**EXERCISE 08**

**Source Code**

sbit sw1 at RA1\_bit;

sbit sw2 at RA2\_bit;

void main() {

unsigned short duty\_cycle = 0;

unsigned short duty\_value = 0;

bit prev\_sw1;

bit prev\_sw2;

prev\_sw1 = 0;

prev\_sw2 = 0;

CMCON = 0x07; // Disable comparators

TRISB = 0x00; // Set PORTB as output

PORTB = 0x00;

TRISA = 0x06; // RA1 and RA2 as inputs, others as outputs

PORTA = 0x00;

PWM1\_Init(5000); // 5kHz PWM frequency

PWM1\_Start(); // Start PWM

while(1) {

// Edge detection for SW1 (Increase brightness)

if (sw1 == 1 && prev\_sw1 == 0) {

if (duty\_cycle < 100) {

duty\_cycle += 10; // Increase brightness by 10%

}

}

// Edge detection for SW2 (Decrease brightness)

if (sw2 == 1 && prev\_sw2 == 0) {

if (duty\_cycle >= 10) {

duty\_cycle -= 10; // Decrease brightness by 10%

}

}

// Convert 0–100% to 0–255 for PWM1\_Set\_Duty

duty\_value = (unsigned short)((duty\_cycle \* 255) / 100);

PWM1\_Set\_Duty(duty\_value);

// Save previous button states

prev\_sw1 = sw1;

prev\_sw2 = sw2;

// Simple software debounce delay

Delay\_ms(100);

}

}

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Circuit**

**A diagram of a circuit board

AI-generated content may be incorrect.**