Segment 6

8255A Programmable Peripheral Interface (PPI)

Content
• Why 8255A?
• Handshaking and Handshaking Signal
• Parallel Data Transfer
• 8255A Internal Block Diagram
• Description of 8255A Internal Block Diagram
• Programming Modes of 8255A
• Related Problems

Prepared By:
Mohammed Abdul kader
Lecturer, EEE, IIUC
Why 8255A??

- Intel 8255A is a general purpose parallel I/O interface.
- The peripheral devices are slower than the microprocessor. PPI makes an inter-relation between microprocessor and peripheral devices.
- It provides three I/O port (Port A, Port B and Port C)
Handshaking

The making of inter relation between slower peripheral device and microprocessor is called handshaking.

Handshaking Signal

Before making the inter-relation between peripheral device and microprocessor the PPI send some signals to microprocessor and peripheral device to perform the process, these signals are called handshaking signal. 8255-based devices that perform handshaking support following handshaking signals:

1) Strobe Input (STB)
2) Input Buffer Full (IBF)
3) Output Buffer Full (OBF)
4) Acknowledge Input (ACK)
5) Interrupt
Read Operation

**STB goes low** indicates that data are loads into port latch.

**IBF Becomes high** (at high to low transition of STB) indicates that input latch contains data.

**INTR Becomes high** (at low to high transition of STB) CPU goes interrupt subroutine to read data. **RD becomes low.**

**IBF becomes low** when read complete, RD becomes high and IBF goes low.

**IBF low means input latch has no data** (Read complete)
Handshaking Signal (Continued)
Parallel Data Transfer:

**Simple I/O:**
This data transfer method is used when the I/O devices need no communication before the data transfer. Such devices are thermostat, LED. The crossed lines on the wave form represent the time at which a new data byte becomes valid on the output lines of the port.

```
<table>
<thead>
<tr>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>
```

**Simple Strobe I/O:**
The sending device, such as a keyboard, outputs parallel data on the data lines, and then outputs an $STB$ signal to let the receiving device know that valid data is present.
Parallel Data Transfer (Continued):

**Single handshake Data transfer:**

- The sending device outputs some parallel data and sends an $STB$ signal to the receiving device. (i.e. sending device says receiving device, “I have some data for you”)
- As a response of $STB$ signal receiver device reads data and send an acknowledge signal to indicate that the data has been read. (i.e. by acknowledge signal receiving device says to sending device, “your sending data is received and I am ready to get new data”)

![Diagram of single handshake data transfer](image)
Parallel Data Transfer (Continued):

**Double Handshake Data Transfer:**

- The sending device asserts its $STB$ line low to ask the receiving device “are you ready?”
- The receiving device raises its ACK line high to say “I am ready”.
- The peripheral device then sends the byte of data and raises its $STB$ line high to say “Here is some valid data for you”.
- After it has read in the data, the receiving data drops its ACK line low to say “I have the data”. The receiving device is then ready to be requested for accepting the next data byte.
8255A Internal Block Diagram
Description of 8255A Internal Block Diagram

**Data Bus Buffer**
- This three-state bi-directional 8-bit buffer is used to interface the 8255 to the system data bus.
- Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU.
- Control words and status information are also transferred through the data bus buffer.

**Read/Write Control Logic**
The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words.

*(CS)* Chip Select. A "low" on this input pin enables the communication between the 8255 and the CPU.

*(RD)* Read. A "low" on this input pin enables 8255 to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255.

*(WR)* Write. A "low" on this input pin enables the CPU to write data or control words into the 8255.

*(A0 and A1)* These input signals, control the selection of one of the three ports or the control word register.

*(RESET)* Reset. A "high" on this input initializes the control register to 9Bh and all ports (A, B, C) are set to the input mode.

<table>
<thead>
<tr>
<th>A1</th>
<th>A0</th>
<th>SELECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>PORT A</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>PORT B</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>PORT C</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>CONTROL</td>
</tr>
</tbody>
</table>
Description of 8255A Internal Block Diagram (Continued)

Group A and Group B Controls

Each of the Control blocks (Group A and Group B) accepts "commands" from the CPU as "control word" and configure the ports (Port A, Port B and Port C) accordingly.

Port A and upper 4 bits of Port C are controlled by Group A
Port B and lower part of Port C are controlled by Group B

Ports A, B, and C

The 8255 has three 8 bit I/O ports and each one can be connected to the physical lines of an external device. All can be configured to a wide variety of functional characteristics by the system software. These ports are labeled as PA0-PA7 (PortA), PB0-PB7 (PortB) and PC0-PC7 (PortC).

GND (Ground) and Vcc

Summary of 8255A Pins

Port Pins: 24 (Port A = 8, Port B = 8, Port C = 8)
Control Pins: 6 (RD, WR, CS, RESET, A1, A0)
Data Lines: 8
Power Supply: 2 (VCC, GND)

Total 40 pins
Modes of Operation

8255 can be configured in two modes
• BSR (Bit Set Reset) Mode
• I/O (Input-Output) Mode: Mode 0, Mode 1 and Mode 2

Modes are configured by Control Word

Control Word

A Control Word is an 8-bit data that stored in control register. Control Words are two types: (a) BSR Control Word (b) Mode definition Control Word
BSR Mode (Configured by Bit Set-Reset Control Word)

- If bit 7 of control word is a logic 0 then 8255 will be configured as BSR (Bit Set Rest) mode.
- In this mode we can set or reset the pins of port C

Problems: (a) Write a control word to reset PC5. (Ans: 0AH)
             (b) Write a Control Word to Set PC2. (Ans: 05H)

N.B: Don’t Cares are Generally set as zero.
I/O Mode (Configured by Mode definition CR)

- If bit 7 of the control word is a logical 1 then the 8255 will be configured as I/O mode.
- I/O mode consists of Mode0, Mode1 and Mode2.

**Mode 0**

- Port A works as simple input or output without handshaking.
- Port B works as simple input or output without handshaking.
- Port C can be used together as an additional 8 bit port or they can be used individually as two 4-bit ports.
- When used as outputs, the Port C lines can be individually set or reset by sending a special control word to the control register address.
**I/O Mode (Cont.)**

**Mode 1**
- Used for handshake input/output operation.
- Port B is initialized in mode 1 for either input or output, Pins PC0, PC1 and PC2 function as handshake lines.
- Port A can also be configured as input or output in mode 1. But handshake signal pins are not same for input and output mode as like Port B.
- If port A is initialized in mode 1 as handshake input port, then pins PC3, PC4 and PC5 function as handshake signals. (PC6 and PC7 are available for using as input lines or output lines)
- If port A is initialized as handshake output port, then PC3, PC6 and PC7 function as handshake signals. (PC4 and PC5 are available for using as input or output lines)
I/O Mode (Cont.)

Connection of Handshaking lines when Port A and Port B configured as Input in Mode 1

Connection of Handshaking lines when Port A and Port B configured as Output in Mode 1
I/O Mode (Cont.)

Mode 2

- Only port A can be initialized in mode 2.
- In mode 2, port A can be used for “bi-directional handshake” data transfer i.e. data can be input or output on the same eight lines.
- Pins PC3, PC4, PC5, PC6, PC7 used as handshake lines for port A.
- Port B is operating in either mode 0 or mode 1.
- If port B is in mode 0, then PC0, PC1 and PC2 used for I/O.
- If port B is in mode 1, then PC0, PC1 and PC2 used as handshake lines.
Handshaking Lines in Mode 2

8086

WR

8255

Data Bus

Port A

A0
PC5

A1
PC4

CS
PC7

WR
PC6

INTRA

Port B

PC3

IBF_A

8255

STB_A

PC5

OBF_B

PC4

ACK_B

PC7

INTRA

Input/Output Device

Mode 0 or Mode 1

IBF_B or OBF_B if Port B in Mode 1
I/O if Port B in Mode 0

ACK_B or STB_B if Port B in Mode 1
I/O if Port B in Mode 0

INTRB if Port B in Mode 1
I/O if Port B in Mode 0

Lecture materials on "8255 Programmable Peripheral Interface" Prepared by- Mohammed Abdul kader, Lecturer, EEE, IIUC 18
## I/O Mode (Cont.)

### 8255A modes summarization

<table>
<thead>
<tr>
<th>Mode</th>
<th>8- Bit Port A</th>
<th>8-Bit Port B</th>
<th>Handshake lines for mentioned Mode</th>
<th>Other Port C pins available for I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>I/O</td>
<td>I/O</td>
<td>N/A</td>
<td>PC7- PC4 and PC3- PC0</td>
</tr>
<tr>
<td>1</td>
<td>Input</td>
<td>I/O</td>
<td>PC3, PC4, PC5 (Port A) PC0, PC1, PC2 (Port B)</td>
<td>PC6, PC7</td>
</tr>
<tr>
<td>1</td>
<td>Output</td>
<td>I/O</td>
<td>PC3, PC6, PC7 (Port A) PC0, PC1, PC2 (Port B)</td>
<td>PC4, PC5</td>
</tr>
<tr>
<td>2</td>
<td>Bi-Directional</td>
<td>N/A</td>
<td>PC3, PC4, PC5, PC6, PC7</td>
<td>PC0, PC1, PC2 (If Port B in Mode 0)</td>
</tr>
</tbody>
</table>
I/O Mode (Cont.)
Configuring I/O Mode
I/O Mode is configured by Mode Definition Control Word

![Diagram showing the configuration of I/O Mode through Mode Definition Control Word](image-url)
Problems

Problem 1: Write a control word to configure port A as input port in mode 0 and port B in mode 1 as output port.

Solution:

<table>
<thead>
<tr>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

The control word is 94H.

N.B. D0 and D3 are low if port C is used as output or if unused.

Problem 2: A control word is given CW=CDH. Explain the conditions of ports of 8255A.

Solution:

<table>
<thead>
<tr>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

D7=1; I/O Mode.
D6=1 and D5=0; Port A is in Mode 2.
D4=0; Port A is output port
D3=1; Port C (Upper) is input port.
D2=1; Port B is in Mode 1.
D1=0; Port B is output Port.
D0=0; Port C (Lower) is input port.

Q3: Configure Port A in Mode 2, Port B as o/p in mode 1.
**Problems**

**Problem 3:** Configure Port A in Mode 2, Port B as o/p in mode 1.

**Solution:**

<table>
<thead>
<tr>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1</td>
<td>0</td>
<td>X</td>
</tr>
</tbody>
</table>

Control word is C4H / C5H..... etc

**Problem 4:** Write an 8086 assembly language procedure to read an ASCII character from a keyboard via PORT A of an 8255 PPI when PORT C bit PC4 is strobed low. Assume a base address of 20H.

**Solution:**

```
PORTA EQU 20H
PORTC EQU 22H
CONTROL EQU 23H

READ PROC NEAR
    MOV AL, 98H ; 1001 1000
    OUT CONTROL, AL ; Initialize PORTS

READ1:
    IN AL, PORTC ; Is Strobe PC4 Low?
    TEST AL, 10H ; 0001 0000
    JNZ READ1
    IN AL, PORTA ; Read ASCII Character
    RET

READ ENDP
```

Lecture materials on "8255 Programmable Peripheral Interface" Prepared by-Mohammed Abdul kader, Lecturer, EEE, IIUC
Problem 5: Write an 8086 assembly language procedure to send an ASCII character, stored in register AH, to a printer via PORT B of an 8255 PPI when PORTC bit PCO is strobed low and after an active low acknowledge signal is detected on PORT C bit PC5 from the printer. Assume a base address of 60H.

Solution:

PORTB EQU 61H  
PORTC EQU 62H  
CONTROL EQU 63H  
PRINT PROC NEAR  
  MOV AL, 88H ; 1000 1000  
  OUT CONTROL, AL;  
PRINT1:  
  IN AL, PORTC ; Is Acknowledge PC5 Low?  
  TEST AL, 20H ; 0010 000  
  JNZ PRINT1  
  MOV AL, AH ; Send character  
  OUT PORTB, AL;  
  MOV AL, FEH ; 1111 1110  
  OUT PORTC, AL ; strobe output PC0  
  RET  
PRINT ENDP
Problem 6: An 8086-8255 based microcomputer is required to drive an LED connected to bit 2 of Port B based on two switch inputs connected to bit 6 and 7 of port A. If both switches are either high or low, LED will turn on; otherwise, it will remain OFF. Assume base address of 60H. Write an 8086 assembly language program to accomplish this.

Solution:

PORTA EQU 60H
PORTB EQU 61H
CONTR EQU 63H
MOV AL, 10010000B; Configure Port A as input and Port B as output.
OUT CNTRL, AL;
MAIN : IN AL, PORTA;
   AND AL, 11000000B;
   JPE LEDON; (JPE= Jump if parity even, p=1)
   MOV AL, 00H;
   OUT PORTB, AL;
   JMP MAIN;
LEDON : MOV AL, 00000100B;
   OUT PORTB, AL;
   JMP MAIN;
**Problem 7:** Figure shows an 8255A interfaced with 8086 microprocessor. Perform the following-
(a) Identify the Port Address.
(b) Identify the Mode 0 control word to configure Port A and Port \( C_U \) as output ports and Port B and Port \( C_L \) as input ports.
(c) Write a program to read the DIP switches and display the reading from Port B at Port A, and from Port \( C_L \) at Port \( C_U \).
Solution of problem 7:

(a) When A3 to A7 are high then chip select ($CS$) is enabled.

<table>
<thead>
<tr>
<th>A15</th>
<th>A14</th>
<th>A13</th>
<th>A12</th>
<th>A11</th>
<th>A10</th>
<th>A9</th>
<th>A8</th>
<th>A7</th>
<th>A6</th>
<th>A5</th>
<th>A4</th>
<th>A3</th>
<th>A2</th>
<th>A1</th>
<th>A0</th>
<th>Address of Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>00F8H (Port A)</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>00F9H (Port B)</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>00FAH (Port C)</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>00FBH (CR)</td>
</tr>
</tbody>
</table>

(b) Control Word

<table>
<thead>
<tr>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Mod Set  | Port A in Mode 0  | Port A I/O  | Port C upper o/p  | Port B in Mode 0  | Port B input  | Port C lower input

Control Word = 83 H
Problems

Solution of problem 7(c)
Program

PPICR EQU 00FBH
PPIC  EQU 00FAH
PPIB  EQU 00F9H
PPIA  EQU 00F8H
MOV AL, 83H
OUT PPICR, AL
IN AL, PPIB
OUT PPIA, AL
IN AL, PPIC
AND AL, 0FH
MOV CL, 04H
ROL AL, CL
OUT PPIC, AL
HLT

OR

Solution of problem 7(c)
Program

PPICR EQU 00FBH
PPIC  EQU 00FAH
PPIB  EQU 00F9H
PPIA  EQU 00F8H
MOV AL, 83H
OUT PPICR, AL
IN AL, PPIB
OUT PPIA, AL
IN AL, PPIC
AND AL, 0FH
MOV CL, 04H
SHL AL, CL
OUT PPIC, AL
HLT
Problem 8:
Write a BSR control word subroutine to set bit PC7 and PC3 and reset them after 10ms. Use the previous schematic (Figure of problem 7). Also write the delay procedure considering the processor clock at 5 MHz

Solution:
BSR Control Word:

<table>
<thead>
<tr>
<th></th>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
<th>Control Word</th>
</tr>
</thead>
<tbody>
<tr>
<td>To set Bit PC7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0FH</td>
</tr>
<tr>
<td>To reset PC7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0EH</td>
</tr>
<tr>
<td>To set PC3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>07H</td>
</tr>
<tr>
<td>To reset PC3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>06H</td>
</tr>
</tbody>
</table>

Address of control register: 8003H (See previous problem)
Duration of 1 clock pulse= (1/5MHz) = 200 ns
So, for 10ms we have to count = (10ms/200ns) = 50,000

Subroutine:

```assembly
PPICR EQU 8003H
BSR: MOV AL, 0FH
    OUT PPICR, AL
    MOV AL, 07H
    OUT PPICR, AL
CALL DELAY
    MOV AL, 06H
    OUT PPICR, AL
    MOV AL, 0EH
    OUT PPICR, AL
    RET

DELAY  PROC NEAR
    MOV CX, 50,000
    HERE: LOOP HERE
    RET
DELAY  ENDP
```
Problem 9: Write an assembly language instruction set to drive the DC motor interfaced with 8086 microprocessor as shown in figure. The switches SW0 and SW1 control the motor status according to the table.

<table>
<thead>
<tr>
<th>SW1</th>
<th>SW0</th>
<th>Motor status</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>No rotation</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>Forward rotation</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>Reverse rotation</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>No rotation</td>
</tr>
</tbody>
</table>

Solution:

<table>
<thead>
<tr>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
<th>Control Word</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>82H</td>
</tr>
</tbody>
</table>
PORTA EQU 00F8H
PORTB EQU 00F9H
CONTR EQU 00FBH

MOV AL, 82H
OUT CNTRL, AL;  Control word sent to CR

MAIN : IN  AL, PORTB;
   AND AL, 00000011B;  Check switch
   JPE  STOP;  (JPE=Jump if parity even, p=1)
; if two switches are on or off, parity will be even

TEST AL, 00000001B; check whether SW0 is on or off
JZ FORWARD ; if SW0 is on, result becomes zero (Z=1)
MOV AL, 00000001B;
OUT  PORTA, AL; motor rotate reverse direction
JMP MAIN;

STOP : MOV  AL, 00000000B;
OUT PORTA, AL;
JMP  MAIN;
FORWARD: MOV AL, 00000010
OUT  PORTA, AL
JMP MAIN
Assignment-1: A 8086-8255A based system is given in fig. The Port A of 8255A is connected with a 7-segment display. Write an assembly language program to display the numbers 1,3,5,7,9 repeatedly.
Assignment-2: An 8X8 dot matrix LED display is connected to 8255-8086 based system. The rows of the display are connected to PORTA and columns to PORTB as shown in Figure. A. The internal circuit of the dot matrix display is shown in Figure. B. Write an assembly language program to show Bengali number “1” on the display. Assume that the addresses of PORTA, PORTB and control register are 60H, 61H and 63H respectively.

Figure. A.  
Figure. B.