

BAB II

DASAR TEORI

2.1 Tinjauan Pustaka

Penelitian terhadap pencegahan pelanggaran batas maritim pernah dilakukan oleh I Made Andi Arsana dari Universitas Gajah Mada. Penelitian tersebut merupakan salah satu contoh implementasi alat yang memanfaatkan teknologi sederhana berbasis arduino uno dan sensor GPS untuk mencegah adanya pelanggaran batas maritim oleh nelayan tradisional di daerah pesisir yang dekat dengan perbatasan maritim. Namun, sistem ini memiliki kekurangan karena belum adanya metode pemantauan sehingga belum cukup relevan jika ingin di implementasikan sebagai sebuah sistem peringatan dan pemantauan. [4]

Pada penelitian lain, Ria Juliani Dewi, Angga Rusdinar, dan Ig. Prasetya Dwi Wibawa berhasil mengembangkan sistem *monitoring* posisi perahu nelayan dengan menggunakan GPS dan formula *Geofencing haversine* untuk mengetahui letak posisi nelayan ketika berada di laut. Sistem ini juga menghitung jarak antara titik koordinat yang dibaca dengan garis pembatas yang dibuat, dan memberikan peringatan dengan menyalakan *buzzer* jika jarak tersebut lebih dari 100 meter. Tingkat akurasi data koordinat yang dibaca memiliki range jarak sejauh 20 meter dengan rata-rata akurasi sebesar 91,01% [6]

Selain itu, Ramadhan et al. memanfaatkan LoRa (*Long Range*) sebagai media komunikasi data dalam sistem *monitoring* lingkungan di area tambak. Teknologi LoRa dikenal dengan kemampuan jangkauan sinyal yang luas dan efisiensi konsumsi daya yang rendah, sehingga sangat cocok diterapkan di area terbuka seperti tambak atau wilayah pesisir. Meskipun begitu, penelitian ini lebih berfokus pada pemantauan parameter lingkungan dan belum mengoptimalkan sistem alarm yang berorientasi pada keselamatan pengguna.[7]

Selanjutnya, studi yang dilakukan oleh Putra dan timnya mengkaji penggunaan LoRa untuk pelacakan objek bergerak seperti alat berat di area pertambangan. Sistem ini berhasil memanfaatkan keunggulan LoRa dalam pengiriman data posisi, namun keterbatasan terletak pada sistem pengolahan data yang masih bersifat lokal, serta tidak adanya integrasi dengan platform visualisasi data seperti peta digital

interaktif ataupun sistem notifikasi berbasis protokol komunikasi modern seperti MQTT.[8]

Berdasarkan kelemahan dari beberapa penelitian terdahulu, dalam penelitian ini peneliti merancang sebuah rancang bangun sistem alarm berbasis GPS dengan kombinasi komunikasi LoRa dan protokol MQTT yang ditujukan untuk mendukung nelayan dalam mencegah pelanggaran batas wilayah maritim Indonesia. Sistem yang kami kembangkan dilengkapi dengan fitur peringatan otomatis jika kapal melebihi batas wilayah yang telah ditentukan. Dengan pemanfaatan LoRa, sistem mampu menjaga efisiensi konsumsi daya dan cakupan jangkauan sinyal yang luas, sementara integrasi MQTT mendukung komunikasi data yang lebih ringan dan stabil, bahkan dalam kondisi koneksi yang minim *bandwidth*. Kehadiran sistem ini diharapkan dapat membantu nelayan tradisional dalam meningkatkan keselamatan pelayaran serta mengurangi potensi pelanggaran batas maritim secara tidak disengaja.[7]

2.2 Pelanggaran Batas Maritim Indonesia

Batas maritim adalah garis imajiner yang menentukan wilayah perairan suatu negara berdasarkan hukum internasional. Batas ini mencakup perairan yang menjadi hak kedaulatan negara dan wilayah di mana negara memiliki hak ekonomi eksklusif. Penetapan batas maritim bertujuan untuk mengatur hak, yurisdiksi, dan kewajiban negara dalam mengelola sumber daya laut serta menjaga keamanan wilayah perairan. [9]

Penentuan batas maritim suatu negara diatur dalam Konvensi Perserikatan Bangsa-Bangsa tentang Hukum Laut (*UNCLOS*) 1982, yang membagi wilayah laut menjadi beberapa zona diantaranya zona Laut Teritorial yang berjarak 12 mil laut (22,5 km) dari garis pangkal pantai. Dalam zona ini, negara memiliki hak penuh untuk menerapkan hukum, mengatur sumber daya dan menjaga keamanan. Selanjutnya merupakan Zona Tambahan yang berada setelah laut teritorial hingga 24 mil laut atau sekitar 44,4 km dari garis pangkal pantai. Negara memiliki hak untuk mencegah dan menindak pelanggaran hukum terkait kepabeanaan, perpajakan, imigrasi, dan kesehatan di zona ini. Lalu ada Zona Ekonomi Eksklusif (ZEE) yang

membentang 200 mil laut (370,4 km) dari garis pangkal dan merupakan wilayah di mana negara memiliki hak eksklusif untuk mengeksplorasi serta mengeksploitasi sumber daya alam, baik di perairan maupun dasar laut.[10] Negara lain dapat melakukan pelayaran dan penerbangan bebas, tetapi tidak boleh mengeksploitasi sumber daya alam tanpa izin negara pemilik ZEE. Terakhir yaitu Landas Kontinen yang Merupakan dasar laut yang secara geologi merupakan kelanjutan dari daratan suatu negara hingga 350 mil laut dari garis pangkal. Negara berhak mengeksplorasi sumber daya alam non-hayati seperti minyak dan gas bumi di zona ini.[11]



Gambar 2. 1 Peta Batas Maritim Indonesia [12]

Sebagai negara kepulauan, Indonesia memiliki batas maritim yang kompleks karena berbatasan dengan banyak negara seperti Malaysia, Singapura, Filipina, Vietnam, Thailand, India, Timor Leste, Papua Nugini, dan Australia. Pada penelitian kali ini, peneliti membuat alat bantu untuk mencegah pelanggaran batas maritim yang letak nya cukup strategis dan berbatasan langsung dengan beberapa negara tetangga diantaranya dengan Malaysia di Selat Malaka dan dengan Singapura di Selat Singapura. Pelanggaran batas maritim sering terjadi di Indonesia, baik oleh kapal nelayan asing yang melakukan *illegal fishing* maupun oleh kapal-kapal yang melintasi perairan tanpa izin. Tetapi, terkadang banyak juga nelayan tradisional yang masih tidak patuh dan memahami secara spesifik tentang perbatasan teritorial tersebut. Bahkan nelayan kerap kali berurusan dengan hukum

negara lain karena telah melanggar perbatasan laut teritorial yang melibatkan Indonesia dan negara tetangga. Dengan luasnya wilayah maritim Indonesia dan seringnya terjadi pelanggaran batas, diperlukan sistem pemantauan berbasis teknologi yang efektif. Sistem *GPS-based alarm* dan LoRa yang dikembangkan dalam penelitian ini dapat membantu mencegah nelayan Indonesia melanggar batas maritim dengan memberikan peringatan dini sebelum melewati batas yang telah ditentukan. Selain itu, penggunaan LoRa dengan *repeater* dapat memperluas jangkauan komunikasi untuk memastikan nelayan tetap terhubung dengan pusat pemantauan di darat. [13]

2.3 Nelayan

Nelayan adalah individu atau kelompok yang melakukan kegiatan penangkapan ikan dan sumber daya laut lainnya sebagai mata pencaharian utama maupun tambahan. Menurut Undang-Undang Republik Indonesia Nomor 45 Tahun 2009 tentang Perubahan atas Undang-Undang Nomor 31 Tahun 2004 tentang Perikanan, nelayan adalah orang yang mata pencahariannya melakukan penangkapan ikan. Berdasarkan skala usaha dan alat tangkap yang digunakan, nelayan dapat dikategorikan menjadi Nelayan Tradisional, Nelayan Skala Kecil, dan Nelayan Industri atau Modern. Pada penelitian kali ini penulis berfokus terhadap Nelayan Tradisional. [14]

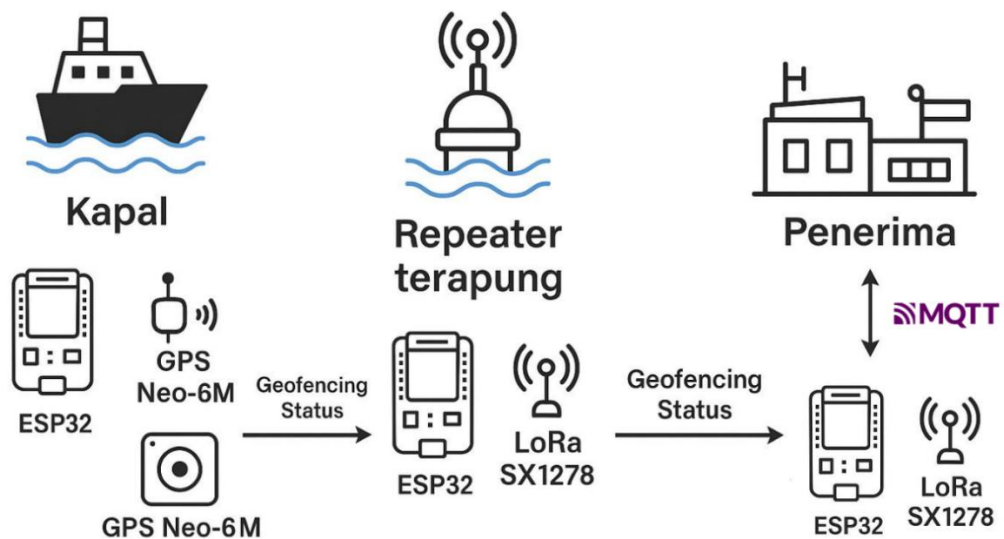
Nelayan tradisional adalah kelompok nelayan yang menggunakan alat tangkap sederhana seperti jaring, pancing, dan perangkap dalam mencari ikan. Mereka umumnya beroperasi di perairan yang relatif dekat dari pantai dan bergantung pada pengetahuan turun-temurun dalam navigasi serta teknik penangkapan ikan. Nelayan tradisional memiliki peran yang signifikan dalam sektor perikanan, terutama di daerah pesisir. Beberapa kontribusi mereka meliputi penyediaan sumber protein lokal, menjaga keberlanjutan perikanan, serta menyokong ekonomi lokal. [13]



Gambar 2. 2 Nelayan Tradisional [15]

2.4 Komponen Utama (*Transmitter, Repeater, Receiver*)

Dalam metode komunikasi *peer to peer*, sistem ini dibagi menjadi 3 Komponen utama yang menunjang keberjalanan sistem antara lain *Transmitter*, *Repeater*, dan juga *Receiver*. Penulis menggunakan metode komunikasi seperti ini dengan harapan dapat memperluas jarak jangkauan komunikasi dan juga pengiriman data.



Gambar 2. 3 Ilustrasi Komponen Utama

2.4.1 *Transmitter (GPS-Based Alarm)*

Pada kasus kali ini yang bertugas sebagai *transmitter* ataupun *nodes* dari sistem tersebut adalah *GPS-Based Alarm*. Disebut sebagai *transmitter* karena alat *GPS-Based alarm* tersebut akan menginisiasikan metode *Geofencing* dengan formula *line crossing detection* untuk melakukan perbandingan data lokasi yang telah di tentukan dengan data lokasi dari modul GPS. Pada saat lokasi *real-time* melewati atau mendekati batas yang telah di tentukan, maka *GPS-Based alarm* tersebut akan mengeluarkan *output* berupa *buzzer* serta mengirimkan data lokasi secara periodik menuju *Repeater*.

2.4.2 *Repeater*

Komponen selanjutnya yaitu *repeater* yang merupakan penerima sinyal dari *transmitter*. Komponen ini nantinya akan menjadi jembatan komunikasi antara *Transmitter* dan *Receiver*. Komponen ini baiknya di implementasikan di tengah lautan agar berada di antara kedua komponen yang lain nya. Sinyal yang diterima akan dikirimkan kembali menuju *Receiver* yang ada di pesisir pantai. Implementasi *Repeater* merupakan salah satu solusi yang dapat dilakukan untuk memperluas jarak jangkauan dari komunikasi LoRa tersebut.

2.4.3 *Receiver (Gateway)*

Sesuai dengan namanya, *receiver* merupakan komponen penerima sinyal yang dikirimkan oleh *transmitter* dan diteruskan oleh *repeater*. Komponen ini akan menerima sinyal radio dari *Repeater*, lalu data tersebut akan di upload menuju *broker* menggunakan protokol MQTT sebagai data lokasi yang akan di simpan ke dalam *database* sebagai sumber data dari *dashboard* pemantauan lokal.

2.5 *GPS-Based Alarm*

GPS-based alarm adalah sistem peringatan yang memanfaatkan teknologi *Global Positioning System* (GPS) untuk melacak lokasi suatu objek secara *real-time* dan memberikan notifikasi jika objek tersebut mendekati atau melewati batas yang telah ditentukan. Sistem ini sering digunakan dalam berbagai aplikasi, termasuk pemantauan kendaraan, keamanan wilayah, dan dalam konteks penelitian

ini, sebagai alat bantu bagi nelayan agar tidak melewati batas maritim Indonesia. Teknologi ini bekerja dengan menerima sinyal dari minimal empat satelit GPS untuk menentukan koordinat lokasi menggunakan metode triangulasi. Setelah koordinat diperoleh, sistem akan membandingkannya dengan batas wilayah yang telah ditetapkan sebelumnya (*geofence*) untuk menentukan apakah kapal berada dalam zona aman atau berisiko melanggar batas maritim. Dalam sistem ini, tipe data yang digunakan untuk membandingkan lokasi *real-time* dengan batas wilayah berbentuk koordinat geospasial seperti *latitude* (garis lintang) dan *longitude* (garis bujur) yang dinyatakan dalam format desimal (misalnya, -6.2088, 106.8456 untuk lokasi Jakarta). [5]

Data batas wilayah disimpan dalam bentuk *array* atau *database* geospasial yang berisi daftar koordinat yang membentuk area tertentu. Perbandingan antara lokasi kapal dengan batas wilayah dilakukan menggunakan metode matematika geospasial, seperti perhitungan jarak *Haversine Formula* atau metode *Geofencing* dengan *Line crossing detection*, di mana sistem menentukan apakah titik koordinat kapal berada di dalam atau di luar garis batas maritim yang telah ditetapkan. Ketika sistem mendeteksi bahwa kapal mendekati atau melewati batas, *GPS-based alarm* akan mengaktifkan peringatan dalam berbagai bentuk alarm suara, lampu indikator, atau notifikasi digital pada layar perangkat yang digunakan nelayan. Selain memberikan peringatan lokal, sistem juga dapat mengirimkan data lokasi secara berkala ke pusat pemantauan menggunakan komunikasi LoRa.

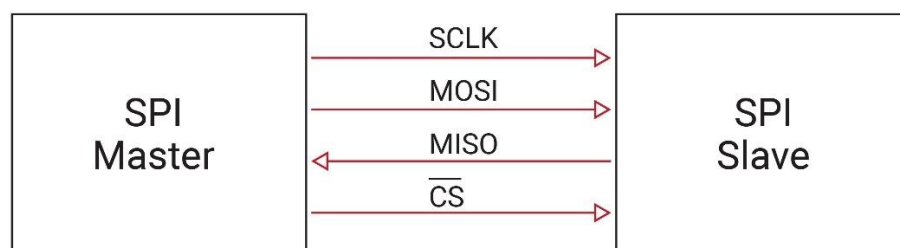
Dengan implementasi sistem ini, nelayan dapat lebih sadar akan batas wilayah yang tidak boleh mereka langgar, sehingga mengurangi risiko penangkapan oleh otoritas asing akibat pelanggaran batas maritim yang tidak disengaja. Di sisi lain, pihak berwenang dapat lebih mudah memantau pergerakan kapal dan segera menindaklanjuti jika ada aktivitas yang mencurigakan. Hal ini menjadikan *GPS-based alarm* sebagai solusi efektif dalam menjaga keamanan maritim dan meningkatkan kepatuhan nelayan terhadap batas wilayah perairan Indonesia.

2.6 Protokol Komunikasi

Protokol komunikasi merupakan kumpulan aturan yang digunakan untuk mengatur pertukaran data antar perangkat dalam sistem digital, sehingga data dapat dikirim dan diterima secara akurat dan efisien. Dalam sistem embedded atau IoT, protokol komunikasi berperan penting untuk memastikan integrasi antar komponen berjalan dengan baik. Beberapa protokol komunikasi yang umum digunakan meliputi SPI (*Serial Peripheral Interface*), UART (*Universal Asynchronous Receiver/Transmitter*), LoRa (*Long Range*), dan MQTT (*Message Queuing Telemetry Transport*).

2.6.1 SPI (*Serial Peripheral Interface*)

SPI (*Serial Peripheral Interface*) merupakan salah satu protokol komunikasi serial sinkron yang kerap digunakan dalam sistem embedded untuk pertukaran data antara mikrokontroler sebagai master dan berbagai perangkat periferil sebagai slave. Protokol ini dikembangkan oleh Motorola dan dikenal karena kecepatan transfer datanya yang tinggi, struktur yang sederhana, serta kemampuannya untuk melakukan komunikasi full-duplex.



Gambar 2. 4 Ilustrasi Komunikasi SPI [16]

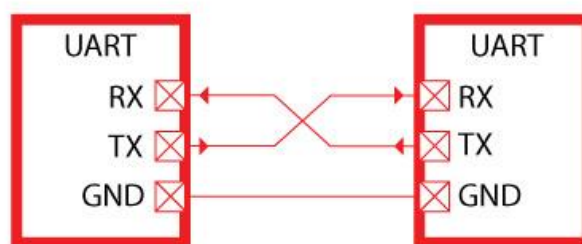
Gambar 2.4 menjelaskan bahwa SPI menggunakan empat jalur utama, yaitu MOSI (*Master Out Slave In*), MISO (*Master In Slave Out*), SCK (*Serial Clock*), dan CS atau NSS (*Chip Select/Slave Select*). Dalam konteks komunikasi antara mikrokontroler dan modul LoRa RA-02 SX1278, SPI berperan sebagai penghubung utama yang memungkinkan mikrokontroler untuk mengontrol dan mengakses fungsi-fungsi dari chip SX1278. Melalui protokol SPI, mikrokontroler dapat mengirimkan perintah untuk mengatur parameter seperti frekuensi operasi

(contoh: 433 MHz), mode kerja (*sleep, standby, transmit, receive*), *bandwidth, spreading factor, coding rate*, dan pengelolaan buffer FIFO untuk data yang akan dikirim maupun diterima. [16]

Dalam penelitian ini, komunikasi SPI menjadi aspek teknis yang sangat penting karena menentukan keberhasilan dalam proses integrasi antara mikrokontroler dan modul LoRa. Kemampuan SPI dalam mentransfer data dengan kecepatan tinggi dan latensi rendah dapat mendukung kebutuhan sistem komunikasi LoRa yang efisien, responsif, dan efektif, khususnya dalam aplikasi *Internet of Things* (IoT) yang memerlukan pertukaran data secara berkala atau *real-time*. Dengan memanfaatkan komunikasi SPI, sistem dapat dirancang untuk mengoptimalkan pengiriman data lokasi dan status perangkat *transmitter* ke *repeater* maupun *gateway*, sehingga mendukung sistem *monitoring* yang menjadi keluaran dari sistem ini.

2.6.2 UART (*Universal Asynchronous Receiver/Transmitter*)

UART (*Universal Asynchronous Receiver Transmitter*) adalah salah satu metode komunikasi data yang paling umum digunakan. Komunikasi ini bersifat serial dan bekerja secara asinkron, artinya pengiriman data tidak membutuhkan sinyal clock tambahan. Cukup dengan menyamakan kecepatan pengiriman data (*baud rate*) antara dua perangkat, komunikasi sudah bisa dilakukan.



Gambar 2. 5 Ilustrasi Komunikasi UART [17]

Gambar 2.5 merupakan ilustrasi komunikasi UART yang menggunakan dua jalur utama, yaitu TX (*transmit*) untuk mengirim data dan RX (*receive*) untuk

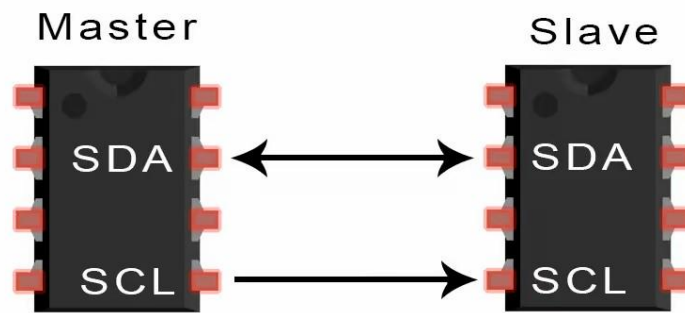
menerima data. Data dikirim secara berurutan, dimulai dari bit *start*, diikuti oleh data utama, dan diakhiri dengan bit *stop*. Protokol ini sangat efisien dan sederhana, sehingga cocok digunakan dalam sistem ini.[17] Pada penelitian kali ini, UART digunakan dalam komunikasi antara mikrokontroler dan modul GPS NEO-6M. Modul GPS ini berfungsi menerima sinyal dari satelit dan mengirimkan data lokasi seperti koordinat (*latitude* dan *longitude*), waktu, dan kecepatan. Data tersebut dikirim dalam format teks standar yang disebut NMEA, melalui pin TX modul GPS ke pin RX mikrokontroler. Modul GPS NEO-6M dikonfigurasi dengan baud rate 9600 bps, yang juga perlu disesuaikan di sisi mikrokontroler agar komunikasi berjalan lancar. Setelah data diterima, mikrokontroler akan memproses atau “mem-parsing” data NMEA tersebut untuk mengambil informasi penting yang dibutuhkan.

Kelebihan dari komunikasi UART adalah implementasinya yang mudah karena hanya memerlukan dua kabel, serta sudah banyak didukung oleh berbagai jenis mikrokontroler salah satunya yaitu ESP32.

2.6.3 I2C (*Inter-Integrated Circuit*)

I2C (Inter-Integrated Circuit) adalah salah satu jenis komunikasi serial yang populer dan sering digunakan untuk menghubungkan mikrokontroler dengan berbagai perangkat tambahan, seperti sensor, modul memori, modul waktu nyata (RTC), atau modul tampilan seperti LCD I2C. Protokol ini hanya menggunakan dua jalur utama, yaitu SDA (Serial Data Line) untuk mengirim dan menerima data, serta SCL (Serial Clock Line) untuk mengatur sinyal clock atau waktu transfer data. Keunggulan utama dari I2C adalah kemampuannya menghubungkan banyak perangkat hanya dengan dua kabel, sehingga sangat efisien dan sederhana, terutama dalam proyek-proyek berbasis mikrokontroler.

Pada penelitian kali ini, I2C berperan dalam komunikasi antara LCD I2C dengan mikrokontroler ESP32. Modul LCD I2C merupakan pengembangan dari LCD paralel standar yang dilengkapi dengan konverter I2C, sehingga hanya memerlukan dua pin untuk pengoperasiannya.



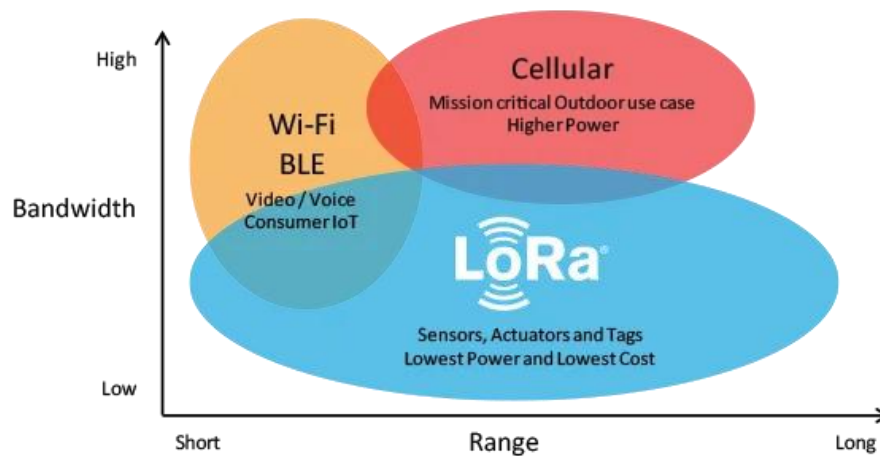
Gambar 2. 6 Ilustrasi Komunikasi I2C [18]

Gambar 2.6 menunjukkan ilustrasi komunikasi I2C antara mikrokontroler dan LCD I2C. Komunikasi dimulai ketika mikrokontroler (sebagai master) mengirim sinyal start, diikuti oleh alamat dari LCD (sebagai slave) beserta perintah untuk menulis data. LCD merespons dengan sinyal acknowledge (ACK). Setelah itu, mikrokontroler mengirim data berupa karakter atau perintah tampilan (seperti posisi kursor, pembersihan layar, dan sebagainya) secara berurutan. Setelah setiap byte diterima, LCD memberikan ACK, dan proses ini berlanjut hingga seluruh data ditransmisikan. Komunikasi diakhiri dengan sinyal stop yang dikirim oleh mikrokontroler.

Dengan komunikasi ini, mikrokontroler dapat mengontrol tampilan pada LCD dengan efisien dan menggunakan sedikit jalur koneksi, menjadikannya sangat cocok untuk sistem yang memiliki keterbatasan jumlah pin..

2.6.4 LoRa (*Long Range*)

LoRa (*Long Range*) merupakan teknologi berbasis nirkabel yang dibuat oleh Semtech demi kebutuhan transmisi data dan komunikasi jarak jauh. Memiliki keunggulan terhadap konsumsi daya yang rendah, serta implementasi yang cukup mudah. LoRa banyak digunakan dalam kebutuhan IoT yang memerlukan komunikasi jarak jauh dengan kebutuhan daya yang rendah. Dalam spesifikasinya, LoRa dapat melakukan transmisi data jarak jauh hingga 10 km di area terbuka dan 2-3km di area ramai penduduk.



Gambar 2. 7 Perbandingan Protokol Komunikasi LoRa [19]

Gambar 2.7 menunjukkan bahwa protokol komunikasi LoRa memiliki jangkauan yang lebih jauh jika di bandingkan dengan protokol komunikasi lain seperti WiFi, BLE, dan Seluler. LoRa juga memiliki keunggulan dalam efisiensi daya, dimana hanya memiliki penerimaan arus sebesar 10 mA. Selain itu, LoRa juga dapat bertahan di kondisi lingkungan yang memiliki banyak *noise* dengan menggunakan teknologi *spread spectrum* yang memungkinkan LoRa dapat menyebarkan sinyal lebih luas daripada yang di perlukan untuk transmisi data normal. [20]

Tabel 2. 1 Detail LoRa *Physical Layer*

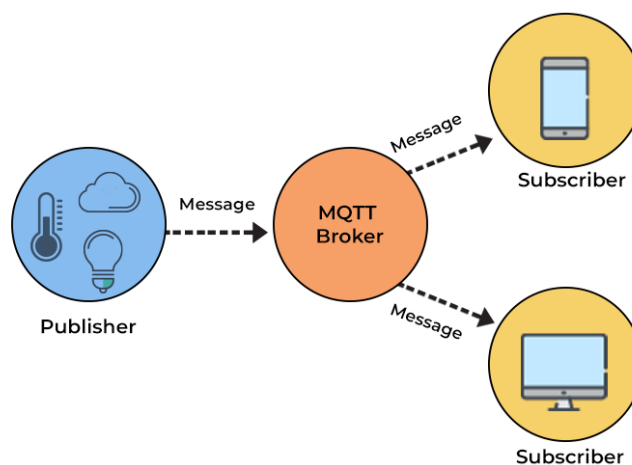
LoRa <i>Physical Layer</i>			
<i>Preamble</i>	<i>Sync</i>	<i>Payload</i>	CRC
<i>Message</i>			

Tabel 2.1 merupakan struktur frame pada Physical Layer LoRa yang merupakan bagian penting dari proses komunikasi LoRa itu sendiri. Lapisan ini bertanggung jawab untuk modulasi sinyal radio. LoRa menggunakan *Chirp Spread Spectrum* (CSS) untuk memungkinkan komunikasi jarak jauh dengan daya rendah dan tahan terhadap interferensi. Bagian – bagian physical layer LoRa terdiri dari *Preamble* yang berfungsi untuk menyinkronkan *receiver* (penerima) dengan sinyal yang akan datang, *Sync Message* yang berfungsi sebagai penanda akhir dari *preamble* dan memberi tahu *receiver* bahwa *payload* akan segera datang.

Selanjutnya yaitu *Payload* yang berfungsi sebagai bagian utama dari frame yang akan membawa data aktual. Pada penelitian kali ini, *payload* akan membawa data kondisi serta titik koordinat dari *transmitter*. Lalu CRC merupakan pilihan opsional, tetapi sangat penting untuk mendeteksi kesalahan saat transmisi data.

2.6.5 MQTT (*Message Queuing Telemetry Transport*)

Merupakan sebuah protokol komunikasi yang ringan dan berorientasi pesan. Dirancang khusus untuk aplikasi IoT dan komunikasi antar suatu sistem. MQTT juga terkenal dengan efisiensi pengiriman data yang tinggi, menggunakan topik sebagai alat untuk mengelompokkan serta mengatur sebuah pesan yang akan dikirimkan.



Gambar 2. 8 Ilustrasi MQTT Process [21]

Gambar 2.8 menunjukkan sebuah ilustrasi penerapan MQTT dengan *publisher/subscriber*, ada dua peran utama yaitu penerbit yang mengirimkan data, dan pelanggan yang menerima data. Keduanya tidak berkomunikasi langsung satu sama lain. Sebagai gantinya, ada pihak ketiga yang disebut *broker* (perantara), yang mengatur pengiriman pesan dari penerbit ke pelanggan. Penerbit adalah perangkat atau aplikasi yang mengirim pesan ke *broker*. Pesan ini nantinya akan disalurkan ke pelanggan yang membutuhkannya. Pelanggan adalah perangkat atau aplikasi yang telah mendaftar (*subscribe*) untuk menerima pesan tertentu.

Satu klien MQTT bisa bertindak sebagai penerbit, pelanggan, atau keduanya sekaligus. *Broker* akan memfilter pesan berdasarkan topik (kategori atau label pesan), sehingga pelanggan hanya menerima pesan yang sesuai dengan minatnya.

Topik dalam MQTT bisa disusun secara hierarkis (bertingkat), sehingga mudah untuk mengatur dan mengelola berbagai jenis data. Dengan model ini, penerbit tidak perlu tahu siapa pelanggannya, begitu juga sebaliknya. Semua komunikasi berlangsung melalui *broker*, membuat sistem ini efisien, terstruktur, dan mudah diperluas.[21]

Pada kondisi kali ini, penulis menggunakan metode *publisher-subscriber* yang memungkinkan komunikasi asinkron antara beberapa perangkat. Metode ini juga dapat di aplikasikan secara mudah karena dapat meminimalkan beban jaringan yang digunakan. Penerapan komunikasi tersebut dipilih karena sifatnya yang cukup ringan, dapat diandalkan, mempunyai skalabilitas yang baik, dan mendukung berbagai aplikasi termasuk pemantauan sistem GPS yang akan menjadi topik dari penelitian ini.

Pada kasus kali ini, MQTT digunakan sebagai protokol komunikasi dari perangkat *gateway* menuju *dashboard* pemantauan. Menggunakan *broker Mosquitto* untuk komunikasi yang memiliki efisiensi tinggi. Selain itu, MQTT pada kasus ini juga membutuhkan adanya koneksi WiFi dari ESP 32 yang ada di *gateway*. Dengan implementasi protokol MQTT penulis dapat mengintegrasikan koneksi antara hardware dan juga *software* di dalam sistem *GPS-Based alarm* tersebut.

2.7 Mikrokontroler ESP 32

ESP32 merupakan salah satu modul mikrokontroler yang sering digunakan dalam pengembangan sistem *Internet of Things* (IoT), karena kemampuannya yang dapat mendukung konektivitas ganda melalui Wi-Fi dan Bluetooth dalam satu chip. Modul ini juga dilengkapi dengan berbagai antarmuka digital yang cukup lengkap seperti SPI, UART, I2C, I2S, dan PWM, sehingga memudahkan integrasi dengan berbagai sensor dan perangkat eksternal. Selain antarmuka digital, ESP32 juga menyediakan fitur input/output analog, yaitu dua ADC (*Analog to Digital Converter*) 12-bit yang mendukung hingga 18 *channel*, serta dua DAC (*Digital to Analog Converter*) 8-bit, yang memungkinkan pemrosesan sinyal analog secara fleksibel.

Pada varian ESP32 DevKit V1, tersedia sekitar 25 pin GPIO multifungsi yang bisa digunakan untuk *input/output* digital, analog, dan komunikasi serial. Kemampuan ESP32 dalam mendukung standar Wi-Fi dan Bluetooth, termasuk BLE (*Bluetooth Low Energy*), membuatnya sangat efisien untuk sistem IoT yang memerlukan konektivitas nirkabel dan konsumsi daya rendah. Selain itu, ESP32 juga dibekali dengan prosesor *dual-core* yang mampu berjalan hingga kecepatan 240 MHz, sehingga cukup andal dalam menangani proses komputasi yang lebih kompleks. [22]

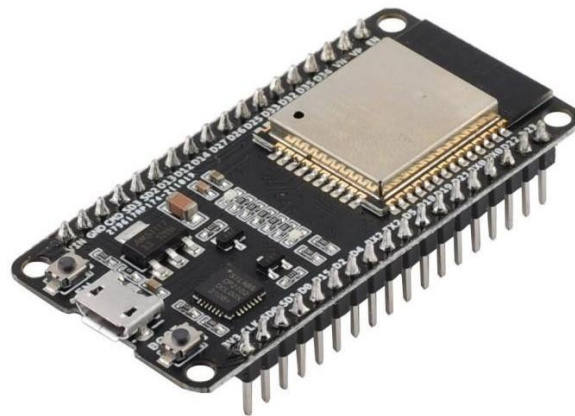
Dalam penelitian ini, ESP32 berperan sebagai unit pemrosesan utama yang bertugas membaca data lokasi dari modul GPS melalui komunikasi serial (UART). Informasi koordinat yang diperoleh kemudian diproses dan dibandingkan dengan batas wilayah (*Geofencing*) yang telah ditentukan. Jika lokasi yang terdeteksi melebihi batas aman, ESP32 akan mengaktifkan alarm peringatan melalui *buzzer*, serta pengiriman data melalui protokol komunikasi LoRa (Long Range). Penggunaan ESP32 pada sistem ini sangat cocok karena mendukung integrasi dengan GPS, LoRa, serta kemampuan pengiriman data *real-time* melalui internet dengan protokol MQTT, serta mendukung pengembangan antarmuka *dashboard monitoring* berbasis HTML.

Selain itu, ESP32 juga mendukung pemrograman menggunakan bahasa Arduino IDE yang memberikan fleksibilitas dalam pengembangan sistem *GPS-based alarm* berbasis IoT untuk berbagai skenario, mulai dari pengawasan, *monitoring* kapal di wilayah perairan, hingga sistem peringatan pelanggaran batas wilayah secara otomatis.[23]. Berikut merupakan Gambar dan juga Tabel spesifikasi dari Mikrokontroller ESP32 Dev Kit V1.

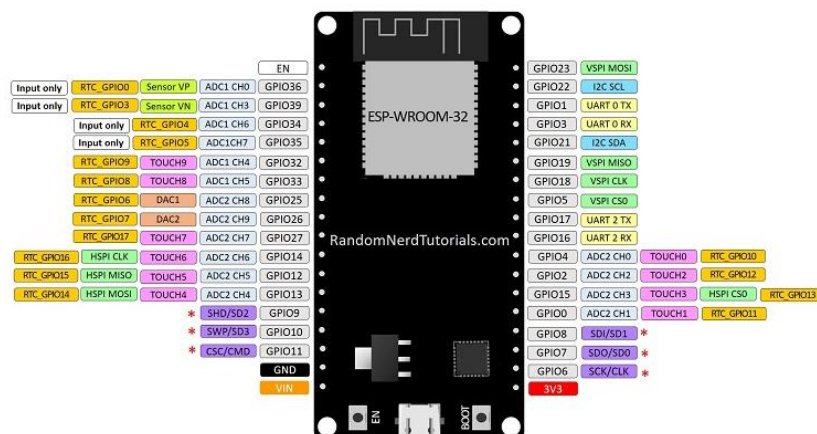
Tabel 2. 2 Spesifikasi ESP 32 DevKit V1

Spesifikasi	Keterangan
Mikrokontroller	ESP 32
Tegangan Operasional	3,0V – 3,6V (Umumnya 3,3V)
Arus Operasional	160 - 260 mA, 80 - 120mA
Tegangan Input	5V via USB <i>Connector</i>

Spesifikasi	Keterangan
Jumlah Pin Digital I/O	34
Jumlah Pin Analog	18
Kecepatan Clock	240 MHz
Fitur Tambahan	WiFi 802.11 b/g/n, Bluetooth 4.2



Gambar 2. 9 ESP 32 Dev Kit V1 [22]



2.8 Modul GPS

Dalam penelitian ini, penulis menggunakan dua jenis modul GPS yang berbeda untuk membandingkan kinerjanya. Tujuannya adalah untuk mengetahui modul mana yang lebih efisien, akurat, dan sesuai digunakan dalam sistem GPS-based alarm berbasis LoRa-MQTT. Perbandingan ini dilakukan agar sistem dapat bekerja secara optimal dalam mendeteksi posisi kapal secara *real-time*.

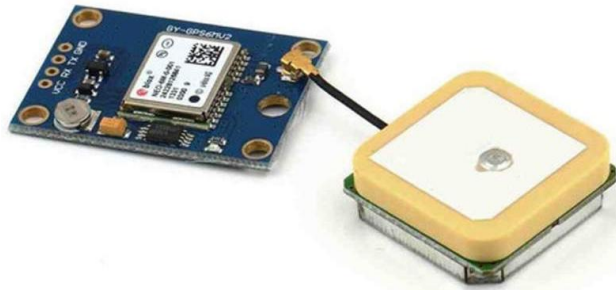
2.8.1 GPS Neo-6M

Modul GPS Neo-6M merupakan salah satu perangkat penerima sinyal GPS yang dikenal memiliki performa andal dan banyak digunakan dalam berbagai aplikasi pemantauan lokasi. Modul ini dikembangkan oleh perusahaan u-blox dan telah menjadi pilihan populer dalam proyek-proyek elektronik maupun sistem navigasi berbasis GPS. Neo-6M mampu memberikan data posisi geografis dan informasi waktu dengan tingkat akurasi yang tinggi, serta memiliki sensitivitas sinyal yang baik sehingga tetap dapat berfungsi optimal meskipun berada di lingkungan dengan kondisi penerimaan sinyal yang kurang ideal. Selain itu, modul ini mendukung beberapa protokol komunikasi standar seperti NMEA dan ublox, sehingga memudahkan proses integrasi dengan mikrokontroler dan sistem elektronik lainnya. [25]

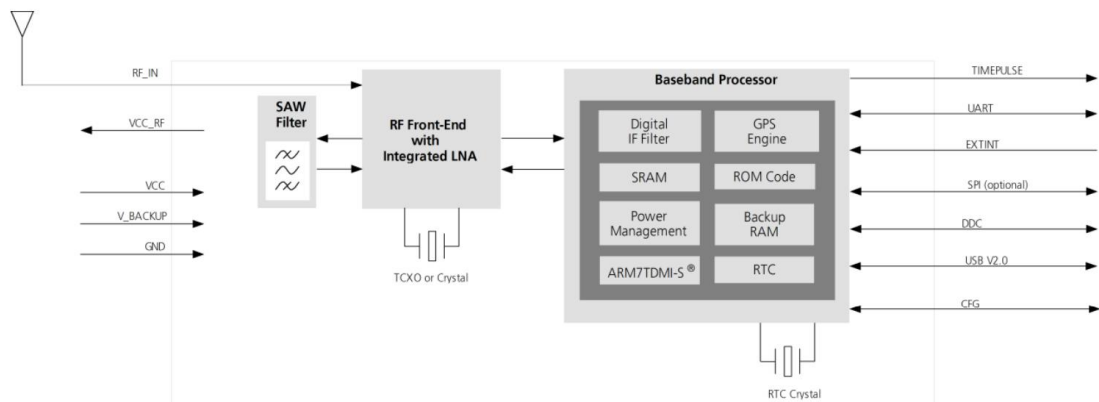
Berikut merupakan Tabel Spesifikasi yang mencakup beberapa fitur kunci :

Tabel 2. 3 Spesifikasi Modul GPS Neo-6M

Spesifikasi	Keterangan
Tegangan Operasi	3.3V – 5V
Komunikasi	UART (<i>baud rate</i> default 9600 bps)
Antena	<i>Patch ceramic</i> + opsional eksternal
Akurasi Posisi	~2.5 meter
Update Rate	1 Hz (maksimal 5 Hz)
Jumlah Satelit	Hingga 22 <i>tracking</i> , 50 <i>channel</i>
Ukuran	±25 mm x 35 mm



Gambar 2.11 Sensor GPS Neo-6M [26]



Gambar 2. 12 Skematik Modul GPS Ublox Neo-6M [25]

2.8.2 GPS Neo-M8N

Modul GPS Neo-M8N merupakan salah satu perangkat penerima sinyal satelit generasi terbaru yang dikembangkan oleh u-blox. Dibandingkan dengan seri sebelumnya (Neo-6M), Neo-M8N menawarkan peningkatan performa, terutama dalam hal sensitivitas, akurasi, serta kecepatan akuisisi posisi. Modul ini mendukung sistem satelit global multi-konstelasi seperti GPS, GLONASS, Galileo, dan BeiDou, sehingga mampu memberikan cakupan dan ketepatan lokasi yang lebih baik, khususnya di area dengan kondisi sinyal yang menantang.

Selain memberikan informasi posisi geografis, Neo-M8N juga mampu menyajikan data kecepatan dan waktu dengan presisi tinggi. Dengan dukungan protokol komunikasi standar seperti NMEA dan u-blox binary protocol, modul ini mudah diintegrasikan dengan berbagai mikrokontroler maupun sistem elektronik

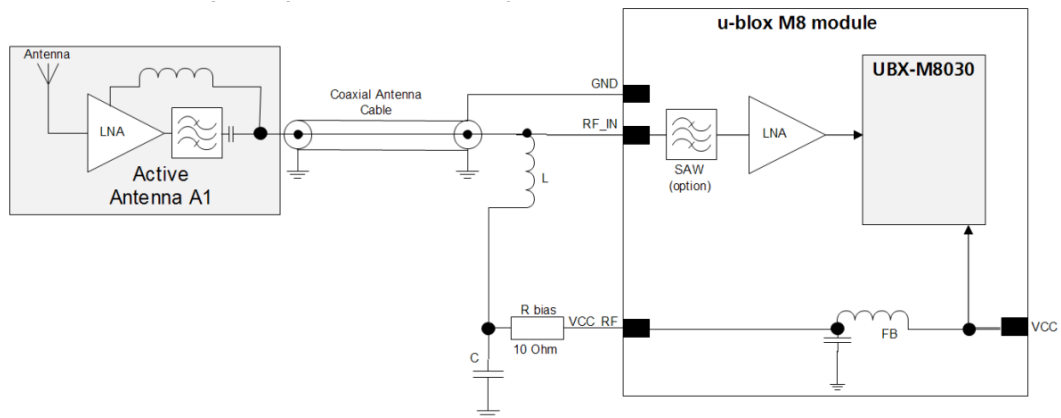
untuk aplikasi pemantauan dan navigasi. Keunggulan lainnya adalah tingkat Time To First Fix (TTFF) yang lebih cepat, serta konsumsi daya yang relatif rendah, sehingga menjadikannya pilihan ideal dalam proyek-proyek berbasis Internet of Things (IoT), sistem pelacakan kendaraan, hingga aplikasi kelautan.[27]

Tabel 2. 4 Spesifikasi Modul GPS Neo-M8N

Spesifikasi	Keterangan
Modul GPS	u-blox NEO-M8N
Tegangan Operasi	3.3V – 5V
Komunikasi	UART, I ² C, SPI (<i>baud rate</i> default 9600 bps)
Antena	<i>Patch ceramic</i> + opsional eksternal
Akurasi Posisi	~2.5 meter
Update Rate	1 Hz (maksimal 10 Hz)
Jumlah Satelit	Mendukung hingga 72 channel GNSS: GPS, GLONASS, Galileo, QZSS, SBAS
Ukuran	±25 mm x 35 mm



Gambar 2. 13 Sensor GPS Neo-M8N [27]



Gambar 2. 14 Skematik Modul GPS Ublox Neo-M8N [27]

Pada penelitian kali ini, kedua modul GPS akan diuji dengan membandingkan data koordinat yang diperoleh dari modul dengan data GPS yang berasal dari *smartphone* peneliti untuk mengetahui tingkat akurasi pembacaan dari modul GPS tersebut. Perbedaan antara titik koordinat modul dan *smartphone* akan dihitung sebagai kesalahan, yang diantaranya mencakup kesalahan lintang (*latitude*) dan kesalahan bujur (*longitude*). Formula untuk mencari nilai kesalahan Modul GPS adalah sebagai berikut :

$$\text{kesalahan } \alpha = | \alpha_{\text{modul}} - \alpha_{\text{smartphone}} | \quad (2.1)$$

$$\text{kesalahan } \beta = | \beta_{\text{modul}} - \beta_{\text{smartphone}} | \quad (2.2)$$

$$\overline{\text{kesalahan } \alpha} = \frac{\sum \text{kesalahan } \alpha}{n} \quad (2.3)$$

$$\overline{\text{kesalahan } \beta} = \frac{\sum \text{kesalahan } \beta}{n} \quad (2.4)$$

Berdasarkan formula 2.1, 2.2, 2.3, dan 2.4 yang telah ditentukan, maka kami akan mendapatkan nilai kesalahan yang berada pada satuan Menit. Oleh karena itu, masih diperlukan formula berikutnya untuk menghasilkan nilai kesalahan yang tepat dalam satuan meter. Pada dasarnya, satu derajat lintang di permukaan bumi mewakili jarak yang tepat, yaitu sekitar 111,320 meter. Sementara itu, konversi untuk bujur bergantung pada posisi lintangnya, dengan satu derajat bujur di ekuator mewakili jarak 111,320 meter, tetapi menurun dengan meningkatnya lintang.

Formula konversi kesalahan lintang dan bujur ke dalam satuan meter adalah sebagai berikut:

$$\text{kesalahan } \alpha \text{ (meter)} = \text{kesalahan } \alpha \text{ (menit)} \times \frac{111,320 \text{ meter}}{60 \text{ menit}} \quad (2.5)$$

$$\text{kesalahan } \beta \text{ (meter)} = \text{kesalahan } \beta \text{ (menit)} \times \frac{111,320 \text{ meter}}{60 \text{ menit}} \times \cos \beta \quad (2.6)$$

Dari dua formula yang ada di atas, kita dapat menggabungkan jarak kesalahan dalam satuan meter dengan menggunakan rumus *Pythagoras*.

$$\text{jarak (meter)} = \sqrt{\{\text{kesalahan } \alpha \text{ (meter)}\}^2 + \{\text{kesalahan } \beta \text{ (meter)}\}^2} \quad (2.7)$$

Dengan perhitungan tersebut, kita dapat mengetahui seberapa besar perbedaan antara koordinat yang dihasilkan oleh modul GPS Neo-6M dan GPS pada *smartphone*, serta mengonversi kesalahan tersebut ke dalam satuan jarak yang lebih mudah di pahami.

2.9 Buzzer 3-24 V

Buzzer merupakan salah satu jenis aktuator elektronik yang berfungsi menghasilkan suara sebagai bentuk peringatan atau notifikasi. Dalam sistem alarm berbasis GPS, perangkat ini berperan penting sebagai alat aktuasi yang memberikan sinyal suara saat suatu objek, seperti kapal keluar dari zona aman yang telah ditentukan menggunakan metode *Geofencing*.

Jenis *buzzer* yang digunakan dalam sistem ini adalah *buzzer* aktif 3–24V. *Buzzer* jenis ini telah dilengkapi dengan rangkaian osilator internal, sehingga cukup diberi tegangan DC dalam rentang 3 hingga 24 volt untuk dapat langsung menghasilkan suara. Hal ini membedakannya dari *buzzer* pasif, yang memerlukan sinyal frekuensi khusus (seperti PWM) dari mikrokontroler agar dapat berfungsi.

Umumnya, *buzzer* 3–24V bekerja pada frekuensi sekitar 2.000 hingga 4.000 Hz dan membutuhkan arus listrik relatif kecil, yaitu sekitar 5 hingga 30 miliampere, tergantung pada tegangan kerja. Oleh karena itu, *buzzer* ini dapat dikendalikan secara langsung oleh mikrokontroler seperti ESP32, atau melalui transistor sebagai driver jika dibutuhkan tegangan dan arus lebih besar. Komponen ditampilkan pada Gambar 2. 15, dan spesifikasi ditampilkan dalam Tabel 2.5.

Tabel 2. 5 Spesifikasi *Buzzer* 3 – 24V

Spesifikasi	Keterangan
Tegangan Kerja	3V – 24V DC
Arus Konsumsi	5 – 30 mA
Frekuensi Suara	2kHz – 4kHz
Diameter Fisik	12 mm – 15 mm
Tipe	Aktif / Pasif

**Gambar 2. 15** *Buzzer* 3-24V [28]

2.10 Metode *Geofencing*

Geofencing adalah suatu teknik pemrosesan spasial berbasis lokasi yang secara umum digunakan untuk mendeteksi posisi dan pergerakan objek dinamis seperti manusia, kendaraan, atau kapal terhadap suatu wilayah geografis tertentu yang telah ditetapkan sebelumnya secara digital. Wilayah ini dikenal dengan istilah “*geofence*”, yaitu area virtual yang dibentuk dalam ruang dua dimensi, biasanya berupa lingkaran, poligon, atau bahkan garis lurus yang mewakili batas tertentu, baik di darat maupun di laut. Dengan memanfaatkan teknologi GPS (*Global Positioning System*), perangkat dapat memperoleh data koordinat lokasi secara *real-time* dan membandingkannya dengan batas-batas digital *geofence* untuk mengetahui apakah objek berada di dalam, di luar, atau mendekati wilayah tersebut. Teknologi ini telah banyak diterapkan dalam berbagai sektor seperti sistem keamanan wilayah, pelacakan armada kendaraan, pengiriman logistik, sistem peringatan dini bencana, serta pengawasan wilayah-wilayah sensitif atau terbatas

aksesnya. Sensor GPS Neo-6M berperan penting dalam sistem ini karena mampu memberikan data lokasi dengan tingkat akurasi yang memadai untuk keperluan deteksi spasial secara terus-menerus.

Keunggulan metode ini terletak pada kemampuannya dalam mendefinisikan batas-batas geografis secara fleksibel dan digital tanpa perlu infrastruktur fisik yang kompleks. Cukup dengan data koordinat dan logika pemrograman, sistem dapat merespons perubahan posisi objek secara otomatis, baik dengan mengirimkan notifikasi, mengaktifkan alarm, atau mencatat kejadian dalam sistem basis data. Dalam konteks penelitian ini, *Geofencing* digunakan untuk memantau posisi kapal terhadap batas maritim suatu negara, khususnya dalam upaya perlindungan wilayah perairan Indonesia. Dengan menggunakan koordinat GPS, sistem dapat mengetahui apakah kapal masih berada dalam batas maritim Indonesia, berada di zona rawan, atau telah melanggar batas tersebut.

Tabel 2. 6 Logika Metode *Geofencing*

Status	Logika
AMAN	$D \geq 0$ dan $S \geq 1\text{KM}$
WASPADA	$D \geq 0$ dan $S < 1\text{KM}$
PELANGGARAN	$D < 0$

Pada penelitian kali ini, status *Geofencing* diklasifikasikan ke dalam tiga kategori utama seperti tercantum dalam Tabel 2. 6, yaitu AMAN, WASPADA, dan PELANGGARAN. Kategori AMAN mengindikasikan bahwa posisi kapal masih berada di dalam wilayah perairan Indonesia yang sah secara hukum. Sementara itu, kategori WASPADA berlaku ketika kapal berada dalam jarak kurang dari 1 kilometer dari batas maritim, yang menunjukkan potensi risiko pelanggaran yang memerlukan perhatian atau tindakan antisipatif. Sedangkan kategori PELANGGARAN digunakan ketika kapal diketahui telah melewati batas maritim Indonesia, yang menandakan telah terjadi pelanggaran wilayah yang bisa berdampak pada aspek hukum, ekonomi, dan keamanan nasional. Dengan

pendekatan ini, metode *Geofencing* tidak hanya menjadi alat pemantauan spasial, tetapi juga berfungsi sebagai sistem pendukung keputusan untuk tindakan cepat dan terukur terhadap pelanggaran batas maritim.

2.10.1 *Line crossing detection*

Line crossing detection merupakan salah satu metode *Geofencing* yang efektif dalam sistem *monitoring* berbasis GPS, khususnya dalam pengawasan batas maritim Indonesia. Metode ini digunakan untuk mendeteksi apakah suatu objek yang dalam hal ini merupakan kapal nelayan telah melintasi suatu garis virtual tertentu. Garis ini didefinisikan oleh dua titik koordinat geografis (*latitude*, dan *longitude*). Metode ini bekerja berdasarkan prinsip geometri dua dimensi, di mana garis batas didefinisikan oleh dua titik koordinat GPS yang merepresentasikan awal dan akhir dari sebuah garis virtual. Dalam penelitian kali ini, garis tersebut berfungsi sebagai garis batas maritim Indonesia. Posisi kapal yang terus menerus diperoleh dari sensor GPS akan dianalisis terhadap garis batas tersebut untuk menentukan klasifikasi kondisi kapal secara *real-time*. Singkatnya, metode ini berjalan dengan menentukan posisi sebelumnya dan posisi sekarang, lalu menghitung arah lintasan dan mengevaluasi apakah garis batas telah dilewati. Berikut merupakan formula yang digunakan dalam metode *Geofencing line crossing detection*. Secara matematis, *Line crossing detection* menggunakan perhitungan determinan dua dimensi untuk mengetahui posisi relatif suatu kapal terhadap garis. Formula yang digunakan adalah :

$$D = (lat_p - lat_1)(lon_2 - lon_1) - (lon_p - lon_1)(lat_2 - lat_1) \quad (2.8)$$

D pada formula 2.8 merupakan determinan dari tiga titik di bidang kartesius yang biasanya digunakan untuk menentukan apakah tiga titik tersebut kolinear atau tidak. Di mana lat_p dan lon_p adalah posisi kapal secara *real-time*, sedangkan lat_1 , lon_1 dan lat_2 , lon_2 merupakan koordinat titik awal dan akhir dari garis batas maritim. Hasil formula di atas akan membagi *output* menjadi dua kondisi yakni kondisi “pelanggaran” jika $D < 0$, dan kondisi “aman”/”waspada” jika $D \geq 0$. Karena klasifikasi *output* yang diinginkan berjumlah tiga, kami perlu membagi kembali klasifikasi yang akan membedakan kondisi “aman” atau “waspada”. Pada penelitian ini, jarak merupakan variabel yang sangat penting untuk menentukan

apakah kapal berada di kondisi aman atau waspada dikarenakan kedua *output* tersebut memiliki keluaran yang berbeda dalam proses sistem. Oleh karena itu, kami merumuskan suatu formula untuk menentukan jarak dari kapal menuju garis batas maritim. Formula yang digunakan adalah:

$$S = \frac{|(lat_2-lat_1)lon_p-(lon_2-lon_1)lat_p+lon_2lat_1-lat_2lon_1|}{\sqrt{(lat_2-lat_1)^2+(lon_2-lon_1)^2}} \quad (2.9)$$

Formula 2.9 disusun bertujuan untuk mendapatkan jarak dari suatu titik ke suatu garis dalam bidang koordinat. Setelah melalui formula tersebut, akan didapatkan kondisi apakah kapal berada dalam kondisi “aman” atau “waspada”. Dengan beberapa formula yang telah ditentukan, output berdasarkan metode *Line crossing detection* dapat di klasifikasikan menjadi tiga *output* yang tentunya memiliki reaksi yang berbeda terhadap pengiriman data maupun aktuasi *buzzer*.

2.11 LoRa SX 1278

Modul LoRa SX1278 adalah salah satu komponen penting dalam sistem komunikasi nirkabel jarak jauh yang dikenal irit daya dan sangat cocok digunakan di lingkungan terbuka. Modul ini mengadopsi teknologi LoRa (*Long Range*) buatan Semtech, yang dirancang khusus untuk mengatasi keterbatasan jangkauan, terutama di area luas seperti lautan. Dengan memanfaatkan modul LoRa ini, data dapat dikirim dalam jarak yang sangat jauh, bahkan bisa mencapai puluhan kilometer di area tanpa halangan serta tanpa memerlukan konsumsi daya yang besar. Hal ini membuat SX1278 sangat ideal untuk sistem pemantauan posisi yang bergantung pada sumber daya baterai, seperti pada kapal nelayan.

Modul SX1278 bekerja di frekuensi 433 - 915 MHz, yang termasuk ke dalam frekuensi ISM (*Industrial, Scientific, and Medical*) dan bebas digunakan tanpa lisensi tambahan di Indonesia. Modul ini juga fleksibel karena bisa disesuaikan pengaturannya, seperti *bandwidth*, *spreading factor*, hingga *coding rate*. Dengan begitu, pengguna bisa menyesuaikan antara kecepatan pengiriman data, kekuatan sinyal, dan jangkauan komunikasi, sesuai dengan kondisi lingkungan maupun kebutuhan aplikasi di lapangan.[29]

Dalam penelitian ini, SX1278 dimanfaatkan untuk mengirim data posisi kapal nelayan yang didapat dari modul GPS, agar bisa dipantau secara *real-time* dari

stasiun penerima di darat. Modul ini mengirimkan data koordinat dalam bentuk paket digital yang ringan, sehingga tetap bisa mengirim informasi meskipun kapal sedang berada jauh dari jangkauan sinyal seluler. Selain itu, LoRa SX1278 juga dilengkapi fitur *Forward Error Correction* (FEC) yang berfungsi mengoreksi kesalahan data akibat gangguan sinyal secara otomatis, sehingga proses pengiriman tetap stabil dan akurat, bahkan dalam kondisi cuaca buruk atau jarak tempuh yang jauh.

Lebih dari itu, dalam sistem ini, LoRa SX1278 juga diintegrasikan dengan protokol MQTT (*Message Queuing Telemetry Transport*) di perangkat *receiver*, agar data yang masuk dari kapal bisa langsung dikirimkan ke server *monitoring* berbasis cloud dengan lebih efisien. Kombinasi antara LoRa dan MQTT ini dirancang agar sistem tidak hanya hemat biaya, namun juga mudah dipasang, aman, dan tetap andal untuk digunakan oleh nelayan tradisional, terutama di wilayah-wilayah yang belum terjangkau oleh jaringan komunikasi seluler.[5]

Sistem yang memanfaatkan LoRa SX1278 dan dikembangkan dalam penelitian ini diharapkan mampu memperluas cakupan pemantauan kapal nelayan, sekaligus mengaktifkan alarm otomatis jika posisi kapal melampaui batas wilayah maritim Indonesia yang telah ditentukan. Langkah ini diharapkan bisa mendukung nelayan agar terhindar dari pelanggaran batas perairan, baik yang disengaja maupun tidak, dan ikut serta dalam menjaga kedaulatan maritim nasional secara lebih efektif. Komponen ditampilkan pada Gambar 2. 16, Skematik pada Gambar 2. 17, dan spesifikasi ditampilkan dalam Tabel 2.7.

Tabel 2. 7 Spesifikasi Modul LoRa Ra-2 SX1278

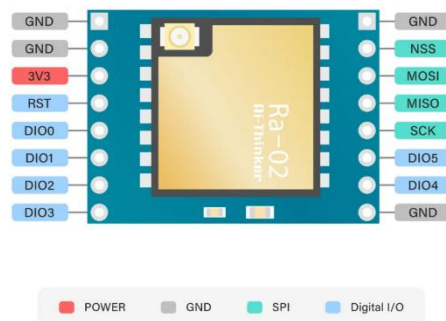
Spesifikasi	Keterangan
Dimensi	25 x 17 x 3 mm
Frekuensi Kerja	433, 868, atau 915 MHz
Kecepatan Data	0,3 hingga 250 kbps
Konsumsi Daya	100 μ hingga 200m A
Spreading Factor	SF7 hingga SF12
GPIO	10 pin

Antarmuka Komunikasi	SPI
Antena	<i>Integrated Patch Antenna (opsional)</i>



Gambar 2. 16 Modul LoRa SX 1278 [30]

RA-02 SX1278 Module



Gambar 2. 17 Skematik LoRa Ra-02 SX1278 [31]

2.12 Catu Daya (Baterai Li-Ion 18650)

Baterai Li-ion 18650 merupakan opsi yang paling ideal untuk menunjang catu daya sistem pemantauan lokasi kapal nelayan berbasis GPS dan LoRa-MQTT karena memiliki kapasitas tinggi, bentuk yang ringkas, serta efisiensi daya yang baik. Baterai ini umumnya memiliki tegangan nominal 3.7V dan kapasitas antara 2200 mAh per sel. Dengan kemampuan untuk diisi ulang ratusan kali, baterai 18650 sangat cocok untuk perangkat yang dioperasikan di area terpencil seperti perairan laut, di mana akses terhadap sumber listrik terbatas. Dalam sistem ini, dua buah baterai disusun secara paralel untuk meningkatkan kapasitas tanpa menaikkan

tegangan, sehingga mampu menyuplai daya ke mikrokontroler ESP32, modul GPS Neo-6M, dan LoRa SX1278 secara berkelanjutan.

Untuk mendukung penggunaan baterai 18650, sistem ini dilengkapi dengan holder baterai berkapasitas dua slot yang dirancang sesuai ukuran standar 18650. Penggunaan holder memudahkan proses pemasangan dan penggantian baterai tanpa perlu menyolder ulang, serta menjaga kestabilan fisik baterai di lingkungan yang bergerak seperti di atas kapal. *Holder* ini juga mendukung integrasi dengan modul manajemen daya TP4056 (untuk pengisian ulang melalui USB) dan *step-up converter* (untuk menaikkan tegangan ke level yang dibutuhkan oleh perangkat). Kombinasi ini menjadikan sistem catu daya mampu menunjang operasional sistem pemantauan nelayan di laut. [32] Komponen ditampilkan pada Gambar 2. 18, dan spesifikasi ditampilkan dalam Tabel 2.8.

Tabel 2. 8 Spesifikasi Baterai Li-Ion 18650

Spesifikasi	Keterangan
Tegangan	3,7V – 4,2V
Kapasitas Umum	2000 – 3000 mAh
Arus Maksimum	2A – 4A discharge



Gambar 2. 18 Baterai Li-Ion 18650 [33]

Pada tahap pengujian di Bab IV, peneliti melakukan analisis terhadap keluaran daya sistem secara keseluruhan dengan mengacu pada nilai tegangan dan arus yang dihasilkan oleh catu daya. Selain itu, dilakukan juga perhitungan mengenai konsumsi arus masing-masing perangkat serta estimasi ketahanan baterai. Perhitungan ini didasarkan pada arus rata-rata, tegangan kerja, dan kapasitas baterai yang digunakan.

$$P = V \times I \quad (2.10)$$

$$T = \frac{C}{I} \quad (2.11)$$

Berdasarkan perumusan pada (2.10) nilai P merupakan daya yang dihasilkan berdasarkan tegangan (V) dan arus (I) yang didapatkan dari pengukuran. Lalu, berdasarkan perumusan (2.11) nilai T merupakan waktu ketahanan baterai dalam satuan jam yang didapatkan dari nilai kapasitas baterai sesuai pada datasheet (C) dan juga arus konsumsi rata-rata (I) yang didapatkan dari pengukuran.

2.12.1 Step-Up Regulator 3.7V – 5V

AKK Voltage Step Up Regulator 3.7V - 5V adalah sebuah modul DC-DC boost converter yang berfungsi untuk menaikkan tegangan input dari sumber rendah seperti baterai 3.7V (misalnya baterai 18650) menjadi tegangan output tetap sebesar 5V. Modul ini menggunakan prinsip kerja boost converter, yaitu menyimpan energi dalam induktor saat saklar aktif, lalu melepaskannya ke beban saat saklar mati, menghasilkan tegangan output yang lebih tinggi dari input. Proses ini dikontrol secara cepat dan efisien oleh rangkaian switching internal, sehingga output tetap stabil.

Sesuai pada tabel 2.9, *step-up converter* digunakan untuk mensuplai tegangan 5V yang stabil untuk perangkat *transmitter* khususnya pada komponen seperti ESP32, modul GPS, dan LCD I2C, meskipun sumber utama berasal dari satu sel baterai Li-ion 3.7V. Modul ini dipilih karena bentuknya yang compact, ringan, dan mudah diaplikasikan, sangat cocok untuk proyek berbasis baterai seperti pada sistem *monitoring* bergerak. Selain itu, efisiensi konversi dayanya yang tinggi membantu memperpanjang masa pakai baterai tanpa menghasilkan panas berlebih.[34] Komponen ditampilkan pada Gambar 2. 19, dan spesifikasi ditampilkan dalam Tabel 2.9.

Tabel 2. 9 Spesifikasi AKK Voltage Step-Up Regulator

Spesifikasi	Keterangan
Tegangan Input	2.5V – 5V

Tegangan Output	5V Tetap
Arus Maksimum	$\pm 1A$ (tergantung suhu dan beban)



Gambar 2. 19 AKK Step-Up Regulator 5V [34]

2.13 LCD I2C 16 x 2

LCD I2C 16x2 merupakan perangkat elektronik yang digunakan untuk menampilkan informasi secara visual dalam bentuk karakter pada dua baris dengan masing-masing 16 kolom. Modul ini sering digunakan dalam sistem berbasis mikrokontroler karena efisien, mudah diprogram, dan hanya memerlukan sedikit jalur komunikasi. Tidak seperti LCD paralel biasa yang membutuhkan banyak pin digital, LCD I2C ini sudah dilengkapi dengan konverter I2C sehingga cukup menggunakan dua pin saja, yaitu SDA (*Serial Data Line*) dan SCL (*Serial Clock Line*) untuk proses integrasinya.

Modul LCD I2C 16x2 bekerja menggunakan protokol komunikasi I2C (*Inter-Integrated Circuit*), di mana mikrokontroler seperti ESP32 bertindak sebagai master dan LCD sebagai *slave*. Dengan komunikasi ini, mikrokontroler dapat mengirimkan data berupa karakter, perintah tampilan, atau pengaturan posisi kursor ke LCD hanya dengan dua kabel data. Efisiensi ini sangat membantu dalam

perancangan sistem yang memiliki keterbatasan jumlah pin GPIO, terutama jika digunakan bersama perangkat lainnya seperti GPS atau LoRa.

Selain kemudahan integrasi, LCD I2C juga memiliki konsumsi daya yang rendah dan kemampuan menampilkan berbagai informasi penting dalam sistem, seperti koordinat GPS, status zona (Aman, Waspada, Pelanggaran), kekuatan sinyal, atau waktu. Dalam sistem ini, LCD I2C digunakan sebagai media tampilan *real-time* untuk memberikan feedback langsung kepada pengguna mengenai status sistem saat proses transmisi data berlangsung antar komponen. Tampilan ini mempermudah pemantauan sistem dan pengambilan keputusan secara cepat berdasarkan informasi yang ditampilkan secara aktual.

Dengan keunggulan efisiensi koneksi, kemudahan penggunaan, dan kompatibilitas tinggi dengan mikrokontroler modern, LCD I2C 16x2 menjadi pilihan ideal dalam sistem *monitoring* ini. Berikut merupakan spesifikasi dari LCD I2C 16x2. [35]

Tabel 2. 10 Tabel Spesifikasi LCD I2C 16x2

Spesifikasi	Keterangan
Tegangan Operasi	3,3V – 5 V
Jenis Layar	LCD Alfnumerik 16 kolom × 2 baris
Konsumsi Daya	±20 mA
Alamat I2C	0x27 atau 0x3F
Ukuran Modul	± 80 mm × 36 mm × 12 mm



Gambar 2. 20 LCD I2C 16x2 [35]

2.14 Arduino IDE

Arduino IDE (*Integrated Development Environment*) merupakan perangkat lunak *open-source* yang dirancang untuk memudahkan pengguna dalam memprogram papan mikrokontroler Arduino maupun papan lain yang kompatibel seperti ESP8266 dan ESP32. Lingkungan pemrograman ini berbasis bahasa C/C++ dan menyediakan antarmuka yang sederhana, sehingga dapat diakses oleh berbagai kalangan, baik pemula maupun profesional. Arduino IDE memiliki fitur utama untuk editor kode, tombol verifikasi dan unggah, serta serial monitor yang berfungsi untuk memantau keluaran data dari perangkat. Salah satu keunggulan dari Arduino IDE adalah kemampuannya untuk menggunakan berbagai pustaka (*library*) eksternal yang memudahkan integrasi dengan sensor, modul komunikasi, dan perangkat lainnya. [36]



Gambar 2. 21 Logo Arduino [36]

Dalam pengembangan sistem berbasis lokasi, Arduino IDE dapat digunakan untuk membaca dan mengolah data dari modul GPS Neo-6M. Modul GPS mengirimkan data dalam format NMEA yang dapat diproses menggunakan library TinyGPS++. *Library* ini memungkinkan pengguna untuk memperoleh informasi posisi berupa koordinat lintang (*latitude*), bujur (*longitude*), kecepatan, arah, serta waktu berbasis UTC. Data yang diperoleh dapat digunakan untuk berbagai aplikasi, salah satunya adalah deteksi keberadaan dalam suatu wilayah geografis menggunakan metode *Geofencing*.

Geofencing adalah metode pemrograman yang membuat pagar virtual berdasarkan koordinat GPS untuk mendeteksi apakah suatu objek berada di dalam

atau di luar area yang telah ditentukan. Arduino IDE memungkinkan implementasi metode ini secara langsung dalam fungsi pemrograman, salah satunya dengan membuat fungsi yang mengevaluasi apakah data GPS termasuk dalam rentang yang ditentukan. Setelah dilakukan evaluasi posisi berdasarkan *Geofencing*, informasi tersebut dapat dikirimkan ke pusat *monitoring* melalui protokol komunikasi LoRa (*Long Range*). Arduino IDE mendukung pengendalian modul LoRa SX1278 menggunakan *library* LoRa.h dan RadioHead RFM95, yang memungkinkan pengguna untuk menginisialisasi frekuensi kerja dan mengirim data sesuai pengalamatan melalui protokol ini. Dengan menggabungkan pembacaan data GPS, evaluasi posisi menggunakan *Geofencing*, dan pengiriman data melalui LoRa, Arduino IDE menjadi platform yang ideal untuk membangun sistem pelacakan dan pemantauan lokasi secara *real-time*.

Dalam protokol komunikasi LoRa jarak jauh, perangkat berbasis Arduino IDE juga dapat difungsikan sebagai *repeater*, yaitu penguat dan penerus sinyal dari *transmitter* agar jangkauan komunikasi dapat diperluas. Selain itu, Arduino IDE juga digunakan untuk membangun perangkat *receiver*, sebagai penerima data melalui sinyal radio yang kemudian memproses atau meneruskan informasi menggunakan MQTT untuk keperluan *dashboard monitoring*, atau *database*.

2.15 Dashboard Pemantauan

Pada proyek sistem ini, arsitektur dirancang agar perangkat di lapangan bisa terhubung langsung dengan platform pemantauan berbasis local *dashboard*. Perangkat *transmitter* yang dipasang di kapal akan mengirimkan data posisi berupa koordinat GPS melalui modul LoRa menuju *receiver*. Data yang diterima kemudian dikirim lewat protokol MQTT dan dikelola oleh *broker* MQTT menggunakan bantuan HiveMq sebagai *broker* MQTT.

Broker MQTT ini berfungsi sebagai pusat distribusi data. Setiap data GPS yang masuk akan dipublikasikan ke topik tertentu. Sistem backend berbasis *script* python kemudian melakukan subscribe pada topik tersebut untuk mengambil data dan menyimpannya secara otomatis ke dalam basis data MySQL. Dengan cara ini,

setiap data baru yang dikirim dari perangkat bisa langsung tersimpan dan siap dipantau.

Untuk kebutuhan *monitoring*, sistem ini menggunakan *dashboard* berbasis HTML dan PHP yang dijalankan secara lokal dengan bantuan XAMPP. *Dashboard* ini terhubung langsung ke *database* MySQL untuk menampilkan data terbaru dalam bentuk tabel, grafik, dan peta interaktif menggunakan Leaflet.js. Posisi kapal ditampilkan secara *real-time* di peta, lengkap dengan status zona berdasarkan logika *Geofencing* (aman, waspada, atau pelanggaran). Informasi tambahan seperti waktu, kekuatan sinyal (RSSI), dan kualitas sinyal (SNR) juga ditampilkan agar pengguna bisa menilai performa komunikasi.

Dengan kombinasi teknologi LoRa untuk komunikasi jarak jauh, MQTT sebagai jalur distribusi data, MySQL untuk penyimpanan, serta *dashboard* HTML-PHP berbasis XAMPP untuk visualisasi, sistem ini dapat bekerja secara menyeluruh dari ujung ke ujung. Mulai dari pengambilan data GPS di lapangan hingga penyajian informasi ke pengguna bisa dilakukan secara efisien, *real-time*, dan mudah dipantau lewat antarmuka web lokal.