Wireless Brushless DC Motor Controller using STM32F401RE

RAYHAN HUSSAIN

EMBEDDED SYSTEM PROJECT

This project aims to create a wireless Brushless DC motor controller using the STM32F401RE microcontroller, capable of controlling the speed and direction of a BLDC motor using wireless communication.

Key Components

1. **STM32F401RE Microcontroller**: The main controller for motor control and wireless communication.

2. **BLDC Motor**: The motor to be controlled.

3. **Motor Driver**: A motor driver IC, such as L298N or similar, for controlling the BLDC motor.

4. **Wireless Communication Module**: For wireless control (e.g., Bluetooth, Wi-Fi, RF module, or Zigbee).

5. **Sensors**: Optional sensors for feedback control (e.g., Hall effect sensors for BLDC position feedback).

6. **Power Supply**: Appropriate power supply for the motor, motor driver, and microcontroller.

7. **User Interface**: Buttons, switches, or a smartphone app for user control.

Project Stages

Stage 1: **Hardware Setup**

1. Assemble the hardware components on a breadboard or custom PCB.
2. Connect the BLDC motor to the motor driver.
3. Connect the motor driver to the STM32F401RE microcontroller.
4. Connect the wireless communication module to the microcontroller.
5. If using sensors, connect the sensors for feedback control.

Stage 2: **STM32F401RE Programming**

1. Set up the STM32F401RE development environment (e.g., STM32CubeIDE).
2. Write code to configure and initialize GPIO pins for motor control.
3. Implement BLDC motor control algorithms (e.g., sensorless control or using Hall effect sensors).
4. Write code for wireless communication (e.g., Bluetooth or Wi-Fi) to receive motor control commands wirelessly.

Stage 3: **User Interface**

1. Create a user interface (e.g., smartphone app or physical controls) for motor speed and direction control.
2. Implement code for sending control commands to the STM32F401RE over the wireless communication module.

Stage 4: **Integration and Testing**

1. Assemble all components.
2. Test motor control and wireless communication.
3. Implement safety mechanisms and error handling in the code.
4. Optimize motor control algorithms for efficiency and performance.

Stage 5: **Documentation and Finalization**

1. Document the project, including schematics, code, and usage instructions.
2. Perform a final test and debug any issues.
3. Package the project for presentation or practical use.