Computer Engineering Department
Course Name: Microprocessor Lab Number: 10636392
Lab Report Grading Sheet

Instructor: Dr.Manar Qamhieh	Experiment #: 5
Academic Year: 2023/2024	Experiment Name: ADC0809 analog-to-digital conversion
Semester:2nd	

Students	
1-Salsabeel Dwaikat	2-Doaa Yasin Jararaa
3-Wala' Essam Ashqar	4-
Performed on: 5/3/024	Submitted on:12/3/2024

Report's Outcomes

ILO __ =( ) %	ILO __ =( ) %	ILO __ =( ) %	ILO __ =( ) %	ILO __ =( ) %

Evaluation Criterion	Grade	Points
Abstract answers to the questions: "What did you do? How did you do it? What did you find?"	0.5	
Introduction and Theory Sufficient, clear, and complete statement of objectives. In addition, it Presents sufficiently the theoretical basis.	1.5	
Apparatus/ Procedure Apparatus sufficiently described to enable another experimenter to identify the equipment needed to experiment. The procedure is sufficiently described.	2	
Experimental Results and Discussion (In-Lab Worksheet) Crisp explanation of experimental results. Comparison of theoretical predictions to experimental results, including discussion of accuracy and error analysis in some cases.	4	
Conclusions and Recommendations Conclusions summarize the major findings from the experimental results with adequate specificity. Recommendations are appropriate in light of the conclusions. Correct grammar.	1	
Appearance The title page is complete, page numbers are applied, content is well organized, correct spelling, fonts are consistent, good visual appeal.	1	
Total	10	

Objectives:

in this Experiment, we have a couple of goals to achieve

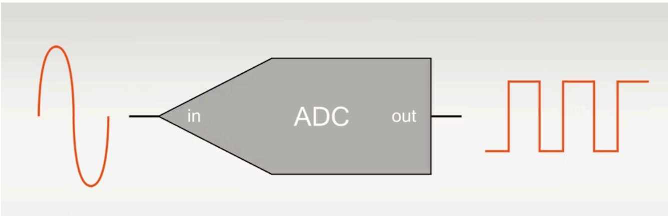
- Understand the basic principles of Analog-to-Digital Conversion (ADC)
- Grasp the use of ADC0809 Converter
- Study the internal structure of the converter.

introduction:

ADC0809 is an 8-bit A/D Converter with an 8-channel multiplexer, which means It uses a successive-approximation conversion technique; so we can connect 8 signals to one converter and interface them by successive-approximation and multiplexing.

It converts analog Input voltages in the range 0-Vcc to 8-bit digital code.

digital circuits can interface with the real world by encoding an analogue signal into a binary code.



an 8-bit ADC will resolve one part in 255, (28 – 1). Thus an analog to a digital converter takes an unknown continuous analog signal and converts it into an “n”- bit binary number of 2n bits.

An example of the analog signal is our voice, we are going to use a potentiometer since it gives continuous readings, it can be also connected to a sensor.

Tools and equipment:

- MML8086K
- ADC IC
- 8255 IC
- MML8086K Software: dice8088
- 7 Segments Display

Procedure:

In order to read the A/D converter from port A and display the analog value (voltage) on the seven-segment display, we are required to develop an assembly program.
To show the output voltage (0-VCC) and one fraction, we must interface the 7-segment display. To do this, we must interface with the 8255:
Address of the control word register: 0FF2BH

Additionally, Port B = 0FF29H, address>Port A = 0FF28H for the 7-seg DATA.
Furthermore, the word format for control is 80H because they are both outputs.
Wires served as the manual's circuit for the interface.

Q- How do I calculate the external voltage of 8-bits?

A- Following this relationship:

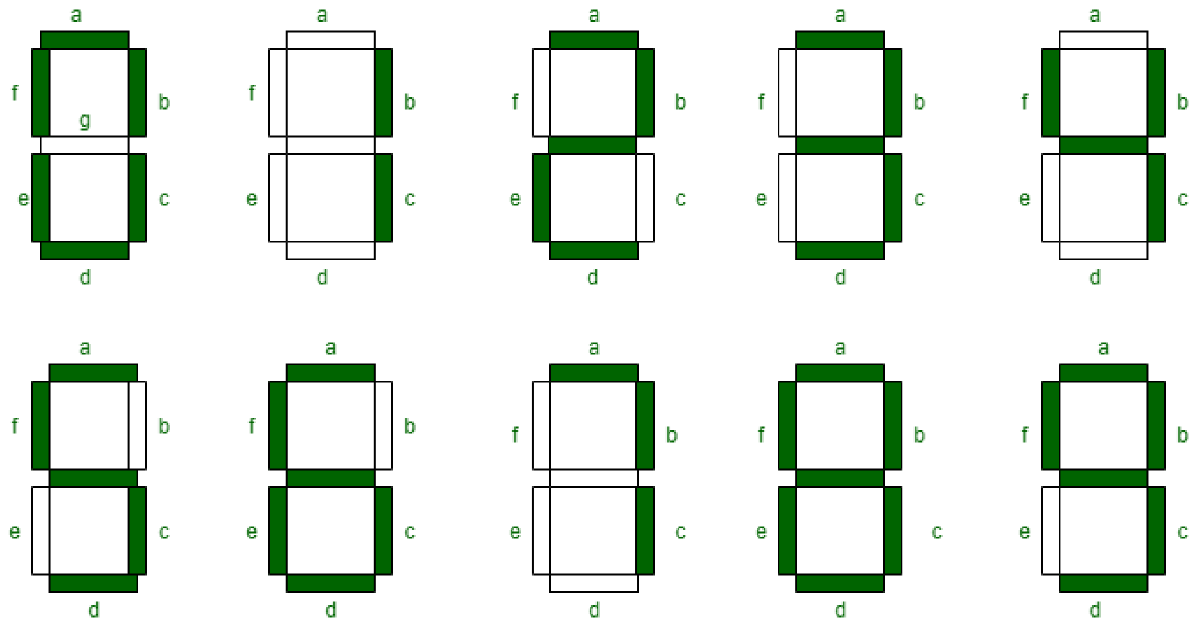
$$Data_{(DC)} = (Data_{(AC)} * 2^n)/V_{ref}$$

Then V_{ref} = 5 Volt and $2^8=256$

$$Data_{(DC)} = (Data_{(AC)} * 256)/5$$

it's equal: $Data_{(DC)} = (Data_{(AC)})/51$

To find the Array declaration:



Code:

First, we need to start the A/D conversion by sending 0x00 to ADC0809 port address. This value sends CS and WR signals and channel addresses.

The ADC main port address is 80h

Before reading the converted result from the ADC0809 port address, we have to wait for the conversion to be done (write a delay code).

```
CODE SEGMENT
ASSUME CS:CODE
ORG 2000H

START:
MOV DX, 0FF2BH
MOV AX ,80H
OUT DX, AX

MAIN:
MOV DX, 0FF80H
MOV AL,0
OUT DX , AL
PUSH CX
MOV CX ,500H
DELAY3: NOP
LOOP DELAY3
POP CX
```

Then we read the analog signal from the ADC 8000H, And since we have an 8-bit bus, the values read vary (0-255),We need to convert it CC reference5) as 5 is our V-into voltage (0(255/51=5) so we need to divide the read value by 5:

```
MOV DX , 0FF80H
IN AL, DX
MOV AH,0 ;we need total AX for division
MOV BL, 51
DIV BL  ;convert to volt
MOV CH,AH
MOV AH,0
MOV CL, AL
```

We want to display each one on a different 7-seg:

- First, we sent the address – location- of the 7-segment:
- then we sent the value but we needed XLAT to convert it into 7-seg code:

```
MOV DX, 0FF29H  ; port B
MOV AL,11101111B ;7SEG
OUT DX, AL  ;the real value -quotient-
MOV AL,CL

MOV BX, OFFSET ARRAY
XLAT
MOV DX, 0FF28H ;DATA port A
OUT DX,AL


JMP MAIN
ARRAY DB 0C0H, 0F9H, 0A4H, 0B0H, 99H,
92H
END START


CODE ENDS
```

fraction display it’s pretty much the same but we added another array(7-segment with dp) and handled the fraction in the code:

```
CODE SEGMENT
ASSUME CS:CODE
ORG 2000H

START:
MOV DX, 0FF2BH
MOV AX ,80H
OUT DX, AX

MAIN:
MOV DX, 0FF80H
MOV AL,0
OUT DX , AL
PUSH CX
MOV CX ,500H
DELAY3: NOP
LOOP DELAY3
POP CX


MOV DX , 0FF80H
IN AL, DX
MOV AH , 0
MOV BL, 51
DIV BL
MOV CH,AH
MOV AH,0
MOV CL, AL

MOV DX, 0FF29H
MOV AL,11101111B
OUT DX, AL
```

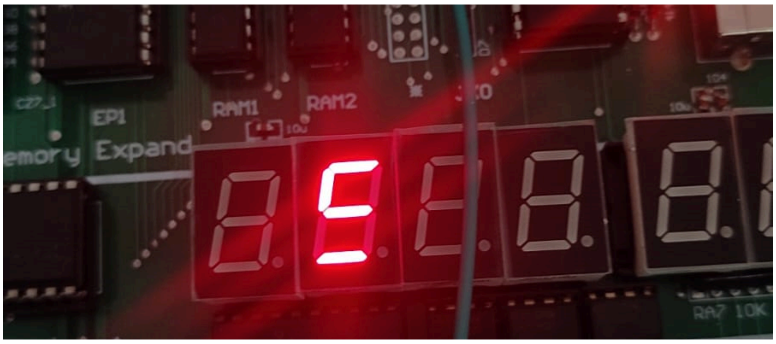
fraction display using a sequence of steps that include dividing an input value by 51 and then using a 7-segment display to show the remainder and quotient as distinct digits.

Divisional Function: 51 is divided by the input value that was read from a device using the IN command. The accumulator (AX) is divided by the value in BL (51), and the result is put in AL and the remainder in AH by the DIV instruction. As a result, the input value is divided into the fractional (remainder) and integer (quotient) portions. Getting ready for the display.

Next, the remainder (AH) and quotient (AL) are ready for presentation. The code first transfers the remainder of the value from AH to CH (to save it) and then clears AH to make AX ready for any additional operations that may be required because the DIV instruction uses AX for both input and output.
Two numbers that will be shown independently on the 7-segment displays are the quotient, which is now in CL, and the remainder, which has been moved to CH.

```
MOV AL,CL
MOV BX, OFFSET ARRAY2
XLAT
MOV DX, 0FF28H
OUT DX,AL

PUSH CX
MOV CX ,500H
DELAY33: NOP
LOOP DELAY33
POP CX

MOV DX, 0FF29H
MOV AL,11110111B
OUT DX, AL
MOV AL,CH

MOV BX, OFFSET ARRAY
XLAT
MOV DX, 0FF28H
OUT DX,AL

JMP MAIN
ARRAY DB 0C0H, 0F9H, 0A4H, 0B0H, 99H,
92H,82H,0F8H, 80H, 98H ;DP IS OFF
ARRAY2 DB 40H, 79H, 24H, 30H, 19H, 12H, 02H, 78H,
0H ,18H; DP IS ON
END START

CODE ENDS
```

output:



Conclusions:

We understood the basic principles of Analog-to-Digital Conversion (ADC) and how to interface it with an input signal that is analog and continues and show the output digital value as the voltage from GND to VCC and display it on a 7-segment display.