BECS 31421 DATE 23/5/2025
EXPERIMENT 08

Pulse width modulation (PWM) with microcontroller

STUDENT NAME: G.P.D. THAMARA
STUDENT NUMBER: EC/2021/005

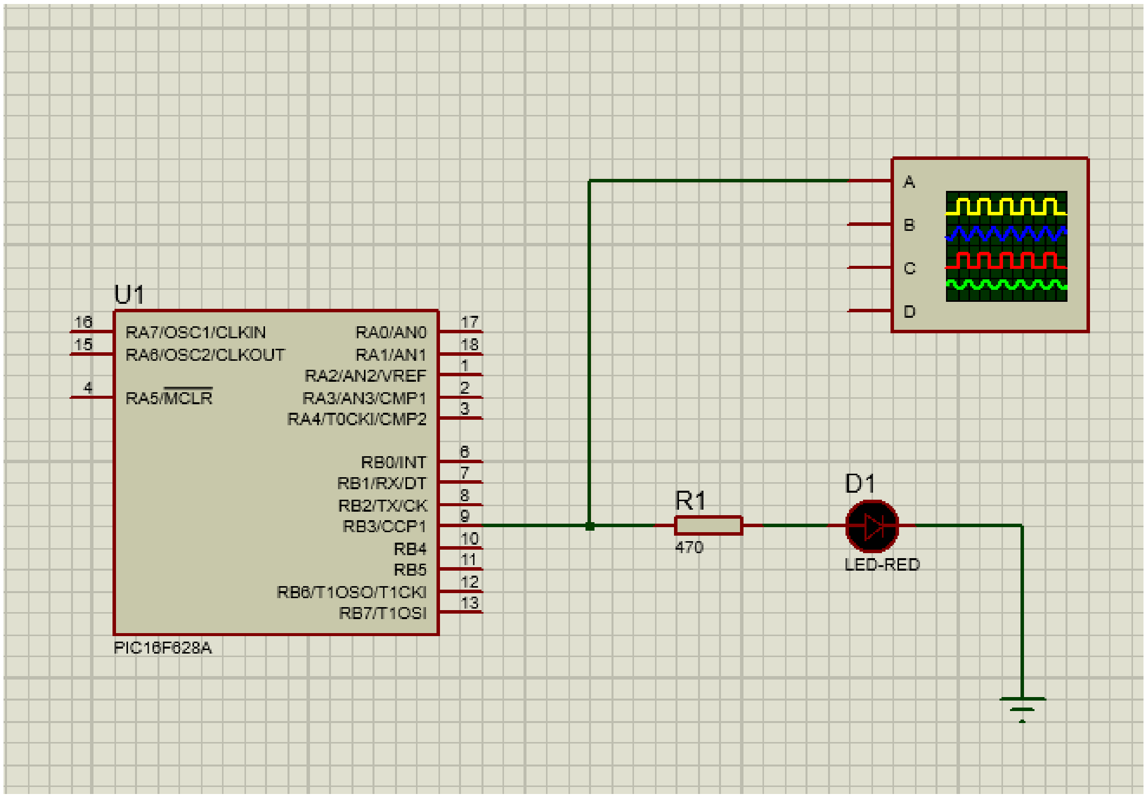
Source Code

```
// PWM LED Brightness Control using PIC16F628A

void main() {
    unsigned short duty_cycle; // Declare duty cycle variable
// Step 1: Configuration settings
    CMCON = 0x07;      // Disable comparators
    TRISB = 0x00;      // Set PORTB as output
    PORTB = 0x00;      // Initialize PORTB to 0
// Step 2: Initialize and start PWM
    PWM1_Init(5000);   // Initialize PWM1 at 5 kHz
    PWM1_Start();     // Start PWM1
// Step 3: Infinite loop to vary brightness
    while(1){
      for (duty_cycle=0 ; duty_cycle <=255; duty_cycle +=5){
           PWM1_Set_Duty(duty_cycle);
           Delay_ms(30);
      }
      for (duty_cycle=255;duty_cycle >=0;duty_cycle -=5){
           PWM1_Set_Duty(duty_cycle);
           Delay_ms(30);
      }
    }
}
```

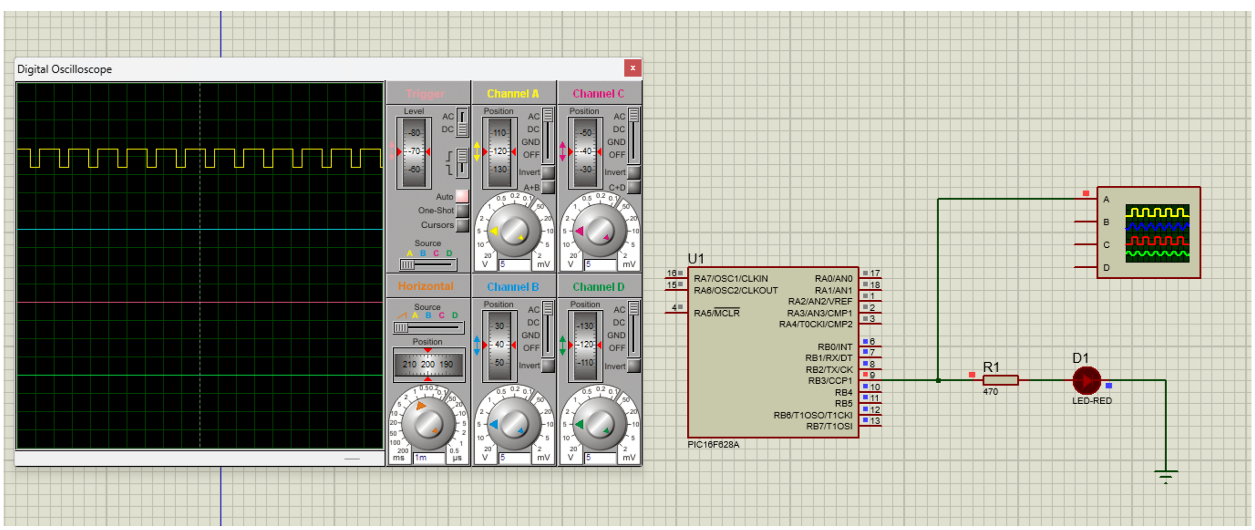

```
// PWM LED Brightness Control using PIC16F628A
void main() {
    unsigned short duty_cycle; // Declare duty cycle variable

    // Step 1: Configuration settings
    CMCON = 0x07;      // Disable comparators
    TRISB = 0x00;      // Set PORTB as output
    PORTB = 0x00;      // Initialize PORTB to 0

    // Step 2: Initialize and start PWM
    PWM1_Init(5000);   // Initialize PWM1 at 5 kHz
    PWM1_Start();     // Start PWM1

    // Step 3: Infinite loop to vary brightness
    while(1){
      for (duty_cycle=0 ; duty_cycle <=255; duty_cycle +=5){
           PWM1_Set_Duty(duty_cycle);
           Delay_ms(30);
      }
      for (duty_cycle=255;duty_cycle >=0;duty_cycle -=5){
           PWM1_Set_Duty(duty_cycle);
           Delay_ms(30);
      }
    }
}
```

BECS 31421 DATE 23/5/2025
EXPERIMENT 08

Circuit



Observations



Discussion

Pulse Width Modulation (PWM) is a widely used method for controlling analog devices with digital signals. Using the PIC16F628A microcontroller, PWM enables precise control of power delivery to components such as LEDs, allowing for adjustable brightness without needing a digital-to-analog converter. This is achieved by modifying the duty cycle—the percentage of time a digital signal remains high in each period.

The PIC16F628A supports PWM through its CCP (Capture/Compare/PWM) module, which can be configured to generate varying duty cycles. In a typical 8-bit resolution setup, values from 0 to 255 correspond to duty cycles from 0% to 100%. A lower duty cycle results in dimmer LED output due to shorter on-time, whereas a higher duty cycle increases brightness as the on-time lengthens.

PWM offers an efficient way to control power without dissipating excessive heat, as power devices switch fully on or off rather than operate in a linear mode. This technique is also power-efficient and scalable across applications such as motor control, audio signal generation, and lighting systems. Using MikroC Pro's PWM library simplifies the implementation, allowing rapid configuration and testing of PWM features in embedded systems. This experiment highlights the importance of PWM in real-world digital control applications.