NT93480

4-bit Microcontroller with LCD Driver

Features

■Memory
- -RAM: 2048× 4-bit
  - Includes: RAM mapped registers
  - Memory registers
  - Stackers
  - LCD memory
- -ROM: 8192× 10-bit
  - Includes: Program code
  - Character Graphics table

■Subroutine nesting
- -16-level subroutine nesting

■LCD Controller/Driver
- -40 segments × 8 commons
- -16-level contrast control
- -Programmable duty: 1/4, 1/8
- -Programmable bias: 1/3, 1/4

■Keyboard
- -8× 4 keys maximum

■Battery-low detection
- -4-level detection for battery-low

■I/O Pins
- -20 general I/O pins
- -16 I/O pins shared with LCD segment pins

■Timers/Counters
- -One 8-bit Watch Timer driven by a standard 32.768KHz oscillator or external clock
- -One 8-bit programmable Timer/Counter with external clock
- -One 8-bit programmable Timer/Counter
- -One 12-bit programmable Timer/Counter

■Communication port
- -1 Serial I/O port
- -1 Infra-red output pin

■Interrupts
- -9 interrupts are available

■Sound generator
- -1 programmable sound generator

■Power-down mode
- -HALT mode
- -STOP mode

■Dual Oscillators System
- -Main oscillator frequency (f_{MAIN}): 2MHz/2 built-in RC or 1MHz ceramic oscillator determined by code option
- -Watch oscillator frequency (f_{WATCH}): 32.768KHz crystal oscillator

■Instruction Execution Time
- -4μs or 8μs

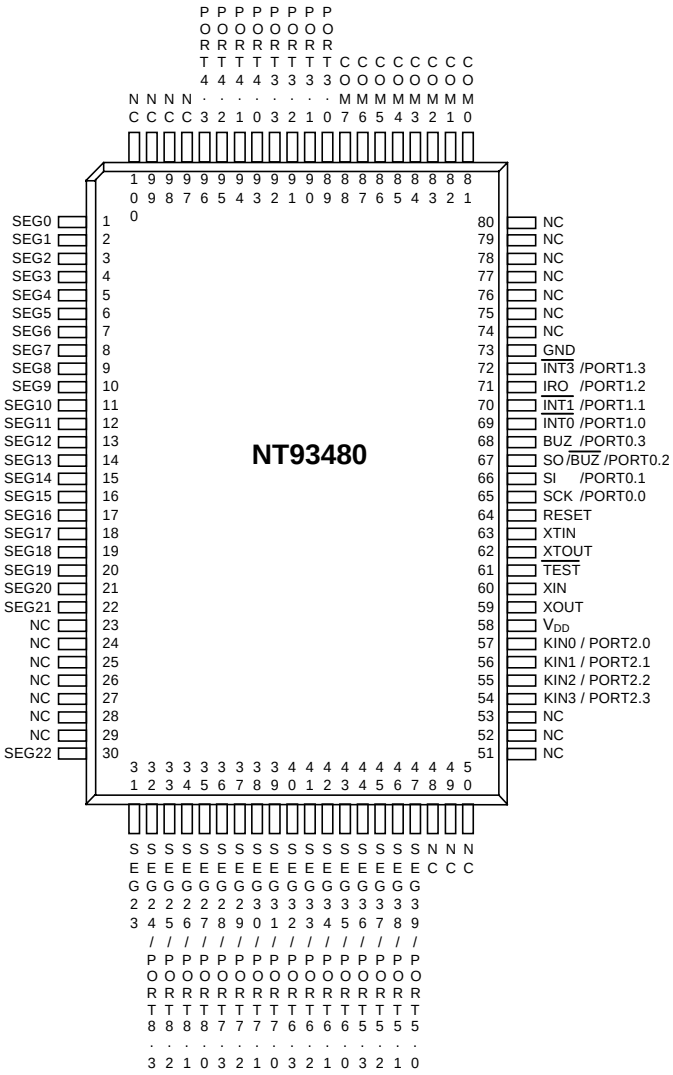
■Power System
- -Operating voltage range: 2.5~6.0V
- -Standby current: 3μA for 3V
- -Operating current: 300μA for 3V

■Package Type
- -100-pin QFP package

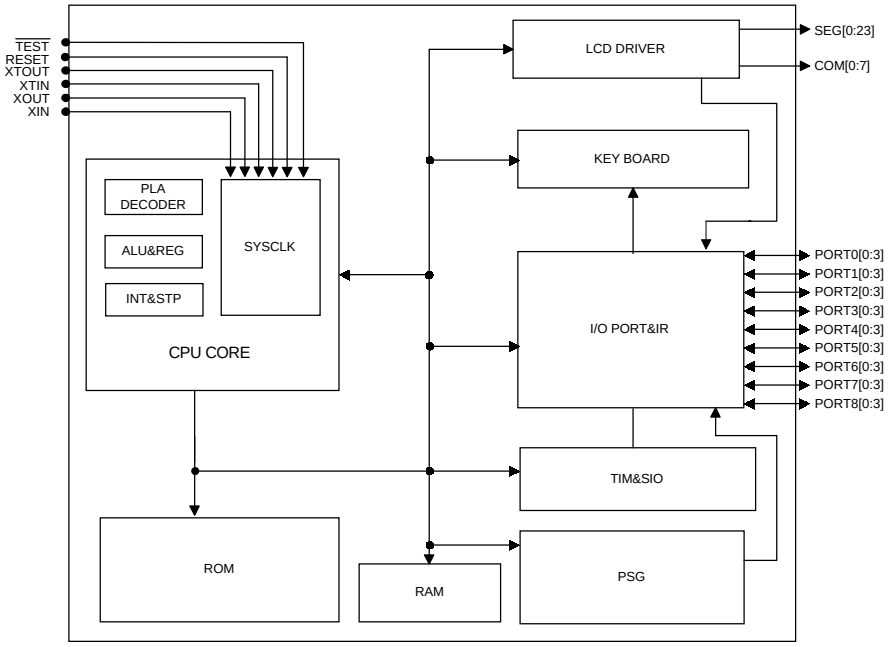
General Description

NT93480 is a single-chip CMOS 4-bit microcontroller for general purpose. To expand its capability, it integrates 4 timers/counters, 1 Infra-red communication port, 1 LCD driver driving up to 320 dots, 1 dual tone sound generator and 20 I/O pins, 9 interrupt vectors providing rapid response to different internal or external events. Furthermore, it offers a really low power consumption mode for saving energy in idle condition.

Pin Configuration



Block Diagram



Pin Description

Pin No.	Designation	I/O	Voltage Range	Description
1~22	SEG0~21	O	GND-V_{DD}	LCD segment signal output
30~31	SEG22~23	O	GND-V_{DD}	LCD segment signal output
32~35	SEG24~27 / PORT8.3~0	O I/O	GND-V_{DD}	LCD segment signal output / Programmable I/O port8, bit3~0
36~39	SEG28~31 / PORT7.3~0	O I/O	GND-V_{DD}	LCD segment signal output / Programmable I/O port7, bit3~0
40~43	SEG32~35 / PORT6.3~0	O I/O	GND-V_{DD}	LCD segment signal output / Programmable I/O port6, bit3~0
44~47	SEG36~39 / PORT5.3~0	O I/O	GND-V_{DD}	LCD segment signal output / Programmable I/O port5, bit3~0
54~57	KIN3~0 / PORT2.3~0	I I/O	GND-V_{DD}	Keyboard input port ($\overline{INT2}$) (See note 1) / Programmable I/O port2, bit3~0
58	V_{DD}	P	-	Power supply input (See note 2)
59	XOUT	O	GND-V_{DD}	1MHz ceramic oscillator output
60	XIN	I	GND-V_{DD}	1MHz ceramic oscillator input or 2MHz RC oscillator input
61	$\overline{TEST}$	I	GND-V_{DD}	Enable TEST MODE when low (for factory used only, user must connect it to V_{DD})
62	XTOUT	O	GND-V_{DD}	32KHz crystal oscillator output
63	XTIN	I	GND-V_{DD}	32KHz crystal oscillator input

Pin Description (continued)

Pin No.	Designation	I/O	Voltage Range	Description
64	RESET	I	GND-V_{DD}	System reset input (High active)
65	SCK / PORT0.0	I/O I/O	GND-V_{DD}	Serial clock pin / Programmable I/O port0, bit0
66	SI / PORT0.1	I I/O	GND-V_{DD}	Serial data input / Programmable I/O port0, bit1
67	SO / $\overline{BUZ}$ / PORT0.2	O O I/O	GND-V_{DD}	Serial data output / Buzzer output (normal high) / Programmable I/O port0, bit2
68	BUZ / PORT0.3	O I/O	GND-V_{DD}	Buzzer output (normal low) / Programmable I/O port0, bit3
69	$\overline{INT0}$ / PORT1.0	I I/O	GND-V_{DD}	External interrupt input ($\overline{INT0}$) (See note 1) / Programmable I/O port1, bit0
70	$\overline{INT1}$ / PORT1.1	I I/O	GND-V_{DD}	External interrupt input ($\overline{INT1}$) (See note 1) / Programmable I/O port1, bit1
71	IRO / PORT1.2	O I/O	GND-V_{DD}	IR communication output Programmable I/O port1, bit2
72	$\overline{INT3}$ / PORT1.3	I I/O	GND-V_{DD}	External interrupt input ($\overline{INT3}$) (See note 1) / Programmable I/O port1, bit3
73	GND	P	-	Ground input
81~88	COM0~7	O	GND-V_{DD}	LCD common signal output (Keyboard scanning outputs)
89~92	PORT3.0~3	I/O	GND-V_{DD}	Programmable I/O port3, bit0~3
93~96	PORT4.0~3	I/O	GND-V_{DD}	Programmable I/O port4, bit0~3
23~29, 48~53, 74~80, 97~100	NC	-	-	-

Notes: 1. All interrupt inputs would be triggered by any negative-edge signal.

2. V_{DD} = 2.5 ~ 6.0 V.

Functional Description

Program memory

The 4-bit CPU can directly address up to 16K words of program memory. The arithmetic logic unit (ALU) performs 4-bit addition, subtraction, logical and other operations in one cycle and branch operations in two or three cycles.

Vector address area ($0000 to $0019)
NT93480 provides a vector address area for program initialization and interrupt service.
They are:

$00-01	Jump to RESET routine	
$02-03	Jump to $\overline{\text{INT0}}$	service routine
$04-05	Jump to $\overline{\text{INT1}}$	service routine
$06-07	Jump to $\overline{\text{INT2}}$	service routine
$08-09	Jump to $\overline{\text{INT3}}$	service routine
$0A-0B	Jump to $\overline{\text{TIMER0}}$	service routine
$0C-0D	Jump to $\overline{\text{TIMER1}}$	service routine
$0E-0F	Jump to $\overline{\text{TIMER2}}$	service routine
$10-11	Jump to $\overline{\text{TIMER3}}$	service routine
$12-13	Jump to Serial I/O	service routine
$14-19	Reserved	

Random Access Data Memory

Resident data memory is organized as 2,048× 4 bits (2048 nibbles). RAM is used for data storage, register storage, stack, and storage of segment data for LCD display RAM. All the interrupt control registers and other special function registers are implemented by means of memory mapping to the internal RAM space. Note that the upper 1K RAM can **only** be accessed by indirect addressing.

RAM Addressing

RAM data may be accessed by either direct or indirect addressing. Direct addressing is addressed by operand itself while indirect addressing is addressed via page register V. Addressing of RAM for each page is performed by register X and Y indirect addressing. There are 16 special digits in RAM that can be addressed directly without the use of X and Y. These digits make up memory register (R0~15).

RAM Mapping

NT93480 employs memory-mapped I/O in which peripheral hardware (such as I/O port mode control and timers) is mapped onto address $000 through $03F in the data memory space.

ROM MAP

$0000 $0019	Vector Address Area
$001A $003F	Zero-page Subroutines
$0040 $1FFF	System Program

RAM MAP

$000	RAM Mapped Register
$040	Memory Register (R0~R15)
$050	Data
$100	LCD Display Area **(Note)**
$200	Stack
$240	Data
$400 $7FF	Data (Indirect addressing only)

Note: $1n0~$1n3, $1nE~1nF, $1m0~1mF are empty space (no memory cell).
(n=0.1.2~7, m=8.9.A~F)

RAM Mapped Register

Register		Contents				Configuration	
Address	**Name**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**	**R/W**	**s/r/t/N**
$000	SENSE0	IE0 (R/W)	IRQ0 (R)	RSP (R/W)	IE (R/W)	-	s/r/t r/t
$001	SENSE1	IE2 (R/W)	IRQ2 (R)	IE1 (R/W)	IRQ1 (R)	-	s/r/t r/t
$002	SENSE2	IET1 (R/W)	IRQT1 (R)	IET0 (R/W)	IRQT0 (R)	-	s/r/t r/t
$003	SENSE3	IET2 (R/W)	IRQT2 (R)	IESI (R/W)	IRQSI (R)	-	s/r/t r/t
$004	TMOD0	Timer 0 mode register				W	N
$005	TMOD1	Timer 1 mode register				W	N
$006	TC00 TL00	Timer 0 counter register 0 Timer 0 load register 0				R W	N
$007	TC01 TL01	Timer 0 counter register 1 Timer 0 load register 1				R W	N
$008	TC10 TL10	Timer 1 counter register 0 Timer 1 load register 0				R W	N
$009	TC11 TL11	Timer 1 counter register 1 Timer 1 load register 1				R W	N
$00A	SIOL	Serial data register , lower nibble				R/W	t/N
$00B	SIOH	Serial data register , upper nibble				R/W	t/N
$00C	SIOM	Serial mode register				W	N
$00D	CKS	Reserved	Reserved	CKS.1	CKS.0	W	N
$00E	-	Reserved				-	-
$00F	TESTM	Test mode register (Confidential)				W	N
$010	PMODA	Port mode register A				W	N
$011	PMODB	Port mode register B				W	N
$012	PMODC	Port mode register C				W	N
$013	TMOD3	Timer 3 mode register				W	N
$014	PMODE	Port mode register E				W	N
$015	PMODF	Port mode register F				W	N
$016	IRM	W1	W0	IRM.1	IRM.0	R/W	s/r/N
$017	LCON	Reserved	Reserved	LCON.1	LCON.0	W	s/r/N
$018	LMOD0	LCD mode register 0				W	N
$019	LMOD1	LCD mode register 1				W	N
$01A	-	Reserved				-	-

RAM Mapped Register (continued)

Register		Contents				Configuration	
Address	**Name**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**	**R/W**	**s/r/t/N**
$01B	-	Reserved				-	-
$01C	TC30 TL30	Timer 3 counter register 0 Timer 3 load register 0				R W	N
$01D	TC31 TL31	Timer 3 counter register 1 Timer 3 load register 1				R W	N
$01E	TC32 TL32	Timer 3 counter register 2 Timer 3 load register 2				R W	N
$01F	CTL0	KPRS (R)	ENBAT (W)	KPAD (W)	KRVS (W)	-	t/N
$020	SPA	Reserved	Reserved	Reserved	SP.10	R/W	t/N
$021	SPB	SP.9	SP.8	SP.7	SP.6	R/W	t/N
$022	SPC	SP.5	SP.4	SP.3	SP.2	R	t/N
$023	SENSE4	Reserved	Reserved	IET3 (R/W)	IRQT3 (R)	-	s/r/t r/t
$024	SENSE5	Reserved	Reserved	IE3 (R/W)	IRQ3 (R)	-	s/r/t r/t
$025	TMOD2	Timer 2 mode register				W	N
$026	TC20 TL20	Timer 2 counter register 0 Timer 2 load register 0				R W	N
$027	TC21 TL21	Timer 2 counter register 1 Timer 2 load register 1				R W	t/N
$028	KREG0	K03	K02	K01	K00	R/W	t/N
$029	KREG1	K13	K12	K11	K10	R/W	t/N
$02A	KREG2	K23	K22	K21	K20	R/W	t/N
$02B	KREG3	K33	K32	K31	K30	R/W	t/N
$02C	KREG4	K43	K42	K41	K40	R/W	t/N
$02D	KREG5	K53	K52	K51	K50	R/W	t/N
$02E	KREG6	K63	K62	K61	K60	R/W	t/N
$02F	KREG7	K73	K72	K71	K70	R/W	t/N
$030	P0	Port 0				R/W	t/N
$031	P1	Port 1				R/W	t/N
$032	P2	Port 2				R/W	t/N
$033	P3	Port 3				R/W	t/N
$034	P4	Port 4				R/W	t/N

RAM Mapped Register (continued)

Register		Contents				Configuration	
Address	**Name**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**	**R/W**	**s/r/t/N**
$035	P5	Port 5				R/W	t/N
$036	P6	Port 6				R/W	t/N
$037	P7	Port 7				R/W	t/N
$038	P8	Port 8				R/W	t/N
$039	-	Reserved				-	-
$03A	SGM	SGM.3	SGM.2	SGM.1	SGM.0	R/W	t/N
$03B	SGD	SGD.3	SGD.2	SGD.1	SGD.0	R/W	t/N
$03C	-	Reserved				-	-
$03D	BAT	Reserved	BAT.2	BAT.1	BAT.0	R	t/N
$03E	CVAR	CVAR.3	CVAR.2	CVAR.1	CVAR.0	W	N
$03F	-	Reserved				-	-

Notes:

s/r/t/N: s: Available for instructions, SM and SMD
r: Available for instructions, RM and RMD
t: Available for instructions, TM and TMD
N: Available for Nibble operations instructions, for example, ITMD

IE: Enable all Interrupts of this chip

IEn: Enable Interrupt n, IRQn: Interrupt request of $\overline{\text{INTn}}$

IETn: Enable Interrupt of Timer n, IRQTn: Interrupt request of Timer n

SP.y: Bit y of stack pointer, RSP: Reset Stack Pointer address

CKS: Clock selection register
CKS.0 and CKS.1 are used to select the system clock speed,
e.g. if CKS=xx01, then system clock = f_{MAIN}(1MHz)/8
if CKS=xx10, then system clock = f_{MAIN} (1MHz)/4

KPRS: It would be set when any key is pressed

KPAD: Enable Key pressed detection.
Once KPAD is set 1, Port2 will always be input no matter which status the Port2 I/O control register is.

KRVS: Reverse keyboard Scanning signal

IRM: IR mode register

SGM: Sound generator mode register

SGD: Sound generator data register

CVAR: Contrast control register (Write-only)

ENBAT: Enable low-battery detection when high

BAT.m: Battery power level data (Please refer to the Battery-low table)

(m = 0, 1, 2 n = 0, 1, 2, 3 x = 0, 1 y = 2, ... , 10)

Register and Flags

The NT93480 provides 9 registers and 2 flags for CPU operation. They are described below:

Accumulator A (4 bits), Register B (4 bits)

Accumulator A and the Register B are 4-bit registers that hold the results of the arithmetic logic unit (ALU). They are the very basic registers for a CPU to execute all the arithmetic calculation, logic operation and data transfer among memories, I/Os and registers.

Register V (4 bits)

Register V is a 4-bit register that holds the page address of RAM.

Register X (4 bits), Register Y (4 bits)

The register X and Y are 4-bit registers, which are used for indirect RAM addressing. For instance:

Bit 10~0	11	10	9	8	7	6	5	4	3	2	1	0
RAM address	V3	V2	V1	V0	X3	X2	X1	X0	Y3	Y2	Y1	Y0

Note: V3 must be set zero.

Register EX (4 bits), Register EY (4bits)

The register EX and EY are 4-bit registers available to assisting registers X and Y, respectively.

Carry Flag, CY (1bit)

The carry flag holds the ALU overflow bit generated by arithmetic operation. It can be set or reset by instruction directly, and is affected by the rotation instruction.

Status Flag, SF (1bit)

The flag holds the ALU comparison, arithmetic instruction and the status of accumulator. This flag would be tested by those instructions of conditional jump and conditional call. After execution of BR, LBR, CAL, or CALL instruction, the status bit would be set. Furthermore, SF will be pushed onto the stack during serving interrupt. Be care that SF will not be restored by instruction RET, but by instruction RETI.

Program Counter PC (14 bits)

The program counter is used for addressing ROM. It is combined of two address portions, page and offset:

Page (6 bits): pc13~8
Offset (8 bits): pc7~0

Stack pointer SP (11 bits)

The stack memory is implemented by means of RAM mapping so as to make the system become more extendible. SP is an 11-bits which holds the start address of the recent level of Stack. Stack is combined of sixteen 4-nibble registers, which holds the program counter of subroutines or interrupt return address. So, 16 levels of stack operation are possible. It can be initialized to the starting address ($23F) of the stack either by chip reset or by transfer instruction.

The following table shows how the stack register is configured:

Reset Stack Pointer Flag, RSP (1 bit)

RSP is used for resetting the stack pointer to $23F and will be cleared while system is in RESET mode. Usually, user needs not do anything about this flag unless he wants to. It is of course that user can reset the stack pointer by setting RSP to one. Afterward, it must be recovered to zero right away before doing anything else.

Stack Register Bits Configuration: (For level 1)

Address.	Bit 3		Bit 2		Bit 1		Bit 0	
$23F	S3	-PC3	S2	-PC2	S1	-PC1	S0	-PC0
$23E	S7	CY	S6	-PC6	S5	-PC5	S4	-PC4
$23D	S11	-PC10	S10	-PC9	S9	-PC8	S8	-PC7
$23C	S15	SF	S14	-PC13	S13	-PC12	S12	-PC11

Note: PC stored in the stack is a negative value

Memory Allocation of Stack Level

Level	Address	Level	Address	Level	Address	Level	Address
16	$203~$200	12	$213~$210	8	$223~$220	4	$233~$230
15	$207~$204	11	$217~$214	7	$227~$224	3	$237~$234
14	$20B~$208	10	$21B~$218	6	$22B~$228	2	$23B~$238
13	$20F~$20C	9	$21F~$21C	5	$22F~$22C	1	$23F~$23C

Interrupts

Nine interrupt sources are available for user:

1. General interrupts:
   - $\overline{INT0}$
   - $\overline{INT1}$
   - $\overline{INT2}$ (Port2)
   - $\overline{INT3}$
2. Timer/counter interrupts:
   - Timer0_output
   - Timer1_output
   - Timer2_output
   - Timer3_output
3. Serial interface interrupt:
   - SI

   (This signal is generated by the built-in serial communication port.)

Interrupt Control Bits and Request flags:

The RAM mapped register from $000 through $003 can only be accessed by bit operation instructions. Any interrupt, $\overline{INTn}$ may be enabled while IE and IEn were set. Flag IRQn will be set if there is an interrupt signal coming up and $\overline{INTn}$ has already been enabled. On the other hand, request flag IRWn can only be set by interrupt signal and also can only be cleared by system RESET.

Interrupt Enable (IE)

The interrupt enable bit, IE, enables all interrupts request when it is set. It can also be by instruction RETI and should be reset by any interrupt service routine.

General interrupts

$\overline{INT0}$, $\overline{INT1}$:

Interrupt request flags, IRQ0/IRQ1, will be set while interrupt input, $\overline{INT0}$ / $\overline{INT1}$, has received any negative edge signal provided that both IE and IE0/IE1 were set. Also, $\overline{INT0}$ or $\overline{INT1}$ can be programmed as an external clock input of Timer0 or Timer1 by the register, TMOD0 or TMOD1, respectively. In addition, IE0/IE1 must be cleared while making it as a timer clock.

$\overline{INT2}$:

The interrupt request flag IRQ2, will be set while any one of the interrupt inputs, $\overline{INT2}$ (Port2.0~3), has received any negative edge signal provided that both IE and IE2 were set. This interrupt function may be disabled by setting bit KPAD ($01F.1) to one. When KPAD is set. Port2 are no longer to be a general interrupt input but key-detection input.

$\overline{INT3}$:

The interrupt request flag IRQ3, will be set while any one of the interrupt input, $\overline{INT3}$ has received any negative edge signal provided that both IE and IE3 were set.

Timer/counter Interrupts

The interrupt request flag IRQTn, will be set by the output signal of Timer/counter n provided that both IE and IETn were set. (n=0, 1, 2, 3)

For details, please refer to the section on Timer/counter.

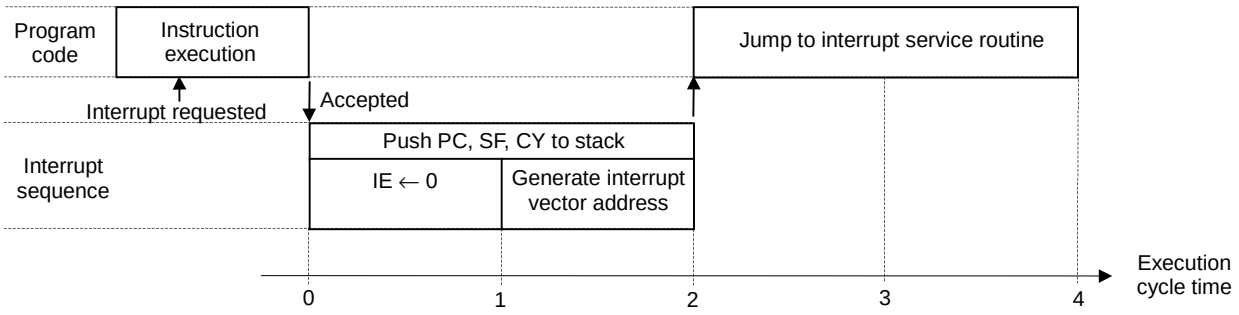
Serial Interface Interrupt

Interrupt request flag IRQSI, will be set by the internal serial interface signal provided that both IE and IESI were set.

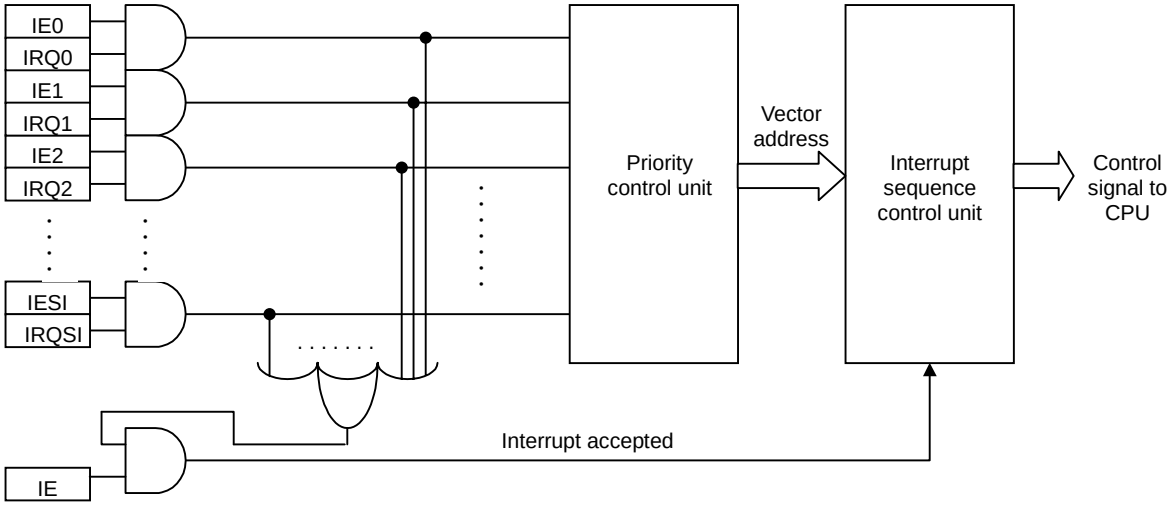
Interrupt Vector Address and execution priority

Interrupt	Signal	Priority	Issued by	Condition	Vector
RESET	External	1st	Reset pin	RESET PIN = 1 for at least 2 instruction cycles	$00
$\overline{INT0}$	External	2nd	IRQ0	IE = IE0 = IRQ0 = 1	$02
$\overline{INT1}$	External	3rd	IRQ1	IE = IE1 = IRQ1 = 1	$04
$\overline{INT2}$	External	4th	IRQ2	IE = IE2 = IRQ2 = 1	$06
$\overline{INT3}$	External	5th	IRQ3	IE = IE3 = IRQ3 = 1	$08
$\overline{TIMER0}$	Internal	6th	IRQT0	IE = IET0 = IRQT0 = 1	$0A
$\overline{TIMER1}$	Internal	7th	IRQT1	IE = IET1 = IRQT1 = 1	$0C
$\overline{TIMER2}$	Internal	8th	IRQT2	IE = IET2 = IRQT2 = 1	$0E
$\overline{TIMER3}$	Internal	9th	IRQT3	IE = IET3 = IRQT3 = 1	$10
SERIAL	Internal	10th	IRQSI	IE = IESI = IRQSI = 1	$12

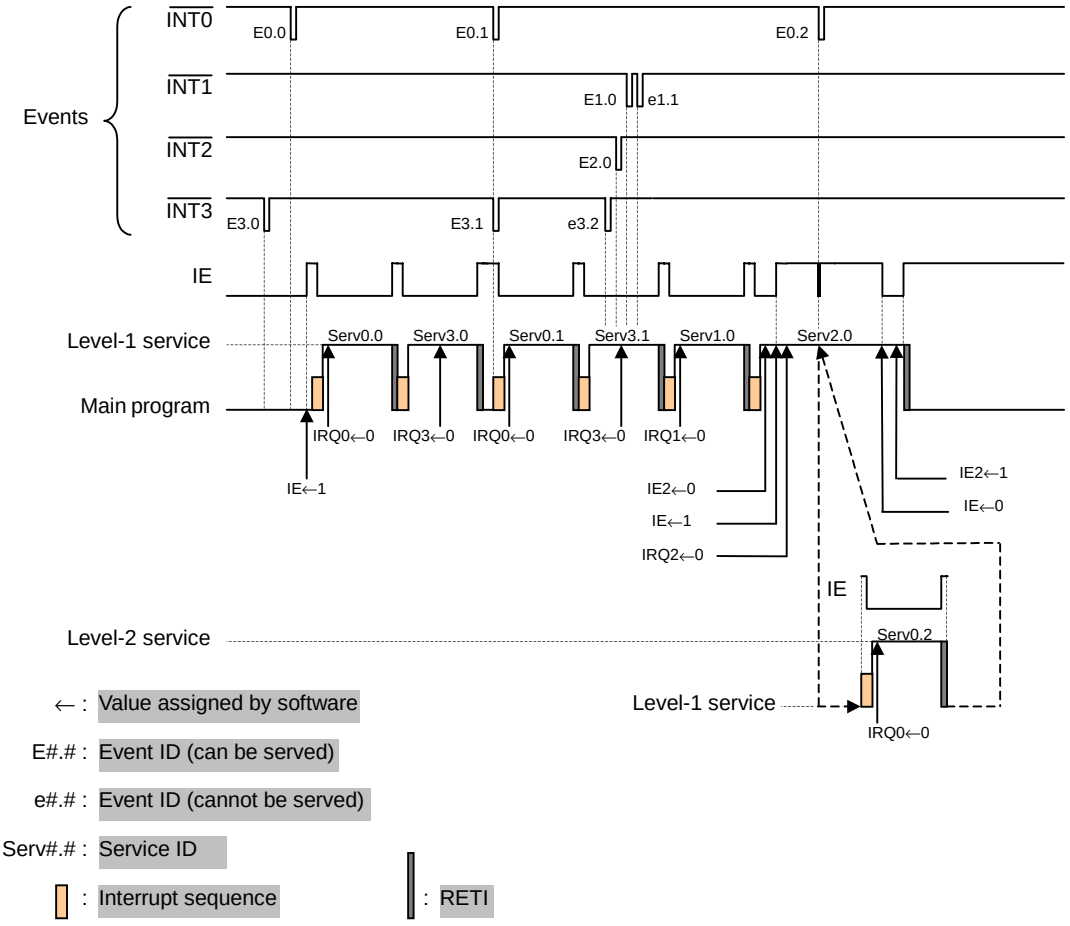
Timing of interrupt sequence



Hardware structure of interrupt control unit



Example of Timing of interrupt events



Input/Output Port

There are 36 I/O pins, which are divided into 9 groups. These pins contain pull-up MOS transistor that can be controlled by program. If these pins are configured as input pins, the pull-up transistor is determined by software. If any port pin is set to one, then pull-up transistor of that pin would be enabled. Otherwise, the pull-up transistor would be disconnected from that pin, Do not let any unused pin float because it may cause the noise problem to the chip. If you do so, it is better to put an external pull-up resistor (about 100KΩ) to that pin so as to keep the noise down.

Port Mode Register	Bit 3	Bit 2	Bit 1	Bit 0	Mode	
PMODA ($010)	PORT3	PORT2	Reserved	PORT0	0:Input	1:Output
PMODB ($011)	PORT7	PORT6	PORT5	Reserved	0:Input	1:Output
PMODC ($012)	Reserved	Reserved	Reserved	PORT8	0:Input	1:Output
PMODE ($014)	PORT1.3	PORT1.2	PORT1.1	PORT1.0	0:Input	1:Output
PMODF ($015)	PORT4.3	PORT4.2	PORT4.1	PORT4.0	0:Input	1:Output

The I/O pin PORT 0.2 is also the PSG and SERIAL output, so it is controlled by the special condition.

PORT 0.2 status	PSG				Serial SIOM.3($00C.3)	I/O Control PMODA.0($010.0)
	TnA(SGA1.3)	ENCHA(SGC.0)	TnB(SGB3.3)	ENCHB(SGC.1)		
Input by default	0	0	0	0	0	0
Normal output	1	0	X	0	0	1
	X	0	1	0	0	1
PSG output	X	1	X	X	0	0
	X	X	X	1	0	0
Serial output	X	X	X	X	1	0

The I/O pin PORT 0.3 is also the PSG output, so it is controlled by the special condition.

PORT 0.3 status	PSG				I/O Control PMODA.0($010.0)
	TnA(SGA1.3)	ENCHA(SGC.0)	TnB(SGB3.3)	ENCHB(SGC.1)	
Input by default	0	0	0	0	0
Normal output	1	0	X	0	1
	X	0	1	0	1
PSG output	X	1	X	X	X
	X	X	X	1	X

Notes: TnA: PSG Channel A mode register 1: Tone 0: Noise
TnB: PSG Channel B mode register 1: Tone 0: Noise
ENCHA: PSG Channel A enable register 1: Enable 0: Disable
ENCHB: PSG Channel B enable register 1: Enable 0: Disable
SIOM.3: Serial I/O enable register 1: Enable 0: Disable
PMODA.0: Port0 I/O control register 1: Output 0: Input

Port0 I/O control example:PORT 0.2 normal output

```
        ORG     0000H
        JMP     RESET
        ORG     0050H
RESET:  ITMD    08H, $03B       ; sound generator control mode: CH1=TONE
        ITMD    0BH, $03A       ; load data to SGA
        ITMD    00H, $03A       ; hold data
        ITMD    00H, $03B       ; disable PSG
        ITMD    08H, $03A       ; load data to SGC
        ITMD    00H, $03A       ; hold data
        ITMD    00H, $00C       ; disable SIO
        ITMD    0FH, $010       ; PORT0 = output
```

Serial Interface

The serial interface is basically an 8-bit Half-duplex Serial Transmitter/Receiver, which consists of two data registers (SIOL, SIOH), a serial mode register (SIOM), and an internal octal counter. During execution of STS instruction, the octal counter would be reset first and then it will increment by one at the rising edge of the transfer clock (SCK). However the octal counter would be reset and the serial interrupt flag will also be set for every 8 transfer clocks (SCK).

Serial Data Register

This 8-bit read/write serial data register consists of a lower nibble (SIOL) and an upper nibble (SIOH). The data stored in serial data register can be shifted out through the SO pin. Similarly, the external data stream can be shifted in and stored in SIOL & SIOH simultaneously via SI pin. Both input data and output data stream are synchronized by the falling edge of the transfer clock (SCK). Read/write operations of the serial data register must be performed after completion of data transfer. Otherwise, the data cannot be guaranteed.

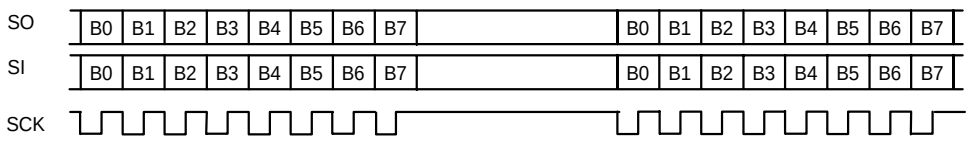
Serial Mode Register (SIOM ($00C))

It is a 4-bit, write-only register, which determines what the I/O mode of SCK is, prescaler divided ratio, and where the clock source from. It may be reset by RESET instruction. Be ware that writing data to SIOM will initialize the data transfer operation no matter whether it is being data transfer or not. It means that the transfer clock will stop, octal counter will be reset and serial interrupt request (IRQSI) will also be set during writing SIOM. Furthermore, execution of STS instruction is necessary after writing SIOM.

Serial mode register (SIOM, $00C)

SIOM.3	SIOM.2	SIOM.1	SIOM0	SCK	Clock source	Serial I/O
0	X	X	X	-	-	Disable
1	0	0	0	Input	External clock	Enable
1	0	0	1	Output	System clock	Enable
1	0	1	0	Output	System clock / 4	Enable
1	0	1	1	Output	System clock / 16	Enable
1	1	0	0	Output	System clock / 64	Enable
1	1	0	1	Output	System clock / 256	Enable
1	1	1	0	Output	System clock / 1024	Enable
1	1	1	1	Output	System clock / 4096	Enable

Serial Transmission Data Format



B7~4 ⟸⟹ SIOH ($00B.3~0);B3~0 ⟸⟹ SIOL ($00A.3~0).

IR communication port

This chip also offers an extremely convenient way of communication with other data equipment through the light media, Infrared. IR Mode register (IRM) is such a register that is responsible for controlling the IR communication data format, such as carrier frequency and logic.

IR Mode Register (IRM)

The user can define the transmission data format by means of programming the IR Mode Register. IRM.1 and IRM.0 define its carrier frequency.

IR Data Transmission

Data transmission of IR communication can be issued by means of programming the IR Mode Register (IRM). The output pin, IRO, will directly respond to the value in P1.2 with the format previously defined after Port1.2 has been defined as an output (PMODE.2=1). It is a static operation, so the output will not be changed until the value in P1.2 is changed. Obviously that this output waveform are fully software controlled. By employing a timer, the programmer can control the waveform and its frequency very accurately. Note that P1.2 will perform as a normal I/O pin when IRM.1=IRM.0=0.

IR Date Receiving

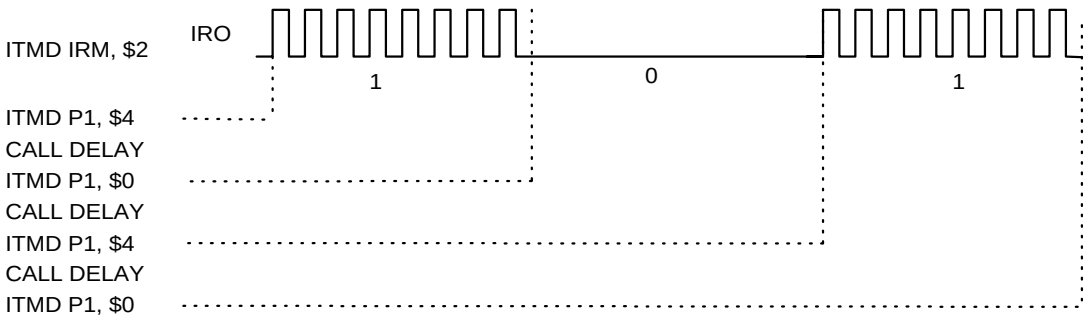
In addition, if pin $\overline{INT3}$ is employed to be the IR input pin, IRQ3 will be issued when there is IR incoming signal. Meanwhile bit P1.3 will directly respond to the logic level of pin $\overline{INT3}$, and CPU has enough time to capture this bit data for further processing at this moment. Note that PMODE.3 was cleared to zero in this case, so that the CPU can read the data from pin $\overline{INT3}$ /P1.3.

IR Mode Register (IRM)

PMODE.2	W1	W0	IRM.1	IRM.0	Output signal	
					P1.2=0	P1.2=1
0	X	X	X	X	High impedance	Pull-high
1	X	X	0	0	LOW	HIGH
1	X	X	0	1	LOW	35.7KHz
1	X	X	1	0	LOW	38.5KHz
1	X	X	1	1	LOW	500KHz

IR Transmission Timing and Data Format

EXAMPLE OF GENERATING A 38.5KHz CARRIER WAVEFORM WITH DATA STREAM 1.0.1.



Timers / Counters

There are 4 Timer/Counters (Timer0~3), in which prescaler and clock selection circuits have been built. By programming timer mode register TMOD0~TMOD3, user can choose different operation modes and speeds. They can also be configured as event counters.

1) TIMER 0 Operation: 8-bit Watch Timer

Timer0 is an 8-bit timer/counter, which consists of two 4-bit write only timer load registers (TL00, TL01) and two 4-bit read only timer/event counter registers (TC00, TC01). These registers share the same address $006 & $007. To program timer load register, user must write data to the lower nibble ($006) first, then the upper nibble ($007).

To read the data of the timer/counter register, the upper nibble ($007) must be read first, then the lower nibble ($006). The order of data reading and writing must be followed, otherwise this counter will not work probably due to the wrong data.

There are three kinds of clock sources that can be chosen by controlling the timer mode register0 (TMOD0, $004):

The first one is the system clock which can be programmed to fmain/4 or fmain/8, by writing 00, 01 or CKS.1~0 respectively.

The second one is the standard watch oscillator, 32768Hz/8.

The last one is the external source, $\overline{\text{INT0}}$.

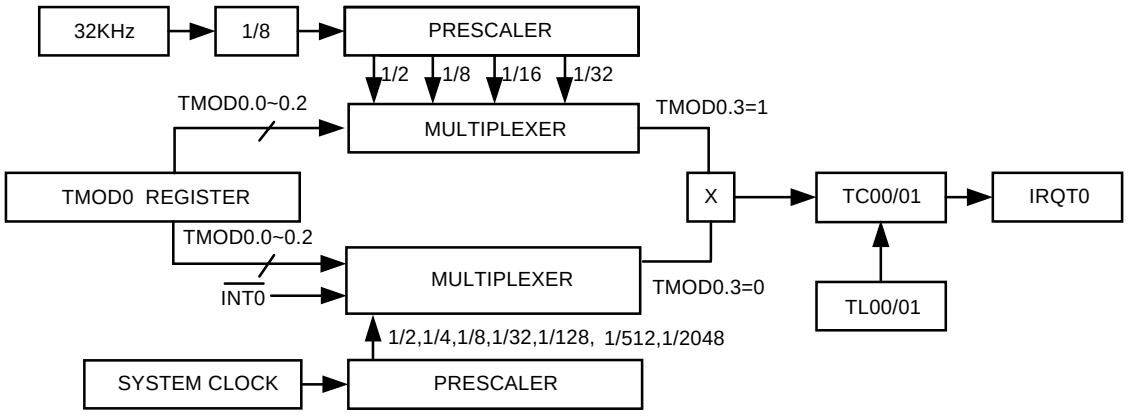
Interrupt request flag (IRQT0) will be set when the 8-bit up-counter has been overflowed (one counting more after $FF has been reached). Finally, reading will remain zero.

For the prescaler value, please refer to the following table:

Timer Mode 0 Register (TMOD0, $004)

TMOD0.3	TMOD0.2	TMOD0.1	TMOD0.0	Prescaler Divide Ratio	Clock Source
0	0	0	0	/2048	1MHz/M (M=4 or 8)
0	0	0	1	/512	1MHz/M (M=4 or 8)
0	0	1	0	/128	1MHz/M (M=4 or 8)
0	0	1	1	/32	1MHz/M (M=4 or 8)
0	1	0	0	/8	1MHz/M (M=4 or 8)
0	1	0	1	/4	1MHz/M (M=4 or 8)
0	1	1	0	/2	1MHz/M (M=4 or 8)
0	1	1	1	-	External source, $\overline{\text{INT0}}$
1	0	0	0	/32	4096Hz (32768Hz/8)
1	0	0	1	/16	4096Hz (32768Hz/8)
1	0	1	0	/8	4096Hz (32768Hz/8)
1	0	1	1	/2	4096Hz (32768Hz/8)
1	1	x	x	-	Reserved

Timer 0 Block Diagram



2) TIMER 1 Operation: 8-bit Timer/Counter

Auto-reload function has been implemented in Timer1. User can select different prescaler factor with or without auto-reload by putting the appropriate value into the timer mode register 1, TMOD1. Besides the internal clock source, 1MHz, an external clock source, $\overline{INT1}$, is also provided.

To program Timer 1 load register, user must write data to the lower nibble, TL10, first, then the TL11.

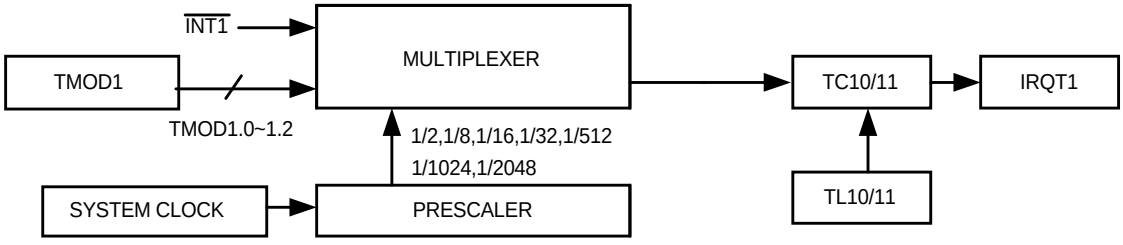
To read the data of this timer/counter register, the upper nibble, TC11 must be read out first, then the lower nibble, TC10. The order must be followed otherwise unexpected counting may occur.

Interrupt request flag (IRQT1) will be set when the 8-bit up-counter has been overflowed.

Timer Mode Register 1: (TMOD1 $005)

TMOD1.3	TMOD1.2	TMOD1.1	TMOD1.0	Prescaler Divide Ratio	Clock Source	Auto-Reload
0	0	0	0	/2048	1MHz/M (M=4 or 8)	No
0	0	0	1	/1024	1MHz/M (M=4 or 8)	No
0	0	1	0	/512	1MHz/M (M=4 or 8)	No
0	0	1	1	/32	1MHz/M (M=4 or 8)	No
0	1	0	0	/16	1MHz/M (M=4 or 8)	No
0	1	0	1	/8	1MHz/M (M=4 or 8)	No
0	1	1	0	/2	1MHz/M (M=4 or 8)	No
0	1	1	1	-	External source, $\overline{INT1}$	No
1	0	0	0	/2048	1MHz/M (M=4 or 8)	Yes
1	0	0	1	/1024	1MHz/M (M=4 or 8)	Yes
1	0	1	0	/512	1MHz/M (M=4 or 8)	Yes
1	0	1	1	/32	1MHz/M (M=4 or 8)	Yes
1	1	0	0	/16	1MHz/M (M=4 or 8)	Yes
1	1	0	1	/8	1MHz/M (M=4 or 8)	Yes
1	1	1	0	/2	1MHz/M (M=4 or 8)	Yes
1	1	1	1	-	External source, $\overline{INT1}$	Yes

TIMER 1 Block Diagram



3) TIMER 2 Operation: 8-bit Timer/Counter

Auto-reload function has been implemented in Timer2. User can select different prescaler factor with or without auto-reload by putting the appropriate value into the timer mode register 2, TMOD2. Only the internal clock source, 1MHz, is provided for Timer 2.

To program Timer 2 load register, user must write data to the lower nibble, TL20, first, then the TL21.

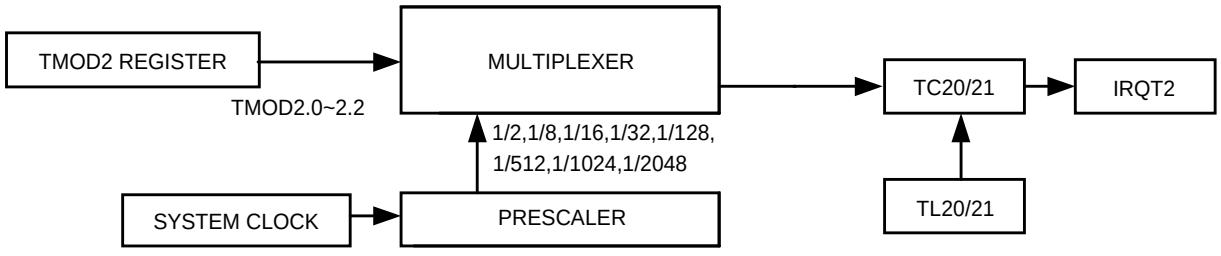
To read the data of this timer/counter register, the upper nibble, TC21 must be read out first, then the lower nibble, TC20. The order must be followed otherwise unexpected counting may occur.

Interrupt request flag (IRQT2) will be set when the 8-bit up-counter has been overflowed.

Timer Mode Register 2: (TMOD2 $0025)

TMOD2.3	TMOD2.2	TMOD2.1	TMOD2.0	Prescaler Divide Ratio	Clock Source	Auto-Reload
0	0	0	0	/2048	1MHz/M (M=4 or 8)	No
0	0	0	1	/1024	1MHz/M (M=4 or 8)	No
0	0	1	0	/512	1MHz/M (M=4 or 8)	No
0	0	1	1	/128	1MHz/M (M=4 or 8)	No
0	1	0	0	/32	1MHz/M (M=4 or 8)	No
0	1	0	1	/16	1MHz/M (M=4 or 8)	No
0	1	1	0	/8	1MHz/M (M=4 or 8)	No
0	1	1	1	/2	1MHz/M (M=4 or 8)	No
1	0	0	0	/2048	1MHz/M (M=4 or 8)	Yes
1	0	0	1	/1024	1MHz/M (M=4 or 8)	Yes
1	0	1	0	/512	1MHz/M (M=4 or 8)	Yes
1	0	1	1	/128	1MHz/M (M=4 or 8)	Yes
1	1	0	0	/32	1MHz/M (M=4 or 8)	Yes
1	1	0	1	/16	1MHz/M (M=4 or 8)	Yes
1	1	1	0	/8	1MHz/M (M=4 or 8)	Yes
1	1	1	1	/2	1MHz/M (M=4 or 8)	Yes

Timer 2 block Diagram



4) TIMER 3 Operation: 12-bit Timer/Counter

Auto-reload function has been implemented in Timer3. User can select different prescaler factor with or without auto-reload by putting the appropriate value into the timer mode register 3, TMOD3. Only the internal clock source, 1MHz, is provided for Timer 3.

To program Timer 3 load register, user must write data to the lower nibble, TL30, first, then the TL31, and finally the TL32.

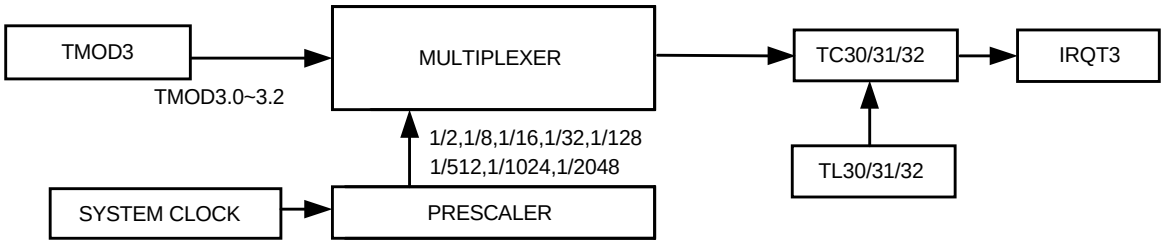
To read the data of this Timer/Counter register, the upper nibble, TC32 must be read out first, then the TC20, and finally the TC30. This order must be followed otherwise unexpected counting may occur.

Interrupt request flag (IRQT3) will be set when the 12-bit up-counter has been overflowed.

Timer Mode Register 3: (TMOD3 $013)

TMOD3.3	TMOD3.2	TMOD3.1	TMOD3.0	Prescaler Divide Ratio	Clock Source	Auto-Reload
0	0	0	0	/2048	1MHz/M (M=4 or 8)	No
0	0	0	1	/1024	1MHz/M (M=4 or 8)	No
0	0	1	0	/512	1MHz/M (M=4 or 8)	No
0	0	1	1	/128	1MHz/M (M=4 or 8)	No
0	1	0	0	/32	1MHz/M (M=4 or 8)	No
0	1	0	1	/16	1MHz/M (M=4 or 8)	No
0	1	1	0	/8	1MHz/M (M=4 or 8)	No
0	1	1	1	/2	1MHz/M (M=4 or 8)	No
1	0	0	0	/2048	1MHz/M (M=4 or 8)	Yes
1	0	0	1	/1024	1MHz/M (M=4 or 8)	Yes
1	0	1	0	/512	1MHz/M (M=4 or 8)	Yes
1	0	1	1	/128	1MHz/M (M=4 or 8)	Yes
1	1	0	0	/32	1MHz/M (M=4 or 8)	Yes
1	1	0	1	/16	1MHz/M (M=4 or 8)	Yes
1	1	1	0	/8	1MHz/M (M=4 or 8)	Yes
1	1	1	1	/2	1MHz/M (M=4 or 8)	Yes

Timer 3 block Diagram



Sound Generator

This built-in sound generator is a programmable dual tone signal generator that offers whatever a single tone, dual tone or single tone with noise signal, Loading different pattern down to the generator registers gives different effect of sound generation. In addition, the tone frequencies can be controlled by programming the Sound Generator Data Register (SGD) through Sound Generator Mode Register (SGM). In addition, the BUZ pin will keep low, and $\overline{BUZ}$ pin keep high, when VOL0=VOL1=0, channel A or B enable.

To down load a data to SGF, for example, the user must make sure that SGM.3 is zero. Afterward, the required data can be written to SGD. Next, the data in SGD has to be transferred directly by writing a $9. Finally, SGM.3 must be cleared to hold the data.

For example:

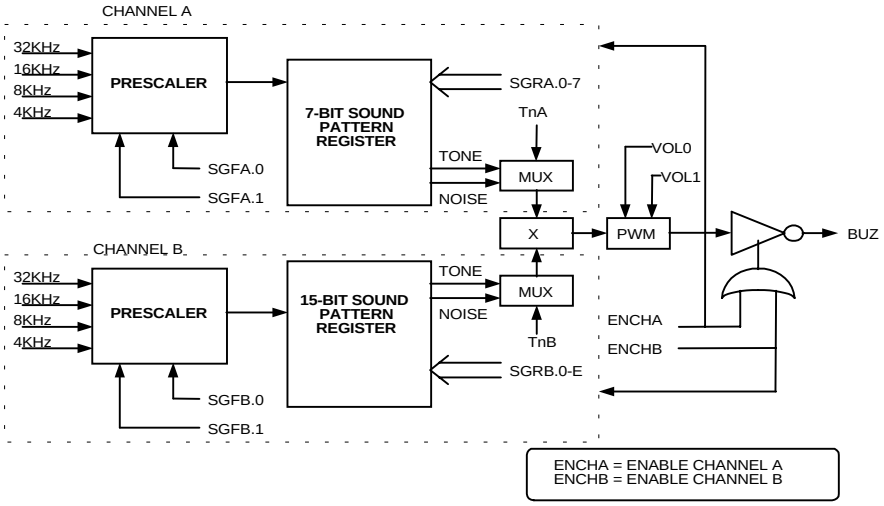
```
ITMD  $0, SGM     ; Clear SGM.3
ITMD  $D, SGD     ; 8KHz for CHA, 32KHz for CHB
ITMD  $9, SGM     ; Load data to SGF
ITMD  $0, SGM     ; Hold data by clear SGM.3
```

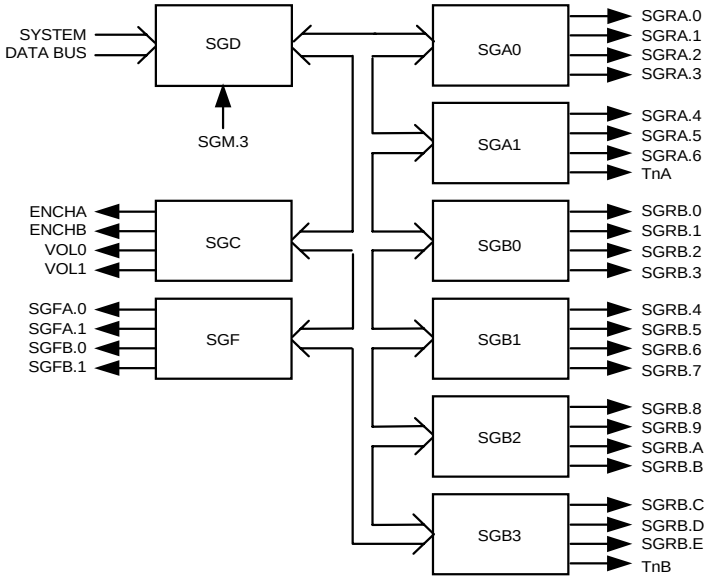
Sound Generator Mode Register: SGM

SGM.3	SGM.2	SGM.1	SGM.0	Data strobe	SG Data Register (SGD)
0	x	x	x	Hold data	x
1	0	0	0	Load data	SGC
1	0	0	1	Load data	SGF
1	0	1	0	Load data	SGA0
1	0	1	1	Load data	SGA1
1	1	0	0	Load data	SGB0
1	1	0	1	Load data	SGB1
1	1	1	0	Load data	SGB2
1	1	1	1	Load data	SGB3

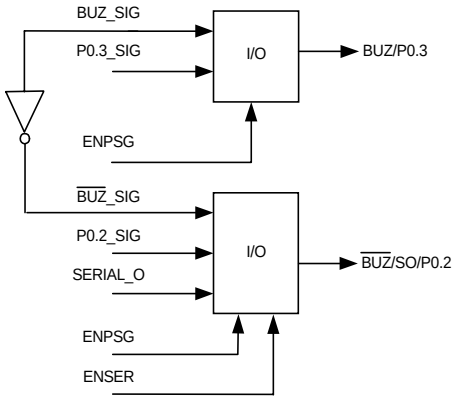
Sound Generator Data Register: SGD

SG Data Register	SGC.3	SGC.2	SGC.1	SGC.0	Descriptions
SGC	X	X	X	ENCHA	Enable channel A (1:Enable 0:disable)
	X	X	ENCHB	X	Enable channel B (1:Enable 0:disable)
	VOL1	VOL0	X	X	4-level volume control of sound output
	0	0			level 0 (BUZ keeps low)
	0	1			Level 1
	1	0			Level 2
	1	1			Level 3 (Maximum)
SGF	X	X	SGFA.1	SGFA.0	4-option clock source for 7-bit generator, 0:2KHz, 1:4KHz, 2:8KHz, 3:16KHz
	SGFB.1	SGFB.0	X	X	4-option clock source for 15-bit generator, 0:2KHz, 1:4KHz, 2:8KHz, 3:16KHz
SGA0	SGRA.3	SGRA.2	SGRA.1	SGRA.0	Nibble 0 of 7-bit generator register
SGA1	X	SGRA.6	SGRA.5	SGRA.4	Nibble 1 of 7-bit generator register
	TnA	X	X	X	Tone selection for channel A, 1 for Tone, 0 for Noise
SGB0	SGRB.3	SGRB.2	SGRB.1	SGRB.0	Nibble 0 of 15-bit generator register
SGB1	SGRB.7	SGRB.6	SGRB.5	SGRB.4	Nibble 1 of 15-bit generator register
SGB2	SGRB.B	SGRB.A	SGRB.9	SGRB.8	Nibble 2 of 15-bit generator register
SGB3	X	SGRB.E	SGRB.D	SGRB.C	Nibble 3 of 15-bit generator register
	TnB	X	X	X	Tone selection for channel B, 1 for Tone, 0 for Noise





P0.2 & P0.3 function



ENPSG	P0.3
0	P0.3_SIG
1	BUZ_SIG

ENPSG	ENSER	P0.2
0	0	P0.2_SIG
0	1	SERIAL_O
1	0	$\overline{BUZ}$ _SIG
1	1	SERIAL_O

Note: BUZ and $\overline{BUZ}$ is floating for application,when TnA=TnB=0 & P0.2=P0.3=0,PSG and SERIAL function disable.
To get $\overline{BUZ}$ _SIG or SERIAL_O, port0 must be set to input mode.

Music Table 1

Following is the music scale reference table for channel A (or channel B) under OSC=1MHz.

Frequency = 1M /2 /256 (1953.13Hz) **(SGFA.1=0, SGFA.0=0 or SGFB.1=0, SGFB.0=0)**						**Frequency = 1M /2 /128 (3906.25Hz)** **(SGFA.1=0, SGFA.0=1 or SGFB.1=0, SGFB.0=1)**					
Note	**Ideal**	**Real1**	**N1=1953.13 / Ideal**	**|N1|**	**Error%**	**Note**	**Ideal**	**Real1**	**N1=3906.25 / Ideal**	**|N1|**	**Error%**
B1	61.74	61.04	31.63	32	-1.14%	**B1**	61.74	62.00	63.27	63	0.42%
C2	65.41	65.10	29.86	30	-0.47%	**C2**	65.41	65.10	59.72	60	-0.47%
D2	73.42	72.34	26.60	27	-1.47%	**D2**	73.42	73.70	53.20	53	0.39%
E2	82.41	81.38	23.70	24	-1.25%	**E2**	82.41	83.11	47.40	47	0.85%
F2	87.31	88.78	22.37	22	1.68%	**F2**	87.31	86.81	44.74	45	-0.58%
G2	98.00	97.66	19.93	20	-0.35%	**G2**	98.00	97.66	39.86	40	-0.35%
A2	110.00	108.51	17.76	18	-1.36%	**A2**	110.00	108.51	35.51	36	-1.36%
B2	123.47	122.07	15.82	16	-1.13%	**B2**	123.47	122.07	31.64	32	-1.13%
C3	130.81	130.21	14.93	15	-0.46%	**C3**	130.81	130.21	29.86	30	-0.46%
D3	146.83	150.24	13.30	13	2.32%	**D3**	146.83	144.68	26.60	27	-1.47%
E3	164.81	162.76	11.85	12	-1.24%	**E3**	164.81	162.76	23.70	24	-1.24%
F3	174.61	177.56	11.19	11	1.69%	**F3**	174.61	177.56	22.37	22	1.69%
G3	196.00	195.31	9.96	10	-0.35%	**G3**	196.00	195.31	19.93	20	-0.35%
A3	220.00	217.01	8.88	9	-1.36%	**A3**	220.00	217.01	17.76	18	-1.36%
B3	246.94	244.14	7.91	8	-1.13%	**B3**	246.94	244.14	15.82	16	-1.13%
E4	329.63	325.52	5.93	6	-1.25%	**C4**	261.63	260.42	14.93	15	-0.46%
G4	392.00	390.63	4.98	5	-0.35%	**D4**	293.66	300.48	13.30	13	2.32%
B4	493.88	488.28	3.95	4	-1.13%	**E4**	329.63	325.52	11.85	12	-1.25%
E5	659.26	651.04	2.96	3	-1.25%	**F4**	349.23	355.11	11.19	11	1.68%
B5	987.77	976.57	1.98	2	-1.13%	**G4**	392.00	390.63	9.96	10	-0.35%
B6	1975.53	1953.13	0.99	1	-1.13%	**A4**	440.00	434.03	8.88	9	-1.36%
						B4	493.88	488.28	7.91	8	-1.13%
						B4	493.88	488.28	7.91	8	-1.13%
						E5	659.26	651.04	5.93	6	-1.25%
						B5	987.77	976.56	3.95	4	-1.13%
						B6	1975.53	1953.13	1.98	2	-1.13%
						B7	3951.07	3906.25	0.99	1	-1.13%

* The value **|N1|** of divider (7 Bit LSFR) is corresponding to the Reg **SGRA6~SGRA0** or **SGRBE~SGRB8** as that will be shown at **Music Table 3**.

Music Table 2

Frequency = 1M /2 /64 (7812.5Hz)
(SGFA.1=1, SGFA.0=0 or SGFB.1=1, SGFB.0=0)

Frequency = 1M /2 /32 (15625Hz)
(SGFA.1=1, SGFA.0=1 or SGFB.1=1, SGFB.0=1)

Note	Ideal	Real1	N1=7812.5 / Ideal	|N1|	Error%	Note	Ideal	Real1	N1=15625 / Ideal	|N1|	Error%
B1	61.74	61.52	126.54	127	-0.36%	**B2**	123.47	123.03	126.55	127	-0.36%
C2	65.41	65.65	119.44	119	0.37%	**C3**	130.81	131.30	119.45	119	0.38%
D2	73.42	73.70	106.41	106	0.39%	**D3**	146.83	147.41	106.42	106	0.39%
E2	82.41	82.24	94.80	95	-0.21%	**E3**	164.81	164.47	94.81	95	-0.20%
F2	87.31	87.78	89.48	89	0.54%	**F3**	174.61	175.56	89.49	89	0.55%
G2	98.00	97.66	79.72	80	-0.35%	**G3**	196.00	195.31	79.72	80	-0.35%
A2	110.00	110.04	71.02	71	0.03%	**A3**	220.00	220.07	71.02	71	0.03%
B2	123.47	124.01	63.27	63	0.44%	**B3**	246.94	248.02	63.27	63	0.44%
C3	130.81	130.21	59.72	60	-0.46%	**C4**	261.63	260.42	59.72	60	-0.46%
D3	146.83	147.41	53.21	53	0.39%	**D4**	293.66	294.81	53.21	53	0.39%
E3	164.81	166.22	47.40	47	0.86%	**E4**	329.63	332.45	47.40	47	0.85%
F3	174.61	173.61	44.74	45	-0.57%	**F4**	349.23	347.22	44.74	45	-0.57%
G3	196.00	195.31	39.86	40	-0.35%	**G4**	392.00	390.63	39.86	40	-0.35%
A3	220.00	217.01	35.51	36	-1.36%	**A4**	440.00	434.03	35.51	36	-1.36%
B3	246.94	244.14	31.64	32	-1.13%	**B4**	493.88	488.28	31.64	32	-1.13%
C4	261.63	260.42	29.86	30	-0.46%	**C5**	523.25	520.83	29.86	30	-0.46%
D4	293.66	289.35	26.60	27	-1.47%	**D5**	587.33	578.70	26.60	27	-1.47%
E4	329.63	325.52	23.70	24	-1.25%	**E5**	659.26	651.04	23.70	24	-1.25%
F4	349.23	355.11	22.37	22	1.68%	**F5**	698.46	710.23	22.37	22	1.68%
G4	392.00	390.63	19.93	20	-0.35%	**G5**	783.99	781.25	19.93	20	-0.35%
A4	440.00	434.03	17.76	18	-1.36%	**A5**	880.00	868.06	17.76	18	-1.36%
B4	493.88	488.28	15.82	16	-1.13%	**B5**	987.77	976.56	15.82	16	-1.13%
B4	493.88	488.28	15.82	16	-1.13%	**C6**	1046.50	1041.67	14.93	15	-0.46%
C5	523.25	520.83	14.93	15	-0.46%	**D6**	1174.66	1201.92	13.30	13	2.32%
D5	587.33	600.96	13.30	13	2.32%	**E6**	1318.51	1302.08	11.85	12	-1.25%
E5	659.26	651.04	11.85	12	-1.25%	**F6**	1396.91	1420.45	11.19	11	1.69%
F5	698.46	710.23	11.19	11	1.68%	**G6**	1567.98	1562.50	9.97	10	-0.35%
G5	783.99	781.25	9.97	10	-0.35%	**A6**	1760.00	1736.11	8.88	9	-1.36%
A5	880.00	868.06	8.88	9	-1.36%	**B6**	1975.53	1953.13	7.91	8	-1.13%
B5	987.77	976.56	7.91	8	-1.13%	**E7**	2637.02	2604.17	5.93	6	-1.25%
						G7	3135.96	3125.00	4.98	5	-0.35%
						B7	3951.07	3906.25	3.95	4	-1.13%

* The value **|N1|** of divider (7 Bit LSFR) is corresponding to the Reg **SGRA6~SGRA0** or **SGRBE~SGRB8** as that will be shown at **Music Table 3**.

Music Table 3.

SGRA (.6~.0) or SGRB (.E~.8)	|N1|	SGRA (.6~.0) or SGRB (.E~.8)	|N1|	SGRA (.6~.0) or SGRB (.E~.8)	|N1|	SGRA (.6~.0) or SGRB (.E~.8)	|N1|	SGRA (.6~.0) or SGRB (.E~.8)	|N1|	SGRA (.6~.0) or SGRB (.E~.8)	|N1|	SGRA (.6~.0) or SGRB (.E~.8)	|N1|	SGRA (.6~.0) or SGRB (.E~.8)	|N1|
01	127	14	111	16	95	3E	79	12	63	6F	47	4B	31	15	15
02	126	28	110	2C	94	7D	78	24	62	5E	46	17	30	2A	14
04	125	51	109	59	93	7A	77	49	61	3D	45	2E	29	55	13
08	124	23	108	33	92	74	76	13	60	7B	44	5D	28	2B	12
10	123	47	107	67	91	68	75	26	59	76	43	3B	27	57	11
20	122	0F	106	4E	90	50	74	4D	58	6C	42	77	26	2F	10
41	121	1E	105	1D	89	21	73	1B	57	58	41	6E	25	5F	9
03	120	3C	104	3A	88	43	72	36	56	31	40	5C	24	3F	8
06	119	19	103	75	87	07	71	6D	55	63	39	39	23	7F	7
0C	118	72	102	6A	86	0E	70	5A	54	46	38	73	22	7E	6
18	117	64	101	54	85	1C	69	35	53	0D	37	66	21	7C	5
30	116	48	100	29	84	38	68	6B	52	1A	36	4C	20	78	4
61	115	11	99	53	83	71	67	56	51	34	35	19	19	70	3
42	114	22	98	27	82	62	66	2D	50	69	34	32	18	60	2
05	113	45	97	4F	81	44	65	5B	49	52	33	65	17	40	1
0A	112	0B	96	1F	80	09	64	37	48	25	32	4A	16		

Liquid Crystal Display (LCD)

NT93480 can directly drive an LCD panel of up to 320 dots (40 segments x 8 commons). LCD driver contains:

-LCD controller/driver
-Display RAM for storing display data ($100~ $1FF)
$1n0~$1n3 empty space (no memory cell)
$1n4~$1nD LCD RAM
$1nE~$1nF empty space (no memory cell)
$1m0~$1mF empty space (no memory cell)
(n=0.1.2~7, m=8.9.A~F)
-40 segment output pins (SEG0~ SEG39)
-8 common output pins (COM0~ COM7)
-16 level contrast control

LCD control register, LCON, is used to turn the LCD display on and off, so as to save energy in some cases. Also, it can control the LCD bias voltage to accommodate different type of LCD.

LCD mode register 0, LMOD0, is in charge of controlling the frame frequency of LCD and the display modes.

Register LMOD1 offer 5 choices of total number of segment to serve different application requirement.. In addition, When LCON.0 is set, the LCD would always be enabled even in STOP or HALT mode.

LCD Control Register: LCON ($017) and LMOD1.0 ($19.0)

LMOD1.0	LCON.3	LCON.2	LCON.1	LCON.0	Function			
X	X	X	X	0	Disable LCD (LCD Power off)			
X	X	X	X	1	Enable LCD (LCD Power on)			
					Duty	**Bias**	**COM0~3**	**COM4~7**
0	0	0	1	X	1/8	1/4	COM Pin	COM Pin
0	0	1	1	X	1/8	1/3	COM Pin	COM Pin
0	1	0	1	X	1/4	1/3	COM Pin	Pull up to V_0
0	1	1	1	X	1/4	1/3	COM Pin	
X	X	X	0	X	inhibited			
1	X	X	X	X	inhibited			

LCD Mode Register 0: LMOD0 ($018)

LMOD0.3	LMOD0.2	LMOD0.1	LMOD0.0	Function
X	X	0	0	All LCD dots off (blank)
X	X	0	1	All LCD dots on
X	X	1	0	Normal display
X	X	1	1	Normal display
0	0	X	X	Frame Frequency = 32Hz
0	1	X	X	Frame Frequency = 64Hz
1	0	X	X	Frame Frequency = 128Hz
1	1	X	X	Frame Frequency = 256Hz

LCD Mode Register 1: LMOD1 ($019)

LMOD1.3	LMOD1.2	LMOD1.1	SEG24~27	SEG28~31	SEG32~35	SEG36~39	Total No. of Segments
0	0	0	SEG port	SEG port	SEG port	SEG port	40
0	0	1	SEG port	SEG port	SEG port	Port5	36
0	1	0	SEG port	SEG port	Port6	Port5	32
0	1	1	SEG port	Port7	Port6	Port5	28
1	x	x	Port8	Port7	Port6	Port5	24

LCD RAM Mapped Address

LCD data RAM ($104~$17D) is of dual port control; data can be transferred to segment pins automatically without a lot of complicated software control statements. So, after the bit value of a display segment is set, the LCD display will be turned on. Similarly, after the bit value is reset to zero, it will be turned off again, The following diagram shows the configuration of the RAM Mapped of LCD. Each bit in that area represents a segment value, SEGn, corresponding to a COMn and can be set (on) or reset (off) by bit or nibble operation instructions.

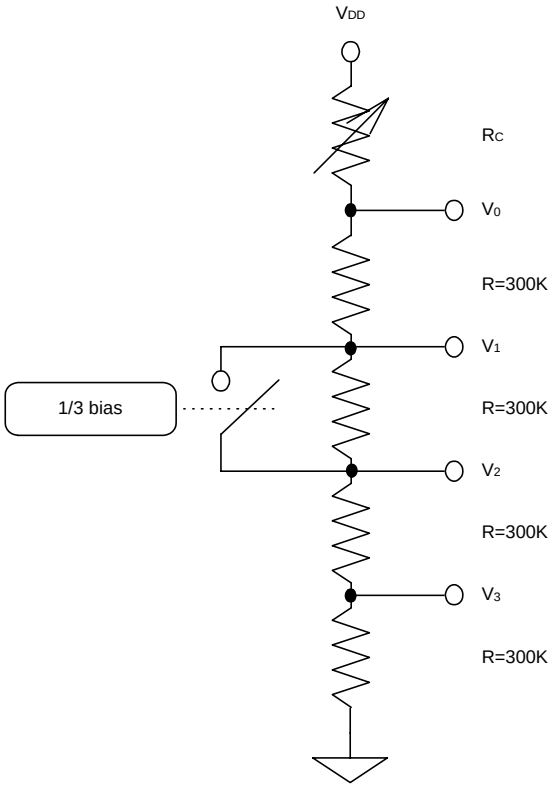
LCD RAM Area Configuration

COM0					COM1					.....	COM7				
Addr	B3	B2	B1	B0	Addr	B3	B2	B1	B0	.....	Addr	B3	B2	B1	B0
100h	-	-	-	-	110h	-	-	-	-	.....	170h	-	-	-	-
101h	-	-	-	-	111h	-	-	-	-	.....	171h	-	-	-	-
102h	-	-	-	-	112h	-	-	-	-	.....	172h	-	-	-	-
103h	-	-	-	-	113h	-	-	-	-	.....	173h	-	-	-	-
104h	SEG3	SEG2	SEG1	SEG0	114h	SEG3	SEG2	SEG1	SEG0	.....	174h	SEG3	SEG2	SEG1	SEG0
105h	SET7	SEG6	SEG5	SEG4	115h	SET7	SEG6	SEG5	SEG4	.....	175h	SET7	SEG6	SEG5	SEG4
106h	SEG11	SEG10	SEG9	SEG8	116h	SEG11	SEG10	SEG9	SEG8	.....	176h	SEG11	SEG10	SEG9	SEG8
107h	SEG15	SEG14	SEG13	SEG12	117h	SEG15	SEG14	SEG13	SEG12	.....	177h	SEG15	SEG14	SEG13	SEG12
108h	SEG19	SEG18	SEG17	SEG16	118h	SEG19	SEG18	SEG17	SEG16	.....	178h	SEG19	SEG18	SEG17	SEG16
109h	SEG23	SEG22	SEG21	SEG20	119h	SEG23	SEG22	SEG21	SEG20	.....	179h	SEG23	SEG22	SEG21	SEG20
10Ah	SEG27	SEG26	SEG25	SEG24	11Ah	SEG27	SEG26	SEG25	SEG24	.....	17Ah	SEG27	SEG26	SEG25	SEG24
10Bh	SEG31	SEG30	SEG29	SEG28	11Bh	SEG31	SEG30	SEG29	SEG28	.....	17Bh	SEG31	SEG30	SEG29	SEG28
10Ch	SEG35	SEG34	SEG33	SEG32	11Ch	SEG35	SEG34	SEG33	SEG32	.....	17Ch	SEG35	SEG34	SEG33	SEG32
10Dh	SEG39	SEG38	SEG37	SEG36	11Dh	SEG39	SEG38	SEG37	SEG36	.....	17Dh	SEG39	SEG38	SEG37	SEG36
10Eh	-	-	-	-	11Eh	-	-	-	-	.....	17Eh	-	-	-	-
10Fh	-	-	-	-	11Fh	-	-	-	-	.....	17Fh	-	-	-	-

Notes: $1n0~$1n3 empty space (no memory cell)
$1n4~$1nD LCD RAM
$1nE~$1nF empty space (no memory cell)
$1m0~$1mF empty space (no memory cell)
(n=0.1.2~7, m=8.9.A~F)

LCD Contrast Control

LCD contrast can easily be adjusted by software. Physically, as you can see in the circuit below, different V_0 can give different LCD contrast. Similarly, different value of resister Rc can give different V_0. So, people can change the value of Rc by programming the register, CVAR ($03E). On the table below, it lists all the Rc values according to different programming value of register, CVAR. Be ware that the resultants are not in simple ascending order.



CVAR.3	CVAR.2	CVAR.1	CVAR.0	Rc (Ω)
0	0	0	0	210K
0	0	0	1	180K
0	0	1	0	150K
0	0	1	1	120K
0	1	0	0	90K
0	1	0	1	60K
0	1	1	0	30K
0	1	1	1	0
1	0	0	0	240K
1	0	0	1	270K
1	0	1	0	300K
1	0	1	1	330K
1	1	0	0	360K
1	1	0	1	390K
1	1	1	0	420K
1	1	1	1	450K

Low Power Consumption Modes

To save power, user can issue one of the Low Power consumption modes by instructions, STOP or HAIT. Both of these modes can make the CPU sleeping. It means that CPU does nothing anyway until an external interrupt or a reset signal comes up. STOP mode will save more power than that of the others (HALT or NORMAL), however, it takes a little bit longer to wake up the CPU due to the setting time for main oscillator.

Operation Mode	Issued by instruction	Main oscillator	Watch oscillator	RAM	Register & Flag	I/O	Released by
STOP Mode	STOP	Stops	Alive	Hold	Hold	Hold	Reset, $\overline{INT0}$ ~ $\overline{INT3}$ TIMER0 , Serial I/O
HALT Mode	HALT	Alive	Alive	Hold	Hold	Hold	Any interrupt or reset

Battery-low Detection

To monitor power consumption, the function of battery-low detection is enabled by setting ENBAT ($01F.2) to one. The Battery-low detection table shows that the 3bits, BAT.2~0 ($03D.2~0), would respond while the battery power drops to a certain rank of voltage level.

* : For reference only

Battery-low detection table

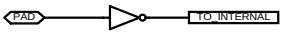
Power Supply Level	BAT.2	BAT.1	BAT.0
*4.3V < V_{DD}	0	0	0
*3.2V < V_{DD} ≦ *4.3V	1	0	0
*2.3V < V_{DD} ≦ *3.2V	1	1	0
V_{DD} ≦ *2.3V	1	1	1

Pin definition with its structure

TYPE 1

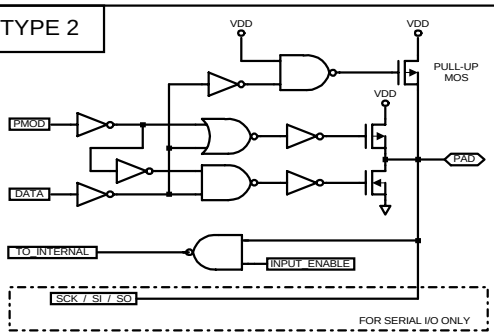
RESET

TYPE 4



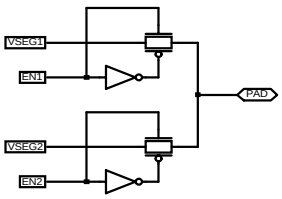
TEST

TYPE 2



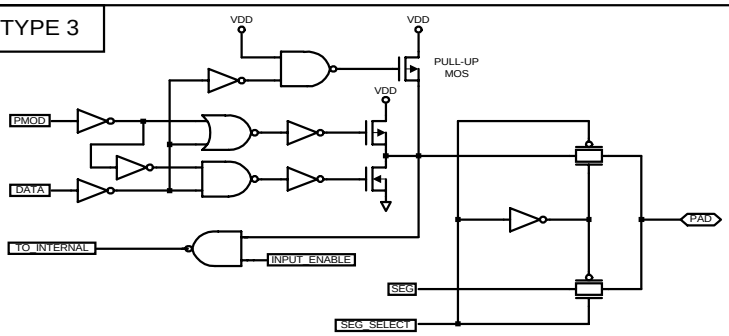
PORT0, PORT1, PORT2, PORT3, PORT4

TYPE 5



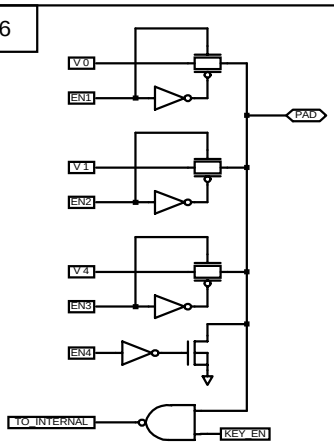
SEG0 to SEG23

TYPE 3



PORT5, PORT6, PORT7 & PORT8

TYPE 6



COM0 to COM7

Reset

To reset NT93480, the reset pin should be held high at least two instruction cycles. The following table shows the initial conditions for different operation mode after RESET.

Hardware	RESET Input while in HALT/ STOP mode	RESET Input while in Normal mode
Program Counter (PC)	$0000	$0000
Carry Flag (CY)	Retained	Undefined
Stack Pointer (SP)	$023F	$023F
Status Flag (SF)	1	1
RAM	Retained	Undefined
Register (A, B, V, Y, EX, EY)	Retained	Undefined
I/O Ports - Output latch status - Output Pull-up status - I/O mode registers	 1 1 0	 1 1 0
Timer/Counter - Counters (TCxx) - Loaders (TLxx) - Mode registers (TMODn)	 $00 $00 $00	 $00 $00 $00
Serial Interface - SIOM - Internal octal counter - Serial data register (SIOH, SIOL)	 0 0 Retained	 0 0 Undefined
LCD - Control register (LCON) - Mode register (LMOD0~1)	 0 0	 0 0
Interrupt - Interrupt enable (IE) - All Interrupt enable flags - All Interrupt request flags	 0 0 0	 0 0 0

Note: n=0,1,2,3

RAM Mapped Register Initial Value in Normal mode

Name	Address	Initial	R/W	Note
CKS.1,0	$00D.1,0	10B	W	01:Fmain /8, 10: Fmain/4
KPAD	$01F.1	0B	W	0: disable 1: enable
KRVS	$01F.0	0B	W	0: non-reverse 1: reverse
BAT.0	$03D.0	0B	R	0: V_{DD} >*2.3V 1: V_{DD}≦*2.3V
BAT.1	$03D.1	0B	R	0: V_{DD} >*3.2V 1: V_{DD}≦*3.2V
BAT.2	$03D.2	0B	R	0: V_{DD} >*4.3V 1: V_{DD}≦*4.3V
CVAR.3~0	$03E.3~0	0000B	W	Refer to the section of LCD Contrast Control
ENBAT	$01F.2~0	000B	W	0: disable 1: enable
SGM	$03A.3~0	0000B	R/W	Refer to the section of Sound Generator
SGD	$03B.3~0	0000B	R/W	Refer to the section of Sound Generator
W1,W0	$016.3~2	00B	R/W	-
IRM.1~0	$016.1~0	00B	R/W	Refer to section of IR Mode Register

* : For reference only

Addressing Mode

There are 8 addressing modes: 3 for RAM addressing, 5 for ROM addressing.

RAM addressing:

1: Register indirect addressing

Bit 11~0	11	10	9	8	7	6	5	4	3	2	1	0
RAM address	V3	V2	V1	V0	X3	X2	X1	X0	Y3	Y2	Y1	Y0

Note: V3 must be set zero.

2: Direct addressing

Bit 11~0	11	10	9	8	7	6	5	4	3	2	1	0
RAM address	0	0	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

3: Memory Register Addressing

Bit 11~0	11	10	9	8	7	6	5	4	3	2	1	0
RAM address	0	0	0	0	0	1	0	0	M3	M2	M1	M0

ROM addressing

1: Direct addressing

Bit 13~0	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ROM address	P3	P2	P1	P0	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0

P3~P0 The LSB 4-bit of the 1st instruction opcode.
A9~A0 The 10-bit of the 2nd instruction opcode.
Example: [LBR $addr]

2: Short range addressing:

Bit 13~0	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ROM address	0	0	0	0	0	0	0	0	A5	A4	A3	A2	A1	A0

0 Zero supplied by CPU.
A5~A0 The LSB 6-bit of instruction opcode.
Example: [CAL $addr]

3: Current Page addressing:

Bit 13~0	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ROM address	PC13	PC12	PC11	PC10	PC9	PC8	A7	A6	A5	A4	A3	A2	A1	A0

PC13~PC8 The MSB 6-bit of current PC.
A7~A0 The LSB 8-bit of instruction opcode.
Example: [BR $addr]

4: Table jump address data addressing

Bit 13~0	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ROM address	0	0	P3	P2	P1	P0	B3	B2	B1	B0	A3	A2	A1	A0

P3~P0 The LSB 4-bit of instruction opcode.
B3~B0 The value of REG B.
A3~A0 The value of REG A.
Example: [TJMP page]
Note: Using this instruction, W1, W0 must be set zero.

.

5: Table data addressing

Bit 13~0	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ROM address	0(W1)	0(W0)	P3	P2	P1	P0	B3	B2	B1	B0	A3	A2	A1	A0

W1,W0 The bit3, bit2 of the RAM mapped register W/IR.
P3~P0 The LSB 4-bit of instruction opcode.
B3~B0 The value of REG B.
A3~A0 The value of REG A.
ROM data=RO9, RO8, RO7, RO6, RO5, RO4, RO3, RO2, RO1, RO0.
If RO8=1,Reg. (B, A)←RO7~RO0.
If RO9=1,PORT3, 2←RO7~RO0.
Example:[T (table page address)]
Note: Using this instruction, PC13 and PC12 are zero if the program address jumps into range $0000H~$00FFH or $0F00H~1FFFH. Else if the address jumps into range $0100H~0EFFH, PC13 and PC12 are supplied with W1 and W0.

Instruction Set

1. Transfer Instructions

1) Immediate Instruction (4)
2) Register to Register Instruction (8)
3) RAM address Instruction (13)
4) RAM Register Instruction (27)

2. Bit Manipulation instruction (6)

3. Compare Instruction (12)

4. Arithmetic and Logical Instructions (25)

5. Control Instruction

1) Branch Instruction (4)
2) Subroutine Stack control Instruction (4)
3) CPU control Instruction (4)
4) Table data generator Instruction (1)

Descriptions

1. Transfer Instructions

1) Immediate Instructions (4)

Operation	Mnemonic	Code	SF	W/C	Function
Store immediate to A	ITA i	13i		1/1	A←#i
Store immediate to B	ITB i	10i		1/1	B←#i
Store immediate to Memory, increment Y	ITMIY i	19i	NZ	1/1	M←#i, Y←Y+1
Store immediate to direct Memory	ITMD i, $addr	2Ai, d		2/2	M←#i, direct addressing

Notes: W/C = Word(s)/Cycle(s)
i = immediate 4-bit data
d =10-bit RAM direct addressing
NZ, NB, OVF conditions are reflected in the Status Flag (SF)

2) Register to register Instruction (8)

Operation	Mnemonic	Code	SF	W/C	Function
Store B to A	BTA	044		1/1	A←B
Store V to A	VTA	200,000		2/2	A←V
Store Y to A	YTA	0AF		1/1	A←Y
Store EX to A	EXTA	064		1/1	A←EX
Store EY to A	EYTA	054		1/1	A←EY
Store R(m) to A	RTA m	17m		1/1	A←R(m)
Store A to B	ATB	0C4		1/1	B←A
Exchange A and R(m)	XAR m	1Fm		1/1	A↔R(m)

Note: m = memory register index (0~15)

3) RAM Address Instructions (13)

Operation	Mnemonic	Code	SF	W/C	Function
Store immediate to V	ITV i(4)	0Fi		1/1	V←#i(4)
Store immediate to X	ITX i(4)	12i		1/1	X←#i(4)
Store immediate to Y	ITY i(4)	11i		1/1	Y←#i(4)
Store A to X	ATX	0E4		1/1	X←A
Store A to Y	ATY	0D4		1/1	Y←A
Store A to V	ATV	210,000		2/2	V←A
Increment Y	IY	05C	NZ	1/1	Y←Y+1
Decrement Y	DY	0DF	NB	1/1	Y←Y−1
Y add A	YAA	058	OVF	1/1	Y←Y+A
Subtract A from Y	YSA	0D8	NB	1/1	Y←Y−A
Exchange X and EX	XEX	001		1/1	X↔EX
Exchange Y and EY	XEY	002		1/1	Y↔EY
Exchange (X,Y) and (EX, EY)	XEXY	003		1/1	X↔EX, Y↔EY

4) RAM Register Instruction (27)

Operation	Mnemonic	Code	SF	W/C	Function
Store Memory to A	MTA	090		1/1	A←M
Store Memory to A, Exchange X and EX	MTAX	091		1/1	A←M, X↔EX
Store Memory to A, Exchange Y and EY	MTAY	092		1/1	A←M, Y↔EY
Store Memory to A, Exchange X and EX, Exchange Y and EY	MTAXY	093		1/1	A←M, X↔EX, Y↔EY
Store direct Memory to A	MTAD $addr	290,d		2/2	A←M, direct addressing
Store Memory to B	MTB	040		1/1	B←M
Store Memory to B, Exchange X and EX	MTBX	041		1/1	B←M, X↔EX
Store Memory to B, Exchange Y and EY	MTBY	042		1/1	B←M, Y↔EY
Store Memory to B, Exchange X and EX, Exchange Y and EY	MTBXY	043		1/1	B←M, X↔EX, Y↔EY
Store A to Memory	ATM	098		1/1	M←A
Store A to Memory, Exchange X and EX	ATMX	099		1/1	M←A, X↔EX

4) RAM Register Instruction (27) (continued)

Operation	Mnemonic	Code	SF	W/C	Function
Store A to Memory, Exchange Y and EY	ATMY	09A		1/1	M←A, Y↔EY
Store A to Memory, Exchange X and EX, Exchange Y and EY	ATMXY	09B		1/1	M←A, X↔EX, Y↔EY
Store A to direct Memory	ATMD $addr	298, d		2/2	M←A, direct addressing
Store A to Memory, increment Y	ATMIY	050	NZ	1/1	M←A, Y←Y+1
Store A to Memory, increment Y, Exchange X and EX	ATMIYX	051	NZ	1/1	M←A, Y←Y+1, X↔EX
Store A to Memory, Decrement Y	ATMDY	0D0	NB	1/1	M←A, Y←Y−1
Store A to Memory, Decrement Y, Exchange X and EX	ATMDYX	0D1	NB	1/1	M←A, Y←Y−1, X↔EX
Exchange A and Memory	XAM	080		1/1	A↔M
Exchange A and Memory, Exchange X and EX	XAMX	081		1/1	A↔M, X↔EX
Exchange A and Memory, Exchange Y and EY	XAMY	082		1/1	A↔M, Y↔EY
Exchange A and Memory, Exchange X and EX, Exchange Y and EY	XAMXY	083		1/1	A↔M, X↔EX, Y↔EY
Exchange A and direct Memory	XAMD $addr	280, d		2/2	A↔M, direct addressing
Exchange B and Memory	XBM	0C0		1/1	B↔M
Exchange B and Memory, Exchange X and EX	XBMX	0C1		1/1	B↔M, X↔EX
Exchange B and Memory, Exchange Y and EY	XBMY	0C2		1/1	B↔M, Y↔EY
Exchange B and Memory, Exchange X and EX, Exchange Y and EY	XBMXY	0C3		1/1	B↔M, X↔EX, Y↔EY

2. Bit Manipulation Instruction (6)

Operation	Mnemonic	Code	SF	W/C	Function
Set Memory Bit	SM n	088+n		1/1	M.n←1
Set direct Memory Bit	SMD n, $addr	288+n, d		2/2	M.n←1, direct addressing
Reset Memory Bit	RM n	084+n		1/1	M.n←0
Reset direct Memory Bit	RMD n, $addr	284+n, d		2/2	M.n←0 direct addressing
Test Memory Bit	TM n	08C+n	M.n	1/1	SF←Bit value
Test direct Memory Bit	TMD n, $addr	28C+n, d	M.n	2/2	SF←Bit value direct addressing

Note: "n" is a 2-bit number, that indicates the bit location of a specific memory/register.

3. Compare Instruction (12)

Operation	Mnemonic	Code	SF	W/C	Function
Immediate not Equals to Memory	INEM i	02i	NZ	1/1	i ≠ M
Immediate not Equals to direct Memory	INEMD i, $addr	22i, d	NZ	2/2	i ≠ M (direct)
A not Equals to Memory	ANEM	008	NZ	1/1	A ≠ M
A not Equals to direct Memory	ANEMD $addr	208, d	NZ	2/2	A ≠ M (direct)
B not Equals to Memory	BNEM	048	NZ	1/1	B ≠ M
Y not Equals to immediate	YNEI i	07i	NZ	1/1	Y ≠ i
Immediate Less than or Equals to Memory	ILEM i	03i	NB	1/1	i ≤ M
Immediate Less than or Equals to direct Memory	ILEMD i, $addr	23i, d	NB	2/2	i ≤ M (direct)
A Less than or Equals to Memory	ALEM	018	NB	1/1	A ≤ M
A Less than or Equals to direct Memory	ALEMD $addr	218, d	NB	2/2	A ≤ M (direct)
A Less then or Equals to immediate	ALEI i	1Bi	NB	1/1	A ≤ i
B Less than or Equals to Memory	BLEM	0C8	NB	1/1	B ≤ M

4. Arithmetic and Logical Instruction (25)

Operation	Mnemonic	Code	SF	W/C	Function
Add immediate to A	AAI i	18i	OVF	1/1	A←A+i
Increment B	IB	04C	NZ	1/1	B←B+1
Decrement B	DB	0CF	NB	1/1	B←B-1
Decimal adjust for addition	DAA	0A6		1/1	To be adjusted if CY=1
Decimal adjust for subtraction	DAS	0AA		1/1	To be adjusted if CY=1
Negate A	NEGA	060		1/1	A← $\overline{A}$ +1
Complement B	NOTB	240		1/1	B← $\overline{B}$
Rotate right A with CY	RORC	0A0		1/1	
Rotate left A with CY	ROLC	0A1		1/1	
Set CY	SC	0EF		1/1	CY←1
Reset CY	RC	0EC		1/1	CY←0
Test CY	TC	06F	CY	1/1	
Add Memory to A	AAM	004	OVF	1/1	A←A+M
Add direct Memory to A	AAMD $addr	204, d	OVF	2/2	A←A+M (direct)
Add Memory to A with CY	AAMC	014	OVF	1/1	A←A+M+CY, CY←OVF
Add direct Memory to A with CY	AAMCD $addr	214, d	OVF	2/2	A←A+M+CY, CY←OVF direct addressing
Subtract A from Memory with CY	MSAC	094	NB	1/1	A←M-A- $\overline{CY}$, CY←NB
Subtract A from direct Memory with CY	MSACD $addr	294, d	NB	2/2	A←M-A- $\overline{CY}$ CY←OVF direct addressing
OR A and B	ORB	248		1/1	A←A OR B
AND A with Memory	ANDM	09C	NZ	1/1	A←A AND M
AND A with direct Memory	ANDMD $addr	29C, d	NZ	2/2	A←A AND M (direct)
OR A with Memory	ORM	00C	NZ	1/1	A←A OR M
OR A with direct Memory	ORMD $addr	20C, d	NZ	2/2	A←A OR M (direct)
XOR A with Memory	XORM	01C	NZ	1/1	A←A XOR M
XOR A with direct Memory	XORMD $addr	21C, d	NZ	2/2	A←A XOR M (direct)

5. Control Instructions

1) Branch Instructions (4)

Operation	Mnemonic	Code	SF	W/C	Function

Branch on status flag =1	BR $addr(8)	300+x	1	1/1	If SF=0 then
					PC←PC+1
					else
					PC←(PC&$3F00+x)

Long branch on status flag=1	LBR $addr(14)	270+p, d	1	2/2	branch if SF=1
Long jump unconditional	JMP $addr(14)	250+p, d		2/2	
Table jump	TJMP i	0B0+t		1/1	

Note: x = 8-bit address
p = 4-bit address
t = 4-bit page address
d = 10-bit address

2) Subroutine stack control Instruction (4)

Operation	Mnemonic	Code	SF	W/C	Function
Zero-page subroutine call on status (Please refer to ROM MAP)	CAL $addr(6)	2C0+a	1	1/2	Call if SF=1
Subroutine call on status	CALL $addr(14)	260+p, d	1	2/2	Call if SF=1
Return from subroutine	RET	010		1/3	
Return form interrupt	RETI	011	SF	1/3	IE←1, restore CY

Note: a = 6-bit address
p = 4-bit address
d = 10-bit page address

3) CPU control Instruction (4)

Operation	Mnemonics	Code	SF	W/C	Function
NO operation	NOP	000		1/1	CPU no operation
HALT mode	HALT	24C		1/1	Enter Halt Mode
STOP mode	STOP	24D		1/1	Enter Stop Mode
Start serial transmission	STS	244		1/1	Enter Serial Transmission

4) Table data generation instruction (1)

Operation	Mnemonics	Code	SF	W/C	Function
Table pattern generation	Tp	2Bp		1/1	

Note: p = 4-bit address

For example:
The opcode of " T5" instruction will generate the following address

Bit 13~0	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
ROM address	0(W1)	0(W0)	0	1	0	1	B3	B2	B1	B0	A3	A2	A1	A0

And the referred ROM data =RO9, RO8, RO7, RO6, RO5, RO4, RO3, RO2, RO1, RO0.
If RO8 is set to 1,the content of Register B and A is RO7~RO0 (B=RO7~RO4, A=RO3~RO0).
If RO9 is set to 1,the content of PORT 3,2 is RO7~RO0.
W1, W0 refer to the section PROGRAM APPLICATION NOTICE.

PROGRAM APPLICATION NOTICE

1 W1,W0 Register:

W1, W0 are Bit3, Bit2 of $16 register to expand the TP address up to 8K ROM.

If the program is running in the address ranges $0100H ~ $0EFFH, W1 and W0 must be set zero. And the default values of W1 and W0 are zero.

2 TP Instruction:

When the programmer uses the instruction TP, zero will be given to PC13 and PC12 by CPU at first. So the PC address is in the range $0000H~$0FFFH. And then, If the address is in range $0000H~$00FFH or $0F00H~$0FFFH, PC13 and PC12 are still zero supplied by CPU. Else if the address is in ranges $0100H~$0EFFH, PC13 and PC12 are supplied with W1 and W0.

The TP instruction addressing range is shown in the following table:

PC address of TP instruction	W1,W0	Table data address Bit13~0
$0000H ~ $00FFH	00~01H	W1~0,P3~0,B3~0,A3~0
$0100H ~ $0EFFH	Must be zero	0,0,P3~0,B3~0,A3~0
$0F00H ~ $1FFFH	00~01H	W1~0,P3~0,B3~0,A3~0

Notes: W1~0: RAM mapped register $016H bit3,bit2

P3~0: The LSB 4 bit of instruction opcode; and **P3~0 = 1~E is available**; **P3~0 = 0 or F is inhibited**;

B3~0: Internal register B bit3~0;

A3~0: Internal register A bit3~0.

For example:

The TP instruction is running at address $1400H, and the table data is in ROM address $1345H

```
ORG   1400H
ITA   #05H          ; A=05H
ITB   #04H          ; B=04H
ITMD  04H, $016H    ; W1 W0 =01
T     03
```

Note: At first, the ROM address of the table data is $0345H, and because the address is in range $0100H~$0EFFH, the final ROM address of the table data will be $1345H since W1 W0 =01.

3 TJMP Instruction

Before using the instruction TJMP, make sure that the value of W1 and W0 are zero.

TJMP instruction will make the program only jump to the range $0000~$0FFF.

The TJMP instruction addressing range is shown in the following table:

PC address of TJMP instruction	W1,W0	Object PC address
$0000H ~ $1FFFH	Must be zero	0,0,T3~0,B3~0,A3~0

Notes: W1~0: RAM mapped register $016H bit3,bit2;

T3~0: The LSB 4 bit of instruction opcode;

B3~0: Internal register B bit3~0;

A3~0: Internal register A bit3~0.

When the programmer uses the instruction TJMP, at first, zero will be given to PC13 and PC12 by CPU. So the program address is in range $0000H~$0FFFH. And then, If the address is in range $0000H~$00FFH or $0F00H~$0FFFH, PC13 and PC12 are still zero supplied by CPU. Else if the address is in range $0100H~$0EFFH, PC13 and PC12 are supplied with W1 and W0. And as the program goes on, errors may occur.

For example:

The TJMP instruction is running at address $1400H, and Object PC address is $0D45H

```
ORG   1400H
ITA   #05H          ; A=05H
ITB   #04H          ; B=04H
ITMD  00H, $016H    ; W1 W0 =00
TJMP  0D            ; after this instruction
                    ; PC will be $0D45H
```

4 Interrupt

The interrupt service program must be put in range $0000H~$00FFH or $0F00H~$1FFFH.

After setting W1 and W0 in range $0000H~$00FFH or $0F00H~$1FFFH, if the interrupt occurs and the Interrupt entrance is in range $0100H~$0EFFH, program will be wrong because of the effect of W1 and W0.

Electrical Characteristics

Absolute Maximum Ratings*

Power Supply Voltage . . . . . -0.5V to 6.0V
Input Voltage . . . . . -0.5V to V_{DD} +0.5V
Output Voltage . . . . . -0.5V to V_{DD} +0.5V
Power Dissipation 0.05W
Operating Temperature . . . . . 0℃ to 70℃
Storage Temperature . . . . . -55℃ to 135℃

***Comments:**

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

DC Electrical Characteristics (Temperature=0℃ to 70℃, V_{DD} =3.0V± 10%, GND=0V)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions
Operating Current	I_{DD}		200	300	μA	V_{DD} =3V -Normal mode -LCD on without loading
Standby Current	I_{SB}			3	μA	V_{DD} =3V -Stop mode -LCD off
Input High Voltage	V_{IH1}	0.9 V_{DD} V_{DD} -0.3		V_{DD} +0.3 V_{DD}+0.3	V	RESET XIN,XTIN
	V_{IH2}	0.8V_{DD}		V_{DD}+0.3	V	All pins except RESET,XIN and XTIN
Input Low Voltage	V_{IL1}	-0.3 -0.3		0.1V_{DD} 0.3	V	RESET XIN,XTIN
	V_{IL2}	-0.3		0.2 V_{DD}	V	All pins except RESET,XIN and XTIN
Output High Voltage	V_{OH}	V_{DD} -1.0			V	All output pins; I_{OH}=-0.5mA
Output Low Voltage	V_{OL}			0.6	V	All output pins; I_{OL}=1.6mA
Output current of general ports: Port0.3~0, Port1.3~0, Port2.3~0, Port3.3~0, Port4.3~0	I_{OH1} I_{OL1}	-1.2 3.0	-1.8 4.5	-	mA	V_{DD} =3.0V, V_{OH}=2.5V V_{DD} =3.0V, V_{OL}=0.5V
Output current of shared ports: Port5.3~0, Port6.3~0, Port7.3~0, Port8.3~0	I_{OH2} I_{OL2}	-0.9 1.8	-1.2 2.5	-	mA	V_{DD} =3.0V, V_{OH}=2.5V V_{DD} =3.0V, V_{OL}=0.5V
Pull-up Resistance	R_{UP}	150	200	250	KΩ	Port pins with Pull-up pins ; $V_{I/O}$ = GND

Note: All port pins should be given a voltage level by means of internal pull-high transistor or an external logic voltage level (GND / V_{DD})

DC Electrical Characteristics (Temperature=0℃ to 70℃, V_{DD} =5.0V± 10%, GND=0V)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions
Operating Current	I_{DD}		600	900	μA	V_{DD} =5V -Normal mode -LCD on without loading
Standby Current	I_{SB}			3	μA	V_{DD} =5V -Stop mode -LCD off
Input High Voltage	V_{IH1}	0.9 V_{DD} V_{DD} -0.3		V_{DD} +0.3 V_{DD}+0.3	V	RESET XIN,XTIN
	V_{IH2}	0.8V_{DD}		V_{DD}+0.3	V	All pins except RESET,XIN and XTIN
Input Low Voltage	V_{IL1}	-0.3 -0.3		0.1V_{DD} 0.3	V	RESET XIN,XTIN
	V_{IL2}	-0.3		0.2 V_{DD}	V	All pins except RESET,XIN and XTIN
Output High Voltage	V_{OH}	V_{DD} -1.0			V	All output pins; I_{OH}=-1mA
Output Low Voltage	V_{OL}			0.6	V	All output pins; I_{OL}=2mA
Output current of general ports: Port0.3~0, Port1.3~0, Port2.3~0, Port3.3~0, Port4.3~0	I_{OH1} I_{OL1}	-2.4 3.0	-3.0 4.5	-	mA	V_{DD} =5.0V, V_{OH}=4.5V V_{DD} =5.0V, V_{OL}=0.5V
Output current of shared ports: Port5.3~0, Port6.3~0, Port7.3~0, Port8.3~0	I_{OH2} I_{OL2}	-1.2 2.0	-1.8 3.0	-	mA	V_{DD} =5.0V, V_{OH}=4.5V V_{DD} =5.0V, V_{OL}=0.5V
Pull-up Resistance	R_{UP}	80	110	140	KΩ	Port pins with Pull-up pins ; $V_{I/O}$ = GND

Note: All port pins should be given a voltage level by means of internal pull-high transistor or an external logic voltage level (GND / V_{DD})

AC Electrical Characteristics (Temperature=0℃ to 70℃, V_{DD} =3.0V± 10%, GND=0V)

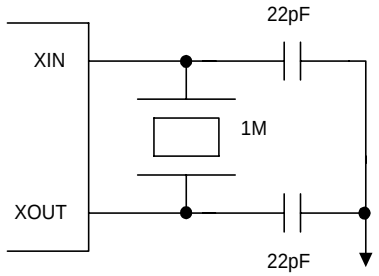
Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Main oscillator frequency	f_{MAIN}	2/2-10%	2/2	2/2+10%	MHz	Built-in RC oscillator
			1		MHz	Ceramic oscillator
Watch oscillator frequency	f_{WATCH}		32.768		KHz	
Watch oscillator start-up time	t_{WS}			1	Sec	
Reset Input High Duration	t_{RES}	2			t_{CYC}	
Instruction Cycle	t_{CYC}	4.0		8.0	μs	

AC Electrical Characteristics (Temperature=0℃ to 70℃, V_{DD} =5.0V± 10%, GND=0V)

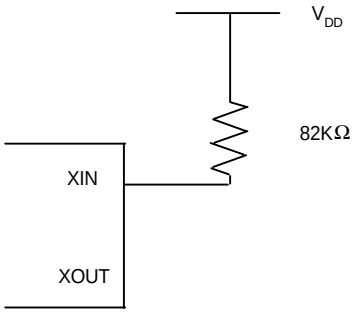
Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Main oscillator frequency	f_{MAIN}	2/2-10%	2/2	2/2+10%	MHz	Built-in RC oscillator
			1		MHz	Ceramic oscillator
Watch oscillator frequency	f_{WATCH}		32.768		KHz	
Watch oscillator start-up time	t_{WS}			1	Sec	
Reset Input High Duration	t_{RES}	2			t_{CYC}	
Instruction Cycle	t_{CYC}	4.0		8.0	μs	

Application Circuit (For Reference Only)

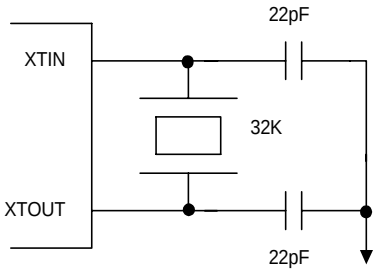
1) 1MHz Ceramic oscillator



2) 2MHz RC oscillator



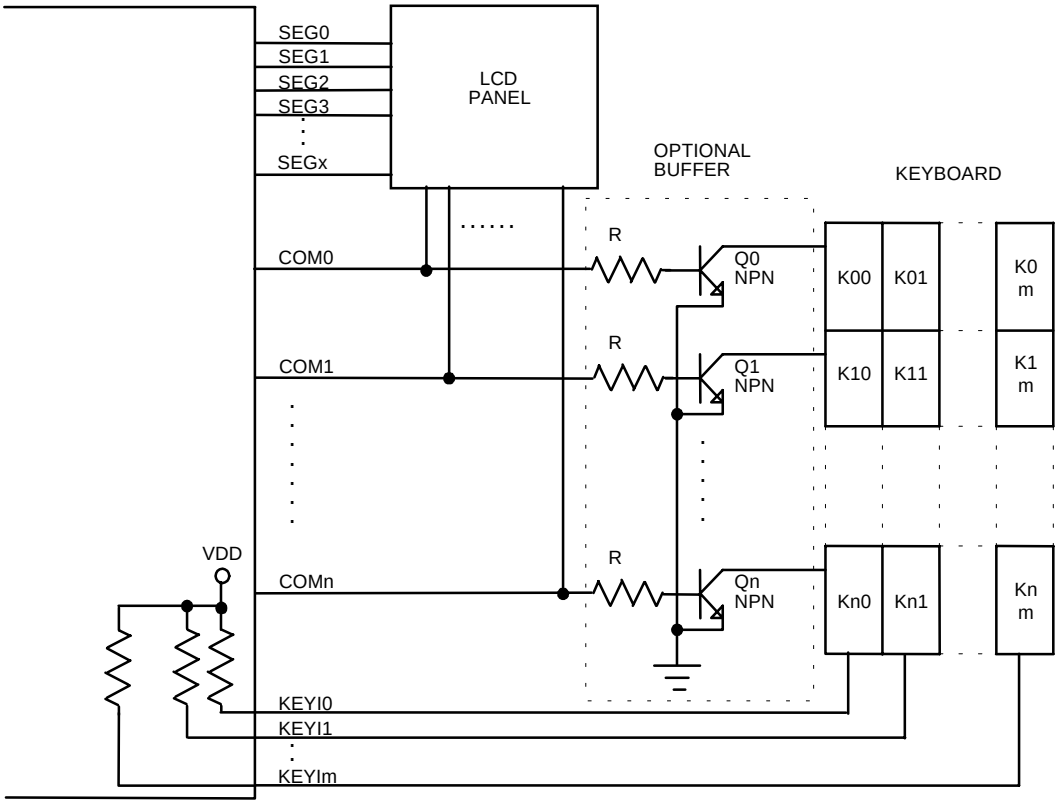
3) 32KHz Crystal oscillator



4) n × m Optional Keyboard (n=4 or n=8, m=4)

To expand the number of keypads, sharing the keyboard scan-line with LCD driver (COM0~n) may be the best way to do so, It means the LCD drivers (COM0~n) serve not only the LCD display but keyboard scanning. Furthermore, port 2 ($\overline{INT2}$) is chosen for detecting if any key was pressed. Port 2 only behaves as a normal I/O port unless the KPAD bit is set. After setting KPAD bit to 1, the built-in hardware circuit will automatically scan and sample the keyboard all the time, The sampling data would then be put into Keyboard registers ($028 to $02F). If there is any one of the 32 bits going low (normal high), the KPRS bit will be set right away.

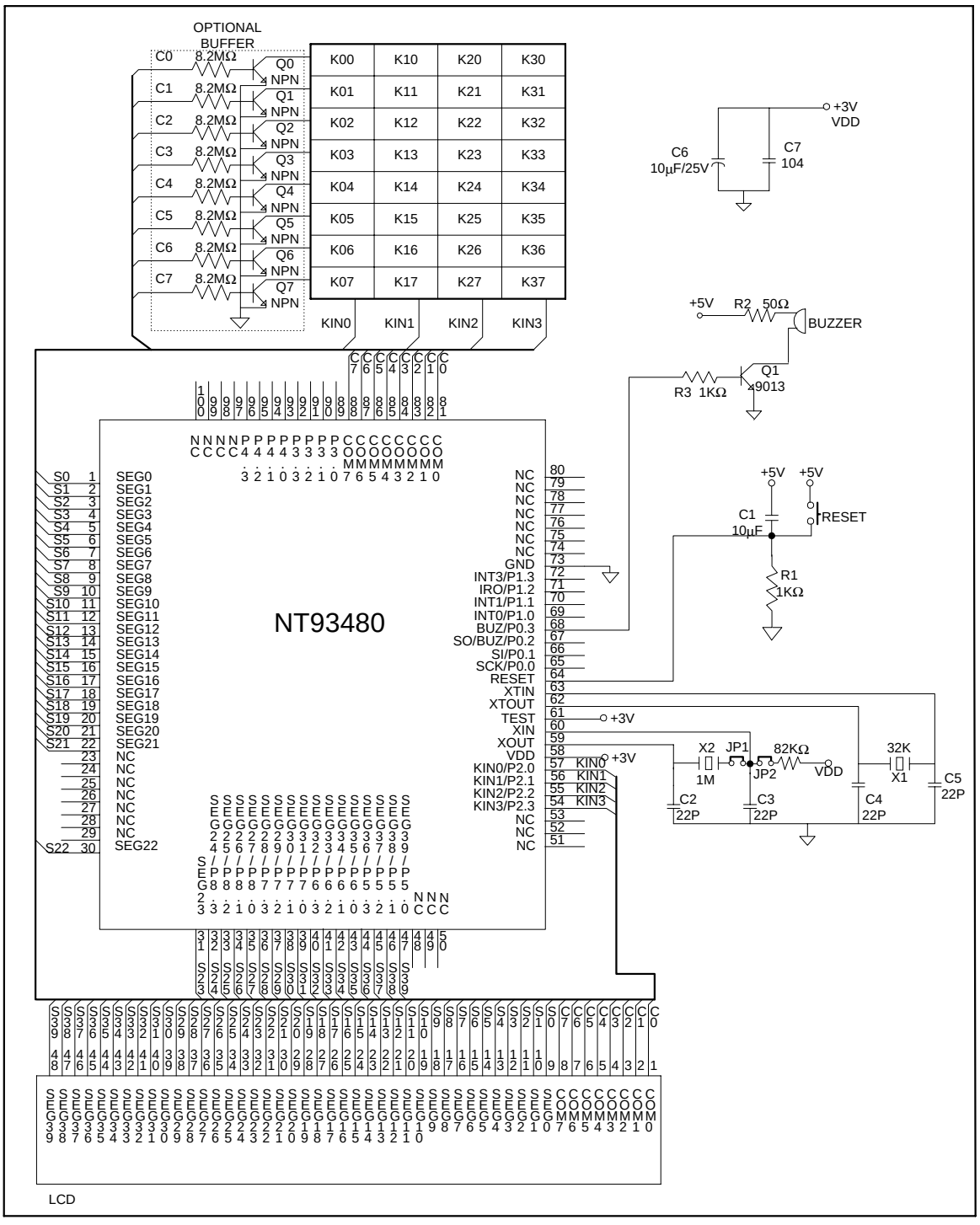
To prevent affecting the display quality from pressing key(s), a buffer may be added to isolate LCD driver signal. In this case the scanning signal must be reversed once before sending out to the keyboard. To do so just set KRVS bit. In addition, port2 must be set to input mode with data $F to activate the pull-high transistors, and LCD must be enabled whenever using this keyboard scanning function.



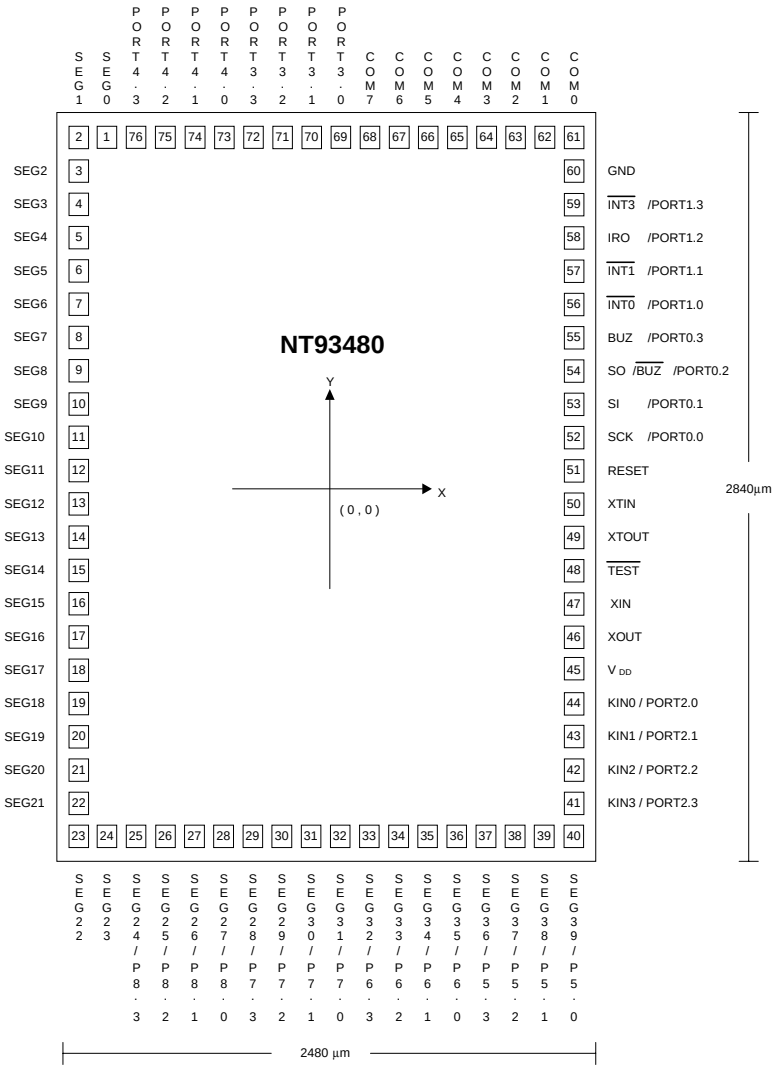
Note:

During the software programming, disposing the debouncing problem in keyboard application is necessary. Because there's no disposal for this problem in the hardware.

5) 640 Dots LCD, 32 Key Application Circuit



Bonding Diagram



* Substrate Connect to GND

* Pad window area: 90 µm x 90 µm

Bonding Dimensions

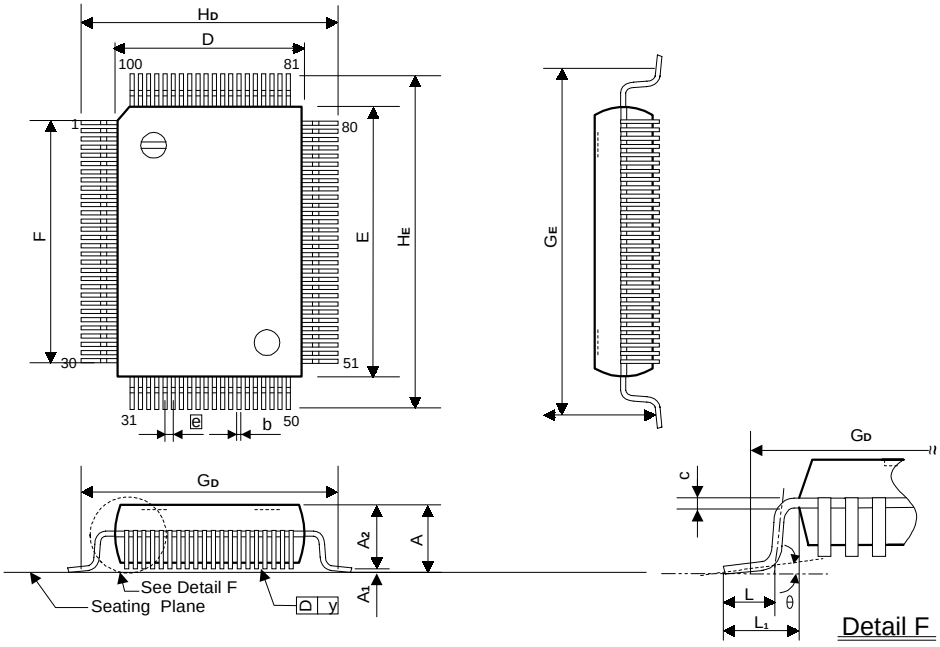
(unit: μm)

No.	Designation	X	Y	No.	Designation	X	Y
1	SEG0	-970	1280	39	PORT51	920	-1280
2	SEG1	-1102	1280	40	PORT50	1052	-1280
3	SEG2	-1102	1150	41	PORT23	1102	-1150
4	SEG3	-1102	1020	42	PORT22	1102	-1020
5	SEG4	-1102	900	43	PORT21	1102	-900
6	SEG5	-1102	780	44	PORT20	1102	-780
7	SEG6	-1102	660	45	V_{DD}	1102	-660
8	SEG7	-1102	540	46	XOUT	1102	-540
9	SEG8	-1102	420	47	XIN	1102	-420
10	SEG9	-1102	300	48	$\overline{TEST}$	1102	-300
11	SEG10	-1102	180	49	XTOUT	1102	-180
12	SEG11	-1102	60	50	XTIN	1102	-60
13	SEG12	-1102	-60	51	RESET	1102	60
14	SEG13	-1102	-180	52	PORT00	1102	180
15	SEG14	-1102	-300	53	PORT01	1102	300
16	SEG15	-1102	-420	54	PORT02	1102	420
17	SEG16	-1102	-540	55	PORT03	1102	540
18	SEG17	-1102	-660	56	PORT10	1102	660
19	SEG18	-1102	-780	57	PORT11	1102	780
20	SEG19	-1102	-900	58	PORT12	1102	900
21	SEG20	-1102	-1020	59	PORT13	1102	1020
22	SEG21	-1102	-1150	60	GND	1102	1150
23	SEG22	-1102	-1280	61	COM0	1052	1280
24	SEG23	-970	-1280	62	COM1	920	1280
25	PORT83	-844	-1280	63	COM2	794	1280
26	PORT82	-718	-1280	64	COM3	668	1280
27	PORT81	-592	-1280	65	COM4	542	1280
28	PORT80	-466	-1280	66	COM5	416	1280
29	PORT73	-340	-1280	67	COM6	290	1280
30	PORT72	-214	-1280	68	COM7	164	1280
31	PORT71	-88	-1280	69	PORT30	38	1280
32	PORT70	38	-1280	70	PORT31	-88	1280
33	PORT63	164	-1280	71	PORT32	-214	1280
34	PORT62	290	-1280	72	PORT33	-340	1280
35	PORT61	416	-1280	73	PORT40	-466	1280
36	PORT60	542	-1280	74	PORT41	-592	1280
37	PORT53	668	-1280	75	PORT42	-718	1280
38	PORT52	794	-1280	76	PORT43	-844	1280

Ordering Information

Part No.	Package
NT93480H	CHIP FORM
NT93480F	100L QFP

Package Information QFP 100L Outline Dimensions unit: inches/mm



Symbol	Dimensions in inches	Dimensions in mm
A	0.130 Max.	3.30 Max.
A_1	0.004 Min.	0.10 Min.
A_2	0.112 ± 0.005	2.85 ± 0.13
b	0.012 $^{+0.004}_{-0.002}$	0.31 $^{+0.10}_{-0.05}$
c	0.006 $^{+0.004}_{-0.002}$	0.15 $^{+0.10}_{-0.05}$
D	0.551 ± 0.005	14.00 ± 0.13
E	0.787 ± 0.005	20.00 ± 0.13
e	0.026 ± 0.006	0.65 ± 0.15
F	0.742 NOM.	18.85 NOM.
G_D	0.693 NOM.	17.60 NOM.
G_E	0.929 NOM.	23.60 NOM.
H_D	0.740 ± 0.012	18.80 ± 0.31
H_E	0.976 ± 0.012	24.79 ± 0.31
L	0.047 ± 0.008	1.19 ± 0.20
L_1	0.095 ± 0.008	2.41 ± 0.20
y	0.006 Max.	0.15 Max.
θ	0° ~ 12°	0° ~ 12°

Notes:

1. Dimensions D&E do not include resin fins.
2. Dimensions G_D & G_E are for PC Board surface mount pad pitch design reference only.