IS32CS8974

MCU with 2KB ECC SRAM/32KB ECC E-Flash for Touch Key Applications

GENERAL DESCRIPTION

CS8974 is a general-purpose MCU with 32KB code memory (organized as 32Kx16) of embedded-flash memory and 2KB (organized as 2Kx13) SRAM for data manipulations. Both SRAM and e-Flash implement built-in ECC that corrects 1-bit error and detects two-bit errors. CPU can access the e-Flash through program address read and through Flash Controller which can perform software read/write operations of e-Flash for EEPROM emulations.

CPU in CS8974 is 1-T 8051 with enhanced multiplication and division accelerator. There are two clock sources for the system, one is a 16MHz IOSC (manufacturer calibrated +/- 2%) and the other one is SOSC32KHz (typical 32KHz) which is divided by a slow oscillator (SOSC) 128KHz. Both clock sources have a clock programmable divider for scaling down the frequency to save power dissipations. The clock selections are combined with flexible power management schemes, including NORMAL, IDLE, STOP, and SLEEP modes to balance speed and power consumption.

There are T0/T1/T2/T3/T4/T5 timers coupled with CPU and two WDT where WDT0 is clocked by SYSCLK, and WDT2/WDT3 are clocked by a non-stop SOSC32KHz. An 8-bit/16-bit checksum and 16-bit CRC accelerator is included. There are an EUART/LIN controller, I2C master and slave controllers, and a SPI master/slave controller. The interfaces of these controllers are multiplexed with GPIO pins. Other useful peripherals include a buzzer/melody control, 6 channels of 8-bit PWM, and one channel of timer/capture and quadrature decoder.

Analog peripherals include touch key controllers with up to 20-bit resolution employing dual-slope charge-sharing capacitance conversion. The touch key controller also has shield output capability for moisture immunity. The touch key controller allows sleep mode (5uA) and auto-detection for wakeup. The maximum number of key scans is 19. IS32CS8974 can support proximity sensing.

CS8974 also provides a flexible means of flash programming that supports ISP and IAP. The protection of data loss is implemented in hardware by access restriction of critical storage segments. The code security is reinforced with sophisticated writer commands and ISP commands. The on-chip breakpoint processor also allows easy debugging which can be integrated with ISP. A reliable power-on-reset circuit and low supply voltage detection allow reliable operations under harsh environments.

Applications

- Touch key applications with high robustness and reliability requirements
- Automotive and appliance

FEATURES

CPU and Memory

- Up to 25MHz 1-Cycle 8051 CPU core (16MHz zero wait state)
- 16-bit Timers T0/T1/T2/T3/T4 and 24-bit Timer T5
- Checksum and CRC accelerator
- WDT1 by SYSCLK, WDT2/WDT3 by SOSC32KHz
- Clock fault monitor
- Integrated breakpoint controller and debug port through I^2C slave
- All GPIO pins can be assigned to two external interrupts
- Power saving modes – IDLE, STOP, and SLEEP
- 256B IRAM and 1792B XRAM with ECC
- 32Kx16 Flash Memory and two 512x16 Information Block
  - Program read with hardware ECC
  - Software read/write direct access
  - Code security and data loss protection
  - 100K Endurance and 10 years Retention

Clock Sources

- Internal oscillator at 16MHz of +/- 2% accuracy
  - Spread Spectrum option
- Internal low-power slow oscillator 128KHz
- External clock option

Digital Peripherals

- 6 CH 8-bit center-aligned PWM controller with trigger interrupt and polarity control
- Timer/Capture and quadrature decoder
- Buzzer and melody waveform generator
- One I^2C Master, two I^2C Slave
  - I2CS1 allows address match wakeup and two address
  - I2CS2 for ISP and debug
- One SPI Master/Slave Controller
- One 8051 UART and One full-duplex LIN-capable EUART2

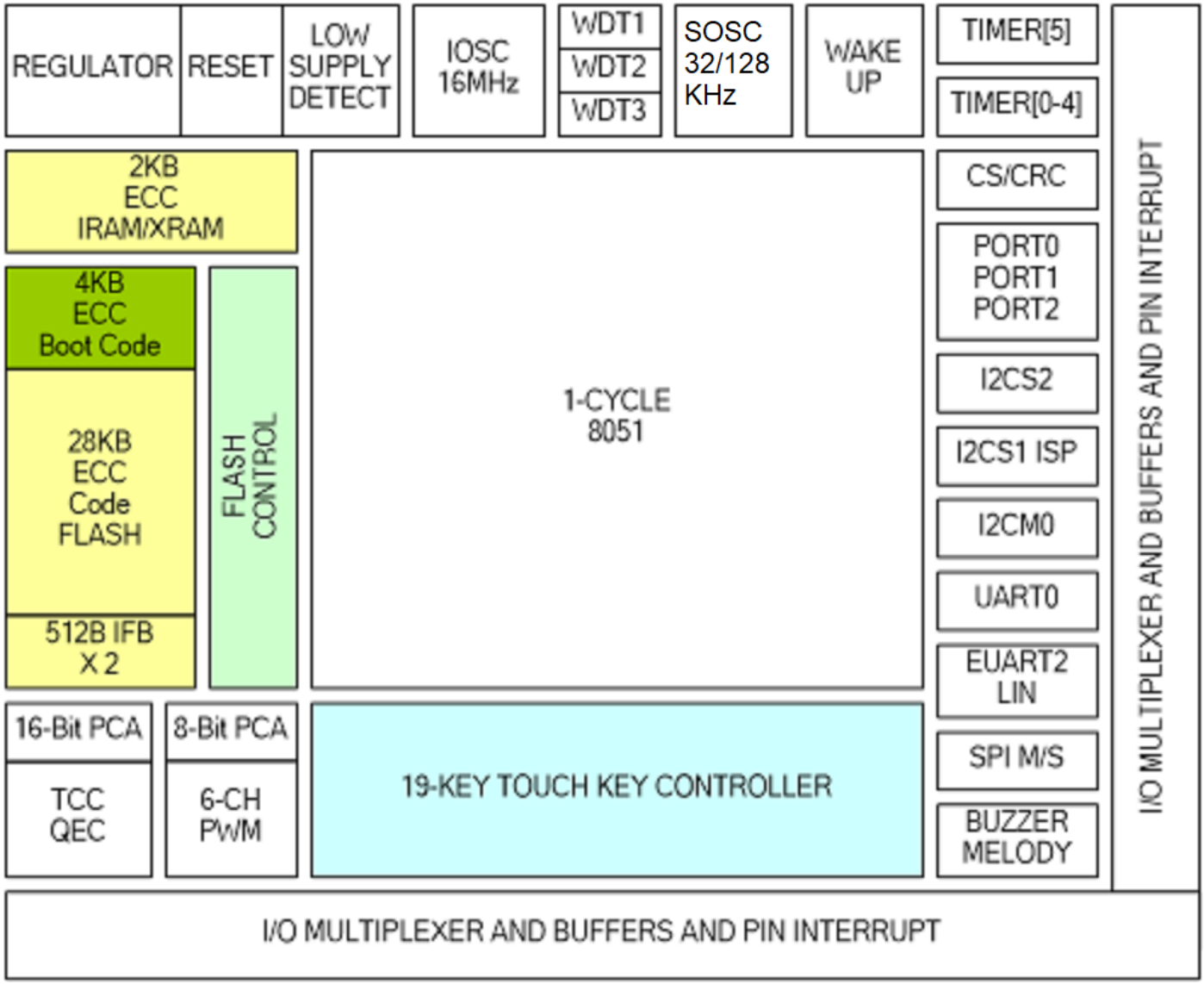
Analog Peripherals

- Capacitance sense touch-key controller
  - Dual slope charge transfer for higher PSRR and CMRR with up to 20-bit resolutions
  - Up to 19 key inputs with low power wakeup (5uA)
  - Shield output for moisture immunity
- Power-on reset and Low voltage detection (2.0V-4.5V)

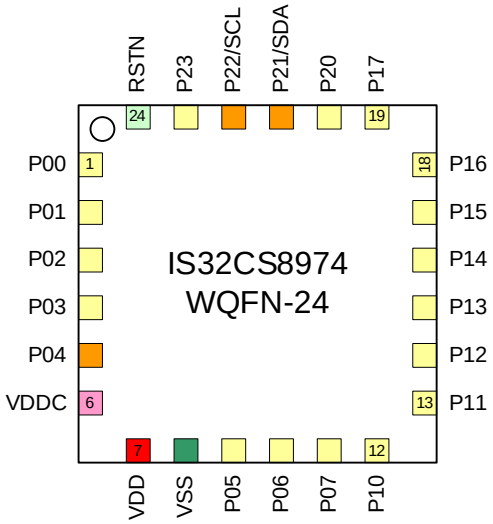
Miscellaneous

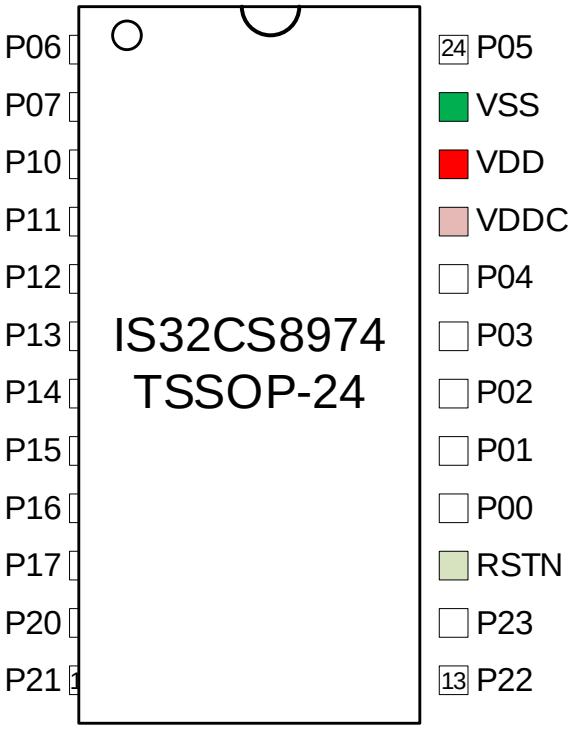
- Up to 20 GPIO pins
  - Noise filters and Dual edge interrupt/wakeup
- 2.5V to 5.5V single supply
- Low power standby (1uA) in SLEEP mode
- Operating temperature -40°C to 125°C
- TSSOP-24 and wettable flank QFN-24 package
- RoHS & Halogen-Free compliant package
- TSCA compliance
- AEC-Q100 qualification

BLOCK DIAGRAM



PINOUT


RSTN
P23
P22/SCL
P21/SDA
P20
P17
24
19
P00
1
P01
P02
P03
P04
VDDC
6
18
P16
P15
P14
P13
P12
13
P11
IS32CS8974
WQFN-24
7
12
VDD
VSS
P05
P06
P07
P10


P06
P07
P10
P11
P12
P13
P14
P15
P16
P17
P20
P21
IS32CS8974
TSSOP-24
24 P05
VSS
VDD
VDDC
P04
P03
P02
P01
P00
RSTN
P23
13 P22

PIN Multifunction Table

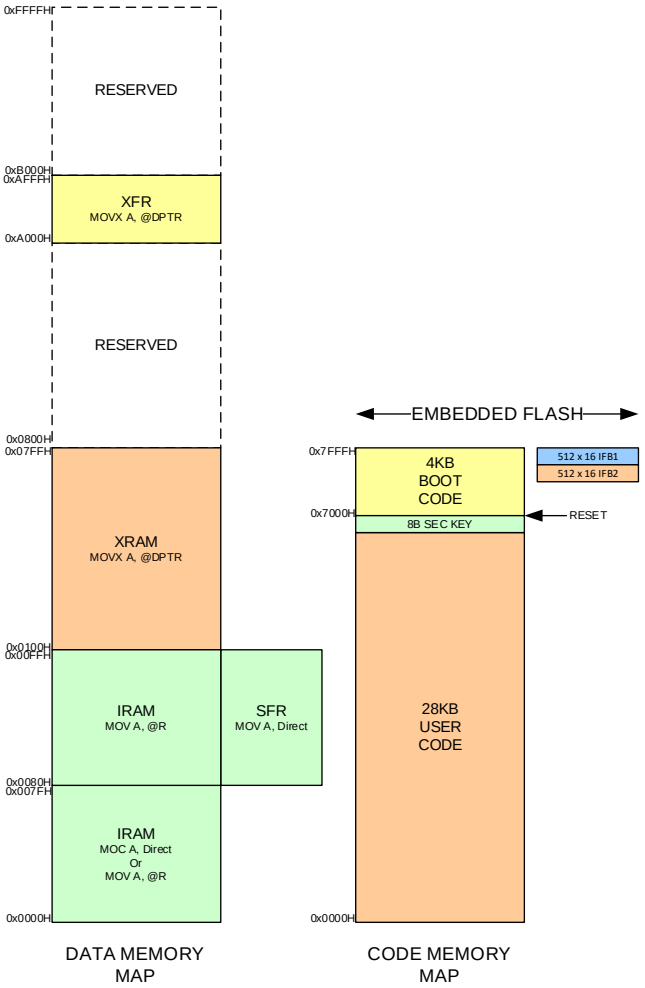
PIN# Q/S*	MFCFG 0	MFCFG 1	MFCFG 2	MFCFG 3	MFCFG 4	MFCFG 5	MFCFG 6	MFCFG 7	ANIO 1	ANIO 2
1/16	P00	PHA	XCAPT	SSN	BZ	TX0	PWM0	-	KEY	SHIELD
2/17	P01	PHB	CC	MOSI	T0	RX0	PWM1	-	KEY	SHIELD
3/18	P02	INDEX	XCAPT	MISO	SSDA1	TX2	PWM2	-	KEY	SHIELD
4/19	P03	XCAPT	TC	SCLK	SSCL1	RX2	PWM3	-	KEY	SHIELD
5/20	P04	PHA	TC	CC	BZ	TKC2	PWM4	XCLKIN	CREF	CREF
6/21	**VDDC**	Core supply 1.50V at normal mode, 1.40V at sleep mode. Connect 1uF and 0.1uF to VSS for decoupling.								
7/22	**VDD**	Power supply 2.2V to 5.5V								
8/23	**VSS**	Ground supply 0V.								
9/24	P05	PHB	XCAPT	MISO	T0	TX2	PWM5	-	KEY	SHIELD
10/1	P06	INDEX	CC	MOSI	T1	RX2	PWM0	-	KEY	SHIELD
11/2	P07	XCAPT	TC	SCLK	T2	TX2	PWM1	-	KEY	SHIELD
12/3	P10	PHA	CC	SSN	T0	RX2	PWM2	-	KEY	SHIELD
13/4	P11	PHB	TC	CC	T1	BZ	PWM3	XCLKIN	KEY	SHIELD
14/5	P12	INDEX	XCAPT	SSCL2	MSCL	SSCL1	PWM4	-	KEY	SHIELD
15/6	P13	XCAPT	CC	SSDA2	MSDA	SSDA1	PWM5	-	KEY	SHIELD
16/7	P14	PHA	TC	SSN	CC	TX2	PWM0	-	KEY	SHIELD
17/8	P15	PHB	XCAPT	SSN	T2	BZ	PWM1	-	KEY	SHIELD
18/9	P16	INDEX	TC	MISO	CC	RX2	PWM2	-	KEY	SHIELD
19/10	P17	XCAPT	TC	MOSI	CC	TX2	PWM3	-	KEY	SHIELD
20/11	P20	PHA	XCAPT	SCLK	BZ	RX2	PWM4	-	KEY	SHIELD
21/12	P21	PHB	CC	SSDA2	MSDA	SSDA1	PWM5	-	KEY	SHIELD
22/13	P22	INDEX	TC	SSCL2	MSCL	SSCL1	PWM0	-	KEY	SHIELD
23/14	P23	XCAPT	CC	SSN	RX2	TX2	PWM1	-	KEY	SHIELD
24/15	**RSTN**	External reset input, low active. Internal 6K Ohm pull-up.								

1. More than one function can be enabled. The outputs are OR-ed.
2. Input for GPIO port, interrupt/wakeup is always enabled. For other functions, the inputs are multiplexed to the specific function blocks.
3. Pin 21 (P21) as SDA and Pin 22 (P22) as SCL are used for In-System-Programming (ISP).
4. Pin 19 (P17) as CEB, Pin 20 (P20) as SCK, Pin 21 (P21) as SDI, Pin 22 (P22) as SDO, along with Pin 24 (RSTN) are used in Writer Mode. Pin 23 (P23) for Flash TBIT ready output is optional for Writer Mode. RSTN is also necessary for Writer Mode.
5. Pin number is shown in WQFN24/TSSOP24.
6. **If customers would like to use our CS89XX Touch Key Library software tool, please refer to our IS3XCS89XX Touch Key Library Tool User's Manual before starting your hardware schematics design.**

MEMORY MAP

There is a total of 256 bytes of internal RAM in CS8974, the same as standard 8052. And there is a total of 1792 bytes of auxiliary RAM allocated in the 8051 extended RAM area at 0x0100h – 0x07FFh. Programs can use "MOVX" instructions to access the XRAM.

There is a 32Kx16 (64KB) embedded Flash memory for code storage. For CPU program access (Read-only), the lower byte is used for actual access, and the upper byte is used for ECC check. The ECC is performed in nibble bases with each nibble in the high byte corresponding to the nibbles in the low byte. ECC in this case is capable of one-bit correction and two-bit detection for each nibble. This is significantly more robust than 8:5 ECC. ECC check is through hardware and is performed automatically. The embedded Flash can also be accessed through the Flash controller. For erase operations, the page size of the Flash is 512x16. There are two 512x16 IFB blocks in Flash. The first IFB is used for manufacturing and calibration data, and some areas are for user OTP data. The 2nd IFB is open for user applications with no restriction. Also, there is an 8-byte code security key located at the last 8 bytes of user program space for protection from pirate access to information.



REGISTER MAP SFR (0x80 – 0xFF)

The SFR address map maintains maximum compatibilities to the most commonly existing 8051-like MCU. The following table shows the SFR address map. Since SFR can be accessed by direct addressing mode, registers of built-in peripherals that require fast access are mostly located in SFR. XFR is mainly used for on-chip peripheral control and configurations.

	0	1	2	3	4	5	6	7
0XF0	B	-	CLSR	CHSR	I2CMSA	I2CMCR	I2CMBUF	I2CMTP
0XE0	ACC	-	-	-	-	-	-	-
0XD0	PSW	-	-	-	-	-	-	-
0XC0	-	-	SCON2	I2CMTO	PMR	STATUS	MCON	TA
0XB0	-	-	-	-	-	-	-	-
0XA0	P2	SPICR	SPIMR	SPIST	SPIDATA	SFIFO2	SBUF2	SINT2
0X90	P1	EXIF	WTST	DPX	-	DPX1	-	-
0X80	P0	SP	DPL	DPH	DPL1	DPH1	DPS	PCON
	8	9	A	B	C	D	E	F
0XF8	EXIP	MD0	MD1	MD2	MD3	MD4	MD5	ARCON
0XE8	EXIE	CH	MXAX	I2CSCON1A	I2CSST1	I2CSADR1	I2CSDAT1	-
0XD8	WDCON	CL	DPXR	I2CSCON2	I2CSST2	I2CSADR2	I2CSDAT2	-
0XC8	T2CON	TB	RLDL	RLDH	TL2	TH2	-	T34CON
0XB8	IP	-	-	-	-	-	-	-
0XA8	IE	-	-	I2CSCON1B	TL4	TH4	TL3	TH3
0X98	SCON0	SBUF0	-	ESP	-	ACON	I2CSADR3	WKMASK
0X88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	CKSEL

REGISTER MAP XFR (0xA000 – 0xAFFF)

	0	1	2	3	4	5	6	7
A000	REGTRM	IOSCITRM	IOSCVTRM	-	-	-	-	SOSCTRM
A010	LVDCFG	LVDTHD	LVDHYS	-	TSTMON	-	BSTCMD	RSTCMD
A020	FLSHDATL	FLSHDATH	FLSHADL	FLSHADH	FLSHECC	FLSHCMD	ISPCLKF	FLSHPRTC
A030	FLSHPRT0	FLSHPRT1	FLSHPRT2	FLSHPRT3	FLSHPRT4	FLSHPRT5	FLSHPRT6	FLSHPRT7
A040	NTAFRQL	NTAFRQH	NTADUR	NTAPAU	NTBFRQL	NTBFRQH	NTBDUR	NTBPAU
A050	TCCFG1	TCCFG2	TCCFG3	-	TCPRDL	TCPRDH	TCCMPL	TCCMPH
A060	TCCPTRL	TCCPTRH	TCCPTFL	TCCPTFH	-	-	-	-
A070	QECFG1	QECFG2	QECFG3	-	QECNTL	QECNTH	QEMAXL	QEMAXH
	8	9	A	B	C	D	E	F
A008	TK2CFGA	TK2CFGB	TK2CMD	TK2CNTL	TK2CNTH	PECCCFG	PECCADL	PECCADH
A018	TK3CFGA	TK3CFGB	TK3CFGC	TK3CFGD	TK3HDTYL	TK3HDTYH	TK3LDTYL	TK3LDTYH
A028	TK3BASEL	TK3BASEH	TK3THDL	TK3THDH	TK3PUD	DECCCFG	DECCADL	DECCADH
A038	-	-	-	-	-	-	-	-
A048	BZCFG	NTPOW	NOTETU	-	-	-	-	-
A058	-	-	-	-	-	-	-	-
A068	T5CON	TL5	TH5	TT5	-	-	-	-
A078	CCCFG	-	-	-	CCDATA0	CCDATA1	CCDATA2	CCDATA3

	0	1	2	3	4	5	6	7
A080	PWMCFG1	PWMCFG2	PWMCFG3	-	-	-	-	-
A090	LINCTRL	LINCNTRH	LINCNTRL	LINSBRH	LINSBRL	LININT	LININTEN	-
A0A0	-	SBAUD3H	SBAUD3L	SBAUD4H	SBAUD4L	-	-	-
A0B0	LINTCON	TXDTOL	TXDTOH	RXDTOL	RXDTOH	BSDCLRL	BSDCLRH	BSDWKC
A0C0	-	-	-	-	-	-	-	-
A0D0	-	-	-	-	-	-	-	-
A0E0	BPINTF	BPINTE	BPINTC	BPCTRL	-	-	-	-
A0F0	PC1AL	PC1AH	PC1AT	-	PC2AL	PC2AH	PC2AT	-
	8	9	A	B	C	D	E	F
A088	PWM0DTY	PWM1DTY	PWM2DTY	PWM3DTY	PWM4DTY	PWM5DTY	-	-
A098	DBPCIDL	DBPCIDH	DBPCIDT	DBPCNXL	DBPCNXH	DBPCNXT	STEPCTRL	SI2CDBGID
A0A8	-	-	-	-	-	-	-	-
A0B8	BSDACT	-	-	-	-	-	-	-
A0C8	-	-	-	-	-	-	-	-
A0D8	WDT2CF	WDT2L	WDT2H	WDT3CF	WDT3L	WDT3H		
A0E8	-	-	-	-	-	-	-	-
A0F8	-	-	-	-	-	-	-	-

	0	1	2	3	4	5	6	7
A100	IOCFGO00	IOCFGO01	IOCFGO02	IOCFGO03	IOCFGO04	IOCFGO05	IOCFGO06	IOCFGO07
A110	IOCFGI00	IOCFGI01	IOCFGI02	IOCFGI03	IOCFGI04	IOCFGI05	IOCFGI06	IOCFGI07
A120	MFCFG00	MFCFG01	MFCFG02	MFCFG03	MFCFG04	MFCFG05	MFCFG06	MFCFG07
A130	IOCFGO20	IOCFGO21	IOCFGO22	IOCFGO23	IOCFGO24	IOCFGO25	IOCFGO26	IOCFGO27
A140	IOCFGI20	IOCFGI21	IOCFGI22	IOCFGI23	IOCFGI24	IOCFGI25	IOCFGI26	IOCFGI27
A150	MFCFG20	MFCFG21	MFCFG22	MFCFG23	MFCFG24	MFCFG25	MFCFG26	MFCFG27
A160	-	-	-	-	-	-	-	-
A170	-	-	-	-	-	-	-	-
	8	9	A	B	C	D	E	F
A108	IOCFGO10	IOCFGO11	IOCFGO12	IOCFGO13	IOCFGO14	IOCFGO15	IOCFGO16	IOCFGO17
A118	IOCFGI10	IOCFGI11	IOCFGI12	IOCFGI13	IOCFGI14	IOCFGI15	IOCFGI16	IOCFGI17
A128	MFCFG10	MFCFG11	MFCFG12	MFCFG13	MFCFG14	MFCFG15	MFCFG16	MFCFG17
A138	IOCFGO30	IOCFGO31	IOCFGO32	IOCFGO33	IOCFGO34	IOCFGO35	IOCFGO36	IOCFGO37
A148	IOCFGI30	IOCFGI31	IOCFGI32	IOCFGI33	IOCFGI34	IOCFGI35	IOCFGI36	IOCFGI37
A158	MFCFG30	MFCFG31	MFCFG32	MFCFG33	MFCFG34	MFCFG35	MFCFG36	MFCFG37
A168	-	-	-	-	-	-	-	-
A178	-	-	-	-	-	-	-	-

	0	1	2	3	4	5	6	7
A180	IOCFGO40	IOCFGO41	IOCFGO42	IOCFGO43	IOCFGO44	IOCFGO45	IOCFGO46	IOCFGO47
A190	IOCFGI40	IOCFGI41	IOCFGI42	IOCFGI43	IOCFGI44	IOCFGI45	IOCFGI46	IOCFGI47
A1A0	MFCFG40	MFCFG41	MFCFG42	MFCFG43	MFCFG44	MFCFG45	MFCFG46	MFCFG47
A1B0	IOCFGO60	IOCFGO61	IOCFGO62	IOCFGO63	IOCFGO64	IOCFGO65	IOCFGO66	IOCFGO67
A1C0	IOCFGI60	IOCFGI61	IOCFGI62	IOCFGI63	IOCFGI64	IOCFGI65	IOCFGI66	IOCFGI67
A1D0	MFCFG60	MFCFG61	MFCFG62	MFCFG63	MFCFG64	MFCFG65	MFCFG66	MFCFG67
A1E0	-	-	-	-	-	-	-	-
A1F0	-	-	-	-	-	-	-	-
	8	9	A	B	C	D	E	F
A188	IOCFGO50	IOCFGO51	IOCFGO52	IOCFGO53	IOCFGO54	IOCFGO55	IOCFGO56	IOCFGO57
A198	IOCFGI50	IOCFGI51	IOCFGI52	IOCFGI53	IOCFGI54	IOCFGI55	IOCFGI56	IOCFGI57
A1A8	MFCFG50	MFCFG51	MFCFG52	MFCFG53	MFCFG54	MFCFG55	MFCFG56	MFCFG57
A1B8	IOCFGO70	IOCFGO71	IOCFGO72	IOCFGO73	IOCFGO74	IOCFGO75	IOCFGO76	IOCFGO77
A1C8	IOCFGI70	IOCFGI71	IOCFGI72	IOCFGI73	IOCFGI74	IOCFGI75	IOCFGI76	IOCFGI77
A1D8	MFCFG70	MFCFG71	MFCFG72	MFCFG73	MFCFG74	MFCFG75	MFCFG76	MFCFG77
A1E8	-	-	-	-	-	-	-	-
A1F8	-	-	-	-	-	-	-	-

1. 8051 CPU

1.1 CPU Register

ACC (0xE0) Accumulator R/W (0x00)

	7	6	5	4	3	2	1	0
RD	ACC[7-0]							
WR	ACC[7-0]							

ACC is the CPU accumulator register and engages in the direct operations of many instructions. ACC is bit addressable.

B (0xF0) B Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	B[7-0]							
WR	B[7-0]							

B register is used in standard 8051 multiply and divide instructions and is also used as an auxiliary register for temporary storage. B is also bit addressable.

PSW (0xD0) Program Status Word R/W (0x00)

	7	6	5	4	3	2	1	0
RD	CY	AC	FO	RS1	RS0	OV	UD	P
WR	CY	AC	FO	RS1	RS0	OV	UD	P

CY	Carry Flag
AC	Auxiliary Carry Flag (BCD Operations)
F0	General Purpose Flag 0
RS1, RS0	Register Bank Select
OV	Overflow Flag
UD	User Defined (reserved)
P	Parity Flag

SP (0x81) Stack Pointer R/W (0x00)

	7	6	5	4	3	2	1	0
RD	SP[7-0]							
WR	SP[7-0]							

PUSH will result ACC to be written to SP+1 address. POP will load ACC from IRAM with the address of SP.

ESP (0x9B) Extended Stack Pointer R/W (0x00)

	7	6	5	4	3	2	1	0
RD	ESP[7-0]							
WR	ESP[7-0]							

In FLAT address mode, ESP and SP together form a 16-bit address for stack pointer. ESP holds the higher byte of the 16-bit address.

STATUS (0xC5) Program Status Word RO(0x00)

	7	6	5	4	3	2	1	0
RD	-	HIP	LIP	-	SPTA1	SPRA1	SPTA0	SPRA0
WR	-	-	-	-	-	-	-	-

HIP	High Priority (HP) Interrupt Status HIP=0 indicates no HP interrupt HIP=1 indicates HP interrupt progressing
LIP	Low Priority (LP) Interrupt Status LIP=0 indicates no LP interrupt

	LIP=1 indicates LP interrupt progressing
SPTA1	UART1 Transmit Activity Status SPTA1=0 indicates no UART1 transmit activity SPTA1=1 indicates UART1 transmit active
SPRA1	UART1 Receive Activity Status SPRA1=0 indicates no UART1 receive activity SPRA1=1 indicates UART1 receive active
SPTA0	UART0 Transmit Activity Status SPTA0=0 indicates no UART0 transmit activity SPTA0=1 indicates UART0 transmit active
SPRA0	UART0 Receive Activity Status SPRA0=0 indicates no UART0 receive activity SPRA0=1 indicates UART0 receive active

The program should check status conditions before entering SLEEP, STOP, or IDLE modes to prevent loss of intended functions from delayed entry until these events are finished.

1.2 Addressing Timing and Memory Modes

The clock speed of an MCU with embedded flash memory is usually limited by the access time of on-chip flash memory. While in modern process technology, the CPU can operate much faster and the access time of flash memory is usually around 40 nanoseconds, which becomes a bottleneck for CPU performance. To mitigate this problem, a programmable wait state function is incorporated to allow a faster CPU clock rate to access slower embedded flash memory. The wait state is controlled by the WTST register as shown in the following,

WTST (0x92) R/W (0x07)

	7	6	5	4	3	2	1	0
RD	-	-	-	-	WTST3	WTST2	WTST1	WTST0
WR	-	-	-	-	WTST3	WTST2	WTST1	WTST0

WTST[3-0] Wait State Control register. WTST sets the wait state in CPU clock period.

WTST3	WTST2	WTST1	WTST0	Wait State Cycle
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

The default setting of the wait state control register after reset is 0x07 and the software must initialize the setting to change the wait state setting. Using a SYSCLK of 4MHz, the WTST can be set to minimum because one clock period is 250ns, which is longer than the typical embedded flash access time. If SYSCLK is above 16MHz, then WTST should be set higher than 1 to allow enough read access time.

MCON (0xC6) XRAM Relocation Register R/W (0x00) TA Protected

	7	6	5	4	3	2	1	0
RD	MCON[7-0]							
WR	MCON[7-0]							

MCON holds the starting address of XRAM in 2KB steps. For example, if MCON[7-0]=0x01, the starting address is 0x001000h. MCON is not meaningful in this chip because it only contains on-chip XRAM and MCON should not be modified from 0x00.

The LARGE mode, addressing mode is compatible with standard 8051 in 16-bit address. FLAT mode extends the program address to 20-bit and expands the stack space to 16-bit data space. The data space is always 16-bit in either LARGE or FLAT mode.

ACON (0x9D) R/W (0x00) TA Protected

	7	6	5	4	3	2	1	0
RD	-	-	IVECSEL	-	DPXREN	SA	AM1	AM0
WR	-	-	IVECSEL	-	DPXREN	SA	AM1	AM0

ACON is addressing mode control register.

IVECSEL Interrupt Vector Selection
INTVSEC=1 maps the interrupt vector to B000 space.
INTVSEC=0 maps to normal 0x0000 space

DPXREN DPXR Register Control Bit.
If DPXREN is 0, “MOVX, @Ri” instruction uses P2 (0xA0) register and XRAM Address [15-8].
If DPXREN is 1, DPXR (0xDA) register and XRAM Address [15-8] are used.

SA Extended Stack Address Mode Indicator. This bit is read-only.
0 – 8051 standard stack mode where stack resides in internal 256-byte memory
1 – Extended stack mode. The stack pointer is ESP: SP in 16-bit addressing to data space.

AM1, AM0 AM1 and AM0 Address Mode Control Bits
00 – LARGE address mode in 16-bit
1x – FLAT address mode with 20-bit program address

1.3 MOVX A, @Ri Instructions

DPXR (0xDA) R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DPXR[7-0]							
WR	DPXR[7-0]							

DPXRis used to replace P2 [7-0] for the high byte of XRAM address bit [15-7] for ”MOVX, @Ri” instructions only if DPXREN=1.

MXAX (0xEA) MOVX Extended Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	MXAX[7-0]							
WR	MXAX[7-0]							

MXAX is used to provide the top 8-bit address for “MOVX @Ri” instructions only. MXAX does not affect other MOVX instructions.

When accessing XRAM using “MOVX, @DPTR” instruction, the address of XRAM access is formed by DPHi: DPLi depending on which data pointer is selected. Another form of MOVX instruction is “MOVX, @Ri”. This instruction provides an efficient programming method to move content within a 256-byte data block. In the “@RI” instruction, the XRAM address [15-7] can be derived from two sources. If ACON.DPXREN = 0, the high order address [15-8] is from P2 (0xA0), if ACON.DPXREN = 1, the high order address is from DPXR (0xDA) register.

The maximum addressing space of XRAM is up to 16MB and requires a 24-bit address. For “MOVX, @DPTR”, the XRAMADDR [23-16] is from either DPX (0x93) or DPX1 (0x95) depending on which data pointer is selected. For “MOVX, @Ri”, the XRAMUADDR [23-16] is from MXAX (0xEA) register.

1.4 Dual Data Pointers and MOVX operations

In standard 8051, there is only one data pointer DPH: DPL to perform MOVX. The enhanced CPU provides 2nd data pointer DPH1:DPL1 to speed up the movement or copy of data block. The active DPTR is selected by setting DPS (Data Pointer Select) register. Through the control DPS, efficient programming can be achieved.

DPS (0x86) Data Pointer Select R/W (0x00)

	7	6	5	4	3	2	1	0
RD	ID1	ID0	TSL	-	-	-	-	SEL
WR	ID1	ID0	TSL	-	-	-	-	SEL

ID[1:0] Define the operation of Increment Instruction of DPTR, “INC DPTR”. Standard 8051 only has increment DPTR instruction. ID [1-0] changes the definitions of “INC DPTR” instruction and allows flexible modifications of DPTR when “INC DPTR” instructions are executed.

ID1	ID0	SEL=0	SEL=1
0	0	INC DPTR	INC DPTR1
0	1	DEC DPTR	INC DPTR1
1	0	INC DPTR	DEC DPTR1
1	1	DEC DPTR	DEC DPTR1

TSL Enable toggling selection of DPTR selection. When this bit is set, the selection of DPTR is toggled when DPTR is used in instruction and executed.

SEL DPTR selection bit. Set to select DPTR1, and clear to select DPTR. SEL is also affected by the state of ID [1:0] and TSL after DPTR is used in an instruction. When read, SEL reflects the current selection of command.

DPL (0x82) Data Pointer Low R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DPL[7-0]							
WR	DPL[7-0]							

DPL register holds the low byte of data pointer, DPTR.

DPH (0x83) Data Pointer High R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DPH[7-0]							
WR	DPH[7-0]							

DPH register holds the high byte of data pointer, DPTR.

DPL1 (0x84) Extended Data Pointer Low R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DPL1[7-0]							
WR	DPL1[7-0]							

DPL1 register holds the low byte of extended data pointer 1, DPTR1.

DPH1 (0x85) Extended Data Pointer High R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DPH1[7-0]							
WR	DPH1[7-0]							

DPH1 register holds the high byte of extended data pointer 1, DPTR1.

DPX (0x93) Data Pointer Top R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DPX[7-0]							
WR	DPX[7-0]							

DPX is used to provide the top 8-bit address of DPTR when the address is above 64KB. The lower 16-bit address is formed by DPH and DPL. DPX is not affected in LARGE mode and will form a full 24-bit address in FLAT mode, meaning auto increment and decrement when DPTR is changed. DPX value has no effect if on-chip data memory is less than 64KB.

DPX1 (0x95) Extended Data Pointer Top R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DPX1[7-0]							
WR	DPX1[7-0]							

DPX1 is used to provide the top 8-bit address of DPTR when the address is above 64KB. The lower 16-bit address is formed by DPH1 and DP1L. DPX1 is not affected in LARGE mode and will form a full 24-bit address in Flat mode, meaning auto increment and decrement when DPTR is changed. DPX1 value has no effect if on-chip data memory is less than 64KB.

1.5 Interrupt System

The CPU implements an enhanced Interrupt Control that allows a total of 15 interrupt sources and each with two programmable priority levels. The interrupts are sampled at the rising edge of SYSCLK. If interrupts are present and enabled, the CPU enters the interrupt service routine by vectoring to the highest priority interrupt. Among the 15 interrupt sources, 7 of them are from CPU internal integrated peripherals, 6 of them are from on-chip external peripherals, and 2 of them are used for external pin interrupt expansion. When an interrupt is shared, the interrupt service routine must determine which source is requesting the interrupt by examining the corresponding interrupt flag of sharing peripherals.

The following table shows the interrupt sources and corresponding interrupt vectors. The Flag Reset column shows whether the corresponding interrupt flag is cleared by hardware (self-cleared) or software. Software can only clear the interrupt flag but not set the interrupt flag. The Natural Priority column shows the inherent priority if more than one interrupt is assigned to the same priority level. The interrupts assigned with higher priority levels always get serviced first compared with interrupts assigned with lower priority levels regardless of the natural priority sequence.

Interrupt	Peripheral Source Description	Vectors (*Note) IVECSEL=0/1	FLAG RESET	Natural Priority
PINT0	Expanded Pin INT0.x	0x0003/0xX003	Software	1
TF0	Timer 0	0x000B/0xX00B	Hardware	2
PINT1	Expanded Pin INT1.x	0x0013/0xX013	Software	3
TF1	Timer 1	0x001B/0xX01B	Hardware	4
TI0/RI0	UART0	0x0023/0xX023	Software	5
TF2	Timer 2	0x002B/0xX02B	Software	6
TI2/RI2	EUART2/LIN/LIN_FAULT	0x0033/0xX033	Software	7
I2CM	I^2C Master	0x003B/0xX03B	Software	8
INT2	LVT	0x0043/0xX043	Software	9
INT3	TKC2/TKC3	0x004B/0xX04B	Software	10
INT4	Reserved	0x0053/0xX053	Software	11
WDIF	Watchdog WDT1	0x005B/0xX05B	Software	12
INT6	PWM/TCC/QE	0x0063/0xX063	Software	13
INT7	SPI/I2C Slave	0x006B/0xX06B	Software	14
INT8	T3/T4/T5/Buzzer	0x0073/0xX073	Software	15
ECC	ECC/WDT2	0x007B/0xX07B	Software	0

BKP	Break Point	0xX080	Software	0
DBG	I2CS Debug	0xX0C0	Software	0

* Note: When IVECSEL=1, the interrupt vector is relocated to the top available 4KB memory space for boot code usage. Therefore, X value is based on the MCU embedded flash size like X=F for 64K, X=B for 48K, X=7 for 32K, and X=3 for 16K flash size. In addition to the 15 peripheral interrupts, there are two highest priority interrupts associated with debugging and breakpoint. DBG interrupt is generated when I²C slave is configured as a debug port and a debug request from the host matches the debug ID. BKP interrupt is generated when breakpoint match condition occurs. DBG has a higher priority than BKP. The BKP and DBG interrupts are not affected by the global interrupt enable, EA bit, IE register (0xA8).

The interrupt-related registers are listed in the following. Each interrupt can be individually enabled or disabled by setting or clearing the corresponding bit in IE, EXIE, and integrated peripherals' control registers.

IE (0xA8) Interrupt Enable Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	EA	ES2	ET2	ES0	ET1	PINT1EN	ET0	PINT0EN
WR	EA	ES2	ET2	ES0	ET1	PINT1EN	ET0	PINT0EN

EA	Global Interrupt Enable bit.
ES2	LIN-capable16550-likeUART2 Interrupt Enable bit.
ET2	Timer 2 Interrupt Enable bit.
ES0	UART0 Interrupt Enable bit.
ET1	Timer 1 Interrupt Enable bit.
PINT1EN	Pin PINT1.x Interrupt Enable bit.
ET0	Timer 0 Interrupt Enable bit.
PINT0EN	Pin PINT0.x Interrupt Enable bit.

EXIE (0xE8) Extended Interrupt Enable Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	EINT8	EINT7	EINT6	EWDI	EINT4	EINT3	EINT2	EI2CM
WR	EINT8	EINT7	EINT6	EWDI	EINT4	EINT3	EINT2	EI2CM

EINT8	Timer 3, Timer 4, Timer 5, and Buzzer Interrupt Enable bit.
EINT7	SPI and I²C Slave Interrupt Enable bit.
EINT6	PWM, Timer with Compare/Capture (TCC), Quadrature Encoder (QE) Interrupt Enable bit.
EWD1	Watchdog Timer Interrupt Enable bit.
EINT4	Reserved
EINT3	Touch Key Controller II (TKC2) and Touch Key Controller III (TKC3) Interrupt Enable bit.
EINT2	Low Voltage Detection (LVT) Interrupt Enable bit.
EI2CM	I²C Master Interrupt Enable bit.

Each interrupt can be individually assigned to either high or low. When the corresponding bit is set to 1, it indicates it is of high priority.

IP (0xB8) Interrupt Priority Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-	PS2	PT2	PS0	PT1	PX1	PT0	PX0
WR	-	PS2	PT2	PS0	PT1	PX1	PT0	PX0

PS2	LIN-capable 16550-like UART2 Priority bit.
PT2	Timer 2 Priority bit.
PS0	UART 0 Priority bit.
PT1	Timer 1 Priority bit.
PX1	Pin Interrupt INT1 Priority bit.
PT0	Timer 0 Priority bit.
PX0	Pin Interrupt INT0 Priority bit.

EXIP (0xF8) Extended Interrupt Priority Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PINT8	PINT7	PINT6	PWDI	PINT4	PINT3	PINT2	PI2CM
WR	PINT8	PINT7	PINT6	PWDI	PINT4	PINT3	PINT2	PI2CM

PINT8	INT8 Timer 3, Timer 4, Timer 5 and Buzzer Priority bit.
PINT7	INT7 SPI and I²C Slave Priority bit.
PINT6	INT6 PWM, Timer with Compare/Capture (TCC) and Quadrature Encoder (QE) Priority bit.
PWDI	Watchdog Priority bit.
PINT4	Reserved for INT4 Priority bit.
PINT3	INT3 Touch Key Controller II (TKC2) and Touch Key Controller III (TKC3) Priority bit.
PINT2	INT2 Low Voltage Detection (LVT) Priority bit.
PI2CM	I²C Master Priority bit.

EXIF (0x91) Extended Interrupt Flag R/W (0x00)

	7	6	5	4	3	2	1	0
RD	INT8F	INT7F	INT6F	-	INT4F	INT3F	INT2F	I2CMIF
WR	-	-	-	-	-	-	-	I2CMIF

INT8F	INT8 Timer 3, Timer 4, Timer 5, and Buzzer Interrupt Flag bit
INT7F	INT7 SPI and I²C Slave interrupt Flag bit
INT6F	INT6 PWM, Timer with Compare/Capture (TCC) and Quadrature Encoder (QE) Interrupt Flag bit
INT4F	Reserved for INT4 Interrupt Flag bit
INT3F	INT3 Touch Key Controller II (TKC2) and Touch Key Controller III (TKC3) Interrupt Flag bit
INT2F	INT2 Low Voltage Detection (LVT) Interrupt Flag bit
I2CMIF	I²C Master Interrupt Flag bit. This bit must be cleared by software
Note:	Writing to INT2F to INT8F has no effect.

The interrupt flag of internal peripherals is stored in the corresponding flag registers in the peripheral and EXIF registers. These peripherals include T0, T1, T2, and WDT. Software needs to clear the corresponding flags located in the peripherals (for T0, T1, T2, and WDT). For I2CM, the interrupt flag is located in the EXIF register bit I2CMIF. This needs to be cleared by software.

INT2 to INT8 is used to connect to the external peripherals. INT2F to INT8F is the direct equivalent of the interrupt flag from the corresponding peripherals. These peripherals include Timer 3, Timer 4, Timer 5, Buzzer, SPI, I2CS, PWM, TCC, QE, TKC2, TKC3, etc.

WKMASK (0x9F) R/W (0xFF) Wake Up Mask Register TB Protected

	7	6	5	4	3	2	1	0
RD	WEINT8	WEINT7	WEINT6	WEINT4	WEINT3	WEINT2	WEPINT1	WEPINT0
WR	WEINT8	WEINT7	WEINT6	WEINT4	WEINT3	WEINT2	WEPINT1	WEPINT0

WEINT8	Set this bit to allow INT8 to trigger the wake-up of CPU from STOP modes.
WEINT7	Set this bit to allow INT7 to trigger the wake-up of CPU from STOP modes.
WEINT6	Set this bit to allow INT6 to trigger the wake-up of CPU from STOP modes.
WEINT4	Set this bit to allow INT4 to trigger the wake-up of CPU from STOP modes.
WEINT3	Set this bit to allow INT3 to trigger the wake-up of CPU from STOP modes.
WEINT2	Set this bit to allow INT2 to trigger the wake-up of CPU from STOP modes.
WEPINT1	Set this bit to allow INT1 to trigger the wake-up of CPU from STOP modes.
WEPINT0	Set this bit to allow INT0 to trigger the wake-up of CPU from STOP modes.

WKMASK register defines the wake-up control of the interrupt signals from the STOP/SLEEP mode. The wake-up is performed by these interrupts and SYSCLK resumes if the internal oscillator is turned on. The interrupt can be set as a level trigger or an edge trigger and the wake-up always runs in accordance with the edge. The wake-up control is wired separately from the interrupt logic, therefore, after waking up, the CPU does not necessarily enter the interrupt service routine if the corresponding interrupt is not enabled. In this case, the CPU continues onto the next instruction, which initiates the STOP/SLEEP mode. Extra attention should be exerted for the modes of exit and re-entry to ensure proper operation.

All clocks are stopped in STOP/SLEEP mode, therefore, peripherals which require clock such as Timer 3, Timer 4, Buzzer, SPI, PWM, UART0, and LVD cannot perform the wake-up function. Only external pins and peripherals that do not require a clock (or can use SOSC32KHz clock) can be used for wake-up purposes. Such peripherals are like I2CS1, LIN, WDT2, Timer 5, and TK3.

PINT0 and PINT1 are used for external GPIO pin Interrupts. All GPIO pins can be enabled to generate PINT0 or PINT1 depending on its MFCFG register setting. Each GPIO pin also contains the rising/falling edge detections and either or both edges can be used for interrupt triggering. The same signaling can be used for generating wake-up.

TCON (0x88) R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TF1	TR1	TF0	TR0	PINT1F	-	PINT0F	-
WR	-	TR1	-	TR0	PINT1F	-	PINT0F	-

TF1	Timer 1 Interrupt Flag bit. TF1 is cleared by hardware when entering the interrupt routine.
TR1	Timer 1 Run Control bit. Set to enable Timer 1.
TF0	Timer 0 Interrupt Flag. TF0 is cleared by hardware when entering the interrupt routine.
TR0	Timer 0 Run Control bit. Set to enable Timer 0.
PINT1F	Pin INT1 Interrupt Flag bit. PINT1F is cleared by hardware when entering the interrupt routine.
PINT0F	Pin INT0 Interrupt Flag bit. PINT0F is cleared by hardware when entering the interrupt routine.

1.6 Register Access Control

One important aspect of the embedded MCU is its reliable operations in a harsh environment. Many system failures result from the accidental loss of data or changes of critical registers that may lead to catastrophic effects. The CPU provides several protection mechanisms, which are described in this section.

TA (0xC7) Time Access A Control Register2 WO xxxxxxx0

	7	6	5	4	3	2	1	0
RD	-	-	-	-	-	-	-	TASTAT
WR	TA Register							

TA access control emulates a ticket that must be purchased before modifying a critical register. To modify or write into a TA protected register, TA must be accessed in a predefined sequence to obtain the ticket. The ticket is used when an intended modification operation is done to the TA protected register. To obtain the next access a new ticket must be obtained again by performing the same predefined sequence on TA. TA does not limit the read access of the TA protect registers. The TA protected register includes RWT bit of WDCON (0xD8), MCON (0xC6), and ACON (0x9D) registers. The following predefined sequence is required to modify the content of MCON.

```
MOV TA, #0xAA;
MOV TA, #0x55;
MOV MCON, #0x01;
```

Once access is granted, there is no time limitation of access. The access is voided if any operation is performed in TA address. When read, TASTAT indicates whether TA is locked or not (1 indicates “unlock” and 0 indicates “lock”).

TB (0xC9) Time Access B Control Register2 RW (0x00)

	7	6	5	4	3	2	1	0
RD	-	-	-	-	-	-	-	TBSTAT
WR	TB Register							

TB access control functions are like TA control, except the ticket is for multiple uses with a time limit. Once access is granted, the access is open for 256 clock periods and then expires. The software can also read TB addresses to obtain the current TB status. The TB protected registers include two SFR registers, CKSEL (0x8F) and WKMASK (0x9F), and several XFR registers, such as FLSHCMD (0xA025), ISPCLKF (0xA026), FLSHPRTC (0xA027), FLSHPRT0 (0xA030), BPINTE (0xA0E1), and SI2C_DebugID (0xA09F) etc. To modify registers with TB protection, the following procedure must be performed.

```
MOV TB, #0xAA
MOV TB, #0x55
```

This action creates a timed window of 256 SYSCLK periods to allow write access of these TB protected registers. If any of the above-mentioned sequences are repeated before the 128 cycles expire, a new 128 cycles is extended. The current 256 cycles can be terminated immediately by writing #0x00 to TB registers, such as

MOV TB, #0x00

It is recommended to terminate the TB access window once the user program finishes the modifications of TB protected registers.

Because TA and TB are critical reassurance of the reliable operation of the MCU that prevents accidental hazardous uncontrollable modifications of critical registers, the operation of these two registers should bear extreme cautions. It is strongly advised that these two registers should be turned on only when needed. Both registers use synchronous CPU clock, therefore, it is imperative that any running tasks of TA and TB should be terminated before entering IDLE mode or STOP mode. Both modes turn off the CPU clock and if TA and TB are enabled, they stay enabled until the CPU clock resumes, and thus may create vulnerabilities for critical registers.

Another reliability concern of embedded Flash MCU is that the important content in the Flash can be accidentally erased. This concern is addressed by the content protection in the Flash controller.

1.7 Clock Control and Power Management Modes

This section describes the clock control and power-saving modes of the CPU and its integrated peripherals. The settings are controlled by PCON (0x87) and PMR (0xC4) registers. The register description is defined as follows.

PCON (0x87) R/W (0x00)

	7	6	5	4	3	2	1	0
RD	SMOD0	-	-	-	-	-	-	-
WR	SMOD0	-	-	-	-	SLEEP	STOP	IDLE

SMOD0 — UART 0 Baud Rate Control. This is used to select double baud rate in mode 1, 2 or 3 for UART0 using Timer 1 overflow. This definition is the same as standard 8051.

SLEEP — Sleep Mode Control Bit. When this bit and the Stop bit are set to 1, the clock of the CPU and all peripherals is disabled and enters SLEEP mode. The SLEEP mode exits when non-clocked interrupts or resets occur. Upon exiting SLEEP mode, the Sleep bit and Stop bit in PCON is automatically cleared. In terms of power consumption, the following relationship applies: IDLE mode > STOP mode > SLEEP mode. SLEEP mode is the same as STOP mode, except it also turns off the band gap and the regulator. It uses a very low power back-up regulator (< 5uA). When waking up from SLEEP mode, it takes a longer time (< 64 IOSC clock cycles) compared with STOP mode because the regulator requires more time to stabilize.

STOP — Stop Mode Control Bit. The clock of the CPU and all peripherals is disabled and enters STOP mode if the Sleep bit is in the reset state. The STOP mode can only be terminated by non-clocked interrupts or resets. Upon exiting STOP mode, the Stop bit in PCON is automatically cleared.

IDLE — Idle Bit. If the IDLE bit is set, the system goes into IDLE mode. In Idle mode, CPU clock becomes inactive and the CPU and its integrated peripherals such as WDT, T0/T1/T2, and UART0 are reset. But the clocks of external peripherals and CPU like PCA, LIN-capable16550-like UART2, SPI, T3, I^2C slave, and the others are still active. This allows the interrupts generated by these peripherals and external interrupts to wake up the CPU. The exit mechanism of IDLE mode is the same as STOP mode. The Idle bit is automatically cleared after the exit of the IDLE mode.

PMR (0xC4) R/W (010xxxxx)

	7	6	5	4	3	2	1	0
RD	CD1=0	CD0	SWB	-	-	-	-	-
WR	-	CD0	SWB	-	-	-	-	-

CD1, CD0 — Clock Divider Control. These two bits control the entry of PMM mode. When CD0=1, and CD1=0, full speed operation is in effect. When CD0=1, and CD1=1, the CPU enters PMM mode where CPU and its integrated peripherals operate at a clock rate divided by 257. Note that in PMM mode, all integrated peripherals such as UART0, LIN-capable 16550-like UART2, WDT, and T0/T1/T2 run at this reduced rate, and thus may not function properly. All external peripherals to CPU still operate at full speed in PMM mode.

NOTE: CD1 is internally hardwired to 0. This implementation does not support PMM mode.

SWB Switch Back Control bit. Setting this bit allows the actions to occur in integrated peripherals to automatically switch back to the normal operation mode.

NOTE: PMM mode is not supported.

CKSEL (0x8F) R/W (0x0C) System Clock Selection Register TB Protected

	7	6	5	4	3	2	1	0
RD	IOSCDIV[3-0]				-	-	CLKSEL[1]	CLKSEL[0]
WR	IOSCDIV[3-0]				REGRDY[1]	REGRDY[0]	CLKSEL[1]	CLKSEL[0]

IOSCDIV[3-0] IOSC Pre-Divider. The default is IOSC.

IOSCDIV[3-0]	SYSCLK
0	IOSC
1	IOSC/2
2	IOSC/4
3	IOSC/6
4	IOSC/8
5	IOSC/10
6	IOSC/12
7	IOSC/14
8	IOSC/16
9	IOSC/32
10	IOSC/64
11	IOSC/128
12	IOSC/256
13	IOSC/256
14	IOSC/256
15	IOSC/256

REGRDY[1-0] Wake-up delay time for main regulator stable time from reset or from sleep mode wakeup. Default is the longest delay at 256 SOSC32KHz.

REGRDY[1]	REGRDY[0]	Delay time
0	0	4 SOSC32KHz cycle
0	1	16 SOSC32KHz cycle
1	0	64 SOSC32KHz cycle
1	1	256 SOSC32KHz cycle

CLKSEL[1-0] Clock Source Selection

These two bits define the clock source of the system clock SYSCLK. The selections are shown in the following table. The default setting after reset is IOSC.

CLKSEL[1]	CLKSEL[0]	SYSCLK
0	0	IOSC (through divider)
0	1	SOSC32KHz (32KHz)
1	0	IOSC (through divider)
1	1	XCLKIN

WKMASK (0x9F) R/W (0xFF) Wake-Up Mask Register TB Protected

	7	6	5	4	3	2	1	0
RD	WEINT8	WEINT7	WEINT6	WEINT4	WEINT3	WEINT2	WEPINT1	WEPINT0
WR	WEINT8	WEINT7	WEINT6	WEINT4	WEINT3	WEINT2	WEPINT1	WEPINT0

WEINT8	Set this bit to allow INT8 to trigger the wake-up of CPU from STOP modes.
WEINT7	Set this bit to allow INT7 to trigger the wake-up of CPU from STOP modes.
WEINT6	Set this bit to allow INT6 to trigger the wake-up of CPU from STOP modes.
WEINT4	Set this bit to allow INT4 to trigger the wake-up of CPU from STOP modes.
WEINT3	Set this bit to allow INT3 to trigger the wake-up of CPU from STOP modes.
WEINT2	Set this bit to allow INT2 to trigger the wake-up of CPU from STOP modes.
WEPINT1	Set this bit to allow INT1 to trigger the wake-up of CPU from STOP modes.
WEPINT0	Set this bit to allow INT0 to trigger the wake-up of CPU from STOP modes.

WKMASK register defines the wake-up control of the interrupt signals from the STOP/SLEEP mode. The wake-up is performed by these interrupts and SYSCLK resumes if the internal oscillator is turned on. The interrupt can be set as a level trigger or an edge trigger and the wake-up always runs in accordance with the edge. The wake-up control is wired separately from the interrupt logic, and therefore after waking up, the CPU does not necessarily enter the interrupt service routine if the corresponding interrupt is not enabled. In this case, the CPU continues onto the next instruction, which initiates the STOP/SLEEP mode. Extra attention should be paid to the modes of exit and re-entry to ensure proper operation.

All clocks are stopped in STOP/SLEEP mode, therefore, peripherals that require clock such as I^2C slave, UARTx, LVD, and T3/T4 cannot perform the wake-up function. Only external pins and peripherals that do not require a clock can be used for wake-up purposes. Such peripherals are LIN Wakeup and Timer 5 with SOSC32KHz.

IDLE Mode

IDLE mode provides power saving by stopping SYSCLK from CPU and its integrated peripherals while other peripherals are still in operation with SYSCLK. Thus, other peripherals still function normally and can generate interrupts that wake up the CPU from IDLE mode. The IDLE mode is enabled by setting IDLE bit to 1.

When the CPU is in idle mode, no processing is possible. All integrated internal peripherals such as T0/T1/T2, UART0, LIN-capable 16550-like UART2 and I^2C Master are inaccessible during idling. The IDLE mode can be exited by hardware reset through RSTN pin or by external interrupts as well as the interrupts from external peripherals that are OR-ed with the external interrupts. The triggering external interrupts need to be enabled properly. Upon exiting from IDLE mode, the CPU resumes operation as the clock is turned on. CPU immediately vectors to the interrupt service routine of the corresponding interrupt sources that wake up the CPU. When the interrupt service routine is complete, RETI returns to the program and immediately follows the one that invokes the IDLE mode. Upon returning from IDLE mode to normal mode, the idle bit in PCON is automatically cleared.

STOP Mode

STOP mode provides further power reduction by stopping SYSCLK to all circuits. In STOP mode, IOSC oscillator is disabled. STOP mode is entered by setting STOP=1. To achieve minimum power consumption, it is essential to turn off all peripherals with DC current consumption. It is also important that the software switches to the IOSC clock and disables all other clock generators before entering STOP mode. This is critical to ensure a smooth transition when resuming its normal operations. Upon entering STOP mode, the system uses the last edge of IOSC clock to shut down the IOSC clock generator.

Valid interrupt/wakeup event or reset will result in the exit of STOP mode. Upon exit, STOP bit is cleared by hardware and IOSC is resumed. The triggering interrupt source must be enabled and its wake-up bit is set in the WKMASK register. CPU will resume normal operation and use previous clock settings. When an interrupt occurs, the CPU immediately vectors to the interrupting service routine of the corresponding interrupt source. When the interrupt service routine completes, RETI returns to the program immediately to execute the instruction that invokes the STOP mode.

The on-chip 1.5V regulator for core circuits is still enabled along with its reference voltage. As a result, the power consumption due to the regulator and its reference circuit is still around 100uA to 200uA. The advantage of STOP mode is its immediate resumption of the CPU.

SLEEP Mode

SLEEP mode achieves very low standby consumption by putting the on-chip 1.5V regulator in disabled state. An ultra-low-power backup regulator supplies (typical 1.4v) the internal core circuit and maintains the logic state and SRAM data. The total current drain in SLEEP mode is less than 1uA. Only the backup regulator and the SOSC32KHz circuit are still in operation in SLEEP mode.

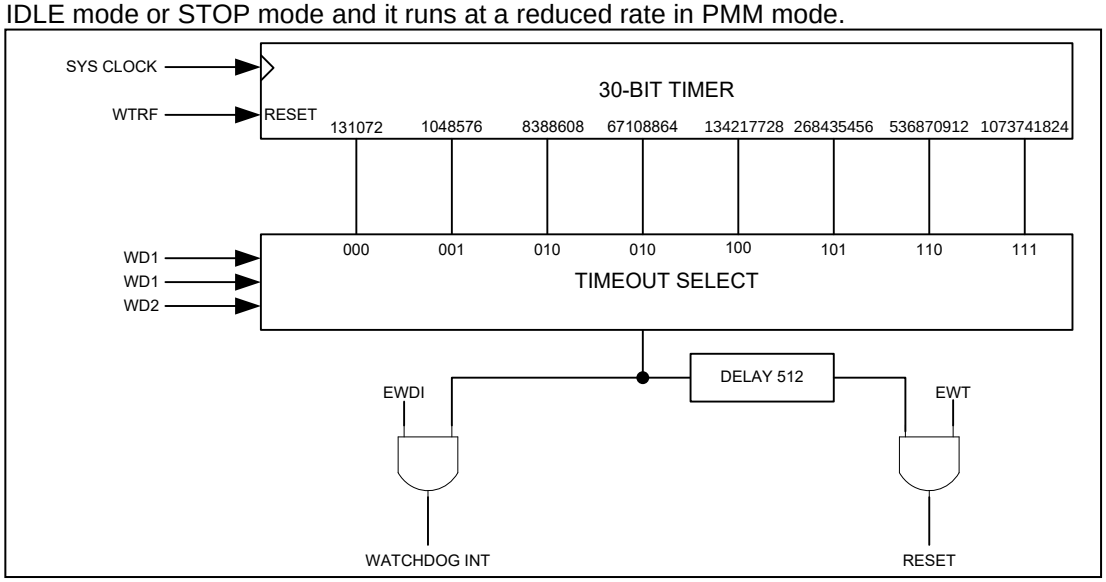
The exit of SLEEP mode is the same interrupt/wakeup event as in STOP node, and in addition the on-chip regulator is enabled, then after a delay set by REGRDY (clocked by SOSC32KHz), SYSCLK is resumed. REGRDY delay is necessary to ensure stable operation of the regulator. The larger the decoupling capacitance, the longer delay should be set.

Clock Control

The clock selection is defined by CKSEL register (0x8F). There are two selections either from divided IOSC or SOSC32KHz. The default selection is divided IOSC. The typical power consumption of CPU is 0.150mA/MHZ.

1.8 Watchdog Timer

The Watchdog Timer is a 30-bit timer that can be used by a system supervisor or as an event timer. The Watchdog timer can be used to generate an interrupt or to issue a system reset depending on the control settings. This section describes the register related to the operation of Watchdog Timer and its functions. The following diagram shows the structure of the Watchdog Timer. WDT shares the same clock with the CPU, thus, WDT is disabled in IDLE mode or STOP mode and it runs at a reduced rate in PMM mode.



WDCON (0xD8) R/W (0x02) TA protected on bit 0 RWT only

	7	6	5	4	3	2	1	0
RD	-	-	-	-	WDIF	WTRF	EWT	-
WR	-	-	-	-	WDIF	WTRF	EWT	RWT

WDIF	WDT Interrupt Flag bit. This bit is set when the session expires regardless of a WDT interrupt is enabled or not. Note the WDT interrupt enable control is located in EIE (0xE8). 4 EWDI bit. It must be cleared by software.
WTRF	WDT Reset Flag bit. WDRF is cleared by hardware reset including RSTN, POR etc. WTRF is set to 1 after a WDT reset occurs. It can be cleared by software. WTRF can be used by software to determine if a WDT reset has occurred.
EWT	Watchdog Timer Reset Enable bit. Set this bit to enable the watchdog reset function. The default WDT reset is enabled and WDT timeout is set to maximum.
RWT	Reset the Watchdog timer. Writing 1 to RWT resets the WDT timer. RWT bit is not a register and does not hold any value. The clearing action of Watchdog timer is protected by TA access. In another word, to clear Watchdog timer, TA must be unlocked and then followed by writing RWT bit to 1. If TA is still locked, the program can write 1 into RWT bit, but it does not reset the Watchdog timer.

CKCON (0x8E) R/W (0xC4)

	7	6	5	4	3	2	1	0
RD	WD1	WD0	T2CKDCTL	T1CKDCTL	T0CKDCTL	WD2	-	-
WR	WD1	WD0	T2CKDCTL	T1CKDCTL	T0CKDCTL	WD2	-	-

T2CKDCTL	Timer 2 Clock Source Division Factor Control Flag. Setting this bit to 1 sets the Timer 2 division factor to 4, the Timer 2 clock frequency equals CPU clock frequency divided by 4.

Setting this bit to 0 (the default power-on value) sets the Timer 2 division factor to 12, and the Timer 2 clock frequency equals CPU clock frequency divided by 12.

T1CKDCTL Timer 1 Clock Source Division Factor Control Flag. Setting this bit to 1 sets the Timer 1 division factor to 4, and the Timer 1 clock frequency equals CPU clock frequency divided by 4. Setting this bit to 0 (the default power-on value) sets the Timer 1 division factor to 12, and the Timer 1 clock frequency equals CPU clock frequency divided by 12.

T0CKDCTL Timer 0 Clock Source Division Factor Control Flag. Setting this bit to 1 sets the Timer 0 division factor to 4, and the Timer 0 clock frequency equals CPU clock frequency divided by 4. Setting this bit to 0 (the default power-on value) sets the Timer 0 division factor to 12, and the Timer 0 clock frequency equals CPU clock frequency divided by 12.

WD[2:0] This register controls the time-out value of WDT as in the following table. The time out value is shown as follows and the default is set to maximum:

WD2	WD1	WD0	Time Out Value
0	0	0	131072
0	0	1	1048576
0	1	0	8388608
0	1	1	67108864
1	0	0	134217728
1	0	1	268435456
1	1	0	536870912
1	1	1	1073741824

A second 16-bit Watchdog Timer (WDT2) clocked by the independent nonstop SOSC32KHz (32KHz) is included. WDT2 can be used to generate interrupt/wakeup timing from STOP/SLEEP mode, or generate software reset.

WDT2CF (0xA0D8h) WatchDog Timer 2 Configure Registers R/W (0xA7) TB Protected

	7	6	5	4	3	2	1	0
RD	-	WDT2REN	WDT2RF	WDT2IEN	WDT2CS[2-0]			WDT2IF
WR	WDT2CLR	WDT2REN	WDT2RF	WDT2IEN	WDT2CS[2-0]			WDT2IF

WDT2CLR WDT2 Counter Clear
Writing “1” to WDT2CLR clears the WDT2 count to 0. It is self-cleared by hardware.

WDT2REN WDT2 Reset Enable
WDT2REN=1 configures WDT2 to perform software reset.

WDT2RF WDT2 Reset Flag
WDT2RF is set to “1” after a WDT2 reset occurs. This must be cleared by software by writing “0”.

WDT2IEN WDT2 Interrupt Enable
WDT2IEN=1 enables WDT2 interrupt.

WDT2CS[2-0] WDT2 Clock Scaling

WDT2CS[2-0]	Clock SOSC32KHz Divider	WDT2Period
000	2^8	8 msec
001	2^8	8 msec
010	2^8	8 msec
011	2^8	8 msec
100	2^12	128 msec
101	2^13	256 msec
110	2^14	512 msec
111	2^15	1024 msec

WDT2IF WDT2 Interrupt Flag
WDT2IF is set to “1” after a WDT2 interrupt. This must be cleared by software by writing “0”.

Please note that the longest effective time WDT2 can be set is approximately 18 hours.

WDT2L (0xA0D9h) Watchdog Timer 2 Time Out Value Low Byte RW (0xFF) TB Protected

	7	6	5	4	3	2	1	0
RD	WDT2CNT[7-0]							
WR	WDT2[7-0]							

WDT2H (0xA0DAh) Watchdog Timer 2 Time Out Value High Byte RW (0x0F) TB Protected

	7	6	5	4	3	2	1	0
RD	WDT2CNT[15-8]							
WR	WDT2[15-8]							

WDT2L and WDT2H hold the time-out value for watchdog timer 2. When the counter reaches the WDT2 time-out value, an interrupt or reset is generated. Reading this register returns the current count value.

A third Watchdog Timer (WDT3) is also included for further enhancement of fault recovery. WDT3 cannot be disabled in normal mode. The clock scaling of WDT3 is the same as WDT2.

WDT2CS[2-0]	Clock SOSC32KHz Divider	WDT3 Period
000	2^8	8 msec
001	2^8	8 msec
010	2^8	8 msec
011	2^8	8 msec
100	2^12	128 msec
101	2^13	256 msec
110	2^14	512 msec
111	2^15	1024 msec

Therefore, the longest time of WDT3 is about 1 second timing 2^16 equal approximately to 18 hours. In default setting, the time of WDT3 is 8 msec timing 2^8 equal approximately to 2 seconds.

WDT3CF (0xA0DBh) WatchDog Timer 3 Configure Registers R/W (0xD1) TB Protected

	7	6	5	4	3	2	1	0
RD	-	-			-			WDT3RF
WR	WDT3CLR	-			-			WDT3RF

WDT3CLR WDT3 Counter Clear
Writing “1” to WDT3CLR clears the WDT3 count to 0. It is self-cleared by hardware.

WDT3RF WDT3 Reset Flag
WDT3RF is set to “1” after a WDT3 reset occurs. This must be cleared by software by writing “0”.

WDT3L (0xA0DCh) Watchdog Timer 3 Time Out Value Low Byte RO (0xFF) TB Protected

	7	6	5	4	3	2	1	0
RD	WDT3CNT[7-0]							
WR	WDT3[7-0]							

WDT3H (0xA0DDh) Watchdog Timer 3 Time Out Value High Byte RO (0x00) TB Protected

	7	6	5	4	3	2	1	0
RD	WDT3CNT[15-8]							
WR	WDT3[15-8]							

WDT3L and WDT3H hold the time out value for watchdog timer 3. When the counter reaches WDT2 time out value, a reset is generated. Reading this register returns the current count value.

1.9 System Timers – T0 and T1

The CPU contains three 16-bit timers/counters, Timer 0, Timer 1 and Timer 2. In timer mode, Timer 0, Timer 1 registers are incremented every 12 SYSCLK period when the appropriate timer is enabled. In the timer mode, Timer 2 registers are incremented every 12 or 2 SYSCLK period (depending on the operating mode). In the counter mode, the timer

registers are incremented every falling edge on their corresponding inputs: T0, T1, and T2. These inputs are read every SYSCLK period.

Timer 0 and Timer 1 are fully compatible with the standard 8051. Timers 0 and 1 are controlled by TCON (0x88) and TMOD (0x89) registers while each timer consists of two 8-bit registers TH0 (0x8C), TL0 (0x8A), TH1 (0x8D), TL1 (0x8B).

TCON (0x88) R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TF1	TR1	TF0	TR0	PINT1F	-	PINT0F	-
WR	-	TR1	-	TR0	PINT1F	-	PINT0F	-

TF1	Timer 1 Interrupt Flag bit. TF1 is cleared by hardware when entering the interrupt routine.
TR1	Timer 1 Run Control bit. Set to enable Timer 1.
TF0	Timer 0 Interrupt Flag. TF0 is cleared by hardware when entering the interrupt routine.
TR0	Timer 0 Run Control bit. Set to enable Timer 0.
PINT1F	Pin INT1 Interrupt Flag bit. PINT1F is cleared by hardware when entering the interrupt routine.
PINT0F	Pin INT0 Interrupt Flag bit. PINT0F is cleared by hardware when entering the interrupt routine.

TMOD (0x89h) Timer 0 and 1 Mode Control Register

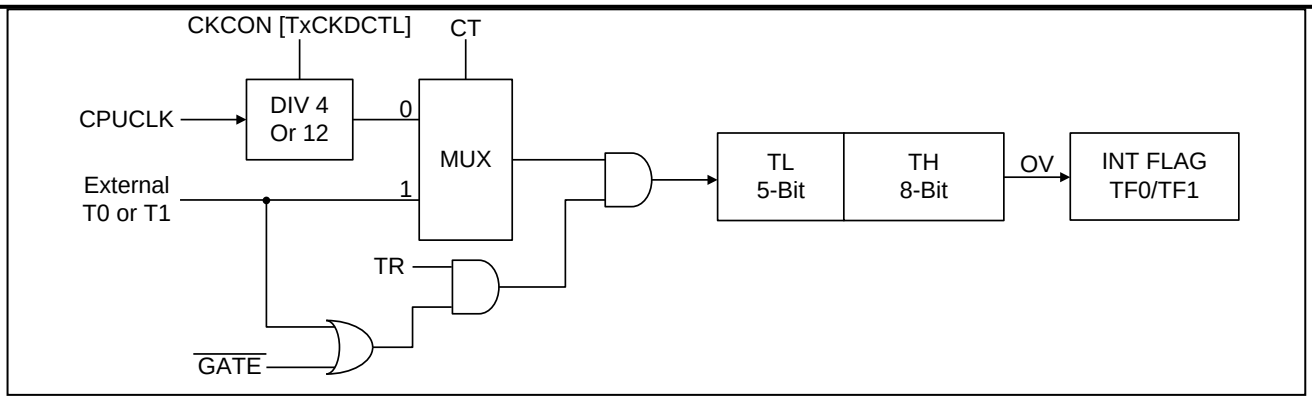
	7	6	5	4	3	2	1	0
RD	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0
WR	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0

GATE1	Timer 1 Gate Control bit. Set to enable external T1 to function as gating control of the counter.
CT1	Counter or Timer Mode Select bit. Set CT1 to access external T1 as the clock source. Clear CT1 to use the internal clock.
T1M1	Timer 1 Mode Select bit.
T1M0	Timer 1 Mode Select bit.
GATE0	Timer 0 Gate Control bit. Set to enable external T0 to function as gating control of the counter.
CT0	Counter or Timer Mode Select bit. Set CT0 to use external T0 as the clock source. Clear CT0 to use the internal clock.
T0M1	Timer 0 Mode Select bit.
T0M0	Timer 0 Mode Select bit.

M1	M0	Mode	Mode Descriptions
0	0	0	TL serves as a 5-bit pre-scaler and TH functions as an 8-bit counter/timer. They form a 13-bit operation.
0	1	1	TH and TL are cascaded to form a 16-bit counter/timer.
1	0	2	TL functions as an 8-bit counter/timer and auto-reloads from TH.
1	1	3	TL functions as an 8-bit counter/timer. TH functions as an 8-bit timer, which is controlled by GATE1. Only Timer 0 can be configured in Mode 3. When this happens, Timer 1 can only be used where its interrupt is not required.

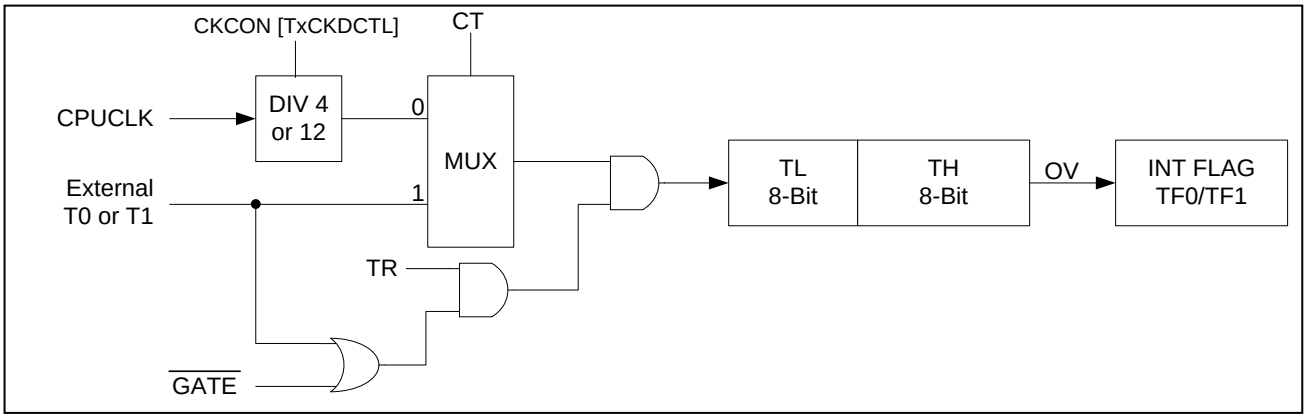
Mode 0

In this mode, TL serves as a 5-bit pre-scaler and TH functions as an 8-bit counter/timer. Both work together as a 13-bit counter/timer. The Mode 0 operation is shown in the following diagram.



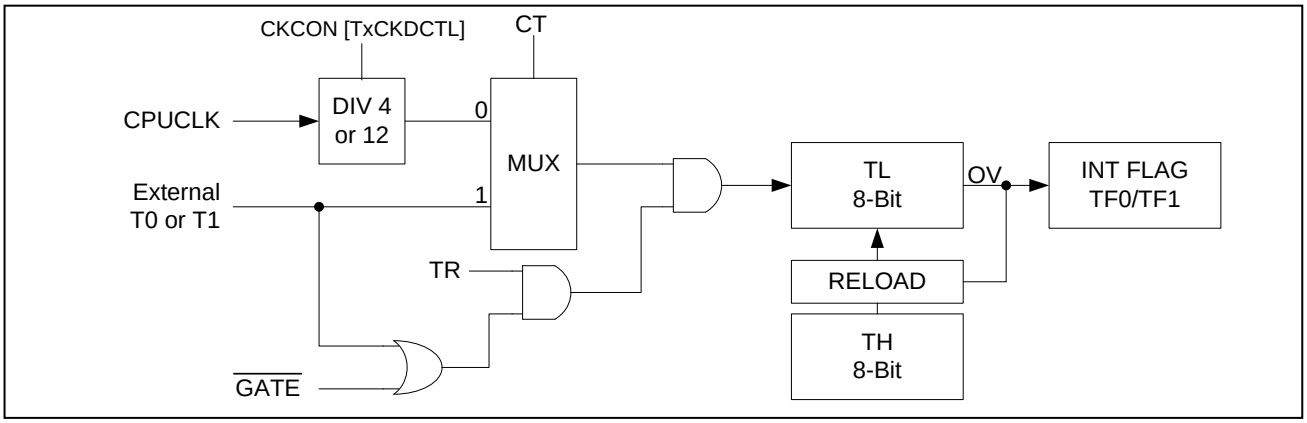
Mode 1

Mode 1 operates the same way Mode 0 does, except TL is configured as 8-bit, and thus, forms a 16-bit counter/timer. This is shown as the following diagram.



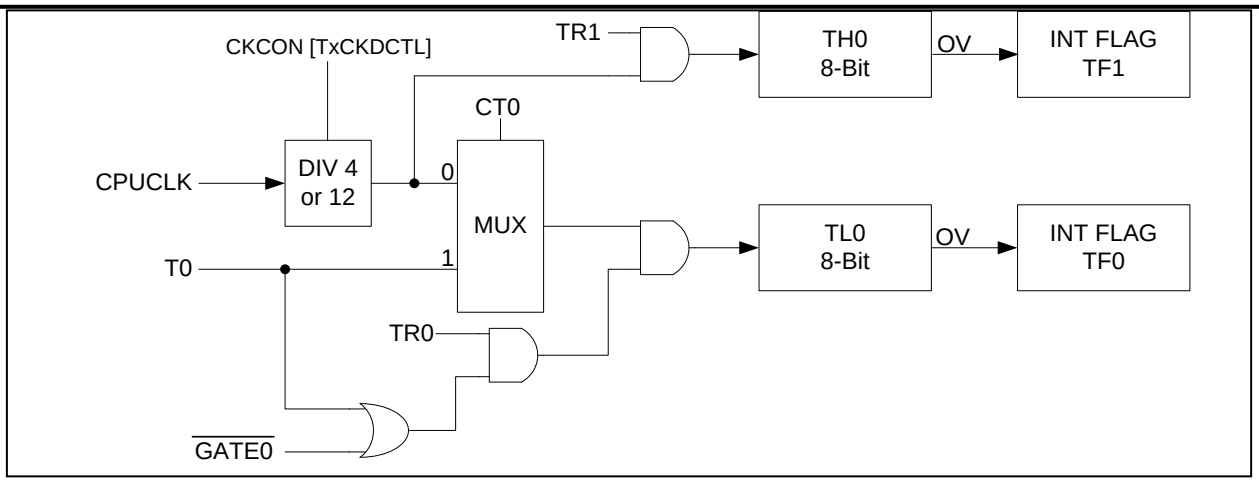
Mode 2

Mode 2 configures the timer as an 8-bit re-loadable counter. The counter is TL while TH stores the reload data. The reload occurs when TL overflows. The operation is shown in the following diagram:



Mode 3

Mode 3 is a special mode for Timer 0 only. In this mode, Timer 0 is configured as two separate 8-bit counters. TL0 uses control and interrupt flags of Timer 0, whereas TH0 uses control and interrupt flag of Timer 1. Since Timer 1's control and flag are occupied, Timer 2 can only be used for counting purposes such as Baud rate generating while Timer 0 is in Mode 3. The operation flow of Mode 3 is shown in the following diagram.


CKCON [TxCKDCTL]
TR1
TH0
8-Bit
OV
INT FLAG
TF1
CT0
CPUCLK
DIV 4
or 12
0
MUX
TL0
8-Bit
OV
INT FLAG
TF0
T0
1
TR0
GATE0

1.10 System Timer – T2

Timer 2 is fully compatible with the standard 8052 timer 2. Timer 2 can be used as the re-loadable counter, capture timer, or baud rate generator. Timer 2 uses five SFR as counter registers, capture registers and a control register.

T2CON (0xC8h) Timer 2 Control and Configuration Register

	7	6	5	4	3	2	1	0
RD	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2	CPRL2
WR	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2	CPRL2

TF2 — Timer 2 Interrupt Flag bit
TF2 must be cleared by software. TF2 is not set when RCLK or TCLK is set (that means Timer 2 is used as an UART0 Baud rate generator).

EXF2 — T2EX Falling Edge Flag bit
This bit is set when T2EX has a falling edge when EXEN2=1. EXF2 must be cleared by software.

RCLK — Receive Clock Enable bit
1 – UART0 receiver is clocked by Timer 2 overflow pulses
0 – UART0 receiver is clocked by Timer 1 overflow pulses

TCLK — Transmit Clock Enable bit
1 – UART0 transmitter is clocked by Timer 2 overflow pulses
0 – UART0 transmitter is clocked by Timer 1 overflow pulses

EXEN2 — T2EX Function Enable bit.
1 – Allows capture or reload as T2EX falling edge appears
0 – Ignore T2EX events

TR2 — Start/Stop Timer 2 Control bit
1 – Start
0 – Stop

CT2 — Timer 2 Timer/Counter Mode Select bit
1 – External event counter uses T2 pin as the clock source
0 – Internal clock timer mode

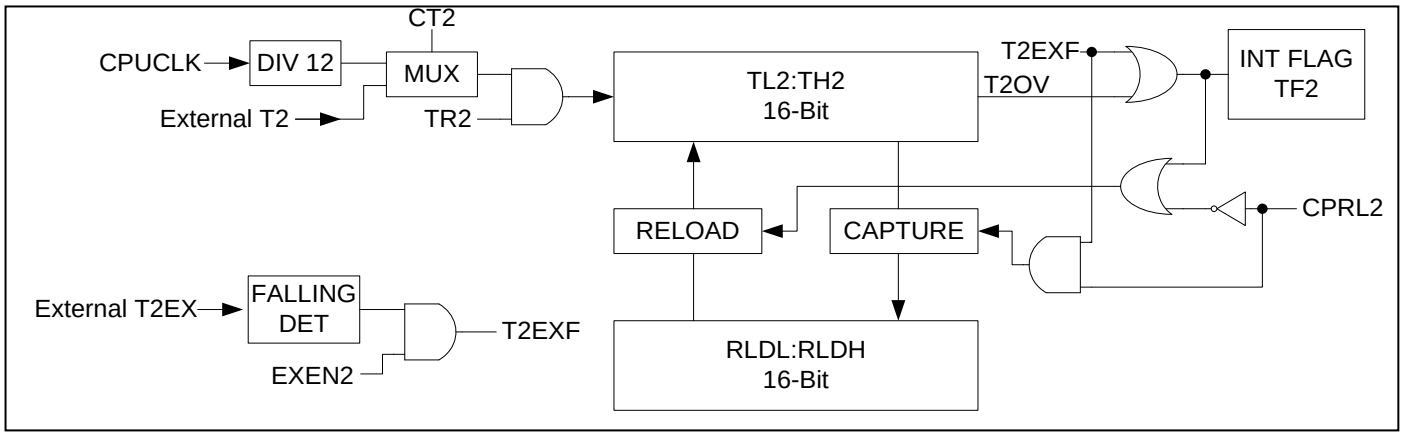
CPRL2 — Capture/Reload Select bit
1 – Use T2EX pin falling edge for capture
0 – Automatic reload on Timer 2 overflow or falling edge of T2EX (when EXEN2=1). If RCLK or TCLK is set (Timer 2 is used as a baud rate generator), this bit is ignored and an automatic reload is forced on Timer 2 overflows.

Timer 2 can be configured in three modes of operations –Auto-reload Counter, Capture Timer, or Baud Rate Generator. These modes are defined by RCLK, TCLK, CPRL2 and TR2 bits of T2CON registers. The definition is illustrated in the following table:

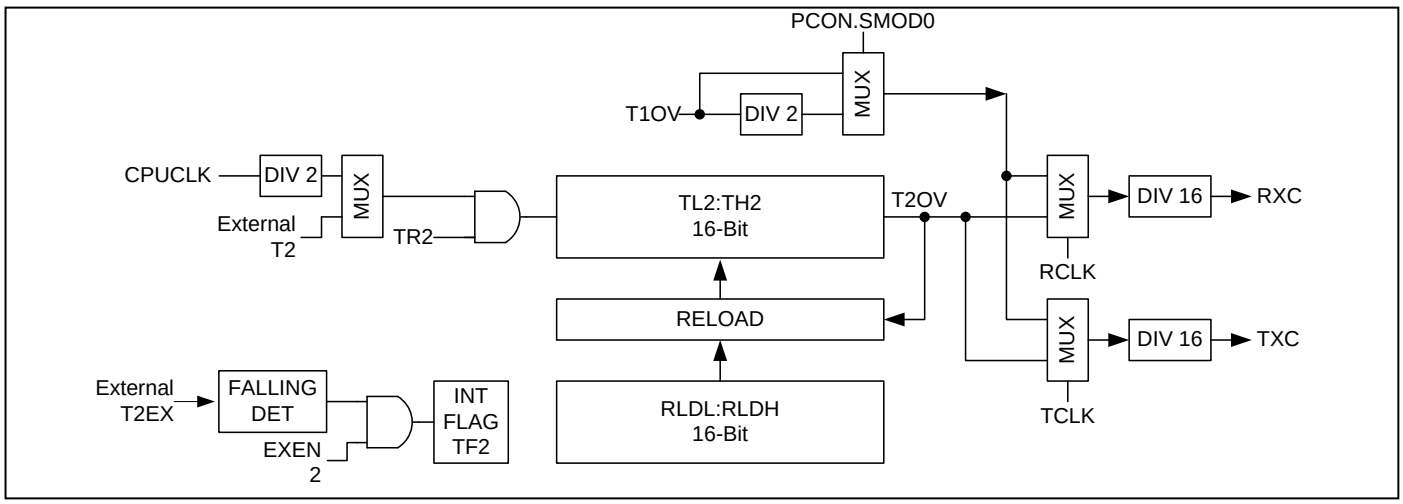
RCLK or TCLK	CPRL2	TR2	Mode Descriptions
0	0	1	16-bit Auto-reload Counter mode. Timer 2 overflow sets the TF2 interrupt flag and TH2/TL2 is reloaded with RLDH/RLHL register.
0	1	1	16-bit Capture Timer mode. Timer 2's overflow sets TF2 interrupt flag. When EXEN2=1, TH2/TL2 content is captured into RLDH/RLDL when T2EX falling edge occurs.
1	X	1	Baud Rate Generator mode. Timer 2's overflow is used for configuring UART0.
X	X	0	Timer 2 is stopped.

The block diagram of the Timer 2 operating in Auto-reload Counter and Capture Timer modes are shown in the following diagram:

External T2 and External T2EX are tied together in this product.

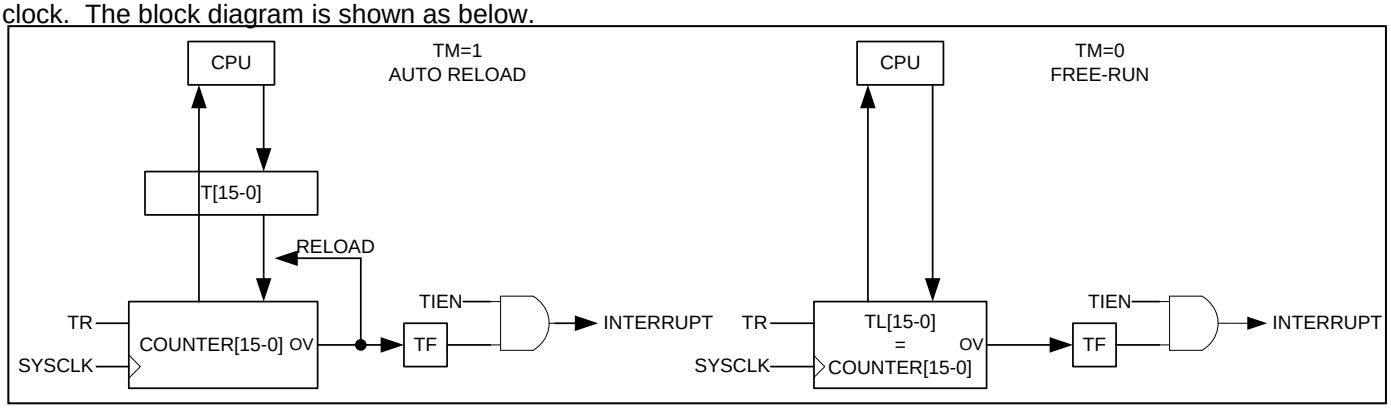


The block diagram of the Timer 2 operating in Baud Rate Generator is shown in the following diagram:



1.11 System Timer – T3 and T4

Both Timer 3 and Timer 4 are simple 16-Bit reload timers or free-run counters and are clocked by the system clock. The block diagram is shown as below.



T34CON (0xCFh) Timer 3 and Timer 4 Control and Status Register

	7	6	5	4	3	2	1	0
RD	TF4	TM4	TR4	T4IEN	TF3	TM3	TR3	T3IEN
WR	TF4	TM4	TR4	T4IEN	TF3	TM3	TR3	T3IEN

TF4	Timer 4 Overflow Interrupt Flag bit. TF4 is set by hardware when overflow condition occurs. TF4 must be cleared by software.
TM4	Timer 4 Mode Control bit. TM4 = 1 set timer 4 as auto reload, and TM4=0 set timer 4 as free-run.
TR4	Timer 4 Run Control bit. Set to enable Timer 4, and clear to stop Timer 4.

T4IEN	Timer 4 Interrupt Enable bit T4IEN=0 disables the Timer 4 overflow interrupt. T4IEN=1 enables the Timer 4 overflow interrupt.
TF3	Timer 3 Overflow Interrupt Flag bit TF3 is set by hardware when overflow condition occurs. TF3 must be cleared by software.
TM3	Timer 3 Mode Control bit. TM3 = 1 sets timer 3 as auto reload, and TM3=0 sets timer 3 as free-run.
TR3	Timer 3 Run Control bit. Set to enable Timer 3, and clear to stop Timer 3.
T3IEN	Timer 3 Interrupt Enable bit T3IEN=0 disables the Timer 3 overflow interrupt. T3IEN=1 enables the Timer 3 overflow interrupt.

TL3 (0xAEh) Timer 3 Low Byte Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	T3[7-0]							
WR	T3[7-0]							

TH3 (0xAFh) Timer 3 High Byte Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	T3[15-8]							
WR	T3[15-8]							

TL4 (0xACh) Timer 4 Low Byte Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	T4[7-0]							
WR	T4[7-0]							

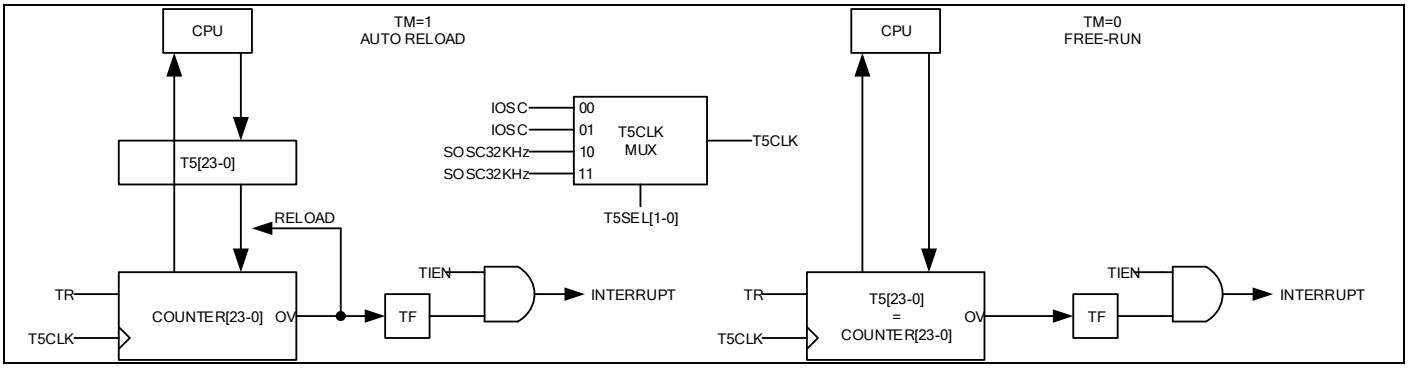
TH4 (0xADh) Timer 4 High Byte Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	T4[15-8]							
WR	T4[15-8]							

T3[15-0] and T4[15-0] function differently when being read or written. When written in auto-reload mode, its reload value register is written, and in free-run mode, the counter value is written immediately. When read, the return value is always the present counter value. There is no snapshot buffer in the read operation, so software should always read the high byte then the low byte.

1.12 System Timer – T5

T5 is a 24-Bit simple timer. It can select four different clock sources and can be used for extended sleep mode wake up. The clock sources include IOSC and SOSC32KHz. T5 can be configured either as free-run mode or auto-reload mode. Timer 5 does not depend on the SYSCLK; therefore, it continues to count under STOP or SLEEP mode if the clock source is present. The following diagram shows the block diagram of Timer 5.



T5CON (0xA068h) Timer 5 Control and Status Register

	7	6	5	4	3	2	1	0
RD	TF5	T5SEL[1]	T5SEL[0]	TM5	TR5	-	-	T5IEN
WR	TF5	T5SEL[1]	T5SEL[0]	TM5	TR5	-	-	T5IEN

TF5 Timer 5 Overflow Interrupt Flag bit
TF5 is set by hardware when overflow condition occurs. TF5 must be cleared by software.

T5SEL[1-0] Timer 5 Clock Selection bits
T5SEL[1-0] = 00, IOSC
T5SEL[1-0] = 01, IOSC
T5SEL[1-0] = 10, SOSC32KHz
T5SEL[1-0] = 11, SOSC32KHz

TM5 Timer 5 Mode Control bit. TM5=1 sets timer 5 as auto reload, and TM5=0 sets timer 5 as free-run.

TR5 Timer 5 Run Control bit. Set to enable Timer 5, and clear to stop Timer 5.

T5IEN Timer 5 Interrupt Enable bit.
T5IEN=0 disables the Timer 5 overflow interrupt.
T5IEN=1 enables the Timer 5 overflow interrupt.

TL5 (0xA069) Timer5 Low Byte Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	T5[7-0]							
WR	T5[7-0]							

TH5 (0xA06A) Timer5 Medium Byte Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	T5[15-8]							
WR	T5[15-8]]							

TT5 (0xA063) Timer5 High Byte Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	T5[23-16]							
WR	T5[23-16]							

T5[23-0] functions differently when being read or written. When written in auto-reload mode, its reload value register is written, and in free-run mode, the counter value is written immediately. When read, the return value is always the present counter value. There is no snapshot buffer in the read operation, so software should always read the high byte then the low byte.

1.13 Multiplication and Division Unit (MDU)

MDU provides acceleration on unsigned integer operations of 16-bit multiplications, 32-bit division, and shifting and normalizing operations. The following table shows the execution characteristics of these operations. The MDU does not contain the operation completion status flag. Therefore, the most efficient utilization of MDU uses NOP delay for the required clock time of the MDU operation types. The number of clock cycles required for each operation is shown in the following table and it is counted from the last write of the writing sequence.

Operations	Result	Reminder	# of Clock Cycle
32-bit division by 16-bit	32-bit	16-bit	17
16-bit division by 16-bit	16-bit	16-bit	9
16-bit multiplication by 16-bit	32-bit	-	10
32-bit normalization	-	-	3 – 20
32-bit shift left/right	-	-	3 – 18

The MDU is accessed through MD0 to MD5 which contains the operands and the results, and the operation is controlled by ARCON register.

ARCON (0xFF) MDU Control R/W 00000000

	7	6	5	4	3	2	1	0
RD	MDEF	MDOV	SLR	SC4	SC3	SC2	SC1	SC0
WR	MDEF	MDOV	SLR	SC4	SC3	SC2	SC1	SC0

MDEF — MDU Error Flag bit. Set by hardware to indicate MDx being written before the previous operation completes. MDEF is automatically cleared after reading ARCON.

MDOV — MDU Overflow Flag bit. MDOV is set by hardware if the dividend is zero or the result of multiplication is greater than 0x0000FFFFh

SLR — Shift Direction Control bit. SLR = 1 indicates a shift to the right and SLR =0 indicates a shift to the left.

SC4-0 — Shift Count Control and Result bit. If SC0-4 is written with 00000, the normalization operation is performed by MDU. When the normalization is completed, SC4-0 contains the number of shifts performed during the normalization. If SC4-0 is written with a non-zero value, then the shift operation is performed by MDU with the number of shifts specified by SC4-0 value.

MD0 (0xF9) MDU Data Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	MD0[7-0]							
WR	MD0[7-0]							

MD1 (0xFA) MDU Data Register 1 R/W 00000000

	7	6	5	4	3	2	1	0
RD	MD1[7-0]							
WR	MD1[7-0]							

MD2 (0xFB) MDU Data Register 2 R/W 00000000

	7	6	5	4	3	2	1	0
RD	MD2[7-0]							
WR	MD2[7-0]							

MD3 (0xFC) MDU Data Register 3 R/W 00000000

	7	6	5	4	3	2	1	0
RD	MD3[7-0]							
WR	MD3[7-0]							

MD4 (0xFD) MDU Data Register 4 R/W 00000000

	7	6	5	4	3	2	1	0
RD	MD4[7-0]							
WR	MD4[7-0]							

MD5 (0xFE) MDU Data Register 5 R/W 00000000

	7	6	5	4	3	2	1	0
RD	MD5[7-0]							
WR	MD5[7-0]							

MDU operation consists of three phases.

1. Loading MD0 to MD5 data registers in an appropriate order depending on the operation.
2. Execution of the operations.
3. Reading results from MD0 to MD5 registers.

The following list shows the MDU read and write sequences. Each operation has its unique writing sequence and reading sequence of MD0 to MD5 registers, therefore, a precise access sequence is required.

Division – 32-bit divide by 16-bit or 16-bit divide by 16-bit

Follow the following write-sequence. The first write of MD0 resets the MDU and initiates the MDU error flag mechanism. The last write incites calculation of MDU.

Write MD0 with Dividend LSB byte
Write MD1 with Dividend LSB+1 byte
Write MD2 with Dividend LSB+2 byte (ignore this step for 16-bit divide by 16-bit)
Write MD3 with Dividend MSB byte (ignore this step for 16-bit divide by 16-bit)
Write MD4 with Divisor LSB byte
Write MD5 with Divisor MSB byte

Then follow the following read-sequence. The last read prompts MDU for the next operations.

Read MD0 with Quotient LSB byte
Read MD1 with Quotient LSB+1 byte
Read MD2 with Quotient LSB+2 byte (ignore this step for 16-bit divide by 16-bit)
Read MD3 with Quotient MSB byte (ignore this step for 16-bit divide by 16-bit)
Read MD4 with Remainder LSB byte
Read MD5 with Remainder MSB byte
Read ARCON to determine error or overflow condition

Please note that if the sequence is violated, the calculation may be interrupted and result in errors.

Multiplication – 16-bit multiply by 16-bit

Follow the following write sequence.

Write MD0 with Multiplicand LSB byte
Write MD4 with Multiplier LSB byte
Write MD1 with Multiplicand MSB byte
Write MD5 with Multiplier MSB byte

Then follow the following read sequence.

Read MD0 with Product LSB byte
Read MD1 with Product LSB+1 byte
Read MD2 with Product LSB+2 byte
Read MD3 with Product MSB byte
Read ARCON to determine error or overflow condition

Normalization – 32-bit

Normalization is obtained with integer variables stored in MD0 to MD3. After normalization, all leading zeroes are removed by shift left operations. To start the normalization operation, SC4-0 in ARCON is first written with 00000. After completion of the normalization, SC4-0 is updated with the number of leading zeroes and the normalized result is restored on MD0 to MD3. The number of the shift of the normalization can be used as exponents. The following write sequence should be followed. The last write to ARCON initiates the normalization operations by MDU.

Write MD0 with Operand LSB byte
Write MD1 with Operand LSB+1 byte
Write MD2 with Operand LSB+2 byte
Write MD3 with Operand MSB byte
Write ARCON with SC4-0 = 00000

Then follow the following read sequence.

Read MD0 with Result LSB byte
Read MD1 with Result LSB+1 byte
Read MD2 with Result LSB+2 byte
Read MD3 with Result MSB byte
Read SC[4-0] from ARCON for normalization count or error flag

Shift – 32-bit

Shift is done with integer variables stored in MD0 to MD3. To start the shift operation, SC4-0 in ARCON is first written with shift count and SLR with shift direction. After completion of the Shift, the result is stored back to MD0 to MD3.The following write sequence should be followed. The last write to ARCON initiates the normalization operations by MDU.

Write MD0 with Operand LSB byte

Write MD1 with Operand LSB+1 byte

Write MD2 with Operand LSB+2 byte

Write MD3 with Operand MSB byte

Write ARCON with SC4-0 = Shift count and SLR with shift direction

Then follow the following read sequence.

Read MD0 with Result LSB byte

Read MD1 with Result LSB+1 byte

Read MD2 with Result LSB+2 byte

Read MD3 with Result MSB byte

Read ARCON's for error flag

MDU Flag

The error flag (MDEF) of MDU indicates improperly performed operations. The error mechanism starts at the first MD0 write and finishes with the last read of MD result register. MDEF is set if current operation is interrupted or restarted by improper write of MD register before the operation completes. MDEF is cleared if the operations and proper write/read sequences successfully complete. The overflow flag (MDOV) of MDU indicates an error of operations. MDOV is set if

The divisor is zero

Multiplication overflows

Normalization operation is performed on already normalized variables (MD3.7 =1)

1.14 Serial Port – UART0

UART0 is full duplex and fully compatible with the standard 8052 UART. The receive path of the UART0 is double-buffered that can commence reception of a second byte before previously received byte is read from the receive register. Writing to SBUF0 loads the transmit register while reading SBUF0, reads a physically separate receive register. The UART0 can operate in four modes: one synchronous (Mode 0) and three asynchronous modes (Mode 1, 2, and 3). Mode 2 and Mode 3 share a special provision for multi-processor communications. This feature is enabled by setting SM2 in SCON0 register. The master processor first sends out an address byte, which identifies the slave. An address byte differs from a data byte in the 9th bit: 1 defines an address byte, whereas 0 defines a data byte. When SM2 is set to 1, no slave can be interrupted by a data byte. The addressed slave clears its SM2 bit and prepares to receive the following incoming data bytes. The slaves that are not addressed leave their SM2 set and ignore the incoming data. The UART0-related registers are SBUF0, SCON0, PCON, IE, and IP.

SCON0 (0x98h) UART0 Configuration Register

	7	6	5	4	3	2	1	0
RD	SM0	SM1	SM2	REN	TB8	RB8	TIF	RIF
WR	SM0	SM1	SM2	REN	TB8	RB8	TIF	RIF

SM0, SM1 UART Operation Mode

MODE	SM0	SM1	Description
0	0	0	Synchronous Shift Register Mode Baud rate = SYSCLK/12
1	0	1	8-Bit UART Mode Baud rate = Timer 1 or Timer 2 overflow rate. This is selected in T2CON registers.
2	1	0	9-Bit UART Mode, fixed baud rate Baud rate = SYSCLK/64 (PCON.SMOD0 = 0) or SYSCLK/32 (PCON.SMOD0 = 1)
3	1	1	9-Bit UART Mode, variable baud rate

			Baud rate = Timer 1 or Timer 2 overflow rate. This is selected in TCON registers.

SM2	Set to enable a multiprocessor communication as a slave device.
REN	Set REN=1 to enable UART PMM switch back function. REN=0 disables this function. In PMM mode, if REN=1, then any transition on RX of UART triggers the exit of PMM mode into normal mode.
TB8	The transmit-value of 9th bit in 9-bit UART mode (mode 2 and mode 3). Set or cleared by CPU depending on the function of the 9th bit as a parity check bit or a multi-processor.
RB8	The receive-value of 9th bit in 9-bit UART mode (mode 2 and mode 3). Set or cleared by hardware.
TIF	Transmit Interrupt Flag bit. Set by hardware after completion of a serial transmission and must be cleared by software. The interrupt enable bit is located in IE (0xA8) and the interrupt priority is located in IP (0xB8).
RIF	Receive Interrupt Flag bit. Set by hardware after completion of a serial reception and must be cleared by software. The interrupt enable bit is located in IE (0xA8) and the interrupt priority is located in IP (0xB8).

SBUF0 (0x99h) UART0 Data Buffer Register

	7	6	5	4	3	2	1	0
RD	RB[7-0]							
WR	TB[7-0]							

SBUF0 is used for both transmission and reception. Writing a data byte into SBUF0 puts this data in UART0's transmit buffer and starts a transmission. Reading a byte from SBUF means data being read from the UART0's receive buffer.

Mode 0

Mode 0 is a simple synchronous shift register mode. TXD0 outputs the shift clock, which is fixed at CPUCLK/12. RXD0 is a bidirectional I/O port that serves as a data-shifting port. To utilize this mode, TXD0 pin must be enabled as an output pin, while RXD0 needs to be configured as an open-drain type of I/O port. The shift data changes at the rising edge of the shift clock and is valid at the falling edge of the shift clock. The transmission starts when a new byte is written in SBUF0 as TI is cleared to 0. When the byte is transmitted, TI is set and the UART0 waits for the next byte to be transmitted. The reception is initiated by setting REN=1 and RI cleared to 0. When a byte is received, RI is set by UART0.

Mode 1

8-bit UART mode. RXD0 is the serial input and TXD0 is the serial output. To utilize this mode, the corresponding RXD0 and TXD0 pin configuration should also be set correctly. 10-bit data (including a Start bit, 8 data bit, and a Stop bit) are transferred. For UART0, the baud rate is set by Timer 1 or Timer 2 overflow rate. The control is determined by SMOD0.PCON, and RCLK.T2CON, TCLK.T2CON. When SMOD0.PCON is 1, Timer 1 overflow is selected, and SMOD0.PCON is 0, Timer 1 overflow rate divided by 2 is selected. And if RCLK.T2CON, or TCLK.T2CON is set, the Timer 2 overflow rate is selected and overwrites the SMOD0 setting.

Mode 2

9-bit UART mode. RXD0 is the serial input and TXD0 is the serial output. To utilize this mode, the corresponding RXD0 and TXD0 pins should be configured correctly. 11-bit data including a Start bit (always 0), 8 data bits, a programmable 9th bit, and a Stop bit (always 1) are transferred. The 9th bit can be configured as a parity bit configured by software through TB8 in SCON0. The received 9th bit can be read from TB8. The software determines the correctness of the parity check. The baud rate in Mode 2 is fixed at 1/32 or 1/64 of CPU clock. This is controlled by SMOD0 in PCON register.

Mode 3

Like Mode 2 (9-bit UART mode). RXD0 is the serial input and TXD0 is the serial output. To utilize this mode, the corresponding RXD0 and TXD0 pins should also be configured properly. 11-bit data including a Start bit (always 0), 8 data bits, a programmable 9th bit, and a Stop bit (always 1) are transferred. The 9th bit can serve as a parity bit configured by software through TB8 in SCON0. The received 9th bit can be read from TB8. The software determines the correctness of the parity check. The mechanism of the baud rate control in Mode 3 is like that in Mode 1, which is determined by Timer 1 or Timer 2 overflow and is set by SMOD0, and T2CON.

1.15 I²C Master

The I²C master controller provides the interface to I²C slave devices. It can be programmed to operate with arbitration and clock synchronization to allow it to operate in multi-master configurations. The master uses SCL and SDA pins. The controller contains a built-in 8-bit timer to allow various I²C bus speeds. The maximum I²C master bus speed is limited to SYSCLK/12.

I2CMTP (0xF7h) I²C Master Time Period R/W 00000000

	7	6	5	4	3	2	1	0
RD	I2CMTP[7-0]							
WR	I2CMTP[7-0]							

This register set the frequency of I²C bus clock. If I2CMTP[7-0] is equal to or larger than 0x01, SCL_FREQ = SYSCLK_FREQ/8/(1 + I2CMTP). If I2CMTP[7-0] = 0x00, SCL_FREQ = SYSCLK_FREQ /12.

I2CMSA (0xF4) I²C Master Slave Address R/W 00000000

	7	6	5	4	3	2	1	0
RD	SA[6-0]							RS
WR	SA[6-0]							RS

SA[6-0] Slave Address. SA[6-0] defines the slave address the I²C master uses to communicate.

RS Receive/Send Bit. RS determines if the following operation is to RECEIVE (RS=1) or SEND (RS=0).

I2CMBUF (0xF6) I²C Master Data Buffer Register R/W 00000000

	7	6	5	4	3	2	1	0
RD	RD[7-0]							
WR	TD[7-0]							

I2CMBUF functions as a transmit-data register when written and as a receive-data register when read. When written, TDis sent to the bus by the next SEND or BURST SEND operations. TD[7] is sent first. When read, RD contains the 8-bit data received from the bus upon the last RECEIVE or BURST RECEIVE operation.

I2CMCR (0xF5) I²C Master Control and Status Register R/W 00000000

	7	6	5	4	3	2	1	0
RD	-	BUSBUSY	IDLE	ARBLOST	DATAACK	ADDRACK	ERROR	BUSY
WR	CLEAR	INFILEN	-	HS	ACK	STOP	START	RUN

The I2CMCR register is used for setting control when it is written, and as a status signal when read.

CLEAR Reset I2C Master State Machine
Set CLEAR=1 will reset the state machine. CLEAR is self-cleared when reset is completed.

INFILEN Input Noise Filter Enable. When IFILEN is set, pulses shorter than 50 nsec on inputs of SDA and SCL are filtered out.

IDLE This bit indicates that I²C master is in the IDLE mode.

BUSY This bit indicates that I²C master is receiving or transmitting data, and other status bits are not valid.

BUSBUSY This bit indicates that the external I²C bus is busy and access to the bus is not possible. This bit is set/reset by START and STOP conditions.

ERROR This bit indicates that an error occurs in the last operation. The errors include slave address is not acknowledged, or transmitted data is not acknowledged, or the master controller loses arbitration.

ADDRERR This bit is automatically set when the last operation slave address transmitted is not acknowledged.

DATAERR This bit is automatically set when the last operation transmitted data is not acknowledged.

ARBLOST This bit is automatically set when the last operation I²C master controller loses the bus arbitration.

START, STOP, RUN and HS, RS, ACK bits are used to drive I²C Master to initiate and terminate a transaction. The Start bit generates START, or REPEAT START protocol. The Stop bit determines if the cycle stops at the end of the data cycle or continues to burst. To generate a single read cycle, the designated address is written in SA, RS is set to

1, and bits ACK=0, STOP=1, START=1, RUN=1 are set in I2CMCR to perform the operation and then STOP. When the operation is completed (or aborted due to errors), I²C master generates an interrupt. The ACK bit must be set to 1. This causes the controller to send an ACK automatically after each byte transaction. The ACK bit must be reset when set to 0 when the master operates in receive mode and not to receive further data from the slave devices.

The following table lists the permitted control bits combinations in master IDLE mode.

HS	RS	ACK	STOP	START	RUN	OPERATIONS
0	0	-	0	1	1	START condition followed by SEND. Master remains in TRANSMITTER mode
0	0	-	1	1	1	START condition followed by SEND and STOP
0	1	0	0	1	1	START condition followed by RECEIVE operation with negative ACK. Master remains in RECEIVER mode
0	1	0	1	1	1	START condition followed by RECEIVE and STOP
0	1	1	0	1	1	START condition followed by RECEIVE. Master remains in RECEIVER mode
0	1	1	1	1	1	Illegal command
1	0	0	0	0	1	Master Code sending and switching to HS mode

The following table lists the permitted control bits combinations in master TRANSMITTER mode.

HS	RS	ACK	STOP	START	RUN	OPERATIONS
0	-	-	0	0	1	SEND operation. Master remains in TRANSMITTER mode
0	-	-	1	0	0	STOP condition
0	-	-	1	0	1	SEND followed by STOP condition
0	0	-	0	1	1	REPEAT START condition followed by SEND. Master remains in TRANSMITTER mode
0	1	-	1	1	1	REPEAT START condition followed by SEND and STOP condition
0	1	0	0	1	1	REPEAT START condition followed by RECEIVE operation with negative ACK. Master remains in TRANSMITTER mode
0	1	0	1	1	1	REPEAT START condition followed by SEND and STOP condition.
0	1	1	0	1	1	REPEAT START condition followed by RECEIVE. Master remains in RECEIVER mode.
0	1	1	1	1	1	Illegal command

The following table lists the permitted control bits combinations in master RECEIVER mode.

HS	RS	ACK	STOP	START	RUN	OPERATIONS
0	-	0	0	0	1	RECEIVE operation with negative ACK. Master remains in RECEIVE mode
0	-	-	1	0	0	STOP condition
0	-	0	1	0	1	RECEIVE followed by STOP condition
0	-	1	0	0	1	RECEIVE operation. Master remains in RECEIVER mode
0	-	1	1	0	1	Illegal command
0	1	0	0	1	1	REPEAT START condition followed by RECEIVE operation with negative ACK. Master remains in RECEIVER mode
0	1	0	1	1	1	REPEAT START condition followed by RECEIVE and STOP conditions
0	1	0	1	1	1	REPEAT START condition followed by RECEIVE. Master remains in RECEIVER mode
0	0	-	0	1	1	REPEAT START condition followed by SEND. Master remains in TRANSMITTER mode.

0	0	-	1	1	1	REPEAT START condition followed by SEND and STOP conditions

All other control-bit combinations not included in three tables above are NOP. In Master RECEIVER mode, STOP should be generated only after data negative ACK executed by Master or address negative ACK executed by slave. Negative ACK means SDA is pulled low when the acknowledge clock pulse is generated.

I2CMTO (0xC3) I²CTime Out Control Register R/W 00000000

	7	6	5	4	3	2	1	0
RD	I2CMTOF	I2CMTO[6-0]						
WR	I2CMTOEN	I2CMTO[6-0]						

I2CMTOEN I2CM Time Out Enable

I2CMTOF I2CM Time Out Flag
This bit is set when a timeout occurs. It is cleared when I2CM CLEAR command is issued.

I2CMTO[6-0] I2CM Time Out Setting
The TO time is set to (I2CMTO[6-0]+1)*8*BT. When a time out occurs, an I2CM interrupt will be generated.

1.16 Checksum/CRC Accelerator

To enhance the performance, a hardware Checksum/CRC Accelerator is included and closely coupled with CPU. This provides the most used checksum and CRC operation for 8/16/24/32-bit data width. For 8-bit data, one SYSCLK cycle is used. For 16-bit data, two SYSCLK cycles are used. For 32-bit data, four SYSCLK cycles are used.

CCCFG (0xA078h) Checksum/CRC Accelerator Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DWIDTH[1-0]		REVERSE	NOCARRY	SEED	-	-	BUSY
WR	DWIDTH[1-0]		REVERSE	NOCARRY	SEED	CRCMODE[2-0]		

DWIDTH[1-0] Data Input Width
00 – set input as 8-bit wide
01 – set input as 16-bit wide
10 – set the input as 24-bit wide
11 – set the input as 32-bit wide

REVERSE Reverse Input MSB/LSB Sequence
REVERSE=0 is for LSB first operations.
REVERSE=1 is for MSB first operation.
The reverse order is based on the data width. For example, if the data width is 32-bit, and REVERSE=1, then CCDATA[0] holds MSB, and CCDATA[31] holds LSB.
REVERSE=0 does not affect output result and SEED ordering i.e. CCDATA[31] always holds MSB, CCDATA[0] always holds LSB.
The following table shows the MSB/LSB relationship

DWIDTH	REVERSE=0	REVERSE=1
0	CRCIN[7-0] = CCDATA[7-0]	CRCIN[7-0] = CCDATA[0-7]
1	CRCIN[15-0] = CCDATA[15-0]	CRCIN[15-0] = CCDATA[0-15]
2	CRCIN[23-0] = CCDATA[23-0]	CRCIN[23-0] = CCDATA[0-23]
3	CRCIN[31-0] = CCDATA[31-0]	CRCIN[31-0] = CCDATA[0-31]

NOCARRY Carry Setting for Checksum
NOCARRY=0 uses the previous carry result for the new result.
NOCARRY=1 discard previous carry result.

SEED Seed Entry
SEED=1 results in writing into CCDATA as the SEED value.
SEED=0 for normal data inputs.
The MSB/LSB ordering of SEED entry from CCDATA is not affected by REVERSE.

CRCMODE[2-0] Defines CRC/Checksum Mode
000 – Accelerator is disabled and clock gated off
001 – 8-bit Checksum

010 – 32-bit Checksum
011 – CRC-16 (IBM 0x8005)
X16+X15+X2+1
100 – CRC-16 (CCITT 0x1021)
X16+X12+X5+1
101 – CRC-32 (ANSI 802.3 0x104C11DB7)
X32+X26+C23+X22+X16+X12+X11+X10+X8+X7+X5+X4+X2+X1+1
110 – Reserved
111 – CRC and Checksum Clear

The first step for the programmer is to set the CRCMODE[2-0] for the Checksum or CRC operation and then write "111" to CRCMODE[2-0] to reset the Checksum/CRC states and restore the default seed value (for checksum, seed value=0x00 or 0x00000000, for CRC seed value = 0xFFFF or 0xFFFFFFF).

BUSY CRC Status

BUSY=1 indicates the results is not yet completed. Since only up to two cycles are used to calculate the Checksum or CRC, there is no need to check BUSY status before the next data entry and reading the results.

CCDATA registers are the data I/O port for Checksum/CRC Accelerator. For 8-bit data width only CCDATA[7-0] should be used. For data width wider than 8-bit, high byte should always be written first, and writing the low byte (CCDATA0) completes the data entry and starts the calculations. When SEED=1, the data written goes to CS or CRC seed value. The SEED value entry bit ordering is not affected by REVERSE setting. The result of accelerator can be directly read out from CCDATA registers also not affected by REVERSE setting.

CCDATA0 (0xA07Ch) Checksum/CRC Data Register 0 R/W 00000000

	7	6	5	4	3	2	1	0
RD	CCDATA[7-0]							
WR	CCDATA[7-0]							

CCDATA1 (0xA07Dh) Checksum/CRC Data Register 1 R/W 00000000

	7	6	5	4	3	2	1	0
RD	CCDATA[15-0]							
WR	CCDATA[15-0]							

CCDATA2 (0xA07Eh) Checksum/CRC Data Register 2 R/W 00000000

	7	6	5	4	3	2	1	0
RD	CCDATA[23-16]							
WR	CCDATA[23-16]							

CCDATA3 (0xA07Fh) Checksum/CRC Data Register 3 R/W 00000000

	7	6	5	4	3	2	1	0
RD	CCDATA[31-24]							
WR	CCDATA[31-24]							

1.17 Break Point and Debug Controller

The CPU core also includes a Break Point Controller for software debugging purposes and handling exceptions. Program Counter break point triggers at PC address matching, and there are seven PC matching settings available. Single Step break point triggers at interaction return from an interrupt routine.

Upon the matching of break point conditions, the Break Point Controller issues BKP Interrupt for handling the break points. The BKP Interrupt vector is located at 0x7080. Upon entering the BKP ISR (Break Point Interrupt Service Routine), all interrupts and counters (WDT, T0, T1, and T2) are disabled. To allow further interrupts and continuing counting, the BKP ISR must be enabled. At exiting, the BKP ISR setting must be restored to resume normal operations.

BPINTF (A0E0h) Break Point Interrupt Flag Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	STEP_IF	-	-	-	-	-	PC2IF	PC1IF
WR	STEP_IF	-	-	-	-	-	PC2IF	PC1IF

This register is for reading the Break Points interrupt flags.

STEP_IF This bit is set when the Break Point conditions are met by a new instruction fetching from an interrupt routine. This bit must be cleared by software.

PC2IF – PC1IF These bits are set when Break Point conditions are met by PC2 – PC1 address. These bits must be cleared by software.

BPINTE (A0E1h) Break Point Interrupt Enable Register R/W (0x00) TB Protected

	7	6	5	4	3	2	1	0
RD	STEP_IE	-	-	-	-	-	PC2IE	PC1IE
WR	STEP_IE	-	-	-	-	-	PC2IE	PC1IE

This register controls the enabling of individual Break Points interrupt.

STEP_IE Set this bit to enable Single Step event break point interrupt.

PC2IE – PC1IE Set these bits to enable PC2 to PC1 address match break point interrupts.

BPINTC (A0E2h) Break Point Interrupt Control Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-	-	-	-	-	-	-	-
WR	-	-	-	-	-	-	-	-

This register is reserved for other applications.

BPCTRL (A0E3h) DBG and BKP ISR Control and Status Register R/W (0xFC)

	7	6	5	4	3	2	1	0
RD	DBGINTEN	DBGWDTEN	DBGT2EN	DBGT1EN	DBGT0EN	-	-	DBGGST
WR	DBGINTEN	DBGWDTEN	DBGT2EN	DBGT1EN	DBGT0EN	-	-	DBGGST

When entering the DBG or BKP ISR (Interrupt Service Routine), all interrupts and timers are disabled. The enabled bits are cleared by hardware reset in this register. As the interrupts and timers are disabled, the ISR can process debugging requirements in a suspended state. If a specific timer should be kept active, it must be enabled by ISR after ISR entry. Before the exit of DBG and BKP ISR, the control bits should be enabled to allow the timers to resume operating. This register should be modified only in Debug ISR.

DBGINTEN Set this bit to enable all interrupts (except WDT interrupt). This bit is cleared automatically at the entry of DBG and BKP ISR. Set this bit to allow ISR to be further interrupted by other interrupts. This is sometimes necessary if DBG or BKP ISR needs to use UART or I^2C, for example.

DBGWDEN Set this bit to allow WDT counting during the DBG and BKP ISR. This bit should always be set before exiting the ISR.

DBGT2EN Set this bit to allow T2 counting during the DBG and BKP ISR. This bit should always be set before exiting the ISR. This bit only controls the counting but not T2 interrupt.

DBGT1EN Set this bit to allow T1 counting during the DBG and BKP ISR. This bit should always be set before exiting the ISR. This bit only controls the counting but not T1 interrupt.

DBGT0EN Set this bit to allow T0 counting during the DBG and BKP ISR. This bit should always be set before exiting the ISR. This bit only controls the counting but not T0 interrupt.

DBGST This bit indicates the DBG and BKP ISR status. It is set to 1 when entering DBG and BKP ISR. It should be cleared when exiting the DBG and BKP ISR. Checking this bit allows other interrupt routines to determine whether it is a sub-service of the DBG and BKP ISR.

PC1AL (A0F0h) Program Counter Break Point 1 Low Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PC1AL[7-0]							
WR	PC1AL[7-0]							

This register defines the PC low address for PC match break point 1.

PC1AH (A0F1h) Program Counter Break Point 1 High Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PC1AH[7-0]							
WR	PC1AH[7-0]							

This register defines the PC high address for PC match break point 1.

PC1AT (A0F2h) Program Counter Break Point 1 Top Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PC1AT[7-0]							
WR	PC1AT[7-0]							

This register defines the PC top address for PC match break point 1. PC1AT:PC1HT:PC1LT together form a 24 bit compare value of break point 1 for Program Counter.

PC2AL (A0F4h) Program Counter Break Point 2 Low Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PC2AL[7-0]							
WR	PC2AL[7-0]							

This register defines the PC low address for PC match break point 2.

PC2AH (A0F5h) Program Counter Break Point 2 High Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PC2AH[7-0]							
WR	PC2AH[7-0]							

This register defines the PC high address for PC match break point 2.

PC2AT (A0F6h) Program Counter Break Point 2 Top Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PC2AT[7-0]							
WR	PC2AT[7-0]							

This register defines the PC top address for PC match break point 2. PC2AT:PC2HT:PC2LT together form a 24-bit compare value of PC break point 2 for Program Counter.

Host or program can obtain the status of the break point controller through the current break point address and next PC address register. DBPCID[23-0] contains the PC address of just executed instruction when the break point occurs. DBNXPC[23-0] contains the next PC address to be executed when the breakpoint occurs, therefore, it is usually exactly the same value as the break pointer setting.

DBPCIDL (A098h) Debug Program Counter Address Low Register RO (0x00)

	7	6	5	4	3	2	1	0
RD	DBPCID[7-0]							
WR	-							

DBPCIDH (A099h) Debug Program Counter Address High Register RO (0x00)

	7	6	5	4	3	2	1	0
RD	DBPCID[15-8]							
WR	-							

DBPCIDT (A09Ah) Debug Program Counter Address Top Register RO (0x00)

	7	6	5	4	3	2	1	0
RD	DBPCID[23-16]							
WR	-							

DBPCNXL (A09Bh) Debug Program Counter Next Address Low Register RO (0x00)

	7	6	5	4	3	2	1	0
RD	DBPCNX[7-0]							
WR	-							

DBPCNXH (A09Ch) Debug Program Counter Next Address High Register RO (0x00)

	7	6	5	4	3	2	1	0
RD	DBPCNX[15-8]							
WR	-							

DBPCNXT (A09Dh) Debug Program Counter Next Address Top Register RO (0x00)

	7	6	5	4	3	2	1	0
RD	DBPCNX[23-16]							
WR	-							

STEPCTRL (A09Eh) Single Step Control Enable Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	STEPCTRL[7-0]							
WR	STEPCTRL[7-0]							

To enable single step debugging, STEPCTRL must be written with value 0x96.

1.18 Debug I²C Port

The I²C Slave 2 (I2CS2) can be configured as the debug and ISP port. This is achieved by assigning a predefined debug ID for the I²CSlave address. When a host issues an I²C access to this special address, a DBG interrupt is generated. DBG Interrupt has the highest priority. The DBG interrupt vector is located at 0x70C0. DBG ISR is used to communicate with the host and is usually strongly associated with BKP ISR.

SI2CDBGID (A09Fh) Slave I²C Debug ID Register R/W (0x36) TB Protected

	7	6	5	4	3	2	1	0
RD	DBGSI2C2EN	SI2CDBGID[6:0]						
WR	DBGSI2C2EN	SI2CDBGID[6:0]						

DBGSI2C2EN DBGSI2C2EN=1 enables I2CS2 as debug port. When I2CS2 receives an access of I²C address matching SI2CDBGID[6:0], a debug interrupt is generated.

SI2CDBGID[6:0] Slave I²C ID address for debug function.

1.19 Data SRAM ECC Handling

The data SRAM (IRAM and XRAM) is configured as 2048 x 13-bit. An 8:5 ECC encoder and decoder are implemented to check the SRAM data. ECC check is through hardware and performed automatically. It can correct 1-bit error in each byte and detect 2-bit error in each byte. All generation and checking are done in hardware. It is strongly recommended all SRAM data should be initialized at power-on or after reset if ECC is enabled to avoid initial ECC error. If ECC encounters either an uncorrectable error, hardware will latch the address and triggers an interrupt. Software needs to examine the severity of data corruption and determine appropriate actions. Please also note, switching between ECC and non-ECC mode, all the data in SRAM will be corrupted, and thus, require re-initialization. It is strongly suggested keeping ECC enabled for best reliability as well as noise immunity.

DECCCFG (0xA02Dh) Data ECC Configuration Register R/W (0x80) TB Protected

	7	6	5	4	3	2	1	0
RD	DECCEN	-	DECCIEN2	DECCIEN1		-	DECCIF2	DECCIF1
WR	DECCEN	-	DECCIEN2	DECCIEN1		-	DECCIF2	DECCIF1

DECCEN Data ECC Enable
DECCIEN2 Data ECC Uncorrectable Error Interrupt Enable
DECCIEN1 Data ECC Correctable Error Interrupt Enable
DECCIF2 Data ECC Uncorrectable Error Interrupt Flag
DECCIF2 is set to 1 by hardware when reading SRAM encounters uncorrectable error.
DECCIF2 is set independent of DECCIEN2. DECCIF2 needs to be cleared by software.
DECCIF1 Data ECC Correctable Error Interrupt Flag
DECCIF1 is set to 1 by hardware when reading SRAM encounters correctable error.
DECCIF1 is set independent of DECCIEN2. DECCIF2 needs to be cleared by software.

If a correctable error is encountered, the data will be automatically corrected. To prevent further corruption, software upon DECIF1 interrupt should rewrite the data into the SRAM.

DECCADL (0xA02Eh) Data ECC Configuration and Address Register Low RO (0x00)

	7	6	5	4	3	2	1	0
RD	DECCAD[7-0]							
WR	-							

DECCADH (0xA02Fh) Data ECC Configuration and Address Register High R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DECCAD[15-8]							
WR	-							

DECCAD[15-0] records the address of ECC fault when data SRAM ECC error occurs. It is read-only and reflects the error address that causes DECCIF to be set. If DECCIF is set and not cleared, DECCAD will not be updated if further error is detected.

1.20 Program ECC Handling

The program code stored in e-Flash has built-in ECC checking. The e-Flash is in 16-bit width, and when read by CPU program space accesses, the lower LSB 8-bit is read for instruction and the upper MSB 8-bit contains the ECC value of the LSB 8-bit. The ECC is nibble based, [15-12] is ECC for data [7-4], and [11-8] is ECC for data [3-0]. Four bits ECC for four bits data allows one bit error correction and two bits error detection. This means for an 8-bit code stored, 2-bit error correction is possible, and this greatly increases the reliability of the overall program robustness.

During program fetch and execution, ECC is performed simultaneously by hardware. If any ECC correctable error is detected, the value fetched is corrected, and optionally a PECCIEN1 interrupt can be generated. If any ECC non-correctable error is detected, two options can be configured, either a PECCIEN2 interrupt can be generated, or software reset can be generated. In both PECCIEN interrupt, the address of the error encountered is latched in PECCADL[15-0].

PECCCFG (0xA00Dh) Program ECC Configuration Register R/W (0x80) TB Protected

	7	6	5	4	3	2	1	0
RD	-	-	PECCIEN2	PECCIEN1		-	PECCIF2	PECCIF1
WR	-	-	PECCIEN2	PECCIEN1		-	PECCIF2	PECCIF1

PECCIEN2 Program ECC Uncorrectable Error Interrupt Enable
PECCIEN1 Program ECC Correctable Error Interrupt Enable
PECCIF2 Program ECC Uncorrectable Error Interrupt Flag
PECCIF2 is set to 1 by hardware when program fetching from e-Flash encounters uncorrectable error. PECCIF2 is set independent of PECCIEN2. PECCIF2 needs to be cleared by software.
PECCIF1 Program ECC Correctable Error Interrupt Flag

PECCIF1 is set to 1 by hardware when program fetching from e-Flash encounters correctable error. PECCIF1 is set independent of PECCIEN1 and PECCIF1 needs to be cleared by software.

PECCADL (0xA00Eh) Program ECC Fault Address Register Low RO (0x00)

	7	6	5	4	3	2	1	0
RD	PECCAD[7-0]							
WR	-							

PECCADLH(0xA00Fh) Program ECC Fault Address Register High R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PECCAD[15-8]							
WR	-							

PECCAD[15-0] records the address of ECC fault when Flash ECC error occurs. It is read-only and reflects the last error address.

1.21 Memory and Logic BIST Test

BSTCMD (0xA016h) SRAM Built-In and Logic Self Test R/W (0x00) TB Protected

	7	6	5	4	3	2	1	0
RD	MODE[3-0]				BST	-	FAIL	FINISH
WR	MODE[3-0]				BSTCMD[3-0]			

MODE[3-0] BIST Mode Selection
0000 – Normal Mode
0001 – SRAM MBIST
0010 – Reserved
0011 – Reserved
0100 – Register LBIST
0101 – Reserved
0110 – Reserved
0111 – Reserved
1000 – Normal Mode
1001 – SRAM MBIST and monitor on pins
1010 – Reserved
1011 – Reserved
1100 – Register LBIST and monitor on pins
1101 – Reserved
1110 – Reserved
1111 – Reserved
MODE[3-0] is cleared only by POR and RSTN. Software can read this setting along with the Pass/Fail status to determine which BIST was performed and its result even after a software reset.

BST BIST Status
BST is set to 1 by hardware when BIST in ongoing.

FAIL BIST Test Fail Flag
FAIL is set to 1 by hardware when BIST error has occurred. FAIL is cleared to 0 by hardware when a new BIST command is issued.

FINISH BIST Completion Flag
FINISH is set to 1 by hardware when BIST controller finishes the test. FINISH is cleared to 0 by hardware when a new BIST command is issued.

BSTCMD[3-0] Memory BIST Command
Writing BSTCMD[3-0] with value 4b'0101 causes the BIST controller to perform BIST.
Writing BSTCMD[3-0] with value 4b'1010 causes the BIST controller to perform BIST, and after BIST is completed, it automatically generates a software reset.
Writing BSTCMD[3-0] with value 4b'0000 causes FAIL and FINISH bits to be cleared to 0.

Any other value will either have no effect or abort any ongoing BIST.

After the BSTCMD is issued, CPU is paused until BIST is completed. And any BIST operations will result in the state of CPU in undefined states, and the content of the SRAM undefined. Therefore, it is highly recommended that a software reset or initiation should be performed after any BIST operation. Please also note that MODE[3-0], FINISH, FAIL bits are not cleared by software resets.

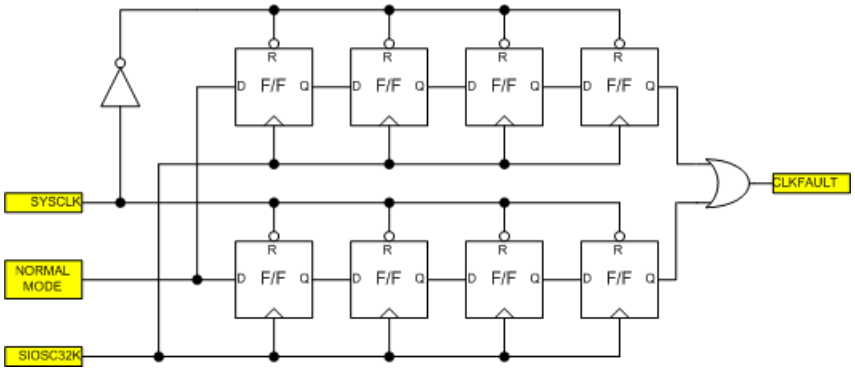
TSTMON (0xA014h) Test Monitor Flag R/W (0x00) TB Protected

	7	6	5	4	3	2	1	0
RD	TSTMON[7-0]							
WR	TSTMON[7-0]							

TSTMON register stores temporary status and is initialized by power-on reset only.

1.22 System Clock Monitoring

SYSCLK in normal running mode is monitored by SOSC32KHz (32KHz). If SYSCLK is not present in normal mode for four SOSC32KHz cycles, a hardware reset is triggered.

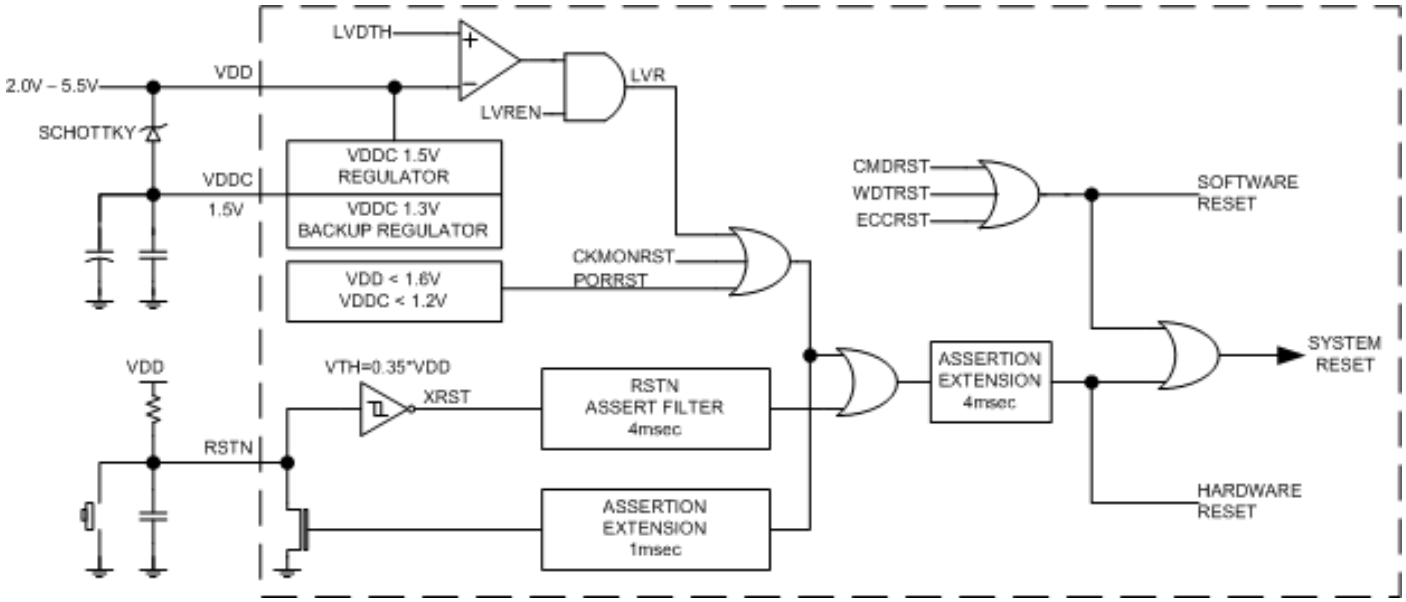


The clock monitoring is default turned off after reset.

1.23 Reset

There are several reset sources and includes both software resets and hardware resets. Software resets include command reset, WDT reset and ECC error reset. Hardware resets include power-on reset (low voltage detect on VDDC), LVD reset (low voltage detect on VDD), SYSCLK monitor reset, and external RSTN reset. Software reset only restores some registers to default values, and hardware reset restores all registers to their default values.

External RSTN reset is filtered so that low going glitches on RSTN with less than 4msec duration are ignored. All other hardware resets, once conditions are met, will be extended by 4 msec when exiting reset. The reset scheme described above is shown in the following diagram.



RSTCMD (0xA017h) Reset Command Register R/W 0x00 TB Protected

	7	6	5	4	3	2	1	0
RD	RSTCKM	RSTECC	-	-	CKMRF	ECCRF	WDTRF	CMDRF
WR	RSTCKM	RSTECC	-	CLRF	RSTCMD[3-0]			

RSTCKM — Reset Enable for Clock Monitor Fault
RENCKM=1 enables reset after clock fault detection. RSTCKM is cleared to 0 after any reset. Default RSTCKM is 0.

RSTECC — Reset Enable for Uncorrectable Code Fetch ECC Error
RSTECC=1 enables reset at e-Flash code fetch ECC error. Default RSTECC is 0.

CKMRF — Clock Monitor Fault Reset Flag
CKMRF is set to 1 by hardware when a clock fault reset has occurred. CKMRF is not cleared by reset except power-on reset.

ECCRF — ECC Error Reset Flag
ECCRF is set to 1 by hardware when an ECC error reset has occurred. ECCRF is cleared to 0 when writing CLRF=0. ECCRF is not cleared by reset except power-on reset.

WDTRF — WDT Reset Flag
WDTRF is set to 1 by hardware when WTRF, WT1RF, or WT2RF is set.

CLRF — Clear Reset Flag
Writing 1 to CLRF will clear CKMRF, ECCRF, WDTRF, and CMDRF. It is self-cleared.

RSTCMD[3-0] — Software Reset Command
Writing RSTCMD[3-0] with consecutive 4b'0101, 4b'1010 sequences will cause a software reset. Any other value will clear the sequence state. These bits are write-only and self-cleared.

Note: Bit 7 RSTCKM and bit 6 RSTECC can't be read.

2. Flash Controller

The flash controller connects the CPU to the on-chip embedded FLASH memory. The FLASH memory functions as the program storage as well as non-volatile data storage. The program access of FLASH does not require any special attention. When an ECC error during program fetch occurs, it causes ECC interrupt or reset.

When FLASH is used as data storage, the software issues commands to the FLASH controller through the XFR registers. And when the FLASH controller processes these commands, CPU is held idle until the command is completed. There is a time-out mechanism for holding CPU in idle to prevent operations hang up.

From FLASH controller point of view, the embedded Flash is always in 16-bit width with no distinction between ECC and data information. For code storage through FLASH controller, ECC byte (upper MSB 8-bit) must be acquired by software access of register FLSHECC. During the read command, ECC is detected but not corrected, the raw content is loaded into FLSHDAT[15-0]. If an ECC error is detected, FAIL status is set after the read command execution.

The e-Flash contains 64 pages (also referred to as Sector), and each page is 512x16. It also contains two IFB (Information Blocks) pages. In Flash operation, the erase command only operates on a page base.

FLSHCMD (A025h) Flash Controller Command Register R/W (0x80) TB Protected

	7	6	5	4	3	2	1	0
RD	WRVFY	BUSY	FAIL	CMD4	CMD3	CMD2	CMD1	CMD0
WR	CYC[2-0]			CMD4	CMD3	CMD2	CMD1	CMD0

WRVFY — Write Result Verify. At the end of a write cycle, hardware reads back the data and compares it with which should be written to the flash. If there is a mismatch, this bit becomes 0. It is reset to 1 by hardware when another ISP command is executed.

BUSY — Flash command is in processing. This bit indicates that Flash Controller is executing the Flash Read, Write, or Sector Erase and other commands are not valid.

FAIL — Command Execution Result. It is set if the previous command execution fails due to any reason. It is recommended that the program should verify the command execution after issuing a command to the Flash controller. It is not cleared by reading, but it will be cleared when a new command is issued. Possible causes of FAIL include address out of range, address falling into the protected region, and ECC error for read.

CYC[2-0] — Flash Command Time Out

CYC[2-0] defines command time out cycle count. The cycle period is defined by ISPCLK, which is SYSCLK/256/(ISPCLKF[7-0]+1). The number of cycles is tabulated as follows.

CYC[2-0]			WRITE	ERASE
0	0	0	55	5435
0	0	1	60	5953
0	1	0	65	6452
0	1	1	69	6897
1	0	0	75	7408
1	0	1	80	7906
1	1	0	85	8404
1	0	0	89	8889

For normal operations, CYC[2-0] should be set to 111.

CMD4 – CMD0 — Flash Command

These bits define commands for the Flash controller. The valid commands are listed in the following table. Any invalid commands do not get executed but return with a Fail bit.

CMD4	CMD3	CMD2	CMD1	CMD0	COMMAND
1	0	0	0	0	Main Memory Read
0	1	0	0	0	Main Memory Sector Erase
0	0	1	0	0	Main Memory Write
0	0	0	1	0	IFB Read
0	0	0	0	1	IFB Write
0	0	0	1	1	IFB Sector Erase
1	0	0	1	0	-

IFB1 contains manufacture data and user OTP, and therefore, IFB write commands are limited to IFB1 (0x0040-0x01FF) and IFB2. IFB Sector Erase is limited to IFB2.
For READ operations, FLSHDATH is the raw data, which is ECC code and FLSHDATL is ECC corrected data. If there is an ECC error, the FAIL status will be set, and corresponding ECC flags, PECCIF1 or PECCIF2 will be set according to the error condition.

FLSHDATL (A020h) Flash Controller Data Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	Flash Read Data Register DATA[7-0]							
WR	Flash Write Data Register DATA[7-0]							

FLSHDATH (A021h) Flash Controller Data Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	Flash Read Data Register DATA[15-8]							
WR	Flash Write Data Register DATA[15-8]							

FLSHADL (A022h) Flash Controller Low Address Data Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	Flash Address Low Byte Register ADDR[7-0]							
WR	Flash Address Low Byte Register ADDR[7-0]							

FLSHADH (A023h) Flash Controller High Address Data Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	Flash Address High Byte Register ADDR[15-8]							
WR	Flash Address High Byte Register ADDR[15-8]							

FLSHECC (A024h) Flash ECC Accelerator Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	ECC[7-0]							
WR	DATA[7-0]							

FLSHECC aids the calculation of ECC value of an arbitrary 8-bit data. The data is written to FLSHECC, and its corresponding ECC value can be read out from ECC.

ISPCLKF (A026h) Flash Command Clock Scaler R/W (0x25)

	7	6	5	4	3	2	1	0
RD	ISPCLKF[7-0]							
WR	ISPCLKF[7-0]							

ISPCLKF[7-0] configures the clock time base for generation of Flash erase and write timing. ISPCLK = SYSCLK * (ISPCLKF[7-0]+1)/256. For correct timing, ISPCLK should be set to approximately 2MHz.

FLSHPRT0 (A030h) Flash Controller Zone Protection Register 0 R/W (0xFF)

	7	6	5	4	3	2	1	0
RD	FLSHPRT[7-0]							
WR	FLSHPRT[7-0]							

FLSHPRT1 (A031h) Flash Controller Zone Protection Register 1 R/W (0xFF)

	7	6	5	4	3	2	1	0
RD	FLSHPRT[15-8]							
WR	FLSHPRT[15-8]							

FLSHPRT2 (A032h) Flash Controller Zone Protection Register 2 R/W (0xFF)

	7	6	5	4	3	2	1	0
RD	FLSHPRT[23-16]							
WR	FLSHPRT[23-16]							

FLSHPRT3 (A033h) Flash Controller Zone Protection Register 3 R/W (0xFF)

	7	6	5	4	3	2	1	0
RD	FLSHPRT[31-24]							
WR	FLSHPRT[31-24]							

FLSHPRT4 (A034h) Flash Controller Zone Protection Register 4 R/W (0xFF)

	7	6	5	4	3	2	1	0
RD	FLSHPRT[39-32]							
WR	FLSHPRT[39-32]							

FLSHPRT5 (A035h) Flash Controller Zone Protection Register 5 R/W (0xFF)

	7	6	5	4	3	2	1	0
RD	FLSHPRT[47-40]							
WR	FLSHPRT[47-40]							

FLSHPRT6 (A036h) Flash Controller Zone Protection Register 6 R/W (0xFF)

	7	6	5	4	3	2	1	0
RD	FLSHPRT[55-48]							
WR	FLSHPRT[55-48]							

FLSHPRT7 (A037h) Flash Controller Zone Protection Register 7 R/W (0xFF)

	7	6	5	4	3	2	1	0
RD	FLSHPRT[63-56]							
WR	FLSHPRT[63-56]							

FLSHPRT partitions the total code space of 64K into 64 uniform 1K zones for protection. If the corresponding bit in the FLSHPRT is 0, the zone protection is on. All bits in FLSHPRT are set to 1 by any reset. A “1” state corresponds to unprotected state. A bit can only be written to “0” by software and cannot be set to “1”. When a bit is “0”, the protection is on and disallows erasure or modifications. For content reliability, the user program should turn off the corresponding access after initialization as soon as possible.

FLSHPRT[31] Flash Zone Protect 31
This bit protects area 0x7C00 – 0x7FFF
FLSHPRT[30] Flash Zone Protect 30
This bit protects area 0x7800 – 0x7BFF
… …
FLSHPRT[4] Flash Zone Protect 4
This bit protects area 0x1000 – 0x13FF
FLSHPRT[3] Flash Zone Protect 3
This bit protects area 0x0C00 – 0x0FFF
FLSHPRT[2] Flash Zone Protect 2
This bit protects area 0x0800 – 0x0BFF
FLSHPRT[1] Flash Zone Protect 1
This bit protects area 0x0400 – 0x07FF
FLSHPRT[0] Flash Zone Protect 0
This bit protects area 0x0000 – 0x03FF

Since there is only 32K code Flash, only FLSHPRT[31-0] is used.

FLSHPRTC (A027h) Flash Controller Code Protection Register R/W 0x(00) TB Protected

	7	6	5	4	3	2	1	0
RD	-							STAT
WR	FLSHPRTC[7-0]							

This register further protects the code space (0x0000 – 0xFFFF). The protection is on after any reset. Software write of “55” into this register turns off protection. However, protection is maintained until a wait time (approximately 300msec) has expired. The 300msec delay prevents any false action due to power or interface transient. Any write other than “55” will turn on the protection immediately. STAT indicates the protection, and STAT=1 indicates the

protection is off, and STAT=0 indicates the protection is on.

To modify or erase the flash (not including IFB) both FLSHPRT and FLSHPRTC conditions need to be satisfied at the same time. IFB1's manufacturing data is always protected while user data can only be written "0". IFB2 are user application data and not protected.

3. I²C Slave Controller 1 (I2CS1)

The I²C Slave Controller 1 is a regular I²C Slave controller with enhanced functions such as clock-stretching and programmable hold time. These enhancements provide significant improvement in compatibilities. I2CS1 shares the SCL/SDA pins with the I2CM1. I2CS1 can also be configured to respond to two I²C addresses – I2CADR1 and I2CADR3. These two addresses can be enabled separately.

In receive mode, the controller detects a valid matching address and issues an ADDRMI interrupt. At the same time, the data bit on SDA line is shifted into the receive buffer. The RCBI interrupt is generated whenever a complete byte is received and is ready to be read from I2CSDAT. If for any reason, the software does not respond to RCBI interrupt in time (i.e., RCBI is not cleared), and a new byte is received, the controller either forces a NACK response on I²C (if CLKSTREN bit is not set) or by pulling and holding SDA low (if CLKSTREN bit is set) to stretch the SCL low duration to force the master into a wait state. In clock stretching mode, SCL is released when the software responds to RCBI interrupt and clears RCBI flag.

In transmit mode, the controller detects a valid matching address and issues an ADDRMI interrupt. At the same time, the data preloaded in the transmit data register through I2CSDAT is transferred to the transmit shift register and is serially shifted out onto SDA line. When this occurs, the controller generates a TXBI interrupt to inform the software that a new byte can be written into I2CSDAT. When the shift register is empty and ready for the next transmit, the slave controller checks if the new byte is written to the I2CSDAT. If TXBI is not cleared, it indicates a lack of new data and the slave controller holds SCL line low to stretch the current clock cycle if CLKSTREN is set. If the clock stretching is not enabled, the slave controller takes the old byte into the shift register and replies with NACK, and thus, causing data corruption. On the other hand, if the master returns the NACK after the byte transfer, this indicates the end of data to the I²C slave. In this case, the I²C slave releases the data line to allow the master to generate a STOP or REPEAT START.

The I²C slave controller also implements the input noise spike filter, and this is enabled by INFILEN bit in the I2CSCON register. The filter is implemented using digital circuit. When INFILEN is set, the spikes less than 1/2 SYSCLK period on the input of SDA and SCL lines are filtered out. If INFILEN is low, no input filtering is done. The following registers are related to I²C Slave Controller. The I²C slave controller uses SYSCLK to sample the SCL and SDA signals, therefore, the maximum allowable I²C bus speed is limited to SYSCLK/8 with conforming data setup and hold times. If setup and hold time cannot be guaranteed, then it is recommended the bus speed is limited to 1/40 SYSCLK.

I2CSCON1A (0xEB) I2CS1 Configuration Register A R/W (0x00)

	7	6	5	4	3	2	1	0
RD	EADRWK	EADDRMI	ESTOPI	ERPSTARTI	ETXBI	ERCBI	CLKSTREN	EACKWK
WR	EADRWK	EADDRMI	ESTOPI	ERPSTARTI	ETXBI	ERCBI	CLKSTREN	EACKWK

EADRWK	Enable Address matched wakeup from SLEEP mode.
EADDRMI	ADDRMI Interrupt Enable bit Set this bit to set ADDRMI interrupt as the I²C slave interrupt. This interrupt is generated when I²C slave receives a matching address.
ESTOPI	STOPI Interrupt Enable bit Set this bit to set STOPI interrupt as the I²C slave interrupt.
ERPSTARTI	RPTSTARTI Interrupt Enable Bit. Set this bit to set RPTSTARTI interrupt as the I²C slave interrupt.
ETXBI	TXBI Interrupt Enable bit Set this bit to allow TXBI interrupt as the I²C slave interrupt.
ERCBI	RCBI Interrupt Enable bit Set this bit to allow RCBI interrupt as the I²C slave interrupt.
CLKSTREN	Clock Stretching Enable bit Set to enable the clock stretching function of the slave controller. Clock stretching is an optional feature defined in I²C specification. If the clock stretching option is enabled (for slave I²C), the data written into transmit buffer is shifted out only after the occurrence of clock stretching, and the data cannot be loaded to transmit shift register. The programmer must write the same data again to the transmit buffer.
INFILEN	Input Noise Filter Enable bit. Set this bit to enable the input noise filter of SDA and SCL lines. When the filter is enabled, it filters out the spike of less than 50nsec.
START	Start Condition.

This bit is set when the slave controller detects a START condition on the SCL and SDA lines. This bit is not particularly useful as the start of transaction can be indicated by address match interrupt. This read-only bit is cleared when STOP condition is detected.

EACKWK
1: Enable clock stretching during system wakeup from sleep and wait until system wakeup completed and asks controller to send ACK to master.
0: Controller sends NACK when address is matched.

I2CSCON1B (0xAB) I2CS1 Configuration Register B R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-	SADR3M	XMT	START	SDAFLT[1-0]		GDFLT[1-0]	
WR	I2CSRST	-	-	-	SDAFLT[1-0]		GDFLT[1-0]	

I2CSRST — I^2C Slave Reset bit
Set this bit causes the Slave Controller to reset all internal state machine. It is self-cleared by hardware.

SDAFLT[1-0] — Delay for SDA input to satisfy SDA to SCL hold time
00 - 20ns RC filter delay
01 - 15ns RC filter delay
10 - 10ns RC filter delay
11 - 5ns RC filter delay

GDFLT [1:0] — Glitch filter for SCL and SDA input
00 - 20ns RC filter delay
01 - 15ns RC filter delay
10 - 10ns RC filter delay
11 - 5ns RC filter delay

SARD3M — Slave Address Match Flag bit. This bit is meaningful only when ADDRMI is set.
SARD3M=0 indicates the received I^2C address matches with I2CSADR1.
SARD3M=1 indicates the received I^2C address matches with I2CSADR3.
This bit is cleared when ADDRMI is cleared.

XMT — This bit is set by the controller when the I^2C slave is in transmit operation; this bit is cleared when the I^2C slave controller is in receive operation.

START — Start Condition.
This bit is set when the slave controller detects a START condition on the SCL and SDA lines. This bit is not particularly useful as the start of transaction can be indicated by address match interrupt. This read-only bit is cleared when STOP condition is detected.

I2CSST1 (0xEC) I2CSA1 Status Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	ADRWKF	ADDRMI	STOPI	RPSTARTI	TXBI	RCBI	FIRSTBT	NACK
WR	CLRWKF	CLRADMI	CLRSTOPI	CLRRPSTI				CLRNACK

CLRWKF — Clear Address Matched Wakeup Flag (ADRWKF)

ADRWKF — Address Matched Wakeup Flag

ADDRMI — Slave Address Matched Interrupt Flag bit
This bit is set when the received address matches the address defined in I2CSADR1. If EADDMI is set, this generates an interrupt. This bit must be cleared by software.

STOPI — Stop Condition Interrupt Flag bit
This bit is set when the slave controller detects a STOP condition on the SCL and SDA lines. This bit must be cleared by software.

RPTSARTI — Repeat Start Condition Interrupt Flag bit
This bit is set when the slave controller detects a REPEAT START condition on the SCL and SDA lines. This bit must be cleared by software.

TXBI — Transmit Buffer Interrupt Flag
This bit is set when the slave controller is ready to accept a new byte for transmit. This bit is cleared when new data is written into I2CSDAT register.

RCBI — Receiver Buffer Interrupt Flag bit
This bit is set when the slave controller puts new data in the I2CSDAT and ready for software-reading. This bit is cleared after the software reads I2CSDAT.

FIRSTBT	This bit is set to indicate the data in the data register as the first byte received after address match. This bit is cleared after the second byte is received. The bit is read only and generated by the slave controller.
NACK	NACK Condition bit This bit is set when the host responds with NACK in the byte transaction. This bit is only meaningful for slave-transmit operation. If the master returns with NACK on the byte transaction, the slave does not upload new data into the shift register. And the slave transmits the old data again as the next transfer, and this re-transmission continues if NACK is repeated until the transmission is successful and returned with ACK. This bit is cleared when a new ACK is detected, or it can be cleared by software.

I2CSADR1 (0xED) I2CS1 Slave Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	I2CSEN1	I2CADDR[6-0]						
WR	I2CSEN1	ADDR1[6-0]						

I2CSEN1	Set this bit to enable the I²C slave controller and ADDR1[6-0] for address matching.
ADDR1[6-0]	7-bit slave address 1
I2CADDR[6-0]	Received slave I2C address

I2CSDAT1 (0xEE) I2CS1 Data Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	I²C Slave Receive Data Register							
WR	I²C Slave Transmit Data Register							

I2CSADR3 (0x9E) I2CS1 2nd Slave Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	I2CSEN2	ADDR2[6-0]						
WR	I2CSEN	ADDR2[6-0]						

I2CSEN2	Set this bit to enable the I²C slave controller and ADDR2[6-0] for address matching. Please note that this can coexist with ADDR1.
ADDR2[6-0]	7-bit slave address 2.

4. I²C Slave Controller 2 (I2CS2)

The I²C Slave Controller 2 has dual functions – as a debug port for communication with host or as a regular I²C slave port. Both functions can coexist. I²C Slave 2 controller also supports the clock stretching functions.

The debug accessed by the host is through I²C slave address defined by SI2CSDBGID register and enabled by DBGSI2C2EN=1. When I2CS2 received this address match, a DBG interrupt is generated. This is described in the Debug and ISP sections. If DBGSI2C2EN=0, then I2CS2 functions as a regular I²C slave. The address of the slave is set by I2CSADR2 register. The MSB in I2CSADDR2 is the enable bit for the I²C slave controller and I2CSADR2[6-0] specifies the actual slave address.

In receive mode, the controller detects a valid matching address and issues an ADDRMI interrupt. At the same time, the data bit on SDA line is shifted into receive buffer. The RCBI interrupt is generated whenever a complete byte is received and is ready to be read from I2CSDAT. If for any reason, the software does not respond to RCBI interrupt in time (i.e., RCBI is not cleared), and a new byte is received, the controller either forces an NACK response on I²C (if CLKSTREN bit is not set) or by pulling and holding SDA low (if CLKSTREN bit is set) to stretch the SCL low duration to force the master into a wait state. In clock stretching mode, SCL is released when the software responds to RCBI interrupt and clears RCBI flag.

In transmit mode, the controller detects a valid matching address and issues an ADDRMI interrupt. At the same time, the data preloaded in the transmit data register through I2CSDAT is transferred to the transmit shift register and is serially shifted out onto SDA line. When this occurs, the controller generates a TXBI interrupt to inform the software that a new byte can be written into I2CSDAT. When the shift register is empty and ready for the next transmit, the slave controller checks if the new byte is written to the I2CSDAT. If TXBI is not cleared, it indicates lack of new data and the slave controller holds SCL line low to stretch the current clock cycle if CLKSTREN is set. If the clock stretching is not enabled, the slave controller takes the old byte into the shift register and replies with NACK, and thus, causes data corruption. On the other hand, if the master returns the NACK after the byte transfer, this indicates the end of data to the I²C slave. In this case, the I²C slave releases the data line to allow the master to generate a STOP or REPEAT START.

The I²C slave controller also implements the input noise spike filter, and this is enabled by INFILEN bit in the I2CSCON register. The filter is implemented using digital circuit. When INFILEN is set, the spikes less than 1/2 SYSCLK period on the input of SDA and SCL lines are filtered out. If INFILEN is low, no input filtering is done. The following registers are related to I²C Slave Controller. Also, the I²C slave controller uses SYSCLK to sample the SCL and SDA signals, and therefore, the maximum allowable I²C bus speed is limited to SYSCLK/8 with conforming data setup and hold times. If setup and hold time cannot be guaranteed, then it is recommended the bus speed is limited to 1/40 SYSCLK.

I2CSCON2 (0xDB) I2CS2 Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-	-	-	-	-	-	-	XMT
WR	I2CSRST	EADDRMI	ESTOPI	ERPSTARTI	ETXBI	ERCBI	CLKSTREN	INFILEN

I2CSRST	I²C Slave Reset bit Setting this bit causes the Slave Controller to reset all internal state machines. Clear this bit for normal operations. Setting this bit clears the I2CSADR2 (I²C slave address x).
EADDRMI	ADDRMI Interrupt Enable bit Set this bit to set ADDRMI interrupt as the I²C slave interrupt. This interrupt is generated when I²C slave receives a matching address.
ESTOPI	STOPI Interrupt Enable bit Set this bit to set STOPI interrupt as the I²C slave interrupt.
ERPSTARTI	RPSTARTI Interrupt Enable Bit Set this bit to set RPSTARTI interrupt as the I²C slave interrupt.
ETXBI	TXBI Interrupt Enable bit. Set this bit to allow TXBI interrupt as the I²C slave interrupt.
ERCBI	RCBI Interrupt Enable bit. Set this bit to allow RCBI interrupt as the I²C slave interrupt.
CLKSTREN	Clock Stretching Enable bit. Set to enable the clock stretching function of the slave controller. Clock stretching is an optional feature defined in I²C specification. If the clock stretching option is enabled (for slave I²C), the data written into transmit buffer is shifted out only after the occurrence of clock stretching, and the data cannot be loaded to transmit shift register. The programmer must write the same data again to the transmit buffer.
INFILEN	Input Noise Filter Enable bit

Set this bit to enable the input noise filter of SDA and SCL lines. When the filter is enabled, it filters out the spike of less than 50nsec.

XMT — This bit is set by the controller when the I²C slave is in transmit operation; this bit is clear when the I²C slave controller is in receive operation.

I2CSST2 (0xDC) I2CS2 Status Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	FIRSTBT	ADDRMI	STOPI	RPSTARTI	TXBI	RCBI	START	NACK
WR	-	ADDRMI	STOPI	RPSTARTI	HOLDT[3]	HOLDT[2]	HOLDT[1]	HOLDT[0]

FIRSTBT — This bit is set to indicate the data in the data register as the first byte received after address match. This bit is cleared after the first byte of the transaction is read. The bit is read only and generated by the slave controller.

ADDRMI — Slave Address Match Interrupt Flag bit
This bit is set when the received address matches the address defined in I2CSADR2. If EADDMI is set, this generates an interrupt. This bit must be cleared by software.

STOPI — Stop Condition Interrupt Flag bit
This bit is set when the slave controller detects a STOP condition on the SCL and SDA lines. This bit must be cleared by software.

RPTSARTI — Repeat Start Condition Interrupt Flag bit
This bit is set when the slave controller detects a REPEAT START condition on the SCL and SDA lines. This bit must be cleared by software.

TXBI — Transmit Buffer Interrupt Flag
This bit is set when the slave controller is ready to accept a new byte for transmit. This bit is cleared when a new data is written into I2CSDAT register.

RCBI — Receiver Buffer Interrupt Flag bit
This bit is set when the slave controller puts a new data in the I2CSDAT and ready for software-reading. This bit is cleared after the software reads I2CSDAT.

START — Start Condition.
This bit is set when the slave controller detects a START condition on the SCL and SDA lines. This bit is not particularly useful as the start of transaction can be indicated by address match interrupt. This read-only bit is cleared when STOP condition is detected.

NACK — NACK Condition.
This bit is set when the host responds with NACK in the byte transaction. This bit is only meaningful for slave-transmit operation. If the master returns with NACK on the byte transaction, the slave does not upload new data into the shift register. And the slave transmits the old data again as the next transfer, and this re-transmission continues if NACK is repeated until the transmission is successful and is returned with ACK. This bit is cleared when a new ACK is detected, or it can be cleared by software.

HOLDT[3-0] — These four bits define the hold time of the peripheral clock (EPPCLK) cycles between SDA to SCL. The I²C specification requires for minimum of 300nsec hold time, so the condition of "TEPPCLK*(HOLDT[3:0]+3) ≧300nsec hold time" equation must be met. For example, if the peripheral clock cycle (EPPCLK) is 20MHz, then HOLD[3-0] should be set to ≧3.

I2CSADR2 (0xDD) I2CS2 Slave Address Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	I2CSEN	ADDR[6-0]						
WR	I2CSEN	ADDR[6-0]						

I2CSENT — Set this bit to enable the I²C slave controller.

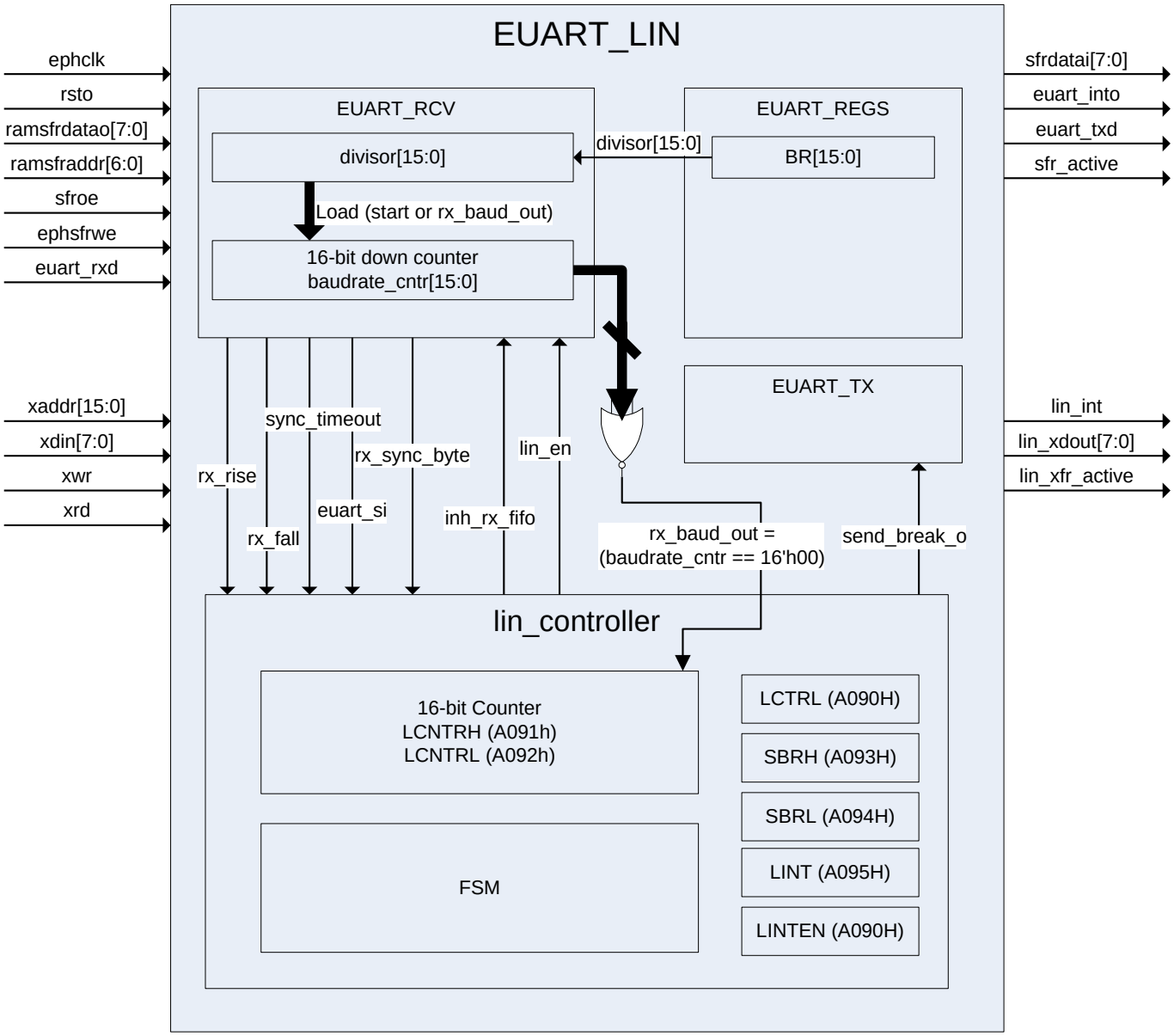
ADDR[6-0] — 7-bit slave address.

I2CSDAT2 (0xDE) I2CS2 Data Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	I²C Slave Receive Data Register							
WR	I²C Slave Transmit Data Register							

5. EUART2 with LIN Controller (EUART2)

LIN-capable 16550-like EUART2 is an enhanced UART controller (EUART) with separate transmit and receive FIFO. Both transmit and receive FIFO are 15-bytes deep and can be parameterized for interrupt triggering. The addition of FIFO significantly reduces the CPU load to handle high-speed serial interface. Transmit FIFO and receive FIFO have respective interrupt trigger levels that can be set based on optimal CPU performance adjustment. The EUART2 also has a dedicated 16-bit Baud Rate generator, and thus, provides accurate baud rate under wide range of system clock frequency. The EUART2 also provides LIN extensions that incorporate message handling and baud-rate synchronization. The block diagram of EUART2 is shown in the following.



The following registers are used for configurations of and interface with EUART2.

SCON2 (0xC2) UART2 Configuration Register 00000000, R/W

	7	6	5	4	3	2	1	0
RD	EUARTEN	SB	WLS[1]	WLS[0]	BREAK	OP	PERR	SP
WR	EUARTEN	SB	WLS[1]	WLS[0]	BREAK	OP	PE	SP

EUARTEN — Transmit and Receive Enable bit
Set to enable EUART2 transmit and receive functions: To transmit messages in the TX FIFO and to store received messages in the RX FIFO.

SB — Stop Bit Control
Set to enable 2 Stop bits, and clear to enable 1 Stop bit.

WLS[1-0]	The number of bits of a data byte. This does not include the parity bit when parity is enabled. 00 - 5 bits 01 - 6 bits 10 - 7 bits 11 - 8 bits
BREAK	Break Condition Control Bit Set to initiate a break condition on the UART interface by holding UART output at low until BREAK bit is cleared.
OP	Odd/Even Parity Control Bit
PE/PERR	Parity Enable / Parity Error status Set to enable parity and clear to disable parity checking functions. If read, PERR=1 indicates a parity error in the current data of RX FIFO.
SP	Parity Set Control Bit When SP is set, the parity bit is always transmitted as 1.

SFIFO2 (0xA5) UART2 FIFO Status/Control Register 00000000 R/W

	7	6	5	4	3	2	1	0
RD	RFL[3-0]				TFL[3-0]			
WR	RFLT[3-0]				TFLT[3-0]			

RFL[3-0] Current Receive FIFO level. This is read-only and indicates the current received FIFO byte count.

RFLT[3-0] Receive FIFO trigger threshold. This is write-only. RDA interrupt will be generated when RFL[3-0] is greater than RFLT[3-0].

RFLT[3-0]	Description
0000	RX FIFO trigger level = 0
0001	RX FIFO trigger level = 1
0010	RX FIFO trigger level = 2
0011	RX FIFO trigger level = 3
0100	RX FIFO trigger level = 4
0101	RX FIFO trigger level = 5
0110	RX FIFO trigger level = 6
0111	RX FIFO trigger level = 7
1000	RX FIFO trigger level = 8
1001	RX FIFO trigger level = 9
1010	RX FIFO trigger level = 10
1011	RX FIFO trigger level = 11
1100	RX FIFO trigger level = 12
1101	RX FIFO trigger level = 13
1110	RX FIFO trigger level = 14
1111	Reset Receive State Machine and Clear RX FIFO

TFL[3-0] Current Transmit FIFO level. This is read-only and indicates the current transmit FIFO byte count.

TFLT[3-0] Transmit FIFO trigger threshold. This is write-only. TRA interrupt will be generated when TFL[3-0] is less than TFLT[3-0].

TFLT[3-0]	Description
0000	Reset Transmit State Machine and Clear TX FIFO
0001	TX FIFO trigger level = 1
0010	TX FIFO trigger level = 2
0011	TX FIFO trigger level = 3
0100	TX FIFO trigger level = 4
0101	TX FIFO trigger level = 5

0110	TX FIFO trigger level = 6
0111	TX FIFO trigger level = 7
1000	TX FIFO trigger level = 8
1001	TX FIFO trigger level = 9
1010	TX FIFO trigger level = 10
1011	TX FIFO trigger level = 11
1100	TX FIFO trigger level = 12
1101	TX FIFO trigger level = 13
1110	TX FIFO trigger level = 14
1111	TX FIFO trigger level = 15

Receive and transmit FIFO can be reset by clearing FIFO operation. This is done by setting BR[11-0]=0 and EUARTEN=0. This also clears RFO, RFU and TFO interrupt flags without writing the interrupt register. The LIN counter LCNTR is also cleared.

SINT2 (0xA7) UART2 Interrupt Status/Enable Register 00000000 R/W

	7	6	5	4	3	2	1	0
RD	INTEN	TRA	RDA	RFO	RFU	TFO	FERR	TI
WR	INTEN	TRAEN	RDAEN	RFOEN	RFUEN	TFOEN	FERREN	TIEN

INTEN — Interrupt Enable bit. Write only
Set to enable UART2 interrupt. Clear to disable interrupt. The fault is 0.

TRA/TRAEN — Transmit FIFO is ready to be filled.
This bit is set when transmit FIFO has been emptied below the FIFO threshold. Write “1” to enable interrupt. The flag is automatically cleared when the condition is absent.

RDA/RDAEN — Receive FIFO is ready to be read.
This bit is set by hardware when received FIFO exceeds the FIFO threshold. Write “1” to enable interrupt. RDA will also be set when RFL < RFLT for bus idle duration longer than RFLT * 16 * Baud Rate. This is to inform the software that there are still remaining unread received bytes in the FIFO.
The flag is cleared when RFL < RFLT and writing “0” on the bit (The interrupt is disabled simultaneously.)

RFO/RFOEN — Receive FIFO Overflow Enable bit
This bit is set when the overflow condition of received FIFO occurs. Write “1” to enable interrupt. The flag can be cleared by software by writing “0” on the bit (The interrupt is disabled simultaneously.), or by FIFO reset action.

RFU/RFUEN — Receive FIFO Underflow Enable bit
This bit is set when the underflow condition of received FIFO occurs. Write “1” to enable interrupt. The flag can be cleared by software by writing “0” on the bit (The interrupt is disabled simultaneously.), or by FIFO reset action.

TFO/TFOEN — Transmit FIFO Overflow Interrupt Enable bit
This bit is set when the overflow condition of transmit FIFO occurs. Write “1” to enable interrupt. The flag can be cleared by software by writing “0” on the bit (The interrupt is disabled simultaneously.), or by FIFO reset action.

FERR/FERREN — Framing Error Enable bit
This bit is set when the framing error occurs as the byte is received. Write “1” to enable interrupt. The flag must be cleared by software by writing “0” on the bit (The interrupt is disabled simultaneously.).

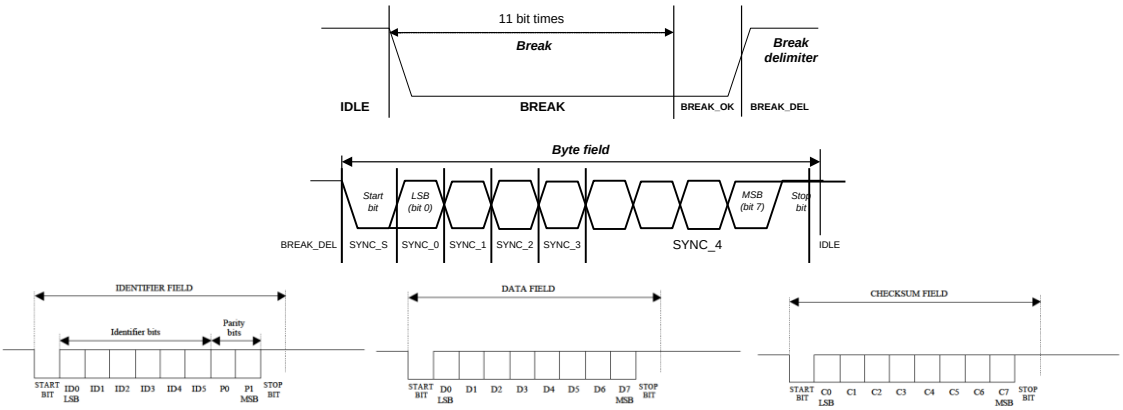
TI/TIEN — Transmit Message Completion Interrupt Enable bit
This bit is set when all messages in the TX FIFO are transmitted, and the TX FIFO becomes empty. Write “1” to enable interrupt. The flag must be cleared by software by writing “0” on the bit (The interrupt is disabled simultaneously.).

SBUF2 (0xA6) UART2 Data Buffer Register 0x00 R/W

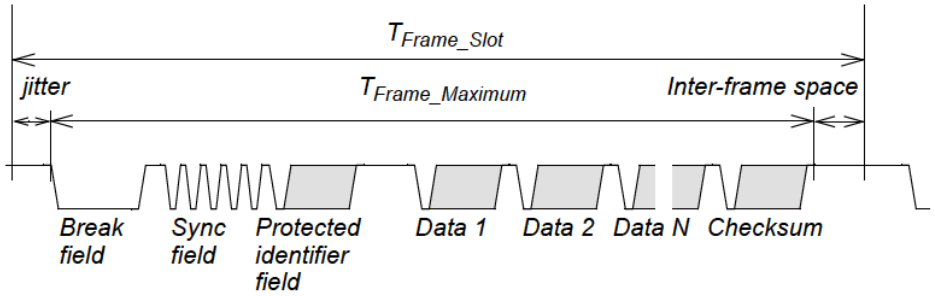
	7	6	5	4	3	2	1	0
RD	EUART2 Receive Data Register							
WR	EUART2 Transmit Data Register							

This register is the virtual data buffer register for both receive and transmit FIFO. When being read, it reads out the top byte of the RX FIFO; when being written, it writes into the top byte of the TX FIFO.

EUART2 can be configured to add LIN capability. The major enhancement of LIN includes master/slave configurations, auto baud-rate synchronization, and frame-based protocol with header. Under LIN extension mode, all EUART2 registers and functions are still effective and operational. LIN is a single-wire bus and it requires external components to combine RX and TX signals externally. LIN is frame-based and consists of message protocols with master/slave configurations. The following diagram shows the basic composition of a header message sent by the master. It starts with BREAK, SYNC byte, ID bytes, DATA bytes, and CRC bytes.



A LIN frame structure is shown as below and the frame time matches the number of bits sent and has a fixed timing.



LIN bus protocol is based on frame. Each frame is partitioned into several parts as shown above. For master to initiate a frame, the software follows the following procedure.

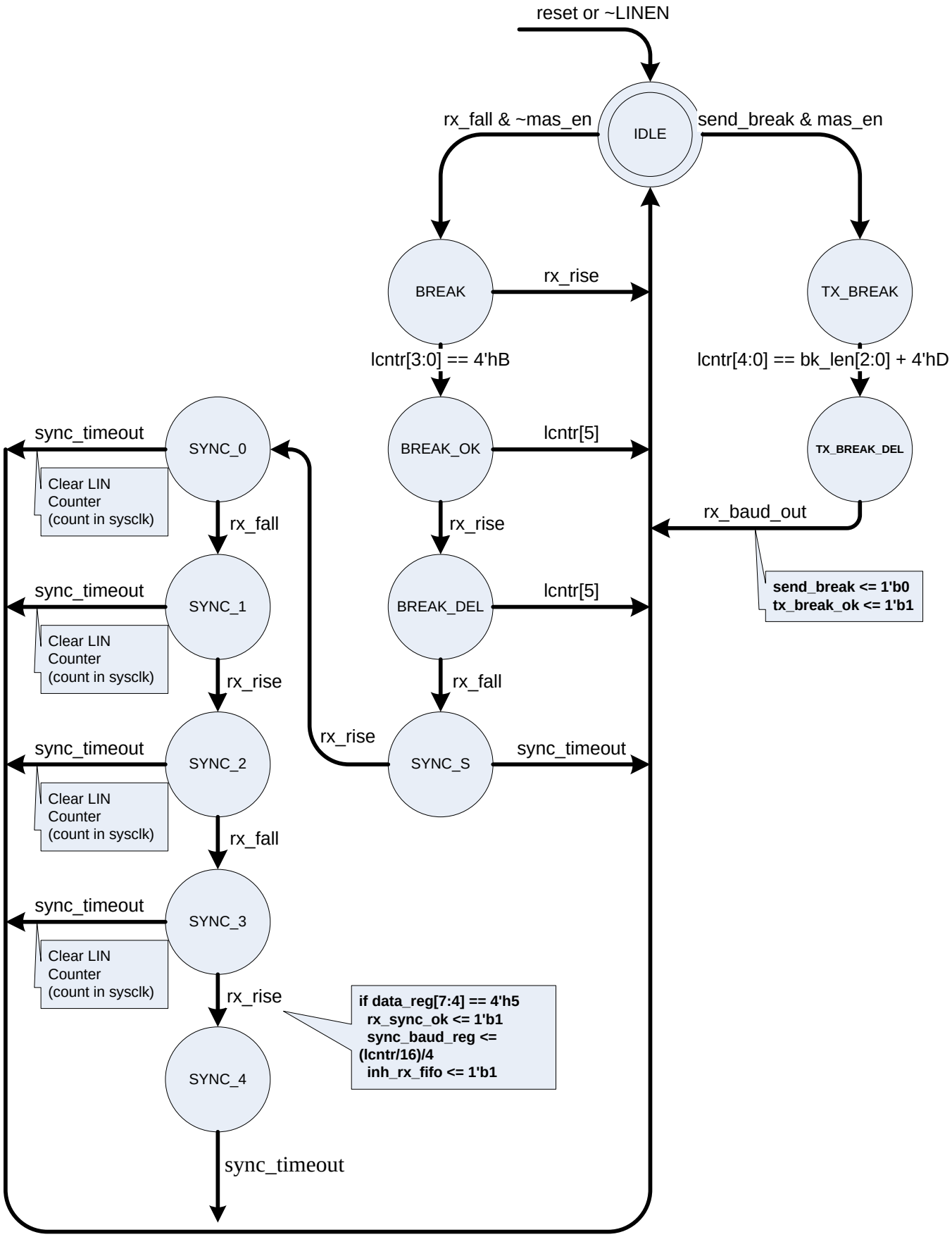
Initiate a SBK command. (SW needs to check if the bus is in idle state, and there is no pending transmit data).

Write “55” into TFIFO.

Write “PID” into TFIFO.

Wait for SBK to complete interrupts and then write the following transmit data if applicable. (This is optional).

The following diagram shows Finite State Machine (FSM) of the LIN extension and is followed by registers within EUART2.


reset or ~LINEN
IDLE
rx_fall & ~mas_en
send_break & mas_en
BREAK
rx_rise
TX_BREAK
lcntr[3:0] == 4'hB
lcntr[4:0] == bk_len[2:0] + 4'hD
BREAK_OK
lcntr[5]
TX_BREAK_DEL
rx_baud_out
send_break <= 1'b0
tx_break_ok <= 1'b1
BREAK_DEL
lcntr[5]
rx_rise
rx_fall
SYNC_S
sync_timeout
rx_rise
SYNC_0
sync_timeout
Clear LIN Counter (count in sysclk)
rx_fall
SYNC_1
sync_timeout
Clear LIN Counter (count in sysclk)
rx_rise
SYNC_2
sync_timeout
Clear LIN Counter (count in sysclk)
rx_fall
SYNC_3
sync_timeout
Clear LIN Counter (count in sysclk)
rx_rise
SYNC_4
if data_reg[7:4] == 4'h5
rx_sync_ok <= 1'b1
sync_baud_reg <= (lcntr/16)/4
inh_rx_fifo <= 1'b1
sync_timeout

LINCTRL (0xA090) LIN Status/Control Register 0x00 R/W

	7	6	5	4	3	2	1	0
RD	LINEN	MASEN	ASU	MASU	SBK	BL[2:0]		
WR	LINEN	MASEN	ASU	MASU	SBK	BL[2:0]		

LINEN — LIN Enable (1: Enable / 0: Disable)
LIN header detection / transmission is functional when LINEN = 1.
※ Before enabling LIN functions, the EUART2 registers must be set correctly：0xB0 is recommended for SCON2.

MASEN — Master Enable bit (1: Master / 0: Slave) LIN operating mode selection. This bit is changeable only when LINEN = 0 (must clear LINEN before changing MASEN).

ASU — Auto-Sync Update Enable (1: Enable / 0: Disable), Write Only
If ASU is 1, the LIN controller will automatically overwrite BR[15-0] with SBR[15-0] and issue an ASUI interrupt when receiving a valid SYNC field.
If ASU is 0, the LIN controller will only notice the synchronized baud rate in SBR[15-0] by issuing a RSI interrupt.
ASU should not be set under UART mode. ASU capability is based on the message containing BREAK and SYNC field in the beginning.
When ASU=1, the auto sync update is performed on every receiving frame and is updated frame by frame.
When ASU is set to 1, LININTEN[SYNCMD] should also be set to 1.

MASU — Message Auto Sync Update Enable
MASU is meaningful only if ASU=0. MASU=1 will enable the auto sync update on the next received frame only. It is self-cleared when the sync update is completed. The software must set MASU again if another auto sync operation is desired.
When MASU is set to 1, LININTEN[SYNCMD] should also be set to 1.

SBK — Send Break (1: Send / 0: No send request)
LINEN and MASEN should be set before setting SBK. When LINEN and MASEN are both 1, set SBK to send a bit sequence of 13+BL[2:0] consecutive dominant bits and 1 recessive bit (Break Delimiter). Once SBK is set, this bit represents the “Send Break” status and CANNOT be cleared by writing to “0”; instead, clearing LINEN cancels the “Send Break” action. In normal cases, SBK is cleared automatically when the transmission of Break Delimiter is completed.

BL[2:0] — Break Length Setting
Break Length = 13 + BL[2:0]. Default BL[2:0] is 3’b000.

LINCNTRH (0xA091) LIN Timer Register High (0xFF) R/W

	7	6	5	4	3	2	1	0
RD	LCNTR15-8]							
WR	LINTMR[15-8]							

LINCNTRL (0xA092) LIN Time Register Low (0xFF) R/W

	7	6	5	4	3	2	1	0
RD	LCNTR[7-0]							
WR	LINTMR[7-0]							

LCNTR[15-0] is read only and is an internal 16-bit counter clocked by the baud rate clock. LINTMR[15-0] is write only and is the timer limit for LCNTR[15-0]. If MASEN=1 as LIN master mode, this timer is used to generate Frame time base. The internal counter LCNTR[15-0] is cleared whenever a “SEND BREAK” command is executed, and when the counter reaches LINTMR [15-0] (LCNTR[15-0] >= LINTMR[15-0]), a LCNTRO interrupt is generated. Thus, the software can write a Frame Time value into LINTMR and use interrupts to initiate frames. If MASEN=0 as LIN slave mode, this timer is used for determining the accumulated bus idle time. The internal counter is cleared whenever a RX transition occurs. When the internal counter reaches LINTMR[15-0], a LCNTRO interrupt is generated. The software can use this interrupt to enter sleep mode by writing the required bus idling time into LINTMR[15-0].

LINSBRH (0xA093) EUART/LIN Baud Rate Register High byte (0x00) RO

	7	6	5	4	3	2	1	0
RD	SBR[15-8]							
WR	BR[15-8]							

LINSBRL (0xA094) EUART/LIN Baud Rate Register Low byte (0x00) RO

	7	6	5	4	3	2	1	0
RD	SBR[7:0]							
WR	BR[7-0]							

SBR[15-0] The acquired Baud Rate under LIN protocol. This is read-only.
SBR[15-0] is the acquired baud rate from last receiving valid sync byte. SBR is meaningful only in LIN-Slave mode.

BR[15-0] The Baud Rate Setting of EUART/LIN. This is write-only. BR[15-0] cannot be 0.
BUAD RATE = SYSCLK/BR[15-0].

When a slave receives a BREAK followed by a valid SYNC field, a RSI interrupt is generated and the acquired baud rate from SYNC field is stored in SBR[15-0]. The acquired baud rate is BAUD RATE = SYSCLK/SBR[15-0]. The software can just update this acquired value into BR[15-0] to achieve synchronization with the master. If Auto-Sync Update (ASU) register bit is enabled under LIN slave mode, LIN controller will automatically perform the update of BR[15-0] with SBR[15-0] and issue another ASUI interrupt when receiving a valid SYNC field.

LININT (0xA095) LIN Interrupt Flag Register (0x00) R/W

	7	6	5	4	3	2	1	0
RD	RXST	BITERR	LSTAT	LIDLE	ASUI	SBKI	RSI	LCNTRO
WR	-	BITERR	BECLRX	BECLRR	ASUI	SBKI	RSI	LCNTRO

RXST Receive Status
RXST is set by hardware when a START bit is detected. It is cleared when STOP condition is detected.

BITERR Bit Error Flag
BITERR is set by hardware when the received bit does not match with transmit bit, if BERIE=1, then this error generates an interrupt. BITERR must be cleared by software.

BECLRX Bit Error Force Clear Transmit Enable
If BECLRX=1, when BITERR is set by hardware, hardware also immediately disables current transmission and clears TX state machines and FIFO.

BECLRR Bit Error Force Clear RECEIVE Enable
If BECLRX=1, when BITERR is set by hardware, hardware also immediately disables current reception and clears RX state machines and FIFO.

LSTAT LIN Bus Status bit (1: Recessive / 0: Dominant), Read only.
LSTAT = 1 indicates that the LIN bus (RX pin) is in recessive state.

LIDLE LIDLE is 1 when LIN bus is idle and not transmitting/receiving LIN header or data bytes. This bit is read only. It is 1 when LINEN = 0.

ASUI Auto-Sync Updated completion Interrupt (1: Set / 0: Clear)
This flag is set when auto baud rate synchronization has been completed and BR[15-0] has been updated with SBR[15-0] by hardware. It must be cleared by writing “1” on the bit.

SBKI Send Break Completion Interrupt bit (1: Set / 0: Clear)
This flag is set when Send Break is completed. It must be cleared by writing “1” in the bit.

RSI Receive Sync Completion Interrupt bit (1: Set / 0: Clear)
This flag is set when a valid Sync byte is received following a Break. It must be cleared by writing “1” to the bit.

LCNTRO LIN Counter Overflow Interrupt bit (1: Set / 0: Clear).
This flag is set when the LIN counter reaches 0xFFFF. It must be cleared by writing “1” in the bit.

LININTEN (0xA096) LIN Interrupt Enable Register (0x00) R/W

	7	6	5	4	3	2	1	0
RD	LINTEN	BERIE	SYNCMD	SYNCVD	ASUIE	SBKIE	RSIE	LCNTRIE
WR	LINTEN	BERIE	SYNCMD	EUARTOPL	ASUIE	SBKIE	RSIE	LCNTRIE

LINTEN	LIN Interrupt Enable (1: Enable / 0: Disable) Set to enable all LIN interrupts. LINT flags should be checked before setting or modifying.
BERIE	Bit Error Interrupt Enable (1: Enable / 0: Disable)
SYNCMD	Synchronization Mode Selection SYNCMD=0 will only allow automatic synchronization of baud rate within +/- 6% deviations. SYNCMD=1 will automatically re-synchronize with the newly received message frame and update the baud rate register with the newly acquired baud rate. SYNCMD should be set to 1 when either ASU or MASU is 1. The new baud rate can be successfully received and must meet the following conditions: 1. Within +/- 50% of the current baud rate setting 2. Break length is less than 32 current baud rate bit times and less than 253952 SYSCLK
SYNCVD	Synchronization Valid Status SYNCVD is updated by the hardware when SYNCMD=1. SYNCVD is set to 1 if the auto synchronization is successful.
EUARTOPL	EUART/LIN output polarity EUARTOPL=1 will reverse the transmit output polarity
ASUIE	Auto-Sync Update Interrupt Enable (1: Enable / 0: Disable)
SBKIE	Send Break Completion Interrupt Enable (1: Enable / 0: Disable)
RSIE	Receive Sync Completion Interrupt Enable (1: Enable / 0: Disable)
LCNTRIE	LIN Counter Overflow Interrupt Enable (1: Enable / 0: Disable)

LINTCON (0xA0B0h) LIN Time Out configuration R/W (0x00)

	7	6	5	4	3	2	1	0
RD	RXDTO[0]	LINRXFEN	RXTOWKE	TXTOWKE	RXDD_F	TXDD_F	RXDDEN	TXDDEN
WR	RXDTO[0]	LINRXFEN	RXTOWKE	TXTOWKE	RXDD_F	TXDD_F	RXDDEN	TXDDEN

RXDTO[0]	RXD Dominant Time Out Timer [0] This is combined with RXDTOH and RXDTOL to form RXDTO[16-0]
LINRXFEN	LIN Break State Exit when RXD dominant fault occurs. LINRXFEN=1 configures the automatic BREAK state exit under RXD dominant fault conditions. LINRXFEN=0 disable this automatic exit (Does not affect other break exit conditions.). Software must take care of the LIN state machine.
RXDDEN	RXD Dominant Fault Interrupt Enable
RXDD_F	RXD Dominant Fault Interrupt Flag RXDD_F is set to 1 by hardware and must be cleared by software.
TXDDEN	TXD Dominant Fault Interrupt Enable
TXDD_F	TXD Dominant Fault Interrupt Flag TXDD_F is set to 1 by hardware and must be cleared by software.
TXTOWKE	TXD Dominant Timeout Wakeup Enable
RXTOWKE	RXD Dominant Timeout Wakeup Enable

TXDTOL (0xA0B1h) LIN TXD Dominant Time Out LOW Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TXDTO[7:0]							
WR	TXDTO[7:0]							

TXDTOH (0xA0B2h) LIN TXD Dominant Time Out HIGH Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TXDTO[15:8]							
WR	TXDTO[15:8]							

TXDTO	TXD Dominant Time Out (TXDTO +1) * IOSCCLK

RXDTOL (0xA0B3h) LIN RXD Dominant Time Out LOW Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	RXDTO[8:1]							
WR	RXDTO[8:1]							

RXDTOH (0xA0B4h) LIN RXD Dominant Time Out HIGH Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	RXDTO[16:9]							
WR	RXDTO[16:9]							

RXDTO RXD Dominant Time Out (RXDTO[16:0] +1) * IOSCCLK

BSDCLR (0xA0B5h) Bus Stuck Dominant Clear Width Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	BSDCLR [7:0]							
WR	BSDCLR [7:0]							

BSDCLR Bus Stuck Dominant Clear Time (BSDCLR +1) * SOSC32KHz

BSDACT (0xA0B6h) Bus Stuck Dominant Active Width Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	BSDACT [7:0]							
WR	BSDACT [7:0]							

BSDACT Bus Stuck Dominant Active Time (BSDCLR +1) * SOSC32KHz

BSDWKC (0xA0B7h) Bus Stuck Dominant Fault Wakeup configuration R/W (0x00)

	7	6	5	4	3	2	1	0
RD	BSDW_F	BFW_F	BSDWEN	BFWEN	WKFLT[3:0]			
WR	BSDW_F	BFW_F	BSDWEN	BFWEN	WKFLT[3:0]			

WKFLT LIN Wakeup time (WKFLT+1) * SOSC32KHz
BFWEN LIN Wakeup/Interrupt Enable
BFW_F LIN Wakeup Interrupt Flag
BFW_F is set to 1 by hardware and must be cleared by software
BSDWEN LIN Bus Stuck Wakeup Interrupt Enable

Due to the implementation of Bit Error detection, if hardware detection of bit error is not used, BECLRX, BECLRR and BERIE should be set low, and ignore BITERR status.

If bit error detection is done by the hardware and at the completion of a message transmission (TI/TIEN Transmit Message Completion Interrupt Flag), software needs to check BITERR, and clear BITERR, BECLRX, BECLRR and BERIE. To start a new transmission, the software should first need to clear BITERR, enable BECLRX, BECLRR, and BERIE, and then write data into FIFO.

6. Serial Peripheral Interface (SPI)

The Serial Peripheral Interface (SPI) is an enhanced synchronous serial hardware, which is compatible with Motorola's SPI specifications. The SPI Controller includes 4-bytes FIFO for both transmit and receive. SPI Interface uses Master-Out-Slave-In (MOSI), Master-In-Slave-Out (MISO), Serial Clock (SCK) and Slave Select (SSN) for interface. SSN is low active and only meaningful in slave mode. Due to oversampling, the maximum SPI clock rate is limited to SYSCLK/4 for both slave and master configurations.

SPICR (0xA1) SPI Configuration Register R/W (0b001000xx)

	7	6	5	4	3	2	1	0
RD	SPIE	SPEN	MSTR	CPOL	CPHA	SCKE	SICKFLT	SSNFLT
WR	SPIE	SPEN	MSTR	CPOL	CPHA	SCKE	SICKFLT	SSNFLT

SPIE SPI interface Interrupt Enable bit.
SPEN SPI interface Enable bit.
MSTR SPI Master/Slave Switch (set as a master; clear as a slave)
CPOL SPI interface Polarity bit: Set to configure the SCK to stay HIGH while the SPI interface is idling and clear to keep it LOW.
CPHA Clock Phase Control bit: If CPOL=0, set to shift output data at rising edge of SCK, and clear to shift output data at falling edge of SCK. If CPOL=1, set to shift output data at falling edge of SCK and clear to shift output data at rising edge of SCK.
SCKE Clock Selection bit in Master Mode: Set to use rising edge of SCK to sample the input data. Clear to use falling edge of SCK to sample the input data.
SSNFLT Enable noise filter function on signal SSN
SICKFLT Enable noise filter function on signals SDI and SCK

In Slave mode, the sampling phase is determined by the combinations of CPOL and CPHA setting and is shown in the following table.

CPOL	CPHA	(Slave mode) SCK edge used for sampling input data	Data shift out
0	0	Rising edge	Falling edge
0	1	Falling edge	Rising edge
1	0	Falling edge	Rising edge
1	1	Rising edge	Falling edge

SPIMR (0xA2) SPI Mode Control Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	ICNT1	ICNT0	FCLR	-	SPR[2-0]			DIR
WR	ICNT1	ICNT0	FCLR	-	SPR[2-0]			DIR

ICNT1, ICNT0 FIFO Byte Count Threshold.
This sets the FIFO threshold for generating SPI interrupts.
00 –the interrupt is generated after 1 byte is sent or received;
01 –the interrupt is generated after 2 bytes are sent or received;
10 –the interrupt is generated after 3 bytes are sent or received;
11 –the interrupt is generated after 4 bytes are sent or received.
FCLR FIFO Clear/Reset
Set to clear and reset transmit and receive FIFO
SPR[2-0] SPI Clock Rate Setting. This is used to control the SCK clock rate of SPI interface.
000 –SCK = SYSCLK/4;
001 – SCK = SYSCLK/6;
010 – SCK = SYSCLK/8;
011 – SCK = SYSCLK/16;
100 – SCK = SYSCLK/32;
101 – SCK = SYSCLK/64;
110 – SCK = SYSCLK/128;
111 – SCK = SYSCLK/256.
DIR Transfer Format
DIR=1 uses MSB-first format.

DIR=0 uses LSB-first format.

SPIST (0xA3) SPI Status Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	SSPIF	ROVR	TOVR	TUDR	RFULL	REMPT	TFULL	TEMPT
WR	SSPIF	ROVR	TOVR	TUDR	-	-	-	-

SSPIF	SPI Interrupt Flag bit. Set by hardware to indicate the completion of data transfer. Clear by assigning this bit to 0 or disabling SPI.
ROVR	Receive FIFO-overrun Error Flag bit. When Receiver FIFO Full Status occurs and SPI receives new data, ROVR is set and generates an interrupt. Clear by assigning this bit to 0 or disabling SPI.
TOVR	Transmit FIFO-overrun Error Flag bit. When Transfers FIFO Full Status occurs and new data is written, TOVR is set and generates an interrupt. Clear by assigning this bit to 0 or disabling SPI.
TUDR	Transmit Under-run Error Flag bit. When Transfers, FIFO Empty Status and new data transmission occur, TUDR is set and generates an interrupt. Clear by written 0 to this bit or disable SPI.
RFULL	Receive FIFO Full Status bit. Set when receiver FIFO is full. Read only.
REMPT	Receive FIFO Empty Status bit. Set when receiver FIFO is empty. Read only.
TFULL	Transmitter FIFO Full Status bit. Set when transfer FIFO is full. Read only.
TEMPT	Transmitter FIF0 Empty Status bit. Set when transfer FIFO is empty. Read only.

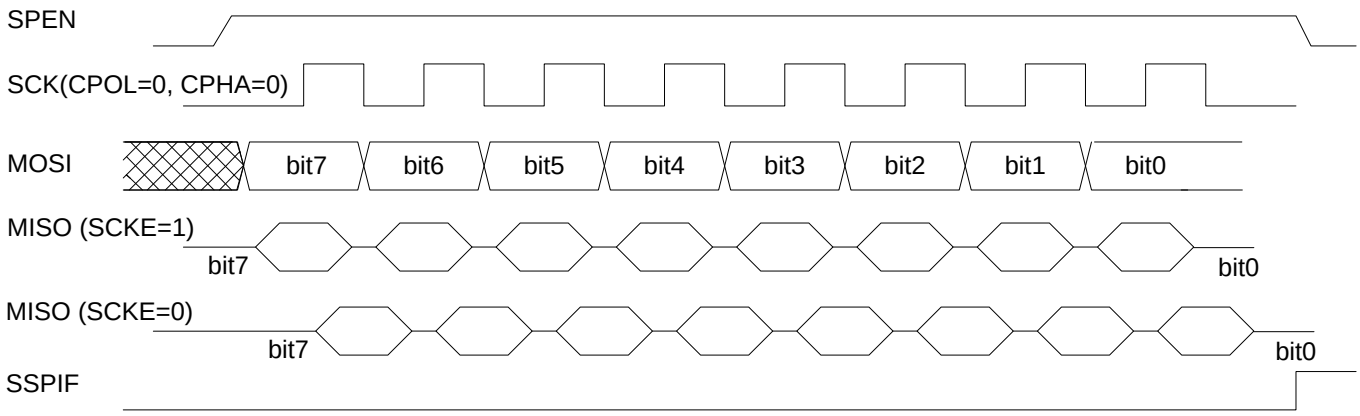
SPIDATA (0xA4) SPI Data Register R/W (0xXX)

	7	6	5	4	3	2	1	0
RD	SPI Receive Data Register							
WR	SPI Transmit Data Register							

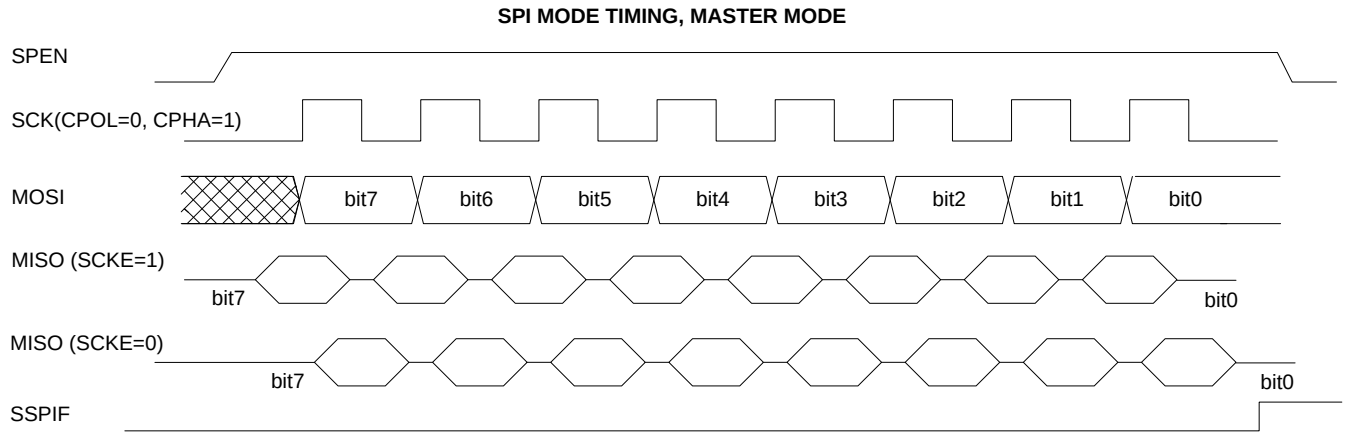
6.1 SPI Master Timing Illustration

6.1.1 CPOL=0 CPHA=0

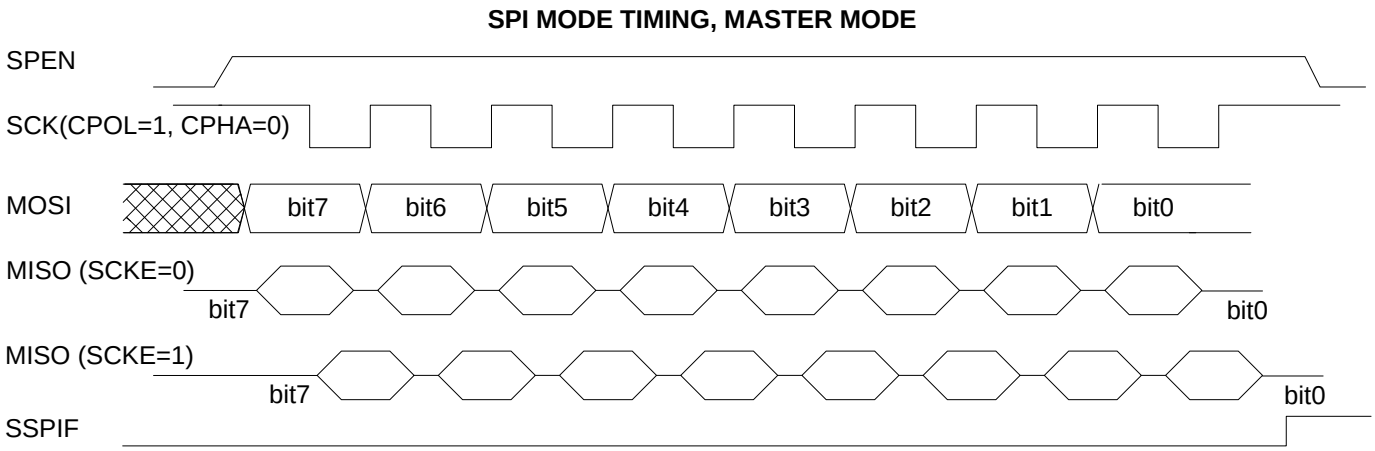
SPI MODE TIMING, MASTER MODE



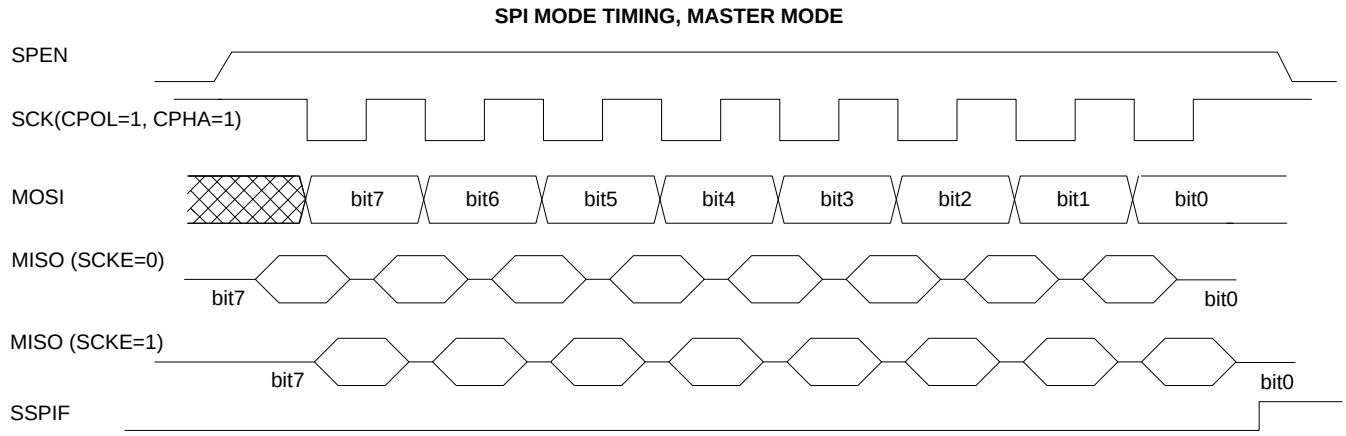
6.1.2 CPOL=0 CPHA=1



6.1.3 CPOL=1 CPHA=0

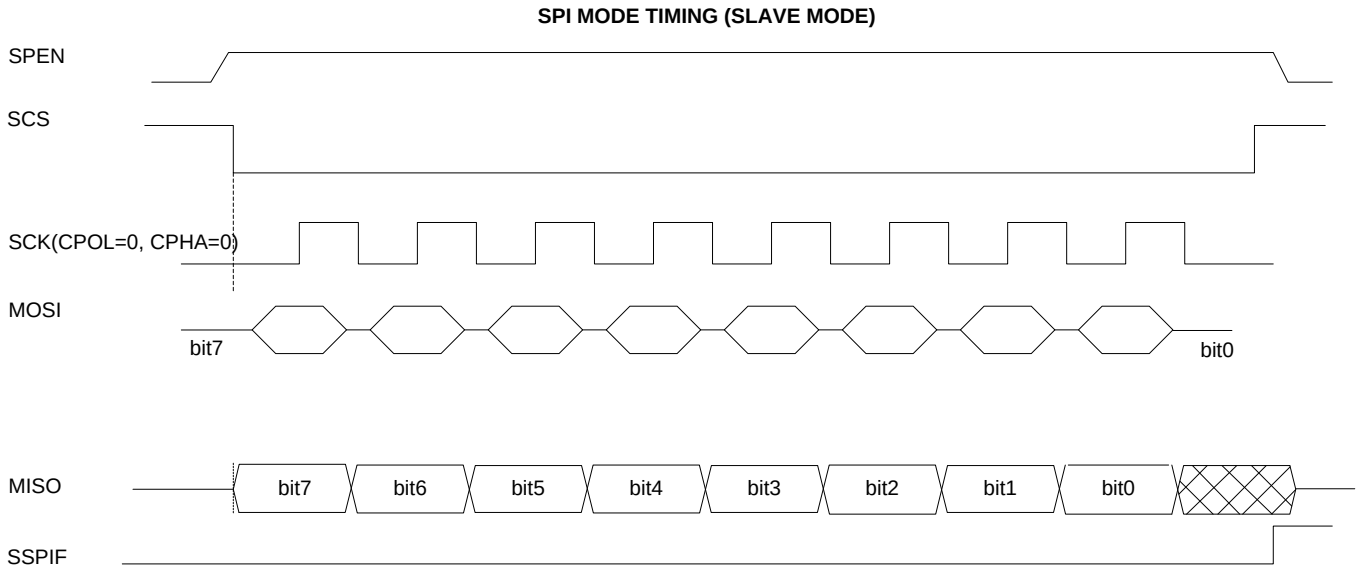


6.1.4 CPOL=1 CPHA=1

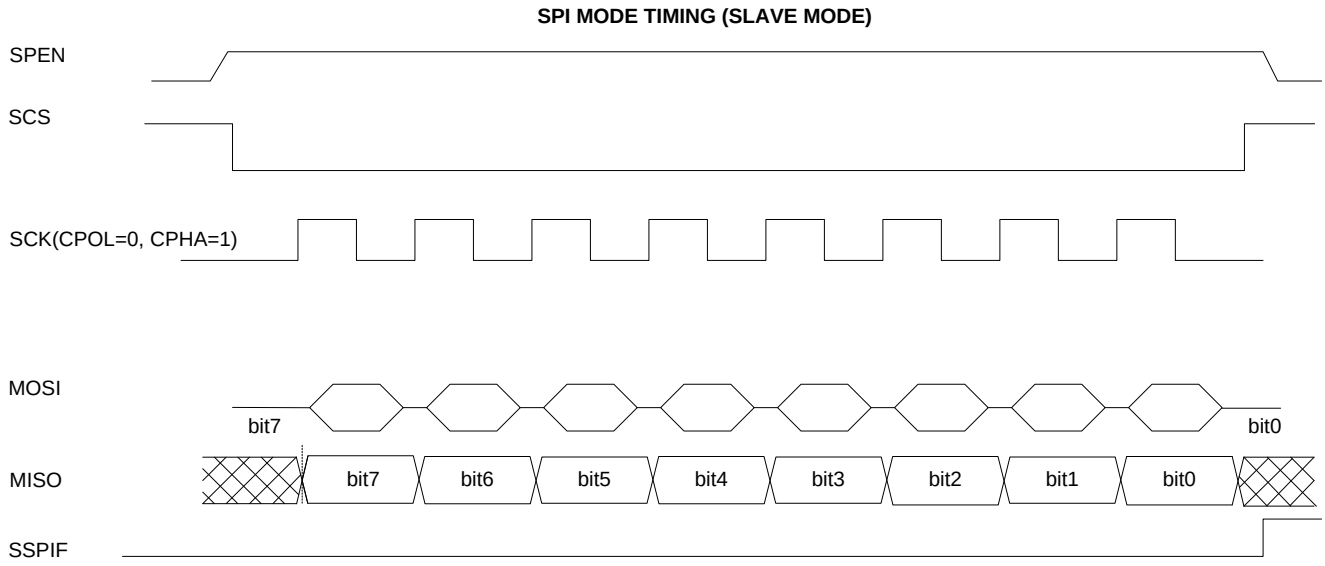


6.2 SPI Slave Timing Illustration

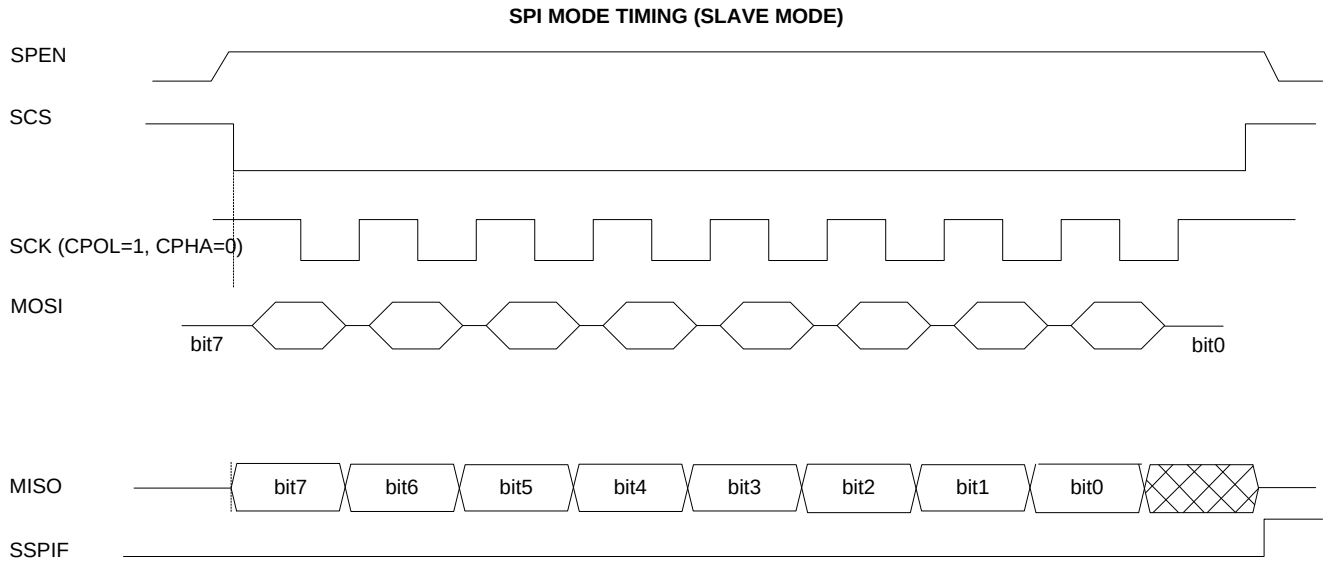
6.2.1 CPOL=0 CPHA=0



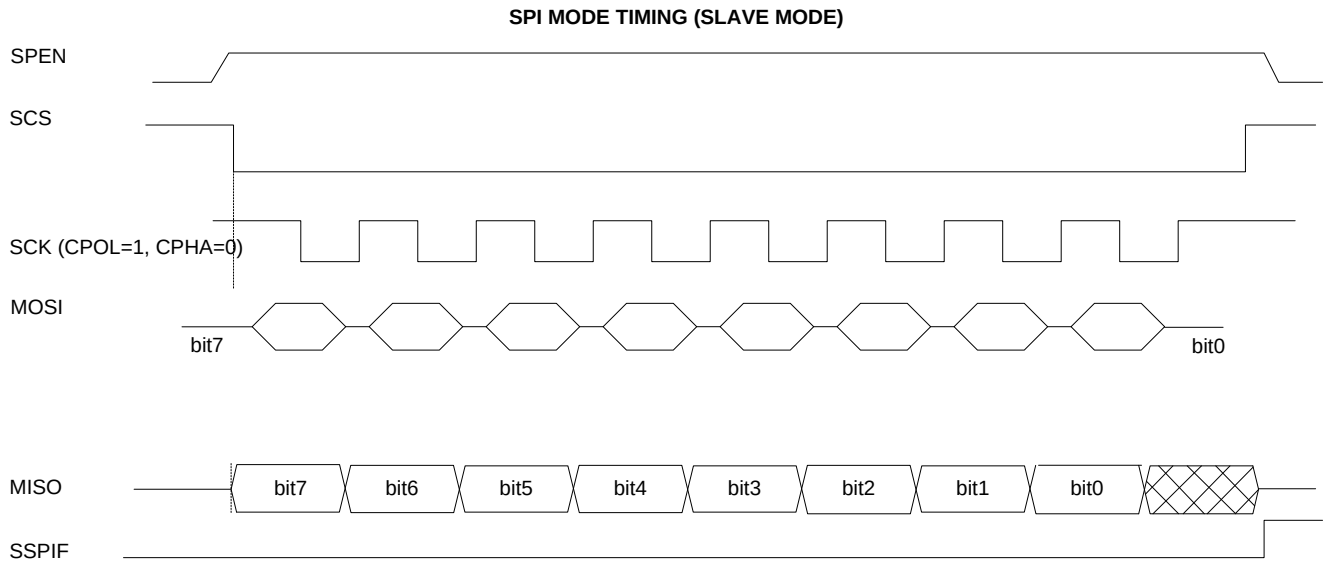
6.2.2 CPOL=0 CPHA=1



6.2.3 CPOL=1 CPHA=0



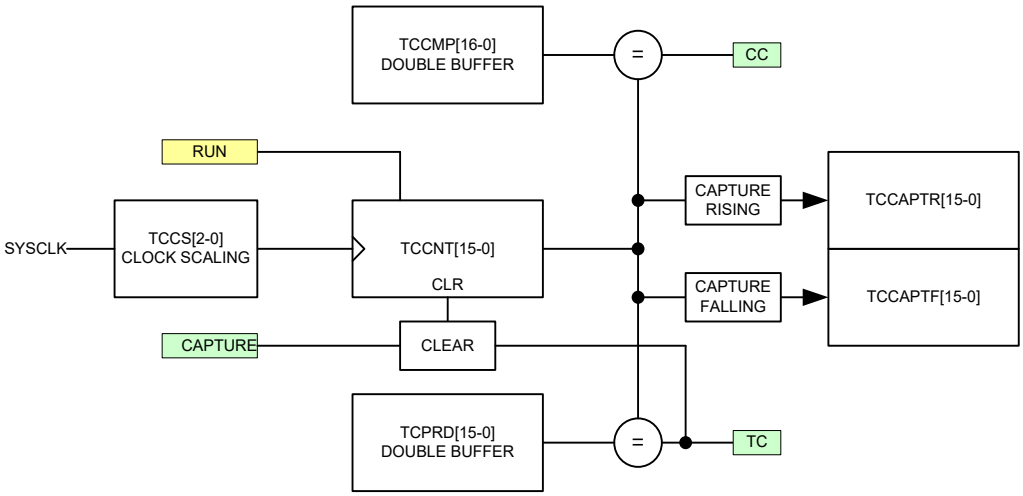
6.2.4 CPOL=1 CPHA=1



This section describes the pin functions and configurations. Almost all signal pins are multi-functional with default setting as a GPIO port pin. Therefore, each signal pin requires two registers to configure the I/O capability and the function selection. The following describes the control and contents of these registers, and the register names and pin names are referenced by their default GPIO port name. The standardized I/O design allows flexible configuration of the digital I/O function such as open-drain, open-source, pull-up, pull-down, bus-holder capabilities. In addition to digital I/O function, the standardized I/O also provides analog I/O capability that can be selected when the GPIO pin is shared with analog peripheral purposes such as analog OPAMP.

7. Timer with Compare/Capture and Quadrature Encoder

The Timer/Capture unit is based on a 16-bit counter with a pre-scalable SYSCLK as a counting clock. The count starts from 0 and reloads when reaching TC (terminal count). TC is met when the count equals the period value. Along with the counting, the count value is compared with COMP and when it matches, a CC condition is met. Note that both PERIOD and COMP registers are double buffered, and therefore, any new value is updated after the current period ends. TC and CC can be used for triggering an interrupt and are also routed to GPIO. The output pulse width of TC and CC is programmable. For CC, it can also be configured as a PWM output. There are two data registers for capture events. The capture event can be from external signals from GPIO with edge selection option, from QE block, or triggered by software. The software can also select if it is necessary to reset the counter or not. This option gives a simpler calculation of consecutive capture events without any offset. The following block diagram shows the TCC implementation.



TCCFG1 (0xA050h) TCC Configuration Register 1 R/W (0x00)

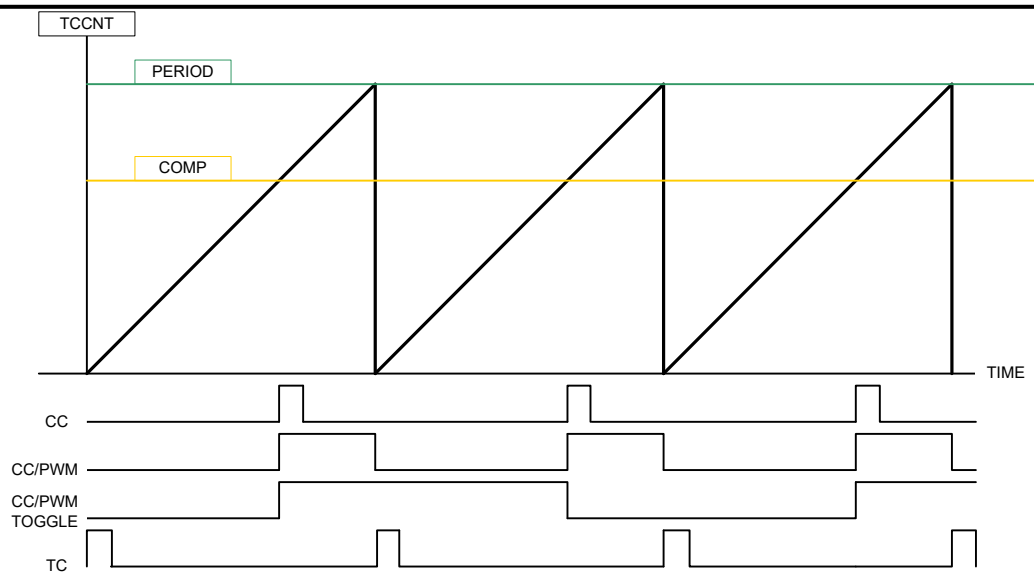
	7	6	5	4	3	2	1	0
RD	TCEN	TCCS[2-0]			CCSEL[1-0]		TCSEL	RUNST
WR	TCEN	TCCS[2-0]			CCSEL[1-0]		TCSEL	RUN

TCEN — TC Enable
TC = 0 disables TC. In the disabled state, TCCNT, and TCCPTR/TCCPTF are cleared to 0. TC and CC are also set to low.
TC = 1 enables TC. RUN bit also needs to set to 1 to start the counter, otherwise, the counter is in pause mode if RUN=0.

TCCS[2-0] — TC Clock Scaling
000 SYSCLK
001 SYSCLK/2
010 SYSCLK/4
011 SYSCLK/8
100 SYSCLK/16
101 SYSCLK/32
110 SYSCLK/64
111 SYSCLK/128

CCSEL[1-0] — CC Output Pulse Select
00 PW = 16 TCCLK
01 PW = 64 TCCLK
10 PWM Waveform (CC = low when TCCNT < CMP, CC = high when TCCNT >= CMP).
11 PWM Toggle waveform (CC toggles when TCCNT = CMP).

TCSEL — TC Output Pulse Select
0 PW = 16 TCCLK
1 PW = 64 TCCLK



RUNST Run Status
Set by hardware to indicate running TC counter. RUNST=1 indicates running.

RUN Run or Pause TC Counter
Writing “0” to RUN will pause the TC counting.
Writing “1” to RUN will resume the TC counting.

TCCFG2 (0xA051h) TC Configuration Register 1 R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-	IDXST	PHAST	PHBST	TCPOL	CCPOL	TCF	CCF
WR	RSTTC	-	-	-	TCPOL	CCPOL	TCF	CCF

RSTTC Reset TC
Writing 1 to RSTTC will reset the TC counter and the capture registers. Once the counter is cleared, TC counter is put in STOP mode. To resume counting, RUN bit must be set by software.

IDXST Index Input real-time status

PHAST PHA input real-time status

PHBST PHB input real-time status

TCPOL TC output polarity

CCPOL CC output polarity

TCF Terminal Count Interrupt Flag
TCF is set to “1” by hardware when terminal count occurs. TCF must be cleared by software by writing “0”.

CCF Compare Match Interrupt Flag
CCF is set to “1” by hardware when a compare match occurs. CCF must be cleared by software by writing “0”.

TCCFG3 (0xA052h) TC Configuration Register 3 R/W (0x00)

	7	6	5	4	3	2	1	0
RD	IENTC	IENCC	QECEN	CPTCLR	XCREN	XCFEN	-	-
WR	IENTC	IENCC	QECEN	CPTCLR	XCREN	XCFEN	SWCPTR	SWCPTF

IENTC TC Interrupt Enable

IENCC CC Interrupt Enable

QECEN QE Capture Enable
QECEC=1 uses the QE output event as the capture event.

CPTCLR Enable Clear Counter after Capture
If CPTCLR=1, the TCCNT is cleared to 0 after each capture event. This allows continuous capture value with the identical initial value.
If CPTCLR=0, the capture event does not affect the TCCNT counting.

XCREN	External Rising Edge Capture Enable XCREN=1 uses external input rising edge as a capture event.
XCFEN	External Falling Edge Capture Enable XCFEN=1 uses external input rising edge as a capture event.
SWCPTR	Software Capture R Writing “1” to SWCPTR will generate a capture event and capture the count value into CAPTR register. This bit is cleared by hardware.
SWCPTF	Software Capture F Writing “1” to SWCPTF will generate a capture event and capture the count value into CAPTF register. This bit is cleared by hardware.

All capture sources are not mutually exclusive, i.e., allow several capture sources to coexist.

TCPRDL (0xA054h) TC Period Register Low Double Buffer R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TCCNT[7-0]							
WR	TCPRD[7-0]							

TCPRDH (0xA055h) TC Period Register High Double Buffer R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TCCNT15-8]							
WR	TCPRD[15-8]							

Note: Writing of PERIOD register must be done high byte first, and then low byte. The writing takes effect at low byte writing. When reading the TCPRD register, it returns the current count value TCCNT[15-0].

TCCMPL (0xA056h) TC Compare Register Low Double Buffer R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TCCMP[7-0]							
WR	TCCMP[7-0]							

TCCMPH (0xA057h) TC Compare Register High Double Buffer R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TCCMP15-8]							
WR	TCCMP[15-8]							

Note: Writing of COMPARE register must be done high byte first, and then low byte. The writing takes effect at low byte writing.

TCCPTRL (0xA060h) TC Capture Register R Low R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TCCPTR[7-0]							
WR	-							

TCCPTRH (0xA061h) TC Capture Register R High R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TCCPTR15-8]							
WR	-							

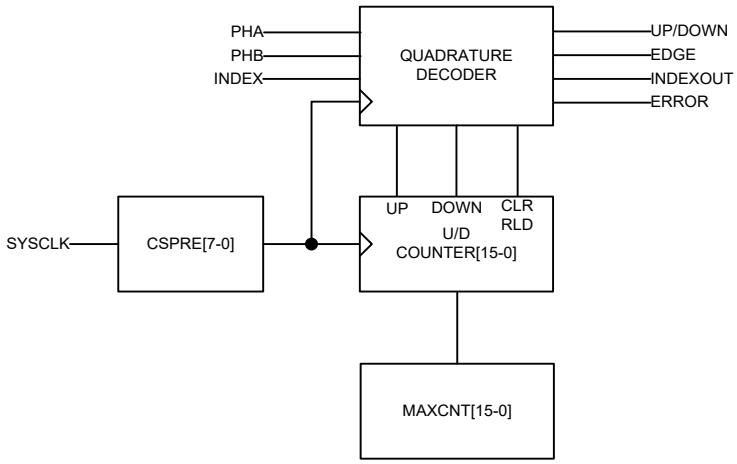
TCCPTFL (0xA062h) TC Capture Register F Low R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TCCPTF[7-0]							
WR	-							

TCCPTFH (0xA063h) TC Capture Register F High R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TCCPTF[15-8]							
WR	-							

The quadrature encoder is clocked by a scaled SYSCLK and has three external inputs through GPIO multi-functions. The three inputs include two signals of 90 degrees phase difference, PHA and PHB, and an index indicating the terminal of the encoder. QE can function as an independent function block and can be configured to couple with TCC and use TCC to calculate the speed information of the encoder. Using TCC to capture TCC count value using the Index input of QE or the terminal count of QE, the speed of QE input can be calculated. The QE unit implementation is shown in the following block diagram.

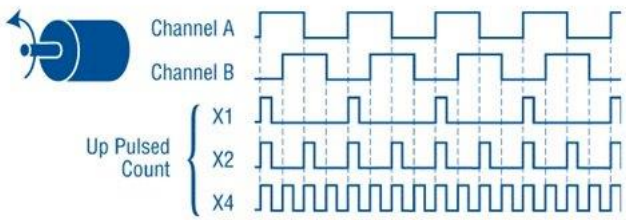


QE Counter is in signed integer format. The MSB (bit 15) indicates the sign, and reload action causes the counter to load a default value 0x8000. The corresponding maximum count register only has 15 valid bits, and MSB bit is not used. The reload action is triggered either by an external INDEX event or the terminal count condition when the counter absolute value equals to MAXCNT value.

QECFG1 (0xA070h) TCC Configuration Register 1 R/W (0x00)

	7	6	5	4	3	2	1	0
RD	QEMODE[1-0]		QECS[1-0]		SWAP	DBCS[2-0]		
WR	QEMODE[1-0]		QECS[1-0]		SWAP	DBCS[2-0]		

MODE[1-0] QE Mode
00 – Disable QE
01 – 1X mode
10 – 2X mode
11 – 4X mode



QECS[1-0] QE Clock Scaling
00 SYSCLK/4
01 SYSCLK/16
10 SYSCLK/64
11 SYSCLK/256

SWAP Swap PHA and PHB

DBCS[2-0] De-Bounce Clock Scaling
000 Disable de-bounce
001 SYSCLK/2
010 SYSCLK/4
011 SYSCLK/8
100 SYSCLK/16
1/32 SYSCLK/32
1/64 SYSCLK/64
1/128 SYSCLK/128
1/256 SYSCLK/256
De-bounce time is three DBCS periods.

QECFG2 (0xA071h) QE Configuration Register 2 R/W (0x00)

	7	6	5	4	3	2	1	0
RD	DIR	ERRF	RLDM[1-0]		TCF	IDXF	DIRF	CNTF
WR	-	ERRF	RLDM[1-0]		TCF	IDXF	DIRF	CNTF

DIR Direction Status
Indicate UP/DOWN direction

ERRF Phase Error Flab
ERRF is set to 1 by hardware if PHA and PHB change value at the same time. ERRF must be cleared by software.

RLDM[1-0] QE Counter Reload Mode
RLDM[1-0] = 00 No Reload, QECNT will count up/down cycling through 0x0000 or 0xFFFF
RLDM[1-0] = 01 Reload using Index event.
Reload QECNT=0 when Index==1 && UP
Reload QECNT=QEMAX when Index==1 && DOWN
RLDM[1-0] = 10 Reload using TC event.
Reload QECNT=0 when QECNT==QEMAX && UP
Reload QECNT=QEMAX when QECNT==0 && DOWN
RLDM[1-0] = 11 Reload using both Index and TC events
Combine Index and TC events and reload whichever occurs earlier.

TCF TC Event Interrupt Flag
TCF is set by hardware when a TC event interrupt has occurred. TCF needs to be cleared by software by writing “0”.

IDXF Index Event Interrupt Flag
IDXF is set by hardware when an Index event interrupt has occurred. IDXF needs to be cleared by software by writing “0”.

DIRF Direction Change Event Interrupt Flag
DIRF is set by hardware when a Direction change event interrupt has occurred. DIRF needs to be cleared by software by writing “0”.

CNTF Count Change Event Interrupt Flag
CNTF is set by hardware when a QE count change event interrupt has occurred. CNTF needs to be cleared by software by writing “0”.

QECFG3 (0xA072h) QE Configuration Register 3 R/W (0x00)

	7	6	5	4	3	2	1	0
RD	IENTC	IENIDX	IENDIR	IENCNT	IENERR	IDXEN	IDXM[1-0]	
WR	IENTC	IENIDX	IENDIR	IENCNT	IENERR	IDXEN	IDXM[1-0]	

IENTC Interrupt Enable for TC
TC condition for QE is defined as the following conditions.
1. QECNT=QEMAX when UP
2. QECNT=0 when DOWN

IENIDX Interrupt Enable for Index event

IENDIR Interrupt Enable for Direction change

IENCNT Interrupt Enable for any QECNT change

IDXEN	Index Input Enable IDXEN=0 gates out the external INDEX input and is gated to 0. IDXEN=1 allows external INDEX.
IDXM[1-0]	Index Match Selection, this is applicable only for X2 and X4 modes. 00 = not gated 01 = PHA gating 10 = PHB gating 11 = PHA and PHB gating

QECNTL (0xA074h) QE Counter Low R/W (0x00)

	7	6	5	4	3	2	1	0
RD	QECNT[7-0]							
WR	QECNTINI[7-0]							

QECNTH (0xA075h) QE Counter High R/W (0x00)

	7	6	5	4	3	2	1	0
RD	QECNT[15-8]							
WR	QECNTINI[15-8]							

Reading QECNT will return the current QE counter value. Writing QECNT will set the current count value. Writing QECNT is allowed only when QE is in the disabled state.

QEMAXL (0xA076h) QE Counter Low R/W (0x00)

	7	6	5	4	3	2	1	0
RD	QEMAX[7-0]							
WR	QEMAX[7-0]							

QEMAXH (0xA077h) QE Counter High R/W (0x00)

	7	6	5	4	3	2	1	0
RD	QEMAX[15-8]							
WR	QEMAX[15-8]							

QEMAX holds the maximum count of the QE counter. When the QEMAX count is reached, a TC event is triggered, and QE counter is reloaded.

8. PWM Controller

PWM controller provides programmable 6 channels 8-bit PWM center-aligned duty cycle outputs. The counting clock of PWM is programmable and the base frequency of the PWM is just the counting clock divided by 512 due to center-alignment. The duty cycle setting is always double buffered and minimum/maximum duty cycle is 0 and 255/256 respectively. PWM outputs are multiplexed with GPIO ports.

PWMCFG1 (0xA080h) PWM Clock Scaling Setting Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PWMEN	CS[6-0]						
WR	PWMEN	CS[6-0]						

PWMEN — PWM Controller Enable
PWMEN=0 clears the counter, resets the PWM state and all channel outputs are forced to 0.
PWMEN=1 allows normal running operation of PWM controller.

CS[6-0] — PWM Counting Clock Scaling
The counting clock is SYSCLK/4/(CS[6-0]+1) or PWM base frequency PWMCLK as
PWMCLK = SYSCLK/512/(CS6-0]+1) or
CS[6-0] = SYSCLK/512/PWMCLK – 1.
Assuming SYSCLK is 16MHz, the PWM base frequency ranges are from 250Hz to 32KHz.

PWMCFG2 (0xA081h) PWM Interrupt Enable and Flag Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-	-	ZINTEN	CINTEN	-	-	ZINTF	CINTF
WR	-	-	ZINTEN	CINTEN	-	-	ZINTF	CINTF

ZINTEN — Zero Interrupt Enable
ZINTEN=1 allows PWM Controller to generate an interrupt when the counter is 0.

CNTEN — Center Interrupt Enable
CINTEN=1 allows PWM Controller to generate interrupt when the counter is at the mid-value.

ZINTF — Zero Interrupt Flag
ZINTF is set to 1 by hardware to indicate a Zero interrupt has occurred. ZINTF must be cleared by software.

CINTF — Center Interrupt Flag
CINTF is set to 1 by hardware to indicate a Center interrupt has occurred. CINTF must be cleared by software.

PWMCFG3 (0xA082h) PWM Configuration 3 Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PRSEN	-	POL[5-0]					
WR	PRSEN	-	POL[5-0]					

PRSEN — Pseudo-Random Sequence Enable
PRSEN=1 will enable a pseudo random sequence to the PWM output width. This can be an effective way to reduce EMI for output. When PRSEN=1, the instantaneous duty cycle will be affected cycle by cycle but the average duty cycle remains the same.

POL[5-0] — Channel Polarity Control
POL[J] = 0 for normal polarity and POL[J]=1 for reverse polarity.

There are 6 independent PWMDTY registers to define the duty cycle. If PWMDTY = 0x00, the output is 0. If PWMDTY = 0xFF, the output duty cycle is 255/256. PWMDTY is always double buffered and is loaded to duty cycle comparator when the current counting cycle is completed.

PWM0DTY (0xA088h) PWM0 Duty Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PWM0DTY[7-0]							
WR	PWM0DTY[7-0]							

PWM1DTY (0xA089h) PWM1 Duty Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PWM1DTY[7-0]							
WR	PWM1DTY[7-0]							

PWM2DTY (0xA08Ah) PWM2 Duty Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PWM2DTY[7-0]							
WR	PWM2DTY[7-0]							

PWM3DTY (0xA08Bh) PWM3 Duty Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PWM3DTY[7-0]							
WR	PWM3DTY[7-0]							

PWM4DTY (0xA08Ch) PWM4 Duty Register R/W (0x00)

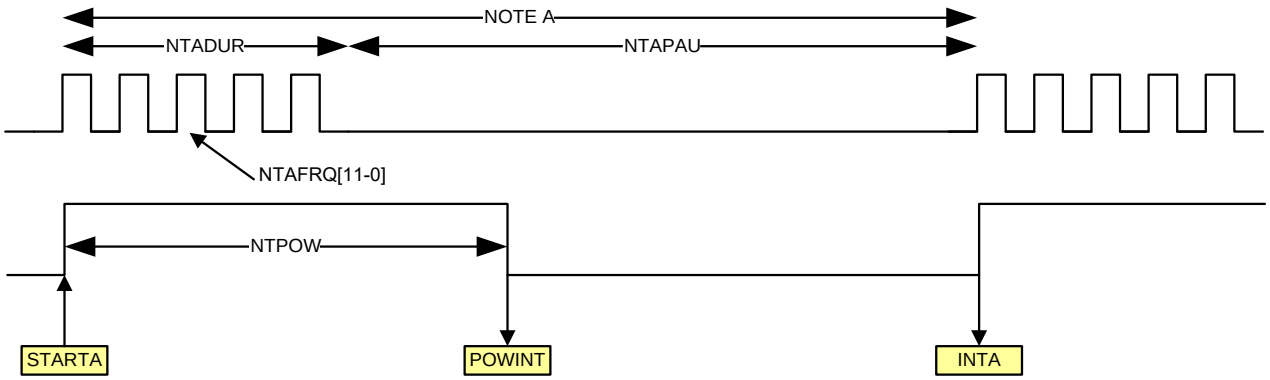
	7	6	5	4	3	2	1	0
RD	PWM4DTY[7-0]							
WR	PWM4DTY[7-0]							

PWM5DTY (0xA08Dh) PWM5 Duty Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PWM5DTY[7-0]							
WR	PWM5DTY[7-0]							

9. Buzzer and Melody Controller

The buzzer and melody controller can be used to generate a simple buzzer sound or a single tone melody. It contains a two note Ping-Pong buffers, each with programmable tone frequency, and duration/pause timer. The tone frequency is derived from SYSCLK divided by either 32 or 64, and the tone frequency is generated with resolution of 12-bit to support precision tone generation with wide octave span. The duration/pause timers can be programmed in 1ms/2ms/4ms/8ms steps. The two notes can be played sequentially once or can be played as Ping-Pong styles for melody. A POW (Power On Width) timer is also included with same time steps, and can be used to generate external power control of the buzzer element. POW timer is started when either note A or B is started.



NTAFRQL (0xA040h) Note A Frequency Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	NTAFRQ[7-0]							
WR	NTAFRQ[7-0]							

NTAFRQH (0xA041h) Note A Frequency Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-		-		NTAFRQ[11-8]			
WR	-		-		NTAFRQ[11-8]			

Tone frequency is SYSCLK/(32 or 64)/(NTAFRQ[11-0]+1).

NTADUR (0xA042h) Note A Duration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	NTADUR[7-0]							
WR	NTADUR[7-0]							

Tone duration is TU * NTADUR[7-0]

NTAPAU (0xA043h) Note A Pause Register Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	NTAPAU[7-0]							
WR	NTAPAU[7-0]							

Tone pause is TU * NTAPAU[7-0]

NTBFRQL (0xA044h) Note B Frequency Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	NTAFRQ[7-0]							
WR	NTAFRQ[7-0]							

NTBFRQH (0xA045h) Note B Frequency Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-		-		NTBFRQ[11-8]			
WR	-		-		NTBFRQ[11-8]			

NTBDUR (0xA046h) Note B Duration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	NTBDUR[7-0]							
WR	NTBDUR[7-0]							

NTBPAU (0xA047h) Note B Pause Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	NTBPAU[7-0]							
WR	NTBPAU[7-0]							

NTPOW (0xA049h) Note Power On Window Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	NTPOW [7-0]							
WR	NTPOW [7-0]							

NTPOW defines a timer after either STARTA or STARTB. It uses the same time unit as duration and pause. When the timer expires, it generates an interrupt by setting INTFP bit.

NOTETU (0xA04Ah) Note Time Unit Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TU[1-0]		-	TBASE	-	-	INTEPOW	INTFP
WR	TU[1-0]		-	TBASE	-	-	INTEPOW	INTFP

TU[1-0] Time Unit
TU[1-0] defines the time unit for duration and pause, and POW timer.
00 = 1msec
01 = 2msec
10 = 4msec
11 = 8msec

TBASE Tone Base Frequency Select
TBASE=0 uses SYSCLK/32 as base.
TBASE=1 uses SYSCLK/64 as base.

INTEPOW POW Timer Interrupt Enable

INTFP POW Interrupt Flag
INTFP is set by hardware when POW timer expires. It must be cleared by software.

BZCFG (0xA048h) Buzzer Configure Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	BZEN	BZPOL	INTENB	INTENA	INTFB	INTFA	BUSYB	BUSYA
WR	BZEN	BZPOL	INTENB	INTENA	INTFB	INTFA	STARTB	STARTA

BZEN Buzzer Control Enable
BZEN=1 enables the buzzer controller.
BZEN=0 disables the buzzer controller.

BZPOL BZOUT Polarity Setting
If BZPOL=1, BZOUT is inverted.
If BZPOL=0, BZOUT has normal polarity.

INTENB Note B End Interrupt Enable
INTENB=1 enables the note B end interrupt. The interrupt is triggered when note B playing is completed.

INTENA	Note A End Interrupt Enable INTENA =1 enables the note A end interrupt. The interrupt is triggered when note A playing is completed.
INTFB	Note B End Interrupt Flag INTFB is set to 1 by hardware if INTENB=1 and Note B playing ends. INTFB needs to be cleared by software writing 0.
INTFA	Note A End Interrupt Flag INTFA is set to 1 by hardware if INTENA=1 and Note A playing ends. INTFA needs to be cleared by software writing 0.
STARTB	Note B Start Command Writing STARTB=1 initiates a session output on the buzzer. Writing 0 to STARTB has no effect. STARTB is self-cleared when the note is completed.
STARTA	Note A Start Command Writing STARTA=1 initiates a session output on the buzzer. Writing 0 to STARTA has no effect. STARTA is self-cleared when the note is completed.
*** Note: If STARTA and STARTB are set to 1 at the same time, Note A is played first followed by Note B. Software can do this for a simple two-notes melody.	
BUSYB	Note B is playing busy Status BUSYB is set to 1 by hardware when the output is active playing note B.
BUSYA	Note A is playing busy Status BUSYA is set to 1 by hardware when the output is active playing note A.

10. Core Regulator and Low Voltage Detection

An on-chip serial regulator converts VDD into VDDC for internal circuit supply voltage. Typical value for VDDC is 1.5V at normal mode. In sleep mode, a backup regulator with a typical value of 1.4V supplies VDDC. The VDDC can be trimmed and calibrated, and the trim value for 1.5V is stored in IFB by the manufacturing test.

REGTRM (0xA000h) Regulator Trim Register R/W (0x80) TB protected

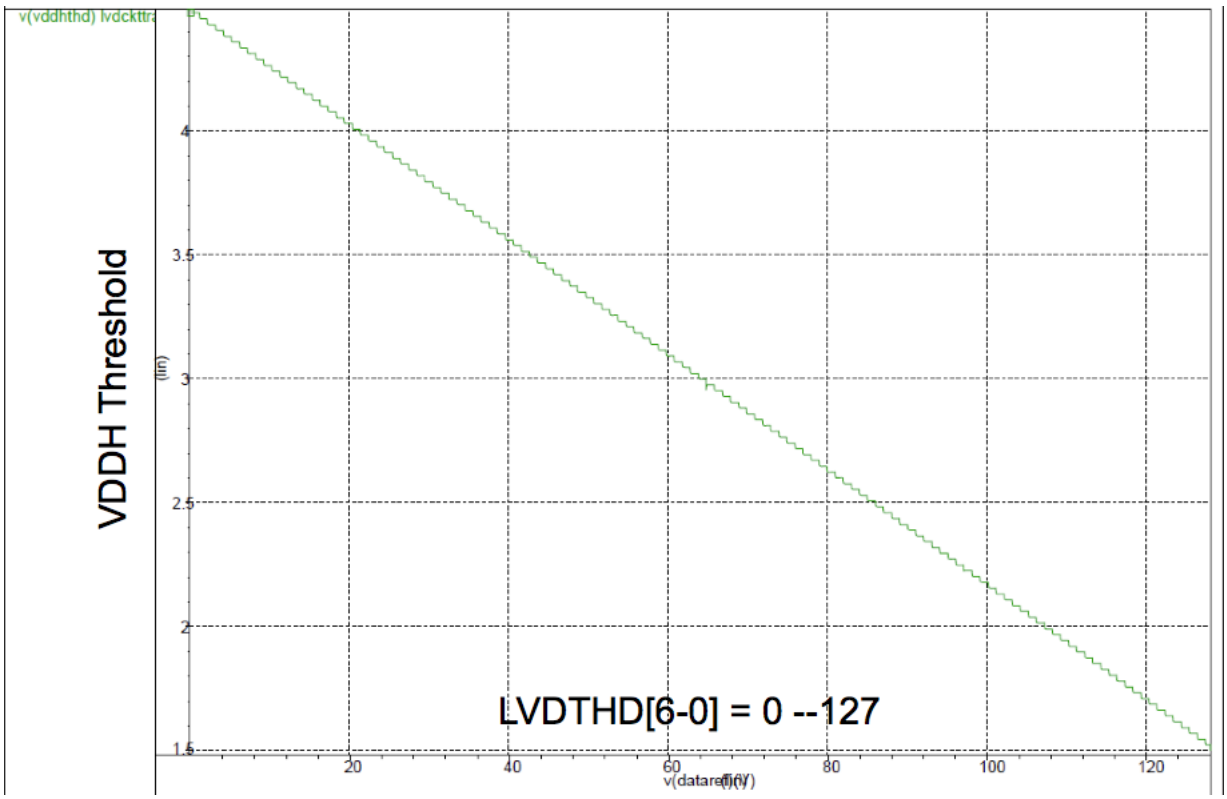
	7	6	5	4	3	2	1	0
RD	REGTRM[7-0]							
WR	REGTRM[7-0]							

10.1 Supply Low Voltage Detection (LVD)

The supply Low Voltage Detection (LVD) circuit detects VDD < VTH condition and can be used to generate an interrupt or reset condition. LVD defaults to a disabled state to save power. An enabled LVD circuit consumes about 100uA to 200uA. The LVDTHD[6-0] sets the compare threshold according to the following equation while LVDTHV is the detection voltage.

LVDTHV = VDDC * (1 + 2 * (1 – LVDTHD[6-0]/128)

= 1.5 + 3 * (1 – LVDTHD[6-0]/128), if VDDC is calibrated to 1.5V.



LVDCFG (A010h) Supply Low Voltage Detection Configuration Register R/W 00001000 TB Protected except bit 0 LVTIF

	7	6	5	4	3	2	1	0
RD	LVDEN	LVREN	LVTEN	LVDFLTEN	RSTNFLTEN	-	-	LVTIF
WR	LVDEN	LVREN	LVTEN	LVDFLTEN	RSTNFLTEN	-	-	LVTIF

LVDEN	LVD Enable bit. Set to turn on supply voltage detection circuits.
LVREN	LVR Enable bit. LVREN = 1 allows low voltage detection condition to cause a system reset.
LVTEN	LVT Enable bit. LVTEN = 1 allows low voltage detection condition to generate an interrupt.
LVDFLTEN	LVD Filter Enable LVDFLTEN = 1 enables a noise filter on the supply detection circuits. The filter is set at around 30usec.
RSTNFLTEN	RSTN Filter Enable

RSTNFLTEN = 1 enables a noise filter on the RSTN circuits. The filter is set at around 30usec. The filter is default on.

LVTIF — Low Voltage Detect Interrupt Flag
LVTIF is set by hardware when LVD detection occurs and must be cleared by software.

LVDTHD (A011h) Supply Low Voltage Detection Threshold Register R/W X1111111 TB Protected

	7	6	5	4	3	2	1	0
RD	-	LVDTHD6	LVDTHD5	LVDTHD4	LVDTHD3	LVDTHD2	LVDTHD1	LVDTHD0
WR	-	LVDTHD6	LVDTHD5	LVDTHD4	LVDTHD3	LVDTHD2	LVDTHD1	LVDTHD0

LVDTHD = 0x00 will set the detection threshold at its minimum, and LVDTHD = 0x7F will set the detection threshold at its maximum.

LVDHYS (A012h) Supply Low Voltage Detection Threshold Hysteresis Register R/W 00000000 TB Protected

	7	6	5	4	3	2	1	0
RD	LVDHYEN	LVDHYS6	LVDHYS5	LVDHYS4	LVDHYS3	LVDHYS2	LVDHYS1	LVDHYS0
WR	LVDHYEN	LVDHYS6	LVDHYS5	LVDHYS4	LVDHYS3	LVDHYS2	LVDHYS1	LVDHYS0

To ensure a solid Low Voltage detection, a digitally controlled hysteresis is used. If LVDHYEN=1, LVD is asserted as a new threshold defined by LVDHYS[6-0] instead of LVDTHD[6-0]. In typical applications, LVDHYS[6-0] should be set to be smaller than LVDTHD[6-0], such that the recovery voltage is higher than the detection voltage.

11. IOSC and SOSC

11.1 IOSC 16MHz

An on-chip 16MHz Oscillator with low-temperature coefficient provides the system clock to the CPU and other logic. IOSC uses VDDC as the power supply and can be calibrated and trimmed. The accuracy of the frequency is +/- 2% within the operating conditions. This oscillator is stopped and enters into stand-by mode when CPU is in STOP/SLEEP mode and resumes oscillation when CPU wakes up.

IOSCITRM (0xA001h) IOSC Coarse Trim Register R/W 0x01 TB Protected

	7	6	5	4	3	2	1	0
RD	SSC[3-0]				SSA[1-0]		ITRM[1-0]	
WR	SSC[3-0]				SSA[1-0]		ITRM[1-0]	

SSC[3-0] SSC[3-0] defines the spread spectrum sweep rate. If SSC[3-0] = 0000, then the spread spectrum is disabled.

SSA[1-0] SSA[1-0] defines the amplitude range of spread spectrum frequency. The frequency is changed by adding SSA[1-0] range to actual IOSCVTRM[7-0].

SSA[1-0] = 11, +/- 32

SSA[1-0] = 10, +/- 16

SSA[1-0] = 01, +/- 8

SSA[1-0] = 00, +/- 4

ITRM[1-0] ITRM[1-0] is the coarse trimming of the IOSC.

IOSCVTRM (0xA002h) IOSC Fine Trim Register R/W 0x80 TB Protected

	7	6	5	4	3	2	1	0
RD	IOSCVTRM[7-0]							
WR	IOSCVTRM[7-0]							

This register provides fine trimming of the IOSC frequency. The higher the value of IOSCVTRM, the lower the frequency is.

The manufacturer trim value is stored in IFB and is trimmed to 16MHz. The user program provides the freedom to set the IOSC at a preferred frequency if the program is able to calibrate the frequency. Once set, the IOSC frequency has accuracy deviation within +/- 2% over the operation conditions. The following lists the range of the typical IOSC frequency for each trimming setting.

ITRM[1-0]=00, F_IOSC = 16.0MHz – 14.0MHz – 12.0MHz (VTRM[7-0]= 00 - 80 - FF)

ITRM[1-0]=01, F_IOSC = 18.5MHz – 16.5MHz – 14.0MHz (VTRM[7-0]= 00 - 80 - FF)

ITRM[1-0]=10, F_IOSC = 21.5MHz – 18.5MHz – 16.0MHz (VTRM[7-0]= 00 - 80 - FF)

ITRM[1-0]=11, F_IOSC = 24.0MHz – 20.5MHz – 17.5MHz (VTRM[7-0]= 00 - 80 - FF)

11.2 SOSC 128KHz

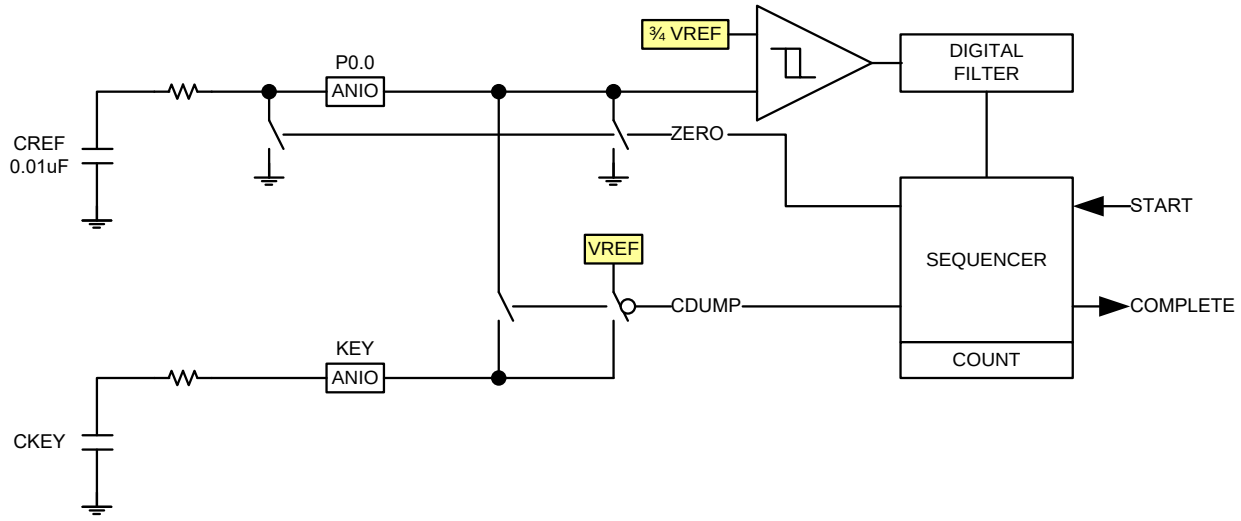
An ultra-low power slow oscillator of 128KHz is also included. SOSC consumes less than 0.5uA from VDDC and is always enabled. The system uses SOSC32KHz = SOSC/4 = 32KHz for system clock, and for wake-up timer T5, and WDT2/WDT3. SOSC is not very accurate and varies chip to chip, but it is relatively stable toward variations of power supply and temperature. Therefore, software can use IOSC to calibrate SOSC through SOCTRM[4-0].

SOSCTRM (0xA007h) SOSC Trim Register R/W (0x10) TB Protected

	7	6	5	4	3	2	1	0
RD	-			SOSCTRM[4-0]				
WR	-	-		SOSCTRM[4-0]				

12. Touch Key Controller II

For a different EMI environment, a second touch-key controller is implemented. This touch key controller is based on switched capacitance. The block diagram is shown in the following. P0.0 is used to connect an external reference capacitor (typically 0.01uF). The sense capacitor is connected through the ANIO selection of the IOCELL. The software will issue a start command to start the detection. The CREF is first zeroed by turning on a switch to VSS and then released. CKEY starts repetitive cycles of being charged to VREF and dumps the charges onto CREF. At the same time the sequencer keeps counting the cycles. When CREF is charged to ¾ VREF, the detection is completed, and the number of cycles is stored in COUNT register. Then an interrupt is issued, and software can read out the COUNT for processing. The interrupt vector is shared with Touch Controller I. The touch key controller II operates on IOSC or reduced frequencies of IOSC.



TK2CFGA (0xA008h) Touch Key Controller II Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TKCIIEN	PSREN	REFSEL	VTHSEL	CDTIME[3-0]			
WR	TKCIIEN	PSREN	REFSEL	VTHSEL	CDTIME[3-0]			

TKCIIEN	Touch Key Controller II Enable
PSREN	Pseudo Random Mode Enable
REFSEL	VREF Selection **VREF=0 uses VDDC as the reference. VDDC should always be used.** VREF=1 uses VDD as the reference.
VTHSEL	Comparator Threshold VTH=0 uses ¾ VREF as the threshold. VTH=1 uses ½ VREF as the threshold.
CDTIME[3-0]	Charge and Dump Base Time Setting CDTIME[3-0] determines the base time for charge and dump duration. The duration is SYSCLK period * (CDTIME[3-0]+1).

TK2CFGB (0xA009h) Touch Key Controller II Configuration Register A R/W (0x00)

	7	6	5	4	3	2	1	0
RD	IOEN	ZERO[2-0]			CFIL[3-0]			
WR	IOEN	ZERO[2-0]			CFIL[3-0]			

IOEN	IOCELL NMOS for Zero CREF Enable This controls ZERO[2-0]*128 SYSCLK IOCELL NMOS (12mA@3V) turn-on.
ZERO[2-0]	CREF is cleared to 0V Duration. The internal switch (2mA@3V) is turned on for (4+128*(ZERO[2-0]+1) +1) SYSCLK duration to force CREF to 0V at the start of conversion.
CFIL[3-0]	Comparator Filter Delay The analog output of the comparator is filtered by CFIL[3-0]. The filter output is asserted when preceding CFIL[3-0] samples (sampled by SYSCLK) are all ones.

TK2CMD (0xA00Ah) Touch Key Controller II Command and Interrupt Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TKCINTEN	TKCCNT[17-16]		ASHIFT	RPT[1-0]		INTF	BUSY
WR	TKCINTEN	-		ASHIFT	RPT[1-0]		INTF	START

TKCINTEN Touch Key Controller II Interrupt Enable
TKCCNTII[17-16] Touch Key Controller II Count MSB
ASHIFT Automatic Shift for RPT
RPT[1-0] Repeat Conversion Cycle Count
RPT[2-0] defines the repetition count of the conversion cycles. The TKCCNTII is accumulated with multiple conversions.
00 = 1
01 = 2
10 = 4
11 = 8
INTF Interrupt Flag
INTF is set by hardware and must be cleared by software.
START Start Conversion
Set START=1 will initiate the conversion sequence and it is self-cleared when conversion is completed.
BUSY Busy Status

TK2CNTL (0xA00Bh) Touch Key Controller II Count Register L R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-							
WR	TK2CNT[7-0]							

TK2CNTH (0xA00Ch) Touch Key Controller II Count Register H R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-							
WR	TK2CNT[15-7]							

TKCCNTII is cleared when each START command is issued. And it contains the charge conversion cycle count when the conversion is done.

13. Touch Key Control III

TK3 is an enhanced TK2 implementation with differential dual slope operations. The capacitance to time conversion goes through two phase of charge transfer: One is charging up and the other one is discharging down using two thresholds equally spaced from ½ VDDC. Each charge transfer is obtained by subtraction of charge on internal reference capacitance and key capacitance. The difference of charge/discharge counting behavior is used to determine the key capacitance change in the ratio of internal capacitance. Better noise immunity from power, ground noise and common-mode noise is achieved by dual slope operation. Also better S/N can also be achieved because only differential charge is used for transfer, and the internal capacitance exhibits better temperature and environmental stability which makes the conversion result less sensitive to the changes of temperature and environment.

CREF, the integration capacitor of the charge transfer, is connected to P04 through ANIO multiplexer and CKEY is connected to other GPIO through multiplexer. A replica signal of CKEY is provided through a buffer and routed out as SHIELD through GPIO. The shield signal can be used to cancel mutual capacitance effect from neighboring signal trace of the detected key and provides better noise immunity against moisture or water.

The total count of charging period is recorded as TKHDT[15-0] and total count of discharge count period is recorded as TKLDT[15-0]. To detect a key press capacitance change, the value of TKHDT[15-0] or TKLDT[15-0] can be processed by software and compared with an average non-pressed count. For high frequency noise rejection, the hardware includes a pseudo-random sequence that randomizes the charge and discharge timing. To further enhance the S/N ratio, the conversion can be set to accumulate up to 16 times by hardware and this effectively increases the resolution to 20-bit by trading off the conversion time. A slow moving average of the duty count value is stored in TKBASE[15-0] and software can use this for baseline calculation to automatically compensate for the environment change.

TK3CFGA (0xA018h) TK3 Configuration Register A R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TK3EN	TKCS[2-0]			SHIELDEN	TKIEN	TKLPM	AUTO
WR	TK3EN	TKCS[2-0]			SHIELDEN	TKIEN	TKLPM	AUTO

TK3EN — TK3 Enable
TK3EN=0 disables the TK3 circuits and clears all states.
TK3EN=1 for TK3 normal operations

TKCS[2-0] — TK3 Clock Select
TKCS[2-0]=000 SYSCLK/2
TKCS[2-0]=001 SYSCLK/4
TKCS[2-0]=010 SYSCLK/6
TKCS[2-0]=011 SYSCLK/8
TKCS[2-0]=100 SYSCLK/10
TKCS[2-0]=101 SYSCLK/16
TKCS[2-0]=110 SYSCLK/32
TKCS[2-0]=111 SOSC/2
SOSC/2 should be used for sleep mode auto wakeup. Typical SOSC/2 is 64KHz.

SHIELDEN — Shield Output Buffer Enable
SHIELDEN=1 enables the shield signal buffer. The buffer consumes about 200uA when enabled.

TKIEN — TK3 Interrupt Enable
TKIEN=1 enables the TK3 interrupt. TK3 interrupt is generated when a counting sequence is completed (including the repeat count if RPT[1-0] is not 00). Interrupt and wakeup are also generated when TKIEN=1 and AUTO=1 after auto-detection threshold is met.
When TK3 interrupt is generated, TKIF is also set to 1 by hardware.

TKLPM — TK3 Low Power Mode
TKLPM=0 for normal mode operations
TKLPM=1 puts the comparator into ultra-low power mode and should be used in auto wakeup power saving mode. In this mode, TKCLK should use SOSC/2 slow clock.

AUTO — Auto Wake Up Mode
AUTO=1 enables auto detect mode. In auto mode, the current duty count register value is compared with baseline plus threshold (either absolute or relative). If duty count value is higher, an interrupt and wakeup are generated.

AUTO=0 enables normal detect mode. In normal mode, writing START with "1" initiates a conversion sequence, and when the duty count is obtained, an interrupt is generated.

TK3CFGB (0xA019h) TK3 Configuration Register B R/W (0x00)

	7	6	5	4	3	2	1	0
RD	RPT[1-0]		INI[1-0]		ASTDLY[1-0]		LFNF[1-0]	
WR	RPT[1-0]		INI[1-0]		ASTDLY[1-0]		LFNF[1-0]	

RPT[1-0] Repeat Sequence Count
00 = No Repeat
01 = 4 times
10 = 8 times
11 = 16 times

INI[1-0] Initial Settling Delay
INI[1-0] defines the number of TKCLK period for initial settling of CREF. The delay is (INI[1-0] + 1) *4*TKCLK.

ASTDLY[1-0] Auto Mode Start Delay
STDLY[1-0] inserts an inter-sequence idle time of (ASTDLY[1-0]+1) * 256 TKCLK at each sequence start. This delay allows the stabilization time from normal mode to sleep mode.

LFNF[1-0] Low Frequency Noise Filter Setting
00 = disables LFNF
Injection noise longer than LFNF[1-0]*8 time is ignored.
In the presence of such noise, the cycle count still continues. The end result is that the sum of DUTYL and DUTYH will not equal to cycle count.

TK3CFGC (0xA01Ah) TK3 Configuration Registers C R/W (0x00)

	7	6	5	4	3	2	1	0
RD	SLOW[1-0]		CYCLE[2-0]			BASEINI	THDSEL	AUTOLFEN
WR	SLOW[1-0]		CYCLE[2-0]			BASEINI	THDSEL	AUTOLFEN

SLOW[1-0] Baseline Slow Moving Average setting
00 = 32 average
01 = 64 average
10 = 128 average
11 = 256 average
The duty value is averaged by SLOW[1-0] conversion and updated to BASELINE register through moving average.

CYCLE[2-0] Cycle Count of each conversion sequence
000 = 1024
001 = 2048
010 = 4096
011 = 8192
100 = 12288
101 = 16384
110 = 32768
111 = 65536
The cycle count is each sequence cycle count. And it is repeated if RPT is not 0.
Conversion always ends with the defined cycle count.

BASEINI Baseline Initial Value
If BASEINI=1, the first DTYL count after entering auto mode is loaded to BASELINE register as its initial value to start moving average.
If BASEINI=0, the value written in BASELINE before entering auto mode is used as the initial value to start moving average.

THDSEL Threshold Value Setting
THDSEL=0 uses TKTHD[15-0] as the threshold to compare with TKLDT[15-0] to generate the interrupt and wakeup.

THDSEL=1 uses TKTHD[15-0] + TKBASE[15-0] as the threshold to compare with TKLDT[15-0] to generate the interrupt and wakeup.

AUTOLFEN — Low Frequency Noise Filtering in Auto mode
If AUTOLFEN=0, low frequency noise filtering in Auto mode is disabled.
If AUTOLFEN=1, low frequency noise filtering in auto mode is enabled.
The low noise filtering status flag is still valid regardless of AUTOLFEN setting. Software can determine whether to discard the current conversion result by checking LFNF flag.

TK3CFGD (0xA01Bh) TK3 Configuration Registers D R/W (0x00)

	7	6	5	4	3	2	1	0
RD	CCHG[2-0]			ASTDLYEN	PSRDEN	LFNF	TKIF	BUSY
WR	CCHG[2-0]			ASTDLYEN	PSRDEN	LFNF	TKIF	START

CCHG[2-0] — Internal Reference Capacitance Select
000 = 10pF
001 = 20pF
010 = 30pF
011 = 40pF
100 = 50pF
101 = 60pF
110 = 70pF
111 = 80pF

ASTDLYEN — Auto Start Delay Enable
ASTDLYEN=1 enables ASTDLY[1-0] delay start for auto mode.
ASTDLYEN=0 disables ASTDLY[1-0] delay.

PSRDEN — Pseudo Random Sequence Enable
PSRDEN=1 enables the random sequence in conversion.
PSRDEN=0 disables the random sequence in conversion.

LFNF — Low Frequency Noise Detection Flag
LFNF is set by hardware if a Low Frequency Noise is detected in the present conversion.
LFNF needs to be cleared to “0” by software.

TKIF — TK3 Interrupt Flag
TKIF is set by hardware when a TK3 interrupt occurred by either conversion sequence completion or a valid detection in auto mode. TKIF needs to be cleared to “0” by software.

START — Start Conversion
Writing “1” into START initiates the conversion sequence. It is cleared by hardware when conversion is complete. Writing “1” to AUTO also starts the conversion in auto mode.

BUSY — Conversion Status
BUSY is set to 1 by hardware and that indicates the conversion sequences are still running.

TK3HDTYL (0xA01Ch) TK3 High Duty Count Register L RO (0x00)

	7	6	5	4	3	2	1	0
RD	TK3HDTY[7-0]							
WR	-							

TK3HDTYH(0xA01Dh) TK3 High Duty Count Register H RO (0x00)

	7	6	5	4	3	2	1	0
RD	TK3HDTY[15-8]							
WR	-							

TK3LDTYL (0xA01Eh) TK3 Low Duty Count Register L RO (0x00)

	7	6	5	4	3	2	1	0
RD	TK3LDTY[7-0]							
WR	-							

TK3LDTYH(0xA01Fh) TK3 Low Duty Count Register H RO (0x00)

	7	6	5	4	3	2	1	0
RD	TK3LDTY[15-8]							
WR	-							

TK3BASEL (0xA028h) TK3 Baseline Register L R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TK3BASE[7-0]							
WR	TK3BASE[7-0]							

TK3BASEH (0xA029h) TK3 Baseline Register H R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TK3BASE[15-8]							
WR	TK3BASE[15-8]							

TK3THDL (0xA02Ah) TK3 Threshold Register L R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TK3THD[7-0]							
WR	TK3THD[7-0]							

TK3THDH (0xA02Bh) TK3 Threshold Register H R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TK3THD[15-8]							
WR	TK3THD[15-8]							

TK3PUD (0xA02Ch) TK3 DC Pull-Up/Pull-Down Control Register H R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PUDIEN	PUDREN	-	-	PUD[3-0]			
WR	PUIDEN	PUDREN	-	-	PUD[3-0]			

TK3PUD is to configure a constant DC pull-up/pull-down on CREF to allow high capacitance touch-key detection. A DC pull-up/pull-down can compensate for the equivalent resistance caused by a high capacitance key. Thus, connecting a switching current source or resistor can maintain touch key detection sensitivity.

PUDIEN	Pull-up/Pull-down DC Current Enable
PUDREN	Pull-up/Pull-down DC Resistor Enable
PUD[3-0]	Pull-up/Pull-down Selection For DC current, PUD[3-0] enables 8uA/4uA/2uA/1uA current source. For Resistor, PUD[3-0] enables 5K/10K/20K/40K resistor.

14. GPIO and Pin Interrupt

Each IO pin has a configurable IO buffer that can meet various interface requirements. The GPIO pins can be configured as external interrupt input pins or wake-up pins. Each port has edge detection logic and latch for rising and falling edge detections. During hardware reset and later-on time, the IO buffer is in a high impedance state with all drives disabled.

IOCFGO (0xA100h – 0xA10Fh, 0xA130h – 0xA13Fh) IO Buffer Output Configuration Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	HIDRV	PDRVEN	NDRVEN	OPOL	ANEN2	ANEN1	PUEN	PDEN
WR	HIDRV	PDRVEN	NDRVEN	OPOL	ANEN2	ANEN1	PUEN	PDEN

HIDRV — High-Speed Drive Enable
HIDRV=1 enables high-speed drive. This reduces output rise and fall time with stronger drive during transient.
HIDRV=0 configures the output buffer with slower edge transitions. HIDRV=0 should be used for EMI sensitive applications.
Both HIDRV=1 or HIDRV=0 have the same DC drive capabilities.

PDRVEN — Output PMOS driver enabled. Set this bit to enable the PMOS of the output driver. DISABLE is the default value.

NDRVEN — Output NMOS driver enable. Set this bit to enable the NMOS of the output driver. DISABLE is the default value.

OPOL — Output Polarity Control
Output buffer data polarity control

ANEN1 — Analog MUX 1 enables control. Set this bit to connect the pin to the internal analog peripheral. DISABLE is the default value.

ANEN2 — Analog MUX 2 enables control. Set this bit to connect the pin to the internal analog peripheral. DISABLE is the default value.

PUEN — Pull up resistor control. Set this bit to enable pull-up resistor connection to the pin. The pull-up resistor is approximately 6K Ohm. DISABLE is the default value.

PDEN — Pull down resistor control. Set this bit to enable pull-down resistor connection to the pin. The pull-down resistor is approximately 6K Ohm. DISABLE is the default value.

IOCFGI (0xA110h – 0xA11Fh, 0xA140h – 0xA14Fh) IO Buffer Input Configuration Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PI1EN	PI0EN	RIF	FIF	INEN	IPOL	DSTAT	INSTAT
WR	PI1EN	PI0EN	RIEN	FIEN	INEN	IPOL	DBN[1-0]	

PI1EN — Pin Interrupt 1 Enable

PI0EN — Pin Interrupt 0 Enable

RIEN — Rising Edge Pin Interrupt Enable

RIF — Rising Edge Pin Interrupt Flag
RIF is set to 1 by hardware after either a PI1 or PI0 rising edge interrupt has occurred. RIF must be cleared by software writing RIEN with “0”. RIEN needs to be enabled if the next rising edge interrupt is required.

FIEN — Falling Edge Pin Interrupt Enable

FIF — Falling Edge Pin Interrupt Flag
FIF is set to 1 by hardware after either a PI1 or PI0 falling edge interrupt has occurred. FIF must be cleared by software writing FIEN with “0”. FIEN needs to be enabled if the next falling edge interrupt is required.

INEN — Input Buffer Enable
INEN=1 enables the input buffer.
INEN=0 disables the input buffer. In the disabled state, the output of input buffer is logic 0. If input is floating or not solid 0 and 1 voltage level, DC current may flow in the input buffer. Disabling input buffer can remove DC leakage of input buffer due to this reason.

IPOL — Input Polarity
IPOL=1 reverses the input logic. IPOL=0 is for normal logic polarity.

DBNST — Real Time Status after De-bounce. DBNST bit is read only.

	The de-bounced input is used for generating interrupt, as well as all other multi-function inputs including PORT registers. The non-de-bounced input can only be read through INSTAT bit.
INSTAT	Real Time Status of Input Buffer. INSTAT is read only.
DBN[1-0]	De-Bounce Time Setting 00 – OFF 01 – 4 SOSC32KHz (130usec) 10 – 16 SOSC32KHz (530usec) 11 – 64 SOSC32KHz (2msec)

MFCFGxx (0xA120 – 0x A12Fh, 0xA150h – 0xA15Fh) Port Multi-Function Configuration Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	MFCFG[7-0]							
WR	MFCFG[7-0]							

Please see PIN OUT section for the description of each port multi-function selection.

15. Information Block IFB

There are two IFB blocks and each one contains 512 x 16 bit information. The address 0x000h to 0x03Fh in first IFB is used to store manufacturer information. Address 0x040 is for wait time of boot code, and 0x041 to 0x043 are used for boot code. The first IFB can be erased only in Writer Mode and can be written using Flash Controller for address beyond 0x40. This is to protect any alteration of the manufacturer and calibration data. The 2nd IFB is open for erase/write for user access. The following table shows the contents of the first IFB for the manufacturer data. Please note that these are in lower LSB bytes. The upper MSB byte contains its corresponding ECC code.

ADDRESS	TYPE	DESCRIPTION
00 – 01	M	IFB Version
02 – 07	M	Product Name
08 - 09	M	Package and Product Code
0A – 0B	M	Product Version and Revision
0C	M	Flash Memory Size
0D	M	SRAM Size
0E – 0F	M	Customer Specific Code
10	M	CP1 Information
11	M	CP2 Information
12	M	CP3 Version
13	M	CP3 BIN
14	M	FT Version
15	M	FT BIN
16 - 1B	M	Last Test Date
1C – 1D	M	Boot Code Version
1E	M	Boot Code Segment
1F	M	Checksum for 0x00 – 0x1E
20	M	REGTRM value for 1.55V
21	M	IOSC ITRM value for 16MHz 5V
22	M	IOSC VTRM value for 16MHz 5V
23	M	LVDTHD value for detection of 4.0V
24	M	LVDTHD value for detection of 3.0V
25	M	IOSC ITRM value for 16MHz 3.3V
26	M	IOSC VTRM value for 16MHz 3.3V
27	M	Reserved
28	M	Reserved
29	M	Reserved
2A	M	Reserved
2B – 2D	M	Reserved
2E – 2F	M	Internal Reference LSB/MSB
30	M	SOSC 128KHZ Trim Value
31 – 33	M	Reserved
34	M	Timer 0 High TRIM *
35	M	Timer 0 Low TRIM *
36 – 38	M	Reserved
39	M	Checksum for 0x20 – 0x39
3A – 3F	M	Retention Value
40	M/U	Boot Code Wait Time. Boot code uses this byte to determine the ISP wait-time. This wait-time is necessary for a stable ISP. After the user program is downloaded, the wait time can be reduced to minimize power-on time. Each “1” in bit [1-0] constitutes 1 second, bits [3-2] constitutes 2 second and bit [7] is check of I2CSCL2. For example, 0b10000111 is a 4-second wait time and also checks

		I2CSCL2 pad status. If I2CSCL2 is low, then wait time of 6 second is used regardless of bit [3-0] setting. The maximum wait time is 6 second, and minimum wait time is 0 second.
41 – 43	M	Reserved
44 - 1FF	U	User One-Time Programmable Space

16. Writer Mode

Writer Mode (WM) is used by the manufacturer or by users to program the flash (including IFB) through a dedicated hardware (Writer or Gang Writer). Under this setup, only WM related pins should be connected, and all other unused pins left floating. Writer mode follows a proprietary protocol and is not released to general users. Users must obtain it through a formal written request to the manufacturer and must sign a strict Non-Disclosure-Agreement. The Writer Mode provides the following commands.

ERASE Main Memory

ERASE Main Memory and IFB

READ AND VERIFY Main Memory (8-Byte)

WRITE BYTE Main Memory

READ BYTE IFB

WRITE BYTE IFB

Fast Continuous WRITE

Fast Continuous READ

The writer mode is protected against code piracy. The default state of the device is locked writer mode. Only ERASEMM and ERASEMMIFB, and READVERIFYMM commands can be executed. It can be unlocked by READVERIFYMM the range of 0x06F8 to 0x06FF. These locations contain an 8-byte security key that user can place to secure the e-Flash contents. The probability of guessing the key is 1 in 2^64 = 1.8E19. Since each trial of READVERIFYMM takes 10usec, it takes about 6E6 years to exhaust the combinations. If the key is unknown, a user can choose to issue the ERASEMM command then fully erase the entire contents (including the key). Once fully erased, all data in the flash is 0xFF, and it can be successfully unlocked by READVERIFYMM with 8-bytes of 0xFF. The users must not erase the information in IFB. And the user should not modify the manufacturer data. Any violation of this results in the void of manufacturer warranty. The following pins are used for e-Flash writer mode. P10 is optional.

PIN	IO	Description	Function
P22	O	Flash serial data output.	SDO
P21	I	Flash serial data input	SDI
P20	I	Flash serial clock input.	SCLK
P17	I	Flash serial port enable, low active	SCE
RSTN	I	Write mode entry input using timing sequence	RSTN
P23	O	TBIT status output	TBIT
VDD	I	Power supply for DUT	VDD
VSS	I	Ground supply for DUT	VSS

17. Boot Code and In-System Programming

After production testing of the packaged devices, the manufacturer writes the manufacturer information and calibration data in the IFB. At the last stage, it writes a fixed boot code in the main memory residing from 0x7000 to 0x7FFF. The boot code is executed after resets.

The boot code first reads IFB's wait time setting and scans the I^2C slave for any In-System-Programming request during the wait time duration. If any valid request occurs during the scan, the boot-code proceeds to follow the request and performs the programming from the host. Otherwise, the boot code jumps to 0x0000 after the wait time is expired. The default available ISP commands are as below.

UNLOCK

DEVICE NAME

BOOTC VERSION

READ AND VERIFY Main Memory (8-Byte)

ERASE Main Memory excluding Boot Code

ERASE SECTOR Main Memory

WRITE BYTE Main Memory

SET ADDRESS

CONTINUOUS WRITE

CONTINUOUS READ

READ BYTE IFB

WRITE BYTE IFB

Like writer mode, ISP is in the locked state at default. No command is accepted under a locked state. To unlock the ISP, an 8-byte READ and VERIFY of 0x06FF8 to 0x06FFF must be successfully executed. Thus, the default ISP boot program provides similar code security as the Writer mode.

18. Electrical Specifications

18.1 Absolute Maximum Ratings

SYMBOL	PARAMETER	RATING	UNIT	NOTE
VDD	Supply Voltage	5.5	V	
TA	Ambient Operating Temperature	-40 – 125	°C	
TSTG	Storage Temperature	-65 – 150	°C	

18.2 Recommended Operating Condition

SYMBOL	PARAMETER	RATING	UNIT	NOTE
VDD	Supply Voltage for IO and 1.5V regulator	2.5– 5.5	V	
TA	Ambient Operating Temperature	-40 – 125	°C	

18.3 DC Electrical Characteristics (VDD=2.5V to 5.5V TA=-40°C to 125°C)

SYMBOL	PARAMETER	MIN	TYP	MAX	UNIT	NOTE
Power Supply Current						
IDD Normal	Total IDD through VDD at 16MHz	-	7	-	mA	
IDD versus Frequency	Total IDD Core Current versus Frequency	-	150	-	µA/ MHz	
IDD, Stop	IDD, stop mode	-	150	-	µA	Main regulator on
IDD, Sleep	IDD, sleep mode, 25°C	-	1	3	µA	
	IDD, sleep mode, 125°C	-	15	30	µA	Main regulator off
GPIO DC Characteristics						
VOH,4.5V	Output High Voltage 1 mA	-	-0.2	-0.4	V	Reference to VDD
VOL,4.5V	Output Low Voltage 8 mA	-	0.3	0.5	V	Reference to VSS
VOH,3.0V	Output High Voltage 1 mA	-	-0.3	-0.5	V	Reference to VDD
VOL,3.0V	Output Low Voltage 8 mA	-	0.3	0.5	V	Reference to VSS
IIOT	Total IO Sink and Source Current	-100	-	100	mA	
VIH	Input High Voltage	¾VDD	-	-	V	
VIL	Input Low Voltage	-	-	¼VDD	V	
VIHYS	Input Hysteresis	-	600	-	mV	
RPU	Equivalent Pull-Up resistance	-	5K	-	Ohm	
RPU,RSTN	RSTN Pull-Up resistance	-	5K	-	Ohm	
RPD	Equivalent Pull-Down Resistance	-	5K	-	Ohm	
REQAN1	Equivalent ANIO Switch Resistance, 3.3V	-	220	-	Ohm	ANIO1 Switch
	Equivalent ANIO Switch Resistance, 5V	-	70	-	Ohm	ANIO1 Switch
REQAN2	Equivalent ANIO Switch Resistance, 3.3V	-	220	-	Ohm	ANIO2 Switch
	Equivalent ANIO Switch Resistance, 5V	-	70	-	Ohm	ANIO2 Switch
VDDC Characteristics						
VDDCN	Normal Core Voltage 1.55V (Calibrated)	1.45	1.55	1.65	V	Normal Mode
VDDCS	Sleep Core Voltage 1.5V	-	1.40	-	V	Sleep Mode
Low Supply (VDD) Voltage Detection						
VDET	Detection Range	2.0	-	4.8	V	
VDETHYS	Detection Hysteresis	-	100	-	mV	

18.4 AC Electrical Characteristics (VDD =2.5V to 5.5V TA=-40°C to 125°C)

SYMBOL	PARAMETER	MIN	TYP	MAX	UNIT	NOTE
Supply Timing						

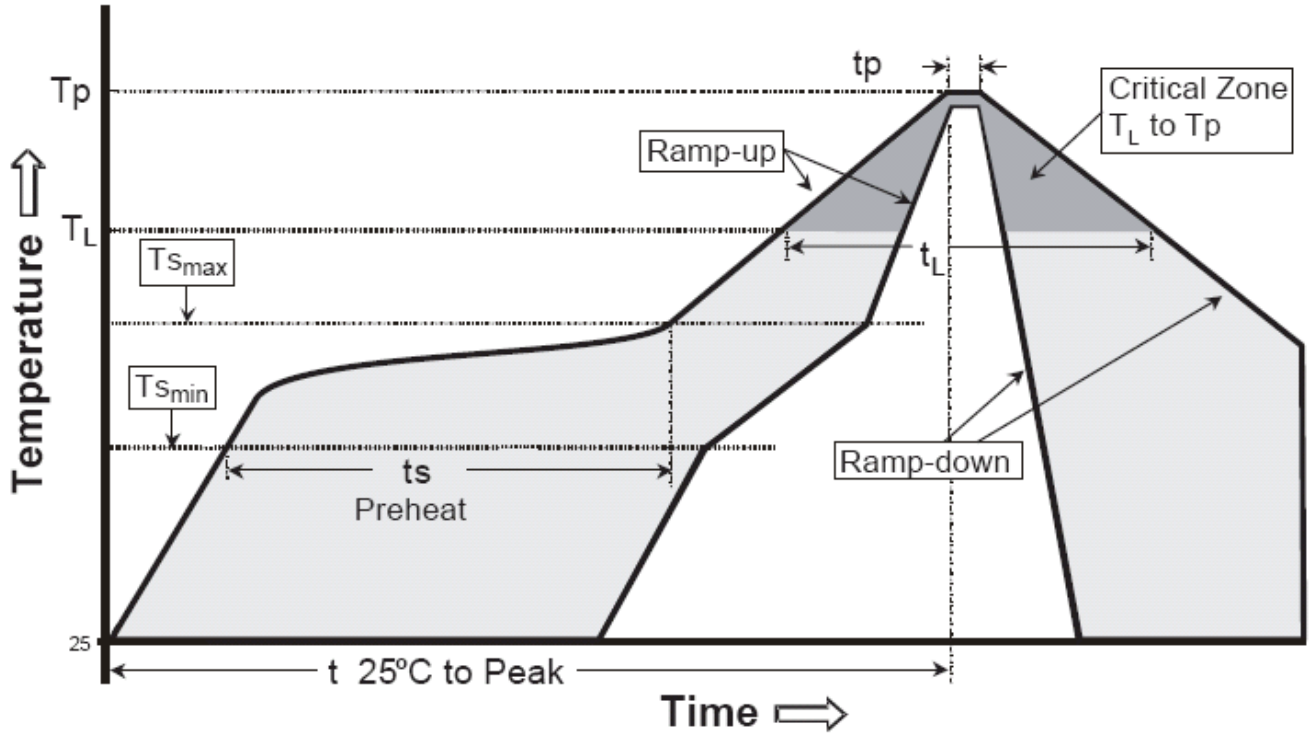
SYMBOL	PARAMETER	MIN	TYP	MAX	UNIT	NOTE
TSUPRU	VDD Ramp Up time	1	-	50	msec	
TSUPRD	VDD Ramp Down Time	-	-	50	msec	
TPOR	Power On Reset Delay	-	5	-	msec	
IOSC						
FIOSC	IOSC Calibrated 16MHz	-1	0	+1	%	
	IOSC Startup Time	-	-	1	µsec	
	Temperature and VDD variation 85°C	-2	0	+2	%	
	Temperature and VDD variation 125°C	-3	0	+3	%	
	Stable Time and Reset for IOSC after power up	2	-	-	msec	After VDD > 2.0V
SOSC						
FSOSC	Slow Oscillator frequency	-	128	-	KHz	
GPIO Timing						
TPD3 ++	Propagation Delay 3.3V No load	-	6	-	nsec	
	Propagation Delay 3.3V 25pF load	-	15	-	nsec	
	Propagation Delay 3.3V 50pF load	-	20	-	nsec	
TPD3 --	Propagation Delay 3.3V No load	-	5	-	nsec	
	Propagation Delay 3.3V 25pF load	-	12	-	nsec	
	Propagation Delay 3.3V 50pF load	-	15	-	nsec	
TPD5 ++	Propagation Delay 3.3V No load	-	5	-	nsec	
	Propagation Delay 3.3V 25pF load	-	12	-	nsec	
	Propagation Delay 3.3V 50pF load	-	16	-	nsec	
TPD5 --	Propagation Delay 3.3V No load	-	4	-	nsec	
	Propagation Delay 3.3V 25pF load	-	9	-	nsec	
	Propagation Delay 3.3V 50pF load	-	12	-	nsec	

18.5 CLASSIFICATION REFLOW PROFILES

Pb-Free Process-Package Classification Temperatures

Package Thickness	Volume mm3<350	Volume mm3: 350-2000	Volume mm3>2000
<1.6 mm	260°C	260°C	260°C
1.6 mm-2.5 mm	260°C	250°C	245°C
>=2.5 mm	250°C	245°C	245°C

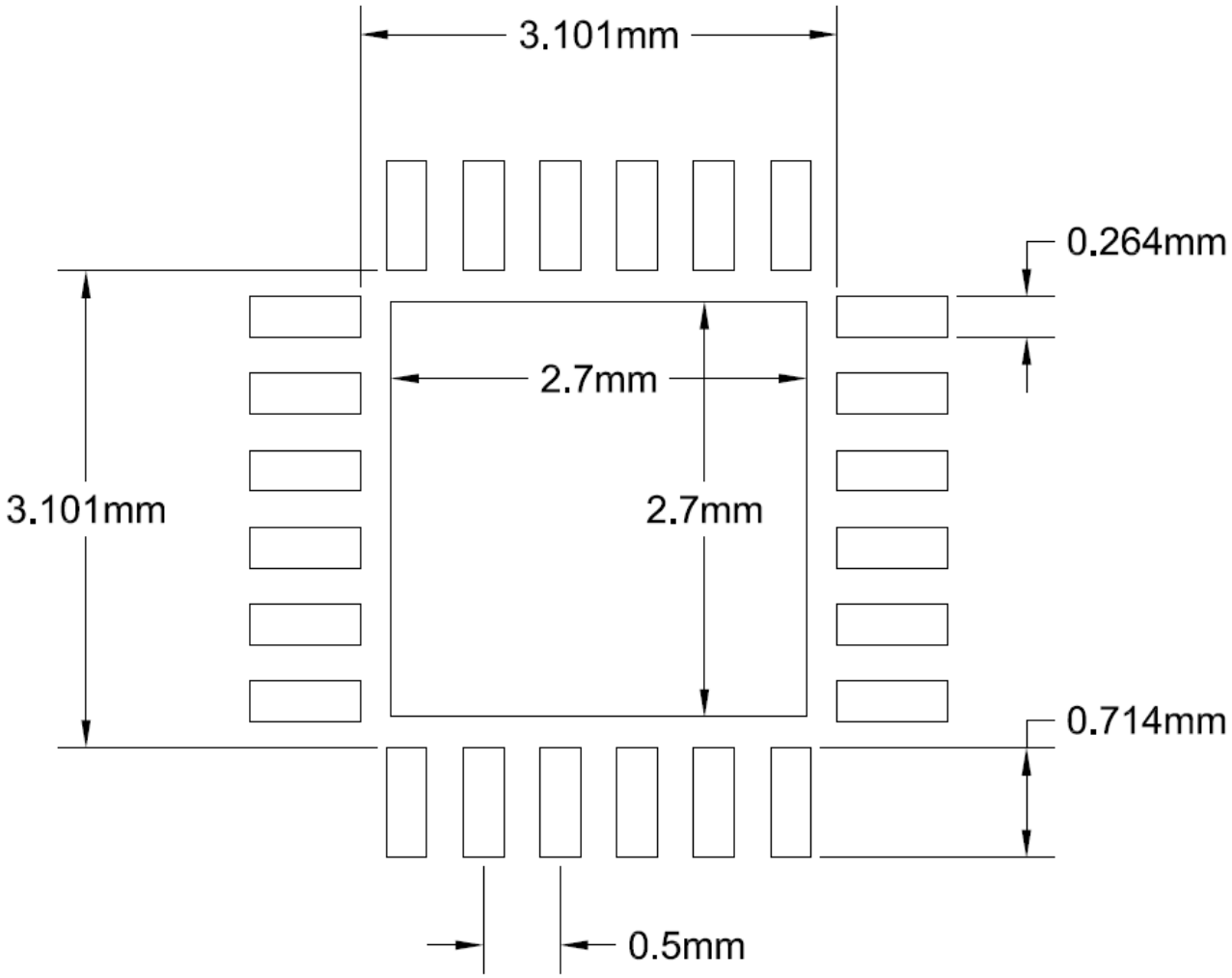
Profile Feature	Pb-Free Assembly
Ramp-Up Rate (TL to Tp)	3 °C / second max.
Preheat – Temperature Min (Tsmin) to Max (Tsmax)	150~200 °C
–To,e (tsmin to tsmax)	60-120 seconds
Time maintained above – Temperature (TL)	217 °C
– Time (tL)	60-150 seconds
Peak package body temperature (Tp)(Note 2)	See package classification
Time within 5°C of specified classification Temperature (tp)	30 second min. (Note 3)
Ramp-Down Rate (Tp to TL)	6 °C / second max.
Time 25 °C to Peak Temperature	8 minutes max.
Number of applicable Temperature cycles	3 cycles max.



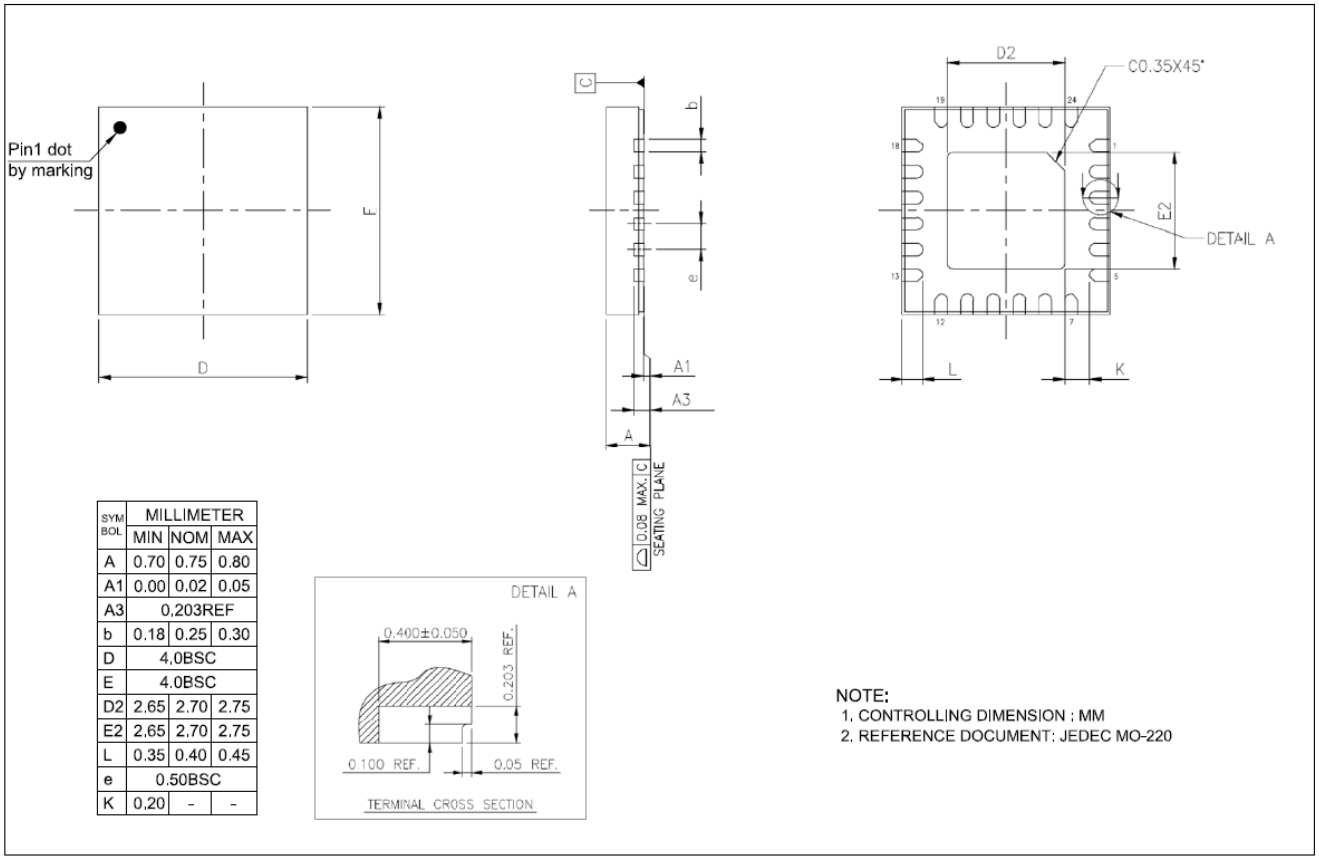
19. Packaging Outline

19.1 24-pin WQFN

RECOMMENDED LAND PATTERN



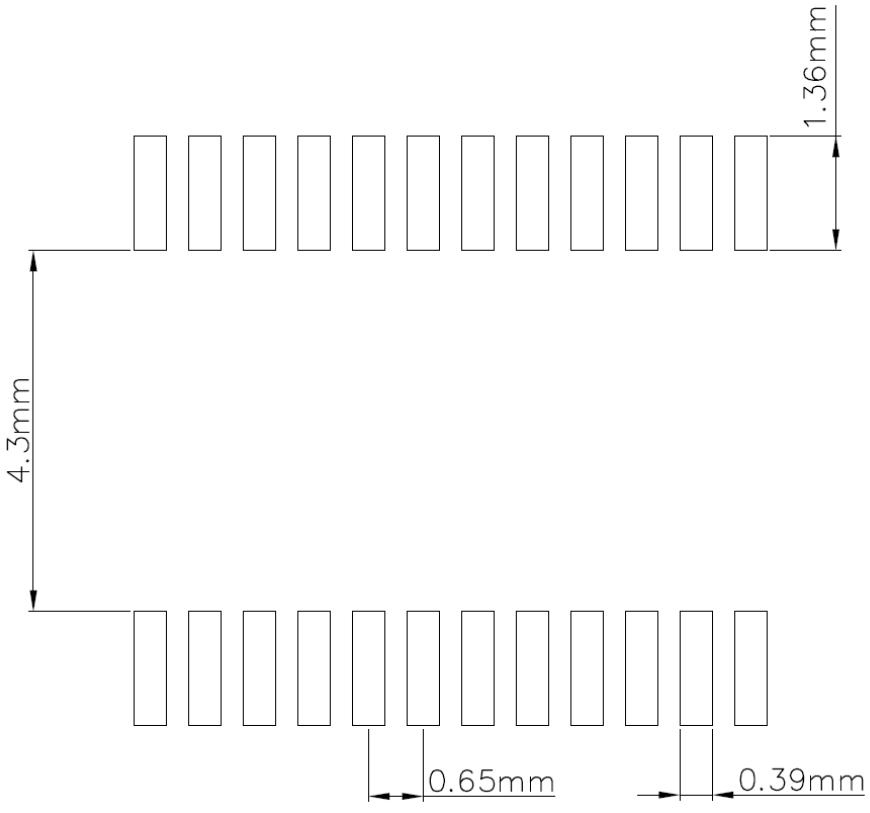
POD



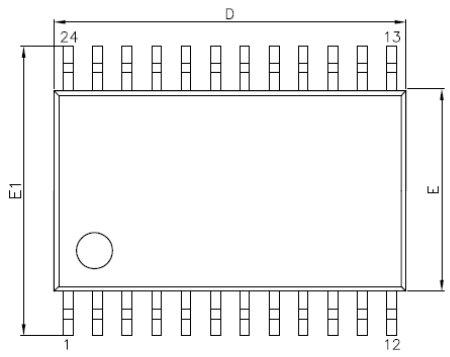
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.70	0.75	0.80
A1	0.00	0.02	0.05
A3	0.203REF		
b	0.18	0.25	0.30
D	4.0BSC		
E	4.0BSC		
D2	2.65	2.70	2.75
E2	2.65	2.70	2.75
L	0.35	0.40	0.45
e	0.50BSC		
K	0.20	-	-

19.2 24-pin TSSOP

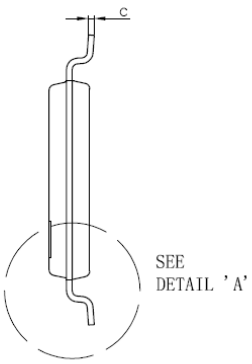
RECOMMENDED LAND PATTERN



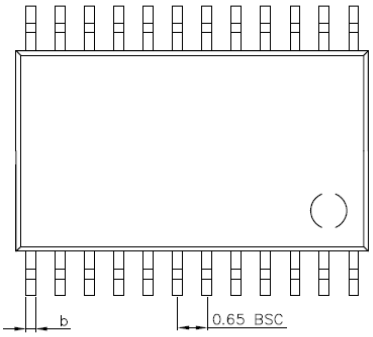
POD



TOP VIEW

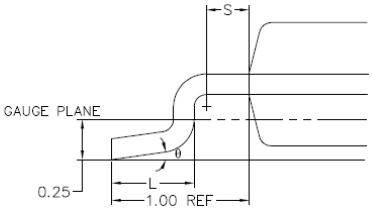


END VIEW

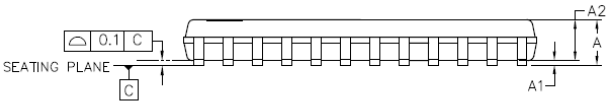


BOTTOM VIEW

SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.20
A1	0.05	—	0.15
A2	0.80	1.00	1.05
D	7.70	7.80	7.90
E	4.30	4.40	4.50
E1	6.40BSC		
L	0.45	0.60	0.75
b	0.19	—	0.30
S	0.20	—	—
c	0.09	—	0.20
θ	0°	—	8°



DETAIL 'A'



SIDE VIEW

20. Ordering Information

Operating temperature: -40°C to 125°C

Order Part No.	Package	QTY
IS32CS8974-ZNLA3-TR	TSSOP-24, Lead-free	2500/Reel
IS32CS8974-QWLA3-TR	WQFN-24, Lead-free	2500/Reel

Copyright © 2023 Lumissil Microsystems. All rights reserved. Lumissil Microsystems reserves the right to make changes to this specification and its products at any time without notice. Lumissil Microsystems assumes no liability arising out of the application or use of any information, products or services described herein. Customers are advised to obtain the latest version of this device specification before relying on any published information and before placing orders for products.
Lumissil Microsystems does not recommend the use of any of its products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure of the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless Lumissil Microsystems receives written assurance to its satisfaction, that:
a.) the risk of injury or damage has been minimized;
b.) the user assumes all such risks; and
c.) potential liability of Lumissil Microsystems is adequately protected under the circumstances

21. REVISIONS

Revision	Detailed Information	Date
A	First formal release	2021.06.08
B	1. Add packaging type “tray” on ordering information 2. Add I/O PAD block diagram 3. Revise SPI timing illustration 4. Update EUART2 and LIN controller SYNCMD description 5. Rename SIOSC to SOSC32KHz, and SOSC32KHz is 32KHz which is divided from SOSC (128KHz) 6. Remove read bit START from I2CSCON2[4]	2022.02.08
C	1. Bits 4~7 of RSTCMD register can't be read The above description is from Section 1.23 Reset – RSTCMD register 2. Revise “IOSC uses VDDC as Power supply and can be calibrated and trimmed.“ instead of early description of VDD18 The above description is for Section 11. IOSC and SOSC 3. Update Section TK2CFGB bit 5~7 ZERO[2:0] description The above description is for Section 12 Touch Key Controller II TK2CFGB 4. IFB address 40 bit 6 I2CSCL1 is not supported. The above description is for Section 15 Information Block IFB 5. Revise TCON register description. The above description applies to TCON descriptions in Section 1.5 Interrupt System and Section 1.9 System Timers – T0 and T1 6. TA/TB Protect support modification: * Remove TA Protect support for register WTST * Only support bit 0 RWT of WDCON register for TA Protect * Modification TB Protect support of Flash Zone protection from FLSHPRT[0] to FLSHPRT[31] * TB Protect support for register LVDCFG except bit 0 LVTIF 7. Add AEC-Q100 qualification 8. Reword some contents for clear explanations 9. Revise product features “Up 20 25MHz 1-Cycle 8051 CPU core (16MHz Zero wait state) “ and 2.5V to 5.5V single power supply 10. Revise VDDC from 3V ~ 5.5V to 2.5V ~5.5V for 18.3 & 18.4 DC and AC Electrical Characteristics 11. Revise some typos	2022.08.26

D	1. Support proximity sensing 2. Update the device package as “RoHS & Halogen-Free compliant” in the product “Features” 3. Modify Boot code execution procedures in Section 17. Boot Code and In-System Programming 4. Definitions of IFB address 2B~2D are all updated as “Reserved” in Section 15. Information Block IFB 5. Update “Features” for interrupt support as “All GPIO pins can be assigned to two external interrupts” 6. Update XRAM up to 0x07FFH in Data Memory Map table in MEMORY MAP section 7. Update the operation descriptions for CRCMODE[2-0] of CCCFG register in Section 1.16 Checksum/CRC Accelerator 8. Update power saving mode support for idle, stop, and sleep in “Features” section. 9. Remove “ZTRGEN” and “CTRGEN” bits for Zero and Center ADC Trigger Enable support of PWMCFG2 register and also ADC information in this document. 10. Add IFB addresses 34 & 35 Timer 0 TRIM function and update IFB address 41~43 as reserved. Please refer to Section 15 Information Block IFB 11. Add “TSCA compliance” support	2023.07.28