ESP CODE LOGIC

Header Files :

The code includes various header files for libraries and modules that it uses, such as WiFi, EEPROM, HTTPClient, BluetoothSerial, etc.

WiFiMulti Setup :

A WiFiMulti object is created to manage multiple WiFi networks. This allows the ESP32 to try connecting to different networks if the first one fails.

BluetoothSerial Setup :

The code checks if Bluetooth is enabled and initializes a BluetoothSerial object, which can be used for Bluetooth communication.

Watchdog Timer Setup :

It sets up a Watchdog Timer (WDT) with a timeout of 8 minutes to prevent the ESP32 from getting stuck.

Global Variables :

Various global variables and flags are declared to control program flow and to keep track of OTA (Over-The-Air) update availability.

Setup Function :

- The setup function is called once during initialization.
- It configures the serial communication and sets the receive buffer size.
- Initializes the EEPROM, which is used to store Wi-Fi credentials.
- Configures various pins for LEDs and hardware switches.
- Retrieves saved Wi-Fi credentials and other parameters from the EEPROM.
- Create HTTP URL and Shared Access Signature (SAS) for connecting to Azure IoT Hub.

- Calls Connect_To_WiFi function to connect to a Wi-Fi network.
- Calls OTA_Param_Update function to check for OTA parameters.

Loop Function :

- The loop function is executed repeatedly.
- It checks if the ESP32 is connected to a Wi-Fi network and reconnects if not.
- Periodically sends an empty packet to Azure IoT Hub to stay synchronized.
- Checks for incoming data on Serial1 (Hardware Serial) and sends it to Azure.
- Resets the watchdog timer to avoid a system reset.

Send_To_Azure Function :

Sends data to Azure IoT Hub using an HTTP POST request.

Handles HTTP response codes and triggers certain actions based on the response.

Send_Json_Packet Function :

Creates and sends a JSON packet to the PLC (Programmable Logic Controller).

Connect_To_WiFi Function :

Attempts to connect to a WiFi network using stored credentials.

Checks the connection status, retrieves network information, and pings Google to verify internet connectivity.

OTA_Param_Update Function :

Sends an HTTP GET request to retrieve OTA update parameters from a specific URL.

Serial_Input Function :

Activates a channel for input from both hardware and Bluetooth serial interfaces.

Reads and processes incoming data, typically used for setting or updating credentials.

Json_Parser Function :

Parses JSON data received from the input channels.

Handles firmware and parameter updates, and updates the EEPROM if necessary.

Initiates firmware updates via HTTP if an update is available.

OTA_Firmware_Update Function :

Wait till 3 minutes and blink led continuously.

Connect to WiFi and Make Http get request to API.

Update the Firmware and keep led off.

After update, it will restart the system.

This code mainly functions as a gateway between an Arduino-based PLC and Azure IoT Hub. It connects to Wi-Fi, retrieves and processes data, and can perform OTA updates for firmware and parameters when necessary. The various LEDs indicate the status of different operations, and the code is designed to be robust and recover from potential errors.