**INSTITUT TEKNOLOGI BANDUNG**

**PROGRAM STUDI TEKNIK ELEKTRO**

Jalan Ganesha No. 10 Gedung Labtek V Lantai 2 🕿 (022)2508135-36, 🖷 (022)2500940

Bandung 40132

**Dokumentasi Produk Tugas Akhir**

Lembar Sampul Dokumen

|  |  |  |
| --- | --- | --- |
| Judul Dokumen | TUGAS AKHIR TEKNIK ELEKTRO:  Sistem Auto Drive untuk Mobil Pengangkut Barang | |
|  |  | |
| Jenis Dokumen | DESAIN SISTEM | |
|  | Catatan: Dokumen ini dikendalikan penyebarannya oleh Prodi Teknik Elektro ITB | |
| Nomor Dokumen | B300-01-TA1718.01.023 | |
|  |  | |
| Nomor Revisi | 01 | |
|  |  | |
| Nama File |  | |
|  |  | |
| Tanggal Penerbitan | 2 November 2017 | |
|  |  | |
| Unit Penerbit | Prodi Teknik Elektro - ITB | |
|  |  | |
| Jumlah Halaman |  | (termasuk lembar sampul ini) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data Pemeriksaan dan Persetujuan | | | | |
| Ditulis | Nama | M. Teguh Subarkah | Jabatan | Anggota |
| Oleh | Tanggal | 12 Oktober 2017 | Tanda Tangan |  |
|  | Nama  Tanggal  Nama  Tanggal | Tsaqif Alfatan N.  12 Oktober 2017  Moch. Nandradi Toyib  12 Oktober 2017 | Jabatan  Tanda Tangan  Jabatan  Tanda Tangan | Anggota  Anggota |
| Diperiksa | Nama | Ir. Arief Syaichu Rochman, M.EngSc, Ph.D | Jabatan | Dosen Pembimbing |
| Oleh | Tanggal | 12 Oktober 2017 | Tanda Tangan |  |
| Disetujui  Oleh | Nama | Ir. Arief Syaichu Rochman, M.EngSc, Ph.D | Jabatan | Dosen Pembimbing |
| Tanggal | 12 Oktober 2017 | Tanda Tangan |  |
| Nama | Dr. techn. Ary Setijadi Prihatmanto | Jabatan | Dosen Pengusul |
| Tanggal | 12 Oktober 2017 | Tanda Tangan |  |

# DAFTAR ISI

DAFTAR ISI 2

Catatan Sejarah Perbaikan Dokumen 3

Proposal Proyek Pengembangan \_\_\_\_\_\_\_\_\_\_\_\_ 4

1 Pengantar 4

1.1 Ringkasan Isi Dokumen 4

1.2 Tujuan Penulisan dan Aplikasi/Kegunaan Dokumen 4

1.3 Referensi 4

1.4 Daftar Singkatan 4

2 Konsep Sistem 5

2.1 Pilihan Sistem 5

2.2 Analisis 5

2.2.1 Kriteria 5

2.2.2 Analisis konsep 5

2.3 Sistem yang akan dikembangkan 5

2.3.1 Metode pemilihan 5

2.3.2 Konsep sistem terpilih 5

3 Desain Sistem 6

3.1 Pemodelan Fungsional Sistem 6

3.2 Pemodelan Tingkah Laku Sistem 6

4 Lampiran 7

# 

# Catatan Sejarah Perbaikan Dokumen

|  |  |
| --- | --- |
| Versi, Tgl, Oleh | Perbaikan |
|  |  |
|  |  |
|  |  |
|  |  |

# Proposal Proyek Pengembangan Sistem Auto Drive untuk Mobil Pengangkut Barang

# Pengantar

## Ringkasan Isi Dokumen

Isi dengan ringkasan eksekutif.

## Tujuan Penulisan dan Aplikasi/Kegunaan Dokumen

Isi dengan tujuan/maksud penulisan dokumen ini, dan ditujukan kepada siapa.

## Referensi

[1] Caesar Wiratama. 2016. Dasar-dasar Auto Pilot atau Flight Controller. <https://aeroengineering.co.id/2016/05/dasar-dasar-autopilot-atau-flight-controller/> . Diakses tanggal 31 Oktober 2017. Pukul 21.30

[2] Ardupilot Dev Team. 2016. Mission Planner Overview. <http://ardupilot.org/planner/docs/mission-planner-overview.html> . Diakses tanggal 31 Oktober 2017. Pukul 22.00

[3]

## Daftar Singkatan

| Singkatan | Arti |
| --- | --- |
| GPS | Global Positioning System |
| PWM | Pulse Width Modulation |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# 

# Konsep Sistem

## Pilihan Sistem

Sistem *Auto Drive* dapat dikembangkan dengan dua sistem yang berbeda. Sistem pertama pengembangan *Auto Drive* ialah sistem Auto Drive yang dikendalikan dengan modul *Auto Pilot*. Kedua, sistem *Auto Drive* dapat dikembangkan dengan metode *dynamic positioning*. Perbedaan kedua sistem tersebut dapat dijelaskan sebagai berikut.

### Sistem Auto Drive dengan modul Auto Pilot

Modul *Auto Pilot* atau *Flight Controller* merupakan sebuah perangkat yang dapat mengendalikan kendaraan baik itu *drone* ataupun *rover* secara otomatis. Modul *Auto Pilot* dapat terintegrasi dengan beberapa subsistem lainnya seperti GPS, modul telekomunikasi , kompas, dan sensor pelengkap seperti sensor kecepatan angin, sensor tekanan, sensor kelembapan. Auto Pilot akan membaca data-data dari subsistem lain seperti membaca posisi kendaraan dari GPS, membaca *heading* kendaraan dari kompas, dan membaca data-data dari sensor-sensor pelengkap lainnya. Seluruh data pembacaan ini akan diolah oleh prosessor *auto pilot*. Beberapa prosessor auto pilot merupakan prosessor 32bit namun ada juga prosessor 8 bit karena dinilai lebih murah. Pengolahan data oleh prosessor ini akan menghasilkan nilai Pulse Width Modulation (PWM) yang dapat digunakan untuk mengatur *throttle* (pengendalian kecepatan)ataupun *steering* (pengedalian stir) dari kendaraan *rover* ataupun dapat mengatur altitude dari kendaraan *drone*.

* Arsistektur Sistem

Berdasarkan prinsip kerja *Auto Pilot* yang telah dijabarkan di atas, arsitektur sistem *auto drive* yang menggunakan modul pengendalian otomatis dapat dilihat sebagai berikut.

Data Flow

Mikro kontroler

Motor Driver

Steering

Motor Driver

Throttle

Auto Pilot

Sensor Deteksi Objek

GPS

Kompas

Communication module

Ground Station

Power Module

Way Point

User

Display

Power Flow

Gambar 1 – Aristektur Sistem Auto Drive dengan Modul Auto Pilot

Meskipun sistem *Auto Drive* telah menggunakan modul auto pilot, sistem juga masih membutuhkan mikrokontroler. Mikrokontroler berfungsi untuk mengolah data deteksi objek yang didapatkan dari sensor pendeteksi objek. Selain itu, mikrokontroler juga berfungsi untuk mendeteksi adanya keadaan darurat yang menyebabkan salah subsistem mati. Ketika keadaan darurat ini terdeteksi, mikrokontroler akan mengirimkan sinyal ke modul daya dan memutus sumber catu daya dari modul daya. Namun karena sistem pengendalian diatur oleh Auto Pilot, mikrokontroler tidak memerlukan memori dan bit yang lebih besar dibandingkan jika menggunakan konsep *dynamic positioning*.

Mikrokontroller juga akan langsung terhubung ke motor driver yang akan menggerakkan motor untuk *steering* dan *throttle*. Motor throttle akan berputar dengan kecepatan yang sebanding dengan PWM yang diberikan oleh auto pilot. Pergerakkan motor steering akan mempertimbangkan posisi mobil melalui data GPS yang dikirimkan ke auto pilot lalu mikrokontroler dan juga mempertimbangkan hasil deteksi objek dari sensor objek.

Pada implementasinya, mobil akan menerima data berupa way-point perjalanan dari pengguna dari ground station. Mobil dan ground station akan berkomunikasi dengan modul telekomunikasi yang memungkinkan ground station mengirimkan data waypoint ataupun memungkinkan mobil untuk dapat mengirimkan data pemantauan kondisi mobil ke ground station.

* Interaksi dengan pengguna (*user interaction*)

Seperti yang telah diungkapkan sebelumnya, terdapat interaksi antara sistem auto drive dengan pengguna. Pertama, interaksi tersebut berupa pengiriman *way-point* atau titik-titik yang akan dituju mobil dalam sebuah misi perjalanan. Kedua, pengguna dapat mengamati atau memantau keadaan mobil baik posisi, kecepatan , status emergency dari mobil.

USER

User Interface

Auto Pilot

Gambar 2 – Diagram interaksi sistem dengan pengguna

Subsistem interaksi dengan pengguna erat kaitannya dengan penggunaan modul auto pilot atau tidak karena baik way-point dan juga pemantauan posisi dan kecepatan mobil berhubungan dengan sistem kerja auto pilot. Penggunaan sistem auto pilot dapat memudahkan sistem antarmuka pengguna yakni dengan menggunakan *software* Mission Planner (MP). Diagram use case dari penggunaan software MP sebagai *user interface* sistem dapat ditunjukkan sebagai berikut.

User

Mengaktifkan software

Menggunakan software

mematikan software

Start & stop

Input way point

Generate route

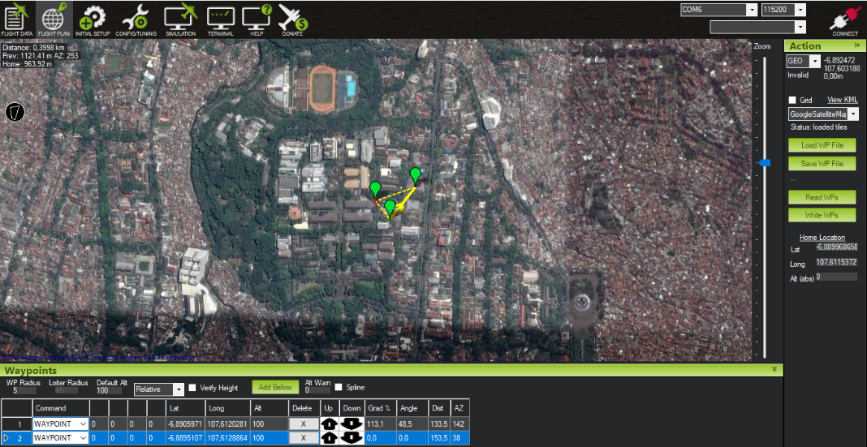
Pemantauan mobil

Gambar 3 – Diagram use case oleh pengguna

MP merupakan software yang terintegrasi dengan auto pilot dan bisa digunakan untuk beberapa jenis sistem auto pilot seperti ardupilot ataupun pixhack. MP menyediakan fitur *flight plan* dan juga *flight data*.

Fitur Flight Plan

*Flight plan* merupakan fitur yang memungkinkan pengguna memasukkan beberapa *way-point* mulai dari titik awal sampai titik akhir yang dituju. Berikut tampilan fitur *flight plan* dari software Mission Planner



Gambar 4 Tampilan fitur *flight plan* dari software Mission Planner

Untuk memasukkan way-point, pengguna dapat melakukan langkah-langkah sebagai berikut.

* Memilih home point di map dengan cara mengarahkan pointer ke lokasi home yang dituju, kemudian klik kanan pada mouse , dan pilih “Set Home Here”
* Menambahkan way-point dengan cara mengeklik kanan pada lokasi way-point yang diinginkan.
* Menghapus way-point dengan cara mengeklik tombol silang pada baris way-point yang ingin dihilangkan
* Menjalankan misi dengan cara memilih “Write WPs” di panel sebelah kanan software. Pasitkan MP sudah terkoneksi dengan perangkat auto pilot sebelumnya.

Fitur Flight Data

Fitur *flight data* memungkinkan pengguna untuk dapat memantau keadaan mobil dari Ground Station. Pertama menu flight data memiliki tampilan map yang dapat menunjukkan lokasi mobil secara *real time*. Posisi mobil akan ditunjukkan oleh ikon rover ataupun ikon drone di map. Kemudian di panel sebelah kiri, MP menunjukkan sinyal GPS dan juga kondisi roll, pitch , dan yaw dari kendaraan. Selanjutnya, di panel sebelah kiri bawah juga terdapat besaran-besaran yang harus dipantau antara lain seperti *altitude* , *Ground Speed* , *Vertical Speed*, *Distance to WP, Distance to MAV ,*dan juga sebagainya. Berikut tampilan fitur flight data dari software MP



Gambar 5 – Tampilan fitur flight data pada software Mission Planner

Selain itu, Software Mission Planner juga memiliki beberapa menu lain sebagai berikut.

* Initial Setup berfungsi untuk mengatur konfigurasi awal dari APM.
* Config/Tuningberfungsi untuk mengkonfigurasi dan mengatur parameter dari APM
* Simulation berfungsi untuk mensimulasikan jalur yang dipakai oleh APM.
* Terminal berfungsi untuk memberikan perintah kepada APM melalui command line interface.
* Algoritma sistem

Algoritma sistem dapat ditinjau dari beberapa subsistem keseluruhan yang terdapat di arsitektur sistem antara lain algoritma auto pilot, algoritma pergerakkan motor, algoritma deteksi objek , algoritma manuver , dan algoritma pematian otomatis.

Keseluruhan subsistem akan diatur oleh mikrokontroller. Algoritma mikrokontroller sebagai control unit dapat dilihat dalam flowchart berikut.

Algoritma dimulai dengan menginisiasi sistem. Inisiasi ini berupa persiapan modul telekomonikasi apakah sudah siap mengirim atau menerima data , menyalakan modul power untuk seluruh sistem, dan juga menyiapkan user interface. Setelah sistem sudah siap maka proses pengambilan data dan pengolahan data akan dilakukan. Pertama, sensor deteksi objek akan merekonstruksi jarak objek-objek di sekitar mobil. Data rekonstruksi ini akan dikirimkan ke mikrokontroler dan akan diolah pada mikrokontroller. Kemudian modul auto pilot juga akan menerima data way-point yang dimasukkan oleh pengguna melalui *software* Mission Planner. Auto pilot akan mengolah input way-point tersebut menjadi nilai PWM yang akan mengendalikan motor untuk pengendalian steering dan motor pengendalian throttle. Data PWM tersebut akan langsung dikirimkan ke mikrokontroller melalui koneksi kabel USB. Selanjutnya, mikrokontroller terlebih dahulu mengolah data deteksi objek dari sensor objek. Apabila terdapat objek yang dideteksi maka nilai batas jalan terhadap mobil akan berubah. Perubahan nilai ini akan dianalisis oleh mikrokontroller untuk mendapatkan gerakan manuver yang sesuai dari mobil. Selanjutnya, mikrokontroller mengolah data PWM kecepatan dari auto pilot yang akan menjadi input pergerakkan motor. Output PWM steering dari auto pilot akan dikombinasikan dengan gerakkan manuver (apabila ditemukan perubahan batas jalan) untuk menghasilkan kendali steering dari mobil. Terakhir, auto pilot melakukan verifikasi apakah way-point yang diberikan oleh user sudah tercapai atau tidak. Apabila tidak, sistem akan melakukan siklus yang sama seperti sebelumnya.

START

Inisialisasi sistem

Menerima data dari sensor deteksi objek dan modul auto pilot

Way-point akhir sudah dicapai ?

END

Ya

Tidak

Batas jalan terhadap mobil tidak benar ?

Manuver menghindar

Ya

Proses Auto Pilot

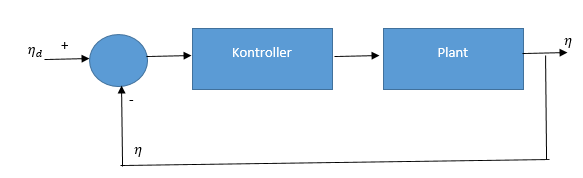
Kendali steering

Kendali throttle

Tidak

### Sistem Auto Drive dengan Dynamic Positioning

*Dynamic Positioning* merupakan sebuah sistem yang biasa digunakan dalam kapal untuk selalu berada pada jalur yang dimau ketika sedang dalam berlayar . Sistem ini mengendalikan posisi dan arah hadap kapal dengan menggunakan *propeller* dan *thruster* . Sistem ini memiliki masukkan posisi kapal (koordinat X dan Y ) dan arah hadap kapal . Kemudian dikendalikan dengan kontroller dan nantinya keluaran posisi dan arah hadap kapal akan di*-feedback-*an terhadap input . Berikut adalah diagram blok dynamic positioning



Dengan

= posisi dan arah yang diinginkan ( x, y, φ)

= posisi dan arah aktual( x, y, φ)

Plant yang dimaksud adalah persamaan dinamis kapal yang memperhatikan faktor hidrodinamis . Persamaan yang dimaksud adalah sebagai berikut



Dimana,

M = Massa kapal

= Akselerasi

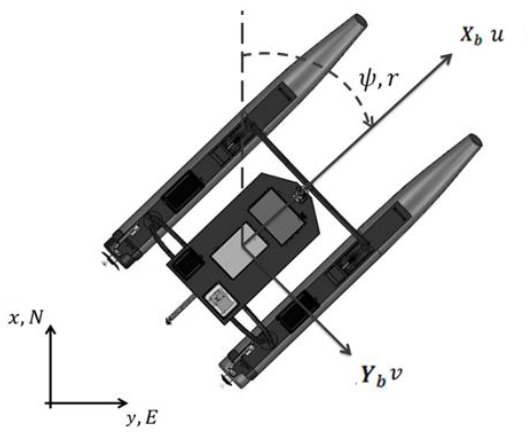
C = Gaya Koriolis

v = kecepatan kapal

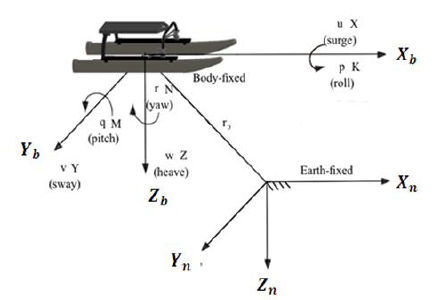
D = Gaya Drag

τ = gaya

Dalam pergerakanya kapal terdapat memiliki 3 DOF , yaitu *surge* , *sway* dan *yaw* seperti diilustrasikan pada gambar di bawah ini



Dalam mendefinisikan koordinat frame juga harus diperhatikan . Pada kendaraan contohnya kapal , terdapat *body-fixed frame* dan *earth-fixed frame*, agar lebih jelas dapat dilihat pada gambar di bawah ini



Kemudian jika dianalasisi , maka pergerakan mobil bisa dikatakan sama dengan pergerakan kapal , karena mobil juga bergerak dalam 3 DOF , yaitu *surge* , *sway* dan *yaw* . Kemudian hal yang dapat membuatnya sama lagi adalah jika kapal dikendalikan oleh propeller dan thruster dapat disamakan dengan mobil dikendalkan oleh roda dan steering.

Kemudian persamaan dinamis dari kapal dapat diadopsi ke dinamika mobil . Jika pada persamaan kapal adalah



Maka karena pada mobil tidak terdapat gaya koriolis dan drag , persamaanya dapat ditulis menjadi

Seperti diketahui bahwa dynamic positioning memerlukan masukkan berupa posisi dan arah , maka dalam implementasinya diperlukan sebuah alat pemetaan untuk mengetahui posisi dan kompas untuk mengetahui arah hadap mobil. Kemudian untuk melakukan pemrosesan kendalinya diperlukan sebuah alat komputasi . Berikut ini adalah arsitektur sistem yang digunakan

* Arsitektur Sistem

Mikro kontroler

Motor Driver

Steering

Motor Driver

Throttle

Sensor Deteksi Objek

GPS

Kompas

Communication module

Ground Station

Power Module

Way Point

User

Display

Power Flow

Mikrokontroller diperlukan untuk memproses sistem dynamic positioning tersebut yang nantinya akan mengirimkan sinyal PWM kepada motor driver untuk menggerakan *throttle* dan *steering*. Selain itu mikrokontroller digunakan sebagai pemrosesan untuk deteksi objek dan memberikan sinyal PWM yang diperlukan jika terdapat *obstacle* baik terhadap motor throttle ataupun steering . Kemudian mikrokontroller berfungsi juga untuk sistem pematian otomatis jika terdapat salah satu fitur tidak berfungsi .

Dalam pemakaianya posisi dan arah hadap diberikan oleh operator pengirim barang , sehingga perlu sebuah antarmuka yang dapat menghubungkan dengan alat komputasi tersebut. Agar bisa mendapatkan antarmuka tersebut , diperlukan sebuah perangkat lunak yang bisa membuat antarmuka dan menghubungkan data dari komputer terhadap alat komputasi yang digunakan .

Karena terdapat fitur pendeteksi objek , maka mikrokontroler berfungsi untuk mengolah data deteksi objek yang didapatkan dari sensor pendeteksi objek. Selain itu, mikrokontroler juga berfungsi untuk mendeteksi adanya keadaan darurat yang menyebabkan salah subsistem mati. Ketika keadaan darurat ini terdeteksi, mikrokontroler akan mengirimkan sinyal ke modul daya dan memutus sumber catu daya dari modul daya.

Pada implementasinya, mobil akan menerima data berupa way-point perjalanan dari pengguna dari ground station. Mobil dan ground station akan berkomunikasi dengan modul telekomunikasi yang memungkinkan ground station mengirimkan data waypoint ataupun memungkinkan mobil untuk dapat mengirimkan data pemantauan kondisi mobil ke ground station.

* Antarmuka

Seperti yang telah diungkapkan sebelumnya, terdapat interaksi antara sistem auto drive dengan pengguna. Pertama, interaksi tersebut berupa pengiriman way-point atau titik-titik yang akan dituju mobil dalam sebuah misi perjalanan. Kedua, pengguna dapat mengamati atau memantau keadaan mobil baik posisi, kecepatan , status emergency dari mobil.

USER

User Interface

Mikrokontroller

Pada penggunaanya operator akan memasukkan way point ,memantau sistem , dan mematikan sistem dalam keadaan darurat. Semua data diproses oleh mikrokonroller , agar data-data tersebut dapat dipantau dengan mudah oleh operator maka perlu adanya software antarmuka . Diagram use case dari penggunaan software antamuka sebagai user interface sistem dapat ditunjukkan sebagai berikut.

User

Mengaktifkan software

Menggunakan software

mematikan software

Start & stop

Input way point

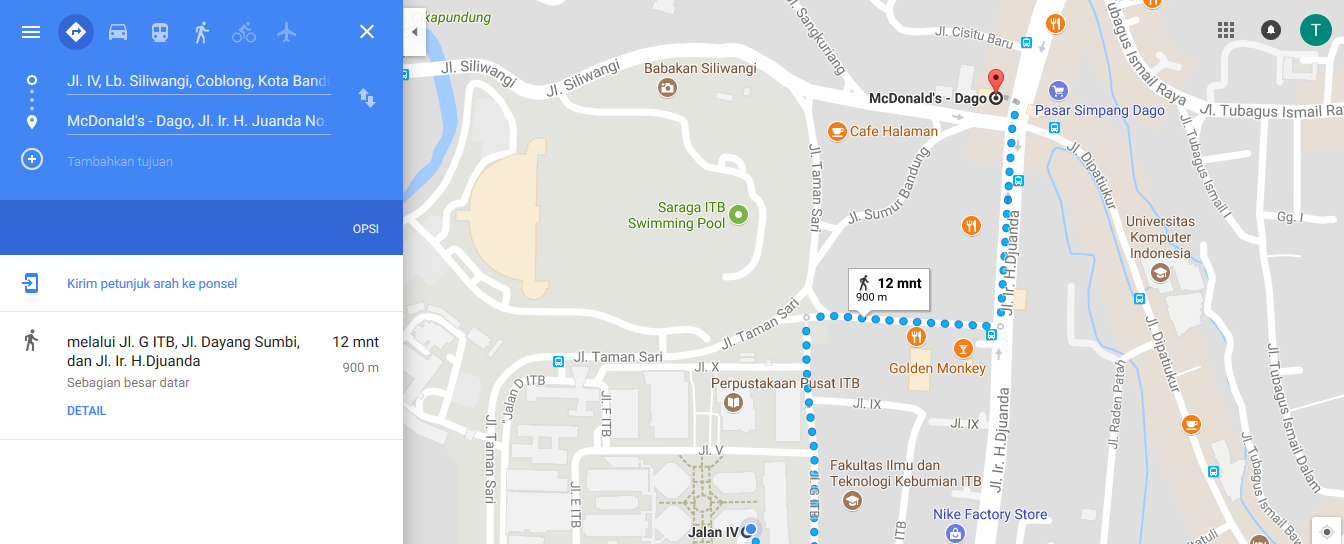
Generate route

Pemantauan mobil

Ketika gps sudah terkoneksi , maka pada tampilan akan diperlihatkan peta dengan titik briu yang merupakan titik “home” seperti di bawah ini

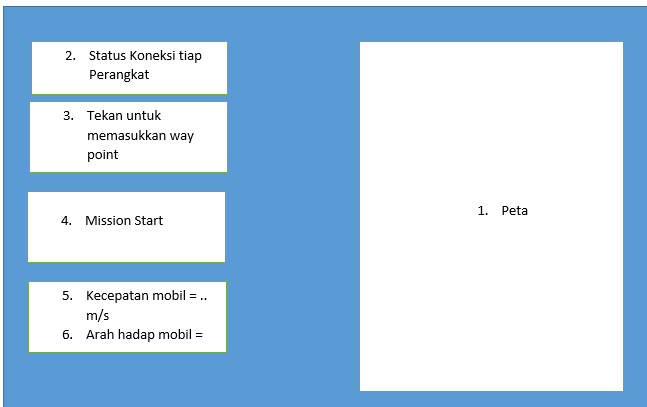


Kemudian pada tampilan akan ada tombol arahan untuk memberikan *way point* . Setelah operator meng-klik tombol tersebut, maka pada tampilan akan diarahkan langsung untuk memasukkan tujuan , ilustrasinya seperti di bawah ini



Setelah itu opsi “mission start” akan bisa ditekan karena way point sudah bisa diberikan dan tulisan tombol akan berubah menjadi mission end. Kemudian mobil akan berjalan ke tujuan tersebut.

Dalam perjalananya , mobil dapat dipantau pada sistem antarmuka ini dengan diberikan beberapa data yaitu , lokasi mobil saat itu (dari peta) , kecepatan mobil , dan arah hadap mobil .

Ilustrasi sistem antarmuka yang nantinya akan dibuat adalah sebagai berikut

* Algoritma Sistem

Algoritma sistem dapat ditinjau dari beberapa subsistem keseluruhan yang terdapat di arsitektur sistem antara lain algortima penerimaan way point, algoritma pergerakkan motor, algoritma deteksi objek , algoritma manuver , dan algoritma pematian otomatis.

Keseluruhan subsistem akan diatur oleh mikrokontroller. Algoritma mikrokontroller sebagai control unit dapat dilihat dalam flowchart berikut

Algoritma dimulai dengan menginisiasi sistem. Inisiasi ini berupa persiapan modul telekomonikasi apakah sudah siap mengirim atau menerima data , menyalakan modul power untuk seluruh sistem, dan juga menyiapkan user interface. Setelah sistem sudah siap maka proses pengambilan data dan pengolahan data akan dilakukan. Pertama, sensor deteksi objek akan merekonstruksi jarak objek-objek di sekitar mobil. Data rekonstruksi ini akan dikirimkan ke mikrokontroler dan akan diolah pada mikrokontroller. Kemudian mikrokontroller akan menerima data way-point yang dimasukkan oleh pengguna melalui software yang dibuat. Mikroontroller akan mengolah data way point dengan dynamic positioning untuk mengolah data pwm throttle dan steering. Data PWM tersebut akan langsung dikirimkan ke mikrokontroller melalui koneksi kabel USB. Selanjutnya, mikrokontroller terlebih dahulu mengolah data deteksi objek dari sensor objek. Apabila terdapat objek yang dideteksi maka nilai batas jalan terhadap mobil akan berubah. Perubahan nilai ini akan dianalisis oleh mikrokontroller untuk mendapatkan gerakan manuver yang sesuai dari mobil. Selanjutnya, mikrokontroller mengolah data PWM kecepatan dari auto pilot yang akan menjadi input pergerakkan motor. Output PWM steering dari mikrokontroller akan dikombinasikan dengan gerakkan manuver (apabila ditemukan perubahan batas jalan) untuk menghasilkan kendali steering dari mobil. Terakhir, mikrokontroller melakukan verifikasi apakah way-point yang diberikan oleh user sudah tercapai atau tidak. Apabila tidak, sistem akan melakukan siklus yang sama seperti sebelumnya

START

Inisialisasi sistem

Menerima data dari sensor deteksi objek dan masukkan way point

Way-point akhir sudah dicapai ?

END

Ya

Tidak

Batas jalan terhadap mobil tidak benar ?

Manuver menghindar

Ya

Proses di Mikrokontroller

Kendali steering

Kendali throttle

Tidak

Jabarkan minimal dua (2) konsep sistem yang akan dikembangkan. Penjabaran meliputi:

* Arsitektur utama sistem
* Interaksi dengan pengguna (*user interaction*)
* Algoritma utama yang akan digunakan
* Modul atau sub-blok yang memenuhi fungsi dan spesifikasi

## Analisis

### Kriteria

Kriteria pemilihan sistem *auto drive* dapat melihat dari toleransi keberhasilan secara dari produk yang telah dicantumkan pada dokumen B200 TA.171801.023. Beberapa spesifikasi produk ditinjau dari subsistem atau komponen yang ada di produk *auto drive* antara lain sebagai berikut.

|  |  |
| --- | --- |
| **Fitur / Fungsi** | **Toleransi keberhasilan** |
| **Sistem pengendali kecepatan** | Mobil mampu berjalan dengan kecepatan maksimal 30 km / jam dengan kondsi mobil tetap berada pada jalur yang diinginkan serta mekanik mobil masih berada dalam kondisi yang baik |
| **Sistem Pengendali Kemudi** | Stir dapat bergerak dengan sudut maksimum 300 dengan respons waktu tercepat selama 1s |
| **Sistem deteksi objek** | Mobil mampu mendeteksi objek halangan di depan, belakang, dan samping mobil dan mampu merespon dari adanya halangan tersebut dalam 1detik. Objek yang dideteksi berupa manusia dan kendaraan yang berlalu lalang. |
| ***User Interface*** | Mobil dapat berjalan 2 detik setelah diberikan masukkan *way point* . Operator dapat mengamati lokasi mobil dengan maksimal waktu tunda sebesar 2 detik dengan ketelitian sistem navigasi sebesar 3 meter. |
| **Sistem Navigasi** | Sistem mampu memantau mobil dengan keakuratan lokasi *real time* minimal 3 meter |
| **Sistem Pematian Otomatis** | Sistem memberikan respon berupa status kendaraan apabila mati karena keadaan darurat dengan respon waktu deteksi sebesar 3 detik |

Maka dari itu, kriteria pemilihan sistem dalam perancangan produk *auto drive* dapat dibagi atas beberapa poin berikut.

* Nilai kecepatan maksimum yang dapat diberikan oleh masing-masing sistem
* Nilai sudut stir maksimum yang dapat diberikan
* Nilai respon waktu terhadap kontrol kecepatan
* Nilai respon waktu terhadap kontrol stir
* Toleransi kesalahan kontrol kecepatan
* Toleransi kesalahan kontrol stir
* Nilai respon waktu terhadap pemantauan kondisi mobil
* Keakuratan sistem pemantauan mobil
* Respon waktu keadaan darurat

Selain dari toleransi keberhasilan, kriteria pemilihan sistem juga dapat dilihat dari beberapa aspek penting lainnya antara lain sebagai berikut

* Kapasitas data yang diperlukan oleh mikrokontroller
* Aspek ekonomi dari perancangan produk untuk tiap sistem yang akan dipilih
* Jumlah maksimal way-point yang dapat diberikan

Tentukan kriteria yang akan digunakan untuk menganalisis konsep sistem.

### Analisis konsep

Analisis setiap konsep sistem yang diusulkan berdasarkan kriteria yang telah ditentukan sebelumnya.

## Sistem yang akan dikembangkan

### Metode pemilihan

Tentukan dan jelaskan metode pemilihan konsep sistem yang akan digunakan. Misal:

1. Memberikan bobot nilai tertentu untuk setiap kriteria.
2. Setiap konsep dinilai berdasarkan hasil analisis.
3. Konsep dengan nilai total tertinggi menjadi konsep yang terpilih

### Konsep sistem terpilih

Nyatakan dengan jelas dan tegas konsep sistem yang terpilih.

# Desain Sistem

Proses desain dilakukan secara iteratif dan bertahap. Metode dekomposisi yang digunakan adalah *top-down*, yaitu dari diagram blok level tinggi dipecah sampai diagram blok terendah. Diagram blok *Hardware* berakhir pada rangkaian. Diagram blok *Software* berakhir dengan *function call* terendah berikut model perilakunya.

Penentuan sub-blok dari diagram sistem dilakukan dengan mempertimbangkan alternatif desain dan melakukan *trade-off* untuk pilihan-pilihan yang ada. Untuk membantu dalam menentukan pilihan, dapat dilakukan simulasi, *prototyping*, atau pengujian.

Dalam membuat desain sistem, hal yang penting untuk diperhatikan adalah *interfacing*. Bagaimana metode komunikasi antar sub-blok, format data, dan sebagainya.

Semua kegiatan yang dilakukan dalam proses desain harus tercatat di dalam dokumen ini.

## Pemodelan Fungsional Sistem

Pemodelan dilakukan dengan cara pemecahan/dekomposisi sistem berdasarkan diagram blok dari konsep sistem yang terpilih. Pemodelan menghasilkan beberapa tingkat sub-sistem. Setiap sub-sistem memiliki deskripsi berupa:

* Masukan,
* Luaran, dan
* Fungsinya.

## Pemodelan Tingkah Laku Sistem

Berisi deskripsi sistem berdasarkan *behavior*(tingkah laku/aturan) sistem dan sub-sistemnya. Deskripsi yang dimaksud antara lain berupa:

* *state diagram*
* *flowchart*
* *data flow diagram*
* *entity relationship diagram*

# Lampiran

Lampirkan dokumen pendukung yang terkait, misalnya dokumen standard yang terkait produk ini serta dokumen rujukan biaya.