INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION
INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
I.C.T.P., P.O. BOX 586, 34100 TRIESTE, ITALY, CABLE: CENTRATOM TRIESTE

H4.SMR. 403/18

FIFTH COLLEGE ON MICROPROCESSORS: TECHNOLOGY AND APPLICATIONS IN PHYSICS

2 - 27 October 1989

Microprocessor Interfacing

A. COLAVITA
Microprocessors Laboratory, ICTP, Trieste

THE PIA MC6821 – LECTURE 1 & 2

- MAIN CHARACTERISTICS OF THE PIA
- FUNCTIONAL CHARACTERISTICS FROM THE SPEC. SHEETS.
- CPU TO PIA CONNECTION
- PIA TO DEVICE CONNECTION
- PROGRAMMING THE PIA.
- AN EXAMPLE

MAIN CHARACTERISTICS OF THE PIA

THE MOTOROLA MC6821 PIA (PERIPHERAL INTERFACE ADAPTER) IS AN IDEA CIRCUIT TO CONNECT THE SIGNALS OF THE 6800 FAMILY OF CPU TO PARALLEL DEVICES AND APPLICATIONS. SOME APPLICATIONS COULD BE:

- PRINTERS
- DISPLAYS
- SWITCHES
- RELAYS
- OPTO-COUPLERS
- STEPPING MOTORS, ETC

APPLICATIONS REQUIRING MANY I/O LINES CAN USE SEVERAL PIAs.

THERE IS A CMOS PIN COMPATIBLE CHIP FOR INDUSTRIAL APPLICATIONS THE MC6822. IT HAS 15-18 V OUTPUT DRIVERS

THE USE OF A CHIP ALWAYS REQUIRES A CAREFUL STUDY OF ITS MAIN CHARACTERISTICS. THEY ARE FOUND ON THE SPEC. SHEET FROM THE MANUFACTURER AND APPLICATIONS NOTES.

THE MAIN DEVICE CHARACTERISTICS ARE:

- 8-BIT BIDIRECTIONAL BUS FOR COMUNICATION WITH CPU
- TWO 8-BIT BIDIRECTIONA (PIN SELECTABLE) PORTS FOR COMMUNICATION WITH PERIPHERALS.
- HANDSHAKE CONTROL LOGIC FOR OPERATION WITH PERIPHERALS.
- 4 PROGRAMMABLE INTERRUPT LINES, TWO BIDIRECTIONAL CONTROL LINES TO THE PERIPHERAL
- PROGRAM CONTROLLED INTERRUPTS
- 1, 1,5 AND 2 MHz VERSIONS
- OUTPUTS DRIVES 2 TTL LOADS
- LOW COST
- 3 CHIP SELECT LINES THAT ALLOWS MULTIPLE PIAs TO BE EASILY CONNECTED ON A BOARD.

PIA TO DEVICE CONNECTION

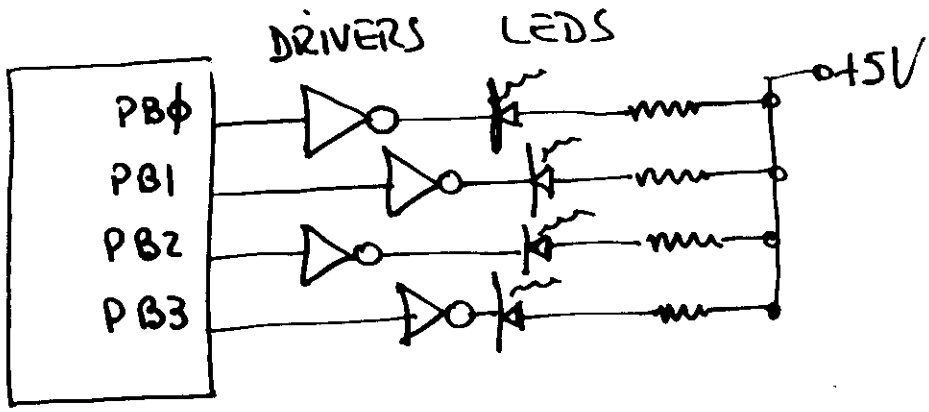
IT HAS TWO ALMOST IDENTICAL PORTS CALLED. A AND B.

EACH LINE PROGRAMMED AS OUTPUTS CAN DRIVE 2 TTL LOADS IN THE LOW STATE.

CONTROL LINES CA1 AND CB1 ARE EDGE SENSITIVE INPUTS AND CAN REACT TO FALLING OR RAISING EDGES.

CONTROL LINES CA2 AND CB2 CAN BE SET AS INPUTS OR OUTPUT ALLOWING THEIR USE AS STROBES. WHEN SET AS INPUTS THEY ARE SIMILAR TO CA1 AND CB1.

EXAMPLE: DRIVING LEDs.



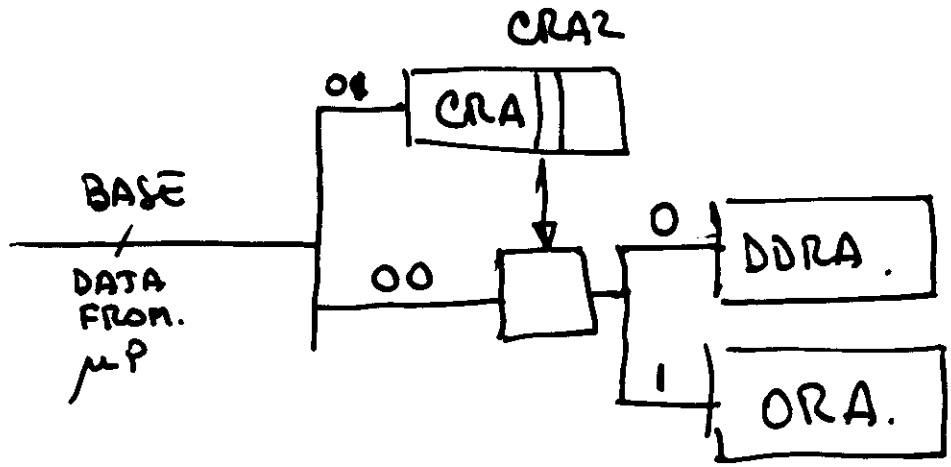
THE FIRST ACTIVITY OF THE PROGRAM HAS TO BE THE INITIALISATION OF THE PIA.

WHEN RESET IS ASSERTED ALL REGISTER'S BITS ARE SET TO ZERO.

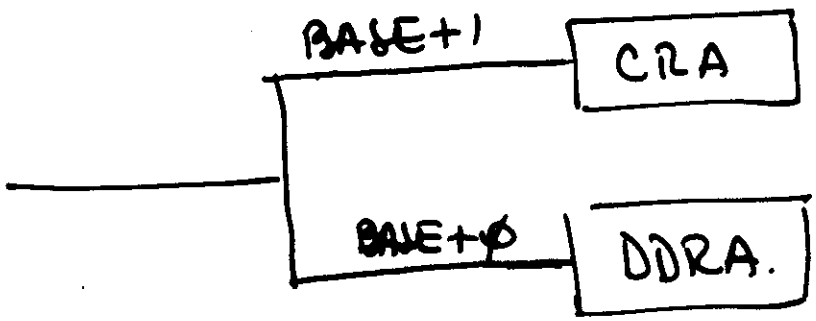
LET US CALL BASE THE BASE ADDRESS THAT SELECTS THE CHIP AND LET US TALK ONLY ABOUT PORT A REMEMBERING THAT PORT B IS IDENTICAL.

	RS1	RS0	REGISTER
BASE + 0	0	0	DDRA OR ORA
BASE + 1	0	1	CRA

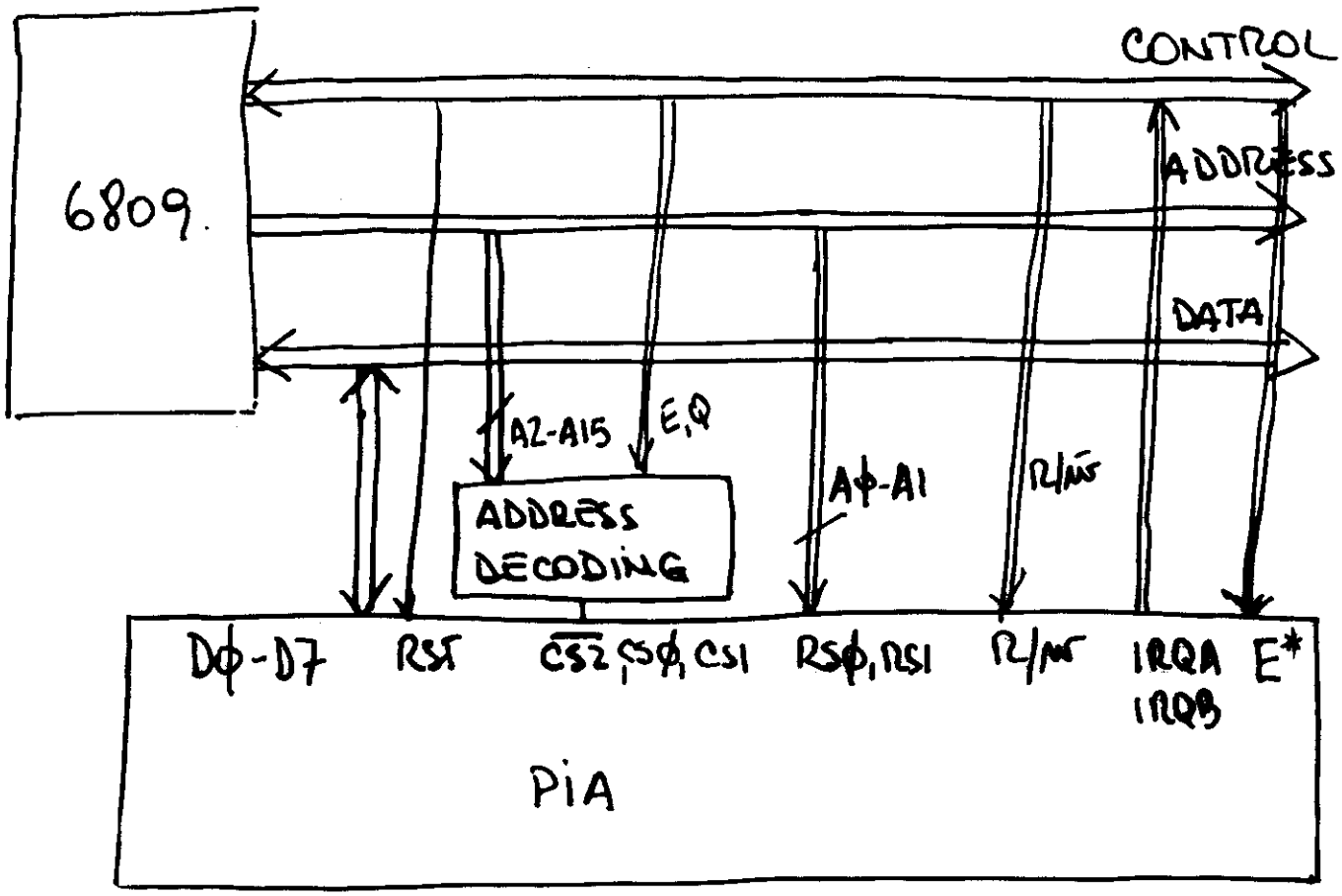
CRA = CONTROL REGISTER A.
DDRA = DATA DIRECTION REGISTER A
ORA = OUTPUT REGISTER A.



AFTER RESET THE PROGRAMMER SEES THE PIA AS.



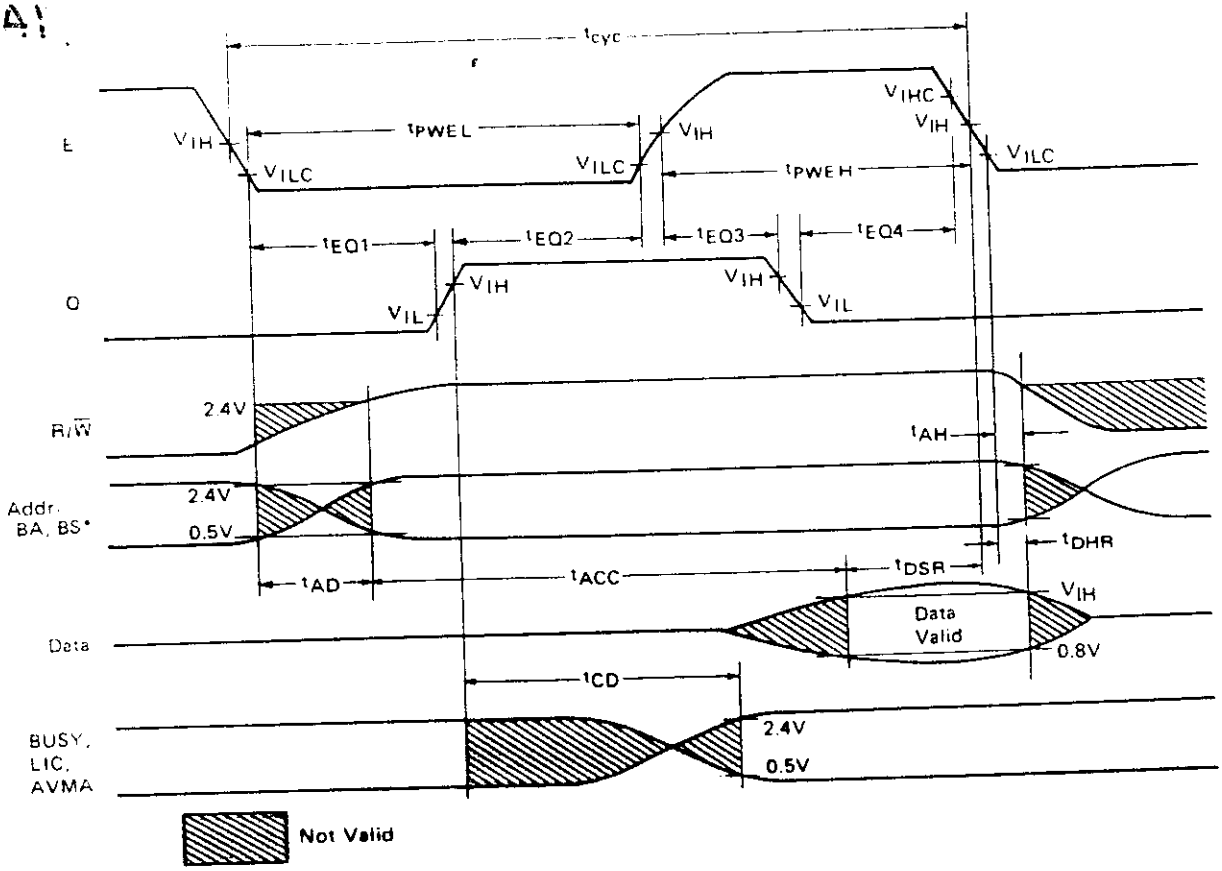
CPU TO PIA CONNECTION



$CS0 = CS1 = +5V$ ✓

- A0-A1 ARE CONNECTED TO RS0-RS1 TO SELECT THE INTERNAL REGISTERS.
- AS SHOWN ALL TRANSFERS TAKE PLACE AT THE TRAILING EDGE OF E.
- A2-A15 AND E,Q GO INTO A DECODING SYSTEM. A VMA (VALID ADDRESS) SIGNAL HAS TO BE GENERATED.
  VMA = (E+Q), WHEN THE RIGHT BASE ADDRESS DURING VMA IS SEEN BY THE BOX A CS (CHIP SELECT) LINE IN THE PIA IS ACTIVATED.
- THE R/W SIGNAL DETERMINES THE DIRECTION OF DATA FLOW.

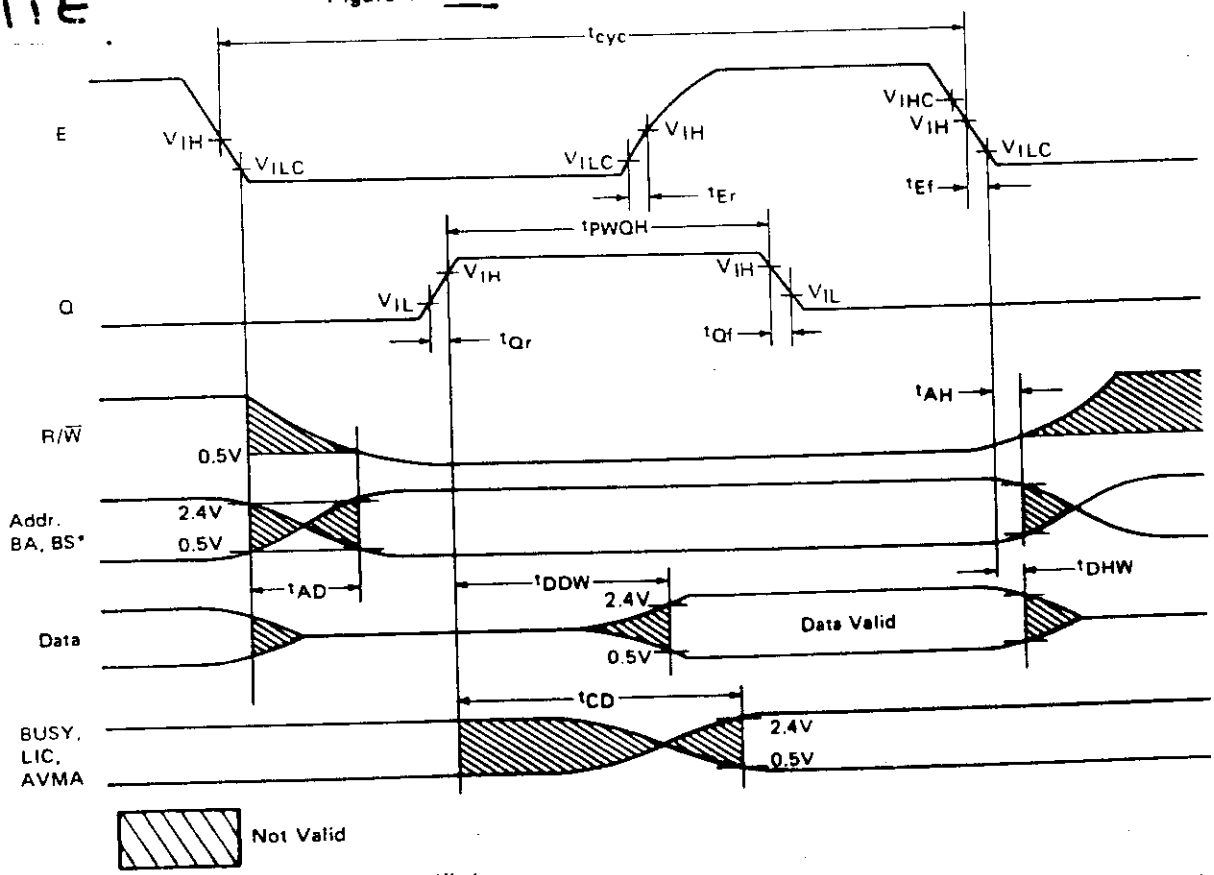
READ



(NOTE) Waveform measurements for all inputs and outputs are specified at logic "High" = V_{IHmin} and logic "Low" = V_{ILmax} unless otherwise specified.

Figure 1 Read Data from Memory or Peripherals

WRITE



(NOTE) Waveform measurements for all inputs and outputs are specified at logic "High" = V_{IHmin} and logic "Low" = V_{ILmax} unless otherwise specified.

Figure 2 Write Data to Memory or Peripherals

FIGURE 14 – $\overline{IRQ}$ RELEASE TIME

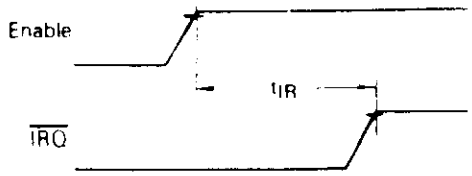
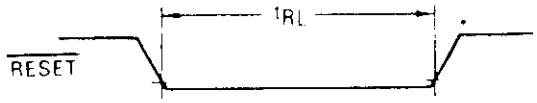


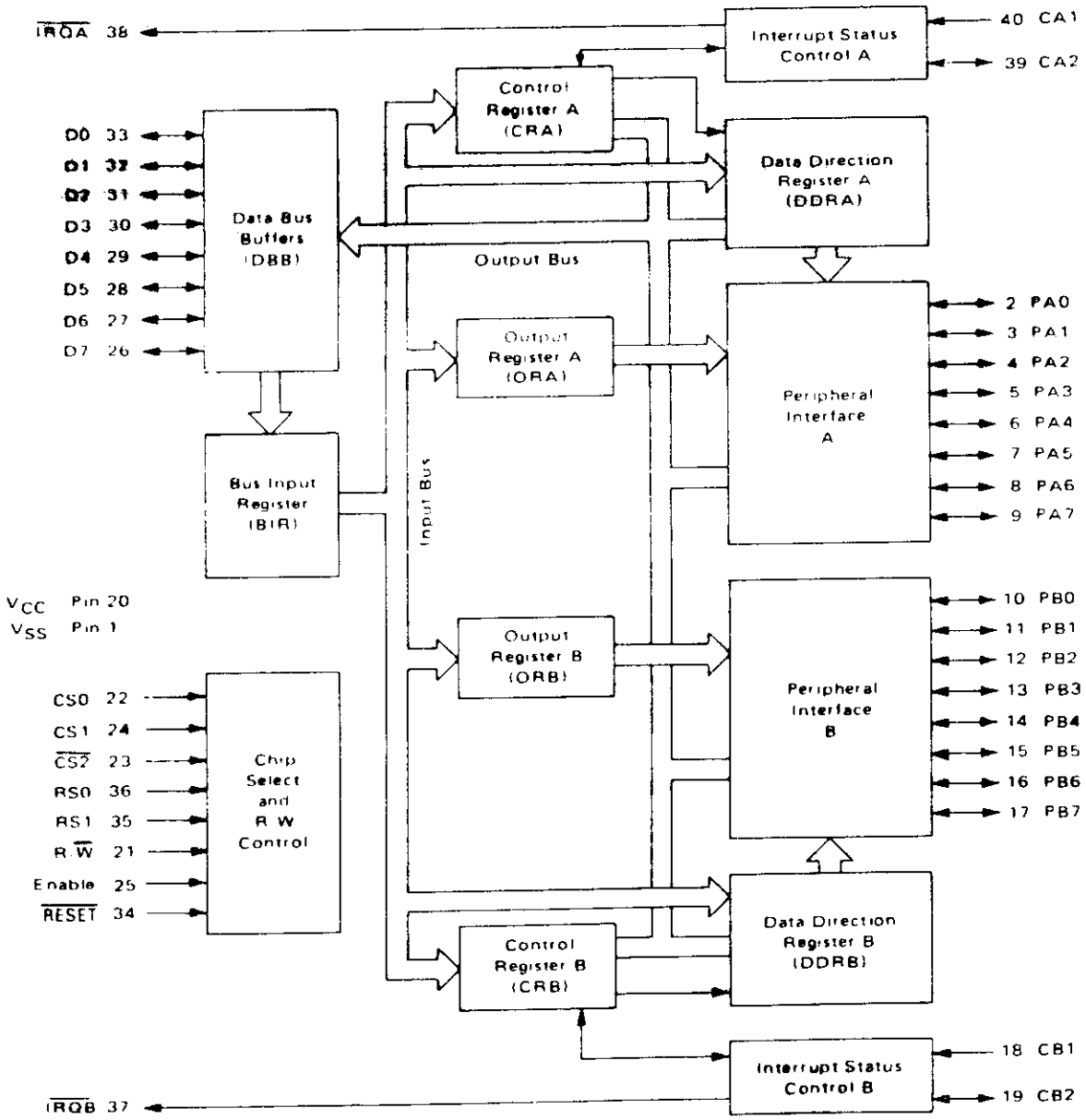
FIGURE 15 – $\overline{RESET}$ LOW TIME



*The $\overline{RESET}$ line must be a V_{IH} for a minimum of 1.0 μs before addressing the PIA.

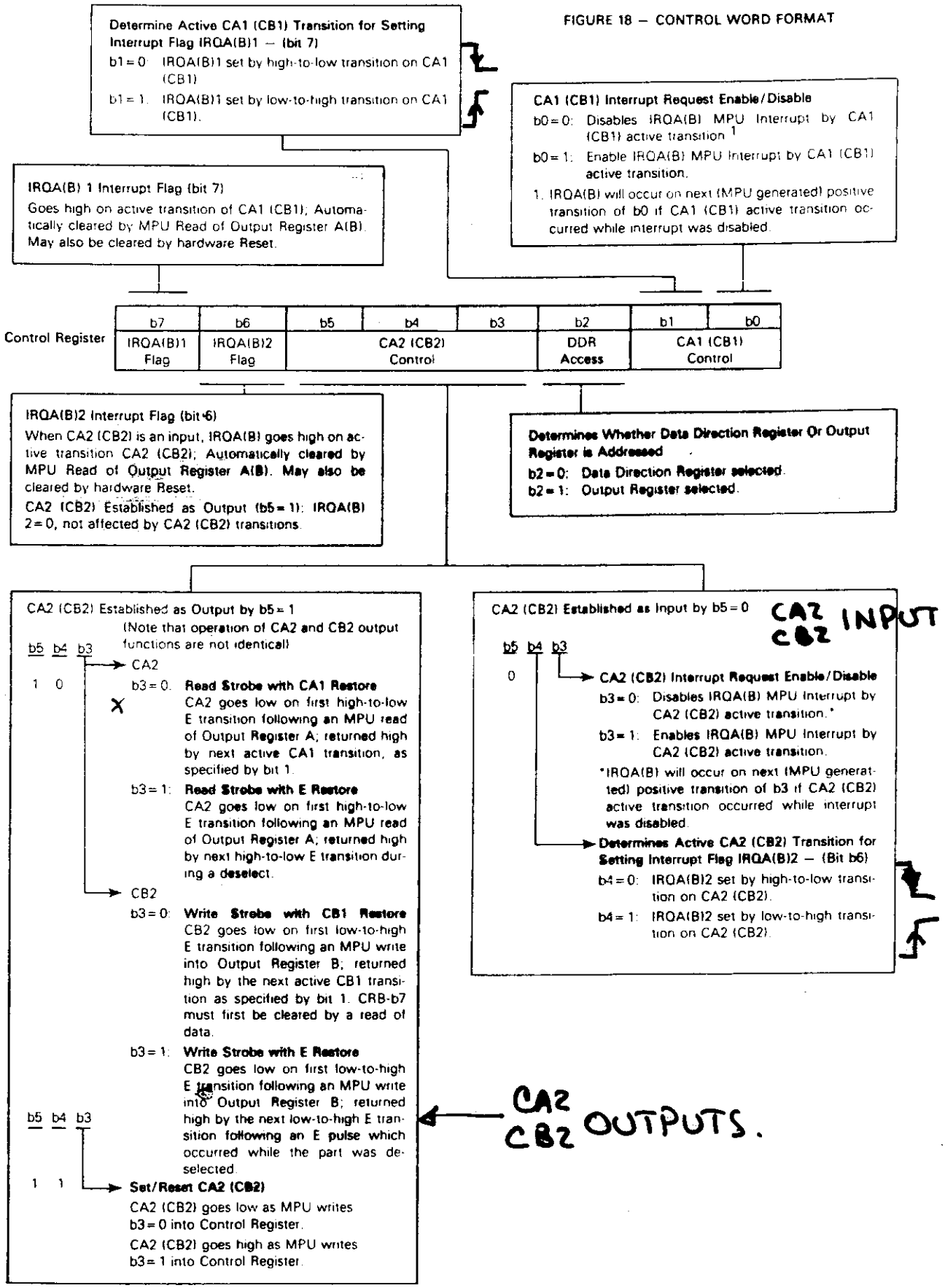
Note: Timing measurements are referenced to and from a low voltage of 0.8 volts and a high voltage of 2.0 volts, unless otherwise noted

FIGURE 16 – EXPANDED BLOCK DIAGRAM



CB1
CA1 is EDGE SENSITIVE

FIGURE 18 – CONTROL WORD FORMAT



PB0-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 CB1) – 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 (CA2) – 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 (CB2) – 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 used 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.

3

INTERNAL CONTROLS

INITIALIZATION

A $\overline{\text{RESET}}$ has the effect of zeroing all PIA registers. This will set PA0-PA7, PB0-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 RS0 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

		Control Register Bits		
RS1	RS0	CRA-2	CRB-2	Location Selected
0	0	1	X	Peripheral Register A
0	0	0	X	Data Direction Register A
0	1	X	X	Control Register A
1	0	X	1	Peripheral Register B
1	0	X	0	Data Direction Register B
1	1	X	X	Control Register B

X = Don't Care

PORT A-B HARDWARE CHARACTERISTICS

As shown in Figure 17, the MC6821 has a pair of I/O ports whose characteristics differ greatly. The A side is designed to drive CMOS logic to normal 30% 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 CRB)

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-2)

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 to RS0 and RS1. A "1" in bit 2 allows access of the Peripheral Interface Register, while a "0" causes the Data Direction Register to be addressed.

Interrupt Flags (CRA-6, CRA-7, CRB-6, and CRB-7) – 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-5) – 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 (CB2) 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.

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 (D0-D7) – The bidirectional data lines (D0-D7) 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 (off) 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 the E pulse.

Read/Write (R/$\overline{W}$) – 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.

$\overline{\text{RESET}}$ – The active low $\overline{\text{RESET}}$ line is used to reset all register bits in the PIA to a logical zero (low). This line can be used as a power-on reset and as a master reset during system operation.

Chip Selects (CS0, CS1, and $\overline{CS2}$) – These three input signals are used to select the PIA. CS0 and CS1 must be high and $\overline{CS2}$ 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 (RS0 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 ($\overline{IRQA}$ and $\overline{IRQB}$) – The active low Interrupt Request lines ($\overline{IRQA}$ and $\overline{IRQB}$) 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 (CA1, CA2, CB1, CB2). 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 (PA0-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 "1" output and less than 0.8 volt for a logic "0" output. Loading the output 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 B Peripheral Data (PB0-PB7) – The peripheral data lines in the B Section of the PIA can be programmed to act as either inputs or outputs in a similar manner to PA0-PA7. They have three-state capability, 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

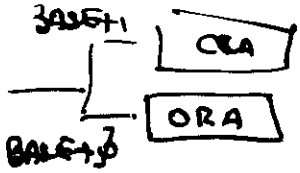
FOR THE MOMENT WE WILL CARE ONLY ABOUT THE USE OF CRA-2 FORGETTING FOR A WHILE ABOUT THE OTHERS (PROGRAM-CONTROLLED).

A NORMAL INITIALISATION SEQUENCE IS.

1) SET CRA-2 = ϕ TO BE SURE TO ACCESS DDRA

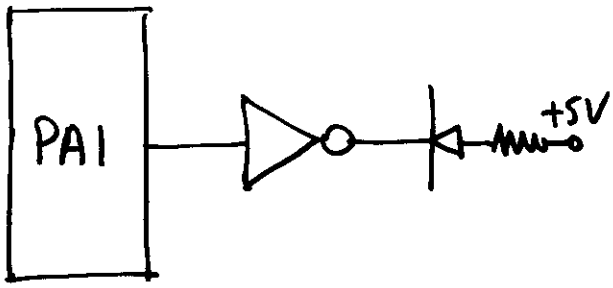
2) SELECT DIRECTION OF LINES
   0 = INPUT , 1 = OUTPUT

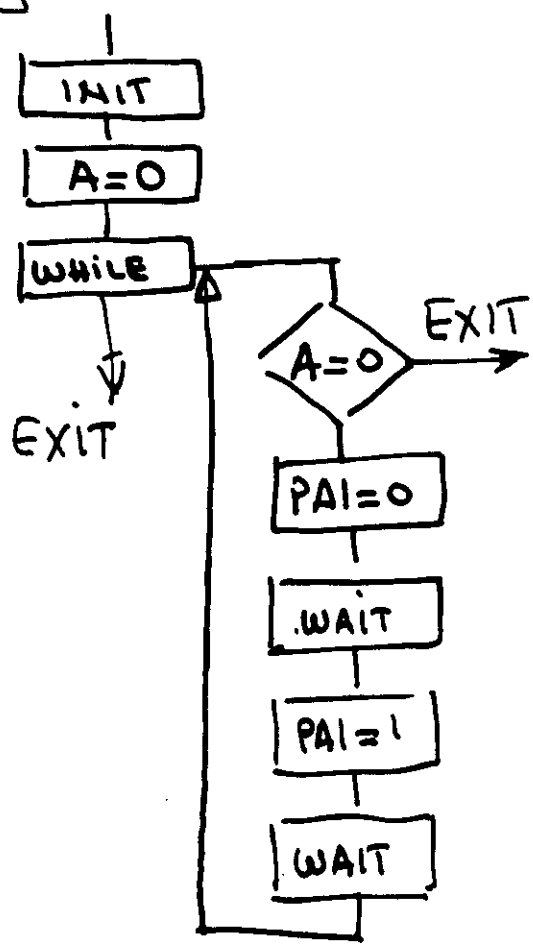
3) SET CRA-2 = 1 TO ACCESS ORA AND BE ABLE TO EXCHANGE DATA WITH THE PERIPHERAL.



A CASE STUDY

MAKE A LED BLINK.





```
*  PROGRAM TO MAKE LED BLINK, THRU PA1
BASE    EQU  $XXXX      DEVICE ADDRESSES
DDRA    EQU  BASE+0
ORA     EQU  BASE+0
CRA     EQU  BASE+1

        ORG  $1000
INIT    CLR  CRA        SURE TO ADDRESS DDRA.
        LDA  #$02       PA1=OUTPUT, ALL OTHER INPUTS
        STA  DDRA
        LDA  #$04       SURE TO ADDRESS ORA.
        STA  CRA.
        CLR  TEST       TEST=0
WHILE   LDA  TEST
        BNE  QUIT       QUIT IF TEST≠0
DO      LDA  #$00       TURN LED OFF
        STA  ORA
        BSR  WAIT       WAIT
        LDA  #$02       TURN LED ON.
        STA  ORA
        BSR  WAIT       WAIT
        BRA  WHILE
QUIT    NOP             QUIT IS NEVER USED.
*
WAIT    ≡               WAIT TO SEE LED ON OR OFF
        RTS.
*
TEST    RMB  1          DUMMY VARIABLE FOR WHILE
        END
```