

BAB 2

DASAR TEORI DAN TINJAUAN PUSTAKA

2.1. Metodologi Penelitian

Metode penelitian yang digunakan dalam penulisan tugas akhir ini adalah metode studi literatur dan observasi. Studi literatur dilakukan dengan cara mengumpulkan dan mempelajari penelitian sebelumnya ataupun penelitian yang berhubungan dengan perancangan sistem pengontrolan peralatan listrik dari jarak jauh berbasis suara menggunakan kecerdasan buatan *Natural Language Processing* (NLP). Observasi yaitu melakukan pengamatan pada objek terkait guna untuk mengumpulkan data yang diperlukan untuk pembuatan alat. Penelitian yang dilakukan berupa perancangan perangkat keras dan perancangan perangkat lunak. Kemudian dilakukan perakitan perangkat keras, selanjutnya dilakukan pengujian keseluruhan sistem. Jika alat yang dirancang tidak berfungsi dengan baik maka dilakukan diagnosa error kesalahan. Setelah dilakukan diagnosa dilanjutkan dengan *troubleshooting*. Apabila alat yang dirancang sesuai dengan tujuan penelitian maka dilanjutkan dengan pengambilan data dan pengukuran. Data tersebut ditampilkan ke dalam bentuk tabel atau grafik sehingga memudahkan proses analisis. Selanjutnya dilakukan analisis kinerja terhadap alat yang dirancang bangun. Selanjutnya dilakukan evaluasi untuk menarik kesimpulan dari hasil penelitian yang telah dilakukan disertai menilai kinerja alat.

2.2. ESP32 DEVKITC V4



Gambar 2. 1 Modul ESP32 DEVKITC V4

ESP32 adalah sebuah mikrokontroler yang diluncurkan oleh Espressif Systems pada tahun 2016. Mikrokontroler tersebut sangat populer di kalangan pengembang IoT karena memiliki kemampuan koneksi WiFi dan Bluetooth serta dapat diatur melalui program Arduino IDE.

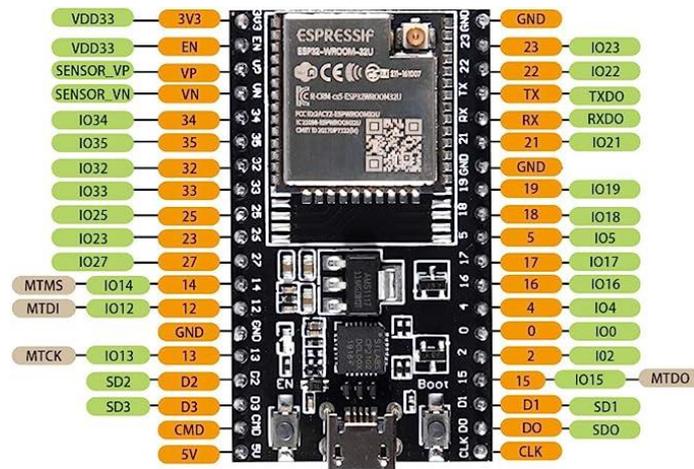
ESP32 memiliki arsitektur dual-core Xtensa LX6 dengan kecepatan hingga 240 MHz, RAM sebesar 520 KB, dan Flash Memory sebesar 4 MB. Selain itu, ESP32 juga dilengkapi dengan berbagai macam fitur lain seperti ADC (Analog to Digital Converter), DAC (Digital to Analog Converter), PWM (Pulse Width Modulation), SPI (Serial Peripheral Interface), I2C (Inter-Integrated Circuit) dan masih banyak lagi.

Dalam proyek-proyek IoT, ESP32 digunakan sebagai otak atau controller untuk menghubungkan perangkat-perangkat elektronik ke internet sehingga dapat dikontrol dari jarak jauh melalui aplikasi web atau mobile app. Dengan harga yang terjangkau serta dukungan komunitas yang besar, membuat ESP32 menjadi pilihan yang tepat bagi para pengembang dalam membangun sistem IoT.

ESP32 memiliki berbagai macam fitur dan spesifikasi yang membuatnya menjadi pilihan populer di kalangan pengembang IoT. Berikut ini adalah beberapa fitur dan spesifikasinya:

1. Arsitektur Dual-core Xtensa LX6 dengan kecepatan hingga 240 MHz.
2. Memiliki RAM sebesar 520 KB dan Flash Memory sebesar 4 MB.
3. Mendukung koneksi WiFi (802.11 b/g/n) dan Bluetooth Low Energy (BLE).
4. Dilengkapi dengan antena onboard untuk koneksi WiFi serta dukungan eksternal untuk antena Bluetooth.
5. Dapat diatur menggunakan program Arduino IDE maupun ESP-IDF (Espressif IoT Development Framework).
6. Memiliki banyak pin GPIO, ADC, DAC, PWM, SPI, I2C, dan UART yang dapat digunakan dalam proyek-proyek elektronik.
7. Mendukung mode deep sleep sehingga konsumsi daya dapat ditekan saat tidak digunakan.

Fitur-fitur tersebut membuat ESP32 sangat cocok digunakan sebagai otak atau controller dalam sistem IoT seperti smart home automation, monitoring lingkungan melalui sensor-sensor elektronik atau bahkan robotika.



Gambar 2. 2 Pin Out Modul ESP32 DEVKITC V4

Berikut adalah susunan pin pada ESP32 DevKitC V4:

1. Pin 3V3: Sumber tegangan 3.3V yang dapat digunakan untuk menyediakan daya pada sensor atau perangkat eksternal yang membutuhkan tegangan 3.3V.
2. Pin GND: Ground atau koneksi tanah.
3. Pin EN: Pin enable, digunakan untuk mengaktifkan atau menonaktifkan modul ESP32.
4. Pin BOOT: Pin boot digunakan untuk memilih mode boot. Dalam kebanyakan kasus, pin ini dihubungkan ke ground (GND) untuk memilih mode boot normal.
5. Pin GPIO0: Pin input/output digital yang dapat digunakan untuk mengatur status boot dan memilih mode boot alternatif.
6. Pin GPIO1 hingga GPIO39: Pin input/output digital yang dapat digunakan untuk berbagai keperluan, seperti pengendalian perangkat eksternal, pembacaan sensor, atau komunikasi dengan perangkat lain.
7. Pin ADC: Pin input analog untuk membaca sinyal analog dari sensor atau perangkat eksternal.
8. Pin AREF: Pin referensi tegangan analog eksternal.
9. Pin TX0 dan RX0: Pin komunikasi serial UART untuk pengiriman (TX) dan penerimaan (RX) data.
10. Pin TX2 dan RX2: Pin komunikasi serial UART tambahan.
11. Pin SDA dan SCL: Pin komunikasi I2C untuk komunikasi dengan perangkat I2C eksternal.
12. Pin SCK, MISO, dan MOSI: Pin komunikasi SPI untuk komunikasi dengan perangkat SPI eksternal.

13. Pin SCK, SD0, CMD, D0, D1, D2, D3, dan D4: Pin komunikasi SDIO untuk menghubungkan kartu SD atau perangkat eksternal lain yang kompatibel dengan protokol SDIO.
14. Pin DAC1 dan DAC2: Pin output DAC (Digital-to-Analog Converter) untuk menghasilkan sinyal analog.
15. Pin VP dan VN: Pin input analog untuk pengukuran tegangan pada saluran yang berbeda.
16. Pin GPIO34 hingga GPIO39: Pin input/output digital yang mendukung fitur pulldown dan pullup internal.

2.3. Modul Relay

Relay merupakan sebuah komponen elektronik yang memiliki fungsi sebagai saklar. Cara kerja relay dengan menggunakan induksi elektromagnetik yang telah dialiri listrik akan membuat saklar terhubung ataupun terputus. Didalam relay terdapat sebuah *coil* dan *contact*, *coil* merupakan gulungan kawat yang akan menerima arus listrik, sedangkan *contact* merupakan saklar yang akan tersambung dan tertutup dengan ada atau tidaknya arus listrik.



Gambar 2. 3 Pin Relay 2 channel

Relay 2 channel adalah sebuah modul elektronik yang berfungsi sebagai saklar atau switch untuk mengontrol arus listrik pada dua jalur sekaligus. Modul ini terdiri dari dua buah relay yang masing-masing dapat diatur dengan sinyal kontrol digital.

Setiap relay pada modul ini memiliki tiga pin utama seperti Gambar 2.4, yaitu :

1. Pin VCC dan GND digunakan untuk memberikan catu daya ke modul.
Pin IN digunakan sebagai input sinyal kontrol digital yang berasal dari mikrokontroler atau board Arduino

Dalam penggunaannya, Relay 2 channel seringkali dipakai dalam proyek-proyek elektronik seperti kendali lampu, motor DC, pompa air dan lain-lain. Dengan menggunakan

Relay 2 channel, kita dapat dengan mudah mengendalikan perangkat-perangkat tersebut melalui program komputer maupun aplikasi mobile.

Dalam sistem kendali lampu berbasis IoT menggunakan framework ESP Rainmaker, relay digunakan sebagai saklar untuk mengontrol lampu. Relay terdiri dari empat komponen dasar, yaitu elektromagnet (coil), armature, switch contact point (saklar), dan spring. Fungsi dari coil adalah sebagai alat penarik kontak point agar dapat terhubung atau terputus disesuaikan dengan relay yang digunakan. Relay digunakan ketika diperlukan untuk mengontrol suatu sirkuit dengan sinyal daya rendah yang independen, atau ketika beberapa sirkuit harus dikontrol oleh satu sinyal. Relay dapat digunakan untuk mengendalikan sirkuit tegangan tinggi dengan bantuan dari sinyal tegangan rendah. Dalam sistem kendali lampu berbasis IoT menggunakan framework ESP Rainmaker, relay digunakan untuk mengontrol lampu dengan bantuan dari sinyal daya rendah yang dikirimkan oleh ESP Rainmaker.

2.4. Library Framework ESP Rainmaker

Library yang di gunakan oleh framework esp rainmaker adalah sebagai berikut :

Library WiFiProv.h adalah salah satu library yang digunakan dalam ESP RainMaker untuk melakukan provisioning Wi-Fi pada perangkat ESP32. Berikut adalah kegunaan library WiFiProv.h:

1. Library ini digunakan untuk melakukan provisioning Wi-Fi pada perangkat ESP32.
2. Library ini memungkinkan pengguna untuk mengatur koneksi Wi-Fi pada perangkat ESP32 dengan mudah dan cepat.
3. Library ini dapat digunakan bersama dengan library RMaker.h dan library Wifi.h untuk membangun solusi AIoT menggunakan ESP32 dan ESP RainMaker.

Dalam penggunaannya, library WiFiProv.h dapat digunakan untuk melakukan provisioning Wi-Fi pada perangkat ESP32 dengan mudah dan cepat. Library ini merupakan salah satu dari beberapa library yang tersedia dalam ESP RainMaker dan dapat digunakan bersama dengan library lainnya untuk membangun solusi AIoT menggunakan ESP32 dan ESP RainMaker.

2.4.1 Framework ESP Rainmaker



Gambar 2. 4 ESP Rainmaker

ESP Rainmaker adalah kerangka kerja yang dikembangkan oleh Espressif Systems yang memungkinkan pengembang untuk dengan mudah menghubungkan perangkat Internet of Things (IoT) mereka ke cloud dan mengontrolnya melalui aplikasi seluler atau web. ESP Rainmaker menyediakan alat dan API yang memudahkan konfigurasi dan pengaturan perangkat IoT, serta integrasi dengan platform cloud seperti Amazon Web Services (AWS) atau Google Cloud Platform (GCP). Dengan ESP Rainmaker, pengguna dapat dengan cepat mengembangkan aplikasi IoT yang terhubung dengan mudah dan aman.

ESP Rainmaker bekerja dengan menghubungkan perangkat IoT ke cloud melalui protokol Wi-Fi. Perangkat IoT yang kompatibel dengan ESP Rainmaker akan memiliki modul Wi-Fi yang terintegrasi, seperti modul ESP32 dari Espressif Systems.

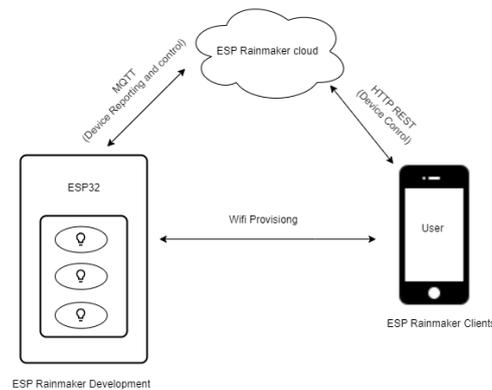
Proses kerja ESP Rainmaker dimulai dengan konfigurasi perangkat IoT menggunakan aplikasi seluler atau web. Pengguna dapat mengatur parameter seperti nama perangkat, jaringan Wi-Fi yang akan digunakan, dan kredensial cloud yang diperlukan untuk menghubungkan perangkat ke platform cloud.

Setelah konfigurasi selesai, perangkat IoT akan terhubung ke jaringan Wi-Fi dan mencoba menghubungkan ke server cloud menggunakan kredensial yang telah dikonfigurasi. ESP Rainmaker menyediakan API yang memudahkan pengembang untuk mengatur komunikasi antara perangkat IoT dan server cloud.

Setelah terhubung ke server cloud, perangkat IoT dapat mengirim dan menerima data dari aplikasi seluler atau web. Pengguna dapat mengontrol perangkat IoT, mengatur parameter, atau menerima data sensor melalui aplikasi seluler atau web yang terhubung dengan server cloud.

ESP Rainmaker juga menyediakan fitur keamanan seperti enkripsi data dan otentikasi pengguna untuk memastikan bahwa komunikasi antara perangkat IoT dan server cloud aman.

Dengan cara kerja yang sederhana dan alat yang mudah digunakan, ESP Rainmaker memungkinkan pengembang untuk dengan cepat menghubungkan perangkat IoT mereka ke cloud dan mengembangkan aplikasi IoT yang terhubung dengan mudah.



Gambar 2. 5 Arsitektur Framework ESP Rainmaker

Untuk membuat tombol atau relay di ESP Rainmaker, Anda dapat mengikuti langkah-langkah berikut:

1. Siapkan perangkat relay dan tombol yang akan digunakan.
2. Buat node baru di ESP Rainmaker dengan menggunakan fungsi `initNode()`.
3. Inisialisasi perangkat relay dan tombol dengan menggunakan fungsi yang disediakan oleh ESP Rainmaker, seperti `Switch` atau `Button`.
4. Tambahkan perangkat relay dan tombol ke dalam node dengan menggunakan fungsi `addDevice()`.
5. Implementasikan kode untuk menangani penerimaan data dari tombol atau perangkat lainnya. Anda dapat menggunakan fungsi yang disediakan oleh ESP Rainmaker, seperti `addCb()` untuk menambahkan callback function.
6. Atur pin GPIO sebagai output untuk mengontrol relay dan pin GPIO sebagai input untuk membaca status tombol dengan menggunakan fungsi `pinMode()`.
7. Implementasikan kode untuk membaca status tombol dan mengontrol relay sesuai dengan status tombol tersebut dengan menggunakan fungsi `digitalRead()` dan `digitalWrite()`.
8. Uji coba program dan pastikan tombol dan relay berfungsi dengan baik.