Name: Tarunyaa Sivakumar
Pennkey: tarunyaa

Lab 2: Morse Code Detector

1.

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

int main(void)
{
    Initialize();

    PORTB |= (1<<PORTB1);
    PORTB |= (1<<PORTB2);
    PORTB |= (1<<PORTB3);
    PORTB |= (1<<PORTB4);

}
```

2.

```
void Initialize()
{
    DDRD &= ~(1<<DDD7);
    DDRB |= (1<<DDB1);
    PORTB &= ~(1<<PORTB1);
}

int main(void)
{
    Initialize();
    while(1) {
        if (PIND & (1<< PIND7)){
            PORTB |= (1<<PORTB1);
        }
        else {
            PORTB &= ~(1<<PORTB1);
        }
    }
}
```

Name: Tarunyaa Sivakumar
Pennkey: tarunyaa

3.

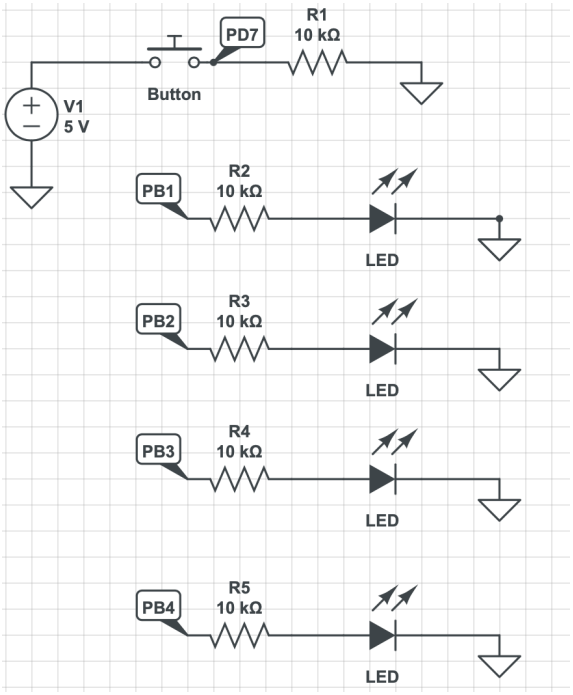
```
void Initialize()
{
    DDRD &= ~(1<<DDD7);
    DDRB |= (1<<DDB1);
}

int main(void)
{
    Initialize();
    int i = 0;
    while(1) {
        if (PIND & (1<< PIND7)) {
            _delay_ms( ms: 500);
            i += 1;
            if (i == 5){
                i = 1;
            }
        }
        if (i == 1) {
            PORTB |= (1<<PORTB1);
            PORTB &= ~(1<<PORTB2);
            PORTB &= ~(1<<PORTB3);
            PORTB &= ~(1<<PORTB4);
        }
        if (i == 2) {
            PORTB |= (1<<PORTB2);
            PORTB &= ~(1<<PORTB1);
            PORTB &= ~(1<<PORTB3);
            PORTB &= ~(1<<PORTB4);
        }
```

```
        if (i == 3) {
            PORTB |= (1<<PORTB3);
            PORTB &= ~(1<<PORTB1);
            PORTB &= ~(1<<PORTB2);
            PORTB &= ~(1<<PORTB4);
        }
        if (i == 4) {
            PORTB |= (1<<PORTB4);
            PORTB &= ~(1<<PORTB1);
            PORTB &= ~(1<<PORTB2);
            PORTB &= ~(1<<PORTB3);
        }
    }
}
```

Name: Tarunyaa Sivakumar
Pennkey: tarunyaa

4.



5.

An advantage of using interrupts over polling is that interrupts allow for highly predictable serving latencies. This is because they immediately interrupt the CPU when triggered by an event. This is unlike in polling where peripherals are forced to wait until a condition is met.

An advantage of using polling over interrupts is that interrupts can happen at any time and cause errors in other parts of the program, unlike in polling.

6.

Number of ticks in 30 ms = $\frac{16\times10^6}{1000}\times30 = 480{,}000$

Number of ticks in 200 ms = $\frac{16\times10^6}{1000}\times200 = 3{,}200{,}000$

Number of ticks in 400 ms = $\frac{16\times10^6}{1000}\times400 = 6{,}400{,}000$

7. Prescalers enable the timers or system clock on the Atmega328P to count at a different frequency to the 16 MHz that's set on the main system clock. More specifically, by using prescalers that are available from 1 to 1024, we can access timer frequencies from 16 MHz to 15.625 kHz. This is useful when we want to slow down the speed of a task or collect fewer data points to decrease memory usage.

Name: Tarunyaa Sivakumar
Pennkey: tarunyaa

8.
https://drive.google.com/file/d/1lUGBnvtrjx0jDPTFwhSmw8kX1HQdMvNT/view?usp=sharing

9. SOMEDAY I WILL RULE YOU ALL