



TUGAS AKHIR - TE095561

SISTEM PEMBERIAN PAKAN PADA KANDANG KUCING VIA SMS

Alfi Bahar
NRP 2211039046

Dosen Pembimbing
Rudy Dikairono, ST., MT.
Eko Pujatno Matni, S.pd.

PROGRAM STUDI D3 TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015



TUGAS AKHIR - TE095561

SISTEM PEMBERIAN PAKAN PADA KANDANG KUCING VIA SMS

Alfi Bahar
NRP 2211039046

Dosen Pembimbing
Rudy Dikairono, ST., MT.
Eko Pujiatno Matni, S.pd.

PROGRAM STUDI D3 TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015



FINAL PROJECT - TE095561

CAT'S FOODER WITH SMS SYSTEM

Alfi Bahar
NRP 2211039046

Counsellor
Rudy Dikairono, ST., MT.
Eko Pujiatno Matni, Spd.

*Electrical Engineering D3 Program
Industrial Technology Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2015*

**SISTEM PEMBERIAN PAKAN PADA KANDANG KUCING
VIA SMS**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya
Pada
Bidang Studi Elektro Industri
Program Studi D3 Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember**

**Menyetujui :
Dosen Pembimbing,**



Rudy Dikairono, ST., MT.
NIP. 198103252005011002

**SISTEM PEMBERIAN PAKAN PADA KANDANG KUCING VIA
SMS**

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Memperoleh Gelar Ahli Madya Teknik
Pada
Bidang Studi Elektro Industri
Program Studi D3 Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember

Menyetujui :
Dosen Pembimbing,

Eko Pujiatno Matni, S.pd.
NIP. 197103301994031002

**SURABAYA
JUNI, 2015**

ABSTRAK

SISTEM PEMBERIAN PAKAN PADA KANDANG KUCING VIA SMS

Nama Mahasiswa :

1. Alfi Bahar

NRP. 2211039046

Dosen Pembimbing :

1. Rudy Dikairono, ST., MT.

NIP. 198103252005011002

2. Eko Pujiatno Matni, S.pd.

NIP. 197103301994031002

ABSTRAK

Pada masa kini, teknologi pemberian pakan hewan masih tergolong tradisional. Biasanya para pemilik hewan hanya memberikan langsung makanannya kepada hewan peliharaannya tersebut secara manual. Hal ini tentu saja sangat menyulitkan bagi mereka yang memiliki mobilitas tinggi seperti mereka yang harus bekerja, kuliah, sekolah, atau bahkan yang harus terpaksa bepergian dalam waktu yang cukup lama. Belum lagi jika hewan peliharaan tersebut mengalami sakit dan berpotensi menular pada manusia. Oleh karena itu dengan adanya alat pemberi pakan otomatis tentunya akan sangat meringankan para pemilik dalam merawat hewan peliharaannya.

Alat ini menggunakan *microcontroller* ATmega16 dalam perancangan sistemnya. Masukan pada sistem alat ini terdiri dari *modem wavecom* dan *adaptor* 12Volt sebagai sumber tegangan. Sedangkan keluarannya terdiri dari rangkaian *relay* dengan 2 *relay* yang dihubungkan pada *motor* dc yang berfungsi membuka dan menutup pintu penyimpanan pakan.

Dengan penggunaan sms akan mempermudah penggunaan alat ini, yang sebelumnya pemberian pakan dilakukan secara *manual* diganti dengan cara otomatis. Dan tidak membutuhkan waktu lama dengan penggunaan sms yang rata-rata hanya membutuhkan waktu 2 detik dengan tingkat keberhasilan 90% dengan menggunakan waktu *delay* 1 detik, 2 detik atau 3 detik sesuai dengan perintah sms dan *software* yang dimasukkan.

Kata Kunci : *Micocontroller ATmega16, Modem wavecom, Adaptor. Motor dc.*

ABSTRACT

CAT'S FOODER WITH SMS SYSTEM

ABSTRACT

Student's Name : Alfi Bahar
NRP : 2211039046
Supervisor 1 : Rudy Dikairono,ST.,MT.
NIP : 198103252005011002
Supervisor 2 : Eko Pujiatno Matni, Spd.
NIP : 197103301994031002

At this time, animal fooder's technology still using a traditional technology. Usually people just give the food directly using manual prosedure. This way absolutely so hard to do for those who have a very high mobility. Such as who in study, must go for his jobs, and for someone that must go away for a very long time. And the animal that sick absolutely must be canrantine. So thist project (Cat's fooder with SMS system) can really help the owner that have a beloved pet.

This project using ATmega16 microcontroller's at it systems. An input for this system comes from a Wavecom modem and a 12V adaptor as a voltage supply. On the other hands, output build by 2channels of relay that connectly to DC motors that funtionally for close-open the foods gate.

Using SMS system will really help the owner for use this things, wich on the older time people just give the food directly, now replaced by an otomatically system. And using DC motors for close-open will very evesiensically and cut the time so the ownwer just need 2 second with 90% of succees. And with an optional program owner gived different delay times 1 second. 2, or 3 seconds. As same as program that saved on the microcontroller.

Keywords: *ATmega16 Micocontroller, Wavecom Modem, Adaptor. Motor dc.*

KATA PENGANTAR

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadirat Allah SWT yang telah melimpahkan rahmat, hidayah serta karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul :

“SISTEM PEMBERIAN PAKAN PADA KANDANG KUCING VIA SMS”

Tugas Akhir ini merupakan sebagian syarat untuk menyelesaikan mata kuliah dan memperoleh nilai pada Tugas Akhir.

Dengan selesainya Tugas Akhir ini penulis menyampaikan terima kasih sebesar-besarnya kepada:

1. Orang Tua atas limpahan doa, kasih sayang, dukungan dan dorongan baik berupa moril atau materil bagi penulis.
2. Bapak Eko Setijadi, ST. MT. Ph.D selaku Ketua Program D3 Teknik Elektro Bidang Studi Teknik Listrik, FTI-ITS Surabaya.
3. Bapak Rudy Dikairono, ST., MT. dan Bapak Eko Pujiatno Matni, S.pd. selaku dosen pembimbing.
4. Seluruh staf pengajar dan administrasi Prodi D3 Teknik Elektro FTI-ITS.
5. Seluruh mahasiswa D3 Teknik Elektro khususnya angkatan 2011.
6. Semua pihak yang telah banyak membantu untuk menyelesaikan Tugas Akhir ini yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari masih banyak kekurangan dalam Tugas Akhir ini. Kritik dan saran untuk perbaikan tugas ini sangat diperlukan. Akhir kata semoga tugas ini dapat bermanfaat bagi kita semua.

Surabaya, Juni 2015

Penulis

DAFTAR ISI

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	iii
ABSTRAK	vii
<i>ABSTRACT</i>	ix
KATA PENGANTAR.....	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii

BAB I PENDAHULUAN

1.1 Latar Belakang	1
1.2 Permasalahan.....	1
1.3 Batasan Masalah	1
1.4 Tujuan	2
1.5 Sistematika Laporan.....	2
1.6 Relevansi	3

BAB II TEORI PENUNJANG

2.1. Mikrokontroler AVR ATMEGA16.....	5
2.1.1 Konfigurasi Pin AVR ATMEGA16	7
2.2 <i>Driver Motor</i>	9
2.3 <i>Modem Wavecom</i>	10
2.4 LCD 2x16	12
2.5 <i>Motor DC Gearbox</i>	15
2.7 <i>Adaptor DC 12V</i>	16

BAB. III PERANCANGAN DAN PEMBUATAN ALAT

3.1 Deskripsi Kerja Alat	19
3.2 Perancangan Perangkat keras (<i>Hardware</i>).....	19
3.2.1 Perancangan Rangkaian <i>Relay</i>	20
3.2.2 Modul <i>Microcontroller</i> ATMega16	22
3.3 Perancangan <i>Software</i>	22
3.4 Perancangan Mekanik.....	24

BAB. IV PENGUJIAN DAN ANALISA DATA

4.1 Pengujian Perangkat Keras	27
4.1.1 Pengujian Pada <i>Microcontroller</i> ATMega16	27
4.1.2 Pengujian <i>Adaptor</i>	28

4.1.3	Pengujian Rangkaian <i>Relay</i>	29
4.2	Pengujian Perangkat Lunak (<i>Software</i>)	30
4.2.1	Pengujian Program.....	30
4.3	Pengujian Sistem Keseluruhan	33

BAB. V PENUTUP

5.1.	Kesimpulan	37
5.2.	Saran	37

DAFTAR PUSTAKA	39
LAMPIRAN 1 LISTING PROGRAM	A-1
LAMPIRAN 2 DATASHEET	B-1
DAFTAR RIWAYAT HIDUP	C-1

DAFTAR GAMBAR

DAFTAR GAMBAR

Gambar 2.1	ATMega16.....	5
Gambar 2.2	Konfigurasi Kaki (pin) ATMEGA16.....	7
Gambar 2.3	<i>Relay 2 Channel</i>	9
Gambar 2.4	<i>Modem Wavecom</i>	10
Gambar 2.5	Konfigurasi Pin <i>Modem Wavecom</i>	12
Gambar 2.6	Konfigurasi <i>LCD</i>	13
Gambar 2.7	<i>LCD 2x16</i>	14
Gambar 2.8	<i>Motor DC Dengan Gearbox</i>	15
Gambar 2.9	<i>Adaptor DC 12V</i>	16
Gambar 2.10	Rangkaian <i>Adaptor DC 12V</i>	18
Gambar 3.1	Diagram Alat Secara Keseluruhan.....	19
Gambar 3.2	Posisi Peletakan <i>Relay</i> dibawah <i>Modem Wavecom</i>	21
Gambar 3.3	Pemberian <i>Input Power</i>	21
Gambar 3.4	Modul ATMega16	22
Gambar 3.5	Perancangan <i>Software</i>	23
Gambar 3.6	Penginputan <i>Software</i> pada <i>Microcontroller</i>	23
Gambar 3.7	Mekanik Pintu Pakan.....	24
Gambar 3.8	Rancangan Kotak Penyimpanan <i>Microcontroller</i> , <i>Driver Motor</i> , dan <i>Modem Wavecom</i>	25
Gambar 3.9	Perancangan Kotak Penyimpanan Pakan.....	26
Gambar 4.1	Bentuk Fisik <i>Adaptor DC 12V</i>	29
Gambar 4.2	Bentuk Fisik Rangkaian <i>Relay</i>	30
Gambar 4.3	Penempatan <i>Motor DC</i> Sebagai Pembuka Pintu Pakan.....	34
Gambar 4.4	Bentuk Perancangan Peralatan Elektronik.....	34
Gambar 4.5	Penempatan Peralatan Elektronik pada <i>Cover</i>	35
Gambar 4.6	<i>Relay Motor</i>	35
Gambar 4.7	Penempatan Alat pada Kandang	36

DAFTAR LABEL

DAFTAR TABEL

Tabel 2.1	Fungsi Khusus <i>Port B</i>	8
Tabel 4.1	Pengujian Adaptor.....	28
Tabel 4.2	Pengujian Rangkaian Relay	29
Tabel 4.3	Pengujian dari program P1 dengan <i>Delay</i> 1000ms	31
Tabel 4.4	Pengujian dari Program P2 dengan <i>Delay</i> 2000ms	32
Tabel 4.5	Pengujian dari Program P3 dengan <i>Delay</i> 3000ms	32

BAB I
PENDAHULUAN

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Salah satu kebutuhan utama pada hewan adalah makanan. Terutama pada hewan peliharaan yang tidak dapat mencari makan secara mandiri. Seperti kucing peliharaan jenis tertentu. Maka dibutuhkan perhatian yang secara terus menerus dari pemilik kucing tersebut. Sehingga terkadang terasa sangat menyita waktu bagi pemilik kucing tersebut, terutama bagi pemilik dengan jam kerja panjang atau bagi pemilik yang akan melakukan perjalanan jauh dan tidak dapat membawa kucing peliharaannya. Solusi selama ini adalah dengan menitipkan kucing peliharaan tersebut kepada *pet shop*, tetapi tentu saja dengan biaya yang tidak sedikit. Selain itu terdapat pula resiko penularan penyakit dari kucing lain yang berdekatan kandangnya, kucing pun menjadi tidak akrab lagi dengan pemilik setelah pulang dari penitipan tersebut, sehingga bagi pemilik kucing yang akan melakukan perjalanan jauh akan sangat kerepotan untuk memberi pakan kucing kesayangannya tersebut. Sementara bagi hewan seperti kucing, makanan tidak dapat diberikan secara langsung dengan jumlah yang banyak, ini adalah yang dilakukan banyak orang yang akan bepergian jauh dan meninggalkan kucing peliharaan mereka sendirian, padahal makan kucing yang bersifat kering sebaiknya disimpan ditempat yang tertutup dan tidak terkena udara bebas dan kucing hanya diberikan makanan sesuai porsi untuk dihabiskan saat itu juga.

1.2 Permasalahan

Melihat latar belakang di atas maka dirumuskan permasalahan sebagai berikut:

- ▶ Bagaimana merancang sebuah sistem pemberian pakan hewan menggunakan sms?

1.3 Batasan Masalah

Agar tugas akhir tidak menyimpang dari ketentuan yang digariskan maka diambil batasan dengan asumsi sebagai berikut:

- ▶ Tugas akhir ini hanya purwarupa karena saya hanya merancang sistem yang dapat dikembangkan dalam skala besar.

1.4 Tujuan

- ▶ Merancang sistem pemberian pakan pada kandang kucing menggunakan sms.
- ▶ Merancang *hardware* berupa kandang kucing dan alat pemberian pakan.

1.5 Sistematika Laporan

Dalam penyusunan buku Tugas Akhir ini, pembahasan mengenai sistem alat yang dibuat dibagi menjadi lima bab dengan sistematika sebagai berikut:

BAB I : PENDAHULUAN

Dalam bab ini mendiskripsikan tentang latar belakang, permasalahan, batasan masalah, tujuan, ruang lingkup, sistematika laporan, serta relevansi penulisan Tugas Akhir ini.

BAB II : TEORI PENUNJANG

Dalam bab ini berisi penjelasan dasar teori mengenai konsep yang digunakan dalam perancangan Tugas Akhir ini, meliputi pembahasan tentang ATmega16, *Modem Wavcom*, dan *Motor DC*.

BAB III: PERANCANGAN DAN PEMBUATAN ALAT

Dalam bab ini akan dibahas secara detail tentang rangkaian-rangkaian yang digunakan serta perangkat lunak berupa program yang membantu pengaktifan alat tersebut.

BAB IV : PENGUJIAN DAN ANALISA DATA

Dalam bab ini berisi data-data pengukuran dan pengujian beserta analisa terhadap prinsip kerja dari alat yang dibuat meliputi pengujian mekanik *hardware* (elektronik), *software* dan pengujian keseluruhan.

BAB V : PENUTUP

Dalam bab ini berisi kesimpulan yang diperoleh dari pembuatan Tugas Akhir ini dan saran-saran untuk pengembangan lebih lanjut.

1.6 Relevansi

Manfaat untuk masyarakat (para pemilik hewan peliharaan):
Membantu menyumbangkan sebuah gagasan baru berupa sistem pemberian pakan kucing sehingga para pemilik hewan dapat lebih mudah dalam perawatan terutama pada pemberian pakan untuk hewan mereka.

Halaman ini Sengaja dikosongkan

BAB II
TEORI PENUNJANG

BAB II TEORI PENUNJANG

Pada bab ini menjelaskan mengenai teori-teori pendukung yang berkaitan dengan topik alat yang akan dibuat meliputi ATmega16, *driver motor*, *modem wavecom*, *lcd 2x16*, *motor dc gearbox*, dan *adaptor dc 12V*.

2.1 Mikrokontroler AVR ATMEGA16

Mikrokontroler ATmega16 Mikrokontroler adalah sebuah sistem komputer lengkap dalam satu *chip*. *Microcontroller* lebih dari sekedar sebuah mikroprosesor karena sudah terdapat atau berisikan ROM (*Read-Only Memory*), RAM (*Read-Write Memory*), beberapa masukan maupun keluaran, dan beberapa peripheral seperti pencacah/pewaktu, *ADC (Analog to Digital converter)*, *DAC (Digital to Analog converter)* dan serial komunikasi. Salah satu mikrokontroler yang banyak digunakan saat ini yaitu mikrokontroler AVR. AVR adalah mikrokontroler RISC (*Reduce Instruction Set Computer*) 8 bit berdasarkan arsitektur Harvard. Secara umum mikrokontroler AVR dapat dikelompokkan menjadi 3 kelompok, yaitu keluarga AT90Sxx, ATmega dan ATtiny. Pada dasarnya yang membedakan masing-masing kelas adalah memori, *peripheral*, dan fiturnya. Seperti mikroprosesor pada umumnya, secara internal mikrokontroler ATmega16 terdiri atas unit-unit fungsionalnya *Arithmetic and Logical Unit (ALU)*, himpunan register kerja, *register* dan *decoder* instruksi, dan pewaktu beserta komponen kendali lainnya. Berbeda dengan mikroprosesor, mikrokontroler menyediakan memori dalam prosesornya (*in chip*).



Gambar 2.1 ATmega16

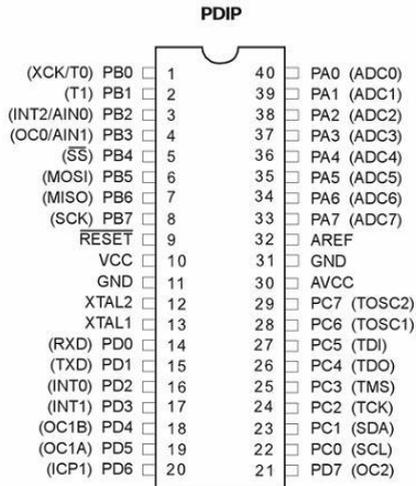
AVR merupakan seri mikrokontroler *Complementary Metal Oxide Semiconductor* (CMOS) 8-bit buatan Atmel berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Hampir semua instruksi pada program dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register *general-purpose*, *timer/counter* fleksibel dengan mode *compare*, interupsi *internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, *power saving mode*, ADC dan PWM. AVR pun mempunyai *In-System Programmable (ISP) Flash on-chip* yang memungkinkan memori program untuk diprogram ulang (*read/write*) dengan koneksi secara serial yang disebut *Serial Peripheral Interface (SPI)*.

AVR memiliki keunggulan dibandingkan dengan mikrokontroler lain, keunggulan mikrokontroler AVR yaitu memiliki kecepatan dalam mengeksekusi program yang lebih cepat, karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock* (lebih cepat dibandingkan mikrokontroler keluarga MCS 51 yang memiliki arsitektur *Complex Intrukstion Set Compute*). ATMEGA16 mempunyai *throughput* mendekati 1 *Millions Instruction Per Second (MIPS)* per MHz, sehingga membuat konsumsi daya menjadi rendah terhadap kecepatan proses eksekusi perintah. Beberapa keistimewaan dari AVR ATMEGA16 antara lain:

1. Mikrokontroler AVR 8 bit yang memiliki kemampuan tinggi dengan konsumsi daya rendah.
2. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16MHz.
3. Memiliki kapasitas *Flash* memori 16 Kbyte, EEPROM 512 Byte dan SRAM 1 Kbyte.
4. Saluran I/O sebanyak 32 buah, yaitu *Port A*, *Port B*, *Port C* dan *Port D*.
5. CPU yang terdiri dari 32 buah *register*.
6. Unit interupsi dan *eksternal*.
7. *Port* USART untuk komunikasi serial.
8. Fitur *peripheral*
 - Tiga buah *Timer/Counter* dengan kemampuan perbandingan (*compare*).
 - Dua buah *Timer/Counter* 8 bit dengan *Prescaler* terpisah dan *ModeCompar*.

- Satu buah *Timer/Counter* 16 bit dengan *Prescaler* terpisah, *Mode Compare* dan *Mode Capture*.
 - Real Time Counter* dengan *Oscillator* tersendiri.
 - Empat kanal PWM.
 - 8 kanal ADC.
 - 8 *Single-ended Channel* dengan keluaran hasil konversi 8 dan 10 resolusi (register ADCH dan ADCL).
 - 7 *Diferrential Channel* hanya pada kemasan *Thin Quad Flat Pack* (TQFP).
 - 2 *Diferrential Channel* dengan *Programmable Gain*.
 - Antarmuka *Serial Peripheral Interface* (SPI) Bus.
 - Watchdog Timer* dengan *Oscillator Internal* 9.
 - On-chip Analog Comparator*.
9. *Non-volatile program memory*.

2.1.1 Konfigurasi *Pin* AVR ATMEGA16



Gambar 2.2 Konfigurasi Kaki (*pin*) ATMEGA16

Konfigurasi *pin* ATMEGA16 dengan kemasan 40 *pin Dual In-line Package* (DIP) dapat dilihat pada Gambar 2.13. dari gambar diatas dapat dijelaskan fungsi dari masing-masing *pin* ATMEGA16 sebagai berikut.

1. VCC merupakan *pin* yang berfungsi sebagai masukan catu daya.
2. GND merupakan *pin Ground*.
3. Port A (PA0 – PA7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* masukan ADC 10.
4. Port B (PB0 – PB7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus, seperti dapat dilihat pada tabel dibawah ini.

Tabel 2.1 Fungsi Khusus Port B

Pin	Pin Fungsi Khusus
PB0	XCK (USART External Clock Input/Output) T0 (Timer/Counter0 External Counter Input)
PB1	T1 (Timer/Counter1 External Counter Input)
PB2	INT2 (External Interupt 2 Input) AIN0 (Analaog Comparator Negative Input)
PB3	OC0 (Timer/Counter0 Output Compare Macth Output) AIN1 (Analaog Comparator Negative Input)
PB4	(SPI Slave Select Input)
PB5	MOSI (SPI Bus Master Output /Slave Input)
PB6	MISO (SPI Bus Master Input/Slave Output)
PB7	SCK (SPI Bus Serial Clock)

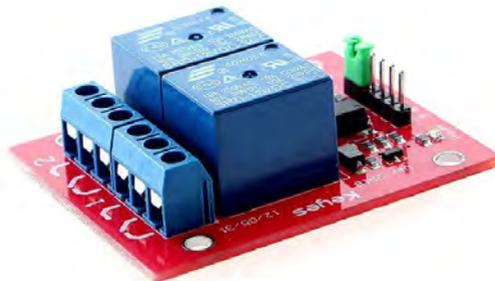
5. Port A (PC0 – PC7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus,
PC0 - SCL (*Two-wire Serial Bus Clock Line*)
PC1 - SDA (*Two-wire Serial BusData Input/Output Line*)
PC2 - TCK (*Joint Test Action Group Test Clock*)
PC3 - TMS (*JTAG Test Mode Select*)
PC4 - TDO (*JTAG Data Out*)
PC5 - TDI (*JTAG Test Data In*)
PC6 - TOSC1 (*Timer Oscillator pin 1*)
PC7 - TOSC2 (*Timer Oscillator pin 2*)
6. Port D (PD0 – PD7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus.
PD0 - RXD (USART Input Pin)

PD1 - TXD (USART Output Pin)
PD2 - INT0 (External Interrupt 0 Input)
PD3 - INT1 (External Interrupt 1 Input)
PD4 - OC1B (Timer/Counter1 Output Compare B Match Output)
PD5 - OC1A (Timer/Counter1 Output Compare A Match Output)
PD6 - ICP (Timer/Counter1 Input Capture Pin)
PD7 - OC2 (Timer/Counter2 Output Compare Match Output)

7. *RESET* merupakan *pin* yang digunakan untuk me-reset mikrokontroler.
8. XTAL1 dan XTAL2, merupakan *pin masukan external clock*.
9. AVCC merupakan *pin masukan tegangan* untuk ADC.
10. AREF merupakan *pin masukan tegangan referensi* untuk ADC.

2.2 Driver Motor

Pada alat ini menggunakan driver motor *relay 2 channel* Rangkaian *relay* pada sistem perancangan alat kali ini digunakan sebagai komponen keluaran dari sistem. *Relay* disini akan aktif jika sms telah sesuai dengan format yang ditentukan sesuai dengan sampel yang disimpan di memori *database*. *Relay* ini dihubungkan dari *microcontroller* menuju motor. Rangkaian ini menggunakan dua buah *relay*.



Gambar 2.3 *Relay 2 Channel*

Relay adalah rangkaian elektronika yang dapat digunakan untuk mengendalikan sesuatu dari jarak jauh. *Relay* sendiri merupakan

saklar magnetis yang paling sering digunakan pada setiap rangkaian elektronika. Dalam dunia elektronika, *relay* sangat berperan penting dalam suatu rangkaian karena dapat mengontrol rangkaian beban arus tinggi dengan arus yang rendah.

Dengan menggunakan rangkaian *relay*, kita juga tidak perlu mengontrol ataupun mengoperasikan suatu perangkat karena dapat dilakukan dari jarak jauh sehingga nantinya dapat mempermudah dan memperlancar pekerjaan. Biasanya, rangkaian *relay* terdapat pada televisi, *transmitter*, *sound system*, rangkaian bel, lampu taman otomatis, pengendali peralatan listrik dan perangkat elektronika lainnya.

2.3 Modem Wavecom



Gambar 2.4 Modem Wavecom

Wavecom adalah pabrikan asal Perancis (bermarkas di kota Issy les Moulineaux, Perancis) yaitu Wavecom SA yang berdiri sejak 1993 bermula sebagai biro konsultan teknologi dan sistim jaringan nirkabel GSM, dan pada 1996 Wavecom mulai membuat desain daripada modul wireless GSM pertamanya dan diresmikan pada 1997, bentuk modul GSM pertama berbasis GSM dan pengkodean khusus yang disebut AT Command. Sulit mencari referensi module tipe apa yang pertama dibuat oleh Wavecom SA.

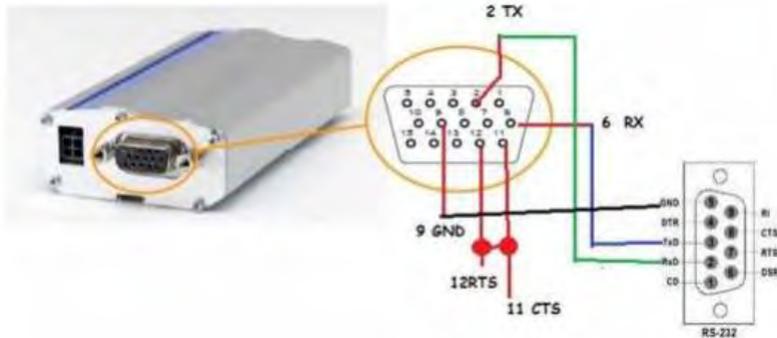
Beberapa kegunaan *modem wavecom*:

1. *SMS Broadcast application*

2. SMS Quiz application
3. SMS Polling
4. SMS auto-reply
5. M2M integration
6. Aplikasi Server Pulsa
7. Telemetri
8. Payment Point Data
9. PPOB

Modem ini menggunakan komunikasi *serial* dalam pengaplikasiannya. Komunikasi *serial* merupakan komunikasi data dengan pengiriman data secara satu per satu pada waktu tertentu. Sehingga komunikasi data serial hanya menggunakan dua kabel yaitu kabel data untuk pengiriman yang disebut *transmit* (TX) dan kabel data untuk penerimaan yang disebut *receive* (RX). Kelebihan dari komunikasi *serial* adalah jarak pengiriman dan penerimaan dapat dilakukan dalam jarak yang cukup jauh dibandingkan dengan komunikasi paralel tetapi kekurangannya kecepatannya lebih lambat dibandingkan komunikasi paralel. Dikenal dua cara komunikasi data secara serial, yaitu komunikasi data secara *sinkron* dan komunikasi data secara *asinkron*. Pada komunikasi data serial *sinkron*, *clock* dikirimkan bersama-sama dengan data serial, sedangkan komunikasi data serial *asinkron*, *clock* tidak dikirimkan bersama data *serial*, tetapi dibangkitkan secara sendiri-sendiri baik pada sisi pengirim (*transmitter*) maupun pada sisi penerima (*receiver*).

komunikasi antara *modem wavecom* dengan mikrokontroler yang digunakan adalah komunikasi serial secara asinkron yang bersifat *full-duplex*, artinya port serial bisa mengirim dan menerima pada waktu yang bersamaan. Perangkat yang digunakan yaitu kabel komunikasi *serial RS232* yang biasa digunakan untuk menghubungkan perifer eksternal seperti *modem* dengan komputer. *Modem* memiliki level tegangan yang berbeda dengan level tegangan TTL ataupun *RS232*, tetapi untuk kompatibilitas *modem* agar bisa terkoneksi dengan *PC* guna berbagai keperluan maka disediakan kabel data yang *compatible* dengan standar *RS232* sebagai *interface* untuk koneksi ke *PC*, untuk konfigurasi *port* data *modem* yang digunakan yaitu *wavecom m1306b*. Dengan alasan inilah maka digunakan komunikasi serial standar *RS232* sebagai dasar *interface* antara *modem* dengan mikrokontroler.



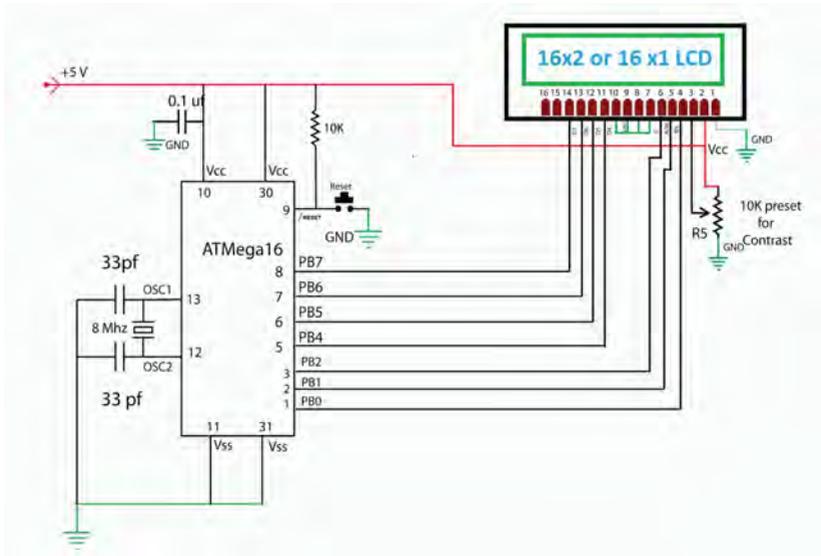
Gambar 2.5 Konfigurasi *Pin Modem Wavecom*

2.4 LCD 2x16

Display elektronik adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD (*Liquid Cristal Display*) adalah salah satu jenis display elektronik yang dibuat dengan teknologi CMOS *logic* yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap *front-lit* atau mentransmisikan cahaya dari *back-lit*. LCD (*Liquid Cristal Display*) berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik.

Material LCD (*Liquid Cristal Display*) LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan *seven-segment* dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. Lapisan sandwich memiliki *polarizer* cahaya vertikal depan dan *polarizer* cahaya horisontal belakang yang diikuti dengan lapisan reflektor. Cahaya yang dipantulkan tidak dapat melewati molekul-molekul yang telah menyesuaikan diri dan segmen yang diaktifkan terlihat menjadi gelap dan membentuk karakter data yang ingin ditampilkan.

Adapun konfigurasi LCD dengan AT Mega 16 adalah sebagai berikut:



Gambar 2.6 Konfigurasi *LCD*

Rangkaian ini sesuai dengan konfigurasi *LCD port* di CodeVisionAVR pada saat inialisasi dan pengaturan di CodeWizardAVR. Sedangkan rangkaian lainnya merupakan sistem minimum mikrokontroler pada umumnya.



Gambar 2.7 LCD 2x16

Pengendali / Kontroler LCD (*Liquid Cristal Display*) dalam modul LCD (*Liquid Cristal Display*) terdapat *microcontroller* yang berfungsi sebagai pengendali tampilan karakter LCD (*Liquid Cristal Display*). *Microntroller* pada suatu LCD (*Liquid Cristal Display*) dilengkapi dengan memori dan *register*. Memori yang digunakan *microcontroler internal LCD* adalah : *DDRAM (Display Data Random Access Memory)* merupakan memori tempat karakter yang akan ditampilkan berada. *CGRAM (Character Generator Random Access Memory)* merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai dengan keinginan. *CGROM (Character Generator Read Only Memory)* merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut merupakan karakter dasar yang sudah ditentukan secara permanen oleh pabrikan pembuat LCD (*Liquid Cristal Display*) tersebut sehingga pengguna tinggal mengambilnya sesuai alamat memorinya dan tidak dapat merubah karakter dasar yang ada dalam *CGROM*.

Register control yang terdapat dalam suatu LCD diantaranya adalah register perintah yaitu register yang berisi perintah-perintah dari mikrokontroler ke panel LCD (*Liquid Cristal Display*) pada saat proses

penulisan data atau tempat status dari panel *LCD (Liquid Cristal Display)* dapat dibaca pada saat pembacaan data. *Register* data yaitu *register* untuk menuliskan atau membaca data dari atau ke *DDRAM*. Penulisan data pada *register* akan menempatkan data tersebut ke *DDRAM* sesuai dengan alamat yang telah diatur sebelumnya. Pin, kaki atau jalur input dan kontrol dalam suatu *LCD (Liquid Cristal Display)* diantaranya adalah: Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan *LCD (Liquid Cristal Display)* dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit. Pin *RS (Register Select)* berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika *low* menunjukkan yang masuk adalah perintah, sedangkan logika *high* menunjukkan data. Pin *R/W (Read Write)* berfungsi sebagai instruksi pada modul jika *low* tulis data, sedangkan *high* baca data. Pin *E (Enable)* digunakan untuk memegang data baik masuk atau keluar. Pin *VLCD* berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 Kohm, jika tidak digunakan dihubungkan ke *ground*, sedangkan tegangan catu daya ke *LCD* sebesar 5 Volt.

2.5 Motor DC Gearbox



Gambar 2.8 Motor DC dengan Gearbox

Konsumsi arus saat berputar tanpa beban sebesar 190 mA (maksimum 250 mA), pada kondisi tanpa beban/hambatan kecepatan putar sekitar 90 rpm. Torsi maksimum 800 gf.cm. *Gearbox* ini memiliki dua sumbu (dual axis) dengan ujung datar (flat axle 1.9mm). Sumbu ganda ini berguna untuk memasangkan roda pada ujung yang satu dan

speed encoder pada sisi seberangnya dengan demikian mikrokontroler dapat mengukur kecepatan aktual dengan menggunakan *komponen opto-interrupter* semacam H2210.

2.6 Adaptor DC 12V



Gambar 2.9 Adaptor DC 12V

Adaptor adalah sebuah rangkaian elektronika yang dapat mengubah tegangan AC menjadi DC. Rangkaian ini adalah alternatif pengganti dari sumber tegangan DC, misalnya baterai dan *accumulator*. Keuntungan dari *adaptor* dibanding dengan baterai atau *accumulator* adalah sangat praktis berhubungan dengan ketersediaan tegangan karena *adaptor* dapat di ambil dari sumber tegangan AC yang ada di rumah, di mana pada jaman sekarang ini setiap rumah sudah menggunakan listrik. Selain itu, *adaptor* mempunyai jangka waktu yang tidak terbatas asal ada tegangan AC, tegangan AC ini sudah merupakan kebutuhan primer dalam kehidupan manusia.

Arus listrik yang kita gunakan di rumah, kantor dan pabrik pada umumnya adalah dibangkitkan, dikirim dan didistribusikan ke tempat masing-masing dalam bentuk Arus Bolak-balik atau arus AC (*Alternating Current*). Hal ini dikarenakan pembangkitan dan pendistribusian arus listrik melalui bentuk arus bolak-balik (AC) merupakan cara yang paling ekonomis dibandingkan dalam bentuk arus searah atau arus DC (*Direct Current*).

Akan tetapi, peralatan elektronika yang kita gunakan sekarang ini sebagian besar membutuhkan arus DC dengan tegangan yang lebih rendah untuk pengoperasiannya. Oleh karena itu, hampir setiap peralatan elektronika memiliki sebuah rangkaian yang berfungsi untuk melakukan konversi arus listrik dari arus AC menjadi arus DC dan juga untuk menyediakan tegangan yang sesuai dengan rangkaian Elektronika-nya. Rangkaian yang mengubah arus listrik AC menjadi DC ini disebut dengan *DC Power Supply* atau dalam bahasa Indonesia disebut dengan *Catu daya DC*. *DC Power Supply* atau *Catu Daya* ini juga sering dikenal dengan nama "*Adaptor*".

Sebuah *DC Power Supply* atau *adaptor* pada dasarnya memiliki 4 bagian utama agar dapat menghasilkan arus DC yang stabil. Keempat bagian utama tersebut diantaranya adalah *Transformer*, *Rectifier*, *Filter* dan *Voltage Regulator*.

Prinsip Kerja DC Power Supply (*Adaptor*) yaitu transformator (*Transformer*) atau disingkat dengan *Trafo* yang digunakan untuk *DC Power supply* adalah Transformator jenis *step-down* yang berfungsi untuk menurunkan tegangan listrik sesuai dengan kebutuhan komponen *elektronika* yang terdapat pada rangkaian *adaptor (DC Power Supply)*. Transformator bekerja berdasarkan prinsip induksi elektromagnetik yang terdiri dari 2 bagian utama yang berbentuk lilitan yaitu lilitan *primer* dan lilitan sekunder. Lilitan *primer* merupakan *input* dari pada Transformator sedangkan *output*-nya adalah pada lilitan sekunder. Meskipun tegangan telah diturunkan, *output* dari Transformator masih berbentuk arus bolak-balik (arus AC) yang harus diproses selanjutnya.

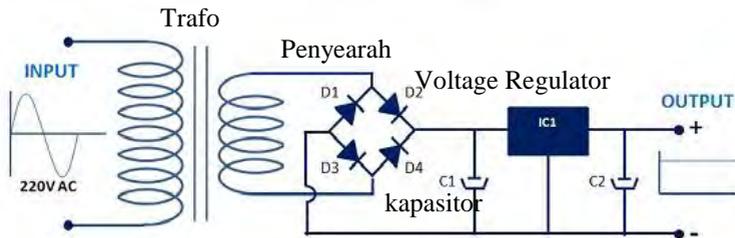
Rectifier atau penyearah gelombang adalah rangkaian Elektronika dalam Power Supply (catu daya) yang berfungsi untuk mengubah gelombang AC menjadi gelombang DC setelah tegangannya diturunkan oleh *transformator step down*. Rangkaian *rectifier* biasanya terdiri dari komponen *dioda*. Terdapat 2 jenis rangkaian *rectifier* yaitu "*Half Wave Rectifier*" yang hanya terdiri dari 1 komponen dioda dan "*Full Wave Rectifier*" yang terdiri dari 2 atau 4 komponen dioda.

Dalam rangkaian *power supply (Adaptor)*, *filter* digunakan untuk meratakan sinyal arus yang keluar dari *rectifier*. *Filter* ini biasanya terdiri dari komponen kapasitor (Kondensator) yang berjenis Elektrolit atau ELCO (Electrolyte Capacitor).

Untuk menghasilkan Tegangan dan Arus DC (arus searah) yang tetap dan stabil, diperlukan *Voltage Regulator* yang berfungsi untuk

mengatur tegangan sehingga tegangan *output* tidak dipengaruhi oleh suhu, arus beban dan juga tegangan input yang berasal *output filter*. *Voltage Regulator* pada umumnya terdiri dari *Dioda Zener*, *Transistor* atau *IC (Integrated Circuit)*.

Pada *DC power supply* yang canggih, biasanya *voltage regulator* juga dilengkapi dengan *short circuit protection* (perlindungan atas hubung singkat), *current limiting* (Pembatas Arus) ataupun *over voltage protection* (perlindungan atas kelebihan tegangan).



Gambar 2.10 Rangkaian *Adaptor DC 12V*

BAB III
PERANCANGAN DAN PEMBUATAN
ALAT

BAB III PERANCANGAN DAN PEMBUATAN ALAT

Bab ini membahas perancangan sistem yang meliputi perancangan *hardware*, *software* dan perancangan mekanik. Detail mengenai pembahasan perancangan akan dibahas pada beberapa subbab berikut.

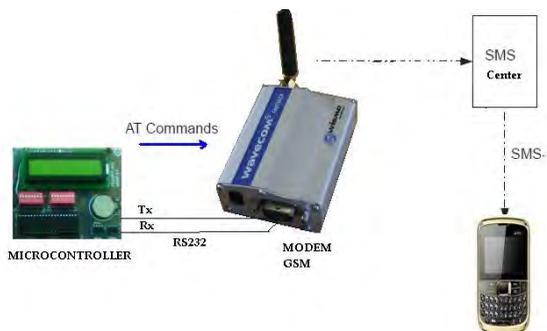
3.1 Deskripsi Kerja Alat

Cara kerja alat ini setelah ATMega16 akan menerima sumber tegangan dari adaptor. Setelah itu, maka *modem* akan melakukan pengiriman sms dengan format “READY” kepada nomor yang telah disimpan pada *microcontroller* sebelumnya, maka alat siap digunakan. Pengguna hanya tinggal membalas pesan dengan format: “P1”, “P2”, atau “P3”. Setelah pengiriman sms tersebut, dengan jeda 1 hingga 2 detik (saat sms berhasil masuk pada modem), pintu bagian bawah tempat penyimpanan pakan akan terbuka sesuai dengan perintah yang dikimkan. Perintah “P1” untuk membuka pintu selama 1 detik, “P2” untuk 2 detik, dan perintah “P3” untuk membuka selama 3 detik.

3.2 Perancangan Perangkat Keras (*Hardware*)

Perancangan perangkat keras pada alat terdiri dari perancangan rangkaian *microcontroller*, *modem wavecom* dan perancangan rangkaian *driver motor*.

Blok diagram dari keseluruhan perancangan dan pembuatan alat pemberi pakan pada kandang ini dengan menggunakan sms yang di desain pada tugas akhir ini terlihat seperti gambar 3.1



Gambar 3.1 Diagram Alat Secara Keseluruhan

Proses kerja rangkaian *relay* ini relay akan dihubungkan dengan rangkaian *microcontroller ATmega16*. Dengan kondisi ini maka *relay* motor mendapatkan tegangan positif (+) dari *relay* lainnya dan mendapatkan tegangan negatif (-) sehingga *relay* akan membentuk *induksi magnet* didalam kumparan dan akan mengaktifkan motor pada pembuka makanan.

Perancangan *hardware* dilakukan dengan merancang rangkaian-rangkaian elektronika dan rancangan mekanik. Perancangan *hardware* ini meliputi:

1. Rangkaian *Microcontroller*
2. Rangkaian *driver relay*

Gambaran secara keseluruhan menjelaskan tentang perancangan sms dengan *ATmega16*. Perancangan tugas akhir ini menggunakan beberapa *hardware*. Dimulai dari *PC* yang digunakan sebagai media perancangan sistem pada program yang akan digunakan oleh *mikrokontroler ATmega16*. Setelah itu *mikrokontroler ATmega16* yang akan mengatur berjalannya sistem antara modem dengan *relay* sebagai kontaktor yang akan mengaktifkan *motor dc*. Pada peralatan ini digunakan dua buah *relay* yang memiliki fungsi berbeda. *Relay* satu berfungsi mengaktifkan *motor dc* agar ketika kontak *relay* aktif maka motor juga akan selalu pada keadaan *standby*. Ini bertujuan agar ketika *relay* satu berada pada kondisi '1' maka *motor* selalu berada pada kondisi 'on' hingga tegangan yang di berikan oleh *power supply* di putus. Sedangkan *relay* dua yang dihubungkan dengan *starter* digunakan sebagai pengatur sistem *motor*. Pengaturannya adalah ketika sms yang masuk sesuai dengan *sms* sampel yang telah disimpan sebelumnya maka *mikrokontroler ATmega16* akan mengaktifkan *relay* dua pada kondisi logika '1' maka motor secara otomatis akan membuka pintu pakan dan menutupnya kembali sesuai dengan perintah sms yang dikirimkan.

3.2.1 Perancangan Rangkaian Relay

Sesuai dengan kegunaan *relay* yang dapat berfungsi sebagai *driver motor* maka disini *relay* akan menerima input dari *microcontroller ATmega16*. Dan akan diteruskan untuk mengontrol *motor dc* dengan *gearbox* sebagai pembuka dan penutup pintu pakan.



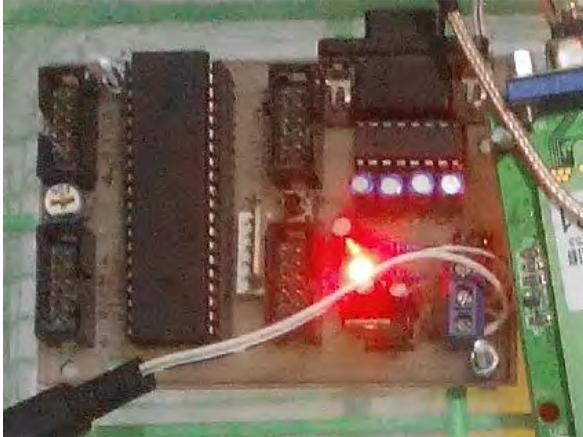
Gambar 3.2 Posisi Peletakan *Relay* di bawah *Modem Wavecom*



Gambar 3.3 Pemberian *Input Power*

3.2.2 Modul *Microcontroller* ATmega16

Perangkat utama yang digunakan oleh perancangan alat ini adalah mikrokontroler ATmega16 yang digunakan sebagai media komunikasi antara *modem wavecom* dengan *relay* sehingga *motor* dapat membuka dan menutup pintu pakan melalui sms. Jadi ATmega16 inilah yang mengatur berjalannya sistem pada perancangan alat ini.



Gambar 3.4 Modul ATmega16

3.3 Perancangan *Software*

Perancangan perangkat lunak (*software*) pada perancangan dan pembuatan alat ini meliputi pembuatan program pengenalan sms pengguna dan sebuah program *LCD* 2x16 dengan menggunakan ATmega16. Perancangan ini menggunakan *software* “cvavr”.

```
D:\Data\pakan kucing\KUC1.c
Notes KUC1.c KUC1.asm KUC1.lst
31 r jg,data,a1,a2;
32 d baca_sms()
33
34 data=getchar(); // baca data masuk persatu karakter
35 while(data!='+') // mencari +
36 {data=getchar();}
37
38     if (data=='+')
39     {
40         data=getchar();
41         if (data=='C')
42         {
43             data=getchar();
44             if (data=='M')
45             {
46                 data=getchar();
47                 if (data=='T')
48                 {
49                     {
50                         data=getchar();
51                         while (data!='/') // cari /
52                         {
53                             data=getchar();
54                             }
55                         while (data!='+') // cari petik "
```

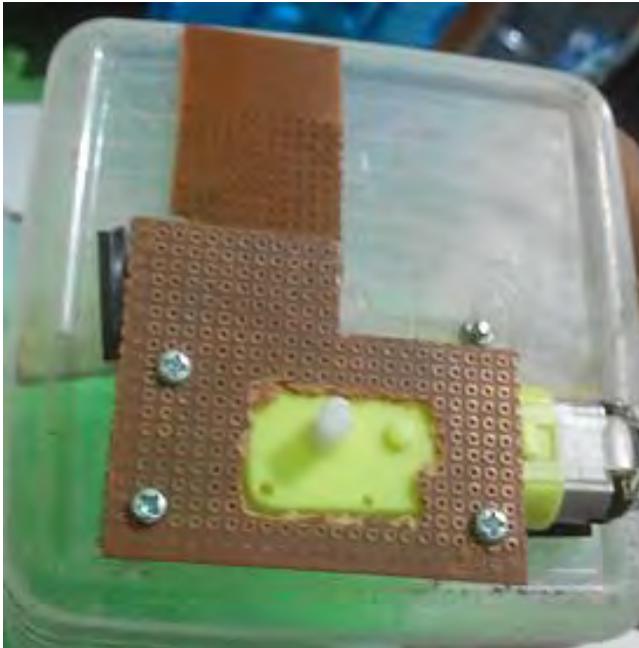
Gambar 3.5 Perancangan *Software*



Gambar 3.6 Penginputan *Software* pada *Microcontroller*

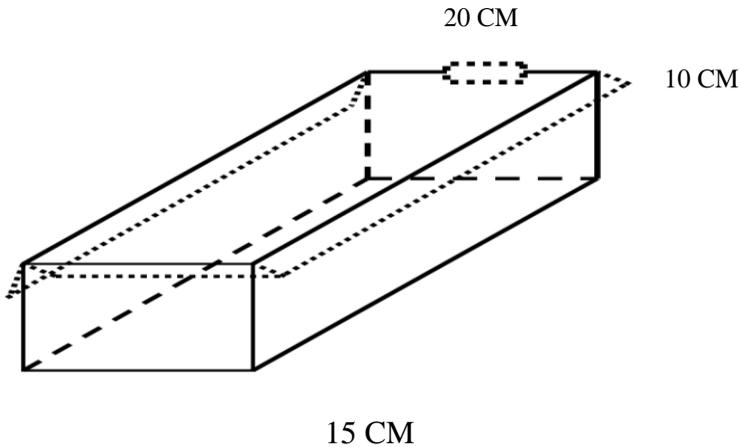
3.4 Perancangan Mekanik

Mekanik pintu pembuka makanan terdiri dari sebuah *motor* yang di tempatkan di bawah penyimpanan pakan. Untuk pintunya terbuat dari bahan plastik atau akrilik atau sesuai dengan kebutuhan yang akan disambung dengan *motor* dc yang sudah disambung pada *relay*.



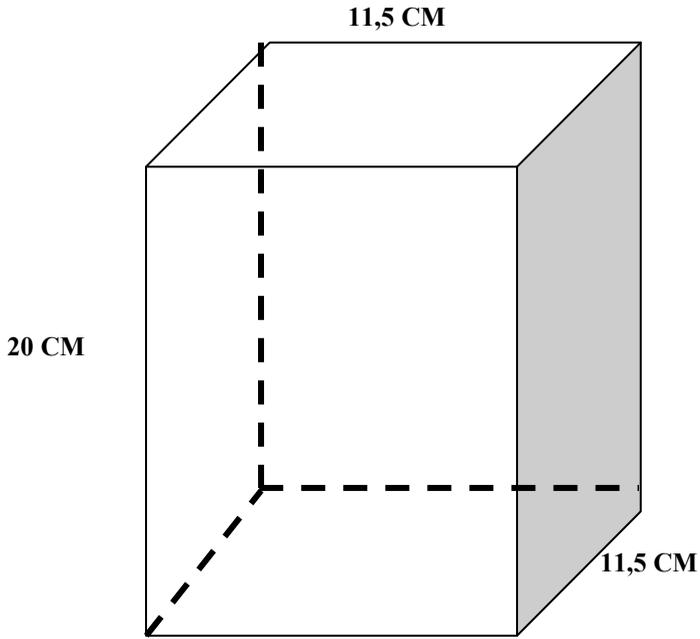
Gambar 3.7 Mekanik Pintu Pakan

Dibawah ini rancangan mekanik dibuat sebuah kotak yang digunakan sebagai tempat penyimpanan rangkaian *microcontroller*, *driver motor*, dan *modem wavecom* agar rangkaian dapat digunakan. Berikut pada Gambar 3.9 merupakan ukuran *box* yang digunakan *microcontroller*.



Gambar 3.8 Rancangan Kotak Penyimpanan *Microcontroller*, *Driver Motor*, dan *Modem Wavecom*

Dibawah ini pada gambar 3.10 adalah rancangan mekanik sebuah kotak yang terbuat dari *tupperware*. Fungsinya sebagai tempat penyimpanan pakan dari peralatan tugas akhir.



Gambar 3.9 Perancangan Kotak Penyimpanan Pakan

BAB IV
PENGUJIAN DAN ANALISA DATA

BAB IV

PENGUJIAN DAN ANALISA DATA

Untuk mengetahui apakah tujuan-tujuan dari pembuatan alat ini telah tercapai atau belum, maka perlu dilakukannya sebuah pengujian dan analisa terhadap alat yang telah dibuat. Dan sebagai acuan yang tidak terpisahkan adalah adanya proses *evaluasi* sehingga akan dapat dilakukan langkah-langkah positif guna membawa alat ini kearah yang lebih baik. Hasil pengujian tersebut akan dianalisa untuk mengetahui penyebab terjadinya kekurangan dalam sistem. Pengujian tersebut meliputi:

1. Pengujian Perangkat Keras (*Hardware*)
 - a. Mikrokontroller ATMega16
 - b. *Adaptor*
 - c. Rangkaian *Relay*
2. Pengujian Perangkat Lunak (*Software*)
 - a. Pengujian Program SMS
3. Pengujian Alat secara Keseluruhan

4.1 Pengujian Perangkat Keras

Perangkat keras (*hardware*) yang dipakai pada tugas akhir kali ini meliputi rangkaian ATMega16, *adaptor* dan rangkaian *relay*. Berikut adalah pembahasan tentang pengujian *mikrokontroller* Atmega16, *adaptor* dan rangkaian *relay* yang dipakai pada perancangan dan pembuatan alat pemberian pakan pada kandang kucing via sms

4.1.1 Pengujian Pada Mikrokontroller ATMega16

Pada perancangan tugas akhir kali ini pengatur sistemnya menggunakan mikrokontroller ATMega16. Pengujiannya dilakukan dengan mengukur keluaran dari ATMega16 dalam keadaan pada logika 0 (*low*) dan pada logika 1 (*high*). Berikut hasil pengujian pada mikrokontroller ATMega16:

- Jika ATMega16 berada pada keadaan logika 0 (*low*) maka keluaran menghasilkan nilai 0 *Volt*.
- Jika ATMega16 berada pada keadaan logika 1 (*high*) maka keluaran menghasilkan nilai 4,83 *Volt*.

4.1.2 Pengujian *Adaptor*

Pada perancangan tugas akhir kali ini *adaptor* digunakan sebagai sumber tegangan yang akan diberikan pada mikrokontroler ATmega16. Rangkaian *adaptor* berikut bertugas menurunkan tegangan AC 220V menjadi DC 12V. Jadi, pada kesimpulannya adaptor berikut mempunyai tegangan keluaran sebesar 12V. Yang akan diuji pada *adaptor* berikut adalah besar tegangan masukan pada *adaptor* dan tegangan keluaran pada *adaptor*.

Berikut pada tabel 4.1 adalah tabel pengujian *adaptor* pada perancangan alat :

Pengukuran Ke -	Tegangan Masukan	Tegangan Keluaran	
	<i>V_{in}</i> (Volt)	<i>V_{out}</i> Tanpa Beban	<i>V_{out}</i> Dengan Beban
1	220V	11,95V	11,90 V
2	220V	11,95 V	11,90 V
3	220 V	11,95 V	11,90 V
4	220V	11,95 V	11,90 V
5	220 V	11,95 V	11,90 V
V rata-rata	220V	11,95 V	11,90 V

Tabel 4.1 Pengujian *Adaptor*

Pengujian tanpa beban dilakukan dengan mengukur keluaran dengan *Volt* meter.

Dibawah ini dilihat pada gambar 4.1 adalah bentuk fisik dari *adaptor* yang digunakan pada perancangan alat pemberian pakan via sms.



Gambar 4.1 Bentuk Fisik *Adaptor* DC 12V

4.1.3 Pengujian Rangkaian *Relay*

Rangkaian yang dipakai pada perancangan tugas akhir ini digunakan sebagai *output* yang bertugas untuk mengaktifkan *motor DC*. *Relay* yang dipakai yaitu *relay 2 channel*. Pengujian pada *relay* dilakukan pada waktu kerja *relay* setelah alat dijalankan.

Berikut pada tabel 4.2 hasil dari pengujian rangkaian *relay* pada perancangan alat:

Pengujian Ke -	Waktu Kerja <i>Relay</i>
1	1,36 detik
2	1,43 detik
3	1,53 detik
4	1,29 detik
5	1,50 detik
6	1,39 detik
7	1,29 detik
8	1,34 detik
9	1,28 detik
10	1,25 detik
Rata – Rata	1,366 detik

Tabel 4.2 Pengujian Rangkaian *Relay*

Dibawah ini dapat dilihat pada gambar 4.2 adalah bentuk fisik dari rangkaian *relay* yang digunakan pada perancangan dan pembuatan alat pemberian pakan dengan menggunakan sms:



Gambar 4.2 Bentuk Fisik Rangkaian *Relay*

4.2 Pengujian Perangkat Lunak (*Software*)

Perangkat lunak (*software*) yang digunakan pada perancangan dan pembuatan pemberian Pakan Kucing Via Sms adalah program CYAVR. Program yang dirancang meliputi program *delay* (lama waktu terbukanya pintu pakan) yaitu berupa: “P1, P2, atau P3”.

4.2.1 Pengujian Program

Perancangan program berikutnya adalah sms pada alat dan. Hal ini bertujuan agar ketika sms yang dikirim pengguna sesuai dengan sampel sms yang telah tersimpan dalam memori, seketika *motor* dapat aktif secara otomatis. Dikatakan berhasil ketika alat diberi inputan power kemudian alat akan mengirim sms kepada nomor yang telah disimpan dengan format “*READY*”. Kemudian alat akan merespon ketika pengguna membalas dengan salah satu format yaitu “P1, P2, atau P3” dan alat merespon dengan membuak pakan maka pengujian ketepatan sms dapat bekerja dengan baik. Artinya, ketika sms yang dikirim tidak

sesuai dengan sampel yang disimpan maka *relay* tidak akan mengaktifkan *motor*, namun sebaliknya ketika sms sesuai dengan sampel yang telah disimpan sebelumnya maka *motor* akan segera aktif secara otomatis. Digunakan beberapa *variabel* pengujian dalam pengujian program kali ini, diantaranya keberhasilan program sampel sms, serta waktu yang diperlukan dalam proses autentifikasi sensor.

Berikut ini pada tabel 4.3, 4.4 dan 4.5 adalah tabel hasil pengujian dari program sms pada alat:

Pengujian Ke -	Waktu Kerja Starter	Tingkat keberhasilan
1	1,69 detik	Gagal
2	1,82 detik	Berhasil
3	1,31 detik	Berhasil
4	1,48 detik	Berhasil
5	1,48 detik	Berhasil
6	1,25 detik	Berhasil
7	1,67 detik	Berhasil
8	1,72 detik	Berhasil
9	1,58 detik	Berhasil
10	1,38 detik	Berhasil
Rata – Rata	1,538 detik	10 % gagal 90% berhasil

Tabel 4.3 Pengujian dari program P1 dengan *Delay* 1000ms

Pengujian Ke -	Waktu Kerja Starter	Tingkat keberhasilan
1	1,32 detik	Gagal
2	1,46 detik	Gagal
3	1,48 detik	Gagal
4	1,57 detik	Berhasil
5	1,36 detik	Berhasil
6	1,52 detik	Berhasil
7	1,36 detik	Berhasil
8	1,42 detik	Berhasil
9	1,64 detik	Berhasil
10	1,53 detik	Berhasil
Rata – Rata	1,466 detik	30 % gagal 70% berhasil

Tabel 4.4 Pengujian dari Program P2 dengan Delay 2000ms

Pengujian Ke -	Waktu Kerja Starter	Tingkat keberhasilan
1	1,34 detik	Gagal
2	1,42 detik	Gagal
3	1,36 detik	Berhasil
4	1,54 detik	Berhasil
5	1,63 detik	Berhasil
6	1,36 detik	Berhasil
7	1,44 detik	Berhasil
8	1,32 detik	Berhasil
9	1,62 detik	Berhasil
10	1,54 detik	Berhasil
Rata – Rata	1,457 detik	20 % gagal 80% berhasil

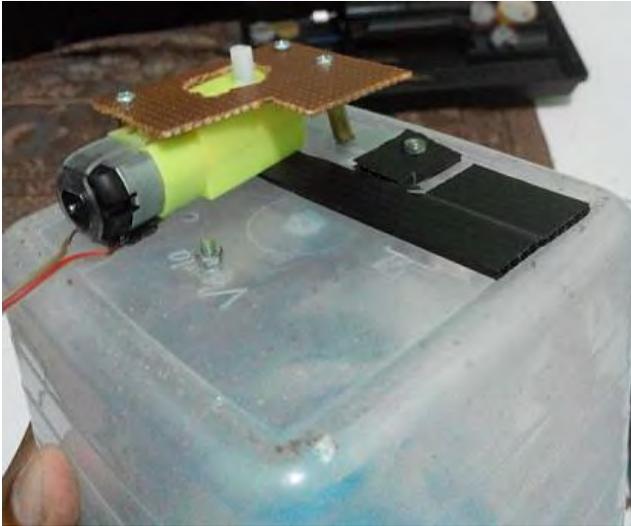
Tabel 4.5 Pengujian dari Program P3 dengan Delay 3000ms

4.3 Pengujian Alat Secara Keseluruhan

Pengujian alat secara keseluruhan dihasilkan beberapa kesimpulan yakni:

- Jika ATmega16 pada logika 0 maka *outputnya* bernilai 0V.
- Jika ATmega16 pada logika 1 maka *outputnya* bernilai 4,83V.
- Tegangan keluaran rata-rata pada *adaptor* tanpa beban adalah 11,95V dengan persentase kesalahan pada rangkaian sebesar 0,55 %
- Tegangan keluaran rata-rata pada *adaptor* dengan beban adalah 11,90V dengan persentase kesalahan pada rangkaian sebesar 1,11 %
- Rata-rata waktu kerja *relay* selama 1,366 *second*.
- Pada saat *delay* 1000ms maka rata-rata waktu kerja selama 1,538 *second* dengan tingkat keberhasilan 10% gagal dan 90% berhasil.
- Pada saat *delay* 2000 ms maka rata-rata waktu kerja selama 1,466 *second* dengan tingkat keberhasilan 30% gagal dan 70% berhasil.
- Pada saat *delay* 3000ms maka rata-rata waktu kerja selama 1,457 *second* dengan tingkat keberhasilan 20% gagal dan 80% berhasil.

Dibawah ini adalah gambar implementasi perancangan alat. Dapat dilihat pada gambar 4.3 merupakan penempatan *motor DC* sebagai pembuka pintu pakan. Dan pada gambar 4.4, 4.5, 4.6 dan 4.7 merupakan alat elektronik perancangan alat serta penempatan alat pada kandang.



Gambar 4.3 Penempatan *motor* DC sebagai pembuka pintu pakan



Gambar 4.4 Bentuk Perancangan Peralatan Elektronik



Gambar 4.5 Penempatan Peralatan Elektronik pada Cover



Gambar 4.6 Relay Motor



Gambar 4.7 Penempatan Alat pada Kandang

Halaman ini sengaja dikosongkan

BAB I
PENDAHULUAN

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Salah satu kebutuhan utama pada hewan adalah makanan. Terutama pada hewan peliharaan yang tidak dapat mencari makan secara mandiri. Seperti kucing peliharaan jenis tertentu. Maka dibutuhkan perhatian yang secara terus menerus dari pemilik kucing tersebut. Sehingga terkadang terasa sangat menyita waktu bagi pemilik kucing tersebut, terutama bagi pemilik dengan jam kerja panjang atau bagi pemilik yang akan melakukan perjalanan jauh dan tidak dapat membawa kucing peliharaannya. Solusi selama ini adalah dengan menitipkan kucing peliharaan tersebut kepada *pet shop*, tetapi tentu saja dengan biaya yang tidak sedikit. Selain itu terdapat pula resiko penularan penyakit dari kucing lain yang berdekatan kandangnya, kucing pun menjadi tidak akrab lagi dengan pemilik setelah pulang dari penitipan tersebut, sehingga bagi pemilik kucing yang akan melakukan perjalanan jauh akan sangat kerepotan untuk memberi pakan kucing kesayangannya tersebut. Sementara bagi hewan seperti kucing, makanan tidak dapat diberikan secara langsung dengan jumlah yang banyak, ini adalah yang dilakukan banyak orang yang akan bepergian jauh dan meninggalkan kucing peliharaan mereka sendirian, padahal makan kucing yang bersifat kering sebaiknya disimpan ditempat yang tertutup dan tidak terkena udara bebas dan kucing hanya diberikan makanan sesuai porsi untuk dihabiskan saat itu juga.

1.2 Permasalahan

Melihat latar belakang di atas maka dirumuskan permasalahan sebagai berikut:

- ▶ Bagaimana merancang sebuah sistem pemberian pakan hewan menggunakan sms?

1.3 Batasan Masalah

Agar tugas akhir tidak menyimpang dari ketentuan yang digariskan maka diambil batasan dengan asumsi sebagai berikut:

- ▶ Tugas akhir ini hanya purwarupa karena saya hanya merancang sistem yang dapat dikembangkan dalam skala besar.

1.4 Tujuan

- ▶ Merancang sistem pemberian pakan pada kandang kucing menggunakan sms.
- ▶ Merancang *hardware* berupa kandang kucing dan alat pemberian pakan.

1.5 Sistematika Laporan

Dalam penyusunan buku Tugas Akhir ini, pembahasan mengenai sistem alat yang dibuat dibagi menjadi lima bab dengan sistematika sebagai berikut:

BAB I : PENDAHULUAN

Dalam bab ini mendiskripsikan tentang latar belakang, permasalahan, batasan masalah, tujuan, ruang lingkup, sistematika laporan, serta relevansi penulisan Tugas Akhir ini.

BAB II : TEORI PENUNJANG

Dalam bab ini berisi penjelasan dasar teori mengenai konsep yang digunakan dalam perancangan Tugas Akhir ini, meliputi pembahasan tentang ATmega16, *Modem Wavcom*, dan *Motor DC*.

BAB III: PERANCANGAN DAN PEMBUATAN ALAT

Dalam bab ini akan dibahas secara detail tentang rangkaian-rangkaian yang digunakan serta perangkat lunak berupa program yang membantu pengaktifan alat tersebut.

BAB IV : PENGUJIAN DAN ANALISA DATA

Dalam bab ini berisi data-data pengukuran dan pengujian beserta analisa terhadap prinsip kerja dari alat yang dibuat meliputi pengujian mekanik *hardware* (elektronik), *software* dan pengujian keseluruhan.

BAB V : PENUTUP

Dalam bab ini berisi kesimpulan yang diperoleh dari pembuatan Tugas Akhir ini dan saran-saran untuk pengembangan lebih lanjut.

1.6 Relevansi

Manfaat untuk masyarakat (para pemilik hewan peliharaan):
Membantu menyumbangkan sebuah gagasan baru berupa sistem pemberian pakan kucing sehingga para pemilik hewan dapat lebih mudah dalam perawatan terutama pada pemberian pakan untuk hewan mereka.

Halaman ini Sengaja dikosongkan

BAB II
TEORI PENUNJANG

BAB II TEORI PENUNJANG

Pada bab ini menjelaskan mengenai teori-teori pendukung yang berkaitan dengan topik alat yang akan dibuat meliputi ATmega16, *driver motor*, *modem wavocom*, *lcd 2x16*, *motor dc gearbox*, dan *adaptor dc 12V*.

2.1 Mikrokontroler AVR ATMEGA16

Mikrokontroler ATmega16 Mikrokontroler adalah sebuah sistem komputer lengkap dalam satu *chip*. *Microcontroller* lebih dari sekedar sebuah mikroprosesor karena sudah terdapat atau berisikan ROM (*Read-Only Memory*), RAM (*Read-Write Memory*), beberapa masukan maupun keluaran, dan beberapa peripheral seperti pencacah/pewaktu, *ADC (Analog to Digital converter)*, *DAC (Digital to Analog converter)* dan serial komunikasi. Salah satu mikrokontroler yang banyak digunakan saat ini yaitu mikrokontroler AVR. AVR adalah mikrokontroler RISC (*Reduce Instruction Set Computer*) 8 bit berdasarkan arsitektur Harvard. Secara umum mikrokontroler AVR dapat dikelompokkan menjadi 3 kelompok, yaitu keluarga AT90Sxx, ATmega dan ATtiny. Pada dasarnya yang membedakan masing-masing kelas adalah memori, *peripheral*, dan fiturnya. Seperti mikroprosesor pada umumnya, secara internal mikrokontroler ATmega16 terdiri atas unit-unit fungsionalnya *Arithmetic and Logical Unit (ALU)*, himpunan register kerja, *register* dan *decoder* instruksi, dan pewaktu beserta komponen kendali lainnya. Berbeda dengan mikroprosesor, mikrokontroler menyediakan memori dalam prosesornya (*in chip*).



Gambar 2.1 ATmega16

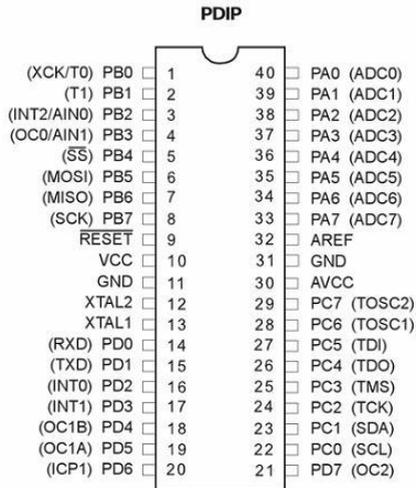
AVR merupakan seri mikrokontroler *Complementary Metal Oxide Semiconductor* (CMOS) 8-bit buatan Atmel berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Hampir semua instruksi pada program dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register *general-purpose*, *timer/counter* fleksibel dengan mode *compare*, interupsi *internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, *power saving mode*, ADC dan PWM. AVR pun mempunyai *In-System Programmable (ISP) Flash on-chip* yang memungkinkan memori program untuk diprogram ulang (*read/write*) dengan koneksi secara serial yang disebut *Serial Peripheral Inteface (SPI)*.

AVR memiliki keunggulan dibandingkan dengan mikrokontroler lain, keunggulan mikrokontroler AVR yaitu memiliki kecepatan dalam mengeksekusi program yang lebih cepat, karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock* (lebih cepat dibandingkan mikrokontroler keluarga MCS 51 yang memiliki arsitektur *Complex Intrukstion Set Compute*). ATMEGA16 mempunyai *throughput* mendekati 1 *Millions Instruction Per Second (MIPS)* per MHz, sehingga membuat konsumsi daya menjadi rendah terhadap kecepatan proses eksekusi perintah. Beberapa keistimewaan dari AVR ATMEGA16 antara lain:

1. Mikrokontroler AVR 8 bit yang memiliki kemampuan tinggi dengan konsumsi daya rendah.
2. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16MHz.
3. Memiliki kapasitas *Flash* memori 16 Kbyte, EEPROM 512 Byte dan SRAM 1 Kbyte.
4. Saluran I/O sebanyak 32 buah, yaitu *Port A*, *Port B*, *Port C* dan *Port D*.
5. CPU yang terdiri dari 32 buah *register*.
6. Unit interupsi dan *eksternal*.
7. *Port* USART untuk komunikasi serial.
8. Fitur *peripheral*
 - Tiga buah *Timer/Counter* dengan kemampuan perbandingan (*compare*).
 - Dua buah *Timer/Counter* 8 bit dengan *Prescaler* terpisah dan *ModeCompar*.

- Satu buah *Timer/Counter* 16 bit dengan *Prescaler* terpisah, *Mode Compare* dan *Mode Capture*.
 - Real Time Counter* dengan *Oscillator* tersendiri.
 - Empat kanal PWM.
 - 8 kanal ADC.
 - 8 *Single-ended Channel* dengan keluaran hasil konversi 8 dan 10 resolusi (register ADCH dan ADCL).
 - 7 *Diferrential Channel* hanya pada kemasan *Thin Quad Flat Pack* (TQFP).
 - 2 *Differential Channel* dengan *Programmable Gain*.
 - Antarmuka *Serial Peripheral Interface* (SPI) Bus.
 - Watchdog Timer* dengan *Oscillator Internal* 9.
 - On-chip Analog Comparator*.
9. *Non-volatile program memory*.

2.1.1 Konfigurasi Pin AVR ATMEGA16



Gambar 2.2 Konfigurasi Kaki (*pin*) ATMEGA16

Konfigurasi *pin* ATMEGA16 dengan kemasan 40 *pin Dual In-line Package* (DIP) dapat dilihat pada Gambar 2.13. dari gambar diatas dapat dijelaskan fungsi dari masing-masing *pin* ATMEGA16 sebagai berikut.

1. VCC merupakan *pin* yang berfungsi sebagai masukan catu daya.
2. GND merupakan *pin Ground*.
3. *Port A* (PA0 – PA7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* masukan ADC 10.
4. *Port B* (PB0 – PB7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus, seperti dapat dilihat pada tabel dibawah ini.

Tabel 2.1 Fungsi Khusus *Port B*

Pin	Pin Fungsi Khusus
PB0	XCK (USART External Clock Input/Output) T0 (Timer/Counter0 External Counter Input)
PB1	T1 (Timer/Counter1 External Counter Input)
PB2	INT2 (External Interupt 2 Input) AIN0 (Analaog Comparator Negative Input)
PB3	OC0 (Timer/Counter0 Output Compare Macth Output) AIN1 (Analaog Comparator Negative Input)
PB4	(SPI Slave Select Input)
PB5	MOSI (SPI Bus Master Output /Slave Input)
PB6	MISO (SPI Bus Master Input/Slave Output)
PB7	SCK (SPI Bus Serial Clock)

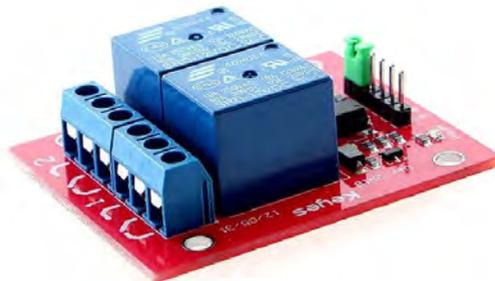
5. *Port A* (PC0 – PC7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus,
PC0 - SCL (*Two-wire Serial Bus Clock Line*)
PC1 - SDA (*Two-wire Serial BusData Input/Output Line*)
PC2 - TCK (*Joint Test Action Group Test Clock*)
PC3 - TMS (*JTAG Test Mode Select*)
PC4 - TDO (*JTAG Data Out*)
PC5 - TDI (*JTAG Test Data In*)
PC6 - TOSC1 (*Timer Oscillator pin 1*)
PC7 - TOSC2 (*Timer Oscillator pin 2*)
6. *Port D* (PD0 – PD7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus.
PD0 - RXD (USART Input Pin)

PD1 - TXD (USART *Output Pin*)
PD2 - INT0 (*External Interrupt 0 Input*)
PD3 - INT1 (*External Interrupt 1 Input*)
PD4 - OC1B (*Timer/Counter1 Output Compare B Match Output*)
PD5 - OC1A (*Timer/Counter1 Output Compare A Match Output*)
PD6 - ICP (*Timer/Counter1 Input Capture Pin*)
PD7 - OC2 (*Timer/Counter2 Output Compare Match Output*)

7. *RESET* merupakan *pin* yang digunakan untuk me-reset mikrokontroler.
8. XTAL1 dan XTAL2, merupakan *pin masukan external clock*.
9. AVCC merupakan *pin masukan tegangan* untuk ADC.
10. AREF merupakan *pin masukan tegangan referensi* untuk ADC.

2.2 Driver Motor

Pada alat ini menggunakan driver motor *relay 2 channel* Rangkaian *relay* pada sistem perancangan alat kali ini digunakan sebagai komponen keluaran dari sistem. *Relay* disini akan aktif jika sms telah sesuai dengan format yang ditentukan sesuai dengan sampel yang disimpan di memori *database*. *Relay* ini dihubungkan dari *microcontroller* menuju motor. Rangkaian ini menggunakan dua buah *relay*.



Gambar 2.3 *Relay 2 Channel*

Relay adalah rangkaian elektronika yang dapat digunakan untuk mengendalikan sesuatu dari jarak jauh. *Relay* sendiri merupakan

saklar magnetis yang paling sering digunakan pada setiap rangkaian elektronika. Dalam dunia elektronika, *relay* sangat berperan penting dalam suatu rangkaian karena dapat mengontrol rangkaian beban arus tinggi dengan arus yang rendah.

Dengan menggunakan rangkaian *relay*, kita juga tidak perlu mengontrol ataupun mengoperasikan suatu perangkat karena dapat dilakukan dari jarak jauh sehingga nantinya dapat mempermudah dan memperlancar pekerjaan. Biasanya, rangkaian *relay* terdapat pada televisi, *transmitter*, *sound system*, rangkaian bel, lampu taman otomatis, pengendali peralatan listrik dan perangkat elektronika lainnya.

2.3 Modem Wavecom



Gambar 2.4 Modem Wavecom

Wavecom adalah pabrikan asal Perancis (bermarkas di kota Issy les Moulineaux, Perancis) yaitu Wavecom SA yang berdiri sejak 1993 bermula sebagai biro konsultan teknologi dan sistim jaringan nirkabel GSM, dan pada 1996 Wavecom mulai membuat desain daripada modul wireless GSM pertamanya dan diresmikan pada 1997, bentuk modul GSM pertama berbasis GSM dan pengkodean khusus yang disebut AT Command. Sulit mencari referensi module tipe apa yang pertama dibuat oleh Wavecom SA.

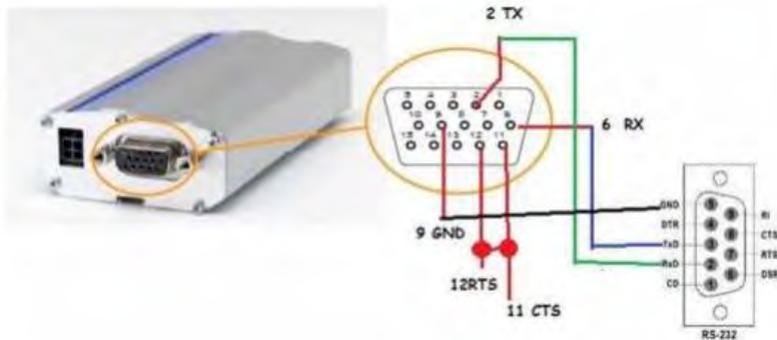
Beberapa kegunaan *modem wavecom*:

1. *SMS Broadcast application*

2. SMS Quiz application
3. SMS Polling
4. SMS auto-reply
5. M2M integration
6. Aplikasi Server Pulsa
7. Telemetri
8. Payment Point Data
9. PPOB

Modem ini menggunakan komunikasi *serial* dalam pengaplikasiannya. Komunikasi *serial* merupakan komunikasi data dengan pengiriman data secara satu per satu pada waktu tertentu. Sehingga komunikasi data serial hanya menggunakan dua kabel yaitu kabel data untuk pengiriman yang disebut *transmit* (TX) dan kabel data untuk penerimaan yang disebut *receive* (RX). Kelebihan dari komunikasi *serial* adalah jarak pengiriman dan penerimaan dapat dilakukan dalam jarak yang cukup jauh dibandingkan dengan komunikasi paralel tetapi kekurangannya kecepatannya lebih lambat dibandingkan komunikasi paralel. Dikenal dua cara komunikasi data secara serial, yaitu komunikasi data secara *sinkron* dan komunikasi data secara *asinkron*. Pada komunikasi data serial *sinkron*, *clock* dikirimkan bersama-sama dengan data serial, sedangkan komunikasi data serial *asinkron*, *clock* tidak dikirimkan bersama data *serial*, tetapi dibangkitkan secara sendiri-sendiri baik pada sisi pengirim (*transmitter*) maupun pada sisi penerima (*receiver*).

komunikasi antara *modem wavecom* dengan mikrokontroler yang digunakan adalah komunikasi serial secara asinkron yang bersifat *full-duplex*, artinya port serial bisa mengirim dan menerima pada waktu yang bersamaan. Perangkat yang digunakan yaitu kabel komunikasi *serial RS232* yang biasa digunakan untuk menghubungkan perifer eksternal seperti *modem* dengan komputer. *Modem* memiliki level tegangan yang berbeda dengan level tegangan TTL ataupun *RS232*, tetapi untuk kompatibilitas *modem* agar bisa terkoneksi dengan *PC* guna berbagai keperluan maka disediakan kabel data yang *compatible* dengan standar *RS232* sebagai *interface* untuk koneksi ke *PC*, untuk konfigurasi *port* data *modem* yang digunakan yaitu *wavecom m1306b*. Dengan alasan inilah maka digunakan komunikasi serial standar *RS232* sebagai dasar *interface* antara *modem* dengan mikrokontroler.



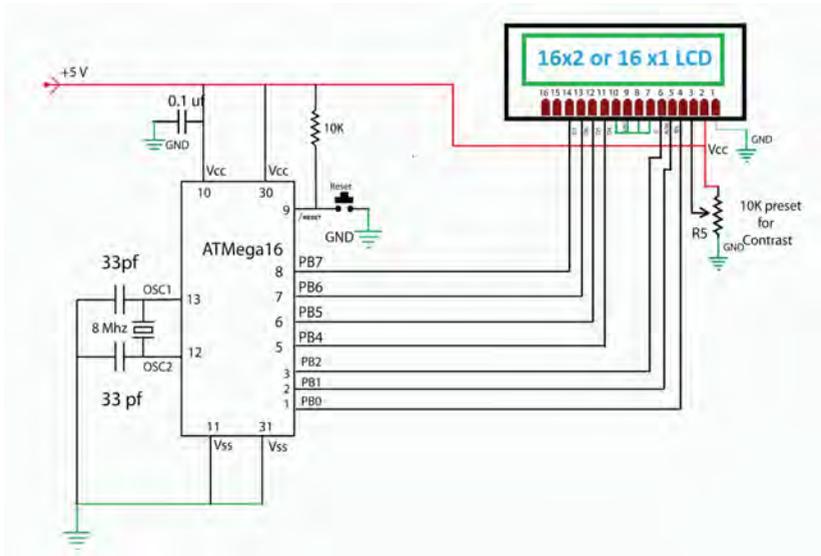
Gambar 2.5 Konfigurasi *Pin Modem Wavecom*

2.4 LCD 2x16

Display elektronik adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD (*Liquid Cristal Display*) adalah salah satu jenis display elektronik yang dibuat dengan teknologi CMOS *logic* yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap *front-lit* atau mentransmisikan cahaya dari *back-lit*. LCD (*Liquid Cristal Display*) berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik.

Material LCD (*Liquid Cristal Display*) LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan *seven-segment* dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. Lapisan sandwich memiliki *polarizer* cahaya vertikal depan dan *polarizer* cahaya horisontal belakang yang diikuti dengan lapisan reflektor. Cahaya yang dipantulkan tidak dapat melewati molekul-molekul yang telah menyesuaikan diri dan segmen yang diaktifkan terlihat menjadi gelap dan membentuk karakter data yang ingin ditampilkan.

Adapun konfigurasi LCD dengan AT Mega 16 adalah sebagai berikut:



Gambar 2.6 Konfigurasi *LCD*

Rangkaian ini sesuai dengan konfigurasi *LCD port* di CodeVisionAVR pada saat inialisasi dan pengaturan di CodeWizardAVR. Sedangkan rangkaian lainnya merupakan sistem minimum mikrokontroler pada umumnya.



Gambar 2.7 LCD 2x16

Pengendali / Kontroler LCD (*Liquid Cristal Display*) dalam modul LCD (*Liquid Cristal Display*) terdapat *microcontroller* yang berfungsi sebagai pengendali tampilan karakter LCD (*Liquid Cristal Display*). *Microntroller* pada suatu LCD (*Liquid Cristal Display*) dilengkapi dengan memori dan *register*. Memori yang digunakan *microcontroler internal LCD* adalah : *DDRAM (Display Data Random Access Memory)* merupakan memori tempat karakter yang akan ditampilkan berada. *CGRAM (Character Generator Random Access Memory)* merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai dengan keinginan. *CGROM (Character Generator Read Only Memory)* merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut merupakan karakter dasar yang sudah ditentukan secara permanen oleh pabrikan pembuat LCD (*Liquid Cristal Display*) tersebut sehingga pengguna tinggal mengambilnya sesuai alamat memorinya dan tidak dapat merubah karakter dasar yang ada dalam *CGROM*.

Register control yang terdapat dalam suatu LCD diantaranya adalah register perintah yaitu register yang berisi perintah-perintah dari mikrokontroler ke panel LCD (*Liquid Cristal Display*) pada saat proses

penulisan data atau tempat status dari panel *LCD (Liquid Cristal Display)* dapat dibaca pada saat pembacaan data. *Register* data yaitu *register* untuk menuliskan atau membaca data dari atau ke *DDRAM*. Penulisan data pada *register* akan menempatkan data tersebut ke *DDRAM* sesuai dengan alamat yang telah diatur sebelumnya. Pin, kaki atau jalur input dan kontrol dalam suatu *LCD (Liquid Cristal Display)* diantaranya adalah: Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan *LCD (Liquid Cristal Display)* dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit. Pin *RS (Register Select)* berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika *low* menunjukkan yang masuk adalah perintah, sedangkan logika *high* menunjukkan data. Pin *R/W (Read Write)* berfungsi sebagai instruksi pada modul jika *low* tulis data, sedangkan *high* baca data. Pin *E (Enable)* digunakan untuk memegang data baik masuk atau keluar. Pin *VLCD* berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 Kohm, jika tidak digunakan dihubungkan ke *ground*, sedangkan tegangan catu daya ke *LCD* sebesar 5 Volt.

2.5 Motor DC Gearbox



Gambar 2.8 Motor DC dengan Gearbox

Konsumsi arus saat berputar tanpa beban sebesar 190 mA (maksimum 250 mA), pada kondisi tanpa beban/hambatan kecepatan putar sekitar 90 rpm. Torsi maksimum 800 gf.cm. *Gearbox* ini memiliki dua sumbu (dual axis) dengan ujung datar (flat axle 1.9mm). Sumbu ganda ini berguna untuk memasangkan roda pada ujung yang satu dan

speed encoder pada sisi seberangnya dengan demikian mikrokontroler dapat mengukur kecepatan aktual dengan menggunakan *komponen opto-interrupter* semacam H2210.

2.6 Adaptor DC 12V



Gambar 2.9 Adaptor DC 12V

Adaptor adalah sebuah rangkaian elektronika yang dapat mengubah tegangan AC menjadi DC. Rangkaian ini adalah alternatif pengganti dari sumber tegangan DC, misalnya baterai dan *accumulator*. Keuntungan dari *adaptor* dibanding dengan baterai atau *accumulator* adalah sangat praktis berhubungan dengan ketersediaan tegangan karena *adaptor* dapat di ambil dari sumber tegangan AC yang ada di rumah, di mana pada jaman sekarang ini setiap rumah sudah menggunakan listrik. Selain itu, *adaptor* mempunyai jangka waktu yang tidak terbatas asal ada tegangan AC, tegangan AC ini sudah merupakan kebutuhan primer dalam kehidupan manusia.

Arus listrik yang kita gunakan di rumah, kantor dan pabrik pada umumnya adalah dibangkitkan, dikirim dan didistribusikan ke tempat masing-masing dalam bentuk Arus Bolak-balik atau arus AC (*Alternating Current*). Hal ini dikarenakan pembangkitan dan pendistribusian arus listrik melalui bentuk arus bolak-balik (AC) merupakan cara yang paling ekonomis dibandingkan dalam bentuk arus searah atau arus DC (*Direct Current*).

Akan tetapi, peralatan elektronika yang kita gunakan sekarang ini sebagian besar membutuhkan arus DC dengan tegangan yang lebih rendah untuk pengoperasiannya. Oleh karena itu, hampir setiap peralatan elektronika memiliki sebuah rangkaian yang berfungsi untuk melakukan konversi arus listrik dari arus AC menjadi arus DC dan juga untuk menyediakan tegangan yang sesuai dengan rangkaian Elektronika-nya. Rangkaian yang mengubah arus listrik AC menjadi DC ini disebut dengan *DC Power Supply* atau dalam bahasa Indonesia disebut dengan *Catu daya DC*. *DC Power Supply* atau *Catu Daya* ini juga sering dikenal dengan nama "*Adaptor*".

Sebuah *DC Power Supply* atau *adaptor* pada dasarnya memiliki 4 bagian utama agar dapat menghasilkan arus DC yang stabil. Keempat bagian utama tersebut diantaranya adalah *Transformer*, *Rectifier*, *Filter* dan *Voltage Regulator*.

Prinsip Kerja DC Power Supply (*Adaptor*) yaitu transformator (*Transformer*) atau disingkat dengan *Trafo* yang digunakan untuk *DC Power supply* adalah Transformator jenis *step-down* yang berfungsi untuk menurunkan tegangan listrik sesuai dengan kebutuhan komponen *elektronika* yang terdapat pada rangkaian *adaptor (DC Power Supply)*. Transformator bekerja berdasarkan prinsip induksi elektromagnetik yang terdiri dari 2 bagian utama yang berbentuk lilitan yaitu lilitan *primer* dan lilitan sekunder. Lilitan *primer* merupakan *input* dari pada Transformator sedangkan *output*-nya adalah pada lilitan sekunder. Meskipun tegangan telah diturunkan, *output* dari Transformator masih berbentuk arus bolak-balik (arus AC) yang harus diproses selanjutnya.

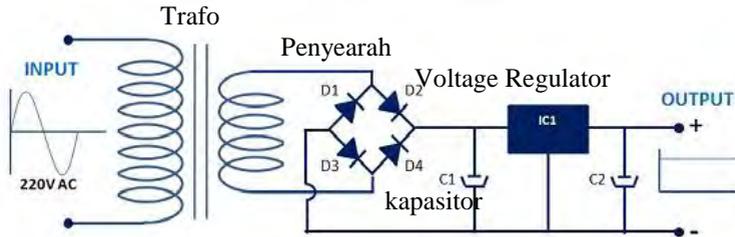
Rectifier atau penyearah gelombang adalah rangkaian Elektronika dalam Power Supply (catu daya) yang berfungsi untuk mengubah gelombang AC menjadi gelombang DC setelah tegangannya diturunkan oleh *transformator step down*. Rangkaian *rectifier* biasanya terdiri dari komponen *dioda*. Terdapat 2 jenis rangkaian *rectifier* yaitu "*Half Wave Rectifier*" yang hanya terdiri dari 1 komponen dioda dan "*Full Wave Rectifier*" yang terdiri dari 2 atau 4 komponen dioda.

Dalam rangkaian *power supply (Adaptor)*, *filter* digunakan untuk meratakan sinyal arus yang keluar dari *rectifier*. *Filter* ini biasanya terdiri dari komponen kapasitor (Kondensator) yang berjenis Elektrolit atau ELCO (Electrolyte Capacitor).

Untuk menghasilkan Tegangan dan Arus DC (arus searah) yang tetap dan stabil, diperlukan *Voltage Regulator* yang berfungsi untuk

mengatur tegangan sehingga tegangan *output* tidak dipengaruhi oleh suhu, arus beban dan juga tegangan input yang berasal *output filter*. *Voltage Regulator* pada umumnya terdiri dari *Dioda Zener*, *Transistor* atau *IC (Integrated Circuit)*.

Pada *DC power supply* yang canggih, biasanya *voltage regulator* juga dilengkapi dengan *short circuit protection* (perlindungan atas hubung singkat), *current limiting* (Pembatas Arus) ataupun *over voltage protection* (perlindungan atas kelebihan tegangan).



Gambar 2.10 Rangkaian *Adaptor DC 12V*

BAB III
PERANCANGAN DAN PEMBUATAN
ALAT

BAB III PERANCANGAN DAN PEMBUATAN ALAT

Bab ini membahas perancangan sistem yang meliputi perancangan *hardware*, *software* dan perancangan mekanik. Detail mengenai pembahasan perancangan akan dibahas pada beberapa subbab berikut.

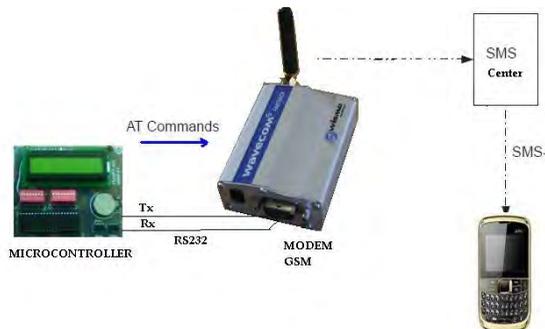
3.1 Deskripsi Kerja Alat

Cara kerja alat ini setelah ATMega16 akan menerima sumber tegangan dari adaptor. Setelah itu, maka *modem* akan melakukan pengiriman sms dengan format “READY” kepada nomor yang telah disimpan pada *microcontroller* sebelumnya, maka alat siap digunakan. Pengguna hanya tinggal membalas pesan dengan format: “P1”, “P2”, atau “P3”. Setelah pengiriman sms tersebut, dengan jeda 1 hingga 2 detik (saat sms berhasil masuk pada modem), pintu bagian bawah tempat penyimpanan pakan akan terbuka sesuai dengan perintah yang dikimkan. Perintah “P1” untuk membuka pintu selama 1 detik, “P2” untuk 2 detik, dan perintah “P3” untuk membuka selama 3 detik.

3.2 Perancangan Perangkat Keras (*Hardware*)

Perancangan perangkat keras pada alat terdiri dari perancangan rangkaian *microcontroller*, *modem wavecom* dan perancangan rangkaian *driver motor*.

Blok diagram dari keseluruhan perancangan dan pembuatan alat pemberi pakan pada kandang ini dengan menggunakan sms yang di desain pada tugas akhir ini terlihat seperti gambar 3.1



Gambar 3.1 Diagram Alat Secara Keseluruhan

Proses kerja rangkaian *relay* ini relay akan dihubungkan dengan rangkaian *microcontroller ATmega16*. Dengan kondisi ini maka *relay* motor mendapatkan tegangan positif (+) dari *relay* lainnya dan mendapatkan tegangan negatif (-) sehingga *relay* akan membentuk *induksi magnet* didalam kumparan dan akan mengaktifkan motor pada pembuka makanan.

Perancangan *hardware* dilakukan dengan merancang rangkaian-rangkaian elektronika dan rancangan mekanik. Perancangan *hardware* ini meliputi:

1. Rangkaian *Microcontroller*
2. Rangkaian *driver relay*

Gambaran secara keseluruhan menjelaskan tentang perancangan sms dengan *ATmega16*. Perancangan tugas akhir ini menggunakan beberapa *hardware*. Dimulai dari *PC* yang digunakan sebagai media perancangan sistem pada program yang akan digunakan oleh *mikrokontroler ATmega16*. Setelah itu *mikrokontroler ATmega16* yang akan mengatur berjalannya sistem antara modem dengan *relay* sebagai kontaktor yang akan mengaktifkan *motor dc*. Pada peralatan ini digunakan dua buah *relay* yang memiliki fungsi berbeda. *Relay* satu berfungsi mengaktifkan *motor dc* agar ketika kontak *relay* aktif maka motor juga akan selalu pada keadaan *standby*. Ini bertujuan agar ketika *relay* satu berada pada kondisi '1' maka *motor* selalu berada pada kondisi 'on' hingga tegangan yang di berikan oleh *power supply* di putus. Sedangkan *relay* dua yang dihubungkan dengan *starter* digunakan sebagai pengatur sistem *motor*. Pengaturannya adalah ketika sms yang masuk sesuai dengan *sms* sampel yang telah disimpan sebelumnya maka *mikrokontroler ATmega16* akan mengaktifkan *relay* dua pada kondisi logika '1' maka motor secara otomatis akan membuka pintu pakan dan menutupnya kembali sesuai dengan perintah sms yang dikirimkan.

3.2.1 Perancangan Rangkaian Relay

Sesuai dengan kegunaan *relay* yang dapat berfungsi sebagai *driver motor* maka disini *relay* akan menerima input dari *microcontroller ATmega16*. Dan akan diteruskan untuk mengontrol *motor dc* dengan *gearbox* sebagai pembuka dan penutup pintu pakan.



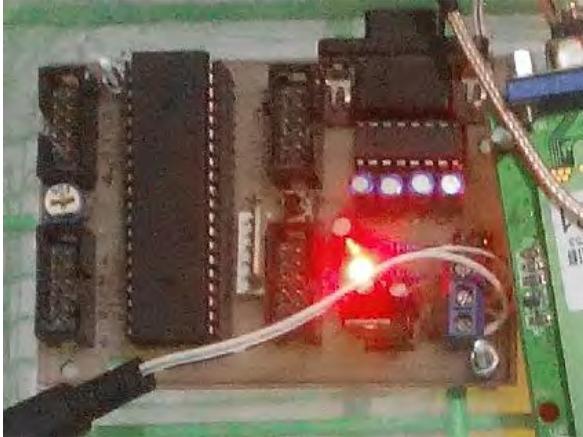
Gambar 3.2 Posisi Peletakan *Relay* di bawah *Modem Wavecom*



Gambar 3.3 Pemberian *Input Power*

3.2.2 Modul *Microcontroller* ATmega16

Perangkat utama yang digunakan oleh perancangan alat ini adalah mikrokontroler ATmega16 yang digunakan sebagai media komunikasi antara *modem wavecom* dengan *relay* sehingga *motor* dapat membuka dan menutup pintu pakan melalui sms. Jadi ATmega16 inilah yang mengatur berjalannya sistem pada perancangan alat ini.



Gambar 3.4 Modul ATmega16

3.3 Perancangan *Software*

Perancangan perangkat lunak (*software*) pada perancangan dan pembuatan alat ini meliputi pembuatan program pengenalan sms pengguna dan sebuah program *LCD* 2x16 dengan menggunakan ATmega16. Perancangan ini menggunakan *software* “cvavr”.

```
D:\Data\pakan kucing\KUC1.c
Notes KUC1.c KUC1.asm KUC1.lst
31 r jg,data,a1,a2;
32 d baca_sms()
33
34 data=getchar(); // baca data masuk persatu karakter
35 while(data!='+') // mencari +
36 {data=getchar();}
37
38     if (data=='+')
39     {
40         data=getchar();
41         if (data=='C')
42         {
43             data=getchar();
44             if (data=='M')
45             {
46                 data=getchar();
47                 if (data=='T')
48                 {
49                     {
50                         data=getchar();
51                         while (data!='/') // cari /
52                         {
53                             data=getchar();
54                             }
55                         while (data!='+') // cari petik "
```

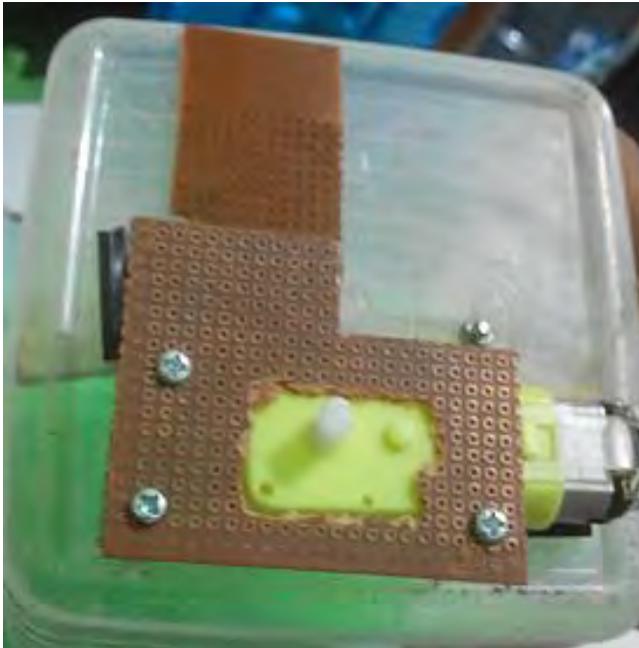
Gambar 3.5 Perancangan *Software*



Gambar 3.6 Penginputan *Software* pada *Microcontroller*

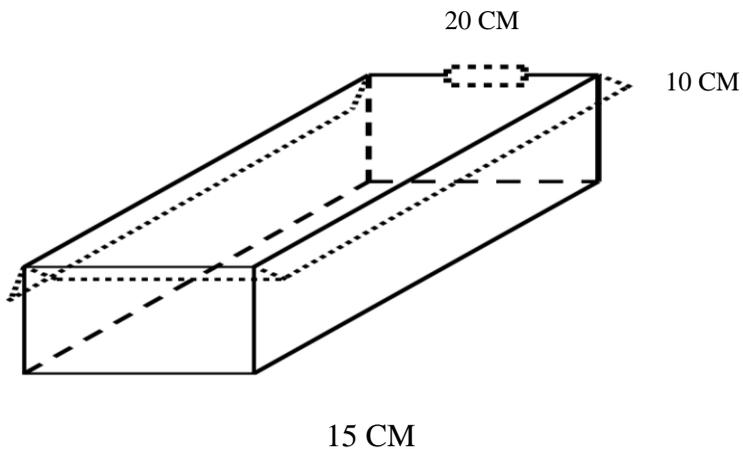
3.4 Perancangan Mekanik

Mekanik pintu pembuka makanan terdiri dari sebuah *motor* yang di tempatkan di bawah penyimpanan pakan. Untuk pintunya terbuat dari bahan plastik atau akrilik atau sesuai dengan kebutuhan yang akan disambung dengan *motor* dc yang sudah disambung pada *relay*.



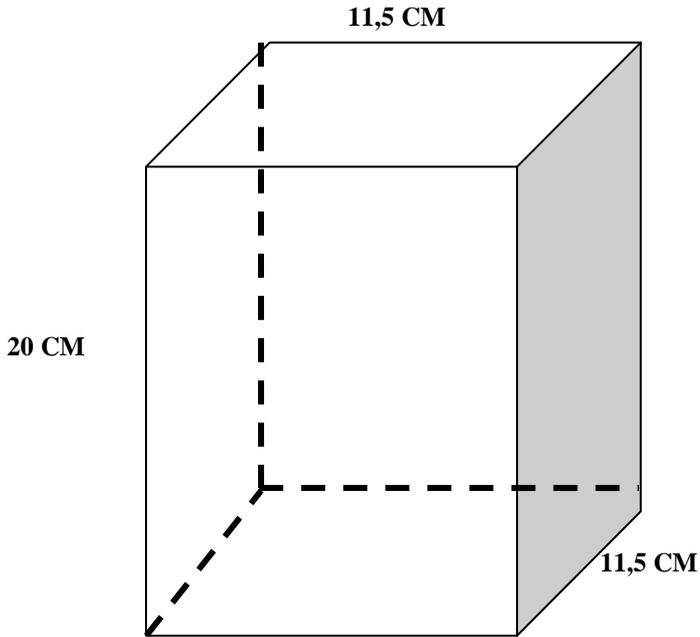
Gambar 3.7 Mekanik Pintu Pakan

Dibawah ini rancangan mekanik dibuat sebuah kotak yang digunakan sebagai tempat penyimpanan rangkaian *microcontroller*, *driver motor*, dan *modem wavecom* agar rangkaian dapat digunakan. Berikut pada Gambar 3.9 merupakan ukuran *box* yang digunakan *microcontroller*.



Gambar 3.8 Rancangan Kotak Penyimpanan *Microcontroller*, *Driver Motor*, dan *Modem Wavecom*

Dibawah ini pada gambar 3.10 adalah rancangan mekanik sebuah kotak yang terbuat dari *tupperware*. Fungsinya sebagai tempat penyimpanan pakan dari peralatan tugas akhir.



Gambar 3.9 Perancangan Kotak Penyimpanan Pakan

BAB IV
PENGUJIAN DAN ANALISA DATA

BAB IV

PENGUJIAN DAN ANALISA DATA

Untuk mengetahui apakah tujuan-tujuan dari pembuatan alat ini telah tercapai atau belum, maka perlu dilakukannya sebuah pengujian dan analisa terhadap alat yang telah dibuat. Dan sebagai acuan yang tidak terpisahkan adalah adanya proses *evaluasi* sehingga akan dapat dilakukan langkah-langkah positif guna membawa alat ini kearah yang lebih baik. Hasil pengujian tersebut akan dianalisa untuk mengetahui penyebab terjadinya kekurangan dalam sistem. Pengujian tersebut meliputi:

1. Pengujian Perangkat Keras (*Hardware*)
 - a. Mikrokontroller ATMega16
 - b. *Adaptor*
 - c. Rangkaian *Relay*
2. Pengujian Perangkat Lunak (*Software*)
 - a. Pengujian Program SMS
3. Pengujian Alat secara Keseluruhan

4.1 Pengujian Perangkat Keras

Perangkat keras (*hardware*) yang dipakai pada tugas akhir kali ini meliputi rangkaian ATMega16, *adaptor* dan rangkaian *relay*. Berikut adalah pembahasan tentang pengujian *mikrokontroller* Atmega16, *adaptor* dan rangkaian *relay* yang dipakai pada perancangan dan pembuatan alat pemberian pakan pada kandang kucing via sms

4.1.1 Pengujian Pada Mikrokontroller ATMega16

Pada perancangan tugas akhir kali ini pengatur sistemnya menggunakan mikrokontroller ATMega16. Pengujiannya dilakukan dengan mengukur keluaran dari ATMega16 dalam keadaan pada logika 0 (*low*) dan pada logika 1 (*high*). Berikut hasil pengujian pada mikrokontroller ATMega16:

- Jika ATMega16 berada pada keadaan logika 0 (*low*) maka keluaran menghasilkan nilai 0 *Volt*.
- Jika ATMega16 berada pada keadaan logika 1 (*high*) maka keluaran menghasilkan nilai 4,83 *Volt*.

4.1.2 Pengujian *Adaptor*

Pada perancangan tugas akhir kali ini *adaptor* digunakan sebagai sumber tegangan yang akan diberikan pada mikrokontroler ATmega16. Rangkaian *adaptor* berikut bertugas menurunkan tegangan AC 220V menjadi DC 12V. Jadi, pada kesimpulannya adaptor berikut mempunyai tegangan keluaran sebesar 12V. Yang akan diuji pada *adaptor* berikut adalah besar tegangan masukan pada *adaptor* dan tegangan keluaran pada *adaptor*.

Berikut pada tabel 4.1 adalah tabel pengujian *adaptor* pada perancangan alat :

Pengukuran Ke -	Tegangan Masukan	Tegangan Keluaran	
	Vin (Volt)	Vout Tanpa Beban	Vout Dengan Beban
1	220V	11,95V	11,90 V
2	220V	11,95 V	11,90 V
3	220 V	11,95 V	11,90 V
4	220V	11,95 V	11,90 V
5	220 V	11,95 V	11,90 V
V rata-rata	220V	11,95 V	11,90 V

Tabel 4.1 Pengujian *Adaptor*

Pengujian tanpa beban dilakukan dengan mengukur keluaran dengan *Volt* meter.

Dibawah ini dilihat pada gambar 4.1 adalah bentuk fisik dari *adaptor* yang digunakan pada perancangan alat pemberian pakan via sms.



Gambar 4.1 Bentuk Fisik *Adaptor* DC 12V

4.1.3 Pengujian Rangkaian *Relay*

Rangkaian yang dipakai pada perancangan tugas akhir ini digunakan sebagai *output* yang bertugas untuk mengaktifkan *motor DC*. *Relay* yang dipakai yaitu *relay 2 channel*. Pengujian pada *relay* dilakukan pada waktu kerja *relay* setelah alat dijalankan.

Berikut pada tabel 4.2 hasil dari pengujian rangkaian *relay* pada perancangan alat:

Pengujian Ke -	Waktu Kerja <i>Relay</i>
1	1,36 detik
2	1,43 detik
3	1,53 detik
4	1,29 detik
5	1,50 detik
6	1,39 detik
7	1,29 detik
8	1,34 detik
9	1,28 detik
10	1,25 detik
Rata – Rata	1,366 detik

Tabel 4.2 Pengujian Rangkaian *Relay*

Dibawah ini dapat dilihat pada gambar 4.2 adalah bentuk fisik dari rangkaian *relay* yang digunakan pada perancangan dan pembuatan alat pemberian pakan dengan menggunakan sms:



Gambar 4.2 Bentuk Fisik Rangkaian *Relay*

4.2 Pengujian Perangkat Lunak (*Software*)

Perangkat lunak (*software*) yang digunakan pada perancangan dan pembuatan pemberian Pakan Kucing Via Sms adalah program CYAVR. Program yang dirancang meliputi program *delay* (lama waktu terbukanya pintu pakan) yaitu berupa: “P1, P2, atau P3”.

4.2.1 Pengujian Program

Perancangan program berikutnya adalah sms pada alat dan. Hal ini bertujuan agar ketika sms yang dikirim pengguna sesuai dengan sampel sms yang telah tersimpan dalam memori, seketika *motor* dapat aktif secara otomatis. Dikatakan berhasil ketika alat diberi inputan power kemudian alat akan mengirim sms kepada nomor yang telah disimpan dengan format “*READY*”. Kemudian alat akan merespon ketika pengguna membalas dengan salah satu format yaitu “P1, P2, atau P3” dan alat merespon dengan membuak pakan maka pengujian ketepatan sms dapat bekerja dengan baik. Artinya, ketika sms yang dikirim tidak

sesuai dengan sampel yang disimpan maka *relay* tidak akan mengaktifkan *motor*, namun sebaliknya ketika sms sesuai dengan sampel yang telah disimpan sebelumnya maka *motor* akan segera aktif secara otomatis. Digunakan beberapa *variabel* pengujian dalam pengujian program kali ini, diantaranya keberhasilan program sampel sms, serta waktu yang diperlukan dalam proses autentifikasi sensor.

Berikut ini pada tabel 4.3, 4.4 dan 4.5 adalah tabel hasil pengujian dari program sms pada alat:

Pengujian Ke -	Waktu Kerja Starter	Tingkat keberhasilan
1	1,69 detik	Gagal
2	1,82 detik	Berhasil
3	1,31 detik	Berhasil
4	1,48 detik	Berhasil
5	1,48 detik	Berhasil
6	1,25 detik	Berhasil
7	1,67 detik	Berhasil
8	1,72 detik	Berhasil
9	1,58 detik	Berhasil
10	1,38 detik	Berhasil
Rata – Rata	1,538 detik	10 % gagal 90% berhasil

Tabel 4.3 Pengujian dari program P1 dengan *Delay* 1000ms

Pengujian Ke -	Waktu Kerja Starter	Tingkat keberhasilan
1	1,32 detik	Gagal
2	1,46 detik	Gagal
3	1,48 detik	Gagal
4	1,57 detik	Berhasil
5	1,36 detik	Berhasil
6	1,52 detik	Berhasil
7	1,36 detik	Berhasil
8	1,42 detik	Berhasil
9	1,64 detik	Berhasil
10	1,53 detik	Berhasil
Rata – Rata	1,466 detik	30 % gagal 70% berhasil

Tabel 4.4 Pengujian dari Program P2 dengan *Delay* 2000ms

Pengujian Ke -	Waktu Kerja Starter	Tingkat keberhasilan
1	1,34 detik	Gagal
2	1,42 detik	Gagal
3	1,36 detik	Berhasil
4	1,54 detik	Berhasil
5	1,63 detik	Berhasil
6	1,36 detik	Berhasil
7	1,44 detik	Berhasil
8	1,32 detik	Berhasil
9	1,62 detik	Berhasil
10	1,54 detik	Berhasil
Rata – Rata	1,457 detik	20 % gagal 80% berhasil

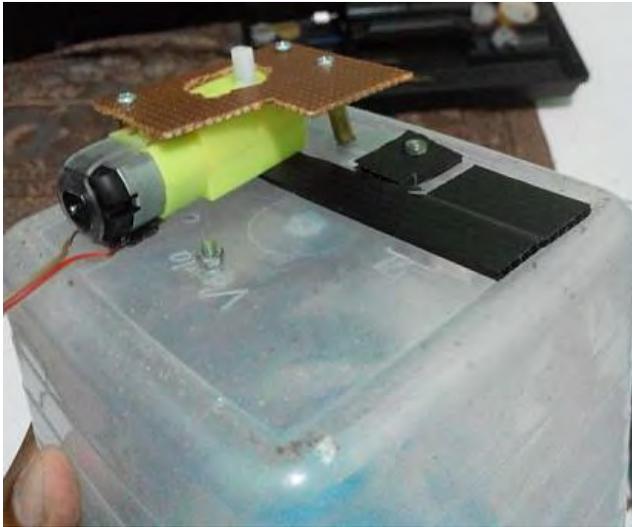
Tabel 4.5 Pengujian dari Program P3 dengan *Delay* 3000ms

4.3 Pengujian Alat Secara Keseluruhan

Pengujian alat secara keseluruhan dihasilkan beberapa kesimpulan yakni:

- Jika ATmega16 pada logika 0 maka *outputnya* bernilai 0V.
- Jika ATmega16 pada logika 1 maka *outputnya* bernilai 4,83V.
- Tegangan keluaran rata-rata pada *adaptor* tanpa beban adalah 11,95V dengan persentase kesalahan pada rangkaian sebesar 0,55 %
- Tegangan keluaran rata-rata pada *adaptor* dengan beban adalah 11,90V dengan persentase kesalahan pada rangkaian sebesar 1,11 %
- Rata-rata waktu kerja *relay* selama 1,366 *second*.
- Pada saat *delay* 1000ms maka rata-rata waktu kerja selama 1,538 *second* dengan tingkat keberhasilan 10% gagal dan 90% berhasil.
- Pada saat *delay* 2000 ms maka rata-rata waktu kerja selama 1,466 *second* dengan tingkat keberhasilan 30% gagal dan 70% berhasil.
- Pada saat *delay* 3000ms maka rata-rata waktu kerja selama 1,457 *second* dengan tingkat keberhasilan 20% gagal dan 80% berhasil.

Dibawah ini adalah gambar implementasi perancangan alat. Dapat dilihat pada gambar 4.3 merupakan penempatan *motor DC* sebagai pembuka pintu pakan. Dan pada gambar 4.4, 4.5, 4.6 dan 4.7 merupakan alat elektronik perancangan alat serta penempatan alat pada kandang.



Gambar 4.3 Penempatan *motor* DC sebagai pembuka pintu pakan



Gambar 4.4 Bentuk Perancangan Peralatan Elektronik



Gambar 4.5 Penempatan Peralatan Elektronik pada Cover



Gambar 4.6 Relay Motor



Gambar 4.7 Penempatan Alat pada Kandang

DAFTAR PUSTAKA

DAFTAR PUSTAKA

- [1], Pengetahuan Dasar ATmega16, <URL: <http://www.engineersgarage.com/electronic-components/atmega16-microcontroller/>> , Agustus 2012
- [2], Andrianto, Beri, 2008, Pemrograman Mikrokontroler AVR ATMEGA16 Menggunakan Bahasa C (Codevision AVR), Bandung, Informatika.Jakarta,2009.
- [3], Bejo, C dan AVR Rahasia Kemudahan Bahasa C dalam Mikrokontroler ATmega8535, Yogyakarta, Graha Ilmu. , Agus., 2008
- [4], ATmega 16 Datasheet, <URL : <http://www.alldatasheet.com/datasheet-pdf/pdf/78532/ATMEL/ATMEGA16.html>, 1 Agustus 2013
- [5], Teori Relay Elektromekanik, <URL : <http://elektronika-dasar.web.id/teori-elektronika/teori-relay-elektro-mekanik/>>, 1 Agustus 2013
- [6], Teori Modem Wavecom, <URL: <http://fungsimodemwavecom.tumblr.com/>> , Juni 2015
- [7], Sensor Arus Listrik ACS712, <URL : depokinstruments.com/2012/03/29/sensor-arus-listrik-ac712/>, 2 Agustus 2013

Halaman ini Sengaja dikosongkan

LAMPIRAN

LAMPIRAN A

LISTING PROGRAM

Program SMS Microcontroller ATmega16

This program was produced by the
CodeWizardAVR V2.04.4a Advanced
Automatic Program Generator
© Copyright 1998-2009 Pavel Haiduc, HP InfoTech s.r.l.
<http://www.hpinfotech.com>

Project :
Version :
Date : 4/30/2015
Author : NeVaDa
Company : metrocom
Comments:

Chip type : ATmega16
Program type : Application
AVR Core Clock frequency: 11.059200 MHz
Memory model : Small
External RAM size : 0
Data Stack size : 256

*****/

```
#include <mega16.h>
#include <stdlib.h>
#include <delay.h>
#include <stdio.h>
```

```
// Declare your global variables here
char jg,data,a1,a2;
void baca_sms()
{
    data=getchar(); // baca data masuk persatu karakter
    while(data!='+')// mencari +
```

```

{data=getchar();}

    if (data=='+')
    {
        data=getchar();
        if (data=='C')
        {
            data=getchar();
            if (data=='M')
            {
                data=getchar();
                while (data!='') // cari /
                {
                    data=getchar();
                }
                while (data!='+') // cari petik "
                {
                    data=getchar();
                }
                while (data!="") // cari petik "
                {
                    data=getchar();
                }
                jg=getchar(); // baca Spasi
                jg=getchar(); // baca enter
                a1=getchar(); // baca data SMS
                a2=getchar();

//=====>>>
                if (a1=='P' && a2=='1')
                {
                    PORTB.2=1;
                    delay_ms(25);
                    PORTB.2=0;

```

```

1          delay_ms(1000); //setting waktu buka pakan

          PORTB.3=1;
          delay_ms(20);
          PORTB.3=0;
          }

//=====>>>>
          if (a1=='P' && a2=='2')
          {
          PORTB.2=1;
          delay_ms(25);
          PORTB.2=0;
          delay_ms(2000); //setting waktu buka pakan
2
          PORTB.3=1;
          delay_ms(20);
          PORTB.3=0;
          }

//=====>>>>
          if (a1=='P' && a2=='3')
          {
          PORTB.2=1;
          delay_ms(25);
          PORTB.2=0;
          delay_ms(3000); //setting waktu buka pakan
3
          PORTB.3=1;
          delay_ms(20);
          PORTB.3=0;
          }

          printf("AT+CMGD=1"); // hapus SMS
          putchar(13);
          delay_ms(300);
          printf("AT+CMGD=2"); // hapus SMS

```

```

        putchar(13);
        delay_ms(300);

    }
}
}
}
//AT+CMT

}
void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTA=0x00;
DDRA=0x00;

// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=Out Func2=Out
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=0 State2=0 State1=T
State0=T
PORTB=0x00;
DDRB=0x0C;

// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTC=0x00;
DDRC=0x00;

```

```
// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTD=0x00;
DDRD=0x00;
```

```
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;
```

```
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
```

```

OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x47;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

```

```
//=====> let's start
delay_ms(10000);
printf("AT+CMGF=1");
putchar(13);
printf("AT+CNMI=1,2,0,0,0");
putchar(13);
putchar(10);
delay_ms(10);
printf ("AT+CMGS=");
putchar(34);
printf("085706670178");
putchar(34);
putchar(13);
delay_ms(10);
printf("READY");
putchar(13);
putchar(26);
putchar(13);
putchar(10);
delay_ms(1000);

while (1)
{
    baca_sms();
};
}
```

Halaman ini Sengaja dikosongkan

LAMPIRAN 2

DATASHEET

1. *Dataheet ModemWavecom*



Fastrack modem M12 series

Fastrack modem M1206 User Guide

Reference: WM_FFJ_M12_UGD_001
Revision: 002
Date: 18th September 2003



WAVECOM confidential ©

This document is the sole and exclusive property of WAVECOM. Not to be distributed or divulged without prior written agreement.

Ce document est la propriété exclusive de WAVECOM. Il ne peut être communiqué ou divulgué à des tiers sans son autorisation préalable.

Page : 1 / 38

1 General description

1.1 Presentation

FASTRACK M1206 modem is a self-contained E-GSM/GSM-GPRS 900/1800 dual-band modem and is GPRS class 10 capable.

This modem supports the following transmissions:

- Data,
- Fax,
- Short Messages (Point to point and Cell Broadcast),
- Voice calls.

The modem comprises several interfaces:

- LED function indicating the operating status,
- External antenna (via SMA connector),
- RS232 Serial and control link (via 15-pin SUB HD connector),
- Power supply (via 4-pin Micro-Fit™ connector),
- SIM card holder.

The main features of the modem are the following:

- 2 Watts E-GSM 900 radio section.
- 1 Watt GSM1800 radio section.
- 32 Mbits of Flash memory and 4 Mbits of SRAM,
- Real Time Clock with calendar.
- Echo Cancellation + noise reduction.
- Full GSM or GSM / GPRS software stack.
- Hardware GPRS class 10 capable.
- Complete shielding.
- A DC Power supply,
- A RS232 serial link,
- Audio interface for:
 - microphone,
 - speaker.
- A 3V / 5V SIM interface.

Modem mechanical case is made out of aluminium profile ended by two holding bridles at each extremity.

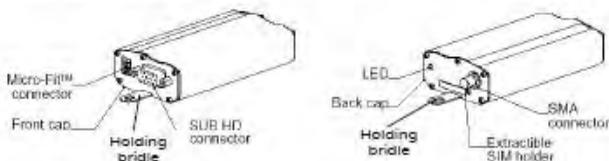


Figure 1: FASTRACK M1206 modem presentation

1.2 External connections

1.2.1 Connectors

1.2.1.1 General

FASTRACK M1206 modem has three external connections:

- Antenna connector: SMA connector for RF connection to the antenna,
- Sub D high density 15-pin connector for:
 - RS232 serial link connection,
 - Audio lines (microphone and speaker) connection,
 - BOOT and RESET signals connection.
- Power supply connector: 4-pin Micro FIT connector for DC Power Supply.

1.2.1.2 Antenna connector



Figure 2: Antenna connector

1.2.1.3 Sub HD 15-pin connector



Figure 8: Sub HD 15-pin connector

Pin #	Signal (CCITT / EIA)	I/O	I/O type	Description	Comment
1	CT109 / DCD	O	STANDARD RS232	RS232 Data Carrier Detect	
2	CT103 / TX	I	STANDARD RS232	RS232 Transmit serial data	
3	BOOT	I	CMOS	Boot	Active low. Pull down through 1K for Flash downloading
4	Microphone (+)	I	Analog	Microphone positive line	
5	Microphone (-)	I	Analog	Microphone negative line	
6	CT104 / RX	O	STANDARD RS232	RS232 Receive serial data	
7	CT107 / DSR	O	STANDARD RS232	RS232 Data Set Ready	
8	CT108-2 / DTR	I	STANDARD RS232	RS232 Data Terminal Ready	
9	GND	-	GND	Ground	
10	Speaker (+)	O	Analog	Speaker positive line	

Pin #	Signal (CCITT / EIA)	I/O	I/O type	Description	Comment
11	CT106 / CTS	O	STANDARD RS232	RS232 Clear To Send	
12	CT105 / RTS	I	STANDARD RS232	RS232 Request To Send	
13	CT123 / RI	O	STANDARD RS232	RS232 Ring Indicator	
14	RESET	I/O	Schmitt	Modem reset	Active low
15	Speaker [-]	O	Analog	Speaker negative line	

1.2.1.4 Power supply connector:



Figure 4: Power supply connector

Pin #	Signal	I/O	I/O type	Description	Comment
1	V+BATT	I	Power supply	Battery input	High current
2	GND		Power supply	Ground	
3	NC				Reserved
4	NC				Reserved

2 Functional description

2.1 Architecture

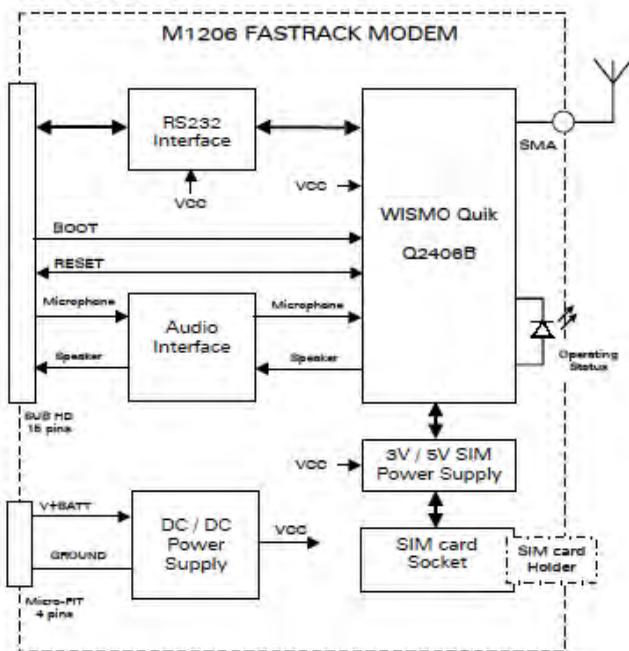


Figure 6: Functional architecture

2.2 RS232 serial link

2.2.1 General presentation

The RS232 interface performs the voltage level adaptation (V24/CMOS ↔ V24/V28) between the internal WISMO module (DCE) and the external world (DTE).

The RS232 interface is protected (by ESD protection) against electrostatic surges on the RS232 lines.

Filtering guarantees:

- EMI/RFI protection in input and output,
- Signal smoothing.

A flexible 6-wire serial interface is available complying with RS232 protocol signaling.

The signals available are:

- TX data (CT103/TX),
- RX data (CT104/RX),
- Request To Send (CT105/RTS),
- Clear To Send (CT106/CTS),
- Data Terminal Ready (CT108-2/DTR),
- Data Set Ready (CT107/DSR).

The 2 additional signals are Data Carrier Detect (CT109/DCD) and Ring Indicator (CT125/RI).

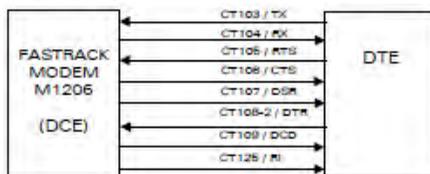


Figure 7: RS232 Serial Link signals

RS232 interface has been designed to allow a certain flexibility in the use of the serial interface signals. However, the use of TX, RX, CTS and RTS signals is mandatory which is not the case for DTR, DSR, DCD and RI signals which can be not used.

2.2.2 Pin out description

Pin description

Signal	Sub HD connector Pin number	I/O	I/O type RS232 STANDARD	Description
CT103/TX	2	I	TX	Transmit serial data
CT104/RX	6	O	RX	Receive serial data
CT105/RTS	12	I	RTS	Request To Send
CT106/CTS	11	O	CTS	Clear To Send
CT107/DSR	7	O	DSR	Data Set Ready
CT108-2/DTR	8	I	DTR	Data Terminal Ready
CT109/DCD	1	O	DCD	Data Carrier Detect
CT125/RI	13	O	RI	Ring Indicator
CT102/GND	9			Ground

2.3 RESET

2.3.1 General presentation

This signal is used to force a reset procedure by providing low level during at least 500 μ s.

This signal has to be considered as an emergency reset only. A reset procedure is automatically driven by an internal hardware during the power-up sequence.

This signal can also be used to provide a reset to an external device. It then behaves as an output. If no external reset is necessary this input can be left open, if used (emergency reset), it has to be driven by an open collector or an open drain output:

- RESET pin 14 = 0, for Modem Reset,
- RESET pin 14 = 1, for normal mode.

Pin description

Signal	Sub HD 15-Pin connector Pin number	I/O	I/O type	Description
RESET	14	I/O	SCHMITT	Modem Reset

3 Characteristics

3.1 Basic services

The basic services of the Fastrack modem M1206 are given in the table below.

	GSM	DCS
Standard	900 MHz.	1800 MHz
	E-GSM compliant Class 4 (2W). GSM phase 2.	E-GSM compliant Class 1 (1W) GSM phase 2..
Interface	Serial interface RS232 V.24/V.28 AT command set based on V.23ter and GSM 07.05 & 07.07. Auto-bauding function between 2400 bits/s and 19200 bits/s No auto-framing available	Serial interface RS232 V.24/V.28 AT command set based on V.23ter and GSM 07.05 & 07.07. Auto-bauding function between 2400 bits/s and 19200 bits/s No auto-framing available
SMS	Mobile Originated (MO) and Mobile Terminated (MT). Mode Text & PDU point to point. Cell broadcast.	Mobile Originated (MO) and Mobile Terminated (MT). Mode Text & PDU point to point. Cell broadcast.
Data	In accordance with GSM 07.05 Asynchronous 2400, 4800, 9600 and 14400 bits/s. Transparent and Non Transparent mode In Non Transparent Mode: 300, 1200, 1200/75 bauds.	In accordance with GSM 07.05 Asynchronous 2400, 4800, 9600 and 14400 bits/s. Transparent and Non Transparent mode In Non Transparent Mode: 300, 1200, 1200/75 bauds.
Fax	Mode 3.1 kHz (PSTN) and V110 (ISDN) 2400/4800/7200/9600 bits/s, GSM teleservice 62 in Transparent Mode. Class 1 & Class 2. Group 3 compatible.	Mode 3.1 kHz (PSTN) and V110 (ISDN) 2400/4800/7200/9600 bits/s, GSM teleservice 62 in Transparent Mode. Class 1 & Class 2. Group 3 compatible.
Audio	Half rate / Full rate / Enhanced Full rate operation. Accessories (options): <ul style="list-style-type: none"> ▪ Handset, ▪ Car Kit. 	Half rate / Full rate / Enhanced Full rate operation. Accessories (options): <ul style="list-style-type: none"> ▪ Handset, ▪ Car Kit.
GPRS	Class 10. Coding schemes: CS1 to CS4 Compliant with SMG31bis	Class 10. Coding schemes: CS1 to CS4 Compliant with SMG31bis

4 Using the modem

4.1 Getting started with the modem

4.1.1 Mounting the modem

To mount the modem, bind it using the holding bridles as shown in the diagram below.

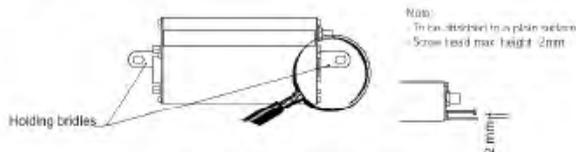


Figure 9: Modem mounting

4.1.2 Setting up the Fastrack modem

To set up the modem, do the following:

- Press SIM card holder ejector (yellow button) with a sharp object (the tip of a pen for example).
- Insert the SIM card in the holder.
- Verify the SIM card fits in the holder properly.
- Connect the antenna to the SMA connector.
- Connect both sides of the serial and control cable (13-pin Sub HD connector on the modem side).
- Connect the power supply cable to the power supply source.

Note: for automotive application, it is recommended to connect the V+BATT line of the modem directly to the positive terminal of the battery.

- Plug the power supply cable into the modem and switch on the external power supply.
- Now the modem is ready to work. Refer to chapter 5 for some AT commands to configure the modem.

5 AT commands for the modem

The table below reminds the main AT commands required for getting started the modem.

For other AT commands available or further information about the AT commands, refer to document [1].

Description	AT commands	Modem's response	Comment
Enter PIN Code	AT+CPIN=1234	OK	PIN Code accepted.
		+CME ERROR: 16	Incorrect PIN Code (with +CMEE = 1 mode).
		+CME ERROR: 3	PIN already entered (with +CMEE = 1 mode).
Modem synchronization checking	AT+CREG?	CREG=<mode>, 1	Modem synchronised on the network.
		CREG=<mode>, 2	Synchronization lost, re-synchronization attempt.
		CREG=<mode>, 0	Modem not synchronised on the network, no synchronization attempt.
Receiving an incoming call	ATA	OK	Answer the call.
Initiate a call	ATD<phone number>; (Don't forget the « ; » at the end for « voice » call)	OK	Communication established.
		CME ERROR: 11	PIN code not entered (with +CMEE = 1 mode).
		CME ERROR: 3	AOC credit exceeded or a communication is already established.
Initiate an emergency call	ATD112; (Don't forget the « ; » at the end for « voice » call)	OK	Communication established.
Communication loss		NO CARRIER	
Hang up	ATH	OK	
Store the parameters in EEPROM	AT&W	OK	The configuration settings are stored in EEPROM.

2. Datasheet ATmega16

Features

- High-performance, Low-power Atmel® AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single-look Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
 - 16 Kbytes of In-System Self-programmable Flash program memory
 - 512 Bytes EEPROM
 - 1 Kbyte Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Look Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- JTAG (IEEE Std. 1149.1 Compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuse, and Look Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four PWM Channels
 - 8-channel, 10-bit ADC
 - 8 Single-ended Channels
 - 7 Differential Channels in TQFP Package Only
 - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages
 - 2.7V - 5.5V for ATmega16L
 - 4.5V - 5.5V for ATmega16
- Speed Grades
 - 0 - 8 MHz for ATmega16L
 - 0 - 16 MHz for ATmega16
- Power Consumption @ 1 MHz, 3V, and 25°C for ATmega16L
 - Active: 1.1 mA
 - Idle Mode: 0.35 mA
 - Power-down Mode: < 1 µA



8-bit AVR[®]
Microcontroller
with 16K Bytes
In-System
Programmable
Flash

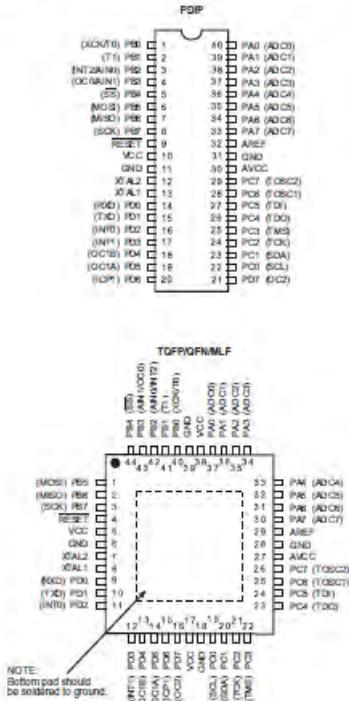
ATmega16
ATmega16L

Rev. 2482E-AVR-07/00



Pin Configurations

Figure 1. Pinout ATmega16



Disclaimer

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

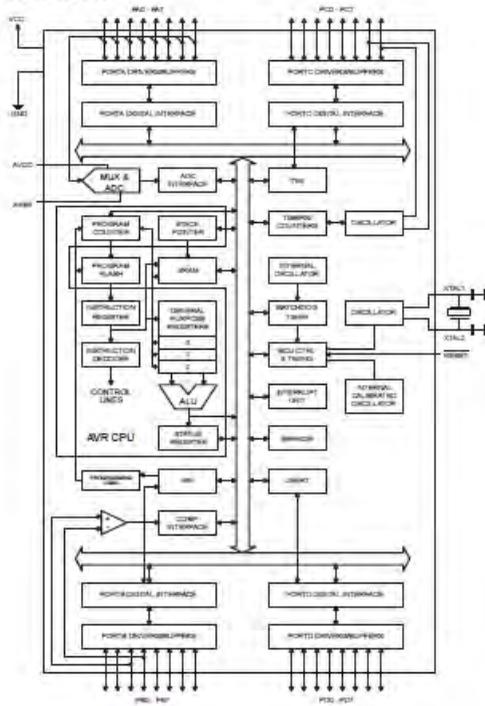


Overview

The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 2. Block Diagram



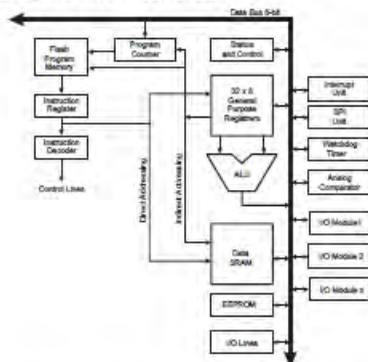
AVR CPU Core

Introduction

This section discusses the AVR core architecture in general. The main function of the CPU core is to ensure correct program execution. The CPU must therefore be able to access memories, perform calculations, control peripherals, and handle interrupts.

Architectural Overview

Figure 3. Block Diagram of the AVR MCU Architecture



In order to maximize performance and parallelism, the AVR uses a Harvard architecture – with separate memories and buses for program and data. Instructions in the program memory are executed with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is In-System Reprogrammable Flash memory.

The fast-access Register File contains 32 x 8-bit general purpose working registers with a single clock cycle access time. This allows single-cycle Arithmetic Logic Unit (ALU) operation. In a typical ALU operation, two operands are output from the Register File, the operation is executed, and the result is stored back in the Register File – in one clock cycle.

Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing – enabling efficient address calculations. One of these address pointers can also be used as an address pointer for look up tables in Flash Program memory. These added function registers are the 16-bit X-register, Y-register, and Z-register, described later in this section.

The ALU supports arithmetic and logic operations between registers or between a constant and a register. Single register operations can also be executed in the ALU. After an arithmetic operation, the Status Register is updated to reflect information about the result of the operation.



General Purpose Register File

The Register File is optimized for the AVR Enhanced RISC Instruction set. In order to achieve the required performance and flexibility, the following input/output schemes are supported by the Register File:

- One 8-bit output operand and one 8-bit result input
- Two 8-bit output operands and one 8-bit result input
- Two 8-bit output operands and one 16-bit result input
- One 16-bit output operand and one 16-bit result input

Figure 4 shows the structure of the 32 general purpose working registers in the CPU.

Figure 4. AVR CPU General Purpose Working Registers



Most of the instructions operating on the Register File have direct access to all registers, and most of them are single cycle instructions.

As shown in Figure 4, each register is also assigned a data memory address, mapping them directly into the first 32 locations of the user Data Space. Although not being physically implemented as SRAM locations, this memory organization provides great flexibility in access of the registers, as the X-, Y-, and Z-pointer Registers can be set to index any register in the file.

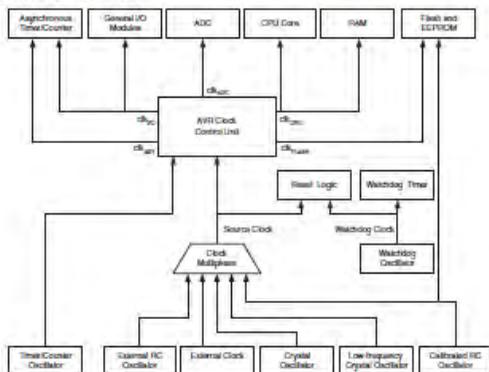


System Clock and Clock Options

Clock Systems and their Distribution

Figure 11 presents the principal clock systems in the AVR and their distribution. All of the clocks need not be active at a given time. In order to reduce power consumption, the clocks to modules not being used can be halted by using different sleep modes, as described in “Power Management and Sleep Modes” on page 32. The clock systems are detailed Figure 11.

Figure 11. Clock Distribution



CPU Clock – clk_{CPU}

The CPU clock is routed to parts of the system concerned with operation of the AVR core. Examples of such modules are the General Purpose Register File, the Status Register and the data memory holding the Stack Pointer. Halting the CPU clock inhibits the core from performing general operations and calculations.

I/O Clock – clk_{IO}

The I/O clock is used by the majority of the I/O modules, like Timer/Counters, SPI, and USART. The I/O clock is also used by the External Interrupt module, but note that some external interrupts are detected by asynchronous logic, allowing such interrupts to be detected even if the I/O clock is halted. Also note that address recognition in the TWI module is carried out asynchronously when clk_{IO} is halted, enabling TWI address reception in all sleep modes.

Flash Clock – clk_{FLASH}

The Flash clock controls operation of the Flash interface. The Flash clock is usually active simultaneously with the CPU clock.

Asynchronous Timer Clock – clk_{ASY}

The Asynchronous Timer clock allows the Asynchronous Timer/Counter to be clocked directly from an external 32 kHz clock crystal. The dedicated clock domain allows using this Timer/Counter as a real-time counter even when the device is in sleep mode.



Standby Mode

When the SM2..0 bits are 110 and an external crystal/resonator clock option is selected, the SLEEP instruction makes the MCU enter Standby mode. This mode is identical to Power-down with the exception that the Oscillator is kept running. From Standby mode, the device wakes up in six clock cycles.

Extended Standby Mode

When the SM2..0 bits are 111 and an external crystal/resonator clock option is selected, the SLEEP instruction makes the MCU enter Extended Standby mode. This mode is identical to Power-save mode with the exception that the Oscillator is kept running. From Extended Standby mode, the device wakes up in six clock cycles.

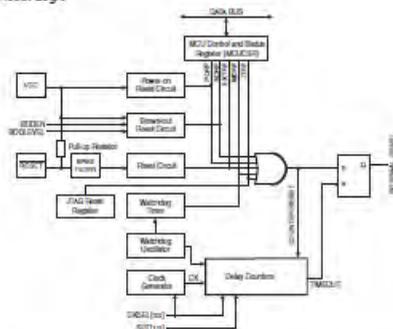
Table 14. Active Clock Domains and Wake Up Sources in the Different Sleep Modes

Sleep Mode	Active Clock domains					Oscillators		Wake-up Sources					
	clk_CPU	clk_FCPU	clk_IO	clk_ADC	clk_ASR	Main Clock Source Enabled	Timer Osc. Enabled	INT2 INT1 INT0	TWI Address Match	Timer 2	SPM / EEPROM Ready	ADC	Other IO
Idle			X	X	X	X	X ⁽²⁾	X	X	X	X	X	X
ADC Noise Reduction				X	X	X	X ⁽²⁾	X ⁽²⁾	X	X	X	X	
Power Down								X ⁽²⁾	X				
Power Save					X ⁽²⁾		X ⁽²⁾	X ⁽²⁾	X	X ⁽²⁾			
Standby ⁽¹⁾						X		X ⁽²⁾	X				
Extended Standby ⁽¹⁾					X ⁽²⁾	X	X ⁽²⁾	X ⁽²⁾	X	X ⁽²⁾			

Notes:

1. External Crystal or resonator selected as clock source.
2. If AS2 bit in ASBR is set.
3. Only INT2 or level interrupt INT1 and INT0.



Figure 15. Reset Logic

Table 15. Reset Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Units
V_{POR}	Power-on Reset Threshold Voltage (rising)			1.4	2.3	V
	Power-on Reset Threshold Voltage (falling) ¹⁾			1.3	2.3	
V_{RES}	RESET Pin Threshold Voltage		0.1V _{CC}		0.9V _{CC}	
t_{RES}	Minimum pulse width on RESET Pin				1.5	μs
V_{BOR}	Brown-out Reset Threshold Voltage ²⁾	BODLEVEL = 1	2.5	2.7	3.2	V
		BODLEVEL = 0	3.6	4.0	4.5	
t_{BOD}	Minimum low voltage period for Brown-out Detection	BODLEVEL = 1		2		μs
		BODLEVEL = 0		2		
V_{IHBR}	Brown-out Detector hysteresis			50		mV

Notes: 1. The Power-on Reset will not work unless the supply voltage has been below V_{POR} (falling).
 2. V_{BOR} may be below nominal minimum operating voltage for some devices. For devices where this is the case, the device is tested down to $V_{OC} = V_{BOR}$ during the production test. This guarantees that a Brown-out Reset will occur before V_{OC} drops to a voltage where correct operation of the microcontroller is no longer guaranteed. The test is performed using BODLEVEL = 1 for ATmega16L and BODLEVEL = 0 for ATmega16. BODLEVEL = 1 is not applicable for ATmega16.



ATmega16 Typical Characteristics

The following charts show typical behavior. These figures are not tested during manufacturing. All current consumption measurements are performed with all I/O pins configured as inputs and with internal pull-ups enabled. A sine wave generator with rail-to-rail output is used as clock source.

The power consumption in Power-down mode is independent of clock selection.

The current consumption is a function of several factors such as: operating voltage, operating frequency, loading of I/O pins, switching rate of I/O pins, code executed and ambient temperature. The dominating factors are operating voltage and frequency.

The current drawn from capacitive loaded pins may be estimated (for one pin) as $C_L \cdot V_{CC} \cdot f$ where C_L = load capacitance, V_{CC} = operating voltage and f = average switching frequency of I/O pin.

The parts are characterized at frequencies higher than test limits. Parts are not guaranteed to function properly at frequencies higher than the ordering code indicates.

The difference between current consumption in Power-down mode with Watchdog Timer enabled and Power-down mode with Watchdog Timer disabled represents the differential current drawn by the Watchdog Timer.

Active Supply Current Figure 145. Active Supply Current vs. Frequency (0.1 MHz - 1.0 MHz)

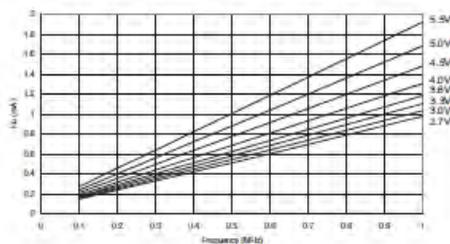


Figure 150. Active Supply Current vs. Frequency (1 MHz - 20 MHz)

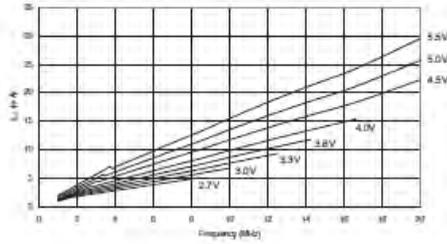
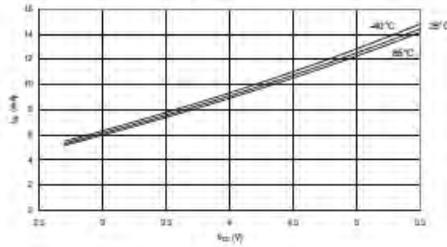


Figure 151. Active Supply Current vs. V_{CC} (Internal RC Oscillator, 8 MHz)



BAB V
PENUTUP

BAB V

PENUTUP

Bab penutup berisi tentang kesimpulan-kesimpulan yang didapatkan selama proses pembuatan Tugas Akhir ini beserta saran-saran untuk perbaikan dan pengembangannya.

5.1 Kesimpulan

Berdasarkan rumusan masalah, cara untuk membuat sistem pemberian pakan pada kandang kucing adalah dengan berbagai peralatan seperti ATmega16, *relay*, *motor dc*, modem wavecom yang dirangkai sedemikian rupa sehingga ATmega16 dapat mengontrol *motor dc* sebagai pembuka pintu pakan agar makanan dapat diberikan pada hewan peliharaan.

Berdasarkan proses simulasi, implementasi, pengujian dan analisis sistem dapat ditarik kesimpulan sebagai berikut:

- Dengan 10 kali pengujian rata – rata waktu respon *motor* selama 1,2 detik setelah sms dikirimkan dari *handphone* pemilik.
- Dalam 10 kali pengujian dengan menggunakan perintah baik P1, P2, P3 diperoleh keberhasilan hampir 90%. Dengan delay antara 1 sampai 3 detik dari mulai sms dikirim hingga *motor* membuka pintu tempat penyimpanan pakan .
- Dengan menggunakan “P3” lebih efektif karena makanan yang keluar paling banyak.

5.2 Saran

Saran yang dapat diajukan untuk penelitian lebih lanjut mengenai topik ini adalah:

1. Dusahakan pada pengujian selanjutnya waktu respon pada motor dapat lebih diefektifkan lagi.
2. Dusahakan pada pengujian selanjutnya sistem penyimpanan makanan dapat diperbesar dimensi wadahnya.

DAFTAR PUSTAKA

DAFTAR PUSTAKA

- [1], Pengetahuan Dasar ATmega16, <URL: <http://www.engineersgarage.com/electronic-components/atmega16-microcontroller/>> , Agustus 2012
- [2], Andrianto, Beri, 2008, Pemrograman Mikrokontroler AVR ATMEGA16 Menggunakan Bahasa C (Codevision AVR), Bandung, Informatika.Jakarta,2009.
- [3], Bejo, C dan AVR Rahasia Kemudahan Bahasa C dalam Mikrokontroler ATmega8535, Yogyakarta, Graha Ilmu. , Agus., 2008
- [4], ATmega 16 Datasheet, <URL : <http://www.alldatasheet.com/datasheet-pdf/pdf/78532/ATMEL/ATMEGA16.html>, 1 Agustus 2013
- [5], Teori Relay Elektromekanik, <URL : <http://elektronika-dasar.web.id/teori-elektronika/teori-relay-elektro-mekanik/>>, 1 Agustus 2013
- [6], Teori Modem Wavecom, <URL: <http://fungsimodemwavecom.tumblr.com/>> , Juni 2015
- [7], Sensor Arus Listrik ACS712, <URL : depokinstruments.com/2012/03/29/sensor-arus-listrik-ac712/>, 2 Agustus 2013

LAMPIRAN

LAMPIRAN A

LISTING PROGRAM

Program SMS Microcontroller ATmega16

This program was produced by the
CodeWizardAVR V2.04.4a Advanced
Automatic Program Generator
© Copyright 1998-2009 Pavel Haiduc, HP InfoTech s.r.l.
<http://www.hpinfotech.com>

Project :
Version :
Date : 4/30/2015
Author : NeVaDa
Company : metrocom
Comments:

Chip type : ATmega16
Program type : Application
AVR Core Clock frequency: 11.059200 MHz
Memory model : Small
External RAM size : 0
Data Stack size : 256

*****/

```
#include <mega16.h>
#include <stdlib.h>
#include <delay.h>
#include <stdio.h>
```

```
// Declare your global variables here
char jg,data,a1,a2;
void baca_sms()
{
    data=getchar(); // baca data masuk persatu karakter
    while(data!='+')// mencari +
```

```

{data=getchar();}

    if (data=='+')
    {
        data=getchar();
        if (data=='C')
        {
            data=getchar();
            if (data=='M')
            {
                data=getchar();
                if (data=='T')
                {
                    data=getchar();
                    while (data!='') // cari /
                    {
                        data=getchar();
                    }
                    while (data!='+') // cari petik "
                    {
                        data=getchar();
                    }
                    while (data!="") // cari petik "
                    {
                        data=getchar();
                    }
                    jg=getchar(); // baca Spasi
                    jg=getchar(); // baca enter
                    a1=getchar(); // baca data SMS
                    a2=getchar();

//=====>>>
                    if (a1=='P' && a2=='1')
                    {
                        PORTB.2=1;
                        delay_ms(25);
                        PORTB.2=0;

```

```

1          delay_ms(1000); //setting waktu buka pakan

          PORTB.3=1;
          delay_ms(20);
          PORTB.3=0;
          }

//=====>>>>
          if (a1=='P' && a2=='2')
          {
          PORTB.2=1;
          delay_ms(25);
          PORTB.2=0;
          delay_ms(2000); //setting waktu buka pakan
2
          PORTB.3=1;
          delay_ms(20);
          PORTB.3=0;
          }

//=====>>>>
          if (a1=='P' && a2=='3')
          {
          PORTB.2=1;
          delay_ms(25);
          PORTB.2=0;
          delay_ms(3000); //setting waktu buka pakan
3
          PORTB.3=1;
          delay_ms(20);
          PORTB.3=0;
          }

          printf("AT+CMGD=1"); // hapus SMS
          putchar(13);
          delay_ms(300);
          printf("AT+CMGD=2"); // hapus SMS

```

```

        putchar(13);
        delay_ms(300);

    }
}
}
}
//AT+CMT

}
void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTA=0x00;
DDRA=0x00;

// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=Out Func2=Out
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=0 State2=0 State1=T
State0=T
PORTB=0x00;
DDRB=0x0C;

// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTC=0x00;
DDRC=0x00;

```

```

// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTD=0x00;
DDRD=0x00;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;

```

```

OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x47;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

```

```
//=====> let's start
delay_ms(10000);
printf("AT+CMGF=1");
putchar(13);
printf("AT+CNMI=1,2,0,0,0");
putchar(13);
putchar(10);
delay_ms(10);
printf ("AT+CMGS=");
putchar(34);
printf("085706670178");
putchar(34);
putchar(13);
delay_ms(10);
printf("READY");
putchar(13);
putchar(26);
putchar(13);
putchar(10);
delay_ms(1000);

while (1)
{
    baca_sms();
};
}
```

LAMPIRAN 2

DATASHEET

1. *Dataheet ModemWavecom*



Fastrack modem M12 series

Fastrack modem M1206 User Guide

Reference: WM_FFJ_M12_UGD_001

Revision: 002

Date: 18th September 2003



wavecom confidential ©

This document is the sole and exclusive property of WAVECOM. Not to be distributed or divulged without prior written agreement.

Ce document est la propriété exclusive de WAVECOM. Il ne peut être communiqué ou divulgué à des tiers sans son autorisation préalable.

Page : 1 / 38

1 General description

1.1 Presentation

FASTRACK M1206 modem is a self-contained E-GSM/GSM-GPRS 900/1800 dual-band modem and is GPRS class 10 capable.

This modem supports the following transmissions:

- Data,
- Fax,
- Short Messages (Point to point and Cell Broadcast),
- Voice calls.

The modem comprises several interfaces:

- LED function indicating the operating status,
- External antenna (via SMA connector),
- RS232 Serial and control link (via 15-pin SUB HD connector),
- Power supply (via 4-pin Micro-Fit™ connector),
- SIM card holder.

The main features of the modem are the following:

- 2 Watts E-GSM 900 radio section.
- 1 Watt GSM1800 radio section.
- 32 Mbits of Flash memory and 4 Mbits of SRAM,
- Real Time Clock with calendar.
- Echo Cancellation + noise reduction.
- Full GSM or GSM / GPRS software stack.
- Hardware GPRS class 10 capable.
- Complete shielding.
- A DC Power supply,
- A RS232 serial link,
- Audio interface for:
 - microphone,
 - speaker.
- A 3V / 5V SIM interface.

Modem mechanical case is made out of aluminium profile ended by two holding bridles at each extremity.

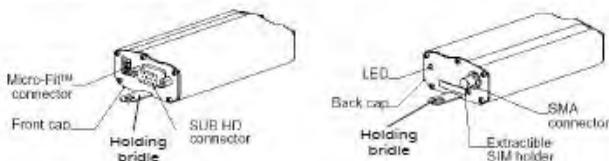


Figure 1: FASTRACK M1206 modem presentation

1.2 External connections

1.2.1 Connectors

1.2.1.1 General

FASTRACK M1206 modem has three external connections:

- Antenna connector: SMA connector for RF connection to the antenna,
- Sub D high density 13-pin connector for:
 - RS232 serial link connection,
 - Audio lines (microphone and speaker) connection,
 - BOOT and RESET signals connection.
- Power supply connector: 4-pin Micro FIT connector for DC Power Supply.

1.2.1.2 Antenna connector



Figure 2: Antenna connector

1.2.1.3 Sub HD 15-pin connector



Figure 8: Sub HD 15-pin connector

Pin #	Signal (CCITT / EIA)	I/O	I/O type	Description	Comment
1	CT109 / DCD	O	STANDARD RS232	RS232 Data Carrier Detect	
2	CT103 / TX	I	STANDARD RS232	RS232 Transmit serial data	
3	BOOT	I	CMOS	Boot	Active low. Pull down through 1K for Flash downloading
4	Microphone (+)	I	Analog	Microphone positive line	
5	Microphone (-)	I	Analog	Microphone negative line	
6	CT104 / RX	O	STANDARD RS232	RS232 Receive serial data	
7	CT107 / DSR	O	STANDARD RS232	RS232 Data Set Ready	
8	CT108-2 / DTR	I	STANDARD RS232	RS232 Data Terminal Ready	
9	GND	-	GND	Ground	
10	Speaker (+)	O	Analog	Speaker positive line	

Pin #	Signal (CCITT / EIA)	I/O	I/O type	Description	Comment
11	CT106 / CTS	O	STANDARD RS232	RS232 Clear To Send	
12	CT105 / RTS	I	STANDARD RS232	RS232 Request To Send	
13	CT123 / RI	O	STANDARD RS232	RS232 Ring Indicator	
14	RESET	I/O	Schmitt	Modem reset	Active low
15	Speaker [-]	O	Analog	Speaker negative line	

1.2.1.4 Power supply connector:



Figure 4: Power supply connector

Pin #	Signal	I/O	I/O type	Description	Comment
1	V+BATT	I	Power supply	Battery input	High current
2	GND		Power supply	Ground	
3	NC				Reserved
4	NC				Reserved

2 Functional description

2.1 Architecture

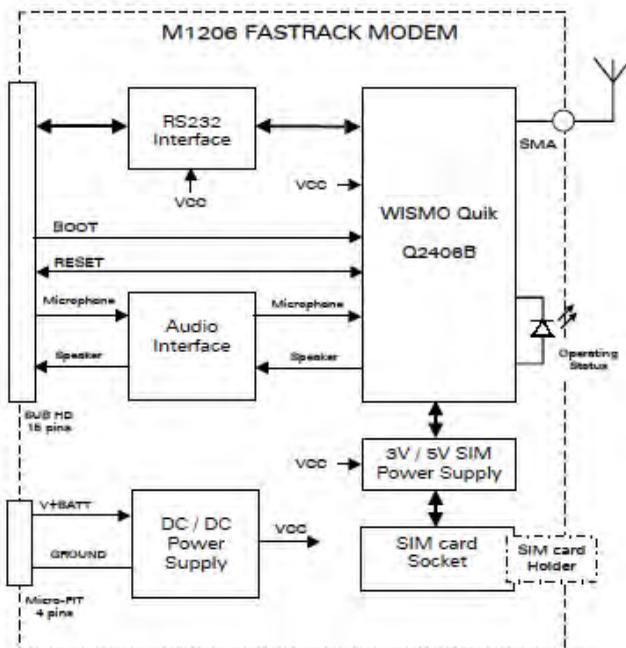


Figure 6: Functional architecture

2.2 RS232 serial link

2.2.1 General presentation

The RS232 interface performs the voltage level adaptation (V24/CMOS ↔ V24/V28) between the internal WISMO module (DCE) and the external world (DTE).

The RS232 interface is protected (by ESD protection) against electrostatic surges on the RS232 lines.

Filtering guarantees:

- EMI/RFI protection in input and output,
- Signal smoothing.

A flexible 6-wire serial interface is available complying with RS232 protocol signaling.

The signals available are:

- TX data (CT103/TX),
- RX data (CT104/RX),
- Request To Send (CT105/RTS),
- Clear To Send (CT106/CTS),
- Data Terminal Ready (CT108-2/DTR),
- Data Set Ready (CT107/DSR).

The 2 additional signals are Data Carrier Detect (CT109/DCD) and Ring Indicator (CT125/RI).

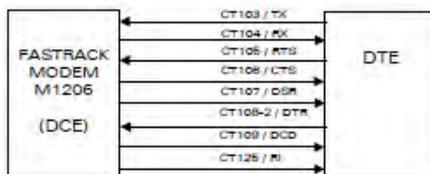


Figure 7: RS232 Serial Link signals

RS232 interface has been designed to allow a certain flexibility in the use of the serial interface signals. However, the use of TX, RX, CTS and RTS signals is mandatory which is not the case for DTR, DSR, DCD and RI signals which can be not used.

2.2.2 Pin out description

Pin description

Signal	Sub HD connector Pin number	I/O	I/O type RS232 STANDARD	Description
CT103/TX	2	I	TX	Transmit serial data
CT104/RX	6	O	RX	Receive serial data
CT105/RTS	12	I	RTS	Request To Send
CT106/CTS	11	O	CTS	Clear To Send
CT107/DSR	7	O	DSR	Data Set Ready
CT108-2/DTR	8	I	DTR	Data Terminal Ready
CT109/DCD	1	O	DCD	Data Carrier Detect
CT125/RI	13	O	RI	Ring Indicator
CT102/GND	9			Ground

2.3 RESET

2.3.1 General presentation

This signal is used to force a reset procedure by providing low level during at least 500 μ s.

This signal has to be considered as an emergency reset only. A reset procedure is automatically driven by an internal hardware during the power-up sequence.

This signal can also be used to provide a reset to an external device. It then behaves as an output. If no external reset is necessary this input can be left open, if used (emergency reset), it has to be driven by an open collector or an open drain output:

- RESET pin 14 = 0, for Modem Reset,
- RESET pin 14 = 1, for normal mode.

Pin description

Signal	Sub HD 15-Pin connector Pin number	I/O	I/O type	Description
RESET	14	I/O	SCHMITT	Modem Reset

3 Characteristics

3.1 Basic services

The basic services of the Fastrack modem M1206 are given in the table below.

	GSM	DCS
Standard	900 MHz.	1800 MHz
	E-GSM compliant Class 4 (2W). GSM phase 2.	E-GSM compliant Class 1 (1W) GSM phase 2..
Interface	Serial interface RS232 V.24/V.28 AT command set based on V.23ter and GSM 07.05 & 07.07. Auto-bauding function between 2400 bits/s and 19200 bits/s No auto-framing available	Serial interface RS232 V.24/V.28 AT command set based on V.23ter and GSM 07.05 & 07.07. Auto-bauding function between 2400 bits/s and 19200 bits/s No auto-framing available
SMS	Mobile Originated (MO) and Mobile Terminated (MT). Mode Text & PDU point to point. Cell broadcast.	Mobile Originated (MO) and Mobile Terminated (MT). Mode Text & PDU point to point. Cell broadcast.
Data	In accordance with GSM 07.05 Asynchronous 2400, 4800, 9600 and 14400 bits/s. Transparent and Non Transparent mode In Non Transparent Mode: 300, 1200, 1200/75 bauds.	In accordance with GSM 07.05 Asynchronous 2400, 4800, 9600 and 14400 bits/s. Transparent and Non Transparent mode In Non Transparent Mode: 300, 1200, 1200/75 bauds.
Fax	Mode 3.1 kHz (PSTN) and V110 (ISDN) 2400/4800/7200/9600 bits/s, GSM teleservice 62 in Transparent Mode. Class 1 & Class 2. Group 3 compatible.	Mode 3.1 kHz (PSTN) and V110 (ISDN) 2400/4800/7200/9600 bits/s, GSM teleservice 62 in Transparent Mode. Class 1 & Class 2. Group 3 compatible.
Audio	Half rate / Full rate / Enhanced Full rate operation. Accessories (options): <ul style="list-style-type: none"> ▪ Handset, ▪ Car Kit. 	Half rate / Full rate / Enhanced Full rate operation. Accessories (options): <ul style="list-style-type: none"> ▪ Handset, ▪ Car Kit.
GPRS	Class 10. Coding schemes: CS1 to CS4 Compliant with SMG31bis	Class 10. Coding schemes: CS1 to CS4 Compliant with SMG31bis

4 Using the modem

4.1 Getting started with the modem

4.1.1 Mounting the modem

To mount the modem, bind it using the holding bridles as shown in the diagram below.

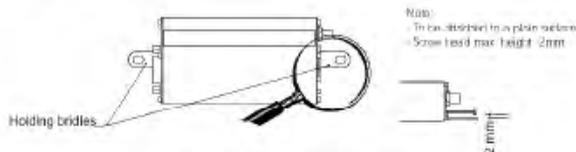


Figure 9: Modem mounting

4.1.2 Setting up the Fastrack modem

To set up the modem, do the following:

- Press SIM card holder ejector (yellow button) with a sharp object (the tip of a pen for example).
- Insert the SIM card in the holder.
- Verify the SIM card fits in the holder properly.
- Connect the antenna to the SMA connector.
- Connect both sides of the serial and control cable (13-pin Sub HD connector on the modem side).
- Connect the power supply cable to the power supply source.

Note: for automotive application, it is recommended to connect the V+BATT line of the modem directly to the positive terminal of the battery.

- Plug the power supply cable into the modem and switch on the external power supply.
- Now the modem is ready to work. Refer to chapter 5 for some AT commands to configure the modem.

5 AT commands for the modem

The table below reminds the main AT commands required for getting started the modem.

For other AT commands available or further information about the AT commands, refer to document [1].

Description	AT commands	Modem's response	Comment
Enter PIN Code	AT+CPIN=1234	OK	PIN Code accepted.
		+CME ERROR: 16	Incorrect PIN Code (with +CMEE = 1 mode).
		+CME ERROR: 3	PIN already entered (with +CMEE = 1 mode).
Modem synchronization checking	AT+CREG?	CREG=<mode>, 1	Modem synchronised on the network.
		CREG=<mode>, 2	Synchronization lost, re-synchronization attempt.
		CREG=<mode>, 0	Modem not synchronised on the network, no synchronization attempt.
Receiving an incoming call	ATA	OK	Answer the call.
Initiate a call	ATD<phone number>; (Don't forget the « ; » at the end for « voice » call)	OK	Communication established.
		CME ERROR: 11	PIN code not entered (with +CMEE = 1 mode).
		CME ERROR: 3	AOC credit exceeded or a communication is already established.
Initiate an emergency call	ATD112; (Don't forget the « ; » at the end for « voice » call)	OK	Communication established.
Communication loss		NO CARRIER	
Hang up	ATH	OK	
Store the parameters in EEPROM	AT&W	OK	The configuration settings are stored in EEPROM.

2. Datasheet ATmega16

Features

- High-performance, Low-power Atmel® AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single-look Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
 - 16 Kbytes of In-System Self-programmable Flash program memory
 - 512 Bytes EEPROM
 - 1 Kbyte Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Look Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- JTAG (IEEE Std. 1149.1 Compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Look Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescalers and Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four PWM Channels
 - 8-channel, 10-bit ADC
 - 8 Single-ended Channels
 - 7 Differential Channels in TQFP Package Only
 - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages
 - 2.7V - 6.6V for ATmega16L
 - 4.5V - 6.6V for ATmega16
- Speed Grades
 - 0 - 8 MHz for ATmega16L
 - 0 - 16 MHz for ATmega16
- Power Consumption @ 1 MHz, 3V, and 25°C for ATmega16L
 - Active: 1.1 mA
 - Idle Mode: 0.35 mA
 - Power-down Mode: < 1 µA



8-bit AVR[®]
Microcontroller
with 16K Bytes
In-System
Programmable
Flash

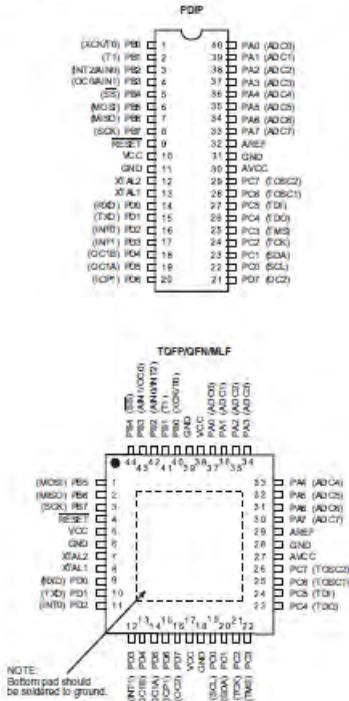
ATmega16
ATmega16L

Rev. 2482E-AVR-07/00



Pin Configurations

Figure 1. Pinout ATmega16



Disclaimer

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.



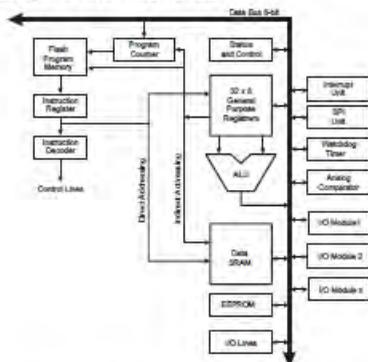
AVR CPU Core

Introduction

This section discusses the AVR core architecture in general. The main function of the CPU core is to ensure correct program execution. The CPU must therefore be able to access memories, perform calculations, control peripherals, and handle interrupts.

Architectural Overview

Figure 3. Block Diagram of the AVR MCU Architecture



In order to maximize performance and parallelism, the AVR uses a Harvard architecture – with separate memories and buses for program and data. Instructions in the program memory are executed with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is In-System Reprogrammable Flash memory.

The fast-access Register File contains 32 x 8-bit general purpose working registers with a single clock cycle access time. This allows single-cycle Arithmetic Logic Unit (ALU) operation. In a typical ALU operation, two operands are output from the Register File, the operation is executed, and the result is stored back in the Register File – in one clock cycle.

Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing – enabling efficient address calculations. One of these address pointers can also be used as an address pointer for look up tables in Flash Program memory. These added function registers are the 16-bit X-register, Y-register, and Z-register, described later in this section.

The ALU supports arithmetic and logic operations between registers or between a constant and a register. Single register operations can also be executed in the ALU. After an arithmetic operation, the Status Register is updated to reflect information about the result of the operation.



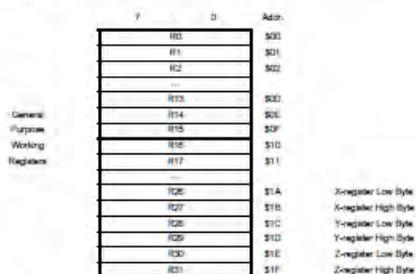
General Purpose Register File

The Register File is optimized for the AVR Enhanced RISC Instruction set. In order to achieve the required performance and flexibility, the following input/output schemes are supported by the Register File:

- One 8-bit output operand and one 8-bit result input
- Two 8-bit output operands and one 8-bit result input
- Two 8-bit output operands and one 16-bit result input
- One 16-bit output operand and one 16-bit result input

Figure 4 shows the structure of the 32 general purpose working registers in the CPU.

Figure 4. AVR CPU General Purpose Working Registers



Most of the instructions operating on the Register File have direct access to all registers, and most of them are single cycle instructions.

As shown in Figure 4, each register is also assigned a data memory address, mapping them directly into the first 32 locations of the user Data Space. Although not being physically implemented as SRAM locations, this memory organization provides great flexibility in access of the registers, as the X-, Y-, and Z-pointer Registers can be set to index any register in the file.

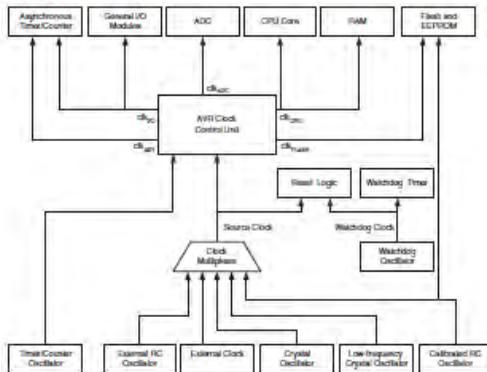


System Clock and Clock Options

Clock Systems and their Distribution

Figure 11 presents the principal clock systems in the AVR and their distribution. All of the clocks need not be active at a given time. In order to reduce power consumption, the clocks to modules not being used can be halted by using different sleep modes, as described in "Power Management and Sleep Modes" on page 32. The clock systems are detailed Figure 11.

Figure 11. Clock Distribution



CPU Clock – clk_{CPU}

The CPU clock is routed to parts of the system concerned with operation of the AVR core. Examples of such modules are the General Purpose Register File, the Status Register and the data memory holding the Stack Pointer. Halting the CPU clock inhibits the core from performing general operations and calculations.

I/O Clock – $clk_{I/O}$

The I/O clock is used by the majority of the I/O modules, like Timer/Counters, SPI, and USART. The I/O clock is also used by the External Interrupt module, but note that some external interrupts are detected by asynchronous logic, allowing such interrupts to be detected even if the I/O clock is halted. Also note that address recognition in the TWI module is carried out asynchronously when $clk_{I/O}$ is halted, enabling TWI address reception in all sleep modes.

Flash Clock – clk_{FLASH}

The Flash clock controls operation of the Flash interface. The Flash clock is usually active simultaneously with the CPU clock.

Asynchronous Timer Clock – clk_{ASY}

The Asynchronous Timer clock allows the Asynchronous Timer/Counter to be clocked directly from an external 32 kHz clock crystal. The dedicated clock domain allows using this Timer/Counter as a real-time counter even when the device is in sleep mode.



Standby Mode

When the SM2..0 bits are 110 and an external crystal/resonator clock option is selected, the SLEEP instruction makes the MCU enter Standby mode. This mode is identical to Power-down with the exception that the Oscillator is kept running. From Standby mode, the device wakes up in six clock cycles.

Extended Standby Mode

When the SM2..0 bits are 111 and an external crystal/resonator clock option is selected, the SLEEP instruction makes the MCU enter Extended Standby mode. This mode is identical to Power-save mode with the exception that the Oscillator is kept running. From Extended Standby mode, the device wakes up in six clock cycles.

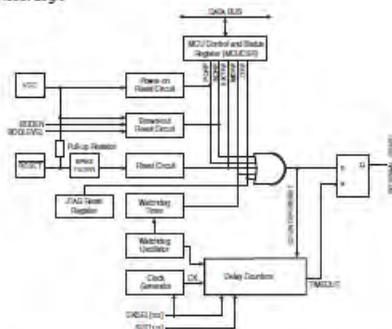
Table 14. Active Clock Domains and Wake Up Sources in the Different Sleep Modes

Sleep Mode	Active Clock domains					Oscillators		Wake-up Sources					
	clk_CPU	clk_FUZZ	clk_I/O	clk_ADC	clk_ASR	Main Clock Source Enabled	Timer Osc. Enabled	INT2 INT1 INT0	TWI Address Match	Timer 2	SPM / EEPROM Ready	ADC	Other IO
Idle			X	X	X	X	X ⁽²⁾	X	X	X	X	X	X
ADC Noise Reduction				X	X	X	X ⁽²⁾	X ⁽²⁾	X	X	X	X	
Power Down								X ⁽²⁾	X				
Power Save					X ⁽²⁾		X ⁽²⁾	X ⁽²⁾	X	X ⁽²⁾			
Standby ⁽¹⁾						X		X ⁽²⁾	X				
Extended Standby ⁽¹⁾					X ⁽²⁾	X	X ⁽²⁾	X ⁽²⁾	X	X ⁽²⁾			

Notes:

1. External Crystal or resonator selected as clock source.
2. If AS2 bit in ASBR is set.
3. Only INT2 or level interrupt INT1 and INT0.



Figure 15. Reset Logic

Table 15. Reset Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Units
V_{POR}	Power-on Reset Threshold Voltage (rising)			1.4	2.3	V
	Power-on Reset Threshold Voltage (falling) ¹⁾			1.3	2.3	
V_{RST}	RESET Pin Threshold Voltage		0.1V _{CC}		0.9V _{CC}	
t_{RST}	Minimum pulse width on RESET Pin				1.5	μs
V_{BOR}	Brown-out Reset Threshold Voltage ²⁾	BODLEVEL = 1	2.5	2.7	3.2	V
		BODLEVEL = 0	3.6	4.0	4.5	
t_{BOD}	Minimum low voltage period for Brown-out Detection	BODLEVEL = 1		2		μs
		BODLEVEL = 0		2		
V_{IHWR}	Brown-out Detector hysteresis			50		mV

Notes: 1. The Power-on Reset will not work unless the supply voltage has been below V_{POR} (falling).
 2. V_{BOR} may be below nominal minimum operating voltage for some devices. For devices where this is the case, the device is tested down to $V_{CC} = V_{BOR}$ during the production test. This guarantees that a Brown-out Reset will occur before V_{CC} drops to a voltage where correct operation of the microcontroller is no longer guaranteed. The test is performed using BODLEVEL = 1 for ATmega16L and BODLEVEL = 0 for ATmega16. BODLEVEL = 1 is not applicable for ATmega16.



ATmega16 Typical Characteristics

The following charts show typical behavior. These figures are not tested during manufacturing. All current consumption measurements are performed with all I/O pins configured as inputs and with internal pull-ups enabled. A sine wave generator with rail-to-rail output is used as clock source.

The power consumption in Power-down mode is independent of clock selection.

The current consumption is a function of several factors such as: operating voltage, operating frequency, loading of I/O pins, switching rate of I/O pins, code executed and ambient temperature. The dominating factors are operating voltage and frequency.

The current drawn from capacitive loaded pins may be estimated (for one pin) as $C_L \cdot V_{CC} \cdot f$ where C_L = load capacitance, V_{CC} = operating voltage and f = average switching frequency of I/O pin.

The parts are characterized at frequencies higher than test limits. Parts are not guaranteed to function properly at frequencies higher than the ordering code indicates.

The difference between current consumption in Power-down mode with Watchdog Timer enabled and Power-down mode with Watchdog Timer disabled represents the differential current drawn by the Watchdog Timer.

Active Supply Current Figure 145. Active Supply Current vs. Frequency (0.1 MHz - 1.0 MHz)

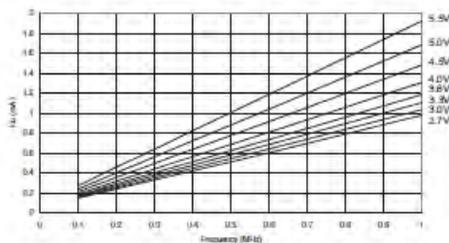


Figure 150. Active Supply Current vs. Frequency (1 MHz - 20 MHz)

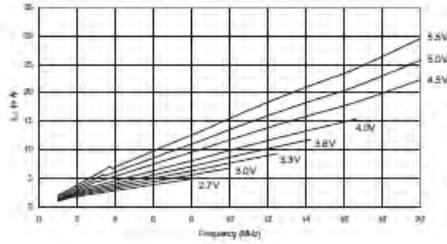
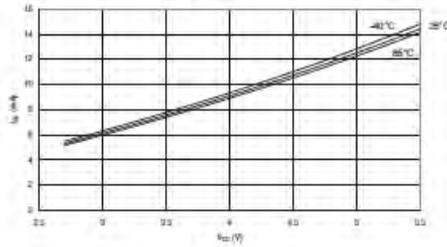


Figure 151. Active Supply Current vs. V_{CC} (Internal RC Oscillator, 8 MHz)



DAFTAR RIWAYAT HIDUP

RIWAYAT HIDUP PENULIS



Nama : Alfi Bahar
TTL : Bogor, 29 Oktober
1993
Jenis Kelamin : Laki-laki
Agama : Islam
Alamat Rumah : JL. Tirtaraja No 2
Jatirejo – Diwek -
Jombang
Telp/HP : 085706670178
E-mail : alfi.bahar@gmail.com
Hobi : Fotografi

RIWAYAT PENDIDIKAN

- 1999 – 2005 : MI Cukir
- 2005 – 2008 : MTs SS Tebuireng Jombang
- 2008 – 2011 : SMA Negeri 3 Jombang
- 2011 – sekarang : Bidang Studi Elektro Industri, Program D3
Teknik Elektro, ITS

PENGALAMAN KERJA

-

PENGALAMAN ORGANISASI

-