TOSHIBA MOS TYPE DIGITAL INTEGRATED CIRCUIT

# TMP47C22F

SILICON MONOLITHIC SILICON GATE CMOS

PRELIMINARY

## CMOS 4-BIT SINGLE CHIP MICROCOMPUTER (TLCS-47C) TMP47C22F

### GENERAL DESCRIPTION

The TLCS-47 is the high speed and high performance, 4-bit single chip microcomputer series designed for general purpose use.

The TLCS-47 has variously powerful functions in order to meet with the advanced and complicated applications, which will be made in near future. In addition, software compatible NMOS family (TLCS-47N) and CMOS family (TLCS-47C) are also provided.

The TMP47C22F is a chip countaining LCD driver for the TLCS-47C. The memory capacity consists of ROM 2,048 x 8 bits and RAM 192 x 4 bits. The TMP4700C (NMOS) is an evaluator chip used for the system development.

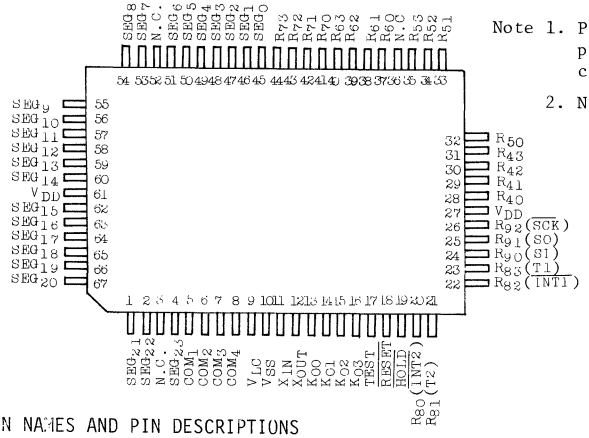
### FEATURES

- 4-bit single chip microcomputer with built-in
  ROM, RAM, input/output port, divider, timer/counter, and serial port.
- Instruction execution time: 4 μs (at 4 MHz clock)
- Effective instruction set
  90 instructions, software compatible in the series
- Subroutine nesting: Maximum 15 levels
- 6 interrupts (External: 2, Internal: 4)
  Independently latched control and multiple interrupt control
- Input/output port (27 pins)

| | | |
|---|---|---|
| Input | 1 port | 4 pins |
| I/O | 4 ports | 16 pins |
| I/O (Note) | 2 ports | 7 pins |

  Note: These I/O ports are also used for the interrupt input, timer/counter input, and serial port; therefore, it is programmably selectable for each application.
- Table look-up and table search function (Instruction)
  Table can be set up in the whole ROM area.
- 12-bit timer/counter (2 channels)
  Event counter, timer, and pulse width measurement mode is programmably selectable.
- Serial port with 4-bit buffer
  Receive/Transfer mode is programmably selectable.
  External/internal clock and leading/trailing edge mode are programmably selectable.
- 18-stage divider (with 4-stage prescaler)
  Frequency applied for timer interrupt of divider is programmably selectable.
- LCD drive circuit (automatic display) built-in
  - LCD direct drive is available (Max. 12-digit display at 1/4 duty LCD)
  - 1/4, 1/3, 1/2 duties or static drive are programmably selectable.
- Hold function
  Battery operation/condenser backup is avilable.
- On Chip oscillator
- TTL/CMOS compatible
- +5V single power supply
- 67-pin flat package
- Si-gate CMOS LSI

PIN CONNECTIONS (Top View)



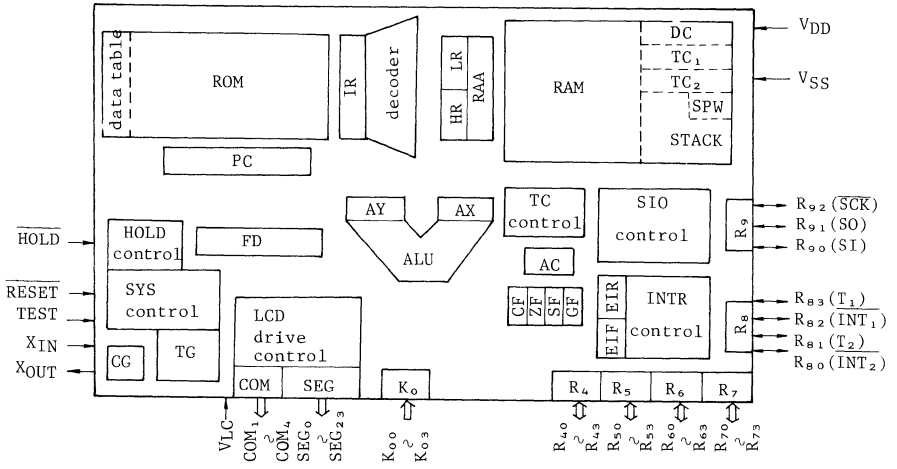

Note 1. Pin 27 is connected to pin 61 through external circuit.

2. N.C. No connection

PIN NAMES AND PIN DESCRIPTIONS

| Pin Names | No.of Pins | I/O | Functions |
|---|---|---|---|
| $K_{03} \sim K_{00}$ | 4 | Input | Input port |
| $R_{43} \sim R_{40}$ | 4 | I/O | I/O port |
| $R_{53} \sim R_{50}$ | 4 | I/O | " |
| $R_{63} \sim R_{60}$ | 4 | I/O | " |
| $R_{73} \sim R_{70}$ | 4 | I/O | " |
| $R_{83}$ (T1) | 1 | I/O | I/O port or timer/counter input |
| $R_{82}$ (INT1) | 1 | I/O | I/O port or interrupt input |
| $R_{81}$ (T2) | 1 | I/O | I/O port or timer/counter input |
| $R_{80}$ (INT2) | 1 | I/O | I/O port or interrupt input |
| $R_{92}$ ($\overline{SCK}$) | 1 | I/O | I/O port or shift clock for serial port |
| $R_{91}$ (SO) | 1 | I/O | " or serial output |
| $R_{90}$ (SI) | 1 | I/O | " or serial input |
| $SEG_{23} \sim SEG_0$ | 24 | Output | LCD Segment driver output |
| $COM_4 \sim COM_1$ | 4 | Output | LCD Common driver output |
| $X_{IN}$, $X_{OUT}$ | 2 | Input, Output | Resonator connection terminal |
| $\overline{RESET}$ | 1 | Input | Initialize signal input |
| $\overline{HOLD}$ | 1 | Input | Hold signal input |
| TEST | 1 | Input | (Low level is input.) |
| $V_{DD}$ | 1 | Power supply | +5V |
| $V_{SS}$ | 1 | Power supply | 0V |
| $V_{LC}$ | 1 | Power supply | LCD drive power supply |

BLOCK DIAGRAM




BLOCK NAME AND DESCRIPTION

| Block Names | Functions |
|---|---|
| PC | Program counter (12 bits) |
| ROM | Program memory |
| IR, decoder | Instruction register, Decoder |
| HR, LR | H register (Page assignment of RAM), L register (address assignment in RAM page), (each 4-bit register). |
| RAA | RAM address buffer register (8 bits) |
| RAM | Data memory |
| STACK | Save area of program counter and flags(RAM area) |
| SPW | Stack pointer word (RAM area) |
| DC, data table | Data counter (12 bits,RAM area), Data table (ROM area) |
| AX, AY | Tempoerary register of ALU input |
| ALU | Arithmetic and logic unit |
| AC | Accumulator |
| FLAG(CF,ZF,SF,GF) | Flags |
| K, R | Ports |
| INTR control | Interrupt control<br>(EIF: Enable interrupt master F/F, EIR: Enable interrupt register) |
| FD | Frequency divider (4-stage prescaler + 18 stages) |
| $TC_1$, $TC_2$ | 12-bit timer/counter 2 channels (RAM area) |
| TC control | Timer/counter control |
| SIO control | Serial port control |
| LCD drive control (COM,SEG) | LCD drive control |
| HOLD control | Control of hold function |
| SYS CONTROL | Generation of various internal control signals |
| CG, TG | Clock generator, timing generator |

## FUNCTIONAL DESCRIPTION

Concerning the TMP47C22F, the configuration and functions of hardwares are described.

As the description has been provided with priority on those parts differing from the TMP47C20P (The TLCS-47C standard chip), the technical material for the TMP47C20P shall also be referred to.

### 1. System Configuration

The configuration will be explained with priority given primarily to the LCD drive circuit.

### 1.1 Program Counter (PC), Program Memory (ROM)

The TMP47C22F is in 32 page configuration in a unit of 64 words per page with the built-in 12 bit program counter and 2,048 x 8 bit (000 ∿ 7FF addresses) program memory.

Further, as the TMP47C22F has no built-in output ports $P_1$ and $P_2$, the instruction (OUTB @HL) and PLA data conversion table cannot be used.

The relationship between ROM capacity and addresses is shown in Fig. 1.3.1.

### 1.2 H Register (HR), L Register (LR)

The H and L registers are 4-bit registers used as the data memory address pointer or general purpose registers, respectively.

1.3 RAM Address Buffer Register (RAA), Data Memory (RAM)

The TMP47C22F contains a data memory with 192 x 4-bit (addresses 00 ∿ BF) and is in 12 pages configuration in a unit of 16 words per page.

On the other hand, since RAM address buffer register (RAA) has 8-bit length, addresses C0 ∿ FF have no physical RAM, but the higher order 2 bits ($RAA_7$ and $RAA_6$) are decoded to [(00), (01) and (1*). * denotes "don't care."]; therefore, when addresses C0 ∿ FF are accessed on a program, RAM equivalent to addresses 80 ∿ BF is accessed. In other words, on a program a specific address of RAM is addressed to addresses C0 ∿ FF, while on the TMP 47C22F, RAM equivalent to addresses 80 ∿ BF is allocated.

The relationship between RAM capacity and addresses is shown in Fig. 1.3.1.

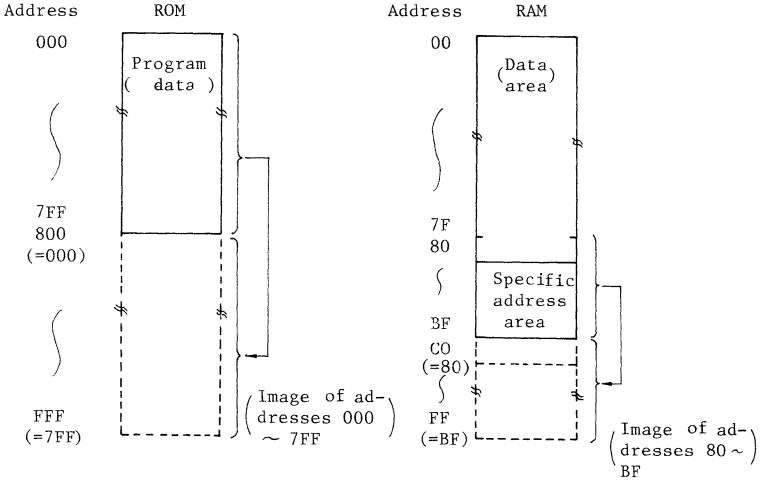



Fig. 1.3.1 ROM/RAM Capacity and Addresses

1.4 ALU, Accumulator (AC), Flag (FLAG)

The ALU is a circuit used for various operation for 4-bit binary data. It performs the operation designated by the instruction, and outputs the 4-bit result, carry (C), and zero detection signal (Z).

The accumulator is a 4-bit register to use a source operand for the arithmetic operation, and in which the result is stored.

Flag is a 4-bit register used to store the conditions of arithmetic operation, and of which the set/reset conditions are specified by the instruction. The flag consisting of CF, ZF, SF, and GF is saved in the stack when the interrupt is accepted. By executing the (RETI) instruction, it is restored from the stack to the condition immediately before the interrupt is accepted.

1.5 Port (PORT)

Data transfer to/from the external circuitry, and command/status/data transfer between of the built-in peripheral circuitry are carried out by the input/output instructions.

Since the TMP47C22F has no built-in outputs ports $P_1$ and $P_2$, (OUTB @HL) instruction cannot be used; therefore 12 kinds of instructions become available as effective input/output instructions.

The details to specify the input/output circuit format of ports and initialization of the output latches are 2.3 Input/Output Port (Input/Output Circuit Format).

| Port address | Symbol (Input/ Output) | Port, Register (Input/Output) | Input/Output Instructions | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | IN %P, A<br>IN %P,@HL | OUT A , %P<br>OUT@HL,%P | OUT#K,%P | OUTB @HL | SET %P,b<br>CLR %P,b | TEST %P,b<br>TESTP%P,b | SET @L<br>CLR @L<br>TEST @L |
| 00 | IP00/OP00 | $K_0$ Input port/ - | 0 | | | | | 0 | |
| 01 | IP01/OP01 | - | | | | | | | |
| 02 | IP02/OP02 | - | | | | | | | |
| 03 | IP03/OP03 | - | | | | | | | |
| 04 | IP04/OP04 | $R_4$ I/O port | 0 | 0 | 0 | | 0 | 0 | 0 |
| 05 | IP05/OP05 | $R_5$ " | 0 | 0 | 0 | | 0 | 0 | 0 |
| 06 | IP06/OP06 | $R_6$ " | 0 | 0 | 0 | | 0 | 0 | 0 |
| 07 | IP07/OP07 | $R_7$ " | 0 | 0 | 0 | | 0 | 0 | 0 |
| 08 | IP08/OP08 | $R_8$ " | 0 | 0 | 0 | | 0 | 0 | |
| 09 | IP09/OP09 | $R_9$ " | 0 | 0 | 0 | | 0 | 0 | |
| 0A | IP0A/OP0A | - | | | | | | | |
| 0B | IP0B/OP0B | - | | | | | | | |
| 0C | IP0C/OP0C | - | (*) Serial buffer register (Reception) | | | | | | |
| 0D | IP0D/OP0D | - | (**) Serial buffer register (Transmission) | | | | | | |
| 0E | IP0E/OP0E | Status input/ - | 0 | | | | | 0 | |
| 0F | IP0F/OP0F | (*) / (**) | 0 | 0 | 0 | | | | |
| 10 | /OP10 | /Hold control | | 0 | | | | | |
| 11 | /OP11 | / - | | | | | | | |
| 12 | /OP12 | / - | | | | | | | |
| 13 | /OP13 | / - | (a) Control with timer interrupt of divider | | | | | | |
| 14 | /OP14 | / - | (b) LCD drive control (1), (2) | | | | | | |
| 15 | /OP15 | / - | (c) Timer/counter 1 control | | | | | | |
| 16 | /OP16 | / - | (d) Timer counter 2 control | | | | | | |
| 17 | /OP17 | / - | (e) Serial port control | | | | | | |
| 18 | /OP18 | / - | | | | | | | |
| 19 | /OP19 | / (a) | | 0 | | | | | |
| 1A | /OP1A | / (b) (1) | | 0 | | | | | |
| 1B | /OP1B | / " (2) | | 0 | | | | | |
| 1C | /OP1C | / (c) | | 0 | | | | | |
| 1D | /OP1D | / (d) | | 0 | | | | | |
| 1E | /OP1E | / - | | | | | | | |
| 1F | /OP1F | / (e) | | 0 | | | | | |

Note 1: Inputs (IP10 ∿ IP1F) of port addresses 10 ∿ 1F remain undefined.
Note 2: Port addresses with "-" mark are reserved addresses and cannot be used user's program.

Table 1.5.1 Port Address Allocation and Input/Output Instructions

(1) $K_0$ ($K_{03} \sim K_{00}$) Port

This is a 4-bit port used for input.

(2) $R_4$ ($R_{43} \sim R_{40}$), $R_5$ ($R_{53} \sim R_{50}$), $R_6$ ($R_{63} \sim R_{60}$), $R_7$ ($R_{73} \sim R_{70}$) Port

Each of these ports is a 4-bit I/O port with a latch. The latch should be set to "1" when the port is used as an input port.

Pins $R_{73}$ - $R_{40}$ can be used for bit scanning for set/reset and test according to the contents of the L register by executing the (SET @L), (CLR @L) and (TEST @L) instructions.

(3) $R_8$ ($R_{83} \sim R_{80}$) Port

This is a 4-bit I/O port with a latch. The latch should be set to "1" when the port is used as an input port.

It is a port common to external interrupt input or external timer/counter input. When it is used as normal I/O port, some measures, such as inhibition of the external interrupt input acceptance or disable of the mode depending on the external input of the timer/counter should be taken in a program.

(4) $R_9$ ($R_{92} \sim R_{90}$) Port

This is a 3-bit I/O port with a latch, and the latch must be set to "1" when it is used as input port.

The $R_9$ port is also used as serial port. The port used as normal I/O port is not entirely influenced by disabling the serial port.

Pin $R_{93}$ is not mounted in the port, but "1" is read by accessing to pin $R_{93}$ in a program.

1.6 Interrupt Control Circuit (INTR)

Interrupt factors are composed of two from the external circuitry, and four from the internal circuitry. By setting the interrupt latch provided for each factor, an interrupt is generated to the CPU.
The interrupt latch is set when the edge of the input signal is detected.

1.7 Frequency Divider (FD)

The divider ($FD_1$ - $FD_{18}$) is made up 18-stage binary counter, and its output is used to generate various internal timing.

1.8 Timer/counter ($TC_1$, $TC_2$)

Two channels of 12-bit binary counter is contained to count time or event.

Count Operation

When the rising edge of the count pulse is detected, the count latch is set to send a count request to the CPU.

The maximum frequency applied to the timer/counter is as follows. In the timer mode, the maximum frequency is determined by a command.

The maximum frequency applied to the external input pin in the pulse width measurement mode should be the frequency level available for analyzing the count value in the program.
Normally, the frequency sufficiently slower than the disignated internal pulse rate is applied to the external input pin.

The maximum frequency applied to the external input pin under the event counter mode is dependent upon the operating state of the LCD drive circuit.

(a) At time of blanking operation

Frequency applied at time of a single channel operation is fc/64 Hz. When 2 channels are operated simultaneously, timer/counter 1 is fc/64 Hz and timer/counter 2 is fc/80 Hz.

(b) When LCD display is enabled

Frequency applied at time of a single channel operation is fc/128 Hz. When 2 channels are operated simultaneously, both timer/counter 1 and timer/counter 2 are fc/144 Hz.

## 1.9 Serial Port (SIO)

A 4-bit serial port with a buffer is provided to transfer the serial data from/to the external circuitry. According to the contents of the command register, either one of transmit mode, receive (trailing edge shift) mode or receive (leading edge shift) mode can be selected.

## 1.10 Hold Control Circuit (HOLDC)

The hold function is the function to hold the status immediately before the system operation is stopped at low power consumption making the most of the features of CMOS.

The hold function is controlled by $\overline{\text{HOLD}}$ terminal input and command register.

1.11 LCD Drive Circuit (LCDC)

The TMP47C22F has the built-in circuit that directly drives the liquid crystal display (LCD) and its control circuit.
The TMP47C22F has the following connecting terminals with LCD:

(a) Common output terminals ($COM_1$ - $COM_4$)

(b) Segment output terminals ($SEG_0$ - $SEG_{23}$)

In addition, $V_{LC}$ terminal is provided as the drive power terminal.

As display data transfer operations to the drive circuit are entirely executed by the hardware automatically on the TMP47C22F, it is possible to illuminate LCD if only display data is stored in the data memory.

The devices that can be directly driven is selectable from LCD devices of following drive methods:

(a) 1/4 duty (1/3 bias) LCD
Max. 96 segments (12 digits x 8 segments) can be driven.

(b) 1/3 duty (1/3 bias) LCD
Max. 72 ssgments (8 digits x 9 segments) can be driven.

(c) 1/2 duty (1/2 bias) LCD
Max. 48 segments (6 digits x 8 segments) can be driven.

(d) Static LCD
Max. 24 segments (3 digits x 8 segments) can be driven.

(1) Circuit configuration

The LCD drive circuit consists of the function blocks shown in Fig. 1.11.1.

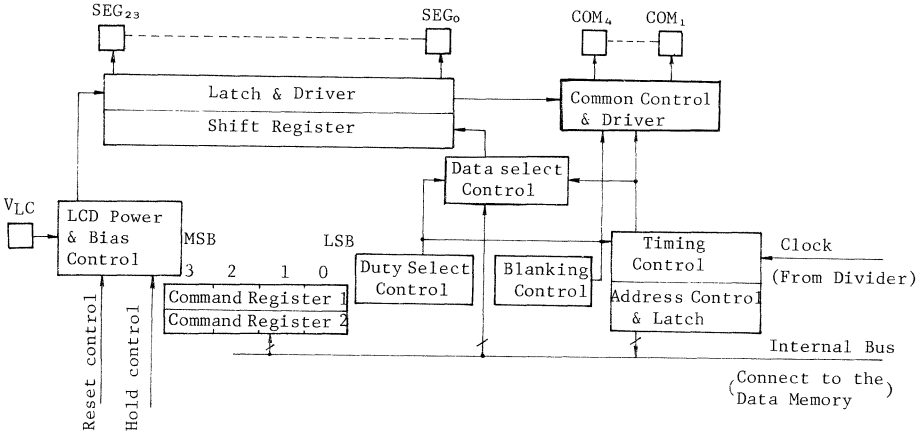



Fig. 1.11.1 LCD Drive Circuit

(2) Control of drive circuit

The operation of LCD drive circuit is controlled by the command. The command registers are accessed as port addresses OP1A and OP1B, and are reset to "8" and "0" at initialization, respectively.

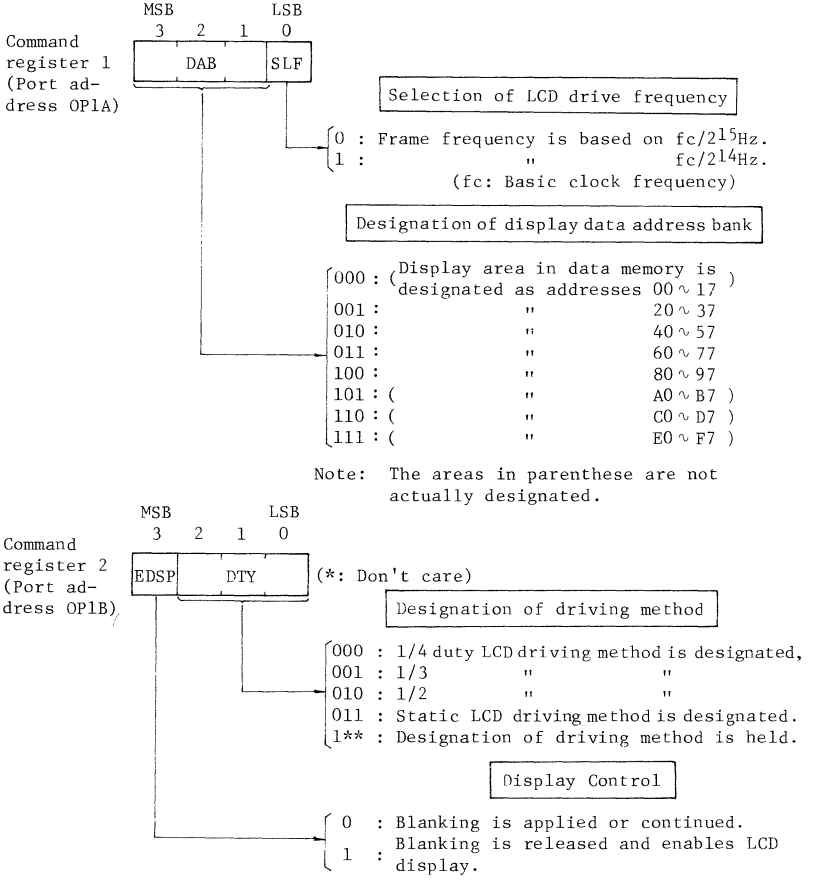


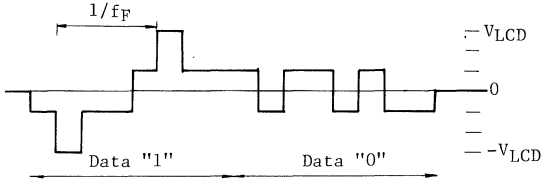

Fig. 1.11.2 Control of Drive Circuit

Drive waveform of LCD

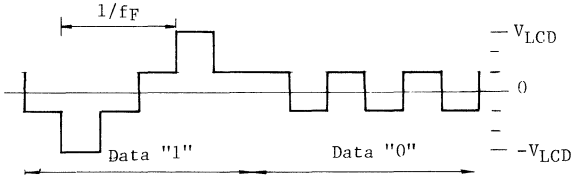
The LCD drive method is selected according to DTY of command register 2. DTY is reset to "0" at initialization.

The drive method is initialized according to a LCD used in the initial program. (In the case of a 1/4 duty LCD, it is set at initialization.) Thereafter, DTY sets disable code only.

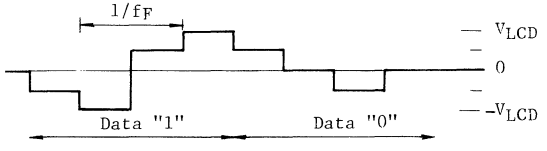
Examples of LCDs and their drive waveforms are shown in Fig. 1.11.3.



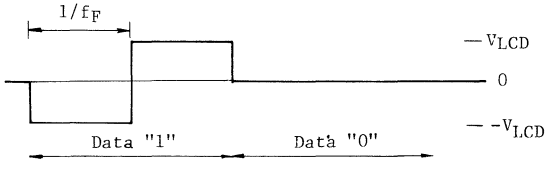

(a) 1/4 duty (1/3 bias) drive




(b) 1/3 duty (1/3 bias) drive




(c) 1/2 duty (1/2 bias) drive




(d) Static drive

(Note) $f_F$ : LCD Frame frequency, $V_{LCD}=V_{DD}-V_{LC}$

Fig. 1.11.3 LCD Drive Waveform (COM-SEG Terminals)

LCD Frame frequency

Frame frequency (LCD drive frequency) is given by the built-in frequency divider. It is possible to select base frequency (either one of 2 kind frequencies obtained from the divider) by SLF of command register 1. SLF is reset to "0" at the initialization.

Frame frequency ($f_F$) is set according to the drive method and base frequency as shown in the following table:

| SLF | Base frequency (Hz) | Frame Frequency (Hz) | | | |
|---|---|---|---|---|---|
| | | 1/4 Duty | 1/3 Duty | 1/2 Duty | Static |
| 0 | $\frac{f_c}{2^{15}}$ | $\frac{f_c}{2^{15}}$ | $\frac{4}{3} \cdot \frac{f_c}{2^{15}}$ | $\frac{4}{2} \cdot \frac{f_c}{2^{15}}$ | $\frac{f_c}{2^{15}}$ |
| | ($f_c$=4 MHz) | $\fallingdotseq$ 122 | $\fallingdotseq$ 163 | $\fallingdotseq$ 244 | $\fallingdotseq$ 122 |
| 1 | $\frac{f_c}{2^{14}}$ | $\frac{f_c}{2^{15}}$ | $\frac{4}{3} \cdot \frac{f_c}{2^{15}}$ | $\frac{4}{2} \cdot \frac{f_c}{2^{15}}$ | $\frac{f_c}{2^{15}}$ |
| | ($f_c$=2 MHz) | $\fallingdotseq$ 122 | $\fallingdotseq$ 163 | $\fallingdotseq$ 244 | $\fallingdotseq$ 122 |

($f_c$: Basic clock frequency)

Table 1.11.1 LCD Frame Frequency Setting

LCD drive voltage

The $V_{LC}$ terminal is the LCD drive power terminal. LCD drive voltage ($V_{LCD}$) is given by $V_{DD} - V_{LC}$. Therefore, if CPU operating voltage and LCD drive voltage are same, connect the $V_{LC}$ terminal to the $V_{SS}$ terminal.

Drive voltage applied to the LCD drive circuit is internally turned ON/OFF according to the operating state of CPU. That is, at the time of initialize operation and hold operation, the built-in power switch is automatically turned off to cut off drive voltage.

The LCD power switch turned off by the initialize operation is automatically turned on when EDSP (MSB of command register 2) is set at "1" and voltage is applied to the drive circuit. Thereafter, as the power switch is not turned off by the blanking control by means of a program, drive voltage is kept applied to the drive circuit.

On the other hand, the power switch is also turned off at the time of the hold operation, LCD display is turned off and the hold operation is executed at low power consumption. After the hold is released, the TMP47C22F is automatically returned to the state immediately before the hold operation was started.

Further, when the built-in power switch is OFF, $V_{DD}$ level voltage is generally at either COM terminals or SEG terminals.

(3) Display operation

Display data setting

Display data is stored in the display area (max. 24 words) in the data memory. The conversion process of ordinary data into LCD display data is executed by instructions (ROM data referring instruction is mainly used.).

Display data converted and stored in the display area is automatically transfered to the LCD drive circuit and displayed by the hardware without any participation by a program. Therefore, change of display pattern is possible by changing only data in the display area in the data memory by a program.

The LCD segment (dot) corresponds to each bit in the display area in the data memory on the one-for-one basis. This relation is shown in Fig. 1.11.4.

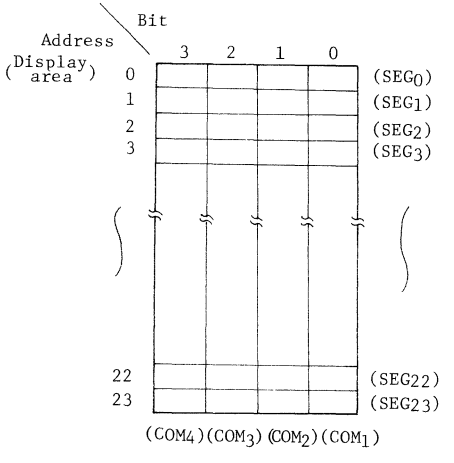



Fig. 1.11.4 LCD Diaply Data Area (Data Memory)

Where, each bit of the display data memory shows data of segment (dot) equivalent to SEGi, COMj ($0 \leq i \leq 23$, $1 \leq j \leq 4$), and when data is "1", the LCD illuminates.

Number of segments that can be driven varies depending upon the LCD drive method. This denotes that even in the display area of the data memory, number of bits used for storing display data varies.

(a) 1/4 duty LCD ($COM_3$ - $COM_1$ are used)
All bits in the display area becomes display data.

(b) 1/3 duty LCD ($COM_3$ - $COM_1$ are used)
Bit 2 - Bit 0 only become display data.

(c) 1/2 duty LCD ($COM_2$ - $COM_1$ are used)
Bit 1 and Bit 0 only become display data.

(d) Static LCD ($COM_1$ only is used)
Bit 0 only becomes display data.

Therefore, the data memory bits that are not used for storing display data or are equivalent to addresses to which no LCD is connected in the display area can be used for storing ordinary user's processing data.

As stated above, the data memory is used for storing display data (max. 24 words), and it is possible to set an address space in the data memory, to which this display area is to be set, by DAB of command register 1 (See Fig. 1.11.2.).

As the command register 1 is reset at "8" at initialization, the display area is initialized to 80 - 97 addresses.

Transfer of display data

Display data that has been set in the display area of the data memory is automatically transfered to the drive circuit. This operation is executed in the following sequence.

A display data transfer request is sent from the LCD drive circuit to CPU. Upon completion of an instruction under execution (if the timer/counter processing and the interrupt acceptance processing exist, after they are executed), CPU sends segment (dot) data in the display data area to the drive circuit in one instruction cycle.

This data sending cycle is taken place when drive voltage is kept applied to the LCD display drive circuit. Therefore, after intialize operation, this cycle is not taken place until EDSP is set to "1". Frequency of data sending cycle insertion is as follows:

(a) In case of other than static drive at SLF=0, 24 times in 512 instruction cycles.

(b) In case of static drive at SLF=0, 24 times in 2,048 instruction cycles.

(c) In case of other than static drive at SLF=1, 24 times in 256 instruction cycles.

(d) In case of static drive at SLF=1, 24 times in 1,024 instruction cycles.

Therefore, when LCD display is enable, the apparent speeds in above cases are decreased by 4.9, 1.2, 10.3 and 2.4%, respectively. For instance, in case of other than the static drive at SLF=0. The apparent speed is 4.2 μs to 4 μs instruction execution speed.

Blanking Operation

When EDSP (MSB of command register 2) is reset to "0", the LCD display becomes blank. EDSP is reset to "0" at initialization.

The blanking operation turns off the LCD by conditioning non-lighting operation level voltage to COM terminals. On the other hand, the SEG terminals are kept continued at normal operating state. (In the case of static drive, no voltage is applied to COM-SEG terminals when the LCD is turned off by data, however, as the blanking operation keeps the COM terminal at constant $V_{LCD}/2$ level, the LCD is turned off and the state between COM-SEG terminals where the LCD is driven by $V_{LCD}/2$. Therefore, note that the display state is somewhat different in these cases.) For drive waveforms, refer to Figl 1.11.6 - Fig. 1.11.9.

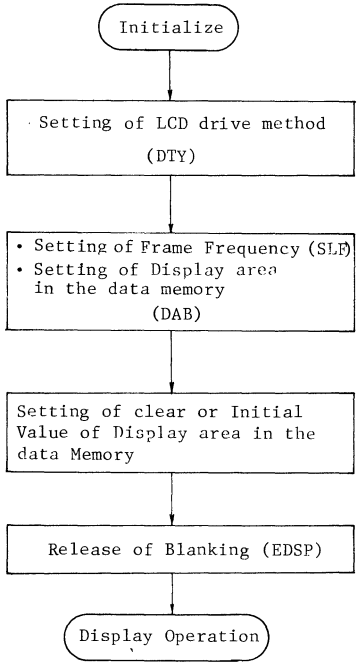
When EDSP is set at "1", the LCD display is enabled and the LCD display is made according to data stored in the display area of the data memory.

Further, when EDSP is initially set at "1" after the initialization, the LCD power switch is also turned ON and drive voltage is applied to the drive circuit.

LCD Display Control by Program

Provided that EDSP has been set at "1", the LCD is automatically turned ON according to data stored in the display area of the data memory. However, prior to actual display operation it is normally necessary to initialize as shown in Fig. 1.11.5.

To drive the 1/4 duty LCD, 80 - 97 addresses in the display area of the data memory are used, and to operate it at SLF = 0 (low speed operation), when EDSP is set to "1" after initialization of data in the display area, the display operation is started.




(* : Setting command)

: Set "00**" at port address OP1B. [Note 1] (Thereafter, use "1/0 1**" only.

: Set "****" at port address OP1A. [Note 2]

: Set "11**" at port address OP1B.

[Note 1] Classification of commands for port address OP1B.

| | | |
|---|---|---|
| "0000" ∿ "0011" | : | Setting of LCD drive method |
| "01**" | : | Blanking by program |
| "11**" | : | Releasing of blanking (display enable) |
| "10**" | : | Cannot be used |

[Note 2] Normally, only one time of setting is required at the time of initialization, but as an exception, commands should be set at port address OP1A under the blanking state whenever the display area are switched.

Fig. 1.11.5 Initialization of LCD Drive by Program

Examples of display data when a numeral display is made by using the 1/4 duty LCD are shown in Table 1.11.2. For the connecting method of COM terminals and SEG terminals, the example shown in Fig. 1.11.6 is used.

| Numeral | Display | Display data memory | | Numeral | Display | Display data memory | |
|---|---|---|---|---|---|---|---|
| | | High order address | Low order address | | | High order address | Low order address |
| 0 | 0. | 1 1 0 1 | 1 1 1 1 | 5 | 5 | 1 0 1 1 | 0 1 0 1 |
| 1 | 1 | 0 0 0 0 | 0 1 1 0 | 6 | 6 | 1 1 1 1 | 0 1 0 1 |
| 2 | 2 | 1 1 1 0 | 0 0 1 1 | 7 | 7 | 0 0 0 1 | 0 1 1 1 |
| 3 | 3 | 1 0 1 0 | 0 1 1 1 | 8 | 8 | 1 1 1 1 | 0 1 1 1 |
| 4 | 4 | 0 0 1 1 | 0 1 1 0 | 9 | 9 | 1 0 1 1 | 0 1 1 1 |

Table 1.11.2 Examples of Display Data (1/4 Duty LCD)

Further, examples of display data when a numeral display similar to Table 1.11.2 is made by using the 1/3 duty LCD are shown in Table 1.11.3. For the connecting method of COM terminals and SEG terminals, the example shown in Fig. 1.11.7 is used.

| Numeral | Display data memory | | | Numeral | Display data memory | | |
|---|---|---|---|---|---|---|---|
| | High order address | Middle order address | Low order address | | High order address | Middle order address | Low order address |
| 0 | * * 1 1 | * 1 0 1 | * 1 1 1 | 5 | * * 0 1 | * 1 1 1 | * 0 1 0 |
| 1 | * * 0 0 | * 0 0 0 | * 0 1 1 | 6 | * * 1 1 | * 1 1 1 | * 0 1 0 |
| 2 | * * 1 0 | * 1 1 1 | * 0 0 1 | 7 | * * 0 1 | * 0 0 1 | * 0 1 1 |
| 3 | * * 0 0 | * 1 1 1 | * 0 1 1 | 8 | * * 1 1 | * 1 1 1 | * 0 1 1 |
| 4 | * * 0 1 | * 0 1 0 | * 0 1 1 | 9 | * * 0 1 | * 1 1 1 | * 0 1 1 |

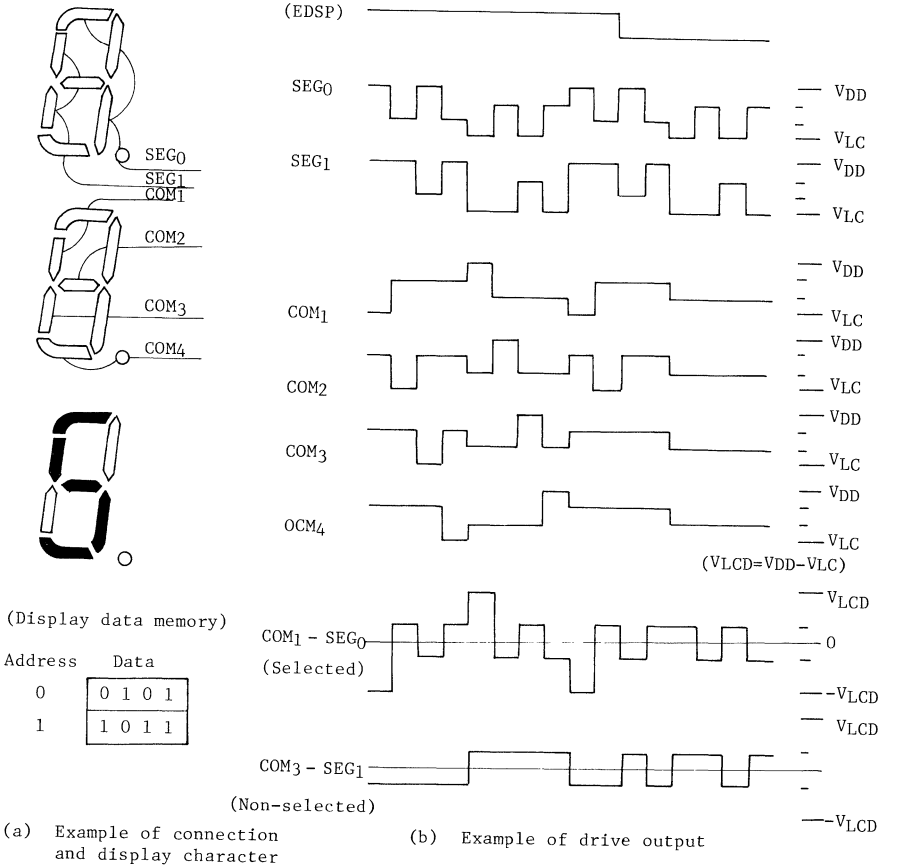
(* : don't care)

Table 1.11.3 Examples of Display Data (1/3 Duty LCD)

Display Output

The following are the examples of display output from LCD drive circuit according to each drive method.

1/4 Duty (1/3 Bias) Drive




(Display data memory)

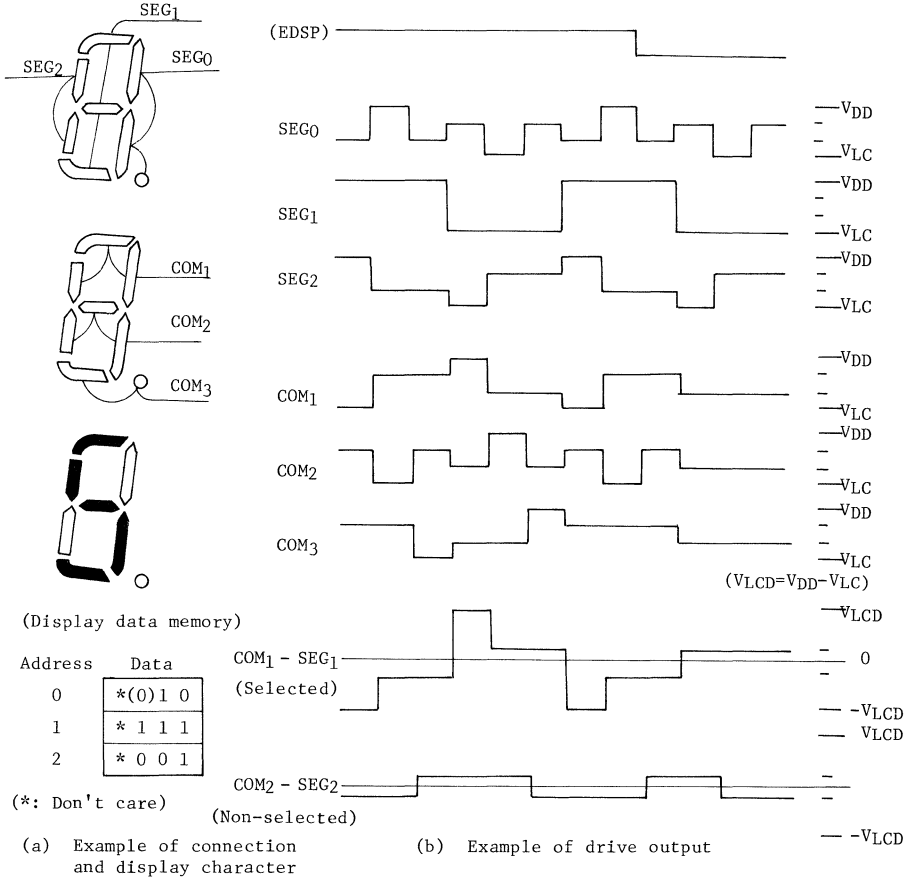
| Address | Data |
|---|---|
| 0 | 0 1 0 1 |
| 1 | 1 0 1 1 |

(a) Example of connection and display character

(b) Example of drive output

Fig. 1.11.6 Example of 1/4 Duty LCD Display Output

1/3 Duty (1/3 Bias) Drive




(Display data memory)

| Address | Data |
|---|---|
| 0 | *(0)1 0 |
| 1 | * 1 1 1 |
| 2 | * 0 0 1 |

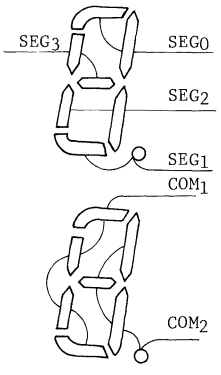
(*: Don't care)

(a) Example of connection and display character

(b) Example of drive output

Fig. 1.11.7 Example of 1/3 Duty LCD Display Output

1/2 Duty (1/2 Bias) Drive

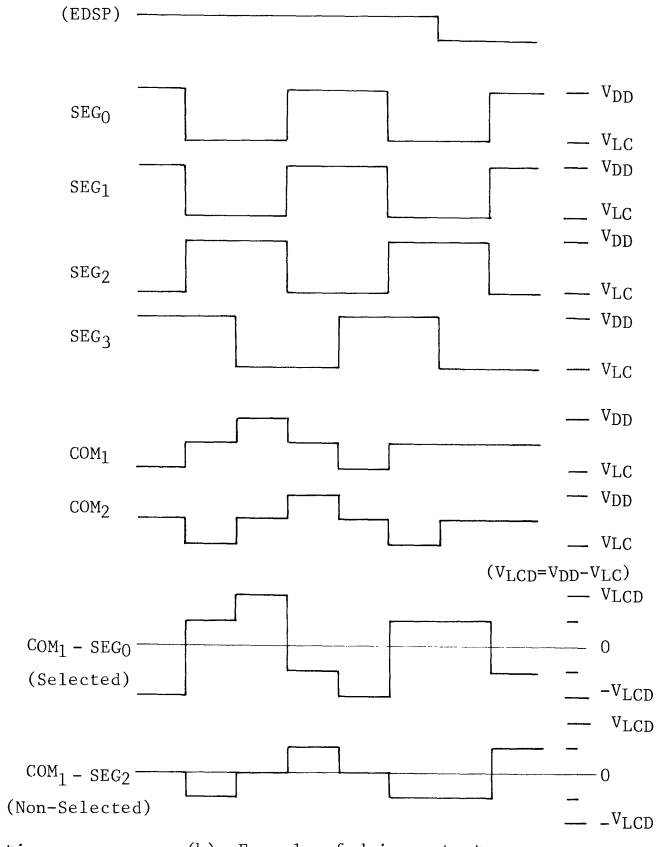



(Display data memory)

| Address | Data |
|---|---|
| 0 | * * 0 1 |
| 1 | * * 0 1 |
| 2 | * * 1 0 |
| 3 | * * 1 1 |

(*: Don't care)

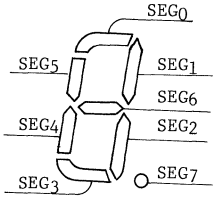
(a) Example of connection and display character




(b) Example of drive output

Fig. 1.11.8 Example of 1/2 Duty LCD Display Output

Static Drive



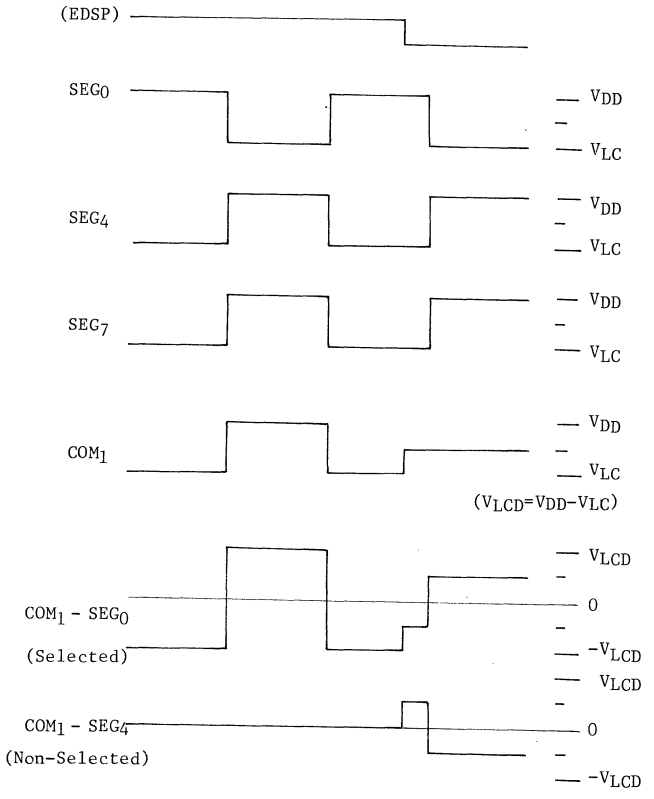


(Display data memory)

| Address | Data |
|---|---|
| 0 | * * * 1 |
| 1 | * * * 0 |
| 2 | * * * 1 |
| 3 | * * * 1 |
| 4 | * * * 0 |
| 5 | * * * 1 |
| 6 | * * * 1 |
| 7 | * * * 0 |

(*: Don't care)

(a) Example of connection and display character




(b) Example of drive output

Fig. 1.11.9 Example of Static LCD Display Output

2. Basic operation and pin operation

2.1 Instruction cycle, basic clock generation

As the oscillation circuit has been built in, when the external ($X_{IN}$, $X_{OUT}$) are connected to the oscillator, required clocks can be easily obtained. Further, this oscillation circuit is the Schmitt circuit. The clocks obtainable from the oscillation circuit are called the basic clock (CP, fc Hz). The basic clock is input into the timing generator and system control circuit from where various control signals are generated.

The instruction execution and the internal hardware control are synchronized with the basic clock. An instruction cycle consists of four machine cycles ($M_1 \sim M_4$), and each machine cycle requires four basic clock times.

2.2 Initialization operation, Hold function, interrupt input and others

Initialization operation is performed by keeping the $\overline{\text{RESET}}$ pin to the low level. By this initialize operation, the internal registers are initialized and at the same time, the LCD power switch is turned OFF. Further, no pull-up resistor is built in the $\overline{\text{RESET}}$ termianl of the TMP47C22F.

The hold function is the function to hold the status just before the system operation is stopped at low power consumption by making the most of the features of CMOS. The $\overline{\text{HOLD}}$ terminal is the signal input for the hold operation request and hold operation release request.

Two pins ($\overline{INT_1}$, $\overline{INT_2}$) are provided for the external interrupt input. Since these pins are common pins with $R_8$ port, they can be used as I/O pins respectively, if not used as the interrupt input pins. The interrupt latch is set by the falling edge of the external interrupt inputs.

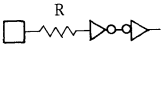
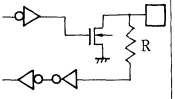
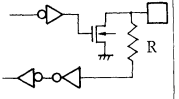
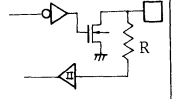
The TEST terminal is used at time of the shippint test. When a user's system is to be operated, low level voltage should be positively applied. Further, the TEST terminal of the TMP 47C22F has no built-in pull-down resistor.

2.3 Input/Output port

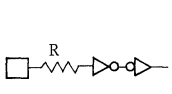
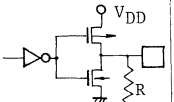
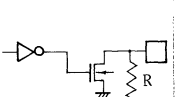
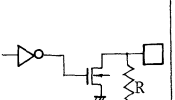
Input/output Circuit Format

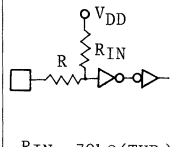
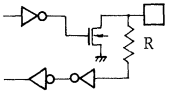
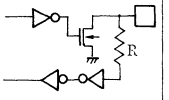
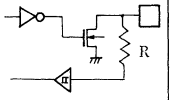
The input/output circuit format of the input/output port is shown following. For the TMP47C22F, any of the input/output circuit systems shown in the following tables can be selected. You can specify your input/output circuit system when requesting the program tape. "IOCODE GD" is employed if not specified.

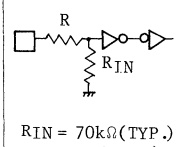
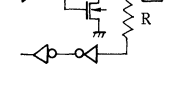
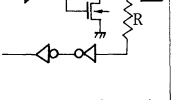
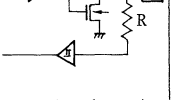
Input/Output Circuit Code ($\overline{\text{IOCODE}}$) GA

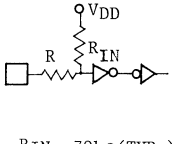
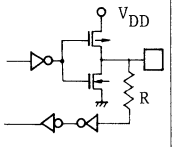
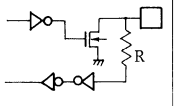
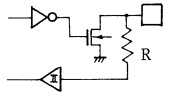
| Circuit \ Port | Input ($K_0$) | I/O ($R_4$, $R_5$, $R_6$) | I/O ($R_7$) | I/O ($R_8$, $R_9$) |
|---|---|---|---|---|
| I/O equivalent circuit |   R = 1kΩ (TYP.) |   R = 1kΩ (TYP.) |   R = 1kΩ (TYP.) |   R = 1kΩ (TYP.) |
| Remark | o No resistor is contained | o Sink open drain output<br>o Output latch is initialized to the high level | o Sink open drain output<br>o Output latch is initialized to the high level | o Schmitt circuit input<br>o Sink open drain output<br>o Output latch is initialized to the high level |

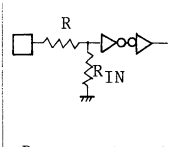
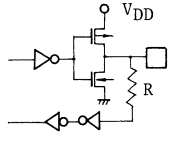
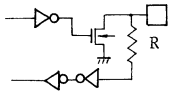
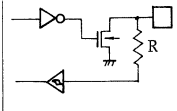
Input/Output Circuit Code ($\overline{\text{IOCODE}}$) GD

| Circuit \ Port | Input ($K_0$) | I/O ($R_4$, $R_5$, $R_6$) | I/O ($R_7$) | I/O ($R_8$, $R_9$) |
|---|---|---|---|---|
| I/O equivalent circuit |   R = 1kΩ (TYP.) |   R = 1kΩ (TYP.) |   R = 1kΩ (TYP.) |   R = 1kΩ (TYP.) |
| Remark | o No resistor is contained | o Push-pull output<br>o Output latch is initialized to the high level | o Sink open drain output<br>o Output latch is initialized to the high level | o Schmitt circuit input<br>o Sink open drain output<br>o Output latch is initialized to the high level |

| Input/Output Circuit Code (IOCODE) GB | | | | |
|---|---|---|---|---|
| Port / Circuit | Input ($K_0$) | I/O ($R_4$, $R_5$, $R_6$) | I/O ($R_7$) | I/O ($R_8$, $R_9$) |
| I/O equivalent circuit |  $R_{IN}$ = 70kΩ(TYP.) R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |
| Remark | o Pull-up resistor is contained | o Sink open drain output<br>o Output latch is initialized to the high level | o Sink open drain output<br>o Output latch is initialized to the high level | o Schmitt circuit input<br>o Sink open drain output<br>o Output latch is initialized to the high level |

| Input/Output Circuit Code (IOCODE) GC | | | | |
|---|---|---|---|---|
| Port / Circuit | Input ($K_0$) | I/O ($R_4$, $R_5$, $R_6$) | I/O ($R_7$) | I/P ($R_8$, $R_9$) |
| I/O equivalent circuit |  $R_{IN}$ = 70kΩ(TYP.) R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |
| Remark | o Pull-down resistor is contained | o Sink open drain output<br>o Output latch is initialized to the high level | o Sink open drain output<br>o Output latch is initialized to the high level | o Schmitt circuit input<br>o Sink open drain output<br>o Output latch is initialized to the high level |

| Input/Output Circuit Code (IOCODE) GE | | | | |
|---|---|---|---|---|
| Port / Circuit | Input ($K_0$) | I/O ($R_4$, $R_5$, $R_6$) | I/O ($R_7$) | I/O ($R_8$, $R_9$) |
| I/O equivalent circuit |  $R_{IN}$ = 70kΩ(TYP.)<br>R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |
| Remark | o Pull-up resistor is contained | o Push-pull output<br>o Output latch is initialized to the high level | o Sink open drain output<br>o Output latch is initialized to the high level | o Schmitt circuit input<br>o Sink open drain output<br>o Output latch is initialized to the high level |

| Input/Output Circuit Code (IOCODE) GF | | | | |
|---|---|---|---|---|
| Port / Circuit | Input ($K_0$) | I/O ($R_4$, $R_5$, $R_6$) | I/O ($R_7$) | I/O ($R_8$, $R_9$) |
| I/O equivalent circuit |  $R_{IN}$=70kΩ(TYP.)<br>R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |  R = 1kΩ (TYP.) |
| Remark | o Pull-down resistor is contained | o Push-pull output<br>o Output latch is initialized to the high level | o Sink open drain output<br>o Output latch is initialized to the high level | o Schmitt circuit input<br>o Sink open drain output<br>o Output latch is initialized to the high level |

3. Instructions

The TLCS-47 series microcomputer is provided with 90 instructions, which are software compatible within the series. The instructions of the TLCS-47 series is consist of 1-byte instructions or 2-byte instructions. To classify them in terms of the execution time, there are 1-cycle instructions and 2-cycle instructions.

1-byte, 1-cycle instructions are mainly used in this series, and are arranged so as to improve the program efficiency.

The TMP47C22F is software compatible with other versions of the TLCS-47 series; however, since it has no built-in output ports $P_1$ and $P_2$, (OUTB @HL) instruction cannot be used, so that 89 instructions become available as effective instructions.

| | | |
|---|---|---|
| 1-byte | 1-cycle instruction | 40 |
| 1-byte | 2-cycle instruction | 11 - 1 |
| 2-byte | 2-cycle instruction | 39 |
| | Total | 90-1 |

(a) Classification by byte/cycle

| | |
|---|---|
| Move instruction (Note) | 22 |
| Compare instruction | 6 |
| Arithmetic instruction | 16 |
| Logical instruction | 9 |
| Bit manipulation instruction | 24 |
| Input/Output instruction | 6 - 1 |
| Branch·subroutine instruction | 6 |
| Other instruction | 1 |
| Total | 90 - 1 |

(Note): Including ROM data referring instructions

(b) Classification by function

Table 3.0.1 Classification of instructions

| Item: Classification | Assembler: Mnemonic | Object Code: Binary: 1st Byte | Object Code: Binary: 2nd Byte | Object Code: Hexadecimal: 1st Byte | Object Code: Hexadecimal: 2nd Byte | Function | *1 Flags: CF | ZF | SF | Execution Cycle | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Move | LD A ,@HL | 00 00 11 00 | | 0 C | | (AC)←M[(H•L)] | – | Z | 1 | 1 | |
| | LD A , x | 00 11 11 00 $x_7x_6x_5x_4x_3x_2x_1x_0$ | | 3 C | $x_H$ $x_L$ | (AC)←M[x] | – | Z | 1 | 2 | |
| | LD HL, x | 00 10 10 00 $x_7x_6x_5x_4x_3x_2x_1x_0$ | | 2 8 | $x_H$ $x_L$ | (LR)←M[x'],(HR)←M[x'+1], x'=$x_7x_6x_5x_4x_3x_2$00 | – | – | 1 | 2 | |
| | LD A ,#k | 01 00 $k_3k_2k_1k_0$ | | 4 k | | (AC)←k | – | Z | 1 | 1 | |
| | LD H ,#k | 11 00 $k_3k_2k_1k_0$ | | C k | | (HR)←k | – | – | 1 | 1 | |
| | LD L ,#k | 11 10 $k_3k_2k_1k_0$ | | E k | | (LR)←k | – | – | 1 | 1 | |
| | LDL A ,@DC | 00 11 00 11 | | 3 3 | | (AC)←$ROM_L$[(DC)] | – | Z | 1 | 2 | |
| | LDH A ,@DC+ | 00 11 00 10 | | 3 2 | | (AC)←$ROM_H$[(DC)],(DC)←(DC)+1 | – | Z | 1 | 2 | *2 |
| | ST A ,@HL | 00 00 11 11 | | 0 F | | M[(H•L)]←(AC) | – | – | 1 | 1 | |
| | ST A ,@HL+ | 00 01 10 10 | | 1 A | | M[(H•L)]←(AC),(LR)←(LR)+1 | – | Z | $\overline{C}$ | 1 | *3 |
| | ST A ,@HL– | 00 01 10 11 | | 1 B | | M[(H•L)]←(AC),(LR)←(LR)–1 | – | Z | $\overline{B}$ | 1 | *3 |
| | ST A , x | 00 11 11 11 $x_7x_6x_5x_4x_3x_2x_1x_0$ | | 3 F | $x_H$ $x_L$ | M[x]←(AC) | – | – | 1 | 2 | |
| | ST #k,@HL+ | 11 11 $k_3k_2k_1k_0$ | | F k | | M[(H•L)]←k,(LR)←(LR)+1 | – | Z | $\overline{C}$ | 1 | *3 |
| | ST #k, y | 00 10 11 01 $k_3k_2k_1k_0y_3y_2y_1y_0$ | | 2 D | k y | M[y]←k | – | – | 1 | 2 | |
| | MOV H , A | 00 01 00 00 | | 1 0 | | (AC)←(HR) | – | Z | 1 | 1 | |
| | MOV L , A | 00 01 00 01 | | 1 1 | | (AC)←(LR) | – | Z | 1 | 1 | |
| | XCH A , H | 00 11 00 00 | | 3 0 | | (HR)⇄(AC) | – | Z | 1 | 2 | *2 |
| | XCH A , L | 00 11 00 01 | | 3 1 | | (LR)⇄(AC) | – | Z | 1 | 2 | *2 |
| | XCH A ,EIR | 00 01 00 11 | | 1 3 | | (EIR)⇄(AC) | – | – | 1 | 1 | |
| | XCH A ,@HL | 00 00 11 01 | | 0 D | | M[(H•L)]⇄(AC) | – | Z | 1 | 1 | *2 |
| | XCH A , x | 00 11 11 01 $x_7x_6x_5x_4x_3x_2x_1x_0$ | | 3 D | $x_H$ $x_L$ | M[x]⇄(AC) | – | Z | 1 | 2 | *2 |
| | XCH HL, x | 00 10 10 01 $x_7x_6x_5x_4x_3x_2x_1x_0$ | | 2 9 | $x_H$ $x_L$ | M[x']⇄(LR),M[x'+1]⇄(HR), x'=$x_7x_6x_5x_4x_3x_2$00 | – | – | 1 | 2 | |
| Compare | CMPR A ,@HL | 00 01 01 10 | | 1 6 | | null←M[(H•L)]–(AC) | $\overline{B}$ | Z | $\overline{Z}$ | 1 | |
| | CMPR A , x | 00 11 11 10 $x_7x_6x_5x_4x_3x_2x_1x_0$ | | 3 E | $x_H$ $x_L$ | null←M[x]–(AC) | $\overline{B}$ | Z | $\overline{Z}$ | 2 | |
| | CMPR A ,#k | 11 01 $k_3k_2k_1k_0$ | | D k | | null←k–(AC) | $\overline{B}$ | Z | $\overline{Z}$ | 1 | |
| | CMPR H ,#k | 00 11 10 00 11 01 $k_3k_2k_1k_0$ | | 3 8 | D k | null←k–(HR) | – | Z | $\overline{B}$ | 2 | |
| | CMPR L ,#k | 00 11 10 00 10 01 $k_3k_2k_1k_0$ | | 3 8 | 9 k | null←k–(LR) | – | Z | $\overline{B}$ | 2 | |
| | CMPR y ,#k | 00 10 11 10 $k_3k_2k_1k_0y_3y_2y_1y_0$ | | 2 E | k y | null←k–M[y] | $\overline{B}$ | Z | $\overline{Z}$ | 2 | |
| Arithmetic | INC A | 00 00 10 00 | | 0 8 | | (AC)←(AC)+1 | – | Z | $\overline{C}$ | 1 | |
| | INC L | 00 01 10 00 | | 1 8 | | (LR)←(LR)+1 | – | Z | $\overline{C}$ | 1 | |
| | INC @HL | 00 00 10 10 | | 0 A | | M[(H•L)]←M[(H•L)]+1 | – | Z | $\overline{C}$ | 1 | |
| | DEC A | 00 00 10 01 | | 0 9 | | (AC)←(AC)–1 | – | Z | $\overline{B}$ | 1 | |
| | DEC L | 00 01 10 01 | | 1 9 | | (LR)←(LR)–1 | – | Z | $\overline{B}$ | 1 | |
| | DEC @HL | 00 00 10 11 | | 0 B | | M[(H•L)]←M[(H•L)]–1 | – | Z | $\overline{B}$ | 1 | |
| | ADDC A ,@HL | 00 01 01 01 | | 1 5 | | (AC)←(AC)+M[(H•L)]+(CF) | C | Z | $\overline{C}$ | 1 | |
| | ADD A ,@HL | 00 01 01 11 | | 1 7 | | (AC)←(AC)+M[(H•L)] | – | Z | $\overline{C}$ | 1 | |
| | ADD A ,#k | 00 11 10 00 00 00 $k_3k_2k_1k_0$ | | 3 8 | 0 k | (AC)←(AC)+k | – | Z | $\overline{C}$ | 2 | |
| | ADD H ,#k | 00 11 10 00 11 00 $k_3k_2k_1k_0$ | | 3 8 | C k | (HR)←(HR)+k | – | Z | $\overline{C}$ | 2 | |
| | ADD L ,#k | 00 11 10 00 10 00 $k_3k_2k_1k_0$ | | 3 8 | 8 k | (LR)←(LR)+k | – | Z | $\overline{C}$ | 2 | |
| | ADD @HL,#k | 00 11 10 00 01 00 $k_3k_2k_1k_0$ | | 3 8 | 4 k | M[(H•L)]←M[(H•L)]+k | – | Z | $\overline{C}$ | 2 | |
| | ADD y ,#k | 00 10 11 11 $k_3k_2k_1k_0y_3y_2y_1y_0$ | | 2 F | k y | M[y]←M[y]+k | – | Z | $\overline{C}$ | 2 | |
| | SUBRC A ,@HL | 00 01 01 00 | | 1 4 | | (AC)←M[(H•L)]–(AC)–($\overline{CF}$) | $\overline{B}$ | Z | $\overline{B}$ | 1 | |
| | SUBR A ,#k | 00 11 10 00 00 01 $k_3k_2k_1k_0$ | | 3 8 | 1 k | (AC)←k–(AC) | – | Z | $\overline{B}$ | 2 | |
| | SUBR @HL,#k | 00 11 10 00 01 01 $k_3k_2k_1k_0$ | | 3 8 | 5 k | M[(H•L)]←k–M[(H•L)] | – | Z | $\overline{B}$ | 2 | |
| Logical | ROLC A | 00 00 01 01 | | 0 5 | | CF→AC→ (rotate left by 1 bit) | C | Z | $\overline{C}$ | 1 | *4 |
| | RORC A | 00 00 01 11 | | 0 7 | | AC→CF→ (rotate right by 1 bit) | C | Z | $\overline{C}$ | 1 | *4 |
| | AND A ,@HL | 00 01 11 10 | | 1 E | | (AC)←(AC)∧M[(H•L)] | – | Z | $\overline{Z}$ | 1 | |
| | AND A ,#k | 00 11 10 00 00 11 $k_3k_2k_1k_0$ | | 3 8 | 3 k | (AC)←(AC)∧k | – | Z | $\overline{Z}$ | 2 | |
| | AND @HL,#k | 00 11 10 00 01 11 $k_3k_2k_1k_0$ | | 3 8 | 7 k | M[(H•L)]←M[(H•L)]∧k | – | Z | $\overline{Z}$ | 2 | |
| | OR A ,@HL | 00 01 11 01 | | 1 D | | (AC)←(AC)∨M[(H•L)] | – | Z | $\overline{Z}$ | 1 | |
| | OR A ,#k | 00 11 10 00 00 10 $k_3k_2k_1k_0$ | | 3 8 | 2 k | (AC)←(AC)∨k | – | Z | $\overline{Z}$ | 2 | |
| | OR @HL,#k | 00 11 10 00 01 10 $k_3k_2k_1k_0$ | | 3 8 | 6 k | M[(H•L)]←M[(H•L)]∨k | – | Z | $\overline{Z}$ | 2 | |
| | XOR A ,@HL | 00 01 11 11 | | 1 F | | (AC)←(AC)∀M[(H•L)] | – | Z | $\overline{Z}$ | 1 | |

(continued)

| Item: Classification | Assembler Mnemonic | Object Code: Binary 1st Byte | Binary 2nd Byte | Hexadecimal 1st Byte | Hexadecimal 2nd Byte | Function | Flags *1 CF | Flags *1 ZF | Flags *1 SF | Execution Cycle |
|---|---|---|---|---|---|---|---|---|---|---|
| Bit Manipulation | TEST CF | 00 00 01 10 | | 0 6 | | (SP)←($\overline{CF}$),(CF)←0 | 0 | – | * | 1 |
| | TEST A , b | 01 01 11 $b_1b_0$ | | 5 C+b | | (SP)← (AC)<b> | – | – | * | 1 |
| | TEST @HL, b | 01 01 10 $b_1b_0$ | | 5 8+b | | (SP)←M[(H•L)]<b> | – | – | * | 1 |
| | TEST y , b | 00 11 10 01 10 $b_1b_0y_3y_2y_1y_0$ | | 3 9 8+b y | | (SP)←M[y]<b> | – | – | * | 2 |
| | TEST %p , b | 00 11 10 11 10 $b_1b_0p_3p_2p_1p_0$ | | 3 B 8+b p | | (SP)← P[p]<b> | – | – | * | 2 |
| | TEST @L | 00 11 01 11 | | 3 7 | | (SP)← P[(LR)<3:2>+4]<(LR)<1:0>> | – | – | * | 2 |
| | TESTP CF | 00 00 01 00 | | 0 4 | | (SP)← (CF),(CF)←1 | 1 | – | * | 1 |
| | TESTP ZF | 00 00 11 10 | | 0 E | | (SP)← (ZF) | – | – | * | 1 |
| | TESTP GF | 00 00 00 01 | | 0 1 | | (SP)← (GF) | – | – | * | 1 |
| | TESTP y , b | 00 11 10 01 11 $b_1b_0y_3y_2y_1y_0$ | | 3 9 C+b y | | (SP)←M[y]<b> | – | – | * | 2 |
| | TESTP %p, b | 00 11 10 11 11 $b_1b_0p_3p_2p_1p_0$ | | 3 B C+b p | | (SP)← P[p]<b> | – | – | * | 2 |
| | SET GF | 00 00 00 11 | | 0 3 | | (GF)← 1 | – | – | 1 | 1 |
| | SET @HL, b | 01 01 00 $b_1b_0$ | | 5 b | | M[(H•L)]<b>←1 | – | – | 1 | 1 |
| | SET y , b | 00 11 10 01 00 $b_1b_0y_3y_2y_1y_0$ | | 3 9 b y | | M[y]<b>← 1 | – | – | 1 | 2 |
| | SET %p , b | 00 11 10 11 00 $b_1b_0p_3p_2p_1p_0$ | | 3 B b p | | P[p]<b>← 1 | – | – | 1 | 2 |
| | SET @L | 00 11 01 00 | | 3 4 | | F[(LR)<3:2>+4]<(LR)<1:0>>←1 | – | – | 1 | 2 |
| | CLR GF | 00 00 00 10 | | 0 2 | | (GF)←0 | – | – | 1 | 1 |
| | CLR @HL, b | 01 01 01 $b_1b_0$ | | 5 4+b | | M[(H•L)]<b>←0 | – | – | 1 | 1 |
| | CLR y , b | 00 11 10 01 01 $b_1b_0y_3y_2y_1y_0$ | | 3 9 4+b y | | M[y]<b>←0 | – | – | 1 | 2 |
| | CLR %p , b | 00 11 10 11 01 $b_1b_0p_3p_2p_1p_0$ | | 3 B 4+b p | | P[p]<b>←0 | – | – | 1 | 2 |
| | CLR @L | 00 11 01 01 | | 3 5 | | P[(LR)<3:2>+4]<(LR)<1:0>>←0 | – | – | 1 | 2 |
| | CLR IL , r | 00 11 01 10 11 $r_5r_4r_3r_2r_1r_0$ | | 3 6 C+$r_{HR}$$r_{LR}$ | | (INTL)<5:0>←(INTL)<5:0>∧r<5:0> | – | – | 1 | 2 |
| | EICLR IL , r | 00 11 01 10 01 $r_5r_4r_3r_2r_1r_0$ | | 3 6 4+$r_{HR}$$r_{LR}$ | | (EIP)←0,(INTL)<5:0>←(INTL)<5:0>∧r<5:0> | – | – | 1 | 2 |
| | DICLR IL , r | 00 11 01 10 10 $r_5r_4r_3r_2r_1r_0$ | | 3 6 8+$r_{HR}$$r_{LR}$ | | (EIP)←0,(INTL)<5:0>←(INTL)<5:0>∧r<5:0> | – | – | 1 | 2 |
| Input/Output | IN %p , A | 00 11 10 10 00 10 $p_3p_2p_1p_0$ | | 3 A 2 p | | (AC)←P[p] | – | Z | $\overline{Z}$ | 2 |
| | IN %p ,@HL | 00 11 10 10 01 10 $p_3p_2p_1p_0$ | | 3 A 6 p | | M[(H•L)]←P[p] | – | – | $\overline{Z}$ | 2 |
| | OUT A ,%p | 00 11 10 10 10 $\overline{p_4}$0 $p_3p_2p_1p_0$ | | 3 A 8+2$\overline{p_4}$ p | | p[p]←(AC) , p = $p_4$•$p_3$ $p_2$ $p_1$ $p_0$ | – | – | 1 | 2 |
| | OUT @HL,%p | 00 11 10 10 11 $\overline{p_4}$0 $p_3p_2p_1p_0$ | | 3 A C+2$\overline{p_4}$ p | | p[p]←M[(H•L)], p = $p_4$•$p_3$ $p_2$ $p_1$ $p_0$ | – | – | 1 | 2 |
| | OUT #k ,%p | 00 10 11 00 $k_3k_2k_1k_0p_3p_2p_1p_0$ | | 2 C k p | | p[p]←k | – | – | 1 | 2 |
| | OUTB @HL | 00 01 00 10 | | 1 2 | | P[2]•P[1]←ROM[F•(B+(CF))•M[(H•L)]] | – | – | 1 | 2 |
| Branch•Subroutine | BS a | 01 10 $a_{11}a_{10}a_9a_8a_7a_6a_5a_4a_3a_2a_1a_0$ | | 6 $a_H$ $a_M$ $a_L$ | | If SF=1 then(PC)←a else null. | – | – | 1 | 2 | |
| | BSS a | 10 $d_5d_4d_3d_2d_1d_0$ | | 8+$d_H$ $d_L$ | | If SF=1 then(PC)←a else null,a=(PC)<11:6>+d | – | – | 1 | 1 | *5 |
| | CALL a | 00 10 0 $a_{10}a_9a_8a_7a_6a_5a_4a_3a_2a_1a_0$ | | 2 $a_H$ $a_M$ $a_L$ | | STACK[(SPW)]←(PC),(SPW)←(SPW)−1 , (PC)←a,0≦a≦2,047 | – | – | – | 2 | *5 |
| | CALLS a | 01 11 $n_3n_2n_1n_0$ | | 7 n | | STACK[(SPW)]←(PC),(SPW)←(SPW)−1, (PC)←a,a=8n+6(n≠0),134(n=0) | – | – | – | 2 | *5 |
| | RET | 00 10 10 10 | | 2 A | | (SPW)←(SPW)+1 , (PC)←STACK[(SPW)] | – | – | – | 2 | |
| | RETI | 00 10 10 11 | | 2 B | | (SPW)←(SPW)+1 , (FLAG•PC)←STACK[(SPW)], (EIF)←1 | * | * | * | 2 | |
| Other | NOP | 00 00 00 00 | | 0 0 | | no operation | – | – | – | 1 | |

Note 1. Setting Condition of Flag.
"C" indicates the carry output from the most significant position in the addition operation, and "B" indicates the borrow output from the most significant position in the subtraction operation.
"Z" indicates the zero detection signal to which "1" is applied only when either the ALU output of the processing result or all four bits of the data transferred to the accumulator are zero.
The flag is set to "C", "$\overline{C}$", "B", "Z", "$\overline{Z}$", "1", or "0" according to the data processing rusult. The value specified by the function is set to the flag with the mark "*", and the mark "–" denotes no change in the state of the flag.
Note 2. The zero flag is set according to the data set in the accumulator.
Note 3. The flags(ZF,SF)are set according to the result of increment or decrement of the L register.
NOte 4. The carry is the data shifted out from the accumulator.
Note 5. The contents of the program counter indicate the next address of the instruction to be executed.

## ELECTRICAL CHARACTERISTICS

### ABSOLUTE MAXIMUM RATINGS ($V_{SS}$ = 0V)

| SYMBOL | ITEM | RATING | UNIT |
|---|---|---|---|
| $V_{DD}$ | Supply Voltage | -0.5 ∿ 7 | V |
| $V_{LC}$ | Supply Voltage (LCD Drive) | -0.5 ∿ $V_{DD}$+0.5 | |
| $V_{IN}$ | Input Voltage | -0.5 ∿ $V_{DD}$+0.5 | V |
| $V_{OUT1}$ | Output Voltage (Except open drain terminal) | -0.5 ∿ $V_{DD}$+0.5 | V |
| $V_{OUT2}$ | Output Voltage (Open drain terminal) | -0.5 ∿ 10 | |
| $P_D$ | Poser Dissipation (Topr = 70°C) | 400 | mW |
| Tsld | Soldering Temperature · Time | 260 (10 sec.) | °C |
| Tstg | Storage Temperature | -55 ∿ 125 | |
| Topr | Operating Temperature | -30 ∿ 70 | |

### RECOMMENDED OPERATING CONDITIONS ($V_{SS}$ = 0V)

| SYMBOL | ITEM | CONDITION | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| Topr | Operating Temperature | | -30 | 70 | °C |
| $V_{DD}$ | Supply Voltage | | 4.5 | 6 | V |
| $V_{DDH}$ | Supply Voltage (Hold) | | 2 | 6 | |
| $V_{LC}$ | Supply Voltage (LCD Drive) | | 0 | $V_{DD}$-2.7 | |
| $V_{IH1}$ | High Level Input Voltage (Except Schmitt circuit input) | $V_{DD} \geq 4.5V$ | $V_{DD}$x0.7 | $V_{DD}$ | V |
| $V_{IH2}$ | High Level Input Voltage (Schmitt circuit input) | | $V_{DD}$x0.75 | $V_{DD}$ | |
| $V_{IH3}$ | High Level Input Voltage | $V_{DD} < 4.5V$ | $V_{DD}$x0.9 | $V_{DD}$ | |
| $V_{IL1}$ | Low Level Input Voltage (Except Schmitt circuit input) | $V_{DD} \geq 4.5V$ | 0 | $V_{DD}$x0.3 | |
| $V_{IL2}$ | Low Level Input Voltage (Schmitt circuit input) | | 0 | $V_{DD}$x0.25 | |
| $V_{IL3}$ | Low Level Input Voltage | $V_{DD} < 4.5V$ | 0 | $V_{DD}$x0.1 | |
| $f_C$ | Clock Frequency | | 0.4 | 4.2 | MHz |
| $t_{WCH}$ | High Level Clock Pulse Width (Note 1) | $V_{IN} = V_{IH}$ | 80 | - | nS |
| $t_{WCL}$ | Low Level Clock Pulse Width (Note 1) | $V_{IN} = V_{IL}$ | 80 | - | |

(Note 1) For external clock operation

## D.C. CHARACTERISTICS ($V_{SS}$=0V, $V_{DD}$=5V±10%, $T_{opr}$=−30 ∼ 70°C)

| SYMBOL | PARAMETER | | CONDITION | MIN. | TYP. (NOTE.1) | MAX. | UNIT |
|---|---|---|---|---|---|---|---|
| $V_{HS}$ | HYSTERESIS VOLTAGE (SCHMITT CIRCUIT INPUT) | | | – | [illegible] | – | V |
| $I_{IN1}$ | INPUT CURRENT (KO, $\overline{RESET}$, $\overline{HOLD}$, TEST) (NOTE2) | | $V_{DD}$=5.5V, $V_{IN}$=[illegible]/5.5V | – | – | ±[illegible] | μA |
| $I_{IN2}$ | INPUT CURRENT (OPEN DRAIN R PORT) | | | – | – | ±2 | |
| $I_{IL}$ | LOW LEVEL INPUT CURRENT (PUSH-PULL R PORT) | | $V_{DD}$=5.5V, $V_{IN}$=0.4V | – | – | −1 | mA |
| $R_{IN}$ | INPUT RESISTANCE (KO WITH INPUT RESISTOR) | | | 80 | [illegible] | 1[illegible] | kΩ |
| $I_{LO}$ | OUTPUT LEAKAGE CURRENT (OPEN DRAIN R PORT) | | $V_{DD}$=5.5V, $V_{OUT}$=5.5V | – | – | 2[illegible] | μA |
| $V_{OH}$ | OUTPUT VOLTAGE | HIGH LEVEL (PUSH-PULL R PORT) | $V_{DD}$=4.5V, $I_{OH}$=−200μA | 3.4 | – | – | V |
| $V_{OL}$ | | LOW LEVEL (R PORT) | $V_{DD}$=4.5V, $I_{OL}$=1.6mA | – | – | 0.4 | |
| $R_{OS4}$, $R_{OS5}$ | OUTPUT IMPEDANCE | HIGH·LOW LEVEL (SEG) (NOTE.4,5) | $V_{DD}$=5V, $V_{LCD}$(NOTE.3)=5V, $V_{OUT}$: $V_{DD}$−0.5V, $V_{LC}$+0.5V | – | 1.2 | T.B.D | kΩ |
| $R_{OC4}$, $R_{OC5}$ | | HIGH·LOW LEVEL (COM) (NOTE.4,5) | | – | 1.0 | T.B.D | |
| $R_{OS3}$, $R_{OS2}$ | | 2/3, 1/3 LEVEL (SEG) (NOTE.4) | $V_{DD}$=5V, $V_{LCD}$=5V, $V_{OUT}$=4.5V/3.5V | – | 1.0 | T.B.D | |
| $R_{OC3}$, $R_{OC2}$ | | 2/3, 1/3 LEVEL (COM) (NOTE.4) | | – | 1.2 | T.B.D | |
| $R_{OS1}$ | | 1/2 LEVEL (SEG) (NOTE.4) | $V_{DD}$=5V, $V_{LCD}$=5V, $V_{OUT}$=0.5±0.5V | | 1.0 | T.B.D | |
| $R_{OC1}$ | | 1/2 LEVEL (COM) (NOTE.5) | | – | 1.2 | T.B.D | |
| $V_{O3}$ | OUTPUT VOLTAGE | 2/3 LEVEL (SEG, COM) (NOTE.4) | $V_{DD}$=5V, $V_{LCD}$=5V | 4.3−0.2 | 4 | 4.3+0.2 | V |
| $V_{O1}$ | | 1/2 LEVEL (SEG, COM) (NOTE.5) | | 2.5−0.2 | 2.5 | 2.5+0.2 | |
| $V_{O1}$ | | 1/3 LEVEL (SEG, COM) (NOTE.4) | | 1.7−0.2 | 1.7 | 1.7+0.2 | |
| $I_{DDO}$ | SUPPLY CURRENT (AT OPERATING) (NOTE.6) | | $V_{DD}$=$V_{DDH}$=5.5V, $V_{LC}$=$V_{SS}$, $f_c$=4MHz | – | 2 | T.B.D | mA |
| $I_{DDH}$ | SUPPLY CURRENT (AT HOLDING) (NOTE.6) | | $V_{IN}$=5.3/0.2V (all valid), $C_L$=20pF, $C_{XIN}$=$C_{XOUT}$=15pF | – | 0.5 | T.B.D | μA |

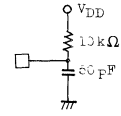
(NOTE.1) TYP. VALUES SHOW THOSE WHEN $T_{OPR}$=25°C, $V_{DD}$=5V.
(NOTE.2) WHEN THE KO PORT HAS A BUILT-IN INPUT RESISTER, CURRENT BY RESISTER IS EXCLUDED.
(NOTE.3) $V_{LCD}$: $V_{DD}$−$V_{LC}$.
(NOTE.4) SHOWS ON-RESISTANCE AT TIME OF LEVEL SWITCHING WHEN THE 1/4 OR 1/3 DUTY LCD IS USED.
(NOTE.5) SHOWS ON-RESISTANCE AT TIME OF LEVEL SWITCHING WHEN THE 1/2 DUTY OR STATIC LCD IS USED.
(NOTE.6) WHEN KO PORT HAS A BUILT-IN INPUT RESISTER, CURRENT VALUE IS THAT AT TIME OF OPEN. FURTHER, VOLTAGE LEVEL AT R PORT IS VALID.

## A.C. CHARACTERISTICS ($V_{SS}$=0V, $V_{DD}$=5V±10%, $T_{opr}$=−30 ∼ 70°C)

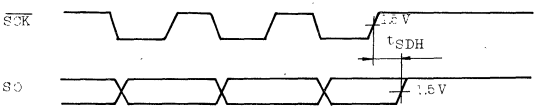
| SYMBOL | PARAMETER | CONDITION | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|---|
| tcy | INSTRUCTION CYCLE TIME | | 3.8 | – | 4.3 | μS |
| $t_{SDH}$ | SHIFT DATA HOLD TIME | (NOTE 1) | 3tcy−500 | – | – | nS |

## A.C. TIMING CHART

(NOTE.1) $\overline{SCK}$, SO TERMINAL EXTERNAL CIRCUIT

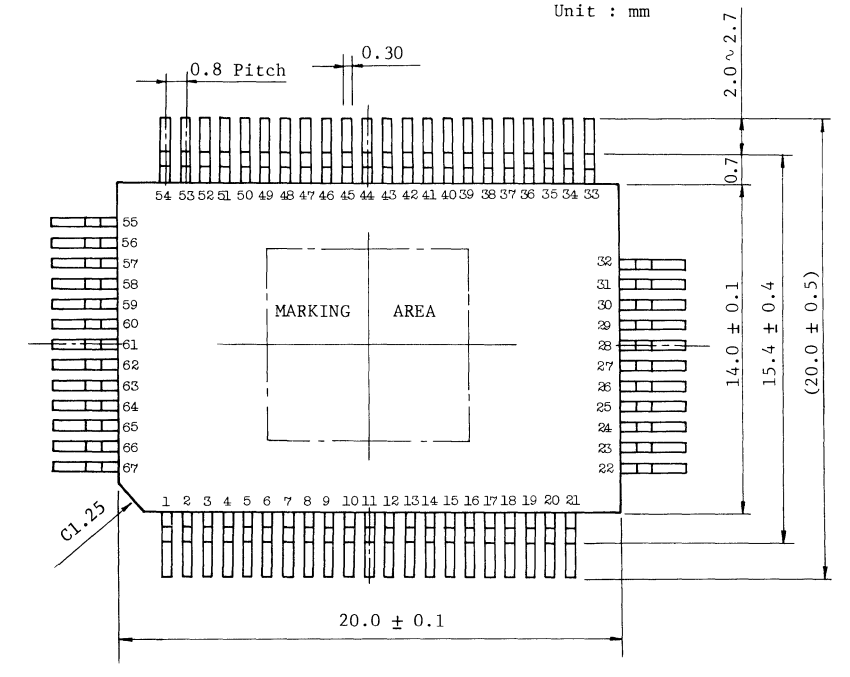


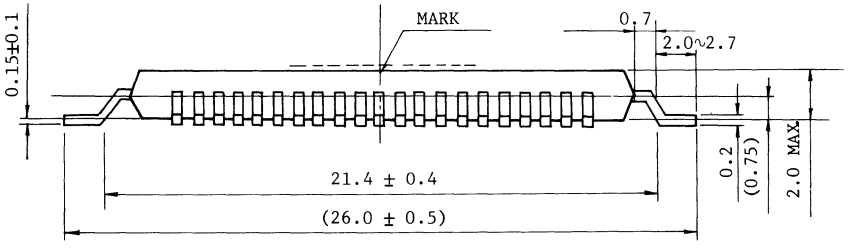

• SERIAL PORT (Completion of Transmission)



EXTERNAL DIMENSIONS

Unit : mm




Weight 1.3g (TYP.)

## Specification of program tape and input/output circuit format

The TMP47C22F will be able to made enginnering samples (ES) if you specify the program data and input/output circuit format by use of a paper tape.

The paper tape format is equivalent to the Hex. format of Intel Co. (Format I).

The program data should be specified within the address space corresponding to the built-in ROM capacity; the addresses 000 - 7FF denote the address range in the TMP47C22F.

1. Specification of input/output circuit format

The paper tape of Format I starts recording the program data after record mark ":", but the input/output circuit code should be specified just before the first record mark.

The "IŌCŌDE XX" format is used to define the input/output circuit code. XX denotes the proper input/output circuit code (two alphabets).

(Note) If the input/output circuit code is not specified, "IŌCŌDE GD" is employed. It should be noted that if the specified format is different from the standard one, and if the specified input/output circuit code is illegal, such specifications may be considered to have not been made.

(Example of tape list)

```
TŌSHIBA MICRŌCŌMPUTER TLCS-47
IŌCŌDE GD
:100000000665C7D79CF50F3F951FED55A8FF16E570
:1000100088884DDE76E31F5D8ABA6DF292F113F5C1
:100020004FF1F
                    :
                    :
                    :
:1007E000B53D42E0EC32546025B7308CDD52063D1D
:1007F000B4BE9E9E345B6138060B20BC372BF60BD6
:00000001FF
```

2. Program tape format (Format I)