Description 3
Features 3
Pin Configuration 4
Pin Description 9
Special Function Register (SFR) 11
Function Description 15
1. General Features 15
1.1. Embedded Flash 15
1.2. IO Pads 15
1.3. 2T/1T Selection 15
1.4. RESET 15
1.4.1. Hardware RESET function 15
1.4.2. Software RESET function 16
1.4.3. Time Access Key register (TAKEY) 16
1.4.4. Software Reset register (SWRES) 16
1.4.5. Example of software reset 16
1.5. Clocks 17
2. Instruction Set 18
3. Memory Structure 22
3.1. Program Memory 22
3.2. Data Memory 22
3.2.1. Data memory - lower 128 byte (00h to 7Fh) 23
3.2.2. Data memory - higher 128 byte (80h to FFh) 23
3.2.3. Data memory - Expanded 1024 bytes ($0000 to $03FF) 24
4. CPU Engine 25
4.1. Accumulator 25
4.2. B Register 25
4.3. Program Status Word 26
4.4. Stack Pointer 26
4.5. Data Pointer 26
4.6. Data Pointer 1 27
4.7. Internal RAM control register 27
4.8. Interface control register 27
5. GPIO 29
6. Multiplication Division Unit (MDU) 31
6.1. Operating registers of the MDU 31
6.2. Operation of the MDU 32
6.2.1. First phase: loading the MDx registers, x = 0~5: 32
6.2.2. Second phase: executing calculation. 32
6.2.3. Third phase: reading the result from the MDx registers. 33
6.3. Normalizing 33
6.4. Shifting 33
7. Timer 0 and Timer 1 34
7.1. Timer/counter mode control register (TMOD) 34
7.2. Timer/counter control register (TCON) 35
8. Timer 2 and Capture/Compare Unit 36
8.1. Timer 2 function 38
8.1.1. Timer mode 38
8.1.2. Event counter mode 38
8.1.3. Gated timer mode 38
8.1.4. Reload of Timer 2 38
8.2. Compare function 38
8.2.1. Compare Mode 0 38
8.2.2. Compare Mode 1 39
8.3. Capture function 40
8.3.1. Capture Mode 0 40

8.3.2. Capture Mode 1 40
9. Serial interface 0 and 1 41
9.1. Serial interface 0 42
9.1.1. Mode 0 42
9.1.2. Mode 1 43
9.1.3. Mode 2 43
9.1.4. Mode 3 44
9.2. Serial interface 1 44
9.2.1. Mode A 44
9.2.2. Mode B 45
9.3. Multiprocessor communication of Serial Interface 0 and 1 45
9.4. Baud rate generator 45
9.4.1. Serial interface 0 modes 1 and 3 45
9.4.2. Serial interface 1 modes A and B 46
9.5. Clock source for baud rate 46
10. Watchdog timer 47
11. Interrupt 50
11.1. Priority level structure 52
12. Power Management Unit 54
12.1. Idle mode 54
12.2. Stop mode 54
13. Pulse Width Modulation (PWM) 55
14. IIC function 58
15. SPI function 62
16. KBI – Keyboard Interface 66
17. LVI – Low Voltage Interrupt 69
18. 10-bit Analog-to-Digital Converter (ADC) 70
19. In-System Programming (Internal ISP) 73
19.1. ISP service program 73
19.2. Lock Bit (N) 73
19.3. Program the ISP Service Program 74
19.4. Initiate ISP Service Program 74
19.5. ISP register – TAKEY, IFCON, ISPFAH, ISPFAL, ISPFD and ISPFC 75
Operating Conditions 78
DC Characteristics 78

Product List

SM59R04A2L25, SM59R04A2C25

Description

The SM59R04A2 is a 1T (one machine cycle per clock) single-chip 8-bit microcontroller. It has 16K-byte embedded Flash for program, and executes all ASM51 instructions fully compatible with MCS-51.

SM59R04A2 contains 1KB on-chip RAM, more than 44 GPIOs (LQFP-48 package type), 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 SM59R04A2 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

SM59R04A2ihhkL YWW
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}
Y: year
WW: week

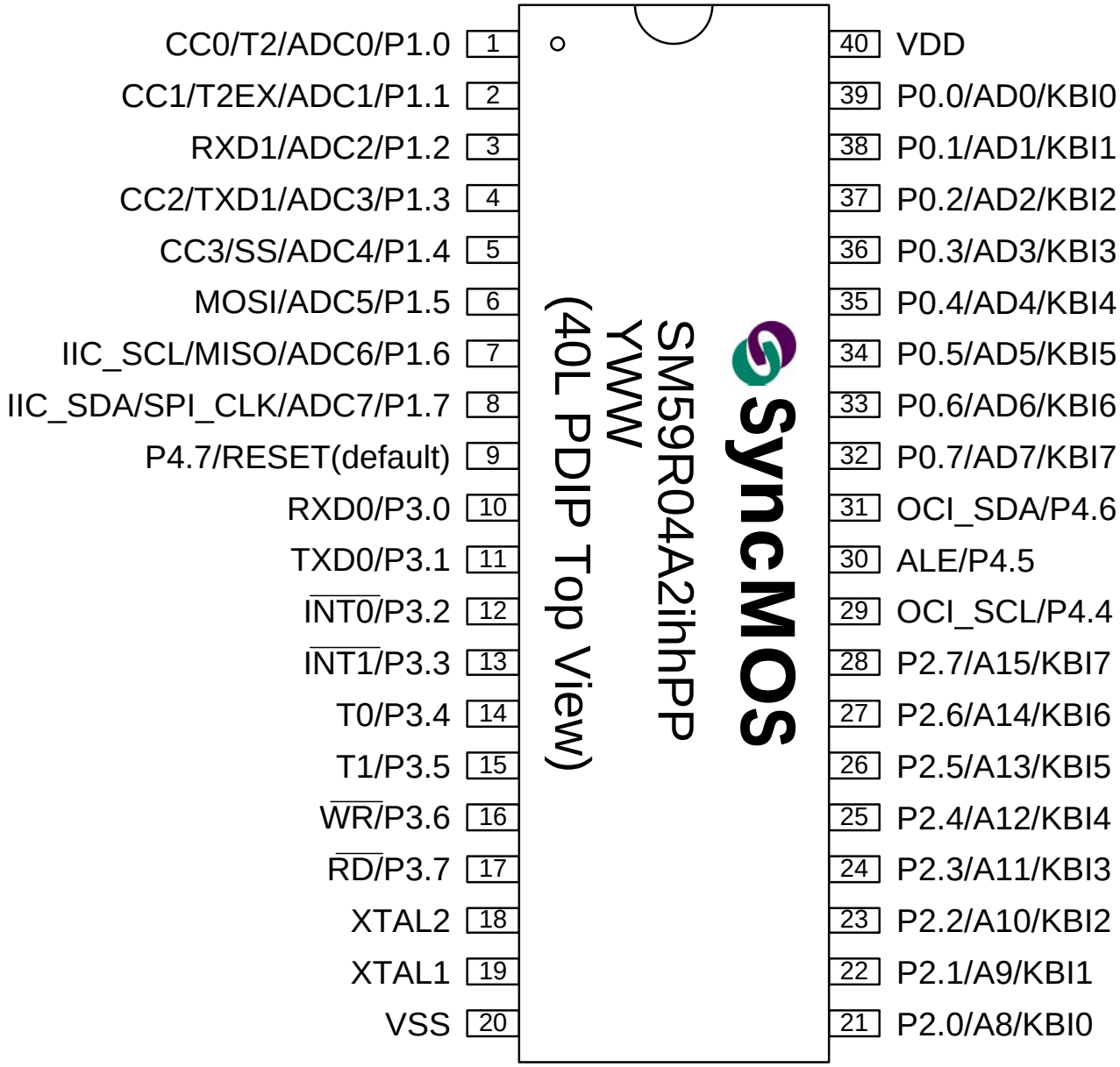
Postfix	Package	Pin / Pad Configuration
P	40L PDIP	Page 4
J	44L PLCC	Page 5
Q	44L PQFP	Page 6
V	48L LQFP	Page 7

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

Features

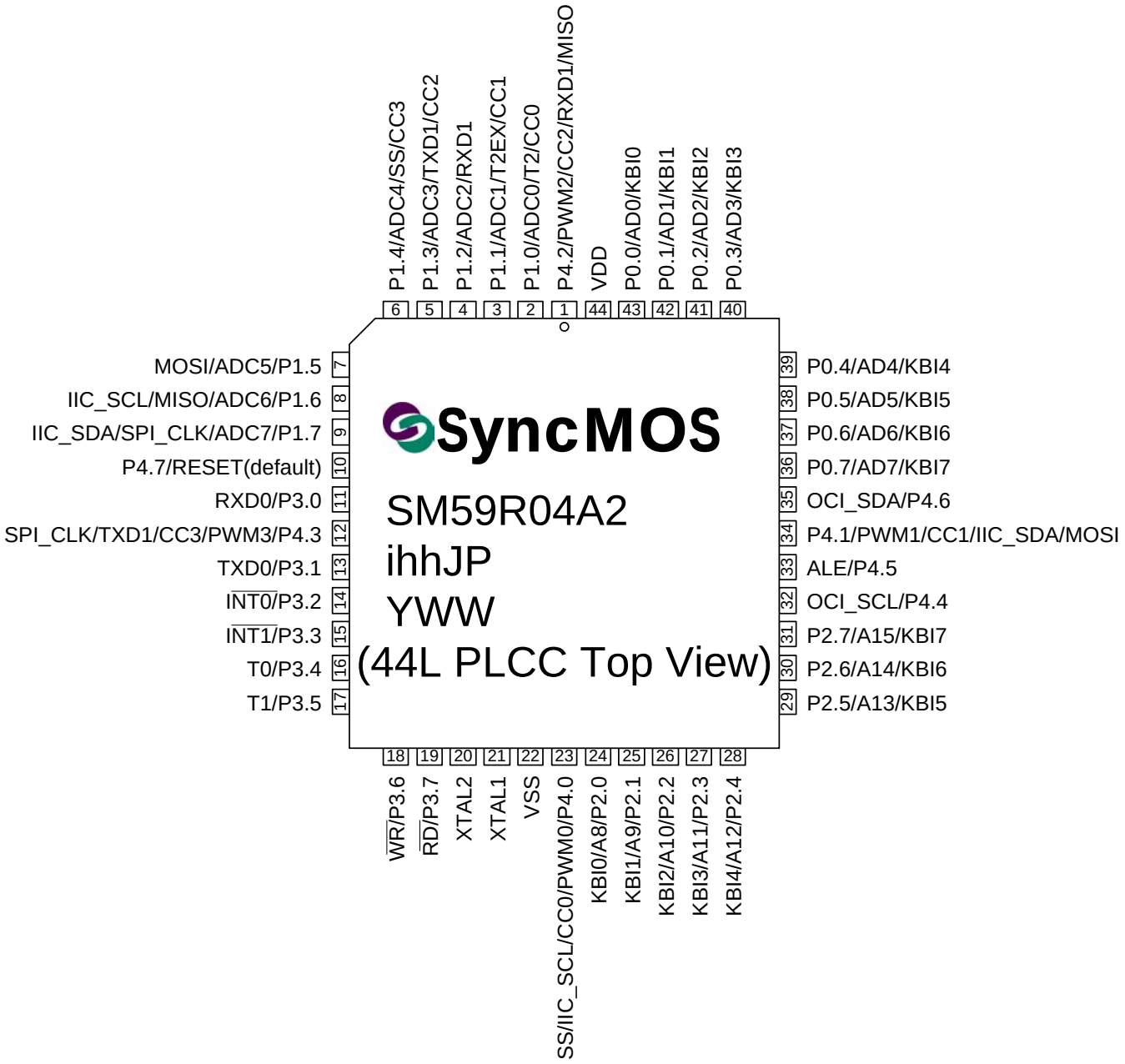
- Operating Voltage: 4.5V ~ 5.5V or 2.7V ~ 3.6V
- High speed architecture of 1 clock/machine cycle (1T), runs up to 25MHz
- 1T/2T modes are software programmable on the fly
- Instruction-set compatible with MCS-51
- Internal OSC with range 1MHz – 24MHz
- 16K bytes on-chip flash program memory
- External RAM addresses up to 64K bytes. Standard 12T interface for external RAM access.
- 256 bytes RAM as standard 8052, plus 1K 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)
- 36 GPIOs(PDIP 40)，40 GPIOs(PLCC 44/PQFP 44，44 GPIOs(LQFP 48)
- GPIOs can select four Type(quasi-bidirectional、push-pull、open drain、input-only)，default is quasi-bidirectional(pull-up)
- 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
- 4-channel 16-bit compare /capture /load functions
- 8-channel 10-bit analog-to-digital converter (ADC)
- On–chip flash memories support ISP/IAP/ICP and EEPROM functions.
- ISP service program space configurable in N*256 byte (N=0 to 16) size.
- On-chip in-circuit emulator (ICE) function with On-Chip Debugger（OCD）
- EMI reduction mode (ALE output inhibited).
- Fast multiplication-division unit (MDU)：16*16, 32/16, 16/16, 32-bit L/R shifting and 32-bit normalization
- Keyboard interface (KBI) on port 0 or port 2 (default) for eight more interrupts.
- LVI/LVR (LVR deglitch 500ns)
- Enhanced user code protection
- Power management unit for idle and power down modes

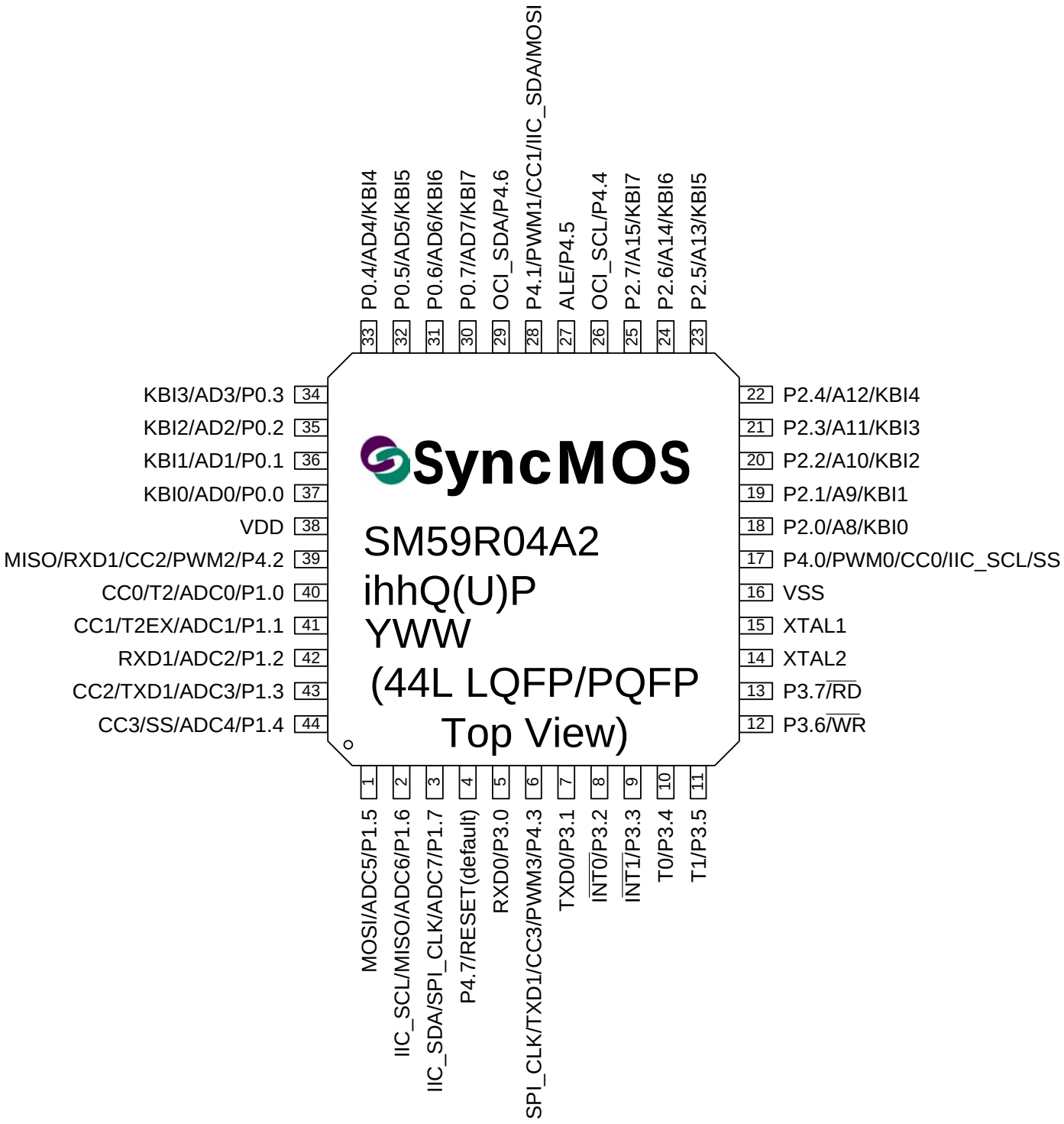
Pin Configuration

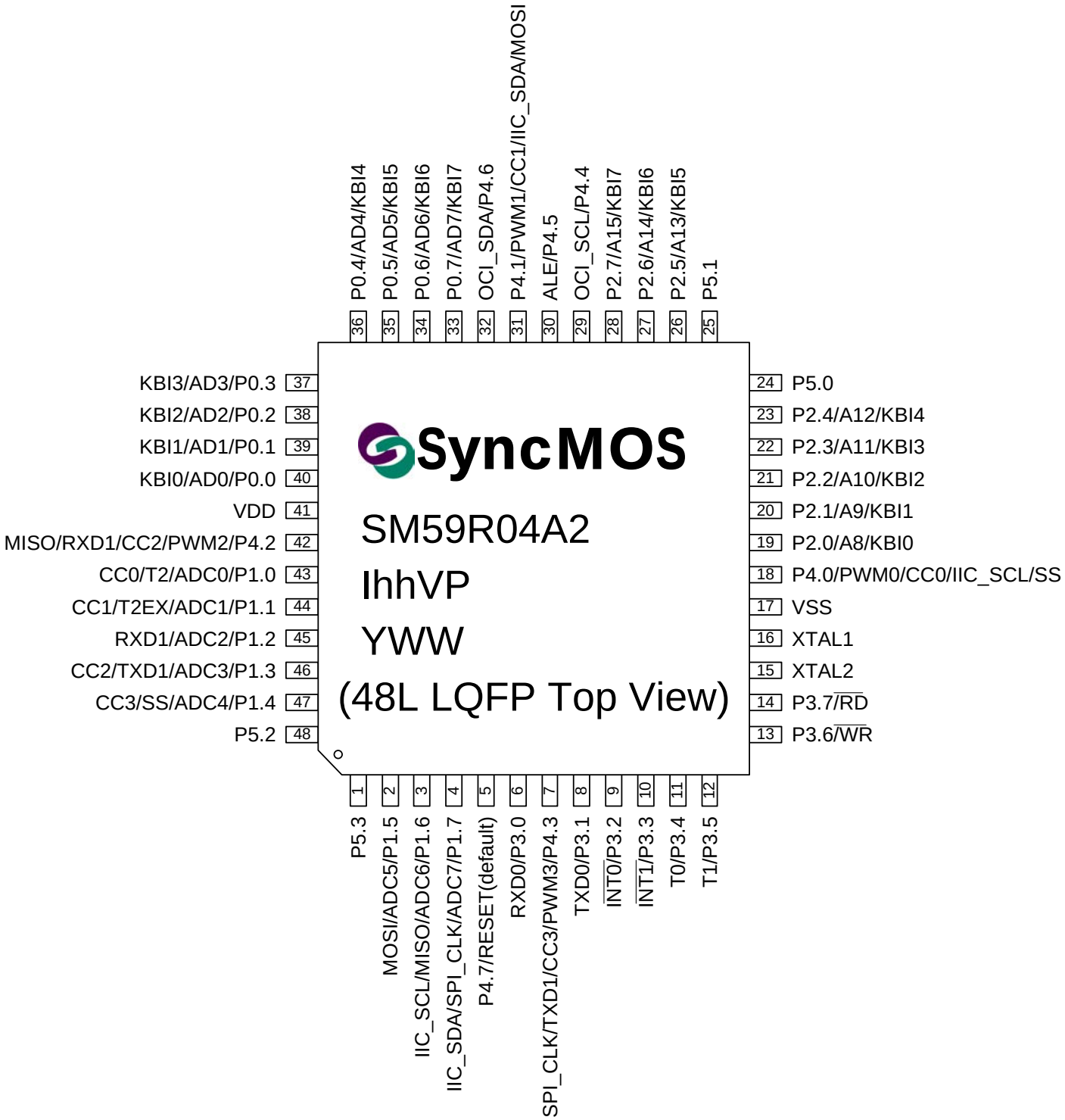


Notes：

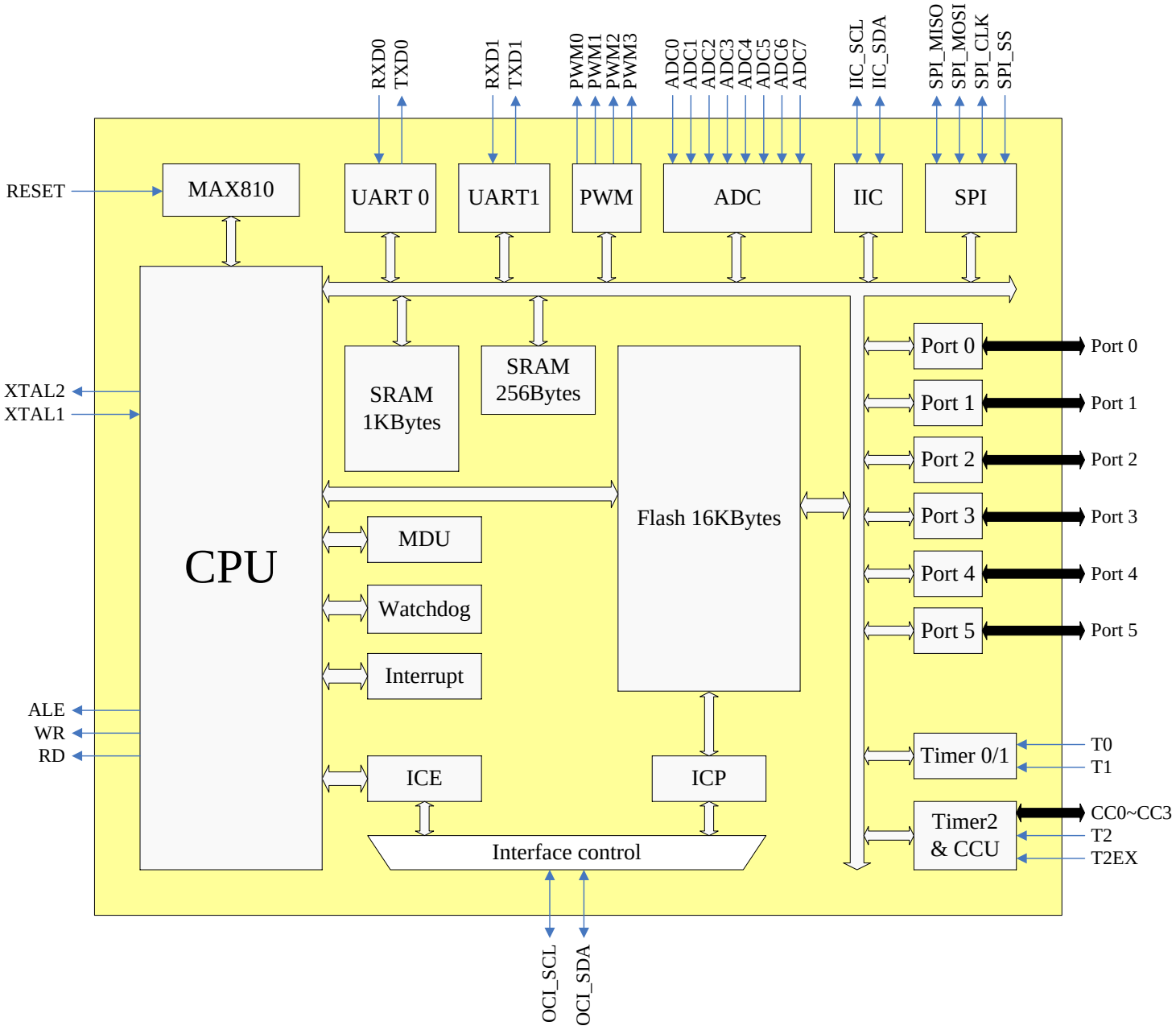
1. The pin Reset/P4.7 factory default is Reset, user must keep this pin at low during power-up. User can configure it to GPIO (P4.7) by a flash programmer.
2. To avoid accidentally entering ISP-Mode(refer to section 19.4), care must be taken not asserting pulse signal at P3.0 during power-up while P2.6、P2.7、P4.3 are set to high.
3. To apply ICP function, OSI_SDA/P4.6 and OCI_SCL/P4.7 must be set to Bi-direction mode if they are configured as GPIO in system.


P1.4/ADC4/SS/CC3
P1.3/ADC3/TXD1/CC2
P1.2/ADC2/RXD1
P1.1/ADC1/T2EX/CC1
P1.0/ADC0/T2/CC0
P4.2/PWM2/CC2/RXD1/MISO
VDD
P0.0/AD0/KBI0
P0.1/AD1/KBI1
P0.2/AD2/KBI2
P0.3/AD3/KBI3
6 5 4 3 2 1 44 43 42 41 40
MOSI/ADC5/P1.5 7
IIC_SCL/MISO/ADC6/P1.6 8
IIC_SDA/SPI_CLK/ADC7/P1.7 9
P4.7/RESET(default) 10
RXD0/P3.0 11
SPI_CLK/TXD1/CC3/PWM3/P4.3 12
TXD0/P3.1 13
INT0/P3.2 14
INT1/P3.3 15
T0/P3.4 16
T1/P3.5 17
SyncMOS
SM59R04A2
ihhJP
YWW
(44L PLCC Top View)
39 P0.4/AD4/KBI4
38 P0.5/AD5/KBI5
37 P0.6/AD6/KBI6
36 P0.7/AD7/KBI7
35 OCI_SDA/P4.6
34 P4.1/PWM1/CC1/IIC_SDA/MOSI
33 ALE/P4.5
32 OCI_SCL/P4.4
31 P2.7/A15/KBI7
30 P2.6/A14/KBI6
29 P2.5/A13/KBI5
18 19 20 21 22 23 24 25 26 27 28
WR/P3.6
RD/P3.7
XTAL2
XTAL1
VSS
SS/IIC_SCL/CC0/PWM0/P4.0
KBI0/A8/P2.0
KBI1/A9/P2.1
KBI2/A10/P2.2
KBI3/A11/P2.3
KBI4/A12/P2.4


P0.4/AD4/KBI4 33
P0.5/AD5/KBI5 32
P0.6/AD6/KBI6 31
P0.7/AD7/KBI7 30
OCI_SDA/P4.6 29
P4.1/PWM1/CC1/IIC_SDA/MOSI 28
ALE/P4.5 27
OCI_SCL/P4.4 26
P2.7/A15/KBI7 25
P2.6/A14/KBI6 24
P2.5/A13/KBI5 23
KBI3/AD3/P0.3 34
KBI2/AD2/P0.2 35
KBI1/AD1/P0.1 36
KBI0/AD0/P0.0 37
VDD 38
MISO/RXD1/CC2/PWM2/P4.2 39
CC0/T2/ADC0/P1.0 40
CC1/T2EX/ADC1/P1.1 41
RXD1/ADC2/P1.2 42
CC2/TXD1/ADC3/P1.3 43
CC3/SS/ADC4/P1.4 44
SyncMOS
SM59R04A2
ihhQ(U)P
YWW
(44L LQFP/PQFP
Top View)
22 P2.4/A12/KBI4
21 P2.3/A11/KBI3
20 P2.2/A10/KBI2
19 P2.1/A9/KBI1
18 P2.0/A8/KBI0
17 P4.0/PWM0/CC0/IIC_SCL/SS
16 VSS
15 XTAL1
14 XTAL2
13 P3.7/RD
12 P3.6/WR
1 MOSI/ADC5/P1.5
2 IIC_SCL/MISO/ADC6/P1.6
3 IIC_SDA/SPI_CLK/ADC7/P1.7
4 P4.7/RESET(default)
5 RXD0/P3.0
6 SPI_CLK/TXD1/CC3/PWM3/P4.3
7 TXD0/P3.1
8 INT0/P3.2
9 INT1/P3.3
10 T0/P3.4
11 T1/P3.5


P0.4/AD4/KBI4 36
P0.5/AD5/KBI5 35
P0.6/AD6/KBI6 34
P0.7/AD7/KBI7 33
OCI_SDA/P4.6 32
P4.1/PWM1/CC1/IIC_SDA/MOSI 31
ALE/P4.5 30
OCI_SCL/P4.4 29
P2.7/A15/KBI7 28
P2.6/A14/KBI6 27
P2.5/A13/KBI5 26
P5.1 25
KBI3/AD3/P0.3 37
KBI2/AD2/P0.2 38
KBI1/AD1/P0.1 39
KBI0/AD0/P0.0 40
VDD 41
MISO/RXD1/CC2/PWM2/P4.2 42
CC0/T2/ADC0/P1.0 43
CC1/T2EX/ADC1/P1.1 44
RXD1/ADC2/P1.2 45
CC2/TXD1/ADC3/P1.3 46
CC3/SS/ADC4/P1.4 47
P5.2 48
SyncMOS
SM59R04A2
IhhVP
YWW
(48L LQFP Top View)
24 P5.0
23 P2.4/A12/KBI4
22 P2.3/A11/KBI3
21 P2.2/A10/KBI2
20 P2.1/A9/KBI1
19 P2.0/A8/KBI0
18 P4.0/PWM0/CC0/IIC_SCL/SS
17 VSS
16 XTAL1
15 XTAL2
14 P3.7/RD
13 P3.6/WR
1 P5.3
2 MOSI/ADC5/P1.5
3 IIC_SCL/MISO/ADC6/P1.6
4 IIC_SDA/SPI_CLK/ADC7/P1.7
5 P4.7/RESET(default)
6 RXD0/P3.0
7 SPI_CLK/TXD1/CC3/PWM3/P4.3
8 TXD0/P3.1
9 INT0/P3.2
10 INT1/P3.3
11 T0/P3.4
12 T1/P3.5

Block Diagram



Pin Description

40L PDIP	44L PLCC	44L PQFP	48L LQFP	Symbol	I/O	Description
	1	39	42	P4.2/PWM2/CC2/RXD1/MISO	I/O	Bit 2 of port 4 & PWM Channel 2 & Timer 2 compare/capture Channel 2 & Serial interface channel 1 receive data & SPI interface Serial Data Master Input or Slave Output pin
1	2	40	43	P1.0/ADC0/T2/CC0	I/O	Bit 0 of port 1 & ADC input channel 0 & Timer 2 external input clock & Timer 2 compare/capture Channel 0
2	3	41	44	P1.1/ADC1/T2EX/CC1	I/O	Bit 1 of port 1 & ADC input channel 1 & Timer 2 capture trigger & Timer 2 compare/capture Channel 1
3	4	42	45	P1.2/ADC2/RXD1	I/O	Bit 2 of port 1 & ADC input channel 2 & Serial interface channel 1 receive data
4	5	43	46	P1.3/ADC3/TXD1/CC2	I/O	Bit 3 of port 1 & ADC input channel 3 & Serial interface channel 1 transmit data or receive clock in mode 0 & Timer 2 compare/capture Channel 2
5	6	44	47	P1.4/ADC4/SS/CC3	I/O	Bit 4 of port 1 & ADC input channel 4 & SPI interface Slave Select pin & Timer 2 compare/capture Channel 3
			48	P5.2	I/O	Bit 2 of port 5
			1	P5.3	I/O	Bit 3 of port 5
6	7	1	2	P1.5/ADC5/MOSI	I/O	Bit 5 of port 1 & ADC input channel 5 & SPI interface Serial Data Master Output or Slave Input pin
7	8	2	3	P1.6/ADC6/MISO/IIC_SCL	I/O	Bit 6 of port 1 & ADC input channel 6 & SPI interface Serial Data Master Input or Slave Output pin & IIC SCL pin
8	9	3	4	P1.7/ADC7/SPI_CLK/IIC_SDA	I/O	Bit 7 of port 1 & ADC input channel 7 & SPI interface Clock pin & IIC SDA pin
9	10	4	5	RESET(default)/P4.7	I/O	Reset pin (default)& Bit 7 of port 4
10	11	5	6	P3.0/RXD0	I/O	Bit 0 of port 3 & Serial interface channel 0 receive/transmit data
	12	6	7	P4.3/PWM3/CC3/TXD1/SPI_CLK	I/O	Bit 3 of port 4 & PWM Channel 3 & Timer 2 compare/capture Channel 3 & Serial interface channel 1 transmit data or receive clock in mode 0 & SPI interface Clock pin
11	13	7	8	P3.1/TXD0	I/O	Bit 1 of port 3 & Serial interface channel 0 transmit data or receive clock in mode 0
12	14	8	9	P3.2/#INT0	I/O	Bit 2 of port 3 & External interrupt 0
13	15	9	10	P3.3/#INT1	I/O	Bit 3 of port 3 & External interrupt 1
14	16	10	11	P3.4/T0	I/O	Bit 4 of port 3 & Timer 0 external input
15	17	11	12	P3.5/T1	I/O	Bit 5 of port 3 & Timer 1 external input
16	18	12	13	P3.6/#WR	I/O	Bit 6 of port 3 & external memory write signal
17	19	13	14	P3.7/#RD	I/O	Bit 7 of port 3 & external memory read signal
18	20	14	15	XTAL2	O	Crystal output
19	21	15	16	XTAL1	I	Crystal input
20	22	16	17	VSS		Power supply
	23	17	18	P4.0/PWM0/CC0/IIC_SCL/SS	I/O	Bit 0 of port 4 & PWM Channel 0 & Timer 2 compare/capture Channel 0 & IIC SCL pin & SPI interface Slave Select pin
21	24	18	19	P2.0 /A8/KBI0	I/O	Bit 0 of port 2 & Bit 8 of external memory address & KBI interrupt 0
22	25	19	20	P2.1 /A9/KBI1	I/O	Bit 1 of port 2 & Bit 9 of external memory address & KBI interrupt 1
23	26	20	21	P2.2 /A10/KBI2	I/O	Bit 2 of port 2 & Bit 10 of external memory address & KBI interrupt 2
24	27	21	22	P2.3 /A11/KBI3	I/O	Bit 3 of port 2 & Bit 11 of external memory address & KBI interrupt 3
25	28	22	23	P2.4 /A12/KBI4	I/O	Bit 4 of port 2 & Bit 12 of external memory address & KBI interrupt 4

40L PDIP	44L PLCC	44L PQFP	48L LQFP	Symbol	I/O	Description
			24	P5.0	I/O	Bit 0 of port 5
			25	P5.1	I/O	Bit 1 of port 5
26	29	23	26	P2.5 /A13/KBI5	I/O	Bit 5 of port 2 & Bit 13 of external memory address & KBI interrupt 5
27	30	24	27	P2.6 /A14/KBI6	I/O	Bit 6 of port 2 & Bit 14 of external memory address & KBI interrupt 6
28	31	25	28	P2.7 /A15/KBI7	I/O	Bit 7 of port 2 & Bit 15 of external memory address & KBI interrupt 7
29	32	26	29	OCI_SCL/P4.4	I/O	On-Chip Instrumentation Clock I/O pin of ICE and ICP functions & Bit 4 of port 4
30	33	27	30	ALE/P4.5	I/O	Address latch enable & Bit 5 of port 4
	34	28	31	P4.1/PWM1/CC 1/IIC_SDA/MOS I	I/O	Bit 1 of port 4 & PWM Channel 1 & Timer 2 compare/capture Channel 1 & IIC SDA pin & SPI interface Serial Data Master Output or Slave Input pin
31	35	29	32	OCI_SDA/P4.6	I/O	On-Chip Instrumentation Command and data I/O pin synchronous to OCI_SCL in ICE and ICP functions & Bit 6 of port 4
32	36	30	33	P0.7/AD7/KBI7	I/O	Bit 7 of port 0 & Bit 7 of external memory address/ data & KBI interrupt 7
33	37	31	34	P0.6/AD6/KBI6	I/O	Bit 6 of port 0 & Bit 6 of external memory address/ data & KBI interrupt 6
34	38	32	35	P0.5/AD5/KBI5	I/O	Bit 5 of port 0 & Bit 5 of external memory address/ data & KBI interrupt 5
35	39	33	36	P0.4/AD4/KBI4	I/O	Bit 4 of port 0 & Bit 4 of external memory address/ data & KBI interrupt 4
36	40	34	37	P0.3/AD3/KBI3	I/O	Bit 3 of port 0 & Bit 3 of external memory address/ data & KBI interrupt 3
37	41	35	38	P0.2/AD2/KBI2	I/O	Bit 2 of port 0 & Bit 2 of external memory address/ data & KBI interrupt 2
38	42	36	39	P0.1/AD1/KBI1	I/O	Bit 1 of port 0 & Bit 1 of external memory address/ data & KBI interrupt 1
39	43	37	40	P0.0/AD0/KBI0	I/O	Bit 0 of port 0 & Bit 0 of external memory address/ data & KBI interrupt 0
40	44	38	41	VDD		Power supply

Special Function Register (SFR)

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

Hex\Bin	X000	X001	X010	X011	X100	X101	X110	X111	Bin/Hex
F8	IICS	IICCTL	IICA1	IICA2	IICRWD	IICS2			**FF**
F0	B	SPIC1	SPIC2	SPITXD	SPIRXD	SPIS		TAKEY	**F7**
E8	P4	MD0	MD1	MD2	MD3	MD4	MD5	ARCON	**EF**
E0	ACC	ISPFAH	ISPFAL	ISPFD	ISPFC		LVC	SWRES	**E7**
D8	P5		P3M0	P3M1	P4M0	P4M1	P5M0	P5M1	**DF**
D0	PSW		P0M0	P0M1	P1M0	P1M1	P2M0	P2M1	**D7**
C8	T2CON	CCCON	CRCL	CRCH	TL2	TH2	PWMMDH	PWMMDL	**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	ADCCS	**AF**
A0	P2								**A7**
98	S0CON	S0BUF	IEN2	S1CON	S1BUF	S1RELL			**9F**
90	P1	AUX		KBLS	KBE	KBF	KBD		**97**
88	TCON	TMOD	TL0	TL1	TH0	TH1		IFCON	**8F**
80	P0	SP	DPL	DPH	DPL1	DPH1	RCON	PCON	**87**
Hex\Bin	**X000**	**X001**	**X010**	**X011**	**X100**	**X101**	**X110**	**X111**	**Bin/Hex**

Note: Special Function Registers reset values and description for SM59R04A2

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
RCON	86h	00h	Internal RAM control register
PCON	87h	40h	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
AUX	91h	00h	Auxiliary register

Register	Location	Reset value	Description
KBLS	93h	00h	Keyboard level selector Register
KBE	94h	00h	Keyboard input enable Register
KBF	95h	00h	Keyboard interrupt flag Register
KBD	96h	00h	Keyboard interface De-bounce control 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
IEN0	A8h	00h	Interrupt Enable Register 0
IP0	A9h	00h	Interrupt Priority Register 0
S0RELL	AAh	00h	Serial Port 0, Reload Register, low byte
ADCC1	ABh	00h	ADC Control 1 Register
ADCC2	ACh	00h	ADC Control 2 Register
ADCDH	ADh	00h	ADC data high byte
ADCDL	AEh	00h	ADC data low byte
ADCCS	AFh	00h	ADC clock select
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
WDTC	B6h	04h	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	00h	Serial Port 0, Reload Register, high byte
S1RELH	BBh	00h	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
CCH2	C5h	00h	Compare/Capture Register 2, high byte
CCL3	C6h	00h	Compare/Capture Register 3, low byte

Register	Location	Reset value	Description
CCH3	C7h	00h	Compare/Capture Register 3, high byte
T2CON	C8h	00h	Timer 2 Control
CCCON	C9h	00h	Compare/Capture 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
PWMMDH	CEh	00h	PWM Max Data Register, high byte.
PWMMDL	CFh	FFh	PWM Max Data Register, low byte.
PSW	D0h	00h	Program Status Word
P0M0	D2h	00h	Port 0 output mode 0
P0M1	D3h	00h	Port 0 output mode 1
P1M0	D4h	00h	Port 1 output mode 0
P1M1	D5h	00h	Port 1 output mode 1
P2M0	D6h	00h	Port 2 output mode 0
P2M1	D7h	00h	Port 2 output mode 1
P5	D8h	0Fh	Port 5
P3M0	DAh	00h	Port 3 output mode 0
P3M1	DBh	00h	Port 3 output mode 1
P4M0	DCh	00h	Port 4 output mode 0
P4M1	DDh	00h	Port 4 output mode 1
P5M0	DEh	00h	Port 5 output mode 0
P5M1	DFh	00h	Port 5 output mode 1
ACC	E0h	00h	Accumulator
ISPFAH	E1h	FFh	ISP Flash Address-High register
ISPFAL	E2h	FFh	ISP Flash Address-Low register
ISPFD	E3h	FFh	ISP Flash Data register
ISPFC	E4h	00h	ISP Flash control register
LVC	E6h	00h	Low voltage control register
SWRES	E7h	00h	Software Reset register
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

Register	Location	Reset value	Description
SPIRXD	F4h	00h	SPI receive data buffer
SPIS	F5h	40h	SPI status register
TAKEY	F7h	00h	Time Access Key register
IICS	F8h	00h	IIC status register
IICCTL	F9h	04h	IIC control register
IICA1	FAh	A0h	IIC channel 1 Address 1 register
IICA2	FBh	60h	IIC channel 1 Address 2 register
IICRWD	FCh	00h	IIC channel 1 Read / Write Data buffer
IICS2	FDh	00h	IIC status2 register

Function Description

1. General Features

SM59R04A2 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 16KB Flash memory via writer or In-System Programming (ISP)。

1.2. IO Pads

The SM59R04A2 has six I/O ports: Port 0, Port 1, Port 2, Port 3, Port 4, and Port 5. Ports 0, 1, 2, 3, 4 are 8-bit ports and Port 5 is a 4-bit port. These are: quasi-bidirectional (standard 8051 port outputs), push-pull, open drain, and input-only. As description in section 5.

The OCI_SCL、ALE、OCI_SDA and RESET can be configured as I/O ports P4.4、P4.5、P4.6 and P4.7 by writer or in ISP mode.

All the pins on P0 ~ P5 are with slew rate adjustment 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 SM59R04A2'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. SM59R04A2 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. 1-1.

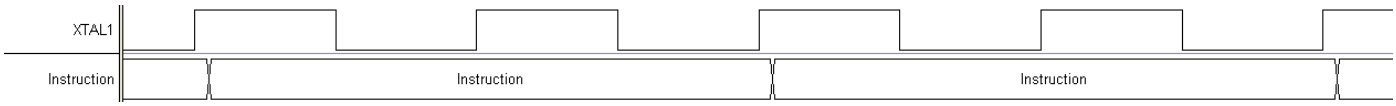


Fig. 1-1(a): The waveform of internal instruction signal in 2T mode

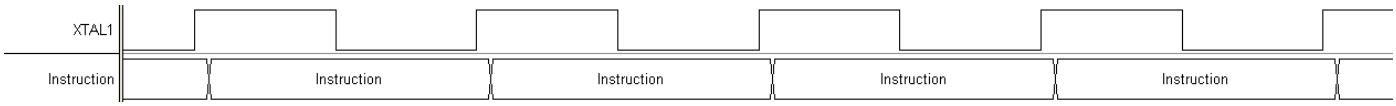


Fig. 1-1(b): The waveform of internal instruction signal in 1T mode

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

1.4.1. Hardware RESET function

SM59R04A2 provides on-chip hardware RESET mechanism ，the reset duration is programmable by writer or ISP。

Internal Reset time
25ms (default)
200ms
100ms
50ms
16ms
8ms
4ms

1.4.2. Software RESET function

SM59R04A2 provides one software reset mechanism to reset whole chip. To perform a software reset, the firmware must write three specific values 55h, AAh and 5Ah sequentially to the TAKEY register to enable the Software Reset register (SWRES) write attribute. After SWRES register obtain the write authority, the firmware can write FFh to the SWRES register. The hardware will decode a reset signal that "OR" with the other hardware reset. The SWRES register is self-reset at the end of the software reset procedure.

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
Software Reset function											
TAKEY	Time Access Key register	F7h	TAKEY [7:0]								00H
SWRES	Software Reset register	E7h	SWRES [7:0]								00H

1.4.3. Time Access Key register (TAKEY)

Mnemonic: TAKEY **Address: F7H**

7	6	5	4	3	2	1	0	Reset
TAKEY [7:0]								00H

Software reset register (SWRES) is read-only by default; software must write three specific values 55h, AAh and 5Ah sequentially to the TAKEY register to enable the SWRES register write attribute. That is:

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah
```

1.4.4. Software Reset register (SWRES)

Mnemonic: SWRES **Address: E7H**

7	6	5	4	3	2	1	0	Reset
SWRES [7:0]								00H

SWRES [7:0]: Software reset register bit. These 8-bit is self-reset at the end of the reset procedure.
SWRES [7:0] = FFh, software reset.
SWRES [7:0] = 00h ~ FEh, MCU no action.

1.4.5. Example of software reset

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah    ; enable SWRES write attribute
MOV SWRES, #FFh    ; software reset MCU
```

1.5. Clocks

The default clock is the external crystal. 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 internal clock sources are from the on-chip RC-Oscillator with programmable frequency outputs as table 1-1,the clock source can set by writer or ICP.

Table 1-1: Selection of clock source

Clock source
external crystal**(default)**
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

2. Instruction Set

All SM59R04A2 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 SM59R04A2 Microcontroller core.

Table 2-1: Arithmetic operations

Mnemonic	Description	Code	Bytes	Cycles
ADD A,Rn	Add register to accumulator	28-2F	1	1
ADD A,direct	Add direct byte to accumulator	25	2	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 external RAM (8-bit addr.) to A	E2-E3	1	3
MOVX A,@DPTR	Move external RAM (16-bit addr.) to A	E0	1	3
MOVX @Ri,A	Move A to external RAM (8-bit addr.)	F2-F3	1	4
MOVX @DPTR,A	Move A to external 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 iump	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 immed. to reg. and jump if not equal	B8-BF	3	4
CJNE @Ri,#data rel	Compare immed. to ind. 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 SM59R04A2 memory structure follows general 8052 structure. It is integrate the expanded 1KB data memory and 16KB program memory.

3.1. Program Memory

The SM59R04A2 has 16KB on-chip flash memory which can be used as general program memory or EEPROM, on which include up to 4K byte specific ISP service program memory space. The address range for the 16K byte is $0000 to $3FFF. The address range for the ISP service program is $3000 to $3FFF. The ISP service program size can be partitioned as N blocks of 256 byte (N=0 to 16). When N=0 means no ISP service program space available, total 16K byte memory used as program memory. When N=1 means address $3F00 to $3FFF reserved for ISP service program. When N=2 means memory address $3E00 to $3FFF reserved for ISP service program…etc. Value N can be set and programmed into SM59R04A2 by the writer or ICP. It can be used to record any data as EEPROM. The procedure of this EEPROM application function is described in the section 19 on internal ISP.

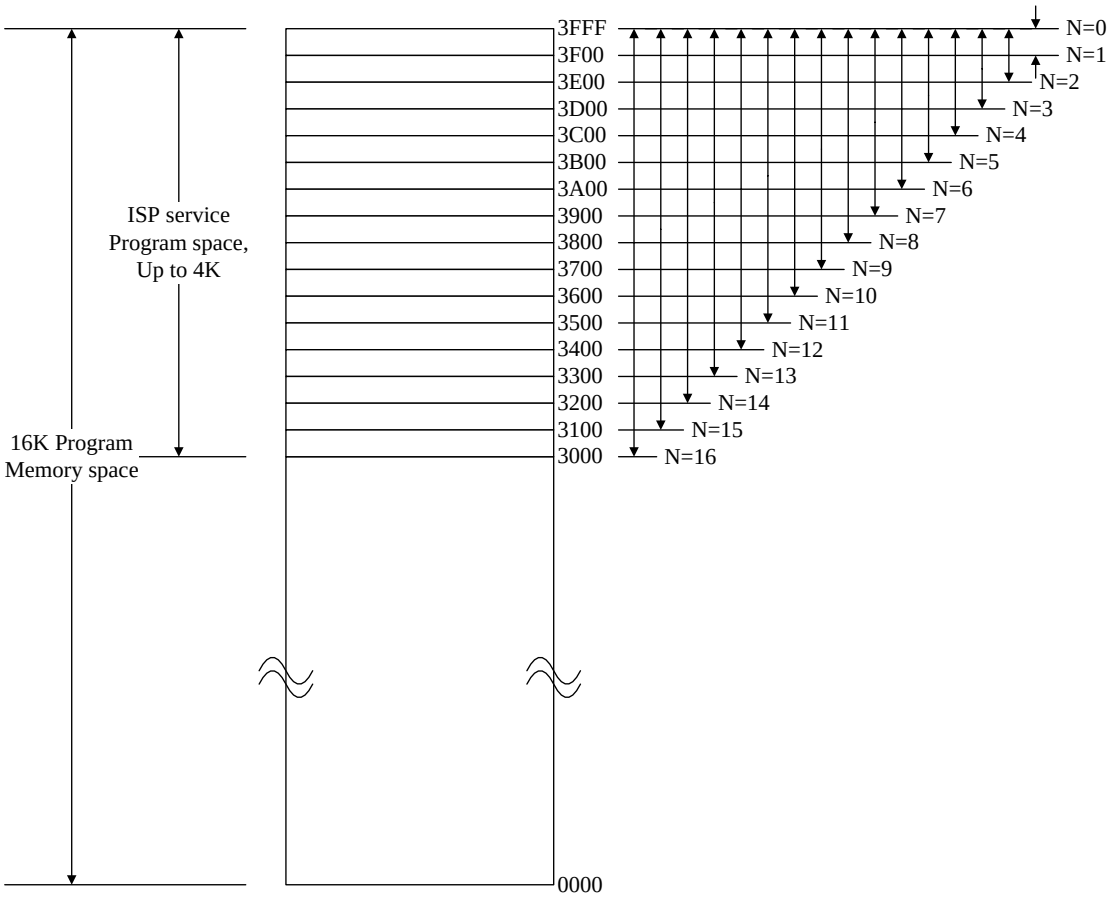


Fig. 3-1: SM59R04A2 programmable Flash

3.2. Data Memory

The SM59R04A2 has 1K + 256Bytes on-chip SRAM, 256 Bytes of it are the same as general 8052 internal memory structure while the expanded 1KBytes on-chip SRAM can be accessed by external memory addressing method (by instruction MOVX.)

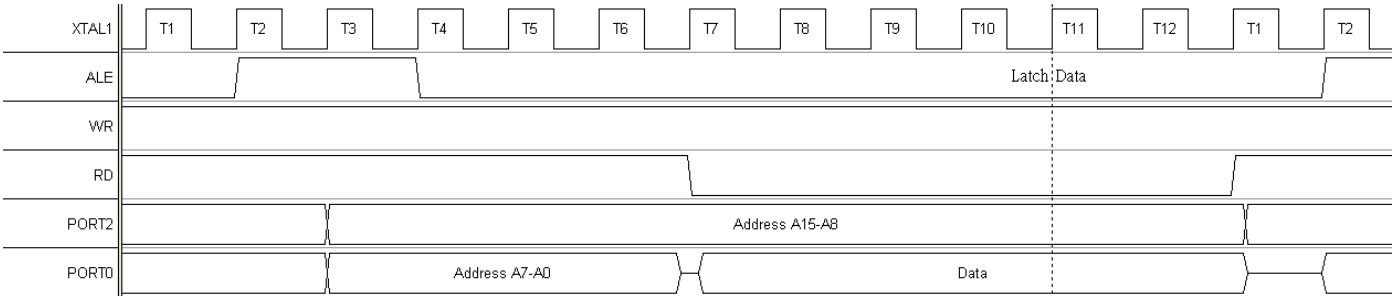


Fig 3-2 (a)：External memory access as read

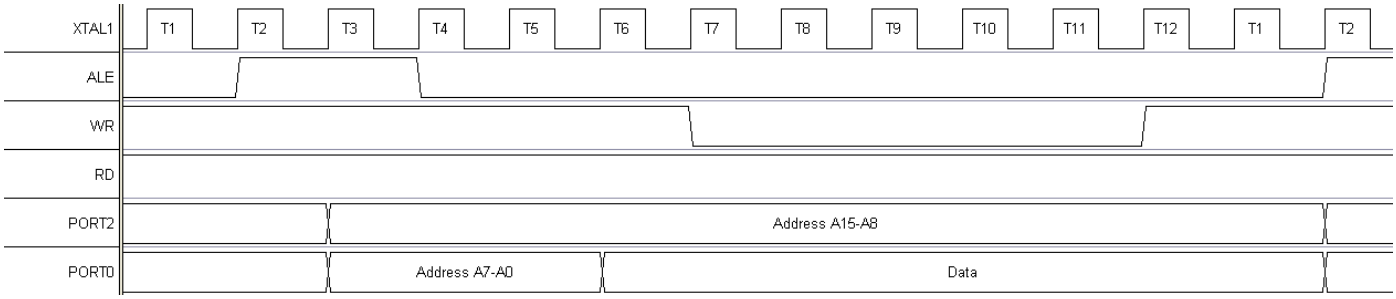


Fig 3-2 (b)：External memory access as write

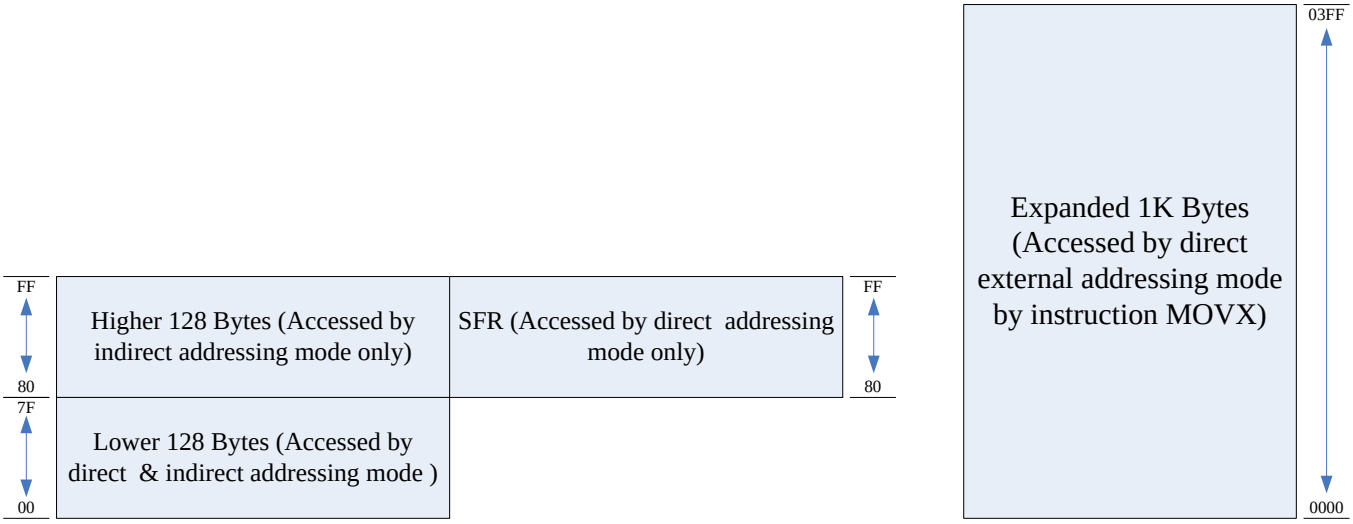


Fig. 3-3: RAM architecture

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

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

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

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

3.2.3. Data memory - Expanded 1024 bytes ($0000 to $03FF)

From external address 0000h to 03FFh is the on-chip expanded SRAM area, total 1K Bytes. This area can be accessed by external direct addressing mode (by instruction MOVX).

If the address of instruction MOVX @DPTR is larger then 03FFh, the SM59R04A2 will generate the external memory control signal automatically.

The address space of instruction MOVX @Ri, i=0, 1 is determined by RCON [7:0] of special function register $86 RCON (internal RAM control register). The default setting of RCON [7:0] is 00h (page0). One page of data RAM is 256 bytes.

When EMEN = 0, the internal 1K expanded RAM is enabled. If access memory space is more than 1K byte, the value of RCON is sent to Port2 to access external RAM.

When EMEN = 1, the internal 1K expanded RAM is disabled. The value of RCON is invalid and high byte address is decided by register context of Port2 register P2 [7:0].

MOVX @Ri, A MOVX A,@Ri	0 ≦ RCON[7:0] ≦ 3	4 ≦ RCON [7:0] ≦ 255
EMEN = 0	Addr [15:8] <= RCON[7:0]	Port2 [7:0] <= RCON[7:0]
EMEN = 1	Port2 [7:0] <= P2 [7:0]	Port2 [7:0] <= P2 [7:0]

4. CPU Engine

The SM59R04A2 engine is composed of four components:

a. Control unit
b. Arithmetic – logic unit
c. Memory control unit
d. RAM and SFR control unit

The SM59R04A2 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
8051 Core											
ACC	Accumulator	E0h	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	00H
B	B register	F0h	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	00H
PSW	Program status word	D0h	CY	AC	F0	RS[1:0]		OV	PSW.1	P	00H
SP	Stack Pointer	81h	SP[7:0]								07H
DPL	Data pointer low 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
AUX	Auxiliary register	91h	BRGS	P4CC	P4SPI	P4UR 1	P4IIC	P0KBI	-	DPS	00H
RCON	Internal RAM control register	86h	RCON[7:0]								00H
IFCON	Interface control register	8Fh	ITS	CDPR	-	-	ALEC[1:0]		EMEN	ISPE	00H

4.1. Accumulator

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

Mnemonic: ACC **Address: E0h**

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

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

4.2. B Register

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

Mnemonic: B **Address: F0h**

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

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

4.3. Program Status Word

Mnemonic: PSW **Address: D0h**

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

CY: Carry flag.
AC: Auxiliary Carry flag for BCD operations.
F0: General purpose Flag 0 available for user.
RS[1:0]: Register bank select, used to select working register bank.

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

OV: Overflow flag.
F1: General purpose Flag 1 available for user.
P: Parity flag, affected by hardware to indicate odd/even number of “one” bits in the Accumulator, i.e. even parity.

4.4. Stack Pointer

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

Mnemonic: SP **Address: 81h**

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

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

4.5. Data Pointer

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

Mnemonic: DPL **Address: 82h**

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

DPL[7:0]: Data pointer Low 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 moves of data block. The standard DPTR is a 16-bit register that is used to address external memory or peripherals. In the SM59R04A2 core 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 LSB of AUX register (DPS).

The user switches between pointers by toggling the LSB of AUX 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: AUX **Address: 91h**

7	6	5	4	3	2	1	0	Reset
BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H

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

4.7. Internal RAM control register

SM59R04A2 has 1K byte on-chip expanded RAM which can be accessed by external memory addressing method only (By instruction MOVX). The address space of instruction MOVX @Ri, i= 0, 1 is determined by RCON [7:0] of RCON. The default setting of RCON [7:0] is 00h (page0).

Mnemonic: RCON **Address: 86h**

7	6	5	4	3	2	1	0	Reset
RCON[7:0]								00H

4.8. Interface control register

Mnemonic: IFCON **Address: 8Fh**

7	6	5	4	3	2	1	0	Reset
ITS	CDPR	-	-	ALEC[1:0]		EMEN	ISPE	00h

ITS: Instruction timing select. (default is 2T)
ITS = 0, 2T instruction mode.
ITS = 1, 1T instruction mode.
CDPR: code protect (Read Only)
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

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

ISPE: ISP function enable bit
ISPE = 1, enable ISP function
ISPE = 0, disable ISP function

5. GPIO

The SM59R04A2 has six I/O ports: Port 0, Port 1, Port 2, Port 3, Port 4, and Port 5. Ports 0, 1, 2, 3, 4 are 8-bit ports and Port 5 is a 4-bit port. These are: quasi-bidirectional (standard 8051 port outputs), push-pull, open drain, and input-only. Two configuration registers for each port select the output type for each port pin. All I/O port pins on the SM59R04A2 may be configured by software to one of four types on a pin-by-pin basis, shown as below:

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
I/O port function register											
P0M0	Port 0 output mode 0	D2h	P0M0 [7:0]								00H
P0M1	Port 0 output mode 1	D3h	P0M1[7:0]								00H
P1M0	Port 1 output mode 0	D4h	P1M0[7:0]								00H
P1M1	Port 1 output mode 1	D5h	P1M1[7:0]								00H
P2M0	Port 2 output mode 0	D6h	P2M0[7:0]								00H
P2M1	Port 2 output mode 1	D7h	P2M1[7:0]								00H
P3M0	Port 3 output mode 0	DAh	P3M0[7:0]								00H
P3M1	Port 3 output mode 1	DBh	P3M1[7:0]								00H
P4M0	Port 4 output mode 0	DCh	P4M0[7:0]								00H
P4M1	Port 4 output mode 1	DDh	P4M1[7:0]								00H
P5M0	Port 5 output mode 0	DEh	-				P5M0[3:0]				00H
P5M1	Port 5 output mode 1	DFh	-				P5M1[3:0]				00H

PxM1.y	PxM0.y	Port output mode
0	0	Quasi-bidirectional (standard 8051 port outputs) (pull-up)
0	1	Push-pull
1	0	Input only (high-impedance)
1	1	Open drain

The OCI_SCL、ALE、OCI_SDA and RESET can be configured as I/O ports P4.4、P4.5、P4.6 and P4.7 by writer or in ISP mode.

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	D8h					P5.3	P5.2	P5.1	P5.0	0Fh
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: D8h**

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

P5.3~ 0: Port5 [3] ~ 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	-	-	-	-	STOP	IDLE	40H
ARCON	Arithmetic Control register	EFh	MDEF	MDOV	SLR	SC[4:0]					00H
MD0	Multiplication/Division Register 0	E9h	MD0[7:0]								00H
MD1	Multiplication/Division Register 1	EAh	MD1[7:0]								00H
MD2	Multiplication/Division Register 2	EBh	MD2[7:0]								00H
MD3	Multiplication/Division Register 3	ECh	MD3[7:0]								00H
MD4	Multiplication/Division Register 4	EDh	MD4[7:0]								00H
MD5	Multiplication/Division Register 5	EEh	MD5[7:0]								00H

6.1. Operating registers of the MDU

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. Control register is ARCON. Any calculation of the MDU overwrites 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 phase three.

The error flag is set when:
1. Phase two in process and write access to mdx registers (restart or interrupt calculations)

The error flag is reset only if:
Phase two 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:

1. Division by Zero
2. Multiplication with a result greater then 0000FFFFh
3. 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 Phase one)

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 normalize sc.0 – sc.4 contains the number of normalizing shifts performed. When sc.4 – sc.0 ≠ 0, shift operation is started. The number of shifts performed is determined by the count written to sc.4 to sc.0.
sc.4 – MSB ... sc.0 – LSB

6.2. Operation of the MDU

The operation of the MDU consists of three phases:

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

The type of calculation the MDU has to perform is selected following the order in which the mdx registers are written to.

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 Multiplicator 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 Multiplicator High	ARCON start conversion

A write to md0 is the first transfer to be done in any case. Next writes must be done as shown in table 6.1 to determine MDU operation. Last write finally starts selected operation.

6.2.2. 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 clear at next calculation.

Mnemonic: PCON **Address: 87h**

7	6	5	4	3	2	1	0	Reset
SMOD	MDUF	-	-	-	-	STOP	IDLE	40h

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

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

Read out sequence of the first MDx registers is not critical but the last read (from MD5 - division and MD3 - multiplication, shift and normalizing) determines the end of a whole calculation (end of phase three).

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.

6.3. Normalizing

All reading zeroes of integers variables in registers MD0 to MD3 are removed by shift left operations. 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, which were done.

6.4. Shifting

SLR bit (ARCON.5) has to contain the shift direction, and ARCON.4 to ARCON.0 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.

7. Timer 0 and Timer 1

The SM59R04A2 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 machine cycles, which means that it counts up after every 12 periods of the clk signal.

In counter mode, the register is incremented when the falling edge is observed at the corresponding input pin T0or T1. Since it takes 2 machine cycles to recognize a 1-to-0 event, the maximum input count rate is 1/2 of the oscillator frequency. There are no restrictions on the duty cycle, however to ensure proper recognition of 0 or 1 state, an input should be stable for at least 1 machine cycle.

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

Mnemonic	Description	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											
AUX	Auxiliary register	91h	BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H
T2CON	Timer 2 control	C8h	T2PS	CC0FR	-	T2R[1:0]		T2CM	T2I[1:0]		00h
CCCON	Compare/Capture Control	C9h	CCI3	CCI2	CCI1	CCI0	CCF3	CCF2	CCF1	CCF0	00H
CCEN	Compare/Capture Enable register	C1h	COCAH3	COCAL3	COCAH2	COCAL2	COCAH1	COCAL1	COCAH0	COCAL0	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: AUX **Address: 91h**

7	6	5	4	3	2	1	0	Reset
BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H

P4CC: P4CC = 0 – Capture/Compare function on P1.
P4CC = 1 – Capture/Compare function on P4.

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: CCCON **Address: C9h**

7	6	5	4	3	2	1	0	Reset
CCI3	CCI2	CCI1	CCI0	CCF3	CCF2	CCF1	CCF0	00H

CCI3: Compare/Capture 3 interrupt control bit.
"1" is enable.
CCI2: Compare/Capture 2 interrupt control bit.
"1" is enable.
CCI1: Compare/Capture 1 interrupt control bit.
"1" is enable.
CCI0: Compare/Capture 0 interrupt control bit.
"1" is enable.
CCF3: Compare/Capture 3 flag set by hardware. This flag can be cleared by software.
CCF2: Compare/Capture 2 flag set by hardware. This flag can be cleared by software.
CCF1: Compare/Capture 1 flag set by hardware. This flag can be cleared by software.
CCF0: Compare/Capture 0 flag set by hardware. This flag can be cleared by software.
Compare/Capture interrupt share T2 interrupt vector.

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.

COCAH2	COCAL2	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)

COCAH0	COCAL0	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 clock (Fosc/12) or in every 24 clock (Fosc/24) 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.

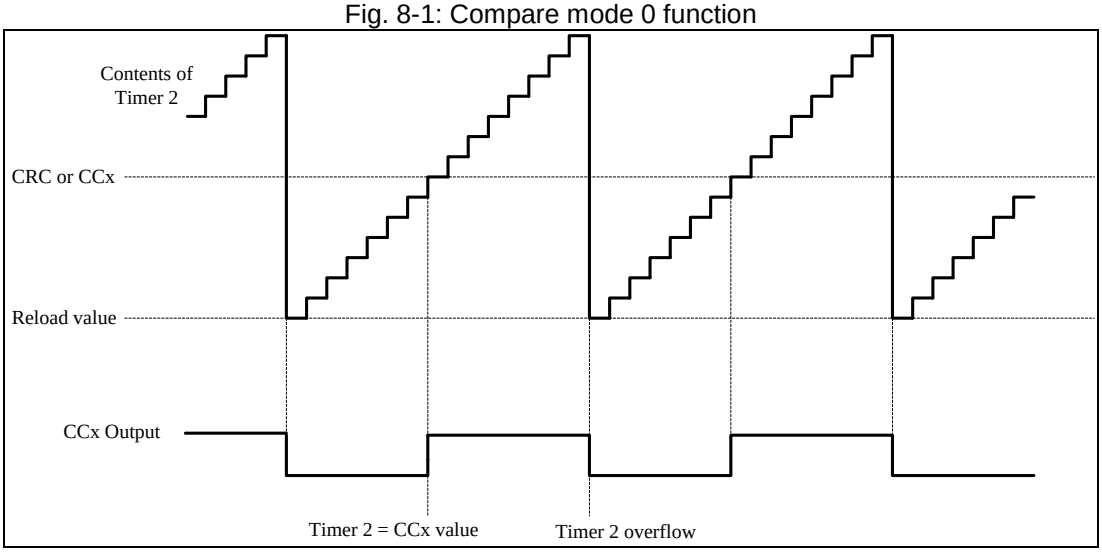
8.2. Compare function

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.

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.

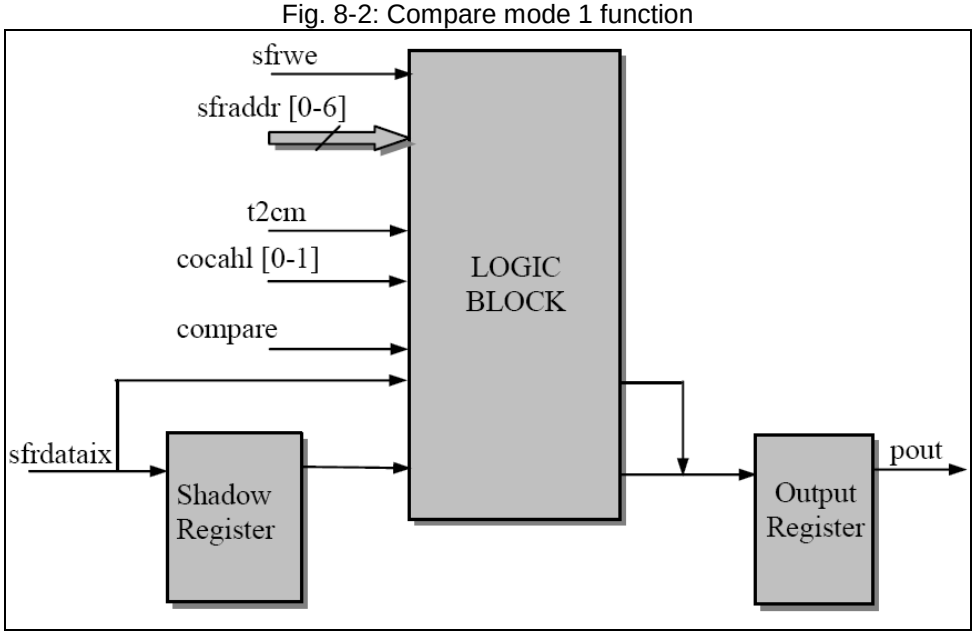
Fig. 8-1: Compare mode 0 function

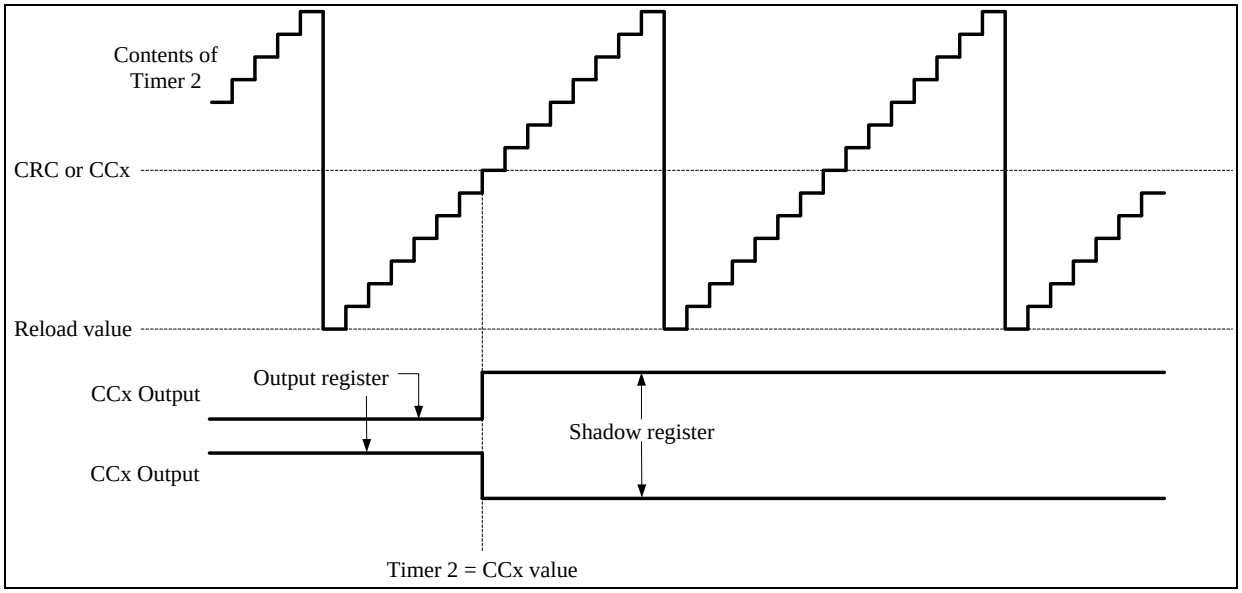


8.2.2. Compare Mode 1

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.

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.

9. Serial interface 0 and 1

There are two serial interfaces for data communication in SM59R04A2, 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	-	-	-	-	STOP	IDLE	40h
AUX	Auxiliary register	91h	BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	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: AUX **Address: 91h**

7	6	5	4	3	2	1	0	Reset
BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H

P4UR1: P4UR1 = 0 – Serial interface 1 function on P1.
P4UR1 = 1 – Serial interface 1 function on P4.

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.

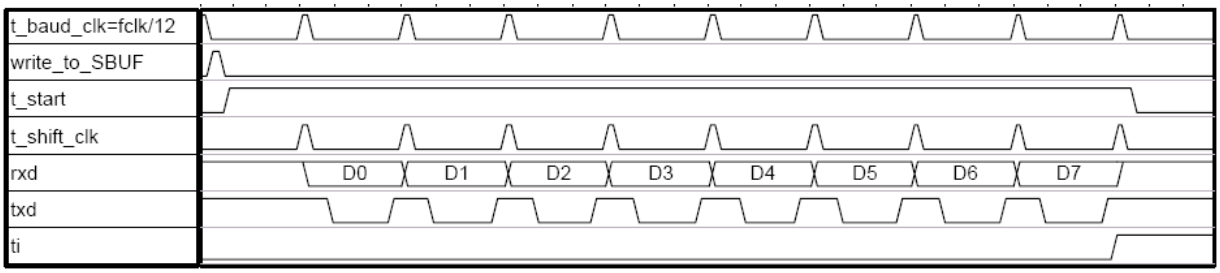


Fig. 9-1: Transmit mode 0 for Serial 0

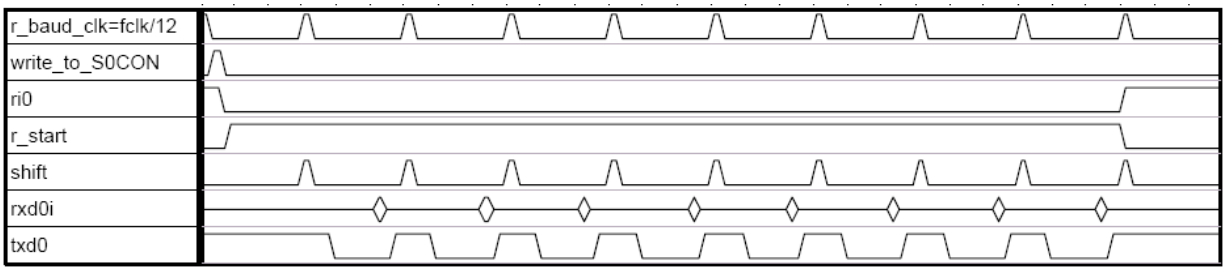


Fig. 9-2: Receive mode 0 for Serial 0

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 use to specify the desired baud rate.

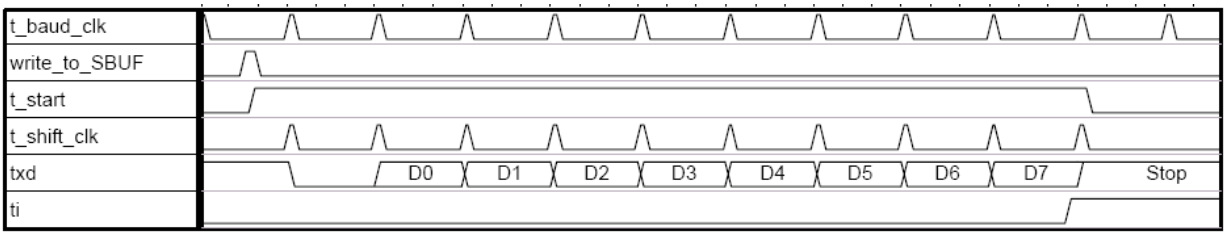


Fig. 9-3: Transmit mode 1 for Serial 0

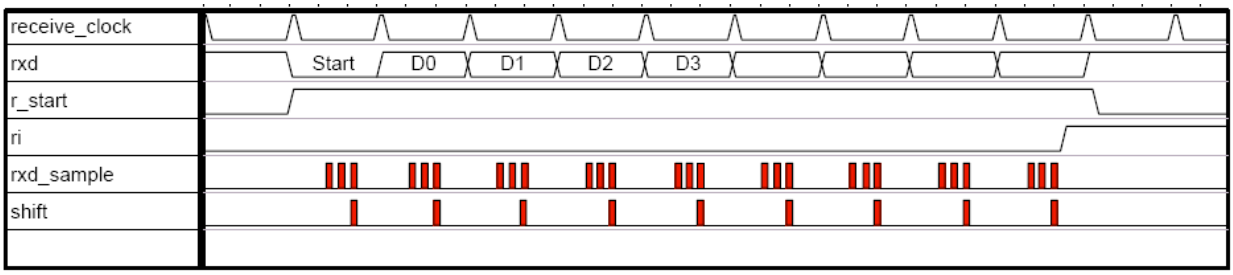


Fig. 9-4: Receive mode 1 for Serial 0

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.

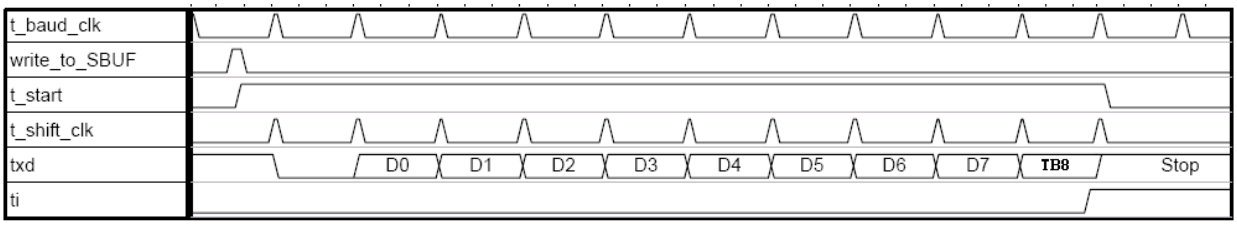


Fig. 9-5: Transmit modes 2 and 3 for Serial 0

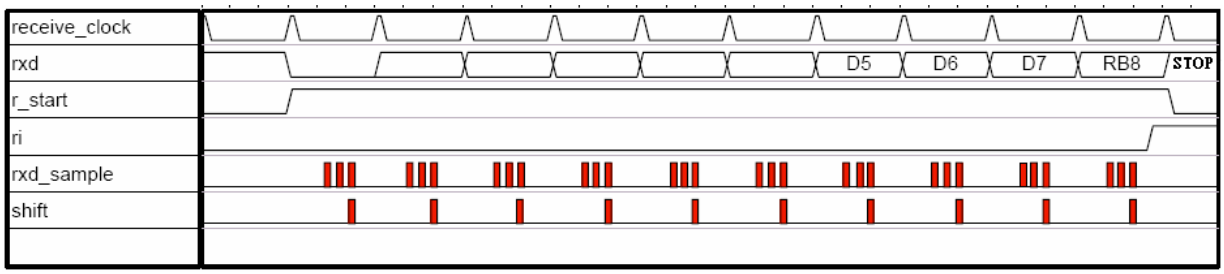


Fig. 9-6: Receive modes 2 and 3 for Serial 0

9.2. Serial interface 1

The interrupt vector is 83h.

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.

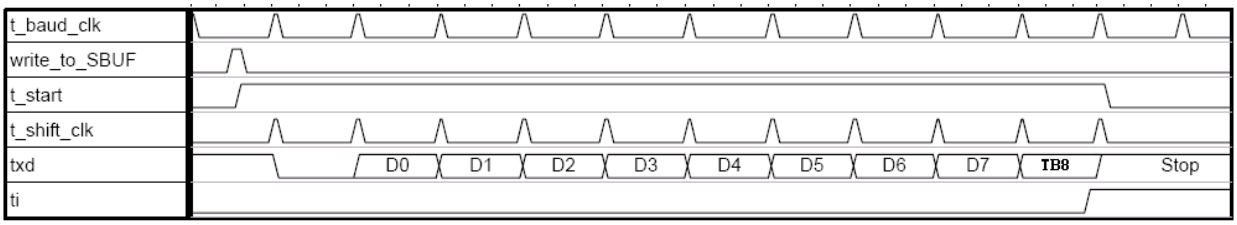


Fig. 9-7: Transmit mode A for Serial 1

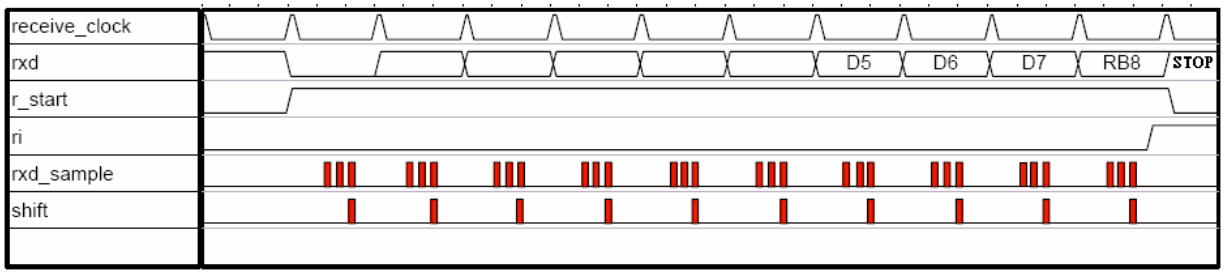


Fig. 9-8: Receive mode A for Serial 1

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.

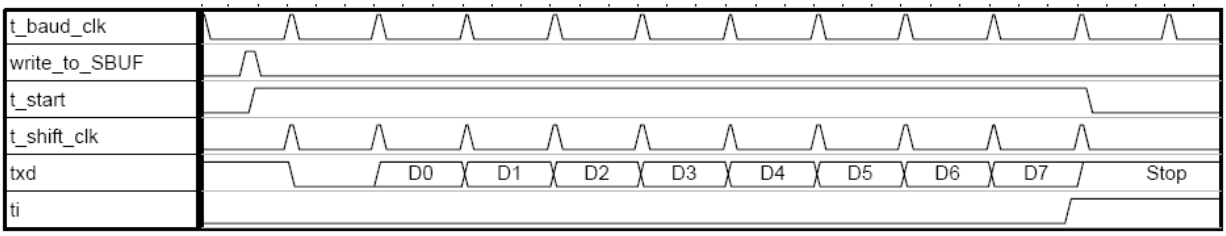


Fig. 9-9: Transmit mode B for Serial 1

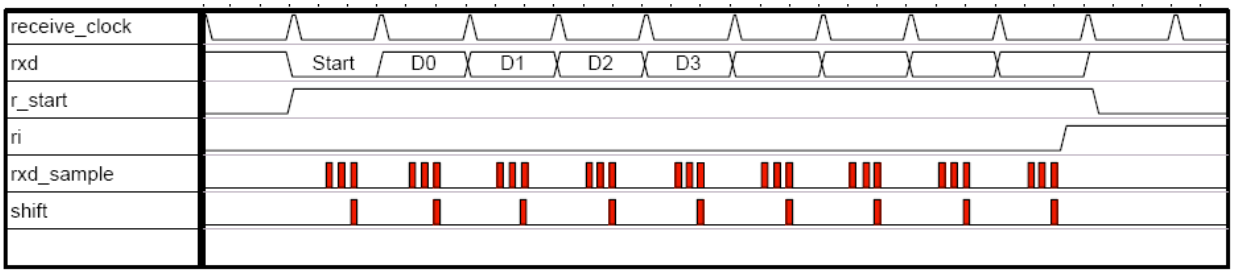


Fig. 9-10: Receive mode B for Serial 1

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 BRGS = 0 (in SFR AUX):

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

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

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

9.4.2. Serial interface 1 modes A and B

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

9.5. Clock source for baud rate

In case of application with higher clock precision requirement, external crystal is usually recommended clock source.

10. Watchdog timer

The Watch Dog Timer (WDT) is an 8-bit free-running counter that generate reset signal if the counter overflows. The WDT is useful for systems which are susceptible to noise, power glitches, or electronics discharge which causing software dead loop or runaway. The WDT function can help user software recover from abnormal software condition. The WDT is different from Timer0, Timer1 and Timer2 of general 8052. To prevent a WDT reset can be done by software periodically clearing the WDT counter. User should check WDTF bit of WDTC register whenever un-predicted reset happened. After an external reset the watchdog timer is disabled and all registers are set to zeros.

The watchdog timer has a free running on-chip RC oscillator (250KHz±20%). The WDT will keep on running even after the system clock has been turned off (for example, in sleep mode). During normal operation or sleep mode, a WDT time-out (if enabled) will cause the MCU to reset. The WDT can be enabled or disabled any time during the normal mode. Please refer the WDTE bit of WDTC register. The default WDT time-out period is approximately 16.38ms (WDTM [3:0] = 0100b).

The WDT has selectable divider input for the time base source clock. To select the divider input, the setting of bit3 ~ bit0 (WDTM [3:0]) of Watch Dog Timer Control Register (WDTC) should be set accordingly.

$$WDTCLK = \frac{250\text{KHz}}{2^{\text{WDTM}}}$$

$$\text{Watchdog reset time} = \frac{256}{\text{WDTCLK}}$$

Table 10.1 WDT time-out period

WDTM [3:0]	Divider (250 KHz RC oscillator in)	Time period @ 250KHz
0000	1	1.02ms
0001	2	2.05ms
0010	4	4.10ms
0011	8	8.19ms
0100	16	16.38ms (default)
0101	32	32.77ms
0110	64	65.54ms
0111	128	131.07ms
1000	256	262.14ms
1001	512	524.29ms
1010	1024	1.05s
1011	2048	2.10s
1100	4096	4.19s
1101	8192	8.39s
1110	16384	16.78s
1111	32768	33.55s

When MCU is reset, the MCU will be read WDTEN control bit status. When WDTEN bit is set to 1, the watchdog function will be disabled no matter what the WDTE bit status is. When WDTEN bit is clear to 0, the watchdog function will be enabled if WDTE bit is set to 1 by program. User can to set WDTEN on the writer or ISP.

The program can enable the WDT function by programming 1 to the WDTE bit premise that WDTEN control bit is clear to 0. After WDTE set to 1, the 8 bit-counter starts to count with the selected time base source clock which set by WDTM [3:0]. It will generate a reset signal when overflows. The WDTE bit will be cleared to 0 automatically when MCU been reset, either hardware reset or WDT reset.

Once the watchdog is started it cannot be stopped. User can refreshed the watchdog timer to zero by writing 0x55 to Watch Dog Timer refresh Key (WDTK) register. This will clear the content of the 8-bit counter and let the counter re-start

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

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 or power on reset.

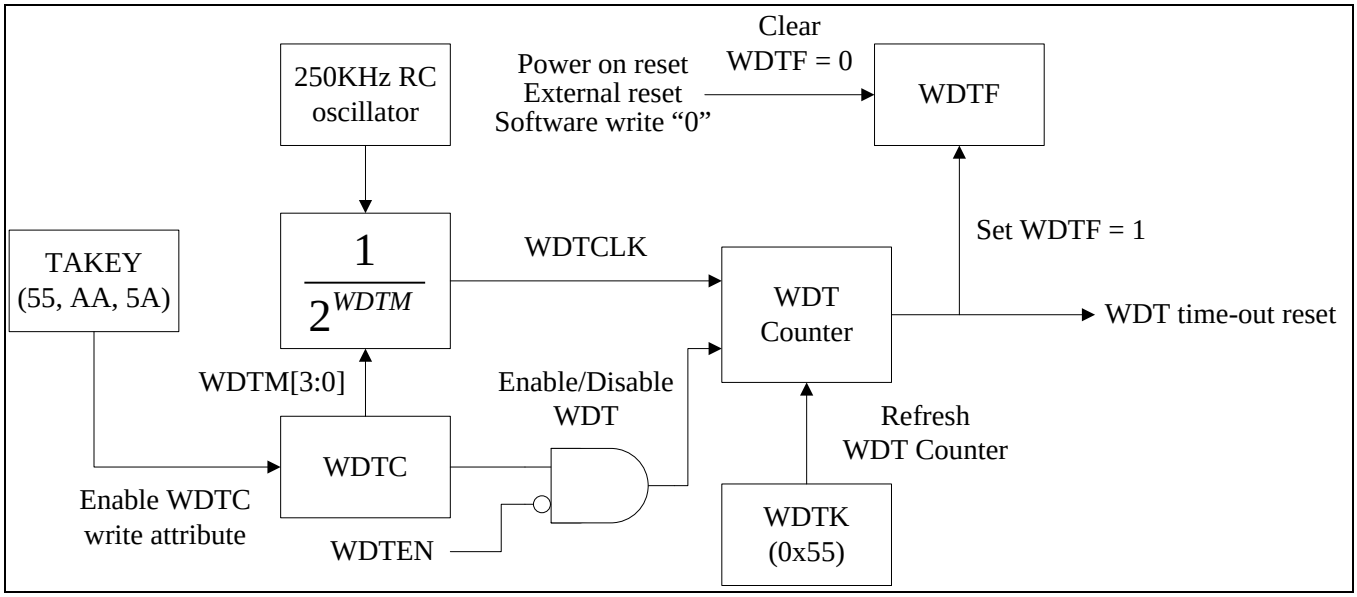


Fig. 10-1: Watchdog timer block diagram

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
Watchdog Timer											
TAKEY	Time Access Key register	F7h	TAKEY [7:0]								00H
WDTC	Watchdog timer control register	B6h	WDTF	-	WDTE	-	WDTM [3:0]				04H
WDTK	Watchdog timer refresh key	B7h	WDTK[7:0]								00H

Mnemonic: TAKEY **Address: F7h**

7	6	5	4	3	2	1	0	Reset
TAKEY [7:0]								00H

Watchdog timer control register (WDTC) is read-only by default; software must write three specific values 55h, AAh and 5Ah sequentially to the TAKEY register to enable the WDTC write attribute. That is:

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah
```

Mnemonic: WDTC **Address: B6h**

7	6	5	4	3	2	1	0	Reset
WDTF	-	WDTE	-	WDTM [3:0]				04H

WDTF: Watchdog timer reset flag.
When MCU is reset by watchdog, WDTF flag will be set to one by hardware. This flag clear by software or external reset or power on reset.

WDTE: Control bit used to enable Watchdog timer.
The WDTE bit can be used only if WDTEN is "0". If the WDTEN bit is "0", then WDT

can be disabled / enabled by the WDTE bit.
0: Disable WDT.
1: Enable WDT.
The WDTE bit is not used if WDTEN is "1". That is, if the WDTEN bit is "1", WDT is always disabled no matter what the WDTE bit status is. The WDTE bit can be read and written.

WDTM [3:0]: WDT clock source divider bit. Please see table 7.8.1 to reference the WDT time-out period.

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, and then the watchdog timer will be cleared to zero.

For example, if enable WDT and select time-out reset period is 327.68ms.
First, programming the information block OP3 bit7 WDTEN to “0”.
Secondly,

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah          ; enable WDTC write attribute.
MOV WDTC, #28h           ; Set WDTM [3:0] = 1000b. Set WDTE =1 to enable WDT
                         ; function.
.
.
.
MOV WDTK, #55h           ; Clear WDT timer to 0.
```

11. Interrupt

The SM59R04A2 provides 13 interrupt sources with four priority levels. Each source has its own request flag(s) located in a special function register. Each interrupt requested by the corresponding flag could individually be enabled or disabled by the enable bits in SFR's IEN0, IEN1, and IEN2.

When the interrupt occurs, the engine 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 next when interrupt occurred.

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

Interrupt response will require a varying amount of time depending on the state of microcontroller when the interrupt occurs. If microcontroller is performing an interrupt service with equal or greater priority, the new interrupt will not be invoked. In other cases, the response time depends on current instruction. 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
PWMIF – PWM interrupt	0043h	8
SPIIF – SPI interrupt	004Bh	9
ADCIF – A/D converter interrupt	0053h	10
KBIIF – keyboard Interface interrupt	005Bh	11
LVIIF – Low Voltage Interrupt	0063h	12
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	IELVI	IEKBI	IEADC	IESPI	IEPWM	00h
IEN2	Interrupt Enable 2 register	9Ah	-	-	-	-	-	-	-	ES1	00h
IRCON	Interrupt request register	C0H	EXF2	TF2	IICIF	LVIIF	KBIIF	ADCIF	SPIIF	PWMIF	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.
ET2=1 – Enable Timer 2 overflow or external reload interrupt.
ES0: ES0=0 – Disable Serial channel 0 interrupt.
ES0=1 – Enable Serial channel 0 interrupt.
ET1: ET1=0 – Disable Timer 1 overflow interrupt.
ET1=1 – Enable Timer 1 overflow interrupt.
EX1: EX1=0 – Disable external interrupt 1.
EX1=1 – Enable external interrupt 1.
ET0: ET0=0 – Disable Timer 0 overflow interrupt.
ET0=1 – Enable Timer 0 overflow interrupt.
EX0: EX0=0 – Disable external interrupt 0.
EX0=1 – Enable external interrupt 0.

Interrupt Enable 1 register(IEN1)

Mnemonic: IEN1 **Address: B8h**

7	6	5	4	3	2	1	0	Reset
EXEN2	-	IEIIC	IELVI	IEKBI	IEADC	IESPI	IEPWM	00h

EXEN2: Timer 2 reload interrupt enable.
EXEN2 = 0 – Disable Timer 2 external reload interrupt.
EXEN2 = 1 – Enable Timer 2 external reload interrupt.
IEIIC: IIC interrupt enable.
IEIICS = 0 – Disable IIC interrupt.
IEIICS = 1 – Enable IIC interrupt.
IELVI: LVI interrupt enable.
IELVI = 0 – Disable LVI interrupt.
IELVI = 1 – Enable LVI interrupt.
IEKBI: KBI interrupt enable.
IEKBI = 0 – Disable KBI interrupt.
IEKBI = 1 – Enable KBI interrupt.
IEADC: A/D converter interrupt enable

IEADC = 0 – Disable ADC interrupt.
IEADC = 1 – Enable ADC interrupt.
IESPI: SPI interrupt enable.
IESPI = 0 – Disable SPI interrupt.
IESPI = 1 – Enable SPI interrupt.
IEPWM: PWM interrupt enable.
IEPWM = 0 – Disable PWM interrupt.
IEPWM = 1 – Enable PWM interrupt.

Interrupt Enable 2 register(IEN2)

Mnemonic: IEN2 **Address: 9Ah**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	-	ES1	00h

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

Mnemonic: IRCON **Address: C0h**

7	6	5	4	3	2	1	0	Reset
EXF2	TF2	IICIF	LVIIF	KBIIF	ADCIF	SPIIF	PWMIF	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.
LVIIF: LVI interrupt flag.
KBIIF: KBI interrupt flag.
ADCIF: A/D converter end interrupt flag.
SPIIF: SPI interrupt flag.
PWMIF: PWM 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	PWM interrupt
Timer 0 interrupt	-	SPI interrupt
External interrupt 1	-	ADC interrupt
Timer 1 interrupt	-	KBI interrupt
Serial channel 0 interrupt	-	LVI interrupt
Timer 2 interrupt	-	IIC interrupt

Each group of interrupt sources can be programmed individually to one of four priority levels by setting or clearing one bit in the special function register IP0 and one in IP1. If requests of the same priority level will be 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	PWM 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	-	KBI interrupt
IP1.4, IP0.4	Serial channel 0 interrupt	-	LVI 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	
PWM interrupt	
Timer 0 interrupt	
SPI interrupt	
External interrupt 1	
ADC interrupt	
Timer 1 interrupt	
KBI interrupt	
Serial channel 0 interrupt	
LVI 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	-	-	-	-	STOP	IDLE	40h

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

12.1. Idle mode

Setting the IDLE bit of PCON register invokes the IDLE mode. The IDLE mode leaves internal clocks and peripherals running. Power consumption drops because the CPU is not active. The CPU can exit the IDLE state with any interrupts or a reset.

12.2. Stop mode

Setting the STOP bit of PCON register invokes the STOP mode. All internal clocking in this mode is turn off. The CPU will exit this state only if interrupts asserted from external INT0/1, KBI, and LVI or a hardware reset by WDT and LVR.

13. Pulse Width Modulation (PWM)

SM59R04A2 provides four-channel PWM outputs.
The interrupt vector is 43h.

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	PWMCS[2:0]			-	PWM3EN	PWM2EN	PWM1EN	PWM0EN	00H
PWMD0H	PWM 0 Data register high byte	BCh	PWMP0	-	-	-	-	-	PWMD0[9:8]		00H
PWMD0L	PWM 0 Data register low byte	BDh	PWMD0[7:0]								00H
PWMD1H	PWM 1 Data register high byte	BEh	PWMP1	-	-	-	-	-	PWMD1[9:8]		00H
PWMD1L	PWM 1 Data register low byte	BFh	PWMD1[7:0]								00H
PWMD2H	PWM 2 Data register high byte	B1h	PWMP2	-	-	-	-	-	PWMD2[9:8]		00H
PWMD2L	PWM 2 Data register low byte	B2h	PWMD2[7:0]								00H
PWMD3H	PWM 3 Data register high byte	B3h	PWMP3	-	-	-	-	-	PWMD3[9:8]		00H
PWMD3L	PWM 3 Data register low byte	B4h	PWMD3[7:0]								00H
PWMMDH	PWM Max Data register high byte	CEh	-	-	-	-	-	-	PWMMD[9:8]		00H
PWMMDL	PWM Max Data register low byte	CFh	PWMMD[7:0]								FFH

Mnemonic: PWMC **Address: B5h**

7	6	5	4	3	2	1	0	Reset
PWMCS[2:0]			-	PWM3EN	PWM2EN	PWM1EN	PWM0EN	00H

PWMCS[2:0]: PWM clock select.

PWMCS [2:0]	Mode
000	Fosc
001	Fosc/2
010	Fosc/4
011	Fosc/6
100	Fosc/8
101	Fosc/12
110	Timer 0 overflow
111	Timer 0 external input (P3.4/T0)

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 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
PWMP0	-	-	-	-	-	PWMD0[9:8]		00H

Mnemonic: PWMD0L **Address: BDh**

7	6	5	4	3	2	1	0	Reset
PWMD0[7:0]								00h

PWMP0: PWM channel 0 idle polarity select.
"0" – PWM channel 0 will idle low.
"1" – PWM channel 0 will idle high.
PWMD0[9:0]: PWM channel 0 data register.

Mnemonic: PWMD1H **Address: BEh**

7	6	5	4	3	2	1	0	Reset
PWMP1	-	-	-	-	-	PWMD1[9:8]		00H

Mnemonic: PWMD1L **Address: BFh**

7	6	5	4	3	2	1	0	Reset
PWMD1[7:0]								00H

PWMP1: PWM channel 1 idle polarity select.
"0" – PWM channel 1 will idle low.
"1" – PWM channel 1 will idle high.
PWMD1[9:0]: PWM channel 1 data register.

Mnemonic: PWMD2H **Address: B1h**

7	6	5	4	3	2	1	0	Reset
PWMP2	-	-	-	-	-	PWMD2[9:8]		00H

Mnemonic: PWMD2L **Address: B2h**

7	6	5	4	3	2	1	0	Reset
PWMD2[7:0]								00H

PWMP2: PWM channel 2 idle polarity select.
"0" – PWM channel 2 will idle low.
"1" – PWM channel 2 will idle high.
PWMD2[9:0]: PWM channel 2 data register.

Mnemonic: PWMD3H **Address: B3h**

7	6	5	4	3	2	1	0	Reset
PWMP3	-	-	-	-	-	PWMD3[9:8]		00H

Mnemonic: PWMD3L **Address: B4h**

7	6	5	4	3	2	1	0	Reset
PWMD3[7:0]								00H

PWMP3: PWM channel 3 idle polarity select.
"0" – PWM channel 3 will idle low.
"1" – PWM channel 3 will idle high.
PWMD3[9:0]: PWM channel 3 data register.

Mnemonic: PWMMDH **Address: CEh**

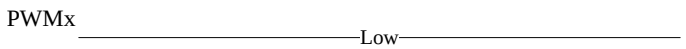
7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	PWMMD[9:8]		00H

Mnemonic: PWMMDL **Address: CFh**

7	6	5	4	3	2	1	0	Reset
PWMMD[7:0]								FFH

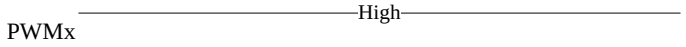
PWMMD[9:0]: PWM Max Data register.
PWM count from 0000h to PWMMD[9:0]. When PWM count data equal PWMMD[9:0] is overflow.

PWMPx = 0 & PWMDx = 00h



PWMPx = 0 & PWMDx ≠ 00h

PWMPx = 1 & PWMDx = 00h



PWMPx = 1 & PWMDx ≠ 00h

$$\text{PWM period} = \frac{\text{PWMMD} + 1}{\text{PWM clock}}$$

$$\text{Leader pulse} = \frac{\text{PWMDx}}{\text{PWM clock}}$$

14. IIC function

The IIC module uses the SCL (clock) and the SDA (data) line to communicate with external IIC interface. Its speed can be selected to 400Kbps (maximum) by software setting the IICBR [2:0] control bit. The IIC module provided 2 interrupts (RXIF, TXIF). 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.

The interrupt vector is 6Bh.

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
IIC function											
AUX	Auxiliary register	91h	BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H
IICCTL	IIC control register	F9h	IICEN		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	IICRWD[7:0]								00H
IICS2	IIC status2 register	FDh	-	-	-	-	AB_EN	BF_EN	AB_F	BF	00H

Mnemonic: AUX **Address: 91h**

7	6	5	4	3	2	1	0	Reset
BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H

P4IIC: P4IIC = 0 – IIC function on P1.
P4IIC = 1 – IIC function on P4.

Mnemonic: IICCTL **Address: F9h**

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

IICEN: Enable IIC module
IICEN = 1 is Enable
IICEN = 0 is Disable.
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 the IICA1 or IICA2(selected by MAS control bit).When module is finished to send out address, 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 set the MStart bit, the module will generate a start condition to the SDA and SCL lines and send out the calling address which is stored in the IICA1 or IICA2 (selected by MAS control bit). When software cleared this bit, the module will generate a stop condition to the SDA and SCL.

RxIF: The data Receive Interrupt Flag (RXIF) is set after the IICRWD (IIC Read Write Data Buffer) is loaded with a newly receive data.

TxIF: The data Transmit Interrupt Flag (TXIF) is set when the data of the IICRWD (IIC Read Write Data Buffer) is downloaded to the shift register.

RDR: The MCU 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: When MCU finish writing data to IICRWD, the MCU needs to set this bit to '1' to inform the IIC module to send the data in the IICRWD. After IIC module finishes sending the data from IICRWD, this bit will be cleared automatically.

RxAK: The Acknowledge Status indicate bit. When clear, it means an acknowledge signal has been received after the complete 8 bits data transmit on the bus.

TxAK: The Acknowledge status transmit bit. When received complete 8 bits data, this bit will set (NoAck) or clear (Ack) and transmit to master to indicate the receive status. Actually, it is sent as the 9th bit in one byte transmission as show in Fig. 14-1.

RW: The slave mode read (received) or wrote (transmit) on the IIC bus. When this bit is clear, the slave module received data on the IIC bus (SDA).(Slave mode only)

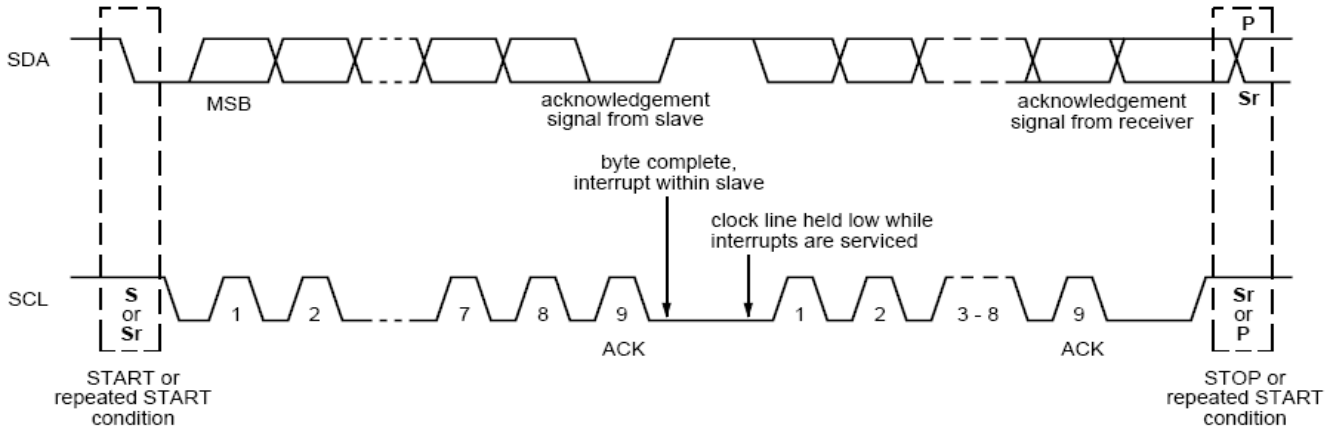


Fig. 14-1: Acknowledgement bit in the 9th bit of a byte transmission

Mnemonic: IICA1 **Address: FAH**

7	6	5	4	3	2	1	0	Reset
IICA1[7:1]							Match1 or RW1	A0H
R/W							R or R/W	

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.

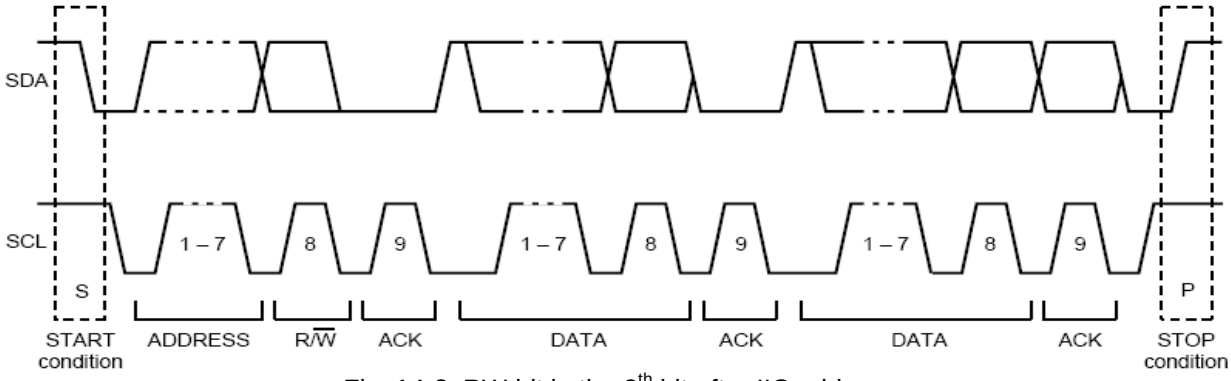


Fig. 14-2: RW bit in the 8th bit after IIC address

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.

Mnemonic: IICS2 **Address: FDH**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	AB_EN	BF_EN	AB_F	BF	00H

AB_EN: Arbitration lost enable bit. (Master mode only)
If set AB_EN bit, the hardware will check arbitration lost. Once arbitration lost occurred, hardware will return to IDLE state. If this bit is cleared, hardware will not care arbitration lost condition. Set this bit when multi-master and slave connection. Clear this bit when single master to single slave.

BF_EN: Bus busy enable bit. (Master mode only)
If set BF_EN bit, hardware will not generate a start condition to bus until BF=0. Clear this bit will always generate a start condition to bus when MStart is set. Set this bit when multi-master and slave connection. Clear this bit when single master to single slave.

AB_F: Arbitration lost bit. (Master mode only)
In multi-master condition, when send out data bit “1” but return back “0”, bus arbitration lost occurred and this bit will be set. Software need to clear this bit and check until BF=0 to resend data again.

BF: Bus busy bit. (Master mode only)
If detect scl=0 or sda=0 or bus start, this bit will be set. If detect stop and a period passed(about 4.7us), this bit will be cleared. This bit can be cleared by software to return ready state.

15. SPI function

Serial Peripheral Interface (SPI) is a synchronous protocol that allows a master device to initiate communication with slave devices.
The interrupt vector is 4Bh.

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 slave 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.

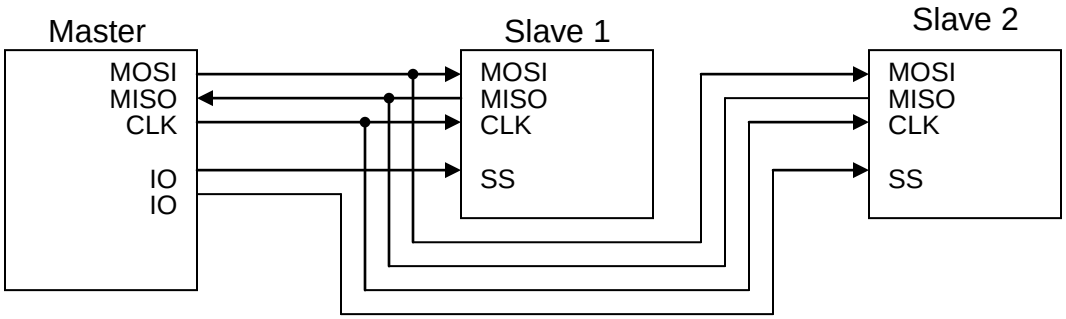


Fig. 15-1: SPI signals between master and slave devices

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

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
SPI function											
AUX	Auxiliary register	91h	BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H
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: AUX **Address: 91h**

7	6	5	4	3	2	1	0	Reset
BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H

P4SPI: P4SPI = 0 – SPI function on P1.
P4SPI = 1 – SPI function on P4.

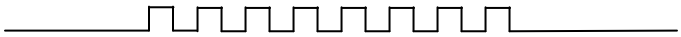
Mnemonic: SPIC1 **Address: F1h**

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

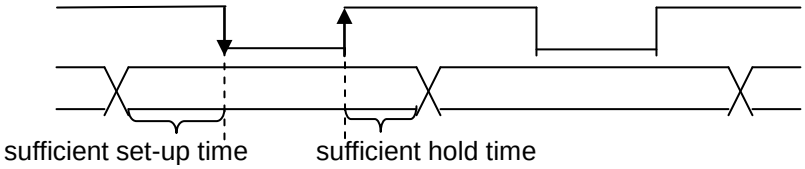
SPIEN: Enable SPI module. “1” is Enable. “0” is Disable.
SPIMSS: Master or Slave mode Select
“1” is Master mode.
“0” is Slave mode.
SPISSP: Slave Select (SS) active polarity (slave mode used only)
“1” - high active.
“0” - low active.
SPICKP: Clock idle polarity

“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, SM59R04A2 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.



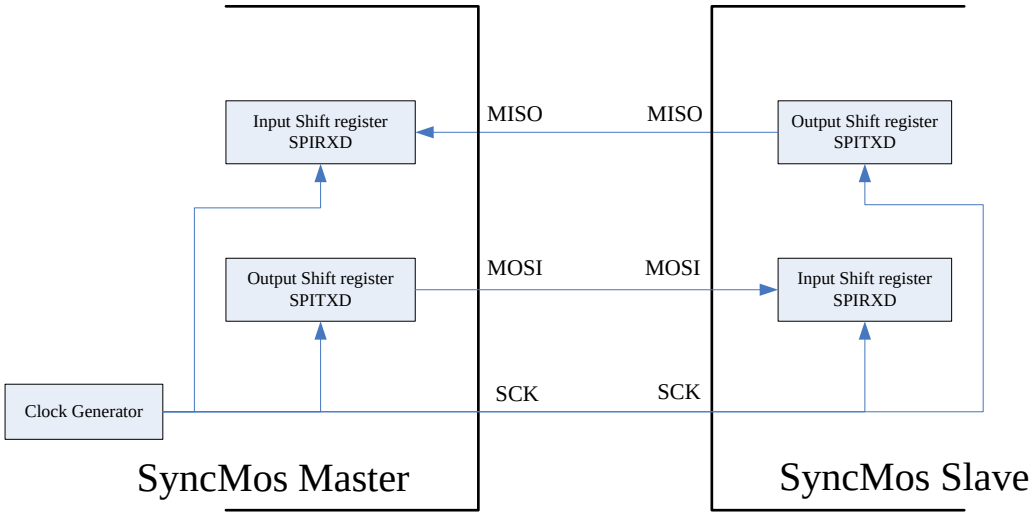
SPIBR[2:0]: SPI baud rate select (master mode used only), here Fosc is the external crystal or oscillator frequency :

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. KBI – Keyboard Interface

Keyboard interface (KBI) 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 P2 or P0 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 5Bh.

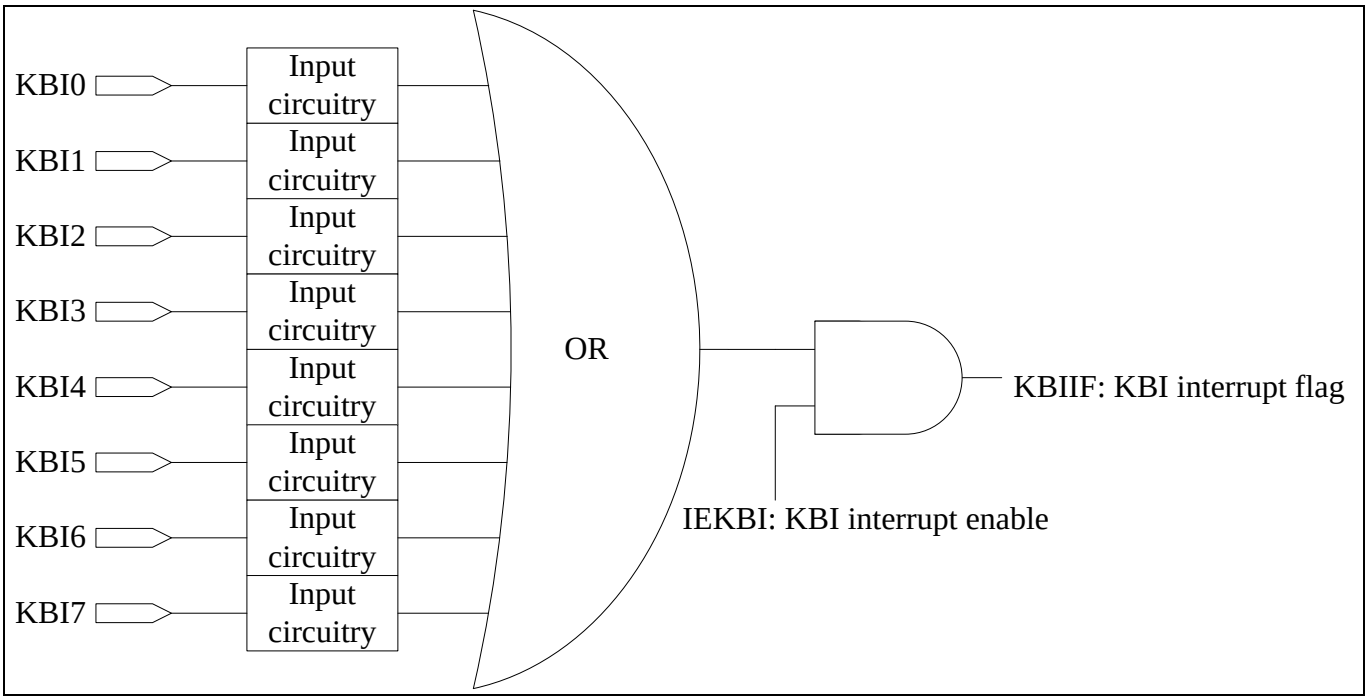


Figure 16.1 keyboard interface block diagram

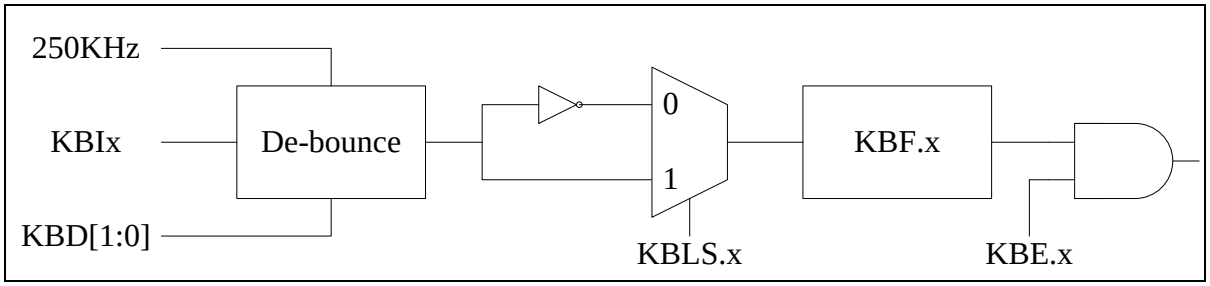


Figure 16.2 keyboard input circuitry

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
KBI function											
AUX	Auxiliary register	91h	BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H
KBLS	KBI level selection	93h	KBLS7	KBLS6	KBLS5	KBLS4	KBLS3	KBLS2	KBLS1	KBLS0	00H
KBE	KBI input enable	94h	KBE7	KBE6	KBE5	KBE4	KBE3	KBE2	KBE1	KBE0	00H
KBF	KBI flag	95h	KBF7	KBF6	KBF5	KBF4	KBF3	KBF2	KBF1	KBF0	00H
KBD	KBI De-bounce control register	96h	KBDEN	-	-	-	-	-	KBD1	KBD0	00H

Mnemonic: AUX **Address: 91h**

7	6	5	4	3	2	1	0	Reset
BRGS	P4CC	P4SPI	P4UR1	P4IIC	P0KBI	-	DPS	00H

P0KBI: P0KBI = 0 – KBI function on P2.
P0KBI = 1 – KBI function on P0.

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: Keyboard Line 7 level selection bit
0 : enable a low level detection on KBI7.
1 : enable a high level detection on KBI7.
KBLS.6: Keyboard Line 6 level selection bit
0 : enable a low level detection on KBI6.
1 : enable a high level detection on KBI6.
KBLS.5: Keyboard Line 5 level selection bit
0 : enable a low level detection on KBI5.
1 : enable a high level detection on KBI5.
KBLS.4: Keyboard Line 4 level selection bit
0 : enable a low level detection on KBI4.
1 : enable a high level detection on KBI4.
KBLS.3: Keyboard Line 3 level selection bit
0 : enable a low level detection on KBI3.
1 : enable a high level detection on KBI3.
KBLS.2: Keyboard Line 2 level selection bit
0 : enable a low level detection on KBI2.
1 : enable a high level detection on KBI2.
KBLS.1: Keyboard Line 1 level selection bit
0 : enable a low level detection on KBI1.
1 : enable a high level detection on KBI1.
KBLS.0: Keyboard Line 0 level selection bit
0 : enable a low level detection on KBI0.
1 : enable a high level detection on KBI0.

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: Keyboard 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: Keyboard 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: Keyboard 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: Keyboard 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: Keyboard 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: Keyboard 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: Keyboard 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: Keyboard 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: Keyboard Line 7 flag
This is set by hardware when KBI7 detects a programmed level.
It generates a Keyboard interrupt request if KBE.7 is also set. It must be cleared by software.

KBF.6: Keyboard Line 6 flag
This is set by hardware when KBI6 detects a programmed level.
It generates a Keyboard interrupt request if KBE.6 is also set. It must be cleared by software.

KBF.5: Keyboard Line 5 flag
This is set by hardware when KBI5 detects a programmed level.
It generates a Keyboard interrupt request if KBE.5 is also set. It must be cleared by software.

KBF.4: Keyboard Line 4 flag
This is set by hardware when KBI4 detects a programmed level.
It generates a Keyboard interrupt request if KBE.4 is also set. It must be cleared by software.

KBF.3: Keyboard Line 3 flag
This is set by hardware when KBI3 detects a programmed level.
It generates a Keyboard interrupt request if KBE.3 is also set. It must be cleared by software.

KBF.2: Keyboard Line 2 flag
This is set by hardware when KBI2 detects a programmed level.
It generates a Keyboard interrupt request if KBE.2 is also set. It must be cleared by software.

KBF.1: Keyboard Line 1 flag
This is set by hardware when KBI1 detects a programmed level.
It generates a Keyboard interrupt request if KBE.1 is also set. It must be cleared by software.

KBF.0: Keyboard Line 0 flag
This is set by hardware when KBI0 detects a programmed level.
It generates a Keyboard interrupt request if KBE.0 is also set. It must be cleared by software.

Mnemonic: KBD **Address: 96H**

7	6	5	4	3	2	1	0	Reset
KBDEN	-	-	-	-	-	KBD.1	KBD.0	00H

KBDEN: Enable KBI de-bounce function. The default KBI function is enabled.
KBDEN = 0, enable KBI de-bounce function. The de-bounce time is selected by KBD [1:0].
KBDEN=1, disable KBI de-bounce function. The KBI input pin without de-bounce mechanism.

KBD[1:0]: Select KBI de-bounce time. If KBDEN = “0”, the default de-bounce time is 320 ms.
KBD[1:0] = 00, the de-bounce time is 320 ms.
KBD[1:0] = 01, the de-bounce time is 160 ms.
KBD[1:0] = 10, the de-bounce time is 80 ms.
KBD[1:0] = 11, the de-bounce time is 40 ms.

17. LVI – Low Voltage Interrupt

The interrupt vector 63h.

Mnemonic: LVC **Address: E6h**

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

LVI_EN: Low voltage interrupt function enable bit.
LVI_EN = 0 : disable low voltage detect function.
LVI_EN = 1 : enable low voltage detect function.
LVRXE: External low voltage reset function enable bit.
LVRXE = 0 : disable external low voltage reset function.
LVRXE = 1 : enable external low voltage reset function.

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

The SM59R04A2 provides eight channels 10-bit ADC. The Digital output DATA [9:0] were put into ADCD [9:0]. The ADC interrupt vector is 53H.

The ADC SFR show as below:

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
ADC											
ADCC1	ADC Control register 1	ABh	ADC7EN	ADC6EN	ADC5EN	ADC4EN	ADC3EN	ADC2EN	ADC1EN	ADC0EN	00H
ADCC2	ADC Control register 2	ACh	Start	ADJUST	-	-	-	ADCCH[2:0]			00H
ADCDH	ADC data high byte	ADh	ADCDH [7:0]								00H
ADCDL	ADC data low byte	AEh	ADCDL [7:0]								00H
ADCCS	ADC clock select	AFh	-	-	-	ADCCS[4:0]					00H

Mnemonic: ADCC1 **Address: ABh**

7	6	5	4	3	2	1	0	Reset
ADC7EN	ADC6EN	ADC5EN	ADC4EN	ADC3EN	ADC2EN	ADC1EN	ADC0EN	00H

ADC7EN: ADC channels 7 enable.
ADC7EN = 1 – Enable ADC channel 7
ADC6EN: ADC channels 6 enable.
ADC6EN = 1 – Enable ADC channel 6
ADC5EN: ADC channels 5 enable.
ADC5EN = 1 – Enable ADC channel 5
ADC4EN: ADC channels 4 enable.
ADC4EN = 1 – Enable ADC channel 4
ADC3EN: ADC channels 3 enable.
ADC3EN = 1 – Enable ADC channel 3
ADC2EN: ADC channels 2 enable.
ADC2EN = 1 – Enable ADC channel 2
ADC1EN: ADC channels 1 enable.
ADC1EN = 1 – Enable ADC channel 1
ADC0EN: ADC channels 0 enable.
ADC0EN = 1 – Enable ADC channel 0

Mnemonic: ADCC2 **Address: ACh**

7	6	5	4	3	2	1	0	Reset
Start	ADJUST	-	-	-	ADCCH[2:0]			00H

Start: When this bit is set, the ADC will be start conversion.
ADJUST: Adjust the format of ADC conversion DATA.
ADJUST = 0: (default value)
ADC data high byte ADCD [9:2] = ADCDH [7:0].
ADC data low byte ADCD [1:0] = ADCDL [1:0].
ADJUST = 1:
ADC data high byte ADCD [9:8] = ADCDH [1:0].
ADC data low byte ADCD [7:0] = ADCDL [7:0].
ADCCH[2:0]: ADC channel select.

ADCCH [2:0]	Channel
000	0
001	1

010	2
011	3
100	4
101	5
110	6
111	7

ADJUST = 0:

Mnemonic: ADCDH **Address: ADh**

7	6	5	4	3	2	1	0	Reset
ADCD[9]	ADCD[8]	ADCD[7]	ADCD[6]	ADCD[5]	ADCD[4]	ADCD[3]	ADCD[2]	00H

Mnemonic: ADCDL **Address: AEh**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	ADCD[1]	ADCD[0]	00H

ADJUST = 1:

Mnemonic: ADCDH **Address: ADh**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	ADCD[9]	ADCD[8]	00H

Mnemonic: ADCDL **Address: AEh**

7	6	5	4	3	2	1	0	Reset
ADCD[7]	ADCD[6]	ADCD[5]	ADCD[4]	ADCD[3]	ADCD[2]	ADCD[1]	ADCD[0]	00H

ADCD[9:0]: ADC data register.

Mnemonic: ADCCS **Address: AFh**

7	6	5	4	3	2	1	0	Reset
-	-	-	ADCCS[4]	ADCCS[3]	ADCCS[2]	ADCCS[1]	ADCCS[0]	00H

ADCCS[4:0]: ADC clock select.
*The ADC clock maximum 12.5MHz.
*The ADC Conversion rate maximum 500KHz.

ADCCS[4:0]	ADC Clock(Hz)	Clocks for ADC Conversion
00000	Fclk/2	46
00001	Fclk/4	92
00010	Fclk/6	138
00011	Fclk/8	184
00100	Fclk/10	230
00101	Fclk/12	276
00110	Fclk/14	322
00111	Fclk/16	368
01000	Fclk/18	414
01001	Fclk/20	460
01010	Fclk/22	506
01011	Fclk/24	552
01100	Fclk/26	598
01101	Fclk/28	644
01110	Fclk/30	690
01111	Fclk/32	736
10000	Fclk/34	782
10001	Fclk/36	828
10010	Fclk/38	874

10011	Fclk/40	920
10100	Fclk/42	966
10101	Fclk/44	1012
10110	Fclk/46	1058
10111	Fclk/48	1104
11000	Fclk/50	1150
11001	Fclk/52	1196
11010	Fclk/54	1242
11011	Fclk/56	1288
11100	Fclk/58	1334
11101	Fclk/60	1380
11110	Fclk/62	1426
11111	Fclk/64	1472

$$ADC\_Clock = \frac{\text{Fclk}}{2 \times (ADCCS + 1)}$$

$$ADC\_Conversion\_Rate = \frac{\text{ADC\_Clock}}{23}$$

19. In-System Programming (Internal ISP)

The SM59R04A2 can generate flash control signal by internal hardware circuit. Users utilize flash control register, flash address register and flash data register to perform the ISP function without removing the SM59R04A2 from the system. The SM59R04A2 provides internal flash control signals which can do flash program/chip erase/page erase/protect functions. User need to design and use any kind of interface which SM59R04A2 can input data. User then utilize ISP service program to perform the flash program/chip erase/page erase/protect functions.

19.1. ISP service program

The ISP service program is a user developed firmware program which resides in the ISP service program space. After user developed the ISP service program, user then determine the size of the ISP service program. User need to program the ISP service program in the SM59R04A2 for the ISP purpose.

The ISP service programs were developed by user so that it should includes any features which relates to the flash memory programming function as well as communication protocol between SM59R04A2 and host device which output data to the SM59R04A2. For example, if user utilize UART interface to receive/transmit data between SM59R04A2 and host device, the ISP service program should include baud rate, checksum or parity check or any error-checking mechanism to avoid data transmission error.

The ISP service program can be initiated under SM59R04A2 active or idle mode. It can not be initiated under power down mode.

19.2. Lock Bit (N)

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

The ISP service program space address range $3000 to $3FFF. It can be divided as blocks of N*256 byte. (N=0 to 16). When N=0 means no ISP function, all of 16K byte flash memory can be used as program memory. When N=1 means ISP service program occupies 256 byte while the rest of 15.75K byte flash memory can be used as program memory. The maximum ISP service program allowed is 4K byte when N=16. Under such configuration, the usable program memory space is 12K byte.

After N determined, SM59R04A2 will reserve the ISP service program space downward from the top of the program address $3FFF. The start address of the ISP service program located at $3x00 while x is an even number, depending on the lock bit N. As shown in Table 19-1.

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

Table 19.1 ISP code area.

N	ISP service program address
0	No ISP service program
1	256 bytes ($3F00h ~ $3FFFh)
2	512 bytes ($3E00h ~ $3FFFh)
3	768 bytes ($3D00h ~ $3FFFh)
4	1.0 K bytes ($3C00h ~ $3FFFh)
5	1.25 K bytes ($3B00h ~ $3FFFh)
6	1.5 K bytes ($3A00h ~ $3FFFh)
7	1.75 K bytes ($3900h ~ $3FFFh)
8	2.0 K bytes ($3800h ~ $3FFFh)
9	2.25 K bytes ($3700h ~ $3FFFh)
10	2.5 K bytes ($3600h ~ $3FFFh)
11	2.75 K bytes ($3500h ~ $3FFFh)

12	3.0 K bytes ($3400h ~ $3FFFh)
13	3.25 K bytes ($3300h ~ $3FFFh)
14	3.5 K bytes ($3200h ~ $3FFFh)
15	3.75 K bytes ($3100h ~ $3FFFh)
16	4.0 K bytes ($3000h ~ $3FFFh)

ISP service program configurable in N*256 byte (N= 0 ~ 16)

19.3. Program the ISP Service Program

After Lock Bit N is set and ISP service program been programmed, the ISP service program memory will be protected (locked) automatically. The lock bit N has its own program/erase timing. It is different from the flash memory program/erase timing so the locked ISP service program can not be erased by flash erase function. If user needs to erase the locked ISP service program, he can do it by writer only. User can not change ISP service program when SM59R04A2 was in system.

19.4. Initiate ISP Service Program

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

(1) Blank reset. Hardware reset with first flash address blank ($0000=#FFH) will load the PC with start address of ISP service program. The hardware reset includes Internal (power on reset) and external pad reset.

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

(3) Enters ISP service program by hardware setting. User can force SM59R04A2 enter ISP service program by setting P2.6, P2.7 “active low” or P4.3 “ active low” during hardware reset period. The hardware reset includes Internal (power on reset) and external pad reset. In application system design, user should take care of the setting of P2.6, P2.7 or P4.3 at reset period to prevent SM59R04A2 from entering ISP service program.

(4) Enter’s ISP service program by hardware setting, the port3.0 will be detected the two clock signals during hardware reset period. The hardware reset includes Internal (power on reset) and external pad reset. And detect 2 clock signals after hardware reset.

During hardware reset period, the hardware will detect the status of P2.6/P2.7/P4.3/P3.0. If they meet one of above conditions, chip will switch to ISP mode automatically. After ISP service program executed, user need to reset the SM59R04A2, either by hardware reset or by WDT, or jump to the address $0000 to re-start the firmware program.

There are 8 kinds of entry mechanisms for user different applications. This entry method will select on the writer or ISP.

(1) First Address Blank. i.e. $0000 = 0xFF. And triggered by Internal reset signal.
(2) First Address Blank. i.e. $0000 = 0xFF. And triggered by PAD reset signal.
(3) P2.6 = 0 & P2.7 = 0. And triggered by Internal reset signal.
(4) P2.6 = 0 & P2.7 = 0. And triggered by PAD reset signal.
(5) P4.3 = 0. And triggered by Internal reset signal.
(6) P4.3 = 0. And triggered by PAD reset signal.
(7) P3.0 input 2 clocks. And triggered by Internal reset signal.
(8) P3.0 input 2 clocks. And triggered by PAD reset signal.

19.5. ISP register – TAKEY, IFCON, ISPFAH, ISPFAL, ISPFD and ISPFC

Mnemonic	Description	Direct	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	RESET
ISP function											
TAKEY	Time Access Key register	F7h	TAKEY [7:0]								00H
IFCON	Interface Control register	8Fh	ITS	CDPR	-	-	ALEC.1	ALEC.0	EMEN	ISPE	00H
ISPFAH	ISP Flash Address - High register	E1h			ISPFAH [5:0]						FFH
ISPFAL	ISP Flash Address - Low register	E2h	ISPFAL [7:0]								FFH
ISPFD	ISP Flash Data register	E3h	ISPFD [7:0]								FFH
ISPFC	ISP Flash Control register	E4h	EMF1	EMF2	EMF3	EMF4	-	ISPF.2	ISPF.1	ISPF.0	00H

Mnemonic: TAKEY **Address: F7H**

7	6	5	4	3	2	1	0	Reset
TAKEY [7:0]								00H

ISP enable bit (ISPE) is read-only by default, software must write three specific values 55h, AAh and 5Ah sequentially to the TAKEY register to enable the ISPE bit write attribute. That is:

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah
```

Mnemonic: IFCON **Address: 8FH**

7	6	5	4	3	2	1	0	Reset
ITS	CDPR	-	-	ALEC[1]	ALEC[0]	EMEN	ISPE	00H

The bit 0 (ISPE) of IFCON is ISP enable bit. User can enable overall SM59R04A2 ISP function by setting ISPE bit to 1, to disable overall ISP function by set ISPE to 0. The function of ISPE behaves like a security key. User can disable overall ISP function to prevent software program be erased accidentally. ISP registers ISPFAH, ISPFAL, ISPFD and ISPFC are read-only by default. Software must be set ISPE bit to 1 to enable these 4 registers write attribute.

Mnemonic: ISPFAH **Address: E1H**

7	6	5	4	3	2	1	0	Reset
		ISPFAH5	ISPFAH4	ISPFAH3	ISPFAH2	ISPFAH1	ISPFAH0	FFH

ISPFAH [5:0]: Flash address-high for ISP function

Mnemonic: ISPFAL **Address: E2H**

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

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

The ISPFAH & ISPFAL provide the 14-bit flash memory address for ISP function. The flash memory address should not include the ISP service program space address. If the flash memory address indicated by ISPFAH & ISPFAL registers overlay with the ISP service program space address, the flash program/page erase of ISP

function executed thereafter will have no effect.

Mnemonic: ISPFD **Address: E3H**

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

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

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

Mnemonic: ISPFC **Address: E4H**

7	6	5	4	3	2	1	0	Reset
EMF1	EMF2	EMF3	EMF4	-	ISPF[2]	ISPF[1]	ISPF[0]	00H

EMF1: Entry mechanism (1) flag, clear by reset. (Read only)
EMF2: Entry mechanism (2) flag, clear by reset. (Read only)
EMF3: Entry mechanism (3) flag, clear by reset. (Read only)
EMF4: Entry mechanism (4) flag, clear by reset. (Read only)
ISPF [2:0]: ISP function select bit.

ISPF[2:0]	ISP function
000	Byte program
001	Chip protect
010	Page erase
011	Chip erase
100	Write option
101	Read option
110	Erase option
111	-

One page of flash memory is 256 byte

The Option function can access the Internal reset time select(description in section 1.4.1)、clock source select(description in section 1.5)、P4[4:7] pins function select(description in section 5)、WDTEN control bit(description in section 10)、or ISP entry mechanisms select(description in section 19)。
When chip protected or no ISP service, option can only read.

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

To perform byte program/page erases ISP function, user need to specify flash address at first. When performing page erase function, SM59R04A2 will erase entire page which flash address indicated by ISPFAH & ISPFAL registers located within the page.

e.g. flash address: $XYMN

page erase function will erase from $XY00 to $XYFF

To perform the chip erase ISP function, SM59R04A2 will erase all the flash program memory except the ISP service program space. To perform chip protect ISP function, the SM59R04A2 flash memory content will be read #00H.

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

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah          ; enable ISPE write attribute
MOV IFCON, #01H          ; enable SM59R04A2 ISP function
```

```
MOV ISPFAH, #10H        ; set flash address-high, 10H
MOV ISPFAL, #05H        ; set flash address-low, 05H
MOV ISPFD, #22H         ; set flash data to be programmed, data = 22H
MOV ISPFC, #00H         ; start to program #22H to the flash address $1005H
```

Operating Conditions

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

DC Characteristics

T_A = -40℃ to 85℃, V_{CC} = 5.0V

Symbol	Parameter	Valid	Min	Max	Units	Conditions
VIL1	Input Low-voltage	Port 0,1,2,3,4,5	-0.5	0.8	V	Vcc=5V
VIL2	Input Low-voltage	RES, XTAL1	0	0.8	V	
VIH1	Input High-voltage	Port 0,1,2,3,4,5	2.0	V_{CC} + 0.5	V	
VIH2	Input High-voltage	RES, XTAL1	70%Vcc	V_{CC} + 0.5	V	
VOL	Output Low-voltage	Port 0,1,2,3,4,5		0.4	V	IOL=4.9mA Vcc=5V
VOH1	Output High-voltage using Strong Pull-up(1)	Port 0,1,2,3,4,5	90% V_{CC}		V	IOH= -4.6mA
VOH2	Output High-voltage using Weak Pull-up(2)	Port 0,1,2,3,4,5	2.4		V	IOH= -250uA
			75% V_{CC}		V	IOH= -162uA
			90% V_{CC}		V	IOH= -73uA
IIL	Logic 0 Input Current	Port 0,1,2,3,4,5		-75	uA	Vin= 0.45V
ITL	Logical Transition Current	Port 0,1,2,3,4,5		-650	uA	Vin= 2.0V
ILI	Input Leakage Current	Port 0,1,2,3,4,5		±10	uA	0.45V<Vin<Vcc
RRST	Reset Pull-down Resistor	RES	50	300	kΩ	
CIO	Pin Capacitance			10	pF	Freq= 1MHz, Ta= 25℃
ICC	Power Supply Current	VDD		12	mA	Active mode, 12MHz
				11	mA	Idle mode, 12MHz
				30	uA	Power down mode

Notes： 1. Port in Push-Pull Output Mode

2. Port in Quasi-Bidirectional Mode

Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Units**
V_{LVI}	Low Voltage Interrupt Voltage Level	3.0	3.5	4.1	V
V_{LVR}	Low Voltage Reset Voltage Level	2.6	3.1	3.5	V

Notes: The V_{LVI} always above V_{LVR} about 0.4V.

T_A = -40℃ to 85℃, V_{CC} = 3.3V

Symbol	Parameter	Valid	Min	Max	Units	Conditions
VIL1	Input Low-voltage	Port 0,1,2,3,4,5	-0.5	0.8	V	Vcc=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	V_{CC} + 0.5	V	
VIH2	Input High-voltage	RES, XTAL1	70%Vcc	V_{CC} + 0.5	V	
VOL	Output Low-voltage	Port 0,1,2,3,4,5		0.4	V	IOL=3.2mA Vcc=3.3V
VOH1	Output High-voltage using Strong Pull-up(1)	Port 0,1,2,3,4,5	90% V_{CC}		V	IOH= -2.3mA
VOH2	Output High-voltage using Weak Pull-up(2)	Port 0,1,2,3,4,5	2.4		V	IOH= -77uA
			90% V_{CC}		V	IOH= -33uA
IIL	Logic 0 Input Current	Port 0,1,2,3,4,5		-75	uA	Vin= 0.45V
ITL	Logical Transition Current	Port 0,1,2,3,4,5		-650	uA	Vin=1.5V
ILI	Input Leakage Current	Port 0,1,2,3,4,5		±10	uA	0.45V<Vin<Vcc
RRST	Reset Pull-down Resistor	RES	50	300	kΩ	
CIO	Pin Capacitance			10	pF	Freq= 1MHz, Ta= 25℃
ICC	Power Supply Current	VDD		11	mA	Active mode, 12MHz
				10	mA	Idle mode, 12MHz
				20	uA	Power down mode

Notes： 1. Port in Push-Pull Output Mode

2. Port in Quasi-Bidirectional Mode

Symbol	Parameter	Min	Typ	Max	Units
V_{LVI}	Low Voltage Interrupt Voltage Level	1.9	2.3	2.6	V
V_{LVR}	Low Voltage Reset Voltage Level	1.6	2.1	2.3	V

Notes: The V_{LVI} always above V_{LVR} about 0.2V.