8085
BASIC INTERFACING CONCEPTS
BASIC INTERFACING IN MICROPROCESSOR

![Diagram of basic interfacing in microprocessor]

- Microprocessor
- Address Bus
- Data Bus
- Control Bus
- Memory
- I/O device
BASIC INTERFACING IN MICROPROCESSOR

• In memory interfacing, 8 bit data line, 16 bit address line, control signals are connected to corresponding lines of memory IC.

• In I/O device interfacing, 8 bit data line, only 8 bit address line, control signals are connected to corresponding lines of I/O devices.
Classification of I/O Interfacing

Input/Output Interface

CPU initiated
  - Conditional
  - Unconditional

Device initiated
  - Interrupt
  - DMA
Data Transfer using I/O STRUCTURE

I/O

Programmed I/O

Standard I/O or Isolated I/O or Port I/O

Memory Mapped I/O

External

Maskable

Interrupt I/O

Non-maskable

Direct Memory Access

Block transfer DMA

Due to exceptional conditions

Cycle stealing DMA

Software interrupts

Demand transfer DMA
Data Transfer using I/O STRUCTURE

- There are **three major types of data transfer** between the microprocessor and I/O device.

- Programmed I/O: In programmed I/O the **data transfer is accomplished through an I/O port** and controlled by software.

- Interrupt driven I/O: In interrupt driven I/O, the **I/O device will interrupt the processor**, and initiate data transfer.

- Direct memory access (DMA): In DMA, the data transfer between memory and I/O can be **performed by bypassing the microprocessor**.
INTERFACING I/O DEVICES
INTERFACING OF INPUT AND OUTPUT DEVICE

I/P device

Tri state Buffer

O/P device

Address decoder

Logic

D0- D7

AD0- AD7
INTERFACING OF INPUT AND OUTPUT DEVICE

- **I/O mapped or programmed interfacing** scheme is commonly used.

- The **data lines are connected to the I/O devices through Tri-state buffer**.

- **Tri- State buffer is enabled from address decoder logic**.
INTERFACING OF INPUT AND OUTPUT DEVICE

• The address decoder logic makes an enable signal according to the address data coming from microprocessor.

• These address is the address of a ports.

• IN and OUT instruction is used for data transfer
  • Eg. IN ,Port address; IN 02
  • OUT, Port address; OUT 03
I/O SCHEMES
Classification of I/O Interfacing

I/O interfacing techniques

I/O devices can be interfaced in two ways

1) I/O mapped I/O
2) Memory mapped I/O
I/O Addressing Schemes

I/O interfacing techniques

<table>
<thead>
<tr>
<th>Memory mapped I/O</th>
<th>I/O mapped I/P</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) In this device add is 16 bit (A0-A15)</td>
<td>1) In this device add is 8 bit (A0-A7)</td>
</tr>
<tr>
<td>2) MEMR^ and MEWR^ control signals are used</td>
<td>2) IOR^ and IOW^ control signals are used</td>
</tr>
<tr>
<td>3) Instructions are LDA add, STA Add, MOV A,M</td>
<td>3) Instruction are IN Add, OUT Add</td>
</tr>
<tr>
<td>5) No. of I/O devices interface = 65536 (Theoretically)</td>
<td>5) No. of I/O devices interface = 256 only</td>
</tr>
</tbody>
</table>
### I/O Mapped I/O Addressing

**I/O address space**

selected when IO/M* = 1

<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>00H</td>
<td></td>
</tr>
<tr>
<td>67H</td>
<td></td>
</tr>
<tr>
<td>FFH</td>
<td></td>
</tr>
</tbody>
</table>
I/O Mapped I/O addressing

1 A_7
1 A_6
1 A_5
0 A_4
1 A_3
1 A_2
1 A_1
1 A_0
0 RD*
1 IO/M* → NAND → To CS* of input port chip
INTERFACING OF I&O DEVICE

To Data line of μP
INTERFACING OF I&O DEVICE

[Diagram showing an integrated circuit (IC) 7404 connected to a LED with a resistor and power supply (VCC)]
INTERFACING OF I&O DEVICE

Darlington Connection

Input

6.2K
INTERFACING OF I&O DEVICE

LOGIC 1 = ON
LOGIC 0 = OFF

CONTROL

10K

DIODE

VCC

RELAY COIL

LOAD

RELAY CONTACT

GND

NPN
INTERFACING OF I&O DEVICE
INTERFACING INPUT DEVICES
INTERFACING OF INPUT DEVICE

- I/P Port

- I/P PORT Buffer

- D0-D7

- Enable

- Data from Keyboard
INTERFACING OF INPUT DEVICE

The diagram shows a microprocessor connected to various interfaces (Interface 1 to Interface 6) which then connect to different input devices (Input device 1 to Input device 3).
INTERFACING OF INPUT DEVICE

I/P device

O/P device

Tri state Buffer

AD0-AD7

D0-D7

I/O/ M

RD

Active High

Not Using
INTERFACING OF INPUT DEVICE

I/P device

O/P device

Tri state Buffer

AD0- AD7

D0- D7

I/O/ M

RD

Active Low

Not Using

Active Low
INTERFACING OF INPUT DEVICE

• The address lines are decoded to generate a signal that is active when the particular port is being accessed.

• An IORD signal is generated by combining the IO/M and the RD signals from the microprocessor.
INTERFACING OF INPUT DEVICE

- Lets **choose I/O port 0FH** for the Input devices.

- So, the buffers must be enabled when:
  - RD = 0
  - IO/M = 1
  - A0-A8 = 0FH
INTERFACING OF INPUT DEVICE

A0

A7

I/O/M

RD

Buffer Enable
INTERFACING OF INPUT DEVICE

[Diagram showing a SPST switch connected to VCC via a 2.2K resistor, labeled "To µP"]
INTERFACING OF INPUT DEVICE

DIP Switch
INTERFACING OF INPUT DEVICE

Decode logic for a Dip-Switch Input Port

Active Low
INTERFACING OF INPUT DEVICE

- **Program:**
  - IN 0FH ;input data from port 0F into A
  - Loop2 CPI Data; Data according to switch position
  - JNZ Loop1; What to do?
  - Perform the operation for which when switch is pressed
  - Loop1 JMP Loop2; Repeat checking of switch condition
  - HLT
INTERFACING OUTPUT DEVICES
INTERFACING OF OUTPUT DEVICE

O/p Port

D0-D7 → O/P PORT latch → D0-D7

Enable

Data to Display
INTERFACING OF OUTPUT DEVICE

I/P device

O/P device

Tri state Buffer

AD0- AD7

D0- D7

I/O/ M

WR

Active High

Not Using

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High

Active High
The address lines are decoded to generate a signal that is active when the particular port is being accessed.

An IOWR signal is generated by combining the IO/M and the WR signals from the microprocessor.
The Latch will be enabled when:

- $WR = 0$
- $IO/M = 1$
- The address on $A_8 - A_{15} = FFH$
INTERFACING OF OUTPUT DEVICE

8085

LED

R

GND
Finally, to write the program:

MVI A, Data ;load the data to be displayed

OUT FF ;send the data to output port FF

HLT ;End
INTERFACING OF OUTPUT DEVICE

Common Cathode

Common Anode

0: OFF
1: ON

0: ON
1: OFF

+Vcc
## INTERFACING OF OUTPUT DEVICE

<table>
<thead>
<tr>
<th>D13 D12 D11 D10</th>
<th>State</th>
<th>Digit</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0 0</td>
<td>Empty</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td>0 0 0 1</td>
<td>Low</td>
<td>L</td>
<td></td>
</tr>
<tr>
<td>0 0 1 0</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>0 0 1 1</td>
<td>Half</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>0 1 0 0</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>0 1 0 1</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>0 1 1 0</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>0 1 1 1</td>
<td>High</td>
<td>H</td>
<td></td>
</tr>
<tr>
<td>1 0 0 0</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>1 0 0 1</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>1 0 1 0</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>1 0 1 1</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>1 1 0 0</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>1 1 0 1</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>1 1 1 0</td>
<td>Error</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>1 1 1 1</td>
<td>Full</td>
<td>F</td>
<td></td>
</tr>
</tbody>
</table>
INTERFACING AS MEMORY
MAPPED I/O
Memory Mapped I/O addressing

<table>
<thead>
<tr>
<th>Address</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>00000H</td>
<td></td>
</tr>
<tr>
<td>00001H</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>0067H</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>FFFFFH</td>
<td></td>
</tr>
</tbody>
</table>
Memory Mapped I/O Interfacing

(a) Separate I/O and memory space
(b) Memory-mapped I/O
(c) Hybrid
Memory Mapped I/O Interfacing

- I/O Devices and memory share the same address space.
- Each I/O Device is assigned a unique set of addresses.
- When the processor places a particular address on the address lines, the device recognizing this address
- The processor requests either a read or a write operation, and the requested data is transferred over the data lines.
Memory Mapped I/O Interfacing

• Input and output transfer using memory mapped I/O are not limited to the accumulator.
• Same of 8085 instructions can be used for memory mapped I/O ports.

• MOV r, m move the connects of input port whose address is available in (H,L) register pair to any internal register.

• LDA address load the acc with the content of the input port whose address is available as a second and third byte of the instruction.
Memory Mapped I/O Interfacing

Diagram:
- Memory/IO
- Address Bus
- Address Code Decoder
- Input Device
- Output Device
- CPU
- RAM
- CS Interface 1
- CS Interface 2
- Data Bus
PPI INTERFACING
Peripheral Interfacing

How 8255 works
A/D Interfacing

8255A

<table>
<thead>
<tr>
<th>PA₃</th>
<th>1</th>
<th>40</th>
<th>PA₄</th>
</tr>
</thead>
<tbody>
<tr>
<td>PA₂</td>
<td>2</td>
<td>39</td>
<td>PA₅</td>
</tr>
<tr>
<td>PA₁</td>
<td>3</td>
<td>38</td>
<td>PA₆</td>
</tr>
<tr>
<td>PA₀</td>
<td>4</td>
<td>37</td>
<td>PA₇</td>
</tr>
<tr>
<td>RD</td>
<td>5</td>
<td>36</td>
<td>WR</td>
</tr>
<tr>
<td>CS</td>
<td>6</td>
<td>35</td>
<td>RESET</td>
</tr>
<tr>
<td>Vₛₛ</td>
<td>7</td>
<td>34</td>
<td>D₀</td>
</tr>
<tr>
<td>A₁</td>
<td>8</td>
<td>33</td>
<td>D₁</td>
</tr>
<tr>
<td>A₀</td>
<td>9</td>
<td>32</td>
<td>D₂</td>
</tr>
<tr>
<td>PC₇</td>
<td>10</td>
<td>31</td>
<td>D₃</td>
</tr>
<tr>
<td>PC₆</td>
<td>11</td>
<td>30</td>
<td>D₄</td>
</tr>
<tr>
<td>PC₅</td>
<td>12</td>
<td>29</td>
<td>D₅</td>
</tr>
<tr>
<td>PC₄</td>
<td>13</td>
<td>28</td>
<td>D₆</td>
</tr>
<tr>
<td>PC₃</td>
<td>14</td>
<td>27</td>
<td>D₇</td>
</tr>
<tr>
<td>PC₂</td>
<td>15</td>
<td>26</td>
<td>Vcc</td>
</tr>
<tr>
<td>PC₁</td>
<td>16</td>
<td>25</td>
<td>PB₇</td>
</tr>
<tr>
<td>PC₀</td>
<td>17</td>
<td>24</td>
<td>PB₆</td>
</tr>
<tr>
<td>PB₀</td>
<td>18</td>
<td>23</td>
<td>PB₅</td>
</tr>
<tr>
<td>PB₁</td>
<td>19</td>
<td>22</td>
<td>PB₄</td>
</tr>
<tr>
<td>PB₂</td>
<td>20</td>
<td>21</td>
<td>PB₃</td>
</tr>
</tbody>
</table>

PC₇ - PC₄

PC₃ - PC₀

PB₇ - PB₀

D₀ - D₇

8255A

8255A

(+5V)Vcc

(0V)Vₛₛ
# A/D Interfacing

<table>
<thead>
<tr>
<th>Pin</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$D_0 - D_7$</td>
<td>Data lines</td>
</tr>
<tr>
<td>RESET</td>
<td>Reset input</td>
</tr>
<tr>
<td>CS</td>
<td>Chip select</td>
</tr>
<tr>
<td>RD</td>
<td>Read control</td>
</tr>
<tr>
<td>WR</td>
<td>Write control</td>
</tr>
<tr>
<td>$A_0, A_1$</td>
<td>Internal address</td>
</tr>
<tr>
<td>$PA_7 - PA_0$</td>
<td>Port-A pins</td>
</tr>
<tr>
<td>$PB_7 - PB_0$</td>
<td>Port-B pins</td>
</tr>
<tr>
<td>$PC_7 - PC_0$</td>
<td>Port-C pins</td>
</tr>
<tr>
<td>$V_{cc}$</td>
<td>+5V</td>
</tr>
<tr>
<td>$V_{ss}$</td>
<td>0V (GND)</td>
</tr>
</tbody>
</table>
A/D Interfacing
## A/D Interfacing

<table>
<thead>
<tr>
<th>Internal Device</th>
<th>Decoder input and enable</th>
<th>Input to address pins of 8255</th>
<th>Hexa Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port-A</td>
<td>0 0 0 1</td>
<td>x x 0 0</td>
<td>10</td>
</tr>
<tr>
<td>Port-B</td>
<td>0 0 0 1</td>
<td>x x 0 1</td>
<td>11</td>
</tr>
<tr>
<td>Port-C</td>
<td>0 0 0 1</td>
<td>x x 1 0</td>
<td>12</td>
</tr>
<tr>
<td>Control Register</td>
<td>0 0 0 1</td>
<td>x x 1 1</td>
<td>13</td>
</tr>
</tbody>
</table>

*Note: Don’t care "x" is considered as zero.*
8255 control word

GROUP B
PORT C (LOWER)
1 = INPUT
0 = OUTPUT

PORT B
1 = INPUT
0 = OUTPUT

MODE SELECTION
0 = MODE 0
1 = MODE 1

GROUP A
PORT C (UPPER)
1 = INPUT
0 = OUTPUT

PORT A
1 = INPUT
0 = OUTPUT

MODE SELECTION
00 = MODE 0
01 = MODE 1
1x = MODE 2

MODE SET FLAG
1 = ACTIVE
A/D INTERFACING
A/D Interfacing

- In most of the cases, the PIO 8255 is used for interfacing the analog to digital converters with microprocessor through its ports.

- The analog to digital converters is treated as an input device by the microprocessor, that sends an initializing signal to the ADC to start the analogy to digital data conversation process.

- The process of analog to digital conversion is a slow process, and the microprocessor has to wait for the digital data till the conversion is over.
A/D Interfacing

• After the conversion is over, the ADC sends end of conversion EOC signal to inform the microprocessor that the conversion is over and the result is ready at the output buffer of the ADC.

• These tasks of issuing an SOC pulse to ADC, reading EOC signal from the ADC and reading the digital output of the ADC are carried out by the CPU using 8255 I/O ports.
A/D Interfacing

- The time taken by the ADC from the active edge of SOC pulse till the active edge of EOC signal is called as the conversion delay of the ADC.

- Successive approximation techniques and dual slope integration techniques are the most popular techniques used in the integrated ADC chip.

- The analog to digital converter chips 0808 and 0809 are 8-bit CMOS, successive approximation converters.
ADC 0804

Pin Configuration:

1. CS (Chip Select)
2. RD (Read Data)
3. WR (Write Data)
4. CLK IN (Clock Input)
5. INTR (Interrupt)
6. VIN (+) (Input Positive)
7. VIN (-) (Input Negative)
8. AGND (Analog Ground)
9. VREF/2 (Reference Input)
10. DGND (Digital Ground)
11. DB7 (MSB) (Data Output 7)
12. DB6
13. DB5
14. DB4
15. DB3
16. DB2
17. DB1
18. DB0 (LSB) (Data Output 0)
19. CLK R (Clock Return)
20. VCC (OR VREF) (Power Supply)
ADC Interfacing

- **CS**: Active low input used to activate the ADC0804 chip.

  **RD (data enable)**: Active low input used to get converted data out of the ADC0804 chip. When CS = 0, if a high-to-low pulse is applied to the RD pin, the 8-bit digital output shows up at the D0-D7 data pins.

  **WR (start conversion)**: Active low input used to inform the ADC0804 to start the conversion process. If CS = 0 when WR makes a low-to-high transition, the ADC0804 starts converting the analog input value of Vin to an 8-bit digital number. When the data conversion is complete, the INTR pin is forced low by the ADC0804.
ADC Interfacing

- **CLK IN and CLK R**: Connect to external capacitor and resistor for self-clocking, \( f = 1/(1.1RC) \). The clock affects the conversion time and this time cannot be faster than 110 micros.

**INTR (end of conversion)** This is an active low output pin. When the conversion is finished, it goes low to signal the CPU that the converted data is ready to be picked up. After INTR goes low, we make CS = 0 and send a high-to-low pulse to the RD pin to get the data out of the ADC0804 chip.
ADC Interfacing

- **Vin (+) and Vin (-):** These are the differential analog inputs where $\text{Vin} = \text{Vin (}+) - \text{Vin (-)}$. Often the Vin (-) pin is connected to ground and the Vin (+) pin is used as the analog input to be converted to digital.

- **VCC:** This is the +5V power supply. It is also used as a reference voltage when the Vref/2 (pin 9) input is open.

<table>
<thead>
<tr>
<th>Vref/2 (V)</th>
<th>Vin (V)</th>
<th>Step Size (mV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not connected</td>
<td>0 to 5</td>
<td>$5/255 = 19.60$</td>
</tr>
<tr>
<td>2.0</td>
<td>0 to 4</td>
<td>$4/255 = 15.69$</td>
</tr>
<tr>
<td>1.5</td>
<td>0 to 3</td>
<td>$3/255 = 11.76$</td>
</tr>
<tr>
<td>1.28</td>
<td>0 to 2.56</td>
<td>$2.56/255 = 10.04$</td>
</tr>
<tr>
<td>1.0</td>
<td>0 to 2</td>
<td>$2/255 = 7.84$</td>
</tr>
<tr>
<td>0.5</td>
<td>0 to 1</td>
<td>$1/255 = 3.92$</td>
</tr>
</tbody>
</table>
ADC Interfacing

- \( V_{\text{ref}/2} \) : Input voltage pin used for the reference voltage. If this pin is open, the analog input voltage for the ADC is ranged from 0 to 5 volts. This is an optional input pin. It is used only when the input signal range is small. When pin 9 is at 2V, the range is 0-4V, i.e., Twice the voltage at pin 9. Pin 6 (V+), Pin 7 (V-) : The actual input is the difference in voltages applied to these pins. The analogue input can range from 0 to 5V.

**D0 – D7 output PINs of ADC:** D0 – D7 are the digital data output pins. These are the tri-state buffered and the converted data is accessed only when CS = 0 and RD is forced low. The output voltage:
ADC Interfacing
A/D Interfacing [0808 ]
A/D Interfacing

- The ADC 0808 is 8-channel 8-bit ADC chip. It has 8 analog inputs i.e. IN0-IN7.
A/D Interfacing
<table>
<thead>
<tr>
<th>LABEL</th>
<th>OPCODE</th>
<th>OPERAND</th>
<th>COMMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVI</td>
<td>A,98H</td>
<td></td>
<td>Initialize 8255</td>
</tr>
<tr>
<td>OUT</td>
<td>83H (CWR)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MVI</td>
<td>A,0BH</td>
<td></td>
<td>Selects Channel IN3 and gets High to Low SOC and ALE signals</td>
</tr>
<tr>
<td>OUT</td>
<td>82H(PORT C)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MVI</td>
<td>A,03H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT</td>
<td>82H(PORT C)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L1:</td>
<td>IN</td>
<td>82H(PORT C)</td>
<td>Checks EOC</td>
</tr>
<tr>
<td>RAL</td>
<td></td>
<td></td>
<td>Is conversion complete? if no then jump to L1</td>
</tr>
<tr>
<td>JNC</td>
<td>L1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MVI</td>
<td>A,01H</td>
<td></td>
<td>Is conversion complete? if yes then enable Output</td>
</tr>
<tr>
<td>OUT</td>
<td>81H(PORT B)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IN</td>
<td>80H(PORT A)</td>
<td></td>
<td>Read data from Port A</td>
</tr>
<tr>
<td>STA</td>
<td>2050H</td>
<td></td>
<td>Store Data</td>
</tr>
<tr>
<td>HLT</td>
<td></td>
<td></td>
<td>Stop</td>
</tr>
</tbody>
</table>
A/D Interfacing

Timing Diagram of ADC 0808
D/A INTERFACING
D/A Interfacing

- The digital to analog converters convert binary number into their equivalent voltages.
- The DAC find applications in areas like digitally controlled gains, motors speed controls, programmable gain amplifiers etc.
# D/A Interfacing

<table>
<thead>
<tr>
<th></th>
<th>DAC0800</th>
<th>DAC0802</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolution (Bits)</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>DAC: Channels</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Architecture</td>
<td>Multiplying DAC</td>
<td>Multiplying DAC</td>
</tr>
<tr>
<td>Interface</td>
<td>Parallel</td>
<td>Parallel</td>
</tr>
<tr>
<td>Output Type</td>
<td>Current</td>
<td>Current</td>
</tr>
<tr>
<td>Output Range Min. (V or mA)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Output Range Max. (V or mA)</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Settling Time (μs)</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>Reference: Type</td>
<td>Ext</td>
<td>Ext</td>
</tr>
<tr>
<td>Rating</td>
<td>Catalog</td>
<td>Catalog</td>
</tr>
<tr>
<td>Pin/Package</td>
<td>16PDIP, 16SOIC</td>
<td>16SOIC</td>
</tr>
<tr>
<td>Approx. Price (US$)</td>
<td>0.51</td>
<td>1ku</td>
</tr>
<tr>
<td>Operating Temperature Range (°C)</td>
<td>0 to 70</td>
<td>0 to 70</td>
</tr>
</tbody>
</table>
D/A Interfacing

Reference Voltage

Digital Value → DAC → Analog Voltage

$V_{CC}$ +12V

$B_1 - B_8$

$V_{ref}$

$-12V$

$V_{ref}$

$V_{out}$

$2R$

$V$

$\text{COMP}$
D/A Interfacing
D/A Interfacing
# D/A Interfacing

<table>
<thead>
<tr>
<th>MEMORY ADDRESS</th>
<th>OPCODE</th>
<th>MNEMONICS</th>
</tr>
</thead>
<tbody>
<tr>
<td>9100</td>
<td>3E 80</td>
<td>MVI A,80</td>
</tr>
<tr>
<td>9102</td>
<td>D3 23</td>
<td>OUT \textit{CNTRL}</td>
</tr>
<tr>
<td>9104</td>
<td>3E 00</td>
<td>START: MVI A, 00</td>
</tr>
<tr>
<td>9106</td>
<td>D3 20</td>
<td>LOOP1: OUT PORTA</td>
</tr>
<tr>
<td>9108</td>
<td>3C</td>
<td>INR A</td>
</tr>
<tr>
<td>9109</td>
<td>C2 06 91</td>
<td>JNZ LOOP1</td>
</tr>
<tr>
<td>910C</td>
<td>C3 04 91</td>
<td>JMP \textit{START}</td>
</tr>
</tbody>
</table>
D/A Interfacing

- AD 7523 8-bit Multiplying DAC: This is a 16 pin DIP, multiplying digital to analog converter, containing R-2R ladder for D-A conversion along with single pole double thrown NMOS switches to connect the digital inputs to the ladder.

- Supply range is from +5V to +15V, while
- Vref may be anywhere between -10V to +10V.
- The maximum analog output voltage will be anywhere between -10V to +10V, when all the digital inputs are at logic high state.
D/A Interfacing

Pin Diagram of AD 7523
D/A Interfacing
D/A Interfacing
# D/A Interfacing

<table>
<thead>
<tr>
<th>DIGITAL INPUT MSB</th>
<th>LSB</th>
<th>ANALOG OUTPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>111111111</td>
<td></td>
<td>$-V_{\text{REF}} \left( \frac{255}{256} \right)$</td>
</tr>
<tr>
<td>100000001</td>
<td></td>
<td>$-V_{\text{REF}} \left( \frac{129}{256} \right)$</td>
</tr>
<tr>
<td>100000000</td>
<td></td>
<td>$-V_{\text{REF}} \left( \frac{128}{256} \right) = -\frac{V_{\text{REF}}}{2}$</td>
</tr>
<tr>
<td>011111111</td>
<td></td>
<td>$-V_{\text{REF}} \left( \frac{127}{256} \right)$</td>
</tr>
<tr>
<td>000000001</td>
<td></td>
<td>$-V_{\text{REF}} \left( \frac{1}{256} \right)$</td>
</tr>
<tr>
<td>000000000</td>
<td></td>
<td>$-V_{\text{REF}} \left( \frac{0}{256} \right) = 0$</td>
</tr>
</tbody>
</table>