

SKRIPSI

**PENERAPAN MULTITASKING PADA EDGE COMPUTING
UNTUK MONITORING TANAMAN HIDROPONIK FARM**

Disusun dan diajukan oleh:

DEVY NOVIANI BADJARAD

D121 171 014



PROGRAM STUDI SARJANA TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

GOWA

2023



LEMBAR PENGESAHAN SKRIPSI

PENERAPAN MULTITASKING PADA EDGE COMPUTING UNTUK MONITORING TANAMAN HIDROPONIK FARM

Disusun dan diajukan oleh

Devy Noviani Badjarad
D121 171 014

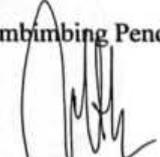
Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka
Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika
Fakultas Teknik Universitas Hasanuddin
Pada tanggal 13 Desember 2023
dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,


Adnan, S.T., M.T., Ph.D.
NIP 197404262005011002


Dr. Eng. Zulkifli Tahir, S.T., M.Sc.
NIP 198404032010121004

Ketua Program Studi,


Prof. Dr. H. Indrabayu S.T., M.T., M. Bus.Sys., IPM, ASEAN. Eng.
NIP 197507162002121004



PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini ;
Nama : Devy Noviani Badjarad
NIM : D121171014
Program Studi : Teknik Informatika
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

PENERAPAN MULTITASKING PADA EDGE COMPUTING UNTUK MONITORING TANAMAN HIDROPONIK FARM

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

Semua informasi yang ditulis dalam skripsi yang berasal dari penulis lain telah diberi penghargaan, yakni dengan mengutip sumber dan tahun penerbitannya. Oleh karena itu semua tulisan dalam skripsi ini sepenuhnya menjadi tanggung jawab penulis. Apabila ada pihak manapun yang merasa ada kesamaan judul dan atau hasil temuan dalam skripsi ini, maka penulis siap untuk diklarifikasi dan mempertanggungjawabkan segala resiko.

Segala data dan informasi yang diperoleh selama proses pembuatan skripsi, yang akan dipublikasi oleh Penulis di masa depan harus mendapat persetujuan dari Dosen Pembimbing.

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

Gowa, 13 Desember 2023

Yang Menyatakan

38960AKX817734992

Devy Noviani Badjarad



ABSTRAK

DEVY NOVIANI BADJARAD. *Penerapan Multitasking pada Edge Computing untuk Monitoring Tanaman Hidroponik Farm* (dibimbing oleh Bapak Adnan, S.T., M.T., Ph.D., dan Bapak Dr. Eng. Zulkifli Tahir, S.T., M.Sc.,)

Banyaknya *node* sensor yang akan mengirimkan data dalam *monitoring* tanaman pada kebun hidroponik akan meningkatkan beban kerja pada *gateway*, karena selain menerima data *gateway* juga melakukan tugas komputasi. *Gateway* dengan pemrograman konvensional akan berjalan secara sekuensial, di mana *task* akan diproses setelah *task* sebelumnya telah selesai diproses, sehingga dapat menyebabkan data yang dikirim oleh *node* sensor tidak sempat diterima atau hilang. Oleh karena itu, untuk mengurangi terjadinya data yang hilang dan meningkatkan efisiensi sistem, diperlukan penerapan sistem *multitasking* yang dapat menjalankan beberapa *task* secara bersamaan tanpa mengganggu pekerjaan *task* yang lain. Pada penelitian ini, *multitasking* diterapkan pada sebuah mikrokontroler menggunakan *freeRTOS* yang merupakan *Real Time Operation System*. Sistem akan dibagi menjadi tiga *task*, dimana *task* pertama bertugas untuk menerima data dari *node* sensor, *task* kedua bertugas untuk memisahkan data dan mengirim data ke *Thingspeak*, serta *task* ketiga bertugas untuk melakukan prediksi data menggunakan algoritma regresi linier. Untuk *node* sensor, akan digunakan *Arduino Uno* sebagai mikrokontroler yang akan dihubungkan dengan modul LoRa. *Node* sensor 1 akan dihubungkan dengan sensor DHT11 dan sensor HC-SR04, *node* sensor 2 dan *node* sensor 3 akan dihubungkan dengan sensor pH dan sensor TDS. Dari hasil pengujian menggunakan 1 *node* sensor, 2 *node* sensor dan 3 *node* sensor, *gateway* tanpa *multitasking* berhasil menerima sebesar 85.78% data dari total pengujian yang dilakukan, namun dengan penerapan *multitasking* terjadi peningkatan yang signifikan, di mana *gateway* menerima data sebesar 98.49% dari total pengujian yang dilakukan. Dengan peningkatan ini, tingkat kehilangan data pada *gateway* juga mengalami penurunan dari 14.22% menjadi 1.51%.

Kata Kunci: *multitasking*, *node* sensor, *gateway*, *freeRTOS*



ABSTRACT

DEVY NOVIANI BADJARAD. *Implementation of Multitasking in Edge Computing for Hydroponic Plant Monitoring Farm.* (supervised by Mr. Adnan, S.T., M.T., Ph.D., and Mr. Dr. Eng. Zulkifli Tahir, S.T., M.Sc.,)

The growing quantity of sensor nodes engaged in transmitting data for plant monitoring in hydroponic farms will impose a heavier burden on the gateway. This is due to the fact that the gateway is responsible not only for receiving data but also for executing computing operations. In the context of conventional programming, a gateway operates in a sequential manner, wherein actions are executed one after another, potentially leading to the non-receipt or loss of data transmitted by the sensor nodes. Hence, in order to mitigate data loss and enhance system efficacy, it is imperative to deploy a multitasking system that enables concurrent execution of numerous tasks without mutual disruption. This study implements multitasking on a microcontroller by utilizing FreeRTOS as real-time operating system. The system has three distinct duties. The first task is responsible for the reception of data from the sensor nodes. The second task is responsible for splitting data and sending data to Thingspeak. Lastly, the third task is dedicated to data prediction through the utilization of a linear regression algorithm. The sensor nodes employ an Arduino Uno microcontroller, which is interconnected with LoRa modules. Sensor node 1 is equipped with the DHT11 and HC-SR04 sensors, while sensor node 2 is equipped with the pH sensor, and sensor node 3 is equipped with the TDS sensor. Based on the test using 1 node sensor, 2 node sensors, and 3 node sensors, it was observed that the gateway operating without multitasking effectively received 85.78% of the data from the total tests conducted. Nevertheless, the introduction of multitasking resulted in a notable surge, as the gateway successfully received 98.49% of the data from the total tests conducted. The observed increase in data loss on the gateway led to a corresponding fall from 14.22% to 1.51%.

Keywords: multitasking, node sensor, gateway, freeRTOS



DAFTAR ISI

LEMBAR PENGESAHAN SKRIPSI	i
PERNYATAAN KEASLIAN.....	ii
ABSTRAK	iii
ABSTRACT.....	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	viii
DAFTAR SINGKATAN DAN ARTI SIMBOL	ix
DAFTAR LAMPIRAN.....	x
KATA PENGANTAR	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan.....	3
1.4 Manfaat Penelitian.....	3
1.5 Batasan Masalah.....	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 <i>Internet of Things</i>	5
2.2 <i>Edge Computing</i>	6
2.3 LPWAN (<i>Low Power Wide Area Network</i>)	8
2.3.1 NB-IoT	10
2.3.2 Sigfox.....	11
2.3.3 LoRa.....	12
2.4 <i>Real Time Operation System (RTOS)</i>	13
2.5 freeRTOS.....	14
2.6 <i>Multitasking</i>	14
2.7 Sensor	15
2.7.1 Sensor DHT11.....	15
2.7.2 Sensor HC-SR04	16
2.7.3 Sensor pH.....	17
2.7.4 Sensor TDS	18
<i>Thingspeak</i>	18
Algoritma Regresi Linear	19



BAB III METODOLOGI PENELITIAN.....	20
3.1 Tahapan Penelitian	20
3.2 Waktu dan Lokasi Penelitian.....	21
3.3 Alat dan Bahan Penelitian	21
3.4 Tahap Persiapan	21
3.5 Tahap Perancangan Alat.....	22
3.5.1 Tahap Perancangan <i>Hardware</i> (Alat)	22
3.5.2 Tahap Perwujudan Alat.....	24
3.6 Tahap Pembuatan <i>Software</i>	27
3.6.1 Perancangan <i>Software Node Sensor</i>	27
3.6.2 Perancangan <i>Software Gateway</i>	28
1. Mekanisme <i>FreeRTOS Scheduler</i>	29
2. Pisah Data	30
3. Penerapan Algoritma Regresi Linier	31
4. Penerapan Model Antrian M/M/1 dalam Analisis Kinerja <i>Gateway</i> ..	33
BAB IV HASIL DAN PEMBAHASAN	35
4.1 Pengambilan Data.....	35
4.2 Hasil dan Pembahasan Pengujian pada Node Sensor.....	36
4.3 Hasil dan Pembahasan Pengujian pada Gateway	37
4.3.1 Hasil dan Pembahasan Pengujian Sistem Tanpa <i>Multitasking</i>	39
1. Hasil Pengujian dengan 1 <i>Node Sensor</i>	39
2. Hasil Pengujian dengan 2 <i>Node Sensor</i>	40
3. Hasil Pengujian dengan 3 <i>Node Sensor</i>	42
4.3.2 Hasil dan Pembahasan Pengujian Sistem dengan <i>Multitasking</i>	44
1. Hasil Pengujian dengan 1 <i>Node Sensor</i>	45
2. Hasil Pengujian dengan 2 <i>Node Sensor</i>	46
3. Pengujian dengan 3 <i>Node Sensor</i>	48
4. Hasil dan Pembahasan Pengujian pada <i>Task2</i>	55
5. Hasil dan Pembahasan Pengujian pada <i>Task 3</i>	61
BAB V KESIMPULAN DAN SARAN.....	63
5.1 KESIMPULAN	63
5.2 SARAN	64
DAFTAR PUSTAKA	64
LAMPIRAN.....	68



DAFTAR GAMBAR

Gambar 2. 1 Arsitektur sigfox.....	11
Gambar 2. 2 Arsitektur LoRa.....	12
Gambar 2. 3 Konsep <i>multitasking</i>	15
Gambar 2. 4 Sensor DHT11.....	15
Gambar 2. 5 Sensor Ultrasonik HC-SR04.....	16
Gambar 2. 6 Sensor derajat keasaman (pH).....	17
Gambar 2. 7 Sensor <i>Total Dissolved Solid</i> (TDS).....	18
Gambar 2. 8 <i>Thingspeak</i>	18
Gambar 3. 1 Diagram tahap penelitian.....	20
Gambar 3. 2 Diagram blok sistem.....	22
Gambar 3. 3 Skema desain <i>node</i> sensor 1.....	22
Gambar 3. 4 Skema desain <i>node</i> sensor.....	23
Gambar 3. 5 Skema desain <i>gateway</i>	24
Gambar 3. 6 Perwujudan <i>node</i> sensor 1.....	25
Gambar 3. 7 Perwujudan <i>node</i> sensor 2.....	25
Gambar 3. 8 Perwujudan <i>gateway</i>	26
Gambar 3. 9 <i>Flowchart</i> <i>node</i> sensor.....	28
Gambar 3. 10 <i>Flowchart</i> penerapan <i>multitasking</i> pada <i>edge computing</i>	28
Gambar 3. 11 Mekanisme <i>scheduler</i>	30
Gambar 3. 12 <i>Flowchart</i> pisah data.....	31
Gambar 3. 13 <i>Flowchart</i> penerapan algoritma regresi linear.....	32
Gambar 4. 1 Pemasangan <i>node</i> sensor 2 dan 3 pada penampungan air.....	35
Gambar 4. 2 Pemasangan <i>node</i> sensor 1 pada tanaman <i>hidroponik</i>	35
Gambar 4. 3 Pengiriman data dari <i>node</i> sensor 1 ke <i>gateway</i>	36
Gambar 4. 4 Pengiriman data <i>node</i> sensor 2 ke <i>gateway</i>	36
Gambar 4. 5 Hasil penerimaan data <i>node</i> sensor 1 pada <i>gateway</i>	37
Gambar 4. 6 Hasil penerimaan data <i>node</i> sensor 2 pada <i>gateway</i>	38
Gambar 4. 7 Hasil penerimaan data <i>node</i> sensor 3 pada <i>gateway</i>	38
Gambar 4. 8 Grafik perbandingan data <i>loss</i> dan data diterima sistem tanpa <i>multitasking</i> dan sistem <i>multitasking</i>	53
Gambar 4. 9 Grafik total perbandingan data diterima dan data <i>loss</i>	54
Gambar 4. 10 Grafik perbandingan nilai intensitas lalu lintas.....	55
Gambar 4. 11 Hasil Pisah data <i>node</i> sensor 1.....	55
Gambar 4. 12 Hasil pisah data <i>node</i> sensor 2.....	56
Gambar 4. 13 Hasil pisah data <i>node</i> sensor 3.....	56
Gambar 4. 14 Grafik <i>field</i> 1 <i>thingspeak</i>	57
Gambar 4. 15 Grafik <i>field</i> 2 <i>thingspeak</i>	58
l. 16 Grafik <i>field</i> 3 <i>thingspeak</i>	59
l. 17 Grafik <i>field</i> 4 <i>thingspeak</i>	59
l. 18 Grafik <i>field</i> 5 <i>thingspeak</i>	60
l. 19 <i>Plot</i> regresi linier.....	61
l. 20 Hasil implementasi algoritma regresi linier.....	62



DAFTAR TABEL

Tabel IV. 1 Hasil perbandingan data terkirim dan diterima pada sistem tanpa <i>multitasking</i>	40
Tabel IV. 2 Hasil pengamatan pada <i>node</i> sensor ke <i>gateway</i> sistem tanpa <i>multitasking</i>	42
Tabel IV. 3 Hasil perbandingan data terkirim dan diterima pada sistem tanpa <i>multitasking</i>	42
Tabel IV. 4 Hasil perbandingan data terkirim dan diterima pada sistem tanpa <i>multitasking</i>	43
Tabel IV. 5 Hasil pengamatan pada <i>node</i> sensor ke <i>gateway</i> sistem <i>multitasking</i>	46
Tabel IV. 6 Hasil perbandingan data terkirim dan diterima pada sistem <i>multitasking</i>	46
Tabel IV. 7 Hasil pengamatan pada <i>node</i> sensor ke <i>gateway</i> sistem <i>multitasking</i>	48
Tabel IV. 8 Hasil perbandingan data terkirim dan diterima pada sistem dengan <i>multitasking</i>	48
Tabel IV. 9 Hasil perbandingan data terkirim dan diterima pada sistem <i>multitasking</i>	50
Tabel IV. 10 Total perbandingan jumlah data <i>loss</i> dan data diterima	50



DAFTAR SINGKATAN DAN ARTI SIMBOL

Lambang/Singkatan	Arti dan Keterangan
IoT	<i>Internet of Things</i>
LPWAN	<i>Low Power Wide Area Network</i>
LoRa	<i>Long Range</i>
TDS	<i>Total Dissolved Solid</i>
PPM	<i>Parts per Million</i>
NB-IoT	<i>Narrowband-Internet of Things</i>
ISM	<i>Industrial, Scientific, and Medical</i>
RMSE	<i>Root Mean Square Error</i>
RTOS	<i>Real Time Operation System</i>
RH	<i>Relative Humidity</i>
V	<i>Volt</i>
GND	<i>Ground</i>
VCC	<i>Voltage Common Collector</i>
y	Variabel terikat (dependen)
X	Variabel bebas (independen)
a	Konstanta
b	Koefisien regresi
T	Kode unik suhu pada <i>node</i> sensor 1
H	Kode unik kelembaban pada <i>node</i> sensor 1
D	Kode unik jarak pada <i>node</i> sensor 1
PH12	Kode unik pH pada <i>node</i> sensor 2
TDS1	Kode unik TDS pada <i>node</i> sensor 3



DAFTAR LAMPIRAN

Lampiran 1 <i>Source Code Gateway Tanpa Multitasking</i>	68
Lampiran 2 <i>Source Code Gateway Multitasking</i>	75
Lampiran 3 <i>Source Code Node Sensor 3</i>	83
Lampiran 4 <i>Source Code Node Sensor 2</i>	89
Lampiran 5 <i>Source Code Node Sensor 3</i>	92
Lampiran 6 Lembar Perbaikan Skripsi	95



KATA PENGANTAR

Puji syukur penulis panjatkan atas kehadiran Allah SWT berkat rahmat dan karunia-Nya sehingga Tugas Akhir yang berjudul “**PENERAPAN *MULTITASKING PADA EDGE COMPUTING UNTUK MONITORING SISTEM TANAMAN HIDROPONIK FARM***” ini dapat terselesaikan dengan baik sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Penulis menyadari betul bahwa dalam penyusunan tugas akhir ini terdapat banyak kendala, namun berkat dorongan, bantuan, bimbingan, serta dukungan dari berbagai pihak, sehingga penyusunan tugas akhir ini dapat terselesaikan dengan baik. Ucapan terima kasih serta penghargaan yang setinggi-tingginya penulis sampaikan kepada:

1. Orang tua penulis, alm. Bapak Ahmad Badjarad dan almh. Ibu Sarnita atas segala cinta dan kasih sayang yang diberikan kepada penulis semasa hidup serta selalu menjaga penulis dalam doa-doa papi dan mami, sehingga penulis dapat sampai pada tahap 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, pikiran, tenaga, dan perhatian yang luar biasa untuk mengarahkan penulis dalam menyusun tugas akhir.
3. Bapak Prof. Dr. Ir. Indrabayu S.T., M.T., M. Bus.Sys., IPM, ASEAN. Eng. selaku Ketua Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.
4. Saudara penulis Desy Apriliani Badjarad dan Alvin Rezky Badjarad atas segala perhatian, kasih sayang, doa dan dukungan serta menjadi penyemangat penulis selama ini, serta kedua keponakan lucu penulis Ling-Ling dan Abqary yang sudah menghibur selama ini.



urga besar Badjarad serta Aneka Makmur yang telah memberi banyak nangan, motivasi dan doa selama penulis menempuh pendidikan sejak kecil.

6. Sahabat-sahabat penulis, mbak Delil, mbak Izah, mbak Pume yang sudah dengan sabar mendengarkan segala keluh kesah dan curhatan-curhatan penulis selama menjalankan pendidikan perkuliahan dan memberikan doa, dukungan, serta semangat bagi penulis.
7. Teman seperjuangan penulis, Irma, Taslinda, Jumraini, Fitriani, Alfa, Wahyudi, Herlina, Fadil, Saphirah, Wahyu, Alisyah yang telah banyak membantu, mendukung, dan memberikan semangat dan juga menjadi partner dalam suka maupun duka.
8. Teman-teman RECOGN17ER atas dukungan dan semangat yang telah diberikan kepada penulis selama awal perkuliahan sampai penyelesaian tugas akhir.
9. Anggota Lab Riset IoTPC Fakultas Teknik Universitas Hasanuddin sebagai teman bertukar pikiran yang telah memberikan banyak masukan dan bantuan kepada penulis selama melakukan penelitian, dan juga anggota Lab UBICON yang telah menghibur penulis saat *stuck* dalam melakukan penelitian.
10. Segenap Dosen dan Staf Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah banyak membantu penulis selama masa perkuliahan.
11. Semua pihak yang tidak dapat disebutkan satu persatu yang telah turut mendoakan dan mendukung penulis selama proses penyelesaian tugas akhir ini.

Akhir kata, penulis berharap semoga Allah SWT berkenan membalas segala kebaikan dari semua pihak yang telah membantu. Semoga Tugas Akhir ini dapat memberikan manfaat serta menambah wawasan ilmu untuk pembaca dan juga penulis sendiri.

Makassar, 2023

Penulis



BAB I

PENDAHULUAN

1.1 Latar Belakang

Miliaran perangkat pintar kini dapat terhubung ke internet dalam bentuk *Internet of Things* (IoT) karena kemajuan teknologi jaringan. Data yang dihasilkan oleh perangkat IoT sangat penting bagi organisasi yang tertarik untuk meningkatkan produktivitas dan pendapatan mereka. Namun, manajemen dan analisis data dalam jumlah besar seperti itu rumit dan menantang bagi organisasi yang mengandalkan paradigma komputasi konvensional. *Edge computing* mendapatkan popularitas dalam konteks ini karena IoT menjadi umum dalam memproses data di tepi jaringan (Hassan et al., 2018).

Edge computing merupakan suatu konsep pengkomputasian data yang mana terjadi sebelum pada bagian “*edge*” dengan cara data dari seluruh perangkat IoT akan dikumpulkan, disimpan dan dianalisis secara lokal pada bagian “*edge*” sebelum data dikirim ke *cloud*. *Edge computing* memiliki potensi untuk mengatasi permasalahan waktu respon yang terbilang lambat serta *bandwidth* yang besar (Shi et al., 2016).

Dalam *edge computing*, perangkat akhir tidak hanya mengkonsumsi data tetapi juga menghasilkan data. Dan di tepi jaringan, perangkat tidak hanya meminta layanan dan informasi dari *cloud* tetapi juga menangani tugas komputasi termasuk pemrosesan, penyimpanan, *caching*, dan penyeimbangan beban pada data yang dikirim ke dan dari *cloud*. *Edge* harus dirancang dengan cukup baik untuk menangani tugas-tugas tersebut secara efisien, andal, aman, dan dengan mempertimbangkan privasi. Dengan demikian harus mendukung persyaratan seperti diferensiasi, ekstensibilitas, isolasi, dan keandalan (Shi & Dustdar, 2016).

Pertanian pintar telah mulai menggabungkan solusi IoT untuk meningkatkan efisiensi operasional, memaksimalkan hasil, dan meminimalkan pemborosan melalui pengumpulan data lapangan waktu nyata, analisis data, dan penerapan mekanisme kontrol. Selain itu, beragam aplikasi berbasis IoT seperti pertanian dan irigasi pintar sangat membantu peningkatan proses pertanian. Dengan , IoT dianggap sebagai salah satu solusi yang menjanjikan untuk al pertanian yang terhubung untuk mengatasi masalah berbasis pertanian



dan meningkatkan kualitas dan kuantitas produksi pertanian (Alharbi & Aldossary, 2021).

Dalam pertanian, beberapa parameter dapat dipantau secara *remote*. Besaran tersebut diantaranya: suhu, kelembaban, kecepatan angin, intensitas cahaya matahari, dan kebasahan tanah. Sangat bermanfaat untuk menganalisis besaran tersebut dalam tabel atau grafik *historical*. Hasil analisis ini diperlukan untuk melakukan tindakan atau keputusan untuk memaksimalkan hasil pertanian tersebut (Hardana, 2019).

Banyaknya *node* sensor yang mengirimkan data dalam *monitoring* hidroponik *farm* meningkatkan beban pada *gateway*, karena selain menerima data, *gateway* juga memiliki tugas untuk menjalankan proses komputasi. Sehingga *gateway* harus meluangkan waktunya bukan hanya untuk sekedar menerima data tetapi juga melakukan komputasi, jika terlalu berat beban sebuah *gateway*, ada kemungkinan hanya memiliki sedikit waktu untuk menerima data dari *node* sensor hidroponik *farm* sehingga dapat terjadi hilangnya data.

Pada penelitian yang telah dilakukan oleh Auliati Nisa pada tahun 2018 mengatakan bahwa, banyaknya *node* sensor yang mengirimkan data ke *gateway* dalam waktu yang bersamaan rentan mengalami *collision* sehingga menyebabkan ada data yang hilang atau tidak diterima. Selain *collision*, penyebab hilangnya data juga dikarenakan adanya proses komputasi yang sedang dilakukan oleh *gateway* yang menggunakan pemrograman secara konvensional sehingga menyebabkan data yang dikirim tidak sempat untuk diterima. Dikarenakan, pemrograman konvensional berjalan secara *sequential* dimana *task* akan diproses setelah *task* sebelumnya telah selesai diproses. *Gateway* hanya akan menerima data yang lain ketika data yang sebelumnya telah selesai diproses dan dikirim ke *cloud*.

Berdasarkan permasalahan diatas, untuk mengurangi terjadinya data yang hilang dan mengefisienkan kerja sistem maka diterapkannya sistem *multitasking*, sehingga sistem dapat menjalankan beberapa *task* dalam satu waktu tanpa mengganggu pekerjaan *task* yang lain, serta dengan menggunakan *multitasking* kita



mberikan prioritas yang adil pada *gateway* dalam melakukan komputasi menerima data. Konsep *multitasking* diterapkan pada *edge* dalam hal ini adalah dikarenakan dengan menggunakan *edge computing* dapat menghemat

bandwidth serta mengurangi latensi karena data akan diproses di tepi jaringan. Serta keuntungan dari *edge computing* yaitu memungkinkan reaksi cepat terhadap peristiwa melalui aktuasi. Serta diharapkan dapat membuat keputusan dan memberikan informasi yang tepat dalam waktu yang aktual.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah:

1. Apakah kinerja sisi penerima (*edge device*) bertanggung jawab pula terhadap hilangnya data yang dikirim oleh sensor via LoRa?
2. Bagaimana hubungan antara kecepatan *edge device* memproses data yang ada dengan jumlah banyaknya data yang hilang dari sensor?
3. Apakah *multitasking* memberikan peningkatan penerimaan data dari sensor LoRa dibanding pemrograman konvensional?

1.3 Tujuan

Tujuan yang ingin dicapai dari penelitian ini yaitu:

1. Untuk mendapatkan bukti bahwa kinerja sisi penerima (*edge device*) bertanggung jawab terhadap hilangnya data yang dikirim oleh sensor via LoRa.
2. Untuk mendapatkan pengetahuan mengenai hubungan antara kecepatan *edge device* memproses data yang ada dengan jumlah data yang hilang dari sensor.
3. Untuk mendapatkan bukti bahwa *multitasking* memberikan peningkatan penerimaan data dari *node* sensor LoRa dibanding pemrograman konvensional.

1.4 Manfaat Penelitian

Adapun manfaat dari penelitian ini yaitu:

1. Penelitian ini diharapkan dapat meningkatkan responsivitas *edge device* dalam melayani sejumlah skala besar *node* sensor.
2. Penelitian ini diharapkan dapat menambah pengetahuan dan memberikan kontribusi untuk pengembangan literatur khususnya mengenai penerapan *tasking* pada *edge computing*. Selain itu penelitian ini diharapkan dapat jadi referensi bagi penelitian-penelitian selanjutnya.



1.5 Batasan Masalah

1. Tanaman yang dimonitoring adalah selada air.
2. Pada penelitian ini akan menggunakan modul LoRa sebagai alat komunikasi.
3. Sensor yang digunakan yaitu sensor DHT11, sensor pH, sensor TDS, sensor HC-SR04.
4. Penelitian ini berfokus pada bagaimana cara untuk mengurangi data yang hilang akibat beratnya komputasi yang dilakukan pada *gateway*.
5. Mikrokontroler yang digunakan adalah ESP32.



BAB II TINJAUAN PUSTAKA

2.1 Internet of Things

Konsep IoT diciptakan oleh anggota komunitas pengembangan *Radio Frequency Identification* (RFID) pada tahun 1999. Saat ini, perkembangan IoT sangat meningkat didorong oleh pertumbuhan perangkat seluler, komunikasi yang semakin mudah dilakukan, komputasi awan, dan analitik data. Tujuan *Internet of Things* adalah untuk memungkinkan segala sesuatunya selalu terhubung kapan saja dengan apa saja dan siapa saja yang ideal menggunakan jaringan dan layanan apapun. *Internet of Things* mengacu pada penggunaan perangkat dan sistem yang terhubung untuk memanfaatkan data yang diperoleh dari sensor, aktuator yang tertanam dalam mesin, atau objek fisik lainnya. Bagi konsumen, IoT berpotensi memberikan solusi yang dapat meningkatkan efisiensi energi, keamanan, kesehatan, pendidikan, dan aspek kehidupan sehari-hari lainnya. Sedangkan untuk perusahaan, IoT dapat mendukung solusi yang meningkatkan pengambilan keputusan dan produktivitas di bidang manufaktur, ritel, pertanian, dan sektor lainnya (Thamrin et al., 2020).

Internet of Things (IoT) telah menemukan penerapannya di beberapa bidang, seperti industri yang terhubung, kota pintar, rumah pintar energi pintar, mobil yang terhubung, pertanian pintar, gedung dan kampus yang terhubung, perawatan kesehatan, logistik, di antara domain lainnya. IoT bertujuan untuk mengintegrasikan dunia fisik dengan dunia virtual dengan menggunakan internet sebagai media untuk berkomunikasi dan bertukar informasi. IoT telah didefinisikan sebagai sistem perangkat komputasi yang saling terkait, mesin mekanis dan digital, objek, hewan, atau orang yang dilengkapi dengan pengidentifikasi unik dan kemampuan untuk mengirim data melalui jaringan tanpa memerlukan manusia ke manusia atau manusia ke manusia (Elijah et al., 2018).



Internet of Things adalah revolusi baru internet. Objek membuat diri mereka dan mereka memperoleh kecerdasan dengan membuat atau memungkinkan mereka terkait konteks berkat fakta bahwa mereka dapat mengkomunikasikan

informasi tentang diri mereka sendiri. Mereka dapat mengakses informasi yang telah dikumpulkan oleh hal-hal lain, atau mereka dapat menjadi komponen dari layanan yang kompleks. Transformasi ini seiring dengan munculnya kemampuan komputasi awan dan transisi internet menuju Ipv6 dengan kapasitas pengalaman yang hampir tidak terbatas. Tujuan *Internet of Things* adalah untuk memungkinkan segala sesuatu terhubung kapan saja, dimana saja, dengan apa saja dan siapa saja, idealnya menggunakan jalur atau jaringan dan layanan apapun (Elijah et al., 2018).

Definisi umum *Internet of Things* didefinisikan sebagai: *Internet of Things* (IoT) adalah jaringan objek fisik. Internet bukan hanya jaringan komputer, tetapi telah berkembang menjadi jaringan perangkat dari semua jenis dan ukuran, kendaraan, ponsel pintar, peralatan rumah tangga, mainan, kamera, peralatan medis dan sistem industri, hewan, manusia, bangunan, semua terhubung, semua berkomunikasi dan berbagi informasi berdasarkan protokol yang ditetapkan untuk mencapai reorganisasi cerdas, penentuan posisi, penelusuran, keamanan dan kontrol, bahkan pemantauan *online* waktu nyata pribadi, peningkatan *online*, kontrol proses dan administrasi (Patel & Keyur, 2016).

Menurut Patel (2016), mendefinisikan IoT menjadi tiga kategori yang pertama *People to People*, kedua *People to things*, ketiga *Things machine to Things machine*, berinteraksi melalui Internet. *Internet of Things Vision: Internet of Things (IoT)* adalah sebuah konsep dan paradigma yang mempertimbangkan kehadiran yang meluas di lingkungan berbagai hal atau objek yang melalui koneksi nirkabel dan kabel dan skema pengalaman yang unik dapat berinteraksi satu sama lain dan bekerja sama dengan hal atau objek lain untuk membuat aplikasi atau layanan baru dan mencapai tujuan bersama. Dalam konteks ini, tantangan penelitian dan pengembangan untuk menciptakan dunia yang cerdas sangat besar. Dunia dimana dunia nyata, digital, dan virtual bertemu untuk menciptakan lingkungan cerdas yang membuat energi, transportasi, kota, dan banyak area lainnya menjadi lebih cerdas.

2.2 Edge Computing



Struktur komputasi awan telah menjadi pusat banyak ekosistem IoT. Saat ini, semakin banyak orang mengandalkan aplikasi dan layanan yang secara langsung maupun tidak langsung (melalui ketergantungan operator) yang didukung oleh Sistem

Komputasi Awan. Secara paralel, kemajuan dalam komputasi seluler dan teknologi manufaktur telah menyebabkan ledakan di perangkat konsumen seperti *Smartphone*, tablet, jam tangan pintar, pelacak kebugaran, perangkat realitas virtual, sensor IoT, aktuator, dan banyak lagi. Menjadi semakin menantang untuk memproses sifat yang beragam dan volume besar data yang dihasilkan oleh perangkat ini. *Big Data* tersebut dicirikan oleh volume, kejujuran, kecepatan dan variasi. Aplikasi dan layanan IoT yang menangani *Big Data* semacam itu mengandalkan komputasi awan untuk menghasilkan pola informasi, abstraksi tingkat tinggi, dan pemrosesan lebih lanjut. Tetapi skenario IoT yang muncul dan generasi berikutnya (misalnya kendaraan otonom) memerlukan pemrosesan dan aktuasi data *real-time* yang menuntut data IoT untuk diproses di dekat asalnya, yaitu tepi jaringan IoT. *Edge Computing* (EC), memungkinkan pengolahan data lokal dan terdistribusi dan telah mendapatkan banyak momentum baik dari akademisi maupun industri (Datta & Bonnet, 2017).

Edge computing merupakan suatu konsep pengkomputasian data yang mana terjadi sebelum pada bagian “*edge*” dengan cara data dari seluruh perangkat IoT akan dikumpulkan, disimpan, dan dianalisis secara local pada bagian “*edge*” sebelum data dikirim ke *cloud*. *Edge computing* memiliki potensi untuk mengatasi permasalahan waktu respon yang terbilang lambat serta *bandwidth* yang besar serta *edge computing* juga mampu meningkatkan keamanan dan privasi data yang ada pada sebuah perangkat IoT (Shi et al., 2016).

Edge computing memungkinkan analisis data secara *real time*. Menganalisis data di tempat pembuatan data dapat mengurangi latensi pembuatan informasi dari data yang dikumpulkan. oleh karena itu, *edge devices* dapat mengumpulkan dan menganalisis data dari perangkat sekitarnya, sehingga memungkinkan pembuat keputusan untuk memberikan wawasan tindaklanjuti lebih cepat dari sebelumnya. *edge device* juga dapat membantu mengurangi *bandwidth* jaringan dan biaya karena data akan dianalisis secara lokal. hal ini dapat membantu banyak organisasi di industri, termasuk manufaktur, layanan kesehatan, telekomunikasi, dan sehingga kebutuhan akan konsep IoT meningkat (Hassan et al., 2018).



Banyaknya data-data mentah yang dihasilkan oleh perangkat IoT membuat komputasi awan secara konvensional tidak cukup efisien dalam menangani semua data-data ini dalam artian akan menambah latensi serta *bandwidth* ketika komunikasi data terjadi. *Edge computing* memungkinkan untuk melakukan komputasi pada bagian tepi dari sebuah sistem IoT yang berada diantara perangkat IoT dan juga *cloud*. Pada dasarnya penggunaan *edge computing* adalah menempatkan komputasi pada bagian yang lebih dekat pada sistem IoT yaitu bagian *edge* sehingga hal ini sangat berpengaruh dalam latensi komunikasi data dibandingkan dengan perangkat IoT yang langsung menggunakan komputasi awan. Dengan menggunakan *edge computing* dapat mengurangi sekitar 30% - 40% konsumsi energi yang digunakan serta mengurangi latensi dari 900 ms menjadi 169 ms sehingga didapatkan 20 kali lipat lebih efisien dalam proses menjalankan sistem dari segi latensi serta energi yang dibutuhkan (Shi et al., 2016).

Dalam paradigma komputasi *edge*, hal-hal yang tidak hanya konsumen data, tetapi juga berperan sebagai produsen data. Di *edge*, hal-hal tidak hanya dapat meminta layanan dan konten dari *cloud*, tetapi juga melakukan tugas komputasi dari *cloud*. *Edge* dapat melakukan komputasi *offloading*, penyimpanan data, *caching* dan pemrosesan, serta mendistribusikan layanan permintaan dan pengiriman dari *cloud* ke pengguna. Dengan pekerjaan tersebut di jaringan, *edge* itu sendiri perlu dirancang dengan baik untuk memenuhi persyaratan secara efisien dalam layanan seperti keandalan, keamanan, dan perlindungan privasi (Shi et al., 2016).

2.3 LPWAN (*Low Power Wide Area Network*)

Dengan munculnya *Internet of Things* (IoT) dan komunikasi mesin ke mesin (M2M), pertumbuhan besar-besaran dalam penyebaran *node* sensor diharapkan segera. Survei IoT yang dilaporkan web Forbes pada tahun 2019, memperkirakan lebih dari 75 miliar perangkat yang akan terhubung ke IoT pada tahun 2025. Dengan perkembangan yang sangat diantisipasi di bidang kecerdasan buatan, *machine*



analisis data, dan teknologi *Blockchain*, ada potensi besar untuk secara sial menumbuhkan penyebaran dan aplikasi LPWAN di hampir semua masyarakat, profesi, dan industri. Perkembangan tersebut memungkinkan

perangkat seperti sensor, kendaraan, robot, mesin atau objek lain semacam itu untuk terhubung ke Internet. Jaringan memungkinkan mereka mengirim data dan parameter yang dirasakan ke perangkat atau server terpusat jarak jauh, misalnya *cloud*. *Cloud* menyediakan kecerdasan untuk membuat keputusan atau tindakan yang tepat (Chaudhari et al., 2020).

Aplikasi IoT memiliki persyaratan khusus seperti jarak jauh, kecepatan data rendah, konsumsi energi rendah, dan efektivitas biaya. Teknologi radio jarak pendek yang banyak digunakan (misalnya *ZigBee*, *Bluetooth*) tidak sesuai untuk skenario yang memerlukan transmisi jarak jauh. Solusi berdasarkan komunikasi seluler (misal 2G, 3G, dan 4G) dapat memberikan cakupan yang lebih luas, tetapi menghabiskan energi perangkat yang berlebihan. Oleh karena itu, kebutuhan aplikasi IoT telah mendorong munculnya teknologi komunikasi nirkabel baru yaitu *Low Power Wide Area Network (LPWAN)* (Mekki et al., 2019).

LPWAN semakin mendapatkan popularitas di komunitas industri dan penelitian karena karakteristik komunikasinya yang berdaya rendah, jarak jauh, dan berbiaya rendah. Ini menyediakan komunikasi jarak jauh dari hingga 10 sampai 40 kilometer di zona pedesaan dan 1 sampai 5 kilometer di zona perkotaan. Aspek LPWAN yang menjanjikan ini telah mendorong studi eksperimental terbaru tentang kinerja LPWAN di lingkungan luar dan dalam ruangan (Mekki et al., 2019).

Teknologi LPWAN dirancang untuk memberikan cakupan area yang luas. Hal ini didapat dengan menggunakan pita frekuensi dibawah 1 GHz (Sub-GHz band) dan teknik modulasi khusus. Penggunaan Sub-GHz *band* mempunyai kelebihan karena memiliki redaman propagasi yang lebih kecil dan beroperasi di wilayah frekuensi yang bukan wilayah sistem komunikasi radio pita lebar pada umumnya seperti *Wi-Fi*, *Bluetooth*, *Wi-Max* dan *Broadband Wireless Access* lainnya. Hal ini memberikan keuntungan dalam meminimalisir interferensi. Teknik modulasi yang digunakan LPWAN dirancang untuk mencapai *budget* 150 dB sehingga mampu mencapai puluhan kilometer pada rural area. Mekanisme transfer data LPWAN juga memberikan kelebihan dimana menjadikan pengguna dapat beroperasi dengan konsumsi daya rendah dan juga dengan biaya operasional. Teknologi LPWAN dapat dibagi menjadi dua yaitu



teknologi berbasis 3GPP dan teknologi proprietary atau non-3GPP. Beberapa contoh teknologi 3GPP adalah LTE-M, EC-GSM dan NB-IoT. Sedangkan contoh teknologi non-3GPP adalah, SigFox, Ingenu RPMA, *Wightless*, dan LoRaWAN (Istiana et al., 2020).

2.3.1 NB-IoT

Narrowband Internet of Things (NB-IoT), *Machine Type Communications* (MTC) dan *Extended Coverage GSM* (EC-GSM) adalah tiga teknologi baru yang diperkenalkan oleh Proyek Kemitraan Generasi Ketiga (3GPP) untuk jaringan *Low-Power Wide-Area* (LPWA). Meskipun NB-IoT terintegrasi ke dalam standar LTE, NB-IoT menyederhanakan banyak fitur LTE untuk mengurangi biaya dan meminimalkan konsumsi energi. Tujuan utama NB-IoT adalah untuk mendukung *massive machine-type communication* (mMTC) dan memungkinkan komunikasi tingkat daya rendah, berbiaya rendah, dan juga laju data kecil. Selain itu, dengan memanfaatkan pita yang lebih sempit, harga chip NB-IoT juga diturunkan sehingga dapat digunakan secara luas.

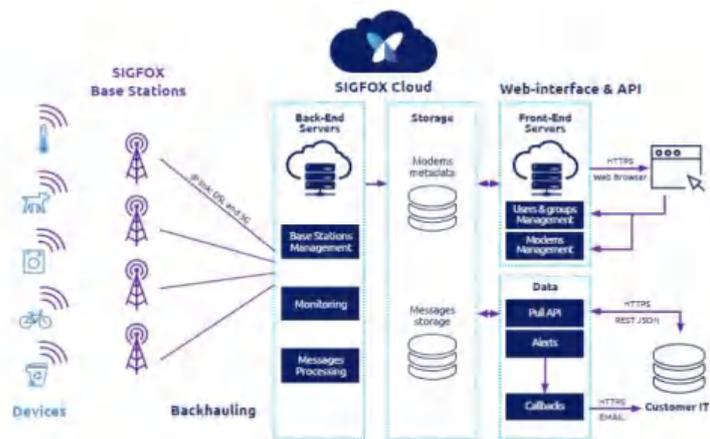
Istilah *Narrowband* mengacu pada *bandwidth* NB-IoT maksimum 200 kHz yang dapat berdampingan dalam spektrum *Global System for Mobile Communications* (GSM) atau dengan menempati salah satu *LTE Physical Resource Blocks* (PRBs) sebagai *in-band* atau sebagai *guard-band* maupun *standalone* dalam lisensi spektrum frekuensi yang sama. Karena beroperasi berdampingan dalam spektrum LTE, NB-IoT juga mengikuti numerologi LTE karena menggunakan *Orthogonal Frequency Division Multiplexing* (OFDM) dan *Single-Carrier Frequency Division Multiple Access* (SC-FDMA) dalam skema transmisi *downlink* dan *uplink* dan menggunakan modulasi *quadrature phase shift keying* (QPSK) (Istiana et al., 2020).

NB-IoT memungkinkan konektivitas hingga 100 ribu perangkat per sel dan berpotensi untuk meningkatkan kapasitas dengan menambahkan lebih banyak *carrier* NB-IoT. Kecepatan data dibatasi hingga 200 kbps untuk *downlink* dan 200 kbps *uplink*. Ukuran *payload* maksimum untuk setiap pesan adalah 1600 *byte*. Masa pakai NB-IoT dapat mencapai 10 tahun masa pakai baterai ketika rata-rata transmisi 200 *byte* per hari (Istiana et al., 2020).



2.3.2 Sigfox

Sigfox adalah operator jaringan LPWAN yang menawarkan mengakhiri solusi konektivitas IoT berdasarkan pada teknologi yang dipatenkannya. Sigfox menyebarkan stasiun pangkalan miliknya yang dilengkapi dengan radio yang ditentukan perangkat lunak kognitif dan menghubungkannya ke *server back end* menggunakan jaringan berbasis IP. Perangkat akhir yang terhubung ke *Base Transceiver Station (BTS)* menggunakan modulasi *Binari Phase Shift Keying (BPSK)* modulasi dalam pita ultra-sempit (100 Hz) GHz ISM *band carrier*. Sigfox menggunakan *band ISM* yang tidak berlisensi, misalnya 868 MHz di Eropa, 915 MHz di Amerika Utara, dan 433 MHz di Asia.



Gambar 2. 1 Arsitektur sigfox

Gambar 2.1 merupakan arsitektur dasar Sigfox adalah kunci untuk meminimalkan CAPEX dan OPEX. Sigfox *Software Defined Radio (SDR)* membantu mengatasi biaya perangkat keras yang tinggi untuk BTS. Tidak ada perangkat keras khusus yang digunakan selain algoritma perangkat lunak untuk menangani demodulasi secara efektif. Ini mengurangi secara signifikan *Total Cost Operation (TCO)*. Data dikirim melalui udara ke stasiun pangkalan, kemudian melewati *backhaul*. *Backhaul* umumnya menggunakan konektivitas DSL dan 3G atau 4G sebagai cadangan. Ketika salah satu dari keduanya tidak tersedia,



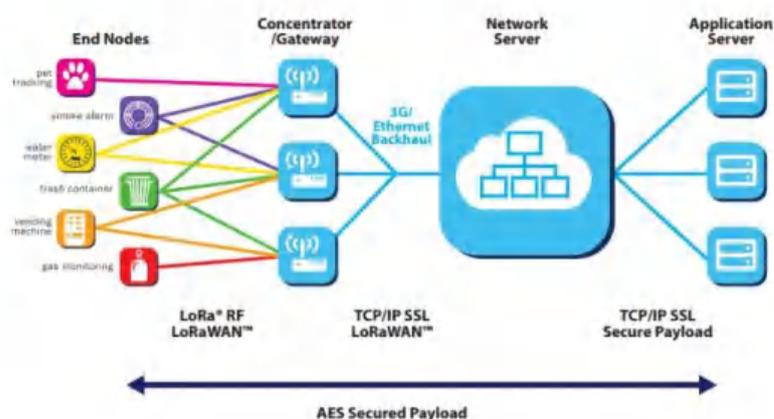
tas satelit dapat digunakan sebagai teknologi cadangan alternatif. Bagian menangani pemrosesan pesan. Ada banyak kemungkinan replikasi dari ig sama yang tiba di jaringan inti tetapi hanya satu yang harus disimpan.

Server jaringan inti juga memonitor status jaringan dan mengelola stasiun pangkalan secara global (Istiana et al., 2020).

2.3.3 LoRa

LoRa yang merupakan singkatan dari “*Long Range*”, adalah sistem komunikasi nirkabel jarak jauh berdaya rendah untuk perangkat kecil berdaya rendah. Ini membuatnya sangat cocok tidak hanya untuk perangkat yang bekerja dengan baterai tetapi juga untuk implementasi IoT. LoRa adalah teknologi lapisan fisik yang memodulasi sinyal dalam bentuk sub-GHZ ISM band menggunakan teknik *spread spectrum* eksklusif. LoRa menggunakan enam faktor penyebaran (SF7 ke SF12) untuk mengadaptasi *data rate* dan *range trade off*.

Protokol komunikasi berbasis LoRa yang disebut LoRaWAN telah distandarisasi oleh *LoRa-Alliance* (versi pertama pada tahun 2015). Beberapa komponen jaringan ditentukan dan diperlukan untuk membentuk LoRaWAN yaitu perangkat akhir, *gateway*, dan server jaringan. Dalam LoRaWAN *node* jaringan tidak terkait dengan *gateway* tertentu. Sebagai gantinya, data yang dikirimkan oleh suatu simpul biasanya diterima oleh banyak *gateway* dalam jangkauan *network* LoRa. Setiap *gateway* akan meneruskan paket yang diterima dari *end-node* ke server jaringan berbasis *cloud* melalui beberapa *backhaul* (baik seluler, ethernet, satelit atau Wi-Fi). Gambar 2.2 dibawah merupakan struktur arsitektur LoRa.



Gambar 2. 2 Arsitektur LoRa



2.4 Real Time Operation System (RTOS)

Sistem waktu nyata adalah sistem yang harus menghasilkan respon yang tepat dalam batas waktu yang telah ditentukan. Jika respon yang dihasilkan melewati batas waktu tersebut, maka akan terjadi degradasi performansi atau bahkan kegagalan sistem. Suatu sistem dikatakan *real time* jika memiliki *deadline* atau jangka waktu penyelesaian tertentu, namun tetap mengutamakan ketepatan dan performa yang tinggi dalam prosesnya. Agar dapat mewujudkan sistem yang dapat bekerja secara *real time* dibutuhkan sebuah sistem operasi yang khusus diperuntukkan untuk operasi *real time* yang dikenal dengan nama *real time operation system* (RTOS) (Kurniawan et al., 2010).

Real time operation system telah berkembang secara dramatis dalam beberapa dekade terakhir. Aplikasi sistem ini dianggap kuat dan selalu menjadi tantangan besar bagi desainer. Sistem ini harus memastikan bahwa batasan waktu terpenuhi saat menjalankan aktivitas yang kompleks. RTOS adalah jenis sistem operasi yang dimaksudkan untuk beroperasi secara *real time*, seringkali tanpa penunda *buffer*, yang memproses data saat masuk. Persyaratan waktu untuk pemrosesan (termasuk penundaan OS) diukur dalam 10 detik atau peningkatan yang lebih pendek (Ismael et al., 2021).

RTOS adalah sistem operasi yang mendukung dan menjamin respons tepat waktu terhadap kejadian eksternal dan internal sistem waktu nyata. RTOS mengeksekusi program-program dalam sebuah pola yang teratur. RTOS merupakan sistem operasi yang berbeda dari sistem operasi pada umumnya. Sistem operasi pada komputer akan bekerja sesaat ketika daya masuk ke komputer, kemudian program komputer berjalan. Sedangkan RTOS, sistem operasi ini dijalankan oleh sebuah program otomatis yang merupakan sebuah *kernel*. Ketika sistem dinyalakan, maka *kernel* akan menyala terlebih dahulu kemudian menyalakan sistem operasi *real time*. Komponen utama dari *kernel* RTOS adalah penjadwal tugas yang bertanggung jawab untuk menetapkan sumber daya sistem ke tugas sesuai dengan

yang sesuai.

kernel merupakan salah satu bagian dari sistem *multitasking* yang memiliki fungsi sebagai manajemen dari seluruh *task*, mengatur komunikasi



antar *task* dan yang terpenting adalah mengatur waktu untuk CPU sehingga tidak terjadi *crash*. Layanan standar yang disediakan oleh *kernel* adalah *context switching*. Penggunaan *kernel* akan menyederhanakan perancangan sistem dengan cara membagi aplikasi ke dalam beberapa *task* yang dikelola oleh *kernel* (Kurniawan et al., 2010)

2.5 freeRTOS

FreeRTOS adalah *portable, open source* (berlisensi di bawah GPL yang dimodifikasi), sistem operasi mini *real time* yang dikembangkan oleh *Real Time Engineers Ltd*. Ini porting ke 23 arsitektur perangkat keras mulai dari 8-bit hingga 32-bit mikrokontroler, dan mendukung banyak alat pengembangan. Keuntungan utamanya adalah portabilitas, skalabilitas, dan kesederhanaan. Kernel inti sederhana dan kecil, terdiri dari tiga atau empat (tergantung pada pengguna coroutine) file C dengan beberapa fungsi *assembler*, dengan citra biner antara 4 hingga 9 KB (Inam et al., 2011).

FreeRTOS adalah salah satu jenis *Real Time Operating System* 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 *processor* 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*. Fitur tersebut diantaranya *real time scheduling functionality, inter-task communication, timing, dan synchronization primitives* (Wisnu Jatmiko, Petrus Mursanto, 2015).

2.6 Multitasking

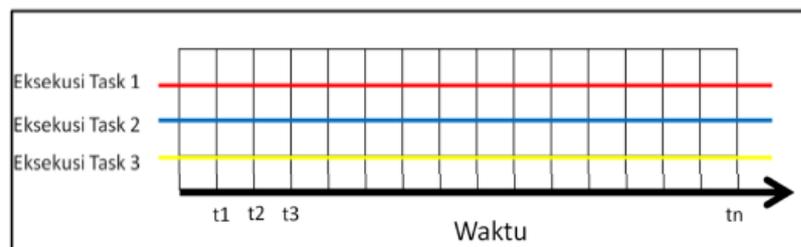
Sistem operasi memiliki komponen utama untuk mengatur adanya *multitasking*. *Multitasking* sendiri adalah kemampuan sistem operasi untuk menjalankan banyak *task* atau *thread* dalam rentang waktu tertentu. Setiap *task* terbentuk dari setiap proses yang harus dijalankan oleh sebuah komputer sistem tersepat. Konsep *multitasking* dari sistem operasi akan



memudahkan desain sebuah aplikasi yang kompleks (Wisnu Jatmiko, Petrus Mursanto, 2015).

Konsep *multitasking* dari sistem operasi akan memudahkan desain sebuah aplikasi yang kompleks. Gambar 2.3 dibawah merupakan gambar dari konsep *multitasking*. Terdapat beberapa keuntungan dari *multitasking* sebagai berikut : (Wisnu Jatmiko, Petrus Mursanto, 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*

2.7 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.

Adapun sensor yang akan digunakan pada penelitian ini, yaitu:

2.7.1 Sensor DHT11



Gambar 2. 4 Sensor DHT11

Gambar 2.4 merupakan sensor DHT11. Sensor kelembaban dan suhu DHT11 adalah periferan ekonomis yang diproduksi oleh D-Robotics UK. Alat ini mampu mengukur kelembaban relatif antara 20 dan 90% RH dalam kisaran suhu pengoperasian 0 hingga 50 °C dengan akuisisi $\pm 5\%$ RH. Suhu juga diukur dalam kisaran 0 hingga 50 °C dengan akurasi ± 2 °C. Kedua nilai dikembalikan dengan resolusi 8-bit.

DHT11 merupakan sensor suhu dan kelembaban dengan keluaran sinyal digital yang dikalibrasi. Dengan menggunakan teknik akuisisi sinyal digital eksklusif dan teknologi penginderaan suhu dan 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.

2.7.2 Sensor HC-SR04



Gambar 2. 5 Sensor Ultrasonik HC-SR04

Gelombang ultrasonik dibangkitkan melalui *piezoelektrik* dengan frekuensi tertentu. *Piesoelektrik* akan menghasilkan gelombang ultrasonik (umumnya berfrekuensi 40 kHz) ketika sebuah osilator diterapkan pada benda tersebut. Secara umum, alat ini akan menembakkan gelombang ultrasonik menuju suatu area atau suatu target, setelah gelombang menyentuh permukaan target, maka gelombang dapat dipantulkan kembali. Gelombang pantulan dari target akan ditangkap oleh sensor, kemudian sensor akan menghitung selisih antara waktu pengirim gelombang dan waktu gelombang pantul diterima (Frima Yudha & Sani, 2019).

Gambar 2.5 diatas merupakan sensor HC-SR04. Sensor ultrasonik HC-SR04 akan sonar untuk menentukan jarak suatu objek seperti yang dilakukan elawar. Sensor ini terdiri dari rangkaian pemancar ultrasonik yang in transmitter dan penerima ultrasonik yang disebut receiver. Kisaran jarak



yang dapat diukur sekitar 2-450 cm. Dengan tingkat presisi 0,3 cm. Sudut deteksi yang bisa ditangani tidak lebih dari 15°. Arus yang dibutuhkan sebesar +5V. Jumlah pin adalah 4. Perangkat ini menggunakan dua pin digital untuk mengkomunikasikan jarak yang terbaca. Pengoperasian tidak terpengaruh oleh sinar matahari atau bahan hitam, meskipun secara akustik bahan lembut seperti kain sulit dideteksi.

2.7.3 Sensor pH



Gambar 2. 6 Sensor derajat keasaman (pH)

Gambar 2.6 menunjukkan Sensor pH meter adalah sensor yang mengukur tingkat keasaman (asam) atau kebasaan (basa) dari suatu larutan atau cairan. Prinsip kerja sensor pH yaitu berada pada sensor *probe* yang berupa elektroda kaca yang diisi dengan larutan HCL yang berada pada ujung sensor yang mengukur jumlah ion H_3O^+ di dalam larutan.

Sensor pH didasarkan pada perbedaan potensial elektrokimia yang terjadi antara larutan yang diketahui di dalam elektroda dan larutan yang tidak diketahui di luar elektroda. Sensor ini memiliki elektroda kaca dengan ujung bulat. Terdapat pada tiga pin pada sensor yaitu pin DATA, VCC dan GND.



2.7.4 Sensor TDS



Gambar 2. 7 Sensor *Total Dissolved Solid* (TDS)

Gambar 2.7 merupakan sensor TDS atau *Total Dissolved Solid* adalah sensor yang digunakan untuk mengukur konsentrasi massa kimiawi yang terlarut pada suatu zat, satuan yang digunakan yaitu *part per million* (ppm) atau *milligram per liter* (mg/l). Prinsip kerja dari sensor TDS meter yaitu menggunakan metode konduktivitas listrik yang memiliki kemampuan menghantarkan listrik yang berhubungan dengan konsentrasi padatan terlarut yang terionisasi di dalam air, dimana dua buah *probe* dicelupkan kedalam cairan atau larutan kemudian rangkaian pemrosesan sinyal akan menghasilkan *output* yang menunjukkan konduktivitas larutan tersebut. Sensor ini memiliki tiga pin yaitu pin DATA, VCC dan GND.

2.8 Thingspeak



Gambar 2. 8 *Thingspeak*

Gambar 2.8 merupakan logo dari *ThingSpeak*. *Thingspeak* adalah platform sumber API IoT terbuka berbasis web yang komprehensif dalam an data sensor dari berbagai aplikasi IoT dan menggabungkan keluaran n bentuk grafis di tingkat web. *Thingspeak* berkomunikasi dengan bantuan



koneksi internet yang bertindak sebagai ‘paket data’ pembawa antara ‘things’ yang terhubung dan *cloud thingspeak* mengambil, menyimpan, menganalisis, mengamati, dan mengerjakan data dari sensor yang terhubung ke mikrokontroler.

Fitur yang paling utama dari *thingspeak* adalah *channel* yang memiliki bidang untuk data, bidang untuk lokasi, dan bidang untuk status. Setelah *channel* dibuat data dapat diproses dan divisualisasikan menggunakan MATLAB. Thingspeak juga menyediakan fitur untuk membuat *channel* secara publik untuk menganalisis dan memperkirakan nya melalui publik ataupun secara pribadi.

2.9 Algoritma Regresi Linear

Regresi (*regression*) adalah proses identifikasi relasi dan pengaruhnya pada nilai-nilai objek. Regresi bertujuan menemukan suatu fungsi yang memodelkan data dengan meminimalkan galat atau selisih antara nilai prediksi dengan nilai sebenarnya. Regresi dapat dipandang sebagai alat ukur untuk mengetahui adanya korelasi antar variabel dan bisa juga untuk menentukan tingkat perubahan suatu variabel terhadap variabel lainnya (Jayanti, 2018)

Menurut Cohen (2020), analisis regresi linier sederhana digunakan dalam situasi dimana satu variabel independen dihipotesiskan mempengaruhi satu variabel dependen. Dan menurut Kavitha (2017), regresi linier ini untuk mengetahui arah hubungan antara variabel independen dengan variabel dependen apakah positif atau negatif dan untuk memprediksi nilai dari variabel dependen apabila nilai variabel independen mengalami kenaikan atau penurunan. Persamaan umum metode regresi linier sederhana adalah :

$$y = a + bX \quad (2.1)$$

Dimana :

y = variabel terikat (dependen)

X = variabel bebas (independen)



anta

sien regresi