

## BAB 3

### METODE PENELITIAN

Metode yang digunakan pada penelitian ini yaitu dengan metode *black box*. Metode *black box* merupakan metode pengujian fungsionalitas sistem tanpa harus memperhatikan logika dan proses yang dilakukan perangkat, sehingga pengujian ini hanya difokuskan pada nilai masukan dan keluaran dari sistem. Pengujian yang dimaksud untuk mengetahui apakah fungsi-fungsi keluaran dari alat sudah berjalan sesuai yang dirancang peneliti atau masih ada kesalahan dalam perancangan. Jika sistem gagal ataupun ada kesalahan dalam keluaran data yang dikeluarkan maka sistem dinyatakan butuh perbaikan [32].

#### 3.1 Alat dan Bahan

Alat dan bahan yang digunakan pada penelitian ini terbagi menjadi dua bagian yaitu perangkat *Hardware* dan perangkat *Software*. Alat dan bahan yang digunakan pada perancangan penelitian dapat dilihat pada Tabel 3.1 Perangkat *Hardware* dan Tabel 3.2 Perangkat *Software*.

**Tabel 3.1 Perangkat *Hardware*.**

No	Alat dan Bahan	Jumlah
1	Laptop	1
2	Arduino Uno R3 Atmega 328 SMD	3
3	LoRa <i>Shield</i> RFM95W	3
4	<i>Mikrokontroler</i> Esp 32	1
5	GPS Unlbox neo-7M	3
6	Sensor <i>Sharp</i> IR	3
7	Sensor <i>RTC</i>	3
8	<i>Breadboard</i>	1
9	Kabel <i>Jumper</i>	~

**Tabel 3.2 Perangkat *Software*.**

No	Nama <i>Software</i>
1	Arduino IDE
2	<i>ThingSpeak</i>

Dalam memahami penelitian ini, berikut ini akan menjelaskan spesifikasi dari alat dan bahan dalam Tabel 3.1 yang digunakan dalam penelitian ini.

### 1. Laptop

Pada penelitian ini laptop sangat penting dibutuhkan untuk merangkai, melakukan pemrograman dan mengolah data selama proses penelitian. Minimum spesifikasi laptop yang digunakan RAM 4 GB, ROM penyimpanan *Solid-state drive* (SSD) dengan prosesor *intel i3/ryzen 3* agar dapat memprogram dan menjalankan simulasi merangkai alat. Pada laptop juga sudah dilengkapi dengan *software* Arduino IDE dan *ThingSpeak*.

### 2. Arduino Uno R3 Atmega 328 SMD

Arduino Uno berfungsi sebagai mikroprosesor untuk mengatur program dan mengendalikan berbagai komponen elektronika pada penelitian ini. Pada penelitian Arduino Uno yang digunakan yaitu Arduino Uno R3 Atmega 328 SMD, untuk spesifikasi Arduino Uno R3 Atmega 328 SMD dapat dilihat pada Tabel 3.3 di bawah.

**Tabel 3.3 Spesifikasi Arduino R3 Atmega 328 SMD.**

Arduino Uno R3 Atmega 328 SMD	
Mikrokontroler	ATmega328P
Tegangan Operasi	5V
Tegangan <i>Input</i>	7-12V
Tegangan <i>Input</i>	6-20V (batas)
Pin I/O Digital	14 (6 di antaranya <i>output</i> PWM)
Pin I/O Digital	PWM 6
Pin <i>Input</i> Analog	6
Arus DC per Pin I/O	20 mA
Arus DC	3.3V Pin 50 mA
<i>Flash</i> Memory	32 KB (ATmega328P) (0.5 KB bootloader)
SRAM	2KB (ATmega328P)
EEPROM	1KB (ATmega328P)
Kecepatan/Jam	16Hz

### 3. LoRa *Shield* RFM95W

Perangkat RFM95W adalah satu modul LoRa yang berkomunikasi menggunakan modul radio dengan frekuensi tertentu yang mampu menjangkau jarak komunikasi sejauh  $\pm 5$  kilometer. Modul RFM95W mengadopsi teknologi

*spread spectrum* LoRa. Memiliki kemampuan anti *blocking* dan anti *jamming* yang kuat serta hemat energi. Modul RFM95W merupakan jaringan *wireless transceiver* yang bekerja pada frekuensi 915 MHz. Spesifikasi LoRa *Shield* RFM95 sebagai mana dapat dilihat pada Tabel 3.4 berikut :

**Tabel 3.4 Spesifikasi LoRa *Shield* RFM95W.**

LoRa <i>Shield</i> RFM95W	
Modem	LoRaTM
Anggaran tautan maksimum	168 dB.
<i>Output</i> RF konstan	100 mW vs. catu V(+20 dBm)
PA efisiensi tinggi	+14 dBm.
Laju bit yang dapat diprogram	Hingga 300 kbps.
Sensitivitas tinggi	Hingga -148 dBm.
Ujung depan anti peluru	IIP3 = -12,5 dBm.
Imunitas pemblokiran	sangat baik.
Arus RX rendah	10,3 mA
Retensi register	200 nA
<i>Synthesizer</i> terintegrasi	resolusi 61 Hz.
Modulasi	FSK, GFSK, MSK, GMSK, LoRaTM dan OOK.
Sinkronisasi untuk pemulihan jam	bit bawaan
RSSI Rentang Dinamis	127 dB.
Sensor RF	Otomatis
Sensor CAD	AFC ultra cepat.
Mesin paket	256 byte dengan CRC.
Indikator baterai lemah.	lampu
Ukuran Modus	16*16mm

#### 4. Mikrokontroler Esp-32

Esp-32 merupakan sebuah mikrokontroler yang memiliki kelebihan untuk menghubungkan sebuah projek ke jaringan *internet* maupun menggunakan *Bluetooth*. Pada penelitian ini Esp-32 berfungsi untuk menghubungkan Rx ke jaringan *internet* sehingga data yang diterima pada Rx dapat ditampilkan pada *web dashboard ThingSpeak*. Spesifikasi Mikrokontroler Esp-32 yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.5.

**Tabel 3.5 Spesifikasi Mikrokontroler Esp-32.**

Mikrokontroler Esp-32	
Tegangan	3.3 Volt
<i>Processor</i>	<i>Testilica L108 32 bit</i>
Kecepatan <i>Processor</i>	<i>Dual 160 MHz</i>
RAM	520K
GPIO	34
ADC	7
Dukungan 802.11	(11b/g/n/e/)
<i>Bluetooth</i>	(BLE ( <i>Bluetooth Low Energy</i> ))
SPI	3
I2C	2
UART	3

#### 5. GPS *Unlbox neo-7M*

GPS digunakan untuk mengetahui letak *longitude* dan *latitude* dari suatu alat yang menggunakan GPS. Penggunaan GPS pada penelitian ini bertujuan untuk mengetahui titik lokasi objek yang diteliti. GPS yang digunakan pada penelitian ini yaitu GPS *Unlbox neo-7M*. Untuk spesifikasi GPS ditunjukkan pada Tabel 3.6 berikut.

**Tabel 3.6 Spesifikasi GPS *Unlbox neo-7M*.**

GPS <i>Unlbox neo-7M</i>	
Penerima GPS	Mandiri
9600 baud	pengaturan <i>default</i> ; dapat diubah
VCC	3,3V - 5V
LED <i>Onboard</i>	Berkedip (dapat Sinyal)
Modul GPS	<i>U-blox NEO-7M</i>
GPS Dalam Ruangan	sensitivitas pelacakan -162 dBm
Teknologi	<i>anti-jamming</i>
Mendukung	SBAS (WAAS, EGNOS, MSAS, GAGAN)
Mesin pemosisian	memiliki 50 saluran & 2 juta korelator efektif
Pembaruan posisi	5Hz
Suhu pengoperasian	-40 SAMPAI 85C
Soket UART	TTL
Penyimpanan pengaturan	EEProm

#### 6. Sensor *Sharp* IR (9GP2Y0A21YK0F)

Sensor *Sharp* IR merupakan sensor untuk mengukur jarak suatu benda. Sensor jarak digunakan pada penelitian bertujuan untuk mengetahui ketinggian permukaan air laut terhadap bagan ikan tancap agar dapat memastikan air laut sedang pasang atau surut. Dalam pengukuran ini peneliti menggunakan sensor *Sharp* IR GP2Y0A21YK0F yang sebagaimana dapat dilihat pada gambar 2.9 pada sub bab 2, dan untuk spesifikasi sensor GP2Y0A21YK0F yang digunakan dapat dilihat pada Tabel 3.7;

**Tabel 3.7 Spesifikasi Sensor IR GP2Y0A21YK0F.**

Sensor <i>Sharp</i> IR GP2Y0A21YK0F	
Ukuran:	1,75" × 0,75" × 0,53"
Berat:	3,5 gram
Kisaran maksimum:	80 cm
Kisaran minimal:	10 cm
Tingkat pengambilan sampel:	26Hz <sup>1</sup>
Tegangan operasi minimum:	4,5V
Tegangan operasi maksimum:	5,5V
Arus suplai:	30mA2
Jenis keluaran:	tegangan analog
Diferensial tegangan keluaran:	1,9V

#### 7. Sensor RTC

Sensor RTC digunakan untuk memberikan waktu yang akurat dalam pengiriman data dan penerimaan data dalam *device*. Sensor RTC juga mempunyai kemampuan untuk menjaga ketepatan waktu di seluruh *device*. Tujuan dibuatnya RTC agar mengetahui dan mempermudah dalam melakukan analisis data untuk menentukan Nilai ToA dan nilai *delay* pada pengiriman dan penerimaan data dalam komunikasi LoRa pada penelitian ini.

#### 8. *Breadboard*

*Breadboard* digunakan sebagai papan proyek untuk perakitan komponen elektronik. *Breadboard* digunakan agar uji coba dalam perakitan alat dapat dilakukan dengan mudah yaitu dengan menancapkan komponen ke *breadboard* tanpa melalui tahap penyolderan, sehingga meminimalisir kerusakan komponen pada saat uji coba untuk perakitan.

### 9. Kabel *Jumper*

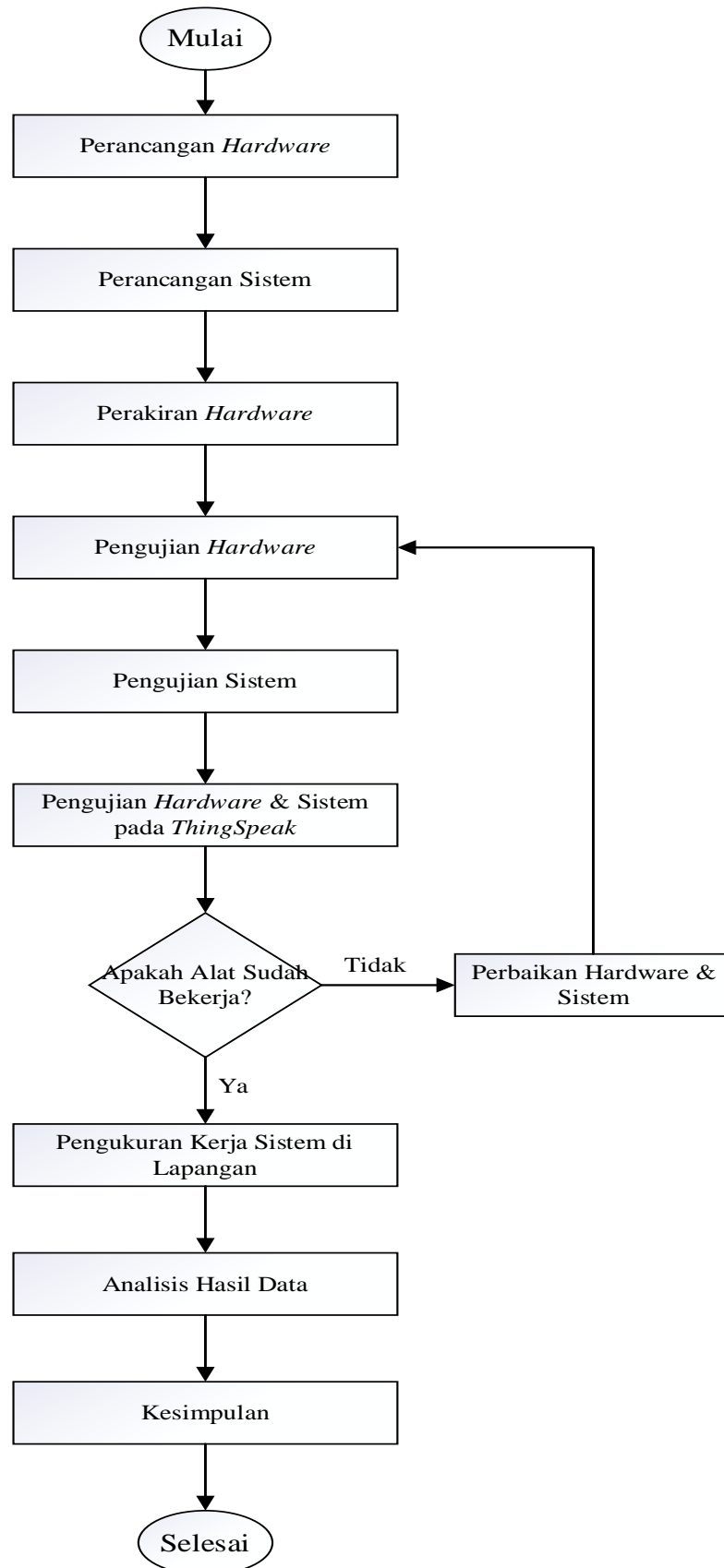
Kabel *jumper* merupakan kabel yang dipergunakan sebagai penghubung satu komponen dengan komponen lain pada saat perakitan alat, ataupun untuk menghubungkan jalur rangkaian yang terputus pada *breadboard* pada saat melakukan percobaan.

## 3.2 Alur Penelitian

Alur penelitian merupakan serangkaian tindakan yang sistematis dan terorganisir yang membantu peneliti untuk merencanakan, melaksanakan, dan menganalisis penelitian yang diteliti dari awal penelitian hingga akhir penelitian untuk mencapai tujuan akhir yang diharapkan.

Alur penelitian dalam penelitian ini dilakukan dalam beberapa tahapan yaitu tahapan perancangan *hardware*, perancangan sistem, perakitan *hardware*, pengujian *hardware*, pengujian sistem, pengujian *hardware* dan sistem pada *ThingSpeak* dan tahap analisis. Alur proses dalam pengerjaan penelitian pengembangan sistem pemantauan bagan ikan dengan menggunakan teknologi LoRa pada penelitian ini dapat dilihat pada Gambar 3.1 *Flowchart* alur penelitian.

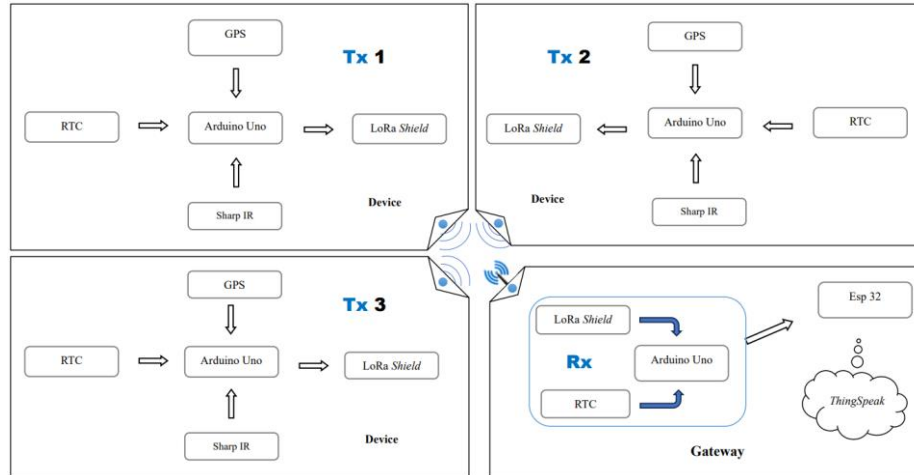
Langkah awal yang dilakukan pada penelitian yaitu perancangan *hardware*. Terdapat dua *hardware* yang dirancang dalam penelitian ini. *Hardware* yang dirancang yaitu *hardware device transmitter* dan *hardware device receiver*. Kemudian dilanjutkan dengan merancang sistem yang akan dijalankan dalam komunikasi perangkat keras tersebut. Setelah selesai merancang *hardware* dan sistem, langkah berikutnya merupakan perakitan *hardware* yang diikuti dengan pengujian *hardware* dan pengujian sistem pada penelitian. Jika hasil pengujian *hardware* dan pengujian sistem telah berhasil, maka langkah selanjutnya ialah menguji *hardware* dan sistem terhadap *ThingSpeak*. Pengujian *hardware* dan pengujian sistem pada *ThingSpeak* bertujuan untuk memvalidasi bahwa proses komunikasi telah berjalan dengan baik. Jika pemrosesan data tidak berjalan dengan baik, akan dilakukan perbaikan pada *hardware* dan sistem, dan kemudian kembali ke tahap pengujian. Apabila pengujian berhasil, penelitian akan melanjutkan dengan pengambilan data di lapangan. Dari hasil data di lapangan maka dilanjutkan dengan analisis data serta mengambil kesimpulan dari penelitian yang dilakukan.



**Gambar 3.1** *Flowchart* Alur Penelitian.

### 3.2.1 Perancangan *Hardware*

Dalam sebuah penelitian dibutuhkan sebuah perancangan untuk memahami gambaran bentuk dari penelitian yang dilakukan. Perancangan *hardware* pada penelitian ini dapat dilihat pada Gambar 3.2 Perancangan *Hardware* Keseluruhan Sistem.



**Gambar 3.2 Perancangan *Hardware* dan sistem Keseluruhan.**

Perancangan *hardware* terdiri dari dua bagian perencanaan yaitu perancangan *device transmitter* (Tx) dan perancangan *device receiver* (Rx) atau disebut *gateway*. Penelitian ini membutuhkan tiga *device* Tx dan satu *device* Rx yang akan diletakkan pada tiga bagan ikan yang berbeda. Data yang didapat dari setiap *device* Tx akan dikirimkan ke *device* Rx yang ada pada *gateway* kemudian data diteruskan ke jaringan *internet* melalui esp-32 dan data akan masuk dan ditampilkan di *ThingSpeak*.

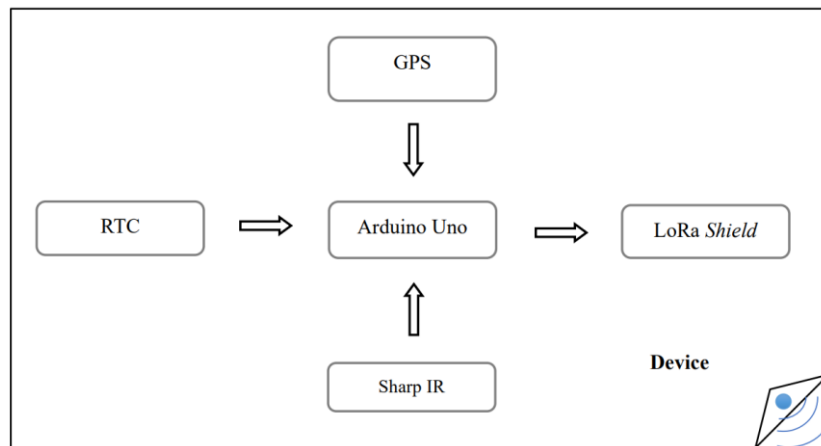
#### 3.2.1.1 Perancangan pada *Device Transmitter* (Tx)

*Device* Tx adalah perangkat pengirim yang dilengkapi dengan berbagai komponen keras, yang dirancang khusus untuk berinteraksi dengan jaringan, sehingga memungkinkan transmisi data secara efisien. Perancangan *hardware device* Tx dapat dilihat pada Gambar 3.3 Perancangan *Hardware Device Transmitter*. Dari gambar *device* Tx terdapat beberapa sensor yang digunakan yaitu sensor *Sharp IR* digunakan untuk mengukur ketinggian permukaan air laut dan kemudian sensor GPS yang digunakan sebagai alat mengetahui letak lokasi *latitude* dan *longitude* dari *device* Tx. Arduino Uno



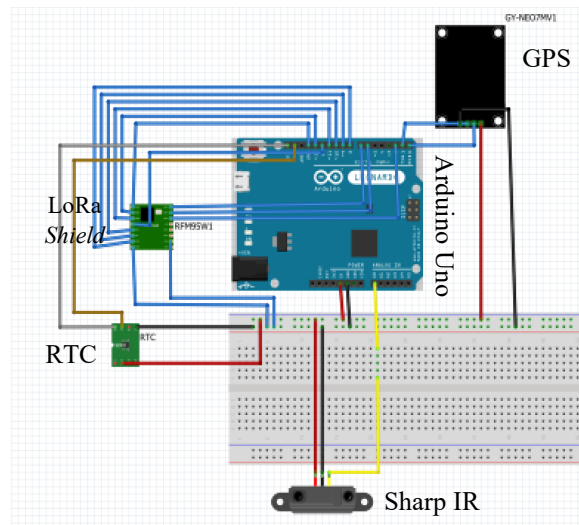
pada *device* Tx merupakan sebagai mikrokontroler atau sebagai pemroses nilai sensor yang dihubungkan pada Arduino dan LoRa *Shield*.

Untuk menjalankan sensor dan memproses data dari sensor pada Arduino, dilakukan pemrograman menggunakan *software* Arduino IDE agar dapat membaca sensor yang terhubung pada *device* Tx. Data dari Arduino akan dikirim menuju LoRa *Shield*. LoRa *Shield* adalah protokol komunikasi yang menghubungkan *transmitter* ke *receiver* (*gateway*), dimana pada *gateway* juga terdapat LoRa *Shield* sebagai jalur komunikasi dengan *device* Tx.



**Gambar 3.3 Perancangan *Hardware Device Transmitter*.**

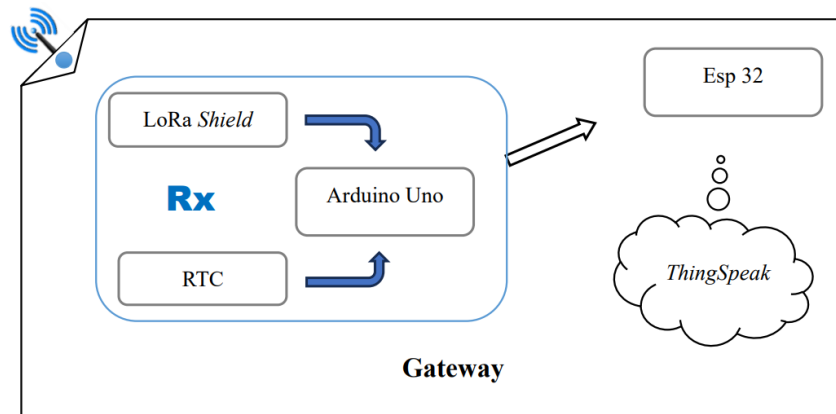
Bentuk rangkaian perancangan *device* Tx dapat dilihat pada Gambar 3.4. Dari gambar terlihat bahwa Arduino Uno merupakan komponen yang sangat penting karena berfungsi sebagai mikrokontroler untuk seluruh sensor dan tempat di mana LoRa *Shield* berada sebagai jalur komunikasi.



**Gambar 3.4 Rangkaian *Device Transmitter*.**

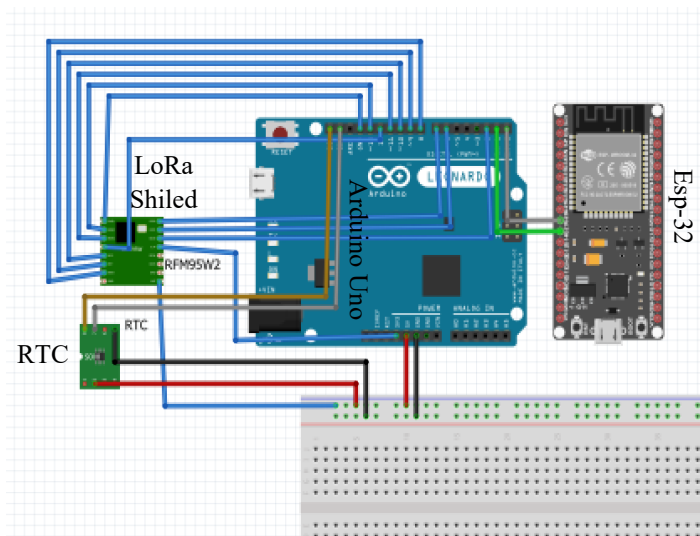
### 3.2.1.2 Perancangan pada *Device Receiver (Rx)*

*Device Rx* adalah perangkat fisik yang terletak pada *receiver* atau *gateway* dan berfungsi menerima data dari *device Tx*. Data yang diterima oleh Rx kemudian dikirimkan ke *web dashboard ThingSpeak*. Perancangan *gateway* yang dirancang yaitu menggunakan esp-32. Esp-32 dihubungkan ke *gateway* yang dimana esp-32 berfungsi untuk menghubungkan *gateway* ke *internet*, sehingga data yang diterima oleh *gateway* dapat diteruskan dan dikirimkan ke *web dashboard ThingSpeak* dan ditampilkan di *ThingSpeak*. Perancangan *hardware gateway* dapat dilihat pada Gambar 3.5 Perancangan *Hardware gateway*.



**Gambar 3.5 Perancangan *Hardware Gateway*.**

Dari perancangan *hardware gateway* hasil rangkaian *device gateway* yang ditunjukkan pada Gambar 3.6 Rangkaian *Device Gateway*.

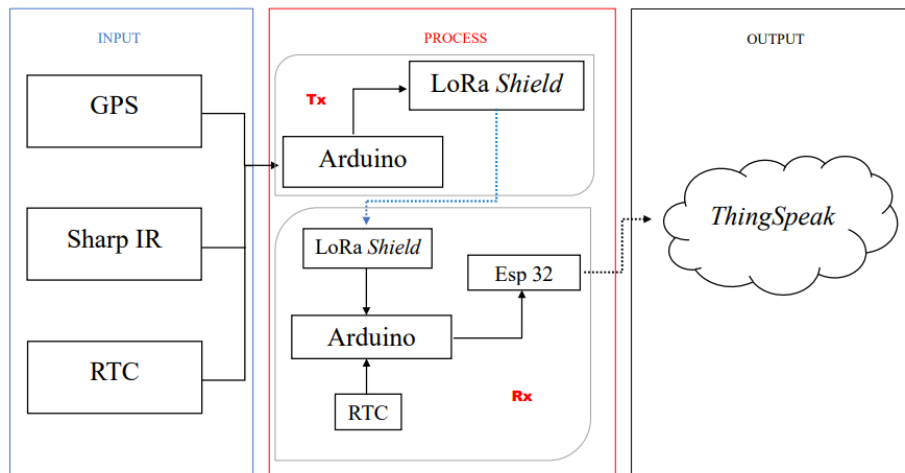


**Gambar 3.6 Rangkaian *Device Gateway*.**

### 3.2.2 Perancangan Sistem

Dalam perancangan sistem hal dasar yang perlu diketahui adalah bagaimana menggambarkan proses kerja keseluruhan *hardware*. Proses kerja dari keseluruhan *hardware* dapat dilihat melalui diagram blok pada Gambar 3.7 Diagram Blok Perancangan Sistem. Gambar 3.7 menjelaskan bagaimana proses kerja dari sistem yang digunakan dengan secara singkat atau hanya dengan garis besarnya saja. Diagram blok perancangan sistem yang dirancang terdiri dari 3 program kerja yaitu *input*, *process*, dan *output*.

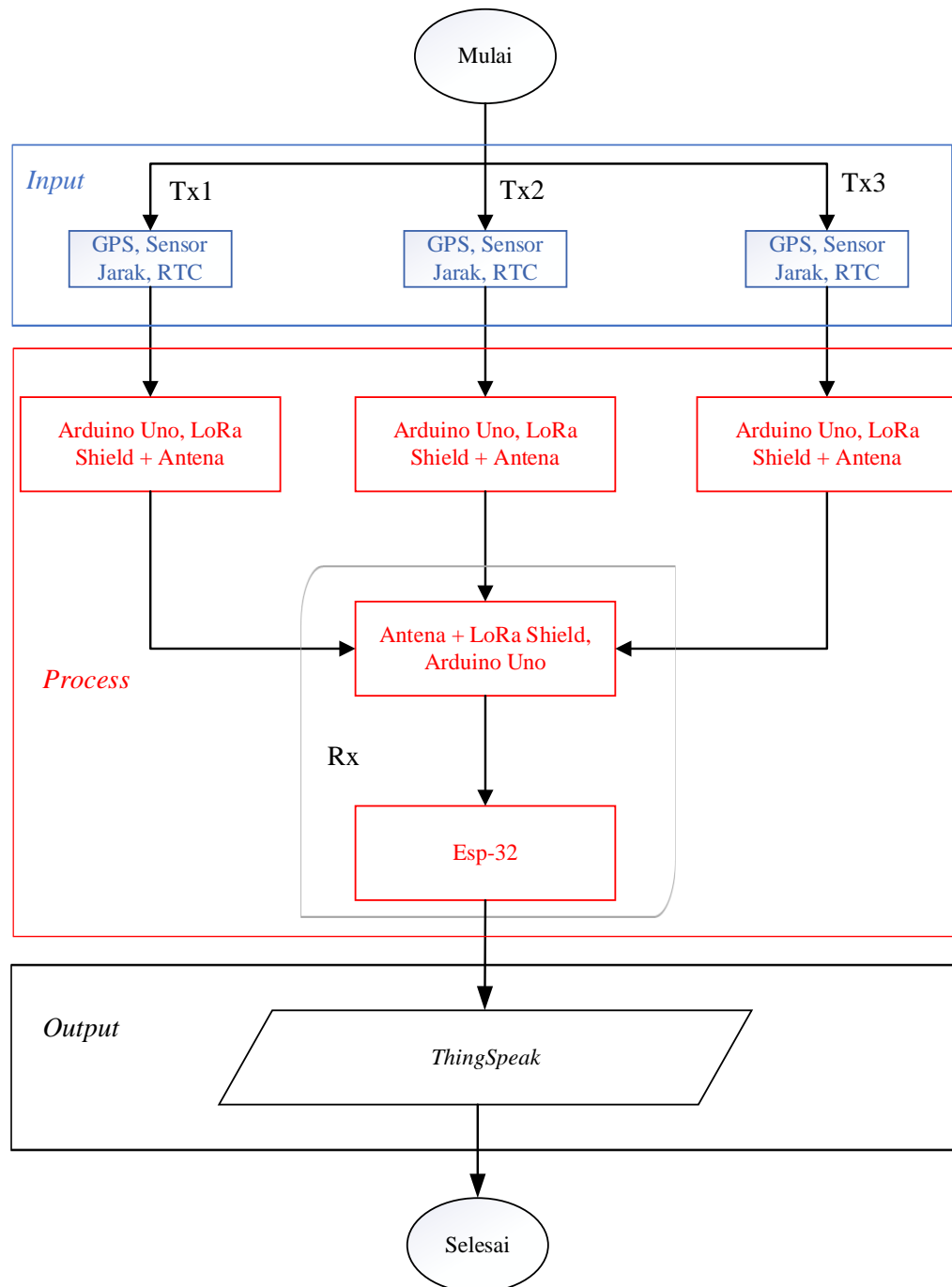
*Input* pada perancangan meliputi sensor GPS, dan sensor *Sharp IR* sebagai data. Kemudian yang kedua yaitu *process*. *Process* terdiri dari Arduino Uno, *LoRa Shield* yang terdapat pada *device Tx* dan *device Rx*. *LoRa Shield* merupakan penghubung antara Tx dan Rx menggunakan jaringan *wireless*, data yang masuk pada Rx melalui *LoRa Shield* diproses pada Arduino uno dan dikirimkan ke esp-32 dan dari esp-32 dikirimkan ke *output*. *Output* merupakan hasil akhir atau keluaran data yang ditampilkan. Dalam penelitian ini *output* yang digunakan yaitu *ThingSpeak*, data diterima *ThingSpeak* yaitu data yang dikirimkan oleh esp-32 melalui jaringan *internet*.



**Gambar 3.7 Diagram Blok Perancangan Sistem.**

Untuk menjalankan *hardware* dan sistem dapat berjalan dengan semestinya maka dilakukan pemrograman menggunakan *software* Arduino IDE. Arduino *Integrated Development Environment* (IDE) merupakan perangkat lunak yang digunakan untuk memprogram mikrokontroler Arduino maupun esp-32. Pemrograman ini bertujuan untuk mengendalikan sensor dan berbagai perangkat

elektronik lainnya yang terhubung ke papan Arduino maupun esp-32. Bahasa program yang digunakan pada *software* Arduino IDE yaitu menggunakan bahasa C dan C++ sehingga mudah untuk dipahami baik bagi pemula dalam bidang pemrograman. Melalui cara kerja diagram *blok*, maka *flowchart* dari sistem komunikasi penelitian ini dapat dilihat pada Gambar 3.8 *Flowchart* Perancangan Sistem.



**Gambar 3.8** *Flowchart* Perancangan Sistem.

Sejalan dengan pengembangan komunikasi dan perancangan sistem seperti pada *flowchart*, tahapan awal yang dilakukan merupakan percobaan komunikasi *point to point*. Percobaan komunikasi *point to point* dilakukan bertujuan sebagai landasan awal dalam pengembangan komunikasi *multipoint to point*. Dari gambar *flowchart* terdapat tiga proses kerja yang dikirimkan melalui Tx ke Rx.

Pada *input* perancangan sistem terdapat tiga *device*, dimana setiap *device* terdiri dari 3 *inputan* data yang berasal dari sensor GPS, sensor *Sharp* IR dan RTC. Data yang di *input* dari sensor GPS, *Sharp* IR dan RTC masuk dalam mikrokontroler Arduino uno kemudian dari mikrokontroler dikirim melalui LoRa *Shield* dan antena. Data diterima melalui antena dan LoRa *Shield* dalam komunikasi *wireless* yang ada pada *device* Rx. Data tersebut kemudian masuk dan diproses Arduino uno pada Rx dan kemudian diteruskan ke mikrokontroler esp-32. Melalui modul *wifi* pada esp-32 data yang masuk diproses dan dikirim ke *output* yang ditampilkan pada *ThingSpeak*.

### 3.2.3 Pengujian *Hardware* dan Sistem

Pengujian *hardware* dan sistem dilakukan secara bersamaan setelah *hardware* telah dirakit dan *program* dapat berjalan pada *hardware*. Pengujian *sistem* dilakukan bertujuan untuk melihat dan mengetahui apakah sistem yang dirancang dapat berjalan sesuai dengan *flowchart* dan dapat diimplementasikan atau tidak. Untuk pengujian perangkat dapat dilakukan dengan melakukan percobaan pengambilan data dari setiap sensor. Pengujian yang dilakukan yaitu memvalidasi data yang dikeluarkan oleh setiap sensor. Validasi sensor GPS dilakukan dengan mengukur jarak selisih *latitute longitude* yang dikeluarkan oleh sensor pada *serial monitor* dengan nilai sensor *google map*.

Dalam pengujian sensor GPS dilakukan dengan menggunakan perhitungan dengan menggunakan formula persamaan (3.1):

$$\text{Jarak selisih} = \sqrt{(\text{lat1} - \text{lat2})^2 + (\text{long1} - \text{long2})^2} \times 111,319 \text{ km} \quad (3.1)$$

Keterangan :

*lat1* = Koordinat garis lintang GPS kendaraan.

*lat2* = Koordinat garis lintang *google maps*.

*long1* = Koordinat garis bujur GPS kendaraan.

*long2* = Koordinat garis bujur GPS *google maps*.

1 derajat bumi = 111,319 km [33].

Untuk mengukur data yang dikeluarkan oleh sensor jarak dapat dilakukan dengan mengukur keakuratan dengan meter sebagai alat ukur jarak yang digunakan. Dalam pengujian RTC dapat dilakukan dengan memvalidasi data yang dimunculkan pada *serial monitor* dan waktu jam *digital* yang berasal dari *website* BMKG. Selanjutnya untuk memastikan *hardware* dan sistem telah berfungsi dan berjalan yaitu dengan melihat data keseluruhan *device* yang dikirimkan telah masuk atau belum pada *web dashboard ThingSpeak*.

#### **3.2.4 Perakitan Hardware**

Perakitan *hardware* merupakan proses penyusunan komponen alat dan bahan yang bertujuan untuk merangkai *hardware*. Hal pertama yang dilakukan ialah pemasangan modul LoRa ke *board* ataupun ke mikrokontroler Arduino uno R3. Kemudian pemasangan sensor-sensor yang digunakan dalam *hardware*. Setelah semua komponen terpasang maka selanjutnya melakukan pemrograman. Pemrograman dilakukan untuk mengatur algoritma pengiriman data dan penerimaan data melalui jaringan LoRa. Ini melibatkan penulisan kode pemrograman yang sesuai untuk mikrokontroler Arduino uno R3 dan modul LoRa yang digunakan agar sistem komunikasi LoRa dapat berjalan. Setelah perakitan komponen dan pemrograman berjalan maka dilanjut dengan pengujian yang bertujuan untuk memastikan bahwa semua komponen berfungsi dengan baik dan sistem dapat beroperasi seperti yang diharapkan.

#### **3.2.5 Pengukuran Kinerja Sistem di Lapangan**

Pengukuran kinerja sistem dilakukan untuk mengevaluasi sejauh mana suatu sistem yang dirancang telah berjalan dengan melihat hasil dari pengujian *hardware* dan sistem, serta melihat data yang dikirimkan ke *ThingSpeak*. Hasil pengukuran boleh dibandingkan dengan standar atau kriteria yang telah ditetapkan yang bertujuan untuk menilai sejauh mana sistem dapat beroperasi memenuhi harapan. Ada beberapa kriteria parameter yang digunakan dalam pengukuran kinerja sistem perangkat penelitian ini, antara lain RSSI (*Received Signal Strength*

*Indicator*), SNR (*Signal to Noise Ratio*), ToA (*Time on Air*) dan *Delay*. Pengukuran perangkat akan disesuaikan dengan pengujian *hardware* dan sistem secara langsung yang dilakukan di lapangan. Penempatan *device* diletakkan pada tempat yang berbeda, sehingga pada saat pengolahan data memiliki perbedaan jarak antara *device* sebagai analisis pengukuran kinerja kuat sinyal komunikasi LoRa.