CMOS 4-BIT MICROCONTROLLER

TMP42C40P, TMP42C60P

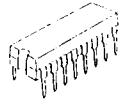
The 42C40/60 are compact, high speed 4-bit single chip microcomputers integrating ROM, RAM input/output ports, and interval timer. The 42C40/60 are the standard type devices in the TLCS-42 CMOS series.
The 42C40/60 are suitable for control of home appliances (such as fans, air-conditioners, refrigerators), audio equipments, games, toys and so on.

PART No.	ROM	RAM	PACKAGE	PIGGYBACK BOARD
TMP42C40P	512 x 8-bit	32 x 4-bit	DIP16	BM4211B
TMP42C60P	1024 x 8-bit			

FEATURES

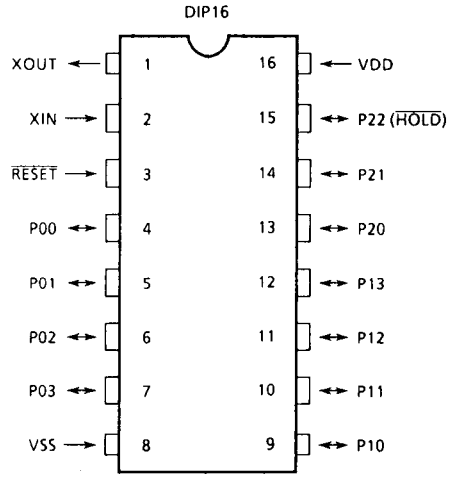
- ◆4-bit single chip microcomputer
- ◆Instruction execution time :
  1.0μs (at 5MHz, 4.5 to 6.0V)
  2.5μs (at 2MHz, 4.0 to 6.0V)
- ◆44 basic instructions
  - All instructions are one byte object code
  - Table look-up instructions
- ◆Stack for subroutine call : 1 level
- ◆I/O port (11 pins)
  - I/O 3ports 11pins
    (all ports are programmable I/O)
- ◆Interval Timer (14 stages)
- ◆Hold function
  Battery/Capacitor back-up
- ◆Clock generator
  Ceramic resonator / RC oscillation (mask option)
- ◆Real Time Emulator : BM4221A

DIP16

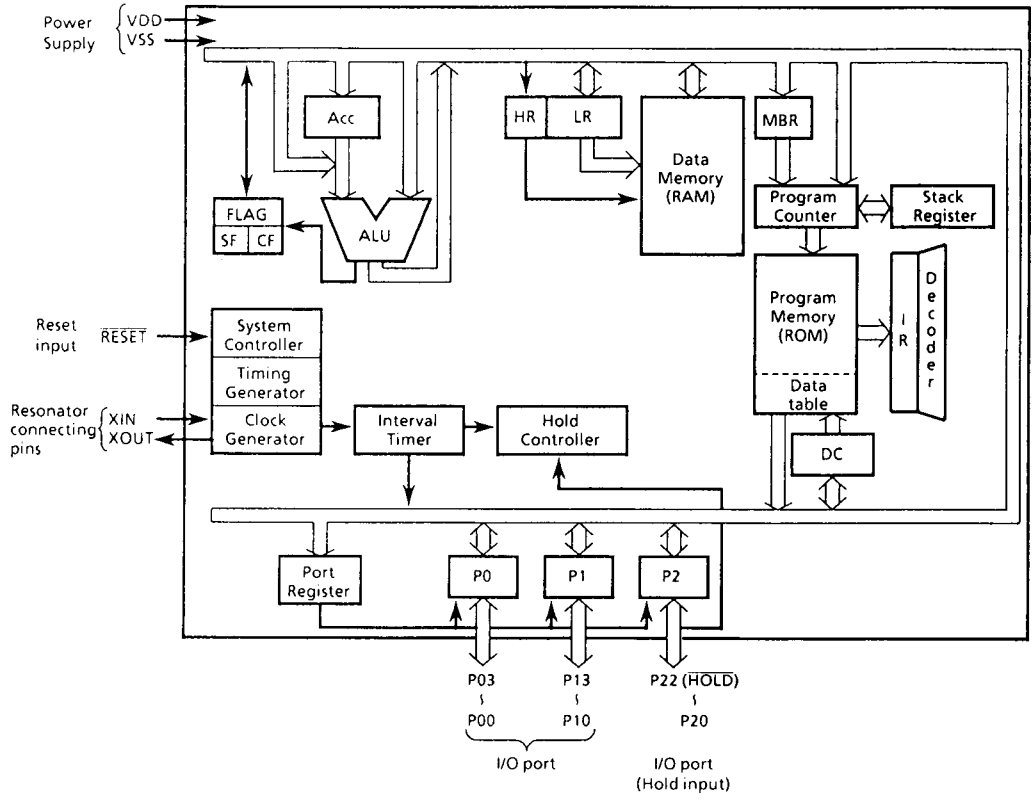


TMP42C40P
TMP42C60P

PIN ASSIGNMENT (TOP VIEW)



BLOCK DIAGRAM



PIN FUNCTION

PIN NAME	INPUT/OUTPUT	FUNCTION
P03 - P00	I/O	4-bit programmable I/O ports with latch, input/output mode can be specified by [MOV A, P] instruction.
P13 - P10		
P22 ($\overline{HOLD}$)	I/O (INPUT)	3-bit programmable I/O port with latch. Hold request/release signal input
P21	I/O	The four input/output modes can be selected with an I/O control instruction [MOV A, P].
P20		
XIN	INPUT	Resonator connecting pins. For inputting external clock, XIN is used and XOUT is opened.
XOUT	OUTPUT	
$\overline{RESET}$	INPUT	Reset signal input
VDD	Power Supply	+ 5V
VSS		0V (GND)

OPERATIONAL DESCRIPTION

1. SYSTEM CONFIGURATION

(1) Program Counter (PC)
(2) Memory Bank Register (MBR)
(3) Stack Register
(4) Data Counter (DC)
(5) Program Memory (ROM)
(6) H Register(HR), L Register (LR)
(7) Data Memory (RAM)
(8) ALU, Accumulator
(9) Flags
(10) Clock Generator, Timing Generator
(11) I/O Ports
  a. Port Register (PR)
(12) Interval Timer
(13) Hold Controller
(14) Reset Circuit

Concerning the above component parts, the configuration and functions of hardwares are described.

2. INTERNAL CPU FUNCTION

2.1 Program Counter (PC)

The program counter is a 10-bit binary counter which indicates the address of the program memory storing the next instruction to be executed. Normally, the PC is incremented every time the instruction is fetched. When a branch instruction or a subroutine instruction has been executed the specified values listed in Table 2-1 are set to the PC. The PC is initialized to "0" during reset.

The PC can directly address a 1024-byte address space. However, with the branch and subroutine call instructions, the following points must be considered:

(1) Branch instruction [BSS a], [LD MBR, #k]
In executing [BSS a] instruction, when the branch condition is satisfied (SF = 1), the value specified in the instruction is loaded to the lower 6 bits of the after first incrementing the PC. That is, [BSS a] becomes the in-page branch instruction. When [BSS a] is stored at the last address of the page, the upper 4 bits of the PC point the next page, so that branch is made to the next page.

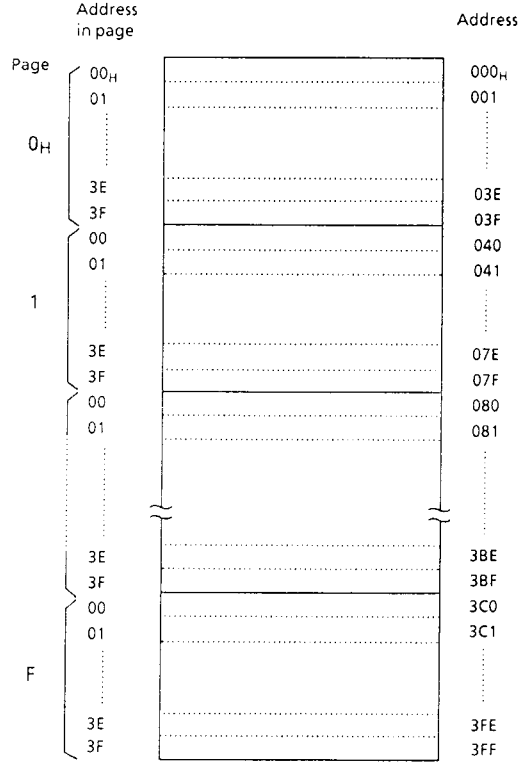


Figure 2-1. Configuration of Program Memory

In addtion, branching to the entire program memory space is possible when SF = 1 by executing [BSS a] instruction immediately after [LD MBR, #k] instruction. At that time, the contents of the memory bank register (MBR) are loaded to the upper 4 bits of the PC.

(2) Subroutine call instruction [CALLS a]
In executing [CALLS a] instruction, the value specified by the instruction is loaded to the PC after the contents of the PC are saved to the stack register. The entry address of the subroutine should be an even number address within the range of 002_H through $01E_H$.

Instruction or Operation		Condition		Program Counter (PC)									
				Page Assignment				Address assignment in page					
				PC_9	PC_8	PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0
Execution of Instruction	LD MBR, #k + BSS a	SF = 1 (Branch condition is satisfied)		contents of MBR				immediate data specified by the instruction					
		SF = 0 (Branch condition is not satisfied)		+2									
	BSS a	SF = 1	Lower 6-bit address ≠ 111111	hold				immediate data specified by the instruction					
			Lower 6-bit address = 111111	+1				immediate data specified by the instruction					
		SF = 0		+1									
	CALLS a			0	0	0	0	0	immediate data specified by the instruction				0
	RET			return address restored from stack register									
	Others			+1									
Reset				0	0	0	0	0	0	0	0	0	0

Table2-1. Configuration and Status change of Program Counter

2.2 Memory Bank Register (MBR)

The memory bank register is a 4-bit register used only for writing to specify pages when branching to the entire program memory space.
Specified values are loaded to the MBR by [LD MBR, #k] instruction. The contents of the MBR are loaded to the upper 4 bits of the PC only when [BSS a] instruction is executed immediately after [LD MBR, #k] instruction. If another instruction is executed after [LD MBR, #k] , followed by [BSS a], the contents of the MBR are not loaded to the PC.

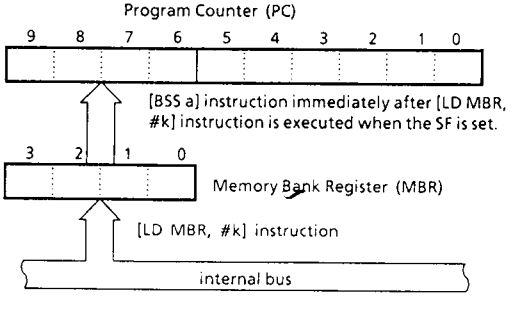


Figure 2-2. Operation of Memory Bank Register

2.3 Stack Register (STACK)

The stack register is a 10-bit register used to save the contents of the PC (return address) before jumping to a subroutine when the [CALLS a] instruction is executed.
One level subroutine can be used. If there are two calls, the second return address is saved but the first return address is lost.
When returning from a subroutine, contents of the STACK are restored to the PC by [RET] instruction.

Example : When [CALLS a] instruction is stored at address $08E_H$, "$08F_H$" is saved to the STACK when this instruction is executed. The contents of the STACK ($08F_H$) are restored to the PC by [RET] instruction.

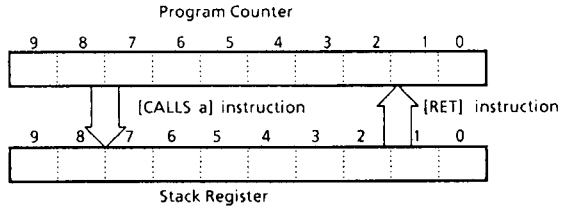


Figure 2-3. Operation of Stack Register

2.4 Data Counter (DC)

The data counter is a 4-bit register to specify addresses when fixed data stored in the data table of the program memory (ROM) are read out.

In addition to a function for transferring data from accumulator, the DC also has increment and decrement functions ; therefore, it can also be used as a general-purpose register.

The fixed data can be read out using the table look-up instructions.

(1) Table look-up instructions [LDL A,@DC], [LDH A,@DC]

The table look-up instructions read either the upper 4 bits or the lower 4 bits of the fixed data stored at the address specified by the data counter (DC) and stores them into the accumulator.

[LDL A,@DC] instruction reads the lower 4 bits, and [LDH A,@DC] instruction reads the upper 4 bits.

When the table look-up instruction is executed, the lower 4 bits of the ROM address are the contents of the data counter and all of the upper 6 bits become "1".

Consequently, the final 16 bytes ($1F0_H$-$1FF_H$ for the 42C40 and $3F0_H$-$3FF_H$ for the 42C60) of the program memory are specified by the DC and can also be used as fixed data.

Example : If the contents of the DC is "6" and the contents of program memory address $3F6_H$ is "$3E_H$" : "E_H" will be stored to the accumulator when [LDL A,@DC] instruction is executed and "3_H" is stored to the accumulator when [LDH A,@DC] instruction is executed.

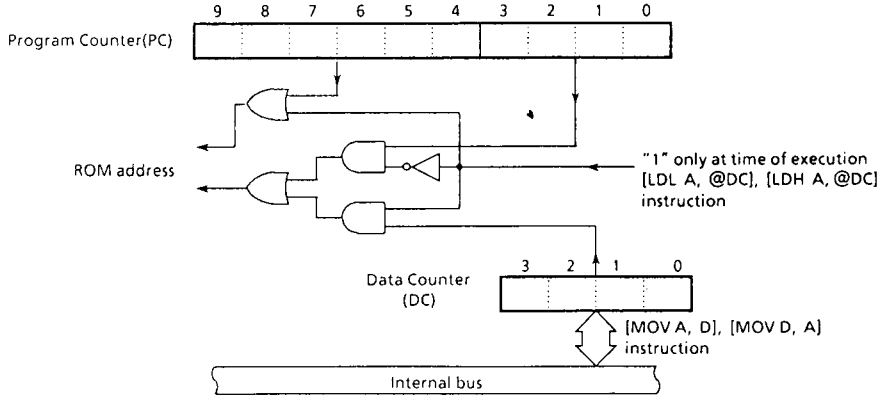


Figure 2-4. Operation of Data Counter

2.5 Program Memory (ROM)

The 42C40 has 512 x 8 bits (addresses 000_H through $1FF_H$) of program memory (mask ROM), the 42C60 has 1024 x 8 bits (addresses 000_H through $3FF_H$).

On the 42C40, no physical program memory exists in the address range 200_H through $3FF_H$. However, if this space is accessed by program, the most significant bit of each address is always regard as "0" and the contents of the program memory corresponding to the address 000_H through $1FF_H$ are read.

Programs and fixed data are stored in the program memory. The instruction to be executed next is read from the address indicated by the contents of the PC. The fixed data can be read by using the table look-up instructions.

2.5.1 Program Memory Map

Figure 2-5 shows the program memory map. Address 000_H through $01E_H$ and $3F0_H$ through $3FF_H$ ($1F0_H$ through $1FF_H$ for the 42C40) of the program memory are also used for special purposes.

Example : When the data at address $3F6_H$ are read out and stored to the accumulator with the table look-up instruction, the data at address $1F6_H$ will actually be stored. That is, the data table for the 42C40 is located at addresses $1F0_H$ through $1FF_H$.

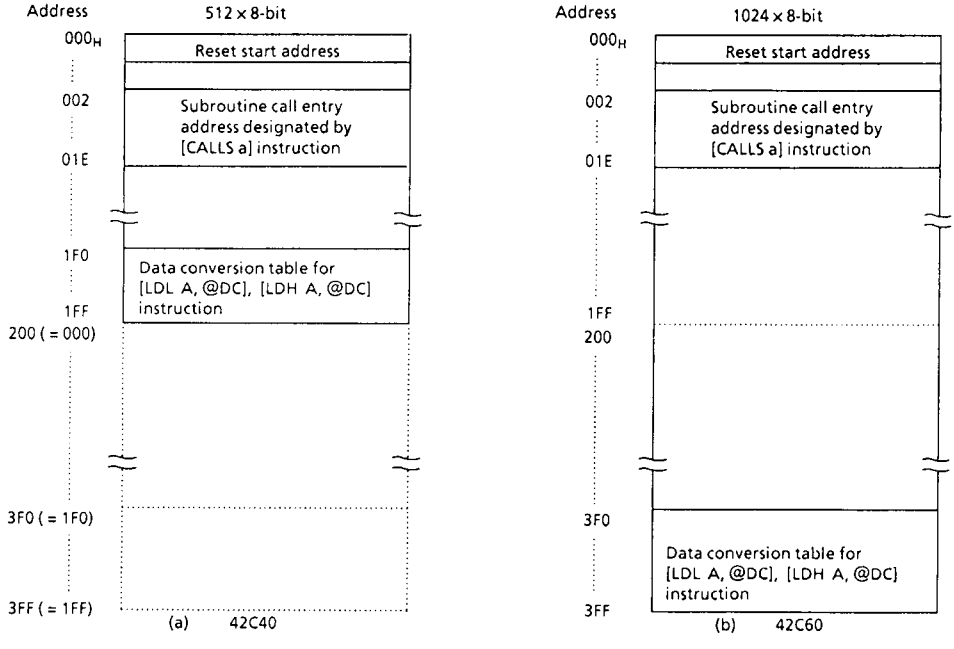


Figure 2-5. Program Memory Map

2.6 H Register (HR), L Register (LR)

The H register is a 1-bit register and the L register is a 4- bit register. The H and L registers form a pair (HL) and are used as the pointer for specifying data memory (RAM) addresses. The L register can also be used as a general-purpose register.

The data memory is configured in 1-page, 16-word (1 word = 4 bits) units. The H register specifies pages and the L register specifies addresses within the page.

2.7 Data Memory (RAM)

The 42C40/60 have 32 x 4 bits (addresses 00_H through $1F_H$) of the data memory (RAM).

Figure 2-6 shows configuration of HR, LR and data memory.

The data memory is addressed in one of the two ways (addressing modes):

(1) Register-indirect addressing mode

In this mode, a page is specified by the H register and an address in the page by L register.

Example : To write immediate data "B_H" to address $1E_H$ in the RAM.

```
SET     H               ; HR ← 1
LD      L, #0EH         ; LR ← EH
ST      #0BH, @HL       ; RAM [1EH] ← BH
```

(2) Direct addressing mode

The address of 8 words on page 0 of the data memory can be specified directly in the lower 3 bits of the instruction field. The transfer of data to the accumulator with [LD A, x] and [ST A, x] instructions is executed with 1 byte without being affected by the HL register pair.

Example : To load the contents of data memory address 05_H into the accumulator.

```
LD      A, 05H          ; Acc ← RAM [05H]
```

When power-on is performed, the contents of the RAM become unpredictable, so that they must be initialized by the initialization routine.

Example : To clear RAM

```
        LD      L, #0           ; LR ← 0
SCLR RAM: CLR   H               ; HR ← 0
        ST      #0, @HL         ; RAM [HL] ← 0
        SET     H               ; HR ← 1
        ST      #0, @HL         ; RAM [HL] ← 0
        ADD     L, #1           ; LR ← LR+1
        B       SCLR RAM
```

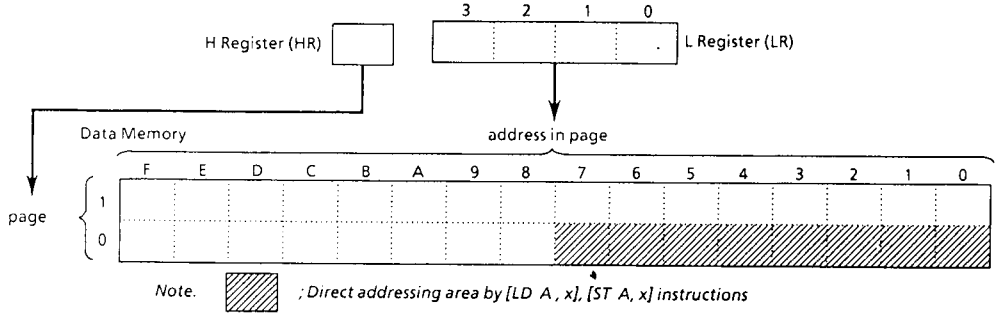


Figure 2-6. Configuration of HR, LR and Data Memory

2.8 ALU and Accumulator

2.8.1 Arithmetic / Logic Unit (ALU)

The ALU performs the arithmetic and logic operations specified by instructions on 4-bit binary data and outputs the result of the operation, the carry information (C), and the zero detect information (Z).

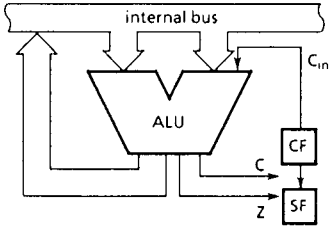
(1) Carry information (C)

The carry information indicates a carry-out from the most significant bit in an addition. A subtraction is performed as addition of two's complement, so that, with a subtraction, the carry information indicates that there is no borrow to the most significant bit.

(2) Zero detect information (Z)
This information is "1" when the logic operation result or the input data is "0000_B".

Example : The carry information (C) and zero detect information (Z) for 4-bit operations.

Operation				Result	C	Z
4	+	2	=	6	0	–
7	+	9	=	0	1	–
8	–	1	=	7	1	–
5	–	8	=	$-3\ (1101_B)$	0	–
6	$\vee$	9	=	F_H	–	0
2	$\wedge$	C_H	=	0	–	1



Note. C_{in} *indicates the carry input specified by instruction.*

Figure 2-7. ALU and Flags

2.8.2 Accumulator (Acc)

The accumulator is a 4-bit register used to hold source data or results of the operations and data manipulations.

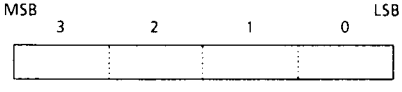


Figure 2-8. Accumulator

2.9 Flags (FLAG)

There are a carry flags (CF) and a status flags (SF) which can be set and cleared under conditions specified by instructions. During reset, the SF is initialized to "1", the CF is not affected.

(1) Carry Flag (CF)
The CF holds the carry information (C) from the ALU when an addition or subtraction instruction involving a carry is executed. Also, CF manipulation instructions can be set to "1" or cleared to "0".

① Addition/substruction with carry instructions [ADDC A, @HL], [SUBRC A, @HL]
The CF becomes the input (C_{in}) to the ALU to hold the carry information (carry/non-borrow).

② Carry flag manipulation instruction [SET CF], [CLR CF]
The CF is set to "1" by [SET CF], and is cleared to "0" by [CLR CF] instruction.

③ Carry flag test instructions [TESTP CF]
The CF is not affected by this instruction.

(2) Status Flag (SF)
The SF provides the branch condition for a branch instruction. Branch is performed when this flag is set to "1". Normally the SF is set to "1", so that any branch instruction can be regarded as an unconditional branch instruction. When a branch instruction is executed upon set or clear of the SF according to the condition specified by an instruction, this instruction becomes a conditional branch instruction.

Example : The SF and Branch instruction

```
ADD     A, #1       ; Acc ← Acc + 1
BSS     SLAB1       ; Conditional branch (Branches if carry-out
                    ; occurs by the operation immediately before.)
LD      A, @HL      ; Acc ← RAM [HL]
BSS     SLAB2       ; Unconditional branch (The SF is set to "1" by
                    ; the instruction immediately before.)
```

2.10 Clock Generator and Timing Generator

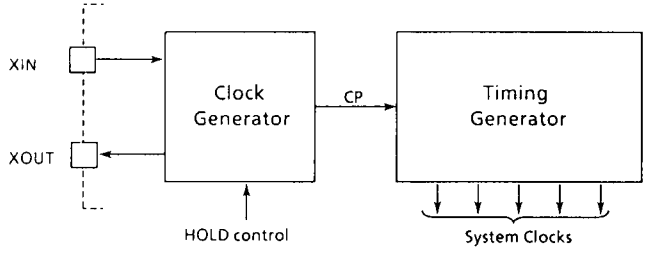


Figure 2-9. Clock Generator and Timing Generator

2.10.1 Clock Generator

The clock generator is the basic clock (CP) generation circuit used as reference for the system clock supplied to the CPU and peripheral hardware. The basic clock can easily be obtained by connecting an oscillator to the XIN and XOUT pins (RC oscillation is also possible, depending on the mask option). Input from an external clock is also possible.

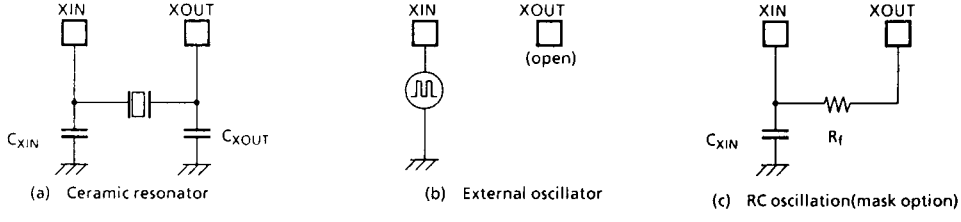


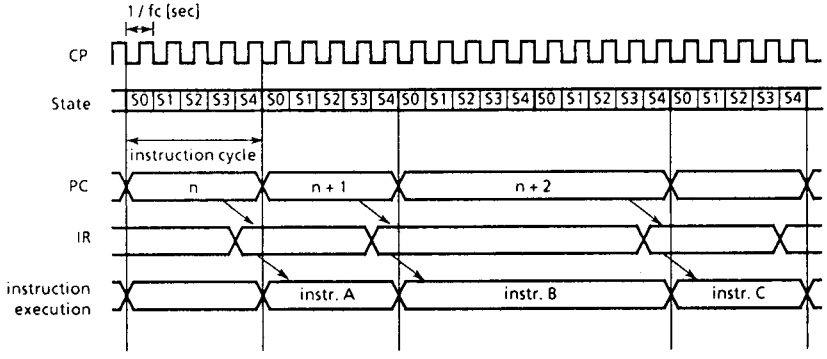
Figure 2-10. Example of Oscillator Connection

2.10.2 Timing Generator

The timing generator produces the system clocks from basic clock pulse which are supplied to the CPU and the peripheral hardware.

2.10.3 Instruction Cycle

The instruction execution and the on-chip peripheral hardware operations are performed in synchronization with the basic clock pulse (CP : fc [Hz]). The smallest unit of instruction execution is called an instruction cycle. The instruction set of the TLCS-42 series consists of 1-cycle instructions and 2-cycle instructions. The former requires 1 cycle for their execution ; the latter, 2 cycles. Each instruction cycle consists of 5 states (S0 through S4). Each state consists of one basic clock pulse, and the instruction cycle time is 5/fc [sec]. Figure 2-11 shows CPU basic operation.



Example : instr. A ··· ADD A, @HL (1-cycle instruction)
instr. B ··· LD A, x (2-cycle instruction)
instr. C ··· INC @HL (1-cycle instruction)

Figure 2-11. Instruction Cycle

3. PERIPHERAL HARDWARE FUNCTION

3.1 Ports

Input/output instructions (4 types) are used for data transfer to external circuits and for internal circuits control.

① I/O port ; Data transfer with external circuit
② Command register ; Control of internal circuit

There are two types of ports to which the port addresses 00_H-05_H are assigned. Any of these ports can be selected by specifying the address using an input/output instruction. Input/Output selection of programmable input/output port is controlled by the port register.

3.1.1 Port Register (PR)

The port register is a 4-bit dedicated register for writing and which selects the input/output modes of the programmable input/output ports.

The P0, P1 and P2 ports of the TLCS-42 CMOS series are programmable I/O ports, so data must be loaded to the port register and the I/O mode must be selected by executing the I/O control instruction [MOV A, P]. Also, the port register is initialized to "0" during reset. Consequently, all of the programmable I/O ports are set input mode until the I/O control instruction is executed.

Example : Sets ports P03-P00, P22, P21 to input mode, and P13-P10, P20 to output mode.

```
LD    A, #0110B    ; PR ← 0110B
MOV   A, P
```

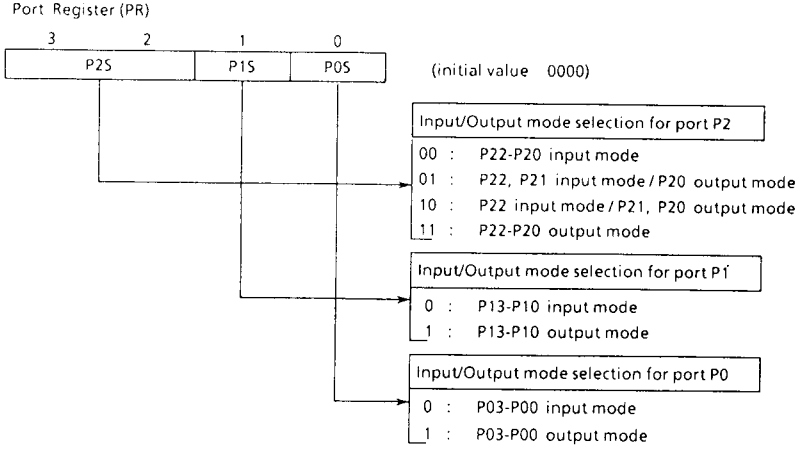


Figure 3-1. Port Register and Control of programmable I/O ports

3.1.2 I/O Ports

The 42C40/60 have 3 I/O ports (11 pins) each as follows:

① P0, P1 ; 4-bit programmable I/O
② P2 ; 3-bit programmable I/O (P22 pin is shared by hold request/release signal input pin.)

Output data are held by the latch when the output instructions [OUT A, %p] and [OUT @HL, %p] are executed, regardless of the I/O mode of the port. There is no input latch so external input data are either held externally until read, or are read several times before processing, when necessary.

(1) Ports P0 (P03-P00), P1 (P13-P10), P2 (P22-P20)

Ports P0, P1 and P2 are programmable input/output ports. P0 and P1 are 4-bit ports, and P2 is a 3-bit port.

There are 16 I/O modes which can be selected by programs. When an output instruction is executed in one of the input modes, the data are merely held by the latch and are not output to a pin ; however, the latched data can be output to a pin by executing the input/output control instruction. The latch is initialized to "1" during reset.

P22 pin is also used as the hold operation release signal input pin ($\overline{HOLD}$) ; therefore, it is necessary to set an input mode before starting a hold operation. Hysteresis input is used for the hold operation release signal.

The highest bit (P23) of port P2 is not included but interval timer output (ITS) is read when an input instruction is executed.

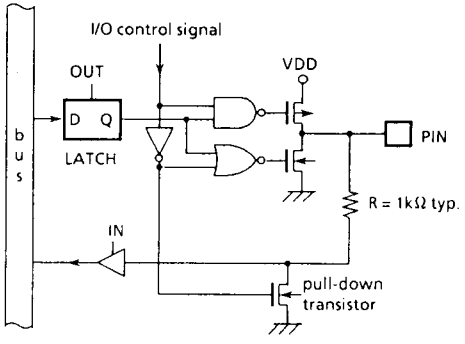


Figure 3-3. Circuitry of programmable I/O port

Port P0 (Port address OP00 / IP00)

3	2	1	0
P03	P02	P01	P00

Port P1 (Port address OP01 / IP01)

3	2	1	0
P13	P12	P11	P10

Port P2 (Port address OP02 / IP02)

3	2	1	0
ITS	P22 ($\overline{HOLD}$)	P21	P20

Figure 3-2. Ports P0, P1, P2

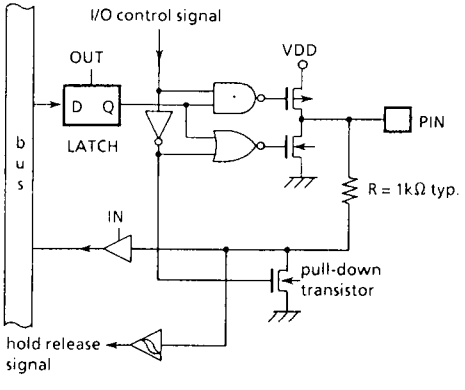


Figure 3-4. Circuitry of P22 ($\overline{HOLD}$) pin

3.1.3 I/O Timing

(1) Input timing

External data is read from an input port or an I/O port in the S1 state of the second instruction cycle during the input instruction (2-cycle instruction) execution. This timing cannot be recognized from the outside, so that the transient input such as chattering must be processed by program.

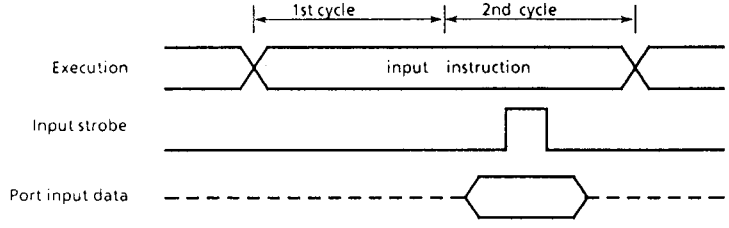


Figure 3-5. Input Timing

(2) Output timing

Data is output to an output port or an I/O port in the S3 state of the second instruction cycle during the output instruction (2-cycle instruction) execution. The port changes at the rise of the output latch pulse.

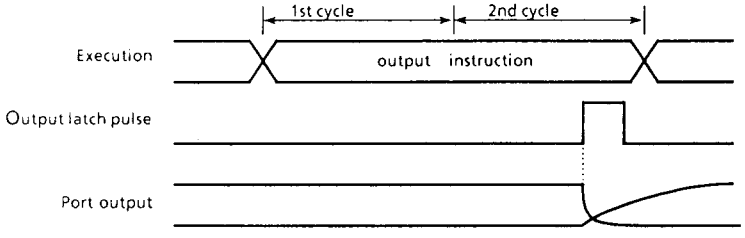


Figure 3-6. Output Timing

3.2 Interval Timer

The interval timer divides the basic clock (CP) with a 14-stage binary counters and an output constant frequency pulses. The following are interval applications. The intirval timer is cleared to "0" when the hold operation mode is released.

① Generation of an interval time with a fixed frequency (the interval timer output)

② Generation of warm-up time at release of the hold operating mode

3.2.1 Interval Timer Output

Interval timer produces pulse output which is the basic clock (frequency: fc) divided by 2048. These are approximately 1ms pulses when oscillation frequency is fc = 2MHz.

Interval timer output is transferred to the accumulator or data memory as port P2, bit 3 (P23) data by the port P2 input instructions [IN %IP02, A] and [IN %IP02, @HL].

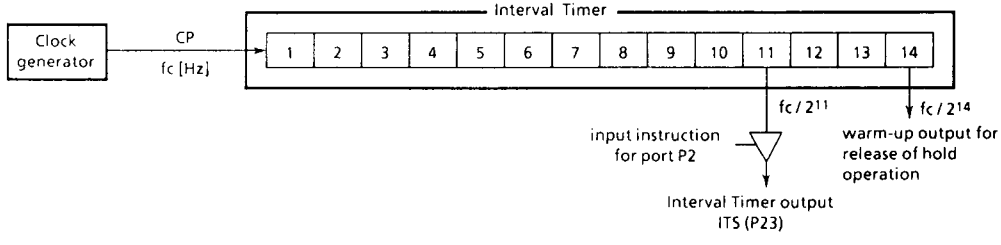


Figure 3-7. Interval Timer

Example : To wait until the interval timer output data are changed.

```
        IN      %IP02,@HL       ; RAM [HL] ← IP02
SWAIT : IN      %IP02,A         ; Acc ← IP02
        XOR     A,@HL           ; The MSB of the Acc and of the RAM are
        ADD     A,#8              compared and the results transferred
                                  to the SF.
        B       SWAIT           ; When the results match, there is a
                                  branch to SWAIT.
```

4. POWER SAVING FUNCTION

The 42C40/60 have the hold operating mode intended to save the power.

4.1 Hold Operating Mode

The hold function stops system operation and holds the internal status active just before stop. Hold operation controlled by the $\overline{HOLD}$ pin and HOLD instruction The $\overline{HOLD}$ pin is also used as the P22 pin.

4.1.1 Start of Hold operating Mode

Hold operation is started by executing the HOLD instruction and continues as long as the $\overline{HOLD}$ pin is at low level. The following statuses are held during hold operation:

① Oscillation and all internal operations are stopped.
② The interval timer is cleared to "0".
③ The status of the data memory, register and port latch are maintained as they were just before the hold operation, except that the SF is set to "1".
④ The PC holds two addresses ahead of the HOLD instruction. After hold operation is released, the instruction following the HOLD instruction is executed.

4.1.2 Release of Hold Operating Mode

The hold operation can be released to assume to normal operation by inputting the high level to the $\overline{HOLD}$ pin. Hold operation is released in the following sequence:

① Oscillation is started.
② Warming up allows sufficient time for oscillation to stabilize. During warm-up, internal operation stops. The warm-up time is 2^{13}/fc [sec].
③ When the warm-up time has elapsed, normal operation resumes with the instruction following the HOLD instruction.

Note : The warm-up time is obtained by dividing the basic clock with an interval timer; therefore, when there is fluctuation of the oscillation frequency when a hold operation is released, the warm-up time contains an error. Thus, it is necessary to set an approximate warm-up time value.

In addition, hold operation can also be released by setting the $\overline{RESET}$ pin to low level. In this case, the normal reset operation is performed immediately and normal operation resumes as soon as the reset mode is released; therefore, it is necessary to keep the $\overline{RESET}$ pin at low level and continue warm-up until oscillation stabilizes.

Note : The following cautions must be observed when releasing a hold operation with a low hold voltage. Before releasing a hold operation, it is necessary to increase the power supply voltage to the specified voltage range. The $\overline{RESET}$ pin is at high level at this time and increases together with the power supply voltage. If an external time constant circuit is connected at this time, the $\overline{RESET}$ pin input voltage will increase at a slower pace than the power supply voltage; therefore, if the $\overline{RESET}$ pin input voltage drops below the non-inverted high level input voltage of the $\overline{RESET}$ pin input (hysteresis input), there is danger of a reset occurring.

Hold operation will not start even if the Hold instruction is executed with the $\overline{HOLD}$ pin input at high level. Instead, the release (warm-up) sequence will start immediately. The warm-up time in this case is unstable and varies between 0 and 2^{13}/fc [sec]. Consequently, when the HOLD instruction is executed, it is necessary to check if the $\overline{HOLD}$ pin input is at low level (hold operation request).

Example: To test $\overline{HOLD}$ pin input to execute HOLD instruction

```
SWAIT  :  IN      %IP02,@HL      ; RAM [HL] ← IP02
          TEST    @HL,2          ; Waits until HOLD pin input goes low
          B       SHOLD
          B       SWAIT
SHOLD  :  HOLD                   ; Starts the Hold operation
```

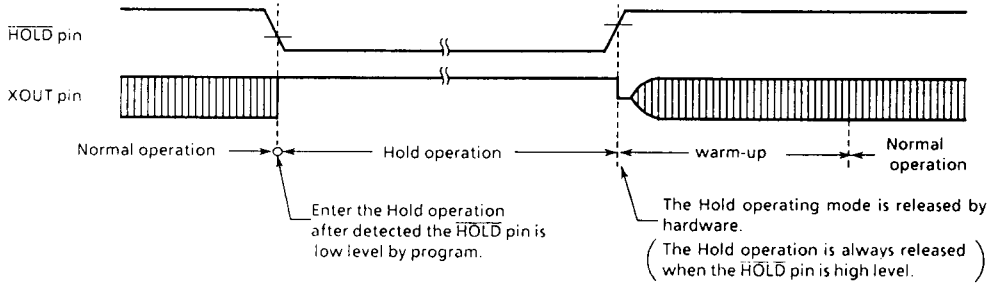


Figure 4-1. Hold operating mode

Note : Power consumption by the oscillation circuit and internal hardware is reduced during hold operation but power consumption by external circuit interfaces is not directly related to hold operation hardware operation; therefore, caution must be observed in system design and interface circuit design. For example, if the power supply voltage of an external circuit is dropped to 0V and the output is at high level, as shown in Figure 4-3, current may flow through the parasitic diodes of the external circuit in some cases. The relevant pin must be set to low level.

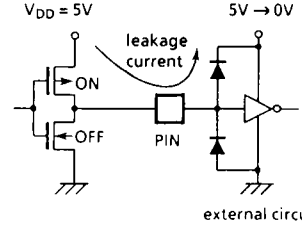


Figure 4-2. External circuit interface during the Hold operating mode

5. RESET FUNCTION

The internal status is initialized if a reset action is performed when the power supply voltage is within the operating voltage range and the $\overline{RESET}$ pin is held at low level for a minimum of 3 instruction cycles (15 basic clocks) while oscillation is stable.

When the $\overline{RESET}$ pin input goes high, the reset is released and program execution starts from address 000_H.

The $\overline{RESET}$ pin is hysteresis input with a pull-up resistor (typ.160kΩ) and simple power-on-reset can be done by

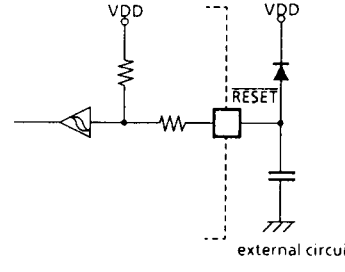


Figure 5-1. Simplified Power-On-Reset circuit

On-chip hardware	Initial value	On-chip hardware	Initial value
Program Counter (PC)	000_H	Output Latch (I/O port)	Refer to "I/O Circuitry"
Status Flag (SF)	1		
Port Register (PR)	0000_B		

Table 5-1. Initialization of Internal states by Reset action

ELECTRICAL CHARACTERISTICS

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

PARAMETER	SYMBOL	PIN	RATING	UNIT
Supply Voltage	V_{DD}		− 0.5 to 7	V
Input Voltage	V_{IN}		− 0.5 to V_{DD} + 0.5	V
Output Voltage	V_{OUT}		− 0.5 to V_{DD} + 0.5	V
Output Current (total)	ΣI_{OUT}		30	mA
Power Dissipation [T_{opr} = 85°C]	PD		300	mW
Soldering Temperature (time)	T_{sld}		260 (10sec)	°C
Storage Temperature	T_{stg}		− 55 to 125	°C
Operating Temperature	T_{opr}		− 40 to 85	°C

RECOMMENDED OPERATING CONDITIONS (V_{SS} = 0V, T_{opr} = − 40 to 85°C)

PARAMETER	SYMBOL	PIN	CONDITIONS	Min.	Max.	UNIT
Supply Voltage	V_{DD}			4.0	6.0	V
Input High Voltage	V_{IH1}	Except Hysteresis Input		V_{DD} × 0.7	V_{DD}	V
	V_{IH2}	Hysteresis Input		V_{DD} × 0.8		
Input Low Voltage	V_{IL1}	Except Hysteresis Input		0	V_{DD} × 0.3	V
	V_{IL2}	Hysteresis Input			V_{DD} × 0.2	
Clock Frequency	f_C		V_{DD} = 4.0 to 6.0V	0.2	2.0	MHz
			V_{DD} = 4.5 to 6.0V		5.0	

D.C. CHARACTERISTICS (V_{SS} = 0V, T_{opr} = −40 to 85°C)

PARAMETER	SYMBOL	PIN	CONDITIONS	Min.	Typ.	Max.	UNIT
Hysteresis Voltage	V_{HS}	Hysteresis Voltage		—	0.3	—	V
Input High Current	I_{IH}	Ports P0, P1, P2	V_{DD} = 5V, V_{IN} = 5V	15	30	80	μA
Input Resistance	R_{IN}	$\overline{RESET}$		65	160	340	KΩ
Output High Voltage	V_{OH}	Ports P0, P1, P2	V_{DD} = 5V, I_{OH} = −5μA	4.7	4.9	—	V
Output High Current	I_{OH}		V_{DD} = 4V, V_{OH} = 2.4V	−1.0	−4.5	—	mA
Output Low Current	I_{OL}		V_{DD} = 4.5V, V_{OL} = 0.4V	1.6	5.0	—	mA
Supply Current (In the Normal operating mode)	I_{DD}		V_{DD} = 6V, fc = 2MHz	—	0.8	3.0	mA
Supply Current (In the HOLD operating mode)	I_{DDH}		V_{DD} = 6V	—	0.1	5.0	μA

Note 1. Typ. values shows those at V_{DD} = 5V, T_{opr} = 25°C.

Note 2. Supply Current in the Normal operating mode : $\overline{RESET}$ pin is 0V, and XOUT pin and ports are opened in the external clock operation.

Note 3. Supply Current in the Hold operating mode : All pins except the power supply pins (VDD, VSS) are opened.

A.C. CHARACTERISTICS (V_{SS} = 0V, T_{opr} = −40 to 85°C)

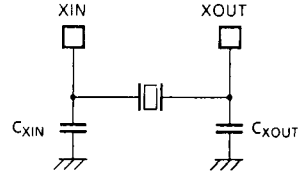
PARAMETER	SYMBOL	CONDITIONS		Min.	Typ.	Max.	UNIT
Instruction Cycle Time	t_{cy}	V_{DD} = 4.0 to 6.0V		2.5	—	25	μs
		V_{DD} = 4.5 to 6.0V		1.0			
High level Clock pulse Width	t_{WCH}	V_{IN} = V_{IH}	For external clock operation	100	—	—	ns
Low level Clock pulse Width	t_{WCL}	V_{IN} = V_{IL}					

RECOMMENDED (V_{SS} = 0V, V_{DD} = 4.0 to 6.0 V, T_{opr} = − 40 to 85℃)

(1) Ceramic Resonator

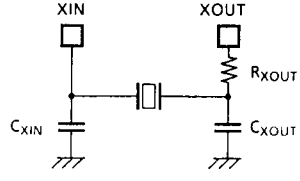
5MHz (V_{DD} = 4.5 to 6.0V)
CSA5.00MG (MURATA) $C_{XIN} = C_{XOUT}$ = 30pF

2MHz
CSA2.00MG (MURATA) $C_{XIN} = C_{XOUT}$ = 100pF



1MHz
CSB1000D (MURATA) $C_{XIN} = C_{XOUT}$ = 330pF

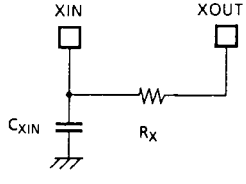
455KHz
CSB455E (MURATA) $C_{XIN} = C_{XOUT}$ = 220pF, R_{XOUT} = 5.6KΩ



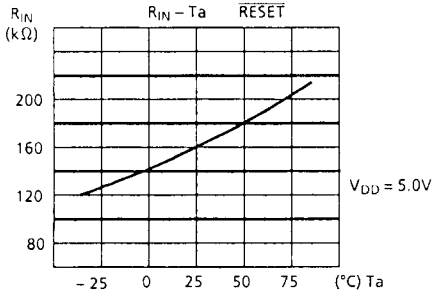
(2) RC oscillation

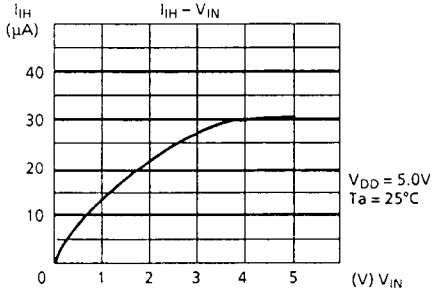
1MHz typ. R_X = 15KΩ, C_{XIN} = 100pF

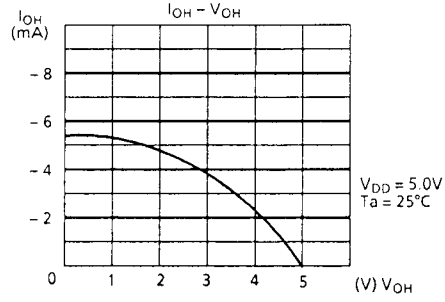
500KHz typ. R_X = 36KΩ, C_{XIN} = 100pF

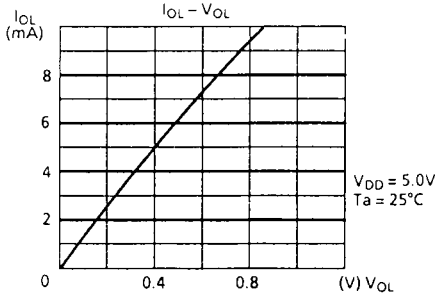


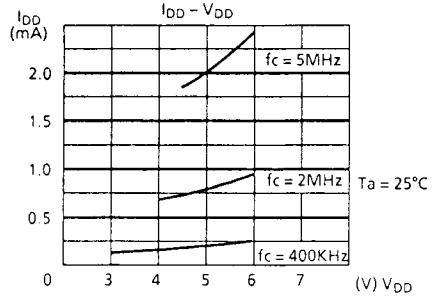
TYPICAL CHARACTERISTICS

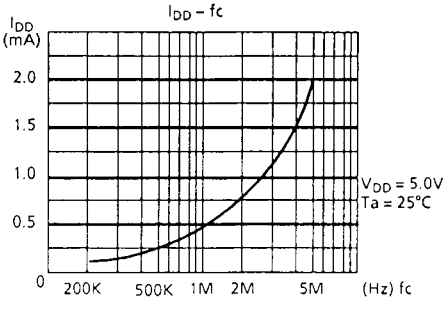


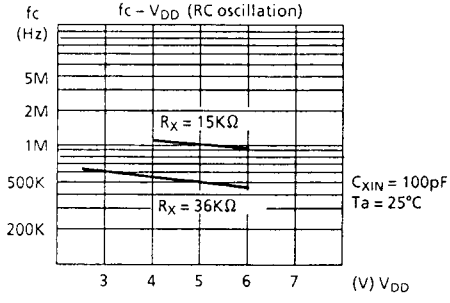


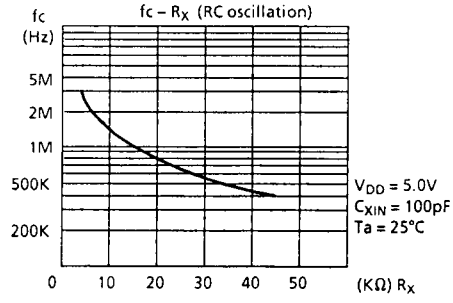


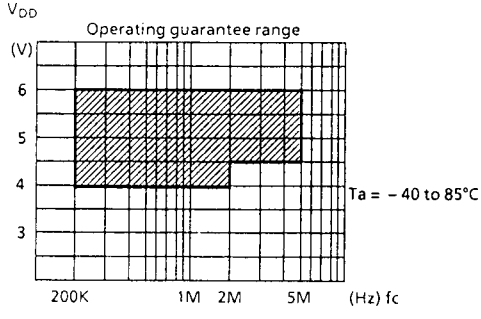







fc
(Hz)
fc – VDD (RC oscillation)
5M
2M
1M
500K
200K
RX = 15KΩ
RX = 36KΩ
CXIN = 100pF
Ta = 25°C
3
4
5
6
7
(V) VDD


fc
(Hz)
fc – RX (RC oscillation)
5M
2M
1M
500K
200K
VDD = 5.0V
CXIN = 100pF
Ta = 25°C
0
10
20
30
40
50
(KΩ) RX

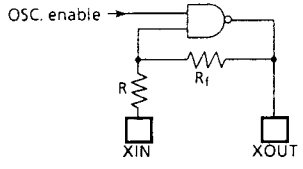
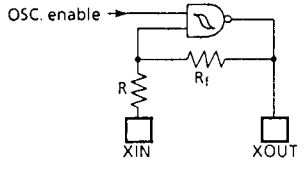
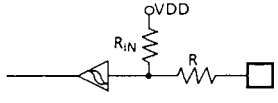

VDD
(V)
Operating guarantee range
6
5
4
3
Ta = – 40 to 85°C
200K
1M
2M
5M
(Hz) fc

INPUT/OUTPUT CIRCUITRY

(1) Control pins

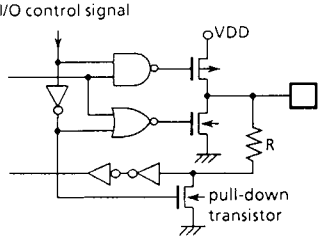
Input/Output circuitries of the 42C40/60 control pins are shown below.

Oscillation type can be chosen by mask option either a ceramic resonator (external clock input) or an RC oscillator.

CONTROL PIN	I/O	CIRCUITRY	REMARKS
XIN XOUT	INPUT OUTPUT	Option : Ceramic resonator or External clock input 	Resonator connecting pins R = 1KΩ typ. R_f = 1MΩ typ.
		Option : RC oscillation 	Resonator connecting pins Hysteresis input R = 1KΩ typ. R_f = 1MΩ typ.
RESET	INPUT		Hysteresis input Pull-up resistor R_{IN} = 160KΩ typ. R = 1KΩ typ.

(2) I/O ports

All I/O ports of 42C40/60 are programmable I/O. I/O mode can be specified by [MOV A,P] instruction.

PORT	I/O	CIRCUITRY	INITIAL STATE	REMARKS
P0 P1 P2	I/O		INPUT	Programmable I/O port Input mode: Input with pull-down transistor Output mode: push-pull R = 1KΩ typ.