

# HD6303Y, HD63A03Y, HD63B03Y, HD63C03Y CMOS MPU (Micro Processing Unit)

The HD6303Y is a CMOS 8-bit single-chip microprocessing unit which contains a CPU compatible with the CMOS 8-bit microcomputer HD6301V, 256 bytes of RAM, 24 parallel I/O pins, Serial Communication Interface (SCI) and two timers.

## ■ FEATURES

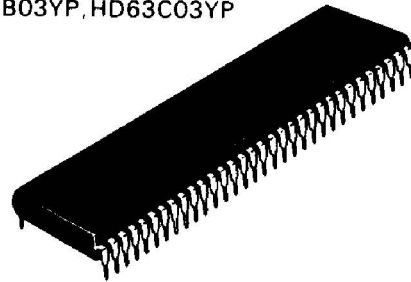
- Instruction Set Compatible with the HD6301V1
- 256 Bytes of RAM
- 24 Parallel I/O Pins
- Parallel Handshake Interface (Port 6)
- Darlington Transistor Drive (Port 2, 6)
- 16-Bit Programmable Timer
  - Input Capture Register × 1
  - Free Running Counter × 1
  - Output Compare Register × 2
- 8-Bit Reloadable Timer
  - External Event Counter
  - Square Wave Generation
- Serial Communication Interface (SCI)
  - Asynchronous Mode (8 Transmit Formats, Hardware Parity)
  - Clocked Synchronous Mode
- Memory Ready
  - 3 Kinds of Memory Ready
- Halt
- Error Detection (Address Error, Op-code Error)
- Interrupt – External 3, Internal 7
- Maximum 65k Bytes Address Space
- Low Power Dissipation Mode
  - Sleep Mode
  - Standby Mode (Hardware Standby, Software Standby)
- Minimum Instruction Execution Time –  $0.5\mu\text{s}$  ( $f = 2\text{MHz}$ )
- Wide Range of Operation
 

|                                      |   |  |
|--------------------------------------|---|--|
| $V_{CC} = 3 \text{ to } 5.5\text{V}$ | ( $f = 0.1 \text{ to } 0.5\text{MHz}$ ) |  |
| $V_{CC} = 5\text{V} \pm 10\%$        | }                                       | $f = 0.1 \text{ to } 1.0\text{MHz}$ : HD6303Y  |
|                                      |   | $f = 0.1 \text{ to } 1.5\text{MHz}$ : HD63A03Y |
|                                      |   | $f = 0.1 \text{ to } 2.0\text{MHz}$ : HD63B03Y |
|                                      |   | $f = 0.1 \text{ to } 3.0\text{MHz}$ : HD63C03Y |

## ■ PROGRAM DEVELOPMENT SUPPORT TOOLS

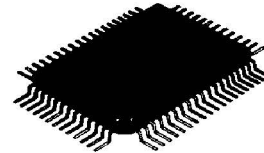
- Cross assembler and C compiler software for IBM PCs and compatibles
- In circuit emulator for use with IBM PCs and compatibles

HD6303YP, HD63A03YP,  
HD63B03YP, HD63C03YP



(DP-64S)

HD6303YF, HD63A03YF,  
HD63B03YF, HD63C03YF



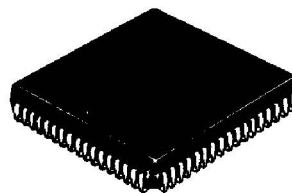
(FP-64)

HD6303YH, HD63A03YH,  
HD63B03YH, HD63C03YH



(FP-64A)

HD6303YCP, HD63A03YCP,  
HD63B03YCP, HD63C03YCP

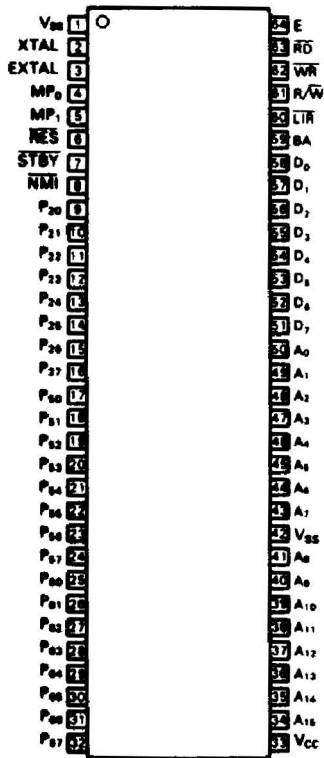


(CP-68)



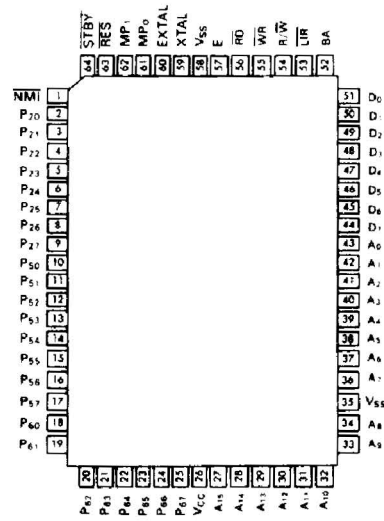
■ PIN ARRANGEMENT

- HD6303YP, HD63A03YP, HD63B03YP, HD63C03YP



(Top View)

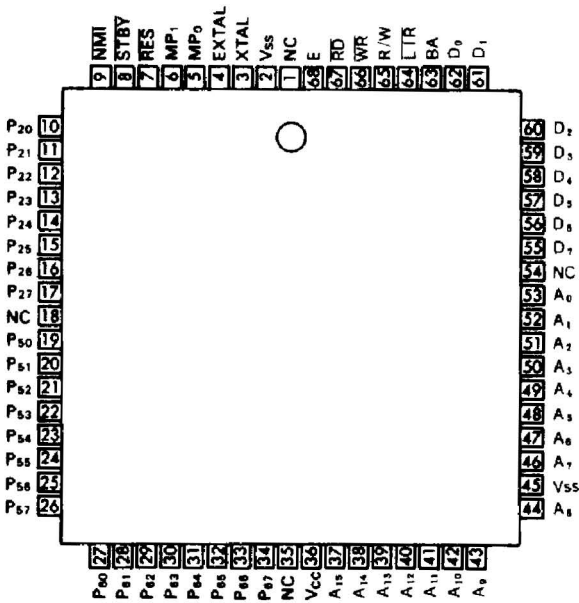
- HD6303YF, HD63A03YF, HD63B03YF, HD63C03YF



(Top View)

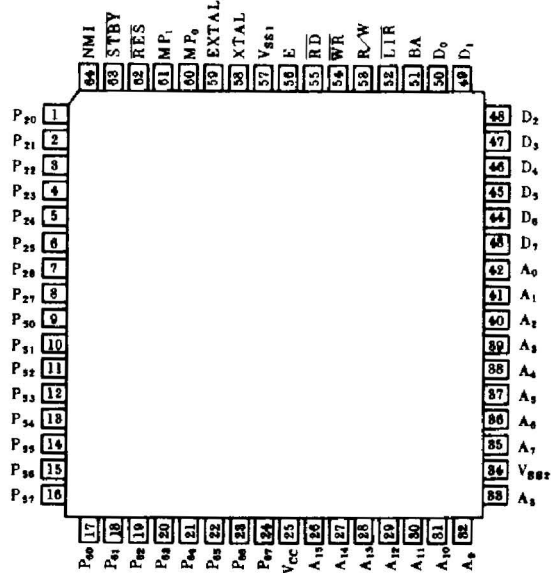


- HD6303YCP, HD63A03YCP, HD63B03YCP, HD63C03YCP



(Top View)

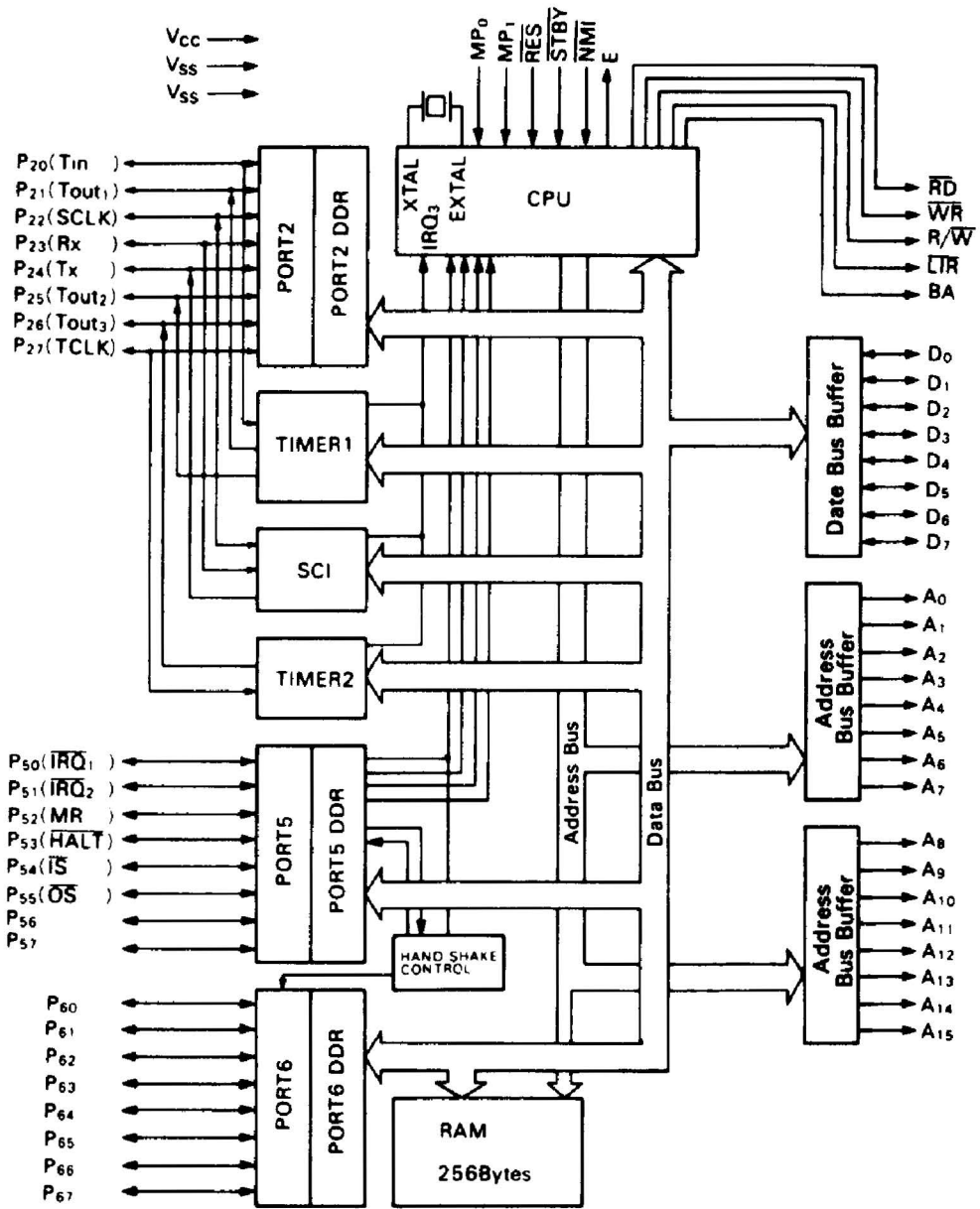
- HD6303YH, HD63A03YH, HD63B03YH, HD63C03YH



(Top View)



■ BLOCK DIAGRAM



## ■ ABSOLUTE MAXIMUM RATINGS

| Item                  | Symbol    | Value                 | Unit |
|-----------------------|-----------|-----------------------|------|
| Supply Voltage        | $V_{CC}$  | -0.3 ~ +7.0           | V    |
| Input Voltage         | $V_{in}$  | -0.3 ~ $V_{CC} + 0.3$ | V    |
| Operating Temperature | $T_{opr}$ | -20 ~ +75             | °C   |
| Storage Temperature   | $T_{stg}$ | -55 ~ +150            | °C   |

(NOTE) This product has protection circuits in input terminal from high static electricity voltage and high electric field. But be careful not to apply overvoltage more than maximum ratings to these high input impedance protection circuits. To assure the normal operation, we recommend  $V_{in}, V_{out}: V_{SS} \leq (V_{in} \text{ or } V_{out}) \leq V_{CC}$ .

## ■ ELECTRICAL CHARACTERISTICS

● DC CHARACTERISTICS ( $V_{CC} = 5.0V \pm 10\%$ ,  $V_{SS} = 0V$ ,  $T_a = -20^\circ C \sim +75^\circ C$ , unless otherwise noted.)

| Item                        | Symbol  | Test Condition       | min                                       | typ                 | max                    | Unit |     |     |    |
|-----------------------------|---|----------------------|---|---------------------|------------------------|------|-----|-----|----|
| Input "High" Voltage        | RES, STBY   |                      | $V_{CC} - 0.5$                            | -                   | $V_{CC} + 0.3$         | V    |     |     |    |
|                             | EXTAL   |                      | $V_{CC} \times 0.7$                       | -                   |                        |      |     |     |    |
|                             | Other Inputs  |                      | 2.0                                       | -                   |                        |      |     |     |    |
| Input "Low" Voltage         | All Inputs  | $V_{IL}$             | -0.3                                      | -                   | 0.8***                 | V    |     |     |    |
| Input Leakage Current       | NMI, RES, STBY, MP <sub>0</sub> , MP <sub>1</sub>   | $ I_{in} $           | $V_{in} = 0.5 \sim V_{CC} - 0.5V$         | -                   | -                      | 1.0  | μA  |     |    |
| Three State Leakage Current | A <sub>0</sub> ~A <sub>15</sub> , D <sub>0</sub> ~D <sub>7</sub> , RD, WR, R/W, Ports 2, 5, 6 | $ I_{TSI} $          | $V_{in} = 0.5 \sim V_{CC} - 0.5V$         | -                   | -                      | 1.0  | μA  |     |    |
| Output "High" Voltage       | All Outputs   | $V_{OH}$             | $I_{OH} = -200\mu A$                      | 2.4                 | -                      | -    | V   |     |    |
|                             |   |                      | $I_{OH} = -10\mu A$                       | $V_{CC} - 0.7$      | -                      | -    | V   |     |    |
| Output "Low" Voltage        | All Outputs   | $V_{OL}$             | $I_{OL} = 1.6mA$                          | -                   | -                      | 0.4  | V   |     |    |
| Darlington Drive Current    | Ports 2, 6  | $-I_{OH}$            | $V_{out} = 1.5V$                          | 1.0                 | -                      | 10.0 | mA  |     |    |
| Input Capacitance           | All Inputs  | $C_{in}$             | $V_{in} = 0V, f = 1MHz, T_a = 25^\circ C$ | -                   | -                      | 12.5 | pF  |     |    |
| Current Dissipation*        | $I_{STB}$   | Non Operation        |   | -                   | 3.0                    | 15.0 | μA  |     |    |
|                             |   |                      | $I_{SLP}$                                 | Sleeping (f=1MHz**) | -                      | 1.5  | 3.0 | mA  |    |
|                             |   |                      |   |                     | Sleeping (f=1.5MHz**)  | -    | 2.3 | 4.5 | mA |
|                             |   |                      |   |                     | Sleeping (f=2MHz**)*** | -    | 3.0 | 6.0 | mA |
|                             | Sleeping (f=3 MHz)  | -                    |   |                     | 4.5                    | 9.0  | mA  |     |    |
|                             | $I_{CC}$  | Operating (f=1MHz**) | -   | 7.0                 | 10.0                   | mA   |     |     |    |
|                             |   |                      | Operating (f=1.5MHz**)                    | -                   | 10.5                   | 15.0 | mA  |     |    |
|                             |   |                      | Operating (f=2MHz**)***                   | -                   | 14.0                   | 20.0 | mA  |     |    |
| Operating (f=3 MHz)         |   |                      | -   | 21.0                | 30.0                   | mA   |     |     |    |
| RAM Standby Voltage         | $V_{RAM}$   |                      | 2.0                                       | -                   | -                      | V    |     |     |    |

\*  $V_{IH} \text{ min} = V_{CC} - 1.0V, V_{IL} \text{ max} = 0.8V$  (All output terminals are at no load.)

\*\* Current Dissipation of the operating or sleeping condition is proportional to the operating frequency. So the typ. or max. values about Current

Dissipations at X MHz operation are decided according to the following formula:

typ. value (f = X MHz) = typ. value (f = 1MHz) × X

max. value (f = X MHz) = max. value (f = 1MHz) × X

(both the sleeping and operating)

\*\*\* SCLK 0.6V (-20°C~0°C)

2



# HD6303Y, HD63A03Y, HD63B03Y, HD63C03Y

● AC CHARACTERISTICS ( $V_{CC} = 5.0V \pm 10\%$ ,  $V_{SS} = 0V$ ,  $T_a = -20 \sim +75^\circ C$ , unless otherwise noted.)

## BUS TIMING

| Item                                       | Symbol           | Test Condition | HD6301Y0 |      |      | HD63A01Y0 |      |      | HD63B01Y0 |      |      | HD63C01Y0 |      |      | Unit      |
|--|------------------|----------------|----------|------|------|-----------|------|------|-----------|------|------|-----------|------|------|-----------|
|  |                  |                | min.     | typ. | max. | min.      | typ. | max. | min.      | typ. | max. | min.      | typ. | max. |           |
| Cycle Time                                 | $t_{cyc}$        |                | 1        | —    | 10   | 0.666     | —    | 10   | 0.5       | —    | 10   | 0.333     | —    | 10   | $\mu s$   |
| Enable Rise Time                           | $t_{Er}$         |                | —        | —    | 25   | —         | —    | 25   | —         | —    | 25   | —         | —    | 20   | ns        |
| Enable Fall Time                           | $t_{Ef}$         |                | —        | —    | 25   | —         | —    | 25   | —         | —    | 25   | —         | —    | 20   | ns        |
| Enable Pulse Width "High" Level*           | $PW_{EH}$        |                | 450      | —    | —    | 300       | —    | —    | 220       | —    | —    | 140       | —    | —    | ns        |
| Enable Pulse Width "Low" Level*            | $PW_{EL}$        |                | 450      | —    | —    | 300       | —    | —    | 220       | —    | —    | 140       | —    | —    | ns        |
| Address, $R/\bar{W}$ Delay Time*           | $t_{AD}$         |                | —        | —    | 250  | —         | —    | 190  | —         | —    | 160  | —         | —    | 120  | ns        |
| Data Delay Time                            | Write $t_{DDW}$  |                | —        | —    | 200  | —         | —    | 160  | —         | —    | 120  | —         | —    | 100  | ns        |
| Data Set-up Time                           | Read $t_{DSR}$   |                | 80       | —    | —    | 70        | —    | —    | 60        | —    | —    | 50        | —    | —    | ns        |
| Address, $R/\bar{W}$ Hold Time*            | $t_{AH1}$        |                | 80       | —    | —    | 50        | —    | —    | 40        | —    | —    | 20        | —    | —    | ns        |
| Data Hold Time                             | Write* $t_{HW1}$ | Fig. 1         | 80       | —    | —    | 50        | —    | —    | 40        | —    | —    | 20        | —    | —    | ns        |
| $\bar{RD}$ , $\bar{WR}$ Address Hold Time* | $t_{AH2}$        |                | 70       | —    | —    | 50        | —    | —    | 40        | —    | —    | 20        | —    | —    | ns        |
| $\bar{RD}$ , $\bar{WR}$ Data Hold Time*    | $t_{HW2}$        |                | 70       | —    | —    | 50        | —    | —    | 40        | —    | —    | 20        | —    | —    | ns        |
| Data Hold Time                             | Read $t_{HR}$    |                | 0        | —    | —    | 0         | —    | —    | 0         | —    | —    | 0         | —    | —    | ns        |
| $\bar{RD}$ , $\bar{WR}$ Pulse Width*       | $PW_{RW}$        |                | 450      | —    | —    | 300       | —    | —    | 220       | —    | —    | 140       | —    | —    | ns        |
| $\bar{RD}$ , $\bar{WR}$ Delay Time         | $t_{RWD}$        |                | —        | —    | 40   | —         | —    | 40   | —         | —    | 40   | —         | —    | 40   | ns        |
| $\bar{RD}$ , $\bar{WR}$ Hold Time          | $t_{HRW}$        |                | —        | —    | 20   | —         | —    | 20   | —         | —    | 20   | —         | —    | 20   | ns        |
| $\bar{LIR}$ Delay Time                     | $t_{DLR}$        |                | —        | —    | 200  | —         | —    | 160  | —         | —    | 120  | —         | —    | 80   | ns        |
| $\bar{LIR}$ Hold Time                      | $t_{HLR}$        |                | 10       | —    | —    | 10        | —    | —    | 10        | —    | —    | 5         | —    | —    | ns        |
| Peripheral Read Access Time                | $t_{ACC}$        |                | —        | —    | —    | —         | —    | —    | —         | —    | —    | 180       | —    | —    | ns        |
| MR Set-up Time*                            | $t_{SLR}$        | Fig. 2         | 400      | —    | —    | 280       | —    | —    | 230       | —    | —    | 170       | —    | —    | ns        |
| MR Hold Time*                              | $t_{HMR}$        |                | —        | —    | 100  | —         | —    | 70   | —         | —    | 50   | —         | —    | 25   | ns        |
| E Clock Pulse Width at MR                  | $PW_{EMR}$       |                | —        | —    | 9    | —         | —    | 9    | —         | —    | 9    | —         | —    | 9    | $\mu s$   |
| Processor Control Set-up Time              | $t_{PCS}$        | Fig. 3, 13, 14 | 200      | —    | —    | 200       | —    | —    | 200       | —    | —    | 100       | —    | —    | ns        |
| Processor Control Rise Time                | $t_{PCr}$        | Fig. 2, 3      | —        | —    | 100  | —         | —    | 100  | —         | —    | 100  | —         | —    | 50   | ns        |
| Processor Control Fall Time                | $t_{PCf}$        |                | —        | —    | 100  | —         | —    | 100  | —         | —    | 100  | —         | —    | 50   | ns        |
| BA Delay Time                              | $t_{BA}$         | Fig. 3         | —        | —    | 250  | —         | —    | 190  | —         | —    | 160  | —         | —    | 120  | ns        |
| Oscillator Stabilization Time              | $t_{RC}$         | Fig. 14        | 20       | —    | —    | 20        | —    | —    | 20        | —    | —    | 20        | —    | —    | ms        |
| Reset Pulse Width                          | $PW_{RST}$       |                | 3        | —    | —    | 3         | —    | —    | 3         | —    | —    | 3         | —    | —    | $t_{cyc}$ |

\*These timings change in approximate proportion to  $t_{cyc}$ . The figures in this characteristics represent those when  $t_{cyc}$  is minimum (= in the highest speed operation).

## Peripheral Port Timing

| Item  | Symbol                | Test condition | HD6303Y |     |     | HD63A03Y |     |     | HD63B03Y |     |     | HD63C03Y |     |     | Unit |
|---|-----------------------|----------------|---------|-----|-----|----------|-----|-----|----------|-----|-----|----------|-----|-----|------|
|   |                       |                | min     | typ | max | min      | typ | max | min      | typ | max | min      | typ | max |      |
| Peripheral Data Set Up Time                             | Port 2,5,6 $t_{pDSU}$ | Fig. 5         | 200     | —   | —   | 200      | —   | —   | 200      | —   | —   | 200      | —   | —   | ns   |
| Peripheral Data Hold Time                               | Port 2,5,6 $t_{pDH}$  |                | 200     | —   | —   | 200      | —   | —   | 200      | —   | —   | 200      | —   | —   | ns   |
| Delay Time (From Enable Fall Edge to Peripheral Output) | Port 2,5,6 $t_{pWD}$  | Fig. 6         | —       | —   | 300 | —        | —   | 300 | —        | —   | 300 | —        | —   | 300 | ns   |
| Input Strobe Pulse Width                                | $t_{pWIS}$            |                | 200     | —   | —   | 200      | —   | —   | 200      | —   | —   | 200      | —   | —   | ns   |
| Input Data Hold Time                                    | Port 6 $t_{IH}$       | Fig.10         | 150     | —   | —   | 150      | —   | —   | 150      | —   | —   | 150      | —   | —   | ns   |
| Input Data Set-Up Time                                  | Port 6 $t_{IS}$       |                | 100     | —   | —   | 100      | —   | —   | 100      | —   | —   | 100      | —   | —   | ns   |
| Output Strobe Delay Time                                | $t_{DSD1}$            | Fig.11         | —       | —   | 200 | —        | —   | 200 | —        | —   | 200 | —        | —   | 200 | ns   |
|   | $t_{DSD2}$            |                | —       | —   | 200 | —        | —   | 200 | —        | —   | 200 | —        | —   | 200 | ns   |



# HD6303Y, HD63A03Y, HD63B03Y, HD63C03Y

## TIMER, SCI TIMING

| Item  | Symbol       | Test Condition | HD6303Y |      |      | HD63A03Y |      |      | HD63B03Y |      |      | HD63C03Y |      |      | Unit       |
|---|--------------|----------------|---------|------|------|----------|------|------|----------|------|------|----------|------|------|------------|
|   |              |                | min.    | typ. | max. | min.     | typ. | max. | min.     | typ. | max. | min.     | typ. | max. |            |
| Timer 1 Input Pulse Width                               | $t_{PWT}$    | Fig. 9         | 2.0     | —    | —    | 2.0      | —    | —    | 2.0      | —    | —    | 2.0      | —    | —    | $t_{cyc}$  |
| Delay Time (Enable Positive Transition to Timer Output) | $t_{TOD}$    | Fig. 7, 8      | —       | —    | 400  | —        | —    | 400  | —        | —    | 400  | —        | —    | 400  | ns         |
| SCI Input Clock Cycle                                   | Async. Mode  | Fig. 9         | 1.0     | —    | —    | 1.0      | —    | —    | 1.0      | —    | —    | 1.0      | —    | —    | $t_{cyc}$  |
|   | Clock Sync.  | Fig. 4         | 2.0     | —    | —    | 2.0      | —    | —    | 2.0      | —    | —    | 2.0      | —    | —    | $t_{cyc}$  |
| SCI Transmit Data Delay Time (Clock Sync. Mode)         | $t_{TXD}$    |                | —       | —    | 220  | —        | —    | 220  | —        | —    | 220  | —        | —    | 220  | ns         |
| SCI Receive Data Set-up Time (Clock Sync. Mode)         | $t_{SRX}$    | Fig. 4         | 260     | —    | —    | 260      | —    | —    | 260      | —    | —    | 260      | —    | —    | ns         |
| SCI Receive Data Hold Time (Clock Sync. Mode)           | $t_{HRX}$    |                | 100     | —    | —    | 100      | —    | —    | 100      | —    | —    | 100      | —    | —    | ns         |
| SCI Input Clock Pulse Width                             | $t_{PWSCK}$  | Fig. 9         | 0.4     | —    | 0.6  | 0.4      | —    | 0.6  | 0.4      | —    | 0.6  | 0.4      | —    | 0.6  | $t_{Scyc}$ |
| Timer 2 Input Clock Cycle                               | $t_{cyc}$    |                | 2.0     | —    | —    | 2.0      | —    | —    | 2.0      | —    | —    | 2.0      | —    | —    | $t_{cyc}$  |
| Timer 2 Input Clock Pulse Width                         | $t_{PWTCCK}$ |                | 200     | —    | —    | 200      | —    | —    | 200      | —    | —    | 200      | —    | —    | ns         |
| Timer 1-2, SCI Input Clock Rise Time                    | $t_{CKr}$    |                | —       | —    | 100  | —        | —    | 100  | —        | —    | 100  | —        | —    | 50   | ns         |
| Timer 1-2, SCI Input Clock Fall Time                    | $t_{CKf}$    |                | —       | —    | 100  | —        | —    | 100  | —        | —    | 100  | —        | —    | 50   | ns         |

2



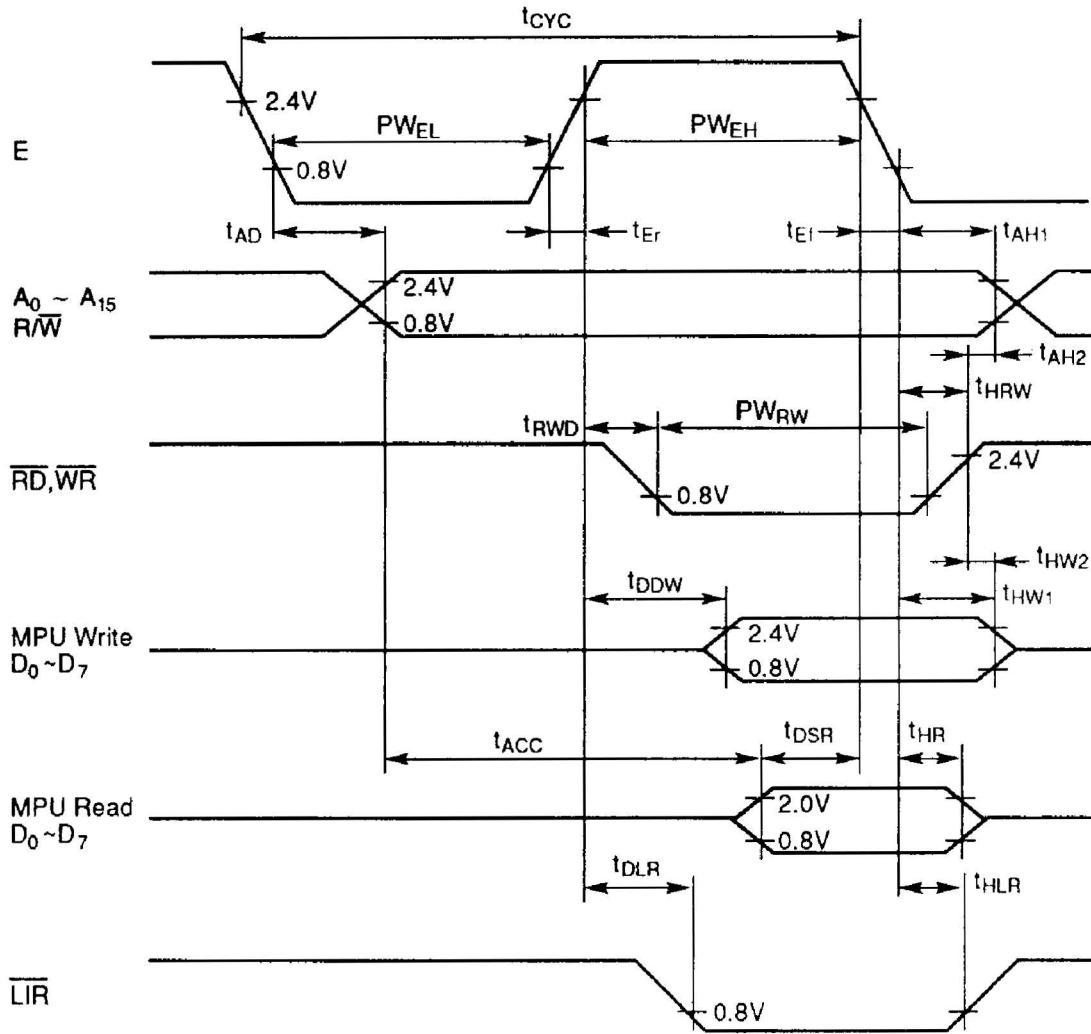


Figure 1 Bus Timing

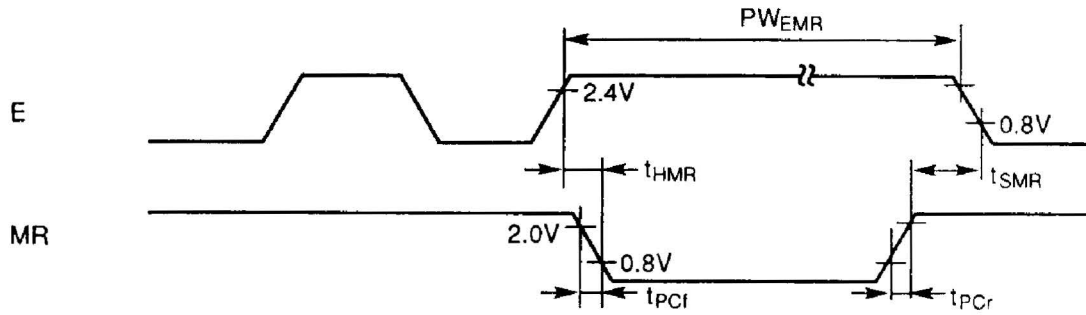


Figure 2 Memory Ready and E Clock Timing



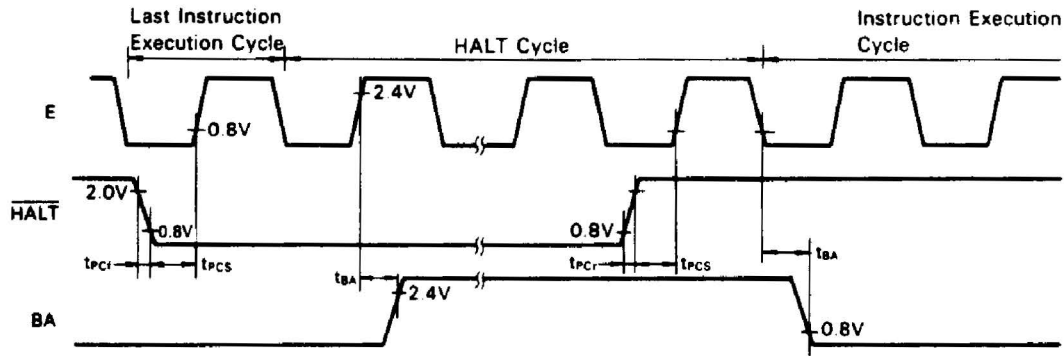


Figure 3 HALT and BA Timing

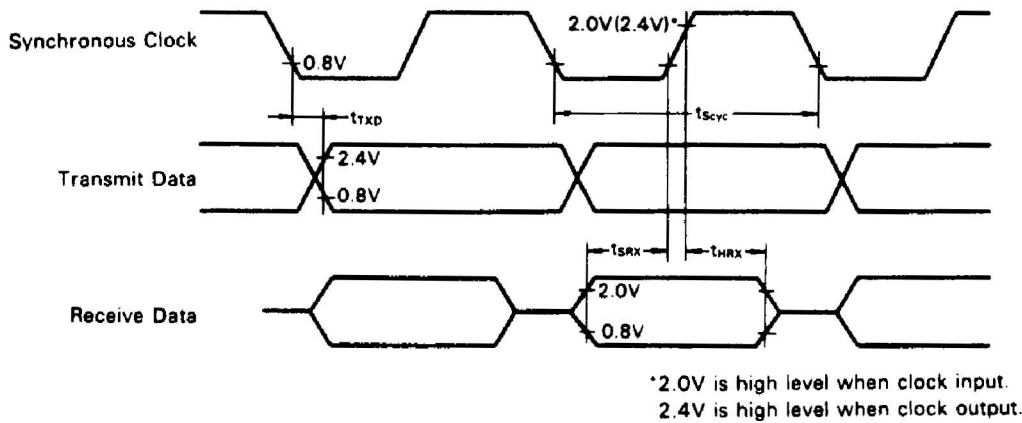


Figure 4 SCI Clocked Synchronous Timing

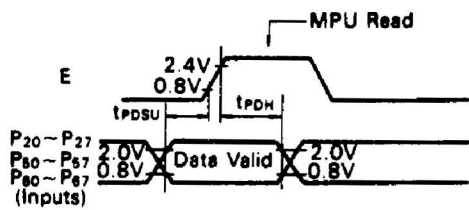


Figure 5 Port Data Set-up and Hold Times (MPU Read)

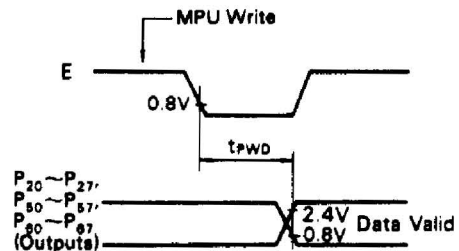


Figure 6 Port Data Delay Times (MPU Write)





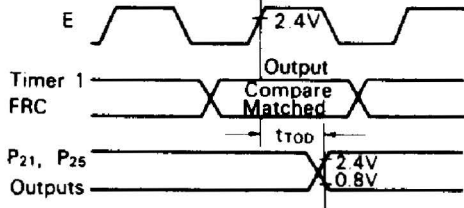


Figure 7 Timer 1 Output Timing

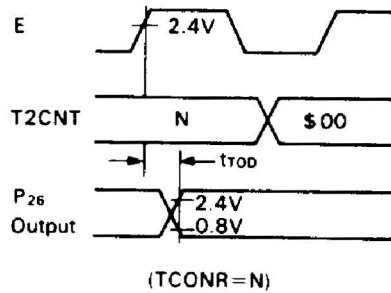


Figure 8 Timer 2 Output Timing

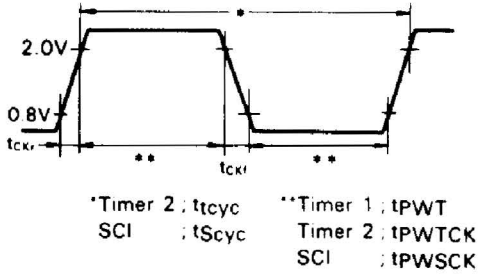


Figure 9 Timer 1-2, SCI Input Clock Timing

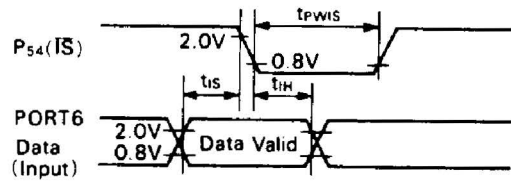


Figure 10 Port 6 Input Latch Timing

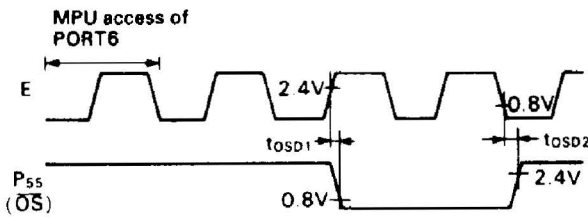


Figure 11 Output Strobe Timing

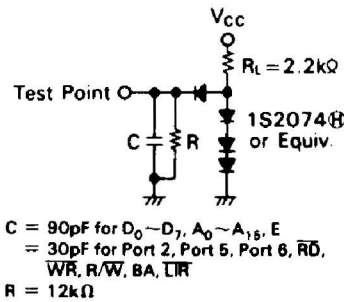


Figure 12 Bus Timing Test Loads (TTL Load)

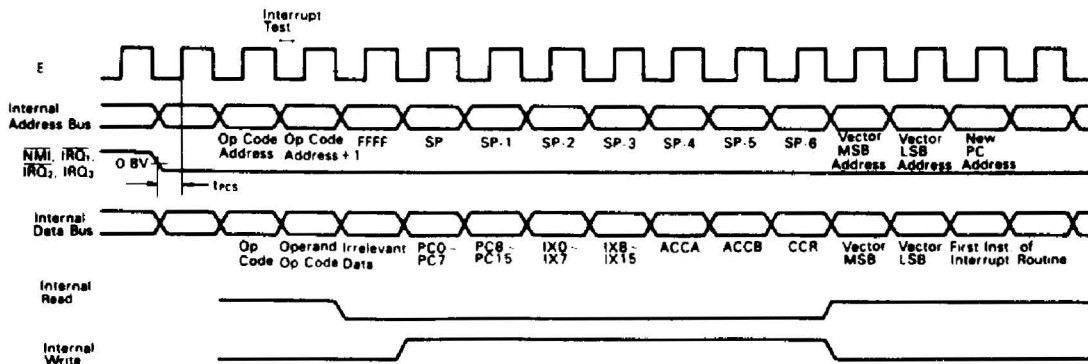


Figure 13 Interrupt Sequence



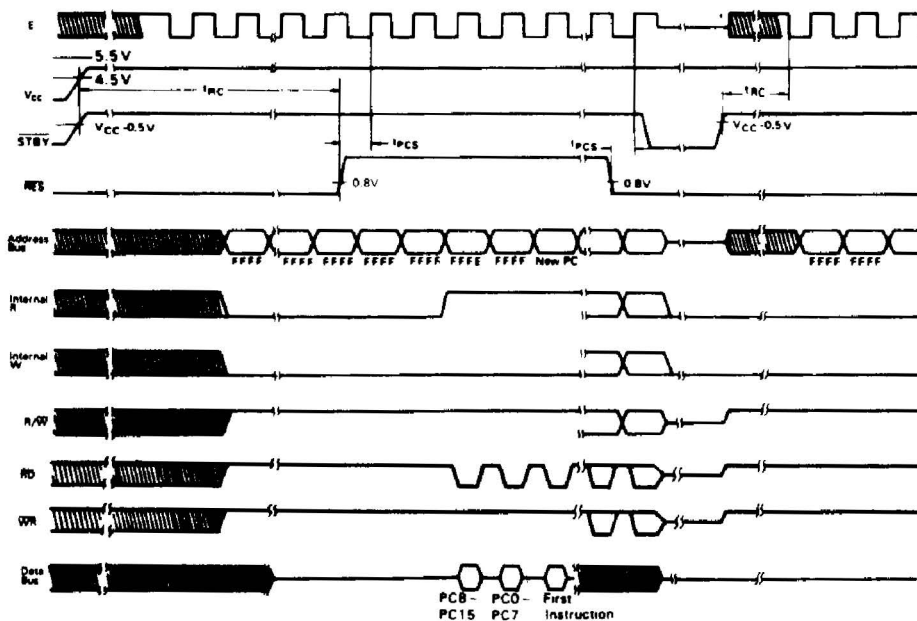


Figure 14 Reset Timing

FUNCTIONAL PIN DESCRIPTION

**V<sub>CC</sub>, V<sub>SS</sub>**  
 V<sub>CC</sub> and V<sub>SS</sub> provide power to the MPU with 5V ± 10% supply. In the case of low speed operation (f<sub>max</sub> = 500kHz), the MPU can operate with 3 to 5.5 volts. Two V<sub>SS</sub> pins should be tied to ground.

**XTAL, EXTAL**

These two pins interface with an AT-cut parallel resonant crystal. Divide-by-four circuit is on chip, so if 4MHz crystal oscillator is used, the system clock is 1MHz for example.

EXTAL pin can be driven by the external clock with 45% to 55% duty. The system clock which is one fourth frequency of the external clock is generated in the LSI. The external clock frequency should be less than four times of the maximum operating frequency. When using the external clock, XTAL pin should be open. Fig. 15 shows examples of connection circuit. The crystal and C<sub>L1</sub>, C<sub>L2</sub> should be mounted as close as possible to XTAL and EXTAL pins. Any line must not cross the line between the crystal oscillator and XTAL, EXTAL.

AT Cut Parallel Resonant Crystal Oscillator

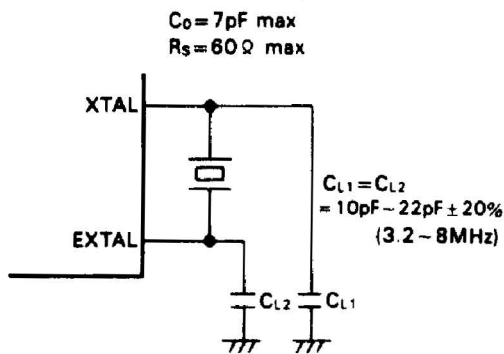


Figure 15 Connection Circuit

**STBY**

This pin makes the MPU standby mode. In "Low" level, the oscillation stops and the internal clock is stabilized to make reset condition. To retain the contents of RAM at standby mode, "0" should be written into RAM enable bit (RAME). RAME is the bit 6 of the RAM/port 5 control register at \$0014. RAM is disabled by this operation and its contents is sustained.

Refer to "LOW POWER DISSIPATION MODE" for the standby mode.

**Reset (RES)**

This pin resets the MPU from power OFF state and provides a startup procedure. During power-on, RES pin must be held "Low" level for at least 20ms.

The CPU registers (accumulator, index register, stack pointer, condition code register except for interrupt mask bit), RAM and the data register of ports are not initialized during reset, so their contents are undefined in this procedure.

To reset the MPU during operation, RES should be held "Low" for at least 3 system-clock cycles. At the 3rd cycle during "Low" level, all the address buses become "High". When RES remains "Low", the address buses keep "High". If RES becomes "High", the MPU starts the next operation.

- (1) Latch the value of the mode program pins; MP<sub>0</sub> and MP<sub>1</sub>.
- (2) Initialize each internal register (Refer to Table 4).
- (3) Set the interrupt mask bit. For the CPU to recognize the maskable interrupts  $\overline{IRQ}_1$ ,  $\overline{IRQ}_2$  and  $\overline{IRQ}_3$ , this bit should be cleared in advance.
- (4) Put the contents (=start address) of the last two addresses (\$FFFE, \$FFFF) into the program counter and start the program from this address. (Refer to Table 1).

**Enable (E)**

This pin provides a TTL-compatible system clock to external circuits. Its frequency is one fourth that of the crystal oscillator or external clock. This pin can drive one TTL load and 90pF capacitance.

**Non-Maskable Interrupt (NMI)**

When the falling edge of the input signal is detected at this pin, the CPU begins non-maskable interrupt sequence internally. As



well as the  $\overline{IRQ}$  mentioned below, the instruction being executed at NMI signal detection will proceed to its completion. The interrupt mask bit of the condition code register doesn't affect non-maskable interrupt at all.

In response to an NMI interrupt, the contents of the program counter, index register, accumulators and condition code register will be saved onto the stack. Upon completion of this sequence, a vector is fetched from \$FFFC and \$FFFD to transfer their contents into the program counter and branch to the non-maskable interrupt service routine.

(Note) At reset start, the stack pointer should be initialized on an appropriate memory area and then the falling edge be input to NMI pin.

• Interrupt Request ( $\overline{IRQ}_1$ ,  $\overline{IRQ}_2$ )

These are level-sensitive pins which request an internal interrupt sequence to the CPU. At interrupt request, the CPU will complete

the current instruction before the acceptance of the request. Unless the interrupt mask in the condition code register is set, the CPU starts an interrupt sequence; if set, the interrupt request will be ignored. When the sequence starts, the contents of the program counter, index register, accumulators and condition code register will be saved onto the stack, then the CPU sets the interrupt mask bit and will not acknowledge the maskable request. During the last cycle, the CPU fetches vectors depicted in Table 1 and transfers their contents to the program counter and branches to the service routine.

The CPU uses the external interrupt pins ( $\overline{IRQ}_1$  and  $\overline{IRQ}_2$ ) also as port pins  $P_{50}$  and  $P_{51}$ , so it provides an enable bit to Bit 0 and 1 of the RAM port 5 control register at \$0014. Refer to "RAM/PORT 5 CONTROL REGISTER" for the details.

When one of the internal interrupts, ICI, OCI, TOI, CMI or SIO is generated, the CPU produces internal interrupt signal ( $IRQ_3$ ).  $IRQ_3$  functions just the same as  $\overline{IRQ}_1$  or  $\overline{IRQ}_2$  except for its vector address. Fig. 16 shows the block diagram of the interrupt circuit.

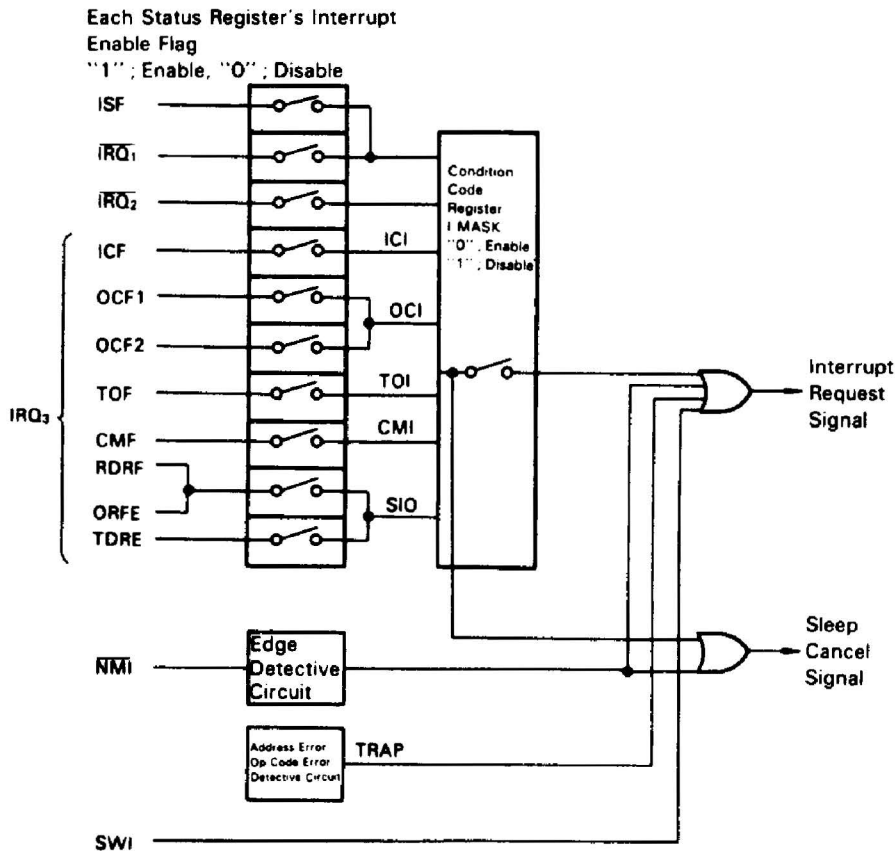


Figure 16 Interrupt Circuit Block Diagram



Table 1 Interrupt Vector Memory Map

| Priority                    | Vector |      | Interrupt                                    |
|-----------------------------|--------|------|--|
|                             | MSB    | LSB  |  |
| Highest<br>↑<br>↓<br>Lowest | FFFE   | FFFF | RES  |
|                             | FFEE   | FFEF | TRAP   |
|                             | FFFC   | FFFD | NMI  |
|                             | FFFA   | FFFB | SWI<br>(Software Interrupt)                  |
|                             | FFF8   | FFF9 | IRQ <sub>1</sub> , ISF (port 6 Input Strobe) |
|                             | FFF6   | FFF7 | ICI<br>(Timer 1 Input Capture)               |
|                             | FFF4   | FFF5 | OCI<br>(Timer 1 Output Compare 1, 2)         |
|                             | FFF2   | FFF3 | TOI<br>(Timer 1 Overflow)                    |
|                             | FFEC   | FFED | CMI<br>(Timer 2 Counter Match)               |
|                             | FFEA   | FFEB | IRQ <sub>2</sub>                             |
|                             | FFFO   | FFF1 | SIO<br>(RDRF + ORFE + TDRE + PER)            |

• **Mode Program (MP<sub>0</sub>, MP<sub>1</sub>)**

Set MP<sub>0</sub> "High" and MP<sub>1</sub> "Low".

• **Read/Write (R/W)**

This signal, usually be in read state ("High"), shows whether the CPU is in read ("High") or write ("Low") state to the peripheral or memory devices. This can drive one TTL load and 30pF capacitance.

• **RD, WR**

These signals show active low outputs when the CPU is reading/writing to the peripherals or memories. This enables the CPU easy to access the peripheral LSI with RD and WR input pins. These pins can drive one TTL load and 30pF capacitance.

• **Load Instruction Register (LIR)**

This signal shows the instruction opcode being on data bus (active low). This pin can drive one TTL load and 30pF capacitance.

• **Memory Ready (MR; P<sub>52</sub>)**

This is the input control signal which stretches the system clock's "High" period to access low-speed memories. HD6303Y can select three kinds of low-speed memory access method by RAM/Port 5 Control Register's MRE bit and AMRE bit. In the case that CPU accesses low-speed memories by the external MR signal (MRE="1", AMRE="0"), the system clock operates in normal sequence when this signal is in "High".

But this signal in "Low", the "High" period of the system clock will be stretched depending on its "Low" level duration in integral multiples of the cycle time. This allows the CPU to interface with low-speed memories (See Fig. 2). Up to 9μs can be stretched.

During internal address space access or nonvalid memory access, MR is prohibited internally to prevent decrease of operation speed. Even in the halt state, MR can also stretch "High" period of system clock to allow peripheral devices to access low-speed memo-

ries. Refer to "RAM/PORT 5 CONTROL REGISTER" for more details.

• **Halt (HALT; P<sub>53</sub>)**

This is an input control signal to stop instruction execution and to release buses. When this signal switches to "Low", the CPU stops to enter into the halt state after having executed the present instruction. When entering into the halt state, it makes BA "High" and also an address bus, data bus, RD, WR, R/W high impedance. When an interrupt is generated in the halt state, the CPU uses the interrupt handler after the halt is cancelled. When halted during the sleep state, the CPU keeps the sleep state, while BA is "High" and releases the buses. Then the CPU returns to the previous sleep state when the HALT signal becomes "High".

(Note) Please don't switch the HALT signal to "Low" when the CPU executes the WAI instruction and is in the interrupt wait state to avoid the trouble of the CPU's operation after the halt is cancelled.

• **Bus Available (BA)**

This is an output control signal which is normally "Low" but "High" when the CPU accepts HALT and releases the buses. The HD6800 and HD6802 make BA "High" and release the buses at WAI execution, while the HD6303Y doesn't make BA "High" under the same condition.

■ **PORT**

The HD6303Y provides three 8-bit I/O ports. Each port provides Data Direction Register (DDR) which controls the I/O state by the bit.

Table 2 Port and Data Direction Register Address

| Port   | Port Address | Data Direction Register |
|--------|--------------|-------------------------|
| Port 2 | \$0003       | \$0001                  |
| Port 5 | \$0015       | \$0020                  |
| Port 6 | \$0017       | \$0016                  |

• **Port 2**

An 8-bit I/O port. Port 2 DDR (P2DDR) controls the I/O state. This port provides DDR corresponding to each bit and can define input or output by the bit ("0" for input, "1" for output).

As Port 2 DDR is cleared during reset, it will be an input port.

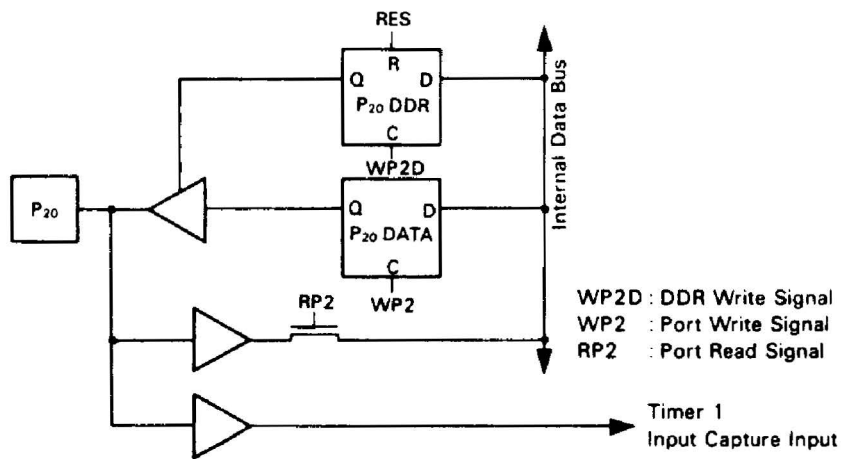
Port 2 is also used as an I/O pin for timer 1, Timer 2 and the SCI. Pins for Timers and the SCI set or reset each DDR depending on their functions and become I/O pins. When port 2 functions as an I/O port after used as I/O pins of the timers or the SCI, the I/O direction of the pins remain as it is used as the I/O pin of timer and SCI.

Port 2 can drive one TTL load and 30pF capacitance. This port can produce 1mA when V<sub>out</sub>=1.5V to drive directly the base of Darlington transistor.

• **P<sub>20</sub> (Tin)**

P<sub>20</sub> is also used as an external input pin for the input-capture. This pin is an I/O port which is an input or output as defined by the Data Direction Register (P<sub>20</sub>DDR) ("0" for an input and "1" for an output). Then either a signal to or from P<sub>20</sub> ("to" for an output port, "from" for an input port) is always input to the Timer 1 input capture.

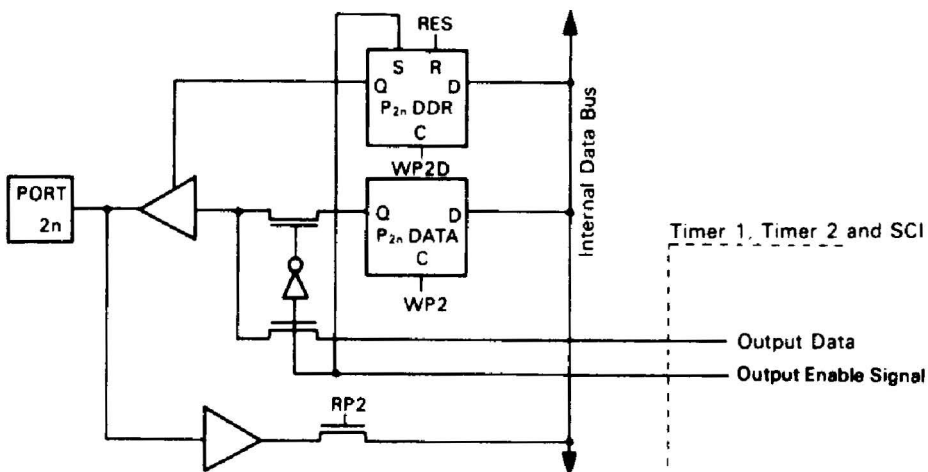




**P<sub>21</sub> (Tout 1), P<sub>24</sub> (Tx), P<sub>25</sub> (Tout 2), P<sub>26</sub> (Tout 3)**

These four pins can be also used as output pins for Timer 1, Timer 2 and a transmit output of the SCI. Timer 1, and the SCI

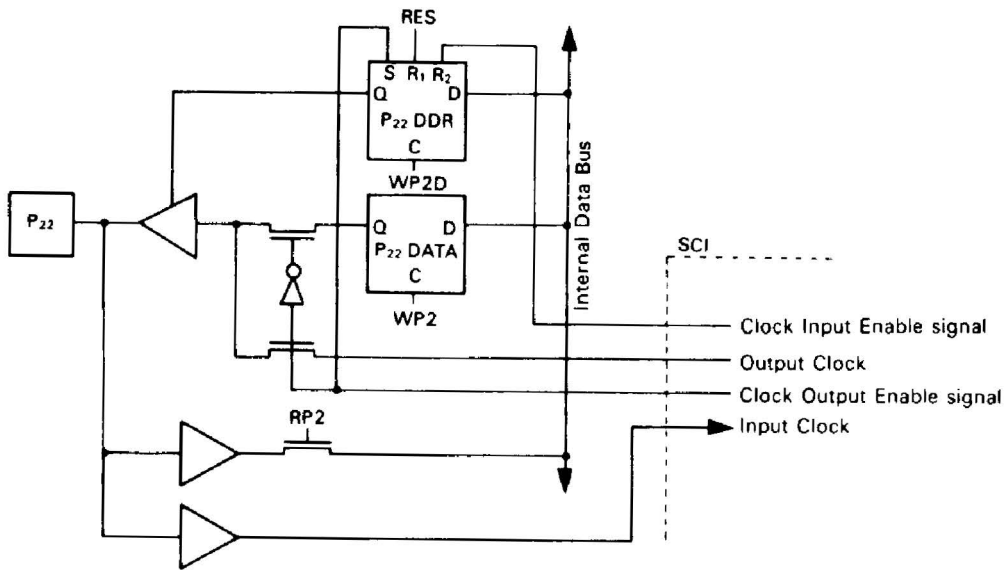
have a register which enables output. By setting these registers, they automatically will be output pins of timer or the SCI.



**P<sub>22</sub> (SCLK)**

P<sub>22</sub> is also used as a clock I/O pin for the SCI. It is selected as a clock input or output pin by the operating mode of the SCI. It is used

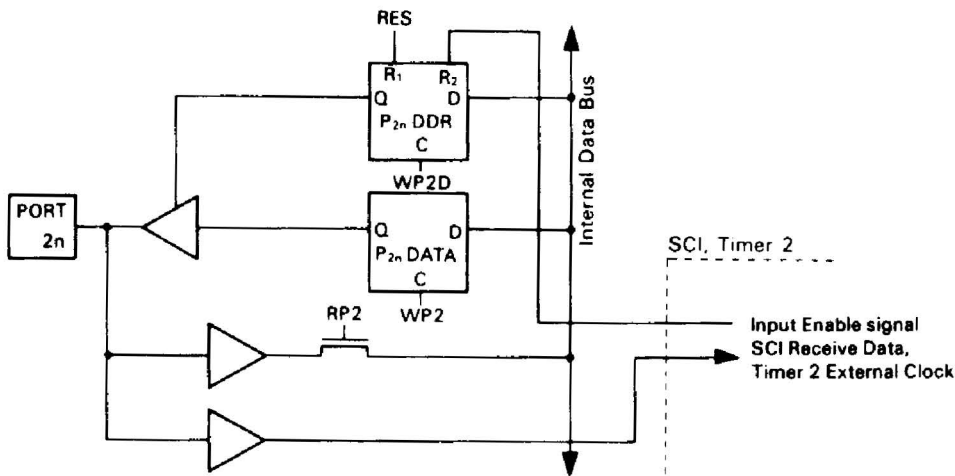
as an I/O port when the SCI has no clock input or output (as an output port if P<sub>22</sub> DDR=1, as an input port if P<sub>22</sub> DDR=0).



**P<sub>23</sub> (Rx), P<sub>27</sub> (TCLK)**

P<sub>23</sub> and P<sub>27</sub> are also used as received data input pins for the SCI and external clock input pins for Timer 2. The SCI and Timer 2 have registers which enable input. If the registers are set, the DDR (P<sub>23</sub>DDR, P<sub>27</sub>DDR) are cleared and P<sub>23</sub> and P<sub>27</sub> will be input pins for Rx and TCLK.

Since the SCI will be a clocked synchronous mode by an external clock-input during reset, the DDR of P<sub>22</sub> is cleared automatically and P<sub>22</sub> is an input port. Set the SCI to a mode where P<sub>22</sub> is not used (CC0 or CC1 of the RMC Register is "0" or "1" respectively) and write "1" to the P<sub>22</sub> DDR to make P<sub>22</sub> an output port.



| MSB             |                 |                 |                 | LSB             |                 |                 |                 |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| P <sub>27</sub> | P <sub>26</sub> | P <sub>25</sub> | P <sub>24</sub> | P <sub>23</sub> | P <sub>22</sub> | P <sub>21</sub> | P <sub>20</sub> |
| DDR             | DDR             | DDR             | DDR             | DDR             | DDR             | DDR             | DDR             |

PORT2 DDR (\$0001)  
(Write only, \$00 during reset.)

|                 |                 |                 |                 |                 |                 |                 |                 |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| P <sub>27</sub> | P <sub>26</sub> | P <sub>25</sub> | P <sub>24</sub> | P <sub>23</sub> | P <sub>22</sub> | P <sub>21</sub> | P <sub>20</sub> |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|

PORT2 (\$0003)  
(R/W, not initialized during reset.)



**Port 5**

An 8-bit I/O port. The DDR of port 5 controls I/O state. Each bit of port 5 has a DDR which defines I/O state ("0" for input and "1" for output).

During reset, the DDR of port 5 is cleared and port 5 becomes an input port.

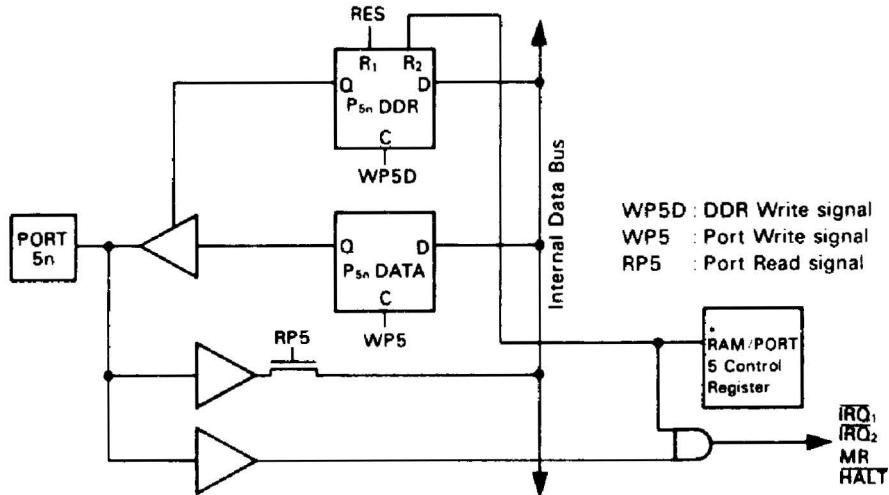
Port 5 is also usable as  $\overline{IRQ}_1$ ,  $\overline{IRQ}_2$ ,  $\overline{HALT}$ , MR and the strobed signal of port 6 for handshake ( $\overline{IS}$ ,  $\overline{OS}$ ). It is set to input or output automatically if it is used as these control signal pins (except  $P_{54}$ ,  $\overline{IS}$ ). Since the DDR of port 5, as is port 2, is set or reset by the control signal, I/O directions of the I/O ports are retained after the control signal is disabled. Port 5 can drive one TTL load and 90pF capacitance.

**$P_{50}$  ( $\overline{IRQ}_1$ ),  $P_{51}$  ( $\overline{IRQ}_2$ )**

$P_{50}$  and  $P_{51}$  are also usable as interrupt pins. The RAM/port 5 control registers of  $\overline{IRQ}_1$  and  $\overline{IRQ}_2$  have enable bits (IQ1E, IQ2E). When these bits are set to "1",  $P_{50}$  and  $P_{51}$  will automatically be interrupt input pins.

**$P_{52}$  (MR),  $P_{53}$  ( $\overline{HALT}$ )**

$P_{52}$  and  $P_{53}$  are also usable as MR and  $\overline{HALT}$  inputs. MR and  $\overline{HALT}$  have enable bits (MRE, HLTE) in the RAM/Port 5 Control Register as  $\overline{IRQ}_1$  and  $\overline{IRQ}_2$ . Since MRE is cleared during reset,  $P_{52}$  is usable as an I/O port, and HLTE is set during reset, the DDR of  $P_{53}$  will be automatically reset to be a  $\overline{HALT}$  input pin. HLTE of the RAM/Port 5 Control Register has to be cleared to use  $P_{53}$  as an I/O port.



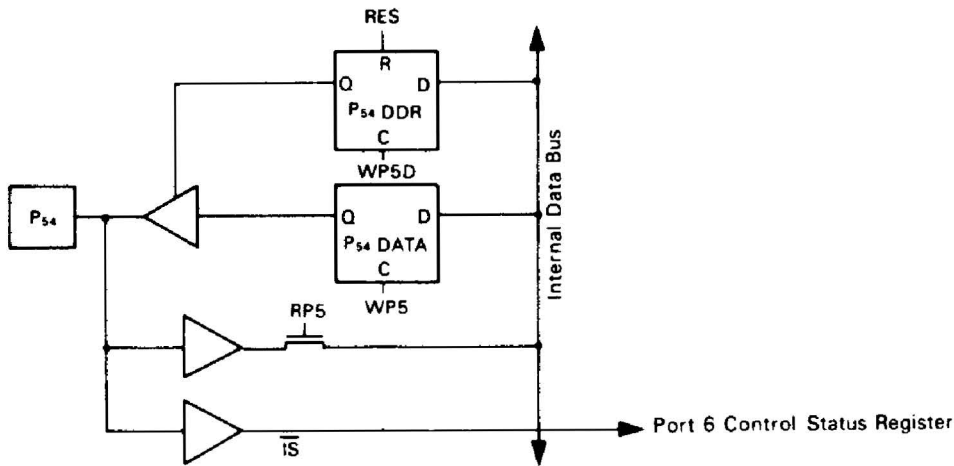
\* Initializing value during reset:  
 IRQ1E = "0", IRQ2E = "0", MRE = "0", HLTE = "1"

WP5D : DDR Write signal  
 WP5 : Port Write signal  
 RP5 : Port Read signal

**$P_{54}$  ( $\overline{IS}$ )**

$P_{54}$  is also usable as the input strobe ( $\overline{IS}$ ) for port 6 handshake interface. This pin, as is  $P_{20}$ , is always an I/O port. If  $P_{54}$  is used as an

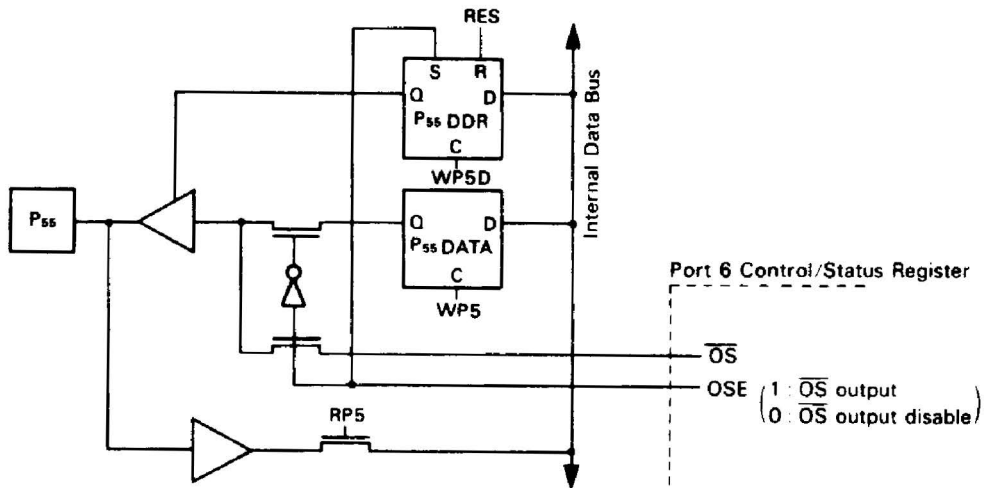
output port (set the DDR of  $P_{54}$  to "1"), an output signal from  $P_{54}$  will be the input to  $\overline{IS}$ .



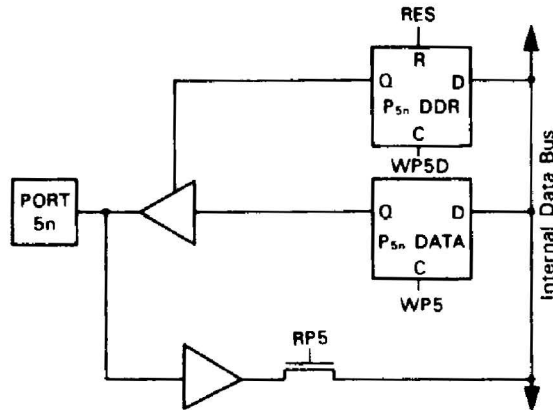
**P<sub>55</sub> ( $\overline{OS}$ )**

P<sub>55</sub> is also usable as the output strobe ( $\overline{OS}$ ) for port 6 handshake interface. It will be an I/O port during reset, and an  $\overline{OS}$  output pin

by setting the  $\overline{OS}$  enable register (OSE) of the port 6 Control Status Register (P6CSR).



P<sub>56</sub>, P<sub>57</sub>  
P<sub>56</sub> and P<sub>57</sub> are I/O ports.



| MSB                 |                     |                     |                     |                     |                     |                     | LSB                 |  |
|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|--|
| P <sub>57</sub> DDR | P <sub>56</sub> DDR | P <sub>55</sub> DDR | P <sub>54</sub> DDR | P <sub>53</sub> DDR | P <sub>52</sub> DDR | P <sub>51</sub> DDR | P <sub>50</sub> DDR | PORT5 DDR (\$0020)<br>(Write only, \$00 during reset.) |
| P <sub>57</sub>     | P <sub>56</sub>     | P <sub>55</sub>     | P <sub>54</sub>     | P <sub>53</sub>     | P <sub>52</sub>     | P <sub>51</sub>     | P <sub>50</sub>     | PORT5 (\$0015)<br>(R/W, not initialized during reset.) |



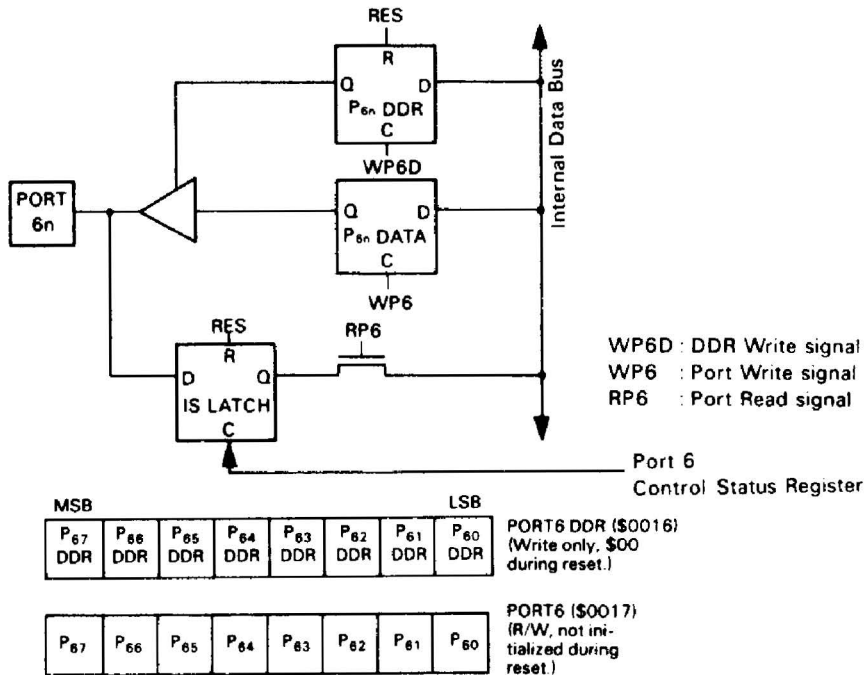


● **Port 6**

8-bit I/O port. Port 6 DDR controls I/O state. Each bit of port 6 has a DDR and designates input or output ("0" for input, "1" for output). During reset, Port 6 DDR is cleared and port 6 becomes an input port.

Port 6 controls parallel handshake interface besides functions as an I/O port. Therefore, it provides DDRs to control and IS LATCH to latch the input data.

Port 6 can drive one TTL load and 30pF capacitance. It can drive directly the base of Darlington transistor as port 2.



■ **BUS**

● **Address Bus (A<sub>0</sub> ~ A<sub>15</sub>)**

Address Bus (A<sub>0</sub> ~ A<sub>15</sub>) is used for addressing the memory and peripheral LSI.

This bus can interface with the bus of HMCS 6800 and drive one TTL load and 90pF capacitance.

● **Data Bus (D<sub>0</sub> ~ D<sub>7</sub>)**

8-bit parallel data bus for data transmit between the memory or peripheral LSI. This bus can drive one TTL load and 90pF capacitance.

■ **RAM/PORT 5 CONTROL REGISTER**

The control register located at \$0014 controls on-chip RAM and port 5.

RAM/Port 5 Control Register (RP5CR)

|          |      |           |       |      |     |                    |                    |        |
|----------|------|-----------|-------|------|-----|--------------------|--------------------|--------|
| 7        | 6    | 5         | 4     | 3    | 2   | 1                  | 0                  |        |
| STBY PWR | RAME | STBY FLAG | AMR E | HLTE | MRE | IRQ <sub>2</sub> E | IRQ <sub>1</sub> E | \$0014 |

**Bit 0, Bit 1  $\overline{IRQ_1}$ ,  $\overline{IRQ_2}$  Enable Bit (IRQ<sub>1</sub>E, IRQ<sub>2</sub>E)**

When using P<sub>50</sub> and P<sub>51</sub> as interrupt pins, write "1" in these bits. When the bit is set to "1", the DDRs corresponding to P<sub>50</sub> and

P<sub>51</sub> are cleared and become  $\overline{IRQ_1}$  input pin and  $\overline{IRQ_2}$  input pin. When IRQ<sub>1</sub>E and IRQ<sub>2</sub>E are set, P<sub>50</sub> and P<sub>51</sub> cannot be used as an output ports. When "0", the CPU doesn't accept an external interrupt or a sleep cancellation by the external interrupt. These bits are cleared during reset.

**Bit 2 Memory Ready Enable Bit (MRE)**

When using P<sub>52</sub> as an input pin of the "memory ready" signal, write "1" in this bit. When set, P<sub>52</sub> DDR is automatically cleared and becomes the MR input pin. The bit is cleared during reset.

**Bit 3 Halt Enable Bit (HLTE)**

When using P<sub>53</sub> as an input pin of the  $\overline{HALT}$  signal, write "1" in this bit. When this bit is set, P<sub>53</sub> DDR is automatically cleared and becomes the Halt input pin. If the bit is "0", the Halt function is inhibited and P<sub>53</sub> is used as an I/O port. The bit is set to "1" during reset.

**Bit 4 Auto Memory Ready Enable Bit (AMRE)**

When the bit is set and the CPU accesses the external address, "memory ready" operates automatically and stretches the E clock's "High" duration for one system clock. When MRE bit of bit 2 is cleared and when the CPU accesses the external address space, the function operates. When MRE bit is set and then the CPU accesses the external address space with P<sub>52</sub> (MR) pin in "low", "memory ready" operates automatically. This bit is set to "1" during reset.



Table 3 "Memory Ready" Function

| MRE | AMRE | Function   |
|-----|------|--|
| 0   | 0    | "Memory ready" inhibited.  |
| 0   | 1    | When the CPU accesses the external address, "High" duration of E clock automatically becomes one-cycle longer. This state is retained during reset.  |
| 1   | 0    | "Memory ready" operates by P <sub>52</sub> (MR) pin. The function is the same as that of the HD6301X0.   |
| 1   | 1    | When the CPU accesses the external address space with the P <sub>52</sub> (MR) pin in "low", the "auto memory ready" operates. This function is effective if it has both "high-speed memory" and "slow memory" outside. Input $\overline{CS}$ signal of "slow memory" to MR pin. |

**Bit 5 Standby Flag (STBY FLAG)**

By clearing this flag, HD6303Y gets into the standby mode by software. This flag is set to "1" during reset, so the standby mode is canceled with  $\overline{RES}$  pin in "low". The  $\overline{RES}$  pin should be in "low" until oscillation becomes stable (min. 20ms.). If the STBY pin in is in "low", the standby mode can not be canceled with the  $\overline{RES}$  pin in "low".

**Bit 6 RAM Enable (RAME)**

On-chip RAM can be disabled by this control bit. By resetting the MPU, "1" is set to this bit, and on-chip RAM is enabled. When

this bit is cleared (=logic "0") on-chip RAM is invalid and the CPU can read data from external memory. This bit should be "0" before getting into the standby mode to protect on-chip RAM data.

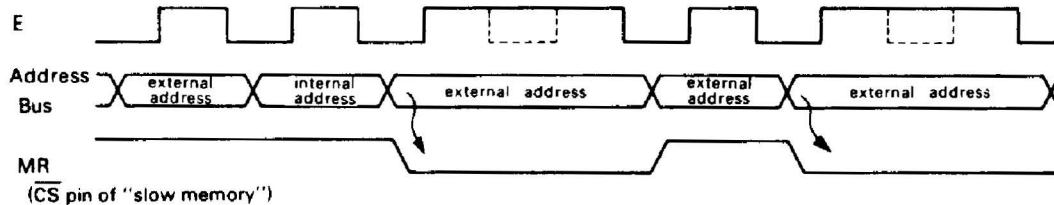
**Bit 7 Standby Power Bit (STBY PWR)**

When V<sub>CC</sub> is not provided in standby mode, this bit is cleared. This is a flag for read/write and can be read by software. If this bit is set before standby mode, and remains set even after returning from standby mode, V<sub>CC</sub> voltage is provided during standby mode and the on-chip RAM data is valid.

(a) MRE=0, AMRE=1



(b) MRE=1, AMRE=1



(c) MRE=1, AMRE=0 (HD6301X0 Compatible Mode)

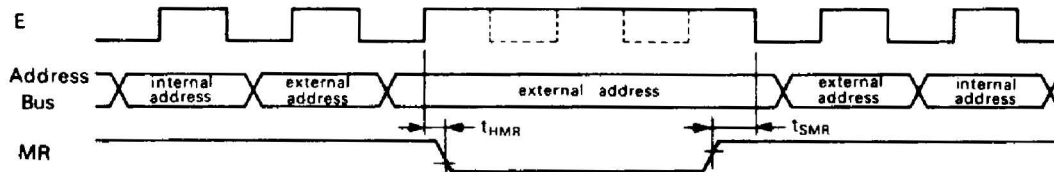


Figure 17 Memory Ready Timing

**Port 6 Control/Status Register**

This is the Control/Status Register for parallel handshake interface using Port 6. The functions are as follows;

- 1) Latches input data to Port 6 at the  $\overline{IS}$  (P<sub>54</sub>) falling edge.
- 2) Outputs a strobe signal OS (P<sub>53</sub>) outward by reading or writing to port 6.
- 3) When IS FLAG is set at the  $\overline{IS}$  falling edge, an interrupt occurs.

The following shows Port 6 Control/Status Register (P6CSR).

|          |                            |     |     |              |   |   |   |        |
|----------|----------------------------|-----|-----|--------------|---|---|---|--------|
| 7        | 6                          | 5   | 4   | 3            | 2 | 1 | 0 |        |
| IS* FLAG | IS IRQ <sub>1</sub> ENABLE | OSE | OSS | LATCH ENABLE | - | - | - | \$0021 |

\*Bit 7 is Read-Only bit



# HD6303Y, HD63A03Y, HD63B03Y, HD63C03Y

**Bit 0**

**Bit 1** Not used.

**Bit 2**

**Bit 3: Latch Enable**

This register controls the input latch for Port 6 (ISLATCH). When this bit is set to "1", the input data to port 6 will be latched inward at the  $\overline{IS}$  ( $P_{64}$ ) falling edge. An input latch will be canceled by reading Port 6, which enables to latch the next data. If cleared, the input latch remains canceled and this bit functions as a usual input port. This bit is cleared during reset.

**Bit 4: OSS Output Strobe Select**

This register initiates an output strobe ( $\overline{OS}$ ) from  $P_{55}$  by reading or writing to port 6. When cleared,  $\overline{OS}$  occurs by reading Port 6. When set,  $\overline{OS}$  occurs by writing to Port 6. This bit is cleared during reset.

**Bit 5: OSE Output Strobe Enable**

This register decides the enabling or disabling of the output

strobe. When cleared,  $P_{55}$  functions as an I/O port. When set,  $P_{55}$  functions as an  $\overline{OS}$  output pin. ( $P_{55}$  DDR is set by OSE.) This bit is cleared during reset.

**Bit 6: IS IRQ<sub>1</sub> Enable Input Strobe Interrupt Enable**

When set, an  $\overline{IRQ_1}$  interrupt to the CPU occurs by setting IS FLAG of bit 7. When cleared, the interrupt does not occur. This bit is cleared during reset.

**Bit 7: IS Flag Input Strobe Flag**

This flag is set at the IS ( $P_{64}$ ) falling edge. This flag is for read-only. When set, the flag is cleared by reading or writing to Port 6 after reading the Port 6 Control Status Register. This bit is cleared during reset.

**■ MEMORY MAP**

The MPU can address up to 65k bytes. Memory map is shown in Fig. 20. 40 addresses (\$0000 ~ \$0027 except \$00, \$02, \$04, \$05, \$06, \$07, \$18) are the internal registers as shown in Table 4.

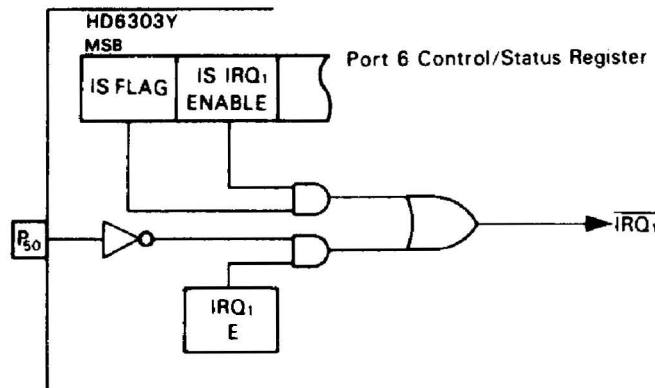


Figure 18 Input Strobe Interrupt block Diagram

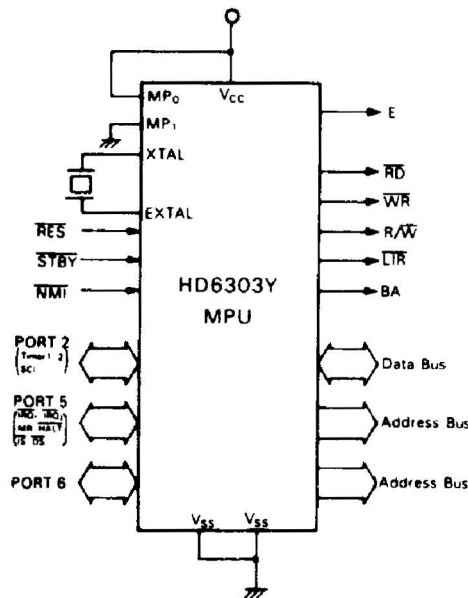


Figure 19 HD6303Y Operating Function



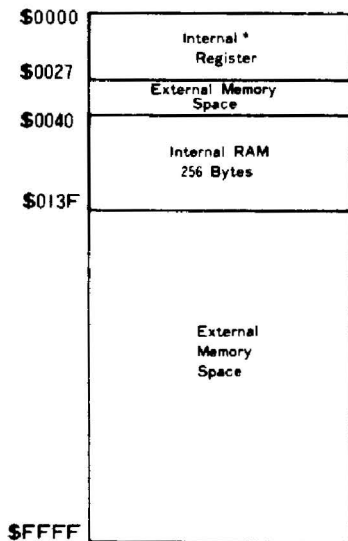
Table 4 Internal Register

| Address | Register                             | Abbreviation | R/W** | Initialized value during reset*** |
|---------|--------------------------------------|--------------|-------|-----------------------------------|
| 00*     | Port 1 DDR (Data Direction Register) | P1DDR        | W     | \$FE                              |
| 01      | Port 2 DDR                           | P2DDR        | W     | \$00                              |
| 02*     | Port 1                               | PORT1        | R/W   | indefinite                        |
| 03      | Port 2                               | PORT2        | R/W   | indefinite                        |
| 04*     | Port 3 DDR                           | P3DDR        | W     | \$FE                              |
| 05*     | Port 4 DDR                           | P4DDR        | W     | \$00                              |
| 06*     | Port 3                               | PORT3        | R/W   | indefinite                        |
| 07*     | Port 4                               | PORT4        | R/W   | indefinite                        |
| 08      | Timer Control/Status Register 1      | TCSR1        | R/W   | \$00                              |
| 09      | Free Running Counter (MSB)           | FRCH         | R/W   | \$00                              |
| 0A      | Free Running Counter (LSB)           | FRCL         | R/W   | \$00                              |
| 0B      | Output Compare Register 1 (MSB)      | OCR1H        | R/W   | \$FF                              |
| 0C      | Output Compare Register 1 (LSB)      | OCR1L        | R/W   | \$FF                              |
| 0D      | Input Capture Register (MSB)         | ICRH         | R     | \$00                              |
| 0E      | Input Capture Register (LSB)         | ICRL         | R     | \$00                              |
| 0F      | Timer Control/Status Register 2      | TCSR2        | R/W   | \$10                              |
| 10      | Rate/Mode Control Register           | RMCR         | R/W   | \$C0                              |
| 11      | Tx/Rx Control Status Register 1      | TRCSR1       | R/W   | \$20                              |
| 12      | Receive Data Register                | RDR          | R     | \$00                              |
| 13      | Transmit Data Register               | TDR          | W     | indefinite                        |
| 14      | RAM/Port 5 Control Register          | RP5CR        | R/W   | \$F8 or \$78                      |
| 15      | Port 5                               | PORT5        | R/W   | indefinite                        |
| 16      | Port 6 DDR                           | P6DDR        | W     | \$00                              |
| 17      | Port 6                               | PORT6        | R/W   | indefinite                        |
| 18      | Port 7                               | PORT7        | R/W   | indefinite                        |
| 19      | Output Compare Register 2 (MSB)      | OCR2H        | R/W   | \$FF                              |
| 1A      | Output Compare Register 2 (LSB)      | OCR2L        | R/W   | \$FF                              |
| 1B      | Timer Control/Status Register 3      | TCSR3        | R/W   | \$20                              |
| 1C      | Time Constant Register               | TCONR        | W     | \$FF                              |
| 1D      | Timer 2 Up Counter                   | T2CNT        | R/W   | \$00                              |
| 1E      | Tx/Rx Control Status Register 2      | TRCSR2       | R/W   | \$28                              |
| 1F****  | Test Register*                       | TSTREG       | —     | —                                 |
| 20      | PORT 5 DDR                           | P5DDR        | W     | \$00                              |
| 21      | PORT 6 Control/Status Register       | P6CSR        | R/W   | \$07                              |
| 22      | —                                    | —            | —     | —                                 |
| 23      | —                                    | —            | —     | —                                 |
| 24      | —                                    | —            | —     | —                                 |
| 25      | —                                    | Reserved     | —     | —                                 |
| 26      | —                                    | —            | —     | —                                 |
| 27      | —                                    | —            | —     | —                                 |

\* External address.  
 \*\* R: Read-only register, W: Write-only register, R/W: Read/Write register.  
 \*\*\* When empty bit is in the register, it is set to "1".  
 \*\*\*\* Register for test. Don't access this register.

2





\*This mode does not include the addresses: \$00, \$02, \$04, \$05, \$06, \$07 or \$18 which can be used externally.

Figure 20 HD6303Y Memory Map

**■ TIMER 1**

The HD6303Y provides a 16-bit programmable timer which can simultaneously measure an input waveform and generate two independent output waveforms. The pulse widths of both input/output waveforms vary from microseconds to seconds.

Timer 1 is configured as follows (refer to Fig. 22).

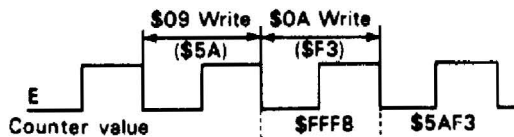
- Control/Status Register 1 (8 bit)
- Control/Status Register 2 (7 bit)
- Free Running Counter (16 bit)
- Output Compare Register 1 (16 bit)
- Output Compare Register 2 (16 bit)
- Input Capture Register (16 bit)

**● Free-Running Counter (FRC)(\$0009:000A)**

The key timer element is a 16-bit free-running counter driven and incremented by system clock. The counter value is readable by software without affecting the counter. The counter is cleared during reset.

When writing to the upper byte (\$09), the CPU writes the preset value (\$FFF8) into the counter (address \$09, \$0A) regardless of the write data value. But when writing to the lower byte (\$0A) after the upper byte writing, the CPU writes not only lower byte data into lower 8 bit, but also upper byte data into higher 8 bit of the FRC.

The counter will be as follows when the CPU writes to it by double store instructions (STD, STX, etc.)



In the case of the CPU write (\$5AF3) to the FRC

Figure 21 Counter Write Timing

**● Output Compare Register (OCR) (\$000B, \$000C; OCR1) (\$0019, \$001A; OCR2)**

The output compare register is a 16-bit read/write register which can control an output waveform. The data of OCR is always compared with the FRC.

When the data matches, output compare flag (OCF) in the timer control/status register (TCSR) is set. If an output enable bit (OE) in the TCSR2 is "1", an output level bit (OLVL) in the TCSR will be output to bit 1 (OCR 1) and bit 5 (OCR 2) of port 2. To control the output level again by the next compare, the value of OCR and OLVL should be changed. The OCR is set to \$FFFF at reset. The compare function is inhibited for a cycle just after a write to the upper byte of the OCR or FRC. This is to set the 16-bit value valid in the counter register for compare. In addition, it is because counter is to set \$FFF8 at the next cycle of the CPU's upper byte write to the FRC.

\* For data write to the FRC or the OCR, 2-byte transfer instruction (such as STX, etc.) should be used.

**● Input Capture Register (ICR) (\$000D : 000E)**

The input capture register is a 16-bit read-only register which stores the FRC's value when external input signal transition generates an input capture pulse. Such transition is controlled by input edge bit (IEDG) in the TCSR1.

In order to input the external input signal to the edge detector, a bit of the DDR corresponding to bit 0 of port 2 should be cleared ("0"). When an input capture pulse occurs by external input signal transition at the next cycle of CPU's high-byte read of the ICR, the input capture pulse will be delayed by one cycle. In order to ensure the input capture operation, a CPU read of the ICR needs 2-byte transfer instruction. The input pulse width should be at least 2 system cycles. This register is cleared (\$0000) during reset.

**● Timer Control/Status Register 1 (TCSR1) (\$0008)**

The timer control/status register 1 is an 8-bit register. All bits are readable and the lower 5 bits are also writable. The upper 3 bits are read-only which indicate the following timer status.

- Bit 5 The counter value reached to \$0000 as a result of counting-up (TOF).
- Bit 6 A match has occurred between the FRC and the OCR 1 (OCF1).
- Bit 7 Defined transition of the timer input signal causes the counter to transfer its data to the ICR (ICF).

The followings are the each bit descriptions.

Timer Control/Status Register 1

|     |      |     |      |       |      |      |       |        |
|-----|------|-----|------|-------|------|------|-------|--------|
| 7   | 6    | 5   | 4    | 3     | 2    | 1    | 0     |        |
| ICF | OCF1 | TOF | EICI | EOCI1 | ETOI | IEDG | OLVL1 | \$0008 |

**Bit 0 OLVL1 Output Level 1**

OLVL1 is transferred to port 2, bit 1 when a match occurs between the counter and the OCR1. If bit 0 of the TCSR2 (OE1) is set to "1", OLVL1 will appear at bit 1 of port 2.

**Bit 1 IEDG Input Edge**

This bit determines which edge, rising or falling, of input signal of bit 0 of port 2 will trigger data transfer from the counter to the ICR. For this function, the DDR corresponding to port 2, bit 0 should be cleared beforehand.

IEDG=0, triggered on a falling edge ("High" to "Low")

IEDG=1, triggered on a rising edge ("Low" to "High")

**Bit 2 ETOI Enable Timer Overflow Interrupt**

When this bit is set, an internal interrupt (IRQ<sub>3</sub>) by TOI interrupt is enabled. When cleared, the interrupt is inhibited.

**Bit 3 EOCI1 Enable Output Compare Interrupt 1**



When this bit is set, an internal interrupt (IRQ<sub>3</sub>) by OC11 interrupt is enabled. When cleared, the interrupt is inhibited.

**Bit 4 EIC1 Enable Input Capture Interrupt**

When this bit is set, an internal interrupt (IRQ<sub>3</sub>) by ICI interrupt is enabled. When cleared, the interrupt is inhibited.

**Bit 5 TOF Timer Overflow Flag**

This read-only bit is set when the counter increments from \$FFFF by 1. Cleared when the counter's MSB byte (\$0009) is read by the CPU after the TCSR1 read at TOF=1.

**Bit 6 OCF1 Output Compare Flag 1**

This read-only bit is set when a match occurs between the OCR1 and the FRC. Cleared when writing to the OCR1 (\$000B or \$000C) after the TCSR1 or TCSR2 read at OCF=1.

**Bit 7 ICF Input Capture Flag**

This read-only bit is set when an input signal of port 2, bit 0 makes a transition as defined by IEDG and the FRC is transferred to the ICR. Cleared when reading the upper byte (\$000D) of the ICR after the TCSR1 or TCSR2 read at ICF=1.

• **Timer Control/Status Register 2 (TCSR2) (\$000F)**

The timer control/status register 2 is a 7-bit register. All bits are readable and the lower 4 bits are also writable. But the upper 3 bits are read-only which indicate the following timer status.

**Bit 5** A match has occurred between the FRC and the OCR2 (OCF2).

**Bit 6**

Timer Control/Status Register 2

|     |      |      |   |       |       |     |     |        |
|-----|------|------|---|-------|-------|-----|-----|--------|
| 7   | 6    | 5    | 4 | 3     | 2     | 1   | 0   |        |
| ICF | OCF1 | OCF2 | - | EOCI2 | OLVL2 | OE2 | OE1 | \$000F |

**Bit 7** The same status flag as the ICF flag of the TCSR1, bit 7. The followings are the each bit descriptions.

**Bit 0 OE1 Output Enable 1**

This bit enables the OLVL1 to appear at port 2, bit 1 when a match has occurred between the counter and the output compare register 1. When this bit is cleared, bit 1 of port 2 will be an I/O port. When set, it will be an output of OLVL1 automatically.

**Bit 1 OE2 Output Enable 2**

This bit enables the OLVL2 to appear at port 2, bit 5 when a match has occurred between the counter and the output compare register 2. When this bit is cleared, port 2, bit 5 will be an I/O port. When set, it will be an output of OLVL2 automatically.

**Bit 2 OLVL2 Output Level 2**

OLVL2 is transferred to port 2, bit 5 when a match has occurred between the counter and the OCR2. If bit 5 of the TCSR2 (OE2), is set to "1", OLVL2 will appear at port 2, bit 5.

**Bit 3 EOC12 Enable Output Compare Interrupt 2**

When this bit is set, an internal interrupt (IRQ<sub>3</sub>) by OC12 interrupt is enabled. When cleared, the interrupt is inhibited.

**Bit 4** Not used

**Bit 5 OCF2 Output Compare Flag 2**

This read-only bit is set when a match has occurred between the counter and the OCR2. Cleared when writing to the OCR2 (\$0019 or \$001A) after the TCSR2 read at OCF2=1.

**Bit 6 OCF1 Output Compare Flag 1**

**Bit 7 ICF Input Capture Flag**

OCF1 and ICF are dual addressed. If which register, TCSR1 or TCSR2, CPU reads, it can read OCF1 and ICF to bit 6 and bit 7.

Both the TCSR1 and TCSR2 will be cleared during reset.  
(Note) If OE1 or OE2 is set to "1" before the first output compare match occurs after reset restart, bit 1 or bit 5 of port 2 will produce "0" respectively.

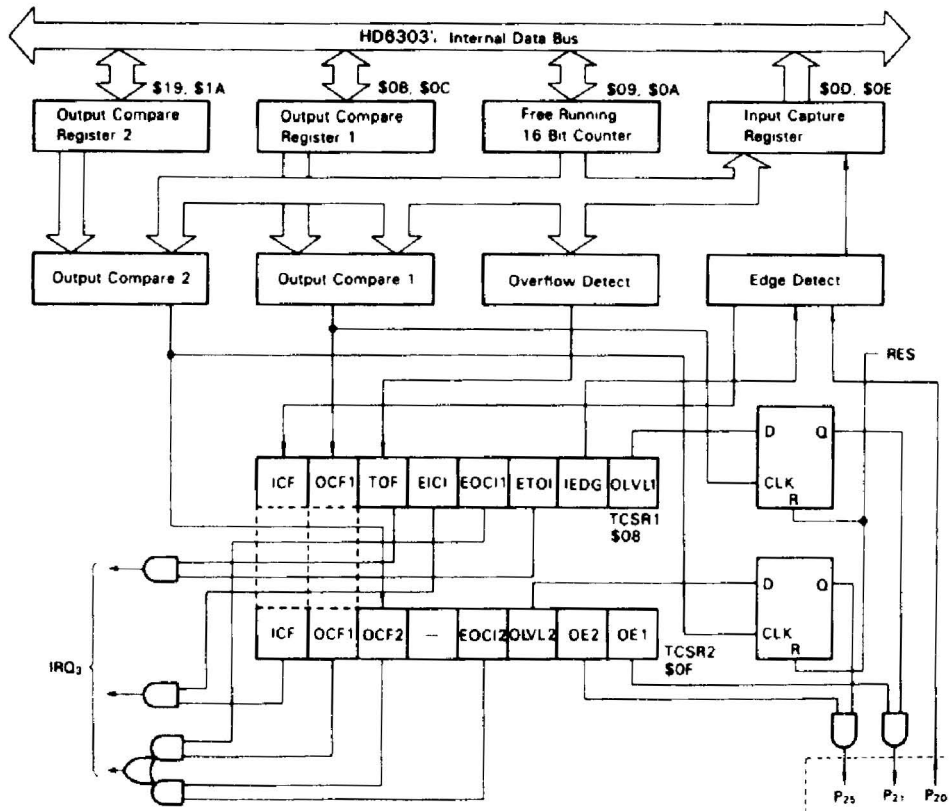


Figure 22 Timer 1 Block Diagram



**■ TIMER 2**

In addition to the timer 1, the HD6303Y provides an 8-bit re-loadable timer, which is capable of counting the external event. The timer 2 contains a timer output, so the MPU can generate three independent waveforms. (Refer to Fig. 23.)

The timer 2 is configured as follows:

- Control/Status Register 3 (7 bits)
- 8-bit Up Counter
- Time Constant Register (8 bits)

**• Timer 2 Up Counter (T2CNT) (\$001D)**

This is an 8-bit up counter which operates with the clock decided by CKS0 and CKS1 of the TCSR3. The CPU can read the value of the counter without affecting the counter. In addition, any value can be written to the counter by software even during counting.

The counter is cleared when a match occurs between the counter and the TCONR or during reset.

If the write operation is made by software to the counter at the cycle of counter clear, it does not reset the counter but put the write data to the counter.

**• Time Constant Register (TCONR) (\$001C)**

The time constant register is an 8-bit write only register. The data of register is always compared with the counter.

When a match has occurred, the counter match flag (CMF) of the timer control status register 3 (TCSR3) is set and the value

selected by TOS0 and TOS1 of the TCSR3 will appear at port 2, bit 6. When CMF is set, the counter will be cleared simultaneously and then start counting from \$00. This enables regular interrupts and waveform outputs without any software support. The TCONR is set to "\$FF" during reset.

**• Timer Control/Status Register 3 (TCSR3) (\$001B)**

The timer control/status register 3 is a 7-bit register. All bits are readable and 6 bits except for CMF can be written.

The followings are each pin descriptions.

Timer Control/Status Register 3

|     |      |   |     |      |      |      |      |        |
|-----|------|---|-----|------|------|------|------|--------|
| 7   | 6    | 5 | 4   | 3    | 2    | 1    | 0    | \$001B |
| CMF | ECMI | — | T2E | TOS1 | TOS0 | CKS1 | CKS0 |        |

**Bit 0 CKS0 Input Clock Select 0**

**Bit 1 CKS1 Input Clock Select 1**

Input clock to the counter is selected as shown in Table 5 depending on these two bits. When an external clock is selected, bit 7 of port 2 will be a clock input automatically. Timer 2 detects the rising edge of the external clock and increments the counter. The external clock is countable up to half the frequency of the system clock.

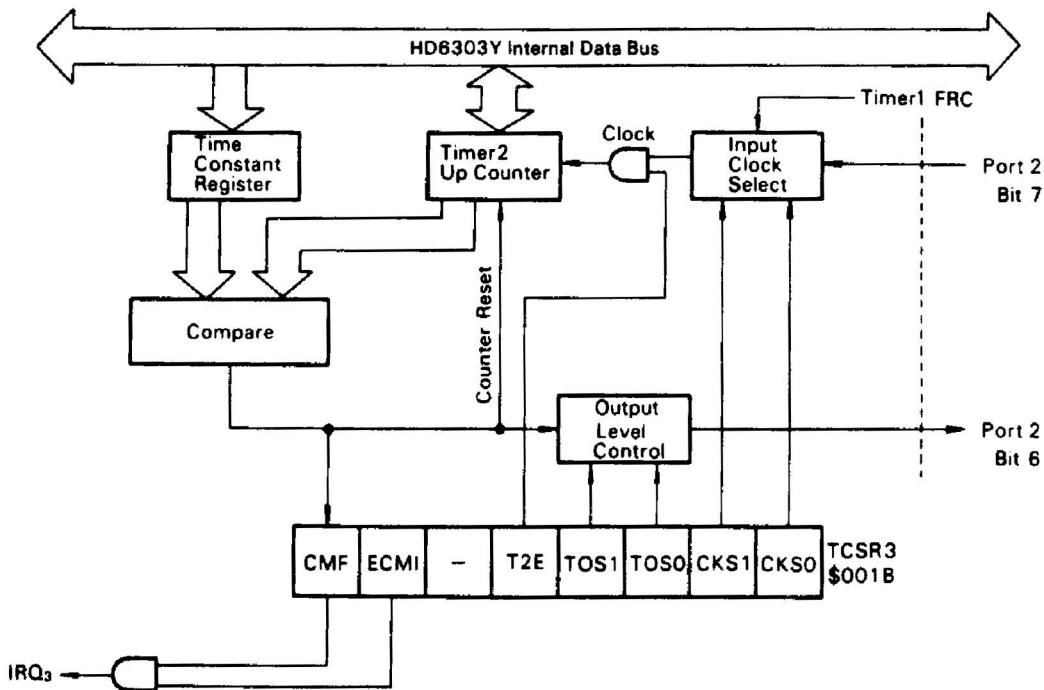


Figure 23 Timer 2 Block Diagram



Table 5 Input Clock Select

| CKS1 | CKS0 | Input Clock to the Counter |
|------|------|----------------------------|
| 0    | 0    | E clock                    |
| 0    | 1    | E clock/B*                 |
| 1    | 0    | E clock/128*               |
| 1    | 1    | External clock             |

\* These clocks come from the FRC of the timer 1. If one of these clocks is selected as an input clock to the up counter, the CPU should not write to the FRC of the timer 1.

**Bit 2 TOS0 Timer Output Select 0**

**Bit 3 TOS1 Timer Output Select 1**

When a match occurs between the counter and the TCONR timer 2 outputs shown in Table 6 will appear at port 2, bit 6 depending on these two bits. When both TOS0 and TOS1 are "0", bit 6 of port 2 will be an I/O port.

Table 6 Timer 2 Output Select

| TOS1 | TOS0 | Timer Output           |
|------|------|------------------------|
| 0    | 0    | Timer Output Inhibited |
| 0    | 1    | Toggle Output*         |
| 1    | 0    | Output "0"             |
| 1    | 1    | Output "1"             |

\* When a match occurs between the counter and the TCONR, timer 2 output level is reversed. This leads to production of a square wave with 50% duty to the external without any software support.

**Bit 4 T2E Timer 2 Enable Bit**

When this bit is cleared, a clock input to the up counter is inhibited and the up counter stops. When set to "1", a clock

selected by CKS1 and CKS0 (Table 5) is input to the up counter. (Note) P<sub>26</sub> outputs "0" when T2E bit cleared and timer 2 set in output enable condition by TOS1 or TOS0. It also outputs "0" when T2E bit set "1" and timer 2 set in output enable condition before the first counter match occurs.

**Bit 5 Not Used.**

**Bit 6 ECMI Enable Counter Match Interrupt**

When this bit is set, an internal interrupt (IRQ<sub>3</sub>) by CMI is enabled. When cleared, the interrupt is inhibited.

**Bit 7 CMF Counter Match Flag**

This read-only bit is set when a match occurs between the up counter and the TCONR. Cleared by writing "0" at CMF=1 by software (unable to write "1" by software). Each bit of the TCSR3 is cleared during reset.

**■ SERIAL COMMUNICATION INTERFACE (SCI)**

The Serial Communication Interface (SCI) in the HD6303Y contains the following two operating modes: asynchronous mode by the NRZ format, and clocked synchronous mode which transfers data synchronously with the clock. In the asynchronous mode, data length, parity bits and number of stop bits can be selected, and eight transfer formats are provided.

The SCI consists of the following registers as shown in Fig. 24 Block Diagram.

- Transmit/Receive Control Status Register 1 (TRCSR1)
- Rate/Mode Control Register (RMCR)
- Transmit/Receive Control Status Register 2 (TRCSR2)
- Receive Data Register (RDR)
- Receive Shift Register
- Transmit Data Register (TDR)
- Transmit Shift Register

To operate the SCI, initialize the RMCR and TRCSR2, after selecting the desirable operating mode and transfer format. Next, set the enable bit (TE or RE) of the TRCSR1. Operating mode and transfer format should be changed when the enable bit (TE, RE) is cleared. When setting the TE or RE again after changing the operating mode or transfer format, interval of more than a 1-bit cycle of the baud rate or bit rate is necessary. If a 1-bit cycle or more is not allowed, the SCI block may not be initialized.

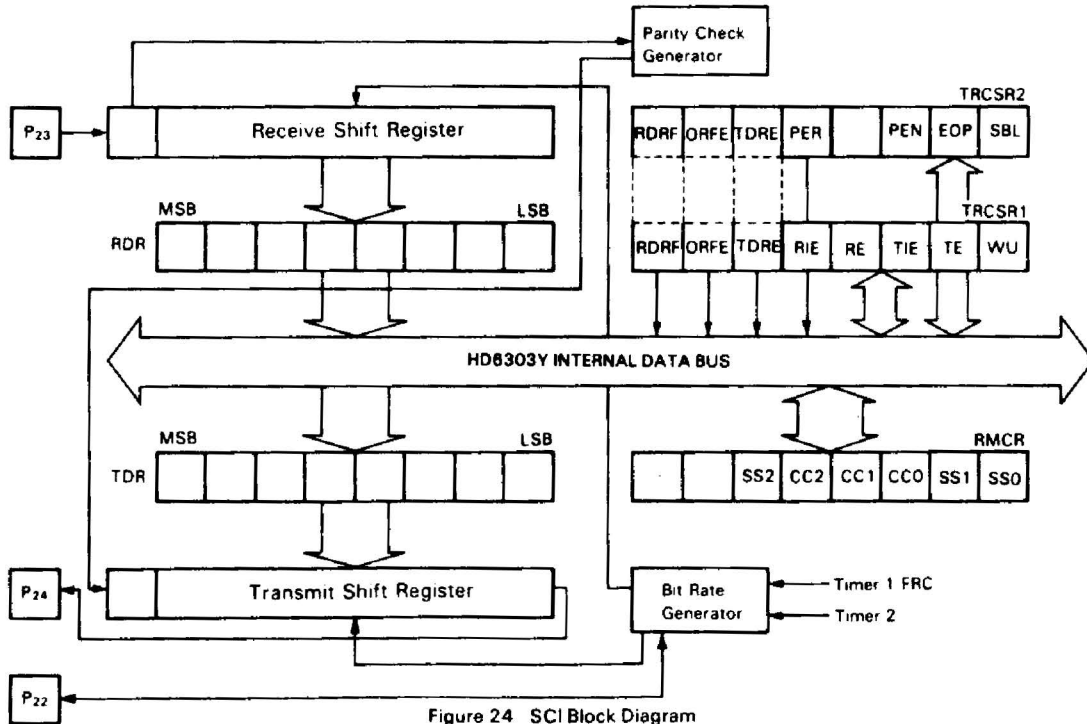


Figure 24 SCI Block Diagram





## • Asynchronous Mode

Asynchronous mode contains 8 transfer formats as shown in Fig. 25.

Data transmission is enabled by setting TE bit of the TRCSR1, then port 2, bit 4 will unconditionally become a serial output independently of the corresponding DDR.

To transmit data, set the desirable transmit format with RMCR and TRCSR2. When the TE bit is set, the data can be transmitted after transmitting the one frame of preamble ("1").

The conditions at this stage are as follows.

- 1) If the TDR is empty (TDRE=1), consecutive 1's are produced to indicate the idle state.
- 2) If the TDR contains data (TDRE=0), data is sent to the Transmit Shift Register and data transmit starts.

During data transmit, a start bit of "0" is transmitted first. Then 7-bit or 8-bit data (starts from bit 0) is transmitted. With PEN=1, the parity bit, even or odd, selected by EOP bit is added, lastly the stop bit (1 bit or 2 bits) is sent.

When the TDR is "empty", hardware sets TDRE flag bit. If the CPU doesn't respond to the flag in proper timing (the TDRE is in set condition till the next normal data transfer starts from the transmit data register to the transmit shift register), "1" is transferred instead of the start bit "0" and continues to be transferred till data is provided to the data register. While the TDRE is "1", "0" is not transferred.

Data receive is possible by setting RE bit. This makes port 2, bit 3 a serial input. The operation mode of data receive is decided by

the contents of the TRCSR2 and RMCR at first, and set RE bit of TRCSR1. The first "0" (space) synchronizes the receive bit flow. Each bit of the following data will be strobed in the middle. If a stop bit is not "1", a framing error assumed and ORFE is set.

When a framing error occurs, receive data is transferred to the Receive Data Register and the CPU can read the error-generating data. This makes it possible to detect a line break.

When PEN bit is set, the parity check is done. If the parity bit does not match the EOP bit, a parity error occurs and the PER bit is set, not the RDRF bit. Also, when the parity error occurs the receive data can be read just like in the case of the framing error.

The RDRF flag is set when the data is received without a framing error and a parity error.

If RDRF is still set when receiving the stop bit of the next data, ORFE is set to indicate the overrun generation. CPU can get the receive data by reading RDR. When 7 bit data format is selected, the 8th bit of RDR is "0".

When the CPU read the receive Data Register as a response to RDRF flag or ORFE flag after having read TRCSR, RDRF or ORFE is cleared.

(Note) Clock Source in Asynchronous Mode

If CC1:CC0=10, the internal bit rate clock is provided at P<sub>22</sub> regardless of the values for TE or RE. Maximum clock rate is  $E \div 16$ .

If both CC1 and CC0 are set, an external TTL compatible clock must be connected to P<sub>22</sub> at sixteen times (16x) the desired bit rate, but not greater than E.

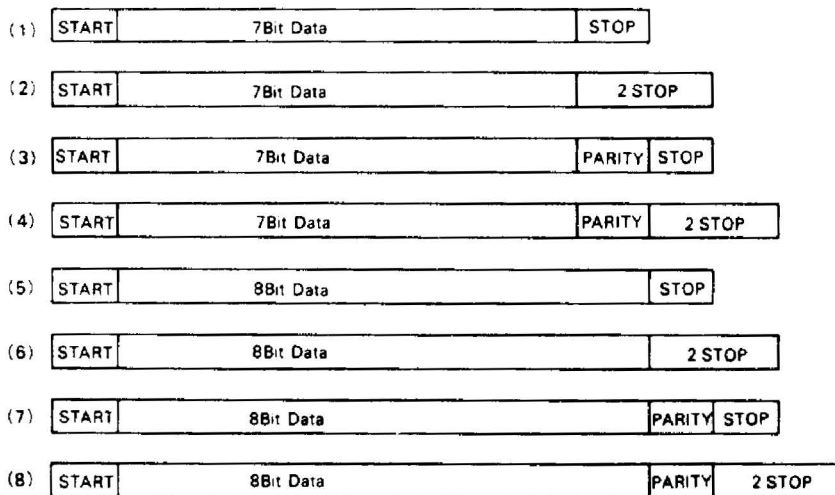


Figure 25 Asynchronous Mode Transfer Format

## • Clocked Synchronous Mode

In the clocked synchronous mode, data transmit is synchronized with the clock pulse. The HD6303Y SCI provides functionally independent transmitter and receiver which makes full duplex operation possible in the asynchronous mode. But in the clocked synchronous mode an SCI clock I/O pin is only P<sub>22</sub>, so the simultaneous receive and transmit operation is not available. In this mode, TE and RE should not be in set condition ("1") simultaneously. Fig. 26 gives a synchronous clock and a data format in the clocked synchronous mode.

### 1) Data transmit

Data transmit is realized by setting TE bit in the TRCSR1. Port 2, bit 4 becomes an output unconditionally independent of the value of the corresponding DDR.

Both the RMCR and TRCSR should be set in the desirable operating condition for data transmit.

When an external clock input is selected and the TDRE flag is "0", data transmit is performed from port 2, bit 4, synchronizing with 8 clock pulses input from external to port 2, bit 2.

Data is transmitted from bit 0 and the TDRE is set when the Transmit Shift Register (TSR) is "empty". More than 9th clock pulse of external are ignored.

When data transmit is selected to the clock output, the MPU produces transmit data and synchronous clock at TDRE flag clear.

### 2) Data receive

Data receive is enabled by setting RE bit. Port 2, bit 3 will be a serial input. The operating mode of data receive is decided by the TRCSR1 and the RMCR.

If the external clock input is selected, 8 external clock pulses and the synchronized receive data are input to port 2, bit 2 and bit 3 respectively. The MPU put receive data into the receive data shift register by this clock and set the RDRF flag at the termination of 8 bit



data receive. More than 9th clock pulse of external input are ignored. When RDRF is cleared, the MPU starts receiving the next data instantly. So, RDRF should be cleared with P<sub>22</sub> "High".

When data receive is selected with the clock output, 8 synchronous clocks are output to the external by setting RE bit. So re-

ceive data should be input from external synchronously with this clock. When the first byte data is received, the RDRF flag is set. After the second byte, receive operation is performed by sending the synchronous clock to the external after clearing the RDRF bit.

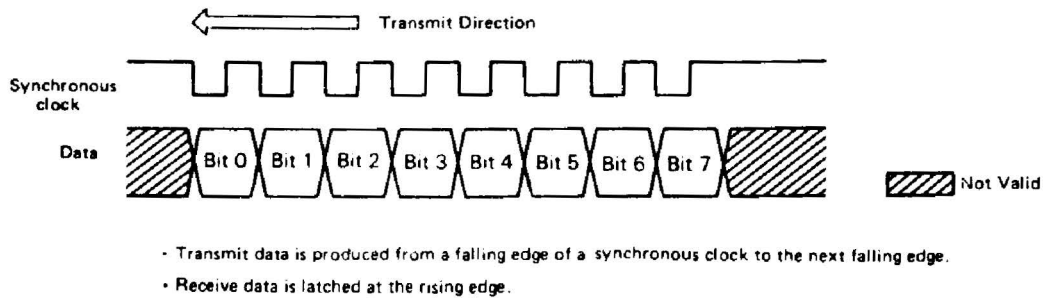


Figure 26 Clocked Synchronous Mode Format

• **Transmit/Receive Control Status Register (TRCSR1) (\$0011)**

The TRCSR1 is composed of 8 bits which are all readable. Bits 0 to 4 are also writable. This register is initialized to \$20 during reset. Each bit functions are as follows.

Transmit/Receive Control Status Register

|      |      |      |     |    |     |    |    |
|------|------|------|-----|----|-----|----|----|
| 7    | 6    | 5    | 4   | 3  | 2   | 1  | 0  |
| RDRF | ORFE | TDRE | RIE | RE | TIE | TE | WU |

\$0011

**Bit 0 WU Wake-up**

In a typical multi-processor configuration, the software protocol provides the destination address at the first byte of the message. In order to make uninterested MPU ignore the remaining message, a wake-up function is available. By this, uninterested MPU can inhibit all further receive processing till the next message starts.

Then wake-up function is triggered by consecutive 1's with 1 frame length. The software protocol should provide the idle time between messages.

By setting this bit, the MPU stops data receive till the next message. The receive of consecutive "1" with one frame length wakes up and clears this bit by hardware and then the MPU restarts receive operation. However, the RE flag should be already set before setting this bit. In the clocked synchronous mode WU is not available, so this bit should not be set.

**Bit 1 TE Transmit Enable**

When this bit is set, transmit data will appear at port 2, bit 4 after one frame preamble in asynchronous mode, while in clocked synchronous mode it appears immediately. This is executed regardless of the value of the corresponding DDR. When TE is cleared, the serial I/O doesn't affect port 2, bit 4.

**Bit 2 TIE Transmit Interrupt Enable**

When this bit is set, an internal interrupt (IRQ<sub>3</sub>) is enabled when TDRE (bit 5) is set. When cleared, the interrupt is inhibited.

**Bit 3 RE Receive Enable**

When set, a signal is input to the receiver from port 2, bit 3 regardless of the value of the DDR. When RE is cleared, the serial I/O doesn't affect port 2, bit 3.

**Bit 4 RIE Receive Interrupt Enable**

When this bit is set, an internal interrupt (IRQ<sub>2</sub>) is enabled when RDRF (bit 7) or ORFE (bit 6) is set. When cleared, the interrupt is inhibited.

**Bit 5 TDRE Transmit Data Register Empty**

TDRE is set by hardware when the TDR is transferred to the Transmit Shift Register in the asynchronous mode, while in clocked synchronous mode when the TDSR is "empty". This bit is cleared by reading the TRCSR1 or TRCSR2 and writing new transmit data to the TDR when TDRE=1. TDRE is set to "1" during reset.

**Bit 6 ORFE Overrun Framing Error**

ORFE is set by hardware when an overrun or a framing error is generated (during data-receive only). An overrun error occurs when new receive data is ready to be transferred to the RDR during RDRF still being set. A framing error occurs when a stop bit is "0". But in clocked synchronous mode, this bit is not affected. This bit is cleared by reading the TRCSR1 or TRCSR2, and the RDR, when RDRF=1. ORFE is cleared during reset.

**Bit 7 RDRF Receive Data Register Full**

RDRF is set by hardware when data is received normally and transferred from the Receive Shift Register (RSR) to the RDR. This bit is cleared by reading TRCSR1 or TRCSR2, and the RDR, when RDRF=1. This bit is cleared during reset.

• **Transmit Rate/Mode Control Register (RMCR)**

The RMCR controls the following serial I/O:

- Baud Rate
- Data Format
- Clock source
- Port 2, Bit 2 Function
- Operation Mode

All bits are readable/writable. Bit 0 to 5 of the RMCR are cleared during reset.

Transfer Rate/Mode Control Register

|   |   |     |     |     |     |     |     |
|---|---|-----|-----|-----|-----|-----|-----|
| 7 | 6 | 5   | 4   | 3   | 2   | 1   | 0   |
| - | - | SS2 | CC2 | CC1 | CC0 | SS1 | SS0 |

\$0010

**Bit 0 SS0**

**Bit 1 SS1** Speed Select

**Bit 5 SS2**

These bits control the baud rate used for the SCI. Table 7 lists the available baud rates. The timer 1 FRC (SS2=0) and the timer 2 up counter (SS2=1) provide the internal clock to the SCI. When selecting the timer 2 as a baud rate clock source, it functions as a baud rate generator. The timer 2 generates the baud rate listed in Table 8 depending on the value of the TCONR.

(Note) When operating the SCI with internal clock, do not perform write operation to the timer/counter which is the



**Table 7 SCI Bit Times and Transfer Rates**

**(1) Asynchronous Mode**

| SS2 | SS1 | SS0 | XTAL     | 2.4576MHz            | 4.0MHz                 | 4.9152MHz              |
|-----|-----|-----|----------|----------------------|------------------------|------------------------|
|     |     |     | E        | 614.4kHz             | 1.0MHz                 | 1.2288MHz              |
| 0   | 0   | 0   | E = 16   | 26 $\mu$ s/38400Baud | 16 $\mu$ s/62500Baud   | 13 $\mu$ s/76800Baud   |
| 0   | 0   | 1   | E = 128  | 208 $\mu$ s/4800Baud | 128 $\mu$ s/7812.5Baud | 104 $\mu$ s/9600Baud   |
| 0   | 1   | 0   | E = 1024 | 1.67ms/600Baud       | 1.024ms/976.6Baud      | 833.3 $\mu$ s/1200Baud |
| 0   | 1   | 1   | E = 4096 | 6.67ms/150Baud       | 4.096ms/244.1Baud      | 3.333ms/300Baud        |
| 1   | —   | —   | —        | *                    | *                      | *                      |

\* When SS2 is "1", Timer 2 provides SCI clocks. The baud rate is shown as follows with the TCONR as N.

$$\text{Baud Rate} = \frac{f}{32(N+1)} \quad \left( \begin{array}{l} f: \text{input clock frequency to the} \\ \text{timer 2 counter} \\ N = 0 \sim 255 \end{array} \right)$$

**(2) Clocked Synchronous Mode\***

| SS2 | SS1 | SS0 | XTAL    | 4.0 MHz         | 6.0 MHz          | 8.0 MHz         | 12.0 MHz          |
|-----|-----|-----|---------|-----------------|------------------|-----------------|-------------------|
|     |     |     | E       | 1.0 MHz         | 1.5 MHz          | 2.0 MHz         | 3.0 MHz           |
| 0   | 0   | 0   | E = 2   | 2 $\mu$ s/bit   | 1.33 $\mu$ s/bit | 1 $\mu$ s/bit   | 0.667 $\mu$ s/bit |
| 0   | 0   | 1   | E = 16  | 16 $\mu$ s/bit  | 10.7 $\mu$ s/bit | 8 $\mu$ s/bit   | 5.33 $\mu$ s/bit  |
| 0   | 1   | 0   | E = 128 | 128 $\mu$ s/bit | 85.3 $\mu$ s/bit | 64 $\mu$ s/bit  | 42.7 $\mu$ s/bit  |
| 0   | 1   | 1   | E = 512 | 512 $\mu$ s/bit | 341 $\mu$ s/bit  | 256 $\mu$ s/bit | 171 $\mu$ s/bit   |
| 1   | —   | —   | —       | **              | **               | **              | **                |

\* Bit rates in the case of internal clock operation. In the case of external clock operation, the external clock is operatable up to DC - 1/2 system clock.

\*\* The bit rate is shown as follows with the TCONR as N.

$$\text{Bit Rate } (\mu\text{s/bit}) = \frac{4(N+1)}{f} \quad \left( \begin{array}{l} f: \text{input clock frequency to the} \\ \text{timer 2 counter} \\ N = 0 \sim 255 \end{array} \right)$$

**Table 8 Baud Rate and Time Constant Register Example**

| Baud Rate (Baud) | XTAL | 2.4576MHz | 3.6864MHz | 4.0MHz | 4.9152MHz | 8.0MHz |
|------------------|------|-----------|-----------|--------|-----------|--------|
| 110              |      | 21*       | 32*       | 35*    | 43*       | 70*    |
| 150              |      | 127       | 191       | 207    | 255       | 51*    |
| 300              |      | 63        | 95        | 103    | 127       | 207    |
| 600              |      | 31        | 47        | 51     | 63        | 103    |
| 1200             |      | 15        | 23        | 25     | 31        | 51     |
| 2400             |      | 7         | 11        | 12     | 15        | 25     |
| 4800             |      | 3         | 5         | —      | 7         | 12     |
| 9600             |      | 1         | 2         | —      | 3         | —      |
| 19200            |      | 0         | —         | —      | 1         | —      |
| 38400            |      | —         | —         | —      | 0         | —      |

\* E/8 clock is input to the timer 2 up counter and E clock otherwise.

**Table 9 SCI Format and Clock Source Control**

| CC2 | CC1 | CC0 | Format     | Mode                | Clock Source | Port 2, Bit 2 | Port 2, Bit 3   | Port 2, Bit 4 |
|-----|-----|-----|------------|---------------------|--------------|---------------|---|---------------|
| 0   | 0   | 0   | 8-bit data | Clocked Synchronous | External     | Input         | When the TRCSR1, RE bit is "1", bit 3 is used as a serial input.  |               |
| 0   | 0   | 1   | 8-bit data | Asynchronous        | Internal     | Not Used**    |   |               |
| 0   | 1   | 0   | 8-bit data | Asynchronous        | Internal     | Output*       |   |               |
| 0   | 1   | 1   | 8-bit data | Asynchronous        | External     | Input         |   |               |
| 1   | 0   | 0   | 8-bit data | Clocked Synchronous | Internal     | Output        | When the TRCSR1, TE bit is "1", bit 4 is used as a serial output. |               |
| 1   | 0   | 1   | 7-bit data | Asynchronous        | Internal     | Not Used**    |   |               |
| 1   | 1   | 0   | 7-bit data | Asynchronous        | Internal     | Output*       |   |               |
| 1   | 1   | 1   | 7-bit data | Asynchronous        | External     | Input         |   |               |

\* Clock output regardless of the TRCSR1, bit RE and TE.

\*\* Not used for the SCI.



clock source of the SCI.

- Bit 2 **CC0**
- Bit 3 **CC1** Clock Control/Format Select\*
- Bit 4 **CC2**

These bits control the data format and the clock source (refer to Table 9).

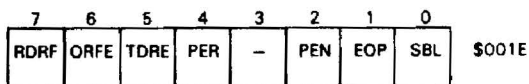
\* CC0, CC1 and CC2 are cleared during reset and the MPU goes to the clocked synchronous mode of the external clock operation. Then the MPU automatically set port 2, bit 2 into the clock input state. When using port 2, bit 2 as an output port, the DDR of port 2 should be set to "1" and CC1 and CC0 to "0" and "1" respectively.

- Bit 6 Not Used.
- Bit 7 Not Used

• **Transmit/Receive Control Status Register 2 (TRCSR2)**

The TRCSR2 is a 7-bit register which can select a data format in the asynchronous mode. The upper 3 bits are the same address as the TRCSR1. Therefore, the RDRF, ORFE and TDRE can be read by either the TRCSR1 or TRCSR2. Bits 0 to 2 of the TRCSR2 are used for read/write. Bits 4 to 7 are used only for read.

Transmit/Receive Control Status Register 2



Bit 0 **SBL Stop Bit Length**

This bit selects the stop bit length in the asynchronous mode.

If this bit is "0", the stop bit is 1-bit. If "1", the stop bit is 2-bit. This bit is cleared during reset.

Bit 1 **EOP Even/Odd Parity**

This bit selects the parity generated and checked when the PEN is "1". If this bit is "0", the parity is even. If "1", it is odd. This bit is cleared during reset.

Bit 2 **PEN Parity Enable**

This bit decides whether the parity bit should be generated and checked in the asynchronous mode or not. If this bit is "0", the parity bit is neither generated nor checked. If "1", it is generated and checked. This bit is cleared during reset.

The 3 bits above do not affect the SCI operation in the clocked synchronous mode.

Bit 3 Not Used

Bit 4 **PER Parity Error**

This bit is set when the PEN is "1" and a parity error occurs. It is cleared by reading the RDR after reading the TRCSR2, when PER = 1.

Bit 5 **TDRE**

Transmit Data Register Empty

Bit 6 **ORFE**

Overrun/Framing Error

Bit 7 **RDRF**

Receive Data Register Full

\* Each flag of the TDRE, ORFE, and RDRF can be read from either the TRCSR1 or TRCSR2.

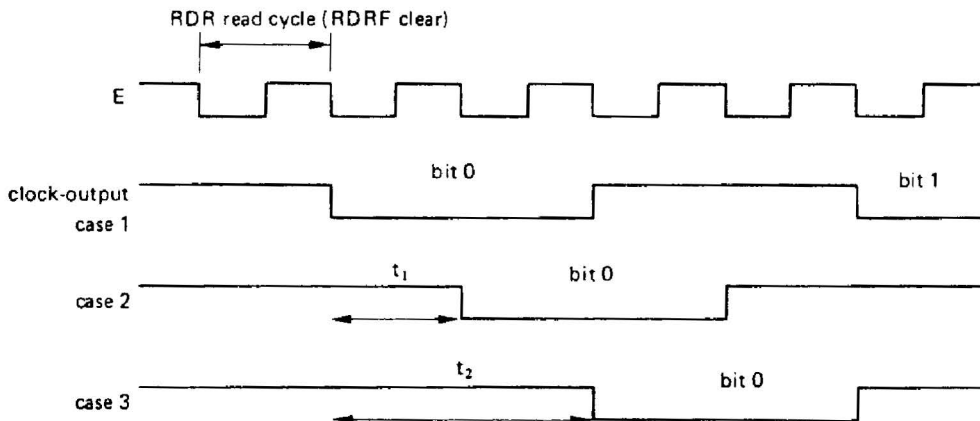


■ **PRECAUTION 1**

In the synchronous clocked receive operation with clock-output, there are three cases for clock pulse timing after RDRF clear as shown below.

Please consider above in designing system, since transmitting receiving time is not uniform.

The clock-output of case 1 or case 2 is determined by "1" or "0" of SCI internal operation clock of RDRF clearing cycle. In addition, in the case of low voltage operation ( $V_{CC} < 4.5V$ ), the clock-output of case 1 may transfer to case 3.



(note) When bit rate is  $E/2$ ,  $t_1 = E$ , and  $t_2 = 2E$ .  
 $E/16$ ,  $t_1 = 8E$ ,  $t_2 = 16E$ .  
 $E/128$ ,  $t_1 = 64E$ ,  $t_2 = 128E$ .  
 $E/512$ ,  $t_1 = 256E$ ,  $t_2 = 512E$ .

Diagram for Precaution 1



## ■ PRECAUTION 2

When transmitting through clock-synchronous serial communication interface, TE bit should not be cleared with TDRE of TRCSR (\$11) is "0".

The TDRE set and clear conditions of SCI are shown as follows.

|      | Set condition   | Clear condition   |
|------|---|---|
| TDRE | <ol style="list-style-type: none"> <li>1. TDR → transmit shift register (asynchronous)</li> <li>2. Transmit shift register is empty. (clock-synchronous)</li> <li>3. RES = 0</li> </ol> | When writing to TDR after TRCSR read, with TDRE = 1, TDRE is cleared. |

If transmit data is written to TDR, and then TE bit is cleared with TDRE = 0 to stop transmitting, TDRE remains "0".

In this case, even if TE bit is set and transmit data is written again, the TDR data is not transmitted.

Please note that TE bit must be cleared after the last data has been transmitted.

(This caution is not applied to asynchronous serial communication interface.)

## ■ TIMER, SCI STATUS FLAG

Table 10 shows the set and reset conditions of each status flag in the timer 1, timer 2 and SCI.

Table 10 Timer 1, Timer 2 and SCI Status Flag

|         |         | Set Condition  | Clear Condition  |
|---------|---------|--|--|
| P6CSR   | IS FLAG | Falling edge input to P <sub>54</sub> (IS)   | <ol style="list-style-type: none"> <li>1. Read the P6CSR then read or write the PORT6, when IS FLAG = 1</li> <li>2. RES = 0</li> </ol>           |
| Timer 1 | ICF     | FRC → ICR by Rising or Falling edge input to P <sub>20</sub><br>(Selecting with the IEDG bit)  | <ol style="list-style-type: none"> <li>1. Read the TCSR1 or TCSR2 then ICRH, when ICF = 1</li> <li>2. RES = 0</li> </ol>                         |
|         | OCF1    | OCR1 = FRC   | <ol style="list-style-type: none"> <li>1. Read the TCSR1 or TCSR2 then write to the OCR1H or OCR1L, when OCF1 = 1</li> <li>2. RES = 0</li> </ol> |
|         | OCF2    | OCR2 = FRC   | <ol style="list-style-type: none"> <li>1. Read the TCSR2 then write to the OCR2H or OCR2L, when OCF2 = 1</li> <li>2. RES = 0</li> </ol>          |
|         | TOF     | FRC = \$FFFF + 1 cycle   | <ol style="list-style-type: none"> <li>1. Read the TCSR1 then FRCH, when TOF = 1</li> <li>2. RES = 0</li> </ol>                                  |
| Timer 2 | CMF     | T2CNT = TCONR  | <ol style="list-style-type: none"> <li>1. Write "0" to CMF, when CMF = 1</li> <li>2. RES = 0</li> </ol>  |
| SCI     | RDRF    | Receive Shift Register → RDR   | <ol style="list-style-type: none"> <li>1. Read the TRCSR1 or TRCSR2 then RDR, when RDRF = 1</li> <li>2. RES = 0</li> </ol>                       |
|         | ORFE    | <ol style="list-style-type: none"> <li>1. Framing Error (Asynchronous Mode) Stop Bit = 0</li> <li>2. Overrun Error (Asynchronous Mode) Receive Shift Register → RDR when RDRF = 1</li> </ol>     | <ol style="list-style-type: none"> <li>1. Read the TRCSR1 or TRCSR2 then RDR, when ORFE = 1</li> <li>2. RES = 0</li> </ol>                       |
|         | TDRE    | <ol style="list-style-type: none"> <li>1. Asynchronous Mode TDR → Transmit Shift Register</li> <li>2. Clocked Synchronous Mode Transmit Shift Register is "empty"</li> <li>3. RES = 0</li> </ol> | Read the TRCSR1 or TRCSR2 then write to the TDR, when TDRE = 1   |
|         | PER     | Parity when PEN = 1  | <ol style="list-style-type: none"> <li>1. Read the TRCSR2 then RDR, when PER = 1</li> <li>2. RES = 0</li> </ol>                                  |

(Note) → ; Transfer = ; equal

ICRH: Upper byte of ICR  
OCR1H: Upper byte of OCR1  
OCR2H: Upper byte of OCR2

OCR1L: Lower byte of OCR1  
OCR2L: Lower byte of OCR2  
FRCH: Upper byte of FRC



## ■ LOW POWER DISSIPATION MODE

The HD6303Y provides two low power dissipation modes; sleep and standby.

### ● Sleep Mode

The MPU goes to the sleep mode by SLP instruction execution. In the sleep mode, the CPU stops its operation, while the registers' contents are retained. In this mode, the peripherals except the CPU such as timers, SCI, etc. continue their functions. The power dissipation of sleep-condition is one fourth that of operating condition.

The MPU returns from this mode by an interrupt,  $\overline{RES}$  or  $\overline{STBY}$ ; it goes to the reset state by  $\overline{RES}$  and the standby mode by  $\overline{STBY}$ . When the CPU acknowledges an interrupt request, it cancels the sleep mode, returns to the operation mode and branches to the interrupt routine. When the CPU masks this interrupt, it cancels the sleep mode and executes the next instruction. However, for example, if the timer 1 or 2 prohibits a timer interrupt, the CPU doesn't cancel the sleep mode because of no interrupt request.

This sleep mode is effective to reduce the power dissipation for a system with no need of the HD6303Y's consecutive operation.

### ● Standby Mode

The MPU goes to the standby mode with the  $\overline{STBY}$  "Low" or by clearing the  $\overline{STBY}$  flag. In this mode, the HD6303Y stops all the clocks and goes to the reset state. In this mode, the power dissipation is reduced to several  $\mu A$ . During standby, all pins, except the power supply ( $V_{CC}$ ,  $V_{SS}$ ), the  $\overline{STBY}$ ,  $\overline{RES}$  and XTAL (which outputs "0"), go to the high impedance state. In this mode, power ( $V_{CC}$ ) is supplied to the HD6303Y, and the contents of RAM is retained. The MPU returns from this mode during reset. When the MPU goes to the standby mode with  $\overline{STBY}$  "Low", it will restart at the timing shown in Fig. 27(a). When the MPU goes to the standby mode by clearing the  $\overline{STBY}$  flag, it will restart only by keeping the  $\overline{RES}$  "Low" for longer than the oscillating stabilization time. (Fig. 27(b))

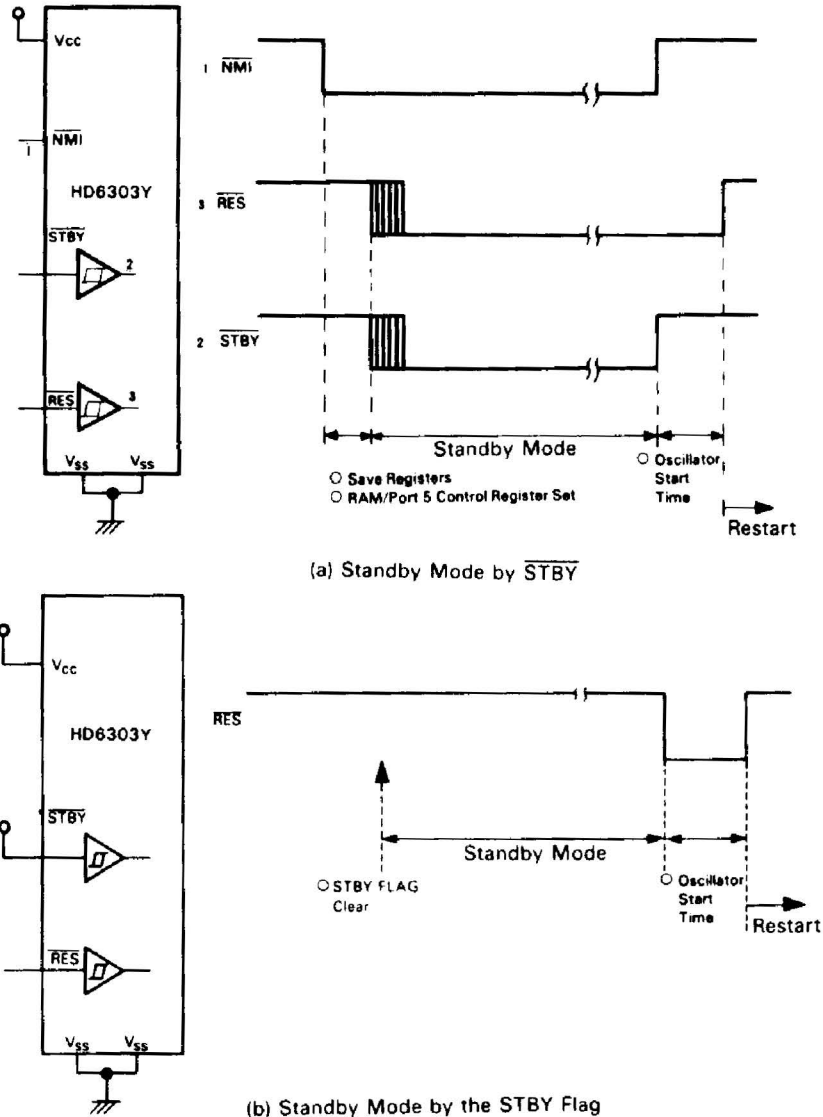


Figure 27 Standby Mode Timing



## ■ TRAP FUNCTION

The CPU generates an interrupt with the highest priority (TRAP) when fetching an undefined instruction or an instruction from non-memory space. The TRAP prevents the system-burst caused by noise or a program error.

### ● Op Code Error

When fetching an undefined op code, the CPU saves registers as well as a normal interrupt and branches to the TRAP (\$FFEE, \$FFEF). This has the priority next to reset.

### ● Address Error

When an instruction fetch is made from the address of internal register, the MPU generates an interrupt as well as an op code error. But on the system with no memory in its external memory area, this function is not applicable if an instruction fetch is made from the external non-memory area. Addresses where an address error occurs are from \$0000 to \$0027.

This function is available only for an instruction fetch and is not applicable to the access of normal data read/write.

(Note) The TRAP interrupt provides a retry function differently from other interrupts. This is a program flow return to the address where the TRAP occurs when a sequence returns to a main routine from the TRAP interrupt routine by RTI. The retry can prevent the system burst caused by noise, etc.

However, if another TRAP occurs, the program repeats the TRAP interrupt forever, so the consideration is necessary in programming.

## ■ INSTRUCTION SET

The HD6303Y provides object code upward compatible with the HD6801 to utilize all instruction set of the HMCS6800. It also reduces the execution times of key instructions for throughput improvement.

Bit manipulation instruction, change instruction of the index register and accumulator and sleep instruction are also added.

The followings are explained here.

- CPU Programming Model (refer to Fig. 28)
- Addressing Mode
- Accumulator and Memory Manipulation Instruction (refer to Table 11)
- New Instruction
- Index Register and Stack Manipulation Instruction (refer to Table 12)
- Jump and Branch Instruction (refer to Table 13)
- Condition Code Register Manipulation (refer to Table 14)
- Op Code Map (refer to Table 15)

### ● Programming Model

Fig. 28 depicts the HD6303Y programming model. The double accumulator D consists of accumulator A and B, so when using the accumulator D, the contents of A and B are destroyed.

### ● CPU Addressing Mode

The HD6303Y provides 7 addressing modes. The addressing mode is determined by an instruction type and code. Tables 11 through 15 show addressing modes of each instruction with the execution times counted by the machine cycle.

When the clock frequency is 4MHz, the machine cycle time becomes microseconds directly.

### Accumulator (ACCX) Addressing

Only an accumulator is addressed and the accumulator A or B is selected. This is a one-byte instruction.

### Immediate Addressing

This addressing locates a data in the second byte of an instruction. However, LDS and LDX locate a data in the second and third byte exceptionally. This addressing is a 2 or 3-byte instruction.

### Direct Addressing

In this addressing mode, the second byte of an instruction shows

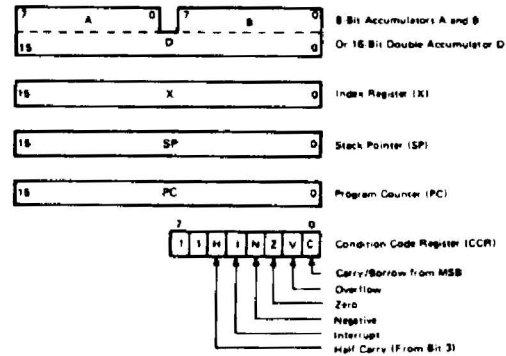


Figure 28 CPU Programming Model

the address where a data is stored. 256 bytes (\$0 through \$255) can be addressed directly. Execution times can be reduced by storing data in this area so it is recommended to make it RAM for users' data area in configuring a system. This is a 2-byte instruction, while 3 byte with regard to AIM, OIM, EIM and TIM.

### Extended Addressing

In this mode, the second byte shows the upper 8 bit of the data stored address and the third byte the lower 8 bit. This indicates the absolute address of 3 byte instruction in the memory.

### Indexed Addressing

The second byte of an instruction and the lower 8 bit of the index register are added in this mode. As for AIM, OIM, EIM and TIM, the third byte of an instruction and the lower 8 bits of the index register are added.

This carry is added to the upper 8 bit of the index register and the result is used for addressing the memory. The modified address is retained in the temporary address register, so the contents of the index register doesn't change. This is a 2-byte instruction except AIM, OIM, EIM and TIM (3-byte instruction).

### Implied Addressing

An instruction itself specifies the address. This is, the instruction addresses a stack pointer, index register, etc. This is a one-byte instruction.

### Relative Addressing

The second byte of an instruction and the lower 8 bits of the program counter are added. The carry or borrow is added to the upper 8 bit. So addressing from -126 to +129 byte of the current instruction is enabled. This is a 2-byte instruction.

### (Note) CLI, SEI Instructions and Interrupt Operation

When accepting the IRQ at a preset timing with CLI and SEI instructions, more than 2 cycles are necessary between the CLI and SEI instructions. For example, the following program (a)(b) don't accept the IRQ but (c) accepts it.

|     |     |     |
|-----|-----|-----|
| .   | .   | .   |
| .   | .   | .   |
| .   | .   | .   |
| .   | .   | CLI |
| CLI | CLI | NOP |
| SEI | NOP | NOP |
| .   | SEI | SEI |
| .   | .   | .   |
| .   | .   | .   |
| .   | .   | .   |
| .   | .   | .   |
| (a) | (b) | (c) |

The same thing can be said to the TAP instruction instead of the CLI and SEI instructions.



Table 11 Accumulator, Memory Manipulation Instructions

| Operations               | Mnemonic | Addressing Modes |   |   |        |   |   |       |   |   |        |   |   | Boolean/<br>Arithmetic Operation | Condition Code Register |                          |   |   |   |   |   |   |   |
|--------------------------|----------|------------------|---|---|--------|---|---|-------|---|---|--------|---|---|----------------------------------|-------------------------|--------------------------|---|---|---|---|---|---|---|
|                          |          | IMMED            |   |   | DIRECT |   |   | INDEX |   |   | EXTEND |   |   |                                  | IMPLIED                 |                          |   | 5 | 4 | 3 | 2 | 1 | 0 |
|                          |          | OP               | ~ | # | OP     | ~ | # | OP    | ~ | # | OP     | ~ | # |                                  | OP                      | ~                        | #   | H | I | N | Z | V | C |
| Add                      | ADDA     | 8B               | 2 | 2 | 9B     | 3 | 2 | AB    | 4 | 2 | BB     | 4 | 3 |                                  |                         | A + M → A                | 1   | • | 1 | 1 | 1 | 1 |   |
|                          | ADDB     | CB               | 2 | 2 | DB     | 3 | 2 | EB    | 4 | 2 | FB     | 4 | 3 |                                  |                         | B + M → B                | 1   | • | 1 | 1 | 1 | 1 |   |
| Add Double               | ADDD     | C3               | 3 | 3 | D3     | 4 | 2 | E3    | 5 | 2 | F3     | 5 | 3 |                                  |                         | A ← B + M, M + 1 → A ← B | •   | • | 1 | 1 | 1 | 1 |   |
| Add Accumulators         | ABA      |                  |   |   |        |   |   |       |   |   |        |   |   | 1B                               | 1                       | 1                        | A + B → A   | 1 | • | 1 | 1 | 1 |   |
| Add With Carry           | ADCA     | 89               | 2 | 2 | 99     | 3 | 2 | A9    | 4 | 2 | B9     | 4 | 3 |                                  |                         | A + M + C → A            | 1   | • | 1 | 1 | 1 | 1 |   |
|                          | ADCB     | C9               | 2 | 2 | D9     | 3 | 2 | E9    | 4 | 2 | F9     | 4 | 3 |                                  |                         | B + M + C → B            | 1   | • | 1 | 1 | 1 | 1 |   |
| AND                      | ANDA     | 84               | 2 | 2 | 94     | 3 | 2 | A4    | 4 | 2 | B4     | 4 | 3 |                                  |                         | A ← M ← A                | •   | • | 1 | 1 | R | • |   |
|                          | ANDB     | C4               | 2 | 2 | D4     | 3 | 2 | E4    | 4 | 2 | F4     | 4 | 3 |                                  |                         | B ← M ← B                | •   | • | 1 | 1 | R | • |   |
| Bit Test                 | BIT A    | 85               | 2 | 2 | 95     | 3 | 2 | A5    | 4 | 2 | B5     | 4 | 3 |                                  |                         | A ← M                    | •   | • | 1 | 1 | R | • |   |
|                          | BIT B    | C5               | 2 | 2 | D5     | 3 | 2 | E5    | 4 | 2 | F5     | 4 | 3 |                                  |                         | B ← M                    | •   | • | 1 | 1 | R | • |   |
| Clear                    | CLR      |                  |   |   |        |   |   | 6F    | 5 | 2 | 7F     | 5 | 3 |                                  |                         | 00 ← M                   | •   | • | R | S | R | R |   |
|                          | CLRA     |                  |   |   |        |   |   |       |   |   |        |   |   | 4F                               | 1                       | 1                        | 00 ← A  | • | • | R | S | R | R |
|                          | CLRB     |                  |   |   |        |   |   |       |   |   |        |   |   | 5F                               | 1                       | 1                        | 00 ← B  | • | • | R | S | R | R |
| Compare                  | CMPA     | 81               | 2 | 2 | 91     | 3 | 2 | A1    | 4 | 2 | B1     | 4 | 3 |                                  |                         | A ← M                    | •   | • | 1 | 1 | 1 | 1 |   |
|                          | CMPB     | C1               | 2 | 2 | D1     | 3 | 2 | E1    | 4 | 2 | F1     | 4 | 3 |                                  |                         | B ← M                    | •   | • | 1 | 1 | 1 | 1 |   |
| Compare Accumulators     | CBA      |                  |   |   |        |   |   |       |   |   |        |   |   | 11                               | 1                       | 1                        | A ← B   | • | • | 1 | 1 | 1 | 1 |
| Complement, 1's          | COM      |                  |   |   |        |   |   | 63    | 6 | 2 | 73     | 6 | 3 |                                  |                         | M ← M                    | •   | • | 1 | 1 | R | S |   |
|                          | COMA     |                  |   |   |        |   |   |       |   |   |        |   |   | 43                               | 1                       | 1                        | A ← A   | • | • | 1 | 1 | R | S |
|                          | COMB     |                  |   |   |        |   |   |       |   |   |        |   |   | 53                               | 1                       | 1                        | B ← B   | • | • | 1 | 1 | R | S |
| Complement, 2's (Negate) | NEG      |                  |   |   |        |   |   | 60    | 6 | 2 | 70     | 6 | 3 |                                  |                         | 00 ← M ← M               | •   | • | 1 | 1 | ⊕ | ? |   |
|                          | NEGA     |                  |   |   |        |   |   |       |   |   |        |   |   | 40                               | 1                       | 1                        | 00 ← A ← A  | • | • | 1 | 1 | 1 | 2 |
|                          | NEGB     |                  |   |   |        |   |   |       |   |   |        |   |   | 50                               | 1                       | 1                        | 00 ← B ← B  | • | • | 1 | 1 | ⊕ | 3 |
| Decimal Adjust, A        | DAA      |                  |   |   |        |   |   |       |   |   |        |   |   | 19                               | 2                       | 1                        | Converts binary add of BCD characters into BCD format | • | • | 1 | 1 | 1 | 3 |
| Decrement                | DEC      |                  |   |   |        |   |   | 6A    | 6 | 2 | 7A     | 6 | 3 |                                  |                         | M ← 1 → M                | •   | • | 1 | 1 | ⊖ | • |   |
|                          | DECA     |                  |   |   |        |   |   |       |   |   |        |   |   | 4A                               | 1                       | 1                        | A ← 1 → A   | • | • | 1 | 1 | ⊖ | • |
|                          | DECB     |                  |   |   |        |   |   |       |   |   |        |   |   | 5A                               | 1                       | 1                        | B ← 1 → B   | • | • | 1 | 1 | ⊖ | • |
| Exclusive OR             | EORA     | 8B               | 2 | 2 | 9B     | 3 | 2 | AB    | 4 | 2 | BB     | 4 | 3 |                                  |                         | A ⊕ M → A                | •   | • | 1 | 1 | R | • |   |
|                          | EORB     | C8               | 2 | 2 | D8     | 3 | 2 | E8    | 4 | 2 | F8     | 4 | 3 |                                  |                         | B ⊕ M → B                | •   | • | 1 | 1 | R | • |   |
| Increment                | INC      |                  |   |   |        |   |   | 6C    | 6 | 2 | 7C     | 6 | 3 |                                  |                         | M + 1 → M                | •   | • | 1 | 1 | ⊕ | • |   |
|                          | INCA     |                  |   |   |        |   |   |       |   |   |        |   |   | 4C                               | 1                       | 1                        | A + 1 → A   | • | • | 1 | 1 | ⊕ | • |
|                          | INCB     |                  |   |   |        |   |   |       |   |   |        |   |   | 5C                               | 1                       | 1                        | B + 1 → B   | • | • | 1 | 1 | ⊕ | • |
| Load Accumulator         | LDAA     | 86               | 2 | 2 | 96     | 3 | 2 | A6    | 4 | 2 | B6     | 4 | 3 |                                  |                         | M → A                    | •   | • | 1 | 1 | R | • |   |
|                          | LDAB     | C6               | 2 | 2 | D6     | 3 | 2 | E6    | 4 | 2 | F6     | 4 | 3 |                                  |                         | M → B                    | •   | • | 1 | 1 | R | • |   |
| Load Double Accumulator  | LDD      | CC               | 3 | 3 | DC     | 4 | 2 | EC    | 5 | 2 | FC     | 5 | 3 |                                  |                         | M + 1 → B, M → A         | •   | • | 1 | 1 | R | • |   |
| Multiply Unsigned        | MUL      |                  |   |   |        |   |   |       |   |   |        |   |   | 3D                               | 7                       | 1                        | A × B → A ← B   | • | • | • | • | • | ⊕ |
| OR, Inclusive            | ORAA     | 8A               | 2 | 2 | 9A     | 3 | 2 | AA    | 4 | 2 | BA     | 4 | 3 |                                  |                         | A ← M ← A                | •   | • | 1 | 1 | R | • |   |
|                          | ORAB     | CA               | 2 | 2 | DA     | 3 | 2 | EA    | 4 | 2 | FA     | 4 | 3 |                                  |                         | B ← M ← B                | •   | • | 1 | 1 | R | • |   |
| Push Data                | PSHA     |                  |   |   |        |   |   |       |   |   |        |   |   | 36                               | 4                       | 1                        | A ← Map, SP ← 1 → SP                                  | • | • | • | • | • | • |
|                          | PSHB     |                  |   |   |        |   |   |       |   |   |        |   |   | 37                               | 4                       | 1                        | B ← Map, SP ← 1 → SP                                  | • | • | • | • | • | • |
| Pull Data                | PULA     |                  |   |   |        |   |   |       |   |   |        |   |   | 32                               | 3                       | 1                        | SP + 1 → SP, Map → A                                  | • | • | • | • | • | • |
|                          | PULB     |                  |   |   |        |   |   |       |   |   |        |   |   | 33                               | 3                       | 1                        | SP + 1 → SP, Map → B                                  | • | • | • | • | • | • |
| Rotate Left              | ROL      |                  |   |   |        |   |   | 69    | 6 | 2 | 79     | 6 | 3 |                                  |                         | M ← M ← M                | •   | • | 1 | 1 | ⊕ | 1 |   |
|                          | ROLA     |                  |   |   |        |   |   |       |   |   |        |   |   | 49                               | 1                       | 1                        | A ← M ← M   | • | • | 1 | 1 | ⊕ | 1 |
|                          | ROLB     |                  |   |   |        |   |   |       |   |   |        |   |   | 59                               | 1                       | 1                        | B ← M ← M   | • | • | 1 | 1 | ⊕ | 1 |
| Rotate Right             | ROR      |                  |   |   |        |   |   | 66    | 6 | 2 | 76     | 6 | 3 |                                  |                         | M ← M ← M                | •   | • | 1 | 1 | ⊕ | 1 |   |
|                          | RORA     |                  |   |   |        |   |   |       |   |   |        |   |   | 46                               | 1                       | 1                        | A ← M ← M   | • | • | 1 | 1 | ⊕ | 1 |
|                          | RORB     |                  |   |   |        |   |   |       |   |   |        |   |   | 56                               | 1                       | 1                        | B ← M ← M   | • | • | 1 | 1 | ⊕ | 1 |

(Note) Condition Code Register will be explained in Note of Table 14.

(continued)



2





• **Additional Instruction**

In addition to the HD6801 instruction set, the HD6303Y prepares the following new instructions.

- AIM ..... (M)·(IMM) → (M)  
Executes "AND" operation to immediate data and the memory contents and stores its result in the memory.
- OIM ..... (M)+(IMM) → (M)  
Executes "OR" operation to immediate data and the memory contents and stores its result in the memory.
- EIM ..... (M)⊕(IMM) → (M)  
Executes "EOR" operation to immediate data and the memory contents and stores its result in the memory.

TIM ..... (M)·(IMM)

Executes "AND" operation to immediate data and changes the relative flag of the condition code register.

These are the 3-byte instructions; the first byte is op code, the second immediate data and the third address modifier.

XGDX ..... (ACCD)←(IX)

Exchanges the contents of accumulator and the index register.

SLP

Goes to the sleep mode. Refer to "LOW POWER DISSIPATION MODE" for more details of the sleep mode.

Table 12 Index Register, Stack Manipulation Instructions

| Pointer Operations    | Mnemonic | Addressing Modes |   |        |    |       |   |        |   |         |    | Boolean/<br>Arithmetic Operation | Condition Code Register |    |   |   |  |   |   |   |   |   |   |
|-----------------------|----------|------------------|---|--------|----|-------|---|--------|---|---------|----|----------------------------------|-------------------------|----|---|---|--|---|---|---|---|---|---|
|                       |          | IMMED            |   | DIRECT |    | INDEX |   | EXTEND |   | IMPLIED |    |                                  | H                       | I  | N | Z   | V  | C |   |   |   |   |   |
|                       |          | OP               | # | OP     | #  | OP    | # | OP     | # | OP      | #  |                                  |                         |    |   |   |  |   |   |   |   |   |   |
| Compare Index Reg     | CPX      | 8C               | 3 | 3      | 9C | 4     | 2 | AC     | 5 | 2       | BC | 5                                | 3                       |    |   | X - M M + 1                                     | •  | • | • | • | • | • |   |
| Decrement Index Reg   | DEX      |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 09 | 1 | 1   | X - 1 - X  | • | • | • | • | • | • |
| Decrement Stack Ptr   | DES      |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 34 | 1 | 1   | SP - 1 - SP  | • | • | • | • | • | • |
| Increment Index Reg   | INX      |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 08 | 1 | 1   | X + 1 - X  | • | • | • | • | • | • |
| Increment Stack Ptr   | INS      |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 31 | 1 | 1   | SP + 1 - SP  | • | • | • | • | • | • |
| Load Index Reg        | LDX      | CE               | 3 | 3      | DE | 4     | 2 | EE     | 5 | 2       | FE | 5                                | 3                       |    |   | M → X <sub>H</sub> , (M + 1) → X <sub>L</sub>   | •  | • | • | • | • | • |   |
| Load Stack Ptr        | LDS      | 8E               | 3 | 3      | 9E | 4     | 2 | AE     | 5 | 2       | BE | 5                                | 3                       |    |   | M → SP <sub>H</sub> , (M + 1) → SP <sub>L</sub> | •  | • | • | • | • | • |   |
| Store Index Reg       | STX      |                  |   |        | DF | 4     | 2 | EF     | 5 | 2       | FF | 5                                | 3                       |    |   | X <sub>H</sub> → M, X <sub>L</sub> → (M + 1)    | •  | • | • | • | • | • |   |
| Store Stack Ptr       | STS      |                  |   |        | 9F | 4     | 2 | AF     | 5 | 2       | BF | 5                                | 3                       |    |   | SP <sub>H</sub> → M, SP <sub>L</sub> → (M + 1)  | •  | • | • | • | • | • |   |
| Index Reg → Stack Ptr | TXS      |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 35 | 1 | 1   | X - 1 - SP   | • | • | • | • | • | • |
| Stack Ptr → Index Reg | TSX      |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 30 | 1 | 1   | SP + 1 - X   | • | • | • | • | • | • |
| Add                   | ABX      |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 3A | 1 | 1   | B + X - X  | • | • | • | • | • | • |
| Push Data             | PSHX     |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 3C | 5 | 1   | X <sub>L</sub> → M <sub>sp</sub> , SP - 1 → SP<br>X <sub>H</sub> → M <sub>sp</sub> , SP - 1 → SP | • | • | • | • | • | • |
| Pull Data             | PULX     |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 38 | 4 | 1   | SP + 1 - SP, M <sub>sp</sub> → X <sub>H</sub><br>SP + 1 - SP, M <sub>sp</sub> → X <sub>L</sub>   | • | • | • | • | • | • |
| Exchange              | XGDX     |                  |   |        |    |       |   |        |   |         |    |                                  |                         | 18 | 2 | 1   | ACCD ← IX  | • | • | • | • | • | • |

(Note) Condition Code Register will be explained in Note of Table 14.





Table 14 Condition Code Register Manipulation Instructions

| Operations           | Mnemonic | Addressing Modes |   |   | Boolean Operation | Condition Code Register |   |   |   |   |   |   |
|----------------------|----------|------------------|---|---|-------------------|-------------------------|---|---|---|---|---|---|
|                      |          | IMPLIED          |   |   |                   | 5                       | 4 | 3 | 2 | 1 | 0 |   |
|                      |          | OP               | ~ | # |                   | H                       | I | N | Z | V | C |   |
| Clear Carry          | CLC      | 0C               | 1 | 1 | 0 → C             | •                       | • | • | • | • | • | R |
| Clear Interrupt Mask | CLI      | 0E               | 1 | 1 | 0 → I             | •                       | R | • | • | • | • | • |
| Clear Overflow       | CLV      | 0A               | 1 | 1 | 0 → V             | •                       | • | • | • | • | R | • |
| Set Carry            | SEC      | 0D               | 1 | 1 | 1 → C             | •                       | • | • | • | • | • | S |
| Set Interrupt Mask   | SEI      | 0F               | 1 | 1 | 1 → I             | •                       | S | • | • | • | • | • |
| Set Overflow         | SEV      | 08               | 1 | 1 | 1 → V             | •                       | • | • | • | • | S | • |
| Accumulator A → CCR  | TAP      | 06               | 1 | 1 | A → CCR           | ⑩                       |   |   |   |   |   |   |
| CCR → Accumulator A  | TPA      | 07               | 1 | 1 | CCR → A           | •                       | • | • | • | • | • | • |

LEGEND

- OP Operation Code (Hexadecimal)
- ~ Number of MCU Cycles
- M<sub>SP</sub> Contents of memory location pointed by Stack Pointer
- # Number of Program Bytes
- + Arithmetic Plus
- Arithmetic Minus
- Boolean AND
- + Boolean Inclusive OR
- ⊕ Boolean Exclusive OR
- M Complement of M
- Transfer into
- 0 Bit = Zero
- 00 Byte = Zero

CONDITION CODE SYMBOLS

- H Half-carry from bit 3 to bit 4
- I Interrupt mask
- N Negative (sign bit)
- Z Zero (byte)
- V Overflow, 2's complement
- C Carry/Borrow from/to bit 7
- R Reset Always
- S Set Always
- ‡ Set if true after test or clear
- Not Affected

(Note) Condition Code Register Notes: (Bit set if test is true and cleared otherwise)

- ① (Bit V) Test: Result = 10000000?
- ② (Bit C) Test: Result ≠ 00000000?
- ③ (Bit C) Test: BCD Character of high-order byte greater than 10? (Not cleared if previously set)
- ④ (Bit V) Test: Operand = 10000000 prior to execution?
- ⑤ (Bit V) Test: Operand = 01111111 prior to execution?
- ⑥ (Bit V) Test: Set equal to N ⊗ C = 1 after the execution of instructions
- ⑦ (Bit N) Test: Result less than zero? (Bit 15=1)
- ⑧ (All Bit) Load Condition Code Register from Stack.
- ⑨ (Bit I) Set when interrupt occurs. If previously set, a Non-Maskable Interrupt is required to exit the wait state.
- ⑩ (All Bit) Set according to the contents of Accumulator A.
- ⑪ (Bit C) Result of Multiplication Bit 7=1? (ACCB)

Table 15 OP-Code Map

| OP CODE |    |      |      |      | ACC A | ACC B | IND  | EXT DIR | ACCA or SP |      |      |      | ACCB or X |      |      |      |      |  |  |
|---------|----|------|------|------|-------|-------|------|---------|------------|------|------|------|-----------|------|------|------|------|--|--|
|         | HI | 0000 | 0001 | 0010 | 0011  | 0100  | 0101 | 0110    | 0111       | 1000 | 1001 | 1010 | 1011      | 1100 | 1101 | 1110 | 1111 |  |  |
| LO      | 0  | 1    | 2    | 3    | 4     | 5     | 6    | 7       | 8          | 9    | A    | B    | C         | D    | E    | F    |      |  |  |
| 0000    | 0  | SBA  | BRA  | TSX  | NEG   |       |      |         | SUB        |      |      |      | 0         |      |      |      |      |  |  |
| 0001    | 1  | NOP  | CBA  | BRN  | INS   | AIM   |      |         |            | CMP  |      |      |           | 1    |      |      |      |  |  |
| 0010    | 2  |      |      | BHI  | PULA  | OIM   |      |         |            | SBC  |      |      |           | 2    |      |      |      |  |  |
| 0011    | 3  |      |      | BLS  | PULB  | COM   |      |         |            | SUBD |      |      |           | ADDD |      |      |      |  |  |
| 0100    | 4  | LSRD |      |      | BCC   | DES   | LSR  |         |            |      | AND  |      |           |      | 4    |      |      |  |  |
| 0101    | 5  | ASLD |      |      | BCS   | TXS   | EIM  |         |            |      | BIT  |      |           |      | 5    |      |      |  |  |
| 0110    | 6  | TAP  | TAB  | BNE  | PSHA  | ROR   |      |         |            | LDA  |      |      |           | 6    |      |      |      |  |  |
| 0111    | 7  | TPA  | TBA  | BEQ  | PSHB  | ASR   |      |         |            | STA  |      |      |           | STA  |      |      |      |  |  |
| 1000    | 8  | INX  | XGDX | BVC  | PULX  | ASL   |      |         |            | EOR  |      |      |           | 8    |      |      |      |  |  |
| 1001    | 9  | DEX  | DAA  | BVS  | RTS   | ROL   |      |         |            | ADC  |      |      |           | 9    |      |      |      |  |  |
| 1010    | A  | CLV  | SLP  | BPL  | ABX   | DEC   |      |         |            | ORA  |      |      |           | A    |      |      |      |  |  |
| 1011    | B  | SEV  | ABA  | BMI  | RTI   | TIM   |      |         |            | ADD  |      |      |           | B    |      |      |      |  |  |
| 1100    | C  | CLC  |      |      | BGE   | PSHX  | INC  |         |            |      | CPX  |      |           |      | LDD  |      |      |  |  |
| 1101    | D  | SEC  |      |      | BLT   | MUL   | TST  |         |            |      | BSR  | JSR  |           |      |      | STD  |      |  |  |
| 1110    | E  | CLI  |      |      | BGT   | WAI   | JMP  |         |            |      | LDS  |      |           |      | LDX  |      |      |  |  |
| 1111    | F  | SEI  |      |      | BLE   | SWI   | CLR  |         |            |      | STS  |      |           |      | STX  |      |      |  |  |

UNDEFINED OP CODE

\* Only each instructions of AIM, OIM, EIM, TIM



■ CPU OPERATION

● CPU Instruction Flow

When operating, the CPU fetches an instruction from a memory and executes the required function. This sequence starts with  $\overline{RES}$  cancel and repeats itself limitlessly if not affected by a special instruction or a control signal. SWI, RTI, WAI and SLP instructions change this operation, while  $\overline{NMI}$ ,  $\overline{IRQ_1}$ ,  $\overline{IRQ_2}$ ,  $\overline{IRQ_3}$ ,  $\overline{HALT}$  and  $\overline{STBY}$  control it. Fig. 29 gives the CPU mode transition and Fig. 30 the CPU system flow chart. Table 16 shows CPU operating states

and port states.

● Operation at Each Instruction Cycle

Table 17 shows the operation at each instruction cycle. By the pipeline control of the HD6303Y, MULT, PUL, DAA and XGDX instructions, etc. prefetch the next instruction. So attention is necessary to the counting of the instruction cycles because it is different from the usual one—from op code fetch to the next instruction op code.

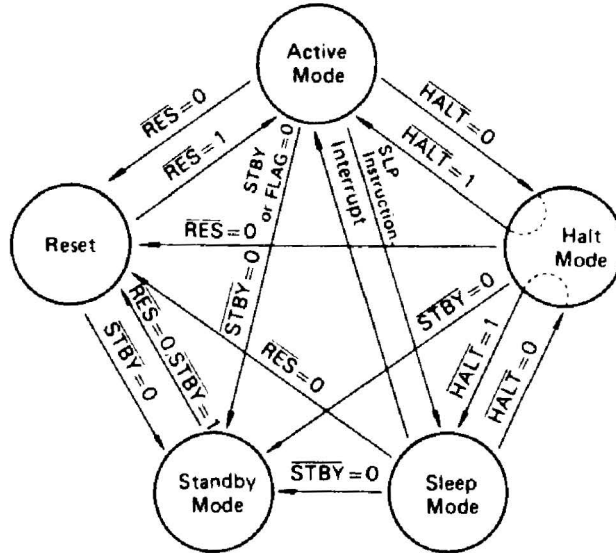


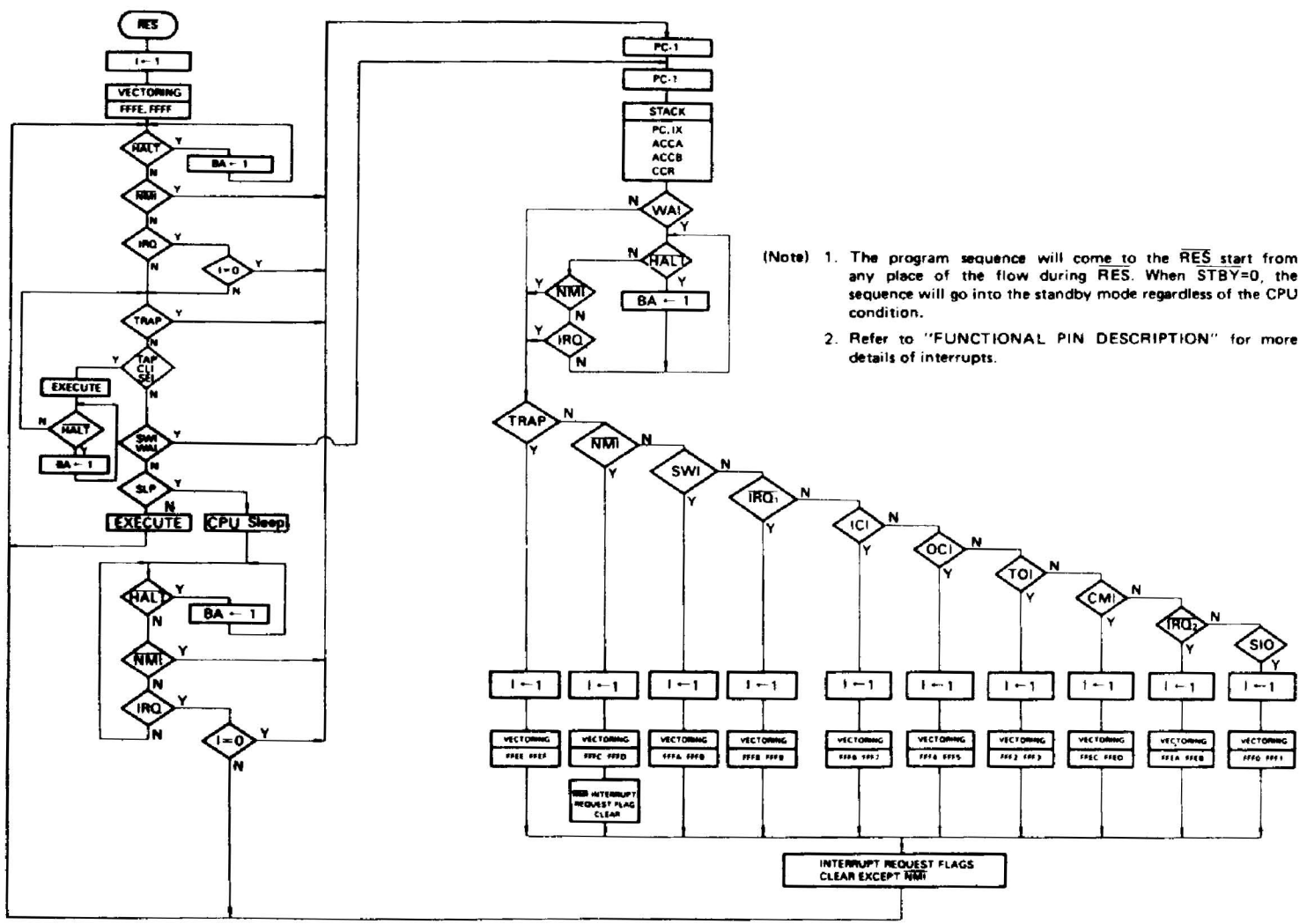
Figure 29 CPU Operation Mode Transition

Table 16 CPU Operation State and Port, Bus, Control Signal State

| Port                             | Reset | STBY <sup>*3</sup> | HALT | Sleep |
|----------------------------------|-------|--------------------|------|-------|
| A <sub>0</sub> ~ A <sub>7</sub>  | H     | T                  | T    | H     |
| Port 2                           | T     | T                  | Keep | Keep  |
| D <sub>0</sub> ~ D <sub>7</sub>  | T     | T                  | T    | T     |
| A <sub>8</sub> ~ A <sub>15</sub> | H     | T                  | T    | H     |
| Port 5                           | T     | T                  | Keep | Keep  |
| Port 6                           | T     | T                  | Keep | Keep  |
| Control Signal                   | *1    | T                  | *2   | *1    |

\*1  $\overline{RD}$ ,  $\overline{WR}$ ,  $\overline{R/W}$ ,  $\overline{LIR}$  = H, BA = L  
 \*2  $\overline{RD}$ ,  $\overline{WR}$ ,  $\overline{R/W}$  = T,  $\overline{LIR}$ , BA = H  
 \*3 E pin goes to high impedance state.





(Note) 1. The program sequence will come to the  $\overline{RES}$  start from any place of the flow during RES. When STBY=0, the sequence will go into the standby mode regardless of the CPU condition.  
 2. Refer to "FUNCTIONAL PIN DESCRIPTION" for more details of interrupts.

Figure 30 HD6303Y System Flow Chart



Table 17 Cycle-by-Cycle Operation

| Address Mode & Instructions |      | Cycles | Cycle # | Address Bus           | R/W | RD | WR | LIR | Data Bus                  |
|-----------------------------|------|--------|---------|-----------------------|-----|----|----|-----|---------------------------|
| <b>IMMEDIATE</b>            |      |        |         |                       |     |    |    |     |                           |
| ADC                         | ADD  | 2      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Operand Data              |
| AND                         | BIT  |        | 2       | Op Code Address+2     | 1   | 0  | 1  | 0   | Next Op Code              |
| CMP                         | EOR  |        |         |                       |     |    |    |     |                           |
| LDA                         | ORA  |        |         |                       |     |    |    |     |                           |
| SBC                         | SUB  |        |         |                       |     |    |    |     |                           |
| ADDD                        | CPX  | 3      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Operand Data (MSB)        |
| LDD                         | LDS  |        | 2       | Op Code Address+2     | 1   | 0  | 1  | 1   | Operand Data (LSB)        |
| LDX                         | SUBD |        | 3       | Op Code Address+3     | 1   | 0  | 1  | 0   | Next Op Code              |
| <b>DIRECT</b>               |      |        |         |                       |     |    |    |     |                           |
| ADC                         | ADD  | 3      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Address of Operand (LSB)  |
| AND                         | BIT  |        | 2       | Address of Operand    | 1   | 0  | 1  | 1   | Operand Data              |
| CMP                         | EOR  |        | 3       | Op Code Address+2     | 1   | 0  | 1  | 0   | Next Op Code              |
| LDA                         | ORA  |        |         |                       |     |    |    |     |                           |
| SBC                         | SUB  |        |         |                       |     |    |    |     |                           |
| STA                         |      | 3      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Destination Address       |
|                             |      |        | 2       | Destination Address   | 0   | 1  | 0  | 1   | Accumulator Data          |
|                             |      |        | 3       | Op Code Address+2     | 1   | 0  | 1  | 0   | Next Op Code              |
| ADDD                        | CPX  | 4      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Address of Operand (LSB)  |
| LDD                         | LDS  |        | 2       | Address of Operand    | 1   | 0  | 1  | 1   | Operand Data (MSB)        |
| LDX                         | SUBD |        | 3       | Address of Operand+1  | 1   | 0  | 1  | 1   | Operand Data (LSB)        |
|                             |      |        | 4       | Op Code Address+2     | 1   | 0  | 1  | 0   | Next Op Code              |
| STD                         | STS  | 4      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Destination Address (LSB) |
| STX                         |      |        | 2       | Destination Address   | 0   | 1  | 0  | 1   | Register Data (MSB)       |
|                             |      |        | 3       | Destination Address+1 | 0   | 1  | 0  | 1   | Register Data (LSB)       |
|                             |      |        | 4       | Op Code Address+2     | 1   | 0  | 1  | 0   | Next Op Code              |
| JSR                         |      | 5      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Jump Address (LSB)        |
|                             |      |        | 2       | FFFF                  | 1   | 1  | 1  | 1   | Restart Address (LSB)     |
|                             |      |        | 3       | Stack Pointer         | 0   | 1  | 0  | 1   | Return Address (LSB)      |
|                             |      |        | 4       | Stack Pointer-1       | 0   | 1  | 0  | 1   | Return Address (MSB)      |
|                             |      |        | 5       | Jump Address          | 1   | 0  | 1  | 0   | First Subroutine Op Code  |
| TIM                         |      | 4      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Immediate Data            |
|                             |      |        | 2       | Op Code Address+2     | 1   | 0  | 1  | 1   | Address of Operand (LSB)  |
|                             |      |        | 3       | Address of Operand    | 1   | 0  | 1  | 1   | Operand Data              |
|                             |      |        | 4       | Op Code Address+3     | 1   | 0  | 1  | 0   | Next Op Code              |
| AIM                         | EIM  | 6      | 1       | Op Code Address+1     | 1   | 0  | 1  | 1   | Immediate Data            |
| OIM                         |      |        | 2       | Op Code Address+2     | 1   | 0  | 1  | 1   | Address of Operand (LSB)  |
|                             |      |        | 3       | Address of Operand    | 1   | 0  | 1  | 1   | Operand Data              |
|                             |      |        | 4       | FFFF                  | 1   | 1  | 1  | 1   | Restart Address (LSB)     |
|                             |      |        | 5       | Address of Operand    | 0   | 1  | 0  | 1   | New Operand Data          |
|                             |      |        | 6       | Op Code Address+3     | 1   | 0  | 1  | 0   | Next Op Code              |

(Continued)



# HD6303Y, HD63A03Y, HD63B03Y, HD63C03Y

| Address Mode & Instructions | Cycles | Cycle #             | Address Bus         | R/W | RD | WR | LIR                   | Data Bus                      |
|-----------------------------|--------|---------------------|---------------------|-----|----|----|-----------------------|-------------------------------|
| <b>INDEXED</b>              |        |                     |                     |     |    |    |                       |                               |
| JMP                         | 3      | 1                   | Op Code Address + 1 | 1   | 0  | 1  | 1                     | Offset                        |
|                             |        | 2                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
|                             |        | 3                   | Jump Address        | 1   | 0  | 1  | 0                     | First Op Code of Jump Routine |
| ADC    ADD                  | 4      | 1                   | Op Code Address + 1 | 1   | 0  | 1  | 1                     | Offset                        |
| AND    BIT                  |        | 2                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
| CMP    EOR                  |        | 3                   | IX + Offset         | 1   | 0  | 1  | 1                     | Operand Data                  |
| LDA    ORA                  |        | 4                   | Op Code Address + 2 | 1   | 0  | 1  | 0                     | Next Op Code                  |
| SBC    SUB                  |        | 4                   | Op Code Address + 2 | 1   | 0  | 1  | 0                     | Next Op Code                  |
| TST                         | 4      | 1                   | Op Code Address + 1 | 1   | 0  | 1  | 1                     | Offset                        |
| STA                         |        | 2                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
| STX                         |        | 3                   | IX + Offset         | 0   | 1  | 0  | 1                     | Accumulator Data              |
|                             |        | 4                   | Op Code Address + 2 | 1   | 0  | 1  | 0                     | Next Op Code                  |
| ADD    LDD                  | 5      | 1                   | Op Code Address + 1 | 1   | 0  | 1  | 1                     | Offset                        |
| CPX    LDX                  |        | 2                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
| SUBD                        |        | 3                   | IX + Offset         | 1   | 0  | 1  | 1                     | Operand Data (MSB)            |
|                             |        | 4                   | IX + Offset + 1     | 1   | 0  | 1  | 1                     | Operand Data (LSB)            |
| STD    STS                  |        | 5                   | Op Code Address + 2 | 1   | 0  | 1  | 0                     | Next Op Code                  |
| STX                         | 5      | 1                   | Op Code Address + 1 | 1   | 0  | 1  | 1                     | Offset                        |
| JSR                         |        | 2                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
|                             |        | 3                   | Stack Pointer       | 0   | 1  | 0  | 1                     | Return Address (LSB)          |
| ASL    ASR                  |        | 4                   | Stack Pointer - 1   | 0   | 1  | 0  | 1                     | Return Address (MSB)          |
|                             |        | 5                   | IX + Offset         | 1   | 0  | 1  | 0                     | First Subroutine Op Code      |
| COM    DEC                  | 6      | 1                   | Op Code Address + 1 | 1   | 0  | 1  | 1                     | Offset                        |
| INC    LSR                  |        | 2                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
| NEG    ROL                  |        | 3                   | IX + Offset         | 1   | 0  | 1  | 1                     | Operand Data                  |
| ROR                         |        | 4                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
|                             |        | 5                   | IX + Offset         | 0   | 1  | 0  | 1                     | New Operand Data              |
| TIM                         |        | 6                   | Op Code Address + 2 | 1   | 0  | 1  | 0                     | Next Op Code                  |
| CLR                         | 1      | Op Code Address + 1 | 1                   | 0   | 1  | 1  | Offset                |                               |
|                             | 2      | FFFF                | 1                   | 1   | 1  | 1  | Restart Address (LSB) |                               |
|                             | 3      | IX + Offset         | 1                   | 0   | 1  | 1  | Operand Data          |                               |
|                             | 4      | IX + Offset         | 0                   | 1   | 0  | 1  | 00                    |                               |
|                             | 5      | Op Code Address + 2 | 1                   | 0   | 1  | 0  | Next Op Code          |                               |
| AIM    EIM                  | 7      | 1                   | Op Code Address + 1 | 1   | 0  | 1  | 1                     | Immediate Data                |
| OIM                         |        | 2                   | Op Code Address + 2 | 1   | 0  | 1  | 1                     | Offset                        |
|                             |        | 3                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
| OIM                         |        | 4                   | IX + Offset         | 1   | 0  | 1  | 1                     | Operand Data                  |
|                             |        | 5                   | FFFF                | 1   | 1  | 1  | 1                     | Restart Address (LSB)         |
| OIM                         |        | 6                   | IX + Offset         | 0   | 1  | 0  | 1                     | New Operand Data              |
|                             |        | 7                   | Op Code Address + 3 | 1   | 0  | 1  | 0                     | Next Op Code                  |

(Continued)

2





# HD6303Y, HD63A03Y, HD63B03Y, HD63C03Y

| Address Mode & Instructions                             | Cycles | Cycle # | Address Bus           | R/W | $\overline{RD}$ | $\overline{WR}$ | $\overline{LIR}$ | Data Bus                  |
|---|--------|---------|-----------------------|-----|-----------------|-----------------|------------------|---------------------------|
| <b>EXTEND</b>   |        |         |                       |     |                 |                 |                  |                           |
| JMP   | 3      | 1       | Op Code Address+1     | 1   | 0               | 1               | 1                | Jump Address (MSB)        |
|   |        | 2       | Op Code Address+2     | 1   | 0               | 1               | 1                | Jump Address (LSB)        |
|   |        | 3       | Jump Address          | 1   | 0               | 1               | 0                | Next Op Code              |
| ADC ADD TST<br>AND BIT<br>CMP EOR<br>LDA ORA<br>SBC SUB | 4      | 1       | Op Code Address+1     | 1   | 0               | 1               | 1                | Address of Operand (MSB)  |
|   |        | 2       | Op Code Address+2     | 1   | 0               | 1               | 1                | Address of Operand (LSB)  |
|   |        | 3       | Address of Operand    | 1   | 0               | 1               | 1                | Operand Data              |
|   |        | 4       | Op Code Address+3     | 1   | 0               | 1               | 0                | Next Op Code              |
| STA   | 4      | 1       | Op Code Address+1     | 1   | 0               | 1               | 1                | Destination Address (MSB) |
|   |        | 2       | Op Code Address+2     | 1   | 0               | 1               | 1                | Destination Address (LSB) |
|   |        | 3       | Destination Address   | 0   | 1               | 0               | 1                | Accumulator Data          |
|   |        | 4       | Op Code Address+3     | 1   | 0               | 1               | 0                | Next Op Code              |
| ADD<br>CPX LDD<br>LDS LDX<br>SUBD                       | 5      | 1       | Op Code Address+1     | 1   | 0               | 1               | 1                | Address of Operand (MSB)  |
|   |        | 2       | Op Code Address+2     | 1   | 0               | 1               | 1                | Address of Operand (LSB)  |
|   |        | 3       | Address of Operand    | 1   | 0               | 1               | 1                | Operand Data (MSB)        |
|   |        | 4       | Address of Operand+1  | 1   | 0               | 1               | 1                | Operand Data (LSB)        |
|   |        | 5       | Op Code Address+3     | 1   | 0               | 1               | 0                | Next Op Code              |
| STD STS<br>STX  | 5      | 1       | Op Code Address+1     | 1   | 0               | 1               | 1                | Destination Address (MSB) |
|   |        | 2       | Op Code Address+2     | 1   | 0               | 1               | 1                | Destination Address (LSB) |
|   |        | 3       | Destination Address   | 0   | 1               | 0               | 1                | Register Data (MSB)       |
|   |        | 4       | Destination Address+1 | 0   | 1               | 0               | 1                | Register Data (LSB)       |
|   |        | 5       | Op Code Address+3     | 1   | 0               | 1               | 0                | Next Op Code              |
| JSR   | 6      | 1       | Op Code Address+1     | 1   | 0               | 1               | 1                | Jump Address (MSB)        |
|   |        | 2       | Op Code Address+2     | 1   | 0               | 1               | 1                | Jump Address (LSB)        |
|   |        | 3       | FFFF                  | 1   | 1               | 1               | 1                | Restart Address (LSB)     |
|   |        | 4       | Stack Pointer         | 0   | 1               | 0               | 1                | Return Address (LSB)      |
|   |        | 5       | Stack Pointer-1       | 0   | 1               | 0               | 1                | Return Address (MSB)      |
|   |        | 6       | Jump Address          | 1   | 0               | 1               | 0                | First Subroutine Op Code  |
| ASL ASR<br>COM DEC<br>INC LSR<br>NEG ROL<br>ROR         | 6      | 1       | Op Code Address+1     | 1   | 0               | 1               | 1                | Address of Operand (MSB)  |
|   |        | 2       | Op Code Address+2     | 1   | 0               | 1               | 1                | Address of Operand (LSB)  |
|   |        | 3       | Address of Operand    | 1   | 0               | 1               | 1                | Operand Data              |
|   |        | 4       | FFFF                  | 1   | 1               | 1               | 1                | Restart Address (LSB)     |
|   |        | 5       | Address of Operand    | 0   | 1               | 0               | 1                | New Operand Data          |
|   |        | 6       | Op Code Address+3     | 1   | 0               | 1               | 0                | Next Op Code              |
| CLR   | 5      | 1       | Op Code Address+1     | 1   | 0               | 1               | 1                | Address of Operand (MSB)  |
|   |        | 2       | Op Code Address+2     | 1   | 0               | 1               | 1                | Address of Operand (LSB)  |
|   |        | 3       | Address of Operand    | 1   | 0               | 1               | 1                | Operand Data              |
|   |        | 4       | Address of Operand    | 0   | 1               | 0               | 1                | 00                        |
|   |        | 5       | Op Code Address+3     | 1   | 0               | 1               | 0                | Next Op Code              |

(Continued)



# HD6303Y, HD63A03Y, HD63B03Y, HD63C03Y

| Address Mode & Instructions |      | Cycles | Cycle #                         | Address Bus  | R/W                             | $\overline{RD}$                 | $\overline{WR}$                 | $\overline{LIR}$                | Data Bus   |
|-----------------------------|------|--------|---------------------------------|--|---------------------------------|---------------------------------|---------------------------------|---------------------------------|--|
| <b>IMPLIED</b>              |      |        |                                 |  |                                 |                                 |                                 |                                 |  |
| ABA                         | ABX  | 1      | 1                               | Op Code Address + 1  | 1                               | 0                               | 1                               | 0                               | Next Op Code   |
| ASL                         | ASLD |        |                                 |  |                                 |                                 |                                 |                                 |  |
| ASR                         | CBA  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| CLC                         | CLI  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| CLR                         | CLV  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| COM                         | DEC  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| DES                         | DEX  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| INC                         | INS  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| INX                         | LSR  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| LSRD                        | ROL  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| ROR                         | NOP  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| SBA                         | SEC  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| SEI                         | SEV  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| TAB                         | TAP  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| TBA                         | TPA  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| TST                         | TSX  |        |                                 |  |                                 |                                 |                                 |                                 |  |
| TXS                         |      |        |                                 |  |                                 |                                 |                                 |                                 |  |
| DAA                         | XGDX | 2      | 1<br>2                          | Op Code Address + 1<br>FFFF  | 1<br>1                          | 0<br>1                          | 1<br>1                          | 0<br>1                          | Next Op Code<br>Restart Address (LSB)  |
| PULA                        | PULB | 3      | 1<br>2<br>3                     | Op Code Address + 1<br>FFFF<br>Stack Pointer + 1   | 1<br>1<br>1                     | 0<br>1<br>0                     | 1<br>1<br>1                     | 0<br>1<br>1                     | Next Op Code<br>Restart Address (LSB)<br>Data from Stack   |
| PSHA                        | PSHB | 4      | 1<br>2<br>3<br>4                | Op Code Address + 1<br>FFFF<br>Stack Pointer<br>Op Code Address + 1                      | 1<br>1<br>0<br>1                | 0<br>1<br>1<br>0                | 1<br>1<br>0<br>1                | 1<br>1<br>1<br>0                | Next Op Code<br>Restart Address (LSB)<br>Accumulator Data<br>Next Op Code  |
| PULX                        |      | 4      | 1<br>2<br>3<br>4                | Op Code Address + 1<br>FFFF<br>Stack Pointer + 1<br>Stack Pointer + 2                    | 1<br>1<br>1<br>1                | 0<br>1<br>0<br>0                | 1<br>1<br>1<br>1                | 0<br>1<br>1<br>1                | Next Op Code<br>Restart Address (LSB)<br>Data from Stack (MSB)<br>Data from Stack (LSB)  |
| PSHX                        |      | 5      | 1<br>2<br>3<br>4<br>5           | Op Code Address + 1<br>FFFF<br>Stack Pointer<br>Stack Pointer - 1<br>Op Code Address + 1 | 1<br>1<br>0<br>0<br>1           | 0<br>1<br>1<br>1<br>0           | 1<br>1<br>0<br>0<br>1           | 1<br>1<br>1<br>1<br>0           | Next Op Code<br>Restart Address (LSB)<br>Index Register (LSB)<br>Index Register (MSB)<br>Next Op Code  |
| RTS                         |      | 5      | 1<br>2<br>3<br>4<br>5           | Op Code Address + 1<br>FFFF<br>Stack Pointer + 1<br>Stack Pointer + 2<br>Return Address  | 1<br>1<br>1<br>1<br>1           | 0<br>1<br>0<br>0<br>0           | 1<br>1<br>1<br>1<br>1           | 1<br>1<br>1<br>1<br>0           | Next Op Code<br>Restart Address (LSB)<br>Return Address (MSB)<br>Return Address (LSB)<br>First Op Code of Return Routine   |
| MUL                         |      | 7      | 1<br>2<br>3<br>4<br>5<br>6<br>7 | Op Code Address + 1<br>FFFF<br>FFFF<br>FFFF<br>FFFF<br>FFFF<br>FFFF                      | 1<br>1<br>1<br>1<br>1<br>1<br>1 | 0<br>1<br>1<br>1<br>1<br>1<br>1 | 1<br>1<br>1<br>1<br>1<br>1<br>1 | 0<br>1<br>1<br>1<br>1<br>1<br>1 | Next Op Code<br>Restart Address (LSB)<br>Restart Address (LSB)<br>Restart Address (LSB)<br>Restart Address (LSB)<br>Restart Address (LSB)<br>Restart Address (LSB) |

(Continued)

2



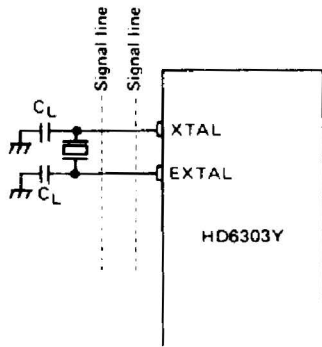
# HD6303Y, HD63A03Y, HD63B03Y, HD63C03Y

| Address Mode & Instructions |     | Cycles | Cycle #                        | Address Bus               | R/W | RD | WR | LIR | Data Bus                        |  |
|-----------------------------|-----|--------|--------------------------------|---------------------------|-----|----|----|-----|---------------------------------|--|
| <b>IMPLIED</b>              |     |        |                                |                           |     |    |    |     |                                 |  |
| WAI                         |     | 9      | 1                              | Op Code Address + 1       | 1   | 0  | 1  | 1   | Next Op Code                    |  |
|                             |     |        | 2                              | FFFF                      | 1   | 1  | 1  | 1   | Restart Address (LSB)           |  |
|                             |     |        | 3                              | Stack Pointer             | 0   | 1  | 0  | 1   | Return Address (LSB)            |  |
|                             |     |        | 4                              | Stack Pointer - 1         | 0   | 1  | 0  | 1   | Return Address (MSB)            |  |
|                             |     |        | 5                              | Stack Pointer - 2         | 0   | 1  | 0  | 1   | Index Register (LSB)            |  |
|                             |     |        | 6                              | Stack Pointer - 3         | 0   | 1  | 0  | 1   | Index Register (MSB)            |  |
|                             |     |        | 7                              | Stack Pointer - 4         | 0   | 1  | 0  | 1   | Accumulator A                   |  |
|                             |     |        | 8                              | Stack Pointer - 5         | 0   | 1  | 0  | 1   | Accumulator B                   |  |
|                             |     |        | 9                              | Stack Pointer - 6         | 0   | 1  | 0  | 1   | Conditional Code Register       |  |
| RTI                         |     | 10     | 1                              | Op Code Address + 1       | 1   | 0  | 1  | 1   | Next Op Code                    |  |
|                             |     |        | 2                              | FFFF                      | 1   | 1  | 1  | 1   | Restart Address (LSB)           |  |
|                             |     |        | 3                              | Stack Pointer + 1         | 1   | 0  | 1  | 1   | Conditional Code Register       |  |
|                             |     |        | 4                              | Stack Pointer + 2         | 1   | 0  | 1  | 1   | Accumulator B                   |  |
|                             |     |        | 5                              | Stack Pointer + 3         | 1   | 0  | 1  | 1   | Accumulator A                   |  |
|                             |     |        | 6                              | Stack Pointer + 4         | 1   | 0  | 1  | 1   | Index Register (MSB)            |  |
|                             |     |        | 7                              | Stack Pointer + 5         | 1   | 0  | 1  | 1   | Index Register (LSB)            |  |
|                             |     |        | 8                              | Stack Pointer + 6         | 1   | 0  | 1  | 1   | Return Address (MSB)            |  |
|                             |     |        | 9                              | Stack Pointer + 7         | 1   | 0  | 1  | 1   | Return Address (LSB)            |  |
|                             |     |        | 10                             | Return Address            | 1   | 0  | 1  | 0   | First Op Code of Return Routine |  |
| SWI                         |     | 12     | 1                              | Op Code Address + 1       | 1   | 0  | 1  | 1   | Next Op Code                    |  |
|                             |     |        | 2                              | FFFF                      | 1   | 1  | 1  | 1   | Restart Address (LSB)           |  |
|                             |     |        | 3                              | Stack Pointer             | 0   | 1  | 0  | 1   | Return Address (LSB)            |  |
|                             |     |        | 4                              | Stack Pointer - 1         | 0   | 1  | 0  | 1   | Return Address (MSB)            |  |
|                             |     |        | 5                              | Stack Pointer - 2         | 0   | 1  | 0  | 1   | Index Register (LSB)            |  |
|                             |     |        | 6                              | Stack Pointer - 3         | 0   | 1  | 0  | 1   | Index Register (MSB)            |  |
|                             |     |        | 7                              | Stack Pointer - 4         | 0   | 1  | 0  | 1   | Accumulator A                   |  |
|                             |     |        | 8                              | Stack Pointer - 5         | 0   | 1  | 0  | 1   | Accumulator B                   |  |
|                             |     |        | 9                              | Stack Pointer - 6         | 0   | 1  | 0  | 1   | Conditional Code Register       |  |
|                             |     |        | 10                             | Vector Address FFFA       | 1   | 0  | 1  | 1   | Address of SWI Routine (MSB)    |  |
|                             |     |        | 11                             | Vector Address FFFB       | 1   | 0  | 1  | 1   | Address of SWI Routine (LSB)    |  |
|                             |     |        | 12                             | Address of SWI Routine    | 1   | 0  | 1  | 0   | First Op Code of SWI Routine    |  |
| SLP                         |     | 4      | 1                              | Op Code Address + 1       | 1   | 0  | 1  | 1   | Next Op Code                    |  |
|                             |     |        | 2                              | FFFF                      | 1   | 1  | 1  | 1   | Restart Address (LSB)           |  |
|                             |     |        | 3                              | Sleep                     |     |    |    |     |                                 |  |
|                             |     |        | 4                              | FFFF                      | 1   | 1  | 1  | 1   | Restart Address (LSB)           |  |
|                             |     |        | 4                              | Op Code Address + 1       | 1   | 0  | 1  | 0   | Next Op Code                    |  |
| <b>RELATIVE</b>             |     |        |                                |                           |     |    |    |     |                                 |  |
| BCC                         | BCS | 3      | 1                              | Op Code Address + 1       | 1   | 0  | 1  | 1   | Branch Offset                   |  |
| BEQ                         | BGE |        | 2                              | FFFF                      | 1   | 1  | 1  | 1   | Restart Address (LSB)           |  |
| BGT                         | BHI |        | 3                              | Branch Address Test = "1" | 1   | 0  | 1  | 0   | First Op Code of Branch Routine |  |
| BLE                         | BLS |        | Op Code Address + 1 Test = "0" |                           |     |    |    |     | Next Op Code                    |  |
| BLT                         | BMT |        |                                |                           |     |    |    |     |                                 |  |
| BNE                         | BPL |        |                                |                           |     |    |    |     |                                 |  |
| BRA                         | BRN |        |                                |                           |     |    |    |     |                                 |  |
| BVC                         | BVS |        |                                |                           |     |    |    |     |                                 |  |
| BSR                         |     | 5      | 1                              | Op Code Address + 1       | 1   | 0  | 1  | 1   | Offset                          |  |
|                             |     |        | 2                              | FFFF                      | 1   | 1  | 1  | 1   | Restart Address (LSB)           |  |
|                             |     |        | 3                              | Stack Pointer             | 0   | 1  | 0  | 1   | Return Address (LSB)            |  |
|                             |     |        | 4                              | Stack Pointer - 1         | 0   | 1  | 0  | 1   | Return Address (MSB)            |  |
|                             |     |        | 5                              | Branch Address            | 1   | 0  | 1  | 0   | First Op Code of Subroutine     |  |



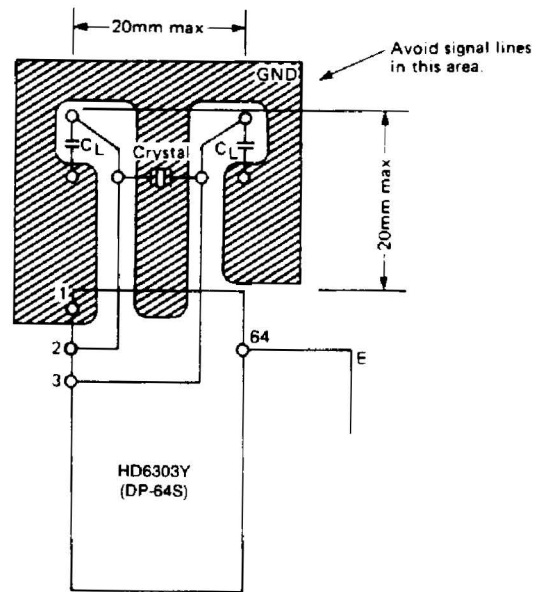
**WARNING CONCERNING THE BOARD DESIGN OF OSCILLATION CIRCUIT**

When designing a board, note that crosstalk may disturb the normal oscillation if signal lines are placed near the oscillation circuit as shown in Figure 31. Place the crystal and  $C_L$  as close to the HD6303Y as possible.



Do not use this kind of printed-circuit board design.

Figure 31 Warning concerning board design of oscillation circuit



(Top View)

Figure 32 Example of Oscillation Circuits in Board Design

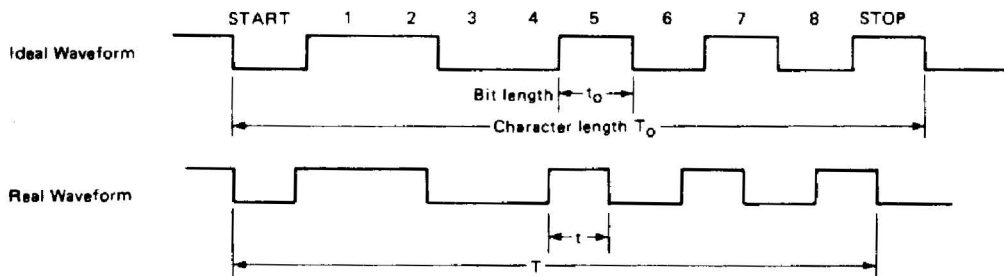
**RECEIVE MARGIN OF THE SCI**

Receive margin of the SCI contained in the HD6303Y is shown in Table 18.

Note: SCI = Serial Communication Interface

Table 18

|         | Bit distortion tolerance<br>( $t-t_0$ ) / $t_0$ | Character distortion tolerance<br>( $T-T_0$ ) / $T_0$ |
|---------|---|---|
| HD6303Y | ±43.7%  | ±4.37%  |



■ **WARNING CONCERNING WAI INSTRUCTION**

If the HALT signal is accepted by the MCU while the WAI instruction is executing, the CPU will not operate correctly after HALT mode is canceled.

WAI is a instruction which waits for an interrupt. The corresponding interrupt routine is executed after an interrupt occurs.

However, during the execution of the WAI instruction, HALT input makes the CPU malfunction and fetch an abnormal interrupt vectoring address.

In HALT mode, the CPU operates correctly without the WAI instruction, and WAI is executed correctly without HALT input. Therefore, if HALT input is necessary, make interrupts wait during the loop routine, as shown in Figure 33.

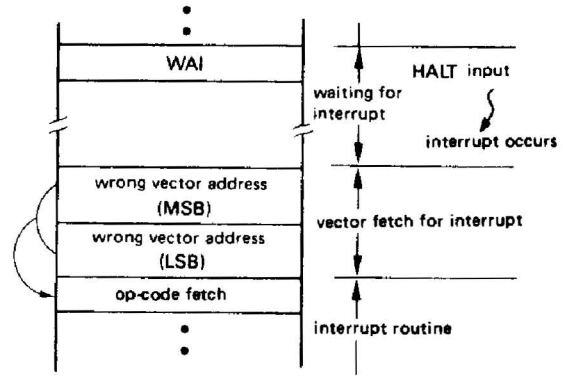


Figure 33 MAC function during WAI

■ **WRITE-ONLY REGISTER**

When the CPU reads a write-only register, the read data is always \$FF, regardless of the value in the write-only register. Therefore, be careful of the results of instructions which read write-only register and perform an arithmetic or logical operation on its contents, such as AIM, ADD, or ROL. is executed, because the arithmetic or logical operation is always done with the data \$FF. In particulars, don't use the AIM, OIM or EIM instruction to manipulate the DDR bit of PORT.

■ **WARNING CONCERNING POWER START-UP**

RES must be held low for at least 20 ms when the power starts up. In this case, the internal reset function is not effective until the oscillation begins at power-on. The RES signal is input to the LSI in synchronism with the internal clock  $\phi$  (shown in Figure 35.)

Therefore, after power starts up, the LSI conditions such as its I/O ports and operating mode, are unstable. Fix the level of I/O ports by means of an external circuit to determine the level for system operation during the oscillator stabilization time.

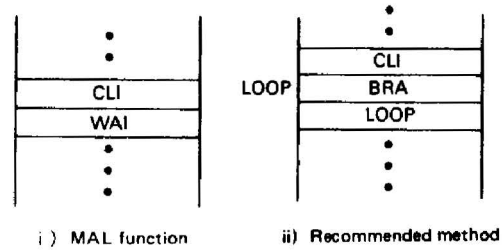


Figure 34 Program to wait for interrupt

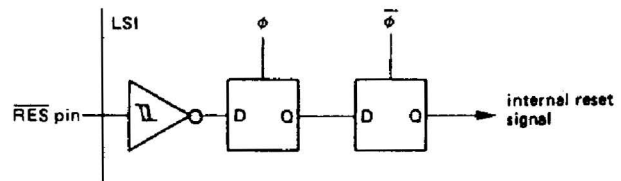


Figure 35 RES circuit

