

SN8F5602 Series Datasheet

8051-based Microcontroller

SN8F5602

SN8F56011

1 Device Overview

1.1 Features

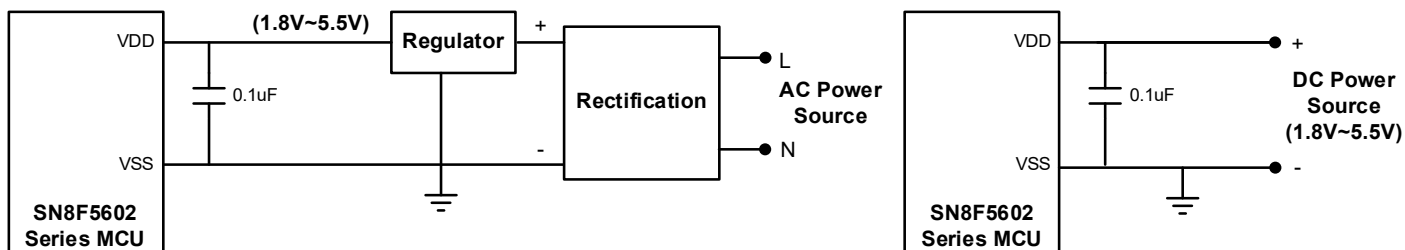
- ◆ **Enhanced 8051 microcontroller** with reduced instruction cycle time (up to 12 times 80C51)
- ◆ **Up to 16MHz flexible CPU frequency**
Internal 32 MHz Clock Generator (IHRC)
- ◆ **Memory configuration**
18 KB on-chip Flash memory (IROM)
256 bytes internal RAM (IRAM)
768 bytes external RAM (XRAM)
- ◆ **17 interrupt sources with priority levels control and unique interrupt vectors**
16 internal interrupts
1 external interrupts: INTO
- ◆ **2 set of DPTR**
- ◆ **3 set 16-bit timers with auto-reload**
- ◆ **1 set 16-bit PWM generator:**
Each PWM generator has 1 output channel with individual duty and inverter function
- ◆ **12-bit SAR ADC** with 16 external and 4 internal channels, and 5 internal reference voltages
- ◆ **Interface**
2 set UARTs with individual interrupt vector
1 set I2C with SMBus Support
- ◆ **5 set Comparators**
- ◆ **1 set OP-amp**
- ◆ **One channel 2KHz/4KHz buzzer output**
- ◆ **On-Chip Debug Support:**
Single-wire debug interface
3 hardware breakpoints
Unlimited software breakpoints
ROM data security/protection
- ◆ **Support for embedded boot loader**
- ◆ **Debug interface ON/OFF control**
- ◆ **Watchdog and programmable external reset**
- ◆ **1.7-4.5V 8 section wide range voltage detector**
- ◆ **Wide supply voltage (1.8 V – 5.5 V) and temperature (-40 °C to 105 °C) range**

1.2 Applications

- Home automation
- IH Application

1.3 Power Noise Immunity

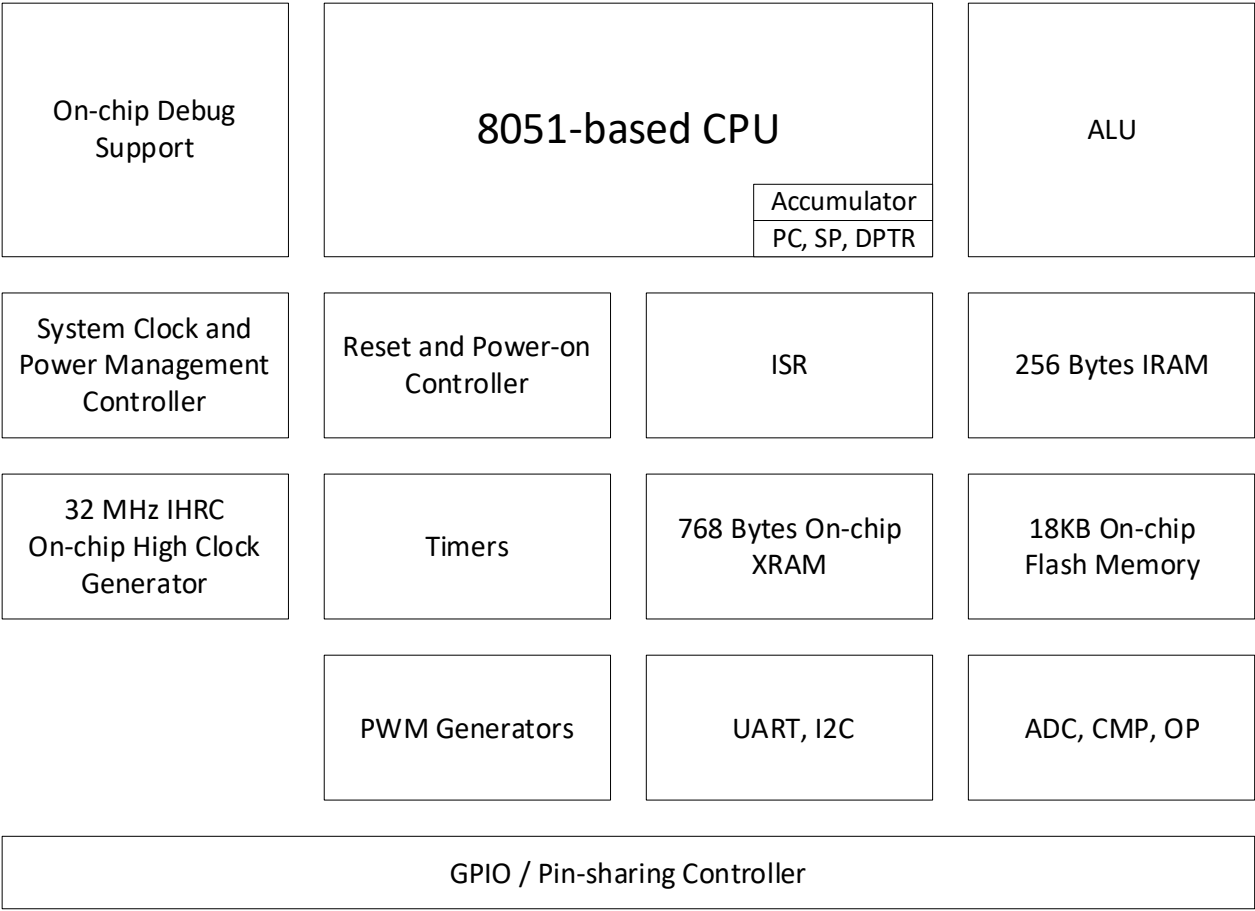
The MCU has high-performance operation with robust reliability characteristics. The power pin is VDD. VDD pin is connect to DC power source from DC-DC inverter or regulator and connects a 0.1uF capacitor to VSS pin (ground). For high power noise immunity, connect the C (0.1uF) as near as possible to VDD pin of micro-controller.



1.4 Features Selection Table

	I/O	PWM Channels	I2C	UART	ADC ext. Channels	OP	CMP	Ext. INT	Package Types
SN8F5602	18	1	1	2	16	1	5	1	PDIP20 SOP20
SN8F56011	14	1	1	2	12	1	5	1	PDIP16 SOP16

1.5 Block Diagram



2 Table of Contents

1	Device Overview.....	2
2	Table of Contents	5
3	Revision History.....	6
4	Pin Assignments	8
5	CPU	14
6	Special Function Registers.....	23
7	Reset and Power-on Controller	42
8	System Clock and Power Management.....	50
9	System Operating Mode	58
10	Interrupt.....	64
11	GPIO	74
12	External Interrupt.....	83
13	Timer 0	85
14	Timer 1	88
15	Timer 2	92
16	PWM0.....	96
17	2K/4K BUZZER GENERATOR.....	102
18	ADC.....	104
19	UART0/1	114
20	I2C	126
21	Comparator	138
22	OPA	159
23	CRC	162
24	In-System Program	165
25	Clock Fine-Tuning	170
26	ILRC Auto-Calibration	172
27	Single Wire Asynchronous Interface (SWAT)	174
28	Electrical Characteristics	175
29	Instruction Set	180
30	Development Environment	185
31	SN8F5602 Starter-Kit.....	187
32	IH Application Circuit.....	190
33	ROM Programming Pin.....	191
34	Ordering Information	194
35	Package Information	195

3 Revision History

Revision	Date	Description
1.0	Apr. 2025	First issue.
1.1	May 2025	1. IH Application Circuit update. 2. Add CALIVALENB bit description. 3. Update Starter-Kit schematic and add Power Noise Immunity chapter.
1.2	Jun. 2025	1. Remove appendix chapter. 2. Modify typing error. 3. Update Low Speed Clock chapter. 4. Modify Development Environment chapter description.
1.3	Jul. 2025	1. Add High speed clock and low speed clock chapter. 2. Modify security description. 3. Modify clock fine-tuning register. 4. Remove SN8F5000 Debug Tool Manual description.
1.4	Sep. 2025	1. Modify OP chapter. 2. Modify clock fine-tuning register.
1.5	April. 2026	1. Modify typing error.

SONIX reserves the right to make change without further notice to any products herein to improve reliability, function or design. SONIX does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. SONIX products are not designed, intended, or authorized for use as components in systems intended, for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SONIX product could create a situation where personal injury or death may occur. Should Buyer purchase or use SONIX products for any such unintended or unauthorized application. Buyer shall indemnify and hold SONIX and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that SONIX was negligent regarding the design or manufacture of the part.

4 Pin Assignments

4.1 Pin Diagrams

● SN8F5602P/S (PDIP20/SOP20)

VSS	1	U	20	VDD
BZ/INT0/P00	2		19	P10/RST/AIN0/AVREFH/URX0
PWM0/P01	3		18	P11/AIN1/UTX0
CM4N/CM0P/AIN15/P02	4		17	P12/AIN2/CM0O/SWAT/URX1 ⁽¹⁾ /UTX1 ⁽¹⁾
CM0N/AIN14/P03	5		16	P13/AIN3/CM1O/SCL ⁽¹⁾
CM3P/CM1N/AIN13/P04	6		15	P14/AIN4/CM2O/SDA ⁽¹⁾
CM2N/AIN12/P05	7		14	P15/AIN5/CM3O/URX1
SCL/URX0 ⁽¹⁾ /AIN11/P06	8		13	P16/AIN6/CM4O/UTX1
SDA/UTX0 ⁽¹⁾ /AIN10/P07	9		12	P42/AIN7/OPP
OPO/AIN9/P40	10		11	P41/AIN8/OPN

● SN8F56011S (PDIP16/SOP16)

VDD	1	U	16	P10/RST/AIN0/AVREFH/ URX0
BZ/INT0/P00	2		15	P11/AIN1/UTX0
PWM0/P01	3		14	P12/AIN2/CM0O/SWAT/URX1 ⁽¹⁾ /UTX1 ⁽¹⁾
CM4N/CM0P/AIN15/P02	4		13	VSS
CM0N/AIN14/P03	5		12	P13/AIN3/CM1O/SCL ⁽¹⁾
CM3P/CM1N/AIN13/P04	6		11	P14/AIN4/CM2O/SDA ⁽¹⁾
CM2N/AIN12/P05	7		10	P42/AIN7/OPP
OPO/AIN9/P40	8		09	P41/AIN8/OPN

*** Note: The pins which are not pin-out shall be set correctly to decrease power consumption in low-power modes. Strongly recommended to set these pins as input pull-up.**

4.2 Pin Allocation Table

● SN8F5602 (PDIP20/SOP20)

Port	Open-Drain	Wakeup	External Interrupt	ADC	ADC External Reference	UART	I2C	CMP	OP	PWM	External Reset	Debug interface
P0.0	-	V	V	-	-	-	-	-	-	-	-	-
P0.1	V*	-	-	-	-	-	-	-	-	PWM0	-	-
P0.2	-	V	-	AIN15	-	-	-	CM0P CM4N	-	-	-	-
P0.3	-	V	-	AIN14	-	-	-	CM0N	-	-	-	-
P0.4	-	V	-	AIN13	-	-	-	CM1N CM3P	-	-	-	-
P0.5	-	V	-	AIN12	-	-	-	CM2N	-	-	-	-
P0.6	V	V	-	AIN11	-	URX0 ⁽¹⁾	SCL	-	-	-	-	-
P0.7	V	V	-	AIN10	-	UTX0 ⁽¹⁾	SDA	-	-	-	-	-
P1.0	V	V	-	AIN0	AVREFH	URX0	-	-	-	-	RST	-
P1.1	V	V	-	AIN1	-	UTX0	-	-	-	-	-	-
P1.2	V	V	-	AIN2	-	UTX1 ⁽¹⁾ URX1 ⁽¹⁾	-	CM0O	-	-	-	SWAT
P1.3	-	V	-	AIN3	-	-	SCL ⁽¹⁾	CM1O	-	-	-	-
P1.4	-	V	-	AIN4	-	-	SDA ⁽¹⁾	CM2O	-	-	-	-
P1.5	V	V	-	AIN5	-	URX1	-	CM3O	-	-	-	-
P1.6	V	V	-	AIN6	-	UTX1	-	CM4O	-	-	-	-
P4.0	-	-	-	AIN9	-	-	-	-	OPO	-	-	-
P4.1	-	-	-	AIN8	-	-	-	-	OPN	-	-	-
P4.2	-	-	-	AIN7	-	-	-	-	OPP	-	-	-

*: Output only

4.3 Pin Descriptions

Power Pins

Pin Name	Type	Description
VDD	Power	Power supply
VSS	Power	Ground (0 V)

Port 0

Pin Name	Type	Description
P0.0	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
INT0	Digital Input	INT0: external interrupt 0.
BZ	Digital Output	BZ: Buzzer output.
P0.1	Digital O	GPIO: Output only pin.
PWM0	Digital Output	PWM0: programmable PWM output.
P0.2	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN15	Analog Input	ADC: input channel 15.
CM0P	Analog Input	CM0P: The positive input pin of comparator 0.
CM4N	Analog Input	CM4N: The negative input pin of comparator 4.
P0.3	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN14	Analog Input	ADC: input channel 14.
CM0N	Analog Input	CM0N: The negative input pin of comparator 0.
P0.4	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN13	Analog Input	ADC: input channel 13.
CM1N	Analog Input	CM1N: The negative input pin of comparator 1.
CM3P	Analog Input	CM3P: The positive input pin of comparator 3.
P0.5	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN12	Analog Input	ADC: input channel 12.
CM2N	Analog Input	CM2N: The negative input pin of comparator 2.
P0.6	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN11	Analog Input	ADC: input channel 11.
URX0 ⁽¹⁾	Digital Input	UART0: reception pin.
SCL	Digital I/O	I2C: clock output (master) or clock input (slave).
P0.7	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.

AIN10	Analog Input	ADC: input channel 10.
UTX0 ⁽¹⁾	Digital Input	UART0: transmission pin.
SDA	Digital I/O	I2C: data pin.

Port 1

Pin Name	Type	Description
P1.0	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
Reset	Digital Input	System reset (active low).
AIN0	Analog Input	ADC: input channel 0.
AVREFH	Analog Input	ADC: external reference voltage.
URX0	Digital Input	UART0: reception pin.
P1.1	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN1	Analog Input	ADC: input channel 1.
UTX0	Digital Output	UART0: transmission pin.
P1.2	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
SWAT	Digital I/O	Debug interface.
AIN2	Analog Input	ADC: input channel 2.
CM00	Digital Output	CM00: The output pin of comparator 0.
URX1 ⁽¹⁾	Digital Input	UART1: reception pin.
UTX1 ⁽¹⁾	Digital Output	UART1: transmission pin.
P1.3	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN3	Analog Input	ADC: input channel 3.
CM10	Digital Output	CM10: The output pin of comparator 1.
SCL ⁽¹⁾	Digital I/O	I2C: clock output (master) or clock input (slave).
P1.4	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN4	Analog Input	ADC: input channel 4.
CM20	Digital Output	CM20: The output pin of comparator 2.
SDA ⁽¹⁾	Digital I/O	I2C: data pin.
P1.5	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN5	Analog Input	ADC: input channel 5.
CM30	Digital Output	CM30: The output pin of comparator 3.
URX1	Digital Input	UART1: reception pin.

P1.6	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors. Level change wake-up.
AIN6	Analog Input	ADC: input channel 6.
CM4O	Digital Output	CM4O: The output pin of comparator 4
UTX1	Digital Output	UART1: transmission pin.

Port 4

Pin Name	Type	Description
P4.0	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors.
AIN9	Analog Input	ADC: input channel 9.
OPO	Analog Output	OPO: The output pin of OP amp.
P4.1	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors.
AIN8	Analog Input	ADC: input channel 8.
OPN	Analog Input	OPN: The negative input pin of OP amp.
P4.2	Digital I/O	GPIO: Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistors.
AIN7	Analog Input	ADC: input channel 7.
OPN	Analog Input	OPP: The positive input pin of OP amp.

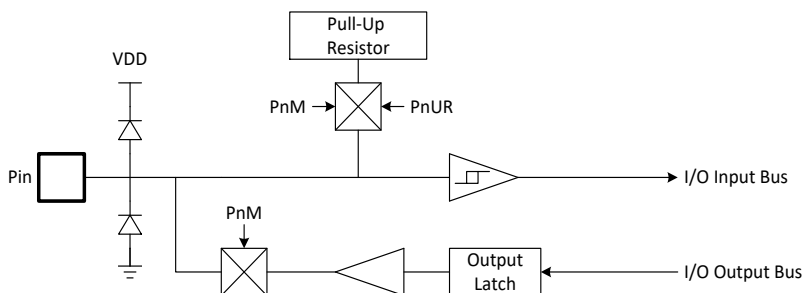
4.4 Pin Characteristic

Port	Open-Drain	Sink Current 15mA VSS+0.5V	Drive Current 15mA VSS+0.5V	External Interrupt	Wakeup (Level change)	Shared Pin
P0.0	-	V	V	V	V	INT0/BZ
P0.1	V*	V	-	-	-	PWM0
P0.2	-	V	V	-	V	AIN15/CM0P/CM4N
P0.3	-	V	V	-	V	AIN14/CM0N
P0.4	-	V	V	-	V	AIN13/CM1N/CM3P
P0.5	-	V	V	-	V	AIN12/CM2N
P0.6	V	V	V	-	V	AIN11/URX0 ⁽¹⁾ /SCL
P0.7	V	V	V	-	V	AIN10/UTX0 ⁽¹⁾ /SDA
P1.0	V	V	V	-	V	RST/AIN0/AVREFH/URX0
P1.1	V	V	V	-	V	AIN1/UTX0
P1.2	V	V	V	-	V	AIN2/CM0O/SWAT/URX1 ⁽¹⁾ /UTX1 ⁽¹⁾
P1.3	V	V	V	-	V	AIN3/CM1O/SCL ⁽¹⁾
P1.4	-	V	V	-	V	AIN4/CM2O/SDA ⁽¹⁾
P1.5	V	V	V	-	V	AIN5/CM3O/URX1
P1.6	V	V	V	-	V	AIN6/CM4O/UTX1
P4.0	-	V	V	-	-	AIN9/OPO
P4.1	-	V	V	-	-	AIN8/OPN
P4.2	-	V	V	-	-	AIN7/OPP

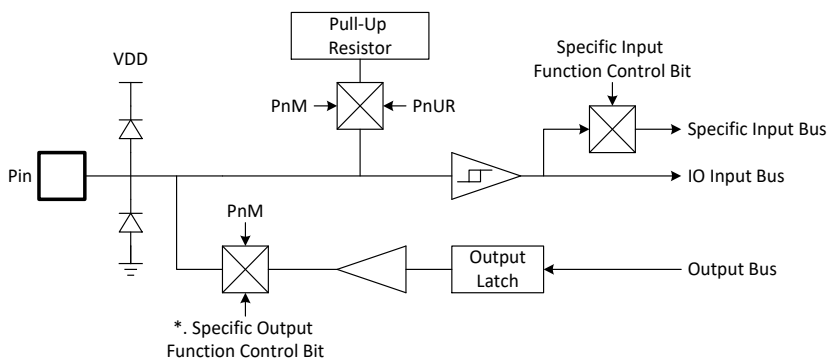
*: Output only

4.5 Pin Circuit Diagrams

Normal Bi-direction I/O Pin.

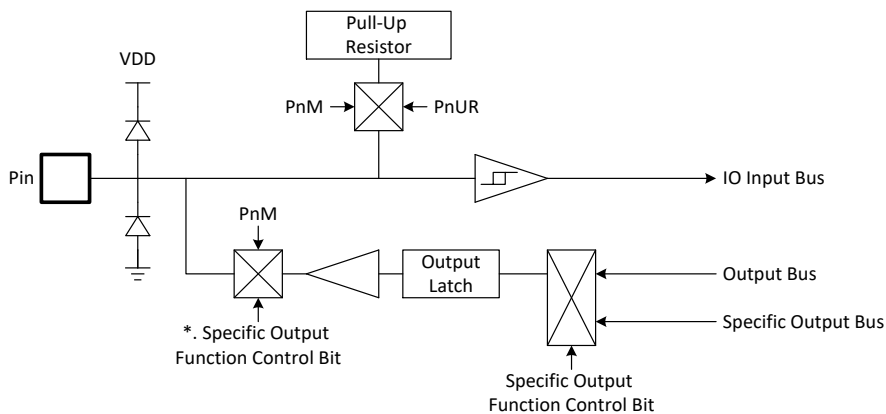


Bi-direction I/O Pin Shared with Specific Digital Input Function, e.g. INT0.



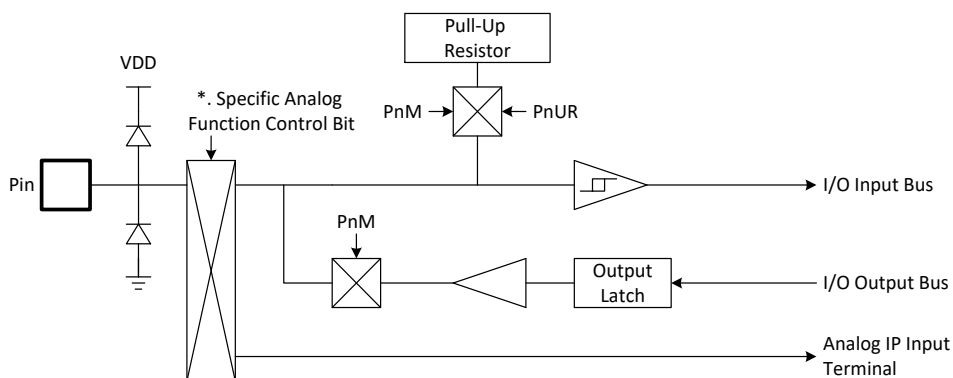
*. Some specific functions switch I/O direction directly, not through PnM register.

Bi-direction I/O Pin Shared with Specific Digital Output Function, e.g. UART.



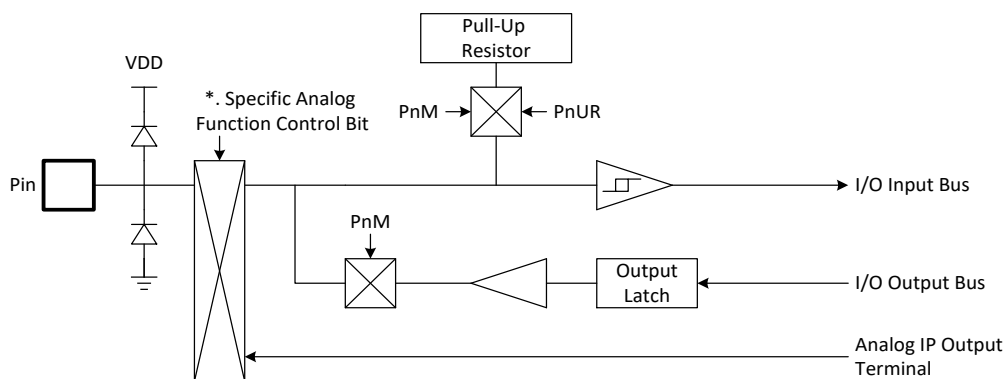
*. Some specific functions switch I/O direction directly, not through PnM register.

Bi-direction I/O Pin Shared with Specific Analog Input Function, e.g. ADC.



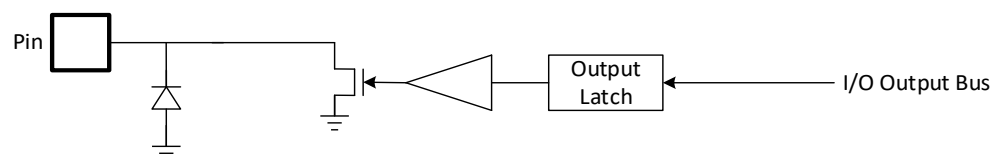
*. Some specific functions switch I/O direction directly, not through PnM register.

Bi-direction I/O Pin Shared with Specific Analog Output Function, e.g. OP...



*. Some specific functions switch I/O direction directly, not through PnM register.

Open-drain shared pin, output only: PWM0

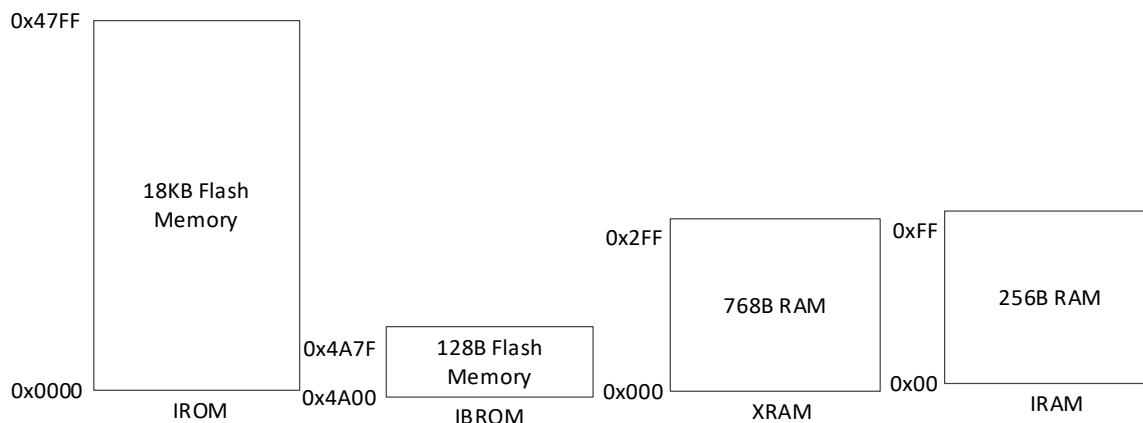


5 CPU

SN8F5000 family is an enhanced 8051 microcontroller (MCU). It is fully compatible with MCS-51 instructions, hence the ability to cooperate with modern development environment (e.g. Keil C51). SN8F5000 CPU has 9.4 to 12.1 times faster than the original 8051 at the same frequency.

5.1 Memory Organization

SN8F5602 builds in three on-chip memories: internal RAM (IRAM), external RAM (XRAM), program memory (IROM) and information block memory (IBROM). The internal RAM is a 256-byte RAM which has higher access performance (direct and indirect addressing). By contrast, the external RAM has 768-byte of size, but it requires a longer access period. The program memory is an 18 KB non-volatile memory and has a maximum 16 MHz speed limitation. The information block memory is a 128B non-volatile memory and supports user to read device tracking data or series number.



5.2 Internal RAM (IRAM)

256 X 8-bit RAM (Internal Data Memory)

Address	RAM Location	
000h	Work Register Area	
01Fh		
020h		
02Fh		
030h	Bit Addressable Area	
...		
...		
07Fh		
080h	General Purpose Area (Indirect Access)	Special Function Register (Direct Access)
...		
...		
...		
0FFh		

00h-7Fh of RAM is direct and indirect access RAM

080h-0FFh store special function registers.

End of Bank 0

The 256-byte data RAM in internal data memory is a standard 8051 RAM access configuration. The upper 128-byte RAM is general purpose RAM and can configure by direct addressing access and indirect addressing access. The lower 128-byte can be indirect access RAM in general purpose or direct access RAM in special function register (SFR).

- 0x0000-0x007F: General purpose RAM contains work register area and bit addressable area. In this area, direct or indirect addressing can be used.
- 0x0000-0x001F: Work register area includes 4-bank. Each bank has 8 work registers (R0 - R7) which is selected by RS0/RS1 in PSW register.
- 0x0020-0x002F: Bit addressable area.

In the bit addressable area, user can read or write any single bit in this range by using the unique address for that bit. Supports 16bytes bit addressable RAM area giving 128 addressable bits. Each bit has individual address in the range from 00H to 7FH. Thus, the bit can be addressed directly. Bit0 of the byte 20H has bit address 00H and Bit 7 of the byte 20H has bit address 07H. Bit0 of the byte 2FH has bit address 78H and Bit 7 of the byte 2FH has bit address 7FH. When set "SETB 42H", it means the bit2 of the byte 28H is set.

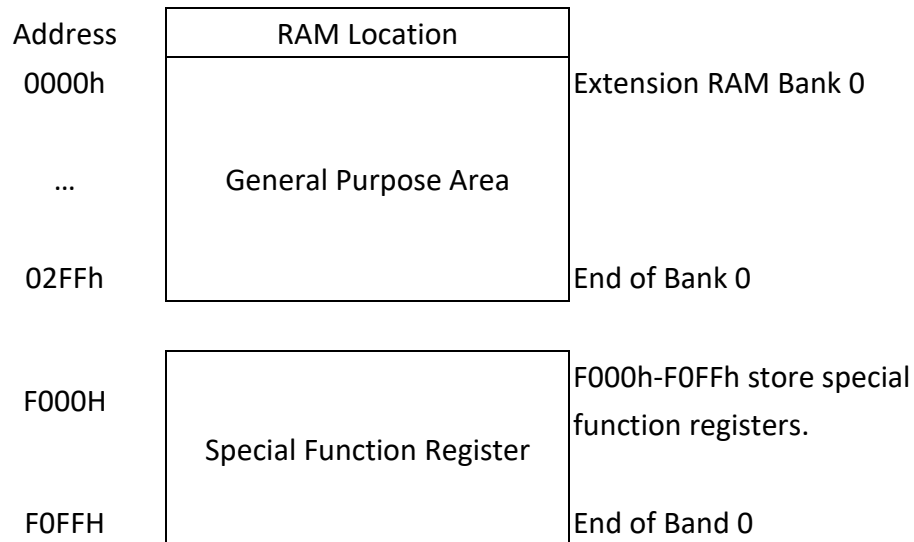
Bit Addressable Area	Byte Address	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
	0x20	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07
	0x21	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
	0x22	0x10	0x11	0x12	0x13	0x14	0x15	0x16	0x17
	0x23	0x18	0x19	0x1A	0x1B	0x1C	0x1D	0x1E	0x1F
	0x24	0x20	0x21	0x22	0x23	0x24	0x25	0x26	0x27
	0x25	0x28	0x29	0x2A	0x2B	0x2C	0x2D	0x2E	0x2F
	0x26	0x30	0x31	0x32	0x33	0x34	0x35	0x36	0x37
	0x27	0x38	0x39	0x3A	0x3B	0x3C	0x3D	0x3E	0x3F
	0x28	0x40	0x41	0x42	0x43	0x44	0x45	0x46	0x47
	0x29	0x48	0x49	0x4A	0x4B	0x4C	0x4D	0x4E	0x4F
	0x2A	0x50	0x51	0x52	0x53	0x54	0x55	0x56	0x57
	0x2B	0x58	0x59	0x5A	0x5B	0x5C	0x5D	0x5E	0x5F
	0x2C	0x60	0x61	0x62	0x63	0x64	0x65	0x66	0x67
	0x2D	0x68	0x69	0x6A	0x6B	0x6C	0x6D	0x6E	0x6F
	0x2E	0x70	0x71	0x72	0x73	0x74	0x75	0x76	0x77
	0x2F	0x78	0x79	0x7A	0x7B	0x7C	0x7D	0x7E	0x7F

- 0x0080~0x00FF: General purpose area in indirect addressing access or special function register in direct addressing access.

5.3 External RAM (XRAM)

768 X 8-bit XRAM (Extension Data Memory)

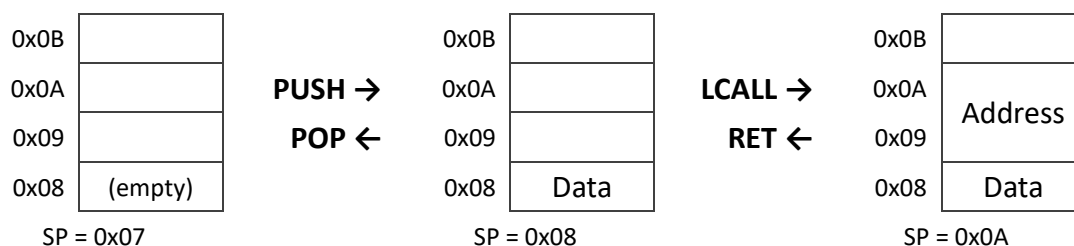
The external RAM enlarges the capacity of variables; it is the lowest access performance in the contrast of internal RAM. Since frequently used variables and local variables are expected to store in internal RAM, the vast majority of external RAM usages are specific. It can be allocated as a variable storage area for lower priority tasks, or look-up table preloaded from ROM to speed up the access period.



The upper 768-byte XRAM is general purpose RAM and can configure by MOVX instruction access. The lower 256 bytes of XRAM are special function registers (SFRs) that can also be accessed by MOVX instruction.

5.4 Stack

Stack can be assigned to any area of internal RAM (IRAM). However, it requires manual assignment to ensure its area does not overlap other RAM's variables. An overflow or underflow stack could also mistakenly overwrite other RAM's variables; thus, these factors should be considered while arrange the size of stack.



By default, stack pointer (SP register) points to 0x07 which means the stack area begin at IRAM

address 0x08. In other word, if a planned stack area is assigned from IRAM address 0xC0, the appropriate SP register is anticipated to set at 0xBF after system reset.

An assembly PUSH instruction costs one byte of stack. LCALL, ACALL instructions and interrupt respectively costs two bytes stack. POP-instruction decreases one count, and a RET/RETI subtract two counts of stack pointer.

5.5 Program Memory (IROM)

The program memory is a non-volatile storage area where stores code, look-up ROM table, and other data with occasional modification. It can be updated by debug tools like SN-Link3, and a program can also self-update via in-system program process (refer to In-system Program).

Address	ROM	Comment
0000H	Reset vector	Reset vector
0001H	General purpose area	User program
0002H		
0003H	INT0 Interrupt vector	Interrupt vector
000BH	TIMER0 Interrupt vector	
0013H	-	
001BH	TIMER1 Interrupt vector	
0023H	-	
002BH	TIMER2 Interrupt vector	
0033H	-	
003BH	-	
0043H	I2C Interrupt vector	
004BH	-	
0053H	UART0 RX Interrupt vector	
005BH	UART0 TX Interrupt vector	
0063H	UART1 RX Interrupt vector	
006BH	UART1 TX Interrupt vector	
0073H	-	
007BH	-	
0083H	PWM0 Interrupt vector	
008BH	-	
0093H	-	
009BH	-	
00A3H	-	
00ABH	ADC Interrupt vector	
00B3H	-	
00BBH	LVD Interrupt vector	
00C3H	Comparator 0 Interrupt vector	
00CBH	Comparator 1 Interrupt vector	
00D3H	Comparator 2 Interrupt vector	
00DBH	Comparator 3 Interrupt vector	

00E3H	Comparator 4 Interrupt vector	
00EBH	-	
00ECH	General purpose area	User program
.		
.		
.		
.		End of user program
47F6H	Reserved	
47F7H		
.		
47FDH		
47FEH		
47FFH		

The main block ROM includes reset vector, Interrupt vector and general purpose area. The reset vector is program beginning address. The interrupt vector is the head of interrupt service routine when any interrupt occurring. The general purpose area is main program area including main loop, sub-routines and data table.

- 0x0000 Reset vector: Program counter points to 0x0000 after any reset events (power on reset, reset pin reset, watchdog reset, LVD reset...).
- 0x0001~0x0002: General purpose area to process system reset operation.
- 0x0003~0x00EB: Multi interrupt vector area. Each of interrupt events has a unique interrupt vector.
- 0x00EC~0x47FF: General purpose area for user program and ISP (flash function).

5.6 Program Memory Security

The SN8F5602 provides security options at the disposal of the designer to prevent unauthorized access to information stored in FLASH memory. When enable security option, the ROM code is secured and not dumped complete ROM contents. ROM security rule is all address ROM data protected and outputs 0x00.

5.7 Data Pointer

A data pointer helps to specify the XRAM and IROM address while performing MOVX and MOVC instructions. The microcontroller has two set of data pointer (DPH/DPL and DPH1/DPL1) which is selectable by DPS register. The DPC register controls two functions: next DPTR selection and automatically increase/decrease DPTR function.

The next DPTR selection can specify which DPTR is anticipated to use after perform MOVX @DPTR instruction. In other word, the DPS can automatically swap between the two data pointers. To enable this function: write 0 to DPSEL and fill 1 to NDPS firstly, then write 1 to DPSEL and fill 0 to NDPS

register.

The automatically increase/decrease DPTR function can make an increment or decrement after perform MOVX @DPTR instruction. As a result, it enables a continuous external RAM access without re-specified DPTR value. Those functions are controlled by the DPC Register, where there are separate DPC register bits for each DPTR, to provide high flexibility in data transfers. The DPC Register address 0x93 points to the window where the actual DPC is selected using the DPS Register, same as for the DPTR.

5.8 Stack and Data Pointer Register

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SP	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0
DPL	DPL7	DPL6	DPL5	DPL4	DPL3	DPL2	DPL1	DPL0
DPH	DPH7	DPH6	DPH5	DPH4	DPH3	DPH2	DPH1	DPH0
DPL1	DPL17	DPL16	DPL15	DPL14	DPL13	DPL12	DPL11	DPL10
DPH1	DPH17	DPH16	DPH15	DPH14	DPH13	DPH12	DPH11	DPH10
DPS	-	-	-	-	-	-	-	DPSEL
DPC	-	-	-	-	NDPS	ATMS	ATMD	ATME

SP Register (0x81)

Bit	Field	Type	Initial	Description
7..0	SP	R/W	0x07	Stack pointer

DPL Register (0x82)

Bit	Field	Type	Initial	Description
7..0	DPL[7:0]	R/W	0x00	Low byte of DPTR0

DPH Register (0x83)

Bit	Field	Type	Initial	Description
7..0	DPH[7:0]	R/W	0x00	High byte of DPTR0

DPL1 Register (0x84)

Bit	Field	Type	Initial	Description
7..0	DPL1[7:0]	R/W	0x00	Low byte of DPTR1

DPH1 Register (0x85)

Bit	Field	Type	Initial	Description
7..0	DPH1[7:0]	R/W	0x00	High byte of DPTR1

DPS Register (0x92)

Bit	Field	Type	Initial	Description
7..1	Reserved	R	0x00	
0	DPSEL	R/W	0	DPTR selection 0: DPH/DPL (DPTR0) is selected 1: DPH1/DPL1 (DPTR1) is selected

DPC Register (0x93)

Bit	Field	Type	Initial	Description
7..4	Reserved	R	0x0	
3	NDPS	R/W	0	Next DPTR selection The DPSEL loads this bit automatically after perform any MOVX @DPTR instruction.
2..1	ATMS/ATMD	R/W	00	Automatically increase/decrease DPTR (if ATME applied) 00: +1 after any MOVX @DPTR instruction 01: -1 after any MOVX @DPTR instruction 10: +2 after any MOVX @DPTR instruction 11: -2 after any MOVX @DPTR instruction
0	ATME	R/W	0	Automatically increase/decrease DPTR function 0: Disable 1: Enable

6 Special Function Registers

The SN8F5602 offers many functions that can be controlled via special function registers. Special function registers are placed at both the IRAM address and the XRAM address. The IRAM registers can be accessed by direct addressing, while the XRAM registers are accessed via the MOVX instruction. Register description will be classified according to the addressing range. The 8-bit address refers to the IRAM register and the 16-bit address refers to the XRAM register.

6.1 IRAM Special Function Register Memory Map

BIN HEX	000	001	010	011	100	101	110	111
F8	-	P0M	P1M	-	-	P4M	-	PFLAG
F0	B	CRCM	CRCDATL	CRCDATH	SYSMOD	-	-	SRST
E8	-	-	-	-	-	-	-	-
E0	ACC	-	-	-	-	-	-	LVDCON
D8	-	BZM	I2CDAT	I2CADR	I2CCON	I2CSTA	SMBSEL	SMBDST
D0	PSW	ADCAL	ADM	ADB	ADR	VREFH	-	-
C8	IEN2	IRCON3	CMOUT	-	-	-	-	-
C0	P4	DEGCMD	-	P1W	-	-	-	-
B8	IEN1	IP1	PECMD	PEROML	PEROMH	PERAM	PERAMCNT	PEROMHH
B0	-	PW0M	PW0YL	PW0YH	PW0DL	PW0DH	PW0M1	-
A8	IEN0	IPO	FRQL	FRQH	FRQCMD	LFRQL	LFRQH	CLKCAL
A0	-	-	-	-	-	-	-	IRCON2
98	IEN3	S0BUF	S0RELL	S0RELH	S1CON	S1BUF	S1RELL	S1RELH
90	P1	-	DPS	DPC	SMODMX	SMODCAL	S0CON2	S0CON
88	TCON	-	-	-	-	-	-	PEDGE
80	P0	SP	DPL	DPH	DPL1	DPH1	WDTR	PCON

* **Note: All SFRs in the left-most column are bit-addressable. (Every 0x0/0x8-ending SFR addresses are bit-addressable).**

6.2 IRAM Special Function Register Description

0x80 - 0x9F Registers Description

Register	Address	Description
P0	0x80	Port 0 data buffer.
SP	0x81	Stack pointer register.
DPL	0x82	Data pointer 0 low byte register.
DPH	0x83	Data pointer 0 high byte register.
DPL1	0x84	Data pointer 1 low byte register.
DPH1	0x85	Data pointer 1 high byte register.
WDTR	0x86	Watchdog timer clear register.
PCON	0x87	System mode register.
TCON	0x88	Timer 0 / 1 controls register.
-	0x89	-
-	0x8A	-
-	0x8B	-
-	0x8C	-
-	0x8D	-
-	0x8E	-
PEDGE	0x8F	External interrupt edge controls register.
P1	0x90	Port 1 data buffer.
-	0x91	-
DPS	0x92	Data pointer selects register.
DPC	0x93	Data pointer controls register.
SMODMX	0x94	UART pins selection matrix.
SMODCAL	0x95	UARTs baud rate controls register.
S0CON2	0x96	UARTs baud rate control s register.
S0CON	0x97	UART0 control register.
IEN3	0x98	Interrupts enable register.
S0BUF	0x99	UART0 data buffer.
S0RELL	0x9A	UART0 reload low byte register.
S0RELH	0x9B	UART0 reload high byte register.
S1CON	0x9C	UART1 control register.
S1BUF	0x9D	UART1 data buffer.
S1RELL	0x9E	UART1 reload low byte register.
S1RELH	0x9F	UART1 reload high byte register.

0xA0 - 0xBF Registers Description

Register	Address	Description
-	0xA0	-
-	0xA1	-
-	0xA2	-
-	0xA3	-
-	0xA4	-
-	0xA5	-
-	0xA6	-
IRCON2	0xA7	Interrupts request register.
IEN0	0xA8	Interrupts enable register.
IPO	0xA9	Interrupts priority register.
FRQL	0xAA	Clock fine tuning controls buffer low byte.
FRQH	0xAB	Clock fine tuning controls buffer high byte.
FRQCMD	0xAC	Clock fine tuning command register.
LFRQL	0xAD	Low clock fine tuning controls buffer low byte.
LFRQH	0xAE	Low clock fine tuning controls buffer high byte.
CLKCAL	0xAF	ILRC auto-calibration register.
-	0xB0	-
PWM0M	0xB1	PWM0 controls register.
PW0YL	0xB2	PWM0 cycle controls buffer low byte.
PW0YH	0xB3	PWM0 cycle controls buffer high byte.
PW0DL	0xB4	PWM0 duty controls buffer low byte.
PW0DH	0xB5	PWM0 duty controls buffer high byte.
PWM0M1	0xB6	PWM0 controls register.
-	0xB7	-
IEN1	0xB8	Interrupts enable register.
IP1	0xB9	Interrupts priority register.
PECMD	0xBA	In-System Program command register.
PEROML	0xBB	In-System Program ROM address low byte.
PEROMH	0xBC	In-System Program ROM address high byte.
PERAM	0xBD	In-System Program RAM mapping address.
PERAMCNT	0xBE	ISP Program data length register
PEROMHH	0xBF	ROM access target block register

0xC0 - 0xDF Registers Description

Register	Address	Description
P4	0xC0	Interrupts request register.
DEGCMD	0xC1	Debug mode switch control register
-	0xC2	-
P1W	0xC3	Port 1 wake-up controls register.
-	0xC4	-
-	0xC5	-
-	0xC6	-
-	0xC7	-
IEN2	0xC8	Interrupts enable register.
IRCON3	0xC9	Interrupts request register.
CMOUT	0xCA	Comparator output flag register.
-	0xCB	-
-	0xCC	-
-	0xCD	-
-	0xCE	-
-	0xCF	-
PSW	0xD0	System flag register.
ADCAL	0xD1	ADC controls register.
ADM	0xD2	ADC controls register.
ADB	0xD3	ADC data buffer.
ADR	0xD4	ADC resolution selects register.
VREFH	0xD5	ADC reference voltage controls register.
-	0xD6	-
-	0xD7	-
-	0xD8	-
BZM	0xD9	Buzzer control register.
I2CDAT	0xDA	I2C data buffer.
I2CADR	0xDB	Own I2C slave address.
I2CCON	0xDC	I2C interface operation control register.
I2CSTA	0xDD	I2C Status Code.
SMBSEL	0xDE	SMBus mode controls register.
SMBDST	0xDF	SMBus internal timeout register.

0xE0 - 0xFF Registers Description

Register	Address	Description
ACC	0xE0	Accumulator register.
-	0xE1	-
-	0xE2	-
-	0xE3	-
-	0xE4	-
-	0xE5	-
-	0xE6	-
LVDCON	0xE7	LVD controls register.
-	0xE8	-
-	0xE9	-
-	0xEA	-
-	0xEB	-
-	0xEC	-
-	0xED	-
-	0xEE	-
-	0xEF	-
B	0xF0	Multiplication/ division instruction data buffer.
CRCM	0xF1	CRC control register.
CRCDATL	0xF2	CRC data low byte register.
CRCDATH	0xF3	CRC data high byte register.
SYSMOD	0xF4	System controls register.
-	0xF5	-
-	0xF6	-
SRST	0xF7	Software reset controls register.
-	0xF8	-
P0M	0xF9	Port 0 input/output mode register.
P1M	0xFA	Port 1 input/output mode register.
-	0xFB	-
-	0xFC	-
P4M	0xFD	Port 4 input/output mode register.
-	0xFE	-
PFLAG	0xFF	Reset flag register.

6.3 System Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ACC	ACC7	ACC6	ACC5	ACC4	ACC3	ACC2	ACC1	ACC0
B	B7	B6	B5	B4	B3	B2	B1	B0
PSW	CY	AC	F0	RS1	RS0	OV	F1	P

ACC Register (0xE0)

Bit	Field	Type	Initial	Description
7..0	ACC[7:0]	R/W	0x00	The ACC is an 8-bit data register responsible for transferring or manipulating data between ALU and data memory. If the result of operating is overflow (OV) or there is carry (C or AC) and parity (P) occurrence, then these flags will be set to PSW register.

B Register (0xF0)

Bit	Field	Type	Initial	Description
7..0	B[7:0]	R/W	0x00	The B register is used during multiplying and division instructions. It can also be used as a scratch-pad register to hold temporary data.

PSW Register (0xD0)

Bit	Field	Type	Initial	Description
7	CY	R/W	0	Carry flag. 0: Addition without carry, subtraction with borrowing signal, rotation with shifting out logic "0", comparison result < 0. 1: Addition with carry, subtraction without borrowing, rotation with shifting out logic "1", comparison result ≥ 0.
6	AC	R/W	0	Auxiliary carry flag. 0: If there is no a carry-out from 3rd bit of Accumulator in BCD operations. 1: If there is a carry-out from 3rd bit of Accumulator in BCD operations.
5	F0	R/W	0	General purpose flag 0. General purpose flag available for user.
4..3	RS[1:0]	R/W	00	Register bank select control bit, used to select working register bank. 00: 00H – 07H (Bnak0) 01: 08H – 0FH (Bnak1) 10: 10H – 17H (Bnak2) 11: 18H – 1FH (Bnak3)
2	OV	R/W	0	Overflow flag. 0: Non-overflow in Accumulator during arithmetic Operations. 1: overflow in Accumulator during arithmetic Operations.
1	F1	R/W	0	General purpose flag 1. General purpose flag available for user.
0	P	R	0	Parity flag. Reflects the number of '1's in the Accumulator. 0: if Accumulator contains an even number of '1's. 1: Accumulator contains an odd number of '1's.

6.4 XRAM Special Function Register Memory Map

BIN HEX	000	001	010	011	100	101	110	111
F0F8	-	-	-	-	-	-	-	-
F0F0	-	-	-	-	-	-	-	-
F0E8	-	-	-	-	-	-	-	-
F0E0	-	-	-	-	-	-	-	-
F0D8	-	-	-	-	-	-	-	-
F0D0	-	-	-	-	-	-	-	-
F0C8	-	-	-	-	-	-	-	-
F0C0	-	-	-	-	-	-	-	-
F0B8	-	-	-	-	-	-	-	-
F0B0	-	-	-	-	-	-	-	-
F0A8	-	-	-	-	-	-	-	-
F0A0	-	-	-	-	-	-	-	-
F098	-	-	-	-	-	-	-	-
F090	-	-	-	-	-	-	-	-
F088	-	-	-	-	-	-	-	-
F080	-	-	-	-	-	-	-	-
F078	-	-	-	-	-	-	-	-
F070	T2M	T2CH	T2CL	T2RH	T2RL	-	-	-
F068	T1M	T1CH	T1CL	T1RH	T1RL	-	-	-
F060	T0M	T0CH	T0CL	T0RH	T0RL	-	-	-
F058	-	-	-	-	-	-	-	-
F050	-	-	-	-	-	-	-	-
F048	-	-	-	-	-	-	-	-
F040	INTREF	-	-	-	-	-	-	-
F038	CM4M	CMDB4	CM4REF	-	-	-	-	-
F030	CM2M	CMDB2	CM2REF	-	CM3M	CMDB3	CM3REF	-
F028	CM0M	CMDB0	-	-	CM1M	CMDB1	CM1REF	-
F020	OPM	-	OPCAL	-	-	-	-	-
F018	ODCON	-	-	-	-	-	-	-
F010	P0CON	P1CON	-	-	P4CON	-	-	-
F008	-	-	-	-	-	-	-	-
F000	P0UR	P1UR	-	-	P4UR	-	-	-

* *Note: the XRAM registers are accessed via the MOVX instruction.*

6.5 XRAM Special Function Register Description

0xF000 - 0xF01F Registers Description

Register	Address	Description
P0UR	0xF000	Port 0 pull-up controls register.
P1UR	0xF001	Port 1 pull-up controls register.
-	0xF002	-
-	0xF003	-
P4UR	0xF004	Port 4 pull-up controls register.
-	0xF005	-
-	0xF006	-
-	0xF007	-
-	0xF008	-
-	0xF009	-
-	0xF00A	-
-	0xF00B	-
-	0xF00C	-
-	0xF00D	-
-	0xF00E	-
-	0xF00F	-
P0CON	0xF010	Port 0 configuration controls register.
P1CON	0xF011	Port 1 configuration controls register.
-	0xF012	-
-	0xF013	-
P4CON	0xF014	Port 4 configuration controls register.
-	0xF015	-
-	0xF016	-
-	0xF017	-
ODCON	0xF018	I/O open drain control register.
-	0xF019	-
-	0xF01A	-
-	0xF01B	-
-	0xF01C	-
-	0xF01D	-
-	0xF01E	-
-	0xF01F	-

0xF020 - 0xF03F Registers Description

Register	Address	Description
OPM	0xF020	OP controls register.
-	0xF021	-
OPCAL	0xF022	OPA calibration register.
-	0xF023	-
-	0xF024	-
-	0xF025	-
-	0xF026	-
-	0xF027	-
CM0M	0xF028	Comparator 0 controls register.
CMDB0	0xF029	Comparator 0 de-bounce register.
-	0xF02A	-
-	0xF02B	-
CM1M	0xF02C	Comparator 1 controls register.
CMDB1	0xF02D	Comparator 1 de-bounce register.
CM1REF	0xF02E	Comparator 1 reference voltage register.
-	0xF02F	-
CM2M	0xF030	Comparator 2 controls register.
CMDB2	0xF031	Comparator 2 de-bounce register.
CM2REF	0xF032	Comparator 2 reference voltage register.
-	0xF033	-
CM3M	0xF034	Comparator 3 controls register.
CMDB3	0xF035	Comparator 3 de-bounce register.
CM3REF	0xF036	Comparator 3 reference voltage register.
-	0xF037	-
CM4M	0xF038	Comparator 4 controls register.
CMDB4	0xF039	Comparator 4 de-bounce register.
CM4REF	0xF03A	Comparator 4 reference voltage register.
-	0xF03B	-
-	0xF03C	-
-	0xF03D	-
-	0xF03E	-
-	0xF03F	-

0xF40 - 0xF05F Registers Description

Register	Address	Description
INTREF	0xF040	Internal reference voltage register.
-	0xF041	
-	0xF042	
-	0xF043	
-	0xF044	
-	0xF045	
-	0xF046	
-	0xF047	
-	0xF048	
-	0xF049	
-	0xF04A	
-	0xF04B	
-	0xF04C	-
-	0xF04D	-
-	0xF04E	-
-	0xF04F	-
-	0xF050	-
-	0xF051	-
-	0xF052	-
-	0xF053	-
-	0xF054	-
-	0xF055	-
-	0xF056	-
-	0xF057	-
-	0xF058	-
-	0xF059	-
-	0xF05A	-
-	0xF05B	-
-	0xF05C	-
-	0xF05D	-
-	0xF05E	-
-	0xF05F	-

0xF60 - 0xF07F Registers Description

Register	Address	Description
T0M	0xF060	Timer 0 controls register.
T0CH	0xF061	Timer 0 counting high byte register.
T0CL	0xF062	Timer 0 counting low byte register.
T0RH	0xF063	Timer 0 reload controls buffer high byte.
T0RL	0xF064	Timer 0 cycle controls buffer low byte
-	0xF065	-
-	0xF066	-
-	0xF067	-
T1M	0xF068	Timer 1 controls register.
T1CH	0xF069	Timer 1 counting high byte register.
T1CL	0xF06A	Timer 1 counting low byte register.
T1RH	0xF06B	Timer 1 cycle controls buffer high byte.
T1RL	0xF06C	Timer 1 cycle controls buffer low byte
-	0xF06D	-
-	0xF06E	-
-	0xF06F	-
T2M	0xF070	Timer 2 controls register.
T2CH	0xF071	Timer 2 counting high byte register.
T2CL	0xF072	Timer 2 counting low byte register.
T2RH	0xF073	Timer 2 cycle controls buffer high byte.
T2RL	0xF074	Timer 2 cycle controls buffer low byte
-	0xF075	-
-	0xF076	-
-	0xF077	-
-	0xF078	-
-	0xF079	-
-	0xF07A	-
-	0xF07B	-
-	0xF07C	-
-	0xF07D	-
-	0xF07E	-
-	0xF07F	-

0xF80 - 0xF09F Registers Description

Register	Address	Description
-	0xF080	-
-	0xF081	-
-	0xF082	-
-	0xF083	-
-	0xF084	-
-	0xF085	-
-	0xF086	-
-	0xF087	-
-	0xF088	-
-	0xF089	-
-	0xF08A	-
-	0xF08B	-
-	0xF08C	-
-	0xF08D	-
-	0xF08E	-
-	0xF08F	-
-	0xF090	-
-	0xF091	-
-	0xF092	-
-	0xF093	-
-	0xF094	-
-	0xF095	-
-	0xF096	-
-	0xF097	-
-	0xF098	-
-	0xF099	-
-	0xF09A	-
-	0xF09B	-
-	0xF09C	-
-	0xF09D	-
-	0xF09E	-
-	0xF09F	-

0xF0A0 - 0xF0BF Registers Description

Register	Address	Description
-	0xF0A0	-
-	0xF0A1	-
-	0xF0A2	-
-	0xF0A3	-
-	0xF0A4	-
-	0xF0A5	-
-	0xF0A6	-
-	0xF0A7	-
-	0xF0A8	-
-	0xF0A9	-
-	0xF0AA	-
-	0xF0AB	-
-	0xF0AC	-
-	0xF0AD	-
-	0xF0AE	-
-	0xF0AF	-
-	0xF0B0	-
-	0xF0B1	-
-	0xF0B2	-
-	0xF0B3	-
-	0xF0B4	-
-	0xF0B5	-
-	0xF0B6	-
-	0xF0B7	-
-	0xF0B8	-
-	0xF0B9	-
-	0xF0BA	-
-	0xF0BB	-
-	0xF0BC	-
-	0xF0BD	-
-	0xF0BE	-
-	0xF0BF	-

0xF0C0 - 0xF0DF Registers Description

Register	Address	Description
-	0xF0C0	-
-	0xF0C1	-
-	0xF0C2	-
-	0xF0C3	-
-	0xF0C4	-
-	0xF0C5	-
-	0xF0C6	-
-	0xF0C7	-
-	0xF0C8	-
-	0xF0C9	-
-	0xF0CA	-
-	0xF0CB	-
-	0xF0CC	-
-	0xF0CD	-
-	0xF0CE	-
-	0xF0CF	-
-	0xF0D0	-
-	0xF0D1	-
-	0xF0D2	-
-	0xF0D3	-
-	0xF0D4	-
-	0xF0D5	-
-	0xF0D6	-
-	0xF0D7	-
-	0xF0D8	-
-	0xF0D9	-
-	0xF0DA	-
-	0xF0DB	-
-	0xF0DC	-
-	0xF0DD	-
-	0xF0DE	-
-	0xF0DF	-

0xF0E0 - 0xF0FF Registers Description

Register	Address	Description
-	0xF0E0	-
-	0xF0E1	-
-	0xF0E2	-
-	0xF0E3	-
-	0xF0E4	-
-	0xF0E5	-
-	0xF0E6	-
-	0xF0E7	-
-	0xF0E8	-
-	0xF0E9	-
-	0xF0EA	-
-	0xF0EB	-
-	0xF0EC	-
-	0xF0ED	-
-	0xF0EE	-
-	0xF0EF	-
-	0xF0F0	-
-	0xF0F1	-
-	0xF0F2	-
-	0xF0F3	-
-	0xF0F4	-
-	0xF0F5	-
-	0xF0F6	-
-	0xF0F7	-
-	0xF0F8	-
-	0xF0F9	-
-	0xF0FA	-
-	0xF0FB	-
-	0xF0FC	-
-	0xF0FD	-
-	0xF0FE	-
-	0xF0FF	-

6.6 Register Declaration

SN8F5602 has many registers to control various functions, but SFR name is not predefined in the C51 / A51 compiler. To make programming easier and therefore need to add header files to declare SFR name.

When using the assembly code programs, please add the following sentence.

```
1 $NOMOD51 ; Do not recognize the 8051-specific predefined special register.
2 #define __XRAM_SFR_H__ ; For XRAM Register Declaration.
3 #include <SN8F5602.H>
```

When using the C code programs, please add the following sentence.

```
1 #define __XRAM_SFR_H__ // For XRAM Register Declaration.
2 #include <SN8F5602.H>
```

After adding the header file, user can use name of registers to program. During compilation, the compiler will register name translate into register position through the header file.

Different devices need to use a different header file to declare, but the option file is to use the same.

Device	Header file	Options file
SN8F5602	SN8F5602.h	OPTIONS_SN8F5602.A51
SN8F56011	SN8F56011.h	OPTIONS_SN8F56011.A51

*** Note:**

- 1. The XRAM registers are declared in the main program file and are available for all program files.***
- 2. The XRAM registers are declared only in the main program to avoid duplicate declarations. Therefore, "# define __XRAM_SFR_H__" word string do not add to other program files, to avoid compilation errors.***
- 3. Before including the header file, you must define the "__XRAM_SFR_H__" word string to avoid compilation errors.***

6.7 XRAM Register Description

Although the XRAM register is accessed through the MOVX instruction, it is also possible to program code with the SFR name without referencing the address.

When using the assembly code program, you can refer to the following method.

```

1 MOV    DPTR, #PW1M
2 MOV    A, #08H
3 MOVX   @DPTR, A
4
5 MOV    DPTR, #POUR
6 MOVX   A, @DPTR
7 MOV    TMP, A ; TMP is direct addressing variable

```

When using the C code program, you can refer to the following method.

```

1 PW0M = 0x08;
2 TMP = POUR; // TMP is direct addressing variable

```

When using assembly code, there is another way to write code faster. The header file has two macros to provide quick and easy programming.

```

1 WRXSFR XRAM register, direct addressing variable (or register, #data)
2 RDXSFR direct addressing variable (or register), XRAM register

```

Example:

```

1 WRXSFR PW0M, #08H
2 WRXSFR POUR, B
3 WRXSFR PW0YH, TMP ; TMP is direct addressing variable
4
5 RDXSFR B, PW0M
6 RDXSFR TMP, POUR ; TMP is direct addressing variable

```

★ **Note:**

1. Using macro does not increase the speed of instructions, but simplifies the program code.

2. The macro is already included in the header file and can be used directly.

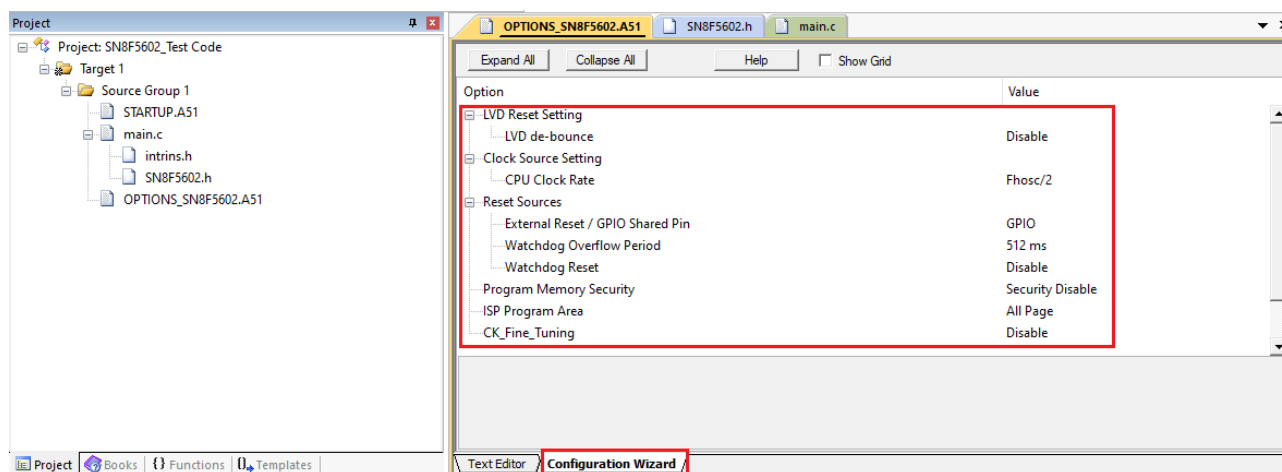
7 Reset and Power-on Controller

The reset and power-on controller has four reset sources: low voltage detectors (LVDs), watchdog, programmable external reset pin, and software reset. The first three sources would trigger an additional power-on sequence. Subsequently, the microcontroller initializes all registers and starts program execution with its reset vector (ROM address 0x0000).

7.1 Configuration of Reset and Power-on Controller

SONiX publishes a *SN8F5602_OPTIONS.A51* file in *SN-Link Driver for Keil C51.exe* (downloadable on cooperative website: www.sonix.com.tw). This *options* file contains appropriate parameters of reset sources and CPU clock source selection, and is strongly recommended to add to Keil project. The option items are as following:

- LVD Reset Setting
- CPU Clock Source
- Program Memory Security
- ISP Program Area
- CK_Fine_Tuning
- Reset Source : External Reset / GPIO Shared Pin
- Reset Source : Watchdog Reset & Overflow Period



The code option is the system hardware configurations including watchdog timer operation, LVD option, reset pin option and flash ROM security control. The code option items are as following table:

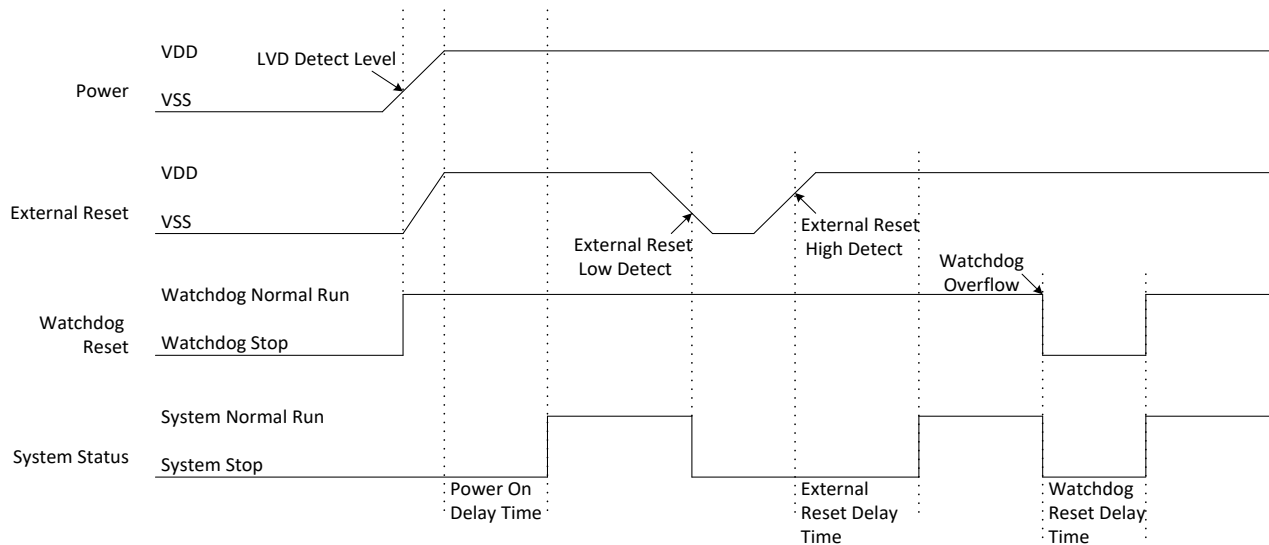
Code Option	Content	Function Description
-------------	---------	----------------------

Program Memory Security	Security Disable	Disable ROM code Security function
	Security Enable	Enable ROM code Security function
	Security Configuration	All address ROM data are protected expect address 0x4680 ~ 0x477F, only address 0x4680 ~ 0x477F ROM data can be accessed
ISP Program Area	All Page	All address can perform ISP function
	Page 141~ Page 142	Only address 0x4680 ~ 0x477F, can perform ISP function
CPU Clock Rate	001 = $F_{osc} / 2$ 010 = $F_{osc} / 4$ 011 = $F_{osc} / 8$ 100 = $F_{osc} / 16$ 101 = $F_{osc} / 32$ 110 = $F_{osc} / 64$ 111 = $F_{osc} / 128$	Fcpu clock rate
CK_Fine_Tuning	Disable	Disable CK_Fine_Tuning
	Enable	Enable CK_Fine_Tuning
LVD De-bounce	Disable	LVD 1.7V reset without De-bounce.
	4* F_{ILRC}	LVD 1.7V reset with 4* F_{ILRC} De-bounce.
	8* F_{ILRC}	LVD 1.7V reset with 8* F_{ILRC} De-bounce.
	16* F_{ILRC}	LVD 1.7V reset with 16* F_{ILRC} De-bounce.
External Reset	Reset with De-bounce	Enable External reset pin with De-bounce
	Reset without De-bounce	Enable External reset pin without De-bounce
	GPIO	Enable GPIO
Watchdog Reset	Always	Watchdog timer is always on enable even in STOP mode and IDLE mode
	Enable	Enable watchdog timer. Watchdog timer stops in STOP mode and IDLE mode
	Disable	Disable Watchdog function
Watchdog Overflow Period	64ms	Watchdog timer clock source $F_{ILRC} / 4$
	128ms	Watchdog timer clock source $F_{ILRC} / 8$
	256ms	Watchdog timer clock source $F_{ILRC} / 16$
	512ms	Watchdog timer clock source $F_{ILRC} / 32$

7.2 Power-on Sequence

A power-on sequence would be triggered by LVD, watchdog, and external reset pin. It takes place

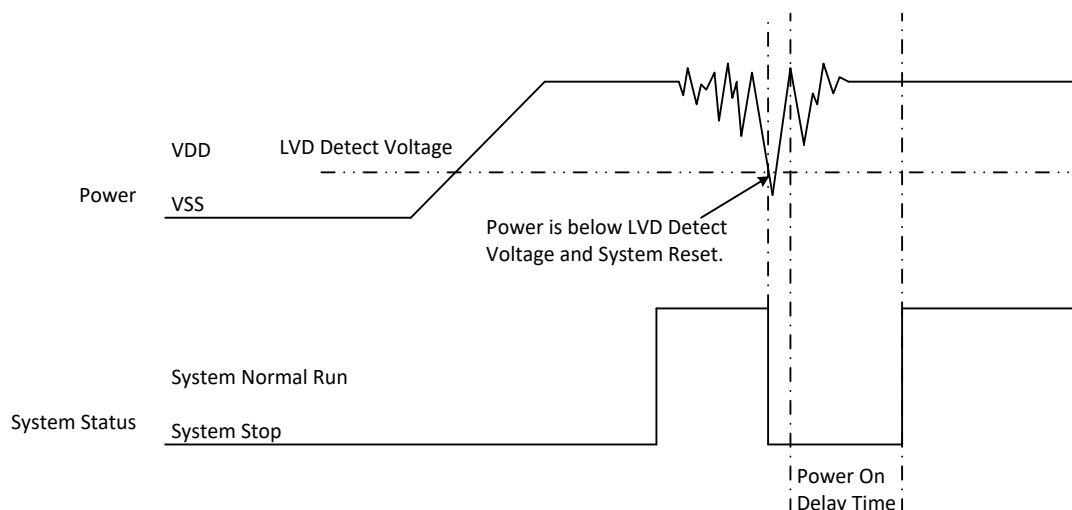
between the end of reset signal and program execution. Overall, it includes two stages: power stabilization period, and clock stabilization period.



The power stabilization period spends 5.7 ms in typical condition. Afterward the microcontroller fetches CPU Clock Source selection automatically. The selected clock source would be driven, and the system counts 2048 times of the clock period and 5 times of the internal low-speed oscillator clocks to ensure its reliability.

7.3 LVD Reset

The low voltage detectors monitor VDD pin's voltage at only one level: 1.7 V. Depend on low voltage detection configuration, the comparison result can be seen as a system reset signal. The table below lists low voltage detection, and the results of VDD pin's condition.



Condition	LVD
$VDD \leq 1.7V - 4.5V$	Reset

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVDCON	-	-	INTLVL2	INTLVL1	INTLVL0	RSTLVL2	RSTLVL1	RSTLVL0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
IEN2	ECMP4	ECMP3	ECMP2	ECMP1	ECMP0	ET2	EADC	ELVD
IRCON2	CM4F	CM3F	CM2F	CM1F	CM0F	TF2	ADCF	LVDF

LVDCON Register (0xE7)

Bit	Field	Type	Initial	Description
7..6	Reserved	R	0x0	
5..3	INTLVL[2:0]	R/W	000	LVD flag voltage by selection 000: 1.7V 001: 2.1V 010: 2.5V 011: 2.9V

				100: 3.3V
				101: 3.7V
				110: 4.1V
				111: 4.5V
2..0	RSTLVL[2:0]	R/W	000	LVD reset voltage by selection
				000: 1.7V
				001: 2.1V
				010: 2.5V
				011: 2.9V
				100: 3.3V
				101: 3.7V
				110: 4.1V
				111: 4.5V

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
	Else			Refer to other chapter(s)

IEN2 Register (0xC8)

Bit	Field	Type	Initial	Description
0	ELVD	R/W	0	LVD interrupt control bit. 0: Disable LVD interrupt function. 1: Enable LVD interrupt function.
	Else			Refer to other chapter(s)

IRCON2 Register (0xA7)

Bit	Field	Type	Initial	Description
0	LVDF	R/W	0	LVD interrupt request flag. 0: None LVD interrupt request 1: LVD interrupt request.
	Else			Refer to other chapter(s)

7.4 Watchdog Reset

Watchdog is a periodic reset signal generator for the purpose of monitoring the execution flow. Its internal timer is expected to be cleared in a check point of program flow; therefore, the actual reset signal would be generated only after a software problem occurs. Writing 0x5A to WDTR is the proper method to place a check point in program.

```
1 WDTR = 0x5A;
```

Watchdog timer interval time = $256 * 1 / (\text{Internal Low-Speed oscillator frequency} / \text{WDT Pre-scalar})$
 $= 256 / (F_{ILRC} / \text{WDT Pre-scalar}) \dots \text{sec}$

Internal low-speed oscillator	WDT pre-scalar	Watchdog interval time
$F_{ILRC} = 16 \text{ kHz}$	$F_{ILRC} / 4$	$256 / (16000 / 4) = 64 \text{ ms}$
	$F_{ILRC} / 8$	$256 / (16000 / 8) = 128 \text{ ms}$
	$F_{ILRC} / 16$	$256 / (16000 / 16) = 256 \text{ ms}$
	$F_{ILRC} / 32$	$256 / (16000 / 32) = 512 \text{ ms}$

The operation mode of watchdog is configurable in options file:

Always mode counts its internal timer in all CPU operation modes (normal, IDLE, SLEEP);

Enable mode counts its internal timer during CPU stays in normal mode, and it would not trigger watchdog reset in IDLE and STOP modes;

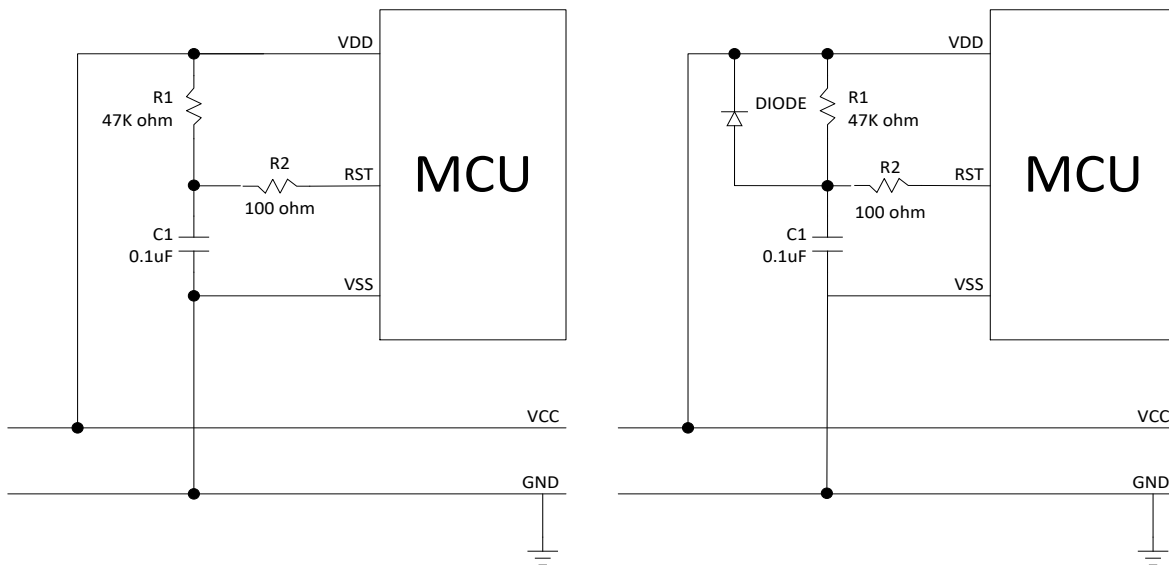
Disable mode suspends its internal timer at all CPU modes, and the watchdog would not trigger in this condition.

When watchdog is operating in always mode, the system will consume additional power.

7.5 External Reset Pin

Programmable external reset pin is configurable in *options file*. Once it is enabled, it monitors its shared pin's logic level. A logical low (lower than 30% of VDD) would immediately trigger system reset until the input is recovered to high (larger than 70% of VDD).

An optional de-bounce period can improve reset signal's stability. Instead of immediate reset, the system reset requires an 8-ms-long logic low to avoid bouncing from a button key. Any signal lower than de-bounce period would not affect the CPU's execution.



* **Note:**

1. The reset circuit is no any protection against unusual power or brown out reset on the left side of the figure.
2. The R2 100 ohm resistor of “Simply reset circuit” and “Diode & RC reset circuit” is necessary to limit any current flowing into reset pin from external capacitor C in the event of reset pin breakdown due to Electrostatic Discharge (ESD) or Electrical Over-stress (EOS) on the right side of the figure.

7.6 Software Reset

A software reset would be generated after consecutively set SRSTREQ register. As a result, this procedure enables firmware’s ability to reset microcontroller (e.g. reset after firmware update). The following sample C code repeatedly set the least bit of SRST register to perform software reset.

```
1  SRST = 0x01;
2  SRST = 0x01;
```

7.7 Reset and Power-on Controller Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PFLAG	POR	WDT	RST	-	FLSHR	-	-	-
SRST	-	-	-	-	-	-	-	SRSTREQ
WDTR	WDTR7	WDTR6	WDTR5	WDTR4	WDTR3	WDTR2	WDTR1	WDTR0

PFLAG Register (0xFF)

Bit	Field	Type	Initial	Description
7	POR	R	-	This bit is automatically set if the microcontroller has been reset by LVD.
6	WDT	R	-	This bit is automatically set if the microcontroller has been reset by watchdog.
5	RST	R	-	This bit is automatically set if the microcontroller has been reset by external reset pin.
4	-	R	0	-
3	FLSHR	R	1	This bit is automatically set if the microcontroller has been reset successfully.
2..0	Reserved	R	0	

SRST Register (0xF7)

Bit	Field	Type	Initial	Description
7..1	Reserved	R	0	
0	SRSTREQ	R/W	-	Read: This bit is automatically set if the microcontroller has been reset by software reset. Write: Consecutively set this bit for two times to trigger software reset.

WDTR Register (0x86)

Bit	Field	Type	Initial	Description
7..0	WDTR[7:0]	W	-	Watchdog clear is controlled by WDTR register. Moving 0x5A data into WDTR is to reset watchdog timer.

8 System Clock and Power Management

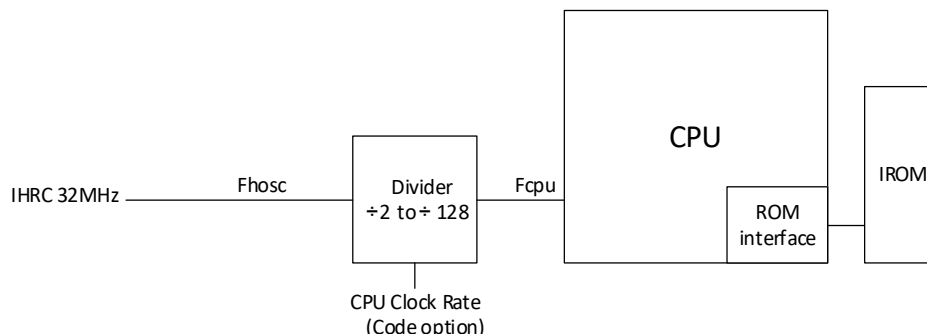
For power saving purpose, the microcontroller built in three different operation modes: normal, IDLE, and STOP mode.

The normal mode means that CPU and peripheral functions are under normally execution. The system clock is based on the combination of source selection, clock divider, and program memory wait state. IDLE mode is the situation that temporarily suspends CPU clock and its execution, yet it remains peripherals' functionality (e.g. timers, PWM, UART, and I2C). STOP mode disables all functions and clock generator until a wakeup signal to return normal mode.

8.1 System Clock

The microcontroller includes an on-chip clock generator (IHRC 32MHz). The reset and power-on controller automatically loads clock source selection during power-on sequence. Therefore, the selected clock source is seen as 'Fhosc' domain which is a fixed frequency at any time.

Subsequently, the selected clock source (Fhosc) is divided by 2 to 128 times which is controlled by code option.



ROM interface is built in between CPU and IROM (program memory). It optionally extends the data fetching cycle in order to support lower speed program memory.

$$\text{IROM fetching cycle (Instruction cycle)} \leq 16\text{MHz}$$

* **Note:** CPU clock source from IHRC limit divider = 2, CPU clock source from others don't limit divider.

System clock rate and program memory extended cycle limitation as follows.

	CPU Clock Source
CPU Clock Rate	IHRC 32M
000 = Fosc / 1	-
001 = Fosc / 2	V
010 = Fosc / 4	V
011 = Fosc / 8	V
100 = Fosc / 16	V
101 = Fosc / 32	V
110 = Fosc / 64	V
110 = Fosc / 128	V

8.2 High Speed Clock

High-speed clock has only internal type. The internal high-speed oscillator is 32MHz RC type. The high-speed oscillator is fixed.

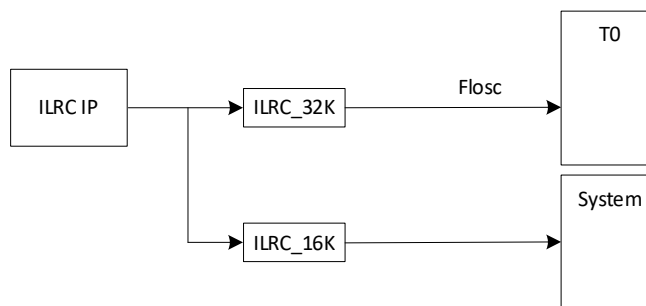
Mode	Clock Source Type	Frequency	XIN/XOUT Pin	External Components	Capacitor to GND
IHRC 32M	Internal RC oscillator	32 MHz	XIN/XOUT = GPIO	None	N/A

8.3 Low Speed Clock

SN8F5602 supplies low speed clock (Fosc) for specific functions, such as T0. The Fosc has one clock source selection: internal 32 kHz RC (ILRC_32K).

Low-speed clock has only one internal type. The internal low-speed oscillator is RC type which can support two kinds of frequency, ILRC_16K and ILRC_32K. ILRC_16K is only used for system operation.

Mode	Clock Source Type	Frequency	XIN/XOUT Pin	External Components	Capacitor to GND
ILRC 32K	Internal RC oscillator	32 KHz	XIN/XOUT = GPIO	None	N/A



8.4 Power Management

After the end of reset signal and power-on sequence, the CPU starts program execution at the speed of F_{cpu} . Overall, the CPU and all peripherals are functional in this situation (categorized as normal mode).

The least two bits of PCON register (IDLE at bit 0 and STOP at bit 1) control the microcontroller's power management unit.

If IDLE bit is set by program, only CPU clock source would be gated. Consequently, peripheral functions (such as timers, PWM, and I2C) and clock generator (IHRC 32 MHz/crystal driver) remain execution in this status. Any change from P0/P1 input and interrupt events can make the microcontroller turns back to normal mode, and the IDLE bit would be cleared automatically.

If STOP bit is set, by contrast, CPU, peripheral functions, and clock generator are suspended. Data storage in registers and RAM would be kept in this mode. Any change from P0/P1 can wake up the microcontroller and resume system's execution. STOP bit would be cleared automatically.

For user who is develop program in C language, IDLE and STOP macros is strongly recommended to control the microcontroller's system mode, instead of set IDLE and STOP bits directly.

```

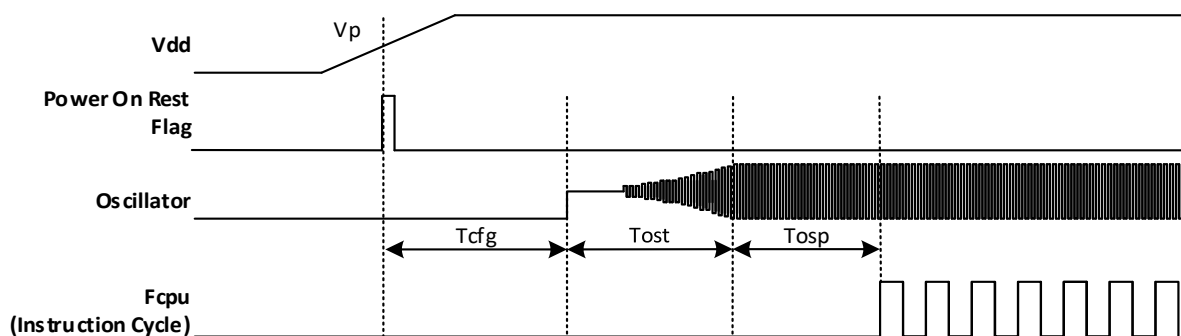
1  IDLE();
2  STOP();
  
```

SN8F5602 build in STWK bit (the 0th bit in SYSMOD register) to enable or disable Fosc clock in STOP mode. If STWK=0, both Fosc and Fosc are suspended in STOP mode. If STWK=1, Fosc clock keeps running in STOP mode.

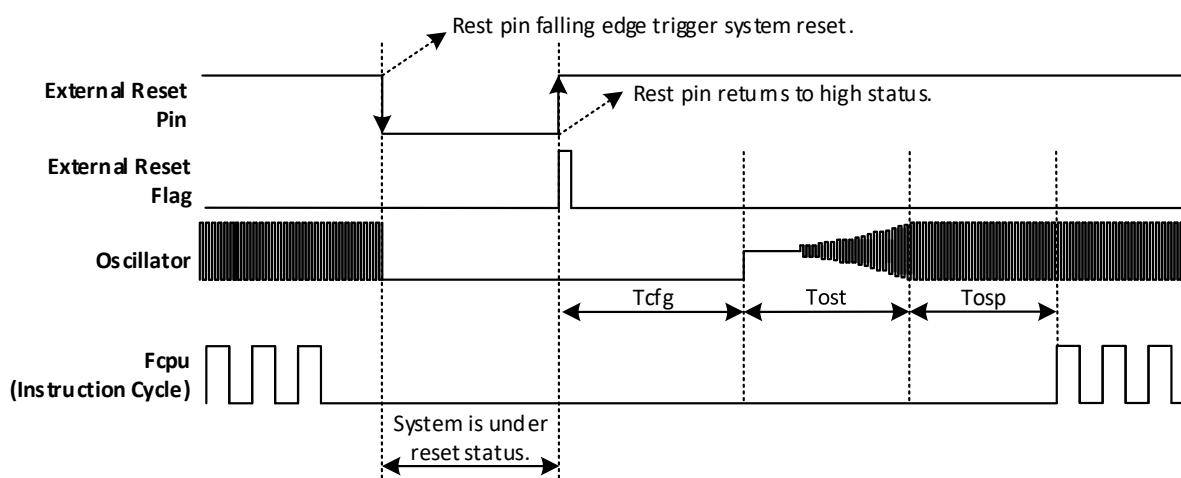
8.5 System Clock Timing

Parameter	Symbol	Description	Typical
Hardware configuration time	Tcfg	$11 * F_{ILRC} + 3 * 2^{15} * F_{IHRC}$	3.76ms @ $F_{ILRC} = 16\text{kHz}$ & $F_{IHRC} = 32\text{MHz}$
Oscillator start up time	Tost	The start-up time is depended on oscillator's material, factory and architecture. Normally, the low-speed oscillator's start-up time is lower than high-speed oscillator. The RC type oscillator's start-up time is faster than crystal type oscillator.	
Oscillator warm-up time	Tosp	Oscillator warm-up time of reset condition. 2ms.....RC type oscillator, e.g. internal high-speed RC type oscillator. (Power on reset, LVD reset, watchdog reset, external reset pin active.)	2ms @ $F_{hosc} = 32\text{MHz}$
		Oscillator warm-up time of power down mode wake-up condition. $4 * F_{ILRC}$RC type oscillator, e.g. internal high-speed RC type oscillator.	250us @ $F_{hosc} = 32\text{MHz}$
		Oscillator warm-up time of idle down mode wake-up condition. $3 * F_{ILRC}$RC type oscillator, e.g. internal high-speed RC type oscillator.	188us @ $F_{ILRC} = 16\text{kHz}$ & $F_{IHRC} = 32\text{MHz}$

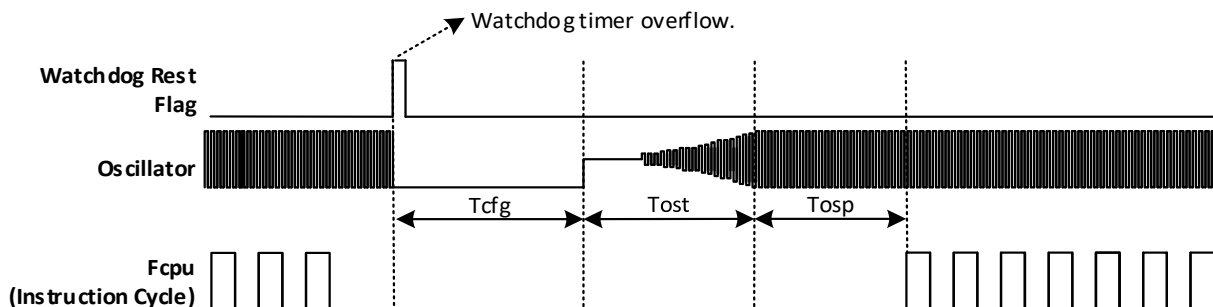
● Power On Reset Timing



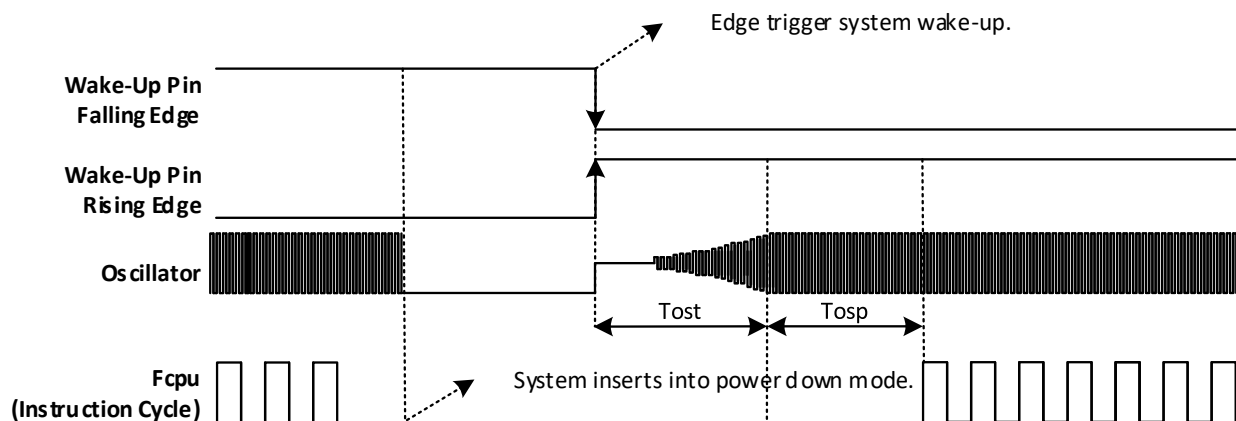
● External Reset Pin Reset Timing



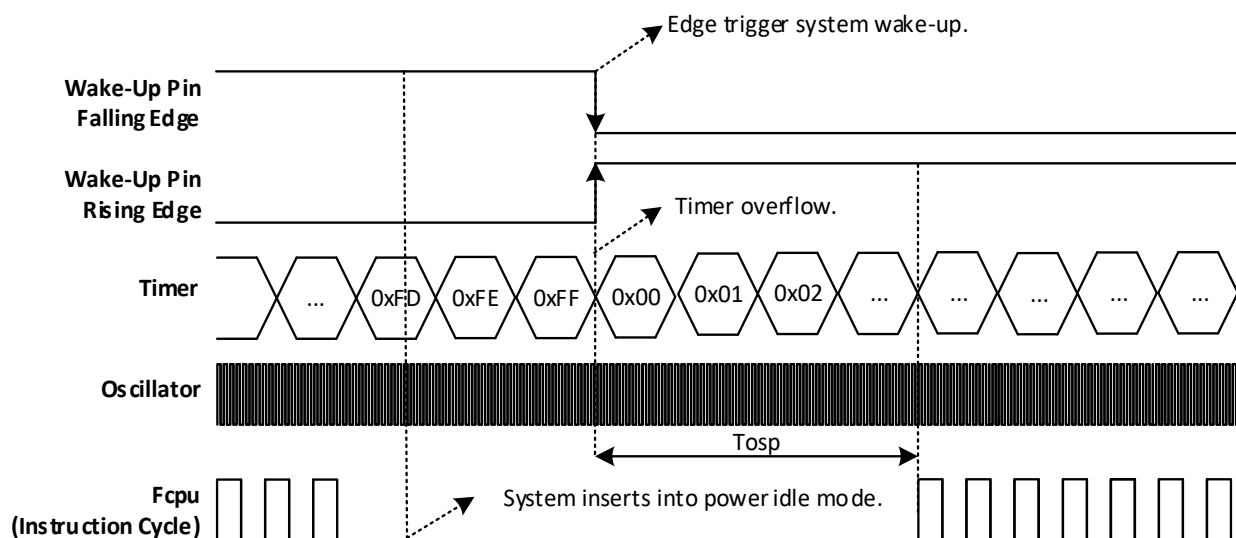
● Watchdog Reset Timing



● STOP Mode Wake-up Timing

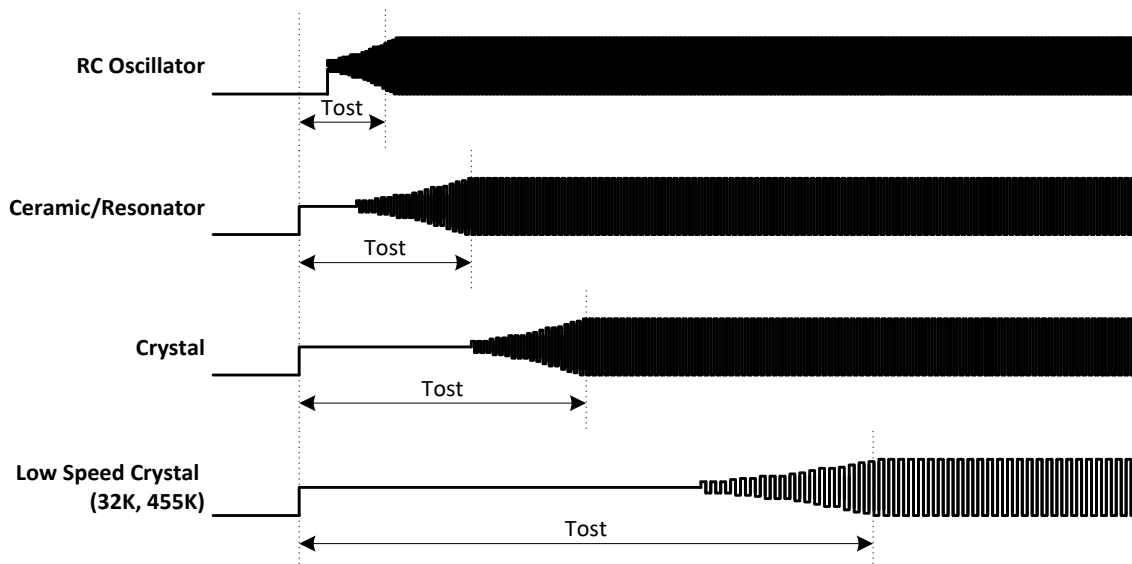


● IDLE Mode Wake-up Timing



● **Oscillator Start-up Time**

The start-up time is depended on oscillator's material, factory and architecture. Normally, the low-speed oscillator's start-up time is lower than high-speed oscillator.



8.6 System Clock and Power Management Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	-	-	-	-	-	GF0	STOP	IDLE
P1W	-	P16W	P15W	P14W	P13W	P12W	P11W	P10W
SYSMOD	-	-	-	-	-	-	-	STWK

PCON Register (0x87)

Bit	Field	Type	Initial	Description
7..3	Reserved	R	0	
2	GF0	R/W	0	General Purpose Flag
1	STOP	R/W	0	1: Microcontroller switch to STOP mode
0	IDLE	R/W	0	1: Microcontroller switch to IDLE mode

P1W Register (0xC3)

Bit	Field	Type	Initial	Description
7..0	P1nW	R/W	0	0: Disable P1.n wakeup functionality 1: Enable P1.n wakeup functionality

SYSMOD Register (0xF4)

Bit	Field	Type	Initial	Description
7..1	Reserved	R	0x00	
0	STWK	R/W	0	0: Both Fhosc and Fosc are suspended in STOP mode. 1: Fosc keep running in STOP mode*.

* Before entering STOP mode, the STWK bit setting must be earlier than the STOP bit.

9 System Operating Mode

The chip builds in three operating mode for difference clock rate and power saving reason. These modes control oscillators, op-code operation and analog peripheral devices' operation.

- Normal mode: System high-speed operating mode
- IDLE mode: System idle mode (Green mode)
- STOP mode: System power saving mode (Sleep mode)

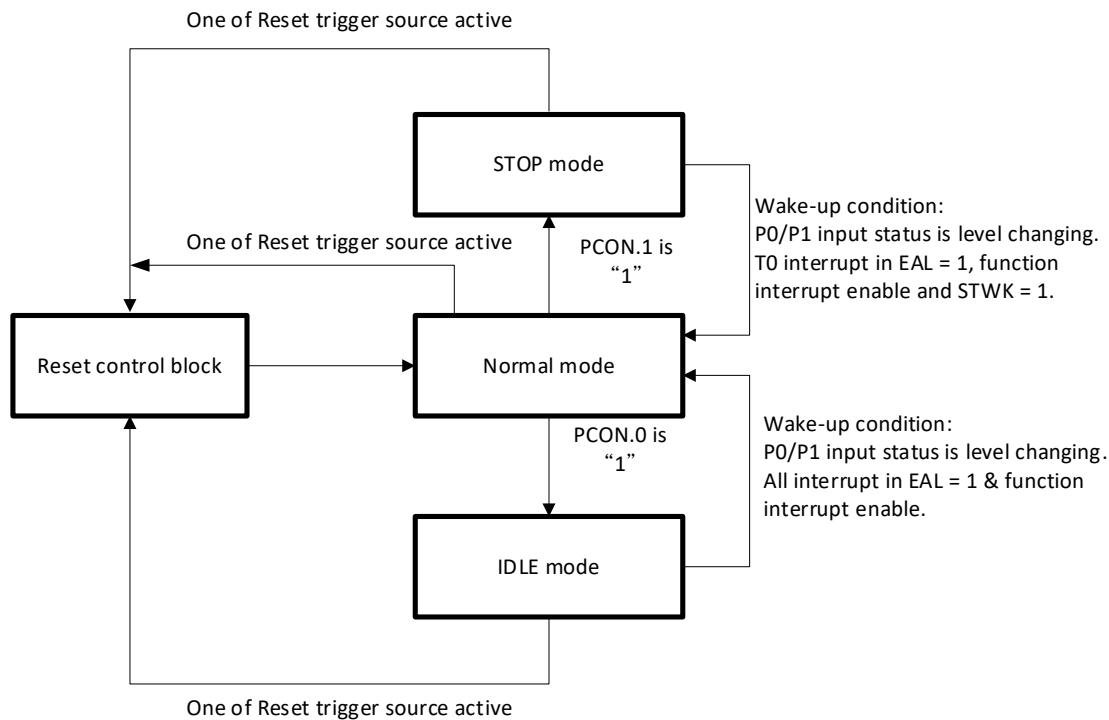


Table 9-1 The operating mode clock control

Operating Mode	Normal Mode	IDLE Mode	STOP Mode
IHRC	IHRC: Running	IHRC: Running	Stop
ILRC	Running	Running	STWK=1: Running Watchdog always: Running
CPU instruction	Executing	Stop	Stop
Timer 0 (Timer)	Active by TOEN	Active by TOEN	Active as Timer 0 clock source is Fosc (ILRC 32kHz) & STWK=1
Timer 1/2 (Timer)	Active as enable	Active as enable	Inactive
PWM0	Active as enable	Active as enable	Inactive
UART0/1	Active as enable	Active as enable	Inactive
I2C	Active as enable	Active as enable	Inactive
ADC	Active as enable	Active as enable	Inactive
CMP	Active as enable	Active as enable	Inactive
OP	Active as enable	Active as enable	Inactive
Watchdog timer	By Watchdog Code option	By Watchdog Code option	By Watchdog Code option
Internal interrupt	All active	All active	T0 interrupt is active when STWK = 1 and clock source is Fosc (ILRC 32kHz). Other inactive.
External interrupt	All active	All active	All inactive
Wakeup source	-	P0, P1, Reset, All interrupt in EAL = 1 & function interrupt enable	P0, P1, Reset, T0 enable & clock source is Fosc (ILRC 32kHz) & STWK=1 & function interrupt enable.

- IHRC: Internal high-speed oscillator RC type.
- ILRC: Internal low-speed oscillator RC type.

9.1 Normal Mode

The Normal Mode is system high clock operating mode. The system clock source is from high speed oscillator. The program is executed. After power on and any reset trigger released, the system inserts into normal mode to execute program. When the system is wake-up from STOP/IDLE mode, the system also inserts into normal mode. In normal mode, the high speed oscillator is active, and the power consumption is largest of all operating modes.

- The program is executed, and full functions are controllable.
- The system rate is high speed.
- The high speed oscillator and internal low speed RC type oscillator are active.
- Normal mode can be switched to other operating modes through PCON register.
- STOP/IDLE mode is wake-up to normal mode.

9.2 STOP Mode

The STOP mode is the system ideal status. No program execution and oscillator operation. Only internal regulator is active to keep all control gates status, register status and SRAM contents. The STOP mode is waked up by P0/P1 hardware level change trigger. P0 wake-up function is always enables. The STOP mode is wake-up to normal mode. Inserting STOP mode is controlled by stop bit of PCON register. When stop = 1, the system inserts into STOP Mode. After system wake-up from STOP mode, the stop bit is disabled (zero status) automatically.

- The program stops executing, and full functions are disabled.
- All oscillators including internal high speed oscillator and internal low speed oscillator stop.
- Only internal regulator is active to keep all control gates status, register status and SRAM contents.
- The system inserts into normal mode after wake-up from STOP mode.
- The STOP mode wake-up source is P0/P1 level change trigger, T0 enable & clock source is Fosc (ILRC 32kHz) & STWK=1 & function interrupt enable.

9.3 IDLE Mode

The IDLE mode is another system ideal status not like STOP mode. In STOP mode, all functions and hardware devices are disabled. But in IDLE mode, the system clock source keeps running, so the power consumption of IDLE mode is larger than STOP mode. In IDLE mode, the program isn't executed, but the timer with wake-up function is active as enabled, and the timer clock source is the non-stop system clock. The IDLE mode has 2 wake-up sources. One is the P0/P1 level change trigger wake-up. The other one is any interrupt in EAL = 1 & function interrupt enable. That's mean users can setup any function with interrupt enable, and the system is waked up until the interrupt issue. Inserting IDLE mode is controlled by idle bit of PCON register. When idle = 1, the system inserts into IDLE mode. After system wake-up from IDLE mode, the idle bit is disabled (zero status) automatically.

- The program stops executing, and full functions are disabled.
- The oscillator to be the system clock source keeps running, and the other oscillators operation is depend on system operation mode configuration.
- If inserting IDLE mode from normal mode, the system insets to normal mode after wake-up.
- The IDLE mode wake-up sources are P0/P1 level change trigger.
- If the function clock source is system clock, the functions are workable as enabled and under IDLE mode, e.g. Timer, PWM, event counter...
- All interrupt in EAL = 1 & function interrupt enable can wake-up in IDLE mode.

9.4 Wake up

Under STOP mode (sleep mode) or idle mode, program doesn't execute. The wakeup trigger can wake the system up to normal mode. The wakeup trigger sources are external trigger (P0/P1 level change) and internal trigger (any interrupt in EAL = 1 & function interrupt enable). The wakeup function builds in interrupt operation issued request flag and trigger system executing interrupt service routine as system wakeup occurrence.

When the system is in STOP mode the high clock oscillator stops. When waked up from STOP mode, MCU waits for 4 internal low-speed oscillator clocks as the wakeup time to stable the oscillator circuit. After the wakeup time, the system goes into the normal mode.

Parameter	Symbol	Description	Typical
Oscillator warm-up time	Tosp	Oscillator warm-up time of power down mode wake-up condition. 4*F _{ILRC}RC type oscillator, e.g. internal high-speed RC type oscillator.	250us @ F _{hosc} = 32MHz
		Oscillator warm-up time of idle down mode wake-up condition. 3*F _{ILRC}RC type oscillator, e.g. internal high-speed RC type oscillator.	188us @ F _{ILRC} = 16kHz & F _{IHRC} = 32MHz

When the system is in STOP mode the high clock oscillator stops. When waked up from STOP mode, MCU waits for 4 internal low-speed oscillator clocks as the wakeup time to stable the oscillator circuit. When waked up from IDLE mode, MCU waits for 3 internal low-speed oscillator clocks. After the wakeup time, the system goes into the normal mode.

The value of the internal high clock oscillator RC type wakeup time is as the following.

$$\text{The Wakeup time} = 1/F_{ILRC} * 4 + \text{high clock start-up time}$$

Example: In STOP mode (sleep mode), the system is waked up. After the wakeup time, the system goes into normal mode. The wakeup time is as the following.

$$\text{The wakeup time} = 1/F_{ILRC} * 4 = 250 \text{ us} + \text{high clock start-up time (F}_{hosc} = 32\text{MHz)}$$

* ***Note: The high clock start-up time is depended on the VDD and oscillator type of high clock.***

Under STOP mode and green mode, the I/O ports with wakeup function are able to wake the system up to normal mode. The wake-up trigger edge is level changing in rising edge or falling edge. The Port 0 and Port 1 have wakeup function. Port 0 wakeup functions always enables, but the Port 1 is controlled by the P1W register.

P1W Register (0xC3)

Bit	Field	Type	Initial	Description
7..0	P1nW	R/W	0	0: Disable P1.n wakeup functionality 1: Enable P1.n wakeup functionality

10 Interrupt

The MCU provides 17 interrupt sources (1 external and 16 interrupt) with 4 priority levels. Each interrupt source includes one or more interrupt request flag(s). When interrupt event occurs, the associated interrupt flag is set to logic 1. If both interrupt enable bit and global interrupt (EAL=1) are enabled, the interrupt request is generated and interrupt service routine (ISR) will be started. Some interrupt request flags must be cleared by software. However, most interrupt request flags can be cleared by hardware automatically. In the end, ISR is finished after complete the RETI instruction. The summary of interrupt source, interrupt vector, priority order and control bit are shown as the table below.

Table 10-1 The interrupt list

Interrupt	Enable Interrupt	Request (IRQ)	IRQ Clearance	Priority / Vector
System Reset	-	-	-	0 / 0x0000
INT0	EX0	IE0	Automatically	1 / 0x0003
TIMER0	ET0	TF0	Automatically	2 / 0x000B
TIMER1	EX1	TF1	Automatically	3 / 0x001B
TIMER2	ET2	TF2	Automatically	4 / 0x002B
I2C	EI2C	SI	By firmware	5 / 0x0043
UART0 RX	EU0RX	RI0	By firmware	6 / 0x0053
UART0 TX	EU0TX	TI0	By firmware	7 / 0x005B
UART1 RX	EU1RX	RI1	By firmware	8 / 0x0063
UART1 TX	EU1TX	TI1	By firmware	9 / 0x006B
PWM0	EPW0	PW0F	Automatically	10 / 0x0083
ADC	EADC	ADCF	Automatically	11 / 0x00AB
LVD	ELVD	LVDF	Automatically	12 / 0x00BB
CMP0	ECMP0	CM0F	Automatically	13 / 0x00C3
CMP1	ECMP1	CM1F	Automatically	14 / 0x00CB
CMP2	ECMP2	CM2F	Automatically	15 / 0x00D3
CMP3	ECMP3	CM3F	Automatically	16 / 0x00DB
CMP4	ECMP4	CM4F	Automatically	17 / 0x00E3

10.1 Interrupt Operation

Interrupt operation is controlled by interrupt request flag and interrupt enable bits. Interrupt request flag is interrupt source event indicator, no matter what interrupt function status (enable or disable). Both interrupt enable bit and global interrupt (EAL=1) are enabled, the system executes interrupt operation when each of interrupt request flags is active. The program counter points to interrupt vector (0x03 – 0xEB) and execute ISR.

10.2 Interrupt Priority

Each interrupt source has its specific default priority order. If two interrupts occur simultaneously, the higher priority ISR will be service first. The lower priority ISR will be serviced after the higher priority ISR completes. The next ISR will be service after the previous ISR complete, no matter the priority order.

For special priority needs, 4-level priority levels (Level 0 – Level 3) are used. All interrupt sources are classified into 6 priority groups (Group0 – Group5). Each group can be set one specific priority level. Priority level is selected by IP0/IP1 registers. Level 3 is the highest priority and Level 0 is the lowest. The interrupt sources inside the same group will share the same priority level. With the same priority level, the priority rule follows default priority.

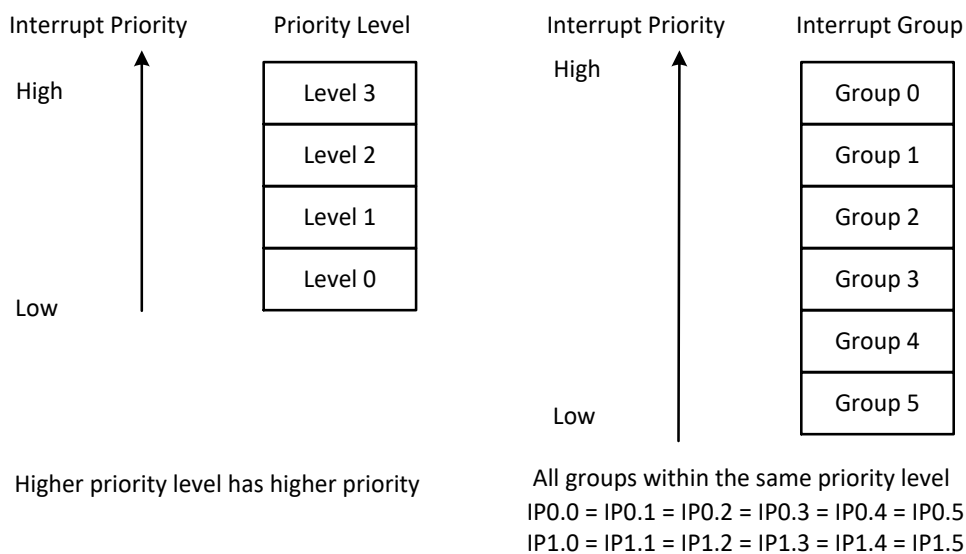
Priority Level	IP1.x	IP0.x
Level 0	0	0
Level 1	0	1
Level 2	1	0
Level 3	1	1

The ISR with the higher priority level can be serviced first; even can break the on-going ISR with the lower priority level. The ISR with the lower priority level will be pending until the ISR with the higher priority level completes.

Group	Interrupt Source				
Group 0	INT0	-	PWM0	CMP0	I2C
Group 1	TIMER0	-	-	CMP1	-
Group 2	-	-	-	CMP2	UART0 RX
Group 3	TIMER1	-	-	CMP3	UART0 TX
Group 4	-	-	-	CMP4	UART1 RX
Group 5	TIMER2	LVD	ADC	-	UART1 TX

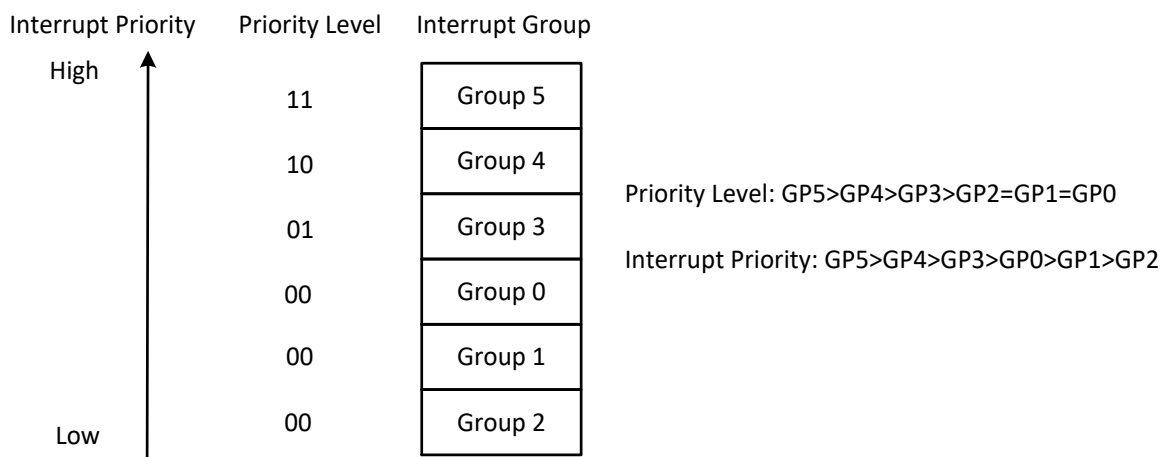
When more than one interrupt request occurs, the highest priority request must be executed first. Choose the highest priority request according natural priority and priority level. The steps are as the following:

1. Choose the groups which have the highest priority level between all groups.
2. Choose the group which is the highest nature priority between the groups with the highest priority level.
3. Choose the ISR which has the highest nature priority inside the group with the highest priority.



As the example, group5 has the highest priority level and group0~group2 have the lowest priority level. It means the interrupt vector in group5 has the highest interrupt priority, the 2nd interrupt priority in group4 and the 3rd interrupt priority in group3. Group0~ group2 have the same priority level thus the nature priority rule will be followed. Therefore, interrupt priority will be group5> group4> group3> group0> group1> group2.

```
MOV    IP0, #00101000B    ; Set group0 - group5 in different priority level.
MOV    IP1, #00110000B
```



IP0, IP1 Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IP0	-	-	IP05	IP04	IP03	IP02	IP01	IP00
IP1	-	-	IP15	IP14	IP13	IP12	IP11	IP10

IP0 Register (0xA9)

Bit	Field	Type	Initial	Description
5..0	IP0[5:0]	R/W	0	Interrupt priority. Each bit together with corresponding bit from IP1 register specifies the priority level of the respective interrupt priority group.
Else	Reserved	R	0	

IP1 Register (0xB9)

Bit	Field	Type	Initial	Description
5..0	IP1[5:0]	R/W	0	Interrupt priority. Each bit together with corresponding bit from IP0 register specifies the priority level of the respective interrupt priority group.
Else	Reserved	R	0	

10.3 Interrupt Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
IEN1	-	-	EU1RX	EU1TX	EUORX	EUOTX	-	EI2C
IEN2	ECMP4	ECMP3	ECMP2	ECMP1	ECMP0	ET2	EADC	ELVD
IEN3	-	-	-	-	-	EPW0	-	-

IRCON2	CM4F	CM3F	CM2F	CM1F	CM0F	TF2	ADCF	LVDF
IRCON3	-	-	-	-	-	PW0F	-	-
TCON	TF1	-	TF0	-	-	-	IE0	-
S0CON	S0M0	S0M1	S0M20	REN0	TB80	RB80	TI0	RI0
S1CON	S1M0	S1M1	S1M20	REN1	TB81	RB81	TI1	RI1
I2CCON	CR2	ENS1	STA	STO	SI	AA	CR1	CRO

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Enable all interrupt control bit. 0: Disable all interrupt function. 1: Enable all interrupt function.
3	ET1	R/W	0	T1 timer interrupt control bit. 0: Disable T1 interrupt function. 1: Enable T1 interrupt function.
1	ET0	R/W	0	T0 timer interrupt control bit. 0: Disable T0 interrupt function. 1: Enable T0 interrupt function
0	EX0	R/W	0	External P0.0 interrupt (INT0) control bit. 0: Disable INT0 interrupt function. 1: Enable INT0 interrupt function.
Else	Reserved	R	0	

IEN1 Register (0xB8)

Bit	Field	Type	Initial	Description
5	EU1RX	R/W	0	UART1 RX interrupt control bit. 0: Disable UART RX interrupt function. 1: Enable UART RX interrupt function.
4	EU1TX	R/W	0	UART1 TX interrupt control bit. 0: Disable UART TX interrupt function. 1: Enable UART TX interrupt function.
3	EU0RX	R/W	0	UART0 RX interrupt control bit. 0: Disable UART RX interrupt function. 1: Enable UART RX interrupt function.
2	EU0TX	R/W	0	UART0 TX interrupt control bit. 0: Disable UART TX interrupt function. 1: Enable UART TX interrupt function.
0	EI2C	R/W	0	I2C interrupt control bit. 0: Disable I2C interrupt function. 1: Enable I2C interrupt function.
Else	Reserved	R	0	

IEN2 Register (0xC8)

Bit	Field	Type	Initial	Description
7	ECMP4	R/W	0	CMP4 interrupt control bit. 0: Disable CMP4 interrupt function. 1: Enable CMP4 interrupt function.
6	ECMP3	R/W	0	CMP3 interrupt control bit. 0: Disable CMP3 interrupt function. 1: Enable CMP3 interrupt function.
5	ECMP2	R/W	0	CMP2 interrupt control bit. 0: Disable CMP2 interrupt function. 1: Enable CMP2 interrupt function.
4	ECMP1	R/W	0	CMP1 interrupt control bit. 0: Disable CMP1 interrupt function. 1: Enable CMP1 interrupt function.
3	ECMP0	R/W	0	CMP0 interrupt control bit. 0: Disable CMP0 interrupt function. 1: Enable CMP0 interrupt function.
2	ET2	R/W	0	Timer 2 interrupt control bit. 0: Disable T2 interrupt function. 1: Enable T2 interrupt function.
1	EADC	R/W	0	ADC interrupt control bit. 0: Disable ADC interrupt function. 1: Enable ADC interrupt function.
0	ELVD	R/W	0	LVD interrupt control bit. 0 = Disable LVD interrupt function. 1 = Enable LVD interrupt function.

IEN3 Register (0x98)

Bit	Field	Type	Initial	Description
2	EPW0	R/W	0	PWM0 interrupt control bit. 0 = Disable PWM0 interrupt function. 1 = Enable PWM0 interrupt function.
Else	Reserved	R	0	

IRCON2 Register (0xA7)

Bit	Field	Type	Initial	Description
7	CM4F	R/W	0	CMP4 interrupt request flag. 0: None CMP4 interrupt request

				1: CMP4 interrupt request.
6	CM3F	R/W	0	CMP3 interrupt request flag. 0: None CMP3 interrupt request 1: CMP3 interrupt request.
5	CM2F	R/W	0	CMP2 interrupt request flag. 0: None CMP2 interrupt request 1: CMP2 interrupt request.
4	CM1F	R/W	0	CMP1 interrupt request flag. 0: None CMP1 interrupt request 1: CMP1 interrupt request.
3	CM0F	R/W	0	CMP0 interrupt request flag. 0: None CMP0 interrupt request 1: CMP0 interrupt request.
2	TF2	R/W	0	Timer 2 interrupt request flag. 0: None T2 interrupt request 1: T2 interrupt request.
1	ADCF	R/W	0	ADC interrupt request flag. 0: None ADC interrupt request. 1: ADC interrupt request.
0	LVDF	R/W	0	LVD interrupt request flag. 0: None LVD interrupt request. 1: LVD interrupt request.

IRCON3 Register (0xC9)

Bit	Field	Type	Initial	Description
2	PW0F	R/W	0	PWM0 interrupt request flag. 0: None PWM0 interrupt request 1: PWM0 interrupt request.
Else	Reserved	R/W	0	

TCON Register (0x88)

Bit	Field	Type	Initial	Description
7	TF1	R/W	0	Timer 1 interrupt request flag. 0: None T1 interrupt request 1: T1 interrupt request.
5	TF0	R/W	0	Timer 0 interrupt request flag. 0: None T0 interrupt request

				1: T0 interrupt request.
1	IE0	R/W	0	External P0.0 interrupt (INT0) request flag 0: None INT0 interrupt request. 1: INT0 interrupt request.
Else	Reserved	R/W	0	

SOCON Register (0x97)

Bit	Field	Type	Initial	Description
1	TIO	R/W	0	UART0 transmit interrupt request flag. It indicates completion of a serial transmission at UART0. It is set by hardware at the end of bit 8 in mode 0 or at the beginning of a stop bit in other modes. It must be cleared by software. 0: None UART0 transmit interrupt request. 1: UART0 transmit interrupt request.
0	RI0	R/W	0	UART0 receive interrupt request flag. It is set by hardware after completion of a serial reception at UART0. It is set by hardware at the end of bit 8 in mode 0 or in the middle of a stop bit in other modes. It must be cleared by software. 0: None UART0 receive interrupt request. 1: UART0 receive interrupt request.
Else				Refer to other chapter(s)

S1CON Register (0x9C)

Bit	Field	Type	Initial	Description
1	TI1	R/W	0	UART1 transmit interrupt request flag. It indicates completion of a serial transmission at UART1. It is set by hardware at the end of bit 8 in mode 0 or at the beginning of a stop bit in other modes. It must be cleared by software. 0: None UART1 transmit interrupt request. 1: UART1 transmit interrupt request.
0	RI1	R/W	0	UART1 receive interrupt request flag. It is set by hardware after completion of a serial reception at UART1. It is set by hardware at the end of bit 8 in mode 0 or in the middle of a stop bit in other modes. It must be cleared by software.

	0: None UART1 receive interrupt request.
	1: UART1 receive interrupt request.
Else	Refer to other chapter(s)

I2CCON Register (0xDC)

Bit	Field	Type	Initial	Description
3	SI	R/W	0	Serial interrupt flag The SI is set by hardware when one of 25 out of 26 possible I2C states is entered. The only state that does not set the SI is state F8h, which indicates that no relevant state information is available. The SI flag must be cleared by software. In order to clear the SI bit, '0' must be written to this bit. Writing a '1' to SI bit does not change value of the SI.
Else				Refer to other chapter(s)

11 GPIO

The microcontroller has up to 17 bidirectional general purpose I/O pin (GPIO). Unlike the original 8051 only has open-drain output, SN8F5602 builds in push-pull output structure to improve its driving performance.

11.1 Input and Output Control

The input and output direction control is configurable through P0M to P7M registers. These bits specify each pin that is either input mode or output mode.

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0M	P07M	P06M	P05M	P04M	P03M	P02M	-	P00M
P1M	-	P16M	P15M	P14M	P13M	P12M	P11M	P10M
P4M	-	-	-	-	-	P42M	P41M	P40M
ODCON	-	-	-	-	-	-	URT1OD	URT0OD

P0M: 0xF9

Bit	Field	Type	Initial	Description
7	P07M	R/W	0	Mode selection of P0.7 0: Input mode 1: Output mode
6	P06M	R/W	0	Mode selection of P0.6 0: Input mode 1: Output mode
5	P05M	R/W	0	Mode selection of P0.5 0: Input mode 1: Output mode
4	P04M	R/W	0	Mode selection of P0.4 0: Input mode 1: Output mode
3	P03M	R/W	0	Mode selection of P0.3 0: Input mode 1: Output mode
2	P02M	R/W	0	Mode selection of P0.2 0: Input mode 1: Output mode
1	Reserved	R	0	P01 Output mode only.
0	P00M	R/W	0	Mode selection of P0.0

0: Input mode
1: Output mode

P1M: 0xFA

Bit	Field	Type	Initial	Description
7	Reserved	R	0	
6	P16M	R/W	0	Mode selection of P1.6 0: Input mode 1: Output mode
5	P15M	R/W	0	Mode selection of P1.5 0: Input mode 1: Output mode
4	P14M	R/W	0	Mode selection of P1.4 0: Input mode 1: Output mode
3	P13M	R/W	0	Mode selection of P1.3 0: Input mode 1: Output mode
2	P12M	R/W	0	Mode selection of P1.2 0: Input mode 1: Output mode
1	P11M	R/W	0	Mode selection of P1.1 0: Input mode 1: Output mode
0	P10M	R/W	0	Mode selection of P1.0 0: Input mode 1: Output mode

P4M: 0xFD

Bit	Field	Type	Initial	Description
7..3	Reserved	R	00	
2	P42M	R/W	0	Mode selection of P4.2 0: Input mode 1: Output mode
1	P41M	R/W	0	Mode selection of P4.1 0: Input mode 1: Output mode
0	P40M	R/W	0	Mode selection of P4.0

0: Input mode
1: Output mode

*** Note:** The pins which are not pin-out shall be set correctly to decrease power consumption in low-power modes. Strongly recommended to set these pins as input pull-up.

ODCON Register (0xF018)

Bit	Field	Type	Initial	Description
1	URT1OD	R/W	0	UART1 open-drain control bit 0: Disable 1: Enable
0	URT0OD	R/W	0	UART0 open-drain control bit 0: Disable 1: Enable
Else	Reserved	R	0	

11.2 Input Data and Output Data

By a read operation from any registers of P0 to P4, the current pin's logic level would be fetch to represent its external status. This operation remains functional even the pin is shared with other function like UART and I2C which can monitor the bus condition in some case.

A write P0 to P4 register value would be latched immediately, yet the value would be outputted until the mapped P0M – P4M is set to output mode. If the pin is currently in output mode, any value set to P0 to P4 register would be presented on the pin immediately.

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0	P07	P06	P05	P04	P03	P02	P01	P00
P1	-	P16	P15	P14	P13	P12	P11	P10
P4	-	-	-	-	-	P42	P41	P40

P0: 0x80

Bit	Field	Type	Initial	Description
7	P07	R/W	1	Read: P0.7 pin's logic level Write 1/0: Output logic high or low (applied if P07M = 1)
6	P06	R/W	1	Read: P0.6 pin's logic level Write 1/0: Output logic high or low (applied if P06M = 1)

5	P05	R/W	1	Read: P0.5 pin's logic level Write 1/0: Output logic high or low (applied if P05M = 1)
4	P04	R/W	1	Read: P0.4 pin's logic level Write 1/0: Output logic high or low (applied if P04M = 1)
3	P03	R/W	1	Read: P0.3 pin's logic level Write 1/0: Output logic high or low (applied if P03M = 1)
2	P02	R/W	1	Read: P0.2 pin's logic level Write 1/0: Output logic high or low (applied if P02M = 1)
1	P01	R/W	1	Read: P0.1 pin's logic level Write 1/0: Output logic high (external pull-up) or low.
0	P00	R/W	1	Read: P0.0 pin's logic level Write 1/0: Output logic high or low (applied if P00M = 1)

P1: 0x90

Bit	Field	Type	Initial	Description
7	Reserved	R	0	
6	P16	R/W	1	Read: P1.6 pin's logic level Write 1/0: Output logic high or low (applied if P16M = 1)
5	P15	R/W	1	Read: P1.5 pin's logic level Write 1/0: Output logic high or low (applied if P15M = 1)
4	P14	R/W	1	Read: P1.4 pin's logic level Write 1/0: Output logic high or low (applied if P14M = 1)
3	P13	R/W	1	Read: P1.3 pin's logic level Write 1/0: Output logic high or low (applied if P13M = 1)
2	P12	R/W	1	Read: P1.2 pin's logic level Write 1/0: Output logic high or low (applied if P12M = 1)
1	P11	R/W	1	Read: P1.1 pin's logic level Write 1/0: Output logic high or low (applied if P11M = 1)
0	P10	R/W	1	Read: P1.0 pin's logic level Write 1/0: Output logic high or low (applied if P10M = 1)

P4: 0xC0

Bit	Field	Type	Initial	Description
7..3	Reserved	R	0	
2	P42	R/W	1	Read: P4.2 pin's logic level Write 1/0: Output logic high or low (applied if P42M = 1)
1	P41	R/W	1	Read: P4.1 pin's logic level Write 1/0: Output logic high or low (applied if P41M = 1)

0	P40	R/W	1	Read: P4.0 pin's logic level Write 1/0: Output logic high or low (applied if P40M = 1)
---	-----	-----	---	---

11.3 On-chip Pull-up Resistors

The P0UR to P4UR registers are mapped to each pins' internal 100k ohm (in typical value) pull-up resistor.

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0UR	P07UR	P06UR	P05UR	P04UR	P03UR	P02UR	-	P00UR
P1UR	-	P16UR	P15UR	P14UR	P13UR	P12UR	P11UR	P10UR
P4UR	-	-	-	-	-	P42UR	P41UR	P40UR

P0UR: 0xF000

Bit	Field	Type	Initial	Description
7	P07UR	R/W	0	On-chip pull-up resistor control of P0.7 0: Disable* 1: Enable
6	P06UR	R/W	0	On-chip pull-up resistor control of P0.6 0: Disable* 1: Enable
5	P05UR	R/W	0	On-chip pull-up resistor control of P0.5 0: Disable* 1: Enable
4	P04UR	R/W	0	On-chip pull-up resistor control of P0.4 0: Disable* 1: Enable
3	P03UR	R/W	0	On-chip pull-up resistor control of P0.3 0: Disable* 1: Enable
2	P02UR	R/W	0	On-chip pull-up resistor control of P0.2 0: Disable* 1: Enable
1	Reserved	R	0	
0	P00UR	R/W	0	On-chip pull-up resistor control of P0.0 0: Disable* 1: Enable

* Recommended disable pull-up resistor if the pin is output mode or analog function

P1UR: 0xF001

Bit	Field	Type	Initial	Description
7	Reserved	R	0	
6	P16UR	R/W	0	On-chip pull-up resister control of P1.6 0: Disable* 1: Enable
5	P15UR	R/W	0	On-chip pull-up resister control of P1.5 0: Disable* 1: Enable
4	P14UR	R/W	0	On-chip pull-up resister control of P1.4 0: Disable* 1: Enable
3	P13UR	R/W	0	On-chip pull-up resister control of P1.3 0: Disable* 1: Enable
2	P12UR	R/W	0	On-chip pull-up resister control of P1.2 0: Disable* 1: Enable
1	P11UR	R/W	0	On-chip pull-up resister control of P1.1 0: Disable* 1: Enable
0	P10UR	R/W	0	On-chip pull-up resister control of P1.0 0: Disable* 1: Enable

* Recommended disable pull-up resister if the pin is output mode or analog function

P4UR: 0xF004

Bit	Field	Type	Initial	Description
7..3	Reserved	R	0	
2	P42UR	R/W	0	On-chip pull-up resister control of P4.2 0: Disable* 1: Enable
1	P41UR	R/W	0	On-chip pull-up resister control of P4.1 0: Disable* 1: Enable
0	P40UR	R/W	0	On-chip pull-up resister control of P4.0 0: Disable* 1: Enable

* Recommended disable pull-up resistor if the pin is output mode or analog function

11.4 Pin Shared with Analog Function

The microcontroller builds in analog functions, such as AD and LCD. The Schmitt trigger of input channel is strongly recommended to switch off if the pin's shared analog function is enabled.

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0CON	P0CON7	P0CON6	P0CON5	P0CON4	P0CON3	P0CON2	-	P0CON0
P1CON	-	P1CON6	P1CON5	P1CON4	P1CON3	P1CON2	P1CON1	P1CON0
P4CON	-	-	-	-	-	P4CON2	P4CON1	P4CON0

P0CON: 0xF010

Bit	Field	Type	Initial	Description
7	P0CON7	R/W	0	P0.7 configuration control bit*. 0: P0.7 can be analog input pin or digital GPIO pin. 1: P0.7 is pure analog input pin and can't be a digital GPIO pin.
6	P0CON6	R/W	0	P0.6 configuration control bit*. 0: P0.6 can be analog input pin or digital GPIO pin. 1: P0.6 is pure analog input pin and can't be a digital GPIO pin.
5	P0CON5	R/W	0	P0.5 configuration control bit*. 0: P0.5 can be analog input pin or digital GPIO pin. 1: P0.5 is pure analog input pin and can't be a digital GPIO pin.
4	P0CON4	R/W	0	P0.4 configuration control bit*. 0: P0.4 can be analog input pin or digital GPIO pin. 1: P0.4 is pure analog input pin and can't be a digital GPIO pin.
3	P0CON3	R/W	0	P0.3 configuration control bit*. 0: P0.3 can be analog input pin or digital GPIO pin. 1: P0.3 is pure analog input pin and can't be a digital GPIO pin.
2	P0CON2	R/W	0	P0.2 configuration control bit*. 0: P0.2 can be analog input pin or digital GPIO pin. 1: P0.2 is pure analog input pin and can't be a digital GPIO pin.

1	Reserved	R	0	
0	POCON0	R/W	0	P0.0 configuration control bit*. 0: P0.0 can be analog input pin or digital GPIO pin. 1: P0.0 is pure analog input pin and can't be a digital GPIO pin.

* P0CON [7:0] will configure related Port0 pin as pure analog input pin to avoid current leakage.

P1CON: 0xF011

Bit	Field	Type	Initial	Description
7	Reserved	R/W	0	
6	P1CON6	R/W	0	P1.6 configuration control bit*. 0: P1.6 can be analog input pin or digital GPIO pin. 1: P1.6 is pure analog input pin and can't be a digital GPIO pin.
5	P1CON5	R/W	0	P1.5 configuration control bit*. 0: P1.5 can be analog input pin or digital GPIO pin. 1: P1.5 is pure analog input pin and can't be a digital GPIO pin.
4	P1CON4	R/W	0	P1.4 configuration control bit*. 0: P1.4 can be analog input pin or digital GPIO pin. 1: P1.4 is pure analog input pin and can't be a digital GPIO pin.
3	P1CON3	R/W	0	P1.3 configuration control bit*. 0: P1.3 can be analog input pin or digital GPIO pin. 1: P1.3 is pure analog input pin and can't be a digital GPIO pin.
2	P1CON2	R/W	0	P1.2 configuration control bit*. 0: P1.2 can be analog input pin or digital GPIO pin. 1: P1.2 is pure analog input pin and can't be a digital GPIO pin.
1	P1CON1	R/W	0	P1.1 configuration control bit*. 0: P1.1 can be analog input pin or digital GPIO pin. 1: P1.1 is pure analog input pin and can't be a digital GPIO pin.
0	P1CON0	R/W	0	P1.0 configuration control bit*. 0: P1.0 can be analog input pin or digital GPIO pin. 1: P1.0 is pure analog input pin and can't be a digital GPIO pin.

* P1CON [7:0] will configure related Port1 pin as pure analog input pin to avoid current leakage.

P4CON: 0xF014

Bit	Field	Type	Initial	Description
7..3	Reserved	R/W	0	Reserved
2	P4CON2	R/W	0	P4.2 configuration control bit*. 0: P4.2 can be analog input pin or digital GPIO pin. 1: P4.2 is pure analog input pin and can't be a digital GPIO pin.
1	P4CON1	R/W	0	P4.1 configuration control bit*. 0: P4.1 can be analog input pin or digital GPIO pin. 1: P4.1 is pure analog input pin and can't be a digital GPIO pin.
0	P4CON0	R/W	0	P4.0 configuration control bit*. 0: P4.0 can be analog input pin or digital GPIO pin. 1: P4.0 is pure analog input pin and can't be a digital GPIO pin.

* P4CON [7:0] will configure related Port4 pin as pure analog input pin to avoid current leakage.

12 External Interrupt

INT0 is external interrupt trigger sources. Build in edge trigger configuration function and edge direction is selected by PEDGE register. When both external interrupt (EX0) and global interrupt (EAL) are enabled, the external interrupt request flag (IE0) will be set to “1” as edge trigger event occurs. The program counter will jump to the interrupt vector (ORG 0x0003) and execute interrupt service routine. Interrupt request flag will be cleared by hardware before ISR is executed.

12.1 External Interrupt Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PEDGE	-	-	-	-	-	-	EX0G1	EX0G0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
TCON	TF1	-	TF0	-	-	-	IE0	-

PEDGE Register (0x8F)

Bit	Field	Type	Initial	Description
1..0	EX0G[1:0]	R/W	10	External interrupt 0 trigger edge control register. 00: Reserved. 01: Rising edge trigger. 10: Falling edge trigger (default) 11: Both rising and falling edge trigger
Else	Reserved	R	0	Else

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Enable all interrupt control bit. 0: Disable all interrupt function. 1: Enable all interrupt function.
0	EX0	R/W	0	External P0.0 interrupt (INT0) control bit. 0: Disable INT0 interrupt function. 1: Enable INT0 interrupt function.
Else				Refer to other chapter(s)

TCON Register (0x88)

Bit	Field	Type	Initial	Description
1	IE0	R/W	0	External P0.0 interrupt (INT0) request flag

	0: None INT0 interrupt request.
	1: INT0 interrupt request.
Else	Refer to other chapter(s)

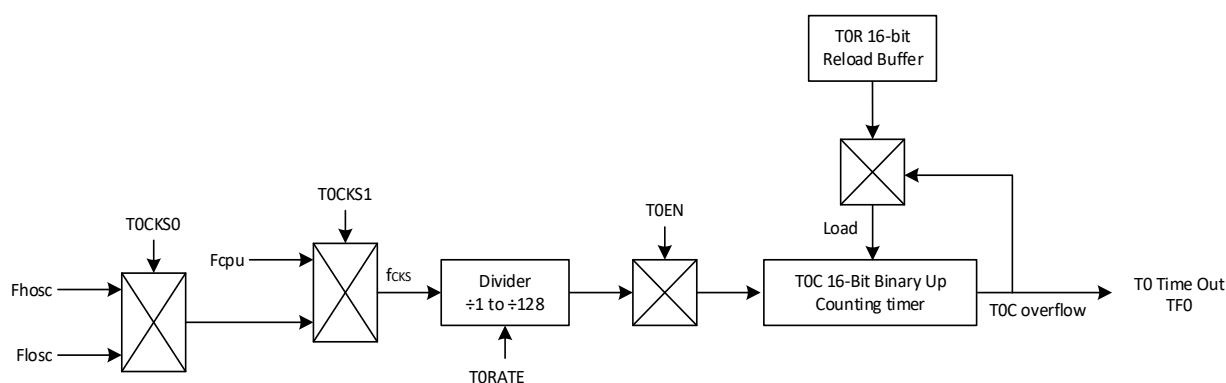
13 Timer 0

Timer 0 has 16-bit up counting timer with specified reload value support. An overflow event (TOCH/TOCL counts from 0xFFFF to 0x0000) issues its TF0 flag for firmware or interrupt controller; meanwhile, the timer duplicates TORH/TORL value to TOCH/TOCL register in the same time. As a result, the timer is actually counts from 0xFFFF to the value of TORH/TORL.

The Timer 0 build in IDLE Mode wake-up function if interrupt enable. When timer overflow occurs (counts from 0xFFFF to 0x0000), TF0 would be issued immediately which can read/write by firmware. Timer 0 interrupt function is controlled by ET0.

13.1 Timer 0 Clock Selection

Timer 0 has three clock source options: Fcpu, Fhosc, Fosc. Timer 0 selects the clock source via the T0CKS[1:0] bit and selects the frequency division by the TORATE [2: 0] bits. The Clock division range is /1 ~ /128.



13.2 Timer 0 Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T0M	TOEN	TORATE2	TORATE1	TORATE0	T0CKS1	T0CKS0	-	-
TOCH	T0C15	T0C14	T0C13	T0C12	T0C11	T0C10	T0C9	T0C8
TOCL	T0C7	T0C6	T0C5	T0C4	T0C3	T0C2	T0C1	T0C0
TORH	T0R15	T0R14	T0R13	T0R12	T0R11	T0R10	T0R9	T0R8
TORL	T0R7	T0R6	T0R5	T0R4	T0R3	T0R2	T0R1	T0R0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
TCON	TF1	-	TF0	-	-	-	IE0	-

TOM Register (0xF060)

Bit	Field	Type	Initial	Description
7	TOEN	R/W	0	Timer 0 function 0: Disable 1: Enable
6..4	TORATE[2:0]	R/W	000	Clock divider of Timer 0 f_{CKS} clock source 000: $f_{CKS} / 128$ 001: $f_{CKS} / 64$ 010: $f_{CKS} / 32$ 011: $f_{CKS} / 16$ 100: $f_{CKS} / 8$ 101: $f_{CKS} / 4$ 110: $f_{CKS} / 2$ 111: $f_{CKS} / 1$
3..2	TOCKS	R/W	0	Timer 0 f_{CKS} clock source selection. 00/01: Fcpu 10: Fhosc 11: Fosc (ILRC_32K)
1..0	Reserved	R/W	000	

TOCH/TOCL Registers (TOCH: 0xF061, TOCL: 0xF062)

Bit	Field	Type	Initial	Description
7..0	TOCH/L	R/W	0x00	16-bit Timer 0 counter buffer.

TORH/TORL Registers (TORH: 0xF063, TORL: 0xF064)

Bit	Field	Type	Initial	Description
7..0	TORH/L	R/W	0x00	16-bit Timer 0 counter reload buffer.

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
1	ETO	R/W	0	Timer 0 interrupt 0: Disable 1: Enable
Else				Refer to other chapter(s)

TCON Register (0x88)

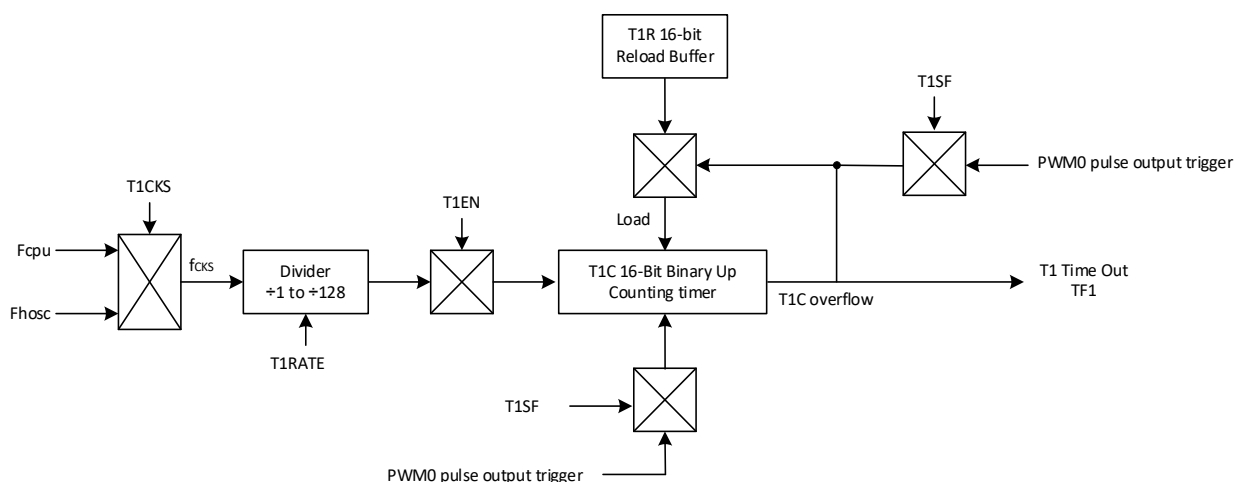
Bit	Field	Type	Initial	Description
-----	-------	------	---------	-------------

5	TF0	R/W	0	Timer 0 overflow event 0: Timer 0 does not have any overflow event 1: Timer 0 has overflowed This bit can be cleared automatically by interrupt handler, or manually by firmware
Else				Refer to other chapter(s)

14 Timer 1

Timer 1 has 16-bit up counting timer with specified reload value support. Timer 1 has two different operation modes: (1) 16-bit up counting timer mode and (2) PWM0 pulse output trigger mode.

The Timer 1 build in IDLE Mode wake-up function if interrupt enable. When timer overflow occurs (counts from 0xFFFF to 0x0000), TF1 would be issued immediately which can read/write by firmware. Timer 1 interrupt function is controlled by ET1.



14.1 Timer 1 Clock Selection

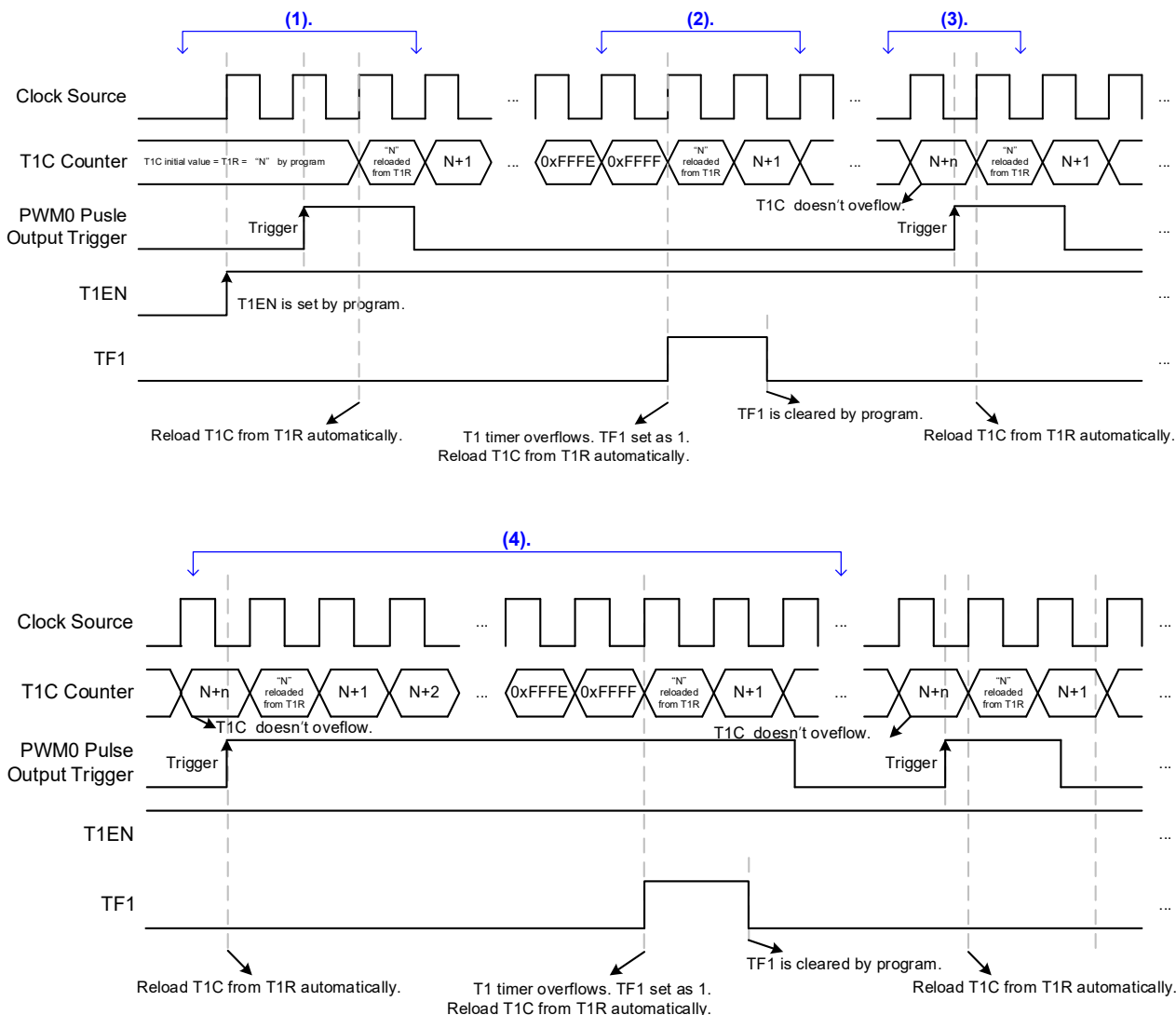
Timer 1 has two clock source options: F_{cpu} , F_{osc} . Timer 1 selects the clock source via the T1CKS bit and selects the frequency division by the T1RATE [2: 0] bits. The Clock division range is $/1 \sim /128$.

14.2 Timer 1 Timer Operation

Timer 1 has 16-bit up counting timer with specified reload value support. An overflow event (T1CH/T1CL counts from 0xFFFF to 0x0000) issues its TF1 flag for firmware or interrupt controller; meanwhile, the timer duplicates T1RH/T1RL value to T1CH/T1CL register in the same time. As a result, the timer is actually counts from 0xFFFF to the value of T1RH/T1RL.

14.3 Timer 1 PWM0 Pulse Output Trigger Mode

T1 timer builds in PWM0 trigger mode controlled by T1SF bit set as "1". When T1 timer's PWM0 trigger mode is enabled ($T1SF = 1$) and PWM0 output trigger is issued, T1 timer starts to count. If T1 timer doesn't overflow, and a new PWM0 output trigger is issued, T1CH/T1CL will be reloaded from T1RH/T1RL automatically and count again. If T1 timer overflows (from 0xFFFF to 0x0000), T1CH/T1CL will be reloaded from T1RH/T1RL automatically, continue counting, and issue a time-out signal to indicate time out event ($TF1 = 1$).



- (1). After T1EN is enabled, T1 timer doesn't count and wait the PWM0 pulse trigger signal. When PWM0 pulse start to output, the trigger signal will trigger T1 timer to count.
- (2). When T1 timer overflows, and no PWM0 pulse output trigger signal, the T1C is reloaded from T1R automatically and issues TF1 time-out flag.
- (3). If T1 timer doesn't overflow, but PWM0 pulse output trigger is issued, the T1C is reloaded from T1R automatically, but not issue TF1 time-out flag.
- (4). If T1 timer overflows, but PWM0 pulse output keeps output status, the T1C is reloaded from T1R automatically and issues TF1 time-out flag.

14.4 Timer 1 Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1M	T1EN	T1RATE2	T1RATE1	T1RATE0	T1CKS	-	-	T1SF
T1CH	T1C15	T1C14	T1C13	T1C12	T1C11	T1C10	T1C9	T1C8

T1CL	T1C7	T1C6	T1C5	T1C4	T1C3	T1C2	T1C1	T1C0
T1RH	T1R15	T1R14	T1R13	T1R12	T1R11	T1R10	T1R9	T1R8
T1RL	T1R7	T1R6	T1R5	T1R4	T1R3	T1R2	T1R1	T1R0
IEN0	EAL	-	-	-	ET1	-	ETO	EXO
TCON	TF1	-	TF0	-	-	-	IE0	-

T1M Register (0xF068)

Bit	Field	Type	Initial	Description
7	T1EN	R/W	0	Timer 1 function 0: Disable 1: Enable
6..4	T1RATE[2:0]	R/W	000	Clock divider of Timer 1 f_{CKS} clock source 000: $f_{CKS} / 128$ 001: $f_{CKS} / 64$ 010: $f_{CKS} / 32$ 011: $f_{CKS} / 16$ 100: $f_{CKS} / 8$ 101: $f_{CKS} / 4$ 110: $f_{CKS} / 2$ 111: $f_{CKS} / 1$
3	T1CKS	R/W	0	Timer 1 f_{CKS} clock source selection. 0: Fcpu 1: Fhosc
0	T1SF	R/W	0	T1 timer trigger function control bit. 0 = Disable (Normal timer mode). 1 = Enable (PWM0 pulse output trigger mode).
Else	Reserved	R/W	000	

T1CH/T1CL Registers (T1CH: 0xF069, T1CL: 0xF06A)

Bit	Field	Type	Initial	Description
7..0	T1CH/L	R/W	0x00	16-bit Timer 1 counter buffer.

T1RH/T1RL Registers (T1RH: 0xF06B, T1RL: 0xF06C)

Bit	Field	Type	Initial	Description
7..0	T1RH/L	R/W	0x00	16-bit Timer 1 counter reload buffer.

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
3	ET1	R/W	0	Timer 1 interrupt 0: Disable 1: Enable
Else				Refer to other chapter(s)

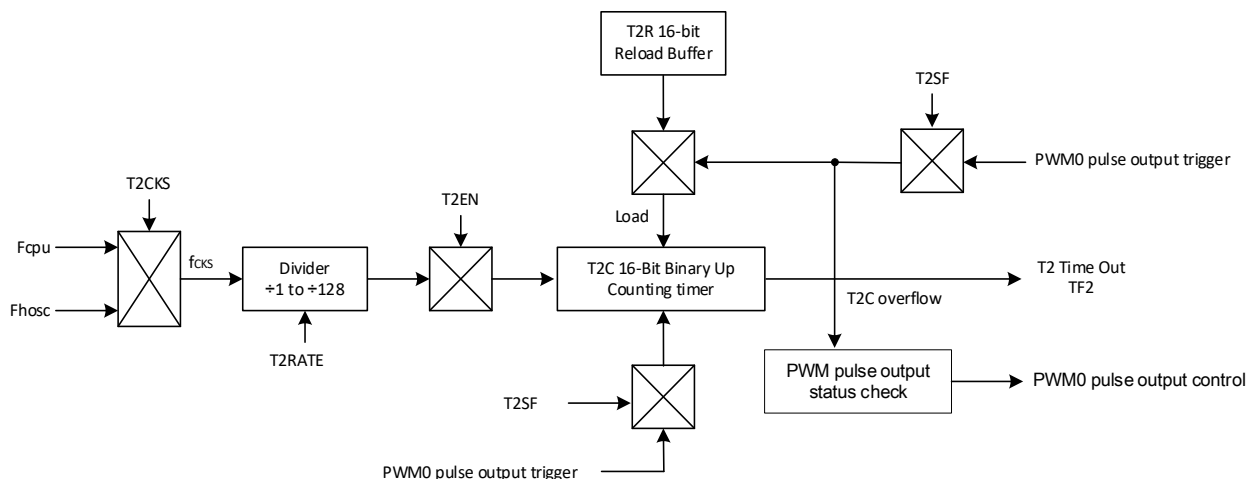
TCON Register (0x88)

Bit	Field	Type	Initial	Description
7	TF1	R/W	0	Timer 1 overflow event 0: Timer 1 does not have any overflow event 1: Timer 1 has overflowed This bit can be cleared automatically by interrupt handler, or manually by firmware
Else				Refer to other chapter(s)

15 Timer 2

Timer 2 has 16-bit up counting timer with specified reload value support. Timer 2 has two different operation modes: (1) 16-bit up counting timer mode and (2) PWM0 pulse output trigger mode.

The Timer 12 build in IDLE Mode wake-up function if interrupt enable. When timer overflow occurs (counts from 0xFFFF to 0x0000), TF2 would be issued immediately which can read/write by firmware. Timer 2 interrupt function is controlled by ET2.



15.1 Timer 2 Clock Selection

Timer 2 has two clock source options: F_{cpu} , F_{osc} . Timer 2 selects the clock source via the $T2CKS$ bit and selects the frequency division by the $T2RATE$ [2: 0] bits. The Clock division range is $/1 \sim /128$.

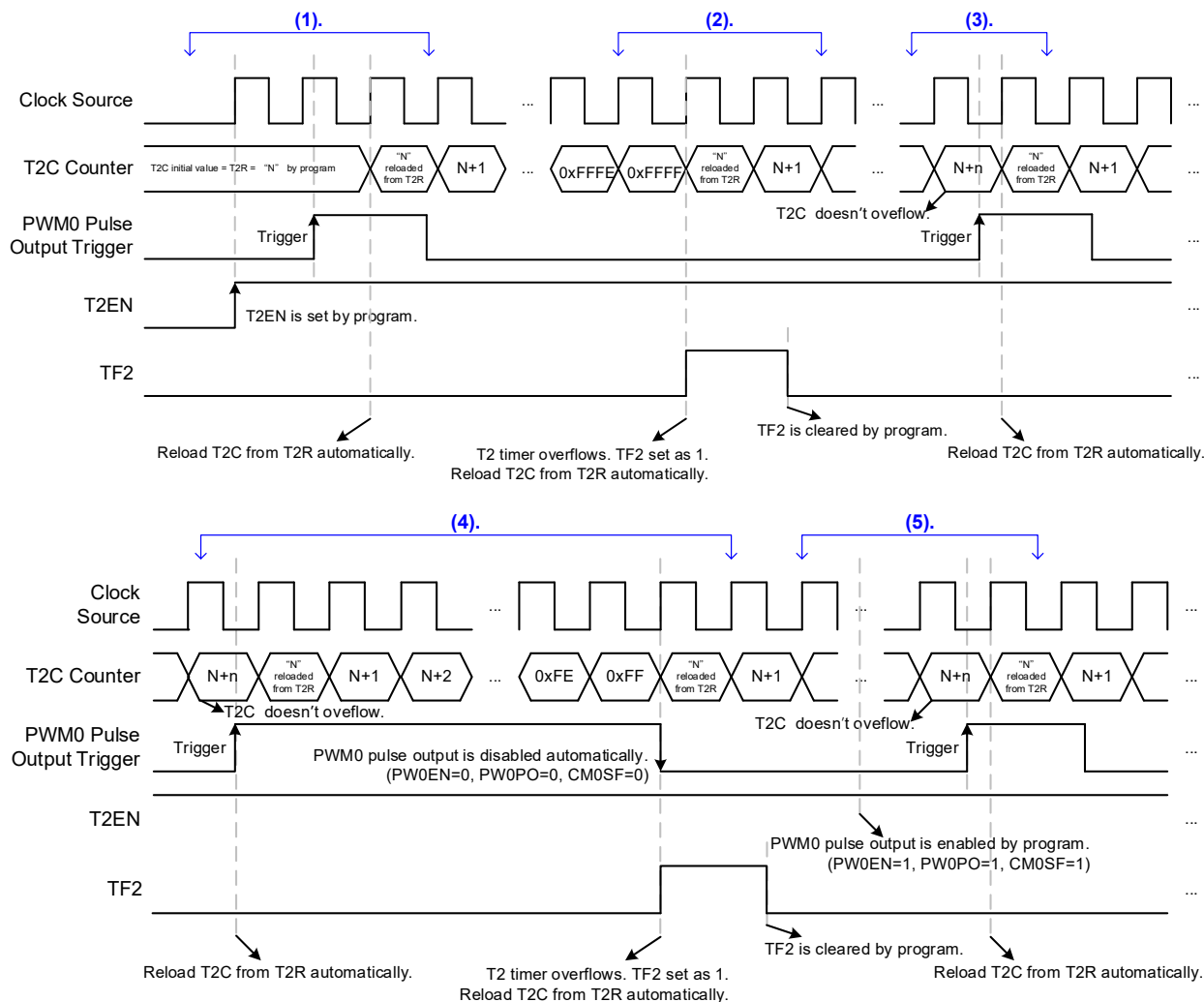
15.2 Timer 2 Timer Operation

Timer 2 has 16-bit up counting timer with specified reload value support. An overflow event ($T2CH/T2CL$ counts from 0xFFFF to 0x0000) issues its TF2 flag for firmware or interrupt controller; meanwhile, the timer duplicates $T2RH/T2RL$ value to $T2CH/T2CL$ register in the same time. As a result, the timer is actually counts from 0xFFFF to the value of $T2RH/T2RL$.

15.3 Timer 2 PWM0 Pulse Output Trigger Mode

T2 timer builds in PWM0 pulse trigger mode controlled by $T2SF$ bit set as "1". When T2 timer's PWM0 pulse output trigger mode is enabled ($T2SF = 1$) and PWM0 pulse trigger is issued, T2 timer starts to count. If T2 timer doesn't overflow, and then a new PWM0 pulse output trigger is issued, T2C will be reloaded from T2R automatically and count again. If T2 timer overflows (from 0xFFFF to 0x0000), and PWM0 pulse output finished, T2C will be reloaded from T2R automatically, continue counting, and issue a time out signal ($TF2 = 1$). If T2 timer overflows, but PWM0 pulse output doesn't

finish, PWM0 pulse output is disabled immediately (PW0PO/PW0EN/CM0SF bits are cleared, and pulse output pin exchanges to GPIO mode.), and then T2C will be reloaded from T2R automatically, continue counting, and issue a time out signal (TF2 = 1). This is an abnormal condition, and the PWM0 pulse must be enabled by program again.



- (1). After T2EN is enabled, T2 timer doesn't count and wait the PWM0 pulse trigger signal. When PWM0 pulse start to output, the trigger signal will trigger T2 timer to count.
- (2). When T2 timer overflows, and no PWM0 pulse output trigger signal, the T2C is reloaded from T2R automatically and issues TF2 time-out flag.
- (3). If T2 timer doesn't overflow, but PWM0 pulse output trigger is issued, the T2C is reloaded from T2R automatically, but not issue TF2 time-out flag.
- (4). If T2 timer overflows, but PWM0 pulse output doesn't finish, PWM0 pulse output is disabled immediately (PW0PO/PW0EN/CM0SF bits are cleared, and pulse output pin exchanges to GPIO mode.), and then T2C will be reloaded from T2R automatically, continue counting, and issue a time out signal (TF2 = 1).
- (5). PWM0 pulse output function is enabled by program, and T2 timer works in PWM0 pulse output

trigger mode.

15.4 Timer 2 Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2M	T2EN	T2RATE2	T2RATE1	T2RATE0	T2CKS	-	-	T2SF
T2CH	T2C15	T2C14	T2C13	T2C12	T2C11	T2C10	T2C9	T2C8
T2CL	T2C7	T2C6	T2C5	T2C4	T2C3	T2C2	T2C1	T2C0
T2RH	T2R15	T2R14	T2R13	T2R12	T2R11	T2R10	T2R9	T2R8
T2RL	T2R7	T2R6	T2R5	T2R4	T2R3	T2R2	T2R1	T2R0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
IEN2	ECMP4	ECMP3	ECMP2	ECMP1	ECMP0	ET2	EADC	ELVD
IRCON2	CM4F	CM3F	CM2F	CM1F	CM0F	TF2	ADCF	LVDF

T2M Register (0xF070)

Bit	Field	Type	Initial	Description
7	T2EN	R/W	0	Timer 2 function 0: Disable 1: Enable
6..4	T2RATE[2:0]	R/W	000	Clock divider of Timer 2 f_{CKS} clock source 000: $f_{CKS} / 128$ 001: $f_{CKS} / 64$ 010: $f_{CKS} / 32$ 011: $f_{CKS} / 16$ 100: $f_{CKS} / 8$ 101: $f_{CKS} / 4$ 110: $f_{CKS} / 2$ 111: $f_{CKS} / 1$
3	T2CKS	R/W	0	Timer 1 f_{CKS} clock source selection. 0: Fcpu 1: Fhosc
0	T2SF	R/W	0	T2 timer trigger function control bit. 0 = Disable (Normal timer mode). 1 = Enable (PWM0 pulse output trigger mode).
Else	Reserved	R/W	000	

T2CH/T2CL Registers (T2CH: 0xF071, T2CL: 0xF072)

Bit	Field	Type	Initial	Description
7..0	T2CH/L	R/W	0x00	16-bit Timer 2 counter buffer.

T2RH/T2RL Registers (T2RH: 0xF073, T2RL: 0xF074)

Bit	Field	Type	Initial	Description
7..0	T2RH/L	R/W	0x00	16-bit Timer 2 counter reload buffer.

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
Else				Refer to other chapter(s)

IEN2 Register (0xC8)

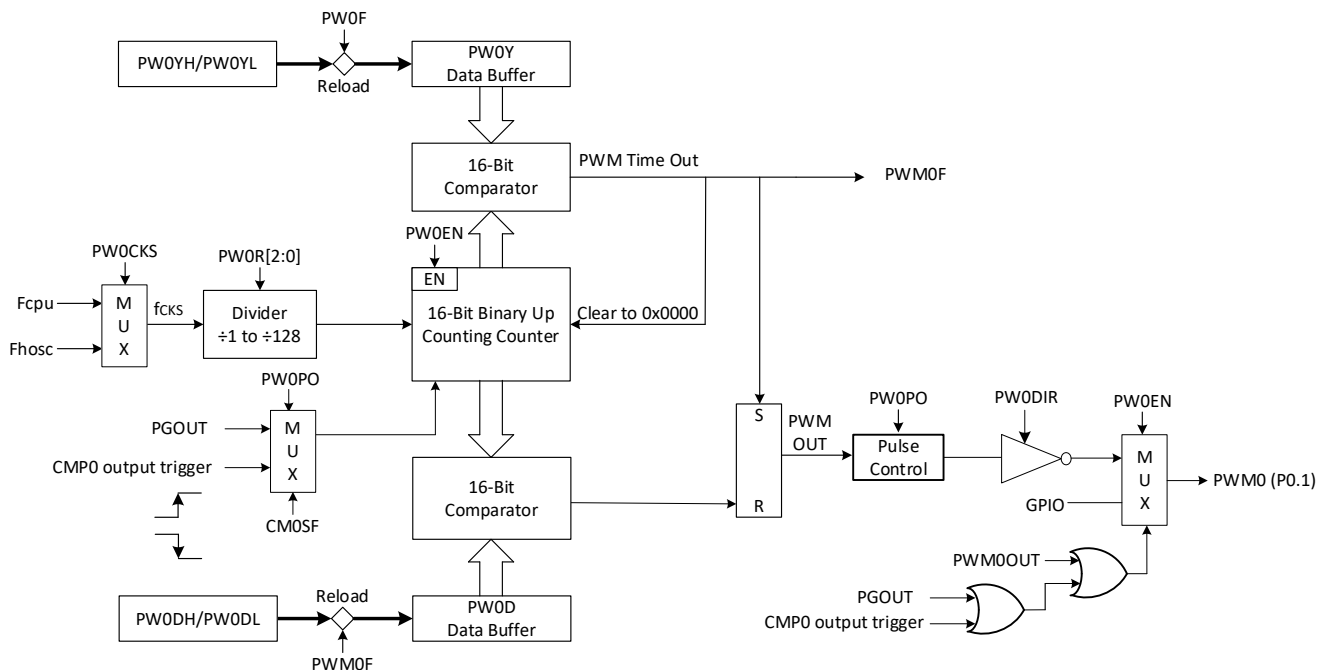
Bit	Field	Type	Initial	Description
2	ET2	R/W	0	Timer 2 interrupt control bit. 0: Disable T2 interrupt function. 1: Enable T2 interrupt function.

IRCON2 Register (0xA7)

Bit	Field	Type	Initial	Description
2	TF2	R/W	0	Timer 2 overflow event 0: Timer 2 does not have any overflow event 1: Timer 2 has overflowed This bit can be cleared automatically by interrupt handler, or manually by firmware
Else				Refer to other chapter(s)

16 PWM0

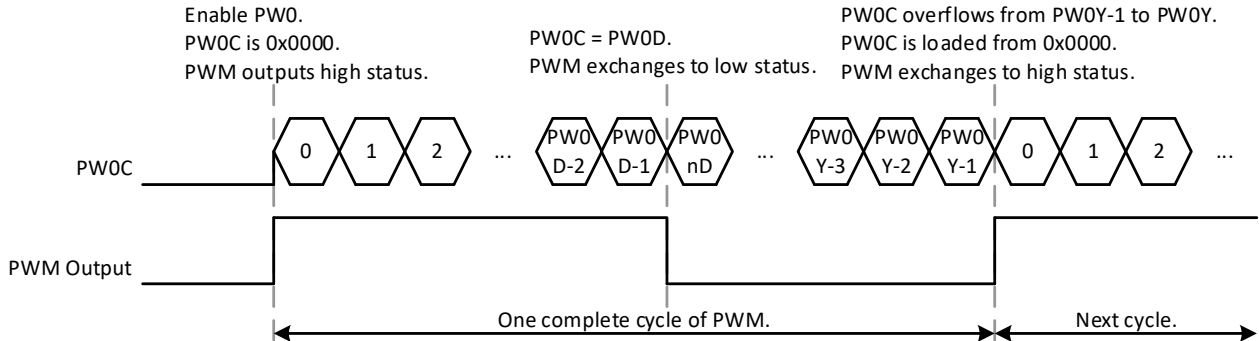
The PW0 timer is a 16-bit up counting timer and supports 1-channel general PWM function, and one pulse PWM function. By the counter reaches the up-boundary value (PW0Y), it clears its counter and triggers an interrupt signal. PWM's duty cycle is controlled by PW0D register. The PWM also support one pulse output signal which can disables itself by the end of first PWM cycle. Thus, only one pulse would be generated in this condition. The PW0 timer builds in IDLE Mode wake-up function if interrupt enable. When timer overflow occurs (counts from PW0Y-1 to PW0Y), PW0F would be issued immediately which can read/write by firmware. PW0 interrupt function is controlled by EPW0.



16.1 General PWM

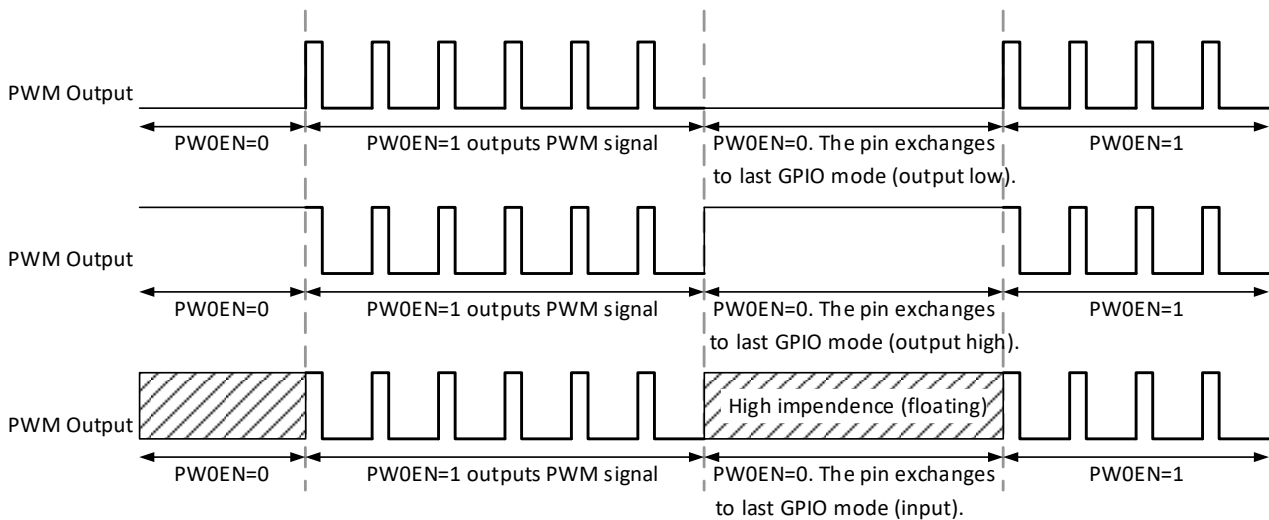
PW0 timer builds in PWM function controlled by PW0EN register and PWM0OUT bit. PWM0 (P0.1) is output pin. The output pin is shared with GPIO pin controlled by PWM0OUT bits. When output PWM function, we must be set PW0EN = 1. When PWM output signal synchronize finishes, the PWM channel exchanges from GPIO to PWM output. When PW0EN = 0, the PWM channel returns to GPIO mode and last status. PWM signal is generated from the result of PW0Y and PW0D comparison combination. When PW0C starts to count or returns to 0x0000, the PWM outputs high status which is the PWM initial status. PW0C is loaded new data from PW0Y register to decide PWM cycle and resolution. PW0C keeps counting, and the system compares PW0C and PW0D. When PW0C = PW0D, the PWM output status exchanges to low and PW0C keeps counting. When PW0 timer overflow occurs (PW0Y-1 to 0x0000), and one cycle of PWM signal finishes. PW0C is reloaded from 0x0000 automatically, and PWM output status exchanges to high for next cycle. PW0D decides the high duty duration, and PW0Y decides the resolution and cycle of PWM. PW0D can't be larger than PW0Y, or

the PWM signal is error. PWM has two clock source options: Fcpu, Fhosc. PWM selects the clock source via the PW0CKS bit and selects the frequency division by the PW0RATE[2:0] bits. The Clock division range is /1 ~ /128.



PWM Period = PW0Y

PWM duty = (PW0D): (PW0Y)

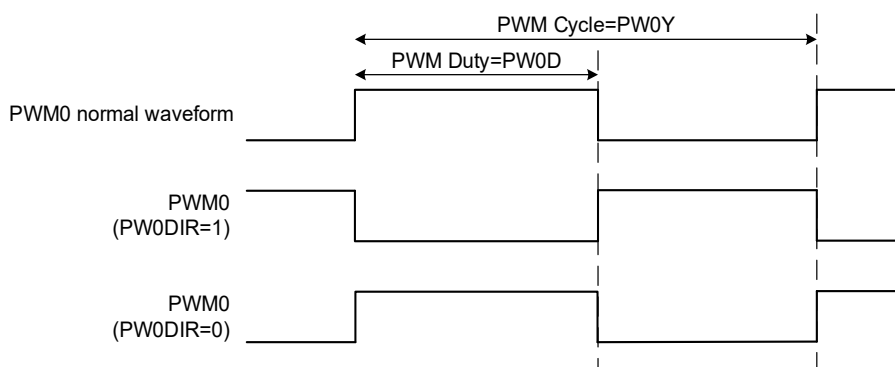


*** Note:**

- 1. PWM period and duty must be greater than system clock Fcpu.**
- 2. Switching between different PWM modes must be done while the PWM function is disabled. Thus, before changing the status of PW0PO or CM0SF bit, the PWM function must be disabled (PW0EN=0) first. Then, update the status of PW0PO or CM0SF bit and re-enable the PWM function (PW0EN=1) in the end.**
- 3. When CM0SF bit is enabled, PWM output is triggered by CMP0 output edge. In this condition, PGOUT bit can't be modified by firmware.**

16.2 Inverse Mode

The PWM builds in inverse output function. The inverse mode is controlled by PWODIR. When PWODIR = 1, the PWM0 outputs the inverse PWM signal of PWM. When PWODIR = 0, the PWM0 outputs the non-inverse PWM signal of PWM. The inverse PWM output waveform is below diagram.

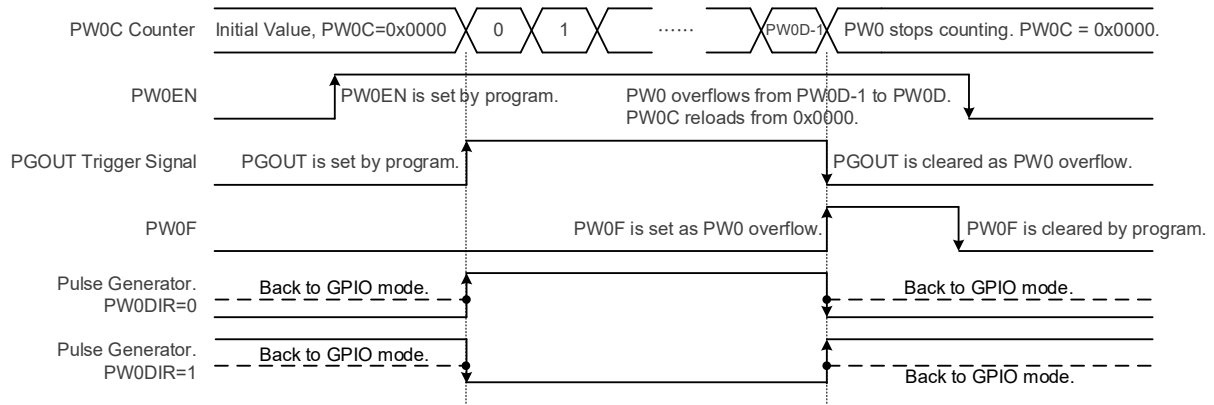


16.3 One Pulse PWM

When PW0PO = 0, PW0 is PWM function mode. When PW0PO = 1 and PW0EN = 1, PW0 is pulse generator mode. The pulse generator needs a start trigger signal to control PW0 counter and pulse signal output. When PW0PO = 1, PW0 counter keeps stopping, and PW0D register value is set by program. PW0D value decides the pulse width. When the trigger event occurs, PW0 counter starts to count, and pulse output pin outputs pulse status. When PW0 counter overflow from PW0D-1 to PW0D, pulse signal finishes and returns to idle status. The PW0C stops counting and reloads new value to 0x0000. In pulse generator mode, the PW0F is issued as PW0 counter overflow from PW0D-1 to PW0D. During pulse generator operating, to change pulse width is through PW0D, not PW0C, or the pulse width would be error.

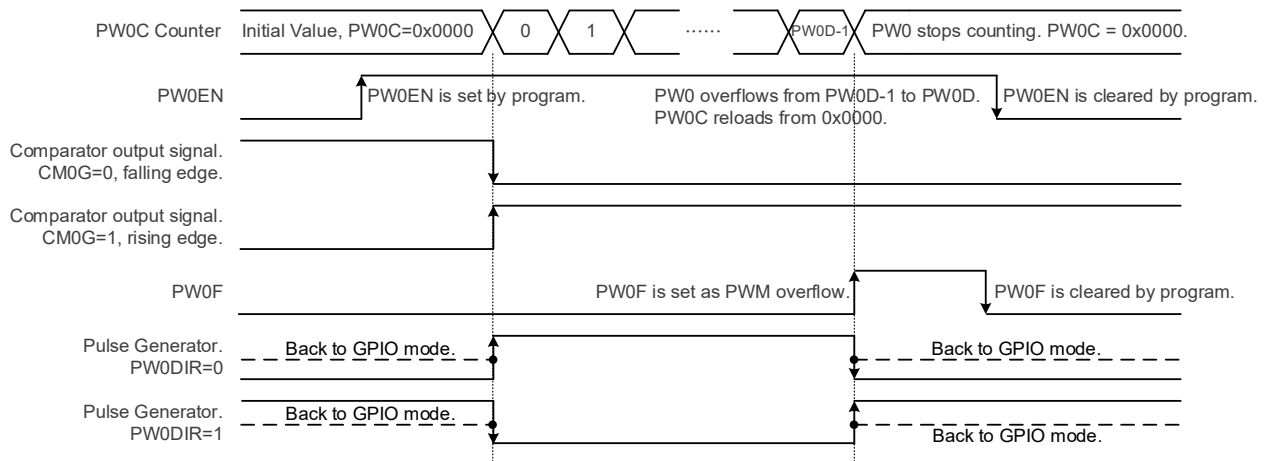
The pulse generator has two trigger sources and control by the CM0SF bit. One is PGOUT bit (CM0SF=0), and the other is comparator 0 output edge (CM0SF=1). If the trigger source is PGOUT bit, set PGOUT bit to output pulse signal by program, and PGOUT bit is cleared as PW0 counter overflow from PW0D-1 to PW0D. To output next pulse is to set PGOUT bit by program again.

- **PW0PO=1, CM0SF=0: PW0EN must be set as "1". PW0 16-bit binary up counter is controlled by PGOUT bit. If PGOUT bit is set as "1" by program, PW0 starts to count. If PWM0 overflows from PW0D-1 to PW0D, PW0 stops counting, PGOUT bit is cleared automatically, PW0F is issued, and PW0C reloads new value from 0x0000. It is necessary to set PGOUT = 1 by program making PW0C counts again.**



If the trigger is comparator 0 output delay edge (rising edge and falling edge controlled by comparator control register's CM0G bit), pulse starts to output as trigger edge condition occurrence. When PW0 overflows, pulse output pin returns to GPIO mode (idle status).

- **PW0PO=1, CM0SF=1: PW0EN must be set as "1". PWM0 16-bit binary up counter is controlled by comparator 0 output edge condition. The trigger edge can be selected through CM0G bit. If comparator output edge occurs, PW0 starts to count. If PW0 overflows from PW0D-1 to PW0D, PW0 stops counting, PW0F is issued, and PW0C reloads new value from 0x0000.**



16.4 PWM0 Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PW0M	PW0EN	PW0rate2	PW0rate1	PW0rate0	PW0CKS	PW0DIR	PW0PO	PWM0OUT
PW0M1	SFDL	-	-	-	-	-	-	PGOUT
PW0YH	PW0Y15	PW0Y14	PW0Y13	PW0Y12	PW0Y11	PW0Y10	PW0Y9	PW0Y8
PW0YL	PW0Y7	PW0Y6	PW0Y5	PW0Y4	PW0Y3	PW0Y2	PW0Y1	PW0Y0
PW0DH	PW0D15	PW0D14	PW0D13	PW0D12	PW0D11	PW0D10	PW0D9	PW0D8
PW0DL	PW0D7	PW0D6	PW0D5	PW0D4	PW0D3	PW0D2	PW0D1	PW0D0
IEN0	EAL	-	-	-	ET1	-	ETO	EX0

IEN3	-	-	-	-	-	EPW0	-	-
IRCON3	-	-	-	-	-	PW0F	-	-

PW0M Registers (0xB1)

Bit	Field	Type	Initial	Description
7	PW0EN	R/W	0	PW0 function 0: Disable 1: Enable*
6..4	PW0R[2:0]	R/W	000	PWM timer clock source 000: $f_{CKS} / 128$ 001: $f_{CKS} / 64$ 010: $f_{CKS} / 32$ 011: $f_{CKS} / 16$ 100: $f_{CKS} / 8$ 101: $f_{CKS} / 4$ 110: $f_{CKS} / 2$ 111: $f_{CKS} / 1$
3	PW0CKS	R/W	0	PWM0 f_{CKS} clock source selection. 0: Fcpu 1: Fhosc
2	PW0DIR	R/W	0	PWM0 (P0.1) inverse output control 0: PWM0 outputs non-inverse PWM signal 1: PWM0 outputs inverse PWM signal
1	PW0PO	R/W	0	One pulse output function control bit. 0: Disable 1: Enable
0	PWM0OUT	R/W	0	PWM0 output control bit. When PW0PO = 0, PWM0OUT control PWM output. 0: Disable PWM0 output function. 1: Enable PWM0 output function, and PWM0 signal outputs through P0.1 pin. When PW0PO = 1, PWM0OUT no function.

* When the period is setting 0x0000, after PWM is set enable bit, the PWM will stop and the period can't update.

PW0M1 Registers (0xB6)

Bit	Field	Type	Initial	Description
7	SFDL	R/W	0	Special function duty buffer load control bit. Only use in CMP2 special function

				0: PW0DH/L is the setting value by FW 1: PW0DH/L is the value shrink by HW.
0	PGOUT	R/W	0	PWM0 output control bit. When PW0PO = 0, PGOUT no function When PW0PO = 1, PGOUT control PWM output. 0: Stop pulse output, or the end of pulse output cleared automatically. 1: Enable pulse output.

PW0YH/PW0YL Registers (PW0YH: 0xB3, PW0YL: 0xB2)

Bit	Field	Type	Initial	Description
7..0	PW0YH/L	R/W	0x00	16-bit PWM0 period control*.

* The period configuration must be setup completely before starting PWM function.

PW0DH/PW0DL Registers (PW0DH: 0xB5, PW0DL: 0xB4)

Bit	Field	Type	Initial	Description
7..0	PW0DH/L	R/W	0x00	16-bit PWM0 duty control*.

* The duty configuration must be setup completely before starting PWM function.

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
Else				Refer to other chapter(s)

IEN3 Register (0x98)

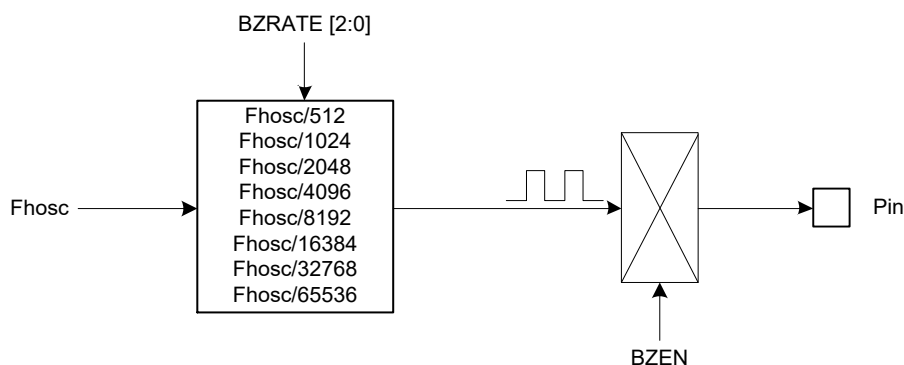
Bit	Field	Type	Initial	Description
2	EPW0	R/W	0	PWM0 interrupt control bit. 0: Disable PWM0 interrupt function. 1: Enable PWM0 interrupt function.
Else				Refer to other chapter(s)

IRCON3 Register (0xC9)

Bit	Field	Type	Initial	Description
2	PW0F	R/W	0	PWM0 interrupt request flag. 0: None PWM0 interrupt request 1: PWM0 interrupt request.
Else				Refer to other chapter(s)

17 2K/4K BUZZER GENERATOR

The MCU builds in Buzzer generator to drive external buzzer device. The buzzer generator purpose is to drive 2KHz or 4KHz buzzer. Adjusting buzzer output frequency is through BZM register. The buzzer output pin is shared with GPIO. When BZEN = 1, the pin outputs buzzer carry signal. When BZEN = 0, the pin returns to GPIO last condition (input mode, output high or output low status).



- The buzzer frequency is divided from Fosc controlled by BZrate bits, and Fosc decides the buzzer frequency. The selection table is as following.

BZrate [2:0]	Buzzer Rate Division	Buzzer Rate
		IHRC_32MHz
000	Fosc/512	64KHz
001	Fosc/1024	32KHz
010	Fosc/2048	16KHz
011	Fosc/4096	8KHz
100	Fosc/8192	4KHz
101	Fosc/16384	2KHz
110	Fosc/32768	1KHz
111	Fosc/65536	500Hz

- The buzzer target frequency is 2KHz and 4KHz. It is important to choice a good Fosc rate to obtain the correct buzzer frequency. The above table shows 2KHz/4KHz buzzer frequency configurations.

17.1 Buzzer Register

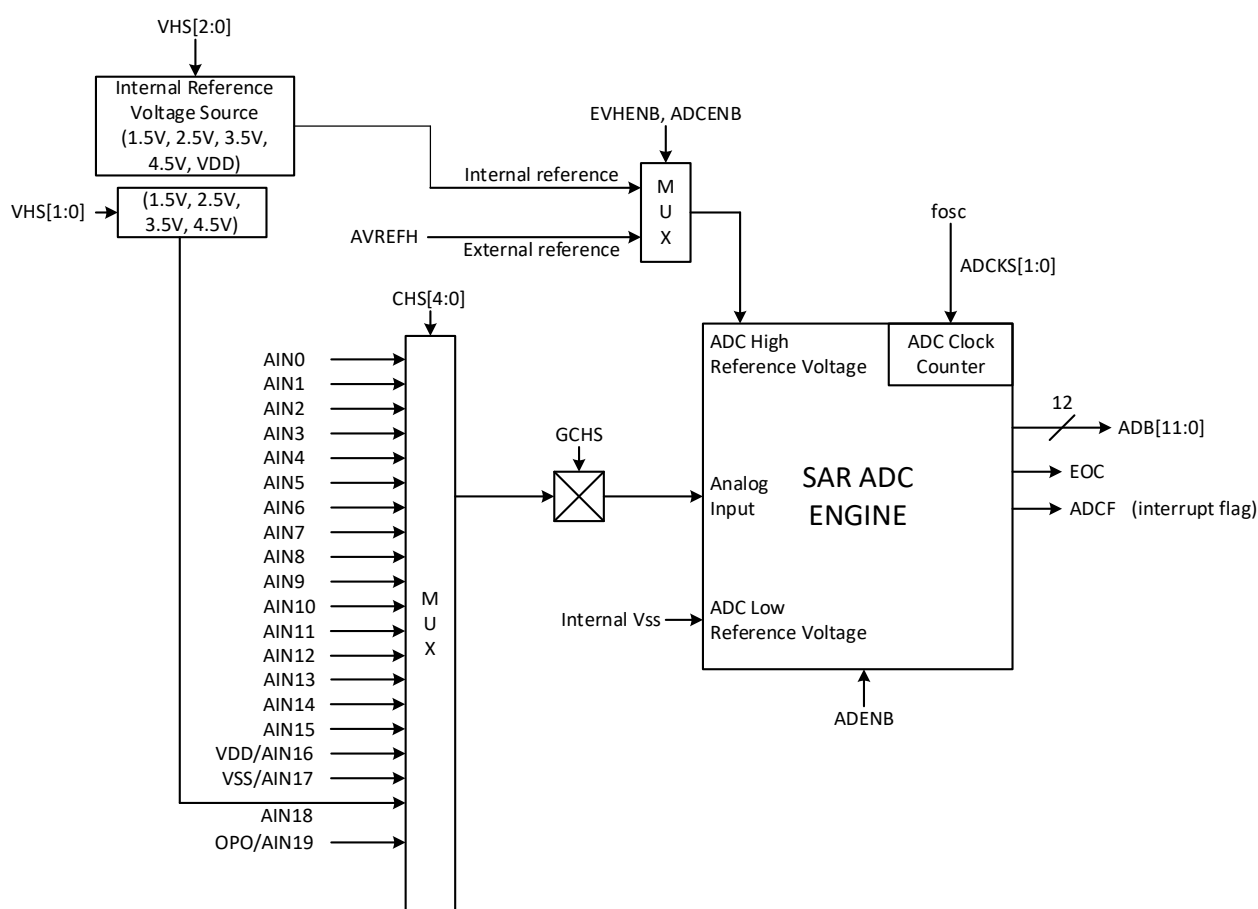
Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BZM	BZEN	BZrate2	BZrate1	BZrate0	-	-	-	-

BZM Register (0xD9)

Bit	Field	Type	Initial	Description
7	BZEN	R/W	0	Buzzer output control bit. 0: Disable BZ output and BZ output pin(P1.6) transfers to I/O last status. 1: Enable BZ output and disable GPIO(P1.6) function.
6..4	BZRATE[2:0]	R/W	000	Buzzer rate control bits. 000: Fhosc/512 001: Fhosc/1024 010: Fhosc/2048 011: Fhosc/4096 100: Fhosc/8192 101: Fhosc/16384 110: Fhosc/32768 111: Fhosc/65536
3..0	Reserved	R	0x0	

18 ADC

The analog to digital converter (ADC) is SAR structure with 16-input sources and up to 4096-step resolution to transfer analog signal into 12-bits digital buffers. The ADC builds in 16-channel input source to measure 16 different analog signal sources. The ADC resolution is 12-bit. The ADC has four clock rates to decide ADC converting rate. The ADC reference high voltage includes 6 sources. Five internal power source including VDD, 4.5V, 3.5V, 2.5V and 1.5V. The other one is external reference voltage input pin from AVREFH pin. The ADC builds in P0CON/P1CON/P4CON registers to set pure analog input pin. After setup ADENB and ADS bits, the ADC starts to convert analog signal to digital data. ADC can work in idle mode. After ADC operating, the system would be waked up from idle mode to normal mode if interrupt enable. ADC can not work in stop mode.



18.1 Configurations of Operation

These configurations must be setup completely before starting ADC converting. ADC is configured using the following steps:

1. Choose and enable the start of conversion ADC input channel. (By CHS[4:0] bits and GCHS bit)
2. The GPIO mode of ADC input channel must be set as input mode. (By PnM register)
3. The internal pull-up resistor of ADC input channel must be disabled. (By PnUR register)
4. The configuration control bit of ADC input channel must be set. (By PnCON register)
5. Choose ADC high reference voltage. (By VREFH register)
6. Choose ADC Clock Rate. (By ADCKS[1:0] bits)
7. Enable ADC control bit ADENB=1.
8. Wait 5 ADC clock cycle.
9. Enable ADC offset calibration bit (ACS=1) and wait for this bit to become zero (ACS=0). It means that the calibration is complete.
10. Enable ADC calibration value enable bit (CALIVALENB=1).
11. Start ADC conversion ADS=1, the ADC ready to convert analog signal to digital data.

When ADC IP is enabled by ADENB bit, it is necessary to make an ADC start-up by program. Writing a 1 to the ADS bit of register ADM. After setup ADENB and ADS bits, the ADC starts to convert analog signal to digital data. The ADS bit is reset to logic 0 when the conversion is complete. When the conversion is complete, the ADC circuit will set EOC and ADCF bits to "1" and the digital data outputs in ADB and ADR registers. If ADC interrupt function is enabled (EADC = 1), the ADC interrupt request occurs and executes interrupt service routine when ADCF is "1" after ADC converting. Clear ADCF by hardware automatically in interrupt procedure.

18.2 ADC input channel

The ADC builds in 16-channel input source (AIN0 – AIN15) to measure 16 different analog signal sources controlled by CHS[4:0] and GCHS bits. The AIN18 is internal 1.5V or 2.5V or 3.5V or 4.5V input channel. There is no any input pin from outside. In this time ADC reference voltage must be internal VDD, not internal 1.5V or 2.5V or 3.5V or 4.5V. AIN18 can be a good battery detector for battery system. To select appropriate internal AVREFH level and compare value, a high performance and cheaper low battery detector is built in the system. AIN16 is VDD channel. AIN17 is VSS channel. AIN19 is OP output (OPO) channel. Other channel is reserved.

CHS[4:0]	Channel	Pin name	Remark
00000	AIN0	P1.0	-
00001	AIN1	P1.1	-
00010	AIN2	P1.2	-
00011	AIN3	P1.3	-
00100	AIN4	P1.4	-
00101	AIN5	P1.5	-
00110	AIN6	P1.6	-
00111	AIN7	P4.2	-
01000	AIN8	P4.1	-
01001	AIN9	P4.0	-
01010	AIN10	P0.7	-
01011	AIN11	P0.6	-
01100	AIN12	P0.5	-
01101	AIN13	P0.4	-
01110	AIN14	P0.3	-
01111	AIN15	P0.2	-
10000	AIN16	VDD	-
10001	AIN17	VSS	-
10010	AIN18	Internal 1.5V or 2V or 3V or 4.5V	Battery detector channel
10011	AIN19	OPO	-
Other	-	-	Reserved

18.2.1 Pin Configuration

ADC input channels are shared with Port0, Port1 and Port4. ADC channel selection is through CHS[4:0] bit. Only one pin of Port0, Port1 and Port4 can be configured as ADC input in the same time. The pins of Port0, Port1 and Port4 configured as ADC input channel must be set input mode, disable internal pull-up and enable P0CON/P1CON/P4CON first by program. After selecting ADC input channel through CHS[4:0], set GCHS bit as “1” to enable ADC channel function.

ADC input pins are shared with digital I/O pins. Connect an analog signal to CMOS digital input pin, especially, the analog signal level is about 1/2 VDD will cause extra current leakage. In the power down mode, the above leakage current will be a big problem. Unfortunately, if users connect more than one analog input signal to Port0, Port1, Port4 will encounter above current leakage situation. Write “1” into PnCON register will configure related pin as pure analog input pin to avoid current leakage.

Note that When ADC pin is general I/O mode, the bit of P0CON, P1CON and P4CON must be set to

“0”, or the digital I/O signal would be isolated.

18.3 Reference Voltage

The ADC builds in six high reference voltage source controlled through VREFH register. There are one external voltage source and five internal voltage source (VDD, 4.5V, 3.5V, 2.5V, 1.5V). When EVHENB bit is “1”, ADC reference voltage is external voltage source from AVREFH/P1.0. In the condition, P1.0 GPIO mode must be set as input mode and disable internal pull-up resistor.

If EVHENB bit is “0”, ADC high reference voltage is from internal voltage source selected by VHS[2:0] bits. If VHS[2:0] is “1xx”, ADC high reference voltage is VDD. If VHS[1:0] is “010”, ADC high reference voltage is 4.5V. If VHS[2:0] is “001”, ADC high reference voltage is 3.5V. If VHS[2:0] is “000”, ADC high reference voltage is 2.5V. If VHS[2:0] is “011”, ADC high reference voltage is 1.5V. The limitation of internal high reference voltage application is VDD can't below each of internal high voltage level, or the level is equal to VDD. If AIN18 channel is selected as internal 1.5 or 2.5V or 3.5V or 4.5V input channel. There is no any input pin from outside. In this time ADC high reference voltage must be internal VDD, not internal 1.5V/2.5V/3.5V/4.5V.

18.3.1 Signal Format

ADC sampling voltage range is limited by high/low reference voltage. The ADC low reference voltage is VSS. The ADC high reference voltage includes internal VDD/4.5V/3.5V/2.5V/1.5V and external reference voltage source from P1.0/AVREFH pin controlled by EVHENB bit. ADC reference voltage range limitation is “(ADC high reference voltage - low reference voltage) $\geq$ 1.5V”. ADC low reference voltage is VSS = 0V. So ADC high reference voltage range is 1.5V to VDD. The range is ADC external high reference voltage range.

- ADC Internal Low Reference Voltage = 0V.
- ADC Internal High Reference Voltage = VDD/4.5V/3.5V/2.5V/1.5V. (EVHENB=0)
- ADC External High Reference Voltage = 1.5V to VDD. (EVHENB=1)

ADC sampled input signal voltage must be from ADC low reference voltage to ADC high reference. If the ADC input signal voltage is over the range, the ADC converting result is error (full scale or zero).

- ADC Low Reference Voltage $\leq$ ADC Sampled Input Voltage $\leq$ ADC High Reference Voltage

18.4 Converting Time

The ADC converting time is from ADS=1 (Start to ADC convert) to EOC=1 (End of ADC convert). The converting time duration is depend on ADC clock rate. 12-bit ADC's converting time is $1 / (\text{ADC clock}) * 16$ sec. ADC only one clock sources: Fosc, ADCKS[2:0] bits: 000/001 = Reserved, 010 = Fosc/4, 011 = Fosc/8, 100 = Fosc/16, 101 = Fosc/32, 110 = Fosc/64, 111 = Fosc/128.

The ADC converting time affects ADC performance. If input high rate analog signal, it is necessary to

select a high ADC converting rate. If the ADC converting time is slower than analog signal variation rate, the ADC result would be error. So to select a correct ADC clock rate to decide a right ADC converting rate is very important.

$$12 \text{ bits ADC conversion time} = \frac{16}{\text{ADC clock rate}}$$

ADCKS[2:0]	ADC clock rate	Fhosc = 32MHz	
		Converting time	Converting rate
010	Fhosc/4	$1/(32\text{MHz}/4)*16$ = 2us	500 kHz
011	Fhosc/8	$1/(32\text{MHz}/8)*16$ = 4us	250 kHz
100	Fhosc/16	$1/(32\text{MHz}/16)*16$ = 8us	125 kHz
101	Fhosc/32	$1/(32\text{MHz}/32)*16$ = 16us	62.5 kHz
110	Fhosc/64	$1/(32\text{MHz}/64)*16$ = 32us	31.25 kHz
111	Fhosc/128	$1/(32\text{MHz}/64)*16$ = 64us	15.625 kHz

18.5 Data Buffer

ADC data buffer is 12-bit length to store ADC converter result. The high byte is ADB register, and the low-nibble is ADR[3:0] bits. The ADB register is only 8-bit register including bit 4 – bit 11 ADC data. To combine ADB register and the low-nibble of ADR will get full 12-bit ADC data buffer. The ADC data buffer is a read-only register and the initial status is unknown after system reset.

Table 18-1 The AIN input voltage vs. ADB output data

AIN n	ADB11	ADB10	ADB9	ADB8	ADB7	ADB6	ADB5	ADB4	ADB3	ADB2	ADB1	ADB0
0/4096*VREFH	0	0	0	0	0	0	0	0	0	0	0	0
1/4096*VREFH	0	0	0	0	0	0	0	0	0	0	0	1
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.
4094/4096*VREFH	1	1	1	1	1	1	1	1	1	1	1	0
4095/4096*VREFH	1	1	1	1	1	1	1	1	1	1	1	1

18.6 ADC Offset Calibration

The ADC builds in ADC offset calibration. The calibration must be run at least one time before conversion starts. The internal calibration register will keep the value even when ADC is disabled. The application must not use the ADC during calibration and must wait until it is complete.

Calibration should be performed before starting A/D conversion. It removes the offset error which may vary from chip to chip due to process variation. The calibration is initiated by setting the ACS bit to 1. After the calibration is complete, the ACS bit automatically reset to 0. Before A/D conversion, it is necessary to enable the calibration value by setting the CALIVALENB bit.

The internal analog calibration is kept if the ADC is disabled. When the ADC operating conditions change (VDD changes are the main contributor to ADC offset variations and temperature change to a lesser extend), it is recommended to re-run a calibration cycle.

18.7 ADC Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADM	ADENB	ADS	EOC	CHS4	CHS3	CHS2	CHS1	CHS0
ADB	ADB11	ADB10	ADB9	ADB8	ADB7	ADB6	ADB5	ADB4
ADR	-	GCHS	-	-	ADB3	ADB2	ADB1	ADB0
VREFH	EVHENB	-	-	-	VREFSW	VHS2	VHS1	VHS0

ADCAL	ACS	CALIVALENB	-	-	-	ADCKS2	ADCKS1	ADCKS0
P0CON	P0CON7	P0CON6	P0CON5	P0CON4	P0CON3	P0CON2	-	P0CON0
P1CON	-	P1CON6	P1CON5	P1CON4	P1CON3	P1CON2	P1CON1	P1CON0
P4CON	-	-	-	-	-	P4CON2	P4CON1	P4CON0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
IEN2	ECMP4	ECMP3	ECMP2	ECMP1	ECMP0	ET2	EADC	ELVD
IRCON2	CM4F	CM3F	CM2F	CM1F	CM0F	TF2	ADCF	LVDF

ADM Register (0xD2)

Bit	Field	Type	Initial	Description
7	ADENB	R/W	0	ADC control bit. In stop mode, disable ADC to reduce power consumption. 0: Disable 1: Enable
6	ADS	R/W	0	ADC conversion control. Write 1: Start ADC conversion (automatically cleared by the end of conversion)
5	EOC	R	0	ADC status bit. 0: ADC progressing 1: End of conversion (automatically set by hardware; manually cleared by firmware)
4..0	CHS[4:0]	R/W	0x00	ADC input channel select bit. 00000: AIN0, 00001: AIN1, 00010: AIN2, 00011: AIN3, 00100: AIN4, 00101: AIN5, 00110: AIN6, 00111: AIN7, 01000: AIN8, 01001: AIN9, 01010: AIN10, 01011: AIN11, 01100: AIN12, 01101: AIN13, 01110: AIN14, 01111: AIN15, 10000: VDD, 10001: VSS, 10010: AIN18 ^{*(1)} , 10011: OPO, Others: Reserved.

*(1) The AIN18 is internal 1.5V or 2V or 3V or 4.5V input channel. There is no any input pin from outside. In this time ADC reference voltage must be internal VDD and External voltage, not internal 1.5V or 2.5V or 3.5V or 4.5V.

ADB Register (0xD3)

Bit	Field	Type	Initial	Description
7..0	ADB[11:4]	R	-	ADC Result Bit [11:4]* in 12-bit ADC resolution mode.

* ADC data buffer is 12-bit length to store ADC converter result. The high byte is ADB register, and the low-nibble is ADR[3:0] bits.

ADR Register (0xD4)

Bit	Field	Type	Initial	Description
7	Reserve	R	0	
6	GCHS	R/W	0	ADC global channel select bit. 0: Disable AIN channel 1: Enable AIN channel
5..4	Reserve	R	0	
3..0	ADB[3:0]	R	-	ADC Result Bit [3:0]* in 12-bit ADC resolution mode.

* ADC data buffer is 12-bit length to store ADC converter result. The high byte is ADB register, and the low-nibble is ADR[3:0] bits.

VREFH Register (0xD5)

Bit	Field	Type	Initial	Description
7	EVHENB	R/W	0	ADC internal reference high voltage control bit. 0: Enable ADC internal VREFH function. AVREFH/P4.0 pin is GPIO. 1: Disable ADC internal VREFH function. AVREFH/P4.0 pin is external AVREFH ^{*(1)} input pin.
6..3	Reserved	R	0	
2..0	VHS[2:0]	R/W	000	ADC internal reference high voltage selects bits. ^{*(2)} 000: VREFH = 2.5V 001: VREFH = 3.5V 010: VREFH = 4.5V 011: VREFH = 1.5V 100: VREFH = VDD and AIN18 = 2.5V 101: VREFH = VDD and AIN18 = 3.5V 110: VREFH = VDD and AIN18 = 4.5V 111: VREFH = VDD and AIN18 = 1.5V

^{*(1)} The AVREFH level must be between the VDD and 1.5V.

^{*(2)} If AIN18 channel is selected as internal 1.5V or 2.5V or 3.5V or 4.5V input channel. There is no any input pin from outside. In this time ADC reference high voltage must be internal VDD or External voltage, not internal 1.5V/2.5V/3.5V/4.5V.

ADCAL Register (0xD1)

Bit	Field	Type	Initial	Description
7	ACS	R/W	0	ADC calibration control ^{*(1)} Write 1: Start ADC calibration (automatically cleared by the end of calibration)
6	CALIVALENB	R/W	0	ADC calibration value enable bit 0: ADC conversion without calibration value 1: ADC conversion with calibration value
5..3	Reserve	R	0	
2..0	ADCKS[2:0]	R/W	0	ADC's clock source select bit. 000/001 = Reserved, 010 = Fhosc/4, 011 = Fhosc/8, 100 = Fhosc/16, 101 = Fhosc/32, 110 = Fhosc/64, 111 = Fhosc/128

^{*(1)} The ADC needs to be calibrated one time after MCU power-up, to ensure accurate conversion results.

P0CON Register (0xF010)

Bit	Field	Type	Initial	Description
7..0	P0CON[7:0]	R/W	0x00	P0 configuration control bit*. 0: P0 can be analog input pin (ADC input pin) or digital GPIO pin. 1: P0 is pure analog input pin and can't be a digital GPIO pin.

* P0CON [7:0] will configure related Port0 pin as pure analog input pin to avoid current leakage.

P1CON Register (0xF011)

Bit	Field	Type	Initial	Description
6..0	P1CON[6:0]	R/W	0x00	P1 configuration control bit*. 0: P1 can be analog input pin (ADC input pin) or digital GPIO pin. 1: P1 is pure analog input pin and can't be a digital GPIO pin.

* P1CON [6:0] will configure related Port1 pin as pure analog input pin to avoid current leakage.

P4CON Register (0xF014)

Bit	Field	Type	Initial	Description
2..0	P4CON[2:0]	R/W	0x00	P4 configuration control bit*. 0: P4 can be analog input pin (ADC input pin) or digital GPIO pin.

1: P4 is pure analog input pin and can't be a digital GPIO pin.

* P4CON [2:0] will configure related Port4 pin as pure analog input pin to avoid current leakage.

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
Else				Refer to other chapter(s)

IEN2 Register (0xC8)

Bit	Field	Type	Initial	Description
1	EADC	R/W	0	ADC interrupt control bit. 0: Disable ADC interrupt function. 1: Enable ADC interrupt function.
Else				Refer to other chapter(s)

IRCON2 Register (0xA7)

Bit	Field	Type	Initial	Description
1	ADCF	R/W	0	ADC interrupt request flag. 0 = None ADC interrupt request. 1 = ADC interrupt request.
Else				Refer to other chapter(s)

19 UART0/1

The UARTn (n=0,1 in the following descriptions) provides a flexible full-duplex synchronous/asynchronous receiver/transmitter. The serial interface provides an up to 500 kHz flexible full-duplex transmission. It can operate in four modes (one synchronous and three asynchronous). Mode 0 is a shift register mode and operates as synchronous transmitter/receiver. In Mode 1-Mode 3 the UART operates as asynchronous transmitter/receiver with 8-bit or 9-bit data. The transfer format has start bit, 8-bit/ 9-bit data and stop bit. Transmission is started by writing to the SnBUF register. After reception, input data are available after completion of the reception in the SnBUF register. TB8n/RB8n bit can be used as the 9th bit for transmission and reception in 9-bit UART mode. Programmable baud rate supports different speed peripheral devices.

The UART features include the following:

- Full-duplex, 2-wire synchronous/asynchronous data transfer.
- Programmable baud rate.
- 8-bit shift register: operates as synchronous transmitter/receiver
- 8-bit / 9-bit UART: operates as asynchronous transmitter/receiver with 8 or 9-bit data bits and programmable baud rate.

19.1 UART Operation

The UARTn UTXn and URXn pins are shared with GPIO. In synchronous mode, the UTXn/URXn shared pins must set output high by software. In asynchronous mode (8-bit/9-bit UART), the UTXn shared pins must set output high and URXn set input high by software. Thus, URXn/UTXn pins will transfers to UART purpose. When UART disables, the UART pins returns to GPIO last status.

The UTXn/URXn pins also support open-drain structure. The open-drain option is controlled by PnOC bit. When PnOC=0, disable UTXn/URXn open-drain structure. When PnOC=1, enable UTXn/URXn open-drain structure. If enable open-drain structure, UTXn/URXn pin must set high level (IO mode control will be ignored) and need external pull-up resistor.

The UARTn supports interrupt function. EUnTX and EUnRX are UARTn transfer interrupt function control bits. UARTn transmitter and receiver interrupt function is controlled by EUnTX and EUnRX respectively. EUnTX=0/EUnRX=0, disable transmitter/receiver interrupt function. EUnTX=1/EUnRX=1, enable UART transmitter/receiver interrupt function. When UARTn interrupt function enables, the program counter points to interrupt vector to do UARTn interrupt service routine after UART operating. TIn/RIn is UARTn interrupt request flag, and also to be the UART operating status indicator when interrupt is disabled. TIn and RIn must clear by software.

UARTn provides four operating mode (one synchronous and three asynchronous) controlled by SnCON register. These modes can be support in different baud rate and communication protocols.

S0M0	S0M1	Mode	Synchronization	Clock Rate	Start Bit	Data Bits	Stop Bit	UART pins' mode and data
0	0	0	Synchronous	Fcpu/12	X	8	X	UTX0 pin: P11M=1 and P11=1 P07M=1 and P07=1 URX0 pin: Transmitter: P10M=1 and P10=1 P06M=1 and P06=1 Receiver: P10M=0 and P10=1 P06M=0 and P06=1 UTX1 pin: P16M=1 and P16=1 P12M=1 and P12=1 URX1 pin: Transmitter: P15M=1 and P15=1 P12M=1 and P12=1 Receiver: P15M=0 and P15=1 P12M=0 and P12=1
0	1	1	Asynchronous	Baud rate generator or T1 overflow rate	1	8	1	UTX/URX pin mode and data switch by hardware.
1	0	2	Asynchronous	Fcpu/64 or Fcpu/32	1	9	1	
1	1	3	Asynchronous	Baud rate generator or T1 overflow rate	1	9	1	

19.2 Mode 0: Synchronous 8-bit Receiver/Transmitter

Mode 0 is a shift register mode. It operates as synchronous transmitter/receiver. The UTXn pin output shift clock for both transmit and receive condition. The URXn pin is used to transmit and receive data. 8-bit data will be transmit and receive with LSB first. The baud rate is Fcpu/12. Data transmission is started by writing data to SnBUF register. In the end of the 8th bit transmission, the TIn flag is set. Data reception is controlled by RENn bit and clearing RIn bits. When RENn=1 and RIn is from 1 to 0, data transmission starts and the RIn flag is set at the end of the 8th bit reception.

* **Note:**

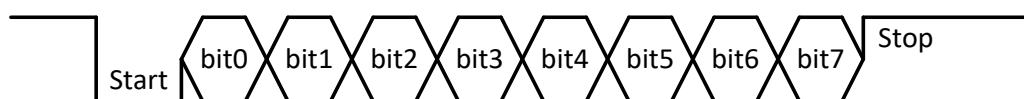
- 1. Only Mode 0, the URXn and UTXn respectively require input and output mode selection to receive/transmit data appropriately.**
- 2. Setting URXn and UTXn initially high because UART block drive the shared pin low signal only.**

19.3 Mode 1: 8-bit Receiver/Transmitter with Variable Baud Rate

Mode 1 supports an asynchronous 8-bit UART with variable baud rate. The transfer format includes 1 start bit, 8 data bits (LSB first) and 1 stop bit. Data is transmitted by UTXn pin and received by URXn pin. The baud rate clock source can be baud rate generator or T1 overflow controlled by BDn bit. When BDn=0, the baud rate clock source is from T1 overflow. When BDn=1, the baud rate clock source is from baud rate generator controlled by SnRELH and SnRELL. Additionally, the baud rate can be doubled by SMODn bit.

Data transmission is controlled by TENn bit. After transmission configuration, load transmitted data into SnBUF, and then UARTn starts to transmit the packet. The TIn flag is set at the beginning of the stop bit.

Data reception is controlled by RENn bit. When RENn=1, data reception function is enabled. Data reception starts by receiving the start bit for master terminal, URXn detects the falling edge of start bit, and then the RIn flag is set in the middle of a stop bit. Until reception completion, input data is stored in SnBUF register and the stop bit is stored in RB8n.



19.4 Mode 2: 9-bit Receiver/Transmitter with Fixed Baud Rate

Mode 2 supports an asynchronous 9-bit UART with fixed baud rate. The transfer format includes 1 start bit, 9 data bits (LSB first) and 1 stop bit. Data is transmitted by UTXn pin and received by URXn pin. The baud rate clock source is fixed to Fcpu/64 or Fcpu/32 and is controlled by SMODn bit. When SMODn=0, baud rate is Fcpu/64. When SMODn=1, baud rate is Fcpu/32.

Data transmission is controlled by TENn bit. After transmission configuration, load transmitted data into SnBUF, and then UARTn starts to transmit the packet. The 9th data bit is taken from TB8n. The

TIn flag is set at the beginning of the stop bit.

Data reception is controlled by RENn bit. When RENn=1, data reception function is enabled. Data reception starts by receiving the start bit for master terminal, URXn detects the falling edge of start bit, and then the RIn flag is set in the middle of a stop bit. Until reception completion, lower 8-bit input data is stored in SnBUF register and the 9th bit is stored in RB8n.



19.5 Mode 3: 9-bit Receiver/Transmitter with Variable Baud Rate

Mode 3 supports an asynchronous 9-bit UART with variable baud rate. The transfer format includes 1 start bit, 9 data bits (LSB first) and 1 stop bit. Data is transmitted by UTXn pin and received by URXn pin. The different between Mode 2 and Mode 3 is baud rate selection. In the Mode 3, the baud rate clock source can be baud rate generator or T1 overflow controlled by BDn bit. When BDn=0, the baud rate clock source is from T1 overflow. When BDn=1, the baud rate clock source is from baud rate generator controlled by SnRELH and SnRELL. Additionally, the baud rate can be doubled by SMODn bit.

Data transmission is controlled by TENn bit. After transmission configuration, load transmitted data into SnBUF, and then UART starts to transmit the packet. The 9th data bit is taken from TB8n. The TIn flag is set at the beginning of the stop bit.

Data reception is controlled by RENn bit. When RENn=1, data reception function is enabled. Data reception starts by receiving the start bit for master terminal, URXn detects the falling edge of start bit, and then the RIn flag is set in the middle of a stop bit. Until reception completion, lower 8-bit input data is stored in SnBUF register and the 9th bit is stored in RB8n.



19.6 Multiprocessor Communication

UART supports multiprocessor communication between a master device and one or more slave device in Mode 2 and Mode 3 (9-bit UART). The master identifies correct slaves by using the 9th data bit. When the communication starts, the master transmits a specific address byte with the 9th bit is

set “1” to selected slaves, and then transmits a data byte with the 9th bit is set “0” in the following transmission.

Multiprocessor communication is controlled by SnM20 bit. When SnM20=0, disable multiprocessor communication. When SnM20=1, enable multiprocessor communication. If SnM20 is set, the UARTn reception interrupt is only generated when the 9th received bit is “1” (RB8n). The slaves will compare received data with its own address data by software. If address byte is match, the slaves clear SnM20 bit to enable interrupt function in the following data transmission. The slaves with unmatched address, their SnM20 keep in “1” and will not generate interrupt in the following data transmission.

19.7 Baud Rate Control

The UARTn mode 0 has a fixed baud rate at Fcpu/12, and the mode 2 has two baud rate selection which is chosen by SMODn bit: Fcpu/64 (SMODn = 0) and Fcpu/32 (SMODn = 1).

The baud rate of UARTn mode 1 and mode 3 is generated by either SnRELH/SnRELL registers (BDn = 1) or Timer 1 overflow period (BDn = 0). The SMODn bit doubles the frequency from the generator.

If the SnRELH/SnRELL is selected (BDn = 1) in mode 1 and 3, the baud rate is generated as following equation.

$$\text{Baud Rate} = 2^{\text{SMODn}} \times \frac{\text{Fcpu}}{(64 - 8 \times \text{CDn}) \times (1024 - \text{SnREL})} \text{ bps}$$

Table 19-1 Recommended Setting for Common UARTn Baud Rates (Fcpu = 16 MHz, CDn=0)

Baud Rate	SMODn	CDn	SnRELH	SnRELL	Accuracy
4800	0	0	0x03	0xCC	0.16 %
9600	0	0	0x03	0xE6	0.16 %
19200	0	0	0x03	0xF3	0.16 %
38400	1	0	0x03	0xF3	0.16 %
56000	1	0	0x03	0xF7	-0.79 %
57600	1	1	0x03	0xF6	-0.79 %
115200	1	1	0x03	0xFB	-0.79 %
128000	1	0	0x03	0xFC	-2.34 %
250000	1	0	0x03	0xFE	0.00 %
500000	1	0	0x03	0xFF	0.00 %

If the Timer 1 overflow period is selected (BDn = 0) in mode 1 and 3, the baud rate is generated as following equation. The Timer 1 must be in 8-bit auto-reload mode which can generate periodically overflow signals.

$$\text{Baud Rate} = 2^{\text{SMODn}} \times \frac{1}{(32 - 4 \times \text{CDn}) \times \text{Timer1 period}} \text{ bps}$$

Table 19-2 Recommended Setting T1 overflow period (T1 clock=32M) for Common UART0 Baud Rates (Fcpu = 16 MHz)

Baud Rate	SMOD0	CD	Timer Period	T1R	Accuracy
4800	0	0	6.510 us	0xFF30	0.16 %
9600	1	0	6.510 us	0xFF30	0.16 %
19200	1	0	3.255 us	0xFF98	0.16 %
38400	1	0	1.628 us	0xFFCC	0.16 %
56000	1	0	1.116 us	0xFFDC	-0.79 %
57600	1	0	1.085 us	0xFFDD	-0.79 %
115200	1	1	0.543 us	0xFFEC	-0.79 %
128000	1	1	0.488 us	0xFFEE	-0.79 %
250000	1	0	0.250 us	0xFFFF8	0.00 %

*** Note:**

1. When baud rate generator source is T1 overflow rate, the max counter value is 0xFFFFB. (Only supports 0x0000~0xFFFFB).

2. When baud rate generator source is T1 overflow rate, the system clock Fcpu must be greater four times to T1 overflow rate.

19.8 Power Saving

The UARTn module has clock gating function for saving power. When RENn bit is 0, the UARTn module internal clocks are halted to reduce power consumption. Conversely, when RENn bit is 1, UARTn internal clocks are run, and registers can access.

19.9 UART0 Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SMODMX	-	-	-	-	UR1MX1	UR1MX0	-	UR0MX
SMODCAL	-	-	SMOD1	SMOD0	-	-	CD1	CD0
SOCON2	-	-	BD1	BD0	-	-	TEN1	TEN0
SOCON	SOM0	SOM1	SOM20	REN0	TB80	RB80	Ti0	RI0
S0BUF	S0BUF7	S0BUF6	S0BUF5	S0BUF4	S0BUF3	S0BUF2	S0BUF1	S0BUF0
SORELH	-	-	-	-	-	-	SOREL9	SOREL8
SORELL	SOREL7	SOREL6	SOREL5	SOREL4	SOREL3	SOREL2	SOREL1	SOREL0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
IEN1	-	-	EU1RX	EU1TX	EUORX	EUOTX	-	EI2C
ODCON	-	-	-	-	-	-	URT1OD	URT0OD

SMODMX Register (0x94)

Bit	Field	Type	Initial	Description
0	UR0MX	R/W	00	UART0 IO select 0: UTX0 is P11 and URX0 is P10 1: UTX0 ⁽¹⁾ is P07 and URX0 ⁽¹⁾ is P06
Else				Refer to other chapter(s)

SMODCAL Register (0x95)

Bit	Field	Type	Initial	Description
4	SMOD0	R/W	0	UART0 baud rate control In UART mode 0: Unused In UART mode 1, 3: The baud rate generated as the equation in section 18.7 (Baud Rate Control). In UART mode 2: 0: Fcpu/64 1: Fcpu/32
0	CD0	R/W	0	UART0 clock divider 0: 16-clocks for a cycle 1: 14-clocks for a cycle
Else				Refer to other chapter(s)

SOCON2 Register (0x96)

Bit	Field	Type	Initial	Description
4	BD0	R/W	0	UART0 Baud rate generators selection (mode 1, 3)

				0: Timer 1 overflow period 1: Controlled by SORELH and SORELL registers
0	TEN0	R/W	0	UART0 TX enable bit 0: Disable 1: Enable
Else				Refer to other chapter(s)

SOCON Register (0x97)

Bit	Field	Type	Initial	Description
7..6	SOM[1:0]	R/W	00	UART0 mode selection 00: Mode 0 01: Mode 1 10: Mode 2 11: Mode 3
5	SOM20	R/W	0	Multiprocessor communication (mode 2, 3) 0: Disable 1: Enable
4	RENO	R/W	0	UART0 module (and reception function) 0: Disable for power saving* 1: Enable for UART operating
3	TB80	R/W	0	The 9 th bit transmission data (mode 2, 3)
2	RB80	R/W	0	The 9 th bit data from reception
1	TIO	R/W	0	UART0 interrupt flag of transmission
0	RIO	R/W	0	UART0 interrupt flag of reception

*** Note: TIO and RIO are clear by software when interrupt is enabled.**

SOBUF Register (0x99)

Bit	Field	Type	Initial	Description
7..0	SOBUF[7:0]	R/W	0x00	Action of writing data triggers UART0 communication (LSB first). Reception data is available to read by the end of packages.

SORELH/SORELL Registers (SORELH: 0x9B, SORELL: 0x9A)

Bit	Field	Type	Initial	Description
15..10	Reserved	R	0x00	
9..0	SOREL[9:0]	R/W	0x3D9	SORELH[1:0] & SORELL[7:0].

UART0 Reload Register used for UART0 baud rate generation.

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
Else				Refer to other chapter(s)

IEN1 Register (0xB8)

Bit	Field	Type	Initial	Description
3	EUORX	R/W	0	UART0 receiver interrupt enable
2	EUOTX	R/W	0	UART0 transmitter interrupt enable
Else				Refer to other chapter(s)

ODCON Register (0xF018)

Bit	Field	Type	Initial	Description
0	UARTOOD	R/W	0	UART0 TX/RX open-drain control bit. 0: Disable, Switch UTX0 and URX0 to push-pull mode 1: Enable, Switch UTX0 and URX0 to open-drain mode
Else				Refer to other chapter(s)

19.10 UART1 Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SMODMX	-	-	-	-	UR1MX1	UR1MX0	-	UR0MX
SMODCAL	-	-	SMOD1	SMOD0	-	-	CD1	CD0
S0CON2	-	-	BD1	BD0	-	-	TEN1	TEN0
S1CON	S1M0	S1M1	S1M20	REN1	TB81	RB81	TI1	RI1
S1BUF	S1BUF7	S1BUF6	S1BUF5	S1BUF4	S1BUF3	S1BUF2	S1BUF1	S1BUF0
S1RELH	-	-	-	-	-	-	S1REL9	S1REL8
S1RELL	S1REL7	S1REL6	S1REL5	S1REL4	S1REL3	S1REL2	S1REL1	S1RELO
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
IEN1	-	-	EU1RX	EU1TX	EUORX	EUOTX	-	EI2C
ODCON	-	-	-	-	-	-	URT1OD	URT0OD

SMODMX Register (0x94)

Bit	Field	Type	Initial	Description
3..2	UR1MX[1:0]	R/W	00	UART1 IO select 00: UTX1 is P16 and URX1 is P15. 01: Reserved. 10: UTX1 ⁽¹⁾ is P12. 11: URX1 ⁽¹⁾ is P12.
Else				Refer to other chapter(s)

SMODCAL Register (0x95)

Bit	Field	Type	Initial	Description
5	SMOD1	R/W	0	UART1 baud rate control. In UART mode 0: Unused. In UART mode 1, 3: The baud rate generated as the equation in section 18.7 (Baud Rate Control). In UART mode 2: 0: Fcpu/64 1: Fcpu/32
1	CD1	R/W	0	UART1 clock divider 0: 16-clocks for a cycle 1: 14-clocks for a cycle
Else				Refer to other chapter(s)

S0CON2 Register (0x96)

Bit	Field	Type	Initial	Description
5	BD1	R/W	0	UART1 Baud rate generators selection (mode 1, 3) 0: Timer 1 overflow period 1: Controlled by S1RELH and S1RELL registers
1	TEN1	R/W	0	UART1 TX enable bit 0: Disable 1: Enable
Else				Refer to other chapter(s)

S1CON Register (0x9C)

Bit	Field	Type	Initial	Description
7..6	S1M[0:1]	R/W	00	UART1 mode selection 00: Mode 0 01: Mode 1

				10: Mode 2
				11: Mode 3
5	S1M20	R/W	0	Multiprocessor communication (mode 2, 3) 0: Disable 1: Enable
4	REN1	R/W	0	UART1 module (and reception function) 0: Disable for power saving* 1: Enable for UART operating
3	TB81	R/W	0	The 9 th bit transmission data (mode 2, 3)
2	RB81	R/W	0	The 9 th bit data from reception
1	TI1	R/W	0	UART1 interrupt flag of transmission
0	RI1	R/W	0	UART1 interrupt flag of reception

*** Note: TI1 and RI1 are clear by software when interrupt is enabled.**

S1BUF Register (0x9D)

Bit	Field	Type	Initial	Description
7..0	S1BUF[7:0]	R/W	0x00	Action of writing data triggers UART1 communication (LSB first). Reception data is available to read by the end of packages.

S1RELH/SRRELL Registers (S1RELH: 0x9F, S1RELL: 0x9E)

Bit	Field	Type	Initial	Description
15..10	Reserved	R	0x0	
9..0	S1REL[9:0]	R/W	0x3D9	S1RELH[1:0] & S1RELL[7:0]. UART1 Reload Register used for UART1 baud rate generation.

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
Else				Refer to other chapter(s)

IEN1 Register (0xB8)

Bit	Field	Type	Initial	Description
5	EU1RX	R/W	0	UART1 receiver interrupt enable
4	EU1TX	R/W	0	UART1 transmitter interrupt enable

Else	Refer to other chapter(s)
------	---------------------------

ODCON Register (0xF018)

Bit	Field	Type	Initial	Description
1	UART1OD	R/W	0	UART1 TX/RX open-drain control bit. 0: Disable, Switch UTX1 and URX1 to push-pull mode 1: Enable, Switch UTX1 and URX1 to open-drain mode
Else	Refer to other chapter(s)			

20 I2C

The I2C is a serial communication interface for data exchanging from one MCU to one MCU or other hardware peripherals. The device can transmit data as a master or a slave with two bi-directional IO, SDA (Serial data line) and SCL (Serial clock line).

When a master transmit data to a slave, it's called "WRITE" operation; when a slave transmit data to a master, it's called "READ" operation. It also supports multi-master communication and keeps data transmission correctly by an arbitration method to decide one master has the control on bus and transmit its data.

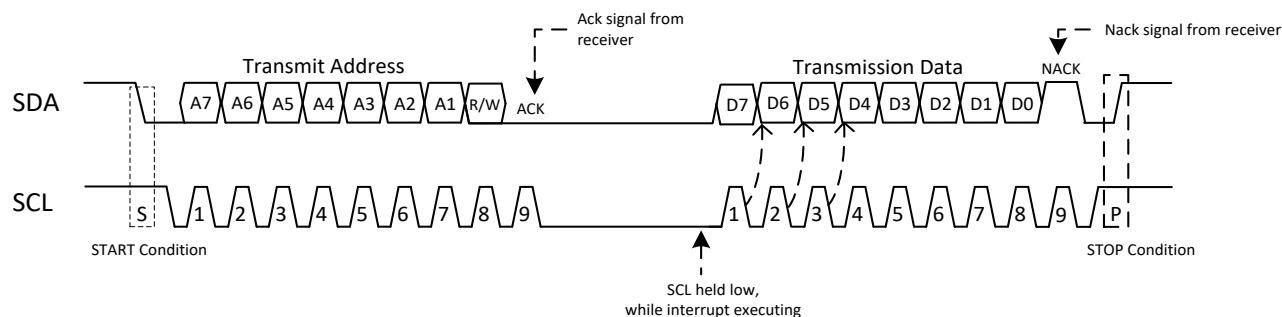
- Master Tx, Rx Mode
- Slave Tx, Rx mode (with general address call) for multiplex slave in single master situation.
- 2-wire synchronous data transfer/receiver.
- Support 100K/400K clock rate.

20.1 I2C Protocol

I2C transmission structure includes a START(S) condition, 8-bit address byte, one or more data byte and a STOP (P) condition. START condition is generated by master to initial any transmission.

Data is transmitted with the Most Significant Bit (MSB) first. In address byte, the higher 7-bit is address bit and the lowest bit is data direction (R/W) bit. When R/W=0, it assigns a "WRITE" operation. When R/W=1, it assigns a "READ" operation.

After each byte is received, the receiver (a master or a slave) must send an acknowledge (ACK). If transmitter can't receive an ACK, it will recognize a not acknowledge (NACK). In WRITE operation, the master will transmit data to the slave and then waits for ACK from slave. In READ operation, the slave will transmit data to the master and then waits for ACK from master. In the end, the master will generate a STOP condition to finish transmission.

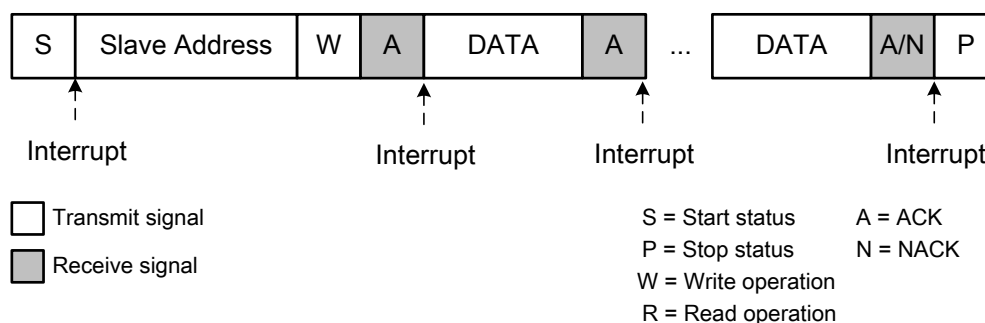


20.2 I2C Transfer Modes

The I2C can operate as a master/slave to execute the 8-bit serial data transmission/reception operation. Thus, the module can operate in one of four modes: Master Transmitter, Master Receiver, Slave Transmitter and Slave Receiver.

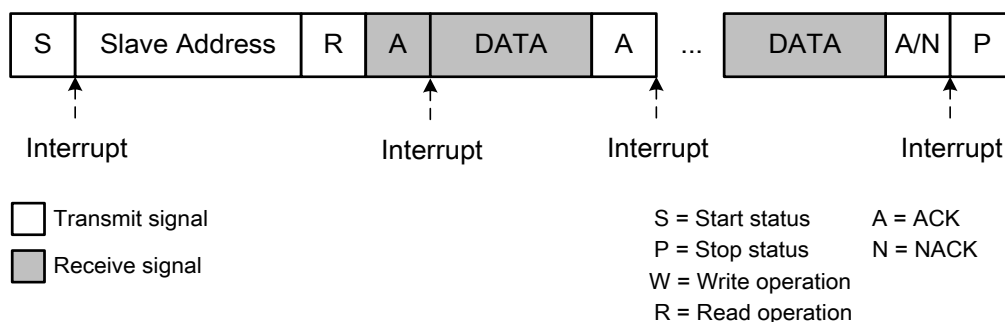
20.2.1 Master Transmitter Mode

The master transmits information to the slave. The serial data is output via SDA while the serial clock is output on SCL. Data transmission starts via generate a START(S) signal. After the START signal, the specific address byte of slave device is sent. The address byte includes 7-bit address bit and an 8th data direction (R/W) bit. The R/W is set "0" to enable the master transmission. In the following, the master transmits one or more data byte to the slave. After each data is transmitted, the master waits for the acknowledge (ACK) from the slave. In the end, the master generates a STOP (P) signal to terminate the data transmission.



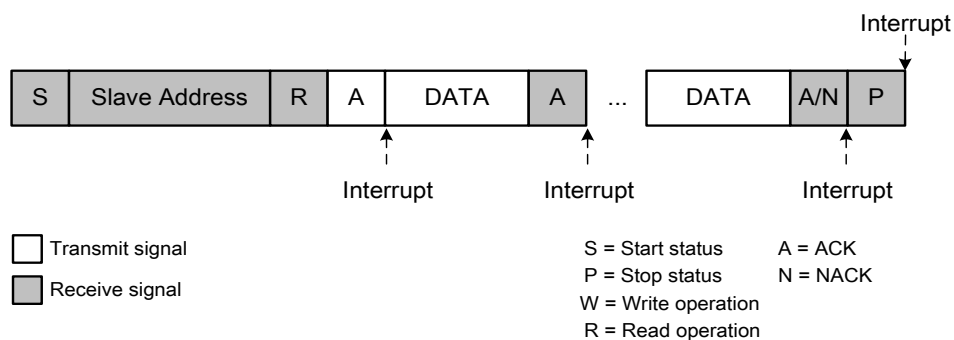
20.2.2 Master Receiver Mode

The master receives the information from the slave. The serial data input via SDA while the serial clock output on SCL. Data reception starts via generate a START(S) signal. After the START signal, the specific address byte of slave device is sent. The address byte includes 7-bit address bit and an 8th data direction (R/W) bit. The R/W is set "1" to enable the master reception. In the following, the master receives one or more data byte from the slave. After each data is received, the master generates the acknowledge (ACK) or not acknowledge (NACK) to the slave via the status of AA bit. In the end, the master generates a STOP (P) signal to terminate the data transmission.



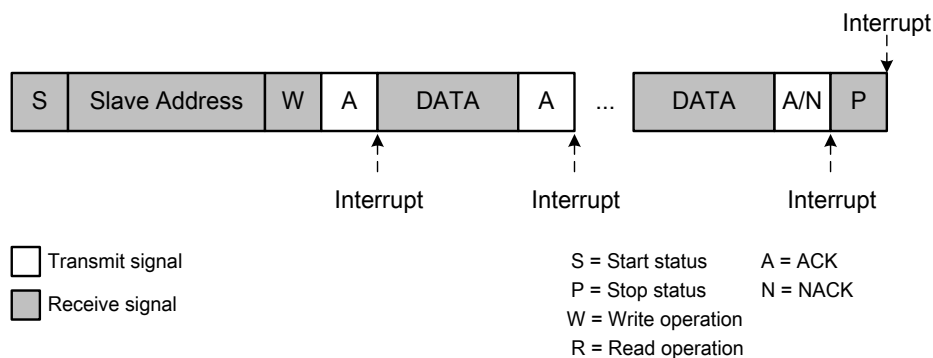
20.2.3 Slave Transmitter Mode

The slave transmits information to the master. The serial data output via SDA while the serial clock input on SCL. Data transmission starts via receive a START(S) signal from the master. After the START signal, the specific address byte of slave device is received. The address byte includes 7-bit address bit and an 8th data direction (R/W) bit. The R/W is set "1" to enable the slave transmission. If the received address byte match the address in I2CADDR register, the slave generate an acknowledge (ACK). Otherwise, if general call address condition is set (GC=1), the slave also generate an acknowledge (ACK) after general call address (0x00) is received. In the following, the slave transmits one or more data byte to the master. After each data is transmitted, the slave waits for the acknowledge (ACK) from the master. In the end, the slave receives a STOP (P) signal from the master to terminate the data transmission.



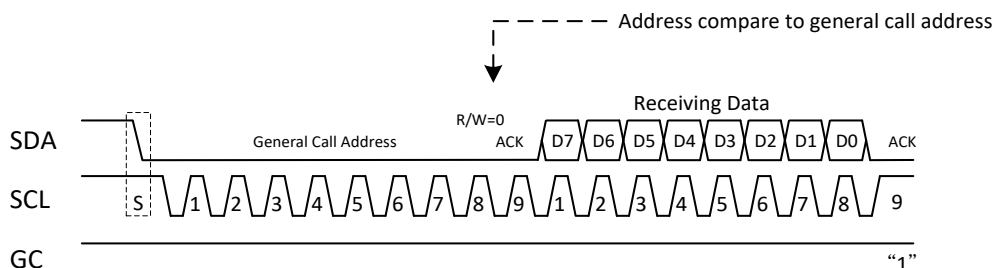
20.2.4 Slave Receiver Mode

The slave receives information from the master. Both the serial data and the serial clock are input on SDA and SCL. Data reception starts via receive a START(S) signal from the master. After the START signal, the specific address byte of slave device is received. The address byte includes 7-bit address bit and an 8th data direction (R/W) bit. The R/W is set "0" to enable the slave reception. If the received address byte match the address in I2CADDR register, the slave generate an acknowledge (ACK). Otherwise, if general call address condition is set (GC=1), the slave also generate an acknowledge (ACK) after general call address (0x00) is received. In the following, the slave receives one or more data byte from the master. After each data is receives, the slave generates the acknowledge (ACK) or not acknowledge (NACK) to the master via the status of AA bit. In the end, the slave receives a STOP (P) signal from the master to terminate the data transmission.



20.3 General Call Address

In I2C bus, the first 7-bit is the slave address. Only the address matches slave address, the slave will response an ACK. The exception is the general call address which can address all slave devices. When this address occur, all devices should response an acknowledge (ACK). The general call address is a special address which is reserved as all "0" of 7-bit address. The general call address function is control by GC bit. Set this bit will enable general call address and clear it will disable. When GC=1, the general call address will be recognized. When GC=0, the general call address will be ignored.



20.4 Serial Clock Generator

In master mode, the SCL clock rate generator's is controlled by CR[2:0] bit of I2CCON register.

When CR[2:0]=000~011, SCL clock rate is from internal clock generator.

$$\text{SCL Clock Rate} = \frac{F_{\text{hosc}}}{\text{Prescaler}} (\text{Prescaler} = 320 \sim 40)$$

When CR[2:0]=1XX, SCL clock rate is from Timer 1 overflow rate.

$$\text{SCL Clock Rate} = \frac{\text{Timer 1 Overflow}}{8}$$

The table below shows the clock rate under different setting.

CR2	CR1	CR0	Fhosc=4M	Fhosc=8M	Fhosc=16M	Fhosc=32M	Clock Divider
0	0	0	100	200	400	-	40
0	0	1	50	100	200	400	80
0	1	0	25	50	100	200	160
0	1	1	12.5	25	50	100	320
1	X	X	(Timer 1 overflow rate)/8				

*** Note:**

1. When I2C function is enabled, SDA/SCL pins switch to I2C pins' mode by hardware.

2. When clock generator source is T1 overflow rate, the max counter value is 0xFFF9. (Only supports 0x0000~0xFFF9).

20.5 Synchronization and Arbitration

In multi-master condition, more than one master may transmit on bus in the same time. It must be decided which master has the control of bus and complete its transmission. Clock synchronization and arbitration are used to configure multi-master transmission. Clock synchronization is executed by synchronizing the SCL signal with another devices.

When two masters want to transmit data in the same, the clock synchronization will start by the High to Low transition on the SCL. If master 1 clock set LOW first, it holds the SCL in LOW status until the clock transit to HIGH status. However, if another master clock still keep LOW status, the Low to High transition of master 1 may not change SCL status (SCL keep LOW). In the other word, SCL keep LOW by the master with the longest clock time in LOW status. The SCL will transit from LOW to HIGH when the all devices clock transit to HIGH status. In the duration, the master1 will keep in HIGH status and wait for SCL transition (from LOW to HIGH), then continue its transmission. After clock synchronization, all devices clock and SCL clock are the same. Arbitration is used to decide which master can complete its transmission by SDA signal. Two masters may send out a START condition and transmit data on bus in the same time. They may influence by each other. Arbitration will force one master to lose the control on bus. Data transmission will keep until master output different data signal. If one master transmits HIGH status and another master transmits LOW status, the SDA will be pull low. The master output High will detect the different with SDA and lose the control on bus. The master with LOW status wins the bus control and continues its transmission. There is no data miss during arbitration.

20.6 System Management Bus Extension

The optional System Management Bus (SMBus) protocol hardware supports 3 types timeout detection: (1) Tmext Timeout Detection: The cumulative stretch clock cycles within one byte. (2)Tsext Timeout Detection: The cumulative stretch clock cycles between start and stop condition. (3)Timeout Detection: The clock low measurement.

Timeout detection is controlled by SMBSEL and SMBDST registers. The SMBEXE bit of SMBSEL is SMBus extension function enable bit. When SMBEXE=1, SMBus extension function is enabled. Otherwise, Disable SMBus extension function. Timeout type and period setting is controlled by SMBTOP[2:0] and SMBDST. The period of SMBus timeout is controlled by three 16-bit buffers of

Tmext, Tsext and Tout. The equation is as following.

$$T_{mext}/T_{sext}/T_{out} = \frac{\text{Timeout Period(sec)} \times F_{cpu}(\text{Hz})}{1024}$$

Tmext is support by two 8-bit register of Tmext_L and Tmext_H . Tmext_L hold the low byte and Tmext_H hold high byte. Tsext is support by two 8-bit register of Tsext_L and Tsext_H . Tsext_L hold the low byte and Tsext_H hold high byte. Tout is support by two 8-bit register of Tout_L and Tout_H . Tout_L hold the low byte and Tout_H hold high byte.

Type	Time out period	Fhosc=32MHz	
		DEC	HEX
Tmext	5ms	156	9C
Tsext	25ms	781	30D
Tout	35ms	1094	446

By the setting of SMBTOP[2:0] to choose register type (as the table below), and write to register by write data to SMBDST register.

SMBTOP[2:0]	SMBDST	Description
000	Tmext_L	Select the low byte of Tmext register.
001	Tmext_H	Select the high byte of Tmext register.
010	Tsext_L	Select the low byte of Tsext register.
011	Tsext_H	Select the high byte of Tsext register.
100	Tout_L	Select the low byte of Tout register.
101	Tout_H	Select the high byte of Tout register.

When the SMBus extension function is enabled the lower 3-bit of I2CSTA hold the information about time out as the table below.

I2CSTA	Description
XXXX X000	No timeout errors.
XXXX XXX1	Tout timeout error.
XXXX XX1X	Tsxt timeout error.
XXXX X1XX	Tmext timeout error.

20.7 Power Saving

The I2C module has clock gating function for saving power. When ENS1 bit is 0, the I2C module internal clocks are halted to reduce power consumption. Conversely, when ENS1 bit is 1, I2C internal clocks are run, and registers can access. The ENS1 bit must be set to 1, before the initial setting I2C.

20.8 I2C Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
I2CDAT	I2CDAT7	I2CDAT6	I2CDAT5	I2CDAT4	I2CDAT3	I2CDAT2	I2CDAT1	I2CDAT0
I2CADR	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	GC
I2CCON	CR2	ENS1	STA	STO	SI	AA	CR1	CR0
I2CSTA	I2CSTA7	I2CSTA6	I2CSTA5	I2CSTA4	I2CSTA3	I2CSTA2	I2CSTA1	I2CSTA0
SMBSEL	SMBEXE	-	-	-	I2CMX	SMBTOP2	SMBTOP1	SMBTOP0
SMBDST	SMBD7	SMBD6	SMBD5	SMBD4	SMBD3	SMBD2	SMBD1	SMBD0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
IEN1	-	-	EU1RX	EU1TX	EU0RX	EU0TX	-	EI2C

I2CDAT Register (0xDA)

Bit	Field	Type	Initial	Description
7:0	I2CDAT[7:0]	R/W	0x00	The I2CDAT register contains a byte to be transmitted through I2C bus or a byte which has just been received through I2C bus. The CPU can read from and write to this 8-bit, directly addressable SFR while it is not in the process of byte shifting. The I2CDAT register is not shadowed or double buffered so the user should only read I2CDAT when an I2C interrupt occurs.

I2CADR Register (0xDB)

Bit	Field	Type	Initial	Description
7:1	I2CADR[6:0]	R/W	0x00	I2C slave address
0	GC	R/W	0	General call address (0X00) acknowledgment 0: ignored 1: recognized

I2CCON Register (0xDC)

Bit	Field	Type	Initial	Description
7,1,0	CR[2:0]	R/W	0	I2C clock rate 000: Fhosc/40 001: Fhosc /80 010: Fhosc /160 011: Fhosc /320 1XX: Timer 1 overflow-period/8
6	ENS1	R/W	0	I2C functionality 0: Disable for power saving* 1: Enable for I2C operating
5	STA	R/W	0	START flag 0: No START condition is transmitted. 1: A START condition is transmitted if the bus is free.
4	STO	R/W	0	STOP flag 0: No STOP condition is transmitted. 1: A STOP condition is transmitted to the I2C bus in master mode.
3	SI	R/W	0	Serial interrupt flag The SI is set by hardware when one of 25 out of 26 possible I2C states is entered. The only state that does not set the SI is state F8h, which indicates that no relevant state information is available. The SI flag must be cleared by software. In order to clear the SI bit, '0' must be written to this bit. Writing a '1' to SI bit does not change value of the SI.
2	AA	R/W	0	Assert acknowledge flag 0: A NACK will be returned when a byte has received 1: An ACK will be returned when a byte has received

* When ENS1 bit is 0, I2C relevant register are unable to access, and the module internal clocks are halted.

I2CSTA Register (0xDD)

Bit	Field	Type	Initial	Description
7:3	I2CSTA[7:3]	R	11111	I2C Status Code
2..0	I2CSTA[2:0]	R	000	SMBus Status Code

I2C status code and status

Mode	Status Code	Status of the I2C	Application software response				Next action taken by I2C hardware	
			To/from I2CDAT	TO I2CCON				
				STA	STO	SI	AA	
Master Transmitter/Receiver	08H	A START condition has been transmitted	Load SLA+R/W	X	0	0	X	SLA+R/W will be transmitted; ACK will be received
	10H	A repeated START condition has been transmitted.	Load SLA+R/W	X	0	0	X	SLA+R/W will be transmitted; ACK will be received
			Load SLA+R/W					SLA+W will be transmitted; I2C will be switched to MST/TRX mode.
Master Transmitter	18H	SLA+W has been transmitted; ACK has been received	Load data byte	0	0	0	X	Data byte will be transmitted; ACK will be received.
			No action	1	0	0	X	Repeated START will be transmitted.
			No action	0	1	0	X	STOP condition will be transmitted; STO flag will be reset.
			No action	1	1	0	X	STOP condition followed by a START condition will be transmitted; STO flag will be reset.
	20H	SLA+W has been transmitted; not ACK has been received	Load data byte*	0	0	0	X	Data byte will be transmitted; ACK will be received.
			No action	1	0	0	X	Repeated START will be transmitted.
			No action	0	1	0	X	STOP condition will be transmitted; STO flag will be reset.
			No action	1	1	0	X	STOP condition followed by a START condition will be transmitted; STO flag will be reset.
	28H	Data byte in I2CDAT has been transmitted; ACK has been received	Load data byte	0	0	0	X	Data byte will be transmitted; ACK bit will be received.
			No action	1	0	0	X	Repeated START will be transmitted.
			No action	0	1	0	X	STOP condition will be transmitted; STO flag will be reset.
			No action	1	1	0	X	STOP condition followed by a START condition will be transmitted; STO flag will be reset.
	30H	Data byte in I2CDAT has been transmitted; not ACK has been received	Load data byte*	0	0	0	X	Data byte will be transmitted; ACK will be received.
			No action	1	0	0	X	Repeated START will be transmitted.
			No action	0	1	0	X	STOP condition will be transmitted; STO flag will be reset.
			No action	1	1	0	X	STOP condition followed by a START condition will be transmitted; STO flag will be reset.
Master Receiver	40H	SLA+R has been transmitted; ACK has been received	No action	0	0	0	0	Data byte will be received; not ACK will be returned
			No action	0	0	0	1	Data byte will be received; ACK will be returned
	48H	SLA+R has been transmitted; not ACK has been received	No action	1	0	0	X	Repeated START condition will be transmitted
			No action	0	1	0	X	STOP condition will be transmitted; STO flag will be reset
			No action	1	1	0	X	STOP condition followed by a START condition will be transmitted; STO flag will be reset
			No action	1	1	0	X	STOP condition followed by a START condition will be transmitted; STO flag will be reset
	50H	Data byte has been received; ACK has been returned	Read data byte	0	0	0	0	Data byte will be received; not ACK will be returned
			Read data byte	0	0	0	1	Data byte will be received; ACK will be returned
58H	Data byte has been received; not ACK has been returned	Read data byte	1	0	0	X	Repeated START condition will be transmitted	
		Read data byte	0	1	0	X	STOP condition will be transmitted; STO flag will be reset	
		Read data byte	1	1	0	X	STOP condition followed by a START condition will be transmitted; STO flag will be reset	

Mode	Status Code	Status of the I2C	Application software response				Next action taken by I2C hardware	
			To/from I2CDAT	TO I2CCON				
STA	STO	SI		AA				
Slave Receiver	60H	Own SLA+W has been received; ACK has been returned	No action	X	0	0	0/1	Data byte will be received and not ACK/ACK will be returned
	68H	Arbitration lost in SLA+R/W as master; own SLA+W has been received, ACK returned	No action	X	0	0	0/1	Data byte will be received and not ACK/ACK will be returned
	70H	General call address (00H) has been received; ACK has been returned	No action	X	0	0	0/1	Data byte will be received and not ACK/ACK will be returned
	78H	Arbitration lost in SLA+R/W as master; general call address has been received, ACK returned	No action	X	0	0	0/1	Data byte will be received and not ACK/ACK will be returned
	80H	Previously addressed with own SLV address; DATA has been received; ACK returned	Read data byte	X	0	0	0/1	Data byte will be received and not ACK/ACK will be returned
	88H	Previously addressed with own SLA; DATA byte has been received; not ACK returned	Read data byte	0	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address
			Read data byte	0	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized
			Read data byte	1	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address; START condition will be transmitted when the bus becomes free

Slave Transmitter			Read data byte	1	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized; START condition will be transmitted when the bus becomes free
	90H	Previously addressed with general call address; DATA has been received; ACK returned	Read data byte	X	0	0	0/1	Data byte will be received and not ACK/ACK will be returned
	98H	Previously addressed with general call address; DATA has been received; not ACK returned	Read data byte	0	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address
			Read data byte	0	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized
			Read data byte	1	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address; START condition will be transmitted when the bus becomes free
			Read data byte	1	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized; START condition will be transmitted when the bus becomes free
	A0H	A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX	No action	0	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address
			No action	0	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized
			No action	1	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address; START condition will be transmitted when the bus becomes free
			No action	1	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized; START condition will be transmitted when the bus becomes free
	A8H	Own SLA+R has been received; ACK has been returned	Load data byte	X	0	0	0	Last data byte will be transmitted and ACK will be received
			Load data byte	X	0	0	1	Data byte will be transmitted; ACK will be received.
	B0H	Arbitration lost in SLA+R/W as master; own SLA+R has been received, ACK has been returned.	Load data byte	X	0	0	0	Last data byte will be transmitted and ACK will be received
			Load data byte	X	0	0	1	Data byte will be transmitted; ACK will be received.
	B8H	Data byte has been transmitted; ACK will be received.	Load data byte	X	0	0	0	Last data byte will be transmitted and ACK will be received
			Load data byte	X	0	0	1	Data byte will be transmitted; ACK will be received.
	C0H	Data byte has been transmitted; not ACK has been received.	No action	0	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address.
			No action	0	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized.
			No action	1	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address; START condition will be transmitted when the bus becomes free.
			No action	1	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized; START condition will be transmitted when the bus becomes free.
	C8H	Last data byte has been transmitted; ACK has been received.	No action	0	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address.
			No action	0	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized.
			No action	1	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or general call address; START condition will be transmitted when the bus becomes free.
			No action	1	0	0	1	Switched to not addressed SLV mode; own SLA or general call address will be recognized; START condition will be transmitted when the bus becomes free.
Miscellaneous	F8H	No relevant state information available; SI=0	No action	No action				Wait or proceed current transfer
	38H	Arbitration lost	No action	0	0	0	X	I2C will be released; A start condition will be transmitted.
			No action	1	0	0	X	When the bus becomes free. (enter to a master mode)
	00H	Bus error during MST or selected slave modes	No action	0	1	0	X	Only the internal hardware is affected in the MST or addressed SLV modes. In all cases, the bus is released and I2C is switched to the not addressed SLV mode. STO flag is reset.

“SLA” means slave address, “R” means R/W=1, “W” means R/W=0

*For applications where NACK doesn't mean the end of communication.

SMBSEL Register (0xDE)

Bit	Field	Type	Initial	Description
7	SMBEXE	R/W	0	SMBus extension functionality 0: Disable 1: Enable
Else	Reserved	R/W	0	
3	I2CMX	R/W	0	I2C IO select bit. 0: I2C SCL is P0.6/ SDA is P0.7. 1: I2C SCL ⁽¹⁾ is P1.3/ SDA ⁽¹⁾ is P1.4.
2..0	SMBTOP[2:0]	R/W	000	SMBus timeout register

SMBDST Register (0xDF)

Bit	Field	Type	Initial	Description
7..0	SMBD[7:0]	R/W	0x00	This register is used to provide a read/write access port to the SMBus timeout registers. Data read or written to that register is actually read or written to the Timeout Register which is pointed by the SMBSEL register.

IEN0 Register (0xA8)

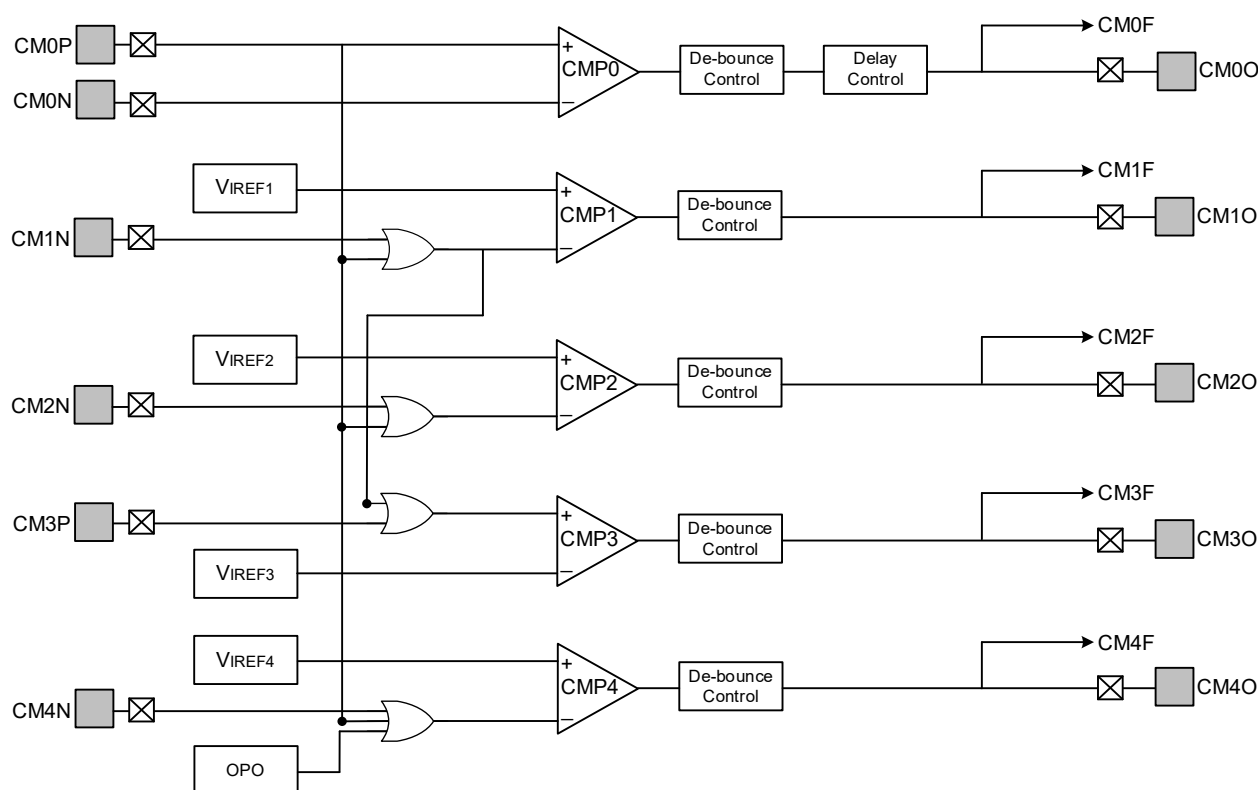
Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
Else				Refer to other chapter(s)

IEN1 Register (0xB8)

Bit	Field	Type	Initial	Description
0	EI2C	R/W	0	Interrupts enable. Refer to Chapter Interrupt
Else				Refer to other chapter(s)

21 Comparator

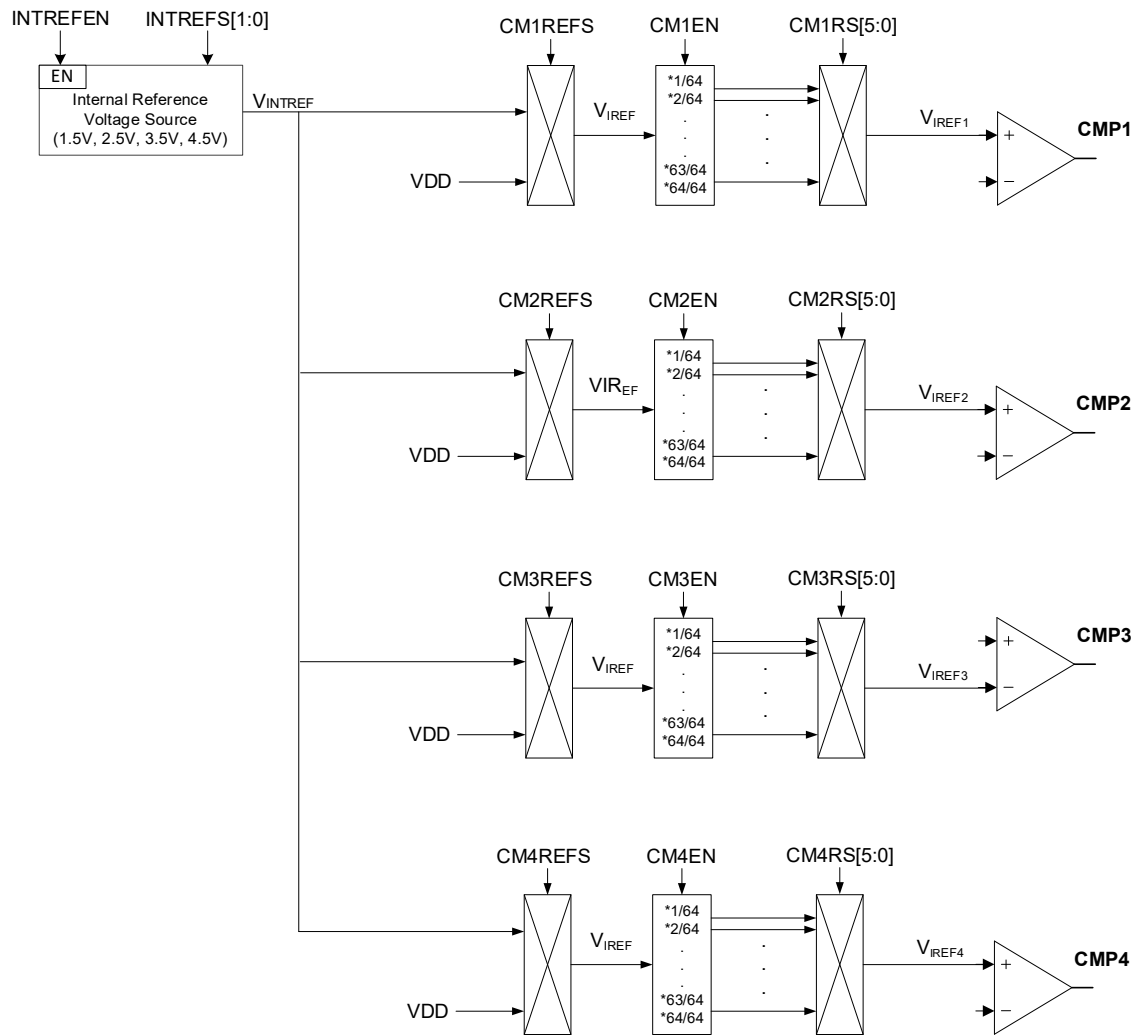
The microcontroller builds in 5 comparator functions (Comparator 0/1/2/3/4). When the positive input voltage is greater than the negative input voltage, the comparator output is high. When the positive input voltage is smaller than the negative input voltage, the comparator output is low. The comparator0/1/2/3/4 builds in P0CON register to set pure analog input pin. The comparator has flag indicator, interrupt function and IDLE Mode weak-up function for different application.



21.1 Internal Reference Voltage

The comparator builds in internal reference voltage function which can be comparator 1/2/3/4 positive or negative reference voltage source (V_{IREFn}). The internal reference voltage includes five internal voltage sources (4.5V, 3.5V, 2.5V, 1.5V and VDD) and controlled by CMnREFS and CMPnRS bits.

When CMnREFS bit is "0", the internal reference voltage source is from VDD. When CMPnREFS bit is "1", the internal reference voltage source is from INTREF function block (V_{INTREF}) and controlled by INTREFEN and INTREFS[1:0] bits. If INTREFS[1:0] is "10", comparator reference voltage is 4.5V. If INTREFS[1:0] is "01", comparator reference voltage is 3.5V. If INTREFS[1:0] is "00", comparator reference voltage is 2.5V. If INTREFS[1:0] is "11", comparator reference voltage is 1.5V.

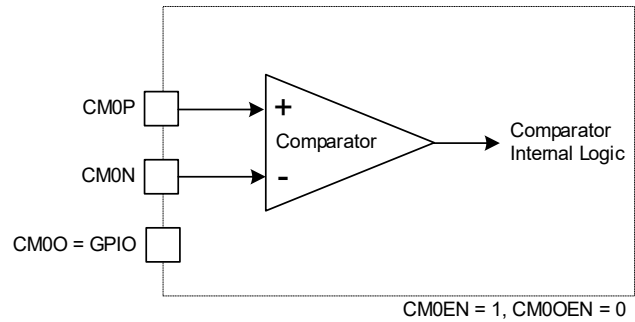
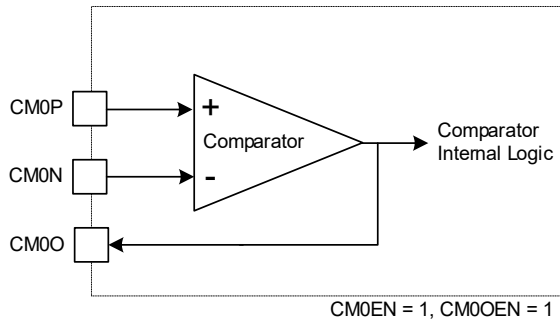


* **Note:**
Comparator 1/2/3/4 share the same internal reference voltage level if INTREF function block (V_{INTREF}) is selected.

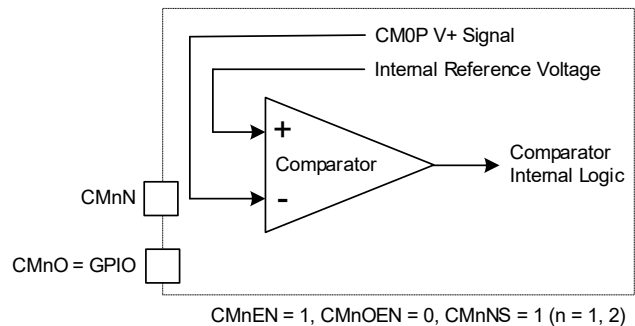
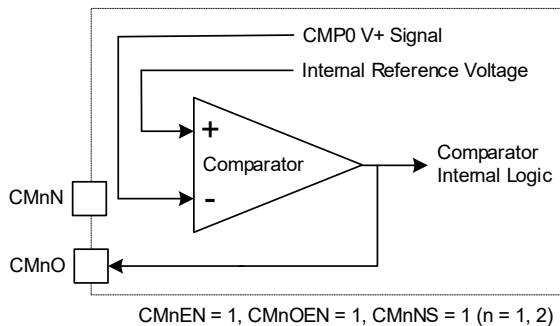
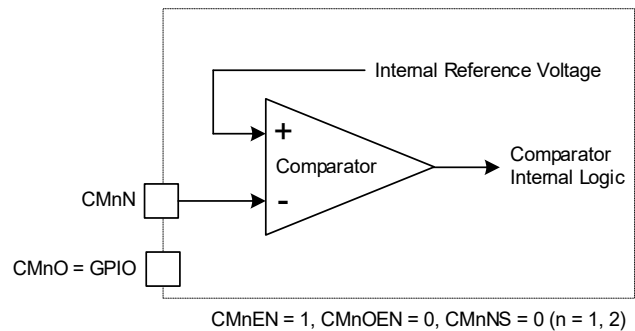
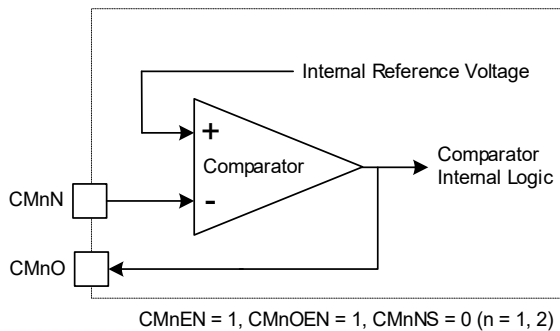
21.2 Configurations of Operation

Comparator pins are shared with GPIO controlled by CMnEN bit.

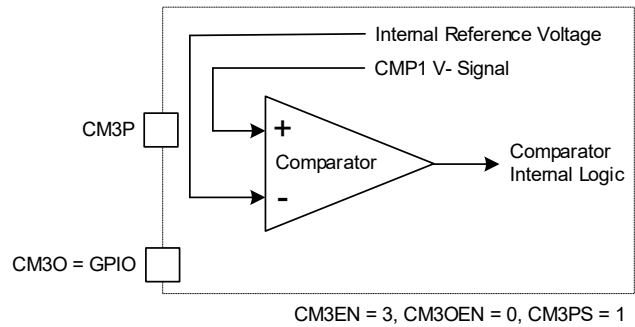
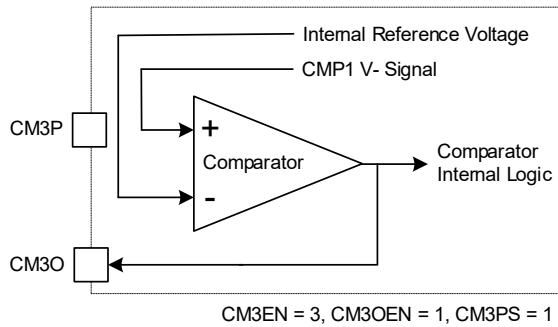
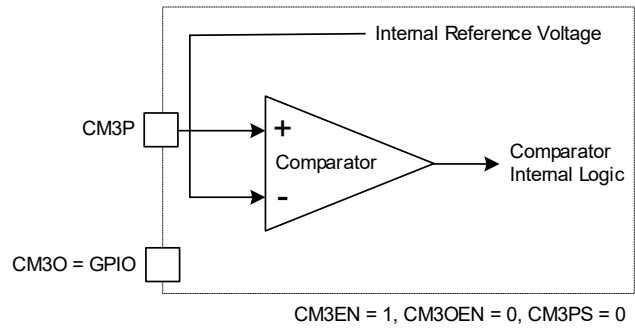
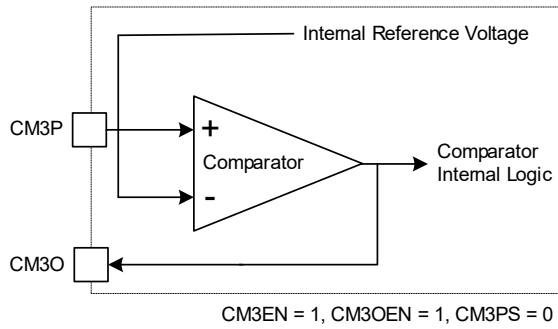
When CM0EN=1, CM0N pin is enabled connected to comparator negative terminal, and CM0P pin is enabled connected to comparator positive terminal. CM0OEN controls comparator output connected to GPIO or not. When CM0OEN=1, Comparator output terminal (CM0O) is connected to GPIO pins and isolate GPIO function.



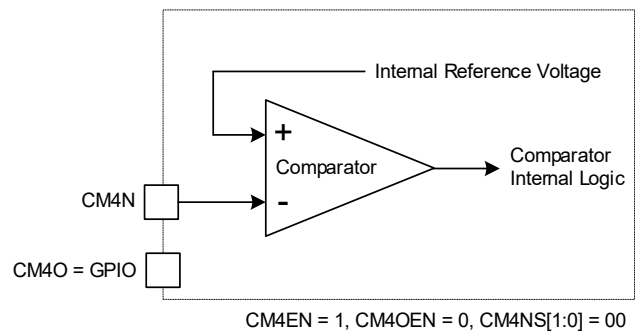
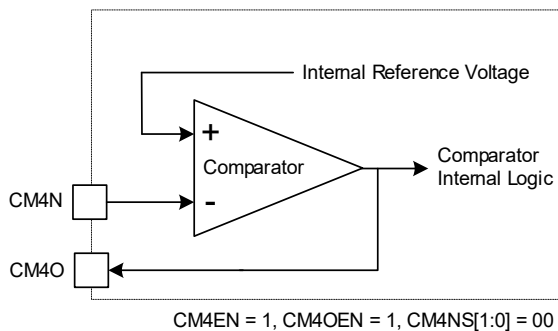
When $CMnEN = 1$ and $CMnNS = 0$ ($n = 1/2$), the comparator positive voltage source is from the internal reference voltage and the comparator negative voltage source is from $CMnN$ pin. When $CMnNS = 1$, the comparator negative voltage source is from $CMP0$ positive terminal signal. $CMnOEN$ controls comparator output connected to GPIO or not. When $CMnOEN=1$, Comparator output terminal ($CMnO$) is connected to GPIO pins and isolate GPIO function.

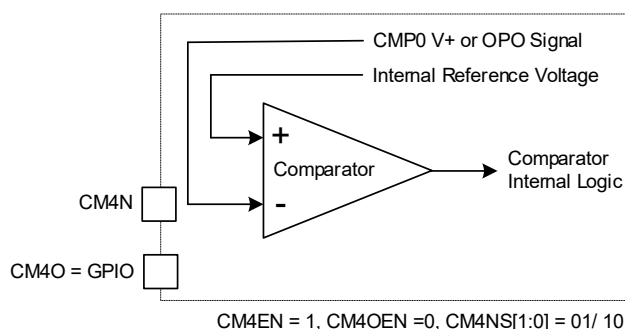
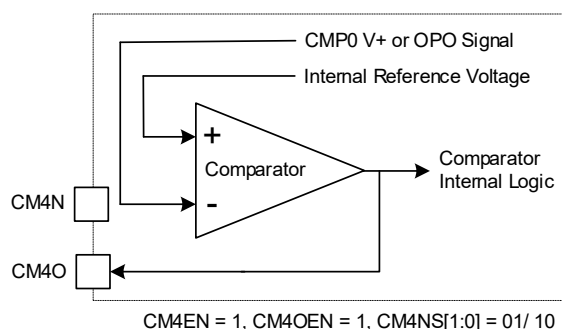


When $CM3EN = 1$ and $CM3PS = 0$, the comparator positive voltage source is from $CM3P$ pin and the comparator negative voltage source is from the internal reference voltage. When $CM3PS = 1$, the comparator negative voltage source is from $CMP1$ negative terminal signal. $CM3OEN$ controls comparator output connected to GPIO or not. When $CM3OEN=1$, Comparator output terminal ($CM3O$) is connected to GPIO pins and isolate GPIO function.



When CM4EN = 1 and CM4NS[1:0] = 00, the comparator positive voltage source is from the internal reference voltage and the comparator negative voltage source is from CM4N pin. When CM4NS = 01, the comparator negative voltage source is from CMP0 positive terminal signal. When CM4NS = 10, the comparator negative voltage source is from OPO signal. CM4NS = 11, the comparator negative voltage source is reserved. CM4OEN controls comparator output connected to GPIO or not. When CM4OEN=1, Comparator output terminal (CM4O) is connected to GPIO pins and isolate GPIO function.





21.3 Comparator Output Function

The comparator output signal is the source of comparator output function. The comparator output function includes:

CMnOUT output flag: The comparator output signal is connected to CMnOUT flag directly. CMnOUT bit responses comparator status immediately. Program reads comparator status from CMnOUT bit.

Comparator extern pin output function: The comparator output status can output to CMnO pin controlled by CMnOEN bit. When CMnOEN=0, the comparator output pin is GPIO mode. If CMnOEN=1, CMnO pin outputs comparator output status and isolates GPIO mode.

Comparator edge trigger and interrupt function: The comparator builds in interrupt function, and the trigger edge is programmable. CMnG[1:0] bit controls comparator trigger edge. When the edge trigger condition occurs, CMnF will be set automatically. To clear CMnF bit must be through program. If ECMn=1, program counter will be pointed to interrupt vector to execute interrupt service routine as CMnF is setting. When the trigger edge direction is equal to interrupt trigger condition, the system will execute interrupt operation immediately.

Comparator IDLE Mode Wake-up function: The comparator's wake-up function only support IDLE Mode (interrupt needs enable), not STOP Mode. If the trigger edge condition (comparator output status exchanging) is found, the system will be wake-up from IDLE Mode. Because in interrupt trigger condition, the CMnF is set as "1". Of course the interrupt routine is executed if the interrupt function enabled.

21.4 Time Control

Comparator builds in de-bounce time function and output delay function. The de-bounce time controls output hysteresis to filter output transition condition. The de-bounce time option has 16-step including no de-bounce, 2/Fcpu, 4/Fcpu, 6/Fcpu, 8/Fcpu, 10/Fcpu, 12/Fcpu, 14/Fcpu, 16/Fcpu, 18/Fcpu, 20/Fcpu, 22/Fcpu, 24/Fcpu, 26/Fcpu, 28/Fcpu, 30/Fcpu controlled by CMOD[3:0] bits.

Table 21-1 Comparator n de-bounce time (n = 0~4)

CMOD[3:0]		Fcpu = 16MHz	Fcpu = 4MHz
0000	No de-bounce	-	-
0001	Fcpu/2	0.125 us	0.500 us
0010	Fcpu/4	0.250 us	1.000 us
0011	Fcpu/6	0.375 us	1.500 us
0100	Fcpu/8	0.500 us	2.000 us
0101	Fcpu/10	0.625 us	2.500 us
0110	Fcpu/12	0.750 us	3.000 us
0111	Fcpu/14	0.875 us	3.500 us
1000	Fcpu/16	1.000 us	4.000 us
1001	Fcpu/18	1.125 us	4.500 us
1010	Fcpu/20	1.250 us	5.000 us
1011	Fcpu/22	1.375 us	5.500 us
1100	Fcpu/24	1.500 us	6.000 us
1101	Fcpu/26	1.625 us	6.500 us
1110	Fcpu/28	1.750 us	7.000 us
1111	Fcpu/30	1.875 us	7.500 us

The output delay function is only used for Comparator 0. The output delay has two clock source options: Fcpu, Fhosc. When CM0CLK = 0, the clock source f_{CKS} is Fhosc. When CM0CLK = 1, the clock source f_{CKS} is Fcpu. the delay option has 16-step including no delay, $4/f_{CKS}$, $8/f_{CKS}$, $12/f_{CKS}$, $16/f_{CKS}$, $20/f_{CKS}$, $24/f_{CKS}$, $28/f_{CKS}$, $32/f_{CKS}$, $36/f_{CKS}$, $40/f_{CKS}$, $44/f_{CKS}$, $48/f_{CKS}$, $52/f_{CKS}$, $56/f_{CKS}$, $60/f_{CKS}$ controlled by CM0DY[3:0] bits.

Table 21-2 Comparator 0 delay time

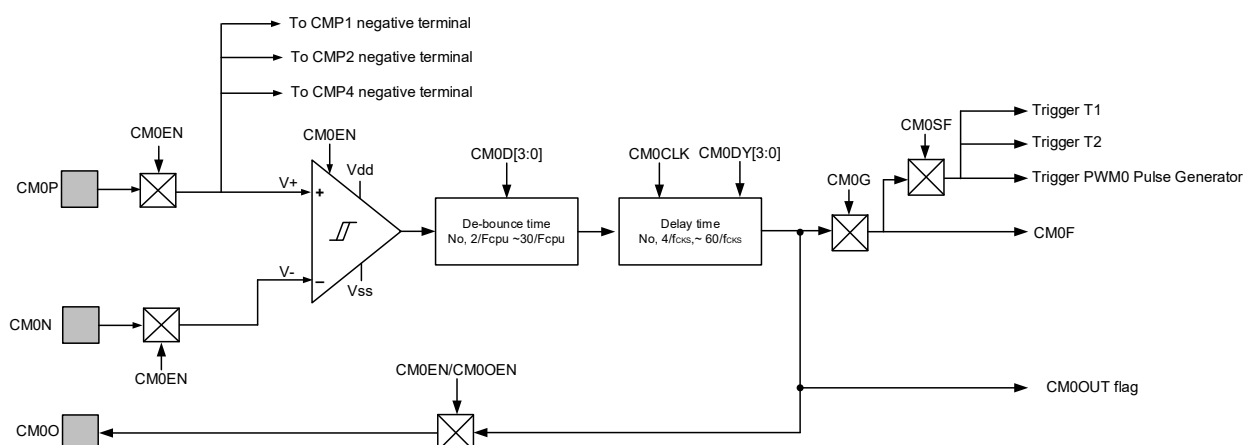
CM0DY[3:0]		CM0CLK = 0 ($f_{CKS} = F_{hosc}$)	CM0CLK = 1 ($f_{CKS} = F_{cpu}$)	
		Fhosc = 32MHz	Fcpu = 16MHz	Fcpu = 4MHz
0000	no de-bounce	-	-	-
0001	$f_{CKS} / 4$	0.125 us	0.250 us	1.000 us
0010	$f_{CKS} / 8$	0.250 us	0.500 us	2.000 us
0011	$f_{CKS} / 12$	0.375 us	0.750 us	3.000 us
0100	$f_{CKS} / 16$	0.500 us	1.000 us	4.000 us
0101	$f_{CKS} / 20$	0.625 us	1.250 us	5.000 us
0110	$f_{CKS} / 24$	0.750 us	1.500 us	6.000 us
0111	$f_{CKS} / 28$	0.875 us	1.750 us	7.000 us
1000	$f_{CKS} / 32$	1.000 us	2.000 us	8.000 us
1001	$f_{CKS} / 36$	1.125 us	2.250 us	9.000 us

1010	$f_{CKS} / 40$	1.250 μs	2.500 μs	10.000 μs
1011	$f_{CKS} / 44$	1.375 μs	2.750 μs	11.000 μs
1100	$f_{CKS} / 48$	1.500 μs	3.000 μs	12.000 μs
1101	$f_{CKS} / 52$	1.625 μs	3.250 μs	13.000 μs
1110	$f_{CKS} / 56$	1.750 μs	3.500 μs	14.000 μs
1111	$f_{CKS} / 60$	1.875 μs	3.750 μs	15.000 μs

21.5 Comparator Function Description

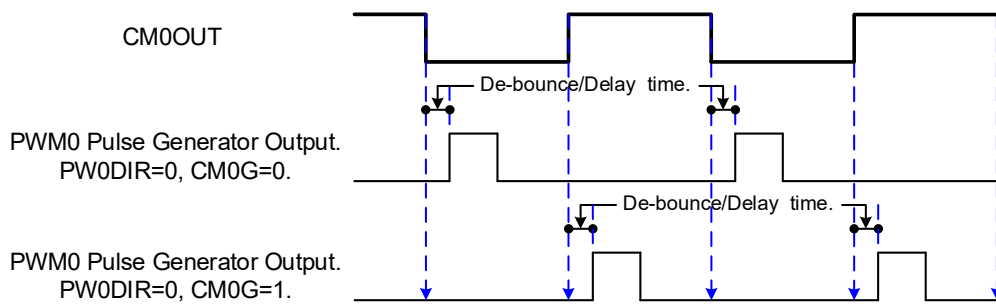
21.5.1 Comparator 0

The comparator 0 builds in PWM0 pulse generator trigger function. The comparator has normal comparator mode and PWM0 pulse output trigger source. The comparator isn't Rail-to-Rail structure. That means the input/output voltage isn't real from $V_{DD} \sim V_{SS}$ (Reference to "Electrical characteristics" chapter).

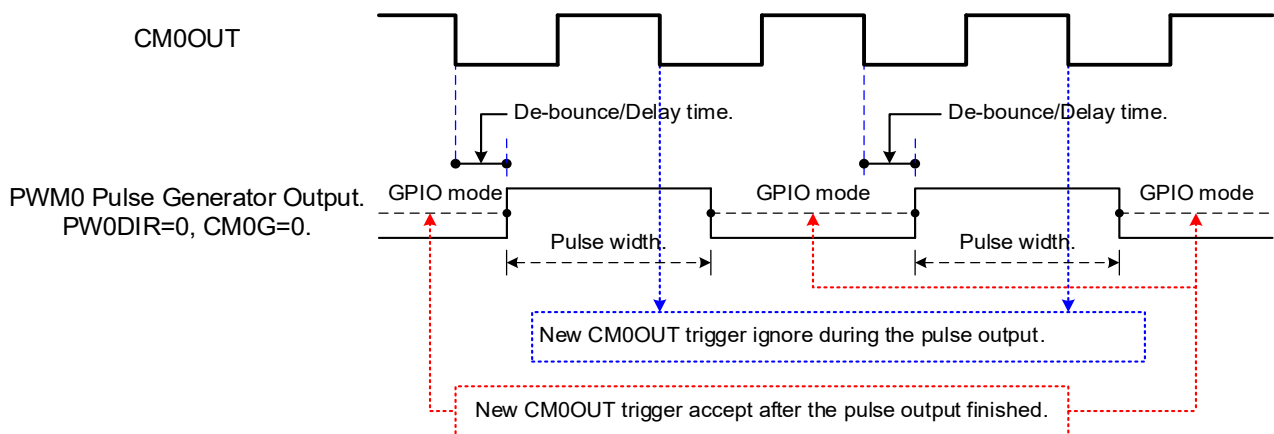


Besides normal comparator function, comparator 0 builds in a special mode to trigger PWM0 pulse generator through comparator output edge and controlled by CM0SF bit. When CM0SF=1, comparator 0 special mode is enabled. If comparator 0 output trigger condition occurs, PWM0 pulse generator is triggered to output a pulse signal, and comparator interrupt function activates. More detail operation is referred to PWM0 pulse generator contents.

- PWM0 pulse generator outputs signal (CM0G = 0/1).



- New PWM0OUT trigger accept after the pulse output finished but ignore during the pulse output.



★ **Note:**

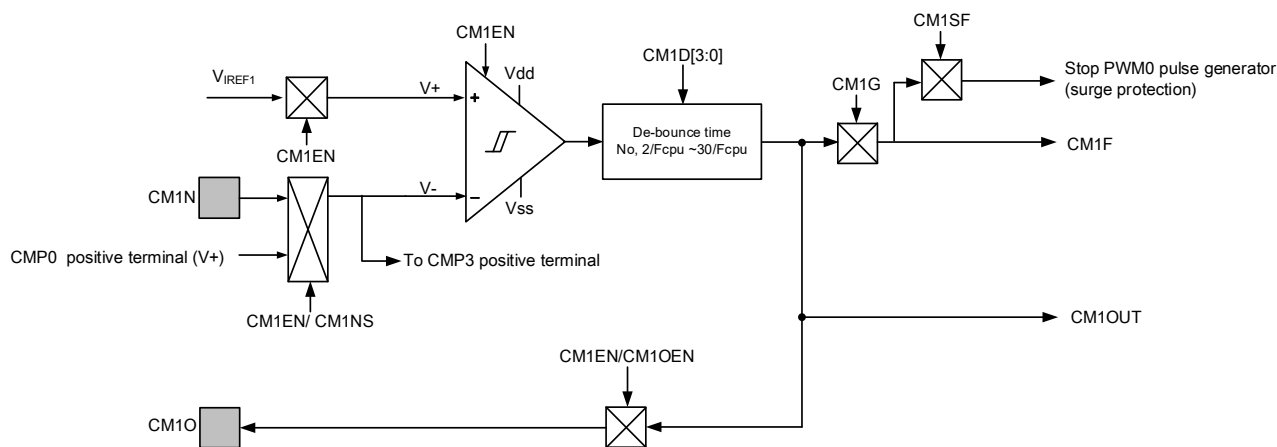
1. Switching between different PWM modes must be done while the PWM function is disabled. Thus, before changing the status of PW0PO or CM0SF bit, the PWM function must be disabled (PW0EN=0) first. Then, update the status of PW0PO or CM0SF bit and re-enable the PWM function (PW0EN=1) in the end.

2. When CM0SF bit is enabled, PWM output is triggered by CMP0 output edge. In this condition, PGOUT bit can't be modified by firmware.

21.5.2 Comparator 1

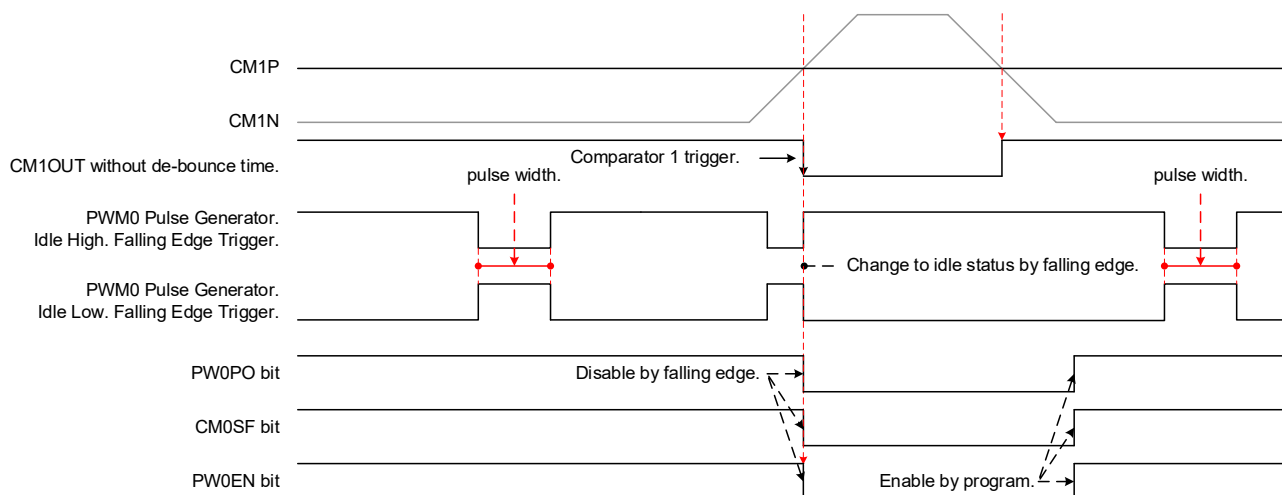
The comparator 1 builds in stopping PWM0 pulse generator function (surge protection). The comparator has normal comparator mode and stopping PWM0 pulse output trigger source. The comparator is not Rail-to-Rail structure. That means the input voltage is not real from Vdd~Vss

(Reference to “Electrical characteristics” chapter). The comparator builds in internal reference voltage connected to comparator positive terminal. Comparator 1 negative voltage is from CMP0 positive terminal or CM1N. positive terminal or CM1N.

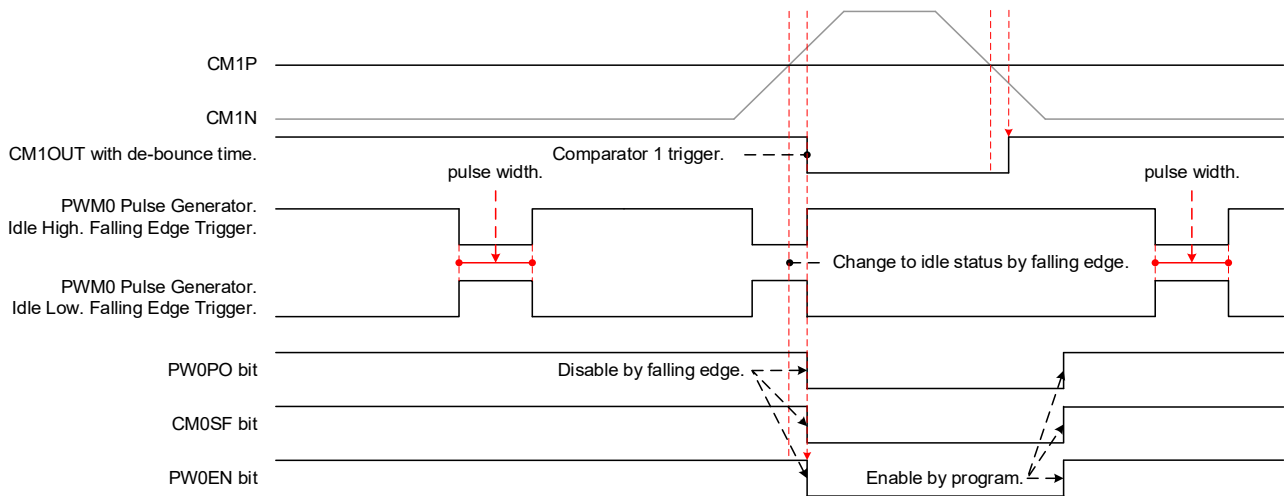


Besides normal comparator function, comparator 1 builds in a special mode for IH-cooker application (stop PWM0 pulse generator output signal, surge protection). The special mode is to trigger PWM0 pulse generator stopping output (surge protection) through comparator output edge and controlled by $CM1SF$ bit. When $CM1SF=1$, comparator 1 special mode is enabled. If comparator 1 output trigger condition occurs, PWM0 pulse generator function is disabled to turn off external device (like IGBT driving pulse signal). In this condition, $PW0PO$, $PW0EN$ and $CM0SF$ bits are cleared to disable pulse output function automatically. Pulse output pin exchanges to GPIO mode and last status. $CM1F$ is issued to indicate surge event. It is necessary to enable pulse generator by program.

- **Stop PWM0 pulse output @falling edge trigger, without de-bounce time.**



- Stop PWM0 pulse output @falling edge trigger, with de-bounce time.

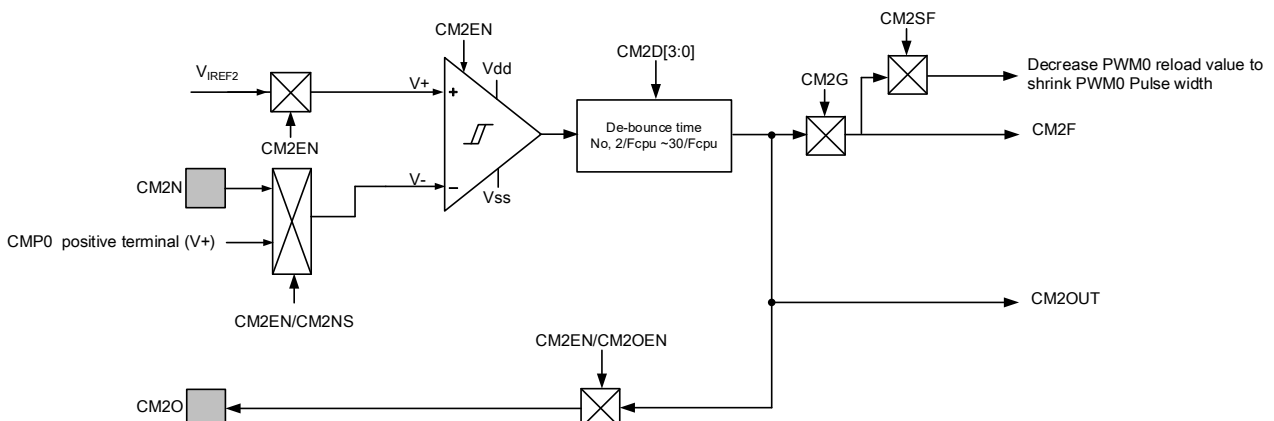


*** Note:**

If PWM0 pulse output is stopped by comparator 1 special mode trigger, the CM0SF, PW0PO and PW0EN bits are cleared automatically. It is necessary to set CM0SF, PW0PO and PW0EN bits by program to recover PWM0 pulse generator function.

21.5.3 Comparator 2

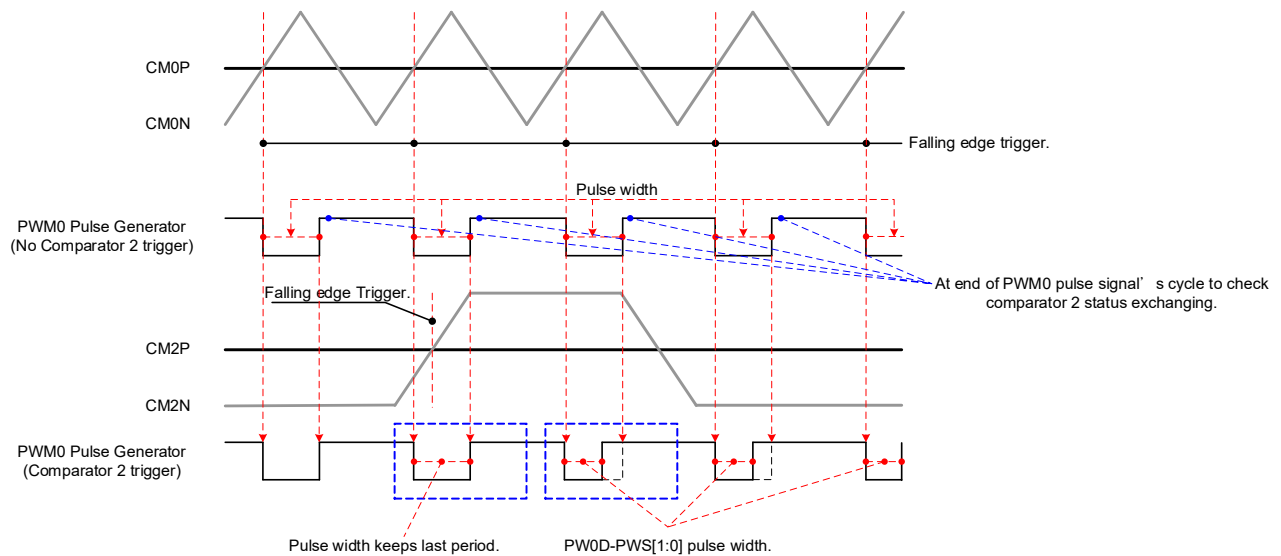
The comparator 2 builds in shrinking PWM0 pulse width function. The comparator has normal comparator mode and shrinking PWM0 pulse width trigger source. The comparator is not Rail-to-Rail structure. That means the input/output voltage is not real from Vdd~Vss (Reference to "Electrical characteristics" chapter). The comparator builds in internal reference voltage connected to comparator positive terminal. Comparator 2 negative voltage is from CMP0 positive terminal or CM2N.



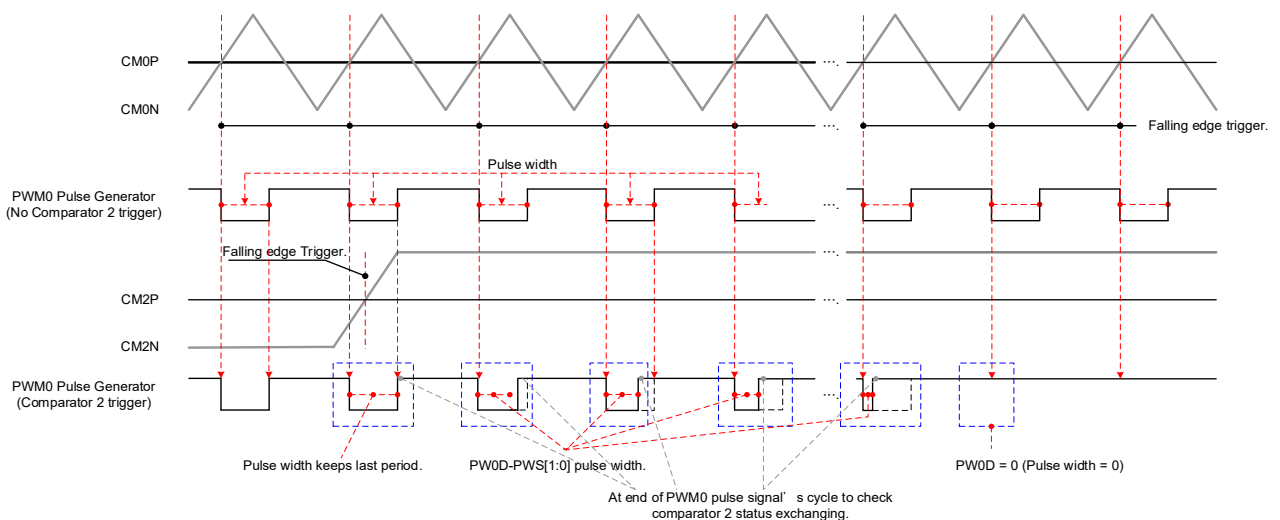
Besides normal comparator function, comparator 2 builds in a special mode to shrink PWM0 pulse width through decreasing PW0D register. The special mode is to trigger decreasing PW0D register and PWM0 pulse width will be shrunk through comparator output edge and controlled by CM2SF bit. When CM2SF = 1, comparator 2 special mode is enabled. If comparator 2 output trigger

condition occurs, PWOD register increases unit time to shrink PWM0 pulse width. The unit time is controlled by PWS[1:0].

Step 1: When comparator 2 first trigger occurs, PW0D - PWS[1:0] once to shrink PWM0 pulse width. PWM0 pulse width reduces unit time automatically. The hardware checks comparator 2 high tension status at the end of PWM0 pulse signal's cycle. If the comparator output status exchanges, to expand PWM0 pulse width through increasing PW0D register by program.



Step 2: If comparator output status doesn't exchange, PW0D - PWS[1:0] at the end of one PW,0 pulse signal's cycle and outputs the new pulse until comparator 2 output status exchanges until PW0D = 0.

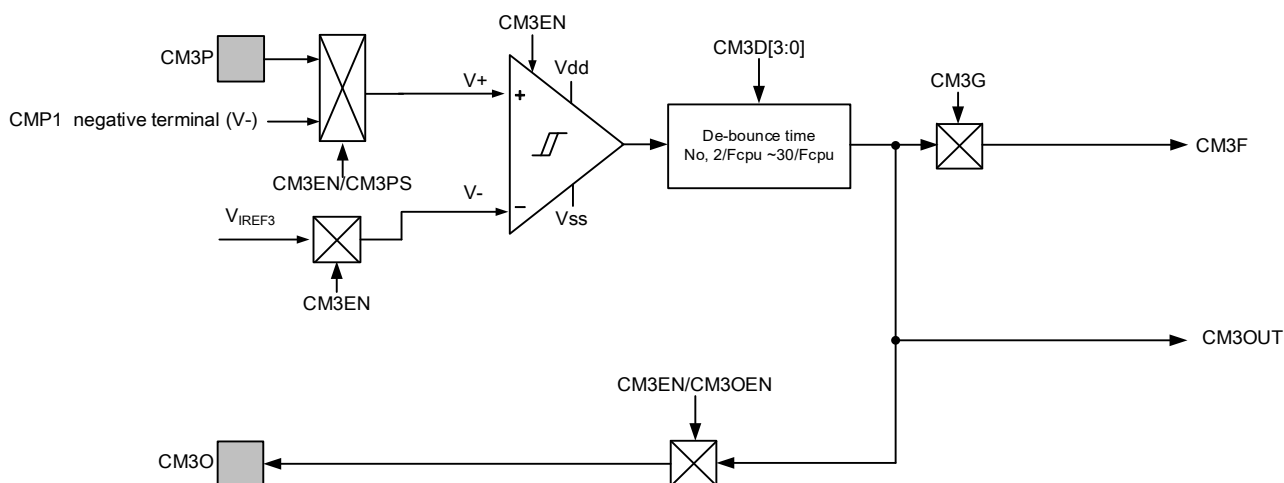


★ **Note:**

If PW0D is decreased to 0x00, PW0D will keep 0x00 and not increase again, even the comparator output status never occurs exchanging

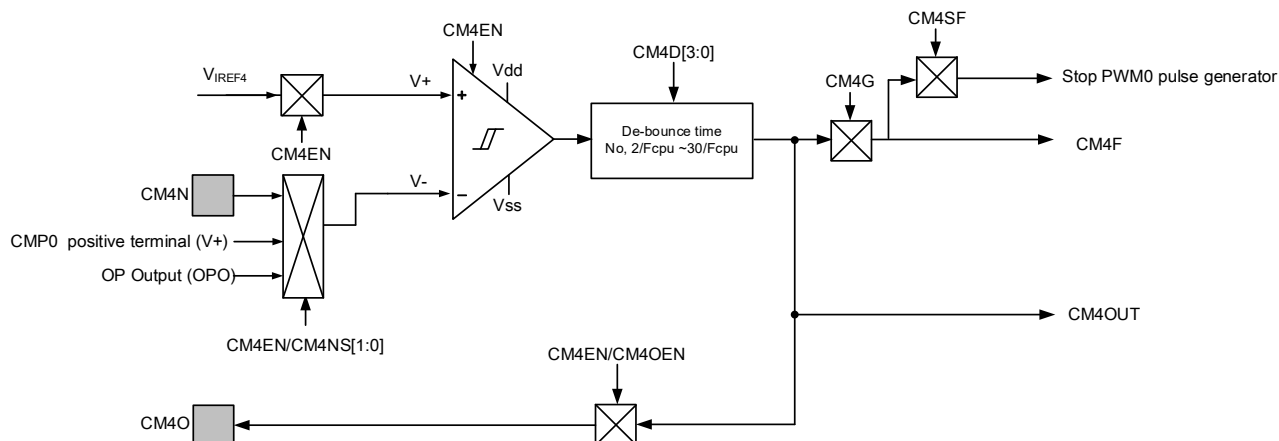
21.5.4 Comparator 3

The comparator 3 only has normal comparator mode. The comparator is not Rail-to-Rail structure. That means the input/output voltage is not real from $V_{dd} \sim V_{ss}$ (Reference to “Electrical characteristics” chapter). The comparator builds in internal reference voltage connected to comparator negative terminal. Comparator 3 positive voltage is from CMP1 negative terminal or CM3P. Comparator 3 positive voltage is from CMP1 negative terminal or CM3P.



21.5.5 Comparator 4

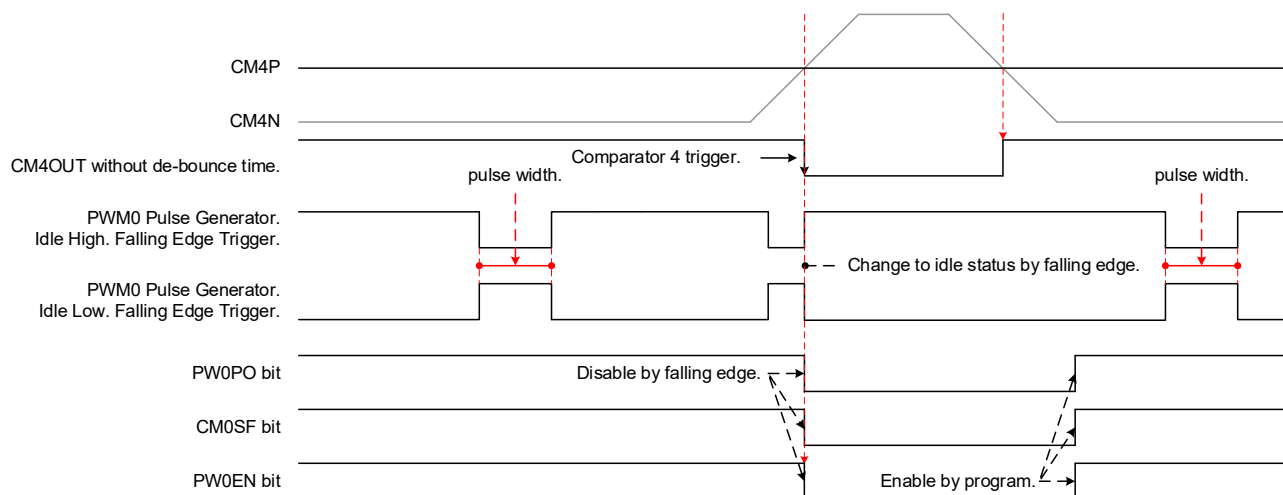
The comparator 4 builds in stopping PWM0 pulse generator function (surge protection). The comparator has normal comparator mode and stopping PWM0 pulse output trigger source. The comparator is not Rail-to-Rail structure. That means the input voltage is not real from $V_{dd} \sim V_{ss}$ (Reference to “Electrical characteristics” chapter). The comparator builds in internal reference voltage connected to comparator positive terminal. Comparator 4 negative voltage is from CMP0 positive terminal, CM4N or OP output (OPO).



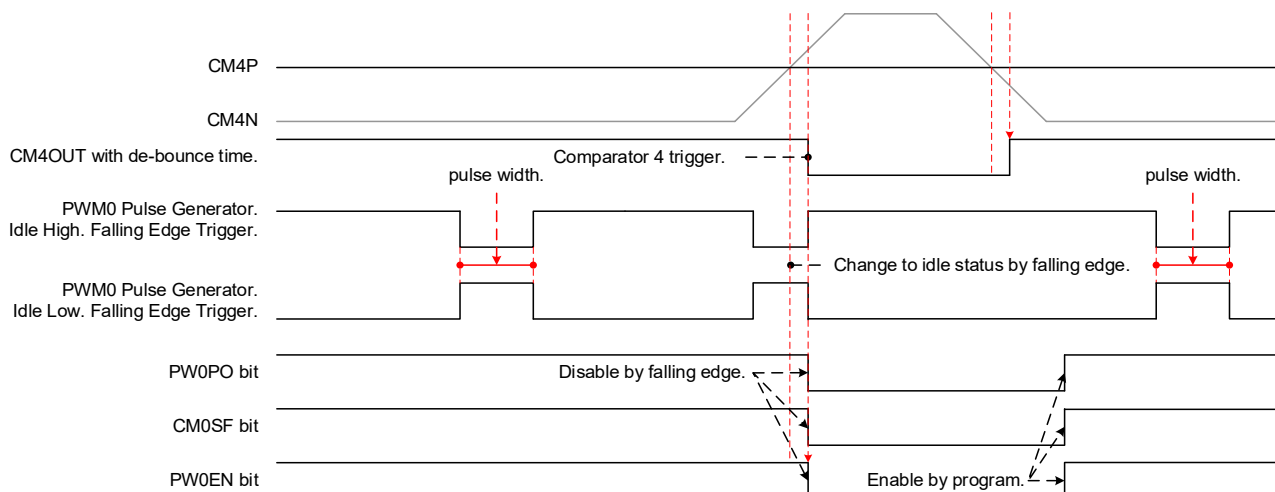
Besides normal comparator function, comparator 4 builds in a special mode for IH-cooker application (stop PWM0 pulse generator output signal, surge protection). The special mode is to trigger PWM0 pulse generator stopping output (surge protection) through comparator output edge

and controlled by CM4SF bit. When CM4SF=1, comparator 4 special mode is enabled. If comparator 4 output trigger condition occurs, PWM0 pulse generator function is disabled to turn off external device (like IGBT driving pulse signal). In this condition, PW0PO, PW0EN and CM0SF bits are cleared to disable pulse output function automatically. Pulse output pin exchanges to GPIO mode and last status. CM4F is issued to indicate surge event. It is necessary to enable pulse generator by program.

● **Stop PWM0 pulse output @falling edge trigger, without de-bounce time.**



● **Stop PWM0 pulse output @falling edge trigger, with de-bounce time.**

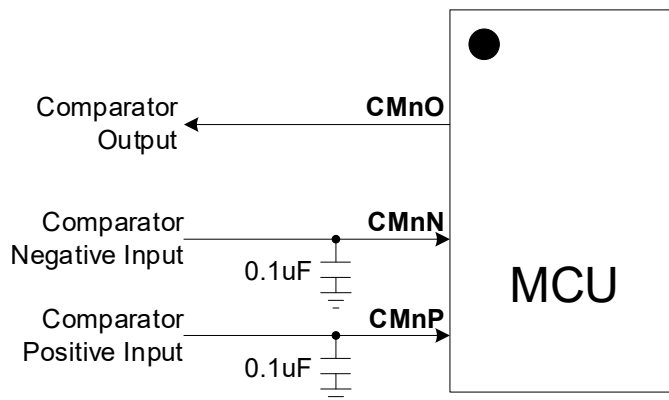


★ **Note:**

If PWM0 pulse output is stopped by comparator 4 special mode trigger, the CM0SF, PW0PO and PW0EN bits are cleared automatically. It is necessary to set CM0SF, PW0PO and PW0EN bits by program to recover PWM0 pulse generator function.

21.6 Comparator Application Notice

The comparator is to compares the positive voltage and negative voltage to output result. The positive and negative sources are analog signal. In hardware application circuit, the comparator input pins must be connected a 0.1uF capacitor to reduce power noise and make the input signal more stable. The application circuit is as following.



21.7 Comparator Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CM0M	CM0EN	CM0OEN	-	CM0SF	CM0G	CM0CLK	-	-
CM1M	CM1EN	CM1OEN	-	CM1SF	CM1G	-	CM1NS	-
CM2M	CM2EN	CM2OEN	-	CM2SF	CM2G	-	CM2NS	-
CM3M	CM3EN	CM3OEN	-	-	CM3G	-	CM3PS	-
CM4M	CM4EN	CM4OEN	-	CM4SF	CM4G	CM4NS1	CM4NS0	-
CMDDB0	CM0DY3	CM0DY2	CM0DY1	CM0DY0	CM0D3	CM0D2	CM0D1	CM0D0
CMDDB1	-	-	-	-	CM1D3	CM1D2	CM1D1	CM1D0
CMDDB2	PWS1	PWS0	-	-	CM2D3	CM2D2	CM2D1	CM2D0
CMDDB3	-	-	-	-	CM3D3	CM3D2	CM3D1	CM3D0
CMDDB4	-	-	-	-	CM4D3	CM4D2	CM4D1	CM4D0
CM1REF	CM1REFS	-	CM1RS5	CM1RS4	CM1RS3	CM1RS2	CM1RS1	CM1RS0
CM2REF	CM2REFS	-	CM2RS5	CM2RS4	CM2RS3	CM2RS2	CM2RS1	CM2RS0
CM3REF	CM3REFS	-	CM3RS5	CM3RS4	CM3RS3	CM3RS2	CM3RS1	CM3RS0
CM4REF	CM4REFS	-	CM4RS5	CM4RS4	CM4RS3	CM4RS2	CM4RS1	CM4RS0
CMOUT	-	-	-	CM4OUT	CM3OUT	CM2OUT	CM1OUT	CM0OUT
INTREF	INTREFEN	-	-	-	-	-	INTREFS1	INTREFS0
IEN0	EAL	-	-	-	ET1	-	ET0	EX0
IEN2	ECMP4	ECMP3	ECMP2	ECMP1	ECMP0	ET2	EADC	ELVD
IRCON2	CM4F	CM3F	CM2F	CM1F	CM0F	TF2	ADCF	LVDF

P0CON	P0CON7	P0CON6	P0CON5	P0CON4	P0CON3	P0CON2	-	P0CON0
-------	--------	--------	--------	--------	--------	--------	---	--------

CM0M Register (0xF028)

Bit	Field	Type	Initial	Description
7	CM0EN	R/W	0	Comparator 0 control bit. 0: Disable. CM0P/CM0N pins are GPIO mode. 1: Enable. CM0P/CM0N pins are CMP input pins.
6	CM0OEN	R/W	0	Comparator 0 output pin control bit. 0: Disable. CM0O is GPIO mode. 1: Enable. CM0O is comparator output pin and isolate GPIO function.
4	CM0SF	R/W	0	Comparator 0 special mode control bit. 0: Disable. Normal comparator function. 1: Enable. CMP0 output edge triggers PWM0 pulse generator.
3	CM0G	R/W	0	Comparator interrupt trigger direction control bit. 0: Falling edge trigger. CMP0 output $V+ < V-$. 1: Rising edge trigger. CMP0 output $V+ > V-$.
2	CM0CLK	R/W	0	Delay time clock source selection. 0: Fhosc. 1: Fcpu.
Else	Reserved	R	0	

CM1M Register (0xF02C)

Bit	Field	Type	Initial	Description
7	CM1EN	R/W	0	Comparator 1 control bit. 0: Disable. CM1N pins are GPIO mode. 1: Enable. CM1N pins are CMP input pins.
6	CM1OEN	R/W	0	Comparator 1 output pin control bit. 0: Disable. CM1O is GPIO mode. 1: Enable. CM1O is comparator output pin and isolate GPIO function.
4	CM1SF	R/W	0	Comparator 1 special mode control bit. 0: Disable. Normal comparator function. 1: Enable. CMP1 output triggers PWM0 pulse stopping. (Surge protection)

3	CM1G	R/W	0	Comparator interrupt trigger direction control bit. 0: Falling edge trigger. CMP1 output $V+ < V-$. 1: Rising edge trigger. CMP1 output $V+ > V-$.
1	CM1NS	R/W	0	CMP1 negative terminal input selection. 0: CM1N pin. 1: CMP0 positive terminal.
Else	Reserved	R	0	

CM2M Register (0xF030)

Bit	Field	Type	Initial	Description
7	CM2EN	R/W	0	Comparator 2 control bit. 0: Disable. CM2N pins are GPIO mode. 1: Enable. CM2N pins are CMP input pins.
6	CM2OEN	R/W	0	Comparator 2 output pin control bit. 0: Disable. CM2O is GPIO mode. 1: Enable. CM2O is comparator output pin and isolate GPIO function.
4	CM2SF	R/W	0	Comparator 2 special mode control bit. 0: Disable. Normal comparator function. 1: Enable. CMP1 output triggers Shrinking PWM0 pulse width.
3	CM2G	R/W	0	Comparator interrupt trigger direction control bit. 0: Falling edge trigger. CMP2 output $V+ < V-$. 1: Rising edge trigger. CMP2 output $V+ > V-$.
1	CM2NS	R/W	0	CMP2 negative terminal input selection. 0: CM2N pin. 1: CMP0 positive terminal.
Else	Reserved	R	0	

CM3M Register (0xF034)

Bit	Field	Type	Initial	Description
7	CM3EN	R/W	0	Comparator 3 control bit. 0: Disable. CM3P pins are GPIO mode. 1: Enable. CM3P pins are CMP input pins.
6	CM3OEN	R/W	0	Comparator 3 output pin control bit. 0: Disable. CM3O is GPIO mode. 1: Enable. CM3O is comparator output pin and isolate GPIO function.

3	CM3G	R/W	0	Comparator interrupt trigger direction control bit. 0: Falling edge trigger. CMP3 output $V_+ < V_-$. 1: Rising edge trigger. CMP3 output $V_+ > V_-$.
1	CM3PS	R/W	0	CMP3 positive terminal input selection. 0: CM3P pin. 1: CMP1 negative terminal.
Else	Reserved	R	0	

CM4M Register (0xF038)

Bit	Field	Type	Initial	Description
7	CM4EN	R/W	0	Comparator 4 control bit. 0: Disable. CM4N pins are GPIO mode. 1: Enable. CM4N pins are CMP input pins.
6	CM4OEN	R/W	0	Comparator 4 output pin control bit. 0: Disable. CM4O is GPIO mode. 1: Enable. CM4O is comparator output pin and isolate GPIO function.
4	CM4SF	R/W	0	Comparator 4 special mode control bit. 0: Disable. Normal comparator function. 1: Enable. CMP4 output triggers PWM0 pulse stopping. (Surge protection)
3	CM4G	R/W	0	Comparator interrupt trigger direction control bit. 0: Falling edge trigger. CMP1 output $V_+ < V_-$. 1: Rising edge trigger. CMP1 output $V_+ > V_-$.
2..1	CM4NS[1:0]	R/W	0	CMP4 negative terminal input selection. 00: CM4N pin. 01: CMP0 positive terminal. 10: OPO output signal. 11: Reserved.
Else	Reserved	R	0	

CMDB0 Register (0xF029)

Bit	Field	Type	Initial	Description
7..4	CM0DY[3:0]	R/W	0	Comparator 0 delay time. f_{CKS} is controlled by CM0CLK bit. 0000: No delay, 0001: $4/f_{CKS}$, 0010: $8/f_{CKS}$, 0011: $12/f_{CKS}$, 0100: $16/f_{CKS}$, 0101: $20/f_{CKS}$, 0110: $24/f_{CKS}$, 0111: $28/f_{CKS}$, 1000: $32/f_{CKS}$,

				1001: 36/f _{CKS} , 1010: 40/f _{CKS} , 1011: 44/f _{CKS} , 1100: 48/f _{CKS} , 1101: 52/f _{CKS} , 1110: 56/f _{CKS} , 1111: 60/f _{CKS}
3..0	CM0D[3:0]	R/W	0	Comparator 0 de-bounce time. 0000: No de-bounce, 0001: 2/Fcpu, 0010: 4/Fcpu, 0011: 6/Fcpu, 0100: 8/Fcpu, 0101: 10/Fcpu, 0110: 12/Fcpu, 0111: 14/Fcpu, 1000: 16/Fcpu, 1001: 18/Fcpu, 1010: 20/Fcpu, 1011: 22/Fcpu, 1100: 24/Fcpu, 1101: 26/Fcpu, 1110: 28/Fcpu, 1111: 30/Fcpu

CMDB1:0xF02D, CMDB3: 0xF035, CMDB4: 0xF039

Bit	Field	Type	Initial	Description
7..4	Reserved.	R	0	
3..0	CMnD[3:0]	R/W	0	Comparator n de-bounce time. (n=1/3/4) 0000: No de-bounce, 0001: 2/Fcpu, 0010: 4/Fcpu, 0011: 6/Fcpu, 0100: 8/Fcpu, 0101: 10/Fcpu, 0110: 12/Fcpu, 0111: 14/Fcpu, 1000: 16/Fcpu, 1001: 18/Fcpu, 1010: 20/Fcpu, 1011: 22/Fcpu, 1100: 24/Fcpu, 1101: 26/Fcpu, 1110: 28/Fcpu, 1111: 30/Fcpu

CMDB2 Register (0xF031)

Bit	Field	Type	Initial	Description
7..6	PWS [1:0]	R/W	00	Comparator 2 output de-bounce unit time select bits. 00b = 1 unit time, 01b = 2 unit times, 10b = 4 unit times, 11b = 8 unit times.
5..4	Reserved.	R	0	
3..0	CM2D[3:0]	R/W	0	Comparator 2 de-bounce time. 0000: No de-bounce, 0001: 2/Fcpu, 0010: 4/Fcpu, 0011: 6/Fcpu, 0100: 8/Fcpu, 0101: 10/Fcpu, 0110: 12/Fcpu, 0111: 14/Fcpu, 1000: 16/Fcpu, 1001: 18/Fcpu, 1010: 20/Fcpu, 1011: 22/Fcpu, 1100: 24/Fcpu, 1101: 26/Fcpu, 1110: 28/Fcpu, 1111: 30/Fcpu

CM1REF: 0xF02E, CM2REF: 0xF032, CM3REF: 0xF036, CM4REF: 0xF03A

Bit	Field	Type	Initial	Description
7	CMnREFS	R/W	0	CMPn internal reference voltage source (V_{REF}) select bit. ($n=1/2/3/4$) 0: VDD. 1: V_{INTREF} .
6	Reserved	R	0	
5..0	CMnRS[5:0]	R/W	0x00	Comparator n internal reference voltage (V_{REFn}) select bit. ($n=1/2/3/4$) 000000: $V_{REFn} = V_{REF}$. 000001~111111: $V_{REFn} = V_{REF} * CMnRS [5:0]/64$

CMOUT Register (0xCA)

Bit	Field	Type	Initial	Description
4	CM4OUT	R	0	Comparator 4 output flag bit. 0: CMP4 V+ voltage is less than V- voltage. 1: CMP4 V+ voltage is larger than V- voltage.
3	CM3OUT	R	0	Comparator 3 output flag bit. 0: CMP3 V+ voltage is less than V- voltage. 1: CMP3 V+ voltage is larger than V- voltage.
2	CM2OUT	R	0	Comparator 2 output flag bit. 0: CMP2 V+ voltage is less than V- voltage. 1: CMP2 V+ voltage is larger than V- voltage.
1	CM1OUT	R	0	Comparator 1 output flag bit. 0: CMP1 V+ voltage is less than V- voltage. 1: CMP1 V+ voltage is larger than V- voltage.
0	CM0OUT	R	0	Comparator 0 output flag bit. 0: CMP0 V+ voltage is less than V- voltage. 1: CMP0 V+ voltage is larger than V- voltage.
Else	Reserved	R	0	

INTREF Register (0xF040)

Bit	Field	Type	Initial	Description
7	INTREFEN	R/W	0	CMP internal reference voltage control bit. 0: Disable. 1: Enable.
6..2	Reserved	R	0	
1..0	INTREFS[1:0]	R/W	00	CMP internal reference voltage selects bits.

00: $V_{INTREF} = 2.5V$

01: $V_{INTREF} = 3.5V$

10: $V_{INTREF} = 4.5V$

11: $V_{INTREF} = 1.5V$

IEN0 Register (0xA8)

Bit	Field	Type	Initial	Description
7	EAL	R/W	0	Interrupts enable. Refer to Chapter Interrupt
Else				Refer to other chapter(s)

IEN2 Register (0xC8)

Bit	Field	Type	Initial	Description
7	ECMP4	R/W	0	CMP4 interrupt control bit. 0: Disable CMP4 interrupt function. 1: Enable CMP4 interrupt function.
6	ECMP3	R/W	0	CMP3 interrupt control bit. 0: Disable CMP3 interrupt function. 1: Enable CMP3 interrupt function.
5	ECMP2	R/W	0	CMP2 interrupt control bit. 0: Disable CMP2 interrupt function. 1: Enable CMP2 interrupt function.
4	ECMP1	R/W	0	CMP1 interrupt control bit. 0: Disable CMP1 interrupt function. 1: Enable CMP1 interrupt function.
3	ECMP0	R/W	0	CMP0 interrupt control bit. 0: Disable CMP0 interrupt function. 1: Enable CMP0 interrupt function.
Else				Refer to other chapter(s)

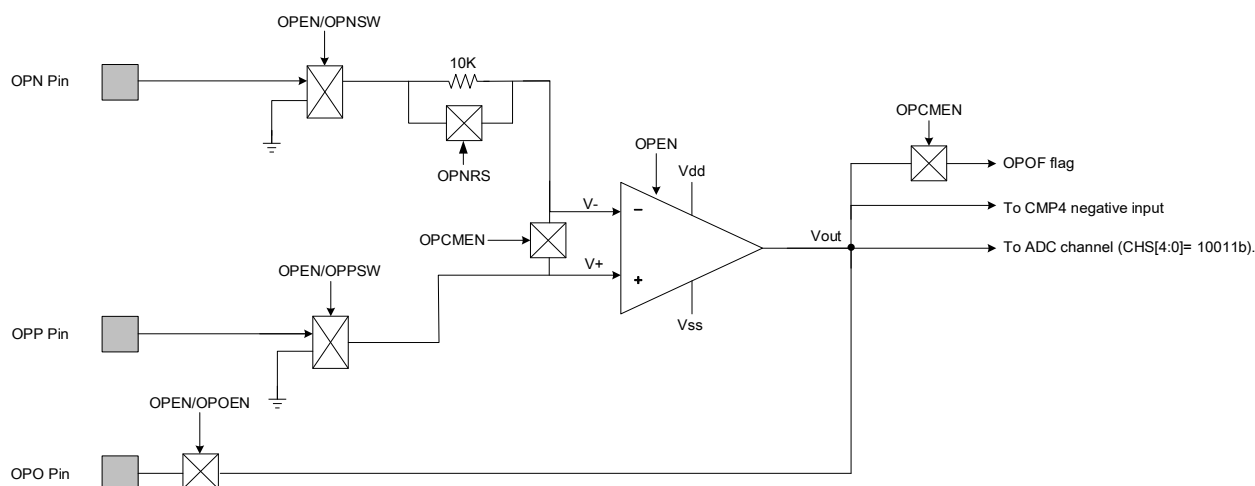
IRCON2 Register (0xA7)

Bit	Field	Type	Initial	Description
7	CM4F	R/W	0	CMP4 interrupt request flag. 0: None CMP4 interrupt request 1: CMP4 interrupt request.
6	CM3F	R/W	0	CMP3 interrupt request flag. 0: None CMP3 interrupt request 1: CMP3 interrupt request.
5	CM2F	R/W	0	CMP2 interrupt request flag.

				0: None CMP2 interrupt request 1: CMP2 interrupt request.
4	CM1F	R/W	0	CMP1 interrupt request flag. 0: None CMP1 interrupt request 1: CMP1 interrupt request.
3	CM0F	R/W	0	CMP0 interrupt request flag. 0: None CMP0 interrupt request 1: CMP0 interrupt request.

22 OPA

The microcontroller builds in one operational amplifier. The OP-Amp isn't Rail-to-Rail structure. That means the input/output voltage isn't real from VDD–VSS (Reference to “Electrical characteristics” chapter). The OP-Amp output pin is programmable to connect with comparator 4 and ADC input channel for voltage measurement.



22.1 Configurations of Operation

The OP-AMP pins are shared with GPIO controlled by OPEN, OPPSW, OPNSW and OPOEN bits. When OPEN=0, OP AMP pins are GPIO mode. When OPEN=1, OP-AMP pins selection table is as following.

OP-AMP	Pin Control	OP Positive Pin	OP Negative Pin	OP Output Pin
OPEN = 0	-	All pins are GPIO mode.		
OPEN = 1	OPPSW = 0	OPP (Vin+)	-	-
	OPPSW = 1	GPIO	-	-
	OPNSW = 0	-	OPON (Vin-)	-
	OPNSW = 1	-	GPIO	-
	OPOEN = 0	-	-	GPIO
	OPOEN = 1	-	-	OP0O (Vout)

If OPPSW = 1, the OPP pin is GPIO mode and OP-Amp positive terminal is programmable to connect internal ground. If OPNSW = 1, the OPN pin is GPIO mode and the OP-Amp negative terminal is programmable to connect internal ground.

OP output pins are also connected to ADC internal AIN19 channel and CMP4 negative terminal. See detailed description of ADC section and Comparator section.

22.2 OPA Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPM	OPNRS	-	OPDRV	OPOEN	OPOF	OPNSW	OPPSW	OPEN
OPCAL	OPCMEN	-	OPC5	OPC4	OPC3	OPC2	OPC1	OPC0

OPM Register (0xF020)

Bit	Field	Type	Initial	Description
7	OPNRS	R/W	0	OPA negative input resistor select bit 0: 10K Ohm. 1: Short.
5	OPDRV	R/W	0	OP-Amp driving mode. 0 = High driving mode. 1 = Low driving mode.
4	OPOEN	R/W	0	OPO output pin control bit. 0 = Disable. OPO pin is GPIO mode. 1 = Enable. OPO pin is OP output pin.
3	OPOF	R/W	0	OP Amp output status flag. When OPCMEN = 1, the OPO bit indicate the OP output status.
2	OPNSW	R/W	0	OP-Amp negative input source select bit. 0: OPN pin is OP-Amp negative input. 1: OPN pin is GPIO function and OP-Amp negative terminal connect to internal ground.
1	OPPSW	R/W	0	OP-Amp positive input source select bit. 0: OPP pin is OP-Amp positive input. 1: OPP pin is GPIO function and OP-Amp positive terminal connect to internal ground.
0	OPEN	R/W	0	OP-Amp enable bit. 0: Disable. OP-Amp disable, OPO/OPP/OPN pins are GPIO mode. 1: Enable. OP-Amp enable, OPO/OPP/OPN pins are OP-Amp input and output pins.

OPCAL Register (0xF022)

Bit	Field	Type	Initial	Description
-----	-------	------	---------	-------------

7	OPCMEN	R/W	0	OP-Amp input offset calibration mode enable bit. 0 = Disable. 1 = Enable. Enable input offset calibration mode.
6	Reserved	R	0	
5..0	OPC[5:0]	R/W	0x00	OP Amp input offset calibration value.

23 CRC

The CRC (cyclic redundancy check) calculation unit is used to get a CRC code from 8-bit data word and a generator polynomial. Among other applications, CRC-based techniques are used to verify data transmission or storage integrity. The CRC calculation circuit helps compute a signature of the software during runtime, to be compared with a reference signature generated at link time and stored at a given memory location.

The CRC generator keeps running in IDLE mode but doesn't have wake-up function. In stop mode, the CRC generator is stop.

- Support
 - CRC-16 polynomial: $X^{16}+X^{15}+X^2+1$
 - CRC-16-CCITT polynomial: $X^{16}+X^{12}+X^5+1$
- Handles 8-bit data size
- Single input/output 8-bit data register
- Input buffer to avoid bus stall during calculation
- 18KB ROM calculating time is 1.2ms.

23.1 CRC Calculation Procedure

The CRC generator supports 2 calculation modes to obtain CRC code including ROM auto-calculation mode and data calculation mode.

ROM auto-calculation mode procedure:

- Support select CRC execute range by CRCROM[1:0] bits.
 - If CRCROM[1:0] = 00, the CRC executes range is 0x0000~0x11FF.
 - If CRCROM[1:0] = 01, the CRC executes range is 0x0000~0x23FF.
 - If CRCROM[1:0] = 10, the CRC executes range is 0x0000~0x467F.
 - If CRCROM[1:0] = 11, the CRC executes range is 0x0000~0x477D.
- Select CRC polynomial by CRCPOL bit. When CRCPOL=0, CRC-16-CCITT ($X^{16}+X^{12}+X^5+1$) is selected. Otherwise, the CRC-16($X^{16}+X^{15}+X^2+1$) is selected.
- Set CRCRST bit to reset initial seed value/ CRC data buffer and BUSY bit to 0.
- Set URRCEN bit to start CRC calculation and check BUSY bit.
- CRC calculation is finished until BUSY bit from 1 to 0. Read CRC code from CRCDATA/L.

Data calculation mode procedure:

- Select CRC polynomial by CRCPOL bit. When CRCPOL=0, CRC-16-CCITT ($X^{16}+X^{12}+X^5+1$) is selected. Otherwise, the CRC-16($X^{16}+X^{15}+X^2+1$) is selected.

- Set CRCRST bit to reset initial seed value/ CRC result and BUSY bit to 0.
- Key in data to CRCDATL.
- Set URRCEN bit to start CRC calculation and check BUSY bit.
- CRC calculation is finished until BUSY bit from 1 to 0. Read CRC code from CRCDATH/L

23.2 CRC Registers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCM	URRCEN	BUSY	CRCRST	CRCPOL	CRCROM1	CRCROM0	-	CRCEN
CRCDATL	CRCDAT7	CRCDAT6	CRCDAT5	CRCDAT4	CRCDAT3	CRCDAT2	CRCDAT1	CRCDAT0
CRCDATH	CRCDAT15	CRCDAT14	CRCDAT13	CRCDAT12	CRCDAT11	CRCDAT10	CRCDAT9	CRCDAT8

CRCM Register (0xF1)

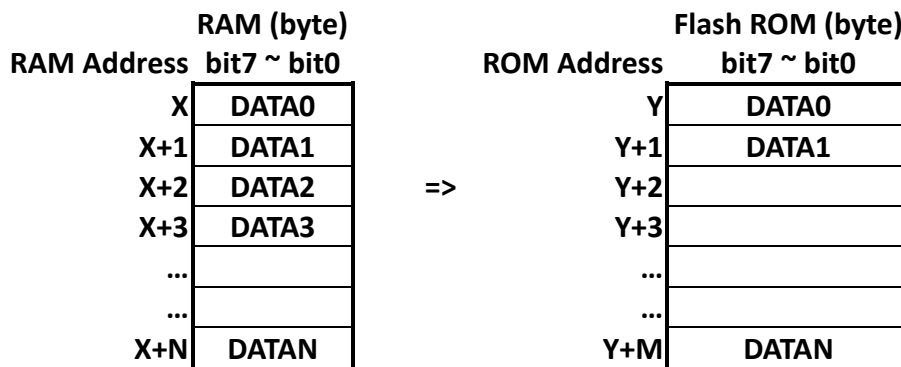
Bit	Field	Type	Initial	Description
7	URRCEN	R/W	0	Enable bit of the CRC calculation for the User ROM. 0: Disable 1: Enable
6	BUSY	R/W	0	Disable. Stop/Finish operation 0: CRC calculation finished. 1: CRC calculation is in process.
5	CRCRST	R/W	0	Reset bit. 0: No effect. 1: Reset the CRC calculation circuit.
4	CRCPOL	R/W	0	CRC Polynomial 0: CRC-16-CCITT 1: CRC-16
3..2 [1:0]	CRCROM	R/W	0x00	CRC EEPROM calculation range. 00: 0x0000~0x11FF (1/4 ROM). 01: 0x0000~0x23FF (1/2 ROM). 10: 0x0000~0x467F (Reserve last 3 page) 11: 0x0000~0x477D (Reserve last 1 page + 2bytes)
0	CRCEN	R/W	0	Enable bit of the CRC IP 0: Disable. 1: Enable.

CRCDATH/CRCDATL Registers (CRCDATH: 0xF3, CRCDATL: 0xF2)

Bit	Field	Type	Initial	Description
15..0	CRCDAT[15:0]	R/W	0x0000	CRCDAT[15:0] 16-bit CRC data.

24 In-System Program

The SN8F5602 MCU integrated device feature in-system programmable (ISP) FLASH memory for convenient, upgradeable code storage. The FLASH memory may be programmed via the SONiX 8051 MCU programming interface or by application code. The SN8F5602 provides security options at the disposal of the designer to prevent unauthorized access to information stored in FLASH memory. ISP Flash ROM provided user an easy way to storage data into Flash ROM. The ISP concept is memory mapping idea that is to move RAM buffer to flash ROM. Choice ROM/RAM address and executing ROM programming command – PECMD, after programming words which controlled by PERAMCNT, PERAML/PERAMCNT data will be programmed into address PEROML/PEROMH.



During Flash program or erase operation, the MCU is stalled, although peripherals (Timers, WDT, I/O, PWM, etc.) remain active. When PECMD register is set to execute ISP program and erase operations, the program counter stops, op-code can't be dumped from flash ROM, instruction stops operating, and program execution is hold not to active. At this time hardware depends on ISP operation configuration to do flash ROM erasing and flash ROM programming automatically. After ISP operation is finished, hardware releases system clock to make program counter running, system returns to last operating mode, and the next instruction is executed. Recommend to add two "NOP" instructions after ISP operations.

- ISP flash ROM erase time = 5ms.....1-page, 128-byte.
- ISP flash ROM program time = 35us.....1- byte.
- ...
- ISP flash ROM program time = 530us.....64- byte.

24.1 Erase

ISP flash ROM erase operation is to clear flash ROM contents to blank status "1". Erasing ROM length is 128-byte and has ROM page limitation. ISP flash ROM erase ROM map is as following:

ISP ROM MAP		ROM address bit0~bit8 (hex) =0
ROM address bit9~bit15 (hex)	0000	These pages include reset vector and interrupt sector. We strongly recommend to reserve the area not to do ISP erase.
	0100	One ISP Erase Page
	0180	One ISP Erase Page
	...	One ISP Erase Page
	3000	One ISP Erase Page
	3080	One ISP Erase Page
		One ISP Erase Page
	4700	One ISP Erase Page
	4780	This page includes ROM reserved area. We strongly recommend to reserve the area not to do ISP erase.

ISP flash ROM erase density is 128- byte which limits erase page boundary. The first 256- byte of flash ROM (0x0000~0x00FF) includes reset vector and interrupt vectors related essential program operation, and the last page 128- byte of flash ROM (0x4780~0x47FF) includes system reserved ROM area, we strongly recommend do not execute ISP flash ROM erase operation in the two pages. Flash ROM area 0x0100~0x477F includes 141-page for ISP flash ROM erase operation.

The first step to do ISP flash ROM erase is to address ROM-page location. The address must be the head location of a page area, e.g. 0x0100, 0x0180...0x4600, 0x4680 and 0x4700. PEROML [7:0] and PEROMH [7:0] define the target starting address [15:0] of flash ROM. Write the start address into PEROML and PEROMH registers, set PECMD register to "0xC3", and the system start to execute ISP flash ROM erase operation.

24.2 Page Program

ISP flash ROM program operation is to write data into flash ROM by program. Program ROM doesn't limit written ROM address and length, but limits 64- byte density of one page. The number of ISP flash ROM program operation can be 1- byte ~ 64- byte at one time, but these words must be in the same page. ISP flash ROM program ROM map is as following:

ISP ROM MAP		ROM address bit0~bit6 (hex) =0
ROM address bit6~bit15 (hex)	0000	These pages include reset vector and interrupt sector. We strongly recommend to reserve the area not to do ISP erase.
	0040	
	0080	
	00C0	
	0100	One ISP Program Page
	0140	One ISP Program Page
	...	One ISP Program Page
	4700	One ISP Program Page
	4740	One ISP Program Page

	4780	This page includes ROM reserved area. We strongly recommend to reserve the area not to do ISP erase.
	47C0	

ISP flash ROM program page density is 64- byte which limits program page boundary. The first 256- byte of flash ROM (0x0000~0x00FF) includes reset vector and interrupt vectors related essential program operation, and the last page 128- byte of flash ROM (0x4780~0x47FF) includes system reserved ROM area, we strongly recommend do not execute ISP flash ROM program operation in the two pages. Flash ROM area 0x0100~0x477F includes 141-page for ISP flash ROM program operation.

ISP flash ROM program operation is a simple memory mapping operation. The first step is to plan a RAM area to store programmed data and keeps the RAM address for ISP RAM addressing. The second step is to plan a ROM area will be programmed from RAM area in ISP flash ROM program operation. The RAM addressing is through PERAML [7:0] 8-bit buffer to configure the start RAM address.

ISP programming length is 1- byte ~64- byte. ISP flash ROM programming length is controlled by PERAMCNT [5:0] bits which is 6-bit format. Before ISP ROM programming execution, set the length by program. PEROML [7:0] and PEROMH [7:0] define the target starting address [15:0] of flash ROM. Write the start address into PEROML and PEROMH registers, set PECMD register to "0xC3" and the system start to execute ISP flash ROM erase operation, set PECMD register to "0x5A" and the system start to execute ISP flash ROM program operation. If the programming length is over ISP flash ROM program page boundary, the hardware immediately stops programming flash ROM after finishing programming the last word of the ROM page. So it is very important to plan right ROM address and programming length.

These configurations must be setup completely before starting Page erase and program. ISP is configured using the following steps:

1. Save program data into IRAM. The data continues for 64 bytes.
2. Set the start address of the content location to PERAM.
3. Set the start address of the anticipated update area to PEROM [15:0]. (By PEROMH/PRROML registers)
4. Set the ISP programming length of PERAMCNT.
5. Write '0xC3' into PECMD [7:0] to page erase (128 bytes/page).
6. Write '0x5A' into PECMD [7:0] to page program (max 64 bytes/page).
7. Write 'NOP' instruction twice.
8. Must Write PEROMHH Bit2 = 0, after Data memory program process.

As an example, assume the program memory (IROM, 0x4700 – 0x477F) is the anticipated update area; the content is already stored in IRAM address 0x40 – 0x7F. To perform the in-system program, simply write starting IROM address 0x4700 to PROMH/PROML registers, and then specify buffer

starting address 0x40 to EPRAM register. Set the ISP programming length of PERAMCNT. Subsequently, write '0xC3' into PECMD [7:0] registers and then write '0x5A' into PECMD [7:0] registers to duplicate the buffer's data to IROM.

*** Note:**

- 1. Watch dog timer should be clear before the Flash write (program) operation, or watchdog timer would overflow and reset system during ISP operating.**
- 2. Don't execute ISP flash ROM program operation for the first page, or affect program operation.**
- 3. ISP operation (page program) actually perform Flash erase and program procedures in the background. Don't execute ISP flash ROM program operation in low-voltage condition (ex. VDD < 2.5V), or ISP operation maybe not complete before power-off.**

24.3 In-system Program Register

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PERAM	PERAM7	PERAM6	PERAM5	PERAM4	PERAM3	PERAM2	PERAM1	PERAM0
PEROMH	PEROM15	PEROM14	PEROM13	PEROM12	PEROM11	PEROM10	PEROM9	PEROM8
PEROML	PEROM7	PEROM6	PEROM5	PEROM4	PEROM3	PEROM2	PEROM1	PEROM0
PECMD	PECMD7	PECMD6	PECMD5	PECMD4	PECMD3	PECMD2	PECMD1	PECMD0
PERAMCNT	-	PRRAM6	PRRAM5	PRRAM4	PRRAM3	PRRAM2	PRRAM1	PRRAM0
PEROMHH	-	-	-	-	-	INFBK	-	-

PEROMHH Register (0xBF)

Bit	Field	Type	Initial	Description
2	INFBK	R/W	0	Main Block / Information Block select switch 0: Main Block 1: Information Block
Else	Reserved	R/W	0	

PERAM Register (0xBD)

Bit	Field	Type	Initial	Description
7..0	PERAM[7:0]	R/W	0x00	The first address of data buffer (IRAM)

PEROMH Register (0xBC)

Bit	Field	Type	Initial	Description
7..0	PEROM[15:8]	R/W	0x00	The first address (15 th – 8 th bit) of program page (IROM)

PEROML Register (0xBB)

Bit	Field	Type	Initial	Description
7..0	PEROM[7:0]	R/W	000	The first address (7 th – 0 th bit) of program page (IROM)

PECMD Register (0xBA)

Bit	Field	Type	Initial	Description
7..0	PECMD[7:0]	W	-	0x5A: Start page program procedure (64 bytes/page) 0xC3: Start page erase (128 bytes/page) Else values: Reserved ^{*(1)}

*(1) Not permitted to write any other to PECMD register.

PERAMCNT Register (0xBE)

Bit	Field	Type	Initial	Description
7	Reserved	R	00	
6..0	PERAM[5:0]	R/W	0x00	Defines the number of byte wanted to be programmed 000000: ISP programming length is 1 -byte. 000001: ISP programming length is 2 -byte. ... 111110: ISP programming length is 63 -byte. 111111: ISP programming length is 64 -byte.

25 Clock Fine-Tuning

SN8F5602 builds in clock fine-tuning function that is a procedure to fine-tune system clock frequency by firmware. The function is enabled by code option (CK_Fine_Tuning). When CK_Fine_Tuning = 0, the clock fine-tuning function is disabled. When CK_Fine_Tuning = 1, the clock fine-tuning function is enabled. After system power-on, the 10-bit initial clock trim value will be loaded to FRQ[9:0] buffer by hardware. The trim value corresponds to IHRC 32MHz. Change the trim value of FRQ[9:0] to modify internal clock frequency.

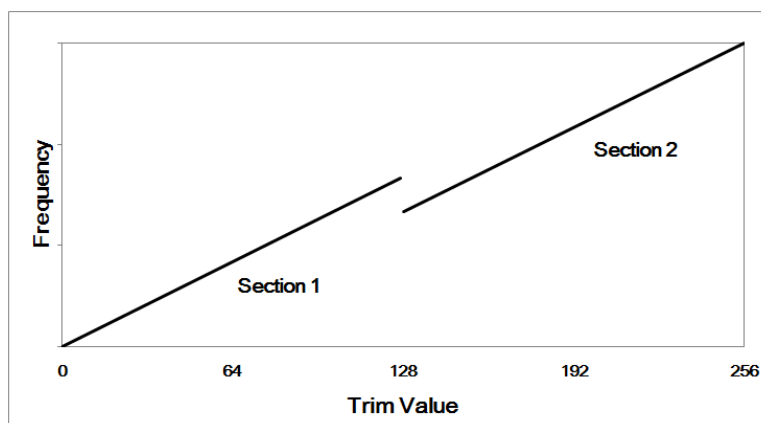
25.1 Clock Trim Section

Clock fine-tuning consists of 8 trim sections as Table 25-1. Each section includes 128 trim steps and each step is about (32MHz *0.1%) in the same section. The larger the Trim value, the faster the frequency.

Table 25-1 Clock Trim Section

Section	Trim Value	Frequency
1	000H ~ 07FH	Low
2	080H ~ 0FFH	
3	100H ~ 17FH	
4	180H ~ 1FFH	
5	200H ~ 27FH	
6	280H ~ 2FFH	
7	300H ~ 37FH	
8	380H ~ 3FFH	High

Each adjacent section has a frequency gap as below. Thus the frequency in trim value =127 will faster than trim value =128.



25.2 Clock Fine-Tuning Procedure

These configurations must be setup completely before starting clock fine-tuning. The steps are as the following:

1. Select code option CK_Fine_Tuning = 1 to enable clock fine-tuning function.
2. As the Max. IROM fetching cycle is 16MHz.
3. Read 10-bit 32MHz trim value from FRQ[9:0]
4. Check the fine-tuning range from the Table 25-1.
5. Write the new clock trim value to FRQ[9:0].
6. Write '0x3C' into FRQCMD [7:0] to trigger clock fine-tuning function.

*** Note: Please check IROM fetching cycle $\leq$ 16MHz to avoid system error.**

25.3 Clock Fine-Tuning Register

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FRQH	-	-	-	-	-	-	FRQ9	FRQ8
FRQL	FRQ7	FRQ6	FRQ5	FRQ4	FRQ3	FRQ2	FRQ1	FRQ0
FRQCMD	FRQCMD7	FRQCMD6	FRQCMD5	FRQCMD4	FRQCMD3	FRQCMD2	FRQCMD1	FRQCMD0

FRQ Register (FRQH:0xAB, FRQL:0xAA)

Bit	Field	Type	Initial	Description
9..0	FRQ[9:0]	R/W	0x00	The system clock calibration value

FRQCMD Register (0xAC)

Bit	Field	Type	Initial	Description
7..0	FRQCMD[7:0]	W	0x00	0x3C: Start clock fine-tuning procedure Else values: Reserved ^{*(1)}

^{*(1)} Not permitted to write any other to FRQCMD register.

26 ILRC Auto-Calibration

26.1 Auto-Calibration Operation

ILRC auto-calibration function is hardware calibration to support 16KHz ILRC. The calibration function is enabled by CALEN bit. After setup CALEN bit, the system starts to calibrate ILRC frequency to 16KHz. Calibration time is about 20ms. The CALEN bit is reset to logic 0 when the calibration is complete. After the calibration is complete, the circuit will set CALDONE bit if ILRC calibration success.

If user want to manually adjust the ILRC frequency, the method is that writes you want trim value to LFRQ[8:0], and then writes 0xB4 to FRQCMD. The LFRQ[8:0] trim value will be loaded into ILRC IP.

26.2 Auto-Calibration Register

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LFRQH	-	-	-	-	-	-	-	LFRQ8
LFRQL	LFRQ7	LFRQ6	LFRQ5	LFRQ4	LFRQ3	LFRQ2	LFRQ1	LFRQ0
FRQCMD	FRQCMD7	FRQCMD6	FRQCMD5	FRQCMD4	FRQCMD3	FRQCMD2	FRQCMD1	FRQCMD0
CLKCAL	CALEN	CALDONE	-	-	-	-	-	-

LFRQ Register (FRQH:0xAE, FRQL:0xAD)

Bit	Field	Type	Initial	Description
15..9	Reserved	R	0x00	
8..0	LFRQ[8:0]	R/W	0x00	The system ILRC calibration value

FRQCMD Register (0xAC)

Bit	Field	Type	Initial	Description
7..0	FRQCMD[7:0]	W	0x00	0x4B: Start ILRC calibration procedure 0xB4: ILRC trim value loaded into ILRC IP Else values: Reserved

CLKCAL Register (0xAF)

Bit	Field	Type	Initial	Description
7	CALEN	R/W	0	ILRC calibration enable bit 0: Disable

				1: Enable (automatically cleared by the end of calibration)
6	CALDONE	R/W	0	ILRC calibration result 0: Calibration fail. 1: Calibration pass, ILRC frequency is 16KHz
5..0	Reserved	R	0x00	

27 Single Wire Asynchronous Interface (SWAT)

SWAT function is a single wire transmission interface for system debug mode.

- Support single-wire debug interface.
- Direct debug access to all memories, registers and peripherals.

27.1 Debug Mode

System debug mode is default enabled. When system power on, SWAT function is enabled and P1.2 acts as SWAT pin for debug interface. User can disable SWAT function by DEGCMD register in order to use P1.2 as GPIO. When DEGCMD= 0x00, SWAT function is disabled and P1.2 acts as GPIO. User can set DEGCMD= 0xA5 to enable SWAT function again. SWAT function can't active under idle/ stop mode. User code must disable SWAT function before system enters idle/ stop mode.

27.2 Internal Pull-Up Resistors on SWAT pin

To avoid any uncontrolled IO levels, the device embeds internal pull-up resistor on SWAT input pin. Once a SWAT function is disabled by DEGCMD register, the GPIO controller takes control again.

27.3 SWAT Register

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DEGCMD	DEGCMD7	DEGCMD6	DEGCMD5	DEGCMD4	DEGCMD3	DEGCMD2	DEGCMD1	DEGCMD0

DEGCMD Register (0xC1)

Bit	Field	Type	Initial	Description
7..0	DEGCMD[7:0]	R/W	0xA5	SWAT pin control 0xA5: Enable SWAT pin. (P1.2 acts as SWAT pin and internal pull-up is enabled) 0x00: Disable SWAT pin. (P1.2 acts as GPIO pin and internal pull-up is disabled) Else values: Reserved ^{*(1)}

^{*(1)} Not permitted to write any other to DEGCMD register.

28 Electrical Characteristics

28.1 Absolute Maximum Ratings

Voltage applied at VDD to VSS	- 0.3V to 6.0V
Voltage applied at any pin to VSS.....	- 0.3V to VDD+0.3V
Operating ambient temperature.....	-40°C to 105°C
Storage ambient temperature	-40°C to 125°C

28.2 System Operation Characteristics

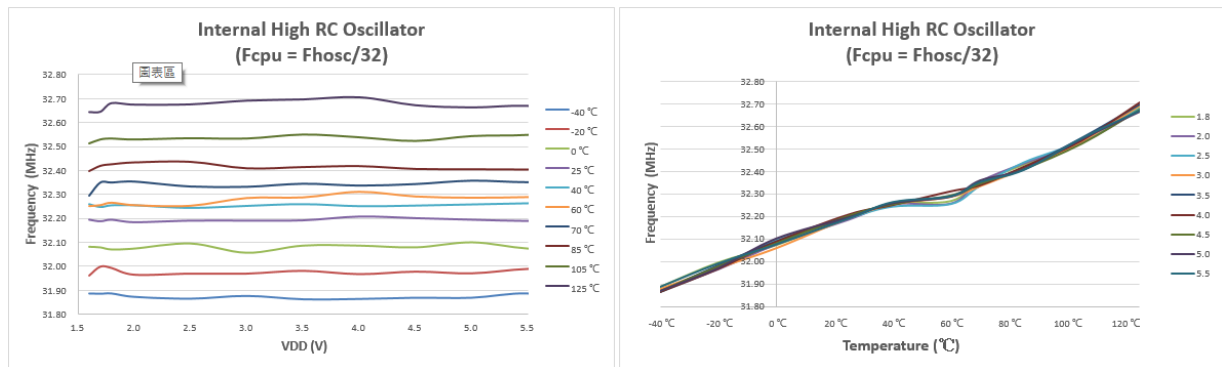
	Parameter	Test Condition	Min	TYP	MAX	UNIT
VDD	Operating voltage	Fcpu = 1MHz	1.8		5.5	V
V _{DR}	RAM data retention Voltage		0.55			V
V _{POR}	VDD rising rate*		0.05			V/ms
I _{DD1}	Normal mode supply current (32MHz IHRC)	VDD = 3V, Fcpu = 1MHz		0.55		mA
		VDD = 5V, Fcpu = 1MHz		0.60		mA
		VDD = 3V, Fcpu = 4MHz		1.35		mA
		VDD = 5V, Fcpu = 4MHz		1.40		mA
		VDD = 3V, Fcpu = 16MHz		2.00		mA
		VDD = 5V, Fcpu = 16MHz		2.05		mA
I _{DD2}	STOP mode supply current	VDD = 3V		1.4	3.8	μA
		VDD = 5V		1.6	4.0	μA
I _{DD3}	STOP mode supply current (ILRC enable STWK=1)	VDD = 5V		2.2		μA
I _{DD5}	IDLE mode supply current (Fcpu = 1MHz)	VDD = 3V, 32MHz IHRC		0.50		mA
		VDD = 5V, 32MHz IHRC		0.55		mA
F _{IHRC}	Internal high clock generator	VDD = 1.8V to 5.5V, 25°C	-0.5%	32	+0.5%	MHz
		VDD = 1.8V to 5.5V, -40°C to 105°C	-5.0%	32	+5.0%	MHz
		VDD = 1.8V to 5.5V, 25°C Non-trimmed	-50%	16	+50%	KHz
F _{ILRC}	Internal low clock generator*	VDD = 1.8V to 5.5V, 25°C Trimmed in normal mode	-3.0%	16	+3.0%	KHz
		VDD = 1.8V to 5.5V, 25°C Trimmed in STOP mode	-8.0%	16	+8.0%	KHz
V _{LVD}	LVD detect voltage	25°C	1.6	1.7	1.8	V
			2.0	2.1	2.2	V

		2.4	2.5	2.6	V
		2.8	2.9	3.0	V
		3.2	3.3	3.4	V
		3.6	3.7	3.8	V
		4.0	4.1	4.2	V
		4.4	4.5	4.6	V
V_{ESD_HBM}	ESD human body mode	4000			V
V_{ESD_CDM}	ESD charged device model	750			V

* Parameter(s) with star mark are non-verified design reference. Ambient temperature is 25°C.

● IHRC Frequency - Temperature Graph

The IHRC Graphs are for design guidance, not tested or guaranteed. In some graphs, the data presented are outside specified operating range. This is for information only and devices are guaranteed to operate properly only within the specified range.



28.3 GPIO Characteristics

	Parameter	Test Condition	Min	TYP	MAX	UNIT
V_{IL}	Low-level input voltage		VSS		0.3VDD	V
V_{IH}	High-level input voltage		0.7VDD		VDD	V
I_{LEKG}	I/O port input leakage current	$V_{IN} = VDD$			2	μA
R_{UP}	Pull-up resister	VDD = 3V	80	100	120	k Ω
		VDD = 5V	48	60	72	k Ω
I_{OH1}	I/O output source current	VDD = 5V, $V_O = VDD - 0.5V$	10	15		mA
I_{OL1}	I/O sink current	VDD = 5V, $V_O = VSS + 0.5V$	10	15		mA
I_{OL2}	I/O sink current	VDD = 5V, $V_O = VSS + 0.5V$, P0.1	8	15		mA

* Ambient temperature is 25°C.

28.4 ADC Characteristics

	Parameter	Test Condition	Min	TYP	MAX	UNIT
V_{ADC}	Operating voltage		2.0		5.5	V
V_{AIN}	AIN channels input voltage	$V_{DD} = 5V$	0		V_{REFH}	V
V_{REFH}	AVREFH pin input voltage	$V_{DD} = 5V$	2.0		V_{DD}	V
	Internal VDD reference voltage	$V_{DD} = 5V$		V_{DD}		V
	Internal 4.5V reference voltage	$V_{DD} = 5V$	-1%	4.5	+1%	V
V_{IREF}	Internal 3.5V reference voltage	$V_{DD} = 5V$	-1%	3.5	+1%	V
	Internal 2.5V reference voltage	$V_{DD} = 5V$	-1%	2.5	+1%	V
	Internal 1.5V reference voltage	$V_{DD} = 5V$	-1%	1.5	+1%	V
I_{AD}	ADC current consumption	$V_{DD} = 3V$		0.40		mA
		$V_{DD} = 5V$		0.45		mA
f_{ADCLK}	ADC clock	$V_{DD} = 5V$			16	MHz
f_{ADSMP}	ADC sampling rate	$V_{DD} = 5V$			1	MHz
t_{ADEN}	ADC function enable period	$V_{DD} = 5V$	100			μs
DNL	Differential nonlinearity*	$f_{ADSMP} = 62.5kHz$		± 2		LSB
		$f_{ADSMP} = 250kHz$		± 2		LSB
INL	Integral Nonlinearity*	$f_{ADSMP} = 62.5kHz$		± 2.5		LSB
		$f_{ADSMP} = 250kHz$		± 2.5		LSB
NMC	No missing code*	$f_{ADSMP} = 62.5kHz$	10	11	12	Bit
		$f_{ADSMP} = 250kHz$		11		Bit
V_{OFFSET}	Input offset voltage	Non-trimmed	-2	0	2	mV

* Parameters with star mark: $V_{DD} = 5V$, $V_{REFH} = 2.4V$, $25^{\circ}C$.

28.5 OPA Characteristics

	Parameter	Test Condition	MIN	TYP	MAX	UNIT
V_{OPA}	Operating voltage		2.2		5.5	V
I_{OPA}	OPA current consumption	$V_{DD} = 5V$, High Current Mode		1		mA
		$V_{DD} = 5V$, Low Current Mode		300		μA
V_{OFFSET}	Input offset voltage	$V_{cm} = V_{ss}$ before calibration.		5		mV
		$V_{cm} = V_{ss}$ after calibration.		2		mV
$PSRR$	Power supply Rejection Ratio*	$V_{CM} = V_{SS}$		80		dB
$CMRR$	Common mode Rejection Ratio*	$V_{CM} = -0.3V$ to $5V$, $V_{DD} = 5V$		80		dB
V_{CM}	Common mode input range	$V_{DD} = 5V$	V_{SS}		$V_{DD}-1.2$	V
V_{OS}	Output voltage swing	High Current Mode	$V_{SS}+0.3$		$V_{DD}-0.3$	V
		Low Current Mode	$V_{SS}+0.1$		$V_{DD}-0.1$	V

T _{OSR}	Output slew rate	VDD = 5V, V _O rising	5	V/μs
		VDD = 5V, V _O falling	5	V/μs
A _{OL}	Open loop gain*	V _O = 0.2V to VDD-0.2V, V _{CM} = VSS Vin=1V. VDD=5V	80	dB
I _{SC}	Output short current	High Current Mode	±30	mA
		Low Current Mode	±10	mA

* Parameters with star mark are non-verified design reference.

28.6 Comparator Characteristics

CMP0 Characteristics

	Parameter	Test Condition	MIN	TYP	MAX	UNIT
V _{CMP}	Operating Voltage		2.2		5.5	V
I _{CMP}	Current consumption*	VDD = 5V		100		μA
V _{OFFSET}	Input offset voltage*	V _{cm} =V _{ss} before calibration.	-5		5	mV
		V _{cm} =V _{ss} after calibration.	-2		2	mV
T _{RS}	Response time	VDD = 5V, V _O rising			100	ns
		VDD = 5V, V _O falling			100	ns
T _{OSR}	Output slew rate	VDD = 5V, V _O rising			100	ns
		VDD = 5V, V _O falling			100	ns
V _{CMR}	Common mode input voltage	VDD = 5V	VSS		VDD-1.2	V

* Parameters with star mark are non-verified design reference.

CMP1/2/3/4 Characteristics

	Parameter	Test Condition	MIN	TYP	MAX	UNIT
V _{CMP}	Operating Voltage		2.2		5.5	V
I _{CMP}	Current consumption*	VDD = 5V		100		μA
V _{OFFSET}	Input offset voltage*	V _{CM} = 0.5VDD	-5		5	mV
T _{RS}	Response time	VDD = 5V, V _O rising			100	ns
		VDD = 5V, V _O falling			100	ns
T _{OSR}	Output slew rate	VDD = 5V, V _O rising			100	ns
		VDD = 5V, V _O falling			100	ns
V _{CMR}	Common mode input voltage	VDD = 5V	VSS+0.5		VDD-0.5	V
V _{IREF}	Internal 4.5V reference voltage	VDD = 5V	-1%	4.5	+1%	V
	Internal 3.5V reference voltage	VDD = 5V	-1%	3.5	+1%	V

Internal 2.5V reference voltage	VDD = 5V	-1%	2.5	+1%	V
Internal 1.5V reference voltage	VDD = 5V	-1%	1.5	+1%	V

* Parameters with star mark are non-verified design reference.

28.7 Flash Memory Characteristics

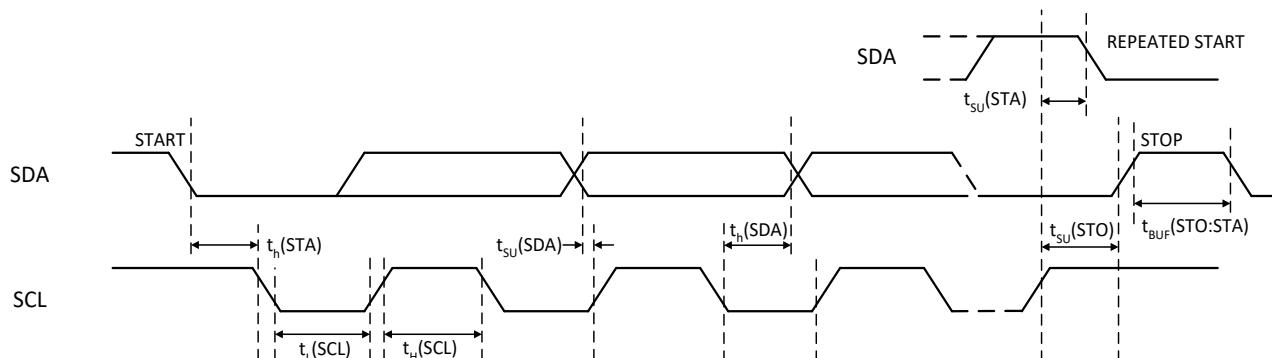
	Parameter	Test Condition	Min	TYP	MAX	UNIT
V _{dd}	Supply voltage		1.8		5.5	V
T _{pen}	Page Endurance time	25°C		*100K		cycle
I _{wrt}	Write current	25°C		3	4	mA
T _{wrt}	Write time	Write 64 bytes, 25°C		0.5	1	ms

* Parameters with star mark are non-verified design reference.

28.8 I2C interface characteristics

Symbol	Parameter	Typical Formula	UNIT
t _H (SCL)	SCL clock high time	2* fsck	μs
t _L (SCL)	SCL clock low time	2* fsck	μs
t _{SU} (SDA)	SDA setup time	2* fsck - 7*Fhosc	μs
t _H (SDA)	SDA data hold time	7*Fhosc	μs
t _H (STA)	START condition hold time	4* fsck	μs
t _{SU} (STA)	Repeated START condition setup time	2* fsck	μs
t _{SU} (STO)	STOP condition setup time	4* fsck + 8*Fhosc	μs
T _{BUF} (STO:STA)	STOP to START condition time	4/3* fsck + 10*Fhosc	μs

*Standard mode I2C fsck = 100KHz, Fast mode I2C fsck = 400KHz, Fhosc=32MHz



29 Instruction Set

This chapter categorizes the SN8F5602 microcontroller's comprehensive assembly instructions. It includes five categories—arithmetic operation, logic operation, data transfer operation, Boolean manipulation, and program branch—which are fully compatible with standard 8051.

Symbol description

Symbol	Description
Rn	Working register R0 - R7
direct	One of 128 internal RAM locations or any Special Function Register
@Ri	Indirect internal or external RAM location addressed by register R0 or R1
#data	8-bit constant (immediate operand)
#data16	16-bit constant (immediate operand)
bit	One of 128 software flags located in internal RAM, or any flag of bit-addressable Special Function Registers
addr16	Destination address for LCALL or LJMP, can be anywhere within the 64-Kbyte page of program memory address space
addr11	Destination address for ACALL or AJMP, within the same 2-Kbyte page of program memory as the first byte of the following instruction
rel	SJMP and all conditional jumps include an 8-bit offset byte. Its range is +127/-128 bytes relative to the first byte of the following instruction
A	Accumulator

Arithmetic operations

Mnemonic	Description
ADD A, Rn	Add register to accumulator
ADD A, direct	Add directly addressed data to accumulator
ADD A, @Ri	Add indirectly addressed data to accumulator
ADD A, #data	Add immediate data to accumulator
ADDC A, Rn	Add register to accumulator with carry
ADDC A, direct	Add directly addressed data to accumulator with carry
ADDC A, @Ri	Add indirectly addressed data to accumulator with carry
ADDC A, #data	Add immediate data to accumulator with carry
SUBB A, Rn	Subtract register from accumulator with borrow
SUBB A, direct	Subtract directly addressed data from accumulator with borrow
SUBB A, @Ri	Subtract indirectly addressed data from accumulator with borrow
SUBB A, #data	Subtract immediate data from accumulator with borrow
INC A	Increment accumulator
INC Rn	Increment register
INC direct	Increment directly addressed location
INC @Ri	Increment indirectly addressed location
INC DPTR	Increment data pointer
DEC A	Decrement accumulator
DEC Rn	Decrement register
DEC direct	Decrement directly addressed location
DEC @Ri	Decrement indirectly addressed location
MUL AB	Multiply A and B
DIV	Divide A by B
DA A	Decimally adjust accumulator

Logic operations

Mnemonic	Description
ANL A, Rn	AND register to accumulator
ANL A, direct	AND directly addressed data to accumulator
ANL A, @Ri	AND indirectly addressed data to accumulator
ANL A, #data	AND immediate data to accumulator
ANL direct, A	AND accumulator to directly addressed location
ANL direct, #data	AND immediate data to directly addressed location
ORL A, Rn	OR register to accumulator

ORL A, direct	OR directly addressed data to accumulator
ORL A, @Ri	OR indirectly addressed data to accumulator
ORL A, #data	OR immediate data to accumulator
ORL direct, A	OR accumulator to directly addressed location
ORL direct, #data	OR immediate data to directly addressed location
XRL A, Rn	Exclusive OR (XOR) register to accumulator
XRL A, direct	XOR directly addressed data to accumulator
XRL A, @Ri	XOR indirectly addressed data to accumulator
XRL A, #data	XOR immediate data to accumulator
XRL direct, A	XOR accumulator to directly addressed location
XRL direct, #data	XOR immediate data to directly addressed location
CLR A	Clear accumulator
CPL A	Complement accumulator
RL A	Rotate accumulator left
RLC A	Rotate accumulator left through carry
RR A	Rotate accumulator right
RRC A	Rotate accumulator right through carry
SWAP A	Swap nibbles within the accumulator

Data transfer operations

Mnemonic	Description
MOV A, Rn	Move register to accumulator
MOV A, direct	Move directly addressed data to accumulator
MOV A, @Ri	Move indirectly addressed data to accumulator
MOV A, #data	Move immediate data to accumulator
MOV Rn, A	Move accumulator to register
MOV Rn, direct	Move directly addressed data to register
MOV Rn, #data	Move immediate data to register
MOV direct, A	Move accumulator to direct
MOV direct, Rn	Move register to direct
MOV direct1, direct2	Move directly addressed data to directly addressed location
MOV direct, @Ri	Move indirectly addressed data to directly addressed location
MOV direct, #data	Move immediate data to directly addressed location
MOV @Ri, A	Move accumulator to indirectly addressed location
MOV @Ri, direct	Move directly addressed data to indirectly addressed location

MOV @Ri, #data	Move immediate data to in directly addressed location
MOV DPTR, #data16	Load data pointer with a 16-bit immediate
MOVC A, @A+DPTR	Load accumulator with a code byte relative to DPTR
MOVC A, @A+PC	Load accumulator with a code byte relative to PC
MOVX A, @Ri	Move external RAM (8-bit address) to accumulator
MOVX A, @DPTR	Move external RAM (16-bit address) to accumulator
MOVX @Ri, A	Move accumulator to external RAM (8-bit address)
MOVX @DPTR, A	Move accumulator to external RAM (16-bit address)
PUSH direct	Push directly addressed data onto stack
POP direct	Pop directly addressed location from stack
XCH A, Rn	Exchange register with accumulator
XCH A, direct	Exchange directly addressed location with accumulator
XCH A, @Ri	Exchange indirect RAM with accumulator
XCHD A, @Ri	Exchange low-order nibbles of indirect and accumulator

Boolean manipulation

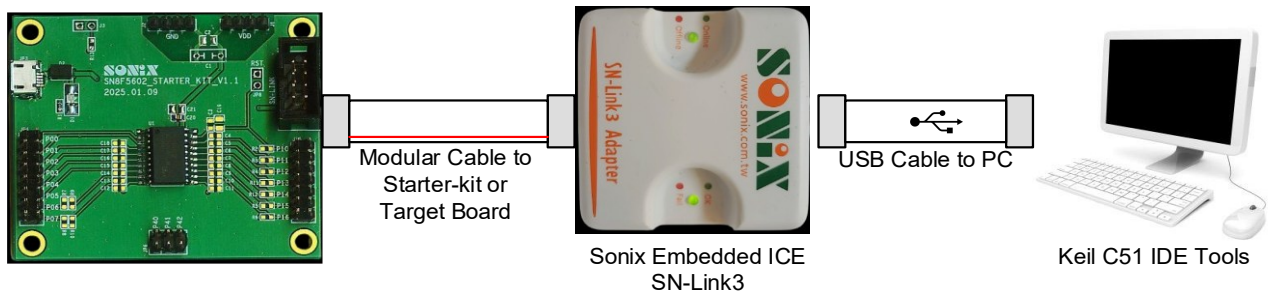
Mnemonic	Description
CLR C	Clear carry flag
CLR bit	Clear directly addressed bit
SETB C	Set carry flag
SETB bit	Set directly addressed bit
CPL C	Complement carry flag
CPL bit	Complement directly addressed bit
ANL C, bit	AND directly addressed bit to carry flag
ANL C, /bit	AND complement of directly addressed bit to carry
ORL C, bit	OR directly addressed bit to carry flag
ORL C, /bit	OR complement of directly addressed bit to carry
MOV C, bit	Move directly addressed bit to carry flag
MOV bit, C	Move carry flag to directly addressed bit

Program branches

Mnemonic	Description
ACALL addr11	Absolute subroutine call
LCALL addr16	Long subroutine call
RET	Return from subroutine
RETI	Return from interrupt
AJMP addr11	Absolute jump
LJMP addr16	Long jump
SJMP rel	Short jump (relative address)
JMP @A+DPTR	Jump indirect relative to the DPTR
JZ rel	Jump if accumulator is zero
JNZ rel	Jump if accumulator is not zero
JC rel	Jump if carry flag is set
JNC rel	Jump if carry flag is not set
JB bit, rel	Jump if directly addressed bit is set
JNB bit, rel	Jump if directly addressed bit is not set
JBC bit, rel	Jump if directly addressed bit is set and clear bit
CJNE A, direct, rel	Compare directly addressed data to accumulator and jump if not equal
CJNE A, #data, rel	Compare immediate data to accumulator and jump if not equal
CJNE Rn, #data, rel	Compare immediate data to register and jump if not equal
CJNE @Ri, #data, rel	Compare immediate to indirect and jump if not equal
DJNZ Rn, rel	Decrement register and jump if not zero
DJNZ direct, rel	Decrement directly addressed location and jump if not zero
NOP	No operation for one cycle

30 Development Environment

SONiX provides an Embedded ICE emulator system to offer SN8F5602 firmware development. The platform is an in-circuit debugger and controlled by Keil C51 IDE software on Microsoft Windows platform. The platform includes SN-Link3, SN8F5602 Starter-kit and Keil C51 IDE software to build a high-speed, low cost, powerful and multi-task development environment including emulator, debugger and programmer. To execute emulation is like run real chip because the emulator circuit integrated in SN8F5602 to offer a real development environment.



30.1 Minimum Requirement

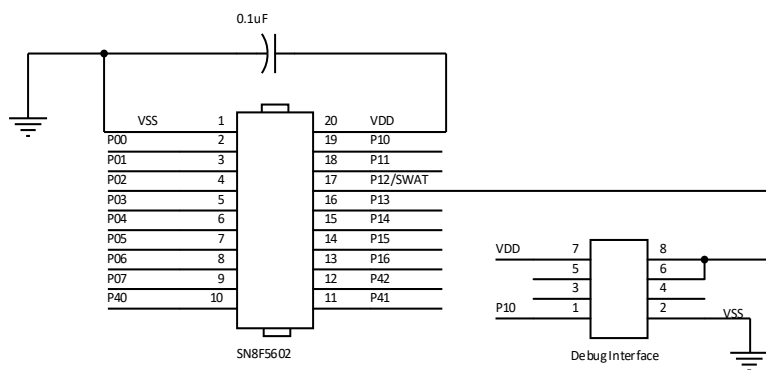
The following items are essential to build up an appropriate development environment. The compatibility is verified on listed versions, and is expected to execute perfectly on later version. SN-Link firmware and software updates and all the latest documentation can be downloaded from the Sonix support webpage at SONiX website (<https://www.sonix.com.tw/article-en-3979-26453>); Keil C51 is downloadable on <https://www.keil.com/download/product/>.

- **SN-Link3 Adapter**
- **SN-Link Driver for Keil C51**
- **Keil C51** version 9.50a or later.

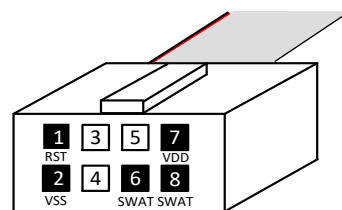
30.2 Debug Interface Hardware

The circuit below demonstrates the appropriate method to connect microcontroller's SWAT pin and SN-Link3 Adapter.

Before starting debug, microcontroller's power (VDD) must be switched off. Connect the SWAT to both 6th and 8th pins of SN-Link, and respectively link VDD and VSS to 7th pin and 2nd pin. A handshake procedure would be automatically started by turn on the microcontroller, and SN-Link's green LED (Run) indicates the success of connection.



example circuit



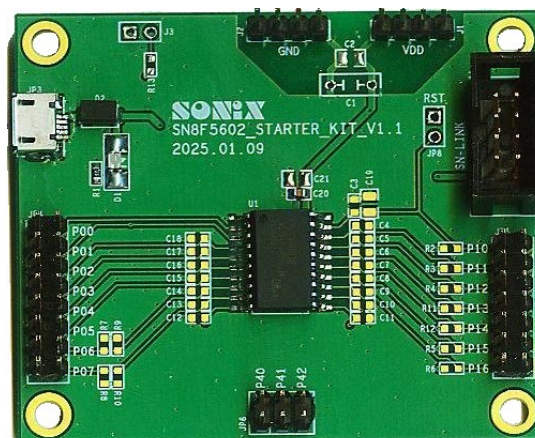
SN-Link header

30.3 Development Tool

SN-Link3 Adapter



Starter-Kit support SN8F5602



MP5 Writer



31 SN8F5602 Starter-Kit

SN8F5000 Starter-Kit provides easy-development platform. It includes SN8F5000 family real chip and I/O connectors to input signal or drive device of user's application. It is a simple platform to develop application as target board not ready. The Starter-Kit can be replaced by target board, because SN8F5000 family integrates embedded ICE in-circuit debugger circuitry.

31.1 Configurations of Circuit

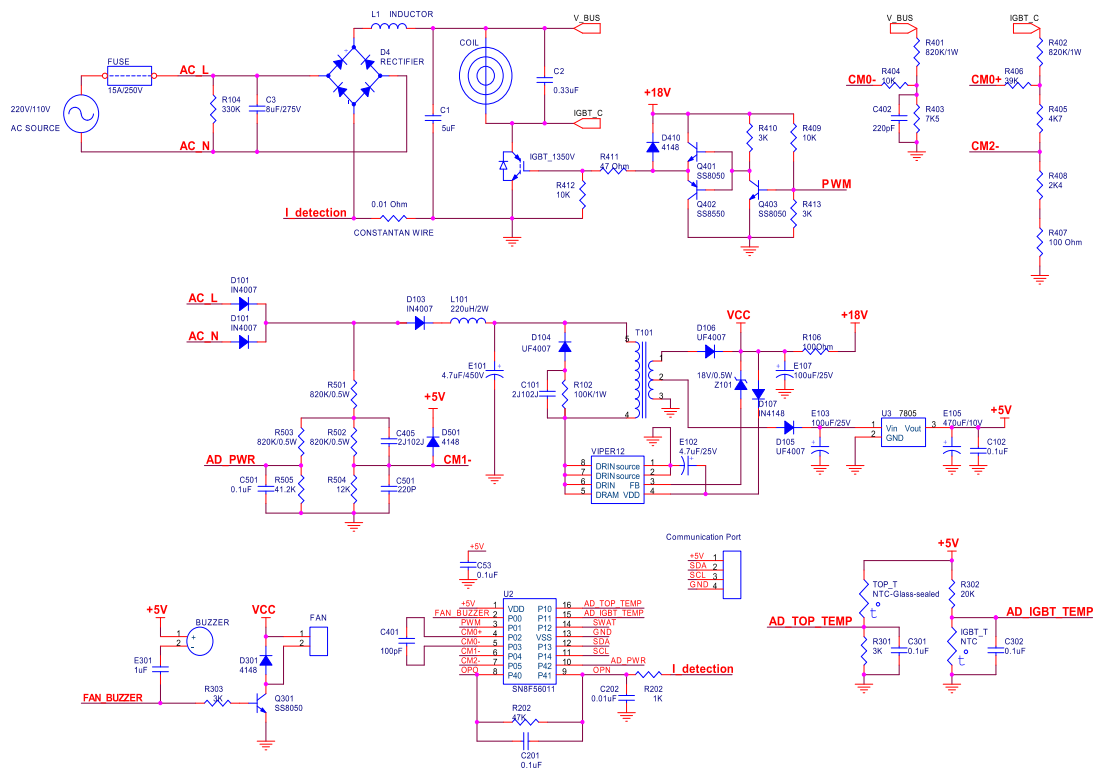
These configurations must be setup completely before starting Starter-Kit developing.

1. Confirm to the circuit board whether elements are complete.
2. The power source of Starter-Kit circuit is chosen from 5.0V, 3.3V, external power or Micro USB via jumper.
3. The power source comes from 5.0V or 3.3V which must be connect to DC 7.5V power adapter.
4. If the power source is chosen from external power, then external power source connects to EXT pin.
5. The "RST" pin needs to connect pull high resister to VDD when external reset is chosen to use.
6. The Debug Port can connect SN-LINK Adapter for emulation or download code.
7. The MCU LED will light up and SN8F5000 family chip will be connected to power when power (VDD) is switched on.

31.4 Component Description

Number	Description
C3 – C18	ADC channel capacitors.
C19	AVREFH capacitor.
C1/C2	Power VDD capacitors
C20-C21	MCU VDD capacitors
D1	MCU direct LED
J11	DC 7.5V power adapter
J1/J2	External power source.
J3	External power source for PWM.
JP3	Micro USB port, VDD power source from USB 5V
JP7/JP8	Debug Port
JP4 – JP6	I/O connector.
R2 – R12	I2C and UART Interface pull-high resisters.
R13	PWM pull-high resister.
U1	SN8F5602 real chip (Sonix standard option).

32 IH Application Circuit



QR_HeatBoard_SN8F56011 V1.1

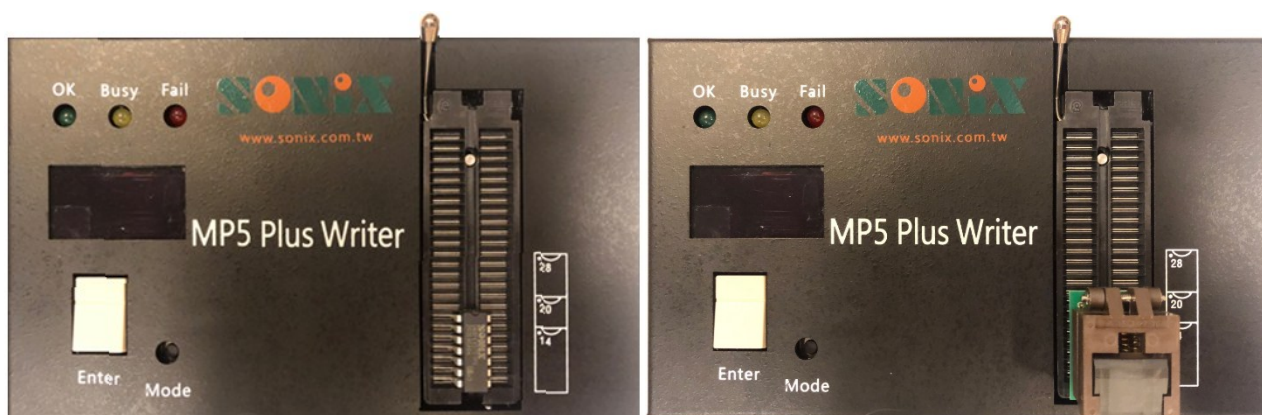
33 ROM Programming Pin

SN8F5602 Series Flash ROM erase/program/verify support SN-Link and MP5 Writer

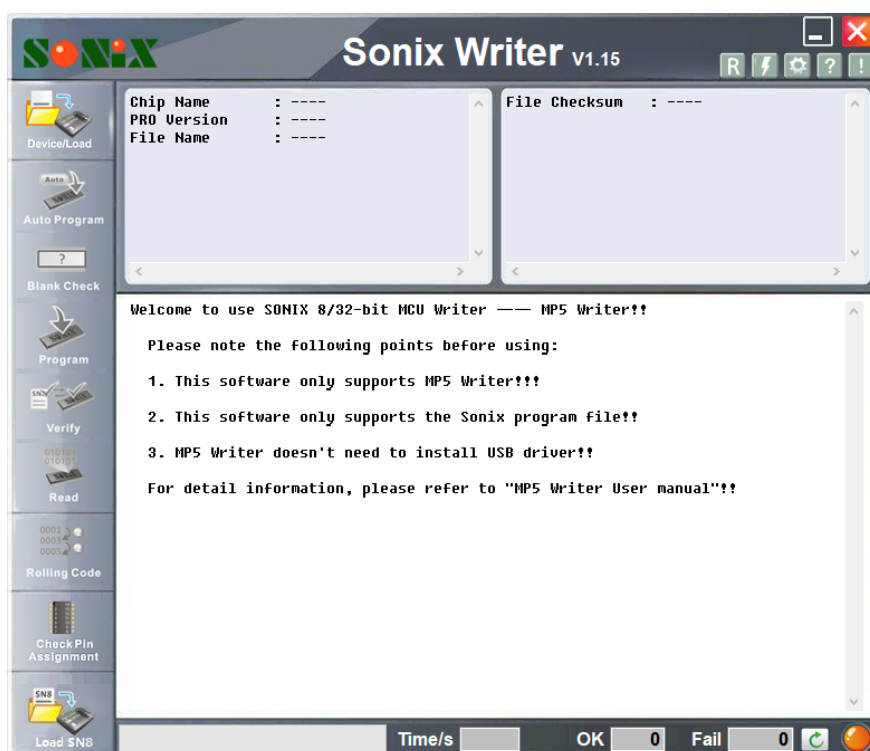
- SN-Link: Debug interface and on board programming.
- MP5 Writer: For SN8F5602 series version mass programming.

33.1 MP5 Hardware Connecting

Different package type with MCU programming connecting is as following, DIP and SOP Illustration.

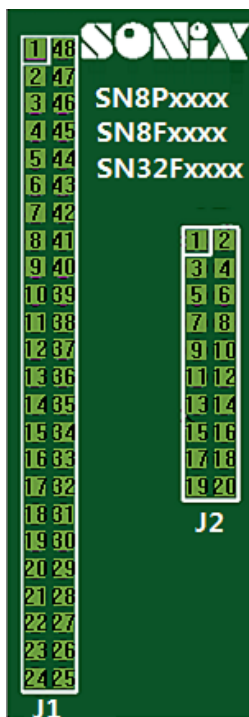


MP5 Software operation interface is as following.



33.2 MP5 Writer Transition Board Socket Pin Assignment

MP5 Writer Transition Board:



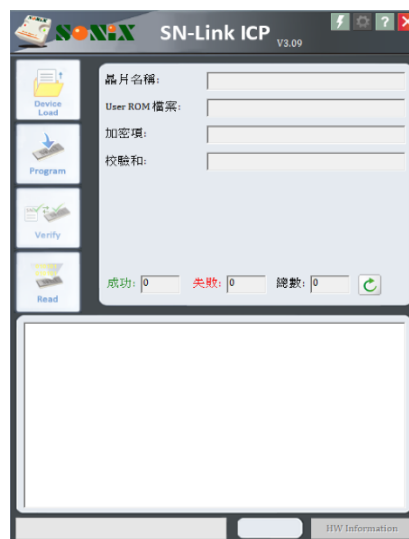
33.3 MP5 Writer Programming Pin Mapping

MP5 Writer programming pins as following.

Writer Connector		MCU Pin Name	SN8F5602P/S		SN8F56011P/S	
J2 Pin Number	J2 Pin Name		MCU Pin Number	J1 Pin Number	MCU Pin Number	J1 Pin Number
1	VDD	VDD	20	34	1	17
2	GND	VSS	1	15	13	29
7, 9	SWAT	P1.2	17	31	14	30
20	PDB	P0.2	4	18	4	20

33.4 SN-Link ICP Programming

SN-Link ICP programming hardware and software are as following.

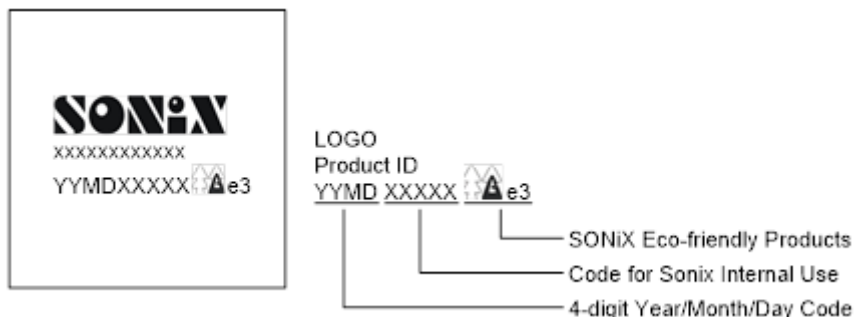


33.5 SN-Link ICP Programming Pin Mapping

SN-Link Connector		MCU Pin Name	SN8F5602P/S	SN8F56011P/S
Pin Number	Pin Name		Pin Number	Pin Number
7	VDD	VDD	20	1
2	GND	VSS	1	13
6, 8	SWAT	P1.2	17	14

34 Ordering Information

The figure below is an example of the marking. Contents such as the product ID or symbol may vary according to different packages.

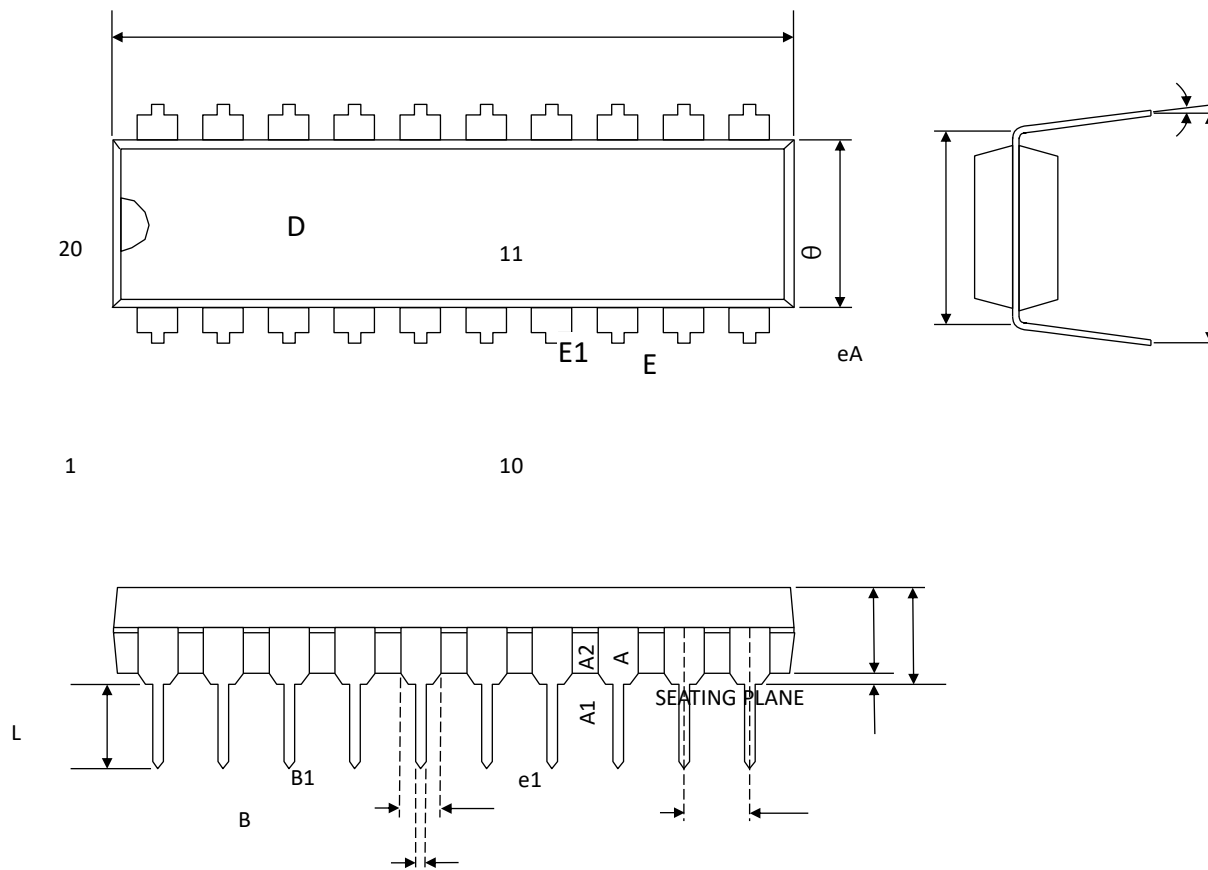


34.1 Device Nomenclature

Full Name	Packing Type
S8F55602W	Wafer
SN8F5602H	Dice
SN8F5602PG	PDIP, 20 pins, Green package
SN8F5602SG	SOP, 20 pins, Green package
SN8F56011PG	PDIP, 16 pins, Green package
SN8F56011SG	SOP, 16 pins, Green package

35 Package Information

35.1 P-DIP20

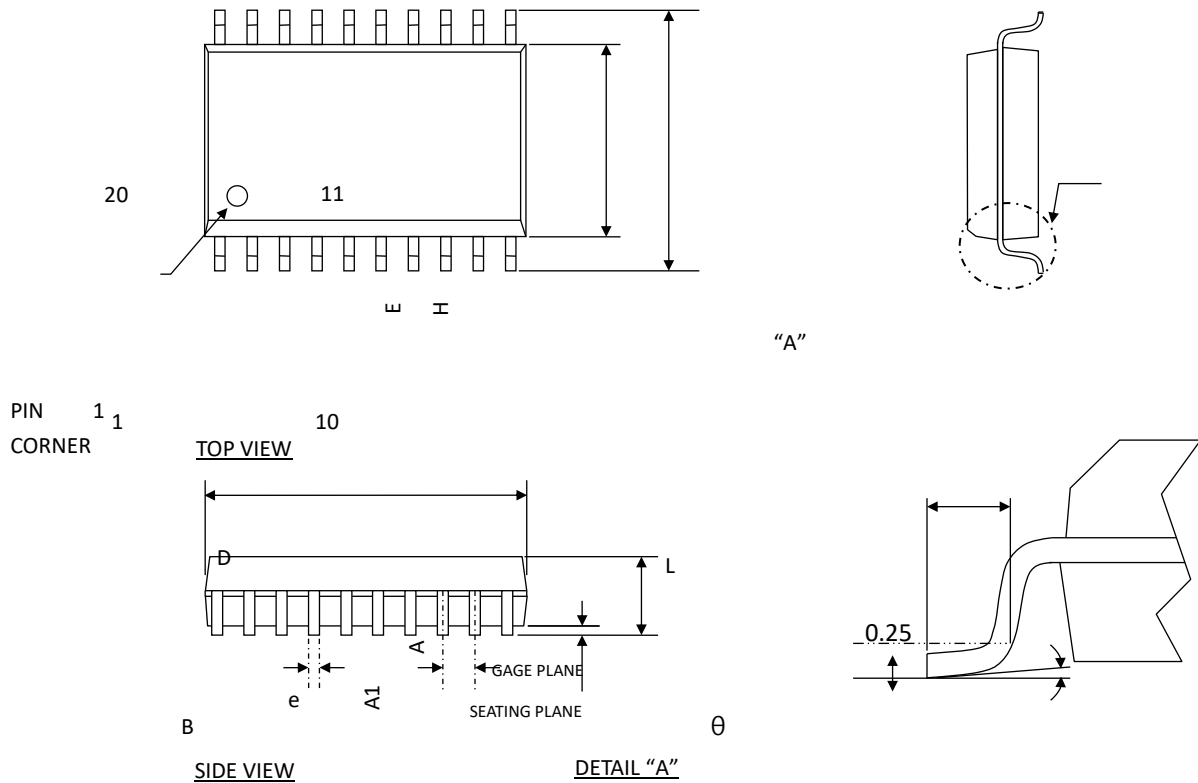


SYMBOLS	Dimension in mm			Dimension in inch		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	--	--	4.45	--	--	0.175
A1	0.35	--	--	0.015	--	--
A2	3.18	3.30	3.43	0.125	0.130	0.135
B	0.46 typ.			0.018 typ.		
B1	1.52 typ.			0.060 typ.		
D	25.70	26.06	26.42	1.012	1.026	1.040
E	7.62 BSC.			0.300 BSC.		
E1	6.05	6.35	6.65	0.238	0.250	0.261
e1	2.54 typ.			0.100 typ.		
L	3.05	3.30	3.56	0.120	0.130	0.140
eA	7.62	9.02	9.53	0.300	0.355	0.375
θ	0°	7°	15°	0°	7°	15°

Notes :

1. JEDEC OUTLINE: MS-001 AD
2. CONTROLLING DIMENSION : inch

35.2 SOP20

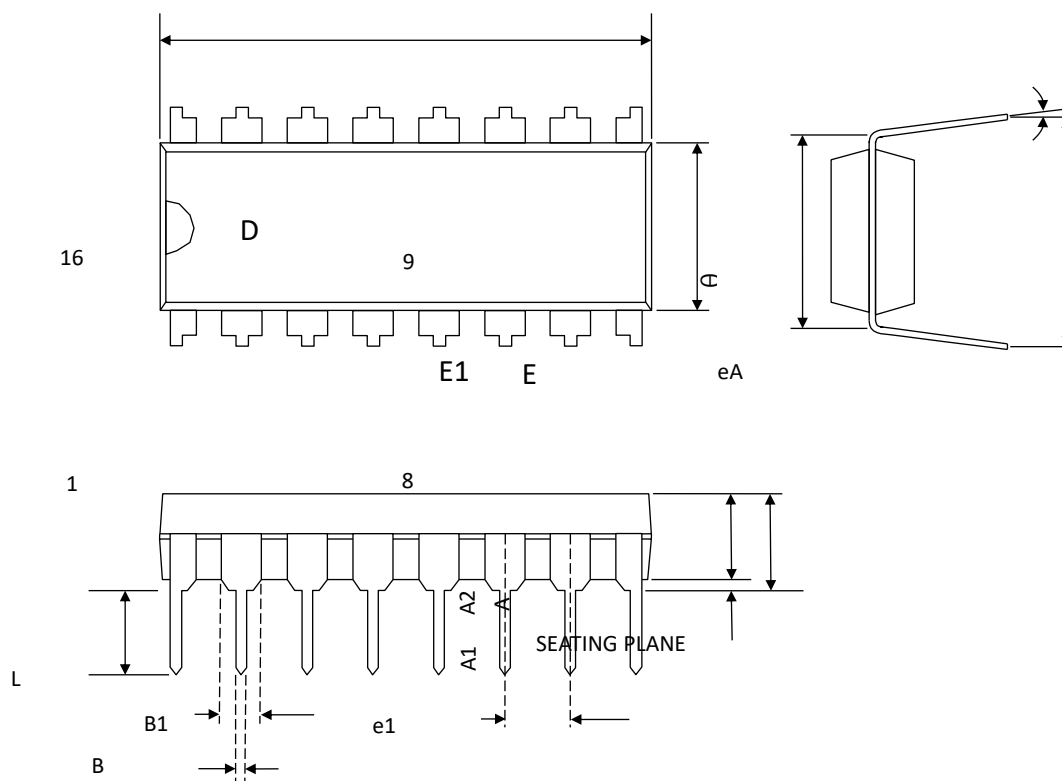


SYMBOLS	Dimension in mm			Dimension in inch		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	--	--	2.65	--	--	0.104
A1	0.10	--	0.30	0.004	--	0.012
B	0.31	0.41	0.51	0.012	0.016	0.020
D	12.80 BSC			0.503		
E	7.50 BSC			0.295		
e	1.27 BSC			0.050 BSC		
H	10.30 BSC			0.405		
L	0.40	--	1.27	0.016	--	0.050
θ	0°	4°	8°	0°	4°	8°

Notes :

1. CONTROLLING DIMENSION : mm
2. JEDEC OUTLINE: MO-013 AC

35.3 P-DIP16

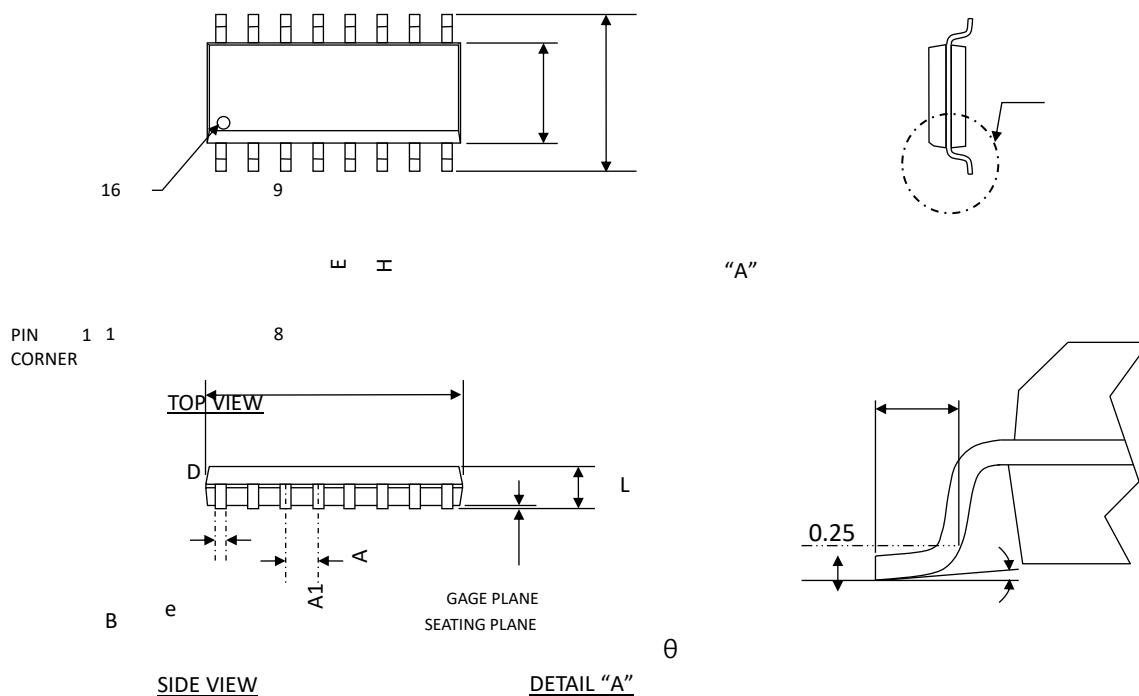


SYMBOLS	Dimension in mm			Dimension in inch		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	--	--	5.33	--	--	0.210
A1	0.38	--	--	0.015	--	--
A2	3.18	3.30	3.43	0.125	0.130	0.135
B	0.46 typ.			0.018 typ.		
B1	1.52 typ.			0.060 typ.		
D	18.67	19.18	19.69	0.735	0.755	0.775
E	7.62 BSC.			0.300 BSC		
E1	6.22	6.35	6.48	0.245	0.250	0.255
e1	2.54 typ.			0.100 typ.		
L	2.92	3.30	3.81	0.115	0.130	0.150
eA	7.62	9.02	9.53	0.300	0.355	0.375
θ	0°	7°	15°	0°	7°	15°

Notes :

1. JEDEC OUTLINE: MS-001 BB
2. CONTROLLING DIMENSION : inch

35.4 SOP16



SYMBOLS	Dimension in mm			Dimension in inch		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	--	--	1.75	--	--	0.069
A1	0.10	--	0.25	0.004	--	0.010
B	0.31	0.41	0.51	0.012	0.016	0.020
D	9.90 BSC			0.389 BSC		
E	3.90 BSC			0.153 BSC		
e	1.27 BSC			0.050 BSC		
H	6.00 BSC			0.236 BSC		
L	0.40	--	1.27	0.016	--	0.050
θ	0°	4°	8°	0°	4°	8°

Notes :

1. CONTROLLING DIMENSION : mm
2. JEDEC OUTLINE: MS-012 AC

SN8F5602 Series Datasheet

8051-based Microcontroller

Corporate Headquarters

10F-1, No. 36, Taiyuan St.
Chupei City, Hsinchu, Taiwan
TEL: +886-3-5600888
FAX: +886-3-5600889

Taipei Sales Office

15F-2, No. 171, Songde Rd.
Taipei City, Taiwan
TEL: +886-2-27591980
FAX: +886-2-27598180
mkt@sonix.com.tw
sales@sonix.com.tw

Hong Kong Sales Office

Unit 2603, No. 11, Wo Shing St.
Fo Tan, Hong Kong
TEL: +852-2723-8086
FAX: +852-2723-9179
hk@sonix.com.tw

Shenzhen Contact Office

26F, Zhongliang Ziyun Building,
Xin'an 2 Road, Bao'an District,
Shenzhen City, China
TEL: +86-755-2671-9666
FAX: +86-755-2671-9786
mkt@sonix.com.tw
sales@sonix.com.tw

USA Office

TEL: +1-714-3309877
TEL: +1-949-4686539
tlightbody@earthlink.net

Japan Office

2F, 4 Chome-8-27 Kudanminami
Chiyoda-ku, Tokyo, Japan
TEL: +81-3-6272-6070
FAX: +81-3-6272-6165
jpsales@sonix.com.tw

FAE Support via email

8-bit Microcontroller Products:
sa1fae@sonix.com.tw
All Products: fae@sonix.com.tw