EXPERIMENT #7

PARALLEL INTERFACING USING THE PERIPHERAL INTERFACE ADAPTER (PIA)

1.0 Procedure

The purpose of this experiment is to introduce the student to the following topics:

- the Peripheral Interface Adapter IC (PIA, MC6821)
- the MC68000's Synchronous Bus Cycle
- the MC68000's Interrupt Generation Mechanism

2.0 Component Requirements

Quantity:	Description: (incl. Mfr., Mfr. Part No.)
1	Peripheral Interface Adapter (PIA); Motorola; MC6821
TBD	Misc. TTL Logic Gates (AND, OR, inverters, buffers, decoders, etc.)

3.0 Background

A. The Peripheral Interface Adapter (PIA)

The Peripheral Interface Adapter or PIA, provides a general purpose means of interfacing peripheral equipment to the MC68000 and MC6800 family of microprocessors. This integrated circuit is capable of interfacing a microprocessor to peripheral devices through two 8-bit bi-directional peripheral data buses (PA0 to PA7, and PB0 to PB7) and four control lines (CA1, CA2, CB1, and CB2).

The functional configuration of the PIA is programmable by the CPU during system initialization. Each of the sixteen peripheral data lines can be programmed to perform either as an input or an output line. Each of the peripheral control lines may be programmed to operate in one of several modes. The speed and ease of programmability provides a high degree of flexibility for the interface.

Internal Architecture

The PIA occupies four consecutive locations in the CPU's memory map. The PIA contains six internal registers, three for Port A, and three for Port B. The two Register Select lines (RS1, RS0), are used to select one of the four registers inside the PIA.

CRA2	CRB2	RS1	RS0	PIA Register Selected
0	X	0	0	Data Direction Register, Port A
1	X	0	0	Peripheral Data Register, Port A
X	X	0	1	Control Register, Port A
X	0	1	0	Data Direction Register, Port B
X	1	1	0	Peripheral Data Register, Port B
X	X	1	1	Control Register, Port B

These registers are defined as follows:

1. Data Direction Register (DDRA or DDRB)

   This register programs each of the eight peripheral data lines (PA0 to PA7 or PB0 to PB7) to act as either an input or an output. Setting a bit equal to "1" defines its corresponding peripheral data line to be an output, while setting a bit equal to "0" defines its corresponding peripheral data line to be an input.

2. Peripheral Data Register (PDRA or

   PDRB) Inputs

   The signals from the peripheral data lines are input into this register. The CPU may then read this register to determine the status of the peripheral data lines.

   Outputs

   The data written to this register by the CPU will appear on the peripheral data lines that are programmed as outputs. A "1" written into this register by the CPU causes a "high" level signal to appear on the corresponding peripheral data line. Similarly, a "0" written into this register by the CPU causes a "low" level signal to appear on the corresponding peripheral data line.

3. Control Register (CRA or CRB)

   This register allows the CPU to configure the operation of the two peripheral control lines, CA1 and CA2 or CB1 and CB2.

   This register also allows the CPU to enable the interrupt lines and monitor the status of the interrupt flags.

   Bit 2 of this register is used, along with the Register Select lines (RS1, RS0), to determine whether the Data Direction Register or the Peripheral Data Register is to be accessed.

   A summary of the format of the control word is shown in the enclosed figures. The preceding discussion is meant to serve as a brief introduction to the PIA. For more detailed information about this device refer to the following texts :

(included as ACIA&PIA.pdf in the ECE441 File Package)

a. 8-Bit Microprocessor & Peripheral Data Book.
Series C. Austin, Texas: Motorola, Inc., 1983

b. Bishop, Ron. Basic Microprocessors and the 6800.
Rochelle Park, NJ: Hayden Book Company, 1979.
(refer specifically to Chapter 9)

B. MC6800 Synchronous Bus Cycle

In order for the 68000 to interface to 6800 type peripherals, the 68000 modifies its bus cycle to meet the 6800 bus cycle timing requirements whenever a 6800 type device is selected. This feature is possible because both types of processors use memory-mapped I/O.

The 6800 interface is provided by three signals on the 68000. These signals are known as: Enable (E), Valid Peripheral Address (*VPA), and Valid Memory Address (*VMA). *VPA and *VMA are active-low signals. A description of each signal follows.

1. Enable (E)

   Enable corresponds to the E or p2 (phase 2) signal in existing 6800 systems. It is the bus clock used by 6800 peripherals to synchronize data transfers. Enable is a free-running clock that is one-tenth the frequency of the 68000's clock signal (CLK). It has a 60/40 duty cycle, which means that it is low for six of 68000 CLK cycles, and high for four of the 68000 CLK cycles. This extended duty cycle allows the 68000 to perform consecutive 6800 accesses.

2. Valid Peripheral Address (*VPA)

   This input signal informs the 68000 that the address on the bus is the address of a 6800 type device and that the bus should conform to the E clock transfer characteristics of the 6800 bus. VPA is derived by decoding the address bus, and qualifying it with Address Strobe (*AS).

3. Valid Memory Address (*VMA)

   This output signal notifies any 6800 peripheral that the address on the address bus is valid, and that the 68000 is synchronized to the Enable (E) signal. The *VMA signal is used as part of the chip select circuitry for the peripheral. This ensures that 6800 type peripherals are selected and unselected at the proper time. The peripheral now runs its cycle during the high portion of the E signal.

   For more information about the 68000's Synchronous Bus Cycle, refer to the chapter entitled "Signal and Bus Operation Description" in Motorola's MC68000 8-/16-/32-Bit Microprocessors User's Manual.

C. Interrupts

Refer to the Background Section of Experiment #6.

4.0 Statement of Problem

In this experiment the student will design and implement the hardware to interface the PIA to the SANPER-1 Educational Lab Unit. The student will also write software routines to write data to one port of the PIA and read it back on another port.

5.0 Preliminary Assignment

1. Draw a detailed schematic diagram of the hardware required to interface the PIA to the SANPER-1 Educational Lab Unit.

   The PIA must reside within the 64K address range of $050000 to $05FFFF. However, not all of this range is available. The following addresses are reserved for other hypothetical devices in the system.

   Device 1: $050000 - $050FFF
   Device 2: $053000 - $05407F
   Device 3: $055000 - $05603F
   Device 4: $059000 - $05A3FF
   Device 5: $05B000 - $05C0BF
   Device 6: $05E000 - $05FF00

   The PIA may be placed in any non-occupied address locations within the 64K range. The user must decode around the above existing devices so that no other device is unintentionally selected. The user must initially use full address decoding to enter within the 64K address space at $50000. The user may then employ partial decoding and use a minimal number of address lines to select the PIA.

   HINT: Use one of the SANPER-1 ELU's Block Select lines as part of your address decoding implementation.

   The PIA's Host or CPU interface must include all the signals shown on the left side of the PIA in Figure 7.1. Each of these signals should be connected to either the MC68000 or the address decoding logic. All unused inputs should be tied to +5V DC through a resistor. Connect the PIA *IRQ line(s) to the Interrupt Request Level 1 (*IRQ1) signal on the System Expansion Board of the SANPER-1 Educational Lab Unit.
   The PIA's peripheral interface lines should be connected as shown on the right side of the PIA in Figure 7.1.

2. Build the hardware from Prelim #1 on breadboard strips.

3. Write an initialization routine to configure the PIA for the following conditions:

   - Port A lines (PA0 to PA7) are inputs
   - Port B lines (PB0 to PB7) are outputs
   - IRQA interrupt enabled and asserted by a high-to-low transition on CA1
   - Port B uses pulse-mode handshaking

4. Write an Interrupt Service Routine (ISR) to read the peripheral data on PA0 to PA7 of Port A of the PIA whenever there is a high-to-low transition on CA1. Display the received character on the terminal.

   Caution: Your ISR should not modify any registers unless they are first saved to the stack. Restore the registers prior to exiting the ISR.

5. Write a program to continuously read a character from the keyboard and output it to the peripheral data side of Port B of the PIA.

6. Write the main program, which executes the initialization subroutine of Prelim #2, and then continuously executes the program of Prelim #5.

6.0 Procedure

1. Connect your hardware to the Breadboard Strip on the front panel of the SANPER-1 Educational Lab Unit.

2. Initialize the exception vector for the PIA interrupt request to point to the beginning of the Interrupt Service Routine from Prelim #4.

3. Run the program from Prelim #6.

4. Press a key on the keyboard and record what happens at the terminal. You must be able to explain the data that appears on the terminal.
   If no data is being displayed on the terminal, you have a problem, and you must debug your hardware and software.
   If data is being displayed on the terminal, but you do not understand what is happening, carefully review the functions of each of the preliminary assignments.

5. Demonstrate to your Lab Instructor that your hardware and software are working properly.

7.0 Discussion

Submit the following to your Lab Instructor as a Final Report:

1. A fully commented listing of all programs and subroutines (they must include both global and local comments).

2. A schematic diagram of your hardware design.

3. Describe, in detail, the function of each of the programs from the Preliminary Assignment.

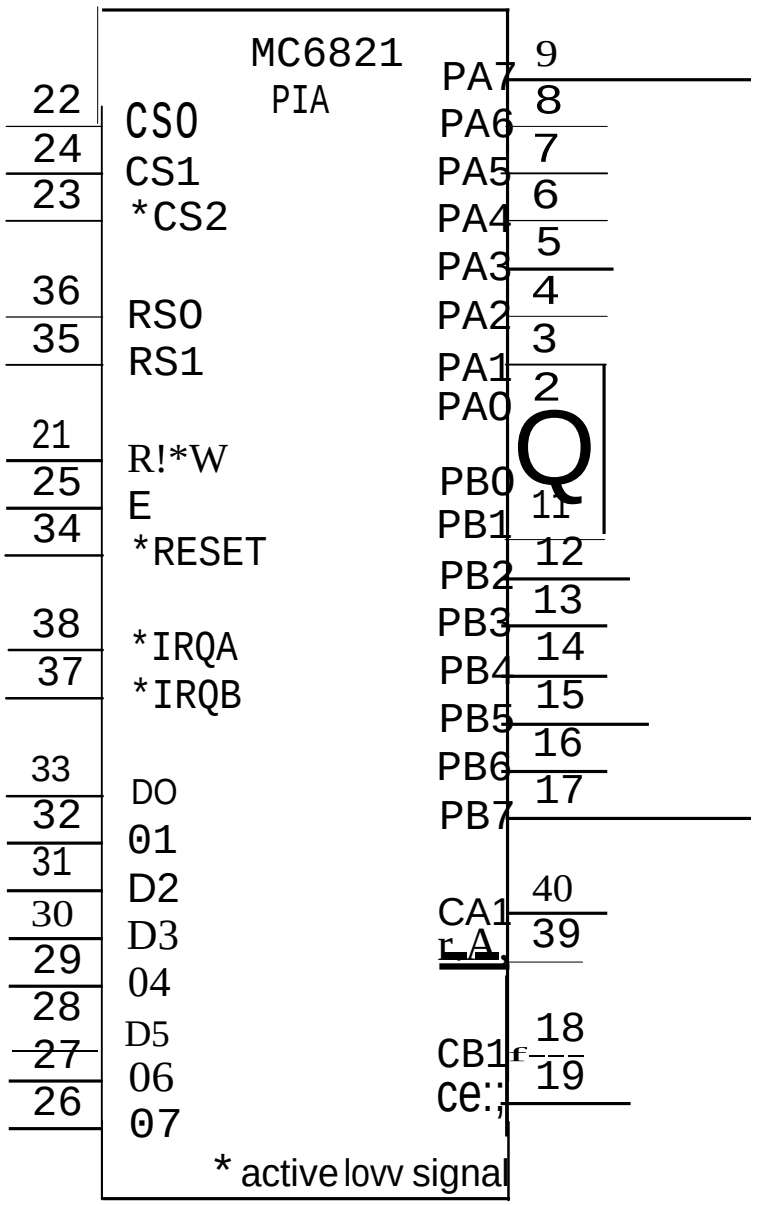
4. Describe what type of interrupt acknowledge method (user or auto vectored) was used for the PIA in this experiment and why?

5. Describe how a microprocessor determines which side of the PIA generated an interrupt request?

6. If PIA Control Register A reads $3F, how has the PIA been configured?

7. Discuss three possible applications of the PIA.

8. Draw and discuss the timing diagram for a read operation using a MC68000 Synchronous Bus Cycle. Indicate the order in which the signals are asserted (and negated) and the timing relationships between them.



Host Interface

Peripheral Interface

FIGURE 7.1 Connection Drawing of PIA

Summary of CA1 Control

Transition of interrupt input line CA1	*Status of bit 1 in CRA (edge)*	*Status of bit 0 in CRA (mask)*	*IRQA1 (interrupt flag) Bit 7 ofCRA*	*Status of IRQA line (MPU interrupt request)*
L_	0	0		Masked (remains high)
L_	0			Goes low (processor interrupted)
_r		0		Masked (remains high)
_r				Goes low (processor interrupted)
L_			0	Remains high
_r	0		0	Remains High

Summary of CB1 Control

Transttion of interrupt input line CB1	*Status of bit 1 in CRB (edge)*	*Status of bit 0 in CRB (mask)*	*IRQB1 (interrupt flag) Bit 7ofCRB*	*Status of !ROB Line (MPU interrupt request)*
L_	0	0		Masked (remains high)
L_	0			Goes low (processor interrupted)
_r		0		Masked (remains high)
_r				Goes low (processor interrupted)
L_			0	Remains high
_r	0		0	Remains high

Summary of CA2 Control

Transition of input CA2	*Status of bit 5 in CRA (1/0 control)*	*Status of bit 4 in CRA (edge)*	*Status of bit 3 in CRA (Mask)*	*!RQA2 (interrupt flag) Bit6 ofCRA*	*Status of IROA line (MPU interrupt request)*
L.	0	0	0		Masked (remains high)
L.	0	0			Goes low (processor interrupted)
_r	0		0		Masked (remains high)
_r	0				Goes low (processor interrupted)
L.	0			0	Remains high
_r	0	0		0	Remains high

Summary of C82 Control

Trans1tion of input CB2	*Status of bit 5 in CRB (1/0 control*	*Status of bit 4 in CRB (edge)*	*Status of btl 3 in CRB (mask)*	*IROB2 (interrupt flag) Bit6 ofCRB*	*Status of !ROB line (MPU interrupt request)*
L.	0	0	0		Masked (remains high)
L.	0	0			Goes low (processor interrupted)
_r	0		0		Masked (remains high)
__j	0				Goes low (processor interrupted)
L.	0			0	Remains high
__j	0	0		0	Remams high

OES HIGH ON TRANSinON OF CAi INTERRUPT IROAI SET TO "1"1

GOES LOW WHEN DATAONA SIDE HAS BEEN READ BY MPU DURING FALUNG EDGE OF ENABLE SIGNAL (12) (LOAD A FROM PIA)

CA2LINE NORMAUY LOW IN-. THIS MODE

ENABLE SIGNAL (121

TIME I

PERIPHERAL SAYS, "HELLO MPU, I HAVE SOME DATA FOR YOU."

PERIPHERAL PtA

PERIPHERAL PIA

IRQAI="I"

CA2

\MPU SAYS, "I GOT YOUR MESSAGE"

TIME3

PERIPHERAL PIA

IRQAI•"O" (UPON MPU READ)

CA2

\MPU SAYS, "OK PERIPHERAL I GOT YOUR INPUT 'DATA, I'M READY FOR MORE."

Handshake mode

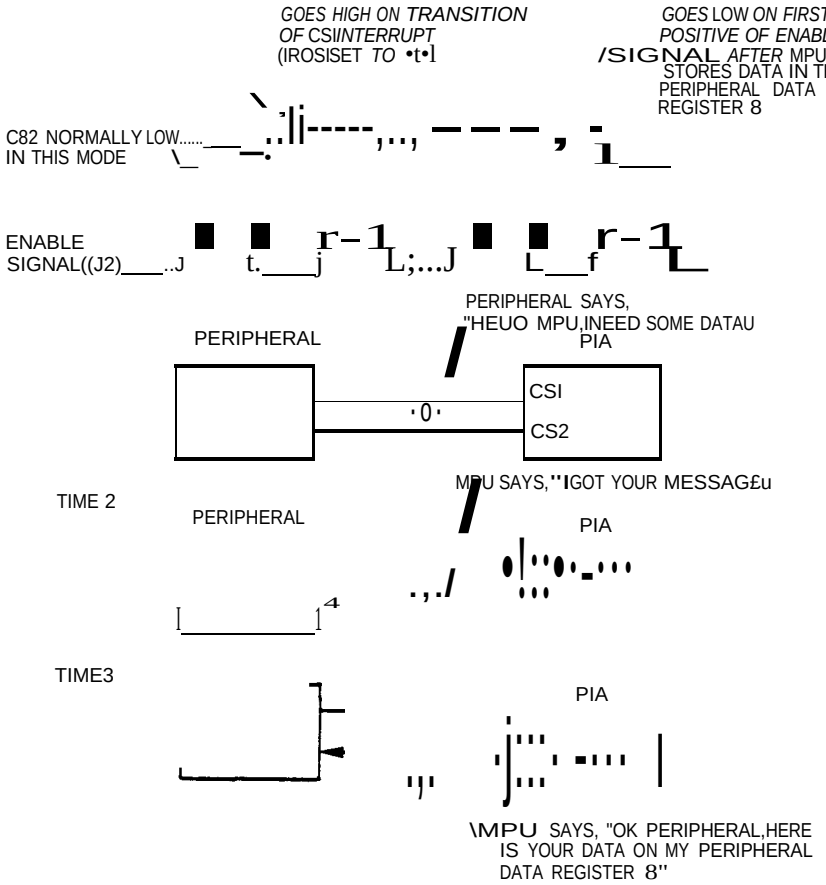
CA2 NORMALLY HIGH IN THIS MODE

GOES LOW AFTER A "READ A SIDE DATA" INSTRUCTION

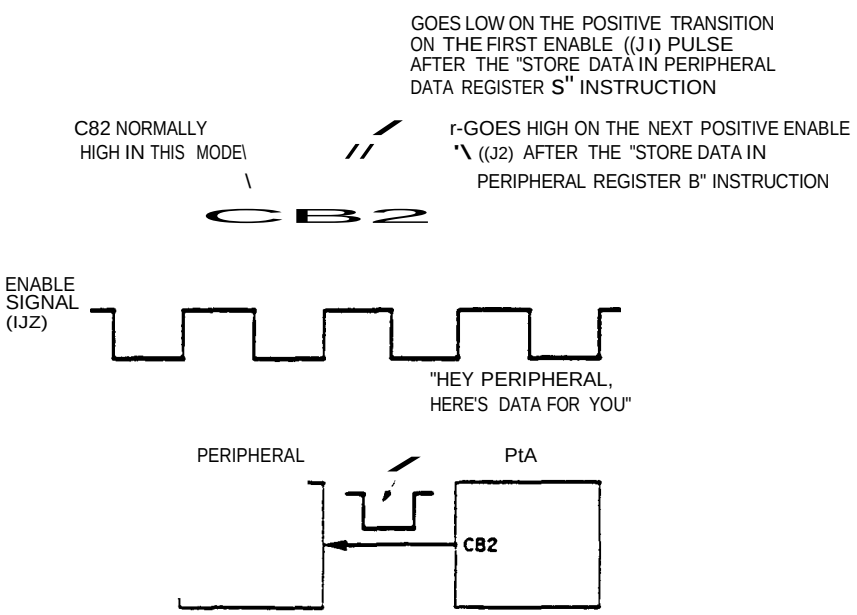
GOES HIGH ON THE NEGATIVE EDGE OF THE NEXT E IGNAL AFTER THE 'READ A SIDE DATA INSTRUCTION"

ENABLE SIGNAL (12)

!IPUT SAYS, HEY PERIPHERAL, P£R1PHERAL -&.rust ....IOUR DATA."



Handshake mode



MC6821

PERIPHERAL INTERFACE ADAPTER (PIA)

The MC6821 Peripheral Interface Adapter provides the universal means of interfacing peripheral equipment to the M6800 family of microprocessors. This device is capable of interfacing the MPU to peripherals through two 8-bit bidirectional peripheral data buses and four control lines. No external logic is required for interfacing to most peripheral devices.

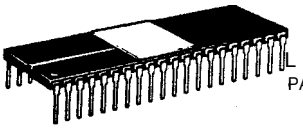
The functional configuration of the PIA is programmed by the MPU during system initialization. Each of the peripheral data lines can be pro- grammed to act as an input or output, and each of the four con- trol/interrupt lines may be programmed for one of several control modes. This allows a high degree of flexibility in the overall operation of the interface.

- 8-Bit Bidirectional Data Bus for Communication with the MPU
- Two Bidirectional 8-Bit Buses for Interface to Peripherals
- Two Programmable Control Registers
- Two Programmable Data Direction Registers
- Four Individually-Controlled Interrupt Input Lines; Two Usable as Peripheral Control Outputs
- Handshake Control Logic for Input and Output Peripheral Operation
- High-Impedance Three-State and Direct Transistor Drive Peripheral Lines
- Program Controlled Interrupt and Interrupt Disable Capability
- CMOS Drive Capability on Side A Peripheral Lines
- Two TTL Drive Capability on All A and B Side Buffers
- TTL-Compatible
- Static Operation

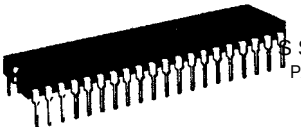
MOS

IN-CHANNEL, SILICON-GATE, DEPLETION LOAD)

PERIPHERAL INTERFACE ADAPTER



L SUFFIX CERAMIC PACKAGE CASE 715

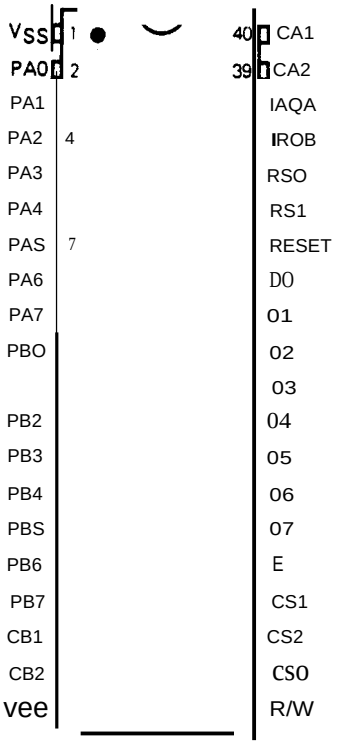


S SUFFIX CERDIP PACKAGE CASE 734

:LASTIC PAC"GE
CASE 711

3

PIN ASSIGNMENT



ORDERING INFORMATION

Package Type	Frequency IMHzl	Temperature	Order Number
Ceramic L Suffix	1.0	00C to 70°C	MC6821L
	1.0	– 40°C tO 85°C	MC6821CL
	1.5	00C to 70°C	MC68A21L
	1.5	-4Q°C to 85°C	MC68A21CL
	2.0	00C to 70°C	MC68821L
Cerdip S Suffix	1.0	00C to 70°C	MC6821S
	1.0	-40°C to B5°C	MC6821CS
	1.5	00C to 70°C	MC68A21S
	1.5	-40°C tO B5°C	MC68A21CS
	20	00C to 70°C	MC68821S
Plastic P Suffix	1.0	0°C to 70°C	MC6821P
	1.0	– 40°C to B5°C	MC6821CP
	1.5	0°C. to 70°C	MC68A21P
	1.5	-40°C tO B5°C	MC68A21CP
	2.0	0°C to 70°C	MC68821P

./

MAXIMUM RATINGS

Characteristics	Symbol	Value	Unit
Supply Voltage	vee	-0.3 to +7.0	v
Input Voltage	Vin	-0.3 to + 7.0	v
Operating Temperature Range MC6821. MC68A21, MC68B21 MC6821C.' MC68A21C	TA	TL to TH 0 to 70 -40 to +85	oc
Storage Temperature Range	Tstg	-55 to+ 150	oc

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Ceramic Plastic Cerdip	8JA	 50 100 60	*"CIW*

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid applications of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that Vin and Vout be constrained to the range GND:s{Vin or VoutlsVcc.

Unused inputs must always be tied to an appropriate logic voltage level {e.g., either GND or Vccl.

II

POWER CONSIDERATIONS

The average chip-junction temperature, TJ, in °C can be obtained from:

TJ=TA+(PD•8JA) (1)

Where:

TA ━Ambient Temperature, °C

8JA•Package Thermal Resistance, Junction-to-Ambient, °C/W

Po•PiNT+ PPORT

PiNT•Icc x Vee. Watts — Chip Internal Power

PPORT•Port Power Dissipation, Watts — User Determined

For most applications PPORT<C PiNT and can be neglected. PpQRT may become significant if the device is configured to drive Darlington bases or sink LED loads.

An approximate relationship between Po and TJ (if PPORT is neglected) is:

Po=K+(TJ+273°C) (2)

Solving equations 1 and 2 for K gives:

K=PD•(TA+273°Cl+8JA•Po2 (3)

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring Po (at equilibrium) for a known TA. Using this value of K the values of Po and TJ can be obtained by solving equations (1) and (2) iteratively for any value ofTA·

DC ELECTRICAL CHARACTERISTICS 1Vcc=5.0 Vdc ±5%, Vss=O, TA= TL to TH unless otherwise noted!.

Characteristic	Symbol	Min	Typ	Max	Unit
BUS CONTROL INPUTS (R/W, Enable, RESET, RSO, RS1, CSO, CS1, CS2)					
Input High Voltage	V1H	vss+-z.o	-	vee	V
Input Low Voltage	VIL	Vss-o.3	-	Vss+O.B	v
Input Leakage Current IV;n= 0 to 5.25 Vl	lin	-	1.0	2.5	,.A
Capacitance {V·n=O, TA=25"C. f=1.0 MHz)	Cin	-	-	7.5	pF
INTERRUPT OUTPUTS IIRQA, IRQB)					
Output Low Voltage ULoad-1.6 mAl	VOL	-	-	vss+0.4	v
Hi-Z Output Leakage Current	Ioz	-	1.0	10	,.A
Capacitance IVin=0, TA = 25"C, f= 1.0 MHz)	Cout	-	-	5.0	pF
DATA BUS (00..07)					
Input High Voltage	VIH	Vss+2.o	-	vee	v
Input Low Voltage	VIL	vss-0.3	-	Vss+O.S	v
Hi-Z Input Leakage Current 1Vin=0.4 to 2.4 Vl	liz	-	2.0	10	,.A
Output High Voltage ULoad = -205 ,.Al	VOH	Vss+2.4	-	-	v
Output Low Voltage IILoad= 1.6 mAl	VOL	-	-	vss + 0 4	v
Capacitance IVin= 0, TA= 25"C, f= 1.0 MHzl	Cin	-	-	12.5	pF

DC ELECTRICAL CHARACTERISTICS (Continued)

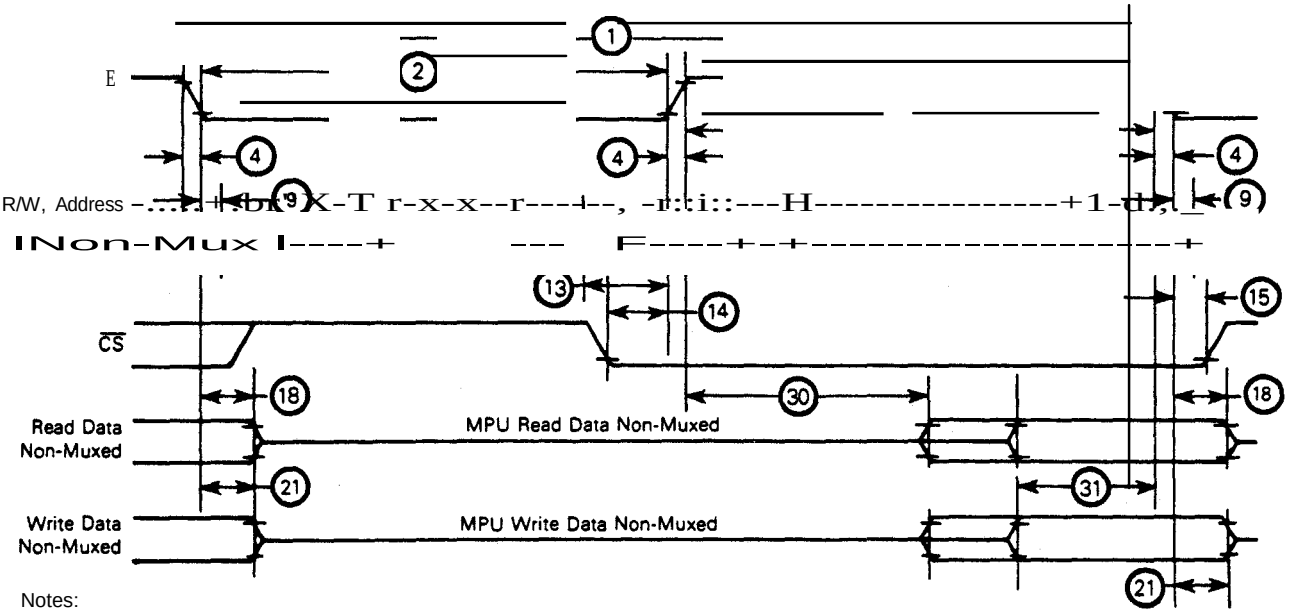
Characteristic		Symbol	Min	Typ	Max	Unit
PERIPHERAL BUS (PAO-PA7, PBO-PB7, CA1, CA2, CB1, CB2)						
Input Leakage Current (Vin = 0 to 5.25 V)	R/W, RESET, RSO, RS1, CSO. CS1, CS2, CA1. CB1, Enable	Iin	–	1.0	2.5	p.A
Hi-Z Input Leakage Current (Vin= 0.4 to 2.4 V)	PBO-PB7, CB2	IiZ	–	2.0	10	p.A
Input High Current (VIH = 2.4 V)	PAO-PA7, CA2	IIH	-200	-400	–	p.A
Darlington Drive Current (Vo = 1.5 V)	PBO-PB7, CB2	IOH	-1.0	–	-10	mA
Input Low Current (VIL = 0.4 V)	PAO-PA7, CA2	IIL	–	-1.3	-2.4	mA
Output High Voltage (ILoad= -200p.A)	PAO-PA7, PBO-PB7, CA2, CB2	VOH	Vss+2.4	–	–	V
(ILoad= – 10p.A)	PAO-PA7, CA2		vcc-1.0	–	–	
Output Low Voltage (ILoad=3.2 mA)		VoL			vss+u.4	V
Capacitance (Vin =0, TA = 25°C, f= 1.0 MHz)		Cin	-	–	10	pF
POWER REQUIREMENTS						
Internal Power Dissipation (Measured at h = 0°C)		PINT	–	–	550	mW

BUS TIMING CHARACTERISTICS (See Notes 1 and 2)

Ident. Number	Characteristic	Symbol	MC6821		MC68A21		MC68B21		Unit
			Min	Max	Min	Max	Min	Max	
1	Cycle Time·	lcyc	1.0	10	0.67	10	0.5	10	p.S
2	Pulse Width, E Low	PWEL	430	–	28)	–	210	–	ns
3	Pulse Width, E High	PWeH	450	–	28)	–	220	–	ns
4	Clock Rise and Fall Time	tr. tf	–	25	–	25	–	20	ns
9	Address Hold Time	lAH	10	–	10	–	10	–	ns
13	Address Setup Time Before E	lAS	80	–	60	–	40	–	ns
14	Chip Select Setup Time Before E	tcs	80	–	60	–	40	–	ns
15	Chip Select Hold Time	tcH	10	–	10	–	10	–	ns
18	Read Data Hold Time	lOHR	20	50•	20	50•	20	50•	ns
21	Write Data Hold Time	lDHW	10	–	10	–	10	–	ns
30	Output Data Delay Time	tDoR	–	290	–	180	–	150	ns
31	Input Data Setup Time	tosw	165	–	80	–	60	–	ns

•The data bus output buffers are no longer sourcing or sinking C\,Jrrent by tDHRmax (High Impedance).

FIGURE 1 — BUS TIMING



Notes:
1. Voltage levels shown are VLs0.4 V. VH2:2.4 V. unless otherwise specifi .
2. Measurement points shown are 0.8 V and 2.0 V, unless otherwise specified.

PERIPHERAL TIMING CHARACTERISTICS (Vcc=5.0 V ±5%, Vss=O V, TA=TL to TH unless otherwise specified)

Characteristic	Symbol	MC6821		MC68A21		MC68B21		Unit	Reference Fig. No.
		Min	Max	Min	Max	Min	Max		
Data Setup Time	tpos	200	–	135	–	100	–	ns	6
Data Hold Time	tpoH	0	–	0	–	0	–	ns	6
Delay Time, Enable Negative Transition to CA2 Negative Transition	ICA2	–	1.0	–	0.670	–	0.500	µs	3, 7, B
Delay Time, Enable Negative Transition to CA2 Positive Transition	TRS1	–	1.0	–	0.670	–	0.500	µs	3, 7
Rise and Fall Times for CA1 and CA2 Input Signals	tr. If	–	1.0	–	1.0	–	1.0	µs	B
Delay Time from CA1 Active Transition to CA2 Positive Transition	IRS2	–	2.0	–	1.35	–	1.0	µs	3,8
Delay Time, Enable Negative Transition to Data Valid	tpow	–	1.0	–	0.670	–	0.5	µs	3, 9, 10
Delay Time, Enable Negative Transition to CMOS Data Valid PAO-PA7, CA2	!CMOS	–	2.0	–	1.35	–	1.0	µs	4, 9
Delay Time, Enable Positive Transition to CB2 Negative Transition	tcs2	–	1.0	–	0.670	–	0.5	µs	3, 11, 12
Delay Time, Data Valid to CB2 Negative Transition	toe	20	–	20	–	20	–	ns	3, 10
Delay Time, Enable Positive Transition to CB2 Positive Transition	tRS1	–	1.0	–	0.670	–	0.5	µs	3, 11
Control Output Pulse Width, CA2/CB2	PWcT	500	–	375	–	250	–	ns	3, 11
Rise and Fall Time for CB1 and CB2 Input Signals	tr. If	–	1.0	–	1.0	–	1.0		12
Delay Time, CB1 Active Transition to CB2 Positive Transition	tRS2	–	2.0	–	1.35	–	1.0	µs	3, 12
Interrupt Release Time, IROA and mLiB	l!R	–	1.60	–	1.10	–	0.85	µs	5, 14
Interrupt Response Time	IRS3	–	1.0	–	1.0	–	1.0	µs	5, 13
Interrupt Input Pulse Time	Pw1	500	–	500	–	500	–	ns	13
RESET Low Time•	tRL	1.0	–	0.66	–	0.5	–	µs	15

•The RESET line must be high a minimum of 1.0 !'s before addressing the PIA.

FIGURE 2 — BUS TIMING TEST LOADS

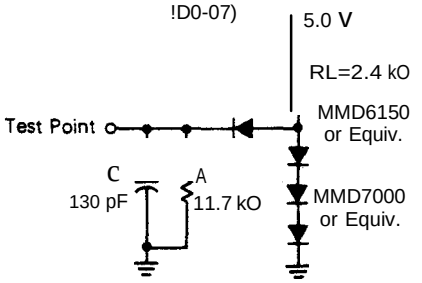


FIGURE 3 — TIL EQUIVALENT TEST LOAD

(PAO-PA7, PBO-PB7, CA2, CB21

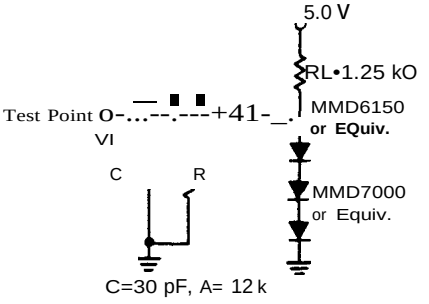


FIGURE 4– CMOS EQUIVALENT TEST LOAD

(PAO-PA7, CA21

FIGURE 5 — NMOS EQUIVALENT TEST LOAD

(iROOnly)

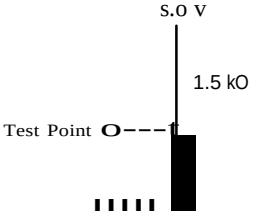


FIGURE 6– PERIPHERAL DATA SETUP AND HOLD TIMES
!Read Model

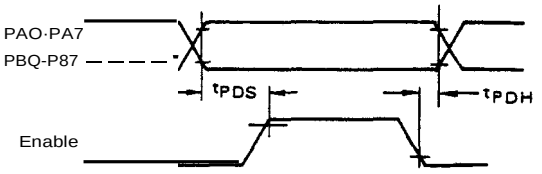
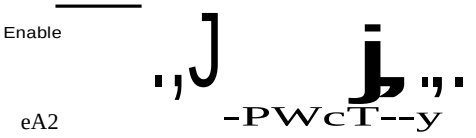


FIGURE 7– CA2 DELAY TIME
!Read Mode; CRA-5= CRA3= 1, CRA-4=0)



• Assumes part was deselected during the previous E pulse.

FIGURE 8– CA2 DELAY TIME
!Read Mode; CRA-5=1, CRA-3=CRA-4=0I

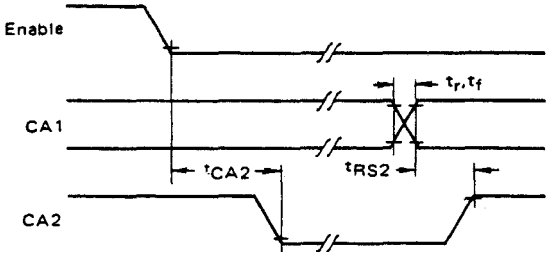
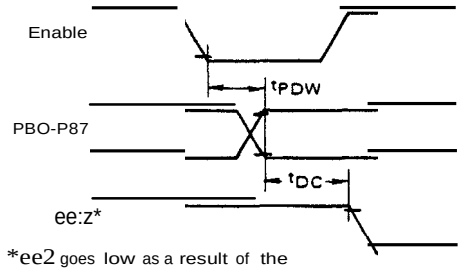


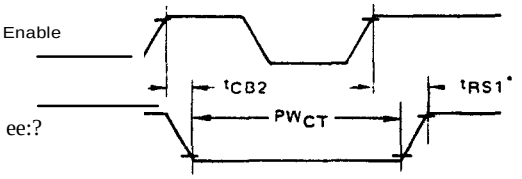
FIGURE 9– PERIPHERAL CMOS DATA DELAY TIMES
(Write Mode; CRA-5= CRA-3= 1, CRA-4=0)

FIGURE 10– PERIPHERAL DATA AND CB2 DELAY TIMES
(Write Mode; CRB-5= CRB-3= 1, CRB-4= 0J



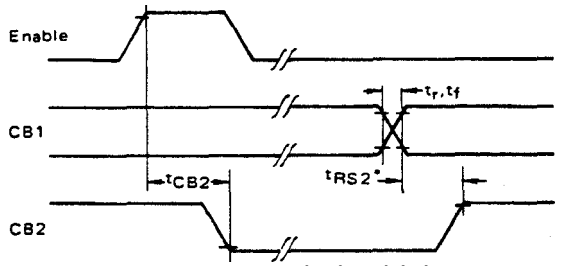
*ee2 goes low as a result of the positive transition of Enable.

FIGURE 11 – CB2 DELAY TIME (Write Mode; CRB-5= CRB-3= 1, CRB-4= 0)



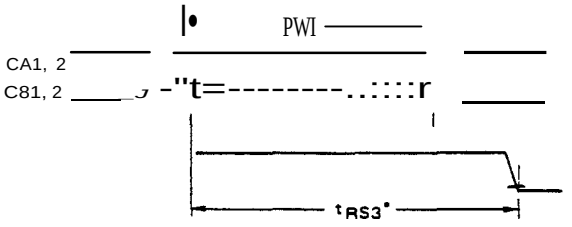
• Assumes part was deselected during the JJrevious E pulse

FIGURE 12– CB2 DELAY TIME
(Write Mode; CRB-5= 1, CRB·3=CRB-4=0I



• Assumes part was deselected during any previous E pulse.

FIGURE 13 – INTERRUPT PULSE WIDTH AND iRQ RESPONSE



• Assumes Interrupt Enable Bits are set.

Note: Timing measurements are referenced to and from a low voltage of 0.8 volts and a high voltage of 2.0 volts. unless otherwise noted.

FIGURE 14 -IRQ RELEASE TIME

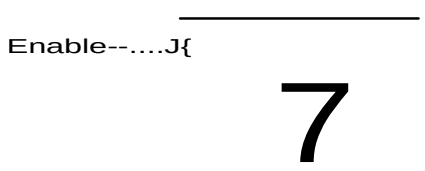
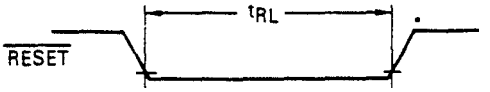
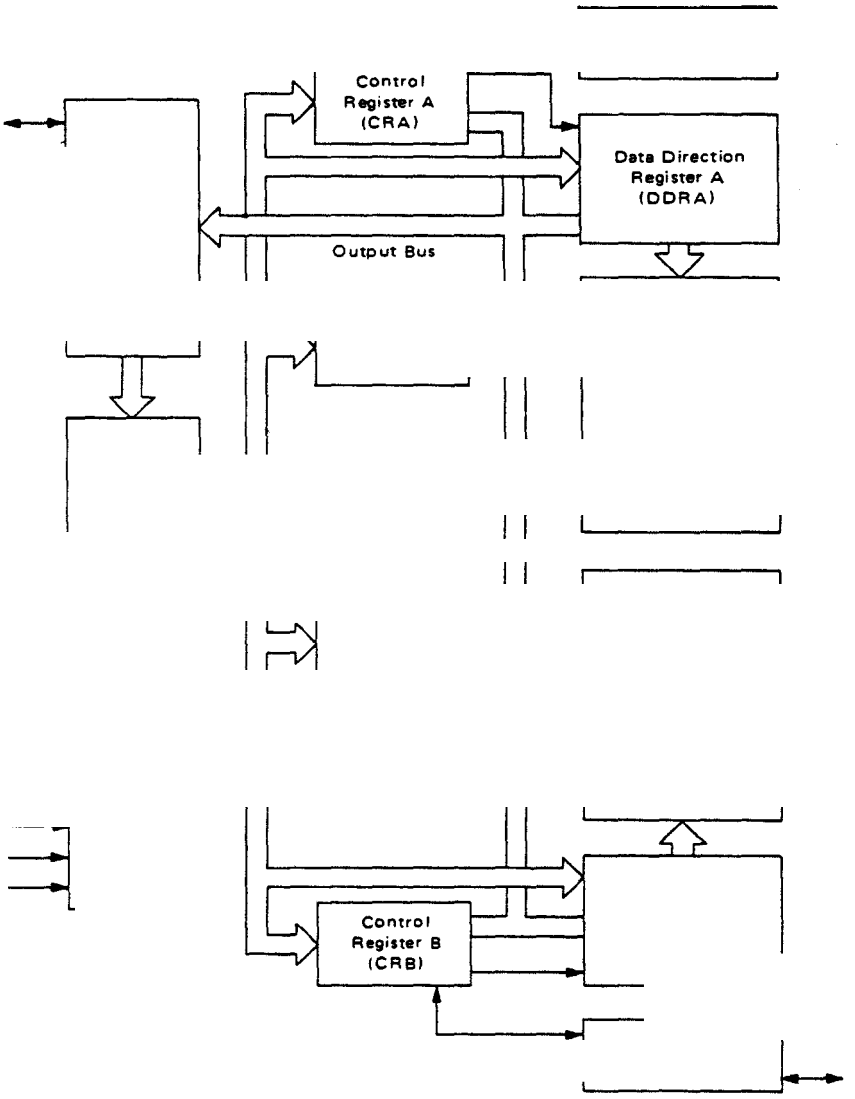


FIGURE 15 — RESET L.OW TIME



·The RESET line must be a VIH for a minimum of 1.0 JI.S before addressing the PIA.

Note: Timing measurements are referenced to and from a low voltage of 0.8 volts and a high voltage of 2.0 volts, unless otherwise noted.



PIA INTERFACE SIGNALS FOR MPU

The PIA interfaces to the M6800 bus with an 8-bit bidirectional data bus, three chip select lines, two register select lines, two interrupt request lines, a read/write line, an enable line and a reset line. To ensure proper operation with the MC6800, MC6802, or MC6808 microprocessors, VMA should be used as an active part of the address decoding.

Bidirectional Data IDO-D7) – The bidirectional data lines ID0-071 allow the transfer of data between the MPU and the PIA. The data bus output drivers' are three-state devices that remain in the high-impedance loffl state except when the MPU performs a PIA read operation. The read/write line is in the read (high) state when the PIA is selected for a read operation.

Enable (E) – The enable pulse, E, is the only timing signal that is supplied to the PIA. Timing of all other signals is referenced to the leading and trailing edges of theE pulse.

Read/Write IR/Wl – This signal is generated by the MPU to control the direction of data transfers on the data bus. A low state on the PIA read/write line enables the input buffers and data is transferred from the MPU to the PIA on the E signal if the device has been selected. A high on the read/write line sets up the PIA for a transfer of data to the bus. The PIA output buffers are enabled when the proper address and the enable pulse E are present.

RESET – The active low RISEi' line is used to reset all register bits in the PIA to a logical zero llowl. This line can be used as a power-on reset and as a master reset during system operation.

Chip Selects ICSO, CS1, and **ffi1** – These three input signals are used to select the PIA. CSO and CS1 must be high and CS2 must be low for selection of the device. Data transfers are then performed under the control of the enable and read/write signals. The chip select lines must be stable for the duration of the E pulse. The device is deselected when any of the chip selects are in the inactive state.

Register Selects IRSO and RS1) – The two register select lines are used to select the various registers inside the PIA. These two lines are used in conjunction with internal Control Registers to select a particular register that is to be written or read.

The register and chip select lines should be stable for the duration of the E pulse while in the read or write cycle.

Interrupt Request (IRQA and 'iRCi81 – The active low Interrupt Request lines (i'R'QA and IRQBI act to interrupt the MPU either directly or through interrupt priority circuitry. These lines are "open drain" (no load device on the chip). This permits all interrupt request lines to be tied together in a wire-OR configuration.

Each Interrupt Request line has two internal interrupt flag bits that can cause the Interrupt Request line to go low. Each flag bit is associated with a particular peripheral interrupt line. Also,four interrupt enable bits are provided in the PIA which may be used to inhibit a particular interrupt from a peripheral device.

Servicing an interrupt by the MPU may be accomplished by a software routine that, on a prioritized basis, sequentially reads and tests the two control registers in each PIA for interrupt flag bits that are set.

The interrupt flags are cleared (zeroed) as a result of an MPU Read Peripheral Data Operation of the corresponding data register. After being cleared, the interrupt flag bit cannot be enabled to be set until the PIA is deselected during an E pulse. The E pulse is used to condition the interrupt control lines ICA1, CA2, C81, CB21. When these lines are used as interrupt inputs, at least one E pulse must occur from the inactive edge to the active edge of the interrupt input signal to condition the edge sense network. If the interrupt flag has been enabled and the edge sense circuit has been properly conditioned, the interrupt flag will be set on the next active transition of the interrupt input pin.

PIA PERIPHERAL INTERFACE LINES

The PIA provides two 8-bit bidirectional data buses and four interrupt/control lines for interfacing to peripheral devices.

Section A Peripheral Data IPAO-PA7) – Each of the peripheral data lines can be programmed to act as an input or output. This is accomplished by setting a "1" in the corresponding Data Direction Register bit for those lines which are to be outputs. A "0" in a bit of the Data Direction Register causes the corresponding peripheral data line to act as an input. During an MPU Read Peripheral Data Operation, the data on peripheral lines programmed to act as inputs appears directly on the corresponding MPU Data Bus lines. In the input mode, the internal pullup resistor on these lines represents a maximum of 1.5 standard TTL loads.

The data in Output Register A will appear on the data lines that are programmed to be outputs. A logical "1" written into the register will cause a "high" on the corresponding data line while a "0" results in a "low." Data in Output Register A may be read by an MPU "Read Peripheral Data A" operation when the corresponding lines are programmed as outputs. This data will be read properly if the voltage on the peripheral data lines is greater than 2.0 volts for a logic **'T'** output•and less than 0.8 volt for a logic "0" output. Loading the outp'ut lines such that the voltage on these lines does not reach full voltage causes the data transferred into the MPU on a Read operation to differ from that contained in the respective bit of Output Register A.

Section 8 Peripheral Data IPBO-P87) – The peripheral data lines in the 8 Section of the PIA can be programmed to act as either inputs or outputs in a similar manner to PAO-PA7. They have three-state capabiity, allowing them to enter a high-impedance state when the peripheral data line is used as an input. In addition, data on the peripheral data lines

PBO-PB7 will be read properly from those lines programmed as outputs even if the voltages are below 2.0 volts for a "high" or above 0.8 V for a "low". As outputs, these lines are compatible with standard TTL and may also be used as a source of at least 1 milliampere at 1.5 volts to directly drive the base of a transistor switch.

Interrupt Input (CA1 and CB11 — Peripheral input lines CA1 and CB1 are input only lines that set the interrupt flags of the control registers..The active transition for these signals is also programmed by the two control registers.

Peripheral Control (CA21 — The peripheral control line CA2 can be programmed to act as an interrupt input or as a peripheral control output. As an output, this line is compatible with standard TTL; as an input the internal pullup resistor on this line represents 1.5 standard TTL loads. The function of this signal line is programmed with Control Register A.

Peripheral Control (CB2l — Peripheral Control line CB2 may also be programmed to act as an interrupt input or peripheral control output. As an input, this line has high input impedance and is compatible with standard TTL. As an output it is compatible with standard TTL and may also be u ed as a source of up to 1 milliampere at 1.5 volts to directly drive the base of a transistor switch. This line is programmed by Control Register B.

INTERNAL CONTROLS

INITIALIZATION

A RESET has the effect of zeroing all PIA registers. This will set PAO-PA7, PBO-PB7, CA2 and CB2 as inputs, and all interrupts disabled. The PIA must be configured during the restart program which follows the reset.

There are six locations within the PIA accessible to the MPU data bus: two Peripheral Registers, two Data Direction Registers, and two Control Registers. Selection of these locations is controlled by the RSO and RS1 inputs together with bit 2 in the Control Register, as shown in Table 1.

Details of possible configurations of the Data Direction and Control Register are as follows:

TABLE 1 -INTERNAL ADDRESSING

RS1	RSO	Control Reg1ster B1t CRA·2	CRB·2	Locauon Selected
0	0	1	X	Peripheral Reg1ster A
0	0	0	X	Data Direct10n Register A
0	1	X	X	Control Register A
1	0	X	1	Peripheral Register B
1	0	X	0	Data D"ection Register B
1	1	X	X	Control Register B

X • Don•t Care

PORT A-8 HARDWARE CHARACTERISTICS

As shown in Figure 17, the MC6821 has a pair of 1/0 ports whose characteristics differ greatly. The A side is designed to drive CMOS logic to normal30% to 70% levels, and incorporates an internal pullup device that remains connected even in the input mode. Because of this, the A side requires more drive current in the input mode than Port B. In contrast, the B side uses a normal three-state NMOS buffer which cannot pullup to CMOS levels without external resistors. The B side can drive extra loads such as Darlingtons without problem. When the PIA comes out of reset, the A port represents inputs with pullup resistors, whereas the B side !input mode also) will float high or low, depending upon the load connected to it.

Notice the differences between a Port A and Port B read operation when in the output mode. When reading Port A, the actual pin is read, whereas the B side read comes from an output latch, ahead of the actual pin.

CONTROL REGISTERS (CRA and CRBI

The two Control Registers (CRA and CRB) allow the MPU to control the operation of the four peripheral control lines CA1, CA2, CB1, and CB2. In addition they allow the MPU to enable the interrupt lines and monitor the status of the interrupt flags. Bits 0 through 5 of the two registers may be written or read by the MPU when the proper chip select and register select signals are applied. Bits 6 and 7 of the two registers are read only and are modified by external interrupts occurring on control lines CA1, CA2, CB1, or CB2. The format of the control words is shown in Figure 18.

DATA DIRECTION ACCESS CONTROL BIT (CRA-2 and CRB-21

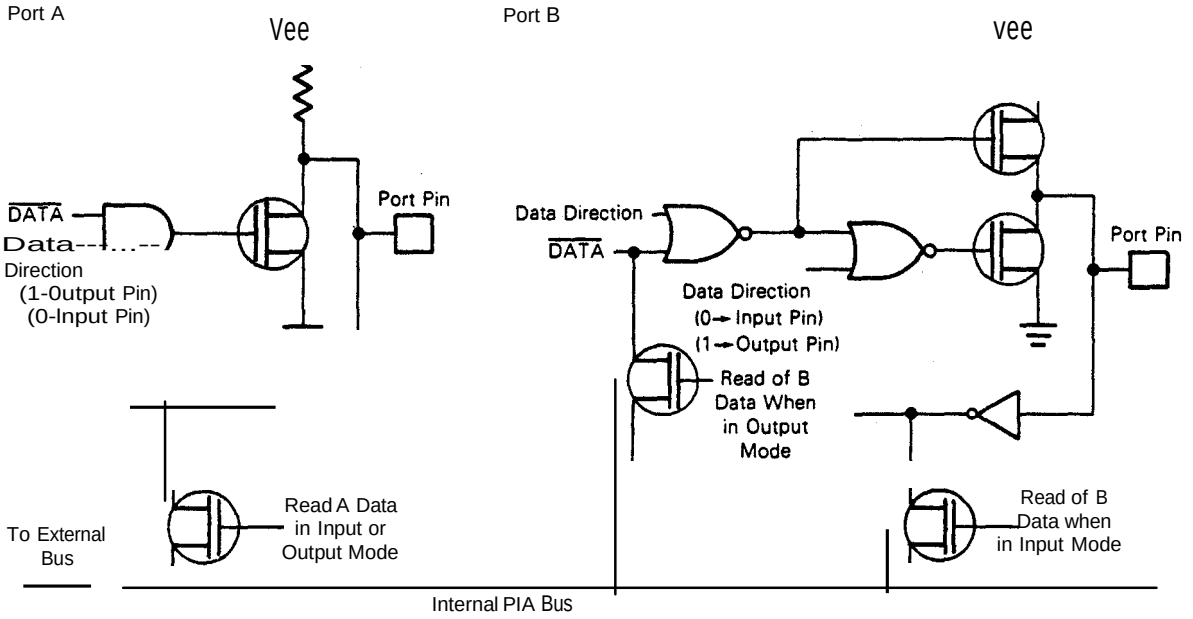
Bit 2, in each Control Register (CRA and CRB), determines selection of either a Peripheral Output Register or the corresponding Data Direction E Register when the proper register select signals are applied toRSO and RS1. A "1" in bit 2 allows access of the Peripheral Interface Register, while a "O" causes the Data Direction Register to be addressed.

Interrupt Flags (CRA-6, CRA-7, CRB-6, and CRB-71 — The four interrupt flag bits are set by active transitions of signals on the four Interrupt and Peripheral Control lines when those lines are programmed to be inputs. These bits cannot be set directly from the MPU Data Bus and are reset indirectly by a Read Peripheral Data Operation on the appropriate section.

Control of CA2 and CB2 Peripheral Control Lines (CRA-3, CRA-4, CRA-5, CRB-3, CRB-4, and CRB-51 — Bits 3, 4, and 5 of the two control registers are used to control the CA2 and CB2 Peripheral Control lines. These bits determine if the control lines will be an interrupt input or an output control signal. If bit CRA-5 (CRB-5) is low, CA2 (CB2) is an interrupt input line similar to CA1 (CB1). When CRA-5 (CRB-5) is high, CA2 tCB2) becomes an output signal that may be used to control peripheral data transfers. When in the output mode, CA2 and CB2 have slightly different loading characteristics.

Control of CA1 and CB1 Interrupt Input Lines (CRA-0, CRB-0, CRA-1, and CRB-11- The two lowest-order bits of the control registers are used to control the interrupt input lines CA1 and CB1. Bits CRA-0 and CRB-0 are used to enable the MPU interrupt signals IRQA and IRQB, respectively. Bits CRA-1 and CRB-1 determine the active transition of the interrupt input signals CA1 and CB1.

FIGURE 17 — PORT A AND PORT B EQUIVALENT CIRCUITS



ORDERING INFORMATION

MC68A21C!

Motorola Integrated Circuit
M6800 Family
Blanks= 1.0 MHz
A= 1.5 MHz
8=2.0 MHz
Device Designation
In M6800 Family
Temperature Range
Blank= O°- + 70°C
C= -4O°- +85°C
Package
P= Plasllc
5= Cerdip
L= Cerar1ic

BETIER PROGRAM

Better program processing is available on all types listed. Add suffix letters to part number.

Level 1 add "S" Level 2 add "D" Level 3 add "DS"

Level 1 "S"= 10 Temp Cycles – I – 25 to 150°C);
Hi Temp testmg at T A max.
Level 2 "'D" = 168 Hour Burn-in at 125°C
Level 3 "DS" =Combination of Level 1 and 2.

FIGURE 18 – CONTROL WORD FORMAT