BAB III
PERANCANGAN SISTEM

Pada bab ini akan dijelaskan tentang perancangan sistem pengiriman dan penerimaan data, pemrosesan data, dan pencarian posisi sudut dari perangkat *tracker*. Perancangan sistem ini dibagi menjadi 5 bagian, yaitu :

1. Diagram blok sistem,

2. Diagram alir sistem,

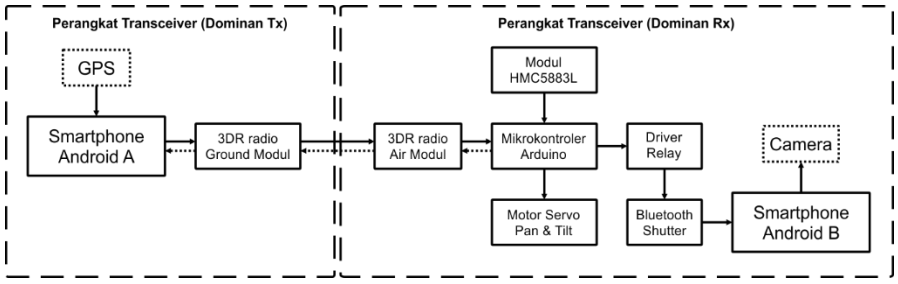
3. Perancangan mekanik,

4. Perancangan elektrikal, dan

5. Perancangan software.

1.1 Diagram blok sistem

Diagram blok sistem yang akan dibuat ditunjukkan pada gambar 3.1.



Gambar I.1 Diagram blok sistem

Perangkat *transceiver* (dominan Tx) terdiri dari *smartphone* android yang telah terinstal aplikasi kendali dan 3DR radio *ground* modul yang terhubung dengan *smartphone* menggunakan kabel USB OTG. Sedangkan perangkat *transceiver* (dominan Rx) terdiri dari 3DR radio *air* modul yang terhubung dengan mikrokontroler Arduino melalui pin serial, modul kompas hmc5883l sebagai penunjuk arah, motor servo sebagai penggerak bagian pan dan tilt, driver relay untuk mengaktifkan *bluetooth shutter* yang telah terhubung dengan *smartphone* yang digunakan untuk pengambilan gambar maupun video.

Berikut penjelasan blok-blok yang terdapat pada Gambar III.1 :

a. GPS

GPS disini menggunakan modul GPS yang ada dalam *Smartphone* Android A yang digunakan untuk mendapatkan lokasi dari *smartphone* dan melakukan set point posisi pada awal pengoperasian perangkat.

b. *Smartphone* Android A

Smartphone Android A merupakan bagian dari perangkat *transceiver* (dominan Tx) yang digunakan untuk menerima data dari GPS dan diolah untuk mendapatkan posisi dalam sudut dan juga jarak dari set point dalam satuan meter.

c. 3DR radio *Ground* Modul

3DR radio *Ground* Modul merupakan bagian dari perangkat *transceiver* (dominan Tx) yang terhubung dengan *smartphone* Android A melalui USB OTG. 3DR radio *Ground* Modul ini digunakan untuk mengirim data hasil olahan ke perangkat *transceiver* (dominan Rx) dan juga menerima *feedback* pada saat melakukan *beep* mode.

d. 3DR radio *Air* Modul

3DR radio *Air* Modul merupakan bagian dari perangkat *transceiver* (dominan Rx) yang digunakan untuk menerima data yang dikirim oleh 3DR radio *Ground* Modul yang kemudian akan diolah kembali oleh mikrokontroler.



Gambar I.2 Modul Air dan Ground Radio 3DR

e. Modul HMC5883L

Modul HMC5883L merupakan bagian dari perangkat *transceiver* (dominan Rx). Modul HMC5883L adalah modul kompas yang memanfaatkan *magnetometer* 3-

Axis. Modul ini akan mendapatkan data dari *magnetometer* yang akan diolah pada mikrokontroler.



Gambar I.3 Modul kompas HMC5883L

f. Motor Servo Pan & Tilt

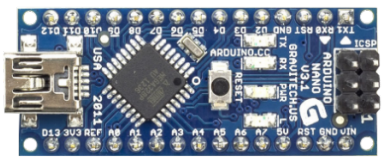
Motor Servo Pan & Tilt adalah bagian dari perangkat *transceiver* (dominan Rx) yang digunakan untuk menggerakan perangkat *transceiver* (dominan Rx) agar selalu mengarah pada perangkat *transceiver* (dominan Tx).



Gambar I.4 Motor Servo

g. Mikrokontroler Arduino

Mikrokontroler Arduino adalah bagian dari perangkat *transceiver* (dominan Rx) yang berfungsi untuk mengolah data masuk dari 3DR radio *Air* Modul, Modul HMC5883L. Mikrokontroler Arduino juga berfungsi untuk mengaktikan *relay driver* dan mengirimkan kode *feedback* ke perangkat *transceiver* (dominan Tx) melalui 3DR radio *Air* Modul.

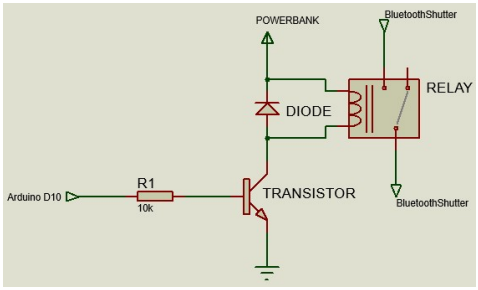


Gambar I.5 Arduino Nano

h. *Driver Relay*

Driver Relay adalah bagian dari perangkat *transceiver* (dominan Rx). *Driver Relay* terdiri atas rangkaian yang mengaktifkan transistor sebagai saklar

menggunakan nilai *output* dari pin Mikrokontroler Arduino dan akan mengaktifkan *Bluetooth Shutter*.



Gambar I.6 Rangkaian skematik driver relay

i. Bluetooth Shutter

Bluetooth Shutter adalah bagian dari perangkat *transceiver* (dominan Rx) yang diaktifkan oleh *driver relay* untuk membuat *smartphone* Android B mengaktifkan kamera untuk mengambil gambar maupun video.



Gambar I.7 Bluetooth Shutter

j. *Smartphone* Android B

Smartphone Android B adalah bagian dari perangkat *transceiver* (dominan Rx) yang berfungsi untuk mengambil gambar maupun video menggunakan aplikasi kamera yang ada pada *smartphone*.

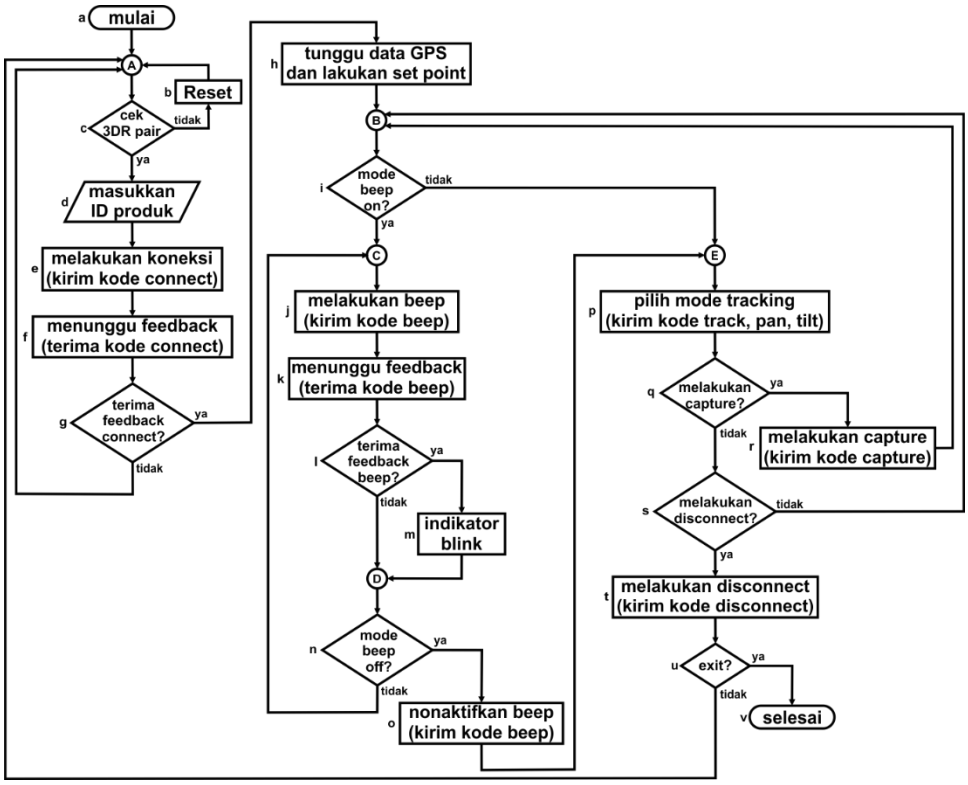
k. Camera

Camera yang digunakan adalah perangkat kamera yang terpasang pada *Smartphone* Android B.

1.2 Diagram alir sistem

Diagram alir sistem yang akan dibuat ditunjukkan pada gambar 3.3 untuk perangkat *transceiver* (dominan Tx) dan gambar 3.4 untuk *transceiver* (dominan Rx).

1.2.1 *Transceiver* (Dominan Tx)



Gambar I.8 Diagram alir transceiver (dominan Tx)

Berikut adalah penjelasan diagram alir sistem bagian *transceiver* (dominan Tx) :

a. Mempersiapkan semua kebutuhan perangkat *transceiver* (dominan Tx).

b. Mematikan dan menyalakan kembali radio 3DR.

c. Melakukan pengecekan pada radio 3DR apakah sudah pairing atau tidak. Jika 3DR sudah pair, lanjutkan ke poin d. Jika tidak, kembali ke poin b.

d. Memasukkan ID produk pada *field* ID.

e. Menekan tombol *connect* pada aplikasi *smartphone*. Aplikasi *smartphone* akan mengirim kode dengan format (ID,kode,pan,tilt) dan akan dikirim melalui radio 3DR ke perangkat *transceiver* (dominan Rx). Kode connect adalah 1.

f. Aplikasi *smartphone* akan menunggu untuk menerima *feedback* yang dikirim oleh perangkat *transceiver* (dominan Rx).

g. Aplikasi *smartphone* melakukan pengecekan apakah telah menerima kode *connect* yang dikirim dari perangkat *transceiver* (dominan Rx). Jika kode *feedback* = "*connect*", maka aplikasi akan masuk ke menu *tracking* dan lanjut ke poin h. Jika tidak maka kembali ke poin c.

h. Setelah masuk ke menu *tracking*, tunggu sampai data *latitude*, *longitude*, dan *altitude* dari GPS didapatkan. Jika data GPS sudah didapatkan, lakukan *set point*. Setelah melakukan *set point*, aplikasi akan mengolah data GPS untuk mendapatkan nilai sudut pan, tilt, dan jarak perangkat *transceiver* (dominan Tx) dan perangkat *transceiver* (dominan Rx).

i. Aplikasi *smartphone* melakukan pengecekan apakah mode *beep* aktif. Jika *beep* aktif, lanjutkan ke poin j. Jika tidak, lanjutkan ke poin p.

j. Aplikasi *smartphone* melakukan pengiriman kode *beep* ke perangkat *transceiver* (dominan Rx). Kode untuk *beep* aktif adalah 2.

k. Aplikasi *smartphone* akan menunggu untuk menerima *feedback* yang dikirim oleh perangkat *transceiver* (dominan Rx).

l. Aplikasi *smartphone* melakukan pengecekan apakah telah menerima *feedback* kode *beep*. Jika kode *feedback* = "*beepon*", lanjutkan ke poin m. Jika tidak, kembali ke poin i.

m. Aplikasi *smartphone* melakukan *blinking* pada icon indikator *beep* setelah menerima *feedback* kode *beep* dari *transceiver* (dominan Rx) dan kembali ke poin i.

n. Aplikasi *smartphone* akan melakukan pengecekan apakah mode *beep* di nonaktifkan. Jika ya, lanjutkan ke poin o. Jika tidak, kembali ke poin j.

o. Aplikasi *smartphone* akan mengirimkan kode *beep* nonaktif ke perangkat *transceiver* (dominan Rx) lalu lanjutkan ke poin p. Kode *beep* nonaktif adalah 3.

p. Aplikasi *smartphone* akan mengirim kode *tracking* beserta nilai sudut pan dan juga sudut tilt yang didapatkan pada poin h. Kode *tracking* baik manual ataupun otomatis adalah 4.

q. Aplikasi *smartphone* melakukan pengecekan apakah tombol *capture* aktif. Jika ya, lanjutkan ke poin p. Jika tidak, lanjutkan ke poin q.

r. Aplikasi *smartphone* melakukan pengiriman kode *capture* ke perangkat *transceiver* (dominan Rx) dan kembali ke poin i. Kode *capture* adalah 5.

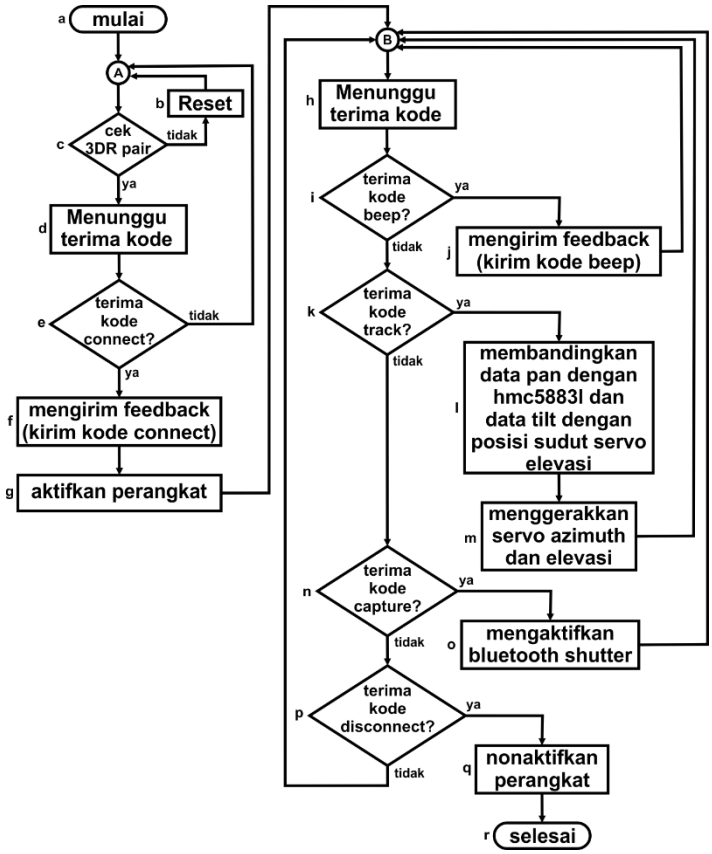
s. Aplikasi *smartphone* melakukan pengecekan apakah dilakukan *disconnect*. Jika ya, lanjutkan ke poin r. Jika tidak, kembali ke poin i.

t. Aplikasi *smartphone* melakukan pengiriman kode *disconnect* ke perangkat *transceiver* (dominan Rx) dan aplikasi akan kembali ke menu utama lalu kembali ke poin i. Kode *disconnect* adalah 0.

u. Aplikasi *smartphone* melakukan pengecekan apakah tombol *exit* aktif. Jika ya, maka lanjutkan ke poin t. Jika tidak, kembali ke poin c.

v. Aplikasi *smartphone* telah diberhentikan.

1.2.2 *Transceiver* (Dominan Rx)



Gambar I.9 Diagram alir transceiver (dominan Rx)

Berikut adalah penjelasan diagram alir sistem bagian *transceiver* (dominan Rx) :

a. Mempersiapkan semua kebutuhan perangkat *transceiver* (dominan Rx).

b. Memasang kembali *powerbank* dan menyalakan kembali perangkat.

c. Melakukan pengecekan pada radio 3DR apakah sudah pairing atau tidak. Jika 3DR sudah pair, lanjutkan ke poin d. Jika tidak, kembali ke poin b.

d. Radio 3DR menunggu sampai menerima kode yang dikirim oleh perangkat *transceiver* (dominan Tx) dan diproses oleh mikrokontroler arduino.

e. Mikrokontroler Arduino melakukan pengecekan apakah kode yang diterima adalah kode *connect*. Jika terima kode = 0, lanjutkan ke poin f. Jika tidak, kembali ke poin c.

f. Mikrokontroler mengirimkan kode feedback connect ke perangkat *transceiver* (dominan Tx) setelah menerima kode *connect*. Kode *feedback connect* adalah "*connect*".

g. Mikrokontroler mengaktifkan perangkat dengan menyalakan indikator LED sebagai tanda bahwa perangkat sudah terhubung satu sama lain dan dapat digunakan.

h. Radio 3DR kembali menunggu sampai menerima kode yang dikirim oleh perangkat *transceiver* (dominan Tx) dan diproses oleh mikrokontroler.

i. Mikrokontroler Arduino melakukan pengecekan apakah kode yang diterima adalah kode *beep*. Jika terima kode = 2, lanjutkan ke poin j. Jika tidak, lanjutkan ke poin k.

j. Mikrokontroler mengirimkan kode *feedback* kode *beep* ke perangkat *transceiver* (dominan Tx) setelah menerima kode *beep* aktif lalu kembali ke poin h. Kode *feedback beep* adalah "*beep*".

k. Mikrokontroler Arduino melakukan pengecekan apakah kode yang diterima adalah kode *track*. Jika terima kode = 4, lanjutkan ke poin l. Jika tidak, lanjutkan ke poin n.

l. Mikrokontroler akan membandingkan nilai pan yang diterima dengan nilai sudut dari modul kompas HMC5883L dan membandingkan nilai tilt yang diterima dengan posisi sudut dari servo elevasi.

m. Mikrokontroler kemudian akan mengirimkan pulsa PWM ke servo *azimuth* untuk menggerakkannya dan membandingkan kembali dengan kompas sampai nilai kompas dan nilai pan sama. Sama halnya dengan servo *azimuth*, mikrokontroler akan mengirimkan pulsa PWM ke servo elevasi sampai posisi sudut servo sama dengan nilai tilt. Setelah sama, kembali ke poin h.

n. Mikrokontroler Arduino melakukan pengecekan apakah kode yang diterima adalah kode *capture*. Jika terima kode = 5, lanjutkan ke poin o. Jika tidak, lanjutkan ke poin p.

o. Mikrokontroler akan mengaktifkan *relay* sehingga *bluetooth shutter* akan mengirimkan perintah *capture* ke *smartphone* yang terpasang pada holder di perangkat *transceiver* (dominan Rx).

p. Mikrokontroler Arduino melakukan pengecekan apakah kode yang diterima adalah kode *disconnect*. Jika terima kode = 0, lanjutkan ke poin q. Jika tidak, kembali ke poin h.

q. Mikrokontroler akan menonaktifkan perangkat *transceiver* (dominan Rx).

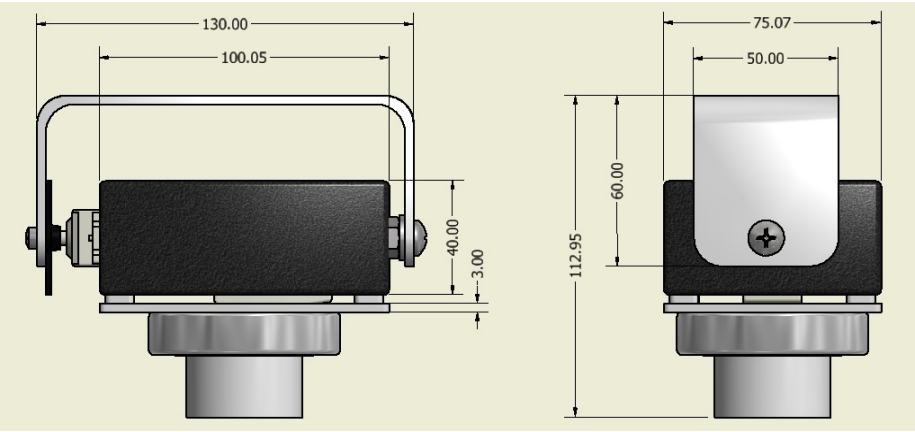
r. Perangkat telah dinonaktifkan.

1.3 Perancangan mekanik

Perangkat *transceiver* (dominan Rx) dirancang dengan ukuran dan berat sekecil mungkin sehingga tidak terlalu membebani kekuatan dari kaki tripod yang dimiliki oleh pengguna saat digunakan dan juga diharapkan mampu menahan beban dari smartphone android yang akan dipasang pada *holder*. Sedangkan perangkat *transceiver* (dominan Tx) dirancang dengan menggunakan *armband* yang telah dimodifikasi sehingga nyaman dan mudah untuk digunakan oleh pengguna.

1.3.1 Mekanik Transceiver (Dominan Rx)

Perangkat *transceiver* (dominan Rx) tersusun dari beberapa bagian yaitu lengan elevasi, *box* kontrol, *bearing*, dan penyangga nylon.



Gambar I.10 Perangkat transceiver (dominan Rx) tampak depan dan samping

1. Lengan elevasi

Lengan elevasi adalah bagian *tracker* yang berfungsi untuk menggerakkan *smartphone* android mengikuti perpindahan pada sudut elevasi. Bagian kanan lengan dihubungkan dengan motor servo *tilt* dan bagian kiri lengan dihubungkan pada *box* dengan poros penyeimbang. Dimensi dari lengan elevasi adalah sebagai berikut :

a. Panjang lengan = 130 mm

b. Lebar lengan = 50 mm

c. Tinggi lengan = 60 mm

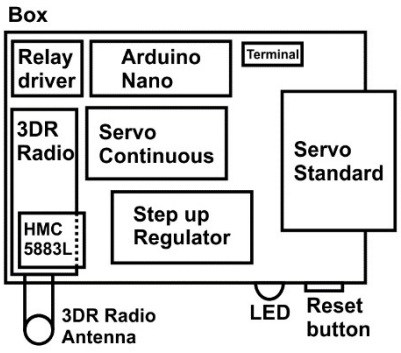
2. Box Kontrol

Box kontrol adalah kotak hitam yang berisi semua modul yang digunakan untuk menjalankan bagian perangkat *transceiver* (dominan Rx). Modul-modul tersebut antara lain adalah motor servo pan (continuous) dan tilt (standard), kompas HMC5883L, radio 3DR Air Modul, *Step up regulator*, *relay driver*, terminal daya, led indikator, dan tombol *reset*. Dimensi dari *box* kontrol adalah sebagai berikut :

a. Panjang *box* = 100 mm

b. Lebar *box* = 75 mm

c. Tinggi *box* = 40 mm



Gambar I.11 Penempatan modul-modul dalam box kontrol

3. Penyangga nylon dan bearing

Pada perangkat *transceiver* (dominan Rx) nylon terhubung dengan servo pan yang berfungsi untuk menyangga perangkat *transceiver* (dominan Rx) dengan tripod ketika melakukan pergerakan azimuth. *Bearing* berfungsi untuk menyeimbangkan *box* kontrol yang terhubung dengan nylon. Dimensi dari *bearing* dan nylon adalah sebagai berikut :

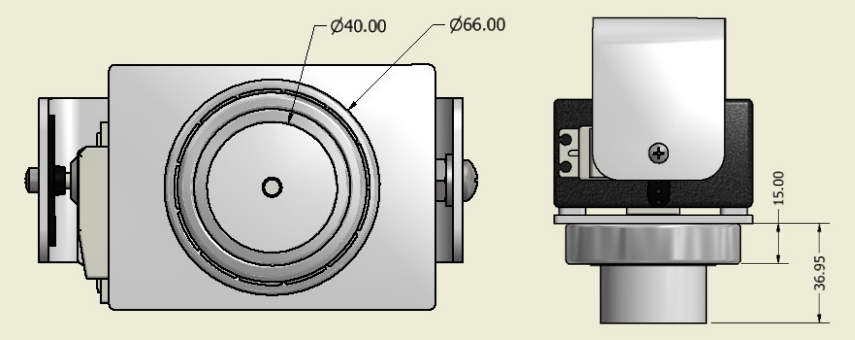
a. Diameter *bearing* luar = 66 mm

b. Diameter *bearing* dalam = 40 mm

c. Tebal *bearing* = 15 mm

d. Diameter nylon = 40 mm

e. Tebal nylon = 36,95 mm



Gambar I.12 Perangkat transceiver (dominan Rx) tampak bawah dan samping

1.3.2 Mekanik Transceiver (Dominan Tx)

Perancangan mekanik perangkat *transceiver* (dominan Tx) dirancang menggunakan *armband* yang telah dimodifikasi sehingga memiliki tempat penyimpanan radio 3DR yang telah terpasang kabel OTG. *Armband* yang digunakan mampu menyimpan *smartphone* android dengan ukuran layar maksimal 4,7 in. Dimensi dari *armband* yang digunakan adalah sebagai berikut :

a. Panjang maksimal = 460.10 mm

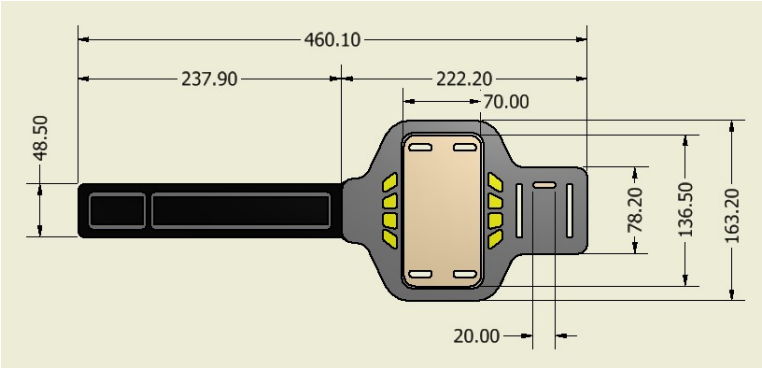
b. Lebar maksimal = 163.20 mm

c. Panjang tali pengikat = 273.90 mm

d. Tinggi layar transparan = 136.50 mm

e. Lebar layar transparan = 70 mm

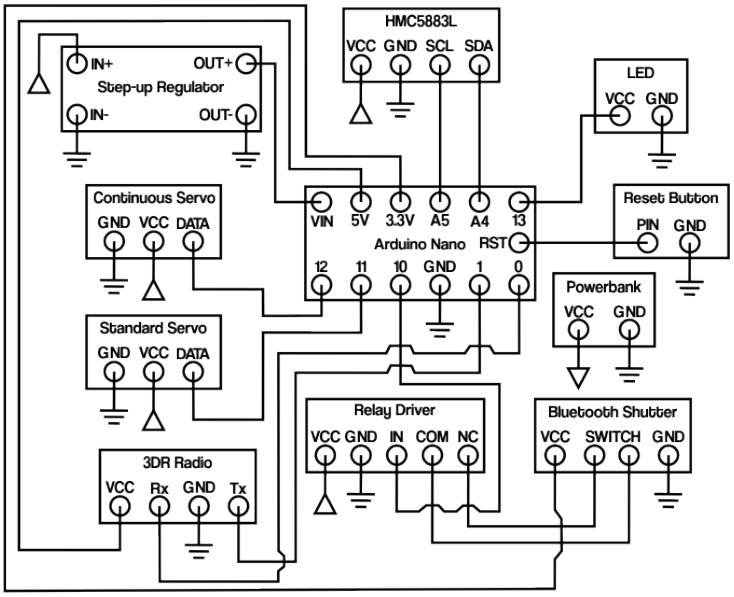
f. Lebar tempat radio = 20 mm



Gambar I.13 Desain armband transceiver (dominan Tx)

1.4 Perancangan elektrikal

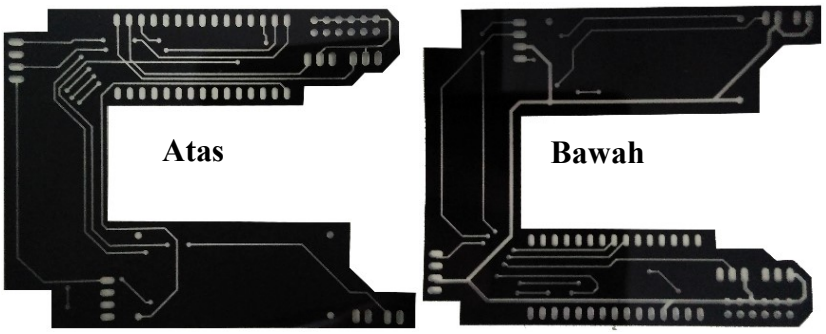
Perancangan elektrikal dibutuhkan pada pembuatan perangkat baik *transceiver* (dominan Rx) maupun *transceiver* (dominan Tx) untuk meningkatkan keamanan serta efisiensi penggunaan dan peletakan komponen.



Gambar I.14 Diagram Skematik Perangkat Transceiver (Dominan Rx)

Sumber daya utama dari perangkat *transceiver* (dominan Rx) adalah *powerbank*. *Powerbank* yang digunakan memiliki tegangan keluaran sebesar ±5 V dan mampu mensuplai arus hingga ±2 A. Seluruh komponen yang digunakan pun membutuhkan sumber tegangan ±5 V kecuali *bluetooth shutter* yang membutuhkan tegangan ±3 V. Arduino Nano mendapatkan tegangan yang telah dinaikkan oleh *step-up regulator*. Selain untuk memaksimalkan kinerja dari Arduino Nano, step-up regulator juga digunakan untuk mengamankan Arduino nano dari *noise* yang ditimbulkan oleh motor servo. Radio 3DR pun mendapatkan tegangan dari Arduino Nano dengan tujuan yang sama.

Pada perangkat *transceiver* (dominan Tx) sumber daya didapatkan oleh 3DR radio dari smartphone android melalui kabel USB OTG yang berfungsi juga sebagai media komunikasi antara *smartphone* android dan 3DR radio ketika melakukan pengiriman kode-kode operasi dan juga data sudut baik pan maupun tilt dan ketika menerima *feedback* berupa kode ketika melakukan operasi *connect* dan *beep*.



Gambar I.15 Desain routing PCB layer atas dan bawah

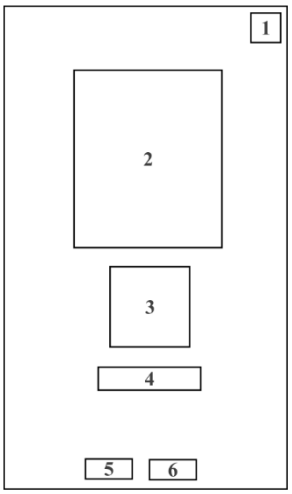
1.5 Perancangan Software

Software yang dibuat dalam sistem ini digunakan pada perangkat *transceiver* (dominan Tx) yang berfungsi sebagai pengendali dari perangkat *transceiver* (dominan Rx).

1.5.1 Perancangan GUI

Perangkat lunak dari sistem ini dibuat menggunakan *basic4android* yang akan digunakan pada bagian *transceiver* (dominan Tx). Perangkat lunak yang

dibuat terdiri dari halaman utama, halaman about, halaman help, dan halaman *tracking.* Berikut adalah keterangan dari tampilan *graphical user interface* (*GUI)* yang telah dibuat :



Gambar I.16 Tampilan menu awal perangkat lunak

Keterangan :

1. Tombol *exit* untuk keluar dari aplikasi.

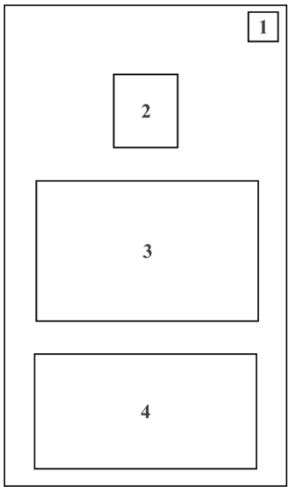
2. Logo.

3. Tombol *connect* untuk memulai koneksi.

4. Field ID produk untuk memasukkan ID produk sebelum melakukan koneksi.

5. Tombol *about* untuk masuk ke halaman tentang perangkat.

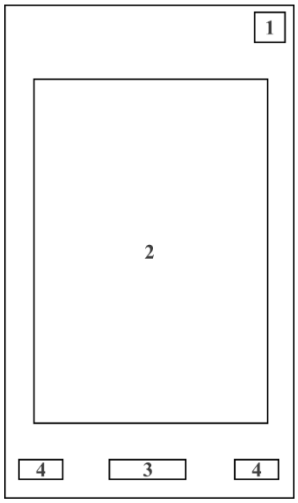
6. Tombol *help* untuk masuk ke halaman bantuan.



Gambar I.17 Tampilan halaman about

Keterangan :

1. Tombol *back* untuk kembali ke menu awal.
2. Logo.
3. Tombol-tombol share dan follow di media sosial
4. Informasi kontak.



Gambar I.18 Tampilan halaman help

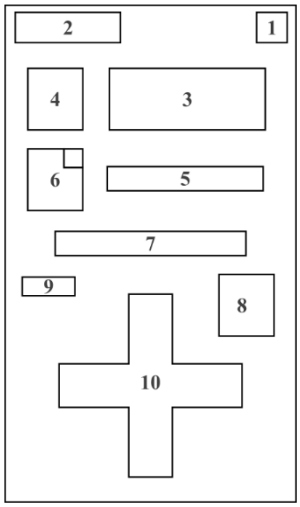
Keterangan :

1. Tombol *back* untuk kembali ke menu utama.

2. Petunjuk tombol dan penggunaan aplikasi.

3. Penunjuk halaman.

4. Tombol pindah halaman.



Gambar I.19 Tampilan menu tracking

Keterangan :

1. Tombol *back* untuk melakukan *disconnect* dan kembali ke menu utama.

2. Logo dan nama aplikasi.

3. Tampilan data GPS.

4. Tombol *set point* untuk melakukan *set point* posisi.

5. Tampilan jarak.

6. Tombol *beep* dan indikator untuk mengaktifkan mode *beep*.

7. Tampilan sudut *pan* dan *tilt*.

8. Tombol *capture* untuk mengirimkan perintah *capture* atau *record*.

9. Tombol *switch track* untuk mengganti mode *track* manual atau *auto*.

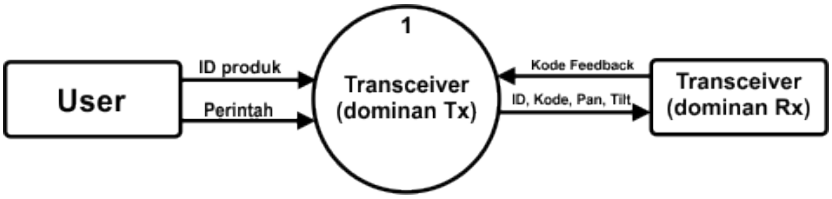
10. *Pad* untuk melakukan *tracking* manual.

1.5.2 Data Flow Diagram (DFD)

Pada bagian ini akan dijelaskan mengenai *Data Flow Diagram* untuk software kendali yang digunakan pada perangkat *transceiver* (dominan Tx).

1. DFD Level 0

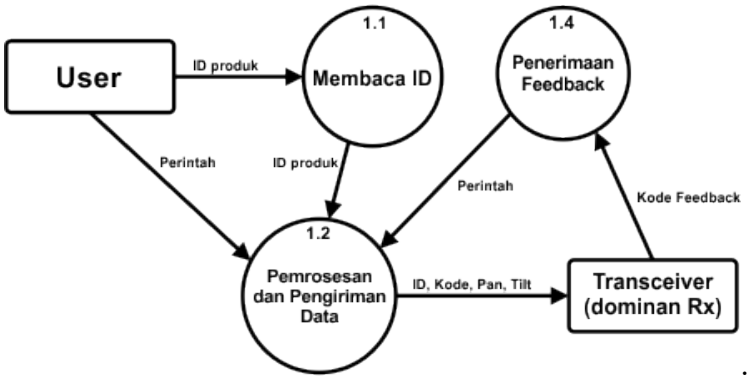
Pada DFD ini proses 1 merupakan proses keseluruhan yang dilakukan oleh aplikasi yang dibuat untuk perangkat *transceiver* (dominan Tx). Aplikasi akan menerima ID produk dan juga perintah-perintah yang diinput oleh user. Aplikasi ini kemudian akan memroses input yang masuk dan mengirimkan data ke perangkat *transceiver* (dominan Rx). Pada perintah-perintah tertentu, aplikasi juga akan menerima kode *feedback* dari perangkat *transceiver* (dominan Rx).



Gambar I.20 Data Flow Diagram Level 0

2. DFD Level 1 Proses 1

Proses 1 memiliki beberapa proses utama yaitu membaca input ID, melakukan pemrosesan dan pengiriman data, dan penerimaan kode *feedback*.



Gambar I.21 Data Flow Diagram Level 1 Proses 1

a. Proses 1.1 (Membaca ID) :

Pada proses ini software membaca apakah ID sudah diinput atau tidak. Jika tidak maka peringatan berupa *pop-up* akan muncul agar dan

memberitahu *user* untuk mengisi ID terlebih dahulu. Setelah ID dimasukkan, *software* akan menunggu sampai perintah *connect* dilakukan.

b. Proses 1.2 (Pemrosesan dan Pengiriman Data) :

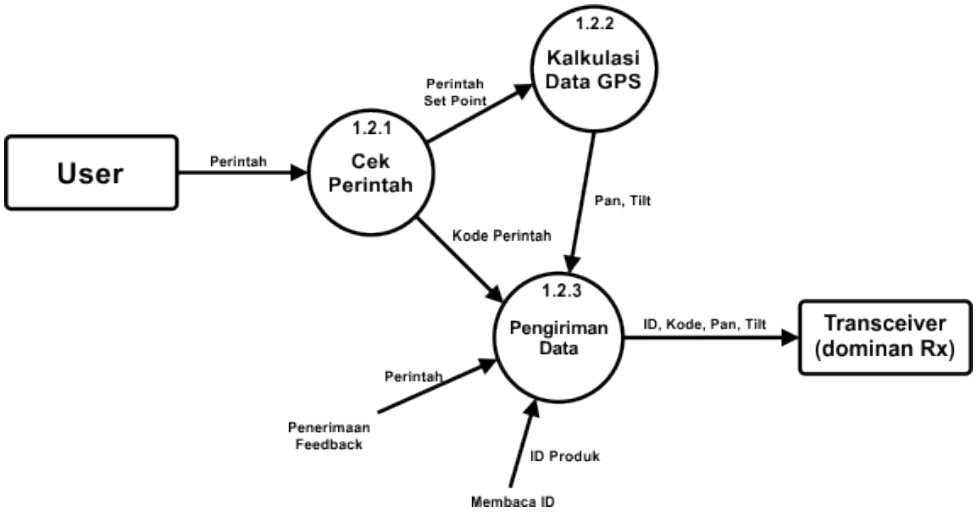
Pada proses ini *software* akan membaca perintah yang dilakukan oleh *user* dengan melakukan pengecekan pada status tombol yang aktif. Setelah melakukan pembacaan data *software* akan memroses perintah yang aktif dan mengirimkan data yang diperlukan.

c. Proses 1.3 (Penerimaan Feedback) :

Pada proses ini *software* akan menerima kode *feedback* dari perangkat *transceiver* (dominan Rx) ketika mengirimkan perintah *connect* dan *beep*.

3. DFD Level 2 Proses 1.2

Sebelum melakukan pengiriman data, *software* akan melakukan beberapa langkah dalam memroses data-data yang diterima.



Gambar I.22 Data Flow Diagram Level 2 Proses 1.2

a. Proses 1.2.1 (Cek Perintah)

Software akan melakukan pengecekan pada perintah yang diterirma dari *user*. Perintah-perintah yang tersedia yaitu *connect*, *disconnect*, *set point* (*tracking*), *beep*, dan *capture*.

b. Proses 1.2.2 (Kalkulasi Data GPS)

Proses ini dilakukan jika perintah set point telah aktif. Data GPS yang didapatkan akan dikalkulasi sehingga mendapatkan nilai sudut pan dan sudut tilt. Setelah mendapatkan nilai sudut, data kemudian akan dikirimkan.

c. Proses 1.2.3 (Pengiriman Data)

Setelah semua data didapatkan, data akan diurutkan dengan format pengiriman data ID, kode, pan, tilt, ~. ID adalah ID dari produk yang tertera pada perangkat *transceiver* (dominan Rx), kode adalah kode perintah dari setiap fungsi, pan dan tilt adalah nilai sudut yang didapatkan setelah melakukan kalkulasi dari data GPS. Symbol "~" adalah ekor dari data yang terkirim sebagai penunjuk bahwa data yang telah diterima secara lengkap.