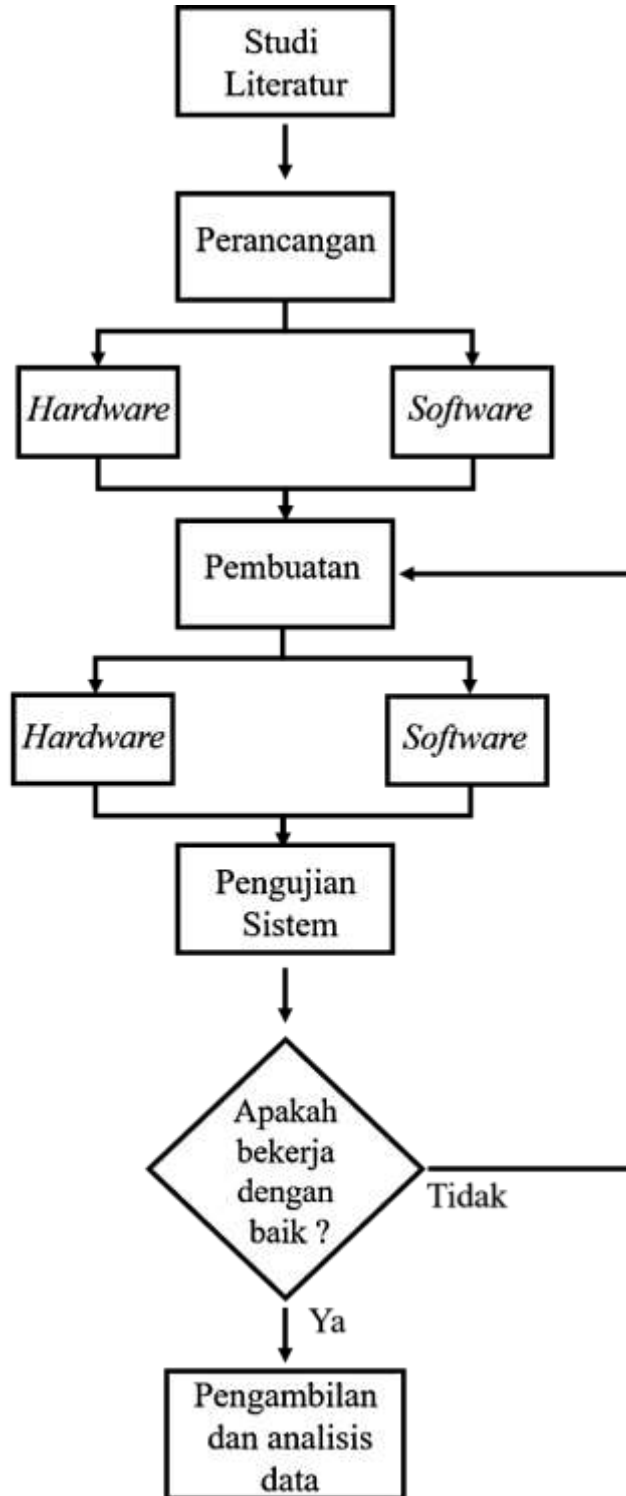


BAB III METODE PENELITIAN

3.1 Desain Penelitian

Penelitian ini akan menggunakan metode eksperimen dan deskriptif untuk mengembangkan sistem kendali suhu oven listrik dengan menerapkan *Internet of things* (IoT). Metode eksperimen akan melibatkan serangkaian uji coba yang meliputi pengukuran suhu oven pada berbagai *setpoint*, pengujian performa oven dalam mencapai suhu *setpoint* dengan variasi waktu dan sumber panas, serta evaluasi kinerja sistem kendali suhu. Uji coba ini akan melibatkan pembuatan alat sistem kendali secara keseluruhan, termasuk perancangan dan pembuatan *hardware* serta pengembangan *software* untuk mengendalikan oven secara otomatis. Selain itu, metode deskriptif akan digunakan untuk melakukan studi literatur mengenai konsep fisika yang terkait dengan sistem kendali suhu dan penggunaan sensor dalam oven. Studi literatur ini akan melibatkan pemahaman tentang prinsip kerja sensor suhu, peran sensor dalam pengukuran suhu yang akurat, dan konsep fisika terkait transfer panas dalam oven.

3.2 Prosedur Penelitian



Gambar 3.1. Diagram Prosedur Penelitian

3.2.1 Studi Literatur

Dalam tahap ini, dilakukan penelusuran literatur yang berkaitan dengan oven berbasis *Internet of things* (IoT). Tujuan dari penelusuran literatur ini adalah untuk memperoleh pemahaman yang mendalam tentang konsep, teknologi, dan pengembangan terkait oven IoT. Penelusuran literatur dilakukan melalui berbagai sumber, seperti jurnal ilmiah, konferensi, publikasi, dan literatur terkait lainnya yang telah diterbitkan oleh peneliti dan ahli di bidang ini. Penelusuran literatur ini melibatkan pengumpulan informasi tentang prinsip kerja oven IoT, komponen-komponen yang digunakan, metode pengendalian suhu, teknik konektivitas internet, serta aplikasi dan manfaat oven IoT dalam kehidupan sehari-hari. Selain itu, juga dilakukan analisis terhadap penelitian dan studi terkait yang telah dilakukan sebelumnya untuk memperoleh wawasan yang lebih mendalam tentang pengembangan oven IoT.

3.2.2 Perancangan *Hardware* dan *Software*

Tahap pengembangan alat dalam penelitian ini dilakukan untuk merancang alat oven berbasis *Internet of things* (IoT) yang terdiri dari *hardware* dan *software*. Pada perancangan, meliputi pembuatan diagram rangkaian dan diagram sistem instrumen untuk oven IoT. Dalam proses ini, Proteus digunakan sebagai aplikasi untuk membuat skematik dan rangkaian pada *breadboard* secara virtual sebelum melanjutkan perancangan pada perangkat ESP32 yang sebenarnya.

Selanjutnya, pada tahap perancangan *software*, meliputi pembuatan diagram alur program yang akan digunakan dalam sistem kendali dan *monitoring* oven IoT. Dalam tahap ini, pemilihan bahasa pemrograman yang sesuai juga menjadi pertimbangan penting. Bahasa pemrograman yang digunakan harus mendukung fungsi dan fitur yang dibutuhkan dalam pengendalian suhu, pengaturan waktu, pemantauan *real-time*, serta integrasi dengan aplikasi Android atau *platform* lainnya yang digunakan untuk mengontrol oven secara *remote*.

3.2.3 Pembuatan *Hardware* dan *Software*

Pada tahap pembuatan alat ini, dilakukan penggabungan dan integrasi

elemen-elemen yang telah dirancang sebelumnya menjadi sebuah sistem yang terpadu. Mikrokontroler ESP32 dihubungkan dengan berbagai komponen seperti sensor suhu, elemen pemanas, relay, dan LCD. Mikrokontroler ESP32 tersebut juga ditempatkan di samping luar oven guna menjaga keamanan dan fungsionalitasnya.

Selain itu, tahap pembuatan alat ini melibatkan pengembangan *software* yang mendukung fungsionalitas oven IoT. Melalui pemrograman, *software* akan mengatur pengendalian suhu, pengaturan waktu, serta koneksi dengan platform Firebase untuk menyimpan dan mengelola data oven secara *real-time*. Selain itu, antarmuka pengguna pada aplikasi juga dibangun untuk memberikan kendali dan pemantauan oven melalui perangkat Android.

Dalam proses ini, pemasangan relay dan elemen pemanas menjadi faktor utama dalam pengendalian suhu. Relay digunakan sebagai pengendali daya yang mengatur pengaktifan dan penghentian elemen pemanas. Melalui *software*, relay akan dikendalikan oleh mikrokontroler ESP32 untuk mengatur suhu oven dengan presisi. Elemen pemanas merupakan komponen utama dalam oven IoT yang bertanggung jawab untuk memanaskan ruang oven dan mencapai suhu yang diinginkan. Elemen pemanas tersebut akan terhubung dengan relay dan dikendalikan oleh *software* yang berjalan di mikrokontroler. Dengan demikian, suhu oven dapat diatur dan dipertahankan sesuai dengan *setpoint* yang telah ditentukan.

Dengan pemasangan relay dan elemen pemanas yang tepat serta pengujian yang cermat, diharapkan oven IoT dapat beroperasi dengan efektif dan memberikan hasil pemanggangan yang optimal. Integrasi antara relay, elemen pemanas, dan mikrokontroler ESP32 sebagai otak sistem akan memungkinkan pengendalian suhu oven yang akurat dan stabil sesuai dengan kebutuhan aplikasi yang diinginkan.

3.2.4 Pengujian Sistem

Sebelum pengujian sistem, dilakukan pengujian masing-masing alat, dilakukan evaluasi terhadap komponen-komponen yang digunakan dalam sistem

oven IoT. *Hardware* seperti sensor suhu, relay, dan elemen pemanas diuji untuk memastikan kinerjanya yang optimal. Pengujian dilakukan untuk memverifikasi bahwa sensor suhu mampu membaca suhu secara akurat, relay dapat mengendalikan elemen pemanas dengan baik, dan elemen pemanas memberikan respons yang sesuai dengan sinyal relay.

Setelah tahap pengujian alat, dilakukan pengujian sistem oven IoT secara keseluruhan. Pada tahap ini, sistem pengendalian suhu yang menggunakan relay dan metode kendali *on-off* diimplementasikan dalam lingkungan yang sesuai. Oven diatur untuk mencapai suhu *setpoint* tertentu, dan sistem pengendalian suhu diuji untuk memastikan bahwa elemen pemanas dihidupkan dan dimatikan secara tepat waktu berdasarkan perbedaan antara suhu aktual dan *setpoint*.

Selama pengujian sistem, dilakukan pemantauan suhu oven dan pemantauan kinerja sistem pengendalian suhu. Data suhu yang tercatat dan waktu respons relay untuk menghidupkan dan mematikan elemen pemanas dianalisis untuk mengevaluasi efektivitas sistem. Pengujian ini bertujuan untuk memverifikasi bahwa sistem pengendalian suhu mampu menjaga suhu oven dalam kisaran toleransi yang diinginkan dengan mengatur penghidupan dan pemadaman elemen pemanas secara on-off.

Selain pengujian terhadap pengendalian suhu, dilakukan pula pengujian komunikasi antara aplikasi pengguna dengan oven IoT. Pengujian ini bertujuan untuk memastikan bahwa aplikasi dapat terhubung dengan oven melalui jaringan internet, dan data suhu serta instruksi pengendalian suhu dapat dikirim dengan baik antara kedua perangkat. Pengujian dilakukan dengan mengirim perintah pengaturan suhu dari aplikasi ke oven, dan memantau apakah oven merespons dengan mengubah suhu sesuai instruksi.

3.2.5 Pengambilan Data dan Analisis Data

Tahap pengambilan data dan analisis data melibatkan proses pengumpulan data suhu oven serta analisis data tersebut untuk mendapatkan wawasan yang lebih dalam mengenai kinerja oven. Pada tahap pengambilan data, sensor suhu oven akan

mengukur dan mencatat data suhu secara periodik. Data suhu ini kemudian dikirim ke sistem komunikasi oven IoT untuk disimpan dalam *database*. Setelah data tersimpan, dilakukan analisis data untuk memahami waktu respon, pola perubahan suhu, dan kinerja oven dalam mencapai *setpoint* suhu yang diinginkan. Waktu respon meliputi *delay time*, *rise time*, *peak time*, *settling time*. *Delay time* mengukur waktu tunda respons sistem dalam mencapai separuh dari *setpoint* suhu. *Rise time* mengindikasikan waktu yang dibutuhkan oleh sistem untuk mencapai *setpoint* suhu secara penuh. *Peak time* menggambarkan waktu saat suhu mencapai nilai puncak dalam osilasi setelah mencapai *setpoint*. Sedangkan *settling time* mengukur waktu yang dibutuhkan oleh sistem untuk mencapai suhu stabil dalam rentang toleransi yang ditentukan (5% atau 2%). Melalui analisis data ini, dapat dievaluasi performa oven dalam hal respons sistem terhadap perubahan suhu, kecepatan mencapai *setpoint*, dan kemampuan menjaga suhu stabil.

3.3 Alat dan bahan

Dalam penelitian ini, fokus utama yaitu untuk mengembangkan sistem kendali suhu pada oven berbasis *Internet of things* (IoT). Sistem ini melibatkan penggunaan alat dan bahan yang mendukung untuk pembuatan sistem tersebut, seperti sensor suhu, mikrokontroler, relay, dan komponen lainnya yang tercantum pada tabel 3.1.

Tabel 3.1. Daftar komponen elektronika yang digunakan

Komponen Elektronika	Jumlah
ESP32	1 buah
Termokopel tipe K +MAX6675	1 buah
Relay	1 buah
LCD 16x2 I2C	1 buah

Komponen-komponen tersebut diaplikasikan pada oven yang dirancang untuk memonitor dan mengendalikan suhu dengan presisi. ESP32 berfungsi sebagai otak utama oven, bertanggung jawab atas pengambilan data suhu dari termokopel tipe K melalui

sensor MAX6675. Data suhu tersebut kemudian ditampilkan secara *real-time* pada LCD 16x2 I2C yang terhubung dengan ESP32. Selain itu, oven ini dilengkapi dengan relay yang berfungsi sebagai saklar pengendali daya untuk mengatur suhu di dalam oven. Relay ini dapat mengontrol sumber daya yang mengalir ke elemen pemanas, sehingga suhu di dalam oven dapat diatur sesuai dengan kebutuhan. Ketika suhu mencapai batas yang ditentukan, relay akan menghentikan aliran daya ke elemen pemanas, mempertahankan suhu yang diinginkan. Spesifikasi oven yang digunakan pada penelitian ini dicantumkan pada tabel 3.2.

Tabel 3.2. Spesifikasi oven yang digunakan

Material Internal	<i>Galvanized Plate</i>
Kapasitas Oven	12 Liter
Voltase	220 Volt
Frekuensi	50 Hz
Daya	600 Watt
Massa	2,05 Kg
Dimensi	355x 265 x 240 mm

3.3.1 Mikrokontroler ESP32

Mikrokontroler yang digunakan dalam penelitian ini adalah ESP32 dengan chip Tensilica Xtensa LX6 yang memiliki kecepatan clock sebesar 160/240MHz. ESP32 menawarkan berbagai fitur dan fasilitas yang penting dalam pengembangan sistem IoT. Selain itu, ESP32 juga didukung oleh berbagai *library* yang tersedia secara luas dan terus dikembangkan oleh komunitas. Keberadaan *library* open-source ini memberikan keuntungan dalam hal fleksibilitas dan kemampuan untuk melakukan iterasi dan penyesuaian yang lebih baik dalam pengembangan sistem.



Gambar 3.2. Mikrokontroler ESP32

Gambar 3.2 menunjukkan rupa dari Mikrokontroler ESP32. ESP32 memiliki total 25 GPIO (*General Purpose Input Output*), dengan 6 pin yang mendukung Analog to Digital Converter (ADC), 2 pin yang mendukung SPI interfaces, 3 pin yang mendukung UART (*Universal Asynchronous Receiver-Transmitter*) interfaces, 3 pin yang mendukung I2C (*Inter Integrated Circuit*) interfaces, 16 pin yang mendukung *Pulse Width Modulation* (PWM), 2 pin yang mendukung *Digital-to-Analog Converter* (DAC), 2 pin yang mendukung I2S (*Inter-IC Sound*) interfaces. Dalam penelitian ini, kita secara intensif memanfaatkan fitur ADC dari ESP32 karena selain mendukung pembacaan sinyal analog, ADC pada ESP32 juga digunakan untuk menghubungkan perangkat ke jaringan Wi-Fi dan mentransfer data ke database. Dengan kapasitas flash memory sebesar 4MB dan kecepatan *clock* sebesar 240MHz, ESP32 mampu menjalankan berbagai komputasi ringan dengan baik (Espressif, 2021).

3.3.2 Termokopel tipe K + MAX6675



Gambar 3.3. Termokopel tipe K + MAX6675

Gambar 3.3. menunjukkan rupa dari Termokopel tipe K + MAX6675. Pengukuran suhu pada oven menggunakan kombinasi termokopel tipe K dan modul MAX6675. Termokopel tipe K mampu mengukur suhu dalam rentang 0°C hingga +1024°C dengan resolusi sebesar 0.25°C (Maxim Integrated, 2021). Sementara itu, modul MAX6675 berfungsi sebagai pengolah sinyal yang bekerja sama dengan termokopel tipe K. Modul ini memiliki fitur penguat sinyal, konversi analog-ke-

digital, dan komunikasi serial SPI. Dengan resolusi pengukuran suhu yang tinggi dan waktu konversi yang cepat, modul MAX6675 mampu menghasilkan keluaran sinyal digital yang dapat dibaca oleh mikrokontroler. Melalui konverter ADC yang terdapat dalam modul, sinyal temperatur yang dikumpulkan oleh termokopel tipe K dapat diolah dan diubah menjadi sinyal digital (Septiana, dkk., 2019). MAX6675 beroperasi pada rentang tegangan 3V hingga 5V dan mengeluarkan data dalam resolusi 12-bit (Prayudha, dkk., 2022)

3.3.3 Relay

Komponen yang digunakan untuk mengendalikan suhu yaitu relay. relay digunakan sebagai perangkat pengatur yang dapat menghubungkan atau memutuskan aliran listrik yang menuju elemen pemanas berdasarkan sinyal yang diterima dari mikrokontroler. Dalam sistem ini, relay diatur dalam mode *Normally open* (NO). Pada mode NO, kontak relay dalam keadaan terbuka ketika tidak ada sinyal kendali yang diterima. Artinya, pada kondisi awal, arus listrik tidak dapat mengalir melalui relay. Namun, ketika sinyal kendali diaktifkan, relay akan menutup dan menghubungkan aliran listrik ke elemen pemanas oven.



Gambar 3.4. Relay

Gambar 3.4 menunjukkan rupa dari relay yang digunakan pada penelitian ini. Komponen relay yang digunakan dalam sistem ini bekerja pada tegangan operasional 5V dan mendapatkan daya dari mikrokontroler. tegangan dan arus yang dapat ditangani oleh relay ini adalah 10A 250VAC. Pengendali relay terhubung ke pin IO D1 pada mikrokontroler. Relay ini memiliki konfigurasi NO (*Normally open*) yang dapat menangani arus hingga 10A pada tegangan 250VAC. Selain itu, relay ini juga memiliki

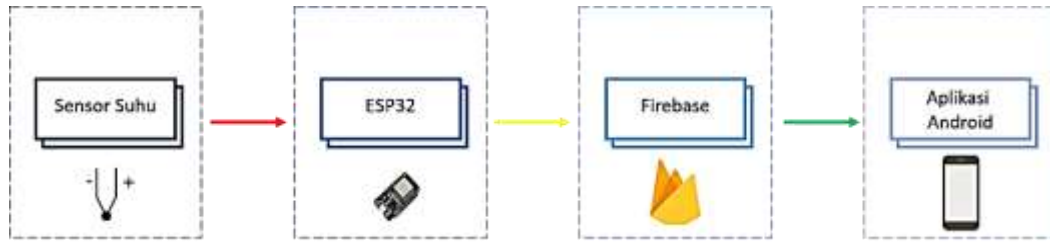
konfigurasi NC (*Normally Closed*) yang dapat menangani arus hingga 10A pada tegangan 250VAC. Terdapat 7 pin IO yang dapat dikonfigurasi, dengan konfigurasi default pada pin D1. Relay ini dilengkapi dengan pengendali transistor dan dioda freewheeling untuk melindungi dari tegangan balik induktif pada kumparan relay. Terdapat indikator LED yang menunjukkan status ON relay ketika aktif.

3.3.4 LCD I2C

LCD I2C 16x2 adalah sebuah layar LCD karakter yang menggunakan antarmuka I2C untuk berkomunikasi dengan mikrokontroler. Layar ini hanya memerlukan 4 pin untuk dikendalikan dan digunakan, sehingga lebih mudah dan rapi dibandingkan dengan menggunakan koneksi langsung ke mikrokontroler. LCD I2C 16x2 dilengkapi dengan sebuah adapter I2C yang menggunakan chip PCF8574 untuk mengubah data I2C dari mikrokontroler menjadi data paralel yang dibutuhkan oleh layar LCD. Layar LCD ini dapat menampilkan karakter ASCII sebanyak 32 karakter pada dua baris. Dengan menggunakan LCD I2C 16x2, pengguna dapat mengontrol dan menampilkan data pada layar LCD dengan lebih mudah dan rapi. Selain itu, pengguna juga dapat menampilkan karakter khusus pada layar LCD dengan menggunakan kode khusus pada program mikrokontroler.

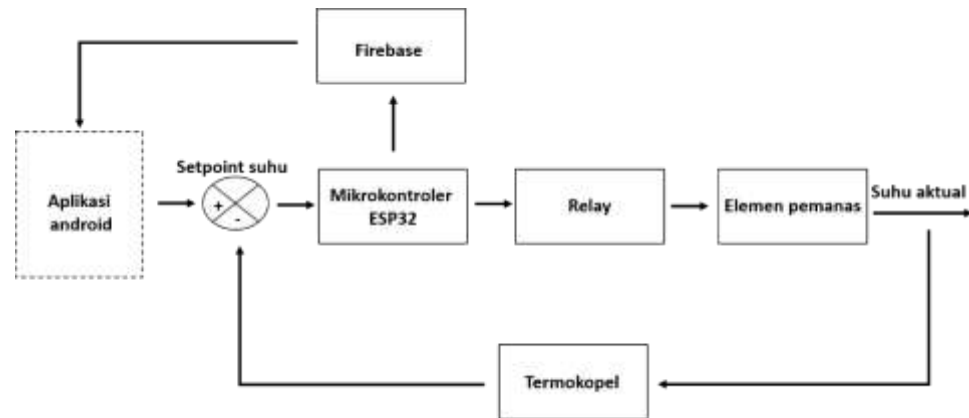
3.4 Sistem Data

Sistem komunikasi data *monitoring* suhu menggunakan ESP32, Firebase, dan aplikasi Android. Pada sistem ini, suhu diukur menggunakan termokopel tipe K yang terhubung ke ESP32. ESP32 berfungsi sebagai mikrokontroler yang mengambil data suhu dari termokopel dan mengirimkannya melalui koneksi WiFi ke Firebase Realtime Database. Firebase bertindak sebagai penyimpan data yang dapat diakses secara *real-time*. Aplikasi Android digunakan sebagai antarmuka pengguna untuk memantau suhu yang terekam dalam database Firebase. Melalui aplikasi Android, pengguna dapat melihat data suhu oven terkini.



Gambar 3.5. Proses *monitoring*

Sistem yang dibangun bukan hanya bisa digunakan untuk *monitoring* suhu aktual, namun bisa juga digunakan mengendalikan suhu sesuai dengan input *setpoint*. Diagramnya ditunjukkan oleh gambar 3.6.

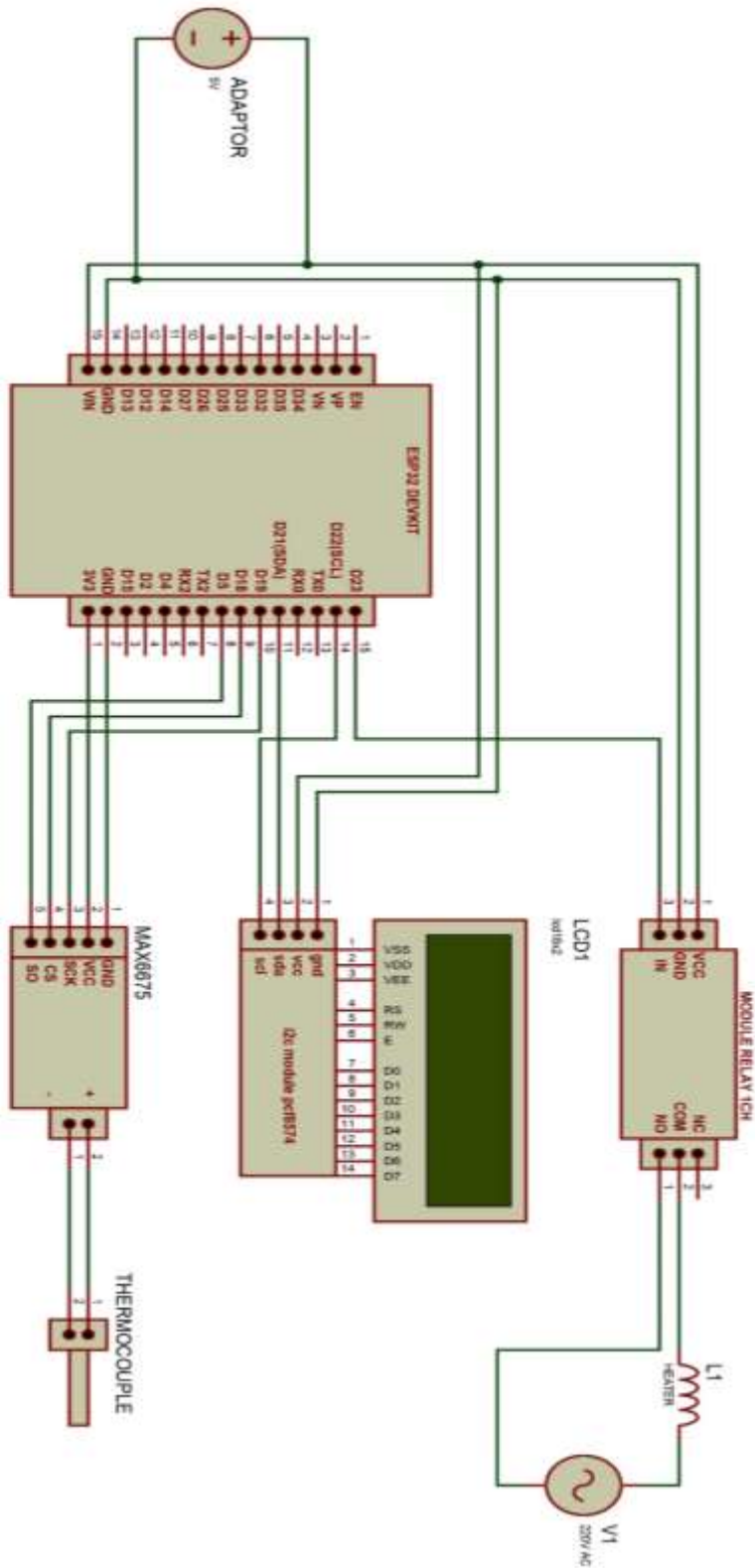


Gambar 3.6. Diagram sistem kendali

Sistem kendali suhu menggunakan diagram blok sebagai representasi visual dari komponen dan alur kerja sistem. Diagram blok ini menggambarkan bagaimana setiap komponen saling terhubung dan berinteraksi dalam mengendalikan suhu. Pada diagram blok sistem kendali suhu, terdapat beberapa komponen utama yang meliputi termokopel tipe K sebagai sensor suhu, ESP32 sebagai mikrokontroler, relay sebagai pengendali daya, dan aplikasi Android sebagai antarmuka pengguna. Input suhu *setpoint* berasal dari aplikasi Android dan diterima oleh ESP32 sebagai suhu acuan. Termokopel tipe K berfungsi untuk mengukur suhu dan mengubahnya menjadi sinyal listrik yang akan diteruskan ke ESP32. ESP32 kemudian akan menerima sinyal dari termokopel dan melakukan pemrosesan data suhu. Selain itu, suhu aktual dapat terbaca pada aplikasi android. Berdasarkan data suhu yang diterima, ESP32 akan mengontrol relay untuk

mengatur daya yang masuk ke elemen pemanas. Relay berperan sebagai pengendali daya yang akan menghubungkan atau memutuskan aliran daya ke elemen pemanas. Melalui kontrol dari ESP32, relay akan diaktifkan atau dinonaktifkan sesuai dengan suhu yang diinginkan. Hal ini memungkinkan sistem untuk menjaga suhu pada *setpoint* yang diinginkan.

Sistem ini mencakup beberapa komponen utama yang saling terhubung dan berinteraksi untuk memantau dan mengendalikan berbagai aspek dalam oven. ESP32 berperan sebagai otak dari sistem, dihubungkan dengan termokopel tipe K, relay dan LCD I2C. Rangkaian komponennya ditampilkan pada gambar 3.7.



Gambar 3.7. Rangkaian skematik

Dimas Syahril Maulana, 2023
**PENGEMBANGAN SISTEM KENDALI SUHU OVEN LISTRIK BERBASIS MIKROKONTROLER ESP32 DENGAN
 MENERAPKAN INTERNET OF THINGS**

Gambar 3.7 merupakan skematik yang menggambarkan hubungan antara mikrokontroler ESP32 dengan termokopel tipe K, relay dan LCD I2C. Skematik ini dirancang untuk memudahkan pengaturan hubungan I/O antara mikrokontroler dan komponen-komponen tersebut. Termokopel tipe K digunakan sebagai sensor suhu yang akan mengukur suhu pada oven, sedangkan relay berfungsi sebagai pengendali daya yang mengatur pemanasan pada elemen pemanas. Melalui skematik ini, mikrokontroler ESP32 dapat menerima sinyal suhu dari termokopel, dan berdasarkan nilai suhu tersebut, mengontrol operasi relay untuk mengatur daya yang masuk ke elemen pemanas.

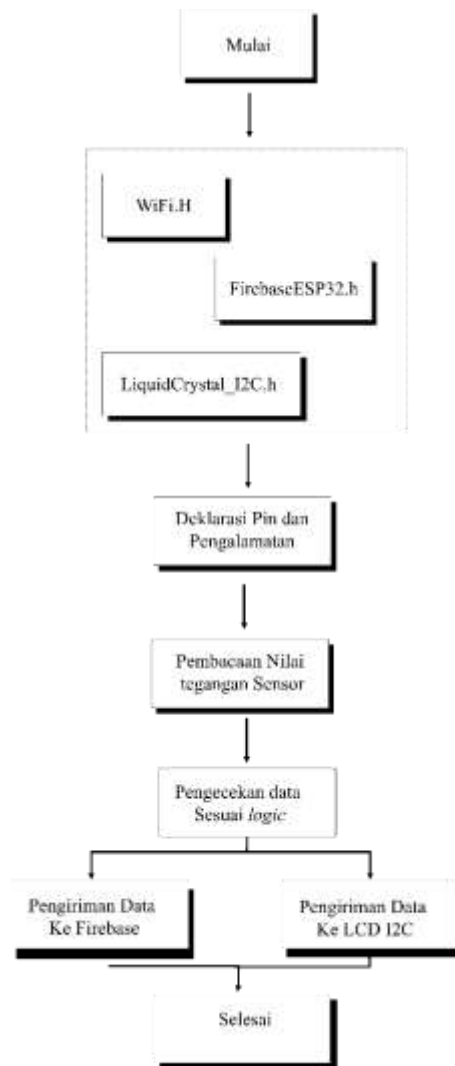
Tabel 3.3. Konektifitas Komponen

Komponen	Pin Komponen	Pin ESP32
Relay	VCC	5V
	GND	GND
	Input	GPIO D23
LCD I2C	VCC	5V
	GND	GND
	SDA	GPIO 21
	SCL	GPIO 22
Termokopel tipe K & MAX6675	VCC	5V
	GND	GND
	SCK	GPIO 19
	CS	GPIO 18
	SO	GPIO 5
Elemen pemanas	COM	
	220V AC-NO	

Tabel 3.1 menunjukkan hubungan antara pin-pin komponen yang digunakan dalam sistem ini dengan pin pada mikrokontroler ESP32. Relay terhubung dengan tiga pin, yaitu VCC, GND, dan Input yang terhubung ke pin 5V, GND, dan GPIO D23 pada ESP32 secara berurutan. LCD I2C terhubung dengan empat pin, yaitu VCC, GND, SDA, dan SCL yang terhubung ke pin 5V, GND, GPIO 21, dan GPIO 22 pada ESP32. Termokopel tipe K & MAX6675 terhubung dengan lima pin, yaitu VCC, GND, SCK, CS, dan SO yang terhubung ke pin 5V, GND, GPIO 19, GPIO 18, dan GPIO 5 pada ESP32. Sedangkan Elemen Pemanas terhubung dengan satu pin, yaitu COM pada relay yang terhubung ke sumber daya AC 220V dan pin NO pada relay. Dengan menggunakan

hubungan pin-pen ini, mikrokontroler ESP32 dapat mengendalikan dan berkomunikasi dengan berbagai komponen dalam sistem ini sesuai dengan fungsinya.

3.5 Perancangan dan Desain Alur Pemograman



Gambar 3.8. Alur pemograman

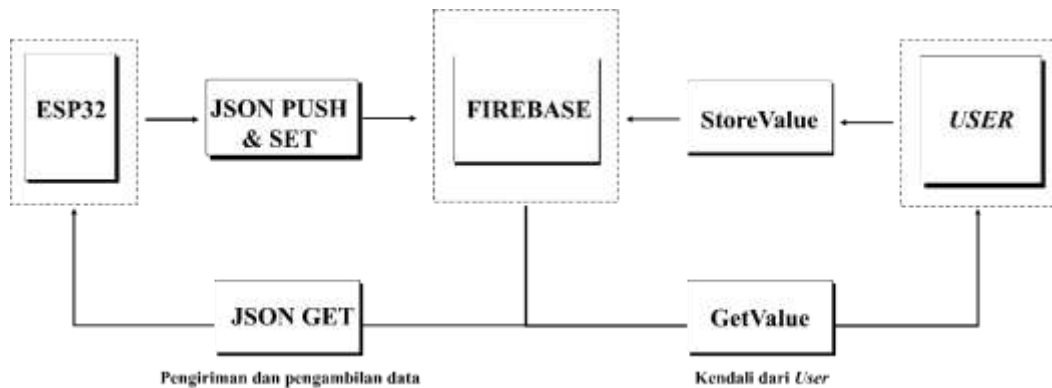
Pada sistem kendali dan *monitoring* suhu oven listrik, proses pemrograman ESP32 pada gambar 3.8 dimulai dengan mengimpor dan mendefinisikan *library* yang diperlukan. *Library* tersebut berfungsi untuk membaca data dari sensor suhu, mengirim data melalui koneksi WiFi, dan menampilkan informasi pada LCD I2C. Selain itu, dalam pemrograman ini juga dilakukan konfigurasi dan koneksi ke

jaringan WiFi menggunakan SSID dan *password* yang relevan. Selanjutnya, untuk menghubungkan dengan database Firebase Google, diperlukan pengaturan alamat host dan token otentikasi yang sesuai.

Deklarasi pin yang digunakan merupakan sinyal digital untuk membaca suhu aktual menggunakan modul max6675 pada GPIO 19, GPIO 18 dan GPIO 5. Deklarasi pin yang digunakan juga melibatkan sinyal digital. Pada LCD I2C, sinyal digital digunakan untuk mengatur komunikasi antara ESP32 dan modul LCD, dengan pin SDA terhubung ke GPIO 4 dan pin SCL terhubung ke GPIO 15. Sedangkan pada relay, sinyal digital digunakan untuk mengendalikan operasi relay dengan pin kontrol terhubung ke GPIO 23.

Di dalam fungsi setup, dilakukan konfigurasi awal seperti inisialisasi serial, pin mode, koneksi ke jaringan WiFi, koneksi ke Firebase, dan penyetelan waktu menggunakan NTPClient. Selain itu, terdapat pula pengaturan awal untuk tampilan LCD dan pemasangan fungsi *tasktimer* pada ticker. Di dalam fungsi *loop*, terdapat dua blok kondisional yang berjalan berdasarkan waktu yang telah ditentukan. Pada blok pertama, dilakukan pembacaan suhu menggunakan sensor MAX6675, pengambilan waktu menggunakan NTPClient, dan pembaruan tampilan LCD. Selanjutnya, data suhu, waktu, dan status relay dikirim ke Firebase menggunakan fungsi *send_firebase*. Pada blok kedua, terdapat pengecekan dan pengambilan data dari Firebase menggunakan fungsi *get_firebase*. Selain itu, dilakukan juga pengiriman data log dan notifikasi berdasarkan kondisi yang telah ditentukan.

Sensor dikonfigurasi dan dialamatkan menggunakan fungsi dari *library* yang digunakan untuk membaca voltase yang diterima dari sensor. Dalam hal ini, mikrokontroler ESP32 bekerja dengan kecepatan pembacaan data serial sebesar 115200 bit/s. Hal ini memungkinkan mikrokontroler untuk menerima input dan menghasilkan output nilai voltase dari sensor dan aktuator dengan frekuensi 115200 bit per detik. Hasil input dari sensor kemudian disimpan dalam variabel yang telah dideklarasikan sebelumnya untuk melakukan perintah berikutnya.



Gambar 3.9. Alur pengiriman dan pengambilan data

Setelah data berhasil dibaca dari sensor dan disimpan dalam variabel yang telah ditentukan, dilakukan pemrosesan matematika untuk menghasilkan data yang mudah dipahami oleh manusia. Data tersebut kemudian dikirim ke database Google dalam format *JavaScript Object Notation* (JSON). Format ini terdiri dari pasangan "key" atau definisi data yang dikirimkan, serta "value" yang merupakan nilai dari data tersebut. Kedua jenis nilai ini bisa berupa angka bulat (integer) atau teks (string). Pengiriman data dilakukan secara nirkabel melalui jaringan Wi-Fi yang telah dikonfigurasi dan menggunakan fitur yang sudah terintegrasi pada ESP32, sehingga tidak diperlukan perangkat tambahan seperti Raspberry-Pi.

Alur program pada sistem kendali suhu oven menggunakan ESP32, sesuai dengan gambar 3.8, melibatkan pengiriman data melalui koneksi Wi-Fi menggunakan metode JSON Push. Metode ini memungkinkan pengiriman data secara asinkronus ke *database*, sehingga seluruh data dapat disimpan selama *database* masih ada. Selain itu, terdapat juga metode JSON Set yang digunakan untuk melakukan perubahan data spesifik pada data yang sudah ada. Pengguna dapat memberikan input berupa suhu *setpoint* kepada mikrokontroler melalui antarmuka pengguna (*user interface*). Pengaturan ini dilakukan dengan menggunakan metode Set JSON untuk mengirimkan data spesifik yang ingin diubah, yaitu pengaktifan relay sebagai aktuator. ESP32 menggunakan metode JSON get untuk mengambil data. Jika nilai yang dipantau berubah, pengguna dapat melihat suhu terbaru melalui metode JSON get. Seluruh data yang dikirimkan dan diperoleh menggunakan sistem database NoSQL, yang memudahkan perubahan

database jika terdapat penambahan fitur secara *real-time*.

Dimas Syahril Maulana, 2023

**PENGEMBANGAN SISTEM KENDALI SUHU OVEN LISTRIK BERBASIS MIKROKONTROLER ESP32 DENGAN
MENERAPKAN INTERNET OF THINGS**

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu