Product List 3
Description 3
Ordering Information 3
Features 3
Pin Configuration 4
Block Diagram 7
Special Function Register (SFR) 9
Function Description 12
1. General Features 12
  1.1 Embedded Flash 12
  1.2 IO Pads 12
  1.3 System Control Register (SCONF) 12
2. Instruction Set 13
3. Memory Structure 17
  3.1 Program Memory 17
  3.2 Data Memory 18
  3.3 Data memory - lower 128 byte (00h to 7Fh) 18
  3.4 Data memory - higher 128 byte (80h to FFh) 18
  3.5 Data memory - Expanded 768 bytes ($0000 to $02FF) 18
4. CPU Engine 22
  4.1 Accumulator 22
  4.2 B Register 22
  4.3 Program Status Word 23
  4.4 Stack Pointer 23
  4.5 Data Pointer 23
5. GPIO 24
6. Timer 0 and Timer 1 25
  6.1 Timer/counter mode control register (TMOD) 25
  6.2 Timer/counter control register (TCON) 26
  6.3 Mode 0 (13-bit Counter/Timer) 26
  6.4 Mode 1 (16-bit Counter/Timer) 27
  6.5 Mode 2 (8-bit auto-reload Counter/Timer) 27
  6.6 Mode 3 (Timer 0 acts as two independent 8 bit Timers / Counters) 27
7. Timer 2 29
  7.1 Capture mode 30
  7.2 Auto-reload (Up or Down Counter) 31
  7.3 Programmable clock out 32
8. Serial interface – UART 34
  8.1 Serial interface 34
    8.1.1 Mode 0 35
    8.1.2 Mode 1 35
    8.1.3 Mode 2 36
    8.1.4 Mode 3 36
  8.2 Multiprocessor Communication of Serial Interface 36
  8.3 Baud Rate Generator 37
    8.3.1 Serial interface Mode 0 37
    8.3.2 Serial interface Mode 2 37
    8.3.3 Serial interface Mode 1 and 3 37
9. Interrupt 38
10. Watch Dog Timer 40
11. Power Management Unit 42
  11.1 Idle mode 42
  11.2 Power Down mode 42
12. Pulse Width Modulation (PWM) 44
13. Two-Wire Series Interface (TWSI) 46
14. In-System Programming (Internal ISP) 49

14.1 ISP service program 49
14.2 Lock Bit (N) 49
14.3 Program the ISP Service Program 50
14.4 Initiate ISP Service Program 50
14.5 ISP register – ISPFAH, ISPFAL, ISPFD and ISPC 51
Operating Conditions 53
DC Characteristics 53

Product List

SM5964BW40PP,
SM5964BW44JP,
SM5964BW44QP,

Description

The SM5964B series product is an 8 - bit single chip micro controller with 64KB flash & 1KB SRAM embedded. It has In-System Programming (ISP) function and is a derivative of the 8052 micro controller family.

SM5964B is a versatile and cost effective controller for those applications which demand up to 36 I/O pins, or applications which need up to 64K byte flash memory either for program or for data or mixed.

To program the on-chip flash memory, a commercial writer is available to do it in parallel programming method. The on-chip flash memory can be programmed in either parallel or serial interface with its ISP feature.

Ordering Information

SM5964BihhkL YWW
i: process identifier { W = 3V ~ 5.5V}
hh: pin count
k: package type postfix {as table below }
L:PB Free identifier
{No text is Non-PB free, ”P” is PB free}
Y: Year Code
WW: Week Code (01-52)

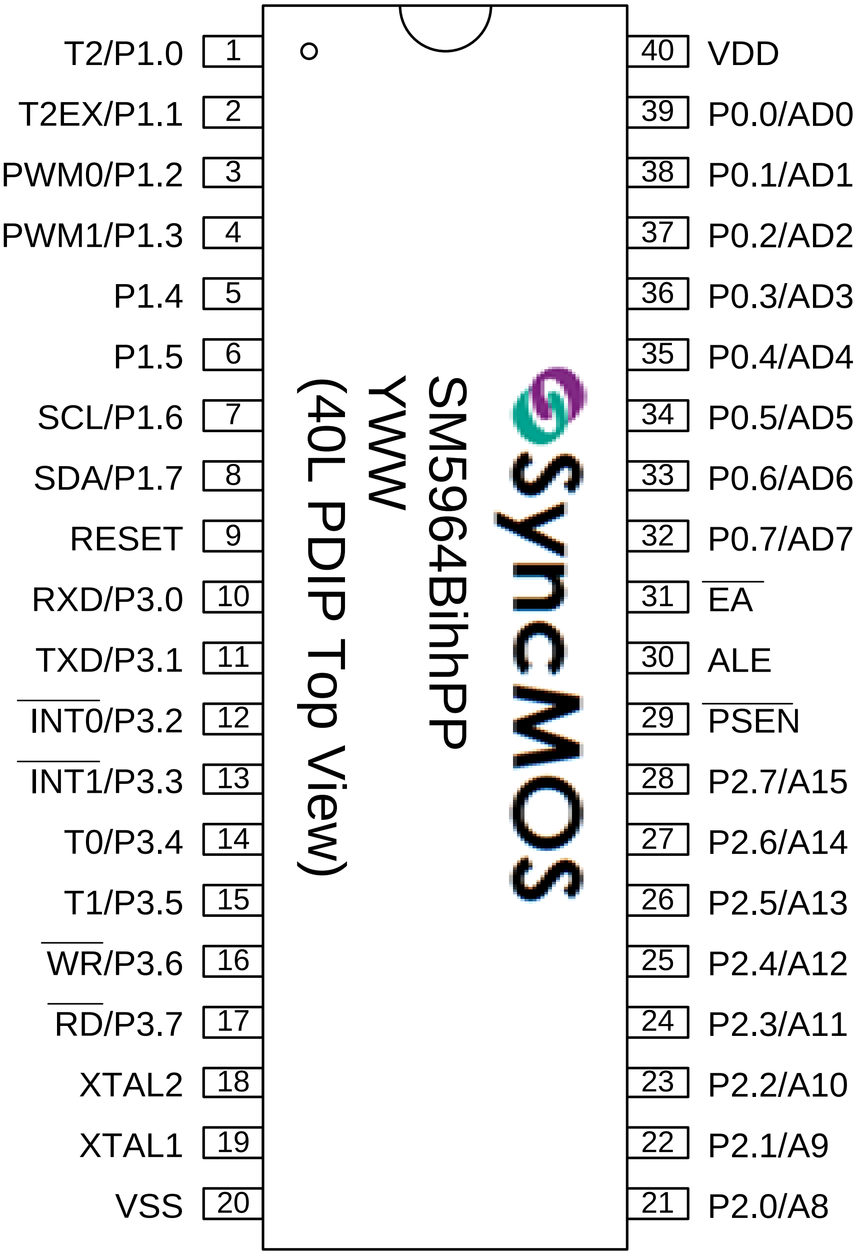
Postfix	Package
P	PDIP
J	PLCC
Q	PQFP

Features

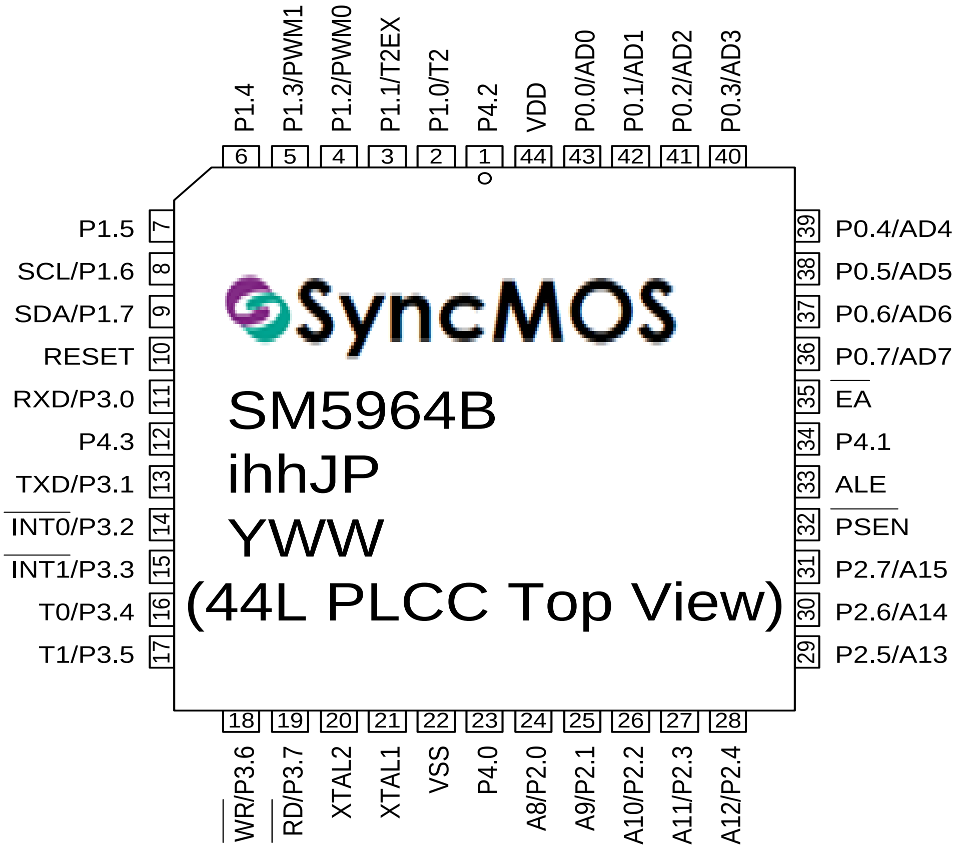
- Main Flash ROM 64KB.
- Working voltage 3V~5.5V runs up to 40MHz
- General 8052 family compatible with 12 clocks in one machine cycle.
- 6 clocks in one machine cycle is also supported.
- 256 bytes SRAM as standard 8052.
- On-chip 768 bytes expanded RAM.
- 16-bit Data Pointers (DPTR).
- One serial peripheral interfaces in full duplex mode (UART).
  - Synchronous mode, fixed baud rate.
  - 8-bit UART mode, variable baud rate.
  - 9-bit UART mode, fixed baud rate.
  - 9-bit UART mode, variable baud rate.
- Three 16-bit Timer/Counters (Timer 0, 1, 2).
- One watch dog timer (WDT).
- One IIC interface (Master / Slave mode).
- Two 8-bit/5-bit configurable PWM output channels.
- ISP/IAP functions.
- ISP service program space configurable in N*512 byte (N=0 to 8) size.
- Seven interrupt sources with two priority levels.
- Four 8-bit I/O ports and additional one 4-bit I/O ports.
- IO PAD ESD over 4KV.
- Enhance user code protection.
- Power management unit for IDLE and power down modes.

Pin Configuration

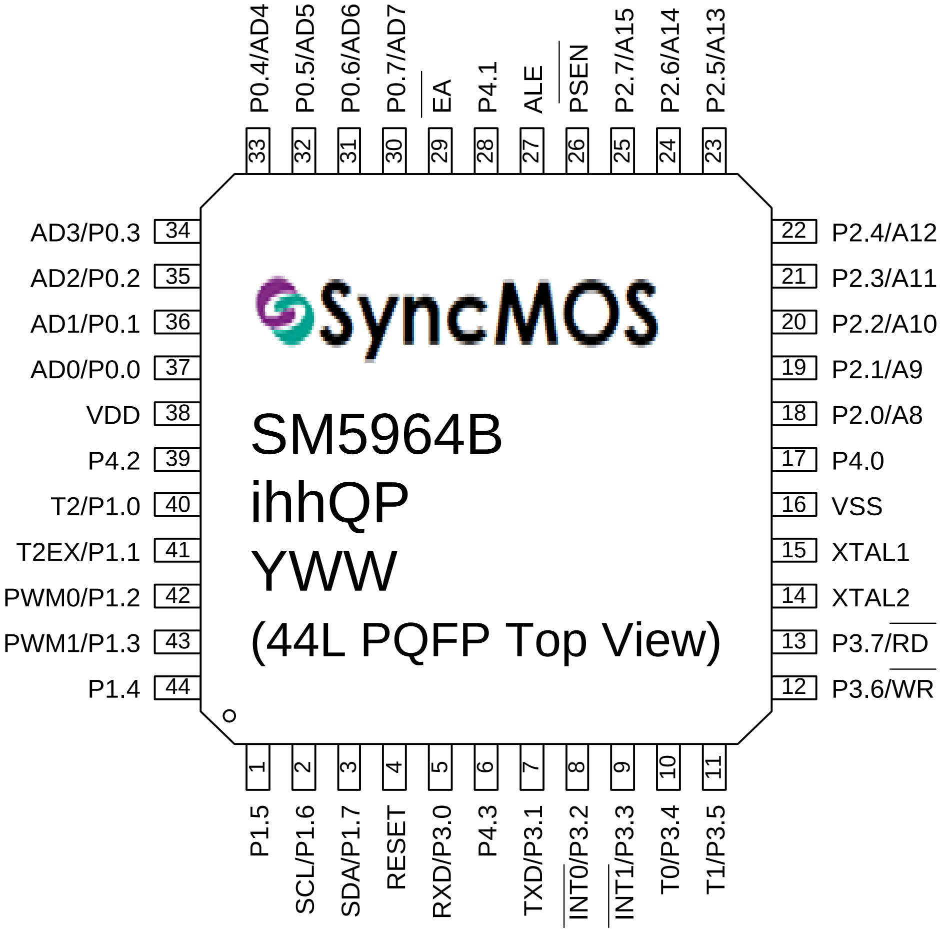
40 Pin PDIP


T2/P1.0 1
T2EX/P1.1 2
PWM0/P1.2 3
PWM1/P1.3 4
P1.4 5
P1.5 6
SCL/P1.6 7
SDA/P1.7 8
RESET 9
RXD/P3.0 10
TXD/P3.1 11
INT0/P3.2 12
INT1/P3.3 13
T0/P3.4 14
T1/P3.5 15
WR/P3.6 16
RD/P3.7 17
XTAL2 18
XTAL1 19
VSS 20
40 VDD
39 P0.0/AD0
38 P0.1/AD1
37 P0.2/AD2
36 P0.3/AD3
35 P0.4/AD4
34 P0.5/AD5
33 P0.6/AD6
32 P0.7/AD7
31 EA
30 ALE
29 PSEN
28 P2.7/A15
27 P2.6/A14
26 P2.5/A13
25 P2.4/A12
24 P2.3/A11
23 P2.2/A10
22 P2.1/A9
21 P2.0/A8
SyncMOS
SM5964BihhPP
YWW
(40L PDIP Top View)

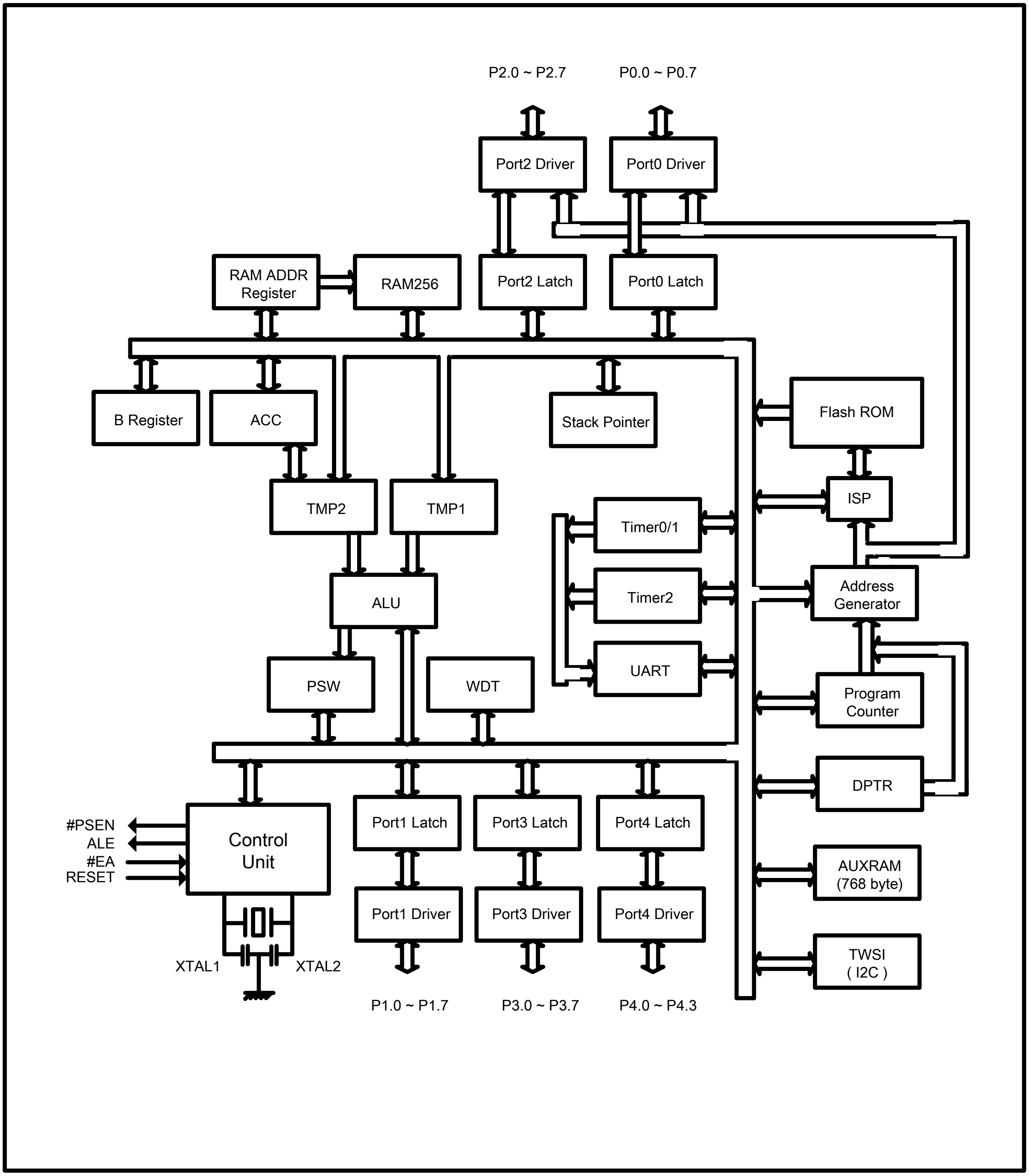
44 Pin PLCC


P1.4
P1.3/PWM1
P1.2/PWM0
P1.1/T2EX
P1.0/T2
P4.2
VDD
P0.0/AD0
P0.1/AD1
P0.2/AD2
P0.3/AD3
6
5
4
3
2
1
44
43
42
41
40
P1.5
SCL/P1.6
SDA/P1.7
RESET
RXD/P3.0
P4.3
TXD/P3.1
INT0/P3.2
INT1/P3.3
T0/P3.4
T1/P3.5
7
8
9
10
11
12
13
14
15
16
17
SyncMOS
SM5964B
ihhJP
YWW
(44L PLCC Top View)
39
38
37
36
35
34
33
32
31
30
29
P0.4/AD4
P0.5/AD5
P0.6/AD6
P0.7/AD7
EA
P4.1
ALE
PSEN
P2.7/A15
P2.6/A14
P2.5/A13
18
19
20
21
22
23
24
25
26
27
28
WR/P3.6
RD/P3.7
XTAL2
XTAL1
VSS
P4.0
A8/P2.0
A9/P2.1
A10/P2.2
A11/P2.3
A12/P2.4

44 Pin PQFP


P0.4/AD4 33
P0.5/AD5 32
P0.6/AD6 31
P0.7/AD7 30
EA 29
P4.1 28
ALE 27
PSEN 26
P2.7/A15 25
P2.6/A14 24
P2.5/A13 23
AD3/P0.3 34
AD2/P0.2 35
AD1/P0.1 36
AD0/P0.0 37
VDD 38
P4.2 39
T2/P1.0 40
T2EX/P1.1 41
PWM0/P1.2 42
PWM1/P1.3 43
P1.4 44
22 P2.4/A12
21 P2.3/A11
20 P2.2/A10
19 P2.1/A9
18 P2.0/A8
17 P4.0
16 VSS
15 XTAL1
14 XTAL2
13 P3.7/RD
12 P3.6/WR
SyncMOS
SM5964B
ihhQP
YWW
(44L PQFP Top View)
1 P1.5
2 SCL/P1.6
3 SDA/P1.7
4 RESET
5 RXD/P3.0
6 P4.3
7 TXD/P3.1
8 INT0/P3.2
9 INT1/P3.3
10 T0/P3.4
11 T1/P3.5

Block Diagram


P2.0 ~ P2.7
P0.0 ~ P0.7
Port2 Driver
Port0 Driver
RAM ADDR Register
RAM256
Port2 Latch
Port0 Latch
B Register
ACC
Stack Pointer
Flash ROM
TMP2
TMP1
ISP
Timer0/1
Timer2
Address Generator
ALU
UART
PSW
WDT
Program Counter
DPTR
#PSEN
ALE
#EA
RESET
Control Unit
Port1 Latch
Port3 Latch
Port4 Latch
AUXRAM (768 byte)
Port1 Driver
Port3 Driver
Port4 Driver
TWSI (I2C)
XTAL1
XTAL2
P1.0 ~ P1.7
P3.0 ~ P3.7
P4.0 ~ P4.3

Pin Description

40L PDIP	44L PLCC	44L PQFP	Symbol	I/O	Description
	1	39	P4.2	I/O	Bit 2 of port 4
1	2	40	P1.0/T2	I/O	Bit 0 of port 1 & Timer 2 external input clock
2	3	41	P1.1/T2EX	I/O	Bit 1 of port 1 & Timer 2 capture trigger
3	4	42	P1.2/PWM0	I/O	Bit 2 of port 1 & PWM Channel 0
4	5	43	P1.3/PWM1	I/O	Bit 3 of port 1 & PWM Channel 1
5	6	44	P1.4	I/O	Bit 4 of port 1
6	7	1	P1.5	I/O	Bit 5 of port 1
7	8	2	P1.6/SCL	I/O	Bit 6 of port 1 & IIC SCL pin
8	9	3	P1.7/SDA	I/O	Bit 7 of port 1 & IIC SDA pin
9	10	4	RESET	I	Reset pin
10	11	5	P3.0/RXD	I/O	Bit 0 of port 3 & Serial interface channel receive data
	12	6	P4.3	I/O	Bit 3 of port 4
11	13	7	P3.1/TXD	I/O	Bit 1 of port 3 & Serial interface channel Transmit data
12	14	8	P3.2/INT0	I/O	Bit 2 of port 3 & Interrupt 0
13	15	9	P3.3/INT1	I/O	Bit 3 of port 3 & Interrupt 1
14	16	10	P3.4/T0	I/O	Bit 4 of port 3 & Timer 0 external input
15	17	11	P3.5/T1	I/O	Bit 5 of port 3 & Timer 1 external input
16	18	12	P3.6/WR	I/O	Bit 6 of port 3 & external memory write
17	19	13	P3.7/RD	I/O	Bit 7 of port 3 & external memory read
18	20	14	XTAL2	I/O	Crystal output
19	21	15	XTAL1	I/O	Crystal input
20	22	16	VSS	I	Ground
	23	17	P4.0	I/O	Bit 0 of port 4
21	24	18	P2.0/A8	I/O	Bit 0 of port 2 & bit 8 of external memory address
22	25	19	P2.1/A9	I/O	Bit 1 of port 2 & bit 9 of external memory address
23	26	20	P2.2/A10	I/O	Bit 2 of port 2 & bit 10 of external memory address
24	27	21	P2.3/A11	I/O	Bit 3 of port 2 & bit 11 of external memory address
25	28	22	P2.4/A12	I/O	Bit 4 of port 2 & bit 12 of external memory address
26	29	23	P2.5/A13	I/O	Bit 5 of port 2 & bit 13 of external memory address
27	30	24	P2.6/A14	I/O	Bit 6 of port 2 & bit 14 of external memory address
28	31	25	P2.7/A15	I/O	Bit 7 of port 2 & bit 15 of external memory address
29	32	26	PSEN	O	program storage enable
30	33	27	ALE	O	address latch enable
	34	28	P4.1	I/O	Bit 1 of port 4
31	35	29	EA	I	external access
32	36	30	P0.7/AD7	I/O	Bit 7 of port 0 & data/address bit 7 of external memory
33	37	31	P0.6/AD6	I/O	Bit 6 of port 0 & data/address bit 6 of external memory
34	38	32	P0.5/AD5	I/O	Bit 5 of port 0 & data/address bit 5 of external memory
35	39	33	P0.4/AD4	I/O	Bit 4 of port 0 & data/address bit 4 of external memory
36	40	34	P0.3/AD3	I/O	Bit 3 of port 0 & data/address bit 3 of external memory
37	41	35	P0.2/AD2	I/O	Bit 2 of port 0 & data/address bit 2 of external memory
38	42	36	P0.1/AD1	I/O	Bit 1 of port 0 & data/address bit 1 of external memory
39	43	37	P0.0/AD0	I/O	Bit 0 of port 0 & data/address bit 0 of external memory
40	44	38	VDD	I	Power supply

Special Function Register (SFR)

A map of the Special Function Registers is shown as below:

Hex\Bin	X000	X001	X010	X011	X100	X101	X110	X111	Bin/Hex
F8									FF
F0	B				ISPFAH	ISPFAL	ISPFD	ISPC	F7
E8									EF
E0	ACC								E7
D8	P4								DF
D0	PSW			PWMC0	PWMC1				D7
C8	T2CON	T2MOD	RCAP2L	RCAP2H	TL2	TH2			CF
C0	TWSIS	TWSIA	TWSIC1	TWSIC2	TWSITXD	TWSIRXD			C7
B8	IP	IP1						SCONF	BF
B0	P3			PWDD0	PWMD1				B7
A8	IE	IE1	IFR						AF
A0	P2								A7
98	SCON	SBUF		P1CON				WDTC	9F
90	P1								97
88	TCON	TMOD	TL0	TL1	TH0	TH1			8F
80	P0	SP	DPL	DPH		RCON	DBANK	PCON	87
Hex\Bin	X000	X001	X010	X011	X100	X101	X110	X111	Bin/Hex

Note: Special Function Registers reset values and description for SM5964B.

	Register	Location	Reset value	Description
1	P0	80H	FFH	Port 0
2	SP	81H	07H	Stack Pointer
3	DPL	82H	00H	Data Pointer 0 low byte
4	DPH	83H	00H	Data Pointer 0 high byte
5	RCON	85H	00H	RAM Control
6	DBANK	86H	01H	Data RAM bank select
7	PCON	87H	00H	Power Control
8	TCON	88H	00H	Timer/Counter Control
9	TMOD	89H	00H	Timer Mode Control
10	TL0	8AH	00H	Timer 0, low byte
11	TL1	8BH	00H	Timer 1, low byte
12	TH0	8CH	00H	Timer 0, high byte
13	TH1	8DH	00H	Timer 1, high byte
14	P1	90H	FFH	Port 1
15	SCON	98H	00H	Serial Port 0, Control Register
16	SBUF	99H	00H	Serial Port 0, Data Buffer
17	P1CON	9BH	00H	P1 control
18	WDTC	9FH	00H	Watch Dog Timer Control
19	P2	A0H	FFH	Port 2
20	IE	A8H	00H	Interrupt Enable
21	IE1	A9H	00H	Interrupt Enable 1
22	IFR	AAH	00H	Interrupt Flag for read
23	P3	B0H	FFH	Port 3
24	PWMD0	B3H	00H	PWM Data 0
25	PWMD1	B4H	00H	PWM Data 1
26	IP	B8H	00H	Interrupt Priority
27	IP1	B9H	00H	Interrupt Priority 1
28	SCONF	BFH	02H	System Control Register
29	TWSIS	C0H	00H	TWSI Status
30	TWSIA	C1H	A0H	TWSI Address
31	TWSIC1	C2H	01H	TWSI Control 1
32	TWSIC2	C3H	00H	TWSI Control 2
33	TWSITXD	C4H	FFH	TWSI Transmit Data
34	TWSIRXD	C5H	00H	TWSI Receive Data
35	T2CON	C8H	00H	Timer 2 Control
36	T2MOD	C9H	00H	Timer 2 Mode
37	RCAP2L	CAH	00H	Timer2 Capture Low
38	RCAP2H	CBH	00H	Timer2 Capture High
39	TL2	CCH	00H	Timer 2, low byte
40	TH2	CDH	00H	Timer 2, high byte
41	PSW	D0H	00H	Program Status Word

42	PWMC0	D3H	00H	PWM Control 0
43	PWMC1	D4H	00H	PWM Control 1
44	P4	D8H	xFH	Port 4
45	ACC	E0H	00H	Accumulator
46	B	F0H	00H	B Register
47	ISPFAH	F4H	00H	ISP Flash Address-High Register
48	ISPFAL	F5H	00H	ISP Flash Address-Low Register
49	ISPFD	F6H	00H	ISP Flash Data Register
50	ISPC	F7H	00H	ISP control Register

Function Description

1. General Features

SM5964B is an 8-bit micro-controller. All of its functions and the detailed meanings of SFR will be given in the following sections.

1.1 Embedded Flash

The program can be loaded into the embedded 64KB Flash memory via its writer or In-System Programming (ISP).

1.2 IO Pads

The SM5964B has Five I/O ports: Port 0, Port 1, Port 2, Port 3, Port 4. Port 0~Port 3 are 8-bit ports. Port 4 is 4-bit port. These are: quasi-bidirectional (standard 8051 port outputs) with Port 1~4, and open drain with Port 0.

All the pads are with slew rate to reduce EMI. The IO pads can withstand ESD in human body mode guaranteeing the SM5964B's quality in high electro-static environments.

1.3 System Control Register (SCONF)

Mnemonic: SCONF **Address: BFh**

7	6	5	4	3	2	1	0	Reset
WDR	-	-	PDWUE	-	ISPE	OME	ALEI	02H

WDR: Watch Dog Timer Reset.

When system reset by Watch Dog Timer overflow, WDR will be set to 1.

User should check WDR bit whenever un-predicted reset happened.

PDWUE: Power down wake-up enable bit.

Set 1 to enable wake-up from power-down state by external pin int0 or int1.

ISPE: ISP function enable bit.

When Enable the ISP function, ISPE will be set to 1.

OME: On-chip 768B expanded RAM enable bit.

Set 1 to enable on-chip 768B expanded RAM access.

ALEI: ALE output inhibit bit.

When default, It can inhibit the clock signal in (Fosc/6) Hz output to the ALE pin.

When set to 1, the ALE pin output will stop to reduce EMI.

2. Instruction Set

All SM5964B instructions are binary code compatible and perform the same functions as they do with the industry standard 8051. The following tables give a summary of the instruction set cycles of the SM5964B Microcontroller core.

Table 2-1: Arithmetic operations

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

Table 2-2: Logic operations

Mnemonic	Description	Code	Bytes	Cycles
ANL A,Rn	AND register to accumulator	58-5F	1	1
ANL A,direct	AND direct byte to accumulator	55	2	1
ANL A,@Ri	AND indirect RAM to accumulator	56-57	1	1
ANL A,#data	AND immediate data to accumulator	54	2	1
ANL direct,A	AND accumulator to direct byte	52	2	1
ANL direct,#data	AND immediate data to direct byte	53	3	2
ORL A,Rn	OR register to accumulator	48-4F	1	1
ORL A,direct	OR direct byte to accumulator	45	2	1
ORL A,@Ri	OR indirect RAM to accumulator	46-47	1	1
ORL A,#data	OR immediate data to accumulator	44	2	1
ORL direct,A	OR accumulator to direct byte	42	2	1
ORL direct,#data	OR immediate data to direct byte	43	3	2
XRL A,Rn	Exclusive OR register to accumulator	68-6F	1	1
XRL A,direct	Exclusive OR direct byte to accumulator	65	2	1
XRL A,@Ri	Exclusive OR indirect RAM to accumulator	66-67	1	1
XRL A,#data	Exclusive OR immediate data to accumulator	64	2	1
XRL direct,A	Exclusive OR accumulator to direct byte	62	2	1
XRL direct,#data	Exclusive OR immediate data to direct byte	63	3	2
CLR A	Clear accumulator	E4	1	1
CPL A	Complement accumulator	F4	1	1
RL A	Rotate accumulator left	23	1	1
RLC A	Rotate accumulator left through carry	33	1	1
RR A	Rotate accumulator right	03	1	1
RRC A	Rotate accumulator right through carry	13	1	1
SWAP A	Swap nibbles within the accumulator	C4	1	1

Table 2-3: Data transfer

Mnemonic	Description	Code	Bytes	Cycles
MOV A,Rn	Move register to accumulator	E8-EF	1	1
MOV A,direct	Move direct byte to accumulator	E5	2	1
MOV A,@Ri	Move indirect RAM to accumulator	E6-E7	1	1
MOV A,#data	Move immediate data to accumulator	74	2	1
MOV Rn,A	Move accumulator to register	F8-FF	1	1
MOV Rn,direct	Move direct byte to register	A8-AF	2	2
MOV Rn,#data	Move immediate data to register	78-7F	2	1
MOV direct,A	Move accumulator to direct byte	F5	2	1
MOV direct,Rn	Move register to direct byte	88-8F	2	2
MOV direct1,direct2	Move direct byte to direct byte	85	3	2
MOV direct,@Ri	Move indirect RAM to direct byte	86-87	2	2
MOV direct,#data	Move immediate data to direct byte	75	3	2
MOV @Ri,A	Move accumulator to indirect RAM	F6-F7	1	1
MOV @Ri,direct	Move direct byte to indirect RAM	A6-A7	2	2
MOV @Ri,#data	Move immediate data to indirect RAM	76-77	2	1
MOV DPTR,#data16	Load data pointer with a 16-bit constant	90	3	2
MOVX A,@Ri	Move external RAM (8-bit addr.) to A	E2-E3	1	2
MOVX A,@DPTR	Move external RAM (16-bit addr.) to A	E0	1	2
MOVX @Ri,A	Move A to external RAM (8-bit addr.)	F2-F3	1	2
MOVX @DPTR,A	Move A to external RAM (16-bit addr.)	F0	1	2
MOVC A,@A+DPTR	Move code byte relative to DPTR to accumulator	93	1	2
MOVC A,@A+PC	Move code byte relative to PC to accumulator	83	1	2
PUSH direct	Push direct byte onto stack	C0	2	2
POP direct	Pop direct byte from stack	D0	2	2
XCH A,Rn	Exchange register with accumulator	C8-CF	1	1
XCH A,direct	Exchange direct byte with accumulator	C5	2	1
XCH A,@Ri	Exchange indirect RAM with accumulator	C6-C7	1	1
XCHD A,@Ri	Exchange low-order nibble indir. RAM with A	D6-D7	1	1

Table 2-4: Program branches

Mnemonic	Description	Code	Bytes	Cycles
ACALL addr11	Absolute subroutine call	xxx11	2	2
LCALL addr16	Long subroutine call	12	3	2
RET	from subroutine	22	1	2
RETI	from interrupt	32	1	2
AJMP addr11	Absolute jump	xxx01	2	2
LJMP addr16	Long iump	02	3	2
SJMP rel	Short jump (relative addr.)	80	2	2
JMP @A+DPTR	Jump indirect relative to the DPTR	73	1	2
JZ rel	Jump if accumulator is zero	60	2	2
JNZ rel	Jump if accumulator is not zero	70	2	2
JC rel	Jump if carry flag is set	40	2	2
JNC	Jump if carry flag is not set	50	2	2
JB bit,rel	Jump if direct bit is set	20	3	2
JNB bit,rel	Jump if direct bit is not set	30	3	2
JBC bit,direct rel	Jump if direct bit is set and clear bit	10	3	2
CJNE A,direct rel	Compare direct byte to A and jump if not equal	B5	3	2
CJNE A,#data rel	Compare immediate to A and jump if not equal	B4	3	2
CJNE Rn,#data rel	Compare immed. to reg. and jump if not equal	B8-BF	3	2
CJNE @Ri,#data rel	Compare immed. to ind. and jump if not equal	B6-B7	3	2
DJNZ Rn,rel	Decrement register and jump if not zero	D8-DF	2	2
DJNZ direct,rel	Decrement direct byte and jump if not zero	D5	3	2
NOP	No operation	00	1	1

Table 2-5: Boolean manipulation

Mnemonic	Description	Code	Bytes	Cycles
CLR C	Clear carry flag	C3	1	1
CLR bit	Clear direct bit	C2	2	1
SETB C	Set carry flag	D3	1	1
SETB bit	Set direct bit	D2	2	1
CPL C	Complement carry flag	B3	1	1
CPL bit	Complement direct bit	B2	2	1
ANL C,bit	AND direct bit to carry flag	82	2	2
ANL C,/bit	AND complement of direct bit to carry	B0	2	2
ORL C,bit	OR direct bit to carry flag	72	2	2
ORL C,/bit	OR complement of direct bit to carry	A0	2	2
MOV C,bit	Move direct bit to carry flag	A2	2	1
MOV bit,C	Move carry flag to direct bit	92	2	2

3. Memory Structure

The SM5964B memory structure follows general 8052 structure. It is 64KB program memory.

3.1 Program Memory

The SM5964B has 64KB on-chip flash memory which can be used as general program memory, on which include up to 4K byte specific ISP service program memory space. The address range for the 64K byte is $0000h to $FFFFh. The address range for the ISP service program is $F000h to $FFFFh. The ISP service program size can be partitioned as N blocks of 512 byte (N=0 to 8). When N=0 means no ISP service program space available, total 64K byte memory used as program memory. When N = 1 means address $FE00h to $FFFFh reserved for ISP service program. When N=2 means memory address $FC00h to $FFFFh reserved for ISP service program…etc. Value N can be set and programmed into SM5964B configuration by writer. As shown in Fig. 3-1.

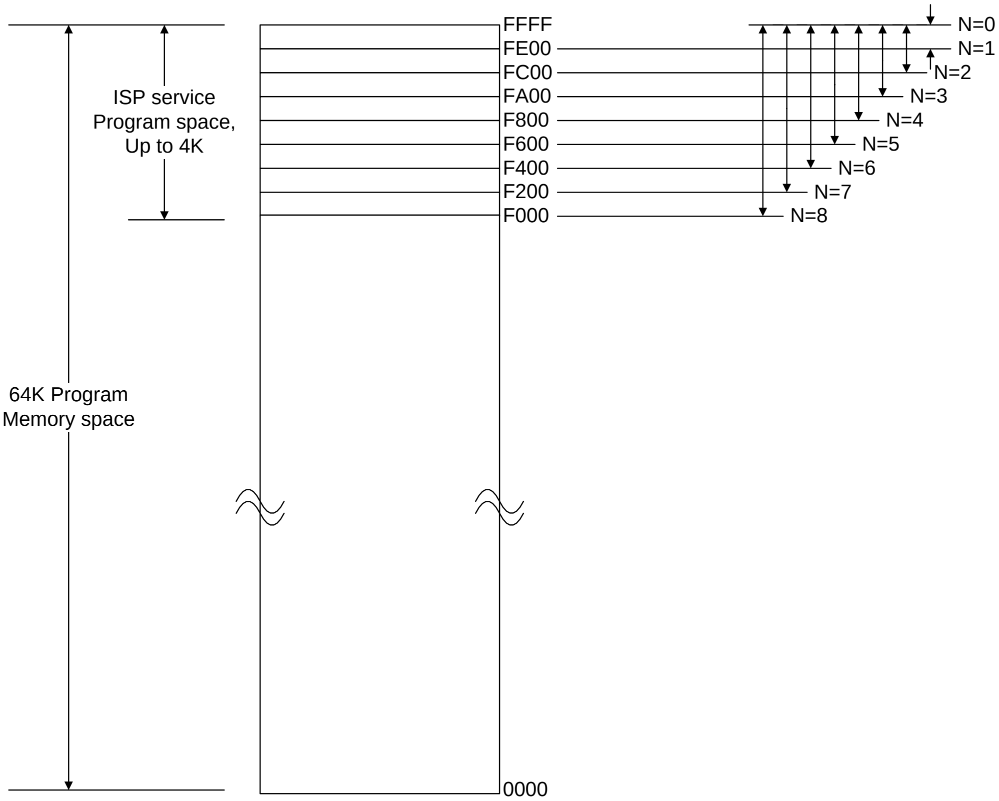

ISP service
Program space,
Up to 4K
64K Program
Memory space
FFFF
FE00
FC00
FA00
F800
F600
F400
F200
F000
0000
N=0
N=1
N=2
N=3
N=4
N=5
N=6
N=7
N=8

Fig. 3-1: SM5964B programmable Flash

3.2 Data Memory

The SM5964B has 256B +768B on-chip SRAM. 256B of it are the same as general 8052 internal memory structure while the expanded 768 Bytes on-chip SRAM can be accessed by external memory addressing method (by instruction MOVX). As shown in Fig. 3-2.

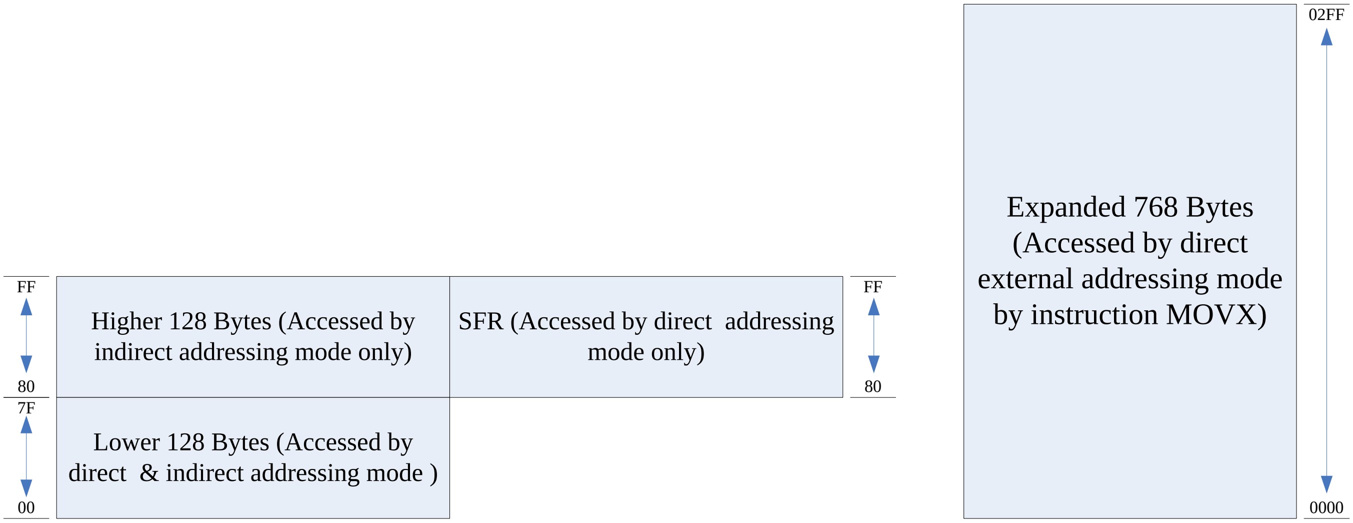

FF
80
7F
00
Higher 128 Bytes (Accessed by indirect addressing mode only)
SFR (Accessed by direct addressing mode only)
FF
80
Lower 128 Bytes (Accessed by direct & indirect addressing mode)
Expanded 768 Bytes (Accessed by direct external addressing mode by instruction MOVX)
02FF
0000

Fig. 3-2: RAM architecture

3.3 Data memory - lower 128 byte (00h to 7Fh)

Data memory 00h to FFh is the same as 8052.
The address 00h to 7Fh can be accessed by direct and indirect addressing modes.
Address 00h to 1Fh is register area.
Address 20h to 2Fh is memory bit area.
Address 30h to 7Fh is for general memory area.

3.4 Data memory - higher 128 byte (80h to FFh)

The address 80h to FFh can be accessed by indirect addressing mode.

Address 80h to FFh is data area.

3.5 Data memory - Expanded 768 bytes ($0000 to $02FF)

From external address 0000h to 02FFh is the on-chip expanded SRAM area, total 768 Bytes. This area can be accessed by external direct addressing mode (by instruction MOVX).

Mnemonic	Description	Dir.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RST
Expanded RAM											
SCONF	System Configuration Register	BFh	WDR	-	-	PDWUE	-	ISPE	OME	ALEI	02H
RCON	Internal RAM Control Register	85h	-	-	-	-	-	-	RAMS1	RAMS0	00H
DBANK	Data Bank Control Register	86h	BSE	-	-	-	BS3	BS2	BS1	BS0	01H

Mnemonic: SCONF **Address: BFh**

7	6	5	4	3	2	1	0	Reset
WDR	-	-	PDWUE	-	ISPE	OME	ALEI	02H

OME: On-chip 768B expanded RAM enable bit.

Set 1 to enable on-chip 768B expanded RAM access.

There are two methods to access on-chip 768 bytes of expanded RAM. The first one is to use MOVX instruction either by MOVX @DPTR or by MOVX @Ri. The other is to use a specific addressing bank switching scheme combined with direct addressing MOV instruction.

Method 1 : By MOVX instruction

The 768 bytes on-chip expanded RAM can be accessed by MOVX instruction with OME bit in SCONF register enabled. If MOVX @DPTR is used, the address range larger than 768B will force to access external data memory automatically. If OME is disabled, IC will always access external data memory. OME is 1 in default setting.

The 768 bytes of on-chip expanded RAM can also be accessed by MOVX @Ri instruction. The content in Ri is concatenated with RAMS1 and RAMS0 bit in RCON register in order to address on-chip 768bytes expanded RAM and off-chip RAM totally up to 64KB memory range.

RAMS1 and RAMS0 bit in RCON register is the 256B bank selection while using MOVX @Ri to access expanded RAM once OME bit in SCONF register is set.

Mnemonic: RCON **Address: 85h**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	RAMS1	RAMS0	00H

OME	RAMS1	RAMS0	Mapped on-chip expanded RAM address	Note
1	0	0	\$0000 - \$00FF	
1	0	1	\$0100 - \$01FF	
1	1	0	\$0200 - \$02FF	
1	1	1	-	Mapped to off-chip RAM address {P2 , Ri}
0	x	x	-	Mapped to off-chip RAM address {P2 , Ri}

Table 3-1 : Mapped address for on-chip expanded RAM

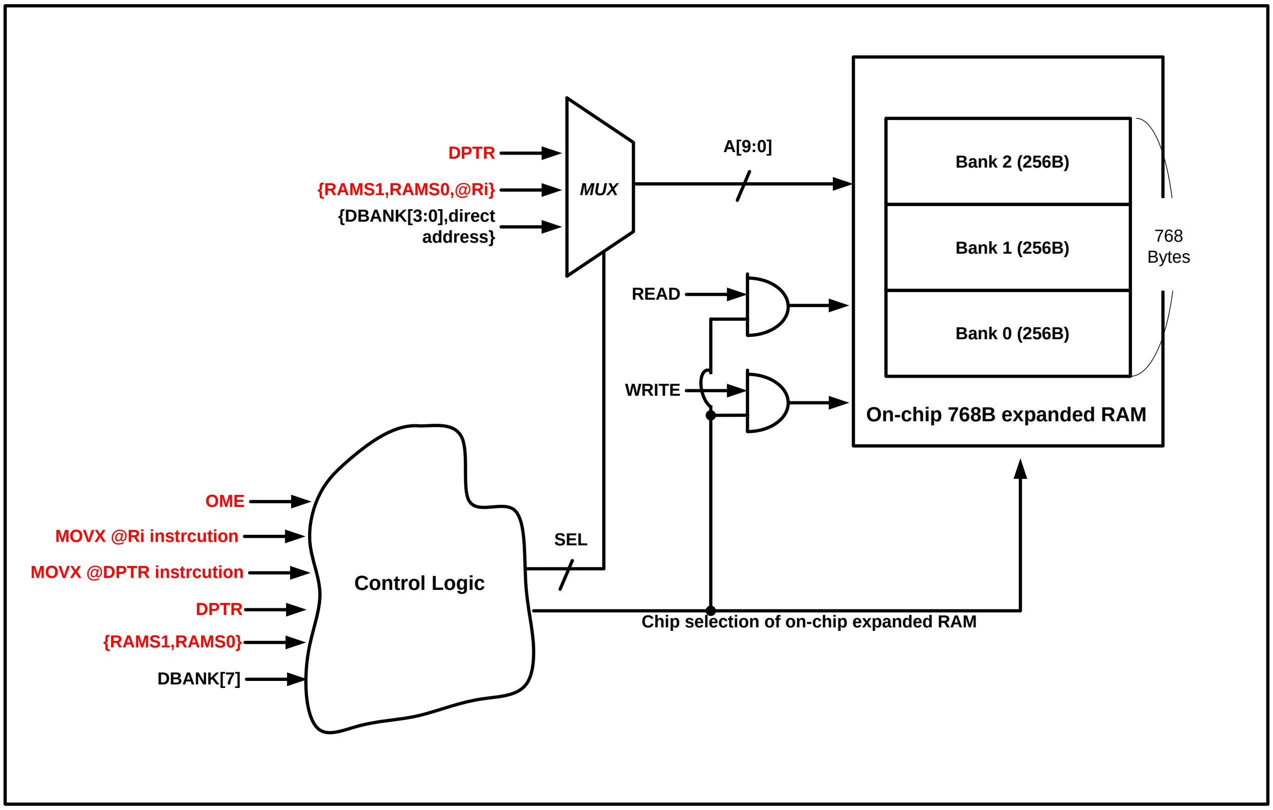

DPTR
{RAMS1,RAMS0,@Ri}
{DBANK[3:0],direct address}
MUX
A[9:0]
Bank 2 (256B)
Bank 1 (256B)
Bank 0 (256B)
768 Bytes
On-chip 768B expanded RAM
READ
WRITE
OME
MOVX @Ri instrcution
MOVX @DPTR instrcution
DPTR
{RAMS1,RAMS0}
DBANK[7]
Control Logic
SEL
Chip selection of on-chip expanded RAM

Fig. 3-3: Access on-chip expanded RAM scheme

Method 2 : By direct addressing MOV instruction

The 768B on-chip expanded data RAM and 256B scratchpad RAM is combined together to form a 1KB memory space. This 1KB space is logically partitioned into 16 pieces of 64B RAM bank. This 1KB space can be accessed through a single-addressing-type MOV instruction with bank-switch technique. In this technique, the 64B address range $40 - $7F in direct addressing MOV instruction is used as mapping window and is concatenated with bit3 – bit0 in DBANK register to address up to 1KB memory space.

While accessed by direct addressing MOV instruction, the 768B expanded RAM is address-offset by 256 bytes upward and concatenates with scratchpad RAM to form a 1KB memory space. Hence 768B expanded RAM occupies address space from $100 to $2FF and 256B scratchpad RAM is located from $000 to $0FF.

With the address mapping window and bank switching scheme, user can use single type MOV instruction to access entire 1K bytes on-chip data memory space. For example, users can have following assembly codes to write data 0x30 into expanded 768B data ram address $101 :

```
MOV    DBANK,#88H
MOV    A,#30H
MOV    41H, A
```

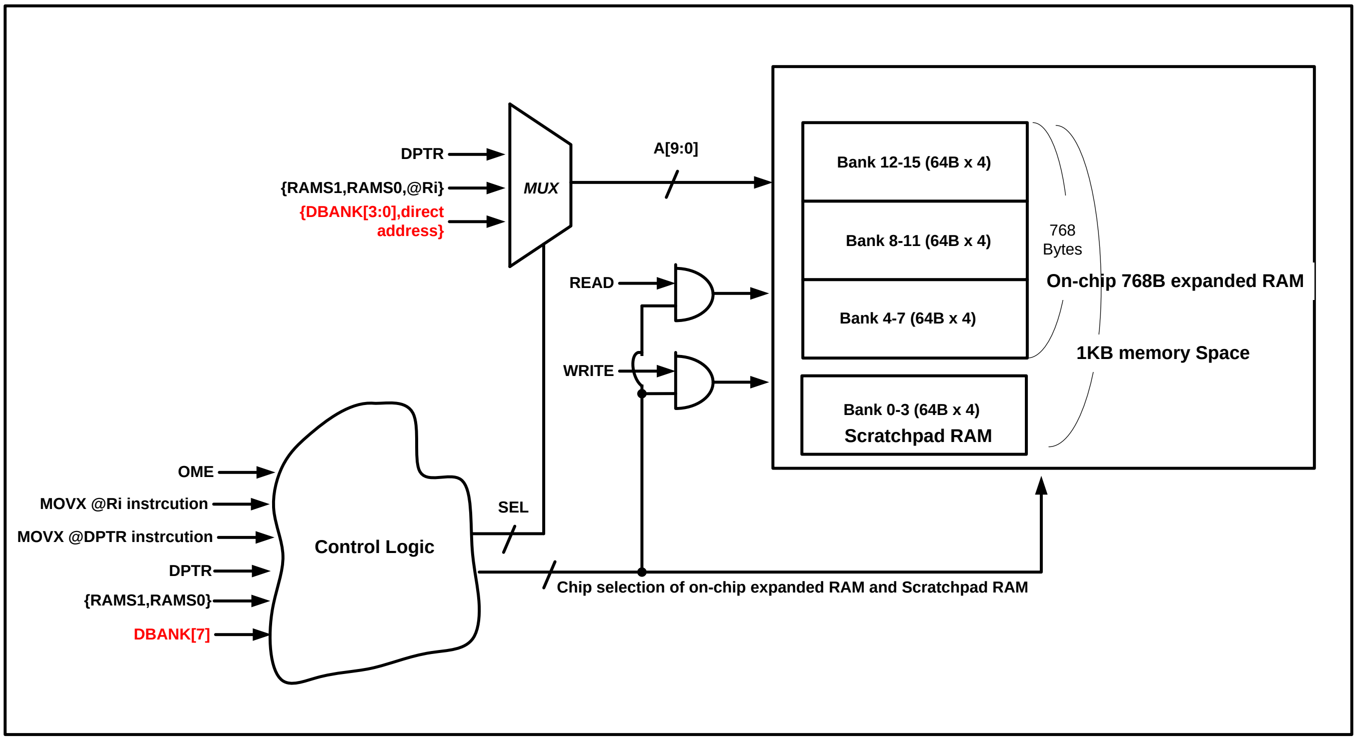

DPTR
{RAMS1,RAMS0,@Ri}
{DBANK[3:0],direct address}
MUX
A[9:0]
READ
WRITE
Bank 12-15 (64B x 4)
Bank 8-11 (64B x 4)
Bank 4-7 (64B x 4)
768 Bytes
On-chip 768B expanded RAM
1KB memory Space
Bank 0-3 (64B x 4)
Scratchpad RAM
OME
MOVX @Ri instrcution
MOVX @DPTR instrcution
DPTR
{RAMS1,RAMS0}
DBANK[7]
Control Logic
SEL
Chip selection of on-chip expanded RAM and Scratchpad RAM

Fig.3-4: Access on-chip expanded RAM and scratchpad RAM with both in single 1KB addressing space scheme

Mnemonic: DBANK **Address: 86h**

7	6	5	4	3	2	1	0	Reset
BSE	-	-	-	BS3	BS2	BS1	BS0	01H

BSE: Set 1 to enable data banking function.

BS[3:0]: One is selected from 16 pieces of 64B data memory bank.

BSE	BS3	BS2	BS1	BS0	Mapped window : $40 - $7F Logically addressed range in 1K memory space	Physical address
1	0	0	0	0	$000 – $03F	Scratchpad RAM ($00 – $FF)
1	0	0	0	1	$040 – $07F	
1	0	0	1	0	$080 – $0BF	
1	0	0	1	1	$0C0 – $0FF	
1	0	1	0	0	$100 – $13F	Expanded RAM ($000 – $2FF)
1	0	1	0	1	$140 – $17F	
1	0	1	1	0	$180 – $1BF	
1	0	1	1	1	$1C0 – $1FF	
1	1	0	0	0	$200 – $23F	
1	1	0	0	1	$240 – $27F	
1	1	0	1	0	$280 – $2BF	
1	1	0	1	1	$2C0 – $2FF	
1	1	1	0	0	$300 – $33F	
1	1	1	0	1	$340 – $37F	
1	1	1	1	0	$380 – $3BF	
1	1	1	1	1	$3C0 – $3FF	
0	x	x	x	x	Mapping is off	Mapping is off

Table3-2: Bank mapping address

4. CPU Engine

The SM5964B engine is composed of four components:

(1) Control unit
(2) Arithmetic – logic unit
(3) Memory control unit
(4) RAM and SFR control unit

The SM5964B engine allows to fetch instruction from program memory and to execute using RAM or SFR. The following chapter describes the main engine register.

Mnemonic	**Description**	**Dir.**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**	**RST**
8051 Core											
ACC	Accumulator	E0h	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	00H
B	B register	F0h	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	00H
PSW	Program status word	D0h	CY	AC	F0	RS[1:0]		OV	PSW.1	P	00H
SP	Stack Pointer	81h	SP[7:0]								07H
DPL	Data pointer low	82h	DPL[7:0]								00H
DPH	Data pointer high	83h	DPH[7:0]								00H

4.1 Accumulator

ACC is the Accumulator register. Most instructions use the accumulator to store the operand.

Mnemonic: ACC **Address: E0h**

7	6	5	4	3	2	1	0	Reset
ACC.7	ACC.6	ACC05	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	00h

ACC[7:0]: The A (or ACC) register is the standard 8052 accumulator.

4.2 B Register

The B register is used during multiply and divide instructions. It can also be used as a scratch pad register to store temporary data.

Mnemonic: B **Address: F0h**

7	6	5	4	3	2	1	0	Reset
B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	00h

B[7:0]: The B register is the standard 8052 register that serves as a second accumulator.

4.3 Program Status Word

Mnemonic: PSW **Address: D0h**

7	6	5	4	3	2	1	0	Reset
CY	AC	F0	RS [1:0]		OV	F1	P	00h

CY: Carry flag.

AC: Auxiliary Carry flag for BCD operations.

F0: General purpose Flag 0 available for user.

RS[1:0]: Register bank select, used to select working register bank.

RS[1:0]	Bank Selected	Location
00	Bank 0	00h – 07h
01	Bank 1	08h – 0Fh
10	Bank 2	10h – 17h
11	Bank 3	18h – 1Fh

OV: Overflow flag.

F1: General purpose Flag 1 available for user.

P: Parity flag, affected by hardware to indicate odd/even number of "one" bits in the Accumulator, i.e. even parity.

4.4 Stack Pointer

The stack pointer is a 1-byte register initialized to 07h after reset. This register is incremented before PUSH and CALL instructions, causing the stack to start from location 08h.

Mnemonic: SP **Address: 81h**

7	6	5	4	3	2	1	0	Reset
SP [7:0]								07h

SP[7:0]: The Stack Pointer stores the scratchpad RAM address where the stack begins. In other words, it always points to the top of the stack.

4.5 Data Pointer

The data pointer (DPTR) is 2-bytes wide. The lower part is DPL, and the highest is DPH. It can be loaded as a 2-byte register (e.g. MOV DPTR, #data16) or as two separate registers (e.g. MOV DPL,#data8). It is generally used to access the external code or data space (e.g. MOVC A,@A+DPTR or MOVX A,@DPTR respectively).

Mnemonic: DPL **Address: 82h**

7	6	5	4	3	2	1	0	Reset
DPL [7:0]								00h

DPL[7:0]: Data pointer Low

Mnemonic: DPH **Address: 83h**

7	6	5	4	3	2	1	0	Reset
DPH [7:0]								00h

DPH [7:0]: Data pointer High

5. GPIO

Port 0 ~ Port 4 are the general purpose IO of this controller. Most of the ports are multiplexed with the other outputs, e.g., Port 3[0] is also used as RXD in the UART application. Port 0 is open-drain in the input and output high condition, so external pull-up resistors are required. As for the other ports, the pull-up resistors are built internally.

For general purpose applications, every pin can be assigned to either high or low independently because their SFRs are bit addressable as given below：

Mnemonic: P0 **Address: 80h**

7	6	5	4	3	2	1	0	Reset
P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	FFh

P0.7~ 0: Port0 [7] ~ Port0[0]

Mnemonic: P1 **Address: 90h**

7	6	5	4	3	2	1	0	Reset
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	FFh

P1.7~ 0: Port1 [7] ~ Port1 [0]

Mnemonic: P2 **Address: A0h**

7	6	5	4	3	2	1	0	Reset
P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	FFh

P2.7~ 0: Port2 [7] ~ Port2 [0]

Mnemonic: P3 **Address: B0h**

7	6	5	4	3	2	1	0	Reset
P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	FFh

P3.7~ 0: Port3 [7] ~ Port3 [0]

Mnemonic: P4 **Address: D8h**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	P4.3	P4.2	P4.1	P4.0	xFh

P4.3~ 0: Port4 [3] ~ Port4 [0]

6. Timer 0 and Timer 1

The SM5964B has three 16-bit timer/counter registers: Timer 0, Timer 1 and Timer 2. All can be configured for counter or timer operations.

In timer mode, the Timer 0 register or Timer 1 register is incremented every machine cycles, due to 12 oscillator periods in a machine cycle, the count rate is 1/12 of the oscillator frequency. If in 6T mode, the count rate is 1/6 of the oscillator frequency.

In counter mode, the register is incremented when the falling edge is observed at the corresponding input pin T0 or T1. Since it takes 2 machine cycles to recognize a 1-to-0 event, the maximum input count rate is 1/2 of the oscillator frequency. There are no restrictions on the duty cycle, however to ensure proper recognition of 0 or 1 state, an input should be stable for at least 1 machine cycle.

Four operating modes can be selected for Timer 0 and Timer 1. Two Special Function registers (TMOD and TCON) are used to select the appropriate mode.

Mnemonic	Description	Dir.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RST
Timer 0 and 1											
TL0	Timer 0 , low byte	8Ah	TL0[7:0]								00H
TH0	Timer 0 , high byte	8Ch	TH0[7:0]								00H
TL1	Timer 1 , low byte	8Bh	TL1[7:0]								00H
TH1	Timer 1 , high byte	8Dh	TH1[7:0]								00H
TMOD	Timer Mode Control	89h	GATE	C/T	M1	M0	GATE	C/T	M1	M0	00H
TCON	Timer/Counter Control	88h	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00H

6.1 Timer/counter mode control register (TMOD)

Mnemonic: TMOD **Address: 89h**

7	6	5	4	3	2	1	0	Reset
GATE	C/T	M1	M0	GATE	C/T	M1	M0	00h
Timer 1				Timer 0				

GATE: If set, enables external gate control (pin INT0 or INT1 for Counter 0 or 1, respectively). When INT0 or INT1 is high, and TRx bit is set (see TCON register), a counter is incremented every falling edge on T0 or T1 input pin.

C/T: Selects Timer or Counter operation. When set to 1, a counter operation is performed, when cleared to 0, the corresponding register will function as a timer.

M1	M0	Mode	Function
0	0	Mode0	13-bit counter/timer, with 5 lower bits in TL0 or TL1 register and 8 bits in TH0 or TH1 register (for Timer 0 and Timer 1, respectively). The 3 high order bits of TL0 and TL1 are hold at zero.
0	1	Mode1	16-bit counter/timer.
1	0	Mode2	8-bit auto-reload counter/timer. The reload value is kept in TH0 or TH1, while TL0 or TL1 is incremented every machine cycle. When TLx overflows, a value from THx is copied to TLx.

1	1	Mode3	If Timer 1 M1 and M0 bits are set to 1, Timer 1 stops. If Timer 0 M1 and M0 bits are set to 1, Timer 0 acts as two independent 8 bit timers / counters.

6.2 Timer/counter control register (TCON)

Mnemonic: TCON **Address: 88h**

7	6	5	4	3	2	1	0	Reset
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00h

TF1: Timer 1 overflow flag set by hardware when Timer 1 overflows. This flag can be cleared by software and is automatically cleared when interrupt is processed.

TR1: Timer 1 Run control bit. If cleared, Timer 1 stops.

TF0: Timer 0 overflow flag set by hardware when Timer 0 overflows. This flag can be cleared by software and is automatically cleared when interrupt is processed.

TR0: Timer 0 Run control bit. If cleared, Timer 0 stops.

IE1: Interrupt 1 edge flag. Set by hardware, when falling edge on external pin INT1 is observed. Cleared when interrupt is processed.

IT1: Interrupt 1 type control bit. Selects falling edge or low level on input pin to cause interrupt. IT1=1, interrupt 1 select falling edge trigger. IT1=0, interrupt1 select low level trigger.

IE0: Interrupt 0 edge flag. Set by hardware, when falling edge on external pin INT0 is observed. Cleared when interrupt is processed.

IT0: Interrupt 0 type control bit. Selects falling edge or low level on input pin to cause interrupt. IT0=1, interrupt 0 select falling edge trigger. IT0=0, interrupt 0 select low level trigger.

6.3 Mode 0 (13-bit Counter/Timer)

The timer register is configured as a 13-bit register. As the count rolls over from all 1s to all 0s, it sets the timer interrupt flag TFx. The counted input is enabled to the timer when TRx = 1 and either GATE=0 or INTx = 1. Mode 0 operation is the same for Timer0 and Timer1.

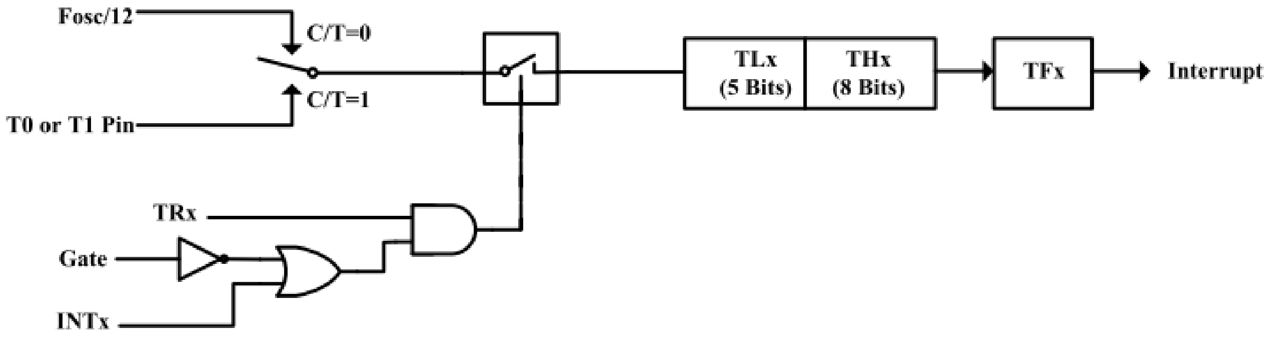


Fig. 6-1: Mode 0 -13 bit Timer / counter operation

6.4 Mode 1 (16-bit Counter/Timer)

Mode1 is the same as Mode0, except that the timer register is being run with all 16 bits.

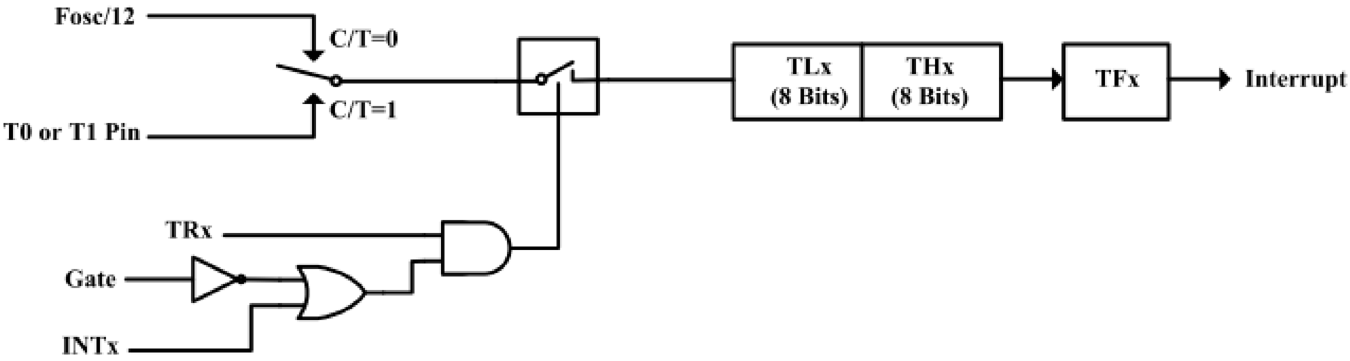

Fosc/12
C/T=0
C/T=1
T0 or T1 Pin
TLx
(8 Bits)
THx
(8 Bits)
TFx
Interrupt
TRx
Gate
INTx

Fig. 6-2: Mode 1 16 bit Counter/Timer operation

6.5 Mode 2 (8-bit auto-reload Counter/Timer)

Mode 2 configures the timer register as an 8-bit counter(TLx) with automatic reload. Overflow from TLx not only set TFx, but also reload TLx with the content of THx, which is determined by software. The reload leaves THx unchanged. Mode 2 operation is the same for Timer0 and Timer1.

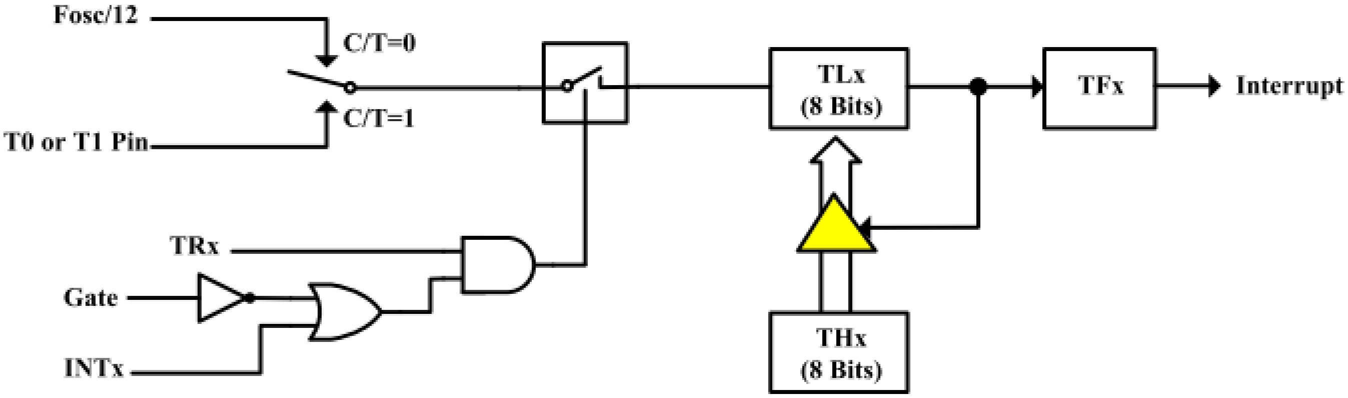

Fosc/12
C/T=0
C/T=1
T0 or T1 Pin
TLx
(8 Bits)
TFx
Interrupt
TRx
Gate
INTx
THx
(8 Bits)

Fig. 6-3: Mode 2 8-bit auto-reload Counter/Timer operation.

6.6 Mode 3 (Timer 0 acts as two independent 8 bit Timers / Counters)

Timer1 in Mode3 simply holds its count, the effect is the same as setting TR1 = 1. Timer0 in Mode 3 enables TL0 and TH0 as two separate 8-bit counters. TL0 uses the Timer0 control bits such like C/T, GATE, TR0, INT0 and TF0. TH0 is locked into a timer function (can not be external event counter) and take over the use of TR1, TF1 from Timer1. TH0 now controls the Timer1 interrupt.

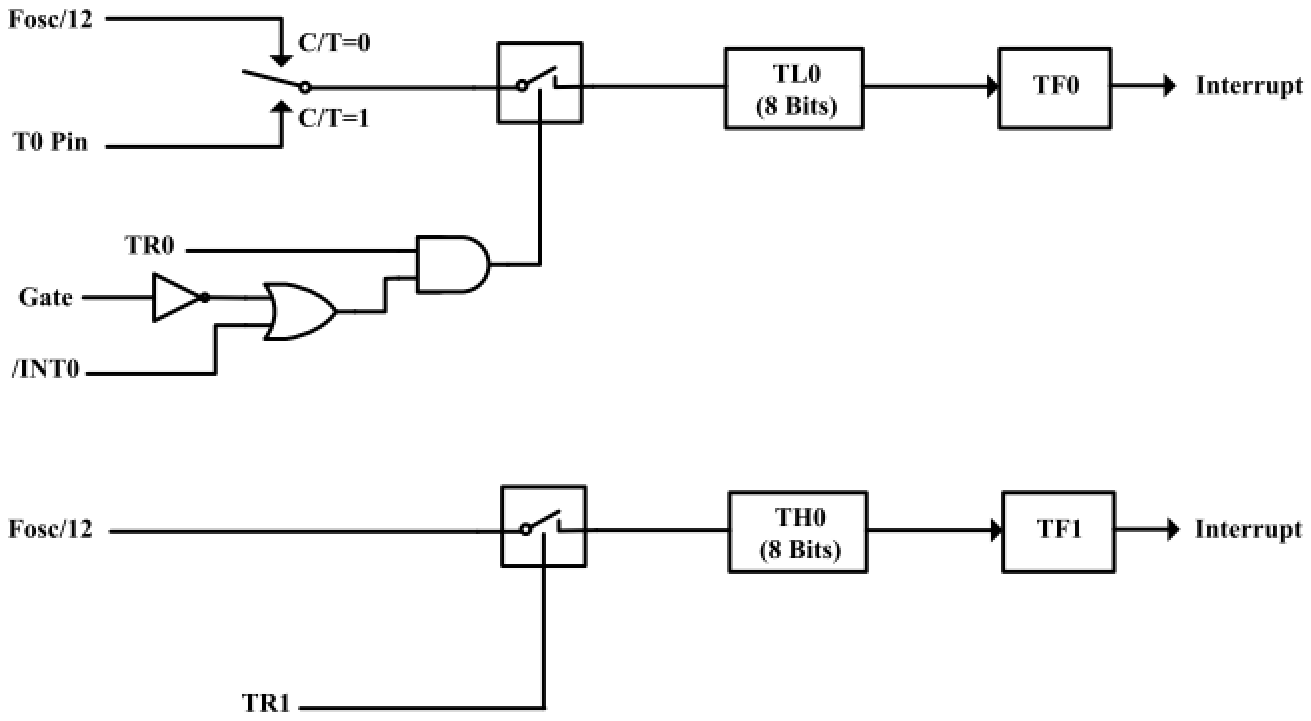

Fosc/12
C/T=0
T0 Pin
C/T=1
TL0
(8 Bits)
TF0
Interrupt
TR0
Gate
/INT0
Fosc/12
TH0
(8 Bits)
TF1
Interrupt
TR1

Fig. 6-4: Mode 3 Timer 0 acts as two independent 8 bit Timers / Counters operatin

7. Timer 2

Timer2 is a 16-bit timer/counter which can operate as either an event timer or an event counter as selected by C/T2 in the special function register T2CON.

Mnemonic	**Description**	**Dir.**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**	**RST**
Timer 2											
TL2	Timer 2 , low byte	CCh	TL2[7:0]								00h
TH2	Timer 2 , high byte	CDh	TH2[7:0]								00h
RCAP2L	Reload and capture data low byte	CAh	RCAP2L[7:0]								00h
RCAP2H	Reload and capture data high byte	CBh	RCAP2H[7:0]								00h
T2MOD	Timer 2 Mode	C9h	-	-	-	-	-	-	T2OE	DCEN	00h
T2CON	Timer 2 Control Register	C8h	TF2	EXF2	RCLK	TCLK	EXEN 2	TR2	$C/\overline{T2}$	CP/ $\overline{RL2}$	00h

Mnemonic: T2MOD **Address: C9h**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	T2OE	DCEN	00H

T2OE: Timer 2 Output Enable bit. It enables Timer 2 overflow rate to toggle P1.0.

DCEN: Down Count Enable bit. When set, this allows Timer 2 to be configured as an up/down Counter.

Mnemonic: T2CON **Address: C8h**

7	6	5	4	3	2	1	0	Reset
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	$C/\overline{T2}$	CP/ $\overline{RL2}$	00H

TF2: Timer 2 overflow flag is set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.

EXF2: Timer 2 external flag is set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 does not cause an interrupt in up/down counter mode (DCEN = 1).

RCLK: Receive clock enable. When set, causes the serial port to use Timer 2 overflow pluses for it's receive clock in serial port Modes 1 and 3. RCLK = 0 causes Timer 1 overflows to be used for the receive clock.

TCLK: Transmit clock enable. When set, causes the serial port to use Timer 2 overflow pulses for it's transmit clock in serial port Modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.

EXEN2: Timer 2 external enable bit. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.

TR2: Start/Stop control for Timer 2. TR2 = 1 starts the timer.

C/$\overline{T2}$: Timer or counter select for Timer 2. C/$\overline{T2}$ = 0 for timer function. C/$\overline{T2}$ = 1 for external event counter (falling edge triggered).

CP/$\overline{RL2}$: Capture/Reload select. CP/$\overline{RL2}$ = 1 causes captures to occur on negative transitions at T2EX if EXEN2 = 1. CP/$\overline{RL2}$ = 0 causes automatic reloads to occur when Timer 2 overflows or negative transitions occur at T2EX when EXEN2 = 1. When either RCLK or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

Table 7-1 : Timer 2 Operating Modes

RCLK + TCLK	CP/RL2	TR2	DCEN	Mode
x	x	0	x	OFF
1	x	1	0	Baud-Rate Generation
0	1	1	0	Capture
0	0	1	0	Auto-Reload Up-only
0	0	1	1	Auto-Reload Up/Down

7.1 Capture mode

In the capture mode, there are two options selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16-bit timer or counter which upon overflow sets bit TF2 in T2CON. This bit can then be used to generate an interrupt. If EXEN2 = 1, Timer 2 still does the above, but with the added feature that a 1-to-0 transition at external input T2EX causes the current value in TH2 and TL2 to be captured into RCAP2H and RCAP2L, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set. The EXF2 bit, like TF2, can generate an interrupt.

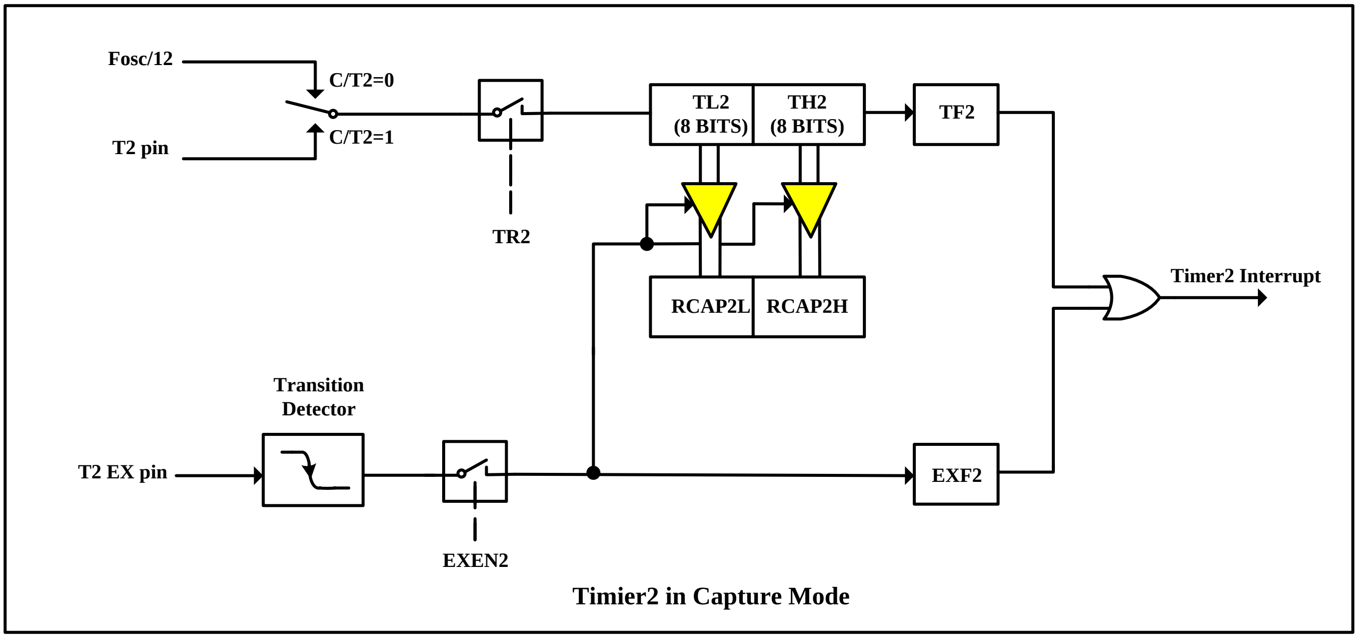


Fig. 7-1： Timer 2 in capture mode

7.2 Auto-reload (Up or Down Counter)

Timer 2 can be programmed to count up or down when configured in its 16-bit auto-reload mode. This feature is invoked by a bit named DCEN (Down Counter Enable) located in the SFR T2MOD. Upon reset, the DCEN bit is set to 0 so that Timer 2 will default to count up. When DCEN is set, Timer 2 can count up or down depending on the value of the T2EX pin.

Fig. 7-2 shows Timer 2 automatically counting up when DCEN = 0. In this mode there are two options selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 counts up to FFFFh and then sets the TF2 bit upon overflow. The overflow also causes the timer registers to be reloaded with the 16-bit value in RCAP2H and RCAP2L. The values in RCAP2H and RCAP2L are preset by software. If EXEN2 = 1, a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at external input T2EX. This transition also sets the EXF2 bit. Both the TF2 and EXF2 bits can generate an interrupt if enabled.

Setting the DCEN bit enables Timer 2 to count up or down as shown in Fig. 7-3. In this mode the T2EX pin controls the direction of count. A logic 1 at T2EX makes Timer 2 count up. The timer will overflow at FFFFh and set the TF2 bit. This overflow also causes the 16-bit value in RCAP2H and RCAP2L to be reloaded into the timer registers, TH2 and TL2, respectively.

A logic 0 at T2EX makes Timer 2 count down. Now the timer underflows when TH2 and TL2 are equal to the values stored in RCAP2H and RCAP2L. The underflow sets the TF2 bit and causes FFFFH to be reloaded into the timer registers.

The EXF2 bit toggles whenever Timer 2 overflows or underflows. This bit can be used as a 17th bit of resolution if desired. In this operating mode, EXF2 does not flag an interrupt.

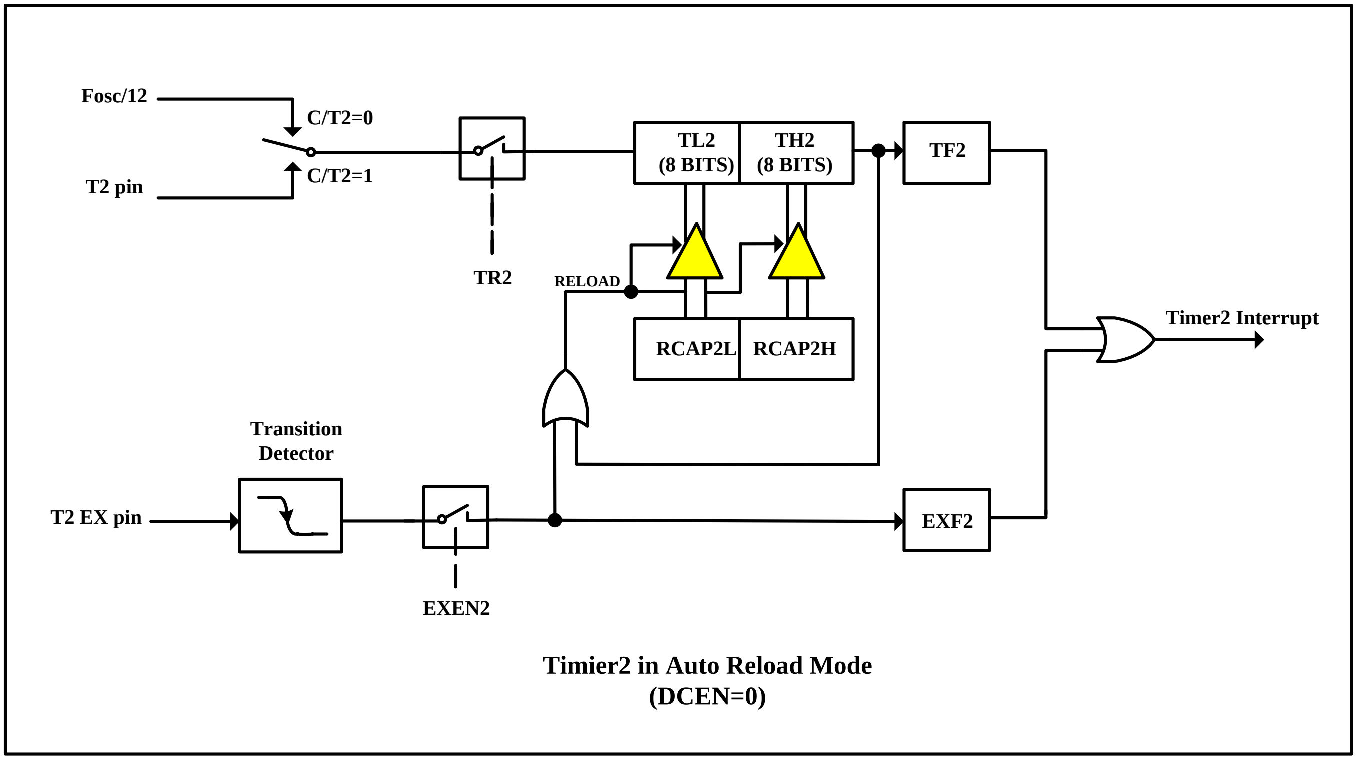


Fig. 7-2：Timer 2 in auto reload mode (DCEN=0)

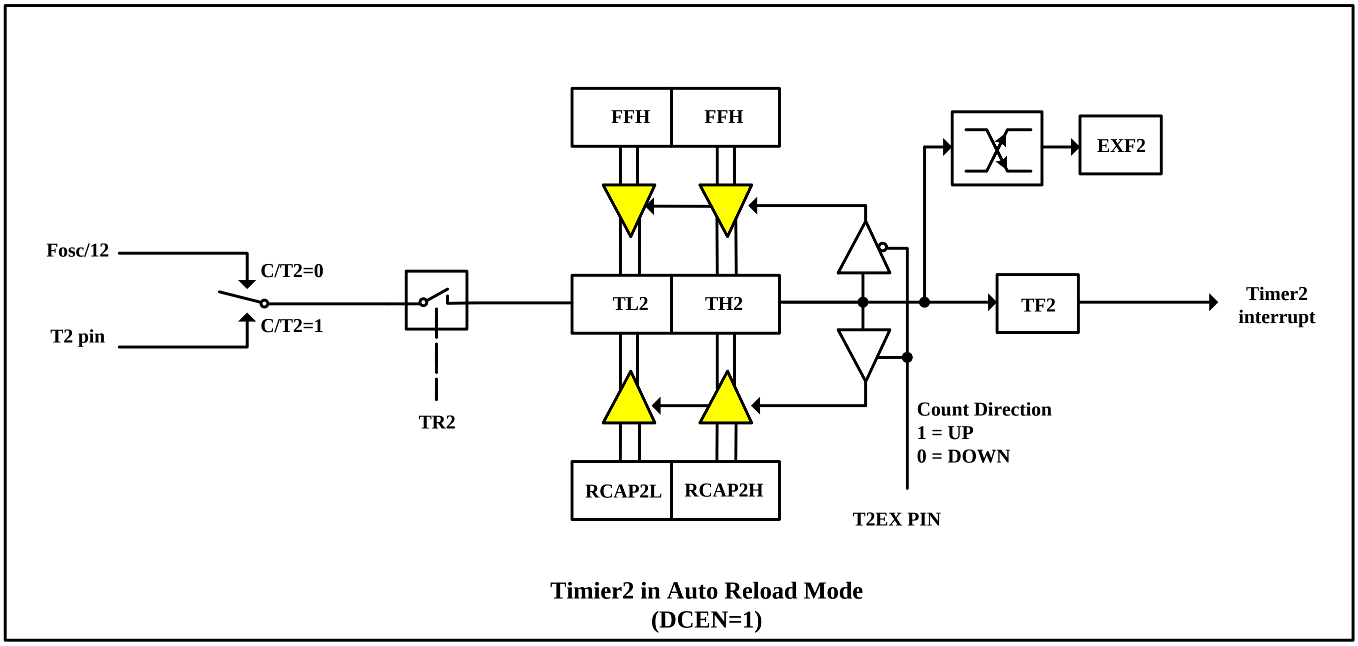


Fig. 7-3： Timer 2 in auto reload mode (DCEN=1)

7.3 Programmable clock out

A 50% duty cycle clock can be programmed to come out on P1.0. This pin, besides begin a regular I/O pin, has two alternate functions. It can be programmed (1) to input the external clock for Timer/Counter 2 or (2) to output a 50% duty cycle clock. An example is that the clock output ranges from 61Hz to 4MHz at a 16MHz oscillator frequency if in 12T mode.

To configure the Timer/Counter 2 as a clock generator, bit $C/\overline{T2}$ (T2CON.1) must be cleared and bit T2OE(T2MOD.1) must be set. Bit TR2 (T2CON.2) starts and stops the timer.

The Clock-Out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L) as shown in this equation:

$$\text{Clock-Out Frequency} = \frac{\text{Oscillator Frequency}}{4 \times (65536 - RCAP2H, RCAP2L)}$$

In the clock-out mode, Timer 2 roll-overs will not generate an interrupt. This is similar to when Timer 2 is used as a baud-rate generator. It is possible to use Timer 2 as a baud-rate generator and a clock generator simultaneously. Note, however, that the baud-rate and clock-out frequencies can not be determined independently from one another since they both use RCAP2H and RCAP2L.

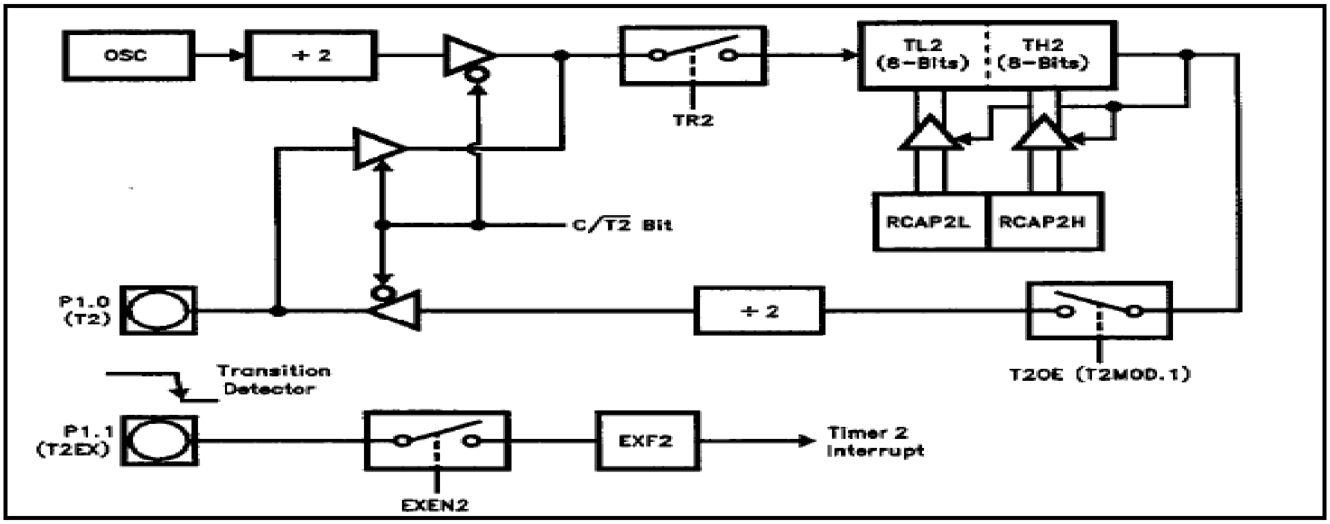

OSC
÷ 2
TR2
TL2 (8-Bits)
TH2 (8-Bits)
RCAP2L
RCAP2H
C/$\overline{T2}$ Bit
P1.0 (T2)
÷ 2
T2OE (T2MOD.1)
Transition Detector
P1.1 (T2EX)
EXEN2
EXF2
Timer 2 Interrupt

Fig. 7-4： Timer 2 in clock-out mode

8. Serial interface – UART

The serial buffer consists of two separate registers, a transmit buffer and a receive buffer.

Writing data to the Special Function Register SBUF sets this data in serial output buffer and starts the transmission. Reading from the SBUF reads data from the serial receive buffer. The serial port can simultaneously transmit and receive data. It can also buffer 1 byte at receive, which prevents the receive data from being lost if the CPU reads the first byte before transmission of the second byte is completed.

Mnemonic	**Description**	**Dir.**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**	**RST**
Serial interface											
PCON	Power Control	87H	SMOD	-	-	-	GF1	GF0	PD	IDLE	00H
SCON	Serial Port Control Register	98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00H
SBUF	Serial Port Data Buffer	99H	SBUF[7:0]								00H

Mnemonic: SCON **Address: 98h**

7	6	5	4	3	2	1	0	Reset
SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00H

SM0, SM1: Serial Port mode selection.

SM0	SM1	Mode
0	0	0
0	1	1
1	0	2
1	1	3

The 4 modes in UART, Mode 0 ~ 3, are explained later.

SM2: Enables multiprocessor communication feature.

REN: If set, enables serial reception. Cleared by software to disable reception.

TB8: The 9th transmitted data bit in modes 2 and 3. Set or cleared by the CPU depending on the function it performs such as parity check, multiprocessor communication etc.

RB8: In modes 2 and 3, it is the 9th data bit received. In mode 1, if SM2 is 0, RB8 is the stop bit. In mode 0, this bit is not used. Must be cleared by software.

TI: Transmit interrupt flag, set by hardware after completion of a serial transfer. Must be cleared by software.

RI: Receive interrupt flag, set by hardware after completion of a serial reception. Must be cleared by software.

8.1 Serial interface

The Serial Interface can operate in the following 4 modes:

SM0	**SM1**	**Mode**	**Description**	**Board Rate**
0	0	0	Shift register	Fosc/12
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	Fosc/32 or Fosc/64
1	1	3	9-bit UART	Variable

Here Fosc is the crystal or oscillator frequency.

The serial port is full duplex, can transmit and receive simultaneously. The serial port receive and transmit share the same SFR – SBUF, but actually there is two SBUF in the chip, one is for transmit and the other is for receive. The serial port can be operated in 4 different modes.

8.1.1 Mode 0

Pin RXD serves as input and output. TXD outputs the shift clock. 8 bits are transmitted with LSB first. The baud rate is fixed at 1/12 of the crystal frequency. Reception is initialized in Mode 0 by setting the flags in SCON as follows: RI = 0 and REN = 1. In other modes, a start bit when REN = 1 starts receiving serial data. As shown in Fig. 8-1 and Fig. 8-2

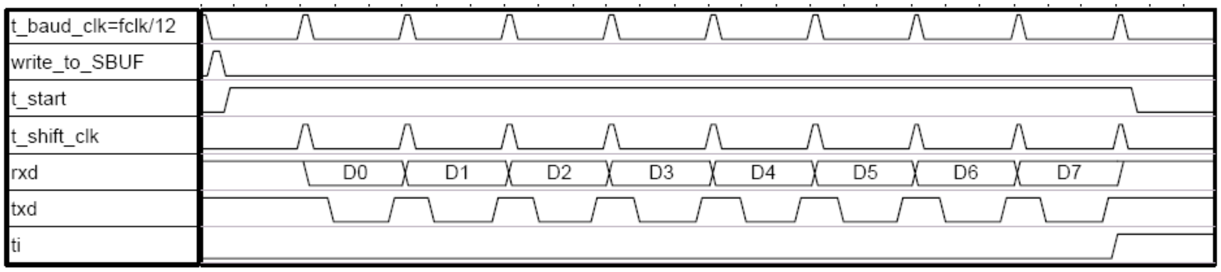

t_baud_clk=fclk/12
write_to_SBUF
t_start
t_shift_clk
rxd
txd
ti
D0
D1
D2
D3
D4
D5
D6
D7

Fig. 8-1: Transmit mode 0

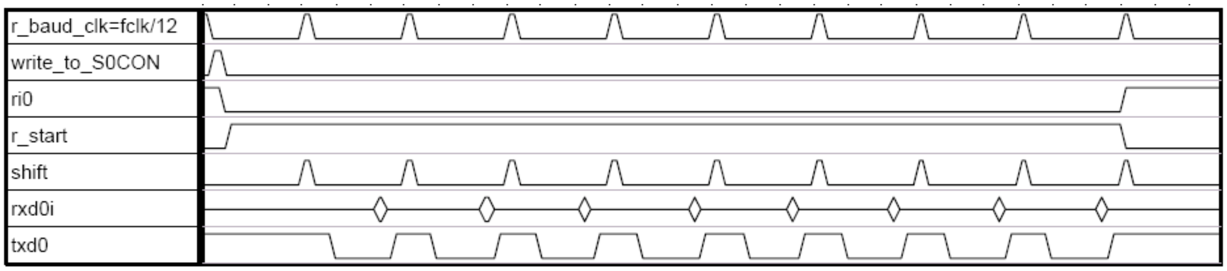

r_baud_clk=fclk/12
write_to_S0CON
ri0
r_start
shift
rxd0i
txd0

Fig. 8-2: Receive mode 0

8.1.2 Mode 1

Pin RXD serves as input, and TXD serves as serial output. No external shift clock is used, 10 bits are transmitted: a start bit (always 0), 8 data bits (LSB first), and a stop bit (always 1). On receive, a start bit synchronizes the transmission, 8 data bits are available by reading SBUF, and stop bit sets the flag RB8 in the Special Function Register SCON. In mode 1 either internal baud rate generator or timer 1 can be use to specify baud rate. As shown in Fig. 8-3 and Fig. 8-4

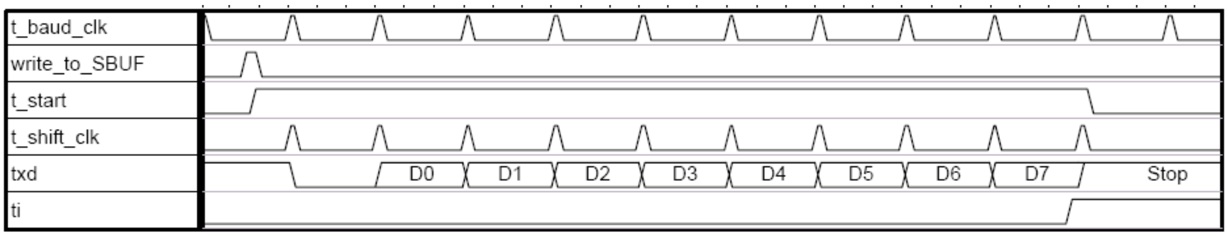

t_baud_clk
write_to_SBUF
t_start
t_shift_clk
txd
ti
D0
D1
D2
D3
D4
D5
D6
D7
Stop

Fig. 8-3: Transmit mode 1

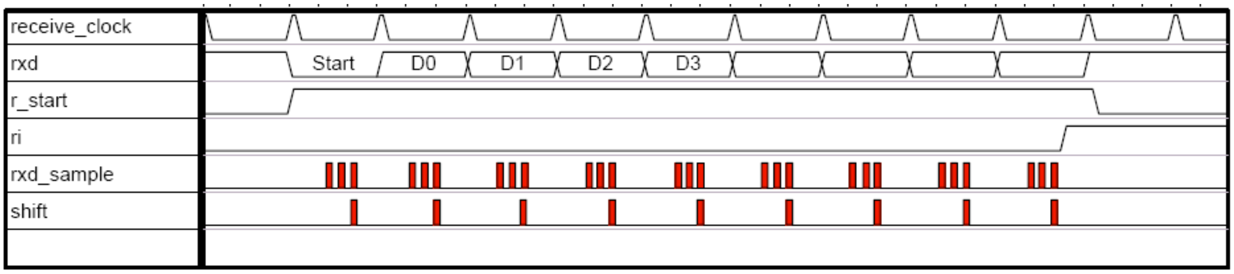

receive_clock
rxd
r_start
ri
rxd_sample
shift
Start
D0
D1
D2
D3

Fig. 8-4: Receive mode 1

8.1.3 Mode 2

This mode is similar to Mode 1, with two differences. The baud rate is fixed at 1/32 (SMOD=1) or 1/64(SMOD=0) of oscillator frequency and 11 bits are transmitted or received: a start bit (0), 8 data bits (LSB first), a programmable 9th bit, and a stop bit (1). The 9th bit can be used to control the parity of the serial interface: at transmission, bit TB8 in SCON is output as the 9th bit, and at receive, the 9th bit affects RB8 in Special Function Register SCON.

8.1.4 Mode 3

The only difference between Mode 2 and Mode 3 is that in Mode 3 either internal baud rate generator or timer 1 can be use to specify baud rate. As shown in Fig. 8-5 and Fig. 8-6.

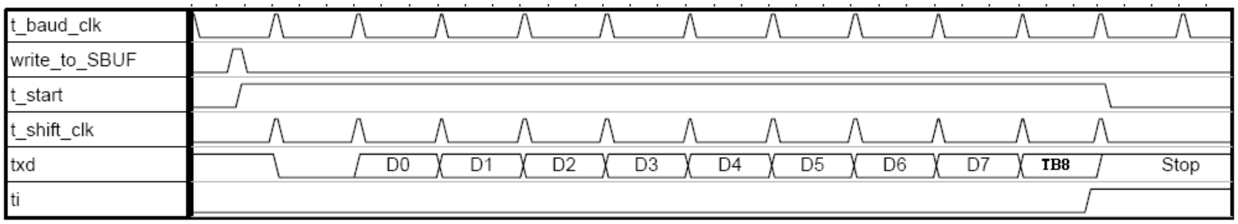

t_baud_clk
write_to_SBUF
t_start
t_shift_clk
txd
ti
D0
D1
D2
D3
D4
D5
D6
D7
TB8
Stop

Fig. 8-5: Transmit modes 2 and 3

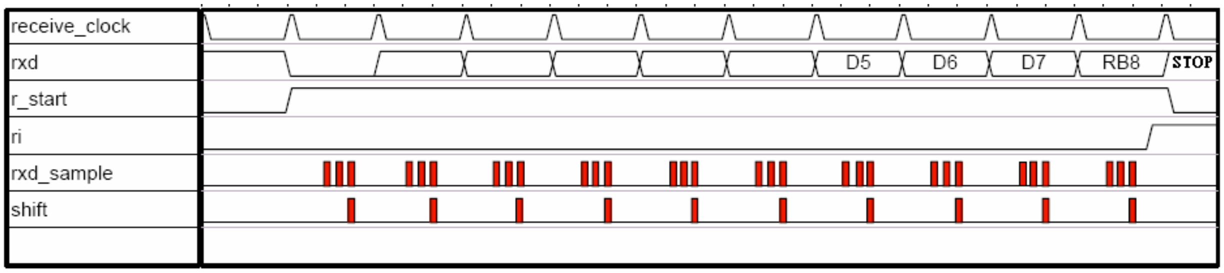

receive_clock
rxd
r_start
ri
rxd_sample
shift
D5
D6
D7
RB8
STOP

Fig. 8-6: Receive modes 2 and 3

8.2 Multiprocessor Communication of Serial Interface

The feature of receiving 9 bits in Modes 2 and 3 of Serial Interface can be used for multiprocessor communication. In this case, the slave processors have bit SM2 in SCON set to 1. When the master processor outputs slave's address, it sets the 9th bit to 1, causing a serial port receive interrupt in all the slaves. The slave processors compare the received byte with their network address. If there is a match, the addressed slave will clear SM2 and receive the rest of the message, while other slaves will leave SM2 bit unaffected and ignore this message. After addressing the slave, the host will output the rest of the message with the 9th bit set to 0, so no serial port receive interrupt will be generated in unselected slaves.

8.3 Baud Rate Generator

8.3.1 Serial interface Mode 0

Baud Rate = $\dfrac{\text{Fosc}}{12}$

8.3.2 Serial interface Mode 2

Baud Rate = $\dfrac{2^{SMOD}}{64} \times$ (Fosc)

8.3.3 Serial interface Mode 1 and 3

8.3.3.1 Using Timer 1 to Generate Baud Rates.

Baud Rate = $\dfrac{2^{SMOD}}{32} \times$ (Timer 1 overflow rate) = $\dfrac{2^{SMOD}}{32} \times \dfrac{\text{Fosc}}{12 \times [256 - TH1]}$

8.3.3.2 Using Timer 2 to Generate Baud Rates.

Baud Rate = $\dfrac{\text{Timer 2 overflow rate}}{32} = \dfrac{\text{Fosc}}{32 \times [65536 - (\text{RCAP2H}, \text{RCAP2L})]}$

9. Interrupt

The SM5964B provides 7 interrupt sources with two priority levels. Each source has its own request flag(s) located in a special function register. Each interrupt requested by the corresponding flag could individually be enabled or disabled by the enable bits in SFR's IE.

When the interrupt occurs, the engine will vector to the predetermined address as given in Table 9-1. Once interrupt service has begun, it can be interrupted only by a higher priority interrupt. The interrupt service is terminated by a return from instruction RETI. When an RETI is performed, the processor will return to the instruction that would have been next when interrupt occurred.

When the interrupt condition occurs, the processor will also indicate this by setting a flag bit. This bit is set regardless of whether the interrupt is enabled or disabled. Each interrupt flag is sampled once per machine cycle, and then samples are polled by hardware. If the sample indicates a pending interrupt when the interrupt is enabled, then interrupt request flag is set. On the next instruction cycle the interrupt will be acknowledged by hardware forcing an LCALL to appropriate vector address.

Interrupt response will require a varying amount of time depending on the state of microcontroller when the interrupt occurs. If microcontroller is performing an interrupt service with equal or greater priority, the new interrupt will not be invoked. In other cases, the response time depends on current instruction.

Table 9-1: Interrupt vectors

Priority level	Interrupt Request Flags	Interrupt Vector Address	Interrupt Number *(use Keil C Tool)
1 (highest)	IE0 – External interrupt 0	0003h	0
2	TF0 – Timer 0 interrupt	000Bh	1
3	IE1 – External interrupt 1	0013h	2
4	TF1 – Timer 1 interrupt	001Bh	3
5	RI/TI – Serial channel interrupt	0023h	4
6	TF2/EXF2 – Timer 2 interrupt	002Bh	5
7	Two Wire Serial Interface	003Bh	7

*See Keil C about C51 User's Guide about Interrupt Function description

Mnemonic	Description	Dir.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RST
Interrupt											
IE	Interrupt Enable Register	A8H	EA	-	ET2	ES	ET1	EX1	ET0	EX0	00H
IE1	Interrupt Enable Register 1	A9H	-	-	-	-	-	-	ETWSI	-	00H
IP	Interrupt Priority Register	B8H	-	-	PT2	PS	PT1	PX1	PT0	PX0	00H
IP1	Interrupt Priority Register 1	B9H	-	-	-	-	-	-	PTWSI	-	00H

Mnemonic: IE **Address: A8h**

7	6	5	4	3	2	1	0	Reset
EA	-	ET2	ES	ET1	EX1	ET0	EX0	00h

EA: EA=0 – Disable all interrupt.
EA=1 – Enable all interrupt.

ET2: ET2=0 – Disable Timer 2 overflow or external reload interrupt.
ET2=1 – Enable Timer 2 overflow or external reload interrupt.

ES: ES=0 – Disable Serial channel interrupt.
ES=1 – Enable Serial channel interrupt.

ET1: ET1=0 – Disable Timer 1 overflow interrupt.
ET1=1 – Enable Timer 1 overflow interrupt.

EX1: EX1=0 – Disable external interrupt 1.
EX1=1 – Enable external interrupt 1.

ET0: ET0=0 – Disable Timer 0 overflow interrupt.
ET0=1 – Enable Timer 0 overflow interrupt.

EX0: EX0=0 – Disable external interrupt 0.
EX0=1 – Enable external interrupt 0.

Mnemonic: IE1 **Address: A9h**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	ETWSI	-	00h

ETWSI: ETWSI =0 – Disable TWSI interrupt.
ETWSI =1 – Enable TWSI interrupt.

Mnemonic: IP **Address: B8h**

7	6	5	4	3	2	1	0	Reset
-	-	PT2	PS	PT1	PX1	PT0	PX0	00H

PT2: Timer2 interrupt priority bit.
PS: Serial port interrupt priority bit.
PT1: Timer1 interrupt priority bit.
PX1: External interrupt 1 priority bit.
PT0: Timer 0 interrupt priority bit.
PX0: External interrupt 0 priority bit.

Mnemonic: IP1 **Address: B9h**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	PTWSI	-	00H

PTWSI: TWSI interrupt priority bit.

Interrupt Priority Table

IP.x	Priority Level
1	1 (highest)
0	2

10. Watch Dog Timer

The Watch Dog Timer (WDT) is an 16-bit free-running counter that generate reset signal if the counter overflows. The WDT is useful for systems which are susceptible to noise, power glitches, or electronics discharge which causing software dead loop or runaway. The WDT function can help user software recover from abnormal software condition. The WDT is different from Timer0, Timer1 and Timer2 of general 8052. To prevent a WDT reset can be done by software periodically clearing the WDT counter. User should check WDR bit of SCONF register whenever un-predicted reset happened. After an external reset the watchdog timer is disabled and all registers are set to zeros.

The WDT has selectable divider input for the time base source clock. To select the divider input, the setting of bit2 ~ bit0 (PS[2:0]) of Watch Dog Timer Control Register (WDTC) should be set accordingly. As shown in Table 10-1.

To enable the WDT is done by setting 1 to the bit 7 (WDTE) of WDTC. After WDTE set to 1, the 16-bit counter starts to count with the selected time base source clock which set by PS2~PS0. It will generate a reset signal when overflows. The WDTE bit will be cleared to 0 automatically when SM5964B been reset, either hardware reset or WDT reset.

To reset the WDT is done by setting 1 to the bit 5 (CLEAR) of WDTC. This will clear the content of the 16-bit counter and let the counter re-start to count from the beginning.

Table 10-1: WDT time-out period

PS[2:0]	**Divider (dividing of Fosc)**	**Time period @ 40MHz**
000	8	13.1ms
001	16	26.21ms
010	32	52.42ms
011	64	104.8ms
100	128	209.71ms
101	256	419.43ms
110	512	838.86ms
111	1024	1677.72ms

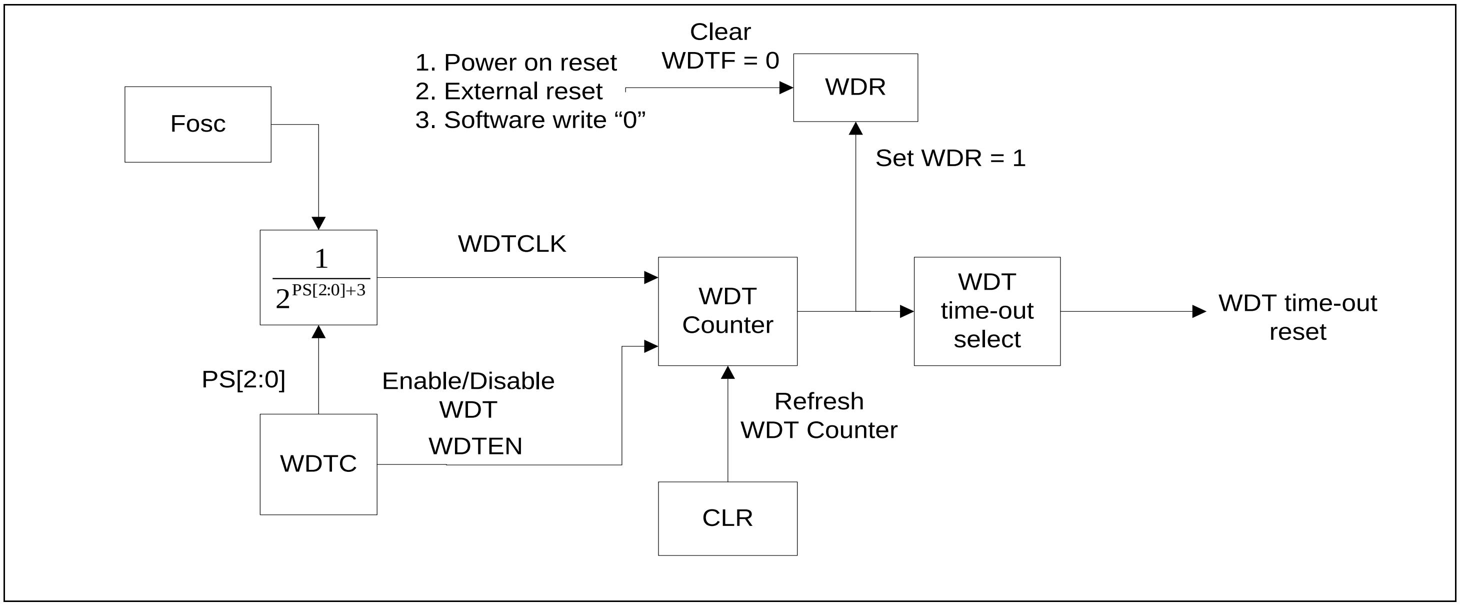


Fig. 10-1: Watchdog timer block diagram

Mnemonic	**Description**	**Dir.**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**	**RST**
Watchdog Timer											
WDTC	Watchdog Timer Control Register	9FH	WDTE	-	CLEAR	-	-	PS [2:0]			00H
SCONF	System Control Register	BFH	WDR	-	-	PDWUE	-	ISPE	OME	ALEI	02H

Mnemonic: WDTC **Address: 9Fh**

7	6	5	4	3	2	1	0	Reset
WDTE	-	CLEAR	-	-	PS [2:0]			00H

WDTE: Watch Dog Timer enable bit.

CLEAR: Watch Dog Timer clear bit.
If CLEAR bit set to1, setting this bit the Watchdog timer counter clear and re-start to count from the Beginning.

PS[2:0]: Watch Dog timer over flow period setting.

Mnemonic: SCONF **Address: BFh**

7	6	5	4	3	2	1	0	Reset
WDR	-	-	PDWUE	-	ISPE	OME	ALEI	02H

WDR Watch Dog Timer Reset.

When system reset by Watch Dog Timer overflow, WDR will be set to 1.

User should check WDR bit whenever un-predicted reset happened.

11. Power Management Unit

Power management unit serves two power management modes, Idle and Power Down, for the users to do power saving function.

Mnemonic: PCON **Address: 87h**

7	6	5	4	3	2	1	0	Reset
SMOD	-	-	-	GF1	GF0	PD	IDLE	00h

GF1: General-purpose flag bit.

GF0: General-purpose flag bit.

PD: Power Down mode control bit. Setting this bit turning on the PD Mode.
PD bit is always read as 0

IDLE: Idle mode control bit. Setting this bit turning on the Idle Mode.
Idle bit is always read as 0

11.1 Idle mode

An instruction that sets PCON.0 causes that to be the last instruction executed before going into the idle mode, the internal clock is gated off to the CPU but not to the interrupt, timer and serial port functions.

There are two ways to terminate the idle. Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware, terminating the idle mode. The interrupt will be serviced, and following RETI, the next instruction to be executed will be the one following the instruction that put the device into idle. Another way to wake-up from idle is to pull RESET high to generate internal hardware reset.

11.2 Power Down mode

An instruction that sets PCON.1 cause that to be the last instruction executed before going into the Power-Down mode. In the power-down mode, the on-chip oscillator is stopped. The contents of on-chip RAM and SFRs are maintained. Be carefully to keep RESET pin active for at least 10ms in order for a stable clock.

The SM5964B can be resumed from power-down state by RESET pin or external interrupt INT0/ INT1. When it is woken-up by RESET, the program will execute from the address 0000H. When it is woken-up by INT0 or INT1, the program will execute from the corresponding interrupt service routine.

To enable wake-up by external interrupt pins, the associated interrupt control register (EA, EX0/EX1) must be configured correctly. Additionally, the control bit PDWUE in SCONF register must be enabled as well.

Mnemonic: SCONF **Address: BFh**

7	6	5	4	3	2	1	0	Reset
WDR	-	-	PDWUE	-	ISPE	OME	ALEI	02H

PDWUE: Power down wake-up enable bit.
Set 1 to enable wake-up from power-down state by external pin int0 or int1.

The wake-up is initiated by an interrupt event at INT 0 or INT 1 pin, and is followed by an internal clock de-bouncing procedure. The de-bouncing logic effectively avoids CPU to run at unstable clock oscillation.

Pin Status in IDLE Mode and Power-Down Mode

Mode	Program Memory	ALE	PSEN	Port0	Port1	Port2	Port3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-Down	Internal	0	0	Data	Data	Data	Data
Power-Down	External	0	0	Float	Data	Data	Data

12. Pulse Width Modulation (PWM)

There are two PWM channels in SM5964B. The resolution of PWM channel can be 8-bit or 5-bit depending on the setting on corresponding PBS bit in PWMCn register, where n-0 or 1.

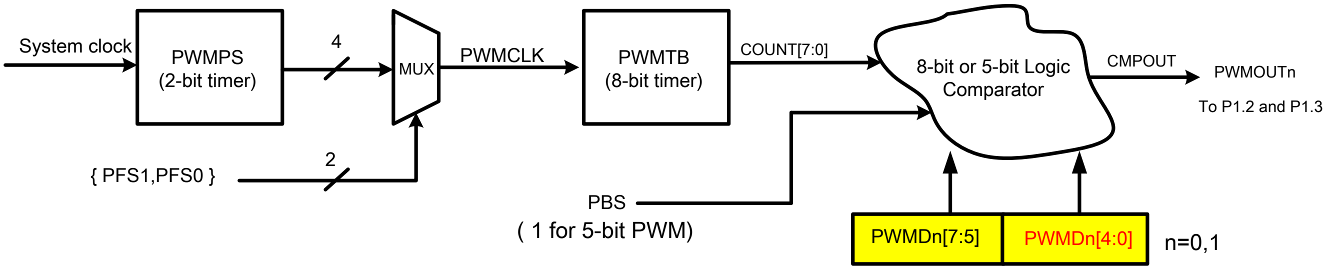

System clock
PWMPS
(2-bit timer)
4
MUX
PWMCLK
PWMTB
(8-bit timer)
COUNT[7:0]
8-bit or 5-bit Logic Comparator
CMPOUT
PWMOUTn
To P1.2 and P1.3
{ PFS1,PFS0 }
2
PBS
(1 for 5-bit PWM)
PWMDn[7:5]
PWMDn[4:0]
n=0,1

Figure : PWMn functional block

Mnemonic	Description	Dir.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RST
					PWM						
P1CON	P1 Control Register	9BH	SDAE	SCLE	-	-	PWM1E	PWM0E	-	-	00H
PWMC0	PWM Control Register 0	D3H	-	-	-	-	-	PBS	PFS1	PFS0	00H
PWMC1	PWM Control Register 1	D4H	-	-	-	-	-	PBS	PFS1	PFS0	00H
PWMD0	PWM Data Register 0	B3H	D0.7	D0.6	D0.5	D0.4	D0.3	D0.2	D0.1	D0.0	00H
PWMD1	PWM Data Register 1	B4H	D1.7	D1.6	D1.5	D1.4	D1.3	D1.2	D1.1	D1.0	00H

Mnemonic: P1CON **Address: 9BH**

7	6	5	4	3	2	1	0	Reset
SDAE	SCLE	-	-	PWM1E	PWM0E	-	-	00h

PWM1E: Set 1 to configure P1[3] as PWM channel 1 output.

PWM0E: Set 1 to configure P1[2] as PWM channel 0 output.

Mnemonic: PWMC[0:1] **Address: D3h and D4h**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	PBS	PFS1	PFS0	00h

PBS: When set, the PWM is 5 bits resolution.

PFS [1:0]: The PWM clock divider select.

PFS1	PFS0	PWM clock divider select
0	0	2
0	1	4
1	0	8
1	1	16

Mnemonic: PWMD[0:1] **Address: B3h and B4h**

7	6	5	4	3	2	1	0	Reset
Dn.7	Dn.6	Dn.5	Dn.4	Dn.3	Dn.2	Dn.1	Dn.0	00h

Where n=0 ~1.
Dn.7 ~ Dn.0 :
They are 8-bit PWM data for 8-bit resolution.
Dn.4 ~ Dn.0 are PWM data for 5-bit resolution.

13. Two-Wire Series Interface (TWSI)

Two Wire Serial Interface, is a serial interface in SM5964B, that is function compatible with IIC 400kps specification and is capable to communicate with standard IIC devices via configuring SM5964B as one of IIC device types - master transmitter, master receiver, slave transmitter or slave receiver device.

Mnemonic	Description	Dir.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RST
TWSI											
P1CON	P1 Control Register	9BH	SDAE	SCLE	-	-	PWM1E	PWM0E	-	-	00H
TWSIS	TWSI Status Register	C0H	RXIF	TXIF	TFAIL	NAKIF	-	RXACK	MST	TXACK	00H
TWSIA	TWSI Address Register	C1H	ADR.6	ADR.5	ADR.4	ADR.3	ADR.2	ADR.1	ADR.0	ADRMK	A0H
TWSIC1	TWSIC Control Register 1	C2H	TWSIE	-	-	-	Bus Busy	TWSIFS2	TWSIFS1	TWSIFS0	01H
TWSIC2	TWSIC Control Register 2	C3H	MATCH	SRW	-	-	RESTART	-	-	MRW	00H
TWSITXD	TWSI TX Data Register	C4H	TXD.7	TXD.6	TXD.5	TXD.4	TXD.3	TXD.2	TXD.1	TXD.0	FFH
TWSIRXD	TWSI RX Data Register	C5H	RXD.7	RXD.6	RXD.5	RXD.4	RXD.3	RXD.2	RXD.1	RXD.0	00H
IFR	Interrupt Flag Register	AAH	-	-	-	-	-	-	TWSIIF	-	00H

Mnemonic: P1CON **Address: 9Bh**

7	6	5	4	3	2	1	0	Reset
SDAE	SCLE	-	-	PWM1E	PWM0E	-	-	00h

SDAE: Set 1 to configure P1[7] as SDA pin of TWSI.

SCLE: Set 1 to configure P1[6] as SCL pin of TWSI.

Mnemonic: TWSIS **Address: C0h**

7	6	5	4	3	2	1	0	Reset
RXIF	TXIF	TFAIL	NAKIF	-	RXACK	MST	TXACK	00h

RXIF: TWSI interrupt flag for data receiving. It is set after the TWSI RXD (Receive Data Buffer) is loaded with new received data.

TXIF: TWSI interrupt flag for data transmitting. It is set when the data of TWSITXD (Transmit Data Buffer) is downloaded onto the shift register or the TWSIA is downloaded onto the shift register at Master Transmit mode.

TFAIL: This flag is set when the data transmit is failed. (Master mode only)

NAKIF: Non-acknowledge interrupt flag. It is only set in the master mode when there is no acknowledge bit detected after one byte data or calling address is transferred.

RXACK: The acknowledge status indicator. When clear, it means an acknowledge signal has been received after the complete 8-bit data transmit on the bus. (Read only)

MST: Set 1 to force working at TWSI master mode.

TXACK: The acknowledge bit for response to transmitter or master addressing. When receiving complete 8-bit data, setting this bit to respond with NACK otherwise ACK is responded.

Mnemonic: TWSIA **Address: C1h**

7	6	5	4	3	2	1	0	Reset
ADR.6	ADR.5	ADR.4	ADR.3	ADR.2	ADR.1	ADR.0	ADRMK	A0h

ADR[6:0]: These 7 bits define slave address on the TWSI/IIC bus.

ADRMK: Address Mask bit. Its only compare 4 bits MSB when set this bit. When this bit is set, ADR.2 – ADR.0 is masked to excluded from the address comparison. In other words, it will be addressed “hit” and respond with ACK as long as ADR6-ADR.3 is matched.

Mnemonic: TWSIC1 **Address: C2h**

7	6	5	4	3	2	1	0	Reset
TWSIE	-	-	-	BusBusy	TWSIFS2	TWSIFS1	TWSIFS0	01h

TWSIE: TWSI function enable bit.

BusBusy: When TWSI bus is detected with “START” condition, this bit is set. When TWSI bus is detected with “STOP” condition, this bit is cleared. (Read only)

TWSIFS[2:0]: TWSI clock rate selector at Master mode.

TWSIFS[2:0]	SCL Frequency
000	Xtal/32
001	Xtal/64 (default)
010	Xtal/128
011	Xtal/256
100	Xtal/512
101	Xtal/1024
110	Xtal/2048
111	Xtal/4096

Mnemonic: TWSIC2 **Address: C3h**

7	6	5	4	3	2	1	0	Reset
MATCH	SRW	-	-	RESTART	-	-	MRW	00h

MATCH: When the first received data (following the START signal) in TWSIRXD register is matches with the address that address register (TWSIA) set, this bit will set. (Read only & Slave mode only)

SRW: The slave mode read (received) or wrote (transmit) on the TWSI bus. When this bit is set, the slave module transmit data on the TWSI bus (SDA). When this bit is clear, the slave module received data on the TWSI bus (SDA). (Read only)

RESTART: This bit only set by master mode. The master will send a start signal then send TWSIA after the ACK signal when this bit setting. If NAKIF was set (the NACK signal was received), the master mode will release, and this bit will clear. (Master mode only)

MRW: This bit is determined the data transmit direction. And this bit will transmit to bus as bit0 at Address (Address is collection TWSIA [7:1] and MRW as 8 bits data). When clear this bit the master is in transmits mode and clear is in receive mode.

Mnemonic: TWSITXD **Address: C4h**

7	6	5	4	3	2	1	0	Reset
TXD.7	TXD.6	TXD.5	TXD.4	TXD.3	TXD.2	TXD.1	TXD.0	FFh

The data written into this register will be automatically downloaded to the shift register when the module detects a calling address is matched and the bit 0 of the received data is one (Slave transmit mode) or when the data in the shift register has been transmitted with received acknowledge bit (RXAK) =0 in transmit mode.

Mnemonic: TWSIRXD **Address: C5h**

7	6	5	4	3	2	1	0	Reset
RXD.7	RXD.6	RXD.5	RXD.4	RXD.3	RXD.2	RXD.1	RXD.0	00h

The TWSI Receive Data Buffer (TWSIRXD) contains the last received data when the MATCH flag is one or the calling address from master when the MATCH flag is zero. The TWSIRXD register will be updated after a data byte is received and the previous received data had been read out, otherwise the TWSI module will pull down to SCL line to inhabit the next data transfer. It is a read-only register. The read operation of this register will clear the RXIF flag. After the RXIF flag is cleared, the register can load the received data again and set the RXIF flag to generate interrupt request for reading the newly received data.

Mnemonic: IFR **Address: AAh**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	TWSIIF	-	00h

TWSIIF: It is the logic-ORed result of following TWSI flags : RXIF, TXIF, TFIF and NAKIF. Firmware can poll this bit to check whether TWSI’s flag is set. (Read only)

14. In-System Programming (Internal ISP)

The SM5964B can generate flash control signal by internal hardware circuit. Users utilize flash control register, flash address register and flash data register to perform the ISP function without removing the SM5964B from the system. The SM5964B provides internal flash control signals which can do flash program/chip erase/page erase/protect functions. User need to design and use any kind of interface which SM5964B can input data. User then utilize ISP service program to perform the flash program/chip erase/page erase/protect functions.

14.1 ISP service program

The ISP service program is a user developed firmware program which resides in the ISP service program space. After user developed the ISP service program, user then determine the size of the ISP service program. User need to program the ISP service program in the SM5964B for the ISP purpose.

The ISP service programs were developed by user so that it should includes any features which relates to the flash memory programming function as well as communication protocol between SM5964B and host device which output data to the SM5964B. For example, if user utilize UART interface to receive/transmit data between SM5964B and host device, the ISP service program should include baud rate, checksum or parity check or any error-checking mechanism to avoid data transmission error.

14.2 Lock Bit (N)

The Lock Bit N has two functions: one is for service program size configuration and the other is to lock the ISP service program space from flash erase function.

The ISP service program space address range $F000 to $FFFF. It can be divided as blocks of N*512 byte. (N=0 to 8). When N=0 means no ISP function, all of 4K byte flash memory can be used as program memory. When N=1 means ISP service program occupies 512 byte while the rest of 3.5K byte flash memory can be used as program memory. The maximum ISP service program allowed is 4K byte when N=8. Under such configuration, the usable program memory space is 4K byte.

The lock bit N function is different from the flash protect function. The flash erase function can erase all of the flash memory except for the locked ISP service program space. If the flash not has been protected, the content of ISP service program still can be read. If the flash has been protected, the overall content of flash program memory space including ISP service program space can not be read. As given in Table 14-1.

Table 14-1 ISP code area

	ISP service program address
0	No ISP service program
1	512 bytes ($FE00h ~ $FFFFh)
2	1K bytes ($FC00h ~ $FFFFh)
3	1.5K bytes ($FA00h ~ $FFFFh)
4	2K bytes ($F800h ~ $FFFFh)
5	2.5K bytes ($F600h ~ $FFFFh)
6	3K bytes ($F400h ~ $FFFFh)
7	3.5K bytes ($F200h ~ $FFFFh)
8	4 K bytes ($F000h ~ $FFFFh)

ISP service program configurable in N*512 byte (N= 0 ~ 8)

14.3 Program the ISP Service Program

After Lock Bit N is set and ISP service program been programmed, the ISP service program memory will be protected (locked) automatically. The lock bit N has its own program/erase timing. It is different from the flash memory program/erase timing so the locked ISP service program can not be erased by flash erase function. If user needs to erase the locked ISP service program, he can do it by writer only. User can not change ISP service program when SM5964B was in system.

14.4 Initiate ISP Service Program

To initiate the ISP service program is to load the program counter (PC) with start address of ISP service program and execute it. There are three ways to do so:

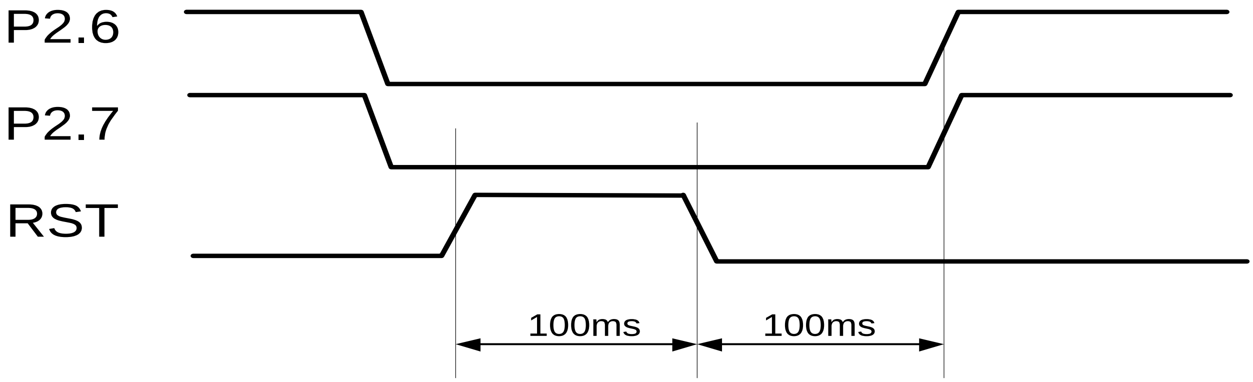
(1) Blank reset. Hardware reset with first flash address blank ($0000=#FFH) will load the PC with start address of ISP service program.

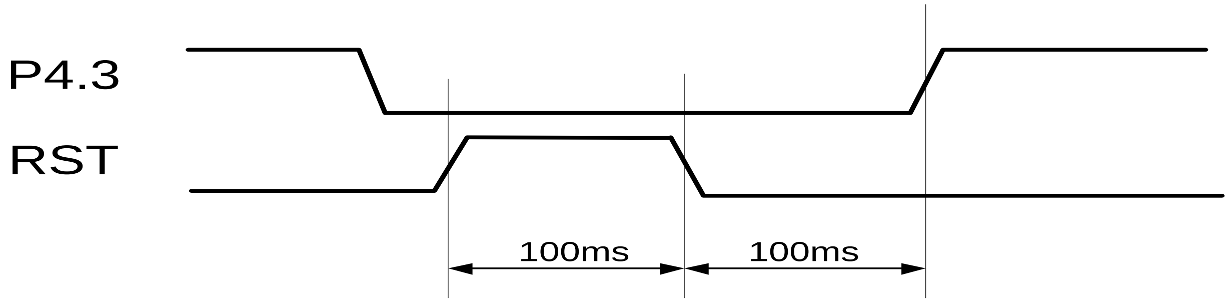
(2) Execute jump instruction can load the start address of the ISP service program to PC.

(3) RESET is asserted with P2.6 and P2.7 both at low state. The default is enable. User can change enable or disable by writer.

(4) RESET is asserted with P4.3. The default is enable. User can change enable or disable by writer.

During the strobe window, the hardware will detect the status of P2.6&P2.7 or P4.3. If they meet one of above conditions, chip will switch to ISP mode automatically. After ISP service program executed, user need to reset the SM5964B, either by hardware reset or by WDT, or jump to the address $0000 to re-start the firmware program.





14.5 ISP register – ISPFAH, ISPFAL, ISPFD and ISPC

Mnemonic	Description	Dir.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RST
ISP function											
ISPFAH	ISP Flash Address – High Register	F4h	ISPFAH [7:0]								00H
ISPFAL	ISP Flash Address - Low Register	F5h	ISPFAL [7:0]								00H
ISPFD	ISP Flash Data Register	F6h	ISPFD [7:0]								00H
ISPC	ISP Control Register	F7h	START	ISPPGSE	ISPPGS[1:0]		-	-	ISPF[1:0]		00H

Mnemonic: ISPFAH **Address: F4H**

7	6	5	4	3	2	1	0	Reset
ISPFAH7	ISPFAH6	ISPFAH5	ISPFAH4	ISPFAH3	ISPFAH2	ISPFAH1	ISPFAH0	00H

ISPFAH [7:0]: Flash address-high for ISP function.

Mnemonic: ISPFAL **Address: F5H**

7	6	5	4	3	2	1	0	Reset
ISPFAL7	ISPFAL6	ISPFAL5	ISPFAL4	ISPFAL3	ISPFAL2	ISPFAL1	ISPFAL0	00H

ISPFAL [7:0]: Flash address-Low for ISP function.

The ISPFAH & ISPFAL provide the 16-bit flash memory address for ISP function. The flash memory address should not include the ISP service program space address. If the flash memory address indicated by ISPFAH & ISPFAL registers overlay with the ISP service program space address, the flash program/page erase of ISP function executed thereafter will have no effect.

Mnemonic: ISPFD **Address: F6H**

7	6	5	4	3	2	1	0	Reset
ISPFD7	ISPFD6	ISPFD5	ISPFD4	ISPFD3	ISPFD2	ISPFD1	ISPFD0	00H

ISPFD [7:0]: Flash data for ISP function.

The ISPFD provide the 8-bit data register for ISP function.

Mnemonic: ISPC **Address: F7h**

7	6	5	4	3	2	1	0	Reset
START	ISPPGSE	ISPPGS[1:0]		-	-	ISPF[1:0]		00H

START: ISP function start bit.
= 1: start ISP function which indicated by bit 1, bit 0 (ISPF[1:0]).
= 0: no operation.

ISPPGSE: ISP Page Selection Enable bit.
= 1: Enable.
= 0: Disable.

ISPPGS[1:0]: ISP Page Selection.

ISPPGSE, ISPPGS1,ISPPGS0	Per page in ISP page erase operation
000	512Byte
001	512Byte
010	512Byte
011	512Byte
100	512Byte
101	256Byte
110	128Byte
111	Reserved

ISPF [1:0]: ISP function select bit.

ISPF[1:0]	ISP function
00	Byte program
01	Chip protect
10	Page erase
11	Chip erase

The choice ISP function will start to execute once the software write data to ISPC register.

To perform byte program/page erases ISP function, user need to specify flash address at first. When performing page erase function, SM5964B will erase entire page which flash address indicated by ISPFAH & ISPFAL registers located within the page.

To perform the chip erase ISP function, SM5964B will erase all the flash program memory except the ISP service program space. To perform chip protect ISP function, the SM5964B flash memory content will be read #00H. The program will miss the interrupt if it happens during the ISP funtion.

e.g. ISP service program to do the byte program - to program #22H to the address $1005H

```
CLR EA              ;disable interrupt
MOV ISPFD, #55h
MOV ISPFD, #0AAh
MOV ISPFD, #55h   ; enable ISPE write attribute
ORL SCONF, #04H    ; enable ISP function
MOV ISPFAH, #10H  ; set flash address-high, 10H
MOV ISPFAL, #05H   ; set flash address-low, 05H
MOV ISPFD, #22H    ; set flash data to be programmed, data = 22H
MOV ISPC, #80H    ; start to program #22H to the flash address $1005H
ANL SCONF, #0FBH   ; disable ISP function
SETB EA             ; enable interrupt
```

Operating Conditions

Symbol	Description	Min.	Typ.	Max.	Unit.	Remarks
TA	Operating temperature	-40	25	85	℃	Ambient temperature under bias
VDD	Supply voltage	3		5.5	V	

DC Characteristics

TA = -40℃ to 85℃, VCC = 5.0V

Symbol	Parameter	Valid	Min.	Max.	Unit	Test Conditions
VIL1	Input Low Voltage	port 0,1,2,3,#EA	-0.5	0.8	V	
VIL2	Input Low Voltage	RES, XTAL1	0	0.8	V	
VIH1	Input High Voltage	port 0,1,2,3,#EA	2.0	Vcc+0.5	V	
VIH2	Input High Voltage	RES, XTAL1	70%Vcc	Vcc+0.5	V	
VOL1	Output Low Voltage	port 0, ALE, #PSEN		0.45	V	IOL=3.2mA
VOL2	Output Low Voltage	port 1,2,3,		0.45	V	IOL=1.6mA
VOH1	Output High Voltage	port 0	2.4		V	IOH=-800uA
			90%Vcc		V	IOH=-80uA
VOH2	Output High Voltage	port 1,2,3,ALE,#PSEN	2.4		V	IOH=-60uA
			90%Vcc		V	IOH=-10uA
IIL	Logical 0 Input Current	port 1,2,3		-75	uA	Vin=0.45V
ITL	Logical Transition Current	port 1,2,3		-650	uA	Vin=2.0V
ILI	Input Leakage Current	port 0, #EA		±10	uA	0.45V<Vin<Vcc
R RES	Reset Pull-down Resistance	RES	50	300	Kohm	
C IO	Pin Capacitance			10	pF	Freq=1MHz, Ta=25 ℃
I CC	Power Supply Current	Vdd		20	mA	Active mode, 16MHz
				6.5	mA	Idle mode, 16MHz
				15	uA	Power down mode