

# F3861 Peripheral Input/Output

Microprocessor Product

### Description

The Fairchild F3861 Peripheral Input/Output (PIO) device provides two 8-bit I/O ports, external interrupt, and a programmable timer. An 8-bit wide bidirectional data bus transfers I/O data bytes between the F3870 Central Processing Unit (CPU) and the PIO.

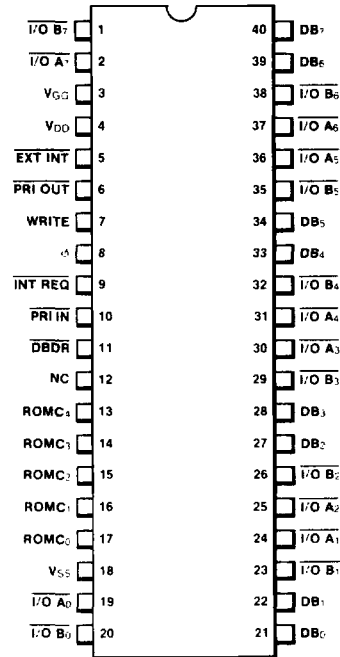
The PIO is used in systems that require the I/O capability and interrupt functions of the F3851 PSU but do not need the read-only memory (ROM) storage of the PSU. The PIO is pin-compatible with the PSU.

The F3861 PIO has five versions available, each with its own set of preassigned I/O port addresses and interrupt vectors.

The F3861 is manufactured using isoplanar N-channel, silicon-gate technology; therefore, power dissipation is very low (less than 250 mW).

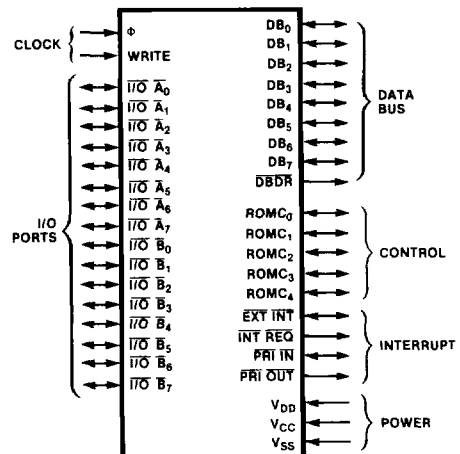
- 16 Bidirectional, Individually Controlled I/O Lines Organized as Two 8-bit Ports
- Programmable Timer--Preset, Start, Stop, and Read-Back Ability; Selectable Timer Count Rates
- Full Interrupt Level--Daisy-Chain Expandable, Independent Interrupt Address Vectors for Timer and External Interrupt
- Pulse Width Measurement Capability
- TTL and LSTTL Compatible
- +5 V and +12 V Power Supplies
- 2-MHz Operation
- Low Power Dissipation, Typically Less Than 250 mW

### Connection Diagram



3

### Signal Functions



**Device Organization**

The peripheral input/output device includes I/O logic, timer logic, interrupt logic, data bus logic and control logic, as illustrated in figure 1.

The interrupt logic responds to an interrupt request signal originating from internal timer logic or an external device. Based on priority considerations, the interrupt request is passed on to the F3850 CPU. The programmable timer uses a polynomial shift register in conjunction with interrupt logic to generate real-time intervals.

The 8-bit data bus in the PIO is the main path for transfer of information between the F3850 CPU and other devices in the F8 microprocessor system. The device contains four preassigned I/O port addresses: the two lowest are assigned to the two I/O ports (A and B) and are used to transfer data to and from external devices. The other two I/O addresses are assigned to two internal registers of the PIO that control interrupt logic and are treated as I/O ports.

**Signal Descriptions**

The F3861 input and output signals are described in table 1.

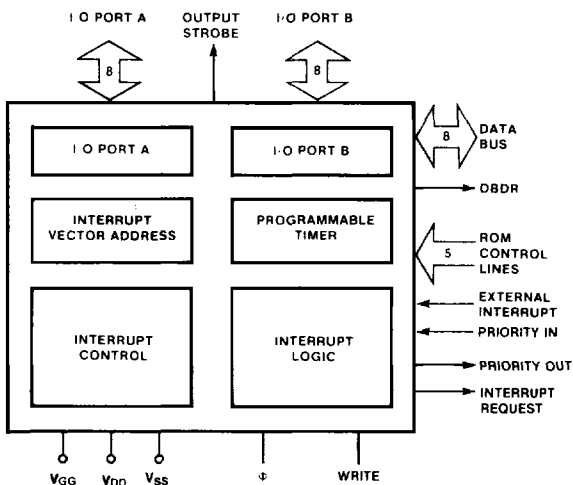
**System Clock Timing**

All timing within the F3861 PIO is controlled by the  $\phi$  and WRITE signals, which are input from the F3850 CPU. Refer to the F3850 data sheet for a description of these clock signals. The WRITE clock refreshes and updates PIO registers, which are dynamic. The  $\phi$  clock also drives the programmable timer.

**I/O Ports**

The PIO has two bidirectional 8-bit I/O ports used to transmit data between itself and external devices. In binary notation, the address for port A is XXXXXX00 and for port B is XXXXXX01, where the X binary digits are the unique I/O port select code for the PIO (see table 2). For example, if the port select code is 000001, port A may be called port 4 and port B may be called port 5. (The PIO port select code is never designated as all 0s, since ports 0 and 1 are reserved for the F3850 CPU.) In addition, the interrupt control Port (ICP) is addressed as port XXXXXX10 and the binary timer is addressed as port XXXXXX11, which become ports 6 and 7, respectively, for the port select code example given above.

**Figure 1 F3861 Block Diagram**



**Table 2 F3861 Port Addresses**

ADDRESS	ASSIGNED TO
XXXX XX00	I/O Port A
XXXX XX01	I/O Port B
XXXX XX10	Interrupt Control Register
XXXX XX11	Programmable Timer

The port and interrupt address vector assignments are given in table 3.

**Table 3 F3861 Port and Address Assignments (HEX)**

Version	Port Addresses	Interrupt Address Vector	
		Timer	External
F3861A	4-7	0600	068C
F3861B	8-B	0340	03C0
F3861C	20-23	0320	03A0
F3861D	24-27	0360	03E0
F3861E	4-7	0020	00A0

## F3861

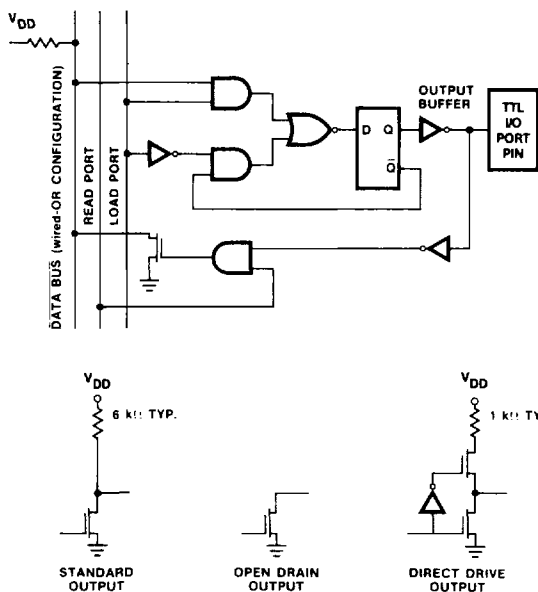
**Table 1 F3861 Signal Descriptions**

Mnemonic	Pin No.	Name	Description
<b>Clock</b> $\phi$ WRITE	8 7	Clock	The two clock input signals originate at the F3850 CPU.
<b>I/O Ports</b> I/O A <sub>0</sub> – I/O A <sub>7</sub>	19, 24, 25, 30 31, 36 37, 2	I/O Ports A	Bidirectional ports through which the PIO communicates with logic external to the microprocessor system.
I/O B <sub>0</sub> – I/O B <sub>7</sub>	20, 23, 226, 29, 31, 35, 38, 1	I/O Ports B	
<b>Control</b> ROMC <sub>0</sub> ROMC <sub>4</sub>	17, 16, 15, 14, 13	Read Only Memory Control	Input signals that originate at the F3850 CPU and control internal functions of the PIO.
<b>Data Bus</b> DB <sub>0</sub> – DB <sub>7</sub>	21, 22, 27, 28, 33, 34, 39, 40	Data Bus	Bidirectional three-state lines that link the PIO to all other devices within the microprocessor system.
DBDR	11	Data Bus Drive	A low output, open drain signal that indicates the data bus currently contains data flowing from the PIO.
<b>Interrupt</b> EXT INT	5	External Interrupt	A high-to-low transition on this input signal is interpreted as an interrupt request from an external device.
$\overline{\text{INT REQ}}$	9	Interrupt Request	This output signal is the INT REQ input to the F3850 CPU; it must be output low to interrupt the CPU, which occurs only if PRI IN is low and PIO interrupt control logic is requesting an interrupt.
$\overline{\text{PRI IN}}$	10	Priority In	Unless this input signal is low, the PIO does not set the INT REQ signal low in response to an interrupt.
$\overline{\text{PRI OUT}}$	6	Priority Out	This output signal becomes the PRI IN signal to the next device in the interrupt-priority daisy chain; it is output high unless the PRI IN signal is entering the PIO low and the PIO is not requesting an interrupt.
<b>Power</b> V <sub>DD</sub>	4	Power Supply	5 V (± 5%)
V <sub>GG</sub>	3	Power Supply	+ 12 V (± 5%)
V <sub>SS</sub>	18	Ground	System ground—0 V; V <sub>DD</sub> and V <sub>GG</sub> are referenced to V <sub>SS</sub> .

**Port Pin Description**

An output instruction (OUT or OUTS) causes the contents of the CPU accumulator (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 that is described later). The I/O pins on the PIO are logically inverted; the schematic of an I/O pin and available output drive options are shown in figure 2. Each output pin has an output latch that holds the data last output to that pin. The I/O ports of the PIO are configured in the standard pull-up option.

**Figure 2 I/O Pin Diagram with Output Buffer Options**



Each I/O port pin is a wire-AND structure between an internal output data latch and the external signal. The latch is loaded from the data bus. The output latches are not initialized by the system reset sequence.

When outputting data through an I/O port, the pin can be connected directly to a TTL gate input; data is input to the pin from a TTL device output. Since the I/O pin and the TTL device output are wire-ANDed, it is possible for the state of one to affect the transfer of data out from the I/O pin or in from the TTL device output. In most cases, therefore, I/O port bits should be set for a high level (logic 0) before data input to prevent incoming logic zeros from being masked by logic ones present at the port from previous outputs. However, the ability to mask bits of a port to logic 1 is useful during some input functions.

**Programmable Timer**

The 8-bit shift register, addressable as an I/O port, functions as a polynomial timer. This timer is loaded with a value of delay; it counts down this value of delay and, after the programmed interval, generates an interrupt through the interrupt logic of the PIO.

The OUT or OUTS instruction is used to load the interval value into the programmable timer; the port number is H'07', H'0B', H'23', or H'27', as appropriate. The timer times out after a time interval given by the product

$$(\text{period of } \Phi \text{ clock}) \times (\text{timer counts}) \times 31$$

The timer continues to run after a time-out; subsequent time-outs occur at intervals of 7905  $\Phi$  clock periods. The timer does not run if it is loaded with the value H'FF'.

**Interrupt Logic**

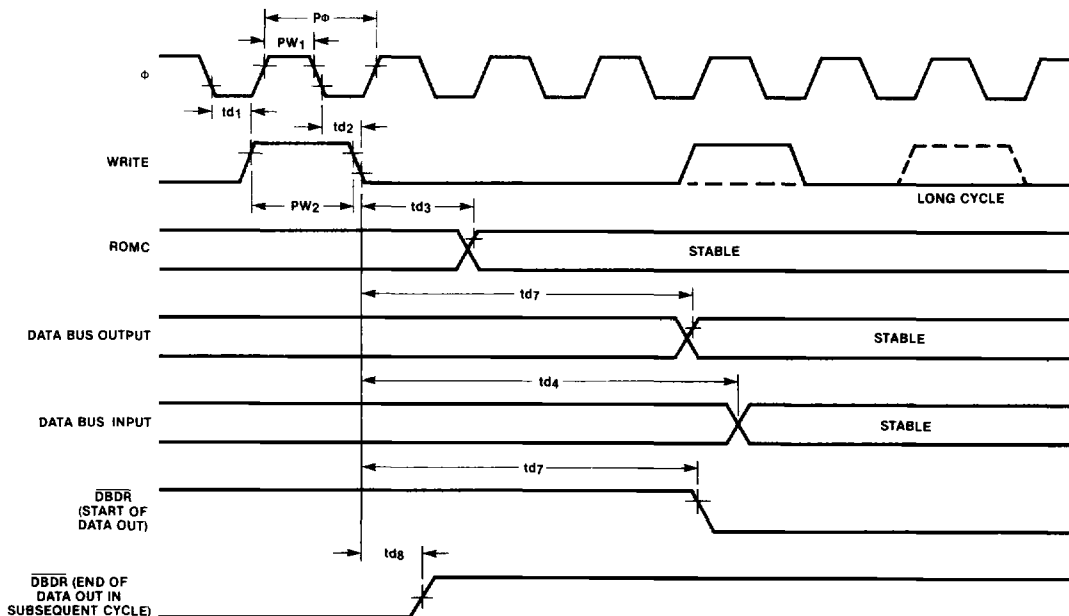
The interrupt logic block is programmed by output instructions to the interrupt control port (port H'06', H'0A', H'22', or H'26', as appropriate). Only the least significant two bits are used; their interpretation is as follows.

**Contents of ICP Interpretation**

Contents of ICP	Interpretation
B'XXXXXX00'	Disable all interrupts
B'XXXXXX01'	Enable external interrupt, disable timer interrupt
B'XXXXXX10'	Disable all interrupts
B'XXXXXX11'	Disable external interrupt, enable timer interrupt

Note: The X designation represents "don't care" binary digits.

Figure 3 PIO Data Bus Timing



3

### Instruction Execution

The PIO responds to signals that are output by the F3850 CPU in the course of implementing instruction cycles. Figure 3 illustrates timing during PIO data output to the data bus. This timing applies whenever a PIO is the data source. The PIO places data on the data bus, even in the worst case, in time for the setup required by any F3850 CPU destination. The PIO receives a byte input from the data bus when commanded by an output instruction to load one of its two I/O ports or internal registers. Data bus timing requirements for input to the PIO are also shown in figure 3; signal characteristics are given in the "Timing Characteristics" section. The data bus drive (DBDR) signal is low while data output by the PIO is stable on the data bus. Thus, a  $\overline{DBDR}$  low signal indicates that the data bus currently contains data flowing from a PIO. For systems with more than one program storage unit (PSU) or PIO, the DBDR output signals may be wire-ORed and the result used as a bus data flow direction indicator. The  $\overline{DBDR}$  signal may remain low until timing interval  $td$  of the next instruction cycle following the one in which  $\overline{DBDR}$  was set low.

The PIO device executes the OUT instruction in the same manner as the OUTS instruction; the same is true for the IN and INS instructions. The difference between the long- and short-form instructions is only in the source of the I/O address.

The F8 input/output instructions place the I/O port address on the data bus during one instruction cycle and then use the data bus in the following instruction cycle to do the actual I/O data movement. The read only memory control (ROMC) lines coming from the F3850 CPU signal the PIO that an I/O data movement is occurring during the current instruction cycle. Therefore, the PIO needs to recognize whether the contents of the data bus during the instruction cycle just prior matched any of its four assigned I/O addresses, wherever the ROMC lines indicate an I/O transfer. The address select logic constantly monitors the data bus for a match to any of the four addresses and holds the information of a match through the following cycle.

Input instructions that select a port cause the contents of the selected port to be placed on the data bus during the input cycle. Only the two I/O ports (lowest two addresses)

respond to input instructions. Output instructions that select a port transfer the contents of the data bus to that port. Outputs of the latches change at the end of the I/O transfer cycle.

### Interrupt Handling

A typical F8 system interrupt interconnection is shown in figure 4. Each PSU and PIO has a  $\overline{\text{PRI}} \text{ IN}$  and  $\overline{\text{PRI}} \text{ OUT}$  line so that they can be daisy-chained together in any order to form a priority level of interrupts. When a PIO receives an interrupt (either timer or external), it pulls its  $\overline{\text{PRI}} \text{ OUT}$  output signal high, signaling all lower priority peripherals that it has a higher priority interrupt request pending on the CPU. Also, when the PIO device  $\overline{\text{PRI}} \text{ IN}$  input signal is pulled high by a higher priority peripheral, signaling the PIO that there is a still higher priority interrupt request, it passes that signal along by pulling its  $\overline{\text{PRI}} \text{ OUT}$  signal high. When the CPU processes an interrupt request, it commands the interrupting device to place its interrupt vector address on the data bus. Only the device with a  $\overline{\text{PRI}} \text{ IN}$  signal low and an interrupt request pending responds. Should there be another lower priority device with a pending request, it does not respond at that time because its  $\overline{\text{PRI}} \text{ IN}$  input signal is high.

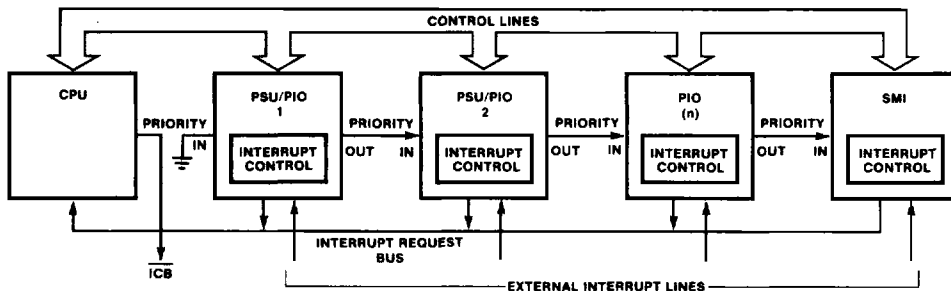
If there are both a timer interrupt request and an external interrupt request when the CPU starts to process the requests, the timer interrupt is handled first.

Within each local interrupt control circuit is a 16-bit interrupt address vector. This vector is the address to which the program counter is set after an interrupt is acknowledged and is therefore the address of the first executable instruction of the interrupt routine.

The interrupt address is unique to the version of the PIO device selected by the user. Fifteen bits are fixed: bits 0 through 6 and bits 8 through 15. Bit 7 ( $2^7$ ) is dependent on the type of interrupt. This bit is a 0 for internal timer-generated interrupts and a 1 for external interrupts. When the interrupt logic sends an interrupt request signal and the CPU is enabled to service it, the normal state sequence of the CPU is interrupted at the end of an instruction. The CPU signals the interrupt circuits through the five ROMC lines. The requesting local interrupt circuit sends a 16-bit interrupt address vector (from the interrupt address generator) onto the data bus in two consecutive bytes. The address is made available to the program counter through the address demultiplexer circuits. It is simultaneously made available to all other devices connected to the data bus and is the address of the next instruction to be executed. The program counter of each memory device is set with this new address while the stack register is loaded with the previous contents of the program counter. The information in the program counter is lost. Thus, the next instruction to be executed is determined by the value of the interrupt address vector.

The interrupt control bit (ICB) of the CPU (loaded in the W register) allows interrupts to be recognized. Clearing the ICB prevents acknowledgement of interrupts. The ICB is cleared during power-on, during external reset, and after an interrupt is acknowledged. The interrupt status of the PSU, PIO, or memory interface (MI) device is not affected by execution of the disable interrupt (DI) instruction from the CPU. At the conclusion of most instructions, the fetch logic checks the state of the interrupt request line. If an interrupt occurs the next instruction fetch cycle is suspended and the system is forced into an interrupt sequence.

Figure 4 F8 System Interrupt Interconnection



### Interrupt Sequence

Figure 5 details the interrupt sequence that occurs, whether the interrupt request is from an external source through the  $\overline{\text{EXT INT}}$  pin or from the PIO device internal timer. Events are labeled A through G.

#### Event A

An interrupt request must satisfy a set-up time requirement. If not satisfied, the  $\overline{\text{INT REQ}}$  signal delays going low until the next negative edge of the WRITE clock.

#### Event B

Event B represents the instruction being executed when the interrupt occurs. The last cycle of B is normally the instruction fetch for the next cycle. However, if B is not a privileged instruction and the CPU interrupt control bit is set, the last cycle becomes a freeze cycle rather than a fetch. At the end of the freeze cycle the interrupt request latches are inhibited from altering the interrupt daisy chain so that sufficient time is allowed for the daisy chain to settle.

If B is a privileged instruction, the instruction fetch is not replaced by a freeze cycle; instead, the fetch is performed and the next instruction is executed. Although unlikely to be encountered, a series of privileged instructions would be executed sequentially. One more instruction (a protected instruction) is executed after the last privileged instruction. The last cycle of the protected instruction then performs the freeze.

In figure 5, the dashed lines on the  $\overline{\text{EXT INT}}$  (EI) timing illustrate the last opportunity for the  $\overline{\text{EXT INT}}$  signal to cause the last cycle of a nonprotected instruction to become a freeze cycle. The freeze cycle is a short cycle (four  $\phi$  clock periods) in all cases except where B is the decrement scratchpad instruction, in which case the freeze cycle is a long cycle (six  $\phi$  clock periods).

The  $\overline{\text{INT REQ}}$  signal goes low on the next negative edge of the WRITE signal if both the  $\overline{\text{PRI IN}}$  signal is low and the appropriate interrupt enable bit of the ICP is set.

#### Event C

This is a no-operation (NO-OP) long cycle, allowing time for the  $\overline{\text{PRI IN}}/\overline{\text{PRI OUT}}$  chain to settle. At a 2-MHz  $\phi$  clock rate, a total of seven PIO, PSU, or MI devices can be daisy-chained without the need for look-ahead logic.

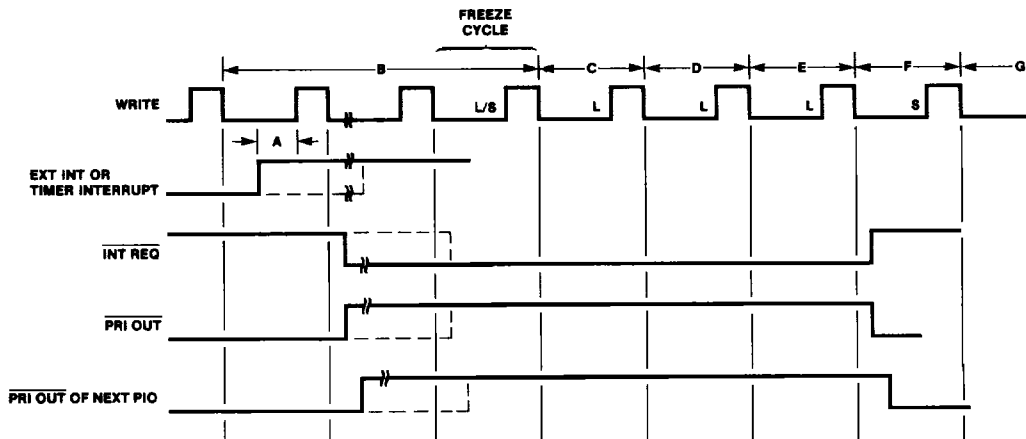
#### Event D

In PSU circuits, the program counter (PO) is pushed to the stack register (P) to save the return address. The interrupting PIO places the lower eight bits of the interrupt vector address onto the data bus. This is always a long cycle.

#### Event E

In this long cycle, the PIO places the upper eight bits of the interrupt vector address onto the data bus.

Figure 5 Interrupt Sequence



**Event F**

In this short cycle, the PIO interrupting interrupt request latch is cleared. Also, the CPU interrupt control bit is cleared, thus disabling interrupts until an EI instruction is performed. Additionally, during Event F, the **PRI IN/PRI OUT** daisy-chain freeze is removed, since the interrupt vector address has been passed to the CPU. Another action is the fetch of the instruction from the interrupt address.

**Event G**

This event starts executing the first instruction of the interrupt service routine.

**Summary of Interrupt Sequence**

For the PIO, the interrupt response time is defined as the time elapsed between the occurrence of the **EXT INT** signal going active (or the timer transition to H'N') and the beginning of execution of the first instruction of the interrupt service routine. The interrupt response time is a variable dependent on what the microprocessor is doing when the interrupt request occurs.

As shown in figure 5, the minimum interrupt response time is three long cycles plus two short cycles plus one write clock pulse width plus a set-up time of an **EXT INT** signal prior to the leading edge of the write pulse, a total of 27  $\phi$  clock periods plus the set-up time. At 2 MHz, this is 14.25  $\mu$ s. Although the maximum could theoretically be infinite, a practical maximum is 35  $\mu$ s (based on the interrupt request occurring near the beginning of a PI and LR K, P sequence).

**ROMC States**

Table 4 shows the function performed by the PIO device for each ROMC command. Each function is performed entirely within one machine cycle (one cycle of the WRITE clock). All other ROMC states are decoded as NO-OP.

**Table 4 PIO Functions Versus ROMC States**

ROMC State		PIO Functions
Binary	Hex	
01111	0F	If this circuit is interrupting and is highest in the priority chain, move lower half of interrupt vector into the data bus.
10000	10	Place interrupt circuitry in an inhibit state that prevents altering the interrupt priority chain.
10011	13	If the contents of the data bus in the prior cycle was an address of I/O ports on this device, move the current contents of the data bus into the appropriate port (I/O A, I/O B, timer or control).
11011	1B	If the contents of the data bus in the prior cycle was an address of I/O ports on this device, move the contents of the appropriate I/O port onto the data bus (I/O A or I/O B).

**Timing Characteristics**

Timing signals are illustrated in figures 3, 6, and 7; the signal timing characteristics are presented in table 5.

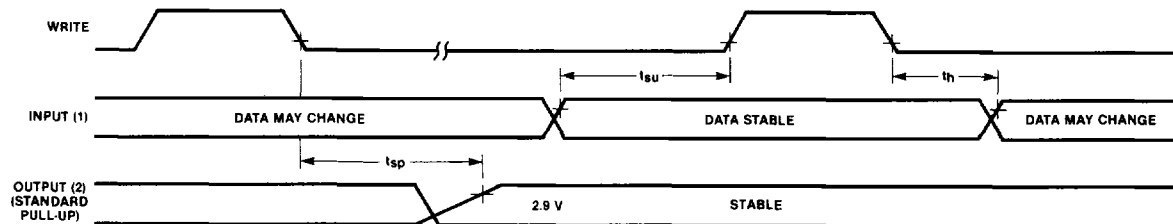
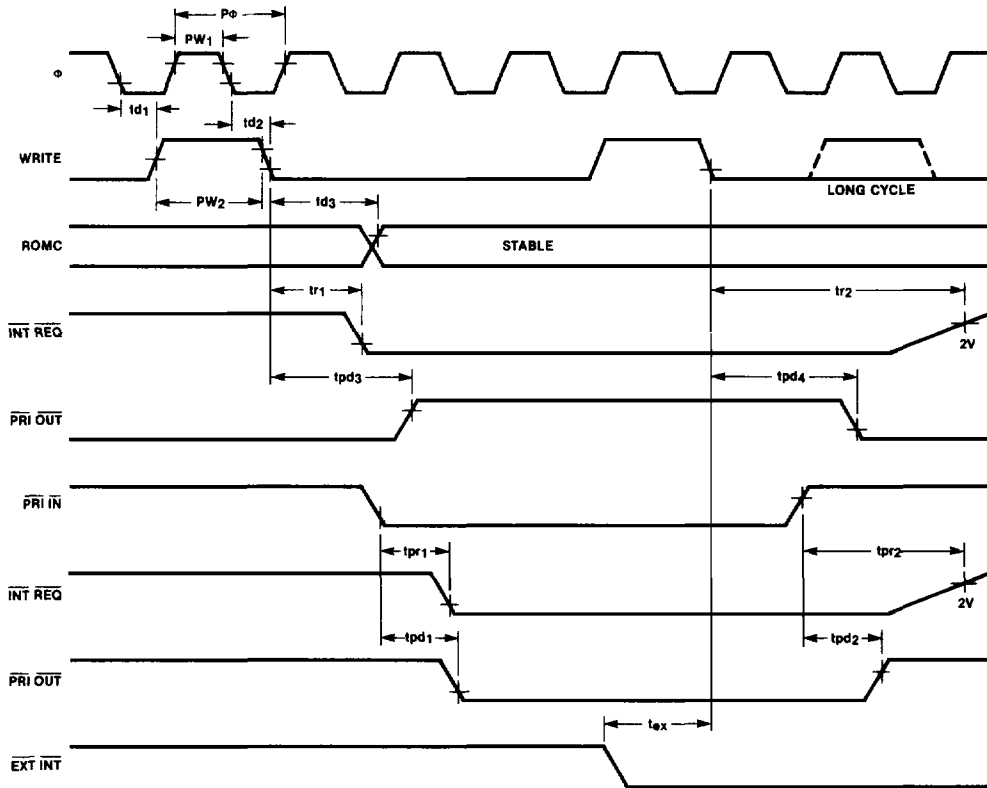
**Figure 6 F3861 Input/Output Timing**

Figure 7 F3861 Interrupt Logic Timing



3

Note: Timing measurements are made at valid logic level to valid logic level of the signals references, unless otherwise noted.

# F3861

**Table 5 F3861 Timing Characteristics**

The ac characteristics are  $V_{SS} = 0V$ ,  $V_{CC} = +5V(\pm 5\%)$ ,  
 $T_A = 0\text{ C to } +70\text{ }^\circ\text{C}$ .

Symbol	Parameter	Min	Typ	Max	Units	Test Conditions
$P\Phi$	$\Phi$ Period	0.5		10	$5\mu\text{s}$	
$PW_1$	$\Phi$ Pulse Width	180		$P\Phi - 180$	ns	$t_r, t_f = \text{ns typ.}$
$td_1$	$\Phi$ to WRITE + Delay	60		250	ns	$C_L = 100\text{ pF}$
$td_2$	$\Phi$ to WRITE - Delay	60		225	ns	$C_L = 100\text{ pF}$
$td_4$	WRITE to DB Input Delay			$2P\Phi + 1.0$	$\mu\text{s}$	
$PW_2$	WRITE Pulse Width	$P\Phi - 100$		$P\Phi$	ns	$t_r, t_f = 50\text{ ns typ.}$
$PW_S$	WRITE Period; Short		$4P\Phi$			
$PW_L$	WRITE Period; Long		$6P\Phi$			
$td_3$	WRITE to ROMC Delay			550	ns	
	WRITE to DB Output Delay					
$td_7$	WRITE to DBDR - Delay	$2P\Phi + 100 - td_2$	$2P\Phi + 200$	$2P\Phi + 850 - td_2$	ns	$C_L = 100\text{ pF}$
$td_8$	WRITE to DBDR + Delay		200		ns	Open Drain
$tr_1$	WRITE to INT REQ - Delay			430	ns	$C_L = 100\text{ pF}(1)$
$tr_2$	WRITE to INT REQ + Delay			430	ns	$C_L = 100\text{ pF}(3)$
$tp_{r1}$	PRI IN to INT REQ - Delay			240	ns	$C_L = 100\text{ pF}(2)$
$tp_{r2}$	PRI IN to INT REQ + Delay			240	ns	$C_L = 100\text{ pF}$
$tp_{d1}$	PRI IN to PRI OUT - Delay					
$tp_{d2}$	PRI IN to PRI OUT + Delay			365	ns	$C_L = 50\text{ pF}$
$tp_{d3}$	WRITE to PRI OUT + Delay			700	ns	$C_L = 50\text{ pF}$
$tp_{d4}$	WRITE to PRI OUT - Delay			640	ns	$C_L = 50\text{ pF}$
$^{**}t_{sp}$	WRITE to Output Stable			2.5	$\mu\text{s}$	$C_L = 50\text{ pF}$ Standard Pull-up
$^{*}t_{su}$	I/O Setup Time	1.3			$\mu\text{s}$	
$^{*}t_h$	I/O Hold Time	0			ns	
$^{*}t_{ex}$	EXT INT Setup Time	400			ns	

**Notes:**

1. Assume Priority In was enabled ( $\overline{\text{PRI IN}} = 0$ ) in the previous F8 cycle before the interrupt is detected in the PIO.
2. The PSU has an interrupt pending before priority in is enabled.
3. Assume the pin is tied to the INT REQ input of the F3850 CPU.
4. The starred \* parameters in the table represent those most frequently of importance when interfacing to an F8 system. Other parameters are typically those that are relevant between F8 chips and are not normally of concern to the user.
5. Input and output capacitance is 3 to 5 pF typical on all pins except  $V_{DD}$ ,  $V_{GG}$ , and  $V_{SS}$ .

**DC Characteristics**

The dc characteristics of the F3861 PIO are supplied in table 6.

**Table 6 F3861 PIO DC Characteristics**

Symbol	Parameter	Signal	Min	Max	Units	Test Conditions
$V_{IH}$ $V_{IL}$ $V_{OH}$ $V_{OL}$ $I_{IH}$ $I_{OL}$	Input High Voltage Input Low Voltage Output High Voltage Output Low Voltage Input High Current Input Low Current	Data Bus (DBO-DB7)	3.5 $V_{SS}$ 3.9 $V_{SS}$	$V_{DD}$ 0.8 $V_{DD}$ 0.4 1 -1	V V V V $\mu$ A $\mu$ A	$I_{OH} = -100 \mu$ A $I_{OL} = 1.6$ mA $V_{IN} = 6$ V, 3-State mode $V_{IN} = V_{SS}$ , 3-State mode
$V_{IH}$ $V_{IL}$ $I_L$	Input High Voltage Input Low Voltage Leakage Current	Clock Lines ( $\Phi$ ,WRITE)	4.0 $V_{SS}$	$V_{DD}$ 0.8 1	V V $\mu$ A	$V_{IN} = 6$ V
$V_{IH}$ $V_{IL}$ $I_L$	Input High Voltage Input Low Voltage Leakage Current	Priority In and Control Lines (PRI IN, ROMCO- ROMC4)	3.5 $V_{SS}$	$V_{DD}$ 0.8 1	V V $\mu$ A	$V_{IN} = 6$ V
$V_{OH}$ $V_{OL}$	Output High Voltage Output Low Voltage	Priority Out ( $\overline{PRI\ OUT}$ )	3.9 $V_{SS}$	$V_{DD}$ 0.4	V V	$I_{OH} = -100 \mu$ A $I_{OL} = 100 \mu$ A
$V_{OH}$ $V_{OL}$ $I_L$	Output High Voltage Output Low Voltage Leakage Current	Interrupt Request (INT REQ)	$V_{SS}$	0.4 1	V V $\mu$ A	Open Drain Output (1) $I_{OL} = 1$ mA $V_{IN} = 6$ V
$V_{OH}$ $V_{OL}$ $I_L$	Output High Voltage Output Low Voltage Leakage Current	Data Bus Drive ( $\overline{DBDR}$ )	$V_{SS}$	0.4 1	V V $\mu$ A	External Pull-up $I_{OL} = 2$ mA $V_{IN} = 6$ V
$V_{IH}$ $V_{IL}$ $V_{IC}$ $I_{IH}$ $I_{IL}$ $I_{IL}$	Input High Voltage Input Low Voltage Input Clamp Voltage Input High Current Input Low Voltage Input Low Current	External Interrupt (EXT INT)	3.5	1.2 15 10 -225 -500	V V V $\mu$ A $\mu$ A $\mu$ A	$I_{IH} = 185 \mu$ A $V_{IN} = V_{DD}$ $V_{IN} = 2$ V $V_{IN} = V_{SS}$
$V_{OH}$ $V_{OH}$ $V_{OL}$ $V_{IH}$ $V_{IL}$ $I_{IL}$ $I_{IL}$ $I_L$	Output High Voltage Output High Voltage Output Low Voltage Input High Voltage Input Low Voltage Leakage Current, DC/PC Leakage Current, DC/PL/DM Input Low Current	I/O Port (Standard Pull-Up)	3.9 2.9 $V_{SS}$ 2.9 $V_{SS}$	$V_{DD}$ $V_{DD}$ 0.4 $V_{DD}$ 0.8 -1.6 -2.0 -1.6	V V V V V mA mA mA	$I_{OH} = -30 \mu$ A $I_{OH} = -100 \mu$ A $I_{OL} = 2$ mA Internal Pull-up to $V_{DD}$ (3) $V_{IN} = 0.4$ V $V_{IN} = 0.4$ V $V_{IN} V_{SS}$ (4) (7)

**Notes:**

1. Pull-up resistor to  $V_{DD}$  on CPU.
2. Positive current is defined as conventional current flowing into the pin referenced.
3. Hysteresis input circuit provides additional 0.3 V noise immunity while internal/external pull-up provides TTL compatibility.
4. Measured on K a high-level I/O port OUT port.
5.  $V_{SS} = ^\circ$ V,  $V_{DD} = \pm 5$ V  $\pm 5\%$ ,  $V_{GG} = +12$ V  $\pm 5\%$ ,  $T_A = 0^\circ$ C to  $70^\circ$ C
6. Output device off.
7. -2.0 mA for extended temperature range.

# F3861

The supply currents are given in table 7.

**Table 7 Supply Currents**

Symbol	Parameter	Min	Typ	Max	Units	Test Conditions
$I_{DD}$	$V_{DD}$ Current		30	70	mA	F = 2 MHz, Outputs Unloaded
$I_{GG}$	$V_{GG}$ Current		10	18	mA	f = 2 MHz, Outputs Unloaded

### Absolute Maximum Ratings

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.

$V_{GG}$	+ 15 V, - 0.3 V
$V_{DD}$	+ 17 V, - 0.3 V
External Interrupt Input	- 600 $\mu$ A, + 225 $\mu$ A
All Other Inputs and Outputs	+ 7 V, - 0.3 V
Storage Temperature	- 55°C, + 150°C
Operating Temperature	0°C, + 70°C

**Note**  
All voltages are with respect to  $V_{SS}$ .

### Recommended Operating Ranges

Part Number	Supply Voltage ( $V_{DD}$ )			$V_{GG}$			$V_{SS}$
	Min	Typ	Max	Min	Typ	Max	
F3861	+ 4.75 V	+ 5 V	+ 5.25 V	+ 11.4 V	+ 12 V	+ 12.6 V	0 V

### Ordering Information

Part Number	Package	Temperature Range
F3861	Ceramic	C
F3861 DM	Ceramic	M
*F3861 PC	Plastic	C

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  
 \* Version A, B, C, D, and E are stocked items.