

EE443 - Embedded Systems
Experiment 3 Laboratory Report
Edge Detection and Delay Generation

Name: M. Serdar Karaman

Number: 190206038

Date: 16 - November - 2016

Objective

Taking input from button and detection transition on signal. Creating delay function with NOPs.

Experimental Work

Experiment code

```
#include <avr/io.h>
#include <util/delay.h>
#define _NOP() do { __asm__ __volatile__ ("nop"); } while (0)

void mydelay(void){
    int i = 0;
    for (i = 0; i < 1000; i++) {
        __asm__ volatile ( "NOP" );
    }
}

void change (void){
    if (( PIND & _BV(PIND1) ) == 0x00){ //SW1 and SW0 pressed (1 clock)
        _NOP(); // to make right and left shift time equal.
        if(PORTB == 0b00000001){ //(1 clock)
            _NOP();
            PORTB = 0b10000000; //Turn to start (1 clock)
        }
        else{// + jump (1 clock)
            PORTB= PORTB >> 1; //shift right (1 clock))
        }
    }
    else{ //SW1 not pressed, SW0 pressed + jump (1 clock)
        if(PORTB == 0b10000000){ //(1 clock)
            _NOP();
            PORTB = 0b00000001; //Turn to end (1 clock)
        }
        else{// + jump (1 clock)
            PORTB= PORTB << 1; //shift left (1 clock)
        }
    }
}

int main(void){
    DDRB |= 0b11111111; // all output
    DDRD |= 0b11111100; //0-1 input, others output
    PORTD |= 0b00000011; //Pull-up resistor for inputs
    PORTB = 0b00010000; //initiate
    unsigned int PDsave = 0x00;
    unsigned int PDsave_old = 0x00;
    int count = 0;

    while(1){
        PDsave_old = PDsave;
        PDsave = PIND;

        if(((PDsave_old & 0x01) == 0x01) && ((PDsave & 0x01) == 0x00)){ // 1 to 0
            transition at SW0
            change();
            count = 0;
        }
        else if (((PDsave_old & 0x01) == 0x00) && ((PDsave & 0x01) == 0x00)){ //
            no change 0 - 0
            if (count>250){ //wait until 250ms
                change();
            }
        }
    }
}
```

```

        count = 0;
    }
    count++;
}

mydelay();//1ms

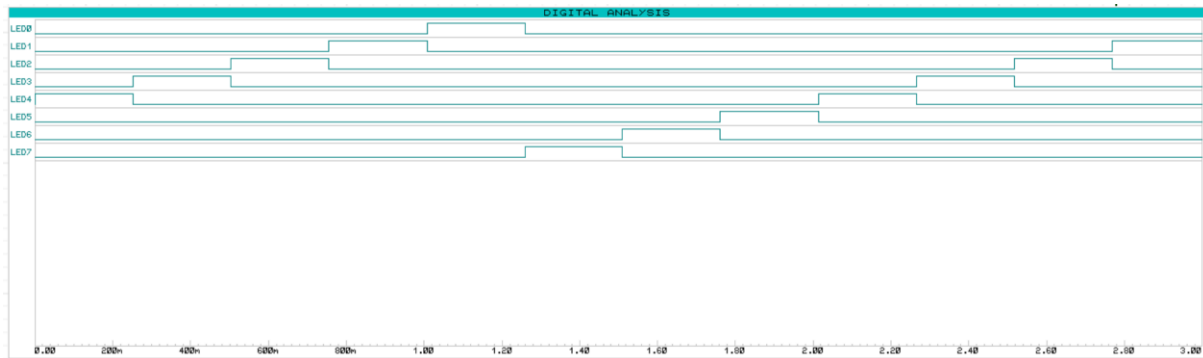
}

return 0;
}

```

When there is a transition at SW0 led will rotate. If there is no transition but SW0 is pressed more than 250ms leds will rotate again. SW1 controls the direction of rotation.

At the below, SW0 and SW1 pressed leds rotating LED7 to LED0. For each leds on time is 250ms.



LSS file

```

00000000 <.text>:
0: 0c 94 34 00    jmp     0x68 ; 0x68
4: 0c 94 3e 00    jmp     0x7c ; 0x7c
8: 0c 94 3e 00    jmp     0x7c ; 0x7c
c: 0c 94 3e 00    jmp     0x7c ; 0x7c
10: 0c 94 3e 00   jmp     0x7c ; 0x7c
14: 0c 94 3e 00   jmp     0x7c ; 0x7c
18: 0c 94 3e 00   jmp     0x7c ; 0x7c
1c: 0c 94 3e 00   jmp     0x7c ; 0x7c
20: 0c 94 3e 00   jmp     0x7c ; 0x7c
24: 0c 94 3e 00   jmp     0x7c ; 0x7c
28: 0c 94 3e 00   jmp     0x7c ; 0x7c
2c: 0c 94 3e 00   jmp     0x7c ; 0x7c
30: 0c 94 3e 00   jmp     0x7c ; 0x7c
34: 0c 94 3e 00   jmp     0x7c ; 0x7c
38: 0c 94 3e 00   jmp     0x7c ; 0x7c
3c: 0c 94 3e 00   jmp     0x7c ; 0x7c
40: 0c 94 3e 00   jmp     0x7c ; 0x7c
44: 0c 94 3e 00   jmp     0x7c ; 0x7c
48: 0c 94 3e 00   jmp     0x7c ; 0x7c
4c: 0c 94 3e 00   jmp     0x7c ; 0x7c
50: 0c 94 3e 00   jmp     0x7c ; 0x7c
54: 0c 94 3e 00   jmp     0x7c ; 0x7c
58: 0c 94 3e 00   jmp     0x7c ; 0x7c
5c: 0c 94 3e 00   jmp     0x7c ; 0x7c
60: 0c 94 3e 00   jmp     0x7c ; 0x7c
64: 0c 94 3e 00   jmp     0x7c ; 0x7c
68: 11 24         eor     r1, r1
6a: 1f be         out     0x3f, r1 ; 63
6c: cf ef         ldi     r28, 0xFF ; 255
6e: d8 e0         ldi     r29, 0x08 ; 8
70: de bf         out     0x3e, r29 ; 62

```

```

72: cd bf      out    0x3d, r28      ; 61
74: 0e 94 62 00 call    0xc4      ; 0xc4
78: 0c 94 cf 00 jmp     0x19e     ; 0x19e
7c: 0c 94 00 00 jmp     0          ; 0x0
80: 80 e0      ldi     r24, 0x00      ; 0
82: 90 e0      ldi     r25, 0x00      ; 0
84: 00 00      nop
86: 01 96      adiw    r24, 0x01      ; 1
88: 23 e0      ldi     r18, 0x03      ; 3
8a: 88 3e      cpi     r24, 0xE8      ; 232
8c: 92 07      cpc     r25, r18
8e: d1 f7      brne    -12          ; 0x84
90: 08 95      ret
92: 49 99      sbic    0x09, 1 ; 9
94: 08 c0      rjmp    .+16          ; 0xa6
96: 00 00      nop
98: 85 b1      in      r24, 0x05      ; 5
9a: 81 30      cpi     r24, 0x01      ; 1
9c: 59 f0      breq     .+22          ; 0xb4
9e: 85 b1      in      r24, 0x05      ; 5
a0: 86 95      lsr     r24
a2: 85 b9      out     0x05, r24      ; 5
a4: 08 95      ret
a6: 85 b1      in      r24, 0x05      ; 5
a8: 80 38      cpi     r24, 0x80      ; 128
aa: 41 f0      breq     .+16          ; 0xbc
ac: 85 b1      in      r24, 0x05      ; 5
ae: 88 0f      add     r24, r24
b0: 85 b9      out     0x05, r24      ; 5
b2: 08 95      ret
b4: 00 00      nop
b6: 80 e8      ldi     r24, 0x80      ; 128
b8: 85 b9      out     0x05, r24      ; 5
ba: 08 95      ret
bc: 00 00      nop
be: 81 e0      ldi     r24, 0x01      ; 1
c0: 85 b9      out     0x05, r24      ; 5
c2: 08 95      ret
c4: 84 b1      in      r24, 0x04      ; 4
c6: 8f ef      ldi     r24, 0xFF      ; 255
c8: 84 b9      out     0x04, r24      ; 4
ca: 8a b1      in      r24, 0x0a      ; 10
cc: 8c 6f      ori     r24, 0xFC      ; 252
ce: 8a b9      out     0x0a, r24      ; 10
d0: 8b b1      in      r24, 0x0b      ; 11
d2: 83 60      ori     r24, 0x03      ; 3
d4: 8b b9      out     0x0b, r24      ; 11
d6: 80 e1      ldi     r24, 0x10      ; 16
d8: 85 b9      out     0x05, r24      ; 5
da: 20 e0      ldi     r18, 0x00      ; 0
dc: 30 e0      ldi     r19, 0x00      ; 0
de: 40 e0      ldi     r20, 0x00      ; 0
e0: 50 e0      ldi     r21, 0x00      ; 0
e2: 71 e0      ldi     r23, 0x01      ; 1
e4: e0 e8      ldi     r30, 0x80      ; 128
e6: 89 b1      in      r24, 0x09      ; 9
e8: 68 2f      mov     r22, r24
ea: c9 01      movw    r24, r18
ec: 81 70      andi    r24, 0x01      ; 1
ee: 90 70      andi    r25, 0x00      ; 0
f0: 20 ff      sbrs    r18, 0
f2: 02 c0      rjmp    .+4          ; 0xf8
f4: 60 ff      sbrs    r22, 0
f6: 15 c0      rjmp    .+42         ; 0x122
f8: 89 2b      or      r24, r25
fa: 39 f4      brne    .+14         ; 0x10a
fc: 60 fd      sbrc    r22, 0
fe: 05 c0      rjmp    .+10         ; 0x10a
100: 4b 3f      cpi     r20, 0xFB      ; 251
102: 51 05      cpc     r21, r1
104: 1c f5      brge    .+70         ; 0x14c
106: 4f 5f      subi    r20, 0xFF      ; 255
108: 5f 4f      sbci    r21, 0xFF      ; 255
10a: 80 e0      ldi     r24, 0x00      ; 0
10c: 90 e0      ldi     r25, 0x00      ; 0
10e: 00 00      nop

```

```

110: 01 96      adiw  r24, 0x01      ; 1
112: 23 e0      ldi   r18, 0x03      ; 3
114: 88 3e      cpi   r24, 0xE8      ; 232
116: 92 07      cpc   r25, r18
118: d1 f7      brne  .-12      ; 0x10e
11a: 86 2f      mov   r24, r22
11c: 90 e0      ldi   r25, 0x00      ; 0
11e: 9c 01      movw  r18, r24
120: e2 cf      rjmp  .-60      ; 0xe6
122: 49 99      sbic  0x09, 1 ; 9
124: 0a c0      rjmp  .+20      ; 0x13a
126: 00 00      nop
128: 85 b1      in    r24, 0x05      ; 5
12a: 81 30      cpi   r24, 0x01      ; 1
12c: 21 f1      breq  .+72      ; 0x176
12e: 85 b1      in    r24, 0x05      ; 5
130: 86 95      lsr   r24
132: 85 b9      out   0x05, r24      ; 5
134: 40 e0      ldi   r20, 0x00      ; 0
136: 50 e0      ldi   r21, 0x00      ; 0
138: e8 cf      rjmp  .-48      ; 0x10a
13a: 85 b1      in    r24, 0x05      ; 5
13c: 80 38      cpi   r24, 0x80      ; 128
13e: 01 f1      breq  .+64      ; 0x180
140: 85 b1      in    r24, 0x05      ; 5
142: 88 0f      add   r24, r24
144: 85 b9      out   0x05, r24      ; 5
146: 40 e0      ldi   r20, 0x00      ; 0
148: 50 e0      ldi   r21, 0x00      ; 0
14a: df cf      rjmp  .-66      ; 0x10a
14c: 49 99      sbic  0x09, 1 ; 9
14e: 0a c0      rjmp  .+20      ; 0x164
150: 00 00      nop
152: 85 b1      in    r24, 0x05      ; 5
154: 81 30      cpi   r24, 0x01      ; 1
156: c9 f0      breq  .+50      ; 0x18a
158: 85 b1      in    r24, 0x05      ; 5
15a: 86 95      lsr   r24
15c: 85 b9      out   0x05, r24      ; 5
15e: 41 e0      ldi   r20, 0x01      ; 1
160: 50 e0      ldi   r21, 0x00      ; 0
162: d3 cf      rjmp  .-90      ; 0x10a
164: 85 b1      in    r24, 0x05      ; 5
166: 80 38      cpi   r24, 0x80      ; 128
168: a9 f0      breq  .+42      ; 0x194
16a: 85 b1      in    r24, 0x05      ; 5
16c: 88 0f      add   r24, r24
16e: 85 b9      out   0x05, r24      ; 5
170: 41 e0      ldi   r20, 0x01      ; 1
172: 50 e0      ldi   r21, 0x00      ; 0
174: ca cf      rjmp  .-108     ; 0x10a
176: 00 00      nop
178: e5 b9      out   0x05, r30      ; 5
17a: 40 e0      ldi   r20, 0x00      ; 0
17c: 50 e0      ldi   r21, 0x00      ; 0
17e: c5 cf      rjmp  .-118     ; 0x10a
180: 00 00      nop
182: 75 b9      out   0x05, r23      ; 5
184: 40 e0      ldi   r20, 0x00      ; 0
186: 50 e0      ldi   r21, 0x00      ; 0
188: c0 cf      rjmp  .-128     ; 0x10a
18a: 00 00      nop
18c: e5 b9      out   0x05, r30      ; 5
18e: 41 e0      ldi   r20, 0x01      ; 1
190: 50 e0      ldi   r21, 0x00      ; 0
192: bb cf      rjmp  .-138     ; 0x10a
194: 00 00      nop
196: 75 b9      out   0x05, r23      ; 5
198: 41 e0      ldi   r20, 0x01      ; 1
19a: 50 e0      ldi   r21, 0x00      ; 0
19c: b6 cf      rjmp  .-148     ; 0x10a
19e: f8 94      cli
1a0: ff cf      rjmp  .-2      ; 0x1a0

```