PWM-BASED DC MOTOR SPEED CONTROL

TEAM MEMBERS

Gowri Vinod : AM.EN.U4CSE21226

Sabarinath U : AM.EN.U4CSE21249

Gaadha V S : AM.EN.U4CSE21222

Hemanth Lakshman : AM.EN.U4CSE21227

Keshagni Vikranth Goud : AM.EN.U4CSE21276

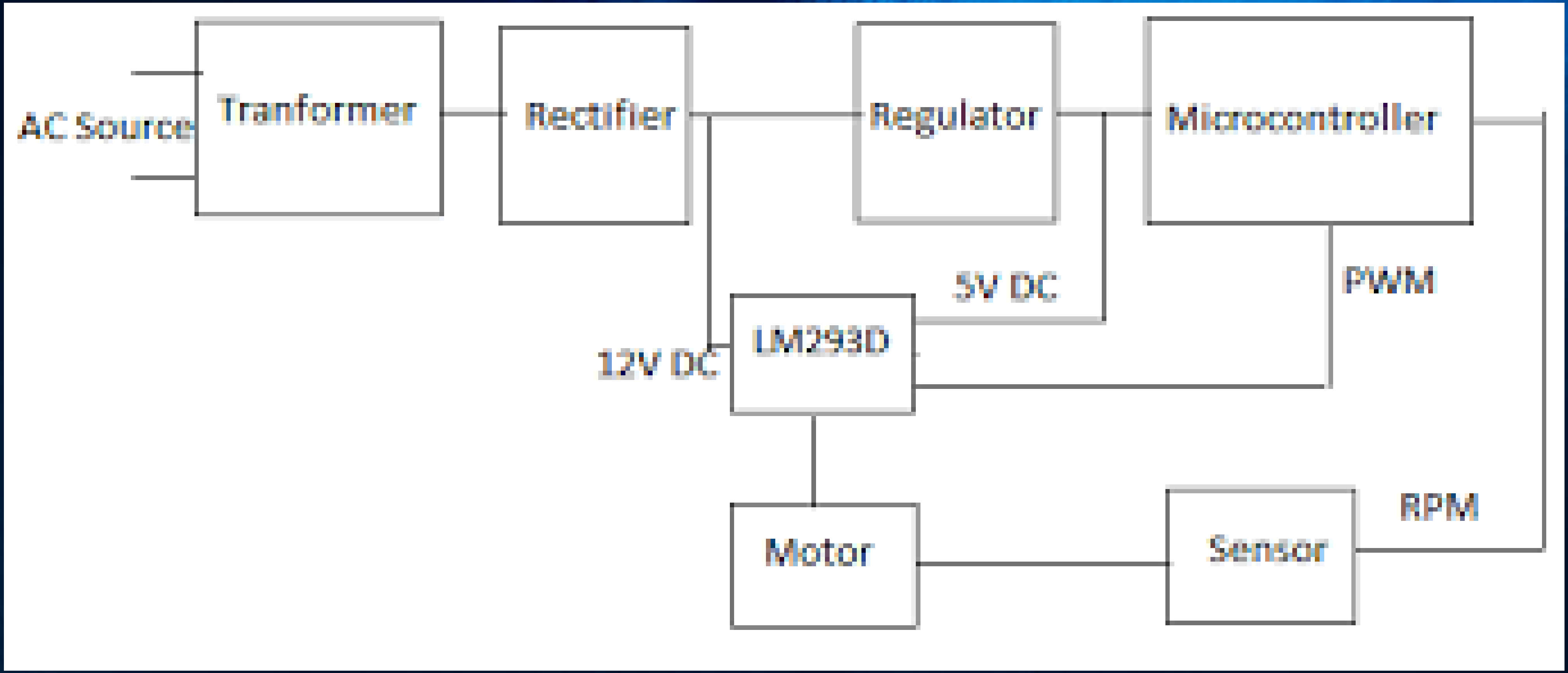
OBJECTIVES

- Develop a PWM-based system to regulate DC motor speed.
- Employ LPC2148 microcontroller, push buttons, optoisolator, Darlington transistor (TIP122), and passive components for accurate speed control.
- Study PWM and motor control principles for efficient manipulation of DC motor speed.
- Develop the project as an educational tool to improve understanding of microcontroller programming and hardware interfacing.
- Demonstrate diverse applications, from industrial automation to smart home integration, highlighting the adaptability of PWM-based motor control.

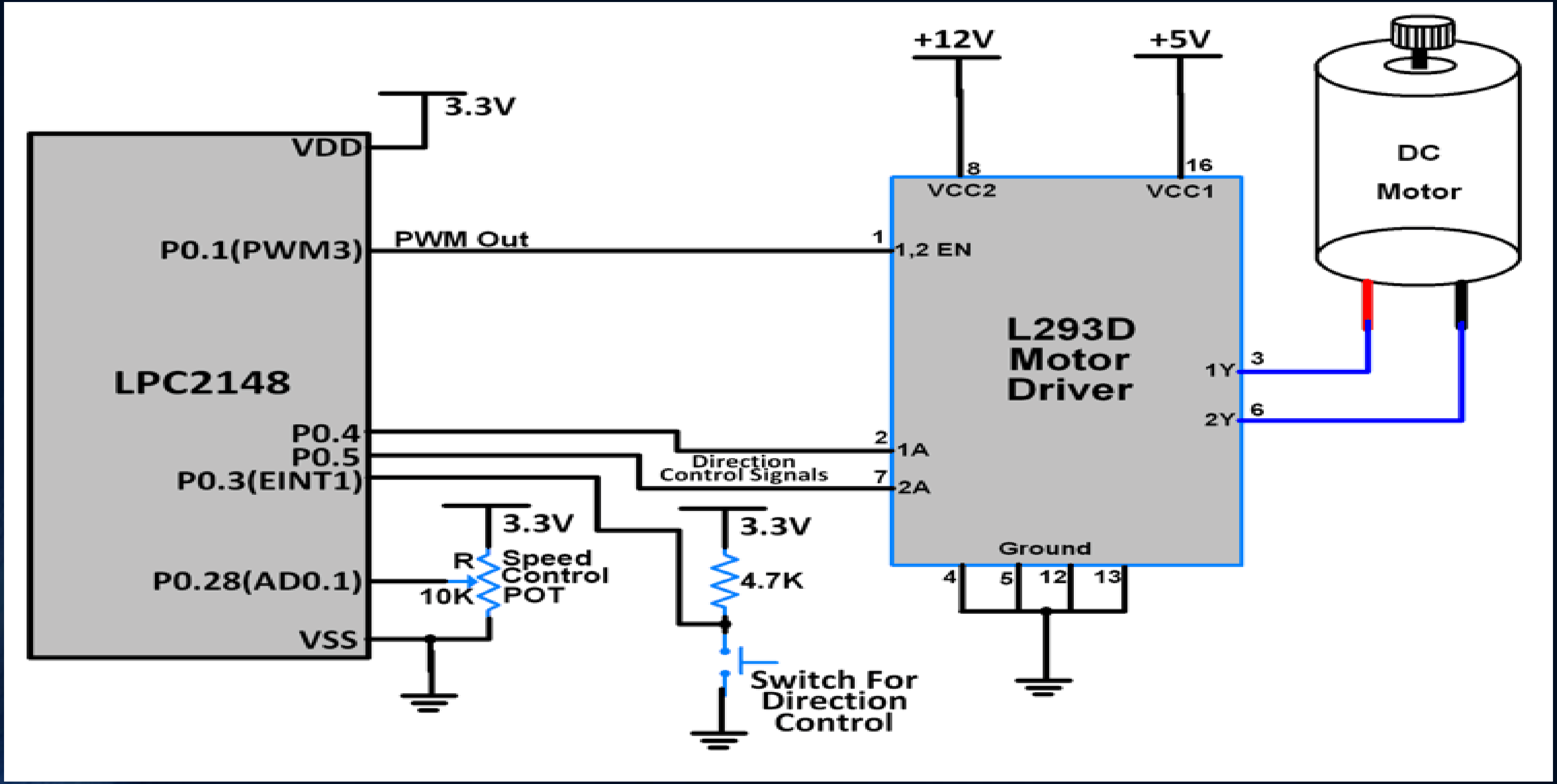
COMPONENTS

- LPC2148 (Micro-Controller)
- Push Buttons
- Optoisolator
- Darlington Transistir (TIP122)
- DC Motor (12V Rating)
- Other Passive Components
- Power Supply

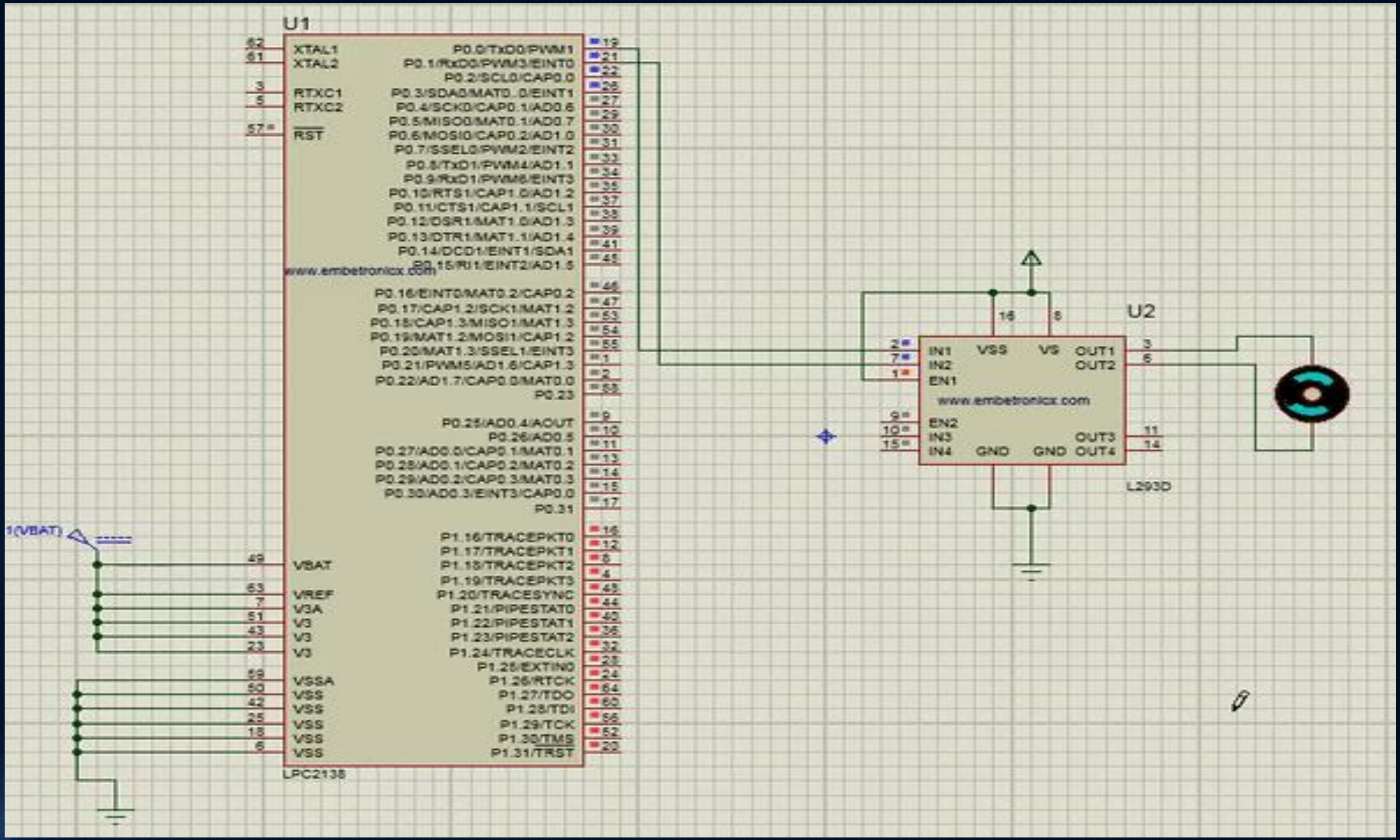
BLOCK DIAGRAMS



Circuit Diagram



Software


U1
XTAL1
XTAL2
RTXC1
RTXC2
RST
P0.0/TxD0/PWM1
P0.1/RxD0/PWM3/EINT0
P0.2/SCL0/CAP0.0
P0.3/SDA0/MAT0.0/EINT1
P0.4/SCK0/CAP0.1/AD0.6
P0.5/MISO0/MAT0.1/AD0.7
P0.6/MOSI0/CAP0.2/AD1.0
P0.7/SSEL0/PWM2/EINT2
P0.8/TxD1/PWM4/AD1.1
P0.9/RxD1/PWM6/EINT3
P0.10/RTS1/CAP1.0/AD1.2
P0.11/CTS1/CAP1.1/SCL1
P0.12/DSR1/MAT1.0/AD1.3
P0.13/DTR1/MAT1.1/AD1.4
P0.14/DCD1/EINT1/SDA1
P0.15/RI1/EINT2/AD1.5
www.embetronicx.com
P0.16/EINT0/MAT0.2/CAP0.2
P0.17/CAP1.2/SCK1/MAT1.2
P0.18/CAP1.3/MISO1/MAT1.3
P0.19/MAT1.2/MOSI1/CAP1.2
P0.20/MAT1.3/SSEL1/EINT3
P0.21/PWM5/AD1.6/CAP1.3
P0.22/AD1.7/CAP0.0/MAT0.0
P0.23
P0.25/AD0.4/AOUT
P0.26/AD0.5
P0.27/AD0.0/CAP0.1/MAT0.1
P0.28/AD0.1/CAP0.2/MAT0.2
P0.29/AD0.2/CAP0.3/MAT0.3
P0.30/AD0.3/EINT3/CAP0.0
P0.31
P1.16/TRACEPKT0
P1.17/TRACEPKT1
P1.18/TRACEPKT2
P1.19/TRACEPKT3
P1.20/TRACESYNC
P1.21/PIPESTAT0
P1.22/PIPESTAT1
P1.23/PIPESTAT2
P1.24/TRACECLK
P1.25/EXTIN0
P1.26/RTCK
P1.27/TDO
P1.28/TDI
P1.29/TCK
P1.30/TMS
P1.31/TRST
1(VBAT)
VBAT
VREF
V3A
V3
V3
V3
VSSA
VSS
VSS
VSS
VSS
VSS
LPC2138
U2
IN1
IN2
EN1
VSS
VS
OUT1
OUT2
www.embetronicx.com
EN2
IN3
IN4
GND
GND
OUT3
OUT4
L293D

WORKING

1. **Component Collection**: Gather all necessary components including the LPC2148 microcontroller, DC motor (12V), Darlington transistor (TIP122), optoisolator, push buttons, diode (1N4004), decoupling capacitor, resistors, and a power supply.
2. **Proteus Simulation**:

- Start Proteus and create a new project.
- Place the LPC2148 microcontroller on the workspace.
- Add the DC motor, Darlington transistor (TIP122), optoisolator, push buttons, diode, and other passive components to the schematic.
- Connect the components as per the project design. Ensure that the push buttons are connected to the appropriate pins on the LPC2148 for speed control. The optoisolator should interface the microcontroller and motor control circuitry.
- The Darlington transistor is used to amplify the PWM signal from the microcontroller to drive the motor.
- Add power sources (5V for the microcontroller and 12V for the motor).

1. **Code Development**:
- Use Keil IDE to write and compile the code for the LPC2148.
- The code should initialize the PWM functionality, read the push button states, and adjust the PWM duty cycle accordingly (100% for fast speed, 50% for slow speed).
- Transfer the compiled code to Proteus and associate it with the LPC2148.

2. **Simulation and Testing**:
- Run the simulation in Proteus.
- Test the operation by pressing the "Fast" and "Slow" buttons in the simulation to see the change in motor speed.
- Observe the PWM signal changes and motor response.

3. **Analysis and Debugging**:
- If the motor doesn't respond as expected, check the connections, component values, and the code for errors.
- Use the debugging tools in Proteus and Keil to troubleshoot issues.

4. **Finalization**:
- Once the system works as expected in the simulation, document the results.
- Prepare a report detailing the design, simulation process, and outcomes.

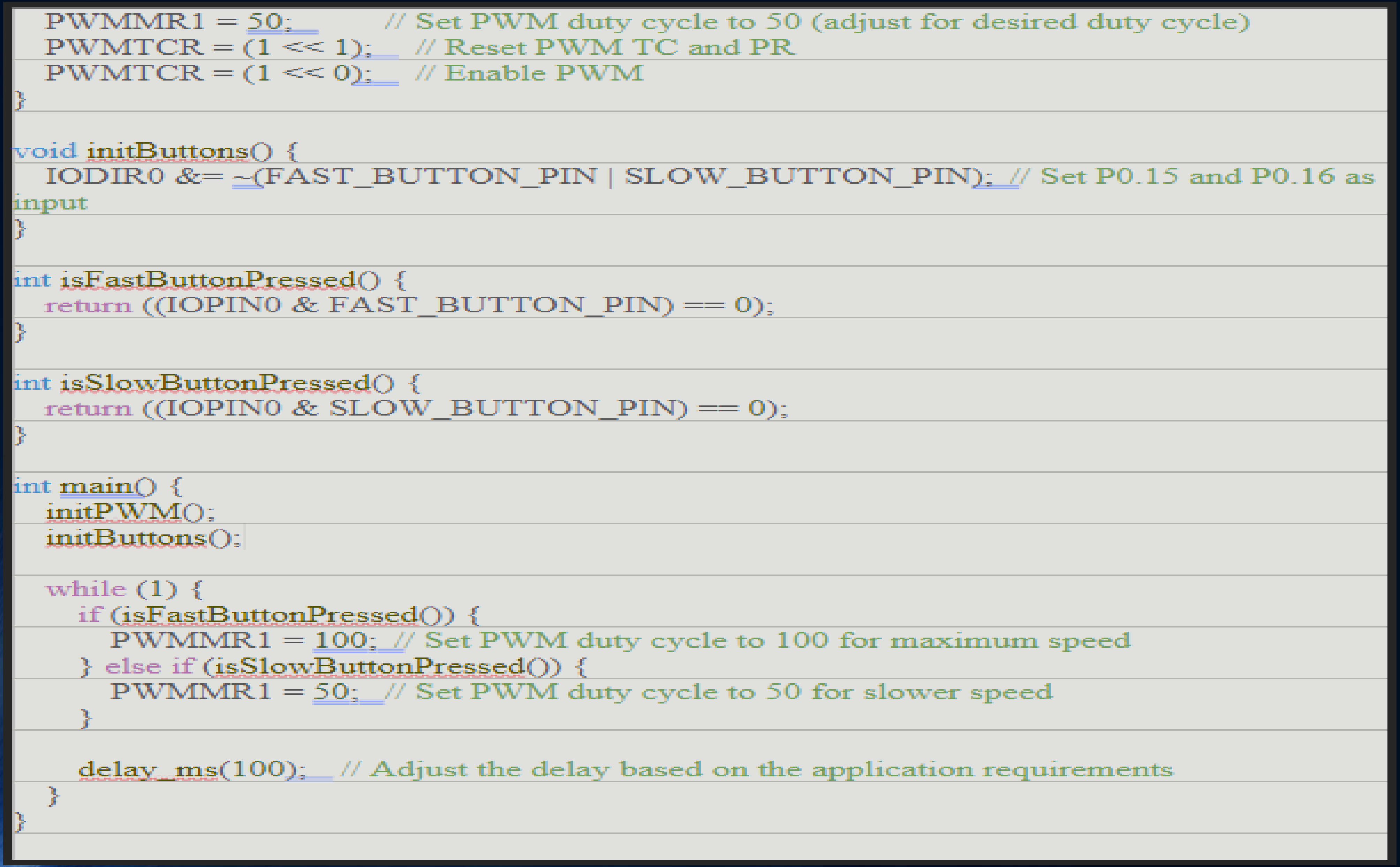
5. **Physical Implementation** (optional):
- If required, after successful simulation, build the actual circuit on a breadboard or PCB using the same components and connections tested in Proteus.
- Load the same code into a physical LPC2148 microcontroller and test the system with an actual motor.

```c
#include <lpc214x.h>


#define MOTOR_CONTROL_PIN (1 << 21) // Pin P0.21 for motor control
#define FAST_BUTTON_PIN  (1 << 15) // Pin P0.15 for the Fast button
#define SLOW_BUTTON_PIN  (1 << 16) // Pin P0.16 for the Slow button

void delay_ms(unsigned int count) {
  unsigned int i, j;
  for (i = 0; i < count; i++)
    for (j = 0; j < 10000; j++);
}

void initPWM() {
  PINSEL0 |= (1 << 10);  // Configure P0.21 as PWM output
  PWMPCR = (1 << 14);    // Enable PWM1
  PWMPR = 30;            // Set the PWM prescaler for a frequency of approximately 1 kHz
  PWMMR0 = 100;          // Set PWM period to 100 (adjust for desired frequency)
```



```
  PWMMR1 = 50;        // Set PWM duty cycle to 50 (adjust for desired duty cycle)
  PWMTCR = (1 << 1);   // Reset PWM TC and PR
  PWMTCR = (1 << 0);   // Enable PWM
}

void initButtons() {
  IODIR0 &= ~(FAST_BUTTON_PIN | SLOW_BUTTON_PIN); // Set P0.15 and P0.16 as
input
}

int isFastButtonPressed() {
  return ((IOPIN0 & FAST_BUTTON_PIN) == 0);
}

int isSlowButtonPressed() {
  return ((IOPIN0 & SLOW_BUTTON_PIN) == 0);
}

int main() {
  initPWM();
  initButtons();

  while (1) {
    if (isFastButtonPressed()) {
      PWMMR1 = 100; // Set PWM duty cycle to 100 for maximum speed
    } else if (isSlowButtonPressed()) {
      PWMMR1 = 50; // Set PWM duty cycle to 50 for slower speed
    }

    delay_ms(100);  // Adjust the delay based on the application requirements
  }
}
```

CONCLUSION

•Successful demonstration of PWM-based DC motor speed control project, showcasing the effective application of embedded systems and microcontroller programming.
•Developed a comprehensive understanding of PWM principles and their practical implementation in motor control.
•Key project components, including the LPC2148 microcontroller, push buttons, optoisolator, and Darlington transistor, enabled precise motor speed control.
•Project's significance lies in its educational value, offering hands-on experience in microcontroller programming and hardware interfacing, with versatile applications in various industries.
•Future directions include refining the system for specific industrial applications, incorporating advanced control algorithms, and exploring additional features, establishing a solid foundation for ongoing innovation in embedded systems and motor control.

THANK YOU