

## DESAIN DAN REALISASI FIRMWARE UNTUK PERANGKAT UTAMA HOME CONTROL UNIT

### DESIGN AND REALIZATION OF FIRMWARE FOR HOME CONTROL UNIT MAIN DEVICES

Megatari Wahyuningrum Handani<sup>1</sup>, Agung Nungroho Jati, S.T.,M.T<sup>2</sup>, Casi Setianingsih, S.T.,M.T<sup>3</sup>

<sup>1,2,3</sup> Prodi S1 Teknik Komputer, Fakultas Teknik Elektro, Universitas Telkom

<sup>1</sup>megatariwnh@student.telkomuniversity.ac.id, <sup>2</sup>agungnj @telkomuniversity.co.id,

<sup>3</sup>setiacasie@telkomuniversity.ac.id

---

#### Abstrak

Setiap orang pasti banyak yang menginginkan rumah nyaman serta aman ketika rumah dalam keadaan kosong dengan waktu lama. Namun masih terdapat beberapa orang yang belum dapat mengantisipasi kondisi tersebut, misalnya seperti kondisi keamanan rumah jika terdapat pencuri yang masuk atau kondisi lampu rumah yang lupa dimatikan saat bepergian, dengan demikian akan menimbulkan kerugian seperti kehilangan barang dan penggunaan listrik yang meningkat. Dari permasalahan yang terjadi, maka dirancanglah sebuah sistem rumah pintar yang dapat mengontrol dan memonitoring rumah pada jarak jauh. Sistem rumah pintar ini dirancang menggunakan beberapa sensor dan papan pengontrol yang terdiri dari sensor DHT22 sebagai pengukur suhu, sensor gas MQ2 untuk mendeteksi gas, sensor PIR untuk mendeteksi pergerakan, RFID untuk mengenali identitas, dan STM32F407 sebagai program controller. Papan pengontrol akan dibuat berbasis mikrokontroler yang ditunjang dengan Real-Time Clock, modul Wi-Fi, 8 port masukan, 8 port keluaran, dan 4 port modul Extended. Papan pengontrol dapat dikonfigurasi menggunakan software setup melalui komunikasi I2C dan SPI, sedangkan aplikasi android menggunakan koneksi Wi-Fi. Penggunaan sistem ini memiliki proses pengimplementasian yang mudah bagi pengguna tanpa membutuhkan instalasi yang rumit, sehingga dapat langsung digunakan. Hasil yang didapatkan dari penelitian ini adalah microcontroller dapat merespon permintaan data dan dapat mengeluarkan data output yang sesuai melalui koneksi I2C, Wi-Fi, dan modul Extended dengan kehandalan yang optimal, sehingga sistem perancangan rumah pintar dapat memenuhi kebutuhan bagi pengguna.

**Kata kunci :** Modul Wi-Fi, Aplikasi Android, Software Setup, Komunikasi I2C, SPI

---

#### Abstract

Everyone must have many who want a comfortable and safe home when the house is empty for a long time. However, there are still some people who have not anticipated this condition, for example, such as a home security condition if there is a thief entering or the condition of the house lights that are forgotten is switched off while traveling, thus causing losses such as loss of goods and increased electricity use. From the problems that occur, then a smart home system is designed that can control and monitor homes at a distance. This smart home system is designed using several sensors and a control board consisting of a DHT22 sensor as a temperature gauge, MQ-2 gas sensor for detecting gas, PIR sensor for detecting movement, LDR sensor for detecting light, RFID for identifying identity, and STM32F407 as a program controller. The controller board will be made based on the microcontroller which is supported by Real-Time Clock, Wi-Fi module, 8 digital input ports, 8 output ports, and 4 extended module ports. The controller board can be configured using setup software via I2C and SPI communication, while the android application uses a WiFi connection. The use of this system has an easy implementation process for users without requiring complicated installation, so it can be directly used. The results obtained from this study is that the microcontroller can respond to data requests and can output the appropriate output data through a I2C connection, Wi-Fi, and extended modules with optimal reliability, so that the smart home design system can meet the needs of users

**Keywords:** Wi-Fi Module, Android Application, Setup Software, I2C Communication, SPI

---

#### 1. Pendahuluan

Setiap orang pasti memiliki banyak yang menginginkan rumah yang nyaman dan aman[1]. Namun, ada beberapa orang yang merasa tidak nyaman ketika meninggalkan rumah untuk waktu yang lama, seperti halnya keamanan rumah jika pemiliknya lupa mengunci pintu, memungkinkan pencuri memasuki rumah atau ketika pemiliknya lupa mematikan listrik peralatan, kompor gas, dan lampu rumah. Ini bisa menyebabkan kerugian besar bagi penghuni rumah. Dari masalah yang terjadi, dibangun sistem rumah pintar yang telah diprogram untuk mengintegrasikan dan mengendalikan perangkat atau peralatan rumah tangga secara otomatis dan efisien. Proyek Rumah Pintar dimulai pada awal 1980-an sebagai proyek Pusat Penelitian Nasional untuk Asosiasi Pembangunan Perumahan Nasional (NAHB) dengan kerjasama dari kumpulan utama mitra industri [2]. Meski rumah pintar saat ini bukanlah hal baru, sistem rumah pintar paling canggih yang ada saat ini membutuhkan perubahan infrastruktur besar [3]. Sistem rumah pintar dan keamanan rumah biasanya fokus pada satu sistem tertentu, seperti sistem kunci pintu, alarm gas, atau hanya menyalakan dan mematikan lampu, sehingga sistem yang berbeda diperlukan untuk

menjalankan [4]. Tujuan dari perancangan sistem ini adalah untuk memudahkan pengguna dalam meningkatkan keamanan, dan kenyamanan dengan menggunakan RTOS (Real Time Operating System). RTOS pada sistem ini adalah sistem operasi yang digunakan untuk memenuhi kebutuhan aplikasi yang bersifat real time, yang berarti membutuhkan waktu kinerja pada saat itu ketika sistem sedang berjalan. Sistem ini dapat mengontrol peralatan rumah tangga sesuai dengan instruksi pengguna dari fungsi-fungsi dalam perangkat lunak pengaturan atau aplikasi android secara real time, sehingga data yang dikirim dari aplikasi atau pengaturan perangkat lunak akan diterima oleh papan pengontrol dan akan dieksekusi sesuai dengan data yang diterima. Fungsi akan sesuai dengan hasil, jika dijalankan menggunakan Wi-Fi dan jaringan komunikasi data I2C [5].

## 2. Dasar Teori

### 2.1 Papan Pengontrol

Papan pengontrol yang digunakan adalah Mikroprosesor tertanam STM32F407VGT6 memiliki banyak keunggulan seperti konsumsi daya yang rendah, kinerja tinggi, dan hemat biaya. Frekuensi jam operasi dapat mencapai 168MHz dalam tegangan pengenal dengan CPU 32-bit CPU Cortex-M4 dan fungsi FPU yang kuat. Selain mikroprosesor, ada mikroprosesor RISC STM32F103VET6 32bit. Prosesor bekerja pada 72MHZ, dengan karakter kinerja yang kuat dan konsumsi daya yang rendah, waktu nyata, dan biaya rendah [6].

### 2.2 Port I / O

Pada papan pengontrol, ada 8 port I / O yang terhubung ke sensor, yaitu sensor PIR, sensor gas MQ-2, dan sensor LDR. Sensor PIR digunakan untuk mendeteksi gerakan dari rentang sensor. Sensor PIR terbuat dari sensor piroelektrik yang dapat mendeteksi tingkat radiasi inframerah [7]. Sensor gas MQ-2 adalah modul deteksi keamanan yang digunakan untuk mendeteksi kebocoran gas di dalam ruangan. Ketika konsentrasi gas yang mudah terbakar di ruang yang terdeteksi mencapai nilai tertentu, modul kontrol utama akan segera mengirim informasi ke pengontrol, dan pengguna, pada saat yang sama memicu perangkat alarm [8]. Dalam sistem saat ini, sensor LDR digunakan untuk mendeteksi cahaya sebagai pengontrol cahaya, sehingga ketika level cahaya rendah resistansi akan tinggi [9].

### 2.3 Real Time Clock

RTC (Real Time Clock) yang digunakan dalam sistem saat ini adalah DS2302. Alat ini berfungsi untuk menghitung dan menjalankan jam (detik, menit, jam) dan kalender (hari, tanggal, bulan, tahun) secara akurat dan menyimpan data waktu secara real time. Alat ini dianggap akurat sebagai penghitung waktu. Perangkat ini dapat dikonfigurasi menggunakan perangkat lunak yang Free RTOS yang terdiri dari Tasks.c, Queue.c, List.c, dan Port.c untuk mengatur waktu, antrian pesan, dan fungsi manajemen memori. Dalam FreeRTOS, setiap utas eksekusi disebut tugas yang tidak memiliki batasan jumlah tugas. Saat tugas berjalan dengan laba, tugas yang memiliki prioritas tinggi dalam status siap pakai dapat menggunakan CPU untuk memastikan persyaratan sistem *Real-Time* [10].

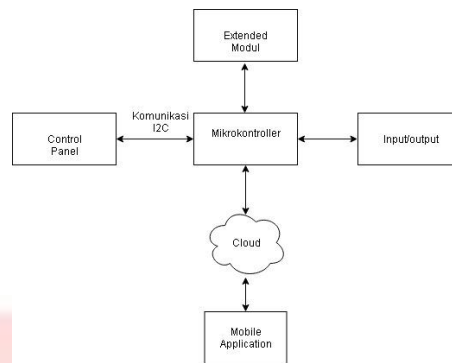
### 2.4 Modul Komunikasi

Sistem yang digunakan saat ini, menggunakan komunikasi serial dan Wi-Fi sebagai konektivitas sehingga sistem dapat terhubung dengan aplikasi android dan aplikasi desktop. Komunikasi Serial I2C adalah protokol untuk komunikasi serial antara IC, dan sering disebut Two Wire Interface (TWI). Bus digunakan untuk komunikasi antara mikrokontroler dan perangkat periferil seperti memori, sensor suhu dan I/O expander. Komunikasi dilakukan melalui dua jalur: SDA (data serial) dan SCL (serial clock). Setiap perangkat I2C memiliki alamat 7-bit yang unik [11]. Komunikasi serial ini mendukung komunikasi lebih dari dua perangkat dan komunikasi dua arah karena setiap perangkat dapat bertindak sebagai master. Setiap perangkat diprogram dengan alamat masing-masing dan pada saat suatu perangkat master mengirimkan data ke alamat tertentu, maka data akan melalui kedua pin komunikasi ini dan semua perangkat yang terhubung akan menerima data tersebut, tetapi hanya alamat yang dikirim yang dapat mengenali data tersebut [12]. Modul Wi-Fi ESP8266 adalah modul Wi-Fi yang penggunaannya harus dengan beberapa tahap konektivitas untuk dapat terhubung ke papan pengontrol. Modul ini harus memberikan kekuatan 3,3v [13].

## 3. Perancangan dan Implementasi

### 3.1. Gambaran Umum Sistem

Pada bagian ini akan menjelaskan sistem dalam pengontrolan dan monitoring keamanan rumah, dimana sistem ini berfungsi untuk mematikan dan menyalakan lampu berdasarkan pergerakan orang, relay akan menyala berdasarkan nilai suhu dan berdasarkan *controlling user, security system* pada pintu rumah menggunakan RFID, dan *alarm control system*, seperti saat akan terjadi kebakaran yang dimana alarm akan berbunyi jika nilai gas melewati nilai yang sudah ditentukan. Sistem tersebut dapat dikontrol dan dimonitoring melalui *control panel* dan *application*. Fungsi tersebut dapat dilihat pada gambaran umum sistem rumah pintar dengan menggunakan akses internet (*Wi-Fi*), dan Komunikasi I2C.



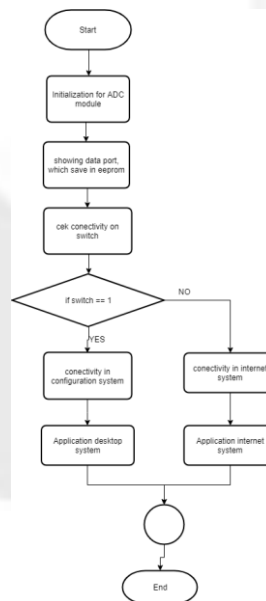
Gambar 1 Gambaran Umum Sistem Saat Ini

Berdasarkan Gambar 1 tersebut, terdapat mekanisme dari sistem yang telah diberi garis secara umum, yaitu sebagai berikut:

1. Control panel akan terhubung dengan microcontroller melalui Komunikasi Serial. Pada control panel ini dilakukan konfigurasi yang dapat mengaktifkan dan menonaktifkan bagian input dan output sesuai keinginan pengguna dan dilakukan verifikasi security system. Data tersebut dapat disimpan pada Antares.
2. Microcontroller berfungsi untuk memberikan dan memeriksa sinyal pada masing-masing port yang digunakan. Microcontroller juga akan meminta aktivasi pada modul extended, lalu extended akan mengirimkan aktivasinya ke microcontroller melalui komunikasi USB.
3. Log aktivasi yang sudah dikirim oleh modul extended akan diteruskan ke server oleh controller melalui jaringan Wi-Fi.
4. Aplikasi akan meminta data aktivasi pada server, lalu akan menampilkan data tersebut.
5. Aplikasi mobile mengambil data log dan konfigurasi pada server dan menampilkannya.

### 3.2 Perancangan Perangkat Lunak

Perancangan yang dilakukan pada perangkat lunak, berupa sistem operasi pada papan pengontrol secara *real time*. Perancangan yang dilakukan pada program utama terdiri dari dua proses komunikasi, yaitu proses program utama dengan proses interupsi data yang diterima oleh aplikasi dekstop, dan proses program utama dengan proses mengirim dan menerima data ke Antares melalui ESP8266. Pada proses program utama dengan proses interupsi data terdapat beberapa inialisasi pada setiap kanal dan modul, memuat konfigurasi dari penyimpanan *internal* dan menerima dan memberikan sinyal setiap kanal I/O.

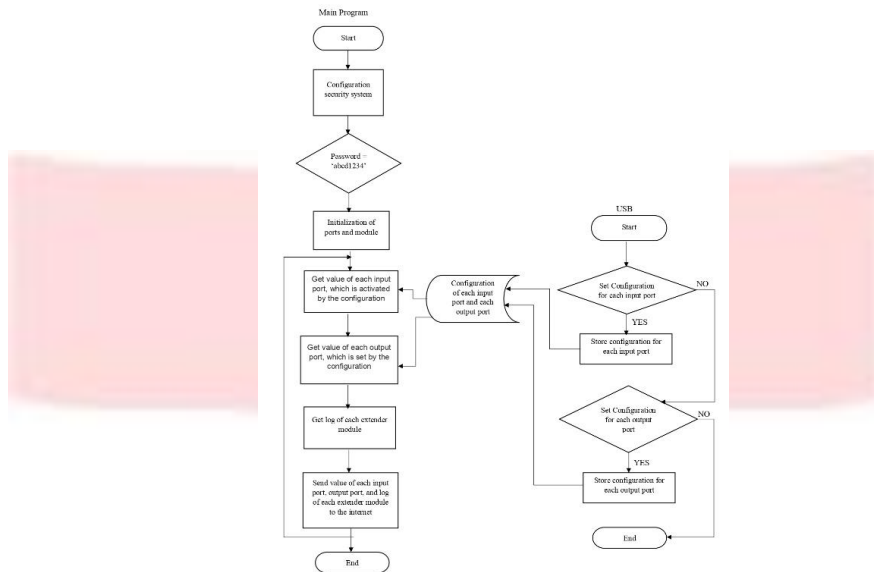


Gambar 2 Flowchart keseluruhan program mikrokontroler

Adapun penjelasan dari proses keseluruhan pada program mikrokontroler, seperti pada Gambar 2 sebagai berikut:

- a. Melakukan inialisasi untuk semua modul ADC, untuk memastikan semua sistem dan *task* dapat berjalan dengan sesuai.

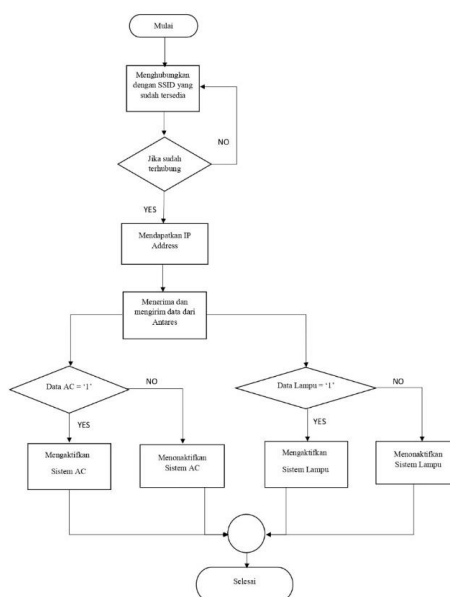
- b. Setelah melakukan inialisasi, maka langkah selanjutnya adalah mengecek konektivitas yang sedang berjalan, apakah koneksi konfigurasi atau koneksi internet. Pada kondisi ini koneksi konfigurasi bernilai '1' sedangkan pada koneksi internet bernilai '0'. Nilai tersebut dilihat dari nilai switch yang digunakan.
- c. Jika switch berada diposisi '1' maka koneksi yang sedang berjalan adalah koneksi konfigurasi dan jika posisi switch bernilai '0' maka koneksi yang sedang berjalan adalah koneksi internet.



Gambar 3 Flowchart Komunikasi Mikrokontroler dengan USB

Adapun penjelasan dari proses komunikasi antara program utama dengan komunikasi USB, seperti pada Gambar 3, sebagai berikut:

- a. Melakukan verifikasi *security system* pada aplikasi desktop. Jika *password* pada aplikasi dekstop sesuai dengan *password* yang sudah ditanam pada mikrokontroler, maka mikrokontroler akan langsung masuk ke bagian inialisasi awal yaitu konfigurasi port.
- b. Inialisasi pada port *input* sebagai port masukan dan port *output* sebagai port keluaran. Masing-masing port akan mendapatkan nilai sinyal, yang mana port masukan akan aktif jika dilakukan konfigurasi, dan port keluaran akan di aktifkan berdasarkan peraturan dari port masukan.
- c. Apabila port masukan dan port keluaran sudah diaktifkan, maka akan di simpan pada eeprom agar pengguna tidak melakukan setting awal pada port masukan dan keluaran, sehingga pengguna hanya dapat melakukan penambahan setting pada port masukan dan port keluaran.



Gambar 4 Alur Proses Program Utama dengan Komunikasi Antares

Adapun penjelasan dari proses program utama dengan proses komunikasi ke Antares, seperti pada Gambar 4 sebagai berikut:

- Mikrokontroler akan mendapatkan IP Address apabila sudah terhubung dengan SSID yang sudah tersedia, jika mikrokontroler sudah mendapatkan IP *address* maka mikrokontroler dapat terhubung dengan Antares.
- Setelah mikrokontroler terhubung dengan Antares, maka mikrokontroler akan menerima data dari Antares berupa '1' dan '0' untuk mengaktifkan dan menonaktifkan system AC dan sistem Lampu. Jika mikrokontroler tidak dapat terhubung dengan Antares, maka mikrokontroler akan terus mencoba menghubungkan lagi dengan SSID yang sudah tersedia hingga dapat menerima data dari Antares.

### 3.3 Implementasi

Dalam penelitian ini, dilakukan tahapan implementasi pada komunikasi I2C yaitu pengguna akan melakukan konfigurasi pada bagian *security system* di aplikasi desktop, seperti saat sebelum pengguna melakukan konfigurasi pada port, maka pengguna harus memasukkan *password*. Setelah itu, pengguna dapat melakukan konfigurasi port yang terhubung dengan *controller*, sehingga *control panel* dapat mengirimkan data setup ke program utama, lalu menyimpan data setup ke eeprom. Selain itu akan dilakukan tahapan implementasi pada aplikasi android yang dapat mengirim data ke *cloud* dan dapat menerima data dari *cloud*. Pada pengujian *microcontroller* akan di program untuk dapat menerima dan mengirim sinyal ke 8 port *input*, dan 8 port *output*. Sinyal yang akan dibaca oleh controller pada saat *input* adalah *HIGH* dan *LOW* yang akan dikonversikan ke bilangan biner 1 dan 0, sedangkan pada bagian *output microcontroller* akan membaca *ON* dan *OFF*. Pada pengujian RTC, program utama telah di program untuk membaca dan mengatur waktu yang telah berjalan.

## 4. Pengujian dan Analisis

Terdapat beberapa pengujian dan analisis yang telah dilakukan pada keseluruhan sistem, yaitu sebagai berikut:

### 4.1 Pengujian Konfigurasi dengan Aplikasi Desktop dan EEPROM

Pengujian ini dilakukan untuk memastikan bahwa papan pengontrol dapat melakukan komunikasi dengan aplikasi desktop melalui komunikasi serial dan dapat menyimpan dan membaca data pada EEPROM.

Tabel 1 Hasil Percobaan *Secret Code* Pada Aplikasi *Desktop*

Percobaan	<i>Secret Code</i>	Status
1	\$9ec8285e16edcb7	1
2	\$9ec8285e16edcb7	1
3	\$9ec8285e16edcb7	1
4	\$abcd	0
5	\$9ec8285e16edcb8	0
6	\$9ec8285e16edcb7	1
7	\$pqbc	0
8	\$9ec8285e16edcb7	1
9	\$9ec8285e16edcb7	1
10	\$9ec8285e16edcb7	1

Hasil Percobaan pada Tabel 1 ini menunjukkan bahwa *secret code* yang dikirim oleh aplikasi desktop dapat diterima oleh mikrokontroler dengan mengirimkan kembali angka '1' kepada aplikasi desktop. Adapun hasil yang tidak sesuai pada percobaan 4 dan 7, hal tersebut dikarenakan setelah aplikasi masukkan *code* yang tidak sesuai, maka mikrokontroler akan mengirim angka '0'. Setelah memasukkan *code* yang salah, jika kita memasukkan *code* yang benar maka *code* yang benar akan masuk bagian *setting*, sedangkan jika *code* salah bernilai 0, maka *setting*-annya pun bernilai 0 dan jika setelah itu memasukkan *code* yang benar, maka nilai yang dikirim mikro akan kembali '1'.

```

COM9 - Tera Term VT
File Edit Setup Control Window Help
COM/
StartTask03 Run
21:16:26.706 -> 11100110011
EEPROM Write OK
EEPROM Read OK
p q b c
EEPROM Write OK
EEPROM Res
ite OK
EEPROM Read OK
1 1 1 0 0
I2C Not OK
72
Temp: 30, RH: 99
I2C Not OK
99
Temp: 31, RH: 79
I2C Not OK
72
Temp: 31, RH: 79
I2C Not OK
79
Temp: 31, RH: 79

```

Gambar 5 Hasil Konfigurasi Pada Port *Input* dan *Output*.

Berdasarkan hasil pengujian yang sudah dilakukan pada Gambar 5, maka pada saat mikrokontroler menerima '\$11110000111100000', maka mikrokontroler akan langsung menjalankan port *input* dan *output* yang telah diaktifkan oleh aplikasi desktop.

Tabel 2 Hasil Pengujian Penyimpanan Data Pada EEPROM

Percobaan	Data yang Dikirim	Data yang Dibaca
1	\$11100	\$11100
2	\$abcd	\$abcd
3	\$01000	\$01000
4	\$10000	\$10000
5	\$12345	\$12345
6	\$0	\$0
7	-	-
8	-	-
9	-	-
10	-	-
11	\$1	\$1
12	\$123456	\$12345
13	\$1110001	\$11100
14	\$000011	\$00001
15	\$11100	\$11100

Berdasarkan hasil pengujian pada Tabel 2 yang telah dilakukan, maka saat aplikasi menerima '1' dari mikrokontroler maka aplikasi dapat melakukan setting dan data *setting* tersebut akan langsung tersimpan pada eeprom dan jika pengguna ingin melihat data yang tersimpan pada eeprom sebelumnya, maka pengguna harus memasukkan kode '\$pqrs', maka mikro akan mengirim data tersebut ke aplikasi desktop. Data tersebut juga akan tetap tersimpan apabila tidak diberikan daya listrik. Pada percobaan 7 dan 10 EEPROM tidak dapat menerima data yang diberikan aplikasi android, sedangkan pada percobaan 12 hingga 14 data yang dibaca oleh mikrokontroler hanya 5 data, hal tersebut terjadi karena pada program eeprom hanya dapat menyimpan 5 data saja, sehingga data ke 6,7,8, ...,20 dst tidak terbaca.

#### 4.2 Pengujian Sensor PIR

Pengujian ini dilakukan agar dapat mengetahui sensitifitas sensor dalam mendeteksi objek baik pada jarak terdekat dan terjauh terhadap objek, Objek yang akan dideteksi dalam pengujian yaitu manusia. Dalam pengujian ini, akan dilakukan lima kali percobaan dari tiap objek pada jarak 1 hingga 5 meter dan berikut merupakan hasil pengujian sensitifitas sensor PIR terhadap objek.

Tabel 3 Hasil Pengujian Sensitifitas Sensor PIR Terhadap Objek

Objek	Jarak (Meter)	Percobaan	Respon Led
Gerakan Manusia	1	1	Menyala
		2	Menyala
	2	1	Menyala
		2	Menyala
	3	1	Menyala
		2	Menyala
	4	1	Menyala
		2	Menyala
	5	1	Menyala
		2	Menyala
	6	1	Menyala
		2	Menyala
	7	1	Menyala
		2	Tidak Menyala
	8	1	Menyala
		2	Tidak Menyala
	9	1	Menyala
		2	Tidak Menyala
	10	1	Menyala
		2	Tidak Menyala

Berdasarkan hasil pengujian pada Tabel 3 ini didapatkan bahwa sensor PIR dapat mendeteksi setiap pergerakan hingga jarak 10 meter, namun pada jarak 7 hingga 10 meter sensor sudah mulai sulit mendeteksi pergerakan manusia, dikarenakan tingkat sensitivitas yang sulit dijangkau.

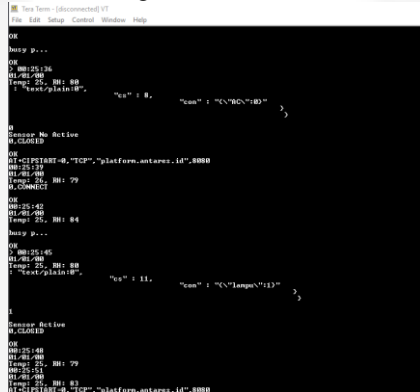
**4.3 Pengujian Modul Wi-Fi ESP8266 dan Aplikasi Android**

Pengujian ini dilakukan agar dapat memastikan bahwa mikrokontroler dapat terhubung dengan jaringan internet dan dapat mengirim dan menerima data melalui modul *Wi-Fi*. Pengujian yang dilakukan adalah pengujian koneksi dan pengujian dalam pengiriman dan penerimaan data ke/dari server Antares.

Tabel 4 Hasil Pengujian Koneksi Pada *Module Wi-Fi* ESP8266

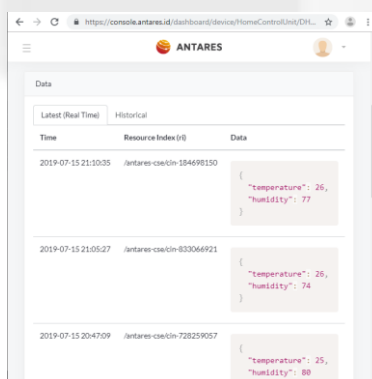
Jarak (meter)	Masih Terkoneksi dengan Kondisi	
	Tanpa Penghalang	Terhalang Dinding dan Pintu
± 1.2	Ya	Ya
± 2.4	Ya	Ya
± 3.6	Ya	Ya
± 4.8	Ya	Ya
± 6.0	Ya	Ya
± 7.2	Ya	Ya
± 8.4	Ya	Ya
± 18.0	Ya	Ya
± 24.0	Tidak	Tidak

Pada Tabel 4 menunjukkan bahwa modul *Wi-Fi* ESP8266 masih terkoneksi dengan baik pada jarak maksimal ± 18.0 meter dengan kondisi yang dimana tidak ada penghalang dan yang terhalang dinding dan pintu, sehingga mikrokontroler berhasil mengirim data ke Antares, sedangkan hasil pengujian pada pengiriman dan penerimaan data ke/dari server Antares adalah sebagai berikut.



Gambar 6 menerima data dari antares ke mikrokontroler

Pada Gambar 6 berdasarkan hasil pada serial monitor, data berhasil diterima oleh mikrokontroler, seperti “AC: 0” menandakan bahwa sistem AC tidak diakifkan, sedangkan “Lampu: 1”, maka sistem Lampu akan diaktifkan. Setelah mikrokontroler mendapatkan perintah tersebut, maka mikrokontroler siap untuk menjalankan sistem sesuai perintah pengguna, sedangkan pada Gambar 9 merupakan hasil dari mengirim data suhu dari mikrokontroler ke Antares untuk melakukan monitoring.



Gambar 7 Data Suhu yang Dikirim dari Papan Pengontrol

Berdasarkan hasil dari pengujian yang telah dilakukan pada Gambar 7, data yang terakhir tersimpan pada server Antares. Hasil pengujian pada ESP8266 adalah papan pengontrol dapat mengirim data I/O ke Antares berupa data suhu untuk monitoring dan nilai '1' untuk menandakan sistem sedang berjalan. Hasil Pengujian ini juga ditunjukkan pada Tabel 2, yaitu hasil konektivitas ESP8266 dengan jaringan internet, sehingga mikrokontroler dapat mengirim dan menerima data dari Antares.

## 5. Kesimpulan

Berdasarkan pengujian, perancangan, implementasi, dan analisis yang telah dilakukan, dapat disimpulkan sebagai berikut:

1. Papan pengontrol dapat dikonfigurasi dengan mengirimkan *secret code* dari aplikasi *desktop* melalui komunikasi USB serial dengan akurasi 100%.
2. Papan pengontrol dapat mengirimkan log aktivitasnya ke Antares pada jarak maksimal  $\pm 18$  meter melalui jaringan *Wi-Fi* ESP8266 dengan akurasi 88,88%.
3. Sensor PIR dapat mendeteksi gerakan dengan maksimal jarak 5 meter dengan akurasi 80%, karena pada meter ke 7 hingga 10 nilai sensitivitas yang dimiliki sensor PIR sudah mulai berkurang.
4. Pengaturan data dapat dikirim, ditampilkan, dan dibaca ke / dari aplikasi *desktop* melalui EEPROM dan dapat disimpan di EEPROM dengan akurasi 73,3%, karena dalam percobaan 7 hingga 10 ada kendala pada tegangan yang diterima oleh RTC .

## Daftar Pustaka:

- [1] X. Wen and Y. Wang, "Design of Smart Home Environment Monitoring System Based on Raspberry Pi," *2018 Chinese Control Decis. Conf.*, pp. 4259–4263, 2018.
- [2] B. Hamed, "Design & Implementation of Smart House Control Using LabVIEW," *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, no. May, 2017.
- [3] C. Engineering, "RTOS Based Home Automation System Using," pp. 910–915, 2015.
- [4] S. Karaca, "A Low Cost Smart Security and Home Automation System Employing an Embedded Server and a Wireless Sensor Network," *2016 International Conference on Consumer Electronics-Berlin 2016*.
- [5] A. Q. Bolaji, R. A. Kamaldeen, O. F. Samson, A. T. Abdullahi, and S. K. Abubakar, "A Digitalized Smart Mobile Home Automation and Security System via Bluetooth / Wi-Fi Using Android Platform," vol. 2, no. 6, pp. 93–99, 2017.
- [6] H. Sahala *et al.*, "SYSTEM DESIGN AUTOMATION AND SECURITY OF SMART HOME USING THE RASPBERRY PI 3 WITH CONTROL CENTER TELEGRAM," *2018 International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH)*, vol. 5, no. 1, pp. 1096–1103, 2018.
- [7] I. Kaur and A. Prof, "Microcontroller Based Home Automation System With Security," *2010 Fifth HCT Information Technology Trends (ITT)*, vol. 1, no. 6, pp. 60–65, 2010.
- [8] F. Masykur, F. Prasetyowati, P. Studi, T. Informatika, U. M. Ponorogo, and R. Pi, "APLIKASI RUMAH PINTAR ( SMART HOME ) PENGENDALI PERALATAN," vol. 3, no. 1, pp. 51–58, 2016.
- [9] Z. Xiaodong and Z. Jie, "Design and Implementation of Smart Home Control System Based on STM32," *2018 Chinese Control Decis. Conf.*, pp. 3023–3027, 2018.
- [10] H. Zhang and W. Kang, "Design of the Data Acquisition System Based on STM32," *Procedia Comput. Sci.*, vol. 17, pp. 222–228, 2013.
- [11] T. Akhir, J. T. Elektro, F. Sains, D. A. N. Teknologi, and U. S. Dhrama, "Pusat pengontrol lampu pada rumah pintar berbasis," 2017.
- [12] Q. Wang and Y. Wang, "Design and implementation of smart home remote monitoring system based on ARM11," *2018 Chinese Control Decis. Conf.*, pp. 209–213, 2018.
- [13] S. Priyadarshni, S. Sakthigurusamy, U. Susmedha, M. Suryapriya, and L. Sushmitha, "INTELLIGENCE HOME AUTOMATION SYSTEM USING LDR," pp. 140–146, 2017.