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 #: 5
Academic Year: 2019/2020	Experiment: 8279 Keyboard / Display Interface
Semester: Summer Semester	

Students	
1- Mohammad Badawi	2- Taher Anaya
3-	4-
Performed on: 20th 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 8279 device to display our group number on the seven segment displays, display a 0-9 counter, and finally, use the keypad to display different symbols on the seven segment displays.

Objectives:

- To be familiar with the 8279 and its usages.
- To interface a seven segment display array with the CPU using the 8279.
- To interface a 4x4 keypad with the CPU using the 8279.

Procedure:

This experiment is divided into 3 parts:

1- In the first part, we will show our group number on the seven segment displays "GP. 1".
2- In the second part, we will display a 0-9 BCD counter on the seven segment displays.
3- In the last part, will be using the 8279 to interface the keypad provided by the kit with the CPU, to display certain symbols on the seven segments.

The 8279 is a programmable device that that can interface a keyboard/display with the CPU.

First of all, we need to configure our device, so it can perform in the way we want. The command word contains of 8 bits, were the most significant 3-bits determine the behavior of the device:

D_7	D_6	D_5	Function
0	0	0	Keyboard/Display Mode Set
0	0	1	Program Clock
0	1	0	Read FIFO/Sensor RAM
0	1	1	Read Display RAM
1	0	0	Write Display RAM
1	0	1	Display Write Inhibit/Blanking
1	1	0	Clear
1	1	1	End Interrupt/Error Mode Set

Procedure: *(cont.)*

In our experiment, we will work with the Keyboard/Display Mode "000", but we also need to configure the clock "001", and we will also clear our device "110". The selection address is 8001H and the data address is 8000H.

Keyboard/Display Mode:

The keyboard display mode will allow us to select different option:

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	0	0	D	D	K	K	K

For D_4 and D_3:

D_4	D_3	Display Option
0	0	Eight 8-bit character display with left entry
0	1	Sixteen 8-bit character display with left entry
1	0	Eight 8-bit character display with right entry
1	1	Sixteen 8-bit character display with right entry

We will be using the first mode "00". This mode will allow us to enter our 8 bit characters from left to right.

For D_2, D_1 and D_0:

D_2	D_1	D_0	Keyboard Option
0	0	0	Encoded Scan Keyboard with 2-key lockout
0	0	1	Decoded Scan Keyboard with 2-key lockout
0	1	0	Encoded Scan Keyboard with N-key roll over
0	1	1	Decoded Scan Keyboard with N-key roll over
1	0	0	Encoded Scan Sensor Matrix
1	0	1	Decoded Scan Sensor Matrix
1	1	0	Strobed Input Encoded Display Scan
1	1	1	Strobed Input Decoded Display Scan

We will be choosing "000". This will allow us to activate the selection lines "S_L" by sending logic 1 (Active High), and it prevents 2 keys from being recognized if they were pressed at the same time.

```
6   ; KEYPAD CONFIGURATION
7   MOV DX, 8001H
8   MOV AL, 0
9   OUT DX, AL
```

Procedure: *(cont.)*

Program Clock:

The 8279 needs an internal clock as the device scans keys and performs key debouncing.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	0	1	P	P	P	P	P

The values from D_4 to D_0 determine the scale factor. In this experiment, we will be working at frequency of 100 KHz, that means our scale factor will be 18 “10010”.

```
11  ; PROGRAM CLOCK CONFIGURATION
12  MOV AL, 00110010B
13  OUT DX, AL
```

Clear:

Before using the device, we need to clear the display RAM/FIFO:

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	1	0	C_D	C_D	C_D	C_F	C_A

- C_A clears both the display RAM and FIFO.
- C_F clears FIFO status and resets the IRQ line.
- C_D C_D C_D clears all rows of the display RAM.

```
15  ; CLEAR CONFIGURATION
16  MOV AL, 11011111B
17  OUT DX, AL
```

To ensure our configuration is ready, we will call a delay procedure:

```
51  DELAY PROC NEAR USES CX
52        MOV CX, 1000
53        L1:
54        NOP
55        LOOP L1
56        RET
57  DELAY ENDP
```

The configuration process is the same for all parts.

Procedure: *(cont.)*

Part 1:

In this part, we will display “GP. 1” on the seven segment display.

First of all, we need to select which display we will be working with, our displays are addressed from 85H to 80H from left to right. To select a certain display, we need to send the selection code to the address 8001H.

After selecting the seven segment display, we can now send the letter/number on the data bus (address 8000H).

The seven segment displays have a different decoding pattern than we normally know:

d	c	b	a	dp	g	f	e

So, if we want to display the letter G, we need to set both b and dp to high “00101000”. This will be the value that will be sent to the data bus at address 8000H.

```
21  MOV DX, 8001H
22  MOV AL, 84H
23  OUT DX, AL
24
25  MOV DX, 8000H
26  MOV AL, 00101000B
27  OUT DX, AL
28
29  CALL DELAY
```

Next, we need to select the next seven segment “83H” and send the letter P with dp active “11000000”.

```
31  MOV DX, 8001H
32  MOV AL, 83H
33  OUT DX, AL
34
35  MOV DX, 8000H
36  MOV AL, 11000000B
37  OUT DX, AL
38
39  CALL DELAY
```

Procedure: *(cont.)*

Part 1: (*cont.)*

Finally, we need to select the next display and send the number 1 to the databus "10011111".

```
41  MOV DX, 8001H
42  MOV AL, 82H
43  OUT DX, AL
44
45  MOV DX, 8000H
46  MOV AL, 10011111B
47  OUT DX, AL
48
49  CALL DELAY
```

Part 2:

In this part, we will be displaying a 1 digit 0-9 BCD counter on the seven segment.

The way this can be done is pretty simple, we define an array with all numbers already decoded, then, every second passes, we just need to select the following element, and repeat.

This array contains all the digits from 0 to 9

```
48  ARR DB 0CH, 9FH, 4AH, 0BH, 99H, 29H, 28H, 8FH, 8H, 9H
```

After the configuration, we need to select which display we will displaying the numbers on:

```
21  MOV DX, 8001H
22  MOV AL, 82H
23  OUT DX, AL
```

Now we need to make a counter that iterates over all the elements in the array each second.

Procedure: *(cont.)*

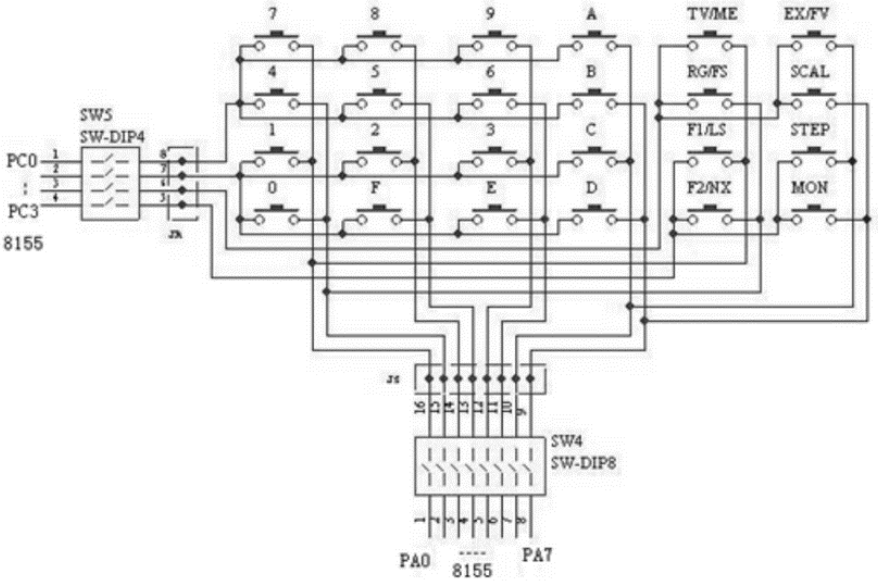
Part 2: (*cont.*)

```
25  MOV DX, 8000H
26  L2:
27  MOV CX, 10
28  LEA SI, ARR
29  L3:
30  MOV AL, [SI]
31  OUT DX, AL
32  INC SI
33  PUSH CX
34  CALL DELAY
35  POP CX
36  LOOP L3
37  JMP L2
```

Part 3: (*cont.*)

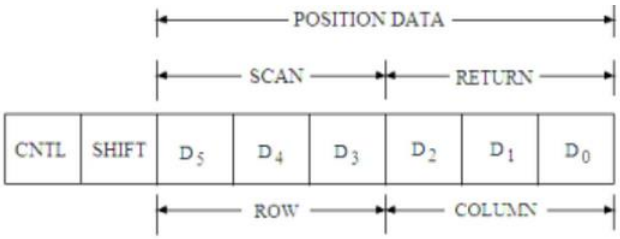
In this part, we will be using the 4x4 keypad on the kit to display certain symbols/numbers on the kit.

Once a certain key is pressed, the 8279 will respond with a 6-bit data that determine the location of the pressed key:



Procedure: *(cont.)*

Part 3: (*cont.*)



From the given graph, we can see that the key arrangement is a little bit odd. The first column has 8 different rows were each row has its own character:

Row	0	1	2	3	4	5	6	7
Character	7	4	8	5	9	6	A	B

The second column:

Row	0	1	2	3	4	5	6	7
Character	1	0	2	F	3	E	C	D

So, our idea is to make 2 different arrays, were each array determines a certain column, then we can use the row number to access a certain element in that array:

```
65  ARR1 DB 8FH, 99H, 8H, 29H, 9H, 28H, 88H, 38H
66  ARR2 DB 9FH, 0CH, 4AH, 0E8H, 0BH, 68H, 6CH, 1AH
```

To work with that idea, first we need to check whether a key is pressed:

```
20  L1:
21  NO:
22  MOV DX, 8001H
23  IN AL, DX
24  AND AL, 07H
25  JZ NO
```

Now we can read the 6-bit code that represents the key location:

```
27  MOV DX, 8000H
28  IN AL, DX
```

Procedure: *(cont.)*

Part 3: (*cont.)*

The AL 8-bit register contains the 6-bit key location, we need to mask it to get rid of the first 2 bits.

```
29  AND AL, 3FH
```

Now, we will use AL to store the column number and BL to store the row number:

```
30  MOV BL, AL
31  MOV BH, 0
32  AND AL, 00000111B
33  AND BL, 00111000B
34  SAR BL, 3
```

If the first column is selected, we will make SI point to the first array, and if the second column is selected, we will make SI point to the second array:

```
35  CMP AL, 00000000B
36  JE A1
37  CMP AL, 00000001B
38  JE A2
```

```
49  A1:
50  LEA SI, ARR1
51  JMP CONT
52  A2:
53  LEA SI, ARR2
54  JMP CONT
```

Now all we need to do is select the display we will be working with, then we will output the character in the array according to the value of BL (we set BH earlier to 0, because we cannot do SI + BL as the size mismatches, so we used BX instead):

```
40  CONT:
41  MOV DX, 8001H
42  MOV AL, 82H
43  OUT DX, AL
44  MOV DX, 8000H
45  MOV AL, [SI + BX]
46  OUT DX, AL
47  JMP L1
```

Conclusion:

In the end, we learned how to use the 8279 device to interface a display/keypad with the microprocessor. Additionally, we got to use the seven segment displays on the kit to display characters/symbols.