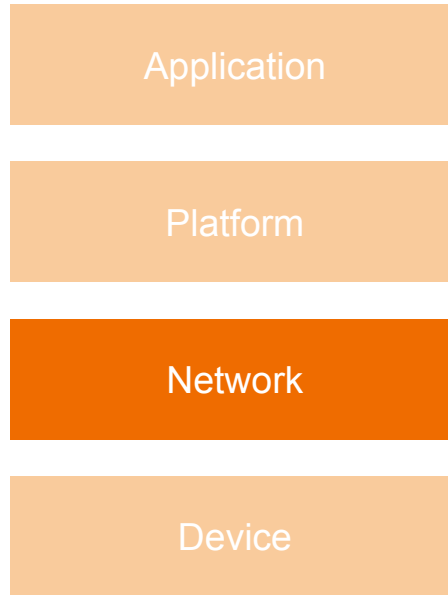

LoRa & LoRaWAN Connectivity

— Fariz Alemuda —

IoT Stack



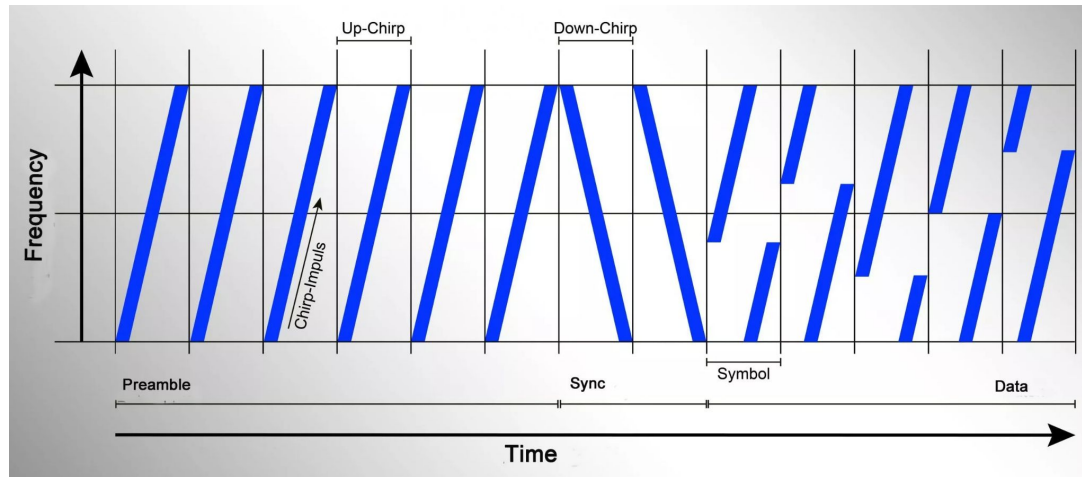
IoT Network



Apa itu LoRa & LoRaWAN?

LoRa vs LoRaWAN

LoRa adalah sebuah teknologi modulasi berbasis Chirp Spread Spectrum (CSS) technology. CSS mengirimkan informasi via gelombang radio yang “menanjak tajam” atau “menunduk tajam”, mirip dengan modulasi komunikasi yang digunakan lumba-lumba dan kelelawar

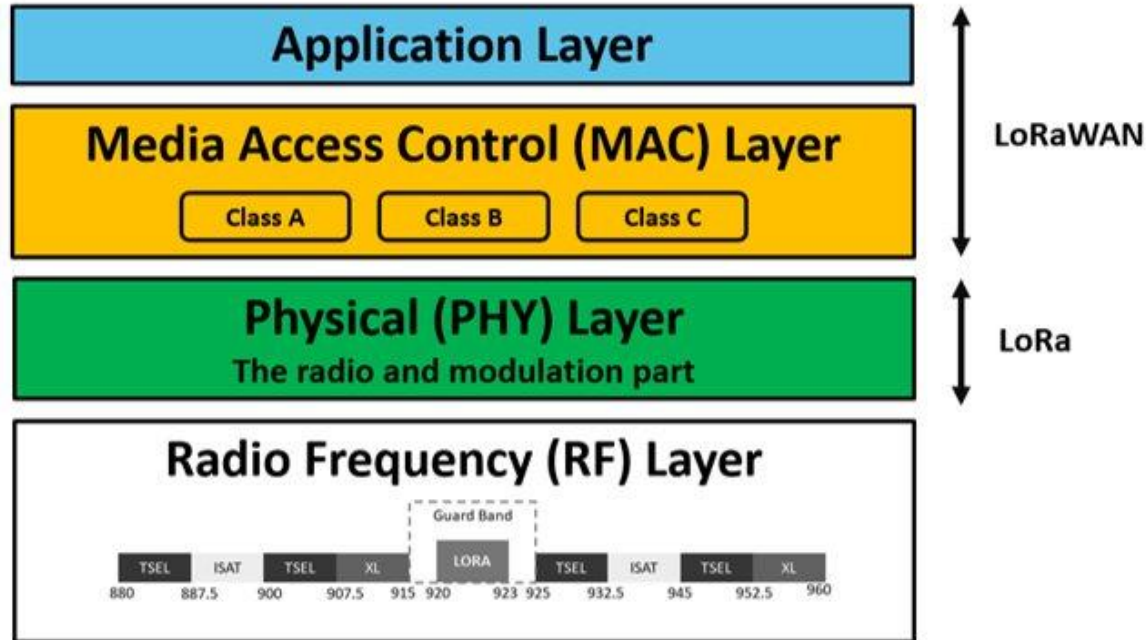


LoRa vs LoRaWAN

LoRaWAN adalah Media Access Control (MAC) layer protocol yang dibuat berbasis protokol modulasi LoRa. LoRaWAN merupakan software layer yang membantu mengatur device management, security, schedule pengiriman, format data, dll.

The LoRaWAN protocol dibuat dan dikembangkan oleh LoRa Alliance.

LoRa vs LoRaWAN



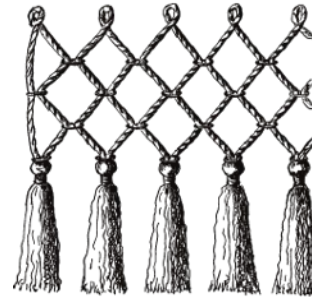
LoRa vs LoRaWAN

LoRa



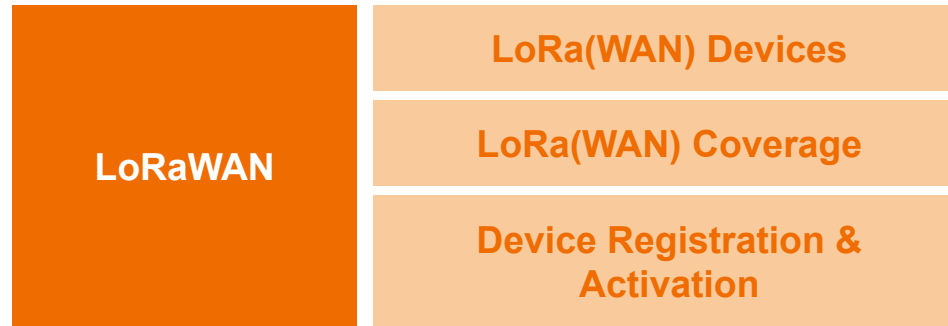
Tali

LoRaWAN



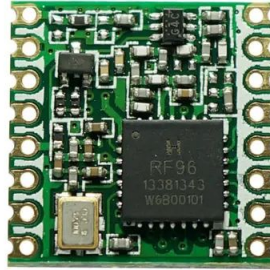
Jaring

How to Connect?



LoRaWAN Device

Chipsets Option

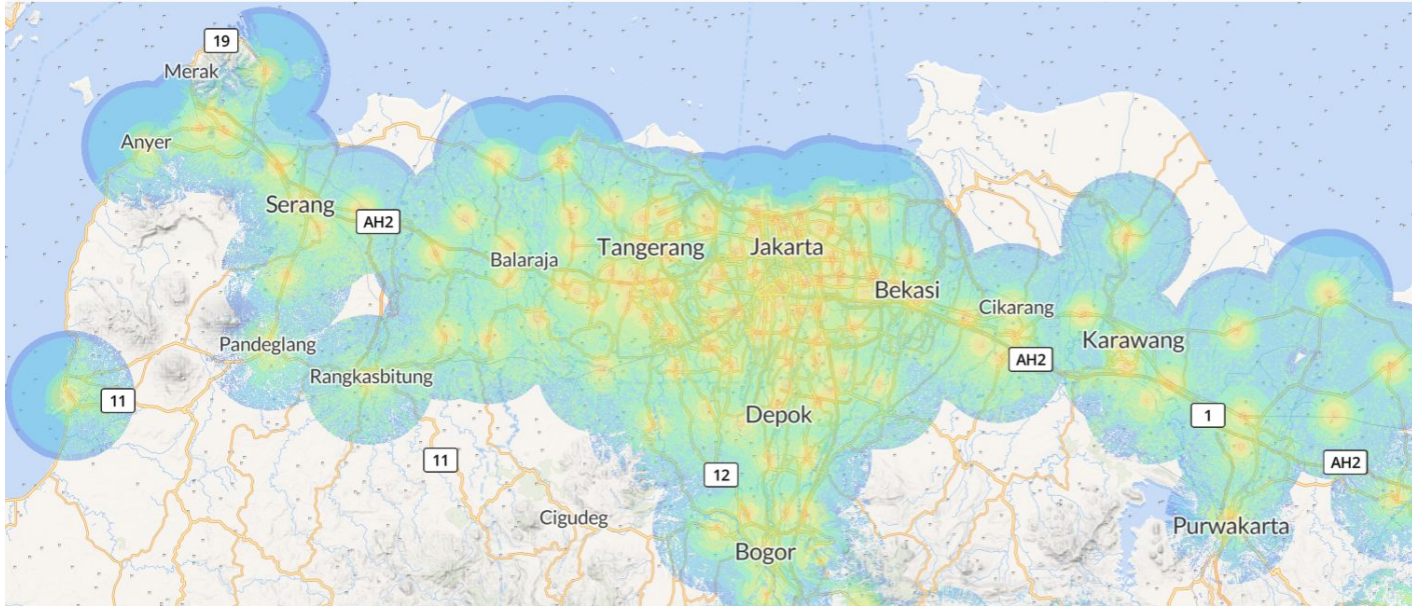


RFM95W



- LoRaTM Modem.
- 168 dB maximum link budget.
- +20 dBm - 100 mW constant RF output vs. V supply.
- +14 dBm high efficiency PA.
- Programmable bit rate up to 300 kbps.
- High sensitivity: down to -148 dBm.
- Bullet-proof front end: IIP3 = -12.5 dBm.
- Excellent blocking immunity.
- Low RX current of 10.3 mA, 200 nA register retention.
- Fully integrated synthesizer with a resolution of 61 Hz.
- FSK, GFSK, MSK, GMSK, LoRaTM and OOK modulation.
- Built-in bit synchronizer for clock recovery.
- Preamble detection.
- 127 dB Dynamic Range RSSI.
- Automatic RF Sense and CAD with ultra-fast AFC.
- Packet engine up to 256 bytes with CRC.
- Built-in temperature sensor and low battery indicator.
- Module Size : 16*16mm

Coverage ANTARES Connectivity LoRaWAN

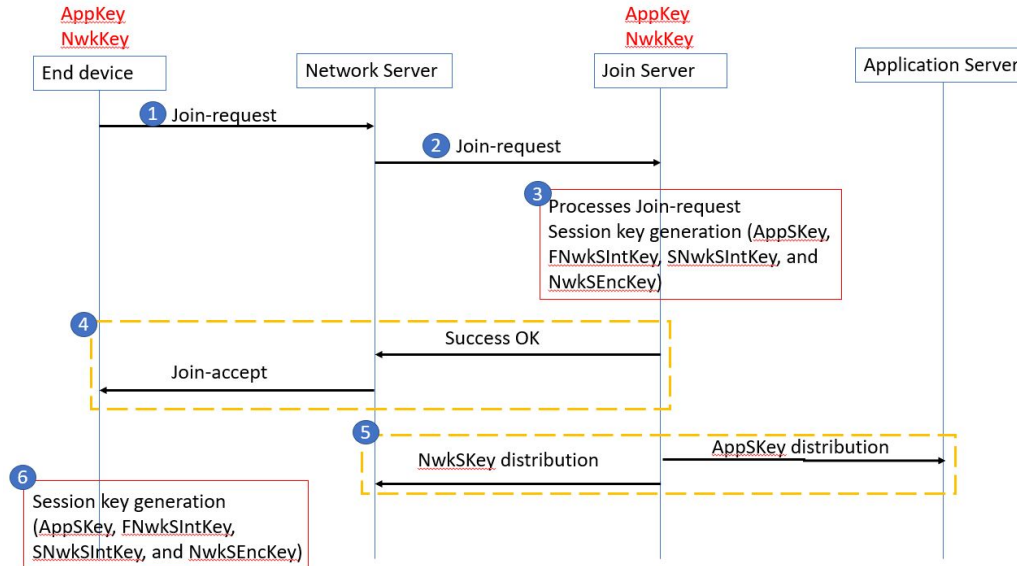


<https://www.telkomiot.com/coverage>

Device Activation LoRaWAN

- **Over-The-Air-Activation (OTAA)** - metode aktivasi paling aman dan rekomen bagi end-device. Perangkat akan melakukan join procedure sebelum bisa berkomunikasi dengan Network Server sehingga device address dan security key bersifat dinamis.
- **Activation By Personalization (ABP)** - metode aktivasi yang memaksa device menyimpan device address dan parameter security lainnya sehingga membuat metode ini lebih tidak secure dibandingkan OTAA dan device harus dikonfigurasi secara non-remote.

Device Activation LoRaWAN OTAA



Device Activation LoRaWAN ABP



Let's Do Some Hands-On

Sesi Hands-On LoRaWAN

1

Kirim Data Dummy ke ANTARES

2

Kirim Data Sensor Environment ke ANTARES


3


Kirim Data LDR di ANTARES


Mengirimkan Data Dummy via LoRaWAN


Registrasi & Aktivasi Device LoRa

Home / Applications / workshop-widyaedu / SmartFarm





[Package](#) [Data](#) [Set LoRa](#) [Subscribe](#) 



Hierarchical URI
<https://platform.antares.id:8443/~/-/antares-cse/antares-id/workshop-widyaedu/SmartFarm> 

Non-hierarchical URI
<https://platform.antares.id:8443/~/-/antares-cse/cnt-P4g9A4UbQ3-reXqP> 

Platform Package
Antares Unlimited - Unlimited
Package ID: 6255c7d00d4bf2220408a24a
Active

Connectivity Package
-
Package ID: -

 [How to Send Data](#) 

 [How to Subscribe to Device Data](#) 

Registrasi & Aktivasi Device LoRa

Set LoRa Device

LoRa Device Class ⓘ

☐ Class A ☐ Class B ☒ Class C

LoRa Activation Mode

☐ OTAA ☒ ABP

ABP Parameters

☒ Inherit ☐ Custom

Application EUI

0000000000000001

Device EUI

Random 16 hexadecimal characters

Device Address

Random 8 hexadecimal characters

Network Session Key

00000000000000000000000000000000

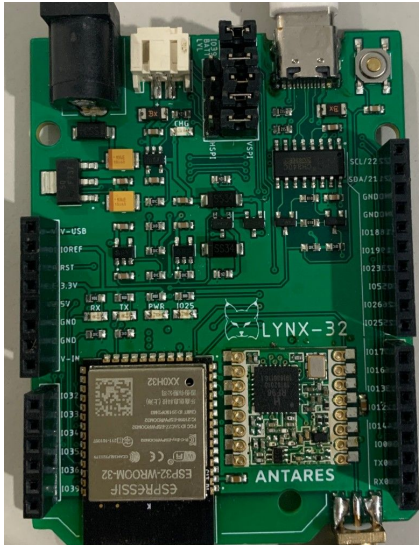
Application Session Key

00000000000000000000000000000000

Are you sure you want to set this device as LoRa device?

Cancel **Set LoRa**

Mengirimkan Data Dummy



LoRaWAN®



ANTARES

Mengirimkan Data Dummy

1

```
#include <lorawan.h>

//ABP Credentials
const char *devAddr = "49cd811c";
const char *nwksKey = "0594ffbb72f6e0a70000000000000000";
const char *appSKey = "0000000000000000190d124ae0e44a69";

const unsigned long interval = 500; // interval to send message
unsigned long previousMillis = 0; // will store last time message sent
unsigned int counter = 0; // message counter

char myStr[50];
byte outStr[255];
byte recvStatus = 0;
int port, channel, freq;
bool newmessage = false;
bool isSent; //Mengirim data hanya 1x

const sRFM_pins RFM_pins = {
    .CS = 5,
    .RST = 0,
    .DIO0 = 27,
    .DIO1 = 2,
};

void setup() {
    // Setup loraid access
    Serial.begin(9600);
    delay(2000);
    if (!lora.init()) {
        Serial.println("RFM95 not detected");
        delay(5000);
        return;
    }
}
```

```
// Set LoRaWAN Class change CLASS_A or CLASS_C
lora.setDeviceClass(CLASS_A);

// Set Data Rate
lora.setDataRate(SF12BW125);

// Set FramePort Tx
lora.setFramePortTx(5);

// set channel to random
lora.setChannel(MULTI);

// Set TxPower to 15 dBi (max)
lora.setTxPower(15);

// Put ABP Key and DevAddress here
lora.setNwksKey(nwksKey);
lora.setAppSKey(appSKey);
lora.setDevAddr(devAddr);
}

void loop() {
    // Check interval overflow
    /*if (millis() - previousMillis > interval) {
        previousMillis = millis();*/

    if (isSent==false && millis() - previousMillis > interval) {
        previousMillis = millis();
        isSent=true;
        sprintf(myStr, "Lora Counter-%d", counter++); //kalo mau ngirim cmn 1x

        Serial.print("Sending: ");
        Serial.println(myStr);
        lora.sendUplink(myStr, strlen(myStr), 0);
    }
}
```

ITARES

2

Mengirimkan Data Dummy

3

```
port = lora.getFramePortTx();
channel = lora.getChannel();
freq = lora.getChannelFreq(channel);
Serial.print(F("fport: ")); Serial.print(port);Serial.print(" ");
Serial.print(F("Ch: ")); Serial.print(channel);Serial.print(" ");
Serial.print(F("Freq: ")); Serial.print(freq);Serial.println(" ");

}

// Check Lora RX
lora.update();

recvStatus = lora.readDataByte(outStr);
if (recvStatus) {
    newmessage = true;
    int counter = 0;
    port = lora.getFramePortRx();
    channel = lora.getChannelRx();
    freq = lora.getChannelRxFreq(channel);

    for (int i = 0; i < recvStatus; i++)
    {
        if (((outStr[i] >= 32) && (outStr[i] <= 126)) || (outStr[i] == 10) ||
(outStr[i] == 13))
            counter++;
    }
    if (port != 0)
    {
        if (counter == recvStatus)
        {
            Serial.print(F("Received String : "));
            for (int i = 0; i < recvStatus; i++)
            {
                Serial.print(char(outStr[i]));
            }
        }
    }
}
```

```
    }
    Serial.println();
    Serial.print(F("fport: ")); Serial.print(port);Serial.print(" ");
    Serial.print(F("Ch: ")); Serial.print(channel);Serial.print(" ");
    Serial.print(F("Freq: ")); Serial.println(freq);Serial.println("
");
}
else
{
    Serial.print(F("Received Mac Cmd : "));
    for (int i = 0; i < recvStatus; i++)
    {
        Serial.print(outStr[i], HEX); Serial.print(" ");
    }
    Serial.println();
    Serial.print(F("fport: ")); Serial.print(port);Serial.print(" ");
    Serial.print(F("Ch: ")); Serial.print(channel);Serial.print(" ");
    Serial.print(F("Freq: ")); Serial.println(freq);Serial.println("
");
}
}
```

4

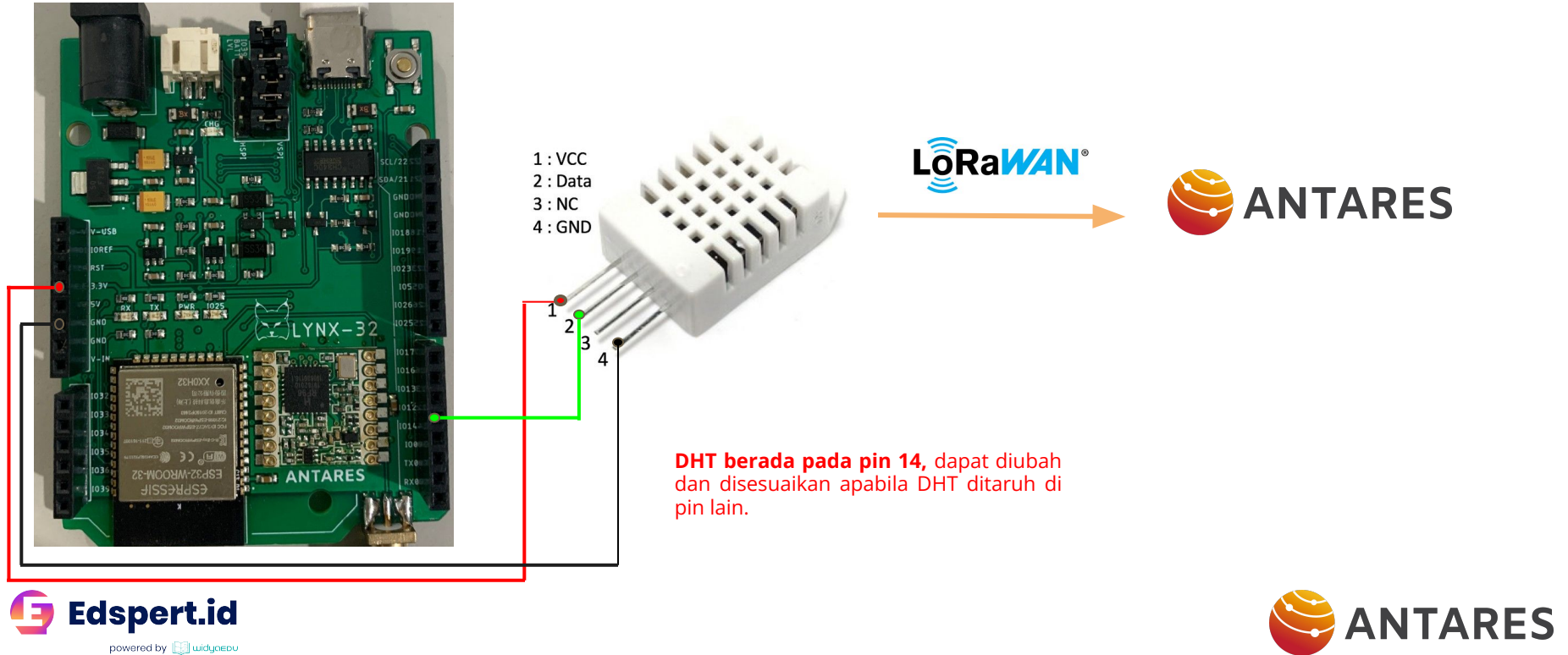
Output Mengirimkan Data Dummy

The screenshot displays the ANTARES web application interface. On the left is a sidebar menu with options: Overview, Applications (highlighted in orange), Widgets (with a Beta badge), Documentation, Account, Packages (with a New badge), and User Keys. The main content area shows a log entry for the timestamp 2022-07-26 14:33:57 and the identifier /antares-cse/cin-qa6nX4bDQPizKHJg. The output is a JSON object representing dummy data for an uplink transmission.

```
{
  "type": "uplink",
  "port": 5,
  "data": "Lora Counter-0",
  "counter": 0,
  "devEui": "56bdfbf34fe71ba4",
  "radio": {
    "gps_time": null,
    "hardware": {
      "snr": 11.8,
      "rssi": -47
    },
    "datarate": 2,
    "modulation": {
      "bandwidth": 125000,
      "spreading": 10
    },
    "delay": 0.02098870277404785,
    "freq": 921.6,
    "size": 27
  }
}
```


Mengirimkan Data Sensor Environment

Mengirimkan Data Sensor Environment



Mengirimkan Data DHT22

1

```
#include <lorawan.h>

//ABP Credentials
const char *devAddr = "49cd811c";
const char *nwksKey = "0594ffbb72f6e0a70000000000000000";
const char *appSKey = "0000000000000000190d124ae0e44a69";

const unsigned long interval = 10000; // 10 s interval to send message
unsigned long previousMillis = 0; // will store last time message sent
unsigned int counter = 0; // message counter

char myStr[50];
byte outStr[255];
byte recvStatus = 0;
int port, channel, freq;
bool newmessage = false;
bool isSent; //DIGUNAKAN APABILA HANYA INGIN MENGIRIM DATA 1x

const sRFM_pins RFM_pins = {
  .CS = 5,
  .RST = 0,
  .DIO0 = 27,
  .DIO1 = 2,
};
```

```
#include "DHTesp.h"

// deklarasi variabel
// set pin yang digunakan
#define DHTPIN 14

// deklarasi object sensor
// set tipe DHT dan pin yang digunakan
DHTesp dht;

void setup() {

  Serial.begin(9600);

  Serial.println("EDSPERT - Akuisisi sensor DHT22 via ESP32");

  // inisiasi sensor DHT
  dht.setup(DHTPIN, DHTesp::DHT22);

  if (!Iora.init()) {
    Serial.println("RFM95 not detected");
    delay(5000);
    return;
  }
}
```

2

Mengirimkan Data DHT22



3

```
// Set LoRaWAN Class change CLASS_A or CLASS_C
lora.setDeviceClass(CLASS_A);

// Set Data Rate
lora.setDataRate(SF10Bw125);

// Set FramePort Tx
lora.setFramePortTx(5);

// set channel to random
lora.setChannel(MULTI);

// Set TxPower to 15 dBi (max)
lora.setTxPower(15);

// Put ABP Key and DevAddress here
lora.setNwkSKey(nwkSKey);
lora.setAppSKey(appSKey);
lora.setDevAddr(devAddr);
}

void loop() {

  /*// Check interval overflow
  if (millis() - previousMillis > interval) {
    previousMillis = millis(); Digunakan apabila ingin looping*/

  if (isSent==false && millis() - previousMillis > interval) {
    previousMillis = millis();
    isSent=true;
    sprintf(myStr, "Lora Counter-%d", counter++); // Mengirim data 1x
```

4

```
TempAndHumidity data = dht.getTempAndHumidity();

// ekstrak data temperature
float temp = data.temperature;
// ekstrak data humidity
float hum = data.humidity;

// menampilkan data di serial
Serial.println("Suhu: " + String(temp, 2) + "°C");
Serial.println("Kelembaban: " + String(hum, 1) + "%");
Serial.println("---");

sprintf(myStr, "Temp:%.2f C,Humidity:%.1f persen", temp, hum);

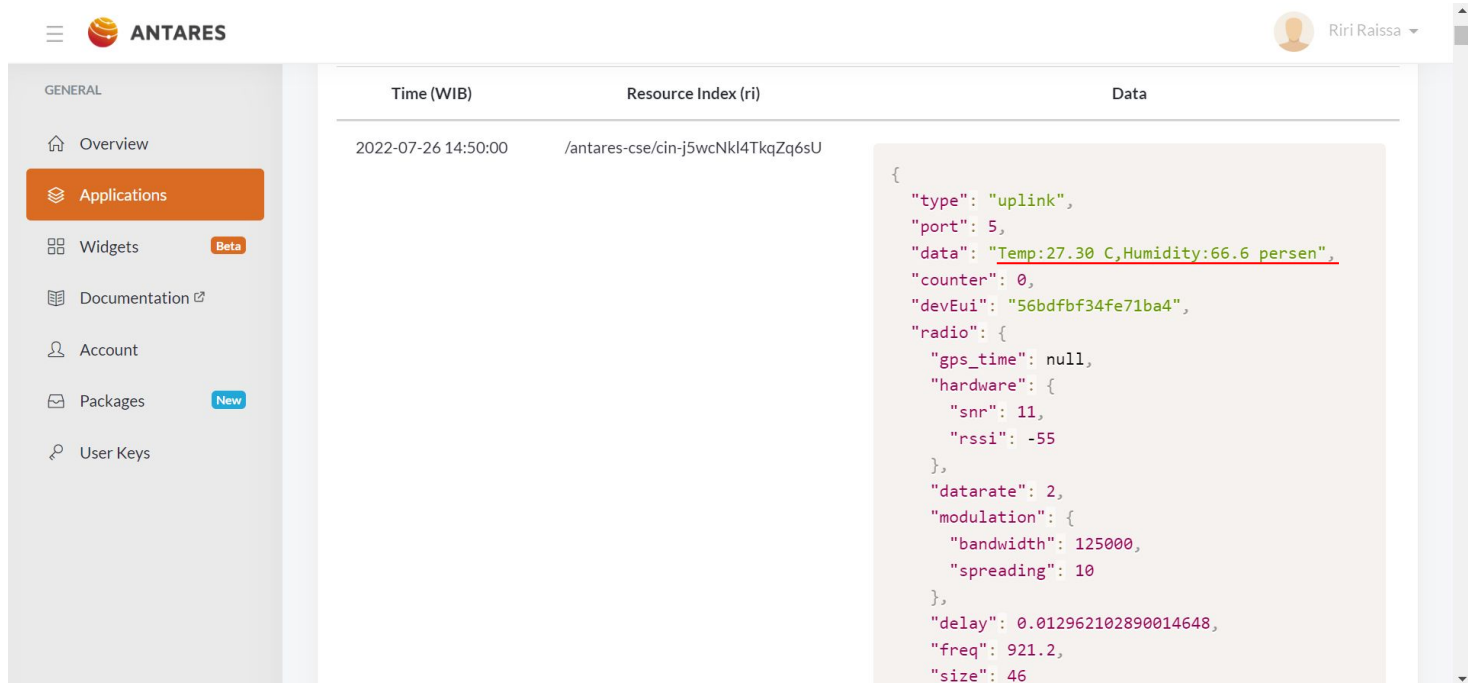
//SET UP LORA
Serial.println(myStr);
lora.sendUplink(myStr, strlen(myStr), 0);

// waktu jeda sampling data
// minimal 2 detik
// delay(2000);

}

// Check Lora RX
lora.update();
recvStatus = lora.readDataByte(outStr);
if (recvStatus) {
  newmessage = true;
  int counter = 0;
}
}
```

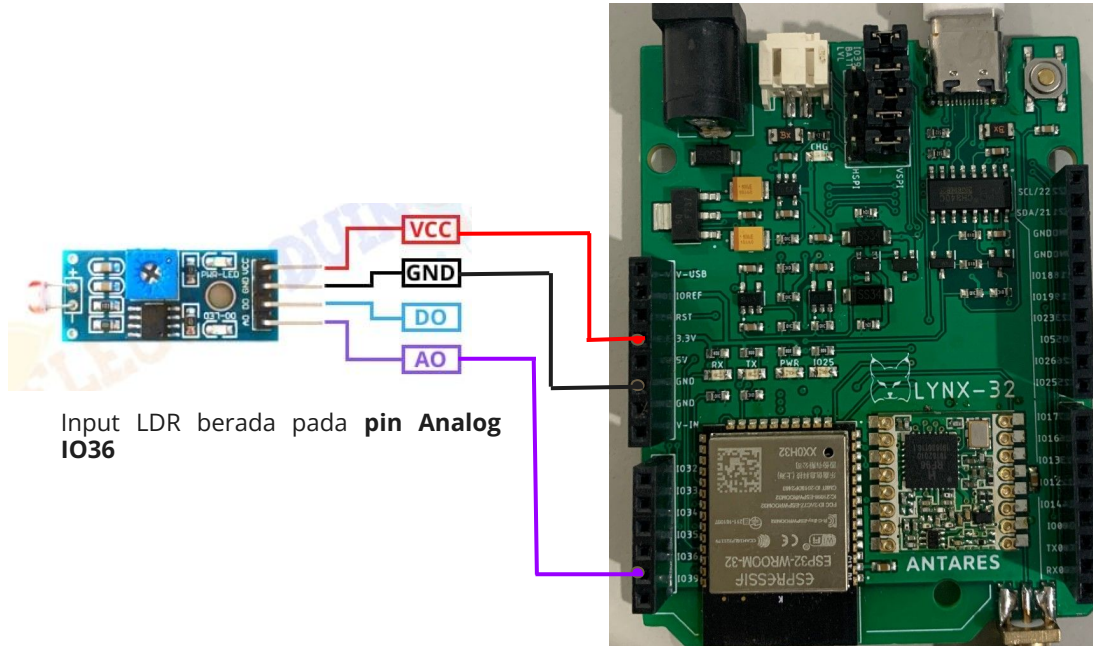
Output Mengirimkan Data Sensor Environment



The screenshot displays the ANTARES web application interface. On the left is a sidebar with a 'GENERAL' section containing links for Overview, Applications (highlighted in orange), Widgets (marked Beta), Documentation, Account, Packages (marked New), and User Keys. The main content area features a table with three columns: Time (WIB), Resource Index (ri), and Data. The table contains one row of data. The 'Data' column shows a JSON object representing sensor data, with the 'data' field value underlined in red.

Time (WIB)	Resource Index (ri)	Data
2022-07-26 14:50:00	/antares-cse/cin-j5wcNkl4TkqZq6sU	<pre>{ "type": "uplink", "port": 5, "data": "Temp:27.30 C,Humidity:66.6 persen", "counter": 0, "devEui": "56bdfbf34fe71ba4", "radio": { "gps_time": null, "hardware": { "snr": 11, "rssi": -55 }, "datarate": 2, "modulation": { "bandwidth": 125000, "spreading": 10 }, "delay": 0.012962102890014648, "freq": 921.2, "size": 46 } }</pre>

Mengirimkan Data Sensor LDR



LoRaWAN®

ANTARES

Mengirimkan Data Sensor LDR

```
#include <lorawan.h>

//ABP Credentials
const char *devAddr = "49cd811c";
const char *nwksKey = "0594ffbb72f6e0a70000000000000000";
const char *appSKey = "0000000000000000190d124ae0e44a69";

const unsigned long interval = 500; // 5 s interval to send message
unsigned long previousMillis = 0; // will store last time message sent
unsigned int counter = 0; // message counter

char myStr[50];
byte outStr[255];
byte rcvStatus = 0;
int port, channel, freq;
bool newmessage = false;
bool isSent; //DIGUNAKAN APABILA HANYA INGIN MENGIRIM DATA 1x

const sRFM_pins RFM_pins = {
    .CS = 5,
    .RST = 0,
    .DIO0 = 27,
    .DIO1 = 2,
};

//Penggunaan LDR
const int pinSensor = A0;

void setup()
{
    // inisiasi Serial comm dengan baud rate 9600
    Serial.begin(9600);
    pinMode(pinSensor, INPUT);
}
```

```
if (!lora.init()) {
    Serial.println("RFM95 not detected");
    delay(500);
    return;
}

// Set LoRaWAN Class change CLASS_A or CLASS_C
lora.setDeviceClass(CLASS_A);

// Set Data Rate
lora.setDataRate(SF10BW125);

// Set FramePort Tx
lora.setFramePortTx(5);

// set channel to random
lora.setChannel(MULTI);

// Set TxPower to 15 dBi (max)
lora.setTxPower(15);

// Put ABP Key and DevAddress here
lora.setNwkSKey(nwksKey);
lora.setAppSKey(appSKey);
lora.setDevAddr(devAddr);
}

void loop() {

    // Check interval overflow
    /*if (millis() - previousMillis > interval) {
        previousMillis = millis(); Digunakan apabila ingin looping*/
}
```

2




1

Mengirimkan Data Sensor LDR

3

```
if (isSent==false && millis() - previousMillis > interval) {  
    previousMillis = millis();  
    isSent=true;  
    sprintf(myStr, "Lora Counter-%d", counter++); //kalo mau ngirim cmn 1x  
  
    int analogValue = analogRead(A0);  
  
    //Print  
    Serial.println("ADC Value:");  
    Serial.println(analogValue, DEC); // prints the value read  
  
    sprintf(myStr, "ADC Value: %d", analogValue);  
  
    //SET UP LORA  
    Serial.println(myStr);  
    lora.sendUplink(myStr, strlen(myStr), 0);  
  
    // CHECK LORA RX  
    lora.update();  
    recvStatus = lora.readDataByte(outStr);  
    if (recvStatus) {  
        newmessage = true;  
        int counter = 0;  
    }  
}
```


Output Mengirimkan Data Sensor LDR Gelap

 **ANTARES**

Riri Raissa ▾

GENERAL

Overview

Applications

Widgets Beta

Documentation ↗


Account

Packages New

User Keys

Time (WIB)	Resource Index (ri)	Data
2022-07-26 14:58:28	/antares-cse/cin-AJkF7nAgSaeXM1sq	<pre>{ "type": "uplink", "port": 5, "data": "ADC Value: 3467", "counter": 0, "devEui": "56bdfbf34fe71ba4", "radio": { "gps_time": null, "hardware": { "snr": 12.8, "rssi": -49 }, "datarate": 2, "modulation": { "bandwidth": 125000, "spreading": 10 }, "delay": 0.023694992065429688, "freq": 921.2, "size": 28 } }</pre>

Output Mengirimkan Data Sensor LDR Terang

 **ANTARES**

Riri Raissa ▾

GENERAL

Overview

Applications

Widgets Beta

Documentation ↗

Account

Packages New

User Keys

Time (WIB)	Resource Index (ri)	Data
2022-07-26 14:59:44	/antares-cse/cin-x12sjJNTQn6tw4L3	<pre>{ "type": "uplink", "port": 5, "data": "ADC Value: 0", "counter": 0, "devEui": "56bdfbf34fe71ba4", "radio": { "gps_time": null, "hardware": { "snr": 11.8, "rssi": -39 }, "datarate": 2, "modulation": { "bandwidth": 125000, "spreading": 10 }, "delay": 0.01218867301940918, "freq": 921.4, "size": 25 } }</pre>

TUGAS

Kirimkan data sensor DHT22 dan LDR secara bersamaan ke 1 device ANTARES

HINT

Gabungkan kode sesi 2 dan sesi 3