An-Najah National University
Faculty of Engineering and IT

جامعة النجاح الوطنية
كلية الهندسة وتكنولوجيا المعلومات

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

Instructor: Dr. Aladdin Masri	Experiment #: 6
Academic Year: 2019/2020	Experiment: ADC0809
Semester: Summer Semester	

Students	
1- Mohammad Badawi	2- Taher Anaya
3-	4-
Performed on: 24th of August	Submitted on: 3rd of September

Report's Outcomes

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

Evaluation Criterion	Grade	Points
Abstract answers of 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 to Presents sufficiently the theoretical basis.	1.5	
Apparatus/ Procedure Apparatus sufficiently described to enable another experimenter to identify the equipment needed to conduct the experiment. Procedure 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 appropriate in light of conclusions. Correct grammar.	1	
Appearance Title page is complete, page numbers applied, content is well organized, correct spelling, fonts are consistent, good visual appeal.	1	
Total	10	

Abstract:

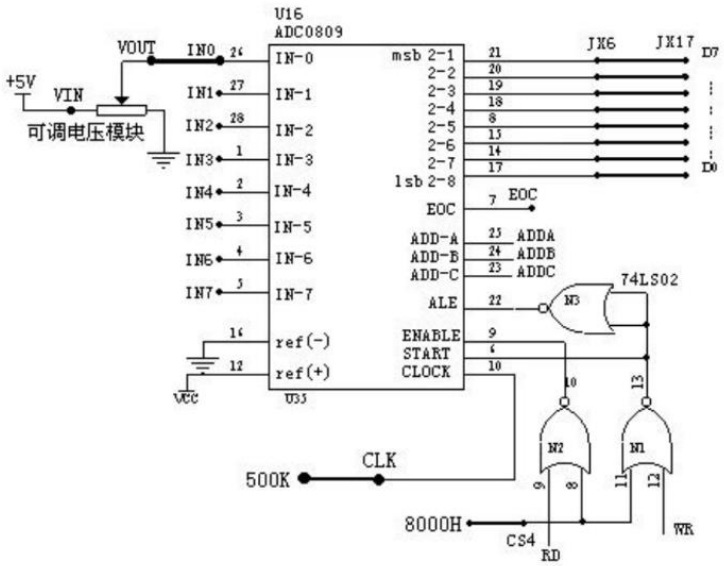
In this experiment, we will be using the ADC0809 to display the voltage value read from the potentiometer on the seven segment displays including the fraction.

Objectives:

- To learn about the ADC0809 digital-to-analog convertor.
- To learn how to use the ADC0809.

Procedure:

The ADC0809 is a digital-to-analog convertor that converts a certain analog signal into its equivalent 8 bit digital value depending the V_{ref} values on the ADC0809.



In our case, $V_{ref}(-)$ will be the GND, and $V_{ref}(+)$ will be the VCC (5 V). This means that if the analog value at a certain point was equal to $V_{ref}(+)$, then the equivalent digital value will be the maximum (0FFH), and if the analog value at a certain point was equal to $V_{ref}(-)$, the then the equivalent digital value will be the minimum (00H). Any point in between, for example $V_{ref}(+)/2$, will be a value between 0FFH and 00H (in our case it will be 7FH).

In this experiment, an ADC is connected to a potentiometer, where the potentiometer gives an analog signal that lies between 0 and 5V. The ADC will convert this value into its digital equivalent. Then, we will display the converted value on the seven segment displays by returning the value to its mathematical analog form.

Procedure: *(cont.)*

The ADC0809 will do most of the work, as it will convert the analog value on its own. Our job is to read the value from the ADC0809 on address 8000H, return it to its analog value, then display the decoded analog value on the seven segment displays using the 8255.

First of all, we will construct an array that has all the decoded values from 0 to 9, as we will use the integer value of the analog signal to access a certain element in that array:

```
66  ARR DB 0C0H, 0F9H, 0A4H, 0B0H, 99H, 92H, 82H, 0F8H, 80H, 90H
```

Now, we need to configure the 8255, port B will select the seven segment display and port A will carry the data, we will also point SI to our array:

```
 7  MOV DX, 0FF2BH
 8  MOV AL, 80H
 9  OUT DX, AL
10
11  LEA SI, ARR
```

Now we need to send 0 to our ADC0809 at address 8000H so it can start converting the analog value for us, then we will wait for a small portion of time by calling a delay procedure:

```
13  L1:
14  MOV DX, 8000H
15  MOV AL, 0
16  OUT DX, AL
17
18  CALL DELAY
```

Now, we will read the converted value and store it in AL. Since the value is ranged between 0 – 255, if we need to convert it back to range 0 – 5, we will divide the digital value by 51. The integer value will be stored in AL, and the reminder will be stored in AH:

```
20  MOV DX, 8000H
21  IN AL, DX
22  MOV AH, 0
23  MOV BL, 51
24  DIV BL
```

First of all, we will display the integer value. We will select on of the seven segments (sending 0F7H to port B in our case), then we will use the value in AL to access the corresponding decoded value inside the array that we defined earlier (note that we moved the value of AX into BX, otherwise the value of AX will be lost after selecting the seven segment), we will also call a small delay procedure to ensure no issues will occur:

Procedure: *(cont.)*

Notes:

- Line 32 will store the reminder value in AH because we will set BH to 0.
- XORing AL with 80H will make the decimal point visible without making a new array.

```
26  MOV BX, AX
27
28  MOV DX, 0FF29H
29  MOV AL, 0F7H
30  OUT DX, AL
31
32  MOV AX, BX
33
34  MOV DX, 0FF28H
35  MOV BH, 0
36  MOV AL, [SI + BX]
37  XOR AL, 80H
38  OUT DX, AL
39
40  CALL DELAY
```

Now we need to display the fraction value stored in AH on the following seven segment display. First, let us select the following seven segment:

```
42  MOV DX, 0FF29H
43  MOV AL, 0FBH
44  OUT DX, AL
```

Now, to convert the reminder into an integer, first we will multiply it by 10, then we will divide it by 51. After that, we will use the integer value to access its corresponding decoded pattern from the array. Finally we will repeat the whole process:

```
46  MOV DX, 0FF28H
47  MOV AL, AH
48  MOV AH, 0
49  MOV BL, 10
50  MUL BL
51  MOV BL, 51
52  DIV BL
```

```
53  MOV BL, AL
54  MOV BH, 0
55  MOV AL, [SI + BX]
56  OUT DX, AL
57
58  JMP L1
```

Conclusion:

In the end, we learned how does the ADC0809 works and how it can be used with the microprocessor. We also learned one of the infinite usages of the ADC0809.