8-Bit MCU with 32K Flash and 1K SRAM

GENERAL DESCRIPTION

IS31CS8969 is a general-purpose micro-controller with 32K embedded flash memory and 1K SRAM. The CPU is based on 1-T 8051 with T0/T1/T2 and additional 16-bit T3/T4, 24-bit T5 and a 30-bit WDT. Embedded in the CPU core are also a full-duplex UART port, an enhanced EUART port with LIN capability, one I²C master/slave and two I²C pure slave controllers, one SPI mater/slave controller, up to 20 GPIO pins with each GPIO pin configurable as external interrupt and wake up.

The flexibility in clock setting includes an on-chip 16MHz precision oscillator with the accuracy deviation of +/-2%, or a low power internal 32KHz oscillator. The clock selections are combined with flexible power management schemes, including NORMAL, PMM, IDLE, and STOP, and SLEEP modes to balance CPU speed and power consumption.

A Programmable Counter Array (PCA) with 6 channels of Capture/Compare/PWM modules can be used for various purposes controlling external devices. There are additional 2 independent 8-bit PWM and a buzzer waveform generator with frequency range of 128Hz to 2048Hz and programmable duty cycle.

Other digital peripherals include a EUART2 with 16-byte FIFO, which support full LIN protocols, and an I2C slave controller, and a SPI Master/Slave controller with 4-byte FIFO.

Analog peripherals include a high performance 12-bit Analog to Digital Converter (ADC) with 30usec conversion time and a Programmable Gain Amplifier as ADC front-end. There are an on-chip temperature sensor, and a calibrated voltage reference within the ADC block. A 10-bit voltage output Digital to Analog Converter (DAC) is also included.

IS31CS8969 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 break point processor also allows easy debugging which can be integrated with ISP. Reliable power-on-reset circuit and low supply voltage detection allows reliable operations under harsh environments.

FEATURES

CPU and Memory

- 1-Cycle 8051 CPU core up to 16MHz
- 16-bit Timers T0/T1/T2/T3/T4 and 24-bit T5
- Programmable 30-bit Watch Dog Timer
- Integrated break point controller and debug port through I²C slave
- External interrupts and wake up shared with GPIO pins
- Power saving modes – Normal, PMM, IDLE, STOP, and SLEEP modes
- Wake-Up noise filter – up to 4 msec
- 256B IRAM and 768B XRAM
- 32KB Flash Memory and 128B Information Block
  - Code security and data loss protection
  - Endurance: 100K cycles and Retention: 10 years @85°C

Clock Sources

- Internal oscillator at 16MHz of +/- 2% accuracy
- Internal low power 32KHz oscillator

Digital Peripherals

- 16-bit PCA and 6 channel of CCP modules
  - Capture/Compare/Timer Mode
  - 8/16-bit PWM Mode and 8-bit WPWM Mode
- Two 8-bit PWM Controller
- One buzzer waveform generator
- One I²C Master
- One I²C pure Slave
  - Support two consecutive addresses
- One SPI Master/Slave Controllers
  - 4 bytes transmit and receive FIFO
- One 8051 UART
- One full-duplex LIN-capable EUART2
  - 16-bit baud rate register
  - 16 bytes transmit and receive FIFO

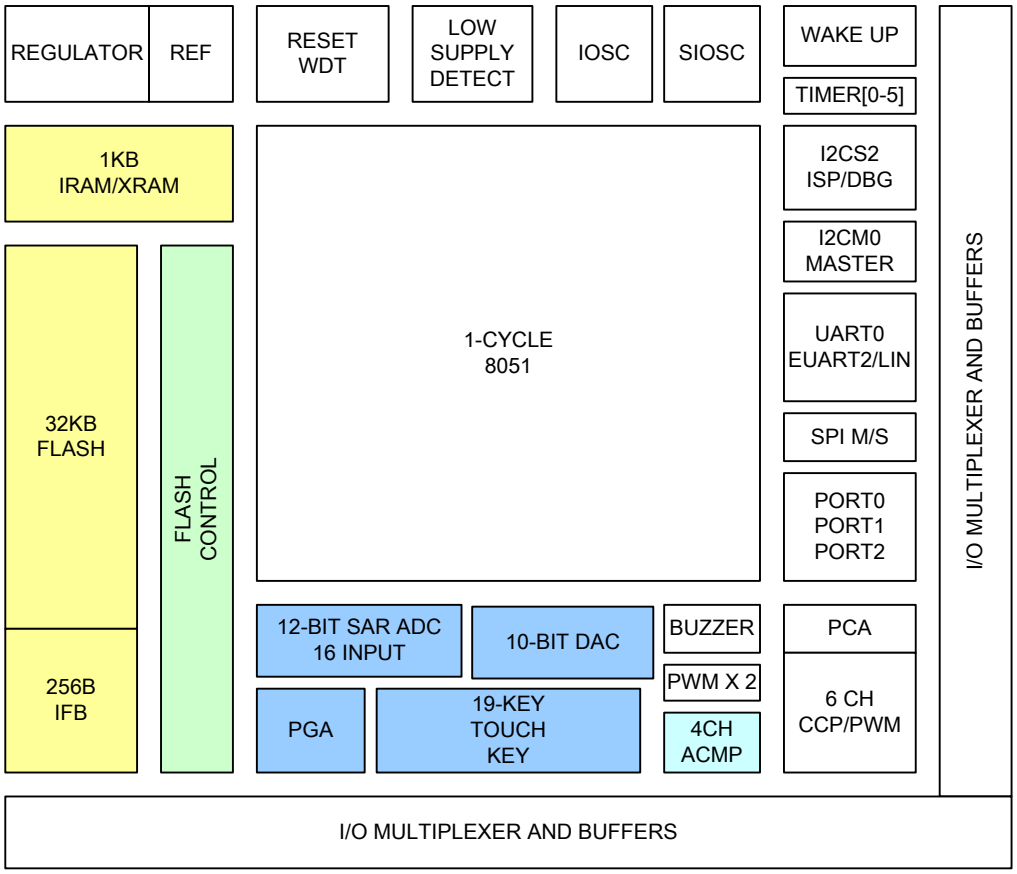
Analog Peripherals

- 12-bit monotonic SAR ADC
  - 500KHz, 128µsec conversion time (1.8~2.4V)
  - 4Mhz, 16usec conversion time (2.5~5.5V)
  - Programmable Gain Amplifier
  - Up to 16 input shared with GPIO
  - On-chip 1.1V reference
  - On-chip temperature sensor
- 10-bit voltage output DAC
- 4-Channel Analog Comparator
- Capacitance sense touch-key controller scan up to 20 keys through shared GPIO
- Power on reset (1.5V) and LVD/LVR (1.8V-5.5V)

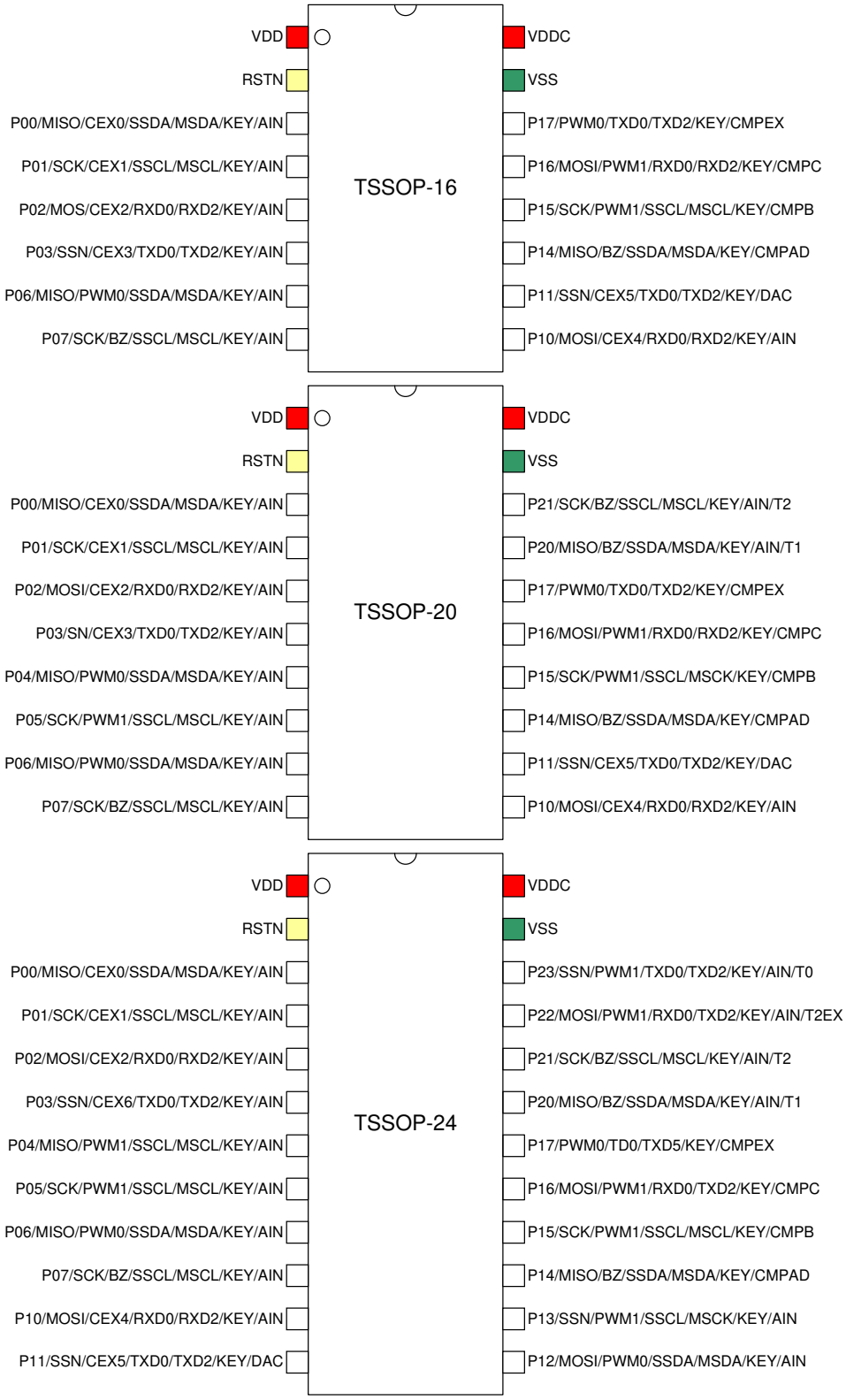
Miscellaneous

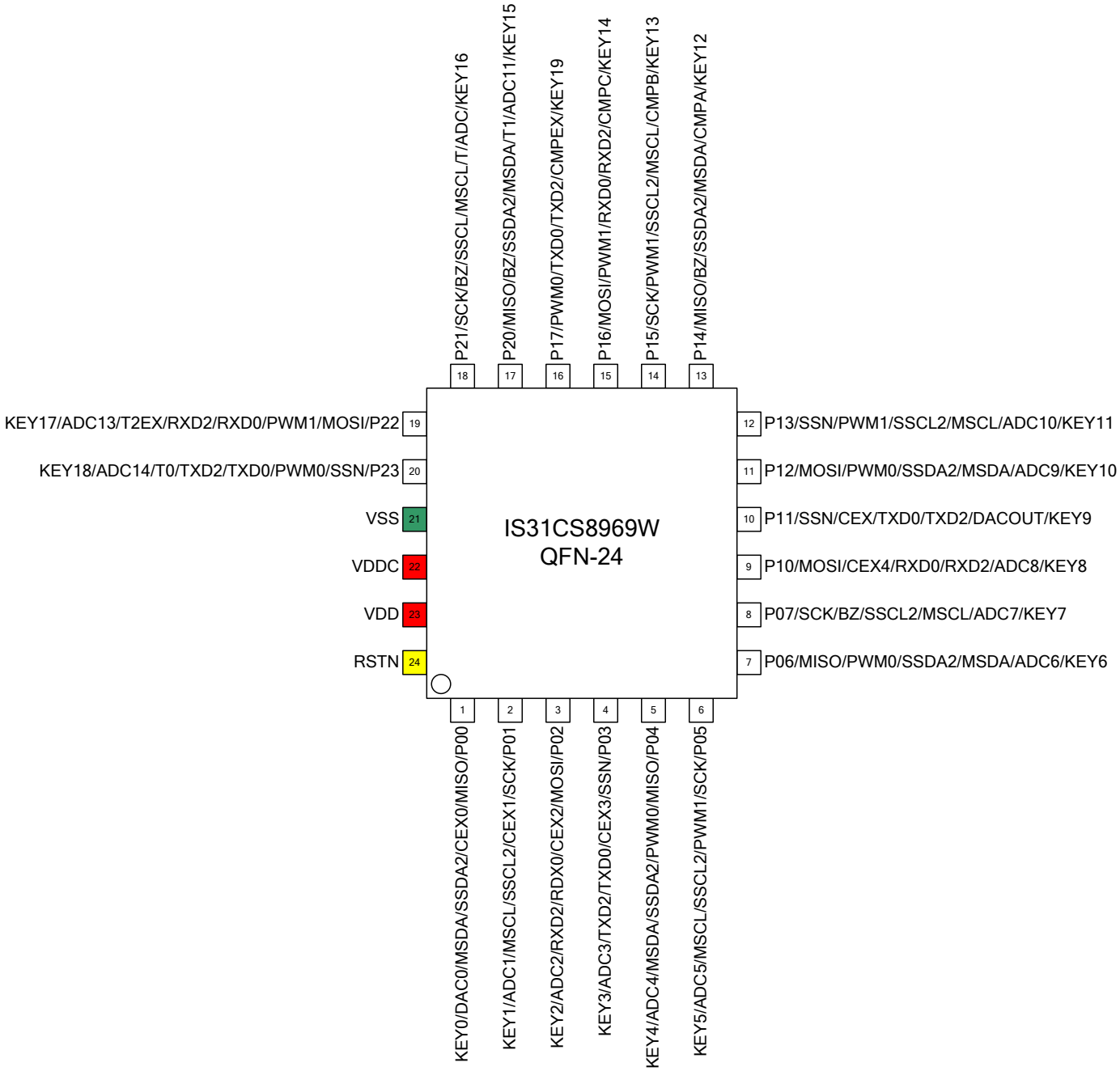
- Up to 20 GPIO pins
- 2.2V to 5.5V single supply with on-chip 1.8V regulator. Operating down to 1.8V
- Active current < 300uA/MHz in NORMAL mode
- Low power standby (< 20uA) in SLEEP mode
- Operating temperature -45°C to 85°C
- RoHS compliant packages
  - TSSOP - 24
  - TSSOP - 20
  - TSSOP – 16
  - QFN - 24

BLOCK DIAGRAM



PIN CONNECTION





Note: The Part Number and Logo is for reference only and do not reflect the actual marking on the package.

PIN DESCRIPTIONS

PIN NAME	TYPE	PIN #	DR
P0.0	IOCELL2A I/O, A	03 /03 /03 /01	**Port 0.0 GPIO**
			8051 P0.0 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			MISO
			This pin can be configured as SPI MISO I/O.
			CEX0
			This pin can also be configured as the CEX pin for CCP0. CEX is an input for compare/capture mode, and an output for PWM mode.
			SSDA2
			This pin can be configured as I2CS2 SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSDA
			This pin can be configured as I2CM SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY0
			This pin can be configured as input to touch key controller by setting IOCFGP0.0 ANEN.
			ADC0
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP0.0 ANEN1.
P0.1	IOCELL2A I/O, A	04 /04 /04 /02	**Port 0.1 GPIO**
			8051 P0.1 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			SCK
			This pin can be configured as SPI SCK I/O.
			CEX1
			This pin can also be configured as the CEX pin for CCP1. CEX is an input for compare/capture mode, and an output for PWM mode.
			SSCL2
			This pin can be configured as I2CS2 SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSCL
			This pin can be configured as I2CM SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY1
			This pin can be configured as input to touch key controller by setting IOCFGP0.1 ANEN.
			ADC1
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP0.1 ANEN1.
P0.2	IOCELL2A I/O, A	05 /05 /05	**Port 0.2 GPIO**
			8051 P0.2 GPIO.
			PINT

PIN NAME	TYPE	PIN #	DR
		/03	This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			MOSI
			This pin can be configured as SPI MOSI I/O.
			CEX2
			This pin can also be configured as the CEX pin for CCP2. CEX is an input for compare/capture mode, and an output for PWM mode.
			RXD0
			This pin can also be configured as the receiving input pin for UART0.
			RXD2
			This pin can also be configured as the receiving input pin for EUART2.
			KEY2
			This pin can be configured as input to touch key controller by setting IOCFGP0.2 ANEN.
			ADC2
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP0.2 ANEN1.
P0.3	IOCELL2A I/O, A	06 /06 /06 /04	**Port 0.3 GPIO**
			8051 P0.3 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			SSN
			This pin can be configured as SPI SSN input.
			CEX3
			This pin can also be configured as the CEX pin for CCP3. CEX is an input for compare/capture mode, and an output for PWM mode.
			TXD0
			This pin can also be configured as the transmit output pin for UART0.
			TXD2
			This pin can also be configured as the transmit output pin for EUART2.
			KEY3
			This pin can be configured as input to touch key controller by setting IOCFGP0.3 ANEN.
			ADC3
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP0.3 ANEN1.
P0.4	IOCELL2A I/O, A	07 /07 /--- /05	**Port 0.4 GPIO**
			8051 P0.4 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			MISO
			This pin can be configured as SPI MISO I/O.
			PWM0
			This pin can also be configured as the PWM0 output
			SSDA2
			This pin can be configured as I2CS2 SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSDA
			This pin can be configured as I2CM SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.

PIN NAME	TYPE	PIN #	DR
			KEY4
			This pin can be configured as input to touch key controller by setting IOCFGP0.4 ANEN.
			ADC4
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP0.4 ANEN1.
P0.5	IOCELL2A I/O, A	08 /08 /--- /06	**Port 0.5 GPIO**
			8051 P0.5 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input.
			SCK
			This pin can be configured as SPI SCK I/O.
			PWM1
			This pin can also be configured as the PWM1 output
			SSCL2
			This pin can be configured as I2CS2 SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSCL
			This pin can be configured as I2CM SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY5
			This pin can be configured as input to touch key controller by setting IOCFGP0.5 ANEN.
			ADC5
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP0.5 ANEN1.
P0.6	IOCELL2A I/O, A	09 /09 /07 /07	**Port 0.6 GPIO**
			8051 P0.6 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			MISO
			This pin can be configured as SPI MISO I/O.
			PWM0
			This pin can also be configured as the PWM0 output
			SSDA2
			This pin can be configured as I2CS2 SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSDA
			This pin can be configured as I2CM SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY6
			This pin can be configured as input to touch key controller by setting IOCFGP0.6 ANEN.
			ADC6
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP0.6 ANEN1.
P0.7	IOCELL2A	10	**Port 0.7 GPIO**

PIN NAME	TYPE	PIN #	DR
	I/O, A	/10 /08 /08	8051 P0.7 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			SCK
			This pin can be configured as SPI SCK I/O.
			BZOUT
			This pin can also be configured as the Buzzer output
			SSCL2
			This pin can be configured as I2CS2 SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSCL
			This pin can be configured as I2CM SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY7
			This pin can be configured as input to touch key controller by setting IOCFGP0.7 ANEN.
			ADC7
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP0.7 ANEN1.
P1.0	IOCELL2A I/O, A	11 /11 /09 /09	**Port 1.0 GPIO**
			8051 P1.0 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			MOSI
			This pin can be configured as SPI MOSI I/O.
			CEX4
			This pin can also be configured as the CEX pin for CCP4. CEX is an input for compare/capture mode, and an output for PWM mode.
			RXD0
			This pin can also be configured as the receiving input pin for UART0.
			RXD2
			This pin can also be configured as the receiving input pin for EUART2.
			KEY 8
			This pin can be configured as input to touch key controller by setting IOCFGP1.0 ANEN.
			ADC8
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP1.0 ANEN1.
P1.1	IOCELL2A I/O, A	12 /12 /10 /10	**Port 1.1 GPIO**
			8051 P1.1 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			SSN
			This pin can be configured as SPI SSN input.
			CEX5
			This pin can also be configured as the CEX pin for CCP5. CEX is an input for compare/capture mode, and an output for PWM mode.
			TXD0

PIN NAME	TYPE	PIN #	DR
			This pin can also be configured as the transmit output pin for UART0.
			TXD2
			This pin can also be configured as the transmit output pin for EUART2.
			KEY9
			This pin can be configured as input to touch key controller by setting IOCFGP1.1 ANEN.
			DACOUT
			This pin can also be configured as the DAC output by setting IOCFGP1.1 ANEN1.
P1.2	IOCELL2A I/O, A	13 /--- /--- /11	**Port 1.2 GPIO**
			8051 P1.2 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			MOSI
			This pin can be configured as SPI MOSI I/O.
			PWM0
			This pin can also be configured as the PWM0 output
			SSDA2
			This pin can be configured as I2CS2 SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSDA
			This pin can be configured as I2CM SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY10
			This pin can be configured as input to touch key controller by setting IOCFGP1.2 ANEN.
			ADC9
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP1.2 ANEN1.
P1.3	IOCELL2A I/O, A	14 /--- /--- /12	**Port 1.3 GPIO**
			8051 P1.3 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			SSN
			This pin can be configured as SPI SSN input.
			PWM1
			This pin can also be configured as the PWM1 output
			SSCL2
			This pin can be configured as I2CS2 SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSCL
			This pin can be configured as I2CM SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY11
			This pin can be configured as input to touch key controller by setting IOCFGP1.3 ANEN.
			ADC10

PIN NAME	TYPE	PIN #	DR
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP1.3 ANEN1.
P1.4	IOCELL2A I/O, A	15 /13 /11 /13	**Port 1.4 GPIO**
			8051 P1.4 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			MISO
			This pin can be configured as SPI MISO I/O.
			BZOUT
			This pin can also be configured as the Buzzer output
			SSDA2
			This pin can be configured as I2CS2 SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSDA
			This pin can be configured as I2CM SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY12
			This pin can be configured as input to touch key controller by setting IOCFGP1.4 ANEN.
			CMPA
			This pin can be configured as input to analog comparator A by setting IOCFG1.4 ANEN1.
P1.5	IOCELL2A I/O, A	16 /14 /12 /14	**Port 1.5 GPIO**
			8051 P1.5 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			SCK
			This pin can be configured as SPI SCK I/O.
			PWM1
			This pin can also be configured as the PWM1 output
			SSCL2
			This pin can be configured as I2CS2 SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSCL
			This pin can be configured as I2CM SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			KEY13
			This pin can be configured as input to touch key controller by setting IOCFGP1.5 ANEN.
			CMPB
			This pin can be configured as input to analog comparator B by setting IOCFG1.5 ANEN1.
P1.6	IOCELL2A I/O, A	17 /15 /13	**Port 1.6 GPIO**
			8051 P1.6 GPIO.
			PINT

PIN NAME	TYPE	PIN #	DR
		/15	This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			MOSI
			This pin can be configured as SPI MOSI I/O.
			PWM1
			This pin can also be configured as the PWM1 output
			RXD0
			This pin can also be configured as the receiving input pin for UART0.
			RXD2
			This pin can also be configured as the receiving input pin for EUART2.
			KEY14
			This pin can be configured as input to touch key controller by setting IOCFGP1.6 ANEN.
			CMPC
			This pin can be configured as input to analog comparator C by setting IOCFG1.6 ANEN1.
P1.7	IOCELL2A I/O, A	18 /16 /14 /16	**Port 1.7 GPIO**
			8051 P1.7 GPIO.
			PINT
			This pin can be configured as the PINT0 or PINT1 pin external interrupt input
			PWM0
			This pin can also be configured as the PWM0 output
			TXD0
			This pin can also be configured as the transmit output pin for UART0.
			TXD2
			This pin can also be configured as the transmit output pin for EUART2.
			KEY19
			This pin can be configured as input to touch key controller by setting IOCFGP1.7 ANEN.
			CMPEX
			This pin can be configured as input to analog comparator external threshold by setting IOCFG1.7 ANEN1.
P2.0	IOCELL2A I/O, A	19 /17 /--- /17	**Port 2.0 GPIO**
			8051 P2.0 GPIO.
			MISO
			This pin can be configured as SPI MISO I/O.
			BZOUT
			This pin can also be configured as the Buzzer output
			SSDA2
			This pin can be configured as I2CS2 SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSDA
			This pin can be configured as I2CM SDA I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			T1
			This pin can be configured as T1 external input
			KEY15
			This pin can be configured as input to touch key controller by setting IOCFGP2.0 ANEN.
			ADC11

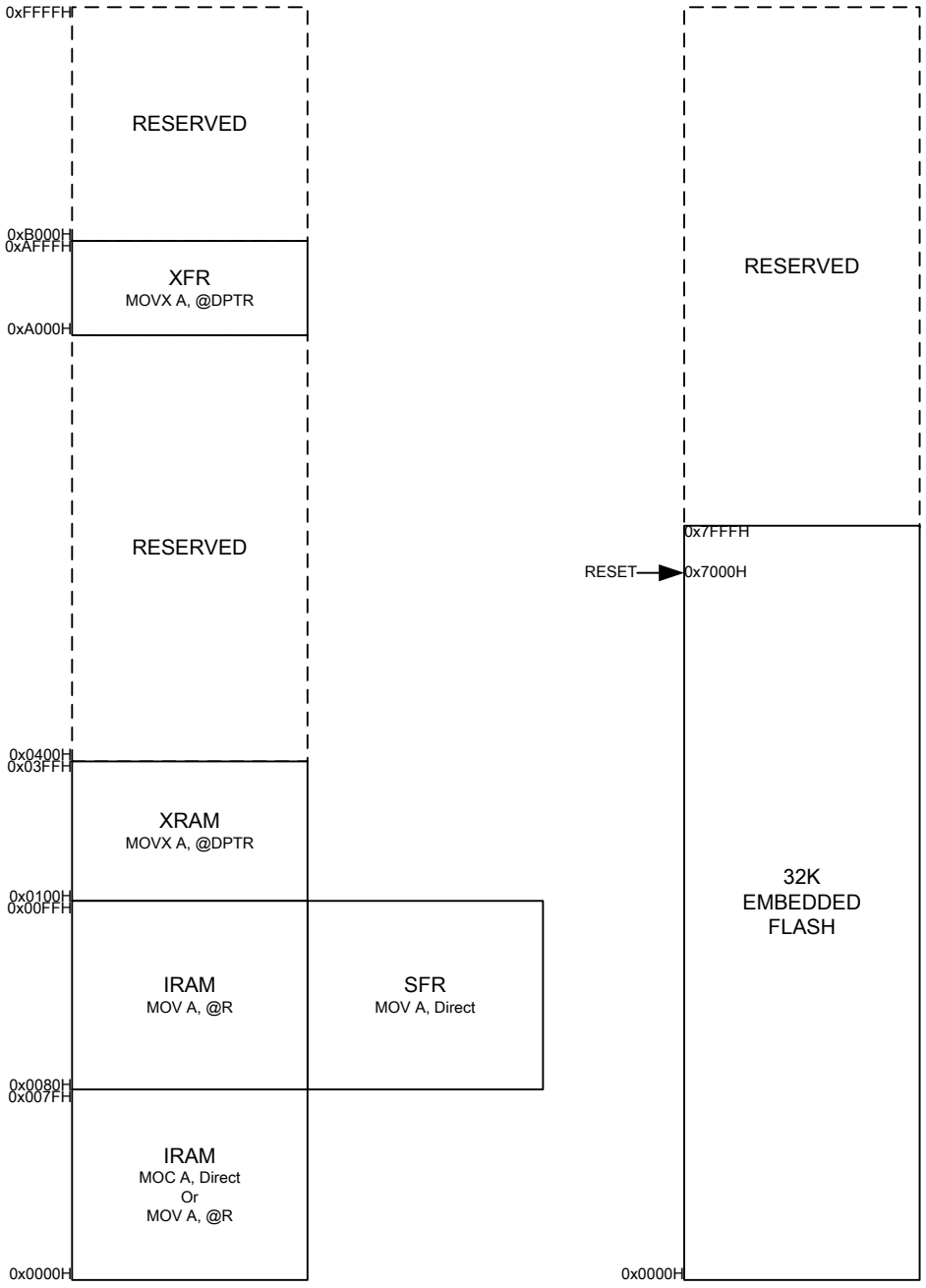
PIN NAME	TYPE	PIN #	DR
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP2.0 ANEN1.
P2.1	IOCELL2A I/O, A	20 /18 /--- /18	**Port 2.1 GPIO**
			8051 P2.1 GPIO.
			SCK
			This pin can be configured as SPI SCK I/O.
			BZOUT
			This pin can also be configured as the Buzzer output
			SSCL2
			This pin can be configured as I2CS2 SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			MSCL
			This pin can be configured as I2CM SCL I/O. The IOCOFG must also be configured as Open- Drain and external pull up resistor connecting to VDD is necessary.
			T2
			This pin can be configured as T2 external input
			KEY16
			This pin can be configured as input to touch key controller by setting IOCFGP2.1 ANEN.
			ADC12
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP2.1 ANEN1.
P2.2	IOCELL2A I/O, A	21 /--- /--- /19	**Port 2.2 GPIO**
			8051 P2.2 GPIO.
			MOSI
			This pin can be configured as SPI MOSI I/O.
			PWM1
			This pin can also be configured as the PWM1 output
			RXD0
			This pin can also be configured as the receiving input pin for UART0.
			RXD2
			This pin can also be configured as the receiving input pin for EUART2.
			T2EX Timer 2 Trigger
			This pin also can be configured as T2EX signal for Timer 2. T2EX is the Timer 2 trigger input.
			KEY17
			This pin can be configured as input to touch key controller by setting IOCFGP2.2 ANEN.
			ADC13
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP2.2 ANEN1.
P2.3	IOCELL IO, A	22 /--- /--- /20	**Port 2.3 GPIO**
			8051 P2.3 GPIO.
			SSN
			This pin can be configured as SPI SSN input.
			PWM0
			This pin can also be configured as the PWM0 output
			TXD0
			This pin can also be configured as the transmit output pin for UART0.

PIN NAME	TYPE	PIN #	DR
			TXD2
			This pin can also be configured as the transmit output pin for EUART2.
			T0
			This pin can be configured as T0 external input
			KEY18
			This pin can be configured as input to touch key controller by setting IOCFGP2.3 ANEN.
			ADC14
			Thin pin can be configured as input to ADC and PGA by setting IOCFGP2.3 ANEN1.
VSS	G	23/19/ 15/21	Ground connection.
VDDC	P	24/20/ 16/22	**Internal Regulator Output. 1.6V ~ 2.0V** Typical decoupling capacitors of 0.1uF and 10uF should be connected between VDDC and VSS.
VDD	P	1/1/ 1/23	VDD supplies power to I/O buffers as well as analog circuits such as ADC and comparators. A good decoupling capacitor between VDD and VSS pins is critical for good performance. VDD ranges from 1.8V to 5.5V
RSTN	I/O, A	2/2/ 2/24	Typically connect a resistor to VDD and a capacitor to VSS. RSTN is pulled low actively when LVR occurs. The threshold of RSTN is set at 0.3VDD. RSTN is also used for special test mode and writer mode entry.
			Reset Low Active.

Note: "P" denotes power supply pins
"G" denotes ground pins. All VSS pins are internally shorted resistively.
"O", "IO", "A" denotes output only, input/output, and analog types.
"IOCELL2A" denotes IOCELL with two analog inputs
PIN # : TSSOP-24 / TSSOP-20 / TSSOP-16 / QFN-24

MEMORY MAP

There are total 256 bytes internal RAM in IS31CS8969, the same as standard 8052. There are total 1792 bytes auxiliary RAM allocated in the 8051 extended RAM at 0x0100h – 0x07FFh. Programs can use "MOVX" instruction to access the XRAM. The 32KB embedded flash occupies the code address space from 0x0000h – 0x7FFFh. The CPU reset to address 0x7000h. The memory map is shown in the following:



REGISTER MAP SFR(0x80 – 0xFF) and XFR (0xA000 – 0xAFFF)

The SFR address map maintains maximum compatibilities to most commonly used 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	**PCACON**	**CCAP3L**	**CCAP3H**	**CCAP4L**	**CCAP4H**	**CCAP5L**	**CCAP5H**
0XD0	PSW	**PCAMOD**	**CCAP0L**	**CCAP0H**	**CCAP1L**	**CCAP1H**	**CCAP2L**	**CCAP2H**
0XC0	**-**	**-**	SCON2	-	PMR	STATUS	MCON	TA
0XB0	**-**	**-**	**CCAPM0**	**CCAPM1**	**CCAPM2**	**CCAPM3**	**CCAPM4**	**CCAPM5**
0XA0	P2	SPICR	SPIMR	SPIST	SPIDAT	SFIFO2	SBUF2	SINT2
0X90	P1	EXIF	WTST	DPX	**CMPST**	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	**-**	**-**	**-**	**-**	**-**
0XD8	WDCON	CL	DPXR	I2CSCON2	I2CSST2	I2CSADR1	I2CSDAT2	I2CSADR2
0XC8	T2CON	TB	RLDL	RLDH	TL2	TH2	**ADCTTL**	**T34CON**
0XB8	IP	**ADCPGA**	**ADCL**	**ADCH**	-	**-**	**-**	**-**
0XA8	IE	**ADCCFG**	**-**	**-**	**T4L**	**T4H**	**T3L**	**T3H**
0X98	SCON0	SBUF0	**-**	ESP	**-**	ACON	**-**	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	T5CON	T5L	T5H	T5T	XOSCCFG
A010	LVDCFG	LVDTHD	FLSHADM	INTPCT1	INTPCT2	-	LVDHYS	COMPCFG
A020	FLSHCMD	FLSHDAT	FLSHADH	FLSHADL	ISPCLKF	CNTPCTL	CNTPCTH	-
A030	CMPCFGAB	CMPCFGCD	CMPVTH0	CMPVTH1	-	-	DACL	DACH
A040	IOCFGP0.0	IOCFGP0.1	IOCFGP0.2	IOCFGP0.3	IOCFGP0.4	IOCFGP0.5	IOCFGP0.6	IOCFGP0.7
A050	MFCFGP0.0	MFCFGP0.1	MFCFGP0.2	MFCFGP0.3	MFCFGP0.4	MFCFGP0.5	MFCFGP0.6	MFCFGP0.7
A060	IOCFGP2.0	IOCFGP2.1	IOCFGP2.2	IOCFGP2.3	-			
A070	MFCFGP2.0	MFCFGP2.1	MFCFGP2.2	MFCFGP2.3	-			
	8	9	A	B	C	D	E	F
A008	-	-	-	-	-	-	-	
A018	-	-	-	-	-	-	-	
A028	PIOEDGR0	PIOEDGR1					GPWKCFG	-
A038	PIOEDGF0	PIOEDGF1					-	-
A048	IOCFGP1.0	IOCFGP1.1	IOCFGP1.2	IOCFGP1.3	IOCFGP1.4	IOCFGP1.5	IOCFGP1.6	IOCFGP1.7
A058	MFCFGP1.0	MFCFGP1.1	MFCFGP1.2	MFCFGP1.3	MFCFGP1.4	MFCFGP1.5	MFCFGP1.6	MFCFGP1.7
A068	-	-	-	-	-	-	-	
A078	-	-	-	-	-	-	-	

	0	1	2	3	4	5	6	7
A080	-	-	-	-			-	-
A090	LINCTRL	LINCNTRH	LINCNTRL	LINSBRH	LINSBRL	LININT	LININTEN	-
A0A0	-	-	-	-	-	PCACPS	CLRLD	CHRLD
A0B0	-	-	-	-	-	-	-	-
A0C0								
A0D0								
A0E0	BPINTF	BPINTE	BPINTC	BPCTRL	PC5AL	PC5AH	PC5AT	-
A0F0	PC1AL	PC1AH	PC1AT	-	PC2AL	PC2AH	PC2AT	-
	8	9	A	B	C	D	E	F
A088	**PWM0CFG**	**PWM0DTY**	**PWM1CFG**	**PWM1DTY**	PWMFG	-	-	-
A098	DBPCIDL	DBPCIDH	DBPCIDT	DBPCNXL	DBPCNXH	DBPCNXT	-	-
A0A8								
A0B8								
A0C8								
A0D8								
A0E8	PC6AL	PC6AH	PC6AT	-	PC7AL	PC7AH	PC7AT	SI2CDBGID
A0F8	PC3AL	PC3AH	PC3AT	-	PC4AL	PC4AH	PC4AT	STEPCTR-

	0	1	2	3	4	5	6	7
A100	-	-	-	-	-	-	-	-
A110	-	-	-	-	-	-	-	-
A120	-	-	-	-	-	-	-	-
A130	TKCCFG	TKCAFG	TKCFFG	TKCATH0	TKCATH1	TKCATH2	-	-
A140	TKCTMR0	TKCTMR1	TKCTMR2	-	TKCVAL0	TKCVAL1	TKCVAL2	-
A150	-	-	-	-	-	-	-	-
A160	-	-	-	-	-	-	-	-
A170	-	-	-	-	-	-	-	-
	8	9	A	B	C	D	E	F
A108	-	-	-	-	-	-	-	-
A118	-	-	-	-	-	-	-	-
A128	BZFRQ	BZFMCFG	BZSNFMCT	BZCFG	BZSNMUTE	-	-	-
A138	CAPMVFT	CAPMVFH	CAPMVFM	CAPMVFL	CAPMVST	CAPMVSH	CAPMVSM	CAPMVSL
A148	TKCMVF0	TKCMVF1	TKCMVF2		TKCMVS0	TKCMVS1	TKCMVS2	
A158	-	-	-	-	-	-	-	-
A168	-	-	-	-	-	-	-	-
A178	-	-	-	-	-	-	-	-

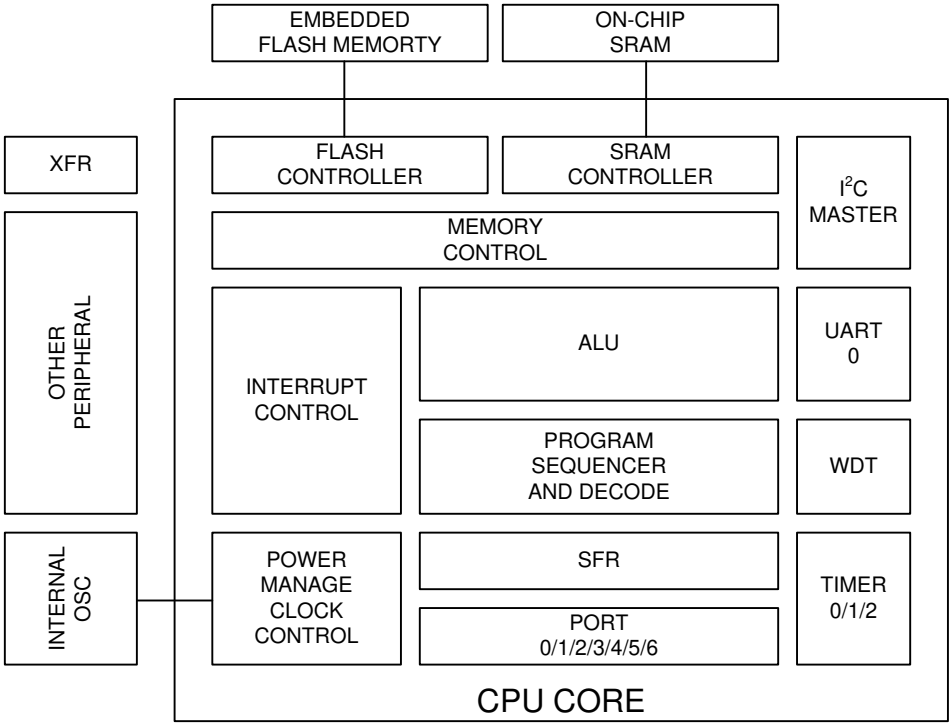
1. Enhanced 1-Cycle 8051 CPU

The CPU core is an enhanced version of standard 8051 used by series of ISSI MCU products. The CPU core is in RISC architecture and maintains binary instruction set compatible with the industry standard 8051. There is average 10 times performance enhancement in typical applications. The CPU operates at 20-bit addressing space that allows up to 1M bytes of program and data space for expansion. The CPU includes the following enhanced features compared with standard 8051:

- 16-bit LARGE addressing mode and 20-bit FLAT addressing mode control register ACON
- Two data pointers DPTR and DPTR1, and additional DPS, DPX, DPX1, MXAX registers for MOVX instruction
- 8-bit stack pointer for LARGE mode and 16-bit extended stack pointer for FLAT mode control register ESP
- Hardware Multiplication and Division Unit (MDU) provides 12 times faster performance using MD[5-0] and ARCON
- Programmable wait state for program space for on-chip flash memory using WTST register
- 256 Bytes of Direct Data Memory
- Enhanced Interrupt Controller allows 15 interrupt sources and 2 priority levels.
- Power Saving modes include IDLE mode, Power Management mode (PMM), and STOP mode. The PMM mode also supports switchback features.
- Access Control of critical registers - TA, and TB registers
- Eight break pointers allows integration of common IDE

In addition to standard 8051 peripherals, the CPU core also integrates the following peripherals. These peripherals are in the same CPU clock domain.

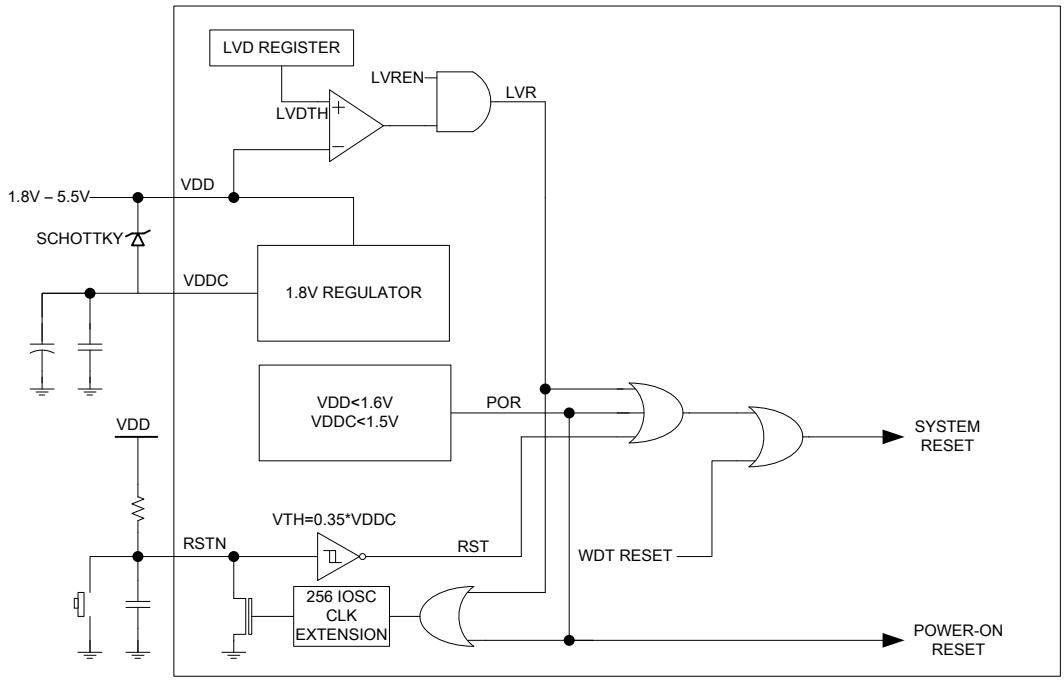
- Six 8-Bit I/O ports
- 30-bit Watch Dog Timer. WDT, WDCON, and CKCON registers
- Three 16-bit Timers, T0/T1 and T2. TCON, RLDL, RLDH, TL2, TH2, and T2CON registers
- UART0.
- I2C Master Controller. I2CMSA, I2CMCR, I2CMBUF, and I2CMTP registers.



The following sections describe in detail these enhanced features and peripherals. Assuming readers are familiar with 8051 standard operations and peripherals, the compatible functions is not covered here.

1.1 System Reset

After system reset, all registers resume the default value. The default value is shown in the register description. The reset conditions include power on/off reset, external RSTN pin being pulled low, low supply voltage detection reset, and WDT reset. The block diagram illustrating these reset conditions is shown as follows:



The power on/off reset (POR) is asserted when VDD is less than 1.6V or VDDC is less than 1.5V. The external RSTN pin can also generate reset to the device. In typical applications, the RSTN should have a resister (R1) connected to VDD and a capacitor (C1) to ground. For a system with a hardware reset control, there is usually a button switch connecting RSTN pin to ground. When the switch is pressed, it causes RSTN to short to ground, and the device enters reset state. The RSTN logic has a built-in filter that ignores RSTN duration shorter than 5usec. It is, therefore, recommended that RSTN needs to be actively pulled low for at least 50usec to guarantee a solid reset. The LVD circuits can detect the main supply voltage level VDD and the threshold can be adjusted. LVD reset is disabled by default, yet may be enabled by the software. The LVD output can be enabled to generate LVR (Low Voltage Reset). Both POR and LVR will also forces RSTN low. This ensures a solid and extended reset when the voltage supply to the internal logic and flash memory is lower than the rated level.

The last reset source is from the watchdog counter (WDT). The WDT reset function is enabled whenever a system reset occurs, and WDT timeout is set to maximum. It is recommended that all software should keep WDT enabled to ensure reliable software executions.

The program counter is loaded with 0x07000 after reset. This differs from standard 8051. In typical cases, 0x0F000 starts Calibration and ISP boot codes and then jumps to 0x0000. The clock selection after reset is set to using internal oscillator automatically. The IOSC is disabled only in STOP and SLEEP modes.

1.2 CPU Registers

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 is involved in 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 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
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.

1.3 Addressing Mode and Memory Operations

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

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

ACON is addressing mode control register.

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] is 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. 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

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

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

DPXR[7-0] is used to replace P2[7-0] for high byte of XRAM address for ”MOVX, @R1” or ”MOVX, @R0” when DPXREN=1.

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 up to 100MHz to 200MHz, but the access time of flash memory is usually around 20 nanoseconds and thus limiting the clock rate to lower than 50MHz. To alleviate this problem, a programmable wait state function is incorporated to allow faster CPU clock rate however slower embedded flash memory. The wait state is controlled by 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 is wait state register that controls the program access wait state only.

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	1
0	0	0	1	2
0	0	1	0	3
0	0	1	1	4
0	1	0	0	5
0	1	0	1	6
0	1	1	0	7
0	1	1	1	8
1	0	0	0	9
1	0	0	1	10
1	0	1	0	11
1	0	1	1	12
1	1	0	0	13
1	1	0	1	14
1	1	1	0	15
1	1	1	1	16

The default setting of the program wait state register after reset is 0x07 and the software must initialize the setting to change the wait state setting. For typical embedded flash, the read access time is specified as 40 nsec. Therefore the user should set the WTST register according to the SYSCLK frequency. For example, using a SYSCLK of 4MHz, the WTST can be set to minimum because one clock period is 250 nsec which is longer than the embedded flash access time. If SYSCLK is above 16MHz, then WTST should be set higher than 1 to allow enough read access time.

1.4 Dual Data Pointers and MOVX operations

In standard 8051/8052, there is only one data pointers DPH:DPL to perform MOVX. The enhanced CPU provides 2nd data pointer DPH1:DPL1 to speed up the movement, or copying of data block. The active DPTR is selected and operation of DPTR is controlled by setting DPS (Data Pointer Select) register. Through the control DPS, efficient programming can be achieved.

DPS (0x86) 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/Decrement functions of selected DPTR for INC DPTR instruction is 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 an 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 top 8-bit address of DPTR when address above 32KB. The lower 16-bit address is formed by DPH and DPL. Since IS31CS8969 only has on-chip data RAM space, DPX value has no effect.

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 top 8-bit address of DPTR when address above 32KB. The lower 16-bit address is formed by DPH and DPL. Since IS31CS8969 only has on-chip data RAM space, DPX value has no effect.

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 top 8-bit address for an “MOVX @R0” or “MOVX @R1” instruction. The lower 16-bit address is formed by P2 and R0/R1 (if DPXREN=0), or formed by DPXR and R0/R1 (if DPXREN=1).

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 4KB steps. For example, if MCON[7-0]=0x01, the starting address is 0x001000h. MCON is not meaningful in IS31CS8969 because it only contains on-chip XRAM and MCON should not be modified from 0x00.

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 "@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 thus requiring 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.

1.5 Interrupt System

The CPU implements an enhanced Interrupt Control that allows total 15 interrupt sources and each with two programmable priority levels. The interrupts are sampled at rising edge of SYSCLK. If interrupts are present and enabled, the CPU enters interrupt service routine by vectoring to the highest priority interrupt. Of the 15 interrupt sources, 7 of them are from CPU internal integrated peripherals, 6 of them are for 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 flags of sharing peripherals.

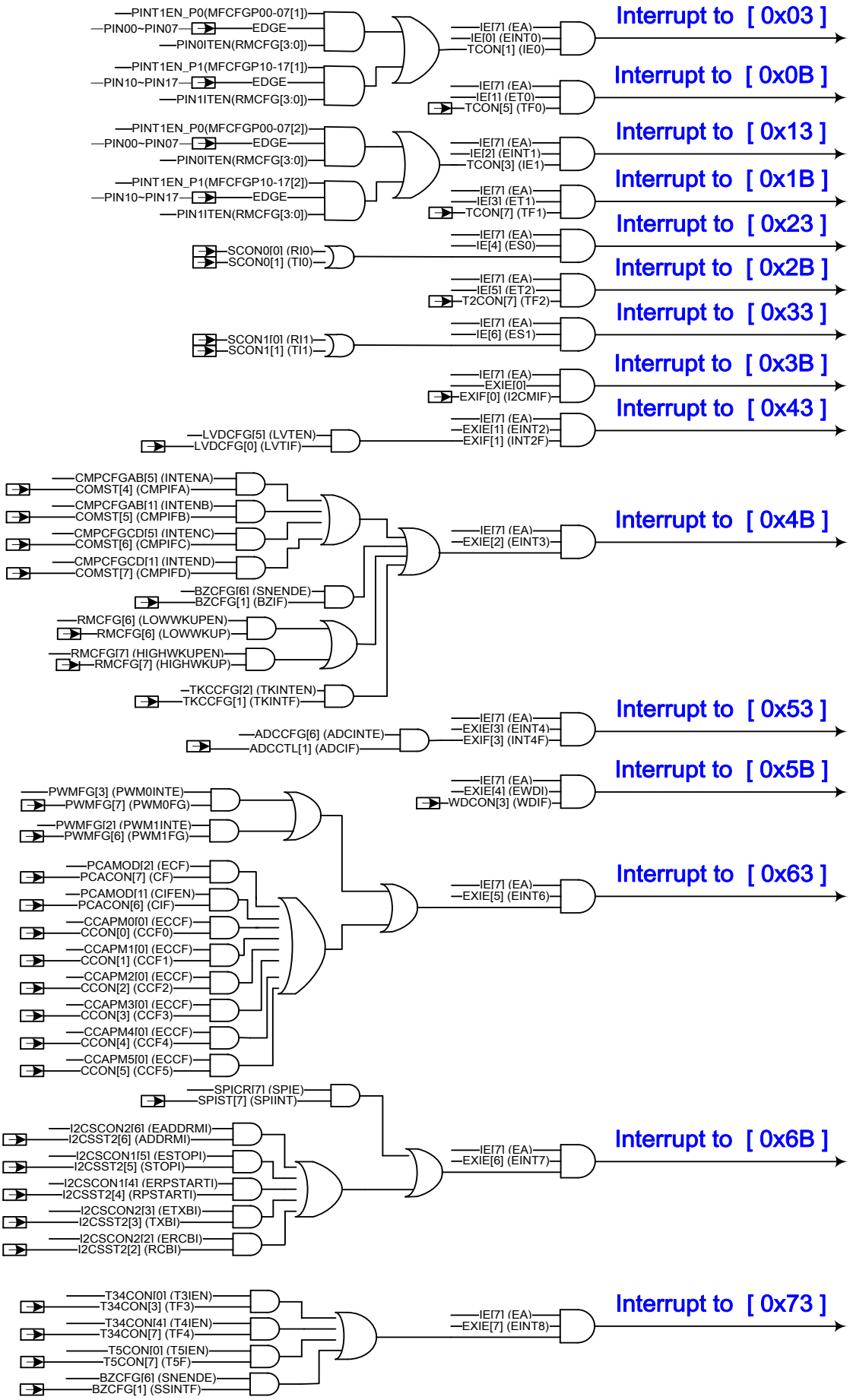
The following table shows the interrupt sources and corresponding interrupt vectors. The Flag Reset column shows whether the corresponding interrupt flag is cleared by hardware (self-cleared) or software. Please note the software can only clear the interrupt flag but not set the interrupt flag. The Natural Priority column shows the inherent priority if more than one interrupts are assigned to the same priority level. Please note that 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	FLAG RESET	Natural Priority
PINT0	Expanded Pin INT0.x	0x03	Software	1
TF0	Timer 0	0x0B	Hardware	2
PINT1	Expanded Pin INT1.x	0x13	Software	3
TF1	Timer 1	0x1B	Hardware	4
TI0/RI0	UART0	0x23	Software	5
TF2	Timer 2	0x2B	Software	6
TI2/RI2	EUART2/LIN	0x33	Software	7
I2CM	I²C Master	0x3B	Software	8
INT2	LVT/LVT18	0x43	Software	9
INT3	Comparator/Buzzer/Touch_Key/ Remote-filter	0x4B	Software	10
INT4	ADC	0x53	Software	11
WDIF	Watchdog	0x5B	Software	12
INT6	PCA/CAPP/PWM16	0x63	Software	13
INT7	SPI/I2CS2	0x6B	Software	14
INT8	Timer 3/Timer 4/Timer 5	0x73	Software	15
BKP	Break Point	0x7B	Software	0
DBG	I2CS Debug	0x83	Software	0

In addition to the 15 peripheral interrupts, there are two highest priority interrupts associated with debugging and break point. 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 break point match condition occurs. DBG has higher priority than BKP. The BKP and DBG interrupts are not affected by global interrupt enable, EA bit, IE reigster (0xA8).

The following diagram shows the interrupt sources and the expanded pin interrupts


PINT1EN_P0(MFCFGP00-07[1])
PIN00~PIN07
EDGE
PIN0ITEN(RMCFG[3:0])
PINT1EN_P1(MFCFGP10-17[1])
PIN10~PIN17
EDGE
PIN1ITEN(RMCFG[3:0])
IE[7] (EA)
IE[0] (EINT0)
TCON[1] (IE0)
Interrupt to [0x03]
IE[7] (EA)
IE[1] (ET0)
TCON[5] (TF0)
Interrupt to [0x0B]
PINT1EN_P0(MFCFGP00-07[2])
PIN00~PIN07
EDGE
PIN0ITEN(RMCFG[3:0])
PINT1EN_P1(MFCFGP10-17[2])
PIN10~PIN17
EDGE
PIN1ITEN(RMCFG[3:0])
IE[7] (EA)
IE[2] (EINT1)
TCON[3] (IE1)
Interrupt to [0x13]
IE[7] (EA)
IE[3] (ET1)
TCON[7] (TF1)
Interrupt to [0x1B]
SCON0[0] (RI0)
SCON0[1] (TI0)
IE[7] (EA)
IE[4] (ES0)
Interrupt to [0x23]
IE[7] (EA)
IE[5] (ET2)
T2CON[7] (TF2)
Interrupt to [0x2B]
SCON1[0] (RI1)
SCON1[1] (TI1)
IE[7] (EA)
IE[6] (ES1)
Interrupt to [0x33]
IE[7] (EA)
EXIE[0]
EXIF[0] (I2CMIF)
Interrupt to [0x3B]
LVDCFG[5] (LVTEN)
LVDCFG[0] (LVTIF)
IE[7] (EA)
EXIE[1] (EINT2)
EXIF[1] (INT2F)
Interrupt to [0x43]
CMPCFGAB[5] (INTENA)
COMST[4] (CMPIFA)
CMPCFGAB[1] (INTENB)
COMST[5] (CMPIFB)
CMPCFGCD[5] (INTENC)
COMST[6] (CMPIFC)
CMPCFGCD[1] (INTEND)
COMST[7] (CMPIFD)
BZCFG[6] (SNENDE)
BZCFG[1] (BZIF)
RMCFG[6] (LOWWKUPEN)
RMCFG[6] (LOWWKUP)
RMCFG[7] (HIGHWKUPEN)
RMCFG[7] (HIGHWKUP)
TKCCFG[2] (TKINTEN)
TKCCFG[1] (TKINTF)
IE[7] (EA)
EXIE[2] (EINT3)
Interrupt to [0x4B]
ADCCFG[6] (ADCINTE)
ADCCTL[1] (ADCIF)
IE[7] (EA)
EXIE[3] (EINT4)
EXIF[3] (INT4F)
Interrupt to [0x53]
PWMFG[3] (PWM0INTE)
PWMFG[7] (PWM0FG)
PWMFG[2] (PWM1INTE)
PWMFG[6] (PWM1FG)
IE[7] (EA)
EXIE[4] (EWDI)
WDCON[3] (WDIF)
Interrupt to [0x5B]
PCAMOD[2] (ECF)
PCACON[7] (CF)
PCAMOD[1] (CIFEN)
PCACON[6] (CIF)
CCAPM0[0] (ECCF)
CCON[0] (CCF0)
CCAPM1[0] (ECCF)
CCON[1] (CCF1)
CCAPM2[0] (ECCF)
CCON[2] (CCF2)
CCAPM3[0] (ECCF)
CCON[3] (CCF3)
CCAPM4[0] (ECCF)
CCON[4] (CCF4)
CCAPM5[0] (ECCF)
CCON[5] (CCF5)
IE[7] (EA)
EXIE[5] (EINT6)
Interrupt to [0x63]
SPICR[7] (SPIE)
SPIST[7] (SPIINT)
I2CSCON2[6] (EADDRMI)
I2CSST2[6] (ADDRMI)
I2CSCON1[5] (ESTOPI)
I2CSST2[5] (STOPI)
I2CSCON1[4] (ERPSTARTI)
I2CSST2[4] (RPSTARTI)
I2CSCON2[3] (ETXBI)
I2CSST2[3] (TXBI)
I2CSCON2[2] (ERCBI)
I2CSST2[2] (RCBI)
IE[7] (EA)
EXIE[6] (EINT7)
Interrupt to [0x6B]
T34CON[0] (T3IEN)
T34CON[3] (TF3)
T34CON[4] (T4IEN)
T34CON[7] (TF4)
T5CON[0] (T5IEN)
T5CON[7] (T5F)
BZCFG[6] (SNENDE)
BZCFG[1] (SSINTF)
IE[7] (EA)
EXIE[7] (EINT8)
Interrupt to [0x73]

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

IE (0xA8) 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-capable 16550-like UART2 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) 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 3/4/5 Interrupt Enable bit.
EINT7	SPI and I²C Slave Interrupt Enable bit.
EINT6	PCA and PWM Interrupt Enable bit.
EWD1	Watchdog Timer Interrupt Enable bit.
EINT4	ADC Interrupt Enable bit.
EINT3	Analog Comparator, Buzzer, Touch Key and Remote filter Interrupt Enable bit.
EINT2	Low Voltage Detection 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) 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) 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	INT8 Timer 3/4/5 Priority bit.
EINT7	INT7 SPI and I²C Slave Priority bit.
EINT6	INT6 PCA and PWM Priority bit.
EWDI	Watchdog Priority bit.
EINT4	INT4 ADC Priority bit.
EINT3	INT3 Analog Comparator, Buzzer, Touch Key and Remote filter Priority bit.
EINT2	INT2 Low Voltage Detection Priority bit.
EI2CM	I²C Master Priority bit.

EXIF (0x91) R/W (0x00)

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

INT8F	INT8 Timer 3/4/5 Interrupt Flag bit
INT7F	INT7 SPI and I²C Slave interrupt Flag bit
INT6F	INT6 PCA and PWM Interrupt Flag bit
INT4F	INT4 ADC Interrupt Flag bit
INT3F	INT3 Analog Comparator, Buzzer, Touch Key and Remote filter Interrupt Flag bit
INT2F	INT2 Low Voltage Detection Interrupt Flag bit
I2CMIF	I²C Master Interrupt Flag bit. This bit must be cleared by software
***	Writing to INT2F to INT8F has no effect.

The interrupt flag of internal peripherals are stored in the corresponding flag registers in the peripheral and EXIF registers. These peripherals include T0, T1, T2, and WDT. Therefore to clear the interrupt flags the software needs to clear the corresponding flags located in the peripherals (for T0, T1, and 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 are used to connect to the external peripherals. INT2F to INT8F are direct equivalents of the interrupt flags from the corresponding peripherals. These peripherals include RTC, I²Cs, PCA, ADC, etc.

PINT0 and PINT1 are used for external GPIO pin Interrupts. All GPIO pin can be enabled to generate the 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	PINT1EG	PINT0F	PINT0EG
WR	TF1	TR1	TF0	TR0	PINT1F	PINT1EG	PINT0F	PINT0EG

TF1	Timer 1 Interrupt Flag bit. TF1 is cleared by hardware when entering the interrupt routine. TF1 can also be cleared by software.
TR1	Timer 1 Run Control bit. Set to enable Timer 1, and clear to disable Timer 1.
TF0	Timer 0 Interrupt Flag bit. TF0 is cleared by hardware when entering the interrupt routine. TF1 can also be cleared by software.
TR0	Timer 0 Run Control bit. Set to enable Timer 0, and clear to disable Timer 0.
PINT1F	Pin INT1 Interrupt Flag bit. PINT1F is set to 1 by hardware when pin interrupt occurs. This must be cleared by software.
PINT1EG	Pin Interrupt 1 Edge or Level Setting. PINT1EG=0 use level interrupt, and PINT1EG=1 use edge interrupt. In current implementations, PINT1EG must be set to 0 to ensure capture of interrupt.
PINT0F	Pin INT0 Interrupt Flag bit. PINT0F is set to 1 by hardware when pin interrupt occurs. This must be cleared by software.
PINT0EG	Pin Interrupt 0 Edge or Level Setting. PINT0EG=0 use level interrupt, and PINT0EG=1 use edge interrupt. In current implementations, PINT0EG must be set to 0 to ensure capture of interrupt.

1.6 Register Access Control

One important aspect of the embedded MCU is its reliable operations under a harsh environment. Many system failures results 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 protected registers. The TA protected register includes 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 the access is granted, there is no time limitation of the access. The access is voided if any operation is performed in TA address. When read, the bit of TA indicates whether TA is locked or not (1 indicates “unlock” and 0 indicates “lock”).

TB (0xC9) Time Access B Control Register2 R/W (0x00)

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

TB access control functions are similar to 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 address to obtain the current TB status. The TB protected registers include two SFR registers, CKSEL (0x8F) and WKMASK (0x9F), and twelve XFR registers lodging REGTRM (0xA000), IOSCITRM (0xA001), IOSCVTRM (0xA002), XOSCCFG (0xA007), LVDCFG (0xA010), LVDTHD (0xA011), LVDHYS (0xA016), CNTPCTL (0xA025), CNTPCTH (0xA026), INTPCT1 (0xA013), INTPCT2 (0xA014), BPINTE (0xA0E1), and SI2C_DebugID (0xA0EF). 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 before-mentioned sequences are repeated before the 128 cycles expires, 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 thus may create vulnerabilities for critical registers.

Another reliability concern of embedded Flash MCU is that the important content on 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 following.

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, 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. In essence, 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 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, 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 arrest. But the clocks of external peripherals and CPU like PCA, ADC, LIN-capable 16550-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 the CPU. The exit mechanism of IDLE mode is the same as STOP mode. Idle bit is automatically cleared at the exit of the IDLE mode.

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

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

CD1, CD0 — Clock Divider Control bit. 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, thus may not function properly. All external peripherals to CPU still operate at full speed in PMM mode.

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

STATUS (0xC5) RO (0x0C)

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

STATUS register can be accessed by program to determine the status of critical events occurring in the integrated peripherals. The program should check status conditions before entering SLEEP, STOP, IDLE, or PMM modes to prevent loss of intended functions from delayed entry until these events are finished.

HIP — High Priority Interrupt Status. This bit reads 1 when there is high priority interrupt processing.

LIP — Low Priority Interrupt Status. This bit reads 1 when there is low priority interrupt processing.

SPTA0 — UART0 Transmit Activity Status. This bit reads 1 when UART0 transmitter is activated.

SPRA0 — UART0 Receive Activity Status. This bit reads 1 when UART0 receiver is activated.

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]

The SIOSC should be selected as the system clock before change the IOSC Pre-divider to avoid any clock glitch.

IOSCDIV[3-0] IOSC Pre-Divider

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/512
14	IOSC/1024
15	IOSC/2048

REGRDY[1-0] Wake up delay time for main regulator stable time from reset or from sleep mode wake up

REGRDY[1]	REGRDY[0]	Delay time
0	0	16 siosc cycle
0	1	32 siosc cycle
1	0	64 siosc cycle
1	1	128 siosc 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	IOSC
1	0	SOSC
1	1	SOSC

WKMASK (0x9F) Wake Up Mask Register R/W (0xFF) 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 mode. The wake-up is performed by these interrupts and if enabled the internal oscillator is turned on and SYSCLK resumes. The interrupt can be set as a level trigger or an edge trigger and the wake-up always runs in accordance with the edge. Please note 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 mode. Extra attention should be exerted as designing the exit and re-entry of modes to ensure proper operation.

Please note that all clocks are stopped in STOP mode, therefore peripherals require clock such as I²C slave, UARTx, ADC, LVD, and T3 cannot perform wake-up function. Only external pins and peripherals that do not require a clock can be used for wake up purposes. Such peripherals in IS31CS8969 are an analog comparator

1.7.1 PMM mode

PMM mode is enabled by setting CD[1:0] bits in PMR register to both 1. In PMM mode, the CPU and its integrated peripheral such as WDT, UART0, LIN-capable 16550-like UART2, T0/T1/T2, T3, and I²C Master operate at 257 times slower than SYSCLK. All other external peripherals such as PCA, ADC etc. are still operating under normal clock. The PMM mode saves power because the CPU, internal Flash memory and SRAM by operating at much slower frequency. The program continues to run while the CPU is operating at a reduced rate. To further save power, the unused external peripherals can be turned off or disabled. Normal mode operation can be recovered from PMM mode by program itself that set CD[1:0] = 01. Another way of recovery is to enable the SWITCHBACK function by setting SWB bit to high in PMR register. When switchback is enabled, the following conditions trigger the CPU to exit PMM mode and resume normal operations.

External Interrupt INT0/1/2/3/4/6/7/8 and any external peripherals interrupt OR-ED with these interrupts.

UART0 receive Start bit detection

UART0 transmit buffer loaded

When an external interrupt is intended to perform switchback, the corresponding interrupt must be enabled and not blocked by higher priority interrupts. In the case of UART-triggered switchback, the triggering is not generated by the UART-associated interrupt. This is because UART operating under PMM mode may not operate correctly to receive or transmit data. The switchback is thus initiated by the reception of the falling edge of the Start bit. The UART receive switchback is enabled only if the associated receive bit (SCON0.4 or SCON1.4) is set. The UART transmit initiated switchback is triggered when UART transmit buffer is loaded. Thus CPU operating under PMM mode recovers to normal mode automatically when it writes in the transmit buffer. Once it recovers, UART operates under normal frequency to correctly transmit the data.

The return of PMM mode after switchback must be activated manually with software. The exit of PMM mode occurs when WDT or external RSTN resets.

Since the purpose of the PMM mode is to save power consumption, the internal oscillator clock IOSC is recommended to be used as the system clock as IOSC consumes significantly less power than the crystal oscillator.

1.7.2 IDLE Mode

IDLE mode provides a further power saving than PMM mode by stopping the clock for CPU and its integrated peripherals while keeping the external peripherals at normal operating conditions. The external peripherals still function normally thus can generate interrupts that wake up the CPU from IDLE mode. The IDLE mode is introduced by setting Idle bit to 1.

When the CPU is idle mode, no processing is possible. All integrated internal peripherals such as T0/T1/T2, UART0, LIN-capable 16550-like UART2, and I²C Master are inaccessible during idling. The IDLE mode can be excited 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 be enabled properly. Upon exiting from IDLE mode, the CPU resumes operation as the clock is being 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 completes, RETI returns to the program and immediately follows the one that invokes the IDLE mode. Upon returning from IDLE mode to normal mode, Idle bit in PCON is automatically cleared. As the purpose of the IDLE mode is to save power, the use of IOSC clock is strongly recommended in place of SYSCLK before entering IDLE mode since it consumes significantly less power than the crystal oscillator or other clock sources.

1.7.3 STOP Mode

STOP mode provides the lowest power consumption by stopping clocks to all components in the system. STOP mode is entered by setting STOP=1. To achieve minimum power consumption, before entering STOP mode, it is essential to turn off all peripherals and the current operating clock oscillators such as crystal oscillator and PLL. It is also important that the software switches to the IOSC clock and disables all other clock generators such as crystal oscillator or PLL clock generator before entering STOP mode. This is critical to ensure a smooth transition when resuming its normal operations. Selecting other clock sources, such as XTAL oscillator or PLL clock as CPU system clock may burden the system as the clock sources may take a significant amount of time to stabilize during the wake-up. Upon entering STOP mode, the system uses the last edge of IOSC clock to shut down the IOSC clock generator. The minimum power consumption state is achieved through this mechanism.

Hardware reset through RSTN pin or by interrupts generated via external pins (INT0 and INT1) or INT2 to INT8 brings the system out of STOP mode. Since all clocks are inactive, none of the peripherals like UART, Timers, I^2C master and slave, ADC, or LVD contribute to the exit of STOP mode. Peripherals like Analog comparator and RTC interrupt; however, can be used to trigger the exit of STOP mode as they are implemented asynchronously or their own clock sources.

The triggering interrupt source must be enabled and its Wake-up bit is set in the WKMASK register. External pins require LOW-level triggers; however the INT flags of on-chip external peripherals require HIGH-level triggers. The IOSC circuit is activated by triggering event and the CPU is woken up at the first IOSC clock edge. Please note that the IOSC is activated as soon as STOP mode exits. As CPU resumes the normal operation using the IOSC clock when an interrupt presents, 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 Stop bit in PCON is automatically cleared by hardware reset during the waking up.

Please note the wake-up control WKMASK register and interrupt enable registers IE and EXIE which are specifically responsible for the wake-up and interrupt. Extra attention should be taken while programming for coherent application design. In STOP mode, clocks of CPU and peripherals are disabled (except RTC). Therefore only external pins and peripherals such as analog comparator and RTC that do not require clock can be used to initiate the wake-up process. Peripherals such as UART, Timers, I^2C master and slave, ADC, or LVD can not generate wake-up interrupt in this mode.

1.7.4 SLEEP Mode

In STOP mode, the main regulator providing 1.8V (VDDC) to internal logic, memory and flash circuits are still active. The regulator and its internal Bandgap reference circuits consumes approximately about 200uA. SLEEP mode is used to further reduce the standby power through turning off the regulator and reference circuits. The logic behavior of SLEEP mode is the same as STOP mode and is entered by setting both STOP and SLEEP bits to 1 in PCON register. In SLEEP mode, a very low-power back-up regulator is used to provide supply voltage to the internal logic, memory and flash circuits. The back-up regulator consumes about 10uA to 20uA, and can supply up to 1mA of load. The output voltage of the back-up regulator is lower than the main regulator, and typically is around 1.45V.

The exit of SLEEP mode is the same as exit of STOP mode by wake-up events, and exits directly back to normal operation and the main regulator is turned on. Note the enabling time of the main regulator is about 10usec, therefore, after wake-up from SLEEP mode, the software should be kept at NOP for at least 20usec before resuming. It is also recommended that if SLEEP mode is used, the decoupling capacitor on VDDC should contains at least 10uF.

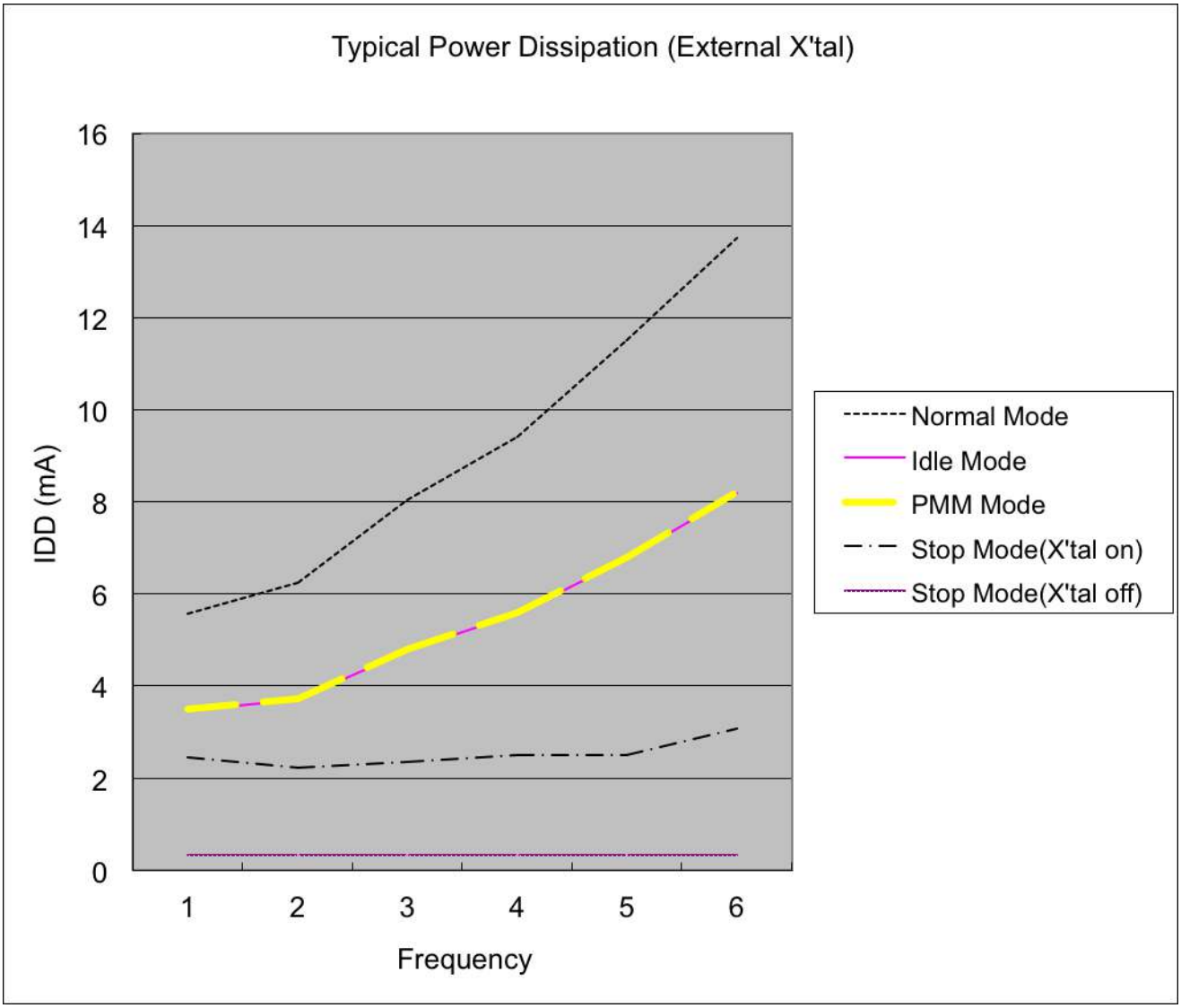
1.7.5 Clock Control

The clock selection is defined by CKSEL register (0x8F). An IOSC is a critical component in MCU although not integrated in the CPU core. It is enabled except in STOP mode. An IOSC also handles critical timing conformance for flash programming and the default manufactured calibrated IOSC is set at 16MHz. Although users can manually reset the IOSC frequency but reset value should not deviate more than 50% from its typical setting to avoid flash performance problems.

An IOSC is recommended that for the transition of clock-source-switching to ensure a smooth and glitch-free transition. This is also true for switching among different power saving modes. Please note that when waking up from STOP mode, the clock selection is switched automatically to IOSC. If other clock sources are preferred, optional configurations are available through software set-up.

When switching clock sources, it is also important to note the crystal oscillator, real time clock and the phase lock loop take a significant amount of time to stabilize. The software needs to be designed to turn on the corresponding clock source first and wait for the stabilization time before CKSEL settings take place.

The typical power dissipation relationship to the CPU frequency is shown in the following graph.



The values of performance frequency in IDLE and PMM modes are close therefore the lines appear overlapped in the graph.

The IDD result does not include the power dissipation of the clock oscillator. The graph shows that during normal operation, the power dissipation increases approximately at ~0.36mA/MHz; in idle mode it increases at about ~0.2mA/MHz (the power dissipation still increases as the frequency increases due to operation of peripheral clock).

WARNING: If an uninstalled clock source is being selected, it may cause the system to hang. There is NO hardware protection against this peril. Therefore extreme precautions must be exerted during programming.

1.8 Break Point 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 0x7B. 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 the 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	PC7IF	PC6IF	PC5IF	PC4IF	PC3IF	PC2IF	PC1IF
WR	STEP_IF	PC7IF	PC6IF	PC5IF	PC4IF	PC3IF	PC2IF	PC1IF

This register is for reading the Break Points interrupt flags.

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

PC7IF – PC1IF These bits are set when Break Point conditions are set by PC7 – 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	PC7IE	PC6IE	PC5IE	PC4IE	PC3IE	PC2IE	PC1IE
WR	STEP_IE	PC7IE	PC6IE	PC5IE	PC4IE	PC3IE	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.

PC7IE – PC1IE Set these bits to enable PC7 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 (b'11111100)

	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 requirement in a suspended state. If a specific timer should be kept active, it must be enabled by ISR after ISR entry. Before 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. This bit is set to 1 when entering DBG and BKP ISR. This signal should be cleared when exiting the DBG and BKP ISR. Checking this bit allows other interrupt routine 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 (b'00000000)

	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 (b'00000000)

	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 (b'00000000)

	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 (b'00000000)

	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 (b'00000000)

	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 (b'00000000)

	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.

PC3AL (A0F8h) Program Counter Break Point 3 Low Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC3AL[7-0]							
WR	PC3AL[7-0]							

This register defines the PC low address for PC match break point 3.

PC3AH (A0F9h) Program Counter Break Point 3 High Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC3AH[7-0]							
WR	PC3AH[7-0]							

This register defines the PC high address for PC match break point 3.

PC3AT (A0FAh) Program Counter Break Point 3 Top Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC3AT[7-0]							
WR	PC3AT[7-0]							

This register defines the PC top address for PC match break point 3. PC3AT:PC3HT:PC3LT together form a 24-bit compare value of break point 3 for Program Counter.

PC4AL (A0FCh) Program Counter Break Point 4 Low Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC4AL[7-0]							
WR	PC4AL[7-0]							

This register defines the PC low address for PC match break point 4.

PC4AH (A0FDh) Program Counter Break Point 4 High Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC4AH[7-0]							
WR	PC4AH[7-0]							

This register defines the PC high address for PC match break point 4.

PC4AT (A0FEh) Program Counter Break Point 4 Top Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC4AT[7-0]							
WR	PC4AT[7-0]							

This register defines the PC top address for PC match break point 4. PC4AT:PC4HT:PC4LT together form a 24-bit compare value of break point 4 for Program Counter.

STEPCTR (A0FFh) Single Step Break Point Interrupt Control Register R/W (0x00)

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

The value "0x96" must be programed into this register to enable the single step break point interrupt

PC5AL (A0E4h) Program Counter Break Point 5 Low Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC5AL[7-0]							
WR	PC5AL[7-0]							

This register defines the PC low address for PC match break point 5.

PC5AH (A0E5h) Program Counter Break Point 5 High Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC5AH[7-0]							
WR	PC5AH[7-0]							

This register defines the PC high address for PC match break point 5.

PC5AT (A0E6h) Program Counter Break Point 5 Top Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC5AT[7-0]							
WR	PC5AT[7-0]							

This register defines the PC top address for PC match break point 5. PC5AT:PC5HT:PC5LT together form a 24-bit compare value of break point 5 for Program Counter.

PC6AL (A0E8h) Program Counter Break Point 6 Low Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC6AL[7-0]							
WR	PC6AL[7-0]							

This register defines the PC low address for PC match break point 6.

PC6AH (A0E9h) Program Counter Break Point 6 High Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC6AH[7-0]							
WR	PC6AH[7-0]							

This register defines the PC high address for PC match break point 6.

PC6AT (A0EAh) Program Counter Break Point 6 Top Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC6AT[7-0]							
WR	PC6AT[7-0]							

This register defines the PC top address for PC match break point 6 PC6AT:PC6HT:PC6LT together form a 24-bit compare value of PC break point 6 for Program Counter.

PC7AL (A0ECh) Program Counter Break Point 7 Low Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC7AL[7-0]							
WR	PC7AL[7-0]							

This register defines the PC low address for PC match break point 7.

PC7AH (A0EDh) Program Counter Break Point 7 High Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC7AH[7-0]							
WR	PC7AH[7-0]							

This register defines the PC high address for PC match break point 7.

PC7AT (A0EEh) Program Counter Break Point 7 Top Address Register R/W (b'00000000)

	7	6	5	4	3	2	1	0
RD	PC7AT[7-0]							
WR	PC7AT[7-0]							

This register defines the PC top address for PC match break point 7. PC7AT:PC7HT:PC7LT together form a 24-bit compare value of break point 7 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 break point occurs, therefore, it is usually exactly the same value of the break pointer setting.

DBPCIDL (A098h) Debug Program Counter Address Low Register RO (b'00000000)

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

DBPCIDH (A099h) Debug Program Counter Address High Register RO (b'00000000)

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

DBPCIDT (A09Ah) Debug Program Counter Address Top Register RO (b'00000000)

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

DBPCNXL (A09Bh) Debug Program Counter Next Address Low Register RO (b'00000000)

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

DBPCNXH (A09Ch) Debug Program Counter Next Address High Register RO (b'00000000)

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

DBPCNXT (A09Dh) Debug Program Counter Next Address Top Register RO (b'00000000)

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

1.9 Debug and ISP

The I²C Slave 2 (I2CS2) can be configured as the debug and ISP port for IS31CS8969. This is achieved by assigning a predefined debug ID for the I²C Slave 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 0x83. DBG ISR is used to communicate with the host and is usually closely associated with BKP ISR.

SI2CDBGID (A0EFh) Slave I²C Debug ID Register R/W (b'00110110) TB Protected

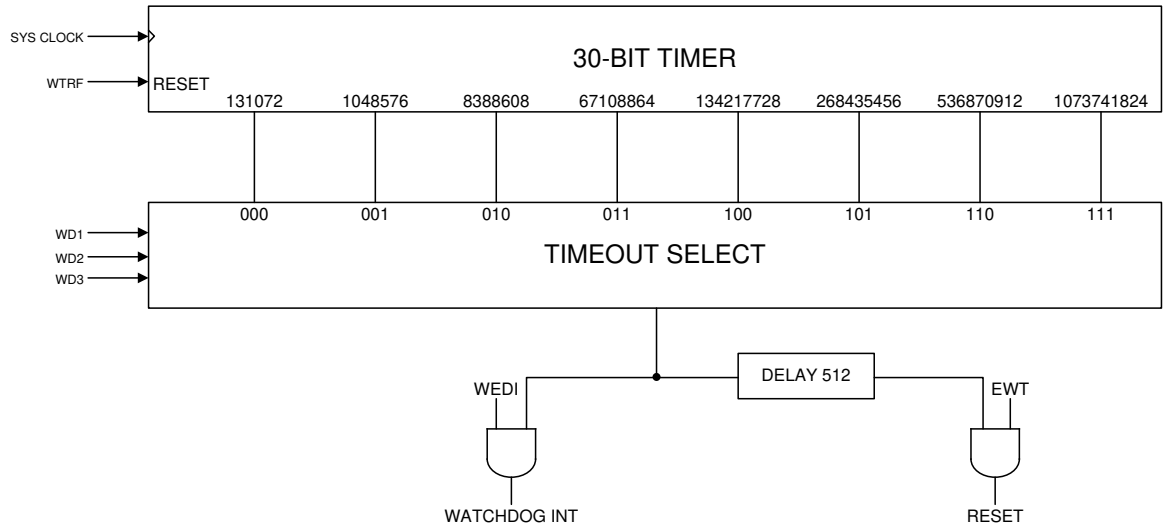
	7	6	5	4	3	2	1	0
RD	DBGSI2C2EN	SI2CDBGID[6:0]						
WR	DBGSI2C2EN	SI2CDBGID[6:0]						

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.

1.10 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. Note WDT shares the same clock with the CPU, thus WDT is disabled in IDLE mode or STOP mode however it runs at a reduced rate in PMM mode.



WDCON (0xD8) R/W (0x00)

	7	6	5	4	3	2	1	0
RD	-	-	-	-	WDIF	WTRF	EWT	RWT
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. A hardware reset generates a WDT reset whereas a software reset does not. It can be cleared by software or external reset from RSTN pin

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. Setting RWT resets the timer counting. The clearing action of Watchdog timer is protected by TA access. In another word, to clear Watchdog timer, TA must be unlocked then 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 (0xC7)

	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, 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, 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, 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, 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 equals 12, 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 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

1.11 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. Timer 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 (0x88h) Timer 0 and 1 Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TF1	TR1	TF0	TR0	PINT1F	PINT1EG	PINT0F	PINT0EG
WR	TF1	TR1	TF0	TR0	PINT1F	PINT1EG	PINT0F	PINT0EG

TF1 Timer 1 Interrupt Flag bit. TF1 is cleared by hardware when entering the interrupt routine. TF1 can also be cleared by software.

TR1 Timer 1 Run Control bit. Set to enable Timer 1, and clear to disable Timer 1.

TF0 Timer 0 Interrupt Flag bit. TF0 is cleared by hardware when entering the interrupt routine. TF1 can also be cleared by software.

TR0 Timer 0 Run Control bit. Set to enable Timer 0, and clear to disable Timer 0.

PINT1F/PINT1EG/PINT0F/PINT0EG

These bits are related to configurations of the expanded interrupt PINT1 and PINT0. These are described in the Interrupt System section.

TMOD (0x89h) Timer 0 and 1 Mode Control Register R/W (0x00)

	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 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 internal clock.

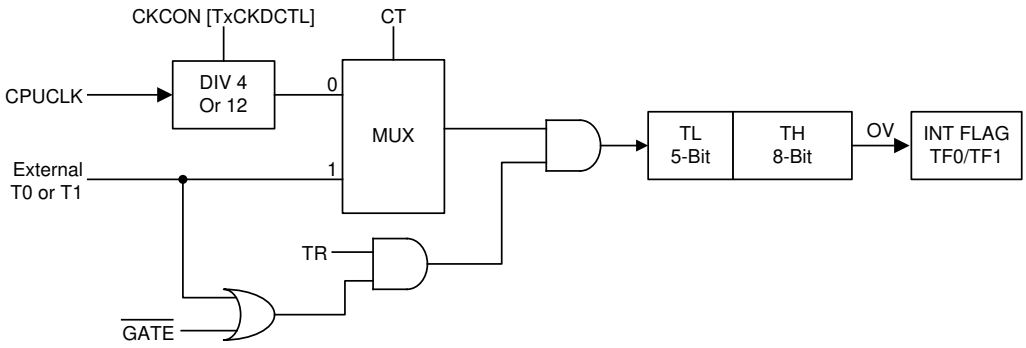
T0M1 Timer 0 Mode Select bit.

T0M0 Timer 0 Mode Select bit.

M1	M0	Mode	Mode Descriptions
0	0	0	TL serves as a 5-bit prescaler 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.

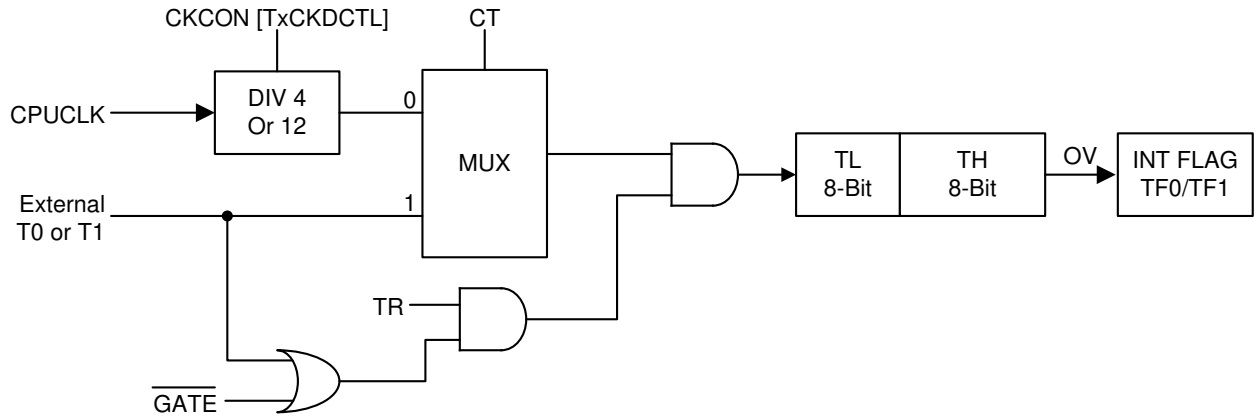
1.11.1 Mode 0

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



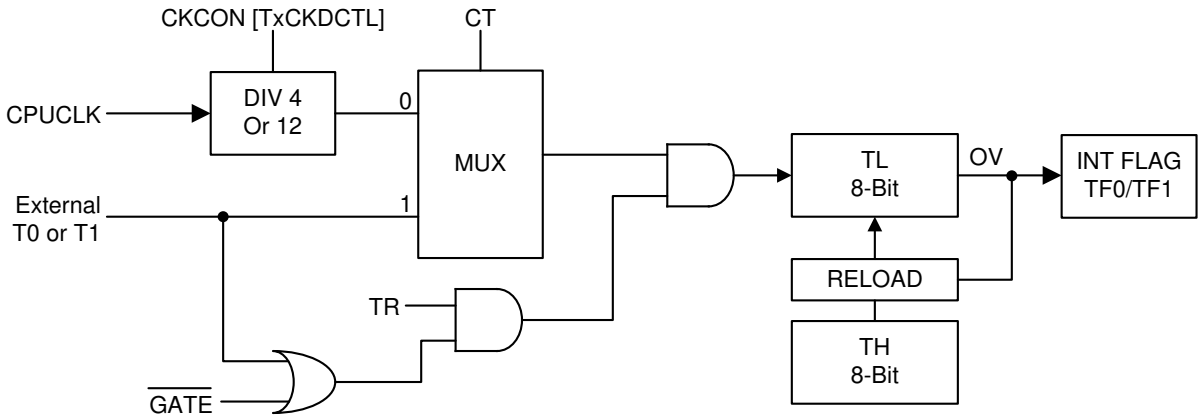
1.11.2 Mode 1

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



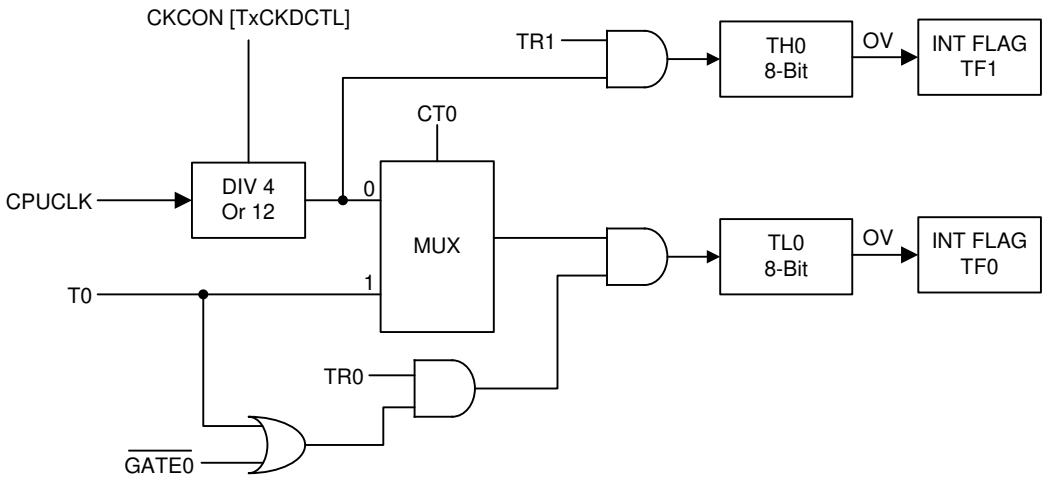
1.11.3 Mode 2

Mode 2 configures the timer as an 8-bit reloadable 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:



1.11.4 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.



1.12 System Timer – T2

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

T2CON (0xC8h) Timer 2 Control and Configuration Register R/W (0x00)

	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 a 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. Because IS31CS8969 does not support T2EX, EXEN2 shall be set to 0.

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

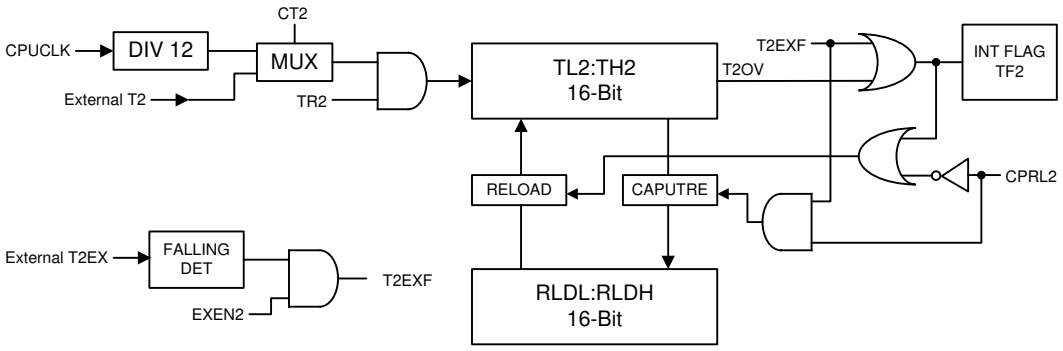
CT2 Timer 2 Timer/Counter Mode Select bit
1 – External event counter uses T2 pin as the clock source
0 – Internal clock timer mode

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.

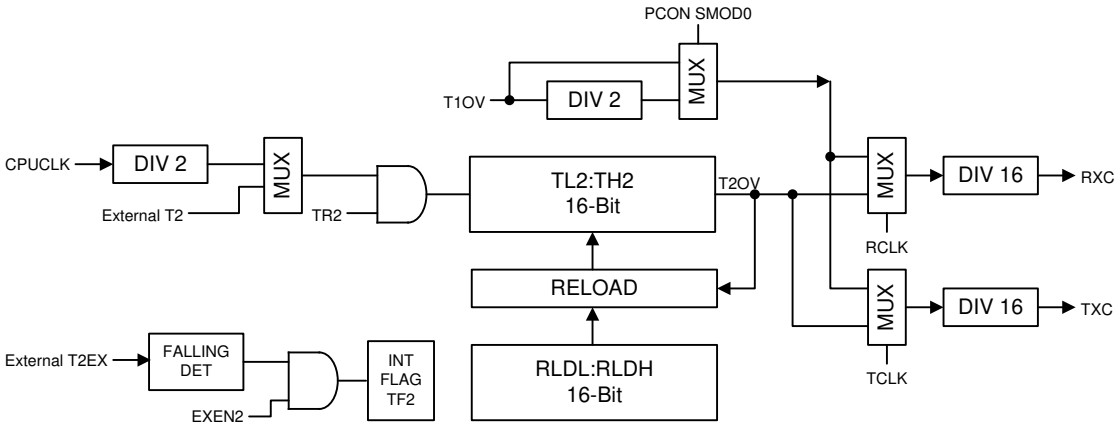
Timer 2 can be configured in three modes of operations – Auto-reload Counter, Capture Timer, or Baud Rate Generator. These modes are defined by RCLK, TCLK, CPRL2 and TR2 bits of T2CON registers. The definition is illustrated in the following table:

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:

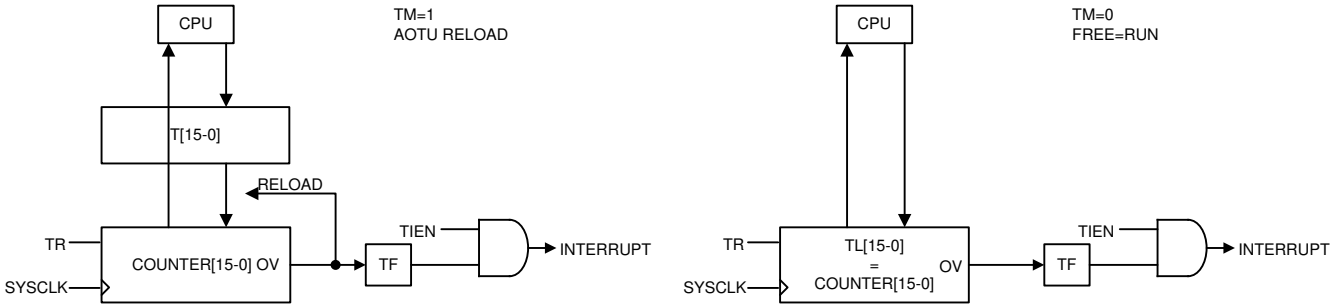


The block diagram of the Timer 2 operating in Baud Rate Generator is shown in the following diagram:



1.13 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 R/W (0x00)

	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 disable the Timer 4 overflow interrupt T4IEN=1 enable 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 set timer 3 as auto reload, and TM3=0 set 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 disable the Timer 3 overflow interrupt
T3IEN=1 enable the Timer 3 overflow interrupt

TL3 (0xAEh) Timer 3 Low Byte Register 0 R/W (0X00)

	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 (0X00)

	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 (0X00)

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

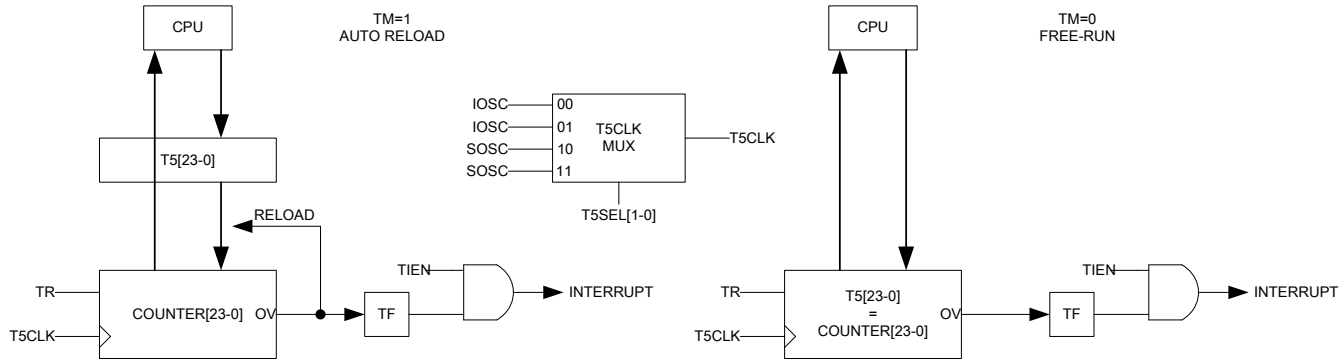
TH4 (0xADh) Timer 4 High Byte Register 0 R/W (0X00)

	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 been 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 been 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.

1.14 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 SIOSC. 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 (0xA003h) Timer 5 Control and Status Register R/W (0x00)

	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, SIOSC
T5SEL[1-0] = 11, SIOSC

TM5 Timer 5 Mode Control bit. TM5=1 set timer 5 as auto reload, and TM5=0 set 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 disable the Timer 5 overflow interrupt
T5IEN=1 enable the Timer 5 overflow interrupt

TL5 (0xA004) Timer 5 Low Byte Register 0 R/W (0X00)

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

TH5 (0xA005) Timer 5 Medium Byte Register 0 R/W (0X00)

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

TT5 (0xA006) Timer 5 High Byte Register 0 R/W (0X00)

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

T5[23-0] functions differently when been 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 been 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.

1.15 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 the 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 that 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 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 performed by MDU. When the normalization is completed, SC4-0 contains the number of shift performed in the normalization. If SC4-0 is written with a non-zero value, then the shift operation is performed by MDU with the number of shift specified by SC4-0 value.

MD0 (0xF9) MDU Data Register 0 R/W (0X00)

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

MD1 (0xFA) MDU Data Register 1 R/W (0X00)

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

MD2 (0xFB) MDU Data Register 2 R/W (0X00)

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

MD3 (0xFC) MDU Data Register 3 R/W (0X00)

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

MD4 (0xFD) MDU Data Register 4 R/W (0X00)

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

MD5 (0xFE) MDU Data Register 5 R/W (0X00)

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

MDU operation consists of three phases.

Loading MD0 to MD5 data registers in an appropriate order depending on the operation.

Execution of the operations.

Reading result 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.

1.15.1 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 if the sequence is violated, the calculation may be interrupted and result in errors.

1.15.2 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

1.15.3 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

1.15.4 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

1.15.5 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)

1.16 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 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 bit 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. An address byte can interrupt slaves. 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 R/W (0x00)

	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 = CPUCLK/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, fix baud rate Baud rate = CPUCLK/64 (PCON.SMOD0 = 0) or CPUCLK/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 R/W (0x00)

	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.

1.16.1 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.

1.16.2 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.

1.16.3 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.

1.16.4 Mode 3

Similar to 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 similar to which in Mode 1. that is determined by Timer 1 or Timer 2 overflow and is set by SMOD0, and T2CON.

1.17 I^{2}C Master

The I^{2}C master controller provides the interface to I^{2}C 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 I^{2}C bus speed. The maximum I2C bus speed is limited to SYSCLK/12.

I2CMTP (0xF7h) I^{2}C Master Time Period R/W (0X00)

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

This register set the period time of I^{2}C bus clock – SCL. The SCL period time is set according to SCLPERIOD = 8 * (1 + I2CMTP) * CPUCLK_PERIORD when the I2CMTP[7-0] is equal to or larger than 0x01. If I2CMTP[7-0] = 0x00, the maximum I2C bus speed is limited to SYSCLK/12.

I2CMSA (0xF4) I²C Master Slave Address R/W (0X00)

	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 (0X00)

	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, TD is 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 (0X00)

	7	6	5	4	3	2	1	0
RD	-	BUSBUSY	IDLE	ARBLOST	DATAACK	ADDRACK	ERROR	BUSY
WR	I2CMRST	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.

I2CMRST Writing 1 to this bit forces the I2CM to perform reset and clear its internal state machine. At the end of the initialization, all SFRs will return to the default value. This bit is cleared automatically by hardware.

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 was 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 a burst. To generate a single read cycle, the designated address is written in SA, RS is set to 1, 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.

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 the FLASH does not require any special attention. When the FLASH is used as data storage, the software sends 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 executed. The embedded Flash memory contains two blocks – Main Memory and Information Block (IFB). The Main Memory is 32K x 8 with uniform 1024 Byte page (sector) size. The Information Block is 256 Byte and sits in a separate sector.

The commands performed by a Flash Controller are defined in FLSHCMD registers. The defined operations allow the user program to use on-chip flash as a program memory, and a non-volatile data memory in In-System-Programming as well as In-Application-Programming. The maximum flexibility of the on-chip flash memory can be achieved through user program. The manufacturer provides a default ISP boot program located on the top sectors of the flash. The preset ISP boot program can be used or modified or replaced based on application requirements.

FLSHCMD (A020h) Flash Controller Command Register R/W 10000000 TB Protected

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

WRVFY Byte 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 represents 0. It is reset to 1 by hardware when another ISP command is executed.

FAIL Command Execution Result. It is set if the previous command execution fails due to any reasons. 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 when a new command is issued. Possible causes of FAIL include address over range, or address falls into protected region.

CYC[2-0] Flash Command Time Out

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

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	1	1	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 Byte Read
0	1	0	0	0	Main Memory Sector Erase
0	0	1	0	0	Main Memory Sector Byte Write
0	0	0	1	0	IFB Byte Read
0	0	0	0	1	IFB Byte Write (0x40 – 0xFF)
0	0	0	1	1	-
1	0	0	1	0	-

For all commands, the address of the flash is composed from FLSHADM:FLSHADH:FLSHADL and the data is referred at FLSHDAT registers. The erase command operation is sector-based, the address of the sector is determined from the high order address bits. For example, to point to the sector of 0x0C000-0xCFFF, the upper 8 bits “0C” are used. And the erase command erases the whole addressed sector contents. For Erase and Write command, the Flash Controller also checks if the destination address falls within the protection zone defined by CNTPCTL and

CNTPCTH registers. If it is protected, the Flash Controller does not execute the command and return with FAIL result bit. For IFB Byte-write, the Flash Controller does not execute the command and return with Fail result bit if the byte address falls into manufacturer data range. Please also note the Fuse block is used for manufacturer to store manufacturing related and calibration data and thus can only be read and not writable or erasable. Fuse block can only be erased or written under writer mode.

ISPCLKF (A024h) Flash Command Clock Scaler R/W 00100101 TB Protected

	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 at 2MHz.

FLSHDAT (A021h) Flash Controller Data Register R/W 00000000 TB Protected

	7	6	5	4	3	2	1	0
RD	Flash Read Data Register							
WR	Flash Write Data Register							

FLSHADL (A023h) Flash Controller Low Address Data Register R/W 00000000 TB Protected

	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 (A022h) Flash Controller High Address Data Register R/W 00000000 TB Protected

	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]							

FLSHADM (A012h) Flash Controller MSB Address Data Register R/W 00000000 TB Protected

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

A very common problem of embedded flash memory is when being used as both data and program storage, which leads to content loss due to software or other problems caused by program flow or noise. It induces executions of modifying stored contents. The design of Flash controller takes into considerations of these events and provides further protection to avoid accidental erasure or modifications of critical information or software codes. When a command is sent to the Flash Controller through FLSHCMD register, the controller checks whether the destination of the command falls in the content protection zones. If it falls within the protection zones, the flash control aborts its operations and returns with command failure message. Two protections zones are defined by 00000 to CNTPCTL, and CNTPCTH to 1FFFF.

CNTPCTL (A025h) Flash Content Protection Low Zone Register R/W 11111111 TB Protected

	7	6	5	4	3	2	1	0
RD	Content Protection Low Register							
WR	Content Protection Low Register							

This register defines the high bound address from 00000h of the flash which is protected against erasure or modifications. The data is processed in 256 Byte increments. The protected region is defined from 00 to (CNTPCTL1 -1) Note that CNTPCTL defaults to FFh which protects the whole 32KB of flash memory. User program needs to write the appropriate data into CNTPCTL to enable erase and write access.

CNTPCTH (A026h) Flash Content Protection High Zone Register R/W 00000000 TB Protected

	7	6	5	4	3	2	1	0
RD	Content Protection High Register							
WR	Content Protection High Register							

This register defines the low bound address from 0xFFFF of the flash to be protected against erasure or modifications. The data is processed in the increments of 256 Byte. The protected region is greater than (CNTPCTH+1) and less than or equal to FF. Note that CNTPCTH defaults to 00h which means the protection of 64KB of flash memory is on.

User program needs to write the appropriate data into CNTPCTH by reading IFB-protected information to protect boot code and expand the protection zone under application considerations.

There is an additional content protection against internal program. This protects sensitive data from unauthorized access. The protection range is from 0x1000 to 0xFFFF of embedded flash memory. The protection is achieved by two special registers, INTPCT1 (0xA013) and INTPCT2 (0xA014). After any reset condition such as power-up, RSTN, LVR, or WDT reset, INTPCT1 and INTPCT2 are initialized to 0x00. The bits in INTPCT1 and INTPCT2 can only be written to "1". When the embedded flash memory has been protected, this means accessing this protected range returns with 0x00 either by program instruction such as "MOVC" or by Flash Main Memory Byte Read access. The internal protection is by default not turned on after reset because both INTPCT1 and INTPCT2 are 0x00. Both registers are protected by TB. To turn on the internal protection, INTPCEN must be enabled by writing a "0x80" into enable the Internal Protection function. When INTPCEN is set, user can define the protection range by program INTPCT1 or INTPCT2. The INTPCEN will be cleared and stuck-on zero after setting any protection range. In other words, the protection range is single time programmable. Once the protection is turned on, it can't be turned off or modified because INTPCEN is stuck-on zero. To restore unprotected state, the chip must go through a reset. The internal protection should be enabled with extreme cautiousness. It is important that once it is turned on, program execution should not reach the protected zone, otherwise unpredicted program errors may occur.

INTPCT1 (A013h) Internal Protection Enable Register R/W 00000000 TB Protected

	7	6	5	4	3	2	1	0
RD	INTPCT1[7-0]							
WR	INTPCT1[7-0]							

This register can be written to "1" only. Writing "0" into any bit of this register does not alter the content. This register is cleared to 0x00 after reset. And the value of this register can be cleared only by a reset.

INTPCT2 (A014h) Internal Protection Enable Register R/W X000000 TB Protected

	7	6	5	4	3	2	1	0
RD	INTPCEN	INTPCT2[6-0]						
WR	INTPCEN	INTPCT2[6-0]						

This register can be written to "1" only. Writing "0" into any bit of this register does not alter the content. This register is set to 0x00 after a reset. And the value of this register can be set only by a reset.

INTPCEN INTPCT1/INTPCT2 updates enable. This bit is single time programmable. After setting any protection range, this bit will be stuck-on zero and prohibit another protection updating. Because the INTPCEN is disabled in the initiation, a "0x80" must be programmed into INTPCT2 to enable INTPCEN. When INTPCEN is set, the expected program protection can be defined by setting INTPC1 or INTPC2[6-0].

The following table summarizes the internal program protection with different INCTPCT1 and INTPCT2 settings and protected range from program read access. Because IS31CS8969 only has 32KB Flash Memory, the setting for the Protected Region from 0x08000 to 0x0FFFF is reserved.

Executed Priority	Register	Protected Region of Embedded Flash Memory
1	INTPCT1[0]=1	0x01000 ~ 0x0FFFF
2	INTPCT1[1]=1	0x02000 ~ 0x0FFFF
3	INTPCT1[2]=1	0x03000 ~ 0x0FFFF
4	INTPCT1[3]=1	0x04000 ~ 0x0FFFF
5	INTPCT1[4]=1	0x05000 ~ 0x0FFFF
6	INTPCT1[5]=1	0x06000 ~ 0x0FFFF
7	INTPCT1[6]=1	0x07000 ~ 0x0FFFF
8	INTPCT1[7]=1	0x08000 ~ 0x0FFFF
9	INTPCT2[0]=1	0x09000 ~ 0x0FFFF
10	INTPCT2[1]=1	0x0A000 ~ 0x0FFFF
11	INTPCT2[2]=1	0x0B000 ~ 0x0FFFF
12	INTPCT2[3]=1	0x0C000 ~ 0x0FFFF
13	INTPCT2[4]=1	0x0D000 ~ 0x0FFFF
14	INTPCT2[5]=1	0x0E000 ~ 0x0FFFF
15	INTPCT2[6]=1	0x0F000 ~ 0x0FFFF

3. Essential Analog Blocks

3.1 On-Chip 1.8V Regulator

An on-chip regulator is used to provide supply for core logic and internal E-Flash memory (VDDC). The regulator is partitioned into a back-up regulator and a main regulator. The main regulator is enabled only in normal, and STOP modes, and disabled in SLEEP mode when the back up regulator is turned on. The main regulator consumes about 100uA itself, while the back-up regulator consumes about 5uA. The back-up regulator can supply up to 500uA of current therefore it is important all peripheral circuits should be kept off during SLEEP mode. After reset, the main regulator defaults to on state. The regulator requires an external capacitor, which should be connected to VDDC pin. A minimum of 1uF plus a 0.1uF in parallel is required for the stability of the regulator. The main regulator outputs about can be adjusted by REGTRM (with default at its maximum) and the back-up regulator outputs about 1.60V. A manufacturer calibrated value of REGTRM for 1.8V is stored in IFB.

REGTRM (A000h) Regulator Trim Register R/W 11111111 TB Protected

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

REGTRM[7-0] Trim Register for main 1.8V regulator.
REGTRM[7-0]=FF corresponds to maximum output level. REGTRM[7-0]=00 corresponds to minimum output level. The in-between value in general is linear to the output level. Typically the maximum is around 2.0V while the minimum is around 1.6V

3.2 1.0V Reference (BGREF)

This reference is derived from on-chip band-gap reference and has very low temperature coefficient and supply voltage dependency. Typical value of BGREF is 1.0V. This reference is connected to ADC input and can be used for calibration.

3.3 Precision Internal 16MHz Oscillator (IOSC)

The internal oscillator is a very important peripheral as it provides the default clock source after reset and other critical timing. The internal oscillator has the salient features that it behaves well during the enable and disable transient. No clock glitches or extra clock edge is generated during the on/off transition, and the oscillator can reach to stable oscillations within very short time typically within 10 cycles. The IOSC consumes around 300uA when enabled. The IOSC is always enabled except entering into STOP mode. And in STOP mode when it is disabled, IOSC only consumes less than 1uA standby current.

Similar to the on-chip regulator, IOSC also exhibits chip-to-chip variations. A calibrated value that set IOSC at 16MHz +/- 2% is stored in IFB. The user program can set this value to IOSC trimming register, IOSCITRM (A001h) and IOSCVTRM (A002h). The IOSC frequency has very little variations over the operation range (-40o – 85oC and VDD = 2.5V – 5.5V). The variation is typically less than +/-2% over the operation conditions. It is possible that user program to set a different frequency other than 16MHz as long as user program provide a calibration method to set IOSC frequency at the desired value at typical operation condition, and it will be stable and accurate over the entire operation range. Please note that the trimming register will be set to its default value after resets, the user program must reinitialize to its calibrated value. The total trim range of the IOSC is roughly from 7MHz up to 24MHz.

The IOSC is also equipped with Spread Spectrum capability for EMI sensitive applications. The SS deviation can be controlled to fit various requirements. However, once SS is enabled, the accuracy of IOSC cannot be maintained.

IOSCITRM (A001h) IOSC Coarse Trim Register R/W 00000010 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 of spread spectrum frequency change. 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 (A002h) IOSC Fine Trim Register R/W 10001110 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 as long as 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= 14.0MHz - 9.5MHz - 7.0MHz (VTRM[7-0]= 00 - 80 - FF)

ITRM[1-0]=01, F_IOSC= 18.0MHz - 12.5MHz - 9.3MHz (VTRM[7-0]= 00 - 80 - FF)

ITRM[1-0]=10, F_IOSC= 22.0MHz - 15.5MHz - 11.5MHz (VTRM[7-0]= 00 - 80 - FF)

ITRM[1-0]=11, F_IOSC= 25.5MHz - 18.5MHz - 13.5MHz (VTRM[7-0]= 00 - 80 - FF)

The trimming of the IOSC should use the following procedure to obtain the default setting for 16MHz.

Set ITRM = 01, and

Set VTRM = 00, measure frequency

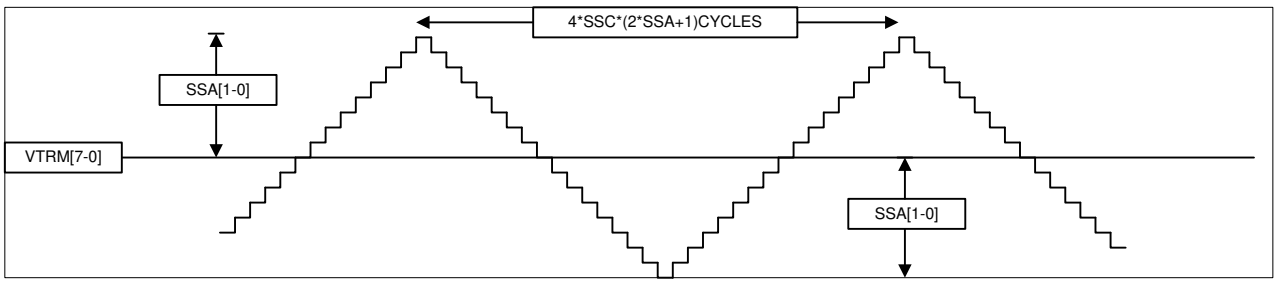
Set VTRM = FF, measure frequency

Set VTRM = 7F, measure frequency

Use binary search to obtain the closest setting for 16MHz

Note: The frequency versus VTRM setting is monotonic. When VTRM = 00, the frequency is highest, and when VTRM = FF, the frequency is lowest.

A hardware Spread Spectrum can be enabled for the IOSC. This is controlled by SSC[3-0]. When SSC[3-0] = 0, the spread spectrum is disabled, and IOSC functions normally as a fixed frequency oscillator. If SSC[3-0] is not 0, then Spread Spectrum is enabled and IOSC frequency is swept according to the setting of SSC[3-0] and SSA[1-0]. The spread is achieved by varying the actual VTRM output to the oscillator circuit thus effectively changes the oscillation frequency. The effect of SSC[3-0] and SSA[1-0] is shown in the following graph.



When Spread Spectrum is enabled, the actual controlling output to IOSC is VTRM[7-0] +/- SSA. This is shown in the graph above as the bold curve. The above example shows SSA[1:0] = 01, and the deviation is +/- 8. SSC[3-0] defines the update time in IOSC cycles. Then we can calculate the period of a complete sweep is 4*SSC*(2SSA+1) IOSC cycles, and we can obtain the sweep frequency from this period. When SS is enabled, the frequency of IOSC varies according to time and setting, therefore, the accuracy of IOSC frequency cannot be guaranteed. Please also note that VTRMOUT is VTRM[7-0] +/- SSA but is bounded by 0 and 255. Therefore for a linear non-clipped sweep, VTRM[7-0] needs to be within the range of SSA – 256-SSA, for example, SSA[10] = 01, then SSA is 8. VTRM[7-0] should be in the range from 8 to 248 to prevent the sweep from being clipped. As Spread Spectrum suggests, the total EMI energy is not reduced, but the energy is spread over wider frequency. It is recommended that SS usage should be carefully evaluated and the setting of spread amplitude and the sweep frequency should be chosen carefully for reducing EMI effect.

3.4 Slow Internal Oscillator (SIOSC)

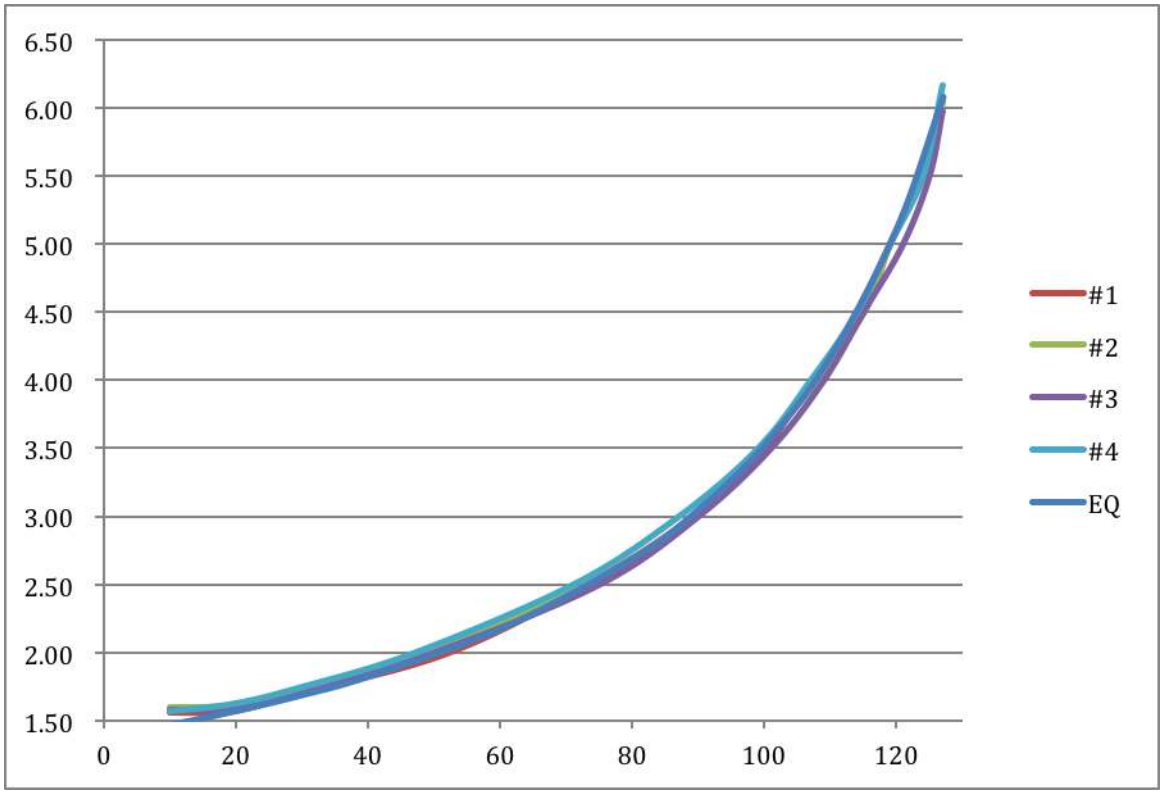
The SIOSC is a 32 KHz low power internal R/C oscillator. The oscillator consumes about 2uA and is always enabled. SIOSC can be used as system clock or as T5 clocking source to provide extended long period timing for wake up purpose. The accuracy of SIOSC is not guaranteed and typically lies within 20KHz to 50KHz, and variations temperature is about +/- 30%.

3.5 Supply Low Voltage Detection (LVD)

The supply Low Voltage Detection (LVD) circuit detects VDD < VTH condition and can be used to generates an interrupt or reset condition. LVD defaults to 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 when LVDTHV is the detection voltage.

LVDTHV = BGREF/(0.168 + 0.55 *(128-LVDTH[6-0])/128))

Where typical value of BGREF is 1.0V. The following graph shows the plot of the equations and four device characteristics.



The manufacturing process also stores the LVDTHD value for detection of 4.0V and 3.0V in IFB. The user program can either use the equation to calculate the desired detection value or uses these two values.

LVDCFG (A010h) Supply Low Voltage Detection Configuration Register R/W 10000000 TB Protected

	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 detect condition to cause a system reset.
LVTEN	LVT Enable bit. LVTEN = 1 allows low voltage detect 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.
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 (A016h) 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 detect, a digital controlled hysteresis is used. If LVDHYEN=1, when LVD is asserted a new threshold defined by LVDHYS[6-0] replaces LVDTHD[6-0]. In typical applications, LVDHYS[6-0] should be set to be greater than LVDTHD[6-0].

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. Please note both functions can coexist. Also the I²C Slave controller support sthe 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 cleared RCBI flag.

In transmit mode, the controller detects a valid matching address and issue 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, 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. The INFILEN bit in the I2CSCON register enables this. The filter is implemented using digital circuit. When INFILEN is set, the spikes less than 1/2 EPPCLK 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.

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

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

I2CSRST	I²C Slave Reset bit. Set this bit causes the Slave Controller to reset all internal state machine. 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 received 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 sets 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; is clear when the I²C slave controller is in receiving 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	ADDR0DC	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.

ADDR0DC — Address Bit 0 Don't Care
ADDR0DC=1 configures the address matching of SADDR1[6-1] or SADDR2[6-1] only. Because it allows ignoring SADDR[0], the I2C slave to respond to two consecutive I2C slave address. In IS31CS8969, the I2C slave controller can respond maximum four I2C slave address.
Default ADDR0DC = 0.

ADDRMI — Slave Address Match Interrupt Flag bit. This bit is set when the received address matches the address defined in I2CSADR2. If EADDRMI 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.

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 very 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. Please note 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. And it is cleared by the start condition.

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.

I2CSADR1 (0xDD) I2CS2 Slave Address1 Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	XMT	I2CADDR[6-0]						
WR	I2CSEN1	SADDR1[6-0]						

I2CSENT1 — Set this bit to enable the I²C slave controller.

SADDR1[6-0] — 7-bit slave address.

I2CADDR[6-0] — Received slave I2C address

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

I2CSADR2 (0xDF) I2CS2 Slave Address2 Register WO (0x00)

	7	6	5	4	3	2	1	0
RD	-	-	-	-	-	-	-	-
WR	I2CSEN2	SADDR2[6-0]						

I2CSENT2 — Set this bit to enable the I²C slave controller.

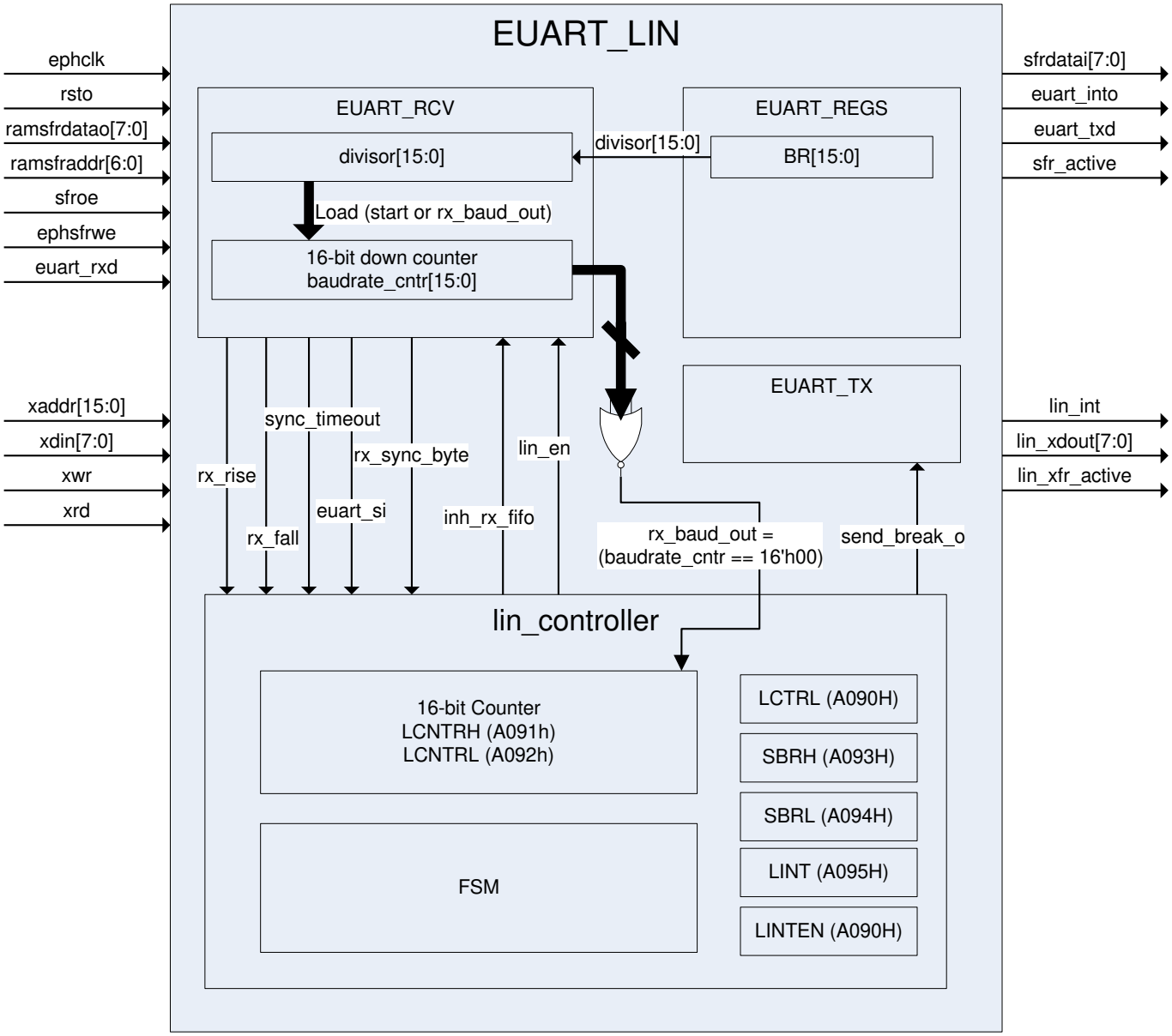
SADDR2[6-0] — 7-bit slave address.

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

	7	6	5	4	3	2	1	0
RD	I^2C Slave Receive Data Register							
WR	I^2C 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 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 R/W 0x00

	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 (0=Even, 1=Odd)

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 R/W 0x00

	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 indicate the current receive 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	Reserved

TFL[3-0] — Current Transmit FIFO level. This is read only and indicate 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	Reserved
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 clear 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 R/W 0x00

	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. Default is 0.

TRA/TRAEN — Transmit FIFO is ready to be filled.
This bit is set when transmit FIFO has been emptied below 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 receive FIFO has been filled above 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 software that there are still remaining unread received bytes in the FIFO.
This 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 overflow condition of receive FIFO occurs. Write “1” to enable interrupt. The flag can be cleared by software, 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 underflow condition of receive FIFO occurs. Write “1” to enable interrupt. The flag can be cleared by software, 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 overflow condition of transmit FIFO occurs. Write “1” to enable interrupt. The flag can be cleared by software, 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 framing error occurs as the byte is received. Write “1” to enable interrupt. The flag must be cleared by software, 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 thus the TX FIFO becomes empty. Write “1” to enable interrupt. The flag must be cleared by software, writing “0” on the bit (the interrupt is disabled simultaneously).

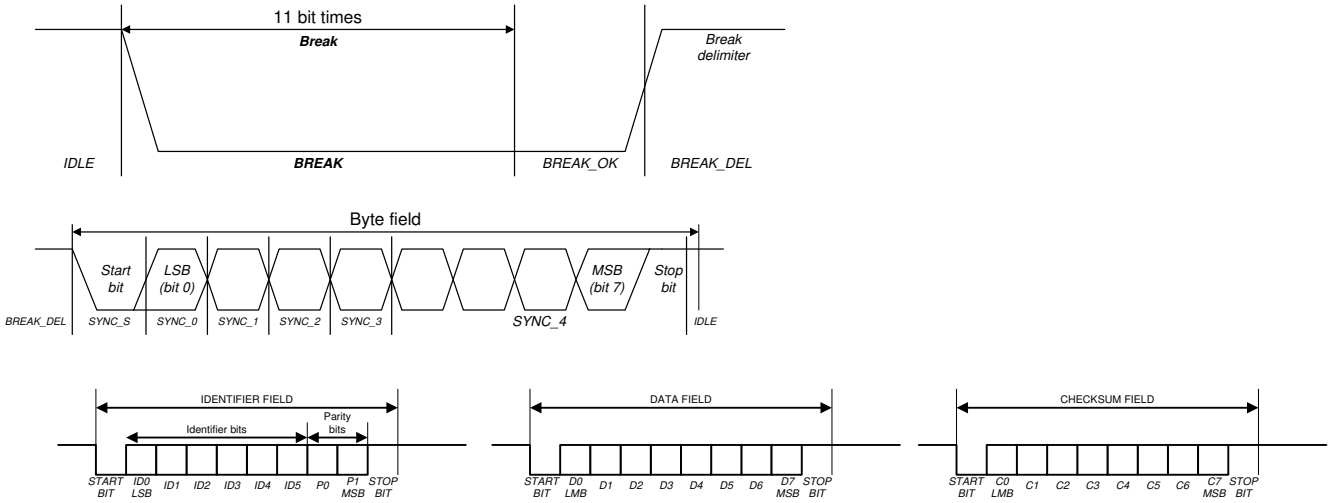
SBUF2 (0xA6) UART2 Data Buffer Register R/W 0x00

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

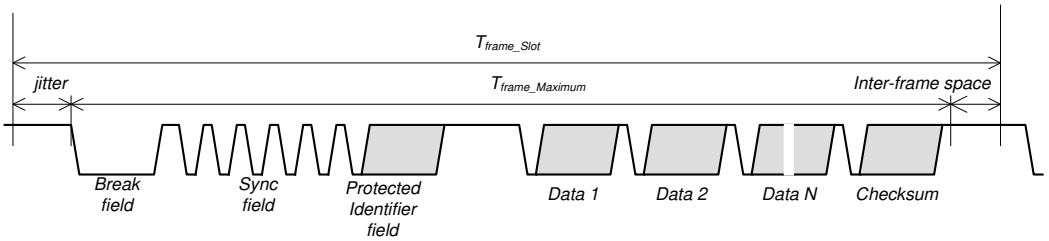
This register is the virtual data buffer register for both receive and transmit FIFO. When being read, it reads out the top byte of the RX FIFO; when being written, it writes into the top byte of the TX FIFO.

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, the SYNC byte, ID bytes, DATA bytes, and CRC bytes.



A LIN frame structure is shown 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.

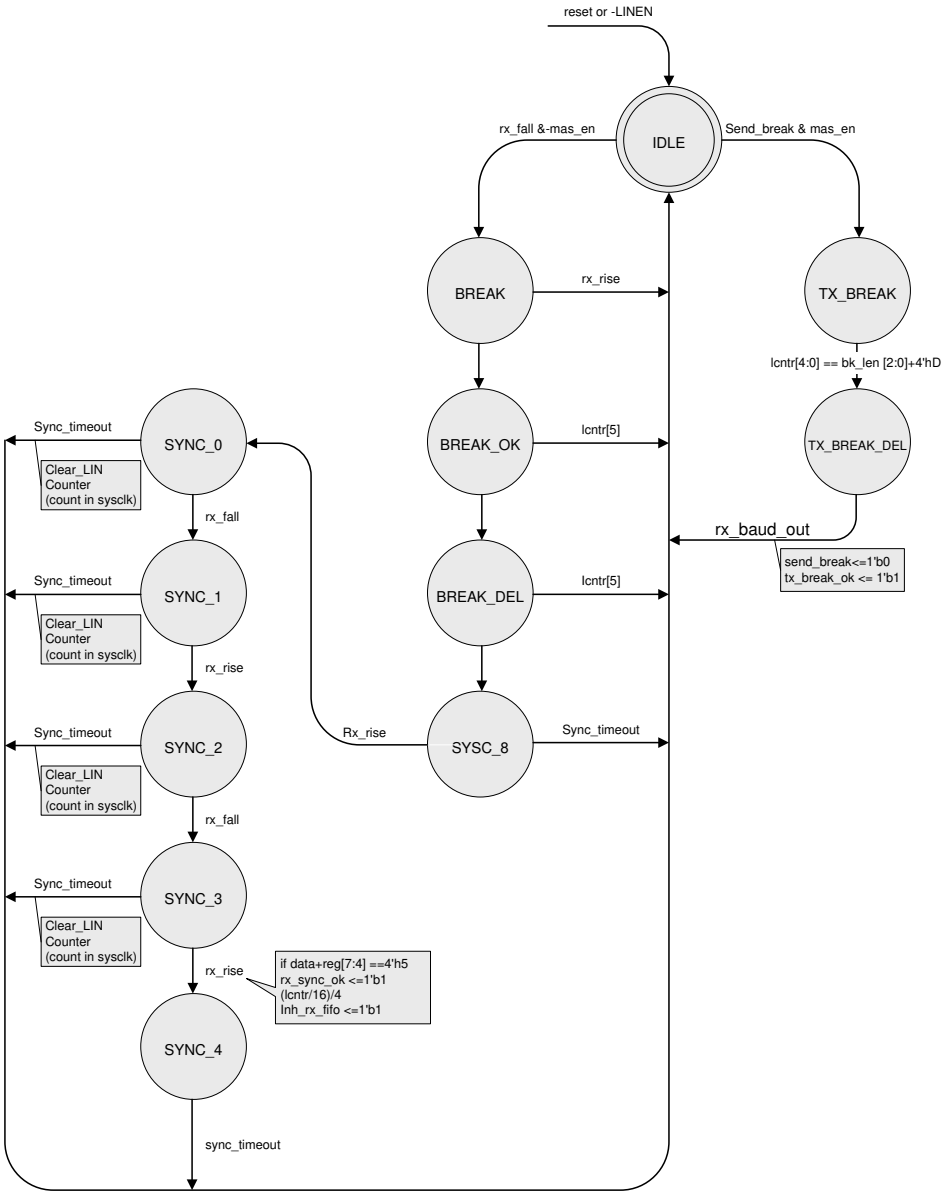
Initiate a SBK command. (SW needs to check if the bus is in idle state, and there is no pending transmit data).

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.



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

	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 received a valid SYNC field.
If ASU is 0, the LIN controller will only notice the synchronized baud rate in SBR[15-0] by issuing an RSI interrupt.
Please note, 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.

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.

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.

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

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

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

	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], an 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 R/W 0x00

	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 R/W 0x00

	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 received 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] can not be 0.
BUAD RATE = SYSCLK/BR[15-0].

When a slave receives a BREAK followed by a valid SYNC field, an 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 received a valid SYNC field.

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

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

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 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 completes. 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” in 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 R/W 0x00

	7	6	5	4	3	2	1	0
RD	LINTEN	EUART_OPL	-	-	ASUIE	SBKIE	RSIE	LCNTRIE
WR	LINTEN	EUART_OPL	-	-	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.
EUART_OPL	EUART/LIN Output logic polarity. EUART_OPL=0 for normal polarity. EUART_OPL=1 will reverse the output polarity of EUART/LIN.
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)

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.

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

	7	6	5	4	3	2	-	0
RD	SPIE	SPEN	MSTR	CPOL	CPHA	SCKE	-	-
WR	SPIE	SPEN	MSTR	CPOL	CPHA	SCKE	-	-

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.

In Slave mode, the sampling phase is determined by the combinations of CPOL and CPHA setting 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]	SPR[1]	SPR[0]	DIR
WR	ICNT1	ICNT0	FCLR	-	SPR[2]	SPR[1]	SPR[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/6;
001 – SCK = SYSCLK/8;
010 – SCK = SYSCLK/16;
011 – SCK = SYSCLK/32;
100 – SCK = SYSCLK/64;
101 – SCK = SYSCLK/128;
110 – SCK = SYSCLK/256;
111 – SCK = SYSCLK/512.

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	-	-	-	-

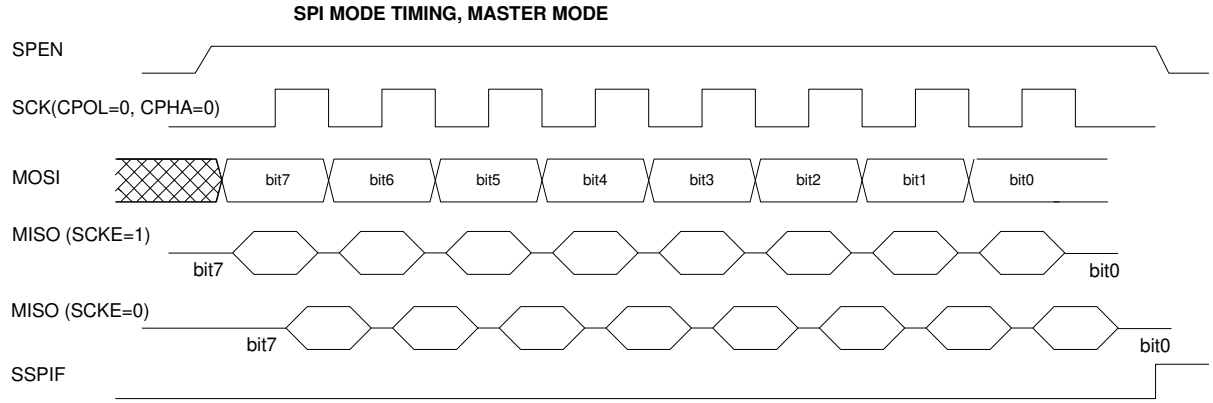
SSPIF	SPI Interrupt Flag bit. Set by hardware to indicate the completion of data transfer. Clear by assigning this bit to 0 or disabling SPI.
ROVR	Receive FIFO-overrun Error Flag bit. When Receiver FIFO Full Status occurs and SPI receives new data, ROVR is set and generates an interrupt. Clear by assigning 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)

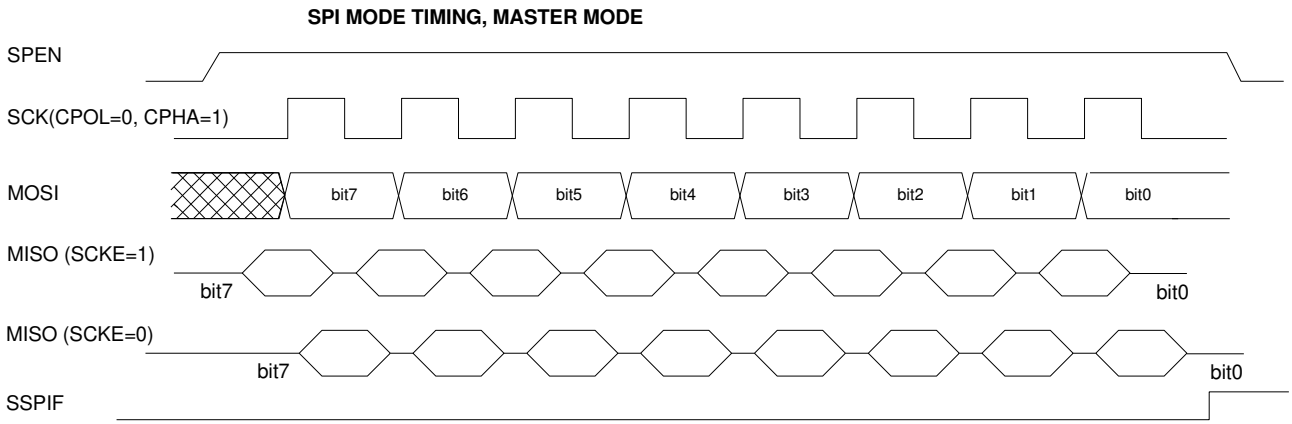
	7	6	5	4	3	2	1	0
RD	SPI Receive Data Register							
WR	SPI Transmit Data Register							

6.1 SPI Master Timing Illustration

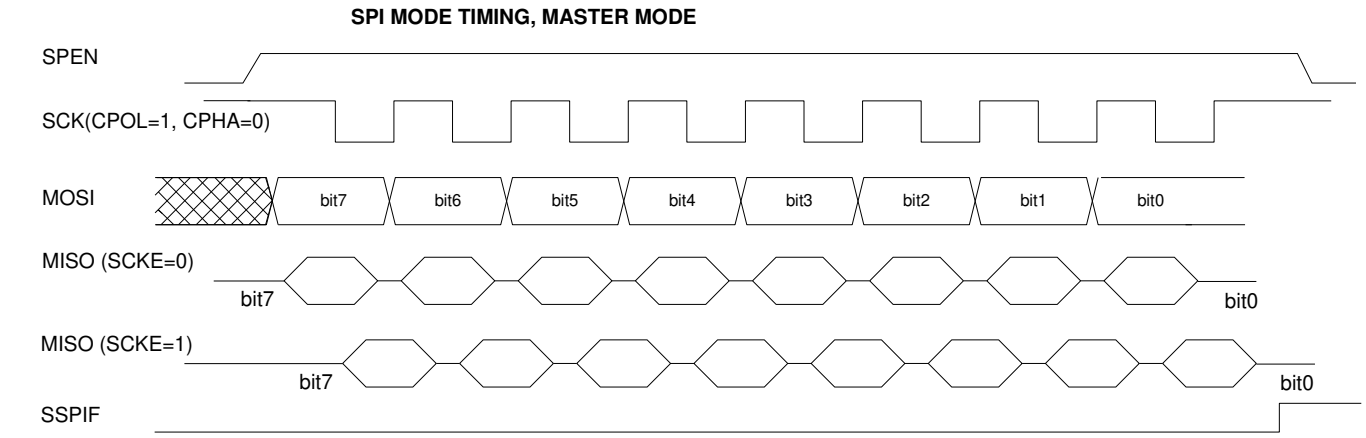
6.1.1 CPOL=0 CPHA=0



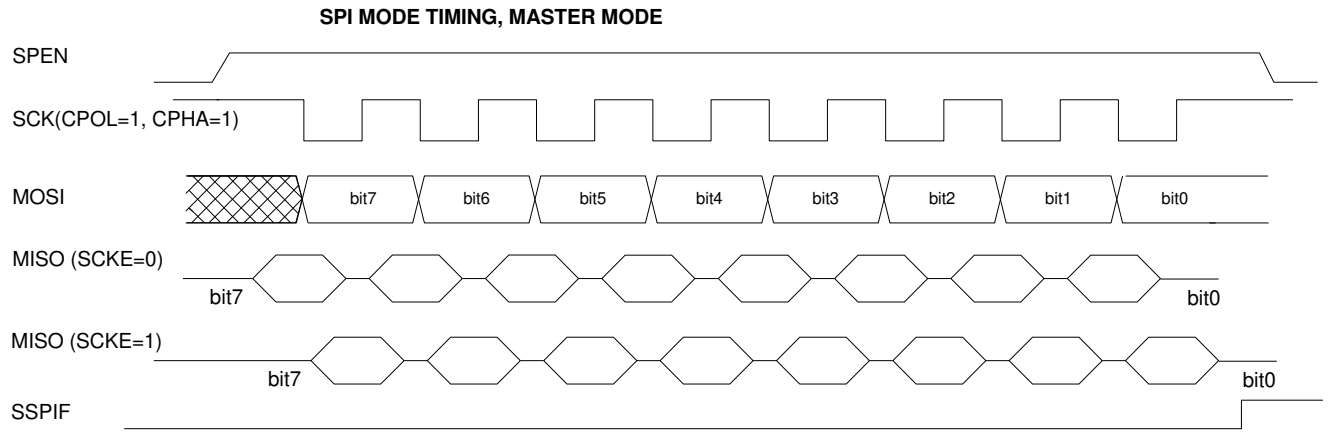
6.1.2 CPOL=0 CPHA=1



6.1.3 CPOL=1 CPHA=0

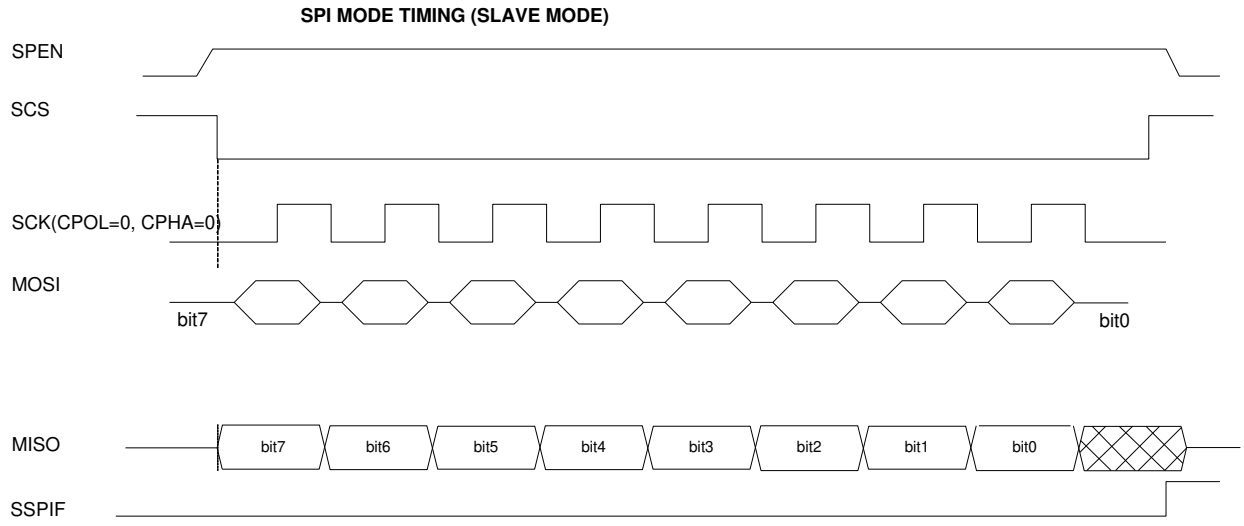


6.1.4 CPOL=1 CPHA=1

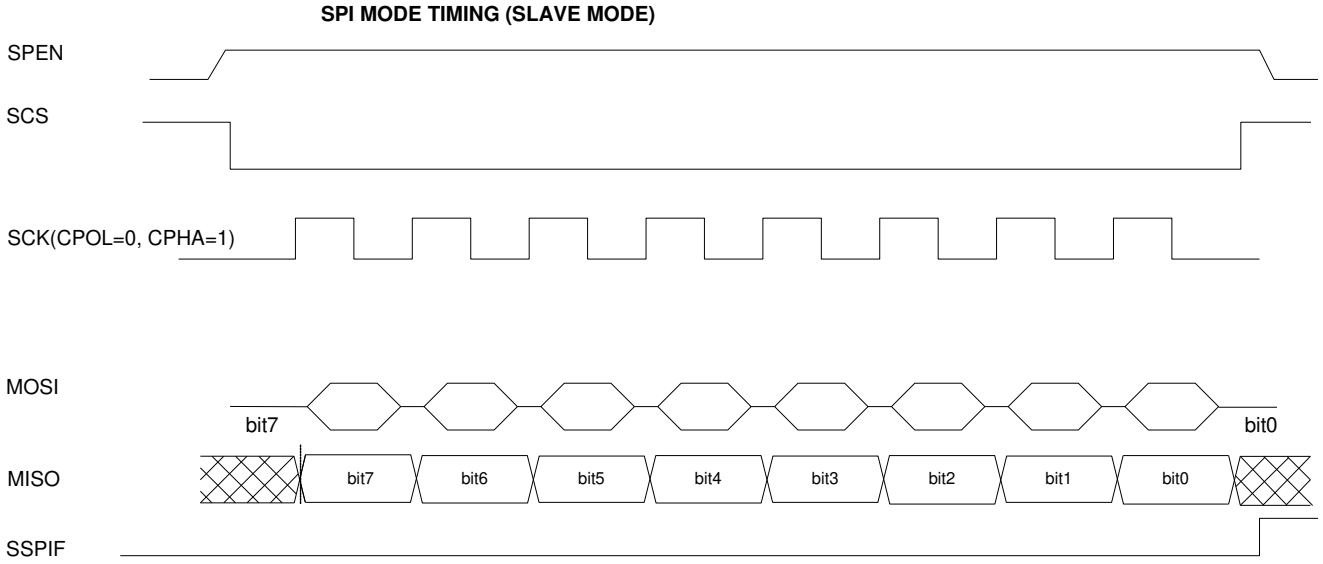


6.2 SPI Slave Timing Illustration

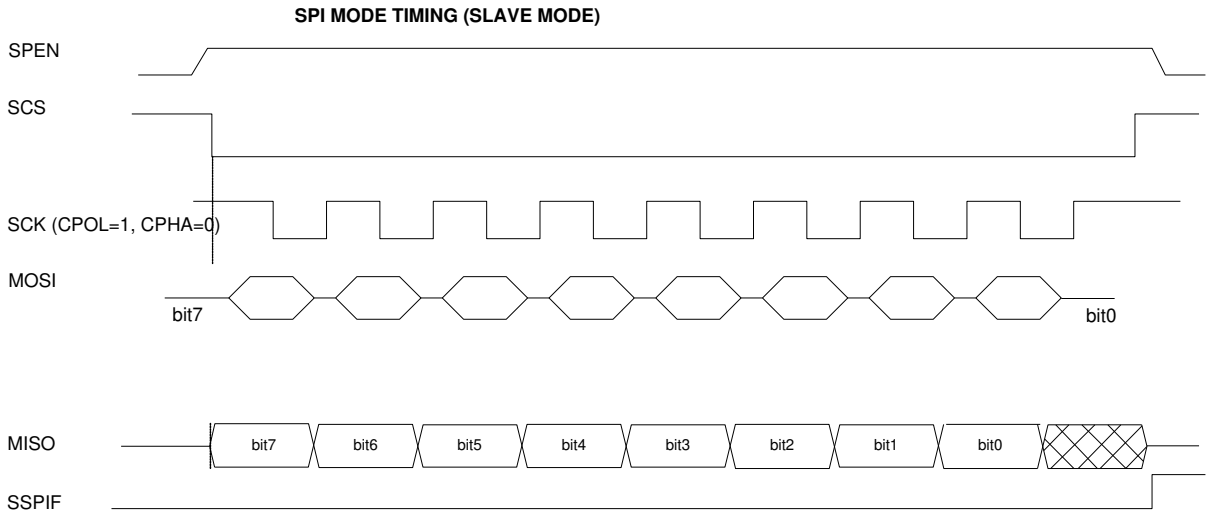
6.2.1 CPOL=0 CPHA=0



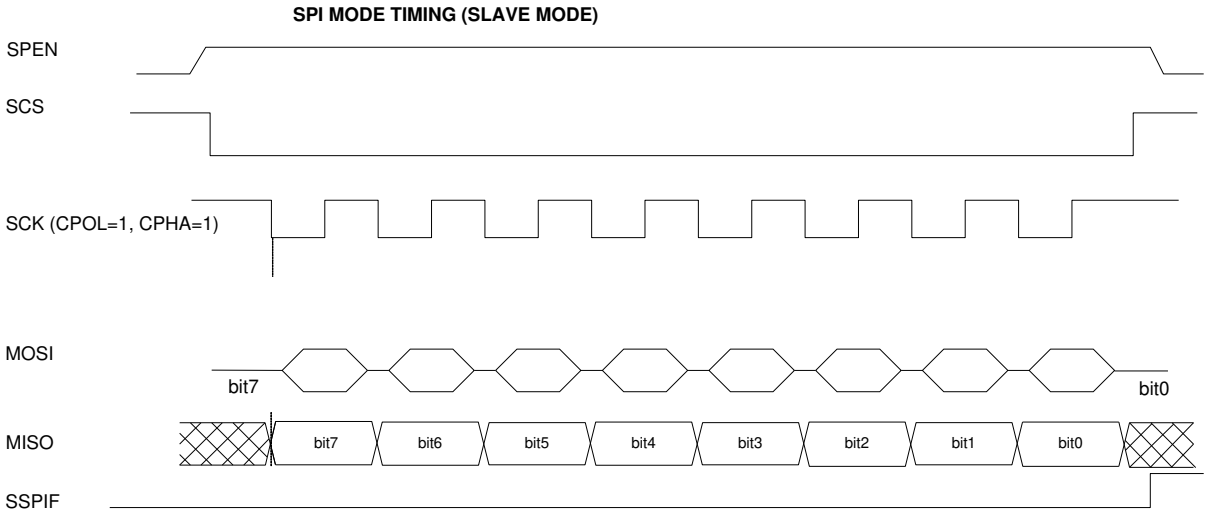
6.2.2 CPOL=0 CPHA=1



6.2.3 CPOL=1 CPHA=0

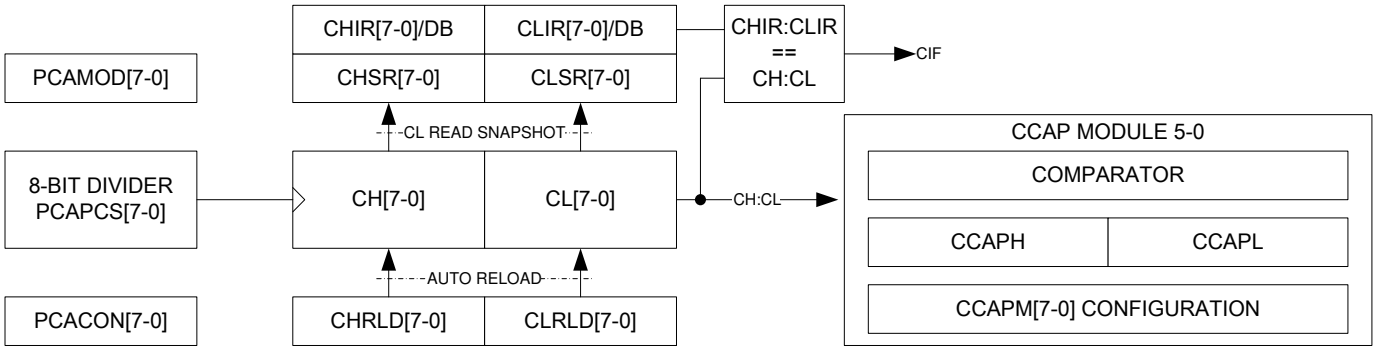


6.2.4 CPOL=1 CPHA=1



7. Programmable Counter Array (PCA) and Compare/Capture/PWM (CCP)

The PCA provides enhanced timing functions with less CPU intervention than the standard 8051 timers T0, T1, and T2. The PCA is partitioned in three parts. The main PCA Counter consists of CH and CL. There are 6 channels of Compare/Capture/PWM modules.



The MAIN COUNTER (CH and CL) is configured and controlled by two registers, CMOD and PCACON. The counter value is accessed by CH and CL registers. The counter can be configured as either FREE-RUN or AUTO-RELOADED mode. The counter values of CH and CL can be captured in CHSR and CLSR triggered by software or hardware. There is also a counter compare register CHIR and CHLR. An interrupt can be enabled at CH:CL == CHIR:CLIR. This allows the PCA to easily synchronize with the software control. CHIR and CLIR are double-buffered.

PCACPS (0xA0A5) PCA Counter Clock Scaling Register R/W (0x00)

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

PCACPS sets the clock input to the PCA at SYSCLK/(PCACS[7-0]+1).

PCAMOD (0xD1) PCA Mode Control Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	CIDL	RLDEN	COUNT8	OVF8EN	PCAEN	ECF	CIFEN	CMPTRIG
WR	CIDL	RLDEN	COUNT8	OVF8EN	PCAEN	ECF	CIFEN	CMPTRIG

CIDL — Counter Control bit in IDLE mode
When CIDL=1, PCA counting is disabled in IDLE mode.
When CIDL=0, normal counting of PCA in IDLE mode persists. PCAEN needs to be 1 for counting.

RLDEN: — AUTO-RELOAD Mode Enable bit
Set RLDEN=1to enable AUTO-RELOAD mode. At overflow, the main counter is reloaded with CHRL and CHRH in 16-bit mode or CHRL in 8-bit mode.
When RLDEN=0: FREE-RUN mode.

COUNT8 — 8-Bit or 16-Bit Counter Mode Select bit
When COUNT8=1 the PCA is configured as an 8-Bit counter. In 8-bit counter mode, OVF8EN must be set as 1.
COUNT8=0 the PCA is configured as a16-Bit counter

OVF8EN — 8-Bit Overflow Enable bit
When OVF8EN=1, the PCA overflow condition occurs at 0xXXFF to 0xXX00. In other words, the overflow condition (CF flag) is set every 256 count. This overflow condition also applies to the 16-bit counter mode.
When OVF8EN=0 the PCA overflow condition occurs at 0xFFFF to 0X0000. This does not work in 16-bit counter mode.

PCAEN — PCA counter Enable bit
Set PCAEN=1 to enable the PCA counter
Set PCAEN=0 to disable the PCA and also clears the counter value. When PCAEN=0, all double buffer is loaded with reload values.

ECF — Counter Overflow Interrupt Flag bit
When ECF=1 the overflow condition interrupt is enabled.

When ECS=0 the overflow interrupt is disabled.

CIFEN — Count Compare Interrupt Enable
Set IFEN=1 to enable CHIR:CLIR == CH:CL interrupt
Set IFEN=0 to disable this interrupt

CMPTRIG — Comparator Trigger Enable
CMPTRIG=1 enables the snapshot of PCA count value into CHSR by analog comparator interrupt.
CMPTRIG=0 disables the triggering.

PCACON (0xE1) PCA Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	CF	CIF	CCF5	CCF4	CCF3	CCF2	CCF1	CCF0
WR	CF	CIF	CCF5	CCF4	CCF3	CCF2	CCF1	CCF0

CF — Counter Overflow Flag bit
CF is set to 1 by hardware when overflow condition occurs. The overflow condition occurs at either of 0xFFFF to 0x0000 (OVF8EN=0) or 0xXXFF to 0xXX00 (OVF8EN=1). This bit must be cleared by software.

CIF — Count Compare Flag bit
CIF is set by hardware when CH:CL == CHIR:CLIR. This bit must be cleared by software.

CCF5 - CCF0 — Module Interrupt Flag 5-0
This is set by hardware as its corresponding module generates an interrupt. These bits must be cleared by software.

CH (0xE9) PCA Main Counter High Byte R/W (0x00)

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

CH holds the upper 8-bit of the main counter value.

CL (0xD9) PCA Main Counter Low Byte R/W (0x00)

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

CL holds the lower 8-bit of the main counter value. Reading CL triggers a snapshot action to copy CH:CL to CHSR:CLSR.

CHRLD (0xA0A7) PCA Counter CH Reload Value Register R/W (0x00)

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

This register holds the reload value for CH in AUTO-RELOAD mode.

CLRLD (0xA0A6) PCA Counter CL Reload Value Register R/W (0x00)

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

This register holds the reload value for CL in AUTO-RELOAD mode.

CHSR (0xF3) PCA Snapshot Register of CH RW (0x00)

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

CHSR[7-0] — CH Snapshot Register. It is read-only.

CHIR[7-0] CH Counter Compare Interrupt . The compare value is double- buffered.

CLSR (0xF2) PCA Snapshot Register of CL RO (0x00)

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

CLSR[7-0] CL Snapshot Register. It is read-only.
CLIR[7-0] CL Counter Compare Interrupt The compare value is double-buffered.

The Compare/Capture modules receive the 16-bit count value from the main counter as the time base. Each module is configured by its mode register CCAPMn and contains two 8-bit registers used for comparing value holder or capturing value in storage. There are several basic modes of operation for CCP modules and each CCP module can be configured in the same or different modes.

CCAPMn CCP Module Configuration Register (0xB2, 0xB3, 0xB4, 0xB5, 0xB6, 0xB7) R/W (0x00)

	7	6	5	4	3	2	1	0
RD	OF	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF
WR	OF	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF

OF, TOG When the module is configured as Timer/Comparator modes, these two bits determine the output flag status on pin CEX when timer is up or comparison matches. The setting only affects the CEX and does not impact the interrupt generation

OF	TOG	CEX
0	0	CEX is unchanged.
0	1	CEX toggles.
1	0	CEX change to low (or remains low).
1	1	CEX change to high (or remains high).

When the module is configured as PWM mode, OF is ignored. Set TOG=1 to enable CEX output to high regardless of PWM value.

ECOM Comparator Enable bit. Set to enable comparator function. Clear to disable the comparator.
CAPP Positive Edge Capture bit. Set to use a positive edge as the capture edge. Clear to disable positive edge capture.
CAPN Negative Edge Capture bit. Set to use a negative edge as the capture edge. Clear to disable negative edge capture
MAT Match Control bit. When MAT = 1, a match of CH/CL with CCAPH/CCAPL causes CCF to be set and generates an interrupt. It also enables a compare edge interrupt in WPWM mode.
PWM Pulse Width Modulation bit. Set to enable PWM function. CEX is the PWM output.
ECCF Enable Capture/Compare/PWM Interrupt bit. Set to enable the CCP module n (n = number of the designated module; there are 6 modules in this case) to generate the interrupt.

CCAPnL CCP Compare Value Low Register (0xD2, 0xD4, 0xD6, 0xE2, 0xE4, 0xE6) R/W (0x00)

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

CCAPnL register holds the compare value or capture value. It is used as PWM value register.

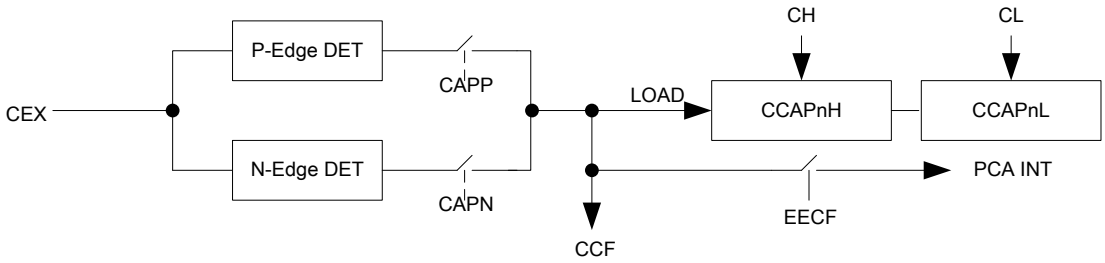
CCAPnH CCP Compare Value High Register (0xD3, 0xD5, 0xD7, 0xE3, 0xE5, 0xE7) R/W (0x00)

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

CCAPnH register holds the compare value or capture value. It is used as PWM value register.

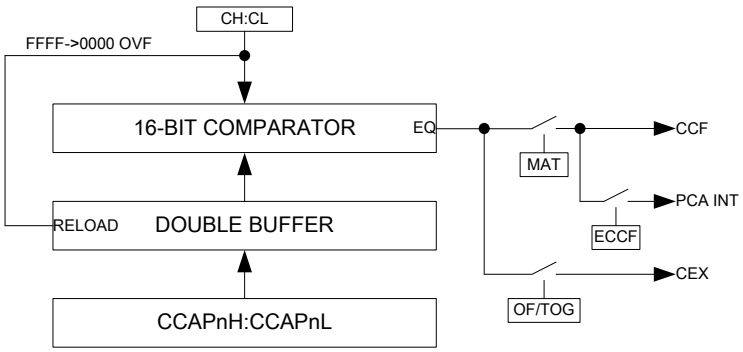
7.1 16-Bit Capture Mode

The capture mode is used to measure the elapse time of an external event between the edges of the enabled external signal when either CAPP or CAPN is set. The external CEX is sampled for transition. When a valid capture edge occurs in CEX, the current CH/CL count value is loaded into CCAPnH and CCAPnL registers. At the same time CCFn in PCACON register is set, and interrupt is generated if enabled. The block diagram of the configuration is shown below:



7.2 16-Bit COMPARE TIMER MODE

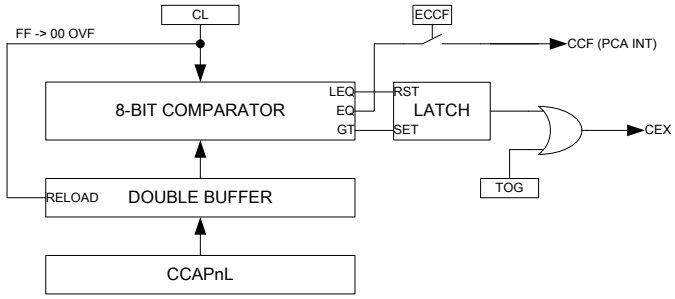
The COMPARE TIMER mode can be used as a software timer or to generate a PWM output. This mode is enabled when ECOM is set and CAPP CAPN are set to low. To allow the compare result to be used, MAT/ECCF also needs to be set. The CCAPnH and CCAPnL hold the 16-bit Timer value and are compared against the incrementing value CH and CL from the main counter. The compare value is double-buffered and is updated when the main counter overflows. This prevents any unexpected comparator output during updating a new value to CCAPnH and CCAPnL. When a match occurs, CCF is set and an interrupt is generated. The block diagram of this mode is shown as following.



The match result can also be used to generate CEX output change. Depending on the CCAPM's OF and TOG setting, CEX output is changed at the compare-match instant. However, the triggering of the change of CEX does not require MAT qualifier. Using CEX, waveform of precision duty cycle waveform or frequency modulation can be generated. The effect of OF/TOG on CEX is described in CCAPM register. To avoid unwanted glitches or a match condition when updating the CCAPnH and CCAPnL registers, when ECOM is set and the writing to CCAPnL causes ECOM to clear. Writing to CCAPnH sets ECOM to start the comparator. Therefore user program should update CCAPnL first and then CCAPnH. Of course, ECOM bit can be controlled directly through CCAPMn register.

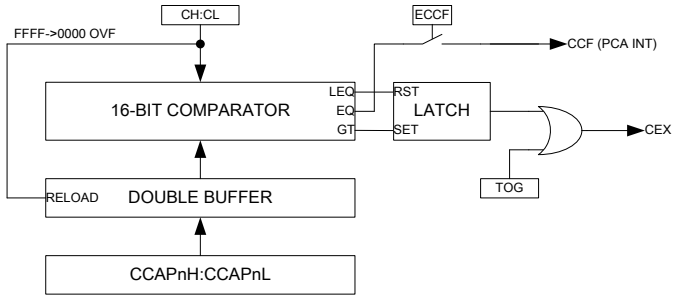
7.3 8-Bit Pulse Width Modulator Mode

This mode is used to generate 8-bit precision PWM output on CEX. The time base of the PWM is provided by CL of the main counter. CCAPnL is used for compare value. When CL $\leq$ CCAPnL, the output is 0 and when CL > CCAPnL, the output is 1. The compare value is double-buffered and is updated when CL overflows from FF to 00. The PWM mode is enabled when ECOM and PWM bits are both set, and CAPP, CAPN are both low. Note that under the above compare method, the maximum CEX duty cycle is 255/256. If TOG is set to 1 in this mode, CEX is forced high to provide 256/256 with full high duty cycle. If ECCF bit is set, then when CCAPnL=CL (i.e. the output change), CCF is also set to 1 by hardware and triggers a PCA interrupt. The following block diagram shows the PWM mode operation.



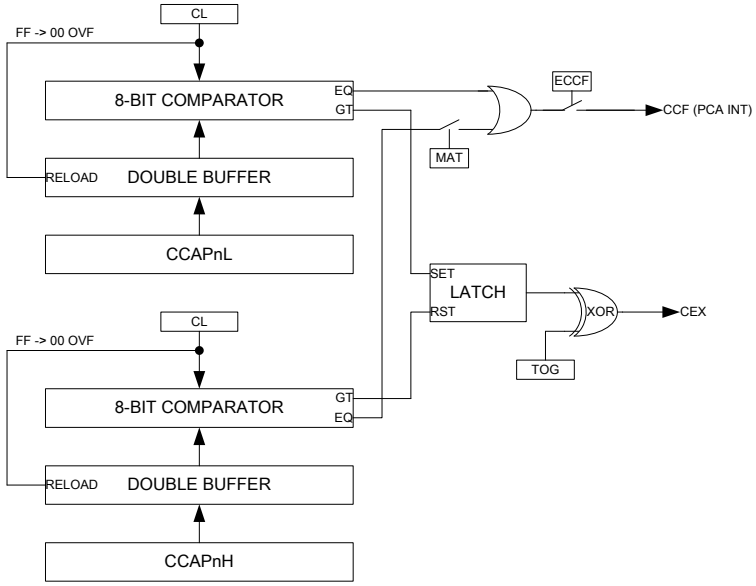
7.4 16-Bit Pulse Width Modulator Mode

This mode is similar to the 8-bit PWM mode except it uses the 16-bit CH:CL count value for the time base of the PWM. The compare value is composed of CCAPnH:CCAPnL and is double-buffered. When CH:CL ≦ CCAPnHL:CCAPnL, the output is 0; when CH:CL > CCAPnHL:CCAPnL, the output is 1. The output can be forced to 1 by setting TOG=1. The PWM mode is enabled when both ECOM and PWM bits and CAPP are set while CAPN is low. An interrupt is enabled by ECCF and triggered when CH:CL==CCAPnH:CCAnL.



7.5 8-BIT Windowed Pulse Width Modulator (WPWM) Mode

This mode is used to generate 8-bit PWM output on CEX. The difference from regular PWM mode is that the CEX becomes high during a window of CL count. CEX becomes high when CL is greater than CCAPnL, CEX is reset to low when CL is greater than CCAPnH. The compare values are double-buffered. Therefore the value in CCAPnH must be larger than CCAPnL to prevent abnormal operations. The output of CEX can be inverted by setting TOG to 1. An interrupt can be enabled by ECCF, if MAT=0, then CL=CCAPnL generates an interrupt. Setting MAT to 1 and CL=CCAPnH also generates an interrupt.



7.6 CCP Function Summary

CCP Function			ECOM	CAPP	CAPN	PWM	OF	TOG
NO OPERATION (Note 1)			0	0	0	0	X	X
16-bit Capture	Triggered by positive edge of CEX		X	0	1	0	X	X
	Triggered by negative edge of CEX			1	0			
	Triggered by both edges of CEX			1	1			
16-bit Compare	CH:CL == CCAPnH:CCAPnL	CEX is unchanged	1	0	0	0	0	0
		CEX toggles					0	1
		CEX = 0 (or stay 0)					1	0
		CEX = 1 (or stay 1)					1	1
8-bit PWM	CEX = 0 when CL <= CCAPnL CEX = 1 when CL > CCAPnL		1	0	0	1	X	0
	CEX = 1							1
16-bit PWM	CEX=0 if CH:CL =< CCAPnH:CCAPnL CEX=1 if CH:CL > CCAPnH:CCAPnL		1	1	0	1	X	0
	CEX=1							1
8-bit WPWM	CEX = 0 when CL > CCAPnH CEX = 1 when CL > CCAPnL		1	1	1	1	X	0
	CEX = 1 when CL > CCAPnH CEX = 0 when CL > CCAPnL							1

ECOM cannot be set to 1 by hardware (when writing to CCAPnH) if all bits (OF, ECOM, CAPP, CAPN, MAT, TOG, PWM) in CCAPM are set to 0 (NO OPERATION mode).

In 16-bit compare mode, ECOM can be set to 1 by hardware (when writing to CCAnPH) or software, and can be cleared to 0 by hardware (when writing to CCAnPL) or software. When ECOM is cleared to 0 in this mode, the CCP function enters NO OPERATION mode. The compare value is CCAPnH:CCAPnL and is double-buffered.

8. PWM0/PWM1 Controller

PWM0/1 controller provides an 8-bit programmable duty cycle output. The counting clock of PWM0/1 is programmable and the base frequency of the PWM0/1 is just the counting clock divided by 256. The duty cycle setting is always double buffered. PWM0 output is connected through P04, P06, P12, P17 or P23. PWM1 output is connected through P05, P13, P15, P16 or P22.

PWM0CFG (0xA088) PWM0 Clock Scaling Setting Register R/W (0x00)

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

CS[7-5] Clock Pre-Divider Setting
- CS[7-5] = 000 Disable PWM0 and Output = 0
- CS[7-5] = 001 /4
- CS[7-5] = 010 /8
- CS[7-5] = 011 /16
- CS[7-5] = 100 /32
- CS[7-5] = 101 /64
- CS[7-5] = 110 /128
- CS[7-5] = 111 Disable PWM0 and Output = 1

CS[4-0] Clock Divider Setting. Range from 0 to 31.

The counting clock is SYSCLK/CS[7-5]/(CS[4-0]+1) and one cycle has 256 counting clocks. Assuming SYSCLK is 16MHz, and we want the PWM base frequency of 120Hz. First we get 16MHz/120Hz/256 = 520.8. Where 520.8 needs to be separated as two multiplicands of CS[7-5] and CS[4-0]. Then by trial and error we can select CS[7-5] = 100, and CS[4-0] = 0x0F. This gives 16MHz/256/32/(15+1) = 122Hz.

PWM0DTY (0xA089) PWM0 Duty Register R/W (0x00)

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

PWM0DTY registers define the PWM0 duty cycle. The maximum duty cycle is 255/256. PWM0DTY is always double buffered and is loaded to duty cycle comparator when the current counting cycle is completed.

PWM1CFG (0xA08A) PWM1 Clock Scaling Setting Register R/W (0x00)

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

CS[7-5] Clock Pre-Divider Setting
- CS[7-5] = 000 Disable PWM1 and Output = 0
- CS[7-5] = 001 /4
- CS[7-5] = 010 /8
- CS[7-5] = 011 /16
- CS[7-5] = 100 /32
- CS[7-5] = 101 /64
- CS[7-5] = 110 /128
- CS[7-5] = 111 Disable PWM1 and Output = 1

CS[4-0] Clock Divider Setting. Range from 0 to 31

PWM1DTY (0xA089) PWM1 Duty Register R/W (0x00)

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

PWM1DTY registers define the PWM1 duty cycle. The maximum duty cycle is 255/256. PWM1DTY is always double buffered and is loaded to duty cycle comparator when the current counting cycle is completed.

PWMFG (0xA08C) PWM0/1 Interrupt Enable and Flag Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PWM0FG	PWM1FG			PWM0INTE	PWM1INTE		
WR	PWM0FG	PWM1FG			PWM0INTE	PWM1INTE		

PWM0FG PWM0 Interrupt Flag
PWM0FG is set by hardware if PWM0INTE=1 and at the time the PWM counter reaches 256.

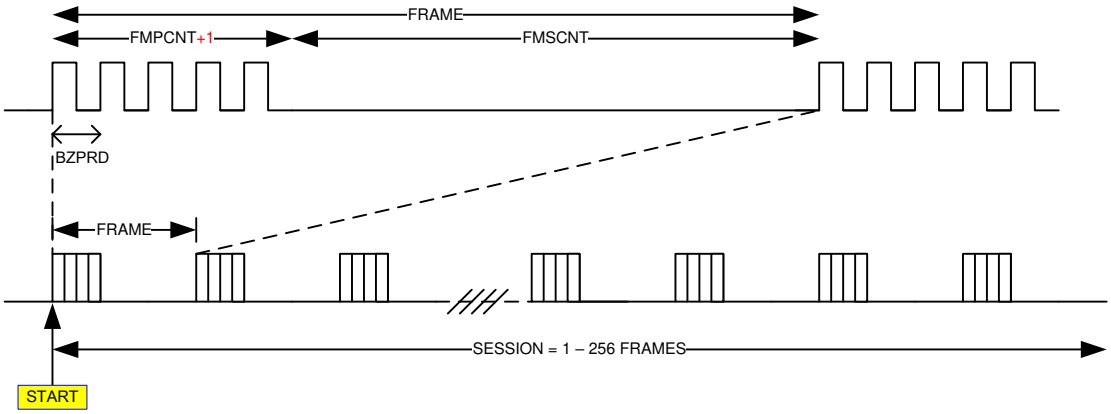
PWM1FG PWM1 Interrupt Flag
PWM1FG is set by hardware if PWM1INTE=1 and at the time the PWM counter reaches 256.

PWM0INTE PWM0 Interrupt Enable
PWM0INTE=0 disable the PWM0 counter interrupt.
PWM0INTE=1 enable the PWM0 counter interrupt.

PWM1INTE PWM1 Interrupt Enable
PWM1INTE=0 disable the PWM1 counter interrupt.
PWM1INTE=1 enable the PWM1 counter interrupt.

9. Buzzer controller

This buzzer controller is a simple sound generator. The waveform generated is shown in the following. BZPRD is determined from the buzzer frequency BZFRQ. The basic element of the waveform is a FRAME where each FRAME is a combination of a number of pulses defined by FMPCNT and a number of pulse duration of silence defined by FMSCNT. This combination allows simple two-tone generations, one defined by BZFRQ, and one sub-harmonic defined by BZFRQ/(FMPCNT+FMSCNT). The SESSION defines the number of FRAMES, which essentially determines the duration of the sound. The output can be configured as software start, which triggers a session output when START is set, or as continuous if CONT is set. In either configuration, session end interrupt can be enabled to inform software for intervention.



BZFRQ (0xA128) Buzzer frequency controlled R/W (0x10)

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

BZFRQ[7-0] Buzzer Frequency Setting

The buzzer frequency is defined by BZCLK/BZFRQ[8-0], BZFRQ[8] is located at BZCFG[4].

BZFMCFG (0xA129) Buzzer Frame Configuration Register R/W (0xF0)

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

FMPCNT[3-0] Pulse Count for Frame On

This number of pulse during Frame on duration is FMPCNT[3-0]

FMSCNT[3-0] Pulse Count for Frame Silence

This number of pulse during Frame silence duration is FMPCNT[3-0]

FMPCNT[3-0] + FMSCNT[3-0] should be greater than 1 for normal operations.

BZSNFMCNT (0xA12A) Buzzer Session Frame Count Register R/W (0x80)

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

SNFMCNT[7-0] Frame Repeat Number of a Session

SNFMCNT[7-0] defines the number of frame in a session.

BZCFG (0xA12B) Buzzer Configure Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	BZEN	SNENDE	BZCSEL	BZFRQ[8]	BZPOL	CONT	SSINTF	BUZY
WR	BZEN	STENDE	BZCSEL	BZFRQ[8]	BZPOL	CONT	SSINTF	SSTART

BZEN Buzzer Control Enable

BZEN=1 enables the buzzer controller

BZEN=0 disables the buzzer controller

SSENDE	Session End Interrupt Enable STENDE=1 enables the session end interrupt. The interrupt is triggered for both continuous and session start modes. STENDE=0 disables the session end interrupt.
BZCSEL	Buzzer Clock Select 0 = IOSC 1 = SIOSC
BZPOL	BZOUT Polarity Setting BZPOL=1, BZOUT is inverted BZPOL=0, normal polarity
CONT	Continuous Mode CONT=1 enables the buzzer output continuously according session format. CONT=0 disables the buzzer continuous mode and output one session by issuing a START.
SSTRAT	Session Start Command START=1 initiate a session output on the buzzer START is self-cleared when the session is completed
SSINTF	Session End Interrupt Flag SNINTF is set by hardware at the end of session if SSENDE=1. This also triggers an interrupt. SSINTF must be cleared by software.
BUSY	Busy Status BUSY is set to 1 by hardware when the output is active. BUSY=0 indicates the output is idle.

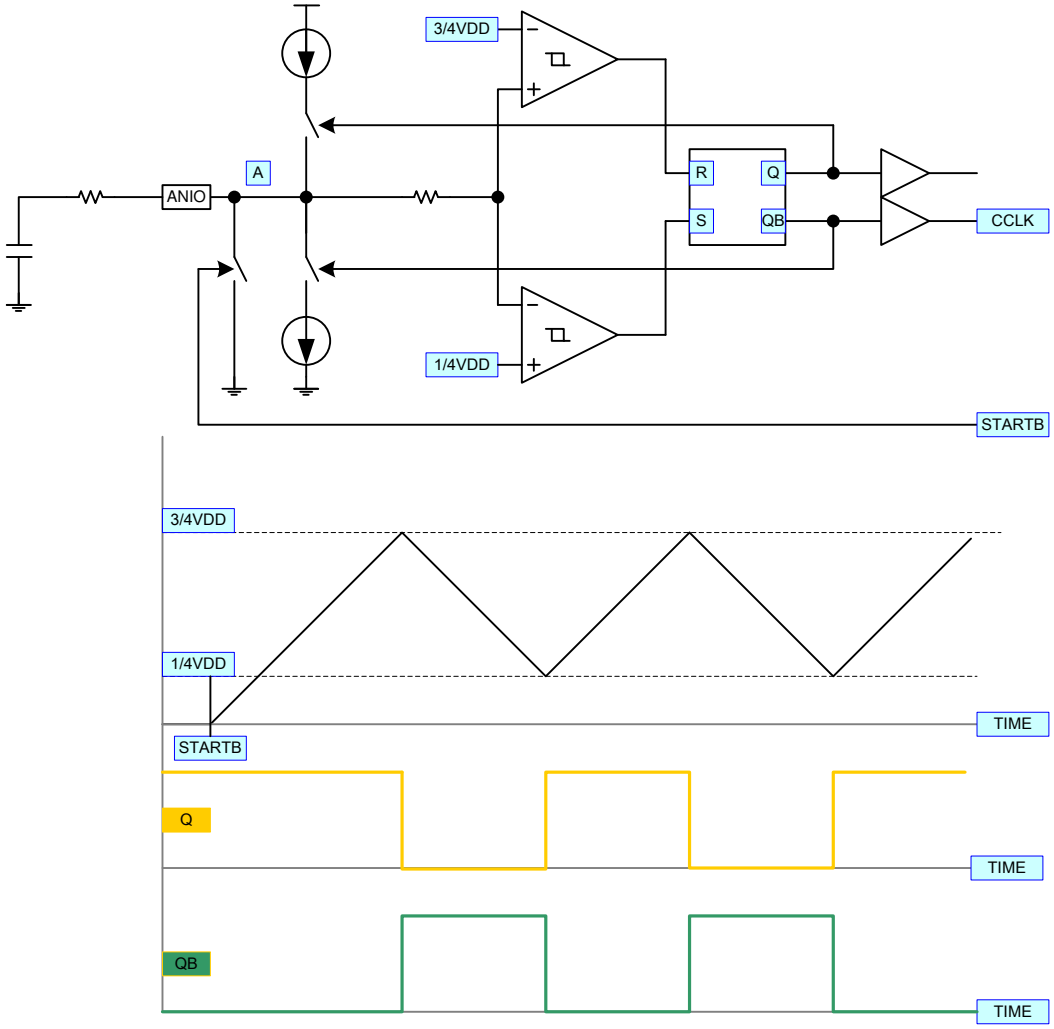
BZSNMUTE (0xA12C) Buzzer Session MUTE Frame Register R/W (0xff)

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

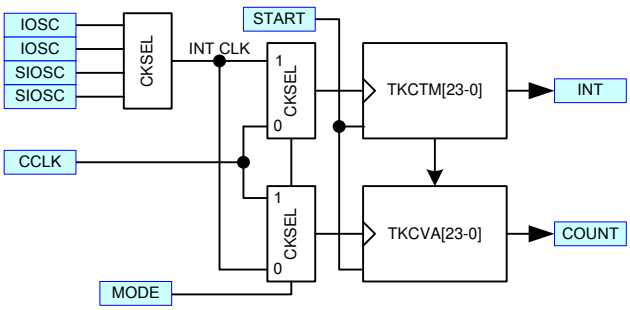
BZSNMUTE [7-0]	start mute Frame number of a Session Mute buzzer when frame count greater than BZSNMUTE in a session.

10. Touch Key Controller

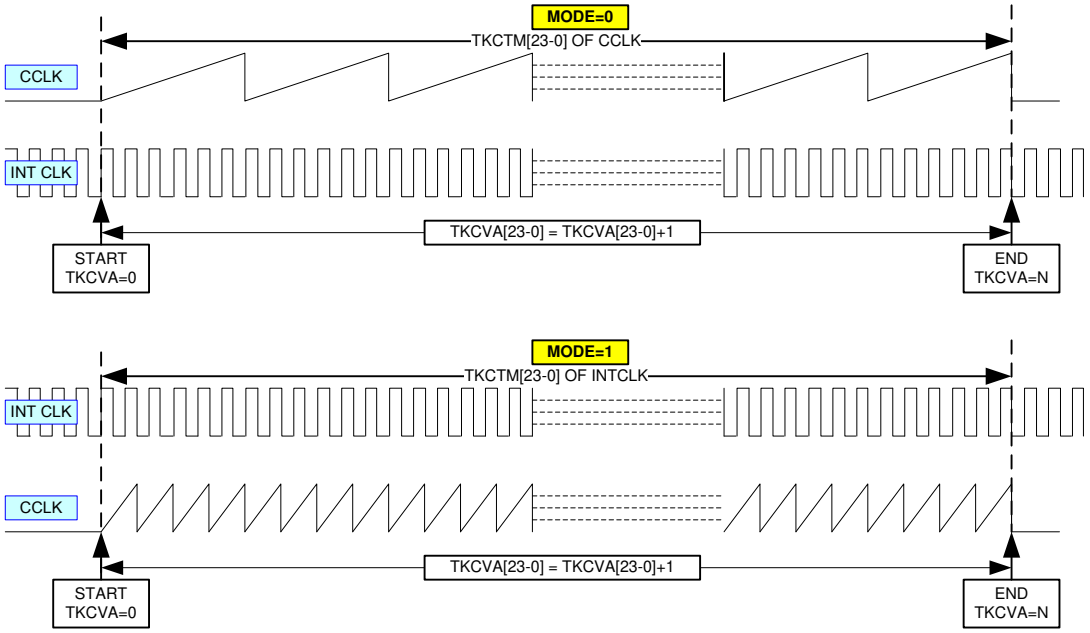
The Touch Key Controller utilizes capacitance based touch key sensing. The touch is detected by detecting the change of an external sensing capacitance. When a key is touched, the capacitance of the key goes up. The capacitance detection is based on a CMOS relaxation oscillator as shown, which uses the key capacitance as the oscillation parameters. The Touch Key Controller in operation typically consume at 25uA (when slow comparator is selected) and at 250uA (when fast comparator is selected). The inputs of the Touch Key Controller are multiplexed with multi-function GPIO's ANIO pins. To use these ports as Touch Key inputs, the ANEN must be enabled and other drivers to be in high-impedance state.



The touch detection algorithm is done in software and is assisted with the following hardware. The hardware consists of one 24-bit Timer and a 24-bit Counter. The clock into the timer and the counter can be interchanged by MODE selection.

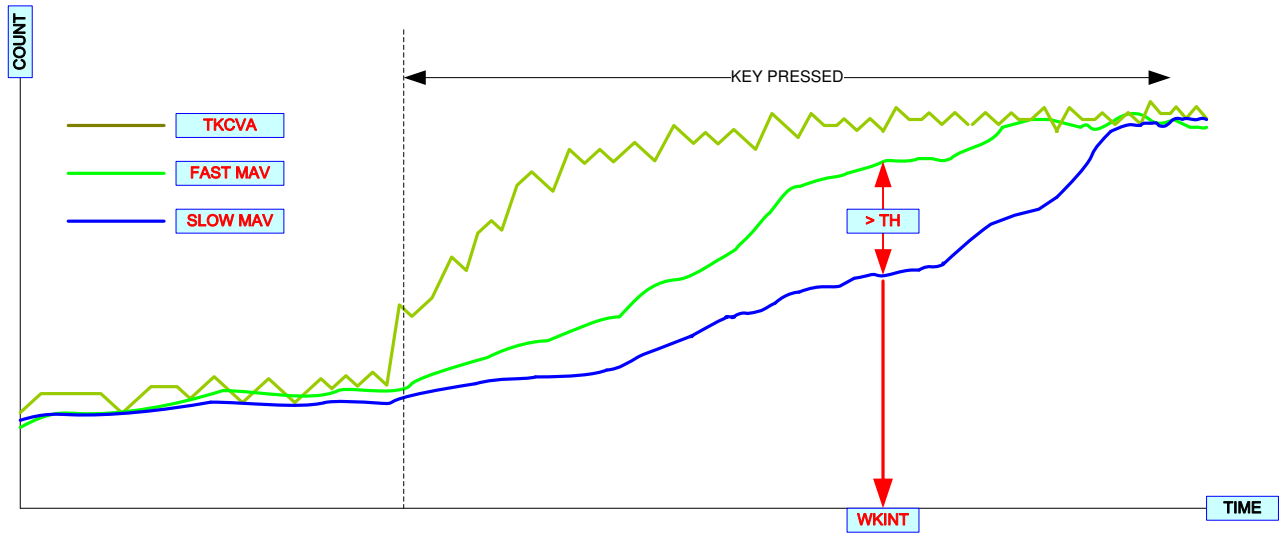


When MODE=0 (MODE0), the counter uses internal clock and timer uses sensing clock. This typically is used when the capacitance oscillation is slower. When MODE=1 (MODE1), the clock relationship is reversed and this is typically used when sensing oscillation is faster. Typical waveforms are shown in the following timing diagram.



The START command clears the counter and load the timer, and then starts the timer and the counter. When the timer (TKCTM) expires, the counter value (TKCVA) is captured and an interrupt is triggered. Two hardware moving average filters, one with slow time constant (TKMAS) and one with fast time constant (TKMAF), are also included in the Touch Key Controller. The software should write the previous moving average into TKMAF and TKMAS before issuing START. When a capture is completed, the moving average filters are also updated. The software can read out TKCVA as well as TKMAF and TKMAS for algorithm use.

The Touch Key Controller also allows auto detection. The auto detection is primarily used fore wake up purpose. The TKCTM and TKCVA functions the same as the normal mode, and the capture cycles continues without interrupt. However, in AUTO mode, TKCTM and TKCVA, as well as TKMAF and TKMAS are limited in 16-bit width. The hardware monitors the TKMAF and TKMAS, and when the fast moving-average is greater than the slow moving-average by a predefined threshold, a wake-up interrupt is triggered. The following diagram illustrates the progressive count curves.



TKCCFG (0xA130h) Touch Key Controller Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	TKCEN	MODE	ICKSEL[1-0]		AUTO	TKINTEN	TKINTF	BUSY
WR	TKCEN	MODE	ICKSEL[1-0]		AUTO	TKINTEN	TKINTF	START

TKCEN Touch Key Controller Enable

MODE Timer/Counter Mode

MODE=0 selects Capacitor Clock as timer and internal clock as counter

MODE=1 selects internal clock as timer and capacitor clock as counter

ICKSEL[1-0]	Internal Clock Select. This also defines the main clock of the TKC 00 = IOSC 01 = IOSC 10 = SIOSC 11 = SIOSC
AUTO	Auto Detection and Wake Up
TKINTEN	Touch Key Interrupt Enable
TKINTF	Touch Key Interrupt Flag
BUSY	Touch Key Timer Active BUSY is set by hardware when the capture is in progress.
START	Start Capacitance Sense Capture Command Write START=1 to start the capture If AUTO=0, one capture is done. If AUTO=1, continuous capture is initiated until the detection is made Write START=0 to abort on-going capture

TKCAFG (0xA131h) Touch Key Controller Analog Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	HSPEED	PSREN	-	IMULT	ISEL[3-0]			
WR	HSPEED	PSREN	-	IMULT	ISEL[3-0]			

HSPEED	High Speed Enable HSPEED=0 set comparator at low speed and also have current scale equals to 1. HSPEED=1 set comparator at high speed and also have current scale equals to 2.
PSREN	Pseudo-Random Sequence Enable PSREN=0 for normal mode operation PSREN=1 uses a pseudo-random 1/0 bit stream of 255 bit length to control the current as 2I or I for each charge/discharge cycle. This random effect helps to counteract the impact of EMI.
IMULT	Current Scale Setting IMULT=0 uses normal scale. 0.125uA – 2.0uA (HSPEED=0), and 0.25uA – 4.0uA (HSPEED=1). IMULT=1 makes the current 5X. 0.625uA – 10uA (HSPEED=0) and 1.25uA – 20uA (HSPEED=1)
ISEL[3-0]	Current Setting The pull-up and pull-down current is set as (ISEL[3-0]) in unit step of 0.125uA. The range is from 0.125uA to 2uA.

TKCFFG (0xA132h) Touch Key Controller Filter Configuration Register R/W (0x00)

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

SCFG[3-0] Slow Moving Average Configuration

SCFG[3:0]	TKCMVS
0	TKCVAL
1	(TKCVAL + 15 * TKCMVS) / 16
2	(TKCVAL + 31 * TKCMVS) / 32
3	(TKCVAL + 63 * TKCMVS) / 64
4	(TKCVAL + 127 * TKCMVS) / 128
5-15	Reserved

FCFG[3-0] Fast Moving Average Configuration

FCFG[3:0]	TKCMVF
0	CAPVAL
1	(CAPMVF *1 + CAPVAL*1)/2
2	(CAPMVF *3 + CAPVAL*1)/4
3	(CAPMVF *7+ CAPVAL*1)/8
4	(CAPMVF *15+ CAPVAL*1)/16

5-15	Reserved

TKCATH0 (0xA133h) Touch Key Auto Detect Threshold 0 R/W (0x00)

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

TKCATH1 (0xA134h) Touch Key Auto Detect Threshold 1 R/W (0x00)

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

TKCATH2 (0xA135h) Touch Key Auto Detect Threshold 2 R/W (0x00)

	7	6	5	4	3	2	1	0
RD	ABSTH	TKCDLY[1-0]		TKCATH[21-16]				
WR	ABSTH	TKCDLY[1-0]		TKCATH[21-16]				

ABSTH Absolute Threshold
ABSTH = 1, use TKCATH[21-0] as an absolute value for threshold.
ABSTH = 0, use TKCATH[9-0] as a ratio threshold. The threshold value is CAPMVS * TKCATH[9-0]/1024.

TKCDLY[1-0] Auto Detection Delay. TKCDLY[1-0] defines the start delay for auto-detection for stabling the moving average to prevent false trigger.
00 = 64 Capture
01 = 256 Capture
10 = 512 Capture
11 = 1024 Capture

TKCTMR0 (0xA140h) Touch Key Timer Register 0 R/W (0x00)

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

TKCTMR1 (0xA141h) Touch Key Timer Register 1 R/W (0x00)

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

TKCTMR2 (0xA142h) Touch Key Timer Register 2 R/W (0x00)

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

TKCVAL0 (0xA144h) Touch Key Value Register 0 RO (0x00)

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

TKCVAL1 (0xA145h) Touch Key Value Register 1 RO (0x00)

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

TKCVAL2 (0xA146h) Touch Key Value Register 2 RO (0x00)

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

TKCMVF0 (0xA148h) Touch Key Fast Moving Average Register 0 RO (0x00)

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

TKCMVF1 (0xA149h) Touch Key Fast Moving Average Register 1 RO (0x00)

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

TKCMVF2 (0xA14Ah) Touch Key Fast Moving Average Register 2 RO (0x00)

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

TKCMVS0 (0xA14Ch) Touch Key Slow Moving Average Register 0 RO (0x00)

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

TKCMAS1 (0xA14Dh) Touch Key Slow Moving Average Register 1 RO (0x00)

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

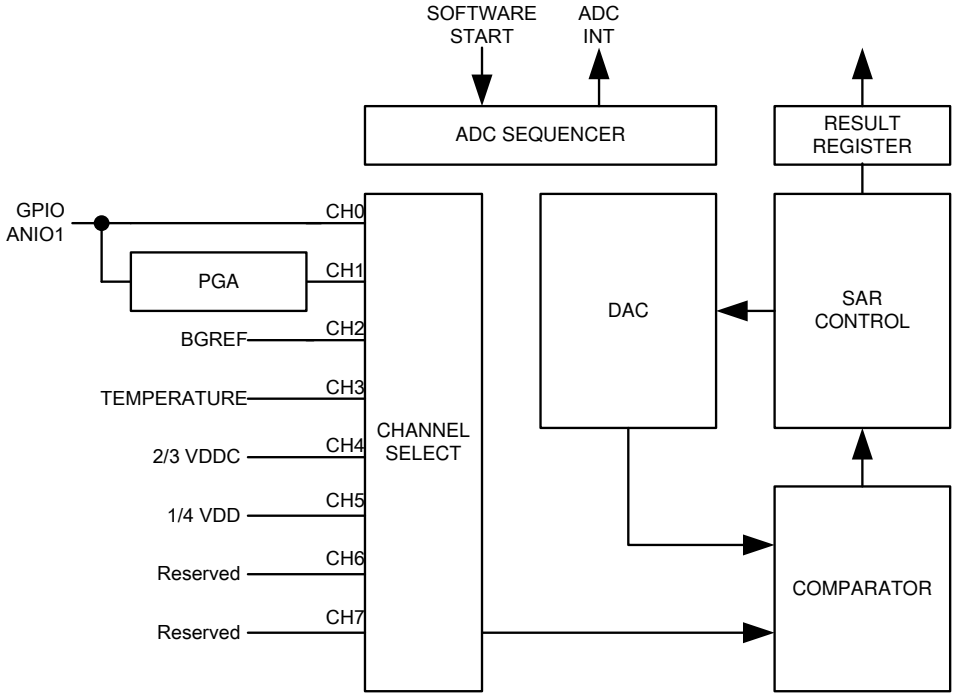
TKCMAS2 (0xA14Eh) Touch Key Slow Moving Average Register 2 RO (0x00)

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

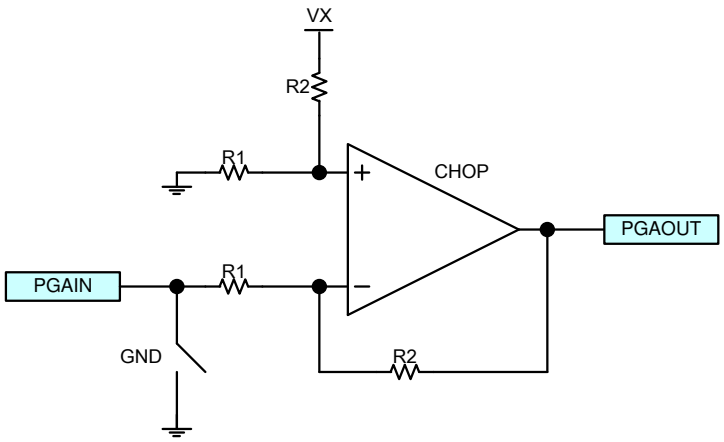
11. 12-Bit SAR ADC (ADC)

The on-chip ADC is a 12-bit SAR based ADC with maximum ADC clock rate of 4MHz (2.5V – 5V) or 500KHz (1.8V – 2.4V) and each conversion takes 64 cycles to complete. The conversion is partitioned into 4-sub conversion (each 16 cycles) for cancellation of internal offsets. Typical ADC accuracy is about 10-Bit to 11-Bit at 5V reference. The ADC clock is programmable and set by the ADC clock scaler. The ADC full range reference can be selected using VDD or VDDC (1.8V). The full range can be calibrated using on-chip 1.1V reference as the input. When enabled, the ADC consumes about 2mA of current. The ADC also includes hardware to perform result average. Average can be set to 1 to 8 times.

In IS31CS8969, The inputs of the ADC are multiplexed with multi-function GPIO's ANIO1 pins. To use these ports as ADC inputs, the ANEN1 must be enabled and other drivers to be in high-impedance state. The block diagram of ADC is shown in the following.



To enhance the resolution of ADC, a Programmable Gain Amplifier (PGA) can be inserted in the input path. The gain setting can be up to 20. The PGA is an inverting type using OPAMP, and its block diagram is shown as following. R1 is approximately 20K Ohm. VX is a voltage reference of VDDH-0.4V or VDDC-0.4V. GND controls the PGA input to 0V. CHOP is used to control OPAMP offset. For precision measurement, software should always first set GND=0 and obtain an ADC result (ADCR1). Then a second result is obtained by connecting to PGAIN (ADCR2). The measurement is obtained by the difference between ADCR1 and ADCR2. In this procedure, offset of the PGA as well as VX is canceled automatically. Because PGA input is connected to the GPIO ANIO bus, when switching the IOCELL's ANIO1, the ADC must also wait after PGA to settle. PGA settling time is approximately 10usec.



ADCCFG (0xA9h) ADC Configuration Register R/W 00000000

	7	6	5	4	3	2	1	0
RD	ADCEN	ADCINTE	ADCFM	TEMPSEN	VDDSEN	PRE[2-0]		
WR	ADCEN	ADCINTE	ADCFM	TEMPSEN	VDDSEN	PRE[2-0]		

ADCEN ADC Enable bit
ADCEN=1 enables ADC.
ADCEN=0 puts ADC into power down mode.
When ADCEN is set from 0 to 1, the program needs to wait at least 20us to allow analog bias to stabilize to ensure ADC's proper functionality.

ADCINTE ADC Interrupt Enable bit
ADCINTE=1 enables the ADC interrupt when conversion completes.
ADCINTE=0 disables the ADC interrupt

ADCFM ADC Result Format Control bit
ADCFM = 1 sets ADC result as MSB justified. ADCAH contains the MSB bits of the result. ADCAL[7-4] contains LSB results and ADCAL[3-0] is filled with 0000.
ADCFM = 0 sets ADC result as LSB justified. ADCAH[7-4] is filled with 0000. ADCAH[3-0] contains MSB result. ADCAL contains the LSB results.

TEMPSEN Temperature Sensor Enable

VDDSEN 2/3VDDC Enable and ¼ VDD Enable
VDDSEN=1 will turn on these two bias network. VDDSEN should be 0 to save power.

PRE[2-0] ADC Clock Divider

PRE[2-0]	ADC CLOCK
0	SYSCLK/2
1	SYSCLK/4
2	SYSCLK/8
3	SYSCLK/16
4	SYSCLK/32
5	SYSCLK/64
6	SYSCLK/128
7	SYSCLK/256

ADCCTL (0xCEh) ADC Control Register R/W 00000000

	7	6	5	4	3	2	1	0
RD	AVG[2-0]			CHSEL[2-0]			ADCIF	CSTART
WR	AVG[2-0]			CHSEL[2-0]			-	CSTART

AVG[2-0] AVG[2-0] controls the hardware averaging logic of ADC readout. It is recommended the setting is changed only when ADC is stopped. If multiple channels are enabled, then each channel is averaged in sequence. The default is 000.

AVG2	AVG1	AVG0	ADC Result
0	0	0	1 Times Average
0	0	1	2 Times Average
0	1	0	4 Times Average
0	1	1	8 Times Average
1	0	0	Test Mode OP/CMP ++
1	0	1	Test Mode OP/CMP +-
1	1	0	Test Mode OP/CMP -+
1	1	1	Test Mode OP/CMP --

CHSEL[2-0] ADC Channel Select

CHSEL[2]	CHSEL[1]	CHSEL[0]	ADC Channel
0	0	0	GPIO ANIO
0	0	1	PGA Output
0	1	0	BGREF Output
0	1	1	Temperature Sensor
1	0	0	2/3VDDC
1	0	1	¼ VDD
1	1	0	Reserved
1	1	1	Reserved

ADCIF — ADC Conversion Completion Interrupt Flag bit
ADCIF is set by hardware when the conversion is completed and new result is written to ADCL and ADCH result registers. If ADC interrupt is enabled, this also generates an interrupt. This bit is cleared when ADCL is read. When this flag is set, no new conversion result is updated.

CSTART — Software Start Conversion bit
Set this CSTART=1 to trigger an ADC conversion on selected channels. This bit is self-cleared when the conversion is done.

ADCPGA (0xB9h) ADC PGA Control Register WO 000XX000

	7	6	5	4	3	2	1	0
RD	-	-	-	-	-	-	-	
WR	PGAEN	REFSEL	VXSEL	-	-	GND	GAIN[1-0]	

PGAEN — PGA Enable

REFSEL — ADC Reference Select
REFSEL=0 uses VDDH as ADC full-scale reference
REFSEL=1 uses VDDC as ADC full-scale reference

VXSEL — VX PGA Reference Select
VXSEL=0 uses VDDH-0.4 as PGA reference
VXSEL=1 uses VDDC-0.4 as PGA reference
Both reference are not accurate and requires calibrations.

GND — When GND=1, the PGA input is isolated from input and is forced to VSS.

GAIN[1-0] — GAIN[1-0] controls the PGA gain. The default is 00.

GAIN[1-0]	Gain
00	5
01	10
10	15
11	20

ADCH and ADCL are the high and low byte result registers respectively, and are read-only. Reading low byte result also clears its corresponding interrupt flag. If the flag is not cleared, no new result is updated. The software should always read the low byte last. The format of the high byte and low byte depends on ADCFM setting.

If ADCFM = 1, the valid ADC Result is located on ADCAH[7:0] and ADCAL[3:0]. If ADCFM = 0, the valid ADC Result is located on ADCAH[3:0] and ADCAL[7:0].

ADCAL (0xBAh) Channel A Result Register Low Byte RO XXXXXXXX

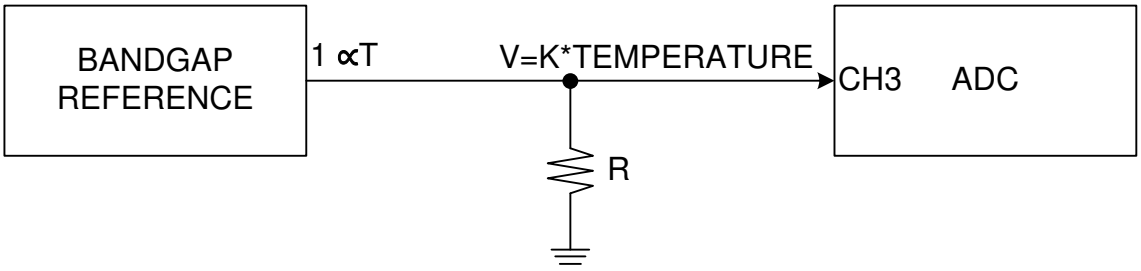
	7	6	5	4	3	2	1	0
RD	ADC Result							
WR	-							

ADCAH (0xBBh) Channel A Result Register High Byte RO XXXXXXXX

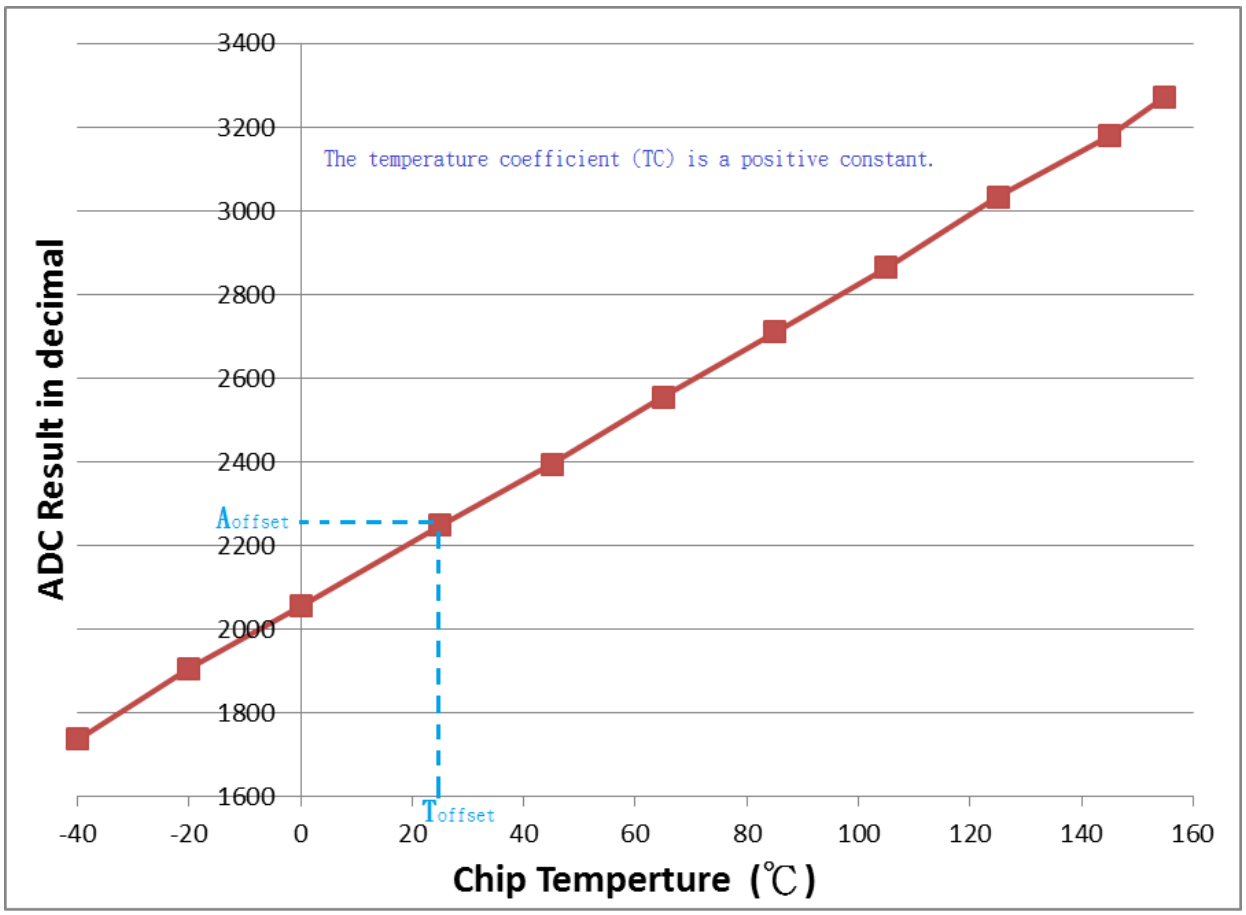
	7	6	5	4	3	2	1	0
RD	ADC Result							
WR	-							

11.1 On-Chip Temperature Sensor

The on-chip temperature sensor is derived from a positive temperature coefficient current from the internal Bandgap circuit. The hardware block diagram is shown in the following:



The current is applied to an on-chip low temperature coefficient resistor and develops to a voltage that is in terms proportional to the temperature. The voltage can be written in as an equation of V = K * Temperature, where K is a constant. The following graph shows the temperature sensor curve with the ADC Result. The vertical axis is the ADC Result in decimal and the horizontal axis is the chip temperature in ℃.



In IS31CS8969, the temperature coefficient (TC) is a positive constant. This calibration value can be found in the Flash Information Block (IFB) address 0x2D. The byte contains the calibrated temperature coefficient for 1℃ in LSB of ADC. This is a binary number in 4:4 formats, i.e. the radix point is between bit 4 and bit 3. For example, 0b0111.1100 refers to 7.75 LSB. If calibration of temperature sensor is not done, then this byte would be written as 0x00.

In IFB 0x2B and 0x2C, these two bytes contain ADC measurement of internal temperature sensor (Aoffset) at calibration temperature (Toffset). The ADC should use VDDC (1.8V) as the full-scale reference. IFB 0x2B is LSB and 0x2C is MSB. The upper 4-bit of the MSB is the offset of the calibration temperature from 20℃. For example, if the

calibration is done at 25℃, then the upper 4-bit should be 0x05. If calibration of temperature sensor is not done, then these two bytes would be written as 0x00.

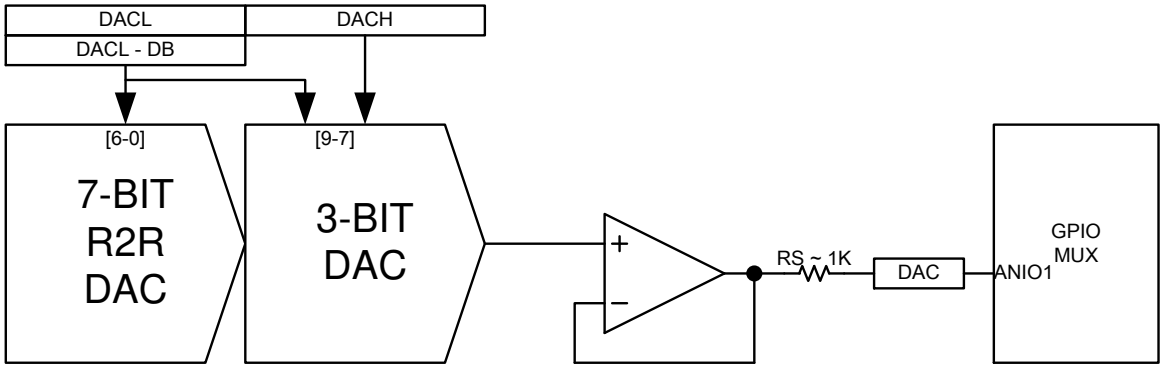
The detection temperature can be represented in the following equation. Where the Adec is the ADC measurement result of ADC at the temperature Tdec.

$$Tdec\ (℃) = (Adec - Aoffset) / TC + 20 + Toffset$$

For example, in IFB TC = 7.75, Toffset = 5, and Aoffset = 2250, if the ADC measurement result Adec = 2870, then the measurement temperature Tdec = (2870-2250) / 7.75 + 20 + 5 = 105 ℃.

12. 10-Bit Voltage Output DAC (VDAC)

A 10-bit voltage output DAC is included. The DAC is composed of LSB 7-bit R2R and MSB 3-bit linear DAC. The output is buffered by unity configured OPAMP. The output range of the DAC is from 0V to VDD. Due to the circuit structure of the OPAMP, the output accuracy will suffer some loss from ¾ VDD to VDD. The output impedance of the buffer is less than 1K Ohm and should not drive high capacitance load. Please also note that the linearity and accuracy of the DAC will suffer when the output is close to rail or 0V because of the OPAMP. The update of DAC must start with low-byte first and then high-byte because the low-byte is double-buffered. The DAC output is connected to P1.1 IOCELL's ANIO1. To use this port as DAC output, P1.1 IOCELL's ANEN1 must be enabled and other drivers to be in high-impedance state.



DACH (0xA037h) DAC High Register R/W 0x00

	7	6	5	4	3	2	1	0
RD	DACEN	-	-	-	-	-	DAC[9-8]	
WR	DACEN	-	-	-	-	-	DAC[9-8]	

DACEN — DAC Enable Control bit
DACEN=1 enables DAC
DACEN=0 disable DAC

DAC[9-8] — DAC[9-8] Data bits
These two bits are MSB of 10-bit DAC data. Writing to this register updates the DAC output.

DACL (0xA036h) DAC Low Register R/W 0x00

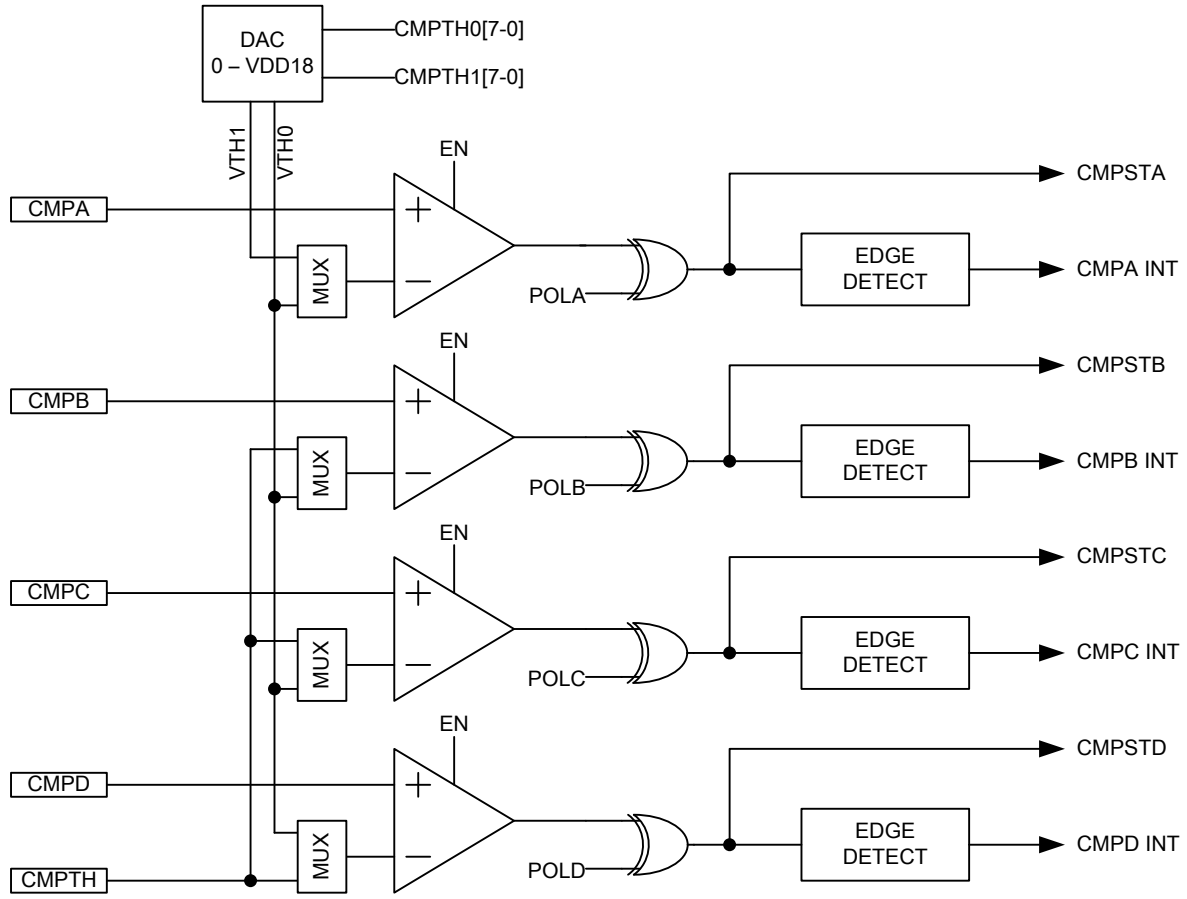
	7	6	5	4	3	2	1	0
RD	DAC[7-0]							
WR	DAC[7-0]							

This register is double-buffered and the output is not updated until DACH is written.

13. Analog Comparators (ACMP)

IS31CS8969 has four analog comparators as its on-chip external peripherals. When enabled, each comparator consumes about 250uA. The input signal range is from 0 to VDD. There are two 8-bit R-2R DAC associated with the comparators to generate the compare threshold. The R-2R DAC uses the internal 1.8V supply as the full-scale range thus limiting the comparator threshold from 0V to 1.8V in 256 steps. Channel B/C/D can select a common external threshold. The inputs of the comparators are multiplexed with multi-function GPIO's ANIO1 pins. To use these ports as comparator inputs, the ANEN1 must be enabled and other drivers to be in high-impedance state. P1.4 is used for CMPA and CMPD thus can be configured to detect two thresholds simultaneously. P1.5 and P1.6 are used for CMPB and CMPC. P1.7 is used for comparator external threshold.

The CPU can read the real-time outputs of the comparator directly through register access. The output is also sent to an edge-detector and any edge transition can be used to trigger an interrupt. The stabilization time from off state to enabled state of the comparator block is about 20usec. The block diagram of the analog comparator is shown in the following diagram.



CMPCFGAB (0xA030h) Analog Comparator A/B Configuration Register R/W 0x00

	7	6	5	4	3	2	1	0
RD	CMPENA	THSELA	INTENA	POLA	CMPENB	THSELB	INTENB	POLB
WR	CMPENA	THSELA	INTENA	POLA	CMPENB	THSELB	INTENB	POLB

CMPENA	Comparator A Enable bit. Set to enable the comparator. When CMPENA is set from 0 to 1, the program needs to wait at least 20us allowing analog bias to stabilize to ensure comparator A's proper functionality.
THSELA	Comparator A Threshold Select bit. THSELA = 0, the comparator A uses VTH0 as the threshold. THSELA = 1, the comparator A uses VTH1 as the threshold.
INTENA	Set to enable the comparator A's interrupt.
POLA	Channel A Output polarity control bit POLA=0 set default polarity POLA=1 reverse the output polarity of the comparator
CMPENB	Comparator B Enable bit. Set to enable the comparator. When CMPENB is set from 0 to 1, the program needs to wait at least 20us allowing analog bias to stabilize to ensure comparator B's proper functionality.

THSELB	Comparator B Threshold Select Bit. THSELB = 0, the comparator B uses VTH0 as the threshold. THSELB = 1, the comparator B uses external threshold.
INTENB	Set to enable the comparator B's interrupt.
POLB	Channel B Output polarity control bit POLB=0 set default polarity POLB=1 reverse the output polarity of the comparator

CMPCFGCD (0xA031h) Analog Comparator C/D Configuration Register R/W 0X00

	7	6	5	4	3	2	1	0
RD	CMPENC	THSELC	INTENC	POLC	CMPEND	THSELD	INTEND	POLD
WR	CMPENC	THSELC	INTENC	POLC	CMPEND	THSELD	INTEND	POLD

CMPENC	Comparator C Enable Bit. Set to enable the comparator. When CMPENC is set from 0 to 1, the program needs to wait at least 20us to allow analog bias to stabilize to ensure comparator C's proper functionality.
THSELC	Comparator C Threshold Select Bit. THSELC = 0, the comparator C uses VTH0 as the threshold. THSELC = 1, the comparator C uses external threshold.
INTENC	Set to enable the comparator C interrupt.
POLC	Channel C Output polarity control bit POLC=0 set default polarity POLC=1 reverse the output polarity of the comparator
CMPEND	Comparator D Enable Bit. Set to enable the comparator. When CMPEND is set from 0 to 1, the program need to wait at least 20us to allow analog bias to stabilize to ensure comparator D's proper functionality.
THSELD	Comparator D Threshold Select Bit. THSELD = 0, the comparator D uses VTH0 as the threshold. THSELD = 1, the comparator D uses external threshold.
INTEND	Set to enable the comparator D interrupt.
POLD	Channel D Output polarity control bit POLD=0 set default polarity POLD=1 reverse the output polarity of the comparator

CMPVTH0 (0xA032h) Analog Comparator Threshold Control Register R/W 0X00

	7	6	5	4	3	2	1	0
RD	VTH0 Register							
WR	VTH0 Register							

CMPVTH0 register controls the comparator threshold VTH0 through 8-bit DAC. When set to 0x00h, the threshold is 0V. When set to 0xFFh, the threshold is at 1.8V. When not used, it should be set to 0x00 to save power consumption.

CMPVTH1 (0xA033h) Analog Comparator Threshold Control Register R/W 0X00

	7	6	5	4	3	2	1	0
RD	VTH1 Register							
WR	VTH1 Register							

CMPVTH1 register controls the comparator threshold VTH1 through 8-bit DAC. When set to 0x00h, the threshold is 0V. When set to 0xFFh, the threshold is at 1.8V. When not used, it should be set to 0x00 to save power consumption.

CMPST (0x94h) Analog Comparator Status Register R/W 0X00

	7	6	5	4	3	2	1	0
RD	CMPIFD	CMPIFC	CMPIFB	CMPIFA	CMPSTD	CMPSTC	CMPSTB	CMPSTA
WR	CMPIFD	CMPIFC	CMPIFB	CMPIFA	CMPHYSB[1-0]		CMPHYSA[1-0]	

CMPIFD	Comparator D Interrupt Flag bit. This bit is set when CMPSTD is toggled and the comparator D setting is enabled. This bit must be cleared by software.
CMPIFC	Comparator C Interrupt Flag bit. This bit is set when CMPSTC is toggled and the comparator C setting is enabled. This bit must be cleared by software.
CMPIFB	Comparator B Interrupt Flag bit. This bit is set when CMPSTB is toggled and the comparator B setting is enabled. This bit must be cleared by software.
CMPIFA	Comparator A Interrupt Flag bit. This bit is set when CMPSTA is toggled and the comparator A setting is enabled. This bit must be cleared by software.
CMPSTD	Comparator D Real-time Output. If the comparator is disabled, this bit is forced low.

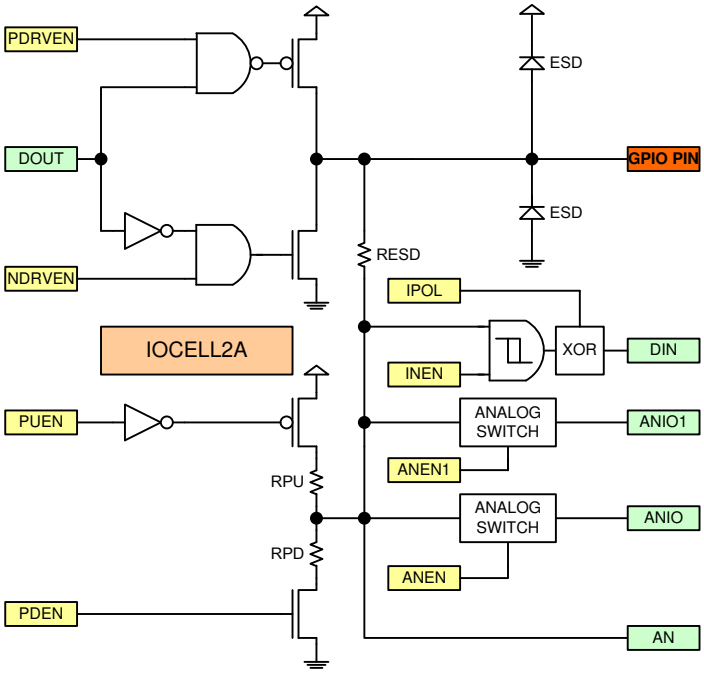
CMPSTC	Comparator C Real-time Output. If the comparator is disabled, this bit is forced low.
CMPSTB	Comparator B Real-time Output. If the comparator is disabled, this bit is forced low.
CMPSTA	Comparator A Real-time Output. If the comparator is disabled, this bit is forced low.
CMPHYSB[1-0]	Comparator B/C/D Hysteresis Enable bit. CMPHYSB[1-0] = 00 disables comparator hysteresis CMPHYSB[1-0] = 01 comparator hysteresis = 10mV CMPHYSB[1-0] = 10 comparator hysteresis = 20mV CMPHYSB[1-0] = 10 comparator hysteresis = 30mV
CMPHYSA[1-0]	Comparator A Hysteresis Enable bit. CMPHYSA[1-0] = 00 disables comparator hysteresis CMPHYSA[1-0] = 01 comparator hysteresis = 10mV CMPHYSA[1-0] = 10 comparator hysteresis = 20mV CMPHYSA[1-0] = 10 comparator hysteresis = 30mV

14. GPIO Port Function and Pin Configurations

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; IS31CS8969 employs a configurable I/O buffer design. 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, ADC input or DAC output.

14.1 IO Cell Configurations

The supply voltage of the I/O buffer uses VDD (2.5V to 5.5V). The input and output level is referenced to VDD and 0V. Since the design is standardized, the I/O design offers a uniform ESD performance. The functional block diagram of the standard I/O buffer is shown in the following diagram.



IOCELL2A is IOCELL with two analog multiplexer switches, ANIO and ANIO1. The control of IOCELL2A uses IOCFG's ANEN1 to replace LATEN.

From the diagram, there are 7 control bits for the IOCFGPx.y register, and these registers are located at XFR 0xA040 – 0xA047 for P0.0 to P0.7, 0xA048 – 0xA04F for P1.0 to P1.7, 0xA060 – 0xA063 for P2.0 to P2.3, and the definitions of IOCFGPx.y are described in the following table.

IOCFGP (0xA040h – 0xA04Fh, 0xA060 – 0xA06F) IO Buffer Configuration Registers R/W (0x00)

	7	6	5	4	3	2	1	0
RD	INEN	ANEN1	PUEN	PDEN	ANEN	PDRVEN	NDRVEN	IPOL
WR	INEN	ANEN1	PUEN	PDEN	ANEN	PDRVEN	NDRVEN	IPOL

INEN	Input buffer control. Set this bit to enable the GPIO input buffer. If the input buffer is not used, it should be disabled to prevent leakage current when pin is floating. DISABLE is the default value.
ANEN1	Analog MUX 1 enables control. Set this bit to connect the pin to the internal analog peripheral. DISABLE is the default value.
PUEN	Pull up resistor enable control. Set this bit to enable pull-up resistor connection to the pin. The pull-up resistor is approximately 100K Ohm. DISABLE is the default value.
PDEN	Pull down resistor enable control. Set this bit to enable pull-down resistor connection to the pin. The pull-down resistor is approximately 100K Ohm. DISABLE is the default value.
ANEN	Analog MUX enable control. Set this bit to connect the pin to the internal analog peripheral. DISABLE is the default value.

PDRVEN	Output PMOS driver enable. 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.
IPOL	Input logic polarity. IPOL=0 for normal polarity and IPOL=1 for reverse polarity.

The following table shows various configurations of the I/O buffer.

IO Functions	INEN	PUEN	PDEN	ANEN/ANEN1	PDRVEN	NDRVEN
Input only	1	0	0	0	0	0
Input /w pull up	1	1	0	0	0	0
Input /w pull down	1	0	1	0	0	0
Output with CMOS push-pull	0	0	0	0	1	1
Output /w NMOS open-drain (sink)	0	0	0	0	0	1
Output /w NMOS open-drain (sink) and weak pull up	0	1	0	0	0	1
Output /w PMOS open-drain (source)	0	0	0	0	1	0
Output /w PMOS open-drain (source) and weak pull down	0	0	1	0	1	0
I/O 8051 like	1	1	0	0	0	1
I/O CMOS	1	0	0	0	1	1
Analog function	0	0	0	1	0	0
Oscillator pin	0	0	0	0	0	0

Please note the following exceptions exist for IOCFG registers.

14.2 GPIO Port Multifunction

Because each signal pin is a multi-functional and the function is shared with GPIO port, therefore each pin requires MFCFGP register to control, which function is in effect and which peripherals are connected to the signal pins. These selection and definitions are pin specific and product specific. The following description describes the selection and control for IS31CS8969 signal pins.

MFCFGP0.0 (0xA050) GPIO P0.0 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY0EN	MSDAEN	SSDA2EN	CEX0EN	MISOEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY0EN	MSDAEN	SSDA2EN	CEX0EN	MISOEN	PINT1EN	PINT0EN	GPIOEN

KEY0EN	Touch Key 0 Enable bit. IOCFGP0.0 ANEN must be enabled too.
MSDAEN	MSDAEN=1 enables this pin as I2CM SDA I/O. This must be configured as OD output.
SSDA2EN	SSDA2EN=1 enables this pin as I2CS2 SDA I/O. This must be configured as OD output.
CEX0EN	CEX0EN=1 enable this pin as CEX I/O for CCP0.
MISOEN	MISOEN=1 uses this pin as SPI MISO I/O.
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP0.1 (0xA051h) GPIO P0.1 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY1EN	MSCLEN	SSCL2EN	CEX1EN	SCKEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY1EN	MSCLEN	SSCL2EN	CEX1EN	SCKEN	PINT1EN	PINT0EN	GPIOEN

KEY1EN	Touch Key 1 Enable bit. IOCFGP0.1 ANEN must be enabled too.
MSCLEN	MSCLEN=1 enables this pin as I2CM SCL I/O. This must be configured as OD output.
SSCL2EN	SSCL2EN=1 enables this pin as I2CS2 SCL I/O. This must be configured as OD output.
CEX1EN	CEX1EN=1 enable this pin as CEX I/O for CCP1.
SCKEN	SCKEN=1 uses this pin as SPI SCK I/O
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP0.2 (0xA052h) GPIO P0.2 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY2EN	RXD2EN	RXD0EN	CEX2EN	MOSIEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY2EN	RXD2EN	RXD0EN	CEX2EN	MOSIEN	PINT1EN	PINT0EN	GPIOEN

KEY2EN	Touch Key 2 Enable bit. IOCFGP0.2 ANEN must be enabled too.
RXD2EN	RXD2EN=1 use this pin as RXD input for EUART2
RXD0EN	RXD0EN=1 uses this pin as RXD input for UART0
CEX2EN	CEX2EN=1 enable this pin as CEX I/O for CCP2.
MOSIEN	MOSIEN=1 uses this pin as SPI MOSI I/O.
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP0.3 (0xA053h) GPIO P0.3 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY3EN	TXD2EN	TXD0EN	CEX3EN	SSNEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY3EN	TXD2EN	TXD0EN	CEX3EN	SSNEN	PINT1EN	PINT0EN	GPIOEN

KEY3EN	Touch Key 3 Enable bit. IOCFGP0.3 ANEN must be enabled too.
TXD2EN	TXD2EN=1 use this pin as TXD output for EUART2
TXD0EN	TXD0EN=1 uses this pin as TXD output for UART0
CEX3EN	CEX3EN=1 enable this pin as CEX I/O for CCP3.
SSNEN	SSNEN=1 uses this pin as SPI SSN input.
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP0.4 (0xA054h) GPIO P0.4 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY4EN	MSDAEN	SSDA2EN	PWM0EN	MISOEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY4EN	MSDAEN	SSDA2EN	PWM0EN	MISOEN	PINT1EN	PINT0EN	GPIOEN

KEY4EN	Touch Key 4 Enable bit. IOCFGP0.4 ANEN must be enabled too.
MSDAEN	MSDAEN=1 enables this pin as I2CM SDA I/O. This must be configured as OD output.
SSDA2EN	SSDA2EN=1 enables this pin as I2CS2 SDA I/O. This must be configured as OD output.
PWM0EN	PWM0EN =1 use this pin as PWM0 output
MISOEN	MISOEN=1 uses this pin as SPI MISO I/O.
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP0.5 (0xA055h) GPIO P0.5 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY5EN	MSCLEN	SSCL2EN	PWM1EN	SCKEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY5EN	MSCLEN	SSCL2EN	PWM1EN	SCKEN	PINT1EN	PINT0EN	GPIOEN

KEY5EN	Touch Key 5 Enable bit. IOCFGP0.5 ANEN must be enabled too.
MSCLEN	MSCLEN=1 enables this pin as I2CM SCL I/O. This must be configured as OD output.
SSCL2EN	SSCL2EN=1 enables this pin as I2CS2 SCL I/O. This must be configured as OD output.
PWM1EN	PWM0EN =1 use this pin as PWM1 output
SCKEN	SCKEN=1 uses this pin as SPI SCK I/O
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP0.6 (0xA056h) GPIO P0.6 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY6EN	MSDAEN	SSDA2EN	PWM0EN	MISOEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY6EN	MSDAEN	SSDA2EN	PWM0EN	MISOEN	PINT1EN	PINT0EN	GPIOEN

KEY6EN	Touch Key 6 Enable bit. IOCFGP0.6 ANEN must be enabled too.
MSDAEN	MSDAEN=1 enables this pin as I2CM SDA I/O. This must be configured as OD output.
SSDA2EN	SSDA2EN=1 enables this pin as I2CS2 SDA I/O. This must be configured as OD output.
PWM0EN	PWM0EN =1 use this pin as PWM0 output.
MISOEN	MISOEN=1 uses this pin as SPI MISO I/O.
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP0.7 (0xA057h) GPIO P0.7 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY7EN	MSCLEN	SSCL2EN	BZEN	SCKEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY7EN	MSCLEN	SSCL2EN	BZEN	SCKEN	PINT1EN	PINT0EN	GPIOEN

KEY7EN Touch Key 7 Enable bit. IOCFGP0.7 ANEN must be enabled too.
MSCLEN MSCLEN=1 enables this pin as I2CM SCL I/O. This must be configured as OD output.
SSCL2EN SSCL2EN=1 enables this pin as I2CS2 SCL I/O. This must be configured as OD output.
BZEN BZEN=1 uses this pin as buzzer output.
SCKEN SCKEN=1 uses this pin as SPI SCK I/O.
PINT1EN Pin Interrupt Enable Control Bit.
PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN Pin Interrupt Enable Control Bit.
PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP1.0 (0xA058h) GPIO P1.0 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY8EN	RXD2EN	RXD0EN	CEX4EN	MOSIEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY8EN	RXD2EN	RXD0EN	CEX4EN	MOSIEN	PINT1EN	PINT0EN	GPIOEN

KEY8EN Touch Key 8 Enable bit. IOCFGP1.0 ANEN must be enabled too.
RXD2EN RXD2EN=1 use this pin as RXD input for EUART2
RXD0EN RXD0EN=1 uses this pin as RXD input for UART0
CEX4EN CEX4EN=1 enable this pin as CEX I/O for CCP4.
MOSIEN MOSIEN=1 uses this pin as SPI MOSI I/O.
PINT1EN Pin Interrupt Enable Control Bit.
PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN Pin Interrupt Enable Control Bit.
PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP1.1 (0xA059h) GPIO P1.1 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY9EN	TXD2EN	TXD0EN	CEX5EN	SSNEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY9EN	TXD2EN	TXD0EN	CEX5EN	SSNEN	PINT1EN	PINT0EN	GPIOEN

KEY9EN Touch Key 9 Enable bit. IOCFGP1.1 ANEN must be enabled too.
TXD2EN TXD2EN=1 use this pin as TXD output for EUART2
TXD0EN TXD0EN=1 uses this pin as TXD output for UART0
CEX5EN CEX5EN=1 enable this pin as CEX I/O for CCP5.
SSNEN SSNEN=1 uses this pin as SPI SSN input.
PINT1EN Pin Interrupt Enable Control Bit.
PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN Pin Interrupt Enable Control Bit.
PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP1.2 (0xA05Ah) GPIO P1.2 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY10EN	MSDAEN	SSDA2EN	PWM0EN	MOSIEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY10EN	MSDAEN	SSDA2EN	PWM0EN	MOSIEN	PINT1EN	PINT0EN	GPIOEN

KEY10EN	Touch Key 10 Enable bit. IOCFGP1.2 ANEN must be enabled too.
MSDAEN	MSDAEN=1 enables this pin as I2CM SDA I/O. This must be configured as OD output.
SSDA2EN	SSDA2EN=1 enables this pin as I2CS2 SDA I/O. This must be configured as OD output.
PWM0EN	PWM0EN =1 use this pin as PWM0 output.
MOSIEN	MOSIEN=1 uses this pin as SPI MOSI I/O.
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP1.3 (0xA05Bh) GPIO P1.3 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY11EN	MSCLEN	SSCL2EN	PWM1EN	SSNEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY11EN	MSCLEN	SSCL2EN	PWM1EN	SSNEN	PINT1EN	PINT0EN	GPIOEN

KEY11EN	Touch Key11 Enable bit. IOCFGP1.3 ANEN must be enabled too.
MSCLEN	MSCLEN=1 enables this pin as I2CM SCL I/O. This must be configured as OD output.
SSCL2EN	SSCL2EN=1 enables this pin as I2CS2 SCL I/O. This must be configured as OD output.
PWM1EN	PWM1EN =1 use this pin as PWM1 output.
SSNEN	SSNEN=1 uses this pin as SPI SSN input.
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP1.4 (0xA05Ch) GPIO P1.4 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY12EN	MSDAEN	SSDA2EN	BZEN	MISOEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY12EN	MSDAEN	SSDA2EN	BZEN	MISOEN	PINT1EN	PINT0EN	GPIOEN

KEY12EN	Touch Key12 Enable bit. IOCFGP1.4 ANEN must be enabled too.
MSDAEN	MSDAEN=1 enables this pin as I2CM SDA I/O. This must be configured as OD output.
SSDA2EN	SSDA2EN=1 enables this pin as I2CS2 SDA I/O. This must be configured as OD output.
BZEN	BZEN=1 uses this pin as buzzer output.
MISOEN	MISOEN=1 uses this pin as SPI MISO I/O.
PINT1EN	Pin Interrupt Enable Control Bit. PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN	Pin Interrupt Enable Control Bit. PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP1.5 (0xA05Dh) GPIO P1.5 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY13EN	MSCLEN	SSCL2EN	PWM1EN	SCKEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY13EN	MSCLEN	SSCL2EN	PWM1EN	SCKEN	PINT1EN	PINT0EN	GPIOEN

KEY13EN Touch Key13 Enable bit. IOCFGP1.5 ANEN must be enabled too.
MSCLEN MSCLEN=1 enables this pin as I2CM SCL I/O. This must be configured as OD output.
SSCL2EN SSCL2EN=1 enables this pin as I2CS2 SCL I/O. This must be configured as OD output.
PWM1EN PWM1EN =1 use this pin as PWM1 output
SCKEN SCKEN=1 uses this pin as SPI SCK I/O
PINT1EN Pin Interrupt Enable Control Bit.
PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN Pin Interrupt Enable Control Bit.
PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP1.6 (0xA05Eh) GPIO P1.6 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY14EN	RXD2EN	RXD0EN	PWM1EN	MOSIEN	PINT1EN	PINT0EN	GPIOEN
WR	KEY14EN	RXD2EN	RXD0EN	PWM1EN	MOSIEN	PINT1EN	PINT0EN	GPIOEN

KEY14EN Touch Key 14 Enable bit. IOCFGP1.6 ANEN must be enabled too.
RXD2EN RXD2EN=1 use this pin as RXD input for EUART2
RXD0EN RXD0EN=1 uses this pin as RXD input for UART0
PWM1EN PWM1EN =1 use this pin as PWM1 output.
MOSIEN MOSIEN=1 uses this pin as SPI MOSI I/O.
PINT1EN Pin Interrupt Enable Control Bit.
PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN Pin Interrupt Enable Control Bit.
PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP1.7 (0xA05Fh) GPIO P1.7 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY19EN	TXD2EN	TXD0EN	PWM0EN	-	PINT1EN	PINT0EN	GPIOEN
WR	KEY19EN	TXD2EN	TXD0EN	PWM0EN	-	PINT1EN	PINT0EN	GPIOEN

KEY19EN Touch Key 19 Enable bit. IOCFGP1.7 ANEN must be enabled too.
TXD2EN TXD2EN=1 use this pin as TXD output for EUART2
TXD0EN TXD0EN=1 uses this pin as TXD output for UART0
PWM0EN PWM0EN =1 use this pin as PWM0 output.
PINT1EN Pin Interrupt Enable Control Bit.
PINT1EN=1 configures this pin as an input condition to PINT1 interrupt. More than one pin can be assigned to PINT1. And one pin can be assigned to both PINT0 both PINT1.
PINT0EN Pin Interrupt Enable Control Bit.
PINT0EN=1 configures this pin as an input condition to PINT0 interrupt. More than one pin can be assigned to PINT0. And one pin can be assigned to both PINT0 both PINT1.
GPIOEN GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP2.0 (0xA070h) GPIO P2.0 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY15EN	MSDAEN	SSDA2EN	BZEN	MISOEN	-	T1EN	GPIOEN
WR	KEY15EN	MSDAEN	SSDA2EN	BZEN	MISOEN	-	T1EN	GPIOEN

KEY15EN	Touch Key 15 Enable bit. IOCFGP2.0 ANEN must be enabled too.
MSDAEN	MSDAEN=1 enables this pin as I2CM SDA I/O. This must be configured as OD output.
SSDA2EN	SSDA2EN=1 enables this pin as I2CS2 SDA I/O. This must be configured as OD output.
BZEN	BZEN=1 enables this pin as buzzer output.
MISOEN	MISOEN=1 uses this pin as SPI MISO I/O.
T1EN	T1EN=1 enables this pin as Timer 1 input.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP2.1 (0xA071h) GPIO P2.1 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY16EN	MSCLEN	SSCL2EN	BZEN	SCKEN	-	T2EN	GPIOEN
WR	KEY16EN	MSCLEN	SSCL2EN	BZEN	SCKEN	-	T2EN	GPIOEN

KEY16EN	Touch Key 16 Enable bit. IOCFGP2.1 ANEN must be enabled too.
MSCLEN	MSCLEN=1 enables this pin as I2CM SCL I/O. This must be configured as OD output.
SSCL2EN	SSCL2EN=1 enables this pin as I2CS2 SCL I/O. This must be configured as OD output.
BZEN	BZEN=1 enables this pin as buzzer output.
SCKEN	SCKEN=1 uses this pin as SPI SCK I/O
T2EN	T2EN=1 uses this pin as Timer 2 Input.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP2.2 (0xA072h) GPIO P2.2 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY17EN	RXD2EN	RXD0EN	PWM1EN	MOSIEN		T2EXEN	GPIOEN
WR	KEY17EN	RXD2EN	RXD0EN	PWM1EN	MOSIEN		T2EXEN	GPIOEN

KEY17EN	Touch Key 17 Enable bit. IOCFGP2.2 ANEN must be enabled too.
RXD2EN	RXD2EN=1 use this pin as RXD input for EUART2
RXD0EN	RXD0EN=1 uses this pin as RXD input for UART0
PWM1EN	PWM1EN =1 use this pin as PWM1 output.
MOSIEN	MOSIEN=1 uses this pin as SPI MOSI I/O.
T2EXEN	T2EXEN=1 enables this pin as T2EX input for Timer 2.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default value.

MFCFGP2.3 (0xA073h) GPIO P2.3 Function Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	KEY18EN	TXD2EN	TXD0EN	PWM0EN	SSNEN		T0EN	GPIOEN
WR	KEY18EN	TXD2EN	TXD0EN	PWM0EN	SSNEN		T0EN	GPIOEN

KEY18EN	Touch Key 18 Enable bit. IOCFGP2.3 ANEN must be enabled too.
TXD2EN	TXD2EN=1 use this pin as TXD output for EUART2
TXD0EN	TXD0EN=1 uses this pin as TXD output for UART0
PWM0EN	PWM0EN =1 use this pin as PWM0 output.
SSNEN	SSNEN=1 uses this pin as SPI SSN input.
T0EN	T0EN=1 enables this pin as Timer 0 input.
GPIOEN	GPIO Function Enable Bit. Set this bit to enable GPIO function. DISABLE is the default

14.3 GPIO Edge Interrupt

The GPIO pins can be configured as external pin interrupt input or for wake up purpose. Each port has edge detection logic and latch for rising and falling edge detections.

PIOEDGR0 (0xA028h) Port0 IO Input Rising Edge Detection Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PR0[7-0]							
WR	PREN0[7-0]							

PR0[7-0] PORT0.0 to PORT0.7 Rising Edge Detection Status
PR0[i] is set by hardware when a rising edge is detected on PORT0.i input if PREN0[i]=0.
PR0[i] is latched and must be cleared by software by writing PREN0[i]=0.
PR0[i] is also used to generate the PIN interrupt if corresponding port PINTEN is enabled.

PREN0 [7-0] Port 0 Falling Edge Detection Enable
PREN0[i]=1 enables the rising edge detection.

PIOEDGF0 (0xA038h) Port0 IO Input Falling Edge Detection Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PF0[7-0]							
WR	PFEN0[7-0]							

PF0[7-0] PORT0.0 to PORT0.7 Falling Edge Detection Status
PF0[i] is set by hardware when a Falling edge is detected on PORT0.i input if PFEN0[i]=0.
PF0[i] is latched and must be cleared by software by writing PREN0[i]=0.
PF0[i] is also used to generate the PIN interrupt if corresponding port PINTEN is enabled.

PFEN0[7-0] Port 0 Falling Edge Detection Enable
PFEN0[i]=1 enables the rising edge detection.

PIOEDGR1 (0xA029h) Port1 IO Input Rising Edge Detection Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PR1[7-0]							
WR	PREN1[7-0]							

PR1[7-0] PORT1.0 to PORT1.7 Rising Edge Detection Status
PR1[i] is set by hardware when a rising edge is detected on PORT1.i input if PREN1[i]=0.
PR1[i] is latched and must be cleared by software by writing PREN1[i]=0.
PR1[i] is also used to generate the PIN interrupt if corresponding port PINTEN is enabled.

PREN1[7-0] Port 1 Falling Edge Detection Enable
PREN1[i]=1 enables the rising edge detection.

PIOEDGF1 (0xA039h) Port1 IO Input Falling Edge Detection Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	PF1[7-0]							
WR	PFEN1[7-0]							

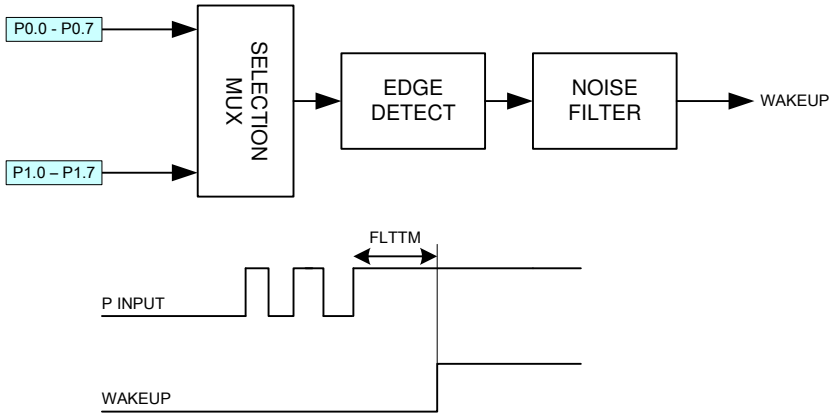
PF1[7-0] PORT1.0 to PORT1.7 Falling Edge Detection Status
PF1[i] is set by hardware when a Falling edge is detected on PORT1.i input if PFEN0[i]=0.
PF1[i] is latched and must be cleared by software by writing PREN1[i]=0.
PF1[i] is also used to generate the PIN interrupt if corresponding port PINTEN is enabled.

PFEN1[7-0] Port 1 Falling Edge Detection Enable
PFEN1[i]=1 enables the rising edge detection.

14.4 GPIO Noise filtered Wake Up

The CPU in IS31CS8969 can enter sleep mode to save standby power consumption. An external pin status change can trigger the wake up of CPU. Each port pin in GPIO Port 0 and Port 1 can be configured as GPIO Edge interrupts as described in 14.3. Any enabled edge interrupt can also serve as wake up event, i.e. the event performs wake up of CPU and at the same time triggers an interrupt.

In additional to the edge interrupt wake-up mechanism, a separate GPIO wake up scheme is included which provide noise filter on the triggering signals. The triggering conditions can be defined as either or both of the input rising and falling edges. A noise filter is included following the edge detection as shown in the following diagram. Any noise pulses less than FLTTM are filtered out. The detection sources are GPIO port P0.0 to P0.7 and P1.0 to P1.7, and only one of this can be selected at one time.



GPWKCFG (0xA02Eh) GPIO Wakeup Configuration Register R/W (0x00)

	7	6	5	4	3	2	1	0
RD	WKRF	WKFF	FLTTM[1:0]		WKSRC[3:0]			
WR	WKRE	WKFE	FLTTM[1:0]		WKSRC[3:0]			

WKRE Rising Edge Wake Up Enable
WKRE=1 enables rising edge wake up

WKRF Rising Edge Wake Up Flag
WKRF is set to 1 by hardware after rising edge wakeup trigger occurred. WKRF must be cleared by setting WKRE=0.

WKFE Falling Edge Wake Up Enable
WKFE=1 enables Falling edge wake up

WKFF Falling Edge Wake Up Flag
WKFF is set to 1 by hardware after rising edge wakeup trigger occurred. WKFF must be cleared by setting WKFE=0.

FLTTM[1:0] Noise Filter Setting
00 = 1 msec
01 = 2 msec
10 = 4 msec
11 = 8 msec

WKSRC[3:0] Wake-Up GPIO Input Selection
0000 = P0.0
0001 = P0.1
0010 = P0.2
0011 = P0.3
0100 = P0.4
0101 = P0.5
0110 = P0.6
0111 = P0.7
1000 = P1.0
1001 = P1.1
1010 = P1.2
1011 = P1.3

1100 = P1.4
1101 = P1.5
1110 = P1.6
1111 = P1.7

15. IFB Block and Writer Mode and Boot Code/ISP

15.1 IFB Block

The main flash memory is 32Kx8 and also contains a separate 256B Information Block (IFB). The IFB is partitioned into two parts. 00 to 4F range contains critical manufacture and calibration information. And 50 to FF range contain user data, which can be programmed one time. The IFB cannot be erased but programmable through Flash Controller Command but it can be erased and written through Writer mode. The user data portion can only serve as One-Time-Programmable storage by the user. The following table shows the IFB contents.

ADDRESS	TYPE	DESCRIPTION
00 – 01	M	IFB Version. 0x00 is MSB and 0x01 is LSB.
02 – 07	M	Product Name. 0x02 and 0x03 is “CS”. These fields use ASCII coding.
08 - 09	M	Package and Product Code
0A – 0B	M	Product Version and Revision. These fields use ASCII coding.
0C	M	Flash Memory Size This byte contains the effective FLASH memory variations. Each bit represents 1/16th of the maximum memory size. For example the largest flash memory is 64KB, then BIT[7-0]=11111111 indicates 64KB. BIT[7-0]=00001111 indicates 32KB FLASH.
0D	M	SRAM Memory Size This byte contains the effective SRAM memory variations. Each bit represents 1/16th of the maximum memory size. For example the largest flash memory is 2KB, then BIT[7-0]=11111111 indicates 2KB. BIT[7-0]=00001111 indicates 1KB SRAM.
0E – 0F	M	These bytes contain customer specific information and are used to differentiate shipment for specific customer. For standard distributions, these bytes remain for 0xFF.
10	M	CP1 Information. This byte is written as 0x00 if CP1 is performed, otherwise 0xFF is present.
11	M	CP2 Information. This byte is written as 0x00 if CP2 is performed, otherwise 0xFF is present.
12	M	CP3 Version. This byte is written a value not equal to 0xFF to indicate CP3 is performed.
13	M	CP3 BIN. The value is binary coding. If CP3 is not performed, this value is 0xFF.
14	M	FT Version. This byte is written a value not equal to 0xFF to indicate FT is performed.
15	M	FT BIN. The value is binary coding. If FT is not performed, this value is 0xFF.
16 - 1B	M	Last Test Date. These fields use ASCII coding.
1C – 1D	M	Boot Code Version. These two bytes contain the boot code version with two digits in ASCII code.
1E	M	Boot Code Segment. This byte contains the MSB byte of the address of the last instruction of the boot code. If no boot code is written, this value is 0xFF.
1F	M	Checksum for 0x00 – 0x1E. Checksum is the XOR results of all bytes.
20	M	REGTRM value for 1.8V
21	M	IOSC ITRM value for 16MHz
22	M	IOSC VTRM value for 16MHz
23	M	LVDTHD value for detection of 4.0V
24	M	LVDTHD value for detection of 3.0V. If not calibrated, this byte should be written 0x00.
25	M	IOSC ITRM value for 12MHz. If not calibrated, this byte should be written 0x00.
26	M	IOSC VTRM value for 12MHz. If not calibrated, this byte should be written 0x00.
27	M	IOSC ITRM value for 8MHz. If not calibrated, this byte should be written 0x00.
28	M	IOSC VTRM value for 8MHz. If not calibrated, this byte should be written 0x00.
29	M	IOSC ITRM value for 4MHz. If not calibrated, this byte should be written 0x00.
2A	M	IOSC VTRM value for 4MHz. If not calibrated, this byte should be written 0x00.
2B – 2C	M	Temperature Offset LSB/MSB. The ADC should use VDD18 as the full-scale reference. 0x2B is LSB and 0x2C is MSB. The upper 4-bit of the MSB is the offset of the calibration temperature from 20C. If not calibrated, these bytes should be written as 0x00.
2D	M	Temperature Coefficient. This byte contains the calibrated temperature coefficient for 10C in LSB of ADC. This is a binary number in 4.4 formats, i.e. the radix point is between bit 4 and bit 3. For example, 0b1011.1010 refers to 11.625 LSB. If calibration

		of temperature sensor is not done, then these two bytes should be written as 0x00.
2E – 2F	M	Internal Reference LSB/MSB
30 – 38	M	Reserved. These bytes are reserved for future extensions and should be written as 0x00.
39	M	Checksum for 0x20 – 0x39. Checksum is the XOR results of all bytes.
3A – 3F	M	Retention Value. These bytes are used to check general flash retention conditions.
40	M/U	Boot Code Wait Time. Boot code uses this byte to determine the ISP wait-time. This wait-time is necessary for stable ISP. After user program download, the wait time can be reduced to minimize power-on time. Each "1" in bit [1-0] constitute 1 second and bits [3-2] constitute 2 second and bits [7-6] are I2CSCL2 and I2CSCL1 check. For example, 0b10000111 is 4 second waits 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	M/U	Boot Code LVR
42	M/U	User Code Protect L
43	M/U	User Code Protect H
44 - FF	U	User One-Time Programmable Space

Note 1: M data cannot be modified and can only be written in writer mode when the entire Flash is erased.

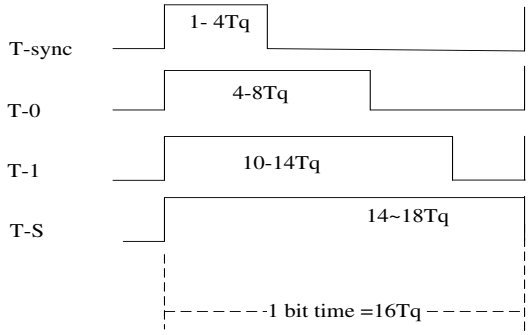
Note 2: U data reads out as FF after the Flash is erased. It can only be programmed once after the Flash is mass erased.

****** The erasure of IFB or modifications of manufacture information in IFB void any manufacture warranty.

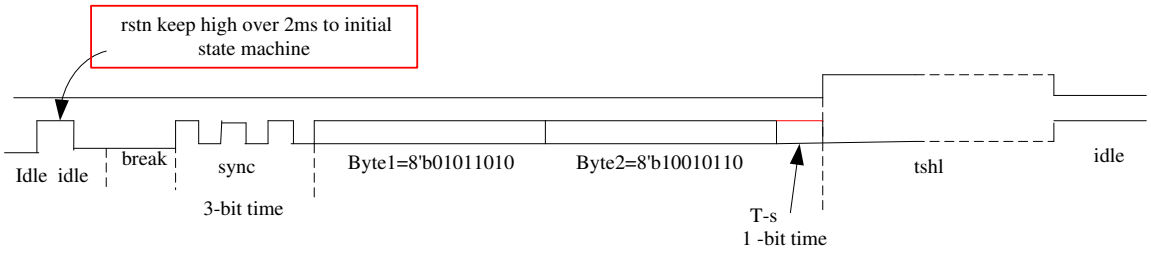
**** This table is for reference only. Please refer to most updated AP note and boot code documents.

15.2 Test Mode and Writer Mode Entry from RSTN

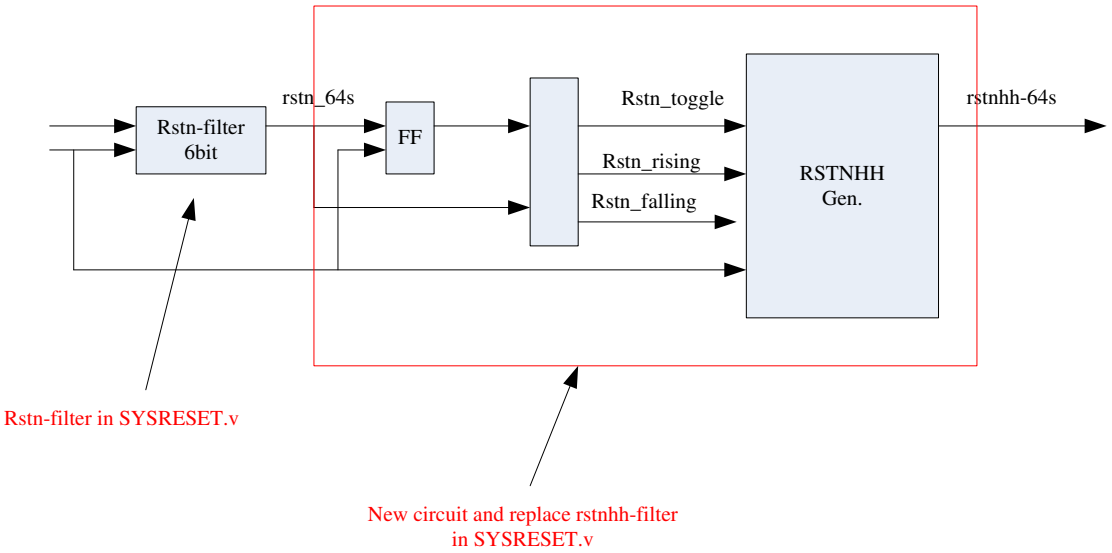
RSTN pin serve as external reset input. With proper input waveform, RSTN is also used to enter test and WRITER mode. The waveform starts with RSTN=0, and uses duty cycle to encode bit stream. Each bit time is divided into 16-T and is encoded into 4 valid states as shown in the following.



The entry is allowed in the following exact sequence. And if RSTN=1 for over 2 msec, then the test mode is forced to exit.



The sequence is initiated for RSTN=0 > 4msec (BREAK field), then followed by three SYNC bit, and two bytes sequence and a T-S bit. RSTN then must return to 0 to maintain the test mode. If the sequence is not exactly matched, a BREAK field must be inserted to start over the sequence. The hardware block diagram is shown in the following and key parameters are also illustrated.



1. idle : porn or any bit timing mis-match
or RSTN go High in Tshl stage
or RSTN =1 over 2ms
or sync-timeout or bittime-over
2. break frame : stage in idle , RSTN go low and keep low > 4ms
3. sync : BREAK frame 的三個 SYNC 後會自動更新(bit time), 並把計算出來的值放進 RSTNCS[11-0] , default bit time = 130us (isoc=16Mhz)
4. byte1 = 8'h5A
5. byte2 = 8'h96
6. Tshh , match TS timing
7. Tshl : RSTNHH=1
8. bittime-over : (T-0 | T-1) period > 18Tq
9. sync-timeout : T-sync period > 12'hfff

15.3 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 set up, 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 power-on state of the device deactivates the writer mode. Only ERASEMM and ERASEMMIFB, and READVERIFYMM commands can be executed. It is activated by READVERIFY the range of 0xFFF8~0xFFFF where a security key can be placed by the user program. The probability of guessing the key is 1 in 2^64 = 1.8E19. Since each trial of READVERIFY 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.

15.3.1 Writer Mode Pins

PIN	PIN#	TYPE	WRITER MODE PIN DESCRIPTION
VDD	1	P	VDD should be connected to a solid 5.0V supply with good decoupling to VSS
VSS	23/19/15	G	Tie to 0V and have good decoupling to VDD
VDDC	24/20/16	PO	Have a 10uF and a 0.1uF good decoupling to VSS
P1.1	12/12/10	O	TDO Data Output
P1.4	15/13/11	I	TDI Data Input
P1.5	16/14/12	I	TCLK Clock Input
P1.6	17/15/13	I	TENB Test Enable Input. Low assertion.

15.4 Boot Code and In-System-Programming

After production testing of the packaged devices, the manufacture 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 0x07000 to 0x77FF. The boot code is executed after resets. The boot code first reads 0x077F0 to 0x77FF, and if any bytes of these is not 0xFF, it skips the remaining of the boot code and jumps to 0x0000 as a normal 8051 reset. If all bytes are in 0x077F0 to 0x77FF are 0xFF, the boot code scans the I^2C slave 0 and 1, as well as UART0 for any In-System-Programming request. This scanning takes about 10msec. If any valid request is valid during the scan, the boot-code proceeds to follow the request and performs the programming from the host. The default ISP commands available are

UNLOCK
DEVICE NAME
BOOTC VERSION
READ AND VERIFY Main Memory (8-Byte)
ERASE Main Memory exclude Boot Code
ERASE SECTOR Main Memory
WRITE BYTE Main Memory
SET ADDRESS
CONTINUOUSE WRITE
CONTINUOUS READ
READ BYTE IFB
WRITE BYTE IFB

Similar to writer mode, ISP is in default locked state. No command is accepted under locked state. To unlock the ISP, an 8-byte READVERIFY of 0x07FF8 to 0x07FFF must be successfully executed. Thus default ISP boot program provides similar code security as the Writer mode.

16. Electrical Characteristics

16.1 Absolute Maximum Ratings

SYMBOL	PARAMETER	RATING	UNIT	NOTE
VDD	Supply Voltage	5.5	V	
TA	Ambient Operating Temperature	-40 – 85	℃	
TSTG	Storage Temperature	-65 – 150	℃	

16.2 Recommended Operating Condition

SYMBOL	PARAMETER	RATING	UNIT	NOTE
VDD	Supply Voltage for IO and 1.8V regulator	2.2 – 5.5	V	
TA	Ambient Operating Temperature	-40 – 85	℃	

16.3 DC Electrical Characteristics (VDDHIO=VDDHA=3.0V to 5.5V TA= -40C to 85C)

SYMBOL	PARAMETER	MIN	TYP	MAX	UNIT	NOTE
Power Supply Current						
IDD Normal	Total IDD through VDD at 16MHz		7		mA	
IDDVF	Total IDD Core Current versus Frequency		0.5		mA/ MHz	
IDD, Stop	IDD, stop mode		150		uA	Main regulator on
IDD, Sleep	IDD, sleep mode		10		uA	Main regulator off
RSTN Reset						
VIHRS	Input High Voltage	+1.1	-	-	V	
VILRS	Input Low Voltage	-	-	0.5	V	
VRSHYS	RSTN Hysteresis	-	0.7	-	V	VDD18=1.8V
GPIO DC Characteristics						
VOH,4.5V	Output High Voltage 1 mA	-	-0.2	-0.5	V	Reference to VDD
VOH,4.5V	Output High Voltage 2 mA	-	-0.3	-0.7	V	Reference to VDD
VOL,4.5V	Output Low Voltage 4 mA	-	0.2	0.4	V	Reference to VSS
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	-0.6	V	Reference to VDD
VOH,3.0V	Output High Voltage 2 mA	-	-0.4	-0.8	V	Reference to VDD
VOL,3.0V	Output Low Voltage 4 mA	-	0.2	0.4	V	Reference to VSS
VOL,3.0V	Output Low Voltage 8 mA	-	0.3	0.6	V	Reference to VSS
VIH	Input High Voltage	2.2	-	-	V	
VIL	Input Low Voltage	-	-	1.1	V	
VIHYS	Input Hysteresis	100	300	600	mV	
RPU	Equivalent Pull-Up resistance, 3.3V		350K		Ohm	
	Equivalent Pull-Up resistance, 5.0V		200K		Ohm	
RPD	Equivalent Pull-Down Resistance, 3.3V		200K		Ohm	
	Equivalent Pull-Down Resistance, 5.0V		125K		Ohm	
REQAN	Equivalent ANIO Switch Resistance, 5V		500		Ohm	One ANIO Switch
	Equivalent ANIO Switch Resistance, 3.3V		800		Ohm	One ANIO Switch
GPIO 2 Analog Switch DC Characteristics						
RPULAT	Equivalent Pull-Up Resistance for Latch		10K		Ohm	Measured at VDDHIO
RPULAT	Equivalent Pull-Up Resistance for Latch		10K		Ohm	Measured at VDDHIO
SAR ADC						
VINSAR	Input DC Range	0	-	VDD-1.2V	V	REF=VDD, FS=VDD
		0	-	VDD18	V	REF=VDD18
LINSARM	SAR ADC Accuracy		+/- 2		LSB	After calibration

	0.5V to REF-0.5V					
FADC	ADC maximum frequency	-	-	8	MHz	VDD>= 3.0V
		-	-	1	MHz	VDD < 3.0V
TCONV	ADC conversion time	-	4	-	usec	VDD >= 3.0V
		-	32	-	usec	VDD < 3.0V
Analog Comparator						
VINCM	Input Common Mode Range	0	-	VDD-1.2	V	
VOLTAGE DAC						
VOUT	Output Range	0		¾ VDD	V	For normal accuracy
LINDAC	ADAC Accuracy		+/- 2		LSB	Normal output range
			+/- 10		LSB	0 – 0.5V
			+/- 12		LSB	¾ VDD to VDD
Low Supply (VDDHR) Voltage Detection						
VDET	Detection Range	1.8	-	5.5	V	Setting by LVDTHD
VDETHYS	Detection Hysteresis		100		mV	

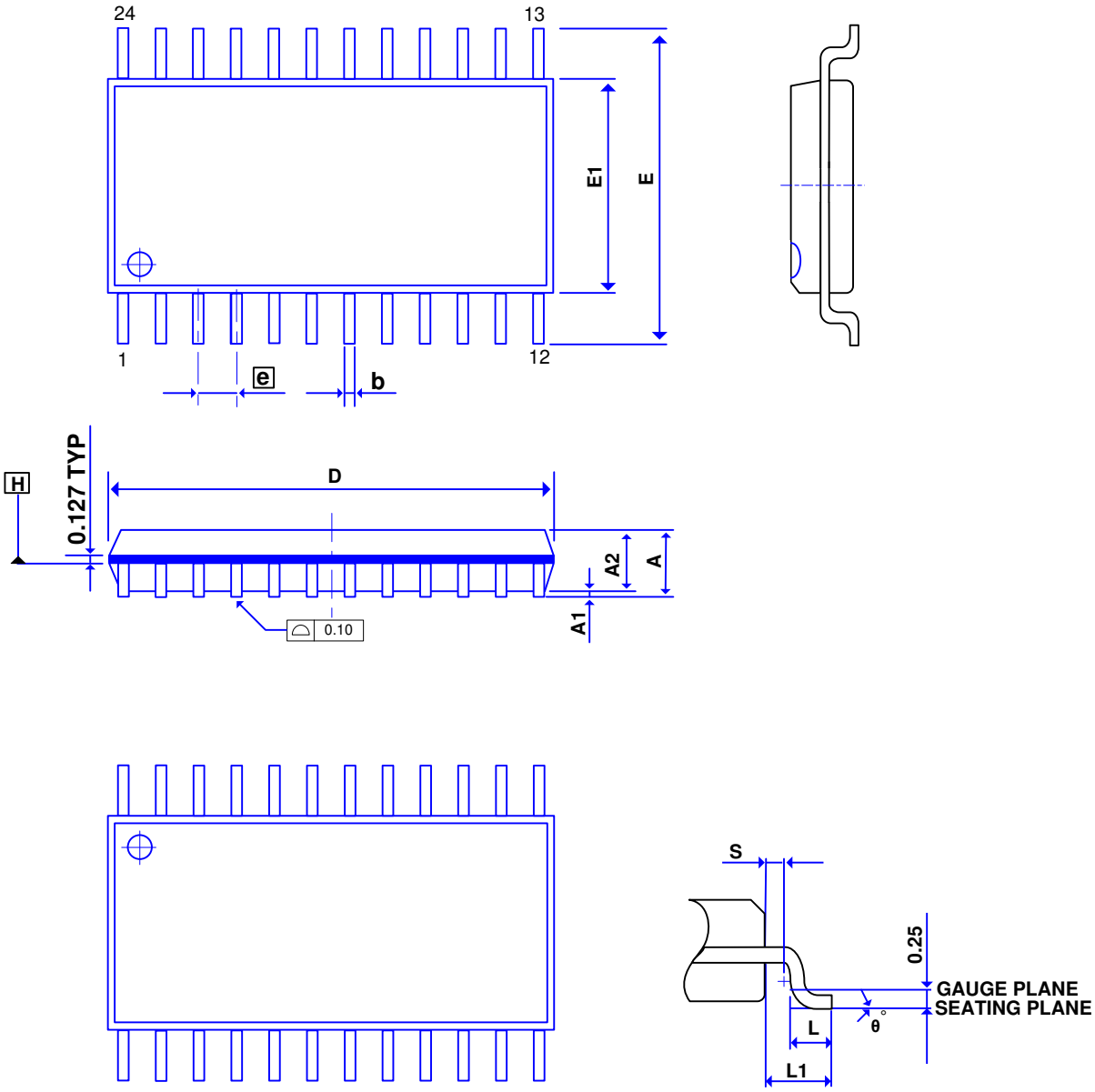
16.4 AC Electrical Characteristics (VDD =3.0V to 5.5V TA=-40C to 85C)

SYMBOL	PARAMETER	MIN	TYP	MAX	UNIT	NOTE
System Clock and Reset						
FSYS	System Clock Frequency		16	25	MHz	
FXOSC	Crystal Oscillator Frequency	1	16	25	MHz	
TSXOSC	Stable Time for XOSC after power up	50	-	-	msec	VDD > 3.0V
Supply Timing						
TSUPRU	Maximum VDD Ramp Up time	-	-	50	msec	
TSUPRD	Maximum VDD Ramp Down Time	-	-	50	msec	
TPOR	Power On Reset Delay	-	10	-	msec	IOSC=16MHz
IOSC						
FIOSC	IOSC calibrated 16MHz	-1	0	+1	%	
	Temperature and VDD variation	-2	0	+2	%	
SIOSC						
FSIOSC	Slow Oscillator frequency		32		KHz	
IO Timing						
TPD3 ++	Propagation Delay 3.3V No load	-	6	-	nsec	
TPD3 ++	Propagation Delay 3.3V 25pF load	-	15	-	nsec	
TPD3 ++	Propagation Delay 3.3V 50pF load	-	20	-	nsec	
TPD3 --	Propagation Delay 3.3V No load	-	5	-	nsec	
TPD3 --	Propagation Delay 3.3V 25pF load	-	12	-	nsec	
TPD3 --	Propagation Delay 3.3V 50pF load	-	15	-	nsec	
TPD5 ++	Propagation Delay 3.3V No load	-	5	-	nsec	
TPD5 ++	Propagation Delay 3.3V 25pF load	-	12	-	nsec	
TPD5 ++	Propagation Delay 3.3V 50pF load	-	16	-	nsec	
TPD5 --	Propagation Delay 3.3V No load	-	4	-	nsec	
TPD5 --	Propagation Delay 3.3V 25pF load	-	9	-	nsec	
TPD5 --	Propagation Delay 3.3V 50pF load	-	12	-	nsec	
Flash Memory Timing						
TEMAC	Embedded Flash Access Time	-	35	45	nsec	TWAIT must > TEMAC

TEMWR	Embedded Flash Write Time	-	20	25	usec	
TEMSER	Embedded Flash Sector Erase Time	-	2	2.5	msec	
TEMMER	Embedded Flash Mass Erase Time	-	10	12	msec	
FSPFM	SPI Flash Clock Frequency	1	64	80	MHz	
SAR ADC						
FSARADC	Maximum SAR ADC Frequency	-	-	4	MHz	
TSARADC	Conversion time of SAR ADC	-	16	-	Cycle	ADC clock cycles
TSADCSE	Set up time for SAR ADC channel select	250	-	-	nsec	
Analog Comparator						
TDACMP	Analog comparator delay	-	-	250	nsec	

17. PACKAGE OUTLINE

17.1 24-pin TSSOP

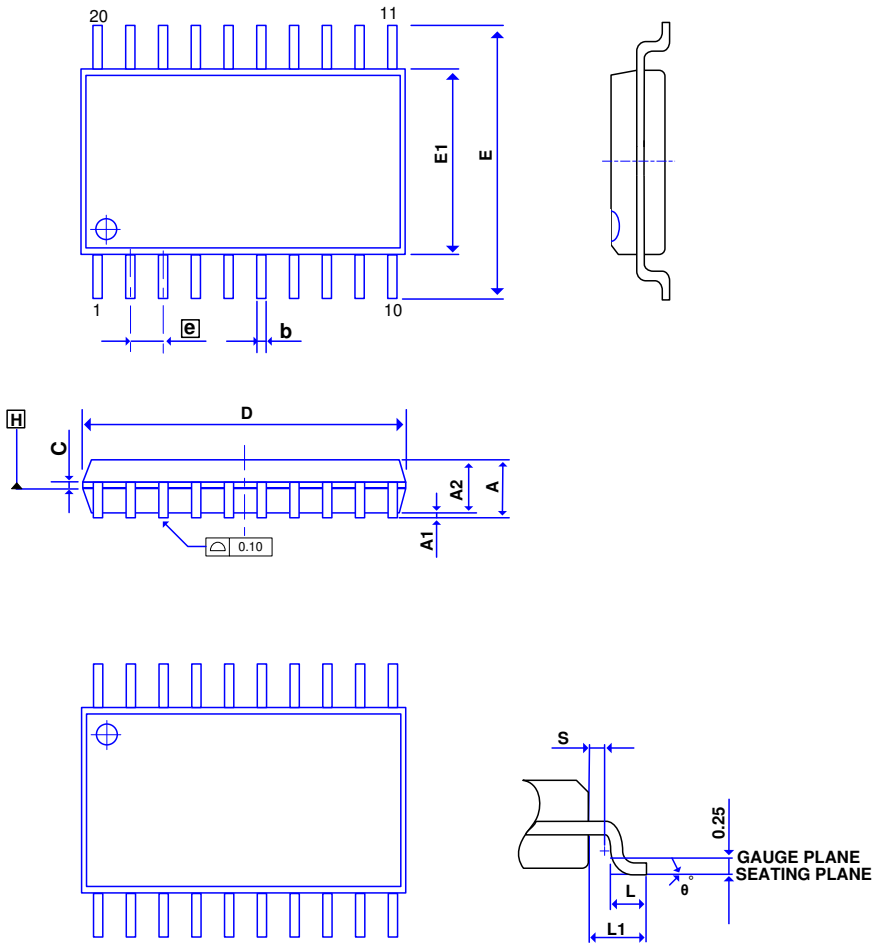


SYMBOL	DIMENSIONS IN MILLIMETERS		
	MIN.	NOM.	MAX.
A	-	-	1.20
A1	0.00	-	0.15
A2	0.80	1.00	1.05
b	0.19	-	0.30
D	7.70	7.80	7.90
E1	4.30	4.40	4.50
E	6.40 BSC		
e	0.65 BSC		
L1	1.00 REF		
L	0.45	0.60	0.75
S	0.20	-	-
θ	0	-	8

Notes:

1. *JEDEC OUTLINE:*
   MO-153 AD REV.F
2. *DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 PER SIDE.*
3. *DIMENSION "E1" DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD SHALL NOT EXCEED 0.25 PER SIDE.*
4. *DIMENSION "b" DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 MM. TOTAL IN EXCESS OF THE "b" DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD IS 0.07 MM.*
5. *DIMENSIONS "D" AND "E1" TO BE DETERMINED AT DATUM PLANE H.*

17.2 20-pin TSSOP

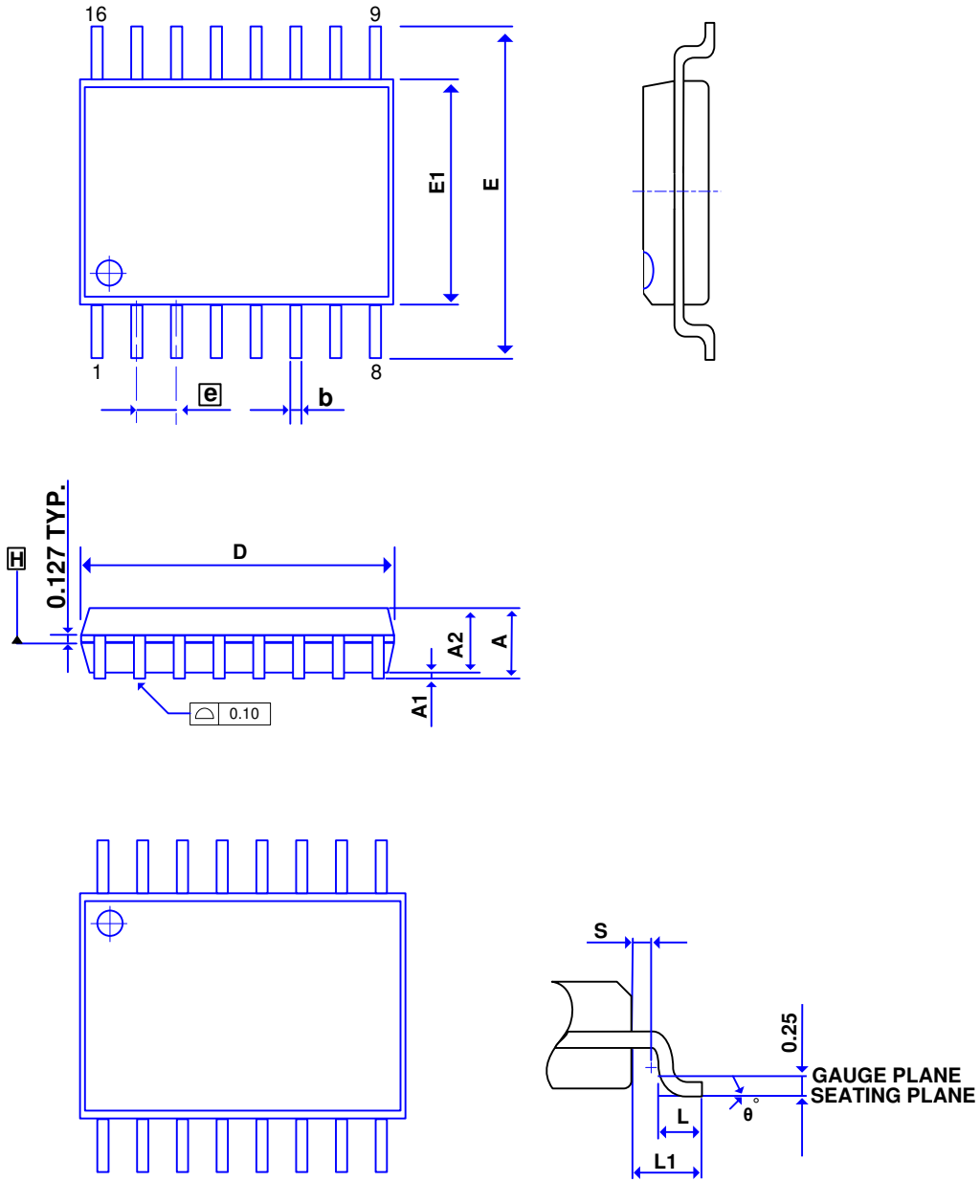


SYMBOL	DIMENSIONS IN MILLIMETERS		
	MIN.	NOM.	MAX.
A	-	-	1.20
A1	0.05	-	0.15
A2	0.80	0.90	1.05
b	0.19	-	0.30
C	0.09	-	0.20
D	6.40	6.50	6.60
E1	4.30	4.40	4.50
E	6.40 BSC		
e	0.65 BSC		
L1	1.00 REF		
L	0.50	0.60	0.75
S	0.20	-	-
θ	0	-	8

Notes:

1. *JEDEC OUTLINE:*
   MO-153 AC REV.F
2. *DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 PER SIDE.*
3. *DIMENSION "E1" DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD SHALL NOT EXCEED 0.25 PER SIDE.*
4. *DIMENSION "b" DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 MM. TOTAL IN EXCESS OF THE "b" DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD IS 0.07 MM.*
5. *DIMENSIONS "D" AND "E1" TO BE DETERMINED AT DATUM PLANE H.*

17.3 16-pin TSSOP



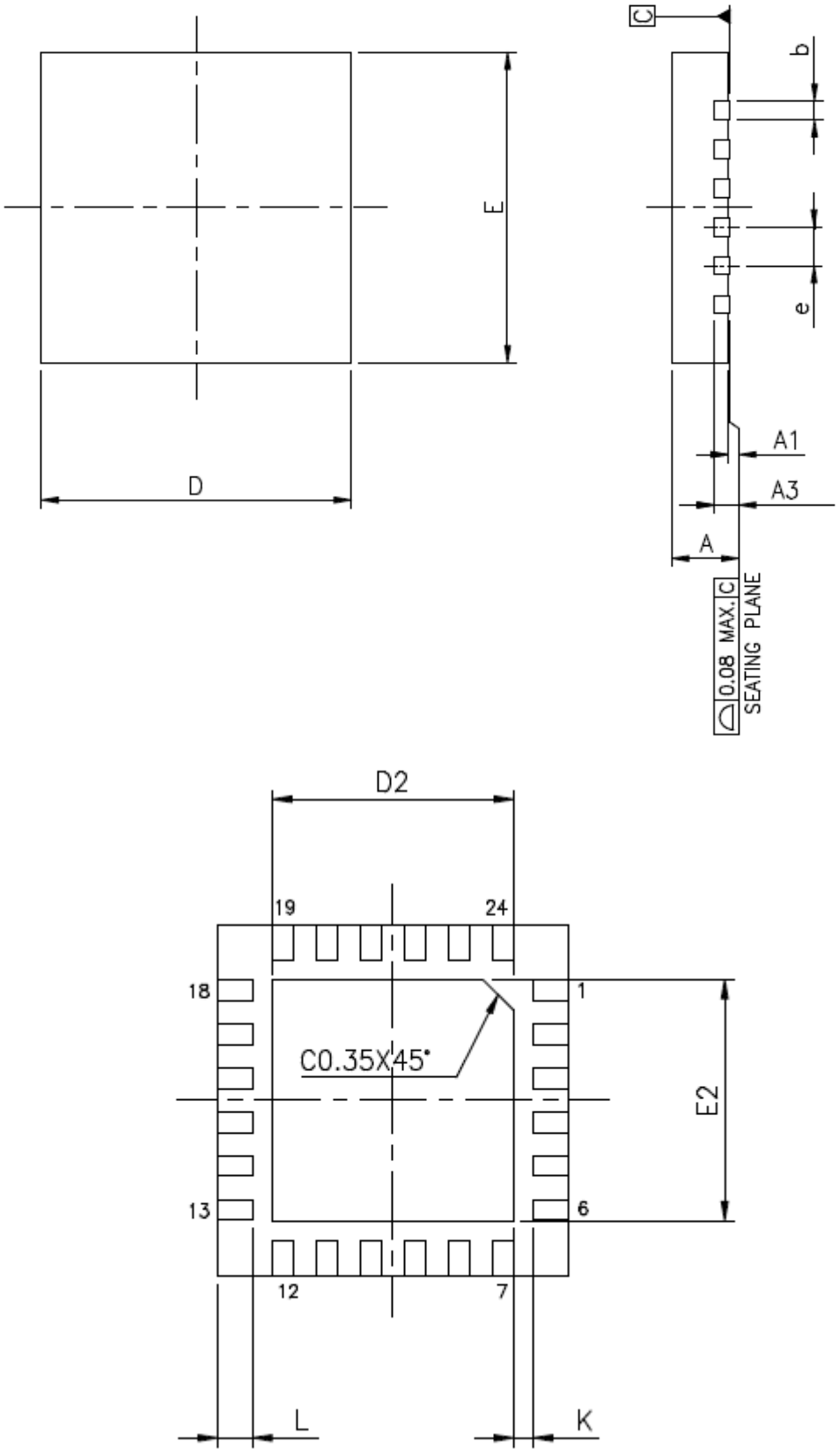
VARIATIONS (ALL DIMENSIONS SHOWN IN MM)

SYMBOL	MIN.	NOM.	MAX.
A	-	-	1.20
A1	0.00	-	0.15
A2	0.80	1.00	1.05
b	0.19	-	0.30
D	4.90	5.00	5.10
E1	4.30	4.40	4.50
E	6.40 BSC		
e	0.65 BSC		
L1	1.00 REF		
L	0.45	0.60	0.75
S	0.20	-	-
θ	0°	-	8°

Notes:

1. *JEDEC OUTLINE:*
   STANDARD: MO-153 AB REV.F
2. *DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS OR GATE BURRS SHALL NOT EXCEED 0.15 PER SIDE.*
3. *DIMENSION "E1" DOES NOT INCLUDE INTERLEAD FLASH OR PROTRUSION. INTERLEAD FLASH OR PROTRUSION SHALL NOT EXCEED 0.25 PER SIDE.*
4. *DIMENSION "b" DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 MM. TOTAL IN EXCESS OF THE "b" DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OF THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD IS 0.07 MM.*
5. *DIMENSIONS "D" AND "E1" TO BE DETERMINED AT DATUM PLANE H.*

17.4 24-pin QFN



	D2			E2			L		
PAD SIZE	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
114X114 MIL	2.60	2.70	2.75	2.60	2.70	2.75	0.35	0.40	0.45

UNIT : mm

SYMBOLS	MIN.	NOM.	MAX.
A	0.80	0.85	0.90
A1	0.00	0.02	0.05
A3		0.20 REF.	
b	0.18	0.25	0.30
D		4.00 BSC.	
E		4.00 BSC.	
e		0.50 BSC.	
K	0.20	—	—

UNIT : mm

NOTES :

1. JEDEC OUTLINE : N/A.
2. DIMENSION b APPLIES TO METALLIZED TERMINAL AND IS MEASURED BETWEEN 0.15mm AND 0.30mm FROM THE TERMINAL TIP. IF THE TERMINAL HAS THE OPTIONAL RADIUS ON THE OTHER END OF THE TERMINAL, THE DIMENSION b SHOULD NOT BE MEASURED IN THAT RADIUS AREA.
3. THE MINIMUM "K" VALUE OF 0.20mm APPLIES.
4. BILATERAL COPLANARITY ZONE APPLIES TO THE EXPOSED HEAT SINK SLUG AS WELL AS THE TERMINALS.

18. ORDERING INFORMATION

Operating temperature -40°C to 85°C

Order Part No.	Package	QTY
IS31CS8969AA-ZNLS2	TSSOP-16, Lead-free	96/Tube
IS31CS8969BA-ZNLS2	TSSOP-20, Lead-free	75/Tube
IS31CS8969CA-ZNLS2	TSSOP-24, Lead-free	60/Tube
IS31CS8969W-QFLS2	QFN-24, Lead-free	490/Plate

19. Revision History

Revision A

Formal mass production release version

V1.25

Change to ISSI naming rule and ordering information

V1.24

Correct ADCIF and ADCINTE in Interrupt Diagram at page 25
ADCIF is a read-only bit
ADCPGA (0xB9h) is write-only SFR
I2CSST2[0] (NACK) cannot be cleared by S/W
PWMFG is located at 0XA08C

V1.23

Correct the typos in FEATURES, Flash Controller CYC[2-0] table, and Electrical Characteristics.
Revise TCON.

V1.22

Revise ADCCTL[0] with CSTART
Update the LVDTH formula
Revise I2C Master SFR. I2CMCR[7] is I2CMRST and I2CMCR[6] is INFILEN

V1.21

Correct CLKSEL default value
Remove SPI SSN function from P1.7
Remove the STATUS (0xC5) description from section 1.2 and merge it into section 1.7

V1.2

Update 3.2 1.0V Reference (BGREF)
Revise 3.5 Supply Low Voltage Detection (LVD), LVD range = 1.8V ~ 5.5V
Add 11.1 On-Chip Temperature Sensor

V1.1 for B0

Add IS31CS8969W with QFN-24 package

V1.0 for B0

Revise the PIN definition included P1.7, P2.0, P2.1, P2.2, and P2.3
Revise Register Map SFR and Register Map XFR
Add LVDHYS at XFR 0xA016 and I2CSADR2
Add EUART_OPL at LININTEN[6]
Update IO Configuration and Multifunction Configuration
Correct some errors in painting.

V0.90

Modify IFB Block description
For Engineering Sample Release

V0.13

WDCON default value is 0x00 not 0x02
Revise IOSCDIV table
Timer0 and Timer2 don't support external input T0 and T2EX function.
MFCFGP1.4[7] is T1EN not KEY12EN , MFCFGP1.5[7] is T2EN not KEY13EN

V0.12

Add VDAC functions.

V0.11

Modified the pin connections and pin descriptions

V0.10

Inherit from CS5523 V0.23.

Add ADDR0DC control in I2CS2.