Lab 2: Morse Code Decoder

ESE3500: Embedded Systems & Microcontroller Laboratory
University of Pennsylvania

In this document, you'll fill out your responses to the questions listed in the Lab 2 Manual. Please fill out your name and link your Github repository below to begin. Be sure that your code on the repo is up-to-date before submission!

For all the questions that require a video, provide a link to the video (e.g. youtube, google drive, etc.).

Student Name: Hunter Stilp
Pennkey: hstilp
GitHub Repository: https://github.com/ese3500/lab-2-morse-HunterStilp

1.

```
void initialize()
{
    DDRB |= (1<<DDB1)|(1<<DDB2)|(1<<DDB3)|(1<<DDB4);//
    PORTB |= (1<<PORTB1)|(1<<PORTB2)|(1<<PORTB3)|(1<<PORTB4);//
}

int main(void)
{
    /* Replace with your application code */
    initialize();
    while (1)
    {
    }
}
```

2.

```
void initialize()
{
    DDRB |= (1<<DDB1)|(1<<DDB2)|(1<<DDB3)|(1<<DDB4);//
    DDRD = (0<<DDD7);
}

int main(void)
{
    /* Replace with your application code */
    initialize();

    while (1)
    {
        if (PIND & (1 << PIND7)) {
            PORTB |= (1<<PORTB1)|(1<<PORTB2)|(1<<PORTB3)|(1<<PORTB4);
        }
        else {
            PORTB &= ~((1<<PORTB1)|(1<<PORTB2)|(1<<PORTB3)|(1<<PORTB4));
        }

    }
}
```

```
void initialize()
{
    DDRB |= (1<<DDB1)|(1<<DDB2)|(1<<DDB3)|(1<<DDB4);//
    DDRD = (0<<DDD7);
}
```

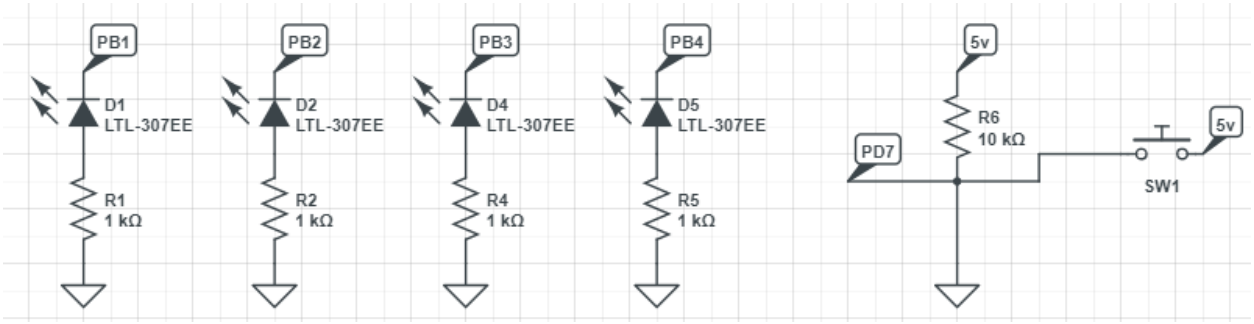
3.

```
void click(void)
{

    cycle += 1;
    if (cycle > 4)
    {
        cycle = 1;
    }

    if (cycle == 1)
    {
        PORTB &= (0<<PORTB4);
        PORTB |= (1<<PORTB1);
    }
    else if (cycle == 2)
    {
        PORTB &= (0<<PORTB1);
        PORTB |= (1<<PORTB2);
    }
    else if (cycle == 3)
    {
        PORTB &= (0<<PORTB2);
        PORTB |= (1<<PORTB3);
    }
    else if (cycle == 4)
    {
        PORTB &= (0<<PORTB3);
        PORTB |= (1<<PORTB4);
    }
}
```

```
int main(void)
{
    /* Replace with your application code */
    initialize();
    int pressed = 0;
    while (1)
    {
        if (PIND & (1 << PIND7)) {
            pressed = 1;
        }
        else
        {
            if (pressed)
            {
                click();
            }
            pressed = 0;
        }
    }
}
```

4. 

5. A advantage would be that interrupts are much more efficient than constantly looping and waiting for a event to happen. (you can do other stuff while you wait. A disadvantage is that multiple interrupts occurring at once can cause errors.
6. For a 16MHz clock, how many “ticks” are in 30ms, 200ms, and 400ms?
    a. 16Khz/ms.
    b. 30ms = 30*16khz = 480khz
    c. 200ms = 200*16khz =3.2Mhz
    d. 400ms = 400*16khz = 6.4Mhz
7. A prescaler allows us to work with a wider range of frequencies because it allows the register that records clock ticks to only increment when a certain number of true

clock frequencies occur. For example if the clock is 16Mhz and the prescaler is 1024 then the register will increment at a frequency of 15.625khz.

8. https://drive.google.com/file/d/1vgfMSCmq7Fsbc11a-sbzvV7b0gCyexsJ/view?usp=share_link
9. someday i will rule you all
10. Did not do this one