

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: Kat Wang

Pennkey: katawang

GitHub Repository: <https://github.com/e3500/lab2-katawang>

1. light up all leds

```
void Initialize() {
    DDRB |= (1 << DDB1);
    DDRB |= (1 << DDB2);
    DDRB |= (1 << DDB3);
    DDRB |= (1 << DDB4);
}

int main() {
    Initialize();

    while (1) {
        PORTB |= (1 << PORTB1);
        PORTB |= (1 << PORTB2);
        PORTB |= (1 << PORTB3);
        PORTB |= (1 << PORTB4);
    }
}
```

2. light up led when press button

```
int main() {
    DDRD &= ~(1 << DDD7);
    while(1) {
        if (PIND & (1 << PIND7)) {
            DDRB |= (1 << DDB1);
            PORTB |= (1 << PORTB1);
        } else {
            DDRB |= (1 << DDB1);
            PORTB &= ~(1 << PORTB1);
        }
    }
}
```

3. toggle leds by pressing button

```

int main() {
  DDRD |= ~(1 << DD07);
  DDRB |= (1 << DDB1);
  PORTB |= (1 << PORTB1);

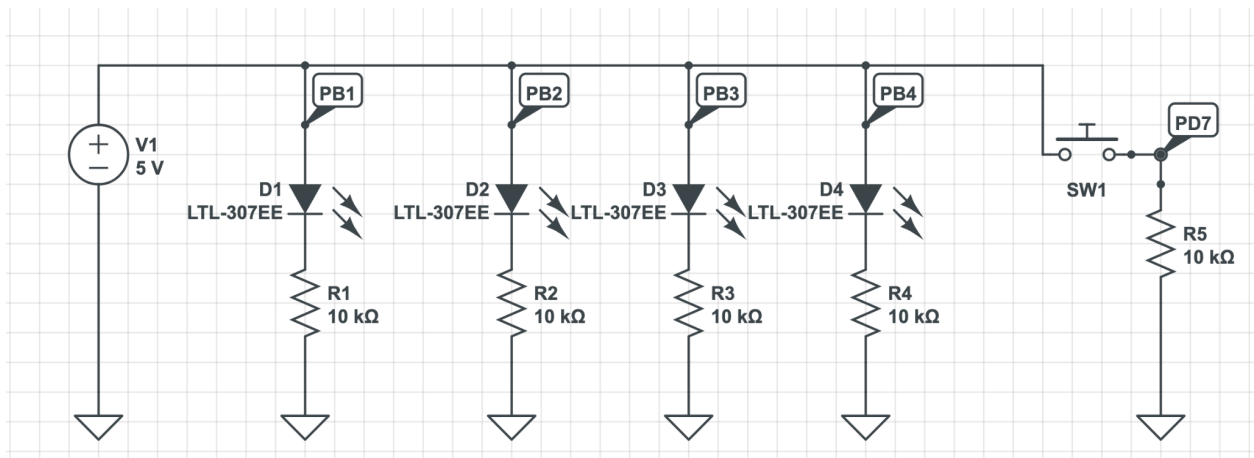
  while(1) {
    if (PIND & (1 << PIND7) && PINB & (1 << PINB1)) {
      DDRB |= (1 << DDB1);
      PORTB |= ~(1 << PORTB1);
      DDRB |= (1 << DDB2);
      PORTB |= (1 << PORTB2);
      _delay_ms( ms: 1000);
    } else if (PIND & (1 << PIND7) && PINB & (1 << PINB2)) {

      DDRB |= (1 << DDB2);
      PORTB |= ~(1 << PORTB2);
      DDRB |= (1 << DDB3);
      PORTB |= (1 << PORTB3);
      _delay_ms( ms: 1000);

    } else if (PIND & (1 << PIND7) && PINB & (1 << PINB3)) {
      DDRB |= (1 << DDB3);
      PORTB |= ~(1 << PORTB3);
      DDRB |= (1 << DDB4);
      PORTB |= (1 << PORTB4);
      _delay_ms( ms: 1000);
    } else if (PIND & (1 << PIND7) && PINB & (1 << PINB4)) {
      DDRB |= (1 << DDB4);
      PORTB |= ~(1 << PORTB4);
      DDRB |= (1 << DDB1);
      PORTB |= (1 << PORTB1);
      _delay_ms( ms: 1000);
    }
  }
}

```

4. Schematic of 4 LEDs with button switch



5. An advantage of using interrupts is that it's a lot more efficient than using polling because every single time the main function loops, you need to check state, whereas the interrupts will just interrupt the execution flow to complete the task. A disadvantage is that an interrupt is more complex and can have more latency when executed by the microcontroller.
6. $\text{ticks} = (\text{time in seconds}) * (\text{clock frequency in Hz})$
 $\text{ticks} = (30 * 10^{-3}) * (16 * 10^6) = 480,000 \text{ in } 30\text{ms}$
 $\text{ticks} = (200 * 10^{-3}) * (16 * 10^6) = 3,200,000 \text{ in } 200\text{ms}$
 $\text{ticks} = (400 * 10^{-3}) * (16 * 10^6) = 6,400,000 \text{ in } 400\text{ms}$

7. The prescaler is a circuit that converts a high frequency clock signal to be a lower one by dividing by powers of 2. This allows you to work with a larger range of frequencies without having to change the clock itself. Dividing the output increases the length of a tick.
8. link to demo: first 3 letters of my name is "KAT"
https://drive.google.com/file/d/1TSXM71-N7VH5_s_Q44Yq4-vViZSDJYxW/view?usp=sharing
 - a. video also submitted in the form in the lab writeup
9. Someday I will rule you all
10. N/A