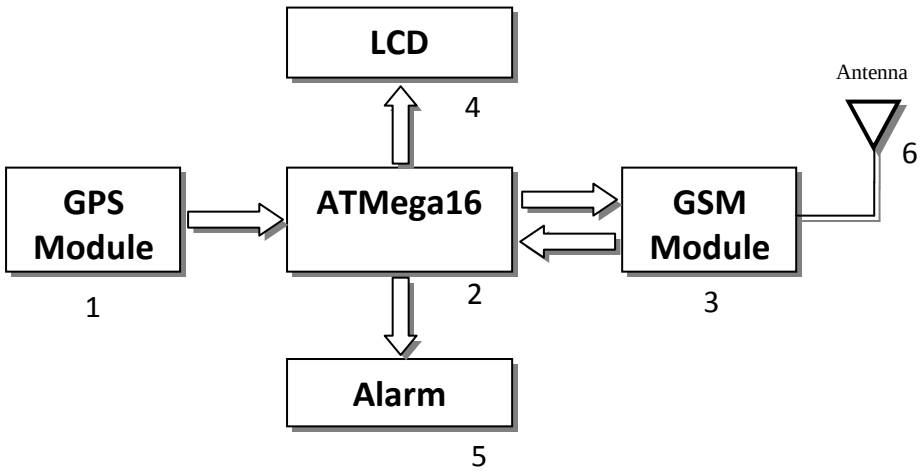
BAB III
PERANCANGAN SISTEM

3.1 Parancangan Sistem

Blok diagram dari sistem yang dibuat pada perancangan Tugas Akhir ini terbagi menjadi 2 bagian, yaitu bagian pengirim dan penerima pada kendaraan patroli dan bagian pengirim serta penerima pada bagian komputer.

3.1.1 Bagian Pengirim dan Penerima Pada Kendaraan Patroli

Diagram blok dari sistem yang dibuat pada perancangan tugas akhir ini terbagi menjadi dua bagian, yaitu diagram blok pada bagian kendaraan patroli, dan diagram blok pada bagian komputer. Diagram blok pada kendaraan patroli dapat di lihat pada gambar 3.1.



Gambar 3.1 *Diagram Blok Alat Pada Kendaraan Patroli*

Keterangan diagram blok Bagian Pengirim dan Penerima Pada Kendaraan Patroli:

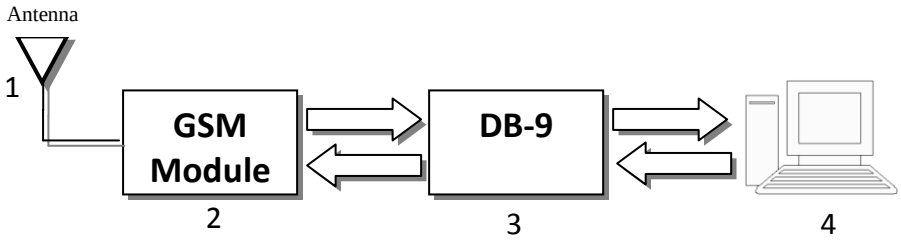
1. Modul GPS HOLUX GR-89 : Sebagai alat penerima sinyal SiRFstarII™ yang dapat memberikan informasi waktu, garis lintang, garis bujur serta ketinggian dari permukaan air laut secara serial (*TTL*).
2. Mikrokontroler ATmega16 : Menjadi pusat pengolah data, baik menyeleksi data maupun mengatur kinerja dari modul lain. Sehingga data yang telah diproses dapat dikirim ke bagian penerima.

3. Modul GSM SIM900 : Sebagai alat komunikasi antara pengirim dan penerima. Data yang dikirimkan berupa titik koordinat, data dikirim melalui SMS ke modul GSM penerima.
4. LCD : Berfungsi untuk menampilkan tulisan sebagai pemberitahuan.
5. Alarm : Berfungsi sebagai pengingat berupa suara.
6. Antenna : Sebagai pengirim dan penerima gelombang frekuensi radio.

Modul GPS akan memberikan data informasi berupa titik koordinat yang akan diolah oleh mikrokontroler ATmega16, setelah mendapatkan data yang di inginkan maka mikrokontroler akan melakukan penyeleksian data header GPS berupa informasi garis lintang dan informasi garis bujur. Data informasi tersebut dikirim ke modul GSM (*Global System for Mobile communication*) tipe SIM900 untuk di transmisikan antara modul GSM yang satu dengan yang lainnya. Sedangkan alarm dan LCD digunakan untuk pengingat dan menampilkan peringatan.

3.1.2 Bagian Pengirim dan Penerima Pada Operator

Diagram blok pada bagian komputer dari sistem yang di buat dapat di lihat pada gambar 3.2.



Gambar 3.2 *Diagram Blok Alat Pada Operator*

Keterangan diagram blok penerima:

1. Antenna : Sebagai penerima gelombang frekuensi radio.
2. Modul GSM SIM900 : Sebagai alat komunikasi antara pengirim dan penerima. Data yang dikirimkan berupa titik koordinat, data dikirim melalui SMS ke modul GSM penerima.

3. DB-9 : Merupakan *port* serial komputer. Berfungsi untuk menerima data serial dari modem penerima dengan level RS-232.
4. Komputer : Terpasang aplikasi yang menampilkan data GPS berupa informasi garis lintang dan informasi garis bujur. Selain itu, komputer PC juga menjadi pusat penyimpanan data informasi GPS.

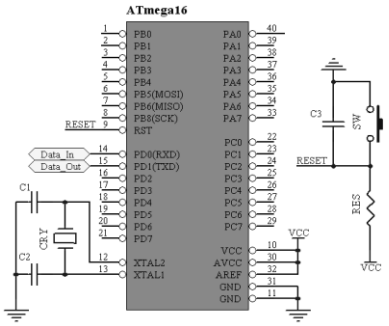
Data dari modul GSM yang berada di bagian pengirim diterima oleh modul GSM kembali yang berada di bagian penerima. Data yang dikirimkan dari modul GSM ke PC terlebih dahulu dikoneksikan dengan MAX 232, dimana MAX 232 adalah sebagai *driver* tegangan yang akan mengkonversi tegangan dari *hardware* agar sesuai dengan tegangan pada komputer sehingga dapat dibaca.

3.2 Perancangan Perangkat Keras

Pada perancangan perangkat keras, terdiri dari sistem minimum ATmega16, modul GSM (SIM900), modul GPS (HOLUX GR-89), antarmuka konektor DB-9 (RS-232) dan sebuah komputer. Pada bagian ini data yang dikirimkan berupa informasi titik koordinat pada suatu tempat, diharapkan bisa diterima oleh bagian penerima untuk diproses, disimpan dan ditampilkan pada layar (monitor).

3.2.1 Mikrokontroler ATmega16

Pada tugas akhir ini menggunakan mikrokontroler ATmega16 ditujukan untuk memberikan perintah ke GPS HOLUX GR-89 untuk menerima data serial dari GPS dan memberikan perintah kepada GSM untuk mengirim dan menerima data. Gambar rangkaian sistem minimum ATmega16 dapat di lihat pada gambar 3.3.



Gambar 3.3 *Rangkaian Sistem Minimum ATmega16*

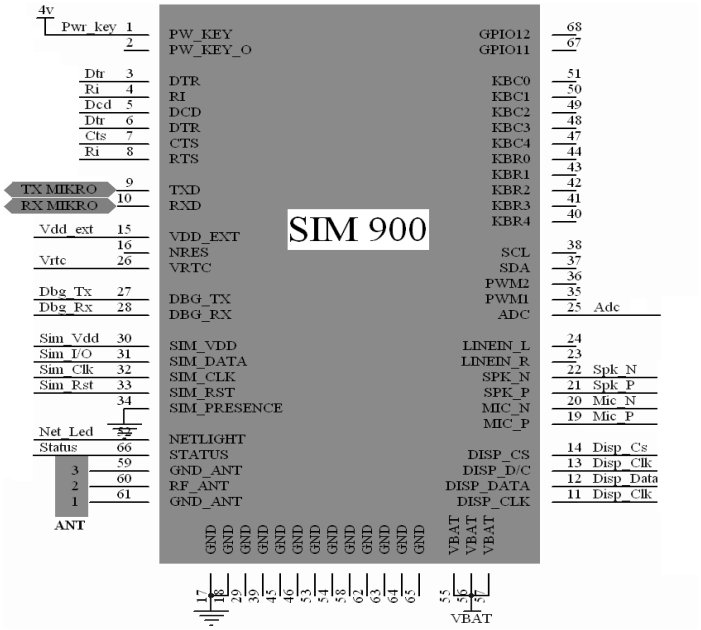
Tabel 3.1 menunjukan penggunaan pin-pin pada mikrokontroler ATmega16:

Tabel 3.1 *Pin-pin yang digunakan Pada ATmega16*

Nama Port	No.Pin	Tipe Pin	Fungsi
PORT D.0	14	I/O	Menerima data serial dari GPS
PORT D.I	15	I/O	Mengirim data ke Modul SIM900

3.2.2 Modul GSM SIM900

Alokasi frekuensi Modul GSM SIM900 yang dipakai di Indonesia sama dengan yang dipakai di sebagian besar dunia terutama Eropa yaitu pada pita 900 MHz, yang dikenal sebagai GSM900, dan pada pita 1800 MHz, yang dikenal sebagai GSM1800 atau DCS (*Digital Communication System*). Daya yang dibutuhkan berada pada tegangan 3.4 – 4.5 volt. Konfigurasi pin dapat di lihat pada gambar 3.4.

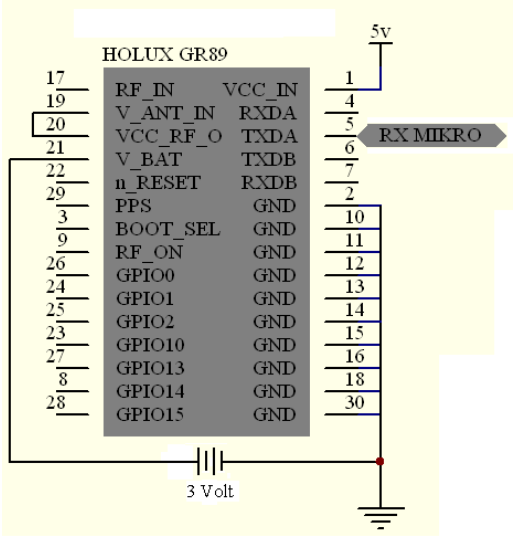


Gambar 3.4 *Konfigurasi Pin GSM SIM900*

3.2.3 Modul GPS HOLUX GR89

Modul GPS HOLUX GR89 memiliki sensitifitas tinggi untuk pencarian sinyal SiRFstarII™ dengan dua pilihan sumber tegangan yang dapat digunakan yaitu 3.3V sampai 5V. Pada rangkaian, modul HOLUX GR89 diberi tegangan 5V dan tegangan antenna sebesar 3V karena catu daya yang digunakan sebesar ±5V.

Baterai *Backup* (pin 21) sebesar 2.6 – 3.6V merupakan tegangan yang diperlukan untuk melakukan penyimpanan data sebelumnya, sedangkan pin 16 (nRESET) digunakan untuk melakukan *reset* pada sistem, sehingga modul HOLUX GR89 melakukan *booting* dan semua data tersimpan dihapus. Gambar rangkaian modul HOLUX GR89 dapat di lihat pada gambar 3.5.



Gambar 3.5 *Rangkaian HOLUX GR89*

Keluaran pada pin 5 (TXDA) adalah data serial berupa karakter dan mempunyai identitas. Identitas dapat diartikan sebagai nama header yang akan dikirim berikutnya. Seperti pada header $GPRMC, data 0..5 menunjukan identitas (GPRMC) dan data berikutnya mulai dari 6..56 merupakan urutan data informasi posisi GPS. Untuk lebih jelasnya ditunjukan oleh Tabel 3.2.

Tabel 3.2 *Header GPRMC pada GPS*

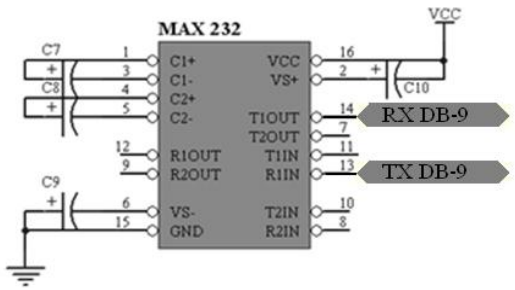
Name	Example	HDR	Unit	Description
Message ID	$GPRMC	0-5		RMC protocol
UTC Time	022054.410	7-16		hhmmss.sss
Latitude (Garis Lintang)	0654.0438	18-26		ddmm.mmmm
N/S Indicator	S	28		N=north or S=South
Longitude (Garis Bujur)	10748.0878	30-39		Dddmm.mmmm
E/W Indicator	E	41		E=east or W=West
Position fix indicator	1	43		
Satellite used	06	45-46		Range 0-12

HDOP	1.2	48-50		Horizontal dilution of precision
MSL altitude	884.6	52-56	M	

3.2.4 Max 232

Kegunaan IC MAX232 adalah sebagai *driver*, yang akan mengkonversi nilai tegangan atau kondisi logika *TTL* dari mikrokontroler agar sesuai dengan level tegangan pada modem komunikasi yang digunakan. IC yang dipakai pada sistem ini memiliki 16 pin dengan tegangan sebesar 5 Volt.

Pada dasarnya IC ini memerlukan komponen tambahan berupa kapasitor ekternal yang dipasangkan pada pin-pin tertentu. Kapasitor ini merupakan rangkaian baku yang berfungsi sebagai *charge pump* untuk menyuplai muatan ke bagian pengubah tegangan, dimana nilai setiap kapasitor yang dipakai bernilai 1uF. Gambar rangkaian Max 232 dapat di lihat pada gambar 3.6.

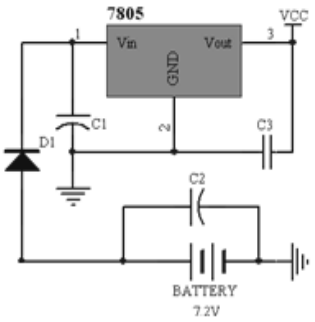


Gambar 3.6 *Rangkaian MAX232*

3.2.5 Catu Daya

Catu daya adalah faktor pendukung yang sangat penting, karena mikrokontroler, modul-modul dan sensor dapat bekerja karena adanya tegangan. Untuk mencatu semua modul yang terpasang maka diperlukan tegangan sebesar 5V DC. Tegangan yang digunakan berasal dari baterai *rechargeable* yang mempunyai nilai tegangan 1.2 Volt per satu baterai. Tegangan total 7,2 Volt didapat dari 6 buah baterai yang disusun secara seri, untuk dapat mencatu pada tegangan 5 volt maka diperlukan sebuah IC regulator LM7805. Dengan tegangan sumber 7,2 Volt IC ini mampu memberikan *output* tegangan yang sesuai yaitu

berkisar antara 5 Volt (nilai terukur 4.9 Volt). Gambar rangkaian dapat di lihat pada gambar 3.6.



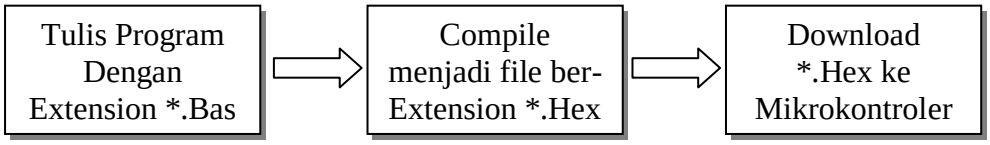
Gambar 3.7 *Rangkaian Regulator 5V DC*

3.3 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada sistem terbagi menjadi dua bagian yaitu pada mikrokontroler atau bagian pengirim dan perangkat lunak pada bagian komputer PC sebagai program antarmuka.

3.3.1 Pemrograman pada Mikrokontroler ATmega16

Pada pemrograman mikrokontroler, digunakan bahasa pemrograman Basic dengan program Basic *compiler* Bascom Avr. Setiap file yang dibuat akan disimpan dengan *extension* *.Bas. File yang telah di *compile* akan menghasilkan file baru dengan extension *.hex. File ini kemudian dimasukan ke dalam *flash* mikrokontroler melalui alat *Flash PEROM Microcontroller*. Langkah mendownload program dapat di lihat pada gambar 3.8.

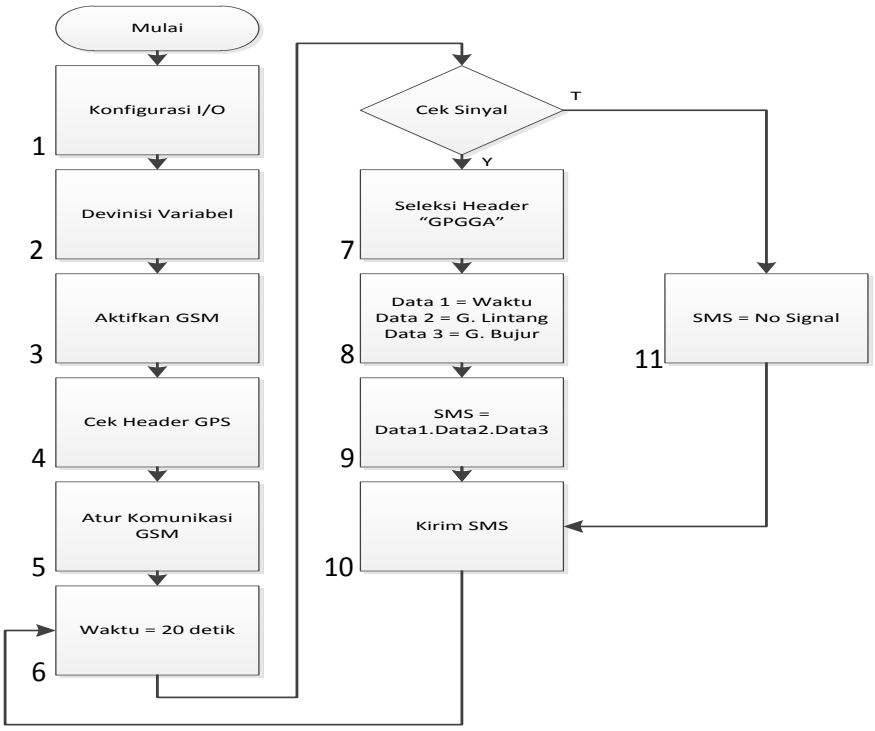


Gambar 3.8 *Diagram Blok Flash Program*

3.3.2 Flowchart Program Sistem Pada Kendaraan Patroli

Menentukan proses-proses yang akan dilakukan dalam perancangan perangkat lunak, merupakan hal yang sangat penting agar sistem dapat bekerja

dengan baik. Berikut adalah diagram alir program utama mikrokontroler pada bagian pengirim dapat dilihat pada gambar 3.9.



Gambar 3.9 *Flowchart program pengirim data pada kendaraan patroli*

Dari gambar 3.9 dapat dijelaskan langkah-langkah dari program pengiriman data pada kendaraan patroli, yang antara lain sebagai berikut:

1. Konfigurasi I/O dan Devinisi Variabel

   Proses pertama yang akan dilakukan mikrokontroler yaitu inisialisasi variabel, setiap variabel akan didefinisikan baik berupa konstanta atau definisi pin-pin yang akan menjadi input dan output.
2. Setelah konfigurasi I/O dan devinisi variabel maka mikrokontroler memberikan perintah untuk mengaktifkan modul GSM.
3. Cek Header GPS

   Header yang di gunakan adalah header $GPRMC.
4. Atur komunikasi GSM

   Pengaturan GSM dilakukan agar modul GSM bisa mengirim dan menerima data berupa sms.

5. Waktu = 20 detik

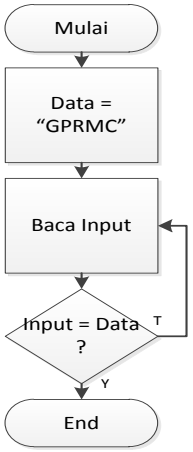
   Waktu disini bersifat *flexible*, user bisa menginputkan berapa detik atau menit waktu yang diinginkan untuk pengiriman data sms. Untuk perancangan alat yang dibuat digunakan waktu 20 detik.

6. Cek sinyal

   Untuk memastikan apakah ada sinyal atau tidak.

7. Seleksi header $GPRMC

   Data header yang diseleksi dari GPS yaitu $GPRMC adalah posisi geografis dari garis lintang atau garis bujur. Flowchart data $GPRMC dapat di lihat pada gambar 3.10.



Gambar 3.10 *Flowchart header GPRMC*

8. Data 1 = Waktu, Data 2 = G. Lintang, Data 3 = G. Bujur adalah sebuah variabel penyimpanan data sementara.

9. SMS = Data1.Data2.Data3

   Isi dari data1.data2.data3 siap untuk di kirim secara bersamaan ke modul GSM penerima.

10. Kirim SMS

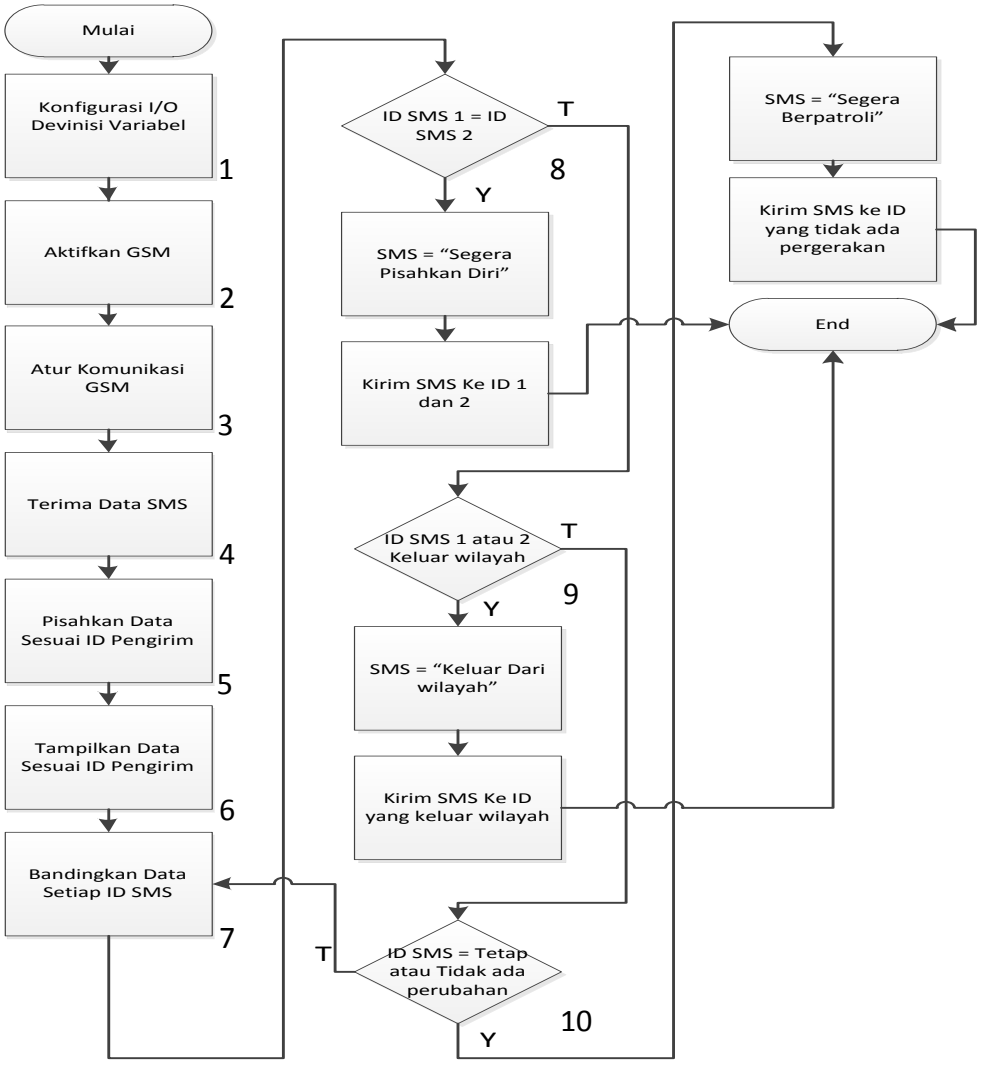
    Setelah siap maka isi sms data1.data2.data3 di kirimkan.

11. SMS = No Signal

    Jika pada proses 6 tidak ada sinyal maka kirimkan data ke pusat informasi berupa sms yang berisi tulisan tidak ada sinyal, jika ada sinyal maka teruskan ke proses 7, 8, 9 dan 10.

3.3.3 Flowchart Program Sistem Pada Operator

Flowchart program sistem penerima data dapat di lihat pada gambar 3.11.



Gambar 3.11 *Flowchart program penerima data pada operator*

Dari gambar 3.11 dapat dijelaskan langkah-langkah dari program penerima data pada operator, yang antara lain sebagai berikut:

1. Konfigurasi I/O dan Devinisi Variabel

   Proses pertama yang akan dilakukan mikrokontroler yaitu inisialisasi variabel, setiap variabel akan di definisikan baik berupa konstanta atau definisi pin-pin yang akan menjadi input dan output.
2. Setelah konfigurasi I/O dan devinisi variabel maka mikrokontroler memberikan perintah untuk mengaktifkan modul GSM.
3. Atur komunikasi GSM

   Pengaturan GSM dilakukan agar modul GSM bisa mengirim dan menerima data berupa sms.
4. Terima data SMS

   Isi dari data sms berupa teks yang berupa informasi garis lintang dan garis bujur dari header GPS.
5. Pisahkan data sesuai ID pengirim

   Pemisahan data dimaksudkan untuk mengetahui letak koordinat setiap kendarran patroli.
6. Tampilkan data sesuai dengan ID pengirim

   Setelah pemisahan data sesuai ID maka data di tampilkan sesuai ID masing-masing.
7. Bandingkan data setiap ID SMS

   Setelah data di terima, setiap data id sms dibandingkan untuk mengelahui pelanggaran yang terjadi.
8. ID SMS 1 = ID SMS 2

   Jika kedua data ID SMS sama maka segera kirimkan perintah kepada setiap ID, sms berupa teks yang berisi peringatan untuk segera memisahkan diri. Jika isi ID SMS berbeda maka lakukan proses 9.
9. ID SMS 1 atau 2 keluar wilayah

   Jika ada ID SMS 1 atau 2 keluar wilayah yang telah ditentukan maka segera kirimkan perintah kepada setiap ID, sms berupa teks yang berisi peringatan untuk segera berpatroli di daerah masing-masing yang telah di tentukan, isi teks adalah keluar dari wilayah. Jika isi ID SMS masih dalam wilayah patroli maka lakukan proses 10.

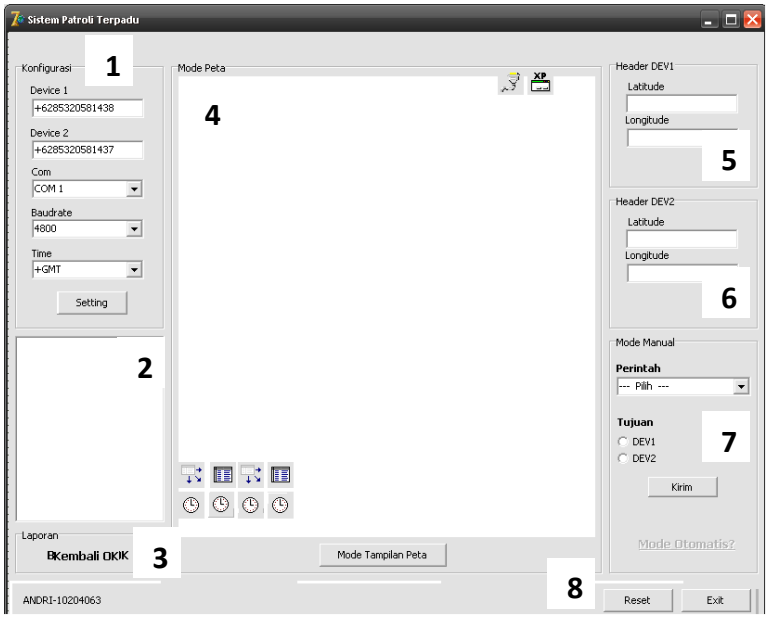
10. ID SMS = tetap atau tidak ada perubahan

    Jika ID SMS = tetap atau tidak ada perubahan maka segera kirimkanperintah kepada setiap ID, sms berupa teks yang berisi peringatan untuk segera berpatroli di daerah masing-masing yang telah di tentukan, isi teks adalah segera berpatroli. Jika isi ID SMS masih berubah datanya maka proses akan kembali ke proses 7.

3.3.4 Perancangan Delphi

Perancangan delphi dibuat untuk mempermudah dalam pengoprasian alat baik dari segi pengaturan ataupun penyimpanan data secara otomatis. Berikut ini adalah gambar perancangan delphi :

3.3.4.1 Perancangan Tampilan Mode Peta



Gambar 3.12 *Tampilan Perancangan Tampilan Mode Peta*

Pada perancangan awal terdiri dari :

1. Group Box konfigurasi terdiri dari dua edit, 3 combo box dan satu button. Edit digunakan untuk menyimpan nomor alat, combo box digunakan untuk memilih com, *baudrate* dan pemilihan waktu sedangkan tombol *setting* digunakan untuk mengaktifkan semua konfigurasi yang telah di konfigurasi.

2. Memo digunakan untuk menyimpan data SMS dari dev satu atau dua sebelum di pisahkan ke data base masing-masing perangkat.
3. Group box laporan berisi laporan tiga kondisi alat yaitu, patroli ok, berpencar ok dan kembali ok.
4. Group box mode peta terdiri dari web browser untuk menampilkan peta yang diakses dari *google maps*.

Untuk mengakses *Google maps* melalui delphi dapat menggunakan *Google maps API*, sintak yang digunakan adalah:

```
'<html> '+
'<head> '+
'<meta name="viewport" content="initial-scale=1.0, user-scalable=yes" /> '+
'<script type="text/javascript"
src="http://maps.google.com/maps/api/js?sensor=true"></script> '+
'<script type="text/javascript"> '+
"+
"+
'  var geocoder; '+
'  var map;  '+
'  var trafficLayer;'+
'  var bikeLayer;'+
'  var markersArray = [];'+
"+
"+
'  function initialize() { '+
'    geocoder = new google.maps.Geocoder();'+
'    var latlng = new google.maps.LatLng(40.714776,-74.019213); '+
'    var myOptions = { '+
'      zoom: 13, '+
'      center: latlng, '+
'      mapTypeId: google.maps.MapTypeId.ROADMAP '+
'    }; '+
'    map = new
google.maps.Map(document.getElementById("map_canvas"), myOptions); '+
'    trafficLayer = new google.maps.TrafficLayer();'+
'    bikeLayer = new google.maps.BicyclingLayer();'+
'  } '+
"+
"+
'  function codeAddress(address) { '+
'    if (geocoder) {'+
'      geocoder.geocode( { address: address}, function(results, status) { '+
'        if (status == google.maps.GeocoderStatus.OK) {'+
'          map.setCenter(results[0].geometry.location);'+
```

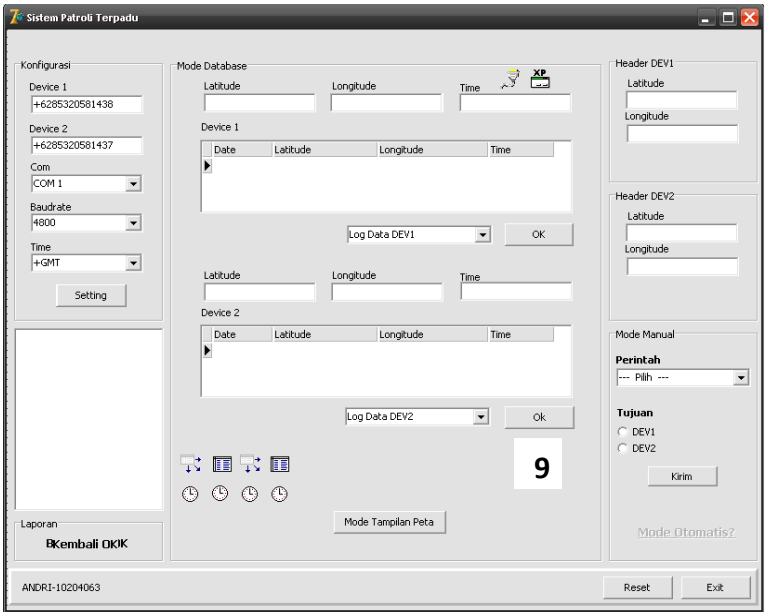
```
'                                          PutMarker(results[0].geometry.location.lat(),
results[0].geometry.location.lng(),
results[0].geometry.location.lat()+","+results[0].geometry.location.lng());'+
//'          var marker = new google.maps.Marker({'+
//'              map: map,'+
//'              position: results[0].geometry.location'+
//'          });'+

'        } else {'+
'          alert("Geocode was not successful for the following reason: " + status);'+
'        }'+
'      });'+
'    }'+
'  }'+
"+
"+
'  function GotoLatLng(Lat, Lang) { '+
'    var latlng = new google.maps.LatLng(Lat,Lang);'+
'    map.setCenter(latlng);'+
'    PutMarker(Lat, Lang, Lat+","+Lang);'+
'  }'+
"+
"+
'function ClearMarkers() {  '+
'  if (markersArray) {        '+
'     for (i in markersArray) {  '+
'       markersArray[i].setMap(null); '+
'     } '+
'  } '+
'}  '+
"+
'  function PutMarker(Lat, Lang, Msg) { '+
'    var latlng = new google.maps.LatLng(Lat,Lang);'+
'    var marker = new google.maps.Marker({'+
'       position: latlng, '+
'       map: map,'+
'       title: Msg+" ("+Lat+","+Lang+")"'+
'  });'+
' markersArray.push(marker); '+
'  }'+
"+
"+
'  function TrafficOn()   { trafficLayer.setMap(map); }'+
"+
'  function TrafficOff()  { trafficLayer.setMap(null); }'+
"+"+
'  function BicyclingOn() { bikeLayer.setMap(map); }'+
"+
```

```
'  function BicyclingOff(){ bikeLayer.setMap(null);}'+
"+
'  function StreetViewOn() { map.set("streetViewControl", true); }'+
"+
'  function StreetViewOff() { map.set("streetViewControl", false); }'+
"+
"+'</script> '+
'</head> '+
'<body onload="initialize()"> '+
'<div id="map_canvas" style="width:100%; height:100%"></div> '+
'</body> '+
'</html> ';
```

5. Group box header DEV 1 berisi data garis lintang dan garis bujur langsung dari header GPS.
6. Group box header DEV 2 berisi data garis lintang dan garis bujur langsung dari header GPS.
7. Group box mode manual berisi perintah-perintah yang ditujukan kepada device 1 ataupun device 2, dan tombol kirim sebagai tombol perintah untuk device 1 dan 2.
8. Tombol reset digunakan untuk mengulang semua seting tanpa keluar dari program sedangkan exit adalah keluar dari keseluruhan program.

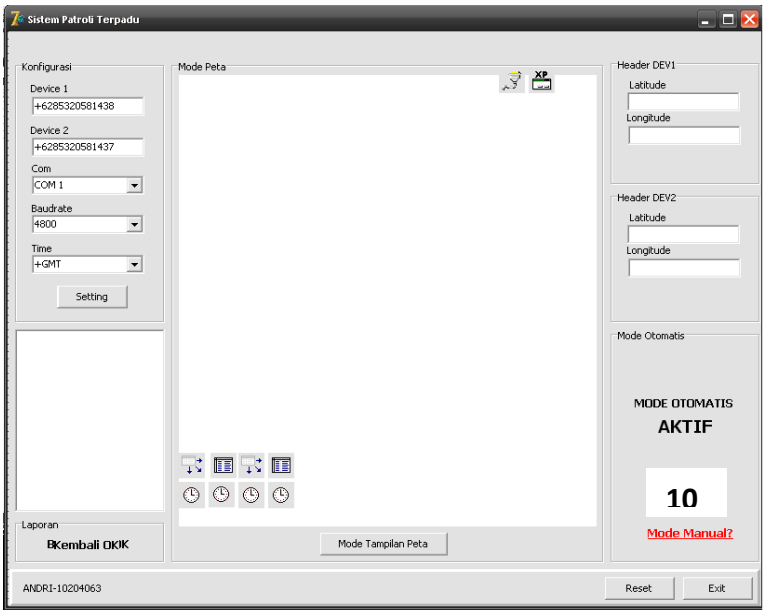
3.3.4.2 Perancangan Tampilan Mode Database



Gambar 3.13 *Tampilan Perancangan Tampilan Mode Database*

9. Group box mode *Database* digunakan untuk menyimpan seluruh data dari device 1 dan 2 yang sudah di konversi kedalam rumus Google maps decimal. Selain itu pada mode database kita dapat melihat seluruh data melalui excel.

3.3.4.3 Perancangan Tampilan Otomatis



Gambar 3.14 *Tampilan Perancangan Mode Otomatis*

10. Group Box mode otomatis berfungsi untuk membuat sistem bekerja dengan otomatis. Sistem ini hanya menyeleksi satu keadaan saja yaitu diam di satu titik terlalu lama.