



LoRa, Wireless sensor network

Penerapan Teknologi LoRa Dalam Sistem Komunikasi *Early Warning System* Untuk Mitigasi Bencana Tsunami

Hadi Candra¹, Lathifah Arief, M.T²

^{1,2} Jurusan Teknik Komputer, Universitas Andalas, Limau Manis, Kec. Pauh, Padang, Sumatera Barat 25163 Indonesia

ARTICLE INFORMATION

Diterima Redaksi: 31 Januari 2021
Revisi Akhir: 25 September 2021
Diterbitkan Online: 31 Oktober 2021

KEYWORDS

LoRa, WSN, Dragino, EWS, Tsunami

CORRESPONDENCE

Phone: +6285272624950
E-mail: hadicandra2@gmail.com

A B S T R A C T

LoRa technology is implemented on prototype to carry out a Tsunami early warning system by sending a trigger code to sound the arduino. the sender uses Dragino LoRa LG01 outdoor while on the other side uses Dragino LoRa node and Arduino Mega. Triggers are input data from Dragino LoRa Gateway in form of string of data which is then converted and sent via LoRa communication. The data that arrives at the node will trigger the program to turn on the siren that has been keep in Arduino Mega. The application of this LoRa is to turn on the sire with a maximum distance. In this research also calculated the capacity of resource at the node that will be sent to the Gateway each time the trigger code is sent. Tests carried out in sub-urban areas with several distance parameters. The maximum distance that can be reached is 500 meters with Receive signal strong indicator (RSSI) -95. While the average time span since the trigger is sent until the siren is sounded is 0.45 seconds.

PENDAHULUAN

Negara Indonesia merupakan negara kepulauan terbesar di dunia. Indonesia memiliki 17.499 pulau dari Sabang hingga Merauke. Data dari Kementerian Kelautan Dan Perikanan Republik Indonesia, luas total wilayah Indonesia adalah 7,81 juta km² yang terdiri dari 2,01 juta km² daratan, 3,25 juta km² lautan, dan 2,55 juta km² Zona Ekonomi Eksklusif (ZEE) [1]. Terdapat 28 wilayah di Kepulauan Negara Kesatuan Republik Indonesia yang dinyatakan sebagai wilayah rawan bencana gempa bumi tektonik, gunung berapi dan tsunami. Diantaranya Nangroe Aceh Darussalam, Sumatera Utara, Sumatera Barat, Bengkulu, Lampung, Banten, Jateng, dan Daerah Istimewa Yogyakarta wilayah selatan, Jawa timur wilayah selatan, Bali, Nusa Tenggara Barat dan Nusa Tenggara Timur [2].

Pada tahun 2017 hampir 3000 kejadian bencana alam terjadi di Indonesia. Data Badan Nasional Penanggulangan Bencana (BNPB) mencatat ada 5 kejadian bencana tsunami terjadi dalam kurun waktu 10 tahun terakhir. Kejadian bencana gempa bumi dan Tsunami di Aceh pada tahun 2004 yang menelan 126.741 jiwa dan lebih dari 750.000 orang kehilangan mata pencaharian menunjukkan besarnya kerugian yang ditimbulkan oleh bencana ini[3]. Hal ini menjadi latar belakang dibentuknya Undang-undang republik Indonesia nomor 24 tahun 2007 tentang Penanggulangan Bencana.

Berdasarkan pasal 44 UU nomor 24 tahun 2007 tentang penanggulangan bencana, ada tiga langkah penyelenggaraan penanggulangan bencana dalam situasi terdapat potensi terjadi bencana yang meliputi ; (1) Kesiapsiagaan, (2) peringatan dini, dan (3) mitigasi bencana[4]. Sistem peringatan dini yang ada saat ini tertuang dalam *Indonesia Tsunami Early Warning System* (InaTEWS). Berdasarkan pedoman pelayanan peringatan dini Tsunami InaTEWS, rantai komunikasi dimulai dari Pusat Peringatan Tsunami Nasional di Badan Meteorologi Klimatologi dan Geofisika (BMKG) Jakarta yang kemudian diteruskan TV/radio nasional, kepada pemerintah daerah (pemda), Badan Nasional Penanggulangan Bencana (BNPB), Tentara Nasional Indonesia (TNI) dan Polisi Republik Indonesia (POLRI) yang kemudian secara hirarki turun ke tingkat daerah di bawah nya. Ada beberapa bentuk peringatan yang dapat diterima masyarakat beresiko tsunami yaitu dari tv/radio, dari sirine, dan dari pemda setempat[3].

Untuk menyebarkan sirine peringatan bahaya tsunami, diperlukan sebuah sistem yang bisa membawa peringatan ke setiap tempat pemukiman/keramaian yang berada dalam zona bahaya tsunami. Untuk itu, diperlukan perangkat yang dapat tetap berkomunikasi dengan keadaan darurat bencana. Penggunaan LoRa yang *Low power* dan *long range*, memungkinkan untuk diterapkan pada kondisi bencana. Selain itu dengan sistem aktivasi yang mengirimkan *trigger*, maka tidak diperlukan *bandwidth* yang besar.

Sebelumnya penelitian tentang LoRa ini dalam konteks mitigasi bencana Tsunami pernah dilakukan, di mana LoRa digunakan untuk mengirimkan data dari sensor tsunami untuk dapat dimonitoring pada server online menggunakan *ThingSpeak*[6]. Perbedaan penelitian tersebut dengan titik fokus yang dibahas kali ini adalah pada penelitian ini, sistem komunikasi dirancang untuk mengirimkan peringatan bencana tsunami yang berupa sirine.

Early Warning System

Sistem Peringatan dini (*Early Warning System*) adalah kombinasi kemampuan teknologi dan kemampuan masyarakat untuk menindaklanjuti hasil dari peringatan dini tersebut. Peringatan dini sebagai bagian dari pengurangan risiko bencana tidak hanya mengenai peringatan yang akurat secara teknis, tetapi juga harus membangun pemahaman risiko yang baik dari suatu peringatan, menjalin hubungan antara penyedia dan penggunaan peringatan, dan juga meningkatkan kemampuan otoritas dan masyarakat untuk bereaksi secara benar terhadap peringatan dini[4].

Untuk mengatur ketertipan, pemerintah sudah mengeluarkan regulasi dengan persyaratan teknis sistem peringatan dini bencana alam. Aturan ini dituangkan dalam Peraturan Menteri Komunikasi Dan Informatika Republik Indonesia Nomor 3 Tahun 2014 Tentang Persyaratan Teknis Sistem Peringatan Dini Bencana Alam Pada Alat Dan Perangkat Penerima Televisi Siaran Digital Berbasis Standar Digital *Video Broadcasting Terrestrial-Second Generation*[5].

Pelaksanaan sistem peringatan dini tsunami di Indonesia memiliki payung hukum yang sudah ditetapkan seperti Undang-Undang, Peraturan Pemerintah, Peraturan Menteri, Peraturan Kepala Badan Nasional Penanggulangan Bencana, dan Surat keputusan[4]. Dilihat dari payung hukum yang melindungi penyelenggaraan sistem peringatan dini ini, negara Republik Indonesia terlihat serius mengaturnya, namun pelaksanaan secara teknis di lapangan mungkin perlu kita awasi bersama lagi agar tidak terjadi kejadian serupa pada 2018.

Mitigasi bencana

Dalam laman situs pusat pendidikan mitigasi bencana (P2MB) Universitas Pendidikan Indonesia dikatakan Mitigasi bencana adalah serangkaian upaya untuk mengurangi risiko bencana, baik melalui pembangunan fisik, maupun penyadaran dan peningkatan kemampuan menghadapi bencana[6]. Setelah menghitung resiko, yang harus kita lakukan ialah melakukan tindakan untuk mengurangi resiko bencana tersebut.

Tsunami

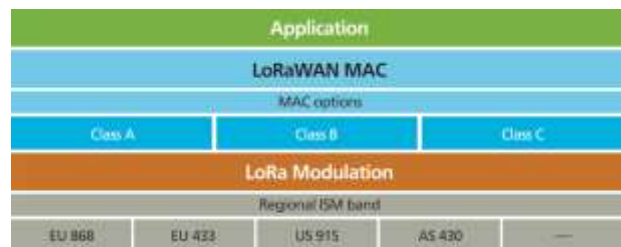
Tsunami adalah rangkaian gelombang laut yang menyerang pantai atau daratan, yang menjalar dengan kecepatan tinggi yang disebabkan oleh terjadinya gangguan yang impulsif di dasar laut[9]. Gelombang tsunami mampu menjalar dengan kecepatan hingga 900km/jam, yang terjadi di daerah episentrum gempa yang menimbulkan tsunami tersebut.

Data dari *United Nations Secretariat for International Strategy for Disaster Reduction* (UNISDR) menyebutkan, dalam paparan terhadap penduduk atau jumlah manusia yang ada di daerah yang mungkin kehilangan nyawa karena bencana, risiko bencana yang dihadapi Indonesia sangatlah tinggi. Untuk potensi bencana

tsunami, Indonesia menempati peringkat pertama dari 265 negara di dunia yang disurvei badan PBB. Resiko ancaman tsunami di Indonesia bahkan lebih tinggi dari Jepang. Dalam hitungan UNISDR, ada 5.402.239 orang berpotensi terkena dampaknya.[7]

Long Range (LoRa)

Menurut *semtech.com*, LoRa atau singkatan dari *Long Range* adalah teknik modulasi *spread spectrum* yang berasal dari teknologi *chirp spread spectrum* (CSS)[8]. Artinya sebuah metode komunikasi dimana semua sinyal komunikasi disebar di seluruh spektrum frekuensi yang tersedia. Lebarnya pita frekuensi yang digunakan, tergantung kepada teknologi yang digunakan. LoRa merupakan pengembangan tingkat lanjut sistem berbasis IoT yang rendah daya dan mampu berkomunikasi jarak jauh. Teknologi LoRa memungkinkan untuk digunakan pada berbagai bidang seperti manajemen energi, pengontrolan polusi, efisiensi infrastruktur, penanganan bencana, dan lain-lain. Teknologi LoRa dapat dimanfaatkan oleh publik, pribadi, jaringan hibrid dan penyedia layanan seperti *Internet Service Provider* (ISP) seluler.



Gambar 1. Ilustrasi LoRa[8]

Penggunaan bandwidth tinggi akan sangat dipengaruhi oleh interferensi. Untuk meningkatkan efisiensi spektral dan kapasitas jaringan, modulasi LoRa membuat enam jenis datarate berbeda dari kode *spreading factor*(SF).

LoRa adalah salah satu bagian dari *wireless Sensor Network*(WSN). *Wireless Sensor Network* (WSN) adalah kesatuan perangkat sensor yang mengumpulkan data dari alam dan mengirimkan kepada administrator[9].

Dragino

Dragino adalah proyek aplikasi OpenWRT yang memungkinkan untuk menanamkan Linux untuk proyek MCU. Dragino bersifat LowCost, menggunakan hardware motherboard linux untuk mikrokontroler. Tujuan dari proyek dragino adalah untuk memecahkan masalah konektivitas mikrokontroler dan meningkatkan produk mikrokontroler seperti arduino. Dragino dapat digunakan untuk mengendalikan robot jarak jauh, pencatatan data, aplikasi web untuk presentasi data, jaringan mesh melalui wifi dan lain-lain. Dragino memiliki perangkat lunak terbuka dan desain perangkat keras terbuka yang dirilis dibawah lisensi *Creative Commons*.



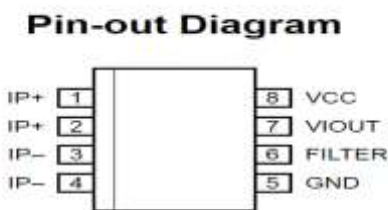
Gambar 2. Dragino

Mikrokontroler

Mikrokontroler merupakan sebuah mikrokomputer yang berfungsi sebagai kendali dan otomatisasi mesin dan proses. Mikrokontroler adalah sebuahn sistem komputer yang terintegrasi pada satu chip mikroprosesor, memiliki memory dan perangkat interface dengan jumlah tertentu[10]

Sensor ACS712

Sensor ACS712 merupakan sensor yang digunakan untuk mengukur arus pada listrik AC maupun DC. Berdasarkan datasheet sensor ini dikatakan sensor yang murah dan presisi. Selain itu juga mudah untuk diaplikasikan[11]. Arus yang mengalir melalui jalur konduksi tembaga pada sensor menghasilkan medanmagnet yang bisa dibaca oleh IC kemudian diubah menjadi tegangan proporsional. Akurasi perangkat dioptimalkan melalui kedekatan sinyal magnetik dengan tranduser. Berikut diagram pin sensor ACS712 :



Gambar 3. Diagram pin sensor ACS712

METODE

Untuk dapat memahami kebutuhan sistem secara rinci, berikut akan diuraikan menjadi dua jenis yaitu kebutuhan fungsional sistem dan kebutuhan non fungsional sistem.

Analisa Kebutuhan Fungsional Sistem

Kebutuhan fungsional sistem adalah segala yang berkaitan dengan fungsi atau tujuan akhir dari sistem yang akan dibangun. Pada bagian ini akan berisi proses-proses yang kemudian akan dilakukan oleh sistem. Proses-proses tersebut antara lain :

1. Sistem harus mampu mendistribusikan data trigger dari laptop hingga menghasilkan sirine pada node dan arduino.
2. Sistem harus mampu berjalan pada keadaan LoS maupun tidak LoS (*line of sigh*).

Analisa Kebutuhan non Fungsional Sistem

Kebutuhan non-fungsional sistem adalah kebutuhan yang didalamnya terdapat faktor yang mempengaruhi kerja sistem yaitu diantaranya :

1. Faktor hambatan, yang berarti ada atau tidaknya hambatan secara langsung.
2. Faktor Jarak, yang berarti sejauh mana kedua device dapat berkomunikasi
3. Faktor Interferensi, yaitu tempat penelitian dilakukan pada lingkungan rawan interferensi dan pada daerah rural.

Perancangan Sistem

Secara umum rancangan sistem akan berbentuk skema berikut :



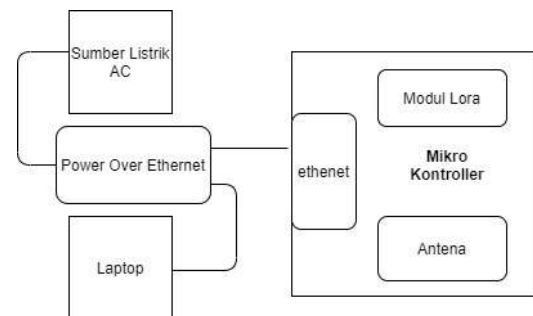
Gambar 4. Rancangan Sistem

Pada gambar 4, LoRa gateway yang didalamnya terdapat arduino dapat dikontrol oleh pihak yang mempunyai otoritas dari jarak jauh menggunakan jaringan internet. Pada LoRa node terdapat arduino yang akan memproses data yang dipicu oleh pengirim yang akan diterjemahkan kedalam kode yang dapat memutar file suara pada arduino dan melewati pengeras suara. File suara disimpan pada arduino. File suara yang mewakili sirine akan terhenti saat kode trigger diterima arduino. Lora gateway akan diusahakan berada pada lokasi yang ditetapkan sebagai daerah aman tsunami. Pada sistem yang akan dirancang, digunakan hanya satu node. Hal ini karena gateway yang digunakan hanya mendukung untuk koneksi satu node.

3.2.1 Perancangan Perangkat Keras

Rancangan perangkat keras yang digunakan pada sisi gateway akan berbeda dengan rancangan perangkat keras pada sisi node. Berikut perbedaannya:

1. Rancangan perangkat keras pada sisi gateway.

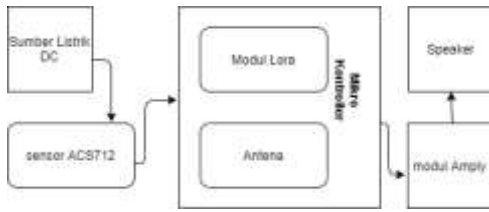


Gambar 5. Rancangan Perangkat Keras pada Gateway

Pada gambar 5, Dragino LoRa gateway diberi catu daya melewati power over ethernet. Tujuan digunakannya power over ethernet ini adalah agar dapat melewatkan data dan power dalam satu kabel lan melalui port ethernet. Selanjutnya perangkat bisa hidup dan diakses melalui laptop. Pada gateway ini, modul lora sudah terpasang langsung didalam paket perangkat dengan arduino yun.

Begitu juga antenna yang digunakan adalah antenna bawaan dari dragino.

2. Rancangan perangkat keras pada sisi node



Gambar 6. Rancangan Perangkat Keras pada Node

Pada sisi node, perangkat diberi catudaya dari baterai aki 12 volt. Kabel dari aki yang masuk menuju mikrokontroler dipasang sensor ACS712 untuk mengukur berapa arus yang terpakai oleh perangkat. Selanjutnya pada board mikrokontroler dipasang modul dragino LoRa node dan antenna. Modul amply dipasang sebagai penguat bagi speaker.

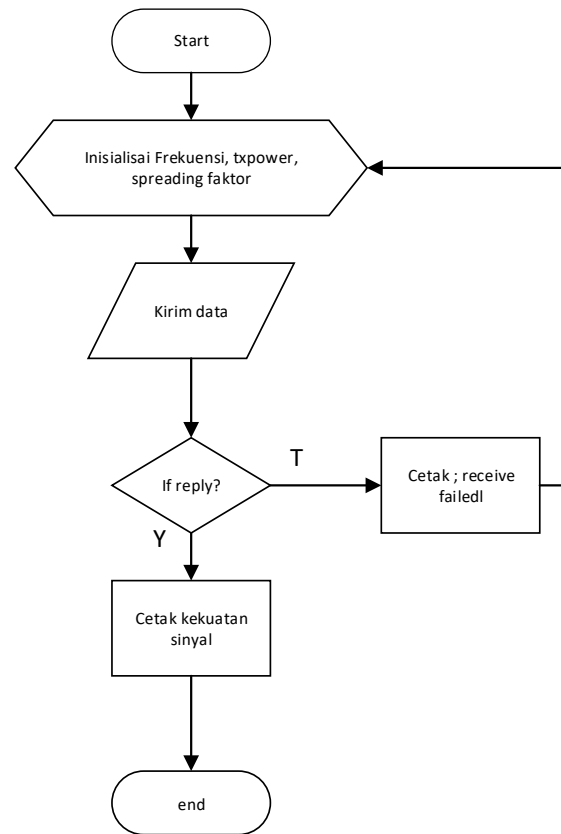
Perancangan Perangkat Lunak

Perangkat lunak yang digunakan pada penelitian ini adalah ; openwrt yang sudah include dalam dragino lora yang didalamnya kita dapat melakukan konfigurasi pada gateway. Kegunaan dan fitur dalam perangkat lunak ini tidak akan dijabarkan lebih lanjut. Hal yang paling penting dalam rancangan ini adalah kode program pada kedua sisi yang akan berkomunikasi.

Pada gateway akan dirancang program yang dapat mengirim trigger berupa bit kode tertentu yang nantinya akan menjadi pemicu untuk memutar file suara pada lora node. Hal ini sengaja dilakukan demi menghemat penggunaan bandwidth lora yang sangat terbatas. Sehingga walaupun dengan bandwidth yang terbatas, komunikasi dan tujuan utama rancangan tetap dapat berjalan dengan baik. Bahasa program yang digunakan adalah bahasa C yang terdapat dalam arduino IDE.

Perancangan Proses

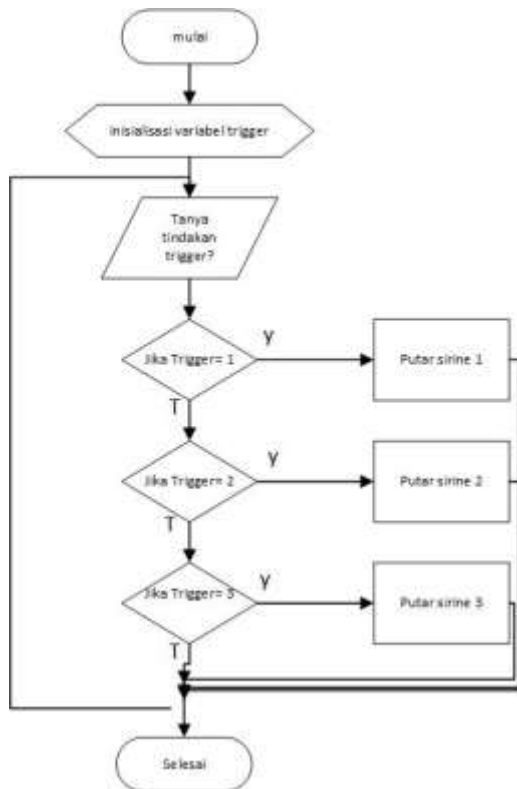
Pada perancangan proses, komunikasi diawali dengan inisialisasi jaringan. Ada dua jenis jaringan yang digunakan yaitu; 1. Jaringan menggunakan protokol TCP/IP dan jaringan menggunakan protokol lora (RF95). Inisialisasi jaringan pada lora gateway diatur pada router. Sedangkan jaringan antar lora akan dijelaskan pada flowchart berikut :



Gambar 7 Flowchart jaringan antar lora

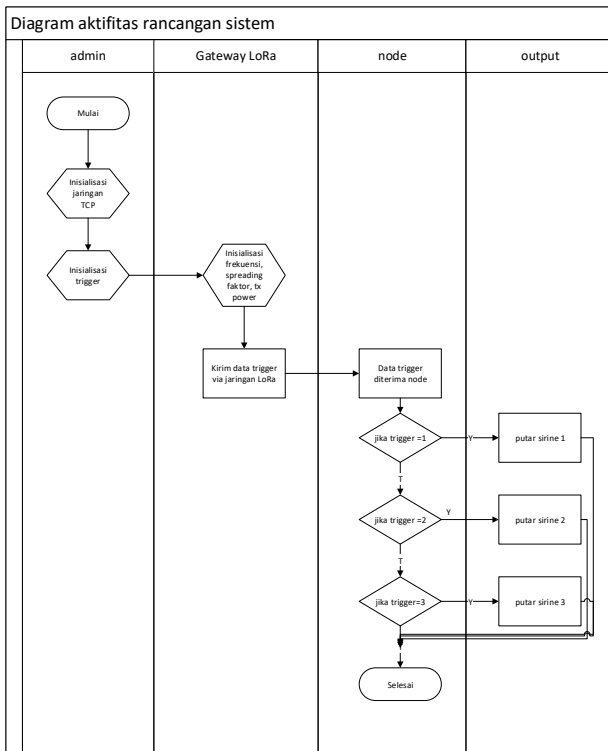
Saat awal dihidupkan, dragino lora akan menginisialisasi frekuensi, daya transmit, dan spreading faktor sesuai dengan yang telah ditetapkan pada baris program. Nilai tersebut akan dipancarkan oleh gateway dan jika ada balasan dari node, maka gateway akan meminta untuk menampilkan kekuatan sinyal dari node, jika tidak artinya tidak ada node terhubung.

Dengan kondisi kedua perangkat telah terhubung, maka beberapa proses dapat dilakukan yaitu untuk distribusi data pemicu (*trigger*). Berikut adalah flowchart pendistribusian data trigger hingga menghasilkan output suara.



Gambar 8. Flowchart aktivasi trigger

Secara keseluruhan, rancangan proses dapat dilihat pada diagram aktivitas berikut :



Gambar 9. Diagram aktifitas proses

HASIL DAN PEMBAHASAN

Penerapan teknologi LoRa dalam sistem komunikasi *Early warning system* untuk mitigasi bencana tsunami ini diimplementasikan sesuai dengan perancangan sistem. Implementasi sistem ini terdiri dari implementasi perangkat

keras, implementasi perangkat lunak, percobaan secara keseluruhan dan analisa hasil pengukuran. Pengujian dan pengukuran dilakukan untuk mengetahui apakah sistem yang telah dirancang sudah bekerja dengan baik dan dapat menganalisa performa perangkat *prototype* yang dirancang.

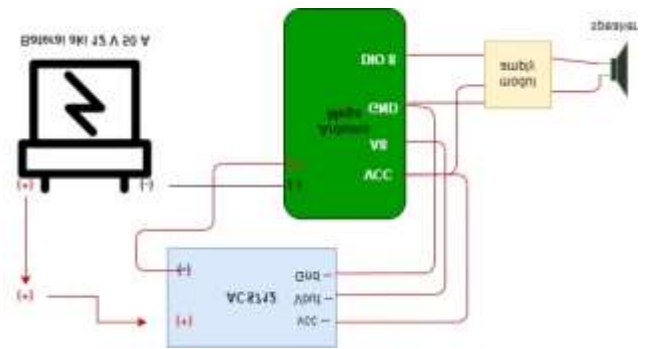
Implementasi Perangkat Keras

Implementasi perangkat keras adalah proses merangkai semua komponen-komponen yang diperlukan oleh sistem, seperti arduino, modul LoRa, *speaker* serta komponen catu daya seperti kabel dan sumber daya.



(a)

(b)



(c)

Gambar 10. (a) Perangkat keras dilihat dari atas (b) Perangkat keras dilihat dari samping (c) skematik perangkat keras pada node

Pada Gambar 4.1 memperlihatkan implementasi perangkat keras dari sistem yang dirancang. Perangkat keras sistem tersebut terdiri dari satu unit Dragino LoRa gateway yang berfungsi sebagai sumber pengirim dan berada di sisi pengguna, modul LoRa node sebagai klien, arduino Mega sebagai mikrokontroler yang akan memproses triger, Speaker, modul ampli, dan catudaya. Dragino LoRa akan terhubung ke komputer pengguna yang menggunakan aplikasi arduino ide dan serial monitor. Perangkat ini terhubung dengan kabel Lan yang sekaligus melewati daya menggunakan PoE. Dragino LoRa gateway yang di dalamnya terdapat arduino yun, bisa diakses menggunakan jaringan Lan.

Untuk mengukur arus yang terpakai sehingga dapat dihasilkan jumlah konsumsi daya pada perangkat. Untuk menghitung daya terpakai menggunakan rumus $P = V \cdot I$ di mana V merupakan tegangan input (5Volt) dan I adalah arus.

Pada sisi gateway ada beberapa perangkat yang digunakan. Dragino lora gateway sudah termasuk didalam nya arduino yun, antena, dan beberapa port ethernet.



Gambar 11. Implementasi perangkat keras pada sisi Gateway

Pada gambar di atas terdapat komponen penting seperti adaptor, Power over ethernet (PoE), kabel lan , dragino Lora dan antena. Penggunaan power over ethernet adalah untuk melewatkan data dan arus menuju perangkat dragino sehingga perangkat *outdoor* bisa hidup tanpa harus memasang instalasi listrik AC di tempat perangkat terpasang. Adaptor yang digunakan memiliki output 12 volt DC dan 1.0 ampere arus. Keluaran pada power over ethernet ada 2 tipe. Port PoE adalah untuk kabel lan menuju perangkat Dragino LoRa, sedangkan port ethernet adalah untuk koneksi ke komputer/laptop.

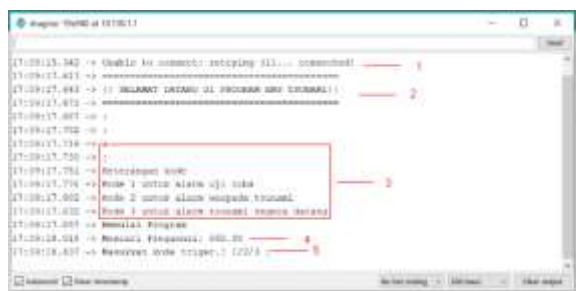
Implementasi Perangkat Lunak

Implementasi perangkat lunak adalah proses pemrograman untuk perangkat keras yang telah dirancang agar sistem dapat berfungsi dengan baik. Pemrograman dilakukan pada kedua perangkat disisi Gateway dan node yang menggunakan arduino Yun dan mega dan menggunakan bahasa pemrograman C pada arduino IDE. Output program pada gateway dan node berupa tampilan pilihan menu dalam serial monitor. Pilihan untuk *trigger* terdiri dari 4 pilihan sebagai berikut :

Tabel 4.1 kode trigger pada menu utama :

Kode trigger	Keterangan	Output
1	Alarm uji coba	Bunyi beep pendek 1 detik dengan frekuensi rendah
2	Waspada Tsunami	Bunyi beep selama 5 detik dengan frekuensi lebih tinggi dari kode 1
3	Tsunami segera datang	Bunyi sirene terus menerus tanpa henti
4	Untuk menonaktifkan sirene	Tidak ada bunyi sirene

Program hanya dapat berjalan pada komputer/laptop yang sudah terinstall arduino IDE dan mempunyai kode sumber program yang sesuai dengan masing-masing perangkat. Berikut adalah tampilan serial monitor untuk perangkat gateway :



Gambar 12. Implementasi perangkat lunak pada gateway

Pada gambar 4.3 terlihat tampilan serial monitor pada sisi gateway yang mempunyai beberapa bagian. Berikut uraian penjelasan gambar di atas :

1. Pada bagian ini adalah proses inialisasi radio frekuensi, dimana pada saat ini sistem akan memeriksa apakah ada perangkat sejenis agar bisa terhubung.
2. Adalah bagian judul yang ditampilkan agar tampilan mudah dipahami pengguna.
3. Pada bagian 3 ini dibuatkan panduan berupa keterangan agar pengguna bisa memilih jenis trigger yang akan diuji.
4. Pada bagian ini adalah proses menerima data dari perangkat dengan frekuensi yang sama.
5. Pada bagian 5 ini pengguna diminta untuk memilih jenis kode trigger yang akan diinputkan. Keterangan trigger bisa dilihat pada bagian 3.

Penjabaran di atas adalah pada tampilan pertama perangkat lunak. Jika kemudian dilakukan pemilihan kondisi seperti panduan sebelumnya, maka akan keluar tampilan seperti berikut :



Gambar 13. Implementasi perangkat lunak pada menu gateway

Gambar di atas adalah tampilan lanjutan dari pemilihan kode diantara 3 kondisi yang tersedia. Serial monitor akan membaca input dari keyboar dan melakukan perbandingan dengan kode yang sudah ditetapkan. Berikut dijelaskan gambar 4.2 :

1. Nilai yang diinputkan dari keyboar dibaca dan kemudian ditampilkan pada serial monitor.
2. Menampilkan pesan dari node yang merupakan identitas node.
3. Menampilkan indikator kekuatan sinyal yang diterima *Receive Signal Strength Indicator*(RSSI).
4. Mengirim balasan untuk node berupa data string.

Pada setiap proses akan ditampilkan waktu *real time* sehingga bisa dilakukan pengukuran berapa lama data sampai dan diproses hingga menghasilkan suara.

Pemrograman untuk menampilkan tampilan serial monitor pada gateway terlihat seperti gambar berikut :

```
void setup()
{
  pinMode(LED, OUTPUT);
  Bridge.begin(BAUDRATE);
  Console.begin();
  Console.println("=====");
  Console.println("|| SELAMAT DATANG DI PROGRAM EMS TSUBANI ||");
  Console.println("=====");
  Console.println(":");
  Console.println(":");
  Console.println(":");
  Console.println("Keterangan kode");
  Console.println("Kode 1 untuk alarm uji coba");
  Console.println("Kode 2 untuk alarm waspada tsunami");
  Console.println("Kode 3 untuk alarm tsunami segera datang");
}
```

Gambar 14. Kode Sumber tampilan pada gateway

Triger yang akan di kirim pada node adalah berasal dari masukkan pada gateway. Berikut kode sumber untuk menunggu input pengguna pada serial monitor:

```
if (rf95.available())
{
  Console.println("Masukkan kode triger. 1/2/3 : ");
  while (Console.available() != 0)
  {
    //menunggu input user
  }
  triger=Console.readString(); //membaca string dari port serial
  Console.println("kode yang anda masukkan adalah : "+triger);
  Console.println ("mengirim kode pada node. . .");
}
```

Gambar 15. Kode Sumber Pembacaan input

Nilai input yang terbaca adalah jenis tipe data String. Untuk mengirim triger yang mempunyai tipe data integer, maka perlu dilakukan konversi dari tipe data string menjadi tipe data integer. Kode sumber untuk melakukan konversi adalah sebagai berikut :

```
String test = triger;
int testData = test.toInt();

if(testData==1)
{
  uint8_t data[] = "1";
  rf95.send(data, sizeof(data));
  rf95.waitPacketSent();
  Console.println("");
  Console.println("=====");
}
else if(testData==2)
{
  uint8_t data[] = "2";
  rf95.send(data, sizeof(data));
  rf95.waitPacketSent();
  Console.println("");
  Console.println("=====");
}
else if(testData==3)
{
  uint8_t data[] = "3";
  rf95.send(data, sizeof(data));
  rf95.waitPacketSent();
  Console.println("");
}
```

Gambar 16. Proses konversi data dan pengiriman

Pada gambar kode sumber di atas, setelah data string dikonversi menjadi integer, nilai integer dimasukkan pada sebuah array yang mempunyai tipe data uint8_t (unsigned integer 8 bit). Hal ini mengikuti aturan yang ada pada library rf95 yang digunakan LoRa.

Implementasi perangkat lunak pada sisi node berbeda dengan perangkat lunak yang ada pada sisi gateway. Pada sisi node, perangkat lunak tidak hanya untuk membuat tampilan di serial monitor dan berkomunikasi dengan gateway, tetapi juga untuk memproses data sensor dan mengeluarkan output suara pada speaker. Berikut tampilan serial monitor di sisi node :



Gambar 17. Tampilan serial monitor pada node

Pada sisi node perangkat lunak melakukan perhitungan perkiraan ketahanan baterai yang kemudian akan dikirim menuju gateway. Data didapat dari hasil kalkulasi nilai sensor arus ACS712 20 A. Hasil pembacaan sensor arus adalah nilai dalam satuan ampere. Untuk mendapatkan nilai ketahanan baterai, kapasitas baterai dalam keadaan penuh didefinisikan terlebih dahulu. Kemudian dikurangi dengan nilai arus yang mengalir dalam satuan detik.

```
if((current) > cutOffLimit ){
  kapasitas = kapasitasT - current;
  kapasitasT = kapasitas;
  t = kapasitasT / current;
  m = t/3600;
}
```

Gambar 18. Program menghitung ketahanan baterai

Pada gambar di atas, kapasitas baterai dalam keadaan penuh dikurangi dengan arus yang mengalir sehingga didapatkan nilai kapasitas dalam waktu tertentu. Kapasitas ampere kemudian dibagi dengan nilai arus yang terbaca sehingga menghasilkan waktu untuk ketahanan baterai dalam satuan detik. Untuk menampilkan waktu dalam satuan jam dilakukan pembagian dengan angka 3600.

Pengujian Sensor Arus

Pengujian sensor arus dilakukan untuk melihat kemampuan sensor dalam membaca arus listrik yang terpakai oleh perangkat. Sensor ACS712 bekerja dengan menghitung tegangan keluaran pada sensor yang kemudian dilakukan pembagian dengan tingkat sensitifitas sensor sehingga didapatkan nilai arus. Proses pengolahan ini dilakukan oleh mikrokontroler yang kemudian akan ditampilkan pada serial monitor.

```
15:43:29.599 -> V: 0.014V, I: 0.14A
15:43:29.633 -> ketahanan baterai: 346 Jam
```

Gambar 19. Tampilan Pengukuran Arus

Pengujian akan dilakukan dengan membandingkan data sensor ACS712 dengan data arus yang diukur dengan multimeter secara manual. Data akan ditampilkan pada tabel 4.1 berikut :

Tabel 4.2 Pengujian Sensor ACS712

Percobaan	Sensor ACS712	Multimeter	Selisih	Persentase error (%)
1	0.095	0.09	0.005	5.5

2	0.046	0.04	0.006	15
3	0.144	0.2	0.056	28
4	0.09	0.1	0.01	10
5	0.046	0.05	0.04	8
Error rata-rata (%)				13.3%

Untuk mengetahui nilai persentase error dilakukan dengan rumus berikut :

$$\% \text{ error} = \frac{\text{pengukuran multimeter} - \text{sensor arus}}{\text{pengukuran multimeter}} \times 100\%$$

Berdasarkan data pada tabel di atas terdapat selisih pembacaan antara sensor ACS dan multimeter. Selisih yang didapat tidak terlalu besar sehingga persentase error hanya 13.3% dan tingkat keberhasilan pengukuran arus oleh sensor ACS712 adalah sebesar 86.7%.

Pengujian modul LoRa

Pengujian modul LoRa dilakukan untuk mengetahui perangkat dapat berkomunikasi satu sama lain. Pengujian dilakukan jarak dekat di atas meja kerja. Pengujian dilakukan dengan mengirim pesan dari Dragino LoRa gateway menuju LoRa node dan kemudian LoRa node membalas pesan dan diterima kembali oleh gateway. Adapun tabel pengujian modul LoRa adalah sebagai berikut :

Tabel 4.3 Pengujian Modul LoRa

Percobaan	Dragino LoRa Gateway		Dragino LoRa Node	
	Transmit	Receive	Transmitt	Receive
1	berhasil	berhasil	berhasil	berhasil
2	berhasil	berhasil	berhasil	berhasil
3	berhasil	berhasil	berhasil	berhasil
4	berhasil	berhasil	berhasil	berhasil
5	berhasil	berhasil	berhasil	berhasil

Percobaan dilakukan sebanyak lima (5) kali dengan keterangan semua berhasil dilakukan. Untuk mengukur persentase keberhasilan, dihitung dengan rumus berikut :

$$\% \text{ berhasil} = \frac{\text{Banyak percobaan yang berhasil}}{\text{Banyak percobaan dilakukan}} \times 100$$

Sehingga persentase keberhasilan uji coba modul LoRa adalah seratus persen (100%).

Pengujian dan analisa perangkat lunak

Pengujian perangkat lunak dilakukan dengan mengamati output program pada serial monitor. Pengamatan dilakukan pada kedua sisi node dan gateway. Pada sisi gateway, program menerima input dari keyboard yang berupa string. Data string lalu dikonversi menjadi *unsigned integer* agar bisa terkirim melewati modul rf95. Untuk menguji program ini, bisa diketahui dari sirine yang berbunyi pada node.

Pengujian perangkat lunak pada node bisa diketahui dari berbunyinya sirine, kemudian data sensor dapat dibaca dan dikirim menuju gateway.

Pengujian dan analisa sistem

Pada bagian ini dilakukan pengujian terhadap sistem yang telah dibangun. Pengujian dapat dilakukan jika implementasi perangkat keras dan perangkat lunak telah dilakukan. Pengujian dilakukan agar dapat mengetahui performa dari sistem yang telah dibuat.

Pengujian pada daerah sub urban

Pengujian ini dilakukan untuk mengetahui peforma sistem yang telah dibuat pada daerah Sub urban.

Tabel 4.4 Pengujian performa sistem pada daerah Sub urban (kampus Universitas Andalas , Limau Manih padang).

no	Jarak (m)	RSSI	Sirine berbunyi	Waktu tunggu (detik)	Kode sesuai dengan sirine
1	10	-34	Iya	0.36	Iya
2	20	-47	Iya	0.48	Iya
3	50	-55	Iya	0.43	Iya
4	100	-60	Iya	0.45	Iya
5	200	-67	Iya	0.34	Iya
6	300	-83	Iya	0.48	Iya
7	400	-94	Iya	0.54	Iya
8	500	-95	Iya	0.50	Iya
9	600	-	Tidak	Tidak diketahui	Tidak diketahui
10	700	-	Tidak	Tidak diketahui	Tidak diketahui
11	800	-	Tidak	Tidak diketahui	Tidak diketahui
12	900	-	Tidak	Tidak diketahui	Tidak diketahui
13	1000	-	Tidak	Tidak diketahui	Tidak diketahui

Berdasarkan data pada tabel di atas, jarak maksimal perangkat dapat berkomunikasi pada penelitian ini adalah 500 meter. Perangkat dragino lora gateway diletakkan di lapangan sedangkan lora node berpindah tempat sesuai dengan jarak pada tabel. Pada tabel di atas, ketika perangkat berkomunikasi yang ditandai dengan adanya keluaran nilai RSSI. Pada keadaan ini, semua percobaan berhasil membunyikan sirine dan sesuai dengan nilai yang diinputkan. Jarak waktu antara pengiriman *triger* dengan berbunyinya sirine pada percobaan di atas tidak lebih dari satu detik. Rata-rata waktu tunggu dalam keadaan perangkat merespon *trigger* adalah sebesar 0.45 detik.

Adapun jarak maksimal yang didapat pada percobaan ini adalah 500 meter dengan keadaan lingkungan uji seperti gambar berikut :



Gambar 20. Area penelitian dilihat dari map

KESIMPULAN

Pada penelitian ini telah dirancang *early warning system* dengan menerapkan teknologi LoRa dengan frekuensi 868 MHZ. Sistem peringatan berupa sirine yang berbunyi pada bagian node yang

dipicu dengan meng-inputkan data manual berupa angka pada laptop. Angka-angka yang menjadi kode didefinisikan dulu pada program sehingga sirine bisa berbunyi sesuai dengan kode yang telah ditetapkan.

Transmisi data trigger dari gateway menuju node dilakukan dengan mengubah input data yang berupa string menjadi integer, kemudian dikonversi lagi menjadi tipe data *unsigned integer 8 bit* (Uin8_t). Karena pada library rf95 modul lora, mekanisme pengiriman data menggunakan tipe data uin8_t.

Pilihan simulasi dapat diputar menjadi sirine dengan membandingkan kode yang dikirim dengan kode yang didefinisikan untuk memutar sirine.

Ketersediaan catudaya pada node dapat diketahui dengan menggunakan sensor ACS712. Sensor ini membaca nilai tegangan yang melewati sensor kemudian dilakukan perhitungan dengan membagi nilai tegangan dengan tingkat sensitifitas sensor. Pengiriman nilai arus dilakukan setiap kali trigger dikirim. Kapasitas catu daya didapat dengan mengurangi kapasitas arus baterai pada saat awal digunakan dengan arus terpakai.

REFERENSI

- [1] Roza, Elviana. 2017. *Maritim Indonesia Kemewahan yang Luar biasa*. Kkp.go.id
- [2] Badan Nasional Penanggulangan Bencana (BNPB). *Data Informasi Bencana Indonesia (DIBI)*. 2019.
- [3] BMKG. InaTEWS. 2012. *Pedoman pelayanan peringatan dini tsunami*. Jakarta : Badan Meteorologi Klimatologi & Geofisika.
- [4] Dzaky Faishal, Fariz . 2019 . *Implementasi LoRa pada Sistem Peringatan Dini Tsunami*. Bandung. Universitas Telkom.
- [5] Kemenkominfo. 2014. *Peraturan Menteri Komunikasi Dan Informatika Republik Indonesia Nomor 3 Tahun 2014 Tentang Persyaratan Teknis Sistem Peringatan Dini Bencana Alam Pada Alat Dan Perangkat Penerima Televisi Siaran Digital Berbasis Standar Digital Video Broadcasting Terrestrial-Second Generation*. Permenkominfo. Jakarta: Kementrian komunikasi dan Informatika.
- [6] Pusat pendidikan mitigasi bencana (P2MB) UPI. 2010. *Mitigasi Bencana*. P2MB Geografi universitas Putera Indonesia.
- [7] Badan Nasional Penanggulangan Bencana. *Data UNISDR*. 2018
- [8] Semtech Lora Technology. 2018. “*what-LoRa-table-illustration-web.gif*”, <https://www.semtech.com/uploads/images/what-LoRa-table-illustration-web.gif>, diakses pada 5 September 2018 pukul 10.50 WIB.
- [9] wiwi, muhammad, Wardi, W., & Agussalim, A. (2020, March 30). Peningkatan Network Lifetime Menggunakan Cluster Based Pada Wireless Sensor Network. *JITCE (Journal of Information Technology and Computer Engineering)*, 4(01), 16-21. <https://doi.org/https://doi.org/10.25077/jitce.4.01.16-21.2020>
- [10] Andesta, D., & Ferdian, R. (2018, September 29). Sistem Keamanan Sepeda Motor Berbasis Mikrokontroler dan Modul GSM. *JITCE (Journal of Information Technology and Computer Engineering)*, 2(02), 51-63. <https://doi.org/https://doi.org/10.25077/jitce.2.02.51-63.2018>
- [11] Allegro MicroSystems. 2007. *Fully Integrated, Hall Effect-Based Linear Current Sensor with 2.1 kVRMS Voltage Isolation and a low-Resistance Current Conductor*. Allegro MicroSystems, Inc. Worcester . Massachusetts.