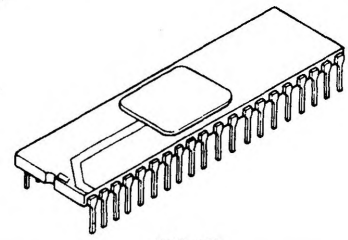
HD6846
COMBO (Combination ROM I/O Timer)

The HD6846 combination chip provides the means, in conjunction with the HD6802, to develop a basic 2-chip microcomputer system. The HD6846 consists of 2048 bytes of mask-programmable ROM, an 8-bit bidirectional data port with control lines, and a 16-bit programmable timer-counter.

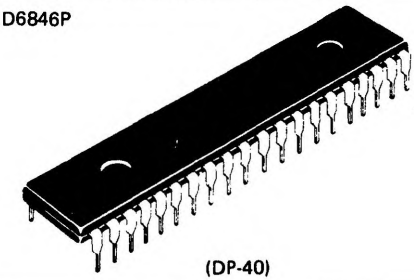
This device is capable of interfacing with the HD6802 (basic HD6800, clock and 128 bytes of RAM) as well as the HD6800 if desired. No external logic is required to interface with most peripheral devices.

HD6846



(DC-40)

HD6846P

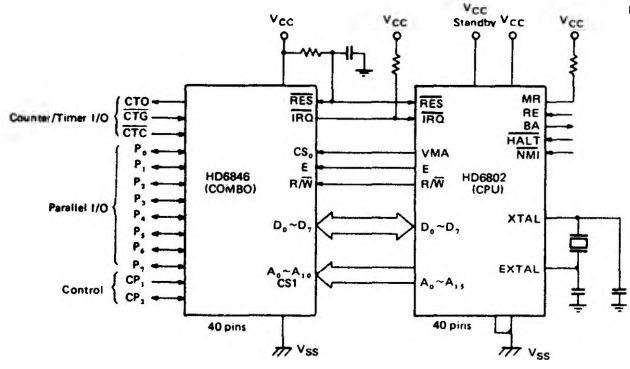


(DP-40)

■ FEATURES

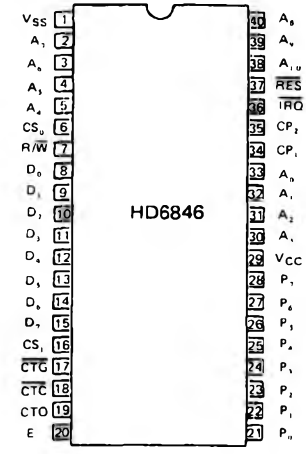
- 2048 8-Bit Bytes of Mask-Programmable ROM
- 8-Bit Bidirectional Data Port for Parallel Interface plus Two Control Lines
- Programmable Interval Timer-Counter Functions
- Programmable I/O Peripheral Data, Control and Direction Registers
- Compatible with the Complete HMCS6800 Microcomputer Product Family
- TTL-Compatible Data and Peripheral Lines
- Single 5-Volt Power Supply
- Compatible with MC6846

■ TYPICAL MICROCOMPUTER



This is a block diagram of a typical cost effective microcomputer. The MPU is the center of the microcomputer system and is shown in a minimum system interfacing with a ROM combination chip. It is not intended that this system be limited to this function but that it be expandable with other parts in the HMCS6800 Microcomputer family.

■ PIN ARRANGEMENT



(Top View)

■ ABSOLUTE MAXIMUM RATINGS

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

* With respect to V_{SS} (SYSTEM GND)

(NOTE) Permanent LSI damage may occur if maximum ratings are exceeded. Normal operation should be under recommended operating conditions. If these conditions are exceeded, it could affect reliability of LSI.

■ RECOMMENDED OPERATING CONDITIONS

Item	Symbol	min	typ	max	Unit
Supply Voltage	V_{CC}*	4.75	5.0	5.25	V
Input Voltage	V_{IL}*	-0.3	–	0.8	V
	V_{IH}*	2.0	–	V_{CC}	V
Operating Temperature	T_{opr}	-20	25	75	°C

* With respect to V_{SS} (SYSTEM GND)

■ ELECTRICAL CHARACTERISTICS

● DC CHARACTERISTICS (V_{CC}=5.0V±5%, V_{SS}=0V, Ta=-20~+75°C, unless otherwise noted.)

Item		Symbol	Test Condition	min.	typ.	max.	Unit
Input "High" Voltage	All Inputs	V_{IH}		2.0	–	V_{CC}	V
Input "Low" Voltage	All Inputs	V_{IL}		-0.3	–	0.8	V
Clock Overshoot/Undershoot	Input "High" Level	V_{OS}		V_{CC} −0.5	–	V_{CC} +0.5	V
	Input "Low" Level			V_{SS} −0.5	–	V_{SS} +0.5	
Input Leakage Current	R/$\overline{W}$, $\overline{RES}$, $\overline{CS_0}$, CS_1, CP_1, $\overline{CTG}$, $\overline{CTC}$, E, A_0~A_{10}	I_{in}	V_{in} = 0 ~ 5.25V	-2.5	–	2.5	μA
Three-State (Off State) Input Current	D_0~D_7, P_0~P_7, CP_2	I_{TSI}	V_{in} = 0.4 ~ 2.4V	-10	–	10	μA
Output "High" Voltage	D_0~D_7	V_{OH}	I_{OH} = -205μA	2.4	–	–	V
	CP_2, P_0~P_7		I_{OH} = -200μA	2.4	–	–	V
	CTO		I_{OH} = -200μA	2.4	–	–	V
Output "Low" Voltage	D_0~D_7	V_{OL}	I_{OL} = 1.6mA	–	–	0.4	V
	Other Outputs		I_{OL} = 3.2mA	–	–	0.4	V
Output "High" Current (Sourcing)	D_0~D_7	I_{OH}	V_{OH} = 2.4V	-205	–	–	μA
	CTO, CP_2, P_0~P_7		V_{OH} = 2.4V	-200	–	–	μA
Output "High" Current (Sourcing) (the current for driving other than TTL, e.g., Darlington Base)	CP_2, P_0~P_7	I_{OH}	V_{OH} = 1.5V	-1.0	–	−10	mA
Output "Low" Current (Sinking)	D_0~D_7	I_{OL}	V_{OL} = 0.4V	1.6	–	–	mA
	Other Outputs			3.2	–	–	
Output Leakage Current (Off State)	$\overline{IRQ}$	I_{LOH}	V_{OH} = 2.4V	–	–	10	μA
Power Dissipation		P_D		–	–	800	mW
Capacitance	E	C_{in}	V_{CC} = 0V V_{in} = 0V T_a = 25°C f = 1MHz	–	–	20	pF
	D_0~D_7	C_{in}		–	–	12.5	pF
	P_0~P_7, CP_2, CTO	C_{out}		–	–	10	pF
	A_0~A_{10}, R/$\overline{W}$	C_{in}		–	–	7.5	pF
	$\overline{RES}$, $\overline{CS_0}$, CS_1, CP_1, $\overline{CTG}$	C_{in}		–	–	10	pF
	$\overline{IRQ}$	C_{out}		–	–	7.5	pF

- **AC CHARACTERISTICS** (V_{CC}=5.0V±5%, V_{SS}=0V, Ta=-20~+75°C, unless otherwise noted.)

1. BUS TIMING

Item	Symbol	Test Condition	min	typ	max	Unit
Enable Cycle Time	t_{cycE}	Fig. 1	1.0	–	10	μs
Enable Pulse Width, "Low"	PW_{EL}		430	–	4500	ns
Enable Pulse Width, "High"	PW_{EH}		430	–	4500	ns
Address Set Up Time	t_{AS}		140	–	–	ns
Data Delay Time	t_{DDR}		–	–	320	ns
Data Hold Time	t_H		10	–	–	ns
Address Hold Time	t_{AH}		10	–	–	ns
Enable Rise and Fall Time	t_{Ef},t_{Er}		–	–	25	ns
Data Set Up Time	t_{DSW}		195	–	–	ns
Reset "Low" Time	t_{RL}		2	–	–	μs
Interrupt Release Time	t_{IR}	Fig. 2	–	–	1.6	μs

2. PALLAREL PERIPHERAL I/O LINE TIMING

Item	Symbol	Test Condition	min	typ	max	Unit
Peripheral Data Setup Time	t_{PDSU}	Fig. 3	200	–	–	ns
Rise and Fall Times CP_1, CP_2	t_{Pr}, t_{Pf}	Fig. 5	–	–	1.0	μs
Delay Time E to CP_2 Fall	t_{CP2}	Fig. 4	–	–	1.0	μs
Delay Time I/O Data CP_2 Fall	t_{DC}		20	–	–	ns
Delay Time E to CP_2 Rise	t_{RS1}		–	–	1.0	μs
Delay Time CP_1 to CP_2 Rise	t_{RS2}	Fig. 5	–	–	2.0	μs
Peripheral Data Delay	t_{PDW}	Fig. 4	–	–	1.0	μs
Peripheral Data Setup Time for Latch	t_{PSU}	Fig. 9	100	–	–	ns
Peripheral Data Hold Time for Latch	t_{PDH}		15	–	–	ns

3. TIMER/COUNTER LINE TIMING

Item	Symbol	Test Condition	min	typ	max	Unit
$\overline{CTC}$, $\overline{CTG}$ Rise and Fall Time	t_{Cr}, t_{Cf}	Fig. 6	–	–	100	ns
$\overline{CTC}$, $\overline{CTG}$ Pulse Width, "High" (Asynchronous Mode)	t_{PWH}		t_{cycE} +250	–	–	ns
$\overline{CTC}$, $\overline{CTG}$ Pulse Width, "Low" (Asychronous Mode)	t_{PWL}		t_{cycE} +250	–	–	ns
$\overline{CTC}$, $\overline{CTG}$ Setup Time (Synchronous Mode)	t_{su}	Fig. 7	200	–	–	ns
$\overline{CTC}$, $\overline{CTG}$ Hold Time (Synchronous Mode)	t_{hd}		50	–	–	ns
CTO Delay Time	t_{CTO}	Fig. 8	–	–	1.0	μs

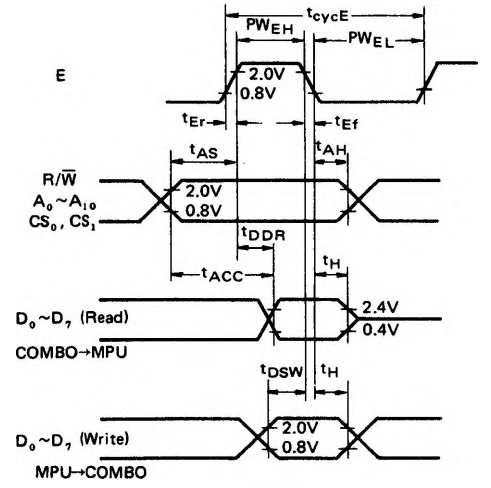


Figure 1 Bus Read/Write Timing

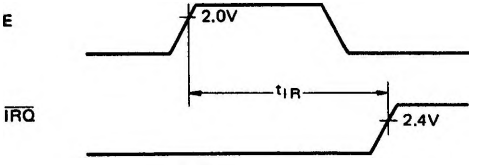


Figure 2 $\overline{IRQ}$ Release Time

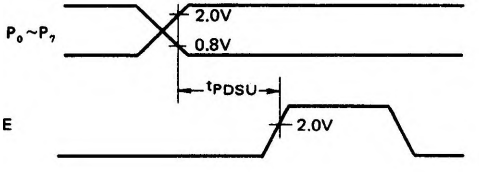


Figure 3 Peripheral Data Set Up Time

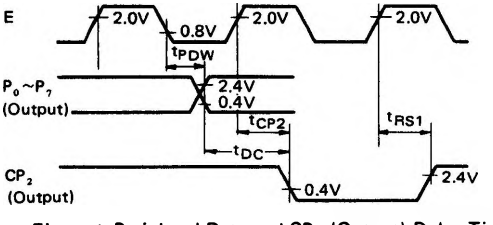


Figure 4 Peripheral Data and CP_2 (Output) Delay Time

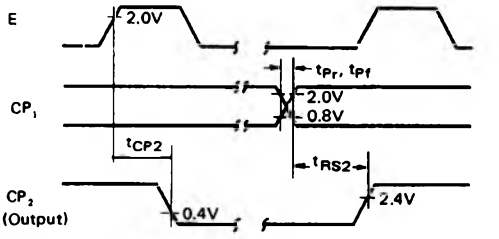


Figure 5 CP_2 (Output) Delay Time

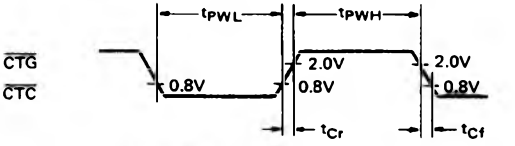


Figure 6 $\overline{CTG}$, $\overline{CTC}$ Pulse Width

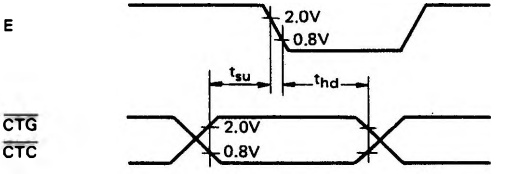


Figure 7 $\overline{CTG}$, $\overline{CTC}$ Setup Time and Hold Time

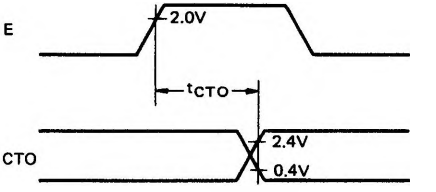


Figure 8 CTO Delay Time

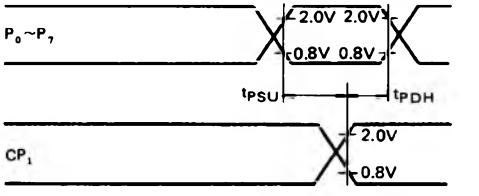


Figure 9 Peripheral Port Latch Setup and Hold Time

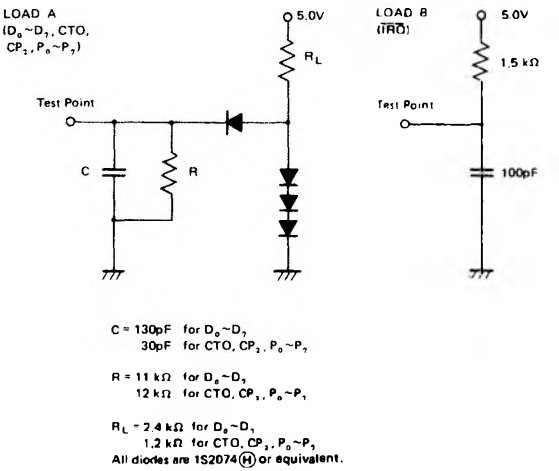


Figure 10 Bus Timing Test Loads

■ GENERAL DESCRIPTION

The HD6846 combination chip may be partitioned into three functional operating sections: programmed storage, timer-counter functions, and a parallel I/O port.

• Programmed Storage

The mask-programmable ROM section is similar to other ROM products of the HMCS6800 family. The ROM is organized in a 2048 by 8-bit array to provide read only storage for a minimum microcomputer system. Two mask-programmable chip selects are available for user definition.

Address inputs $A_0 \sim A_{10}$ allow any of the 2048 bytes of ROM to be uniquely addressed. Bidirectional data lines ($D_0 \sim D_7$) allow the transfer of data between the MPU and the HD 6846.

• Timer-Counter Functions

Under software control this 16-bit binary counter may be programmed to count events, measure frequencies, time intervals, or similar tasks. Internal registers associated with the I/O functions may be selected with A_0, A_1 and A_2. It may also be used for square wave generation, single pulses of controlled duration, and gated signals. Interrupts may be generated from a number of conditions selectable by software programming.

The timer/counter control register allows control of the interrupt enable, output enable, selection of an internal or external clock source, a ÷ 8 prescaler, and operating mode. Input pin $\overline{CTC}$ (counter-timer clock) will accept an asynchronous clock pulse to decrement the internal register for the counter-timer. If the divide-by-8 prescaler is used, the maximum clock rate can be four times the master clock frequency with an absolute maximum of 4 MHz. Gate input ($\overline{CTG}$) accepts an asynchronous TTL-compatible signal which may be used as a trigger or gating function to the counter-timer. A counter-timer output (CTO) is also available and is under software control being dependent on the timer control register, the gate input, and the clock source.

• Parallel I/O Port

The parallel bidirectional I/O port has functional operational characteristics similar to the B port on the HD6821 PIA. This includes 8 bidirectional data lines and two handshake control signals. The control and operation of these lines are completely software programmable.

The interrupt input (CP_1) will set the interrupt flag CSR1 of the composite status register. The peripheral control (CP_2) may be programmed to act as an interrupt input (set CSR2) or as a peripheral control output.

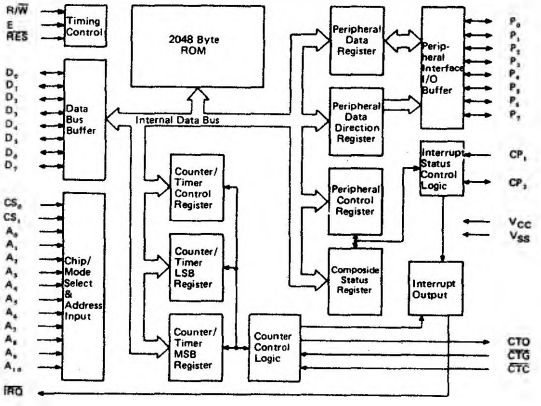


Figure 11 Combination ROM I/O Timer (COMBO)

Basic Block Diagram

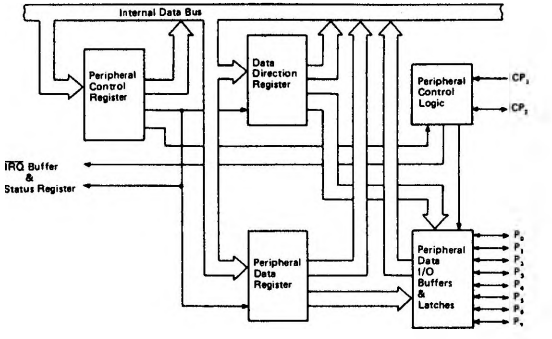


Figure 12 Parallel I/O Port Block Diagram

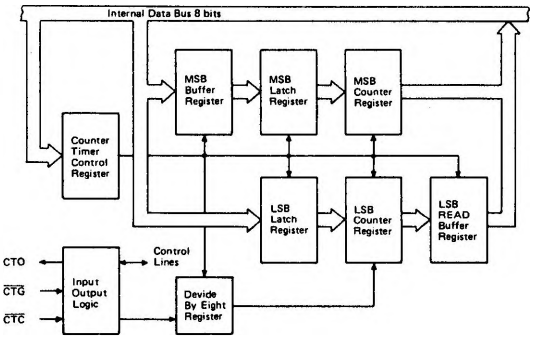


Figure 13 Timer/Counter Block Diagram

■ SIGNAL DESCRIPTION

• Bus Interface

The HD6846 interfaces to the HMCS6800 Bus via an eight bit bidirectional data bus, two Chip Select lines, a Read/Write line, and eleven address lines. These signals, in conjunction with the HMCS6800 VMA output, permit the MPU to control the HD6846.

• Bidirectional Data Bus ($D_0 \sim D_7$)

The bidirectional data lines ($D_0 \sim D_7$) allow the transfer of data between the MPU and the HD6846. The data bus output drivers are three-state devices which remain in the high-impedance (Off) state except when the MPU performs an HD6846 register or ROM read ($R/\overline{W}$ = 1 and I/O Registers or ROM selected).

• Chip Select (CS_0, CS_1)

The CS_0 and CS_1 inputs are used to select the ROM or I/O timer of the HD6846. They are mask programmed to be active "High" or active "Low" as chosen by the user.

• Address Inputs ($A_0 \sim A_{10}$)

The Address Inputs allow any of the 2048 bytes of ROM to be uniquely selected when the circuit is operating in the ROM mode. In the I/O-Timer mode, address inputs A_0, A_1, and A_2 select the proper I/O Register, while A_3 through A_{10} (together with CS_0 and CS_1) can be used as additional qualifiers in the I/O Select circuitry. (See the section on I/O-Timer Select for additional details.)

• Reset ($\overline{RES}$)

The active "Low" state of the $\overline{RES}$ input is used to initialize all register bits in the I/O section of the device to their proper values. (See the section on Initialization for Reset conditions for timer and peripheral registers.)

• Enable (E)

This signal synchronizes data transfer between the MPU and the HD6846. It also performs an equivalent synchronization function on the external clock, reset, and gate inputs of the HD6846 Timer section.

• Read/Write ($R/\overline{W}$)

This signal is generated by the MPU and is used to control the direction of data transfer on the bidirectional data pins. A "Low" level on the $R/\overline{W}$ input enables the HD6846 input buffers and data is transferred to the circuit during the E pulse when the part has been selected. A "High" level on the $R/\overline{W}$ input enables the output buffers and data is transferred to the MPU during E when the part is selected.

• Interrupt Request ($\overline{IRQ}$)

The active "Low" $\overline{IRQ}$ output acts to interrupt the MPU through logic included on the HD6846. This output utilizes an open drain configuration and permits other interrupt request outputs from other circuits to be connected in a wire-OR configuration.

• Peripheral Data ($P_0 \sim P_7$)

The peripheral data lines can be individually programmed as either inputs or outputs via the Data Direction Register. When programmed as outputs, these lines will drive two standard TTL

loads (3.2 mA). They are also capable of sourcing up to 1.0 mA at 1.5 Volts (Logic "1" output.)

When programmed as inputs, the output drivers associated with these lines enter a three-state (high impedance) mode. Since there is no internal pull-up for these lines, they represent a maximum 10μA load to the circuitry driving them -- regardless of logic state.

A logic zero at the $\overline{RES}$ input forces the peripheral data lines to the input configuration by clearing the Data Direction Register. This allows the system designer to preclude the possibility of having a peripheral data output connected to an external driver output during power-up sequence.

- **Interrupt Input (CP_1)**

Peripheral input line CP_1 is an input-only that sets the Interrupt Flags of the Composite Status register. The active transition for this signal is programmed by the peripheral control register for the parallel port. CP_1 may also act as a strobe for the peripheral data register when it is used as an input latch. Details for programming CP_1 are in the section on the parallel peripheral port.

(Note)

Unexpected noise may occur on the peripheral data line when the peripheral data register is loaded with "1". This erroneous noise may occur only when peripheral data line is specified as output and the peripheral data register has already been loaded with "1". Note that peripheral data line doesn't keep "High" level continuously in the case write peripheral data register operation is executed.

- **Peripheral Control (CP_2)**

Peripheral Control line CP_2 may be programmed to act as an Interrupt input or Peripheral Control output. As an input, this line has high impedance and is compatible with standard TTL voltage levels. As an output, it is also TTL compatible and may be used as a source of 1 mA at 1.5 V to directly drive the base of a Darlington transistor switch. This line is programmed by the Peripheral Control Register.

- **Counter Timer Output (CTO)**

The Counter Timer Output is software programmable by selected bits in the timer/counter control register. The mode of operation is dependent on the Timer control register, the gate input, and the clock source. The output is TTL compatible.

- **External Clock Input ($\overline{CTC}$)**

Input pin $\overline{CTC}$ will accept asynchronous TTL voltage level signals to be used as a clock to decrement the Timer. The "High" and "Low" levels of the external clock must be stable for at least one system clock period plus the sum of the setup and hold times for the inputs. The asynchronous clock rate can vary from dc to the limit imposed by System E, setup, and hold times.

The external clock input is clocked in by Enable pulses. Three Enable periods are used to synchronize and process the external clock. The fourth Enable pulse decrements the internal counter. This does not affect the input frequency; it merely creates a delay between a clock input transition and internal recognition of that transition by the HD6846. All references to $\overline{CTC}$ inputs in this document relate to internal recognition of the input transition. Note that a clock transition which does not meet setup and hold time specifications may require an additional Enable pulse for recognition.

When observing recurring events, a lack of synchronization will result in either "System jitter" or "Input jitter" being observed on the output of the HD6846 when using an asynchronous clock and gate input signal. "System jitter" is the result of the input signals being out of synchronization with Enable, permitting signals with marginal set-up and hold time to be recognized by either the bit time nearest the input transition or subsequent bit time. "Input jitter" can be as great at the time between the negative going transitions of the input signal plus the system jitter if the first transition is recognized during one system cycle, and not recognized the next cycle or vice-versa.

- **Gate Inputs ($\overline{CTG}$)**

The input pin $\overline{CTG}$ accepts an asynchronous TTL-compatible signal which is used as a trigger or a clock gating function to the Timer. The gating input is clocked into the HD6846 by the Enable signal in the same manner as the previously discussed clock inputs. That is, a $\overline{CTG}$ transition is recognized on the fourth Enable pulse (provided setup and hold time requirements are met), and the "High" or "Low" levels of the $\overline{CTG}$ input must be stable for at least one system clock period plus the sum of setup and hold times. All references to $\overline{CTG}$ transition in this document relate to internal recognition of the input transition.

The $\overline{CTG}$ input of the timer directly affects the internal 16-bit counter. The operation of $\overline{CTG}$ is therefore independent of the ÷ 8 prescaler selection.

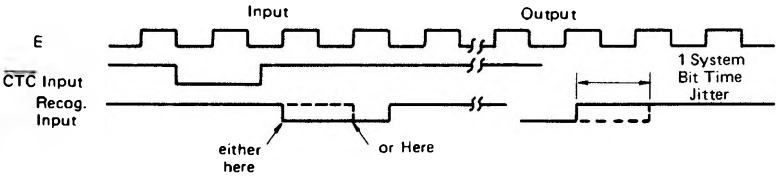
■ FUNCTIONAL SELECT CIRCUITRY

- **I/O-Timer Select Circuitry**

CS_0 and CS_1 are user programmable. Any of the four binary combinations of CS_0 and CS_1 can be used to select the ROM. Likewise, any other combination can be used to select the I/O-Timer. In addition, several address lines are used as qualifiers for the I/O-Timer. Specifically, $A_3 = A_4 = A_5$ = logical "0". A_6 can be programmed to a "1", "0", or don't care. $A_7 = A_8 = A_9 = A_{10}$ = don't care or one line only may be programmed to a logical "1". Figure 14 outlines in diagrammatic form the available chip select options.

- **Internal Addressing**

Seven I/O Register locations within the HD6846 are accessible to the MPU data bus. Selection of these registers is controlled by A_0, A_1, and A_2 (as shown in Table 1) provided the I/O timer is selected. The combination status register is Read-only; all other Registers are Read and Write.



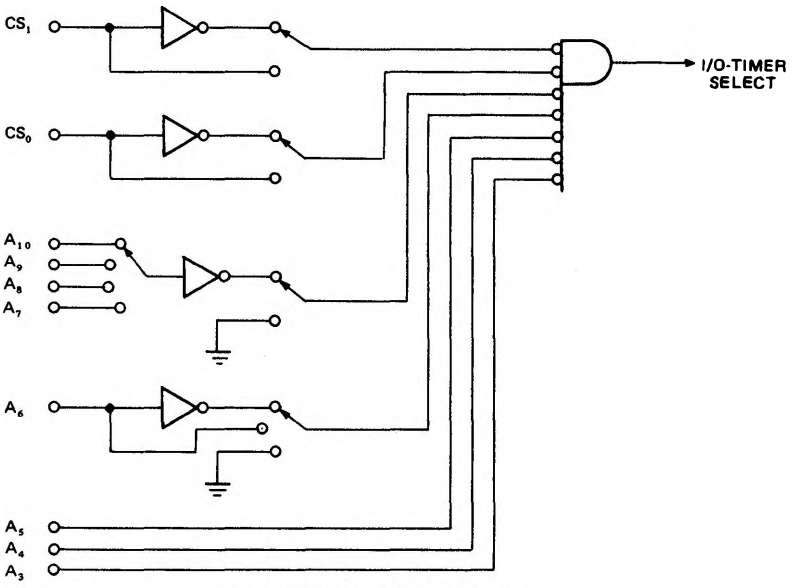


Figure 14 I/O-Timer Select Circuitry

Table 1 Internal Register Addresses

REGISTER SELECTED	A_2	A_1	A_0
Combination Status Register	0	0	0
Peripheral Control Register	0	0	1
Data Direction Register	0	1	0
Peripheral Data Register	0	1	1
Combination Status Register	1	0	0
Timer Control Register	1	0	1
Timer MSB Register	1	1	0
Timer LSB Register	1	1	1
ROM Address	×	×	×

Initialization

When the $\overline{RES}$ input has accepted a "Low" signal, all registers are initialized to the reset state. The data direction and peripheral data registers are cleared. The Peripheral Control Register is cleared except for bit 7 (the $\overline{RES}$ bit). This forces the parallel port to the input mode with Interrupts disabled. To remove the $\overline{RES}$ condition from the parallel port, a "0" must be written into the Peripheral Control Register bit 7 (PCR7).

The counter latches are preset to their maximal count, the Timer control register bits are reset to zero except for Bit 0 (TCR0 is set), the counter output is cleared, and the counter clock disabled. This state forces the timer counter to remain in an inactive state. The combination status register is cleared of all interrupt flags. During timer initialization, the reset bit (CCR0) must be cleared.

ROM

The Mask Programmable ROM section is similar in operation to other ROM products of the HMCS6800 Microprocessor family. The ROM is organized as 2048 words of 8-bits to provide read-only storage for a minimum microcomputer system. The ROM is active when selected by the unique combination of the chip select inputs.

ROM Select

The active levels of CS_0 and CS_1 for ROM and I/O select are a user programmable option. Either CS_0 or CS_1 may be programmed active "High" or active "Low", but different codes must be used for ROM or I/O select. CS_0 and CS_1 are mask programmed simultaneously with the ROM pattern. The ROM Select Circuitry is shown in Figure 15.

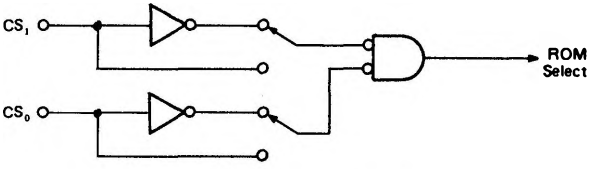


Figure 15 ROM Select Circuitry

■ TIMER OPERATION

The Timer may be programmed to operate in modes which fit a wide variety of applications. The device is fully bus compatible with the HMCS6800 system, and is accessed by Load and Store operations from the MPU.

In a typical application, the timer will be loaded by storing two bytes of data into the counter latch. This data is then transferred into the counter during a Counter Initialization cycle. The counter decrements on each subsequent clock cycle (which may be Enable or an external clock) until one of several predetermined conditions causes it to halt or recycle. Thus the timer is programmable, cyclic in nature, controllable by external inputs or MPU program, and accessible to the MPU at any time.

- **Counter Latch Initialization**

The Timer consists of a 16-bit addressable counter and two 8-bit addressable latches. The function of the latches is to store a binary equivalent of the desired count value minus one. Counter initialization results in the transfer of the latch contents of the counter. It should be noted that data transfer to the counters is always accomplished via the latches. Thus, the counter latches may be accurately described as a 16-bit "counter initialization data" storage register.

In some modes of operation, the initialization of the latches will cause simultaneous counter initialization (i.e. immediate transfer of the new latch data into the counters). It is, therefore, necessary to insure that all-16-bit of the latches are updated simultaneously. Since the HD6846 data bus is 8-bit wide, a temporary register (MSB Buffer Register) is provided for in the Most Significant Byte of the desired latch data. This is a "write-only" register selected via address lines A_0, A_1, and A_2. Data is transferred directly from the data bus to the MSB Buffer when the chip is selected, R/$\overline{W}$ is "Low", and the timer MSB register is selected (A_0 = "0", A_1 = A_2 = "1").

The lower 8-bit of the counter latch can also be referred to as a "write-only" register. Data Bus information will be transferred directly to the LSB of a counter latch when the chip is selected, R/$\overline{W}$ is "Low" and the Timer LSB Register is selected (A_0 = A_1 = A_2 = "1"). Data from the MSB Buffer will automatically be transferred into the Most Significant Byte of the counter latches simultaneously with the transfer of the Data Bus information to the Least Significant Byte of the Counter Latch. For brevity, the conditions for this operation will be referred to henceforth as a "Write Timer Latches Command."

The HD6846 has been designed to allow transfer of two bytes of data into the counter latches from any source, provided the MSB is transferred first. In many applications, the source of data will be an HMCS6800 MPU. It should therefore be noted that the 16-bit store operations of the HMCS6800 family microprocessors (STS and STX) transfer data in the order required by the HD6846. A Store Index Register instruction, for example, results in the MSB of the index register being transferred to the selected address, then the LSB of the index register being written into the next higher location. Thus, either the index register or stack pointer may be transferred directly into a selected counter latch with a single instruction.

A logic zero at the $\overline{RES}$ input also initializes the counter latches. All latches will assume maximum count (65,535) values. It is important to note that an internal reset (Bit zero of the Timer/Control Register Set) has no effect on the counter latches.

- **Counter Initialization**

Counter Initialization is defined as the transfer of data from the latches to the counter with attendant clearing of the Individual Interrupt Flag associated with the counter. Counter Initialization always occurs when a reset condition (external $\overline{RES}$ = "0" or TCR0 = "1") is recognized. It can also occur (dependent on The Timer Mode) with a Write Timer Latches command or recognition of a negative transition of the $\overline{CTG}$ input.

Counter recycling or reinitialization occurs when a clock input is recognized after the counter has reached an all-zero state. In this case, data is transferred from the Latches to the Counter, but the Interrupt Flag is unaffected.

- **Timer Control Register**

The Timer Control register (see Table 2) in the HD6846 is used to modify timer operation to suit a variety of applications. The Timer Control Register has a unique address space (A_0 = "1, A_1 = "0", A_2 = "1") and therefore may be written into at any time. The least significant bit of the Control Register is used as an Internal Reset bit. When this bit is a logic zero, all timers are allowed to operate in the modes prescribed by the remaining bits of the timer control register.

Writing "1" into Timer Control Register Bit 0 (TCR0) causes the counter to be preset with the conents of the counter latches, all counter clocks are disabled, and the timer output and interrupt flag (Status Register) are reset. The Counter Latch and Timer/Control Register are undisturbed by an Internal Reset and may be written into regardless of the state of TCR0.

Timer Control Register Bit 1 (TCR1) is used to select the clock source. When TCR1 = "0", the external clock input $\overline{CTC}$ is selected, and when TCR1 = "1", the timer uses Enable.

Table 2 Format for Timer/Counter Control Register

CONTROL REGISTER BIT	STATE	BIT DEFINITION	STATE DEFINITION
TCR0	0	Internal Reset	Timer Enabled
	1		Timer in Preset State
TCR1	0	Clock Source	Timer uses External Clock ($\overline{CTC}$)
	1		Timer uses ϕ2 System Clock
TCR2	0	÷8 Prescaler Enabler	Clock is not Prescaled
	1		Clock is prescaled by ÷8 Counter
TCR3 TCR4 TCR5	× × ×	Operating Mode Selection	See Table 3
TCR6	0	Timer Interrupt Enable	$\overline{IRQ}$ Masked from Timer
	1		$\overline{IRQ}$ Enabled from Timer
TCR7	0	Timer Output Enable	Counter Output (CTO) Set "LOW"
	1		Counter Output Enabled

Table 3 Counter/Timer Operation Modes

Mode	TCR3	TCR4	TCR5	Counter Initialization	Counter Enable	Counter Clock "CC"	Interrupt Flag	
							Set	Clear
Continuous Mode	0	0	0	$\overline{G}\downarrow$ +W+R	($\overline{G}$=Low) · $\overline{R}$	CE · C	TO	RS–RT or CI
	0	1	0	$\overline{G}\downarrow$ +R	($\overline{G}$=Low) · $\overline{R}$	CE · C	TO	RS–RT or CI
Cascaded Single Shot Mode	0	0	1	$\overline{G}\downarrow$ +W+R	$\overline{R}$	CE · C	TO	RS–RT or CI
Normal Single Shot Mode	0	1	1	$\overline{G}\downarrow$ +R	$\overline{R}$	CE · C	TO	RS–RT or CI
Frequency Comparison Mode	1	0	0	$\overline{(CE+TOF\cdot CE)\cdot G\downarrow\cdot I}$ +R	CE set=$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{T}$ CE reset=W+R+I	CE · C	$\overline{G}\downarrow$ before TO	RS–RT or CI or W
	1	0	1	$\overline{G}\downarrow\cdot\overline{T}$+R	CE set=$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{T}$ CE reset=W+R+I	CE · C	$\overline{G}\downarrow$ before TO	RS–RT or CI or W
Pulse Width Comparison Mode	1	1	0	$\overline{G}\downarrow\cdot\overline{T}$+R	CE set=$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{T}\cdot G$ CE reset=W+R+I+($\overline{G}$=High)	CE · C	$\overline{G}\uparrow$ before TO	RS–RT or CI or W
	1	1	1	$\overline{G}\downarrow\cdot\overline{T}$+R	CE set=$\overline{G}\downarrow\cdot\overline{W}\cdot\overline{R}\cdot\overline{T}\cdot G$ CE reset=W+R+I+($\overline{G}$=High)	CE · C	$\overline{G}\uparrow$ before TO	RS–RT or CI or W

R = External $\overline{RES}$ or Internal Reset TCR0
W = Write Timer Latch
I = Interrupt Flag
$\overline{G}$ = $\overline{CTG}$
C = Clock selected in the internal register
$\overline{G}\downarrow$ = Negative transition of $\overline{CTG}$ signal
$\overline{G}\uparrow$ = Positive transition of $\overline{CTG}$ signal
RS–RT = Read Operation of Timer Counter after the read of Status Register (Normal operation to clear the interrupt)
CI = Counter Initialization (Internal Signal)
TOF = Time Out Flag (Set by $\overline{CI}$·TO, Reset by CI)
TO = Counter Time Out

Timer Control Register Bit 2 (TCR2) enables the ÷ 8 prescaler (TCR2 = "1"). In this mode, the clock frequency is divided by eight before being applied to the counter. When TCR2 = "0" Enable is applied directly to the counter.

TCR3, 4, 5 select the Timer Operating Mode, and are discussed in the next section.

Timer Control Register Bit 6 (TCR6) is used to mask or enable the Timer Interrupt Request. When TCR6 = "0", the Interrupt Flag is masked from the timer. When TCR6 = "1", the Interrupt Flag is enabled into Bit 7 of the Composite Status Register (Composite IRQ Bit), which appears on the $\overline{IRQ}$ output pin.

Timer Control Register Bit Seven (TCR7) has a special function when the timer is in the Cascaded Single Shot mode. (This function is explained in detail in the section describing the mode.) In all other modes, TCR7 merely acts as an output enable bit. If TCR7 = "0", the Counter Timer Output (CTO) is forced "Low". Writing a logic one into TCR7 enables CTO.

- **Timer Operating Modes**

The HD6846 has been designed to operate effectively in a wide variety of applications. This is accomplished by using three bits of the control register (TCR3, TCR4, and TCR5) to define different operating modes of the Timer, outlined in Table 3.

- **Continuous Operating Mode (TCR3 = 0, TCR5 = 0)**

The timer may be programmed to operate in a continuous counting mode by writing zeros into bits 3 and 5 of the timer control register. Assuming that the timer output is enabled (TCR7 = "1"), a square wave will be generated at the Timer Output CTO (See Table 4).

Table 4 Continuous Operating Modes

CONTINUOUS MODE
(TCR3=0, TCR7=1, TCR5=0)

CONTROL REGISTER		INITIALIZATION/OUTPUT WAVEFORMS
TCR4	Counter	CTO
0	Initialization $\overline{G}\downarrow$ +W+R	←(N+1) (T)→ ←(N+1) (T)→ ←(N+1) (T)→
1	$\overline{G}\downarrow$ +R	V_{OH} V_{OL}
		to TO* TO* TO*

$\overline{G}\downarrow$ = Negative Transition $\overline{CTG}$ Input.
W = Write Timer Latches Command.
R = Timer Reset (TCR0=1 or External $\overline{RES}$=0)
N = 16 Bit Number in Counter Latch.
T = Period of Clock Input to Counter.
to = Counter Initialization Cycle.
TO = Counter Time Out (All Zero Condition.)
* Point at which an interrupt may occur.

Either a Timer Reset (TCR0 = "1" or External $\overline{RES}$ = "0") condition or internal recognition of a negative transition of the $\overline{CTG}$ input results in Counter Initialization. A Write Timer Latches command can be selected as a Counter Initialization signal by clearing TCR4.

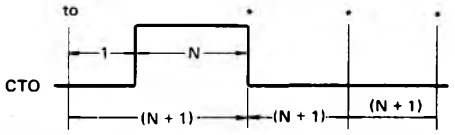
The discussion of the Continuous Mode has assumed the application requires an output signal. It should be noted the Timer operates in the same manner with the output disabled (TCR7 = "0"). A Read Timer Counter command is valid regardless of the state of TCR7.

- **Normal Single-Shot Timer Mode (TCR3 = 0, TCR4 = 1, TCR5 = 1)**

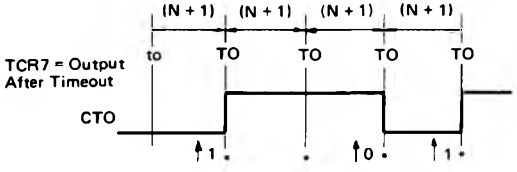
This mode is identical to the Continuous Mode with two exceptions. The first of these is obvious from the name – the output returns to a "Low" level after the initial Time Out and remains "Low" until another Counter Initialization cycle occurs. The output waveform (CTO) is shown in Figure 16.

As indicated in Figure 16, the internal counting mechanism remains cyclical in the Single-Shot Mode. Each Time Out of the counter results in the setting of an Individual Interrupt Flag and re-initialization of the counter.

The second major difference between the Single-Shot and Continuous modes is that the internal counter enable is not dependent on the $\overline{CTG}$ input level remaining in the "Low" state for the Single-Shot mode. Aside from these differences, the two modes are identical.



(A) NORMAL SINGLE-SHOT MODE OUTPUT WAVEFORM



(B) CASCADED SINGLE-SHOT MODE OUTPUT WAVEFORM

1 = Write a "1" into TCR7
0 = Write a "0" into TCR7

*Point at which an interrupt may occur.

(NOTE) All time intervals shown above assume the Gate ($\overline{CTG}$) and Clock ($\overline{CTC}$) signals are synchronized to Enable with the specified setup and hold time requirements.

Figure 16 Single-Shot Modes

- **Cascaded Single-shot Mode (TCR3=0, TCR4=0, TCR5=1)**

This mode is identical to the single-shot mode with two exceptions. First, the output waveform does not return to a "Low" level and remain "Low" after timeout. Instead, the output level remains at its initialized level until it is re-programmed and changed by timeout. The output level may be changed at any timeout or may have any number of timeouts between changes.

The second difference is the method used to change the output level. Timer Control Register Bit 7 (TCR7) has a special function in this mode. The timer output (CTO) is equal to TCR7 clocked by timeout. At every timeout, the content of TCR7 is clocked to and held at the CTO output. Thus, output pulses of length greater than one timer cycle can be generated by cascading timer cycles and counting timeouts with a software program (See Figure 16).

An interrupt is generated at each timeout. To cascade timer cycles, the MPU would need an interrupt routine to: 1) count each timeout and determine when to change TCR7: 2) write into TCR7 the state corresponding to the next desired state of the output waveform (only necessary during the last timer cycle before the output is to change state): and 3) clear the interrupt flag by reading the combination status register followed by Read Timer MSB. It is also possible, if desired, to change the length of the timer cycle by reinitializing the timer latches. This allows more flexibility for obtaining desired times.

- **Time Interval Modes (TCR3 = 1)**

The Time Interval Modes are provided for applications requiring more flexibility of interrupt generation and Counter Initialization. The Interrupt Flag is set in these modes as a function of both Counter Time Out and transitions of the $\overline{CTG}$ input. Counter Initialization is also affected by Interrupt Flag status. The output signal is not defined in any of these modes. Other features of the Time Interval Modes are Outlined in Table 5.

- **Frequency Comparison Mode (TCR3 = 1, TCR4 = 0)**

The timer within the HD6846 may be programmed to compare the period of a pulse (giving the frequency after calculations) at the $\overline{CTG}$ input with the time period required for Counter Time Out. A negative transition of the $\overline{CTG}$ input enables the counter and starts a Counter Initialization cycle – provided that other conditions as noted in Table 3 are satisfied. The counter decrements on each clock signal recognized during or after Counter Initialization until an Interrupt is generated, a

Table 5 Time Interval Modes

TCR3 = 1			
TCR4	TCR5	APPLICATION	CONDITION FOR SETTING INDIVIDUAL INTERRUPT FLAG
0	0	Frequency Comparison	Interrupt Generated if $\overline{CTG}$ Input Period (1/F) is Less Than Counter Time Out (TO).
0	1	Frequency Comparison	Interrupt Generated if $\overline{CTG}$ Input Period (1/F) is Greater Than Counter Time Out (TO).
1	0	Pulse Width Comparison	Interrupt Generated if $\overline{CTG}$ Input "Down Time" is Less Than Counter Time Out (TO).
1	1	Pulse Width Comparison	Interrupt Generated if $\overline{CTG}$ Input "Down Time" is Greater Than Counter Time Out (TO).

Write Timer Latches command is issued, or a Timer Reset condition occurs. It can be seen from Table 3 that an interrupt condition will be generated if TCR5 = "0" and the period of the pulse (single pulse or measured separately repetative pulses) at the $\overline{CTG}$ input is less than the Counter Time Out period. If TCR5 = "1", an interrupt is generated if the reverse is true.

Assume now with TCR5 = "1" that a Counter Initialization has occurred and that the $\overline{CTG}$ input has returned "Low" prior to Counter Time Out. Since there is no Individual Interrupt Flag generated, this automatically starts a new Counter Initialization Cycle. The process will continue with frequency comparison being performed on each $\overline{CTG}$ input cycle until the mode is changed, or a cycle is determined to be above the predetermined limit.

- **Pulse Width Comparison Mode (TCR3 = 1, TCR4 = 1)**

This mode is similar to the Frequency Comparison Mode except for the limiting factor being a positive, rather than negative, transition of the $\overline{CTG}$ input. With TCR5 = "0", an Individual Interrupt Flag will be generated if the zero level pulse applied to the $\overline{CTG}$ input is less than the time period required for Counter Time Out. With TCR5 = "1", the interrupt is generated when the reverse condition is true.

As can be seen in Table 3, a positive transition of the $\overline{CTG}$ input disables the counter. With TCR5 = "0", it is therefore possible to directly obtain the width of any pulse causing an interrupt.

- **Composite Status Register**

The Composite Status Register (CSR) is a read-only register which is shared by the Timer and the Peripheral Data Port of the HD6846. Three individual interrupt flags in the register are set directly via the appropriate conditions in the timer or peripheral port. The composite interrupt flag – and the $\overline{IRQ}$ Output – respond to these individual interrupts only if corresponding enable bits are set in the appropriate Control Registers. (See Figure 17.) The sequence of assertion is not detected. Setting TCR6 while CSR0 is "High" will cause CSR7 to be set, for example.

The Composite Interrupt Flag (CSR7) is clear only if all enabled Individual Interrupt Flags are clear. The conditions for clearing CSR1 and CSR2 are detailed in a later section. The Timer Interrupt Flag (CSR0) is cleared under the following conditions:

1) Timer Reset – Internal Reset Bit (TCR0) = "1" or External $\overline{RES}$ = "0".
2) Any Counter Initialization condition.
3) A Write Timer Latches command if Time Interval modes (TCR3 = "1") are being used.
4) A Read Timer Counter command, provided this is preceded by a Read Composite Status Register while CSR0 is set. This latter condition prevents missing an Interrupt Request generated after reading the Status Register and prior to reading the counter.

The remaining bits of the Composite Status Register (CSR3~CSR6) are unused. They default to a logic zero when read.

■ I/O OPERATION

- **Parallel Peripheral Port**

The peripheral port of the HD6846 contains 8 Peripheral Data lines (P_0~P_7), two Peripheral Control lines (CP_1 and CP_2), a Data Direction Register, a Peripheral Data Register, and a Peripheral Control Register. The port also directly affects two bits (CSR1 and CSR2) of the Composite Status Register.

The Peripheral Port is similar to the "B" side of a PIA (HD6821) with the following exceptions:

1) All registers are directly accessible in the HD6846 Data Direction and Peripheral Data in the HD6821 are located at the same address with Bit Two of the Control Register used for register selection.
2) Peripheral Control Register Bit Two (PCR2) of the HD6846 is used to select an optional input latch function. This option is not available with HD6821 PIA's.
3) Interrupt Flags are located in the HD6846 composite status register rather than Bits 6 and 7 of the Control Register as used in the HD6821.
4) Interrupt Flags are cleared in the HD6821 by reading data from the Peripheral Data Register. HD6846 Interrupt Flags are cleared by either reading or writing to the Peripheral Data Register – provided that this sequence is followed a) Flag Set, b) Read Composite Status Register, c) Read/Write Peripheral Data Register is followed.

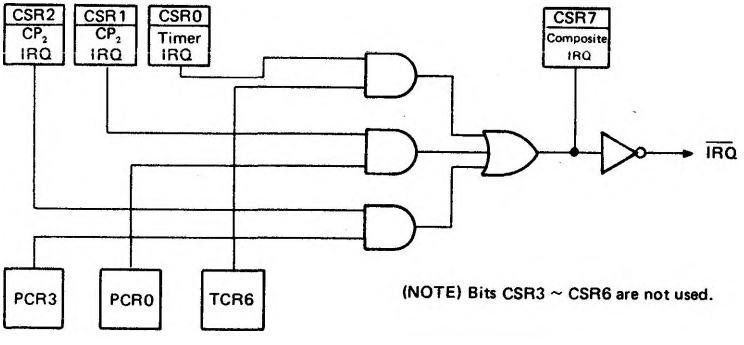


Figure 17 Composite Status Register & Associated Logic

5) Bit 6 of the HD6846 Peripheral Control Register is not used. Bit 7 (PCR7) is an Internal Reset Bit not available on the HD6821.
6) The Peripheral Data lines (and CP_2) of the HD6846 feature internal current limiting which allows them to directly drive the base of Darlington NPN transistors.

- **Data Direction Register**

The MPU can write directly to this eight-bit register to configure the Peripheral Data lines as either inputs or outputs. A particular bit within the register (DDRn) is used to control the corresponding Peripheral Data line (Pn). With DDRn = "0", Pn becomes an input; if DDRn = "1", Pn is an output. As an example, writing Hex $0F into the Data Direction Register results in P_0 thru P_3 becoming outputs and P_4 thru P_7 being inputs. Hex $55 in the Data Direction Register results in alternate outputs and inputs at the parallel port.

- **Peripheral Data Register**

This eight-bit register is used for transferring data between the peripheral data port and the MPU. Any bit corresponding to an output line will be used to drive the output buffer associated with that line. Data in these output bits is normally provided by an MPU Write function. (Input bits – those associated with input lines – are unchanged by a Write Command.) Any input bit will reflect the state of the associated input line if the input latch function is deselected. If the Control Register is programmed to provide input latching, the input bit will retain the state at the time CP_1 was activated until the Peripheral Data Register is read by the MPU.

- **Peripheral Control Register**

This eight-bit register is used to control the reset function as well as for selection of optional functions of the two peripheral control lines (CP_1 and CP_2). The Peripheral Control Register functions are outlined in Table 6.

- **Peripheral Port Reset (PCR7)**

Bit 7 of the Peripheral Control Register (PCR7) may be used to initialize the peripheral section of the HD6846. When this bit is set "High", the peripheral data register, the peripheral data direction register, and the interrupt flags associated with the peripheral port (CSR1 & CSR2) are all cleared. Other bits in the peripheral control register are not affected by PCR7.

PCR7 is set by either a logic zero at the External $\overline{RES}$ input or under program control by writing a "1" into the location. In any case, PCR7 may be cleared only by writing a zero into the location while $\overline{RES}$ is "High". The bit must be cleared to activate the port.

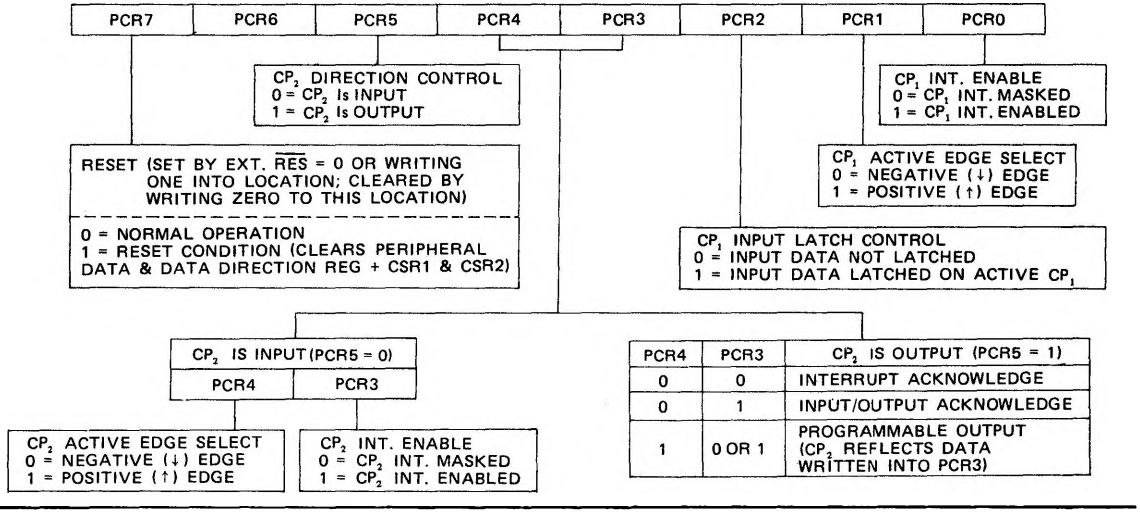
- **Control of CP_1 Peripheral Control Line**

CP_1 may be used an interrupt request to the HD6846, as a strobe to allow latching of input data, or both. In any case, the input can be programmed to be activated by either a positive or negative transition of the signal. Thes options are selected via Control Register Bits PCR0, RCR1 & PCR2.

Control Register Bit 0 (PCR0) is used to enable the interrupt transfer circuitry of the HD6846. Regardless of the state of PCR0, and active transition of CP_1 causes the Composite Status Register Bit One (CSR1) to be set. if PCR0 = "1", this interrupt will be reflected in the Composite Interrupt Flag (CSR7), and thus at the $\overline{IRQ}$ output. CSR1 is cleared by a Peripheral Port Reset condition or by either reading or writing to the peripheral data register after the Composite Status Register is read. The latter alternative is conditional – CSR1 must have been a logic one when the Composite Status Register was lart read. This precludes inadvertent clearing of interrupt flags generated between the time the Status Register is read and the manipulation of peripheral data.

Control Register Bit One (PCR1) is used to select the edge which activates CP_1. When PCR1 = "0", CP_1 is active on negative transitions ("High" to "Low"). "Low" to "High" transitions are sensed by CP_1 when PCR1 = "1"

Table 6 Peripheral Control Register Format (Expanded)



In addition to its use as an interrupt input, CP_1 can be used as a strobe to capture input data in an internal latch.This option is selected by writing a one into Peripheral Control Register Bit Two (PCR2). In operating, the data at the pins designated by the Data Direction Register as inputs will be captured by an active transition of CP_1. An MPU Read of the Peripheral Data Register will result in the captured data being transferred to the MPU – and it also releases the latch to allow capture of new data. Note that successive active transitions with no Read Peripheral Data Command between does not update the input latch. Also, it should be noted that use of the input latch function (which can be deselected by writing a zero into PCR2) has no effect on output data. It also does not affect Interrupt function of CP_1.

- **Control of CP_2 Peripheral Control Line**

CP_2 may be used as an input by writing a zero into PCR5. In this configuration, CP_2 becomes a dual of CP_1 in regard to generation of interrupts. An active transition (as selected by PCR4) causes Bit Two of the Composite Status Register to be set. PCR3 is then used to select whether the CP_2 transition is to cause CSR7 to be set – and thereby cause $\overline{IRQ}$ to go "Low". CP_2 has no effect on the input latch function of the HD6846.

Writing a one into PCR5 causes CP_2 to function as an output. PCR4 then determines whether CP_2 is to be used in a handshake or programmable output mode. With PCR4 = "1", CP_2 will merely reflect the data written into PCR3. Since this can readily be changed under program control, this mode allows CP_2 to be a programmable output line in much the same manner as those lines selected as outputs by the Data Direction Register.

The handshaking mode (PCR5 = "1", PCR4 = "0") allows CP_2 to perform one of two functions as selected by PCR3. With PCR3 = "1", CP_2 will go "Low" on the first Enable positive transition after a Read or Write to the Peripheral Data Register. This Input/Output Acknowledge signal is released (returns "High") on the next positive transition of the Enable signal.

In the Interrupt Acknowledge mode (PCR5 = "1", PCR4 = PCR3 = "0"), CP_2 is set when CSR1 is set by an active transition of CP_1. It is released (goes "Low") on the first positive transition of Enable after CSR1 has been cleared via an MPU Read or Write to the Peripheral Data Register. (Note that the previously described conditions for clearing CSR1 still apply.)

- **Restart Sequence**

A typical restart sequence for the HD6846 will include initialization of both the Peripheral Control & Data Direction Registers of the parallel port. It is necessary to set up the Peripheral Control Register first, since PCR7 = "0" is a condition for writing data into the Data Direction Register. (A logic zero at the external $\overline{RES}$ input automatically sets PCR7.)

- **Summary**

The HD6846 has several optional modes of operation which allow it to be used in a variety of applications. The following tables are provided for reference in selecting these modes.

Table 7 HD6846 Internal Register Addresses

R/$\overline{W}$	A_2	A_1	A_0	REGISTER SELECTED
R	0	0	0	Combination Status Register
R/$\overline{W}$	0	0	1	Peripheral Control Register
R/$\overline{W}$	0	1	0	Data Direction Register
R/$\overline{W}$	0	1	1	Peripheral Data Register
R	1	0	0	Combination Status Register
R/$\overline{W}$	1	0	1	Timer Control Register
R/$\overline{W}$	1	1	0	Timer MSB Register
R/$\overline{W}$	1	1	1	Timer LSB Register
R	x	x	x	ROM Address

Table 8 Composite Status Register

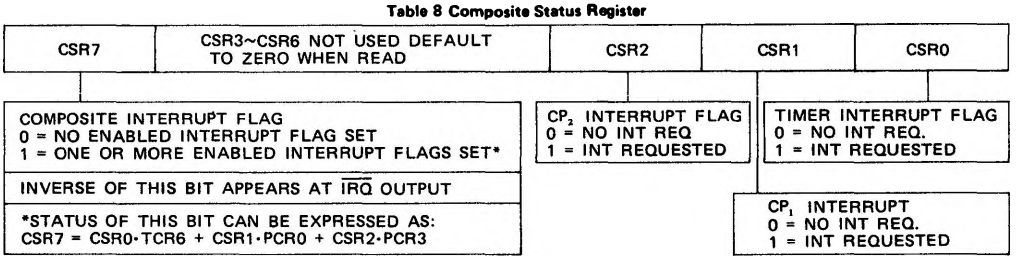
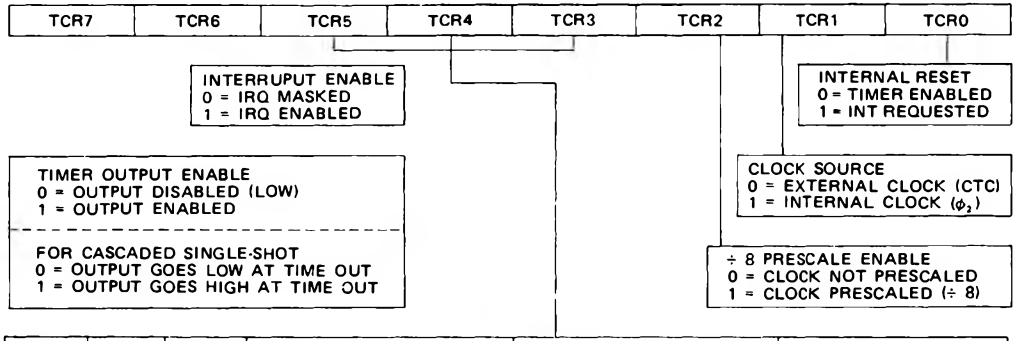


Table 9 Timer Control Register

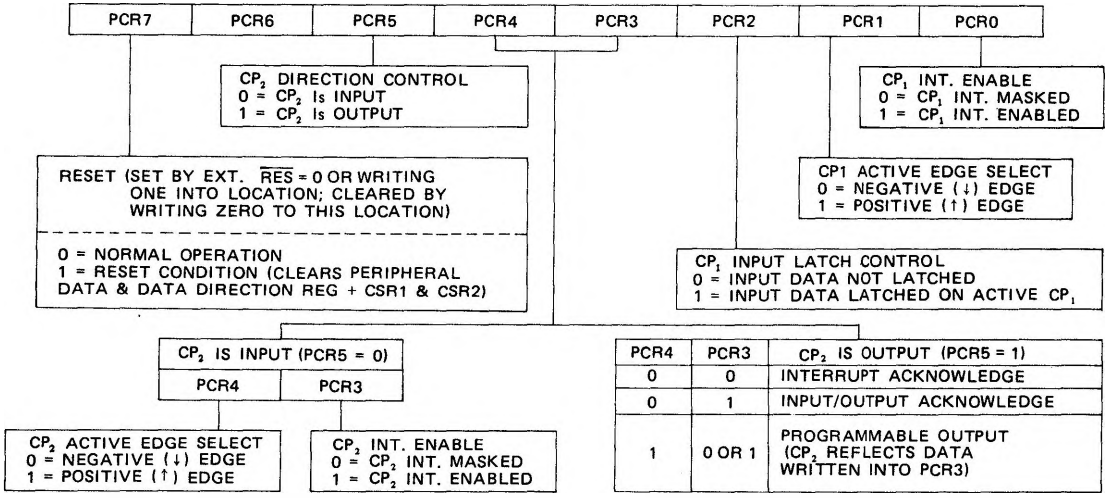


TCR3	TCR4	TCR5	TIME OPERATING MODE	COUNTER INITIALIZATION	INTERRUPT FLAG SET
0	0	0	CONTINUOUS	$\overline{CTG}\downarrow + W + R$	TO
0	0	1	CASCADED SINGLE SHOT	$\overline{CTG}\downarrow + W + R$	TO
0	1	0	CONTINUOUS	$\overline{CTG}\downarrow + R$	TO
0	1	1	NORMAL SINGLE SHOT	$\overline{CTG}\downarrow + R$	TO
1	0	0	FREQUENCY COMPARISON	$\overline{CTG}\downarrow \cdot \overline{I} \cdot (\overline{CE} + TOF \cdot CE) + R$	CTG ↓ BEFORE TO
1	0	1		$\overline{CTG}\downarrow \cdot \overline{I} + R$	TO BEFORE CTG
1	1	0	PULSE WIDTH COMPARISON	$\overline{CTG}\downarrow \cdot \overline{I} + R$	$\overline{CTG}\uparrow$ BEFORE TO
1	1	1		$\overline{CTG}\downarrow \cdot \overline{I} + R$	TO BEFORE $\overline{CTG}\uparrow$

R = RESET CONDITION
W = WRITE TIMER LATCHES
TO = COUNTER TIME OUT
CE = COUNTER ENABLE

$\overline{CTG}\downarrow$ = NEG TRANSISTION OF PIN 17
$\overline{CTG}\uparrow$ = POS TRANSITION OF PIN 17
$\overline{I}$ = INTERRUPT FLAG (CSR0) = 0

Table 10 Peripheral Control Register



PCR4	PCR3	CP2 IS OUTPUT (PCR5 = 1)
0	0	INTERRUPT ACKNOWLEDGE
0	1	INPUT/OUTPUT ACKNOWLEDGE
1	0 OR 1	PROGRAMMABLE OUTPUT (CP2 REFLECTS DATA WRITTEN INTO PCR3)

■ CUSTOM PROGRAMMING

By the programming of a single photomask for the HD6846, the customer may specify the content of the memory and the method of enabling the outputs.

Information on the general options of the HD6846 should be submitted on an Organizational Data form such as that shown in Figure 18 and Figure 19.

Information for custom memory content may be sent to HITACHI in one of two forms (shown in order of preference):

1) Paper tape output of the HMCS6800 Load Module Format or of the BNPF Format
2) Hexadecimal coding using IBM Punch Cards

ORGANIZATIONAL DATA
HD6846 COMBINATION ROM-I/O-TIMER

Customer:

Company _____

Part No. _____

Originator _____

Phone No. _____

Hitachi Use Only:

Quote:

Part No.:

Specif. No.:

Enable Options: (ROM ENABLE MUST DIFFER FROM I/O-TIMER)

	ROM SECTION 1	ROM SECTION 0	I/O-TIMER SECTION 1	I/O-TIMER SECTION 0
CS_0	☐	☐	☐	☐
CS_1	☐	☐	☐	☐

I/O-TIMER SELECT A_6 1 0 x		CHECK ONE COLUMN ONLY				
	A_{10}	x	1	x	x	x
	A_9	x	x	1	x	x
	A_8	x	x	x	1	x
	A_7	x	x	x	x	1

1 ≥ 2.0V
0 ≤ 0.8V
x = NOT USED

Figure 18 Format for Programming General Options

DATE

COMPANY ENGINEER SECTION

CUSTOMERS P/N (if you need) ☐☐☐☐☐☐☐☐☐☐☐☐

TYPE NO. OF ROM

DATA FORMAT
1. HMCS6800 load module format 2. BNPF format

coding media
☐ 1. paper tape
☐ 2. IBM 80 column card

total bytes of data (decimal)

initial ROM address (decimal)

parity (for paper tape)
☐ 1. even ☐ 2. odd ☐ 3. none

total number of cards

for HITACHI reference only

ref. No.	
mask ROM No.	
processed data	
approved data	

designed

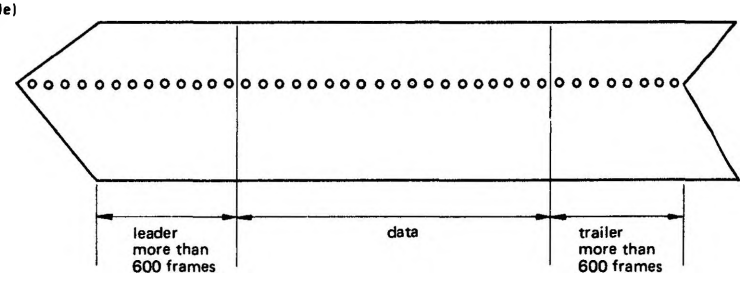
approved

Figure 19 Confirmation sheet of specification for HD6846 series ROM

■ PAPER TAPE

1) Any one inch width tape usually available in market can be used but tape in black color is recommended.
2) Both leader and trailer have more than 600 frames.

(Example)



3) One file data of each chip shall be contained in one reel of paper tape. One file data shall not be divided into more than two reels.
4) Parity
   Parity shall be indicated in "Confirmation sheet of specification". Parity forms are grouped;
   (1) With parity EVEN or ODD
   (2) Without parity
5) 8-bit ASCII code shall be used.

■ CARD

1) Use IBM 80 column card.
2) Use EBCDIC code.
3) Card format is as follows;

column	contents
1 to 71	Free format of data column
72	Blank
73 to 80	Sequential card number, not free format. Least significant digit of decimal sequential number is located in column 80. No alphabet letters. Any sequential number more than 1 can be used.

4) Total number of cards shall be written in "Confirmation sheet of specification".

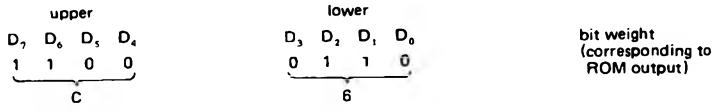
■ DATA FORMAT

• HMCS6800 LOAD MODULE FORMAT

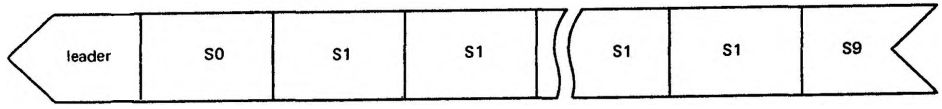
This is object format obtained from HMCS6800 assembler.

1) 8-bit code is divided into upper and lower 4 bits and transformed into hexadecimal number.

(Example) Binary number if 1100 0010 is transformed as follows.

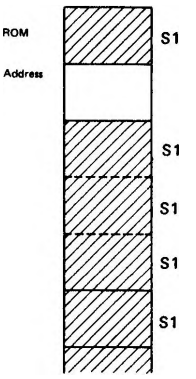


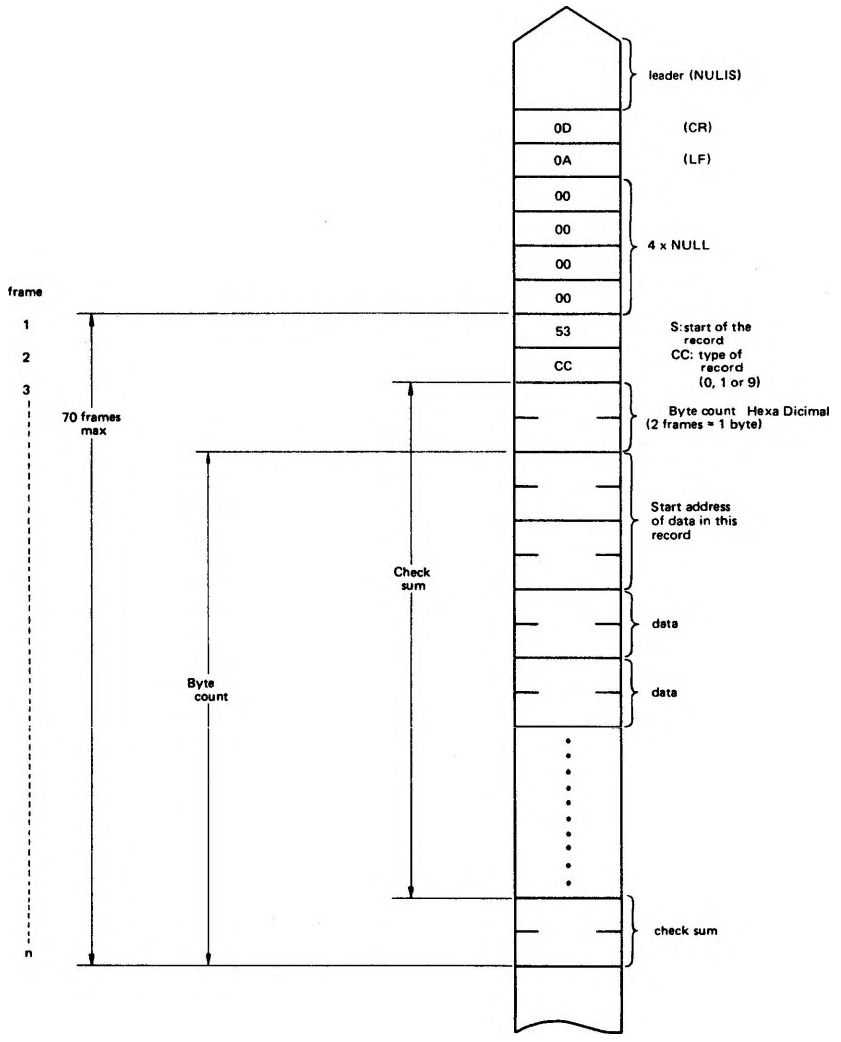
2) Load module structure of paper tape is shown as an example.



S0 is the header record, S1 is data record and S9 is end of file record. Each data record corresponds to each ROM data as shown below. Continuous memory address shall be devided into several records due to limitation of maximum frame number (70 frames = 35 bytes) in one record.

S0, S1 or S9 is distinguished by CC following start of the record S.





Check sum is complement of 1 for sum of each 8-bit.

3) Example of load module format

	frame	CC=30 header record		CC=31 data record		CC=39 end of file record	
1	start of record	53	S	53	S	53	S
2	type of record	30	0	31	1	39	9
3 4	byte count	30 36	06	31 36	16	30 33	03
5 6 7 8	start address of data in this record	30 30 30 30	0000	31 31 30 30	1100	30 30 30 30	0000
9 10	data	34 38	48-H	39 38	98	46 43	FC (check sum)
	data	34 34	44-D	30 32	02		
	data	35 32	52-R				
n	check sum	32 42	2B (check sum)	41 38	A8 (check sum)		

Check sum of header record above is complement of 1 of $(06 + 00 + 00 + 48 + 44 + 52)_{16}$ i.e., 2B.

The start address of data record is incremented for each one byte data, then is compared to the next address in data record and is checked to be sequential or not.

When it is not sequential, hexadecimal 00 is filled as data for that address automatically.

A example of type out of paper tape in HMCS6800 load module format is shown below.

```
header record ... S00600004844522B
data record .....S113F0007EF5587EF7897EFAA77EF9C07EF9C47E24
data record .....S112F010FA657EFA8B7EFAA07EF9DC7EFA247E06
end of file record .... S9030000FC
```

4) Four types of data of ROM code are able to be processed. In any case, header record before data record is needed and so as end of file record after data record.

(a) No vacancy in ROM

Data record is filled with full ROM record of one chip. Therefore address is sequential. Initial ROM address in "Confirmation sheet of specification" is 0.

(b) Vacancy in former part of ROM

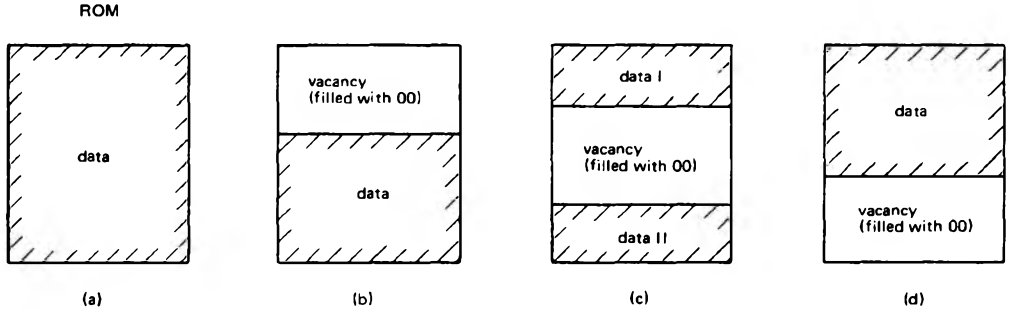
Desired initial address shall be filled in initial ROM address column in "Confirmation sheet of specification". Data of 00 are filled automatically for vacant address.

(c) Vacancy in the middle of ROM

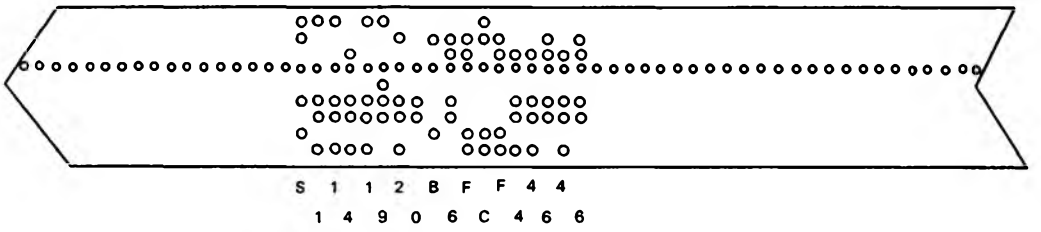
Data of 00 are filled in for vacant address. Initial ROM address for data I is 0 and desired initial address for data II shall be described in "Confirmation sheet of specification".

(d) Vacancy in later part of ROM

When end of file record is read out, data of 00 are filled in thereafter.



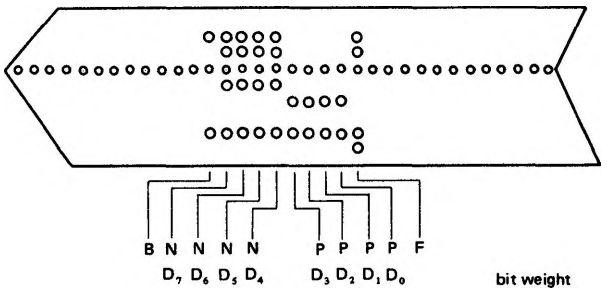
(Example) Paper tape whose data record is S1141920B6FC .....



- **BNPF format**

1) Each word is expressed as BNPF slice which begins word opening mark B, has 8 character bit contents shown by P or N and finishes with end mark F.

(Example) 0F in hexadecimal code is expressed as shown below (paper tape).



2) Any contents between F of the first slice and B of next slice are disregarded.
3) Bit pattern (BNPF) slice for all ROM address shall be indicated. Initial ROM address in "Confirmation sheet of specification" is, therefore always 0 for BNPF.
   - B ..... shows beginning of the word
   - N ..... shows 0 of one bit data
   - P ..... shows 1 of one bit data
   - F ..... shows end of the word

Note 1) X can be used expect for P and N for indication of word contents of BNPF slice. This X means that bit can be either P or N (don't care). X shall be determined by HITACHI for testing and shall be informed to the customer in confirmation table.

Note 2) Expression of B*nF can be used for indicating that the same contents of foregoing slice are applicable from this word to following n words.
For example, when B*4F is indicated at 10th word position, the contents of 9th word are repeated for 10, 11, 12 and 13th word.
(Content of X is not always repeated even in this case.)
n is grater than 1 and less than final address of ROM.

Note 3) When vacancy of ROM exists, combination of Note 1) and Note 2) is usefull.

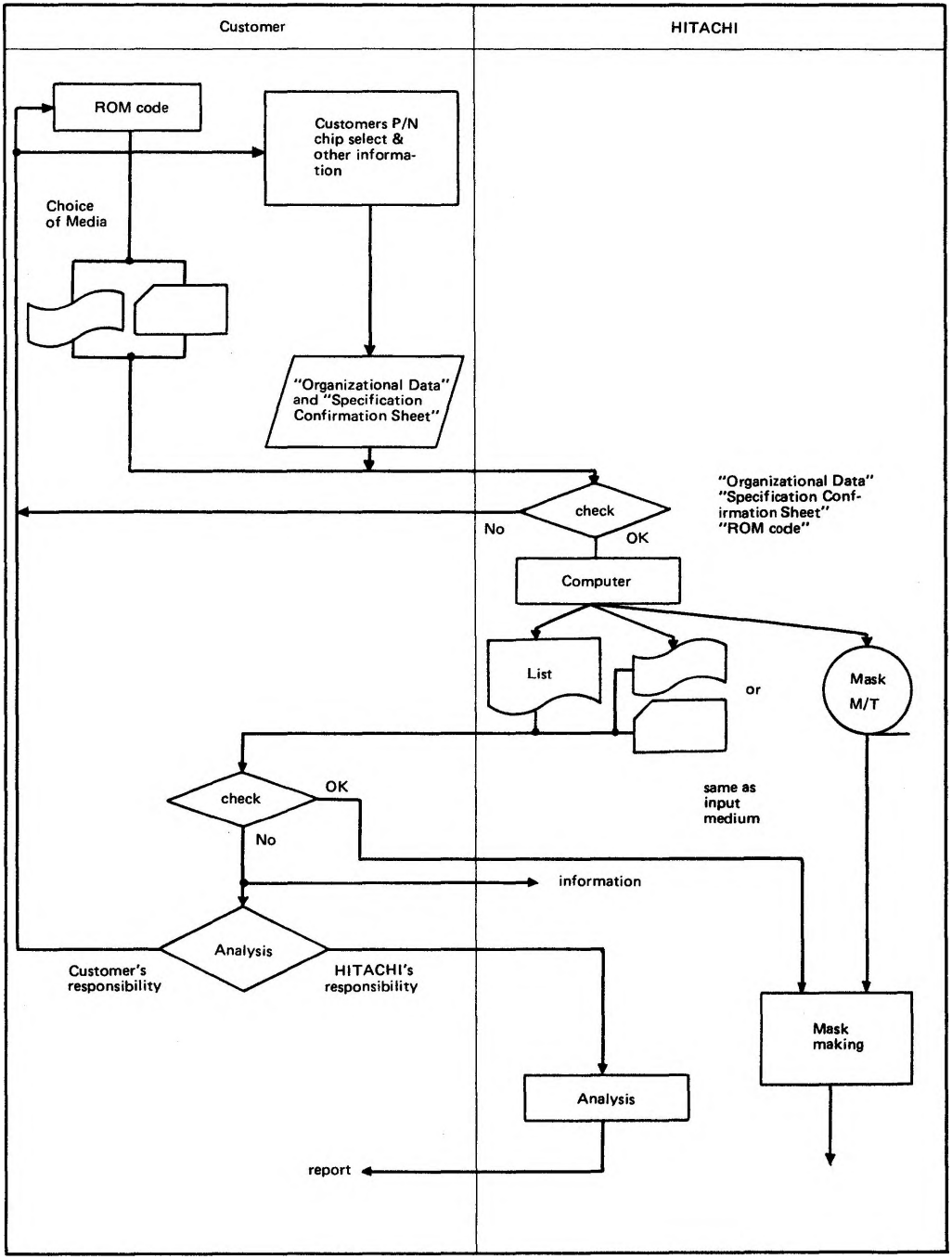


Figure 20 Flow chart of Mask ROM Development