

SIMULASI PENGONTROLAN PINTU HALTE BUS WAY MENGUNAKAN SENSOR LED INFRA RED DAN INTERFACE PORT PARALLEL

Hari Antoni Musril

Institut Agama Islam Negeri (IAIN) Bukittinggi.

Jl. Sjech M. Djamil Djambek Bukittinggi.

Email: kum_ayik.com

ABSTRACT

Along with growth of computer technology, at the moment computer not only exploited for the application of having the character of public like reporting processes, data, but also exploited for the operation of equipment and system controller. In this research, writer try the application of personal computer as system of door control at a halte of bus way by using one tide sensor the LED infrared placed in way in at special band of bus way and one tide sensor the LED infrared again placed in constrictor wall. By exploiting electronics component as controller of door of halte of bus way supported with the language programming Borland Delphi 7.0, writer have earned to design a prototype which can give the amenity of management a halte of bus way.

Key words: sensor LED infra red, interface port parallel

PENDAHULUAN

Komputer (*Personal Computer*) mengalami perkembangan yang sangat signifikan dewasa ini. *Personal Computer* (PC) saat ini dapat dimanfaatkan untuk mempermudah mengontrol dan mengatur peralatan maupun sistem yang berada di luar sistem komputer. Banyak kegiatan yang dapat dilakukan oleh suatu sistem pengontrolan dengan memanfaatkan PC, sehingga tidak begitu besar lagi memanfaatkan tenaga manusia. Salah satunya sistem pengontrolan yang dapat diaplikasikan pada suatu bangunan, dimana adanya pengontrolan berbasis PC yang diterapkan pada bangunan tersebut dapat mempermudah kerja orang yang menggunakan bangunan tersebut.

Bangunan yang memanfaatkan kecerdasan PC banyak diaplikasikan dalam sistem pengontrolan pada penggerakan kegiatan yang sering dilakukan sehari-hari. Salah satu contohnya adalah membuka suatu pintu bangunan yang memanfaatkan *sensor* sebagai pendeteksi kegiatan yang akan dilakukan oleh pintu otomatis tersebut.

Sistem pengontrolan berbasis PC juga dapat diterapkan dalam dunia transportasi. Penerapan tersebut bertujuan untuk meningkatkan efektifitas dan keamanan dalam operasional moda transportasi tersebut. Semakin berkembang jenis transportasi yang ada, maka harus juga didukung dengan kemajuan teknologi untuk mengoptimalkan fungsi dari transportasi tersebut. Perkembangan transportasi darat di Indonesia dewasa ini harus memperhatikan aspek ketepatan waktu, dan keselamatan pengguna jasa serta kenyamanan. Hal ini ditandai dengan telah diterapkannya suatu sistem transportasi darat oleh Provinsi DKI Jakarta yang berusaha memenuhi aspek-aspek di atas, yaitu dengan mengadakan *bus way* sebagai salah satu alternatif angkutan darat.

Bus way dapat dikatakan sebagai angkutan jalan raya yang eksklusif, karena *bus way* memiliki jalur khusus dan halte sebagai tempat pemberhentian yang juga dirancang khusus. Pada perkembangannya, terjadi banyak masalah dalam operasional *bus way* tersebut. Salah satu masalah yang muncul adalah adanya penumpang yang terjatuh pada saat naik

ataupun turun dari *bus way*. Hal tersebut terjadi salahsatunya karena pintu *bus way* belum tertutup rapat, namun *bus way* telah mulai melaju, sehingga ada penumpang yang berada di bibir pintu bus terjatuh.

Untuk memberikan kenyamanan dan keamanan pengguna jasa *bus way*, maka perlu dirancang suatu sistem pengontrolan pada pintu *halte bus way* tersebut sehingga dapat meminimalisir kecelakaan penumpang pada saat akan naik ataupun turun dari *bus way*. Ini mengingat *halte bus way* yang berada lebih tinggi dari badan jalan.

Rumusan masalah pada tulisan ini adalah : (a) Bagaimana membuat pergerakan pintu *halte bus way* menjadi sempurna ?; (b) Bagaimana rancangan pengontrolan pintu *halte bus way* dengan cara otomatis ?; (c) Bagaimana usaha untuk memberikan rasa keamanan bagi pelanggan *bus way* pada saat akan naik maupun turun dari *bus way*, sehingga dapat meminimalisir kecelakaan penumpang di *halte bus way* pada saat naik maupun turun dari *bus way* ?; (d) Bagaimana penerapan *Personal Computer* (PC) sebagai basis kontrol peralatan luar?.

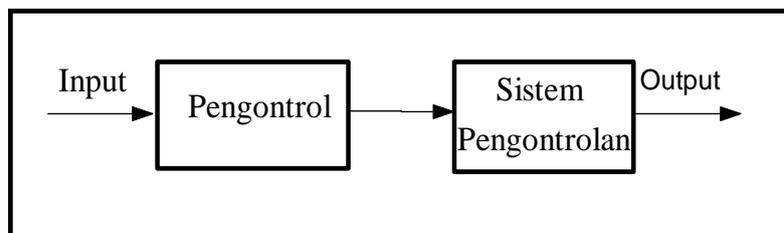
Dari perumusan masalah tersebut di atas, maka dalam tulisan ini akan dibatasi pada permasalahan-permasalahan sebagai berikut : (a) Merancang sensor *led infra* merah untuk membuat proses pengontrolan otomatis pada pintu *halte bus way*; (b) Sistem pengontrolan pintu *halte* ini memanfaatkan motor *stepper* dan (c) Sistem pengontrolan ini menggunakan PC dengan memanfaatkan *interface port parallel* dan bahasa pemrograman delphi.

Sistem Kontrol

Tujuan pengontrolan adalah untuk menciptakan hasil kerja yang optimal, sesuai dengan output yang diharapkan dan memiliki kesalahan sekecil mungkin. Sistem kontrol berdasarkan cara kerjanya dapat dibagi menjadi dua bagian, sistem kontrol *loop* terbuka dan sistem kontrol *loop* tertutup.

Sistem Kontrol Loop Terbuka

Sistem yang keluarannya tidak mempunyai pengaruh terhadap aksi kontrol disebut dengan sistem kontrol *loop* terbuka. Dengan kata lain, sistem kontrol *loop* terbuka keluarannya tidak dapat dipergunakan sebagai perbandingan umpan balik dengan masukan, diagram sistem kontrol *loop* terbuka dapat dilihat pada gambar berikut ini.



Gambar 1 Diagram Sistem Kontrol Loop Terbuka

Dalam suatu sistem kontrol *loop* terbuka, keluaran tidak dapat dibandingkan dengan masukan acuan. Jadi untuk tiap masukan acuan berhubungan dengan kondisi operasi tertentu, sebagai akibat ketepatan dari sistem tergantung kepada kalibrasi.

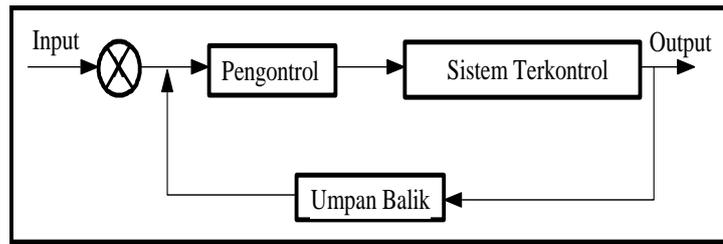
Sistem Kontrol Loop Tertutup

Sistem yang mempertahankan hubungan yang ditentukan antara keluaran dan beberapa masukan acuan, dengan membandingkan mereka dan dengan menggunakan perbedaan sebagai alat kontrol dinamakan sistem kontrol umpan balik yang seringkali disebut sebagai

sistem kontrol *loop* tertutup. Pada sistem kontrol *loop* tertutup, sinyal kesalahan yang bekerja, yaitu perbedaan antara sinyal masukan dan sinyal umpan balik (yang mungkin sinyal keluarannya sendiri atau fungsi dari sinyal keluaran dan turunannya), disajikan ke kontroler sedemikian rupa untuk mengurangi kesalahan dan membawa keluaran sistem ke nilai yang dikehendaki. Istilah kontrol *loop* tertutup selalu berarti penggunaan aksi kontrol umpan balik untuk mengurangi kesalahan sistem.

Suatu kelebihan dari sistem kontrol *loop* tertutup adalah penggunaan umpan balik yang membuat respon sistem relatif kurang peka terhadap gangguan eksternal dan perubahan internal pada parameter sistem. Jadi, mungkin dapat digunakan komponen-komponen yang

relatif kurang teliti dan murah untuk mendapatkan pengontrolan sistem dengan teliti, hal ini tidak mungkin dapat diperoleh pada sistem *loop* terbuka, diagram dari sistem kontrol tertutup dapat dilihat pada gambar di bawah ini.



Gambar 2 Diagram Sistem Kontrol Loop Tertutup

Interface

Jika kita hendak menghubungkan piranti *peripheral* seperti *relay*, motor, *indikator*, *sensor*, pembangkit frekuensi dan lain sebagainya, maka dibutuhkan rangkaian tambahan yang disebut *interface* atau antarmuka. Rangkaian ini bertugas untuk menyesuaikan piranti *peripheral* dengan komputer, karena besarnya tegangan, arus dan daya piranti *peripheral* kebanyakan tidak sesuai dengan yang ada dalam komputer, dan terutama karena kecepatan pengolahannya sangat berbeda dengan komputer, maka besaran-besaran ini harus disesuaikan dengan bantuan *interface*. Pengertian *interface* sendiri adalah rangkaian elektronik yang digunakan untuk menghubungkan antara dua sistem, agar sistem

tersebut bisa berkomunikasi atau proses *handshaking*.

Port Paralel

Salah satu dari *interface* yang standar yang disediakan sistem komputer adalah *port paralel*. Nama lain dari *port paralel* adalah *port printer* karena memang dirancang untuk melayani pencetak paralel. Ada beberapa nama bagi *paralel port*. *Port Paralel* yang bukan di video adapter (*monochrome*) diberi nama LPT1 dan LPT2, masing-masing mempunyai alamat sendiri. Guna memudahkan istilah maka *port paralel* yang di video adapter dinamai LPT0. Agar menyingkat nama *paralel port* selanjutnya disebut LPT. Nama dan nomor register *paralel port* dapat dilihat pada tabel di bawah ini.

Tabel 1 Nama Dan Nomor Register Paralel Port

LPT0	LPT1	LPT2	Sifat	Nama
\$3BC	\$378	\$278	R/W	DP, 8 bit
\$3BE	\$37A	\$27A	R/W	PC, 5/4 bit
\$3BD	\$379	\$279	R	PS, 5 bit

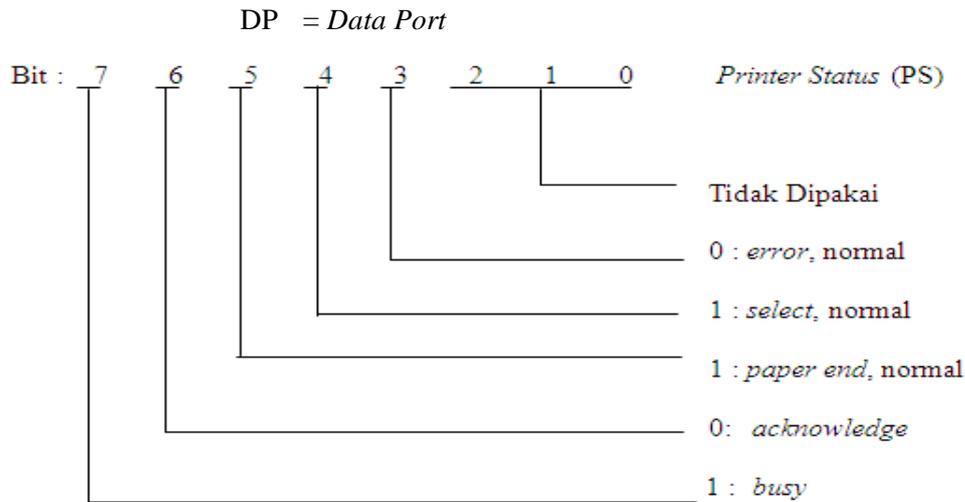
Keterangan :

R = *Read*, dapat dibaca

W = *Write*, dapat ditulis

PC = *Printer Control*

PS = *Printer Status*

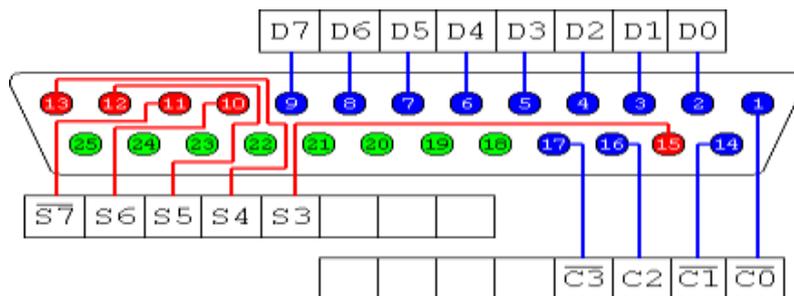


BIOS melakukan pengujian *port paralel* ketika komputer dihidupkan. Bit 7 bersifat *invert*, artinya jika *register printer* status berlogika 1, maka *socket* yang bersangkutan berlogika 0, demikian pula sebaliknya.

Dasar-Dasar Port Paralel

Port paralel mempunyai kemampuan mengirim 8 bit data sedangkan *serial port* hanya dapat mengirim 1 bit data dalam waktu

yang bersamaan. *Port paralel* kemudian menjadi alat yang dapat dihubungkan ke banyak *peripheral device* yang fungsinya sebagai pengontrol dan penerima input dari *external device*. Konektor yang digunakan pada *port paralel* adalah *DB-25* dengan jumlah pin 25 buah. Nama-nama sinyal yang terdapat pada pin konektor *DB-25 port* printer tersebut dapat dilihat pada gambar berikut ini.



Gambar 3 Nomor Pin Konektor DB-25

Pin-pin printer *port* merupakan pintu komunikasi dua arah, dari komputer ke *peripheral* (Output) dan sebaliknya dari *peripheral* ke PC (Input). *Port Paralel*

sebagai *interface* dalam menghubungkan komputer dengan peralatan luar menggunakan *socket DB-25* betina. Penomoran dan nama sinyal LPT dapat dilihat pada tabel berikut ini

Tabel 2 Nomor Dan Nama Sinyal LPT

Nama	Nama sinyal	Socket DB-25	Input/Output
DP-0	Data 0	Pin 2	Output
1	Data 1	Pin 3	Output
2	Data 2	Pin 4	Output
3	Data 3	Pin 5	Output
4	Data 4	Pin 6	Output
5	Data 5	Pin 7	Output
6	Data 6	Pin 8	Output
7	Data 7	Pin 9	Output
PC-0	Strobe	Pin 1	Output
1	Autofeed	Pin 14	Output
2	Init	Pin 16	Output
3	Select IN	Pin 17	Output
PS-3	Error	Pin 15	Input
4	Select	Pin 13	Input
5	Paper End	Pin 12	Input
6	Acknowledge	Pin 10	Input
7	Busy	Pin 11	Input
Ground	Ground	Pin 18-25	---

Nama-nama *signal* serta fungsi yang terdapat pada pin konektor *DB-25 printer port* tersebut adalah :

1. *Strobe* : saluran ini diaktifkan komputer jika ia akan meneruskan data ke printer.
 2. *Data 0* { Data out 0 }
 3. *Data 1* { Data out 1 }
 4. *Data 2* { Data out 2 }
 5. *Data 3* { Data out 3 }
 6. *Data 4* { Data out 4 }
 7. *Data 5* { Data out 5 }
 8. *Data 6* { Data out 6 }
 9. *Data 7* { Data out 7 }
- } Pin Output
10. *ack* : jika printer telah mengolah data yang diterimanya maka dalam waktu maksimal 30 *mikrodetik* ia memberi sinyal jabat tangan (*hand shake*) *acknowledge (/ack)* ini.
 11. *Busy* : ketika printer menerima data atau mencetak, sinyal ini diaktifkan. Demikian pula jika ada gangguan atau dalam status *offline*.
 12. *Paper Empty* : sinyal ini akan terus aktif sampai kertas baru dipasang, sinyal inilah yang akan diambil sebagai pengontrolan yang dilakukan oleh paralel port melalui pemrograman.
 13. *Select (on)* : ketika printer telah dihidupkan maka sinyal ini yang menyatakan printer dalam keadaan *online*.
 14. *Auto feed* : jika sinyal ini diaktifkan, printer pada akhir setiap baris akan pindah ke baris selanjutnya **secara otomatis**.
 15. *Error* : keluaran ini aktif jika ada gangguan seperti printer tidak tersambung atau tidak menyala (*offline*).
 16. *Initialize Printer* : dengan saluran ini printer kembali ke keadaan awal.
 17. *Select Input* : pemilihan printer sebagai piranti DTE (*Data terminal equipment*) berlangsung dengan sinyal ini.
 18. *18-25 ground*.

Tanda garis di atas (“—”) berarti *signal* inverteran oleh perangkat keras. Masing-masing tersebut *aktiv low* atau terjadi proses peng- *signal* tersebut juga dapat dibagi menurut jenis

port addressnya yakni data port, status port dan *control port*. Untuk operasi normal, tujuh atau delapan saluran data sudah mencukupi, ditambah sinyal *strobe* dan *busy* atau sinyal *acknowledge*. Dari informasi pin tersebut terlihat ada 12 pin saluran keluar dan ada 5 pin saluran untuk masuk.

Pengalamatan Port Paralel

Pada saat pertama kali dihidupkan, *BIOS* (*Basic Input/Output System*) menetapkan

jumlah *port* yang ada dan menentukan alamat *port* untuk LPT0, LPT1, LPT2. Setiap *port paralel* standar *IEEE 1248* terdiri atas 3 port address: *data port*, *status port*, dan *control port* yang membentuk hubungan dua arah (*bi-directional*) dari dan ke printer. Data *port* yang biasanya terdapat pada *IBM PC compatible* dapat dilihat pada Tabel 3.

Tabel 3 Tabel Address Printer

Printer	Data Port	Status	Control
LPT 0	3BCH	3BDH	3BEH
LPT 1	378H	379H	37AH
LPT 2	278H	279H	27AH

Motor Stepper

Motor stepper adalah perangkat *elektromekanis* yang bekerja dengan mengubah *pulsa elektronis* menjadi gerakan *mekanis diskrit*. Motor stepper bergerak berdasarkan urutan pulsa yang diberikan kepada motor. Karena itu, untuk menggerakkan motor stepper diperlukan pengendali motor stepper yang membangkitkan pulsa-pulsa *periodik*. Penggunaan motor stepper memiliki beberapa keunggulan dibandingkan dengan penggunaan motor DC biasa, keunggulannya antara lain adalah : (a) Sudut rotasi motor *proporsional* dengan pulsa masukan sehingga lebih mudah diatur; (b) Motor dapat langsung memberikan *torsi* penuh pada saat mulai bergerak; (c) Posisi dan pergerakan *repetisinya* dapat ditentukan secara *presisi*; (d) Memiliki respon yang sangat baik terhadap mulai, stop dan berbalik (perputaran); (e) Sangat *realibel* karena tidak adanya sikat yang bersentuhan dengan rotor seperti pada motor DC; (f) Dapat menghasilkan perputaran yang lambat sehingga beban dapat *dikopel* langsung ke porosnya dan (g) Frekuensi perputaran dapat ditentukan secara bebas dan mudah pada *range* yang luas.

Motor Stepper adalah suatu jenis motor listrik yang dirancang untuk digunakan dalam suatu *direct digital control system*. Gerakan motor stepper dikontrol oleh sinyal-sinyal digital yang dibangkitkan oleh suatu sistem

digital. *Rotor* motor stepper ini tidak berputar secara *kontinue* seperti pada motor AC maupun DC biasa, tetapi berputar setahap demi setahap. Secara umum, setiap pulsa yang diberikan kepada motor stepper akan membuat rotornya berputar satu langkah searah jarum jam maupun berlawanan arah jarum jam, tergantung pada sinyal atau pulsa yang diberikan. Besar langkah tiap putarannya tergantung pada jenis motor yang digunakan, namun umumnya berkisar antara 0,9⁰ sampai 90⁰. Karena rotor sebuah motor stepper berputar setahap demi setahap, motor ini sering dianggap sebagai suatu alat digital yang mengubah pulsa-pulsa listrik menjadi gerakan-gerakan mekanis yang sebanding. Kemampuan motor ini dapat mengulang gerakan-gerakan yang sama dengan baik. Kesalahan langkah yang disebabkan oleh motor stepper dalam suatu sistem biasanya tidak melebihi 5%.

Sensor

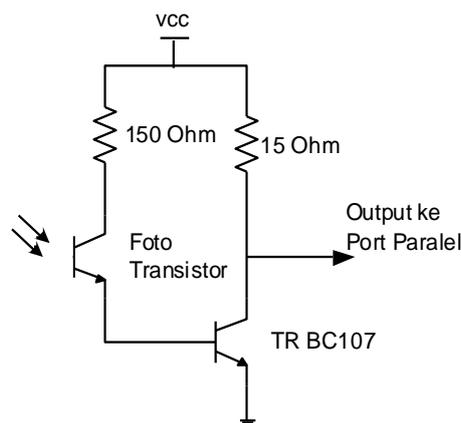
Sensor adalah suatu alat yang mampu mengubah besaran fisik menjadi besaran listrik yang sebanding. Sensor agar dapat bekerja dengan baik harus memiliki beberapa persyaratan yang akan menentukan ketelitian dari sensor tersebut. Persyaratannya adalah (a) Kepekaan, yaitu sensor harus dipilih sedemikian rupa sehingga pada nilai-nilai masukan yang ada dapat diperoleh tegangan listrik keluaran yang diinginkan; (b) Stabilitas

waktu, yaitu untuk nilai masukan tertentu sensor harus dapat memberikan keluaran yang tetap nilainya dalam waktu yang lama; (c) Tidak tergantung temperatur, yaitu keluaran *konversi* tidak tergantung pada temperatur disekelilingnya kecuali sensor temperatur dan (d) *Hysteresis*, yaitu adanya gejala *hysteresis* pada magnetisasi besi.

Fototransistor

Fototransistor adalah komponen elektronika yang berbentuk transistor dengan basis terbuka. *Fototransistor* lebih sensitif terhadap cahaya jika dibandingkan dengan *photodiode*. Perbedaan pokok antara *fototransistor* dengan *photodiode* adalah penguat dc. Jika pada kedua peralatan tersebut diberikan intensitas cahaya dengan jumlah yang sama

maka *fototransistor* akan menghasilkan arus dc yang lebih besar daripada *photodiode*. Makin tinggi *sensitifitas* dari suatu *fototransistor* maka kecepatan makin rendah. *Fototransistor* di sini digunakan sebagai pasangan dari infra merah. Karakteristik dari *fototransistor* adalah sebagai berikut : *fototransistor* adalah *transistor silicon dwi polar NPN* dalam kemasan dengan penutup transparan yang memungkinkan cahaya dapat mencapai sambungan PN-nya. Arus *Emiter* dan *Colector* biasanya sama. Hal ini disebabkan oleh karena sambungan basisnya terbuka dan *fototransistor* biasanya digunakan untuk umpan balik negatif. Dilihat dari kepekaannya *fototransistor* seratus kali lebih peka dari *photodiode*. Untuk rangkaian penerima *fototransistor* dapat dilihat pada gambar 4.



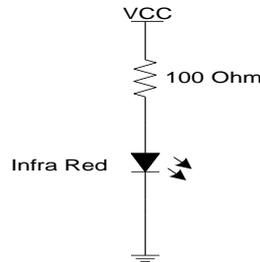
Gambar 4 Rangkaian Penerima (Fototransistor)

Led Infra Merah

Infra merah merupakan LED yang terbuat dari bahan Ga As (*Galium Arsenid*), dimana bahan ini mampu meradiasikan sinyal infra merah. Dalam *spektrum gelombang elektromagnetik*, cahaya *infra red* mempunyai panjang gelombang 1 – 100 m ($10^2 - 10^6$) dan mempunyai *frekwensiantara* 10^{11} Hz hingga 10^{14} Hz . Infra merah ini banyak digunakan untuk pengiriman informasi tanpa kabel (*wireless*).

Pada dasarnya kerja infra merah sama dengan cara kerja LED, jika tegangan arah maju

dikenakan pada infra merah, maka daerah *depleksi* akan mengecil dan potensial penghalang jadi rendah akibatnya *elektron* dari *type N* akan melewati sambungan P-N untuk bergabung dengan *hole* dalam *type P*. *Elektron* bebas dalam pita *konduksi* mempunyai tingkat energi lebih tinggi dari lubang. Jika terjadi penggabungan berarti *elektron* turun ke tingkat energi yang lebih rendah. Turunnya *elektron* ini membebaskan sejumlah energi cahaya kasat mata sehingga infra merah dapat mengemisikan cahaya. Rangkaian pemancar infra merah dapat dilihat pada gambar 5.



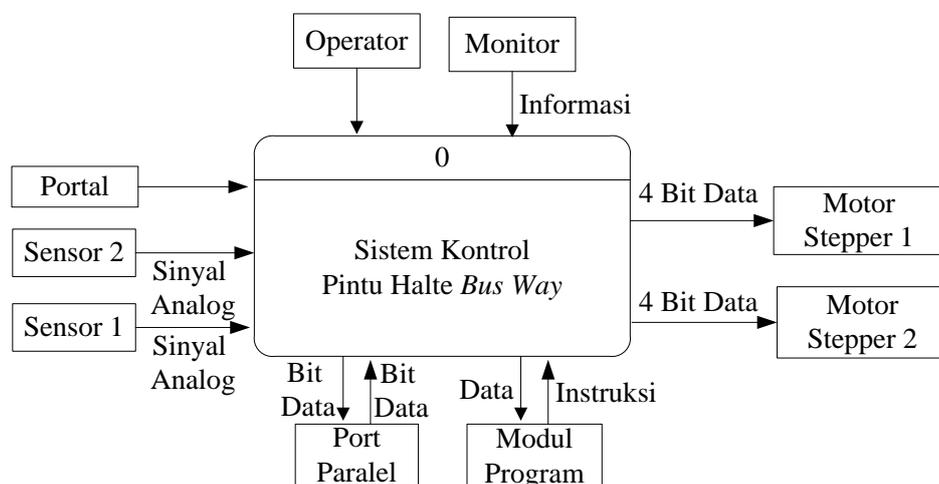
Gambar 5 Rangkaian Pemancar (*Infra Red*)

METODE PENELITIAN

Context Diagram

Dalam proses penganalisaan perlu dilakukan pendefinisian terlebih dahulu terhadap sistem yang dirancang secara menyeluruh. Dimana ruang lingkup pembahasan lebih jelas, *context diagram* dari

“Simulasi Pengontrolan Pintu Halte *Bus Way* Menggunakan Sensor *Led Infra Red* Dan *Interface Port Parallel* Dengan Memanfaatkan File Data Pada Aplikasi *Personal Computer* (PC) didukung bahasa pemrograman *Borland Delphi 7.0*”, dapat dilihat pada gambar 5.



Gambar 5 Context Diagram

Pada *context diagram* di atas terdiri dari sebuah lambang proses yang diberi nama sistem kontrol pintu halte *bus way*. Proses ini berinteraksi dengan beberapa *entity* yang dapat diuraikan sebagai berikut : (1) *Modul Program* : Melakukan pembacaan terhadap pin-pin *port parallel*, baik pembacaan terhadap sinyal-sinyal input yang masuk, memberikan instruksi-

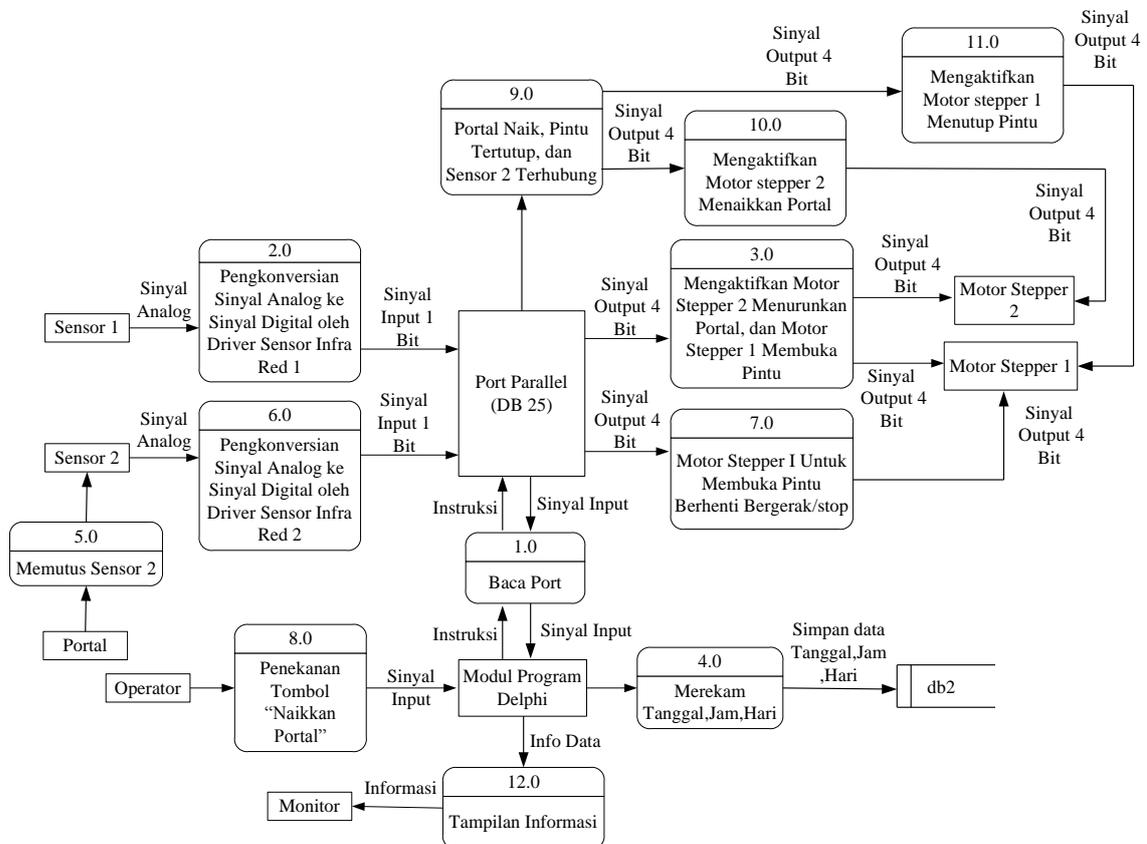
instruksi untuk mengaktifkan pin-pin output sehingga *motor stepper* bergerak, juga melakukan koneksi ke file data. Modul program mengontrol semua proses yang terjadi pada sistem; (2) *Port Paralel* : Berfungsi sebagai *interface* yang menghubungkan antara *internal device* dan *external device*; (3) *Motor Stepper 1* : *Motor stepper* 1 akan menggerakkan dua buah

pintu halte *bus way*, dimana pergerakan ini terjadi sesuai dengan instruksi dari modul program; (4) *Sensor 1*: *Sensor* ini aktif ketika *led infra* merah yang terhubung dengan *sensor* diputus oleh *bus way* yang lewat, maka *sensor* akan mengirimkan data input ke sistem yang kemudian input ini akan diproses oleh *modul* program untuk menggerakkan secara bersamaan *motor stepper 2* untuk menurunkan portal dan *motor stepper 1* untuk membuka pintu halte *bus way*. Input data dari *sensor 1* akan disimpan dalam file data, dimana data yang akan disimpan adalah tanggal, jam, dan hari yang berpedoman pada tanggal, jam, dan hari yang ada pada sistem komputer; (5) *Motor Stepper 2*: Berfungsi untuk menaikkan dan menurunkan *portal*. Adapun *portal* ini berfungsi sebagai batas berhentinya *bus way*. Apabila *bus way* telah berhenti di halte, maka bus tersebut akan bisa berjalan kembali apabila pintu halte telah ditutup dan *portal* telah dinaikkan. Hal ini bertujuan untuk menjamin keselamatan penumpang; (6) *Sensor 2*: *Sensor* ini aktif ketika *led infra* merah yang terhubung dengan

sensor diputus oleh *portal*, maka *sensor* akan mengirimkan data input ke sistem yang kemudian input ini akan diproses oleh *modul* program untuk menghentikan pergerakan *motor stepper 1* yang sedang bergerak membuka pintu halte *bus way*; (7) *Operator*: Merupakan karyawan yang bertugas melakukan pembukaan *portal* pada jalur halte *bus way* pada saat mobil yang berhenti di halte akan bergerak kembali; (8) *Portal*: *Portal* digunakan untuk mengantisipasi terjadinya kecelakaan penumpang pada saat bus berhenti di halte. *Portal* pada saat turun akan memutus *sensor 2* dan (9) *Monitor I*: *Monitor* untuk menampilkan menu program, informasi nilai *sensor*, dan data yang disimpan.

Data Flow Diagram Level 0

Pada sub bab ini dijabarkan mengenai *data flow diagram* yang merupakan uraian lebih terperinci dari sistem yang dirancang. Adapun gambar 6 berikut adalah *data flow diagram level 0* yang diuraikan berdasarkan pada *context diagram* sebelumnya.

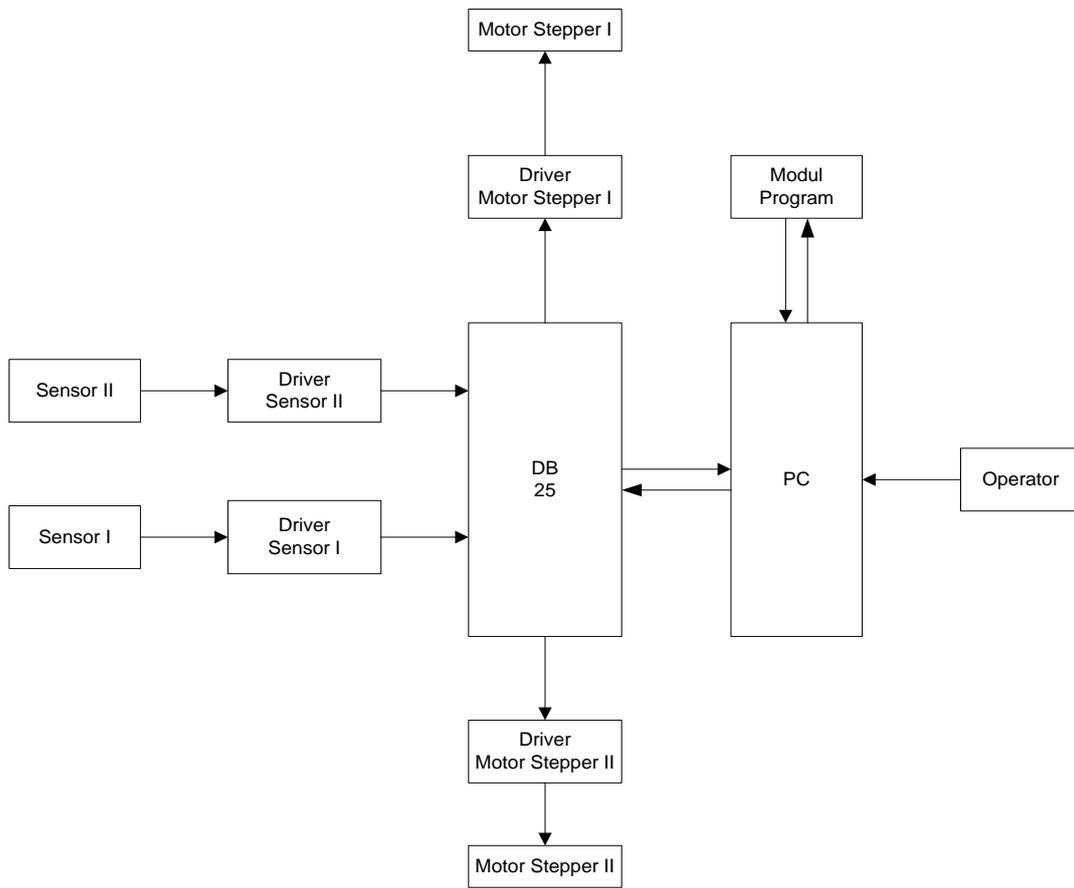


Gambar 6 Data Flow Diagram Level 0

Blok Diagram

Untuk memperjelas dan memahami cara kerja sistem secara keseluruhan, dan keterkaitan

antar *subsistem*, maka dapat dilihat blok diagram pada gambar 7.



Gambar 7 Blok Diagram



Gambar 8 Rancangan Fisik Alat

Rancangan Fisik Alat

Secara umum, rancangan sistem ini keseluruhannya dapat dilihat pada gambar 8. Dari gambar 8 dapat dilihat bahwa sistem pengontrolan pintu halte bus way ini membutuhkan dua pasang sensor dimana satu pasang sensor diletakkan di jalan masuk pada jalur khusus bus way dengan jarak tertentu, satu pasang lagi diletakkan di tembok pembatas sebelah dalam/dekat halte dengan jarak tertentu. Sensor-sensor ini akan memberikan informasi pada mikroprosesor untuk melakukan tindakan menjalankan program sesuai dengan informasi yang diberikan oleh sensor tersebut. Setelah informasi dari sensor terbaca oleh program maka melalui interface port parallel akan dapat mengirimkan data melalui port sehingga rangkaian driver mendapat sinyal untuk dapat menggerakkan motor.

Sensor pertama yang terletak pada jalan masuk ke jalur bus way akan terputus pada saat bus way lewat di antara sensor tersebut, informasi ini akan diterima oleh modul program sehingga akan menyebabkan portal yang terletak di bagian depan pada jalur bus way akan turun dan kedua buah pintu halte juga akan bergerak terbuka. Pada saat bus way memutus sensor, maka pada file data sistem akan mencatat/merekam tanggal, jam, dan hari saat sensor tersebut terputus. Portal yang turun

akan memutus sensor kedua sehingga kedua pintu halte yang sedang bergerak akan berhenti.

Untuk proses selanjutnya, operator akan menaikkan portal dengan cara mengklik tombol "naikkan portal" yang terdapat pada menu pilihan, sehingga portal akan bergerak naik, pintu halte bergerak menutup, dan sensor 2 kembali terhubung. Setelah pintu halte tertutup dan portal telah kembali berada pada posisi awal/dinaikkan maka bus way sudah dapat berjalan meninggalkan halte tersebut.

HASIL DAN PEMBAHASAN

Analisa Rangkaian

Beberapa aspek yang perlu dikembangkan dalam pemahaman terhadap sistem merupakan satu kesatuan prosedur inti dari sistem tersebut. Sistem dikatakan lengkap bila dalam mencapai tujuan yang telah ditetapkan terjadi interaksi antara sub sistem – sub sistem yang ada. Pada sub bab berikut ini akan dijelaskan mengenai analisa perancangan rangkaian secara keseluruhan dan masing - masing rangkaian yang mendukung tercapainya tujuan pembuatan alat ini. Untuk dapat memperjelas rangkaian dan komponen yang digunakan dapat dilihat pada penjelasan berikut ini.

Analisa Rangkaian Keseluruhan

Pada rancangan alat ini terdiri dari beberapa komponen yang saling berhubungan dimana masing – masing komponen diharapkan dapat bekerja sama dengan baik dan sesuai dengan keinginan. Rangkaian ini terdiri atas *interface port paralel* sebagai penghubung antara komputer dengan *external device*, *sensor infra merah* yang akan memberikan input ke sistem dan *motor stepper* sebagai output dari sistem yang dirancang.

Pada sistem pengontrolan pintu halte *bus way* ini terdapat dua rangkaian *driver sensor infra merah* dan juga terdapat dua rangkaian *driver motor stepper*.

Jika komputer diaktifkan dan memproses program yang telah dirancang, maka pada saat *sensor infra merah* 1 terputus oleh *bus way* data akan dikirimkan ke *modul program* untuk selanjutnya disimpan pada file data dan juga akan menghasilkan instruksi untuk menggerakkan *motor stepper* 2 yang akan

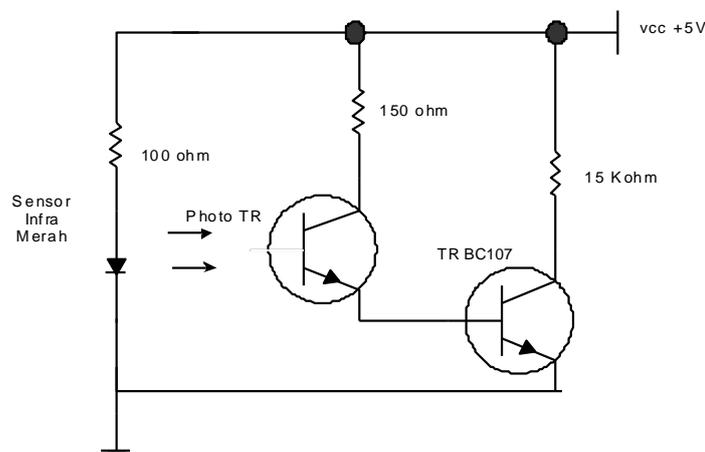
menurunkan portal dan menggerakkan *motor stepper* 1 yang akan membuka kedua buah pintu halte. Pada saat portal turun maka akan memutus *sensor infra merah* 2 sehingga menyebabkan kedua buah pintu pada halte *bus way* yang sedang bergerak akan berhenti.

Selanjutnya saat *bus way* siap untuk meninggalkan halte, maka operator akan mengklik *button* “naikkan portal” pada *menu program*, hal ini akan menyebabkan portal bergerak naik, pintu halte bergerak menutup, dan *sensor* 2 kembali terhubung.

Analisa Rangkaian Perblok

Rangkaian Sensor Infra Red dan Photo Transistor

Agar alat ini akan bekerja dengan sempurna maka diperlukan dua rangkaian *sensor infra merah* dan *photo transistor* diawali dengan pengujian pada *bread board*. Adapun gambar rancangannya dapat dilihat pada gambar 9.



Gambar 9 Rangkaian Sensor Infra Merah Dan Photo Transistor

Rangkaian *sensor infra merah* ini terdiri dari *led infra merah*, *photo transistor*, *TR BC107*, *resistor 15 Kohm*, *resistor 150 ohm*, dan *resistor 100 ohm*. Pada saat *infra merah* yang dipancarkan *led infra merah* tidak terhalang menuju *photo transistor*, *basis photo transistor* mendapat bias tegangan dari cahaya tersebut, akibatnya tegangan pada *basis photo transistor* menjadi naik. Hal ini menyebabkan *photo transistor* menghantar tegangan dari *kolektor* ke *emiter*, tegangan pada *emiter* ini

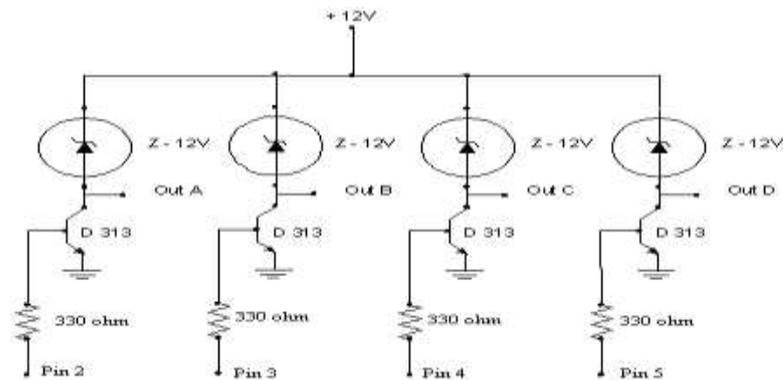
akan diteruskan ke *basis* pada *TR BC107*, yang mengakibatkan transistor aktif dan menghantarkan tegangan dari *kolektor* ke *emiter* sehingga *VCE* bernilai 0,2 Volt ($VCE = 0,2$ Volt). Karena nilai *VCE* bernilai di bawah titik jenuh transistor maka tegangan ini diabaikan sehingga output dari rangkaian berlogika ‘0’.

Sebaliknya pada saat *infra merah* yang dipancarkan *LED infra merah* terhalang menuju *photo transistor*, *photo transistor* tidak

mendapat bias tegangan cahaya, akibatnya tegangan pada *basis photo transistor* menjadi 0 Volt. Hal ini menyebabkan *photo transistor* dalam kondisi *cut off* (menyumbat), sehingga tidak ada tegangan yang dihantarkan ke *basis TR BC107* yang mengakibatkan transistor ini tidak aktif, tegangan dari *VCC* tertahan pada *kolektor* sehingga tegangan antara *kolektor* dan *emiter* mendekati tegangan *VCC* yaitu 4,8 volt ($VCE = 4,8 \text{ Volt}$). Karena tegangan $VCE = 4,8 \text{ Volt}$ maka output dari rangkaian berlogika '1'. Dari hasil pengukuran di atas, maka rangkaian *led infra merah* dan *photo transistor* telah dapat bekerja dengan baik.

Rangkaian Motor Stepper

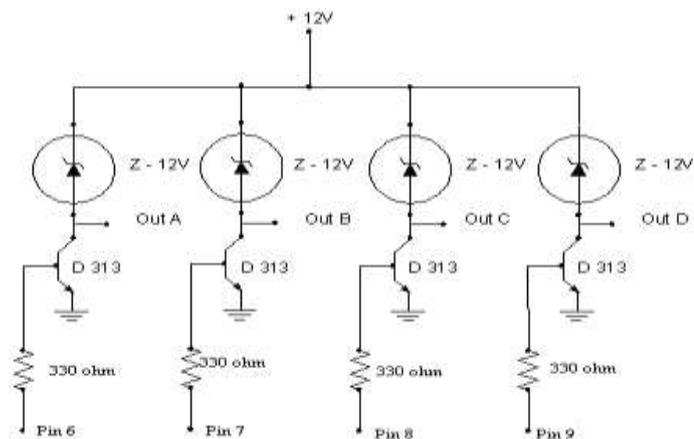
Bentuk dasar *motor stepper* yang paling sederhana terdiri dari sebuah *rotor*, yang merupakan *magnet* permanen dan sebuah *stator* yang dililiti kumparan, sehingga membentuk medan listrik. Jika *stator* diberi arus listrik maka sisi-sisi *rotor* akan membentuk kutub-kutub *magnet*. Jika kutub *magnet stator* dan *rotor* sama, maka kedua *magnet* akan saling tolak menolak sehingga mengakibatkan *rotor* berputar 1 *step*. Arah perputaran ini dapat dua arah tergantung dari faktor mekanik *motor stepper* itu sendiri.



Gambar 10 Rangkaian Driver Motor Stepper 1

Rangkaian driver *motor stepper* pada gambar 10 berfungsi untuk menggerakkan *motor stepper* 1 untuk menggerakkan kedua buah pintu *halte bus way*. *Motor Stepper 1* terhubung ke *Pin 2*, *Pin 3*, *Pin 4*, *Pin 5*

dengan keluaran 1 *bit*. Sedangkan *Motor Stepper 2* akan terhubung ke *Pin 6*, *Pin 7*, *Pin 8* dan *Pin 9* untuk menggerakkan portal naik atau turun. Rangkaian *motor stepper 2* dapat dilihat gambar 11.



Gambar 11 Rangkaian Driver Motor Stepper 2

Blok rangkaian *driver* pada rangkaian *motor stepper* ini berfungsi sebagai *saklar*. Seperti diketahui *motor stepper* bekerja berdasarkan bit-bit yang diterimanya, untuk itu diperlukan transistor sebagai penyambung dan pemutus arus.

Interface

Interface yang digunakan adalah *port paralel* dimana susunan *pin* yang digunakan untuk simulasi pengontrolan pintu halte *bus way* ini adalah : (1) *Pin 2-5* Dihubungkan ke *Motor Stepper 1* dimana fungsi *pin* ini untuk keluaran; (2) *Pin 6-9* untuk ke *Motor Stepper 2* yang juga sebagai saluran keluaran; (3) *Pin 10* dihubungkan ke *sensor infra merah* pertama. Dimana fungsi *pin* ini sebagai saluran *input*; (4) *Pin 11* dihubungkan ke *sensor infra merah*

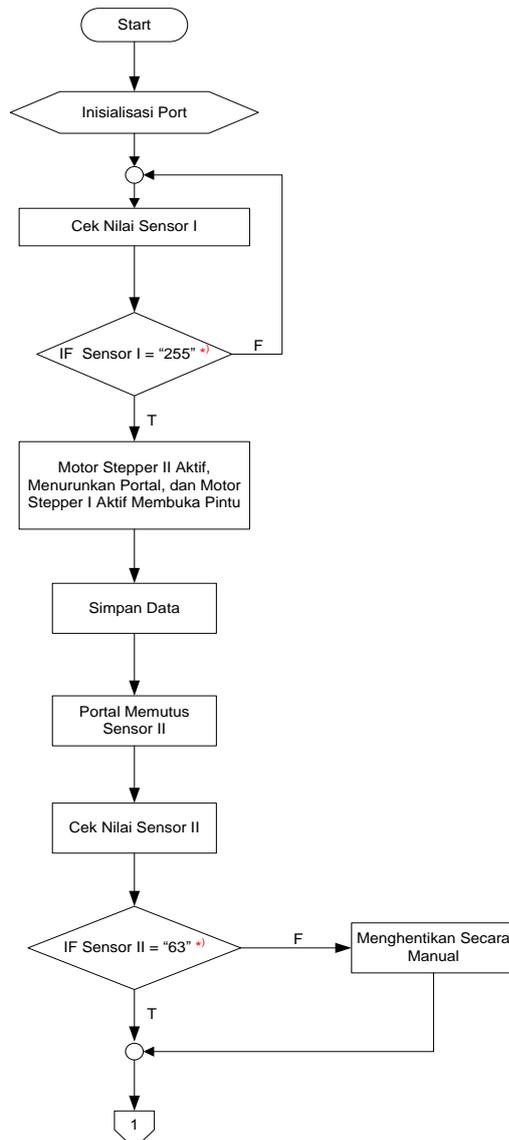
kedua sebagai saluran *input* dan (5) *Pin 25* dihubungkan ke *ground*.

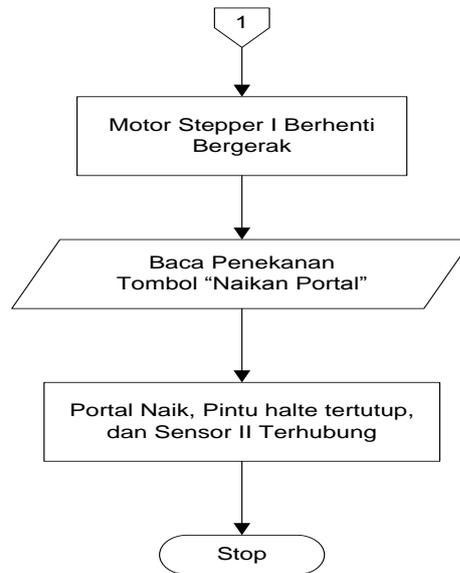
Analisa Program

Pada perancangan sistem dibutuhkan suatu rancangan logika program, logika program ini dapat digambarkan pada *flowchart* untuk memudahkan dalam perancangan *modul program*.

Flowchart Program

Agar *modul program* yang dirancang memiliki struktur dengan kualitas yang baik dan mudah dimengerti maka perlu diawali dengan penentuan logika dalam program. Adapun logika dasar yang digambarkan pada penulisan ini adalah dengan menggunakan *Flow Chart program* yang dapat dilihat pada gambar 12.





Gambar 12 Flowchart Program

Keterangan : *) IF *Sensor I* = "255" , artinya apabila nilai sensor 1 bernilai 255 maka motor stepper II aktif menurunkan portal, dan motor stepper I aktif membuka kedua buah pintu halte.

*) IF *Sensor I* = "63" , artinya apabila nilai sensor 2 bernilai 63 maka motor stepper I yang sedang bergerak membuka pintu halte akan berhenti bergerak.

Algoritma Dalam Pseudocode

Logika Program *Pseudocode* :

Kamus : S1,S2 := Integer

Algoritma :

```

S1 ← input
S2 ← input
If bus way memutus sensor1
then
  If S1 = 255 then
    Begin
      Portal turun;
      Pintu terbuka;
    End;
  Endif;
  If Portal memutus
  sensor2 then
    If S2 = 63 then
      Begin
        Pintu berhenti;
      End;
    Endif;
  Endif;
Endif.
End.
    
```

Rancangan Modul Program

Modul Program : Pada sub bab ini, penulis tidak akan membahas secara menyeluruh, namun hanya membahas tentang beberapa hal yang dianggap penting atau inti dari program ini diantaranya adalah :

1. Modul Program untuk pembacaan *input /output* (i/o) pada *Microsoft Windows Xp* Modul program yang menggunakan fungsi pemanggilan fungsi pada *fileIO.dll* yang telah dicopykan pada *system 32*, dengan *syntak* pemanggilan pada program sebagai berikut:


```

function
PortIn(Port:Word):Byte;stdcall;external
'io.dll';
procedure
PortOut(Port:Word;Data:Byte); stdcall;
external'io.dll';
            
```
2. Modul Program pemanggilan *procedure delay* pada *port parallel*

```

procedure delay(lama:dword);
var temp:dword;
begin
temp:=gettickcount;
repeat
            
```

```

        application.ProcessMessages
        ;
        until      (gettickcount-
temp)>=lama;
    end;
3. Modul Program untuk Mengaktifkan
Motor Stepper
Modul program yang digunakan adalah
program yang digunakan untuk
mengaktifkan port output dari port
parallel.
Modul program untuk membuka pintu
halte bus way :
procedure bukapintu;
var j : integer;
    Begin
        for j := 1 to 45 do
            begin
                portout($378,$01);delay(
10);
                portout($378,$02);delay(
10);
                portout($378,$04);delay(
10);
                portout($378,$08);delay(
10);
            end;
            PortOut($378,$00);
        end;
        procedure
TForm1.Button5Click(Sender:
TObject);
        begin
            bukapintu;
        end;
Modul program untuk mentup pintu
halte bus way :
procedure tutuppintu;
var k : integer;
    Begin
        for k := 1 to 44 do
            begin
                portout($378,$08);delay(
10);
                portout($378,$04);delay(
10);
                portout($378,$02);delay(
10);
                portout($378,$01);delay(
10);
            end;
            PortOut($378,$00);

```

```

        end;
        procedure
TForm1.Button6Click(Sender:
TObject);
        begin
            tutuppintu;
        end;
Modul program untuk menaikkan portal
:
procedure naikportal;
var i : byte;
    begin
        for i:=1 to 14 do
            begin
                portout($378,$10);delay
(10);
                portout($378,$20);delay
(10);
                portout($378,$40);delay
(10);
                portout($378,$80);delay
(10);
            end;
            PortOut($378,$00);
        end;
        procedure
TForm1.Button2Click(Sende
r: TObject);
        Begin
            naikportal;
            tutuppintu;
        end;
Modul program untuk menurunkan
portal :
procedure turunportal;
var i : byte;
    begin
        for i:=1 to 14 do
            Begin
                portout($378,$80);delay(10
);
                portout($378,$40);delay(10
);
                portout($378,$20);delay(10
);
                portout($378,$10);delay(10
);
            End;
            PortOut($378,$00);
        end;

```

```

    procedure
    TForm1.Button4Click(Sender:
    TObject);
begin
    turunportal;
    bukapintu;
end;

```

4. Modul Program untuk Sensor

Modul program yang menggunakan alamat port status yaitu pada pin 10,11 pada port parallel.

```

procedure
TForm1.Timer1Timer(Sender: TObject);
begin
    sensor1 := portin($379);

```

```

Label1.Caption:=IntToStr(se
nsor1);
edit1.Text:=
inttostr(sensor1);
if sensor1 = 255 then
begin
    turunportal;
    bukapintu;
end;
if sensor1 = 63 then
begin
    Button7.Click;
end;
end;

```

KESIMPULAN

Dari semua penjelasan yang tertera dari bab-bab sebelumnya maka dapat penulis ambil kesimpulan seperti di bawah ini :

1. *Personal Computer* (PC) dapat berperan sebagai media pengontrolan yang ditunjang dengan aplikasi bahasa pemrograman dan *peripheral-peripheral*.
2. Jika dibanding dengan sistem-sistem yang telah ada seperti sistem manual, sistem ini jauh lebih hemat.
3. Penggunaan paralel printer *port* sebagai *interface* untuk mengendalikan perangkat keras adalah salah satu cara untuk menghemat dana dan waktu, karena selain sudah langsung tersedia pada IBM PC *compatible*, paralel printer *port* memiliki sekurang-kurangnya 8 jalur output dan 5 jalur untuk input sehingga kemampuannya mengendalikan perangkat keras tersebut tidak diragukan lagi.
4. Penggunaan *paralelportinterface* memiliki kekurangan yakni dalam mencetak informasi ke printer tidak dapat dilakukan selama rangkaian terpasang, hal ini dapat dilakukan dengan cara memanfaatkan printer yang memiliki *port* USB.
5. Otomatisasi yang dilakukan pada pengontrolan buka dan tutup pintu halte *bus way* dapat memberikan keamanan

yang lebih baik terhadap keselamatan pelanggan.

6. Pemakaian file data diperlukan untuk menyimpan data-data yang dibutuhkan dikemudian hari.
7. Pengoperasian rancangan *prototype* ini tidaklah terlalu sulit dan diharapkan nantinya alat ini memberikan hasil yang lebih baik jika dibandingkan dengan sistem manual. Sensor *LEDInfra Red* dapat digunakan sebagai input untuk menggerakkan pintu secara otomatis. Selain itu dapat juga sebagai input ke file data.

DAFTAR KEPUSTAKAAN

- Malvino, Albert Paul. 1990. *Prinsip-Prinsip Elektronika*. Jakarta: Erlangga.
- Nogroho, Widodo. 2002. *Tip dan Trik Pemrograman Delphi*. Yogyakarta: Andi Offset.
- Martina, Inge. 2004. *36 Jam Belajar Komputer Pemrograman Visual Borland Delphi 7*. Jakarta: PT. Elex Media Komputindo.
- Mangkuto, Hengky Alexander. 2005. *Bank Soal Delphi*. Jakarta: PT. Elex Media Komputindo.
- Madgoms. 2001. *Pemrograman Borland Delphi 7*. Yogyakarta: Andi Offset.
- Kusnassriyanto, Saiful Bahri dan Wawan Sjachriyanto. 2005. *Pemrograman Delphi*. Bandung: Informatika.