



**RANCANG BANGUN SISTEM PEMANTAU DAN TRANSMISI
DATA TEKANAN DARAH PADA MOBILE PLATFORM
ANDROID MENGGUNAKAN KONEKSI BLUETOOTH**



UNIVERSITAS
Dinamika

Oleh :

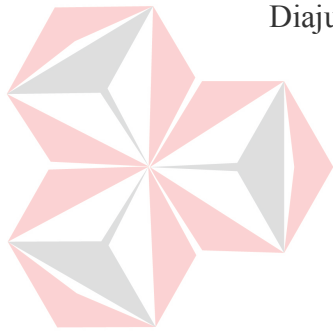
Parlindungan Fernando Nainggolan

10.41020.0070

**SEKOLAH TINGGI MANAJEMEN INFORMATIKAN &
TEKNIK KOMPUTER SURABAYA
2014**

**RANCANG BANGUN SISTEM PEMANTAU
DAN TRANSMISI DATA TEKANAN DARAH PADA MOBILE
PLATFORM ANDROID MENGGUNAKAN KONEKSI BLUETOOTH**

TUGAS AKHIR



Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana Komputer

UNIVERSITAS
Dinamika

Oleh:

Nama : Parlindungan Fernando Nainggolan

Nim : 10.41020.0070

Jurusan : Sistem Komputer

Program : S1 (Strata Satu)

Tugas Akhir

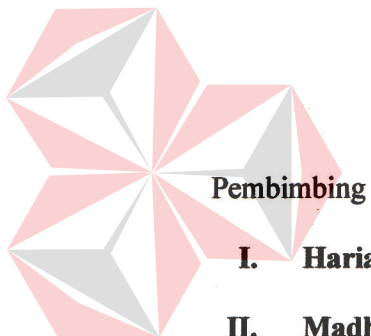
**RANCANG BANGUN SISTEM PEMANTAU DAN TRANSMISI DATA
TEKANAN DARAH PADA MOBILE PLATFORM ANDROID
MENGUNAKAN KONEKSI BLUETOOTH**

Dipersiapkan dan disusun oleh

Nama : Parlindungan Fernando Nainggolan

NIM : 10.41020.0070

Telah diperiksa, diuji dan disetujui oleh Dewan Penguji
pada : Desember 2014



Susunan Dewan Penguji

Pembimbing

- I. Harianto, S.Kom., M.Eng.**
- II. Madha Christian Wibowo, S.Kom.**

Penguji

- I. Dr. Jusak.**
- II. Ira Puspasari, S.Si., M.T.**

Tugas Akhir ini telah diterima sebagai salah satu persyaratan
Untuk memperoleh gelar Sarjana

Dr. Jusak.
Dekan Fakultas Teknologi dan Informatika

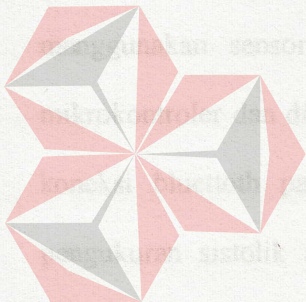
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

PERNYATAAN

Dengan ini menyatakan dengan benar, bahwa Tugas Akhir ini adalah asli hasil karya saya, bukan plagiat baik sebagian maupun apalagi keseluruhan. Karya atau pendapat orang lain yang ada dalam tugas akhir ini adalah semata hanya rujukan yang dicantumkan dalam daftar pustaka saya.

Apabila dikemudian hari ditemukan adanya tindakan plagiat pada karya tugas akhir ini, maka saya bersedia untuk dilakukan pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.

Surabaya, 15 Desember 2014



Parlindungan Fernando Nainggolan
NIM : 10.41020.0070

UNIVERSITAS
Dinamika

Kata kunci: Tensimeter, Android, Mikrokontroler, Bluetooth.

DAFTAR ISI

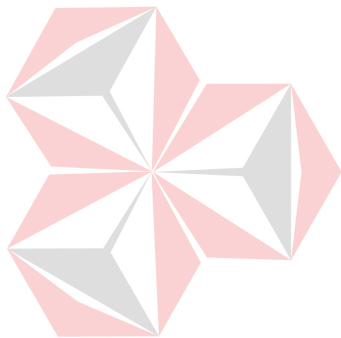
Halaman_Toc365025102

ABSTRAKSI	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR TABEL_Toc365025106.....	x
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xvi
BAB I PENDAHULUAN	1
1.1.Latar Belakang Masalah.....	1
1.2. Perumusan Masalah.....	3
1.3. Pembatasan Masalah.....	3
1.4. Tujuan	3
1.5. Kontribusi	4
1.6. Sistematika Penulisan.....	4
BAB IILANDASAN TEORI	6
2.1. Tekanan Darah	6
2.2. Tensimeter.....	8
2.2.1. MPX5100DP	9

2.2.2. Valve.....	11
2.2.3. Pompa DC	12
2.3. <i>Bluetooth</i>	13
2.3.1. HC-05.....	13
2.4. Mikrokontroler.....	15
2.5. LCD (<i>Liquid Crystal Display</i>).....	17
2.6. <i>Downloader USBASP</i>	19
2.7. <i>Android</i>	20
2.7.1. Basic4Android.....	21
2.8. <i>Motor Driver</i>	22
2.8.1. L298	22
BAB III METODE PENELITIAN	24
3.1. Alat dan Bahan Penelitian	30
3.1.1. Alat Penelitian	30
3.1.2. Bahan Penelitian	31
3.2. <i>Jalan Penelitian</i>	31
3.2.1. Perancangan Perangkat Keras (<i>Hardware</i>)	31
3.2. Perancangan Perangkat Lunak.....	42
3.3. Langkah Pengujian	54

3.3.1. Pengujian Rangkaian Regulator	55
3.3.2. Pengujian Rangkaian <i>Driver</i>	56
3.3.3. Pengujian Rangkaian <i>Minimum</i> Sistem	57
3.3.4. Pengujian Pengiriman dan Penerimaan Data HC-05	58
3.3.5. Pengujian Program Tekanan Darah	60
3.3.6. Pengujian Sistem Secara Keseluruhan.....	60
 BAB IV HASIL DAN PEMBAHASAN	 62_Toc365025106
4.1. Hasil Pengujian Perangkat Keras.....	62
4.1.1. Hasil Pengujian Rangkaian Regulator	62
4.1.2. Hasil Pengujian Rangkaian <i>Driver</i> Motor	63
4.1.3. Hasil Pengujian Rangkaian <i>Minimum</i> Sistem.....	65
4.2. Hasil Pengujian Perangkat Lunak.....	66
4.2.1. Hasil Pengujian Program Pengiriman Serial	66
4.2.2. Hasil Pengujian Program Tekanan Darah.....	71
4.3. Hasil Pengujian Secara Keseluruhan	73
4.3.1. Hasil Pengujian Sampel Pertama.....	74
4.3.2. Hasil Pengujian Sampel Kedua	76
4.3.3. Hasil Pengujian Sampel Ketiga	78
4.3.4. Hasil Pengujian Sampel Keempat Hingga Sampel Keduapuluh ..	80

4.4. Pembahasan Keseluruhan	81
BAB V PENUTUP	84
5.1. Kesimpulan	84
5.2. Saran	85
DAFTAR PUSTAKA	86
LAMPIRAN 1	88
LAMPIRAN 2	96



UNIVERSITAS
Dinamika

DAFTAR GAMBAR

Halaman_Toc365025102

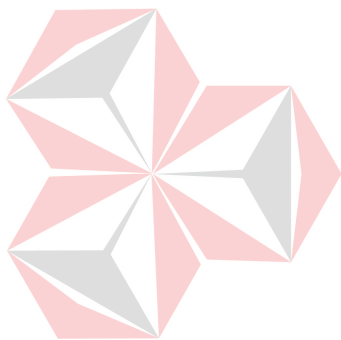
Gambar 2.1 Contoh hasil sinyal <i>output</i> dari sensor tekanan	7
Gambar 2.2 Contoh letak sinyal hasil ekstraksi.....	8
Gambar 2.3. Tensimeter digital.....	9
Gambar 2.4. Tabel kebenaran <i>Input/Output</i> sensor	10
Gambar 2.5. <i>Datasheet</i> MPX5100DP	11
Gambar 2.6. <i>Valve</i> 5V	12
Gambar 2.7. Pompa DC 6V	12
Gambar 2.8. <i>Bluetooth</i> HC-05	15
Gambar 2.9. <i>Minimum</i> Sistem ATmega32.....	15
Gambar 2.10. Konfigurasi <i>pin</i> ATmega32.....	17
Gambar 2.11. LCD 16 x 2	18
Gambar 2.12. Karakter-karakter pada LCD	18
Gambar 2.12. <i>Downloader</i> USBASP.....	20
Gambar 2.13. L298H.....	23
Gambar 3.1. Blok diagram keseluruhan.....	25

Gambar 3.2. Blok diagram alat pengukur tekanan darah.....	26
Gambar 3.3. Blok diagram HC-05	27
Gambar 3.4. Blok diagram <i>Android</i>	28
Gambar 3.5. Blok diagram alir keseluruhan	29
Gambar 3.6. Rangkaian <i>minimum system</i>	33
Gambar 3.7. Rangkaian <i>Reset</i>	34
Gambar 3.8. Rangkaian <i>Oscillator</i>	35
Gambar 3.9. Rangkaian Regulator 5V	36
Gambar 3.10. Bentuk fisik <i>driver</i> motor.....	37
Gambar 3.11. Gambar skematik <i>driver</i> motor.....	39
Gambar 3.12. Rangkaian LCD.....	40
Gambar 3.13. Rangkaian alat pengukur tekanan darah.....	41
Gambar 3.14. Konfigurasi USART pada CVAVR	43
Gambar 3.15. Konfigurasi <i>port</i> pada <i>device manager</i>	44
Gambar 3.16. Konfigurasi awal Hterm	45
Gambar 3.17. Konfigurasi CR-LF	45
Gambar 3.18. Memanggil modul <i>bluetooth</i>	46
Gambar 3.19. <i>Setting boudrate</i>	47
Gambar 3.20. Pemilihan fitur basic4android	48
Gambar 3.21. Langkah-langkah Pengujian	55

Gambar 4.1. Pengujian rangkaian regulator.....	63
Gambar 4.2. Motor DC berputar	64
Gambar 4.3. Tegangan terukur motor	64
Gambar 4.4. Tampilan CMD	65
Gambar 4.5. Tampilan <i>download</i> program	65
Gambar 4.6. Pengiriman dari <i>android</i> ke mikrokontroler.....	66
Gambar 4.7. Hasil Pengiriman 100 karakter pada jarak 2 meter, 4 meter, 6 meter, 8 meter, dan 10 meter	67
Gambar 4.8. Hasil Pengiriman 150 karakter pada jarak 2 meter, 4 meter, 6 meter, 8 meter, dan 10 meter	68
Gambar 4.9. Hasil Pengiriman 200 karakter pada jarak 2 meter, 4 meter, 6 meter, 8 meter, dan 10 meter	69
Gambar 4.10. Hasil pengiriman pada jarak 12 meter.....	70
Gambar 4.11. Proses pompa udara ke dalam manset.....	72
Gambar 4.12. Proses kalibrasi.....	72
Gambar 4.13. Tekanan darah terukur.....	73
Gambar 4.14. Pengambilan data tensimeter acuan sampel 1	74
Gambar 4.15. Pengambilan data tensimeter rancangan sampel 1	75
Gambar 4.16. Pengambilan data tensimeter acuan sampel 2	76
Gambar 4.17. Pengambilan data tensimeter rancangan sampel 2	77

Gambar 4.18.Pengambilan data tensimeter acuan sampel 378

Gambar 4.19.Pengambilan data tensimeter rancangan sampel 379



UNIVERSITAS
Dinamika

DAFTAR LAMPIRAN

Halaman_Toc365025102

LAMPIRAN I	88
LAMPIRAN II.....	96



UNIVERSITAS
Dinamika

DAFTAR TABEL

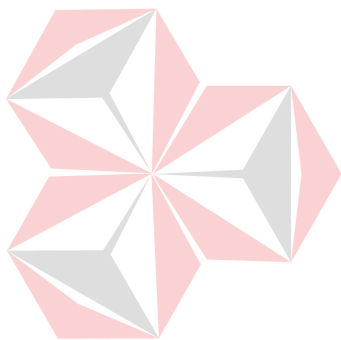
Halaman_Toc365025102

Tabel 2.1. Tabel <i>output</i> MPX5100DP skala volt, Kpa, dan desimal	10
Tabel 2.2. Karakter-karakter pada LCD.....	18
Tabel 3.1. Perancangan <i>interface input/output</i>	34
Tabel 3.2. Pin Input/Output MPX5100DP	41
Tabel 3.3. Pin Input/Output <i>driver</i> motor dan pompa DC	42
Tabel 3.4. Pin Input/Output dan logika <i>valve</i>	42
Tabel 3.5. Pengujian rangkaian regulator	56
Tabel 3.6. Pengujian rangkaian <i>Driver</i> Motor.	57
Tabel 3.7. Pengujian Rangkaian <i>Minimum</i> Sistem.	58
Tabel 3.8. Pengujian pengiriman dan penerimaan data <i>serial</i>	59
Tabel 3.9. Pengujian Program Tekanan Darah.	60
Tabel 3.10. Pengujian sistem Secara Keseluruhan.....	61
Tabel 4.1. Hasil pengujian rangkaian regulator.	62
Tabel 4.2. Hasil Pengujian Serial (jarak dan banyak karakter).....	71
Tabel 4.3. Perbandingan tensimeter acuan dan rancangan pada sampel 1.....	75
Tabel 4.4. Perbandingan tensimeter acuan dan rancangan pada sampel 2.....	77

Tabel 4.5. Perbandingan tensimeter acuan dan rancangan pada sampel 3..... 79

Tabel 4.6. Pengambilan data tensimeter acuan dan rancangan sampel 4 - 20 80

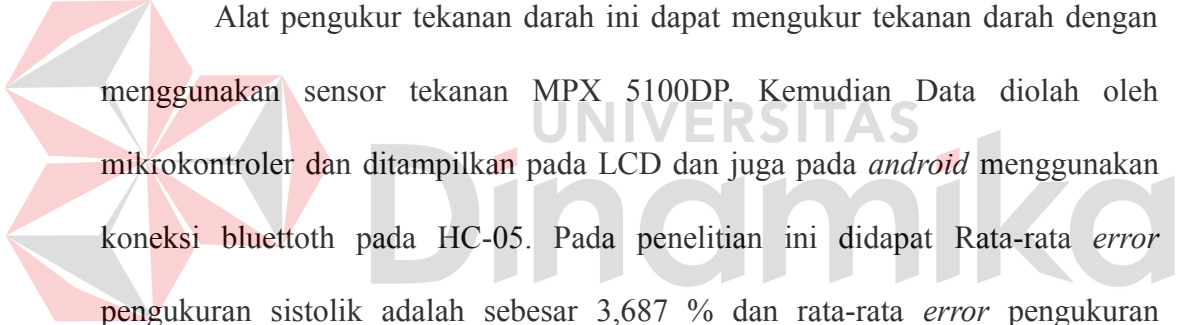
Tabel 4.7. Pengambilan data tensimeter acuan dan rancangan sampel 1 - 20 81



UNIVERSITAS
Dinamika

ABSTRAKSI

Alat pengukur tekanan darah merupakan salah satu alat yang digunakan manusia untuk mengukur tekanan darah. Oleh karena itu banyak teknologi – teknologi yang digunakan untuk mengembangkan alat pengukur tekanan darah. Dalam mengukur tekanan darah dapat menggunakan metode osilometri maupun metode korotkoff, dan pada penelitian ini digunakan metode osilometri. Pada penelitian ini telah dirancang sebuah rancang bangun alat pengukur dan transmisi data tekanan darah pada *android* dengan menggunakan koneksi *Bluetooth*.



Alat pengukur tekanan darah ini dapat mengukur tekanan darah dengan menggunakan sensor tekanan MPX 5100DP. Kemudian Data diolah oleh mikrokontroler dan ditampilkan pada LCD dan juga pada *android* menggunakan koneksi *bluetooth* pada HC-05. Pada penelitian ini didapat Rata-rata *error* pengukuran sistolik adalah sebesar 3,687 % dan rata-rata *error* pengukuran diastolik adalah sebesar 4,768 %. Dari Hasil rata rata *error* tersebut didapatkan nilai standar deviasi nilai sistolik adalah 2.742 dan nilai standar deviasi nilai diastolic adalah 2.221.

Kata kunci :Tensimeter, *Android*, Mikrokontroler, *Bluetooth*.

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Hipertensi adalah penyakit yang sudah menjadi global burden disamping banyak penyakit lainnya yang dapat menyebabkan kematian. Hal ini didasari pernyataan Ketua Umum Perhimpunan Hipertensi Indonesia (PERHI) yaitu Dr. Arieska Ann Soenarto bahwa pada tahun 2000, penyakit tekanan darah tinggi menyumbang 12,8% dari seluruh kematian dan 4,4% dari semua kecacatan (disabilitas).

Tekanan darah yang tinggi adalah salah satu faktor resiko untuk terjadinya serangan jantung, gagal jantung, stroke, *aneurisma arterial*, dan merupakan penyebab utama gagal jantung kronis. Namun, berbagai kemungkinan terjadinya penyakit tekanan darah tinggi dan tekanan darah rendah dapat dihindari sedini mungkin, yaitu dengan melakukan pemeriksaan tekanan darah secara berkala menggunakan tensimeter.

Tensimeter adalah alat kesehatan yang digunakan untuk mengukur tekanan darah dan saat ini semakin dikembangkan untuk keperluan dunia medis. Dimulai dari tensimeter manual hingga tensimeter digital. Tensimeter digital saat ini juga berkembang menjadi tensimeter digital portabel yang dapat dibawa kemana-mana. Hal ini memudahkan pihak rumah sakit untuk mengetahui kondisi pasien secara lebih akurat dan praktis, karena pasien dapat mengukur tekanan darahnya sendiri meskipun tidak berada didalam rumah sakit. Namun, alat ini memiliki kekurangan, yaitu tidak adanya pantauan khusus dari dokter atas kondisi pasien secara *real-time*.

Disisi lain, perkembangan teknologi telah mendukung adanya komunikasi

jarak jauh yang lebih dikenal dengan telemetri. Telemetri dapat digunakan melalui berbagai media transmisi seperti internet, *MMS*, *SMS*, *Bluetooth*, *wifi* dan lainnya. Hal ini sangat menunjang dalam keperluan ambulatori di mana pasien tetap bisa dipantau oleh pihak rumah sakit atau dokter meskipun pasien tidak berada di rumah sakit. Salah satu teknologi yang sedang populer dan berkembang saat ini dan mendukung fasilitas media transmisi tersebut adalah *Android* pada ponsel *Smartphone*. Hampir semua pengguna ponsel *Android* selalu terhubung dengan internet dan dapat berkomunikasi pesan maupun suara. Sehingga dengan adanya integrasi alat ukur tekanan darah ini dengan ponsel *Android* maka dapat dirancang suatu sistem transmisi data alat ukur tekanan darah untuk sistem pemantau dan transmisi data tekanan darah.

(Adidulung, 2011) Alat yang telah diciptakan mahasiswa ITS yang berjudul Alat Pengukur Tekanan Darah Otomatis Berbasis Mikrokontroler Untuk Pasien Rawat Jalan dengan *SMS Gateway* yang masih memiliki keterbatasan kemudahan *Interface* dengan pengguna. Diharapkan pada tugas Akhir ini dapat mengembangkan system dengan pengiriman pada mobile android untuk meningkatkan efisiensi dalam pengiriman dimana disusunlah sebuah tugas akhir yang berjudul Rancang Bangun Sistem Pemantau dan Transmisi Data Tekanan Darah Pada *Mobile Platform Android* Menggunakan Koneksi *Bluetooth*.

1.2 Perumusan Masalah

Adapun permasalahan yang akan dihadapi ke depannya dalam proses pengerjaan tugas akhir ini adalah:

1. Bagaimana merancang alat pengukur tekanan darah dari sensor tekanan ke *microcontroller*.
2. Bagaimana pengiriman dan penerimaan data pengukuran dari *microcontroller* ke *android* melalui *Bluetooth*.

1.3 Pembatasan Masalah

Dalam perancangan dan pembuatan alat ini, terdapat beberapa batasan masalah, antara lain :

1. Transmisi data dilakukan pada *smartphone* berbasis *android* 4.1 jellybean.
2. Tekanan darah yang dapat diukur hanya dibawah 160 mmHg
3. Jarak maksimal antara modul *bluetooth* HC-05 dan perangkat penerima data adalah 10 meter.

1.4 Tujuan

Tujuan dari pembuatan dan perancangan sistem ini adalah.

1. Merancang alat pengukur tekanan darah dari sensor tekanan ke *microcontroller*.
2. Merancang pengiriman dan penerimaan data pengukuran dari *microcontroller* ke *android* menggunakan *bluetooth*.

1.5 Kontribusi

Pembuatan aplikasi pemantau perangkat keras berbasis *android* ini diharapkan dapat diaplikasikan di laboratorium rumah sakit sehingga dapat

membantu dokter yang memantau kondisi pasien pemilik perangkat keras ini dalam pengambilan keputusan. Diharapkan dengan alat ini dapat meningkatkan kesadaran masyarakat akan pentingnya memantau kondisi kesehatan, terutama tekanan darah dan mencegah adanya penyakit lebih lanjut.

1.6 Sistematika Penulisan

Penulisan buku ini secara sistematis diatur dan disusun dalam lima bab yang di dalamnya terdapat beberapa sub bab, di mana akan dijelaskan secara rinci semua penjelasan dalam pembuatan alat ini. Secara ringkas uraian materi dari bab pertama hingga bab terakhir adalah sebagai berikut.

BAB I : PENDAHULUAN

Bab ini merupakan pendahuluan dari karya tulis Tugas Akhir yang membahas mengenai latar belakang masalah, perumusan masalah, pembatasan masalah, tujuan, kontribusi, dan sistematika penulisan.

BAB II : LANDASAN TEORI

Bab ini menjelaskan teori yang mendukung pokok pembahasan tugas akhir yang meliputi definisi yang berkaitan dalam tugas akhir ini. Diantaranya pembahasan tentang tekanan darah, tensimeter, sensor MPX-5100DP, *valve*, modul Bluetooth HC-05 *microcontroller* ATMEGA32, LCD(*Liquid Crystal Display*), Basic4android.

BAB III : METODE PENELITIAN

Bab ini menjelaskan mengenai perancangan dan pembuatan sistem

yang membahas mengenai perangkat keras (*hardware*), perangkat lunak (*software*). Perangkat keras (*hardware*) yang di bahas meliputi modul *microcontroller* Atmega32, modul Bluetooth HC-05 dan juga *Mobile Phone Android*. Sedangkan pembahasan perangkat lunak meliputi *flowchart* yaitu program yang digunakan untuk mengatur semua proses alat.

BAB IV : HASIL DAN PEMBAHASAN

Bab ini menjelaskan tentang pengujian sistem baik *hardware* maupun *software*. Pengujian *hardware* meliputi pompa udara, *valve*, sensor MPX5100DP, modul Bluetooth HC-05 serta modul *microcontroller* Atmega32 yang berfungsi sebagai otak kerja dari alat. Pengujian *software* meliputi program *microcontroller* Atmega32 dengan masing-masing *hardware* dan juga program pada *Mobile Phone Android*.

BAB V : PENUTUP

Bab ini berisi kesimpulan yang menjawab pertanyaan dalam perumusan masalah dan beberapa saran yang bermanfaat dalam pengembangan lebih lanjut dari Tugas Akhir ini.

BAB II

LANDASAN TEORI

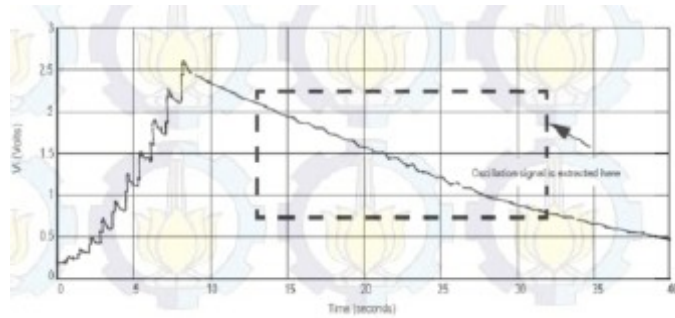
2.1 Tekanan Darah

Tekanan darah adalah kekuatan yang memungkinkan darah mengalir dalam pembuluh darah untuk beredar dalam seluruh tubuh. Darah berfungsi sebagai pembawa oksigen serta zat-zat lain yang dibutuhkan oleh seluruh jaringan tubuh supaya dapat hidup dan dapat melaksanakan masing-masing tugasnya.

Tekanan Darah Sistolik (TDS) menunjukkan tekanan pada arteri bila jantung berkontraksi (denyut jantung) atau tekanan maksimum dalam arteri pada suatu saat. TDS dinyatakan oleh angka yang lebih besar jika dibaca pada alat pengukur tekanan darah. TDS normal 90-120 mmHg. Tekanan Darah Diastolik (TDD) menunjukkan tekanan darah dalam arteri bila jantung berada dalam keadaan relaksasi di antara dua denyutan.

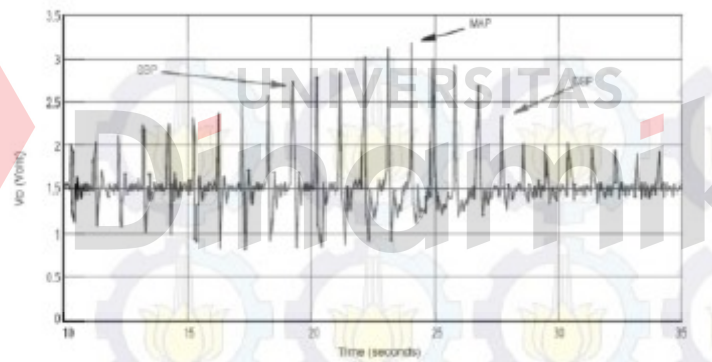
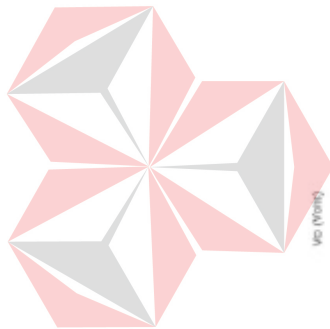
Mengukur nilai tekanan darah dapat menggunakan metode, metode korotkoff dan osilometri. Metode korotkoff adalah metode yang sering digunakan dokter untuk memeriksa pasien dengan menggunakan tensi meter air raksa. Metode ini pengukuran nilai tekanan darah dengan memanfaatkan bunyi denyutan nadi yang disebut detak korotkoff pada saat pengukuran darah.

Metode osilometri adalah metode pembacaan nilai tekanan darah sistolik dan diastolic dengan mengukur amplitudo dari perubahan tekanan yang terjadi dalam cuff selama cuff dipompa hingga udara keluar seluruhnya dari dalam cuff. Contoh proses pengukuran tekanan darah dengan menggunakan metode osilometri adalah sebagai berikut :



Gambar 2.1. Contoh Hasil Sinyal *Output* dari Sensor Tekanan

Pada gambar 1. dimana merupakan contoh sinyal *output* tegangan dari sensor berdasarkan variable waktu saat *handcuff* dipompa pada tekanan tertentu dan dilepas sampai udara terbuang dari *handcuff*.



Gambar 2.2. Contoh Letak sinyal Hasil Ekstraksi

Sinyal hasil ekstraksi seperti gambar 2. diatas dapat ditentukan posisi tekanan Diastolik dan tekanan Sistolik. Berdasar data yang didapat, terdapat 2 pendapat tentang bagaimana mendapatkan tekanan sistolik dan diastolik pada sinyal hasil Ekstraksi:

- A. Tekanan sistolik dapat dihitung dengan membagikan nilai-nilai disebelah kiri MAP (Mean Arterial Pulse). Sedangkan tekanan diastolik dapat dihitung dengan membagikan nilai-nilai puncak di sebelah kanan MAP.

- B. Tekanan sistolik dapat dihitung dengan mengkalikan 0.6 dengan nilai puncak (MAP) sedangkan tekanan diastolik dapat dihitung dengan mengkalikan 0.8 dari nilai Puncak.

Dari kedua pendapat tersebut, Soot W. N well, Ipswich, dan Mass (1993) menyimpulkan bahwa tekanan sistolik didapatkan dari nilai tekanan udara terukur setelah 10 detik dari nilai tekanan udara puncak yang dipompakan dan 5 detik kemudian sebagai tekanan darah diastolik, dan tekanan udara puncak adalah dibatasi pada tekanan 160mmHg.

2.2 Tensimeter

Tensi meter adalah alat di bidang medis yang digunakan untuk mengukur jumlah tekanan darah pada manusia. Alat pengukur tekanan darah sering juga disebut dengan sebutan *sphygmomanometer*. Alat ukur tekanan darah tersebut pertama kali diperkenalkan oleh seorang dokter bedah asal Rusia bernama dr.Nikolai Korotkov. Saat itu jenis tensi meter yang digunakan menggunakan cairan raksa untuk menentukan standar emas di dalam proses pengukuran tensi darah.

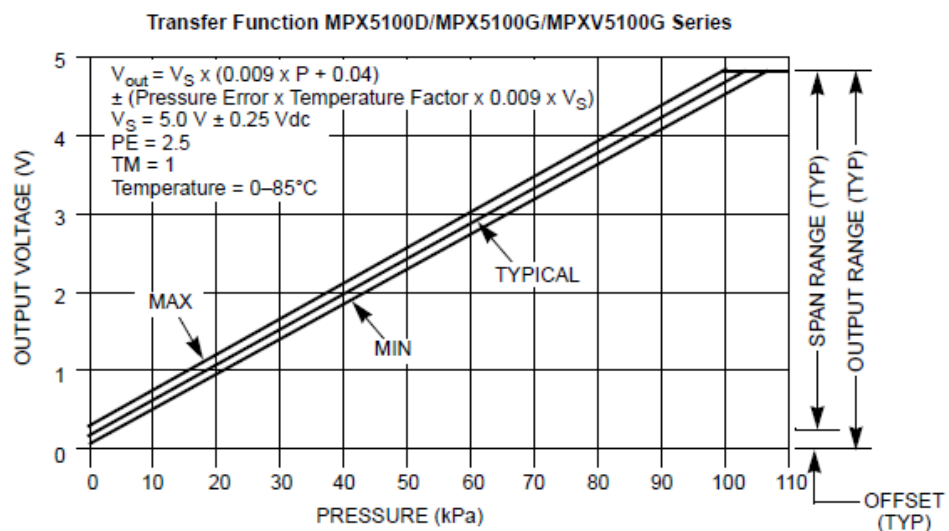


Gambar 2.3. Tensimeter Digital

Tensi meter atau lebih dikenal dengan nama *sphygmomanometer* terdiri atas bagian-bagian berupa pompa udara, penyumbat aliran udara yang bisa diputar, kantong karet yang terbungkus dengan kain serta pembaca tekanan. Saat ini terdapat dua jenis tensimeter yang digunakan dalam memantau tekanan darah, yaitu tensimeter digital dan tensimeter manual/raksa (Soot W, Ipswich, dan Mass, 1993). Dalam perancangan alat pengukur tekanan darah ini penulis menggunakan tensimeter digital sebagai acuan.

2.2.1 MPX 5100 DP

Sensor tekanan pada alat ukur tensimeter digital menggunakan sensor MPX 5100 DP. Sensor ini biasa digunakan untuk mensensing sebuah tekanan fluida yang membutuhkan ketelitian yang sangat akurat. Seperti tekanan udara pada sistem stasiun cuaca, tekanan fluida darah pada alat *blood pressure*. Berikut



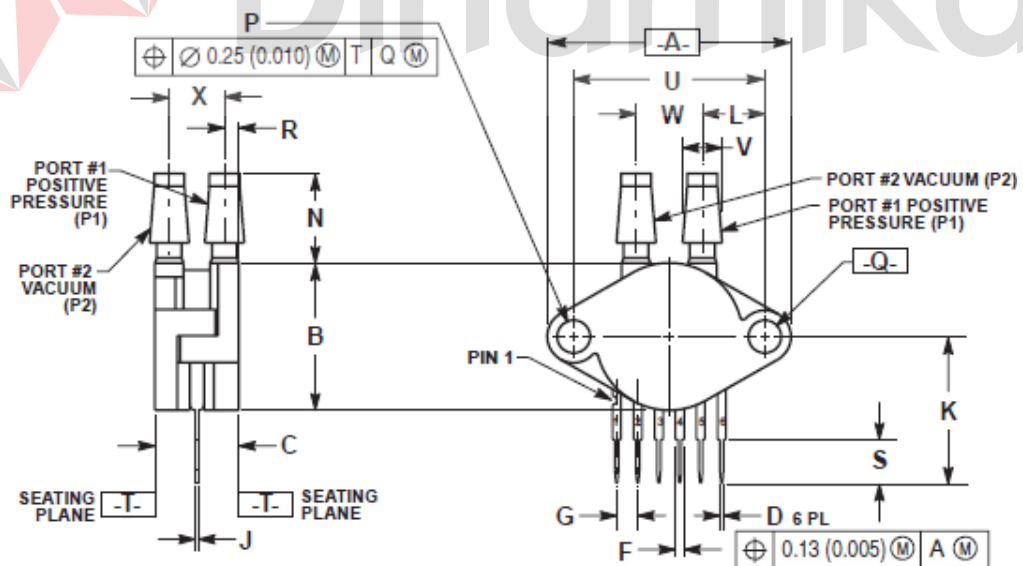
merupakan sinyal *output* sensor terhadap tekanan *input* yang diberikan.

Gambar 2.4. Tabel kebenaran *Input/Output* sensor

Seperti gambar diatas output dari sensor ini memiliki rentang suhu 0 ° hingga 85 ° C. Sensor tekanan MPX5100DP merupakan tranduser yang terbuat dari bahan silicon dan dirancang untuk berbagai aplikasi terutama yang menggunakan mikrokontroler. Sensor ini dilengkapi dengan *chip signal conditioned, temperature compensated* dan *calibrated* dengan *range* pengukuran dengan spesifikasi pada tabel (<http://www.alldatasheet.com /MPX5100.html>, 2012).

Tabel 2.1. Tabel *output* MPX5100DP skala volt,Kpa,dan decimal.

Sinyal	Analog	Sensor Digital
<i>Input</i> Sensor Tekanan MPX 5100 DP	<i>Output</i> Sensor Tekanan MPX 5100 DP	ADC 10 BIT Mikrokontroler
0 – 100 Kpa	0 – 5 Volt	0 -1023 Desimal



Gambar 2.5. Datasheet MPX5100DP

2.2.2 Valve

Valve adalah sebuah perangkat yang mengatur, mengarahkan atau mengontrol aliran dari suatu cairan (gas, cairan, padatan terfluidisasi) dengan membuka, menutup, atau menutup sebagian dari jalan alirannya (R.W. Zappe, 1998). *Valve* akan terbuka katup bila diberi tegangan sebesar 0V pada *port* positif dan sebesar 0V pada *port* negatif, sedangkan katup akan tertutup bila diberi tegangan sebesar 5V pada *port* positif dan sebesar 0V pada *port* negatif. Contoh akrab lainnya termasuk katup kontrol gas di kompor, katup kecil yang dipasang di kamar mandi dan masih banyak lagi. *Valve* dapat dioperasikan secara manual, baik oleh pegangan, tuas pedal dan lain-lain. Selain dapat dioperasikan secara manual katup juga dapat dioperasikan secara otomatis dengan menggunakan prinsip perubahan aliran tekanan, suhu dll. Perubahan ini dapat mempengaruhi diafragma, pegas atau piston yang pada gilirannya mengaktifkan katup secara otomatis. Penulis menggunakan *valve* 5V menjadi aktuator alat pengukur tekanan darah ini.



Gambar 2.6. Valve 5v

2.2.2 Pompa DC 6V

Pompa DC merupakan alat pemompa elektrik yang dapat memompakan udara dengan *input* tegangan DC 6V. Alat ini dapat membantu Anda mendapatkan tekanan angin yang diinginkan untuk memberikan tekanan udara pada

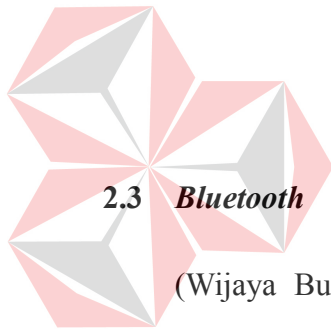
manset/*handcuff* untuk diukur sensor. Pompa ini mampu memberikan tekanan udara yaitu 3500pa - 4500pa. Pipa aliran angin akan dengan cepat memompa angin agar proses pengisian menjadi lebih cepat serta efisien (Adiludung, 2011).



Gambar 2.7.

Pompa DC

6V



2.3 *Bluetooth*

(Wijaya Budi Agung, 2010) *Bluetooth* adalah spesifikasi industri untuk

jaringan kawasan pribadi (*personal area networks* atau PAN) tanpa kabel.

Bluetooth menghubungkan dan dapat dipakai untuk melakukan tukar-menukar

informasi di antara peralatan-peralatan. *Bluetooth* beroperasi dalam pita frekuensi

2,4 Ghz dengan menggunakan sebuah *frequency hopping traceiver* yang mampu

menyediakan layanan komunikasi data dan suara secara *real time* antara *host -*

host bluetooth dengan jarak terbatas. Kelemahan teknologi ini adalah

jangkauannya yang pendek dan kemampuan transfer data yang rendah. HC-05

merupakan modul *bluetooth* yang digunakan dalam pembuatan alat ini.

2.3.1 HC-05

HC-05 Adalah sebuah modul *Bluetooth* SPP (Serial Port Protocol) yang mudah digunakan untuk komunikasi serial wireless (nirkabel) yang mengkonversi *port* serial ke *Bluetooth*. HC-05 menggunakan modulasi *bluetooth* V2.0 + EDR (Enhanced Data Rate) 3 Mbps dengan memanfaatkan gelombang radio berfrekuensi 2,4 GHz. Modul ini dapat digunakan sebagai slave maupun master. HC-05 memiliki 2 mode konfigurasi, yaitu AT mode dan Communication mode. AT mode berfungsi untuk melakukan pengaturan konfigurasi dari HC-05. Sedangkan Communication mode berfungsi untuk melakukan komunikasi *bluetooth* dengan piranti lain.

Dalam penggunaannya, HC-05 dapat beroperasi tanpa menggunakan driver khusus. Untuk berkomunikasi antar *Bluetooth*, minimal harus memenuhi dua kondisi berikut :

1. Komunikasi harus antara master dan slave.
2. Password harus benar (saat melakukan pairing).

Jarak sinyal dari HC-05 adalah 30 meter, spesifikasi dari HC-05 adalah :

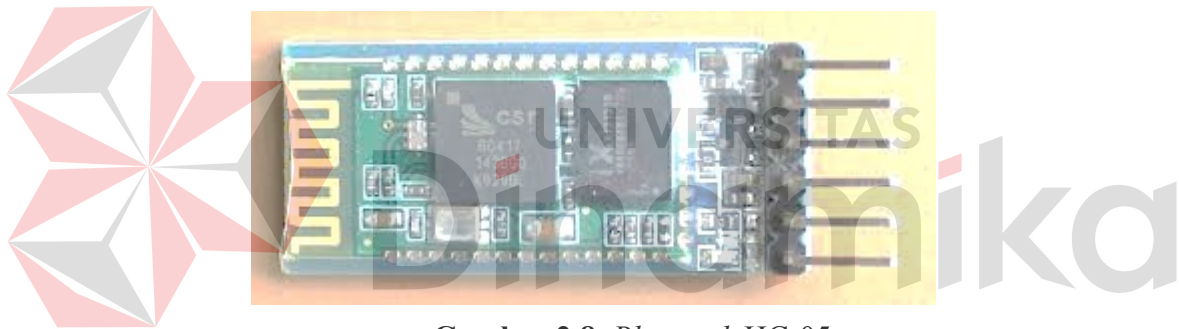
Hardware:

- Sensitivitas -80dBm (Typical)
- Daya transmit RF sampai dengan +4dBm.
- Operasi daya rendah 1,8V – 3,6V I/O.
- Kontrol PIO.
- Antarmuka UART dengan baudrate yang dapat diprogram.
- Dengan antena terintegrasi.

Software :

- Default baudrate 9600, Data bit : 8, Stop bit = 1, Parity : No Parity,
Mendukung baudrate : 9600, 19200, 38400, 57600, 115200, 230400 dan 460800.
- Auto koneksi pada saat device dinyalakan (default).
- Auto reconnect pada menit ke 30 ketika hubungan putus karena range koneksi.

(<http://www.alldatasheet.com/datasheet/3042/MOTOROLA/HC05.html>)

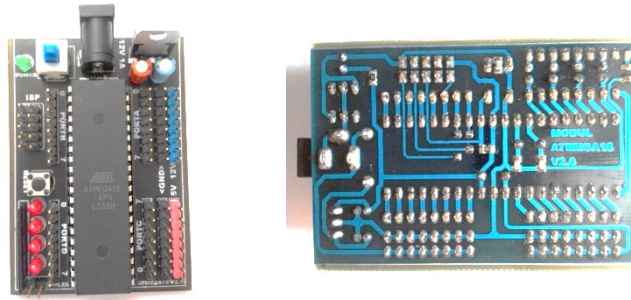


Gambar 2.8. Bluetooth HC-05

2.4 Mikrokontroler

Mikrokontroler adalah otak dari suatu sistem elektronika seperti halnya mikroprosesor sebagai otak komputer. Didalam mikrokontroler terdapat memori dan *Port Input/Output* dalam suatu kemasan *IC (Integrated Circuit)*. Kemampuannya yang *programmable*, fitur yang lengkap (*ADC (Analog to Digital Converter) internal*, *EEPROM (Electrically Erasable Programmable Read-Only Memory) internal*, *Port I/O*, Komunikasi Serial, dll), dan juga harga yang terjangkau memungkinkan mikrokontroler digunakan pada berbagai system

elektronis, seperti pada robot, otomatisasi industri, sistem alarm, peralatan telekomunikasi, hingga peralatan rumah tangga.



Gambar 2.9. *Minimum Sistem ATmega32*

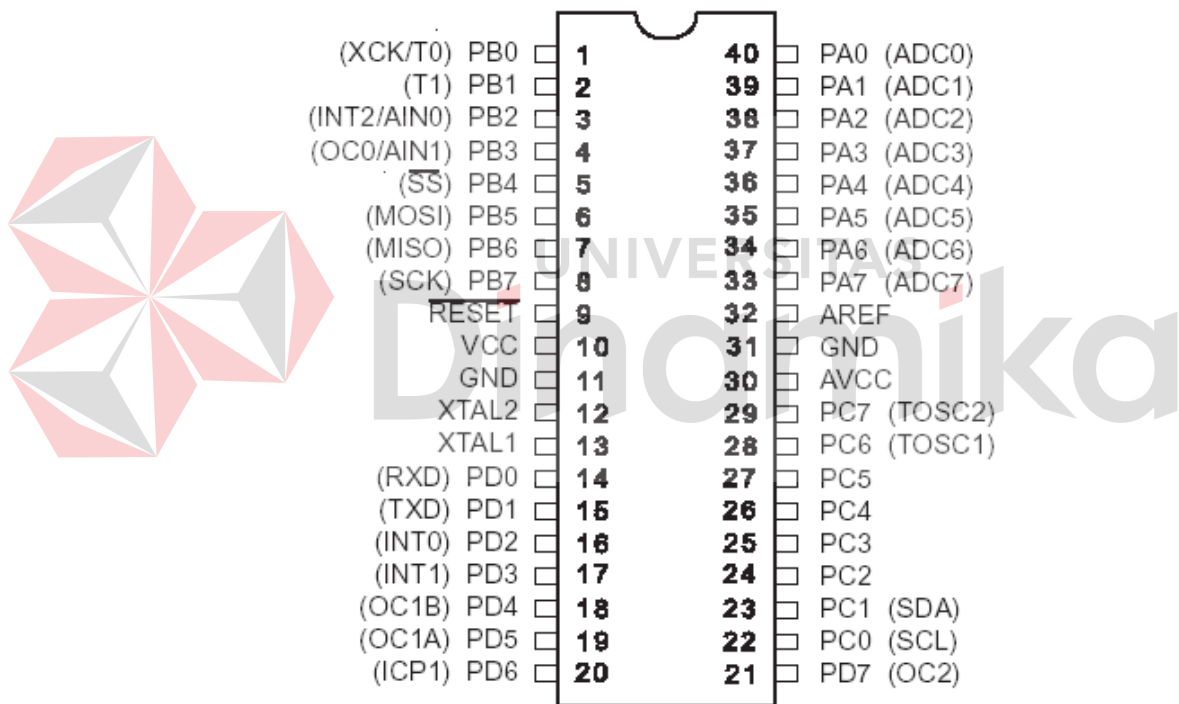
Sebuah mikrokontroler umumnya telah berisi komponen pendukung sistem minimal micropro, yakni memori dan *interface* I/O. Untuk mengontrol robot, maka digunakan mikrokontroler dengan pertimbangan faktor ukuran yang relatif kecil sehingga cocok untuk pengontrol robot dan peralatan-peralatan elektronika. Sistem mikrokontroler lebih banyak melakukan pekerjaan-pekerjaan sederhana yang penting seperti mengendalikan motor, saklar, resistor variabel, atau perangkat elektronik lain.

Fitur-fitur yang dimiliki oleh mikrokontroler AVR ATmega32 adalah sebagai berikut:

1. Saluran I/O ada 32 buah, yaitu *Port A*, *Port B*, *Port C*, dan *Port D*.
2. ADC 10 bit sebanyak 8 channel.
3. Tiga buah *Timer / Counter* dengan kemampuan pembanding.
4. CPU (*Central Processing Unit*) yang terdiri atas 32 buah register.

5. 131 instruksi handal yang umumnya hanya membutuhkan 1 siklus *clock*.
6. Dua buah *timer / counter* 8 bit.
7. Satu buah *timer / counter* 16 bit.
8. Unit interupsi *internal* dan *eksternal*.
9. 4 channel PWM (*Pulse with Modulation*)

(www.atmel.com.Datasheet AVR ATmega32)

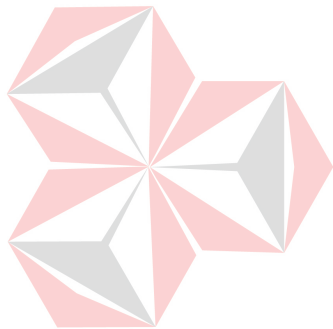


Gambar 2.10. Konfigurasi *pin* ATmega32

2.5 LCD (*Liquid Crystal Display*)

LCD digunakan sebagai *output device* yang mampu menampilkan huruf, angka maupun karakter tertentu. Pengaksesan LCD dilakukan dengan mengirimkan kode perintah seperti bentuk kursor, operasi kursor dan bersih layar

dan mengirimkan kode data yang merupakan kode ASCII dari karakter yang akan ditampilkan.



Gambar 2.11.LCD 16 x 2

	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	25 (1)		0	@	P	`	P				-	9	ε	α	p	
xxxx0001	(2)		!	1	A	Q	a	q			.	7	ç	ä	q	
xxxx0010	(3)		"	2	B	R	b	r			「	イ	ツ	×	þ	ø
xxxx0011	(4)		#	3	C	S	c	s			」	ウ	テ	ε	»	
xxxx0100	(5)		\$	4	D	T	d	t			、	エ	ト	μ	Ω	
xxxx0101	(6)		%	5	E	U	e	u			.	オ	ナ	1	ε	Ü
xxxx0110	(7)		&	6	F	V	f	v			ヲ	カ	ニ	ヨ	ρ	Σ
xxxx0111	(8)		'	7	G	W	g	w			ア	キ	ヲ	ラ	q	π
xxxx1000	(1)		(8	H	X	h	x			イ	ク	ネ	リ	」	Σ
xxxx1001	(2))	9	I	Y	i	y			ウ	ケ	ノ	ル	「	γ
xxxx1010	(3)		*	:	J	Z	j	z			エ	コ	ハ	レ	j	ç
xxxx1011	(4)		+	;	K	[k	[オ	サ	ヒ	ロ	*	π
xxxx1100	(5)		,	<	L	¥	l	¥			ヤ	シ	フ	ワ	φ	π
xxxx1101	(6)		-	=	M]	m]			ユ	ズ	ヘ	ン	も	÷
xxxx1110	(7)		.	>	N	^	n	^			ヨ	セ	ホ	°	ñ	
xxxx1111	(8)		/	?	O	_	o	_			ッ	ツ	マ	°	ö	■

Tabel 2.2. Karakter-karakter pada LCD

Pengaksesan LCD dari Mikrokontroler menggunakan jalur alamat dan data. Terdapat dua jalur alamat *input* yang digunakan, yaitu sebagai *Intruction Input* dan *Data Input*. Beberapa LCD memiliki *pin-out* yang berbeda-beda, tetapi mempunyai diskripsi *pin* yang sama. *Pin-pin* tersebut adalah sebagai berikut.

1. V_{ss} : *Ground*
2. VCC : *+5V Power Supply*
3. VEE : *Power Supply* pengatur kecerahan
4. RS : $RS = 0$, memilih *Intruction Input*
 $RS = 1$, memilih *Data Input*

5. R/W : $R/W = 0$, tulis
 $R/W = 1$, baca
6. E : *Enable*
7. $DB7 - DB 0$: 8 bit *data bus*

(www.alldatasheet.com/datasheet/571342/LUMEX/LCM-S01602DTRM.html)

2.6 Downloader USBASP

Downloader USBasp adalah suatu perangkat yang digunakan untuk menuliskan program (mendownload) pada mikrokontroler ATMEL AVR. Dengan menggunakan perangkat ini kita bisa mengisi file hex yang telah diCompile (di ciptakan) oleh beberapa software seperti Codevision AVR. Banyak kelebihan yang kita miliki dengan menggunakan perangkat ini diantaranya:

1. Kecepatan mengisi program yaitu 5 kBytes/s

2. Fasilitas SCK digunakan untuk mendownload target dengan kecepatan rendah (kurang dari 1,5MHz).
3. Tidak memerlukan supply tambahan dan dapat digunakan untuk mensupply power dari mikrokontroller.
4. Bisa digunakan untuk berbagai macam platform seperti Linux, Mac OS, Windows XP, Windows Vista dan Windows 7.



Gambar 2.12. *Downloader USBASP*

2.7 Android

Android adalah sistem operasi untuk telepon seluler yang berbasis *Linux*. *Android* merupakan sebuah perusahaan software kecil yang didirikan pada bulan Oktober 2003 di Palo Alto, California, USA. Didirikan oleh beberapa senior di beberapa perusahaan yang berbasis IT & Communication; Andy Rubin, Rich Miner, Nick Sears dan Chris White. Menurut Rubin, *Android Inc* didirikan untuk mewujudkan *Mobile* device yang lebih peka terhadap lokasi dan preferensi pemilik. Dengan kata lain, *Android Inc*, ingin mewujudkan *Mobile* device yang lebih mengerti pemiliknya. Terdapat banyak macam aplikasi yang menyediakan pengguna untuk membuat aplikasi pada *mobile android* itu antara lain JDK,

Eclipse, Android SDK, Basic4Android. Dan yang digunakan dalam alat ini adalah aplikasi Basic4Android.

2.7.1 Basic4Android

Basic4Android sendiri merupakan *tools* RAD yang terdiri dari *framework*, *library*, dan IDE yang terintegrasi dengan java dan *android* SDK. Basic4Android didesain sedemikian rupa sehingga memudahkan *developer* untuk mengembangkan aplikasi android menggunakan bahasa Visual Basic dan IDE yang mudah untuk digunakan (<http://www.basic4ppc.com/android/forum>).

Basic4Android juga menyediakan *library-library* lengkap untuk mengakses berbagai macam fitur yang ada pada smartphone seperti sensor, kamera, GPS, dan sebagainya. Adapun beberapa kelebihan dan fitur yang dimiliki oleh Basic4Android adalah :

1. *Simple* dan *Powerfull* RAD (*Rapid Application Development*) *tools* untuk mengembangkan aplikasi *native android* tanpa harus menghabiskan waktu mempelajari pemrograman Eclipse/Java.
2. IDE (*Integrated Development Environment*) lengkap yang fokus 100% pada pengembangan aplikasi Android.
3. Di *compile* kedalam native bytecode, tidak ada tambahan runtime library yang dibutuhkan. APK file yang dihasilkan sama dengan APK yang dihasilkan ketika anda membuat aplikasi menggunakan Eclipse/Java.

4. Performa dari aplikasi yang dihasilkan sama dengan aplikasi yang dibuat dengan Java.
5. Sintaks sama dengan Visual Basic.
6. Mendukung semua *android phone* dan *tablet* dari versi 1.6 sampai dengan 4x.
7. Mendukung semua *android core* (GPS, SQL Database, Widgets, Live Wallpaper, Bluetooth, USB, Web Services, Camera, JSON, XML, Excel CSV, Multitouch, NFC, Views Animation, Push Notification, AdMob, OpenGL, Graphic and Chart, Sensor)

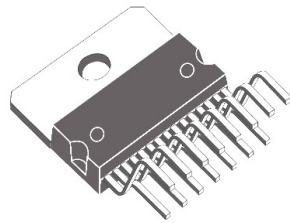
2.8 Motor Driver

Driver motor adalah sebuah rangkaian elektronika yang digunakan untuk mengerakkan kecepatan dari putaran sebuah motor dan menentukan gerak dari motor tersebut, jenis-jenis *driver motor* sangat banyak sekali semua itu bergantung pada jenis motor yang akan dikendalikan. Adapun beberapa jenis dari *driver motor* yaitu (www.toko-elektronika.com/tutorial/pcinterfacing.htm).

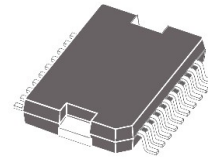
2.8.1 L298H

L298 adalah jenis IC driver motor yang dapat mengendalikan arah putaran dan kecepatan motor DC ataupun Motor stepper. Mampu mengeluarkan output tegangan untuk Motor dc dan motor stepper sebesar 50 volt. IC l298 terdiri dari *Transistor-Transistor Logik (TTL)* dengan gerbang nand yang memudahkan dalam menentukan arah putaran suatu motor dc dan motor stepper. Dapat mengendalikan 2 untuk motor dc namun pada hanya dapat mengendalikan satu

motor *stepper*. Penggunaannya paling sering untuk robot *line follower*. Bentuknya yang kecil memungkinkan dapat meminimalkan pembuatan robot *line follower*.

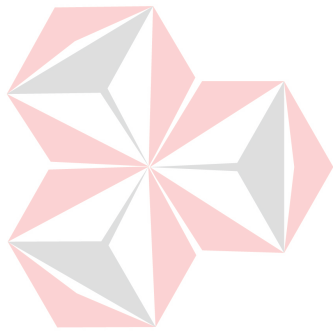


Multiwatt15



PowerSO20

Gambar 2.13. L298H (<https://www.sparkfun.com>, 2014)



UNIVERSITAS
Dinamika

BAB III

METODE PENELITIAN

Metodologi penelitian yang digunakan dalam perancangan sistem ini antara lain studi kepustakaan. Dari data-data yang diperoleh, selanjutnya dilakukan sebuah perancangan sistem yang terdiri dari perancangan dan pembuatan perangkat keras (*hardware*), setelah desain *hardware* selesai dilakukan juga proses perancangan dan pembuatan perangkat lunak (*software*) yang nantinya digunakan sebagai percobaan pada *hardware* mikrokontroler dan juga *hardware mobile phone android*.

Setelah mengetahui masalah apa yang akan dihadapi melalui studi kepustakaan, dilakukanlah perumusan tujuan yang berfungsi sebagai target ketercapaian untuk penyelesaian dari masalah yang telah didapat. Kemudian dilakukan perancangan alat yang meliputi rancang elektrik dan mekanik, kemudian rancang perangkat lunak. Rancang elektrik adalah membuat perancangan sistem elektrik dan merangkai komponen-komponen elektrik.

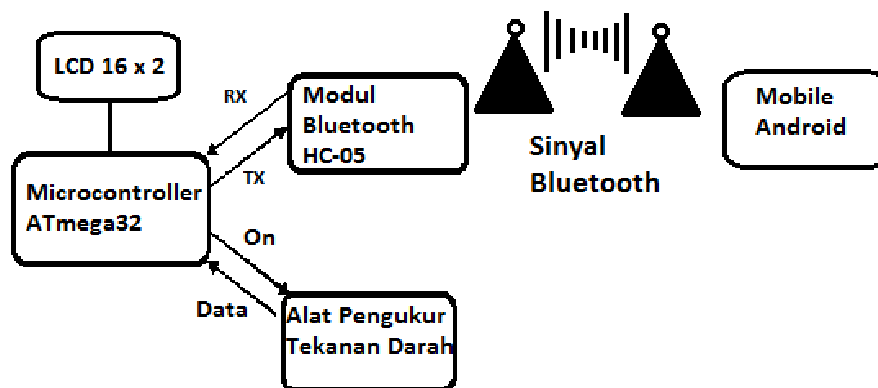
Rancang mekanik adalah membuat rancangan alat secara mekanikal. Pada perancangan elektrik akan digunakan beberapa komponen penting yaitu sensor tekanan MPX5100DP dan mikrokontroler ATmega32. Kemudian terakhir dilakukan perancangan perangkat lunak yaitu dibuat sebuah pemrograman mikro untuk melakukan otomatisasi dari alat. Setelah tiga tahap perancangan dan pembuatan selesai, maka ketiga langkah diujikan. Jika salah satu langkah tidak sesuai dengan perancangan, maka langkah pembuatan diulangi kembali.

Kemudian jika sesuai dengan perancangan dilakukan penggabungan antar ketiganya yaitu integrasi alat.

Setelah dilakukan integrasi alat, maka dilakukan pengujian seluruh komponen yang sudah terintegrasi. Jika pengujian tidak sesuai, maka dilakukan langkah sebelumnya yaitu integrasi alat. Jika pengujian menghasilkan data sesuai, maka dilakukan pengambilan data dan pengujian data atau kalibrasi.

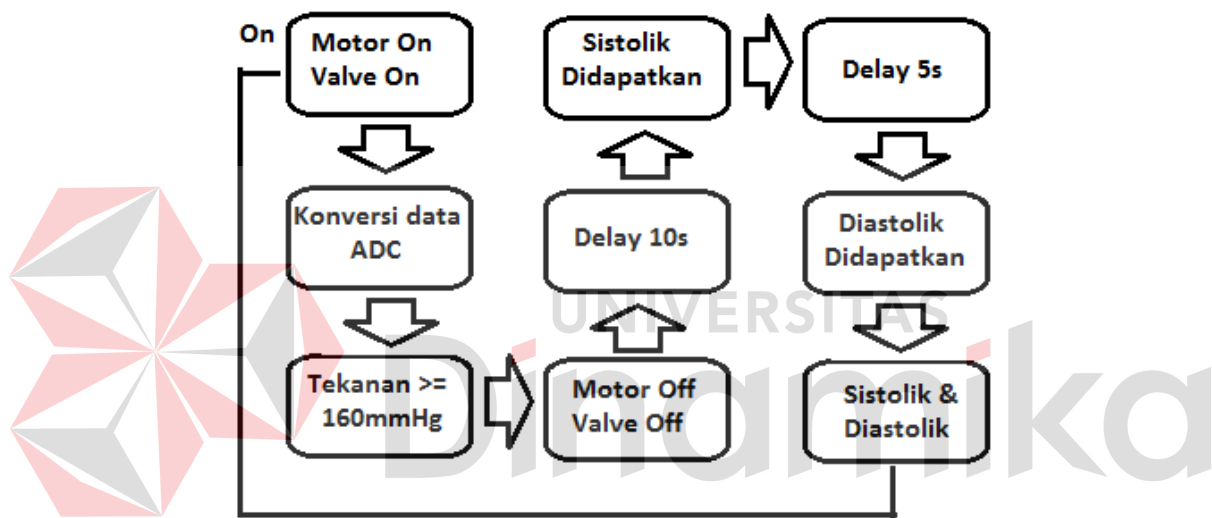
Setelah mendapatkan data dari alat penelitian, kemudian dilakukan analisis dari data yang didapat dan dikembalikan pada tujuan penelitian. Jika masih belum menjawab tujuan dari penelitian maka akan dilakukan perancangan ulang. Dan jika telah sesuai dengan tujuan maka penelitian telah selesai.

Untuk mendapatkan hasil yang dikehendaki dibutuhkan suatu rancangan agar dapat mempermudah dalam memahami sistem yang akan dibuat, oleh karena itu akan dibuat seperti gambar 3.1.



Gambar 3.1.Blok diagram keseluruhan

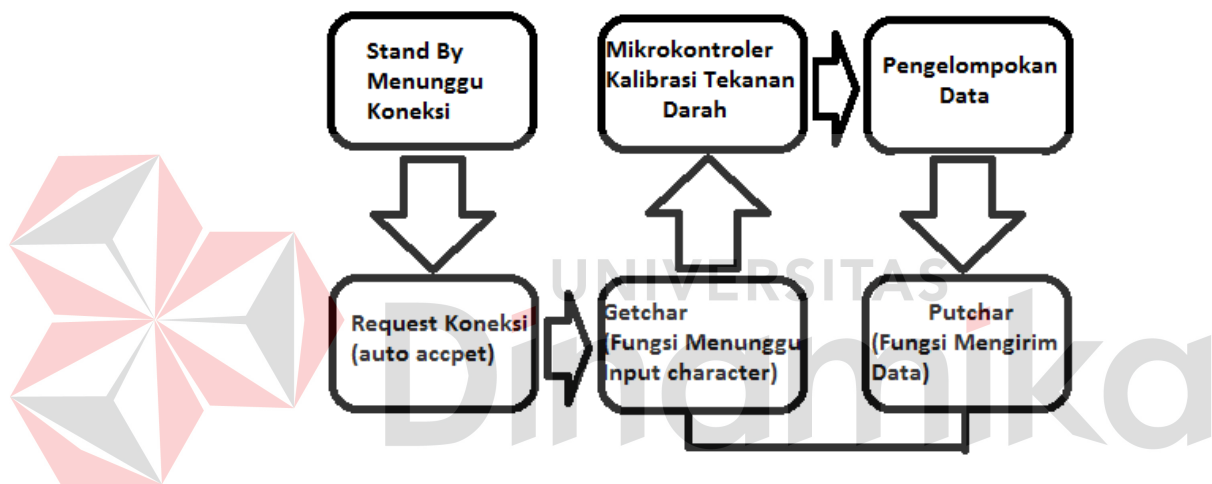
Blok diagram pada Gambar 3.1 adalah blok diagram sistem secara keseluruhan. Sistem ini terdiri dari sebuah *microcontroller* ATmega32 sebagai otak proses kerja alat, alat pengukur tekanan darah, modul *Bluetooth* HC-05, dan LCD 16 x 2. Berikut merupakan blok diagram alat pengukur tekanan darah pada gambar 3.2.



Gambar 3.2. Blok diagram alat pengukur tekanan darah

Alat pengukur tekanan darah yang terdiri dari pompa DC, *valve*, *handcuff*, MPX5100DP, motor driver. Alat pengukur tekanan darah dimulai dengan pemompaan udara kepada *handcuff* yang dilakukan oleh POMPA DC dan dengan aktif nya *valve* agar udara yang dipompa tidak keluar dan tetap berada di *handcuff*. MPX5100DP akan terus mengukur kadar tekanan udara dalam *handcuff* hingga tekanan udara terukur lebih besar dari 160 mmHg. Bila tekanan udara sudah mencapai 160 mmHg pompa akan berhenti memberikan udara, dan

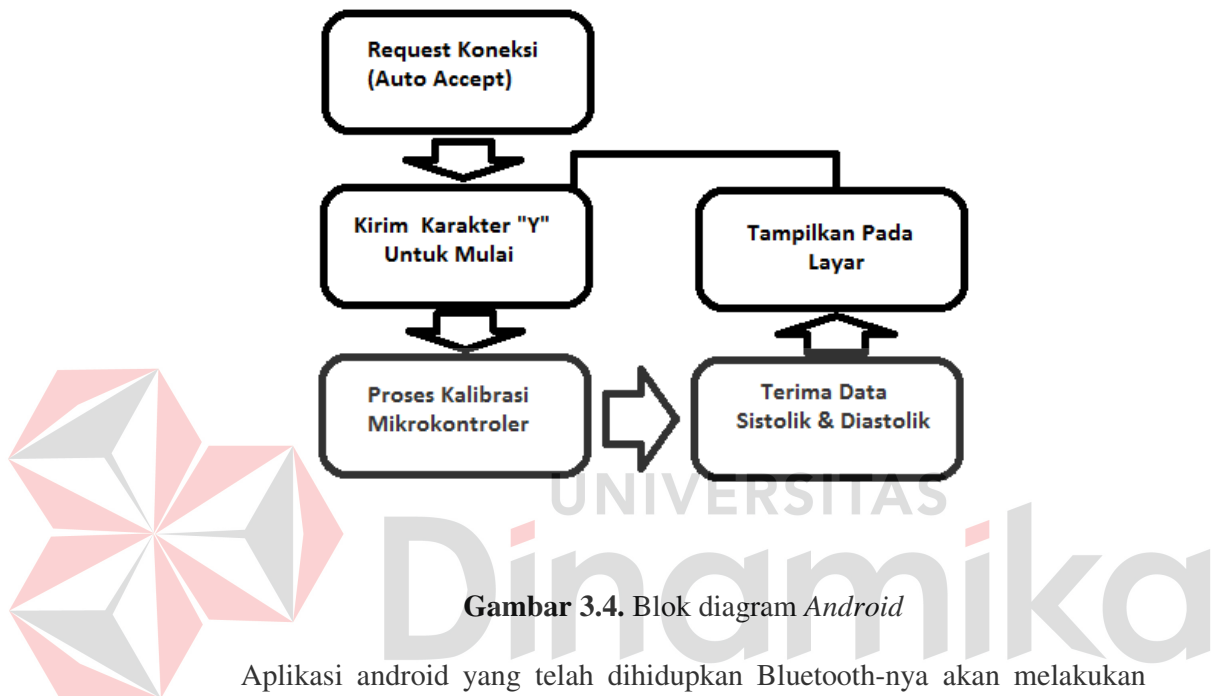
pencatatan data tekanan sistolik dan diastolik pun dimulai. 10s setelah pompa mati akan dicatat sebagai data sistolik dan 15s setelah pompa mati akan dicatat sebagai diastolik, metode pencatatan tersebut merupakan metode osilometri. Setelah data sistolik dan diastolik didapatkan *valve* akan dinon-aktifkan dan udara akan dibuang. Setelah alat pengukur tekanan darah berikut adalah blok diagram dari modul *bluetooth* HC-05 pada gambar 3.3.



Gambar 3.3. Blok diagram HC-05

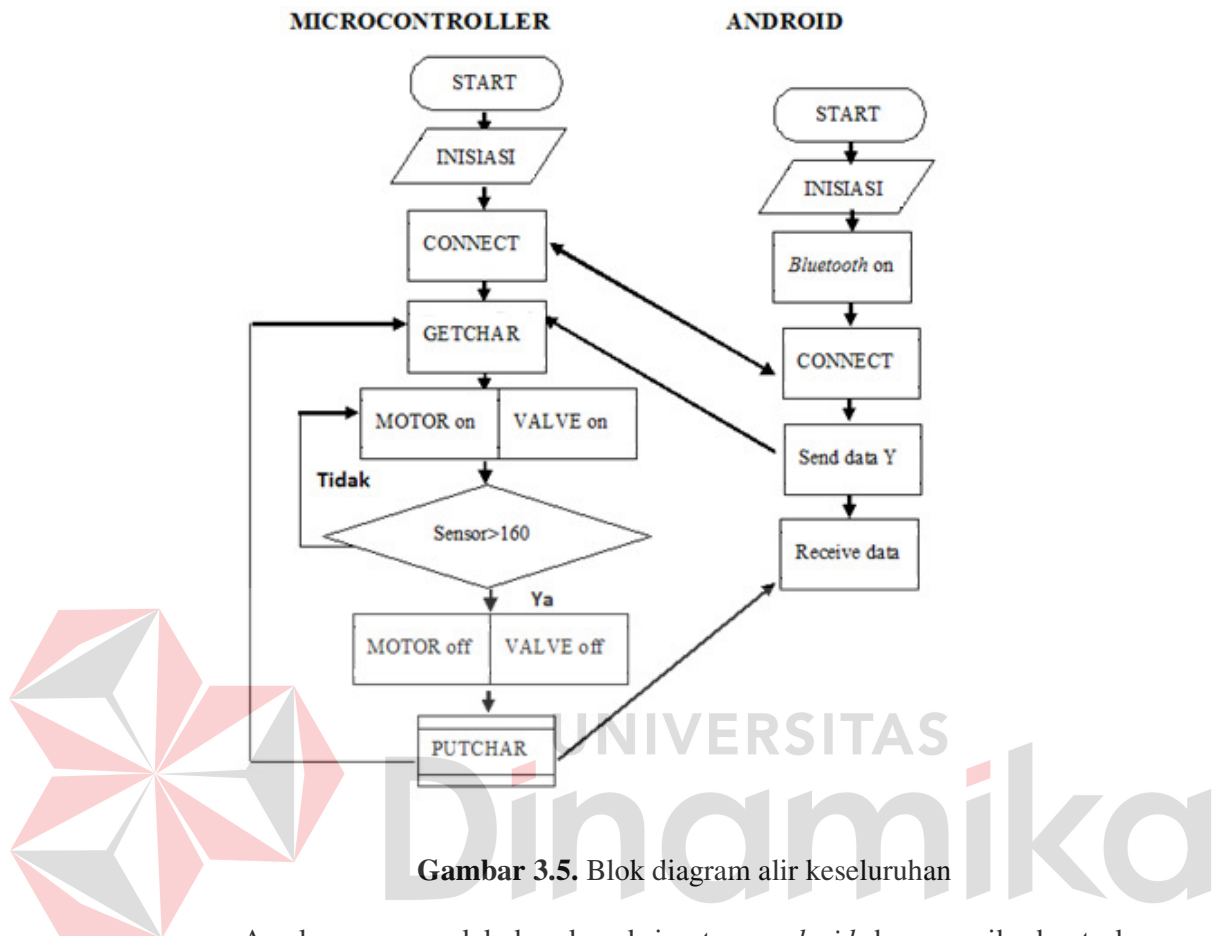
Modul *Bluetooth* akan menunggu hingga ada koneksi dari perangkat lain, Setelah ada permintaan koneksi dengan perangkat lain modul *Bluetooth* akan otomatis menerima permintaan koneksi. Modul *Bluetooth* akan menunggu hingga ada karakter yang dikirimkan oleh perangkat yang telah terhubung menggunakan perintah *getchar* pada mikrokontroler. Setelah ada karakter yang dikirim maka mikrokontroler akan mulai melakukan proses kalibrasi tekanan darah menggunakan alat pengukur tekanan darah. Data hasil kalibrasi tersebut akan

dikirimkan pada perangkat yang terhubung menggunakan perintah `getchar` pada mikrokontroler. Berikut pada gambar 3.4. akan dijabarkan blok diagram pada aplikasi *android*.



Gambar 3.4. Blok diagram *Android*

Aplikasi android yang telah dihidupkan Bluetooth-nya akan melakukan koneksi pada modul *bluetooth* HC-05 dan akan otomatis diterima permintaan koneksi tersebut. User akan mengirimkan karakter “Y” sebagai perintah untuk memulai proses kalibrasi pada alat pengukur tekanan darah. Setelah data hasil proses kalibrasi sudah didapatkan maka android akan menerima data tekanan darah berupa sistolik dan diastolic tersebut dan menampilkan nya pada layar aplikasi. Untuk diagram alir keseluruhan dapat dilihat pada gambar 3.5.



Awalnya user melakukan koneksi antara android dengan mikrokontroler menggunakan HC-05 dengan permintaan request dari android. Setelah android berhasil terhubung dengan mikrokontroler, android memberikan perintah dengan inputan karakter “Y” yang dikirimkan pada mikrokontroler. Mikrokontroler yang menggunakan fungsi *getchar* akan melakukan kalibrasi tekanan darah melalui rangkaian alat pengukur tekanan darah yang dirancang. Setelah data tekanan darah didapatkan maka mikrokontroler akan melakukan pengelompokan data untuk dikirimkan pada android. Setelah pengelompokan data dilakukan maka pengiriman data pada android berupa karakter akan dilakukan. Android akan

menerima data tekanan darah tersebut dan menampilkan nya pada layar untuk interface kepada *user* dan merupakan output data yang diberikan pada *user*.

3.1. Alat dan Bahan Penelitian

Alat dan bahan yang akan digunakan dalam penelitian ini terdiri dari beberapa komponen, alat-alat ini akan digunakan untuk mengukur dan melihat hasil yang akan terjadi terhadap sistem yang akan dibuat oleh peneliti. Adapun alat dan bahan yang akan digunakan sebagai berikut.

3.1.1. Alat Penelitian

Alat yang digunakan untuk menyelesaikan penelitian ini adalah sebagai berikut.

1. Alat pengukur tekanan darah rancangan, digunakan untuk mengukur tekanan darah.
2. *Mobile android*, digunakan untuk menguji pengiriman serial antar *microcontroller* dan *mobile android*serta untuk mengelolah data koordinat.
3. *Driver Motor*, digunakan untuk mengendalikan kecepatan dan arah putaran dari motor DC.
4. *Minimum Sistem*, digunakan untuk menproses semua data pergerakan dan komunikasi *mobile robot*.
5. Penggaris, digunakan untuk mengukur jarak antara alat pengukur tekanan darah dan *android*.
6. Stopwatch, untuk mengukur waktu.

7. *BluetoothHC-95*, digunakan untuk merubah level tegangan pada module wizenet dan mikrokontroler sehingga dapat saling berkomunikasi.
8. *Battery 9V*, digunakan sebagai sumber daya untuk motor *driver*.
9. LCD, interface terhadap *user*.
10. Alat pengukur tekanan darah acuan, digunakan untuk membandingkan alat pengukur tekanan darah rancangan.

3.1.2. Bahan Penelitian

Bahan yang akan diteliti oleh penulis dalam penelitian kali ini meliputi beberapa aspek sebagai berikut.

1. Cara kerja alat pengukur tekanan darah.
2. Komunikasi serial yang terjadi antara alat mikrokontroler dan *android* dalam memberikan data.
3. Tingkat error pengukuran data bila dibandingkan dengan alat pengukur tekanan darah acuan.

3.2. Jalan Penelitian

Tahap-tahap yang akan peneliti lakukan untuk menyelesaikan penelitian ini adalah sebagai berikut.

3.2.1 Perancangan Perangkat Keras (*Hardware*)

Perancangan perangkat keras akan dibahas bagaimana komponen-komponen elektronika yang terhubung pada *hardware* dengan *microcontroller* agar elektronika pendukung dapat bekerja sesuai dengan sistem yang diharapkan

A. Rangkaian *Microcontroller*

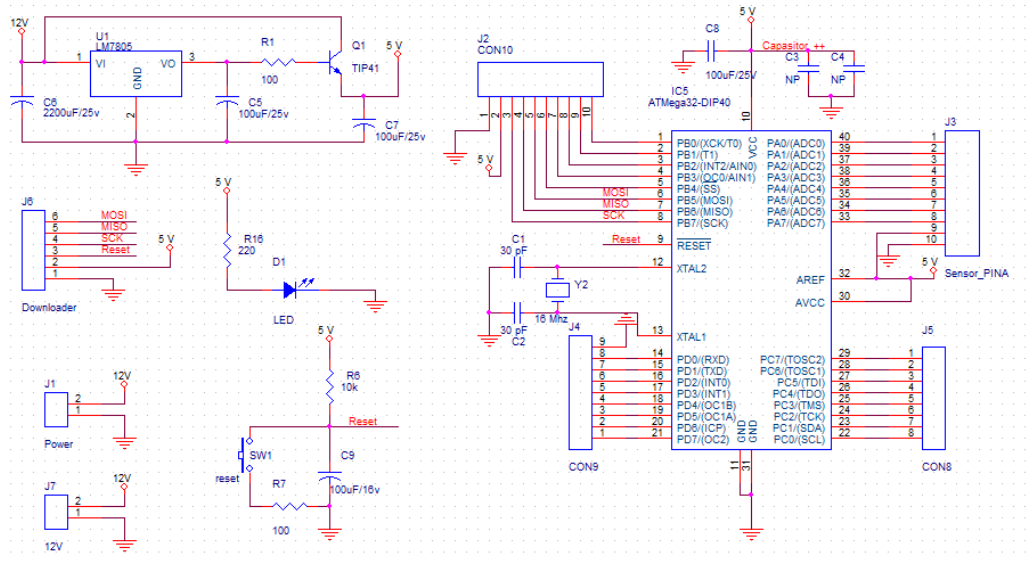
Pada penelitian ini dibuat piranti pengendali menggunakan *microcontroller* keluaran AVR, yaitu ATmega32. Untuk mengaktifkan atau menjalankan *microcontroller* ini diperlukan rangkaian *minimum system*. Rangkaian *minimum system* tersebut terdiri rangkaian *reset*, rangkaian *oscillator*, rangkaian sistem *microcontroller*.

Dalam perancangan perangkat keras *minimum system* ATmega32 terdapat beberapa rangkaian pendukung yaitu rangkaian *reset* dan rangkaian *oscillator*. Pada rangkaian *reset* menggunakan *manual reset*. Pada rangkaian *oscillator* menggunakan komponen kristal 12000MHz sebagai *clk (clock)*.

Untuk menjalankan *microcontroller* dibutuhkan sebuah rangkaian *minimum system* agar *microcontroller* tersebut dapat bekerja dengan baik. Rangkaian *minimum system* terdiri dari rangkaian *reset* dan rangkaian *oscillator*.

Reset pada *microcontroller* ATmega32 terjadi dengan adanya logika *high* “1” selama dua *cycle* pada kaki RST pada *microcontroller* ATmega32. Setelah kondisi *pin* RST kembali *low*, maka *microcontroller* akan menjalankan program dari alamat 0000H. Dalam hal ini *reset* yang digunakan adalah *manual reset*.

Pada *pin* VCC diberi masukan tegangan operasi berkisar antara 4,5 Volt sampai dengan 5,5 Volt. *Pin* RST mendapat masukan dari *manual reset*. Rangkaian *minimum system* dapat dilihat pada Gambar 3.3 berikut.



Gambar 3.6. Rangkaian *minimum system*.

Pin XTAL1 dan XTAL2 merupakan *pinoscillator* bagi *microcontroller* ATmega32. Pin XTAL1 berfungsi sebagai *input* dan XTAL2 sebagai *outputoscillator*. *Oscillator* ini bisa berasal dari kristal atau dari keramik *resonator*. Seperti yang sudah terlihat di atas, *pin* XTAL1 dan XTAL2 dihubungkan dengan komponen XTAL sebesar 12000 MHz. Pada proyek Tugas Akhir ini dibuat rangkaian *oscillatorinternal* yang terbuat dari kristal. Nilai C1 dan C2 masing-masing 100 uF.

Rangkaian I/O dari *microcontroller* mempunyai kontrol direksi yang tiap bitnya dapat dikonfigurasi secara individual, maka dalam perancangan I/O yang digunakan ada yang berupa operasi *port* ada pula yang dikonfigurasi tiap bit I/O. Berikut ini akan diberikan konfigurasi dari I/O *microcontroller* tiap bit yang ada pada masing-masing *port* yang terdapat pada *microcontroller*.

1. Port A, digunakan untuk *input sensor MPX5100DP dengan ADC0*.
2. Port B, digunakan untuk *output LCD*.

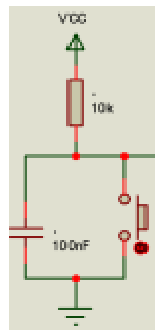
3. Port C, digunakan untuk *outputvalve* dan *pompa DC*.
4. Port D, digunakan untuk *input-output USART*.

Untuk perancangan *interfaceinput* dan *output* pada *microcontroller* yang lebih detail dapat dilihat pada tabel 3.1.

Tabel 3.1. Perancangan *interfaceInput/Output*

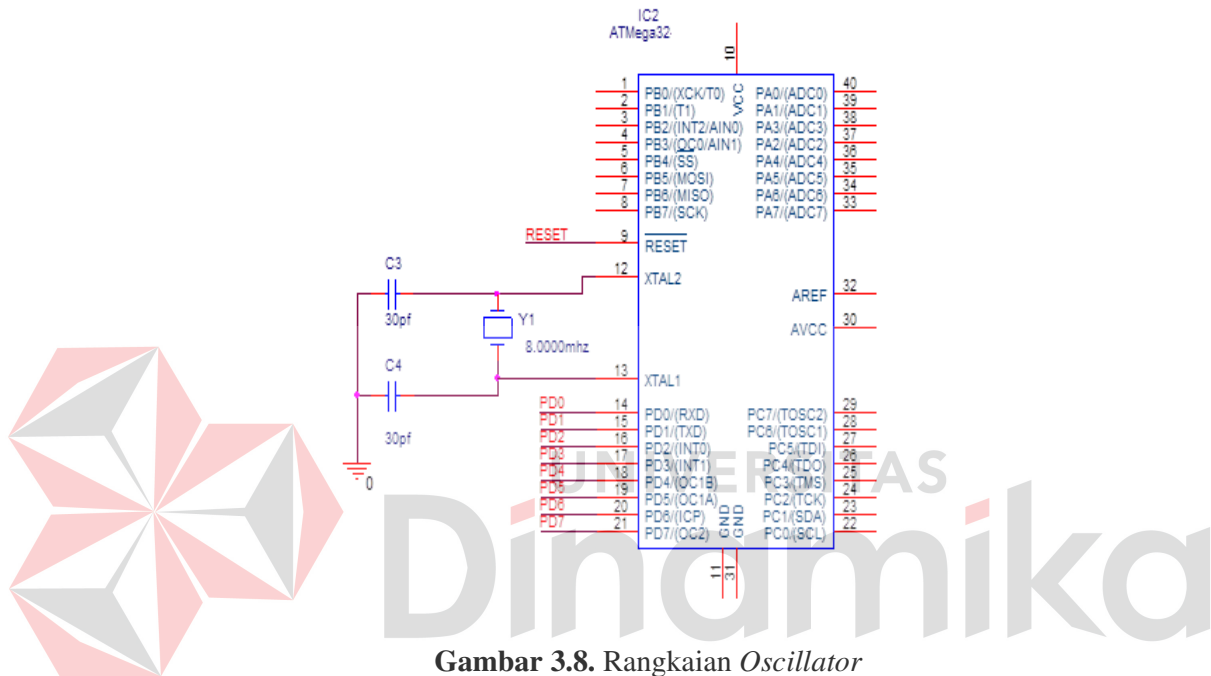
Port	Alokasi Port pada Hardware
ADC0	Input Sensor MPX5100DP
PortB.0	OutputRS LCD
Port B.1	OutputRW LCD
Port B.2	OutputEN LCD
Port B.4	OutputD4 LCD
Port B.5	OutputD5 LCD
Port B.6	OutputD6 LCD
Port B.7	OutputD7 LCD
Port C.0	OutputPompa DC
Port C.1	Output valve
Port D.0	Input-output USART
Port D.1	Input-output USART

Reset pada *microcontroller* ATmega32 terjadi dengan adanya logika *high* “1” selama dua *cycle* pada kaki RST pada *microcontroller* ATmega32. Setelah kondisi *pin* RST kembali *low*, maka *microcontroller* akan menjalankan program dari alamat 0000H. Dalam hal ini *reset* yang digunakan adalah *manual reset*. Rangkaian *reset* dapat dilihat pada Gambar 3.6.



Gambar 3.7. Rangkaian *Reset*

Pin XTAL1 dan XTAL2 merupakan *pinoscillator* bagi *microcontroller* ATmega32. Pin XTAL1 berfungsi sebagai *input* dan XTAL2 sebagai *outputoscillator*. *Oscillator* ini bisa berasal dari kristal atau dari keramik *resonator*. Rangkaian *oscillator* dapat dilihat pada Gambar 3.7.



Gambar 3.8. Rangkaian *Oscillator*

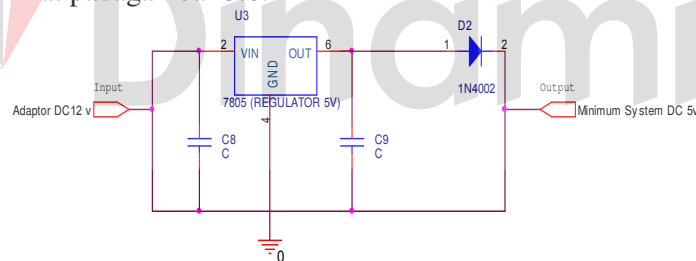
Pada proyek akhir ini dibuat rangkaian *oscillatorinternal* yang terbuat dari kristal. Nilai C1 dan C2 masing-masing 100 pF.

B. Rangkaian L298

Pada proyek ini menggunakan beberapa komponen pendukung untuk membantu kinerja *input* dan *output*. Komponen pendukung ini memiliki rangkaian dan penjelasan sebagai berikut.

Pada bagian regulator untuk proyek ini menggunakan 2 komponen regulator yaitu regulator untuk keluaran 5V dimana dalam proyek ini menggunakan IC 7805 yang bertujuan untuk menstabilkan tegangan dengan keluaran 5V.

Rangkaian ini berfungsi untuk catu daya. Catu daya merupakan pendukung utama bekerjanya suatu sistem. Catu daya yang biasa digunakan untuk menyuplai tegangan sebesar 5 Volt adalah catu daya DC yang memiliki keluaran +5 Volt. Catu daya ini digunakan untuk mensuplay tegangan sebesar 5 Volt. IC 7805 (IC regulator) digunakan untuk menstabilkan tegangan searah. Kapasitor digunakan untuk mengurangi tegangan kejut saat pertama kali saklar catu daya dihidupkan. Sehingga keluaran IC regulator 7805 stabil sebesar 5 Volt DC. Rangkaian regulator terlihat padagambar 3.8.



Gambar 3.9. Rangkaian Regulator 5V

Rangkain *driver* motor L298H ini dibuat untuk mengendalikan kerja motor DC 6V dimana mikrokontroler nantinya akan memberikan inputan pada pin-pin dari L298H ini untuk mengaktifkan dari motor DC.

Rangkaian *driver* motor L298H ini dibuat untuk mendukung kerja dari L298H, karna L298H ini tidak dapat berdiri sendiri melainkan harus ada

rangkaian dan komponen pendukung lainnya seperti catu daya, kapasitor, dan *conector* yang biasanya disebut sebagai *driver* motor.

Perancangan driver motor ini menggunakan *driver* motor *Embedded Module Series* (EMS) 2 A *Dual H-Bridge* merupakan *driver H-Bridge* yang didisain untuk menghasilkan drive 1 arah maupun 2 arah dengan arus kontinyu sampai dengan 2 A pada tegangan 4,5 Volt sampai 36 Volt. Modul ini mampu men-drive beban – beban induktif seperti misalnya *relay*, solenoida, motor DC, motor stepper, dan berbagaimacam beban lainnya.



Gambar 3.10. Bentuk fisik *driver* motor

Modul *H-Bridge* memiliki 2 buah *header* (*Interface Header 1* dan *Interface Header 2*) dan 1 set konektor (*Power & Motor Con*). Pada bagian ini akan dijelaskan deskripsi dan fungsi dari masing-masing header dan konektor tersebut.

Interface Header 1 (J1) berfungsi sebagai input untuk mengendalikan sepasang *driver Hbridge* yang pertama. Berikut deskripsi dari masing-masing pin pada *Interface Header 1*:

No. Pin Nama I/O Fungsi

1. M1IN1 I Pin input untuk menentukan output M1OUT1
2. M1IN2 I Pin input untuk menentukan output M1OUT2
3. NC - Tidak terhubung ke mana-mana

4. M1EN I Pin enable untuk pasangan output M1(M1OUT1 dan M1OUT2)
5. VCC - Terhubung ke catu daya untuk input (5 Volt)
6. PGND - Titik referensi untuk catu daya input

Interface Header 2 (J2) berfungsi sebagai input untuk mengendalikan sepasang *driver Hbridge* yang kedua. Berikut deskripsi dari masing-masing pin pada *Interface Header 2*:

No. Pin Nama I/O Fungsi

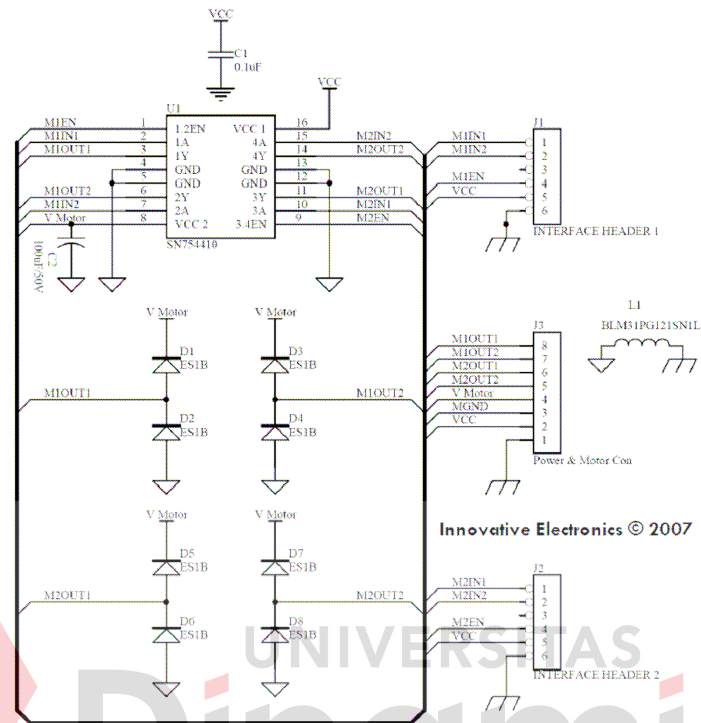
1. M2IN1 I Pin input untuk menentukan output M2OUT1
2. M2IN2 I Pin input untuk menentukan output M2OUT2
3. NC - Tidak terhubung ke mana-mana
4. M2EN I Pin enable untuk pasangan output M2(M2OUT1 dan M2OUT2)
5. VCC - Terhubung ke catu daya untuk input (5 Volt)
6. PGND - Titik referensi untuk catu daya input

Power & Motor Con (J3) berfungsi sebagai konektor untuk catu daya dan beban. Berikut deskripsi dari masing-masing terminal pada *Power & Motor Con*:

Nama Fungsi

1. PGND Titik referensi untuk catu daya input
2. VCC Terhubung ke catu daya untuk input (5 Volt)
3. MGND Titik referensi untuk catu daya output ke beban
4. V MOT Terhubung ke catu daya untuk output ke beban
5. M2OUT2 Output ke beban dari half H-Bridge ke-2 pada pasangan HbridgeM2
6. M2OUT1 Output ke beban dari half H-Bridge ke-1 pada pasangan HbridgeM2
7. M1OUT2 Output ke beban dari half H-Bridge ke-2 pada pasangan HbridgeM1

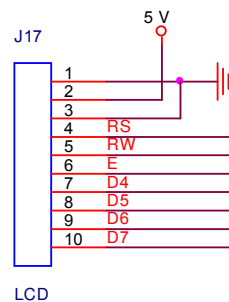
8. M1OUT1 Output ke beban dari half H-Bridge ke-1 pada pasangan HbridgeM1



Gambar 3.11. Gambar skematik *driver* motor

C Rangkaian LCD

Modul *display* merupakan modul yang berfungsi untuk menampilkan menu, intruksi-intruksi program yang akan dijalankan, informasi waktu dan temperatur saat proses dilakukan yang dikirim oleh *microcontroller*. Informasi tersebut ditampilkan pada sebuah LCD 16 x 2.



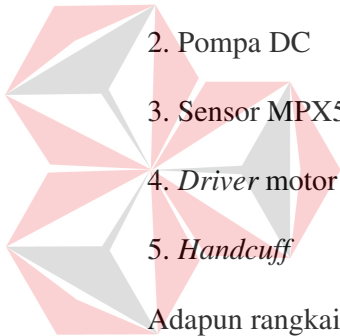
Gambar 3.12. Rangkaian LCD

D Rangkaian alat pengukur tekanan darah

Rangkaian alat pengukur tekanan darah ini dibuat untuk melakukan pengintegrasian antar *input* dan *output* baik dari aktuator maupun dari sensor tiap komponen. Adapun komponen dari alat pengukur tekanan darah ini terdiri dari

1. *Valve*
2. Pompa DC
3. Sensor MPX5100DP
4. *Driver* motor
5. *Handcuff*

Adapun rangkaian dari alat pengukur tekanan darah itu adalah sebagai berikut:



UNIVERSITAS
Dinamika



Gambar 3.13. Rangkaian alat pengukur tekanan darah

Pompa DC dihubungkan pada valve, sensor MPX5100DP dan manset

dengan media 3 jalur selang udara. Jalur pertama dihubungkan dengan Pompa DC untuk memberikan udara, Jalur kedua dihubungkan dengan sensor MPX5100DP yang akan mengukur kadar tekanan udara di dalam rangkaian, sedangkan jalur terakhir dihubungkan dengan valve sebagai media jalan keluar udara. Semua udara yang dipompakan Pompa DC akan terhubung dengan *Handcuff* yang akan digunakan pada pergelangan tangan obyek yang akan di ukur. Pada tabel 3.3, 3.4, 3.5 berikut merupakan daftar pin *input/output* yang digunakan dalam penelitian ini.

Tabel 3.2. Pin Input/Output MPX5100DP.

Pin I/O	Fungsi
Data	<i>Output</i> data pada ADC0 mikrokontroler
5V	Catu daya 5V
<i>Ground</i>	Catu daya <i>Ground</i>

Tabel 3.3. Pin Input/Output driver motor dan pompa DC.

Pin I/O	Fungsi
Pgnd	Catu daya <i>Ground</i> driver motor
VCC	Catu daya 5V driver motor
Mgnd	Catu daya <i>Ground</i> pompa DC
VMot	Catu daya 5V pompa DC
M1Out1	<i>Output</i> motor DC
M1Out2	<i>Output</i> motor DC
M1In1	Input dari mikrokontroler
M1In2	Ground dari mikrokontroler
M1En	<i>Enable</i> dari mikrokontroler

Tabel 3.4. Pin Input/Output dan logika valve.

Pin I/O	Fungsi	VCC	Katup
VCC	Input mikrokontroler	5V	Tertutup
Gnd	Catu daya <i>ground</i>	0V	Terbuka

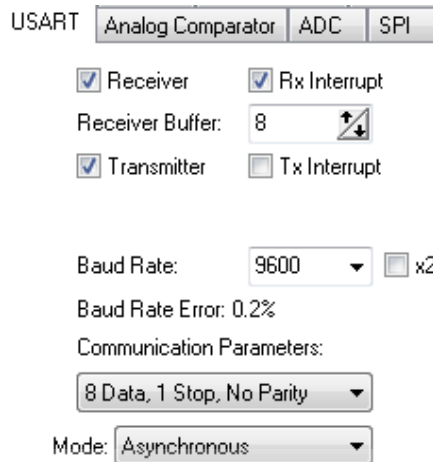
3.2. Perancangan Perangkat Lunak

Pada perancangan perangkat lunak ini dibagi atas beberapa bagian yang memiliki fungsi masing-masing untuk menjalankan alat pengukur tekanan darah ini. Adapun bagian-bagian dari perancangan perangkat lunak ini dibagi sebagai berikut.

A. Konfigurasi Serial Pada Mikrokontroler

Konfigurasi *serial* atau USART (*Universal synchronous asynchronous Receiver Transmitter*) pada CVAVR ini harus sesuai dengan pengaturan yang ada pada aplikasi android, dikarenakan ini adalah pengiriman yang tidak *synchronous* sehingga apabila terjadi perbedaan kecepatan penerimaan data atau pengiriman data maka data yang diterima tidak akan benar (*fail*). Konfigurasi *serial* dari *interfaces* perangkat ini diatur menjadi kecepatan pengiriman data 9600 bps

(BaudRate), dengan jumlah pengiriman data perbitnya yaitu 8 bit, stop bitnya 1 bit, parity dan flow diatur menjadi kosong. Seperti pada gambar berikut.



USART Analog Comparator ADC SPI

☒ Receiver ☒ Rx Interrupt

Receiver Buffer: 8

☒ Transmitter ☐ Tx Interrupt

Baud Rate: 9600 x2

Baud Rate Error: 0.2%

Communication Parameters:

8 Data, 1 Stop, No Parity

Mode: Asynchronous

Gambar 3.14. Konfigurasi USART pada CVAVR.

Penerimaan data *serial* pada mikrokontroler ini akan diinterup, agar pada saat ada suatu perangkat yang mengirimkan data ke mikrokontroler yang melewati komunikasi *serial* maka mikrokontroler akan melakukan penundaan sementara untuk semua proses yang sedang dia lakukan dan melakukan penerimaan data. Setelah data yang dikirimkan telah selesai sepenuhnya barulah kemudian mikrokontroler melanjutkan interaksi yang ada.

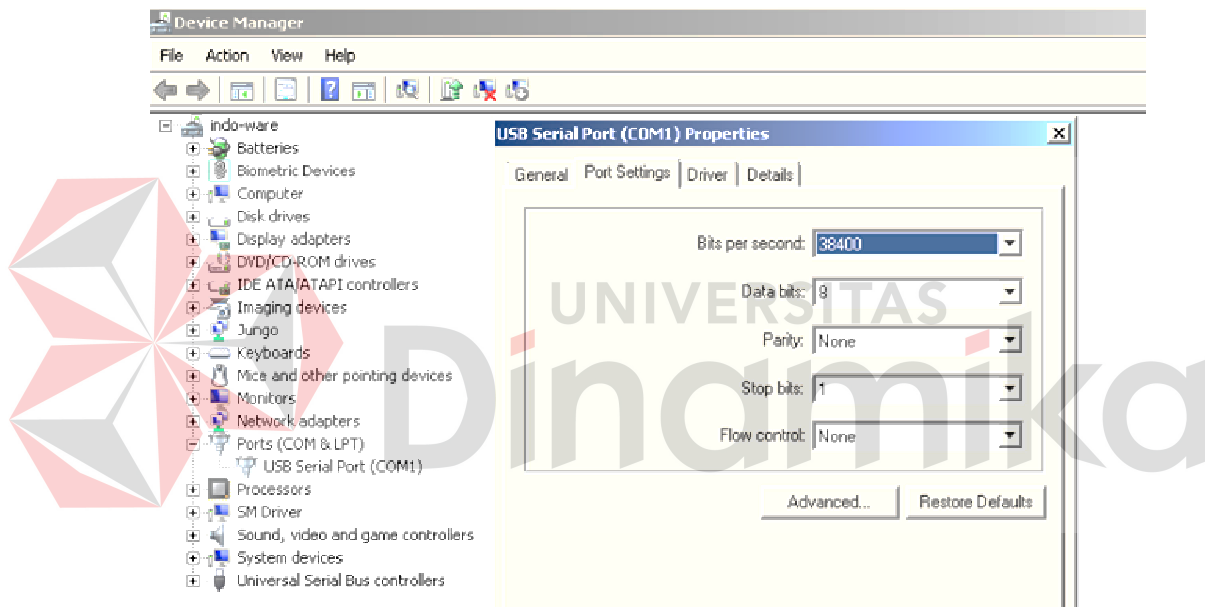
B. Konfigurasi HC-05

HC-05 merupakan modul *Bluetooth* mikrokontroler yang akan digunakan berkomunikasi dengan *mobile android*, sebelum menggunakan HC-05 ada konfigurasi yang harus dilakukan dengan media usb to ttl dengan PC(*Personal Computer*)

Untuk memulai koneksi hubungkan *port* RX milik HC-05 dengan *port* TX pada USB to TTL dan juga *port* TX milik HC-05 dengan *port* RX pada USB to

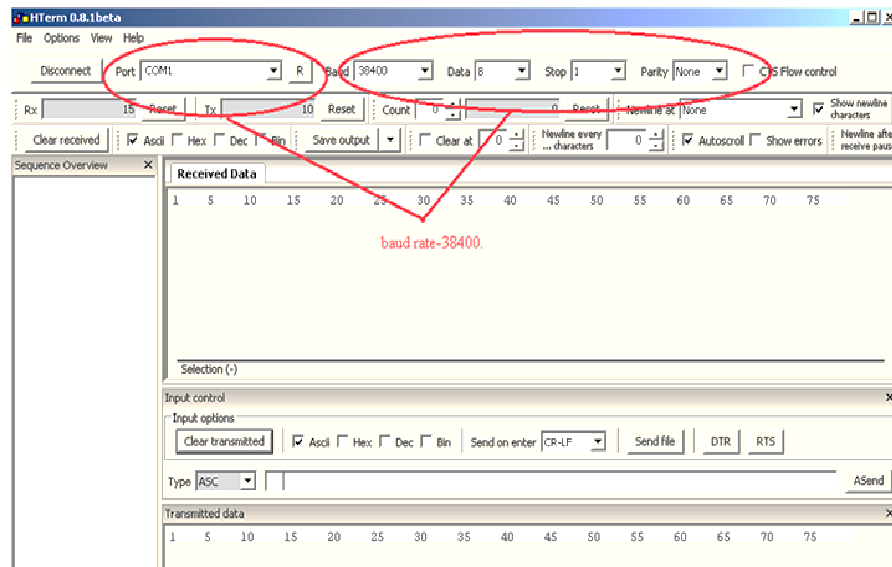
TTL, serta hubungkan juga *port5V* milik USB to TTL dengan 5V milik HC-05 dan juga GROUND milik HC-05 pada *port* GROUND milik USB to TTL. *Trigger pin* key pada HC-05 dengan 5V untuk masuk ke dalam mode AT Command agar dapat mengkonfigurasi modul HC-05.

Konfigurasi di device manager FTDI232 Baud 38400 bps, data bits 8, parity None, stop bits 1, Flow control, *port* COM1



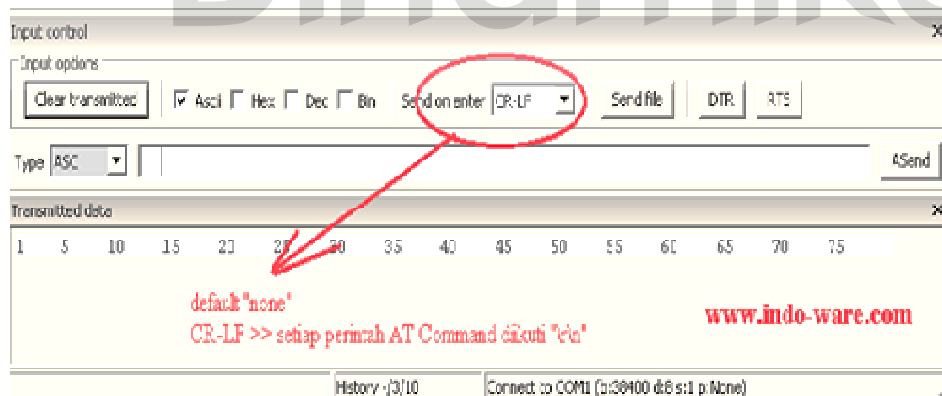
Gambar 3.15. Konfigurasi *port* pada *device manager*.

Setelah itu jalankan aplikasi Hterm dan konfigurasi kan *Port* yang digunakan sama dengan *port* yang digunakan USB to TTL, dalam percobaan ini *port1*. *Settingboudrate* menjadi 38400 data bit 8 stop bit 1 dan parity data none.



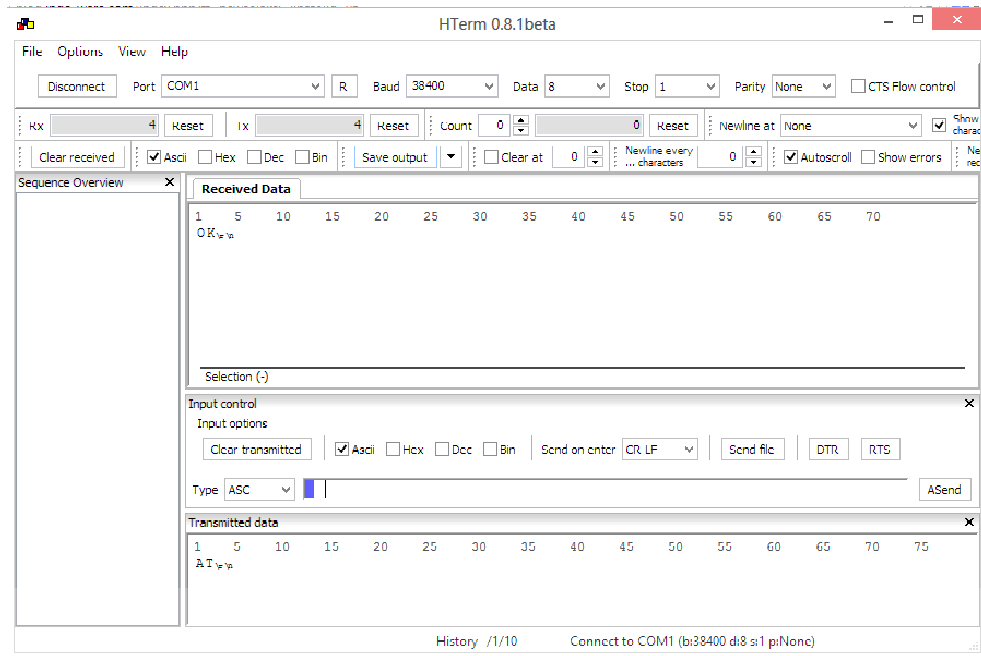
Gambar 3.16. Konfigurasi awal Hterm.

Setting juga untuk setiap komunikasi yang digunakan untuk diakhiri dengan CR-LF, CR-LF ini merupakan kode ASCII ke-13 yang digunakan untuk menandakan akhir dari data yang dikirim



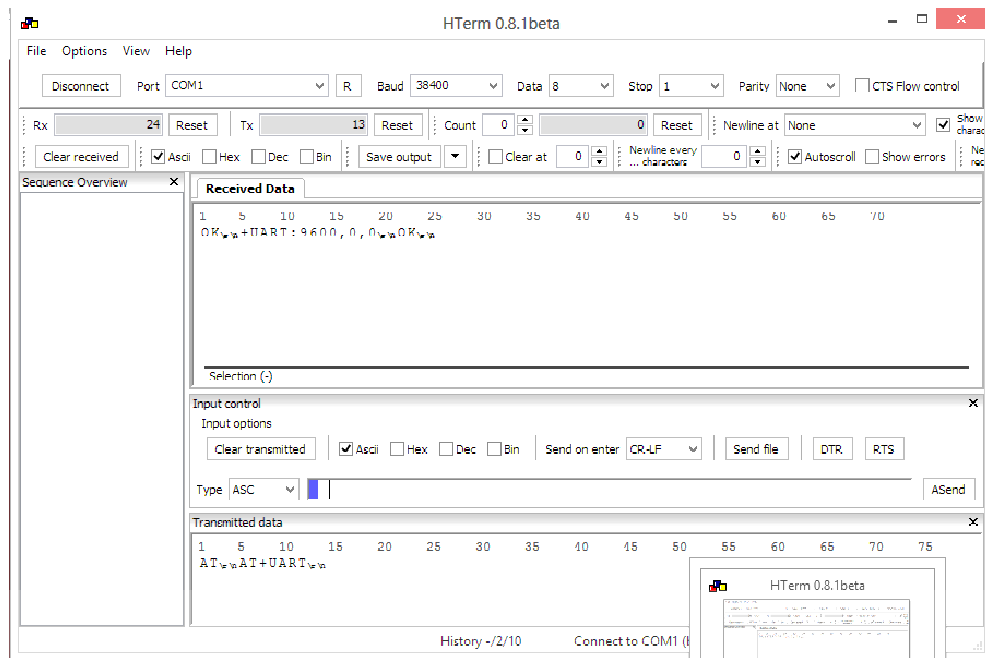
Gambar 3.17. Konfigurasi CR-LF.

Setelah semua *setting* dilakukan lakukan percobaan untuk memanggil modul *Bluetooth* dengan mengetikkan “AT”. Modul akan membalas “OK” bila *setting* yang dilakukan sudah benar.



Gambar 3.18. Memanggil modul *bluetooth*.

Setelah percobaan pemanggilan sudah dilakukan maka saatnya melakukan konfigurasi *boudrate* yang akan digunakan oleh *Bluetooth* dengan memasukkan *boudrate* sebesar 9600,0,0. 9600 berarti *boudrate* yang digunakan adalah 9600, 0 pertama berarti stop bit yang digunakan adalah 1, dan 0 yang ke dua berarti no parity data.

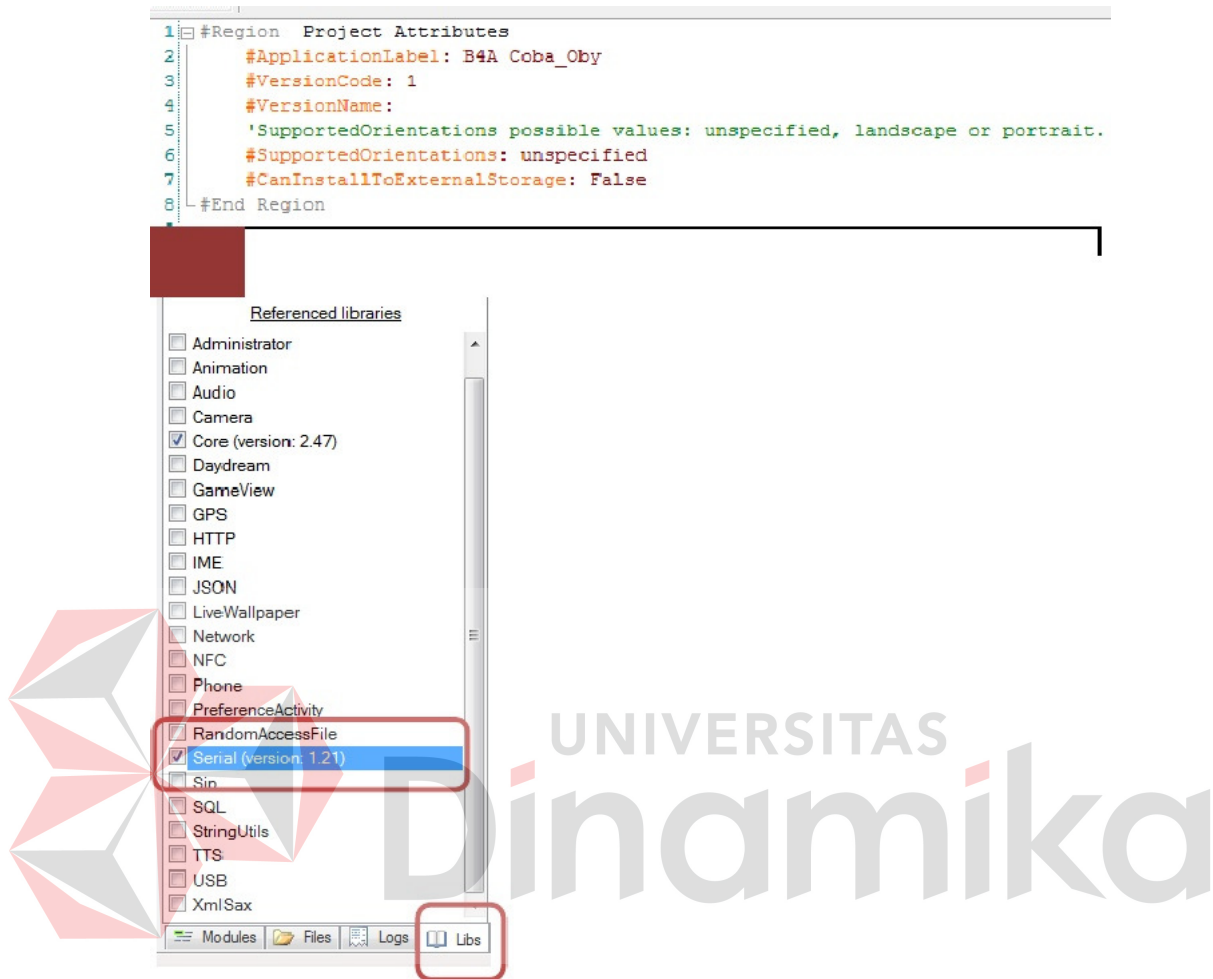


Gambar 3.19. *Setting boudrate.*

Setelah semua *setting* dilakukan maka *Bluetooth* HC-05 siap digunakan. Sambungkan *PIN* tx pada HC-05 pada *PORT* D.0, *PIN* rx pada HC-05 pada *PORT* D.1, *PIN* VCC pada 5V milik mikrokontroler, dan *PIN* ground pada ground milik mikrokontroler.

C. Konfigurasi USART Basic4android

USART dalam *basic4android* merupakan suatu fitur yang diberikan pada pengguna untuk dapat berkomunikasi melalui serial baik pada sesama *mobile* android maupun dengan perangkat lain. Dalam fitur serial dalam *basic4android* *boudrate* default yang digunakan adalah 9600 dan untuk melengkapi kecepatan data diterima dan di tampilkan kita harus menggunakan fitur timer yang telah disediakan.



Gambar 3.20. Pemilihan fitur basic4android

Pertama kali proses yang dilakukan adalah proses permintaan hubungan perangkat, berikut contoh programnya

```

Dim PairedDevices As Map
PairedDevices = serial1.GetPairedDevices
Dim l As List
l.Initialize
For i = 0 To PairedDevices.Size - 1
    l.Add(PairedDevices.GetKeyAt(i))
Dim res As Int
res = InputList(l, "Hubungkan", -1)
If res <> DialogResponse.CANCEL Then
    serial1.Connect(PairedDevices.Get(l.Get(res)))
End If

```

Pada proses permintaan hubungan perangkat ini pertama mengakses daftar perangkat *bluetooth* yang pernah terhubung dan menampilkan nya pada layar. Setelah menampilkan maka user akan diminta memasukkan perangkat yang ingin dihubungkan, setelah perangkat dipilih maka *android* akan mengirimkan permintaan koneksi pada modul bluetooth HC-05. *Android* akan menunggu jawaban dari proses koneksi tersebut dengan program berikut ini

```

If Success Then
    ToastMessageShow("Connected successfully", False)
    Textreader1.Initialize(serial1.InputStream)
    Textwriter1.Initialize(serial1.OutputStream)
    timer1.Enabled = True
    connected = True
Else
    connected = False
    timer1.Enabled = False
    MsgBox>LastException.Message, "Error connecting."
End If

```

Jika sukses dalam proses koneksi maka akan menampilkan notifikasi “*Connected successfully*”, dan melakukan pengaktifan *timer 1*, *textreader 1* sebagai input, *textwriter 1* sebagai output. Jika proses koneksi tidak sukses maka akan menampilkan notifikasi “*Error connecting*” dan akan menonaktifkan *timer 1*. *Timer 1* digunakan untuk proses penerimaan data supaya lebih *real-time*, sedangkan *textreader* digunakan sebagai variabel dalam penerimaan data, dan *textwriter* digunakan sebagai variabel pengiriman data.

D. Perancangan Program Tekanan Darah Pada Mikrokontroler

Mikrokontroler sebagai otak dari penggerak alat tekanan darah ini memiliki akses penuh ke semua sensor dan aktuator dalam alat ini. antara lain akses kepada sensor tekanan darah MPX5100DP, akses kepada aktuator *pompa DC* dan *valve*, akses pada display data seperti LCD dan *mobile android*, dan juga akses kepada

modul *Bluetooth* sebagai media komunikasi antara mikrokontroler dan *mobile android*.

MPX5100DP merupakan sensor tekanan dengan masukan tekanan dan keluaran tegangan. Tegangan yang nanti di keluarkan oleh sensor ini akan digunakan untuk membaca nilai tekanan darah dengan adanya konversi dengan rumus:

```
vin_dis=((float)dis*5/1023)+(0.024);  
kilo_pascal_dis=(vin_dis/0.045)-(4.45);  
milimeter_hg_dis=kilo_pascal_dis*7.5006;
```

dengan konversi tersebut maka di dapatkan nilai tekanan yang ada pada manset(cuff) dari alat ini. Setelah dilakukan konversi tersebut maka alat dapat melakukan proses kalibrasi tekanan darah. Adapun alur dari metode osilometri tersebut diawali dengan mikrokontroler akan melakukan pemompaan dengan menyalakan pompa DC hingga tekanan mencapai 160mmHg. Setelah mencapai nilai tersebut, akan dimatikan pompa dan dilakukan sehingga tekanan pada *cuff* akan berkurang secara berangsur. Kemudian selang 10s dari pompa mati, akan dicatat sebagai tekanan sistolik, dan setelah 15s dari pompa mati akan dicatat sebagai tekanan diastolik(Soot W. N well, Ipswich, dan Mass 1993). Kemudian setelah sistolik dan diastolik dikirimkan kepada *mobile android* dan *solenoid valve* bekerja off untuk mengeluarkan semua udara yang ada pada *cuff*.

Berdasarkan Alur dari proses kalibrasi maka penulisan program adalah sebagai berikut :

```
printf("Proses Kalibrasi");  
step_1:  
pompa=1;  
valve=1;  
awal=read_adc(0);
```

```

vin=((float)awal*5/1023);
pascal=(vin/0.045)-(4.45);
mili=(pascal*7.5006);
ftoa(mili,3,tampil);
lcd_clear();
lcd_gotoxy(0,0);
lcd_puts(tampil);
lcd_gotoxy(0,1);
lcd_putsf("Harap Bersabar");

if(mili>159)
{
    pompa = 0;
    lampu = 1;
    goto step_2;
}
goto step_1;

step_2:
lcd_gotoxy(0,1);
lcd_putsf("Harap tunggu...");
delay_ms(10000);
sis=read_adc(0);
vinsis=((float)sis*5/1023);
pascalsis=(vinsis/0.045)-(4.45);
milisis=((pascalsis*7.5006)-15);
delay_ms(2000);
dis=read_adc(0);
vindis=((float)dis*5/1023);
pascaldis=(vindis/0.045)-(4.45);
milidis=((pascaldis*7.5006)-60);
ftoa(milidis,3,tampil);
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("Sistol: ");
lcd_puts(tampil);
lcd_gotoxy(0,1);
ftoa(milidis,3,tampil);
lcd_putsf("diastol: ");
lcd_puts(tampil);
valve=0;

```

E. Perancangan Program Pengelompokan dan Pengiriman Data

Data tekanan darah yang di kalibrasi merupakan data bilangan decimal yang harus dikirim kepada *mobile android* dengan tingkat akurasi pengiriman data yang tinggi. Untuk dapat memenuhi akurasi data yang tinggi tersebut diperlukan pengelompokan data terlebih dahulu, karna data yang akan dikirimkan merupakan data yang tidak pasti (berubah-ubah). Sedangkan untuk melakukan pengiriman

daa dari mikrokontroler dengan serial ada dua acara yaitu dengan “sprintf” dan “putchar”.

Untuk pengiriman data dengan menggunakan “sprintf”, data yang dikirimkan merupakan data pasti dan berupa beberapa karakter maupun 1 karakter. Sedangkan untuk pengiriman menggunakan “putchar”, data yang dikirimkan data yang tidak pasti dan juga harus berupa 1 karakter. Penulis menggunakan sprintf untuk mengirimkan data sebagai berikut :

```
tus=milisis;
if(tus<10)
{
    ceksatuan = (int)tus % 10;
    mob(ceksatuan);
}
else if(tus>99)
{
    ceksatuan = (int)tus % 10;
    cekpuluhan = (((int)tus%100)-((int)tus%10))/10;
    cekratusan = (((int)tus)-((int)tus%100))/100;
    mob(cekratusan);
    mob(cekpuluhan);
    mob(ceksatuan);
}
else
{
    ceksatuan = (int)tus % 10;
    cekpuluhan = (((int)tus%100)-((int)tus%10))/10;
    mob(cekpuluhan);
    mob(ceksatuan);
}
```

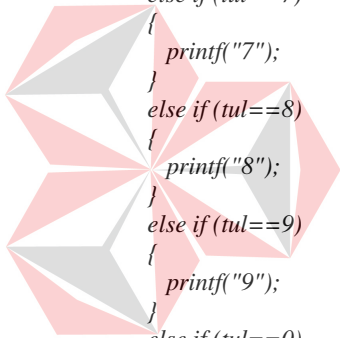
Setelah data telah dikelompokkan, barulah data dapat dikirimkan dengan menggunakan printf sebagai berikut :

```
void mob(int tul)
{
    if (tul==1)
    {
        putchar("1");
    }
    else if (tul==2)
    {
        printf("2");
    }
}
```

```

else if (tul==2)
{
    printf("2");
}
else if (tul==3)
{
    printf("3");
}
else if (tul==4)
{
    printf("4");
}
else if (tul==5)
{
    printf("5");
}
else if (tul==6)
{
    printf("6");
}
else if (tul==7)
{
    printf("7");
}
else if (tul==8)
{
    printf("8");
}
else if (tul==9)
{
    printf("9");
}
else if (tul==0)
{
    printf("0");
}
}

```



UNIVERSITAS
Dinamika

Penulis menggunakan *function* untuk mengirimkan data, karna terdapat kondisi *if* (jika) yang merupakan percabangan dalam mengirimkan tiap karakter dari data hasil kalibrasi tekanan darah tersebut.

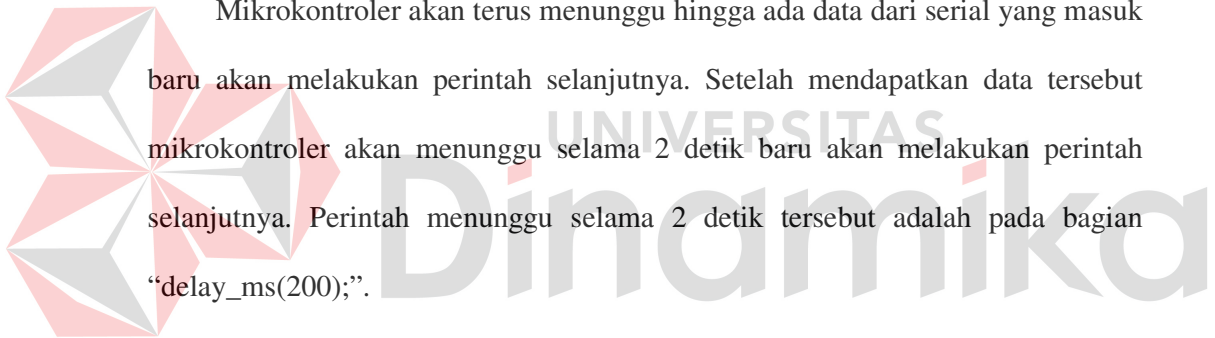
F. Perancangan Program Penerimaan data

Mikrokontroler adalah otak dan aktuator dari alat ini, namun untuk mengaktifkan mikrokontroler dibutuhkan perintah dari *mobile android*. Untuk menunggu dan menerima data dari *mobile android* maka perlu dilakukan program

pengkondisian. Program pengkondisian untuk menunggu data atau perintah dan menerima nya adalah “getchar”. Getchar adalah suatu perintah untuk menahan proses dari tiap alur mikrokontroler hingga mikrokontroler menerima data serial dari perangkat lain.

Adapun program pengkondisian tersebut adalah sebagai berikut :

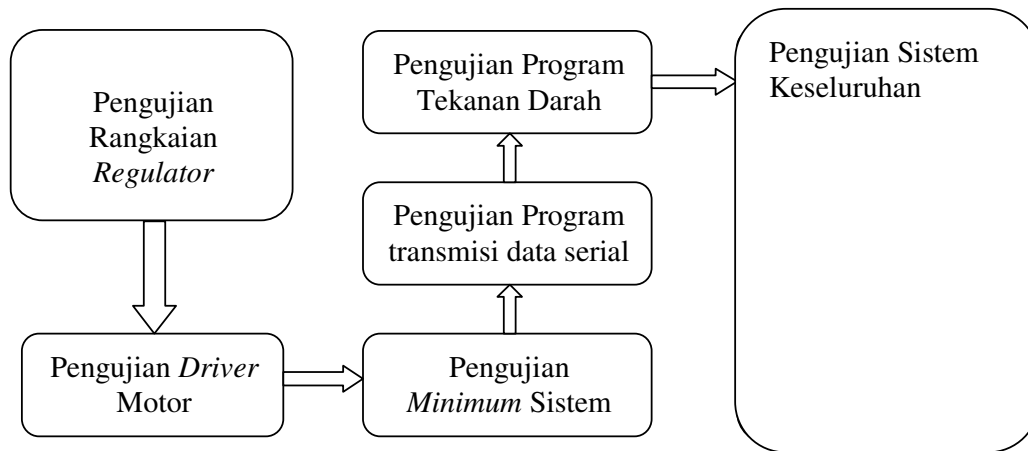
```
lcd_puts("tekan karakter Y untuk memulai");
tampung = getchar();
delay_ms(200);
```



Mikrokontroler akan terus menunggu hingga ada data dari serial yang masuk baru akan melakukan perintah selanjutnya. Setelah mendapatkan data tersebut mikrokontroler akan menunggu selama 2 detik baru akan melakukan perintah selanjutnya. Perintah menunggu selama 2 detik tersebut adalah pada bagian “delay_ms(200);”.

3.3 Langkah Pengujian

untuk menyatakan sebuah sistem ini telah berhasil berjalan atau berkerja dengan baik maka akan dilakukan pengujian terhadap setiap perangkat dan modul yang akan digunakan, Dalam langkah pengujian ini, pengujian dibagi atas beberapa modul atau perangkat secara terpisah terlebih dahulu, yang kemudian akhirnya akan digabungkan menjadi satu pengujian secara keseluruhan. Langkah yang akan dilakukan untuk pengujian ini sesuai dengan gambar blok diagram 3.15 berikut.



Gambar 3.21. Langkah-Langkah Pengujian

Pengujian yang pertama kali dilakukan adalah pengujian terhadap modul *regulator* yang digunakan untuk memberikan sumber tegangan pada semua perangkat yang akan digunakan nantinya, pengujian *drivemotor* sebagai pengujian penguat tegangan pada *pompa DC*, pengujian *minimum* sistem sebagai otak dari sistem yang akan digunakan, pengujian alat tekanan darah sebagai titik fokus sensor, pengujian pengiriman serial yang akan dilakukan antara mikrokontroler dengan perangkat *android*. Yang terakhir adalah pengujian *mobile robot* ini secara keseluruhan dimana semua sistem dari *mobile robot* ini akan digunakan seutuhnya.

3.3.1 Pengujian Rangkaian Regulator

Pengujian rangkaian *regulator* ini adalah untuk memastikan hasil luaran rangkaian ini berada pada tegangan operasi dari perangkat *minimum* sistem, *access point*, *magnetic encoder* dan perangkat lainnya. Prosedur pengujian dari rangkaian *regulator* yang akan digunakan pada penelitian ini adalah sebagai berikut.

1. Nyalakan catu daya pada *power supply*.
2. Atur tegangan *input* dari *power supply* antara 9 Volt hingga 12 Volt.
3. Hubungkan catu daya dengan rangkaian *input regulator*.
4. Gunakan *AvoMeter* pada kaki *output* dari rangkaian untuk melihat besar tegangan keluarannya.

Tabel 3.5. Pengujian Rangkaian Regulator

Tujuan	Alat dan Bahan	Input	Output yang diharapkan	Indikator Keberhasilan
Untuk mengetahui hasil keluaran dari rangkaian regulator	1. <i>Power Supply</i> 2. Rangkaian <i>Regulator</i> 3. <i>AvoMeter</i>	Tegangan <i>Input</i> dari <i>battery</i> sebesar 12V	Hasil Keluaran yang ditunjukkan oleh <i>AvoMeter</i> adalah sebesar 5 Volt	Hasil keluaran yang ditunjukkan dari <i>AvoMeter</i> adalah rentang <i>range</i> antara 4.5V hingga 5V

3.3.2 Pengujian Rangkaian Driver

Tujuan dari pengujian ini adalah untuk mengetahui apakah *driver* motor yang akan digunakan ini dapat menggerakkan motor DC dengan kecepatan yang berbeda dan dapat mengatur arah dari putaran motor DC yang ada. Prosedur pengujian dari *driver* motor yang akan digunakan pada penelitian ini adalah sebagai berikut.

1. Nyalakan catu daya pada *power supply* menjadi 9V
2. Hubungkan kabel power dari motor DC 6V ke *connector* motor dari L298H.
3. Hubungkan semua kaki *input* dari rangkaian *driver* motor ke kabel data.
4. Hubungkan M1IN1 pada tegangan 5V dan M1IN2 pada tegangan 0V/GND
5. Untuk *output*, berikan *input* 5V dari *power supply* ke kabel data M1EN pada rangkaian motor *driver*.

6. Lakukan Pengukuran pada *connector output* tegangan ke motor apakah sudah 9V

Tabel 3.6. Pengujian Rangkaian *Driver Motor*

Tujuan	Alat dan Bahan	Input	Output yang diharapkan	Indikator Keberhasilan
Untuk mengetahui kerja dari <i>drivermotor</i>	1. <i>Power Supply</i> 2. Rangkaian <i>DriverMot</i> or	1. Kaki M1IN1 diberikan tegangan 5V dan M1IN2 diberikan 0V 2. <i>Input</i> 5V dari <i>power supply</i> ke M1EN	1. Tegangan keluaran dari <i>drivermot</i> or diatas 6V	Motor 6V dapat berputar.

3.3.3 Pengujian Rangkaian *Minimum* Sistem

Tujuan dari pengujian rangkaian *minimum* sistem ini adalah untuk mengetahui apakah modul ini dapat digunakan sebagai otak dari penggerak sistem yang ada. Prosedur pengujian dari *minimum* sistem yang akan digunakan dalam penelitian ini adalah sebagai berikut.

1. Aktifkan *power supply* dan hubungkan dengan *minimum* sistem.
2. Pasangkan ATmega32 pada rangkaian *minimum* sistem.
3. Sambungkan *minimum* sistem dengan *downloader* dengan kabel *downloader* USB ASP.
4. Selanjutnya nyalakan komputer dan jalankan program *Code Vision AVR*.
5. Setelah itu *Compile* dan *Build* program yang ingin di download ke dalam mikrokontroler.
6. Untuk *download* program yang telah dibuat kedalam *minimum* sistem maka yang harus dilakukan adalah buka aplikasi *cmd*, lalu buka direktori tempat file disimpan.

7. Langkah terakhir lakukan proses download dengan cara ketik “avrdude -c usbasp -p m32 -u -U flash:w:io.hex”, io.hex adalah nama file hexa yang ingin di download ke dalam mikrokontroler.

Tabel 3.7. Pengujian Rangkaian *Minimum* Sistem

Tujuan	Alat dan Bahan	Input	Output yang diharapkan	Indikator Keberhasilan
Untuk mengetahui keadaan <i>minimum</i> sistem	1. <i>Downloader</i> ASP 2. Rangkaian <i>Minimum</i> sistem 3. <i>Power Supply</i>	1. Jalankan CMD 2. Masukkan perintah download file hex	1. <i>Read Success</i> 2. <i>Load Success</i>	Tampilan <i>ReadSuccess</i> Tampilan <i>Load Program Success</i>

3.3.4 Pengujian Pengiriman dan Penerimaan Data Serial

Pengujian modul HC-05 bertujuan untuk mengubungkan komunikasi yang terjadi antara mikrokontrolerdengan perangkat *android* yang pengirimannya dilakukan secara serial. Prosedur pengujian dari modul HC-05 yang akan digunakan pada *Penelitian* ini adalah sebagai berikut.

1. Nyalakan catu daya pada power supply dengan *input* 5 Volt DC.
2. Hubungkan kaki Rx dari *minimum* sistem dengan kaki Tx HC-05 dan kaki Tx dari *minimumsystem* dengan kaki Rx HC-05. .
3. Nyalakan *android* dan mikrokontroler.
4. Kirimkan data dari *android* kepada mikrokontroler berupa karakter yaitu sebagai berikut.

```

If connected Then
    Textwriter1.WriteLine(textsend.text)
    Textwriter1.Flush
    textsend.text = ""
End If

```

Adapun program pada mikrokontroler adalah sebagai berikut.

```
tampung = getch();
sprintf(hasil,"%c",tampung);
lcd_puts(hasil);
delay_ms(200);
```

5. Kirimkan data dari mikrokontroler ke *android* dengan berupa 100, 150, dan 200 karakter contoh pengiriman 100 karakter adalah sebagai berikut.

```
for (i=0;i<=100;i++)
{
    printf("124/87 ");
}
```

Potongan untuk program penerima yang ada pada *android* adalah sebagai berikut.

```
If connected Then
    If Textreader1.Ready Then
        textsend.text = textlog.text & Textreader1.ReadLine & CRLF
        label1.Text = Textreader1.ReadLine & CRLF
        textlog.SelectionStart = textlog.text.Length
    End If
End If
```

6. Lakukan percobaan pada jarak 2 meter, 4 meter, 6 meter, 8 meter, 10 meter, dan 12 meter dan ukur berapa waktu pengiriman data.

Tabel 3.8. Pengujian pengiriman dan penerimaan data *serial*

Tujuan	Alat dan Bahan	Input	Output yang diharapkan	Indikator Keberhasilan
Untuk mengetahui hasil pengiriman serial	1. <i>Power Supply</i> 2. HC-05 3. <i>android</i> 4. mikrokontroler ATmega32	Data pengiriman <i>serial</i>	Apa yang dikirimkan sama dengan apa yang ditampilkan secara <i>realtime</i>	Tampilan dari <i>android</i> dan <i>minimum</i> sistem mampu menampilkan data yang dikirim

3.3.5 Pengujian Program Tekanan Darah

Pengujian program tekanan darah ini bertujuan untuk dapat menghasilkan kalibrasi yang benar akan nilai tekanan darah pada tubuh. Prosedur pengujian dari program tekanan darah yang akan digunakan pada *Penelitian* ini adalah sebagai berikut.

1. Hubungkan kaki valve 5V pada catu daya 5V dan kaki ground pada ground.
2. Hubungkan kaki motor *driver* pada mikrokontroler dan *pompa DC*.
3. Hubungkan kaki LCD pada mikrokontroler.
4. Buat rangkaian pada *pompa DC* yang dihubungkan pada dua komponen yaitu valve dan manset/*handcuff* dimana tekanan udara yang mengalir akan terukur oleh sensor MPX5100DP.
5. Hubungkan kaki pada sensor MPX5100DP pada kaki mikrokontroler dan catu daya.
6. Hidupkan mikrokontroler dan jalankan program tekanan darah.

Tabel 3.9. Pengujian Program Tekanan Darah

Tujuan	Alat dan Bahan	Input	Output yang diharapkan	Indikator Keberhasilan
Untuk mengetahui hasil kalibrasi dari alat pengukur tekanan darah	1. mikrokontroler 2. <i>Driver</i> motor 3. <i>Pompa DC</i> 4. MPX5100DP 5. LCD 16 x 2 6. Valve	Sensor MPX5100DP	Nilai Tekanan darah	Nilai Tekanan darah yang dapat ditampilkan pada LCD.

3.3.6 Pengujian Sistem Secara Keseluruhan

Pengujian sistem secara keseluruhan ini bertujuan untuk menggerakkan alat pengukur tekanan darah dengan koneksi *bluetooth* sesuai dengan apa yang

diharapkan. Prosedur pengujian dari sistem secara keseluruhan yang akan dilakukan pada penelitian ini adalah sebagai berikut.

1. Nyalakan power mikrokontroler.
2. Koneksikan antara *android* dengan mikrokontroler menggunakan *bluetooth*.
3. Kirimkan karakter “y” untuk memulai kalibrasi tekanan darah.
4. Tunggu hingga proses kalibrasi selesai.
5. Data hasil kalibrasi akan ditampilkan pada *android*.
6. Catat setiap hasil kalibrasi alat.

Tabel 3.10. Pengujian Sistem Secara Keseluruhan

Tujuan	Alat dan Bahan	<i>Input</i>	<i>Output</i> yang diharapkan	Indikator Keberhasilan
Mengkalibrasi tekanan darah dan mengirimkan pada <i>android</i> .	<ol style="list-style-type: none"> 1. Alat Pengukur tekanan darah 2. <i>android</i> 3. Tensimeter lain 	Perintah “y” dari <i>android</i>	Nilai tekanan darah dapat ditampilkan pada <i>android</i> dengan nilai <i>error</i> dibawah 10%	Nilai <i>error</i> hasil kalibrasi dengan tensimeter lain dibawah 10%

BAB IV

HASIL DAN PEMBAHASAN

4.1. Hasil Pengujian Perangkat Keras

Sesuai dengan langkah pengujian perangkat keras yang ada sebelumnya maka didapatkan hasil dari pengujian pada setiap modul sebagai berikut.

4.1.1. Hasil Pengujian Rangkaian Regulator

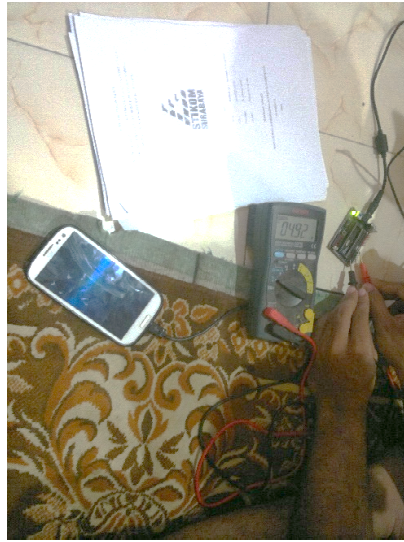
A. Hasil Pengujian

Sesuai dengan langkah pengujian yang dilakukan, didapatkan hasil tegangan dari rangkaian regulator ini seperti tabel 4.1 berikut ini.

Tabel 4.1. Hasil Pengujian Rangkaian Regulator

Pengujian Ke	Luaran Tegangan (Volt)
1	4,92 Volt
2	5,03 Volt
3	4,94 Volt
4	4,92 Volt
5	4,92 Volt
Rata-rata	4,94 Volt

Dapat dilihat pula hasil dari pengujian ini pada gambar 4.1 yang ada berikut ini.



Gambar 4.1. Pengujian Rangkaian Regulator

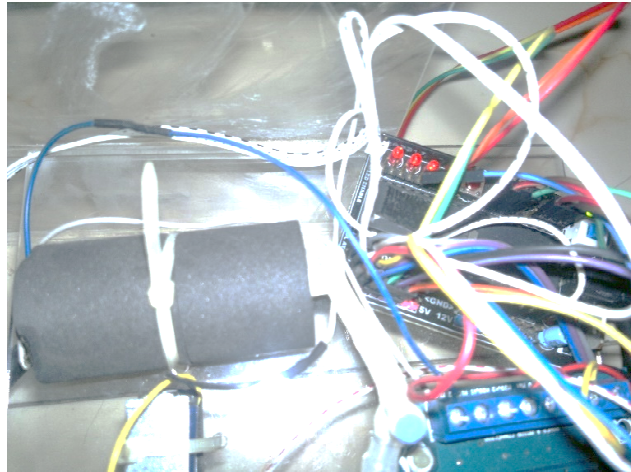
B. Pembahasan

Sesuai dengan hasil pengujian yang sudah dilakukan sebanyak lima kali dan telah sesuai dengan langkah pengujian yang ada, didapatkan bahwa hasil keluaran tegangan dari rangkaian regulator ini adalah sebesar 4.94 volt. Hasil tegangan keluaran ini telah sesuai dengan range kerja dari rangkaian mikrokontroller dan modul lainnya maka rangkaian ini dapat dinyatakan telah berhasil dan dapat digunakan lebih lanjut.

4.1.2. Hasil Pengujian Rangkaian *Driver* Motor

A. Hasil Pengujian

Dari langkah pengujian yang telah dilakukan maka didapatkan hasil dari pengujian untuk mengendalikan putaran motor DC 6Volt ini adalah sebagai berikut.



Gambar 4.2. Motor DC Berputar

Dengan mengikuti langkah pengujian untuk memastikan *output* dari motor *driver* adalah di atas 6V ini didapatkan hasil dari pengujiannya seperti gambar 4.4 berikut.



Gambar 4.3. Tegangan terukur motor

B. Pembahasan

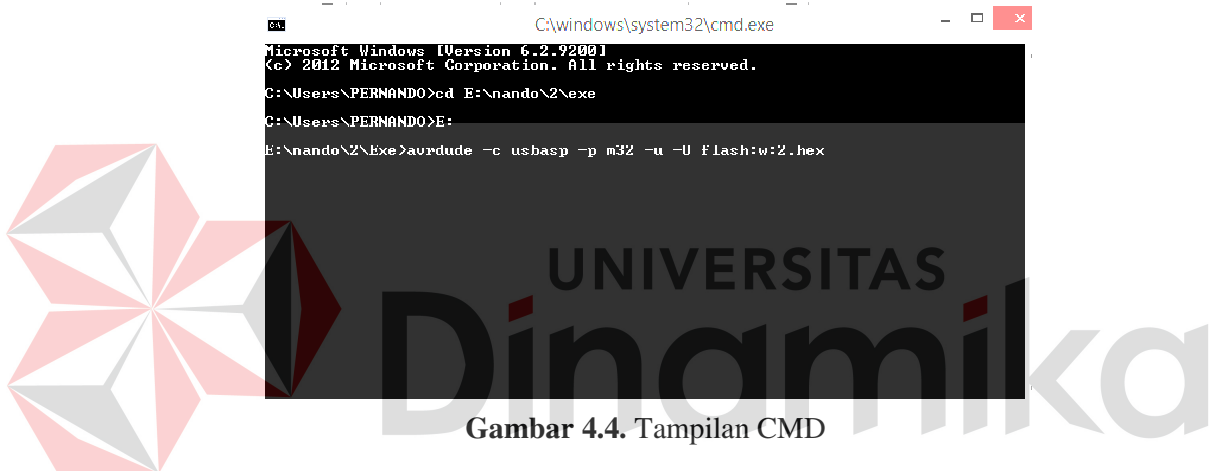
Dari hasil pengujian ingin didapatkan bahwa motor DC 6V telah dapat berputar dengan *input* tegangan 5V dari mikrokontroler. Pada pengujian ini

tegangan yang terukur menggunakan Avo meter adalah 7.75V dan memenuhi target yang diharapkan yang diharapkan yaitu di atas 6V.

4.1.3. Hasil Pengujian Rangkaian *Minimum* Sistem

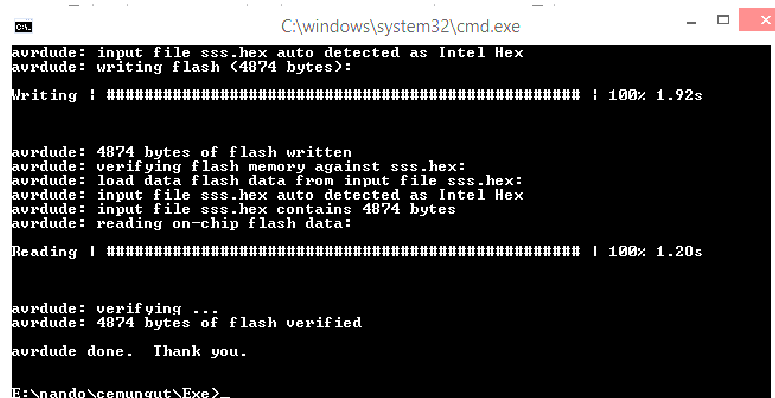
A. Hasil Pengujian

Setelah mengikuti langkah pengujian rangkaian *minimum* sistem maka didapatkan hasil CMD seperti gambar 4.4 berikut ini.



Gambar 4.4. Tampilan CMD

Untuk hasil dari *download* program ke rangkaian *minimum* sistem dapat dilihat seperti gambar 4.5 berikut ini.



Gambar 4.5. Tampilan *Download* Program.

B. Pembahasan

Dari percobaan pada gambar 4.5 CMD, dan gambar 4.6 *download* program berhasil dilakukan maka *minimum* sistem dapat bekerja dengan baik. Tampilan dari gambar 4.5 pada Code Vision AVR yang akan digunakan untuk menuliskan program dan melakukan percobaan terhadap *minimum* sistem. Pada Gambar 4.6 menunjukkan bahwa *minimum* sistem telah berhasil melakukan *download* program ke mikrokontroler sehingga program telah berhasil dijalankan.

4.2. Hasil Pengujian Perangkat Lunak

Sesuai dengan langkah pengujian perangkat lunak yang ada sebelumnya maka didapatkan hasil dari pengujian pada setiap percobaan sesuai dengan langkah pengujian adalah sebagai berikut.

4.2.1. Hasil Pengujian Program Pengiriman Serial

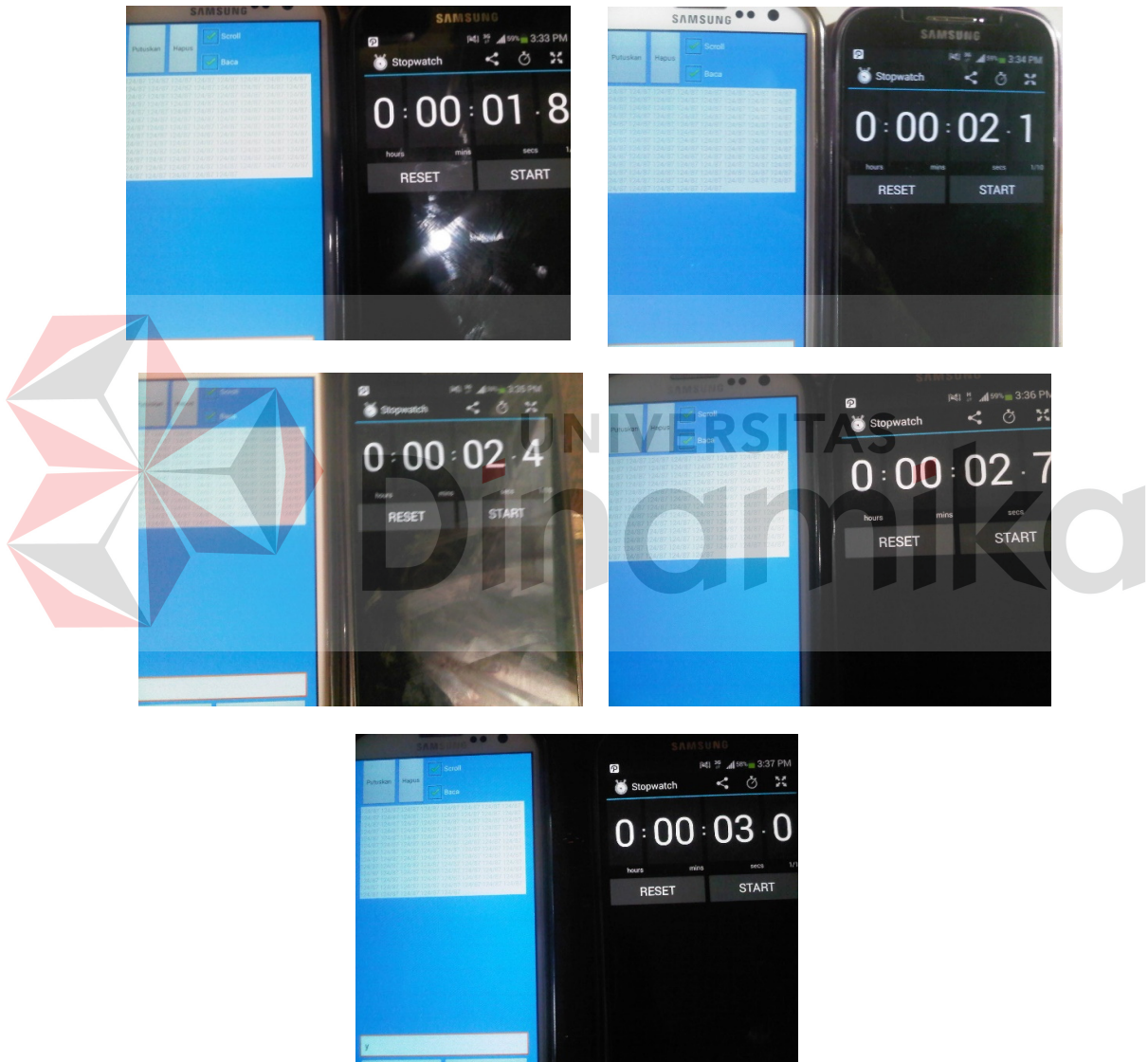
A. Hasil Pengujian

Hasil pengujian dari pengiriman serial ini dapat dilihat pada gambar 4.6 – 4.10 berikut.



Gambar 4.6. Pengiriman dari *android* ke mikrokontroler

Setelah *android* mengirimkan data berupa karakter “y” ke mikrokontroler pengujian selanjutnya yaitu pengiriman data dari mikrokontroler ke *android* untuk ditampilkan dengan prosedur pengujian yang ada. Adapun hasil dari pengiriman data tersebut seperti gambar 4.7, 4.8, dan 4.9.



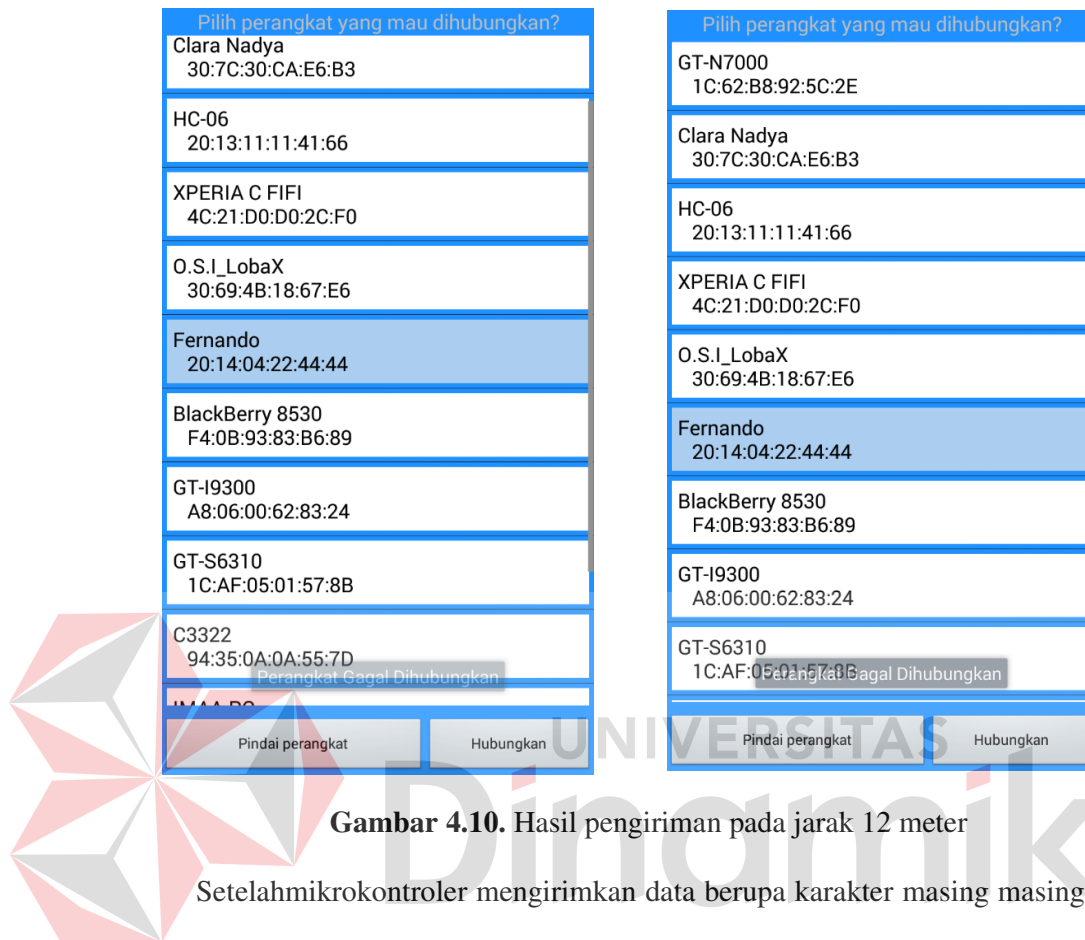
Gambar 4.7. Hasil Pengiriman 100 karakter pada jarak 2 meter, 4meter, 6meter,8meter, dan 10 meter



Gambar 4.8. Hasil Pengiriman 150 karakter pada jarak 2 meter, 4meter, 6meter,8meter, dan 10 meter



Gambar 4.9. Hasil Pengiriman 200 karakter pada jarak 2 meter, 4meter, 6meter,8meter, dan 10 meter



Gambar 4.10. Hasil pengiriman pada jarak 12 meter

Setelah mikrokontroler mengirimkan data berupa karakter masing masing 100, 150, dan 200 pada jarak 2 meter, 4 meter, 6 meter, 8 meter, 10 meter, dan 12 meter maka akan ditampilkan pada kotak textbox yang ada pada aplikasi *android* seperti pada gambar. Namun saat pengiriman dilakukan pada jarak 12 meter terjadi kesalah dalam hubungan *bluetooth*.

B. Pembahasan

Dari hasil pengujian ini didapatkan bahwa *android* dapat menerima data yang dikirimkan oleh mikrokontroler dan begitu juga sebaliknya mikrokontroler dapat menerima pengiriman. Percobaan yang dilakukan adalah pengiriman data karakter oleh mikrokontroler pada *android* dengan rentang waktu berbeda-beda

dalam banyak karakter dikirimkan yang berbeda-beda dan jarak yang berbeda-beda. Berikut adalah tabel hasil dari pengujian

Tabel 4.2. Hasil Pengujian Serial (jarak dan banyak karakter)

Jarak	100 Karakter	150 Karakter	200 Karakter
2 meter	1.8 sekon	2.0 sekon	2.4 sekon
4 meter	2.1 sekon	2.5 sekon	2.5 sekon
6 meter	2.4 sekon	2.9 sekon	3.3 sekon
8 meter	2.7 sekon	3.2 sekon	3.7 sekon
10 meter	3.0 sekon	3.7 sekon	4.2 sekon
12 meter	Gagal	Gagal	Gagal

Dari hasil pengujian serial terhadap banyak karakter dengan berbeda jarak didapatkan terdapat perbedaan waktu pengiriman data yang bertambah secara bertahap terhadap jarak yang lebih jauh dan banyak karakter yang lebih banyak. Asumsi berdasarkan hasil pengujian pengiriman serial adalah waktu pengiriman data yang bertambah lama tersebut karena kecepatan pengiriman merupakan 50 karakter/detik sehingga data yang dikirim dengan banyak karakter yang semakin besar dengan jarak yang semakin jauh akan membuat waktu pengiriman data tersebut menjadi semakin besar dan jarak pengiriman data maksimal adalah sebesar 10 meter.

4.2.2. Hasil Pengujian Program Tekanan Darah

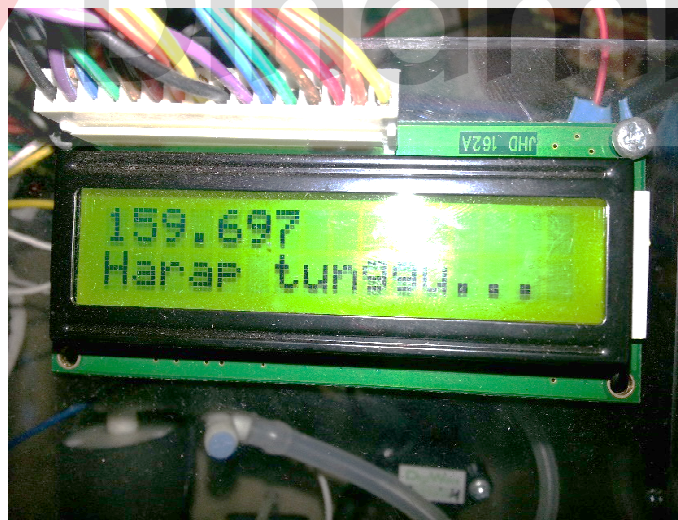
A. Hasil Pengujian

Hasil pengujian dari pengiriman serial ini dapat dilihat pada gambar 4.9 berikut. Pada gambar berikut akan di tampilkan proses pemompaan udara yang dilakukan pompa DC kepada manset, pada bagian ini pompa akan memberikan udara hingga tekanan udara yang ada dalam manset terukur 160mmHg.



Gambar 4.11. Proses pompa udara ke dalam manset

Setelah manset memompa udara hingga udara yang ada dalam manset bertekanan 160, pompa akan berhenti memompa udara dan melakukan kalibrasi seperti pada gambar 4.9.



Gambar 4.12. Proses kalibrasi

Dalam proses kalibrasi terdapat urutan-urutan instruksi yang dilakukan, yaitu menunggu selama sepuluh detik. Setelah sepuluh detik dilalui tekanan udara

akan dicatat sebagai tekanan sistolik, sedangkan tekanan diastolik adalah tekanan udara yang terukur setelah 5 detik dari tekanan sistolik terukur pada gambar 4.10.



Gambar 4.13. Tekanan darah terukur

Setelah tekanan darah sistolik dan diastolic berhasil didapatkan, maka proses yang harus dilakukan adalah menampilkan data tekanan darah tersebut pada LCD seperti pada gambar 4.10.

B. Pembahasan

Dari hasil pengujian ini didapatkan bahwa alat tekanan darah dapat mengukur tekanan darah seseorang dengan menampilkan tekanan sistolik dan diastolik. Proses Pengukuran tekanan darah dilakukan dalam waktu 3 menit.

4.3. Hasil Pengujian Sistem Secara Keseluruhan

Hasil pengujian keseluruhan dari sistem ini terbagi atas dua hasil percobaan. Percobaan pertama menggunakan alat yang penulis rancang, dan percobaan kedua menggunakan tensimeter sebagai acuan. Percobaan yang dilakukan juga dengan menggunakan 20 orang sebagai sampel percobaan.

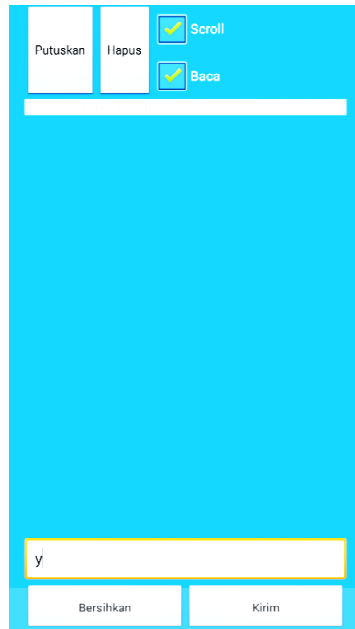
4.3.1. Hasil Pengujian Sampel Pertama

A. Hasil Pengujian

Pada hasil pengujian sampel pertama diambil gambar pengambilan data, dan hasilnya dapat dilihat sebagai berikut.



Gambar4.14. Pengambilan data tensimeter acuan sampel 1



Gambar 4.15. Pengambilandata tensimeter rancangan sampel 1

Adapun hasil dari nilai tekanan darah terukur akan di tampilkan pada tabel 4.2 berikut. Pada pengujian sampel pertama ini dilakukan pada orang yang sama.

Tabel 4.3.Perbandingan tensimeter acuan dan rancangan pada sampel 1

Tensimeter Rancangan	Tensimeter acuan
Sistolik : 119	Sistolik : 111
Diastolik : 77	Diastolik : 73

B. Pembahasan

Dari hasil uji coba yang telah dilakukan maka didapatkan perbedaan hasil pengambilan data pada sampel pertama dapat dihitung sebagai berikut.

Perbedaan Systolik = $119 - 111 = 8 \text{ mmHg}$

Perbedaan Diastolik = $77 - 73 = 4 \text{ mmHg}$

Pada hasil pengujian sistem secara keseluruhan ini nilai perbedaan data untuk sistolik adalah 8mmHg . Dan nilai perbedaan data untuk diastolic adalah 4 mmHg.

4.3.2. Hasil Pengujian Sampel Kedua

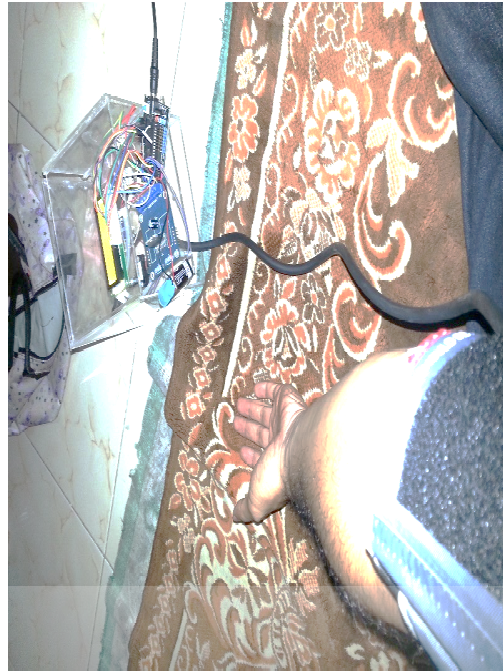
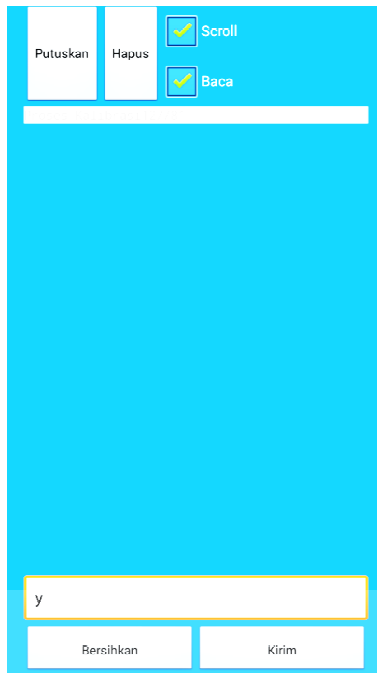
A. Hasil Pengujian

Pada hasil pengujian sampel pertama diambil gambar pengambilan data, dan hasilnya dapat dilihat sebagai berikut.



Gambar 4.16. Pengambilan data

tensimeter acuan sampel 2



Gambar 4.17. Pengambilandata tensimeter rancangan sampel 2

Adapun hasil dari nilai tekanan darah terukur akan di tampilkan pada tabel 4.3 berikut. Pada pengujian sampel pertama ini dilakukan pada orang yang sama.

Tabel 4.4.Perbandingan tensimeter acuan dan rancangan pada sampel 2

Tensimeter Rancangan	Tensimeter acuan
Sistolik : 127	Sistolik : 120
Diastolik : 81	Diastolik : 77

B. Pembahasan

Dari hasil uji coba yang telah dilakukan maka didapatkan perbedaan hasil pengambilan data pada sampel pertama dapat dihitung sebagai berikut.

$$\text{Perbedaan Systolik} = 127 - 120 = 7 \text{ mmHg}$$

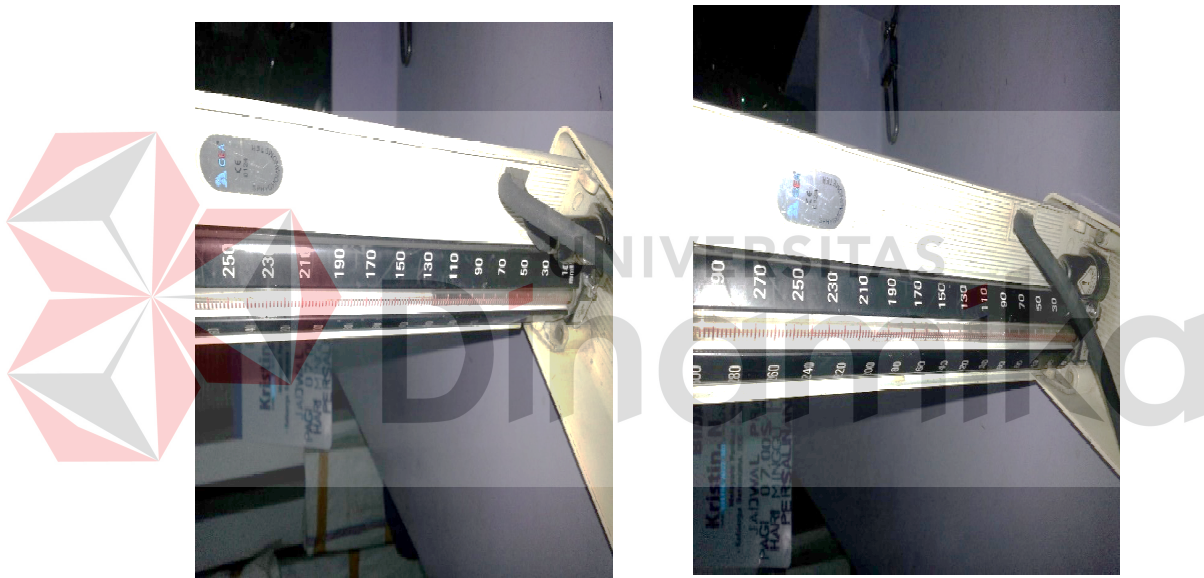
$$\text{Perbedaan Diastolik} = 81 - 77 = 4 \text{ mmHg}$$

Pada hasil pengujian sistem secara keseluruhan ini nilai perbedaan data untuk sistolik adalah 7 mmHg. Dan nilai perbedaan data untuk diastolic adalah 4 mmHg.

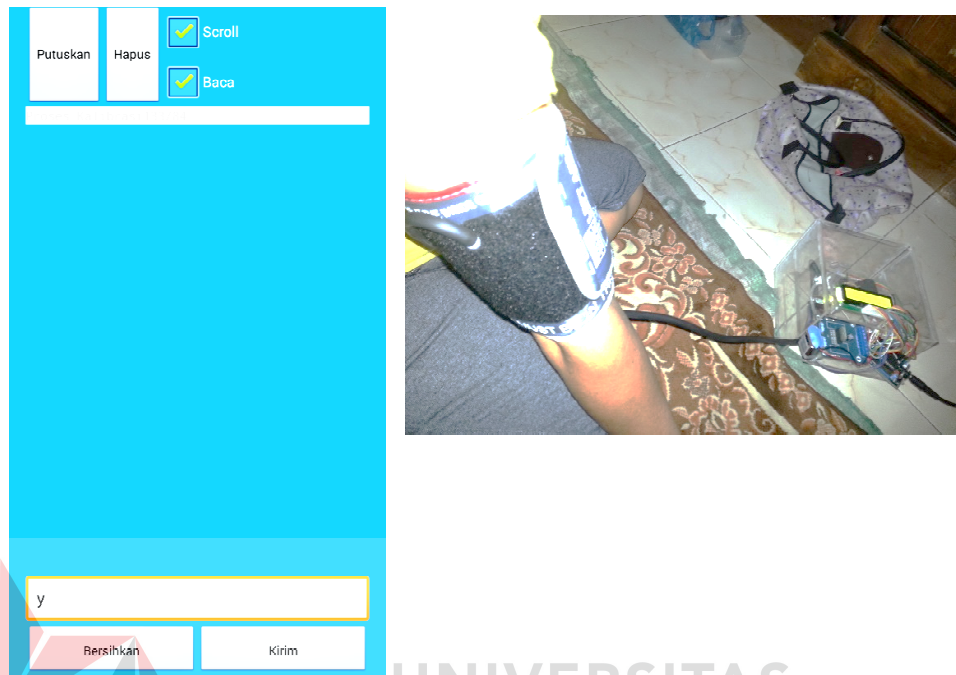
4.3.3. Hasil Pengujian Sampel Ketiga

A. Hasil Pengujian

Pada hasil pengujian sampel pertama diambil gambar pengambilan data, dan hasilnya dapat dilihat sebagai berikut.



Gambar4.18. Pengambilan data tensimeter acuan sampel 3



Gambar 4.19. Pengambilandata tensimeter rancangan sampel 3

Adapun hasil dari nilai tekanan darah terukur akan di tampilkan pada tabel

4.4 berikut. Pada pengujian sampel pertama ini dilakukan pada orang yang sama.

Tabel 4.5.Perbandingan tensimeter acuan dan rancangan pada sampel 3

Tensimeter Rancangan	Tensimeter acuan
Sistolik : 133	Sistolik : 129
Diastolik : 84	Diastolik : 89

B. Pembahasan

Dari hasil uji coba yang telah dilakukan maka didapatkan perbedaan hasil pengambilan data pada sampel pertama dapat dihitung sebagai berikut.

$$\text{Perbedaan Systolik} = 133 - 129 = 4 \text{ mmHg}$$

Perbedaan Diastolik = $89 - 84 = 5$ mmHg

Pada hasil pengujian sistem secara keseluruhan ini nilai perbedaan data untuk sistolik adalah 4mmHg . Dan nilai perbedaan data untuk diastolic adalah 5 mmHg.

4.3.4. Hasil Pengujian Sampel Keempat hingga Sampel Keduapuluh

A. Hasil Pengujian

Pada hasil pengujian sampel pertama diambil gambar pengambilan data, dan hasilnya dapat dilihat sebagai berikut.

Tabel 4.6. Pengambilan data tensimeter acuan dan rancangan sampel 4 - 20

No	Tensimeter acuan		Tensimeter Rancangan		Selisih		Prosentase Selisih	
	Sistolik	Diastolik	Sistolik	Diastolik	Sistolik	Diastolik	Sistolik	Diastolik
4	127	83	129	88	2	5	1.574803	6.024096
5	120	90	122	90	2	0	1.666667	0
6	135	81	141	87	6	6	4.444444	7.407407
7	133	93	135	94	2	1	1.503759	1.075269
8	128	85	128	89	0	4	0	4.705882
9	115	84	121	87	6	3	5.217391	3.571429
10	139	90	145	89	6	1	4.316547	1.111111
11	122	81	123	86	1	5	0.819672	6.17284
12	136	87	139	90	3	3	2.205882	3.448276
13	115	79	119	83	4	4	3.478261	5.063291
14	120	90	130	89	10	1	8.333333	1.111111
15	129	77	133	87	4	10	3.100775	12.98701
16	110	88	115	85	5	3	4.545455	3.409091
17	111	81	119	85	8	4	7.207207	4.938272
18	126	80	129	84	3	4	2.380952	5
19	144	94	145	88	1	6	0.694444	6.382979
20	117	75	124	80	7	5	5.982906	6.666667

4.4 Pembahasan Keseluruhan

Dari hasil pengujian seperti pada tabel 4.3 sampai 4.9 maka didapatkan rata-rata kesalahan data pengukuran tekanan darah pada alat ini adalah sebagai berikut.

Tabel 4.7. Pengambilan data tensimeter acuan dan rancangan sampel 1 - 20

No	Tensimeter acuan		Tensimeter Rancangan		Selisih		Prosentase Selisih (%)	
	Sistolik	Diastolik	Sistolik	Diastolik	Sistolik	Diastolik	Sistolik	Diastolik
1	111	73	119	77	8	4	7.207207	5.479452
2	120	77	127	81	7	4	5.833333	5.194805
3	129	89	133	84	4	5	3.100775	5.617978
4	127	83	129	88	2	5	1.574803	6.024096
5	120	90	122	90	2	0	1.666667	0
6	135	81	141	87	6	6	4.444444	7.407407
7	133	93	135	94	2	1	1.503759	1.075269
8	128	85	128	89	0	4	0	4.705882
9	115	84	121	87	6	3	5.217391	3.571429
10	139	90	145	89	6	1	4.316547	1.111111
11	122	81	123	86	1	5	0.819672	6.17284
12	136	87	139	90	3	3	2.205882	3.448276
13	115	79	119	83	4	4	3.478261	5.063291
14	120	90	130	89	10	1	8.333333	1.111111
15	129	77	133	87	4	10	3.100775	12.98701
16	110	88	115	85	5	3	4.545455	3.409091
17	111	81	119	85	8	4	7.207207	4.938272
18	126	80	129	84	3	4	2.380952	5
19	144	94	145	88	1	6	0.694444	6.382979
20	117	75	124	80	7	5	5.982906	6.666667
Jumlah					89	78	73.61382	95.36697
Rata-Rata					4.45	3.9	3.680691	4.768348

Rata-rata kesalahan sistolik

$$\begin{aligned}
 & \frac{8 + 7 + 4 + 2 + 2 + 6 + 2 + 0 + 6 + 6 + 1 + 3 + 4 + 10 + 4 + 5 + 8 + 3 + 1 + 7}{20} \\
 & = 4,45 \text{ mmHg}
 \end{aligned}$$

Rata-rata kesalahan diastolik

$$\frac{4 + 4 + 5 + 5 + 0 + 6 + 1 + 4 + 3 + 1 + 5 + 3 + 4 + 1 + 10 + 3 + 4 + 4 + 6 + 5}{20} = 3,9 \text{ mmHg}$$

Dari prosentase ke-20 sampel pengukuran nilai tekanan darah dengan menggunakan tensimeter rancangan maka dapat di rumuskan nilai rata-rata *error* dalam pembacaan nilai tekanan darah adalah sebagai berikut :

$$\text{Rata-rata nilai } errorsistolik = \frac{1 + 2 + \dots + 20}{20} = 3,687 \%$$

$$\text{Rata-rata nilai } error \text{ diastolik} = \frac{1 + 2 + \dots + 20}{20} = 4,768 \%$$

Dari hasil rata-rata kesalahan nilai sistolik dan diastolik tersebut dapat diketahui nilai standar deviasi dari sistolik dan diastolik adalah sebagai berikut :

$$\text{Standar deviasi kesalahan sistolik } s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2} = 2,742$$

$$\text{Standar deviasi kesalahan diastolik } s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2} = 2,221$$

Pada hasil pengujian sistem secara keseluruhan ini nilai rata-rata *error* pengukuran didapatkan dari perhitungan secara matematis yang sesuai dengan persamaan diatas untuk ketiga sampel data yang ada. Rata-rata *error* pengukuran sistolik adalah sebesar 3,687 % dan rata-rata *error* pengukuran diastolik adalah sebesar 4,768 %. Standar deviasi dari kesalahan pengukuran sistolik adalah 2.742 % dan standar deviasi dari kesalahan pengukuran diastolik adalah 2.221%. *Error* yang didapatkan berdasarkan pengujian terhadap tensimeter rancangan yang menggunakan metode pengukuran tekanan darah osilometri terhadap tensimeter acuan yang menggunakan pengukuran tekanan darah korotkoff(auskulasi).

Metode osilometri adalah metode Proses pengukuran perubahan tekanan udara yang berada didalam manset yang disebabkan oleh tekanan dari nadi, sedangkan metode korotkoff adalah metode yang memanfaatkan sifat dari bunyi serasi yang diciptakan oleh nadi/jalan utama darah pada saat manset terkembang dan menekan nadi pada lengan bisa dihubungkan dengan tekanan sistolik dan diastolic. Dengan adanya perbedaan metode dalam pengukuran tekanan darah tersebut membuat adanya *error* dalam perbandingan hingga 10 mmHg dalam membandingkan tensimeter rancangan dan tensimeter acuan.



UNIVERSITAS
Dinamika

BAB V

PENUTUP

Berdasarkan hasil pengujian dan evaluasi pada perangkat keras dan perangkat lunak yang dipergunakan dalam perancangan rancang bangun system pemantau dan transmisi data tekanan darah pada *mobile android* menggunakan koneksi *bluetooth* ini, maka dapat diambil beberapa kesimpulan dan saran sebagai berikut.

5.1 Kesimpulan

1. Pengujian pengiriman data serial memiliki waktu pengiriman data yang bertambah secara bertahap terhadap jarak yang lebih jauh. Asumsi dari waktu pengiriman data yang bertambah lama tersebut adalah dengan kecepatan pengiriman data 50 karakter/detik sehingga data yang dikirim dengan banyak karakter yang semakin besar dengan jarak yang semakin jauh akan membuat waktu pengiriman data tersebut menjadi semakin besar.
2. Pengujian alat pengukur tekanan darah menunjukan bahwa alat ini dapat berkomunikasi dengan *mobile android*, baik menerima data maupun mengirimkan data hasil pengukuran tekanan darah.
3. Rata-rata kesalahan pengukuran alat pengukur tekanan darah untuk pengukuran sistolik sebesar 3.687% dengan standar deviasi sebesar 2.742, dan untuk pengukuran diastolik 4.768% dengan standar deviasi sebesar 2.221.

5.2 Saran

Agar pada penelitian selanjutnya alat ini dapat dikembangkan lebih sempurna, maka penulis memberikan saran:

1. Penggunaan metode korotkoff (auskultasi) yang dipadukan dengan ATmega32 dengan sensor MPX5100DP untuk membandingkan ketepatan dari hasil pengukuran.
2. Pengembangan dalam pengukuran tekanan darah diatas 160 mmHg sehingga dapat mengukur tekanan darah semua orang.
3. Agar dapat digunakan produksi masal, dapat ditambahkan sistem penampung data (database) untuk mempermudah dalam memeriksa *history* dari rekam medis dalam pengukuran tekanan darah seseorang.
4. Pada aplikasi *android* dapat ditambahkan indikator tentang tekanan darah yang terukur apakah berada pada keadaan normal atau berada pada keadaan yang terlalu tinggi maupun terlalu rendah.

DAFTAR PUSTAKA

Adiludung, Johan dkk. 2011.

Alat Pengukur Tekanan Darah Otomatis Berbasis Mikrokontroler Untuk Pasien Rawat Jalan dengan SMS Gateway. Surabaya: ITS.

Android Developers. 2010. **Android 2.3**

Platform, <URL: <http://developer.android.com/sdk/android-2.2.html>>

Innovative Electronics. 2010: **SPC (Smart Peripheral Controller) Blue**

Link, <URL: http://www.innovativeelectronics.com/innovative_electronics/download_files/manual/manual%20SPC%20Blue-link.pdf>

Hendrantoro, Gamantyo. 2008. **“Teknologi Komunikasi Nirkabel: Perkembangan Terkin dan Peluang Indonesia”.**

Pidato Pengukuhan untuk Jabatan Guru

Besar Dalam Bidang Sistem Komunikasi Nirkabel.

Oktavianto, Hary. **Implementasi Perhitungan Desimal menggunakan PGA.**

Politeknik Elektronika Surabaya: Surabaya

Soot W. N well, Ipswich, dan Mass. 1993. **Pressure Signal Processing Gauge.**

United US Patient Document: United

R.W. Zappe. **Valve Selection Handbook Fourth Edition. Gulf Professional Publishing, Houston, 1998.**

MPX5100DP Datasheet. (2014, 24 Oktober). **Datasheet MPX5100DP.** Diperoleh

24 Oktober 2014, dari www.alldatasheet.com/MOTOROLA/MPX5100.html

Wijaya Budi Agung, 2010, Proyek Tugas Akhir

: **Rancang Bangun Alat Pengukur Detak Jantung dan Suhu Tubuh Manusia Ber**

basis Komunikasi Bluetooth. D3 Teknik Telekomunikasi. PENS-ITS:

Surabaya

HC-05 Datasheet. (2014, 24 Oktober). **Datasheet HC-05.** Diperoleh

24 Oktober 2014, dari www.alldatasheet.com/datasheet/3042/MOTOROLA/HC

05.html

LCD 16 x 2 Datasheet. (2014, 24 Oktober). **Datasheet LCD 16 x 2.** Diperoleh

24 Oktober 2014, dari www.alldatasheet.com/datasheet/571342/LUMEX/LCM-

S01602DTRM.html

Basic4android. (2014, 24 Oktober). **Basic4android Questions.** Diperoleh

24 Oktober 2014, dari www.basic4ppc.com/android/forum

Driver Motor H-Bridge Transistor. (2014, 24 Oktober) **Rangkaian Driver Motor.**

Diperoleh 24 oktober 2014, dari [www.toko-](http://www.toko-elektronika.com/tutorial/pcinterfacing.htm)

[elektronika.com/tutorial/pcinterfacing.htm](http://www.toko-elektronika.com/tutorial/pcinterfacing.htm)