



***PROTOTYPE SISTEM KONTROL IRIGASI SAWAH
MENGUNAKAN KOMUNIKASI LONG RANGE (LORA)
BERBASIS ANDROID***

**“PERANCANGAN MIKROKONTROLER SISTEM KONTROL
IRIGASI SAWAH”**

**Diajukan sebagai salah satu syarat memperoleh gelar Diploma
Tiga**

Raudhatul Jannah

200332024

PROGRAM STUDI TELEKOMUNIKASI

JURUSAN TEKNIK ELETRO

POLITEKNIK NEGERI JAKARTA

2023



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**PROTOTYPE SISTEM KONTROL IRIGASI SAWAH
MENGUNAKAN KOMUNIKASI *LONG RANGE (LORA)*
BERBASIS *ANDROID***

**“PERANCANGAN MIKROKONTROLER SISTEM KONTROL
IRIGASI SAWAH”**

**Diajukan sebagai salah satu syarat memperoleh gelar Diploma
Tiga**

**POLITEKNIK
NEGERI
Raudhatul Jannah
JAKARTA
200332024**

PROGRAM STUDI TELEKOMUNIKASI

JURUSAN TEKNIK ELETRO

POLITEKNIK NEGERI JAKARTA

2023



HALAMAN PERNYATAAN ORISINALITAS

Tugas Akhir ini adalah hasil karya saya sendiri dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Raudhatul Jannah

NIM : 2003332024

Tanda Tangan :



Tanggal : 3 Agustus 2023

**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LEMBAR PENGESAHAN TUGAS AKHIR

Tugas Akhir diajukan oleh :

Nama : Raudhatul Jannah

NIM : 2003332024

Program Studi : Telekomunikasi

Judul Tugas Akhir : *Prototype* Sistem Kontrol Irigasi Sawah Menggunakan Komunikasi *Long Range* (LoRa) Berbasis Android

Telah diuji oleh tim penguji dalam Sidang Tugas Akhir pada 4 Agustus 2023
dan dinyatakan LULUS.

Pembimbing : Benny Nixon, S.T., M.T.
NIP 196811072000031001

Depok, 31 Agustus 2023

Disahkan Oleh

Ketua Jurusan Elektro



Rika Novita Wardhani, ST., MT

NIP 197011142008122001



KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Diploma tiga Politeknik.

Tugas Akhir ini berjudul “*Prototype* Sistem Kontrol Irigasi Sawah Menggunakan Komunikasi *Long Range* (LoRa) Berbasis Android”. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai penyusunan tugas akhir ini, sangatlah sulit bagi penulis untuk menyelesaikan tugas akhir ini. Oleh karena itu penulis mengucapkan terima kasih kepada :

1. Bapak Benny Nixon, S.T.,M.T dan ibu Yenniwati Rafsyam, SST.,M.T. selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran dalam mengerahkan penulis dalam penyusunan tugas akhir ini;
2. Seluruh dosen serta karyawan di Prodi Telekomunikasi yang telah mendidik dan membantu dalam pembuatan dan penyusunan Tugas Akhir;
3. Orang tua dan keluarga penulis yang telah memberikan bantuan dukungan material dan moral;
4. Siti Azzahra Ramadan selaku partner TA yang telah membantu dalam proses penyelesaian tugas akhir ini sampai selesai;
5. Ridho Pratama yang selalu memberi motivasi, dan dukungan untuk menyelesaikan tugas akhir ini.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga Tugas Akhir ini membawa manfaat bagi pengembangan ilmu.

Depok, Agustus 2023

Penulis

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Prototype Sistem Kontrol Irigasi Sawah Menggunakan Komunikasi Long Range (LoRa) Berbasis Android

“Perancangan Sistem Kontrol Irigasi Sawah Menggunakan Komunikasi Long Range (LoRa)”

ABSTRAK

Sistem kontrol irigasi sawah merupakan sistem yang dirancang untuk mengontrol pengairan pada persawahan dan memberikan peringatan kepada masyarakat apabila air di petak persawahan sudah mengering atau melebihi kapasitas pada saat cuaca tidak menentu. Sistem ini menggunakan komunikasi LoRa, sensor water level, sensor ultrasonik, RTC DS3231, motor servo, LED dan LCD. Sistem ini akan mengaliri air pada petak persawahan sesuai dengan waktu yang telah diinput pada aplikasi android. Data akan didapatkan ketika user menginput target waktu pintu air akan terbuka dan data tersebut akan dikirim oleh LoRa transmitter ke LoRa receiver. Sistem irigasi sawah menggunakan jaringan komunikasi LoRa dimana pada sisi transmitter mampu mengirimkan data kepada sisi receiver pada kondisi LoS tanpa penghalang sejauh 10 meter dengan nilai RSSI yang baik yaitu -75 dBm. Dan pada jarak maksimum yaitu sejauh 200 meter dengan nilai RSSI yang sangat buruk yaitu -108 dBm. Hasil pengujian jarak pengiriman data dengan adanya penghalang tidak jauh berbeda dengan yang tanpa penghalang, dimana pada jarak 10 meter mendapatkan nilai RSSI yang baik sebesar -80 dBm dan pada jarak maksimum yaitu 200 meter mendapatkan nilai RSSI yang sangat buruk yaitu sebesar -113 dBm. Semua pengujian ini dapat dikatakan baik karena LoRa dapat mengirimkan data dengan baik tanpa kerusakan dan juga jarak yang lumayan jauh meskipun mendapatkan nilai RSSI yang buruk.

Kata Kunci : Irigasi, Long Range, LoRa, Sawah, RSSI

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Prototype of Paddy Irrigation Control System Using Long Range Communication (LoRa) Based on Android

“Design of Paddy Irrigation Control System Using Long Range Communication (LoRa)”

ABSTRACT

The rice field irrigation control system is a system designed to control irrigation in rice fields and provide a warning to the public if the water in the rice fields is dry or exceeds capacity during uncertain weather. This system uses LoRa communication, water level sensor, ultrasonic sensor, RTC DS3231, servo motor, LED and LCD. This system will flow water to the rice fields according to the time that has been input on the Android application. Data will be obtained when the user inputs the target time the floodgate will open and the data will be sent by the LoRa transmitter to the LoRa receiver. The rice field irrigation system uses a LoRa communication network where the transmitter side is able to send data to the receiver side at LoS conditions without obstructions as far as 10 meters with a good RSSI value of -75 dBm. And at a maximum distance of 200 meters with a very bad RSSI value of -108 dBm. The results of testing the distance of sending data with obstacles are not much different from those without obstacles, where at a distance of 10 meters you get a good RSSI value of -80 dBm and at a maximum distance of 200 meters you get a very bad RSSI value of -113 dBm. All of these tests can be said to be good because LoRa can transmit data properly without damage and also quite a long distance even though it gets a bad RSSI value.

Keywords : Irrigation, Long Range, LoRa, Rice Fields, RSSI

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

- 1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
- 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Daftar Isi

	Halaman
HALAMAN PERNYATAAN ORISINALITAS	iii
LEMBAR PENGESAHAN TUGAS AKHIR	Error! Bookmark not defined.
KATA PENGANTAR.....	v
ABSTRAK	vi
Daftar Isi	viii
Daftar Gambar	xi
Daftar Tabel.....	xii
Daftar Lampiran	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan.....	2
1.4 Luaran.....	2
BAB II TINJAUAN PUSTAKA.....	3
2.1 Irigasi Sawah	3
2.2 LoRa (Long Range).....	3
2.3 Hope RFM95W LoRa	3
2.4 Sensor Ultrasonik HC-SR04	5
2.5 Real Time Clock (RTC)	6
2.6 Motor Servo.....	6
2.6 ESP32	7
2.7 Liquid Crystal Display I2C 20x4	8
2.8 Light Emitting Diode (LED)	8
2.9 Relay.....	9



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2.10	Pompa Air	9
2.11	Catu Daya	10
2.12	Arduino IDE	12
2.13	Bahasa Pemrograman C++	13
2.14	RSSI (Receive Signal Strength Indicator)	20
BAB III PERANCANGAN DAN REALISASI		21
3.1	Rancangan Alat	21
3.1.1	Deskripsi Alat	21
3.1.2	Cara Kerja Alat	22
3.1.3	Spesifikasi Alat	24
3.1.4	Diagram Blok Sistem	26
3.2	Realisasi Alat.....	27
3.2.1	Perancangan Sistem Mikrokontroler.....	27
3.2.2	Perancangan LoRa ke ESP 32.....	28
3.2.3	Perancangan Sensor Ultrasonik.....	30
3.2.4	Perancangan RTC DS3231	31
3.2.5	Perancangan Motor Servo.....	32
3.2.6	Perancangan LED.....	32
3.2.7	Perancangan LCD	33
3.2.8	Perancangan Power Supply.....	34
3.2.9	Realisasi Pemrograman.....	36
BAB IV PEMBAHASAN.....		48
4.1	Pengujian Power Supply	48
4.1.1	Deskripsi Pengujian Power Supply	48
4.1.2	Alat – Alat Pengujian Power Supply	48
4.1.3	Prosedur Pengujian <i>Power Supply</i>	49



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.1.4	Data Hasil Pengujian <i>Power Supply</i>	49
4.2	Pengujian Jarak Pengiriman Sinyal LoRa	49
4.2.1	Deskripsi Pengujian Jarak Pengiriman Sinyal LoRa	49
4.2.2	Alat – Alat Pengujian Jarak Pengiriman Sinyal LoRa	50
4.2.3	Prosedur Pengujian Jarak Pengiriman Sinyal LoRa	50
4.2.4	Data Hasil Pengujian Jarak Pengiriman Sinyal LoRa.....	50
4.3	Pengujian Sensor Ultrasonik HC-SR04	52
4.3.1	Deskripsi Pengujian Sensor Ultrasonik HC-SR04.....	52
4.3.2	Alat – Alat Pengujian Sensor Ultrasonik HC-SR04	52
4.3.3	Prosedur Pengujian Sensor Ultrasonik HC-SR04	52
4.3.4	Data Hasil Pengujian Sensor Ultrasonik HC-SR04	52
4.4	Analisa Sistem.....	53
BAB V PENUTUP	54
5.1	Simpulan.....	54
5.2	Saran.....	54
DAFTAR PUSTAKA	55
DAFTAR RIWAYAT HIDUP	57
LAMPIRAN	58



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Daftar Gambar

Gambar 2. 1 Saluran Irigasi Sawah.....	3
Gambar 2. 2 Hope RFM95W LoRa	5
Gambar 2. 3 HC-SR04	5
Gambar 2. 4 Real Time Clock (RTC)	6
Gambar 2. 5 Motor Servo.....	6
Gambar 2. 6 Susunan pin DOIT ESP32 Devkit.....	7
Gambar 2. 7 LCD.....	8
Gambar 2. 8 LED	9
Gambar 2. 9 Modul Relay 1 Chanel.....	9
Gambar 2. 10 Pompa Air 5V.....	10
Gambar 2. 11 Rangkaian Power Supply	10
Gambar 2. 12 Tampilan Arduino Ide	12
Gambar 3. 1 Ilustrasi Alat Sistem Kontrol Irigasi Sawah.....	22
Gambar 3. 2 Flowchart Sistem Kontrol Irigasi Sawah	23
Gambar 3. 3 Diagram Blok Sistem	26
Gambar 3. 4 Pemasangan Hope RFM95W LoRa Transmitter pada ESP 32 Expansion Board	28
Gambar 3. 5 Pemasangan Hope RFM95W LoRa Receiver pada ESP 32 Expansion Board.....	29
Gambar 3. 6 Perancangan Sensor Ultrasonik.....	30
Gambar 3. 7 Perancangan RTC.....	31
Gambar 3. 8 Perancangan Motor Servo	32
Gambar 3. 9 Perancangan LED.....	33
Gambar 3. 10 Rangkaian Skematik Power Supply	34
Gambar 3. 11 Layout Power Supply	35
Gambar 3. 12 Tampak Bawah Hasil Rangkaian PCB.....	35
Gambar 3. 13 Tampak Atas Hasil Rangkaian PCB	35



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Daftar Tabel

Tabel 2. 1 Tabel Parameter Kualitas RSSI.....	20
Tabel 3. 1 Spesifikasi Hope RFM95W LoRa	24
Tabel 3. 2 Spesifikasi ESP32	24
Tabel 3. 3 Spesifikasi Sensor Ultrasonik	24
Tabel 3. 4 Spesifikasi Real Time Clock.....	25
Tabel 3. 5 Spesifikasi Motor Servo.....	25
Tabel 3. 6 LED	25
Tabel 3. 7 LCD.....	25
Tabel 3. 9 Power Supply	26
Tabel 3. 10 Pin Komponen dengan Pin ESP 32.....	27
Tabel 3. 11 Penggunaan Pin LoRa Transmitter	29
Tabel 3. 12 Penggunaan Pin LoRa Receiver.....	30
Tabel 3. 13 Penggunaan Pin Ultrasonik.....	31
Tabel 3. 14 Penggunaan Pin RTC	31
Tabel 3. 15 Penggunaan Pin Motor Servo	32
Tabel 3. 16 Penggunaan Pin LED	33
Tabel 3. 17 Penggunaan Pin LCD.....	34
Tabel 4. 1 Hasil Pengujian Catu Daya	49
Tabel 4. 2 Data hasil pengujian jarak pengiriman LoRa tanpa penghalang.....	51
Tabel 4. 3 Data hasil pengujian jarak pengiriman LoRa dengan penghalang.....	51
Tabel 4. 4 Data Pengujian Sensor Ultrasonik HC-SR04.....	53



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Daftar Lampiran

Skematik Rangkaian Power Supply	57
Skematik Rangkaian Transmitter	58
Skematik Rangkaian Receiver	59
Desain Casing Transmitter	60
Desain Casing Receiver	61
Sketch Program	62





BAB I PENDAHULUAN

1.1 Latar Belakang

Irigasi atau pengairan merupakan upaya yang dilakukan manusia untuk mengairi lahan pertanian. Irigasi merupakan sebuah alternatif pengairan lahan tadah hujan pada musim kemarau. Dengan adanya irigasi, lahan tidak lagi mengandalkan hujan yang tidak menentu waktunya. Irigasi merupakan salah satu faktor penting dalam produksi padi sawah. Irigasi sebagai sumber bagi ketersediaan air untuk pertumbuhan tanaman padi.

Pengaturan sistem pengairan sawah yang baik dapat menghemat penggunaan air. Umumnya pemberian air yang dipraktekkan petani pada padi sawah irigasi adalah dengan digenangi terus menerus sehingga sangat boros, penggunaan air berkisar antara 11.000-14.000 m³/ha pada musim kemarau (MK) dan 8.000-10.000 m³/ha pada musim hujan (MH). Selain tidak efisien, cara ini juga berpotensi mengurangi efisiensi serapan hara nitrogen, meningkatkan emisi gas metan ke atmosfer, dan menaikkan rembesan yang menyebabkan makin banyak air irigasi yang dibutuhkan.

Teknik hemat air pada padi sawah merupakan upaya untuk menekan kehilangan air dipetakan sawah untuk mempertahankan atau meningkatkan hasil gabah per satuan luas dan volume air. Pengurangan air akibat perkolasi, rembesan, dan aliran permukaan dapat menekan penggunaan air irigasi.

Hal inilah yang mendasari pengusul untuk membuat sebuah prototype sistem kontrol irigasi sawah. Sistem ini dapat mengirimkan hasil pembacaan ketinggian air yang dikontrol oleh sensor ultrasonik. Lalu informasi dikirim ke android menggunakan teknologi Long Range (LoRa). LoRa memiliki jarak jangkauan yang cukup jauh, konsumsi daya yang rendah dan tidak perlu menggunakan ISP atau internet dalam transmisi data LoRa yang meyakinkan kami untuk membuat alat ini agar dapat membantu memudahkan pekerjaan petani sawah. LoRa akan terintegrasi dengan sensor ketinggian air dan mengirimkan data yang didapatkan dan akan diterima di Lora receiver. Maka dari itu, pada tugas akhir ini akan dibuat sebuah

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

“Prototype sistem kontrol irigasi pada sawah menggunakan komunikasi *Long Range (LoRa)* berbasis android”.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang diuraikan diatas, maka permasalahan yang akan dibahas dalam Tugas Akhir ini adalah sebagai berikut :

1. Bagaimana merancang dan membangun sistem kontrol irigasi pada sawah berbasis komunikasi Long Range (LoRa)?
2. Bagaimana menguji performansi dari penggunaan komunikasi Long Range (LoRa) dari sistem kontrol irigasi sawah?
3. Bagaimana mengoneksikan hasil dari sistem pada aplikasi?

1.3 Tujuan

Tujuan yang ingin dicapai dari tugas akhir ini adalah :

1. Merancang dan membangun sistem kontrol irigasi pada sawah berbasis komunikasi Long Range (LoRa).
2. Menguji performansi dari penggunaan komunikasi Long Range (LoRa) dari sistem kontrol irigasi sawah.
3. Mengoneksikan hasil dari sistem pada aplikasi.

1.4 Luaran

Adapun luaran dari tugas akhir ini adalah :

1. *Prototype* sistem kontrol irigasi sawah menggunakan komunikasi *Long Range (LoRa)*.
2. Laporan Tugas Akhir
3. Artikel Ilmiah berjudul “*Prototype* Sistem Kontrol Irigasi pada Sawah Menggunakan Komunikasi *Long Range (LoRa)* Berbasis Android”

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB V PENUTUP

5.1 Simpulan

Berdasarkan hasil pembahasan tentang “*Prototype* Sistem Kontrol Irigasi Sawah Menggunakan Komunikasi *Long Range* (LoRa) berbasis Android” dengan sub judul “Perancangan Sistem Kontrol Irigasi Sawah Menggunakan Komunikasi *Long Range* (LoRa) berbasis Android” dapat disimpulkan bahwa :

Percanganan sistem mikrokontroler irigasi sawah terdapat sisi *transmitter* dan sisi *receiver*, dimana kedua sisi dapat berjalan dengan baik. ESP32 DevKit V1 dan LoRa RFM juga sudah saling terhubung sehingga bisa melakukan pengiriman dan penerimaan data dan data juga sudah masuk ke firebase.

Sistem irigasi sawah menggunakan jaringan komunikasi LoRa dimana pada sisi *transmitter* mampu mengirimkan data kepada sisi receiver pada kondisi LoS tanpa penghalang sejauh 10 meter dengan nilai RSSI yang baik yaitu - 75 dBm. Dan pada jarak maksimum yaitu sejauh 200 meter dengan nilai RSSI yang sangat buruk yaitu -108 dBm. Hasil pengujian jarak pengiriman data dengan adanya penghalang tidak jauh berbeda dengan yang tanpa penghalang, dimana pada jarak 10 meter mendapatkan nilai RSSI yang baik sebesar -80 dBm dan pada jarak maksimum yaitu 200 meter mendapatkan nilai RSSI yang sangat buruk yaitu sebesar -113 dBm. Semua pengujian ini dapat dikatakan baik karena LoRa dapat mengirimkan data dengan baik tanpa kerusakan dan juga jarak yang lumayan jauh meskipun mendapatkan nilai RSSI yang buruk. Pada pengujian dengan adanya penghalang jarak bisa saja berubah tergantung dari banyak sedikitnya penghalang, semakin banyak penghalang maka semakin dekat jarak pancar LoRa sedangkan semakin sedikit penghalang maka semakin jauh juga daya pancar LoRa.

5.2 Saran

Diharapkan dengan adanya sistem control irigasi sawah berbasis komunikasi LoRa dapat digunakan di kehidupan sehari-hari, terutama digunakan oleh petani untuk mempermudah pengairan sawah.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR PUSTAKA

- Asril, Aprinal Adila, dkk. 2018. *Perancangan dan Implementasi WSN (Wireless Sensor Network) pada Alat Ukur Energi Listrik*. Jurusan Teknik Elektro. Politeknik Negeri Padang ; Padang.
- Azka Dani. 2022. *Pengertian dan Cara Kerja Motor Servo*. <https://wikielektronika.com/pengertian-dan-cara-kerja-motorservo/>
- Batong.Ayub Repa. Dkk. .2020. *Analisa Kelayakan LoRa untuk Jaringan Komunikasi Sistem Monitoring Listrik di Politeknik Negeri Samarinda*. Jurusan Teknik Elektro. Politeknik Negeri Samarinda : Samarinda.
- Dwinata, Dimas. (2017). “*Analisis Kerja Rangkaian Rectifier Signal Amplifier Sebagai Pembersih Siaran Televisi*”. Tugas Akhir. Universitas Muhammadiyah Sumatera Utara, Medan.
- Elga Aris Prastyo. 2022. *Memulai ESP32 dengan LoRa menggunakan Arduino IDE*. Arduino Biz.
- Husna, Tisnainil. 2018. *PROTOTYPE SISTEM PENGATUR IRIGASI SAWAH MENGGUNAKAN METODE IRIGASI ALTERNATE WETTING AND DRYING (AWD) BERBASIS TEKNOLOGI INTERNET OF THINGS*. Laporan Tugas Akhir Sistem Komputer, Universitas Andalas.
- Nengsi, Sri Widya. 2019. *Monitoring Kendaraan Menggunakan Long Range Radio Frekuensi Berbasis WEB*. Jurusan Teknik Informati. UIN Alauddin Makassar ; Makassar.
- Nugraha, Fandhi K. 2016. *Tugas Sensor Ultrasonic HC-SR04*. Universitas Hasanuddin : Makasar.
- Puspasari, Fitri dkk. 2019. *Sensor Ultrasonik HCSR04 Berbeasis Arduino Due Untuk Sistem Monitoring Ketinggian*. Jurusan Teknik Elektro dan Informatika Sekolah Vokasi. Universitas Gadjah Mada : DI Yogyakarta.
- Santoso. Hari. (2015). *Panduan Praktis Arduino Untuk Pemula*. Trenggalek: Elang Sakti.



© Hak Cipta milik Politeknik Negeri Jakarta

Surjono, Herman Dwi. (2007). *Elektronika : Teori dan Penerapan*. Jawa Timur : Cerdas Ulet Kreatif

Widianto. Eko Didik, Faizal. Al Arthur, dkk. 2019. *Simple LoRa Protocol : Protokol Komunikasi LoRa Untuk Sistem Pemantauan Multisensor*. *Jurnal Telekomunikasi, Elektronika, Komuptasi, dan Kontrol*: Vol.5, No. 2, November 2019, Hal. 83-92.

Santoso. Hari. (2015). *Panduan Praktis Arduino Untuk Pemula*. Trenggalek: Elang Sakti.

Firestore, Google. 2023. <https://firebase.google.com/?hl=id>



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengummumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR RIWAYAT HIDUP

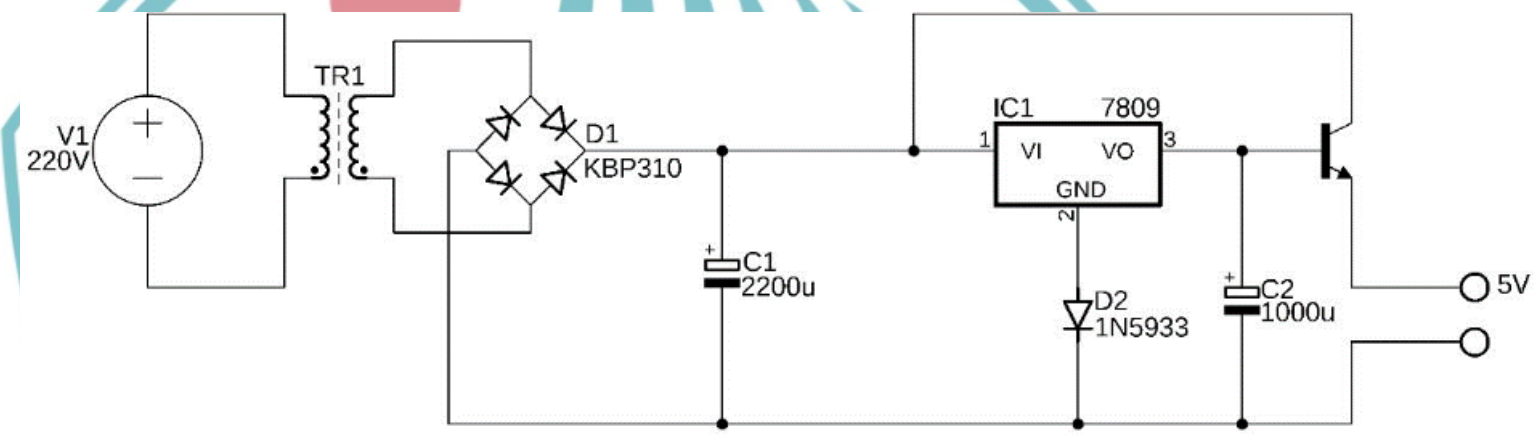


RAUDHATUL JANNAH

Lahir di Payakumbuh, 27 Februari 2002. Lulus dari SD 07 Belakang Balok Bukittinggi tahun 2014, SMP Negeri 1 Bukittinggi tahun 2017, dan SMA Negeri 3 Bukittinggi Tahun 2020. Gelar Diploma Tiga (D3) diperoleh pada tahun 2023 dari Jurusan Teknik Elektro, Program Studi Teknik Telekomunikasi, Politeknik Negeri Jakarta.



LAMPIRAN



01

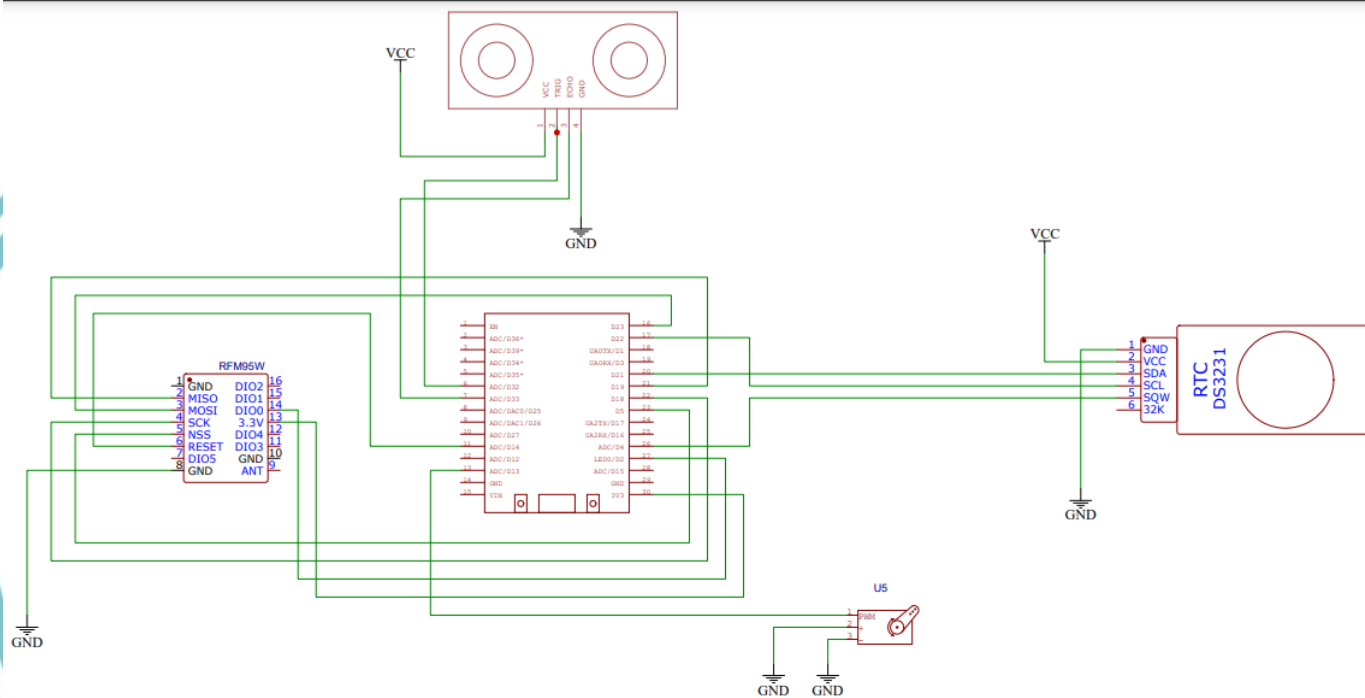
RANGKAIAN POWER SUPPLY



PROGRAM STUDI TELEKOMUNIKASI
 JURUSAN TEKNIK ELEKTRO
 POLITEKNIK NEGERI JAKARTA

Digambar	: Raudhatul Jannah
Diperiksa	: Benny Nixon, S.T., M.T.
Tanggal	:

Hak Cipta :
 1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatuu
 b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



02

SKEMATIK RANGKAIAN SISTEM KONTROL IRIGASI SAWAH MENGGUNAKAN KOMUNIKASI LONG RANGE (LoRa) BERBASIS ANDROID SISI PENGIRIM



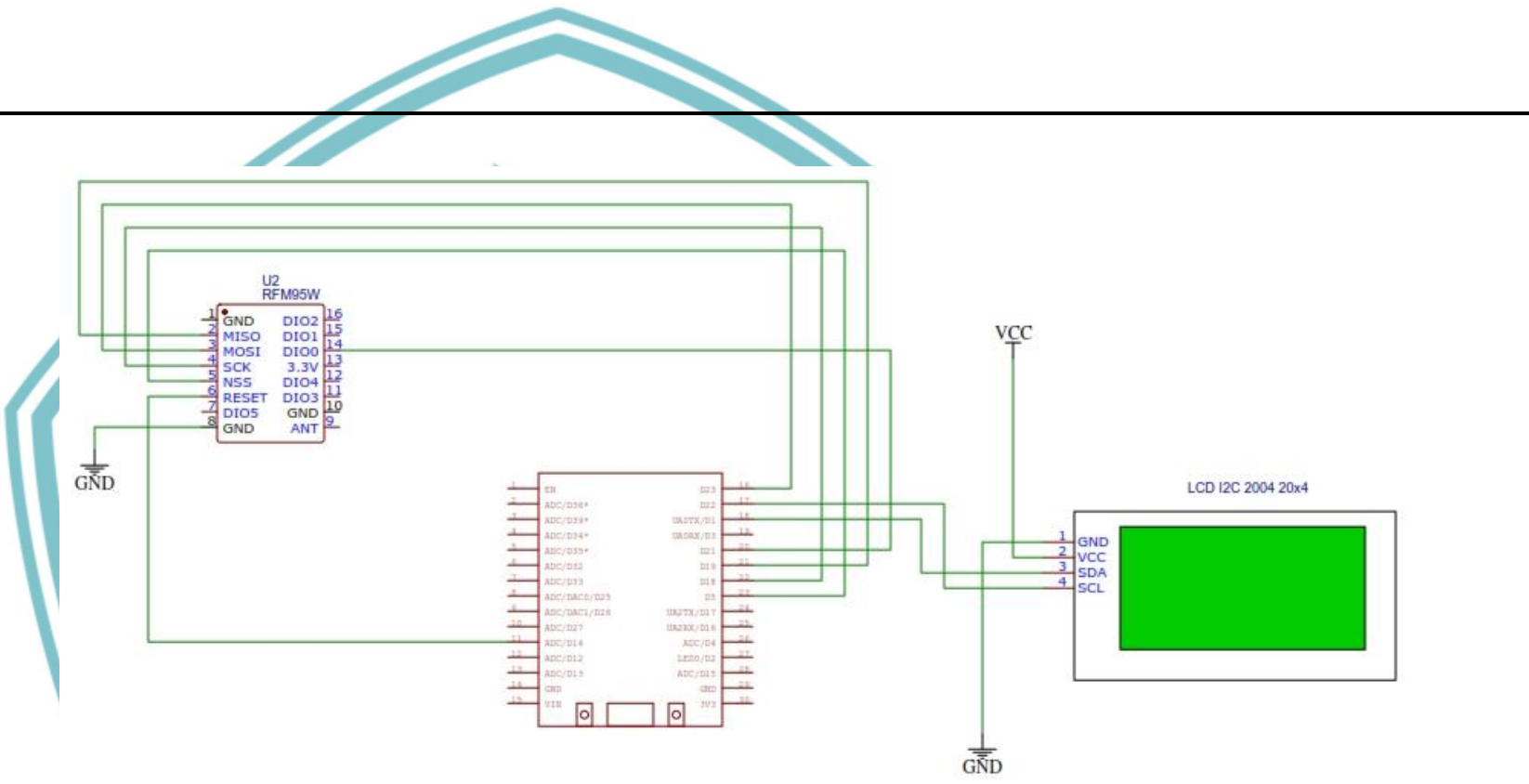
PROGRAM STUDI TELEKOMUNIKASI
 JURUSAN TEKNIK ELEKTRO
 POLITEKNIK NEGERI JAKARTA


Digambar	: Raudhatul Jannah
Diperiksa	: Benny Nixon, S.T., M.T.
Tanggal	:

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatru
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

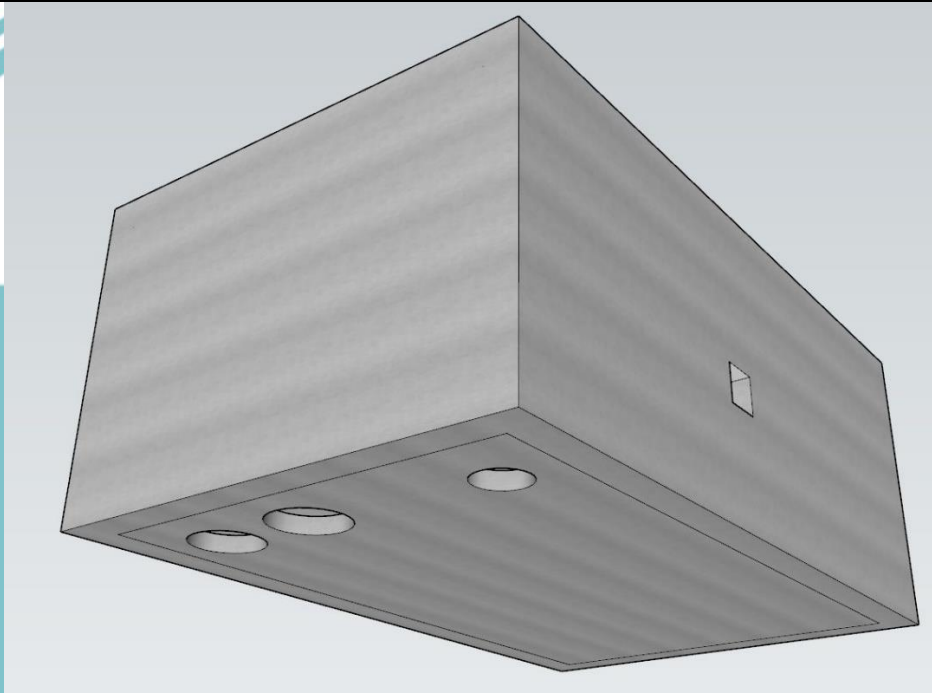
- Hak Cipta :**
1. Diarangi mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatru
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
 2. Diarangi mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta




03	SKEMATIK RANGKAIAN SISTEM KONTROL IRIGASI SAWAH MENGGUNAKAN KOMUNIKASI LONG RANGE (LoRa) BERBASIS ANDROID SISI PENERIMA		
 <p>PROGRAM STUDI TELEKOMUNIKASI JURUSAN TEKNIK ELEKTRO POLITEKNIK NEGERI JAKARTA</p>	Digambar	: Raudhatul Jannah	
	Diperiksa	: Benny Nixon, S.T., M.T.	
	Tanggal	:	

Hak Cipta :

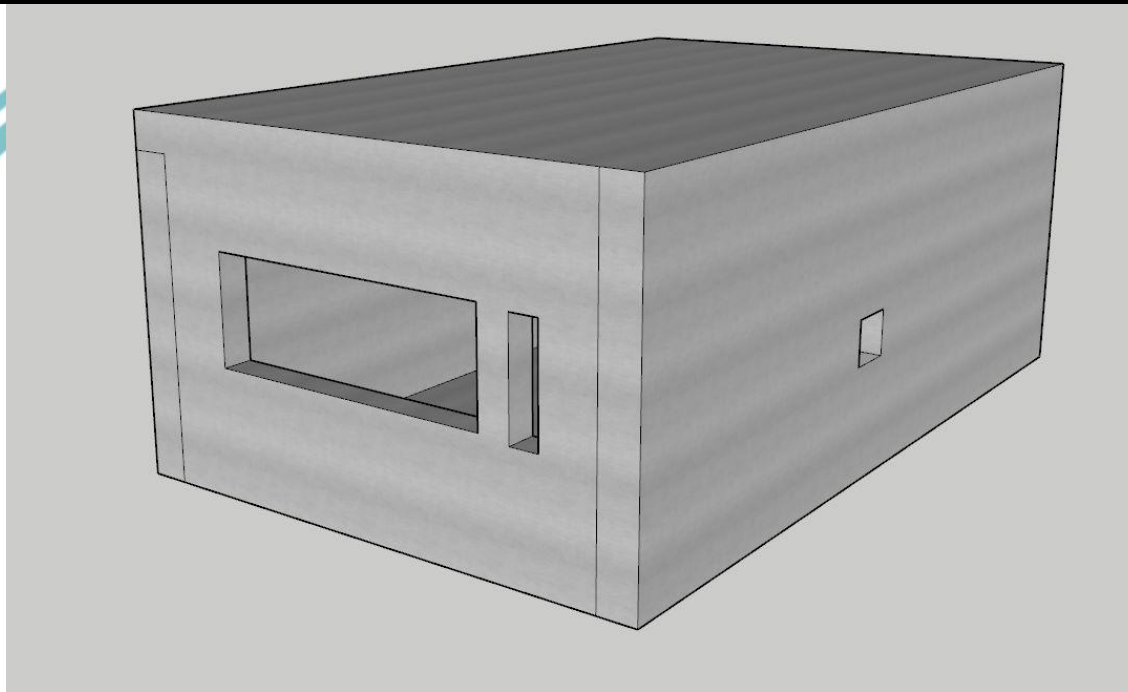
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta




04	DESIGN CASING TRANSMITTER		
	PROGRAM STUDI TELEKOMUNIKASI	Digambar	: Raudhatul Jannah
	JURUSAN TEKNIK ELEKTRO	Diperiksa	: Benny Nixon, S.T., M.T.
	POLITEKNIK NEGERI JAKARTA	Tanggal	:

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



05	NEGERI JAKARTA <i>DESIGN CASING RECEIVER</i>		
	PROGRAM STUDI TELEKOMUNIKASI	<i>Digambar</i>	<i>: Raudhatul Jannah</i>
	JURUSAN TEKNIK ELEKTRO	<i>Diperiksa</i>	<i>: Benny Nixon, S.T., M.T.</i>
	POLITEKNIK NEGERI JAKARTA	<i>Tanggal</i>	<i>:</i>



Sketch Program *Transmitter*

```
// include libraries
#include <Arduino.h>
#include <Wire.h>
#include <SPI.h>
#include <ArduinoJson.h>
#include <LoRa.h>
#include <RTCLib.h>
#include <Adafruit_ADS1X15.h>
#include <ESP32Servo.h>
#define USE_HCSR04 //comment this line if you're using water
level sensor
#define WLEVEL_THRESHOLD 40 //batas nilai tinggi muka air
const int trigPin = 33;
const int echoPin = 32;
//define sound speed in cm/uS
#define SOUND_SPEED 0.034
#define CM_TO_INCH 0.393701
#define DISTANCE_TO_GROUND 100 //jarak sensor ke dasar tanah
dalam mm
#define DISTANCE_TO_SURFACE 50 //jarak sensor ke tinggi muka air
maksimal dalam mm
float MAX_WATER_LEVEL = DISTANCE_TO_GROUND - DISTANCE_TO_SURFACE;
long duration;
float distance_mm;
float distance_cm;
float distance_inch;
Adafruit_ADS1115 ads; /* Use this for the 16-bit version */
int16_t adc0;
float volts0;
#define WLEVEL_OFFSET 0 //jarak sensor ke tanah dalam mm
#define WLEVEL_LRV 0 //nilai adc ketika tinggi muka air = 0 mm
#define WLEVEL_URV 4096 //nilai adc ketika tinggi muka air = 44
mm
RTC_DS3231 rtc;
#define CLOCK_INTERRUPT_PIN 4 //the pin that is connected to SQW
Servo myservo; // create servo object to control a servo
// 16 servo objects can be created on the ESP32
int servoPos = 0; // variable to store the servo position
// Recommended PWM GPIO pins on the ESP32 include 2,4,12-19,21-
23,25-27,32-33
const int servoPin = 13;
#define SERVO_CLOSE 0
#define SERVO_OPEN 180
const int csPin = 5; // LoRa radio chip select
const int resetPin = 14; // LoRa radio reset
const int dio0Pin = 2; // change for your board; must be a
hardware interrupt pin
String outgoing; // outgoing message
byte msgCount = 0; // count of outgoing messages
byte localAddress = 0x01; // address of this device
byte destination = 0x00; // destination to send to
long taskTimer = 0; // last task time
int taskInterval = 500; // interval between task (ms)
//FIREBASE VARIABLES
String datetime_alarm = "alarm";
String datetime_now = "now";
```

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
bool state_alarm = false;
bool state_newtarget = false;
uint8_t target_day = 0;
uint8_t target_hour = 0;
uint8_t target_minute = 0;
unsigned long unixtime_now = 0;
float water_level = 0;
float mapf(float x, float in_min, float in_max, float out_min,
float out_max) {
    return (x - in_min) * (out_max - out_min) / (in_max - in_min) +
out_min;
}
void onAlarm() {
    Serial.println("Alarm occured!");
}
void sendMessage(String outgoing) {
    LoRa.beginPacket(); // start packet
    LoRa.write(destination); // add destination address
    LoRa.write(localAddress); // add sender address
    LoRa.write(msgCount); // add message ID
    LoRa.write(outgoing.length()); // add payload length
    LoRa.print(outgoing); // add payload
    LoRa.endPacket(); // finish packet and send it
    msgCount++; // increment message ID
}
void onReceive(int packetSize) {
    if (packetSize == 0) return; // if there's no packet, return
    Serial.println("-----");
    // read packet header bytes:
    int recipient = LoRa.read(); // recipient address
    byte sender = LoRa.read(); // sender address
    byte incomingMsgId = LoRa.read(); // incoming msg ID
    byte incomingLength = LoRa.read(); // incoming msg length
    String incoming = "";
    while (LoRa.available()) {
        incoming += (char)LoRa.read();
    }
    if (incomingLength != incoming.length()) { // check length for
error
        Serial.println("error: message length does not match length");
        return; // skip rest of function
    }
    // if the recipient isn't this device or broadcast,
    if (recipient != localAddress && recipient != 0xFF) {
        Serial.println("This message is not for me.");
        return; // skip rest of function
    }
    // if message is for this device, or broadcast, print details:
    Serial.println("Received from: 0x" + String(sender, HEX));
    Serial.println("Sent to: 0x" + String(recipient, HEX));
    Serial.println("Message ID: " + String(incomingMsgId));
    Serial.println("Message length: " + String(incomingLength));
    Serial.println("Message: " + incoming);
    Serial.println("RSSI: " + String(LoRa.packetRssi()));
    Serial.println("Snr: " + String(LoRa.packetSnr()));
    Serial.println();
    // Allocate the JSON document
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
// Inside the brackets, 200 is the capacity of the memory pool
in bytes.
// Don't forget to change this value to match your JSON
document.
// Use arduinojson.org/v6/assistant to compute the capacity.
StaticJsonDocument<128> docReceive;
// Deserialize the JSON document
DeserializationError error = deserializeJson(docReceive,
incoming);
// Test if parsing succeeds.
if (error) {
  Serial.print(F("deserializeJson() failed: "));
  Serial.println(error.f_str());
  return;
}
// Fetch values.
// Most of the time, you can rely on the implicit casts.
state_newtarget = docReceive["sNewTarget"].as<bool>();
target_day = docReceive["tDay"].as<uint8_t>();
target_hour = docReceive["tHour"].as<uint8_t>();
target_minute = docReceive["tMinute"].as<uint8_t>();
// Print values.
Serial.print("state_newtarget: ");
Serial.printf("%s\n", state_newtarget ? "true" : "false");
Serial.print("target_day: ");
Serial.println(target_day);
Serial.print("target_hour: ");
Serial.println(target_hour);
Serial.print("target_minute: ");
Serial.println(target_minute);
Serial.println();
delay(1);
if (state_newtarget == true) {
  Serial.println("NEW TARGET!");
  // set alarm 1, 2 flag to false (so alarm 1, 2 didn't happen
so far)
  // if not done, this easily leads to problems, as both
register aren't reset on reboot/recompile
  rtc.clearAlarm(1);
  rtc.clearAlarm(2);
  // stop oscillating signals at SQW Pin
  // otherwise setAlarm1 will fail
  rtc.writeSqwPinMode(DS3231_OFF);
  // turn off alarm 2 (in case it isn't off already)
  // again, this isn't done at reboot, so a previously set alarm
could easily go overlooked
  rtc.disableAlarm(2);
  // print current time
  DateTime dtnow = rtc.now();
  char date[12] = "DD hh:mm:ss";
  dtnow.toString(date);
  Serial.println(date);
  // schedule an alarm 10 seconds in the future
  if (!rtc.setAlarm1(
    dtnow + TimeSpan(target_day, target_hour, target_minute,
0),
    DS3231_A1_Date // this mode triggers the alarm. See
Doxygen for other options
  )) {
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
Serial.println("Error, alarm wasn't set!");
} else {
    Serial.println("Alarm has been set!");
    state_newtarget = false;
String message = ""; // send a message
StaticJsonDocument<192> docSend;
docSend["dtAlarm"] = datetime_alarm;
docSend["dtNow"] = datetime_now;
docSend["sAlarm"] = state_alarm;
docSend["sNewTarget"] = state_newtarget;
docSend["utNow"] = unixtime_now;
docSend["wLevel"] = water_level;
serializeJson(docSend, message);
sendMessage(message);
Serial.println("NODE Sending: " + message);
Serial.println()
void setup() {
Serial.begin(115200); // initialize serial
while (!Serial) {
    delay(1);
}; // if failed, do nothing
Serial.println();
// initializing the rtc
if (!rtc.begin()) {
    Serial.println("Couldn't find RTC!");
    Serial.flush();
    while (1) delay(10);
}
if (rtc.lostPower()) {
    Serial.println("RTC LOST POWER!");
    // this will adjust to the date and time at compilation
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
}
//we don't need the 32K Pin, so disable it
rtc.disable32K();
// Making it so, that the alarm will trigger an interrupt
pinMode(CLOCK_INTERRUPT_PIN, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(CLOCK_INTERRUPT_PIN),
onAlarm, FALLING);
rtc.clearAlarm(2);
rtc.disableAlarm(2);
/*
// set alarm 1, 2 flag to false (so alarm 1, 2 didn't happen so
far)
// if not done, this easily leads to problems, as both register
aren't reset on reboot/recompile
rtc.clearAlarm(1);
rtc.clearAlarm(2);
// stop oscillating signals at SQW Pin
// otherwise setAlarm1 will fail
rtc.writeSqwPinMode(DS3231_OFF);
// turn off alarm 2 (in case it isn't off already)
// again, this isn't done at reboot, so a previously set alarm
could easily go overlooked
rtc.disableAlarm(2);
// print current time
DateTime dtnow = rtc.now();
char date[12] = "DD hh:mm:ss";
dtnow.toString(date);
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
Serial.println(date);
// schedule an alarm 10 seconds in the future
if (!rtc.setAlarm1(
    dtnow + TimeSpan(28, 1, 1, 0),
    DS3231_A1_Date // this mode triggers the alarm. See
Doxygen for other options
)) {
    Serial.println("Error, alarm wasn't set!");
} else {
    Serial.println("Alarm has been set!");
}
*/
#ifdef USE_HCSR04
    pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
    pinMode(echoPin, INPUT); // Sets the echoPin as an Input
#else /* USE_HCSR04 */
    Serial.println("Getting single-ended readings from AIN0..3");
    Serial.println("ADC Range: 1x gain | +/- 4.096V | 1 bit =
0.125mV");
    // The ADC input range (or gain) can be changed via the
following
// functions, but be careful never to exceed VDD +0.3V max, or
to
// exceed the upper and lower limits if you adjust the input
range!
// Setting these values incorrectly may destroy your ADC!
//
ADS1015 ADS1115
// ads.setGain(GAIN_TWOTHIRDS); // 2/3x gain +/- 6.144V 1 bit =
3mV 0.1875mV (default)
ads.setGain(GAIN_ONE); // 1x gain +/- 4.096V 1 bit = 2mV
0.125mV
// ads.setGain(GAIN_TWO); // 2x gain +/- 2.048V 1 bit = 1mV
0.0625mV
// ads.setGain(GAIN_FOUR); // 4x gain +/- 1.024V 1 bit =
0.5mV 0.03125mV
// ads.setGain(GAIN_EIGHT); // 8x gain +/- 0.512V 1 bit =
0.25mV 0.015625mV
// ads.setGain(GAIN_SIXTEEN); // 16x gain +/- 0.256V 1 bit =
0.125mV 0.0078125mV
if (!ads.begin()) {
    Serial.println("Failed to initialize ADS.");
    while (1) {
        delay(1);
    }; // if failed, do nothing
}
#endif /* USE_HCSR04 */
// Allow allocation of all timers
ESP32PWM::allocateTimer(0);
ESP32PWM::allocateTimer(1);
ESP32PWM::allocateTimer(2);
ESP32PWM::allocateTimer(3);
myservo.setPeriodHertz(50); // standard 50 hz servo
myservo.attach(servoPin, 500, 2450); // attaches the servo on
pin 18 to the servo object
// using default min/max of
1000us and 2000us
// different servos may
require different min/max settings
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
180 sweep // for an accurate 0 to
myservo.write(SERVO_CLOSE);
Serial.println("LoRa Duplex");
// override the default CS, reset, and IRQ pins (optional)
LoRa.setPins(csPin, resetPin, dio0Pin); // set CS, reset, IRQ
pin
if (!LoRa.begin(923E6)) { // initialize ratio at 915 MHz
Serial.println("LoRa init failed. Check your connections.");
while (true) {
delay(1);
}; // if failed, do nothing
LoRa.setSyncWord(0x94); // ranges from 0-0xFF, default 0x34,
see API docs
Serial.println("LoRa init succeeded.");
Serial.println();
void loop() {
if (millis() - taskTimer > taskInterval) {
taskTimer = millis(); // timestamp
Serial.println("-----");
// print current time
DateTime dtnow = rtc.now();
char date[12] = "DD hh:mm:ss";
dtnow.toString(date);
Serial.print(date);
datetime_now = date;
unixtime_now = dtnow.unixtime();
// the stored alarm value + mode
DateTime alarm1 = rtc.getAlarm1();
Ds3231Alarm1Mode alarm1mode = rtc.getAlarm1Mode();
char alarm1Date[12] = "DD hh:mm:ss";
alarm1.toString(alarm1Date);
Serial.print(" [Alarm1: ");
Serial.print(alarm1Date);
datetime_alarm = alarm1Date;
Serial.print(", Mode: ");
switch (alarm1mode) {
case DS3231_A1_PerSecond: Serial.print("PerSecond"); break;
case DS3231_A1_Second: Serial.print("Second"); break;
case DS3231_A1_Minute: Serial.print("Minute"); break;
case DS3231_A1_Hour: Serial.print("Hour"); break;
case DS3231_A1_Date: Serial.print("Date"); break;
case DS3231_A1_Day: Serial.print("Day"); break;
// the value at SQW-Pin (because of pullup 1 means no alarm)
Serial.print("] SQW: ");
Serial.print(digitalRead(CLOCK_INTERRUPT_PIN));
// whether a alarm fired
Serial.print(" - Fired: ");
Serial.print(rtc.alarmFired(1));
// Serial.print(" Alarm2: ");
// Serial.println(rtc.alarmFired(2));
// control register values (see
https://datasheets.maximintegrated.com/en/ds/DS3231.pdf page 13)
// Serial.print(" Control: 0b");
// Serial.println(read_i2c_register(DS3231_ADDRESS,
DS3231_CONTROL), BIN);
Serial.println();
#ifdef USE_HCSR04
// Clears the trigPin
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
// Sets the trigPin on HIGH state for 10 micro seconds
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
// Reads the echoPin, returns the sound wave travel time in
microseconds
duration = pulseIn(echoPin, HIGH);
// Calculate the distance
distance_cm = duration * SOUND_SPEED / 2;
distance_mm = distance_cm * 10;
// Convert to inches
distance_inch = distance_cm * CM_TO_INCH;
// Prints the distance in the Serial Monitor
Serial.print("Distance: ");
Serial.print(distance_mm);
Serial.print(" mm | ");
Serial.print(distance_cm);
Serial.print(" cm | ");
Serial.print(distance_inch);
Serial.println(" inch");
water_level = mapf(distance_mm, DISTANCE_TO_GROUND,
DISTANCE_TO_SURFACE, 0, MAX_WATER_LEVEL);
Serial.print("water_level: ");
Serial.println(water_level);
#else
adc0 = ads.readADC_SingleEnded(0);
if (adc0 < 0) adc0 = 0;
volts0 = ads.computeVolts(adc0);
water_level = WLEVEL_OFFSET + mapf(adc0, WLEVEL_LRV,
WLEVEL_URV, 0, 44);
Serial.print("AIN0: ");
Serial.print(adc0);
Serial.print(" - ");
Serial.print(volts0);
Serial.print(" V - ");
Serial.print("water_level: ");
Serial.print(water_level);
Serial.println(" mm");
#endif
if (water_level < 0) water_level = 0;
if (rtc.alarmFired(1)) {
  if (water_level < WLEVEL_THRESHOLD) {
    myservo.write(SERVO_OPEN);
    state_alarm = true;
  } else {
    myservo.write(SERVO_CLOSE);
    // resetting SQW and alarm 1 flag
    // using setAlarm1, the next alarm could now be
configured
    rtc.clearAlarm(1);
    Serial.println();
    Serial.println("Alarm cleared!");
    Serial.println();
    state_alarm = false;
  }
}
// parse for a packet, and call onReceive with the result:
onReceive(LoRa.parsePacket());
}
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
Sketch Sisi Penerima
#include <Temboo.h>
#include <TembooCoAPEdgeDevice.h>
#include <TembooMonitoring.h>
#include <TembooMQTTEdgeDevice.h>
#include <TembooSSL.h>
#include <TembooYunShield.h>
/*
  LoRa Duplex communication
  Sends a message every half second, and polls continually
  for new incoming messages. Implements a one-byte addressing
  scheme,
  with 0xFF as the broadcast address.
  Uses readString() from Stream class to read payload. The Stream
  class'
  timeout may affect other functuons, like the radio's callback.
  For an
  created 28 April 2017
  by Tom Igoe
*/
// include libraries
#include <Arduino.h>
#include <SPI.h>
#include <LoRa.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <ArduinoJson.h>
/**
 * Created by K. Suwatchai (Mobizt)
 * Email: k_suwatchai@hotmail.com
 * Github: https://github.com/mobizt/Firebase-ESP-Client
 * Copyright (c) 2023 mobizt
 */
#if defined(ESP32) || defined(ARDUINO_RASPBERRY_PI_PICO_W)
#include <WiFi.h>
#elif defined(ESP8266)
#include <ESP8266WiFi.h>
#endif
#include <Firebase_ESP_Client.h>
// Provide the token generation process info.
#include <addons/TokenHelper.h>
// Provide the RTDB payload printing info and other helper
functions.
#include <addons/RTDBHelper.h>
/* 1. Define the WiFi credentials */
#define WIFI_SSID "Ashwen"
#define WIFI_PASSWORD "banyakikan"
// For the following credentials, see
examples/Authentications/SignInAsUser/EmailPassword/EmailPassword.
ino
/* 2. Define the API Key */
#define API_KEY "AIzaSyCOpwqMrBHXlmlsmebCc6I1LF5MQtr_bHM"
/* 3. Define the RTDB URL */
#define DATABASE_URL "https://ta-anna-siti-2023-default-rtdb.asia-
southeast1.firebaseio.com/" //<databaseName>.firebaseio.com
or //<databaseName>.<region>.firebasedatabase.app
/* 4. Define the user Email and password that already registered
or added in
* your project */
```


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
#define USER_EMAIL "taannasiti2023@gmail.com"
#define USER_PASSWORD "ta-anna-siti-2023"
// Define Firebase Data object
FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;
#ifdef ARDUINO_RASPBERRY_PI_PICO_W
WiFiMulti multi;
#endif
LiquidCrystal_I2C lcd(0x27, 20, 4); // set the LCD address to
0x27 for a 16 chars and 2 line display
char lcdBuf[32];
const int csPin = 5; // LoRa radio chip select
const int resetPin = 14; // LoRa radio reset
const int dio0Pin = 2; // change for your board; must be a
hardware interrupt pin
String outgoing; // outgoing message
byte msgCount = 0; // count of outgoing messages
byte localAddress = 0x00; // address of this device
byte destination = 0x01; // destination to send to
long taskTimer = 0; // last task time
int taskInterval = 6000; // interval between task (ms)
//FIREBASE VARIABLES
String datetime_alarm = "null";
String datetime_now = "null";
bool state_alarm = false;
bool state_newtarget = false;
uint8_t target_day = 0;
uint8_t target_hour = 0;
uint8_t target_minute = 0;
unsigned long unixtime_now = 0;
float water_level = -1.0;

void sendMessage(String outgoing) {
  LoRa.beginPacket(); // start packet
  LoRa.write(destination); // add destination address
  LoRa.write(localAddress); // add sender address
  LoRa.write(msgCount); // add message ID
  LoRa.write(outgoing.length()); // add payload length
  LoRa.print(outgoing); // add payload
  LoRa.endPacket(); // finish packet and send it
  msgCount++; // increment message ID
}

void onReceive(int packetSize) {
  if (packetSize == 0) return; // if there's no packet, return
  // read packet header bytes:
  int recipient = LoRa.read(); // recipient address
  byte sender = LoRa.read(); // sender address
  byte incomingMsgId = LoRa.read(); // incoming msg ID
  byte incomingLength = LoRa.read(); // incoming msg length
  String incoming = "";
  while (LoRa.available()) {
    incoming += (char)LoRa.read();
  }
  if (incomingLength != incoming.length()) { // check length for
error
    Serial.println("error: message length does not match length");
    return; // skip rest of function
  }
}
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
// if the recipient isn't this device or broadcast,
if (recipient != localAddress && recipient != 0xFF) {
    Serial.println("This message is not for me.");
    return; // skip rest of function
}
Serial.flush();
// if message is for this device, or broadcast, print details:
Serial.println("Received from: 0x" + String(sender, HEX));
Serial.println("Sent to: 0x" + String(recipient, HEX));
Serial.println("Message ID: " + String(incomingMsgId));
Serial.println("Message length: " + String(incomingLength));
Serial.println("Message: " + incoming);
Serial.println("RSSI: " + String(LoRa.packetRssi()));
Serial.println("Snr: " + String(LoRa.packetSnr()));
Serial.println();
// Allocate the JSON document
// Inside the brackets, 192 is the capacity of the memory pool
in bytes.
// Don't forget to change this value to match your JSON
document.
// Use arduinojson.org/v6/assistant to compute the capacity.
StaticJsonDocument<192> docReceive;
// Deserialize the JSON document
DeserializationError error = deserializeJson(docReceive,
incoming);
// Test if parsing succeeds.
if (error) {
    Serial.print(F("deserializeJson() failed: "));
    Serial.println(error.f_str());
    return;
}
// Fetch values.
// Most of the time, you can rely on the implicit casts.
datetime_alarm = docReceive["dtAlarm"].as<String>();
datetime_now = docReceive["dtNow"].as<String>();
state_alarm = docReceive["sAlarm"].as<bool>();
state_newtarget = docReceive["sNewTarget"].as<bool>();
unixtime_now = docReceive["utNow"].as<unsigned long>();
water_level = docReceive["wLevel"].as<float>();

// Print values.
Serial.print("datetime_alarm: ");
Serial.println(datetime_alarm);
Serial.print("datetime_now: ");
Serial.println(datetime_now);
Serial.print("state_alarm: ");
Serial.printf("%s\n", state_alarm ? "true" : "false");
Serial.print("state_newtarget: ");
Serial.printf("%s\n", state_newtarget ? "true" : "false");
Serial.print("unixtime_now: ");
Serial.println(unixtime_now);
Serial.print("water_level: ");
Serial.println(water_level);
Serial.println();
if (state_alarm == false) {
    rygLightWrite(LOW, HIGH, LOW);
} else if (state_alarm == true) {
    rygLightWrite(HIGH, LOW, LOW);
}
}
```

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

lcd.clear();
lcd.setCursor(0, 0);
sprintf(lcdBuf, "ALARM STATE: %s", state_alarm ? "ON" : "OFF");
lcd.print(lcdBuf);
lcd.setCursor(0, 1);
lcd.print("AL1: " + datetime_alarm);
lcd.setCursor(0, 2);
lcd.print("NOW: " + datetime_now);
lcd.setCursor(0, 3);
sprintf(lcdBuf, "WATER LV: %.0f mm", water_level);
lcd.print(lcdBuf);
if (Firebase.ready()) {
  Serial.printf("Set datetime_alarm... %s\n",
    Firebase.RTDB.setString(&fbdo, F("/transaction/datetime_alarm"),
    datetime_alarm) ? "ok" : fbdo.errorReason().c_str());
  Serial.printf("Set datetime_now... %s\n",
    Firebase.RTDB.setString(&fbdo, F("/transaction/datetime_now"),
    datetime_now) ? "ok" : fbdo.errorReason().c_str());
  Serial.printf("Set state_alarm... %s\n",
    Firebase.RTDB.setBool(&fbdo, F("/transaction/state_alarm"),
    state_alarm) ? "ok" : fbdo.errorReason().c_str());
  Serial.printf("Set state_newtarget... %s\n",
    Firebase.RTDB.setBool(&fbdo, F("/transaction/state_newtarget"),
    state_newtarget) ? "ok" : fbdo.errorReason().c_str());
  Serial.printf("Set unixtime_now... %s\n",
    Firebase.RTDB.setInt(&fbdo, F("/transaction/unixtime_now"),
    unixtime_now) ? "ok" : fbdo.errorReason().c_str());
  Serial.printf("Set water_level... %s\n",
    Firebase.RTDB.setFloat(&fbdo, F("/transaction/water_level"),
    water_level) ? "ok" : fbdo.errorReason().c_str());
  Serial.println();
}
}
//LED Red (VAL_R), Yellow (VAL_Y), Green (VAL_G) - digital write
LOW or HIGH
void rygLightWrite(uint8_t VAL_R, uint8_t VAL_Y, uint8_t VAL_G) {
  pinMode(25, OUTPUT);
  pinMode(26, OUTPUT);
  pinMode(27, OUTPUT);
  digitalWrite(25, VAL_R);
  digitalWrite(26, VAL_Y);
  digitalWrite(27, VAL_G);
}
void setup() {
  Serial.begin(115200); // initialize serial
  while (!Serial) {
    delay(1);
  }; // if failed, do nothing
  lcd.begin();
  // Print a message to the LCD.
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("IOT-LORA CONTROL SYS");
  lcd.setCursor(3, 1);
  lcd.print("IRIGASI SAWAH");
  lcd.setCursor(4, 2);
  lcd.print("ANNA -- SITI");
  lcd.setCursor(2, 3);
  lcd.print("PNJ-TE-TT TA2023");
}

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
// override the default CS, reset, and IRQ pins (optional)
LoRa.setPins(csPin, resetPin, dio0Pin); // set CS, reset, IRQ
pin
  if (!LoRa.begin(923E6)) { // initialize ratio at
    915 MHz
      Serial.println("LoRa init failed. Check your connections.");
      while (true) {
        delay(1);
      }; // if failed, do nothing
    }
  LoRa.setSyncWord(0x94); // ranges from 0-0xFF, default 0x34,
  see API docs
  Serial.println("LoRa init succeeded.");
#ifdef ARDUINO_RASPBERRY_PI_PICO_W
  multi.addAP(WIFI_SSID, WIFI_PASSWORD);
  multi.run();
#else
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
#endif
  Serial.print("Connecting to Wi-Fi");
#ifdef ARDUINO_RASPBERRY_PI_PICO_W
  unsigned long ms = millis();
#endif
  while (WiFi.status() != WL_CONNECTED) {
    rygLightWrite(HIGH, HIGH, HIGH);
    Serial.print(".");
    delay(125);
  }
#ifdef ARDUINO_RASPBERRY_PI_PICO_W
  if (millis() - ms > 10000)
    break;
#endif
  rygLightWrite(LOW, LOW, LOW);
  delay(125);
}
rygLightWrite(HIGH, HIGH, HIGH);
Serial.println();
Serial.print("Connected with IP: ");
Serial.println(WiFi.localIP());
Serial.println();
Serial.printf("Firebase Client v%s\n\n",
FIREBASE_CLIENT_VERSION);
/* Assign the api key (required) */
config.api_key = API_KEY;
/* Assign the user sign in credentials */
auth.user.email = USER_EMAIL;
auth.user.password = USER_PASSWORD;

/* Assign the RTDB URL (required) */
config.database_url = DATABASE_URL;
// The WiFi credentials are required for Pico W
// due to it does not have reconnect feature.
#ifdef ARDUINO_RASPBERRY_PI_PICO_W
  config.wifi.clearAP();
  config.wifi.addAP(WIFI_SSID, WIFI_PASSWORD);
#endif
/* Assign the callback function for the long running token
generation task */
config.token_status_callback = tokenStatusCallback; // see
addons/TokenHelper.h
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
// Or use legacy authenticate method
// config.database_url = DATABASE_URL;
// config.signer.tokens.legacy_token = "<database secret>";
// To connect without auth in Test Mode, see
Authentications/TestMode/TestMode.ino
  Firebase.begin(&config, &auth);
  Firebase.reconnectWiFi(true);
#if defined(ESP8266)
  fbdo.setBSSLBufferSize(512, 2048);
#endif
  rygLightWrite(LOW, LOW, LOW);
  Serial.println();
}

void loop() {
  if (millis() - taskTimer > taskInterval) {
    taskTimer = millis(); // timestamp
    if (Firebase.ready()) {
      Serial.print("Get state_newtarget... ");
      if (Firebase.RTDB.getBool(&fbdo,
"/transaction/state_newtarget")) {
        if (fbdo.dataTypeEnum() == fb_esp_rtdb_data_type_boolean)
        {
          state_newtarget = fbdo.to<bool>();
          Serial.printf("%s\n", state_newtarget ? "true" :
"false");
        }
        else {
          Serial.println(fbdo.errorReason());
        }
      }
      Serial.print("Get target_day... ");
      if (Firebase.RTDB.getInt(&fbdo, "/transaction/target_day"))
      {
        if (fbdo.dataTypeEnum() == fb_esp_rtdb_data_type_integer)
        {
          target_day = fbdo.to<int>();
          Serial.println(target_day);
        }
        else {
          Serial.println(fbdo.errorReason());
        }
      }
      Serial.print("Get target_hour... ");
      if (Firebase.RTDB.getInt(&fbdo, "/transaction/target_hour"))
      {
        if (fbdo.dataTypeEnum() == fb_esp_rtdb_data_type_integer)
        {
          target_hour = fbdo.to<int>();
          Serial.println(target_hour);
        }
        else {
          Serial.println(fbdo.errorReason());
        }
      }
      Serial.print("Get target_minute... ");
      if (Firebase.RTDB.getInt(&fbdo,
"/transaction/target_minute")) {
        if (fbdo.dataTypeEnum() == fb_esp_rtdb_data_type_integer)
        {
          target_minute = fbdo.to<int>();
          Serial.println(target_minute);
        }
      }
    }
  }
}
```



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
    } else {
        Serial.println(fbdo.errorReason());
    }
}
if (state_newtarget == true) {
    rygLightWrite(LOW, LOW, HIGH);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("TARGET BARU!");
    lcd.setCursor(0, 1);
    sprintf(lcdBuf, "HARI   : %u", target_day);
    lcd.print(lcdBuf);
    lcd.setCursor(0, 2);
    sprintf(lcdBuf, "JAM    : %u", target_hour);
    lcd.print(lcdBuf);
    lcd.setCursor(0, 3);
    sprintf(lcdBuf, "MENIT  : %u", target_minute);
    lcd.print(lcdBuf);
    delay(2000);
    String message = ""; // send a message
    StaticJsonDocument<128> docSend;
    docSend["sNewTarget"] = state_newtarget;
    docSend["tDay"] = target_day;
    docSend["tHour"] = target_hour;
    docSend["tMinute"] = target_minute;
    serializeJson(docSend, message);
    sendMessage(message);
    Serial.println("GW Sending: " + message);
    Serial.println();

    // parse for a packet, and call onReceive with the result:
    onReceive(LoRa.parsePacket())
```

**POLITEKNIK
NEGERI
JAKARTA**