

SKRIPSI

**PENERAPAN *MULTITASKING* PADA SISTEM HIDROPONIK
TEROTOMASI BERBASIS *INTERNET OF THINGS***

Disusun dan diajukan oleh

RYAN RAFLI

D421 15 010



**DEPARTEMEN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
MAKASSAR
2021**

LEMBAR PENGESAHAN SKRIPSI

**PENERAPAN *MULTITASKING* PADA SISTEM HIDROPONIK
TEROTOMASI BERBASIS *INTERNET OF THINGS***

Disusun dan diajukan oleh

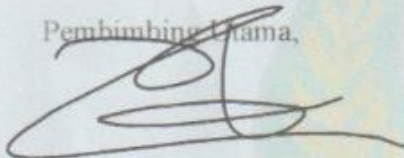
RYAN RAFLI

D421 15 010

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka
Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika Fakultas
Teknik Universitas Hasanuddin pada tanggal 27 Januari 2021
dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

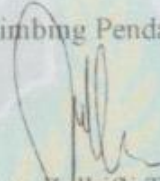
Pembimbing Utama,



Adnan, S.T., M.T., Ph.D.

Nip. 19740426 200501 1 002

Pembimbing Pendamping



Dr. Eng. Xulkifli Tahir, S.T., M.Sc.

Nip. 19840403 201012 1 004



Program Studi,

Amir Muhammad Ilham, S.T., M.IT

Nip. 19731010 199802 1 001

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama : RYAN RAFLI
NIM : D421 15 010
Program Studi : TEKNIK INFORMATIKA
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

PENERAPAN *MULTITASKING* PADA SISTEM HIDROPONIK TEROTOMASI BERBASIS *INTERNET OF THINGS*

Adalah karya tulis saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 02 Februari 2021

Yang Menyatakan



RYAN RAFLI

KATA PENGANTAR

Bismillahirrahmanirrahim

Puji dan Syukur penulis panjatkan kepada Allah *subhanahu wa ta'ala* yang karena limpahan rahmat dan karunia-Nya sehingga tugas akhir yang berjudul “PENERAPAN *MULTITASKING* PADA SISTEM HIDROPONIK TEROTOMASI BERBASIS *INTERNET OF THINGS*” ini dapat penulis selesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Strata-I (S-1) pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Penulis menyadari betul bahwa dalam penyusunan laporan tugas akhir ini tidak lepas dari bantuan, bimbingan, serta dukungan dari berbagai pihak, baik itu sejak masa perkuliahan hingga masa penyusunan tugas akhir ini dilakukan. Oleh karena itu, penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Kedua orang tua penulis, alm. Bapak **Appe Baskara** dan Ibu **Chomisa, S.Pd.I** yang selalu memberikan dukungan, doa, motivasi dan kasih sayang sejak penulis kecil hingga sampai pada keadaan saat ini;
2. Bapak **Adnan, S.T., M.T., Ph.D.**, selaku pembimbing 1 dan Bapak **Dr. Eng. Zulkifli Tahir, S.T., M.Sc.**, selaku pembimbing 2 yang menyediakan waktu, tenaga, pikiran, dan perhatian yang luar biasa untuk mengarahkan penulis dalam menyusun tugas akhir;
3. Bapak **Dr. Amil Ahmad Ilham, S.T., M.IT.**, selaku Ketua Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin;

4. Keluarga besar penulis yang senantiasa memberikan semangat dan motivasi kepada penulis;
5. Para sahabat di Lab Riset IoTPC Fakultas Teknik Universitas Hasanuddin: **Andi Nurul Sri Utami, Bayazid Sustami Mohammad Nasir, Chrisley Silambi Tandung, Fathur Rizqi, Muh. Arief Wicaksono, Nur Azizah Novitami Winarko, Sabtian Juliana, dan Said Syamil Amas**, sebagai teman bertukar pikiran yang telah memberikan begitu banyak masukan dan bantuan kepada penulis selama melakukan penelitian;
6. Keluarga besar Sapulidi: **Charina, Fadel Rezky Ramadhan, Fuad Khairi Hamid, Jusmiati, Khusnul Khatima, Laura Natalia Nainggolan**, (lagi-lagi) **Muh. Arief Wicaksono, Muhammad Zulfachril Asiari, Reka Regina**, (lagi-lagi) **Sabtian Juliana**, (lagi-lagi) **Said Syamil Amas**, dan **Umniyah Nur Aprilyah** yang senantiasa memberikan motivasi, nasehat dan dukungan kepada penulis selama masa perkuliahan penulis;
7. Para sahabat Villa Bujang: **Andi Ardiansyah Yusuf, Fiqar Aprialim**, (lagi-lagi) **Sabtian Juliana**, dan (lagi-lagi) **Said Syamil Amas** yang selalu ada menemani penulis siang dan malam;
8. Para sahabat Khayangan: **Ardya Dwi Rahmasari, Dian Indani**, (lagi-lagi) **Jusmiati**, dan (lagi-lagi) **Umniyah Nur Aprilyah** yang senantiasa membuka pintu untuk berbagi kisah dan pengalaman;
9. Para sahabat penulis: **Irfan Setiawan** dan **Rahmat Junaedi** yang tiada bosan memberikan motivasi untuk selalu melangkah ke depan;

10. Keluarga besar **Hyperv15or** Fakultas Teknik Universitas Hasanuddin atas dukungan dan semangat serta kebersamaan yang diberikan selama ini;
11. segenap Dosen dan staf Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah membantu penulis;
12. orang-orang berpengaruh lainnya yang tanpa sadar telah menjadi inspirasi penulis.

Akhir kata, penulis berharap semoga Allah *subhanahu wa ta'ala* berkenan untuk membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga tugas akhir ini dapat memberikan manfaat bagi pengembangan ilmu. Aamiin.

Wassalam

Gowa, Januari 2021

ABSTRAK

Perangkat *Internet of Things* akan semakin banyak penggunaannya disertai dengan semakin banyaknya variasi hubungan mikrokontroler, sensor, dan aktuator. Hal ini didasari pada penelitian *Business Insider Intelligence* yang memperkirakan akan ada lebih dari 55 miliar perangkat *Internet of Things* (termasuk sistem tersemat) pada tahun 2025. Banyaknya variasi tersebut mempengaruhi efisiensi sistem. Penerapan *multitasking* diperkirakan dapat memberikan pengaruh pada efisiensi sistem yang dibuat. Dengan menggunakan studi kasus hidroponik terotomasi yang memiliki cukup banyak parameter yang harus diperhatikan, *multitasking* dilakukan dengan menggunakan freeRTOS dan TridentTD_EasyFreeRTOS32. Dalam waktu 1 jam freeRTOS mampu mengerjakan tugas sebanyak 24.328 *task* tanpa mempengaruhi masing-masing *task* yang bekerja. Sedangkan TridentTD_EasyFreeRTOS32 mampu mengerjakan tugas sebanyak 23.839 *task* dalam rentang waktu 1 jam. Perbandingan antara penggunaan program dengan penerapan *multitasking* (program dengan freeRTOS dan program dengan TridentTD_EasyFreeRTOS32) dengan program tanpa RTOS adalah sebesar 13:1 *task*.

Kata Kunci : *Internet of Things*, *multitasking*, freeRTOS, TridentTD_EasyFreeRTOS32, hidroponik terotomasi.

DAFTAR ISI

LEMBAR PENGESAHAN SKRIPSI	i
KATA PENGANTAR.....	iii
ABSTRAK	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	x
DAFTAR TABEL	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan Penelitian.....	2
1.4. Manfaat Penelitian.....	3
1.5. Batasan Masalah Penelitian	3
1.6. Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 Internet of Things	5
2.2 ESP32	6
2.3 <i>Multitasking</i>	8
2.3.1 Konsep Multitasking.....	9

2.4 RTOS	10
2.4.1 FreeRTOS	10
2.4.2 TridentTD_EasyFreeRTOS32	11
2.5 Sensor	13
2.5.1 DHT11	14
2.5.2 Sensor pH.....	15
2.5.3 Sensor TDS	16
2.5.4 Sensor Ultrasonik.....	17
2.5.5 Sensor DS18B20.....	18
2.5.6 RTC DS1302.....	19
2.6 Aktuator.....	20
2.6.1 Servo	20
2.6.2 Relay	22
2.7 Antares IoT.....	25
2.8 Hidroponik.....	25
BAB III METODOLOGI PENELITIAN	27
3.1 Tahapan Penelitian	27
3.2 Waktu dan Lokasi Penelitian.....	28
3.3 Instrumen Penelitian.....	28
3.4 Pengambilan Data.....	29

3.5 Perancangan Sistem.....	29
3.5.1 Pembuatan Prototipe Hidroponik	30
3.5.2 Perancangan Perangkat Keras.....	33
3.5.3 Penulisan Program	38
3.6 Pengujian Sistem	42
BAB IV HASIL DAN PEMBAHASAN	44
4.1 Hasil Penelitian.....	44
4.1.1 Pengujian Berdasarkan Jenis	44
4.1.2 Pengujian Berdasarkan Jumlah <i>Task</i>	52
4.2 Pembahasan	57
BAB V PENUTUP.....	63
5.1 Kesimpulan.....	63
5.2 Saran	64
DAFTAR PUSTAKA	65
LAMPIRAN.....	68

DAFTAR GAMBAR

Gambar 2. 1 Konsep Internet of Things.....	6
Gambar 2. 2 ESP32.....	8
Gambar 2. 3 Konsep Multitasking	10
Gambar 2. 4 deklarasi library TridentTD_EasyFreeRTOS32.....	11
Gambar 2. 5 deklarasi variabel sub-tugas dan fungsi	11
Gambar 2. 6 Format standar fungsi sub-tugas	12
Gambar 2. 7 memanggil fungsi tugas	12
Gambar 2. 8 Menghentikan tugas yang berjalan.....	12
Gambar 2. 9 deklarasi variabel.....	13
Gambar 2. 10 mengalokasikan memori untuk suatu task	13
Gambar 2. 11 Sensor Suhu dan Kelembaban DHT11.....	14
Gambar 2. 12 Sensor Derajat Keasaman (pH).....	15
Gambar 2. 13 Sensor Total Dissolved Solid (TDS).....	16
Gambar 2. 14 Sensor Ultrasonik HC-SR04	17
Gambar 2. 15 Sensor Suhu Air DS18B20.....	18
Gambar 2. 16 RTC DS1302.....	19
Gambar 2. 17 Servo.....	20
Gambar 2. 18 Pin Out Kabel Motor Servo.....	21
Gambar 2. 19 Relay.....	22
Gambar 2. 20 Bagian-bagian Relay	23
Gambar 2. 21 Antares	25

Gambar 2. 22 Sistem Hidroponik DFT	26
Gambar 3. 1 Diagram Tahapan Penelitian	27
Gambar 3. 2 Prototipe Hidroponik.....	31
Gambar 3. 3 Netpot	32
Gambar 3. 4 Pompa air celup SP35 12 V DC	33
Gambar 3. 5 Rangkaian mikrokontroler, sensor, dan aktuator	34
Gambar 3. 6 Access Key Akun Antares IoT.....	40
Gambar 3. 7 Add Application Antares IoT	41
Gambar 3. 8 Add Device di Antares IoT	41
Gambar 3. 9 Nama Application dan Nama Device di Antares IoT	42
Gambar 4. 1 Diagram perbandingan jumlah task dikerjakan.....	59

DAFTAR TABEL

Tabel 4. 1 Nilai dan jumlah data program tanpa RTOS.....	45
Tabel 4. 2 Total pengerjaan Task pada program tanpa RTOS.....	45
Tabel 4. 3 Nilai dan jumlah data program dengan freeRTOS.....	47
Tabel 4. 4 Total pengerjaan Task pada program dengan freeRTOS.....	48
Tabel 4. 5 Nilai dan jumlah data program dengan TridentTD_EasyFreeRTOS3250	
Tabel 4. 6 Total pengerjaan Task pada program dengan TridentTD_EasyFreeRTOS32.....	51
Tabel 4. 7 Jumlah pengerjaan program 3 task.....	53
Tabel 4. 8 Jumlah pengerjaan program 5 task.....	54
Tabel 4. 9 Jumlah pengerjaan program 7 task.....	55
Tabel 4. 10 Jumlah pengerjaan program 9 task.....	56
Tabel 4. 11 Tabel perbandingan total task dikerjakan berdasarkan jenis program	58
Tabel 4. 12 jumlah data diterima Antares program berdasarkan jumlah task.....	60
Tabel 4. 13 Tabel jumlah pengerjaan task berdasarkan jumlah task.....	61

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penelitian *Business Insider Intelligence* memperkirakan akan ada lebih dari 41 miliar perangkat *Internet of Things* (termasuk sistem tersemat) pada tahun 2027. penelitian ini menunjukkan banyaknya variasi perangkat IoT yang diperkirakan akan terpasang. Mengingat bahwa perangkat IoT merupakan perangkat yang mengintegrasikan antara sensor, prosesor, dan aktuator, menunjukkan bahwa akan terdapat banyak pula sensor dan aktuator, baik dari segi jumlah maupun dari segi variasi yang akan digunakan. Terlebih lagi, setiap sistem IoT terkadang membutuhkan lebih dari satu sensor dan satu aktuator. Hal ini akan berakibat pada semakin banyak dan kompleksnya data yang akan masuk dan akan diolah.

Banyaknya data sensor yang beragam, baik dari jenis hingga cara pengambilan dan pengolahannya, ditambah banyaknya sensor dan aktuator yang digunakan dalam satu perangkat, akan menyebabkan permasalahan pada efisiensi sistem. Efisiensi sistem yang dimaksud meliputi waktu kerja sistem dan proses pekerjaan yang dilakukan di dalam sistem. Apalagi ketika sistem butuh melakukan pekerjaan tanpa harus mengganggu pekerjaan yang lain. Pada keadaan inilah konsep *multitasking* diperlukan.

ESP32 adalah salah satu platform elektronik yang digunakan dalam konsep IoT maupun sistem tersemat. ESP32 ini pun memiliki banyak keunggulan dibanding mikrokontroler lain yang sering digunakan. Namun dalam pengimplementasiannya, tidak dapat lepas dari permasalahan di atas. Konsep

multitasking yang dapat digunakan salah satunya adalah dengan menggunakan *Real Time Operating System* (RTOS).

Oleh peneliti, penerapannya sendiri menggunakan contoh kasus hidroponik terotomasi. Hidroponik adalah sebuah teknik budidaya tanaman yang memanfaatkan air, dan menghilangkan unsur tanah sebagai media tanamnya. Seperti halnya kebutuhan hidup setiap tanaman, banyak aspek yang akan mempengaruhi keberlangsungan hidup tanaman hidroponik. Aspek ini bertambah karena penghilangan unsur tanah dan digantikan dengan menggunakan air sebagai media penyalur nutrisi tanaman sehingga kebutuhan nutrisi tanaman harus benar-benar diperhatikan. Banyaknya aspek yang perlu diperhatikan tersebut, membuat hidroponik terotomasi akan menjadi contoh kasus yang cocok untuk penerapan metode *multitasking*.

1.2. Rumusan Masalah

Rumusan masalah yang akan diuraikan dalam tugas akhir ini adalah:

1. Berapa banyak *task* yang dapat diterima ESP32 dalam satu satuan waktu menggunakan konsep *multitasking*?
2. Apa perbedaan yang terjadi pada sistem hidroponik terotomasi dengan penerapan *multitasking* dan tanpa *multitasking*?

1.3. Tujuan Penelitian

Penelitian bertujuan untuk:

1. Mengetahui jumlah *task* yang dapat diterima ESP32 jika menerapkan konsep *multitasking*.
2. Mengetahui perbedaan yang terjadi pada sistem hidroponik terotomasi jika menerapkan *multitasking* dan tanpa menerapkan *multitasking*.

1.4. Manfaat Penelitian

Manfaat Penelitian Tugas Akhir ini adalah:

1. Bagi Petani Hidroponik, dapat menjadi pengetahuan mengenai pengembangan metode hidroponik.
2. Bagi Peneliti, dapat menjadi pengetahuan mengenai perbandingan dalam penerapan *multitasking* dan *Internet of Things*.
3. Bagi Institusi Pendidikan, dapat digunakan sebagai referensi dalam pengembangan penelitian topik terkait.

1.5. Batasan Masalah Penelitian

Ruang lingkup pembahasan tugas akhir ini dibatasi hanya mencakup hal-hal berikut:

1. Sistem hidroponik yang digunakan adalah *DFT (Deep Flow Technique)*
2. Mikrokontroler yang digunakan adalah ESP32
3. Menggunakan *Real Time Operating System* untuk *multitasking*

1.6. Sistematika Penulisan

Untuk memberikan gambaran singkat mengenai isi tulisan secara keseluruhan, maka akan diuraikan beberapa tahapan dari penulisan secara sistematis, yaitu :

BAB I PENDAHULUAN

Bab ini menguraikan secara umum mengenai hal yang menyangkut latar belakang, perumusan masalah dan batasan masalah, tujuan, manfaat, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi teori-teori terkait hal-hal yang mendasari dan berhubungan dengan penelitian, termasuk di dalamnya mikrokontroler, hidroponik, sensor, aktuator dan metode-metode yang digunakan dalam penelitian.

BAB III METODOLOGI PENELITIAN

Bab ini berisi tentang perencanaan dan proses penerapan algoritma dan metode-metode dalam pengolahan data, mulai dari perancangan prototipe hingga evaluasi kinerja mikrokontroler.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang hasil pengolahan data serta pembahasan yang disertai tabel hasil penelitian.

BAB V PENUTUP

Bab ini berisi tentang kesimpulan yang didapatkan berdasarkan hasil penelitian yang telah dilakukan serta saran-saran untuk pengembangan lebih lanjut.

BAB II

TINJAUAN PUSTAKA

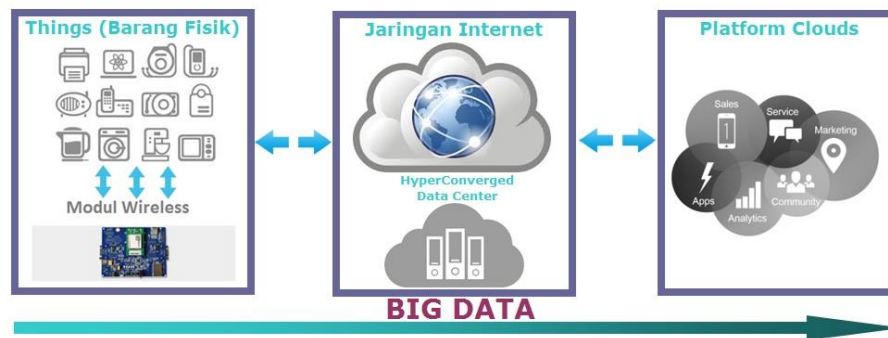
2.1 Internet of Things

Menurut (Metha, 2015) *Internet of Things* atau dikenal juga dengan singkatan IoT, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus yang memungkinkan kita untuk menghubungkan mesin, peralatan, dan benda fisik lainnya dengan sensor jaringan dan aktuator untuk memperoleh data dan mengelola kinerjanya sendiri, sehingga memungkinkan mesin untuk berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperoleh secara independen. (Efendi, 2018)

Menurut *Coordinator and support action for global RFID-related activities and standardisation* menyatakan *internet of things* (IoT) sebagai sebuah infrastruktur koneksi jaringan global, yang mengoneksikan benda fisik dan virtual melalui eksploitasi *data capture* dan teknologi komunikasi. Infrastruktur IoT terdiri dari jaringan yang telah ada dan internet berikut pengembangannya. Hal ini menawarkan identifikasi obyek, identifikasi sensor dan kemampuan koneksi yang menjadi dasar untuk pengembangan layanan dan aplikasi kooperatif yang berdiri secara independen, juga ditandai dengan tingkat otonomi *data capture* yang tinggi, *event transfer*, konektivitas pada jaringan dan juga interoperabilitas. (Setiadi, 2018)

Menurut IEEE (*Institute of Electrical and Electronics Engineers*) *Internet of Things* (IoT) didefinisikan sebagai sebuah jaringan dengan masing-masing benda

yang tertanam dengan sensor yang terhubung ke dalam jaringan internet. (IEEE “Internet of Things” 2014)



Gambar 2. 1 Konsep Internet of Things

Konsep *Internet of Things* mencakup 3 elemen utama yaitu: benda fisik atau nyata yang telah diintegrasikan pada modul sensor, koneksi internet, dan pusat data pada server untuk menyimpan data ataupun informasi dari aplikasi. Penggunaan benda yang terkoneksi ke internet akan menghimpun data yang kemudian terkumpul menjadi ‘*big data*’ untuk kemudian diolah, dianalisis baik oleh instansi pemerintah, perusahaan terkait, maupun instansi lain kemudian di dimanfaatkan bagi kepentingan masing-masing. (Setiadi, 2018) (Efendi, 2018).

2.2 ESP32

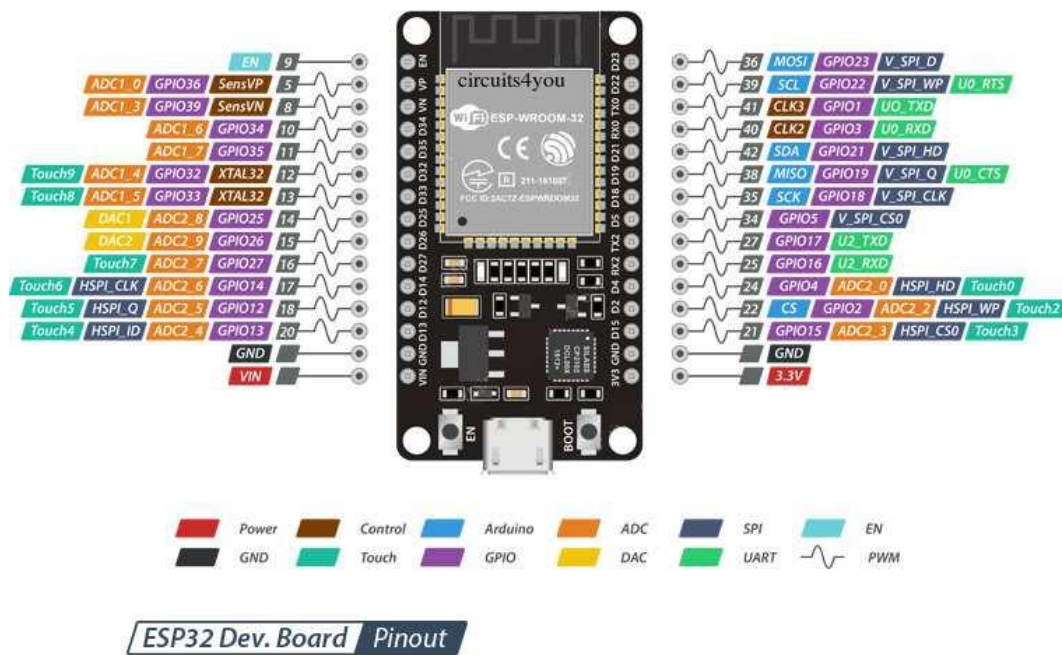
ESP32 merupakan mikrokontroler keluaran espressif yang mana disebutkan dalam websitenya sebagai “*A feature-rich MCU with integrated Wi-Fi and Bluetooth connectivity for a wide-range of application*”. Atau dengan kata lain ESP32 adalah *microcontroller unit* (disingkat: MCU) yang kaya fitur dengan WiFi dan konektivitas Bluetooth terintegrasi untuk beragam aplikasi.

Website Espressif juga menggambarkan kemampuan ESP32, yaitu:

1. *Robust design* : ESP32 mampu berfungsi dengan andal di lingkungan industri, dengan suhu pengoperasian berkisar antara -40°C hingga 125°C . Didukung oleh sirkuit kalibrasi canggih, ESP32 secara dinamis dapat menghapus ketidaksempurnaan sirkuit eksternal dan beradaptasi dengan perubahan kondisi eksternal.
2. *Ultra-Low Power Consumption* : Didesain untuk perangkat *mobile*, perangkat elektronik yang dapat dipakai dan aplikasi IoT, ESP32 mencapai konsumsi daya sangat rendah dengan kombinasi beberapa jenis perangkat lunak berpemilik (*proprietary software*). ESP32 juga mencakup fitur canggih, seperti gating jam berbutir halus (*fine-grained clock gating*), berbagai mode daya dan penskalaan daya dinamis.
3. *High Level of Integration* : ESP32 sangat terintegrasi, dengan sakelar antena built-in, balun RF, *amplifier* daya, *low-noise receive amplifier*, filter, dan modul manajemen daya. ESP32 menambahkan fungsionalitas dan keserbagunaan yang tak ternilai dengan persyaratan *minimal Printed Circuit Board (PCB)*.
4. *Hybrid Wi-Fi and Bluetooth Chip* : ESP32 dapat berfungsi sebagai sistem mandiri yang lengkap atau sebagai perangkat *slave* untuk *MCU host*, mengurangi *overhead* tumpukan komunikasi pada prosesor utama. ESP32 dapat berinteraksi dengan sistem lain untuk menyediakan fungsi Wi-Fi dan Bluetooth melalui antarmuka SPI / SDIO atau I2C / UART.

Gambar 2.2 menunjukkan mikrokontroler ESP32 sebagai penerus dari mikrokontroler ESP8266. Pada mikrokontroler ini sudah tersedia modul WiFi dan

bluetooth sehingga sangat mendukung untuk membuat sistem aplikasi *Internet of Things*. Memiliki 18 ADC (*Analog Digital Converter*), 2 DAC, 16 PWM, 10 sensor sentuh, 2 jalur antarmuka UART, pin antarmuka I2C, I2S, dan SPI. (G, 2019)



Gambar 2. 2 ESP32

2.3 Multitasking

Simultan adalah salah satu faktor yang harus diperhatikan dalam membangun sebuah sistem multi-sensor. Ketika sistem multi-sensor berjalan secara simultan maka akan memberikan hasil yang lebih optimal pada sistem tersebut dengan respons yang lebih cepat. Dengan cepatnya sistem tersebut dalam memproses sensor maka akan mempercepat pemrosesan data oleh pusat pemrosesan. Dalam penerapan multi-sensor sendiri sudah sangat luas, akan tetapi faktor simultan pada sistem tersebut kurang diperhatikan. Salah satu solusi dalam penerapan sistem multi-sensor agar berjalan secara simultan adalah dengan *multitasking* (Zouaoui, 2017).

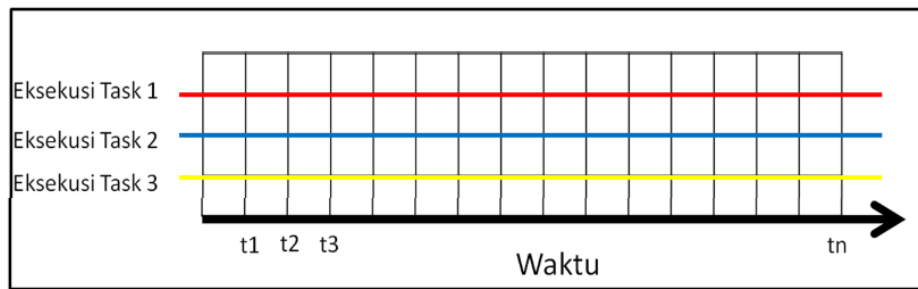
Dibutuhkan sebuah arsitektur yang dapat menampung sistem multi-sensor agar berjalan secara simultan. *Multitasking* adalah salah satu solusi pada masalah tersebut, dengan menjalankan beberapa *task* atau *thread* dalam rentang waktu tertentu. *Multitasking* bisa diartikan mengeksekusi lebih dari satu program secara bersamaan (Putze, 2013).

2.3.1 Konsep Multitasking

Sistem operasi memiliki komponen utama untuk mengatur adanya *multitasking*. *Multitasking* sendiri ialah kemampuan sistem operasi untuk menjalankan banyak *task* atau *thread* dalam rentang waktu tertentu. Setiap *task* yang ada terbentuk dari setiap proses yang harus dijalankan oleh sebuah komputer atau *Embedded system*. (Jatmiko, et al., 2015)

Konsep *multitasking* dari sistem operasi akan memudahkan desain sebuah aplikasi yang kompleks. Terdapat beberapa keuntungan dari *multitasking* sebagai berikut : (Jatmiko, et al., 2015)

- Memungkinkan sebuah aplikasi yang kompleks dibagi menjadi bagian lebih kecil, sederhana, dan mudah diatur.
- Bagian-bagian kecil dari sebuah aplikasi dapat dengan mudah diuji coba, ditelusuri, dan didaur ulang untuk kebutuhan lain.
- Detail alur dari aplikasi serta pengaturan waktu eksekusi yang kompleks dapat dihilangkan dari kode aplikasi. Hal tersebut sudah menjadi tanggung jawab dari sistem operasi itu sendiri.



Gambar 2. 3 Konsep Multitasking

Gambar 2.3 menunjukkan bagaimana konsep *multitasking* bekerja. Tiga *task* akan berjalan dalam rentang waktu yang sama dan waktu eksekusinya sama.

2.4 RTOS

Real Time Operating System (RTOS) adalah sistem operasi yang dioptimalkan untuk digunakan dalam aplikasi tertanam *real time*. RTOS memperkenalkan konsep *real time* yang berarti sistem akan mengerjakan seluruh *task* yang ada sesuai dengan waktu yang ditentukan sehingga tidak ada tugas yang dijalankan melebihi waktu *deadline* yang ditentukan. (Jatmiko, et al., 2015)

Real Time Operating System (RTOS) mengeksekusi program-program dalam sebuah pola yang teratur. RTOS dijalankan oleh program otomatis yang disebut dengan *kernel*. Pada saat sistem operasi dinyatakan, maka *kernel* akan menyala terlebih dahulu kemudian *kernel* tersebut akan menyalakan *Real Time Operating System* (RTOS). (Hifdzullisan, Sumaryo, & Rizal, 2018)

2.4.1 FreeRTOS

FreeRTOS adalah salah satu *Real Time Operating System* (RTOS) yang dapat dijalankan untuk *embedded system*. Sistem operasi ini dapat dipasang pada mikrokontroler. Sistem tertanam biasanya menggunakan satu atau lebih mikrokontroler sebagai *integrated circuit* (IC) untuk menjalankan fungsi tertentu.

Mikrokontroler terdiri dari prosessor sekaligus memori baik ROM, *Flash*, maupun RAM, serta beberapa komponen penunjang seperti *transistor*, *clock*, *resistor*, dan lain-lain. FreeRTOS menyediakan beberapa fitur utama suatu *Real Time Operating System* (RTOS). Fitur tersebut di antaranya *real time scheduling functionality*, *inter-task communication*, *timing*, dan *synchronization primitives*. (Jatmiko, et al., 2015)

2.4.2 TridentTD_EasyFreeRTOS32

TridentTD_EasyFreeRTOS32 merupakan *library* untuk penulisan *multitasking* pada *board* ESP32. *Library* ini dibuat oleh TridentTD. Untuk menjalankan *library* ini, masukkan kode berikut.

```
#include <TridentTD_EasyFreeRTOS32.h>
```

Gambar 2. 4 deklarasi library TridentTD_EasyFreeRTOS32

Kemudian untuk mendeklarasikan sub-tugas dan fungsi yang dipicu oleh tugas dengan kode berikut.

```
TridentOS task1, task2, task3;  
void task1_func(void*), task2_func(void*), task3_func(void*);
```

Gambar 2. 5 deklarasi variabel sub-tugas dan fungsi

Format standar, fungsi sub-tugas yang akan diaktifkan, adalah sebagai berikut.


```

void task1_func(void*) { // ฟังก์ชันที่ task1 เรียกทำงาน - void ใช้ตัวพิมพ์เล็ก
//----พื้นที่สำหรับประกาศตัวแปรที่ใช้ภายใน task นี้เท่านั้น----

//-----
VOID SETUP() { // VOID SETUP() ใน task ให้ใช้ตัวพิมพ์ใหญ่
}
VOID LOOP() { // VOID LOOP() ใน task ให้ใช้ตัวพิมพ์ใหญ่
}
}

```

Gambar 2. 6 Format standar fungsi sub-tugas

Untuk melakukan penundaan pada fungsi tugas, perintah yang digunakan adalah DELAY(.) dengan huruf kapital. Untuk memanggil fungsi tugas, misal task1, perintah yang digunakan adalah.

```
task1.start(task1_func);
```

Gambar 2. 7 memanggil fungsi tugas

Untuk menghentikan tugas yang sedang berjalan, perintah yang digunakan adalah.

```
task1.stop();
```

Gambar 2. 8 Menghentikan tugas yang berjalan

Untuk pertukaran data antar *task*, deklarasikan variabel di dalam area deklarasi variabel dengan standar penamaan variabel yang dimulai dengan huruf x kecil.

```
//----- พื้นที่สำหรับประกาศตัวแปรชั่วคราว task -----  
float xDHT_temp, xDHT_humid; // มีตัว x เล็กหน้าหน้า เพื่อง่ายต่อการสังเกต  
//-----
```

Gambar 2. 9 deklarasi variabel

Terkadang untuk pengoperasian suatu tugas, agar tugas dapat bekerja secara efektif dari waktu ke waktu, maka perlu untuk menyesuaikan ruang memori. Cara mengatur ruang memori untuk suatu *task* dilakukan dengan menggunakan perintah berikut.

```
task1.start(task1_func, NULL, 4096); // ขอใช้พื้นที่ขนาด 4096bytes
```

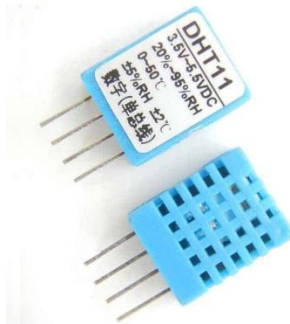
Gambar 2. 10 mengalokasikan memori untuk suatu task

2.5 Sensor

Sensor adalah suatu peralatan yang berfungsi untuk mendeteksi gejala-gejala atau sinyal-sinyal yang berasal dari perubahan suatu energi seperti energi listrik, energi fisika, energi kimia, energi biologi, energi mekanik dan sebagainya (D Sharon, 1982).

Transduser adalah sebuah alat yang bila digerakkan oleh suatu energi di dalam sebuah sistem transmisi, akan menyalurkan energi tersebut dalam bentuk yang sama atau dalam bentuk yang berlainan ke sistem transmisi berikutnya. Transmisi energi ini bisa berupa listrik, mekanik, kimia, optik (radiasi) atau *thermal* (panas). (William D.C, 1993).

2.5.1 DHT11



Gambar 2. 11 Sensor Suhu dan Kelembaban DHT11

DHT11 merupakan sensor temperatur dan kelembaban dengan *output* sinyal digital yang dikalibrasi. Dengan menggunakan teknik akuisisi sinyal digital eksklusif dan teknologi penginderaan suhu & kelembaban, memastikan keandalan tinggi dan stabilitas jangka panjang yang sangat baik. Sumber daya listrik yang dibutuhkan oleh DHT11 adalah 3 V – 5,5 V DC. Format *single-bus data* digunakan untuk komunikasi antara MCU dan sensor DHT11. Satu proses komunikasi sekitar 4 ms. Data terdiri dari bagian desimal dan integral. Transmisi data yang lengkap adalah 40 *bit*, dan sensor mengirimkan *bit* data yang lebih tinggi terlebih dahulu. Format data: data RH integral 8 *bit* + data RH desimal 8 *bit* + data T integral 8 *bit* + data T desimal 8 *bit* + jumlah cek 8 *bit*. Jika transmisi data benar, *check-sum* harus menjadi 8 *bit* terakhir dari "data RH 8 *bit* integral + data RH desimal 8 *bit* + data T integral 8 *bit* + data T desimal 8 *bit*".

Sensor DHT11 adalah salah satu sensor yang dapat mengukur dua parameter lingkungan sekaligus, yakni suhu dan kelembaban udara (*humidity*). Dalam sensor ini terdapat sebuah *thermistor* tipe NTC (*Negative Temperature Coefficient*) untuk mengukur suhu, sebuah sensor kelembaban tipe resistif dan sebuah mikrokontroler

8 bit yang mengelola kedua sensor tersebut dan mengirim hasilnya ke pin *output* dengan format *single-wire bi-directional* (kabel tunggal dua arah). (Ajie, 2016).

2.5.2 Sensor pH



Gambar 2. 12 Sensor Derajat Keasaman (pH)

Sensor derajat keasaman (pH) yang digunakan terdiri atas *Sensor Module* dan *PH Electrode Probe* yang memiliki spesifikasi:

1. *Heating voltage* : 5 0.2 (AC DC)
2. *Working current*: 5 – 10 mA
3. *Detectable concentration range*: pH 0 – 14
4. *Detection temperature range*: 0 – 80
5. *Response time*: 5 s
6. *Settling time*: 60 s
7. *Component power*: 0,5 W
8. *Working temperature*: -10 ~ 50 (nominal temperature 20)
9. *Humidity*: 95% RH (nominal humidity 65% RH)
10. *Module size*: 43 mm x 32 mm x 20 mm
11. *Output*: analog voltage signal output

2.5.3 Sensor TDS



Gambar 2. 13 Sensor Total Dissolved Solid (TDS)

Sensor TDS dapat mengambil data konduktivitas, *total dissolved solid* (TDS), dan kadar garam (salinitas). Sensor TDS bekerja pada tegangan 5 V DC. Memiliki 3 pin yaitu pin daya, *ground* dan data. Memiliki koefisien linieritas data konduktivitas sebesar 0,9639 dan koefisien linieritas data TDS sebesar 0,983. Memiliki sensitivitas pada bahan yang bersifat konduktif, dengan kedalaman cairan pada saat pengukuran sebesar 5,5 cm dari ujung sensor. Rumus persamaan umum konversi data konduktivitas $y = 0,2142x + 494,93$ di mana nilai x adalah nilai ADC, dan nilai y adalah konduktivitas. Rumus persamaan umum konversi data TDS $y = 0,3147x + 281,08$ di mana x adalah nilai ADC dan y adalah nilai TDS. Namun pada implementasinya, sering kali nilai yang dihasilkan oleh sensor tidak akurat, sehingga harus dilakukan kalibrasi ulang.

2.5.4 Sensor Ultrasonik



Gambar 2. 14 Sensor Ultrasonik HC-SR04

Sensor ultrasonik mampu mengkonversi gelombang bunyi ke dalam beberapa satuan seperti jarak, ketinggian dan kecepatan. Teknik pengukuran jarak/panjang ini menggunakan gelombang ultrasonik di udara termasuk metode *echo* pulsa, pancaran pulsa dikirim ke media transmisi dan dipantulkan oleh sebuah objek pada jarak tertentu. Waktu yang diambil dari pemancar ke penerima sebanding dengan jarak objek. Sensor HC-SR04 adalah sensor pengukur jarak berbasis gelombang ultrasonik. Keunggulan sensor ini adalah jangkauan deteksi sekitar 2 cm sampai kisaran 400-500 cm dengan resolusi 1 cm. Sensor HC-SR04 adalah versi *low cost* dari sensor *ultrasonic* PING buatan parallax. Perbedaannya terletak pada pin yang digunakan. HC-SR04 menggunakan 4 pin sedangkan PING buatan Parallax menggunakan 3 pin.

Sensor ultrasonik tipe HC-SR04 merupakan perangkat yang digunakan untuk mengukur jarak dari suatu objek. Kisaran jarak yang dapat diukur sekitar 2 cm - 450 cm. Perangkat ini menggunakan dua pin digital untuk mengkomunikasikan jarak yang terbaca. Prinsip kerja sensor ultrasonik ini bekerja dengan mengirimkan pulsa ultrasonik sekitar 40 KHz, kemudian dapat memantulkan pulsa *echo* kembali,

dan menghitung waktu yang diambil dalam mikrodetik. Kita dapat memicu pulsa secepat 20 kali perdetik dan itu bisa tentukan objek hingga 3 meter.

Sensor ultrasonik tipe HC-SR04 memiliki empat pin, yaitu GND (*power ground*), TRIG (*trigger input pin*), ECHO (*receiver output pin*), dan VCC (5 V *power supply*).

2.5.5 Sensor DS18B20



Gambar 2. 15 Sensor Suhu Air DS18B20

DS18B20 adalah sensor temperatur digital yang dapat dihubungkan dengan mikrokontroler lewat antarmuka *1-Wire*. Sensor ini dikemas secara khusus sehingga kedap air, cocok digunakan sebagai sensor di luar ruangan / pada lingkungan dengan tingkat kelembaban tinggi. Dengan kabel sepanjang 1 meter, penempatan komponen sensor elektronika ini dapat diatur secara fleksibel. Gambar 2.8 menunjukkan bentuk fisik dari sensor suhu DS18B20.

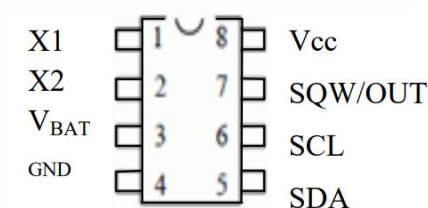
Protokol *1-Wire* hanya membutuhkan 1 kabel koneksi (selain *ground*) untuk mentransmisikan data. Berikut ini adalah ringkasan fitur IC DS18B20:

1. Antarmuka *1-Wire* yang hanya membutuhkan 1 pin I/O untuk komunikasi data.

2. Tidak membutuhkan komponen eksternal tambahan selain 1 buah *pull-up resistor*, artinya hanya menambahkan sebuah sensor yang tersambung dari pin data ke pin vcc sensor suhu DS18B20.
3. Dapat mengukur suhu antara -55°C hingga 125°C dengan akurasi $0,5^{\circ}\text{C}$ pada -10°C s.d. $+85^{\circ}\text{C}$.
4. Kecepatan pendeteksian suhu pada resolusi maksimum kurang dari 750ms.

2.5.6 RTC DS1302

RTC DS1302 merupakan buatan Dallas-Maxim Semiconductor. RTC DS1302 menyediakan informasi detik, menit, jam, tanggal, bulan dan tahun (Pracoyo, 2008). RTC DS1302 memiliki sistem *nonvolatile* (permanen) SRAM 56 bytes di mana alamat dan data dikirimkan secara serial perbit dengan menggunakan sistem *Inter Integrated Circuit* (I2C) yang dikembangkan oleh Philips 37 Semikonduktor (Susilo, 2010). Real Time Clock (RTC) DS1302, diperlihatkan pada Gambar 2.9.



Gambar 2. 16 RTC DS1302

Inter-Integrated Circuit (I2C) merupakan komunikasi serial antar rangkaian terintegrasi. Komunikasi serial I2C selalu diawali dengan kondisi *start* dan diakhiri *stop*. Kondisi *start* adalah ketika terjadi perubahan kondisi dari *high* ke *low* pada

SDA ketika SCL pada kondisi *high*, sedangkan kondisi *stop* adalah ketika terjadi perubahan kondisi dari *low* ke *high* pada SDA ketika SCL pada kondisi *high* (Budiharto dan Gamayel, 2007).

2.6 Aktuator

Actuator (penggerak) dalam pengertian listrik adalah setiap alat yang mengubah sinyal listrik menjadi gerakan mekanis. Biasa digunakan sebagai proses lanjutan dari keluaran suatu proses olah data yang dihasilkan oleh suatu sensor atau *controller*. Pada instrumentasi, aktuator sebagai *output* terakhir dan penerus perintah dari *controller* untuk melakukan tindakan eksekusi / koreksi.

2.6.1 Servo

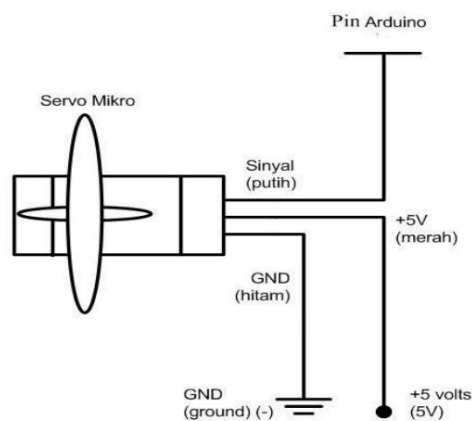


Gambar 2. 17 Servo

Motor servo adalah sebuah motor dengan sistem umpan balik tertutup di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor servo merupakan salah satu jenis motor DC. Berbeda dengan motor *stepper*, motor servo beroperasi secara *close loop*. Poros motor dihubungkan dengan rangkaian kendali, sehingga jika putaran poros belum sampai pada posisi yang diperintahkan maka rangkaian kendali akan terus mengoreksi

posisi hingga mencapai posisi yang diperintahkan. Motor servo banyak digunakan pada peranti R/C (*Remote Control*) seperti mobil, pesawat, helikopter, dan kapal, penggerak pada kamera serta sebagai aktuator robot. Pada robot *boat* pengintai, motor servo digunakan sebagai pengendali kamera pengintai. (Fahmi, 2011).

Motor servo merupakan motor yang berputar lambat, di mana biasanya ditunjukkan oleh *rate* putarannya yang lambat, namun demikian memiliki torsi yang kuat karena pada internal *gear*-nya. Motor servo memiliki 3 kabel yaitu putih sebagai I/O pin, merah sebagai vcc dan hitam sebagai *ground*. Dengan demikian, motor servo dapat dikontrol melalui kabel I/O yang berwarna putih. Pada gambar 2.11 dibawah ini merupakan pin-pin dan pengkabelan dari motor servo yang dihubungkan pada rangkaian pengontrol. (Suyadi, 2014).



Gambar 2. 18 Pin Out Kabel Motor Servo

Di dalam sebuah motor servo terdapat beberapa karakteristik, yaitu :

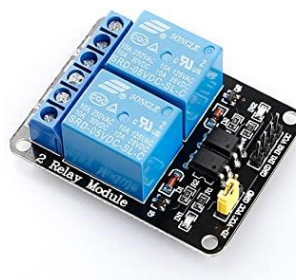
- 3 jalur : *power*, *ground* dan *control*.
- Sinyal *control* mengendalikan posisi.

- Operasional dari motor servo dikendalikan oleh pulsa sebesar 20 ms, dimana lebar pulsa antara 500 μ s dan 2400 μ s menyatakan akhir dari *range* sudut maksimum.
- Konstruksi didalamnya meliputi *internal gear*, potensiometer dan *feedback control*.

Di dalam motor servo terdapat potensiometer yang digunakan sebagai sensor posisi. Potensiometer tersebut dihubungkan dengan *output shaft* untuk mengetahui sudut posisi dari *output gear* pada motor servo. Ketika motor DC (*Direct Current*) berputar, maka *output shaft* juga berputar dan sekaligus memutar potensiometer.

Rangkaian *control* kemudian dapat membaca kondisi potensiometer tersebut untuk mengetahui posisi *actual shaft*. Jika posisinya sesuai dengan yang diinginkan, maka motor dc akan berhenti. Sudut operasi motor servo (*operating angle*) bervariasi tergantung jenis motor servo.

2.6.2 Relay



Gambar 2. 19 Relay

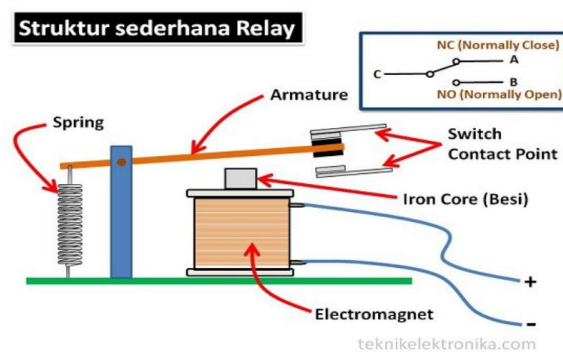
Relay merupakan komponen *output* yang paling sering digunakan pada beberapa peralatan elektronika dan di berbagai bidang lainnya. Relay berfungsi untuk menghubungkan atau memutuskan aliran arus listrik yang dikontrol dengan

memberikan tegangan dan arus tertentu pada koilnya (Setiawan, 21:2011). Ada 2 macam relay berdasarkan tegangan untuk menggerakkan koilnya, yaitu AC dan DC.

Pada dasarnya, Relay terdiri dari 4 komponen dasar yaitu :

1. Elektromagnet (*Coil*)
2. *Armature*
3. *Switch Contact Point* (Saklar)
4. *Spring*

Gambar 2.13 merupakan gambar dari bagian-bagian Relay:



Gambar 2. 20 Bagian-bagian Relay

Kontak poin (*Contact Point*) relay terdiri dari 2 jenis yaitu :

1. *Normally Close* (NC) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *CLOSE* (tertutup).
2. *Normally Open* (NO) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *OPEN* (terbuka).

Berdasarkan gambar di atas, sebuah besi (*Iron Core*) yang dililit oleh sebuah kumparan *coil* yang berfungsi untuk mengendalikan besi tersebut. Apabila kumparan *coil* diberikan arus listrik, maka akan timbul gaya elektromagnet yang kemudian menarik *armature* untuk berpindah dari posisi sebelumnya (NC) ke posisi baru (NO) sehingga menjadi saklar yang dapat menghantarkan arus listrik di posisi barunya (NO). Posisi di mana *armature* tersebut berada sebelumnya (NC) akan menjadi open atau tidak terhubung. Pada saat tidak dialiri arus listrik, *armature* akan kembali lagi ke posisi awal (NC). *Coil* yang digunakan oleh Relay untuk menarik *Contact Poin* ke posisi *close* pada umumnya hanya membutuhkan arus listrik yang relatif kecil.

Beberapa fungsi relay yang telah umum diaplikasikan kedalam peralatan elektronika diantaranya adalah :

1. Relay digunakan untuk menjalankan fungsi logika (*Logic Function*)
2. Relay digunakan untuk memberikan fungsi penundaan waktu (*Time Delay Function*)
3. Relay digunakan untuk mengendalikan sirkuit tegangan tinggi dengan bantuan dari sinyal tegangan rendah
4. Ada juga relay yang berfungsi untuk melindungi motor ataupun komponen lainnya dari kelebihan tegangan ataupun hubung singkat (*Short*)

2.7 Antares IoT



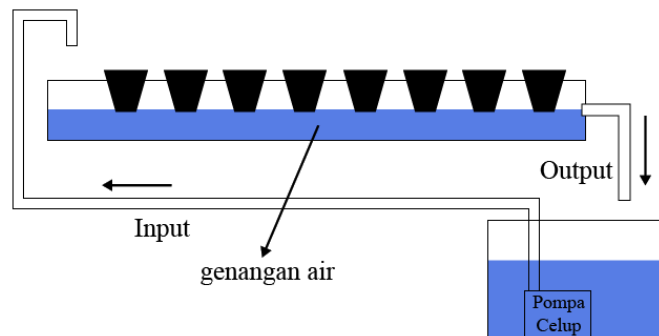
Gambar 2. 21 Antares

Antares adalah produk dan layanan *Internet of Things* (IoT) di bawah naungan PT Telekomunikasi Indonesia. Antares memiliki 4 pilar utama, yaitu IoT platform, IoT Connectivity, IoT Solution, dan Devices. Antares memberikan layanan IoT Platform untuk pengembangan solusi IoT baik perangkat *device* maupun aplikasi berbasis *standard OneM2M*.

Antares menyediakan protokol komunikasi berbasis HTTP, MQTT, Websoket, dan CoAP dengan format data JSON dan XML. Antares menyiapkan *library* dan panduan pengembangan solusi IoT untuk *microcontroller* terkemuka seperti Arduino, ESP, STMico dan lain-lain dengan sistem komunikasi GSM/GPRS, NBIoT, WiFi, LAN, dan LoRaWAN.

2.8 Hidroponik

Sistem Hidroponik DFT adalah metode budidaya tanaman dengan meletakkan akar tanaman dalam air. Prinsip kerja dari sistem hidroponik DFT yaitu mensirkulasikan larutan nutrisi tanaman secara terus menerus selama 24 jam dengan menggunakan motor untuk memompa larutan nutrisi. Teknik hidroponik ini dikategorikan sistem hidroponik tertutup. Umumnya penerapan teknik ini digunakan pada budidaya tanaman daun dan sayuran buah. (Sulistiyo, Erwanto, & Rosanti, 2019)



Gambar 2. 22 Sistem Hidroponik DFT

Teknik DFT dengan menggunakan sistem pipa, dimana aliran nutrisi dengan kedalaman 2 sampai 3 cm dialirkan pada pipa PVC berdiameter 10 cm dan pada pipa tersebut diletakkan tanaman dalam netpot sehingga tanaman akan menerima nutrisi yang mengalir. Pada netpot terdapat material seperti arang sekam atau *rockwool* sebagai tumpukan akar dan bagian bawah dari material tersebut menyentuh larutan nutrisi yang mengalir. Pipa PVC dapat dirangkai dalam satu bidang atau zigzag, tergantung pada jenis tanaman yang dibudidayakan. (Sulistiyo, Erwanto, & Rosanti, 2019)