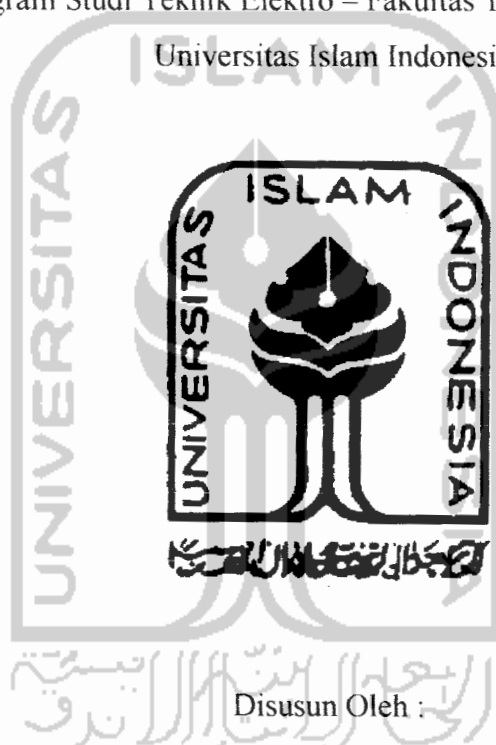


**SISTEM PENGENDALI KECEPATAN MOTOR DC
MELALUI TELEPON SELULER
BERBASIS MIKROKONTROLER AT90S2313**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana teknik
Program Studi Teknik Elektro – Fakultas Teknologi Industri

Universitas Islam Indonesia



Disusun Oleh :

NAMA : Septa Kaurdiansyah

NIM : 00 524 065

**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2007

LEMBAR PENGESAHAN PEMBIMBING

TUGAS AKHIR

**SISTIM PENGENDALI KECEPATAN MOTOR DC
MELALUI TELEPON SELULER
BERBASIS MIKROKONTROLER AT90S2313**



Disusun Oleh:

NAMA : Septa Kaurdiansyah

NIM : 00 524 065

Disetujui :

Yogyakarta, / / 2006

Pembimbing I

Pembimbing II

A handwritten signature in black ink, appearing to be 'Tito Yuwono'.

(Tito Yuwono, ST, MSc)

A handwritten signature in black ink, appearing to be 'Medilla Kusriyanto'.

(Medilla Kusriyanto,ST)

LEMBAR PENGESAHAN PENGUJI
TUGAS AKHIR
SISTIM PENGENDALI KECEPATAN MOTOR DC
MELALUI TELEPON SELULER
BERBASIS MIKROKONTROLER AT90S2313

Disusun Oleh :

NAMA : Septa Kaurdiansyah

NIM : 00 524 065

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Elektro
Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 2006

Tim Penguji

Tito Yuwono, ST. M.Sc.

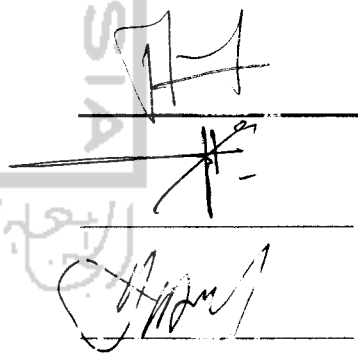
Ketua

Medilla Kusriyanto, ST

Anggota I

Yusuf Aziz Amrulloh, ST

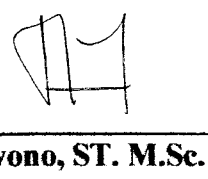
Anggota II



Mengetahui,

Ketua Jurusan Teknik Elektro – Fakultas Teknologi Industri

Universitas Islam Indonesia



Tito Yuwono, ST. M.Sc.

DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN PEMBIMBING	ii
HALAMAN PENGESAHAN PENGUJI	iii
HALAMAN PERSEMBAHAN.....	iv
HALAMAN MOTTO	v
KATA PENGANTAR.....	vi
ABSTRAK	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
 BAB I PENDAHULUAN.....	 1
1.1. Latar Belakang.....	1
1.2. Perumusan Masalah	1
1.3. Batasan Masalah	2
1.4. Tujuan Penelitian	2
1.5. Metode Penelitian	2
1.5.1. Studi pustaka	3
1.5.2. Perancangan sistem.....	3
1.5.3. Pengujian alat	3
1.5.4. Implementasi alat.....	3
1.6. Sistematika Penulisan	3

HALAMAN PERSEMBAHAN

Tugas akhir ini penulis persembahkan untuk orang - orang yang telah turut terlibat dalam proses pengerjaannya :

- ✦ Kedua orang tua-ku, Ayah & Ibu tercinta atas kasih sayang, doa serta perhatian yang tak pernah habis dan tak kenal lelah dalam membesarkan dan mendidik putranya ini.
- ✦ Kakak-kakakku yang tersayang dengan sadar diri dan penuh kerelaan memberikan kucuran dana tanpa banyak tanya, terima kasih walau kantong kalian kering.
- ✦ Anak-anak kost yang tidak ada satupun dari kalian memberikan dukungan namun setidaknya hati tenang tanpa banyak gangguan.



HALAMAN MOTTO

"Allah meninggikan orang – orang yang beriman di antara kamu dan orang – orang yang diberi ilmu pengetahuan beberapa derajat"

(QS Mujaadilah [58]:11)

"Maka, apabila kamu telah selesai (dari suatu urusan), kerjakanlah dengan sungguh – sungguh (urusan) yang lain dan hanya kepada Tuhanmulah hendaknya kamu berharap"

(QS Al Insyirah [94] : 7- 8)

"Allah tidak akan membebani seseorang, kecuali sesuai dengan kesanggupannya. Ia mendapat pahala (dari kebaikan) yang dikerjakannya dan Ia mendapatkan siksa (dari kejahatan) yang dikerjakannya"

(QS Al Baqarah [2]:286)

"Hai orang – orang yang beriman, jadikanlah sabar dan sholat sebagai penolong kalian. Sesungguhnya Allah beserta orang yang sabar"

(QS Al Baqarah [2] : 153)

"Jagalah hati jangan kau kotori..".

(Aa GYM)

"don't waste u'r life becoming a loser that ain't worth caus I've been there".

(syah)

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillah, puji syukur penulis panjatkan kepada Alloh swt. yang telah memberikan nikmat yang sangat besar, yang mana diantaranya penulis masih diberi umur panjang, sehingga dapat menyelesaikan laporan tugas akhir yang berjudul “SISTIM PENGENDALI KECEPATAN MOTOR DC MELALUI TELEPON SELULER BERBASIS MIKROKONTROLER AT90S2313”.

Dewasa ini perkembangan dunia telekomunikasi sangat pesat khususnya telekomunikasi menggunakan telepon seluler. Perkembangan ini disebabkan karena kemudahan dan efektifitas yang tinggi dari penggunaan telepon seluler. Aplikasi pengendalian dengan memanfaatkan teknologi telepon seluler juga berkembang dengan sangat pesat. Sebagai contoh aplikasi telepon seluler untuk mengatur kecepatan motor DC. Kelebihan dari sistim pengendalian motor DC menggunakan telepon seluler adalah pengendalian dapat dilakukan dari sembarang tempat dimana saja dan kapan saja.

Mikrokontroler yang penulis gunakan pada tugas akhir ini adalah mikrokontroler AT90S2313 yang merupakan sebuah mikrokontroler dengan arsitektur RISC (*Reduce Instruction Set Computer*) 8 bit yang dikeluarkan Atmel Corporation. Fitur inti AVR adalah 32 buah register identik 8-bit yang memiliki kemampuan dan sifat seperti sebuah *accumulator*.

Untuk lebih jelasnya tentang gambaran keseluruhan laporan, penulis telah menyusun dalam bentuk laporan yang cukup sederhana ini, dimana didalamnya memaparkan keseluruhan aplikasi pengendalian kecepatan motor DC yang kami ambil sebagai topik dalam pembahasan buku laporan yang kami susun.

Mudah-mudahan laporan (berbentuk buku) yang penulis susun secara sederhana ini, dapat bermanfaat dan dapat dijadikan referensi sebagai pembanding dalam pembuatan laporan lain yang lebih sempurna dalam kasus dan tofik yang sama.

Harapan dari kami sebagai penyusun adalah saran beserta solusi dari pembaca yang memahami permasalahan yang ada dalam laporan, karena hal tersebut sangatlah

membantu dalam memajukan daya pikir kami terutama tentang permasalahan yang dibahas sebagai isi laporan dalam buku ini.

Ucapan terima kasih penulis sampaikan kepada:

- Bapak Tito Yuwono, ST, Msc selaku Dosen Pembimbing I
- Bapak Medilla Kusriyanto, ST selaku Dosen Pembimbing II
- Bapak Yusuf Aziz Amrulloh, ST selaku Dosen Penguji
- Semua pihak yang berkenan membantu pembuatan laporan tugas akhir.

Yogyakarta, 12 Desember 2006

Penyusun



Septa Kaurdiansyah

BAB II DASAR TEORI.....	5
2.1. Mikrokontroler AVR AT90S2313.....	5
2.1.1. Arsitektur Mikrokontroler AVR AT90S2313	6
2.1.2. Konfigurasi Pin AT90S2313	7
2.1.3. Memori AT90S2313	8
2.1.4. Perangkat Lunak Mikrokontroler AT90S2313	9
2.1.5. UART (<i>Universal Asynchronous Receiver Transmitter</i>).....	10
2.1.5. <i>Timer/Counter</i>	10
2.2. Telepon Seluler <i>Siemens M35</i>	12
2.3. <i>AT Command</i>	13
2.4. PDU (<i>Protocol Data Unit</i>).....	13
2.4.1. PDU untuk mengirim SMS ke SMS-Center.....	14
2.4.1. PDU untuk menerima SMS dari SMS-Center.....	18
2.5. Transistor Bipolar	19
2.5.1. Kurva Bias.....	20
2.4.1. Kurva Kolektor	20
2.6. PWM (<i>Pulse Width Modulation</i>).....	22
 BAB III PERANCANGAN SISTEM.....	 26
3.1. Diagram Blok Sistem.....	26
3.2. Perancangan Perangkat Keras	27
3.2.1. Rangkaian Pendeteksi Dering SMS	27
3.2.2. Rangkaian <i>H-Bridge</i> Pengerak Motor DC	28

3.3. Perancangan Perangkat Lunak	31
3.3.2. Diagram Alir Program.....	31
3.3.2. Inisialisasi UART	32
3.3.2. Inisialisasi <i>PWM</i>	33
3.3.4. Mendeteksi dan membaca SMS.....	34
3.3.5. Mengatur kecepatan motor DC.....	35
3.3.6. Konversi format teks ke PDU.....	35
3.3.7. Memprogram AT90S2313	37
 BAB IV PENGUJIAN DAN ANALISA	41
4.1. Pengujian Alat	41
4.2. Pengamatan Pesan SMS.....	42
4.3. Pengamatan Rangkaian Deteksi Dering.....	44
4.4. Analisa Pengaturan Kecepatan Motor DC	45
4.5. Pengamatan sinyal PWM	48
4.6. Kalibrasi alat.....	53
 BAB V PENUTUP	55
5.1. Kesimpulan.....	55
5.2. Saran	55

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR TABEL

Tabel 2.1. Timer/Counter1 Control Register A (TCCR1A)	11
Tabel 2.2. Fungsi Compare mode pada mode PWM	11
Tabel 2.3. Pemilihan mode PWM.....	11
Tabel 2.4. Kode PDU dan no SMS-Center Nasional.....	15
Tabel 2.5. Kode PDU dan no SMS-Center Internasional	15
Tabel 2.6. Rumus hitung validitas SMS.....	17
Tabel 2.7. Skema 7 bit alphabet.....	18
Tabel 3.1. Register UCR (<i>UART Control Register</i>).....	32
Tabel 3.2. Data konversi ke kode 7 bit.....	36
Tabel 3.3. Data konversi 7bit ke 8 bit	36
Tabel 4.1. Data Hasil Pengamatan Alat	42
Tabel 4.3. Sinyal pengamatan rangkaian deteksi dering.....	45
Tabel 4.3. Data kecepatan motor.....	46
Tabel 4.4. Selisih perhitungan data kecepatan.....	48
Tabel 4.5. Bentuk gelombang PWM.....	49
Tabel 4.6. Perbandingan kecepatan motor antara alat dan tachometer	53

DAFTAR GAMBAR

Gambar 2.1. Arsitektur mikrokontroler AT90S2313	6
Gambar 2.2. Konfigurasi pin mikrokontroler AT90S2313	7
Gambar 2.3. Susunan Pin External Siemens M35 tampak bawah	12
Gambar 2.4. Transistor npn dan pnp	21
Gambar 2.5. Kurva $I_B - V_{BE}$	22
Gambar 2.6. Kurva kolektor	22
Gambar 2.6. Pulsa PWM dengan duty cycle 50 %	24
Gambar 3.1. Blok Diagram sistem	26
Gambar 3.2. Rangkaian pendeteksi dering SMS	27
Gambar 3.3. Rangkaian H-bridge penggerak motor DC	28
Gambar 3.4. Diagram alir program	31
Gambar 3.5. Tampilan menu AVR studio	37
Gambar 3.6. Tampilan select new project	37
Gambar 3.7. Tampilan <i>new project</i>	38
Gambar 3.8. Tampilan output compiler	38
Gambar 3.9. Koneksi antara PC printer port dengan port mikrokontroler....	39
Gambar 3.10. Tampilan HexBin2.exe	39
Gambar 3.11. Tampilan <i>chip successfully programed</i>	40
Gambar 4.1. Tampilan alat hasil perancangan	41
Gambar 4.2. Titik pengukuran pada rangkaian deteksi dering	44
Gambar 4.3. Rangkaian pengukuran sinyal PWM	48

ABSTRAK

Aplikasi pengendalian dengan memanfaatkan teknologi telepon seluler sekarang ini berkembang dengan sangat pesat. Hal itu disebabkan karena kelebihan dari sistem pengendalian menggunakan telepon seluler adalah dapat dilakukan dari sembarang tempat dan proses pengendaliannya lebih cepat dan akurat. Maka atas dasar tersebut dalam perancangan tugas akhir ini dibahas mengenai sistem pengendali kecepatan motor DC melalui telepon seluler berbasis mikrokontroler AT90S2313. Pengendalian dilakukan dengan mengirimkan sebuah pesan SMS kenomor handphone pada sistem. Format SMSnya adalah <Arah putaran> spasi <Kecepatan putaran>. Arah putarannya ada 2 yaitu Kanan dan Kiri. Sedangkan kecepatan putaran dari 100 rpm ~ 900 rpm dengan kelipatan setiap 100 rpm. Untuk mengatur kecepatan putaran motor digunakan metode PWM (*Pulse Width Modulation*). Teknik PWM untuk pengaturan kecepatan motor adalah teknik pengaturan kecepatan motor dengan cara merubah-ubah besarnya *duty cycle* pulsa. Besarnya amplitudo dan frekuensi pulsa adalah tetap, sedangkan besarnya *duty cycle* berubah-ubah sesuai dengan kecepatan yang diinginkan, semakin besar *duty cycle* maka semakin cepat pula kecepatan motor, dan sebaliknya semakin kecil *duty cycle* maka semakin pelan pula kecepatan motor. Arah putaran motor kekanan dan kekiri ditentukan dari polaritas yang diberikan pada kedua pin motor DC. Dari perancangan yang sudah dilakukan didapat selisih kecepatan rata-rata sebesar 4.62 sehingga dapat disimpulkan bahwa perancangan alat yang dilakukan dapat mengatur kecepatan motor DC untuk setiap perintah SMS yang berbeda.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dewasa ini perkembangan dunia telekomunikasi sangat pesat khususnya telekomunikasi menggunakan telepon seluler. Perkembangan ini disebabkan karena kemudahan dan efektifitas yang tinggi dari penggunaan telepon seluler. Aplikasi pengendalian dengan memanfaatkan teknologi telepon seluler juga berkembang dengan sangat pesat. Sebagai contoh aplikasi telepon seluler untuk mengatur kecepatan motor DC. Pengendalian kecepatan motor DC yang ada dipasaran pada umumnya menggunakan pengendalian secara manual yaitu dengan mengatur tombol-tombol pengendalinya. Pengendalian motor DC secara manual ini sangat terbatas terhadap tempat dan waktu.

Berdasarkan kelemahan-kelemahan pada sistem pengendalian motor DC pada umumnya maka pada tugas akhir ini dibahas mengenai sistem pengendalian kecepatan motor DC melalui telepon seluler. Kelebihan dari sistem pengendalian motor DC menggunakan telepon seluler adalah pengendalian dapat dilakukan dari sembarang tempat dimana saja dan kapan saja.

1.2. Perumusan Masalah

Berdasarkan latar belakang diatas maka dapat ditarik rumusan masalah, “Bagaimana merancang Sistem pengendali kecepatan motor DC melalui telepon seluler berbasis mikrokontroler AT90S2313”.

1.3. Batasan Masalah

Perancangan sistem pengendali kecepatan motor DC melalui telepon seluler berbasis mikrokontroler AT90S2313 ini memiliki beberapa batasan masalah seperti berikut:

1. Sistem pengendali kecepatan motor DC dibatasi hanya dapat mengatur kecepatan dari motor DC dengan mode PWM
2. Kecepatan putar motor DC yang dapat diatur dari 100 rpm – 900 rpm dengan kelipatan kecepatan tiap 100 rpm

1.4. Tujuan Penelitian

Tujuan yang ingin dicapai pada perancangan alat pengendali kecepatan motor DC melalui telepon seluler berbasis mikrokontroler AT90S2313 sebagai berikut:

1. Membuat alat yang dapat mengatur kecepatan motor DC dari 100 rpm – 900 rpm melalui SMS.
2. Mengaplikasikan ilmu-ilmu yang telah diperoleh di bangku kuliah.
3. Dapat dikembangkan dan dimanfaatkan oleh masyarakat.

1.5. Metode Penelitian

Untuk mencapai hasil yang baik dalam perancangan perangkat keras dan perangkat lunak pengendali kecepatan motor DC melalui telepon seluler, maka perlu dilakukan beberapa metodologi penelitian sebagai berikut:

1.5.1. Studi pustaka

Studi pustaka digunakan untuk mendapatkan informasi yang berkaitan dengan perancangan alat seperti teori tentang pemrograman mikrokontroler AVR AT90S2313, teori tentang perintah *AT Command* telepon seluler *Siemens M35*, dan teori-teori pendukung lainnya.

1.5.2. Perancangan sistem

Perancangan sistem merupakan perancangan alat yang menjadi obyek penelitian. Perancangan sistem meliputi perancangan *hardware* dan *software*.

1.5.3. Pengujian alat

Pengujian alat dilakukan dengan cara menjalankan alat dan memastikan apakah alat sudah berfungsi sesuai dengan perancangan.

1.5.4. Implementasi alat

Implementasi alat dilakukan dengan cara melakukan pengendalian kecepatan motor dengan cara mengirimkan SMS ke alat.

1.6. Sistematika Penulisan

Sistematika penulisan tugas akhir dengan judul “Sistem pengendali kecepatan motor DC melalui telepon seluler berbasis mikrokontroler AT90S2313” adalah sebagai berikut :

BAB I Pendahuluan

Pada bab 1 membahas tentang latar belakang, perumusan masalah, batasan masalah, tujuan penulisan, metodologi penelitian dan sistematika penulisan.

BAB II Dasar Teori

Bab II ini membahas semua teori dan rumus yang merupakan acuan perancangan dan pembuatan rangkaian yang meliputi penjelasan tentang gambar alat, sifat, karakteristik, dan kegunaan komponen-komponen pendukung.

BAB III Perancangan Perangkat Keras dan Perangkat Lunak

Bab III membahas diagram blok sistem dan prinsip kerja masing-masing bagian alat yang dirancang secara lengkap dalam serta pembahasan tentang diagram alir program dan perancangan perangkat lunak.

BAB IV Pembahasan dan Analisa

Pada bab IV ini akan membahas dan menganalisa rangkaian perangkat keras dan hasil yang diperoleh dari perancangan.

BAB V Penutup

Merupakan bab akhir penyusunan laporan tugas akhir, yang berisi tentang kesimpulan dari hasil pengamatan dan saran-saran dari penyusun guna perbaikan dan pengembangan alat ini.

BAB II

DASAR TEORI

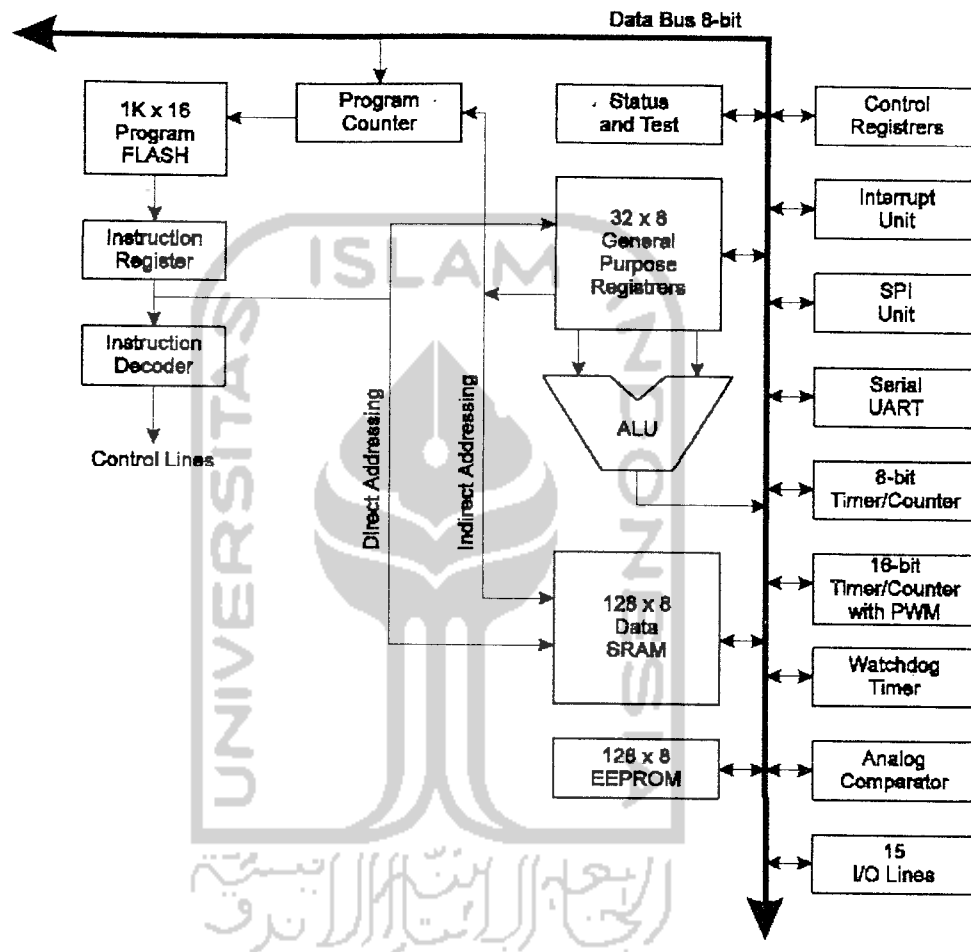
2.1. Mikrokontroler AVR AT90S2313

AVR merupakan sebuah mikrokontroler dengan arsitektur RISC (*Reduce Instruction Set Computer*) 8 bit yang dikeluarkan Atmel Corporation. Sampai saat ini mikrokontroler keluarga AVR adalah ATTinyxx, ATmegaxxx, dan AT90Sxxxx. AVR dikembangkan pertama kali oleh para peneliti di Trondheim, Norwegia, sebelum mereka diakuisisi oleh Atmel di tahun 1995. Fitur inti AVR adalah 32 buah register identik 8-bit yang memiliki kemampuan dan sifat seperti sebuah *accumulator*. AVR dikembangkan berdasarkan arsitektur Harvard, yang memisahkan antara memori program dan memori data. Keunggulan arsitektur Harvard adalah kecepatan eksekusi yang tinggi. Susunan instruksinya bertipe RISC, tetapi variasinya cukup banyak. Dengan desain seperti itu, memungkinkan dihasilkan program yang efisien (sependek mungkin) dengan waktu eksekusi yang sangat cepat (kebanyakan dalam satu *clock*) dengan panjang tiap instruksi rata-rata 16 bit.

Fitur yang dimiliki masing-masing seri keluarga AVR sedikit berbeda. Perbedaan ini menyebabkan beberapa instruksi tidak kompatibel antara satu seri dengan seri yang lain. Selain itu detail arsitekturnya juga sedikit berbeda, perbedaan tersebut di atas bukan sesuatu yang mendasar tetapi merupakan konsekuensi logis yang timbul dari perbedaan fitur masing-masing seri.

2.1.1. Arsitektur Mikrokontroler AVR AT90S2313

Arsitektur mikrokontroler keluarga AVR AT90S2313 ditunjukkan dalam Gambar 2.1. berikut.



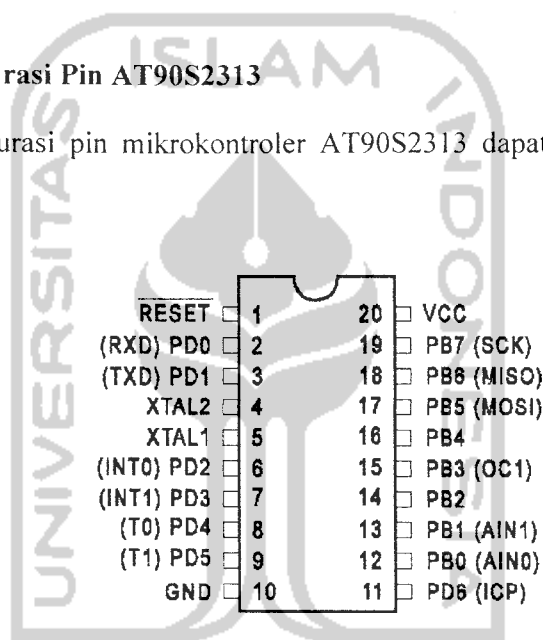
Gambar 2. 1. Arsitektur mikrokontroler AT90S2313

Dalam arsitektur tersebut terlihat bahwa tidak terdapat *accumulator*. Dengan demikian data *input* dan *output* dalam proses ALU disimpan dalam salah satu atau dua *register* dari 32 *register* yang disediakan. Terlihat juga bahwa *bus* data dan *bus* program terpisah secara fisik (tidak secara *logic*), *bus* data memiliki lebar 8 *bit*, sedangkan *bus* program memiliki lebar 16 *bit*.

Proses eksekusi sebuah perintah adalah kode dari *flash Memory* di-load ke register perintah melalui *bus* program dan langsung didekodekan menjadi sejumlah kontrol oleh *decoder* instruksi. Untuk mencapai kecepatan 1 MIPS per MHz AVR menerapkan *pipelining instruction*. *Pipelining* instruksi ini dilakukan dengan cara melakukan pengambilan instruksi selama prosesor mengeksekusi sebuah perintah.

2.1.2. Konfigurasi Pin AT90S2313

Konfigurasi pin mikrokontroler AT90S2313 dapat dilihat pada Gambar 2.2. berikut



Gambar 2. 2. Konfigurasi pin mikrokontroler AT90S2313

Dari gambar konfigurasi pin seperti diatas maka tiap pin dapat dijelaskan sebagai berikut.

- Vcc → Digunakan untuk masukan tegangan sebesar 5 volt.
- Reset → Digunakan untuk reset, mikrokontroler akan mereset program jika pin ini berlogika *low* selama 50ns.

- c. Gnd → Pin *ground*.
- d. Port B → Merupakan 1 kelompok *8-bit bi-directional I/O Port*. Pin ini pada kondisi *tri-state* ketika terjadi *reset*. Dapat diberi *pull-up* secara *internal*. Pada *port* ini juga tersedia fasilitas SCK, MOSI dan MISO yang dapat digunakan untuk keperluan *download*. Selain itu juga tersedia AIN0 dan AIN1 yang dapat digunakan sebagai *input Comparator*. Sedangkan OC1 adalah keluaran untuk *timer/counter1*.
- e. Port D → Merupakan 1 kelompok *7-bit bi-directional I/O Port*. Pin ini pada kondisi *tri-state* ketika terjadi *reset*. Dapat diberi *pull-up* secara *internal*. RXD dan TXD dalam *Port* ini disediakan untuk fasilitas komunikasi UART. INT0 dan INT1 disediakan untuk operasi *external interrupt*. Sedangkan T0 dapat digunakan untuk *trigger* apabila akan digunakan *timer0* dengan menggunakan *external clock*. Fasilitas lain dalam *port* ini adalah T1, yang berfungsi seperti halnya T0, tetapi digunakan pada *Timer1*.
- f. X-TAL 1 → Merupakan *input* dari *inverting* osilator.
- g. X-TAL2 → Merupakan *output* dari *inverting* osilator.

2.1.3. Memori AT90S2313

AT90S2313 memiliki beberapa memori yaitu 2 Kbytes memori *Flash*, memori ini juga sering disebut dengan memori program karena memori ini

digunakan untuk menyimpan program yang dibuat. 128 *bytes* memori EEPROM, 128 *bytes* memori SRAM, dan 32 x 8 bit *register*.

2.1.4. Perangkat Lunak Mikrokontroler AT90S2313

Mikrokontroler AT90S2313 memiliki 118 perangkat instruksi. Perangkat instruksi mikrokontroler AVR dapat dibagi sebagai berikut :

- a. Instruksi transfer data, instruksi ini berfungsi untuk tranfer data antara *register* ke *register*, memori ke memori, *register* ke memori, antarmuka ke *register* dan antar muka ke memori.
- b. Instruksi aritmatika dan *logic*, instruksi aritmatika meliputi penjumlahan, pengurangan, penambahan satu (*increament*), dan pengurangan satu (*decreament*). Instruksi logika dan manipulasi *bit*, yang melaksanakan operasi AND, OR, XOR, perbandingan, penggeseran dan komplemen data.
- c. Instruksi *Bit* dan *Bit-Test*, yaitu instruksi untuk *setting* kondisi tiap *bit*, baik *set* maupun *clear*, bahkan ada beberapa variasi, seperti instruksi putar, hingga *watchdog reset*.
- d. Instruksi percabangan, yang berfungsi mengubah urutan normal pelaksanaan suatu program menjadi sesuai yang dikehendaki. Dengan instruksi ini program yang sedang dilaksanakan akan mencabang ke suatu alamat tertentu. Instruksi percabangan dibedakan atas percabangan bersyarat dan percabangan tanpa syarat.
- e. Instruksi *stack*, *I/O* dan kontrol, yang digunakan untuk mengatur penggunaan *stack*, membaca/menulis *port I/O* serta pengontrolan-pengontrolan.

2.1.5. UART (*Universal Asynchronous Receiver Transmitter*)

AT90S2313 dilengkapi dengan *full duplex Universal Asynchronous Transmitter* (UART) dengan kemampuan :

1. *Baud Rate Generator* yang dapat membangkitkan berbagai macam kecepatan pengiriman (*baud rate-bps*).
2. *Baud Rate* yang tinggi pada kristal berfrekuensi rendah.
3. Mode pengiriman data 8 atau 9 bit.
4. *Noise Filtering*.
5. *Overrun Detection*.
6. *Framming Error Detection*.
7. *False Start Bit Detection*.
8. Tiga interupsi terpisah untuk *TX complete*, *TX Data register Empty*, dan *RX complete*.

2.1.6. Timer/Counter

AT90S2313 memiliki 2 buah *timer counter* yang masing-masing mempunyai seleksi *prescale* terpisah dari sebuah *prescale* yang sama. Yang pertama adalah *timer/counter0* merupakan *timer/counter* 8-bit, dapat dipergunakan sebagai *timer* dengan sumber *clock* diambil dari pin T0. Kedua adalah *timer/counter1* yang merupakan *timer counter* 16-bit. *timer/counter1* dapat digunakan sebagai *timer* dengan sumber *clock* dari *prescale* CK atau sebagai *counter* dengan sumber *clock* dari pin T1. *Timer/counter1* dapat juga digunakan sebagai *PWM (Pulse Width Modulator)*.

Jika *timer/counter1* diaktifkan sebagai PWM ada 3 register yang harus diinisialisasi yaitu *timer/counter1 control Register A* (TCCR1A), *timer/counter1 control register B* (TCCR1B) dan *timer/counter1 output compare register A* (OCR1AH dan OCR1AL)

Tabel 2. 1. Timer/Counter1 Control Register A (TCCR1A)

Bit	7	6	5	4	3	2	1	0	
IOF (IOF)	COM1A1	COM1A0	–	–	–	–	PWM11	PWM10	TCCR1A
Read/Write	R/W	R/W	R	R	R	R	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Pada mode PWM bit COM1A1 dan COM1A0 berfungsi sebagai berikut :

Tabel 2. 2. Fungsi Compare mode pada mode PWM

COM1A1	COM1A0	Keterangan
0	0	PWM tidak diaktifkan
0	1	PWM tidak diaktifkan
1	0	PWM aktif non-inverted
1	1	PWM aktif inverted

Sedangkan bit PWM11 dan PWM10 berfungsi untuk memilih mode PWM seperti Tabel 2.3 berikut.

Tabel 2. 3. Pemilihan mode PWM

PWM11	PWM10	Keterangan
0	0	PWM tidak aktif
0	1	Mode 8-bit PWM
1	0	Mode 9-bit PWM
1	1	Mode 10-bit PWM

2.2. Telepon Seluler *Siemens M35*

Telepon Seluler *Siemens M35* merupakan telepon genggam kelas menengah (*middle end*) yang cukup populer di masanya. Secara teknis, telepon seluler *Siemens M35* juga memiliki daya RF yang cukup tinggi dan sensitivitas penerima yang baik sehingga dapat diandalkan di daerah-daerah yang level amplitudo sinyalnya rendah.

Siemens M35 juga memiliki pin konektor eksternal yang dapat dimanfaatkan untuk pengendalian telepon seluler dari luar dengan piranti bantu mikrontroler ataupun PC (*Personal Computer*) seperti terlihat pada Gambar 2.3. Komunikasi data antara perangkat yang akan dirancang dengan telepon seluler *Siemens M35* menggunakan kabel penghubung tanpa rangkaian tambahan.



Gambar 2. 3. Susunan pin eksternal *siemens M35* tampak bawah
Susunan pin eksternal *Siemens M35* adalah sebagai berikut:

1. *Ground*
2. *Self-Service Recognition/control battery charger*
3. *Load Charging voltage & POWER (ignition)*
4. *No Connection*
5. *Data output*
6. *Data input*
7. *ZUB clock Recognition/control accessories*
8. *ZUB data Recognition/control accessories*

9. *Ground for microphone*
10. *Microphone input*
11. *Loudspeaker*
12. *Ground for external speaker*

2.3. AT Command

AT Command adalah kode-kode atau perintah-perintah yang ada di dalam tampilan menu pada sebuah telepon seluler (*handphone*). AT Command digunakan untuk berkomunikasi dengan piranti diluar *handphone* secara serial seperti misalnya mikrokontroler atau PC. AT Command bertugas mengirim atau menerima data ke atau dari SMS-Centre. AT Command tiap-tiap SMS device bisa berbeda-beda, tapi pada dasarnya adalah sama. AT Command untuk komunikasi dengan SMS-Centre ada beberapa macam. Berikut beberapa AT Command yang penting untuk SMS adalah:

- AT+CMGS → untuk mengirim SMS
- AT+CMGL → untuk memeriksa SMS
- AT+CMGD → untuk menghapus SMS

AT Command untuk SMS, biasanya diikuti oleh data I/O yang diwakili oleh unit-unit data yang sering disebut PDU.

2.4. PDU (*Protocol Data Unit*)

PDU adalah bahasa yang digunakan oleh SMS. Data yang mengalir ke atau dari SMS-Centre harus berbentuk *Protocol Data Unit*. PDU berisi bilangan-bilangan heksadesimal yang mencerminkan bahasa I/O. PDU terdiri atas beberapa

header, *header* untuk mengirim SMS ke SMS-Centre berbeda dengan *header* untuk membaca SMS dari SMS-Centre.

2.4.1. PDU untuk mengirim SMS ke SMS-Centre

Protocol Data Unit (PDU) untuk mengirim SMS terdiri atas 8 *header*, yang dapat dijelaskan sebagai berikut

1. Nomer SMS-Centre

Header pertama ini dibagi atas 3 *subheader* yaitu:

- a. Jumlah pasangan heksadesimal SMS-Centre dalam bilangan heksadesimal.
- b. Kode national/ international nomor SMS-Centre
 - Untuk nasional, kode *subheader*nya yaitu 81
 - Untuk internasional, kode *subheader*nya yaitu 91
- c. No SMS-Centre. Format penulisan nomor SMS-Centre ditulis dalam pasangan heksadesimal yang dibalik-balik. jika tertinggal satu angka heksadesimal tidak memiliki pasangan, angka tersebut akan dipasangkan dengan huruf F didepannya. Misalnya
Sebagai contoh jika nomor SMS-Centrenya adalah 0816124 maka format PDUnya adalah sebagai berikut:

- 05 (banyaknya pasangan nomer setelah ditambah 81)
- 81 (kode nasional, banyaknya 1 pasang)
- 80-61-21-F4 (nomer tujuan SMS-Centre, banyaknya 4 pasang)

Setelah digabung maka format PDUnya menjadi 0581806121F4

Berikut ini adalah tabel beberapa nomer kode PDU dan SMS-Centre operator seluler yang ada di Indonesia.

Tabel 2. 4. Kode PDU dan no SMS-Centre Nasional

No	Operator Seluler	SMS-Centre No	Kode PDU
1	Telkomsel	081 1000000	06818011000000
2	Satelindo	0816124	0581806121F4
3	Excelcom	0818445009	06818081440590
4	Indosat-M3	0855000000	06818055000000

Tabel 2.5. Kode PDU dan no SMS-Centre Internasional

NO	Operator Seluler	SMS-Centre No	Kode PDU
1	Telkomsel	62811000000	07912618010000F0
2	Satelindo	62816124	059126181642
3	Excelcom	62818445009	07912618485400F9
4	Indosat-M3	62855000000	07912658050000F0

2. Tipe SMS

Tipe SMS diisi 11 yang merupakan oktet pertama untuk PDU SMS untuk dikirim (SMS SUBMIT).

3. Nomor Referensi SMS

Nomer referensi ini dibiarkan 0, jadi bilangan heksadesimalnya 00. kemudian akan diberikan sebuah nomer refensi secara otomatis oleh *handphone* atau alat SMS-Gateway

4. Nomer Handphone Penerima

Sama seperti Cara menulis PDU *header* untuk SMS-Centre, *header* ini juga terbagi atas tiga bagian, sebagai berikut:

- a. Jumlah bilangan desimal nomer handphone yang dituju dalam bilangan heksa.
- b. *Natinal/International Code* nomer handphone
 - untuk nasional, kode *subheader*-nya: 81
 - untuk internasional, kode *subheader*-nya: 91
- c. Nomer handphone yang dituju, ditulis dalam pasangan heksa dibalik-balik.

Jika tertinggal satu angka heksa yang tidak memiliki pasangan, angka tersebut dipasangkan dengan huruf F didepannya.

Contohnya untuk nomer handphone yang dituju = 628122954437 dapat ditulis dengan 2 cara sebagai berikut:

- a) 0C (jumlah digit nomor tujuan 12 angka)
- b) 91 (kode nasional, banyaknya 1 pasang)
- c) 26-18-22-59-44-73 (nomer tujuan, banyaknya 6 pasang)

Bila digabung format PDUnya menjadi 0C91261822594473.

5. Bentuk SMS

Bentuk pengiriman SMS ada beberapa macam jenis antara lain:

0 → 00 → dikirim sebagai SMS

1 → 01 → dikirim sebagai telex

2 → 02 → dikirim sebagai fax

6. Skema Encoding Data I/O

Ada dua skema, yaitu:

- a. Skema 7 bit → ditandai dengan angka 0 → 00
- b. Skema 8 bit → ditandai dengan angka lebih besar dari 0 → diubah ke heksadesimal.

Kebanyakan *handphone/SMS Gate way* yang ada sekarang menggunakan skema 7 bit sehingga menggunakan kode 00.

7. Jangka Waktu Sebelum SMS Expired

Jika bagian ini di-skip, berarti tidak membatasi waktu berlakunya SMS. Sedangkan jika diisi dengan suatu bilangan integer yang kemudian diubah ke pasangan heksa tertentu, bilangan yang diberikan tersebut akan mewakili jumlah waktu validitas SMS tersebut. Rumus untuk menghitung jangka waktu validitas SMS adalah sebagai berikut:

Tabel 2. 6. Rumus hitung validitas SMS

Integer (INT)	Jangka Waktu Validitas SMS
0 – 143	$(INT + 1) \times 5$ menit (berarti: 5 menit s/d 12 jam)
144 – 167	12 jam + $((INT - 143) \times 30)$ menit
168 – 196	$(INT - 166) \times 1$ hari
197 – 255	$(INT - 192) \times 1$ minggu

Agar SMS pasti terkirim sampai ke *handphone* penerima sebaiknya tidak memberikan batasan waktu validitasnya.

8. Isi SMS

Header ini terdiri atas 2 *subheader*, yaitu :

- Panjang isi (jumlah huruf dari isi)
- Isi berupa pasangan bilangan heksa

Untuk *handphone/SMS Gateway* berskema encoding 7 bit, jika mengetikkan suatu huruf dari *keypadnya*, berarti telah dibuat 7 angka berurutan dalam skema 7 bit tersebut diperlihatkan pada Tabel 2.7 berikut.

Tabel 2. 7. Skema 7 bit alphabet

				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1
b4	b3	b2	b1		0	1	2	3	4	5	6	7
0	0	0	0	0	@	Δ	SP	0	-	P	"	p
0	0	0	1	1	£	_	!	1	A	Q	a	q
0	0	1	0	2	\$	Φ	"	2	B	R	b	r
0	0	1	1	3	¥	Γ	#	3	C	S	c	s
0	1	0	0	4	É	Λ	■	4	D	T	d	t
0	1	0	1	5	Ê	Ω	%	5	E	U	e	u
0	1	1	0	6	Û	Π	&	6	F	V	f	v
0	1	1	1	7	Ï	Ψ	'	7	G	W	g	w
1	0	0	0	8	Ö	Σ	(	8	H	X	h	x
1	0	0	1	9	Ç	Θ	)	9	I	Y	i	y
1	0	1	0	10	LF	Ξ	*	:	J	Z	j	z
1	0	1	1	11	Ø	Ι	+	;	K	Å	k	å
1	1	0	0	12	á	Æ	,	<	L	Ö	l	ö
1	1	0	1	13	CR	æ	-	=	M	Ñ	m	ñ
1	1	1	0	14	Ä	β	.	>	N	Ü	n	ü
1	1	1	1	15	Å	É	/	2	O	§	o	å

2.4.2. PDU untuk menerima SMS dari SMS-Centre

Kebanyakan *header* hampir sama dengan *header* untuk mengirim SMS, kecuali beberapa yang berbeda. *header* untuk menerima SMS dijelaskan sebagai berikut:

1. No SMS-Centre
2. Tipe SMS → untuk SMS-terima = 4 → 04
3. No handphone pengirim
4. Bentuk SMS
5. Skema encoding

6. Tanggal dan waktu SMS di-stamp di SMS-Centre

Diwakili oleh 12 bilangan heksa (6 pasangan) yang berarti yy/mm/dd
hh:mm:ss

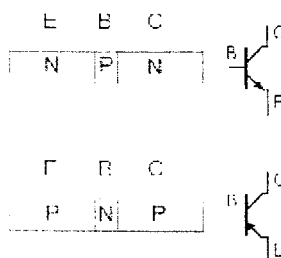
Contoh : 023022512380 → 20/03/22 15:32:08 → 20 maret 2002
15:32:08 WIB

7. Batas waktu validitas → jika tidak dibatasi dilambangkan dengan 00

8. Isi SMS

2.5. Transistor Bipolar

Transistor merupakan dioda dengan dua sambungan (*junction*). Sambungan itu membentuk transistor PNP maupun NPN. Ujung-ujung terminalnya berturut-turut disebut emitor, basis dan kolektor. Transistor jenis ini disebut transistor bipolar, karena struktur dan prinsip kerjanya tergantung dari perpindahan elektron di kutup negatif mengisi kekurangan elektron (*hole*) di kutup positif. Kata bilpolar terdiri dari 2 susu kata yaitu bi yang berarti dua dan polar berarti kutub. Transistor bipolar pertama kali ditemukan pada tahun 1951 oleh William Schockley.



Gambar 2. 4. Transistor NPN dan PNP

2.5.1. Kurva Basis

Hubungan antara I_B dan V_{BE} tentu saja akan berupa kurva dioda. Karena memang telah diketahui bahwa *junction basis-emitor* tidak lain adalah sebuah dioda. Jika hukum Ohm diterapkan pada *loop basis* adalah :

$$I_B = (V_{BB} - V_{BE}) / R_B \dots\dots\dots (2.1)$$

V_{BE} adalah tegangan jepit dioda *junction basis-emitor*. Arus hanya akan mengalir jika tegangan antara *basis-emitor* lebih besar dari V_{BE} . Sehingga arus I_B mulai aktif mengalir pada saat nilai V_{BE} tertentu.



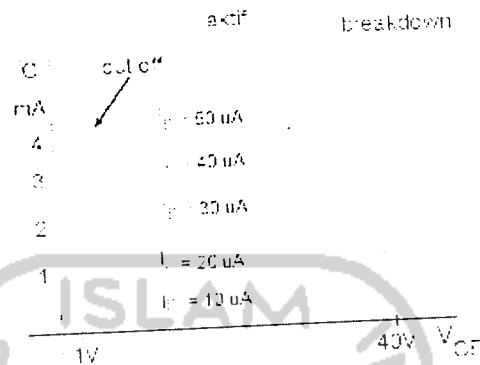
Gambar 2. 5. Kurva $I_B - V_{BE}$

Besar V_{BE} umumnya diketahui sebesar 0.7 volt untuk transistor silikon dan $V_{BE} = 0.3$ volt untuk transistor germanium. Nilai ideal V_{BE} sebesar 0 volt.

2.5.2. Kurva Kolektor

Kurva kolektor digunakan untuk mengetahui bagaimana hubungan antara arus basis I_B , arus kolektor I_C dan tegangan kolektor-emiter V_{CE} . Pada Gambar

2.6. berikut menunjukkan kurva kolektor arus I_C terhadap V_{CE} dimana arus I_B dibuat konstan.



Gambar 2. 6. Kurva kolektor

Dari kurva diatas terlihat ada beberapa bagian yang menunjukkan daerah kerja transistor. Pertama adalah daerah *saturasi*, lalu daerah *cut-off*, kemudian daerah *aktif* dan seterusnya daerah *breakdown*.

2.5.2.1. Daerah Aktif

Daerah kerja transistor yang normal adalah pada daerah aktif, dimana arus I_C konstans terhadap berapapun nilai V_{CE} . Dari kurva ini diperlihatkan bahwa arus I_C hanya tergantung dari besar arus I_B . Daerah kerja ini biasa juga disebut daerah linear

2.5.2.2. Daerah Saturasi

Daerah saturasi adalah daerah mulai dari $V_{CE} = 0$ volt sampai kira-kira 0.7 volt (transistor silikon), Daerah saturasi terjadi akibat dari efek dioda kolektor-basis yang mana tegangan V_{CE} belum mencukupi untuk dapat menyebabkan aliran elektron.

2.5.2.3. Daerah Cut-Off

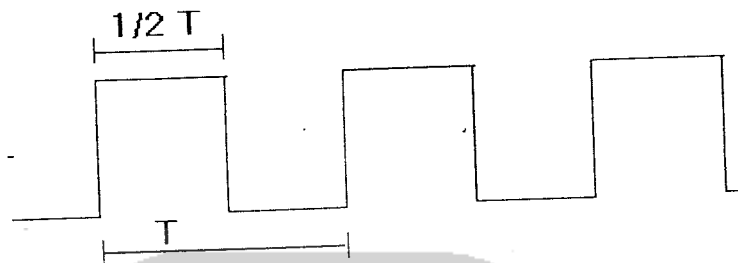
Jika kemudian tegangan V_{CC} dinaikkan perlahan-lahan, sampai tegangan V_{CE} tertentu tiba-tiba arus I_C mulai konstan. Pada saat perubahan ini, daerah kerja transistor berada pada daerah cut-off yaitu dari keadaan saturasi (OFF) lalu menjadi aktif (ON). Perubahan ini dipakai pada system digital yang hanya mengenal angka biner 1 dan 0 yang tidak lain dapat direpresentasikan oleh status transistor OFF dan ON.

2.6. PWM (*Pulse Width Modulation*)

Penggunaan motor DC dewasa ini sangatlah umum, salah satu kelebihan motor DC adalah relatif gampang didapat dan mudah diatur kecepatannya. Secara umum pengaturan kecepatan motor DC adalah dengan menggunakan cara analog. Pada perancangan ini akan dibahas cara mengatur kecepatan motor DC dengan menggunakan mikrokontroler.

Cara pengaturan kecepatan yang digunakan adalah dengan menggunakan teknik PWM (*Pulse Width Modulation*), salah satu teknik untuk mengatur kecepatan motor DC yang umum digunakan. Teknik PWM untuk pengaturan kecepatan motor adalah teknik pengaturan kecepatan motor dengan cara merubah-ubah besarnya *duty cycle* pulsa. Besarnya amplitudo dan frekuensi pulsa adalah tetap, sedangkan besarnya *duty cycle* berubah-ubah sesuai dengan kecepatan yang diinginkan, semakin besar *duty cycle* maka semakin cepat pula kecepatan motor, dan sebaliknya semakin kecil *duty cycle* maka semakin pelan pula kecepatan

motor. Sebagai contoh bentuk pulsa kotak dengan *duty cycle* pulsa 50%. seperti pada Gambar 2.7.



Gambar 2. 7. Pulsa PWM dengan *duty cycle* 50 %

Seperti pada Gambar 2.7, semakin besar *duty cycle* pulsa kotak, maka semakin lama pula posisi logika *high*. Jika motor diatur agar berjalan (nyala) ketika diberi logika *high*, maka jika memberi pulsa seperti pada Gambar 2.7 diatas, maka motor akan berada pada kondisi “nyala-mati-nyala-mati” sesuai dengan bentuk pulsa tersebut. Semakin lama motor berada pada kondisi “nyala” maka semakin cepat pula kecepatan motor tersebut. Motor akan berputar dengan kecepatan maksimum jika mendapat pulsa dengan *duty cycle* 100%. Dengan kata lain motor mendapat logika *high* terus menerus.

Dengan mengatur besarnya *duty cycle* pulsa kotak yang dikirimkan, dapat diatur banyaknya logika *high* yang diberikan pada motor, dengan kata lain mengatur lamanya waktu motor untuk berputar dalam satu periode pulsa. Jika lamanya waktu motor untuk berputar dalam satu periode pulsa ini berubah maka kecepatan putaran motor juga akan berubah, sesuai dengan *duty cycle* atau waktu motor untuk berputar dalam satu periode pulsa.

BAB III

PERANCANGAN SISTEM

3.1. Diagram Blok Sistem

Secara umum diagram blok pengendali kecepatan motor DC melalui telepon seluler berbasis mikrokontroler AT90S2313 seperti pada Gambar 3.1 berikut:



Gambar 3. 1. Diagram blok sistem

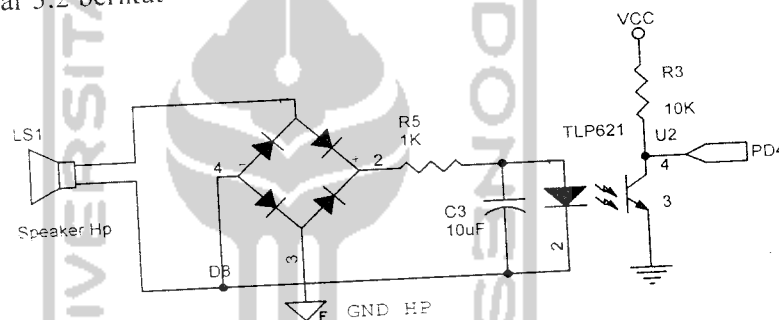
Dari diagram blok diatas dapat dilihat bahwa prinsip kerja dari sistim pengendali kecepatan motor melalui telepon seluler adalah sebagai berikut : komunikasi antar dua buah telepon seluler dilakukan dengan mengirimkan pesan SMS (*Sort Massage Service*) dari telepon pengontrol ke telepon penerima, setelah pesan SMS diterima oleh telepon penerima, maka mikrokontroler akan mengecek apakah pesannya sesuai apa tidak. Pada perancangan ini pesan SMS harus dikirim dengan format <Arah putaran> spasi <Kecepatan putaran>. Arah putarannya ada 2 yaitu Kanan dan Kiri. Sedangkan kecepatan putaran dari 100 – 900 dengan kelipatan 100 rpm. Setelah mikrokontroler membaca SMS pada *inbox* telepon penerima jika format SMS yang diterima sesuai dengan format yang bener maka mikrokontroler akan mengaktifkan rangkaian driver motor DC sesuai dengan arah

dan kecepatan yang diinginkan. Jika pesan yang diterima formatnya salah maka telepon penerima akan mengirimkan pesan “Maaf Format SMS anda salah silahkan ulangi dengan format yang benar” ke telepon pengontrol.

3.2. Perancangan Perangkat Keras

3.2.1. Rangkaian Pendeteksi Dering SMS

Rangkaian pendeteksi dering SMS digunakan untuk mendeteksi adanya SMS yang diterima. Rangkaian pendeteksi dering SMS seperti yang ditunjukkan pada Gambar 3.2 berikut



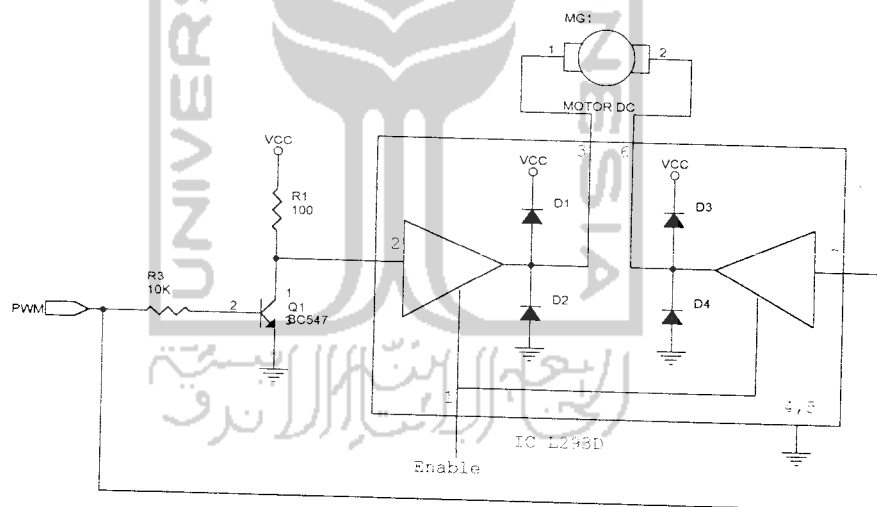
Gambar 3. 2. Rangkaian pendeteksi dering SMS

Dari Gambar 3.2 diatas dapat dijelaskan bahwa jika ada SMS yang diterima maka speaker Hp akan berbunyi. Suara dering pada speaker hp disearahkan oleh diode bridge sehingga keluarannya menjadi tegangan DC yang menyebabkan LED inframerah optocoupler menyala. Jika LED inframerah menyala maka akan menyinari photo transistor yang menyebabkan photo transistor aktif. Saat photo transistor aktif, photo transistor berfungsi seperti saklar tertutup sehingga keluaran rangkaian yang menuju port PD4 mikrokontroler akan berlogika rendah atau '0'. Sedangkan jika tidak ada dering maka LED inframerah

optocoupler tidak menyala dan photo transistor juga tidak aktif. Saat photo transistor tidak aktif photo transistor berfungsi seperti saklar terbuka sehingga keluaran rangkaian akan berlogika tinggi atau '1'. Sehingga dapat dikatakan bahwa logika rendah atau '0' pada port PD4 berarti ada SMS yang diterima sedangkan jika port PD4 berlogika tinggi atau '1' berarti tidak ada SMS yang diterima.

3.2.2. Rangkaian H- Bridge Pengerak Motor DC

Rangkaian H-bridge penggerak motor DC seperti yang ditunjukkan pada Gambar 3.3 berikut.



Gambar 3. 3. Rangkaian penggerak motor DC

Rangkaian penggerak motor DC digunakan untuk menggerakkan motor ke kiri atau ke kanan dengan kecepatan sesuai dengan keluaran PWM mikrokontroler AT90S2313. Rangkaian penggerak motor DC dibangun dengan sebuah IC L293D berisi empat buah *push-pull*. Setiap dua buah *push-pull* dapat

digunakan sebagai sebuah rangkaian H-bridge dan dapat diaktifkan dengan sebuah sinyal *enable*. IC L293D mampu beroperasi pada tegangan 4,5 V sampai 36 V. Besarnya arus yang dapat ditarik adalah 600mA pada kondisi normal serta 1,2 A pada arus puncak (sesaat).

Rangkaian penggerak motor DC pada Gambar 3.3 dapat dijelaskan sebagai berikut. Untuk mengaktifkan sebuah rangkaian H-bridge pin enable diberi logika '1'. Pin 2 dan 7 IC L293D merupakan input sinyal. Untuk membuat motor berputar kedua input ini harus berbeda potensial. Rangkaian transistor Q1 merupakan rangkaian penguat arus dan *inverted* tegangan, dikatakan demikian karena keluaran dari rangkaian tersebut memiliki arus yang lebih besar dari arus masukan dan memiliki tegangan yang berlawanan dengan masukannya. Misalnya tegangan masukannya 5 V maka keluarannya sebesar 0 V dan jika masukannya 0 V keluarannya sebesar 5 V. Sebagai penguat arus, arus kolektor dibuat maksimum yaitu sebesar.

$$I_{C(max)} = \frac{V_{CC}}{R_C}$$

Jika resistor kolektor (R_C) sebesar 100 Ohm maka arus I_C adalah:

$$I_{C(max)} = \frac{5}{100} = 0,05 A$$

Karena pada transistor aktif berlaku persamaan arus $I_C = \beta \cdot I_B$ dan jika β transistor sebesar 125 (hasil pengukuran). Maka untuk membuat arus kolektor agar maksimum, arus basis minimum yang harus dipenuhi adalah sebesar :

$$I_{B(min)} = \frac{I_C}{\beta} = \frac{0,05}{125} = 0,4 \cdot 10^{-3} = 0,4 \text{ mA}$$

Untuk membuat I_B sebesar 4 mA maka besarnya resistor basis (R_4) dapat dihitung sebagai berikut :

$$I_B = \frac{V_{BB} - V_f}{R_B} \quad \text{Dimana :}$$

V_{BB} : Tegangan Basis

V_f : Tegangan dalam transistor 0,3 untuk germanium dan 0,7 untuk silikon

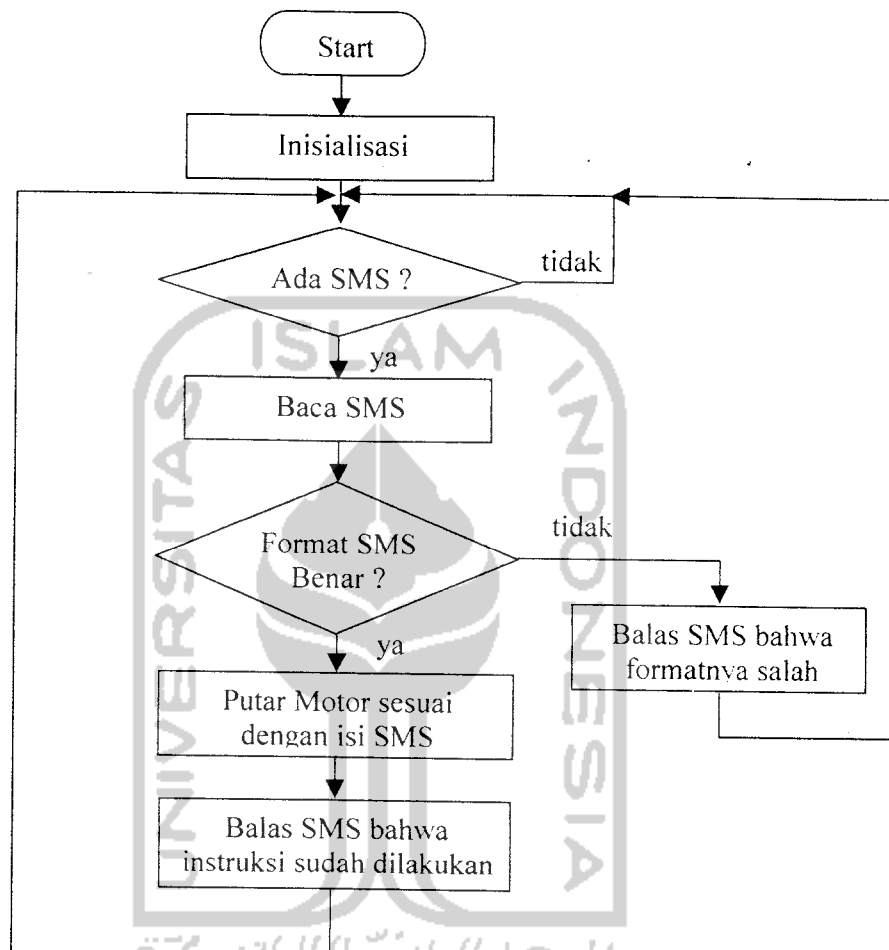
R_B : Resistor basis pada rangkaian diatas sama dengan R_4

$$R_B = \frac{V_{BB} - V_f}{I_B} = \frac{5 - 0,7}{0,4 \cdot 10^{-3}} = 10750 = 10,750 K\Omega \approx 10 K\Omega$$



3.3. Perancangan Perangkat Lunak

3.3.1. Diagram Alir Program



Gambar 3. 4. Diagram alir program

Perancangan program pengendali kecepatan motor melalui telepon seluler dapat dijelaskan sebagai berikut. Setelah sistem dijalankan maka program akan menunggu adanya sms yang diterima. Jika ada sms yang diterima maka format sms yang diterima disimpan di RAM mikrokontroler. Kemudian program akan mencocokkan apakah format smsnya sudah benar apa tidak jika sudah benar maka

program akan mengaktifkan PWM sesuai dengan kecepatan yang diinginkan kemudian program akan mengirim balasan sms yang memberitahukan bahwa instruksi sudah dijalankan. Namun jika format smsnya salah maka program akan mengirim balasan sms yang memberitahukan bahwa format smsnya salah.

3.3.2. Inisialisasi UART

Inisialisasi UART terdiri dari inisialisasi 2 buah register yaitu register UCR (*UART Control Register*) dan UBRR (*UART Bit Rate Register*). UCR terdiri dari kontrol bit untuk *mask interrupt* (bit 5-7) dan *mode* dari UART (bit 2-4), sedang bit 0 dan 1 berfungsi untuk penyimpanan data pada mode pengiriman 9 bit saat CHR9 berlogika 1.

Tabel 3. 1. Register UCR (*UART Control Register*)

Bit	7	6	5	4	3	2	1	0	
SCA/SDA	FNCE	INCE	UDRE	RNEN	TXEN	CHR9	RNBS	TXBS	UCR

Karena piranti masukan hanya membutuhkan UART sebagai pengirim data 8 bit dan tidak ada interupsi dari UART yang dibutuhkan, maka semua bit dalam UCR berlogika 0 kecuali bit TXEN yang mengaktifkan fungsi *transmitter*. *UART Bit Rate Register (UBRR)* berisi data pembagi frekuensi *clock* untuk menghasilkan bit rate yang dibutuhkan. Persamaan hubungan antara frekuensi kristal, nilai UBRR dan *bit rate* adalah sebagai berikut :

$$\text{BAUD} = \frac{f_{\text{ck}}}{16 \cdot (\text{UBRR} + 1)} \quad f_{\text{ck}} = \text{frekuensi kristal}$$

Dengan frekuensi kristal 4 MHz dan *Baud rate* 19200bps (*SIEMENS Command Set*), maka nilai UBRR dapat dihitung sebagai berikut :

$$UBRR = \frac{f_{ck}}{16.BAUD} - 1$$

$$UBRR = \frac{4.10^6}{16.19200} - 1 = 12$$

3.3.3. Inisialisasi PWM

Program PWM digunakan untuk mengatur kecepatan putaran motor. Pwm diinisialisasi dengan mengatur *timer/counter1 control* register A (TCCR1A), *timer/counter1 control* register B (TCCR1B). Kemudian *Timer/counter 1 Output Compare* Register A (OCR1AH dan OCR1AL) diisi data yang akan dibandingkan dengan *Timer/ counter1* secara terus menerus. Besarnya duty cycle PWM diisi pada register OCR1A.

```
ldi    temp,0b10000010    ; atiktifkan PWM pada mode 9-bit dan non-inverted
out    TCCR1A,temp
ldi    temp,0b00000100    ; Set prescalar to 256
out    TCCR1B,temp
out    OCR1AH,titip
out    OCR1AL,titip1
```

3.3.4. Mendeteksi dan membaca SMS

Program mendeteksi sms digunakan untuk mengetahui apakah ada sms yang diterima apa tidak yang kemudian sms tersebut dibaca oleh program pembaca sms sebagai berikut.

```

.equ    pesanSMS    =0x60

        sbi    PortD,4
tunggu: sbic    PIND,4          ; jika ada sms PIND,4 = 0
        rjmp   tunggu
        rcall  Ldelay
        ldi    temp,73         ; jumlah maksimal data sms yang diterima
        ldi    Yl,pesanSMS     ; set pada alamat RAM 0x60
        ldi    zl,low(msg*2)   ; kirim instruksi "AT+CMGL=0",0x0d
        ldi    zh,high(msg*2)  ; yang merupakan perintah AT Command
        rcall  kirim           ; untuk membaca SMS di inbox
jj:      rcall  receive         ; baca data dari Hp
        cpi    Rxbyte,':'       ; tunggu sampai datanya sama dengan ':'
        brne   jj
ii:      rcall  receive         ; baca data dari Hp
        cpi    Rxbyte,0x0A      ; tunggu sampai datanya samadengan 0x0A
        brne   ii
ulangan:
        rcall  receive         ; baca data dari Hp
        st     Y+,Rxbyte       ; simpan di RAM mulai alamat 0x60
        dec    temp
        tst    temp
        brne   ulangan         ; ulangi sebanyak 73 kali

```

3.3.5. Mengatur kecepatan motor DC

Seperti yang dibahas pada bab II bahwa pengaturan kecepatan yang digunakan menggunakan teknik PWM. Pada mikrokontroler AT90S2313 pengaturan PWM dilakukan dengan mengatur register-register TCCR1A, TCCR1B dan ORC1A.

Register TCCR1A digunakan untuk memilih mode PWM yang digunakan apakah menggunakan mode 8 bit, 9 bit atau 10 bit. Register TCCR1B digunakan untuk menentukan nilai prescaler yang digunakan. Register OCR1A digunakan untuk menyimpan nilai duty cyclenya.

Jika digunakan PWM 9 bit, frekuensi kristal sebesar 4 MHz dan prescaler 256 maka frekuensi PWMnya didapat sebagai berikut.

$$f_{PWM} = \frac{f_{TC}}{1022}$$

$$f_{TC} = \frac{f_{osc}}{Prescaler} = \frac{4 \cdot 10^6}{256} = 15625 Hz$$

$$f_{PWM} = \frac{15625}{256} = 15,28 Hz$$

3.3.6. Konversi format teks ke format PDU

Ada dua langkah yang harus dilakukan untuk mengkonversikan isi SMS, yaitu:

1. Mengubah isi SMS ke bentuk kode 7 bit (lihat Gambar 2. 1. Skema 7 bit alphabet).
2. Mengubah kode 7 bit menjadi 8 bit, yang diwakili oleh pasangan heksadesimal.

Contoh: misalnya pesan SMS adalah "hello"

Langkah pertama:

Tiap karakter pesan SMS dikonversi menjadi kode 7 bit seperti pada Tabel 3.2 sebelumnya.

Tabel 3. 2. Data konversi ke kode 7 bit

Karakter SMS	Kode 7 bit						
	b7	b6	b5	b4	b3	b2	b1
h	1	1	0	1	0	0	0
e	1	1	0	0	1	0	1
l	1	1	0	1	1	0	0
l	1	1	0	1	1	0	0
o	1	1	0	1	1	1	1

Langkah kedua:

Oleh karena total 7 bit x 5 huruf = 35 bit, sedangkan yang diperlukan adalah 8 bit x 5 huruf = 40 bit, maka diperlukan 5 bit dummy yang diisi dengan bilangan 0. Setiap 8 bit mewakili suatu pasangan heksa. Tiap 4 bit mewakili suatu angka heksa. 5 bit dummy diletakkan didepan bit ke tujuh karakter terakhir. Sisa bit setelah ditambahkan dengan bit dummy ditaruh di depan bit karakter selanjutnya seperti Tabel 3.3 berikut :

Tabel 3. 3. Data konversi 7 bit ke 8 bit

Karakter SMS	Kode 8 bit								Sisa bit
	b8	b7	b6	b5	b4	b3	b2	b1	
o	0	0	0	0	0	1	1	0	1111
l	1	1	1	1	1	1	0	1	100
l	1	0	0	1	1	0	1	1	00
e	0	0	1	1	0	0	1	0	1
h	0	1	1	0	1	0	0	0	

→ 06

→ FD ↑

→ 9B

→ 32

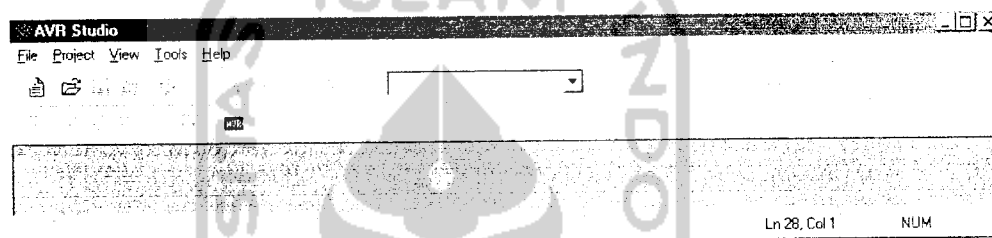
→ E8

Dengan demikian kata "hello" hasil konversinya menjadi: **E8329BFD06**.

3.3.7. Memprogram AT90S2313

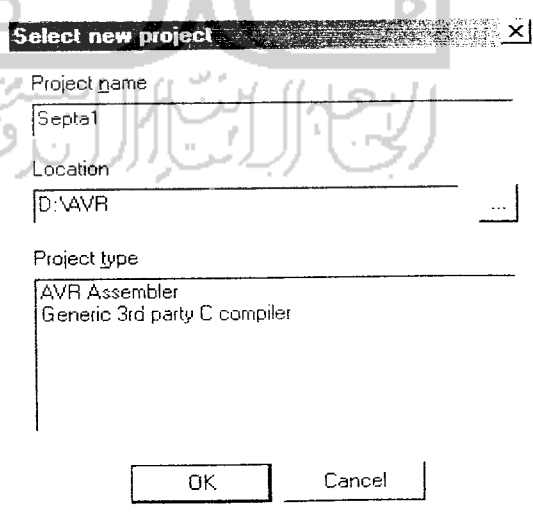
Untuk memprogram chip mikrokontroler AT90S2313 diperlukan beberapa software. Berikut ini prosedur memprogram mikrokontroler AVR AT90S2313.

1. Program yang akan dibuat dapat ditulis disembarang editor seperti misalnya notpad atau dapat juga menggunakan software AVR studio. AVR studio adalah sebuah perangkat lunak yang dirancang untuk melakukan compiler dan simulasi. Tampilan AVR studio seperti Gambar 3.5. berikut



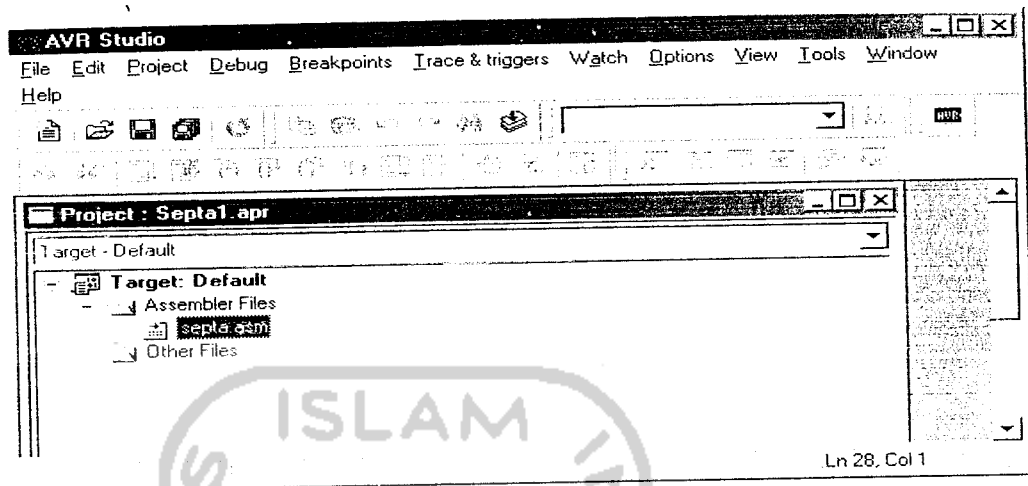
Gambar 3. 5. Tampilan menu AVR studio

2. Untuk mulai membuat projek lakukan langkah-langkah seperti berikut : klik Project → New kemudian muncul tampilan seperti berikut.



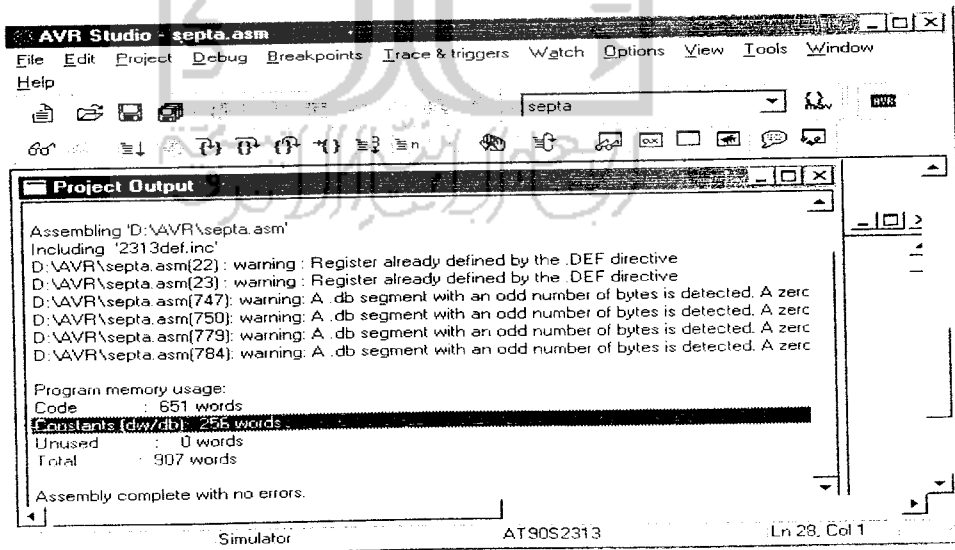
Gambar 3. 6. Tampilan select new project

3. Setelah di OK maka akan muncul tampilan berikut



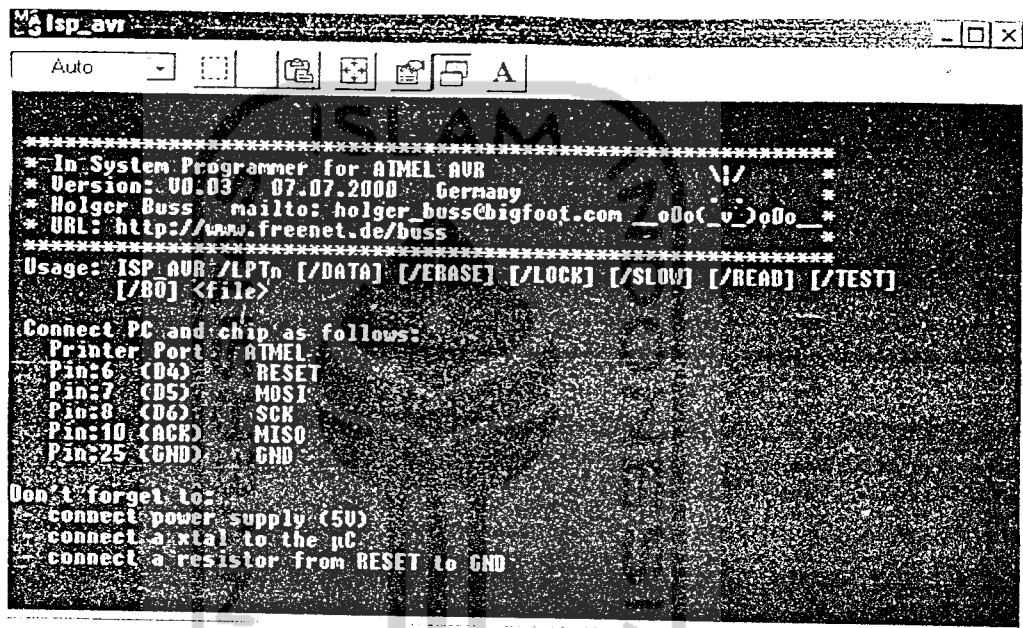
Gambar 3. 7. Tampilan New Project

4. Untuk membuat nama file program klik kanan pada assembler files, kemudian klik *create new file*, tuliskan nama file dengan ekstensi .asm.
5. Setelah program selesai dibuat, untuk melakukan compiler tekan Ctrl+F7 jika program sudah benar maka akan muncul tampilan sebagai berikut



Gambar 3. 8. Tampilan output compiler

6. Untuk melakukan pemrograman ke chip mikrokontroler ada banyak software yang dapat digunakan. Dalam hal ini digunakan software Isp_AVR. Dengan software Isp_AVR cukup hanya menggunakan 4 buah kabel saja dengan hubungan seperti berikut



```

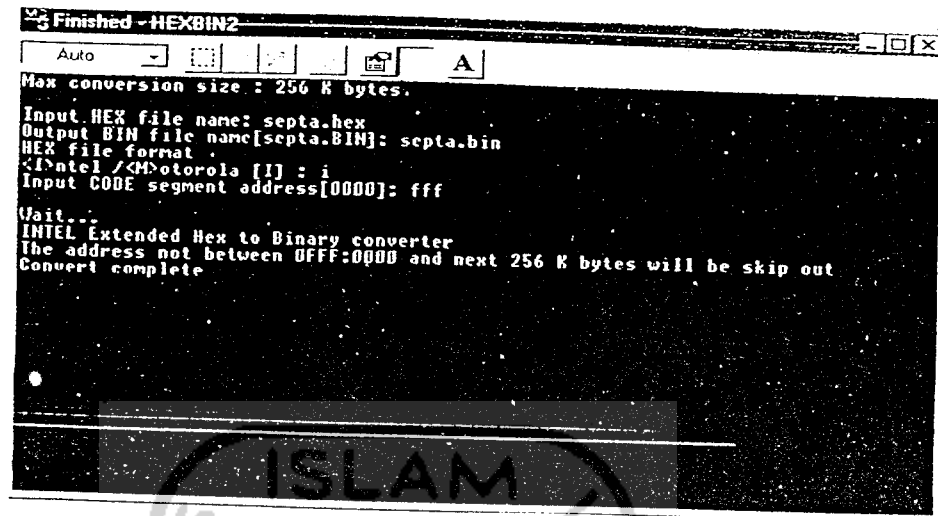
Isp_avr
Auto
*****
* In-System Programmer for ATMEL AVR
* Version: V0.03 07.07.2000 Germany
* Holger Buss mailto: holger_buss@bigfoot.com _oOo(_v_)oOo_
* URL: http://www.freenet.de/buss
*****
Usage: ISP_AVR /LPTn [/DATA] [/ERASE] [/LOCK] [/SLOW] [/READ] [/TEST]
        [/BO] <file>

Connect PC and chip as follows:
Printer Port: ATMEL
Pin:6 (D4) RESET
Pin:7 (D5) MOSI
Pin:8 (D6) SCK
Pin:10 (ACK) MISO
Pin:25 (GND) GND

Don't forget to:
- connect power supply (5V)
- connect a xtal to the µC
- connect a resistor from RESET to GND
  
```

Gambar 3. 9. Koneksi antara PC Printer Port dengan Port Mikrokontroler

7. Sebelum mendownload ke chip mikrokontroler file .hex hasil dari compiler ditransfer agar menjadi file.bin dengan software HexBin2.exe sebagai berikut.



Gambar 3. 10. Tampilan HexBin2.exe

8. Untuk mendownload program dibuat file berektensi .bat sebagai berikut

```
hexbin2 septa.hex septa.bin i 0
isp_AVR.exe /LPT1 /erase septa.bin
```
9. Setelah file.bat dibuat maka proses download program ke chip mikrokontroler dapat dilakukan. Jika proses download berhasil maka tampilannya sebagai berikut



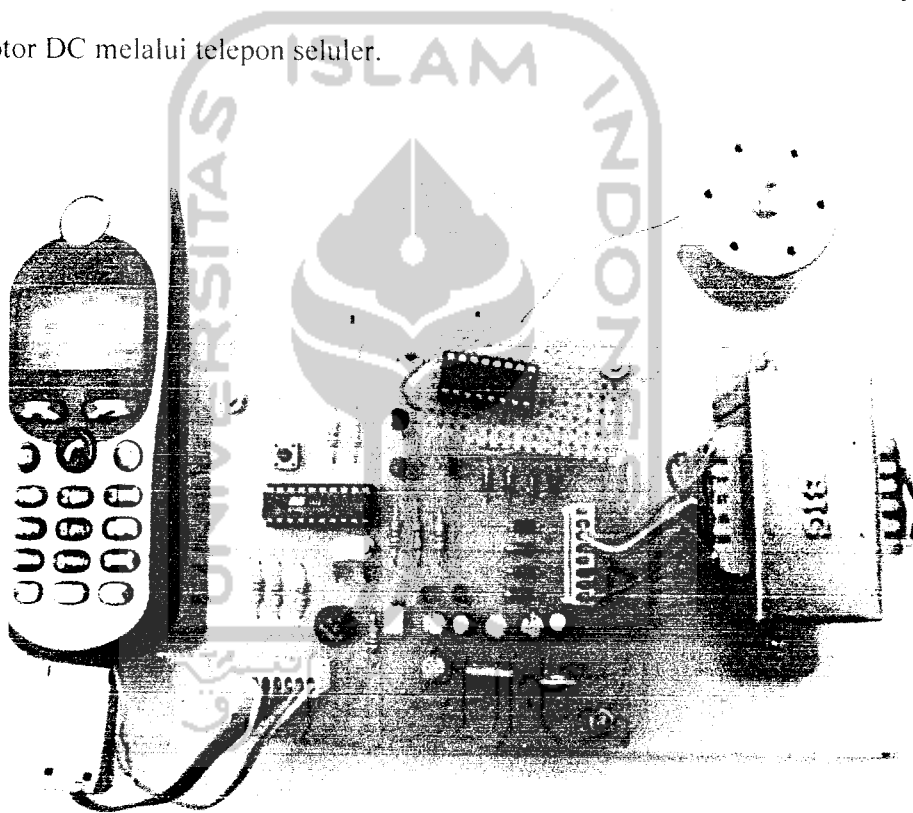
Gambar 3. 11. Tampilan chip sucssfully programed

BAB IV

PENGUJIAN DAN ANALISA

4.1. Pengujian Alat

Gambar 4.1 berikut menunjukkan tampilan alat pengendali kecepatan motor DC melalui telepon seluler.



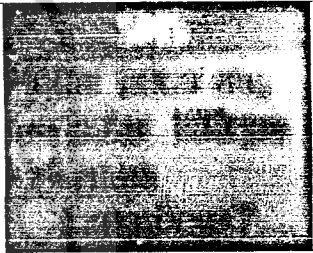
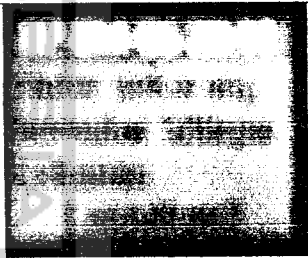
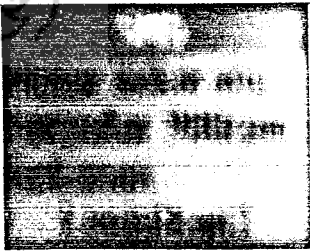
Gambar 4. 1. Tampilan alat hasil perancangan

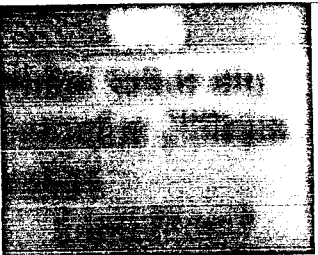
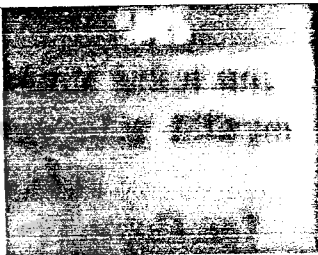
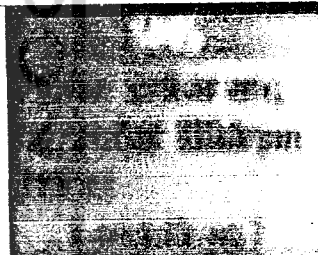
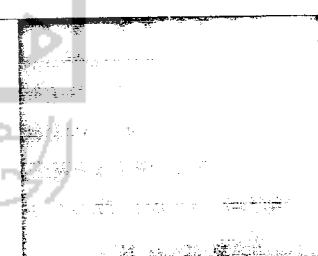
Pengujian alat dilakukan dengan mengirimkan pesan SMS dengan format yang sudah ditentukan ke nomor handphone 081392480461 yang terpasang pada alat.

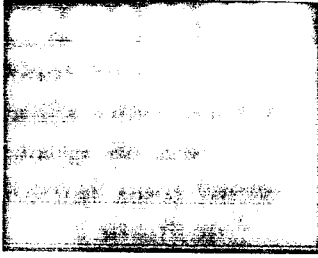
4.2. Pengamatan Pesan SMS

Pengamatan terhadap pesan SMS dimaksudkan untuk mengetahui tanggapan alat terhadap pesan SMS yang diterima. Data hasil pengamatan pesan SMS seperti pada Tabel 4.1 berikut.

Tabel 4. 1. Data Hasil Pengamatan Alat

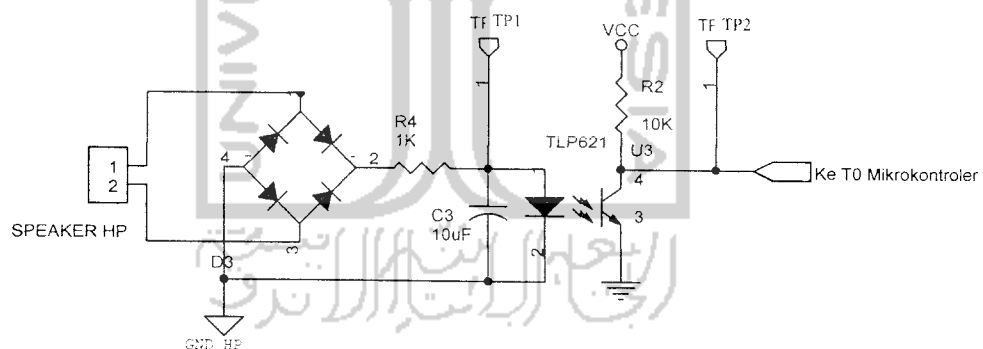
No	Pesan SMS	Putaran Motor	SMS Balasan
1.	Kanan 100	100 rpm	 "Motor sekarang berputar 100rpm kekanan"
2.	Kanan 500	500 rpm	 "Motor sekarang berputar 500rpm kekanan"
3.	Kanan 900	900 rpm	 "Motor sekarang berputar 900rpm kekanan"

4.	Kiri 200	200 rpm	 <p>”Motor sekarang berputar 200rpm kekiri”</p>
5.	Kiri 400	400 rpm	 <p>”Motor sekarang berputar 400rpm kekiri”</p>
6.	Kiri 800	800 rpm	 <p>”Motor sekarang berputar 800rpm kekiri”</p>
7.	Kiri 150	Diam	 <p>”Maaf format SMS anda salah silahkan ulangi dengan format yang benar”</p>

8	Kanan	Diam	 <p>”Maaf format SMS anda salah silahkan ulangi dengan format yang benar”</p>
---	-------	------	---

4.3. Pengamatan Rangkaian Deteksi Dering

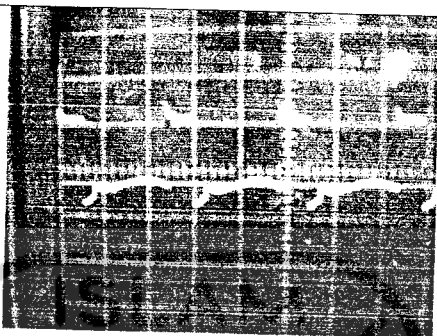
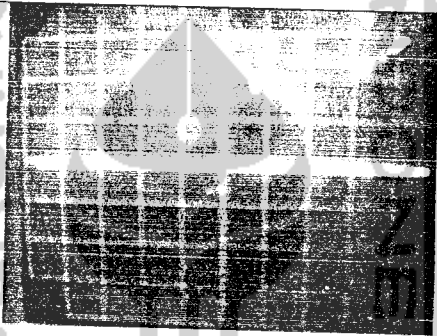
Pengamatan pada rangkaian deteksi dering dimaksudkan untuk mengetahui sinyal dan tegangan saat ada pesan SMS yang diterima maupun saat tidak menerima pesan SMS. Pengamatan dilakukan pada titik-titik pengukuran seperti yang ditunjukkan pada Gambar 4.1 berikut



Gambar 4. 2. Titik pengukuran pada rangkaian deteksi dering

Sinyal hasil pengamatan pada masing-masing titik pengukuran ditunjukkan pada Tabel 4.2 berikut.

Tabel 4. 2. Sinyal pengamatan rangkaian deteksi dering

Kondisi masukan	Bentuk Sinyal	Tegangan Pengukuran (V)	
		TP1	TP2
Ada SMS masuk		1 volt	0 volt
Tidak ada SMS masuk		0 volt	4,9 volt

Dari Tabel 4.2 diatas dapat dilihat bahwa saat ada sinyal dering tegangan pada TP1 sebesar 1 volt yang membuat optocoupler aktif sehingga keluaran pada TP2 sebesar 0 volt tegangan ini akan dideteksi oleh mikrokontroler sebagai indikator bahwa ada SMS baru pada *inbox*. Sedangkan jika tidak ada sinyal dering maka tegangan pada TP1 sebesar 0 volt yang membuat optocoupler tidak aktif sehingga tegangan pada TP2 mendekati tegangan Vcc yaitu sebesar 4,9 volt

4.4. Analisa Pengaturan Kecepatan Motor DC

Perhitungan kecepatan motor dalam satuan rpm (*rotate per minute*) didapat dengan asumsi bahwa banyaknya pulsa dalam 1 detik sama dengan

frekuensi PWM yaitu 15,82 Hz dan 1 pulsa PWM sama dengan 1 putaran motor maka kecepatan maksimum motor yang dapat diputar adalah $15,28 \times 60 \text{ detik} = 916,8 \text{ rpm (rotate per minute)}$.

Seperti yang telah dijelaskan sebelumnya bahwa pengaturan kecepatan motor dengan teknik PWM adalah dengan mengubah-ubah *duty cycle* pulsa PWM. Untuk kecepatan maksimum *duty cycle* 100% dan motor berhenti saat *duty cycle* 50%. maka dengan demikian kecepatan motor setiap perubahan 1% *duty cycle* adalah $916,8/50 = 18,336 \text{ rpm}$. Jika motor berputar ke kanan (CW) saat PWM *high* maka motor akan berputar kekanan (CW) saat *duty cycle* lebih besar dari 50% dan motor berputar kekiri (CCW) saat *duty cycle* lebih kecil dari 50%.

Data pengaturan kecepatan motor DC ditunjukkan pada Tabel 4.3 berikut

Tabel 4. 3. Data kecepatan motor

Duty cycle	Kecepatan (rpm)	Arah Putaran	Duty cycle	Kecepatan (rpm)	Arah Putaran
100%	916.8	Kanan	49%	-18.336	Kiri
99%	898.464	Kanan	48%	-36.672	Kiri
98%	880.128	Kanan	47%	-55.008	Kiri
97%	861.792	Kanan	46%	-73.344	Kiri
96%	843.456	Kanan	45%	-91.68	Kiri
95%	825.12	Kanan	44%	-110.016	Kiri
94%	806.784	Kanan	43%	-128.352	Kiri
93%	788.448	Kanan	42%	-146.688	Kiri
92%	770.112	Kanan	41%	-165.024	Kiri
91%	751.776	Kanan	40%	-183.36	Kiri
90%	733.44	Kanan	39%	-201.696	Kiri
89%	715.104	Kanan	38%	-220.032	Kiri
88%	696.768	Kanan	37%	-238.368	Kiri
87%	678.432	Kanan	36%	-256.704	Kiri
86%	660.096	Kanan	35%	-275.04	Kiri
85%	641.76	Kanan	34%	-293.376	Kiri
84%	623.424	Kanan	33%	-311.712	Kiri
83%	605.088	Kanan	32%	-330.048	Kiri
82%	586.752	Kanan	31%	-348.384	Kiri
81%	568.416	Kanan	30%	-366.72	Kiri
80%	550.08	Kanan	29%	-385.056	Kiri

Tabel 4.3. lanjutan

Duty cycle	Kecepatan (rpm)	Arah Putaran	Duty cycle	Kecepatan (rpm)	Arah Putaran
79%	531.744	Kanan	28%	-403.392	Kiri
78%	513.408	Kanan	27%	-421.728	Kiri
77%	495.072	Kanan	26%	-440.064	Kiri
76%	476.736	Kanan	25%	-458.4	Kiri
75%	458.4	Kanan	24%	-476.736	Kiri
74%	440.064	Kanan	23%	-495.072	Kiri
73%	421.728	Kanan	22%	-513.408	Kiri
72%	403.392	Kanan	21%	-531.744	Kiri
71%	385.056	Kanan	20%	-550.08	Kiri
70%	366.72	Kanan	19%	-568.416	Kiri
69%	348.384	Kanan	18%	-586.752	Kiri
68%	330.048	Kanan	17%	-605.088	Kiri
67%	311.712	Kanan	16%	-623.424	Kiri
66%	293.376	Kanan	15%	-641.76	Kiri
65%	275.04	Kanan	14%	-660.096	Kiri
64%	256.704	Kanan	13%	-678.432	Kiri
63%	238.368	Kanan	12%	-696.768	Kiri
62%	220.032	Kanan	11%	-715.104	Kiri
61%	201.696	Kanan	10%	-733.44	Kiri
60%	183.36	Kanan	9%	-751.776	Kiri
59%	165.024	Kanan	8%	-770.112	Kiri
58%	146.688	Kanan	7%	-788.448	Kiri
57%	128.352	Kanan	6%	-806.784	Kiri
56%	110.016	Kanan	5%	-825.12	Kiri
55%	91.68	Kanan	4%	-843.456	Kiri
54%	73.344	Kanan	3%	-861.792	Kiri
53%	55.008	Kanan	2%	-880.128	Kiri
52%	36.672	Kanan	1%	-898.464	Kiri
51%	18.336	Kanan	0%	-916.8	Kiri
50%	0	Berhenti			

Dari Tabel 4.3 diatas dapat dihitung selisih antara hasil perhitungan dengan data pembulatan kecepatan motor sebagai berikut

Tabel 4.4. Selisih perhitungan data kecepatan

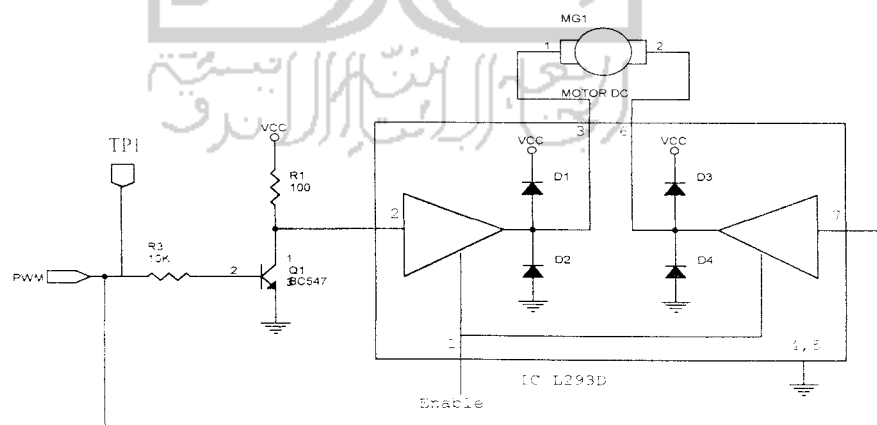
Kecepatan pembulatan	Kecepatan perhitungan	Selisih (Error)
900 rpm	898.464	1.536
800 rpm	806.784	6.784
700 rpm	696.768	3.232
600 rpm	605.088	5.088
500 rpm	495.072	4.928
400 rpm	403.392	3.392
300 rpm	293.376	6.624
200 rpm	201.696	1.696
100 rpm	91.68	8.32

Dari Tabel 4.4 diatas didapat selisih rata-rata untuk 9 buah data adalah :

$$\begin{aligned}\text{Selisih}_{\text{rata-rata}} &= \text{Jumlah selisih} / \text{Jumlah data} \\ &= 41.6/9 = 4.62\end{aligned}$$

4.5. Pengamatan Sinyal PWM

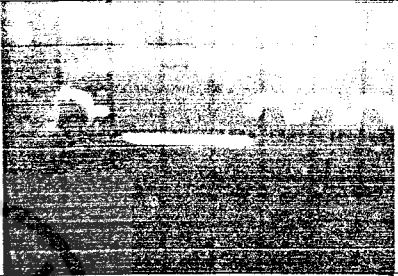
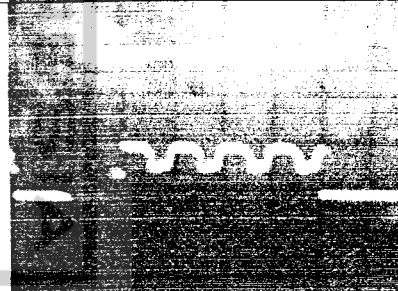
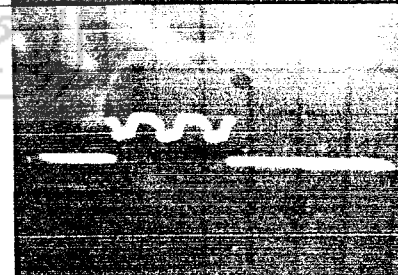
Dari data pada Tabel 4.3 diatas dapat diamati bentuk sinyal PWM untuk setiap perubahan kecepatan dan arah putaran motor. Pengukuran dilakukan pada keluaran PWM yaitu pada titik pengukuran TP1 seperti pada Gambar 4.3 berikut

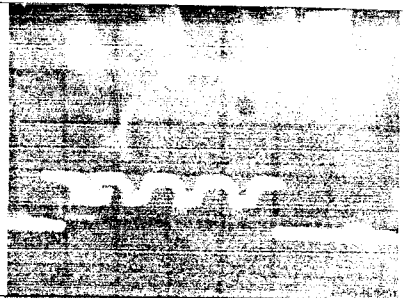
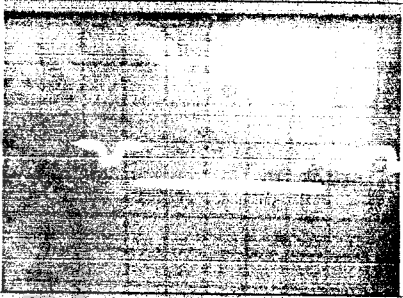
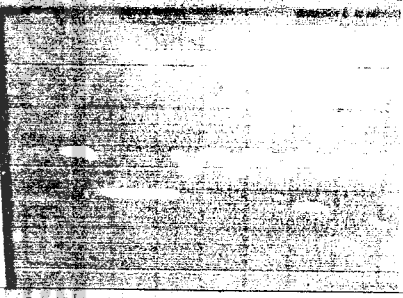
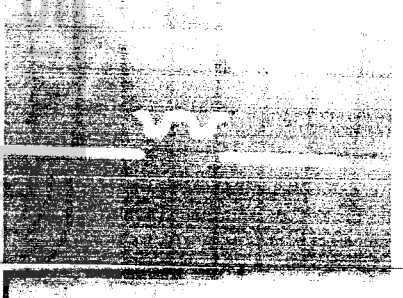
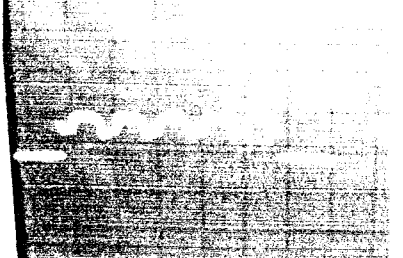


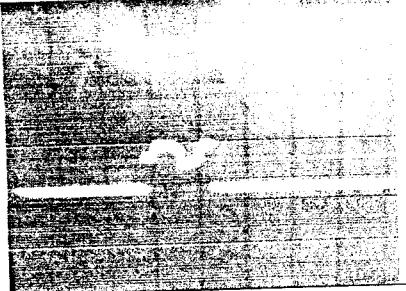
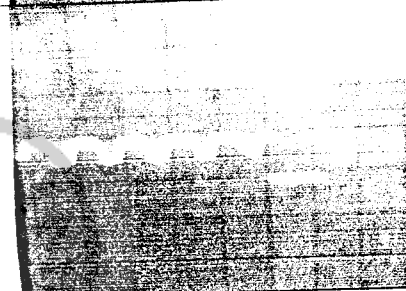
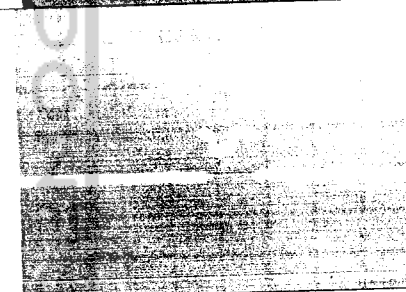
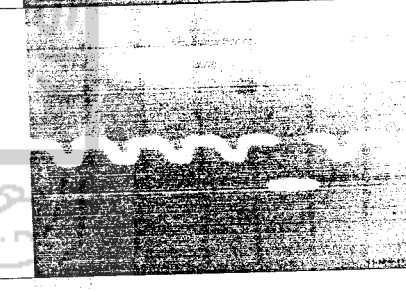
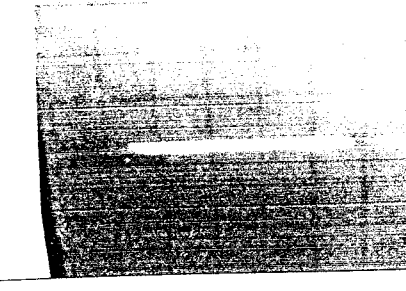
Gambar 4.3. Rangkaian pengukuran sinyal PWM

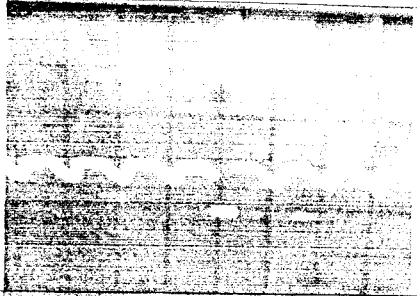
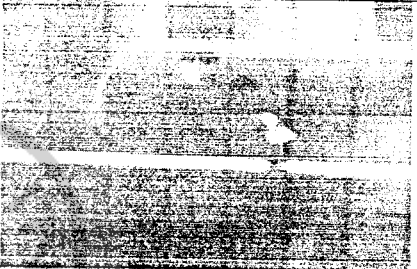
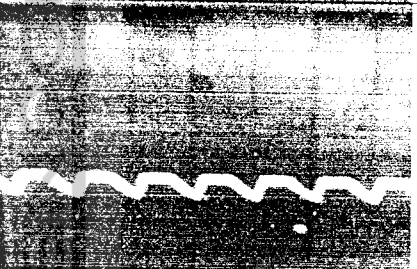
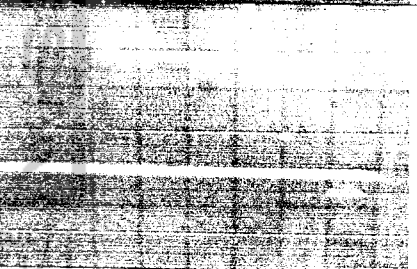
Hasil pengamatan bentuk sinyal PWM pada setiap kecepatan dan arah putaran seperti pada Tabel 4.5 berikut

Tabel 4. 5. Bentuk gelombang PWM

Kecepatan	Arah Putaran	Duty cycle	Sinyal PWM
100 rpm	Kanan	56%	
	Kiri	45 %	
200 rpm	Kanan	61 %	
	Kiri	39 %	

300 rpm	Kanan	66 %	
	Kiri	34 %	
400 rpm	Kanan	72 %	
	Kiri	28 %	
500 rpm	Kanan	77 %	

	Kiri	23 %	
600 rpm	Kanan	83 %	
	Kiri	17 %	
700 rpm	Kanan	88 %	
	Kiri	12 %	

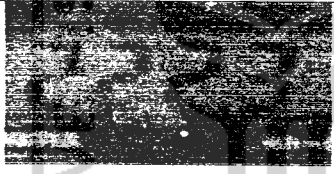
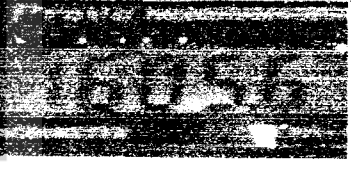
800 rpm	Kanan	94 %	
	Kiri	6 %	
900 rpm	Kanan	99 %	
	Kiri	1 %	

Dari Tabel 4.5 diatas, dapat diamati bahwa sinyal PWM saat putar kanan sinyal dominan *high* dan saat putar kiri sinyal dominan *low*. Untuk putaran ke kanan semakin besar kecepatannya *duty cyclenya* akan semakin besar. Sedangkan untuk putaran kekiri semakin besar kecepatannya *duty caclenya* semakin kecil.

4.6. Kalibrasi Alat

Kalibrasi alat digunakan untuk mengetahui ketelitian dari alat yang dibuat. Pada kalibrasi ini kecepatan putaran motor yang dihasilkan dibandingkan dengan kecepatan yang terbaca oleh tachometer. Data hasil perbandingan kecepatan motor antara alat dan tachometer seperti pada Tabel 4.6 berikut ini:

Tabel 4.6. Perbandingan kecepatan motor antara alat dan tachometer

Kecepatan Alat	Kecepatan Tachometer	Kecepatan Alat	Kecepatan Tachometer
100 rpm		600 rpm	
200 rpm		700 rpm	
300 rpm		800 rpm	
400 rpm		900 rpm	
500 rpm			

Dari data tabel diatas dapat dilihat bahwa terdapat perbedaan kecepatan antara kecepatan yang dibuat oleh alat dengan kecepatan hasil pengukuran Tachometer. Hal ini dapat disebabkan karena beberapa hal seperti :

1. Pembacaan kecepatan pada tachometer tidak stabil atau berubah-ubah yang disebabkan karena permukaan piringan ukur yang digunakan pada motor DC tidak rata atau bergelombang.
2. Asumsi tentang kecepatan putaran dalam satuan rpm yang diambil dalam perancangan tidak sesuai dengan kecepatan putaran sebenarnya. Pada perancangan perhitungan kecepatan motor dalam satuan rpm (*rotate per minute*) diasumsikan sama dengan banyaknya pulsa dalam 1 detik atau sama dengan frekuensi PWM yaitu 15,82 Hz dan 1 pulsa PWM sama dengan 1 putaran motor maka jika kecepatan diambil dalam satuan rpm (*rotate per minute*) kecepatan maksimum motor yang dapat diputar adalah $15,28 \times 60 \text{ detik} = 916,8 \text{ rpm}$. Sehingga dengan demikian nilai *dutycycle* yang dimasukkan ke program juga tidak sesuai dengan kecepatan yang sebenarnya.

BAB V

PENUTUP

5.1. Kesimpulan

Dari perancangan dan pembahasan pada bab III dan IV dapat disimpulkan beberapa hal yaitu :

1. Alat yang dibuat dapat melakukan pengaturan kecepatan untuk setiap perintah SMS yang berbeda
2. Kecepatan motor yang dihasilkan belum sesuai dengan kecepatan yang sebenarnya (hasil pembacaan tachometer). Hal ini mungkin disebabkan karena asumsi kecepatan dari frekuensi *duty cycle* tidak sesuai dengan nilai kecepatan dalam satuan rpm.
3. Pengendalian kecepatan motor dapat dilakukan dari semua jenis nomor handphone dan dari mana saja.

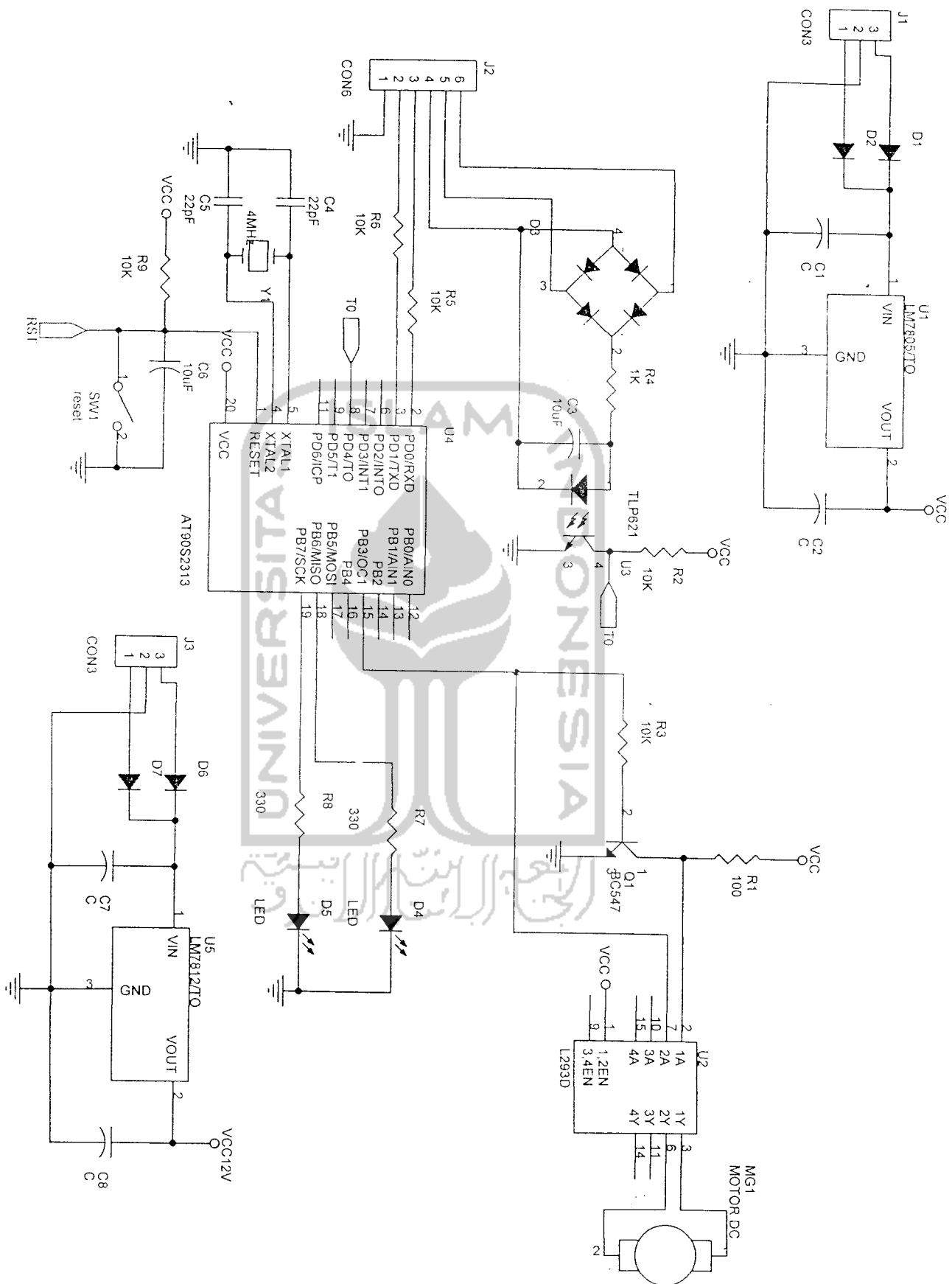
5.2. Saran

Saran yang dapat diberikan untuk pengembangan dari alat ini adalah sebagai berikut: Agar kecepatan motor DC yang dihasilkan alat sesuai dengan kecepatan putaran sebenarnya, maka harus diketahui penurunan dari nilai frekuensi PWM ke satuan kecepatan dalam rpm (*rotate per minutes*).

DAFTAR PUSTAKA

1. Atmel, 2001, *AVR Microcontroller AT90S2313 Datasheet*, <http://www.atmel.com>, Atmel Semiconductor.
2. Atmel, 2001, *AVR Microcontroller AT90S2313 Instruction Set*, <http://www.atmel.com>, Atmel Semiconductor.
3. Malvino, Paul, Albert., 1999, *Prinsip-prinsip Elektronika*, Salemba Teknika, Jakarta.
4. Siemens, 2000, *AT Command Set for the SIEMENS Mobile Phones S35i, C35i, M35i*, <http://www.siemens.com>, Siemens, Germany
5. Microelectronics, Thomson, SGS . 1994. *L293D Push-Pull Four Channel Driver With Ddiodes*, Singapore.





AVRASM ver. 1.56 D:\AVR\septa.asm Fri Sep 29 07:48:23 2006

warning : Register already defined by the .DEF directive
warning : Register already defined by the .DEF directive
D:\AVR\septa.asm(750): warning: A .db segment with an odd number of bytes is
detected. A zero byte is added.
D:\AVR\septa.asm(753): warning: A .db segment with an odd number of bytes is
detected. A zero byte is added.
D:\AVR\septa.asm(782): warning: A .db segment with an odd number of bytes is
detected. A zero byte is added.
D:\AVR\septa.asm(787): warning: A .db segment with an odd number of bytes is
detected. A zero byte is added.

.nolist ;Suppress listing of
include file

```
.equ  pesanSMS      =0x60
.equ   SMSC         =0x62

.def   temp3        =R12
.def   temp2        =R13
.def   jumlahnomor  =R14
.def   temp1        =R15
.def   temp         =R16      ;temporary storage register
.def   timeout      =R17
.def   alamatnomor  =R18
.def   alamatpesan  =R19      ;Data to be transmitted
.def   Rxbyte       =R20      ;Received data
.def   Txbyte       =R21
.def   delay3       =R22
.def   delay1       =R23
.def   delay2       =R24
.def   kecepatan    =R25
.def   titip        =R26
.def   titip1       =R27

.cseg
;=====
.org 0000
000000 c00a      rjmp  reset
000001 0000      nop
000002 0000      nop
000003 0000      nop
000004 0000      nop
000005 0000      nop
000006 0000      nop
000007 0000      nop
000008 0000      nop
000009 0000      nop
00000a 0000      nop
;
00000b ed0f      reset: ldi   temp,RAMEND
00000c bf0d      out    SPL,temp                      ;Init Stack Pointer
00000d ee0e      ldi    temp,0b11101110
00000e bb01      out    DDRD,temp                    ; Set PORTB sebagai input
00000f ef07      ldi    temp,0b11101111
000010 bb07      out    DDRB,temp                    ; Set PORTB sebagai input
000011 e30f      ldi    temp,0b00111111
000012 bb08      out    PORTB,temp                  ; Set PORTB sebagai input
;
UART Initialisation
000013 e00c      ldi    temp,12
000014 b909      out    UBRR,temp
```

```

                                septa.lst
000015 e108                ldi    temp,0b00011000
000016 b90a                out    UCR,temp
000017 9a94    sinil:      sbi    PortD,4

                                tunggu:
000018 9984                sbic    PIND,4
000019 cffe                rjmp    tunggu
00001a ef07                ldi    temp,0b11110111
00001b bb07                out    DDRB,temp                ; Set PORTB sebagai input
00001c e30f                ldi    temp,0b00111111
00001d bb08                out    PORTB,temp                ; Set PORTB sebagai input
00001e d263    sini:      rcall   Ldelay
00001f e409                ldi    temp,73
000020 e6c0                ldi    Yl,pesanSMS

;=====
; Baca SMS yang diterima
;=====
000021 elec                ldi    zl,low(msg*2)
000022 e0f5                ldi    zh,high(msg*2)
000023 d248                rcall   kirim
000024 d253    jj:         rcall   receive
000025 334a                cpi     Rxbyte,':'
000026 f7e9                brne    jj
000027 d250    ii:         rcall   receive
000028 304a                cpi     Rxbyte,0x0A
000029 f7e9                brne    ii
                                ulangan:
00002a d24d                rcall   receive
00002b 9349                st      Y+,Rxbyte
00002c 950a                dec     temp
00002d 2300                tst     temp
00002e f7d9                brne    ulangan
                                BacaSMS:
00002f e6c0                ldi    Yl,pesanSMS
000030 9621                adiw    Yl,1
000031 8108                ld      temp,Y
000032 5300                subi    temp,0x30
000033 3005                cpi     temp,0x05
000034 f411                brne    kesana
000035 e00a                ldi    temp,0x0A
000036 c007                rjmp    lanjut
                                kesana:
000037 3006                cpi     temp,0x06
000038 f411                brne    kesana1
000039 e00c                ldi    temp,0x0C
00003a c003                rjmp    lanjut
                                kesana1:
00003b 3007                cpi     temp,0x07
00003c f409                brne    lanjut
00003d e00e                ldi    temp,0x0E
                                lanjut:
00003e e6c2                ldi    Yl,SMSC
00003f d021                rcall   tambah
000040 95c3                inc     Yl
000041 95c3                inc     Yl
000042 2f2c                mov     alamatnomor,Yl
000043 95c3                inc     Yl
000044 8108                ld      temp,Y
000045 5307                subi    temp,0x37
000046 2ed0                mov     temp2,temp
000047 300b                cpi     temp,0x0B
000048 f411                brne    kesitu
000049 e00c                ldi    temp,0x0C
00004a c003                rjmp    terusin

```

```

                                septa.lst
kesitu:
00004b 300d      cpi      temp,0x0D
00004c f409      brne     terusin
00004d e00e      ldi      temp,0x0E
        terusin:
00004e 2fa0      mov      titip,temp
00004f 9614      adiw     titip,0x04
000050 2eea      mov      jumlahnomor,titip
000051 95c3      inc      Y1
000052 95c3      inc      Y1
000053 d00d      rcall    tambah
000054 e103      ldi      temp,19
000055 d00b      rcall    tambah
000056 95c3      inc      Y1
000057 8108      ld       temp,Y
000058 95c3      inc      Y1
000059 2f3c      mov      alamatpesan,Y1
00005a 5300      subi     temp,0x30
00005b 3008      cpi      temp,0x08
00005c f409      brne     kesitu1
00005d c02f      rjmp     Putarkiri
        kesitu1:
00005e 3009      cpi      temp,0x09
00005f f431      brne     pesansalah
000060 c0da      rjmp     Putarkanan
;
000061 950a      tambah:  dec      temp
000062 95c3      inc      Y1
000063 2300      tst      temp
000064 f7e1      brne     tambah
000065 9508      ret
;
        pesansalah:
000066 d21b      rcall    Ldelay
000067 2d0d      mov      temp,temp2
000068 300a      cpi      temp,0x0A
000069 f421      brne     salah1
00006a e2e8      ikut:   ldi      z1,low(msg1*2)
00006b e0f5      ldi      zh,high(msg1*2)
00006c d1ff      rcall    kirim
00006d c00c      rjmp     laju
00006e 300b      salah1: cpi      temp,0x0B
00006f f409      brne     salah2
000070 cff9      rjmp     ikut
000071 300c      salah2: cpi      temp,0x0C
000072 f421      brne     salah3
000073 e3e4      ikut1:  ldi      z1,low(msg2*2)
000074 e0f5      ldi      zh,high(msg2*2)
000075 d1f6      rcall    kirim
000076 c003      rjmp     laju
000077 300d      salah3: cpi      temp,0x0D
000078 f769      brne     pesansalah
000079 cff9      rjmp     ikut1
00007a d1fd      laju:   rcall    receive
00007b 334e      cpi      Rxbyte,'>'
00007c f7e9      brne     laju
00007d e4e0      ldi      z1,low(noSMSC*2)
00007e e0f5      ldi      zh,high(noSMSC*2)
00007f d1ec      rcall    kirim
000080 2d0e      mov      temp,jumlahnomor
000081 2fc2      mov      Y1,alamatnomor
        sampainol:
000082 9159      ld       txbyte,Y+

```

```

                                septa.lst
000083 d1f0      rcall transmit
000084 950a      dec temp
000085 2300      tst temp
000086 f7d9      brne sampainol
000087 e5e4      ldi z1,low(isisalah*2)
000088 e0f5      ldi zh,high(isisalah*2)
000089 d1e2      rcall kirim
00008a d1f7      rcall Ldelay
00008b d1f6      rcall Ldelay
00008c cf8a      rjmp sinil

```

```

; Putarkiri:
00008d ede4      ldi z1,low(Kiri*2)
00008e e0f5      ldi zh,high(Kiri*2)
00008f 2fc3      mov Y1,alamatpesan
000090 e008      ldi temp,0x08
000091 2ef0      mov temp1,temp
000092 8108      lagi: ld temp,Y
000093 95c8      lpm
000094 1600      cp R0,temp
000095 f681      brne pesansalah
000096 d152      rcall tambah1
000097 f7d1      brne lagi
000098 2fac      mov titip,Y1
000099 edec      ldi z1,low(Kiri100*2)
00009a e0f5      ldi zh,high(Kiri100*2)
00009b e006      ldi temp,0x06
00009c 2ef0      mov temp1,temp
00009d 8108      lagi1: ld temp,Y
00009e 95c8      lpm
00009f 1600      cp r0,temp
0000a0 f449      brne cek_lagi
0000a1 d147      rcall tambah1
0000a2 f7d1      brne lagi1
0000a3 e0a0      ldi titip,0x00
0000a4 eeb0      ldi titip1,0xE0
0000a5 e303      ldi temp,0x33
0000a6 2ef0      mov temp1,temp
0000a7 e301      ldi temp,0x31
0000a8 2ec0      mov temp3,temp
0000a9 c144      rjmp Eksekusi

```

```

cek_lagi:
0000aa 2fca      mov Y1,titip
0000ab eee2      ldi z1,low(Kiri200*2)
0000ac e0f5      ldi zh,high(Kiri200*2)
0000ad e006      ldi temp,0x06
0000ae 2ef0      mov temp1,temp
0000af 8108      lagi2: ld temp,Y
0000b0 95c8      lpm
0000b1 1600      cp r0,temp
0000b2 f449      brne cek_lagi1
0000b3 d135      rcall tambah1
0000b4 f7d1      brne lagi2
0000b5 e0a0      ldi titip,0x00
0000b6 ecb7      ldi titip1,0xC7
0000b7 e303      ldi temp,0x33
0000b8 2ef0      mov temp1,temp
0000b9 e302      ldi temp,0x32
0000ba 2ec0      mov temp3,temp
0000bb c132      rjmp Eksekusi

```

```

cek_lagi1:
0000bc 2fca      mov Y1,titip
0000bd eee8      ldi z1,low(Kiri300*2)

```

```

                                septa.lst
0000be e0f5          ldi      zh,high(Kiri300*2)
0000bf e006          ldi      temp,0x06
0000c0 2ef0          mov      temp1,temp
0000c1 8108          lagi3: ld      temp,Y
0000c2 95c8          lpm
0000c3 1600          cp        r0,temp
0000c4 f449          brne     cek_lagi2
0000c5 d123          rcall    tambah1
0000c6 f7d1          brne     lagi3
0000c7 e0a0          ldi      titip,0x00
0000c8 eab8          ldi      titip1,0xA8
0000c9 e303          ldi      temp,0x33
0000ca 2ef0          mov      temp1,temp
0000cb e303          ldi      temp,0x33
0000cc 2ec0          mov      temp3,temp
0000cd c120          rjmp     Eksekusi

cek_lagi2:
0000ce 2fca          mov      Yl,titip
0000cf eeee          ldi      z1,low(Kiri400*2)
0000d0 e0f5          ldi      zh,high(Kiri400*2)
0000d1 e006          ldi      temp,0x06
0000d2 2ef0          mov      temp1,temp
0000d3 8108          lagi3i: ld      temp,Y
0000d4 95c8          lpm
0000d5 1600          cp        r0,temp
0000d6 f449          brne     cek_lagi3
0000d7 d111          rcall    tambah1
0000d8 f7d1          brne     lagi3i
0000d9 e0a0          ldi      titip,0x00
0000da e8bf          ldi      titip1,0x8F
0000db e303          ldi      temp,0x33
0000dc 2ef0          mov      temp1,temp
0000dd e304          ldi      temp,0x34
0000de 2ec0          mov      temp3,temp
0000df c10e          rjmp     Eksekusi

cek_lagi3:
0000e0 2fca          mov      Yl,titip
0000e1 efe4          ldi      z1,low(Kiri500*2)
0000e2 e0f5          ldi      zh,high(Kiri500*2)
0000e3 e006          ldi      temp,0x06
0000e4 2ef0          mov      temp1,temp
0000e5 8108          lagi4: ld      temp,Y
0000e6 95c8          lpm
0000e7 1600          cp        r0,temp
0000e8 f449          brne     cek_lagi4
0000e9 d0ff          rcall    tambah1
0000ea f7d1          brne     lagi4
0000eb e0a0          ldi      titip,0x00
0000ec e7b0          ldi      titip1,0x70
0000ed e303          ldi      temp,0x33
0000ee 2ef0          mov      temp1,temp
0000ef e305          ldi      temp,0x35
0000f0 2ec0          mov      temp3,temp
0000f1 c0fc          rjmp     Eksekusi

cek_lagi4:
0000f2 2fca          mov      Yl,titip
0000f3 efea          ldi      z1,low(Kiri600*2)
0000f4 e0f5          ldi      zh,high(Kiri600*2)
0000f5 e006          ldi      temp,0x06
0000f6 2ef0          mov      temp1,temp
0000f7 8108          lagi5: ld      temp,Y
0000f8 95c8          lpm
0000f9 1600          cp        r0,temp

```

```

                                septa.lst
0000fa f449                brne    cek_lagi5
0000fb d0ed                rcall   tambah1
0000fc f7d1                brne    lagi5
0000fd e0a0                ldi     titip,0x00
0000fe e5b6                ldi     titip1,0x56
0000ff e303                ldi     temp,0x33
000100 2ef0                mov     temp1,temp
000101 e306                ldi     temp,0x36
000102 2ec0                mov     temp3,temp
000103 c0ea                rjmp    Eksekusi

cek_lagi5:
000104 2fca                mov     Yl,titip
000105 e0e0                ldi     z1,low(Kiri700*2)
000106 e0f6                ldi     zh,high(Kiri700*2)
000107 e006                ldi     temp,0x06
000108 2ef0                mov     temp1,temp
000109 8108                lagi6: ld     temp,Y
00010a 95c8                lpm
00010b 1600                cp      r0,temp
00010c f449                brne    cek_lagi6
00010d d0db                rcall   tambah1
00010e f7d1                brne    lagi6
00010f e0a0                ldi     titip,0x00
000110 e3b8                ldi     titip1,0x38
000111 e303                ldi     temp,0x33
000112 2ef0                mov     temp1,temp
000113 e307                ldi     temp,0x37
000114 2ec0                mov     temp3,temp
000115 c0d8                rjmp    Eksekusi

cek_lagi6:
000116 2fca                mov     Yl,titip
000117 e0e6                ldi     z1,low(Kiri800*2)
000118 e0f6                ldi     zh,high(Kiri800*2)
000119 e006                ldi     temp,0x06
00011a 2ef0                mov     temp1,temp
00011b 8108                lagi7: ld     temp,Y
00011c 95c8                lpm
00011d 1600                cp      r0,temp
00011e f449                brne    cek_lagi7
00011f d0c9                rcall   tambah1
000120 f7d1                brne    lagi7
000121 e0a0                ldi     titip,0x00
000122 e1be                ldi     titip1,0x1E
000123 e303                ldi     temp,0x33
000124 2ef0                mov     temp1,temp
000125 e308                ldi     temp,0x38
000126 2ec0                mov     temp3,temp
000127 c0c6                rjmp    Eksekusi

cek_lagi7:
000128 2fca                mov     Yl,titip
000129 e0ec                ldi     z1,low(Kiri900*2)
00012a e0f6                ldi     zh,high(Kiri900*2)
00012b e006                ldi     temp,0x06
00012c 2ef0                mov     temp1,temp
00012d 8108                lagi8: ld     temp,Y
00012e 95c8                lpm
00012f 1600                cp      r0,temp
000130 f449                brne    cek_lagi8
000131 d0b7                rcall   tambah1
000132 f7d1                brne    lagi8
000133 e0a0                ldi     titip,0x00
000134 e0b5                ldi     titip1,0x05
000135 e303                ldi     temp,0x33

```

```

000136 2ef0          mov     septa.1st
000137 e309          ldi     temp1,temp
000138 2ec0          mov     temp,0x39
000139 c0b4          mov     temp3,temp
                      rjmp    Eksekusi
cek_lagi8:
00013a cf2b          rjmp    pesansalah
;
PutarKanan:
00013b e1e2          ldi     z1,low(Kanan*2)
00013c e0f6          ldi     zh,high(Kanan*2)
00013d 2fc3          mov     Y1,alamatpesan
00013e e00a          ldi     temp,0x0A
00013f 2ef0          mov     temp1,temp
000140 8108          ulang: ld     temp,Y
000141 95c8          lpm
000142 1600          cp     R0,temp
000143 f7b1          brne   cek_lagi8
000144 d0a4          rcall  tambah1
000145 f7d1          brne   ulang
000146 2fac          mov     titip,Y1
000147 e1ec          ldi     z1,low(Kanan100*2)
000148 e0f6          ldi     zh,high(Kanan100*2)
000149 e006          ldi     temp,0x06
00014a 2ef0          mov     temp1,temp
00014b 8108          ulang1: ld     temp,Y
00014c 95c8          lpm
00014d 1600          cp     r0,temp
00014e f449          brne   ceklagi
00014f d099          rcall  tambah1
000150 f7d1          brne   ulang1
000151 e0a1          ldi     titip,0x01
000152 e1be          ldi     titip1,0x1E
000153 e303          ldi     temp,0x33
000154 2ef0          mov     temp1,temp
000155 e301          ldi     temp,0x31
000156 2ec0          mov     temp3,temp
000157 c0d5          rjmp    Eksekusi1
ceklagi:
000158 2fca          mov     Y1,titip
000159 e2e2          ldi     z1,low(Kanan200*2)
00015a e0f6          ldi     zh,high(Kanan200*2)
00015b e006          ldi     temp,0x06
00015c 2ef0          mov     temp1,temp
00015d 8108          ulang2: ld     temp,Y
00015e 95c8          lpm
00015f 1600          cp     r0,temp
000160 f449          brne   ceklagi1
000161 d087          rcall  tambah1
000162 f7d1          brne   ulang2
000163 e0a1          ldi     titip,0x01
000164 e3b7          ldi     titip1,0x37
000165 e303          ldi     temp,0x33
000166 2ef0          mov     temp1,temp
000167 e302          ldi     temp,0x32
000168 2ec0          mov     temp3,temp
000169 c0c3          rjmp    Eksekusi1
ceklagi1:
00016a 2fca          mov     Y1,titip
00016b e2e8          ldi     z1,low(Kanan300*2)
00016c e0f6          ldi     zh,high(Kanan300*2)
00016d e006          ldi     temp,0x06
00016e 2ef0          mov     temp1,temp
00016f 8108          ulang3: ld     temp,Y

```

```

                                septa.lst
000170 95c8                    lpm
000171 1600                    cp
000172 f449                    brne    r0,temp
000173 d075                    rcall   ceklagi2
000174 f7d1                    brne    tambah1
000175 e0a1                    ldi     ulang3
000176 e5b6                    ldi     titip,0x01
000177 e303                    ldi     titip1,0x56
000178 2ef0                    ldi     temp,0x33
000179 e303                    mov     temp1,temp
00017a 2ec0                    ldi     temp,0x33
00017b c0b1                    mov     temp3,temp
                                rjmp     Eksekusi1
                                ceklagi2:
00017c 2fca                    mov     Y1,titip
00017d e2ee                    ldi     z1,low(Kanan400*2)
00017e e0f6                    ldi     zh,high(Kanan400*2)
00017f e006                    ldi     temp,0x06
000180 2ef0                    mov     temp1,temp
000181 8108                    ulang3i:ld temp,Y
000182 95c8                    lpm
000183 1600                    cp
000184 f449                    brne    ceklagi3
000185 d063                    rcall   tambah1
000186 f7d1                    brne    ulang3i
000187 e0a1                    ldi     titip,0x01
000188 e6bf                    ldi     titip1,0x6F
000189 e303                    ldi     temp,0x33
00018a 2ef0                    mov     temp1,temp
00018b e304                    ldi     temp,0x34
00018c 2ec0                    mov     temp3,temp
00018d c09f                    rjmp     Eksekusi1
                                ceklagi3:
00018e 2fca                    mov     Y1,titip
00018f e3e4                    ldi     z1,low(Kanan500*2)
000190 e0f6                    ldi     zh,high(Kanan500*2)
000191 e006                    ldi     temp,0x06
000192 2ef0                    mov     temp1,temp
000193 8108                    ulang4:ld temp,Y
000194 95c8                    lpm
000195 1600                    cp
000196 f449                    brne    ceklagi4
000197 d051                    rcall   tambah1
000198 f7d1                    brne    ulang4
000199 e0a1                    ldi     titip,0x01
00019a e8be                    ldi     titip1,0x8E
00019b e303                    ldi     temp,0x33
00019c 2ef0                    mov     temp1,temp
00019d e305                    ldi     temp,0x35
00019e 2ec0                    mov     temp3,temp
00019f c08d                    rjmp     Eksekusi1
                                ceklagi4:
0001a0 2fca                    mov     Y1,titip
0001a1 e3ea                    ldi     z1,low(Kanan600*2)
0001a2 e0f6                    ldi     zh,high(Kanan600*2)
0001a3 e006                    ldi     temp,0x06
0001a4 2ef0                    mov     temp1,temp
0001a5 8108                    ulang5:ld temp,Y
0001a6 95c8                    lpm
0001a7 1600                    cp
0001a8 f449                    brne    ceklagi5
0001a9 d03f                    rcall   tambah1
0001aa f7d1                    brne    ulang5
0001ab e0a1                    ldi     titip,0x01

```

```

0001ac eab8          ldi      septa.1st
0001ad e303          ldi      titip1,0xA8
0001ae 2ef0          mov      temp,0x33
0001af e306          ldi      temp1,temp
0001b0 2ec0          ldi      temp,0x36
0001b1 c07b          mov      temp3,temp
                      rjmp     Eksekusi1
ceklagi5:
0001b2 2fca          mov      Y1,titip
0001b3 e4e0          ldi      z1,low(Kanan700*2)
0001b4 e0f6          ldi      zh,high(Kanan700*2)
0001b5 e006          ldi      temp,0x06
0001b6 2ef0          mov      temp1,temp
0001b7 8108          ulang6:  ld      temp,Y
0001b8 95c8          lpm
0001b9 1600          cp      r0,temp
0001ba f449          brne    ceklagi6
0001bb d02d          rcall   tambah1
0001bc f7d1          brne    ulang6
0001bd e0a1          ldi      titip,0x01
0001be ecb6          ldi      titip1,0xC6
0001bf e303          ldi      temp,0x33
0001c0 2ef0          mov      temp1,temp
0001c1 e307          ldi      temp,0x37
0001c2 2ec0          mov      temp3,temp
0001c3 c069          rjmp     Eksekusi1
ceklagi6:
0001c4 2fca          mov      Y1,titip
0001c5 e4e6          ldi      z1,low(Kanan800*2)
0001c6 e0f6          ldi      zh,high(Kanan800*2)
0001c7 e006          ldi      temp,0x06
0001c8 2ef0          mov      temp1,temp
0001c9 8108          ulang7:  ld      temp,Y
0001ca 95c8          lpm
0001cb 1600          cp      r0,temp
0001cc f449          brne    ceklagi7
0001cd d01b          rcall   tambah1
0001ce f7d1          brne    ulang7
0001cf e0a1          ldi      titip,0x01
0001d0 eeb0          ldi      titip1,0xE0
0001d1 e303          ldi      temp,0x33
0001d2 2ef0          mov      temp1,temp
0001d3 e308          ldi      temp,0x38
0001d4 2ec0          mov      temp3,temp
0001d5 c057          rjmp     Eksekusi1
ceklagi7:
0001d6 2fca          mov      Y1,titip
0001d7 e4ec          ldi      z1,low(Kanan900*2)
0001d8 e0f6          ldi      zh,high(Kanan900*2)
0001d9 e006          ldi      temp,0x06
0001da 2ef0          mov      temp1,temp
0001db 8108          ulang8:  ld      temp,Y
0001dc 95c8          lpm
0001dd 1600          cp      r0,temp
0001de f449          brne    ceklagi8
0001df d009          rcall   tambah1
0001e0 f7d1          brne    ulang8
0001e1 e0a1          ldi      titip,0x01
0001e2 efb9          ldi      titip1,0xF9
0001e3 e303          ldi      temp,0x33
0001e4 2ef0          mov      temp1,temp
0001e5 e309          ldi      temp,0x39
0001e6 2ec0          mov      temp3,temp
0001e7 c045          rjmp     Eksekusi1

```

```

ceklagi8:
0001e8 ce7d      rjmp    pesansalah
;
    tambah1:
0001e9 95e3      inc     z1
0001ea 95c3      inc     y1
0001eb 94fa      dec     temp1
0001ec 20ff      tst     temp1
0001ed 9508      ret
;
    Eksekusi:
0001ee d093      rcall   Ldelay
0001ef 2d0d      mov     temp,temp2
0001f0 300a      cpi     temp,0x0A
0001f1 f421      brne    bukan
0001f2 e5e2      goo:    ldi     z1,low(msgkiri1*2)
0001f3 e0f6      ldi     zh,high(msgkiri1*2)
0001f4 d077      rcall   kirim
0001f5 c00c      rjmp    terus
0001f6 300b      bukan:  cpi     temp,0x0B
0001f7 f409      brne    bukan1
0001f8 cff9      rjmp    goo
0001f9 300c      bukan1: cpi     temp,0x0C
0001fa f421      brne    bukan2
0001fb e5ee      goo1:   ldi     z1,low(msgkiri2*2)
0001fc e0f6      ldi     zh,high(msgkiri2*2)
0001fd d06e      rcall   kirim
0001fe c003      rjmp    terus
0001ff 300d      bukan2: cpi     temp,0x0D
000200 f769      brne    Eksekusi
000201 cff9      rjmp    goo1
000202 d075      terus: rcall   receive
000203 334e      cpi     Rxbyte,'>'
000204 f7e9      brne    terus
000205 e4e0      ldi     z1,low(nosMSC*2)
000206 e0f5      ldi     zh,high(nosMSC*2)
000207 d064      rcall   kirim
000208 2d0e      mov     temp,jumlahnomor
000209 2fc2      mov     y1,alamatnomor
    sampainol1:
00020a 9159      ld      txbyte,y+
00020b d068      rcall   transmit
00020c 950a      dec     temp
00020d 2300      tst     temp
00020e f7d9      brne    sampainol1
00020f e6ea      ldi     z1,low(pesankiri1*2)
000210 e0f6      ldi     zh,high(pesankiri1*2)
000211 d05a      rcall   kirim
000212 2d5f      mov     txbyte,temp1
000213 d060      rcall   transmit
000214 2d5c      mov     txbyte,temp3
000215 d05e      rcall   transmit
000216 e9ee      ldi     z1,low(pesankiri2*2)
000217 e0f6      ldi     zh,high(pesankiri2*2)
000218 d053      rcall   kirim
000219 d068      rcall   Ldelay
00021a d067      rcall   Ldelay
00021b 9ac7      sbi     portB,7
00021c 98c6      cbi     portB,6
00021d 9a94      -looper: sbi     PortD,4
00021e e802      ldi     temp,0b10000010
00021f bd0f      out     TCCR1A,temp
000220 e004      ldi     temp,0b00000100

```

```

                                septa.lst
000221 bd0e          out      TCCR1B,temp          ; Clear Timer/Counter1 on
Compare Match                                           ; Set prescalar to 256
000222 bdab          out      OCR1AH,titip
000223 bdba          out      OCR1AL,titip1
000224 ef0f          ldi      temp,0b11111111
000225 bb07          out      DDRB,temp
000226 9984          sbic     PIND,4
000227 cff5          rjmp     looper
000228 ef07          ldi      temp,0b11110111
000229 bb07          out      DDRB,temp          ; Set PORTB sebagai input.
00022a e30f          ldi      temp,0b00111111
00022b bb08          out      PORTB,temp          ; Set PORTB sebagai input
00022c cdf1          rjmp     sini
;
;Eksekusi1:
00022d d054          rcall     Ldelay
00022e 2d0d          mov      temp,temp2
00022f 300a          cpi      temp,0x0A
000230 f421          brne     bukan3
000231 ebe6          goo3:    ldi      z1,low(msgknanan1*2)
000232 e0f6          ldi      zh,high(msgknanan1*2)
000233 d038          rcall     kirim
000234 c00c          rjmp     terus1
000235 300b          bukan3:  cpi      temp,0x0B
000236 f409          brne     bukan4
000237 cff9          rjmp     goo3
000238 300c          bukan4:  cpi      temp,0x0C
000239 f421          brne     bukan5
00023a ece2          goo4:    ldi      z1,low(msgknanan2*2)
00023b e0f6          ldi      zh,high(msgknanan2*2)
00023c d02f          rcall     kirim
00023d c003          rjmp     terus1
00023e 300d          bukan5:  cpi      temp,0x0D
00023f f769          brne     Eksekusi1
000240 cff9          rjmp     goo4
000241 d036          terus1:  rcall     receive
000242 334e          cpi      Rxbyte,'>'
000243 f7e9          brne     terus1
000244 e4e0          ldi      z1,low(noSMSC*2)
000245 e0f5          ldi      zh,high(noSMSC*2)
000246 d025          rcall     kirim
000247 2d0e          mov      temp,jumlahnomor
000248 2fc2          mov      Y1,alamatnomor
;
;sampaino12:
000249 9159          ld       txbyte,Y+
00024a d029          rcall     transmit
00024b 950a          dec      temp
00024c 2300          tst      temp
00024d f7d9          brne     sampaino12
00024e ecee          ldi      z1,low(pesankanan1*2)
00024f e0f6          ldi      zh,high(pesankanan1*2)
000250 d01b          rcall     kirim
000251 2d5f          mov      txbyte,temp1
000252 d021          rcall     transmit
000253 2d5c          mov      txbyte,temp3
000254 d01f          rcall     transmit
000255 e0e2          ldi      z1,low(pesankanan2*2)
000256 e0f7          ldi      zh,high(pesankanan2*2)
000257 d014          rcall     kirim
000258 d029          rcall     Ldelay
000259 d028          rcall     Ldelay
00025a 9ac6          sbi      portB,6

```

```

                                septa.1st
00025b 98c7                cbi      portB,7
                                putarmotor:
00025c 9a94                sbi      PortD,4
00025d e802                ldi      temp,0b10000010
00025e bd0f                out      TCCR1A,temp
00025f e004                ldi      temp,0b00000100
000260 bd0e                out      TCCR1B,temp
                                ; Clear Timer/Counter1 on Compare Match
                                ; Set prescalar to 256
000261 bdab                out      OCR1AH,titip
000262 bdba                out      OCR1AL,titip1
000263 ef0f                ldi      temp,0b11111111
000264 bb07                out      DDRB,temp
000265 9984                sbic    PIND,4
000266 cff5                rjmp    putarmotor
000267 ef07                ldi      temp,0b11110111
000268 bb07                out      DDRB,temp
                                ; Set PORTB sebagai input
000269 e30f                ldi      temp,0b00111111
00026a bb08                out      PORTB,temp
                                ; Set PORTB sebagai input
00026b cdb2                rjmp    sini
                                ;
00026c 95c8                kirim: lpm
00026d 2000                tst      r0
00026e f021                breq    rampung
00026f 2d50                mov     txbyte,r0
000270 d003                rcall   transmit
000271 9631                adiw    z1,1
000272 cff9                rjmp    kirim
                                ;
                                rampung:
000273 9508                ret
                                ;
                                transmit:
000274 9b5d                sbis    USR,UDRE
                                ;is UART transmitter ready?
000275 cffe                rjmp    transmit
000276 b95c                out      UDR,txbyte
                                ;sent out char
000277 9508                ret
                                ;
                                receive:
000278 9b5f                sbis    USR,RXC
000279 cffe                rjmp    receive
00027a b14c                in      Rxbyte,UDR
00027b 9508                ret
                                ;
                                delay:
00027c e470                ldi      delay1,64
                                wait:
00027d 957a                dec     delay1
00027e f7f1                brne    wait
00027f 951a                dec     timeout
000280 f7d9                brne    delay
000281 9508                ret
                                ;
                                Ldelay:
000282 e268                ldi      delay3,40
                                waitmore2:
000283 d003                rcall   tunda
000284 956a                dec     delay3
000285 f7e9                brne    waitmore2
000286 9508                ret
                                ;
                                tunda:
000287 ef8f                ldi      delay2,0xff
                                waitsome:

```

```

                                septa.1st
000288 ef7f          ldi      delay1,0xff
                                waitmore:
000289 957a          dec      delay1
00028a f7f1          brne     waitmore
00028b 958a          dec      delay2
00028c f7d9          brne     waitsome
00028d 9508          ret
                                ;
00028e          msg:          .db      "AT+CMGL=0",0x0d,0
00028e 5441
00028f 432b
000290 474d
000291 3d4c
000292 0d30
000293 0000
000294          msg1:          .db      "AT+CMGS=73",0x0d,0
000294 5441
000295 432b
000296 474d
000297 3d53
000298 3337
000299 000d
00029a          msg2:          .db      "AT+CMGS=74",0x0d,0
00029a 5441
00029b 432b
00029c 474d
00029d 3d53
00029e 3437
00029f 000d
0002a0          noSMSC:          .db      "069126180100001100",0
0002a0 3630
0002a1 3139
0002a2 3632
0002a3 3831
0002a4 3130
0002a5 3030
0002a6 3030
0002a7 3131
0002a8 3030
0002a9 0000
0002aa          isisalah:          .db      "0000AA43CD70D80C32BFE5ED301D346D4E416137390C9A87"
0002aa 3030
0002ab 3030
0002ac 4141
0002ad 3334
0002ae 4443
0002af 3037
0002b0 3844
0002b1 4330
0002b2 3233
0002b3 4642
0002b4 3545
0002b5 4445
0002b6 3033
0002b7 4431
0002b8 3433
0002b9 4436
0002ba 4534
0002bb 3134
0002bc 3136
0002bd 3733
0002be 3933
0002bf 4330

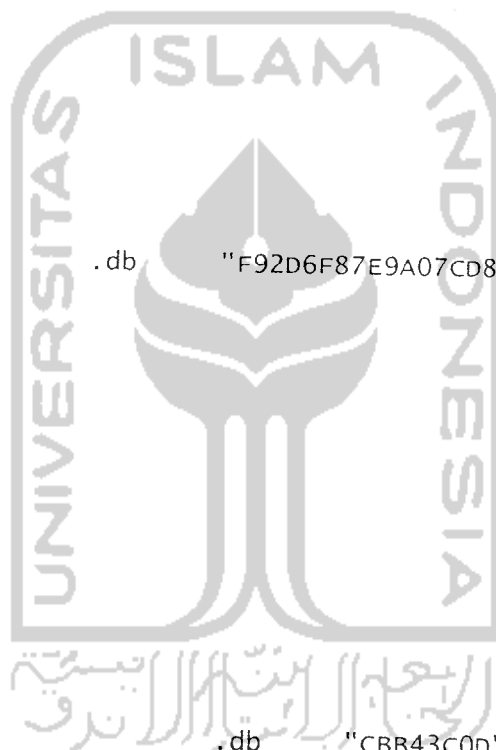
```

septa.1st

0002c0 4139
0002c1 3738

.db "D96134689E6687D1EBB01B546787DDE734885C769FC36E90"

0002c2 3944
0002c3 3136
0002c4 3433
0002c5 3836
0002c6 4539
0002c7 3636
0002c8 3738
0002c9 3144
0002ca 4245
0002cb 3042
0002cc 4231
0002cd 3435
0002ce 3736
0002cf 3738
0002d0 4444
0002d1 3745
0002d2 3433
0002d3 3838
0002d4 4335
0002d5 3637
0002d6 4639
0002d7 3343
0002d8 4536
0002d9 3039



.db "F92D6F87E9A07CD87D0689CBEEB01C", 26,0

0002da 3946
0002db 4432
0002dc 4636
0002dd 3738
0002de 3945
0002df 3041
0002e0 4337
0002e1 3844
0002e2 4437
0002e3 3630
0002e4 3938
0002e5 4243
0002e6 4545
0002e7 3042
0002e8 4331
0002e9 001a

0002ea ; Kiri:

0002ea 4243
0002eb 3442
0002ec 4333
0002ed 4430

0002ee Kiri100: .db "8AC160"

0002ee 4138
0002ef 3143
0002f0 3036

0002f1 Kiri200: .db "92C160"

0002f1 3239
0002f2 3143
0002f3 3036

0002f4 Kiri300: .db "9AC160"

0002f4 4139
0002f5 3143
0002f6 3036

0002f7 Kiri400: .db "A2C160"

septa.1st

0002f7	3241			
0002f8	3143			
0002f9	3036			
0002fa		Kiri500:	.db	"AAC160"
0002fa	4141			
0002fb	3143			
0002fc	3036			
0002fd		Kiri600:	.db	"B2C160"
0002fd	3242			
0002fe	3143			
0002ff	3036			
000300		Kiri700:	.db	"BAC160"
000300	4142			
000301	3143			
000302	3036			
000303		Kiri800:	.db	"C2C160"
000303	3243			
000304	3143			
000305	3036			
000306		Kiri900:	.db	"CAC160"
000306	4143			
000307	3143			
000308	3036			
	;			
000309		Kanan:	.db	"CBB03BEC06"
000309	4243			
00030a	3042			
00030b	4233			
00030c	4345			
00030d	3630			
00030e		Kanan100:	.db	"C56030"
00030e	3543			
00030f	3036			
000310	3033			
000311		Kanan200:	.db	"C96030"
000311	3943			
000312	3036			
000313	3033			
000314		Kanan300:	.db	"CD6030"
000314	4443			
000315	3036			
000316	3033			
000317		Kanan400:	.db	"D16030"
000317	3144			
000318	3036			
000319	3033			
00031a		Kanan500:	.db	"D56030"
00031a	3544			
00031b	3036			
00031c	3033			
00031d		Kanan600:	.db	"D96030"
00031d	3944			
00031e	3036			
00031f	3033			
000320		Kanan700:	.db	"DD6030"
000320	4444			
000321	3036			
000322	3033			
000323		Kanan800:	.db	"E16030"
000323	3145			
000324	3036			
000325	3033			
000326		Kanan900:	.db	"E56030"

septa.1st

000326 3545
000327 3036
000328 3033

000329 ; msgkiri1: .db "AT+CMGS=47",0x0d,0

000329 5441
00032a 432b
00032b 474d
00032c 3d53
00032d 3734
00032e 000d

00032f msgkiri2: .db "AT+CMGS=48",0x0d,0

00032f 5441
000330 432b
000331 474d
000332 3d53
000333 3834
000334 000d

000335 pesankiri1: .db
"0000AA25CD37FD2D07CDCBEBB03CEC3E83C46539BC4E0FCB41",0

000335 3030
000336 3030
000337 4141
000338 3532
000339 4443
00033a 3733
00033b 4446
00033c 4432
00033d 3730
00033e 4443
00033f 4243
000340 4245
000341 3042
000342 4333
000343 4345
000344 4533
000345 3338
000346 3443
000347 3536
000348 3933
000349 4342
00034a 4534
00034b 4630
00034c 4243
00034d 3134
00034e 0000

00034f pesankiri2: .db "184C0E6F83D6E5755A9E06",26,0

00034f 3831
000350 4334
000351 4530
000352 4636
000353 3338
000354 3644
000355 3545
000356 3537
000357 4135
000358 4539
000359 3630
00035a 001a

00035b msgkanan1: .db "AT+CMGS=48",0x0d,0

00035b 5441
00035c 432b



septa.1st

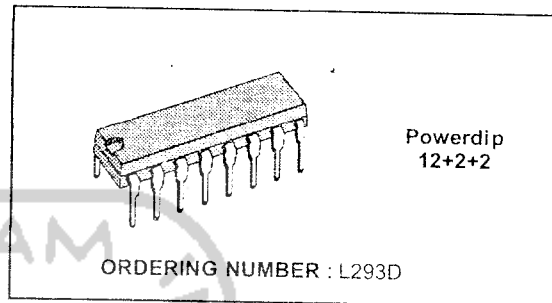
00035d 474d
00035e 3d53
00035f 3834
000360 000d
000361 msgkanaan2: .db "AT+CMGS=49",0x0d,0
000361 5441
000362 432b
000363 474d
000364 3d53
000365 3934
000366 000d
000367 pesankanan1: .db
"0000AA26CD37FD2D07CDCBEBB03CEC3E83C46539BC4E0FCB41",0
000367 3030
000368 3030
000369 4141
00036a 3632
00036b 4443
00036c 3733
00036d 4446
00036e 4432
00036f 3730
000370 4443
000371 4243
000372 4245
000373 3042
000374 4333
000375 4345
000376 4533
000377 3338
000378 3443
000379 3536
00037a 3933
00037b 4342
00037c 4534
00037d 4630
00037e 4243
00037f 3134
000380 0000
000381 pesankanan2: .db "184C0E6F83D6E575D81D7603",26,0000381 3831
000382 4334
000383 4530
000384 4636
000385 3338
000386 3644
000387 3545
000388 3537
000389 3844
00038a 4431
00038b 3637
00038c 3330
00038d 001a

Assembly complete with no errors.

PUSH-PULL FOUR CHANNEL DRIVER WITH DIODES

PRELIMINARY DATA

- 600mA. OUTPUT CURRENT CAPABILITY PER CHANNEL
- 1.2A PEAK OUTPUT CURRENT (NON REPETITIVE) PER CHANNEL
- ENABLE FACILITY
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5V (HIGH NOISE IMMUNITY)
- INTERNAL CLAMP DIODES



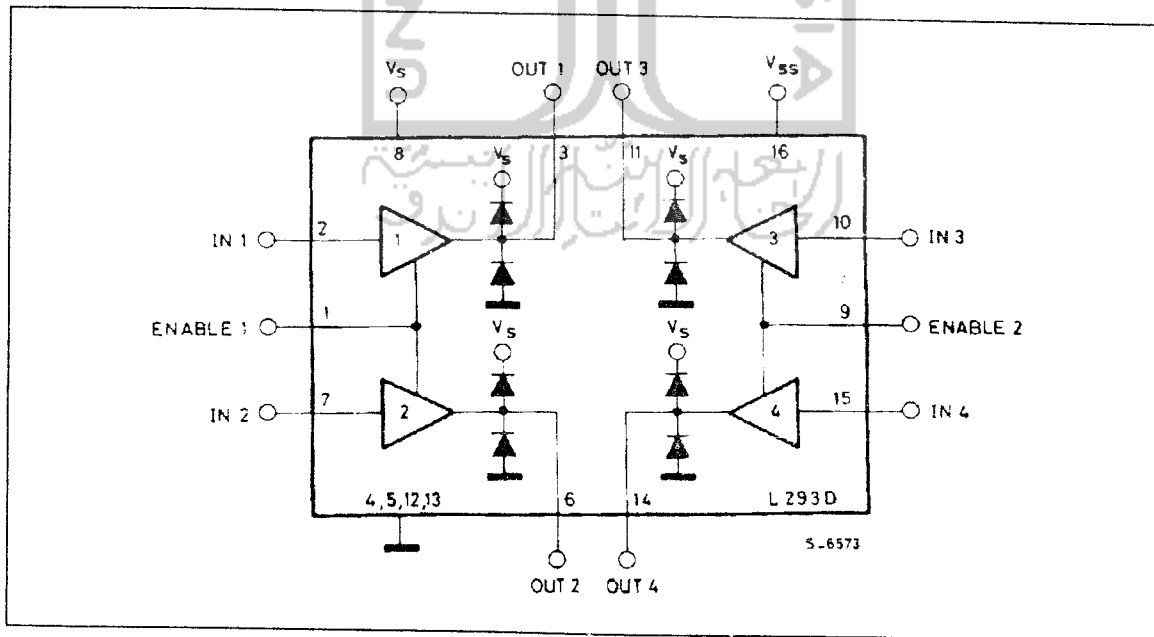
DESCRIPTION

The L293D is a monolithic integrated high voltage, high current four channel driver designed to accept standard DTL or TTL logic levels and drive inductive loads (such as relays solenoids, DC and stepping motors) and switching power transistors. To simplify use as two bridges is pair of channels is equipped with an enable input. A separate supply input is provided for the logic, allowing operation at a low voltage and internal clamp diodes are included.

This device is suitable for use in switching applications at frequencies up to 5 KHz.

The L293D is assembled in a 16 lead plastic package which has 4 center pins connected together and used for heatsinking.

BLOCK DIAGRAM

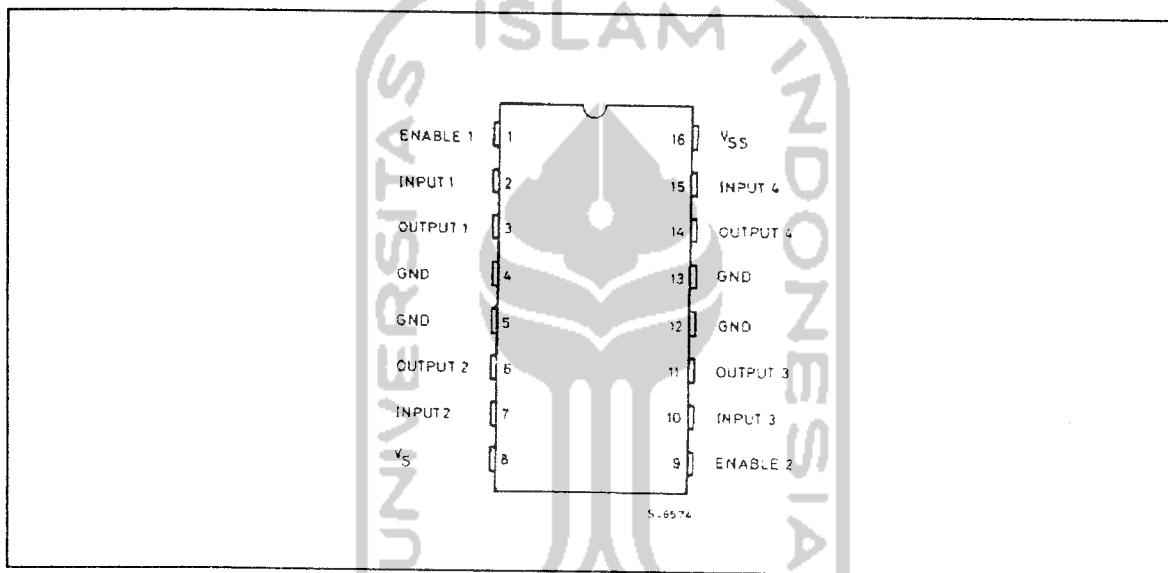


L293D

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_S	Supply Voltage	36	V
V_{SS}	Logic Supply voltage	36	V
V_i	Input voltage	7	V
V_{en}	Enable voltage	7	V
I_o	Peak output current (100 μ s non repetitive)	1.2	A
P_{tot}	Total power dissipation at $T_{ground-pins} = 80^{\circ}\text{C}$	5	W
T_{stg}, T_j	Storage and junction temperature	-40 to 150	$^{\circ}\text{C}$

CONNECTION DIAGRAM



THERMAL DATA

Symbol	Parameter	Value	Unit
$R_{th-j-case}$	Thermal resistance junction-case	max 14	$^{\circ}\text{C/W}$
$R_{th-j-amb}$	Thermal resistance junction-ambient	max 80	$^{\circ}\text{C/W}$

ELECTRICAL CHARACTERISTICS (For each channel, $V_s = 24V$, $V_{ss} = 5V$, $T_{amb} = 25^\circ C$, unless otherwise specified)

Symbol	Parameter	Test conditions	Min.	Typ.	Max.	Unit
V_s	Supply voltage		V_{ss}		36	V
V_{ss}	Logic supply voltage (pin 16)		4.5		36	V
I_s	Total quiescent supply current (pin 8)	$V_i = L \quad I_o = 0 \quad V_{en} = H$		2	6	mA
		$V_i = H \quad I_o = 0 \quad V_{en} = H$		16	24	
		$V_{en} = L$			4	
I_{ss}	Total quiescent logic supply current (pin 16)	$V_i = L \quad I_o = 0 \quad V_{en} = H$		44	60	mA
		$V_i = H \quad I_o = 0 \quad V_{en} = H$		16	22	
		$V_{en} = L$		16	24	
V_{iL}	Input low voltage (pin 2, 7, 10, 15)		-0.3		1.5	V
V_{iH}	Input high voltage (pin 2, 7, 10, 15)	$V_{ss} \leq 7V$	2.3		V_{ss}	V
		$V_{ss} > 7V$	2.3		7	
I_{iL}	Low voltage input current (pin 2, 7, 10, 15)	$V_{iL} = 1.5V$			-10	μA
I_{iH}	High voltage input current (pin 2, 7, 10, 15)	$2.3 \leq V_{iH} \leq V_{ss} - 0.6V$		30	100	μA
V_{enL}	Enable low voltage (pin 1, 9)		-0.3		1.5	V
V_{enH}	Enable high voltage (pin 1, 9)	$V_{ss} \leq 7V$	2.3		V_{ss}	V
		$V_{ss} > 7V$	2.3		7	
I_{enL}	Low voltage enable current (pin 1, 9)	$V_{enL} = 1.5V$		-30	-100	μA
I_{enH}	High voltage enable current (pin 1, 9)	$2.3V \leq V_{enH} \leq V_{ss} - 0.6V$			± 10	μA
V_{CEsatH}	Source output saturation voltage (pin 3, 6, 11, 14)	$I_o = -0.6A$		1.4	1.8	V
V_{CEsatL}	Sink output saturation voltage (pins 3, 6, 11, 14)	$I_o = +0.6A$			1.2	1.8
V_F	Clamp diode forward voltage	$I_o = 600 mA$		1.3		V
t_r	Rise time (*)	0.1 to 0.9 V_o		250		ns
t_f	Fall time (*)	0.9 to 0.1 V_o		250		ns
t_{on}	Turn-on delay (*)	0.5 V_i to 0.5 V_o		750		ns
t_{off}	Turn-off delay (*)	0.5 V_i to 0.5 V_o		200		ns

(*) See fig.1

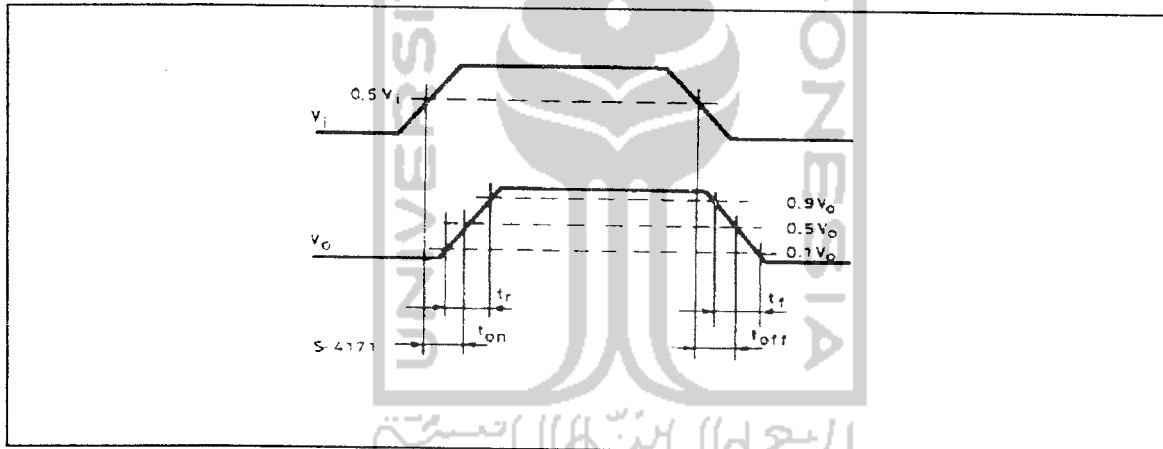
TRUTH TABLE (One channel)

INPUT	ENABLE (*)	OUTPUT
H	H	H
L	H	L
H	L	Z
L	L	Z

Z = High output impedance

(*) Relative to the considered channel

Figure 1. Switching Times



1. Software Interface

1.1. Overview of the Supported AT Command Set

Page	Commands 07.07	Function
7	AT+CGMI	Issue manufacturer ID code
7	AT+CGMM	Issue model ID code
7	AT+CGMR	Output the GSM telephone version
8	AT+CGSN	Output the serial number (IMEI)
8	AT+GSN	Output the serial number (IMEI)
8	AT+CHUP	Terminate call
8	AT+CEER	Query the reason for disconnection of last call
9	AT+CREG	Network registration
9	AT+COPS	Commands concerning selection of network operator
10	AT+CLCK	Switch locks on and off
10	AT+CPWD	Change password to a lock
11	AT+CLIP	Display telephone number of calling party
11	AT+CCFC	Call forwarding
12	AT+CHLD	Call hold and multiparty
12	AT+CPAS	Query the telephone status
13	AT+CPIN	Enter PIN and query lock
13	AT+CBC	Battery charge
14	AT+CSQ	Output signal quality
14	AT+CPBS	Select a telephone book
15	Fehler! Kein gültiges Resultat für Tabelle.	Read a telephone-book entry
15	AT+CPBW	Write a telephone-book entry
16	AT+CMEE	Expanded error messages according to GSM 07.07
17	AT+VTS	Send a DTMF tone
18	AT+VTD	Set duration of a DTMF tone
18	AT+WS46	Select wireless network
18	AT+CSCS	Select TE character set
19	AT+CAOC	Advice of charge
19	AT+CSSN	Supplementary service notifications
20	AT+CRSM	Restricted SIM access
20	AT+CIMI	Output of IMSI
21	AT+CACM	Accumulated call meter
21	AT+CAMM	Accumulated call meter maximum
22	AT+CLCC	List Current Calls
23	AT+CCLK	Clock
23	AT+COPN	Read operator names
23	AT+CPUC	Price per unit and currency table
24	AT+CALM	Alert sound mode
24	AT+CRSL	Ringer sound level
24	AT+CLVL	Loudspeaker volume level
24	AT+CMUT	Mute control
25	AT+CVIB	Vibrator mode

Page	Commands 07.05	Function
26	AT+CSMS	Selection of message service
27	AT+CPMS	Selection of SMS memory
27	AT+CMGF	SMS format
28	AT+CSCA	Address of the SMS service center
28	AT+CNMI	Display new incoming SMS
29	AT+CNMA	Acknowledgment of a short message directly output
30	AT+CMGL	List SMS
31	AT+CMGR	Read in an SMS
31	AT+CMGS	Send an SMS
31	AT+CMSS	Send an SMS from the SMS memory
32	AT+CMGW	Write an SMS to the SMS memory
32	AT+CMGD	Delete an SMS in the SMS memory
32	AT+CSCB	Select cell broadcast messages
32	AT+CMGC	Send an SMS command

Page	Siemens- specific commands	Function
33	AT^SPBS	Select a telephone book (including Siemens-specific books)
33	AT^SDLD	Delete the "last number redial" memory"
34	AT^SPBC	Seek the first entry in the sorted telephone book which begins with the selected (or next available) letter
34	AT^SPBG	Read entry from the sorted telephone book via the sorted index
35	AT^SLOCK	Switch locks (including user-defined locks) on and off
35	AT^SPWD	Change password to a lock (including user-defined locks)
36	AT^SACM	Output ACM (accumulated call meter) and ACMmax
36	AT^SPLM	Read the PLMN
36	AT^SPLR	Read an entry from the preferred-operator list
36	AT^SPLW	Write an entry to the preferred-operator list
37	AT^SCNI	Output call number information
37	AT^SNFV	Set the volume
37	AT^SNFS	Select NF hardware
38	AT^SRTC	Set the ringing tone
38	AT^SCID	Output card ID
38	AT^SCKS	Output SIM card status
39	AT^SPIC	Output PIN counter
39	AT^SMGO	SMS overflow indicator
40	AT^SMGL	List SMS (without status change from <i>unread</i> to <i>read</i>)
40	AT^SMGR	Read SMS record without Changing unread->read
40	AT^SMO	Switch device off
41	AT^SLNG	Language settings
41	AT^SSTK	SIM Toolkit
41	AT^SBNW	Binary Write
43	AT^SBNR	Binary Read

AT+CMGL	List SMS Revision according to GSM 07.05 Version 4.7.0										
Test command AT+CMGL=?	<p>Response +CMGL: (list of supported <stat>s)</p> <p>Parameter <stat></p> <table> <tr><td>0</td><td>"REC UNREAD": received unread messages (default)</td></tr> <tr><td>1</td><td>"REC READ": received read messages</td></tr> <tr><td>2</td><td>"STO UNSENT": stored unsent messages</td></tr> <tr><td>3</td><td>"STO SENT": stored sent messages</td></tr> <tr><td>4</td><td>"ALL": all messages</td></tr> </table>	0	"REC UNREAD": received unread messages (default)	1	"REC READ": received read messages	2	"STO UNSENT": stored unsent messages	3	"STO SENT": stored sent messages	4	"ALL": all messages
0	"REC UNREAD": received unread messages (default)										
1	"REC READ": received read messages										
2	"STO UNSENT": stored unsent messages										
3	"STO SENT": stored sent messages										
4	"ALL": all messages										
Write command AT+CMGL [=<stat>]	<p>Parameter <stat> See Test command</p> <p>Response If PDU mode (+CMGF=0) and command are successful: +CMGL:<index>,<stat>,<[alpha]>,<length> <CR><LF><pdu><CR><LF> +CMGL: <index>,<stat>,<[alpha]>,<length> <CR><LF><pdu><CR><LF> [...]]</p>										
	<p>Parameter <pdu> The PDU begins with the service-center address (according to GSM04.11), followed by the TPDU according to GSM03.40 in hexadecimal format otherwise: +CMS ERROR: <err></p>										

AT+CMGR	Read in an SMS Revision according to GSM 07.05 Version 4.7.0
Test command AT+CMGR=?	Response OK
Write command AT+CMGR= <index>	Parameter <index> Index of message in selected memory <mem1> Response If PDU mode (+CMGF=0) and command are successful: +CMGR: <stat>,[<alpha>],<length><CR><LF><pdu> Parameter <pdu> Siehe "AT+CMGL" otherwise: +CMS ERROR: <err>

AT+CMGS	Send an SMS
Test command AT+CMGS=?	Response OK
Write command If PDU mode (+CMGF=0) +CMGS=<length><CR>PDU is given <ctrl-Z/ESC>	Parameter <length> Length of PDU <pdu> See "AT+CMGL" <mr> Message reference Response If sending is successful: +CMGS: <mr> If sending is not successful: +CMS ERROR: <err>

AT+CMSS	Send an SMS from the SMS memory
Test command AT+CMSS=?	Response OK
Write command +CMSS=<index>[,<da>[,<toda>]]	Parameter <index> Index of message in selected memory <mem1> <da> Destination address in string format <toda> Format of destination address <mr> Message reference Response If sending is successful: +CMSS: <mr> If sending is not successful: +CMS ERROR: <err>

