SS1102C
Integrated MCU with Spread-Spectrum Transceiver (SST)

External Specification

PRELIMINARY (V 1.8)

Siliconians, Inc.
4701 Patrick Henry Drive, Suite 501, Santa Clara, CA 95054
Tel: 408-748-8600 Fax: 408-748-8687
www.siliconians.com or info@siliconians.com

Table of Contents

1. Description

The SS1102C is a CMOS integrated solution for direct sequence spread spectrum digital wireless data communication applications. The chip performs all CPU, baseband modem and peripheral control functions. A baseband spread spectrum modem, a microcontroller, and I/O interfaces are integrated into a single chip. The RF interface is suitable for most RF module designs. A low speed link data path allows for supervisory and setup functions, while the communication channel is used for full duplex data transmission.

The chip can be delivered in the 52-pin or 100-pin package. The 100-pin version has an interface to the external EPROM. The 52-pin version is a mask version of the chip.

2. Features

- Direct sequence spread spectrum transceiver
- Support for MSK
- RF interface signals
- Full and half duplex data transmission modes
- 125Kbps maximum data rate in half duplex mode
- 64Kbps maximum data rate in full duplex mode
- Low speed signalling full duplex data path
- 8051 compatible 8-bit CPU
- 16K bytes on-chip ROM
- 512 x 8 RAM
- Two capture timers
- WatchDog timer
- Time Base timer
- Programmable Serial Peripheral Interface (SPI)
- Low battery detect
- Programmable I/O lines
- Active, Low Power and Power Down Operation modes
- Two on-chip oscillator circuits - up to 24MHz Master clock and low speed 32KHz clock for the Low Power mode
- Power-on Reset
- Supply Voltage: 2.7V-5.5V
- 52-pin package
- Operating temperature: -20C to 85C

3. SS1102C Block Diagram

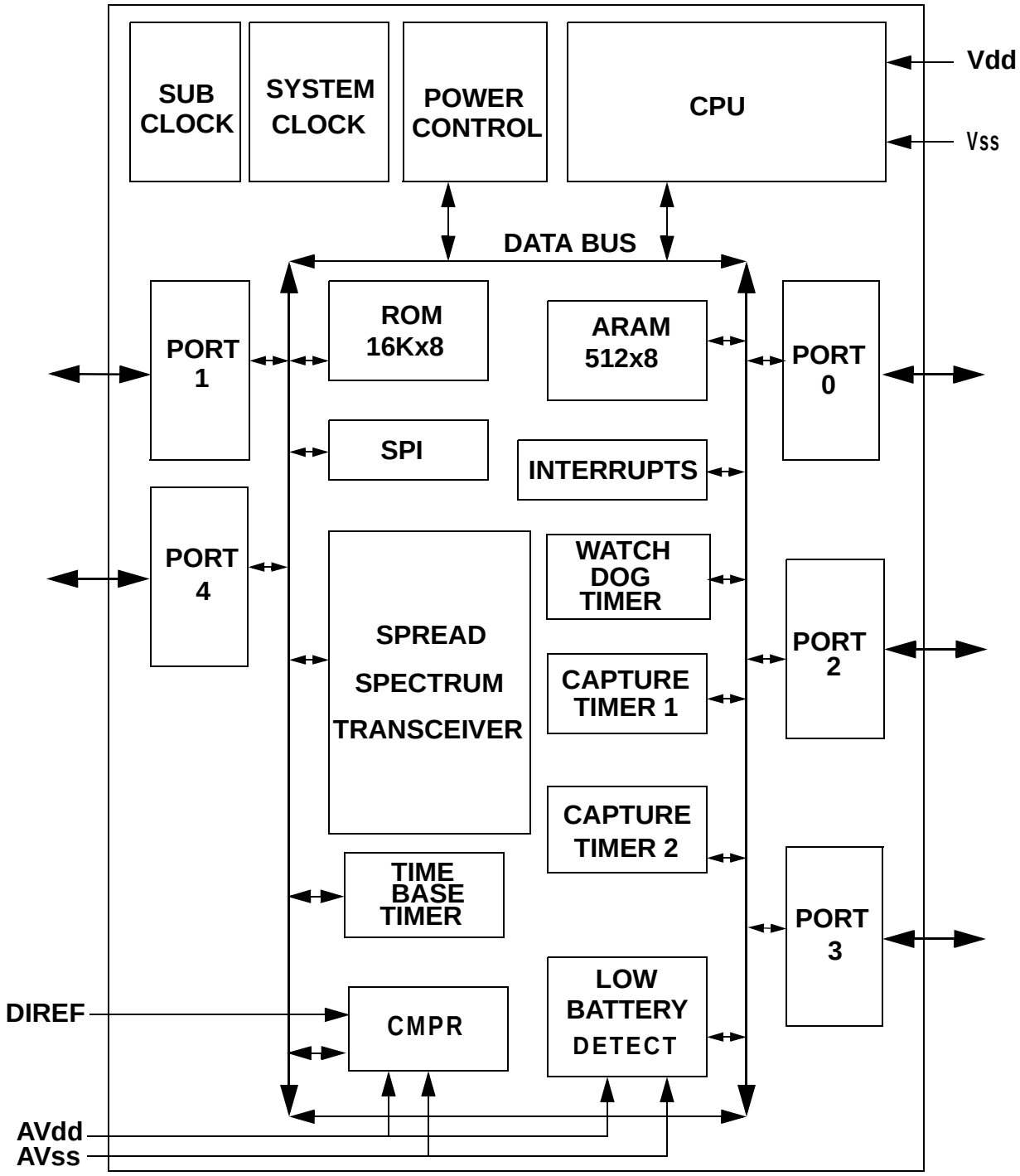


Figure 1. *SS1102C Block Diagram*

4. Pin Description

Port	Pin #	Pin Name	I/O Type	Functions
VDD	1	Vdd		Digital Supply Voltage
NRESET	2	NRESET		Reset
TST	3	TST		Test pin
BXFOUT	4	BXFOUT		Buffered High Frequency Output
XFOUT	5	XFOUT	Oscillator	High frequency crystal output
XFIN	6	XFIN	Oscillator	High frequency crystal input
XSOUT	7	XSOUT	Oscillator	Slow frequency crystal output
XSIN	8	XSIN	Oscillator	Slow frequency crystal input
P4.0	9	RFPWR	I/O 3	RF power switch output/ I/O
P4.1	10	PLLSW	I/O 3	Phase-lock loop switch output/ I/O
P4.2	11	LOCK	I/O 3	Lock Indicator
P4.3	12	I/O	I/O 3	Latched bi-directional I/O
P4.4	13	I/O	I/O 3	Latched bi-directional I/O
P4.5	14	I/O	I/O 3	Latched bi-directional I/O
P4.6	15	I/O	I/O 3	Latched bi-directional I/O
P4.7	16	I/O	I/O 3	Latched bi-directional I/O
P3.0	17	CPTRU2	I/O 3	Capture Timer 2
P3.1	18	CPTRU1	I/O 3	Capture Timer 1
VSS	19	Vss		Digital Ground
P3.2	20	CPTRD1	I/O 3	Capture Timer 1/Down Counter
P3.3	21	TXEN	I/O 3	Transmitter enable output/ I/O
P3.4	22	MODOUT	I/O 3	Modulated output (chips output)
P3.5	23	DI	I/O 2	Digital (DI) / Analog (DI1) Data Input
P3.6	24	IRQ1	I/O 3	External Interrupt 0
P3.7	25	IRQ2	I/O 3	External Interrupt 1
DIREF	26	DIREF	I/O 3	Analog Data Input Comparator External reference voltage

Port	Pin #	Pin Name	I/O Type	Functions
P2.0	27	I/O	I/O 2	Latched bi-directional I/O
P2.1	28	I/O	I/O 2	Latched bi-directional I/O
P2.2	29	I/O	I/O 2	Latched bi-directional I/O
P2.3	30	I/O	I/O 2	Latched bi-directional I/O
P2.4	31	I/O	I/O 2	Latched bi-directional I/O
P2.5	32	I/O	I/O 2	Latched bi-directional I/O
P2.6	33	I/O	I/O 2	Latched bi-directional I/O
P2.7	34	I/O	I/O 2	Latched bi-directional I/O
AVSS	35	AVss		Analog Ground
AVDD	36	AVdd		Analog Supply Voltage
P0.0	37	I/O	I/O 2	Latched bi-directional I/O
P0.1	38	I/O	I/O 2	Latched bi-directional I/O
P0.2	39	I/O	I/O 2	Latched bi-directional I/O
P0.3	40	I/O	I/O 2	Latched bi-directional I/O
P0.4	41	I/O	I/O 2	Latched bi-directional I/O
P0.5	42	I/O	I/O 2	Latched bi-directional I/O
P0.6	43	I/O	I/O 2	Latched bi-directional I/O
P0.7	44	I/O	I/O 2	Latched bi-directional I/O
P1.0	45	I/O	I/O 2	Latched bi-directional I/O
P1.1	46	I/O	I/O 2	Latched bi-directional I/O
P1.2	47	I/O	I/O 2	Latched bi-directional I/O
P1.3	48	I/O	I/O 2	Latched bi-directional I/O
P1.4	49	LBD	I/O 2	Low battery detect
P1.5	50	SPIO	I/O 2	Serial Out
P1.6	51	SPIIN	I/O 2	Serial In
P1.7	52	SPICLK	I/O 2	Serial Clock

TABLE 1. *Pin Description*

52-PIN PQFP

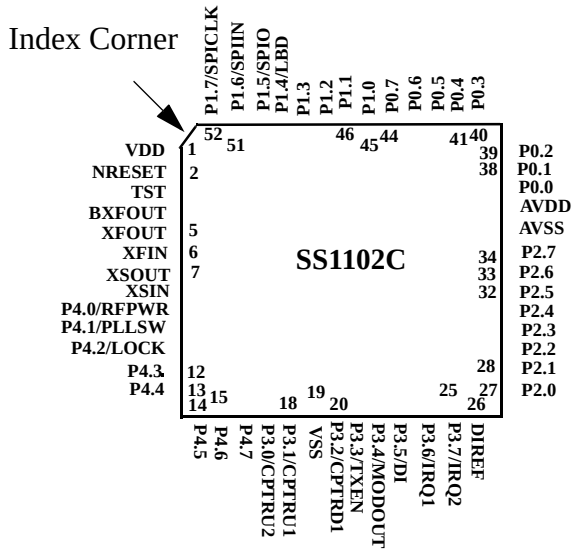


Figure 2. *SS1102C Pinout*

5. CPU

- High performance CMOS 8-bit CPU with the industry standard 80C51 instruction set.
- Extensive boolean processing (single bit logic) capabilities
- Arithmetic: 8 bit including multiply and divide
- Jumps, 8/16 bit address, conditional and unconditional
- Logical separation of program and data memory
- Six addressing modes
- Maximum operating speed 24MHz.

6. Memory organization

6.1. Data memory

The SS1102C has separate address space for Program Memory and Data Memory.

The SS1102C has 512 bytes of onchip data space, 256 bytes must be accessed by MOVX.

By direct and indirect addressing the lowest 128 bytes can be accessed. Then the next 128 bytes are accessed by indirect addressing while direct addressing in this area will access the SFR registers. With the MOVX command there are up to 256 bytes of data memory accessible. DPH should be “0” to use Port.0 and Port2 as GPIO ports.

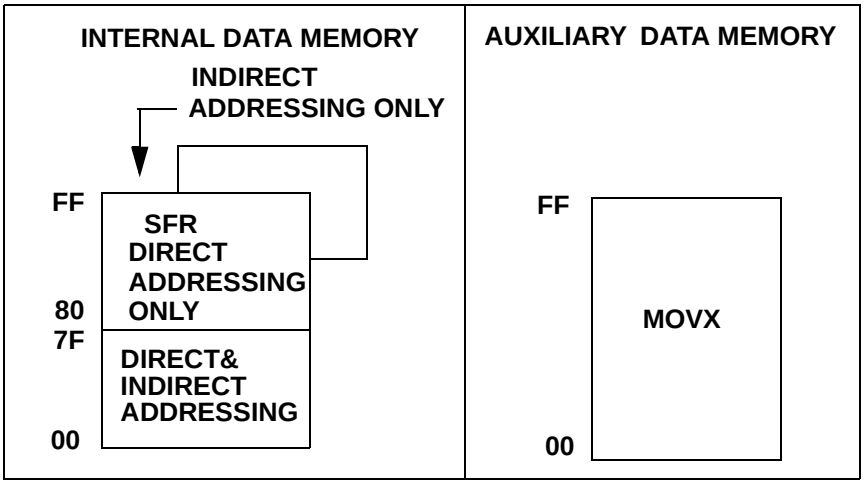


Figure 3. *The SS1102C Data Memory*

6.2. Program memory

The SS1102C program memory will fetch 16Kbytes from internal mask ROM.

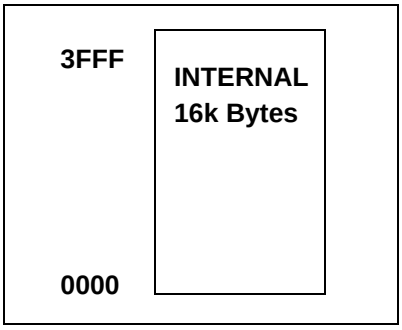


Figure 4. *The SS1102C Program Memory*

7. Special function registers

The SS1102C special function registers (SFR) are accessed with direct memory addressing in the memory space from 80H to FFH. The table below shows the location of each register. The description of the register functions are in the respective peripheral block descriptions and the architectural overview section.

Lo \ Hi	8	9	A	B	C	D	E	F
0	P0	P1	P2	P3	P4	PSW	ACC	B
1	SP		WDCR	TBCR	SSTMCR	CT1CR	CT2CR	SPI1CR
2	DPL		WDCNT	TBCNT		CT1CNTL	CT2CNTL	SPI1DAT
3	DPH			TBDAT		CT1CNTH	CT2CNTH	SPI1SR
4	LBDCR					CT1DATL	CT2DATL	
5	CMPRCR					CT1DATH	CT2DATH	
6		SREL	XICR0			CT1SR		
7	-----	PSCR				ISE0	ISE1	ISE2
8	-----	-----	IE	IP	0:IEIS 1:UW2	INT0	INT1	INT2
9	-----	-----	P0NOP	P1NOP	P2NOP	P3NOP	P4NOP	
A	-----	PCR	P0IO	P1IO	P2IO	P3IO	P4IO	
B	-----	ZCR	P0RD	P1RD	P2RD	P3RD	P4RD	
C	-----							
D	-----							
E	0: CIL 1: CIL	0:LCK 1:TXSHH	0:TXSW/ RXSW 1:RXSHH	0:PNA0 1:PNC0	0:PNA1 1:PNC1	0:PNA2 1:PNC2	0:PNA3 1:PNC3	0:RDI 1:UW1
F	0: CIH 1: UW0	0: - 1:TXSHL	0:SN 1:RXSHL	0:PNB0 1:PND0	0:PNB1 1:PND1	0:PNB2 1:PND2	0:PNB3 1:PND3	MSIZ

TABLE 2. *Special Function Registers Description*

The SFR register definitions are described within each peripheral block description and the general CPU registers SP, DPTR, PSW, ACC, B, and MSIZ are described below. The underlined names are the registers for the CPU and the dashed lines are for registers used in standard 8051 devices but not used by the SS1102C microcontroller.

7.1. Stack Pointer (SP)

The SP register contains the stack pointer. The stack pointer is used to load the program counter into memory during LCALL and ACALL instructions, and is used to retrieve the program counter from memory in RET and RETI instructions. The stack may also be saved or loaded using PUSH and POP instructions, which also increment and decrement the stack pointer. The stack pointer points to the top location of the stack. On reset the stack pointer is set to 07 hex.

7.2. Data Pointer (DPTR)

The Data Pointer (DPTR) is 16 bits in size, and consists of two registers, the Data Pointer High byte (DPH), and the Data Pointer Low byte (DPL). Two 16 bit operations are possible on this register, they are load immediate and increment. This register is used for 16 bit address external memory accesses, for offset code byte fetches, and for offset program jumps. On reset the value of this register is 0000 hex.

7.3. Program Status Word (PSW)

This register contains status information resulting from CPU and ALU operation. The bit definitions are given below:

- **PSW.7** CY. ALU carry flag.
- **PSW.6** AC. ALU auxiliary carry flag.
- **PSW.5** F0. General purpose user definable flag.
- **PSW.4** RS1. Register bank select bit 1.
- **PSW.3** RS0. Register bank select bit 0.
- **PSW.2** OV. ALU overflow flag.
- **PSW.1** F1. User definable flag.
- **PSW.0** P. Parity flag. Set each instruction cycle to indicate odd/even parity in the accumulator.

On reset this register returns 00 hex.

The register bank select bits operate as follows

.

RS1	RS0	Register Bank Select
0	0	RB0. Registers from 00 - 07 hex.
0	1	RB1. Registers from 08 - 0F hex.
1	0	RB2. Registers from 10 - 17 hex.
1	1	RB3. Registers from 18 - 1F hex.

TABLE 3. *Register bank select bits*

7.4. Accumulator (ACC)

This register provides one of the operands for most ALU operations. In the instruction table it is denoted as “A”.

On reset this register returns 00 hex.

7.5. B Register (B)

This register provides the second operand for multiply or divide instructions. Otherwise it may be used as a scratch pad register.

On reset this register returns 00 hex.

7.6. Memory Size Register (MSIZ)

The purpose of this register is to define the amount of available internal program memory. The amount available is:

(MSIZ + 1) $\times$ 256.

On reset this register returns 3F hex or 16K of internal program memory.

- 16K byte internal ROM
- A total of 512 bytes of on-chip data RAM:
  - 256 bytes standard RAM
  - 256 bytes of additional on-chip data memory accessible by the MOVX command (XRAM).

7.7. Interrupts

- Two external interrupt input pins (IRQ1, IRQ2)

- Eleven internal interrupt sources. The following peripheral blocks can generate interrupt request: Watchdog Timer, Time Base Timer, Capture Timer 1, Capture Timer 2, SPI, TXFIFO, RXFIFO, Signaling Word, Signal/Noise Ratio, and Lock, LBD

8. Time Base Timer

8.1. Overview

The Time Base Timer (TB-Timer) is an 8-bit auto-reload timer. The TB-Timer is composed of a input frequency select MUX, an 8-bit up-counter(TBCNT), a Comparator, an 8-bit data register(TBDAT), and a control register(TBCR).

The TB-Timer has the 8 counting clocks that can be selected by TBFS[5:3]. Four of the clocks come from the System Clock block, i.e. F_{1SEC}, F_{1MIN}, F_{1HOUR}, and F_{125mS}. If the slow oscillator is stopped, these four clocks will stop too. These four clocks will be very useful to generate a real time interval and to minimize the number of CPU wake-ups. The period of the $F_{sys}/12$ clock is one machine cycle or one fastest instruction cycle. This clock will run during any power-saving mode except the StopAll mode. During the FastAll or FastPeri mode, the $F_{sys}/12$ clock is equal to $F_{fast}/12$. During the SlowAll or SlowPeri mode, the $F_{sys}/12$ clock is equal to $F_{slow}/12$. The $F_{fast}/2^4$, $F_{fast}/2^7$, and $F_{fast}/2^{10}$ can be used during the FastAll or FastPeri mode only.

After the TBEN bit is set to high, the TBCNT will start to count the negative edge of an input clock. When the TBEN bit is low, the Match signal is never generated even though the contents of the TBDAT and TBCNT are the same. The TBCNT is the 8-bit up-counter that can be cleared by the TBCLR bit or the Match signal. The TBCNT is a modulo-N counter (from 0 to N-1), N is the content of the TBDAT. The match signal will always set the TBINT bit to high. The TBINT bit can be cleared by the TB-Timer Interrupt Acknowledge or by software.

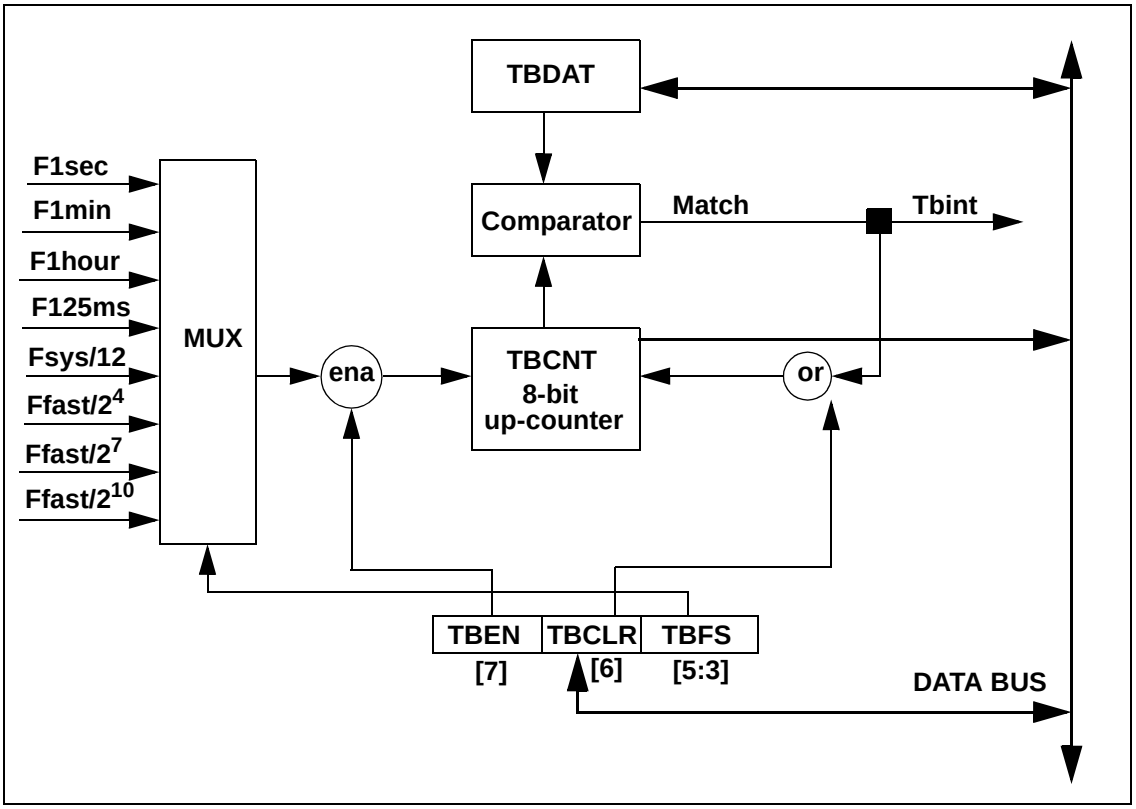


Figure 5. *Time Base Timer*

8.2. Time Base Timer Control Register (TBCR)

Address	B1H							
Bit Addr.								
BIT	7	6	5	4	3	2	1	0
NAME	TBEN	TBCLR	TBFS2	TBFS1	TBFS0	HTST	MTST	
Definition	TB timer ENable	TB counter CLeaR	TB timer input Frequency Select bit-2	TB timer input Frequency Select bit-1	TB timer input Frequency Select bit-0	Hour Clock Test	Minute Clock Test	Reserved
Reset Value	0	0	0	0	0	0	0	0

Read/ Write by Software	R/W	R/W Always 0 read.	R/W	R/W	R/W	R/W	R/W	R
Write by Hardware								

TABLE 4. *Definition of TBCR*

Name	Bit	Description
TBEN	7	TB timer ENable bit [Bit Status: 0 (Initial Value)] A selected input clock is halted. The TBCNT keeps the counter value. [Bit Status: 1] A selected input clock runs. The TBCNT counts up the negative edge of the selected clock.
TBCLR	6	TB counter CLeaR bit If this bit is written with high, the TBCNT (8-bit up-counter) of the TB Timer will be cleared to 00H. Writing low will not affect anything. When read, a low(0) will be always read.
TBFS[5:3]	5 to 3	TB timer input clock Frequency Select bits [Bit Status: 000 (Initial Value)] F_{1SEC} [Bit Status: 001] F_{1MIN} [Bit Status: 010] F_{1HOUR} [Bit Status: 011] F_{125mS} [Bit Status: 100] $F_{sys}/12$ [Bit Status: 101] $F_{fast}/2^4$ [Bit Status: 110] $F_{fast}/2^7$ [Bit Status: 111] $F_{fast}/2^{10}$
HTST	2	Hour clock Test bit [Bit Status: 0 (Initial Value)] Normal hour clock signal is used for F_{1hour}. [Bit Status: 1] A test mode using XSout (32.768KHz) for F_{1hour}. Note: The bit can only be set either under FastAll or FastPeri modes. User should write a 0 to this bit.
MTST	1	Minute clock Test bit [Bit Status: 0 (Initial Value)] Normal minute clock signal is used for F_{1min}. [Bit Status: 1] A test mode using the XSout (32.768KHz) for F_{1min} . Note: The bit can only be set either under FastAll or FastPeri modes. User should write a 0 to this bit.
TBINT	0	Reserved bits

TABLE 5. *Description of TBCR*

8.3. TB-Timer Counter Register (TBCNT)

Address	B2H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	TBCNT[7:0]							
Definition	TB-timer CouNTer register bit 7 to 0 (Modulo N Up-counter: 00 to N-1, N is the content of TBDAT)							
Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R	R	R	R	R	R	R	R
Write by Hardware	Written with a new count value 0 or 1. 0 by TBCLR.	Written with a new count value 0 or 1. 0 by TBCLR.	Written with a new count value 0 or 1. 0 by TBCLR.	Written with a new count value 0 or 1. 0 by TBCLR.	Written with a new count value 0 or 1. 0 by TBCLR.	Written with a new count value 0 or 1. 0 by TBCLR.	Written with a new count value 0 or 1. 0 by TBCLR.	Written with a new count value 0 or 1. 0 by TBCLR.

TABLE 6. *Definition of TBCNT*

Name	Bit	Description
TBCNT[7:0]	7 to 0	TB-timer CouNTer bits A Modulo-N (00 to N-1) up-counter. N is the content of TBDAT. This counter counts up from 00H to N-1. Then, the counter will go back to 00H and the comparator will generate a match signal. The counter will not stop after the Match. Thus, the TB-Timer will generate continuous Match signals with a fixed period. The TB-Timer is an auto-reload timer.

TABLE 7. *Description of TBCNT*

8.4. TB-Timer Data Register (TBDAT)

Address	B3H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	TBDAT[7:0]							
Definition	TB-timer DATa register bit 7 to 0							
Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware	none	none	none	none	none	none	none	none

TABLE 8. *Definition of TBDAT*

Name	Bit	Description
TBDAT[7:0]	7 to 0	TB-timer DATa bits [The first Match period] TB-Timer Match signal period = (1 / $F_{selected\text{-}clock}$) X (TBDAT + 1) - Deviation period 0 <= Deviation period < (1 / $F_{selected\text{-}clock}$) If any clock, which are divided from XSout(32.768KHz), is selected, the Deviation period can be minimized because the Slow Clock prescaler can be cleared by software. [The second or later Match period] TB-Timer Match signal period = (1 / $F_{selected\text{-}clock}$) X TBDAT (No Deviation period)

TABLE 9. *Description of TBDAT*

9. Capture Timers 1/2

9.1. Overview

There are two 16-bit Capture Timer in the MCU, C-Timer1 and C-Timer2. Capture-Timer1 (or C-Timer1) can be used as an Auto-Reload Timer, an Event Counter, an Up Down-Counter or a Capture Timer. Capture-Timer2 (or C-Timer2) can be used as an Auto-Reload Timer, an Event Counter or a Capture Timer. A C-Timer is composed of a input frequency select MUX, a 16-bit up down-counter(16-bit up-counter for C-Timer2), a Comparator, a 16-bit data register(CT1DATH-CT1DATL, CT2DATH-CT2DATL), an Edge Detector for Capture, and a control register(CT1CR, CT2CR).

C-Timer1 can operate in three different modes. They are Auto-Reload Mode, Up Down-Count Mode and Capture Mode. All of these modes can be selected by the CT1MD[2:0] bits in the CT1CR. On the other hand, C-Timer2 can only operate in two different modes. They are Auto-Reload Mode and Capture Mode. All of these modes can be selected by the CT2MD[1:0] bits in the CT2CR.

In Auto-Reload Mode, the C-Timer can be used as an Auto-Reload Timer and an Auto-Reload Event Counter. In Up Down-Count Mode, C-Timer1 can be used as an Up Down-Counter. In Capture mode, the C-Timer can be used as a Capture Timer.

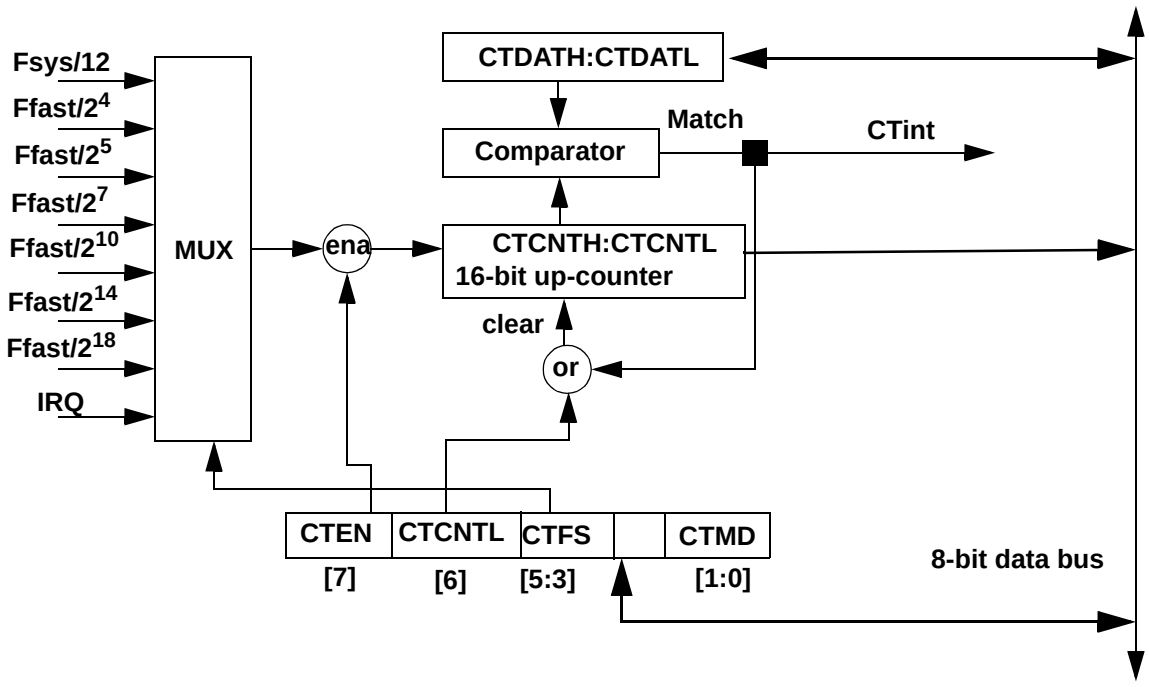


Figure 6. *Capture Timer in Auto-Reload Mode*

9.2. Auto-Reload Mode

If the CT1MD[2:0] bits in the CT1CR are 000, the C-Timer1 will operate in the Auto-Reload Mode.

If the CT2MD[1:0] bits in the CT2CR are 00, the C-Timer2 will operate in the Auto-Reload Mode.

A C-Timer has 8 counting clocks that can be selected by CT1FS[2:0] (or CT2FS[2:0]). The period of the $F_{sys}/12$ clock is one machine cycle or one fastest instruction cycle. This clock will be run during any power-saving mode except the StopAll mode. During the FastAll or FastPeri mode, the $F_{sys}/12$ clock is equal to $F_{fast}/12$. During the SlowAll or SlowPeri mode, the $F_{sys}/12$ clock is equal to $F_{slow}/12$. The $F_{fast}/2^4$, $F_{fast}/2^5$, $F_{fast}/2^7$, $F_{fast}/2^{10}$, $F_{fast}/2^{14}$, and $F_{fast}/2^{18}$ can be used during the FastAll or FastPeri mode only. Two external interrupt pins(CPTRU1 and CPTRU2) are connected to the C-Timer1 and 2. Thus, the timers can be used as the Event Counters. To use this timer as an Event Counter, the CPTRU1 (or CPTRU2) should be assigned as an input port by the port control register.

After the CT1EN(or CT2EN) bit is set to high, the CT1CNTH & CT1CNTL (or CT2CNTH & CT2CNTL) will start to count up the negative edge of a selected input clock. When the CT1EN (or CT2EN) bit is low, the Match signal is never generated even though the contents of the CT1DATH & CT1DATL (or CT2DATH & CT2DATL) and CT1CNTH & CT1CNTL (or CT2CNTH & CT2CNTL) are the same. In Auto-Reload Mode, CT1CNTH & CT1CNTL (or CT2CNTH & CT2CNTL) act as a 16-bit up-counter that can be cleared by the CT1CNTLR (or CT2CNTLR) bit or by the Match signal. The CT1CNTH & CT1CNTL (or CT2CNTH & CT2CNTL) is a modulo-N counter (from 0 to N-1), N is the content of the CT1DATH & CT1DATL (or CT2DATH & CT2DATL). The match signal will always set the CT1INT (or CT2INT) bit to high. The CT1INT (or CT2INT) bit can be cleared by a C-Timer Interrupt Acknowledge or software.

9.3. Up Down-Count Mode

If the CT1MD[2:0] bits in the CT1CR are 1XX (X means don’t care), the C-Timer1 will operate in the Up Down-Count Mode.

After the CT1EN bit is set to high, the 16-bit up down-counter(CT1CNTH & CT1CNTL) will start to count. It counts up on the negative edge of CPTRU1 and counts down on the negative edge of CPTRD1. To use CPTRU1, CT1FS[5:3] must set to 111. In Up Down-Count Mode, CT1CNTH & CT1CNTL act as a 16-bit up down-counter that can be cleared by the CT1CNTLR bit. The CT1CNTH & CT1CNTL is a modulo-65536 counter (from 0 to 65535). The overflow signal will set the CT1OV bit in CT1SR to high and the underflow signal will set the CT1UN bit in CT1SR to high. Both the overflow and underflow signal will always set the CT1INT bit in INT2 register and CT1OVUN bit in CT1SR to high. The CT1INT bit can be cleared by a C-Timer Interrupt Acknowledge or software, and CT1OV, CT1UN and CT1OVUN can be cleared by reading the CT1SR. Another Capture-Timer1 interrupt will not be generated if CT1OVUN is still not being cleared.

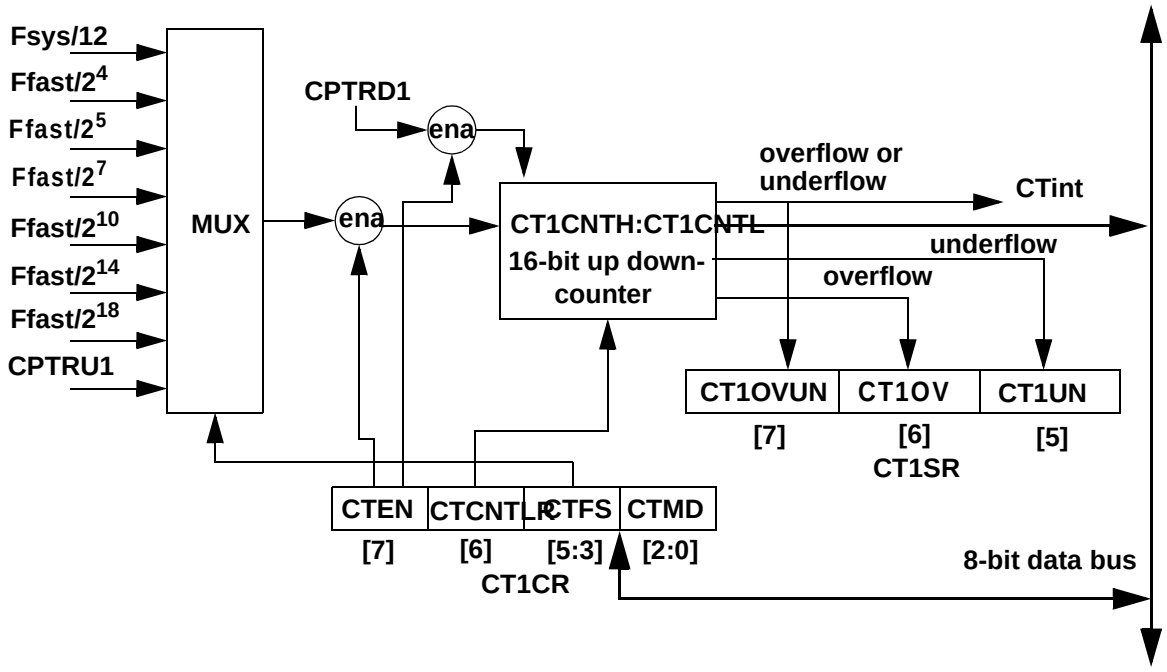


Figure 7. *Capture Timer 1 in Up-Down Count Mode*

9.4. Capture Mode

If the CT1MD[2:0] bits in the CT1CR are 001, 010, or 011, the C-Timer1 will operate in the Capture Mode. Counting clock selection feature is the same with the Auto-Reload Mode.

If the CT2MD[1:0] bits in the CT2CR are 01, 10, or 11, the C-Timer2 will operate in the Capture Mode. Counting clock selection feature is the same with the Auto-Reload Mode.

After the CT1EN (or CT2EN) bit is set to high, the 16-bit counter (CT1CNTH & CT1CNTL or CT2CNTH & CT2CNTL) will start to count up the negative edge of a selected input clock. In Capture Mode, CT1CNTH & CT1CNTL (or CT2CNTH & CT2CNTL) act as a 16-bit up-counter that can be cleared by the CT1CNTLR (or CT2CNTLR) bit. The CT1CNTH & CT1CNTL (or CT2CNTH & CT2CNTL) is a modulo-65536 counter (from 0 to 65535). The overflow signal will always set the CT1INT (or CT2INT) bit and CT1OVUN bit in CT1SR to high. The overflow signal also set the CT1OV to high. The CT1INT (or CT2INT) bit can be cleared by a C-Timer Interrupt Acknowledge or software, and CT1OV and CT1OVUN can be cleared by reading the CT1SR. Another Capture-Timer1 interrupt will not be generated if CT1OVUN is still not being cleared.

During up-counting, the content of 16-bit counter (CT1CNTH & CT1CNTL or CT2CNTH & CT2CNTL) will be capture into the 16-bit latch (CT1DATH & CT1DATL or CT2DATH & CT2DATL) when an event of CPTRU1 (or CPTRU2) is detected. One of three event (positive edge, negative edge, or any edge) can be selected by the CT1MD[1:0] bits (or CT2MD[1:0]) in the CT1CR (or CT2CR). To use this feature, the CPTRU1 pin (or CPTRU2 pin) should be assigned an input port by the port control register. Using the Capture Mode, external clock period or pulse width (high width or low width) can be measured.

Note: While in Capture mode, the CTDATH and CTDATL are still writable by S/W.

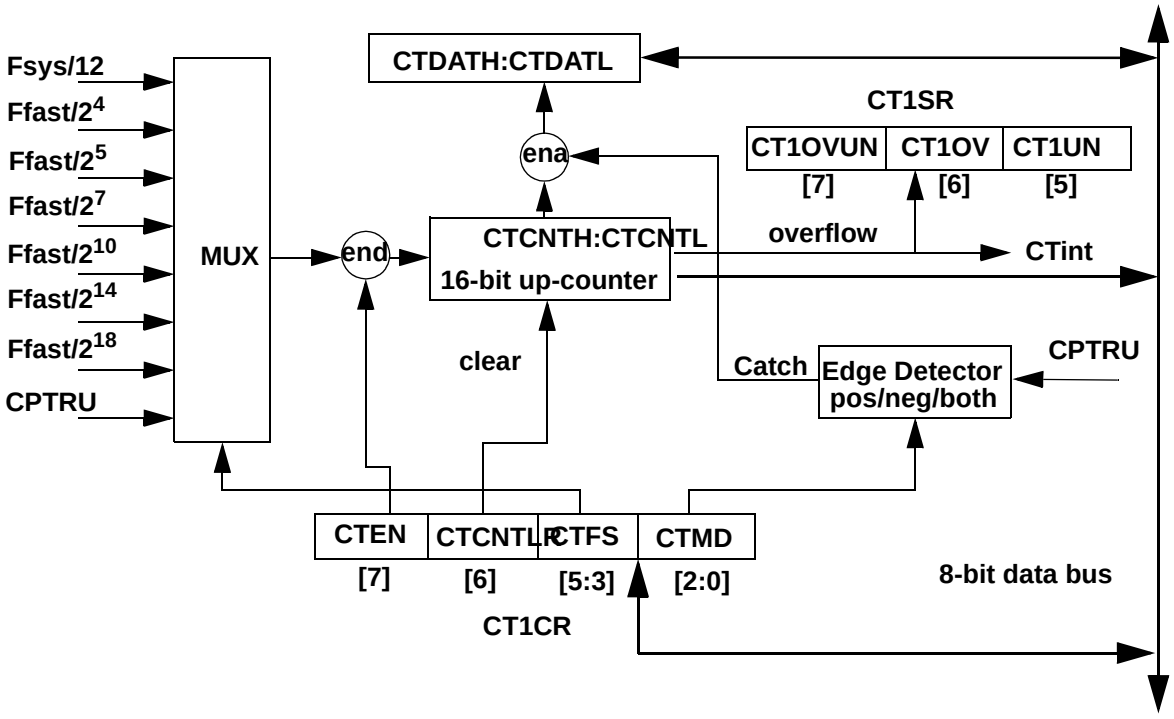


Figure 8. *Capture Timer in Capture Mode*

9.5. Capture Timer Control Registers (CT1CR/CT2CR)

Address	D1H/E1H							
Bit Addr.								
BIT	7	6	5	4	3	2	1	0

NAME	CT1EN CT2EN	CT1CNTLR CT2CNTLR	CT1FS2 CT2FS2	CT1FS1 CT2FS1	CT1FS0 CT2FS0	CT1MD1	CT1MD1 CT2MD1	CT1MD0 CT2MD0
Definition	C-Timer ENable	C-Timer counter CLeaR	C-Timer input Frequency Select bit-2	C-Timer input Frequency Select bit-1	C-Timer input Frequency Select bit-0	C-Timer1 operation MoDe bit-2	C-Timer operation MoDe bit-1	C-Timer operation MoDe bit-0
Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R/W	R/W Always 0 read.	R/W	R/W	R/W	R/W for C-Timer1 R for C-Timer2	R/W	R/W

TABLE 10. *Definition of CT1CR/CT2CR*

Name	Bit	Description (Note: n = 1 or 2)
CT1EN CT2EN	7	C-Timer ENable bit [Bit Status: 0 (Initial Value)] A selected input clock is halted. The CTnCNTH & CTnCNTL keeps counting value. [Bit Status: 1] A selected input clock is run. The CTnCNTH & CTnCNTL counts up the negative edge of the selected clock.
CT1CNTLR CT2CNTLR	6	CTn counter CLeaR bit If this bit is written with high, the CTnCNTH & CTnCNTL (16-bit up-counter) of the C-Timer will be cleared to 0000H. Writing low will not affect anything. When read, a low(0) will be always read.
CT1FS[5:3] CT2FS[5:3]	5 to 3	C-Timer input clock Frequency Select bits [Bit Status: 000 (Initial Value)] $F_{sys}/12$ [Bit Status: 001] $F_{fast}/2^4$ [Bit Status: 010] $F_{fast}/2^5$ [Bit Status: 011] $F_{fast}/2^7$ [Bit Status: 100] $F_{fast}/2^{10}$ [Bit Status: 101] $F_{fast}/2^{14}$ [Bit Status: 110] $F_{fast}/2^{18}$ [Bit Status: 111] CPTRU1 or CPTRU2 To use the CPTRU1 or CPTRU2 pin as an Event Input or Up Down-Count Input, the pin should be assigned to input mode by a port control register.
CT1MD[2:0] CT2MD[2:0]	2 to 0	C-Timer MoDe select bits [Bit Status: 000 (Initial Value)] Auto-Reload Mode [Bit Status: 001] Positive Edge Detect Capture Mode [Bit Status: 010] Negative Edge Detect Capture Mode [Bit Status: 011] Any Edge (Positive or Negative) Detect Capture Mode [Bit Status: 1XX] (Capture-Timer1 only) Up Down-Count Mode CT2MD[2] of C-Timer2 is unused.

TABLE 11. *The Description of CT1CR/CT2CR*

9.6. Capture Timer-1 Status Registers (CT1SR)

Address	D6H							
Bit Addr.								
BIT	7	6	5	4	3	2	1	0
NAME	CT1OVUN	CT1OV	CT1UN					
Definition	C-Timer1 Overflow and Underflow Interrupt	C-Timer1 Overflow	C-Timer1 Underflow	Reserved	Reserved	Reserved	Reserved	Reserved
Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R	R	R	R	R	R	R	R

TABLE 12. *Definition of CT1SR*

Name	Bit	
CT1OVUN	7	C-Timer1 Overflow and Underflow Interrupt bit [Bit Status: 0 (Initial Value)]: No Overflow and no Underflow. [Bit Status: 1]: Overflow or Underflow.
CT1OV	6	C-Timer1 Overflow bit [Bit Status: 0 (Initial Value)]: No Overflow. [Bit Status: 1]: Overflow.
CT1UN	5	C-Timer1 Underflow bit [Bit Status: 0 (Initial Value)]: No Underflow. [Bit Status: 1]: Underflow.
	4 to 0	Reserved bit

TABLE 13. *Description of CT1SR*

9.7. C-Timer Counter Registers (CT1CNTH & CT1CNTL / CT2CNTH& CT2CNTL)

Address	D3H/E3H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	CTnCNTH[7:0]							
Definition	C-Timer-n CouNTer register bit 15 to 8 (Auto-Reload Mode: High 8-bit of the Modulo N Up-counter: 0000H to N-1, N is the content of CTnDATH & CTnDATL) (Capture Mode: High 8-bit of the Modulo 65536 Up-counter: 00 to 65535) (Up Down-Count Mode: High 8-bit of the Modulo 65536 Up and Down- counter: 00 to 65535 (C-Timer1 only))							
Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R	R	R	R	R	R	R	R
Write by Hardware	Written with a new count value 0 or 1. 0 by CTnCNTLR.	Written with a new count value 0 or 1. 0 by CTnCNTLR.	Written with a new count value 0 or 1. 0 by CTnCNTLR.	Written with a new count value 0 or 1. 0 by CTnCNTLR.	Written with a new count value 0 or 1. 0 by CTnCNTLR.	Written with a new count value 0 or 1. 0 by CTnCNTLR.	Written with a new count value 0 or 1. 0 by CTnCNTLR.	Written with a new count value 0 or 1. 0 by CTnCNTLR.

TABLE 14. *Definition of CTnCNTH (n = 1 or 2)*

[

Address	D2H/E2H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	CTnCNTL[7:0]							
Definition	C-Timer-n CouNTer register bit 7 to 0 (Auto-Reload Mode: Low 8-bit of the Modulo N Up-counter: 0000H to N-1, N is the content of CTnDATH & CTnDATL) (Capture Mode: Low 8-bit of the Modulo 65536 Up-counter: 00 to 65535) (Up Down-Count Mode: Low 8-bit of the Modulo 65536 Up and Down- counter: 00 to 65535 (C-Timer1 only))							

Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R	R	R	R	R	R	R	R
Write by Hardware	Written with a new count value 0 or 1. 0 by CTnCNTL R.	Written with a new count value 0 or 1. 0 by CTnCNTL R.	Written with a new count value 0 or 1. 0 by CTnCNTL R.	Written with a new count value 0 or 1. 0 by CTnCNTL R.	Written with a new count value 0 or 1. 0 by CTnCNTL R.	Written with a new count value 0 or 1. 0 by CTnCNTL R.	Written with a new count value 0 or 1. 0 by CTnCNTL R.	Written with a new count value 0 or 1. 0 by CTnCNTL R.

TABLE 15. *Definition of CTnCNTL (n = 1 or 2)*

Name	Bit	Description
CTnCNTH[7:0] & CTnCNTL[7:0]	15 to 0	C-Timer-n CouNTer bits [Auto-Reload Mode] A Modulo-N (0000H to N-1) up-counter. N is the content of CTnDATH & CTnDATL. This counter counts up from 0000H to N-1. Then, the counter will go back to 0000H and the comparator will generate match signal. The Match signal will issue an interrupt request. The counter will not stop after the Match. Thus, any C-Timer will generate continuous Match signals with a fixed period. [Capture Mode] A Modulo-65536 (0000H to 65535) up-counter. This counter counts up from 0000H to 65535. Then, the counter will go back to 0000H and the CTnINT will be set to high. The counter will not stop after the overflow. Thus, any C-Timer overflow will generate continuous interrupt request with a fixed period. [Up Down-Count Mode] (Capture-Timer1 only) A Modulo-65536 (0000H to 65535) up and down-counter. This counter counts up on the negative edge of CPTRU1 and counts down on the negative edge of CPTRD1. Either the overflow or the underflow will set CTnINT bit. The counter will not stop after the overflow or the underflow. Thus, any C-Timer1 overflow and underflow will generate continuous interrupt request.

TABLE 16. The Description of CTnCNTH & CTnCNTL

9.8. C-Timer Data Register (CT1DATH & CT1DATL / CT2DATH & CT2DATL)

Address	D5H/E5H

Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	CTnDATH[7:0]							
Definition	C-Timer-n DATa register bit 15 to 8							
Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware	[Auto-Reload] none [Capture] Written with the content of CTnCNTH [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTH [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTH [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTH [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTH [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTH [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTH [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTH [Up Down-Count] none

TABLE 17. Definition of CTnDATH (n = 1 or 2)

Address	D4H/E4H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	CTnDATL[7:0]							
Definition	C-Timer-n DATa register bit 7 to 0							
Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Write by Hardware	[Auto-Reload] none [Capture] Written with the content of CTnCNTL [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTL [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTL [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTL [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTL [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTL [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTL [Up Down-Count] none	[Auto-Reload] none [Capture] Written with the content of CTnCNTL [Up Down-Count] none

TABLE 18. *Definition of CTnDATL (n = 1 or 2)*

Name	Bit	Description
CTnDATH[7:0] & CTnDATL[7:0]	15 to 0	C-Timer-n DATa bits [The first Match period in Auto-Reload Mode] C-Timer Match signal period = $(1 / F_{selected\text{-}clock})$ X ({CTnDATH,CTnDATL} + 1) - Deviation period 0 <= Deviation period < $(1 / F_{selected\text{-}clock})$ [The second or later Match period in Auto-Reload Mode] C-Timer Match signal period = $(1 / F_{selected\text{-}clock})$ X CTnDATH&CTnDATL (No Deviation period) [In Capture Mode] The CTnDATH & CTnDATL is updated by the content of the CTnCNTH & CTnCNTL when a selected edge of CPTRU1 or CPTRU2 is happened. [In Up Down-Count Mode] The CTnDATH & CTnDATL is not used in Up Down-Count Mode.

TABLE 19. The Description of CTnDATH & CTnDATL

10. WatchDog Timer

10.1. Overview

The WD-Timer (WatchDog Timer) can be used as a watch-dog timer or a basic interval timer. The main purpose of the WD-Timer is to initialize the MCU again from an unexpected device upset state. For instance, if a program falls into an infinite loop

because of noise, the WD-Timer will make the CPU wake up. By enabling the WatchDog Reset, this feature will be active. By disabling the WatchDog Reset and enabling the WatchDog Interrupt, the WD-Timer can be used as a basic interval timer. Only Fsys/12 is supported when under SlowAll or SlowPeri modes.

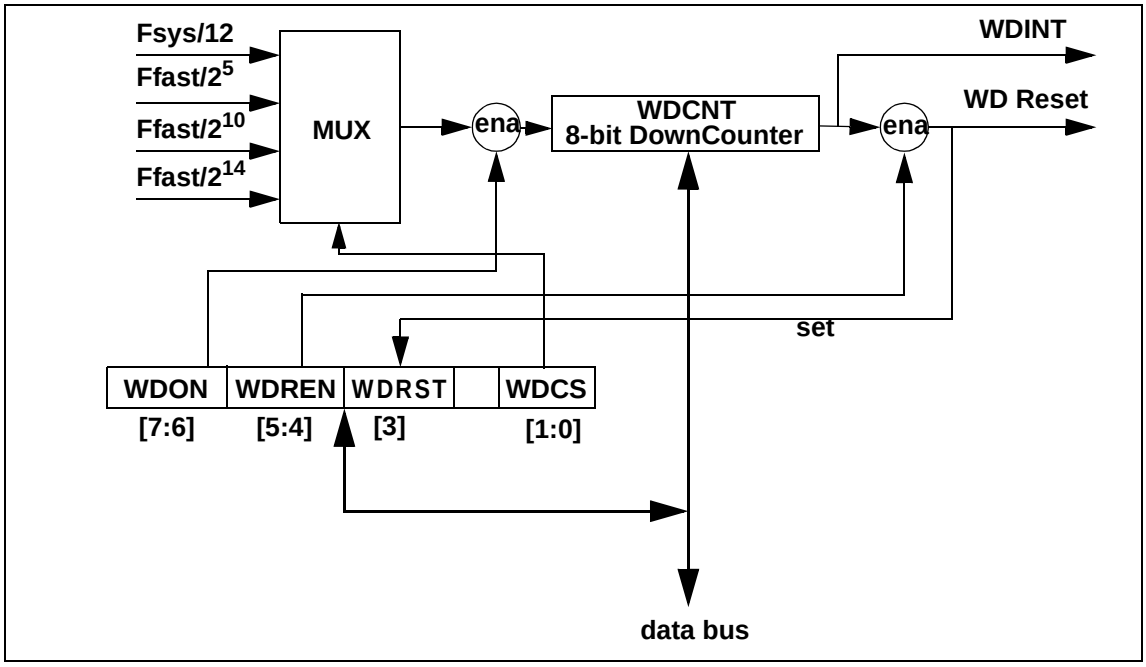


Figure 9. *Watch Dog Timer*

10.2. WatchDog Timer Control Register (WDCR)

Address								
Bit Addr.								
BIT	7	6	5	4	3	2	1	0
NAME	WDON1	WDON0	WDREN1	WDREN0	WDRST		WDCS1	WDCS0
Definition	WD-timer turn ON bit-1	WD-timer turn ON bit-0	WD-timer Reset generation ENable bit-1	WD-timer Reset generation ENable bit-0	WD-timer Reset STatus	Reserved	WD-timer counting Clock Select bit-1	WD-timer counting Clock Select bit-0

Reset Value	0	1	0	1	0: Pin Reset 1: WD Reset	0	0	0
Read/ Write by Software	R/W	R/W	R/W	R/W	R	R	R/W	R/W
Write by Hardware					0 by Exter-nal Pin Reset 1 by WD-Timer Reset			

TABLE 20. Definition of WDCR

Name	Bit	Description
WDON[1:0]	7 to 6	WD-Timer turn ON bits [Bit Status: 01 (Initial Value)] In this mode, the WD-Timer turns off. This bit status makes a selected counting clock frozen. Thus, the 8-bit WD-Timer Register keeps previous value without down-counting. [Bit Status: 10] In this mode, the WD-Timer turns on. This bit status makes a selected counting clock run. Thus, the 8-bit WD-Timer Register operates as a down-counter. [Bit Status: 00 or 11] Do **NOT** write these values on these two bits. In these mode, the WD-Timer turns on just like status-10. The purpose of these two redundant status is to prevent unexpected WD-Timer stop by device upset. While the WD-Timer is running by setting 10 to the WDON[1:0], unexpected electrical shock can make a WDON bit toggle. If another WDON bit retains its value, the WD-Timer will be continuously running without stop.

WDREN[1:0]	5 to 4	WD-Timer Reset generation ENable bits [Bit Status: 01 (Initial Value)] In this mode, the WD-Timer can not generate a Reset signal even though the underflow of the WD-Timer takes place. This allows the WD-Timer to be used as a simple Interval Timer. [Bit Status: 10] In this mode, the WD-Timer will generate a Reset signal if the underflow of the WD-Timer takes place. After WD-Timer Reset, the program will begin at the ROM address 0000H. All registers will be initialized. [Bit Status: 00 or 11] Do **NOT** write these values on these two bits. In these mode, the WD-Timer will generate a Reset signal just like status-10 if the underflow of the WD-Timer takes place. The purpose of these two redundant status is to prevent unexpected WD-Timer behavior by device upset. While the WD-Timer is running with setting 10 to the WDREN[1:0], unexpected electrical shock can make a WDREN bit toggle. If another WDREN bit retain its value, the WD-Timer can generate a Reset signal.

TABLE 21. *The Description of WDCR bit-7 to 4*

Name	Bit	Description
WDRST	3	WD-Timer Reset Status bit [Bit Status: 0] The chip was initialized by the external pin reset signal. [Bit Status: 1] The chip was initialized by the WD-Timer underflow reset signal.
	2	Reserved bit
WDCS[1:0]	1 to 0	WD-Timer counting Clock Select bit [Bit Status: 00] $F_{sys}/12$ [Bit Status: 01] $F_{fast}/2^5 = F_{fast}/32$ [Bit Status: 10] $F_{fast}/2^{10} = F_{fast}/1024$ [Bit Status: 11] $F_{fast}/2^{14} = F_{fast}/16384$

TABLE 22. The Description of WDCR bit-3 to 0

10.3. WatchDog Timer Register (WDCNT)

Address	
Bit Addr.	None

BIT	7	6	5	4	3	2	1	0
NAME	WDCNT[7:0]							
Definition	WatchDog Timer Register (8-bit down-counter)							
Reset Value	FFH							
Read/ Write by Software	R/W							
Write by Hardware	Written with a new count value.							

TABLE 23. Definition of WDCNT

10.4. Operation

- STEP-1:

First of all, a timer interval period should be decided. Then, a WDCNT value can be decided. Any instruction, whose destination is the WDCNT, can be used to write a value to the WDCNT. If the content of the WDCNT is FFH, that is the initial value after any Reset, and FFH is the needed value, you do not have to perform a write instruction to the WDCNT.

The interval period is as follows.

WD-Timer interval time = $(1 / F_{selected\text{-}clock})$ X (WDCNT + 1) - Deviation period

0 <=Deviation period < $(1 / F_{selected\text{-}clock})$

- STEP-2:

You should perform a write instruction to the WDCR with proper content. To turn the WD-Timer on, you must set the WDON[1:0] to 10. While the WD-Timer is running, if you change the content of the WDON[1:0] to 01 the WDCNT will retain its last count value. If the WD-Timer Reset is needed, the WDREN[1:0] should be set to 10. If the WDREN[1:0] is 01, the WD-Timer can be used as a basic interval timer. You can choose a counting clock among four clock sources.

- STEP-3:

While the WD-Timer is running, software must repeatedly re-initialize the WDCNT before the WD Reset is generated. This write operation can be performed without halting the WD-Timer.

- STEP-Underflow:

If the WD Reset is enabled and a WD underflow happens, the MCU goes into Reset state and all registers are initialized again. The Program Counter will point to the address 0000H. The software initialization routine can check the WDRST bit to distinguish the reset source.

In any case, the WDINT bit is set to high when the underflow happens. If the WD-Timer interrupt is enabled by the Interrupt Control Block, the interrupt service routine for the WD-Timer will be served. At this time the WDINT bit will be cleared by hardware.

Note: Writing “FFh” to the down counter register while it is in the “00h” state, might cause the “Underflow”. Writing anything other than “FFh” will not cause the “Underflow”.

11. SPI (Serial Peripheral Interface)

11.1. Overview

The SS1102C includes an 8-bit SPI which allows to communicate with peripheral or microcontroller devices equipped with a compatible SPI function. The SPI supports full-duplex communications by allowing 8-bit of data to be synchronously transmitted and received. An SPI system should contain one master device and several slave devices.

The SS1102C can only be configured as an SPI master.

11.2. SPI Pin and Timing Description

Basically, three pins (MI, MO, SCK) are used to accomplish the communication:

- Master In (MI): MI is configured as a data input. The most significant data bit is receive first.
- Master Out (MO): MO is configured as a data output. The most significant data bit is sent first.
- Serial Clock (SCK): The master clock used to synchronize data transfer through MI and MO. Since SCK is generated by the master device, this pin is configured as an output.

11.3. SPI Timing Description

As shown in Fig. 1, four possible timing relationships may be chosen by using control bits CPOL (clock polarity) and CPHA (clock phase) in the serial peripheral control register(SPI1CR). Both master and slave device must operate with the same timing.

The clock frequency is selected by using control bits SPR0 and SPR1 in the SPI1CR of the master device. The slave device has no control of the clock frequency.

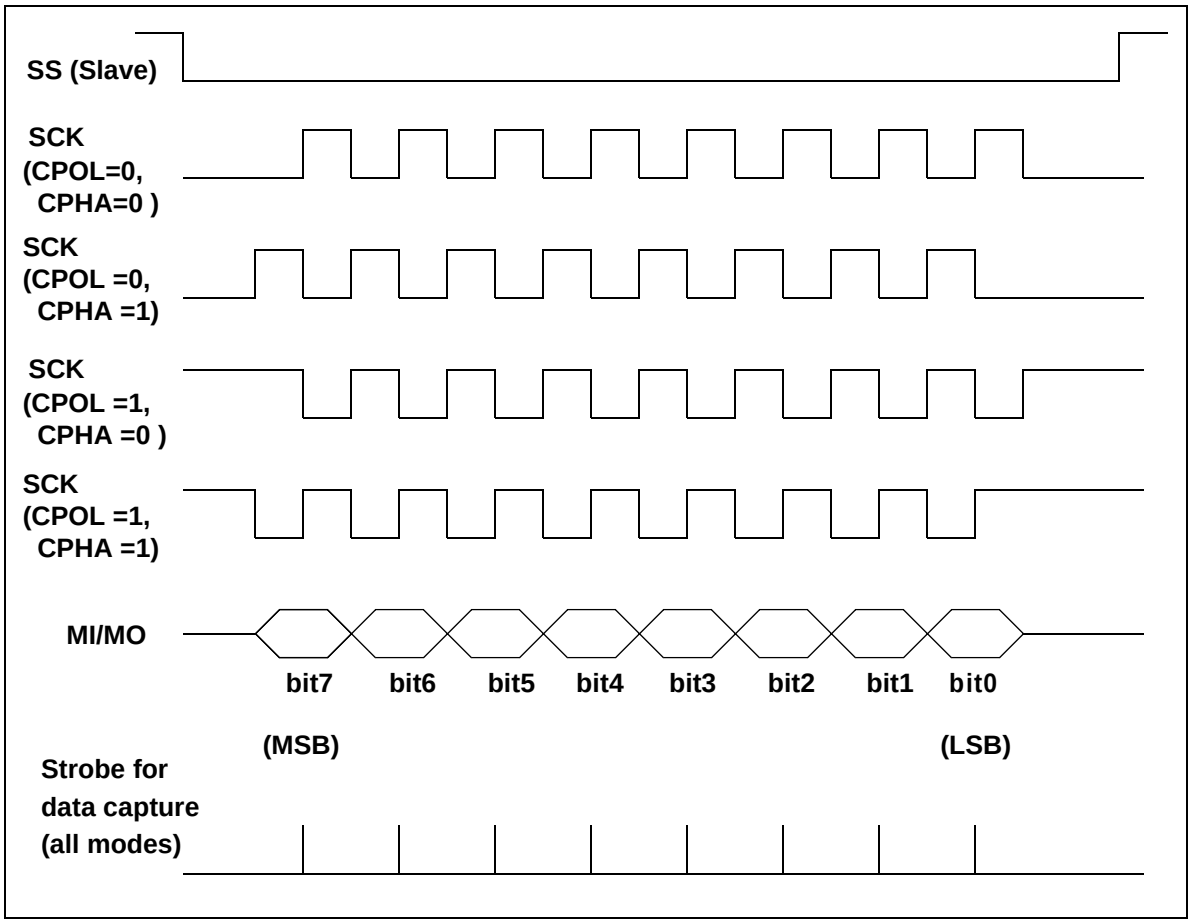


Figure 10. *SPI Timing Diagram*

11.4. SPI Operation

The SPI has a shift register and a receive buffer. The shift register shifts data in and out of the device. The data byte transmitted is replaced by the data byte received. The SPI data register(SPI1DAT) actually represents two distinct devices. Writes to SPI1DAT are writes to the shift register itself. On the other hand, reads of SPI1DAT do not read from the shift register, but rather from a buffer register which is automatically loaded from the shift register upon the completion of a data transfer. This double buffering allows the next data byte to start reception before reading the data that was just received. Any write to the SPI1DAT during data transfer will be ignored, and will cause the write collision(WCOL) status bit in the SPI1SR to be set. After the data transfer is completed, internal interrupt is generated and the SPIF flag of the SPI1SR is set. Clock is generated and data is shifted as soon as data is written to the shift register.

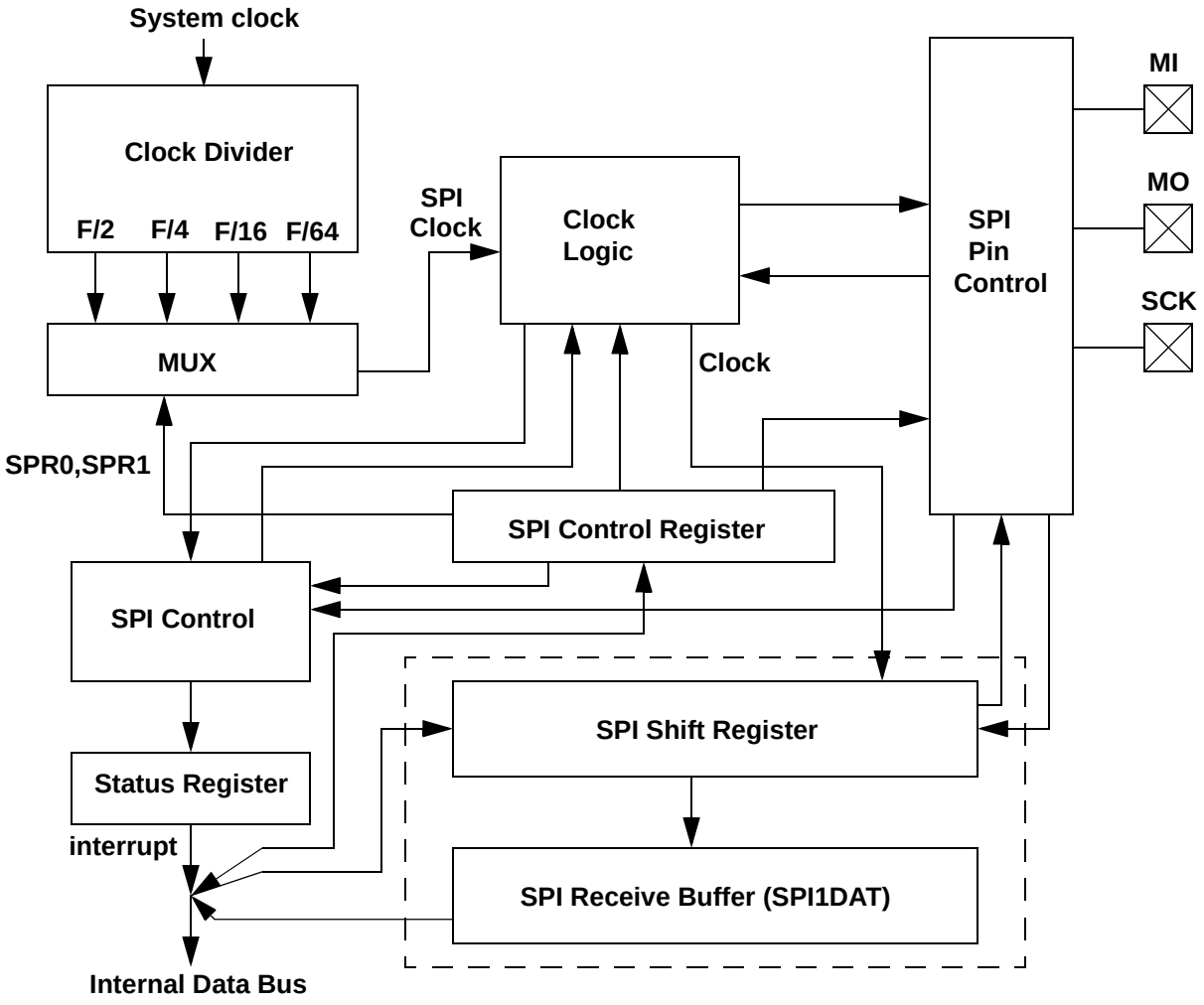


Figure 11. *SPI Block Diagram*

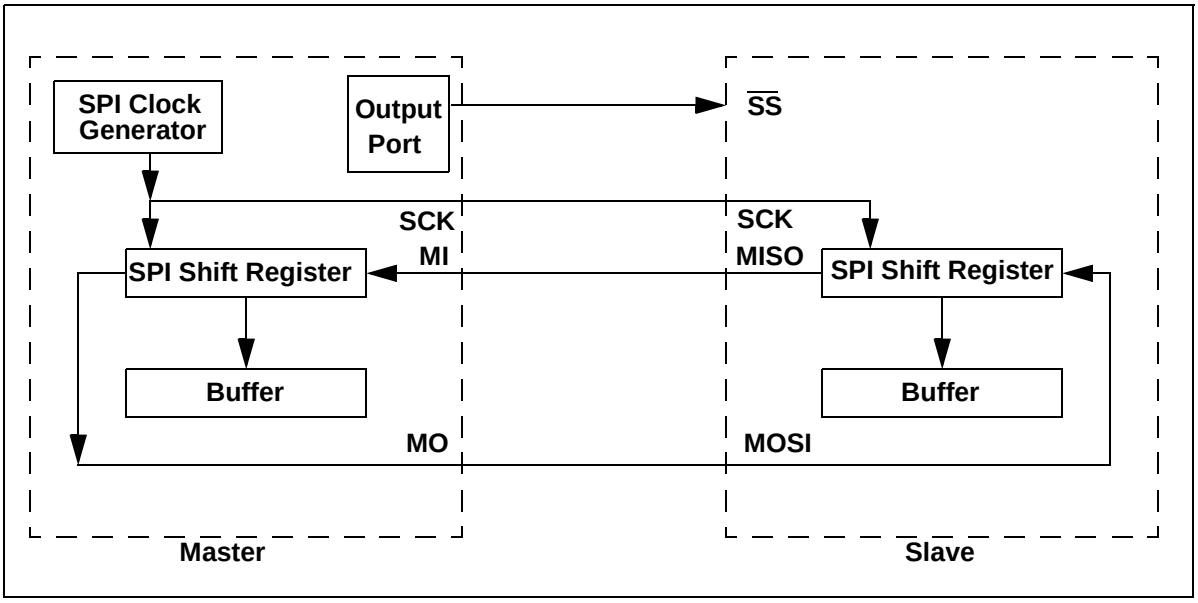


Figure 12. *SPI Master-Slave Interconnection*

11.5. SPI Control Register (SPICR)

Address	F1H							
Bit Addr.								
BIT	7	6	5	4	3	2	1	0
NAME	SPIE	SPE			CPOL	CPHA	SPR1	SPR0
Definition	SPI Interrupt ENable	SPI System Enable bit	Reserved bit	Reserved bit	SPI Clock Polarity Select	SPI Clock Phase	SPI Clock Rate Select Bit 1	SPI Clock Rate Select Bit 0
Reset Value	0	0	0	1	0	0	0	0
Read/ Write by Software	R/W	R/W	R	R	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 24. *Definition of SPICR*

Name	Bit	Description
SPIE	7	SPIE: Serial Peripheral Interrupt Enable 0 = Disables internal SPI interrupt (SPIF) 1 = Enables internal SPI interrupt (SPIF
SPE	6	SPE: Serial Peripheral System Enable 0 = Turn off SPI system (P1.5-P1.7 can be general purpose) 1 = Turn on SPI system (P1.5-P1.7 are for SPI)
Reserved [5:4]	5-4	
CPOL	3	CPOL: Clock Polarity 0 = Idle state for clock is high 1 = Idle state for clock is low
CPHA	2	CPHA: Clock Phase 0 = Transmit data on falling edge and receive data on rising edge 1 = Transmit data on rising edge and receive data on falling edge When CPHA = 0, as soon as SS (Slave Select in slave device) goes low, the transaction begins and data will be sampled on first edge of SCK. When CPHA = 1, the SS (Slave Select in slave device) may be thought of as simple enable control.
SPR[1:0]	1 to 0	SPR1 and SPR0: SPI Clock Rate Select: 00 = Fclk/2 01 = Fclk/4 10 = Fclk/16 11 = Fclk/64

TABLE 25. *Description of SPICR*

11.6. SPI Status Register (SPI1SR)

Address	F3H							
Bit Addr.								
BIT	7	6	5	4	3	2	1	0
NAME	SPIF	WCOL						
Definition	SPI Transfer Complete	SPI Write Collision Detect						
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R	R	R	R	R	R

TABLE 26. *SPI Status Register (SPI1SR)*

Name	Bit	Description
SPIF	7	SPIF: SPI Transfer Complete Flag 0: Data transfer not complete 1: Data transfer complete If SPIF goes high and if SPIE is set, an internal interrupt is generated. SPIF can be cleared by reading SPI1SR followed by an read of the SPI1DAT. A write to SPI1DAT will be ignored if SPIF is set.
WCOL	6	WCOL: SPI Write Collision Detect 0: No collision 1: Collision: Attempt to write to SPI1DAT while data transfer is still in progress. When CPHA is low, a transfer starts when $\overline{SS}$ (Slave Select in slave device) goes low and finishes when $\overline{SS}$ (Slave Select in slave device) goes high. When CPHA is high, a transfer starts when SCK first becomes active while $\overline{SS}$ (Slave Select in slave device) is low. The transfer finishes when SPIF is set. WCOL can be cleared by reading SPI1SR followed by a read of the SPI1DAT.
Reserved [5:0]	5-0	

TABLE 27. *Description of SPI Status Register (SPI1SR)*

11.7. SPI Shift Register (SPIDAT)

Address	F2H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	SPIDAT[7:0]							
Definition								
Reset Value	0	0	0	0	0	0	0	0
Read/ Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware	Written with a new shift value 0 or 1.	Written with a new shift value 0 or 1.	Written with a new shift value 0 or 1.	Written with a new shift value 0 or 1.	Written with a new shift value 0 or 1.	Written with a new shift value 0 or 1.	Written with a new shift value 0 or 1.	Written with a new shift value 0 or 1.

TABLE 28. Definition of SPIDAT

To send the data SP1CR should be set first, then the data should be written into SPI1DAT. In the interrupt subroutine SPIINT should be cleared by the application software. Even though if the SPIINT is not enabled, the flag should be cleared by the software to support another data transmission or receiving phases.

12. Direct Sequence Spread Spectrum Baseband Modem (SSTM)

The SSTM is a low-power, multi-purpose communication module designed to support spread spectrum data communications. The SSTM contains all the baseband function required for an FCC Part 15 compliant device.

The SSTM supports both full-duplex and half-duplex operations in synchronous mode. In the half-duplex, no assumption about higher level protocols is made. Instead, the SSTM is designed to be flexible and can be configured for a variety of uses.

In full-duplex operation communication is achieved by using a time-division duplex (TDD) protocol and burst structure. The SSTM supports full-duplex data rates up to 64Kbps and half-duplex data rates up to 166 Kbps.

The SSTM is made up of five functional modules. These include the receiver, the transmitter, the time-division duplex (TDD) controller, the transmit and receive FIFOs, and the master clock generator.

12.1. Receiver

The receiver module performs all the digital signal processing required by the spread spectrum receiver, including de-correlation and demodulation.

The receiver samples the incoming baseband signal at two samples per PN chip. The samples are correlated with four possible PN sequences in 64-bit parallel correlators. The de-correlated signal is demodulated via a digital phase locked loop. To reduce power consumption, the receiver is powered down while the SSTM is transmitting and consumes peak power only during the brief period of initial acquisition. After acquisition, the receiver goes into tracking/detection mode, where the power consumption of the receiver module is reduced by two orders of magnitude.

12.2. Transmitter

The transmit module generates the spread spectrum binary sequence for output to the RF modulator.

The transmitter logic encodes two consecutive bits of data into one of four possible 32-bit PN sequences. The transmitted PN sequence is further randomized by modulus-2 addition with a fixed 2047-bit long PN sequence. This operation smooths the output spectrum of the transmitted signal and eliminates discrete spectral components. During TDD operation, the transmitter is powered off during the portion of the cycle when the SSTM is in receive mode in order to save power. The transmitter output is at high-impedance state during a receiving period.

12.3. TDD Controller

The time-division duplex (TDD) controller implements the “ping-pong” protocol that allows a full-duplex link to be emulated by a half-duplex radio. The TDD controller also generates the appropriate clock and control signals to other modules of the SSTM. In full-duplex mode, the TDD controller multiplexes and de-multiplexes the overhead bits with the data bit-stream. The TDD controller also uses a digital phase locked loop to maintain an equal read and write rate to the FIFOs as to avoid FIFO overflow or underflow. In addition, the TDD controller contains logic to generate the proper handshaking signals.

12.4. FIFOs

The transmit and receive FIFOs are used to buffer the transmit and receive data. The SSTM includes a 30-byte transmit FIFO and a 30-byte receive FIFO to buffer the input and output data. The control signals for the FIFOs are generated by the TDD controller. The FIFOs provide the internal interrupt signals for the microprocessor.

12.5. Master Clock Generator

The master clock generator generates the various clock signals required by the modules described above. It can be disabled in power saving mode.

12.6. Full-Duplex Operation

Although the SSTM actually only uses a half-duplex channel for communication with the remote device, full-duplex operation is provided by using a time-division duplex (TDD) protocol. The TDD protocol basically configures the SSTM alternatively as a transmitter and as a receiver. When two devices are communicating with each other, one is programmed to be the master, while the other is programmed to be the slave. The TDD protocol ensures that while the master is transmitting, the slave is receiving and vice versa in a timely fashion. The end result is that as far as the user is concerned, the communication link appears to be full-duplex. In order to achieve this, it is necessary for the SSTM to transmit at a higher rate than the actual user data rate. Ideally, for TDD operation, with 100% efficiency and 0% overhead, the SSTM must transmit the data at twice the user data rate since the SSTM has only half the time to transmit the user data (during the other half time period, the SSTM is receiving from the remote station). Overhead such as preamble, unique word (UW), as well as other signaling information bits result in the SSTM transmitting at 2.6 times the effective user data rate. The size of the FIFOs on the SSTM is designed to provide sufficient buffer during both transmit and receive operations so that underflow or overflow of the FIFOs does not occur.

When communication between two devices first commences, the microprocessors must program one of the devices as the Master and the other device as the Slave. The master device transmits periodic bursts as soon as the reset signal is released. The burst timing of the Master is derived from its internal master clock oscillator and can be computed from (EQ 1) below:

$$f_{burst} = \frac{f_{mosc}}{192} \quad \textbf{(EQ 1)}$$

In (EQ 1), f_{burst} is the burst rate and f_{mosc} is the master oscillator frequency. As the transmitter uses a quadrature modulation scheme, the chip rate is 16 times the burst rate or

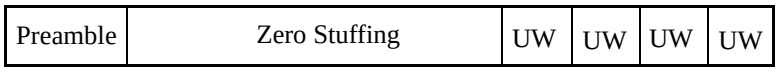
$$f_{chip} = \frac{f_{mosc}}{12} = 16 \times f_{burst} \quad \textbf{(EQ 2)}$$

where f_{chip} is the chip rate and f_{mosc} is the master oscillator frequency. Effectively, the spread spectrum transceiver operates at 16 chips/bit or 32 bits/symbol where each symbol is composed of 2 bits.

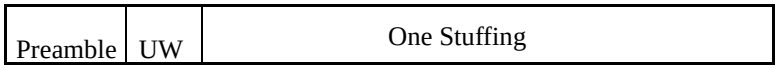
The total number of bits per burst is fixed and equal for both the Master and the Slave. The Slave derives its burst timing from the Master by detecting the UW pulse transmitted by the Master.

12.7. TDD Protocol

Initially, the two communicating devices need to establish “sync”. The TDD protocol achieves this by using a special handshaking protocol. The Master first transmits an “acquisition burst”. The acquisition burst consists of 32 bits of preamble (binary 0’s), followed by 230 bits of “zero stuffing”, and four 22-bit unique words (UW). When the Slave receives the acquisition burst from the Master correctly (by decoding the 4 consecutive UWs), it sends an acquisition burst in response. When the Master receives the acquisition burst, it sends an “empty burst”. An empty burst contains a 32-bit preamble followed by a single 22-bit unique word, and 296-bit of “1” (One stuffing). In response to the master’s empty burst, the Slave also sends an empty burst back to the Master. When the Master receives the empty burst from the Slave, the communication link is considered to have been established and “sync” condition achieved. On the following burst, both the Master and the Slave start genuine data transmission by sending out “data bursts”. Each of the data bursts contain a 32-bit preamble, followed by a 22-bit UW, a 8-bit Signaling Word (SW), and 288 bits of user data. The three different types of burst frame structures are shown in Figure 13.



Acquisition Burst Frame Structure



Empty Burst Frame Structure

Data Burst Frame Structure

Field	Bits
Preamble	32
Unique Word (UW)	22
Signaling Word (SW)	8
Data	288

Figure 13. *Burst Frame Structures*

Each burst cycle also includes 2 “Guard” times to allow for both propagation and RF transceiver switching time. More specifically, G_1 is a 28-bit delay between the time the Master stops transmission and the Slave commences transmission; G_2 is a 28-bit delay between the time the Slave stops transmission and the Master commences transmission. These guard times allow for a minimum delay of 328 µsec delay (for a master oscillator frequency of 16.384 MHz). The total burst cycle is 768 bits long, including 12 bits internal delay (the transmitter turns off 6 bits after the last data bit is latched into the transmitter, the master and slave therefore contribute a total of 12-bit internal delay).

During TDD operation, the receiver will go through several stages. Initially, when the 4 UWs of the acquisition burst has been received and decoded correctly, the receiver (either Master or Slave) declares “locked”. After the empty burst has been decoded, the receiver declares “rlocked” signifying that the remote device has locked. The behavior of the receiver after establishing the RLOCK condition depends on whether the internal state machine is turned on or not (determined by the setting of the configuration word, bit CI_h[4]). When the state machine is turned off, transmission will be turned off whenever the UW is not detected. The Slave then waits for a new acquisition burst while the Master will start the acquisition cycle again by transmitting an acquisition burst. Note that the Master will continue to broadcast acquisition bursts until it has received a proper acquisition burst from the Slave in response. The Master will always revert back to the initial acquisition mode (broadcasting acquisition bursts), whenever it fails to detect the proper UWs from the slave.

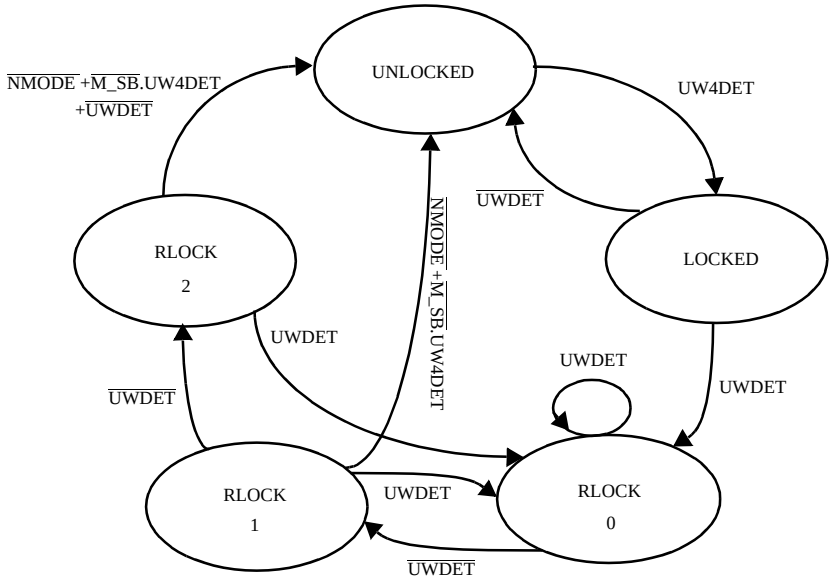


Figure 14. *Receiver Lock State Machine*

If the state machine is turned on, then the receiver will not declare LOCK loss right after UW has not been detected. Instead, it will allow for UW errors in up to two further bursts before declaration LOCK loss. The state diagram for this lock state machine is shown in Figure 14. In the figure, UW4DET indicates the condition when the four UWs have been detected during acquisition burst, UWDET indicate the condition where the single UW in empty and data bursts has been detected. M_SB is the programmed bit (CI_h[0]) which is a binary "1" when the SSTM is programmed to be the master and a binary "0" when programmed as a slave. NMODE is a signal generated by the receive logic when the digital phase locked loop in the receiver has achieved lock, it is similar to an RSSI signal. Note that the NMODE signal is independent of the UW detection. Physically, when NMODE is asserted, it indicates that PN acquisition has been achieved. The LOCKED and RLOCK states are as described previously.

12.8. System Delay

To calculate the delay through the system (see Figure 15.), assume that the transmit FIFO is almost empty at the end of a burst. Then the next bit that enters the transmit FIFO will experience a system delay (excluding propagation delay but including the internal logic delay) of approximately:

$$T_{delay} = 2 \times (T_G + T_{PR} + T_{UW} + T_{SW}) + T_{DAT} + T_{\Delta} \quad \textbf{(EQ 3)}$$

where T_G is the time delay due to Guard Time (note: G = G1 = G2 = 28bits), T_{PR} is the time delay due to preamble, T_{UW} is the time delay of the UW, T_{SW} is the time delay for

signaling word transmission, T_{DAT} is the time delay for data transmission, and T_{Δ} is the internal logic delay. For a 32 Kbps full duplex data rate signal, with a master oscillator (MO) of 16.384 MHz, this system delay translates into 5.625 msec.

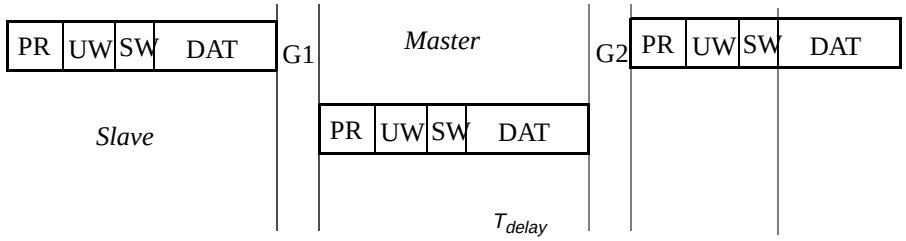


Figure 15. *System Delay*

12.9. Timing Information

12.9.1 Full Duplex Operations

In Full Duplex mode chips communicating with each other should have RTS_N enabled. A chip which is configured as a Master starts the acquisition procedure and data transmission. A Slave waits for a valid spread spectrum signal to arrive.

The SSTM transmits and receives data through TXFIFO and RXFIFO buffers. The writing and reading of data to and from the FIFOs are supported by the internal SS1102C bus. The synchronization of the data exchange between the microprocessor and the SSTM is provided by the use of the internal interrupt signals RXFIFO and TXFIFO connected to the interrupt module of the SS1102C. An interrupt indicating when LOCK is achieved is also sent to the interrupt module.

The interrupts and transfer of data with the FIFOs is allowed once LOCK is achieved. When an SSTM interrupt is sent to the SS1102C interrupt module the corresponding bits of interrupt status register in the interrupt module or the SSTM module indicate that the TXshl bit from TXFIFO or/and RXshh bit from RXFIFO is set. TXFIFO interrupt will be active if TXFIFO index is below the programmed TXshl and will remain active until the index is above the TXshh. The RXFIFO interrupt will be active if RXFIFO index is above the programmed RXshh and will remain active until the index is below the RXshl. Thus the microprocessor must write data into the SSTM for transmission when the TXFIFO interrupt is asserted by the TXFIFO, and it must read data from the SSTM when the RXFIFO interrupt is asserted by the RXFIFO. Difference in value between thresholds high and low should be at least two bytes. The best performance is achieved when the TXshl is 5 and TXshh is 22.

When TXFIFO interrupts indicates that the TXshl control bit is initiated, data bytes should be written into the TXFIFO. To avoid the overflow of the TXFIFO and loss of data, the maximum number of bytes to be written should not be more than (30 - TXshl) bytes. The size of the burst is 288 bits or 30 bytes. If there is no enough user's data to fill

up a burst at the end of transmission of the user's data, "1"s will be sent automatically (when there is no data and transmission is enabled, all "1"s will be sent).

When RXFIFO interrupt indicates that the RXshh bit is initiated, data bytes should be read from the RXFIFO. The high level protocol should take care of the end of transmission to avoid situation when some data can be held in the RXFIFO indefinitely or lost on the Reset. Status bits for the overflow and underflow for the TXFIFO and overflow for the RXFIFO are provided.

The timing relationship between the data and interrupt signals for transmit and receive is shown in Figure 16. below. The rate with which the MCU feeds data to the transmitter depends on the values of thresholds and burst rates.

A typical start-up of the data link is also shown in Figure 17.

In the full-duplex mode the SSTM when programmed as a Master, starts transmission as soon as reset is released (RTS_N is enabled). As long as the Master is powered-on, the SSTM will send out the acquisition burst and try to establish a communication link with a remote Slave. Thus, the communication channel is occupied as soon as the Master resets, and this can occur before any data becomes available for transmission. Once the communication link is established, even when there is no data to be transmitted, the Master and Slave will remain in communication with each other (sending out "1" in the data field) indefinitely or until the Master is disabled.

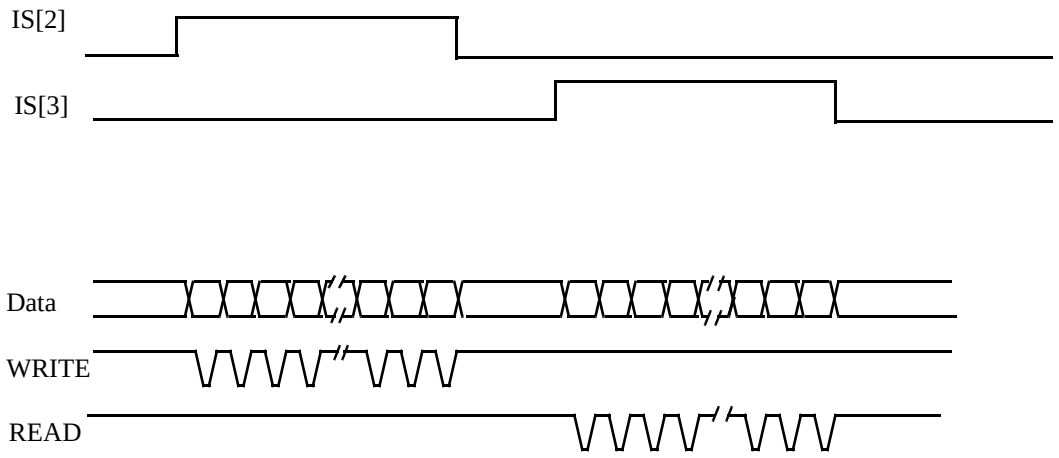


Figure 16. *Full-Duplex Data Interface Timing*

12.9.2 Threshold Value Calculation

The performance of the MCU should be considered when calculating TXshl and RXshh. The time for transmission of the TXshl bytes out of the TX FIFO should be more than the time required by the MCU to write new data into the TX FIFO. Accordingly, the

time for receiving RXshh bytes from the channel should be more than the time required by the MCU to read the data from the RX FIFO.

12.10. Half-duplex Operation

The half-duplex data mode is suitable for applications such as wireless LAN or pocket radio. The SSTM does not make any assumption about the higher level protocol and relies on these higher level protocols to provide the necessary framing, error correction, and preamble. The SSTM will transmit and deliver the data stream without multiplexing any protocol overhead bits, as it is the case for the full-duplex operation. Having the serial data stream from the channel and 8-bit wide parallel bus interface to the CPU, the data bytes should be aligned starting from the first bit of data. this alignment is supported by the SSTM hardware. The data alignment is achieved in the half duplex mode by the inserting the UW into the data stream as follows:

1. On the transmit side a 24 bits UW (22 bits of the UW and 2 bits “don't care”) should be inserted in front of the first data bit by the software. The 2 bits “don’t care” are the two MSBs of Byte0. The UW should not be loaded into the 3 UW registers of SS1105.

2. On the receive side the same 3 bytes of the UW should be loaded into the 3 UW registers of the SS1105.

3. A receiver recognizes the UW by SS1105 hardware and loads the data into the receive FIFO. The interrupt from the receive FIFO will be generated only after the UW was received properly. The data will be aligned in the receive FIFO by bytes.

4. The UW should be reloaded in the receiver each time the direction of transmission is changed (it should be done anyway because the chip will reset when the RTS_N value is changed). Thus if a transmitter became a receiver, the UW should be loaded into the register.

5. All required preamble bits should be sent anyway (at least 70 of them). The “Lock” UW is not received. It means that the “Lock” signal and “Lock” bit can be generated before the interrupt from the RXFIFO.

6. The UW should be inserted into the data stream as follows:

Transmit side: (Byte2 MSB first) (Byte1 MSB first) (Byte0 MSB first) (Preamble bits) ->

The transmit data and preamble should be loaded into the TX fifo in the following order:

Preamble N byte

Byte0

Byte1

Byte2

Receive side:

The UW should be loaded into the UW registers in the following order:

Regtr	Order
UW0	Byte2
UW1	Byte1
UW2	Byte0

It is the user's responsibility to ensure that each packet transmitted contains enough preamble bits so that acquisition can be achieved prior to actual data delivery.

Also, the invalid receive data will be present at the output due to hysteresis of the digital phase-locked loop. Thus, the user must be able to detect the end-of-packet from the data received rather than relying on the SSTM to signify loss of the interrupt signal or loss of the lock.

The SSTM in the Transmit mode (RTS_N is disabled) will transmit "1" if there is no data in the TXFIFO.

Because no overhead in multiplexing is required, in half-duplex mode, the highest data rate supportable is equivalent to the burst rate of the full-duplex mode. For example using a MO of 30.72 MHz this can be set at 160 Kbps. The relationship between the data rate and the required MO is as followed:

$$f_{data} = \frac{f_{mosc}}{192} \quad \textbf{(EQ 4)}$$

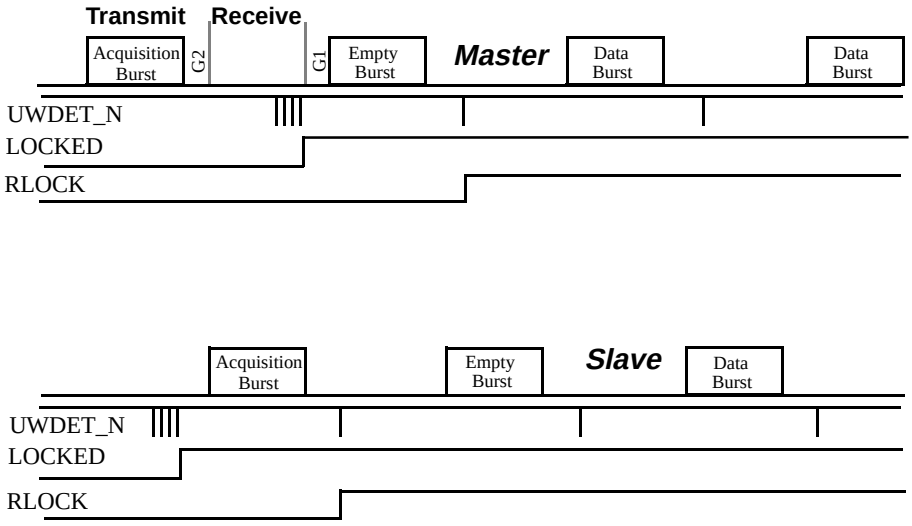
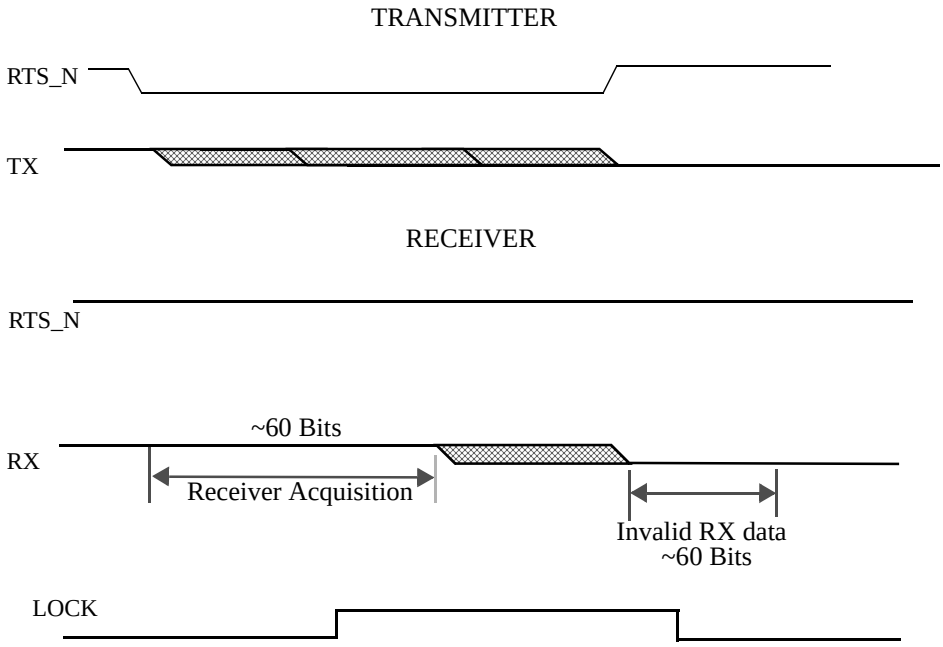


Figure 17. *Typical Communication Link Start-Up*

The timing relationships between RTS_N and data exchange remain unchanged from those of the full-duplex operation. RTS_N designates the direction of transmission. The transmission is directed from a chip with RTS_N enabled to a chip with RTS_N disabled. To change the direction of transmission the value of RTS_N should be reversed. Note that, in the half-duplex mode, when RTS_N changes value the SSTM resets itself. The transmitter stops transmission as soon as RTS_N is disabled. It is up to the user to include and detect end-of-packet information so that invalid data is not erroneously perceived as valid data. Finally, note that there is no provision for CSMA/CD type avoidance in the hardware, it is up to the user or the modem system to implement any desired collision avoidance schemes either in hardware and/or software. A typical timing diagram for half-duplex operation is shown in Figure 18.



* Drawing not to scale.

Figure 18. *Timing for half-duplex Operation*

The receiver will not be locked lock until the initial acquisition process has been completed. This process typically takes from 60-110 bits depending on the condition of the radio link. It is imperative, therefore, for the packet to carry framing information so that the beginning and end of the packet can be detected (for example, a high-speed synchronous data link protocol such as HDLC provides its own framing structure in the packets it transmits). In addition, sufficient preamble bits must be transmitted prior to actual data transmission so that the receiver will be locked and ready to deliver data when actual data arrives.

Note that there is a key difference between full-duplex and half-duplex operation. In a half-duplex operation, there is no concept of Master or Slave. The SSTM will start transmitting “1”s (if there is no data in the TXFIFO) or data when the user asserted the RTS_N control bit. The SSTM Receiver will receive “1”s or data, accordingly.

The SSTM Receiver will generate the interrupt signals and deliver “1”s to the MCU even if there is no transmission process at all. Those “1”s are a content of the RXFIFO.

12.11. Reading the Signaling Word and S/N Data

The Signaling Word (SW) and S/N register data are updated periodically by the SSTM. Once a Signaling Word or a S/N value has been loaded onto the register by the SSTM, the interrupt pending RXSW bit is asserted. Once the interrupt bit has been asserted, no

new data will be loaded onto the register until the interrupt it has been read and cleared by the microprocessor. On the transmit side an interrupt pending TXSW bit is asserted when the Signaling Word has been sent. The Signaling Word is updated or sent once every burst. The Signaling Word may also send interrupts to the CPU in the SS1102C (See the interrupt section for more details). The S/N data is calculated from the AGC circuit inside the SSTM once every 128 data bits (including overhead bits).

12.12. Control Registers

One register is available for controlling the SSTM block within the SS1102C. It is the SSTM control register. This register contains the enable and software reset for the SSTM module. It also contains the control to enable certain outputs of the SSTM to the SS1102 pins. Clearing the Reset bit in SSTMCR will clear the FIFO and the Reset bit will return by the hardware to "1" after it was cleared.

Address	C1H							
Bit Address								
BIT	7	6	5	4	3	2	1	0
NAME		RESET	ENABLE	LOCKOE	PLLSWOE	RFP-WROE	TXE-NOE	MOD-OUTOE
Definition		Software Reset for SSTM 0-reset 1-no reset	Enable for SSTM 1-enable 0-disable	LOCK output to P4.2 1-enable 0-disable	PLLSW output to P4.1 1-enable 0-disable	RFPWR output to P4.0 1-enable 0-disable	TXEN output to P3.3 1-enable 0-disable	MOD-OUT output to P3.4 1-enable 0-disable
Reset Value	0	1	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 29. *Table: Definition of SSTMCR*

12.13. SSTM Control Registers

A total of thirty-one registers are available for storing the various SSTM programming parameters. They are listed below:

Name	Address	Bank Select CI_I[1]	Index	Write/ Read	Functions
CI_h	8F	0	[7:0]	WR/ RD	CI_h[0]: Master/Slave Selection bit (1/0) CI_h[3:1]:Number of errors allowed in the UW CI_h[4]:LockSMon CI_h[5]:Data Mode Selection/Test Mode (1/0) CI_h[6]: Half/Full Duplex Select (1/0) CI_h[7]:RTS_N bit (0 to enable)
CI_I	8E	0	[7:0]	WR/ RD	CI_I[0]: Normal Mode CI_I[1]: High/Low Bank Select CI_I[3:2]:Width of ESD Window CI_I[4]: Set the Width of “Central Region” CI_I[5]: DPLL Accumulator 1 Reset CI_I[7:6]: Set the Width of “Detection Window”
RDI	FE	0	[7:0]	WR/ RD	Write data for TXFIFO Read data for RXFIFO
IE	C8	0	[7:0]	WR	** Writes to this register will have no effect. The Interrupts IS[0] to IS[3] and IS[7] for the SSTM are connected directly to the interrupt module in the SS1102C. Thus enable can be performed in the interrupt module. The status for the remaining interrupts can be read in the SSTM IS status register.
IS	C8	0	[7:0]	RD	IS[0]: RXSW Interrupt Pending IS[1]: S/N Interrupt Pending IS[2]: TXFIFO Threshold Interrupt Pending IS[3]: RXFIFO Threshold Interrupt Pending IS[4]: TXFIFO Underflow Pending IS[5]: TXFIFO Overflow Pending IS[6]: RXFIFO Overflow Pending IS[7]: TXSW Transmit Interrupt Pending
LCK	9E	0	[1:0]	RD	LCK[0]: LOCK Achieved for Full Duplex or NMODE Achieved for Half Duplex LCK[1]: RLOCK Achieved for Full Duplex
TXSW	AE	0	[7:0]	Write	Signaling Word to be transmitted
RXSW	AE	0	[7:0]	Read	Signaling Word received
S/N	AF	0	[7:0]	Read	Signal/Noise indicator
PNA0	BE	0	[7:0]	WR/ RD	PNA[7:0]
PNA1	CE	0	[7:0]	WR/ RD	PNA[15:8]

Name	Address	Bank Select CI_I[1]	Index	Write/ Read	Functions
PNA2	DE	0	[7:0]	WR/ RD	PNA[23:16]
PNA3	EE	0	[7:0]	WR/ RD	PNA[31:24]
PNB0	BF	0	[7:0]	WR/ RD	PNB[7:0]
PNB1	CF	0	[7:0]	WR/ RD	PNB[15:8]
PNB2	DF	0	[7:0]	WR/ RD	PNB[23:16]
PNB3	EF	1	[7:0]	WR/ RD	PNB[31:24]
UW0	8F	1	[7:0]	WR/ RD	UW[7:0]
UW1	FE	1	[7:0]	WR/ RD	UW[15:8]
UW2	C8	1	[7:0]	WR/ RD	UW[21:16]
TXshh	9E	1	[4:0]	WR/ RD	Transmitting FIFO interrupt threshold high
TXshl	9F	1	[4:0]	WR/ RD	Transmitting FIFO interrupt threshold low
RXshh	AE	1	[4:0]	WR/ RD	Receiving FIFO interrupt threshold high
RXshl	AF	1	[4:0]	WR/ RD	Receiving FIFO interrupt threshold low
PNC0	BE	1	[7:0]	WR/ RD	PNC[7:0]
PNC1	CE	1	[7:0]	WR/ RD	PNC[15:8]
PNC2	DE	1	[7:0]	WR/ RD	PNC[23:16]
PNC3	EE	1	[7:0]	WR/ RD	PNC[31:24]
PND0	BF	1	[7:0]	WR/ RD	PND[7:0]

Name	Address	Bank Select CI_I[1]	Index	Write/ Read	Functions
PND1	CF	1	[7:0]	WR/ RD	PND[15:8]
PND2	DF	1	[7:0]	WR/ RD	PND[23:16]
PND3	EF	1	[7:0]	WR/ RD	PND[31:24]

TABLE 30. *Control Registers*

12.14. Configuration Information Bits

The configuration information bits are used to set the various programmable parameters in the SSTM:

CI_I[0] *Should be set to LO.*

CI_I[3:2] *PLSL. Sets the width of the ESD window. CI_I[2] is LSB, CI_I[3] is MSB. See Application Section.*

PLSL	Width of ESD window
0	8 samples wide
1	10 samples wide
2	12 samples wide
3	14 samples wide

CI_I[4] *CNTLR. Sets the width of the "Central Region". (***NOTE***: CNTLR size must be ≤ PLSL size).See Application Section.*

CNTLR	Width of Central Region
0	10 samples wide
1	12 samples wide

CI_l[5] *ACC1RES. DPLL Accumulator 1 reset. See Application Section.*

ACC1RES	Accumulator1 Reset
0	ACC1 NOT reset.
1	Reset ACC1.

CI_l[7:6] *WSL. Sets the width of the "Detection Window". CI_l[6] is LSB, CI_l[7] is MSB. See Application Section.*

WSL	Width of Detection Window
0	4 samples wide
1	6 samples wide
2	8 samples wide
3	10 samples wide

CI_h[0] *M/SB. Set the SSTM to be a Master (HI) or Slave (LO). Note: must be set to LO in half-duplex data mode.*

CI_h[3:1] *T. Number of errors allowed in the UW. CI_h[1]is LSB, CI_h[3] is MSB.*

T	Allowable UW Errors
0	0 bit
1	1 bit
2	2 bits
3	3 bits
4-7	4 bits

CI_h[4] *LockSMon. Enables the Locking State Machine when set to HI (see Figure 14. for Locking State Machine state diagram).*

12.15. RF/IF Analog Interface

The SSTM interfaces with the RF/IF analog radio through the following pins: DI, MODOUT, PLLSW, and RFPWR.

DI is a CMOS-level input fed by the analog receiver. MODOUT is a tri-state output to the analog transmitter. It is in high-impedance state when the SSTM is in the receive mode (TXEN is LO). PLLSW and RFPWR are used respectively to switch the PLL of the analog radio and to power on/off the transmitter power amplifiers. The timings for the RFPWR and PLLSW are shown in Figure 19. and Figure 20. respectively.

The RFPWR switch timing is designed to avoid damage to the sensitive analog receiver. When switching from receive to transmit, RFPWR is delayed relative to TXEN (which can be used for switching the antenna between transmit and receive chain) to ensure that the receiver has been turned off before the transmitter is turned on. Similarly, when switching from transmit to receive, RFPWR is turned off first, before TXEN, to allow extra time for the transmitter to turn off prior to turning on the receiver circuits. Note that the RFPWR timing is valid for both full-duplex and half-duplex modes.

The BURST_CLK shown in Figure 19. is the burst rate clock which is 2.667 times the data rate in full-duplex operation and is equaled to the data rate in half-duplex operation. For example, for a master oscillator of 16.384 MHz, the full-duplex burst rate is 85.333 KHz. Thus, the RFPWR signal will be asserted one burst clock cycle or 11.72 µsec after TXEN assertion and will be de-asserted one burst clock cycle or 11.72 µsec prior to TXEN de-assertion.

The PLLSW timing shown is for the full-duplex mode; for half-duplex operation, PLLSW timing follows that of the RFPWR. The PLLSW signal is designed to switch the RF PLL when different frequencies are used for transmit and receive operation. In this instance, PLLSW is turned on right at the end of receive operation and prior to TXEN assertion to allow the RF PLL to stabilize. Similarly, the PLLSW changes to a LO as soon as transmission is finished and before receiving commences.

The PLLSW is asserted 28 burst clock cycles (duration of G2) prior to TXEN assertion and is de-asserted 1 burst clock cycle before TXEN is de-asserted in the full-duplex mode. Thus, for a master oscillator of 16.384 MHz (corresponding to a burst rate of 85.333 KHz or a burst period of 11.72 µsec), the PLLSW signal will be asserted 328 µsec (28*11.72=328) prior to TXEN assertion.

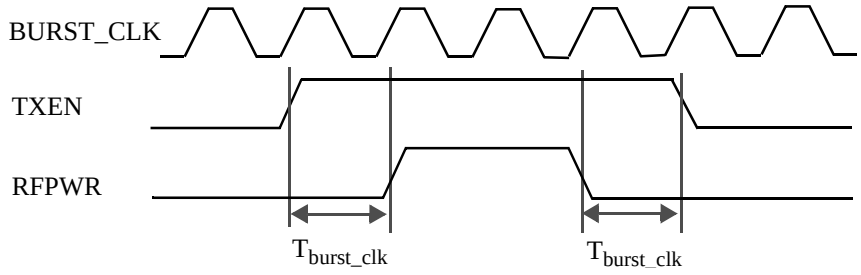


Figure 19. *RFPWR Timing*

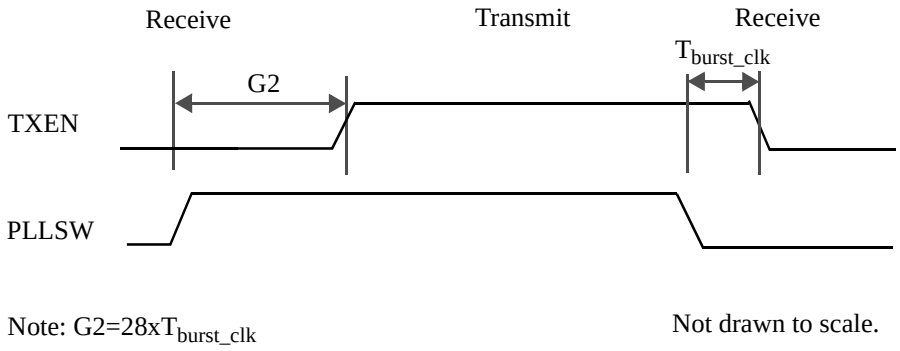


Figure 20. *PLLSW Timing (Full-Duplex Mode)*

12.16. Programming the SSTM

In the following, a brief discussion on programming the SSTM for the various modes of operation is presented. First, the loading of the PN sequences A, B, C, and D is required for operation in all modes.

The programming of the PLSL, CNTLR, ACC1RES, and WSL depend on several environmental and system related factors. For example, the sizing of PLSL and CNTLR windows involves trade-off considerations in the PLL's dynamic performance. In general, if smaller window size is used, the PLL will behave as if it has a smaller loop bandwidth with higher noise filtering but at the expense of slower dynamic response. If larger window size is used, the PLL will respond quicker dynamically but its performance will be degraded because more noise is allowed to enter the system. ***Please note that the width of the Central Zone must be smaller than or equal to the size of the ESD window*** (this means that the ESD window size should NOT be set to 8, it was included on the chip for testing purpose only).

Similarly, the width of the detection window, WSL, should be large enough to take advantage of multipath combining but should not be so large that excessive noise is allowed into the receiver causing receiver sensitivity degradation. In general, these parameters can be experimentally optimized for a particular environment. Otherwise, it is recommended that these parameters be programmed to values in the middle of the programmable range (for example, set PLSL to 12 samples wide, CNTLR to 10 samples wide, and WSL to 8 samples wide).

In the full-duplex or TDD mode, ACC1RES should normally be set to "0" for no ACC1 reset during each "freeze PLL" period. The only instance where ACC1RES should be set "1" to reset ACC1 is if the frequency offset between the transmitter and receiver is known to be very small. In this case, the performance of the PLL will be slightly enhanced if ACC1 is reset during each "freeze PLL" period. In half-duplex operation. ACC1RES should be set to "1" to always reset ACC1.

CI_h[0] is used to set the SSTM to be either a master or a slave. Typically the unit that initiates the signalling process should be programmed to become the master.

NOTE, for half-duplex operation, CI_h[0] MUST be set to LO (e.g., as a slave). There is no concept of master or slave in the half-duplex operation. Instead, the SSTM is keyed by the RTS_N bit enable to go into transmit mode. The SSTM stays in the receive mode otherwise.

The number of allowable errors in UW depends on the application. For example, applications that can tolerate a larger BER can usually allow more UW errors while still maintaining a reasonable communication link.

Finally, CI_h[4] enables or disables the Locking State Machine. The locking state machine when used in conjunction with the programmable allowable UW errors gives the system designer the flexibility to tailor the SSTM for a particular operating environment. Typically, by enabling the Locking State Machine and by allowing more UW errors, the SSTM will continue to operate normally even in a marginal communication link channel without repeatedly loosing lock and going into acquisition. The disadvantage is the corresponding increase in the data errors; for some critical applications, this might not be tolerable. In this case, the number of allowable UW errors can be reduced and the locking state machine turned off.

12.17. PN Sequence and UW Selection

Four 32-bit PN sequences and one 22-bit Unique Word (UW) are required for each spread spectrum communication device. Together, they can constitute a “security code” or “identification code” which can be use to distinguish different users as well as to provide privacy. In addition, the PN sequences and UW participate in the signal acquisition and burst synchronization processes. In order to ensure good system performance, the PN sequences and UW must be selected with some care. In the following, guidelines for choosing the PN sequences and UW are presented.

12.18. PN Sequence Selection

The four PN sequences are used to represent a di-bit symbol in the SSTM. In order to correctly decode the transmitted symbol at the receiver, the following principles should be followed when choosing the four PN sequences.

1. The four PN sequences should be orthogonal to each other. Two PN sequences A and B are orthogonal to each other if,

$$\sum_{i=0}^{31} a_i \cdot b_i = 0 \qquad \textbf{(EQ 5)}$$

   where PN sequence $A = [A_{31}, A_{30}, A_{29},\ldots, A_1, A_0]$, $a_i = -1$ for $A_i = 0$, and $a_i = 1$ for $A_i = 1$; similarly for B.
2. The PN sequences should be even, i.e., each sequence should have the same number of zeros and ones.
3. The PN sequences should not have more than four consecutive identical bits.

With these three criteria outlined above, it is possible to generate a large set of valid PN sequences. Two additional and optional criteria can be used to further identify PN sequences for reduced self- and cross-interference.

4. The auto-correlation side lobes of the PN sequences should be less than the auto-correlation of the main lobe by at least 4.
5. The cross-correlation of PN sequences between sets (one set being the four orthogonal PN sequences for one spread spectrum chip) should also be less than the auto-correlation of the main lobe by at least 4.

12.19. UW Selection

The UW is used in the receiver in full-duplex mode to establish synchronization. To avoid interference, the UW must be chosen such that it has good auto-correlation and cross-correlation properties. The auto-correlation of a sequence A denoted as S_N is defined as,

$$S_N = \sum_{i=0}^{L} a_i \cdot a_{i-N} \qquad \textbf{(EQ 6)}$$

where L is the length of the sequence A, $-L < N < L$, $N \neq 0$, $a_i = -1$ for $i < 0$, and $a_i = a_{i-22}$ for $i \geq L$. A "window" version of S_N can also be defined as per (EQ 6) with the exception that $0 < N < W$, where W is the window size. The desired auto-correlation property is that the maximum value of the auto-correlation S_N (with or without the window where the window size is the size of the detection window, WSL) is less than $L - 2 \times T$, where T is the allowable number of UW errors programmed into the SSTM.

The cross-correlation of two sequences A and B, denoted by R_N is defined as,

$$R_N = \sum_{i=0}^{L} a_i \cdot b_{i-N} \qquad \textbf{(EQ 7)}$$

where L is the length of the sequence, $0 \leq N < L$, and $b_i = b_{i+22}$ for $i < 0$. As before, the desirable cross-correlation property is that the maximum value of the cross-correlation R_N is less than $L - 2 \times T$, where L and T are as defined previously.

Requirements for choosing good UWs can be thus summarized by the following equations.

$$\begin{aligned} S_N &< L - 2 \times T \\ R_N &< L - 2 \times T \end{aligned} \qquad \textbf{(EQ 8)}$$

For example, when T is programmed to be 4, then S_N and R_N should be less than 14 since L is 22.

12.20. Generating the PN and UW sequences

There are many ways of generating the PN and UW sequences, including brute force search of the complete code space. A more efficient but probably not optimal way of generating the PN and UW sequences is to

1. Generate the M and Gold sequences of order N where

$$N = \log_2 L \qquad \textbf{(EQ 9)}$$

   and where L is the length of the PN or UW sequence.
2. Because M and Gold sequences are only $2^N - 1$ bits long, it is necessary to append an additional bit to these sequences so that they are 2^N bits long. The additional bit should be chosen such that the modified M or Gold sequence is even.
3. Apply the criteria outlined in the previous sections to pick out the good set of PN and UW sequences.

13. Power-Saving Modes

13.1. Overview

The MCU (Micro-Controller Unit) provides two kinds of methods to save power consumption. The first method is to stop clock operations partially or wholly. The result is to reduce switching current of CMOS. The other is to make pin status High-Z (impedance). This feature will remove static current through resistors on board. Two methods (clock control scheme and pin control scheme) can be combined together. The Power-Saving Modes are controlled by the PSCR (Power-Saving Control Register) that is an SFR (Special Function Register) mapped on memory address map. Any kind of instruction, whose destination is the PSCR, can change the PSCR content. Thus, the transitions of modes among normal and Power-Saving modes can be controlled by software. In some cases, when CPU stops, hardware signals such like interrupts, an IRQ1 Pin Event, and reset make the device change modes.

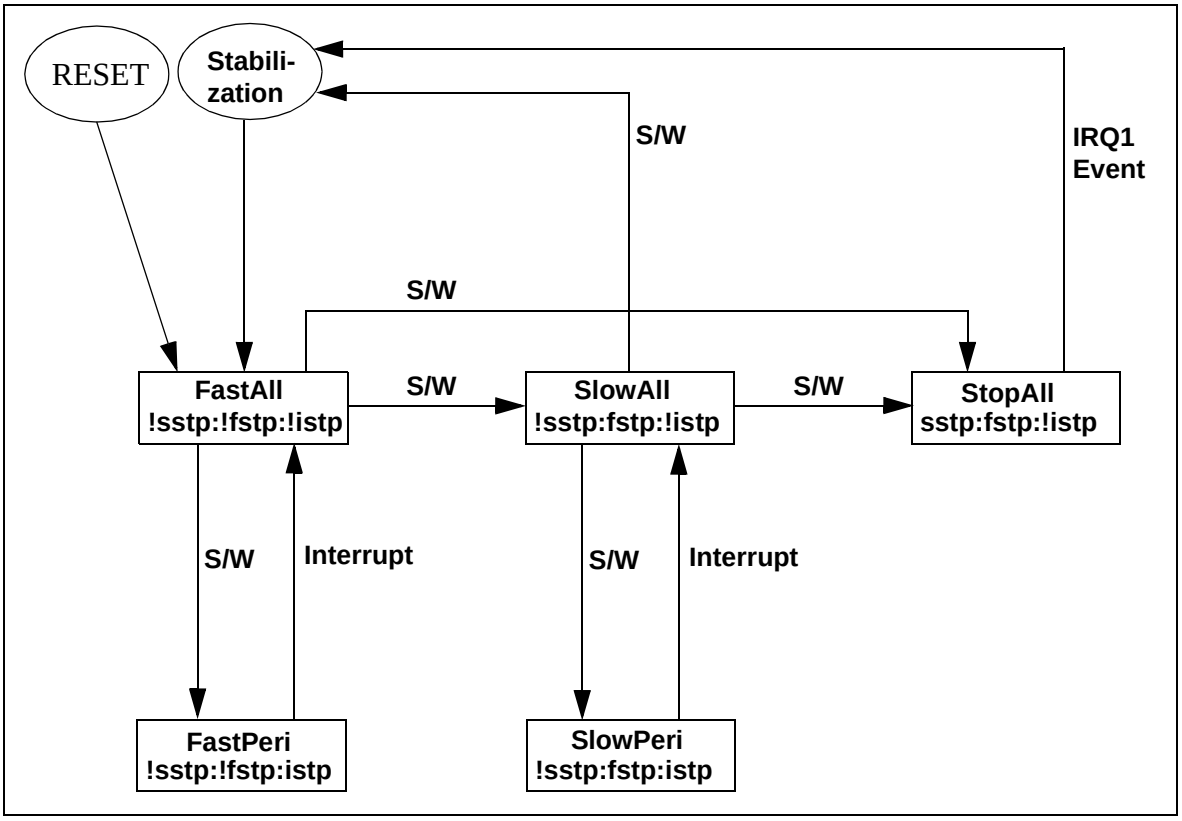


Figure 21. *Power Saving Modes & Transitions*

13.2. Mode Definition and Transition

There are five Clocking Modes, i.e. FastAll, SlowAll, StopAll, FastPeri, and SlowPeri modes. These are controlled by PSCR[2:0] (Power Saving Register bit 2 to 0). According to these bits, it is decided for the Fast Oscillator (up to 24MHz at 5V or 12Mhz at 3V), Slow Oscillator (32.768KHz), and CPU operation or Instruction execution to run or stop.

There are two Pin State Modes, i.e. NorPin (Normal Pin operation mode) and HIZ (High-Z mode). The bit-4 of PSCR(HIZ) selects a mode between them.

In any mode of Clocking Modes, any Pin State Mode can be selected.

It is conceptually possible to make a new clocking power-saving mode because three bits of PSCR are assigned to select it. But, the other cases except listed five modes are never proven. If the user needs a new clocking Power-Saving mode other than listed five, to test and prove it is user's responsibility. We also recommend that only the listed mode transition is used. For other transitions than recommended transitions, user must carefully test it. When Clocking Power-Saving Mode is changed by Software, the next two instructions should be NOP (No Operation).

PSCR bit	bit-2	bit-1	bit-0
Bit Name	SSTP	FSTP	ISTP
FastAll mode	0	0	0
SlowAll mode	0	1	0
StopAll mode	1	1	0
FastPeri mode	0	0	1
SlowPeri mode	0	1	1
Another Cases	Not Proven		

TABLE 31. *Clocking Power-Saving Mode Definition and PSCR*

PSCR bit	bit-4
Bit Name	HIZ
NorPin mode	0
HIZ mode	1

TABLE 32. *Clocking Power-Saving Mode Definition and PSCR*

13.3. FastAll mode (a Clocking Mode)

This is a normal operation mode. All of MCU may operate normally. Both Fast (up to 24MHz) and Slow (32.768KHz) oscillators generate clock signals. The Fast Clock is used as a System Clock to control the CPU and Peripherals.

In the FastAll mode, the content of the PSCR is xxxxx000B (x: don't care, B:binary). After Reset from any mode, the MCU goes into the FastAll mode.

The next available Clocking Modes are the SlowAll mode(PSCR[2:0] = 010B), FastPeri mode(PSCR[2:0] = 001B), and StopAll mode(PSCR[2:0] = 110B). All three mode transitions from FastAll mode are performed by software.

13.4. SlowAll mode (a Clocking Mode)

This is a slow operation mode. All of MCU may operate with low speed(32.768KHz). Fast clock generator is halted. Slow (32.768KHz) oscillators generate clock signals. The Slow Clock is used as a System Clock to control the CPU and Peripherals.

In the SlowAll mode, the content of the PSCR is xxxxx010B (x: don't care, B:binary). After Reset from this mode, the MCU goes into the FastAll mode.

Next available Clocking Modes are the FastAll mode, StopAll mode, and SlowPeri mode. All three mode transitions from SlowAll mode are performed by software.

13.5. StopAll mode (a Clocking Mode)

This is a perfect stop mode. All of MCU stop to operate. Both Fast and Slow clock generators are halted.

In the StopAll mode, the content of the PSCR is xxxxx110B (x: don't care, B:binary). After Reset from this mode the MCU goes into the FastAll mode. In this transition case, most SFR data will be initialized. By an External Event signal(IRQ1), the MCU goes into FastAll mode. In this transition case, all SFR and RAM data will be retained.

13.6. FastPeri mode (a Clocking Mode)

This is a fast Peripheral operation mode. The CPU is halted. All of Peripheral may operate with fast speed. Fast and Slow oscillators generate clock signals. The Fast Clock is used as a System Clock to control the Peripherals.

In the FastPeri mode, the content of the PSCR is xxxxx001B (x: don't care, B:binary). After Reset from this mode, the MCU goes into the FastAll mode.

Next available Clocking Modes is the FastAll mode only. The mode transition from FastPeri mode is performed by an interrupt. In this mode CPU SFRs (Special Function Registers such like PSW, ACC, and etc.) and RAM data will be retained.

13.7. SlowPeri mode (a Clocking Mode)

This is a slow Peripheral operation mode. The CPU is halted. All of Peripheral may operate with slow speed (32.768KHz). Fast oscillator is halted. Slow oscillators generate clock signals. The Slow Clock is used as a System Clock to control the Peripherals.

In the SlowPeri mode, the content of the PSCR is xxxxx011B (x: don't care, B:binary). After Reset from this mode, the MCU goes into the FastAll mode.

Next available Clocking Modes is the SlowAll mode. The mode transition from SlowPeri mode is performed by an interrupt. In this mode CPU SFRs (Special Function Registers such like PSW, ACC, and etc.) and RAM data will be retained.

13.8. NorPin mode (a Pin State Mode)

This is a normal operation mode. All of pin state can be controlled by software or hardware. This is an initial state after Reset. In the NorPin mode, the content of the PSCR is xxx0xxxxB (x: don't care, B:binary). The mode transition from NorPin mode to HIZ mode is performed by software. The NorPin mode can be combined with any kinds of Clocking Power-Saving modes.

13.9. HIZ mode (a Pin State Mode)

This is a Pin State Power-Saving mode. All of normal port pin state become high-impedance state to remove static current. Thus, port pin state can not be controlled by software or other hardware. In the HIZ mode, the content of the PSCR is xxx1xxxxB (x: don't care, B:binary). The mode transition from HIZ mode to NorPin mode is performed by software and reset. The HIZ mode can be combined with any kinds of Clocking Power-Saving modes. In the HIZ mode, the port register content will be retained if software or hardware don't change them. For PORT-0 to 4, each port will be controlled by ZCR register. But, HIZ bit directly controls PORT-4. Please, refer PORT block specification chapter.

Block	FastAll	SlowAll	StopAll	FastPeri	SlowPeri
Fast Oscillator	Run	Stop	Stop	Run	Stop
Slow Oscillator	Run	Run	Stop	Run	Run
System Clock	from Fast OSC	from Slow OSC	Stop	from Fast OSC	from Slow OSC
CPU Clock	from Fast OSC	from Slow OSC	Stop	Stop	Stop
Peripheral Clock	from Fast OSC	from Slow OSC	Stop	from Fast OSC	from Slow OSC
Instruction	Run(fast)	Run(slow)	Stop	Stop	Stop
CPU SFR	Run(fast)	Run(slow)	Retained	Retained	Retained
RAM	Run(fast)	Run(slow)	Retained	Retained	Retained
Peripheral	Run(fast)	Run(slow)	Stop	Run(fast)	Run(slow)
Peri SFR	Run(fast)	Run(slow)	Retained	Run(fast)	Run(slow)

TABLE 33. *Device States in Power-Saving Modes*

13.10. Power-Saving Control Register (PSCR)

Address	97H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME			FTST	HIZ	NOTS	SSTP	FSTP	ISTP
Definition			Fast Stabilization Time for Test	High Impedance	Not Use Slow Clock	Slow clock SToP	Fast clock SToP	CPU clock SToP
Reset Value			0	0	0	0	0	0
Read/ Write by Software			R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware						0 by An IRQ1 Event	0 by An IRQ1 Event	0 by Interrupt

TABLE 34. *Summary of PSCR*

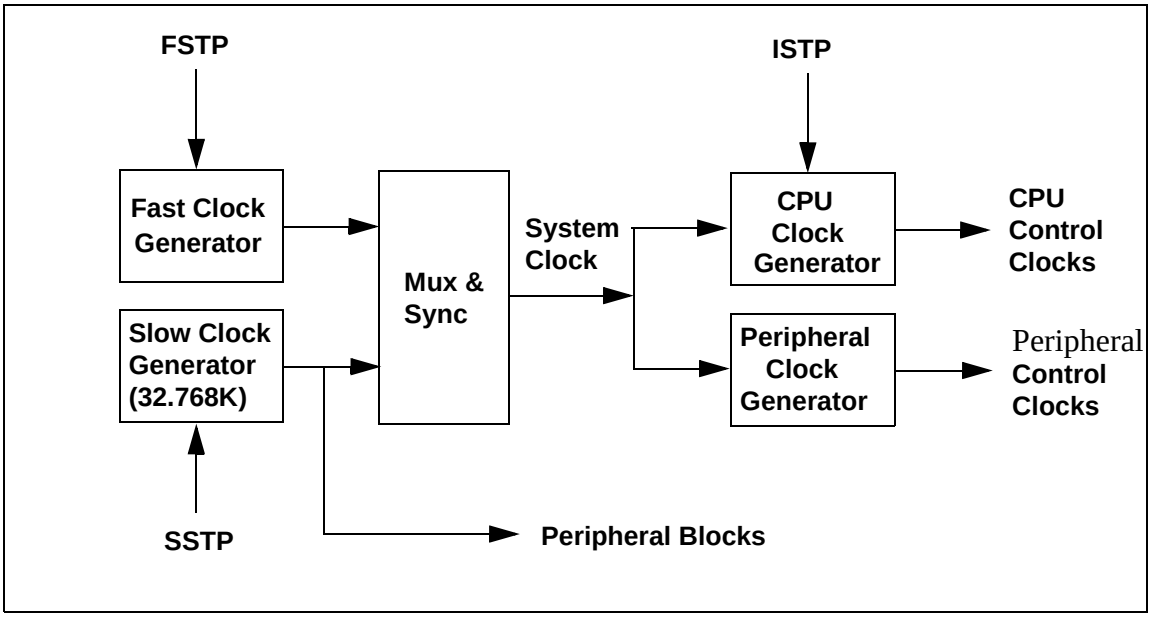


Figure 22. *Clock System & Power-saving Control*

Name	Bit	Definition
FSEL[1:0]	7 to 6	Reserved
FTST	5	Fast Stabilization Time for Test [Bit Status: 0] When the StopAll mode is released by an IRQ1 Event, fast oscillator stabilization time will be 19.4688 mSec (1/10MHz x (2^{18-1}+2^{17-1})). [Bit Status: 1] When the StopAll mode is released by an IRQ1 Event, fast oscillator stabilization time will be 6.4 uSec (1/10MHz x 2^{7-1}).
HIZ	4	PIN State control bit [Bit Status: 0] Normal Pin state. [Bit Status: 1] All normal I/O pins become High-Z (impedance) state.
NOTS	3	Not Use Slow Oscillator bit [Bit Status: 0] Slow oscillator (32.768KHz) is used. [Bit Status: 1] Slow oscillator is not used. The pin XSIN is connected to ground.

SSTP	2	Slow clock SToP bit [Bit Status: 0] Slow oscillator (32.768KHz) operates. [Bit Status: 1] Slow oscillator stops.
FSTP	1	Fast clock SToP bit [Bit Status: 0] Fast oscillator (up to 24MHz) operates and drives all CPU and peripherals. [Bit Status: 1] Fast oscillator stops. Slow oscillator (32.768KHz) may or may not drive all CPU and peripherals.
ISTP	0	Instruction SToP bit [Bit Status: 0] CPU control clocks drive CPU. Thus, instructions may be executed its operations normally. [Bit Status: 1] CPU control clocks stop. Thus, CPU stops to execute instructions.

TABLE 35. *The Definition of Each PSCR bit*

13.11. Stop Release Register (SREL)

Address	96H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	SREL1	SREL0
Definition							Stop Release bit-1	Stop Release bit-0
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R	R	R	R	R	R	R/W	R/W
Write by Hardware								

TABLE 36. *Summary of SREL*

Name	Bit	Definition
SREL[1:0]	1 to 0	Stop Release Bit-1 to 0 [Bit Status: 01] IRQ1 Low Level enabled [Bit Status: 10] IRQ1 High Level enabled [Other Cases] StopAll Mode can not be released by IRQ1 pin event, can be released by only Reset. Stop Release by IRQ1 should be enabled just before going into the StopAll mode. If it is enabled in other mode, unexpected mode transition will be happened.

TABLE 37. *The Definition of Each SREL bit*

14. Port Definitions

14.1. Overview

In the SS1102C, there are two groups of ports, Port-0 to 3 and Port-4. Each port group has different configuration. Thus, the user should carefully read the port specification.

The output drivers of Port-0 and 2, and the input buffer of Port-0 may be used in accesses to external memory. In this application, Port-0 outputs the low byte of the external memory address, time-multiplexed with the byte being written or read. Port-2 outputs the high byte of the external memory address when the address is 16-bits wide. Otherwise the Port-2 pins continue to emit the P2 SFR content.

The Port-1 pins and Port-3 pins are multifunctional. They are not only port pins, but also serve the functions of various special features as listed below:

PORT BIT	ALTERNATE FUNCTION
P1.4	LBD
P1.5	SPIO
P1.6	SPIIN
P1.7	SPICLK
P3.0	CPTRU2
P3.1	CPTRU1

PORT BIT	ALTERNATE FUNCTION
P3.2	CPTRD1
P3.3	TXEN
P3.4	MODOUT
P3.5	DI
P3.6	IRQ1, NWR (External Data Memory Write Strobe)
P3.7	IRQ2, NRD (External Data Memory Read Strobe)

TABLE 38. *Port 1 and Port 3 Functions*

The alternate functions for NWR, NRD can only be activated if the corresponding port bit latch (P3.6, P3.7) in the port SFR contains a 1 (high, initial value). Otherwise the port pin is stuck at 0 (low).

14.2. Read-Modify-Write Feature

Some instructions, that read a port (Port-0 to 3 only), read the latch and others read pin. The instructions that read the latch rather than the pin are the ones that read a value, possibly change it, and then rewrite it to the latch. These are called “read-modify-write” instructions. The instructions listed below are read-modify-write instructions. When the destination operand is a port, or a port bit, these instructions read the latch than the pin:

INSTRUCTIONS	DESCRIPTION
ANL	logical AND, e.g. ANL P0, A
ORL	logical OR, e.g. ORL P1, A
XRL	logical EX-OR, e.g. XRL P2, A
JBC	jump if bit = 1 and clear bit, e.g. JBC P3.0, LABEL
CPL	complement bit, e.g. CPL P3.1
INC	increment, e.g. INC P0
DEC	decrement, e.g. DEC P1
DJNZ	decrement and jump if not zero, e.g. DJNZ P0, LABEL
MOV PX.Y, C	move carry bit to bit Y of Port-X
CLR PX.Y	clear bit Y of Port-X

INSTRUCTIONS	DESCRIPTION
SET PX.Y	set bit Y of Port-X

TABLE 39. *Read-Modify-Write Instructions*

It is not obvious that the last three instructions in this list are read-modify-instructions, but they are. They read the port byte, all 8 bits, modify the addressed bit, then write the new byte back to the latch. The reason that read-modify-write instructions are directed to the latch than the pin is to avoid a possible misinterpretation of the voltage level at the pin. For example, a port bit might be used to drive the base of the transistor. When a 1 is written to the bit, the transistor is turn on. If the CPU then reads the same port bit at the pin rather than the latch, it will read the base voltage of the transistor and interpret it as a 0. Reading the latch rather than the pin will return the correct value of 1.

14.3. PCR (Port Control Register)

Address	9AH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME			P4LATIN	P35INEN	PCR3	PCR2	PCR1	PCR0
Definition	Reserved bit	Reserved bit	Port-4 Input Select 0: Pin In 1:Latch In	Port-3.5 Input Enable bit 0 : Dis-able (DIA) 1: P3.5 Input Enable (DI). This bit is the DI/DIA selection bit	Port-3 I/O Control bit 0: CPU Control 1: P3IO Control	Port-2 I/O Control bit 0: CPU Control 1: P2IO Control	Port-1 I/O Control bit 0: CPU Control 1: P1IO Control	Port-0 I/O Control bit 0: CPU Control 1: P0IO Control
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 40. *Definition of PCR*

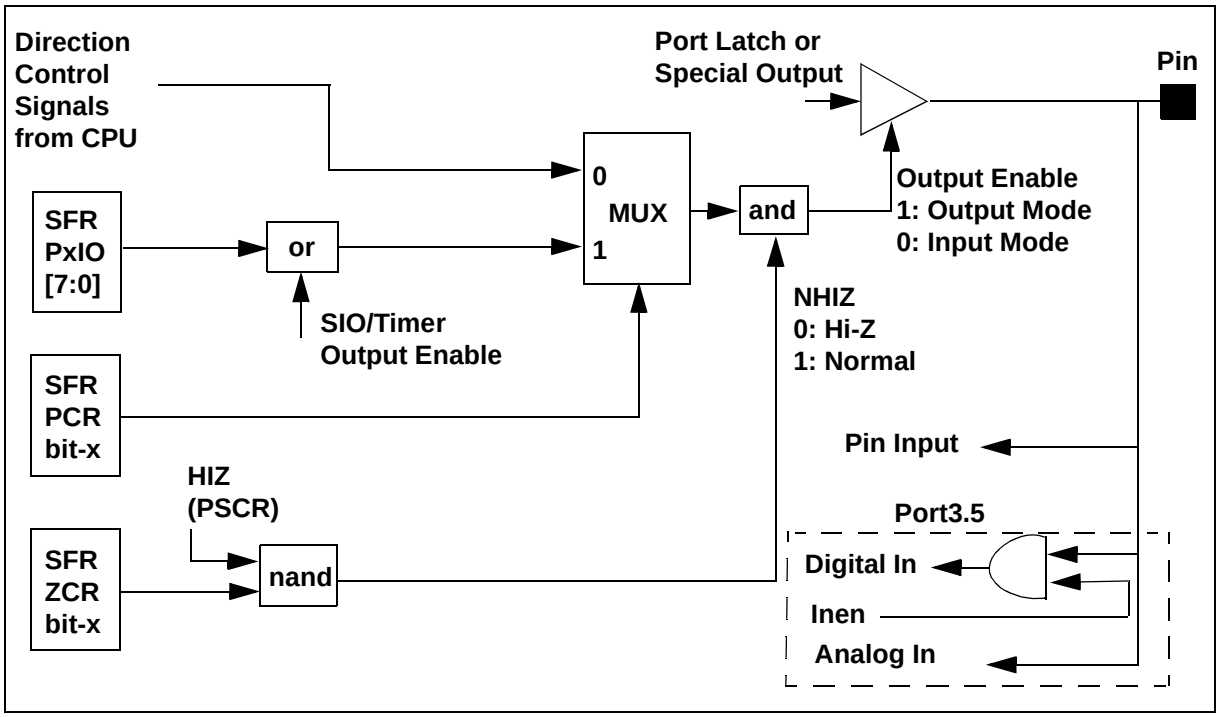


Figure 23. *Pcr & Port Direction Control: Port-0 To 3*

14.4. ZCR (Hi-Z Control Register)

Address	9BH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME				ZCR4	ZCR3	ZCR2	ZCR1	ZCR0
Definition	Reserved bit	Reserved bit	Reserved bit	Port-4 Hi-Z Control bit 0: Disable 1: Hi-Z Enable	Port-3 Hi-Z Control bit 0: Disable 1: Hi-Z Enable	Port-2 Hi-Z Control bit 0: Disable 1: Hi-Z Enable	Port-1 Hi-Z Control bit 0: Disable 1: Hi-Z Enable	Port-0 Hi-Z Control bit 0: Disable 1: Hi-Z Enable
Reset Value	0	0	0	0	0	0	0	0

Read/Write by Software	R	R	R	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 41. Definition of ZCR

14.5. PAD Cell

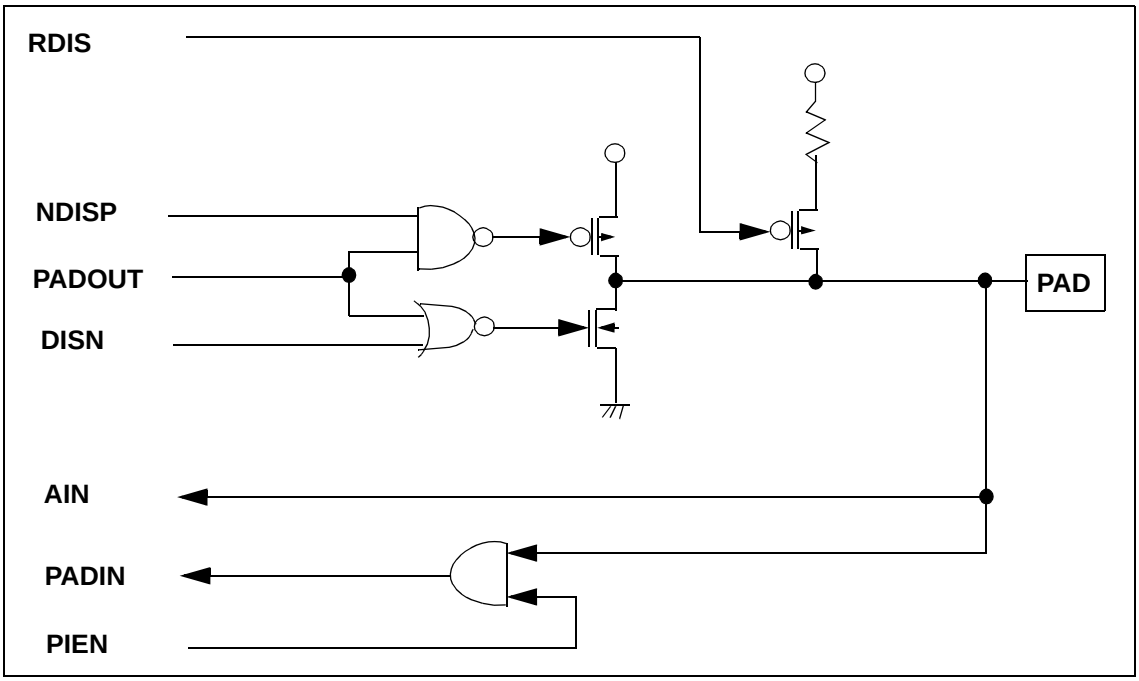


Figure 24. *Pad Cell*

14.6. P0 (Port-0)

Then the Pin-0.0 to 0.7 has two functional modes, A/D Mode (Low Address/Data Mode) and Port Mode (Bi-directional Port Mode). Initial Status is A/D Mode because PCR0 (Port Control Register Bit-0) is 0 (low). In Port Mode, the direction of Port-0 can be decided bit-by-bit. In A/D Mode, writing to Port-0 is prohibited. The read policy in Port Mode follows that described before. Read-Modify-Write instructions read the port latch than the pin.

To use Pin-0.0 to 0.7 as a normal input port, PCR0 should be high. If PCR0 is high, the direction of each Port-0 pin is decided by each P0IO bit. If the P0IO bit-x is high, Port-0.x is output mode. If low, Port-0.x is input mode. Initial status is input mode because P0IO (Port-0 I/O Direction Register) is reset to 00000000B. In input mode, pull-up resistor can be connected by P0RD (Port-0 Resistor Disable Register). If the P0RD bit-x

is high, Port-0.x pull-up resistor is disconnected. If low, Port-0.x pull-up resistor is connected. Initial status of P0RD is 00000000B, thus pull-up resistor is connected. If ZCR0 (High-Z Control Register Bit-0) is high, High-Z control is enabled. At this time, if HIZ bit in the PSCR (Power Saving Control Register) becomes high, pull-up resistor is disconnected and pin status becomes high impedance.

To use Pin-0.0 to 0.7 as a normal output port, PCR0 should be high. If the P0IO bit-x is high, Port-0.x is output mode. In output mode, pull-up resistor is not automatically disconnected regardless P0RD bit-x. If ZCR0 (High-Z Control Register Bit-0) is high, High-Z control is enabled. At this time, if HIZ bit in the PSCR (Power Saving Control Register) becomes high, pin status becomes high impedance.

To use Pin-0.0 to 0.7 as an A/D input/output port, PCR0 should be low (reset value). If PCR0 is low, the direction of each Port-0 pin is controlled by CPU. In input mode, pull-up resistor can be connected by P0RD (Port-0 Resistor Disable Register). In output mode, pull-up resistor is not automatically disconnected regardless P0RD bit-x. For the A/D Mode, the ZCR0 must be low (reset value) not to support Hi-Z state.

When P0NOPNx = 0, Port-0.x output buffer will be CMOS Push-Pull. But, P0NOPNx = 1, Port-0.x output buffer will become N-channel Open Drain.

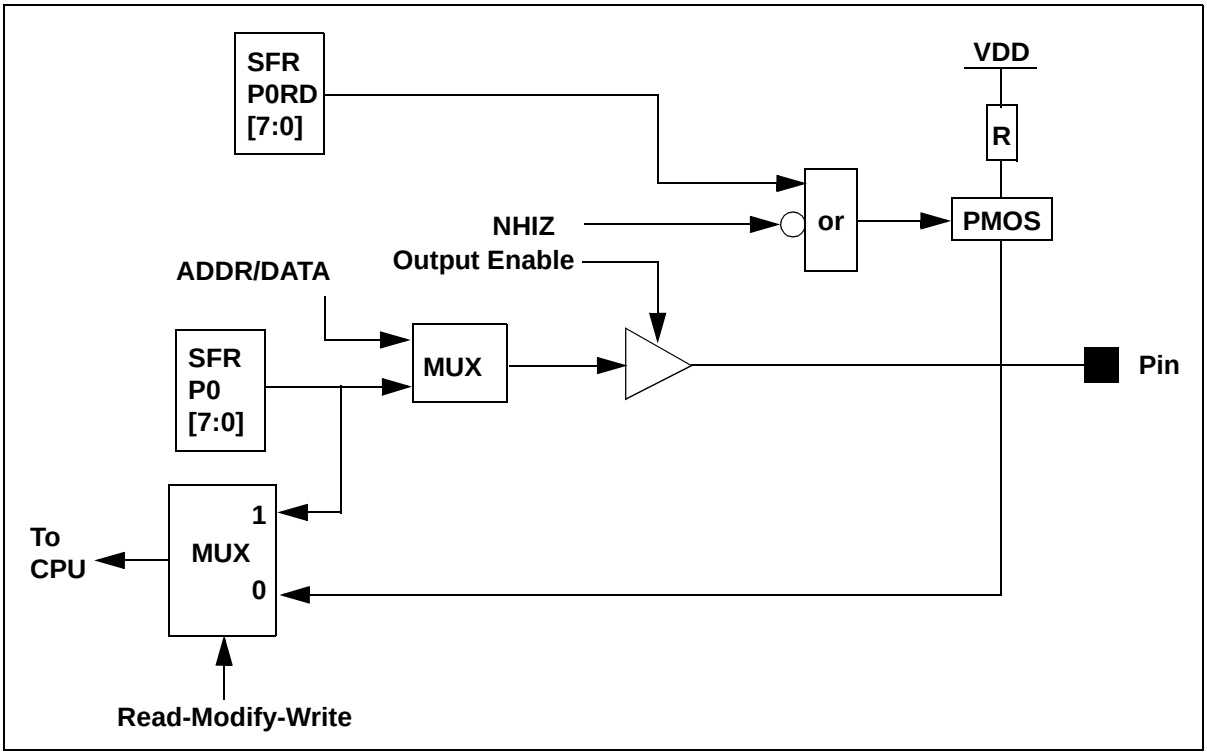


Figure 25. *PORT-0*

Address	80H							
Bit Addr.	87H	86H	85H	84H	83H	82H	81H	80H
BIT	7	6	5	4	3	2	1	0
NAME	P07	P06	P05	P04	P03	P02	P01	P00
Definition	Port-0 Pin or Latch bit-7	Port-0 Pin or Latch bit-6	Port-0 Pin or Latch bit-5	Port-0 Pin or Latch bit-4	Port-0 Pin or Latch bit-3	Port-0 Pin or Latch bit-2	Port-0 Pin or Latch bit-1	Port-0 Pin or Latch bit-0
Reset Value	1	1	1	1	1	1	1	1
Read/Write by Software	R/W RMW Ins: P07 Latch Read Other Ins: P0.7 Pin Read	R/W RMW Ins: P06 Latch Read Other Ins: P0.6 Pin Read	R/W RMW Ins: P05 Latch Read Other Ins: P0.5 Pin Read	R/W RMW Ins: P04 Latch Read Other Ins: P0.4 Pin Read	R/W RMW Ins: P03 Latch Read Other Ins: P0.3 Pin Read	R/W RMW Ins: P02 Latch Read Other Ins: P0.2 Pin Read	R/W RMW Ins: P01 Latch Read Other Ins: P0.1 Pin Read	R/W RMW Ins: P00 Latch Read Other Ins: P0.0 Pin Read
Write by Hardware								

TABLE 42. *Definition of P0*

Address	AAH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P0IO7	P0IO6	P0IO5	P0IO4	P0IO3	P0IO2	P0IO1	P0IO0
Definition	Port-0 I/O Control Register bit-7 0: Input Mode 1: Output Mode	Port-0 I/O Control Register bit-6 0: Input Mode 1: Output Mode	Port-0 I/O Control Register bit-5 0: Input Mode 1: Output Mode	Port-0 I/O Control Register bit-4 0: Input Mode 1: Output Mode	Port-0 I/O Control Register bit-3 0: Input Mode 1: Output Mode	Port-0 I/O Control Register bit-2 0: Input Mode 1: Output Mode	Port-0 I/O Control Register bit-1 0: Input Mode 1: Output Mode	Port-0 I/O Control Register bit-0 0: Input Mode 1: Output Mode
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 43. Definition of P0IO

Address	ABH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P0RD7	P0RD6	P0RD5	P0RD4	P0RD3	P0RD2	P0RD1	P0RD0
Definition	Port-0 Resistor Disable Register bit-7 0: Resistor Enable 1: Disable	Port-0 Resistor Disable Register bit-6 0: Resistor Enable 1: Disable	Port-0 Resistor Disable Register bit-5 0: Resistor Enable 1: Disable	Port-0 Resistor Disable Register bit-4 0: Resistor Enable 1: Disable	Port-0 Resistor Disable Register bit-3 0: Resistor Enable 1: Disable	Port-0 Resistor Disable Register bit-2 0: Resistor Enable 1: Disable	Port-0 Resistor Disable Register bit-1 0: Resistor Enable 1: Disable	Port-0 Resistor Disable Register bit-0 0: Resistor Enable 1: Disable
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 44. *Definition of P0RD*

Address	ABH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P0NOPN7	P0NOPN6	P0NOPN5	P0NOPN4	P0NOPN3	P0NOPN2	P0NOPN1	P0NOPN0
Definition	Port-0 Output Buffer bit-7 0: CMOS Push_Pull 1: N-ch open drain	Port-0 Output Buffer bit-6 0: CMOS Push_Pull 1: N-ch open drain	Port-0 Output Buffer bit-5 0: CMOS Push_Pull 1: N-ch open drain	Port-0 Output Buffer bit-4 0: CMOS Push_Pull 1: N-ch open drain	Port-0 Output Buffer bit-3 0: CMOS Push_Pull 1: N-ch open drain	Port-0 Output Buffer bit-2 0: CMOS Push_Pull 1: N-ch open drain	Port-0 Output Buffer bit-1 0: CMOS Push_Pull 1: N-ch open drain	Port-0 Output Buffer bit-0 0: CMOS Push_Pull 1: N-ch open drain
Reset Value	0	0	0	0	0	0	0	0

Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 45. *Definition of P0NOPN*

14.7. P2 (Port-2)

The Pins-2.0 to 2.7 have two functional modes, ADDR Mode (High Address Mode) and Port Mode (Bi-directional Port Mode). Initial Status is ADDR Mode because PCR2 (Port Control Register Bit-2) is 0 (low). In Port Mode, the direction of Port-2 can be decided bit-by-bit. The read policy in Port Mode follows that described before. Read-Modify-Write instructions read the port latch than the pin.

To use Pin-2.0 to 2.7 as a normal input port, PCR2 should be high. If PCR2 is high, the direction of each Port-2 pin is decided by each P2IO bit. If the P2IO bit-x is high, Port-2.x is output mode. If low, Port-2.x is input mode. Initial status is input mode because P2IO (Port-2 I/O Direction Register) is reset to 00000000B. In input mode, pull-up resistor can be connected by P2RD (Port-2 Resistor Disable Register). If the P2RD bit-x is high, Port-2.x pull-up resistor is disconnected. If low, Port-2.x pull-up resistor is connected. Initial status of P2RD is 00000000B, thus pull-up resistor is connected. If ZCR2 (High-Z Control Register Bit-2) is high, High-Z control is enabled. At this time, if HIZ bit in the PSCR (Power Saving Control Register) becomes high, pull-up resistor is disconnected and pin status becomes high impedance.

To use Pin-2.0 to 2.7 as a normal output port, PCR2 should be high. If the P2IO bit-x is high, Port-2.x is output mode. In output mode, pull-up resistor is not automatically disconnected regardless P2RD bit-x. If ZCR2 (High-Z Control Register Bit-2) is high, High-Z control is enabled. At this time, if HIZ bit in the PSCR (Power Saving Control Register) becomes high, pin status becomes high impedance.

To use Pin-2.0 to 2.7 as an ADDR output port, PCR2 should be low (reset value). If PCR2 is low, the direction of each Port-2 pin is controlled by CPU. In output mode, pull-up resistor is not automatically disconnected regardless P2RD bit-x. For the ADDR Mode, the ZCR2 must be low (reset value) not to support Hi-Z state.

When P2NOPNx = 0, Port-2.x output buffer will be CMOS Push-Pull. But, P2NOPNx = 1, Port-2.x output buffer will become N-channel Open Drain.

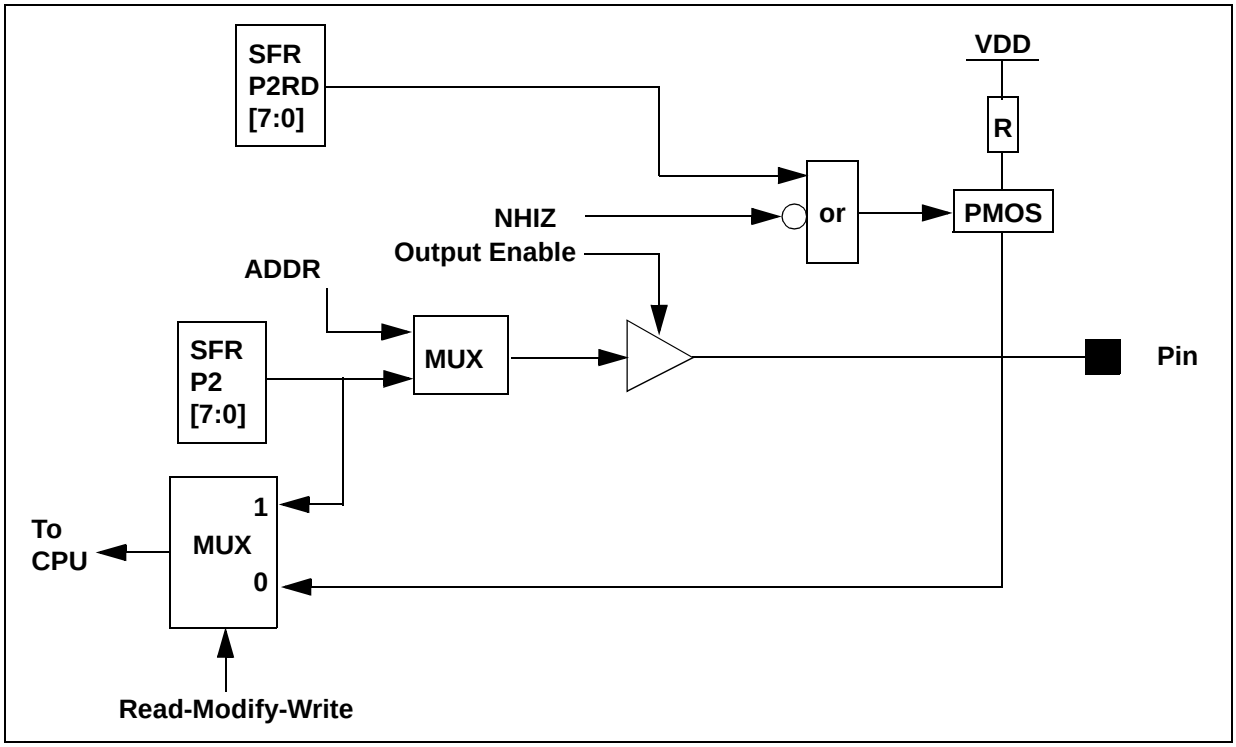


Figure 26. *PORT-2*

Address	A0H							
Bit Addr.	A7H	A6H	A5H	A4H	A3H	A2H	A1H	A0H
BIT	7	6	5	4	3	2	1	0
NAME	P27	P26	P25	P24	P23	P22	P21	P20
Definition	Port-2 Pin or Latch bit-7	Port-2 Pin or Latch bit-6	Port-2 Pin or Latch bit-5	Port-2 Pin or Latch bit-4	Port-2 Pin or Latch bit-3	Port-2 Pin or Latch bit-2	Port-2 Pin or Latch bit-1	Port-2 Pin or Latch bit-0
Reset Value	1	1	1	1	1	1	1	1
Read/Write by Software	R/W RMW Ins: P27 Latch Read Other Ins: P2.7 Pin Read	R/W RMW Ins: P26 Latch Read Other Ins: P2.6 Pin Read	R/W RMW Ins: P25 Latch Read Other Ins: P2.5 Pin Read	R/W RMW Ins: P24 Latch Read Other Ins: P2.4 Pin Read	R/W RMW Ins: P23 Latch Read Other Ins: P2.3 Pin Read	R/W RMW Ins: P22 Latch Read Other Ins: P2.2 Pin Read	R/W RMW Ins: P21 Latch Read Other Ins: P2.1 Pin Read	R/W RMW Ins: P20 Latch Read Other Ins: P2.0 Pin Read

Write by Hardware								

TABLE 46. *Definition of P2*

Address	CAH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P2IO7	P2IO6	P2IO5	P2IO4	P2IO3	P2IO2	P2IO1	P2IO0
Definition	Port-2 I/O Control Register bit-7 0: Input Mode 1: Output Mode	Port-2 I/O Control Register bit-6 0: Input Mode 1: Output Mode	Port-2 I/O Control Register bit-5 0: Input Mode 1: Output Mode	Port-2 I/O Control Register bit-4 0: Input Mode 1: Output Mode	Port-2 I/O Control Register bit-3 0: Input Mode 1: Output Mode	Port-2 I/O Control Register bit-2 0: Input Mode 1: Output Mode	Port-2 I/O Control Register bit-1 0: Input Mode 1: Output Mode	Port-2 I/O Control Register bit-0 0: Input Mode 1: Output Mode
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 47. *Definition of P2IO*

Address	CBH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P2RD7	P2RD6	P2RD5	P2RD4	P2RD3	P2RD2	P2RD1	P2RD0
Definition	Port-2 Resistor Disable Register bit-7 0: Resistor Enable 1: Disable	Port-2 Resistor Disable Register bit-6 0: Resistor Enable 1: Disable	Port-2 Resistor Disable Register bit-5 0: Resistor Enable 1: Disable	Port-2 Resistor Disable Register bit-4 0: Resistor Enable 1: Disable	Port-2 Resistor Disable Register bit-3 0: Resistor Enable 1: Disable	Port-2 Resistor Disable Register bit-2 0: Resistor Enable 1: Disable	Port-2 Resistor Disable Register bit-1 0: Resistor Enable 1: Disable	Port-2 Resistor Disable Register bit-0 0: Resistor Enable 1: Disable

Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 48. *Definition of P2RD*

Address	C9H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P2NOPN7	P2NOPN6	P2NOPN5	P2NOPN4	P2NOPN3	P2NOPN2	P2NOPN1	P2NOPN0
Definition	Port-2 Output Buffer bit-7 0: CMOS Push_Pull 1: N-ch open drain	Port-2 Output Buffer bit-6 0: CMOS Push_Pull 1: N-ch open drain	Port-2 Output Buffer bit-5 0: CMOS Push_Pull 1: N-ch open drain	Port-2 Output Buffer bit-4 0: CMOS Push_Pull 1: N-ch open drain	Port-2 Output Buffer bit-3 0: CMOS Push_Pull 1: N-ch open drain	Port-2 Output Buffer bit-2 0: CMOS Push_Pull 1: N-ch open drain	Port-2 Output Buffer bit-1 0: CMOS Push_Pull 1: N-ch open drain	Port-2 Output Buffer bit-0 0: CMOS Push_Pull 1: N-ch open drain
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 49. *Definition of P2NOPN*

14.8. P1 (Port-1)

The Pin-1.4 to 1.7 has two functional modes, special output Mode (SPIO, SPICLK) and Port Mode (Bi-directional Port Mode). Initial Status of PCR1 (Port Control Register Bit-1) is 0 (low). The PCR1 must be set to high for any case. In Port Mode, the direction of Port-1 can be decided bit-by-bit. The read policy in Port Mode follows that described before. Read-Modify-Write instructions read the port latch than the pin.

To use Pin-1.0 to 1.7 as a normal input port, PCR1 should be high. If PCR1 is high, the direction of each Port-1 pin is decided by each P1IO bit. If the P1IO bit-x is high, Port-1.x is output mode. If low, Port-1.x is input mode. Initial status is input mode because P1IO (Port-1 I/O Direction Register) is reset to 00000000B. In input mode, pull-up resistor can be connected by P1RD (Port-1 Resistor Disable Register). If the P1RD bit-x

is high, Port-1.x pull-up resistor is disconnected. If low, Port-1.x pull-up resistor is connected. Initial status of P1RD is 00000000B, thus pull-up resistor is connected. If ZCR1 (High-Z Control Register Bit-1) is high, High-Z control is enabled. At this time, if HIZ bit in the PSCR (Power Saving Control Register) becomes high, pull-up resistor is disconnected and pin status becomes high impedance. In input mode, P1.4/LBD, P1.6/SPIIN, and P1.7/SPICLK can be used LBD or SPI input pins.

To use Pin-1.0 to 1.7 as a normal output port, PCR1 should be high. If the P1IO bit-x is high, Port-1.x is output mode. In output mode, pull-up resistor is not automatically disconnected regardless P1RD bit-x. If ZCR1 (High-Z Control Register Bit-1) is high, High-Z control is enabled. At this time, if HIZ bit in the PSCR (Power Saving Control Register) becomes high, pin status becomes high impedance.

To use P1.5/SPIO and P1.7/SPICLK as an SPIO output port, PCR1 should be high. And, an SPI output should be enabled in the SPI block.

When P1NOPNx = 0, Port-1.x output buffer will be CMOS Push-Pull. But, P1NOPNx = 1, Port-1.x output buffer will become N-channel Open Drain.

Address	90H							
Bit Addr.	97H	96H	95H	94H	93H	92H	91H	90H
BIT	7	6	5	4	3	2	1	0
NAME	P17	P16	P15	P14	P13	P12	P11	P10
Definition	Port-1 Pin or Latch bit-7	Port-1 Pin or Latch bit-6	Port-1 Pin or Latch bit-5	Port-1 Pin or Latch bit-4	Port-1 Pin or Latch bit-3	Port-1 Pin or Latch bit-2	Port-1 Pin or Latch bit-1	Port-1 Pin or Latch bit-0
Reset Value	1	1	1	1	1	1	1	1
Read/Write by Software	R/W RMW Ins: P17 Latch Read Other Ins: P1.7 Pin Read	R/W RMW Ins: P16 Latch Read Other Ins: P1.6 Pin Read	R/W RMW Ins: P15 Latch Read Other Ins: P1.5 Pin Read	R/W RMW Ins: P14 Latch Read Other Ins: P1.4 Pin Read	R/W RMW Ins: P13 Latch Read Other Ins: P1.3 Pin Read	R/W RMW Ins: P12 Latch Read Other Ins: P1.2 Pin Read	R/W RMW Ins: P11 Latch Read Other Ins: P1.1 Pin Read	R/W RMW Ins: P10 Latch Read Other Ins: P1.0 Pin Read
Write by Hardware								

TABLE 50. *Definition of P1*

Address	BAH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P1IO7	P1IO6	P1IO5	P1IO4	P1IO3	P1IO2	P1IO1	P1IO0

Definition	Port-1 I/O Control Register bit-7 0: Input Mode 1: Output Mode	Port-1 I/O Control Register bit-6 0: Input Mode 1: Output Mode	Port-1 I/O Control Register bit-5 0: Input Mode 1: Output Mode	Port-1 I/O Control Register bit-4 0: Input Mode 1: Output Mode	Port-1 I/O Control Register bit-3 0: Input Mode 1: Output Mode	Port-1 I/O Control Register bit-2 0: Input Mode 1: Output Mode	Port-1 I/O Control Register bit-1 0: Input Mode 1: Output Mode	Port-1 I/O Control Register bit-0 0: Input Mode 1: Output Mode
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 51. *Definition of P1IO*

Address	BBH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P1RD7	P1RD6	P1RD5	P1RD4	P1RD3	P1RD2	P1RD1	P1RD0
Definition	Port-1 Resistor Disable Register bit-7 0: Resistor Enable 1: Disable	Port-1 Resistor Disable Register bit-6 0: Resistor Enable 1: Disable	Port-1 Resistor Disable Register bit-5 0: Resistor Enable 1: Disable	Port-1 Resistor Disable Register bit-4 0: Resistor Enable 1: Disable	Port-1 Resistor Disable Register bit-3 0: Resistor Enable 1: Disable	Port-1 Resistor Disable Register bit-2 0: Resistor Enable 1: Disable	Port-1 Resistor Disable Register bit-1 0: Resistor Enable 1: Disable	Port-1 Resistor Disable Register bit-0 0: Resistor Enable 1: Disable
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 52. *Definition of P1RD*

Address	B9H

Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P1NOPN7	P1NOPN6	P1NOPN5	P1NOPN4	P1NOPN3	P1NOPN2	P1NOPN1	P1NOPN0
Definition	Port-1 Output Buffer bit-7 0: CMOS Push_Pull 1: N-ch open drain	Port-1 Output Buffer bit-6 0: CMOS Push_Pull 1: N-ch open drain	Port-1 Output Buffer bit-5 0: CMOS Push_Pull 1: N-ch open drain	Port-1 Output Buffer bit-4 0: CMOS Push_Pull 1: N-ch open drain	Port-1 Output Buffer bit-3 0: CMOS Push_Pull 1: N-ch open drain	Port-1 Output Buffer bit-2 0: CMOS Push_Pull 1: N-ch open drain	Port-1 Output Buffer bit-1 0: CMOS Push_Pull 1: N-ch open drain	Port-1 Output Buffer bit-0 0: CMOS Push_Pull 1: N-ch open drain
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 53. *Definition of P1NOPN*

14.9. P3 (Port-3)

The Pin-3.0 to 3.7 have two functional modes, Special output Mode (MODOUT, TXEN) and Port Mode (Bi-directional Port Mode). Initial Status of PCR3 (Port Control Register Bit-3) is 0 (low). The PCR3 must be set to high for any case. The direction of Port-3 can be decided bit-by-bit. The read policy in Port Mode follows that described before. Read-Modify-Write instructions read the port latch than the pin.

To use Pin-3.0 to 3.7 as a normal input port, PCR3 should be high. If PCR3 is high, the direction of each Port-3 pin is decided by each P3IO bit. If the P3IO bit-x is high, Port-3.x is output mode. If low, Port-3.x is input mode. Initial status is input mode because P3IO (Port-3 I/O Direction Register) is reset to 00000000B. In input mode, pull-up resistor can be connected by P3RD (Port-3 Resistor Disable Register). If the P3RD bit-x is high, Port-3.x pull-up resistor is disconnected. If low, Port-3.x pull-up resistor is connected. Initial status of P3RD is 00000000B, thus pull-up resistor is connected. If ZCR3 (High-Z Control Register Bit-3) is high, High-Z control is enabled. At this time, if HIZ bit in the PSCR (Power Saving Control Register) becomes high, pull-up resistor is disconnected and pin status becomes high impedance. In input mode, P3.0/CPTRU2, P3.1/CPTRU1, P3.2/CPTRD1, P3.5DI, P3.6/IRQ1, and P3.7/IRQ2 can be used special input pins.

To use Pin-3.0 to 3.7 as a normal output port, PCR3 should be high. If the P3IO bit-x is high, Port-3.x is output mode. In output mode, pull-up resistor is not automatically disconnected regardless P3RD bit-x. If ZCR3 (High-Z Control Register Bit-3) is high, High-Z control is enabled. At this time, if HIZ bit in the PSCR (Power Saving Control Register) becomes high, pin status becomes high impedance.

To use P3.3/TXEN and P3.4/MODOUT as a special output port, PCR3 should be high. And, a special output should be enabled in each block.

To use P3.6/NWR and P3.7/NRD as External Memory Read and Write Pins, PCR3 should be low. And, P3IO.6, P3IO.7, P3.6, and P3.7 latch should be high. To use P3.6/ NWR and P3.7/NRD as normal Port Pins, don't use MOVX instructions that access External Data Memory.

When P3NOPNx = 0, Port-3.x output buffer will be CMOS Push-Pull. But, P3NOPNx = 1, Port-3.x output buffer will become N-channel Open Drain.

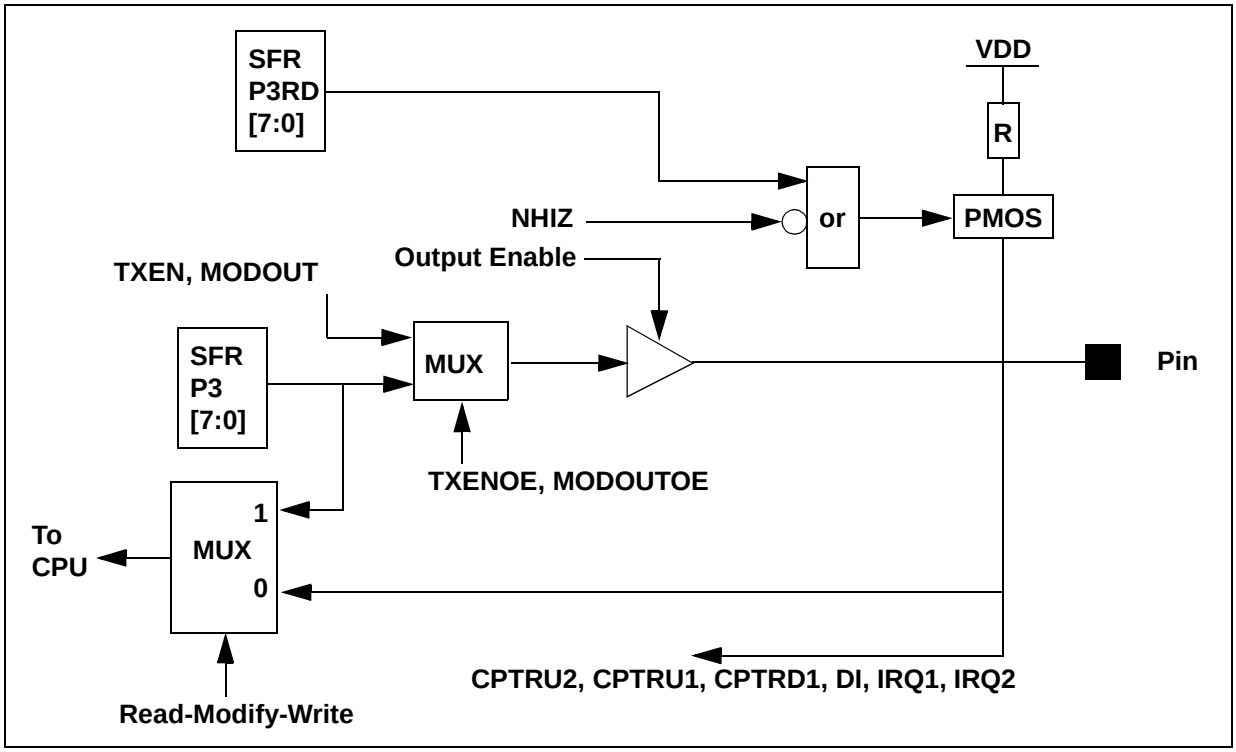


Figure 27. *Port 3*

Address	B0H							
Bit Addr.	B7H	B6H	B5H	B4H	B3H	B2H	B1H	B0H
BIT	7	6	5	4	3	2	1	0
NAME	P37	P36	P35	P34	P33	P32	P31	P30

Definition	Port-3 Pin or Latch bit-7	Port-3 Pin or Latch bit-6	Port-3 Pin or Latch bit-5	Port-3 Pin or Latch bit-4	Port-3 Pin or Latch bit-3	Port-3 Pin or Latch bit-2	Port-3 Pin or Latch bit-1	Port-3 Pin or Latch bit-0
Reset Value	1	1	1	1	1	1	1	1
Read/Write by Software	R/W RMW Ins: P37 Latch Read Other Ins: P3.7 Pin Read	R/W RMW Ins: P36 Latch Read Other Ins: P3.6 Pin Read	R/W RMW Ins: P35 Latch Read Other Ins: P3.5 Pin Read	R/W RMW Ins: P34 Latch Read Other Ins: P3.4 Pin Read	R/W RMW Ins: P33 Latch Read Other Ins: P3.3 Pin Read	R/W RMW Ins: P32 Latch Read Other Ins: P3.2 Pin Read	R/W RMW Ins: P31 Latch Read Other Ins: P3.1 Pin Read	R/W RMW Ins: P30 Latch Read Other Ins: P3.0 Pin Read
Write by Hardware								

TABLE 54. *Definition of P3*

Address	DAH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P3IO7	P3IO6	P3IO5	P3IO4	P3IO3	P3IO2	P3IO1	P3IO0
Definition	Port-3 I/O Control Register bit-7 0: Input Mode 1: Output Mode	Port-3 I/O Control Register bit-6 0: Input Mode 1: Output Mode	Port-3 I/O Control Register bit-5 0: Input Mode 1: Output Mode	Port-3 I/O Control Register bit-4 0: Input Mode 1: Output Mode	Port-3 I/O Control Register bit-3 0: Input Mode 1: Output Mode	Port-3 I/O Control Register bit-2 0: Input Mode 1: Output Mode	Port-3 I/O Control Register bit-1 0: Input Mode 1: Output Mode	Port-3 I/O Control Register bit-0 0: Input Mode 1: Output Mode
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 55. *Definition of P3IO*

Address	DBH

Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P3RD7	P3RD6	P3RD5	P3RD4	P3RD3	P3RD2	P3RD1	P3RD0
Definition	Port-3 Resistor Disable Register bit-7 0: Resistor Enable 1: Disable	Port-3 Resistor Disable Register bit-6 0: Resistor Enable 1: Disable	Port-3 Resistor Disable Register bit-5 0: Resistor Enable 1: Disable	Port-3 Resistor Disable Register bit-4 0: Resistor Enable 1: Disable	Port-3 Resistor Disable Register bit-3 0: Resistor Enable 1: Disable	Port-3 Resistor Disable Register bit-2 0: Resistor Enable 1: Disable	Port-3 Resistor Disable Register bit-1 0: Resistor Enable 1: Disable	Port-3 Resistor Disable Register bit-0 0: Resistor Enable 1: Disable
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 56. *Definition of P3RD*

Address	ABH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P3NOPN7	P3NOPN6	P3NOPN5	P3NOPN4	P3NOPN3	P3NOPN2	P3NOPN1	P3NOPN0
Definition	Port-3 Output Buffer bit-7 0: CMOS Push_Pull 1: N-ch open drain	Port-3 Output Buffer bit-6 0: CMOS Push_Pull 1: N-ch open drain	Port-3 Output Buffer bit-5 0: CMOS Push_Pull 1: N-ch open drain	Port-3 Output Buffer bit-4 0: CMOS Push_Pull 1: N-ch open drain	Port-3 Output Buffer bit-3 0: CMOS Push_Pull 1: N-ch open drain	Port-3 Output Buffer bit-2 0: CMOS Push_Pull 1: N-ch open drain	Port-3 Output Buffer bit-1 0: CMOS Push_Pull 1: N-ch open drain	Port-3 Output Buffer bit-0 0: CMOS Push_Pull 1: N-ch open drain
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 57. *Definition of P3NOPN*

14.10. Port-4

During External Memory Mode, P4.2 to P4.3 are used as ALE, and PSEN.

When External Memory Mode = 0, the Port-4(P4) is a 8-bit bidirectional port that can be used as special output pins (P4.0/RFPWR, P4.1/PLLSW, P4.2/LOCK).

The direction of Port-4 can be decided bit-by-bit. If P4LATIN(PCR bit-5) = 0, CPU reads pin status. If P4LATIN(PCR bit-5) = 1, CPU reads port latch rather than pin.

To use Pin-4.x as a normal input port, P4IOx should be low. The direction of each Port-4 pin is decided by each P4IO bit. If the P4IO bit-x is high, Port-4.x is output mode. If low, Port-4.x is input mode. Initial status is input mode because P4IO (Port-4 I/O Direction Register) is reset to 00000000B. In input mode, pull-up resistor can be connected by P4RD (Port-4 Resistor Disable Register). If the P4RD bit-x is high, Port-4.x pull-up resistor is disconnected. If low, Port-4.x pull-up resistor is connected. Initial status of P4RD is 00000000B, thus pull-up resistor is connected. If HIZ bit in the PSCR (Power Saving Control Register) becomes high, pull-up resistor is disconnected and pin status becomes high impedance.

To use Pin-4.x as a normal output port, P4IOx should be high. If the P4IO bit-x is high, Port-4.x is output mode. In output mode, pull-up resistor is not automatically disconnected regardless P4RD bit-x. If HIZ bit in the PSCR (Power Saving Control Register) becomes high, pin status becomes high impedance.

To use P4.0/RFPWR, P4.1/PLLSW, P4.2/LOCK as a special output port the output should be enabled in register SSTMCR of the SSTM block.

When P4NOPNx = 0, Port-4.x output buffer will be CMOS Push-Pull. But, P4NOPNx = 1, Port4.x output buffer will become N-channel Open Drain.

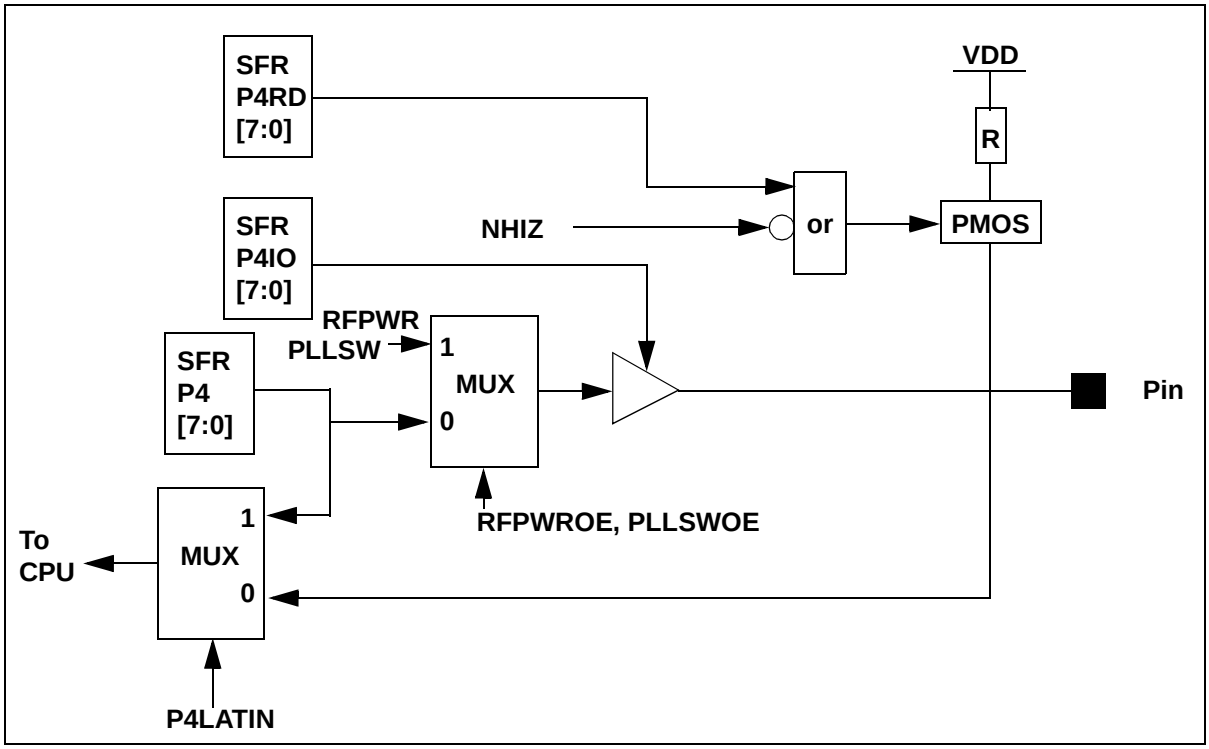


Figure 28. *Port 4*

Address	C0H							
Bit Addr.	C7H	C6H	C5H	C4H	C3H	C2H	C1H	C0H
BIT	7	6	5	4	3	2	1	0
NAME	P47	P46	P45	P44	P43	P42	P41	P40
Definition	Port-4 Pin or Latch bit-7	Port-4 Pin or Latch bit-6	Port-4 Pin or Latch bit-5	Port-4 Pin or Latch bit-4	Port-4 Pin or Latch bit-3	Port-4 Pin or Latch bit-2	Port-4 Pin or Latch bit-1	Port-4 Pin or Latch bit-0
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W P4latin=1: P47 Latch Read P4latin=0: P4.7 Pin Read	R/W P4latin=1: P46 Latch Read P4latin=0: P4.6 Pin Read	R/W P4latin=1: P45 Latch Read P4latin=0: P4.5 Pin Read	R/W P4latin=1: P44Latch Read P4latin=0: P4.4 Pin Read	R/W P4latin=1: P43 Latch Read P4latin=0: P4.3 Pin Read	R/W P4latin=1: P42 Latch Read P4latin=0: P4.2 Pin Read	R/W P4latin=1: P41 Latch Read P4latin=0: P4.1 Pin Read	R/W P4latin=1: P40 Latch Read P4latin=0: P4.0 Pin Read

Write by Hardware								

TABLE 58. *Definition of P4*

Address	EAH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P4IO7	P4IO6	P4IO5	P4IO4	P4IO3	P4IO2	P4IO1	P4IO0
Definition	Port-4 I/O Control Register bit-7 0: Input Mode 1: Output Mode	Port-4 I/O Control Register bit-6 0: Input Mode 1: Output Mode	Port-4 I/O Control Register bit-5 0: Input Mode 1: Output Mode	Port-4 I/O Control Register bit-4 0: Input Mode 1: Output Mode	Port-4 I/O Control Register bit-3 0: Input Mode 1: Output Mode	Port-4 I/O Control Register bit-2 0: Input Mode 1: Output Mode	Port-4 I/O Control Register bit-1 0: Input Mode 1: Output Mode	Port-4 I/O Control Register bit-0 0: Input Mode 1: Output Mode
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 59. *Definition of P4IO*

Address	EBH							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P4RD7	P4RD6	P4RD5	P4RD4	P4RD3	P4RD2	P4RD1	P4RD0
Definition	Port-4 Resistor Disable Register bit-7 0: Resistor Enable 1: Disable	Port-4 Resistor Disable Register bit-6 0: Resistor Enable 1: Disable	Port-4 Resistor Disable Register bit-5 0: Resistor Enable 1: Disable	Port-4 Resistor Disable Register bit-4 0: Resistor Enable 1: Disable	Port-4 Resistor Disable Register bit-3 0: Resistor Enable 1: Disable	Port-4 Resistor Disable Register bit-2 0: Resistor Enable 1: Disable	Port-4 Resistor Disable Register bit-1 0: Resistor Enable 1: Disable	Port-4 Resistor Disable Register bit-0 0: Resistor Enable 1: Disable
Reset Value	0	0	0	0	0	0	0	0

Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 60. *Definition of P4RD*

Address	E9H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME	P4NOPN7	P4NOPN6	P4NOPN5	P4NOPN4	P4NOPN3	P4NOPN2	P4NOPN1	P4NOPN0
Definition	Port-4 Output Buffer bit-7 0: CMOS Push_Pull 1: N-ch open drain	Port-4 Output Buffer bit-6 0: CMOS Push_Pull 1: N-ch open drain	Port-4 Output Buffer bit-5 0: CMOS Push_Pull 1: N-ch open drain	Port-4 Output Buffer bit-4 0: CMOS Push_Pull 1: N-ch open drain	Port-4 Output Buffer bit-3 0: CMOS Push_Pull 1: N-ch open drain	Port-4 Output Buffer bit-2 0: CMOS Push_Pull 1: N-ch open drain	Port-4 Output Buffer bit-1 0: CMOS Push_Pull 1: N-ch open drain	Port-4 Output Buffer bit-0 0: CMOS Push_Pull 1: N-ch open drain
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 61. *Definition of P4NOPN*

15. External Interrupts

15.1. Overview

There are 2 external interrupts, IRQ1 and IRQ2. The external interrupts share pins with the Port-X.0 to Port-X.1. To use an IRQi (i = 0 to 1) pin as an external interrupt pin, the pin must be assigned to input mode by the Port-X Control Register. The external interrupt sources can be programmed to be negative edge activated, low level activated, positive edge activated, or both edge activated by setting or clearing bit IRQi1 and IRQi0 in Register XICRn (External Interrupt Control Register). If IRQiA = 0 and IRQiB = 0, IRQi is negative edge triggered. In this mode if successive samples of the IRQi pin show a high in one sample and a low in the next cycle, interrupt request flag, IRQiF, will be set. Flag bit IRQiF then requests the interrupt. This bit must be cleared by software in the interrupt service routine. The IRQiF bit can be also set or cleared by software. If IRQiA = 1 and IRQiB = 0, IRQi is positive edge triggered. In this mode if successive samples of the IRQi pin show a low in one sample and a high in the next cycle, interrupt request flag IRQiF will be set. Flag bit IRQiF then requests the interrupt. If IRQiA = 1 and IRQiB = 1, IRQi is both (positive or negative) edge triggered. In this mode if successive samples of the IRQi pin show difference between one sample and the next sample, interrupt request flag IRQiF will be set. Flag bit IRQiF then requests the interrupt. If IRQiA = 0 and IRQiB = 1, IRQi is low level activated. If IRQi is low, interrupt request flag IRQiF will be set. In this case, software can not clear IRQiF if the pin IRQi is still low. Thus, make sure the pin IRQi becoming high level (inactivate state) in the interrupt service routine.

]

IRQiA: IRQiB	Active Edge or Level
[IRQiA: IRQiB] = 00	Negative Edge
[IRQiA: IRQiB] = 01	Low Level
[IRQiA: IRQiB] = 10	Positive Edge
[IRQiA: IRQiB] = 11	Both (Positive or Negative) Edge

TABLE 62. *IRQi (i = 0 to 5) Mode Selection*

15.2. Interrupt Control Block

The SS1102C provides 13 interrupt sources (2 external interrupt and 11 internal interrupt). There are two external interrupt pins, IRQ1 to IRQ2. Following peripheral blocks may generate interrupt request, Watch-Dog Timer, Time Base Timer, Capture Timer-1 to 2, SPI, LBD, TXFIFO, RXFIFO, Signaling Word transmit or received, S/N Ratio, and Lock. These interrupt sources are divided into 5 groups.

When an interrupt is generated, the interrupt request flag that generated it should be cleared by software when the service routine is vectored. All of the interrupt request flags can be set or cleared by software, with the same result as though it had been set or cleared by hardware. That is, interrupt can be generated or pending interrupts can be canceled in software. Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Registers ISE0 to ISE2. And, each of interrupt group can be enabled or disabled by setting or clearing a bit in Special Function Register IE. The bit EA in IE contains also a global disable bit (0: disable, 1: Enable), which disables all interrupts at once.

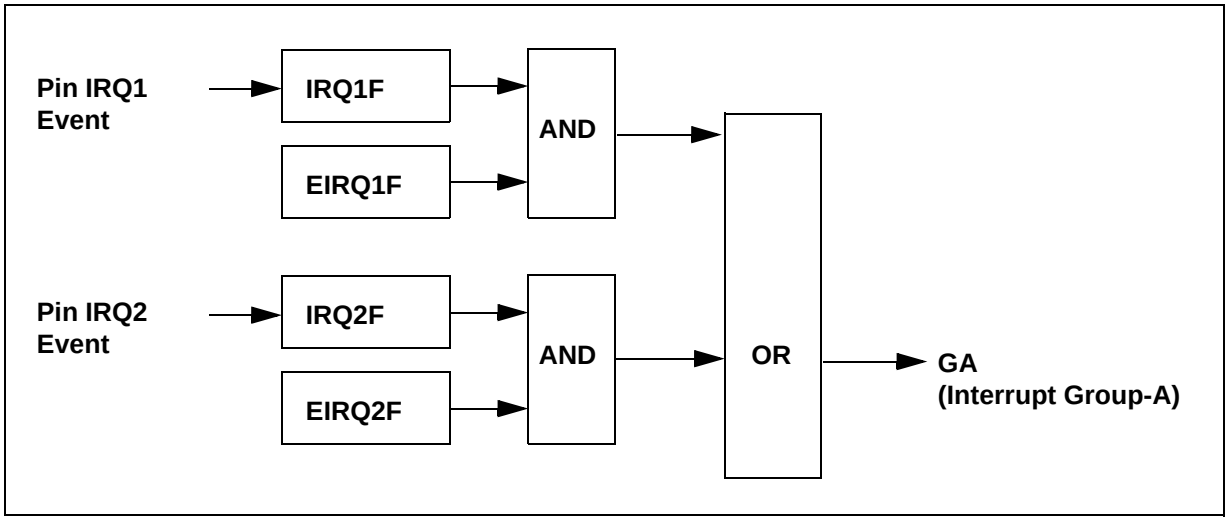


Figure 29. *Block Diagram of Interrupt Group-A*

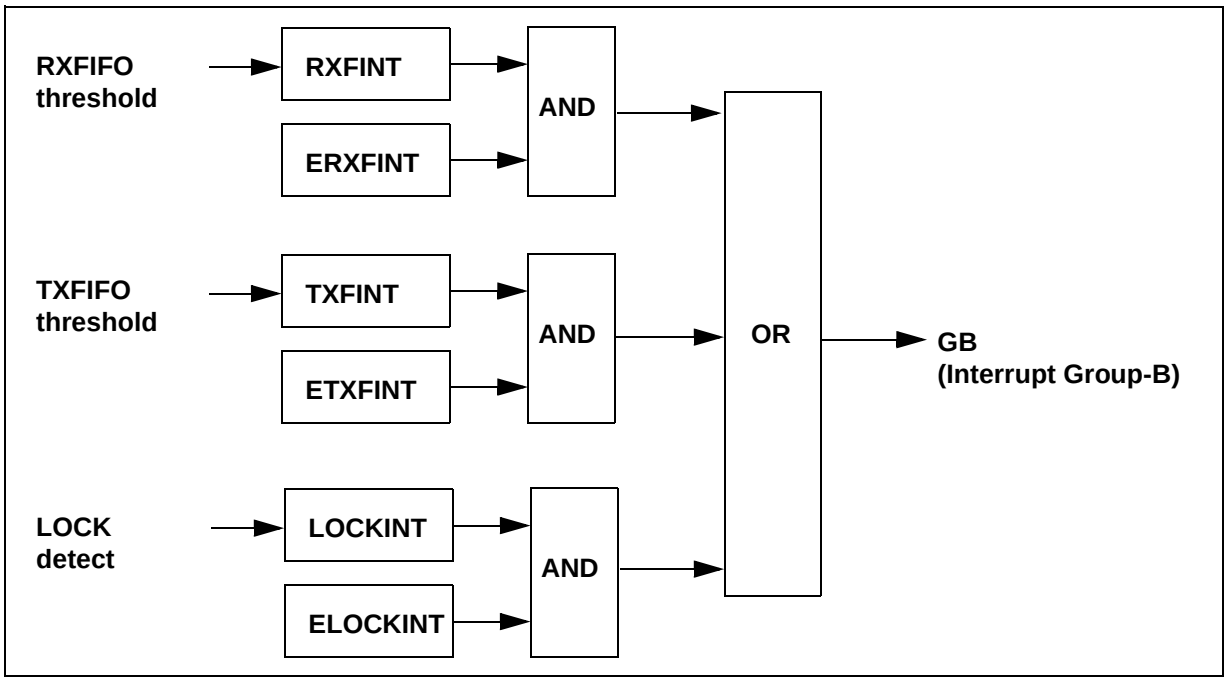


Figure 30. *Block Diagram of Interrupt Group-B*

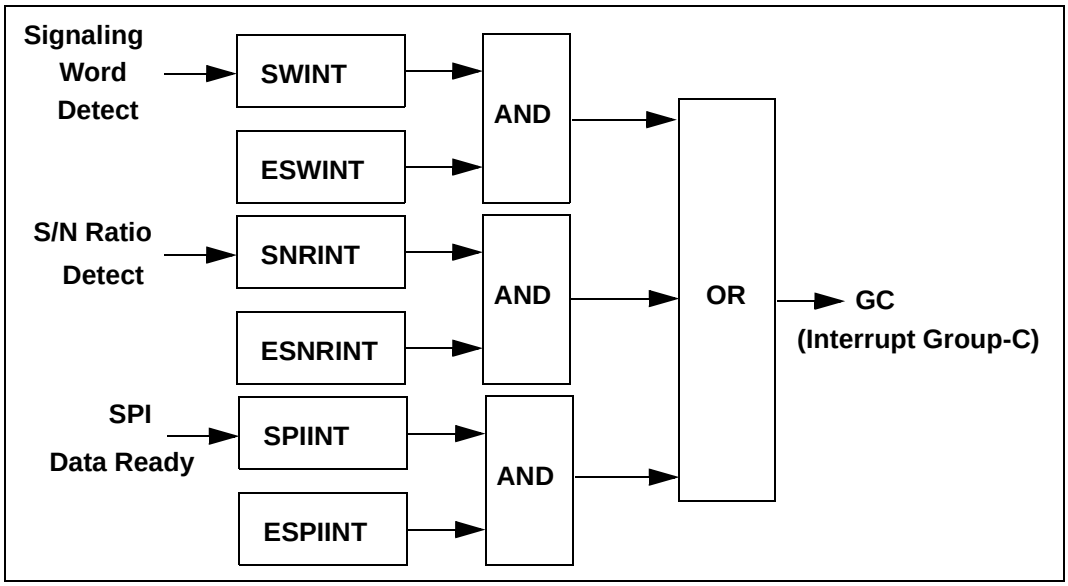


Figure 31. *Block Diagram of Interrupt Group-C*

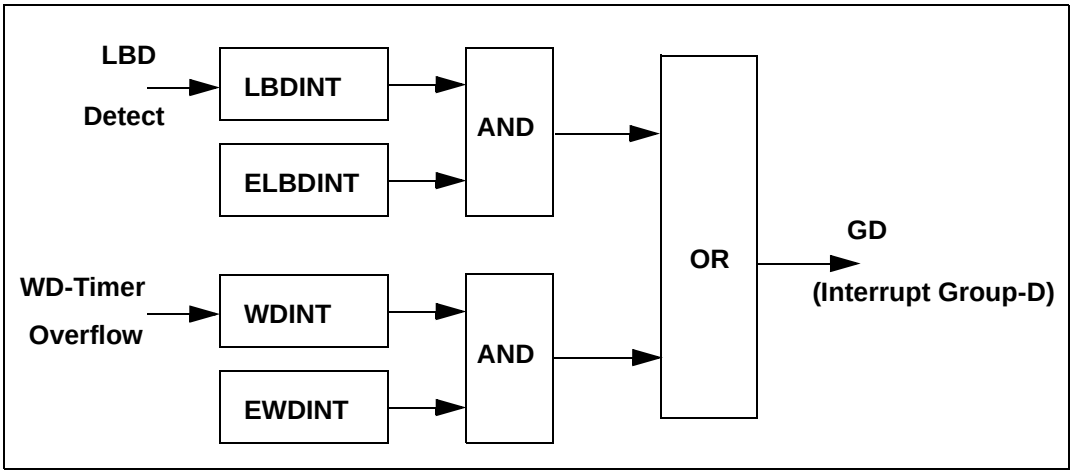


Figure 32. *Block Diagram of Interrupt Group-D*

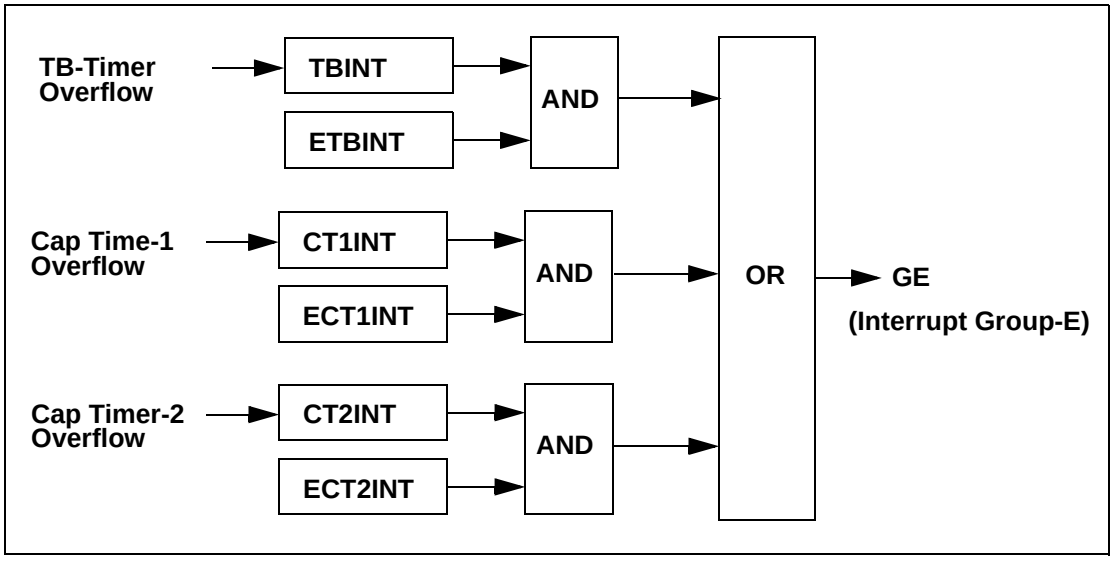


Figure 33. *Block Diagram of Interrupt Group-E*

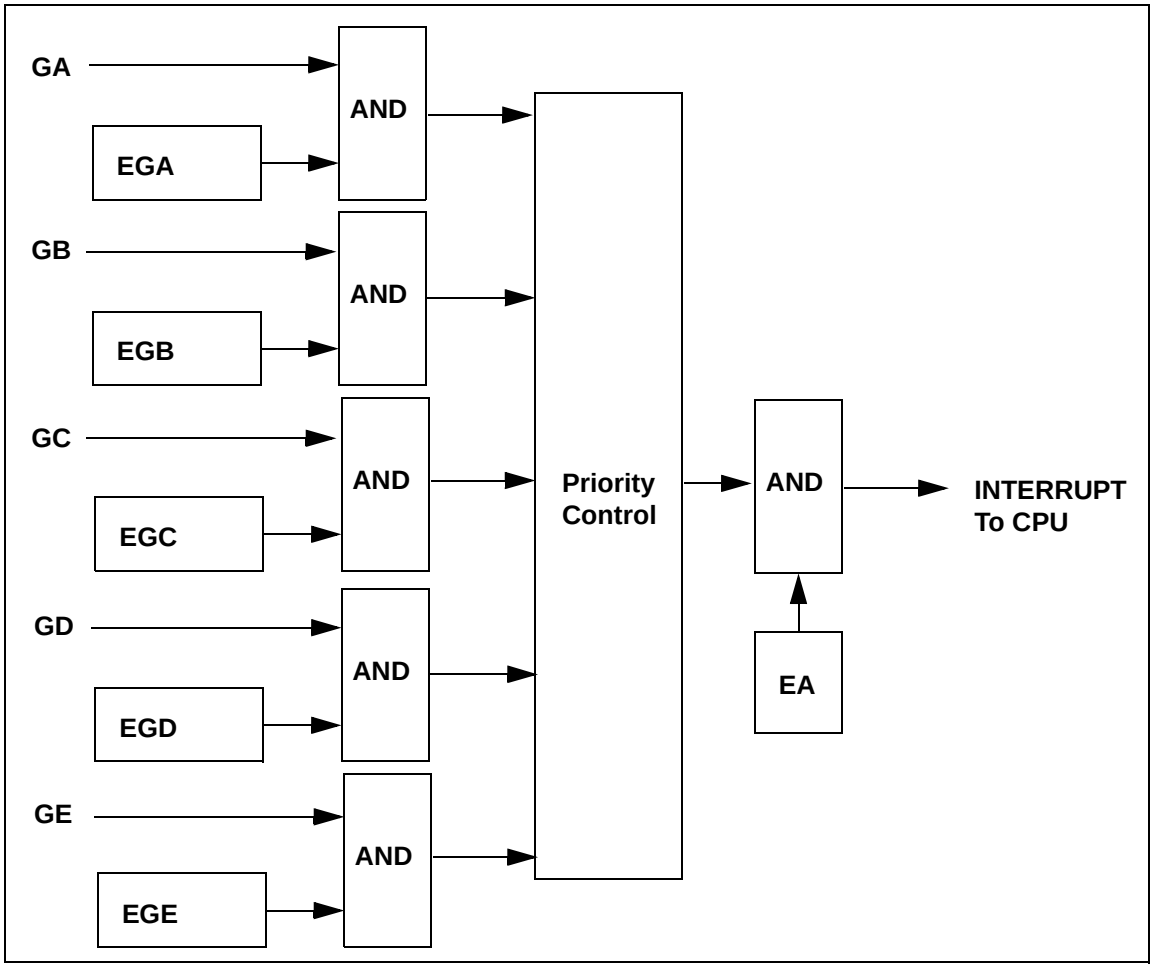


Figure 34. *Interrupt Control Block Diagram*

15.3. Interrupt Request Registers (INT0, INT1, INT2)

]

Address	D8H							
Bit Addr.	DFH	DEH	DDH	DCH	DBH	DAH	D9H	D8H
BIT	7	6	5	4	3	2	1	0
NAME							IRQ2F	IRQ1F
Definition	Reserved bit	Reserved bit	Reserved bit	Reserved bit	Reserved bit	Reserved bit	IRQ2 Request 1: Request 0: No	IRQ1 Request 1: Request 0: No
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R	R	R	R	R	R	R/W	R/W
Write by Hardware							Set by IRQ2 Event	Set by IRQ1 Event

TABLE 63. *Definition of INT0*

Address	E8H							
Bit Addr.	EFH	EEH	EDH	ECH	EBH	EAH	E9H	E8H
BIT	7	6	5	4	3	2	1	0
NAME		SWINT	SNRINT	SPIINT		RXFINT	TXFINT	LOCKINT
Definition	Reserved bit	Signaling-Word Interrupt 1: Request 0: No	S/N Ratio Interrupt 1: Request 0: No	SPI Interrupt 1: Request 0: No	Reserved bit	RXFIFO Interrupt 1: Request 0: No	TXFIFO Interrupt 1: Request 0: No	LOCK Interrupt 1: Request 0: No
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware		Set by SW Detect	Set by S/N Ratio Detect	Set by SPI data Ready		Set by RXFIFO Threshold	Set by TXFIFO Threshold	Set by LOCK Detect

TABLE 64. *Definition of INT1*

Address	F8H							
Bit Addr.	FFH	FEH	FDH	FCH	FBH	FAH	F9H	F8H
BIT	7	6	5	4	3	2	1	0
NAME		TBINT	CT1INT	CT2INT			LBDINT	WDINT
Definition	Reserved bit	Time-Base Timer Interrupt 1: Request 0: No	Capture Timer-1 Interrupt 1: Request 0: No	Capture Timer-2 Interrupt 1: Request 0: No	Reserved bit	Reserved bit	LBD Interrupt 1: Request 0: No	Watch-Dog Timer Interrupt 1: Request 0: No
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware		Set by TB-Timer Overflow	Set by C Timer-1 Overflow	Set by C Timer-2 Overflow			Set by LBD Detect	Set by WDTimer Overflow

TABLE 65. *Definition of INT2*

15.3.1 Interrupt Source Enable Registers (ISE0, ISE1, ISE2)

Address	D7H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME							EIRQ2F	EIRQ1F
Definition	Reserved bit	Reserved bit	Reserved bit	Reserved bit	Reserved bit	Reserved bit	Enable IRQ2F 1: Enable 0: Disable	Enable IRQ1F 1: Enable 0: Disable
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R	R	R	R	R	R	R/W	R/W
Write by Hardware								

TABLE 66. *Definition of ISE0*

Address	E7H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME		ESWINT	ESNRINT	ESPIINT		ERXFINT	ETXFINT	ELockINT
Definition	Reserved bit	Enable Signaling Word Interrupt 1: Enable 0: Disable	Enable S/N Ratio Interrupt 1: Enable 0: Disable	Enable SPI Interrupt 1: Enable 0: Disable	Reserved bit	Enable RXFIFO Interrupt 1: Enable 0: Disable	Enable TXFIFO Interrupt 1: Enable 0: Disable	Enable LOCK Interrupt 1: Enable 0: Disable
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 67. *Definition of ISE1*

Address	F7H							
Bit Addr.	None							
BIT	7	6	5	4	3	2	1	0
NAME		ETBINT	ECT1INT	ECT2INT			ELDBINT	EWDINT
Definition	Reserved bit	Enable Time-Base Timer Interrupt 1: Enable 0: Disable	Enable Capture Timer-1 Interrupt 1: Enable 0: Disable	Enable Capture Timer-2 Interrupt 1: Enable 0: Disable	Reserved bit	Reserved bit	Enable LDB Interrupt 1: Enable 0: Disable	Enable Watch-Dog Timer Interrupt 1: Enable 0: Disable
Reset Value	0	0	0	0	0	0	0	0
Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 68. *Definition of ISE2*

15.3.2 Interrupt Group Enable Register (IE)

Address	A8H							
Bit Addr.	AFH	AEH	ADH	ACH	ABH	AAH	A9H	A8H
BIT	7	6	5	4	3	2	1	0
NAME	EA			EGE	EGD	EGC	EGB	EGA
Definition	Enable All Interrupt 1: Enable 0: Disable	Reserved bit	Reserved bit	Enable Group-E 1: Enable 0: Disable	Enable Group-D 1: Enable 0: Disable	Enable Group-C 1: Enable 0: Disable	Enable Group-B 1: Enable 0: Disable	Enable Group-A 1: Enable 0: Disable
Reset Value	0	unknown	unknown	0	0	0	0	0
Read/Write by Software	R/W			R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 69. Definition of IE

15.3.3 Interrupt Priority Register (IP)

Each interrupt group (Group-A, B, C, D, and E) can also be individually programmed to one of two priority levels by setting or clearing a bit in Special Function Register IP. A low priority interrupt can be interrupted by a high priority interrupt, but not by another low priority interrupt. A high priority interrupt can not be interrupted by another interrupt group. If two requests of different priority levels are received simultaneously, the request of higher priority level is serviced. If request of the same priority level are received simultaneously, an internal polling sequence determines which request is serviced. Thus within each priority level there is a second priority structure determined by the polling sequence, as follows.

]

Number	Group	Request Flags	Vector Addr.	Priority Within Level
1	Group-A	IRQ1/1	0003H	Highest
2	Group-B	RXFINT/TXFINT/LOCKINT	000BH	
3	Group-C	SWINT/SNRINT/SPIINT	0013H	
4	Group-D	LBDINT/WDINT	001BH	

5	Group-E	TBINT/CT1INT/CT2INT	0023H	Lowest

TABLE 70. Internal Priority and Vector Address

Address	B8H							
Bit Addr.	BFH	BEH	BDH	BCH	BBH	BAH	B9H	B8H
BIT	7	6	5	4	3	2	1	0
NAME				PGE	PGD	PGC	PGB	PGA
Definition	Reserved bit	Reserved bit	Reserved bit	Priority of Group-E 1: High Priority 0: Low Priority	Priority of Group-D 1: High Priority 0: Low Priority	Priority of Group-C 1: High Priority 0: Low Priority	Priority of Group-B 1: High Priority 0: Low Priority	Priority of Group-A 1: High Priority 0: Low Priority
Reset Value	unknown	unknown	unknown	0	0	0	0	0
Read/Write by Software				R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 71. *Definition of IP*

15.3.4 Interrupt Processing

The interrupt request flags are sampled at S5P2 of every machine cycle. The samples are polled during the following machine cycle. If one of the flags was in a set condition at S5P2 of the preceding cycle, the polling cycle will find it and the interrupt system will generate an LCALL to the appropriate service routine, provided this hardware generated LCALL is blocked by any of the following conditions:

1. An interrupt of equal or higher priority level is already in progress

2. The current polling cycle is not the final cycle in the execution of the instruction in progress

3. The instruction in progress is RETI.

4. The instruction in progress is any access to the IE or IP registers.

Any of these four conditions will block the generation of the LCALL to the interrupt service routine. Condition 2 ensures that the instruction in progress will be completed

before vectoring to any service routine. Condition 3 & 4 ensures that if the instruction in progress is RETI or any access to IE or IP, then at least one more instruction will be executed before any interrupt is vectored to.

The polling cycle is repeated with each machine cycle, and the values polled are the values that were present at S5P2 of the previous machine cycle. Note then that if an interrupt flag is active but not being responded to for one of the above conditions, if the flag is not still active when the blocking condition is removed, the denied interrupt will not be serviced. In other words, the fact that the interrupt flag once active but not serviced is not remembered. Every polling cycle is new.

The processor acknowledges an interrupt request by executing a hardware generated LCALL to the appropriate service routine. It also clears the flag that generated the interrupt. The hardware generated LCALL pushes the contents of the Program Counter onto the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being vectored to.

Executing proceeds from that location until the RETI instructions is encountered. The RETI instruction informs the processor that this interrupt routine is no longer in progress, then pops the top two bytes from the stack and reloads the Program Counter. Execution of the interrupted program continues from where it left off. Note that a simple RET instruction would also have returned execution to the interrupted program, but it would have left the interrupt control system thinking an interrupt was still in progress.

16. SS1102C Special Modes

16.1. TEST ROM (Using Auxiliary RAM)

The TEST ROM for the SS1102C is initiated by driving the TST and P4.2 pin to a high and the P4.3 pin to a low during reset. This will cause the CPU to fetch opcodes from the onchip auxiliary data ram. The ram is currently 256 bytes. The P4.3 and P4.2 pins will return to the to the general purpose I/O functions after reset has gone inactive.

To load the RAM with opcodes the user should use the “External Memory mode” described below to load the auxiliary data ram using either MOVC or MOVX commands. Then the device can be reset and placed in the test ROM mode without losing the ram data.

16.2. EMULATION MODE

The SS1102C has 22 extra bond pads on the die which are emulation interface I/O. The pins and their functions are listed below. The emulation mode is entered by driving the ENICE pin low. This causes the SS1102C to redirect program and data memory fetches through the emulation interface.

PIN	TYPE	FUNCTION
EAD7-0	I/O	Low byte address and data I/O
EA15-8	OUTPUT	High byte address
EPSEN	OUTPUT	Program store enable for program memory read active low
EALE	OUTPUT	Address latch enable
ENWR	OUTPUT	Data memory write active low
ENRD	OUTPUT	Data memory read active low
ENICE	INPUT	In circuit emulation mode input active low
ENMON	INPUT	Monitor mode active low

TABLE 72. *Extra Pins Function*

In normal operation the SS1102C will fetch opcodes from the internal program memory and data memory is accessed to internal IRAM and, with the MOVX command, to internal XRAM and external data RAM. When emulation mode is entered the internal IRAM continues to be accessed as in normal mode but the program memory fetches and the MOVX data memory fetches are both done through the emulation interface. At the emulation interface the internal program memory fetch is made to look like an external

program memory fetch and all MOVX data memory fetches either internal or external are made to look like external data memory fetches. Both of these fetches are designed to occur through the same emulation interface bus with no disturbances to the port outputs.

The emulator can access the internal IRAM by using the appropriate opcodes to read and write this onchip data memory.

The ENMON input is used by the emulator to stop internal activities such as timers, interrupts, and serial I/O for exercising emulation activities such as step mode, or modifying registers etc.

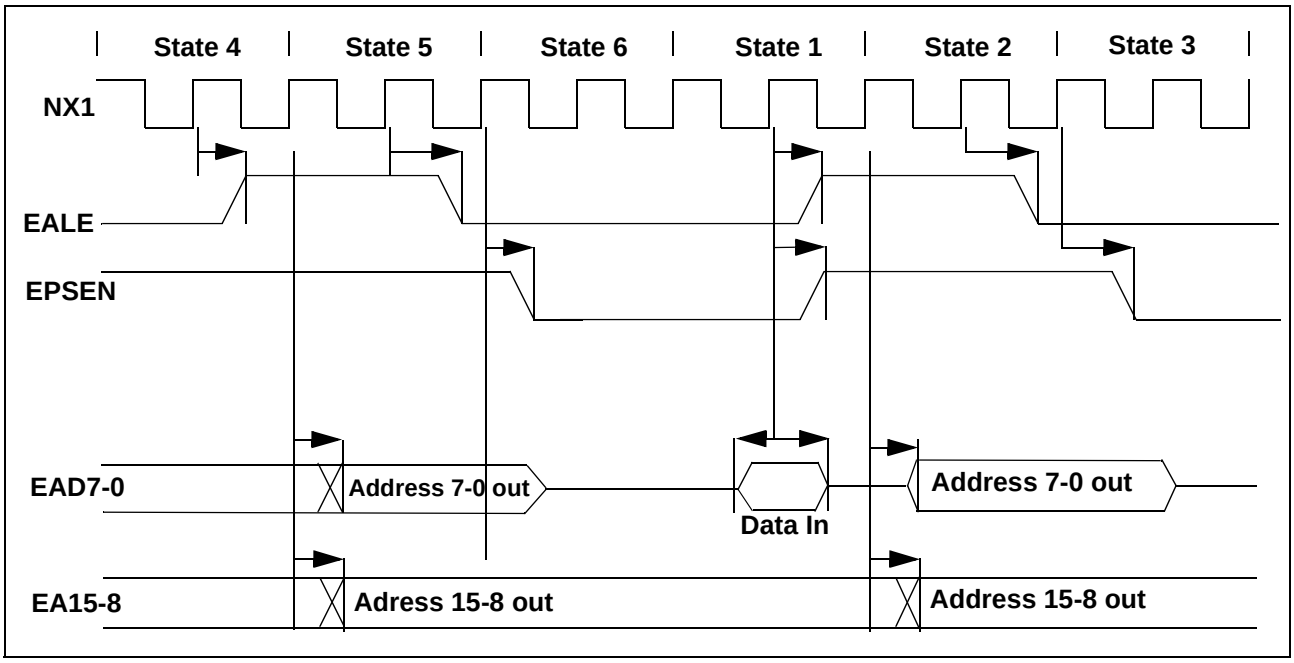


Figure 35. *Emulation Instruction Fetch From Program Memory*

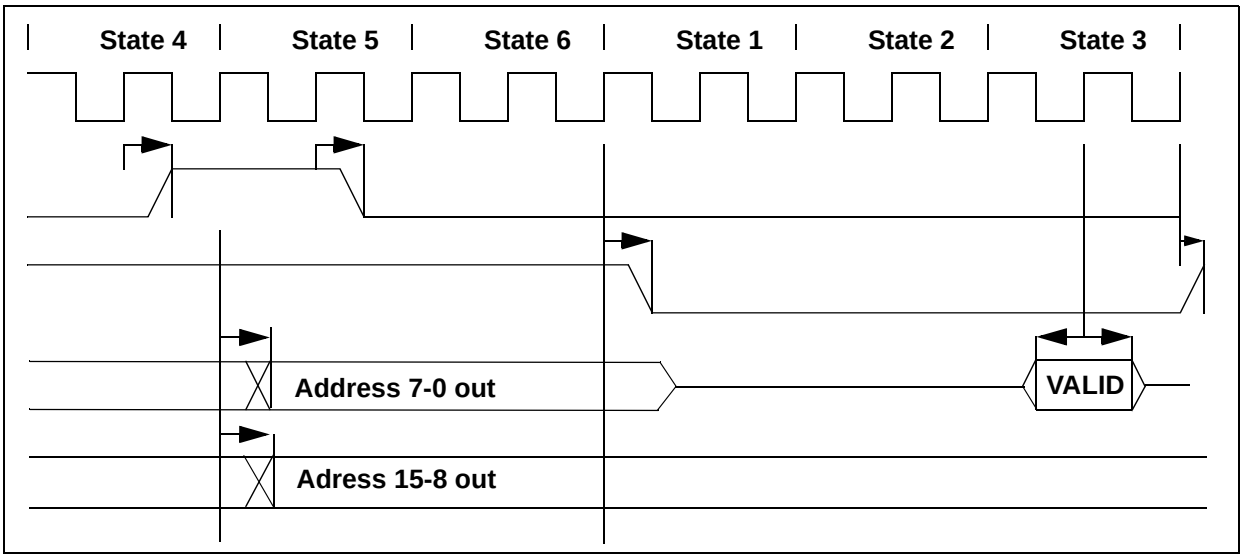


Figure 36. *Emulation Data Memory Read*

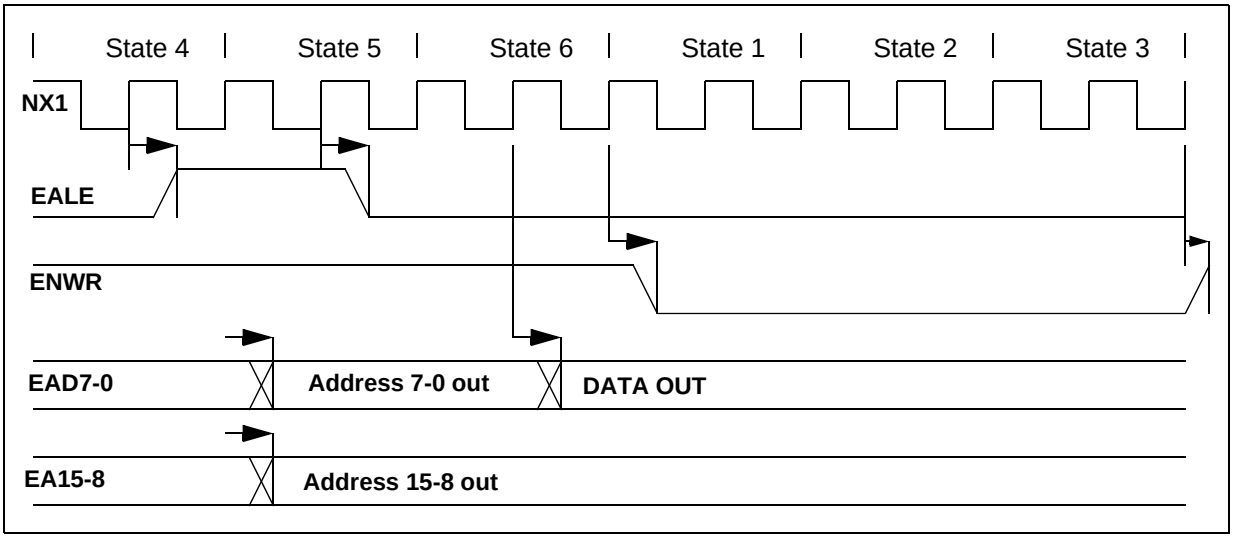


Figure 37. *Emulation Data Memory Write*

16.3. MULTICHIP PROGRAM MEMORY INTERFACE

The SS1102C has 26 extra bond pads on the die which are external ROM memory interface.

The pins and their functions are listed below. Pins beginning with the letter E are shared with the emulation mode. This external ROM memory mode is entered by driving the NXROM pin low. This causes the SS1102C to redirect program memory fetches through this interface. The EPSEN is the NMOE, EAD7-0 is M7-0, and EA15-8 is M15-8 function from the SM8200 core. MD7-0 is the same as the MD7-0 of the SM8200 core being the opcode input to the CPU.

PIN	TYPE	FUNCTION
NXROM	INPUT	Mode control
MD7-0	INPUT	Opcode data input
EAD7-0	OUTPUT	Low byte address
EA15-8	OUTPUT	High byte address
EPSEN	OUTPUT	Program store enable for program memory read active low

TABLE 73. *External Memory Interface Pin Description*

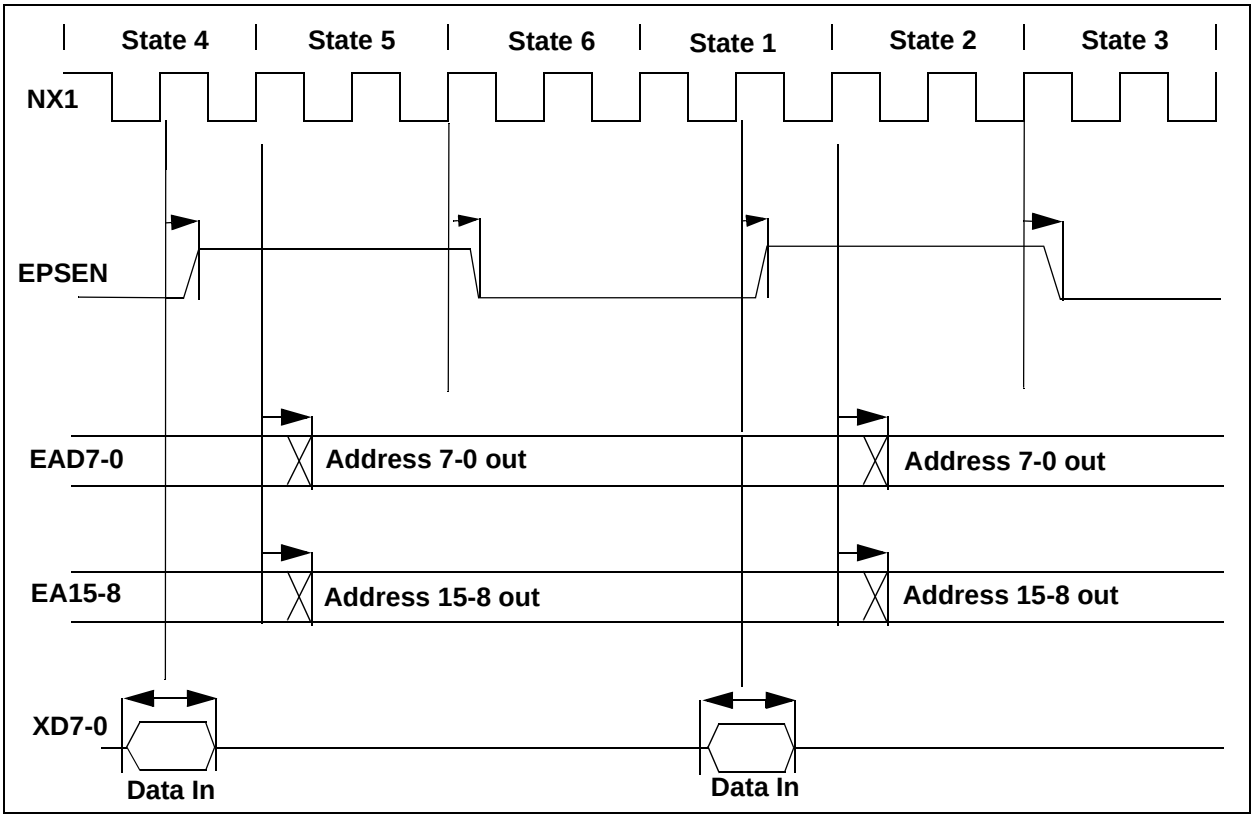


Figure 38. *External Instruction Fetch From Program Memory*

16.4. EXTERNAL MEMORY MODE

The SS1102C is designed to be used in a 48 pin package with internal program and data memory. For test purposes the chip can be put into external memory mode such that external programs and data may be accessed to test the device. This mode is entered by driving the TST pin high with P4.3 and P4.2 low during reset. After reset P4.3 will become the PSEN output and P4.2 will become the ALE output. The remaining pin functions are listed below along with timing diagrams.

PIN	TYPE	FUNCTION
P07-0/AD7-0	I/O	Lower address and data
P27-0/A15-8	OUTPUT	Upper address output
P3.6/NWR	OUTPUT	Data memory write

PIN	TYPE	FUNCTION
P3.7/NRD	OUTPUT	Data memory read
P4.3/PSEN	OUTPUT	Program store enable
P4.2/ALE	OUTPUT	Address latch enable

TABLE 74. *External Memory Pins Description*

17. Comparator module

17.1. General Information

The comparator module converts an incoming analog signal from the RF receiver into a digital signal used as input by the internal digital correlators of the SSTM. This module performs as a 1-bit A/D converter.

When the comparator is enabled it will automatically be placed in the path between the DI input for the SS1102C and the DI input of the internal SSTM module. With the DIREF input this allows the user to input analog DI and a DIREF which is the converted to a digital input to the SSTM module. When the comparator is disabled (default) the DI input of the SS1102C is connected directly to the DI input of the SSTM module and a digital signal is expected.

In this section, a general description of the comparator module is given. This includes a functional description, a block diagram, pin names and definitions, and all the signals necessary for the block and their timing.

17.2. Functional Description

This comparator is a a rail to rail, single supply comparator. It is also high speed and low power. In this comparator, both inputs can change. It has the following features:

supply voltage (Vdd)	2.7 - 3.3 V and 4.5 - 5.5 V
conversion time	100 - 200 nS
input offset voltage	1 - 2 mV
current usage (in active mode)	1 mA max. for 3.3V and 5 mA max. for 5.5 V supply
current usage (in sleep mode)	negligible
input voltage range	0 to supply voltage
output voltage range	rail to rail
temperature range	-20 - +85

TABLE 75. *Comparator Main Features*

17.3. Block Diagram

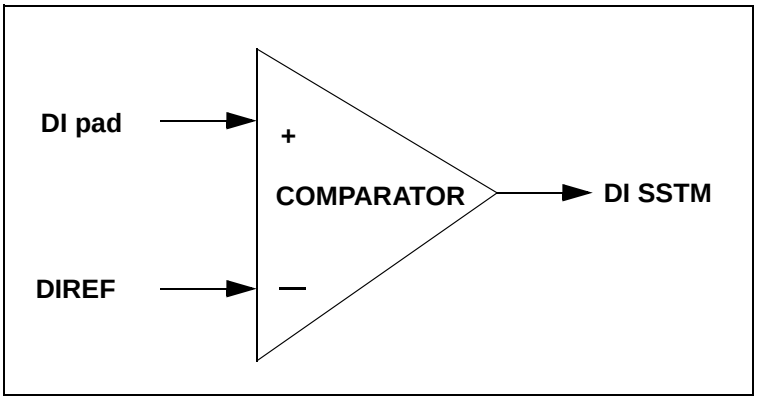


Figure 39. *Comparator Module Block Diagram*

17.4. Comparator Control Register

The comparator control register contains two bits. The enable bit 0 turns on the power and clock to the comparator and automatically redirects the DI input of the SSTM to go from the DI pad input to the comparator output cmpout. The analog input of the DI (P3.5) is connected to the cmp_in1 of the comparator. Note: The user should disable the digital input of the P3.5 port pin (PCR bit 4). The cmpout bit 1 of the comparator control register allows the user to read the comparator output.

Address	85H							
Bit Address								
BIT	7	6	5	4	3	2	1	0
NAME							cmpout	enable
Definition							comparator output value	1-enable the comparator: DI pad > cmp1_in cmpout> DI SSTM 0-disable the comparator: DI pad > DI SSTM
Reset Value	0	0	0	0	0	0	0	0

Read/Write by Software	R	R	R	R	R	R	R	R/W
Write by Hardware							cmpout	

TABLE 76. *Definition of CMPRCR*

17.5. Comparator I/O plot

The following shows a plot of cmpout with respect to cmp_in1, if cmp_in2 is kept constant at 2.5 v.

:

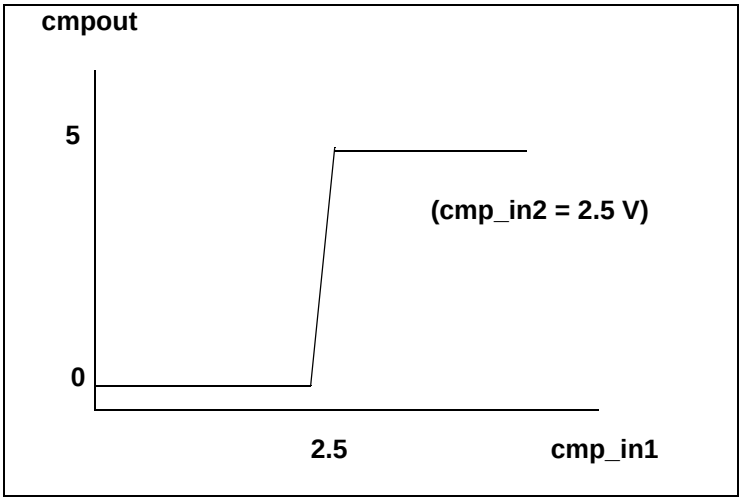


Figure 40. *Comparator Output Plot*

18. Power-On Reset (POR)

18.1. General

The power-on reset circuit provides internal SS1102C reset for most power-up situations.

POR consists of power-up detect block, start-up timer, and reset latch. Output of the reset latch is an internal reset (chip_reset) signal. This signal goes low when Vcc rises to the specified level and the signal goes high after some specified interval of time. This interval is determined by the start-up timer.

Ramping up of Vcc generates the Power-up Detect (PUD) signal. This signal does not come until Vcc achieves a level where the logic circuits of the POR start to operate.

The POR is used only at power-up and should not be used to detect drops in power supply voltage.

The PUD is multiplexed with Nreset (external reset signal).

The SS1102C becomes functional after the chip_reset is generated.

Figures 1 and 2 show timing diagrams.

18.2. Operation

At power-up, the reset latch and the start-up timer are reset to appropriate state by the PUD pulse. After some time interval (2^n pulses of the high frequency clock for an n-bit counter) the start-up timer will trigger the reset latch (if there is no Nreset active) and thus issue the chip_reset signal.

The chip_reset signal will be generated by the internal POR circuit when Nreset signal is held high. The Nreset can be used to override the internal reset.

18.3. Crystal Oscillator Start-Up Time

After Vcc achieves the operational level, the crystal oscillator will require time to stabilize. This is the crystal oscillator start-up time.

Low frequency crystals have a typical start-up time of 1-2sec. Higher frequency crystals have shorter start-up times (1-2ms). Start-up times are voltage dependent. The SS1102C uses the high frequency crystal for the start-up timer because it is the default clock at reset time. The counter delay is designed to be longer than the start-up time of the high frequency crystal so the fast clock will be stabilized before the chip is reset. The user's software should account for the low frequency crystal's start-up time.

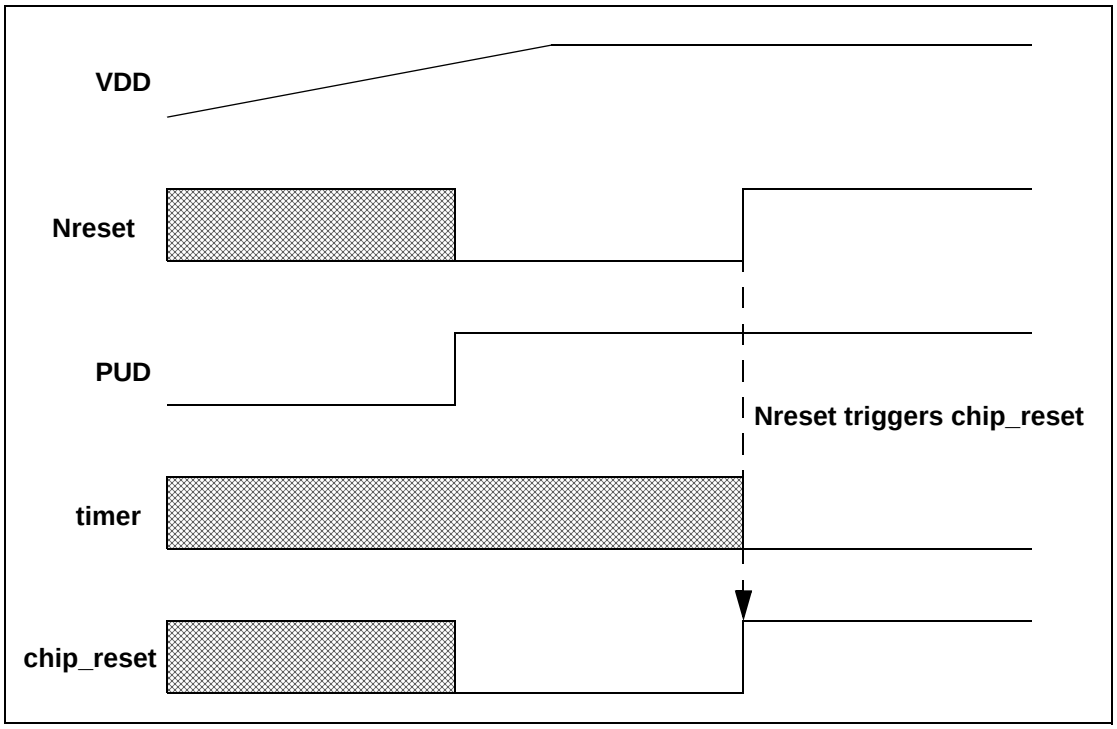


Figure 41. *External Reset Asserted*

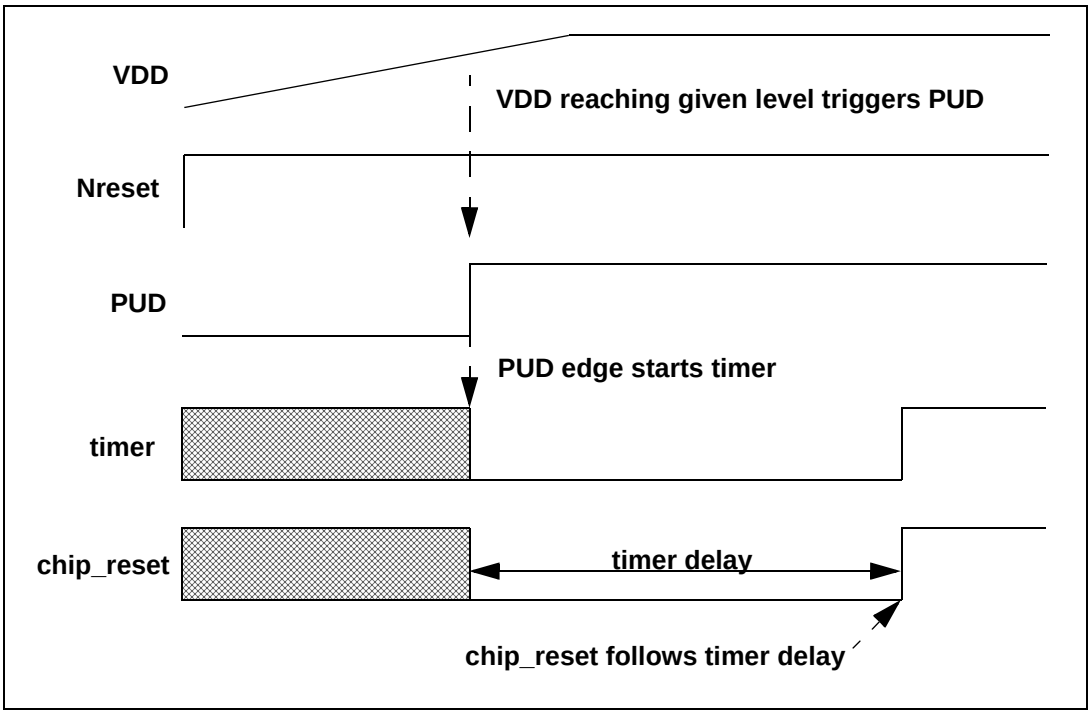


Figure 42. *Internal Reset Asserted (Nreset tied to Vcc)*

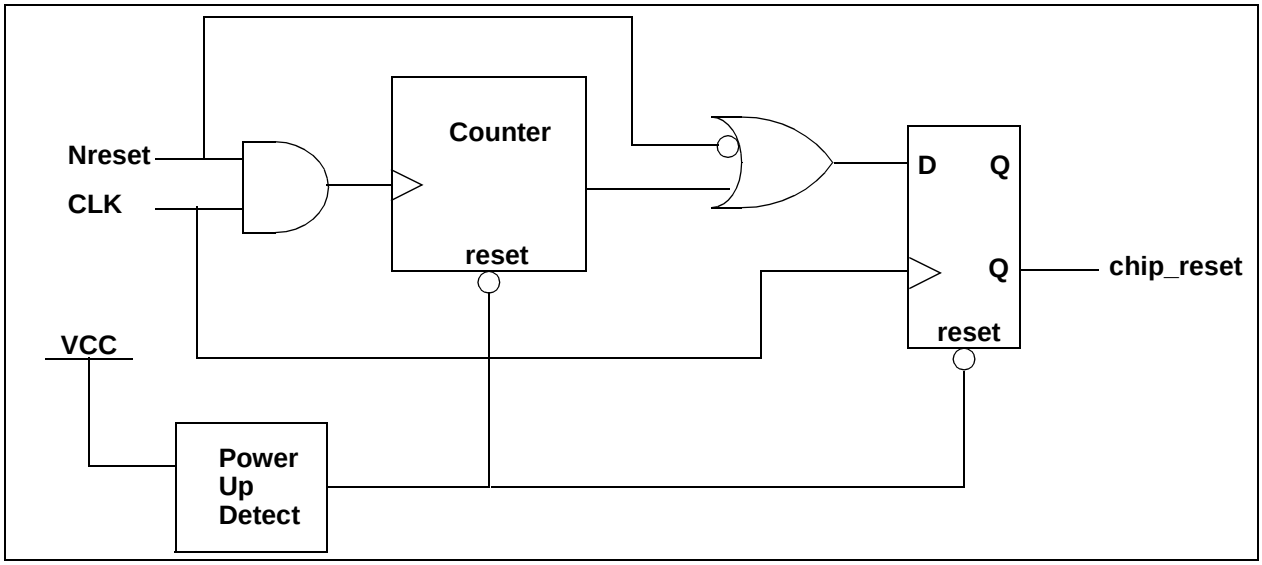


Figure 43. *Power On Reset functional diagram*

19. Low Battery Detect (LBD) - Low Voltage Reset

19.1. General Information

The low battery detect and low voltage reset circuit has two main functions.

- Used as a low battery detector, the circuit will detect the Vcc of the SS1102C falling below the selected programmable level of 2.9, 2.8 or 2.7 volts. Once detected, the circuit will send an interrupt to the SS1102C interrupt block.
- For non-battery usage the circuit can be programmed to detect Vcc falling below 2.6 volts and provide a hardware reset to the SS1102C, thus protecting the user from getting corrupted data on chip from power supply glitches below the 2.6 volt, minimum voltage level.

19.2. Functional Description

The low battery detect and low voltage reset circuit uses a programmable resistor ladder to sense the variation of the power supply Vcc. The sensed voltage is compared against a reference voltage generated on chip. The output of the comparator is used to provide an interrupt signal for the low battery detect function or the reset for the low voltage reset function. The selection of low battery detect thresholds or the low voltage reset is controlled by the microprocessor.

Once a low battery detect threshold has been detected, the circuit could be reprogrammed by the software to use the next lower voltage threshold.

The circuit may be disabled to save power between readings or when not required for a particular application.

supply voltage (Vdd)	2.7 - 3.3V and 4.5 - 5.5V
current enable mode	max. 28uA at 3.3V and max. 41uA at 5.5V supply
current disable mode	3nA at 3.3V and 83nA at 5.5V
temperature range	-20 to +85

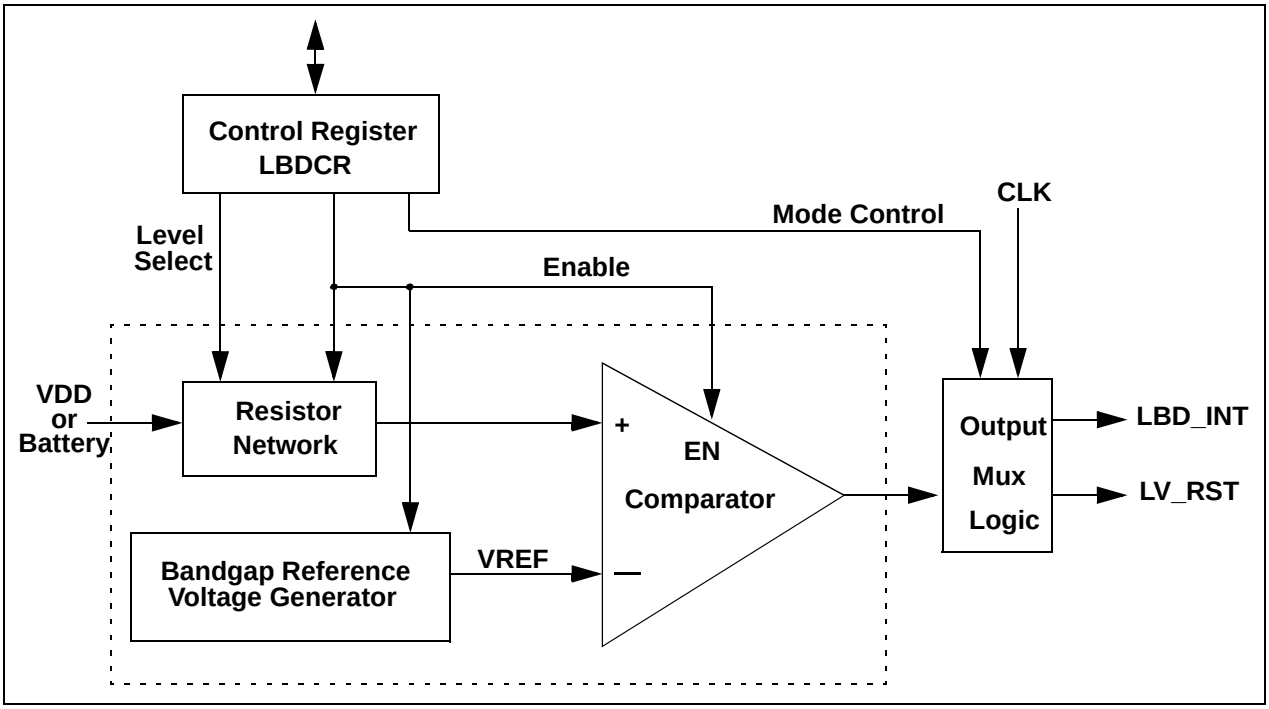


Figure 44. *Low Battery Detect Block Diagram*

19.3. Low Battery Detect, Low Voltage Reset Control Register

Address	84H							
Bit Address								
BIT	7	6	5	4	3	2	1	0
NAME	TST	cmp_out			an_enb	lvs1	lvs0	enable
Definition	test mode	test mode comparator output			i/o pad analog input enable	level select bit-1	level select bit-0	1-enable
Reset Value	0	0	0	0	0	0	0	0

Read/Write by Software	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Write by Hardware								

TABLE 77. *Definition of LBDCR*

Name	Bit	Description
TST	7	Test Mode Enable [Bit Status: 0 (Initial Value)] This status corresponds to normal operation. [Bit Status: 1] This status corresponds to test mode. In the test mode the comparator output is written to bit 6 of this control register while the interrupt and reset outputs of the LBD are disabled.
CMP_OUT	6	Test Mode Comparator Output When in test mode, bit 7 set, this register will reflect the output of the comparator.
	5 to 3	Reserved bit
AN_ENB	3	Analog input enable [Bit Status: 0 (Initial Value)] This status is to set the i/o for the SS1102C. The default is to make it analog input and disable the digital input to the port. [Bit Status: 1] This status is to enable the digital input for the i/o port.
LVS[1:0]	2 to 1	Voltage Level select bits [Bit Status: 00 (Initial Value)] This status corresponds to a low battery detect voltage of 2.9V, level 4. [Bit Status: 01] This status corresponds to a low battery detect voltage of 2.8V, level 3. [Bit Status: 10] This status corresponds to a low battery detect voltage of 2.7V, level 2. [Bit Status: 11] This status corresponds to the low voltage reset detect of 2.6V, level 1

EN	0	Low battery detect, Low voltage reset enable [Bit Status: 0 (Initial Value)] This status corresponds to the peripheral disabled. Power to the resistor network is turned off and the clock input is disabled. Current loss in the peripheral is negligible in this mode. [Bit Status: 1] This status corresponds to the peripheral enabled.

TABLE 78. *Description of LBDCR*

19.4. TIMING DIAGRAMS

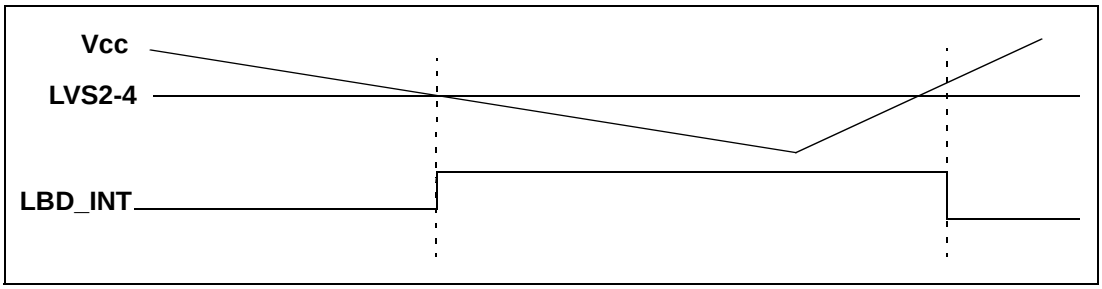


Figure 45. *Low battery detect interrupt timing*

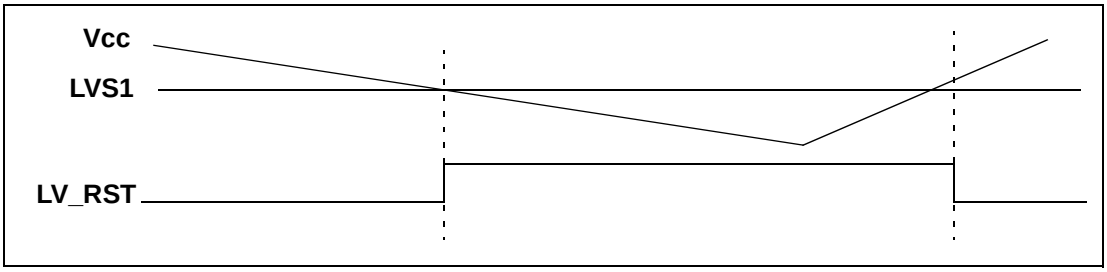


Figure 46. *Low voltage detect reset timing*

20. DC PARAMETRICS

20.1. I/O Characteristics

VIL	VIH	VOL	VOH	IOL	IOH	Vcc	Vss	Temperature
Vss + 0.8 max	Vcc - 0.8 min	Vss + 0.4 max	Vcc-0.4 min	2.96 mA @VOL=Vss+ 0.4V	3.10 mA @VOH= Vcc-0.4V	3V+/ 10%	0V	-20 C degree to +85 C degree

20.2. Power Characteristics

ICC Standby	ICC Active	Vcc	Vss	Temperature
50 uA max	30mA max	3V+(-)10%	0V	-20 C degree to +85 C degree

21. AC Specifications

SPEC	Description	Min	Max	Unit	Condition
TCYC	XFIN oscillator period	83.3		ns	
TCYCH	XFIN high	30		ns	
TCYCL	XFIN low	30		ns	
TCYCR	XFIN rise time	10		ns	
TCYCF	XFIN fall time	10		ns	
TCYS	XSIN oscillator period	31.3		us	
TCYSH	XSIN high	12		us	
TCYSL	XSIN low	12		us	
TCYSR	XSIN rise time	25		ns	
TCYSF	XSIN fall time	25		ns	

SPEC	Description	Min	Max	Unit	Condition
TOD	XFIN to output delay		41	ns	Prog pull-up off
TODRPU	XFIN to output rise delay		2.0	us	Prog pull-up on
TIRQL	minimum interrupt low		5	mc	Note: 1
TRSTL	minimum reset low		5	mc	Note: 1

TABLE 79. *TIMING TABLE*

Note: MC is machine cycle and 1 machine cycle is 12 CPU clock cycles which is either XFIN or XSIN.

21.1. TIMING WAVEFORMS

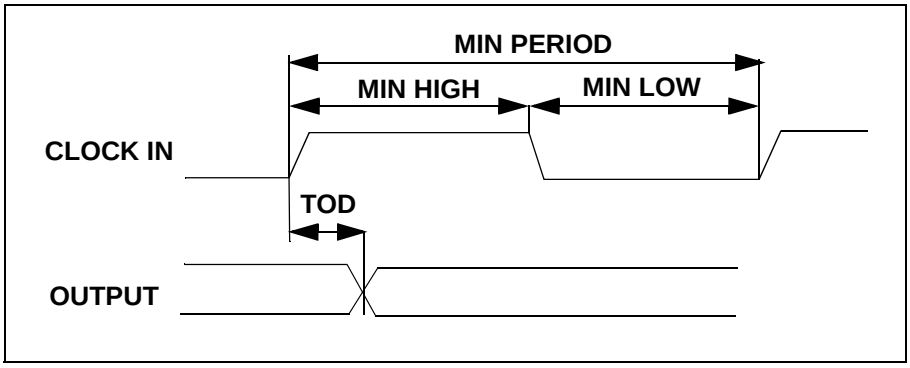


Figure 47. *Clock Input Waveforms XFIN, XSIN*

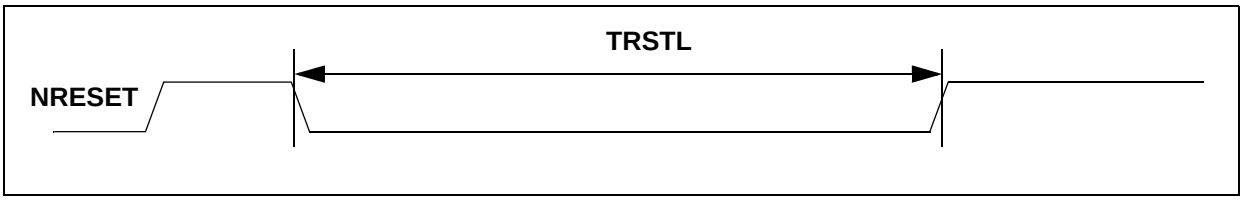


Figure 48. *Reset Timing*

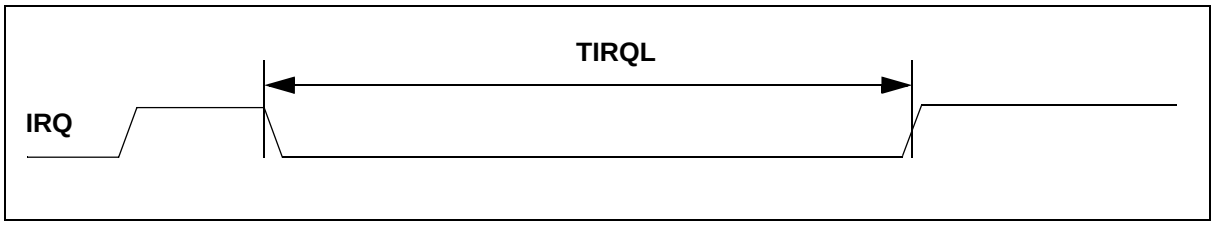
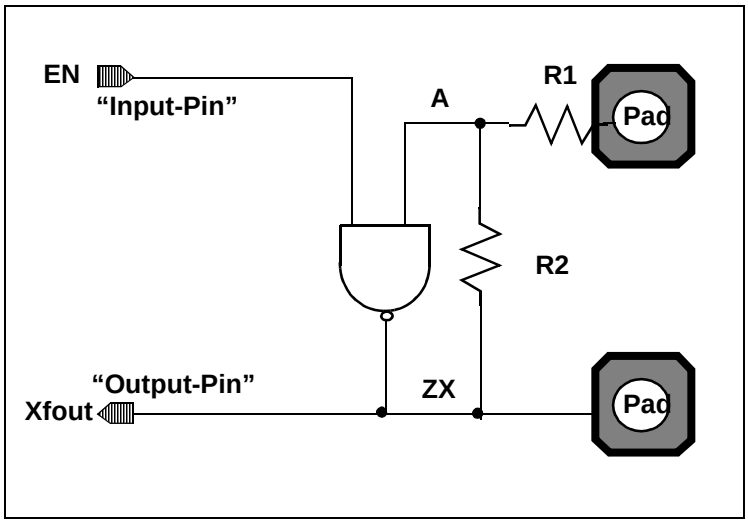


Figure 49. *Interrupt Timing*

22. Oscillator Pad Characteristics



ITEMS	Symbol	Min	Typ	Max	Unit
Supply Voltage	Vcc	3 - 10%	3	3 + 10%	V
Ground	Vss		0		V
Temperature	T	-20	25	+85	degree C
Frequency (*cell 1*)	f	9	12	25	MHz

ITEMS	Symbol	Min	Typ	Max	Unit
Frequency (*cell 2*)	f	1	32.768	60	KHz
Peak-to-Peak Voltage	Vpp			3	V
Duty cycle of Xfout		50 -1%	50%	50 +1%	

23. SS1102C I/O PAD Information

23.1. I/O Type 1 and Type 3

Description: Bidirectional CMOS I/O with active “High” Enable and Pullup-Enable and Open-Drain Enable

ITEMS	Symbol	Min	Typ	Max	Units
Supply Voltage	Vcc	3- 10%	3	3 + 10%	V
Ground	Vss		0		V
Temperature	T	-20	25	+85	degree C
POUT-to-PAD (OUTEN-to-PAD) (Output Drive: 2 mA) @ Cl=50pF	toPLH		6.5		nS
	toPHL		10.6		nS
	TR		12.6		nS
	TF		16.8		nS
PAD-to-PADIN	tiPLH		2		nS
	tiPHL		2		nS
Transistor-pullup	Rpu	50	70	100	KOhm
Output High @ IOH=200 uA	VOH	Vcc - 0.4			V
Output Low @ IOL=200 uA	VOL			Vss + 0.4	V
Leakage Current	Ileakage			1	micro A

ITEMS	Symbol	Min	Typ	Max	Units
Output Current Sink	IOL @ VOL=Vss + 0.4V	2.96			mA
Output Current Sink	IOH @ VOH=Vcc - 0.4V	3.1			mA

TABLE 80. *I/O-Type 1 and Type 3 Specifications*

23.2. I/O Type 2

Description: Bidirectional CMOS I/O with active “High” Enable and Pullup-Enable, and Open-Drain Enable.

ITEMS	Symbol	Min	Typ	Max	Units
Supply Voltage	Vcc	5 - 10%	5	5 + 10%	V
Ground	Vss		0		V
Temperature	T	-20	25	+85	degree C
POUT-to-PAD (OUTEN-to-PAD) (Output Drive: 2 mA) @ Cl=50pF	toPLH		6.5		nS
	toPHL		10.6		nS
	TR		12.6		nS
	TF		16.8		nS
PAD-to-PADIN	tiPLH		2		nS
	tiPHL		2		nS
Transistor-pullup	Rpu	50	70	100	KOhm
Output High @ IOH=200 uA	VOH	Vcc - 0.4			V
Output Low @ IOL=200 uA	VOL			Vss + 0.4	V
Leakage Current	Ileakage			1	micro A

ITEMS	Symbol	Min	Typ	Max	Units
Output Current Sink	IOL @ VOL=Vss + 0.4V	2.96			mA
Output Current Sink	IOH @ VOH=Vcc - 0.4V	3.1			mA

TABLE 81. *I/O-Type 2 Specification*

23.3. I/O Type 4: Input with static Pullup

Description: I/O-Type 4 is Input with static Pullup equivalent 70 KOhm.

ITEMS	Symbol	Min	Typ	Max	Unit
Supply Voltage	Vcc	3 - 10%	3	3 + 10%	V
Ground	Vss		0		V
Temperature	T	-20	25	+85	degree C
PAD-to-PADIN	tiPLH		2		nS
	tiPHL		2		nS
Transistor-pullup	Rpu	50	70	100	KOhm
Leakage Current	Ileakage			1	micro A
Output Current Sink	IOL @ VOL=Vss + 0.4V	2.96			mA

TABLE 82. *I/O-Type 4 Specification*

23.4. Oscillator-Pads with Enable-signal

ITEMS	Symbol	Min	Typ	Max	Unit
Supply Voltage	Vcc	3 - 10%	3	3 + 10%	V
Ground	Vss		0		V
Temperature	T	-20	25	+85	degree C
Frequency (*cell 1*)	f	1	10	25	MHz
Frequency (*cell 2*)	f	1	32.768	60	KHz
Peak-to-Peak Voltage	Vpp			5	V
Duty cycle of Xfout		50 -1%	50%	50 +1%	

TABLE 83. *Oscillator Pads Specification*

24. PAD ASSIGNMENT TABLES

Port	Pin Name	I/O Type	Function
P0.7- P0.0	P0.7- P0.0	io2	I/O
P2.7- P2.0	P2.7- P2.0	io2	I/O
P1.3- P1.0	P1.3- P1.0	io2	I/O
P4.7	I/O	io3	Latched bi-directional I/O
P4.6	I/O	io3	Latched bi-directional I/O
P4.5	I/O	io3	Latched bi-directional I/O
P4.4	I/O	io3	Latched bi-directional I/O
P4.3	I/O	io3	Latched bi-directional I/O
P4.2	LOCK	io3	Lock Indicator

P4.0	RFPWR	io3	RF power switch output/ I/O
P4.1	PLLSW	io3	Phase-lock loop switch output/ I/O
P3.3	TXEN	io3	Transmitter enable output/ I/O
P3.2	CPTRD1	io3	Capture Timer 1/Down Counter
P3.1	CPTRU1	io3	Capture Timer 1
P3.0	CPTRU2	io3	Capture Timer 2
P1.5	SPIO	io3	Serial Out
P1.6	SPIIN	io3	Serial In
P1.7	SPICLK	io3	Serial Clock
P3.6	IRQ1	io3	External Interrupt1
P3.7	IRQ2	io3	External Interrupt2
NRESET	NRESET		Reset
P1.4	LBD	io3	Low battery detect
P3.4	Modout	io3	Modulated output (TXEN tri-states)
P3.5	DI	io3	Digital (DI) / Analog (DI1) Data Input
DIREF	DIREF	io3	Analog Data Input Comparator External reference voltage
XFIN	XFIN		High frequency crystal input
XFOUT	XFOUT		High frequency crystal output
BXFOUT	BXFOUT		Buffered High Frequency Output
XSIN	XSIN		Slow frequency crystal input
XSOUT	XSOUT		Slow frequency crystal output
Vdd	Vdd		Digital Supply Voltage
Vss	Vss		Digital Ground
XVdd	Vdd		Digital Supply Voltage
XVss	Vss		Digital Ground
AVdd	AVdd		Analog Supply Voltage
Avss	AVss		Analog Ground
TST	TST		Test pin

TABLE 84. *Standard 52 Pin Device Chip*

PIN NAME	I/O TYPE	Function Description
BXSOUT		Buffered slow crystal clock output. (Disable with fast clock)
TX	input1	Voice mode transmit serial data in. Pull-up resistor. Test pin
RX		Voice mode receive serial data out. Test pin
MHZ2_ST		Voice mode clock out.
FCLK_RT		Voice mode frame clock out. Test Pin
NXROM	input1	Multichip program memory interface enable. Pull-up resistor.
MD7-0	input1	Multichip program memory data interface. Pull-up resistor.
EAD7-0		Low byte address and data I/O.
EA15-8		High byte address
EPSEN		Program store enable for program memory read active low
EALE		Address latch enable
ENWR		Data memory write active low
ENRD		Data memory read active low
ENICE	input1	In circuit emulation mode input active low. Pull-up Resistor.
ENMON	input1	Monitor mode active low. Pull-up Resistor.

TABLE 85. *Special Function Pins/Pads for Bond-out Chip (Development Version)*

25. Package

- 52-pins PQFP

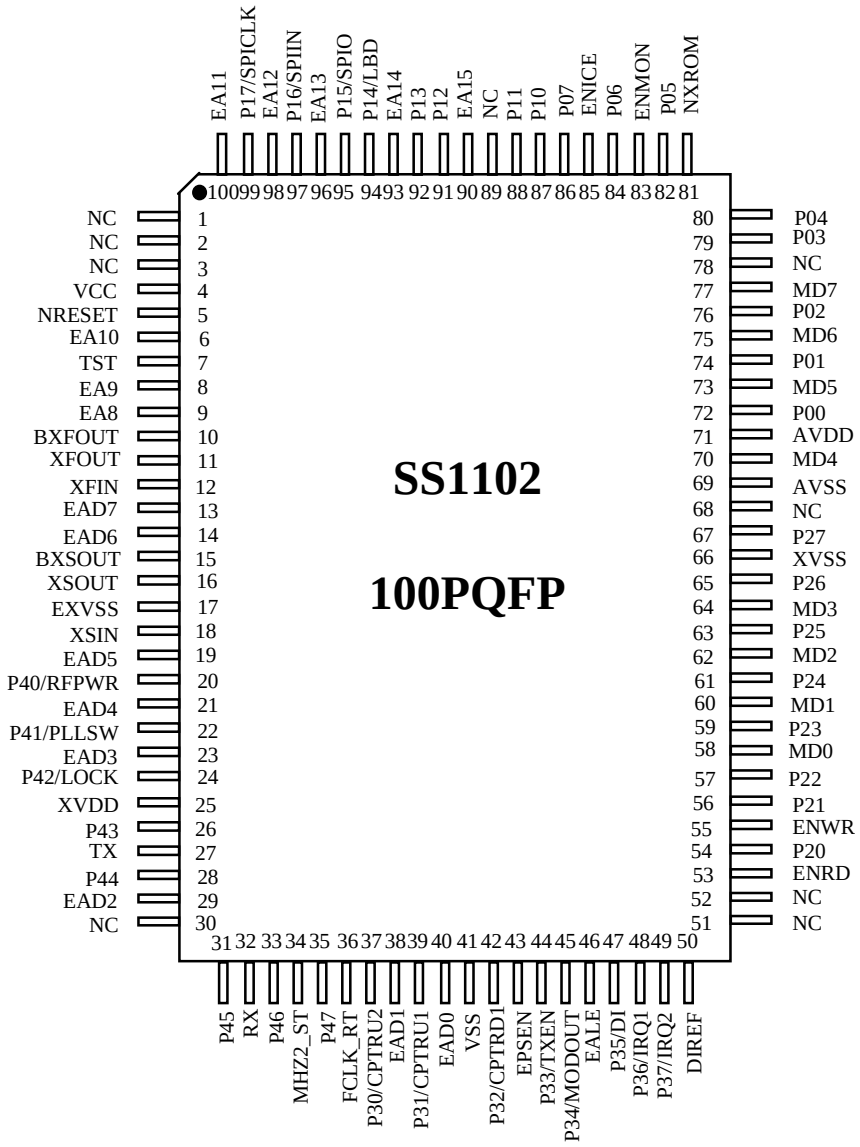
26. Operating temperature:

- 20C to +85C

27. 100-pin Chip.

PIN NAME	Function Description
BXSOUT	Buffered slow crystal clock output. (Disable with fast clock)
TX	Voice mode transmit serial data in. Pull-up resistor. Test Pin
RX	Voice mode receive serial data out. Test Pin
MHZ2_ST	Voice mode clock out. Test Pin
FCLK_RT	Voice mode frame clock out. Test Pin
NXROM	Multichip program memory interface enable. Pull-up resistor.
MD7-0	Multichip program memory data interface. Pull-up resistor.
EAD7-0	Low byte address and data I/O.
EA15-8	High byte address
EPSEN	Program store enable for program memory read active low
EALE	Address latch enable
ENWR	Data memory write active low
ENRD	Data memory read active low
ENICE	In circuit emulation mode input active low. Pull-up Resistor.
ENMON	Monitor mode active low. Pull-up Resistor.

The table above provides the description for additional functional pins of the 100-pin SS1102. The 100-pin SS1102 pinout is shown below.


SS1102
100PQFP
1 NC
2 NC
3 NC
4 VCC
5 NRESET
6 EA10
7 TST
8 EA9
9 EA8
10 BXFOUT
11 XFOUT
12 XFIN
13 EAD7
14 EAD6
15 BXSOUT
16 XSOUT
17 EXVSS
18 XSIN
19 EAD5
20 P40/RFPWR
21 EAD4
22 P41/PLLSW
23 EAD3
24 P42/LOCK
25 XVDD
26 P43
27 TX
28 P44
29 EAD2
30 NC
31 P45
32 RX
33 P46
34 MHZ2_ST
35 P47
36 FCLK_RT
37 P30/CPTRU2
38 EAD1
39 P31/CPTRU1
40 EAD0
41 VSS
42 P32/CPTRD1
43 EPSEN
44 P33/TXEN
45 P34/MODOUT
46 EALE
47 P35/DI
48 P36/IRQ1
49 P37/IRQ2
50 DIREF
51 NC
52 NC
53 ENRD
54 P20
55 ENWR
56 P21
57 P22
58 MD0
59 P23
60 MD1
61 P24
62 MD2
63 P25
64 MD3
65 P26
66 XVSS
67 P27
68 NC
69 AVSS
70 MD4
71 AVDD
72 P00
73 MD5
74 P01
75 MD6
76 P02
77 MD7
78 NC
79 P03
80 P04
81 NXROM
82 P05
83 ENMON
84 P06
85 ENICE
86 P07
87 P10
88 P11
89 NC
90 EA15
91 P12
92 P13
93 EA14
94 P14/LBD
95 P15/SPIO
96 EA13
97 P16/SPIIN
98 EA12
99 P17/SPICLK
100 EA11