SH69P21

OTP 1K 4-bit Micro-controller

Features

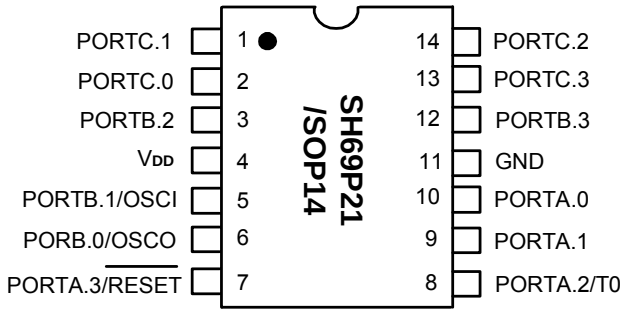
- SH6610C-based single-chip 4-bit micro-controller
- OTP ROM: 1K X 16 bits
- RAM: 88 X 4 bits
  - 24 System control register
  - 64 Data memory
- Operation Voltage: 2.4V - 5.5V
  - fosc = 30kHz - 4MHz, V_{DD} = 2.4V - 5.5V
  - fosc = 4MHz - 8MHz, V_{DD} = 4.5V - 5.5V
- 11CMOS bi-directional I/O pins and 1CMOS input pin
- 4-Level Stack (Including Interrupts)
- One 8-bit auto re-loaded Timer/Counter
- Warm-up Timer
- Powerful Interrupt Sources:
  - External interrupt0: PORTA.0 (Falling/Rising/Double Edge)
  - Timer0 interrupt (Timer0)
  - External Interrupts: PORTB (Falling Edge)
- Oscillator: (Code Option)
  - Crystal oscillator: 32.768kHz, 400kHz - 8MHz
  - Ceramic resonator: 400kHz - 8MHz
  - External RC oscillator: 400kHz - 8MHz
  - Internal RC oscillator: 8MHz
- Instruction Cycle Time (4/fosc)
- Two Low Power Operation Modes: HALT And STOP
- Reset
  - Built-in Watchdog Timer (WDT) (Code Option)
  - Built-in Power-on Reset (POR)
  - Built-in Low Voltage Reset (LVR)
- Two-level low voltage reset (LVR) (Code Option)
- SOP 14/8, SSOP10 Package

General Description

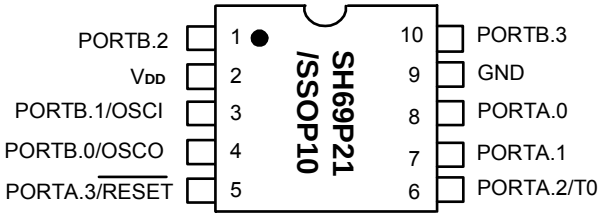
SH69P21 is a single-chip 4-bit micro-controller. This device integrates a SH6610C CPU core, RAM, ROM, PWM, Timer, I/O ports and 8MHz internal RC. The SH69P21 is suitable for alarm or home appliance application.

Pin Configuration

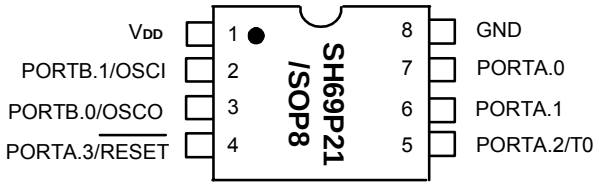
14 PIN



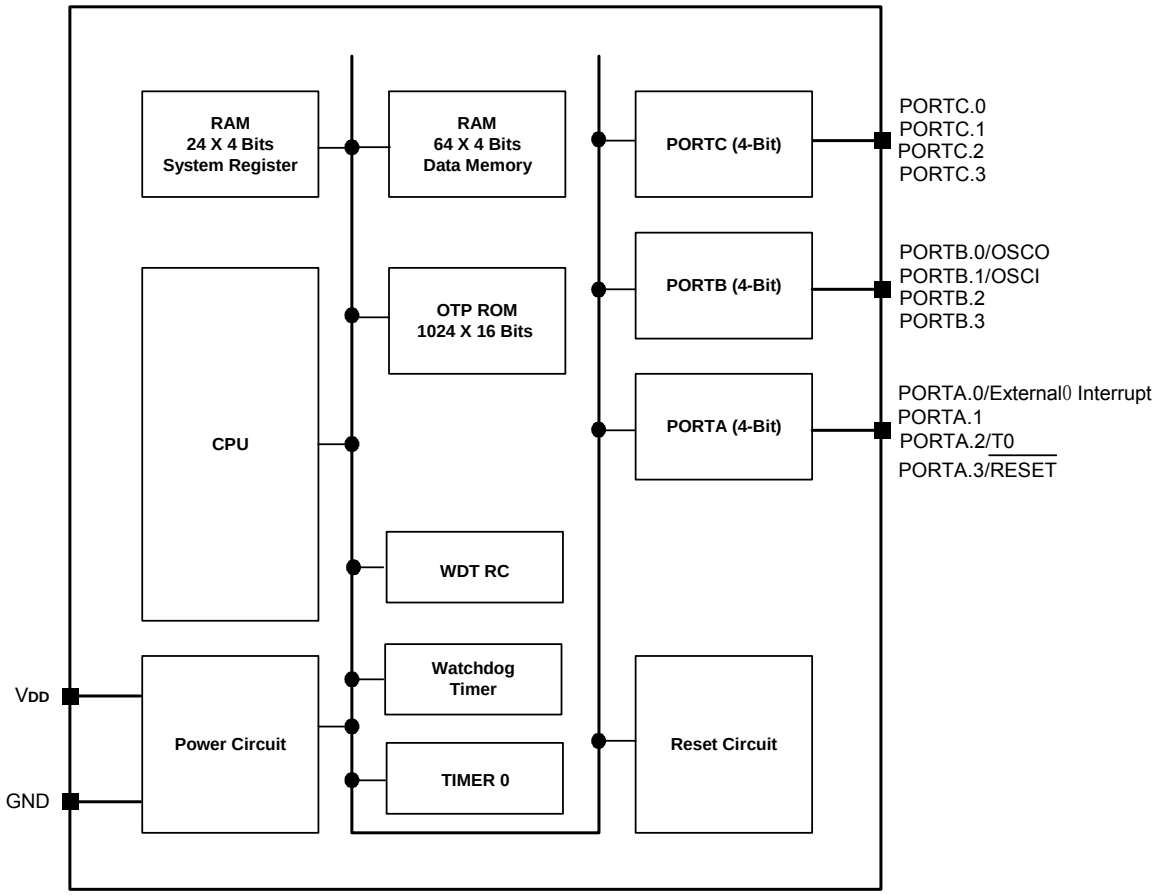
10 PIN



8 PIN



Block Diagram



Pin Description

Pin No.			Pin Name	I/O	Descriptions
14-Pin	10-Pin	8-Pin			
11	9	8	GND	P	Ground pin
4	2	1	V_{DD}	P	Power supply pin
6	4	3	PORTB.0 /OSCO	I/O O	Bit programmable I/O OSC output pin. No output in RC mode
5	3	2	PORTB.1 /OSCI	I/O I	Bit programmable I/O OSC input pin, connected to a crystal, ceramic or external resistor
13, 14, 1, 2	-	-	PORTC.3-0	I/O	Bit programmable I/O
7	5	4	PORTA.3 /$\overline{\text{RESET}}$	I/O I	Bit programmable I/O (open drain) Reset input. (active low, Schmitt trigger input)
8	6	5	PORTA.2 /T0	I/O I	Bit programmable I/O Timer Clock/Counter input pin. (Schmitt trigger input) Shared with PORTC.2
9, 10	7, 8	6, 7	PORTA.1 PORTA.0	I/O I/O I	Bit programmable I/O Bit programmable I/O Vector Interrupt (Active rising/falling/double edge by system register setup)
12, 3	10, 1	-	PORTB.3-2	I/O	Bit programmable I/O

Where, I: Input; O: Output; P: Power; Z: High impedance

OTP Programming Pin Description (OTP Program Mode)

Pin No.			Pin Name	I/O	Sharing Pin	Descriptions
14-Pin	10-Pin	8-Pin				
4	2	1	V_{DD}	P	V_{DD}	Programming Power supply (+5.5V)
7	5	4	V_{PP}	P	$\overline{\text{RESET}}$ /PORTA.3	Programming high voltage Power supply (+11V)
11	9	8	GND	P	GND	Ground
5	3	2	SCK	I	OSCI /PORTB.1	Programming Clock input pin
10	8	7	SDA	I/O	PORTA.0	Programming Data pin

Where, I: Input; O: Output; P: Power; Z: High impedance

Function Description

1. CPU

The CPU contains the following functional blocks: Program Counter (PC), Arithmetic Logic Unit (ALU), Carry Flag (CY), Accumulator, Table Branch Register, Data Pointer (INX, DPH, DPM, and DPL) and Stacks.

1.1. PC

The PC is used for ROM addressing consisting of 12-bit: Page Register (PC11), and Ripple Carry Counter (PC10, PC9, PC8, PC7, PC6, PC5, PC4, PC3, PC2, PC1, PC0).

The program counter is loaded with data corresponding to each instruction. The unconditional jump instruction (JMP) can be set at 1-bit page register for higher than 2K.

The program counter can address only 4K program ROM. (Refer to the ROM description).

1.2. ALU and CY

The ALU performs arithmetic and logic operations. The ALU provides the following functions:

Binary addition/subtraction (ADC, ADCM, ADD, ADDM, SBC, SBCM, SUB, SUBM, ADI, ADIM, SBI, SBIM)
Decimal adjustments for addition/subtraction (DAA, DAS)
Logic operations (AND, ANDM, EOR, EORM, OR, ORM, ANDIM, EORIM, ORIM)
Decisions (BA0, BA1, BA2, BA3, BAZ, BNZ, BC, BNC)
Logic Shift (SHR)

The Carry Flag (CY) holds the ALU overflow that the arithmetic operation generates. During an interrupt service or CALL instruction, the carry flag is pushed into the stack and recovered from the stack by the RTNI instruction. It is unaffected by the RTNW instruction.

1.3. Accumulator (AC)

The accumulator is a 4-bit register holding the results of the arithmetic logic unit. In conjunction with the ALU, data is transferred between the accumulator and system register, or data memory can be performed.

1.4. Table Branch Register (TBR)

Table Data can be stored in program memory and can be referenced by using Table Branch (TJMP) and Return Constant (RTNW) instructions. The TBR and AC are placed by an offset address in program ROM. TJMP instruction branch into address ((PC11 - PC8) X (2^8) + (TBR, AC)). The address is determined by RTNW to return look-up value into (TBR, AC). ROM code bit7 - bit4 is placed into TBR and bit3-bit0 into AC.

1.5. Data Pointer

The Data Pointer can indirectly address data memory. Pointer address is located in register DPH (3-bit), DPM (3-bit) and DPL (4-bit). The addressing range is 000H - 3FFH. Pseudo index address (INX) is used to read or write Data memory, then RAM address bit9 - bit0 which comes from DPH, DPM and DPL.

1.6. Stack

The stack is a group of registers used to save the contents of CY & PC (11-0) sequentially with each subroutine call or interrupt. The MSB is saved for CY and it is organized into 13 bits X 4 levels. The stack is operated on a first-in, last-out basis and returned sequentially to the PC with the return instructions (RTNI/RTNW).

Note:

The stack nesting includes both subroutine calls and interrupts requests. The maximum allowed for subroutine calls and interrupts are 4 levels. If the number of calls and interrupt requests exceeds 4, then the bottom of stack will be shifted out, that program execution may enter an abnormal state.

2. RAM

Built-in RAM contains general-purpose data memory and system register. Because of its static nature, the RAM can keep data after the CPU entering STOP or HALT.

2.1. RAM Addressing

Data memory and system register can be accessed in one instruction by direct addressing. The following is the memory allocation map:

System Register: $000 - $01F
Data Memory: $020 - $05F

2.2. Configuration of System Register

Address	Bit3	Bit2	Bit1	Bit0	R/W	Remarks
$00	IEX0	IET0	-	IEP	R/W	Interrupt enable flags register
$01	IRQX0	IRQT0	-	IRQP	R/W	Interrupt request flags register
$02	-	T0M.2	T0M.1	T0M.0	R/W	Bit2-0: Timer0 Mode register
$03	-	-	-	-	-	Reserved
$04	T0L.3	T0L.2	T0L.1	T0L.0	R/W	Timer0 load/counter low nibble register
$05	T0H.3	T0H.2	T0H.1	T0H.0	R/W	Timer0 load/counter high nibble register
$06 - $07	-	-	-	-	-	Reserved
$08	PA.3	PA.2	PA.1	PA.0	R/W	PORTA data register
$09	PB.3	PB.2	PB.1	PB.0	R/W	PORTB data register
$0A	PC.3	PC.2	PC.1	PC.0	R/W	PORTC data register
$0B - $0D	-	-	-	-	-	Reserved
$0E	TBR.3	TBR.2	TBR.1	TBR.0	R/W	Table Branch Register
$0F	INX.3	INX.2	INX.1	INX.0	R/W	Pseudo index register
$10	DPL.3	DPL.2	DPL.1	DPL.0	R/W	Data pointer for INX low nibble register
$11	-	DPM.2	DPM.1	DPM.0	R/W	Data pointer for INX middle nibble register
$12	-	DPH.2	DPH.1	DPH.0	R/W	Data pointer for INX high nibble register
$13	PBIEN.3	PBIEN.2	PBIEN.1	PBIEN.0	R/W	PORTB interrupt enable flags register
$14	PBIF.3	PBIF.2	PBIF.1	PBIF.0	R/W	PORTB request flags register
$15	PPBCR3	PPBCR.2	PPBCR.1	PPBCR.0	R/W	PORTB pull-high control register
$16	PACR.3	PACR.2	PACR.1	PACR.0	R/W	PORTA input/output control register
$17	PBCR.3	PBCR.2	PBCR.1	PBCR.0	R/W	PORTB input/output control register
$18	PCCR.3	PCCR.2	PCCR.1	PCCR.0	R/W	PORTC input/output control register
$19	PULLEN	-	-	-	R/W	Bit3: PORTA3-1 and PORTC sharing pull-high enable control register
$1A	-	-	DIV.1	DIV.0	R/W	Build-in RC Oscillator division register
$1B	-	-	-	-	-	Reserved
$1C	-	T0GO	T0S	T0E	R/W	Bit0: T0 signal edge, Bit1: T0 signal source Bit2: Timer0 start control/flag register
$1D	PPA0.1	PPA0.0	PA0E.1	PA0E.0	R/W	Bit1-0: PORTA.0 interrupt edge select register Bit3-2: PORTA.0 pull-high/pull low enable control register
$1E	WDT	WDT.2	WDT.1	WDT.0	R/W R	Bit2-0: Watchdog timer control register Bit3: Watchdog timer overflow flag register (Read only)
$1F	-	-	-	-	-	Reserved

3. ROM

The ROM can address 1024 X 16 bits of program area from $000 to $3FF.

3.1. Vector Address Area ($000 to $004)

The program is sequentially executed. There is an area address $000 through $004 that is reserved for a special interrupt service routine such as starting vector address.

Address	Instruction	Remarks
$000	JMP*	Jump to RESET service routine
$001	JMP*	Jump to External 0 interrupt service routine
$002	JMP*	Jump to TIMER0 interrupt service routine
$003	NOP	Reserved
$004	JMP*	Jump to Port interrupt service routine

*JMP instruction can be replaced by any instruction.

4. Initial State

4.1. System Register State

Address	Bit 3	Bit 2	Bit 1	Bit 0	Power On Reset /Pin Reset /Low Voltage Reset	WDT Reset
$00	IEX0	IET0	-	IEP	00-0	00-0
$01	IRQX0	IRQT0	-	IRQP	00-0	00-0
$02	-	T0M.2	T0M.1	T0M.0	-000	-000
$04	T0L.3	T0L.2	T0L.1	T0L.0	xxxx	xxxx
$05	T0H.3	T0H.2	T0H.1	T0H.0	xxxx	xxxx
$08	PA.3	PA.2	PA.1	PA.0	0000	0000
$09	PB.3	PB.2	PB.1	PB.0	0000	0000
$0A	PC.3	PC.2	PC.1	PC.0	0000	0000
$0E	TBR.3	TBR.2	TBR.1	TBR.0	xxxx	uuuu
$0F	INX.3	INX.2	INX.1	INX.0	xxxx	uuuu
$10	DPL.3	DPL.2	DPL.1	DPL.0	xxxx	uuuu
$11	-	DPM.2	DPM.1	DPM.0	-xxx	-uuu
$12	-	DPH.2	DPH.1	DPH.0	-xxx	-uuu
$13	PBIEN.3	PBIEN.2	PBIEN.1	PBIEN.0	0000	0000
$14	PBIF.3	PBIF.2	PBIF.1	PBIF.0	0000	0000
$15	PPBCR3	PPBCR.2	PPBCR.1	PPBCR.0	0000	0000
$16	PACR.3	PACR.2	PACR.1	PACR.0	0000	0000
$17	PBCR.3	PBCR.2	PBCR.1	PBCR.0	0000	0000
$18	PCCR.3	PCCR.2	PCCR.1	PCCR.0	0000	0000
$19	PULLEN	-	-	-	0---	0---
$1A	-	-	DIV.1	DIV.0	--00	--00
$1C	-	T0GO	T0S	T0E	-000	-000
$1D	PPA0.1	PPA0.0	PA0E.1	PA0E.0	0000	0000
$1E	WDT	WDT.2	WDT.1	WDT.0	0000	1000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'.

4.2. Other Initial States

Others	After any Reset
Program counter (PC)	$000
CY	Undefined
Accumulator (AC)	Undefined
Data memory	Undefined

5. System Clock and Oscillator

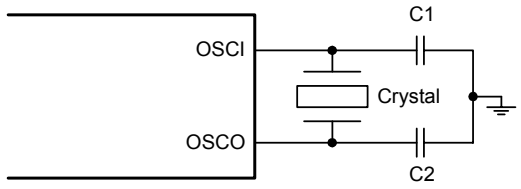
The oscillator generates the basic clock pulses that provide the system clock to supply CPU and on-chip peripherals. System clock fsys = fosc/4.

5.1. Instruction Cycle Time

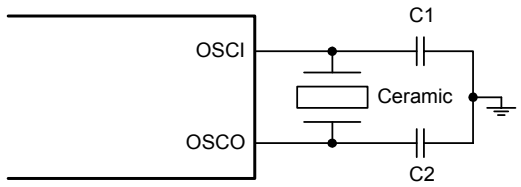
(1) 4/32768Hz (≈122μs) for 32768Hz oscillator.
(2) 4/500kHz (= 8μs) for 500kH oscillator.
(3) 4/8MHz (= 0.5μs) for 8MHz oscillator.

5.2. Oscillator Type

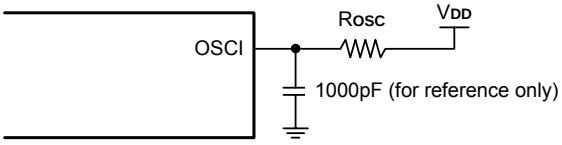
(1) Crystal Oscillator: 32.768kHz or 400k - 8MHz.



(2) Ceramic Resonator: 400kHz - 8MHz.

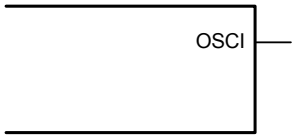


(3) RC Oscillator: 400kHz - 8MHz.



External RC

(4) Internal RC 8MHz:



Build in RC

5.3 Capacitor Selection for Oscillator

Ceramic Resonators			Recommend Type
Frequency	C1	C2	
455kHz	47 - 100pF	47 - 100pF	ZTB 455KHz
			ZT 455E
3.58MHz	-	-	ZTT 3.580M
			ZT 3.58M*
4MHz	-	-	ZTT 4.000M
			ZT 4M*

*- The specified ceramic resonator has internal built-in load capacity

Crystal Oscillator			Recommend Type
Frequency	C1	C2	
32.768kHz	5 - 12.5pF	5 - 12.5pF	DT 38 (φ3x8)
			φ3x8 - 32.768KHz
4MHz	8 - 15pF	8 - 15pF	HC-49U/S 4.000MHz
			49S-4.000M-F16E
8MHz	8 - 15pF	8 - 15pF	HC-49U/S 8.000MHz
			49S-8.000M-F16E

Notes:

1. **Capacitor values are used for design guidance only!**
2. These capacitors were tested with the crystals listed above for basic start-up and operation. **They are not optimized.**
3. Be careful for the stray capacitance on PCB board, the user should test the performance of the oscillator over the expected V_{DD} and the temperature range for the application.

Before selecting crystal/ceramic, the user should consult the crystal/ceramic manufacturer for appropriate value of external component to get best performance, visit http://www.sinowealth.com for more recommended manufactures

5.4. Build-in RC Oscillator Division Register

Build-in RC Oscillator division register (DIV)

Build-in RC Oscillator division register supply for system clock after division.DIV.1-0 used for the Internal RC frequency as shown in Table below.

Build-in RC Oscillator division register ($1A):

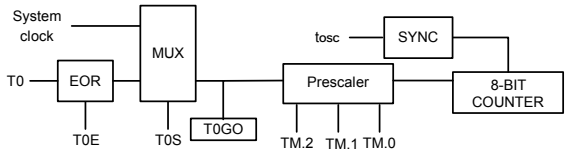
Address	Bit 3	Bit 2	Bit 1	Bit 0	R/W	Remarks
$1A	-	-	DIV.1	DIV.0	R/W	
	-	-	0	0	R/W	Divide Ratio = $/2^0$, Internal RC frequency = 8MHz
	-	-	0	1	R/W	Divide Ratio = $/2^1$, Internal RC frequency = 4MHz
	-	-	1	0	R/W	Divide Ratio = $/2^2$, Internal RC frequency = 2MHz
	-	-	1	1	R/W	Divide Ratio = $/2^4$, Internal RC frequency = 0.5MHz

Note: Build-in RC Oscillator division register is available only for Internal RC Oscillator.

6. Timer0

The timer/counter has the following features:
- 8-bit up-counting timer/counter
- Automatic re-loads counter
- 8-level prescaler
- Internal and external clock select
- Interrupt on overflow from $FF to $00
- Edge select for external event

The following is a simplified timer block diagram:



6.1. Timer0 Configuration and Operation

The Timer0 consists of an 8-bit write-only timer load register (TL0L, TL0H), and an 8-bit read-only timer counter (TC0L, TC0H). Each of them has low order digits and high order digits. Writing data into the timer load register (TL0L, TL0H) can initialize the timer counter.

The low-order digit should be written first, and then the high-order digit. The timer counter is automatically loaded with the contents of the load register when the high order digit is written or counter counts overflow from $FF to $00.

Timer Load Register: Since register H would control the physical READ and WRITE operations.

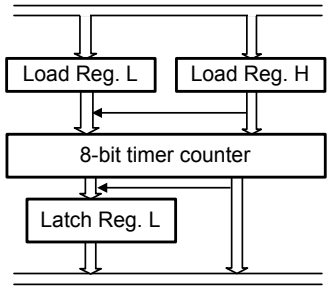
Please follow these steps:

Write Operation:

Low nibble first;
High nibble to update the counter.

Read Operation:

High nibble first;
Low nibble followed.



6.2. Timer0 Mode Register

The timer can be programmed in several different prescalers by setting Timer Mode register (T0M).

The clock source pre-scale by the 8-level counter first, then generate the output plus to timer counter. The Timer Mode registers (T0M) are 3-bit registers used for the timer control as shown in Table 1.

Table 1. Timer0 Mode Register ($02)

T0M.2	T0M.1	T0M.0	Prescaler Divide Ratio	Clock Source
0	0	0	$/2^{11}$	System clock/T0
0	0	1	$/2^{9}$	System clock/T0
0	1	0	$/2^{7}$	System clock/T0
0	1	1	$/2^{5}$	System clock/T0
1	0	0	$/2^{3}$	System clock/T0
1	0	1	$/2^{2}$	System clock/T0
1	1	0	$/2^{1}$	System clock/T0
1	1	1	$/2^{0}$	System clock/T0

Systems Register $1C: (T0)

Address	Bit 3	Bit 2	Bit 1	Bit 0	R/W	Remarks
$1C	-	T0GO	T0S	T0E	R/W	Bit0: T0 signal edge Bit1: T0 signal source
	-	X	X	0	R/W	Increment on low-to-high transition T0 pin
	-	X	X	1	R/W	Increment on high-to-low transition T0 pin
	-	X	0	X	R/W	Used as PORTA.2, Timer0 source is system clock
	-	X	1	X	R/W	Used as T0 input, Timer0 source is T0 input clock
	-	0	X	X	R/W	Timer0 counter disable
	-	1	X	X	R/W	Set “1” start Timer0, T0GO = 1 when Timer0 is running

6.3. External Clock/Event T0 as Timer0 Source

When external clock/event T0 input as Timer0 source, it is synchronized with the CPU system clock. The external source must follow certain constraints. The system clock samples it in instruction frame cycle. Therefore it is necessary to be high (at least 2 tosc) and low (at least 2 tosc). When the prescaler ratio selects /20, it is the same as the system clock input.
The requirement is as follows:

T0H (T0 high time) $\geq$ 2 * tosc + ΔT

T0L (T0 low time) $\geq$ 2 * tosc + ΔT ; ΔT = 20ns

When another prescaler ratio is selected, the TM0 is scaled by the asynchronous ripple counter and so the prescaler output is symmetrical.
Then:

$$\text{T0 high time} = \text{T0 low time} = \frac{N * T0}{2}$$

Where: T0 = Timer0 input period

N = prescaler value

The requirement is:

$$\frac{N * T0}{2} \geq 2 * t_{OSC} + \Delta T \quad \text{or} \quad T0 \geq \frac{4 * t_{OSC} + 2 * \Delta T}{N}$$

So, the limitation is applied for the T0 period time only. The pulse width is not limited by this equation. It is summarized as follows:

$$\text{T0} = \text{Timer0 period} \geq \frac{4 * t_{OSC} + 2 * \Delta T}{N}$$

7. I/O PORT

SH69P21 provides 11 bi-directional I/O ports and 1 input port (PORTA.3 used as open drain port). The PORT data put in register $08 - $0A. The PORT control register ($16 - $18) controls the PORT as input or output. Each I/O port has an internal pull-high resistor.

The internal pull-high resistor of PORTB is controlled by register $15 individually.

The internal pull-high resistor of PORTA3-1 and PORTC is controlled by register $19 shared together.

The internal pull-high resistor of PORTA0 is controlled by register $1D.

Port I/O mapping address is shown as follows:

System Register $08-$0A: PORT data register. System Register $16-$18: PORT control register

Address	Bit3	Bit2	Bit1	Bit0	R/W	Remarks
$08	PA.3	PA.2	PA.1	PA.0	R/W	PORTA data register
$09	PB.3	PB.2	PB.1	PB.0	R/W	PORTB data register
$0A	PC.3	PC.2	PC.1	PC.0	R/W	PORTC data register
$16	PACR.3	PACR.2	PACR.1	PACR.0	R/W	PORTA input/output control register
$17	PBCR.3	PBCR.2	PBCR.1	PBCR.0	R/W	PORTB input/output control register
$18	PCCR.3	PCCR.2	PCCR.1	PCCR.0	R/W	PORTC input/output control register

PA (/B/C) CR.n, (n = 0, 1, 2, 3)
0: Set I/O as an input direction. (Power on initial)
1: Set I/O as an output direction.

Note:

In 10 pin mode, bit 3-0 of the $0AH are reserved; always keep it"0" in the user's program, bit 3-0 of the $18H are reserved; always keep it"1" in the user's program.

In 8 pin mode, bit 3-0 of the $0AH and bit 3-2 of the $09H are reserved; always keep it"0" in the user's program, bit 3-0 of the $18H and bit 3-2 of the $17H are reserved; always keep it"1" in the user's program.

System Register $15, $19: PORT pull-high control register

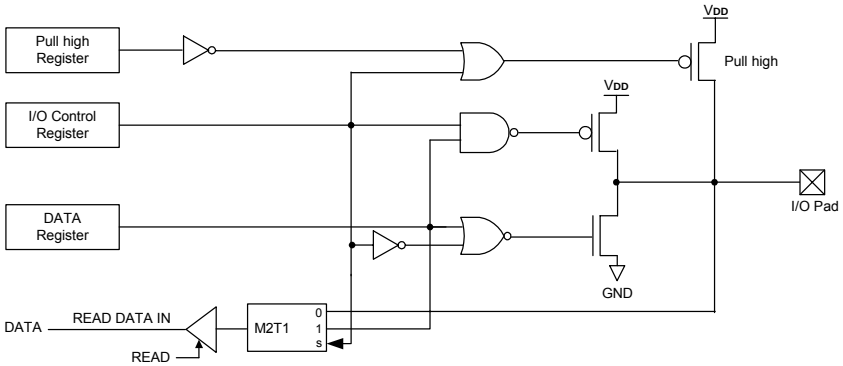
Address	Bit3	Bit2	Bit1	Bit0	R/W	Remarks
$15	PPBCR3	PPBCR.2	PPBCR.1	PPBCR.0	R/W	PORTB pull-high control register
$19	PULLEN	-	-	-	R/W	Bit3: PORTA3-1 and PORTC sharing pull-high enable control register

0: Disable internal pull-high resistor. (Power on initial)
1: Enable internal pull-high resistor.

Notice:

PORTA.3 is shared with $\overline{\text{RESET}}$ pin. It can only be shared as input pin when select internal reset circuit by code option.

Equivalent Circuit for a Single I/O Pin (Excluding PORTA.0):

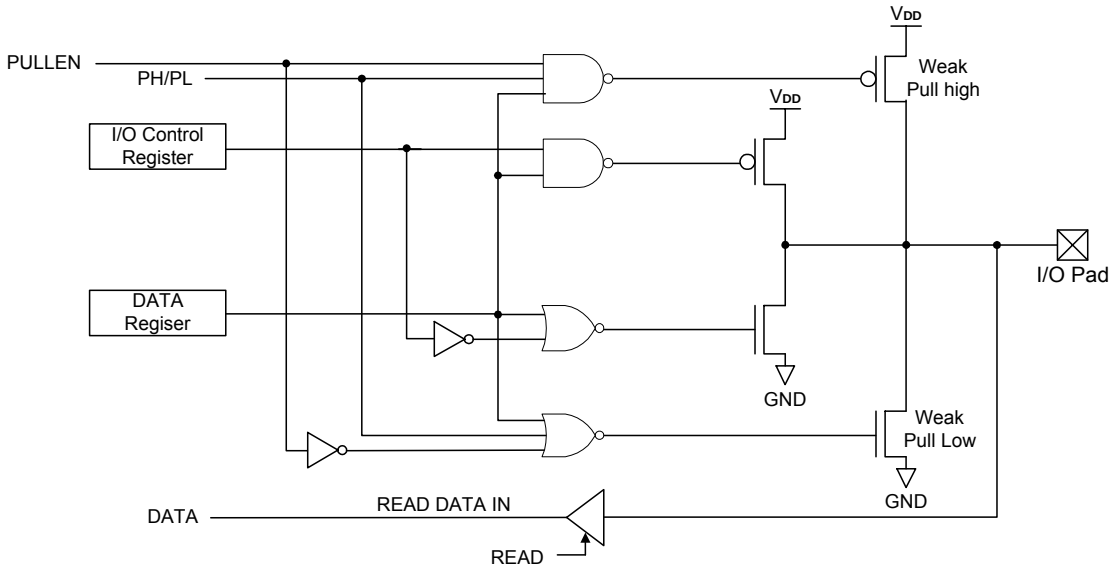


In SH69P21, each output port contains a latch, which can hold the output data. Writing the data register under the output mode can directly transfers data to the corresponding pin. All input ports do not have latches, so the external input data should be held externally until the input data is get from the outside. When a digital I/O port is selected to be an output port, the value of the associated port bit actually represent the value of the output data latch, not the voltage on the pin.

System Register $1D: PORTA.0 pull-high/pull low and interrupt edge select register

Address	Bit3	Bit2	Bit1	Bit0	R/W	Remarks
$1D	PPA0.1	PPA0.0	PA0E.1	PA0E.0	R/W	Bit1-0: External interrupt (PORTA.0) rising/falling /double edge select register Bit3-2: PORTA.0 pull-high/pull low enable control register
	X	X	0	0	R/W	Disable any edge as External interrupt (PORTA.0)
	X	X	0	1	R/W	Select falling edge as External interrupt (PORTA.0)
	X	X	1	0	R/W	Select rising edge as External interrupt (PORTA.0)
	X	X	1	1	R/W	Select double edge as External interrupt (PORTA.0)
	0	X	X	X	R/W	Disable PORTA.0 pull-high/pull low resistor
	1	0	X	X	R/W	Enable PORTA.0 pull-high resistor
	1	1	X	X	R/W	Enable PORTA.0 pull-low resistor

Equivalent Circuit for a Single I/O PORTA.0:



PORT Interrupt

The PORTB is used as port interrupt sources. Since PORTB I/O is bit programmable I/O, only when the PORTB is selected as normal I/O input, the voltage transition from V_{DD} to GND applying to the digital input port can generate a port interrupt. When it's select as analog input, Port interrupt request can not be generated.

The PORTB interrupt control flags are mapped on $13, $14 of the system register. They can be accessed or tested by the read/write operation. Those flags are clear to "0" at initialization by the chip reset. Port Interrupts can be used to wake up the CPU from the HALT or the STOP mode.

System Register $13:

Address	Bit3	Bit2	Bit1	Bit0	R/W	Remarks
$13	PBIEN.3	PBIEN.2	PBIEN.1	PBIEN.0	R/W	PORTB interrupt enable flags register

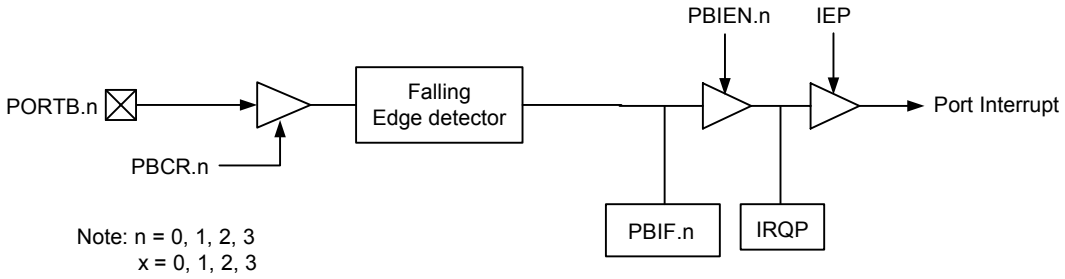
PBIEN.n (n = 0, 1, 2, 3)
0: Disable PORT interrupt. (Power on initial)
1: Enable PORT interrupt.

System Register $14:

Address	Bit3	Bit2	Bit1	Bit0	R/W	Remarks
$14	PBIF.3	PBIF.2	PBIF.1	PBIF.0	R/W	PORTB interrupt request flags register

PBIF.n (n = 0, 1, 2, 3)
0: PORT interrupt is not presented. (Power on initial)
1: PORT interrupt is presented.
Only writing these bits to "0" is available.

Following is the port interrupt function block-diagram for reference.



Port Interrupt Programming Notes:

- Any one of PORTB input pin transitions from V_{DD} to GND would set PBIF.x to "1", in spite of level of the other pin of PORTB.
- If PBIEN.x = 1and IEP = 1, the x of PORTB input pin transitions from V_{DD} to GND would generate an interrupt request (PBIF = 1) and interrupt the CPU, in spite of any level of the other pin of PORTB.

8. Interrupt

Three interrupt sources are available on SH69P21:
- External 0 interrupts (Rising/Falling/Double edge)
- Timer0 interrupt
- PORTB interrupts (Falling edge)

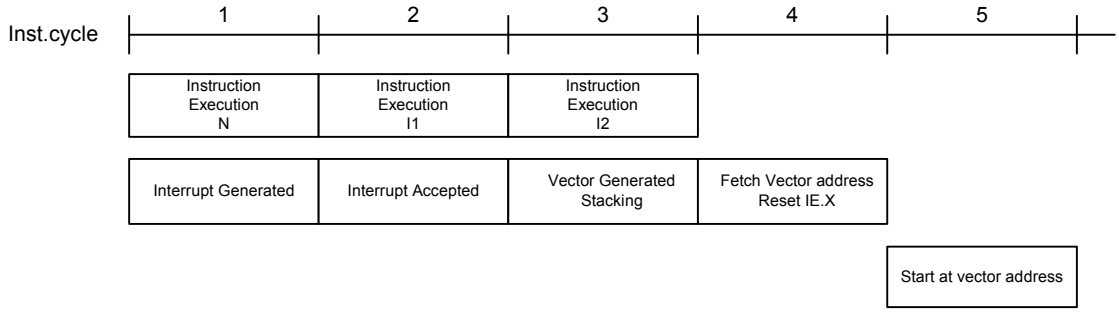
Interrupt Control Bits and Interrupt Service

The interrupt control flags are mapped on $00 and $01 of the system register. They can be accessed or tested by the program. Those flags are cleared to “0” at initialization by the chip reset.

System Register:

Address	Bit 3	Bit 2	Bit 1	Bit 0	R/W	Remarks
$00	IEX0	IET0	-	IEP	R/W	Interrupt enable flags register
$01	IRQX0	IRQT0	-	IRQP	R/W	Interrupt request flags register

When IEx is set to “1” and the interrupt request is generated (IRQx is 1), the interrupt will be activated and vector address will be generated from the priority PLA corresponding to the interrupt sources. When an interrupt occurs, the PC and CY flag will be saved into stack memory and jump to interrupt service vector address. After the interrupt occurs, all interrupt enable flags (IEx) are clear to “0” automatically, so when IRQx is 1 and IEx is set to “1” again, the interrupt will be activated and vector address will be generated from the priority PLA corresponding to the interrupt sources.



Interrupt Servicing Sequence Diagram

Interrupt Nesting

During the CPU interrupt service, the user can enable any interrupt enable flag before returning from the interrupt. The servicing sequence diagram shows the next interrupt and the next nesting interrupt occurrences. If the interrupt request is ready and the instruction of execution N is IE enabled, then the interrupt will start immediately after the next two instruction executions. However, if instruction I1 or instruction I2 disables the interrupt request or enable flag, then the interrupt service will be terminated.

External Interrupt0

If user wants to generate an External0 Interrupt (PORTA.0), the following must be executed:
1. Set the PORTA.0 as input port;
2. Select PORTA.0 pull-high/pull low resistor and the type of external interrupt edge (Rising/Falling/Double edge);
3. Enable Interrupt enable flags register External interrupt 0 (set $00 IEX0 to 1).

Timer0 Interrupt

The input clock of Timer0 is based on system clock or external clock/event T0 input as Timer0 source. The timer overflow from $FF to $00 will generate an internal interrupt request (IRQT0 = 1), If the interrupt enable flag is enabled (IET0 = 1), a timer interrupt service routine will start. Timer interrupt can also be used to wake the CPU from HALT mode.

Port Falling Edge Interrupt

Only the digital input port can generate an external interrupt.

If user wants to generate an External Interrupt (PORTB), the following must be executed:
1. Set the PORTB as input port;
2. Pull high PORTB.n (n = 0, 1, 2, 3);
3. Enable PORTB Interrupt enable flags register External interrupt 0 (set $13 PPBCR.n to 1);
4. Enable Interrupt enable flags register External interrupt (set $00 IEP to 1).

10. Low Voltage Reset (LVR)

The LVR function is to monitor the supply voltage and generate an internal reset in the device. It is typically used in AC line applications or large battery where large loads may be switched in and cause the device voltage to temporarily fall below the specified operating minimum.

The LVR function is selected by Code option.

The LVR circuit has the following functions when LVR function is enabled:

- Generates a system reset when $V_{DD} \leq V_{LVR}$
- Cancels the system reset when $V_{DD} > V_{LVR}$

Here, V_{DD}: Power supply voltage, V_{LVR}: is LVR detect voltage.

11. HALT and STOP Mode

After the execution of HALT instruction, SH69P21 will enter the HALT mode. In the HALT mode, CPU will STOP operating. But peripheral circuit (Timer, watchdog timer) will keep status.

After the execution of STOP instruction, SH69P21 will enter the STOP mode. The whole chip (including oscillator) will STOP operating.

In the HALT mode, SH69P21 can be waked up if any interrupt occurs.

In the STOP mode, SH69P21 can be waked up if port or external Interrupt occurs.

When CPU is awaked from the HALT/STOP by any interrupt source, it will execute the relevant interrupt serve subroutine at first. Then the instruction next to halt/stop is executed.

12. WDT

Watch dog timer is a down-count counter, and its clock source is an independent built-in RC oscillator, so that it will always run even in the STOP mode. The watchdog timer automatically generates a device reset when it overflows. It can be enabled or disabled permanently by using the code option.

The watchdog timer control bits ($1E Bit2 - Bit0) are used to select different overflow frequency. The watchdog timer overflow flag ($1E Bit3) will be automatically set to “1” by hardware when the watchdog timer overflows. By reading or writing the system register $1E, the watchdog timer should re-count before the overflow happens.

System Register $1E: Watchdog Timer (WDT)

Address	Bit 3	Bit 2	Bit 1	Bit 0	R/W	Remarks
$1E	WDT	WDT.2	WDT.1	WDT.0	R/W R	Bit0-2: Watchdog timer control register Bit3: Watchdog timer overflow flag register
	X	0	0	0	R/W	Watchdog timer overflow period is 4096ms
	X	0	0	1	R/W	Watchdog timer overflow period is 1024ms
	X	0	1	0	R/W	Watchdog timer overflow period is 256ms
	X	0	1	1	R/W	Watchdog timer overflow period is 128ms
	X	1	0	0	R/W	Watchdog timer overflow period is 64ms
	X	1	0	1	R/W	Watchdog timer overflow period is 16ms
	X	1	1	0	R/W	Watchdog timer overflow period is 4ms
	X	1	1	1	R/W	Watchdog timer overflow period is 1ms
	0	X	X	X	R	No watchdog timer overflow reset
	1	X	X	X	R	Watchdog timer overflow, WDT reset happens

Note: Watchdog timer overflow period is valid for V_{DD} = 5V.

13. Warm-up Timer

The device has a built-in warm-up timer to eliminate unstable state of initial oscillation when oscillator starts oscillating in the following conditions:

A. Power-on-reset

(1) In Internal RC oscillator mode, fosc = 8MHz, the warm-up counter prescaler divide ratio is $1/2^{13}$ (8192).

(2) In External RC oscillator mode, fosc = 400kHz - 8MHz, the warm-up counter prescaler divide ratio is $1/2^{13}$ (8192).

(3) In Crystal oscillator or Ceramic resonator mode, the warm-up counter prescaler divide ratio is $1/2^{13}$ (8192).

(4) In Crystal oscillator 32.768kHz mode, the warm-up counter prescaler divide ratio is $1/2^{15}$ (32768).

B. Pin reset, wake up from stop mode and low voltage reset

(1) In Internal RC oscillator mode, fosc = 8MHz, the warm-up counter prescaler divide ratio is $1/2^{7}$ (128).

(2) In External RC oscillator mode, fosc = 400kHz-8MHz, the warm-up counter prescaler divide ratio is $1/2^{7}$ (128).

(3) In Crystal oscillator or Ceramic resonator mode, the warm-up counter prescaler divide ratio is $1/2^{12}$ (4096).

(4) In Crystal oscillator 32.768kHz mode, the warm-up counter prescaler divide ratio is $1/2^{15}$ (32768).

14. Code Option

(a) Oscillator Type:

OP_OSC [2:0]:

000 = Reserve
001 = Reserve
010 = Reserve
011 = Internal RC 8MHz (OSCI and OSCO Pin as PORTB.1 and PORTB.0) (default)
100 = External RC 400kHz - 8MHz
101 = Crystal/Ceramic 400kHz - 8MHz (driving ability normal)
110 = Crystal/Ceramic 400kHz - 8MHz (driving ability strong)
111 = Crystal 32.768kHz

Note: When select OP_OSC[2:0] = 101, Oscillator driving ability is normal, suitable in common environment.
When select OP_OSC[2:0] = 101, Oscillator driving ability is strong, suitable in high humidity environment.

(b) Watchdog Timer:

OP_WDT:
0 = Disable (Default)
1 = Enable

(c) Low Voltage Reset:

OP_LVR:
0 = Disable (Default)
1 = Enable

(d) LVR Voltage Range:

OP_LVR0:
0 = High LVR Voltage (4.0V) (Default)
1 = Low LVR Voltage (2.5V)

(e) Reset:

OP_RST:
0 = PORTA.3 as Reset Pin (Default)
1 = PORTA.3 as I/O Pin

Instruction Set

All instructions are one cycle and one-word instructions. The characteristic is memory-oriented operation.

1. Arithmetic and Logical Instruction

1.1. Accumulator Type

Mnemonic	Instruction Code	Function	Flag Change
ADC X (, B)	00000 0bbb xxx xxxx	AC ← Mx + AC + CY	CY
ADCM X (, B)	00000 1bbb xxx xxxx	AC, Mx ← Mx + AC + CY	CY
ADD X (, B)	00001 0bbb xxx xxxx	AC ← Mx + AC	CY
ADDM X (, B)	00001 1bbb xxx xxxx	AC, Mx ← Mx + AC	CY
SBC X (, B)	00010 0bbb xxx xxxx	AC ← Mx + -AC + CY	CY
SBCM X (, B)	00010 1bbb xxx xxxx	AC, Mx ← Mx + -AC + CY	CY
SUB X (, B)	00011 0bbb xxx xxxx	AC ← Mx + -AC +1	CY
SUBM X (, B)	00011 1bbb xxx xxxx	AC, Mx ← Mx + -AC +1	CY
EOR X (, B)	00100 0bbb xxx xxxx	AC ← Mx ⊕ AC	
EORM X (, B)	00100 1bbb xxx xxxx	AC, Mx ← Mx ⊕ AC	
OR X (, B)	00101 0bbb xxx xxxx	AC ← Mx | AC	
ORM X (, B)	00101 1bbb xxx xxxx	AC, Mx ← Mx | AC	
AND X (, B)	00110 0bbb xxx xxxx	AC ← Mx & AC	
ANDM X (, B)	00110 1bbb xxx xxxx	AC, Mx ← Mx & AC	
SHR	11110 0000 000 0000	0 → AC[3], AC[0] → CY; AC shift right one bit	CY

1.2. Immediate Type

Mnemonic	Instruction Code	Function	Flag Change
ADI X, I	01000 iiii xxx xxxx	AC ← Mx + I	CY
ADIM X, I	01001 iiii xxx xxxx	AC, Mx ← Mx + I	CY
SBI X, I	01010 iiii xxx xxxx	AC ← Mx + -I +1	CY
SBIM X, I	01011 iiii xxx xxxx	AC, Mx ← Mx + -I +1	CY
EORIM X, I	01100 iiii xxx xxxx	AC, Mx ← Mx ⊕ I	
ORIM X, I	01101 iiii xxx xxxx	AC, Mx ← Mx | I	
ANDIM X, I	01110 iiii xxx xxxx	AC, Mx ← Mx & I	

1.3. Decimal Adjustment

Mnemonic	Instruction Code	Function	Flag Change
DAA X	11001 0110 xxx xxxx	AC, Mx ← Decimal adjust for add	CY
DAS X	11001 1010 xxx xxxx	AC, Mx ← Decimal adjust for sub	CY

2. Transfer Instruction

Mnemonic	Instruction Code	Function	Flag Change
LDA X (, B)	00111 0bbb xxx xxxx	AC ← Mx	
STA X (, B)	00111 1bbb xxx xxxx	Mx ← AC	
LDI X, I	01111 iiii xxx xxxx	AC, Mx ← I	

3. Control Instruction

Mnemonic	Instruction Code	Function	Flag Change
BAZ X	10010 xxxx xxx xxxx	PC ← X, if AC = 0	
BNZ X	10000 xxxx xxx xxxx	PC ← X, if AC ≠ 0	
BC X	10011 xxxx xxx xxxx	PC ← X, if CY = 1	
BNC X	10001 xxxx xxx xxxx	PC ← X, if CY ≠ 1	
BA0 X	10100 xxxx xxx xxxx	PC ← X, if AC (0) = 1	
BA1 X	10101 xxxx xxx xxxx	PC ← X, if AC (1) = 1	
BA2 X	10110 xxxx xxx xxxx	PC ← X, if AC (2) = 1	
BA3 X	10111 xxxx xxx xxxx	PC ← X, if AC (3) = 1	
CALL X	11000 xxxx xxx xxxx	ST ← CY, PC +1 PC ← X (Not include p)	
RTNW H; L	11010 000h hhh llll	PC ← ST; TBR ← hhhh, AC ← lll	
RTNI	11010 1000 000 0000	CY, PC ← ST	CY
HALT	11011 0000 000 0000		
STOP	11011 1000 000 0000		
JMP X	1110p xxxx xxx xxxx	PC ← X ((Include p)	
TJMP	11110 1111 111 1111	PC ← (PC11-PC8) (TBR) (AC)	
NOP	11111 1111 111 1111	No Operation	

Where,

PC	Program counter	I	Immediate data
AC	Accumulator	⊕	Logical exclusive OR
-AC	Complement of accumulator	|	Logical OR
CY	Carry flag	&	Logical AND
Mx	Data memory	bbb	RAM bank
p	ROM page	B	RAM bank
ST	Stack	TBR	Table Branch Register

Electrical Characteristics

Absolute Maximum Rating*

DC Supply Voltage . . . . . -0.3V to +7.0V

Input/Output Voltage . . . . . .-0.3V to V_{DD} + 0.3V

Operating Ambient Temperature . . . . . .-40°C to +85°C

Storage Temperature. . . . . .-55°C to +125°C

***Comments**

Stresses exceed those listed under **"Absolute Maximum Ratings"** may cause permanent damage to this device. These are stress ratings only. Functional operation of this device under these or any other conditions exceed those indicated in the operational sections of this specification is not implied or intended. Exposure to the absolute maximum rating conditions for extended periods may affect device reliability.

DC Electrical Characteristics (V_{DD} = 2.4 - 5.5V GND = 0V, T_A = 25℃, unless otherwise specified.)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Operating Voltage	V_{DD}	4.5	5.0	5.5	V	4MHz ≤ fosc ≤ 8MHz
		2.4	3.0	3.6	V	30kHz ≤ fosc ≤ 4MHz
Low Voltage Reset voltage	V_{LVR}	2.3	2.5	2.7	V	LVR (Low) enable
		3.8	4.0	4.2	V	LVR (High) enable
Operating Current	I_{OP}	-	1.3	1.5	mA	fosc = 8MHz, V_{DD} = 5.0V All output pins unloaded, execute NOP instruction, WDT off
		-	0.3	0.6	mA	fosc = 4MHz, V_{DD} = 3.0V All output pins unloaded, execute NOP instruction, WDT off
		-	-	300	μA	fosc = 500kHz, V_{DD} = 5.0V All output pins unloaded, execute NOP instruction, WDT off
Stand by Current	I_{SB}	-	-	1	mA	fosc = 8MHz, V_{DD} = 5.0V All output pins unloaded (HALT mode), WDT off, LVR off
		-	-	300	μA	fosc = 4MHz, V_{DD} = 3.0V All output pins unloaded (HALT mode), WDT off, LVR off
		-	10	15	μA	fosc = 32.768kHz, V_{DD} = 5.0V All output pins unloaded (HALT mode), WDT off, LVR off
		-	-	1	μA	V_{DD} = 5.0V All output pins unloaded (STOP mode), WDT off, LVR off
WDT Current	I_{WDT}	-	-	5	μA	V_{DD} = 5.0V
Input Low Voltage	V_{IL}	GND	-	0.3 X V_{DD}	V	I/O Ports, pins tri-state
		GND	-	0.2 X V_{DD}	V	$\overline{\text{RESET}}$, T0, OSCI (Schmitt Trigger input) PORTA0, PORTB (Schmitt Trigger input when used as external interrupt)
Input High Voltage	V_{IH}	0.7 X V_{DD}	-	V_{DD}	V	I/O Ports, pins tri-state
		0.8 X V_{DD}	-	V_{DD}	V	$\overline{\text{RESET}}$, T0, OSCI (Schmitt Trigger input) PORTA0, PORTB (Schmitt Trigger input when used as external interrupt)

(Continued)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Input Leakage Current	I_{IL}	-1	-	1	μA	I/O Ports, GND < $V_{I/O}$ < V_{DD}
		-5	-	-	μA	$V_{\overline{RESET}}$ = GND + 0.25V
		-	1	5	μA	$V_{\overline{RESET}}$ = V_{DD}
		-3	1	3	μA	T0, GND < $V_{I/O}$ < V_{DD}
Pull-up/Pull-low Resistor	R_P	-	150	-	kΩ	Pull-high/Pull-low resistor (V_{DD} = 5.0V)
Output High Voltage	V_{OH}	V_{DD} - 0.7	-	-	V	I/O Ports, I_{OH} = -10mA (V_{DD} = 5.0V)
Output Low Voltage	V_{OL}	-	-	GND + 0.6	V	I/O Ports, I_{OL} = 20mA (V_{DD} = 5.0V)

Limit of special define current

Note:

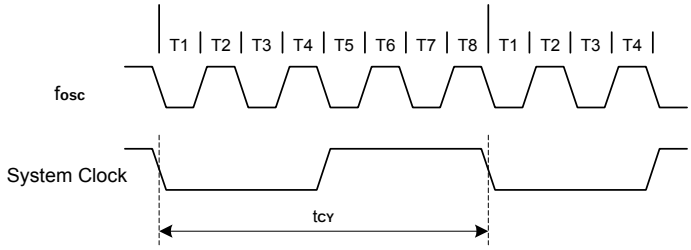
The maximum current of V_{DD} must be less than 100mA.
The maximum current of GND must be less than150mA.
The maximum output current of any low Voltage I/O port must be less than 50mA.
The maximum output current of any high Voltage I/O port must be less than 40mA.

AC Electrical Characteristics (V_{DD} = 2.4V - 5.5V, GND = 0V, T_A = 25℃, unless otherwise specified.)

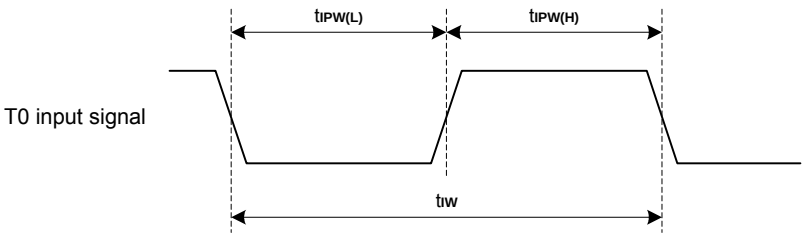
Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Instruction cycle time	T_{CY}	0.5	-	133	μs	f_{OSC} = 30kHz - 8MHz
T0 input width	t_{IW}	(T_{CY} +40)/N	-	-	ns	N = Prescaler divide ratio
Input pulse width	t_{IPW}	$t_{IW/2}$	-	-	ns	
RESET pulse width (low)	T_{RESET}	10	-	-	μs	Low active
Oscillator Start Time	T_{OSC1}	-	-	2	s	f_{OSC} = 32.768kHz
WDT Period	t_{WDT}	1	-	-	ms	
Frequency Stability (RC)	|Δ F|/F	-	-	20	%	External RC Oscillator include temperature and chip-to-chip variation
Internal RC Frequency Variation	|Δ F|/F	-	2	3	%	Internal RC Oscillator, T_A = 25℃, |F(5.0V±0.5V)-F|/F
		-	3	5	%	Internal RC Oscillator, T_A = 5℃-55℃ include temperature and chip-to-chip variation

Timing Waveform

(a) System Clock Timing Waveform:



(b) T0 Input Waveform:



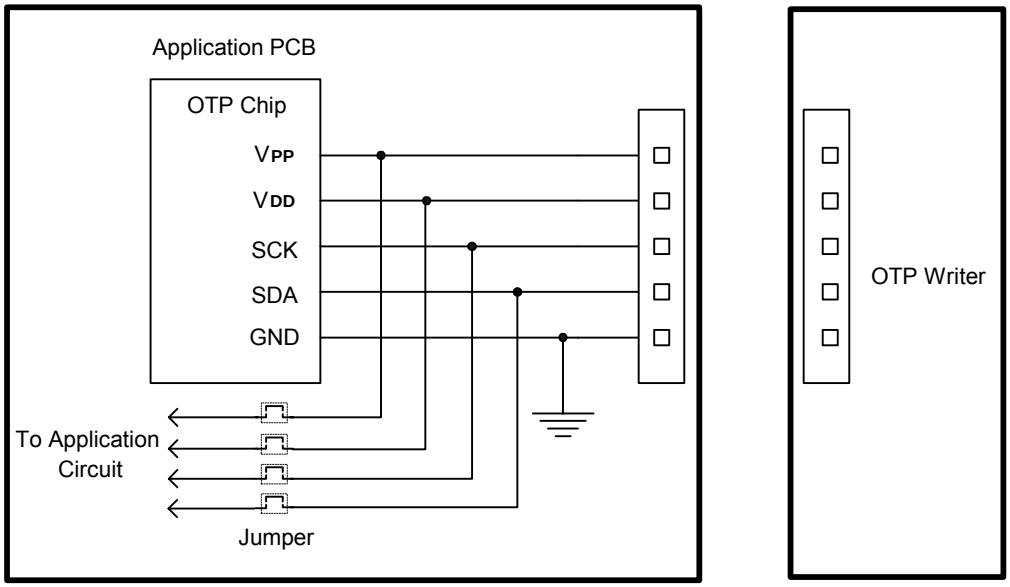
In System Programming notice for OTP

The In System Programming technology is valid for OTP chip.

The Programming Interface of the OTP chip must be set on user's application PCB, and users can assemble all components including the OTP chip in the application PCB before programming the OTP chip. Of course, it's accessible bonding OTP chip only first, and then programming code and finally assembling other components.

Since the programming timing of Programming Interface is very sensitive, therefore four jumpers are needed (V_{DD}, V_{PP}, SDA, SCK) to separate the programming pins from the application circuit as shown in the following diagram.

For a few of OTP chips with many V_{DD} pads, the V_{DD} pads should be connected together.



The recommended steps are the followings:

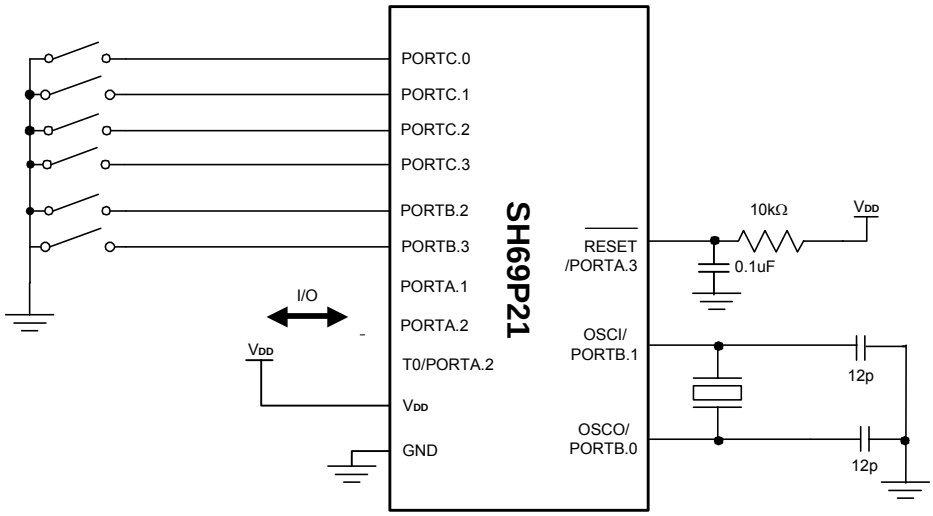
(1) The jumpers are open to separate the programming pins from the application circuit before programming the chip.
(2) Connect the programming interface with OTP writer and begin programming.
(3) Disconnect OTP writer and shorten these jumpers when programming is completed.

For more detail information, please refer to the OTP writer user manual.

Application Circuits (for reference only)

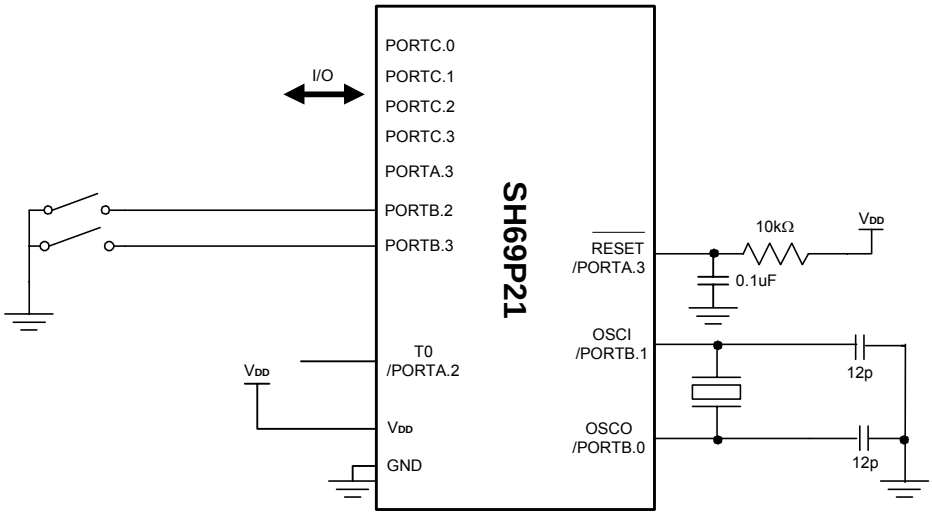
AP1:

(1) Operating Voltage: 3.0V
(2) Oscillator: Crystal 32.768kHz
(3). PORTC & PORTB: Input
(4). PORTA: Input/output



AP2:

(1) Operating Voltage: 5.0V
(2) Oscillator: Crystal 4MHz
(3) PORTA & PORTC: Input/output
(4) PORTB: Input

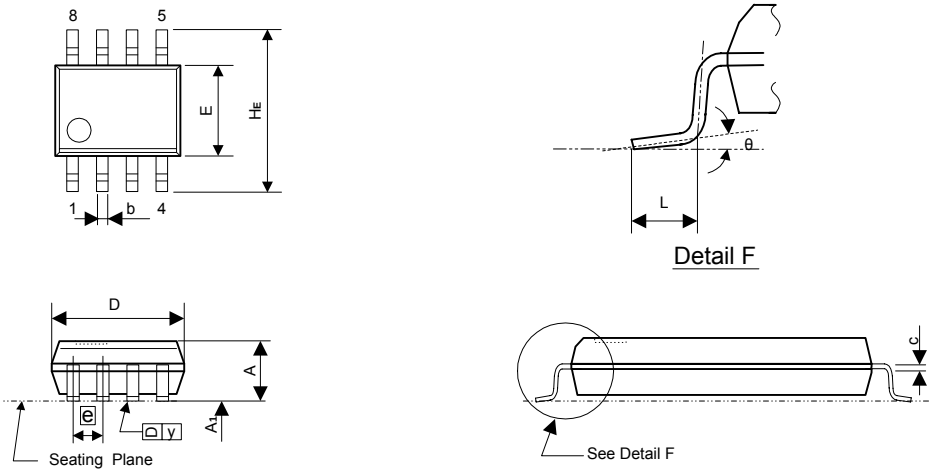


Ordering Information

Part No.	Packages
SH69P21M/014MU	14 SOP
SH69P21M/008MU	8 SOP

Package Information

SOP 8L Outline Dimensions **unit: inches/mm**

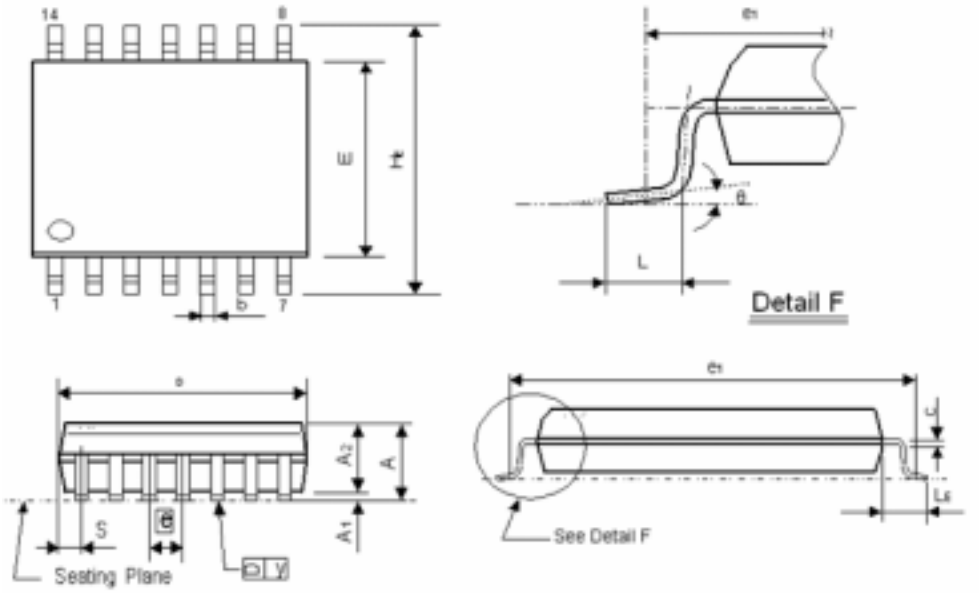


Symbol	Dimensions in inches	Dimensions in mm
A	0.069 Max.	1.75 Max.
	0.053 Min.	1.35 Min.
A1	0.010 Max.	0.25 Max.
	0.004 Min.	0.10 Min.
b	0.016 Typ.	0.41 Typ.
c	0.008 Typ.	0.20 Typ.
D	0.196 Max.	4.98 Max.
	0.189 Min.	4.80 Min.
E	0.157 Max.	3.99 Max.
	0.150 Min.	3.81 Min.
e	0.050 Typ.	1.27 Typ.
HE	0.244 Max.	6.20 Max.
	0.228 Min.	5.79 Min.
L	0.050 Max.	1.27 Max.
	0.016 Min.	0.41 Min.
y	0.004 Max.	0.10 Max.
θ	0° ~ 8°	0° ~ 8°

Notes:

1. The maximum value of dimension D includes end flash.
2. Dimension E does not include resin fins.

SOP 14L Outline Dimensions **unit: inches/mm**



Symbol	Dimensions in inches	Dimensions in mm
A	0.600~0.680	1.524~1.727
A_1	0.0040~0.0098	0.102~0.249
A_2	0,055~0.061	1.3971~1.549
b	0.0150 +0.0050 -0.0012	0.400 +0.108 -0.049
C	0.0080 +0.0018 -0.0005	0.200 +0.050 -0.010
D	0.3380~0.3440	8.585~8.738
E	0.1520~0.1574	3.861~3.998
e	0.050BSC	1.270BSC
H_E	0.2300~0.2440	5.842~6.198
L	0.0160~0.0350	0.406~0.889
y	0.004	0.100
θ	0~8°	0~8°

Notes:

1. The maximum value of dimension D includes end flash.
2. Dimension E does not include resin fins.
3. Dimension S includes end flash.

Data Sheet Revision History

Version	Content	Date
2.0	Add package information of the SOP8, SOP14	May. 2011
1.0	Original	Aug. 2010