
Description of Device to switch on PC remotely

Xiaoliang Wang, Chengze Ye

xiaoliang.wang@fau.de chengze.ye@fau.de

Friedrich Alexander University Erlangen-Nürnberg, Germany

Understanding the Requirements: The objective is to develop a remotely-controlled arguing based on-off switch device that connects to the school's eduroam WiFi network and retrieves information from an Open Google Doc to determine the device's switching state.

1. **Hardware Selection:** The ESP32 microcontroller, equipped with an integrated WiFi module, is chosen for its suitability to the project requirements.
2. **Development Environment Setup:** Install the Arduino IDE or a compatible development environment and configure it to support the ESP32 microcontroller.
3. **ESP32 Programming:** Develop the firmware or program for the ESP32 microcontroller using the Arduino programming language.
 - a. Encompass WiFi connectivity: Set up the WiFi credentials on the microcontroller so that it can connect to the internet, configure the ESP32 to establish a connection with the school's eduroam WiFi network. Retrieve the required WiFi network credentials, including the SSID (network name) and security parameters, from the school's IT department.
 - b. Communication with the Google Doc: Employ the Google Docs API to access and retrieve data from the designated Google Doc. Obtain the requisite API credentials and authentication tokens for accessing the document.
 - c. The logic for switching the device based on retrieved information.
4. **Testing and Debugging:** Thoroughly test the device to ensure proper functionality. Identify and rectify any bugs, errors, or unexpected behavior. Make iterative adjustments to the program as necessary.

5. **for the further optimization and Function Expansion:** Once the device is functioning as intended, consider the following areas for optimization and function expansion:

a. **Power Efficiency:** Optimize power consumption by implementing sleep modes or power management techniques to prolong battery life or reduce energy usage, if applicable.

b. **User Interface:** Enhance the device's usability by incorporating a user interface, such as an LCD display or LEDs, to provide real-time information or status updates.

Pseudo code in the C programming language:

```
// WiFi Connectivity
void configure_wifi() {
    // Configure ESP32 WiFi module with eduroam
    credentials and settings
}

void connect_wifi() {
    // Connect ESP32 to eduroam WiFi network
}

// Google Doc Integration
void retrieve_data_from_google_doc() {
    // Retrieve data from the specified Google Doc using the
    Google Docs API
}

// Decision Making Logic
void switch_on() {
    // Function to switch on the device
}

void switch_off() {
    // Function to switch off the device
}

void update_device_state(char* data) {
    // Update device state based on the retrieved data from
    Google Doc
    if (strcmp(data, "on") == 0) {
```

```
        switch_on();
    } else if (strcmp(data, "off") == 0) {
        switch_off();
    }
}
```

// Main Program

```
int main() {
    configure_wifi();
    connect_wifi();

    while (1) {
        char* data = retrieve_data_from_google_doc();
        update_device_state(data);
    }

    return 0;
}
```

- [1] Schilling T, Bauer M, Biskup C, Haverich A, Hassel T. Engineering of biodegradable magnesium alloy scaffolds to stabilize biological myocardial grafts. *Biomed Eng-Biomed Tech* 2017;62:493–504.
- [2] Birkholz MN, Agrawal G, Bergmann C, Schröder R, Lechner SJ, Pich A, et al. Calcium phosphate/microgel composites for 3D powderbed printing of ceramic materials. *Biomed Eng-Biomed Tech* 2016;61:267–279.