

Device Wi-Fi Commissioning for ESP32

Description

- The **Internet of things (IoT)** is a system of interrelated computing devices, mechanical and digital machines, and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.
- So, in this project, we handle the output pin of ESP32 for switching and dimming a LED bulb using a local Web server. For a local Web server, we do not need an Internet connection. We can handle everything over Wi-Fi.
- To make this work, we use an Android Application to connect the ESP32 to desired Wi-Fi network and control the LED bulb.

Protocols/Functions Implemented

- **Socket Programming-** It is a way of connecting two nodes on a network to communicate with each other. One socket(node) listens on a port at an IP, while other socket reaches out to the other to form a connection. Server forms the listener socket while client reaches out to the server.
- **UDP Broadcast (User Datagram Protocol)-** It is an automatic method that can be used without manually entering the IP address of all devices.

Requirements

- ESP32 chip Microcontroller
- LED with Resistor
- Android Device
- Wi-Fi Network
- Android Application

Procedure

- There are two sides to this project.
 - On ESP32:
 - At first the ESP chip micro-controller is set in AP mode.
 - The SSID and Password of Wi-Fi Router is fed to the chip, which then changes to Station mode and connects to the Wi-Fi.
 - After connecting with Wi-Fi, it Broadcasts it's IP Address in a particular Network ID using UDP.

- On Android APP side:
 - First, we connect to ESP32 which is in AP mode, and feed the chip with Wi-Fi credentials.
 - After sending the credentials, APP connects to the same network and waits for the ESP chip to do the same.
 - Since the ESP chip is broadcasting it's IP Address, we listen to the broadcasted message and save it.
 - From the saved ESP chip's IP Address, we open a socket and perform socket communication and turn the LED ON or OFF.

Testing

- Switched on ESP32 chip.
 - Observed that the server of ESP32 chip has started.
 - Opened Android application.
 - Discover nearby Wi-Fi networks.
 - Onboard the ESP32 chip with Wi-Fi credentials.
 - Observed that ESP chip is connected to Wi-Fi.
 - Observed that ESP chip has started broadcasting it's new assigned IP Address using UDP.
 - In the Android application using NSD, discover the ESP IP Address.
 - IP Address is displayed on the screen.
 - Click on the IP Address to connect to ESP32 chip.
 - Observe that successfully connected.
- Click on LED ON/OFF button on application and observe that led switches ON and OFF.

Output Samples:



