Stony Brook University
College of Engineering and Applied Sciences
ESE 280.L05

Lab 6: Keypad Input Using 74C922 16-Key Encoder

Date: 10/13/2023

Ryan Lin (114737153)
Bench: 6
Breadboard: R2

1. Explain the difference between unconditional data transfer from a keypad and conditional data transfer from a keypad.

The difference between unconditional data transfer from a keypad vs conditional data transfer from a keypad is that unconditional data will be instantly displayed after a press but in a conditional data transfer, the data will only get displayed if there is a certain condition that is met.

2. Why is it important that the 74HC74 flip-flop serving as a "valid data" flag be cleared after reading each key scan code? What would happen in your system if this flip-flop is not cleared?

It is important that the flip-flip serves as a "valid data" flag to be cleared after each key scan code because if we don't clear it, it will be stuck at an input of "1" due to the D pin of the FF being connected to Vcc. Clearing it will reset it to 0 and will only input 1 when a key press is detected.

3. An input from VPORTC_IN provides two kinds of information, a 4-bit value corresponding to the DIP switch positions and a 4-bit key scan code corresponding to the key pressed. After inputting a byte from VPORTC_IN, what operations must be used to "remove" the unwanted data so that all you have is a right justified 4-bit scan key code in r16? Write a sequence of instructions that efficiently accomplish this.

We must use andi r16 with 0xF0 or 0b11110000 which removes any input from the switch because the switch is PC0 - PC3 and any value and with 0 is always 0. Then we shift the register to the right 4 times to make it right justified.

```
in r16, VPORTC_IN
Andi r16, 0xF0
lsr r16
lsr r16
lsr r16
lsr r16
```

4. Assume that we input a byte from VPORTC_IN and we are interested in only the binary value represented by the 4-bit DIP switch, write a sequence of instructions that inputs the data from VPORTC_IN and then leaves r16 with only the right-justified binary value represented by the 4 bit DIP switch.

in r16, VPORTC_IN
andi r16, 0b00001111

5. If you use a 0.1uF capacitor from MM74C922 pin OSC to ground, what will be the scan frequency for each key? How did you determine this value?

The scan frequency for each key would be 159.155.

f = 1/2piRC
F = 1/2pi*10k*0.1 uF
f = 159.155

6. For capacitors of values 0.1, 1 and 10 uF for the KBM capacitor at MM74C922 pin KBM to ground, what will be the debounce time provided by the MM74C922 for each key press? How did you determine this value? Discuss the responsiveness of your system with values of 0.1, 1 and 10 uF for the KBM capacitor. Particularly make note of bouncing or loss of input with these different capacitor values.

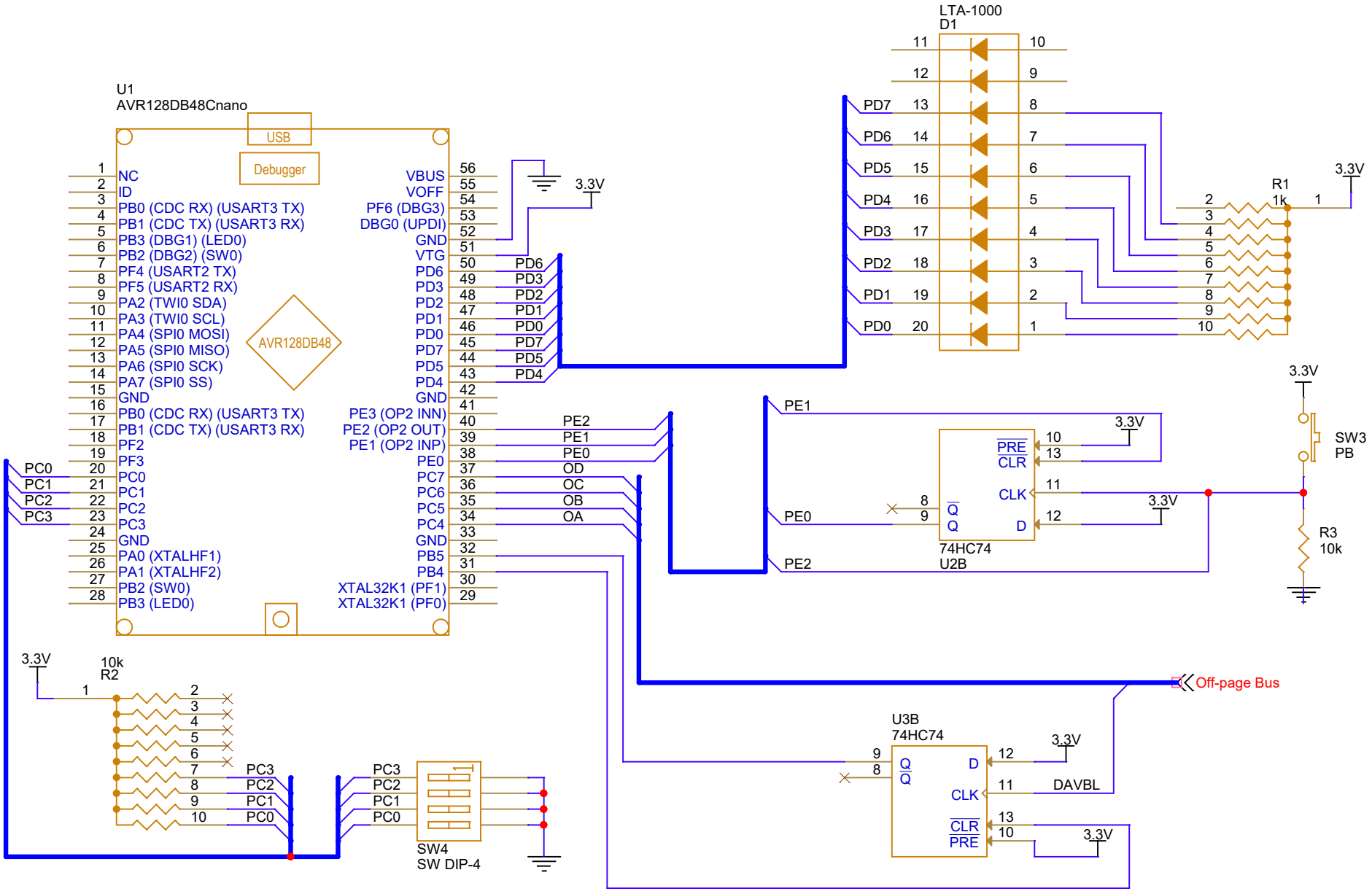
T = RC
*T = 10000 * 0.00001*
T = 0.1 s

*T = 10000 * 1 *10^-6*
T = 0.01s

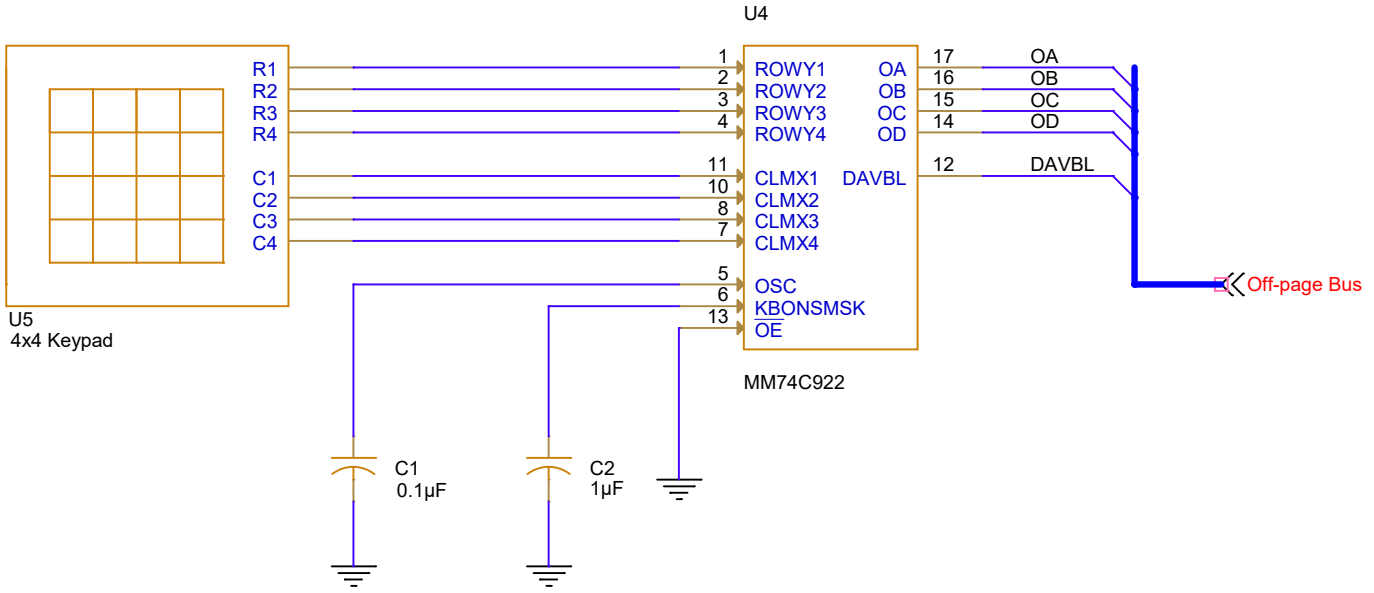
*T = 10000 * 0.1 *10^-6*
T = 1 ms
If we used a capacitor value of 0.1, it will be more responsive.

```
1  ;
2  ; keypad_mapping_test.asm
3  ;
4  ; Created: 10/13/2023 4:18:14 PM
5  ; Author : userESD
6  ;
7
8
9  ; Replace with your application code
10 start:
11     ldi r16, 0xFF ;load r16 with all 1s
12     out VPORTD_DIR, r16 ;PORTD - all pins set to output
13     out VPORTD_OUT, r16 ;turn all the led off
14
15     ;PC7 - PC4 set to input
16     cbi VPORTC_DIR, 7
17     cbi VPORTC_DIR, 6
18     cbi VPORTC_DIR, 5
19     cbi VPORTC_DIR, 4
20
21     cbi VPORTB_DIR, 5 ;PORTB5(Q flipflop) - set to input
22
23     sbi VPORTB_DIR, 4 ;Direction of FF as output
24     cbi VPORTB_OUT, 4 ;initialize flip flop (clear flipflop)
25     sbi VPORTB_OUT, 4
26
27 check_for_1:
28     sbis VPORTB_IN, 5
29     rjmp check_for_1
30
31 read_key:
32     in r18, VPORTC_IN           ;scan input from PortC into register 18
33     lsr r18                     ;right justify 4 times
34     lsr r18
35     lsr r18
36     lsr r18
37     mov r21, r18
38     rcall scan
39
40     com r21                     ;complement r21
41     OUT VPORTD_OUT, r21         ;display onto the bar graph
42     rcall clear_FF              ;clear flip flip
43     rjmp check_for_1
44
45 scan:
46     cpi r21, 16                 ;result 0-15?
47     brlo lookup                 ;if less than 15, branch to lookup
48     clc                         ;no carry clear
49     ldi r21, 0                  ;load r21 with 0
```

```
50      ret
51
52  lookup:
53      ldi ZH, high (table * 2)    ;set Z to point to start of table
54      ldi ZL, low (table *2)
55      ldi r17, $00                ;add offset to Z pointer
56      add ZL, r21
57      adc ZH, r17
58      lpm r21, Z                  ;load the value from table to r21
59      ;rcall clear_FF             ;clear flip flop
60      ret
61
62  clear_FF:
63      cbi VPORTB_IN, 4 ;initialize flip flop (clear flipflop)
64      sbi VPORTB_IN, 4 ;
65      ret
66
67  table: .db $01, $02, $03, $0F, $04, $05, $06, $0e, $07, $08, $09, $0d, $0a,
        $00, $0b, $0c
68
69
70
```



Title	Prelab 06 - Keypad Input Using 74C922 16-Key Encoder	
Size: A	Document Number: Ryan Lin - ESE 280 Fall 2023	Rev: 0.1
Date: Thursday, October 12, 2023	Sheet 1 of 2	



Title	Prelab 06 - Keypad Input Using 74C922 16-Key Encoder	
Size: A	Document Number: Ryan Lin - ESE 280 Fall 2023	Rev: 0.1
Date: Thursday, October 12, 2023	Sheet 2 of 2	