FAIRCHILD
A Schlumberger Company

F6846
ROM-I/O-Timer

Microprocessor Product

Description

The F6846 combination chip provides the means, in conjunction with the F6802, to develop a basic 2-chip microcomputer system. The F6846 consists of 2048 bytes of mask-programmable Read Only Memory (ROM), an 8-bit bidirectional data port with control lines, and a 16-bit programmable timer-counter.

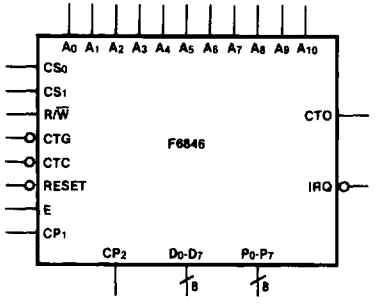
This device is capable of interfacing with the F6802 (basic F6800, clock and 128 bytes of RAM) as well as the F6800 if desired. No external logic is required to interface with most peripheral devices.

- **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 F6800 Microcomputer Product Family**
- **TTL-Compatible Data and Peripheral Lines**
- **Single 5 V Power Supply**

Pin Names

E	Enable (Clock System $\phi 2$) Input
A_0-A_{10}	Address Inputs
CS_0, CS_1	Chip Select Inputs
$R/\overline{W}$	Read/Write Input
$\overline{CTG}$	Counter Gate Input
$\overline{CTC}$	External Clock Input
$\overline{RESET}$	Reset Input
CP_1	Peripheral Interrupt Input
CP_2	Bidirectional Peripheral Control
D_0-D_7	Bidirectional Data Lines
P_0-P_7	Bidirectional Peripheral Data Lines
$\overline{CTO}$	Counter Timer Output
$\overline{IRQ}$	Interrupt Request Output

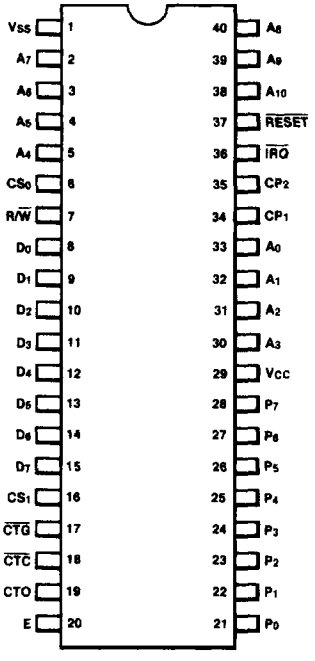
Logic Symbol



V_{CC} = Pin 29
V_{SS} = Pin 1

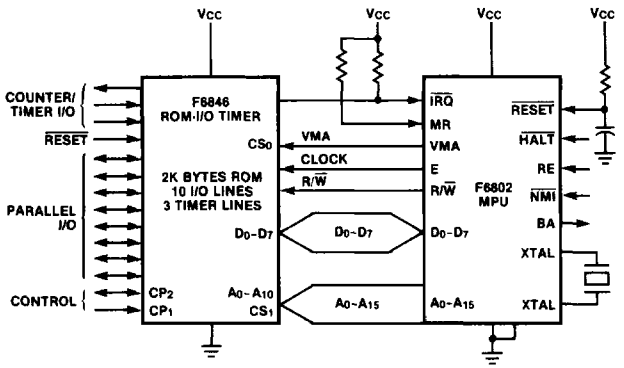
5

Connection Diagram
40-Pin DIP

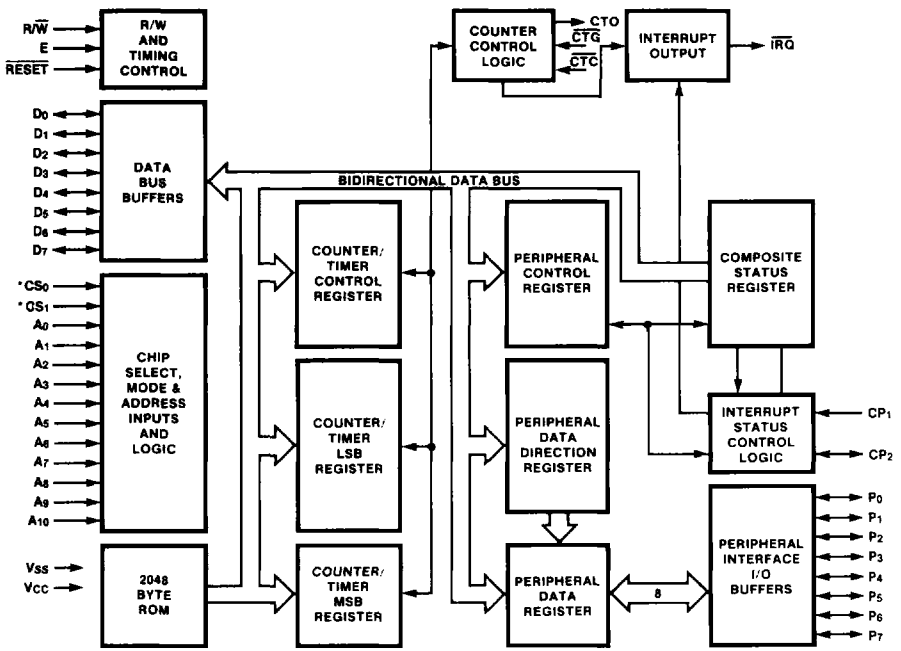


(Top View)

Typical Microcomputer



Block Diagram



*Mask Programmable

Functional Description

The F6846 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 F6800 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-A_{10} allow any of the 2048 bytes of ROM to be uniquely addressed. Bidirectional data lines (D_0-D_7) allow the transfer of data between the MPU and the F6846.

Timer-Counter Functions

Under software control, this 16-bit binary counter may be programmed to count events, measure frequencies and time intervals, or similar tasks. It may also be used for square wave generation, single pulses of control duration and gated delayed 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 and selection of an internal or external clock source, a divide-by-eight prescaler, and operating mode. The Counter-Timer Clock ($\overline{CTC}$) will accept an asynchronous pulse to decrement the internal register for the counter-timer. If the divide-by-eight prescaler is used, the maximum clock rate can be four times the master clock frequency with an absolute maximum of 4 HMz. Counter Gate input ($\overline{CTG}$) accepts an asynchronous TTL-compatible signal that may be used as a trigger or gating function to the counter-timer. The 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 characteristics similar to the B port on the F6821 PIA. This includes eight bidirectional data lines and two handshake control signals. The control and operation of these lines are completely software programmable. Internal registers associated with the I/O functions may be selected with A_0, A_1 and A_2.

The Peripheral Interrupt input (CP_1) will set the interrupt flag (CSR_1) of the Composite Status Register. The Peripheral Control (CP_2) may be programmed to act as an interrupt input (set CSR_2) or as a Peripheral Control output.

Pin Functions

Bus Interface

The F6846 interfaces to the F6800 bus via an 8-bit bidirectional data bus, two chip select lines, a read/write line, and 11 address lines. These signals, in conjunction with the F6800 VMA Output, permit the MPU to control the F6846.

Bidirectional Data Bus (D_0-D_7)

The bidirectional data lines (D_0-D_7) allow the transfer of data between the MPU and the F6846. The data bus output drivers are 3-state devices that remain in the high-impedance (OFF) state except when the MPU performs an F6846 register or ROM read (R/$\overline{W}$ HIGH 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 F6846. They are mask programmed to be active HIGH or active LOW as specified by the user.

Address (A_0-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 Circuitry for additional details.)

Reset ($\overline{RESET}$)

The active LOW state of the $\overline{RESET}$ 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 timer and peripheral register reset conditions.)

Enable (E)

This signal synchronizes data transfer between the MPU and the F6846. It also performs an equivalent synchronization function on the External Clock, $\overline{RESET}$ and Counter Gate inputs of the F6846 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 lines. A LOW level on the R/$\overline{W}$ input enables the F6846 input buffers and data is transferred to the circuit during the $\phi 2$ 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 $\phi 2$ 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 F6846. This output utilizes an open drain configuration and permits interrupt request outputs from other circuits to be connected in a wire-OR configuration.

Peripheral Data (P_0-P_7)
The Peripheral Data lines can be individually programmed as either inputs or outputs via the Peripheral 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 V (logic HIGH output).

When programmed as inputs, the output drivers associated with these lines enter a 3-state (high-impedance) mode. Since there is no internal pull-up for these lines, they represent a maximum 10 μA load to the circuitry driving them, regardless of logic state.

A logic LOW at the $\overline{RESET}$ input forces the Peripheral Data lines to the input configuration by clearing the Peripheral 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.

Peripheral Interrupt (CP_1)
Peripheral Interrupt input CP_1 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.

Peripheral Control (CP_2)
Peripheral Control 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 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. CP_2 is programmed by the Peripheral Control Register.

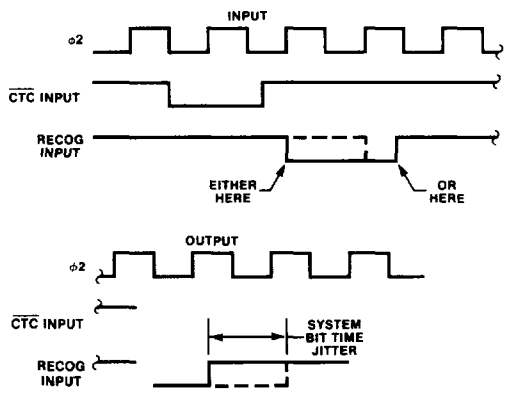
Counter Timer (CTO)
The Counter Timer output is software programmed by selected bits in the Counter/Timer Control Register. The mode of operation is dependent on the Counter/Timer Control Register, the Counter Gate input, and the clock source. The output is TTL compatible.

External Clock ($\overline{CTC}$)
$\overline{CTC}$ 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 set-up and hold times for the inputs. The asynchronous clock rate can vary from dc to the limit imposed by system $\phi 2$, set-up and hold times.

The External Clock input is clocked in by enable (system $\phi 2$) 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 F6846. All references to $\overline{CTC}$ inputs in this document relate to internal recognition of the input transition. Note that a clock transition that does not meet set-up 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 F6846 when using an asynchronous clock and gate input signal. System jitter is the result of input signals out of synchronization with the system $\phi 2$ clock (Enable), permitting signals with marginal set-up and hold time to be recognized within either the bit-time nearest the input transition or subsequent bit-time.

Input jitter can be as great as the time between input signal negative-going transitions plus the system jitter if the first transition is recognized during one system cycle, and not recognized the next cycle or vice-versa.

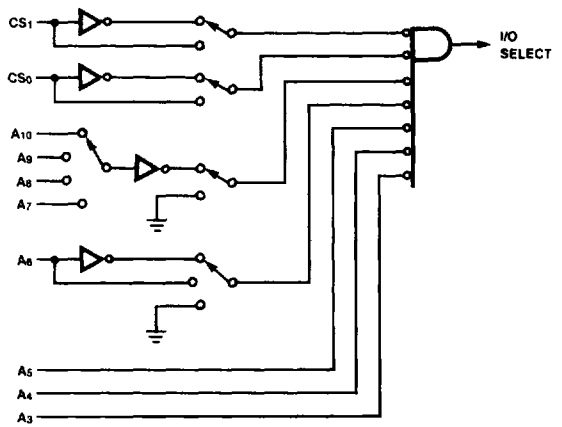


Counter Gate ($\overline{CTG}$)
$\overline{CTG}$ accepts an asynchronous TTL-compatible signal which is used as a trigger or a clock-gating function to the timer. The gating input is clocked into the F6846 by the Enable (system $\phi 2$) signal in the same manner as the previously discussed clock inputs. A $\overline{CTG}$ transition is recognized on the fourth enable pulse, provided set-up and hold time requirements are met. The HIGH or LOW levels of the $\overline{CTG}$ input must be stable for at least one system clock period plus the sum of set-up and hold times. All references to $\overline{CTG}$ transition in this document relate to internal recognition of the input transition.

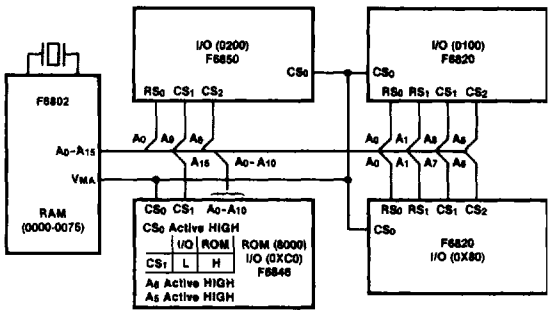
The $\overline{CTG}$ input of the timer directly affects the internal 16-bit counter. The operation of $\overline{CTG}$ is therefore independent of the divide-by-eight prescaler selection.

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, $A_3 = A_4 = A_5 =$ logic L. A_6 can be programmed to an H, L, or don't care (X). $A_7 = A_8 = A_9 = A_{10} =$ don't care or one line only may be programmed to a logic H. The available chip select options are diagramed below.



Memory mapping the I/O can be accomplished by using one of the CS inputs to select between ROM and I/O, applying the F6802 VMA output to the other CS (programmed active HIGH) and using the Address lines to decode Address fields.



Internal Addressing
Seven I/O register locations within the F6846 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. CS_0 and CS_1 must be in the I/O state and the proper register address must be applied to access a particular register. The Composite Status Register is read only, where all other registers are read/write.

Table 1 Internal Register Addresses

Register Selected	A_2	A_1	A_0
Composite Status Register	L	L	L
Peripheral Control Register	L	L	H
Peripheral Data Direction Register	L	H	L
Peripheral Data Register	L	H	H
Composite Status Register	H	L	L
Counter/Timer Control Register	H	L	H
Counter/Timer MSB Register	H	H	L
Counter/Timer LSB Register	H	H	H
ROM Address	X	X	X

L = LOW Voltage Level
H = HIGH Voltage Level
X = Don't Care

Initialization

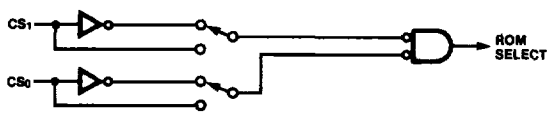
When the RESET input has accepted a LOW signal, all registers are initialized to the reset state. The Peripheral Data Direction and Peripheral Data Registers are cleared. The Peripheral Control Register is cleared except for bit 7 (the reset bit). This forces the parallel port to the input mode with interrupts disabled. To remove the reset condition from the parallel port, an L must be written into the Peripheral Control Register bit 7 (PCR_7).

During initialization the counter latches are preset to their maximal count, the Counter/Timer Control Register bits are reset to L except for bit 0 (TCR_0 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 Composite Status Register is cleared of all interrupt flags. During timer initialization the reset bit (TCR_0) must be cleared.

ROM Select Circuitry

The mask-programmable ROM section is similar in operation to other ROM products of the F6800 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.

The active levels of CS_0 and CS_1 for ROM and I/O select are a user programmable option. Either CS_0 and 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 below.



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 F6800 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 system $\phi 2$ 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 latches store a binary equivalent of the desired count value minus one. Counter initialization results in the transfer of the latch contents to 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 bits of the latches are updated simultaneously. Since the F6846 data bus is eight bits wide, a temporary register (MSB buffer register) is provided for 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 = L; A_1 = A_2 = H).

The lower eight bits 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 Counter/Timer LSB Register is selected (A_0 = A_1 = A_2 = H). Data from the MSB buffer will be transferred automatically 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 of this operation will be referred to henceforth as a *write-timer-latches command.*

The F6846 has been designed to allow transfer of two bytes of data into the counter latches from any source, provided that the MSB is transferred first. In many applications, the source of the data will be an F6800 MPU. It should therefore be noted that the 16-bit store operations of the F6800 family microprocessors (STS and STX) transfer data in the order required by the F6846. A store index register instruction, for example, results in the MSB of the X register being transferred to the selected address, then the LSB of the X register being written into the next higher location. Thus, either the index register or stack pointer contents may be transferred directly into either 8-bit latch with a single instruction.

A logic L at the $\overline{RESET}$ input also initializes the counter latches. All latches will assume maximum count (65,536) values. It is important to note that an internal reset (bit zero of the Counter/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 $\overline{RESET}$ = L or TCR_0 = H) is recognized. It can also occur depending 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 L state. In this case, data is transferred from the latches to the counter, but the interrupt flag is unaffected.

Counter/Timer Control Register
The Counter/Timer Control Register (see *Table 2*) in the F6846 is used to modify timer operation to suit a variety of applications. The Counter/Timer Control Register has a unique address space (A_0 = H, A_1 = L, A_2 = H) and therefore may be written into at any time. The least significant bit of this control register is used as an internal reset bit. When this bit is a logic L, all timers are allowed to operate in the modes prescribed by the remaining bits of the control registers.

Writing H into Counter/Timer Control Register bit 0 (TCR_0) causes the counter to be preset with the contents of the counter latches, all counter clocks to be disabled, and the timer output and interrupt flag (status register) to be reset. The counter latch and Counter/Timer Control Register are undisturbed by an internal reset and may be written into regardless of the state of TCR_0.

Counter/Timer Control Register bit 1 (TCR_1) is used to select the clock source. When TCR_1 = L, the external clock input $\overline{CTC}$ is selected, and when TCR_1 = H, the timer uses system $\phi 2$.

Counter/Timer Control Register bit 2 (TCR_2) enables the divide-by-eight prescaler (TCR_2 = H). In this mode, the clock frequency is divided by eight before being applied to the counter. When TCR_2 = L the clock is applied directly to the counter.

TCR_3, TCR_4, and TCR_5 select the timer operating mode, and are discussed in the next section.

Counter/Timer Control Register bit 6 (TCR_6) is used to mask or enable the timer interrupt request. When TCR_6 = L, the interrupt flag is masked from the timer. When TCR_6 = H, the interrupt flag is enabled into bit 7 of the Composite Status Register (composite IRQ bit), which appears on the $\overline{IRQ}$ output.

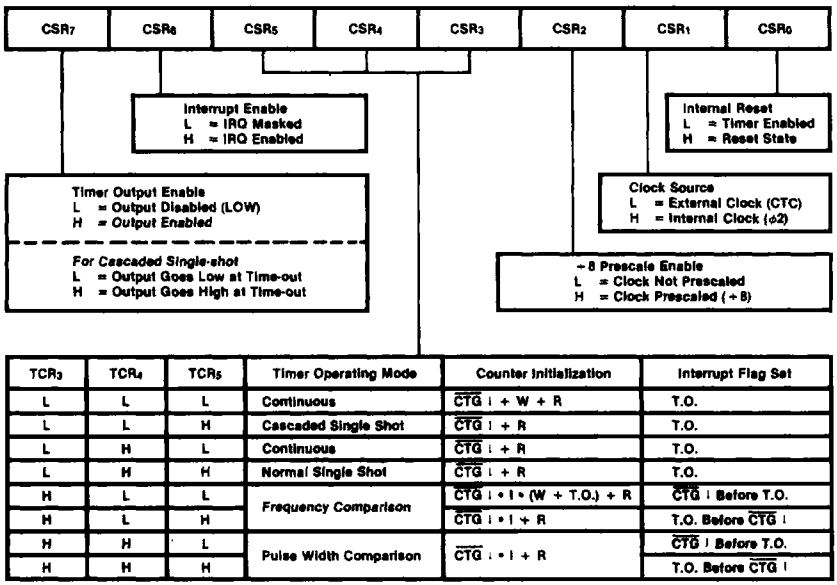
Counter/Timer Control Register bit 7 (TCR_7) has a special function when the timer is in the cascaded single-shot mode. (This function is explained in detail in the section describing the mode.) In all other modes, TCR_7 merely acts as an output enable bit. If TCR_7 = L, the Counter Timer Output (CTO) is forced LOW. Writing a logic L into TCR_7 enables CTO.

5

Table 2 Counter/Timer Control Register Format

Control Register Bit	State	Bit Definition	State Definition
TCR_0	L H	Internal Reset	Timer Enabled Timer in Preset State
TCR_1	L H	Clock Source	Timer uses External Clock ($\overline{CTC}$) Timer uses $\phi 2$ System Clock
TCR_2	L H	÷8 Prescaler Enabler	Clock is not Prescaled Clock is Prescaled by ÷8 Counter
TCR_3 TCR_4 TCR_5	X X X	Operating Mode Selection	See *Table 3*
TCR_6	L H	Timer Interrupt Enable	$\overline{IRQ}$ Masked from Timer $\overline{IRQ}$ Enabled from Timer
TCR_7	L H	Timer Output Enable	Counter Output (CTO) Set LOW Counter Output Enabled

Table 3 Counter/Timer Control Register



TCR_3	TCR_4	TCR_5	Timer Operating Mode	Counter Initialization	Interrupt Flag Set
L	L	L	Continuous	$\overline{CTG}$ ↓ + W + R	T.O.
L	L	H	Cascaded Single Shot	$\overline{CTG}$ ↓ + R	T.O.
L	H	L	Continuous	$\overline{CTG}$ ↓ + R	T.O.
L	H	H	Normal Single Shot	$\overline{CTG}$ ↓ + R	T.O.
H	L	L	Frequency Comparison	$\overline{CTG}$ ↓ • I • (W + T.O.) + R	$\overline{CTG}$ ↓ Before T.O.
H	L	H		$\overline{CTG}$ ↓ • I + R	T.O. Before $\overline{CTG}$ ↓
H	H	L	Pulse Width Comparison	$\overline{CTG}$ ↓ • I + R	$\overline{CTG}$ ↓ Before T.O.
H	H	H			T.O. Before $\overline{CTG}$ ↓

Timer Operating Modes

The F6846 has been designed to operate effectively in a wide variety of applications. This is accomplished by using three bits of the Counter/Timer Control Register (TCR_3, TCR_4 and TCR_5) to define different operating modes of the timer, outlined in *Table 4*.

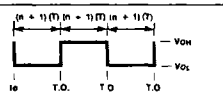
Table 4 Operating Modes

Control Register				
TCR_3	TCR_4	TCR_5	Timer Operating Mode	
L	*	L	Continuous	
L	L	H	Single-shot	Cascaded Single-shot
	H			Normal Single-shot
H	L	*	Frequency Comparison	
H	H	*	Pulse Width Comparison	

*Defines Additional Timer Functions

Continuous Operating Mode (TCR_3 = L, TCR_5 = L)
The timer may be programmed to operate in a continuous counting mode by writing L into bits 3 and 5 of the Counter/Timer Control Register. Assuming that the timer output is enabled (TCR_7 = H), a square wave will be generated at the Counter Timer Output CTO (see *Table 5*).

Table 5 Continuous Operating Modes (TCR_3 = L, TCR_7 = H, TCR_5 = L)

Control Register		Initialization/Output Waveforms	
TCR_2	TCR_4	Counter	Timer Output (2X)
L	L	Initialization $\overline{G}$ + W + R	
L	H	$\overline{G}$ + R	

$\overline{G}$ = Negative transition of $\overline{CTG}$ input
W = Write-timer-latches command
R = Timer Reset (TCR_0 = H or External $\overline{RESET}$ = L)
N = 16-bit number in Counter Latch
T = Period of Clock input to Counter
to = Counter initialization cycle
T.O. = Counter Time-out (all L condition)

Note
All time intervals shown above assume that the $\overline{CTG}$ and $\overline{CTC}$ signals are synchronized to system φ2 with the specified set-up and hold time requirements.

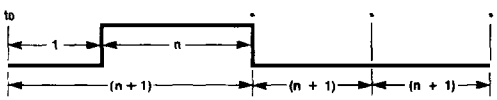
Either a Timer Reset (TCR_0 = H or External Reset = L) condition or internal recognition of a negative transition of the $\overline{CTG}$ input results in counter initialization. A write-timer-latches command can be selected as a counter initialization signal by clearing TCR_4.

The discussion of the continuous mode has assumed that the application requires an output signal. It should be noted that the timer operates in the same manner with the output disabled (TCR_7 = L). A read-timer-counter command is valid regardless of the state of TCR_7.

Single-shot Timer Mode
(TCR_3 = L, TCR_4 = H, TCR_5 = H)
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 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 reinitialization 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.

Normal Single-Shot Mode Output Waveform



H = Write an "H" into TCR_7
L = Write an "L" into TCR_7

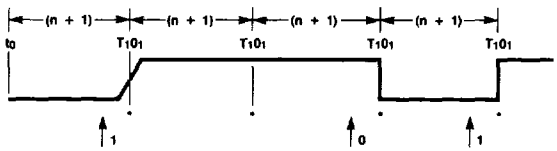
Note
All time intervals shown above assume the $\overline{CTG}$ and $\overline{CTC}$ signals are synchronized to system $\phi 2$ with the specified set-up and hold time requirements.

Cascaded Single-shot Mode
(TCR_3 = L, TCR_4 = L, TCR_5 = L)
This mode is identical to the single-shot mode with two exceptions. First, the output waveform does not return to LOW level and remain LOW after time-out. Instead, the output level remains at its initialized level until it is reprogrammed and changed by time-out. The output level may be changed at any time-out or may have any number of time-outs between changes.

The second difference is the method used to change the output level. Counter/Timer Control Register bit 7 (TCR_7) has a special function in this mode. The Counter Timer Output (CTO) is equal to TCR_7 clocked by time-out. At every time-out the content of TCR_7 is clocked to and held at the CTO. Thus, output pulses of length greater than one timer cycle can be generated by cascading timer cycles and counting time-outs with a software program.

An interrupt is generated at each time-out. To cascade timer cycles, the MPU would need an interrupt routine to: 1) count each time-out and determine when to change TCR_7; 2) write into TCR_7 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 Composite Status Register followed by the 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.

Cascaded Single-shot Mode Output Waveform



H = Write an "H" into TCR_7
L = Write an "L" into TCR_7

Note
All time intervals shown above assume the $\overline{CTG}$ and $\overline{CTC}$ signals are synchronized to system $\phi 2$ with the specified set-up and hold time requirements.

Time Interval Modes (TCR_3 = H)

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 $\overline{CTG}$ input transition. 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 6*.

5

Table 6 Timer Interval Modes (TCR_3 = H)

TCR_4	TCR_5	Application	Condition for Setting Individual Interrupt Flag
L	L	Frequency Comparison	Interrupt generated if $\overline{CTG}$ input period (1/F) is less than counter time-out (T.O.)
L	H	Frequency Comparison	Interrupt generated if $\overline{CTG}$ input period (1/F) is greater than counter time-out (T.O.)
H	L	Pulse Width Comparison	Interrupt generated if $\overline{CTG}$ input *down time* is less than counter time-out (T.O.)
H	H	Pulse Width Comparison	Interrupt generated if $\overline{CTG}$ input *down time* is greater than counter time-out (T.O.)

Frequency Comparison Mode (TCR_3 = H, TCR_4 = L)
The timer within the F6846 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 7* are satisfied. The counter decrements on each clock signal recognized during or after counter initialization until an interrupt is generated, a write-timer-latches command is issued, or a timer reset condition occurs. It can be seen from *Table 7* that an interrupt condition will be generated if TCR_5 = L and the period of the pulse (single pulse or measured separately repetitive pulses) at the $\overline{CTG}$ input is less than the counter time-out period. If TCR_5 = H, an interrupt is generated if the pulse period is greater than the time-out period.

Assume now with TCR_5 = H 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 CTG input cycle until the mode is changed, or a cycle is determined to be above the predetermined limit.

Pulse Width Comparison Mode (TCR_3 = H, TCR_4 = H)
This mode is similar to the frequency comparison mode except for the limiting factor being a positive, rather than a negative, transition of the $\overline{CTG}$ input. With TCR_5 = L, an individual interrupt flag will be generated if the L level pulse applied to the $\overline{CTG}$ input is less than the time period required for counter time-out. With TCR_5 = H, the interrupt is generated when the reverse condition is true.

As can be seen in *Table 8*, a positive transition of the $\overline{CTG}$ input disables the counter. With TCR_5 = L, it is therefore possible to directly obtain the width of any pulse causing an interrupt.

Table 7 Frequency Comparison Mode

TCR_3 = H, TCR_4 = L

Control Reg Bit 5 (TCR_5)	Counter Initialization	Counter Enable Flip-Flop Set (CE)	Counter Enable Flip-Flop Reset (CE)	Interrupt Flag Set (I)
L	$\overline{G}\downarrow \cdot \overline{I} \cdot (\overline{CE} + TO \cdot \overline{CE}) + R$	$\overline{G}\downarrow \overline{W} \cdot \overline{R} \cdot \overline{I}$	$W + R + I$	$\overline{G}\downarrow$ before TO
H	$\overline{G}\downarrow \cdot \overline{I} + R$	$\overline{G}\downarrow \overline{W} \cdot \overline{R} \cdot \overline{I}$	$W + R + I$	TO before $\overline{G}\downarrow$

$\overline{I}$ represents the interrupt for a given timer.

Table 8 Pulse Width Comparison Mode

TCR_3 = H, TCR_4 = H

Control Reg Bit 5 (TCR_5)	Counter Initialization	Counter Enable Flip-Flop Set (CE)	Counter Enable Flip-Flop Reset (CE)	Interrupt Flag Set (I)
L	$\overline{G}\downarrow \cdot \overline{I} + R$	$\overline{G}\downarrow \cdot \overline{W} \cdot \overline{R} \cdot \overline{I}$	$W + R + I + G$	$\overline{G}\downarrow$ before TO
H	$\overline{G}\downarrow \cdot \overline{I} + R$	$\overline{G}\downarrow\uparrow \overline{W} \cdot \overline{R} \cdot \overline{I}$	$W + R + I + G$	TO before $\overline{G}\uparrow$

Differences Between the F6840 and the F6846 Timers

1. Control Registers 1 and 3 are buried (access through Control Register 2 only) in the F6840 timer. In the F6846 all registers are directly accessible.

2. The F6840 has a dual 8-bit continuous mode for generating non-symmetrical waveforms. The F6846 has a cascaded one-shot mode which can accomplish the same function, but also allows the user to generate waveforms longer than one time-out.

3. Because of the different modes, there is a difference in the control registers between the F6840 and the F6846. (See *Table 9*).

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 F6846. 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. The sequence of assertion is not detected. Setting TCR_6 while CSR_0 is HIGH will cause CSR_7 to be set, for example.

The composite interrupt flag (CSR_7) is clear only if all enabled individual interrupt flags are clear. The conditions for clearing CSR_1 and CSR_2 are detailed in a later section. The timer interrupt flag (CSR_0) is cleared under the following conditions:

1. Timer Reset—Internal Reset bit (TCR_0 = H) or External Reset = L

2. Any Counter Initialization condition

3. A write-timer-latches command if time interval modes (TCR_3 = H) are being used.

4. A read-timer-counter command, provided this is preceded by a read Composite Status Register while CSR_0 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 (CSR_3-CSR_6) are unused. They default to a logic L when read.

5

Composite Status Register and Associated Logic

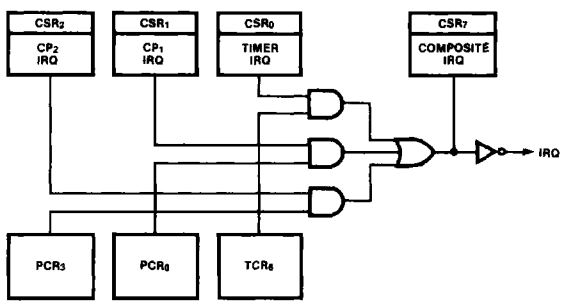
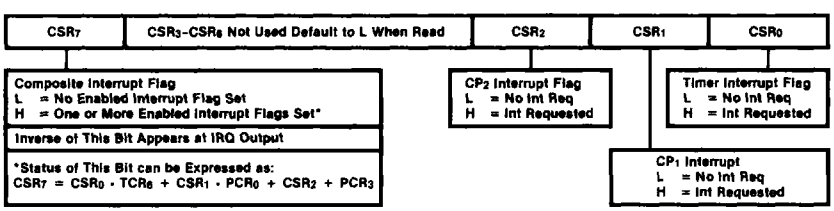


Table 9 F6840 and F6846 Control Register Comparison

Control Register Bit	F6840	F6846
2	16-bit or dual 8-bit mode control	÷8 prescale enable
7	Output enable (all modes)	Output next state (cascaded one-shot mode only), output enable all other modes
0	R1 internal reset R2 control register select R3 timer 3 clock control	Internal reset

Table 10 Composite Status Register Format



I/O Operation

Parallel Peripheral Port

The peripheral port of the F6846 contains eight Peripheral Data lines (P_0–P_7), two peripheral control lines (CP_1 and CP_2), a Peripheral Data Direction Register, a Peripheral Data Register, and a Peripheral Control Register. The port also directly affects two bits (CSR_1 and CSR_2) of the Composite Status Register.

The peripheral port is similar to the "B" side of a PIA (F6821) with the following exceptions:

1. All registers are directly accessible in the F6846. Data direction and peripheral data in the F6821 are located at the same address, with bit 2 of the control register used for register selection.

2. Peripheral Control Register bit 2 (PCR_2) of the F6846 is used to select an optional input latch function. This option is not available with F6821 PIAs.

3. Interrupt flags are located in the F6846 Composite Status Register rather than bits 6 and 7 of the control register as used in the F6821.

4. Interrupt flags are cleared in the F6821 by reading data from the Peripheral Data Register. F6846 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.

5. Bit 6 of the F6846 Peripheral Control Register is not used. Bit 7 (PCR_7) is an internal reset bit not available on the F6821.

6. The Peripheral Data lines (and CP_2) of the F6846 feature internal current limiting which allows them to directly drive the base of Darlington npn transistors.

Peripheral Data Direction Register

The MPU can write directly to this 8-bit register to configure the Peripheral Data lines as either inputs or outputs. A particular bit within the register (DDR_n) is used to control the corresponding Peripheral Data line (P_n). With DDR_n = L, P_n becomes an input; if DDR_n = H, P_n is an output. For example, writing Hex $OF into the Peripheral Data Direction Register results in P_0 through P_3 becoming outputs and P_4 through P_7 inputs. Hex $55 in the Peripheral Data Direction Register results in alternate output and inputs at the parallel port.

Peripheral Data Register

This 8-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 are 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 Peripheral 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 8-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 11.

Peripheral Port Reset (PCR_7)

Bit 7 of the Peripheral Control Register (PCR_7) may be used to initialize the peripheral section of the F6846. When this bit is set HIGH, the Peripheral Data Register, the Peripheral Data Direction Register, and the interrupt flags associated with the peripheral port (CSR_1 and CSR_2) are all cleared. Other bits in the Peripheral Control Register are not affected by PCR_7.

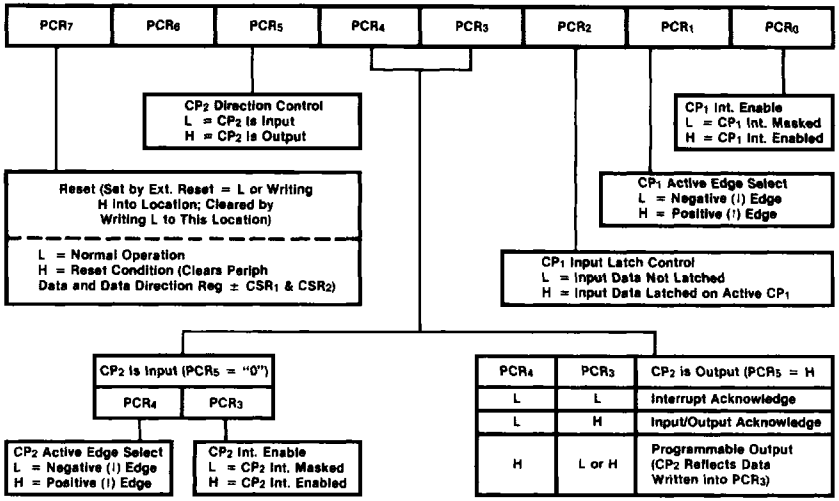
PCR_7 is set by either a logic L at the external $\overline{RESET}$ input or under program control by writing an H into the location. In any case, PCR_7 may be cleared only by writing an L into the location while $\overline{RESET}$ is HIGH. The bit must be cleared to activate the port.

Control of Peripheral Interrupt Line (CP_1)

CP_1 may be used as an interrupt request to the F6846, 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. These options are selected via Peripheral Control Register bits PCR_0, PCR_1 and PCR_2.

Peripheral Control Register bit 0 (PCR_0) is used to enable the interrupt transfer circuitry of the F6846. Regardless of the state of PCR_0, an active transition of CP_1 causes the Composite Status Register bit 1 (CSR_1) to be set. If PCR_0 = H, this interrupt will be reflected in the composite interrupt flag (CSR_7), and thus at the $\overline{IRQ}$ output. CSR_1 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

Table 11 Peripheral Control Register Format (Expanded)



read. The latter alternative is conditional; CSR_1 must have a logic H when the Composite Status Register was last read. This preculudes inadvertent clearing of interrupt flags generated between the time the Composite Status Register is read and the manipulation of peripheral data.

Peripheral Control Register bit 1 (PCR_1) is used to select the edge which activates CP_1. When PCR_1 = H, CP_1 is active on negative transitions (HIGH-to-LOW). LOW-to-HIGH transitions are sensed by CP_1 when PCR_1 = H.

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 HIGH into Peripheral Control Register bit 2 (PCR_2). In operation, the data at the pins designated by the Peripheral 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; 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 an L into PCR_2) has no effect on output data. It also does not affect interrupt function of CP_1.

Control of Peripheral Control Line (CP_2)

CP_2 may be used as an input by writing an L into PCR_5. In this configuration, CP_2 becomes a dual of CP_1 in regard to generation of interrupts. An active transition (as selected by PCR_4) causes bit 2 of the Composite Status Register to be set. PCR_3 is then used to select whether the CP_2 transition is to cause CSR_7 to be set and, thereby, cause $\overline{IRQ}$ to go LOW. CP_2 has no effect on the input latch function of the F6846.

Writing an H into PCR_5 causes CP_2 to function as an output. PCR_4 then determines whether CP_2 is to be used in a handshake or programmable output mode. With PCR_4 = H, CP_2 will merely reflect the data written into PCR_3. 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 Peripheral Data Direction Register.

The handshaking mode (PCR_5 = H, PCR_4 = L) allows CP_2 to perform one of two functions as selected by PCR_3. With PCR_3 = H, CP_2 will go LOW on the first Enable (system $\phi 2$) 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 (PCR_5 = H, PCR_4 = PCR_3 = L), CP_2 is set when CSR_2 is set by an active transition of CP_1. It is released (goes LOW) on the first positive transition of Enable after CSR_1 has been cleared via an MPU read or write to the Peripheral Data Register. (Note that the previously described conditions for clearing CSR_1 still apply.)

Restart Sequence

A typical restart sequence for the F6846 will include initialization of both the Peripheral Control and Peripheral Data Direction Registers of the parallel port. It is necessary to set up the Peripheral Control Register first, since PCR_7 = L is a condition for writing data into the Peripheral Data Direction Register. (A logic L at the external $\overline{RESET}$ input automatically sets PCR_7).

Absolute Maximum Ratings

Supply Voltage, V_{CC}	−0.3 V, +7.0 V
Input Voltage, V_{IN}	−0.3 V, +7.0 V
Operating Temperature, T_A	0 °C, +70 °C
Storage Temperature, T_{stg}	−55 °C, +150 °C
Thermal Resistance, ϕ_{JA}	70 °C/W

Stresses greater than those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Electrical Characteristics V_{CC} = 5.0 V ±5%, V_{SS} = 0, T_A = 0 to +70°C unless otherwise noted

Symbol	Characteristic	Min	Typ	Max	Unit	Condition
V_{IH}	Input HIGH Voltage, All Inputs	2.0			V	
V_{IL}	Input LOW Voltage, All Inputs	−0.3		0.8	V	
V_{OS}	Clock Input HIGH	V_{CC} −0.5		V_{CC} +0.5	V	
	Clock Input LOW	V_{SS} −0.5		V_{SS} +0.5		
V_{OH}	Output HIGH Voltage D_0-D_7 Other Outputs	 2.4 2.4			V	 I_{Load} = −205 μA I_{Load} = −200 μA
V_{OL}	Output LOW Voltage D_0-D_7 Other Outputs			 0.4 0.4	V	 I_{Load} = 1.6 mA I_{Load} = 3.2 mA
I_{IN}	Input Leakage Current CP_1, $\overline{CTG}$, $\overline{CTC}$, E A_0-A_{10} R/$\overline{W}$, $\overline{RESET}$, CS_0, CS_1		 1.0	 2.5	 μA	 V_{IN} = 0 to 5.25 V
I_{TSI}	3-State (OFF State) Input Current D_0-D_7, P_0-P_7, CP_2		2.0	10	μA	V_{IN} = 0.4 to 2.4 V
I_{LOH}	Output Leakage Current (OFF State) $\overline{IRQ}$			10	μA	V_{OH} = 2.4 V
I_{OL}	Output LOW Current (Sinking) D_0-D_7 Other Outputs	 1.6 3.2			mA	V_{OL} = 0.4 V
I_{OH}	Output HIGH Current (Sourcing) D_0-D_7 Other Outputs	 −205 −200			μA	V_{OH} = 2.4 V
	CP_2, P_0-P_7	1.0		−10	mA	V_O = 1.5 V for driving other than TTL
P_D	Power Dissipation			1000	mW	
C_{IN}	Input Capacitance E D_0-D_7 All Other Inputs			 200 12.5 7.5	pF	V_{IN} = 0, T_A = 25°C, f = 1.0 MHz
C_{OUT}	Output Capacitance P_0-P_7, CP_2, CTO $\overline{IRQ}$			 10 5.0	pF	V_{IN} = 0, T_A = 25°C, f = 1.0 MHz
f	Operating Frequency	0.1		1.0	MHz	
$t_{CYC}E$	Clock Cycle Time, E	1.0			μs	
t_{RL}	Clock $\overline{RESET}$ LOW Time	2.0			μs	
t_{IR}	Clock Interrupt Release Time			1.6	μs	*Figure 5*

5

Read/Write Timing *(Figures 1, 2)*

Symbol	Characteristic	Min	Typ	Max	Unit
PW_{EL}	Enable Pulse Width, LOW	430			ns
PW_{EH}	Enable Pulse Width, HIGH	430			ns
t_{AS}	Set-up Time (Address CS_0, CS_1, R/$\overline{W}$)	160			ns
t_{DDR}	Data Delay Time			320	ns
t_H	Data Hold Time	10			ns
t_{AH}	Address Hold Time	10			ns
t_{Er}, t_{Ef}	Rise and Fall Time			25	ns
t_{DSW}	Data Set-up Time	195			ns

Bus Timing Peripheral I/O Lines *(Figures 3, 4, 6, 7 and 11)*

Symbol	Characteristic	Min	Typ	Max	Unit
t_{PDSU}	Peripheral Data Set-up	200			ns
t_{Pr}, t_{Pf}	Rise and Fall Times CP_1, CP_2			1.0	μs
t_{CP2}	Delay Time E to CP_2 Fall			1.0	μs
t_{DC}	Delay Time I/O Data CP_2 Fall	20			ns
t_{RS1}	Delay Time E to CP_2 Rise			1.0	μs
t_{RS2}	Delay Time CP_1 to CP_2 Rise			2.0	μs
t_{PDW}	Peripheral Data Delay			1.0	μs
t_{PSU}	Peripheral Data Set-up Time for Latch	100			ns
t_{PDH}	Peripheral Data Hold Time for Latch	15			ns

Timer-Counter Lines *(Figures 8, 9, and 10)*

Symbol	Characteristic	Min	Typ	Max	Unit
t_{Cr}, t_{Cf}	Input Rise and Fall Time, $\overline{CTC}$ and $\overline{CTG}$			100	ns
t_{PWH}	Input Pulse Width, HIGH (Asynchronous Mode)	t_{cyc} + 250			ns
t_{PWL}	Input Pulse Width, LOW (Asynchronous Mode)	t_{cyc} + 250			ns
t_{su}	Input Set-up Time (Synchronous Mode)	200			ns
t_{hd}	Input Hold Time (Synchronous Mode)	50			ns
t_{CTO}	Output Delay			1.0	μs

Fig. 1 Bus Read Timing (Read Information from F6848)

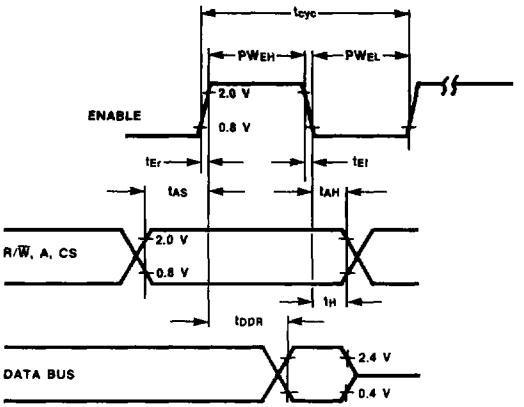


Fig. 2 Bus Write Timing (Write Information from MPU)

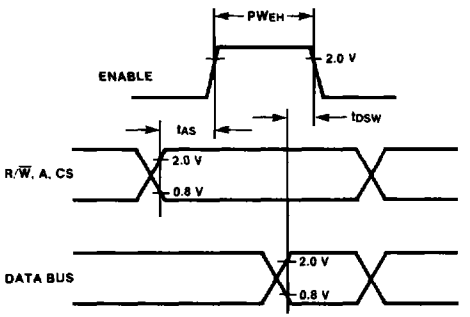


Fig. 3 Peripheral Port Latch Set-up and Hold Time

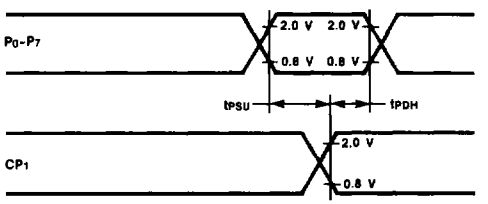
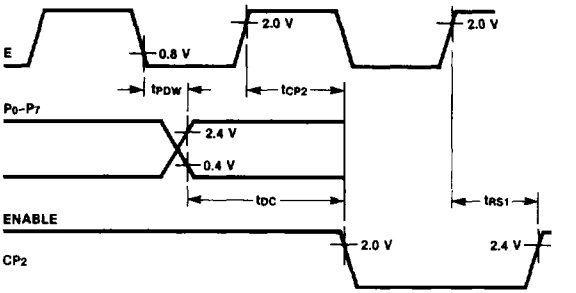


Fig. 4 Peripheral Data and CP_2 Delay (Control Mode PCR_5 = H, PCR_4 = L, PCR_3 = H)



Note
CP_2 goes LOW as the result of positive transition of the second ϕ_2 pulse

Fig. 5 IRQ Release Time

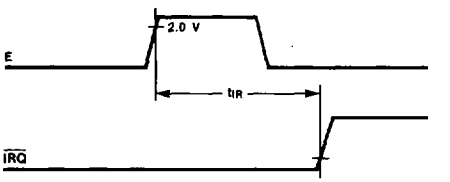


Fig. 6 Peripheral Port Set-up Time

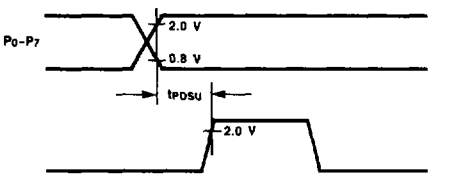


Fig. 7 CP_2 Delay Time (PCR_5 = H, PCR_4 = L, PCR_3 = L)

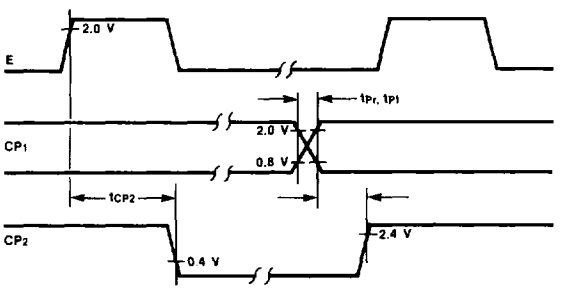


Fig. 8 Input Pulse Widths

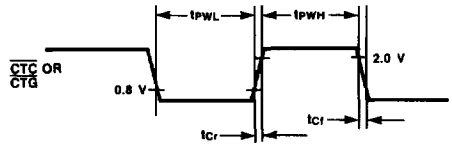
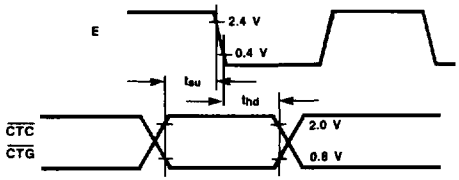


Fig. 9 Input Set-up and Hold Times



Note
This mode is valid only for synchronous operation.

Fig. 10 Output Delay

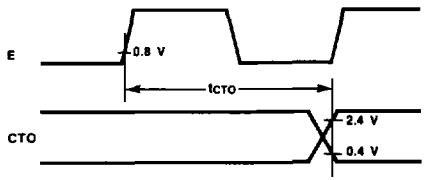
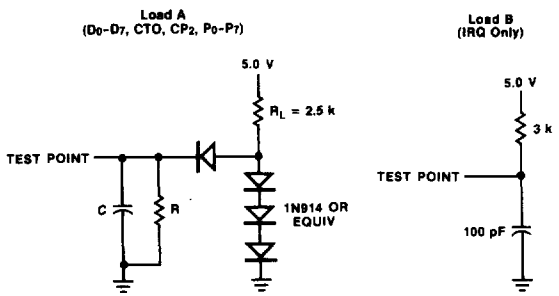


Fig. 11 Bus Timing Test Loads



C = 130 pF for D_0-D_7
= 30 pF for CTO, CP_2, P_0-P_7
R = 11.7 kΩ for D_0-D_7
= 24 kΩ for CTO, CP_2, P_0-P_7

Custom ROM Programming Information
The customer's unique program code pattern may be submitted to Fairchild in several methods. The most convenient and readily verifiable is in the form of 2708, 2716 or 2732 EPROMs. Program code patterns may also be submitted on Fairchild Formulator MKIII floppy disks or on HP cassette tape in Formulator or MIKBUG* format.

Customer Company Name _____
Customer Contact Name _____
Customer Part No. _____
Address _____
Phone No. _____
Fairchild Part No. _____

Fairchild Use Only
SL No. _____
Bid Control No. _____
Field Sales Engineer _____
Date Sent _____

Customer Input Media
- ☐ 2708 EPROM
- ☐ 2716 EPROM
- ☐ 2732 EPROM
- ☐ HP Cassette
  - ☐ Formulator Format

Request for Return Media
- ☐ Listing
- ☐ EPROM (include blank EPROMs)

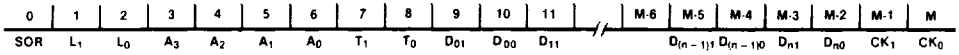
Rom Select*
CS_0 _____
CS_1 _____

I/O Select*
CS_0 _____
CS_1 _____
A_6 _____ (may be unused

Location Select (Select 1 only)
- ☐ A_7 HIGH
- ☐ A_8 HIGH
- ☐ A_9 HIGH
- ☐ A_{10} HIGH
- ☐ Not used

*ROM and I/O Selects must be different.
H = HIGH to Select
L = LOW to Select

Formulator Format



SOR	Start of record defined to be a colon(:)	T_1 T_0	Type field.
L_1 L_0	Length field defined to be the number of packed data bytes per record. Each record is (2 • L) + 11 characters in length inclusive of start of record Length 0 implies end of relocatable module.	D_{01} D_{00}...$D_{(n)1}$ $D_{(n)0}$	Data field.
		CK_1 CK_0	Checksum field defined to be negative modulo 256 summation of all bytes since start of record. A summation of all characters in a record, including the checksum, will result in zero.
A_3 A_2 A_1 A_0	Address Field	All characters other than SOR are ASCII hexadecimal (0-9, A-F).	

5

Ordering Information

Speed	Order Code	Temperature Range
1.0 MHz	F6846P, S	0 °C to + 70 °C

P = Plastic Package; S = Ceramic Package