

BAB 2 DASAR TEORI

2.1 Kajian Pustaka

Penelitian [1] membahas tentang kondisi perairan laut Indonesia yang memiliki potensi perikanan sangat besar. Metode yang digunakan dalam penelitian ini adalah perencanaan pembuatan dan pengembangan perangkat lunak atau model *waterfall*. Adapun parameter yang dibahas adalah kelembaban dan suhu, dengan hasil ditampilkan pada sebuah *web dashboard* LoRa yang memiliki beberapa menu seperti data LoRa, grafik suhu, data admin, dan data *warning* yang memudahkan nelayan dalam memonitor bagan ikan tancap secara *real time*

Penelitian [10] membahas sistem pemantauan komunikasi LoRa pada bagan ikan menggunakan aplikasi Android. Tujuan dari penelitian ini adalah untuk memudahkan nelayan mengetahui letak lokasi bagan ikan dan memudahkan mereka dalam mengelola pemantauan menggunakan aplikasi Android *My LoRa* dengan metodologi pengumpulan data berbasis *Firebase Real Time*. Penelitian ini juga membahas penggunaan Android Studio, Java, *Firebase*, *Google Map Service*, dan sistem notifikasi untuk pengembangan aplikasi. Kesimpulan dari penelitian ini adalah aplikasi yang dibangun menggunakan *database Firebase* sebagai *server*. *Firebase* tersebut dapat terhubung dengan *web* dan alat IoT, sehingga dapat mengirim dan menerima data dari *web* dan alat IoT. Android, *web*, dan alat IoT terdaftar dalam satu *Firebase* yang sama, sehingga data yang diterima di aplikasi akan menampilkan letak koordinat lokasi (*latitude* dan *longitude*) dalam bentuk teks.

Penelitian [11] membahas tentang pembuatan alat pemantauan ketinggian air laut dan kecepatan arus air laut berbasis Arduino. Metode pengukuran dalam penelitian ini berdasarkan prinsip perputaran turbin air yang dibaca oleh sensor *optocoupler*, sementara pengukuran ketinggian air laut dilakukan berdasarkan prinsip konstanta dengan pembacaan yang dilakukan sensor *ultrasonik*. Alat yang dibuat akan merekam data hasil pengukuran dari sensor ke dalam penyimpanan SD card dan akan ditampilkan pada LCD. Dari hasil pengujian, alat sensor *optocoupler* yang dibandingkan dengan alat tachometer menunjukkan persentase akurasi kesalahan sebesar 1,17%. Untuk pengujian ketinggian air laut dengan menggunakan

sensor *ultrasonik* dan penggaris, didapatkan nilai persentase rata-rata galat sebesar 1,01%.

Penelitian yang dilakukan oleh Muhammad Yakob, dkk, berjudul “Perancangan Alat Deteksi Ketinggian Air Berbasis Mikrokontroler Arduino Uno”. Penelitian ini membahas tentang pembuatan alat yang dapat digunakan sebagai alat pendeteksi ketinggian air untuk memperingatkan warga sekitar tentang perubahan ketinggian air. Pada alat pendeteksi ketinggian air berbasis mikrokontroler Arduino Uno, sensor ultrasonik berfungsi sebagai pembaca ketinggian air, kemudian diteruskan ke alat mikrokontroler Arduino Uno untuk mengolah data yang dikirimkan oleh sensor ultrasonik akan menyalakan *buzzer*. Berdasarkan data yang diperoleh dari percobaan, alat yang dirancang dapat digunakan dan alat dapat bekerja dengan normal. Jika ketinggian air kurang dari 30cm dari sensor, bel akan berbunyi, menandakan bahwa ketinggian air naik dan dapat menjadi penanda peringatan bagi warga sekitar.

Penelitian [11] membahas pembuatan alat yang dapat digunakan untuk mendeteksi ketinggian air dan memperingatkan warga sekitar tentang perubahan ketinggian air. Pada alat ini, sensor *ultrasonik* berfungsi sebagai pembaca ketinggian air, kemudian data yang dikirimkan oleh sensor tersebut diolah oleh mikrokontroler Arduino Uno, yang akan menyalakan *buzzer* jika diperlukan. Berdasarkan data dari percobaan, alat yang dirancang dapat digunakan dan bekerja dengan normal. Jika ketinggian air kurang dari 30 cm dari sensor, bel akan berbunyi, menandakan bahwa ketinggian air naik dan dapat menjadi penanda peringatan bagi warga sekitar.

Penelitian [12] membahas mengenai prototipe alat monitoring objek kendaraan dan mempermudah tim SAR dalam pencarian orang hilang saat pendakian di pegunungan. Sensor GPS yang digunakan dalam penelitian ini adalah modul GPS SX1278, dan LoRa yang digunakan adalah LoRaTM SX1278 RF *SEMTECH*. Alat ini dirancang dengan sensor GPS yang mentransmisikan data menggunakan komunikasi *Wireless Long Range* serta memanfaatkan teknologi *Internet of Things* untuk mengolah datanya. Perancangan ini menggunakan komunikasi *Long Range*, di mana dua protokol, yaitu *End-Nodes* dan *Gateway*, saling mengirimkan data lokasi latitude dan longitude objek kendaraan dengan

standar *Low-Power Wide-Area* (LPWAN). Data tersebut kemudian dikirim ke dalam *NodeMCU* dan dari *NodeMCU* dikirimkan ke *Blynk* untuk melihat data lokasi yang diterima.

Perbedaan penelitian saya dengan penelitian pertama terletak pada pengukuran dan *web dashboard* yang digunakan. Pengukuran yang dilakukan mencakup kelembapan dan suhu yang ditampilkan pada *web dashboard* LoRaID. Pada penelitian kedua, perbedaan dengan penelitian saya terletak pada *web dashboard* atau pemantauan akhir yang ditampilkan menggunakan aplikasi *My LoRa* pada *android*. Perbedaan penelitian saya dengan penelitian ketiga terletak pada sensor yang digunakan untuk mengukur arus air, serta pengumpulan data yang disimpan pada SD card dan penampilan data melalui LCD. Pada penelitian keempat, perbedaan dengan penelitian saya terletak pada sensor *buzzer* yang berbunyi ketika ketinggian air meningkat. Perbedaan dengan penelitian kelima terletak pada *web dashboard* yang digunakan, yaitu *web dashboard* *Bylink*.

2.2 Dasar Teori

Dasar teori memiliki peran yang sangat penting dalam penelitian karena memberikan landasan gagasan yang mendalam serta pemahaman menyeluruh terhadap materi penelitian. Kegunaan dasar teori dalam penelitian bertujuan sebagai panduan serta memberikan fondasi yang kokoh, merumuskan hipotesis, memilih metode, dan menguraikan hasil, sehingga meningkatkan kualitas dan makna dari penelitian secara keseluruhan. Berikut ini akan menjelaskan dasar teori yang digunakan dalam penelitian ini.

2.2.1 Bagan Ikan

Bagan ikan merupakan sebuah metode tangkap ikan tradisional yang berasal dari Indonesia yang menggunakan jaring dan lampu untuk menangkap ikan dan masih digunakan sampai saat ini oleh para nelayan di Indonesia. Bagan ikan pertama kali diperkenalkan oleh nelayan Makassar dan Bugis di Sulawesi Selatan dan Tenggara pada tahun 1950-an. Konstruksi bagan ikan terbuat dari bambu yang disusun sedemikian rupa ditanam ke dalam dasar laut di daerah perairan dangkal yang membentuk persegi untuk memudahkan pengangkatan jaring yang akan dibuat untuk menangkap ikan, jaring yang digunakan terbuat dari nilon atau polietilen agar kuat mengangkat ikan dan tahan terhadap air laut [4].

Metode tangkap ikan menggunakan bagan ikan tancap dilakukan pada malam hari atau saat kondisi gelap, lampu atau sumber cahaya dipasang di atas bagan tancap untuk menarik perhatian ikan. Ikan cenderung tertarik pada cahaya dan akan berenang mendekatinya. Setelah banyak ikan berkumpul di sekitar cahaya, jaring yang sudah dipasang di bawah bagan ikan diturunkan untuk menjebak ikan. Jaring kemudian diangkat perlahan-lahan untuk mengumpulkan ikan yang terperangkap. Metode bagan ikan tancap ini cukup efektif dan ramah lingkungan karena dapat menghindari penangkapan ikan yang berlebihan serta merusak habitat bawah laut. Bentuk fisik dari bagan ikan dapat dilihat pada Gambar 2.1.



Gambar 2.1 Bagan Ikan [13].

Dalam bagan ikan terdapat juga rumah kecil yang dibuat nelayan sebagai tempat untuk menyimpan barang maupun untuk berteduh ketika hujan untuk beristirahat sejenak bagi nelayan, adapun alat bantu lainnya yang sangat penting adalah pencahayaan bagi nelayan atau pun lampu LED, selain membantu nelayan bekerja pada malam hari lampu LED juga berfungsi untuk menarik perhatian ikan pada malam hari sehingga membawa ikan ke daerah bagan sehingga ditangkap menggunakan jaring yang diangkat. Bagan ikan merupakan tempat menangkap ikan yang secara permanen atau tidak dapat berpindah pindah sesuka hati dikarenakan konstruksi penangkapannya yang permanen dan banyaknya peralatan yang dibutuhkan untuk memindahkannya [13].

2.2.2 Arduino

Arduino merupakan *platform* perangkat keras dan perangkat lunak *open-source* yang digunakan untuk membangun berbagai proyek komputasi teknologi elektronik dan fisik. Bentuk fisik Arduino dapat dilihat pada Gambar 2.2.



Gambar 2.2 Arduino [5].

Arduino banyak digunakan dalam berbagai aplikasi mulai dari kontrol otomatis, robotika, sistem pengawasan, kontrol perangkat elektronik, seni interaktif, dan *Internet of Things* (IoT). Arduino didasarkan pada mikrokontroler yang dapat diprogram yang menyediakan berbagai pin *input/output* (I/O) yang berfungsi untuk berinteraksi dengan sensor, aktuator, dan komponen elektronik lainnya [11]. Pada dasarnya, Arduino terdiri dari papan utama yang disebut "*board*" dan lingkungan pengembangan terintegrasi *Integrated Development Environment* IDE untuk menulis dan mengunggah program ke papan arduino. Bahasa pemrograman yang digunakan di Arduino IDE adalah C/C++, meskipun ada beberapa perbedaan dalam sintaks dan fungsionalitas khusus untuk pemrograman arduino. Keunggulan Arduino selain harganya murah dan terjangkau, Arduino juga mudah digunakan dan memiliki kompatibilitas yang tinggi bahkan untuk pemula tanpa latar belakang elektronik atau pemrograman yang kuat, Arduino membuatnya mudah untuk membuat proyek elektronik interaktif, memiliki banyak *library* yang mudah dan gampang untuk dicari di *internet* untuk mempermudah dalam bereksperimen dan memiliki aplikasi khusus yang bernama Arduino IDE untuk mengunggah data program ke mikrokontroler Arduino [5].

2.2.3 Arduino IDE

Arduino IDE adalah perangkat lunak keluaran Arduino, yang digunakan untuk mengunggah program yang ditulis oleh program ke mikrokontroler. Arduino IDE sebagai mana yang ditunjukkan Gambar 2.3.



Gambar 2.3 Arduino IDE [5].

Arduino IDE dapat diintegrasikan dengan berbagai *board* Arduino antara lain Arduino Mega 2560, Arduino Uno, Arduino NANO, Arduino *micro*, Arduino MKR1000, Arduino Potenta dan masih banyak lagi. Proses penamaan program pada *board* Arduino membutuhkan *software* Arduino IDE. Penanaman program dilakukan dengan melakukan *compile* pada Arduino IDE. Arduino IDE ini juga dapat mengidentifikasi masalah pada *board* Arduino dengan menghubungkannya dengan Arduino IDE sehingga *board* yang ada dapat diperbaiki pada Arduino IDE. Terdapat 2 fungsi utama dalam pemrograman pada Arduino IDE yaitu fungsi *Void Setup()* dan fungsi *Void Loop*. *Setup* adalah fungsi yang dijalankan hanya sekali oleh Arduino IDE. Berbagai pengaturan awal dilakukan pada fungsi ini agar tidak terganggu dengan proses yang akan dijalankan oleh papan Arduino saat papan Arduino tersebut digunakan. *Loop* merupakan fungsi yang akan dijalankan oleh papan Arduino secara berulang-ulang. Fungsi utama terkait perintah-perintah yang perlu untuk dilakukan oleh papan Arduino ditanamkan pada fungsi *loop* tersebut. Fungsi *loop* sangat rentan terhadap perubahan data sehingga tidak cocok untuk melakukan pengaturan [5].

2.2.4 *Long Range Wide Area Network (LoRaWAN)*

LoRaWAN merupakan salah satu jenis jaringan telekomunikasi *nirkabel* yang dirancang untuk memungkinkan komunikasi jarak jauh dengan *bitrate* rendah. Topologi jaringan LoRaWAN dapat dilihat pada Gambar 2.4.



Gambar 2.4 Jaringan LoRaWAN [15].

Teknologi LoRaWAN berkomunikasi dengan cara nirkabel jarak jauh sehingga memiliki komunikasi yang luas. Komunikasi LoRa dapat berkomunikasi mulai dari 5 km sampai dengan 15 km tergantung dengan lingkungan yang dilalui LoRa *gateway* dalam mengirim data. LoRaWAN berkomunikasi dengan menggunakan modulasi LoRa untuk berkomunikasi dengan *gateway* LoRaWAN dengan frekuensi radio yang diizinkan bebas seperti frekuensi *Industrial Scientific, and Medical* atau disebut dengan ISM. Teknologi LoRa dirancang khusus untuk komunikasi jarak jauh dengan mengonsumsi daya rendah dan biaya yang rendah. Ada beberapa komponen utama dalam jaringan LoRaWAN [14].

Ada beberapa komponen utama dalam jaringan LoRaWAN;

- a. Perangkat LoRa atau disebut dengan LoRa *device* yang di mana merupakan perangkat IOT yang sudah terdapat di dalamnya modulasi LoRa Komunikasi nirkabel. Perangkat LoRa ini dapat berupa sensor, *actuator*, maupun perangkat yang mengumpulkan dan mengirim data.
- b. *Gateway* LoRaWAN atau sering disebut dengan jembatan penghubung antara perangkat dengan *server* jaringan LoRa.
- c. *Server* Jaringan LoRaWAN, merupakan jaringan LoRa yang bertanggung jawab atas pengelolaan serta pengendalian data jaringan.
- d. Aplikasi dan Sistem, sebagai wadah pemantauan ataupun tempat penerimaan data yang dikirimkan dari jaringan LoRaWAN atau sistem lainnya [15].

2.2.5.1 *Long Range (LoRa)*

LoRa adalah teknologi nirkabel untuk komunikasi jarak jauh dengan konsumsi daya yang rendah. Teknologi ini dirancang untuk mendukung aplikasi *Internet of Things (IoT)*, di mana banyak perangkat kecil yang terdistribusi perlu berkomunikasi jarak jauh. Salah satu bentuk dan gambar LoRa dapat dilihat pada Gambar 2.5.



Gambar 2.5 LoRa [16].

LoRa menggunakan modulasi radio yang berbeda dari teknologi nirkabel tradisional. Ini menggunakan modulasi LoRa *Chirp Spread Spectrum (CSS)* yang memungkinkan sinyal radio untuk menempuh jarak yang lebih jauh dengan daya yang lebih kecil. Pada modulasi CSS, data dikirim dalam bentuk sinyal yang frekuensinya terus menerus berubah dalam pola tertentu. Modulasi CSS ini membuat sinyal LoRa memiliki sensitivitas penerimaan yang tinggi, sehingga sinyal dapat dideteksi dan diterima pada jarak yang lebih jauh [16].

Keunggulan menggunakan LoRa adalah dari segi jarak jangkauan komunikasi yang luas, yang dapat mencapai beberapa kilometer tergantung kondisi lingkungan. Selain itu, LoRa juga memberikan konsumsi daya yang rendah, sehingga perangkat LoRa dapat berjalan lama dengan ukuran baterai yang relatif kecil. Teknologi LoRa digunakan dalam berbagai aplikasi IoT seperti pemantauan lingkungan, pemanfaatan jaringan sensor, pengukuran jarak jauh, dan banyak lagi yang berbaur dengan teknologi. LoRa umumnya digunakan dalam jaringan LoRaWAN berbasis infrastruktur, dimana perangkat LoRa berkomunikasi dengan *gateway* LoRaWAN untuk mengirim dan menerima data [12].

2.2.4.2 Gateway LoRaWAN

LoRa *gateway* merupakan perangkat keras atau perangkat fisik berada diantara *node* akhir dan *server* jaringan yang berfungsi menghubungkan perangkat LoRaWAN. Salah satu LoRa *gateway* dapat dilihat pada Gambar 2.6.



Gambar 2.6 Gateway LoRa [17].

Gateway bekerja untuk menerima data dari perangkat LoRaWAN dan mengirimkan data tersebut ke *server* aplikasi. *Gateway* dapat juga mengirimkan perintah ke perangkat LoRaWAN untuk mengontrol alat secara jarak jauh. Penggunaan *gateway* LoRa sangat penting dalam jaringan LoRa dikarenakan LoRa memiliki batasan dalam mengirimkan data secara langsung ke *server* jaringan. *Gateway* LoRa membantu menjembatani transmisi data dari *node* sensor ke *server* jaringan. Selain itu, penempatan letak *gateway* penting untuk mengoptimalkan kinerja jaringan LoRa. Jumlah *gateway* yang diperlukan bergantung pada jangkauan jarak dan sensitivitas penerima perangkat. Penempatan *gateway* dapat dioptimalkan dengan memperhatikan sekitar dan menempatkan di atas yang minim dari *obstacle* atau penghalang agar jaringan LoRaWAN dapat bekerja maksimal [17].

2.2.4.3 Server Jaringan LoRaWAN

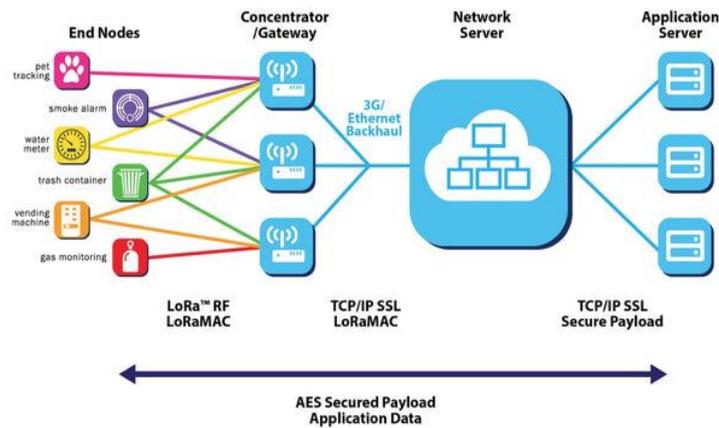
Server Jaringan LoRaWAN atau LoRaWAN *Network Server* merupakan perangkat lunak yang digunakan untuk mengelola jaringan perangkat *Low-Power Wide Area Network* (LPWAN) dengan menggunakan *protocol* LoRaWAN. *Server* jaringan LoRaWAN berfungsi juga sebagai platform komprehensif untuk transfer data antar perangkat dan aplikasi, manajemen

perangkat, manajemen *gateway*, konfigurasi perangkat, manajemen akses pengguna. Selain itu, *server* jaringan LoRaWAN dapat secara otomatis memantau *gateway*, integrasi, dan perangkat agar berfungsi dengan baik dan mengirimkan peringatan status kepada pengguna. Perangkat lunak *Server Jaringan LoRaWAN* dapat berupa *web dashboard* maupun berupa aplikasi sebagai penerima dan wadah untuk memantau alat dengan teknologi LoRa, seperti *The Things Network (TTN)*, *ThingSpeak*, *ChirpStack*, *Actility ThingPark*, *Senet*, *Helium* dan masih banyak *platform* lainnya [18].

2.2.3.4 Komunikasi LoRa

Komunikasi LoRa adalah teknologi komunikasi *nirkabel* yang dirancang khusus untuk mentransmisikan data dalam jangkauan jarak yang jauh dengan konsumsi daya yang rendah. LoRa menggunakan modulasi CSS untuk mengirimkan sinyal yang dapat menembus hambatan dan memiliki sensitivitas tinggi. Komunikasi LoRa dapat beroperasi dalam berbagai skenario, termasuk *point to point (P2P)* dan *multipoint to point*. Komunikasi LoRa *point to point* adalah teknologi nirkabel yang memungkinkan transmisi data jarak jauh antara dua perangkat. Dalam skenario *point to point*, satu perangkat bertindak sebagai pengirim data atau *End Nodes*, sementara perangkat lainnya berperan sebagai penerima atau *Gateway* [19].

Komunikasi LoRa *multipoint to point* melibatkan penggunaan teknologi *nirkabel* yang memungkinkan banyak perangkat berkomunikasi dalam jaringan yang terdistribusi. Dalam skenario *multipoint to point*, beberapa perangkat dapat berfungsi sebagai pengirim dan penerima dalam menangani banyak perangkat dalam satu jaringan yang saling terhubung. Di bawah ini merupakan gambar Komunikasi LoRa. Gambar 2.7 adalah gambar skenario komunikasi *multipoint to point*, tetapi untuk melihat gambar skenario komunikasi loRa *point to point* dapat juga melihat Gambar 2.7, namun cukup memperhatikan 1 *End Nodes* dan 1 *Gateway* [20].



Gambar 2.7 Komunikasi LoRa [20].

2.2.5 Sensor

Sensor dalam penelitian memiliki fungsi penting untuk mengumpulkan data yang akurat dan relevan. Sensor-sensor ini mendeteksi dan mengukur suatu keadaan tertentu dalam lingkungan atau objek yang di teliti, berikut merupakan sensor-sensor yang digunakan dalam penelitian ini.

2.2.5.1 Sensor *Global Positioning System* (GPS)

GPS adalah alat sistem *navigasi* yang berbasis satelit yang digunakan untuk menginformasikan titik lokasi penggunanya di permukaan bumi. *Navigasi* satelit GPS juga dapat menentukan ketepatan, arah dan juga estimasi waktu yang dibutuhkan seperti hal dalam *google maps*. Awalnya GPS digunakan untuk pertahanan militer Amerika Serikat dengan mengorbitkan 24 satelit oleh departemen Pertahanan Amerika Serikat. Pada tahun 1980-an pemerintah membuat sistem yang ada untuk penggunaan sipil sehingga sampai saat ini GPS terus berkembang dan bekerja di kondisi cuaca bagaimana saja di dunia ini dan bekerja 24 jam sehari.

Ada 3 komponen utama dalam GPS yang memaksimalkan kerja GPS

1. Satelit, satelit di mana berfungsi untuk menerima dan menyimpan data yang dikirimkan oleh stasiun pengontrol di bumi. Kemudian menjaga dan mengirimkan informasi data secara kontinu ke *receiver* atau pengguna.
2. Pengontrol, Pengontrol berfungsi untuk mengendalikan dan mengontrol satelit dari bumi baik untuk mengecek kesehatan satelit; penentuan dan prediksi orbit waktu, sinkronisasi waktu antar satelit, dan mengirim data ke satelit.

3. Penerima (*receiver*). Penerima berfungsi menerima data yang dikirimkan dari satelit dan kemudian diproses untuk menentukan posisi, arah, jarak dan waktu yang diperlukan oleh pengguna [21].

Penerima GPS mendapatkan sinyal dari beberapa satelit yang mengorbit bumi. Satelit yang mengorbit bumi dalam orbit pendek ini terdiri dari 24 susunan satelit, dengan 21 satelit aktif dan tiga cadangan. Melalui susunan orbit tertentu, seluruh permukaan bumi dapat menerima satelit GPS, dan kemunculan 4 hingga 8 satelit GPS dapat memberikan informasi posisi dengan presisi tinggi. Selama proses ini, GPS menggunakan alat *navigasi* untuk melakukan proses penandaan lokasi, yang bergantung pada keakuratannya. Keakuratan atau presisi diperlukan saat menentukan koordinat lokasi. Koordinat lokasi ini selalu memiliki faktor kesalahan, juga dikenal sebagai tingkat presisi, semakin banyak satelit yang kita dapatkan, semakin akurat penentuan posisi kita [12].



Gambar 2.8 GPS NEO6MV2 [22].

Gambar 2.8 merupakan salah satu bentuk dan gambaran dari GPS dengan versi GPS NEO6MV2. Komunikasi antara Arduino dan modul GPS menggunakan komunikasi serial *Universal Asynchronous Receiver/Transmitter* (UART). UART merupakan perangkat keras yang digunakan untuk menerima dan mengirim data secara serial antara dua perangkat. UART terdiri dari dua pin: pin Rx dan Tx. Pin Rx digunakan untuk menerima data. Pin Tx digunakan untuk mengirimkan data. Saat menghubungkan dua perangkat menggunakan UART, pin Rx dari satu perangkat terhubung ke pin Tx dari perangkat kedua [22].

2.2.5.2 Sensor Jarak (*Sharp IR*)

Sensor jarak merupakan sebuah sensor yang mampu mendeteksi keberadaan suatu benda yang berada di dekat sensor tanpa harus kontak fisik.

Sensor jarak umumnya digunakan dalam berbagai aplikasi mulai dari kendaraan otonom dan robotika hingga sistem keamanan dan pengukuran industri. Bentuk sensor ini dapat dilihat pada Gambar 2.9.



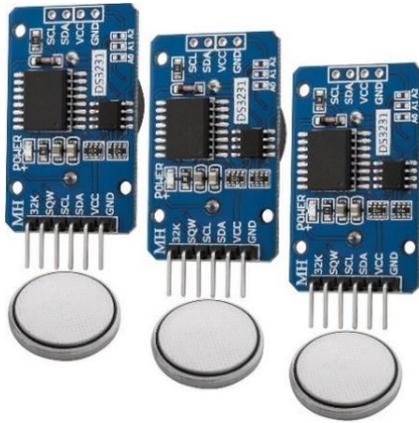
Gambar 2.9 Sensor *Sharp* IR [8].

Prinsip dasar di balik sensor jarak adalah pengukuran waktu perjalanan atau properti fisik lainnya untuk menentukan jarak antara sensor dan objek. Terdapat banyak jenis teknologi pengukuran pada setiap sensor jarak, di bawah ini merupakan salah satu teknologi sensor jarak yang mengukur jarak menggunakan sinar inframerah. Sensor jarak inframerah bekerja dengan cara memancarkan sinar inframerah ke objek yang ingin diukur jaraknya, kemudian mendeteksi pantulan sinar tersebut. Sinar inframerah yang dipantulkan dari objek diukur oleh sensor, dan berdasarkan waktu tempuh atau intensitas pantulan, sensor dapat menghitung jarak antara sensor dan objek tersebut. Informasi jarak ini kemudian dikonversi ke dalam bentuk yang dapat digunakan, seperti data digital atau analog. Sensor jarak yang digunakan pada penelitian ini merupakan sensor jarak IR *Sharp* yang dimana sensor ini bekerja dengan menggunakan teknologi inframerah [8].

2.2.5.3 Sensor *Real time Clock* (RTC)

RTC merupakan jenis sensor perangkat yang dirancang untuk memberikan informasi waktu dengan akurat dan berkelanjutan. Inti dari perangkat RTC adalah osilator kristal yang menghasilkan osilasi sangat stabil yang menjadi dasar pengukuran waktu pada perangkat RTC. RTC dilengkapi dengan baterai cadangan, berfungsi menyuplai daya agar dapat terus menyimpan dan mempertahankan waktu ketika daya utama dimatikan, sehingga optimal untuk

sistem pengaturan waktu yang membutuhkan akurasi tinggi. RTC biasanya digunakan dalam sistem mikrokontroler dan perangkat tertanam untuk melacak waktu dan menjadwalkan tugas tertentu pada rangkaian perangkat. Bentuk fisik RTC dapat dilihat pada Gambar 2.1.



Gambar 2.10 Sensor RTC [23].

RTC memberikan kemampuan untuk melakukan sinkronisasi dengan waktu standar dunia, seperti Waktu UTC, dan diterapkan di komputer, *server*, dan bahkan perangkat konsumen seperti jam tangan pintar dan kamera digital. RTC dapat diakses dan dikonfigurasi melalui antarmuka seperti I2C dan SPI, memungkinkan integrasi yang fleksibel ke dalam berbagai sistem. RTC memiliki fitur seperti kalibrasi dan kemampuan program yang membantu menjaga ketepatan waktu yang penting dalam berbagai aplikasi [23].

2.2.6 Mikrokontroler Esp-32

Esp-32 merupakan sebuah mikrokontroler yang memiliki teknologi baru yang merupakan *upgrade* dari Esp-8266 yang di perkenalkan oleh *Espressif System*. Bentuk dari Esp-32 dapat dilihat *sebagaimana* pada Gambar 2.11 Esp-32



Gambar 2.11 Esp-32 [24].

Esp-32 memiliki kelebihan yang mupun yang bisa menjadi solusi untuk menghubungkan projek yang di rancang ke dalam jaringan *internet*, selain memiliki daya sistem yang rendah Esp-32 juga memiliki kemampuan mode *Bluetooth* ganda dan lebih *fleksibel* dikarenakan hemat daya. Esp-32 cocok digunakan untuk pengaplikasian *Internet Of Things* ternyata sebagai pilihan yang dapat diandalkan di lingkungan industri karena rentang suhu operasi yang luas. Esp-32 dapat bertindak secara mandiri yang lengkap dan bisa juga bertindak sebagai perangkat pendukung dikarenakan pada *chip mikrokontroler* sudah terintegrasi dengan *Wi-Fi*. [24].

2.2.7 Teori Parameter

Dalam dunia Komunikasi terdapat teori yang membahas mengenai parameter sebagai acuan kategori kualitas sinyal dari sebuah teknologi komunikasi, yang di mana memiliki perbedaan dan persamaan dalam parameter kualitas sinyal setiap teknologi komunikasi. Berikut ini akan menjelaskan beberapa parameter yang digunakan dalam mengukur kualitas sinyal maupun hanya untuk mengetahui nilai dari parameter tersebut.

2.2.7.1 Received Signal Strength Indicator (RSSI)

RSSI adalah *parameter* untuk mengukur indikator kekuatan sinyal yang diterima sebuah perangkat *wireless*. Nilai RSSI sangat bergantung pada kondisi lingkungan yaitu jarak dan penghalang. Pengukuran dilakukan berdasarkan *signal strength* atau kekuatan sinyal yang di terima. Hal ini bertujuan untuk mengetahui kuat sinyal dari suatu komunikasi *wireless*, semakin jauh dan semakin banyak penghalangnya maka nilai RSSI tak terkecuali dalam komunikasi kinerja dari LoRa. Dalam *range* kulitas sinyal semakin besar nilai RSSI, maka semakin bagus sinyal tersebut, namun jika nilai RSSI semakin kecil maka kualitas sinyal dinyatakan semakin buruk. *Range parameter* RSSI dapat dilihat pada Tabel 2.1 *Range Parameter* RSSI di bawah ini, untuk mengetahui nilai rata-rata RSSI sebagaimana dapat menggunakan *formula* yang ditunjukkan pada persamaan (2.1) [25].

$$\text{RSSI (dBm)} = \frac{\text{Total Nilai RSSI yang dipenerima}}{\text{Jumlah Sampel RSSI yang dikirimkan}} \quad (2.1)$$

Tabel 2.1 Range Parameter RSSI [26].

Kategori	RSSI (dBm)
-30 to -60	Sangat Kuat: Jarak pemancar dan penerima sangat dekat
-60 to -90	Sangat Baik: Cukup dekat
-90 to -105	Baik: Beberapa data tidak diterima
-105 to -115	Buruk: Bisa menerima tapi sering putus
-115 to -120	Sangat Buruk: Data sinyal lemah sering hilang

2.2.7.2 *Signal-to-Noise Ratio (SNR)*

SNR adalah rasio antara kekuatan sinyal yang diinginkan (*signal*) dengan kekuatan sinyal yang tidak diinginkan (*noise*) dalam suatu sistem komunikasi. Dalam konteks komunikasi LoRa, SNR sangat penting karena LoRa dirancang untuk beroperasi di lingkungan yang penuh dengan *noise* dan interferensi. Lantai kebisingan (*noise floor*) merupakan area dimana semua sumber sinyal interferensi yang tidak diinginkan dapat merusak sinyal yang ditransmisikan dan oleh karena itu transmisi ulang akan terjadi. Biasanya tingkat kebisingan adalah batas sensitivitas fisik, namun LoRa bekerja di bawah tingkat kebisingan.

- a. Nilai SNR LoRa yang umum adalah antara: -20dB dan +10dB
- b. Nilai yang mendekati +10dB berarti sinyal yang diterima bagus.
- c. LoRa dapat mendemodulasi sinyal yang berada pada -7,5 dB hingga -20 dB di bawah dasar kebisingan.

Secara sederhana, semakin tinggi SNR, semakin baik kualitas sinyal yang diterima [27].

Berikut ini merupakan rumus untuk menentukan SNR [28]:

$$\text{SNR(dB)} = 10 \log_{10} \left(\frac{S}{N} \right) \quad (2.2)$$

Keterangan :

s : *Signal (Watt)*

n : *Noise (Watt)*

2.2.7.3 *Time on Air (ToA)*

TOA sangat penting dalam jaringan LoRa karena mempengaruhi sejumlah faktor, termasuk konsumsi daya, kapasitas jaringan, dan efisiensi penggunaan saluran frekuensi. ToA pada teknologi jaringan LoRa merupakan *parameter* waktu yang mengacu pada waktu udara yang diperlukan untuk mengirimkan paket data dari pemancar *node* pengirim atau *transmitter* ke *node* penerima atau *receiver* dalam saluran jaringan komunikasi *nirkabel* LoRa. *Formula* yang digunakan untuk menentukan nilai *Time on Air* dapat dilihat dari persamaan (2.3), dan untuk penyebaran *Time on Air* dalam setiap *spreading factor* pada komunikasi LoRa dapat dilihat pada Tabel 2.2 *Time on Air Spreading Factor*.

$$\text{ToA(s)} = \text{Waktu Pengirim Transmitter} - \text{Waktu Penerima Receiver} \quad (2.3)$$

Tabel 2.2 *Time on Air Spreading Factor* [29].

<i>Spreading Factor (SF)</i>	<i>Time on Air (ms)</i>
7	41
8	72
9	144
10	289
11	578
12	991

2.2.5.4 *Delay*

Delay adalah waktu yang dibutuhkan suatu paket untuk melakukan perjalanan dari awal pengirim ke akhir penerima. Ini juga dapat dinyatakan sebagai waktu tunda yang disebabkan oleh pengiriman paket data dari satu titik ke titik lain (tujuannya). *Delay* dapat dipengaruhi oleh jarak, media fisik, kemacetan pengiriman antrian data, atau waktu proses pengolahan data yang lama. *Delay* yang dihitung pada penelitian ini merupakan *delay* lama pengolahan data dari awal hingga akhir penerimaan. Nilai *delay* dapat dihitung menggunakan formula persamaan (2.4) dan nilai rata-rata *delay* ditunjukkan pada persamaan (2.5) [30]:

$$Delay (s) = \text{Waktu pengiriman awal} - \text{Waktu penerimaan akhir} \quad (2.4)$$

$$\text{Rata - rata Delay}(s) = \frac{\text{Total Delay}}{\text{Total Paket yang diterima}} \quad (2.5)$$

2.2.8 *ThingSpeak*

ThingSpeak adalah *platform* berbasis *cloud* untuk pengumpulan, analisis, dan visualisasi data *Internet of Things* (IoT). Layanan *ThingSpeak* berisi layanan aplikasi *open-source* dan API untuk mengumpulkan menyimpan dan menganalisis data dari beberapa perangkat IOT melalui *internet* atau *Local Area Network* (LAN) menggunakan *Hypertext Transfer Protocol* (HTTP). *Web dashboard ThingSpeak* memberikan kemampuan untuk membuat grafik, *widget*, dan *gauge* yang mempermudah pemahaman data secara visual, selain itu dalam *web dashboard ThingSpeak* dapat dibuat saluran data (*channel*) yang dimana informasi dari berbagai perangkat yang dapat dikumpulkan dan divisualisasikan dalam satu akun *ThingSpeak*. Dengan aksesibilitas melalui *web browser* di komputer atau perangkat *seluler*, *ThingSpeak* sangat memudahkan pengguna untuk memantau dan mengelola data perangkat dari mana saja [31].

2.2.9 Redaman Air Laut

Redaman sinyal radio oleh air laut adalah fenomena di mana sinyal elektromagnetik yang dipancarkan melalui air laut mengalami penurunan intensitas atau kekuatan karena berbagai faktor yang terjadi saat sinyal tersebut merambat melalui medium air laut. Berikut adalah beberapa faktor yang menyebabkan redaman sinyal radio oleh air laut: Penyerapan: Air laut memiliki sifat dielektrik yang menyebabkan penyerapan energi dari gelombang elektromagnetik yang merambat melalui air. Molekul air menyerap sebagian energi dari gelombang radio, terutama pada frekuensi tertentu, yang mengakibatkan penurunan kekuatan sinyal. Penyebaran: Ketika sinyal radio merambat melalui air laut, sebagian energinya akan diserap oleh partikel-partikel di dalam air dan kemudian disebarkan ke segala arah. Proses penyebaran ini menyebabkan redaman sinyal, terutama pada frekuensi yang lebih tinggi. Berikut ini beberapa Faktor yang mempengaruhi redaman pada permukaan air laut :

- a. Hambatan fisik: Selain penyerapan dan penyebaran, faktor-faktor fisik seperti garam, sedimen, dan organisme laut juga dapat mempengaruhi redaman sinyal radio. Misalnya, kandungan garam yang tinggi dalam air laut dapat meningkatkan konduktivitasnya, yang pada saat perambatan dapat menghasilkan hambatan tambahan terhadap sinyal radio.
- b. Kondisi Permukaan Air: Permukaan air yang kasar atau gelombang yang tinggi dapat meningkatkan redaman karena gelombang elektromagnetik dipantulkan atau dihamburkan oleh permukaan air yang tidak rata.
- c. Jarak: Semakin jauh jarak tempuh sinyal melalui air laut, semakin besar redaman yang akan terjadi. Hal ini disebabkan oleh peningkatan jumlah molekul air yang dilewati sinyal dan peningkatan peluang interaksi sinyal dengan partikel-partikel di dalam air.
- d. Kedalaman: Kedalaman air juga dapat memengaruhi redaman sinyal radio. Pada umumnya, semakin dalam air, semakin sedikit gangguan yang disebabkan oleh permukaan air dan gangguan lainnya, sehingga redaman cenderung lebih rendah pada kedalaman yang lebih dalam. Dengan memahami faktor-faktor ini, para insinyur dan ilmuwan dapat mengembangkan model matematika dan simulasi komputer untuk memprediksi redaman sinyal radio yang mungkin terjadi dalam berbagai kondisi laut. Hal ini penting dalam merancang sistem komunikasi bawah air, navigasi kapal, dan aplikasi lain yang melibatkan transmisi sinyal radio di lingkungan laut [32].