A
Project Report On

INTERFACING 8255 & ADC/DAC WITH 8085 μPROCESSOR

Microprocessor & Interfacing (2141001)

BACHELOR OF ENGINEERING
in
ELECTRONICS AND COMMUNICATION ENGINEERING

By

Abhishek Budhbhatti 140080111011
Dharati Bhimani 140080111012
Chaitanya Tejaswi 140080111013
Hemi Choudhary 140080111015
Darshi Contractor 140080111016

Under The Guidance of
Dr. Bhargav C Goradiya
Professor, ET Department.

ELECTRONICS & TELECOMMUNICATION ENGINEERING
DEPARTMENT
BVM ENGINEERING COLLEGE
GUJARAT TECHNOLOGICAL UNIVERSITY
VALLABH VIDYANAGAR-388120

Academic Year- 2015-16

CERTIFICATE

This is to certify that the project report entitled **"INTERFACING 8255 & ADC/DAC WITH 8085 μPROCESSOR",** submitted by **Abhishek Budhbhatti** (140080111011), **Dharati Bhimani** (140080111012), **Chaitanya Tejaswi** (140080111013), **Hemi Choudhary** (140080111015), **Darshi Contractor** (140080111016**)** in the subject of the ***Microprocessor & Interfacing (2141001)*** for the Bachelor of Engineering in Electronics and Communication of BVM Engineering College, Vallabh Vidyanagar, Gujarat Technological University, is the record of work carried out by them under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination.

Under The Guidance Of
Dr. Bhargav C Goradiya
Professor, ET Department.

ELECTRONICS & TELECOMMUNICATION ENGINEERING DEPARTMENT
BVM ENGINEERING COLLEGE
GUJARAT TECHNOLOGICAL UNIVERSITY
VALLABH VIDYANAGAR-388120
Academic Year- 2015-16

8255 BLOCK DIAGRAM

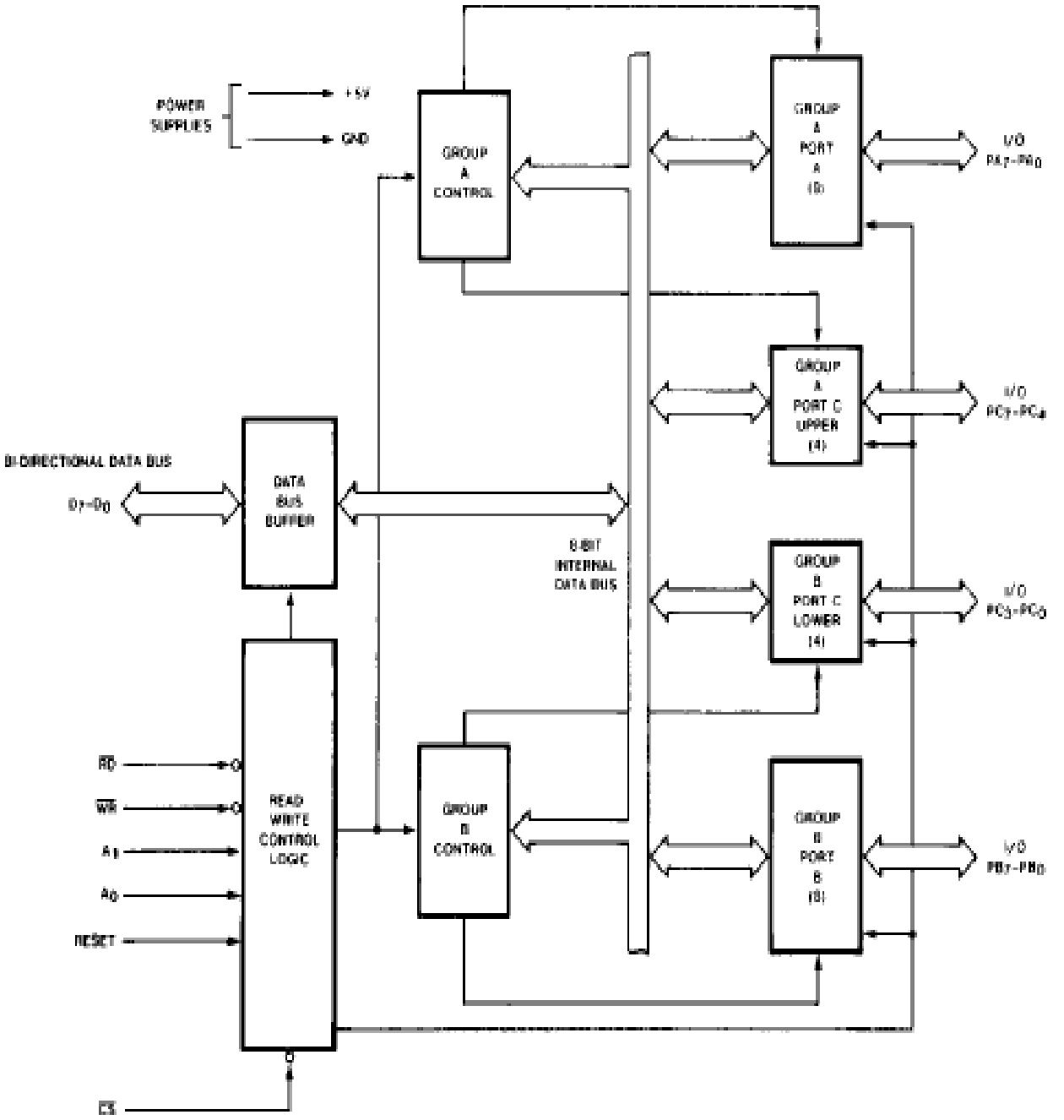


Figure 1. 82C55A Block Diagram

8255 FUNCTIONAL DESCRIPTION

General

1. The 82C55A is a programmable peripheral interfacedevice designed for use in Intel microcomputer systems.
2. Its function is that of a general purpose I/Ocomponent to interface peripheral equipment to the microcomputer system bus. The functional configurationof the 82C55A is programmed by the systemsoftware so that normally no external logic is necessaryto interface peripheral devices or structures.

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interfacethe 82C55A to the system data bus. Data istransmitted or received by the buffer upon executionof input or output instructions by the CPU. Controlwords and status information are also transferredthrough the data bus buffer.

Read/Write and Control Logic

1. The function of this block is to manage all of theinternal and external transfers of both Data and Control or Status words. It accepts inputs from theCPU Address and Control busses and in turn, issuescommands to both of the Control Groups.
2. Group A and Group B ControlsThe functional configuration of each port is programmed by the systems software. In essence, the “CPU ``outputs" a control word to the 82C55A.
3. The control word contains information such as ``mode",``bit set", ``bit reset", etc., that initializes the functionalconfiguration of the 82C55A.
4. Each of the Control blocks (Group A and Group B)accepts ``commands" from the Read/Write ControlLogic, receives ``control words" from the internaldata bus and issues the proper commands to its associatedports.
   Control Group A - Port A and Port C upper (C7±C4)
   Control Group B - Port B and Port C lower (C3±C0)
5. The control word register can be both written andread as shown in the address decode table in thepin descriptions. Figure 6 shows the control wordformat for both Read and Write operations. Whenthe control word is read, bit D7 will always be a logic``1", as this implies control word mode information.

Ports A, B, and C

The 82C55A contains three 8-bit ports (A, B, and C).
All can be configured in a wide variety of functionalcharacteristics by the system software but each hasits own special features or ``personality" to furtherenhance the power and flexibility of the 82C55A.

Port A. One 8-bit data output latch/buffer and one8-bit input latch buffer. Both ``pull-up'' and ``pulldown''bus hold devices are present on Port A.

Port B. One 8-bit data input/output latch/buffer.Only ``pull-up'' bus hold devices are present on Port B.

Port C. One 8-bit data output latch/buffer and one8-bit data input buffer (no latch for input). This portcan be divided into two 4-bit ports under the modecontrol. Each 4-bit port contains a 4-bit latch and itcan be used for the control signal outputs and statussignal inputs in conjunction with ports A and B. Only``pull-up'' bus hold devices are present on Port C.

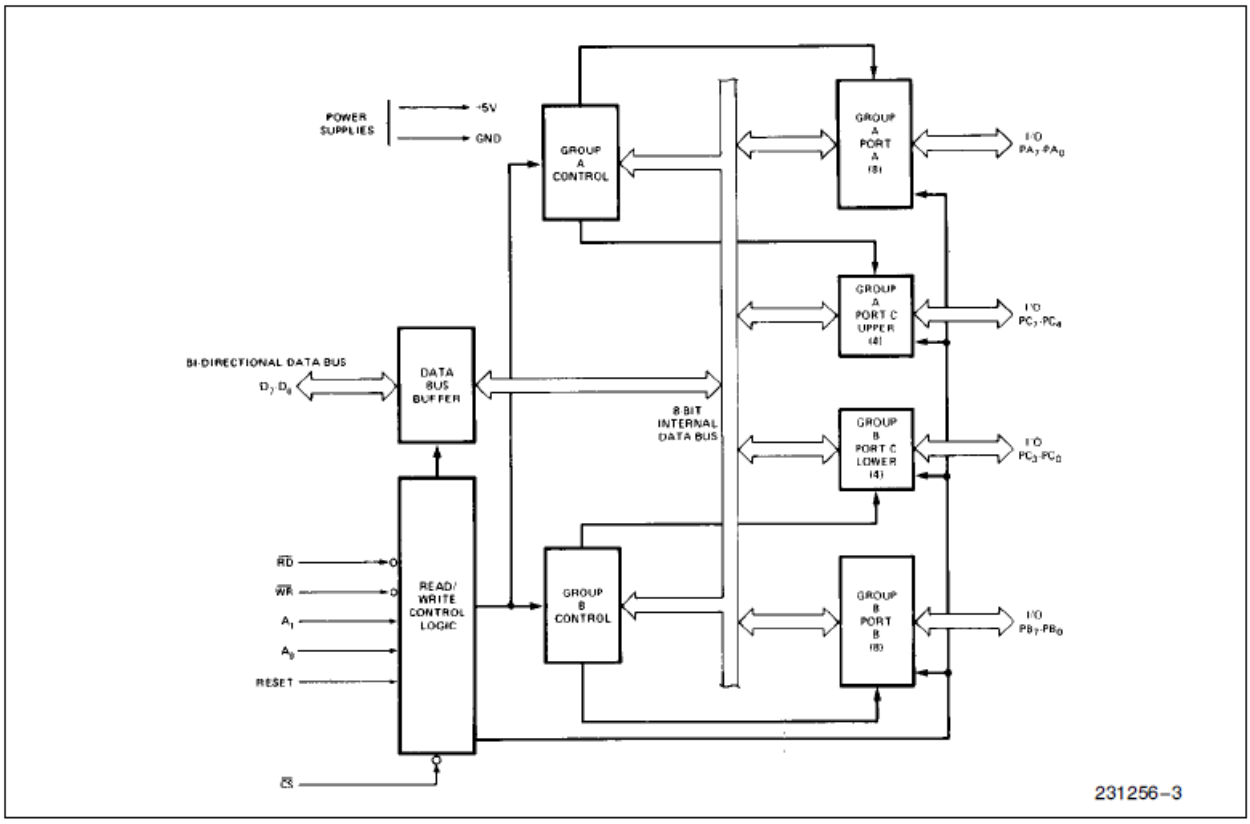


Figure 3. 82C55A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

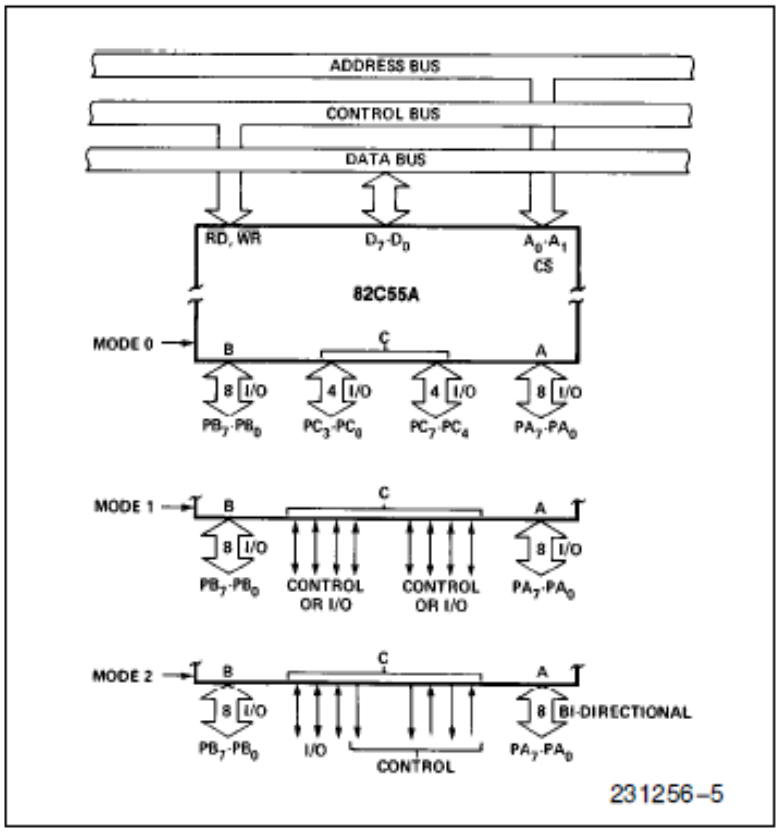
8255 OPERATIONAL DESCRIPTION

Mode Selection

1. There are three basic modes of operation that canbe selected by the system software:
   Mode 0 - Basic input/output
   Mode 1 -Strobed Input/output
   Mode 2 - Bi-directional Bus

2. When the reset input goes ``high" all ports will be set to the input mode with all 24 port lines held at a logic``one" level by the internal bus hold devices. After the reset is removed the 82C55A can remain in the input mode with no additionalinitialization required. This eliminates the needfor pullup or pulldown devices in ``all CMOS" designs. During the execution of the system program,any of the other modes may be selected by using a single output instruction. This allows a single82C55A to service a variety of peripheral devices with a simple software maintenance routine.

3. The modes for Port A and Port B can be separatelydefined, while Port C is divided into two portions asrequired by the Port A and Port B definitions. All ofthe output registers, including the status flip-flops,will be reset whenever the mode is changed. Modesmay be combined so that their functional definitioncan be ``tailored" to almost any I/O structure. Forinstance; Group B can be programmed in Mode 0 tomonitor simple switch closings or display computational results, Group A could be programmed inMode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.



4. The mode definitions and possible mode combinations may seem confusing at first but after a cursoryreview of the complete device operation a simple,logical I/O approach will surface. The design of the82C55A has taken into account things such as efficientPC board layout, control signal definition vs PClayout and complete functional flexibility to supportalmost any peripheral device with no external logic.Such design represents the maximum use of theavailable pins.

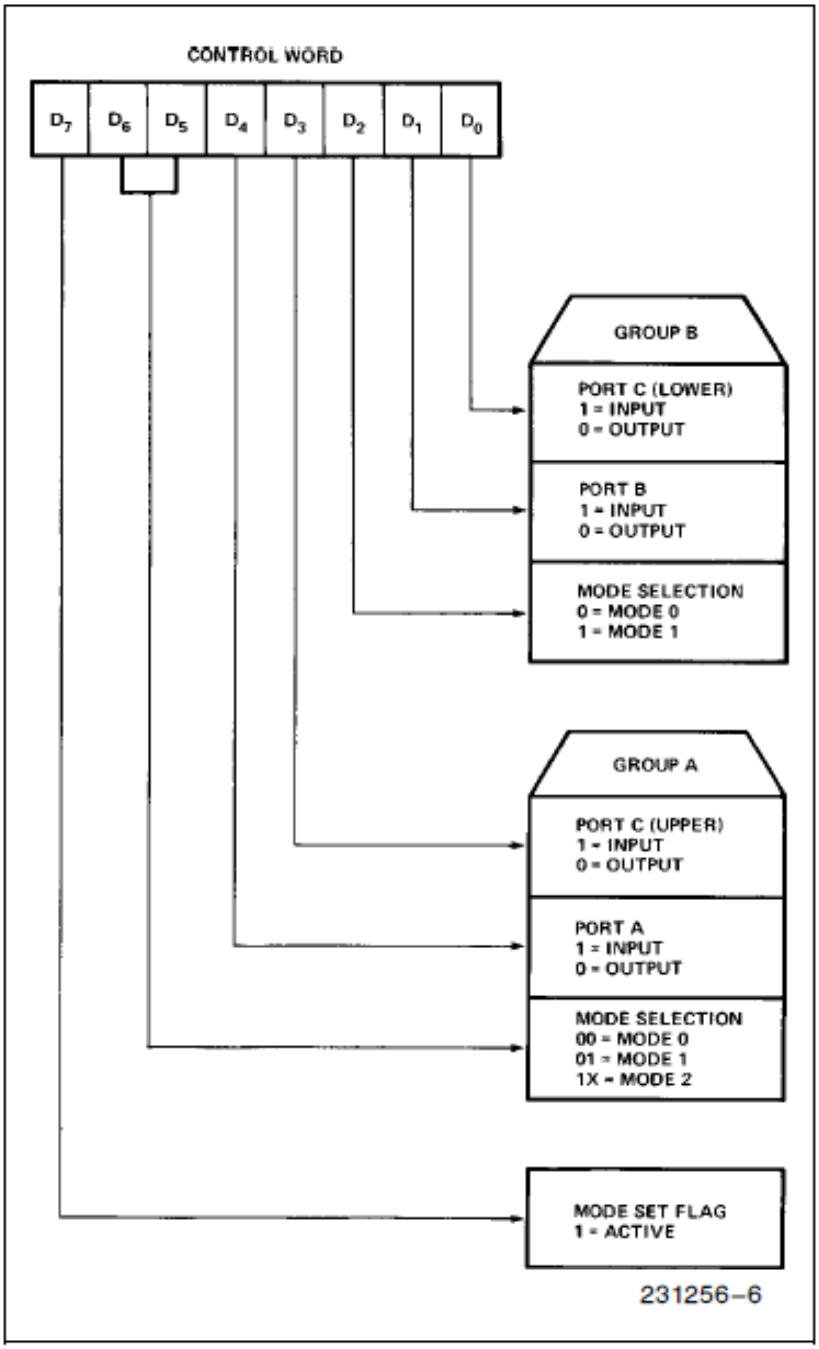


Figure 6. Mode Definition Format

Single Bit Set/Reset Feature

1. Any of the eight bits of Port C can be Set or Resetusing a single OUTput instruction. This feature reducessoftware requirements in Control-based applications.

2. When Port C is being used as status/control for PortA or B, these bits can be set or reset by using the Bit-Set/Reset operation just as if they were data outputports.

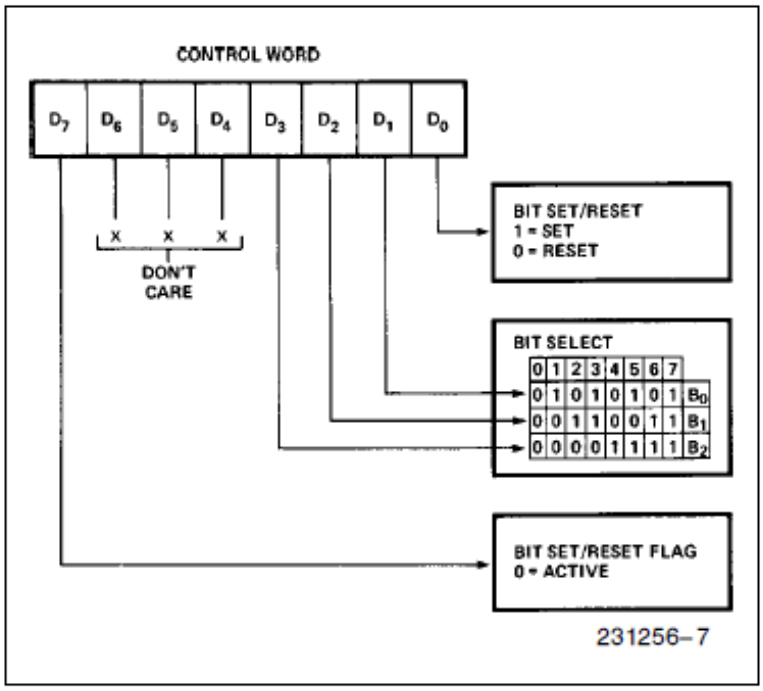


Figure 7. Bit Set/Reset Format

Interrupt Control Functions

1. When the 82C55A is programmed to operate inmode 1 or mode 2, control signals are provided thatcan be used as interrupt request inputs to the CPU.The interrupt request signals, generated from port C,can be inhibited or enabled by setting or resettingthe associated INTE flip-flop, using the bit set/resetfunction of port C.

2. This function allows the Programmer to disallow orallow a specific I/O device to interrupt the CPU withoutaffecting any other device in the interrupt structure.

   INTE flip-flop definition:
   (BIT-SET) - INTE is SET - Interrupt enable
   (BIT-RESET) - INTE is RESET - Interrupt disable

3. Note: All Mask flip-flops are automatically reset duringmode selection and device Reset.

INTERFACING ANALOG TO DIGITAL DATA CONVERTERS

1. This topic is aimed at the study of 8-bit analog to digital converters and their interfacing with 8085. In most of the cases, the PIO 8255 is used for interfacing the analog to digital converters with a microprocessor. This section will emphasize the interfacing techniques of analog to digital converters with 8255.

2. The function of an A/D converter is to produce a digital word which represents the magnitude of some analog voltage or current. The specifications for an A/D converter are very similar to those for D/A converter. The resolution of an A/D converter refers to the number of bits in the output binary word. An 8-bit converter for example has a resolution of 1part in 256. Accuracy and linearity specifications have the same meaning for an A/D converter as they do for a D/A converter. Another important specification for an ADC is its conversion time. This is simply the time it takes the converter to produce a valid output binary code for an applied input voltage. When we refer to a converter as high speed, we mean that it has a short conversion time.

3. The analog to digital converter is treated as an input device by the microprocessor that sends an initializing signal to the ADC to start the analog to digital data conversation process. The start of conversion signal is a pulse of a specific duration. 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. After the conversion is over, the ADC sends end of
4. 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 using8255 I/O ports.

5. The time taken by the ADC from the active edge of SOC pulse (the edge at which the conversion process actually starts) till the active edge of EOC signal is called as the conversion delay of the ADC. Or broadly speaking the time taken by the converter to calculate the equivalent digital data output from the instant of the start of conversion is called conversion delay. It may range any where from a few microseconds in case of fast
6. ADCs to even a few hundred milliseconds in case of slow ADCs. A number of ADCs are available in the market, The selection of ADC for a particular application is done, keeping in mind the required speed, resolution range of operation, power supply requirements, sample and hold device requirements and the cost factors are considered.

7. The available ADCs in the market use different conversion techniques for the conversion of analog signals to digital signals. Parallel converter or flash

converter, Successive approximation and dual slope integration techniques are the most popular techniques used in the integrated ADC chips.

8. Whatever may be the technique used for conversion, a general algorithm for ADC interfacing contains the following steps:

1) Ensure the stability of analog input, applied to the ADC.
2) Issue start of conversion (SOC) pulse to ADC.
3) Read end of conversion (EOC) signal to mark the end of conversion process.
4) Read digital data output of the ADC as equivalent digital output.

9. It may be noted that analog input voltage must be constant at the input of the ADC right from the start of conversion till the end of conversion to get correct results. This may be ensured by a sample and hold circuit which samples the analog signal and holds it constant for specified time duration. The microprocessor may issue a hold signal to the sample and hold circuit. If the applied input changes before the complete conversion process is over, the digital equivalent of the analog input calculated by the ADC may not be correct.

ADC 0808/0809

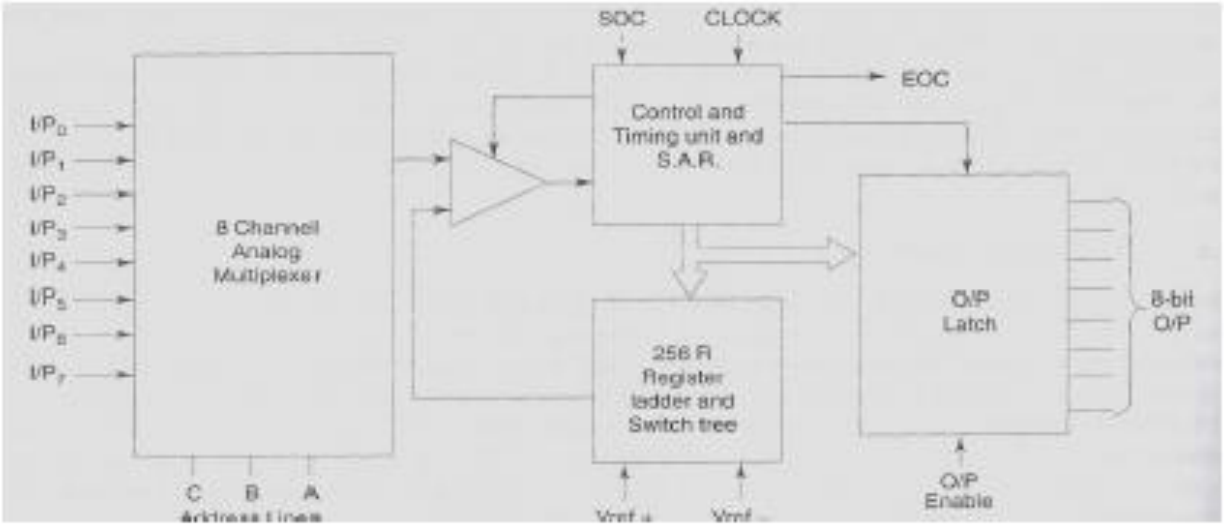


Fig. 1. Block Diagram of ADC 0808/0809

10. The analog to digital converter chips 0808 and 0809 are 8-bit CMOS, ***successive approximation converters***. Successive approximation technique is one of the fast techniques for analog to digital conversion. The conversion delay is 100 μs at a clock frequency of 640 kHz, which is quite low as compared to other converters. These converters do not need any external zero or full scale adjustments as they are already taken care of by internal circuits.

11. These converters internally have a 3:8 analog multiplexer so that at a time eight different analog inputs can be connected to the chips. Out of these eight inputs only one can be selected for conversion by using address lines ADD A, ADD B and ADD C, as shown. Using these address inputs, multichannel data acquisition systems can be designed using a single ADC. The CPU may drive these lines using output port lines in case of multichannel applications. In case of single input applications, these may be hard wired to select the proper input.

12. These are unipolar analog to digital converters, i.e. they are able to convert only positive analog input voltages to their digital equivalents. These chips do not contain any internal sample and hold circuit. If one needs a sample and hold circuit for the conversion of fast, signals into equivalent digital quantities, it has to be externally connected at each of the analog inputs.

Analog I/P selected	Address lines		
	C	B	A
I/P 0	0	0	0
I/P 1	0	0	1
I/P 2	0	1	0
I/P 3	0	1	1
I/P 4	1	0	0
I/P 5	1	0	1
I/P 6	1	1	0
I/P 7	1	1	1

INTERFACING ADC WITH 8085 PROCESSOR

AIM:

To write a program to initiate ADC and to store the digital data in memory

PROGRAM:

```
        MVI     A,10
        OUT     C8
        MVI     A,18
        OUT     C8
        MVI     A,10
        OUT     D0
        XRA     A
        XRA     A
        XRA     A
        MVI     A,00
        OUT     D0
LOOP:   IN      D8
        ANI     01
        CPI     01
        JNZ     LOOP
        IN      C0
        STA     4150
        HLT
```

OBSERVATION:

Compare the data displayed at the LEDs with that stored at location 4150

RESULT:

Thus the ADC was initiated and the digital data was stored at desired location

INTERFACING DIGITAL TO ANALOG ONVERTERS

1. The digital to analog converters convert binary numbers into their analog equivalent voltages or currents. Several techniques are employed for digital to analog conversion.
    1) Weighted resistor network
    2) R-2R ladder network
    3) Current output D/A converter
2. The DAC find applications in areas like digitally controlled gains, motor speed control, programmable gain amplifiers, digital voltmeters, panel meters, etc. D/A converter have many applications besides those where they are used with a microcomputer.
3. In a compact disk audio player for example a 14-or16-bit D/A converter is used to convert the binary data read off the disk by a laser to an analog audio signal. Most speech synthesizer integrated circuits contain a D/A converter to convert stored binary data words into analog audio signals.
4. Characteristics
    1. Resolution: It is a change in analog output for one LSB change in digital input.It is given by: $R = \frac{1}{2^n} \times V_{ref}$.

       For an 8-bit DAC, $R = \frac{1}{2^8} \times 5V = 39.06mV$.
    2. Settling time: It is the time required for the DAC to settle for a full scale code change.

DAC 0800

1. DAC0800 is a monolithic 8-bit DAC manufactured by Nationalsemiconductor.
2. It has settling time around 100ms
3. It can operate on a range of power supply voltage i.e. from 4.5V to+18V. Usually the supply V+ is 5V or +12V. The V- pin can be keptat a minimum of -12V.
4. Resolution of the DAC is 39.06mV

INTERFACING DAC WITH 8085

AIM:

To interface DAC with 8085 to demonstrate the generation of square, saw tooth and triangular wave.

APPARATUS REQUIRED:

- 8085 Trainer Kit
- DAC Interface Board

THEORY:

DAC 0800 is an 8 – bit DAC and the output voltage variation is between – 5V and + 5V.The output voltage varies in steps of 10/256 = 0.04 (appx.). The digital data input and the corresponding output voltages are presented in the Table1.

Input Data in HEX	Output Voltage
00	- 5.00
01	- 4.96
02	- 4.92
...	...
7F	0.00
...	...
FD	4.92
FE	4.96
FF	5.00

Referring to Table1, with 00 H as input to DAC, the analog output is – 5V. Similarly, with FF H as input, the output is +5V. Outputting digital data 00 and FF at regular intervals, to DAC, results in different wave forms namely square, triangular, etc,. The port address of DAC is 08 H.

ALGORITHM:

(a) Square Wave Generation

1. Load the initial value (00) to Accumulator and move it to DAC
2. Call the delay program
3. Load the final value(FF) to accumulator and move it to DAC
4. Call the delay program.
5. Repeat Steps 2 to 5

(b) Saw tooth Wave Generation

1. Load the initial value (00) to Accumulator
2. Move the accumulator content to DAC
3. Increment the accumulator content by 1.
4. Repeat Steps 3 and 4.

(c) Triangular Wave Generation

2. Load the initial value (00) to Accumulator
3. Move the accumulator content to DAC
4. Increment the accumulator content by 1.
5. If accumulator content is zero proceed to next step. Else go to step 3.
6. Load value (FF) to Accumulator
7. Move the accumulator content to DAC
8. Decrement the accumulator content by 1.
9. If accumulator content is zero go to step2. Else go to step 7.

PROGRAM:

(a) Square Wave Generation

```
START:  MVI   A,00
        OUT   Port address of DAC
        CALL  DELAY
        MVI   A,FF
        OUT   Port address of DAC
        CALL  DELAY
        JMP   START

DELAY:  MVI   B,05
L1:     MVI   C,FF
L2:     DCR   C
        JNZ   L2
        DCR   B
        JNZ   L1
        RET
```

(b) Saw tooth Wave Generation

```
START:  MVI   A,00
L1:     OUT   Port address of DAC
        INR   A
        JNZ   L1
        JMP   START
```

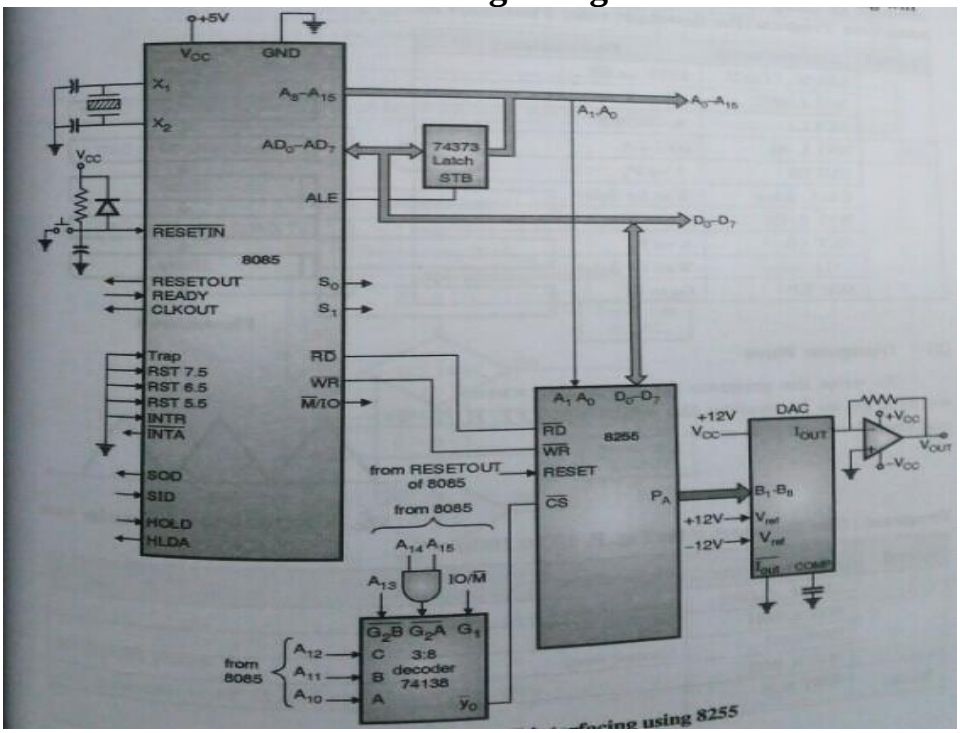
(c) Triangular Wave Generation

```
START:  MVI   L,00
L1:     MOV   A,L
        OUT   Port address of DAC
        INR   L
        JNZ   L1
        MVI   L,FF
L2:     MOV   A,L
        OUT   Port address of DAC
        DCR   L
        JNZ   L2
        JMP   START
```

RESULT:

Thus the square, triangular and saw tooth wave form were generated by interfacing DAC with 8085 trainer kit.

D/A Interfacing Using 8255:



A/D Interfacing Using 8255: