

Product List

SM59R16A2L25, SM59R16A2C25
SM59R08A2L25, SM59R08A2C25

Description

The SM59R16A2/SM59R08A2 is a 1T (one machine cycle per clock) single-chip 8-bit microcontroller. It has 64K/32K-byte embedded Flash for program, and executes all ASM51 instructions fully compatible with MCS-51.

SM59R16A2/SM59R08A2 contains 2KB on-chip RAM, 48 GPIOs, various serial interfaces and many peripheral functions as described below. It can be programmed via writers. Its on-chip ICE is convenient for users in verification during development stage. The high performance of SM59R16A2/ SM59R08A2 can achieve complicated manipulation within short time. About one third of the instructions are pure 1T, and the average speed is 8 times of traditional 8051, the fastest one among all the 1T 51-series. Its excellent EMI and ESD characteristics are advantageous for many different applications.

Ordering Information

SM59R16A2(SM59R08A2)ihhkL

yymm

i: process identifier {L = 2.7V ~ 3.6V, C = 4.5V ~ 5.5V}

hh: working clock in MHz {25}

k: package type postfix {as table below }

L:PB Free identifier

{No text is Non-PB free , "P" is PB free}

yy: year

mm: month

v: version identifier{ A, B,...}

Postfix	Package	Pin / Pad Configuration
W	64L TQFP	Page 2,3
U	64L LQFP	Page 2,3

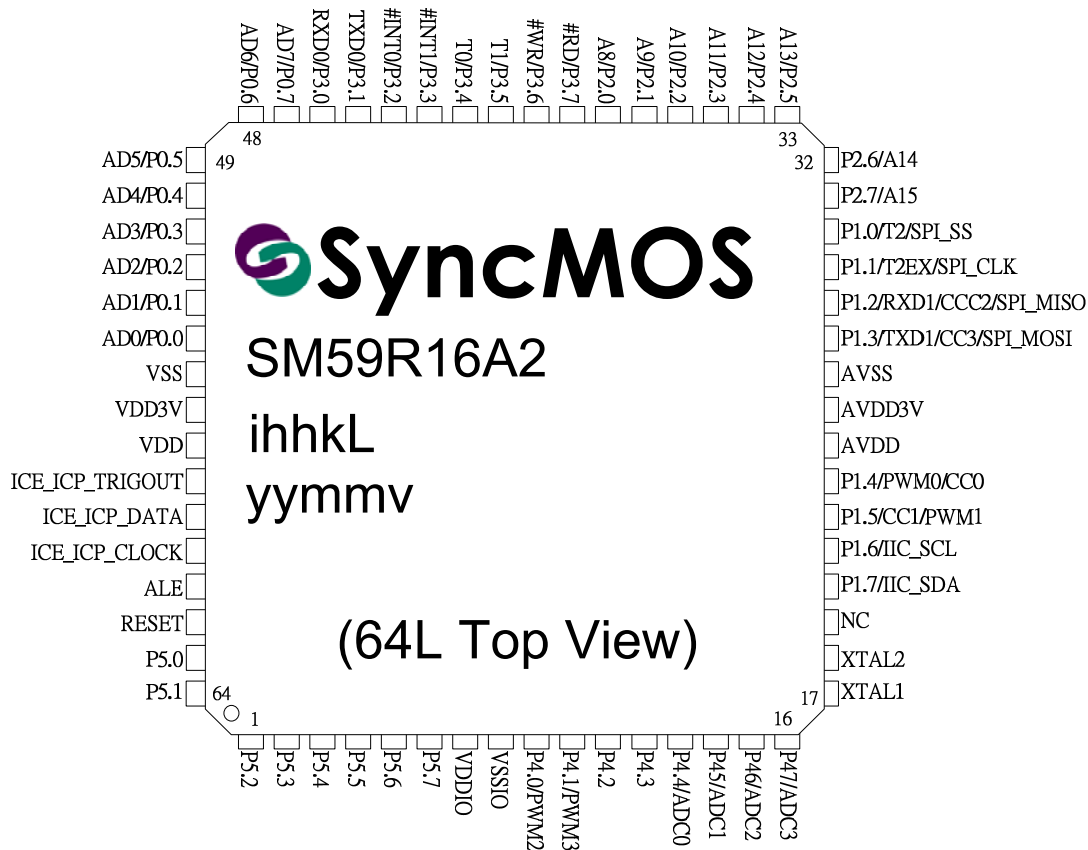
Features

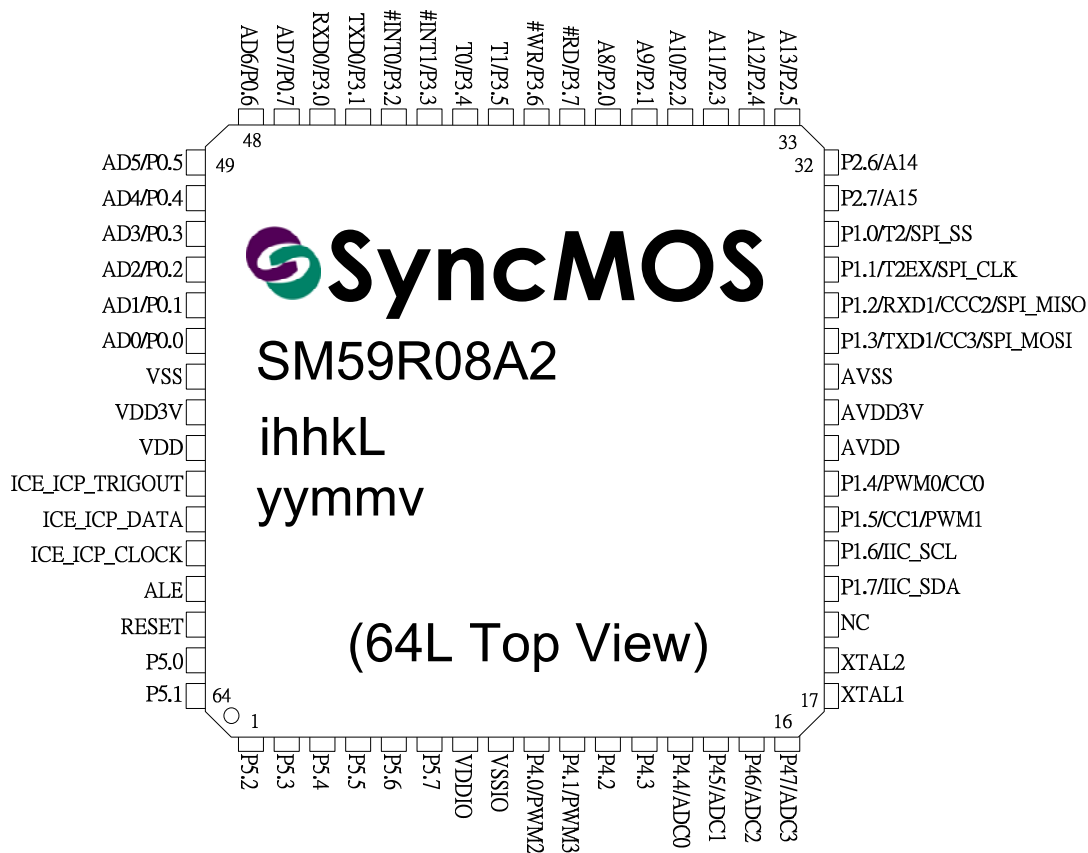
- Operating Voltage: 4.5V ~ 5.5V or 2.7V ~ 3.6V
- High speed architecture of 1 clock/machine cycle (1T), runs up to 25MHz
- 1T/2T can be switched on the fly
- Instruction-set compatible with MCS-51
- Internal OSC with range 1MHz – 24MHz
- 64K/32K bytes on-chip program memory
- External RAM addresses up to 64K bytes. Standard 12T interface for external RAM access.
- 256 bytes RAM as standard 8052, plus 2K bytes on-chip expandable RAM
- Dual 16-bit Data Pointers (DPTR0 & DPTR1)
- Two serial peripheral interfaces in full duplex mode (UART0 & UART1),
- Three 16-bit Timers/Counters. (Timer 0 , 1, 2)
- 48 GPIOs
- External interrupt 0,1 with two priority levels
- Programmable watchdog timer (WDT)
- One IIC interface (Master/Slave mode)
- One SPI interface (Master/Slave mode)
- 4-channel PWM (12-bit, 10-bit, 8-bit options)
- 4-channel 16-bit compare /capture /load functions
- 4-channel 10-bit analog-to-digital converter (ADC)
- ISP/ IAP functions.
- EEPROM function
- On-chip in-circuit emulator (ICE) function with On-Chip Debugger (OCD)
- Fast multiplication-division unit (MDU) : 16*16, 32/16, 16/16, 32-bit L/R shifting and 32-bit normalization
- Expanded External Interrupt (EEI) interface on Port 1 for eight more interrupts
- Enhanced user code protection
- Power management unit for idle and power down modes

Contact SyncMOS : www.syncmos.com.tw
6F, No.10-2 Li- Hsin 1st Road , SBIP, Hsinchu, Taiwan
TEL: 886-3-567-1820 FAX: 886-3-567-1891



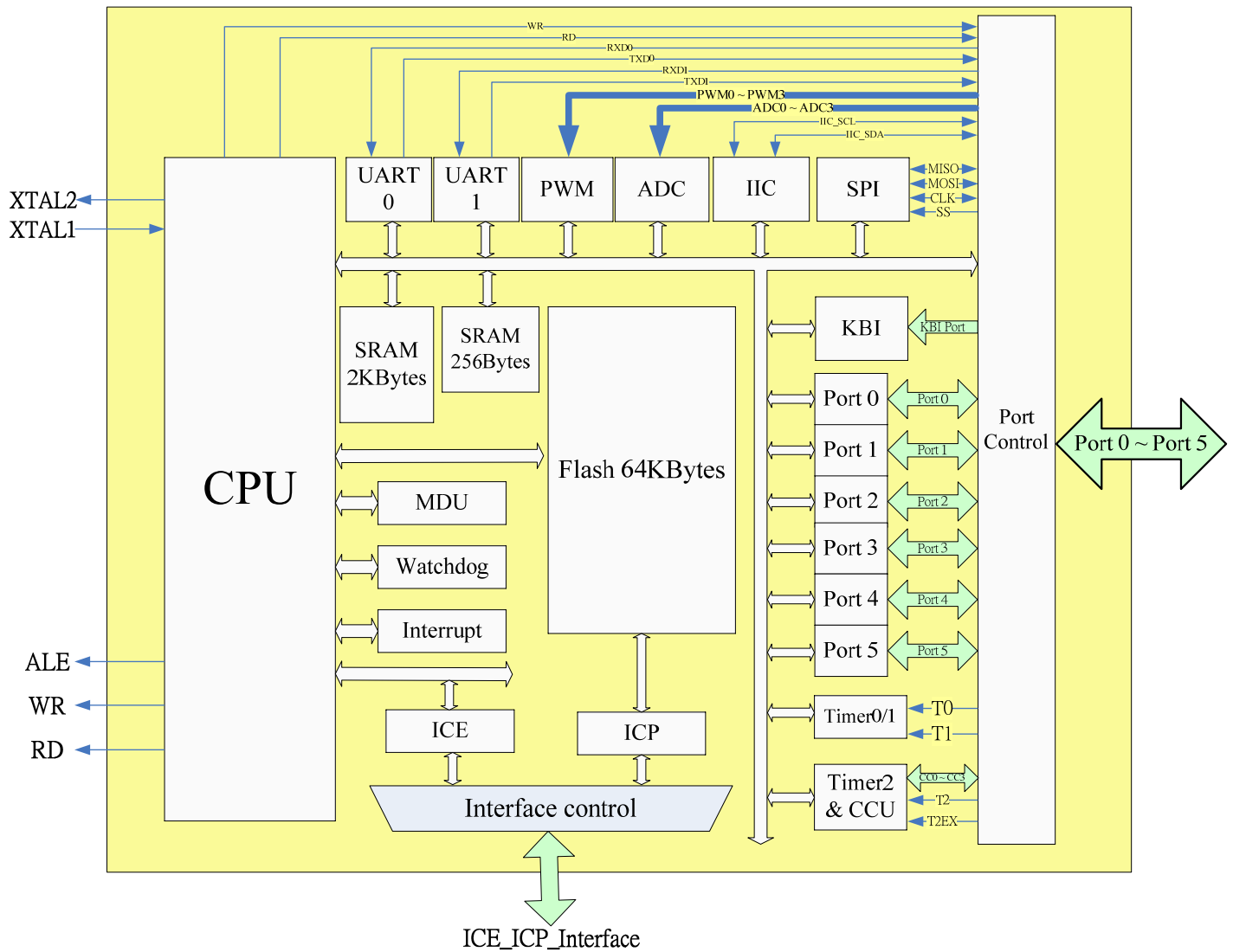
Pin Configuration







Block Diagram





Pin Description

Pin	Name	I/O	Description
1	P5.2	i/o	Bit 2 of port 5
2	P5.3	i/o	Bit 3 of port 5
3	P5.4	i/o	Bit 4 of port 5
4	P5.5	i/o	Bit 5 of port 5
5	P5.6	i/o	Bit 6 of port 5
6	P5.7	i/o	Bit 7 of port 5
7	VDDIO		Power supply
8	VSSIO		Digital ground
9	P4.0/PWM2	i/o	Bit 0 of port 4 & PWM Channel 2
10	P4.1/PWM3	i/o	Bit 1 of port 4 & PWM Channel 3
11	P4.2	i/o	Bit 2 of port 4
12	P4.3	i/o	Bit 3 of port 4
13	P4.4/ADC0	i/o	Bit 4 of port 4 & ADC channel 0
14	P4.5/ADC1	i/o	Bit 5 of port 4 & ADC channel 1
15	P4.6/ADC2	i/o	Bit 6 of port 4 & ADC channel 2
16	P4.7/ADC3	i/o	Bit 7 of port 4 & ADC channel 3
17	XTAL1	i	Crystal input
18	XTAL2	o	Crystal output
19	NC	i	No Connect
20	P1.7/IIC_SDA	i/o	Bit 7 of port 1 & IIC SDA pin
21	P1.6/IIC_SCL	i/o	Bit 6 of port 1 & IIC SCL pin
22	P1.5/PWM1/CC1	i/o	Bit 5 of port 1 & PWM Channel 1 & Timer 2 compare/capture Channel 1
23	P1.4/PWM0/CC0	i/o	Bit 4 of port 1 & PWM Channel 0 & Timer 2 compare/capture Channel 0
24	AVDD		Analog Power supply
25	AVDD3V		Analog Power supply
26	AVSS		Analog ground
27	P1.3/TXD1/CC3/SPI_MOSI	i/o	Bit 3 of port 1 & Serial interface channel 1 & Timer 2 compare/capture Channel 3 & SPI interface Serial Data Input pin
28	P1.2/RXD1/CC2/SPI_MISO	i/o	Bit 2 of port 1 & Serial interface channel 1 & Timer 2 compare/capture Channel 2 & SPI interface Serial Data Out pin
29	P1.1/T2EX/SPI_CLK	i/o	Bit 1 of port 1 & Timer 2 capture trigger & SPI interface Clock pin
30	P1.0/T2/SPI_SS	i/o	Bit 0 of port 1 & Timer 2 external input clock & SPI interface Slave Select pin
31	P2.7/A15	i/o	Bit 7 of port 2 & Bit 15 of external memory address
32	P2.6/A14	i/o	Bit 6 of port 2 & Bit 14 of external memory address
33	P2.5/A13	i/o	Bit 5 of port 2 & Bit 13 of external memory address
34	P2.4/A12	i/o	Bit 4 of port 2 & Bit 12 of external memory address
35	P2.3/A11	i/o	Bit 3 of port 2 & Bit 11 of external memory address
36	P2.2/A10	i/o	Bit 2 of port 2 & Bit 10 of external memory address
37	P2.1/A9	i/o	Bit 1 of port 2 & Bit 9 of external memory address
38	P2.0/A8	i/o	Bit 0 of port 2 & Bit 8 of external memory address
39	P3.7/RD	i/o	Bit 7 of port 3 & external memory read signal
40	P3.6/WR	i/o	Bit 6 of port 3 & external memory write signal

Specifications subject to change without notice contact your sales representatives for the most recent information.



Pin	Name	I/O	Description
41	P3.5/T1	i/o	Bit 5 of port 3 & Timer 1 external input
42	P3.4/T0	i/o	Bit 4 of port 3 & Timer 0 external input
43	P3.3/INT1	i/o	Bit 3 of port 3 & External interrupt 1
44	P3.2/INT0	i/o	Bit 2 of port 3 & External interrupt 0
45	P3.1/TXD0	i/o	Bit 1 of port 3 & Serial interface channel 0
46	P3.0/RXD0	i/o	Bit 0 of port 3 & Serial interface channel 0
47	P0.7/AD7	i/o	Bit 7 of port 0 & Bit 7 of external memory address/data
48	P0.6/AD6	i/o	Bit 6 of port 0 & Bit 6 of external memory address/data
49	P0.5/AD5	i/o	Bit 5 of port 0 & Bit 5 of external memory address/data
50	P0.4/AD4	i/o	Bit 4 of port 0 & Bit 4 of external memory address/data
51	P0.3/AD3	i/o	Bit 3 of port 0 & Bit 3 of external memory address/data
52	P0.2/AD2	i/o	Bit 2 of port 0 & Bit 2 of external memory address/data
53	P0.1/AD1	i/o	Bit 1 of port 0 & Bit 1 of external memory address/data
54	P0.0/AD0	i/o	Bit 0 of port 0 & Bit 0 of external memory address/data
55	VSS		Digital ground
56	VDD3V		Digital Power supply
57	VDD		Digital Power supply
58	ICE_ICP_BUSY	o	Busy (active low during Flash programming) signal in ICE or ICP functions
59	ICE_ICP_DATA	i/o	Command and data IO synchronous to ICE_ICP_CLOCK in ICE or ICP functions
60	ICE_ICP_CLOCK	i	Clock input of ICE and ICP functions
61	ALE	o	Address latch enable
62	RESET	i	Reset pin
63	P5.0	i/o	Bit 0 of port 5
64	P5.1	i/o	Bit 1 of port 5



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	IICS	IICCTL	IICA1	IICA2	IICRWD				FF
F0	B	SPIC1	SPIC2	SPITXD	SPIRXD	SPIS			F7
E8	P4	MD0	MD1	MD2	MD3	MD4	MD5	ARCON	EF
E0	ACC								E7
D8	BRGS								DF
D0	PSW								D7
C8	T2CON		CRCL	CRCH	TL2	TH2			CF
C0	IRCON	CCEN	CCL1	CCH1	CCL2	CCH2	CCL3	CCH3	C7
B8	IEN1	IP1	S0RELH	S1RELH	PWMD0H	PWMD0L	PWMD1H	PWMD1L	BF
B0	P3	PWMD2H	PWMD2L	PWMD3H	PWMD3L	PWMC	WDTC	WDTK	B7
A8	IEN0	IP0	S0RELL	ADCC1	ADCC2	ADCDH	ADCDL	CLKR	AF
A0	P2	PES							A7
98	S0CON	S0BUF	IEN2	S1CON	S1BUF	S1RELL			9F
90	P1	P5	DPS	KBLS	KBE	KBF			97
88	TCON	TMOD	TL0	TL1	TH0	TH1		IFCON	8F
80	P0	SP	DPL	DPH	DPL1	DPH1		PCON	87

Note: Special Function Registers reset values and description for SM59R16A2/SM59R08A2

Register	Location	Reset value	Description
P0	80h	FFh	Port 0
SP	81h	07h	Stack Pointer
DPL	82h	00h	Data Pointer 0 low byte
DPH	83h	00h	Data Pointer 0 high byte
DPL1	84h	00h	Data Pointer 1 low byte
DPH1	85h	00h	Data Pointer 1 high byte
PCON	87h	00h	Power Control
TCON	88h	00h	Timer/Counter Control
TMOD	89h	00h	Timer Mode Control
TL0	8Ah	00h	Timer 0, low byte
TL1	8Bh	00h	Timer 1, low byte
TH0	8Ch	00h	Timer 0, high byte
TH1	8Dh	00h	Timer 1, high byte
IFCON	8Fh	00h	Interface control register
P1	90h	FFh	Port 1
P5	91h	FFh	Port 5
DPS	92h	00h	Data Pointer select Register
KBLS	93h	00h	Expanded External Interrupt (EEI) level selector register
KBE	94h	00h	Expanded External Interrupt (EEI) input enable register

Specifications subject to change without notice contact your sales representatives for the most recent information.



Register	Location	Reset value	Description
KBF	95h	00h	Expanded External Interrupt (EEI) interrupt flag register
S0CON	98h	00h	Serial Port 0, Control Register
S0BUF	99h	00h	Serial Port 0, Data Buffer
IEN2	9Ah	00h	Interrupt Enable Register 2
S1CON	9Bh	00h	Serial Port 1, Control Register
S1BUF	9Ch	00h	Serial Port 1, Data Buffer
S1RELL	9Dh	00h	Serial Port 1, Reload Register, low byte
P2	A0h	FFh	Port 2
PES	A1h	00h	Program Memory Page Erase Control Register
IEN0	A8h	00h	Interrupt Enable Register 0
IP0	A9h	00h	Interrupt Priority Register 0
S0RELL	AAh	D9h	Serial Port 0, Reload Register, low byte
ADCC1	ABh	00h	ADC control register 1
ADCC2	ACH	00h	ADC control register 2
ADCDH	ADh	00h	ADC high data byte
ADCDL	AEnh	00h	ADC low data byte
CLKR	AFh	03h	Clock range register
P3	B0h	FFh	Port 3
PWMD2H	B1h	00h	PWM channel 2 data high byte
PWMD2L	B2h	00h	PWM channel 2 data low byte
PWMD3H	B3h	00h	PWM channel 3 data high byte
PWMD3L	B4h	00h	PWM channel 3 data low byte
PWMC	B5h	00h	PWM control register
WDTCT	B6h	00h	Watchdog timer control register
WDTK	B7h	00h	Watchdog timer refresh key.
IEN1	B8h	00h	Interrupt Enable Register 1
IP1	B9h	00h	Interrupt Priority Register 1
S0RELH	BAh	03h	Serial Port 0, Reload Register, high byte
S1RELH	BBh	03h	Serial Port 1, Reload Register, high byte
PWMD0H	BCh	00h	PWM channel 0 data high byte
PWMD0L	BDh	00h	PWM channel 0 data low byte
PWMD1H	BEh	00h	PWM channel 1 data high byte
PWMD1L	BFh	00h	PWM channel 1 data low byte
IRCON	C0h	00h	Interrupt Request Control Register
CCEN	C1h	00h	Compare/Capture Enable Register
CCL1	C2h	00h	Compare/Capture Register 1, low byte
CCH1	C3h	00h	Compare/Capture Register 1, high byte
CCL2	C4h	00h	Compare/Capture Register 2, low byte



Register	Location	Reset value	Description
CCH2	C5h	00h	Compare/Capture Register 2, high byte
CCL3	C6h	00h	Compare/Capture Register 3, low byte
CCH3	C7h	00h	Compare/Capture Register 3, high byte
T2CON	C8h	00h	Timer 2 Control
CRCL	CAh	00h	Compare/Reload/Capture Register, low byte
CRCH	CBh	00h	Compare/Reload/Capture Register, high byte
TL2	CCh	00h	Timer 2, low byte
TH2	CDh	00h	Timer 2, high byte
PSW	D0h	00h	Program status word
BRGS	D8h	00h	Baud rate generator switch
ACC	E0h	00h	Accumulator
P4	E8h	FFh	Port 4
MD0	E9h	00h	Multiplication/Division Register 0
MD1	EAh	00h	Multiplication/Division Register 1
MD2	EBh	00h	Multiplication/Division Register 2
MD3	ECh	00h	Multiplication/Division Register 3
MD4	EDh	00h	Multiplication/Division Register 4
MD5	EEh	00h	Multiplication/Division Register 5
ARCON	EFh	00h	Arithmetic Control register
B	F0h	00h	B register
SPIC1	F1h	08h	SPI control register 1
SPIC2	F2h	00h	SPI control register 2
SPITxD	F3h	00h	SPI transmit data buffer
SPIRxD	F4h	00h	SPI receive data buffer
SPIS	F5h	40h	SPI status register
IICS	F8h	00h	IIC status register
IICCTL	F9h	04h	IIC control register
IICA1	FAh	A0h	IIC Address 1 register
IICA2	FBh	60h	IIC Address 2 register
IICRWD	FCh	00h	IIC Read/Write register

Function Description

1 General Features

SM59R16A2/SM59R08A2 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/32KB Flash memory via its writer or In-System Programming (ISP). The high-quality Flash has a 100K-write cycle life, suitable for re-programming and data recording as EEPROM.

1.2 IO Pads

The IO pads are compatible to the 8052 series. P0 is open-drain in the input or output high condition, so the external pull-up resistor is required. P1 ~ P5 are designed with internal pull-up resistors. The IO pad structure is given below:

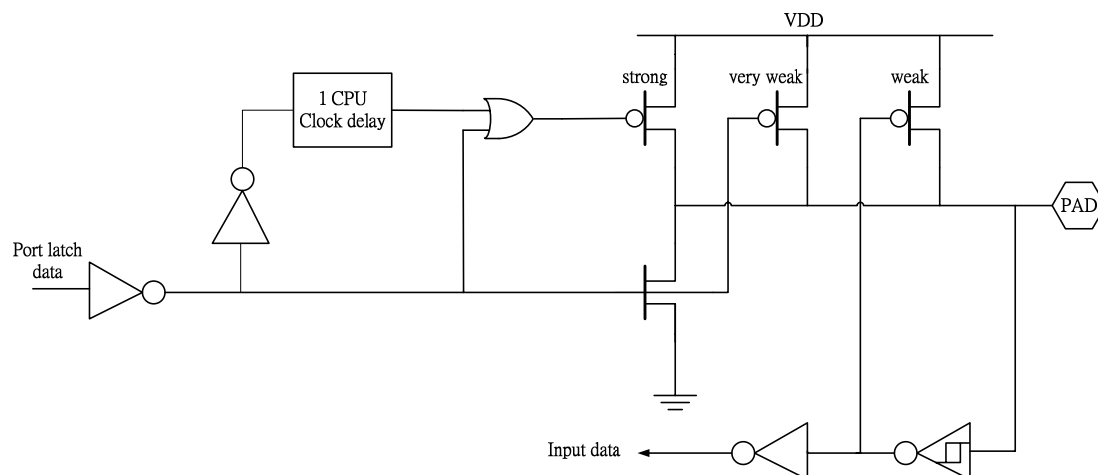


Fig. 1-1: IO pad structure

All the pads for P0 ~ P5 are with slew rate to reduce EMI. The other way to reduce EMI is to disable the ALE output if unused. This is selected by its SFR. The IO pads can withstand 4KV ESD in human body mode guaranteeing the SM59R16A2/SM59R08A2's quality in high electro-static environments.

1.3 2T/1T Selection

The conventional 52-series MCUs are 12T, i.e., 12 oscillator clocks per machine cycle. SM59R16A2/SM59R08A2 is a 2T or 1T MCU, i.e., its machine cycle is two-clock or one-clock. In the other words, it can execute one instruction within two clocks or only one clock. The difference between 2T mode and 1T mode are given in the example in Fig. 1-2.



Fig. 1-2(a): The waveform of internal instruction signal in 2T mode



Fig. 1-2(b): The waveform of internal instruction signal in 1T mode

The default is in 2T mode, and it can be changed to 1T mode if IFCON [7] (at address 8Fh) is set to high any time. Not every instruction can be executed with one machine cycle. The exact machine cycle number for all the instructions are given in the next section.

1.4 Reset

Brownout detection is also one type of internal reset to prevent SM59R16A2/SM59R08A2 from going to unstable condition as described in Section 1.3.

1.5 Clocks

The default clock is the 1MHz clock signal coming from the internal OSC. This clock is used during the initialization stage. The major work of the initialization stage is to determine the clock source used in normal operation.

The clock source can be external and internal. The external clock source is from the crystal via crystal pads XTAL1 and XTA2, or oscillator through XTAL1 only. Here we need to be aware that XTAL1 are not 5V tolerant in 3.3V application, so a 3.3V oscillator source is recommended.

The internal clock sources are from the internal OSC with difference frequency division as given in the next table:

Table 1-1: Selection of clock source

Clock source
external crystal or internal OSC
24MHz from internal OSC
20MHz from internal OSC
16MHz from internal OSC
12MHz from internal OSC
8MHz from internal OSC
4MHz from internal OSC
2MHz from internal OSC
1MHz from internal OSC as default clock used in initialization

There may be 20% variance in the frequency from the internal OSC. It is not recommended to use them in the application requiring accurate frequency.



2 Instruction Set

All SM59R16A2/SM59R08A2 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 SM59R16A2/SM59R08A2 Microcontroller core. Here the “cycles” in the tables means machine cycle, which is two-clock or one-clock depending on IFCON [7].

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	2
ADD A, @Ri	Add indirect RAM to accumulator	26-27	1	2
ADD A, #data	Add immediate data to accumulator	24	2	2
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	2
ADDC A, @Ri	Add indirect RAM to A with carry flag	36-37	1	2
ADDC A, #data	Add immediate data to A with carry flag	34	2	2
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	2
SUBB A, @Ri	Subtract indirect RAM from A with borrow	96-97	1	2
SUBB A, #data	Subtract immediate data from A with borrow	94	2	2
INC A	Increment accumulator	04	1	1
INC Rn	Increment register	08-0F	1	2
INC direct	Increment direct byte	05	2	3
INC @Ri	Increment indirect RAM	06-07	1	3
INC DPTR	Increment data pointer	A3	1	1
DEC A	Decrement accumulator	14	1	1
DEC Rn	Decrement register	18-1F	1	2
DEC direct	Decrement direct byte	15	2	3
DEC @Ri	Decrement indirect RAM	16-17	1	3
MUL AB	Multiply A and B	A4	1	5
DIV	Divide A by B	84	1	5
DAA	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	2
ANL A, @Ri	AND indirect RAM to accumulator	56-57	1	2
ANL A, #data	AND immediate data to accumulator	54	2	2
ANL direct, A	AND accumulator to direct byte	52	2	3
ANL direct, #data	AND immediate data to direct byte	53	3	4
ORL A, Rn	OR register to accumulator	48-4F	1	1
ORL A, direct	OR direct byte to accumulator	45	2	2
ORL A, @Ri	OR indirect RAM to accumulator	46-47	1	2
ORL A, #data	OR immediate data to accumulator	44	2	2
ORL direct, A	OR accumulator to direct byte	42	2	3
ORL direct, #data	OR immediate data to direct byte	43	3	4
XRL A, Rn	Exclusive OR register to accumulator	68-6F	1	1
XRL A, direct	Exclusive OR direct byte to accumulator	65	2	2
XRL A, @Ri	Exclusive OR indirect RAM to accumulator	66-67	1	2
XRL A, #data	Exclusive OR immediate data to accumulator	64	2	2
XRL direct, A	Exclusive OR accumulator to direct byte	62	2	3
XRL direct, #data	Exclusive OR immediate data to direct byte	63	3	4
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	2
MOV A, @Ri	Move indirect RAM to accumulator	E6-E7	1	2
MOV A, #data	Move immediate data to accumulator	74	2	2
MOV Rn, A	Move accumulator to register	F8-FF	1	2
MOV Rn, direct	Move direct byte to register	A8-AF	2	4
MOV Rn, #data	Move immediate data to register	78-7F	2	2
MOV direct, A	Move accumulator to direct byte	F5	2	3
MOV direct, Rn	Move register to direct byte	88-8F	2	3
MOV direct1, direct2	Move direct byte to direct byte	85	3	4
MOV direct, @Ri	Move indirect RAM to direct byte	86-87	2	4
MOV direct, #data	Move immediate data to direct byte	75	3	3
MOV @Ri, A	Move accumulator to indirect RAM	F6-F7	1	3
MOV @Ri, direct	Move direct byte to indirect RAM	A6-A7	2	5
MOV @Ri, #data	Move immediate data to indirect RAM	76-77	2	3
MOV DPTR, #data16	Load data pointer with a 16-bit constant	90	3	3
MOVC A, @A+DPTR	Move code byte relative to DPTR to accumulator	93	1	3
MOVC A, @A+PC	Move code byte relative to PC to accumulator	83	1	3
MOVX A, @Ri	Move Expanded RAM (8-bit addr.) to A	E2-E3	1	3
MOVX A, @DPTR	Move Expanded RAM (16-bit addr.) to A	E0	1	3
MOVX @Ri, A	Move A to Expanded RAM (8-bit addr.)	F2-F3	1	4
MOVX @DPTR, A	Move A to Expanded RAM (16-bit addr.)	F0	1	4
PUSH direct	Push direct byte onto stack	C0	2	4
POP direct	Pop direct byte from stack	D0	2	3
XCH A, Rn	Exchange register with accumulator	C8-CF	1	2
XCH A, direct	Exchange direct byte with accumulator	C5	2	3
XCH A, @Ri	Exchange indirect RAM with accumulator	C6-C7	1	3
XCHD A, @Ri	Exchange low-order nibble indir. RAM with A	D6-D7	1	3



Table 2-4: Program branches

Mnemonic	Description	Code	Bytes	Cycles
ACALL addr11	Absolute subroutine call	xxx11	2	6
LCALL addr16	Long subroutine call	12	3	6
RET	from subroutine	22	1	4
RETI	from interrupt	32	1	4
AJMP addr11	Absolute jump	xxx01	2	3
LJMP addr16	Long jump	02	3	4
SJMP rel	Short jump (relative addr.)	80	2	3
JMP @A+DPTR	Jump indirect relative to the DPTR	73	1	2
JZ rel	Jump if accumulator is zero	60	2	3
JNZ rel	Jump if accumulator is not zero	70	2	3
JC rel	Jump if carry flag is set	40	2	3
JNC	Jump if carry flag is not set	50	2	3
JB bit, rel	Jump if direct bit is set	20	3	4
JNB bit, rel	Jump if direct bit is not set	30	3	4
JBC bit, direct rel	Jump if direct bit is set and clear bit	10	3	4
CJNE A, direct rel	Compare direct byte to A and jump if not equal	B5	3	4
CJNE A, #data rel	Compare immediate to A and jump if not equal	B4	3	4
CJNE Rn, #data rel	Compare immediate to reg. and jump if not equal	B8-BF	3	4
CJNE @Ri, #data rel	Compare immediate to indirect and jump if not equal	B6-B7	3	4
DJNZ Rn, rel	Decrement register and jump if not zero	D8-DF	2	3
DJNZ direct, rel	Decrement direct byte and jump if not zero	D5	3	4
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	3
SETB C	Set carry flag	D3	1	1
SETB bit	Set direct bit	D2	2	3
CPL C	Complement carry flag	B3	1	1
CPL bit	Complement direct bit	B2	2	3
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	2
MOV bit, C	Move carry flag to direct bit	92	2	3



3 Memory Structure

The SM59R16A2/SM59R08A2 memory structure follows general 8052 structures. It manipulates operands in three memory spaces. They are (1) 256 bytes standard RAM, (2) 2K bytes auxiliary RAM, and (3) 64K/32K bytes embedded Flash as program memory.

3.1 Program Memory

The SM59R16A2/SM59R08A2 has 64KB/32KB on-chip Flash memory, which can be used as general program memory. If there is any byte not used as program memory, it can be used to record any data as EEPROM. The detailed way is given in Section 17.

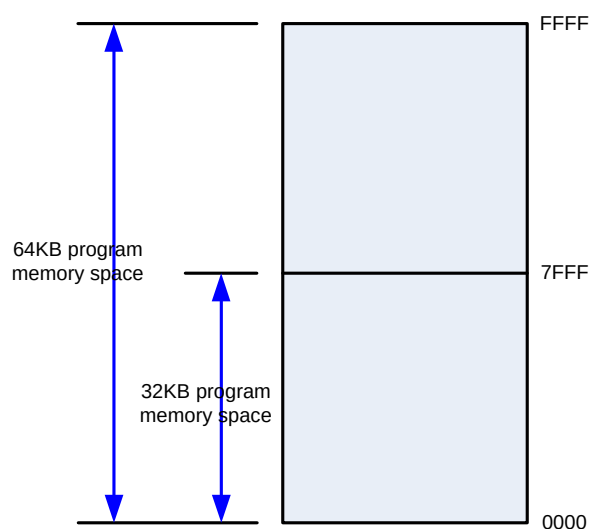


Fig. 3-1: 32KB/64KB programmable Flash

3.2 Data Memory

SM59R16A2/SM59R08A2 has 2048 + 256Bytes on-chip SRAM, the 256 bytes are the same as general 8052 internal memory structure. The expanded 2KB on-chip SRAM can be accessed by external memory addressing method (by instruction MOVX). As for 2KB – 64KB (Address 0800h – FFFFh) memory, they must be accessed as the external one through the interface similar to the conventional interface (P2, P0 are as the address and data bus, P3 [7:6] indicates read or write). If the SFR IFCON [1] = 1, this 2KB on-chip SRAM will be disabled and all the data memory are accessed externally. Even though this MCU is 2T or 1T, the external memory interface is still similar to the conventional 12T ways. An example is given below:

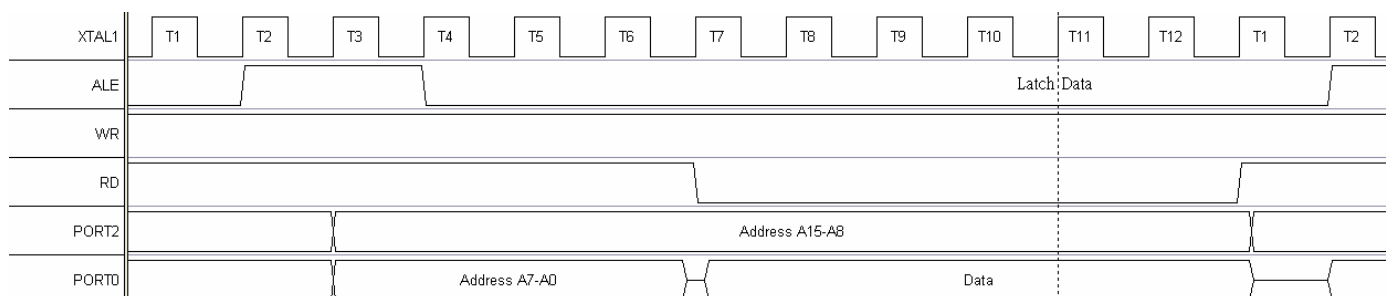


Fig 3-2 (a) : External memory access as read

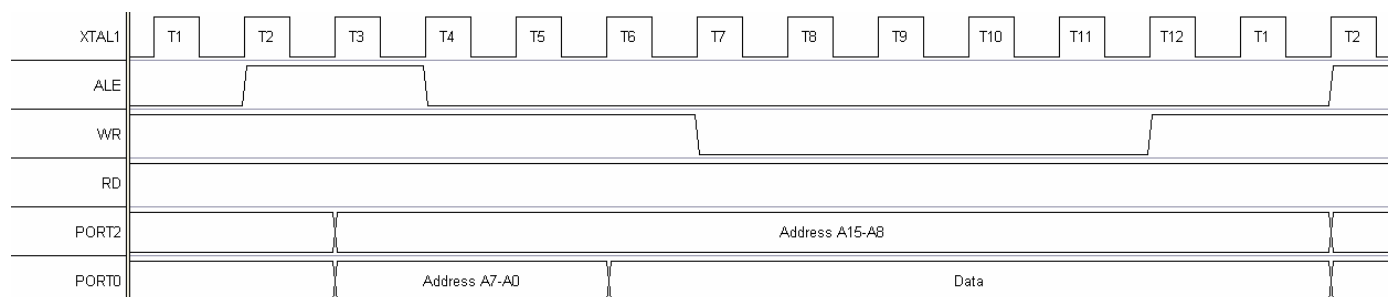


Fig 3-2 (b) : External memory access as write

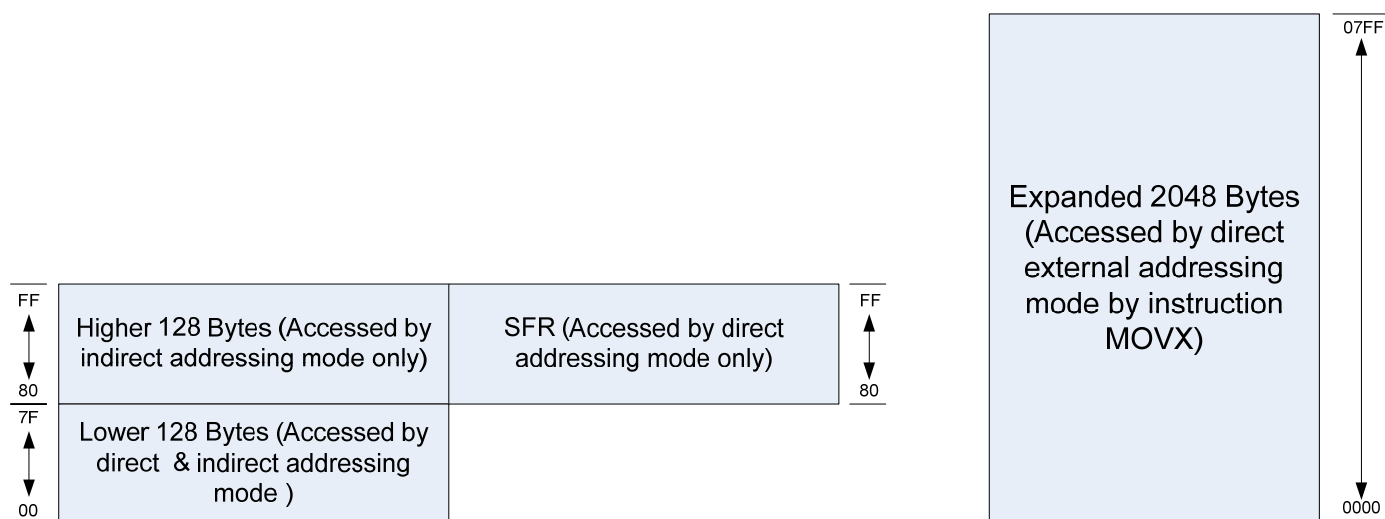


Fig. 3-3: RAM architecture

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

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



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

The address 80h to FFh can only be accessed by indirect addressing mode. It is data area.

3.2.3 Data memory - Expanded 2048 bytes (\$0000 to \$07FF)

From external address 0000h to 07FFh is the on-chip expanded SRAM area, total 2048 Bytes. This area can be accessed by external direct addressing mode (by instruction MOVX).

If the address of instruction MOVX @DPTR is larger than 07FFh, the SM59R16A2 will generate the external memory control signal automatically.

If the SFR IFCON [1] = 1, this 2KB on-chip SRAM will be disabled as if there is no such embedded memory. The default value for IFCON [1] is 0.



4 CPU Engine

The SM59R16A2/SM59R08A2 engine is composed of four components:

- Control unit
- Arithmetic – logic unit
- Memory control unit
- RAM and SFR control unit

The SM59R16A2/SM59R08A2 engine allows to fetch instruction from program memory and to execute using RAM or SFR. The following paragraphs describe the main engine registers.

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
CPU 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	F1	P	00h
SP	Stack Pointer	81h	SP [7:0]								07h
DPL	Data pointer low 0	82h	DPL [7:0]								00h
DPH	Data pointer high 0	83h	DPH [7:0]								00h
DPL1	Data pointer low 1	84h	DPL1 [7:0]								00h
DPH1	Data pointer high 1	85h	DPH1 [7:0]								00h
DPS	Data pointer select	92h	-	-	-	-	-	-	-	DPS.0	00h
IFCON	Interface control register	8Fh	ITS	-	-	-	ALEC[1:0]		DMEN	-	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	ACC.5	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 0

Mnemonic: DPH							Address: 83h	
7	6	5	4	3	2	1	0	Reset
DPH [7:0]								00h

DPH [7:0]: Data pointer High 0



4.6 Data Pointer 1

The dual data pointer accelerates the moving of block data. The standard DPTR is a 16-bit register that is used to address external memory or peripherals. In the SM59R16A2/SM59R08A2, the standard data pointer is called DPTR, the second data pointer is called DPTR1. The data pointer select bit chooses the active pointer. The data pointer select bit is located in the LSB of DPS register (DPS.0).

The user switches the pointer between PDTR and DPTR1 by toggling the LSB of DPS register. All DPTR-related instructions use the currently selected DPTR for any activity.

Mnemonic: DPL1							Address: 84h	
7	6	5	4	3	2	1	0	Reset
DPL1 [7:0]								00h

DPL1[7:0]: Data pointer Low 1

Mnemonic: DPH1							Address: 85h	
7	6	5	4	3	2	1	0	Reset
DPH1 [7:0]								00h

DPH1[7:0]: Data pointer High 1

Mnemonic: DPS							Address: 92h	
7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	-	DPS.0	00h

DPS.0: Data Pointer selects register.
DPS.0 = 1 is selected DPTR1.

4.7 Interface control register

Mnemonic: IFCON							Address: 8Fh	
7	6	5	4	3	2	1	0	Reset
ITS	-	-	-	ALEC[1:0]		DMEN	-	00h

ITS: Instruction timing select.
ITS = 0, 2T instruction mode.
ITS = 1, 1T instruction mode.

ALEC[1:0]: ALE output control register.

ALEC[1:0]	ALE Output
00	Always output
01	No ALE output
10	Only Read or Write have ALE output
11	reserved

DMEN: Internal 2K SRAM disable.(default is enable)
DMEN = 0, Enable internal 2K RAM.
DMEN = 1, Disable internal 2K RAM.



5 Port0 – Port 5

Port 0 ~ Port 5 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. Port0 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 as given below:

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
Ports											
Port 5	Port 5	91h	P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0	FFh
Port 4	Port 4	E8h	P4.7	P4.6	P4.5	P4.4	P4.3	P4.2	P4.1	P4.0	FFh
Port 3	Port 3	B0h	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	FFh
Port 2	Port 2	A0h	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	FFh
Port 1	Port 1	90h	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	FFh
Port 0	Port 0	80h	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	FFh

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: E8h

7	6	5	4	3	2	1	0	Reset
P4.7	P4.6	P4.5	P4.4	P4.3	P4.2	P4.1	P4.0	FFh

P4.7~ 0: Port4 [7] ~ Port4 [0]

Mnemonic: P5

Address: 91h

7	6	5	4	3	2	1	0	Reset
P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0	FFh

P5.7~ 0: Port5 [7] ~ Port5 [0]



6 Multiplication Division Unit (MDU)

This on-chip arithmetic unit provides 32-bit division, 16-bit multiplication, shift and normalize features, etc. All operations are unsigned integer operations.

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
Multiplication Division Unit											
PCON	Power control	87h	SMOD	MDUF	-	PMW	-	-	STOP	IDLE	00h
ARCON	Arithmetic Control register	EFh	MDEF	MDOV	SLR	SC [4:0]					00h
MD0	Multiplication/Div ision Register 0	E9h	MD0 [7:0]								00h
MD1	Multiplication/Div ision Register 1	EAh	MD1 [7:0]								00h
MD2	Multiplication/Div ision Register 2	EBh	MD2 [7:0]								00h
MD3	Multiplication/Div ision Register 3	ECh	MD3 [7:0]								00h
MD4	Multiplication/Div ision Register 4	EDh	MD4 [7:0]								00h
MD5	Multiplication/Div ision Register 5	EEh	MD5 [7:0]								00h

6.1 Operation of the MDU

The operation of the MDU consists of three phases:

6.1.1 First phase: loading the MDx registers, x = 0~5:

The type of calculation the MDU has to perform is selected by the order in which the MDx registers are written to. A write to MD0 is the first transfer to be done in any case. Next writes must be done as shown in table below to determine MDU operation. The last write will start the selected operation.

Table 6-1: MDU registers write sequence

Operation	32bit/16bit	16bit/16bit	16bit x 16bit	shift/normalizing
First write	MD0 Dividend Low	MD0 Dividend Low	MD0 Multiplicand Low	MD0 LSB
	MD1 Dividend	MD1 Dividend High	MD4 Multiplier Low	MD1
	MD2 Dividend		MD1 Multiplicand High	MD2
	MD3 Dividend High			MD3 MSB
	MD4 Divisor Low	MD4 Divisor Low		
Last write	MD5 Divisor High	MD5 Divisor High	MD5 Multiplier High	ARCON start conversion



6.1.2 Second phase: executing calculation.

During executing operation, the MDU works on its own parallel to the CPU. When MDU is finished, the MDUF register will be set to one by hardware and the flag will be cleared at the next calculation.

Mnemonic: PCON						Address: 87h		
7	6	5	4	3	2	1	0	Reset
SMOD	MDUF	-	PMW	-	-	STOP	IDLE	00h

MDUF: MDU finish flag.

When MDU is finished, the MDUF will be set by hardware and the bit will clear by hardware at next calculation.

The following table gives the execution time in every mathematical operation.

Table 6-2: MDU execution times

Operation	Number of Tclk
Division 32bit/16bit	17 clock cycles
Division 16bit/16bit	9 clock cycles
Multiplication	11 clock cycles
Shift	Min. 3 clock cycles, Max. 18 clock cycles
Normalize	Min. 4 clock cycles, Max. 19 clock cycles

6.1.3 Third phase: reading the result from the MDx registers.

The sequence of reading out the first MDx registers is not critical, but we have to be aware that the last read (from MD5 in division operation, or MD3 by multiplication, shift and normalizing) means the end of a whole calculation.

Table 6-3: MDU registers read sequence

Operation	32Bit/16Bit	16Bit/16Bit	16Bit x 16Bit	shift/normalizing
First read	MD0 Quotient Low	MD0 Quotient Low	MD0 Product Low	MD0 LSB
	MD1 Quotient	MD1 Quotient High	MD1 Product	MD1
	MD2 Quotient		MD2 Product	MD2
	MD3 Quotient High			
	MD4 Remainder L	MD4 Remainder Low		
Last read	MD5 Remainder H	MD5 Remainder High	MD3 Product High	MD3 MSB

Here the operation of normalization and shift will be explained more. In normalization, all reading zeroes in registers MD0 to MD3 are removed by shift left. The whole operation is completed when the MSB (most significant bit) of MD3 register contains a '1'. After normalizing, bits ARCON.4 (MSB) to ARCON.0 (LSB) contain the number of shift left operations. As for shift, SLR bit (ARCON.5) has to contain the shift direction, and ARCON.4 to ARCON.0 represent the shift count (which must not be 0). During shift, zeroes come into the left or right end of the registers MD0 or MD3, respectively.



6.2 Operating registers

The MDU is handled by seven registers, which are memory mapped as special function registers. The arithmetic unit allows operations concurrently to and independent of the CPU's activity.

Operands and results registers are MD0 to MD5, and the control register is ARCON.

Any calculation of the MDU will overwrite its operands.

Mnemonic: ARCON							Address: EFh
7	6	5	4	3	2	1	0 Reset
MDEF	MDOV	SLR	SC [4:0]				00h

MDEF: Multiplication Division Error Flag.

The MDEF is an error flag. The error flag is read only. The error flag indicates an improperly performed operation (when one of the arithmetic operations has been restarted or interrupted by a new operation). The error flag mechanism is automatically enabled with the first write to MD0 and disabled with the final read instruction from MD3 (multiplication or shift/normalizing) or MD5 (division) in the third phase.

The error flag is set when:

1. The second phase in process and write access to MDx registers (restart or interrupt calculations)

The error flag is reset only if:

The second phase finished (arithmetic operation successful completed) and read access to MDx registers.

MDOV: Multiplication Division Overflow flag. The overflow flag is read only.

The overflow flag is set when:

Divided by zero

Multiplication with a result greater than 0000FFFFh

Start of normalizing if the most significant bit of MD3 is set (MD3.7=1)

The overflow flag is reset when:

Write access to MD0 register (start the first phase)

SLR: Shift direction bit.

SLR = 0 – shift left operation.

SLR = 1 – shift right operation.

SC [4:0]: Shift counter.

When preset with 00000b, normalizing is selected. After normalized, SC[4:0] contains the number of normalizing shifts performed. When SC[4:0] $\neq$ 0, shift operation is started. The number of shifts performed is determined by the count written to SC [4:0]. SC [4] is MSB and SC[0] is LSB.



7 Timer 0 and Timer 1

SM59R16A2/SM59R08A2 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 12 machines cycles, which means that it counts up after every 12 periods of the crystal or oscillator signal.

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 to ensure proper recognition of 0 or 1 state, so an input should be stable for at least 1 machine cycle.

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

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
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



7.1 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.

M[1:0]: Selects mode for Timer/Counter 0 or Timer/Counter 1.

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.

7.2 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.

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.



8 Timer 2 and Capture/Compare Unit

Timer 2 is not only a 16-bit timer, also a 4-channel unit with compare, capture and reload functions. It is very similar to the programmable counter array (PCA) in some other MCUs except pulse width modulation (PWM).

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
Timer 2 and Capture Compare Unit											
T2CON	Timer 2 control	C8h	T2PS	CC0FR	-	T2R[1:0]		T2CM	T2I[1:0]		00h
CCEN	Compare/Capture Enable register	C1h	COCA H3	COCA L3	COC AH2	COCA L2	COCA H1	COCA L1	COCA H0	COC AL0	00h
TL2	Timer 2, low byte	CCh	TL2[7:0]								00h
TH2	Timer 2, high byte	CDh	TH2[7:0]								00h
CRCL	Compare/Reload/Capture register, low byte	CAh	CRCL[7:0]								00h
CRCH	Compare/Reload/Capture register, high byte	CBh	CRCH[7:0]								00h
CCL1	Compare/Capture register 1, low byte	C2h	CCL1[7:0]								00h
CCH1	Compare/Capture register 1, high byte	C3h	CCH1[7:0]								00h
CCL2	Compare/Capture register 2, low byte	C4h	CCL2[7:0]								00h
CCH2	Compare/Capture register 2, high byte	C5h	CCH2[7:0]								00h
CCL3	Compare/Capture register 3, low byte	C6h	CCL3[7:0]								00h
CCH3	Compare/Capture register 3, high byte	C7h	CCH3[7:0]								00h



Mnemonic: T2CON

Address: C8h

7	6	5	4	3	2	1	0	Reset
T2PS	CC0FR	-	T2R[1:0]		T2CM	T2I[1:0]		00h

T2PS: Prescaler select bit:

T2PS = 0 – timer 2 is clocked with 1/12 of the oscillator frequency.

T2PS = 1 – timer 2 is clocked with 1/24 of the oscillator frequency.

CC0FR: Select active edge:

CC0FR = 0 – falling edge

CC0FR = 1 – rising edge

T2R[1:0]: Timer 2 reload mode selection

T2R[1:0] = 0X – Reload disabled

T2R[1:0] = 10 – Mode 0

T2R[1:0] = 11 – Mode 1

T2CM: Timer 2 Compare mode selection

T2CM = 0 – Mode 0

T2CM = 1 – Mode 1

T2I[1:0]: Timer 2 input selection

T2I[1:0] = 00 – Timer 2 stop

T2I[1:0] = 01 – Input frequency f/12 or f/24

T2I[1:0] = 10 – Timer 2 is incremented by external signal at pin T2

T2I[1:0] = 11 – internal clock input is gated to the Timer 2

Mnemonic: CCEN

Address: C1h

7	6	5	4	3	2	1	0	Reset
COCAH3	COCAL3	COCAH2	COCAL2	COCAH1	COCAL1	COCAH0	COCAL0	00h

COCAH3,COCAL3: Compare/capture mode for Channel 3.

COCAH3	COCAL3	Function
0	0	Compare/capture disable
0	1	Capture on rising edge at pin CC3
1	0	Compare enable
1	1	Capture on write operation into register CCL3

COCAH2,COCAL2: Compare/Capture mode for Channel 2.

COCAH3	COCAL3	Function
0	0	Compare/capture disable
0	1	Capture on rising edge at pin CC2
1	0	Compare enable
1	1	Capture on write operation into register CCL2

COCAH1,COCAL1: Compare/Capture mode for Channel 1.

COCAH1	COCAL1	Function
0	0	Compare/capture disable
0	1	Capture on rising edge at pin CC1
1	0	Compare enable
1	1	Capture on write operation into register CCL1

COCAH0,COCAL0: Compare/Capture mode for CRC register (Channel 0)

COCAH3	COCAL3	Function
0	0	Compare/capture disable
0	1	Capture on falling/rising edge at pin CC0
1	0	Compare enable
1	1	Capture on write operation into register CRCL



8.1 Timer 2 function

Timer 2 can operate as timer, event counter, or gated timer as explained later.

8.1.1 Timer mode

In this mode Timer 2 can be incremented in every 12 machine cycles or in every 24 machine cycles depending on the 2:1 prescaler. The prescaler is selected by bit T2PS in register T2CON.

8.1.2 Event counter mode

In this mode, the timer is incremented when external signal T2 change value from 1 to 0. The T2 input is sampled in every cycle. Timer 2 is incremented in the cycle following the one in which the transition was detected.

8.1.3 Gated timer mode

In this mode, the internal clock which incremented timer 2 is gated by external signal T2.

8.1.4 Reload of Timer 2

Reload (16-bit reload from the crc register) can be executed in the following two modes:

Mode 0: Reload signal is generate by a Timer 2 overflows - auto reload

Mode 1: Reload signal is generate by a negative transition at the corresponding input pin T2EX.

8.2 Compare function

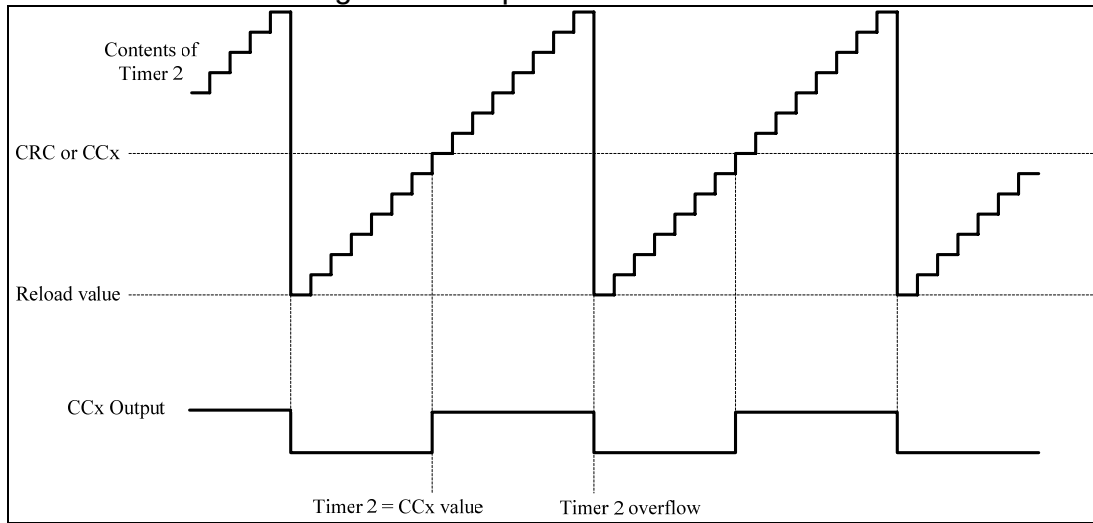
In the four independent comparators, the value stored in any compare/capture register is compared with the contents of the timer register. The compare modes 0 and 1 are selected by bit T2CM. In both compare modes, the results of comparison arrives at Port 1 within the same machine cycle in which the internal compare signal is activated. The port pins P1.2 to P1.5 are the outputs of CC0 to CC3.

8.2.1 Compare Mode 0

In mode 0, when the value in Timer 2 equals the value of the compare register, the output signal changes from low to high. It goes back to a low level on timer overflow. In this mode, writing to the port will have no effect, because the input line from the internal bus and the write-to-latch line are disconnected. The following figure illustrates the function of compare mode 0.



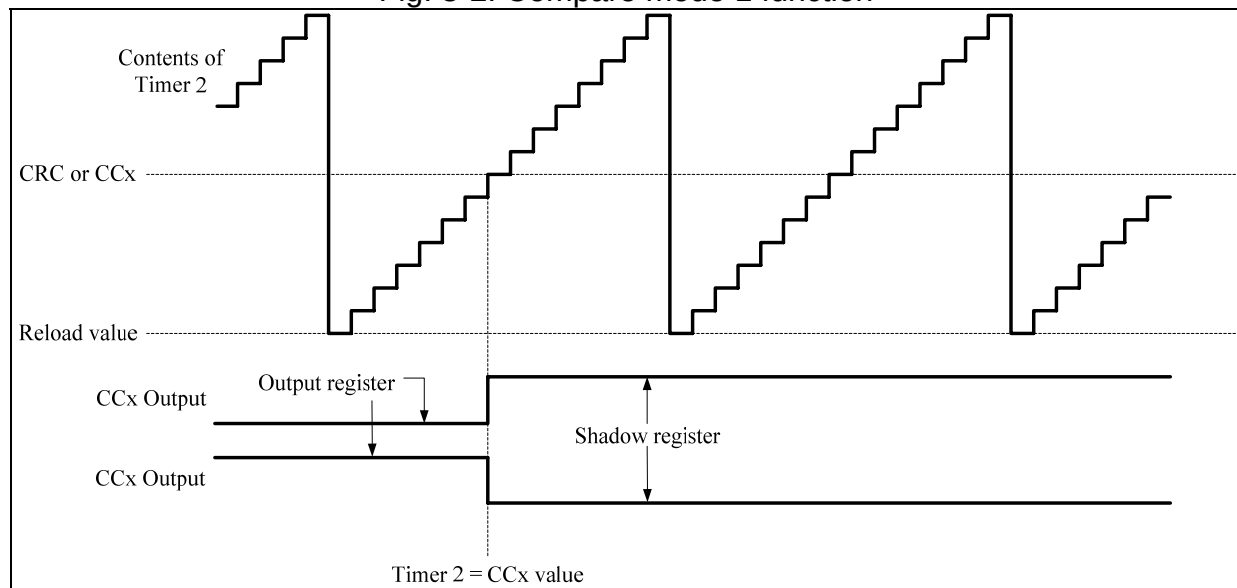
Fig. 8-1: Compare mode 0 function



8.2.2 Compare Mode 1

In compare mode 1, the transition of the output signal can be determined by software. A timer 2 overflow causes no output change. In this mode, both transitions of a signal can be controlled. Fig. 8-2 shows a functional diagram of a register/port configuration in compare Mode 1. In compare Mode 1, the value is written first to the "Shadow Register", when compare signal is active, this value is transferred to the output register.

Fig. 8-2: Compare mode 1 function





8.3 Capture function

Actual timer/counter contents can be saved into registers CCx or CRC upon an external event (mode 0) or a software write operation (mode 1).

8.3.1 Capture Mode 0

In mode 0, value capture of Timer 2 is executed when:

(a) rising edge on input CC1-CC3

(b) rising or falling edge on input CC0 (depending on bit CC0FR)

The contents of Timer 2 will be latched into the appropriate capture register. In this mode, no interrupt request will be generated.

8.3.2 Capture Mode 1

In mode 1, value capture of timer 2 is caused by writing any value into the low-order byte of the dedicated capture register. The value written to the capture register is irrelevant to this function. The contents of Timer 2 will be latched into the appropriate capture register. In this mode, no interrupt request will be generated.



9 Serial interface 0 and 1

There are two serial interfaces for data communication in SM59R16A2/SM59R08A2, they are the so called UART0 and UART1. As the conventional UART, the communication speed can be selected by configuring the baud rate in SFRs. These two serial buffers consists of two separate registers, a transmit buffer and a receive buffer. Writing data to the SFR S0BUF or S1BUF sets this data in serial output buffer and starts the transmission. Reading from the S0BUF or S1BUF 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 second byte before the transmission of the first byte is completed.

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
Serial interface 0 and 1											
PCON	Power control	87h	SMOD	MDUF	-	PMW	-	-	STOP	IDLE	00h
BRGS	Baud rate generator switch	D8h	BRS	-	-	-	-	-	-	-	00h
S0CON	Serial Port 0 control register	98h	SM0	SM1	SM20	REN0	TB80	RB80	TI0	RI0	00h
S0RELL	Serial Port 0 reload register low byte	AAh	S0REL .7	S0REL .6	S0REL .5	S0REL .4	S0REL .3	S0REL .2	S0REL .1	S0REL .0	00h
S0RELH	Serial Port 0 reload register high byte	BAh	-	-	-	-	-	-	S0REL .9	S0REL .8	00h
S0BUF	Serial Port 0 data buffer	99h	S0BUF[7:0]								00h
S1CON	Serial Port 1 control register	9Bh	SM	-	SM21	REN1	TB81	RB81	TI1	RI1	00h
S1RELL	Serial Port 1 reload register low byte	9Dh	S1REL .7	S1REL .6	S1REL .5	S1REL .4	S1REL .3	S1REL .2	S1REL .1	S1REL .0	00h
S1RELH	Serial Port 1 reload register high byte	BBh	-	-	-	-	-	-	S1REL .9	S1REL .8	00h
S1BUF	Serial Port 1 data buffer	9Ch	S1BUF[7:0]								00h



Mnemonic: S0CON

Address: 98h

7	6	5	4	3	2	1	0	Reset
SM0	SM1	SM20	REN0	TB80	RB80	TI0	RI0	00h

SM0, SM1: Serial Port 0 mode selection.

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

The 4 modes in UART0, Mode 0 ~ 3, are explained later.

SM20: Enables multiprocessor communication feature

REN0: If set, enables serial reception. Cleared by software to disable reception.

TB80: 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.

RB80: In modes 2 and 3, it is the 9th data bit received. In mode 1, if SM20 is 0, RB80 is the stop bit. In mode 0, this bit is not used. Must be cleared by software.

TI0: Transmit interrupt flag, set by hardware after completion of a serial transfer. Must be cleared by software.

RI0: Receive interrupt flag, set by hardware after completion of a serial reception. Must be cleared by software.

Mnemonic: S1CON

Address: 9Bh

7	6	5	4	3	2	1	0	Reset
SM	-	SM21	REN1	TB81	RB81	TI1	RI1	00h

SM: Serial Port 1 mode select.

SM	Mode
0	A
1	B

The 2 modes in UART1, Mode A and Mode B, are explained later.

SM21: Enables multiprocessor communication feature.

REN1: If set, enables serial reception. Cleared by software to disable reception.

TB81: The 9th transmitted data bit in mode A. Set or cleared by the CPU depending on the function it performs such as parity check, multiprocessor communication etc.

RB81: In mode A, it is the 9th data bit received. In mode B, if SM21 is 0, RB81 is the stop bit. Must be cleared by software.

TI1: Transmit interrupt flag, set by hardware after completion of a serial transfer. Must be cleared by software.

RI1: Receive interrupt flag, set by hardware after completion of a serial reception. Must be cleared by software.



9.1 Serial interface 0

The Serial Interface 0 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.

9.1.1 Mode 0

Pin RXD0 serves as input and output. TXD0 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 S0CON as follows: RI0 = 0 and REN0 = 1. In the other modes, a start bit when REN0 = 1 starts receiving serial data.

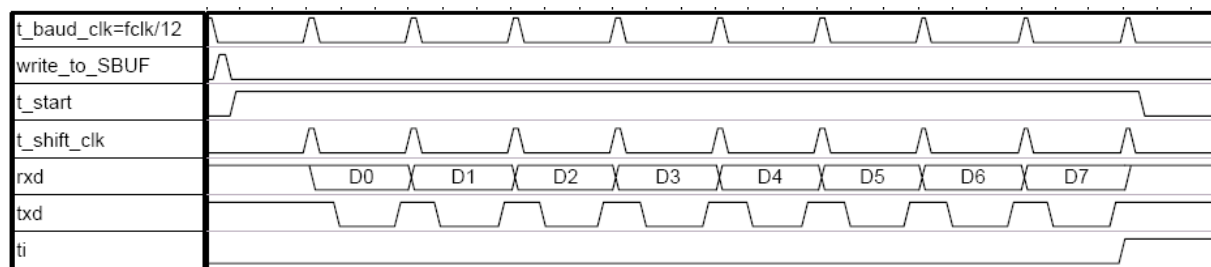


Fig. 9-1: Transmit mode 0 for Serial 0

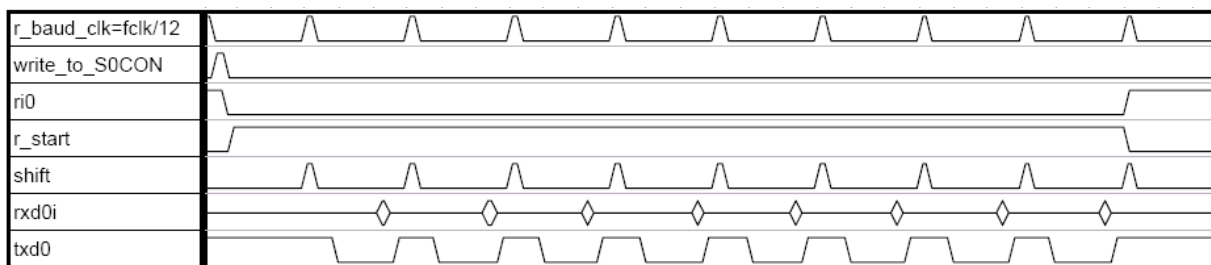


Fig. 9-2: Receive mode 0 for Serial 0

9.1.2 Mode 1

Here Pin RXD0 serves as input, and TXD0 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 S0BUF, and a stop bit sets the flag RB80 in the SFR S0CON. In mode 1, either internal baud rate generator or timer 1 can be used to specify the desired baud rate.



Fig. 9-3: Transmit mode 1 for Serial 0

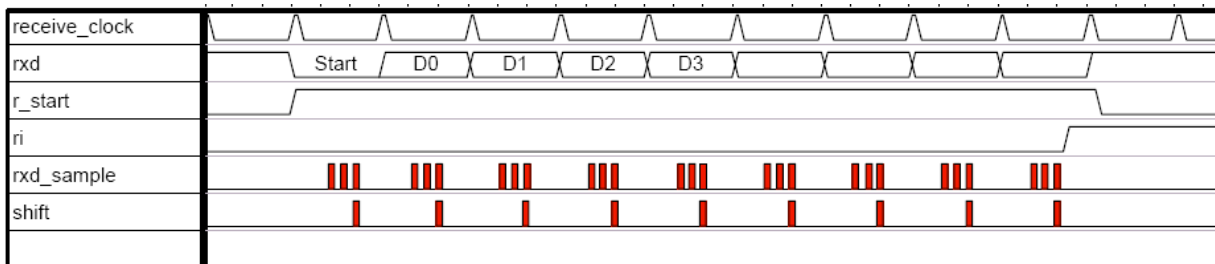


Fig. 9-4: Receive mode 1 for Serial 0

9.1.3 Mode 2

This mode is similar to Mode 1, but 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 Bit 9, and a stop bit (1). Bit 9 can be used to control the parity of the serial interface: at transmission, bit TB80 in S0CON is output as Bit 9, and at receive, Bit 9 affects RB80 in SFR S0CON.

9.1.4 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.

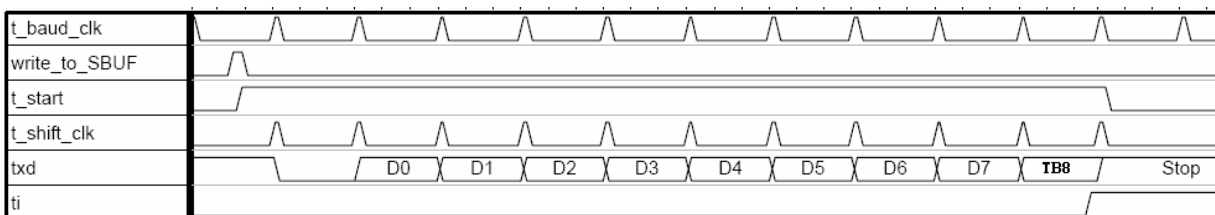


Fig. 9-5: Transmit modes 2 and 3 for Serial 0

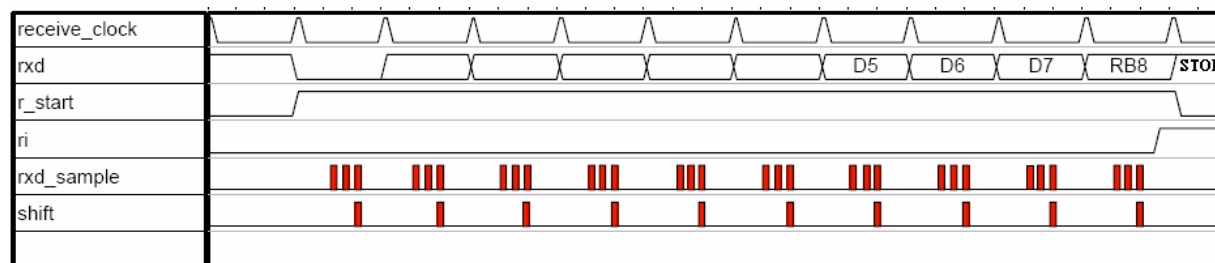


Fig. 9-6: Receive modes 2 and 3 for Serial 0



9.2 Serial interface 1

The Serial Interface 1 can operate in the following 2 modes:

SM	Mode	Description	Baud Rate
0	A	9-bit UART	Variable
1	B	8-bit UART	Variable

9.2.1 Mode A

This mode is similar to Mode 2 and 3 of Serial interface 0, 11 bits are transmitted or received: a start bit (0), 8 data bits (LSB first), a programmable Bit 9, and a stop bit (1). Bit 9 can be used to control the parity of the serial interface: at transmission, bit TB81 in S1CON is outputted as Bit 9, and at receive, Bit 9 affects RB81 in SFR S1CON.



Fig. 9-7: Transmit mode A for Serial 1

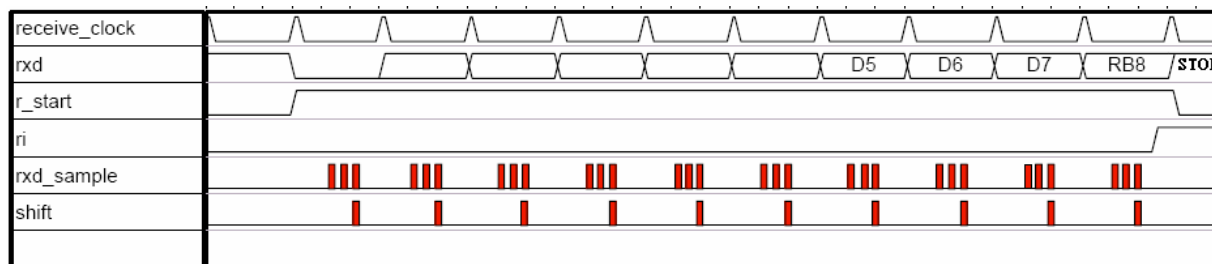


Fig. 9-8: Receive mode A for Serial 1

9.2.2 Mode B

This mode is similar to Mode 1 of Serial interface 0. Pin RXD1 serves as input, and TXD1 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 S1BUF, and stop bit sets the flag RB81 in the SFR S1CON. In mode B, internal baud rate generator is use to specify the baud rate.



Fig. 9-9: Transmit mode B for Serial 1

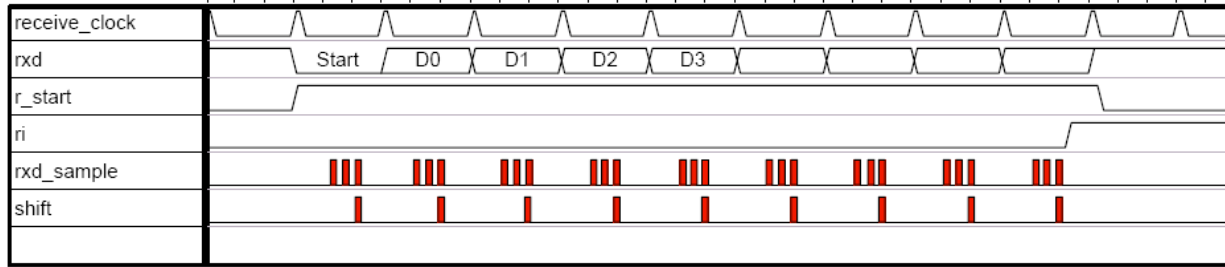


Fig. 9-10: Receive mode B for Serial 1

9.3 Multiprocessor communication of Serial Interface 0 and 1

The feature of receiving 9 bits in Modes 2 and 3 of Serial Interface 0 or in Mode A of Serial Interface 1 can be used for multiprocessor communication. In this case, the slave processors have bit SM20 in S0CON or SM21 in S1CON set to 1. When the master processor outputs slave's address, it sets the Bit 9 to 1, causing a serial port receive interrupt in all the slaves. The slave processors compare the received byte with their network address. If matched, the addressed slave will clear SM20 or SM21 and receive the rest of the message, while other slaves will leave SM20 or SM21 bit unaffected and ignore this message. After addressing the slave, the host will output the rest of the message with the Bit 9 set to 0, so no serial port receive interrupt will be generated in unselected slaves.

9.4 Baud rate generator

9.4.1 Serial interface 0 modes 1 and 3

(a) When BRS = 0 (in SFR BRGS):

$$\text{Baud Rate} = \frac{2^{\text{SMOD}} \times F_{\text{OSC}}}{32 \times 12 \times (256 - \text{TH1})}$$

(b) When BRS = 1 (in SFR BRGS):

$$\text{Baud Rate} = \frac{2^{\text{SMOD}} \times F_{\text{OSC}}}{64 \times (2^{10} - \text{S0REL})}$$

9.4.2 Serial interface 1 modes A and B

$$\text{Baud Rate} = \frac{F_{\text{OSC}}}{32 \times (2^{10} - \text{S1REL})}$$

9.5 Clock source for baud rate

It is not recommended to use the internal OSC as the clock source when the serial interface functions are used. The reason is that the baud rate in the previous section must be as accurate as possible. The internal OSC clock frequency may be varied with $\pm 5\%$. So the user can choose the clock source from external crystal or oscillator.



10 Watchdog timer

The watchdog timer is an 8-bit counter that is incremented once every WDTCLK clock cycles. After an external reset, the watchdog timer is disabled and all registers are set to zeros.

During the initialization period, CPU read the WDTENB and WDTM[3:0] in information block. WDTENB is the disable bit. When this bit is high, the watchdog function will be disabled. The WDTM[3:0] is to set the frequency division for WDTCLK as shown in the figure below. User can to set WDTENB and WDTM[3:0] through the writer.

$$\text{WDTCLK} = \frac{F_{\text{osc}}}{12 \times 2^{\text{WDTM}}}$$
$$\text{Watchdog reset time} = \frac{256}{\text{WDTCLK}}$$

Once the watchdog is started it cannot be stopped. User can refreshed the watchdog timer to zero when WDTK register is written by 55h.

When Watchdog timer is overflow, the WDTF flag will set to one and automatically reset MCU. The WDTF flag can be clear by software or external reset.

The watchdog timer must be refreshed regularly to prevent reset request signal from becoming active.

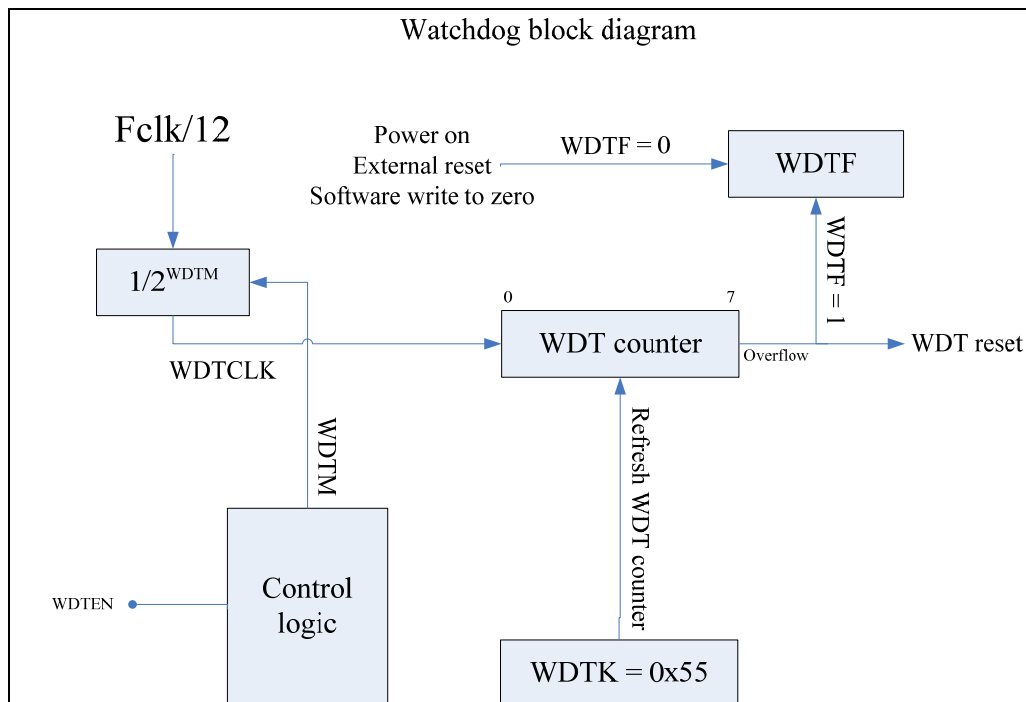


Fig. 10-1: Watchdog timer block diagram



Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
Watchdog Timer											
WDTC	Watchdog timer control register	B6h	WDTF	-	-	-	-	-	-	-	00h
WDTK	Watchdog timer refresh key	B7h	WDTK[7:0]								00h

Mnemonic: WDTC

Address: B6h

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

WDTF: Watchdog timer reset flag.

When MCU is reset by watchdog, WDTF flag will be set to one by hardware.

This flag is cleared by software or external reset.

Mnemonic: WDTK

Address: B7h

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

WDTK: Watchdog timer refresh key.

A programmer must write 0x55 into WDTK register, then the watchdog timer will be cleared to zero.



11 Interrupt

SM59R16A2/SM59R08A2 provides 11 interrupt sources with four priority levels. Each source has its own request flag located in a SFR. Each interrupt requested by the corresponding flag can be enabled or disabled individually by the enable bits in SFR's IEN0, IEN1, and IEN2.

When the interrupt occurs, the CPU will vector to the predetermined address as shown in Table 11-1. 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 the next instruction when the 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, 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 the processor when the interrupt occurs. If the processor is performing an interrupt service with equal or greater priority, the new interrupt will not be invoked. In the other cases, the response time depends on current instruction. The fastest possible response to an interrupt is 7 machine cycles. This includes one machine cycle for detecting the interrupt and six cycles for perform the LCALL.

Table 11-1: Interrupt vectors

Interrupt Request Flags	Interrupt Vector Address	Interrupt Number *(use Keil C Tool)
IE0 – External interrupt 0	0003h	0
TF0 – Timer 0 interrupt	000Bh	1
IE1 – External interrupt 1	0013h	2
TF1 – Timer 1 interrupt	001Bh	3
RI0/TI0 – Serial channel 0 interrupt	0023h	4
TF2/EXF2 – Timer 2 interrupt	002Bh	5
SPIIF – SPI interrupt	004Bh	9
ADCIF – A/D converter interrupt	0053h	10
EEIIF – Expanded External Interrupt	005Bh	11
IICIF – IIC interrupt	006Bh	13
RI1/TI1 – Serial channel 1 interrupt	0083h	16

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



Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
Interrupt											
IEN0	Interrupt Enable 0 register	A8h	EA	-	ET2	ES0	ET1	EX1	ET0	EX0	00h
IEN1	Interrupt Enable 1 register	B8h	EXEN2	-	IEIIC	-	IEEEI	IEADC	IESPI	-	00h
IEN2	Interrupt Enable 2 register	9Ah	-	-	-	-	-	-	-	ES1	00h
IP0	Interrupt priority level 0	A9h	-	-	IP0.5	IP0.4	IP0.3	IP0.2	IP0.1	IP0.0	00h
IP1	Interrupt priority level 1	B9h	-	-	IP1.5	IP1.4	IP1.3	IP1.2	IP1.1	IP1.0	00h

Interrupt Enable 0 register(IEN0)

Mnemonic: IEN0								Address: A8h	
7	6	5	4	3	2	1	0	Reset	
EA	-	ET2	ES0	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.

ES0: ES0 = 0 : disable Serial channel 0 interrupt.

ET1: ET1 = 0 : disable Timer 1 overflow interrupt.

EX1: EX1 = 0 : disable external interrupt 1.

ET0: ET0 = 0 : disable Timer 0 overflow interrupt.

EX0: EX0 = 0 : disable external interrupt 0.

Interrupt Enable 1 register(IEN1)

Mnemonic: IEN1								Address: B8h	
7	6	5	4	3	2	1	0	Reset	
EXEN2	-	IEIIC	-	IEEEI	IEADC	IESPI		00h	

EXEN2: Timer 2 reload interrupt enable

EXEN2 = 0 : disable Timer 2 external reload interrupt.

IEIIC: IIC interrupt enable.

IEIICS = 0 : disable IIC interrupt.

IEEEI: EEI interrupt enable

IEEEI = 0 : disable EEI interrupt

IEADC : A/D converter interrupt enable

IEADC = 0 : disable ADC interrupt.

IESPI: SPI interrupt enable.

IESPI = 0 : disable SPI interrupt.

Interrupt Enable 2 register(IEN2)

Mnemonic: IE2								Address: 9Ah	
7	6	5	4	3	2	1	0	Reset	
-	-	-	-	-	-	-	ES1	00h	

ES1: ES1=0 – Disable Serial channel 1 interrupt.



Interrupt request register(IRCON)

Mnemonic: IRCON							Address: C0h	
7	6	5	4	3	2	1	0	Reset
EXF2	TF2	IICIF	-	EEIIF	ADCIF	SPIIF		00h

EXF2: Timer 2 external reload flag, must be cleared by software.

TF2: Timer 2 overflow flag, must be cleared by software.

IICIF: IIC interrupt flag must be cleared after the RxIF and TxIF at IICS register clear by software.

EEIIF: EEI interrupt flag, must be cleared by software.

ADCIF: A/D converter interrupt flag, must be cleared by software.

SPIIF: SPI interrupt flag, must be cleared by software.

11.1 Priority level structure

All interrupt sources are combined in groups:

Table 11-2: Priority level groups

Groups		
External interrupt 0	Serial channel 1 interrupt	-
Timer 0 interrupt	-	SPI interrupt
External interrupt 1	-	ADC interrupt
Timer 1 interrupt	-	EEI interrupt
Serial channel 0 interrupt	-	-
Timer 2 interrupt	-	IIC interrupt

Each group of interrupt sources can be programmed individually to one of the four priority levels by setting or clearing one bit in the SFRs IP0 and IP1. If requests of the same priority level is received simultaneously, an internal polling sequence determines which request is serviced first.

Mnemonic: IP0							Address: A9h	
7	6	5	4	3	2	1	0	Reset
-	-	IP0.5	IP0.4	IP0.3	IP0.2	IP0.1	IP0.0	00h

Mnemonic: IP1							Address: B9h	
7	6	5	4	3	2	1	0	Reset
-	-	IP1.5	IP1.4	IP1.3	IP1.2	IP1.1	IP1.0	00h

Table 11-3: Priority levels

IP1.x	IP0.x	Priority Level
0	0	Level0 (lowest)
0	1	Level1
1	0	Level2
1	1	Level3 (highest)



Table 11-4: Groups of priority

Bit	Group		
IP1.0, IP0.0	External interrupt 0	Serial channel 1 interrupt	-
IP1.1, IP0.1	Timer 0 interrupt	-	SPI interrupt
IP1.2, IP0.2	External interrupt 1	-	ADC interrupt
IP1.3, IP0.3	Timer 1 interrupt	-	EEl interrupt
IP1.4, IP0.4	Serial channel 0 interrupt	-	-
IP1.5, IP0.5	Timer 2 interrupt	-	IIC interrupt

Table 11-5: Polling sequence

Interrupt source	Sequence
External interrupt 0	
Serial channel 1 interrupt	
Timer 0 interrupt	
SPI interrupt	
External interrupt 1	
ADC interrupt	
Timer 1 interrupt	
EEl interrupt	
Serial channel 0 interrupt	
Timer 2 interrupt	
IIC interrupt	



12 Power Management Unit

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

Mnemonic: PCON							Address: 87h	
7	6	5	4	3	2	1	0	Reset
SMOD	MDUF	-	PMW	-	-	STOP	IDLE	00h

STOP: Stop mode control bit. Setting this bit turning on the Stop Mode.

Stop 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

12.112.1 Idle mode

Setting the IDLE bit of PCON register invokes the IDLE mode. The IDLE mode stop the clock source for CPU but keep the peripherals under running condition. The power consumption will drop because the CPU is not active now. The CPU can exit the IDLE state with any interrupts or a reset.

12.212.2 Stop mode

Setting the STOP bit of PCON register invokes the STOP mode. All internal clocking in this mode is turned off. The CPU will exit this state from a no-clocked external interrupt or a reset condition. Internally generated interrupts (timer, serial port, watchdog ...) are not useful since they require clocking activity.



13 Pulse Width Modulation (PWM)

SM59R16A2/SM59R08A2 provides four-channel PWM outputs. The 4 channels can be used simultaneously. But their configuration (the counting bit number and counting frequency) will be the same defined in one SFR.

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
PWM											
PWMC	PWM Control register	B5h	-	-	PWMM[1:0]		PWM3 EN	PWM2 EN	PWM1 EN	PWM0 EN	00h
PWMD0H	PWM 0 Data register high byte	BCh	-	-	-	-	PWMD0[11:8]				00h
PWMD0L	PWM 0 Data register low byte	BDh	PWMD0[7:0]								00h
PWMD1H	PWM 1 Data register high byte	BEh	-	-	-	-	PWMD1[11:8]				00h
PWMD1L	PWM 1 Data register low byte	BFh	PWMD1[7:0]								00h
PWMD2H	PWM 2 Data register high byte	B1h	-	-	-	-	PWMD2[11:8]				00h
PWMD2L	PWM 2 Data register low byte	B2h	PWMD2[7:0]								00h
PWMD3H	PWM 3 Data register high byte	B3h	-	-	-	-	PWMD3[11:8]				00h
PWMD3L	PWM 3 Data register low byte	B4h	PWMD3[7:0]								00h

Mnemonic: PWMC						Address: B5h		
7	6	5	4	3	2	1	0	Reset
-	-	PWMM[1:0]		PWM3EN	PWM2EN	PWM1EN	PWM0EN	00h

PWMM[1:0] PWM mode select.

] : When PWMM[1:0] = 00 or 11 , the PWM output frequency = $F_{osc}/256$.

When PWMM[1:0] = 01 , the PWM output frequency = $F_{osc}/1024$.

When PWMM[1:0] = 10 , the PWM output frequency = $F_{osc}/4096$.

Also

PWMM[1:0]	Mode
00	8-bit mode
01	10-bit mode
10	12-bit mode
11	8-bit mode

here F_{osc} is the external crystal or oscillator frequency

PWM3EN: PWM Channel 3 enable control bit.

PWM3EN = 1 – PWM Channel 3 enable.

PWM3EN = 0 – PWM Channel 3 disable.

PWM2EN: PWM Channel 2 enable control bit.

PWM2EN = 1 – PWM Channel 2 enable.

PWM2EN = 0 – PWM Channel 2 disable.

PWM1EN: PWM Channel 1 enable control bit.

PWM1EN = 1 – PWM Channel 1 enable.

PWM1EN = 0 – PWM Channel 1 disable.

PWM0EN: PWM 0 Channel 0 enable control bit.

PWM0EN = 1 – PWM Channel 0 enable.

PWM0EN = 0 – PWM Channel 0 disable.



Mnemonic: PWMD0H				Address: BCh			
7	6	5	4	3	2	1	0 Reset
-	-	-	-	PWMD0[11:8]			00h

Mnemonic: PWMD0L				Address: BDh			
7	6	5	4	3	2	1	0 Reset
PWMD0[7:0]							00h

PWMD0[11:0]: PWM channel 0 data register.

Mnemonic: PWMD1H				Address: BEh			
7	6	5	4	3	2	1	0 Reset
-	-	-	-	PWMD1[11:8]			00h

Mnemonic: PWMD1L				Address: BFh			
7	6	5	4	3	2	1	0 Reset
PWMD1[7:0]							00h

PWMD1[11:0]: PWM channel 1 data register.

Mnemonic: PWMD2H				Address: B1h			
7	6	5	4	3	2	1	0 Reset
-	-	-	-	PWMD2[11:8]			00h

Mnemonic: PWMD2L				Address: B2h			
7	6	5	4	3	2	1	0 Reset
PWMD2[7:0]							00h

PWMD2[11:0]: PWM channel 2 data register.

Mnemonic: PWMD3H				Address: B3h			
7	6	5	4	3	2	1	0 Reset
-	-	-	-	PWMD3[11:8]			00h

Mnemonic: PWMD3L				Address: B4h			
7	6	5	4	3	2	1	0 Reset
PWMD3[7:0]							00h

PWMD3[11:0]: PWM channel 3 data register.



14 IIC function

As most of the IIC we have been familiar with, this IIC module uses the SCL (clock) and the SDA (data) line to communicate with the other IIC interfaces. Its speed can be selected up to 400Kbps (maximum) by software setting the SFR IICBR[2:0]. The IIC module can be either master or slave, provided two interrupts (RXIF, TXIF), and has two addresses for data transmission. It will generate START, repeated START and STOP signals automatically in master mode and can detects START, repeated START and STOP signals in slave mode. The maximum communication length and the number of devices that can be connected are limited by a maximum bus capacitance of 400pF. SM59R16A2/SM59R08A2 IIC function is fully compatible to most of the other chips'. So there is no barrier in the mutual communication.

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
IIC function											
IICCTL	IIC control register	F9h	IICEN	BF	MSS	MAS	RStart	IICBR[2:0]			04h
IICS	IIC status register	F8h	MStart	RXIF	TXIF	RDR	TDR	RXAK	TXAK	RW	00h
IICA1	IIC Address 1 register	FAh	IICA1[7:1]							MATCH1 or RW1	A0h
IICA2	IIC Address 2 register	FBh	IICA2[7:1]							MATCH2 or RW2	60h
IICRWD	IIC Read/Write register	FCh	IICSRWD[7:0]								00h

Mnemonic: IICCTL

Address: F9h

7	6	5	4	3	2	1	0	Reset
IICEN	BF	MSS	MAS	RStart	IICBR[2:0]			04h

IICEN: Enable IIC module

IICEN = 1 is Enable

IICEN = 0 is Disable.

BF: Bus failed flag (used in master mode only)

When the module is transmitting a "1" to SDA line but detected as a "0" from SDA line in master mode, it is called as arbitration loss. This bit can be cleared by software.

MSS: Master or slave mode select.

MSS = 1 is master mode.

MSS = 0 is slave mode.

*The software must set this bit before setting others register.

MAS: Master address select (master mode only)

MAS = 0 is to use IICA1.

MAS = 1 is to use IICA2.

RStart: Re-start control bit (master mode only)

When this bit is set, the module will generate a start condition to the SDA and SCL lines (after current ACK) and send out the calling address which is stored in either IICA1 or IICA2 (selected by MAS control bit). After the address is sent out, this bit will be cleared by hardware.



IICBR[2:0]: Baud rate selection (master mode only), where Fosc is the external crystal or oscillator frequency. The default is Fosc/512 for users' convenience.

IICBR[2:0]	Baud rate
000	Fosc/32
001	Fosc/64
010	Fosc/128
011	Fosc/256
100	Fosc/512
101	Fosc/1024
110	Fosc/2048
111	Fosc/4096

Mnemonic: IICS							Address: F8h	
7	6	5	4	3	2	1	0	Reset
MStart	RxIF	TxIF	RDR	TDR	RxAK	TxAK	RW	00h

MStart: Master start control bit (master mode only)

If this bit is set, the module will generate a start condition to the SDA and SCL lines, and send out the calling address which is stored in either IICA1 or IICA2 (selected by MAS control bit). After software clears this bit, the module will generate a stop condition to the SDA and SCL.

RxIF: Data receive interrupt flag

It is set after the IICRWD (IIC read /write data buffer) is loaded with a newly receive data. After software clears this bit, the IICIF (IIC interrupt flag) will cleared.

TxIF: Data transmit interrupt flag

It is set when all the 8 bits in the shift register are transmitted, the 8 bits are from IICRWD (IIC read /write data buffer) downloaded into the shift register. After software clears this bit, the IICIF (IIC interrupt flag) will cleared.

RDR: Read data ready

It is set to high by hardware when a new byte is received and stored in IICRWD. The software must clear this bit after it gets the data from IICRWD. The IIC module is able to write new data into IICRWD only when this bit is cleared.

TDR: Transmit data ready

After putting the data into IICRWD in transmission, the software needs to set this bit to '1' to inform the IIC module to send the data out. After IIC module finishes sending the data from IICRWD, this bit will be cleared automatically.

RxAK: Receive acknowledgement

This is a read-only bit judged by the transmitting side only.

If the IIC module is in the master mode : after it transmits the 8-bit data to the slave side, the slave side will returned RxAK

= 0 : the slave receives the data successfully

= 1 : the slave fails to receive the data

If the IIC module is in the slave mode : after it sends the 8-bit data to the master side, the master side will returned RxAK

= 0 : the master receives the data successfully(in some application, it may be that the master wants more data)

= 1 : the master fails to receive the data (in some applications, it may be that the master does not want any more data)

TxAK: Transmit acknowledgement

It is the corresponding bit of RxAK in the receiving side. It represents the receiving status as explained in RxAK. Actually, it is sent as the 9th bit in one byte transmission as show in Fig. 14-1.

RW: Slave mode read or write

It is a read-only bit used in slave mode only. It is from Bit 0 of IICA1 or IICA2 of the master side as described below

= 0 : master asks this IIC module (in slave mode) to receive data (read)

=1 : master asks this IIC module (in slave mode) to transmit data (write)

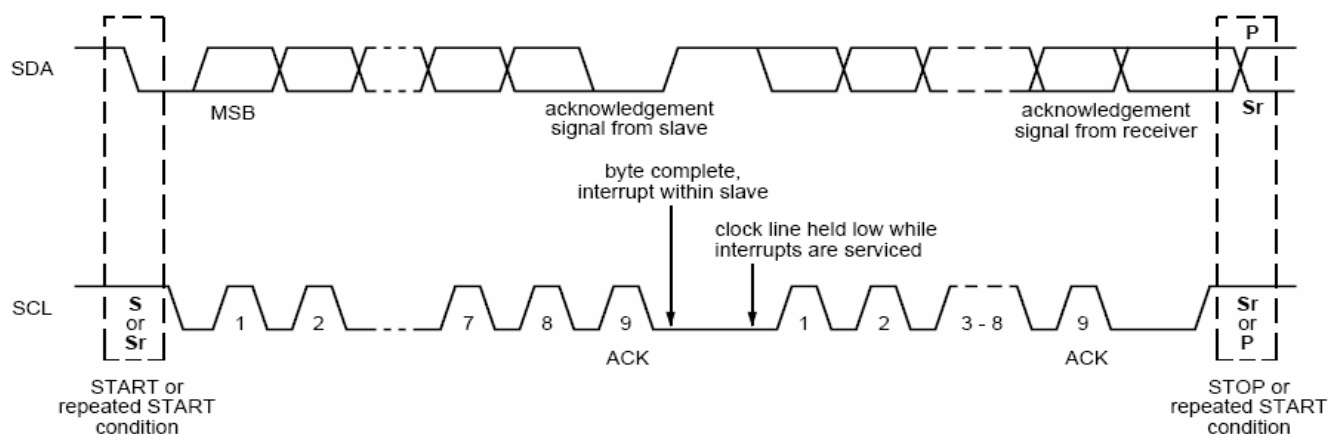


Fig. 14-1: Acknowledgement bit in the 9th bit of a byte transmission

Mnemonic: IICA1							Address: FAh	
7	6	5	4	3	2	1	0	Reset
IICA1[7:1]							Match1 or RW1	A0h

Slave mode:

IICA1[7:1]: IIC Address registers

This is the first 7-bit address for this slave module. It will be checked when an address (from master) is received

Match1: When IICA1 matches with the received address from the master side, this bit will set to 1 by hardware. When IIC bus is stopped, this bit will clear automatically.

Master mode:

IICA1[7:1]: IIC Address registers

This 7-bit address indicate the slave with which it want to communicate.

RW1: This bit will be sent out as RW of the slave side if the module has set the MStart or RStart bit. It appears at the 8th bit after the IIC address as shown in Fig. 14-2. It is used to tell the salve the direction of the following communication. If it is 1, the module is in master receive mode. If 0, the module is in master transmit mode.

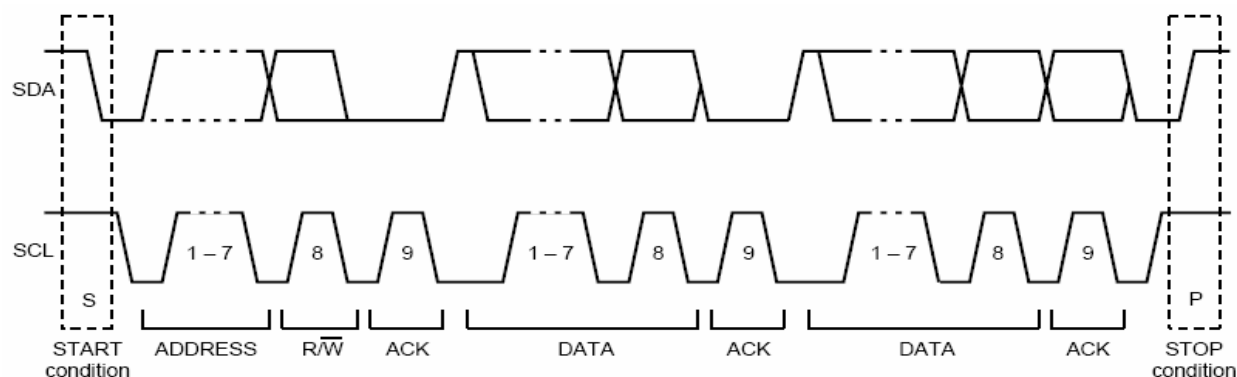


Fig. 14-2: RW bit in the 8th bit after IIC address



Mnemonic: IICA2							Address: FBh	
7	6	5	4	3	2	1	0	Reset
IICA2[7:1]							Match2 or RW2	60h
R/W							R or R/W	

Slave mode:

IICA2[7:1]: IIC Address registers

This is the second 7-bit address for this slave module.

It will be checked when an address (from master) is received

Match2: When IICA2 matches with the received address from the master side, this bit will set to 1 by hardware. When IIC bus is stopped, this bit will clear automatically.

Master mode:

IICA2[7:1]: IIC Address registers

This 7-bit address indicate the slave with which it want to communicate.

RW2: This bit will be sent out as RW of the slave side if the module has set the MStart or RStart bit. It is used to tell the salve the direction of the following communication. If it is 1, the module is in master receive mode. If 0, the module is in master transmit mode.

Mnemonic: IICRWD							Address: FCh	
7	6	5	4	3	2	1	0	Reset
IICRWD[7:0]								00h

IICRWD[7:0]: IIC read write data buffer.

In receiving (read) mode, the received byte is stored here.

In transmitting mode, the byte to be shifted out through SDA stays here.



15 SPI function

Serial Peripheral Interface (SPI) is a synchronous protocol that allows a master device to initiate communication with slave devices. There are 4 signals used in SPI, they are

SPI_MOSI: data output in the master mode, data input in the slave mode,
SPI_MISO: data input in the master mode, data output in the master mode,
SPI_SCK: clock output form the master, the above data are synchronous to this signal
SPI_SS: input in the slave mode.

This slave device detects this signal to judge if it is selected by the master.
In the master mode, it can select the desired slave device by any IO with value = 0. Fig. 15-1 is an example showing the relation of the 4 signals between master and slaves.

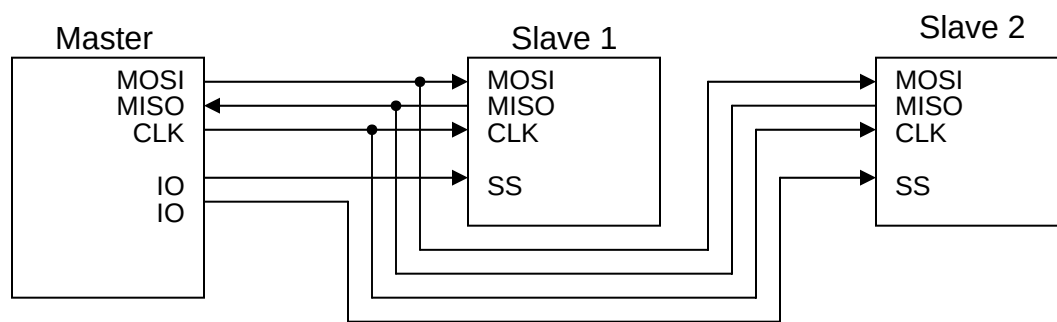


Fig. 15-1: SPI signals between master and slave devices

There is only one channel SPI interface. The SPI SFRs are shown as below:

SPI	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
SPI function											
SPIC1	SPI control register 1	F1h	SPIEN	SPIMSS	SPISSP	SPICKP	SPICKE	SPIBR[2:0]			08h
SPIC2	SPI control register 2	F2h	SPIFD	TBC[2:0]			-	RBC[2:0]			00h
SPIS	SPI status register	F5h	-	SPIMLS	SPIOV	SPITXIF	SPITDR	SPIRXIF	SPIRDR	SPIRS	40h
SPITXD	SPI transmit data buffer	F3h	SPITXD[7:0]								00h
SPIRXD	SPI receive data buffer	F4h	SPIRXD[7:0]								00h



Mnemonic: SPIC1				Address: F1h				
7	6	5	4	3	2	1	0	Reset
SPIEN	SPIMSS	SPISSP	SPICKP	SPICKE	SPIBR[2:0]			08h

SPIEN: Enable SPI module. "1" is Enable. "0" is Disable.

SPIMSS: Master or Slave mode Select

"1" is Master mode.

"0" is Slave mode.

SPISSP: Slave Select (SS) active polarity (slave mode used only)

"1" - high active.

"0" - low active.

SPICKP: Clock idle polarity (master mode used only)

"1" – SCK high during idle. Ex :



"0" - SCK low during idle. Ex :

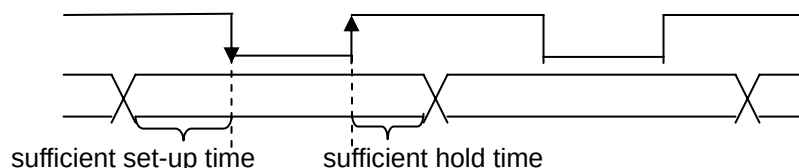


SPICKE: Clock sample edge select.

"1" – data latch in rising edge

"0" – data latch in falling edge.

* To ensure the data latch stability, SM59R16A2/SM59R08A2 generate the output data as given in the following example, the other side can latch the stable data no matter in rising or falling edge.



SPIBR[2:0]: SPI baud rate select (master mode used only), here Fosc is the external crystal or oscillator frequency :

SPIBR[2:0]	Baud rate
0:0:0	Fosc/4
0:0:1	Fosc/8
0:1:0	Fosc/16
0:1:1	Fosc/32
1:0:0	Fosc/64
1:0:1	Fosc/128
1:1:0	Fosc/256
1:1:1	Fosc/512



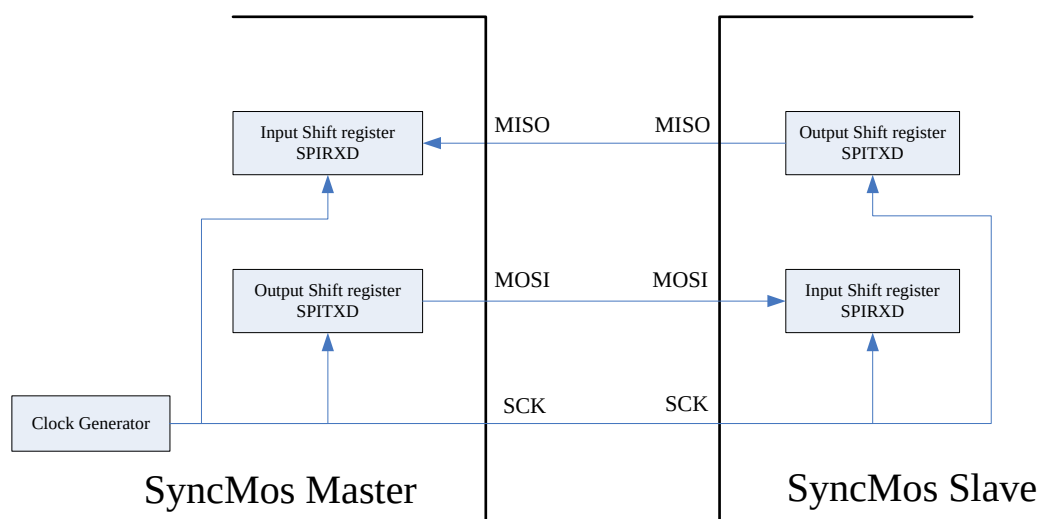
Mnemonic: SPIC2						Address: F2h		
7	6	5	4	3	2	1	0	Reset
SPIFD	TBC[2:0]			-	RBC[2:0]			00h

SPIFD: Full-duplex mode enable.

“1” : enable full-duplex mode.

“0” : disable full-duplex mode.

When it is set, the TBC[2:0] and RBC[2:0] will be reset and keep to zero, i.e., only 8-bit communication is allowed in the full-duplex mode. When the master device transmits data to the slave device via the MOSI line, the slave device responds sends data back to the master device via the MISO line. This implies that full-duplex transmission with both out-data and in-data are synchronized with the same clock SCK as shown below.



TBC[2:0]: SPI transmitter bit counter, here 1-8 bits are allowed except for the full-duplex mode

TBC[2:0]	Bit counter
0:0:0	8 bits output
0:0:1	1 bit output
0:1:0	2 bits output
0:1:1	3 bits output
1:0:0	4 bits output
1:0:1	5 bits output
1:1:0	6 bits output
1:1:1	7 bits output

RBC[2:0]: SPI receiver bit counter, here 1-8 bits are allowed except for the full-duplex mode

RBC[2:0]	Bit counter
0:0:0	8 bits input
0:0:1	1 bit input
0:1:0	2 bits input
0:1:1	3 bits input
1:0:0	4 bits input
1:0:1	5 bits input
1:1:0	6 bits input
1:1:1	7 bits input



Mnemonic: SPIS								Address: F5h
7	6	5	4	3	2	1	0	Reset
-	SPIMLS	SPIOV	SPITXIF	SPITDR	SPIRXIF	SPIRDR	SPIRS	40h

SPIMLS: MSB or LSB output /input first

“1” : MSB output/input first

“0” : LSB output/input first

SPIOV: Overflow flag.

When SPIRDR is set (one byte in SPIRXD but has not been taken away) and the next data also enters (there is no blocking function), this flag will be set to inform that the received data in SPIRXD is damaged by this overflow. It is clear by hardware when SPIRDR is cleared.

SPITXIF: Transmit Interrupt Flag.

This bit is set when the data of the SPITXD register is downloaded to the shift register.

SPITDR: Transmit Data Ready.

When MCU finish writing data to SPITXD register, the MCU needs to set this bit to '1' to inform the SPI module to send the data. After SPI module finishes sending the data from SPITXD or SPITXD is downloaded to shift register, this bit will be cleared automatically.

SPIRXIF: Receive Interrupt Flag.

This bit is set after the SPIRXD is loaded with a newly receive data.

SPIRDR: Receive Data Ready.

When a byte is received, SPIRDR is set as a flag to inform MCU. The MCU must clear this bit after it gets the data from SPIRXD register. If the SPI module on the transmit side writes new data into the SPIRXD before this bit is cleared, then the data will be overwritten.

SPIRS: Receive Start.

This bit set to “1” to inform the SPI module to receive the data into SPIRXD register.

Mnemonic: SPITXD								Address: F3h
7	6	5	4	3	2	1	0	Reset
SPITXD[7:0]								00h

SPITXD[7:0]: Transmit data buffer.

Mnemonic: SPIRXD								Address: F4h
7	6	5	4	3	2	1	0	Reset
SPIRXD[7:0]								00h

SPIRXD[7:0]: Receive data buffer.



16 Expanded External Interrupt (EEI) interface

Expanded External Interrupt (EEI) interface can be connected to an 8 x n matrix keyboard or any similar devices. It has 8 inputs with programmable interrupt capability on either high or low level. These 8 inputs are through P1 and can be the external interrupts to leave from the idle and stop modes. The 8 inputs are independent from each other but share the same interrupt vector.

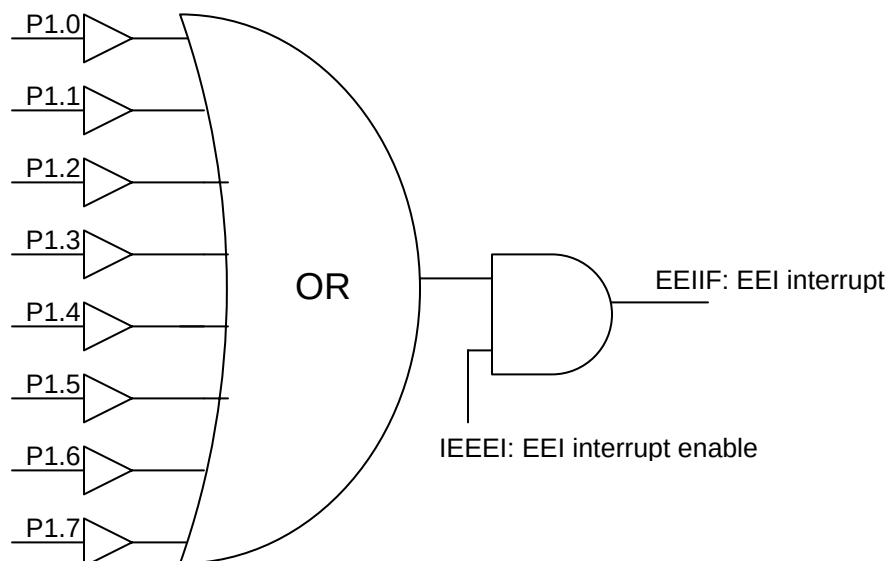


Fig. 16-1: Interrupts from EEI 8 inputs

EEI	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
EEI function											
KBLS	EEI level selection	93h	KBLS7	KBLS6	KBLS5	KBLS4	KBLS3	KBLS2	KBLS1	KBLS0	00h
KBE	EEI input enable	94h	KBE7	KBE6	KBE5	KBE4	KBE3	KBE2	KBE1	KBE0	00h
KBF	EEI flag	95h	KBF7	KBF6	KBF5	KBF4	KBF3	KBF2	KBF1	KBF0	00h



Mnemonic: KBLS

Address: 93h

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

- KBLS.7: EEI line 7 level selection bit
0 : enable a low level detection on P17.
1 : enable a high level detection on P17.
- KBLS.6: EEI line 6 level selection bit
0 : enable a low level detection on P16.
1 : enable a high level detection on P16.
- KBLS.5: EEI line 5 level selection bit
0 : enable a low level detection on P15.
1 : enable a high level detection on P15.
- KBLS.4: EEI line 4 level selection bit
0 : enable a low level detection on P14.
1 : enable a high level detection on P14.
- KBLS.3: EEI line 3 level selection bit
0 : enable a low level detection on P13.
1 : enable a high level detection on P13.
- KBLS.2: EEI line 2 level selection bit
0 : enable a low level detection on P12.
1 : enable a high level detection on P12.
- KBLS.1: EEI line 1 level selection bit
0 : enable a low level detection on P11.
1 : enable a high level detection on P11.
- KBLS.0: EEI line 0 level selection bit
0 : enable a low level detection on P10.
1 : enable a high level detection on P10.

Mnemonic: KBE

Address: 94h

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

- KBE.7: EEI line 7 enable bit
0 : enable standard I/O pin.
1 : enable KBF.7 bit in KBF register to generate an interrupt request.
- KBE.6: EEI line 6 enable bit
0 : enable standard I/O pin.
1 : enable KBF.6 bit in KBF register to generate an interrupt request.
- KBE.5: EEI line 5 enable bit
0 : enable standard I/O pin.
1 : enable KBF.5 bit in KBF register to generate an interrupt request.
- KBE.4: EEI line 4 enable bit
0 : enable standard I/O pin.
1 : enable KBF.4 bit in KBF register to generate an interrupt request.
- KBE.3: EEI line 3 enable bit
0 : enable standard I/O pin.
1 : enable KBF.3 bit in KBF register to generate an interrupt request.
- KBE.2: EEI line 2 enable bit
0 : enable standard I/O pin.
1 : enable KBF.2 bit in KBF register to generate an interrupt request.
- KBE.1: EEI line 1 enable bit
0 : enable standard I/O pin.
1 : enable KBF.1 bit in KBF register to generate an interrupt request.
- KBE.0: EEI line 0 enable bit
0 : enable standard I/O pin.
1 : enable KBF.0 bit in KBF register to generate an interrupt request.



Mnemonic: KBF							Address: 95h	
7	6	5	4	3	2	1	0	Reset
KBF.7	KBF.6	KBF.5	KBF.4	KBF.3	KBF.2	KBF.1	KBF.0	00h

KBF.7: EEI line 7 flag

This is set by hardware when P17 detects a programmed level.

It generates a EEI interrupt request if KBE.7 is also set. It must be cleared by software.

KBF.6: EEI line 6 flag

This is set by hardware when P16 detects a programmed level.

It generates a EEI interrupt request if KBE.6 is also set. It must be cleared by software.

KBF.5: EEI line 5 flag

This is set by hardware when P15 detects a programmed level.

It generates a EEI interrupt request if KBE.5 is also set. It must be cleared by software.

KBF.4: EEI line 4 flag

This is set by hardware when P14 detects a programmed level.

It generates a EEI interrupt request if KBE.4 is also set. It must be cleared by software.

KBF.3: EEI line 3 flag

This is set by hardware when P13 detects a programmed level.

It generates a EEI interrupt request if KBE.3 is also set. It must be cleared by software.

KBF.2: EEI line 2 flag

This is set by hardware when P12 detects a programmed level.

It generates a EEI interrupt request if KBE.2 is also set. It must be cleared by software.

KBF.1: EEI line 1 flag

This is set by hardware when P11 detects a programmed level.

It generates a EEI interrupt request if KBE.1 is also set. It must be cleared by software.

KBF.0: EEI line 0 flag

This is set by hardware when P10 detects a programmed level.

It generates a EEI interrupt request if KBE.0 is also set. It must be cleared by software.

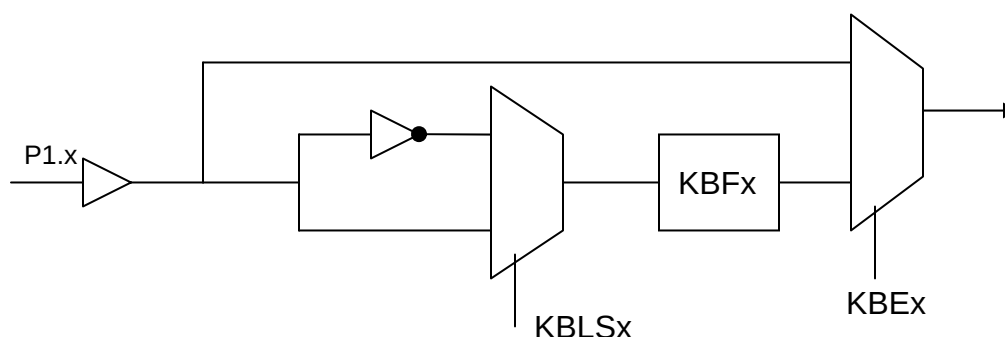


Fig. 16-2: Block diagram of EEI input



17 EEPROM

For any byte in the 64KB/32KB Flash memory which is not programmed, it can be used to record the data. The data can be stored and updated as if there is EEPROM embedded. Since it is not really saved into EEPROM, it is also called “virtual EEPROM” function. The EEPROM function in SM59R16A2/SM59R08A2 is easy to be used. Basically, it is done by hardware circuits to reduce the efforts in firmware coding. The necessary data move, page erase and write back, etc, are executed automatically by this hardware.

It is worth reminding again that the location of the EEPROM can be any byte within the program Flash where no program code occupies. Users must be very careful in doing EEPROM write, not to write to the program area.

The users can only set the PMW (program memory write) bit to do the EEPROM function through ACC register. If PMW = 1, the MOVX instruction will read/write the data from/to the Flash memory directly, instead of the internal or external SRAM. Data overwrite (update) is also supported because the hardware circuits will erase the original data first, then write the new data.

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
EEPROM Function											
PCON	Power Control	87h	SMOD	MDUF	-	PMW	-	-	STOP	IDLE	00h
PES	Program Memory Page Erase Control Register	A1h	EPE	-	-	-	-	-	-	-	00h

Mnemonic: PCON

Address: 87h

7	6	5	4	3	2	1	0	Reset
SMOD	MDUF	-	PMW	-	-	STOP	IDLE	00h

When the PMW is cleared or after reset, the MOVX instructions allow read/write access to the data memory address space again. The software switches the PMW bit to enable access to the program memory address space. The following table shows the program memory instructions when the PMW bit is set.

Mnemonic	Description
MOVX A,@Ri	Move program memory data (8-bits addr.) to ACC
MOVX A,@DPTR	Move program memory data (16-bits addr.) to ACC
MOVX @Ri,A	Move ACC to program memory (8-bits addr.)
MOVX @DPTR,A	Move ACC to program memory (16-bits addr.)

Mnemonic: PES

Address: A1h

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

When enable the EPE bit, the Page Erase (Each page include 512 bytes) function can be executed by below Instructions .



ORL	PCON,#010h	;	Enable Program Memory read/write
MOV	DPTR,#0200h	;	Define page erase area from 0x0200 to 0x03FF
MOV	PES,#080h	;	Enable Page erase function
MOV	A,#0FFH	;	Put 0xFF into ACC register
MOVX	@DPTR,A	;	When this instruction execute , The Program Memory 0x0200 to
		;	0X03FF value will all change to 0xFF
MOV	PES,#00h	;	Disable Page erase function
ANL	PCON,#0EFh	;	Disable Program Memory read/write



18 10-bit Analog-to-Digital Converter (ADC)

SM59R16A2/SM59R08A2 provides four channel 10-bit ADC. This ADC is in SAR architecture with excellent precision.

It is advised that there will be no large current surge caused by IO ports or any other functions when the ADC is measuring. The large current surge may influence the voltage reference, and make the results inaccurate.

The Digital output of the sampled analog signal is put into ADCD [9:0]. The ADC interrupt vector is 53h.

The embedded 4-channel ADC is a 10-bit-resolution device with measurement range 0 ~ 3.3V. The 4 channels are in Port4 [7] ~ Port4 [4]. The following figure shows the precision of this ADC in real application:

P.S. When ADC module used at VDD=5.0V system, User must attention below two items:

1. The Port4 [7 : 4] must output "0000" value to delete the offset voltage before start the ADC convertor.
- 2.

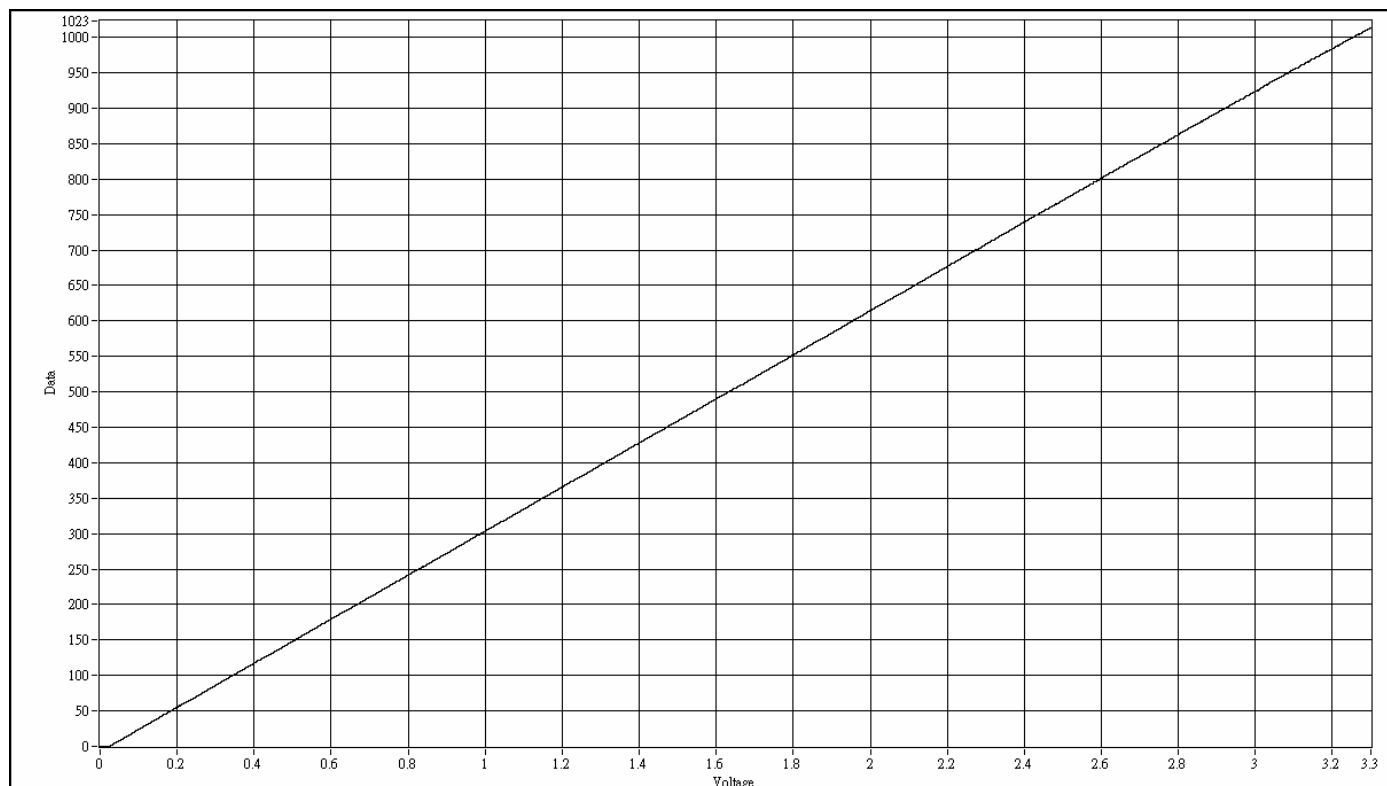


Fig. 20-1: The precision of 10-b ADC (The VDD=3.3V)



Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
ADC											
ADCC1	ADC Control 1	ABh	-	-	-	-	ADC3E	ADC2E	ADC1E	ADC0E	00h
ADCC2	ADC Control 2	ACh	COM	START	ADC8B	-	ADCCCH[1:0]		ADCCS[1:0]		00h
ADCDH	ADC data high byte	ADh	-						ADCDH [1:0]		00h
ADCDL	ADC data low byte	AEh	ADCDL[7:0]								00h

Mnemonic: ADCC1

Address: ABh

7	6	5	4	3	2	1	0	Reset
-	-	-	-	ADC3E	ADC2E	ADC1E	ADC0E	00h

ADC3E: =0 : No external analog input data can be accepted via ADC Chanel 3
=1 : ADC Channel 3 is enable, analog input data can be read through it.
ADC2E: =0 : No external analog input data can be accepted via ADC Chanel 2
=1 : ADC Channel 2 is enable, analog input data can be read through it.
ADC1E: =0 : No external analog input data can be accepted via ADC Chanel 1
=1 : ADC Channel 1 is enable, analog input data can be read through it.
ADC0E: =0 : No external analog input data can be accepted via ADC Chanel 0
=1 : ADC Channel 0 is enable, analog input data can be read through it.

Mnemonic: ADCC2

Address: ACh

7	6	5	4	3	2	1	0	Reset
COM	START	ADC8B	-	ADCCH[1:0]		ADCCS[1:0]		00h

COM: When one conversion is done, COM will be set to 1 to notify the users. It will be clear automatically by hardware. This bit is read only.
START: When this bit is set, the ADC will be start conversion. It will be clear automatically by hardware.
ADC8B: Select 10-bit or 8-bit of ADC converted data.
= 0: (default value) 10-bit data conversion ADCD[9:0], where ADCD [9:8] = ADCDH [1:0] and ADCD [7:0] = ADCDL [7:0]
= 1: 8-bit data conversion ADCD[7:0] = ADCDL [7:0]
ADCCH[1:0] The analog input signal can be chosen with it :
= 00 : Chanel 0 is used as input
= 01 : Chanel 1 is used as input
= 10 : Chanel 2 is used as input
= 11 : Chanel 3 is used as input
The users must also set the corresponding channel enable bit to 1 as described in ADCC1.
ADCCS[1:0]: This is used to select the clock frequency fed to the ADC module :
= 00 : ADC clock is system clock divided by 8
= 01 : ADC clock is system clock divided by 16
= 10 : ADC clock is system clock divided by 32
= 11 : ADC clock is system clock divided by 64
Since ADC takes about 20 ADC clock to finish one conversion, so the fastest speed of one conversion is about 160 system clocks with ADCLK=00

$$\text{ADC Clock} = \frac{F_{clk}}{8 \times 2^{ADCCS}}$$

$$\text{ADC Conversion Time} = \frac{20}{\text{ADC Clock}}$$

$$\text{ADC Sample Rate} = \frac{1}{\text{ADC Conversion Time}}$$



*The ADC clock (Fclk/n) maximum 500KHz.

ADCCS[1:0]	ADC clock	
00	Fclk/8	(Fclk: 1MHz ~ 4MHz)
01	Fclk/16	(Fclk: 4MHz ~ 8MHz)
10	Fclk/32	(Fclk: 8MHz ~ 16MHz)
11	Fclk/64	(Fclk: 16MHz ~ 32MHz)

Mnemonic: ADCDH

Address: ADh

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

ADCDH[1:0]: The high bits of digital output of this ADC

Mnemonic: ADCDL

Address: AEh

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

ADCDL[7:0]: The low bits of digital output of this ADC



Operating Conditions

Symbol	Description	Min.	Typ.	Max.	Unit.	Remarks
TA	Operating temperature	-40	25	85	°C	Ambient temperature under bias
VDD33	Supply voltage	2.7	3.3	3.6	V	
VDD5	Supply voltage	4.5	5.0	5.5	V	

DC Characteristics

(TA = -40 degree C to 85 degree C, Vdd = 3.3V)

Symbol	Parameter	Valid	Min.	Max.	Unit	Test Conditions
VIL1	Input Low Voltage	port 0,1,2,3,4,5	-0.5	0.8	V	Vdd=3.3V
VIL2	Input Low Voltage	RES, XTAL1	0	0.8	V	
VIH1	Input High Voltage	port 0,1,2,3,4,5	2.0	Vdd+0.5	V	
VIH2	Input High Voltage	RES, XTAL1	70%Vdd	Vdd+0.5	V	
VOL1	Output Low Voltage	port 0, ALE		0.45	V	IOL=3.2mA
VOL2	Output Low Voltage	port 1,2,3,4,5		0.45	V	IOL=1.6mA
VOH1	Output High Voltage	port 0	2.4		V	IOH=-800uA
			90%Vdd		V	IOH=-80uA
VOH2	Output High Voltage	port 1,2,3,4,5,ALE	2.4		V	IOH=-60uA
			90%Vdd		V	IOH=-10uA
IIL	Logical 0 Input Current	port 1,2,3,4,5		-75	uA	Vin=0.45V
ITL	Logical Transition Current	port 1,2,3,4,5		-650	uA	Vin=2.0V
ILI	Input Leakage Current	port 0		±10	uA	0.45V<Vin<Vdd
R RES	Reset Pull-down Resistance	RES	50	300	Kohm	
C IO	Pin Capacitance			10	pF	Freq=1MHz, Ta=25 °C
I CC	Power Supply Current	Vdd		25	mA	Active mode, 16MHz
				20	mA	Idle mode, 16MHz
				30	uA	Power down mode



(TA = -40 degree C to 85 degree C, Vdd = 5.0V)

Symbol	Parameter	Valid	Min.	Max.	Unit	Test Conditions
VIL1	Input Low Voltage	port 0,1,2,3,4,5	-0.5	0.8	V	Vdd=5.0V
VIL2	Input Low Voltage	RES, XTAL1	0	0.8	V	
VIH1	Input High Voltage	port 0,1,2,3,4,5	2.0	Vdd+0.5	V	
VIH2	Input High Voltage	RES, XTAL1	70%Vdd	Vdd+0.5	V	
VOL1	Output Low Voltage	port 0, ALE		0.45	V	IOL=3.2mA
VOL2	Output Low Voltage	port 1,2,3,4,5		0.45	V	IOL=1.6mA
VOH1	Output High Voltage	port 0	2.4		V	IOH=-800uA
			90%Vdd		V	IOH=-80uA
VOH2	Output High Voltage	port 1,2,3,4,5,ALE	2.4		V	IOH=-60uA
			90%Vdd		V	IOH=-10uA
IIL	Logical 0 Input Current	port 1,2,3,4,5		-75	uA	Vin=0.45V
ITL	Logical Transition Current	port 1,2,3,4,5		-650	uA	Vin=2.0V
ILI	Input Leakage Current	port 0		±10	uA	0.45V<Vin<Vdd
R RES	Reset Pull-down Resistance	RES	50	300	Kohm	
C IO	Pin Capacitance			10	pF	Freq=1MHz, Ta=25 °C
I CC	Power Supply Current	Vdd		25	mA	Active mode, 16MHz
				20	mA	Idle mode, 16MHz
				30	uA	Power down mode

Note1: Under steady state (non-transient) conditions, IOL must be externally

Limited as follows: Maximum IOL per port pin: 10mA

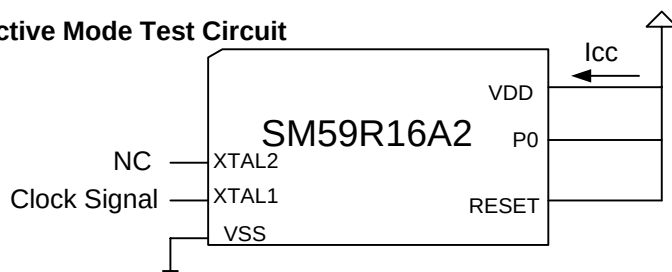
Maximum IOL per 8-bit port: port 0 : 26mA

port 1,2,3,4,5 : 15mA

Maximum total IOL for all output pins : 71mA

If IOL exceeds the condition, VOL may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.

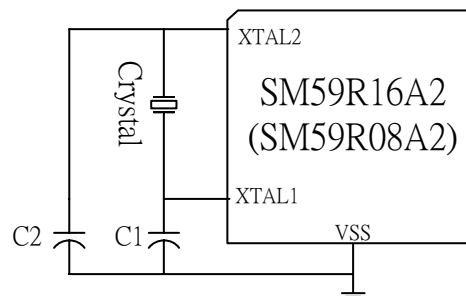
Icc Active Mode Test Circuit





Application Reference

Valid for SM59R16A2/SM59R08A2				
X'tal	2MHz	6MHz	10MHz	12MHz
C1	47 pF	35 pF	30 pF	30 pF
C2	47 pF	35 pF	30 pF	30 pF
X'tal	16MHz	25MHz		
C1	30 pF	25 pF		
C2	30 pF	25 pF		



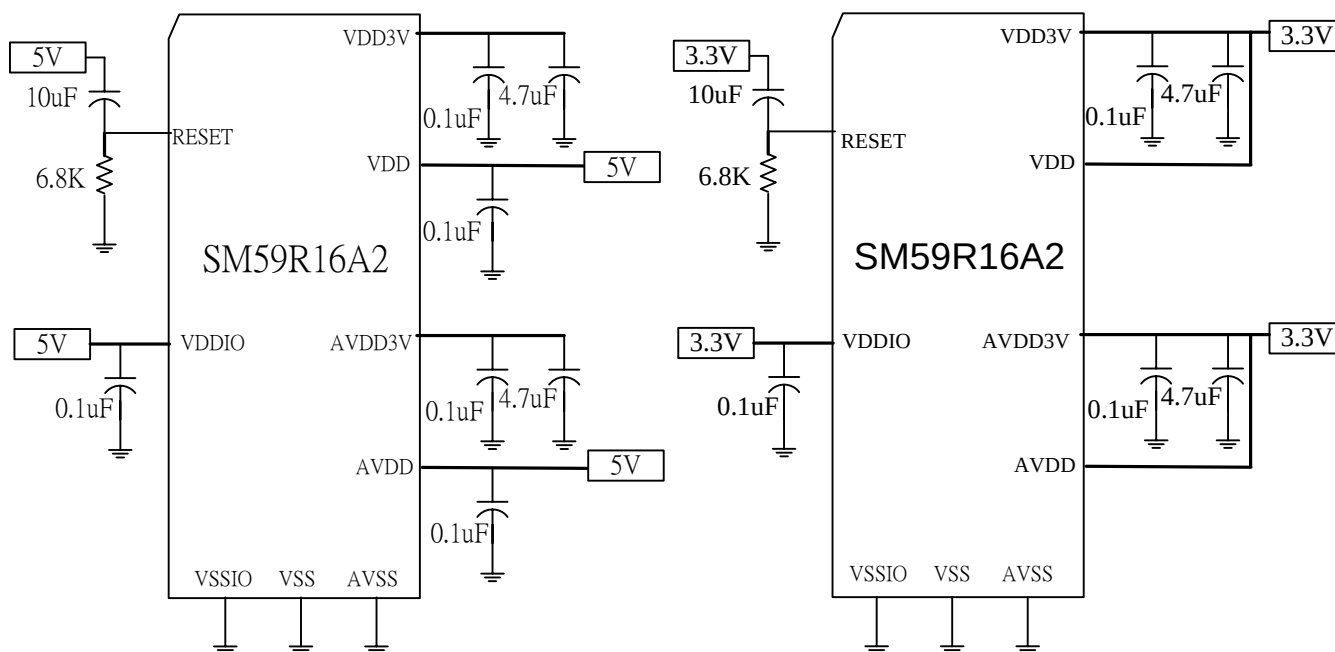
NOTE:

Oscillation circuit may differ with different crystal or ceramic resonator in higher oscillation frequency which was due to each crystal or ceramic resonator has its own characteristics.

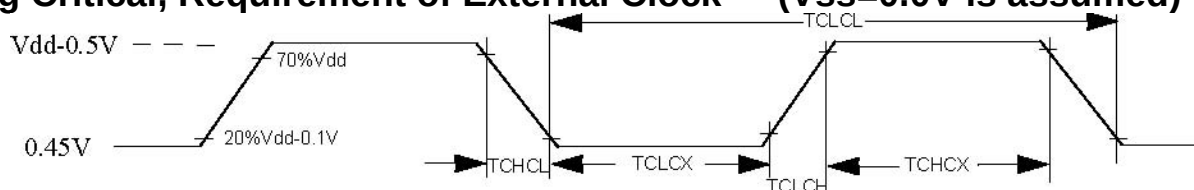
User should check with the crystal or ceramic resonator manufacture for appropriate value of external components.

Please see SM59R16A2/SM59R08A2 application note for details.

Reset Pin and 3.3V Regulator (VDD = 5V or 3.3V)



Timing Critical, Requirement of External Clock (Vss=0.0V is assumed)





MCU writer list		
Company	Contact info	Programmer Model Number
<u>Advantech</u> 7F, No.98, Ming-Chung Rd., Shin-Tien City, Taipei, Taiwan, ROC Web site: http://www.aec.com.tw	Tel:02-22182325 Fax:02-22182435 E-mail: aecwebmaster@advantech.com.tw	Lab Tool - 48XP/UXP Lab Tool – 848/848XP
<u>Hi-Lo</u> 4F.,No.18,Lane 79,Rueiguang Rd.,Neihu,Taipei,Taiwan R.O.C. Web site: http://www.hilosystems.com.tw	Tel: 02-87923301 Fax:02-87923285 E-mail: support@hilosystems.com.tw	All - 100 series
<u>Leap</u> 6th F1-4, Lane 609, Chunghsin Rd., Sec. 5, Sanchung, Taipei , Taiwan, ROC Web site: http://www.leap.com.tw	Tel: 886-2-29991860 Fax:02-29990015 E-mail: service@leap.com.tw	Leap-48
<u>Xeltek Electronic Co., Ltd</u> Bldg 6-31 Meizhiguo garden, #2 Jiangjun Ave., Jiangning, Nanjing, China 211100 Web site: http://www.xeltek-cn.com	Tel: + 86-25-52765201, E-mail: f_l@xeltek.com.cn zxl@xeltek.com.cn	Superpro 280U Superpro 580U Superpro 3000U Superpro 9000U
<u>Guangzhou Zhiyuan Electronic Co.,Ltd</u> Floor 2,No.7 building,Huangzhou Industrial Estate,Chebei Road,Tianhe district,Guangzhou,China 510660 Web site: http://www.embedtools.com/	TEL: +86-20-28872449 E-mail: mcu@programtec.com	SmartPRO 5000U/X8
<u>TianJin Weilei technology ltd</u> Rm 357,Venturetech Center,12 Keyan West Road Nankai District,Tianjin,P.R.C, 300192 Web site: http://www.weilei.com.cn/	TEL: + 86-22-87891218#801 E-mail: weilong@weilei.com.cn cm@weilei.com.cn	VP-890;VP-980;VP-880;VP-680 VP-480;VP-380;VP-280;VP-190