



PRESENTASI SEMINAR HASIL

**Rancang Bangun Early-Warning System untuk Daerah
Rawan Bencana Gerakan Tanah**

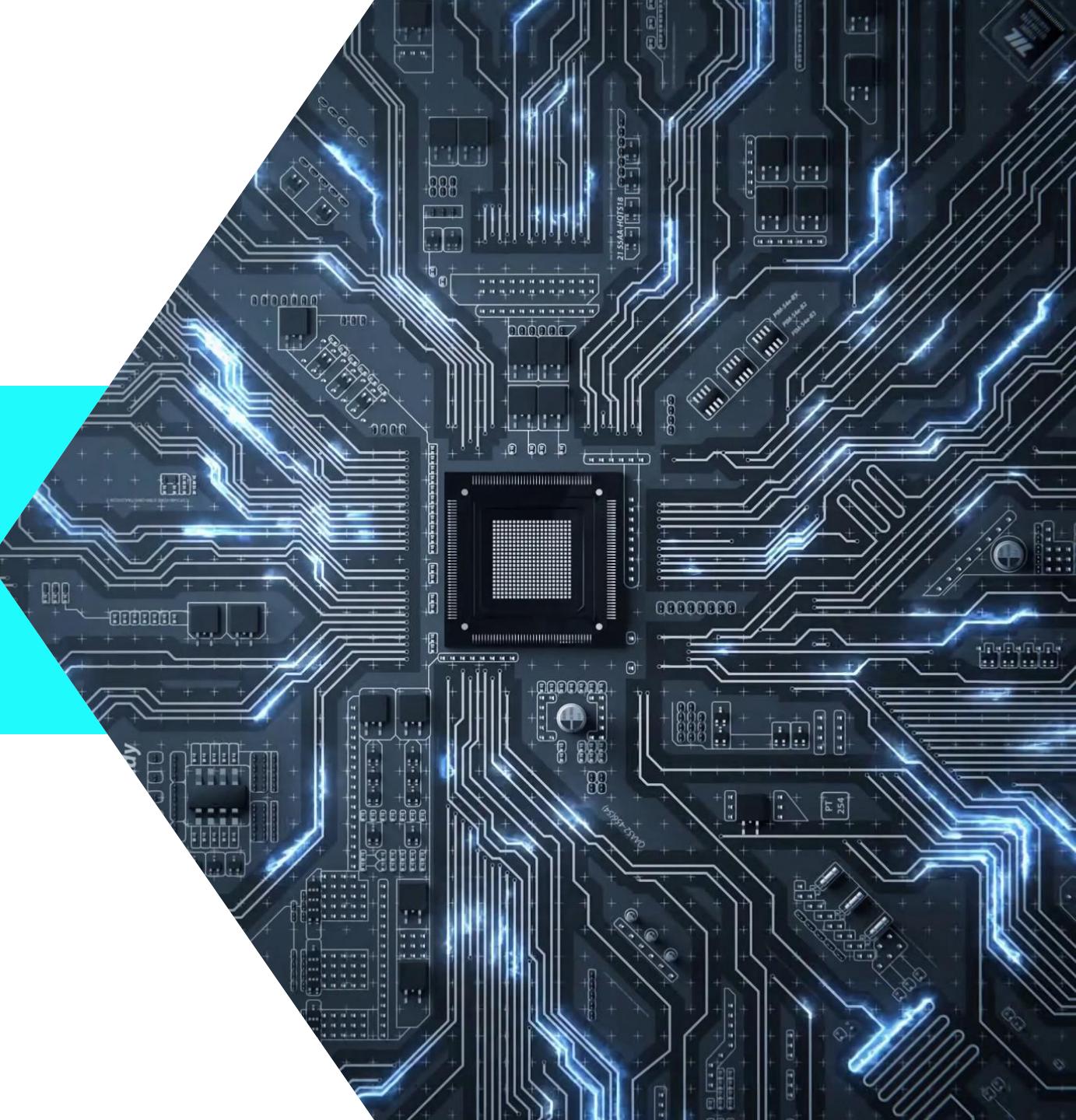
Avif Maulana Azis
140710180048

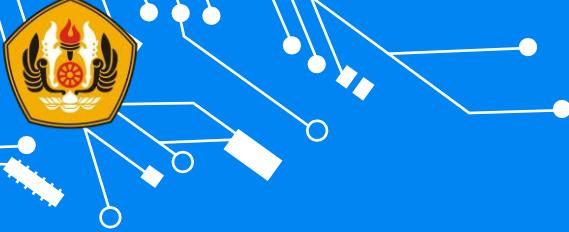
Dosen Pembimbing Utama
Prof. Dr. rer. nat. Yudi Rosandi M. Si.



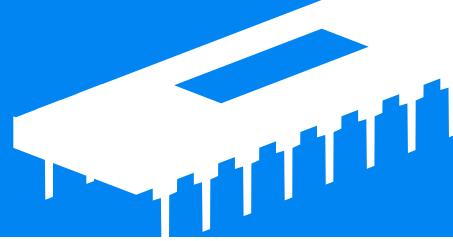
BAB I

PENDAHULUAN





Latar Belakang



Internet of Things System

Internet of things adalah suatu konsep dimana sebuah objek memiliki kemampuan untuk mentransmisikan data melalui jaringan tanpa menggunakan bantuan perangkat komputer dan manusia.



Urgensi Mitigasi : Sistem Peringatan Dini

Sistem peringatan dini adalah sebuah rangkaian sistem yang berfungsi memberikan informasi adanya indikasi akan suatu kejadian (bencana) yang akan terjadi.

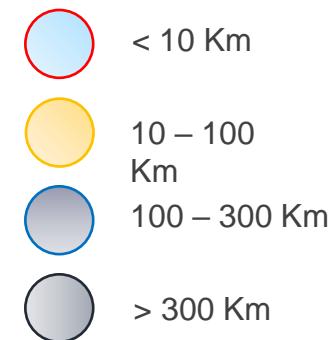
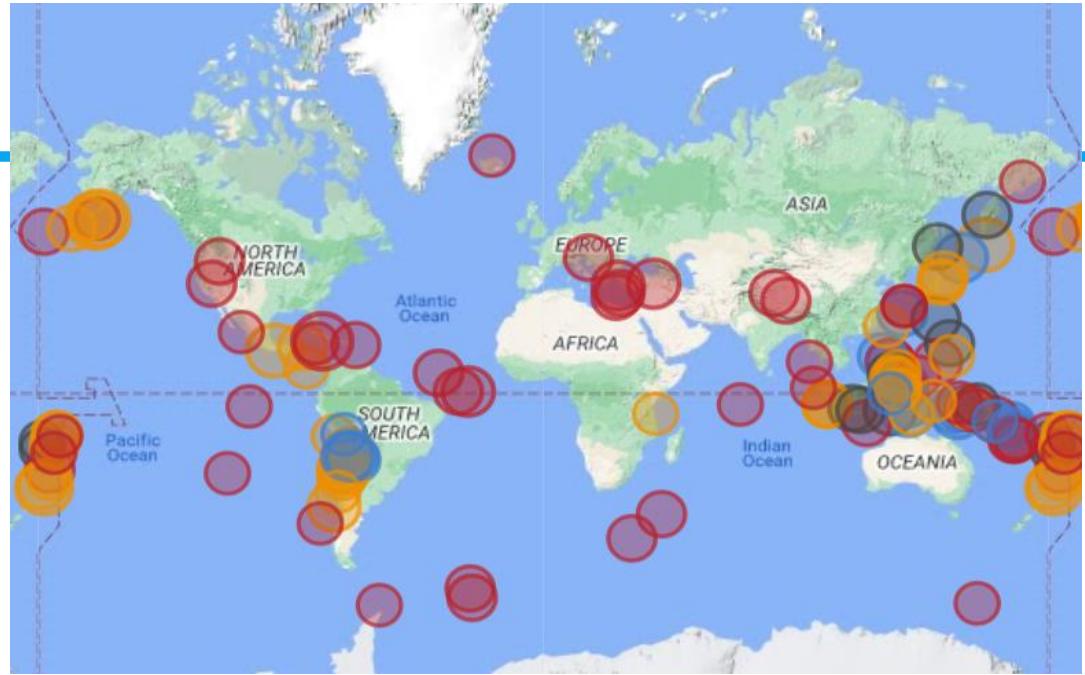
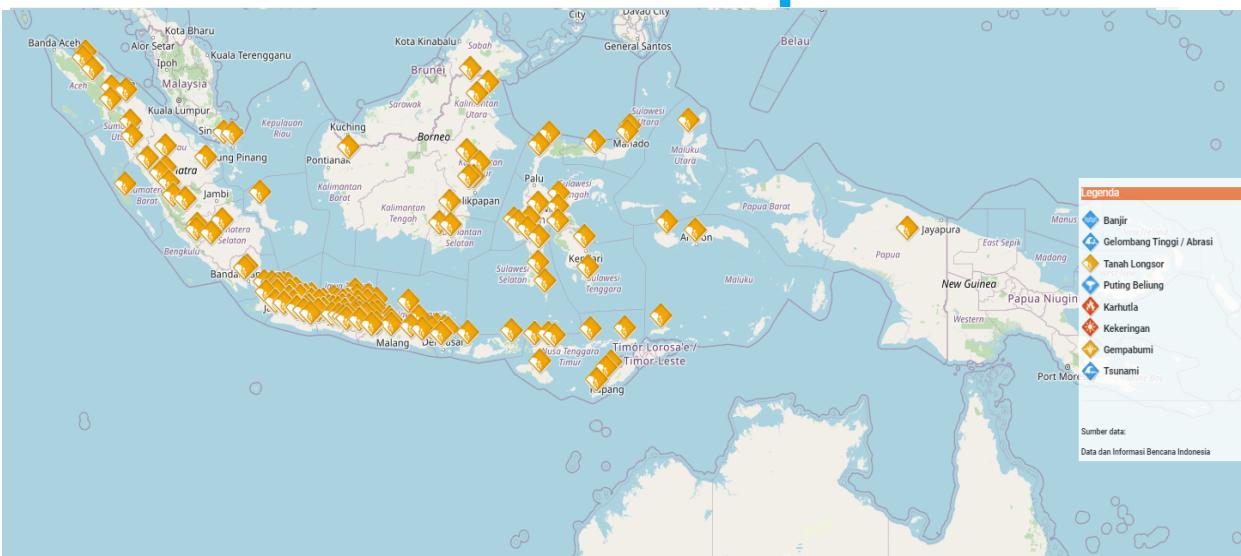
Kondisi Tektonik & Kebencanaan di Indonesia



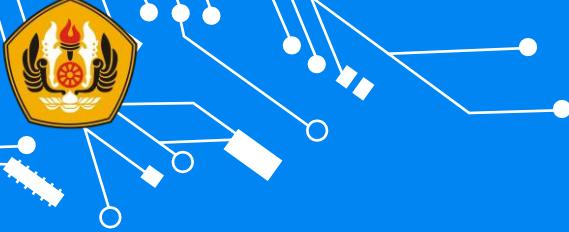


Data Bencana Gerakan Tanah Di Indonesia

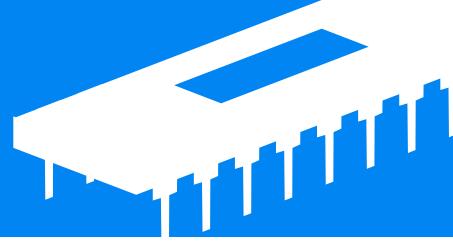
Data Longsor
Indonesia Juli
2020 – Juni 2021
(BNPB, 2021)



- Data gempa bumi dunia tahun 2020. (VolcanoDiscovery, 2021).



Latar Belakang



Internet of Things System

Internet of things adalah suatu konsep dimana sebuah objek memiliki kemampuan untuk mentransmisikan data melalui jaringan tanpa menggunakan bantuan perangkat komputer dan manusia.



Urgensi Mitigasi : Sistem Peringatan Dini

Sistem peringatan dini adalah sebuah rangkaian sistem yang berfungsi memberikan informasi adanya indikasi akan suatu kejadian (bencana) yang akan terjadi.

Kondisi Tektonik & Kebencanaan di Indonesia





Identifikasi Masalah



Terbatasnya sistem peringatan dini secara aktual dan nirkabel yang mendeteksi potensi bencana gerakan tanah.



Maksud Penelitian



1. Membuat suatu sistem peringatan dini yang aktual dan nirkabel dalam upaya meminimalisir risiko bencana yang akan diakibatkan oleh gerakan tanah.
2. Menentukan parameter ambang batas yang akan diimplementasikan pada sistem peringatan dini ini.

Tujuan Penelitian



1. Terbentuknya suatu sistem peringatan dini deteksi bencana gerakan tanah secara aktual serta nirkabel.
2. Mendapatkan parameter ambang batas yang cocok untuk sistem peringatan dini deteksi bencana gerakan tanah.



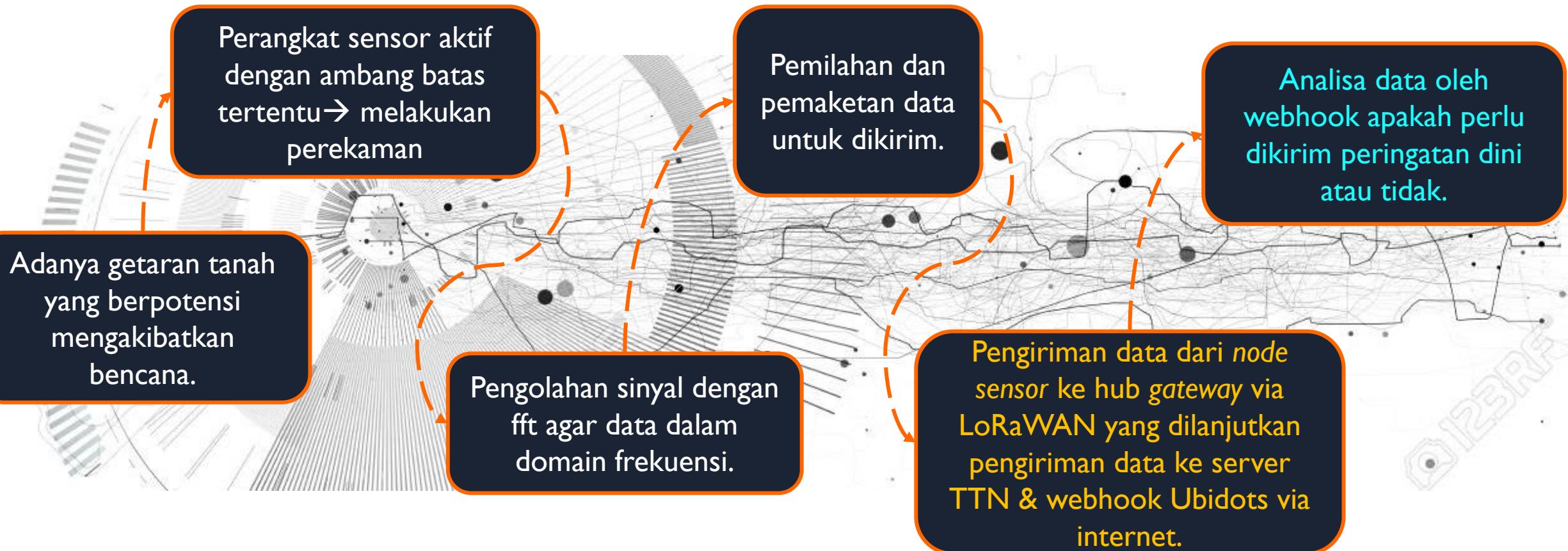
Kegunaan Penelitian



1. Mampu menginfokan peringatan indikasi akan terjadinya bencana yang diakibatkan oleh gerakan tanah secara aktual.
2. Mereduksi risiko bencana yang akan terjadi.



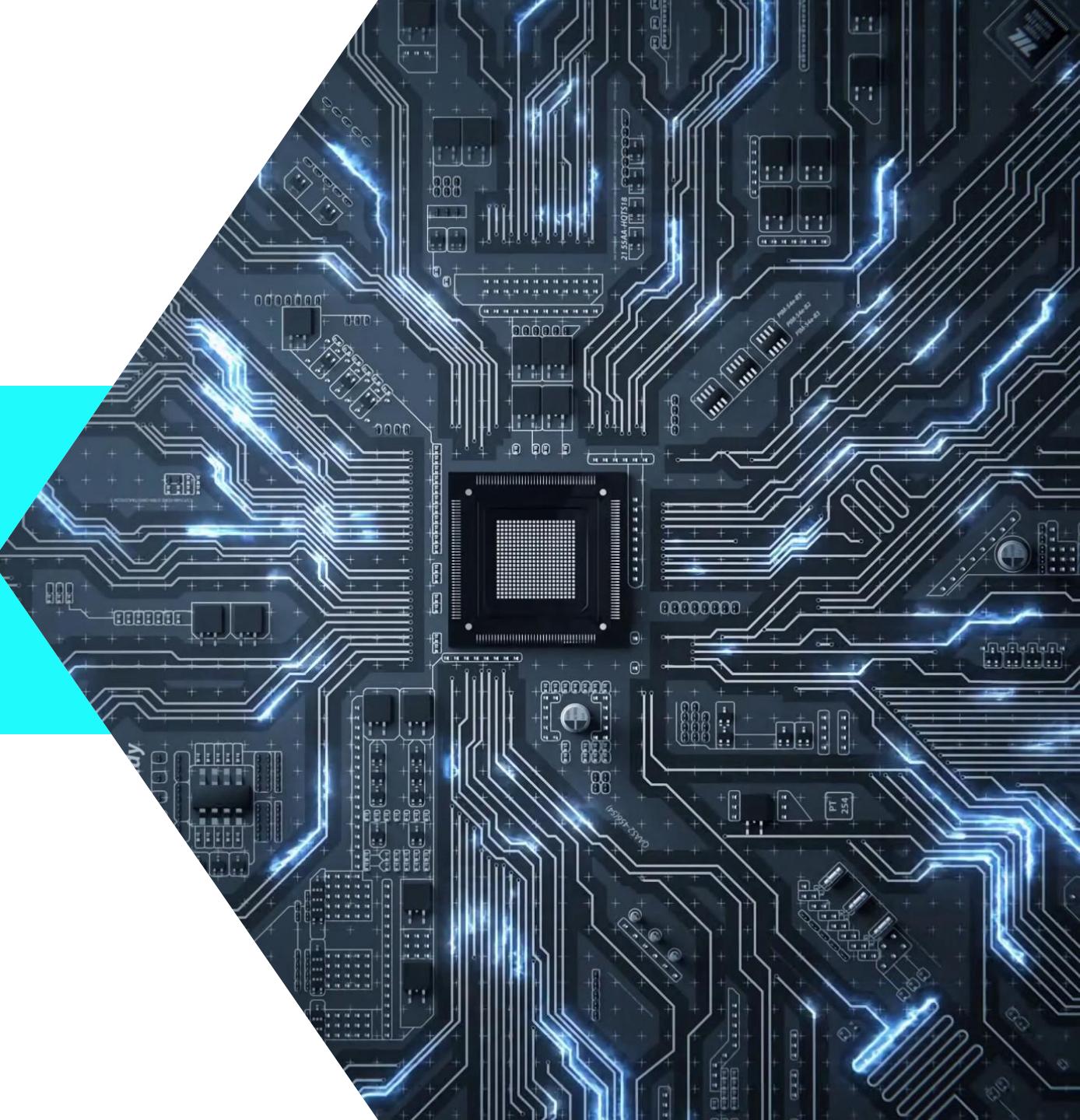
Kerangka Pemikiran

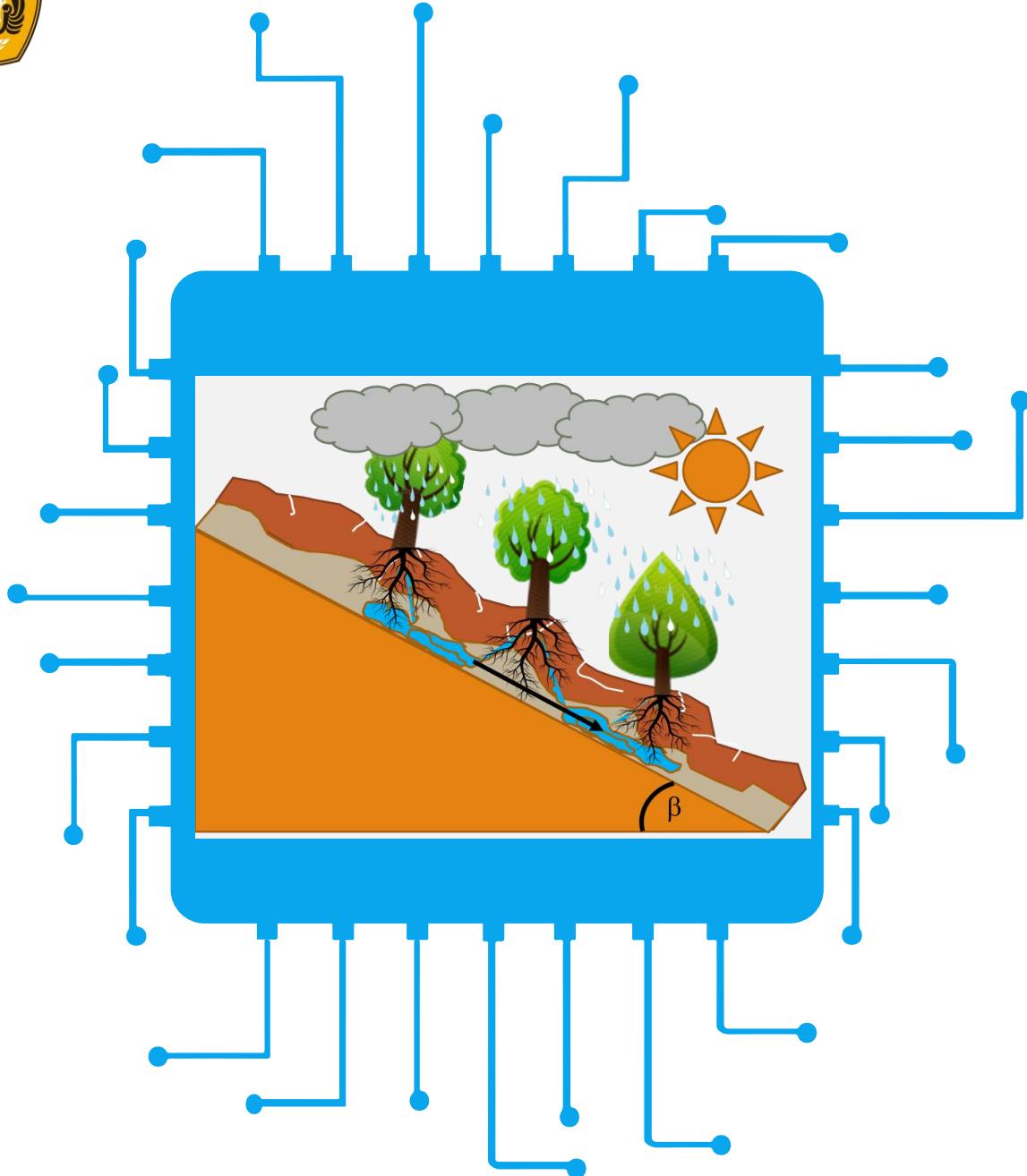




BAB II

TINJAUAN PUSTAKA



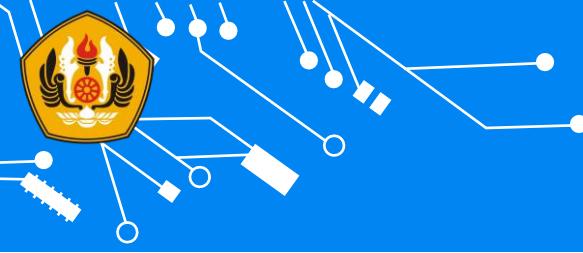


Bencana Gerakan Tanah

Bencana gerakan tanah atau sering kita sebut longsor merupakan pergerakan massa batuan atau tanah menuruni lereng dengan berbagai tipe dan sebab (morphologi, struktur geologi, hidrologi dan tata guna lahan).

Tipe bencana gerakan tanah antara lain yaitu :

- Jatuhan
- Longsoran (slide)
- Aliran
- Rayapan
- Bandang



MITIGASI



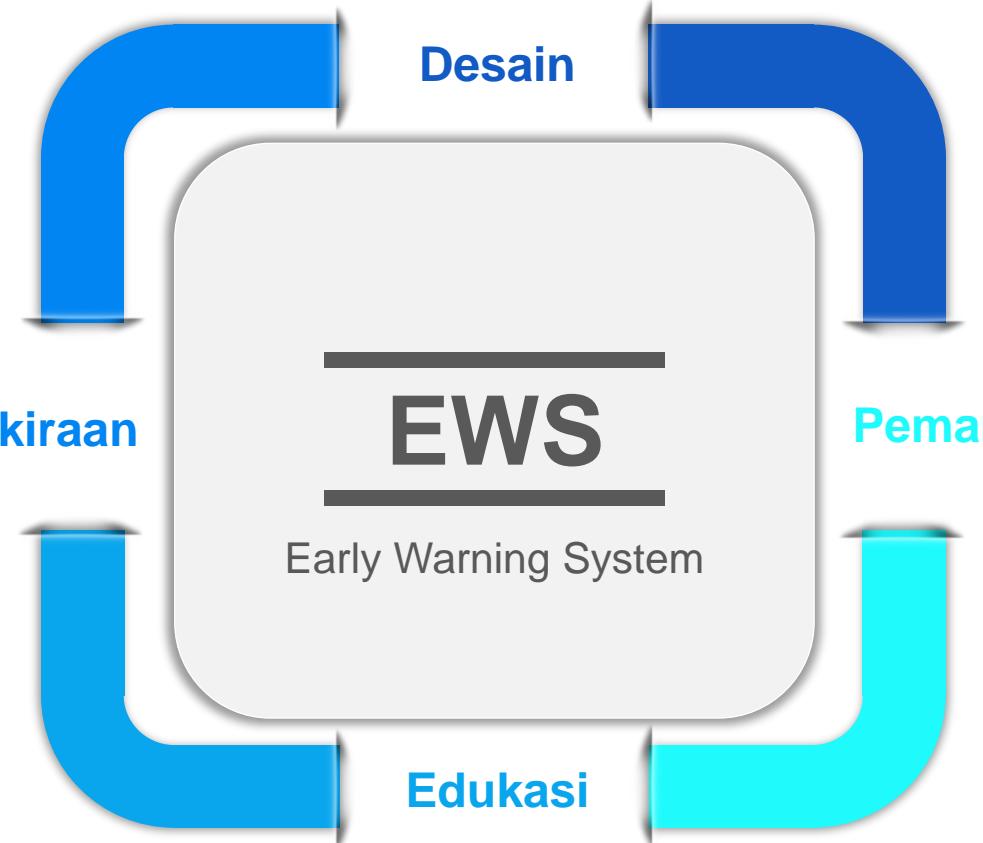
Desain

- Desain Geologi
- Skenario Mitigasi
- Pembuatan Alat dan parameter yang dipantau



Perkiraan

- Interpretasi data
- Penentuan ambang batas
- Metode perkiraan
- Teknis peringatan



Pemantauan

- Instalasi Instrumen
- Perekaman data
- Transmisi data
- Elaborasi data



Edukasi

- Wawasan resiko bencana
- Kebiasaan yang baik
- Respon terhadap peringatan
- Keterlibatan masyarakat

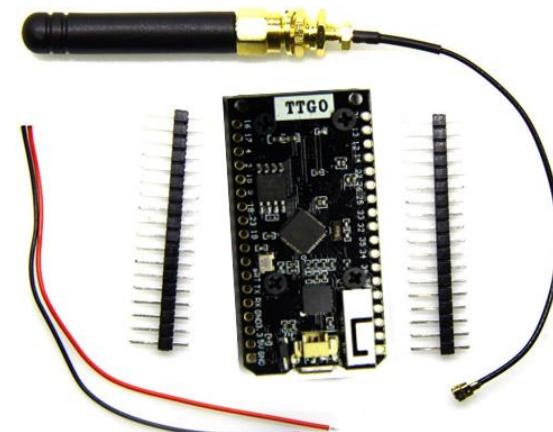




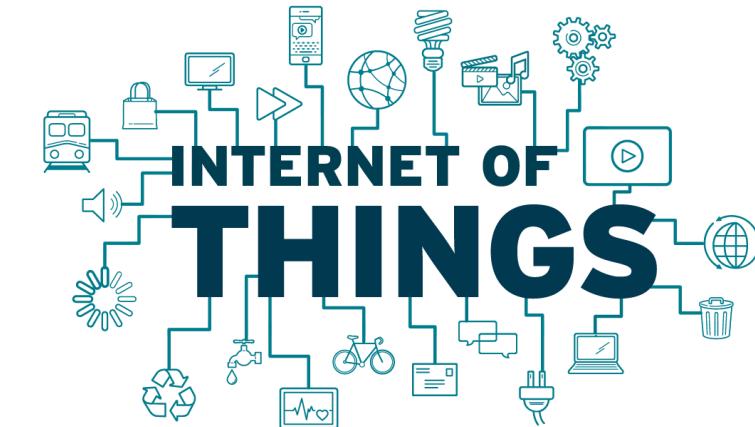
Dragino OLGN-01



Geophone SM-4



ESP TTGO LoRa32 816-915
Mhz



LoRaWAN®



THE THINGS
STACK



THE THINGS
NETWORK



Gmail™
by Google



BAB III

METODOLOGI PENELITIAN

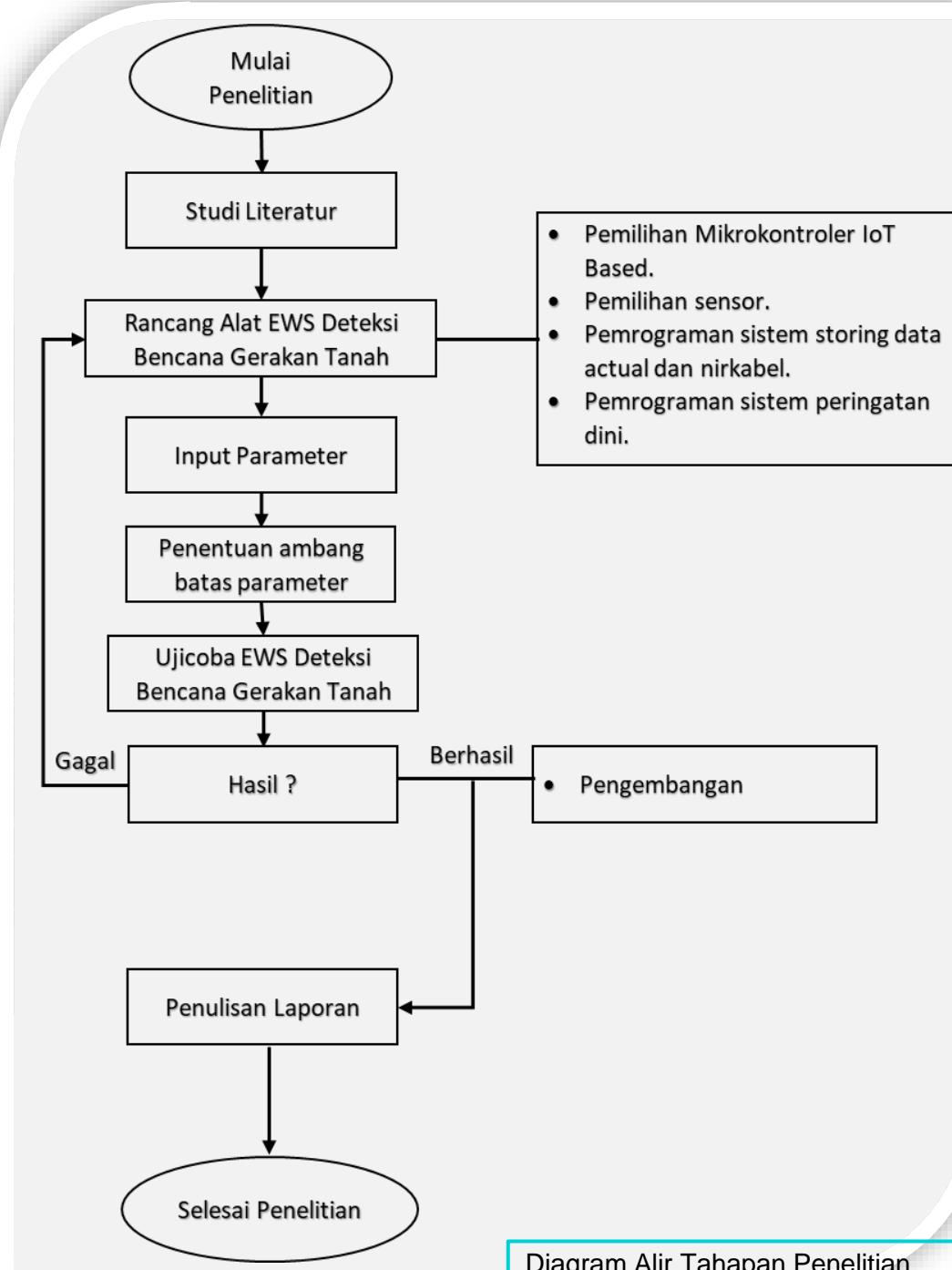
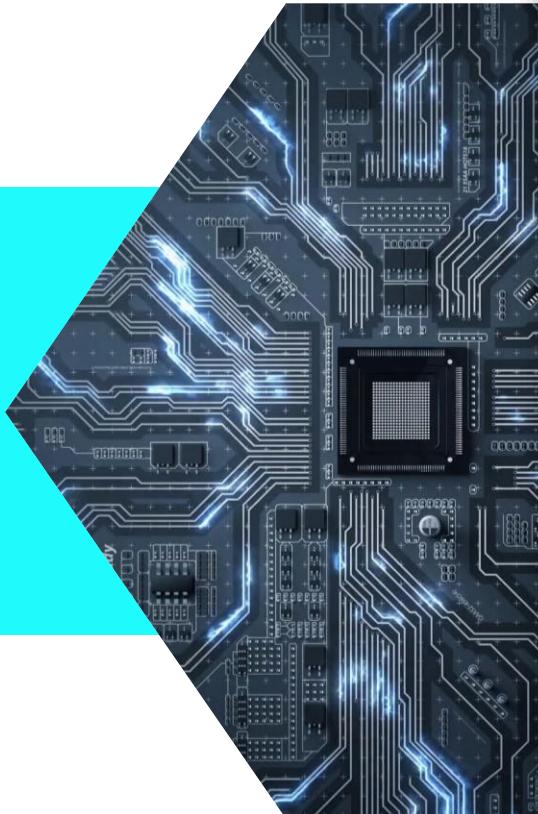
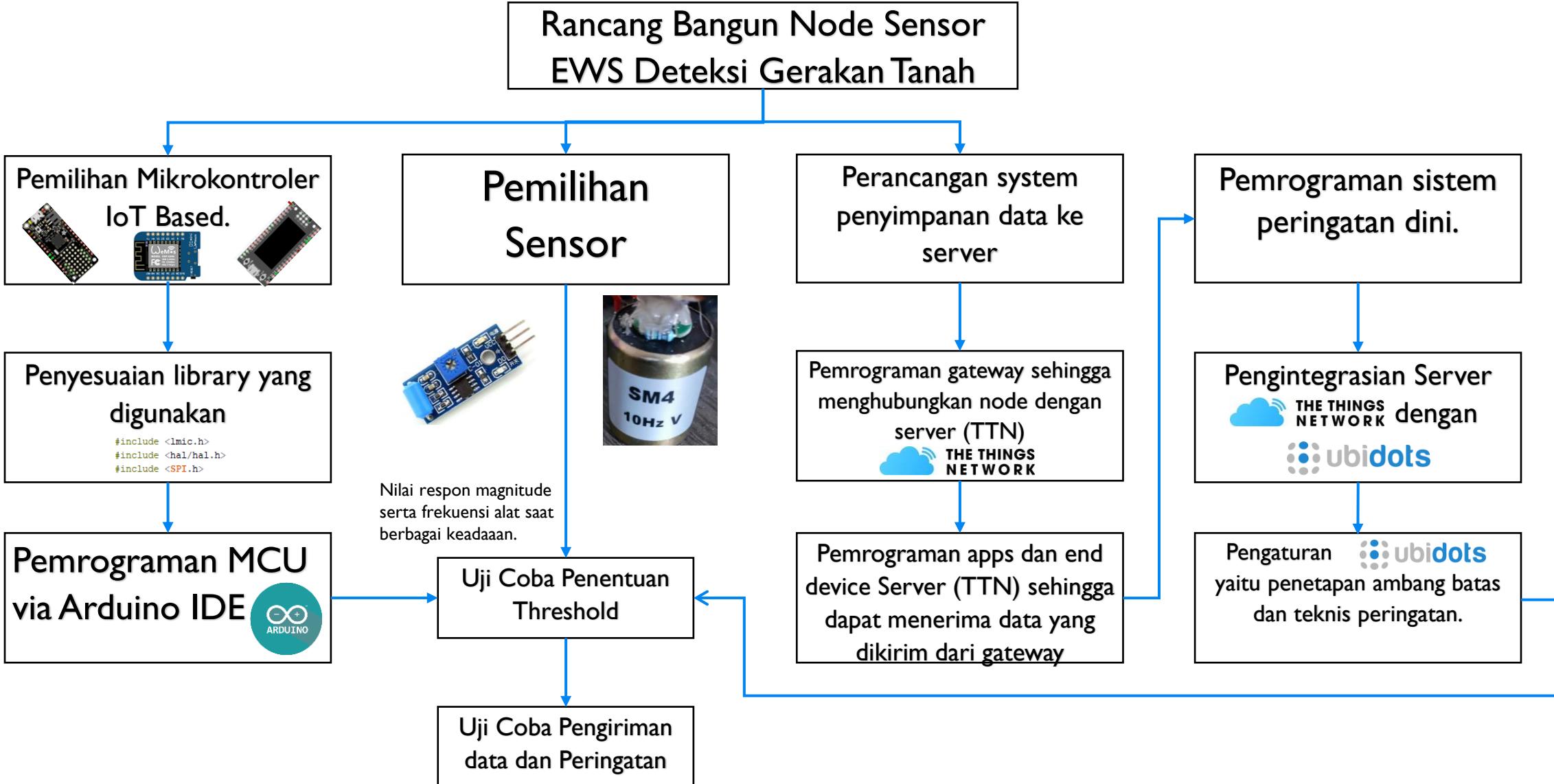


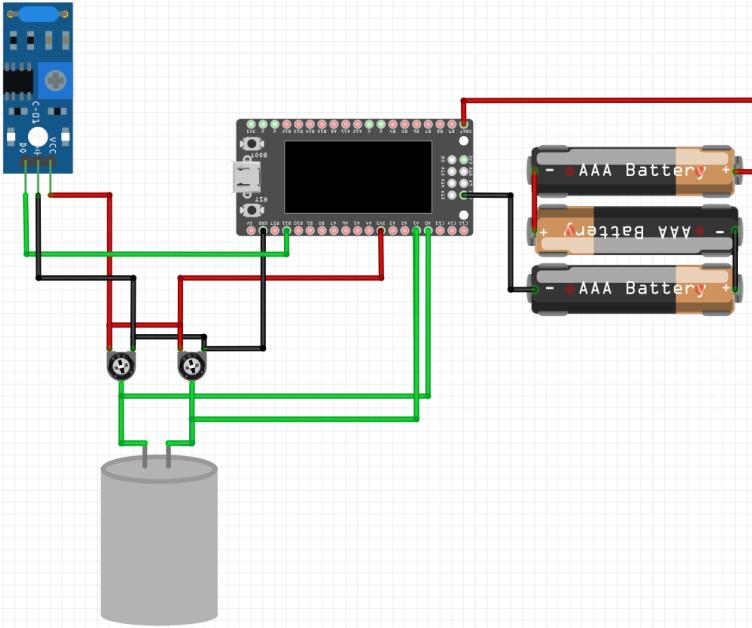


Diagram Alir Rancang Bangun EWS

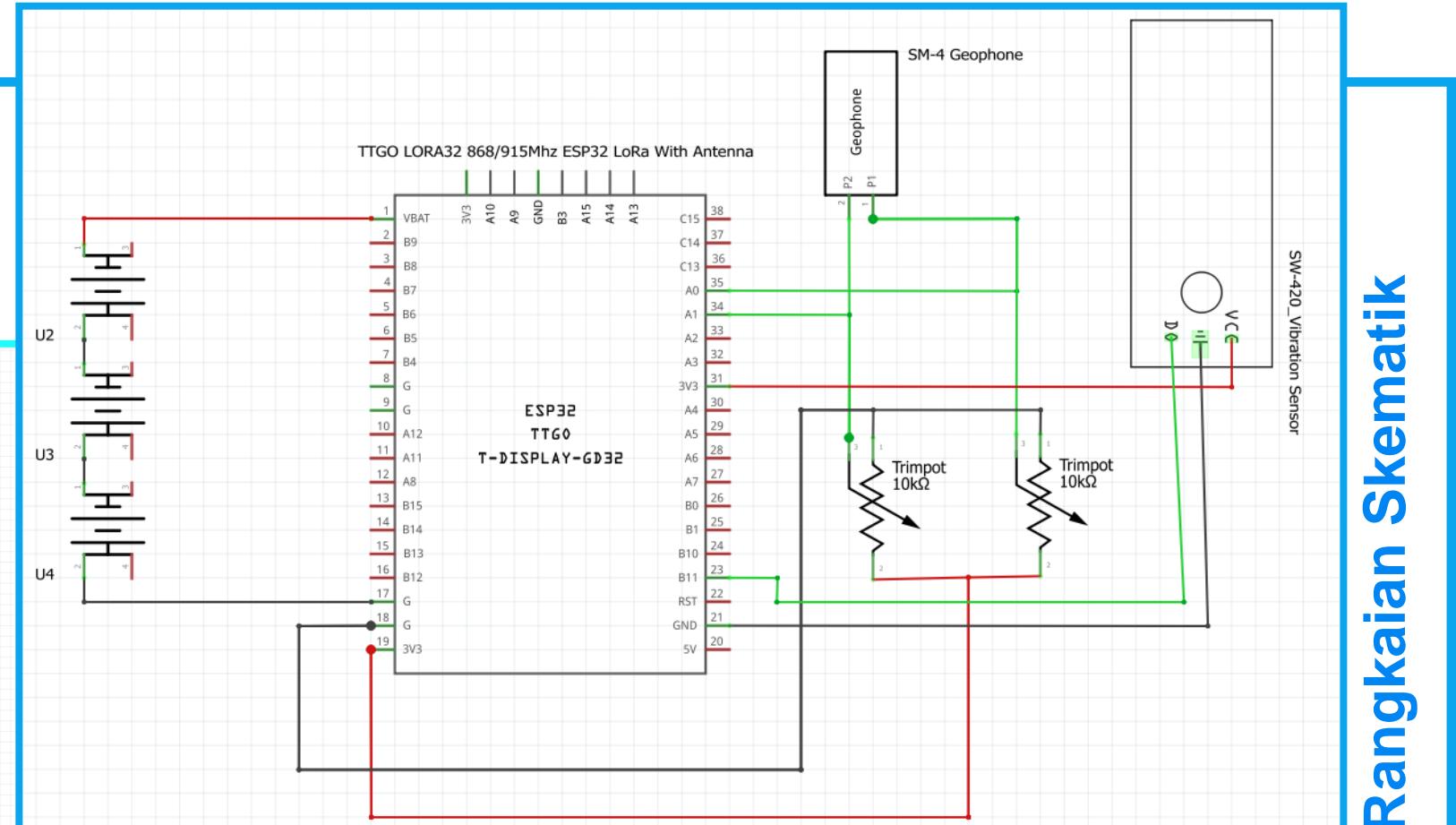




Rangkaian Instrumen



Rangkaian Sketch
Board



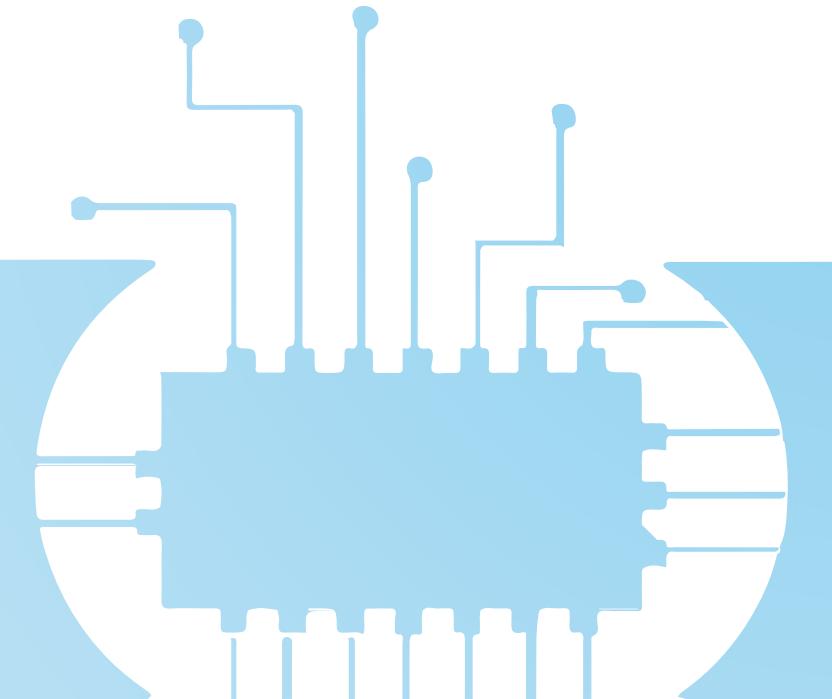
Rangkaian Skematik



Listing Program

```
1. #include <lmic.h>
2. #include <hal/hal.h>
3. #include <SPI.h>
4. #define TIME_TO_SLEEP 3600      /* Time ESP32 will go to
   sleep (in seconds) */
5. int dataset_count = 2;
6. uint16_t amp1,amp2,amp3;
7. float f1, f2, f3;
8. uint8_t state=1;
9. static uint8_t payload[14];
10. static const PROGMEM u1_t NWKSKEY[16] = { 0xB4, 0x01, 0xB2,
    0x28, 0x9F, 0x89, 0x94, 0x01, 0xFB, 0xAF, 0x13, 0x3E, 0xA4,
    0x9E, 0xC9, 0x2D };
11. static const u1_t PROGMEM APPSKEY[16] = { 0xD5, 0x12, 0x63,
    0xFC, 0x63, 0x54, 0xA6, 0xDE, 0x84, 0x07, 0xFC, 0x3F, 0x25,
    0xC4, 0xA2, 0xA9 };
12. static const u4_t DEVADDR = 0x260D739E ; //
13.
14. const lmic_pinmap lmic_pins = {
15.   .nss = 18, // CS PIN
16.   .rxtx = LMIC_UNUSED_PIN,
17.   .rst = 14, // reset pin
18.   .dio = { 26, 33, 32 }
19. };
```

Potongan program inisialisasi library
dan global variabel





Listing Program

```
1. void print_wakeup_reason() {
2.     esp_sleep_wakeup_cause_t wakeup_reason;
3.
4.     wakeup_reason = esp_sleep_get_wakeup_cause();
5.
6.     switch(wakeup_reason)
7.     {
8.         case ESP_SLEEP_WAKEUP_EXTO :
9.             Serial.println("Wakeup caused by external signal using RTC_IO");
10.            state=1;
11.            ESP.restart();
12.            break;
13.        case ESP_SLEEP_WAKEUP_EXT1 : Serial.println("Wakeup caused by external signal using RTC_CNTL"); break;
14.        case ESP_SLEEP_WAKEUP_TIMER : Serial.println("Wakeup caused by timer");
15.            state=0;
16.            break;
17.        case ESP_SLEEP_WAKEUP_TOUCHPAD : Serial.println("Wakeup caused by touchpad"); break;
18.        case ESP_SLEEP_WAKEUP_ULP : Serial.println("Wakeup caused by ULP program"); break;
19.        default : Serial.printf("Wakeup was not caused by deep sleep: %d\n",wakeup_reason); break;
20.    }
21. }
```

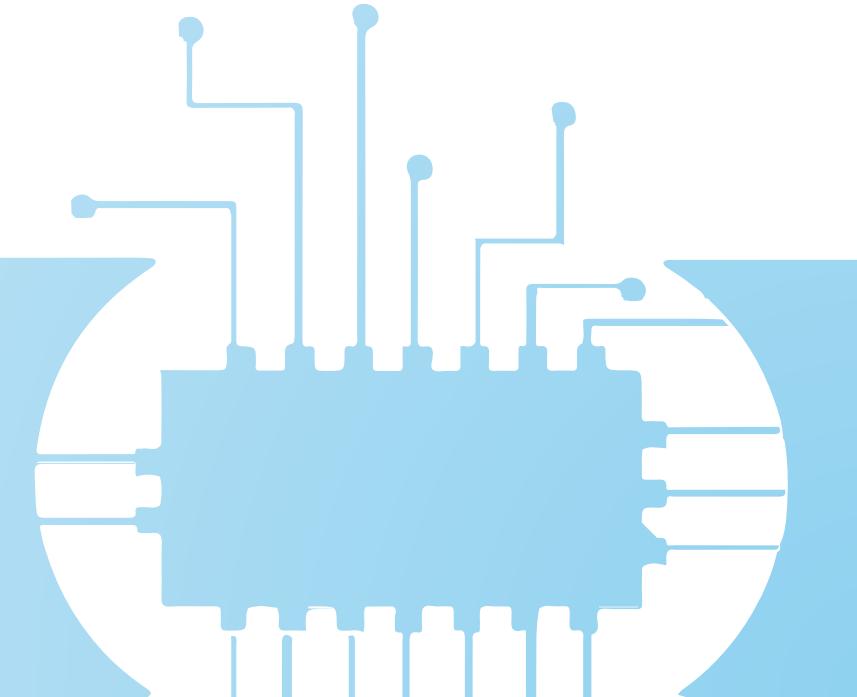
Potongan program fungsi
wakeup_reason



Listing Program

```
1. if(datasent_count == 0)
2. {
3.     esp_sleep_enable_timer_wakeup(TIME_TO_SLEEP *
4.     uS_TO_S_FACTOR);
5.     esp_sleep_enable_ext0_wakeup(GPIO_NUM_15,1); //1 =
High, 0 = Low
6.     //Go to sleep now
7.     Serial.flush();
8.     Serial.println("Going to sleep now");
9. }
10. else {
11.     Serial.print("Need to send : ");
12.     Serial.print(datasent_count);
13.     Serial.println(" more time(s) before sleep.");
14.     delay(100);
15. }
```

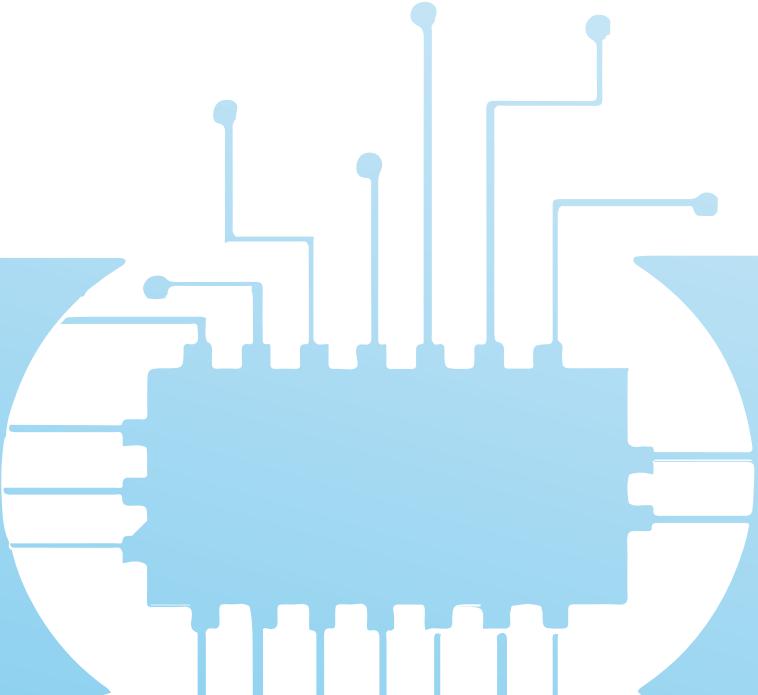
Potongan program fungsi deep sleep



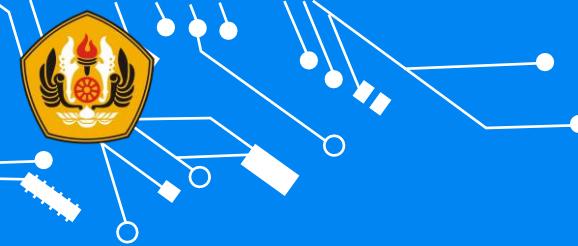


Listing Program

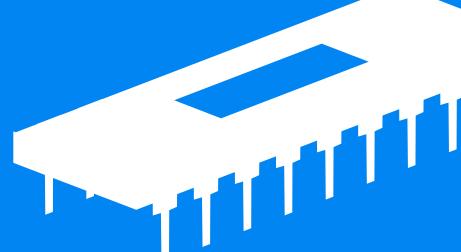
Potongan program fungsi pengiriman data



```
1. void do_send(osjob_t* j) {
2.     // Check if there is not a current TX/RX job running
3.     if (LMIC.opmode & OP_TXRXPEND) {
4.         Serial.println(F("OP_TXRXPEND, not sending"));
5.     } else {
6.         //***** For Acquiring Data *****/
7.         AcquiringData();
8.         //***** For Transferring Data *****/
9.         uint16_t payloadf1 = LMIC_f2sflt16(f1/100);
10.        // int -> bytes
11.        byte f1Low = lowByte(payloadf1);
12.        byte f1High = highByte(payloadf1);
13.        // place the bytes into the payload
14.        payload[0] = f1Low;
15.        payload[1] = f1High;
16.
17.        uint16_t payloadf2 = LMIC_f2sflt16(f2/100);
18.        ....
19.        uint16_t payloadf3 = LMIC_f2sflt16(f3/100);
20.        ...
21.        byte amp1Low = lowByte(amp1);
22.        ....
23.        byte amp2Low = lowByte(amp2);
24.        ....
25.        byte amp3Low = lowByte(amp3);
26.        ...
27.        payload[12] = byte lowByte(state);
28.        ...
29.        LMIC_setTxData2(1, payload, sizeof(payload)-1, 0);
30.        Serial.println(F("Packet queued : "));
31.        ...
32.    }
33. }
```



Pengolahan Data (FFT)



- Transformasi Fourier digunakan untuk mengubah/mentransformasikan sinyal analog dari domain waktu ke domain frekuensi (spektrum frekuensi).
- Dalam sinyal digital yang umumnya merupakan sinyal diskrit, maka digunakanlah DFT (Discrete Fourier Transform) yang telah dikembangkan menggunakan butterfly diagram dan reverse bit pada sistem menjadi FFT (Fast Fourier Transform).
- Rumus spektrum frekuensi dari DFT & FFT :
$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-\frac{i(2\pi kn)}{N}}$$
 Dengan $N=2^n$
- Karena hasil spektrum masih berbentuk bilangan kompleks. Magnitudo spektrum dapat dihitung dengan mencari nilai mutlaknya :

$$|X(k)| = \sqrt{{X(k)_{real}}^2 + {X(k)_{imajiner}}^2}$$



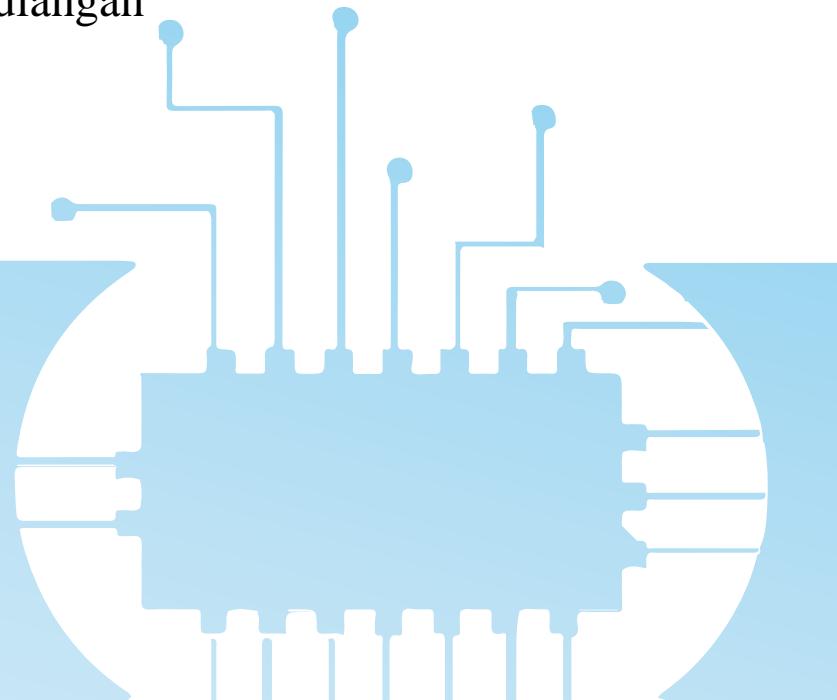
Listing Program

```
1. void setup() {
2.     pinMode(13, OUTPUT);
3.     while (!Serial); // wait for Serial to be initialized
4.     Serial.begin(115200);
5.     Serial.println(F("Starting"));
6.     .....
7.     LMIC_setSession (0x13, DEVADDR, NWKSKEY, APPSKEY);
8.     #endif
9.     #if defined(CFG_eu868)
10.    #elif defined(CFG_kr920)
11.    LMIC_setupChannel(1, 921000000, DR_RANGE_MAP(DR_SF12,
12.        DR_SF7), BAND_MILLI);
13.    #else
14.    #error Region not supported
15.    #endif
16.    // Disable link check validation
17.    LMIC_setLinkCheckMode(0);
18.    // TTN uses SF9 for its RX2 window.
19.    LMIC.dn2Dr = DR_SF9;
20.    // Set data rate and transmit power for uplink
21.    LMIC_setDrTxpow(DR_SF7, 14);
22.    // Start job
23.    do_send(&sendjob);
24. }
```

```
• void loop() {
•     unsigned long now;
•     now = millis();
•     Os_runloop_once();
• }
```

Potongan Program fungsi loop / pengulangan

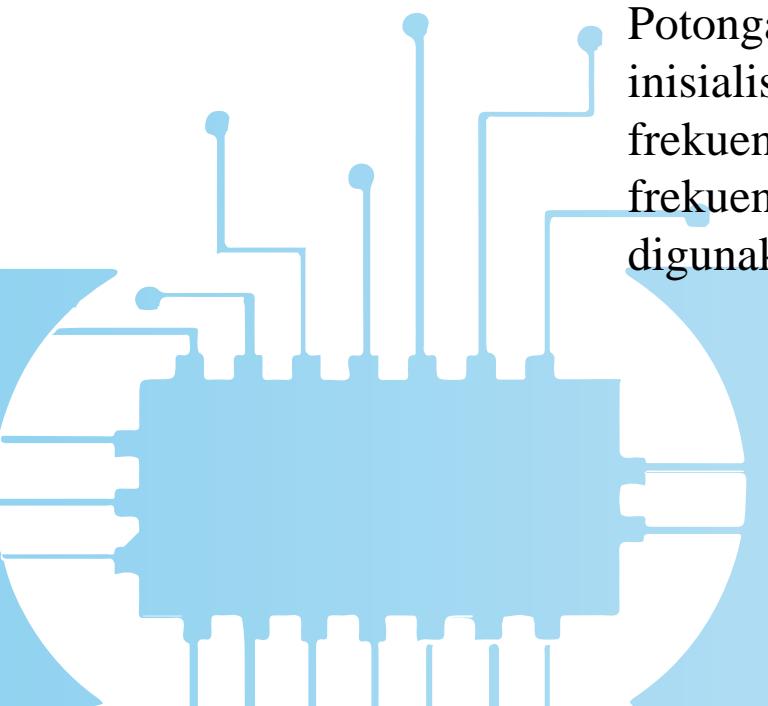
Potongan program fungsi Setup





Listing Program

```
1. #ifndef _lmic_config_h_
2. #define _lmic_config_h_
3.
4. #if CFG_LMIC_REGION_MASK == 0
5. # define CFG_kr920 1
6. //# define CFG_eu868 1
```

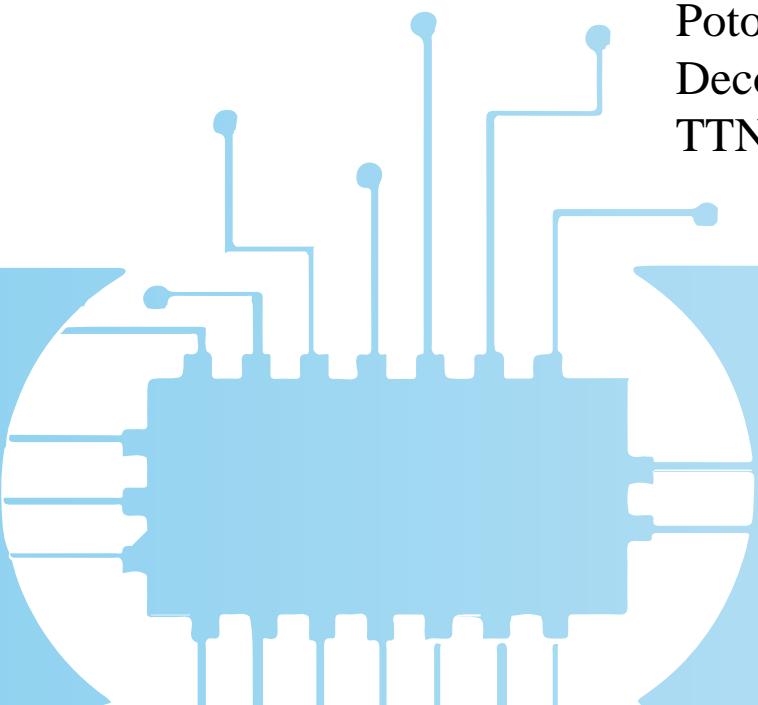


Potongan program fungsi inisialisasi domain frekuensi serta nilai dari frekuensi jaringan yang digunakan

```
1. #ifndef _lorabase_kr920_h_
2. #define _lorabase_kr920_h_
3.
4. #ifndef _LMIC_CONFIG_PRECONDITIONS_H_
5. # include "lmic_config_preconditions.h"
6. #endif
7.
8. .....
9. // freq datarates
10.enum {
11.     KR920_F1 = 921000000, // SF7-12 (DR0-5)
12.     KR920_F2 = 921000000, // SF7-12 (DR0-5)
13.     KR920_F3 = 921000000, // SF7-12 (DR0-5)
14.     KR920_FBCN = 921000000, // beacon/ping
15.     KR920_F14DBM = 921000000, // Allows 14 dBm
16.     KR920_FDOWN = 921000000, // RX2 downlink
17.     frequency
18. };
19. enum {
20.     KR920_FREQ_MIN = 920900000,
21.     KR920_FREQ_MAX = 923300000
22. };
```



Listing Program

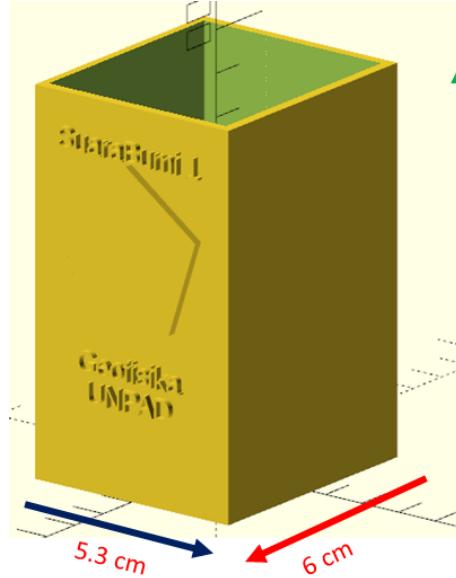


Potongan program fungsi
Decoding pada Server
TTN

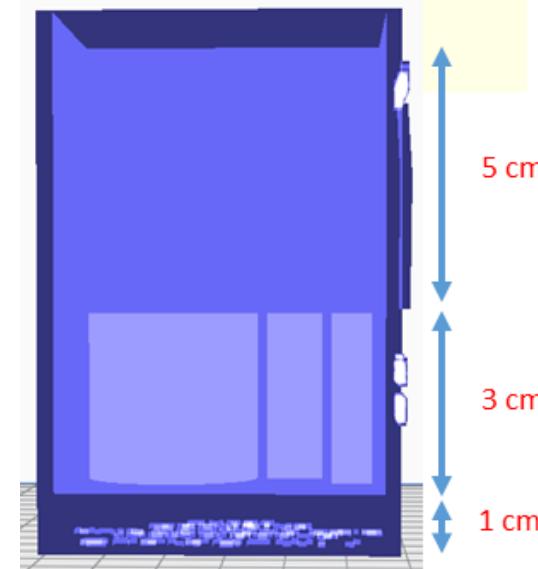
```
1. function Decoder(bytes, port) {  
2. // Decode an uplink message from a buffer  
3. // (array) of bytes to an object of fields.  
4. var decoded = {};  
5. // f1  
6. rawf1 = bytes[0] + bytes[1] * 256;  
7. f1 = sflt162f(rawf1) * 100,  
8. decoded.f1 = +f1.toFixed(2);  
9. // f2  
10. rawf2 = bytes[2] + bytes[3] * 256;  
11. f2 = sflt162f(rawf2) * 100,  
12. decoded.f2 = +f2.toFixed(2);  
13. // f3  
14. rawf3 = bytes[4] + bytes[5] * 256;  
15. f3 = sflt162f(rawf3) * 100,  
16. decoded.f3 = +f3.toFixed(2);  
17. decoded.A1 = (bytes[7] << 8)+ bytes[6];  
18. decoded.A2 = (bytes[9] << 8)+ bytes[8];  
19. decoded.A3 = (bytes[11] << 8)+ bytes[10];  
20. decoded.State = (bytes[13] << 8)+ bytes[12];  
21. return decoded;
```



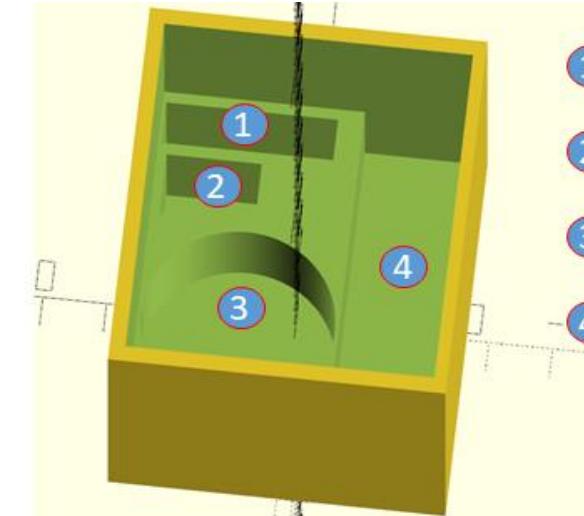
Casing Node Sensor



Tampak Luar



Tampak x-ray dari samping



Tampak dalam dari depan

Sensor's Casing

Bahan pembuatan casing menggunakan PLA+
Proses pembuatan dengan Printer 3D Ender 5-pro
Proses Desain menggunakan aplikasi Openscad dan Ultimaker Cura



Konfigurasi Gateway

Not secure | suarabumi00/cgi-bin/luci/admin/gateway/gateway

suarabumi00 Status System Network Service Logout

Single Channel LoRa Gateway

Configuration to communicate with LoRa devices and LoRaWAN server

LoRaWAN Server Settings

IoT Service	Process LoRa Data via customiz
Debug Level	Little message output
Service Provider	The Things Network
Server Address	ttn-router-asia-se
Server Port	1700
Gateway ID	a840411db54c4150
Mail Address	dragino-1db54c@dragino.com
Latitude	-6.926678
Longitude	107.77411
Radio Power (Unit: dBm)	range 5 ~ 20 dBm

Radio Settings

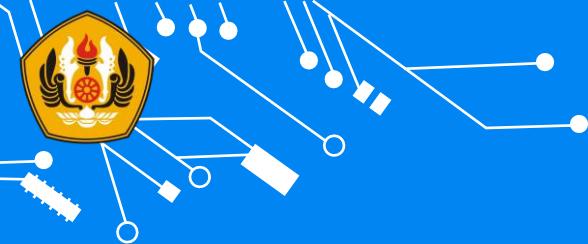
Radio settings for Channel

Frequency (Unit:Hz)	921000000
Spreading Factor	SF7
Coding Rate	4/5
Signal Bandwidth	125 kHz
Preamble Length	8

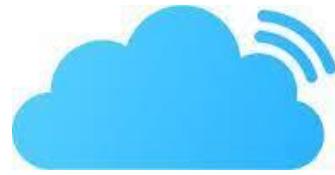
Length range: 6 ~ 65536

S.P = Custom
S.A = http://au1.cloud.thethings.network/

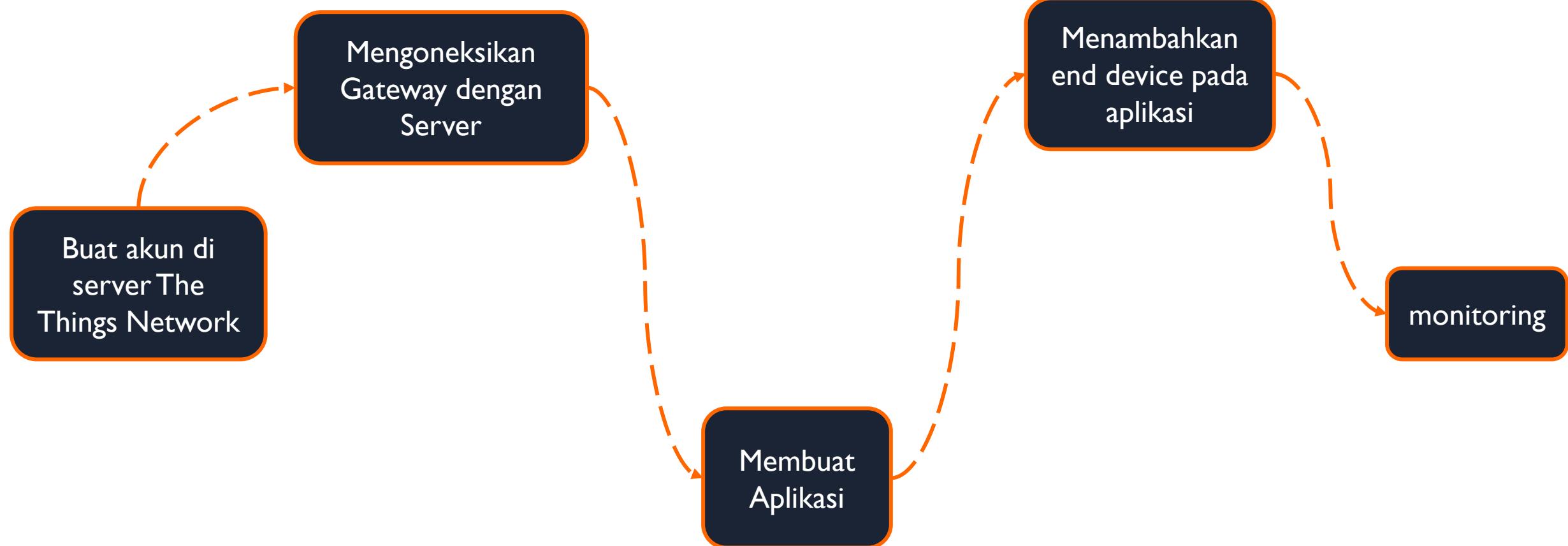




Pengaturan Server

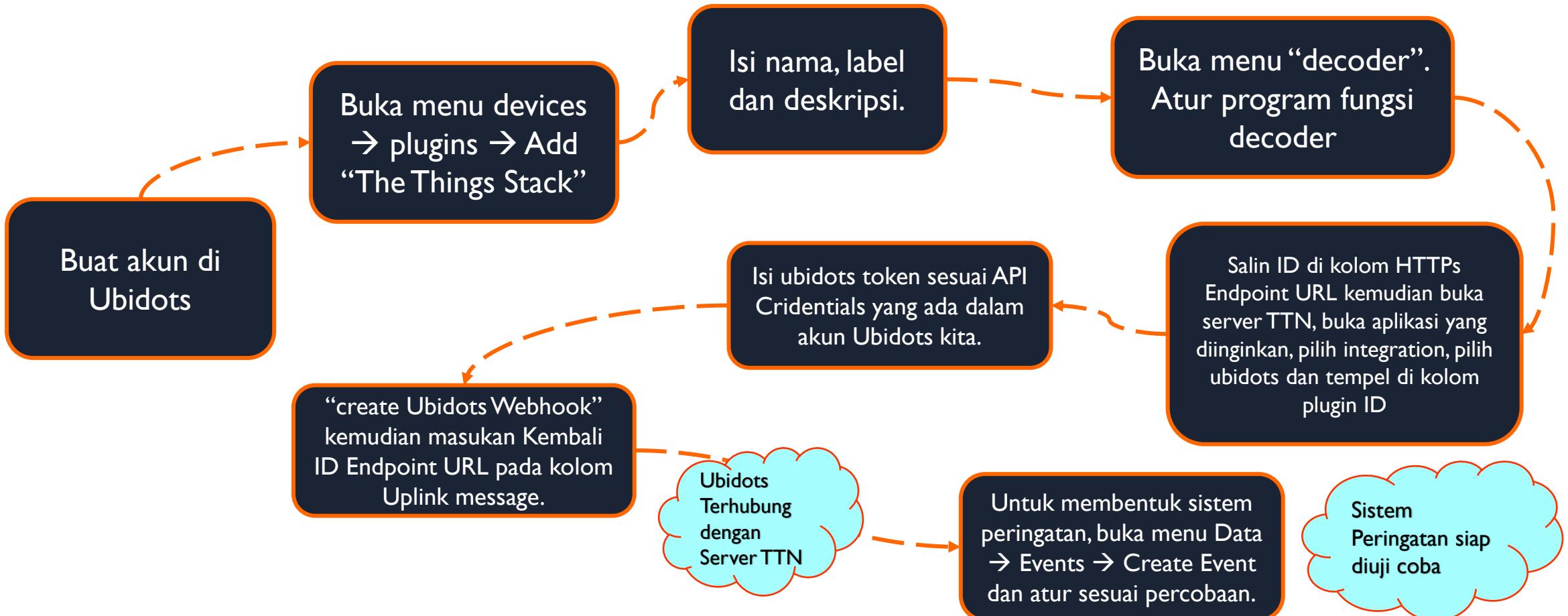


**THE THINGS
NETWORK**





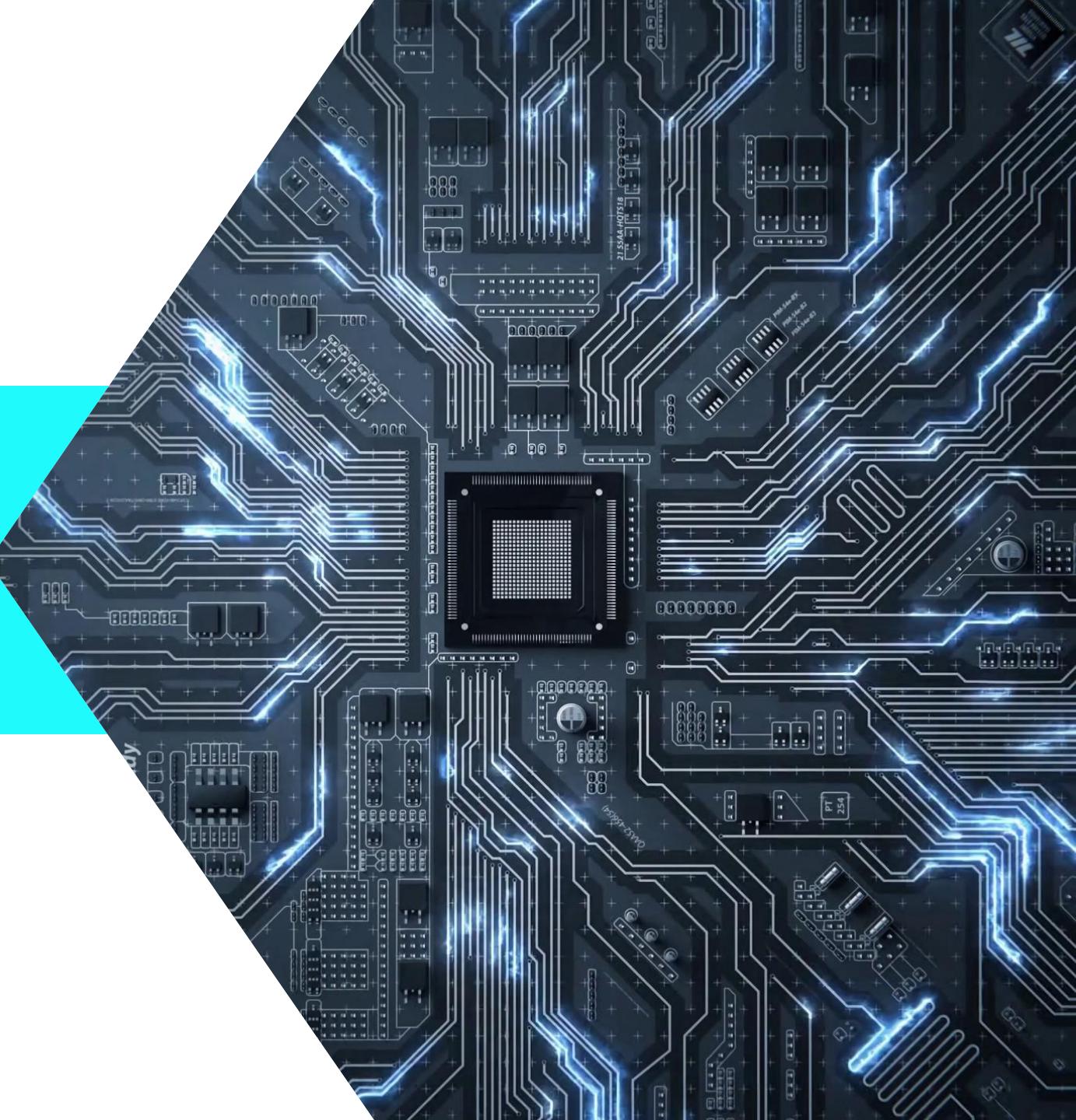
Pengaturan Webhook





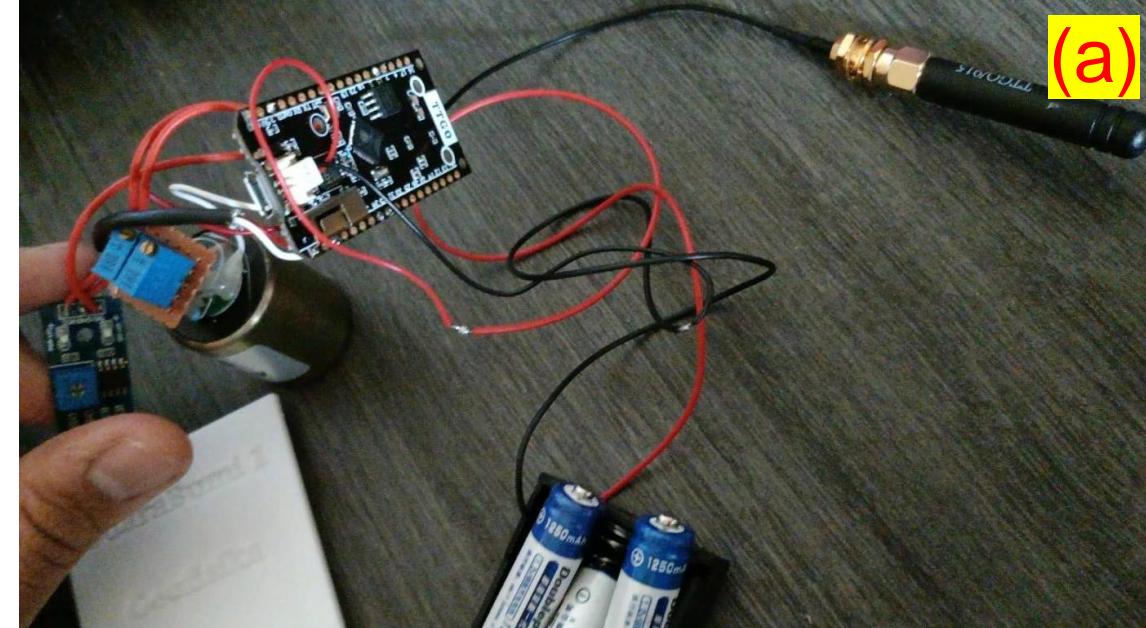
BAB IV

HASIL PENELITIAN

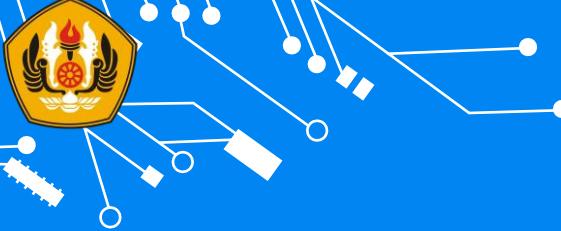




Rangkaian Fisik Instrumen



Penampilan *node sensor* dari komponen interiornya
(a) kemudian tampak luar (b) dan tampak
dalam *node sensor* (c)



Daya Tahan Alat

Tabel perhitungan penggunaan daya mikrokontroler dengan berbagai kondisi



Total daya baterai

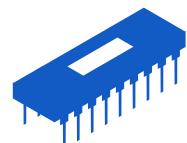
$$P = V \cdot I \cdot n$$

$$P = 1,2 V \cdot 1,25 Ah \cdot 3 = 4,5 WH$$



Total daya yang dibutuhkan
mikrokontroler

$$P = 0,26 A \cdot \frac{2}{60} \cdot 3,6 V = 0,0312 W$$



Total daya yang dibutuhkan SW-
420

$$P = 0,00015 A \cdot \frac{58}{60} \cdot 3,6 V = 0,000522 W$$

$$P = 0,015 A \cdot 3,3 V = 0,0495 W$$

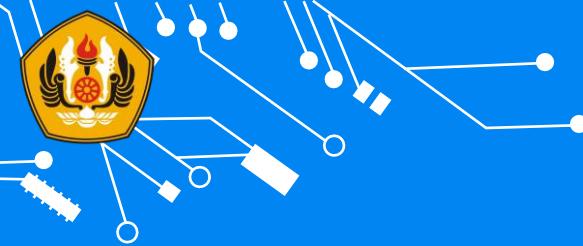


Daya tahan baterai terhadap total
daya yang dibutuhkan rangkaian

$$\text{Daya tahan} = \frac{P_{\text{baterai}}}{P_{\text{rangkaian}}}$$

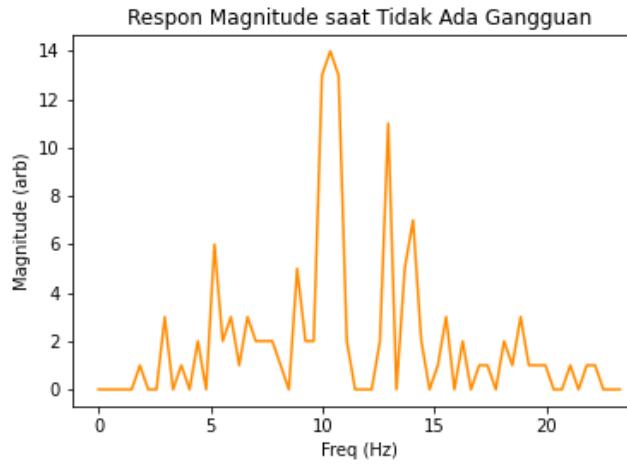
$$\text{Daya tahan} = \frac{4,5}{0,0312 + 0,000522 + 0,0495}$$

$$\text{Daya tahan} = 55.4 H = 2 hari 7.4 jam$$

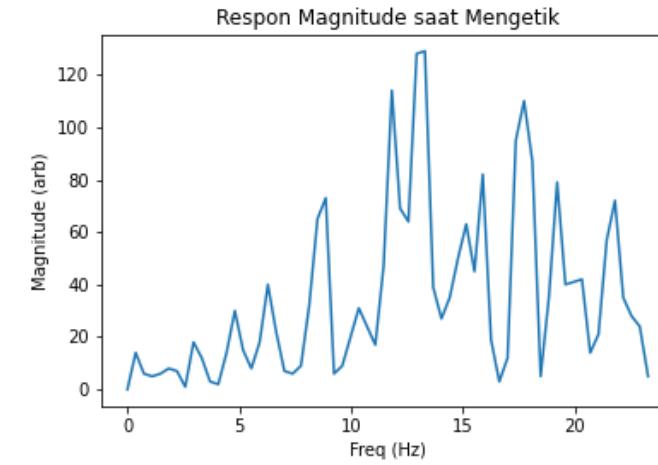


Grafik Uji Coba Perekaman Data

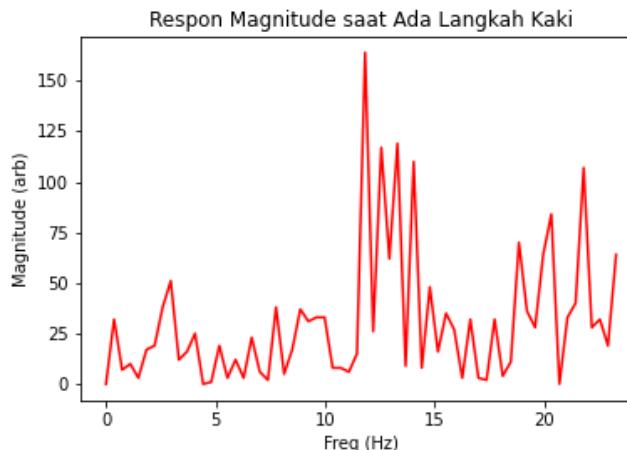
Grafik A



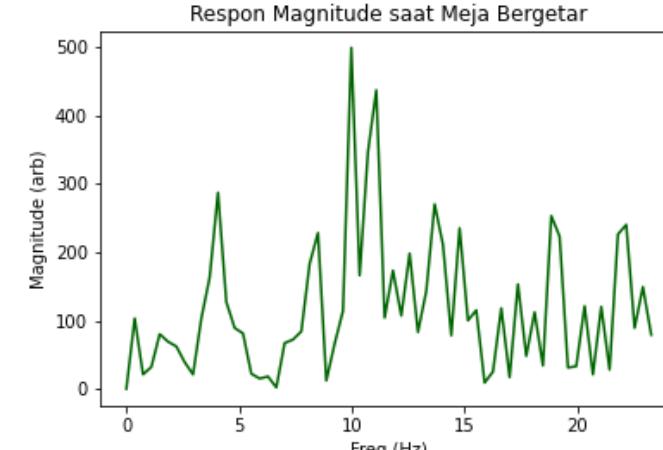
Grafik B



Grafik C



Grafik D



Grafik data hasil rekaman akuisisi saat keadaan tidak ada gangguan (a), saat ada kegiatan mengetik (b), saat ada kegiatan Langkah kaki (c), dan saat meja digetarkan (d)

Nilai Ambang batas = 200



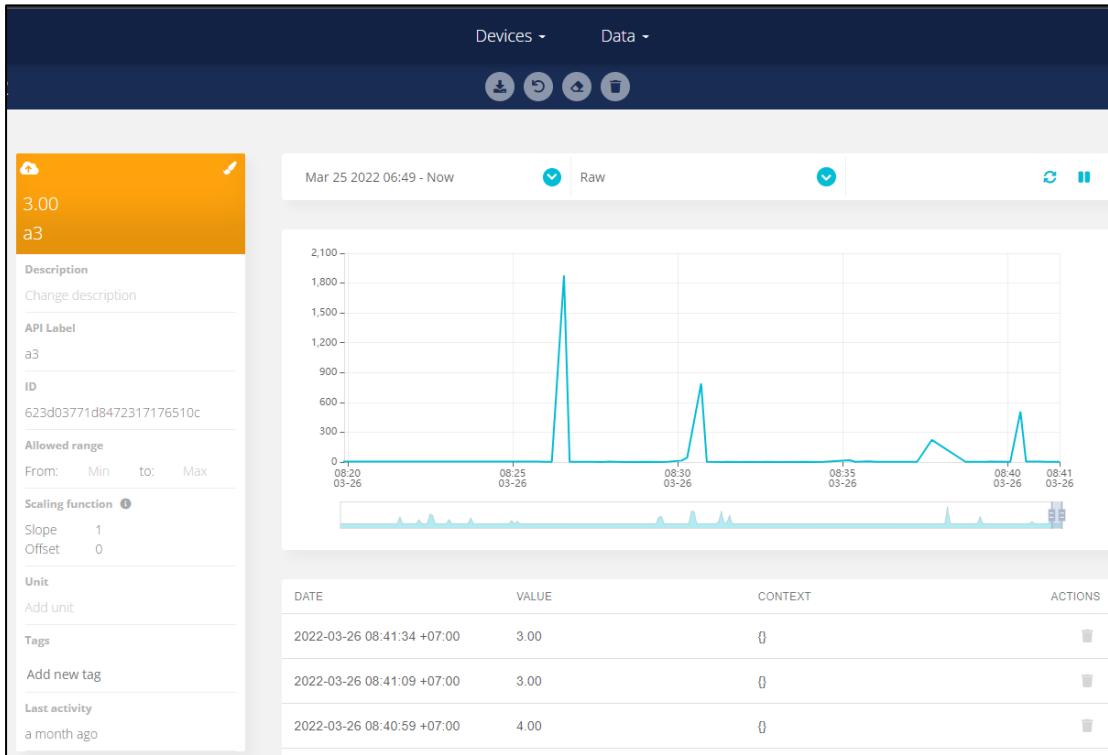
Tampilan Transmisi Data

Time	Type	Data preview	Verbose stream	Export as JSON	Pause
↓ 08:30:53	Schedule data downlink for transmission	Rx1 Delay: 1			
↑ 08:30:53	Forward uplink data message	Payload: { A1: 3, A2: 2, A3: 2, State: 0, f1: 13.06, f2: 1.96, f3: 14.38 } 2E 6C 05 55 9A 6C 03 00 ...	<input type="button" value="Hex"/> <input type="button" value="Raw"/> <input type="button" value="Copy"/>	FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -89	
↑ 08:30:53	Successfully processed data message	DevAddr: 26 0D 73 9E <input type="button" value="Hex"/> <input type="button" value="Raw"/> FCnt: 1 FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -89			
↓ 08:30:43	Schedule data downlink for transmission	Rx1 Delay: 1			
↑ 08:30:43	Forward uplink data message	Payload: { A1: 1110, A2: 908, A3: 790, State: 0, f1: 4.25, f2: 7.84, f3: 8.17 } 70 5D 05 65 3B 65 56 04 ...	<input type="button" value="Hex"/> <input type="button" value="Raw"/> <input type="button" value="Copy"/>	FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -96	
↑ 08:30:43	Successfully processed data message	DevAddr: 26 0D 73 9E <input type="button" value="Hex"/> <input type="button" value="Raw"/> FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -96			
↓ 08:30:18	Schedule data downlink for transmission	Rx1 Delay: 1			
↑ 08:30:17	Forward uplink data message	Payload: { A1: 103, A2: 54, A3: 46, State: 0, f1: 20.25, f2: 20.58, f3: 17.31 } 7B 6E 96 6E 8A 6D 67 00 ...	<input type="button" value="Hex"/> <input type="button" value="Raw"/> <input type="button" value="Copy"/>	FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -97	
↑ 08:30:17	Successfully processed data message	DevAddr: 26 0D 73 9E <input type="button" value="Hex"/> <input type="button" value="Raw"/> FCnt: 1 FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -97			
↓ 08:30:07	Schedule data downlink for transmission	Rx1 Delay: 1			
↑ 08:30:07	Forward uplink data message	Payload: { A1: 17, A2: 16, A3: 15, State: 0, f1: 18.3, f2: 6.86, f3: 13.72 } DB 6D 64 64 64 6C 11 00 ...	<input type="button" value="Hex"/> <input type="button" value="Raw"/> <input type="button" value="Copy"/>	FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -90	
↑ 08:30:07	Successfully processed data message	DevAddr: 26 0D 73 9E <input type="button" value="Hex"/> <input type="button" value="Raw"/> FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -90			
↓ 08:29:42	Schedule data downlink for transmission	Rx1 Delay: 1			
↑ 08:29:42	Forward uplink data message	Payload: { A1: 7, A2: 3, A3: 3, State: 0, f1: 5.23, f2: 9.15, f3: 15.69 } B1 5E DB 65 05 6D 07 00 ...	<input type="button" value="Hex"/> <input type="button" value="Raw"/> <input type="button" value="Copy"/>	FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -86	
↑ 08:29:42	Successfully processed data message	DevAddr: 26 0D 73 9E <input type="button" value="Hex"/> <input type="button" value="Raw"/> FCnt: 1 FPort: 1 Data rate: SF7BW125 SNR: 7.8 RSSI: -86			
↓ 08:29:32	Schedule data downlink for transmission	Rx1 Delay: 1			

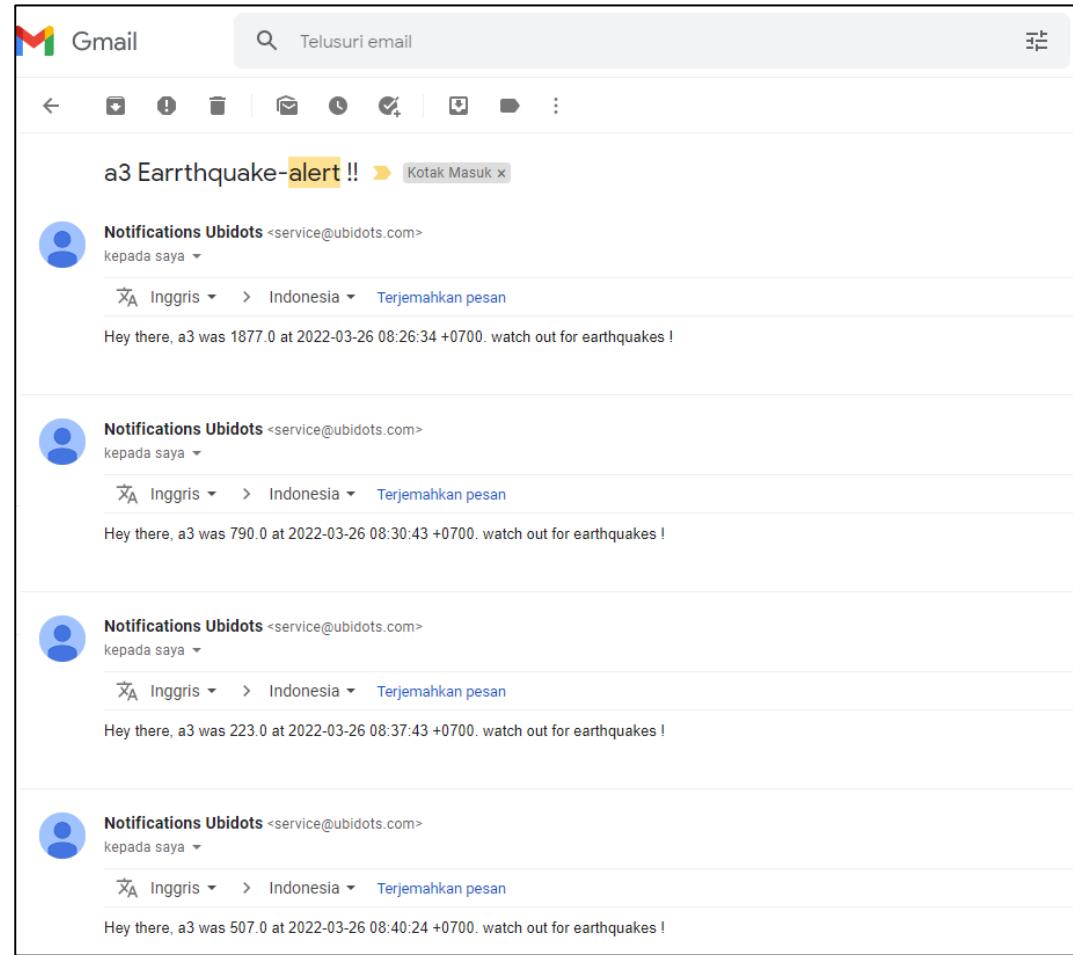
Tampilan data yang diterima oleh Server TTN : The Things Stack V3.



Tampilan Peringatan



Tampilan grafik data terhadap waktu yang diterima oleh webhook Ubidots dari server TTN.



The figure shows a screenshot of a Gmail inbox. The subject of the first email is 'a3 Earrhquake-alert !!'. The message is from 'Notifications Ubidots <service@ubidots.com>' to the recipient. It includes language translation options for English and Indonesian, followed by the message: 'Hey there, a3 was 1877.0 at 2022-03-26 08:26:34 +0700. watch out for earthquakes !'. The subsequent three emails follow the same pattern, each reporting a different seismic event with values 790.0, 223.0, and 507.0 respectively, all occurring at approximately 08:30 on March 26, 2022.

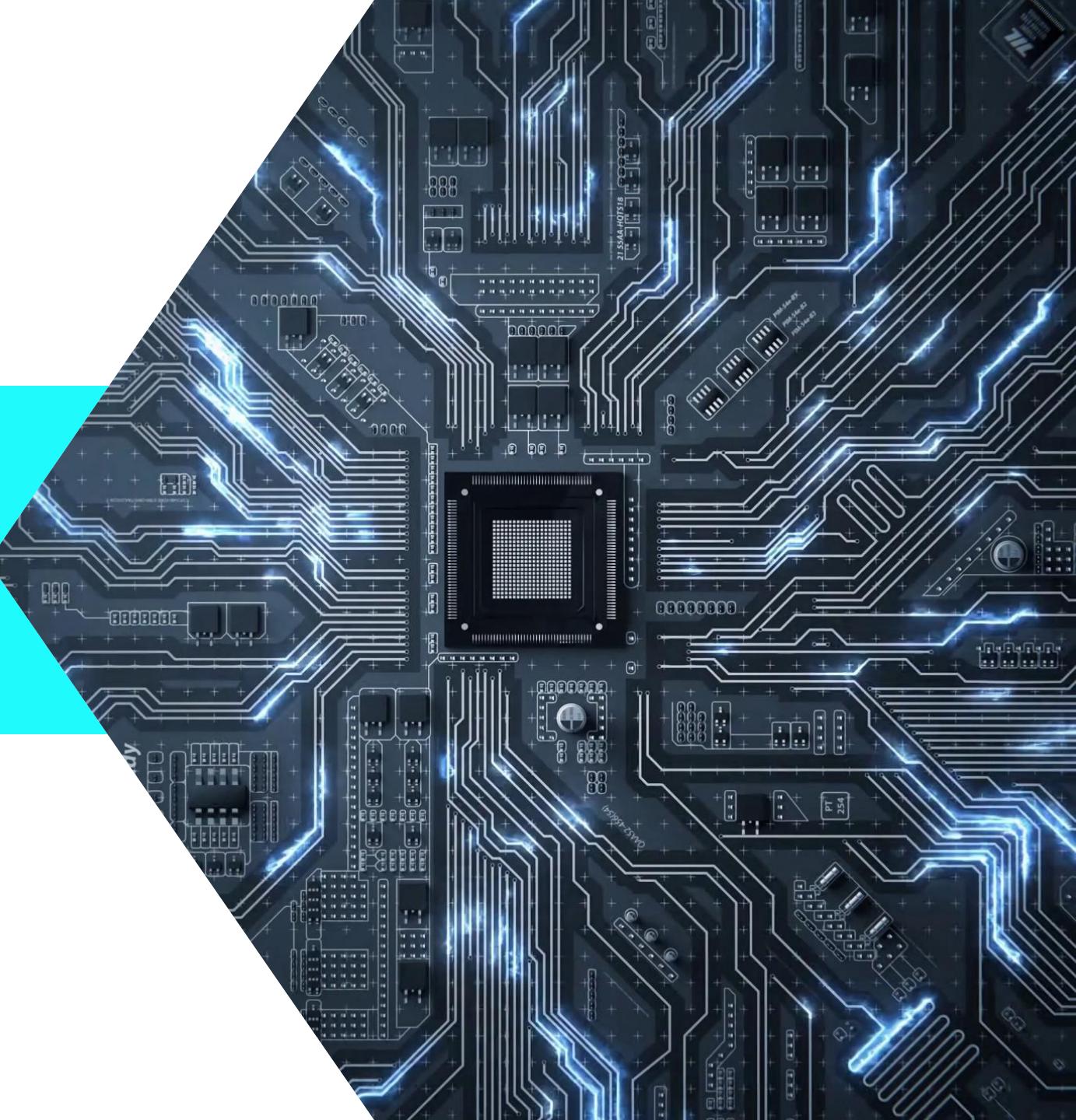
Tampilan email yang masuk dari webhook Ubidots apabila event telah terpicu

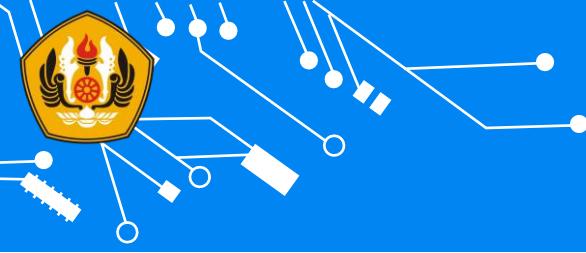


BAB V

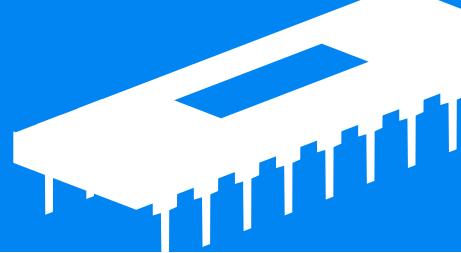
SIMPULAN

DAN SARAN

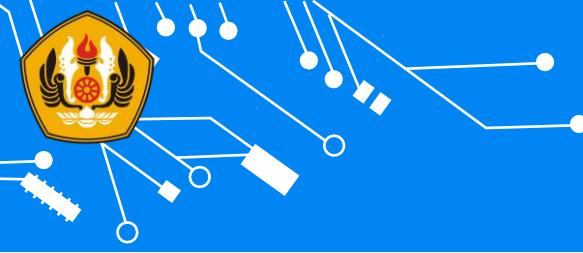




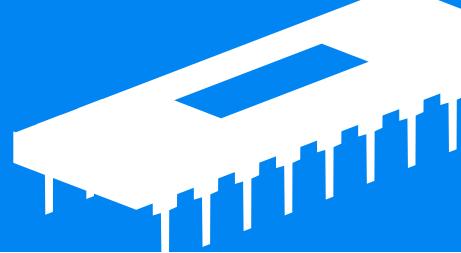
KESIMPULAN



- Nilai ambang batas yang optimal untuk peringatan dini bencana rawan gerakan tanah adalah 200.
- Sistem peringatan dini yang aktual dan nirkabel diwujudkan dengan transmisi data dari *node sensor* MCU TTGO LoRa32 ke *gateway* Dragino via jaringan LoRaWAN. Kemudian *gateway* juga diintegrasikan ke server menggunakan internet. Dengan demikian, selama alat bekerja dan mengirimkan data, kita bisa memantau dari mana saja.
- Antenna jenis yagi memiliki cangkupan transmisi yang cukup jauh dibanding antenna jenis omni, terlihat dari nilai RSSI antenna yagi yang lebih positif dibanding antenna jenis omni pada jarak yang sama.



SARAN



- Menggunakan webhook yang lebih lengkap agar peringatan dapat lebih mudah terjangkau seperti lewat pesan atau lewat media sosial dan lebih menarik (dengan infografis atau grafik).
- Menggunakan mikrokontroler yang satu paket dengan gateway agar lebih mudah mengkonfigurasikannya.
- *Node sensor* menggunakan sumber tenaga yang lebih berkelanjutan seperti memadukannya dengan panel surya untuk meminimalisir frekuensi pemeliharaan ke lapangan.
- *Node sensor* menggunakan komponen elektronik yang menggunakan daya rendah untuk memperpanjang masa hidup alat.
- Untuk transmisi data sebaiknya menggunakan antenna yagi dibanding antenna omni bawaan dari mikrokontroler agar lebih jauh jangkauan transmisinya.



THANK YOU

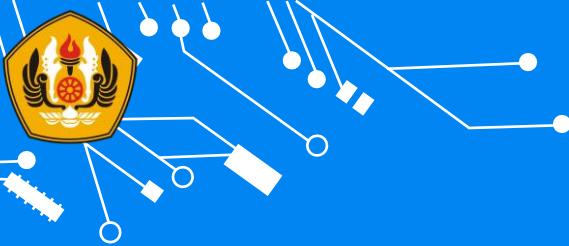


PRESENTASI SEMINAR HASIL

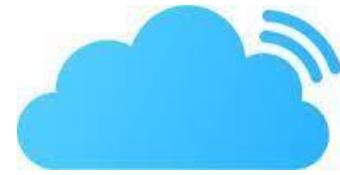
**RANCANG BANGUN EARLY-WARNING SYSTEM
UNTUK DAERAH RAWAN BENCANA GERAKAN
TANAH**

Avif Maulana Azis
140710180048

Dosen Pembimbing utama
Prof. Dr. rer. nat. Yudi Rosandi M. Si.



Pengaturan Server



**THE THINGS
NETWORK**

Buat akun di server The Things Network

Mengoneksikan Gateway dengan Server

- Pilih region server
- Buka menu “console”
- Pilih “go to gateways”
- “add gateways”
- Input nama gateway di kolom gateway id, ssid gateway di kolom gateway EUI, pilih server sesuai yang diinginkan (Australia), kemudian tentukan frekuensi yang akan digunakan (sama dengan gateway).
- “create gateway”

Membuat Aplikasi

- Pilih region server
- Buka menu “console”
- Pilih “go to Applications”
- “add application”
- Input nama, id dan deskripsi aplikasi.
- “register application”

Menambahkan end device pada aplikasi

- Buka aplikasi yang diinginkan.
- Pilih “add end device” dan pilih “manually”.
- Input frequency plan sesuai gateway, lorawan version sesuai gateway (1.0.3), regional parameter version default, kemudian generate DEV & APP EUI serta APPKEY dan nama end device.
- Buka “show advance activation”, pilih activation method menjadi ABP.
- “register end device”

monitoring