W	0	TKB20 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKB19 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKB18 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKB17 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKB16 Pin fix as TK channel: 0: disable 1: enable
E0h	**ACC**	7~0	ACC	R/W	00h	Accumulator
E1h	**MICON**	7	MIEN	R/W	0	Master I^2C enable 0: disable 1: enable
		6	MIACKO	R/W	0	When Master I^2C receive data, send acknowledge to I^2C Bus 0: ACK to slave device 1: NACK to slave device
		5	MIIF	R/W	0	Master I^2C Interrupt flag 0: write 0 to clear it 1: Master I^2C transfer one byte complete
		4	MIACKI	R	–	When Master I^2C transfer, acknowledgement form I^2C bus (read only) 0: ACK received 1: NACK received
		3	MISTART	R/W	0	Master I^2C Start bit 1: start I^2C bus transfer
		2	MISTOP	R/W	1	Master I^2C Stop bit 1: send STOP signal to stop I^2C bus
		1~0	MICR	R/W	00	Master I^2C (SCL) clock frequency selection 00: Fsys/4 (ex. If Fsys=16MHz, I^2C clock is 4M Hz) 01: Fsys/16 (ex. If Fsys=16MHz, I^2C clock is 1M Hz) 10: Fsys/64 (ex. If Fsys=16MHz, I^2C clock is 250K Hz) 11: Fsys/256 (ex. If Fsys=16MHz, I^2C clock is 62.5K Hz)

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
E2h	**MIDAT**	7~0	MIDAT	R/W	00	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
E5h	**EFTCON**	7	EFT2CS	R/W	0	EFT2 Detector enable 0: Disable 1: Enable
		6	EFT1CS	R/W	0	EFT1 Detector enable 0: Disable 1: Enable
		5~4	EFT1S	R/W	0	EFT1 Detector sensitivity adjustment
		3	EFTSLOW	R/W	–	Force SYSCLK to SLOWCLK while EFT detected , Disable ONLY 0: Disable 1: Reserved
		2	EFTWCPU	R/W	0	CPU enter Wait state while EFT detected 0: Disable 1: Enable
		1	EFTWOUT	R/W	1	EFTWAIT output to pin 0: P03 = normal I/O 1: P03 = !EFTWAIT
		0	CKHLDE	R/W	00	clock hold enable 0: Disable 0: Enable
E6h	**EXA**	7~0	EXA	R/W	00h	Expansion accumulator
E7h	**EXB**	7~0	EXB	R/W	00h	Expansion B register
E8h	**P4**	7~0	P4	R/W	FFh	Port 4 data
E9h	**SIADR**	7~1	SA	R/W	64h	Slave I^2C address assigned
		0	SIEN	R/W	0	Slave I^2C enable 0: disable 1: enable
EAh	**SICON**	7	MIIE	R/W	0	I^2C Master interrupt enable 0: disable 1: enable
		6	TXDIE	R/W	0	Slave I^2C transmission completed interrupt enable 0: disable 1: enable
		5	RCD2IE	R/W	0	Slave I^2C DATA2(SITXRCD2) reception completed interrupt enable 0: disable 1: enable
		4	RCD1IE	R/W	0	Slave I^2C DATA1(SIRCD1) reception completed interrupt enable 0: disable 1: enable
		2	TXDF	R/W	1	Slave I^2C transmission completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I^2C transmission complete
		1	RCD2F	R/W	0	Slave I^2C DATA2(SITXRCD2) reception completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I^2C DATA2(SITXRCD2) reception complete enable
		0	RCD1F	R/W	0	Slave I^2C DATA1(SIRCD1) reception completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I^2C DATA1(SIRCD1) reception complete
EBh	**SIRCD1**	7~0	SIRCD1	R	–	Slave I^2C data receive register1 (DATA1)
ECh	**SITXRCD2**	7~0	SITXRCD2	R/W	–	Slave I^2C transmit and receive data register Read: Slave I^2C data receive register2 (DATA2) Write: Slave I^2C data transmission register (TXD)

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
EEF	**BOOTV**	2	RSTV	R/W	1	Change the reset vector, default 1 at power-on reset, other resets will not change user settings. 0: Reset vector = 0x0000 1: Reset vector = 0xE800 or 0xE000 (Determined by BOOTV)
		1~0	BOOTVR	R	–	Power on reset vector select. Read only. Load from CFGWH.BOOTV 00: 0x0000 01: 0x0000 10: 0xE000 (BOOT area 7K bytes) 11: 0xE800 (BOOT area 5K bytes, default)
EFh	**PWRCON**	4	AVPULL	R/W	0	Auto turn-on VPULL when SLOW to FAST 0: disable 1: enable
		3	WARMTIME	R/W	0	Warm up time after PDOWN 0: 128 Clock 1: 64 Clock
		2	ENVPULL	R/W	0	Power control, force VPULL enable 0: disable 1: enable
		1	PWRIDLE	R/W	0	Power control, VPULL control at IDLE mode 0: VDD = LDO @ IDLE mode 1: VDD = VPULL @ IDLE mode
		0	PWRSLOW	R/W	0	Power control, VPULL control at SLOW mode 0: VDD = LDO @ SLOW modle 1: VDD = VPULL @ SLOW mode
F0h	**B**	7~0	B	R/W	00h	B register
F1h	**CRCDL**	7~0	CRCDL	R/W	FFh	16-bit CRC data bit 7~0
F2h	**CRCDH**	7~0	CRCDH	R/W	FFh	16-bit CRC data bit 15~8
F3h	**CRCIN**	7~0	CRCIN	W	–	CRC input data
F5h	**CFGBG**	3~0	BGTRIM	R/W	–	VBG trimming value (Chip Reserved)
F6h	**CFGWL**	6~0	FRCF	R/W	–	FRC frequency adjustment 00h: lowest frequency 7Fh: highest frequency
F7h	**AUX2**	7~6	WDTE	R/W	–	Watchdog Timer Reset control 0x: WDT disable 10: WDT enable in Fast/Slow mode, disable in Idle/Stop/Halt mode 11: WDT always enable
		5	PWRSAV	R/W	–	Power saving mode control 0: No power saving 1: Power saving, disable LVR in IDLE/HALT/STOP mode
		4	VBGOUT	R/W	0	Bandgap voltage output control 0: P3.2 as normal I/O 1: Bandgap voltage output to P3.2 pin, The additional condition VBGEN=1 (AEh.1) should be set.
		3	DIV32	R/W	0	only active when MULDVI16 =1 0: instruction DIV as 16/16 bit division operation 1: instruction DIV as 32/16 bit division operation
		2~1	IAPTE	R/W	00	IAP watchdog timer enable 00: Disable 01: wait 0.8mS trigger watchdog time-out flag 10: wait 3.2mS trigger watchdog time-out flag 11: wait 6.4mS trigger watchdog time-out flag
		0	MULDIV16	R/W	0	0: instruction MUL/DIV as 8*8, 8/8 operation 1: instruction MUL/DIV as 16*16, 16/16 or 32/16 operation

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
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 Timer3, HW auto clear it at next clock cycle.
		5	TKSOCA	R/W	0	Touch Key A Start of Conversion Set 1 to start Touch Key conversion. If SYSCLK is fast enough, this bit will be cleared by H/W at the end of conversion while TKRERUN=0. S/W can also write 0 to clear this flag.
		4	ADSOC	R/W	0	ADC Start of Conversion Set 1 to start ADC conversion. Cleared by H/W at the end of conversion. S/W can also write 0 to clear this flag.
		2	TKSOCB	R/W	0	Touch Key B Start of Conversion Set 1 to start Touch Key conversion. If SYSCLK is fast enough, this bit will be cleared by H/W at the end of conversion while TKRERUN=0. S/W can also write 0 to clear this flag.
		1	T1SEL	R/W	0	Timer1 counter mode (CT1N=1) input select 0: P3.5 (T1) pin (8051 standard) 1: Slow clock divide by 16 (SLOWCLK/16)
		0	DPSEL	R/W	0	Active DPTR Select

Adr	Flash	Bit#	Bit Name	Description
INFO2 0200h	CFGWH	7	PROTN	Flash Code Protect, 0=Protect
		6	XRSTEN	External Pin Reset enable, 0=enable.
		0	BOOTV	Power-on Reset Vector Selection 00: 0x0000 01: 0x0000 10: 0xE000 (BOOT area 7K bytes) 11: 0xE800 (BOOT area 5K bytes, default)

INSTRUCTION SET

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

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

LOGICAL				
Mnemonic	**Description**	**byte**	**cycle**	**opcode**
ANL A,Rn	AND register to A	1	2	58-5F
ANL A,dir	AND direct byte to A	2	2	55
ANL A,@Ri	AND indirect memory to A	1	2	56-57
ANL A,#data	AND immediate to A	2	2	54
ANL dir,A	AND A to direct byte	2	2	52
ANL dir,#data	AND immediate to direct byte	3	4	53
ORL A,Rn	OR register to A	1	2	48-4F
ORL A,dir	OR direct byte to A	2	2	45
ORL A,@Ri	OR indirect memory to A	1	2	46-47
ORL A,#data	OR immediate to A	2	2	44
ORL dir,A	OR A to direct byte	2	2	42
ORL dir,#data	OR immediate to direct byte	3	4	43
XRL A,Rn	Exclusive-OR register to A	1	2	68-6F
XRL A,dir	Exclusive-OR direct byte to A	2	2	65
XRL A, @Ri	Exclusive-OR indirect memory to A	1	2	66-67
XRL A,#data	Exclusive-OR immediate to A	2	2	64
XRL dir,A	Exclusive-OR A to direct byte	2	2	62
XRL dir,#data	Exclusive-OR immediate to direct byte	3	4	63
CLR A	Clear A	1	2	E4
CPL A	Complement A	1	2	F4
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	4	93
MOVC A,@A+PC	Move code byte relative PC to A	1	4	83
MOVX A,@Ri	Move external data(A8) to A	1	4	E2-E3
MOVX A,@DPTR	Move external data(A16) to A	1	4	E0
MOVX @Ri,A	Move A to external data(A8)	1	4	F2-F3
MOVX @DPTR,A	Move A to external data(A16)	1	4	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	4	11-F1
LCALL addr 16	Long jump to subroutine	3	4	12
RET	Return from subroutine	1	4	22
RETI	Return from interrupt	1	4	32
AJMP addr 11	Absolute jump unconditional	2	4	01-E1
LJMP addr 16	Long jump unconditional	3	4	02
SJMP rel	Short jump (relative address)	2	4	80
JC rel	Jump on carry = 1	2	4	40
JNC rel	Jump on carry = 0	2	4	50
JB bit,rel	Jump on direct bit = 1	3	4	20
JNB bit,rel	Jump on direct bit = 0	3	4	30
JBC bit,rel	Jump on direct bit = 1 and clear	3	4	10
JMP @A+DPTR	Jump indirect relative DPTR	1	4	73
JZ rel	Jump on accumulator = 0	2	4	60
JNZ rel	Jump on accumulator≠0	2	4	70
CJNE A,dir,rel	Compare A,direct, jump not equal relative	3	4	B5
CJNE A,#data,rel	Compare A,immediate, jump not equal relative	3	4	B4
CJNE Rn,#data,rel	Compare register,immediate, jump not equal relative	3	4	B8-BF
CJNE @Ri,#data,rel	Compare indirect,immediate, jump not equal relative	3	4	B6-B7
DJNZ Rn,rel	Decrement register, jump not zero relative	2	4	D8-DF
DJNZ dir,rel	Decrement direct byte, jump not zero relative	3	4	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	°C
Storage temperature	–65 ~ +150	

2. DC Characteristics (T_A=25 ℃, V_{CC}=2.3V ~ 5.5V)

Parameter	Symbol	Conditions		Min	Typ	Max	Unit
Operating Voltage	V_{CC}	F_{SYSCLK}=18.432 MHz		2.3	–	5.5	V
Input High Voltage	V_{IH}	All Input	V_{CC}=5V	0.6V_{CC}	–	–	V
			V_{CC}=3V	0.6V_{CC}	–	–	V
Input Low Voltage	V_{IL}	All Input	V_{CC}=5V	–	–	0.2V_{CC}	V
			V_{CC}=3V	–	–	0.2V_{CC}	V
I/O Port Source Current	I_{OH}	All Output	V_{CC}=5V, V_{OH}=0.9V_{CC}	5.5	11	–	mA
			V_{CC}=5V, V_{OH}=0.6V_{CC}	20	34		
			V_{CC}=3V, V_{OH}=0.9V_{CC}	2.5	5	–	
			V_{CC}=3V, V_{OH}=0.6V_{CC}	8	14		
I/O Port Sink Current	I_{OL}	All Output,	V_{CC}=5V, V_{OL}=0.1V_{CC}	40	65	–	mA
			V_{CC}=3V, V_{OL}=0.1V_{CC}	20	28	–	
		P1 @ LED mode (High sink)	V_{CC}=5V, V_{OL}=0.1V_{CC}	100	130	–	
			V_{CC}=3V, V_{OL}=0.1V_{CC}	40	58	–	
Supply Current	I_{DD}	FAST mode V_{CC}=5V	FRC=18.432 MHz	–	3.5	–	mA
		FAST mode V_{CC}=3V	FRC=18.432 MHz	–	3.5	–	
		SLOW mode	V_{CC}=3V	–	0.21	–	
			V_{CC}=5V	–	0.2	–	
		IDLE mode PWRSAV=0	SRC, V_{CC}=5V	–	180	–	μA
			SRC, V_{CC}=3V	–	165	–	
		IDLE mode PWRSAV=1	V_{CC}=5V	–	165	–	
			V_{CC}=3V	–	150	–	
		STOP mode PWRSAV=0	V_{CC}=5V	–	55	–	
			V_{CC}=3V	–	45	–	
		STOP mode PWRSAV=1	V_{CC}=5V	–	11	–	
			V_{CC}=3V	–	3.8	–	
		HATL mode PWRSAV=0	V_{CC}=5V	–	58	–	
			V_{CC}=3V	–	47	–	
		HATL mode PWRSAV=1	V_{CC}=5V	–	13.5	–	
			V_{CC}=3V	–	5.5	–	
System Clock Frequency	F_{SYSCLK}	V_{CC} >LVR_{TH}	V_{CC}=2.5V	–	–	18.432	MHz

Parameter	Symbol	Conditions		Min	Typ	Max	Unit
LVR Reference Voltage	V_{LVR}	T_A=25°C		–	2.24	–	V
				–	2.48	–	
				–	2.72	–	
				–	2.96	–	
				–	3.20	–	
				–	3.44	–	
				–	3.68	–	
				–	3.92	–	
LVR Hysteresis Voltage	V_{HYST}	T_A=25°C		–	±0.1	–	V
LVD Reference Voltage	V_{LVD}	T_A=25°C		–	2.52	–	V
				–	2.62	–	
				–	2.74	–	
				–	2.86	–	
				–	2.99	–	
				–	3.1	–	
				–	3.23	–	
				–	3.35	–	
				–	3.48	–	
				–	3.6	–	
				–	3.72	–	
				–	3.84	–	
				–	3.96	–	
				–	4.08	–	
				–	4.2	–	
				–	4.32	–	
Low Voltage Detection time	t_{LVR}	T_A=25°C		100	–	–	μs
Pull-Up Resistor	R_P	V_{IN}=0V	V_{CC}=5V	–	34	–	KΩ
			V_{CC}=3V		58		

3. Clock Timing (T_A= –40°C ~ +105°C)

Parameter	Condition	Min	Typ	Max	Unit
FRC Frequency	-20℃ ~ 50℃, V_{CC}=2.5V~5.0V	–1%	18.432	+1%	MHz
	-40℃ ~ 70℃, V_{CC}=2.5V~5.0V	–1.5%	18.432	+1.5%	
	-40℃ ~ 85℃, V_{CC}=2.5 ~ 5.0V	–2%	18.432	+2%	
	-40℃ ~ 105℃, V_{CC}=2.5 ~ 5.0V	–3%	18.432	+3%	

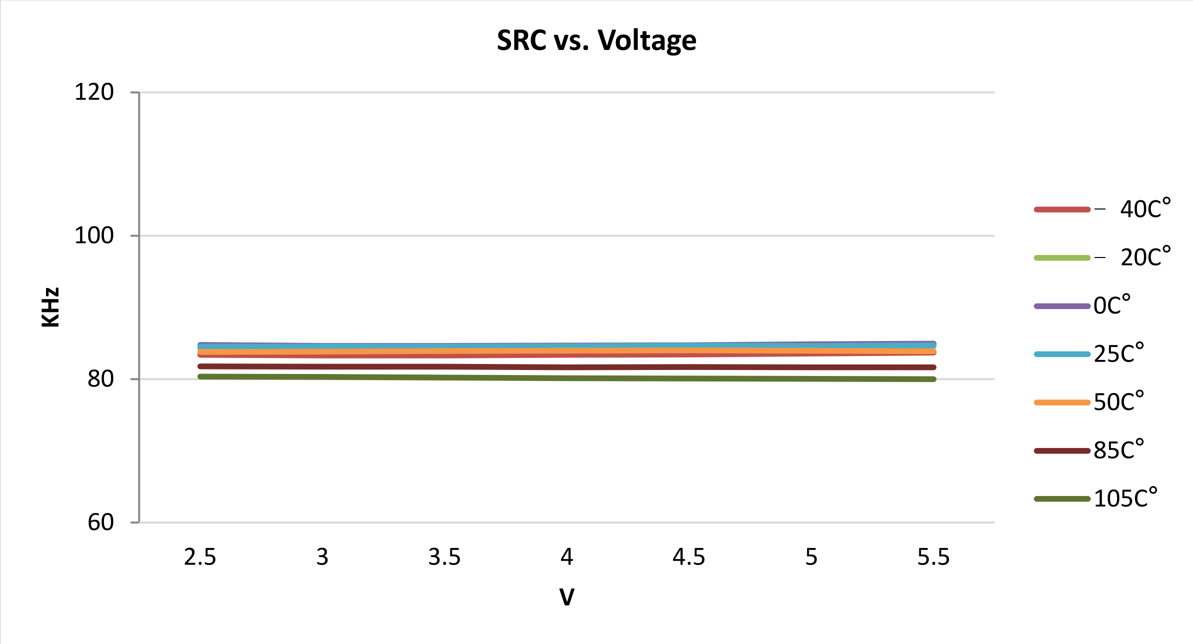
4. Reset Timing Characteristics (T_A= –40°C ~ +105°C)

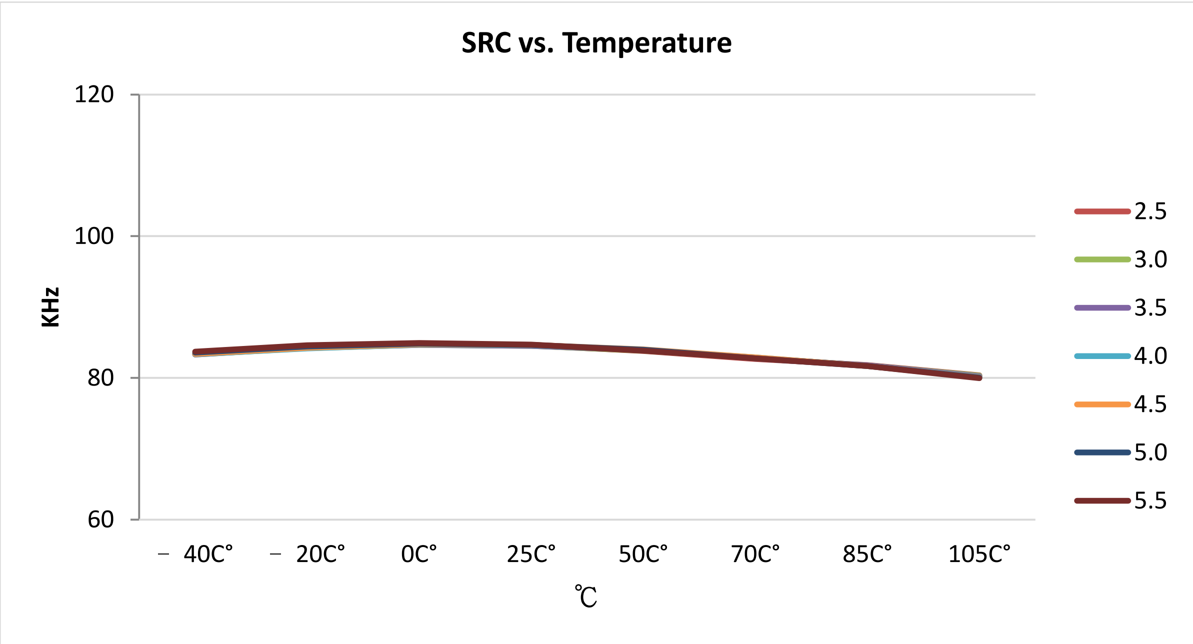
Parameter	Conditions	Min	Typ	Max	Unit
RESET Input Low width	Input V_{CC}=5V ± 10 %	30	–	–	μs
WDT wakeup time	V_{CC}=5V, WDTPSC=11	–	55	–	ms
	V_{CC}=3V, WDTPSC=11	–	57	–	

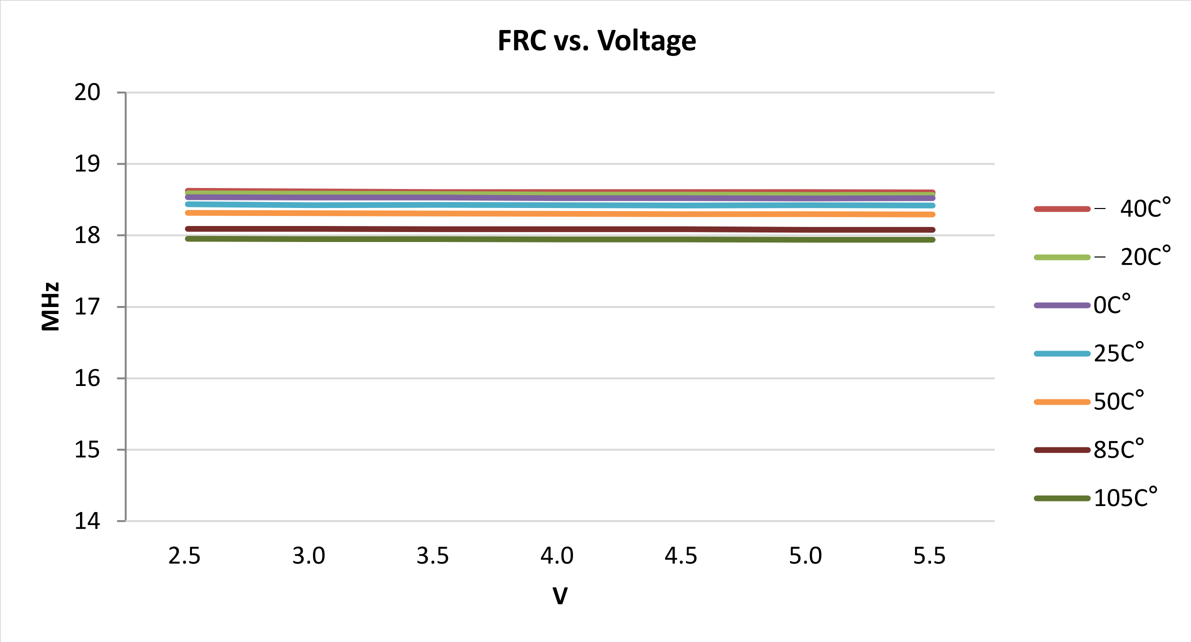
5. ADC Electrical Characteristics (T_A= 25°C, V_{CC}= 3.0V ~ 5.5V, V_{SS}= 0V)

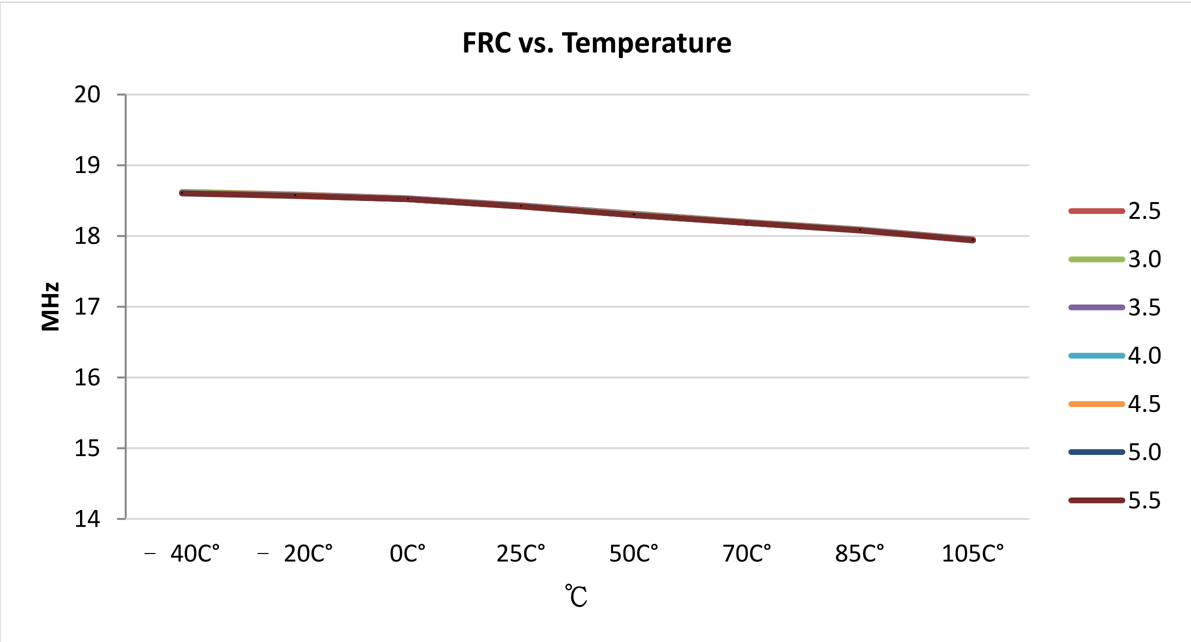
Parameter	Conditions		Min	Typ	Max	Unit
Total Accuracy	V_{CC}=5.12 V, V_{SS}=0V		–	±2.5	±4	LSB
Integral Non-Linearity			–	±3.2	±5	
Max Input Clock (f_{ADC})	Source impedance (Rs < 10K omh)		–	–	2	MHz
	Source impedance (Rs < 20K omh)		–	–	1	
	Source impedance (Rs < 50K omh)		–	–	0.5	
	Source is VBG (ADCHS=1011b)		–	–	0.5	
Conversion Time	F_{ADC} = 1MHz		–	50	–	μs
Bandgap Reference Voltage (V_{BG})	–	V_{CC}=2.5V~5.5V -20°C ~85°C	-1.5%	1.22	+1.5%	V
		V_{CC}=2.5V~5.5V -40°C ~105°C	-2%	1.22	+2%	
ADC Reference Voltage (V_{ADC})	ADCVREFS=1	V_{CC}=3V~5V -20°C ~85°C	-1.5%	2.5	+1.5%	
		V_{CC}=3V~5V -40°C ~105°C	-2%	2.5	+2%	
V_{CC}/4 Reference Voltage ($V_{1/4}$)	–	V_{CC}=5V, 25°C	-0.8%	1.26	+0.8%	
		V_{CC}=3.6V, 25°C	-0.8%	0.907	+0.8%	
Input Voltage	–		V_{SS}	–	V_{CC}	

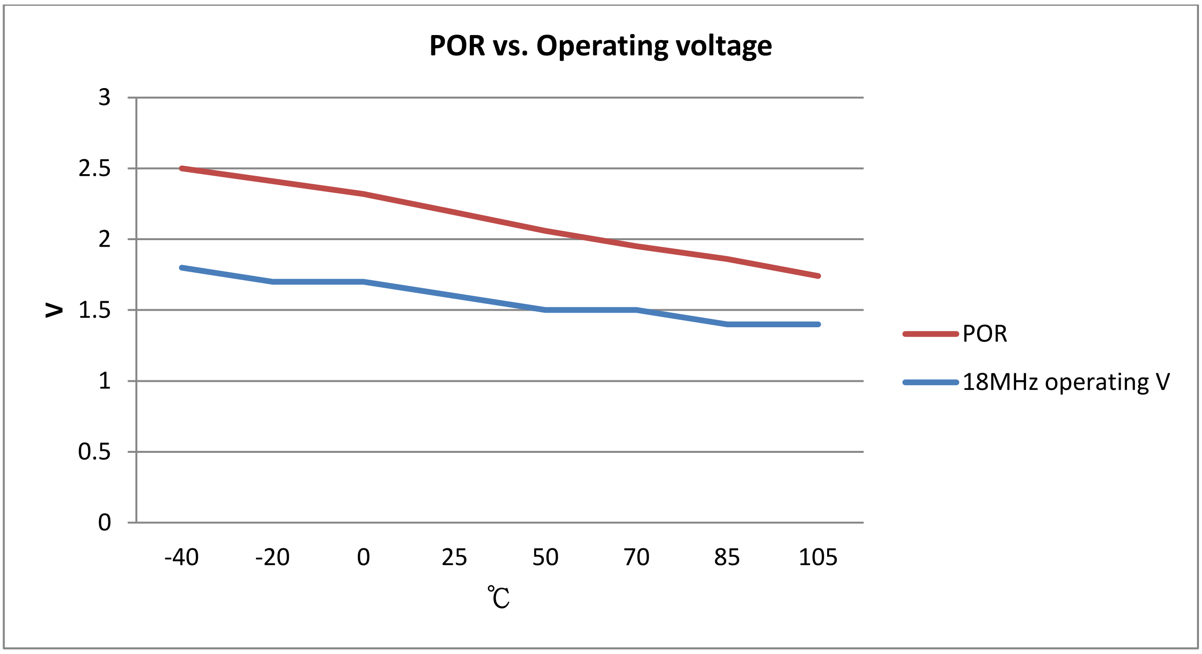
6. Characteristic Graphs

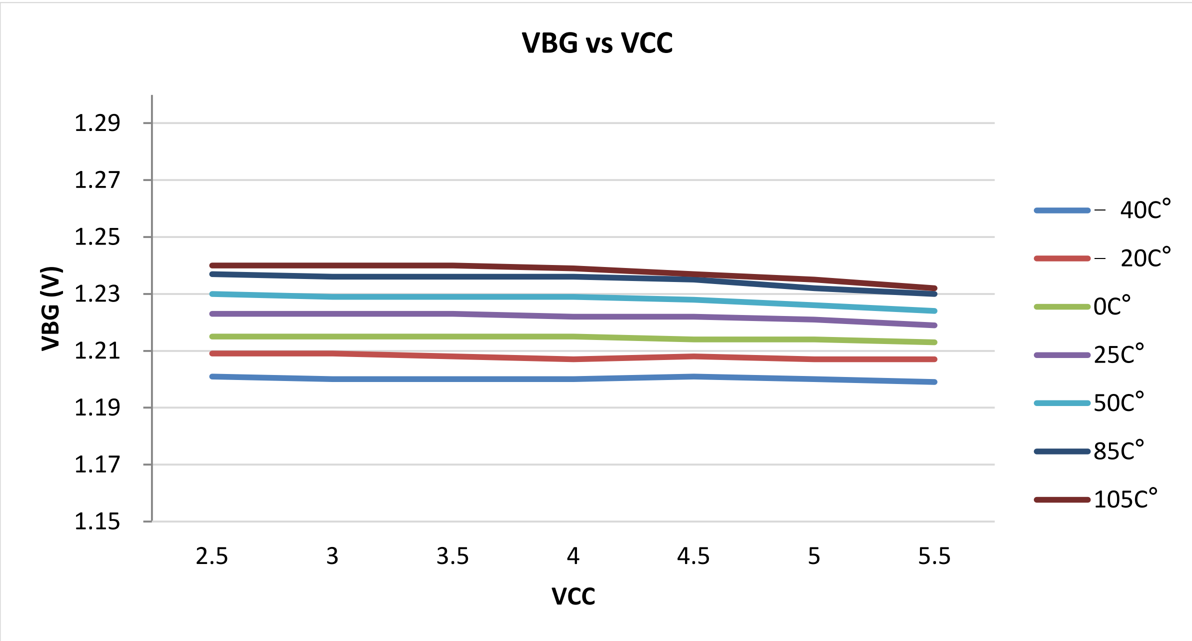





FRC vs. Voltage
20
19
18
17
16
15
14
MHz
2.5
3.0
3.5
4.0
4.5
5.0
5.5
V
－ 40C°
－ 20C°
0C°
25C°
50C°
85C°
105C°


FRC vs. Temperature
20
19
18
17
16
15
14
MHz
－ 40C°
－ 20C°
0C°
25C°
50C°
70C°
85C°
105C°
℃
2.5
3.0
3.5
4.0
4.5
5.0
5.5


POR vs. Operating voltage
3
2.5
2
1.5
1
0.5
0
V
-40
-20
0
25
50
70
85
105
℃
POR
18MHz operating V


VBG vs VCC
1.29
1.27
1.25
1.23
1.21
1.19
1.17
1.15
VBG (V)
2.5
3
3.5
4
4.5
5
5.5
VCC
– 40C°
– 20C°
0C°
25C°
50C°
85C°
105C°

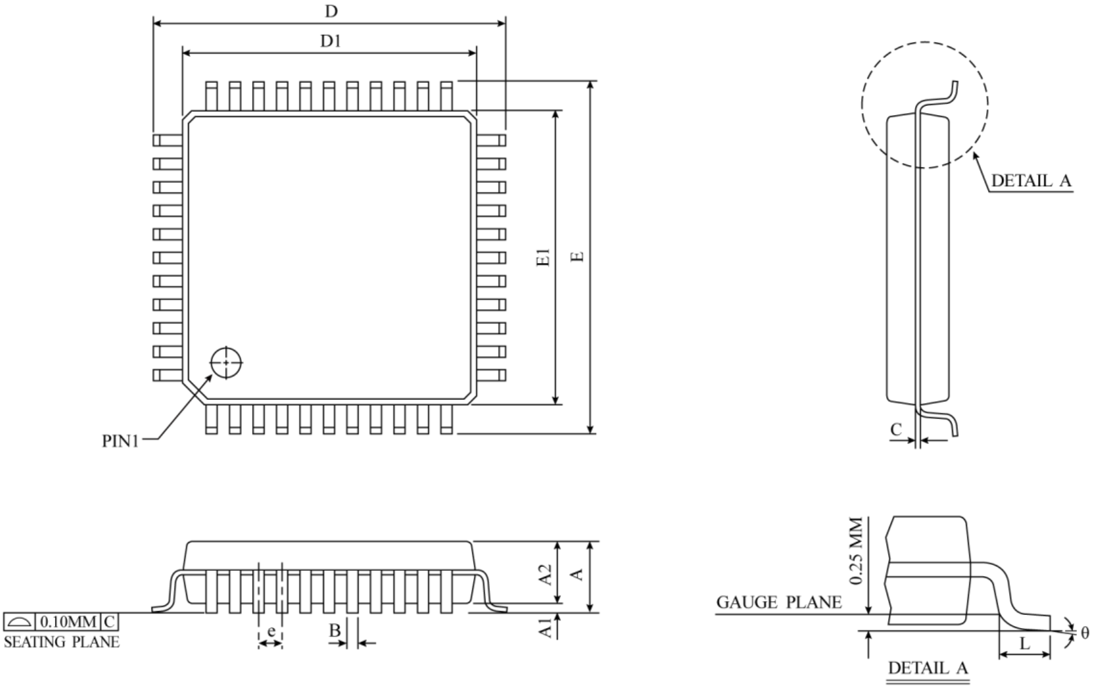
Package and Dice Information

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
TM52eF0C86-MTP	Wafer/Dice blank chip
TM52eF0C85-MTP	
TM52eF0C86-COD	Wafer/Dice with code
TM52eF0C85-COD	
TM52eF0C86-MTP-74	LQFP 44-pin (10*10*1.4mm)
TM52eF0C85-MTP-74	
TM52eF0C86-MTP-23	SOP 28-pin (300 mil)
TM52eF0C85-MTP-23	

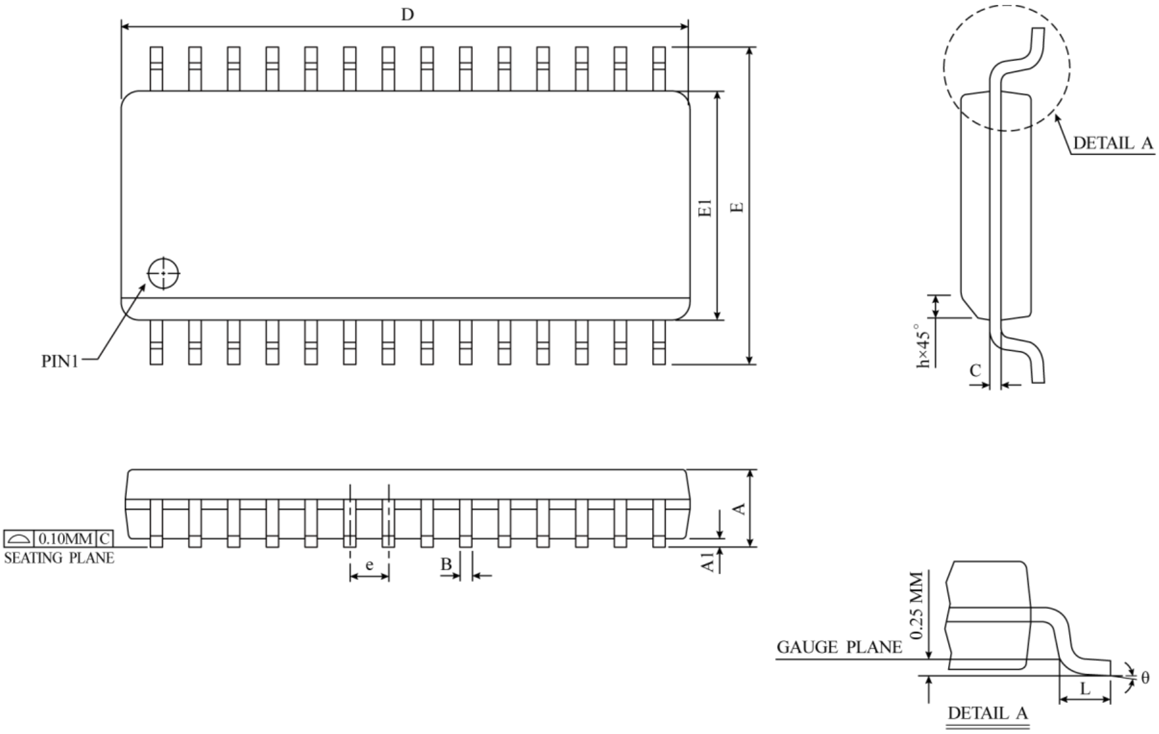
LQFP-44 (10×10mm) Package Dimension



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	0.002	0.004	0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
B	0.30	0.35	0.40	0.012	0.012	0.016
C	0.09	0.13	0.16	0.004	0.006	0.008
D	12.00 BSC			0.472 BSC		
D1	10.00 BSC			0.394 BSC		
E	12.00 BSC			0.472 BSC		
E1	10.00 BSC			0.394 BSC		
e	0.80 BSC			0.031 BSC		
L	0.45	0.60	0.75	0.018	0.024	0.030
θ	0°	3.5°	7°	0°	3.5°	7°
JEDEC	MS-026 (BCB)					

⚠ * 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.

SOP-28 (300mil) Package Dimension



SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.35	2.50	2.65	0.0926	0.0985	0.1043
A1	0.10	0.20	0.30	0.0040	0.0079	0.0118
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.23	0.28	0.32	0.0091	0.0108	0.0125
D	17.70	17.90	18.10	0.6969	0.7047	0.7125
E	10.00	10.33	10.65	0.3940	0.4425	0.4910
E1	7.40	7.50	7.60	0.2914	0.2953	0.2992
e	1.27 BSC			0.050 BSC		
h	0.25	0.50	0.75	0.0100	0.0195	0.0290
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
θ	0°	4°	8°	0°	4°	8°
JEDEC	MS-013 (AE)					

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