**MOTOR FAILURE PREDICTION – STM32F401**

**INTRODUCTION:**

This project embodies a multi-sensor system integrating ultrasonic sensors for distance measurement, a temperature sensor, and status LEDs, along with Bluetooth-based communication.Based on the readings from the sensors, an industry grade motor’s failure prone status can be identified. The key components are:

1. **Ultrasonic Sensors**: Two sensors for distance measurements.
2. **Temperature Sensor**: Measures temperature to control a fan.
3. **Status LEDs**: Indicates distance thresholds for the ultrasonic sensors.
4. **Bluetooth Communication**: Sends temperature data over UART.
5. **Interrupts**: Configures and handles interrupts for external events.

**CODE EXPLANATION:**

**Global Definitions**

**1. Power Pins**

* **VDD (Power Supply)**: Connect to +3.3V.
* **VSS (Ground)**: Connect to GND.

**2. Clock Pins**

* **HSE (High-Speed External Clock)**:
  + **OSC\_IN (PA0)**: Connect to the external crystal oscillator (input).
  + **OSC\_OUT (PA1)**: Connect to the external crystal oscillator (output).
* **LSE (Low-Speed External Clock)**:
  + **LSE\_IN (PC14)**: Connect to a 32.768 kHz crystal oscillator (input).
  + **LSE\_OUT (PC15)**: Connect to the 32.768 kHz crystal oscillator (output).

**3. Debugging and Programming Pins**

* **SWD (Serial Wire Debug)**:
  + **SWDIO (PA13)**: Data Input/Output for debugging.
  + **SWCLK (PA14)**: Debug clock signal.
  + **NRST**: Reset pin (optional for debugging).

**4. GPIO Pins**

* Configurable as Input, Output, Alternate Function (AF), or Analog.
* Refer to the **STM32F401 Reference Manual** for AF mappings.

Example:

* **PA0**: Can be used as GPIO, ADC input, or External Interrupt.
* **PB6/PB7**: Often used for I2C1 (SCL and SDA).
* **PA9/PA10**: Commonly used for USART1 (TX and RX).

**5. Communication Interfaces**

**USART**

* **USART1**:
  + **PA9**: TX (Transmit).
  + **PA10**: RX (Receive).
* **USART2**:
  + **PA2**: TX.
  + **PA3**: RX.

**6. ADC Pins**

* Analog input pins can be connected to GPIO configured as ADC:
  + **PA0-PA5**: ADC1 channels 0-5.
  + **PB0/PB1**: ADC1 channels 8/9.

**1. Initialization**

**1.1 System Clock Configuration**

The Timer\_Config function sets the system clock to use the HSI oscillator (16 MHz) and configures Timer 2 for time-based tasks, such as delay and pulse width measurement.

**1.2 GPIO Configuration**

The GPIO\_Config function initializes GPIO pins for:

* **Output**: Sensor triggers, LEDs, and fan.
* **Input**: Sensor echoes and temperature sensor.

**1.3 ADC Configuration**

The ADC\_Config function configures the ADC for analog readings from the temperature sensor.

**1.4 UART Configuration**

The USART1\_Init function sets up UART1 at 9600 baud for Bluetooth communication.

**1.5 NVIC Configuration**

The NVIC\_Config function enables and configures an external interrupt (EXTI) on PA0. This handles external events such as ultrasonic sensor measurements.

**2. Main Function**

The main function continuously:

1. **Measures Distance**: Reads from both ultrasonic sensors using Measure\_Distance1 and Measure\_Distance2.
2. **Reads Temperature**: Captures temperature data using Read\_Temperature.
3. **Controls Fan**: Adjusts the fan based on the temperature threshold using Update\_Fan.
4. **Updates LEDs**: Reflects distance thresholds for both sensors using Update\_LED1 and Update\_LED2.
5. **Sends Data**: Sends temperature data over Bluetooth using printit.

**3. Sensor Measurement and Control**

**3.1 Distance Measurement**

Measure\_Distance1 and Measure\_Distance2 measure the distance using the ultrasonic sensors. The process:

1. Sends a 10 µs pulse via the trigger pin.
2. Captures the echo pulse width using Timer 2.
3. Converts pulse width to distance using the formula: Distance (cm)=Pulse Width×0.03432\text{Distance (cm)} = \frac{\text{Pulse Width} \times 0.0343}{2}Distance (cm)=2Pulse Width×0.0343​

**3.2 Temperature Measurement**

Read\_Temperature uses the ADC to read analog values from the temperature sensor and converts them to a usable format.

**3.3 Fan Control**

The Update\_Fan function turns the fan on/off based on the TEMP\_THRESHOLD.

**3.4 LED Updates**

Update\_LED1 and Update\_LED2 control the LEDs to indicate distance ranges:

* **Red LED**: Close distance.
* **Yellow LED**: Moderate distance.
* **Green LED**: Far distance.

**4. Communication**

**4.1 UART**

USART1\_Write and USART1\_WriteString handle character and string transmission over UART. printit formats temperature data and sends it via Bluetooth.

**4.2 External Interrupt**

The interrupt handler EXTI0\_IRQHandler clears the pending flag for PA0 and resets LEDs tied to Sensor 2.

**Configurations and Their Uses**

| **Configuration** | **Use** |
| --- | --- |
| **GPIO Configuration** | Initializes pins for sensors, LEDs, fan, and communication. |
| **Timer Configuration** | Provides accurate timing for ultrasonic sensor measurements. |
| **ADC Configuration** | Enables temperature sensing via analog-to-digital conversion. |
| **NVIC Configuration** | Enables external interrupts for sensor events. |
| **UART Configuration** | Facilitates Bluetooth communication for temperature monitoring. |
| **PIN DIAGRAM:** |  |

