**Project Documentation: Home Automation System MQTT based**

**Overview**

This project implements a home automation system using an Arduino Uno, Ethernet shield, and various sensors. It allows for remote control of lights and monitoring of room temperatures through MQTT (Message Queuing Telemetry Transport) protocol.

**Components Used**

* Arduino Uno
* Ethernet Shield
* DHT11 Temperature and Humidity Sensor (x2)
* Relay Modules for Lights (x2)

**Libraries Used**

* SPI.h
* Ethernet.h
* PubSubClient.h
* DHT.h

**Pin Configuration**

* ROOM1\_LIGHT1: Pin 7 (Relay control for Room 1 Light 1)
* ROOM2\_LIGHT1: Pin 8 (Relay control for Room 2 Light 1)
* ROOM1\_DHTPIN: Pin 2 (Data pin for DHT11 sensor in Room 1)
* ROOM2\_DHTPIN: Pin 3 (Data pin for DHT11 sensor in Room 2)

**MQTT Topics**

* Topic for controlling Room 1 Light 1: **home/room1/light1**
* Topic for controlling Room 2 Light 1: **home/room2/light1**
* Topic for feedback and status updates: **home/feedback**
* Topic for publishing Room 1 temperature: **home/room1/temperature**
* Topic for publishing Room 2 temperature: **home/room2/temperature**

**Network Configuration**

* MAC Address: **0xDE, 0xED, 0xBA, 0xFE, 0xFE, 0xED**
* IP Address: **192.168.1.100**
* MQTT Server IP: **192.168.229.60**
* MQTT Server Port: **1883**

**Functions**

**void callback (char\* topic, byte\* payload, unsigned int length)**

This function is the callback handler for MQTT messages. It processes incoming messages and controls the lights accordingly. It also publishes feedback messages and prints status updates to the Serial Monitor.

**void reconnect()**

This function attempts to reconnect to the MQTT server if the connection is lost. It generates a random client ID, attempts to connect, and subscribes to the relevant topics upon successful connection.

**void setup()**

* Initializes pin modes for lights and sets them to OFF.
* Initializes serial communication at 9600 baud.
* Initializes Ethernet connection.
* Initializes DHT sensors in Rooms 1 and 2.
* Checks for the presence of the Ethernet shield and the status of the network link.

**void loop()**

* Checks if the MQTT client is connected. If not, it attempts to reconnect.
* Handles incoming MQTT messages and publishes temperature readings.
* Reads temperature from DHT sensors, publishes the data to MQTT, and prints it to the Serial Monitor.

**Usage**

1. Upload this code to an Arduino Uno with the appropriate hardware connected.
2. Ensure the Ethernet shield is connected and the Ethernet cable is plugged in.
3. Open the Serial Monitor to monitor the system's status.
4. Use an MQTT client to send messages to control lights or receive temperature data.