FAIRCHILD
A Schlumberger Company

F38E70
Single-Chip Microcomputer

Microprocessor Product

Description

The Fairchild single-chip microcomputer series offers a variety of circuits for the high-volume, cost-sensitive markets. The F38E70 is a complete 8-bit microcomputer on a single MOS integrated circuit. The F38E70 is functionally identical to the F3870, except the F38E70 has 2K bytes of EPROM in place of 2K bytes of ROM. The F38E70 can execute the F8 instruction set of more than 70 commands. The device features 2048 bytes of EPROM, 64 bytes of scratchpad RAM, a programmable binary timer, 32 bits of I/O, and a single +5 V power supply requirement.

Utilizing Fairchild's double-ion-implant, N-channel technology and advanced circuit design techniques, the single-chip F38E70 offers maximum cost-effectiveness in many low-to-medium volume systems. When production volume requires large quantities, the transition to the mask-programmed F3870 is very straightforward, with no circuit design changes.

- **Single-Chip Microcomputer**
- **Software-Compatible with F8 Family**
- **2048-Byte EPROM (F38E70-2)**
- **64-Byte Scratchpad RAM**
- **32-Bit (4-Port) TTL-Compatible I/O**
- **Programmable Binary Timer**
  - **Interval Timer Mode**
  - **Pulse Width Measurement Mode**
  - **Event Counter Mode**
- **External Interrupt**
- **Crystal, LC, RC, External, or Internal Time Base**
- **Low Power (375 mW Typical)**
- **Single +5 V ±10% Power Supply**
- **Simple EPROM Programming**

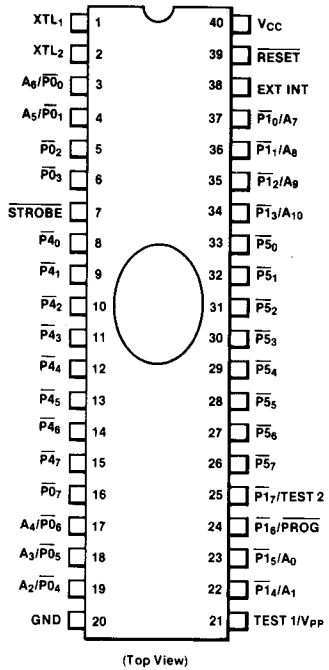
Pin Names

$\overline{P0}_0$-$\overline{P0}_7$	Bidirectional I/O Port 0/Address*
$\overline{P1}_0$-$\overline{P1}_7$	Bidirectional I/O Port 1/Address*
$\overline{P4}_0$-$\overline{P4}_7$	Bidirectional I/O Port 4/Data Out*
$\overline{P5}_0$-$\overline{P5}_7$	Bidirectional I/O Port 5/Data In*
$\overline{STROBE}$	Ready Strobe Output
EXT INT	External Interrupt Input
$\overline{RESET}$	External Reset Input
TEST 1/V_{PP}	Test Line/PROG Voltage Input**
XTL_1, XTL_2	Time Base Input
V_{CC}, GND	Power Supply Lines

*As shown in the connection diagram, some port 0 and port 1 pins are address inputs for programming the F38E70 EPROM section. Ports 4 and 5, Data Out and In, refer to the programming and test modes.

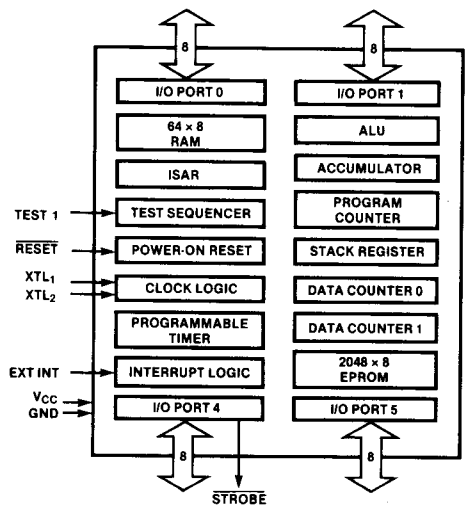
**Caution: applying +25 V to the V_{PP} pin without the presence of V_{CC} will damage the device.

Connection Diagram
40-Pin DIP

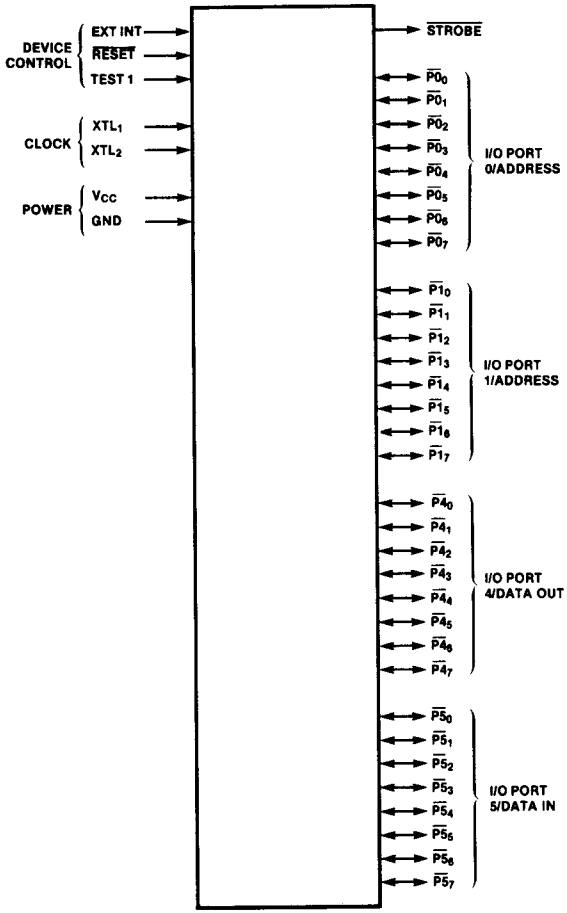


(Top View)

F38E70 Architecture



Signal Functions



Device Organization

This section describes the basic functional elements of the F38E70 as shown in *Figure 1*.

Main Control Logic
The instruction register (IR) receives the operation code (OP code) of the instruction to be executed from the program EPROM via the data bus. During all OP code fetches, eight bits are latched into the IR. Some instructions are completely specified by the upper four bits of the OP code. In those instructions, the lower four bits are an immediate register address or an immediate 4-bit operand. Once latched into the IR, the main control logic decodes the instruction and provides the necessary control gating signals to all circuit elements.

EPROM Address Registers
There are four 11-bit registers associated with the 2K x 8 EPROM. These are the program counter (P0), the stack register (P), the data counter (DC), and the auxiliary data counter (DC1). The program counter is used to address instructions or immediate operands. The stack register is used to save the contents of P0 during an interrupt or subroutine cell. Thus, P contains the return address at which processing is to resume upon completion of the subroutine or the interrupt routine.

The data counter (DC) is used to address data tables. This register is auto-incrementing. Of the two data counters, only DC can access the EPROM. However, the XDC instruction allows DC and DC1 to be exchanged.

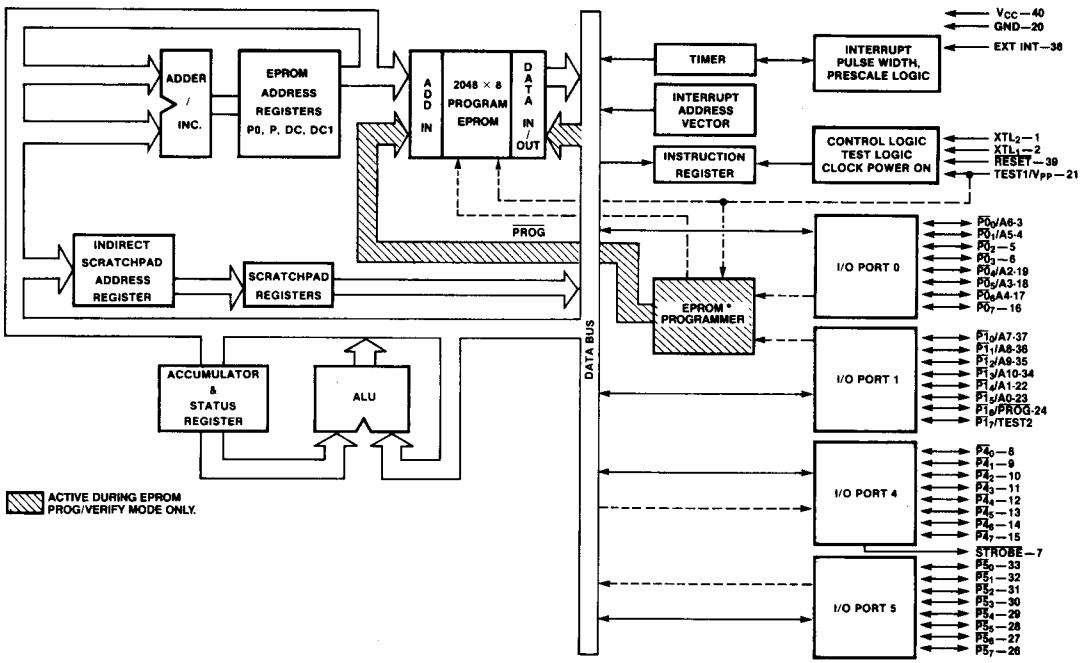
Associated with the address registers is an 11-bit adder/incrementer. This logic element is used to increment P0 or DC when required, and is also used to add displacements to P0 on relative branches or to add the accumulator contents to DC1 with the ADC (add data counter) instruction.

2048 x 8 EPROM
The microcomputer program and data constants are stored in the program EPROM. When an EPROM access is required, the appropriate address register (P0 or DC) is gated onto the EPROM address bus and the EPROM output is gated onto the main data bus. The first byte in the EPROM is location zero.

Scratchpad and ISAR
The scratchpad provides 64 8-bit registers that may be used as general-purpose read/write data memory. The indirect scratchpad address register (ISAR) is a 6-bit register used to address the 64 registers. All 64 registers may be accessed using ISAR. In addition, the lower order 12 registers may also be directly addressed.

The ISAR can be visualized as holding two octal digits. This division of ISAR is important, since a number of instructions increment or decrement only the least significant three bits of ISAR when referencing scratchpad bytes via ISAR. This makes it easy to reference a buffer consisting of up to eight contiguous scratchpad bytes. For example, when the low-order octal digit is incremented or decremented, ISAR is incremented from 27_8 to 20_8 or is decremented from 20_8 to 27_8. This feature of the ISAR is very useful in many

Fig. 1 Block Diagram



Pin Functions

Pin Name	Type	Description
$\overline{P0}_0$-$\overline{P0}_7$ $\overline{P1}_0$-$\overline{P1}_7$ $\overline{P4}_0$-$\overline{P4}_7$ $\overline{P5}_0$-$\overline{P5}_7$	Input/Output	Thirty-two lines that can be individually used as either TTL-compatible inputs or as latched outputs. For EPROM programming, 11 lines of ports 0 and 1 are used as address inputs and one line of port 1 is a program control. Port 5 is EPROM data input, and port 4 is EPROM output for verification.
$\overline{STROBE}$	Output	This pin, which is normally HIGH, provides a single LOW pulse after valid data is present on the $\overline{P4}_0$-$\overline{P4}_7$ pins during an output instruction.
$\overline{RESET}$	Input	$\overline{RESET}$ may be used to externally reset the F38E70. When pulled LOW, the F38E70 resets. When then allowed to go HIGH, the F38E70 begins program execution at the program location H '0000'. RESET is held LOW during EPROM programming.
EXT INT	Input	The external interrupt input. Its active state is software-programmable. This input is also used in conjunction with the timer for pulse width measurement and event counting.
XTL_1, XTL_2	Input	The time base inputs to which a crystal (1 to 4 MHz), LC network, RC network, or an external single-phase clock may be connected. If timing is not critical, the F38E70 operates from its internal oscillator with no external components.
TEST 1/V_{PP}	Input	An input used only in testing and programming the F38E70. For normal circuit functionality, this pin is left unconnected or may be grounded. For EPROM programming, the test pin is connected to the programming voltage (typically 23 V).
$P1_7$/TEST 2	Input	I/O during normal operation; must be HIGH when in verify mode.
V_{CC}	Power	V_{CC} is the power supply input (+5 V ± 10%).

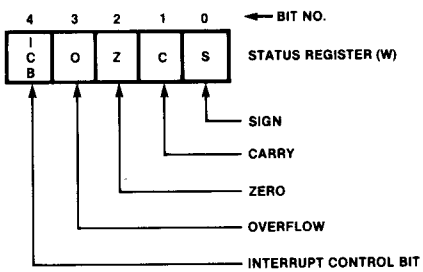
program sequences. All six bits of ISAR may be loaded at one time, or either half may be loaded independently.

Scratchpad registers 9 through 15 (decimal) are given mnemonic names (J, H, K, and Q) because of special linkages between these registers and other registers, such as a stack register. These special linkages facilitate the implementation of multi-level interrupts and subroutine nesting. For example, the instruction LR K, P stores the lower eight bits of the stack register into register 13 (K lower, or KL) and stores the upper three bits of P into the three least significant bits of register 12 (K upper, or KU).

Arithmetic and Logic Unit (ALU)
After receiving commands from the main control logic, the ALU performs the required arithmetic or logic operations (using the data presented on the two input busses) and provides the result on the result bus. The arithmetic operations that can be performed in the ALU are binary add, decimal adjust, add with carry, decrement, and increment. The logic operations that can be performed are AND, OR, exclusive-OR, ones complement, shift right, and shift left. Besides providing the result on the result bus, the ALU also provides four signals representing the status of the result. These signals, stored in the status register (W), represent the CARRY, OVERFLOW, SIGN, and ZERO condition of the result of the operation.

Accumulator
The accumulator (ACC) is the principal register for data manipulation within the F38E70. The ACC serves as one input to the ALU for arithmetic or logical operations. The results of ALU operations are stored back into the ACC.

Status Register
The status register (W) holds five status flags, as follows:



Summary of Status Bits

$$\text{OVERFLOW} = \text{CARRY}_7 \oplus \text{CARRY}_6$$

$$\text{ZERO} = \overline{\text{ALU}}_7 \wedge \overline{\text{ALU}}_6 \wedge \overline{\text{ALU}}_5 \wedge \overline{\text{ALU}}_4 \wedge \overline{\text{ALU}}_3 \wedge \overline{\text{ALU}}_2 \wedge \overline{\text{ALU}}_1 \wedge \overline{\text{ALU}}_0$$

$$\text{CARRY} = \text{CARRY}_7$$

$$\text{SIGN} = \overline{\text{ALU}}_7$$

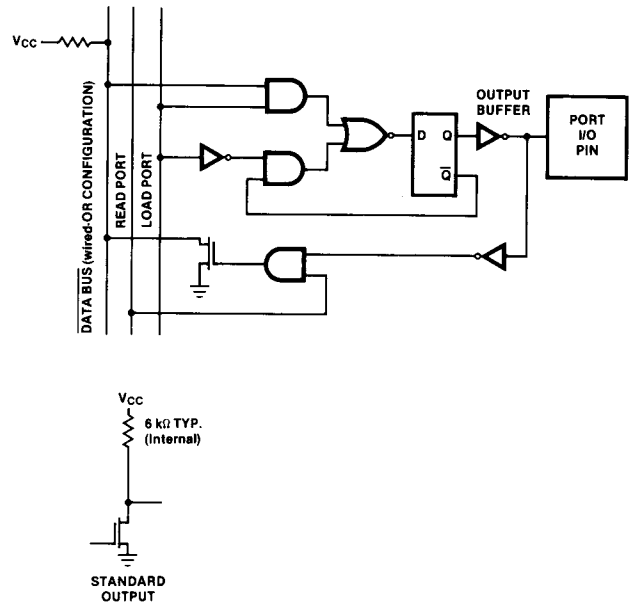
Interrupt Control Bit—The ICB may be used to allow or disallow interrupts in the F38E70. This bit is not the same as the two interrupt enable bits in the interrupt control port (ICP). If the ICB is set and the F38E70 interrupt logic communicates an interrupt request to the CPU section, the interrupt is acknowledged and processed upon completion of the first non-privileged instruction. If the ICB is cleared, an interrupt request is not acknowledged or processed until the ICB is set again.

I/O Ports
The F38E70 provides four complete bidirectional input/output ports: these are ports 0, 1, 4, and 5. An output instruction (OUT or OUTS) causes the contents of the ACC to be latched into the addressed port. An input instruction (IN or INS) transfers the contents of the port to the ACC (port 6 is an exception, which is described later). The I/O buffers on the F38E70 are logically inverted. The schematic of an I/O port is shown in *Figure 2*.

An output ready strobe is associated with port 4. This flag may be used to signal a peripheral device that the F38E70 has just completed an output of new data to port 4. The strobe provides a single low pulse shortly after the output operation is completed, so either edge may be used to signal the peripheral. The $\overline{\text{STROBE}}$ signal may also be used to request new input information from a peripheral simply by doing a dummy output of H '00' to port 4 after completing the input operation.

Timer and Interrupt Control Port
The timer is an 8-bit binary down counter that is software-programmable to operate in one of three modes: the interval timer mode, the pulse width measurement mode, or the event counter mode (the timer characteristics are described in *Table 1*). As shown in *Figure 3*, associated with the timer are an 8-bit register called the interrupt control port, a programmable prescaler, and an 8-bit modulo-N register; *Figure 4* illustrates the timer/interrupt function.

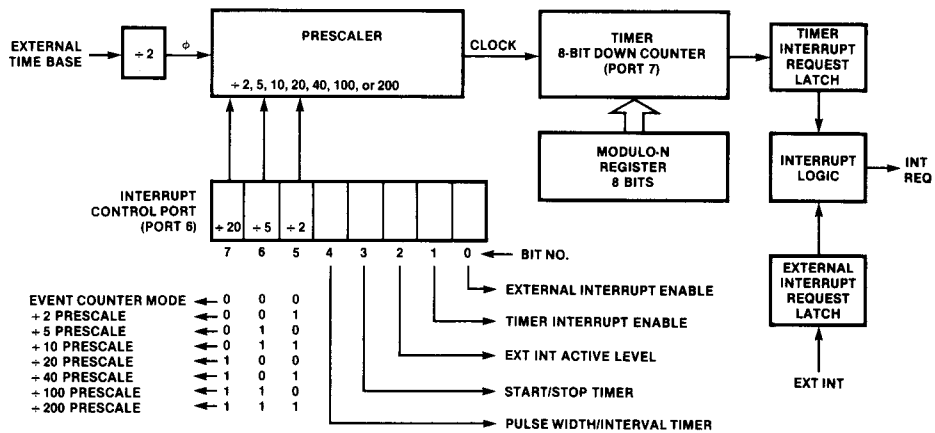
Fig. 2 I/O Port Diagram



All ports are standard output type only.

The $\overline{\text{STROBE}}$ output is always configured similar to a standard output, except that it is capable of driving three TTL loads.

Fig. 3 Timer and Interrupt Control Port Block Diagram



See *Figure 4* for a more detailed functional diagram.

Table 1 Timer Characteristics

Definitions	
Error = indicated time value – actual time value	
tpsc = tφ × prescale value	
Interval Timer Mode	
Single interval error, free-running (note 3)	± 6tφ
Cumulative interval error, free-running (note 3)	0
Error between two timer reads (note 2)	± (tpsc + tφ)
Start timer to stop timer error (notes 1, 4)	+ tφ to – (tpsc + tφ)
Start timer to read timer error (notes 1, 2)	– 5tφ to – (tpsc + 7tφ)
Start timer to interrupt request error (notes 1, 3)	– 2tφ to – 8tφ
Load timer to stop timer error (note 1)	+ tφ to – (tpsc + 2tφ)
Load timer to read timer error (notes 1, 2)	– 5tφ to – (tpsc + 8tφ)
Load timer to interrupt request error (notes 1, 3)	– 2tφ to – 9tφ
Pulse Width Measurement Mode	
Measurement accuracy (note 4)	+ tφ to – (tpsc + 2tφ)
Minimum pulse width of EXT INT pin	2tφ
Event Counter Mode	
Minimum active time of EXT INT pin	2tφ
Minimum inactive time of EXT INT pin	2tφ

Notes
1. All times that entail loading, starting, or stopping the timer are referenced from the end of the last machine cycle of the OUT or OUTS instruction.
2. All times that entail reading the timer are referenced from the end of the last machine cycle of the IN or INS instruction.
3. All times that entail the generation of an interrupt request are referenced from the start of the machine cycle in which the appropriate interrupt request latch is set. Additional time may elapse if the interrupt request occurs during a privileged or multi-cycle instruction.
4. Error may be cumulative if operation is repetitively performed.

The desired timer mode, prescale value, starting and stopping the timer, active level of EXT INT pin, and local enabling or disabling of interrupts are selected by outputting the proper bit configuration from the accumulator to the interrupt control port (port 6) with an OUT or OUTS instruction. Bits within the interrupt control port are defined as follows:

Interrupt Control Port (Port 6)
Bit 0 — External Interrupt Enable
Bit 1 — Timer Interrupt Enable
Bit 2 — EXT INT Active Level
Bit 3 — Start/Stop Timer
Bit 4 — Pulse Width/Interval Timer
Bit 5 — ÷ 2 Timer Prescale Values
Bit 6 — ÷ 5 Timer Prescale Values
Bit 7 — ÷ 20 Timer Prescale Values

A special situation exists when reading the interrupt control port (with an IN or INS instruction). The accumulator is **not** loaded with the content of the ICP; instead, accumulator bits 0 through 6 are loaded with 0s, while bit 7 is loaded with the logic level being applied to the EXT INT pin, thus allowing the status of EXT INT to be determined without the necessity of servicing an external interrupt request. This capability is useful in establishing a high-speed polled handshake procedure or for using EXT INT as an extra input pin if external interrupts are not required and the timer is used only in the interval timer mode.

The rate at which the timer is clocked in the interval timer mode is determined by the frequency of an internal φ clock and by the division value selected for the prescaler. (The internal φ clock operates at one-half the external time base frequency.) If ICP bit 5 is set and bits 6 and 7 are cleared, the prescaler divides φ by two. Likewise, if bit 6 or 7 is individually set, the prescaler divides φ by 5 or 20, respectively. Combinations of bits 5, 6, and 7 may also be selected. For example, if bits 5 and 7 are set while 6 is cleared, the prescaler divides by 40. Thus, possible prescaler values are ÷ 2, ÷ 5, ÷ 10, ÷ 20, ÷ 40, ÷ 100, and ÷ 200.

Any of three conditions causes the prescaler to be reset: when the timer is stopped by clearing the ICP bit 3, on execution of an output instruction to port 7 (the timer is assigned port address 7), or on the trailing edge transition of the EXT INT pin when in the pulse width measurement mode. These last two conditions are explained in more detail below.

An OUT or OUTS instruction to port 7 loads the content of the accumulator to both the timer and the 8-bit modulo-N register, resets the prescaler, and clears any previously stored timer interrupt request. As previously noted, the timer is an 8-bit down counter that is clocked by the prescaler in the interval timer mode and in the pulse width measurement mode. The prescaler is not used in the event counter mode. The modulo-N register is a buffer whose function is to save the value that was most recently output to port 7. The modulo-N register is used in all three timer modes.

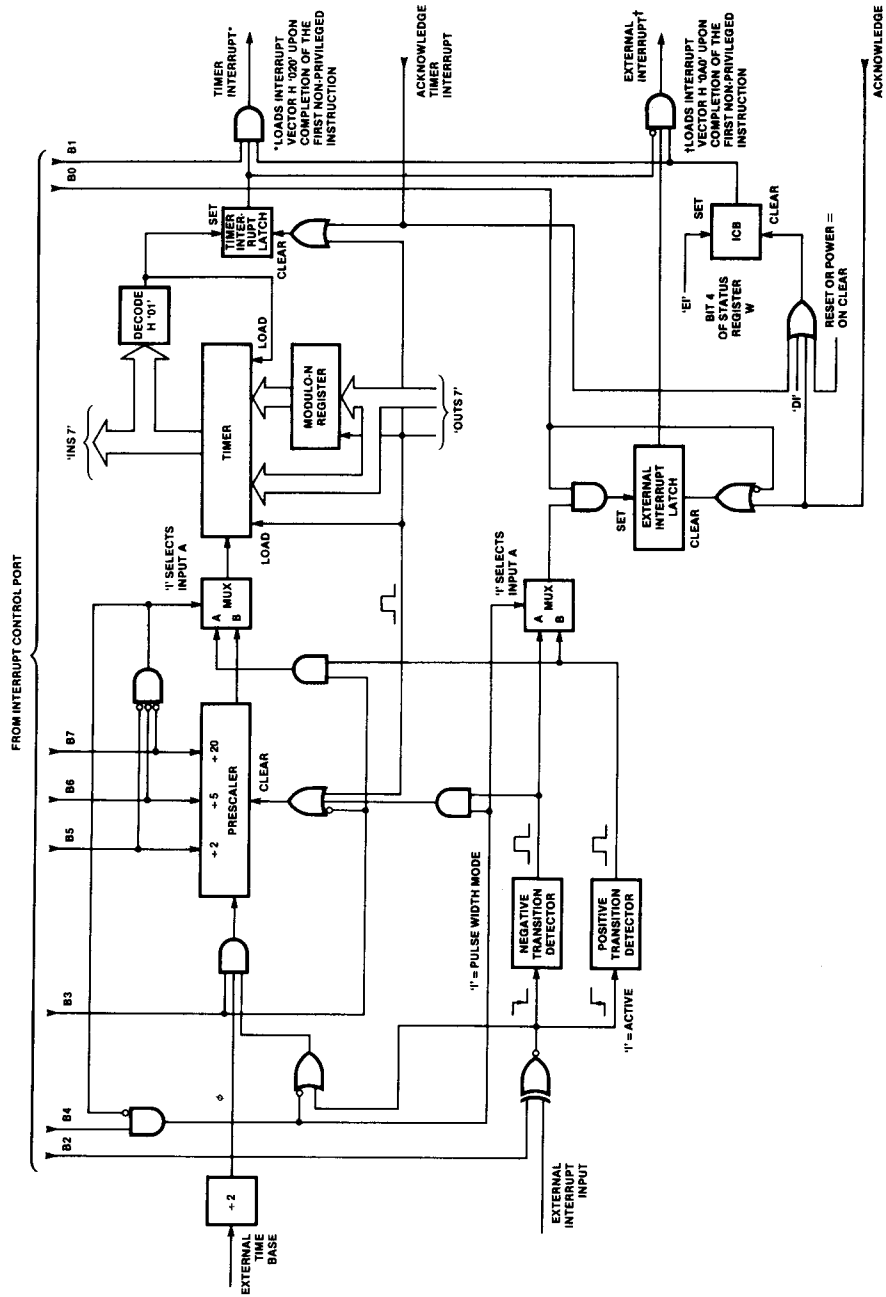


Fig. 4 Timer/Interrupt Functional Diagram

4

Interval Timer Mode—When ICP bit 4 is cleared (logic 0) and at least one prescale bit is set, the timer operates in the interval timer mode. When bit 3 of the ICP is set, the timer starts counting down from the modulo-N value. After counting down to H '01', the timer returns to the modulo-N value at the next count. On the transition from H '01' to H 'N', the timer sets a timer interrupt request latch. Note that the interrupt request latch is set by the transition of H 'N' in the timer, thus allowing a full 256 counts if the modulo-N register is preset to H '00'. If bit 1 of the ICP is set, the interrupt request is passed on to the CPU section of the F38E70. However, if bit 1 of the ICP is a logic 0, the interrupt request is not passed on to the CPU section but the interrupt request latch remains set. If ICP bit 1 is subsequently set, the interrupt request is then passed on to the CPU section. (Recall from the discussion of the status register interrupt control bit that the interrupt request is acknowledged by the CPU section only if ICB is set.) Only two events can reset the timer interrupt request latch: when the timer interrupt request is acknowledged by the CPU section, or when a new load of the modulo-N register is performed.

Consider an example in which the modulo-N register is loaded with H '64' (decimal 100). The timer interrupt request latch is set at the 100th count following the timer start, and the timer interrupt request latch is repeatedly set on precise 100-count intervals. If the prescaler is set at $\div 40$, the timer interrupt request latch is set every 4000 ϕ clock periods. For a 2 MHz ϕ clock (4 MHz time base frequency), this produces 2 ms intervals.

The range of possible intervals is from 2 to 51,200 ϕ clock periods (1 μs to 25.6 ms for a 2 MHz ϕ clock). However, approximately 50 ϕ periods is a practical minimum because the time between setting the interrupt request latch and the execution of the first instruction of the interrupt service routine is at least 29 ϕ periods (the response time is dependent upon how many privileged instructions are encountered when the request occurs). To establish time intervals greater than 51,200 ϕ clock periods is simply a matter of using the timer interrupt service routine to count the number of interrupts, saving the result in one or more of the scratchpad registers until the desired interval is achieved. With this technique, virtually any time interval, or several time intervals, may be generated.

The timer may be read at any time and in any mode using an input instruction (IN 7 or INS 7) and may take place "on-the-fly" without interfering with normal timer operation. Also, the timer may be stopped at any time by clearing bit 3 of the ICP. The timer holds its current contents indefinitely and resumes counting when bit 3 is again set. Recall, however, that the prescaler is reset whenever the timer is stopped; thus, a series of starting and stopping results in a cumulative truncation error.

A summary of other timer errors is given in the timing section. For a free-running timer in the interval timer mode, the time interval between any two interrupt requests may be in error by ± 6 ϕ clock periods, although the cumulative error over many intervals is zero. The prescaler and timer generate precise intervals for setting the timer interrupt request latch, but the time-out may occur at any time within a machine cycle. (There are two types of machine cycles: short cycles, which consist of 4 ϕ clock periods, and long cycles, which consist of 6 ϕ clock periods.) In the multi-chip F8 family, there is a signal called the write clock that corresponds to a machine cycle. Interrupt requests are synchronized with the internal write clock, thus giving rise to the possible ± 6 ϕ error. Additional errors may arise due to the interrupt request occurring while a privileged instruction or multi-cycle instruction is being executed. Nevertheless, for most applications all of the above errors are negligible, especially if the desired time interval is greater than 1 ms.

Pulse Width Measurement Mode—When ICP bit 4 is set (logic 1) and at least one prescale bit is set, the timer operates in the pulse width measurement mode. This mode is used for accurately measuring the duration of a pulse applied to the EXT INT pin. The timer is stopped and the prescaler is reset whenever EXT INT is at its inactive level. The active level of EXT INT is defined by ICP bit 2: if cleared, EXT INT is active LOW; if set, EXT INT is active high. If ICP bit 3 is set, the prescaler and timer start counting when EXT INT transitions to the active level. When EXT INT returns to the inactive level, the timer stops, the prescaler resets, and, if ICP bit 0 is set, an external interrupt request latch is set. (Unlike timer interrupts, external interrupts are not latched if the ICP interrupt enable bit is not set.)

As in the interval timer mode, the timer may be read at any time and may be stopped at any time by clearing ICP bit 3; the prescaler and ICP bit 1 function as previously described, and the timer still functions as an 8-bit binary down counter, with the timer interrupt request latch being set on the timer transition from H '01' to H 'N'. Note that the EXT INT pin has nothing to do with loading the timer; its action is that of automatically starting and stopping the timer and of generating external interrupts. Pulse widths longer than the prescale value times the modulo-N value are easily measured by using the timer interrupt service routine to store the number of timer interrupts in one or more scratchpad registers.

As for accuracy, the actual pulse duration is typically slightly longer than the measured value because the status of the prescaler is not readable and is reset when the timer is stopped. Thus, for maximum accuracy, it is advisable to use a small division setting for the prescaler.

Event Counter Mode—When ICP bit 4 is cleared and all prescale bits (ICP bits 5, 6, and 7) are cleared, the timer operates in the event counter mode. This mode is used for counting pulses applied to the EXT INT pin. If ICP bit 3 is set, the timer decrements on each transition from the inactive level to the active level of the EXT INT pin. The prescaler is not used in this mode, but, as in the other two timer modes, the timer may be read at any time and may be stopped at any time by clearing ICP bit 3; ICP bit 1 functions as previously described, and the timer interrupt request latch is set on the timer transition from H '01' to H 'N'.

Normally, ICP bit 0 should be kept cleared in the event counter mode; otherwise, external interrupts are generated on the transition from the inactive level to the active level of the EXT INT pin.

For the event counter mode, the minimum pulse width required on EXT INT is 2 ϕ clock periods and the minimum inactive time is 2 ϕ clock periods; therefore, the maximum repetition rate is 500 Hz.

External Interrupts
When the timer is in the interval timer mode, the EXT INT pin is available for non-timer-related interrupts. If ICP bit 0 is set, an external interrupt request latch is set when there is a transition from the inactive level to the active level of EXT INT. (EXT INT is an edge-triggered input.) The interrupt request is latched either until acknowleged by the CPU section or until ICP bit 0 is cleared (unlike timer interrupt requests, which remain latched even when ICP bit 1 is cleared). External interrupts are handled in the same fashion when the timer is in the pulse width measurement mode or in the event counter mode, except that only in the pulse width measurement mode is the external interrupt request latch set on the trailing edge of EXT INT, that is, on the transition from the active level to the inactive level.

Interrupt Handling
When either a timer or an external interrupt request is communicated to the CPU section of the F38E70, it is acknowledged and processed at the completion of the first non-privileged instruction if the interrupt control bit of the status register is set. If the interrupt control bit is not set, the interrupt request continues until either the interrupt control bit is set and the CPU section acknowledges the interrupt or the interrupt request is cleared as previously described.

If there are both a timer interrupt request and an external interrupt request when the CPU section starts to process the requests, the timer interrupt is handled first.

When an interrupt is allowed, the CPU section requests that the interrupting element pass its interrupt vector address to the program counter via the data bus. The vector address for a timer interrupt is H '020'. The vector address for external interrupts is H '0A0'. After the vector address is passed to the program counter, the CPU section sends an acknowledge signal to the appropriate interrupt request latch, which clears that latch. The execution of the interrupt service routine then commences. The return address of the original program is automatically saved in the stack register, P.

Power-on Clear
When power is applied to the F38E70, the program counter and the ICB bit of the status register are cleared. Ports 4, 5, 6, and 7 are loaded with H '00' (thus, the I/O pins for ports 4 and 5 are at V_{OH}). The contents of other registers and ports are undefined. The first program instruction is then fetched from EPROM location H '000'.

External Reset
When $\overline{\text{RESET}}$ is taken LOW, the content of the program counter is pushed to the stack register, and the program counter and the ICB bit of the status register are then cleared. The original stack register content is lost. As with power-on clear, ports 4, 5, 6, and 7 are loaded with H '00'. The contents of all other registers and ports are unchanged. When $\overline{\text{RESET}}$ is taken HIGH, the first program instruction is fetched from EPROM location H '000'.

Test Logic
Special test logic is implemented to allow access to the internal main data bus for test purposes.

In normal operation, the TEST pin is unconnected or is connected to GND. When TEST is placed at a TTL level (2.0 V to 2.6 V), port 4 becomes an output of the internal data bus and port 5 becomes a wired-OR input to the internal data bus. The data appearing on the port 4 pins is logically true, whereas input data forced on port 5 must be logically false. When TEST is placed at a HIGH level (6.0 V to 7.0 V), the ports act as above and, additionally, the 2K x 8 program ROM is prevented from driving the data bus. In this mode, operands and

instructions may be forced externally through port 5 instead of being accessed from the program ROM. When TEST is in either the TTL state or the HIGH state, STROBE ceases its normal function and becomes a cycle clock (identical to the F8 multi-chip system write clock, except inverted).

Timing complexities render the capabilities associated with the TEST pin impractical for use in a user's application, but these capabilities are sufficient to enable Fairchild to implement a rapid method for thoroughly testing the F38E70.

F38E70 Clocks

The time base for the F38E70 may originate in one of four external sources. The four external configurations are shown in *Figure 5*. There is an internal 20 pF capacitor between XTL_1 and GND, and also between XTL_2 and GND. Thus, external capacitors are not required. In all external clock modes, the external time base frequency is divided by two to form the internal ϕ clock.

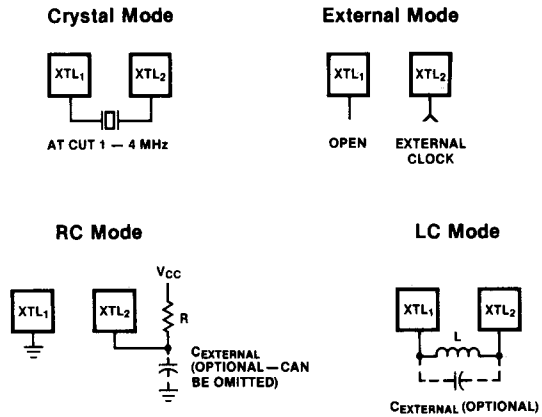
Instruction Set

The F38E70 executes the entire instruction set of the multi-chip F8 family (F3850 family), as shown in *Table 2*. Of course, the STORE instruction is of little use in the F38E70 because only read-only memory exists in the addressing range of the data counter, however, is incremented if STORE is executed).

A summary of programmable registers and ports is given in *Figure 6*, followed by a summary of the F38E70 (F8-compatible) instruction set.

Also, for convenient reference, a programming model of the F38E70 is given in *Figure 7*.

Fig. 5 Clock Configurations



Minimum $R = 4k\Omega$

$C = 20.5\ pF \pm 2.5\ pF + C_{EXTERNAL}$

$f_{MIN} \cong \frac{1}{1.1\,RC + 65\ ns}$

$f_{MAX} \cong \frac{1}{1.0\,RC + 15\ ns}$

Example with $C_{EXTERNAL} = 0$

$R = 15\ k\Omega \pm 5\%$

$f \cong 2.9\ MHz \pm 26\%$

Minimum $L = 0.1$ mH

Minimum $Q = 40$

Maximum $C_{EXTERNAL} = 30$ pF

$C = 10\ pF \pm 1.3\ pF + C_{EXTERNAL}$

$f \cong \frac{1}{2\pi\sqrt{LC}}$

Example with $C_{EXTERNAL} = 0$

$L = 0.3\ mH \pm 10\%$

$f \cong 3.0\ MHz \pm 10\%$

Table 2 F38E70 Instruction Set

Accumulator Group Instructions

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
Add Carry	LNK		ACC←(ACC)+CRY	19	1	1	1/0	1/0	1/0	1/0
Add Immediate	AI	ii	ACC←(ACC) ∧ H'ii'	24 ii	2	2.5	1/0	1/0	1/0	1/0
AND Immediate	NI	ii	ACC←(ACC) H'ii'	21 ii	2	2.5	0	1/0	0	1/0
Clear	CLR		ACC←H'00'	70	1	1				
Compare Immediate	CI	ii	H'ii'+($\overline{ACC}$)+1	25 ii	2	2.5	1/0	1/0	1/0	1/0
Complement	COM		ACC←(ACC)⊕H'FF'	18	1	1	0	1/0	0	1/0
Exclusive OR Immediate	XI	ii	ACC←(ACC)⊕H'ii'	23 ii	2	2.5	0	1/0	0	1/0
Increment	INC		ACC←(ACC)+1	1F	1	1	1/0	1/0	1/0	1/0
Load Immediate	LI	ii	ACC←H'ii'	20 ii	2	2.5	—	—	—	—
Load Immediate Short	LIS	i	ACC←H'0i'	7 i	1	1	—	—	—	—
OR Immediate	OI	ii	ACC←(ACC) V H'ii'	22 ii	2	2.5	0	1/0	0	1/0
Shift Left One	SL	1	SHIFT LEFT 1	13	1	1	0	1/0	0	1/0
Shift Left Four	SL	4	SHIFT LEFT 4	15	1	1	0	1/0	0	1/0
Shift Right One	SR	1	SHIFT RIGHT 1	12	1	1	0	1/0	0	1
Shift Right Four	SR	4	SHIFT RIGHT 4	14	1	1	0	1/0	0	1

Branch Instructions

(In All Conditional Branches, P0 (P0)+2 if the Test Conditions Are Not Met. Execution Is Complete in 30 Cycles.)

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
Branch on Carry	BC	aa	P0←[(P0)+1]+H'aa' if CRY=1	82 aa	2	3/3.5**	—	—	—	—
Branch on Positive	BP	aa	P0←[(P0)+1]+H'aa' if SIGN=1	81 aa	2	3/3.5**	—	—	—	—
Branch on Zero	BZ	aa	P0←[(P0)+1]+H'aa' if ZERO=1	84 aa	2	3/3.5**	—	—	—	—
Branch on True	BT	t,aa	P0←[(P0)+1]+H'aa' if any test is true	8t aa	2	3/3.5**	—	—	—	—
			t = TEST CONDITION							
			2^2 | 2^1 | 2^0							
			ZERO | CRY | SIGN							
Branch if Negative	BM	aa	P0←[(P0)+1]+H'aa' if SIGN=0	91 aa	2	3/3.5**	—	—	—	—
Branch if No Carry	BNC	aa	P0←[(P0)+1]+H'aa' if CARRY≠0	92 aa	2	3/3.5**	—	—	—	—
Branch if No Overflow	BNO	aa	P0←[(P0)+1]+H'aa' if OVF=0	98 aa	2	3/3.5**	—	—	—	—
Branch if Not Zero	BNZ	aa	P0←[(P0)+1]+H'aa' if ZERO=0	94 aa	2	3/3.5**	—	—	—	—
Branch if False Test	BF	t,aa	P0←[(P0)+1]+H'aa' if all false test bits	9t aa	2	3/3.5**	—	—	—	—
			t = TEST CONDITION							
			2^3 | 2^2 | 2^1 | 2^0							
			OVF | ZERO | CRY | SIGN							
Branch if ISAR (Lower) 7	BR7	aa	P0←[(P0)+1]+H'aa' if ISARL≠7	8F aa	2	2.5	—	—	—	—
			P0←(P0)+2 if ISARL=7			2.0	—	—	—	—
Branch Relative	BR	aa	P0←[(P0)+1]+H'aa'	90 aa	2	3.5	—	—	—	—
Jump*	JMP	aaaa	P0←H'aaaa'	29 aaaa	3	5.5	—	—	—	—

Memory Reference Instructions (In All Memory Reference Instructions, the Data Counter Is Incremented DC←DC•1.)

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
Add Binary	AM		ACC←(ACC)+[(DC)]	88	1	2.5	1/0	1/0	1/0	1/0
Add Decimal	AMD		ACC←(ACC)+[(DC)]	89	1	2.5	1/0	1/0	1/0	1/0
AND	NM		ACC←(ACC) ∧ [(DC)]	8A	1	2.5	0	1/0	0	1/0
Compare	CM		[(DC)]+($\overline{ACC}$)+1	8D	1	2.5	1/0	1/0	1/0	1/0
Exclusive OR	XM		ACC←(ACC)⊕[(DC)]	8C	1	2.5	0	1/0	0	1/0
Load	LM		ACC←[(DC)]	16	1	2.5	—	—	—	—
Logical OR	OM		ACC←(ACC) V [(DC)]	8B	1	2.5	0	1/0	0	1/0
Store	ST		(DC)←(ACC)	17	1	2.5	—	—	—	—

*Privileged instruction

** 3.5 cycles if branch taken.

Note

JMP and PI change accumulator contents to the high byte address.

Table 2 F38E70 Instruction Set (Cont.)

Address Register Group Instructions

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
Add to Data Counter	ADC		DC←(DC)+(ACC)	8E	1	2.5	—	—	—	—
Call to Subroutine	PK*		P←(P0); P0U←(r12); PL←(r13)	0C	1	4	—	—	—	—
Call to Subroutine Immediate	PI*	aaaa	P←(P); P0←H'aaaa'‡	28 aaaa	3	6.5	—	—	—	—
Exchange DC	XDC		DC←→DC1	2C	1	2	—	—	—	—
Load Data Counter	LR	DC,Q	DCU←(r14); DCL←(r15)	0F	1	4	—	—	—	—
Load Data Counter	LR	DC,H	DCU←(r10); DCL←(r11)	10	1	4	—	—	—	—
Load DC Immediate	DCI	aaaa	DC←H'aaaa'	2A aaaa	3	6	—	—	—	—
Load Program Counter	LR	P0,Q	P0U←(r14); P0L←(r15)	0D	1	4	—	—	—	—
Load Stack Register	LR	P,K	PU←(r12); PL←(r13)	09	1	4	—	—	—	—
Return From Subroutine	POP*		P0←(P)	1C	1	2	—	—	—	—
Store Data Counter	LR	Q,DC	r14←(DCU); r15←(DCL)	0E	1	4	—	—	—	—
Store Data Counter	LR	H,DC	r10←(DCU); r11←(DCL)	11	1	4	—	—	—	—
Store Stack Register	LR	K,P	r12←(PU); r13←(P)	08	1	4	—	—	—	—

Scratchpad Register Instructions (Refer to Scratchpad Addressing Modes.)

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
Add Binary	AS	r	ACC←(ACC)+(r)	Cr	1	1	1/0	1/0	1/0	1/0
Add Decimal	ASD	r	ACC←(ACC)+(r)	Dr	1	2	1/0	1/0	1/0	1/0
Decrement	DS	r	r←(r)+H'FF'	3r	1	1.5	1/0	1/0	1/0	1/0
Load	LR	A,r	ACC←(r)	4r	1	1	—	—	—	—
Load	LR	A,KU	ACC←(r12)	00	1	1	—	—	—	—
Load	LR	A,KL	ACC←(r13)	01	1	1	—	—	—	—
Load	LR	A,QU	ACC←(r14)	02	1	1	—	—	—	—
Load	LR	A,QL	ACC←(r15)	03	1	1	—	—	—	—
Load	LR	r,A	r←(ACC)	5r	1	1	—	—	—	—
Load	LR	KU,A	r12←(ACC)	04	1	1	—	—	—	—
Load	LR	KL,A	r13←(ACC)	05	1	1	—	—	—	—
Load	LR	QU,A	r14←(ACC)	06	1	1	—	—	—	—
Load	LR	QL,A	r15←(ACC)	07	1	1	—	—	—	—
AND	NS	r	ACC←(ACC) ∧ (r)	Fr	1	1	0	1/0	0	1/0
Exclusive OR	XS	r	ACC←(ACC)⊕(r)	Er	1	1	0	1/0	0	1/0

Miscellaneous Instructions

Operation	Mnemonic OP Code	Operand	Function	Machine Code	Bytes	Cycles	Status Bits OVF	ZERO	CRY	SIGN
Disable Interrupt	DI		RESET ICB	1A	1	2	—	—	—	—
Enable Interrupt*	EI		SET ICB	1B	1	2	—	—	—	—
Input	IN	aa	ACC←(INPUT PORT aa)	26 aa	2	4	0	1/0	0	1/0
Input Short	INS	a	ACC←(INPUT PORT a)	Aa	1	4***	0	1/0	0	1/0
Load ISAR	LR	IS,A	ISAR←(ACC)	0B	1	1	—	—	—	—
Load ISAR Lower	LISL	a	ISARL←a	01101a**	1	1	—	—	—	—
Load ISAR Upper	LISU	a	ISARU←a	01100a**	1	1	—	—	—	—
Load Status Register*	LR	W,J	W←(r9)	1D	1	2	1/0	1/0	1/0	1/0
No-Operation	NOP		P0←(P0)+1	2B	1	1	—	—	—	—
Output	OUT	aa	OUTPUT PORT aa←(ACC)	27 aa	2	4	—	—	—	—
Output Short	OUTS	a	OUTPUT PORT a←(ACC)	Ba	1	4***	—	—	—	—
Store ISAR	LR	A,IS	ACC←(ISAR)	0A	1	1	—	—	—	—
Store Status Register	LR	J,W	r9←(W)	1E	1	1	—	—	—	—

*Privileged instruction
**3-bit octal digit
***Two machine cycles for CPU ports
‡Contents of ACC destroyed

Notes
Each lower case character represents a hexadecimal digit.
Each cycle equals four machine clock periods.
Lower case denotes variables specified by the programmer.

Function Definitons

←	is replaced by
( )	the contents of
(—)	binary ones complement of
+	arithmetic add (binary or decimal)
⊕	logical OR exclusive
Λ	logical AND
V	logical OR inclusive
H'#'	hexadecimal digit

Register Names

a	address variable
A	accumulator
DC	data counter (indirect address register)
DC1	data counter #1 (auxiliary data counter)
DCL	least significant eight bits of data counter addressed
DCU	most significant eight bits of data counter addressed
H	scratchpad register #10 and #11
i and ii	immediate operand
ICB	interrupt control bit
IS	indirect scratchpad address register
ISAR	indirect scratchpad address register
ISARL	least significant three bits of ISAR
ISARU	most significant three bits of ISAR
J	scratchpad register #9
K	registers #12 and #13
KL	register #13
KU	register #12
P0	program counter
P0L	least significant eight bits of program counter
P0U	most significant eight bits of program counter
P	stack register
PL	least significant eight bits of program counter
PU	most significant eight bits of active stack register
Q	registers #14 and #15
QL	register #15
QU	register #14
r	scratchpad register (any address through 11)
W	status register

Scratchpad Addressing Modes (Machine Code Format)

r = C	(hexadecimal) register addressed by ISAR (unmodified)
r = D	(hexadecimal) register addressed by ISAR; ISARL incremented
r = E	(hexadecimal) register addressed by ISAR; ISARL decremented
r = F	(no operation performed)
r = 0-B	(hexadecimal) register 0 through 11 addressed directly from the instruction

Status Register

—	no change in condition
1/0	is set to 1 or 0, depending on conditions
CRY	carry flag

EPROM Programming

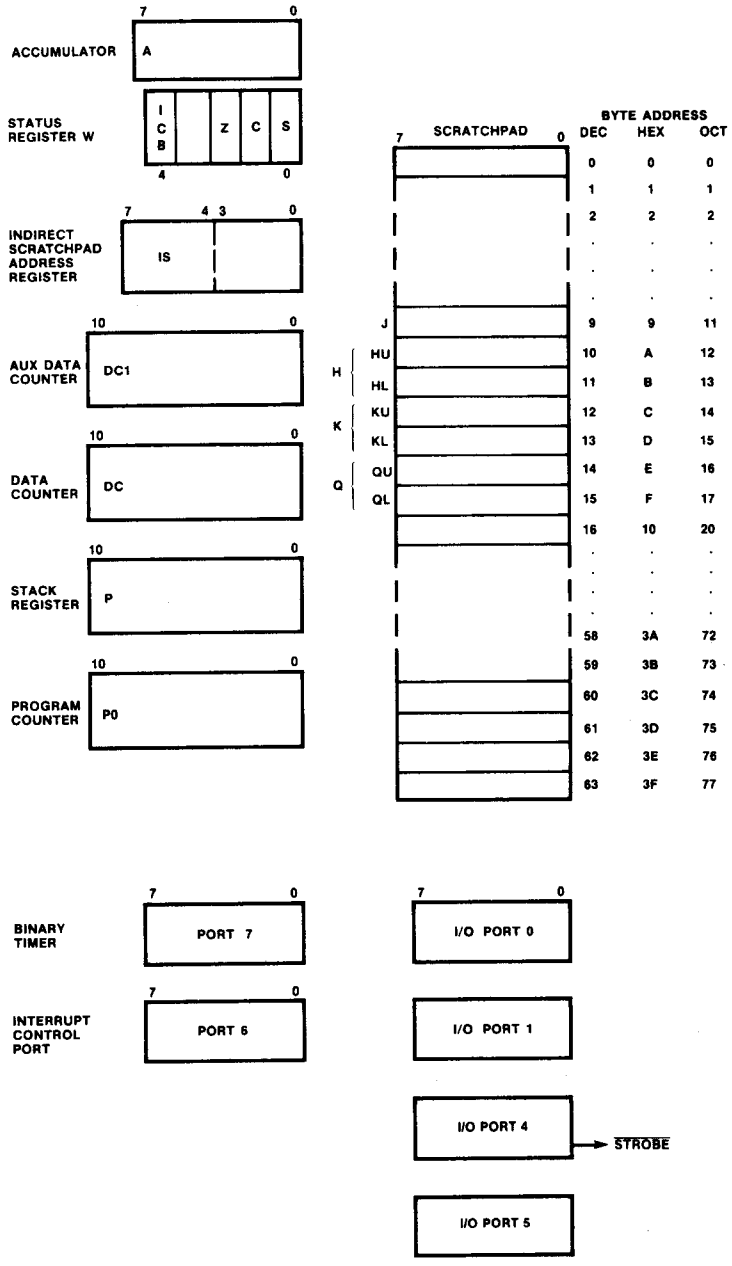
When V_{PP} is applied to the TEST 1 pin, the device goes into the program or verify mode and the I/O ports take on the different functions of $\overline{\text{DATA IN}}$ (port 5), DATA OUT (port 4), EPROM address (11 pins of ports 0 and 1), and $\overline{\text{PROG}}$ (port 1_6). The verify mode exists when $\overline{\text{PROG}}$ is HIGH and TEST 2 = V_{CC}. Port 4 outputs the data content of the EPROM according to the address A_0 through A_{10}. The logical sense is true, and for an unprogrammed location, the outputs are high. During verify mode, the data on port 5 has no effect on the port 4 output. The program mode exists when $\overline{\text{PROG}}$ is taken low. All addresses and DATA IN must be stable before going into this mode. During this mode, the data appearing on port 5 is "burned in" to the EPROM. Note that the sense of port 5 is logically false. At the same time, port 4 outputs the data on the internal data bus, which is exactly equal to the inversion of the data going in on port 5. Port 4 does not indicate satisfactory completion of the EPROM programming in the program mode. The $\overline{\text{PROG}}$ pin must be high before the V_{PP} is applied in order to prevent a programming error.

CAUTION
Applying V_{PP} to the TEST 1/V_{PP} pin without the presence of V_{CC} will damage the device.

F38E70 Erasing Instructions

The contents of the F38E70 EPROM can be erased by exposure to high-intensity shortwave ultraviolet (UV) light with a wavelength of 2537 Angstroms (Å). This can be accomplished with ultraviolet light EPROM erasure devices that are available from several U.S. manufacturers. These erasure devices contain a UV light source, which is usually placed approximately 1 or 2 inches from the EPROM so that the transparent window on top of the device is illuminated. The minimum required integrated dose (intensity x exposure time) of UV light energy incident on the window of the device in order to reliably ensure complete erasure is 15 watt-sec/cm^2. The UV erasure unit should be periodically calibrated if minimum exposure times are to be used. (Minimum exposure times range from 10 to 45 minutes, depending on the model type and age of UV lamp.) If longer exposure times are possible, variations in the output light intensity of the UV light source are not critical.

Fig. 6 Programmable Registers and Ports



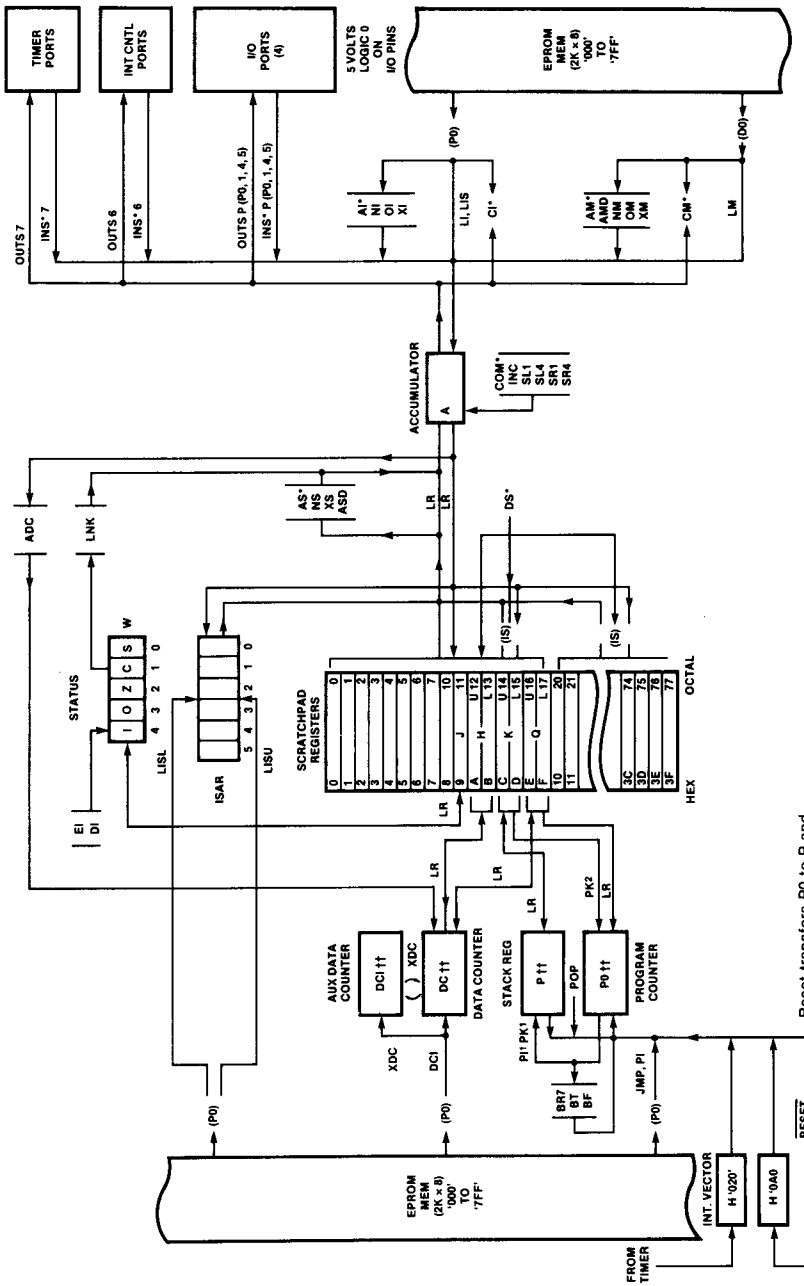


Fig. 7 Programming Model

Note

The instructions PI and PK are shown in two sequential parts (PI_1, PI_2, and PK_1, PK_2).

*These instructions set status.

†The value of the external interrupt input is loaded to bit 7 of the accumulator (with bits 0 through 6 loaded with zeros) when the instruction 'INS 6' is executed. This instruction also sets status.

††P0, P, DC, and DC1 are 11 bit registers.

4

Supplementary Notes

For total software compatibility when expanding into a multi-chip configuration, the F3871 Peripheral Input/Output circuit should be used. The F3871 has the same improved timer (binary count, readable, and three modes of operation) and ready strobe output as the F38E70.

The interrupt control bit of the status register is automatically reset when an interrupt request is acknowledged. It is then the programmer's responsibility to determine when ICB is again set (by executing an EI instruction). This action prevents an interrupt service routine from being interrupted unless the programmer so desires.

When reading the interrupt control port (port 6), bit 7 of the accumulator is loaded with the actual logic level being applied to the EXT INT pin, regardless of the status of ICP bit 2 (the EXT INT active level bit); that is, if EXT INT is at +5 V, bit 7 of the accumulator is set to a logic 1, but if EXT INT is at GND, accumulator bit 7 is reset to logic 0.

In the "F38E70 (F8-compatible) Instruction Set" summary, the number of cycles shown is "nominal" machine cycles. A nominal machine cycle is defined as 4 ϕ clock periods, thus requiring 2 μs for a 2 MHz ϕ clock frequency (4 MHz external time base frequency).

Also, the summary uses the following nomenclature for register names:

F8	F38E70	
PC_0 = P0		Program Counter
PC_1 = P		Stack Register
DC_0 = DC		Data Counter
DC_1 = DC1		Auxiliary Data Counter

This nomenclature is used in order to be consistent with the assembly language mnemonics.

For the F38E70, execution of an INS or OUTS instruction requires two machine cycles for ports 0 and 1, while ports 4 and 5 require four machine cycles.

When an external reset of the F38E70 occurs, P0 is pushed into P and the old contents of P are lost. It must be noted that an external reset is recognized at the start of the machine cycle and not necessarily at the end of an instruction. Thus, if the F38E70 is executing a multi-cycle instruction, that instruction is not completed and the contents of P upon reset may not necessarily be the address of the instruction that would have been executed next. It may, for example, point to an immediate operand if the reset occurred during the second cycle of an LI or CI instruction. Additionally, several instructions (JMP, PI, PK, LR, P0, Q) as well as the interrupt acknowledge sequence modify P0 in parts. That is, they alter P0 by loading first one part, then the other, and the entire operation takes more than one cycle. Should reset occur during this modification process, the value pushed into P is part of the old P0 (the as-yet unmodified part) and part of the new P0 (the already modified part). Thus, care should be taken (perhaps by external gating) to ensure that reset does not occur at an undesirable time if any significance is to be given to the contents of P after a reset occurs.

Absolute Maximum Ratings

The absolute maximum ratings of the F38E70 are as follows:

Temperature (Ambient Under Bias)	0°C, +70°C
Storage Temperature	−55°C, +150°C
Voltage on any Pin with Respect to Ground (Except Test Pin)	−1.0 V, +7 V
Test Pin Voltage with Respect to V_{SS}	−1.0 V, +27 V

CAUTION

Applying V_{PP} to the TEST 1/V_{PP} pin without the presence of V_{CC} will damage the device.

Power Dissipation	1.0 W

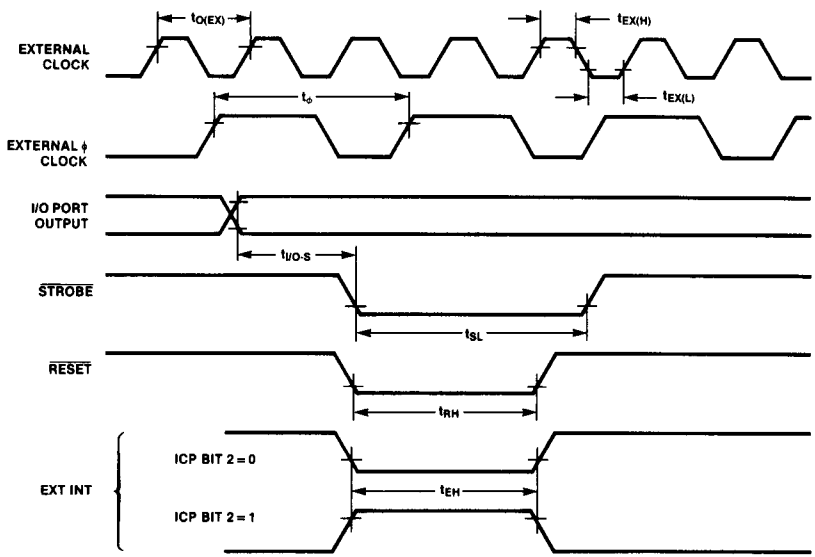
These are stress ratings only, and functional operation at these ratings, or under any conditions above those indicated in this data sheet, is not implied. Exposure to the absolute maximum rating conditions for extended periods of time may affect device reliability, and exposure to stresses greater than those listed may cause permanent damage to the device.

Timing Characteristics $V_{CC} = +5\ V \pm 10\%$, $T_A = 0°C$ to $+70°C$

Signal	Symbol	Characteristic	Min	Max	Unit	Comments (Note 3)
XTL_1 XTL_2	t_0(XTL)	Time Base Period, Crystal Mode	250	5000	ns	4 MHz–2 MHz
	t_0(LC)	Time Base Period, LC Mode	250	5000	ns	4 MHz–2 MHz
	t_0(RC)	Time Base Period, RC Mode	250	5000	ns	4 MHz–2 MHz
	t_0(EX)	Time Base Period, External Mode	250	5000	ns	4 MHz–2 MHz
	t_{EX}(H)	External Clock Pulse Width, High	90	t_0(EX)-100	ns	
	t_{EX}(L)	External Clock Pulse Width, Low	90	t_0(EX)-100	ns	
ϕ	$t\phi$	Internal ϕ Clock Period	$2t_0$ typ.		ns	0.5 μs at 4 MHz ext. time base
$\overline{STROBE}$	$t_{I/O-S}$	Port Output to $\overline{STROBE}$ Delay	$3t\phi - 1000$ min $3t\phi + 250$ max		ns	Note 1
	t_{SL}	$\overline{STROBE}$ Pulse Width, Low	$8t\phi - 250$ min $12t + 250$ max		ns	
$\overline{RESET}$	t_{RH}	$\overline{RESET}$ Hold Time, Low	$6t\phi + 750$ min		ns	
EXT INT	t_{EH}	EXT INT Hold Time, Active State	$6t\phi + 750$ min		ns	Note 2

Notes
1. Load is 50 pF plus 1 standard TTL input.
2. Specification is applicable when the timer is in the interval timer mode. See "Timer Characteristics" for EXT INT requirements when in the pulse width measurement mode or the event counter mode.
3. The timing diagrams are given in *Figure 8*.

Fig. 8 Timing Diagrams



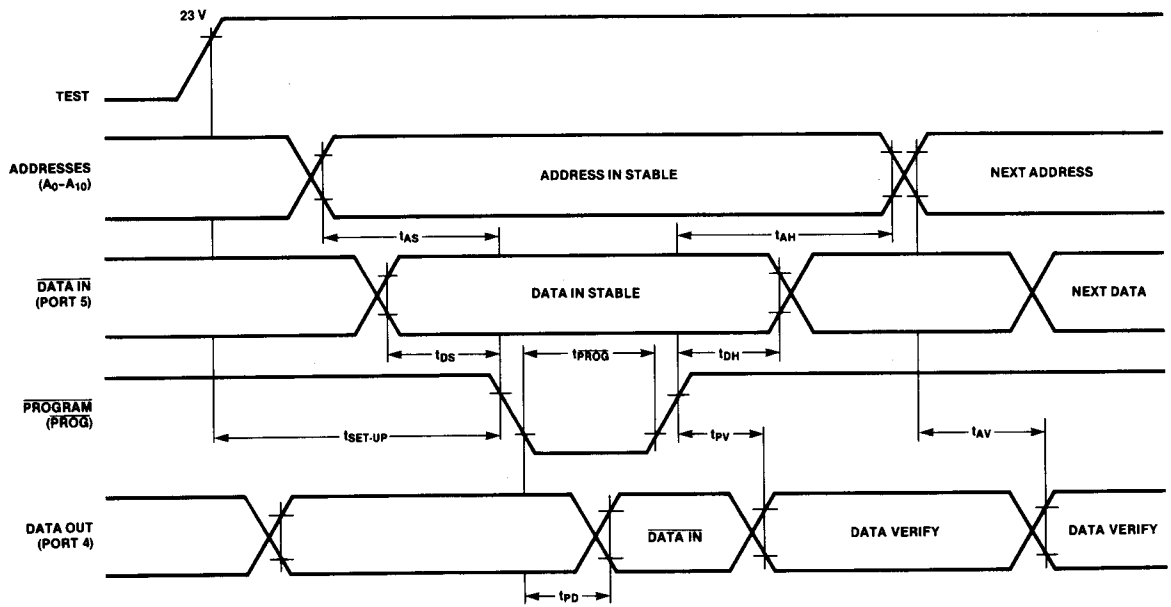
Note
All measurements are referenced to V_{IL} max., V_{IH} min., V_{OL} max., or V_{OH} min.

Program/Verify Timing

Symbol	Parameter	Min	Max	Unit
t_{SET-UP}	23 V Applied to $\overline{PROG}$	5		μs
t_{AS}	Address Set-up Time	1		μs
t_{AH}	Address Hold Time	1		μs
t_{DS}	Data Set-up Time	1		μs
t_{DH}	Data Hold Time	1		μs
t_{AV}	Address to Data Out in Verify		5	μs
t_{PV}	$\overline{PROG}$ to Data Out in Verify		2	μs
t_{PD}	$\overline{PROG}$ to Data Out in Programming		5	μs
t_{PROG}	Programming Time	50	60	ms

Note
Timing diagrams are given in *Figure 9*.

Fig. 9 Program/Verify Timing Diagrams



DC Characteristics $V_{CC} = 5\ V \pm 10\%$, $T_A = 0°C$ to $+70°C$

Symbol	Characteristic	Min	Typ	Max	Unit	Test Conditions
I_{CC}	Power Supply Current		70	100	mA	Outputs Open
P_D	Power Dissipation		375	550	mW	Outputs Open
V_{IHEX}	External Clock Input High Voltage	2.4		5.8	V	
V_{ILHEX}	External Clock Input Low Voltage	−0.3		0.6	V	
I_{HEX}	External Clock Input High Current			100	μA	$V_{IHEX} = 2.4\ V$
I_{ILEX}	External Clock Input Low Current			−100	μA	$V_{ILEX} = 0.6\ V$
V_{IH}	Input High Voltage	2.0		5.8	V	
V_{IL}	Input Low Voltage	−0.3		0.8	V	
I_{IH}	Input High Current (Except Open-Drain and Direct-Drive I/O Ports)			100	μA	$V_{IH} = 2.4\ V$, Internal Pull-Up
I_{IL}	Input Low Current (Except Open-Drain and Direct-Drive Ports)			−1.6	mA	$V_{IL} = 0.4\ V$
I_{LOD}	Leakage Current (Open-Drain Ports)			10	μA	Pull-Down, Device Off
I_{OH}	Output High Current (Except Open-Drain and Direct-Drive Ports)	−100			μA	$V_{OH} = 2.4\ V$
I_{OHDD}	Output Drive Current (Direct-Drive Ports)	−1.5		−8	mA	$V_{OH} = 0.7\ V$ to $1.5\ V$
I_{OL}	Output Low Current	1.8			mA	$V_{OL} = 0.4\ V$
I_{OHS}	Output High Current ($\overline{STROBE}$ Output)	−300			μA	$V_{OH} = 2.4\ V$
I_{OLS}	Output Low Current ($\overline{STROBE}$ Output)	5.0			mA	$V_{OL} = 0.4\ V$
V_{TEST}	Test Pin Voltage for Program/Verify Mode	23	23.5	24	V	
I_{TEST}	Test Pin Current for Program/Verify Mode		20	30	mA	$V_{\overline{PROG}} = 0.4\ V$, $V_{TEST} = 24\ V$

Capacitance $T_A = 25°C$, $f = 2$ MHz

Symbol	Characteristic	Min	Max	Unit	Test Condition
C_{IN}	Input Capacitance: I/O Ports, $\overline{RESET}$, EXT INT		7	pF	Unmeasured pins returned to GND
C_{XTL}	Input Capacitance: XTL_1, XTL_2	18	23	pf	

Typical Thermal Resistance Values

Plastic	
θ_{JA} (Junction to ambient)	60°C/W (still air)
θ_{JA} (Junction to case)	42°C/W
Ceramic	
θ_{JA} (Junction to ambient)	48°C/W (still air)
θ_{JA} (Junction to case)	33°C/W

Ordering Information

Part Number	Package	Temperature Range*
F38E70DC	Ceramic	C
F38E70DL	Ceramic	L
F38E70DM	Ceramic	M
F38E70PC	Plastic	C
F38E70PL	Plastic	L
F38E70PM	Plastic	M

*C = Commercial Temperature Range 0°C to +70°C
L = Limited Temperature Range −40°C to +85°C
M = Military Temperature Range −55°C to +125°C