

PERANCANGAN DAN PEMBUATAN ALAT PENGAMANAN

**SEPEDA MOTOR MENGGUNAKAN RFID
BERBASIS MIKROKONTROLER ATMEGA 16**

SKRIPSI



Disusun Oleh :

**DAULAT KRISTON NAINGGOLAN
NIM: 02.17.057**



**PROGRAM STUDI TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2013**

**PERANCANGAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA
MOTOR MENGGUNAKAN RFID
BERBASIS MIKROKONTROLER ATMEGA 16**

Daulat Kriston Nainggolan, NIM 02.17.057
pakkatnabalau@yahoo.co.id

**Dosen Pembimbing : Irmalia Suryani Faradisa, ST, MT dan Yuli Wahyuni, ST,
MT**

ABSTRAK

Dewasa ini banyak orang yang meninggalkan sepeda motor di areal parkir atau dimana saja. Tetapi mereka tidak mengetahui bahwa sepeda motornya menjadi sasaran para pencuri. Sering pemiliknya terlambat mengetahui bahwa sepeda motornya telah hilang karena alat pengaman yang ada pada umumnya mereka pasang dapat dengan mudah dibobol pencuri. Dengan kemajuan teknologi seperti sekarang penulis mencoba membuat alat yang berfungsi untuk mengamankan sepeda motor.

Penulis mencoba membuat modul keamanan sepeda motor menggunakan RFID yang berbasis mikrokontroler AVR Atmega 16, yang berfungsi sebagai unit pengontrol alat yang dibuat. Sistem ini berfungsi sebagai pengganti kunci konvensional. Sistem keamanan sepeda motor ini dapat memberikan keamanan karena alat ini berfungsi jika RFID Tag yang dimasukkan dalam keadaan benar, maka motor tidak bisa menyala kalau data RFID Tagnya berbeda.

Sistem ini terdiri atas perangkat lunak dan perangkat keras. perangkat keras terdiri dari RFID Reader dan RFID Tag sebagai inputan perangkat lunak terdiri dari bahasa basic dengan BASCOM-AVR IDE yang diisikan dalam mikrokontroler sebagai pemproses data. Sistem ini tidak dapat bekerja jika secara bersamaan ada lebih dari satu RFID Tag dideteksi sekaligus oleh RFID Reader. Dimana tujuan pembuatan system ini untuk memberikan alternatif pengamanan.

Kata kunci: Mikrokontroler, RFID, Pengamanan Sepeda Motor

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Tuhan Yang Maha Mulia atas berkah dan rahmat-Nya penulis dapat menyelesaikan tugas skripsi yang berjudul :

“PERANCANGAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR MENGGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16”

Penulisan skripsi ini disusun untuk memenuhi syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro S-1 Konsentrasi Teknik Elektronika di Institut Teknologi Nasional Malang. Terima kasih yang mendalam penulis haturkan kepada kedua orang tua penulis “P. Nainggolan dan T. Br. Silaban” yang telah berjasa membentuk kepribadian penulis, khususnya memberikan inspirasi yang kuat serta dukungan dan doa yang tiada henti. Terimakasih juga penulis sampaikan kepada :

1. Bapak Ir. Soeparno Djiwo, MT selaku rektor ITN Malang.
 2. Bapak Ir. Anang Subardi, MT selaku Dekan Fakultas Teknologi Industri ITN Malang.
 3. Bapak M. Ibrahim Ashari, ST, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
 4. Bapak Dr. Eng. Aryunto Soetedjo, ST, MT selaku Sekretaris Jurusan Teknik Elektro S-1 ITN Malang.
 5. Ibu Irmalia Suryani Faradisa, ST, MT selaku Dosen Pembimbing I.
 6. Ibu Yuli Wahyuni, ST, MT selaku Dosen Pembimbing II.
 7. Ibu Puji Handayani selaku Recording Jurusan Teknik Elektro ITN Malang.
-

8. Keluarga besar Op. Betram Nainggolan : Kel. Risma Nainggolan, Kel. Ana Nainggolan, Kel. Bidicoff Nainggolan, Kel. Ermita Nainggolan dan John Nainggolan.
9. Keponakan-keponakanku : Gabe, Bagas, Luisa, Enzo, Bertram, Benheart dan Rafael.
10. Sahabat-sahabat penulis yang ada di Malang, akka halak hita.
11. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa skripsi ini masih jauh dari sebuah kesempurnaan, untuk itu kritik dan saran yang membangun dari pembaca sangat penulis harapkan untuk perbaikan penelitian ini.

Akhir kata penulis mohon maaf kepada semua pihak jika dalam proses pembuatan skripsi ini penulis melakukan kesalahan baik yang disengaja maupun tidak disengaja.

Semoga Allah Bapa selalu memberkati dan memberi jalan yang terbaik bagi kita semua. Amin.

Malang, September 2013

Penulis
(Daulat Kriston Nainggolan)

DAFTAR ISI

Lembar Persetujuan	i
Pernyataan Originalitas.....	ii
Abstrak	iii
Kata Pengantar	iv
Daftar Isi	vi
Daftar Tabel	viii
Daftar Gambar.....	ix
BAB I Pendahuluan	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metode Penelitian	2
1.6 Sistematika Penulisan	3
BAB II Landasan Teori	4
2.1. Radio Frequency Identification (RFID)	4
2.2. RFID Reader	6
2.3. RFID Tag (Transponder)	6
2.4. Mikrokontroler ATMEGA 16	8
2.4.1 Arsitektur ATMEGA 16	9
2.4.2 Konfigurasi ATMEGA 16	10
2.5 Relay	11
BAB III Perancangan Sistem	14
3.1 Blok Diagram Rangkaian	14
3.2 Perancangan Sistem Perangkat Keras	15
3.2.1 Tag RFID	15
3.2.2 Perancangan Rangkaian Modul RFID Reader	15
3.2.3 Perancangan Rangkaian Mikrokontroler	17
3.2.4 Perancangan Driver Relay	17
3.3 Diagram Alir Sistem	19
BAB IV Pembuatan Dan Pengujian Sistem	20
4.1 Pengujian Modul RFID	20

4.2 Pengujian Jarak Pembacaan RFID Reader Terhadap Tag RFID Dari Berbagai Arah	21
4.2.1 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Atas	21
4.2.2 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Kanan	22
4.2.3 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Kiri	23
4.3 Pengujian Pembacaan Multi Tag	24
4.4 Pengujian Pembacaan Kode Tag RFID	24
4.5 Pengujian Sistem Secara Keseluruhan	27
BAB V Penutup	29
5.1 Kesimpulan	29
5.2 Saran	29
Daftar Pustaka	30
Daftar Lampiran	31

DAFTAR TABEL

4.1. Pengujian Modul RFID Tanpa Difungsikan	21
4.2. Jarak Pengukuran RFID Reader Terhadap Tag RFID Pada Arah Atas	22
4.3. Jarak Pengukuran RFID Reader Terhadap Tag RFID Pada Arah Kanan	23
4.4. Jarak Pengukuran RFID Reader Terhadap Tag RFID Pada Arah Kiri	24
4.5. Hasil Pengujian Multi Tag	25
4.6. Hasil Tag Terdata dan Tidak Terdata	28

DAFTAR GAMBAR

2.1. Data Transfer RFID Reader dan Tag RFID	6
2.2. Penggolongan RFID Tag	8
2.3. Konfigurasi ATMEGA 16	11
2.4. Rangkaian Darlington	13
3.1. Blok Diagram	15
3.2. Rangkaian Modul RFID Reader	16
3.3. Konfigurasi Pin RFID Reader	17
3.4. Rangkaian Minimum Sistem Mikrokontroler	18
3.5. Rangkaian Relay	19
3.6. Flowchart Sistem	20
4.1. Hasil Pengujian Tag Pertama	26
4.2. Hasil Pengujian Tag Kedua	26
4.3. Hasil Pengujian Tag Ketiga	27
4.4. Hasil Pengujian Tag Keempat	27
4.5. Hasil Pengujian Tag Kelima	28

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teknologi memegang peran penting di era modernisasi seperti pada saat ini, dimana teknologi telah menjadi bagian yang tidak dapat dipisahkan dalam kehidupan sehari – hari. Sehingga mendorong manusia untuk menciptakan sebuah alat yang lebih aman sehingga membantu mempermudah dan mempercepat manusia dalam menyelesaikan pekerjaan dengan hasil yang sebaik-baiknya. Dimana kita bisa mengambil contoh permasalahan yang terdapat pada pengamanan sepeda motor.

Dewasa ini sepeda motor telah menjadi alat transportasi yang paling banyak digunakan di jalanan di Indonesia, dikarenakan harga sepeda motor terjangkau bagi semua kalangan masyarakat. Umumnya untuk pengamanan pabrikan hanya memberikan kunci konvensional. Pencurian sepeda motor sering kita dengar baik secara langsung maupun lewat media massa. Dalam hal pengamanan tambahan sepeda motor sudah sering kita temukan penggunaan alat tambahan seperti penggunaan remote kontrol, maupun dengan menggunakan jasa pengamanan seperti satpam atau petugas parkir.

Dalam hal ini penulis mencoba memberikan opsi lain dalam pengamanan sepeda motor dengan menggunakan RFID (Radio Frequency Identification). RFID sering kita temukan pemakaiannya seperti kartu ATM, KTP Elektronik, kunci kamar hotel, dll. Aplikasi ini menggunakan mikrokontroler sebagai pengontrol utama.

Alat ini menggunakan RFID sebagai pengamanan tambahan, karna kunci konvensional sepeda motor masih digunakan. RFID reader digunakan sebagai pembaca, sedangkan Tag RFID digunakan sebagai kartu identitas atau pengenalan ketika hendak menyalakan sepeda motor. Tag RFID berisi kode-kode yang unik, dimana setiap tag berisi kode yg berbeda-beda. Proses pengamanan sepeda motor ini dimulai dengan menyalakan kunci konvensional sepeda motor ke posisi on, kemudian tag RFID didekatkan ke reader. Reader akan membaca data yang ada pada tag, apabila data pada tag tidak terdata maka motor tidak akan bisa dinyalakan. Sedangkan ketika reader membaca tag dan datanya sesuai, mikrokontroler akan memberikan perintah kepada relay untuk menghubungkan arus pada CDI.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, telah diambil permasalahan yang akan dibahas dalam perancangan dan pembuatan alat ini adalah:

1. Bagaimana membuat dan menempatkan RFID Reader pada sepeda motor.
2. Bagaimana menghubungkan RFID Reader dengan mikrokontroler.
3. Bagaimana membuat instalasi hardware dari mikrokontroler.
4. Bagaimana membuat software pada mikrokontroler ATMEGA 16.

1.3 Tujuan

Tujuan dari penulisan skripsi ini adalah:

1. Untuk merancang dan membuat alat pengamanan sepeda motor menggunakan RFID berbasis mikrokontroler ATMEGA 16.
2. Mempelajari prinsip kerja mikrokontroler ATMEGA 16.
3. Mengaplikasikan teori yang telah didapat dibangku kuliah untuk membuat suatu alat yang sesuai teori yang telah didapat.

1.4 Batasan Masalah

Agar permasalahan mengarah sesuai dengan tujuan yang diharapkan, maka pembahasan dibatasi oleh hal-hal sebagai berikut :

1. Tidak membahas tentang struktur dan jaringan instalasi listrik sepeda motor.
2. Alat akan dikontrol menggunakan mikrokontroler AT MEGA 16.
3. Hanya membahas perancangan dan pembuatan model pengamanan sepeda motor menggunakan RFID dengan mikrokontroler AT MEGA 16.

1.5 Metode Penelitian

Adapun metode penelitian yang digunakan adalah sebagai berikut:

1. Studi Literatur

Pengumpulan data yang dilakukan dengan mencari bahan-bahan kepustakaan dan referensi dari berbagai sumber sebagai landasan teori yang ada hubunganya dengan permasalahan yang dijadikan objek penelitian.

2. Analisa Kebutuhan Sistem

Data dan informasi yang telah diperoleh akan dianalisa agar didapatkan

kerangka global yang bertujuan untuk mendefinisikan kebutuhan sistem di mana nantinya akan digunakan sebagai acuan perancangan sistem.

3. Perancangan dan Implementasi

Berdasarkan data dan informasi yang telah diperoleh serta analisa kebutuhan untuk membangun sistem ini, akan dibuat rancangan kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat dan diimplementasikan kedalam sistem.

4. Eksperimen dan Evaluasi

Pada tahap ini, sistem yang telah selesai dibuat akan diuji coba, yaitu pengujian berdasarkan fungsionalitas program, dan akan dilakukan koreksi dan penyempurnaan program jika diperlukan.

1.6 Sistematika Penulisan

Untuk mempermudah dan memahami pembahasan penulisan skripsi ini, maka sistematika penulisan disusun sebagai berikut :

Bab I : Pendahuluan

Berisi Latar Belakang, Rumusan Masalah, Tujuan Penelitian, Pembatasan Permasalahan, Metode Penelitian dan Sistematika Penulisan.

Bab II : Tinjauan Pustaka

Berisi tentang landasan teori mengenai permasalahan yang berhubungan dengan penelitian yang dilakukan.

Bab III : Perancangan Sistem

Dalam bab ini berisi mengenai analisa kebutuhan sistem baik software maupun hardware yang diperlukan untuk membuat kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat.

Bab IV : Pembuatan dan Pengujian Sistem

Berisi tentang implementasi dari perancangan sistem yang telah dibuat serta pengujian terhadap sistem tersebut.

Bab V : Penutup

Merupakan bab terakhir yang memuat intisari dari hasil pembahasan yang berisikan kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk pengembangan penulisan selanjutnya.

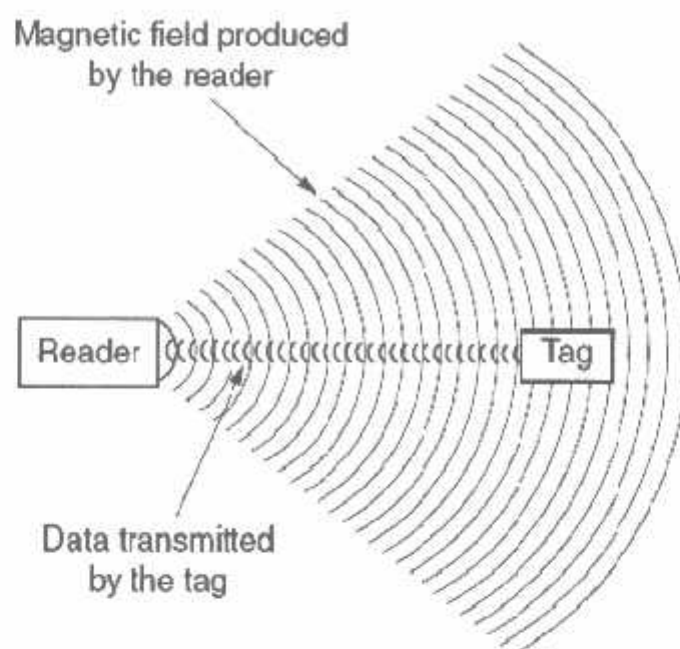
BAB II LANDASAN TEORI

2.1 Radio Frequency Identification (RFID)

Radio Frequency Identification (RFID) adalah proses identifikasi seseorang atau objek dengan menggunakan frekuensi transmisi radio. Radio frequency identification (RFID) menggunakan frekuensi radio untuk membaca informasi dari sebuah devais kecil yang disebut tag atau transponder (Transmitter + Responder). Tag RFID akan mengenali diri sendiri ketika mendeteksi sinyal dari devais yang kompatibel, yaitu pembaca RFID (RFID Reader).

RFID merupakan teknologi yang berfungsi untuk melakukan deteksi dan identifikasi terhadap suatu obyek melalui data yang ditransmisikan melalui frekuensi radio. Sistem tersebut minimal memerlukan sebuah tag (yang berfungsi sebagai transponder), sebuah reader (yang berfungsi sebagai interrogator) dan sebuah antena (yang berfungsi sebagai coupling device). Reader biasanya terhubung dengan sebuah host computer atau perangkat lainnya yang memiliki kecerdasan untuk memproses lebih lanjut tag data dan memutuskan untuk mengambil suatu tindakan. Salah satu elemen penting pada RFID adalah data transfer. Data transfer terjadi ketika terjadi hubungan antara sebuah tag dengan sebuah reader, yang dikenal dengan coupling, melalui antenna baik yang terpasang pada tag tersebut maupun pada.

Coupling pada kebanyakan sistem-sistem RFID menggunakan metode magnetic (inductive) atau electromagnetic (backscatter). Metode yang digunakan tersebut bergantung pada harga, ukuran, kecepatan, dan jangkauan pembacaan serta keakuratan. Pada umumnya komunikasi antara sebuah tag dengan sebuah reader terjadi melalui sebuah physical principle yang dikenal sebagai sebuah backscatter modulation. Pada proses ini, sebuah reader mengirimkan sinyal kepada sebuah tag, dan tag akan menanggapi dengan memantulkan sebagian dari energi ini kembali ke reader. Hal ini dapat diilustrasikan dari gambar 2.1.



Gambar 2.1 Data Transfer RFID Reader dan Tag RFID

Dalam memilih RFID tersebut, terdapat beberapa hal yang harus diperhatikan, yaitu:

1. Jenis tag yang digunakan.
2. Jenis reader yang dipakai.
3. Frekuensi operasional dari sistem.
4. Jarak antara reader dan tag yg diinginkan.

RFID terbagi dari dua komponen, yaitu :

1. RFID Tag
Alat yang menyimpan informasi untuk identifikasi objek.
2. RFID Reader
Alat yang kompatibel dengan RFID Tag yang berkomunikasi secara wireless dengan RFID Tag.

2.2 RFID Reader

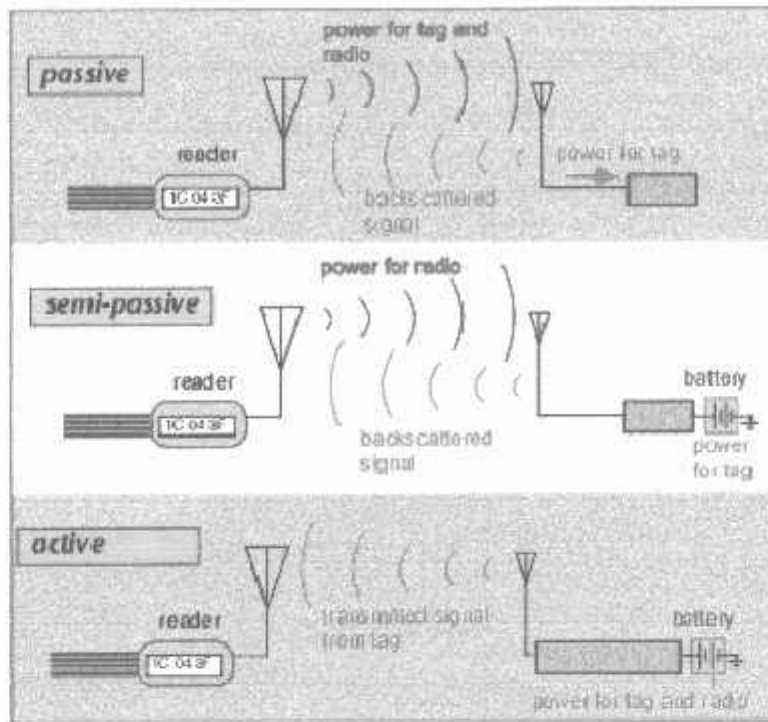
RFID reader memiliki high frequency module (transmitter dan receiver), control module dan juga coupling element (coil dan microwave antenna), yang berfungsi membangkitkan sinyal untuk mengaktifkan RFID tags, sehingga dapat melakukan pengiriman dan penerimaan data. Ketika sebuah RFID tags melewati medan elektromagnetik RFID reader, maka RFID tags tersebut akan mendeteksi sinyal pengaktifan dari reader, dan mengirimkan sinyal balik untuk pemrosesan data yang telah tersimpan dalam memori tag sebagai respon. RFID reader kemudian menterjemahkan data yang dikirimkan oleh RFID tags tersebut sesuai dengan kebutuhan. Proses pembacaan kode-kode data yang terdapat pada RFID tags dilakukan menggunakan gelombang radio, sehingga proses pengidentifikasian menjadi lebih mudah.

2.3 RFID Tag (Transponder)

Tag ID adalah devais yang dibuat dari rangkaian elektronika dan antena yang terintegrasi di dalam rangkaian tersebut. Rangkaian elektronik dari tag RFID umumnya memiliki memori sehingga tag ini mempunyai kemampuan untuk menyimpan data. Memori pada tag dibagi menjadi sel-sel. Beberapa sel menyimpan data Read Only, misalnya serial number yang unik yang disimpan pada saat tag tersebut diproduksi. Sel lain pada RFID mungkin juga dapat ditulis dan dibaca secara berulang.

Tag ini bekerja saat antena mendapatkan sinyal dari reader dan sinyal tersebut akan dipantulkan lagi, sinyal pantul ini biasanya sudah ditambahkan dengan data yang dimiliki tag tersebut. RFID tag ukurannya dapat berbeda-beda, umumnya ukurannya kecil.

Berdasarkan catu daya tag, tag RFID dapat digolongkan menjadi 3 bagian seperti gambar di bawah:



Gambar 2.2 Penggolongan RFID Tag

➤ Tag Aktif

Tag yang catu dayanya diperoleh dari batere, sehingga akan mengurangi daya yang diperlukan oleh RFID Reader. Dan tag dapat mengirimkan informasi dalam jarak yang lebih jauh.

Kelemahan dari tag tipe ini adalah harga yang mahal dan ukurannya yang lebih besar karena lebih kompleks. Semakin banyak fungsi yang dapat dilakukan oleh tag, maka rangkaianannya akan semakin kompleks dan ukurannya semakin besar.

➤ Tag Pasif

Tag yang catu dayanya diperoleh dari medan yang dihasilkan oleh pembaca RFID. Rangkaianannya lebih sederhana, harganya jauh lebih murah, ukurannya kecil dan lebih ringan.

Kelemahannya adalah tag hanya dapat mengirimkan informasi dalam jarak yang dekat dan pembaca RFID harus menyediakan daya tambahan untuk tag RFID.

➤ Tag Semi Pasif

Tag pasif yang menggunakan energi dari baterai untuk keperluan rangkaian, tetapi tidak untuk menghasilkan sinyal. Ketika baterai digunakan untuk memberikan energi pada sensor, maka ini disebut sensor tag. Tag ini cenderung lebih kecil dan lebih murah jika dibandingkan dengan tag aktif, juga memiliki fungsi lebih jika dibandingkan dengan tag pasif karena lebih banyak energi yang tersedia untuk kegunaan lain.

2.4 Mikrokontroler ATMEGA 16

Mikrokontroler dewasa ini menjadi booming dikalangan bidang pendidikan, terutama perguruan tinggi. Mikrokontroler dengan merek ATMEL dengan tipenya meliputi ATtiny, AT90 dan Atmega. Masing-masing tipe memiliki fitur yang berbeda-beda dengan tujuan aplikasi yang spesifik.

Mikrokontroler sebagai sebuah "one chip solution" pada dasarnya adalah rangkaian terintegrasi (Integrated Circuit-IC) yang telah mengandung secara lengkap berbagai komponen pembentuk sebuah komputer. Berbeda dengan penggunaan mikroprosesor yang masih memerlukan komponen luar tambahan seperti RAM, ROM, Timer, dan sebagainya untuk sistem mikrokontroler, tambahan komponen diatas secara praktis hampir tidak dibutuhkan lagi. Hal ini disebabkan semua komponen penting tersebut telah ditanam bersama dengan sistem prosesor ke dalam IC tunggal mikrokontroler bersangkutan. Dengan alasan itu sistem mikrokontroler dikenal juga dengan istilah populer The Real Computer On a Chip (komputer utuh dalam keping tunggal), sedangkan sistem mikroprosesor dikenal dengan istilah yang lebih terbatas yaitu Computer On a Chip (komputer dalam keping tunggal).

Berdasarkan fungsinya, mikrokontroler secara umum digunakan untuk menjalankan program yang bersifat permanen pada sebuah aplikasi yang spesifik (misal aplikasi yang berkaitan dengan pengontrolan dan monitoring). Sedangkan program aplikasi yang dijalankan pada sistem mikroprosesor biasanya bersifat sementara dan berorientasi pada pengolahan data. Perbedaan fungsi kedua sistem diatas secara praktis mengakibatkan kebutuhan minimal yang harus dipenuhi juga akan berbeda (misal ditinjau dari kecepatan detak operasi, jumlah RAM, panjang register, dan lain sebagainya). Hampir tidak dapat disangkal, dewasa ini akan sukar dijumpai seseorang yang masih menggunakan komputer dengan mikroprosesor berbasis 8 atau 16 bit (misal mikroprosesor 8088 dan 8086 produk perusahaan Intel).

Mengapa demikian? Salah satu alasannya, perangkat lunak komputer yang beredar saat ini umumnya mensyaratkan kecepatan CPU yang sangat tinggi (dalam orde Mega bahkan GigaHz) serta memori dengan kapasitas sangat besar (dalam orde MegaByte) yang mana hal tersebut tidak mungkin dapat dipenuhi oleh sistem microprocessor lama tersebut. Sedangkan untuk sistem mikrokontroler, program yang dijalankan biasanya tidak memerlukan sumber daya sebanyak dan sebesar itu. Untuk aplikasi kontrol sederhana dan tingkat menengah, mikrokontroler yang digunakan cukup berbasis 4 sampai 8 bit. Mikrokontroler dengan ukuran lebih besar (misal 16 dan 32 bit) umumnya hanya digunakan untuk aplikasi-aplikasi khusus pada bidang pengolahan citra atau bidang kontrol yang memerlukan kepresisian tinggi.

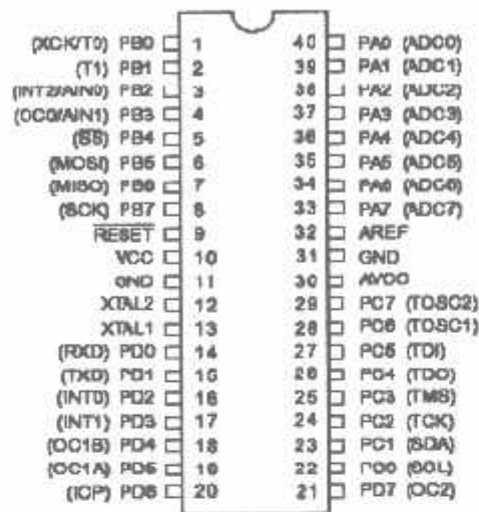
2.4.1 Arsitektur ATMEGA 16

Mikrokontroler ini menggunakan arsitektur Harvard yang memisahkan memori program dari memori data, baik bus alamat maupun bus data, sehingga pengaksesan program dan data dapat dilakukan secara bersamaan (concurrent). Secara garis besar mikrokontroler ATMEGA 16 terdiri dari:

1. Frekuensi clock maksimum 16 Mhz.
2. Kapasitas Flash memori 8 Kbyte dengan kemampuan read while write.
3. EEPROM (Electrically Erasable Program-mable Read Only Memory) sebesar 512 Byte yang dapat diprogram saat operasi.
4. SRAM internal sebesar 1 Kilo byte.
5. Jalur I/O 32 buah, yang terbagi dalam Port A, Port B, Port C dan Port D.
6. CPU 8 bit yang terdiri dari 32 buah register.
7. Interrupt internal maupun eksternal.
8. Port komunikasi SPI (Serial Pheripheral Interface)
9. Komunikasi serial standar USART dengan kecepatan maksimal 2,5 Mbp.
10. Analog komparator.
11. Watchdog timer dengan osilator internal.
12. Timer/counter sebanyak 3 buah.
13. Analog to Digital Converter (ADC) 10 bit sebanyak 8 input.

2.4.2 Konfigurasi ATMEGA 16

Konfigurasi pin-pin pada ATMEGA 16 ditunjukkan pada gambar 2.1 sebagai berikut:



Gambar 2.3 Konfigurasi ATMEGA 16

Konfigurasi pin mikrokontroler ATMEGA 16 terdiri dari 40 pin. Dari gambar di atas dapat terlihat ATMEGA 16 memiliki 8 pin untuk setiap portnya, port A, port B, port C dan port D.

Deskripsi Mikrokontroler ATMEGA 16

- VCC
 - Sebagai tegangan penyuplai.
- Ground
 - Sebagai ground
- Port A (PA7 ... PA0)
 - Port A sebagai input analog ke A/D konverter. Port A juga sebagai 8-bit bi-directional port I/O, jika A/D konverter tidak digunakan. Pin-pin port dapat menyediakan resistor-resistor internal pull-up. Ketika port PA0...PA7 digunakan sebagai input dan pull eksternal yang rendah akan menjadi sumber arus jika resistor-resistor pull-up diaktifkan. Pin-pin port A adalah tri-state ketika kondisi reset menjadi aktif sekalipun clock tidak aktif.
- Port B (PB7 ... PB0)

Port B adalah port I/O 8-bit bi-directional dengan resistor-resistor internal pull-up. Buffer output port B mempunyai karakteristik drive yang simetris dengan kemampuan keduanya sink dan source yang tinggi. Sebagai input, port B yang mempunyai pull eksternal yang rendah akan menjadi sumber arus jika resistor-resistor pull-up diaktifkan. Pin-pin port B adalah tri-state ketika kondisi reset menjadi aktif sekalipun clock tidak aktif.

➤ Port C (PC7 ... PC0)

Port C adalah port I/O 8-bit bi-directional dengan resistor-resistor internal pull-up. Buffer output port C mempunyai karakteristik drive yang simetris dengan kemampuan keduanya sink dan source yang tinggi. Sebagai input, port C yang mempunyai pull eksternal yang rendah akan menjadi sumber arus jika resistor-resistor pull-up diaktifkan. Pin-pin port C adalah tri-state ketika kondisi reset menjadi aktif sekalipun clock tidak aktif. Jika antarmuka JTAG enable, resistor-resistor pull up pada pin-pin PC5(TDI), PC3(TMS), PC2(TCK) akan diaktifkan sekalipun terjadi reset.

➤ Port D (PD7 ... PD0)

Port D adalah port I/O 8-bit bi-directional dengan resistor-resistor internal pull-up. Buffer output port D mempunyai karakteristik drive yang simetris dengan kemampuan keduanya sink dan source yang tinggi. Sebagai input, port D yang mempunyai pull eksternal yang rendah akan menjadi sumber arus jika resistor pull-up diaktifkan.

➤ RESET (Reset input)

➤ XTAL1 (Input oscillator)

➤ XTAL2 (Output oscillator)

➤ AVCC

Pin penyedia tegangan untuk Port A dan konverter A/D.

➤ AREF

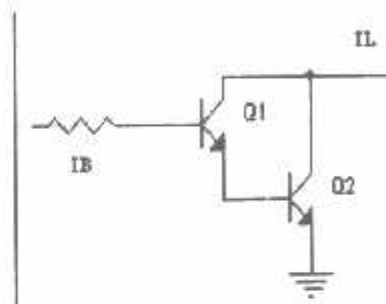
Pin referensi analog untuk konverter A/D.

2.5 Relay

Relay merupakan salah satu komponen output yang paling sering digunakan baik pada industri, otomotif, ataupun peralatan elektronika lainnya. Berfungsi sebagai pemutus dan penghubung aliran arus ke sistem pengapian dan kelistrikan dan alarm.

Relay adalah sebuah saklar auto-mekanik. Relay terdiri dari kumparan yang dapat menimbulkan medan magnet jika dialiri arus dan akan menarik beberapa kontakannya.

Ada 2 macam relay berdasarkan tegangan untuk menggerakkan koilnya yaitu AC dan DC. Pada umumnya relay DC dengan tegangan koil 12V DC, memerlukan arus sekitar 20-30mA. Karena itu tidak dapat langsung menghubungkan output suatu IC logic (TTL/CMOS) atau peripheral lain seperti μC 89C51, PPI 82C55 dengan relay karena I_{OHmax} (arus maximum yang dikeluarkan pada saat logic '1') atau I_{OLmax} (arus maximum yang mampu dibenamkan pada saat logic '0') tidak cukup besar. Karena itu perlu digunakan driver untuk penguat arus yang biasanya berupa transistor dalam konfigurasi Darlington seperti Gambar 2.4 berikut ini.



Gambar 2.4 Rangkaian Darlington

Rangkaian tersebut juga dilengkapi dengan suppression diode. Dioda ini berfungsi untuk mencegah kickback yaitu transient yang terjadi pada koil relay (beban induktif) saat relay dimatikan. Bila terjadi perubahan arus yang cukup besar dalam satuan waktu yang sangat cepat maka tegangan balik ini menjadi sangat besar, dan dapat mengakibatkan kerusakan pada transistor.

Problem lain yang sering terjadi pada kontak relay adalah loncatan bunga api listrik yang dapat memperpendek umur kontak. Bunga api ini terutama terjadi pada beban induktif seperti motor, solenoid, dll. Untuk mencegah hal ini dapat digunakan MOV (Metallic Oxide Varistor) yang dipasang secara paralel dengan kontak. Varistor bersifat seperti resistor dengan nilai resistansinya tergantung pada tegangan. Ketika kontak terbuka, beban induktif menghasilkan tegangan balik yang cukup besar akibat perubahan medan magnet.

Pada saat ini ('protective state') nilai resistansi varistor menjadi sangat kecil dan arus akan mengalir melalui MOV, sehingga transient dapat diredam. Pada saat keadaan normal resistansi MOV sangat besar dan hanya menarik arus yang sangat kecil, bunga api juga menyebabkan sinyal Radio Frequency Interference (RFI) yang dapat mengganggu peralatan – peralatan sensitif.

Karena itu komponen peredam transient seperti MOV sangat diperlukan terutama pada beban induktif. Parameter-parameter penting Varistor :

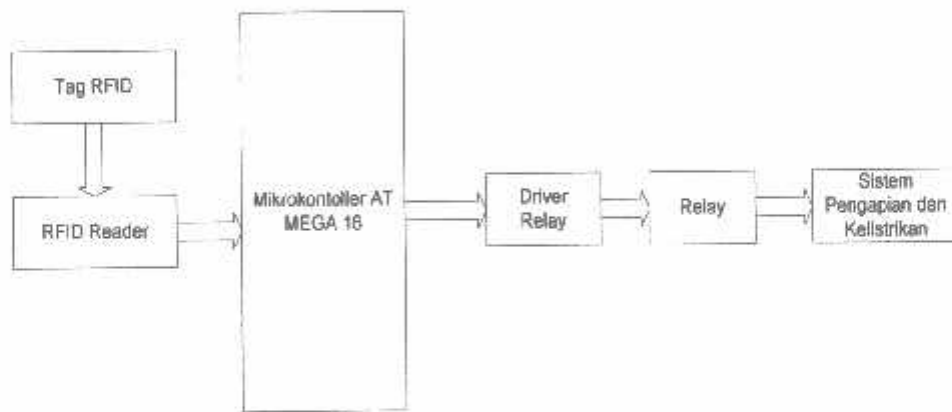
- Tegangan Varistor (tegangan breakdown):
Tegangan pada varistor yang diukur pada arus tertentu (0,1 mA atau 1 mA) selama waktu tertentu.
- Tegangan maksimum yang diperbolehkan:
Tegangan maksimum pada varistor agar tetap pada keadaan normal.

BAB III PERANCANGAN SISTEM

Perancangan sistem pengamanan sepeda motor ini terbagi atas dua bagian, yaitu perancangan hardware dan perancangan software. Perancangan hardware terbagi atas perancangan sistem control, perancangan unit masukan, perancangan dan unit keluaran. Sedangkan perancangan software terdiri dari perancangan program basic (BASCOM).

3.1 Blok Diagram Rangkaian

Pada bagian ini penulis akan membahas tentang alat yang meliputi diagram blok dan realisasi rangkaian. Komponen dalam alat ini dirangkai berdasarkan prinsip kerja masing-masing komponen. Diagram blok rangkaian sebagai berikut:



Gambar 3.1 Blok Diagram

Cara Kerja Kunci Pengaman Sepeda Motor Menggunakan RFID

Adapun cara kerja rangkaian dari kunci kendaraan bermotor menggunakan RFID adalah sebagai berikut ini.

Kartu tag ID dan RFID yang digunakan adalah sebagai media input utama yang berfungsi sebagai pengaman kendaraan bermotor. Kartu tag ID berisikan kode-kode spesifik atau unik dimana setiap kartu tag ID memiliki kode-kode yang berbeda-beda. Hal inilah yang dimanfaatkan sebagai kunci untuk menghidupkan atau memutuskan kabel saklar dan busi serta alaram yang terpasang. Untuk menghidupkan kontak kendaraan bermotor kita terlebih dahulu mengidentifikasi kartu tag ID ke RFID, jika

kartu tag ID memiliki kode yang sesuai dengan input pada mikrokontroller maka mikrokontroller akan memberikan output pada relay menyalakan motor.

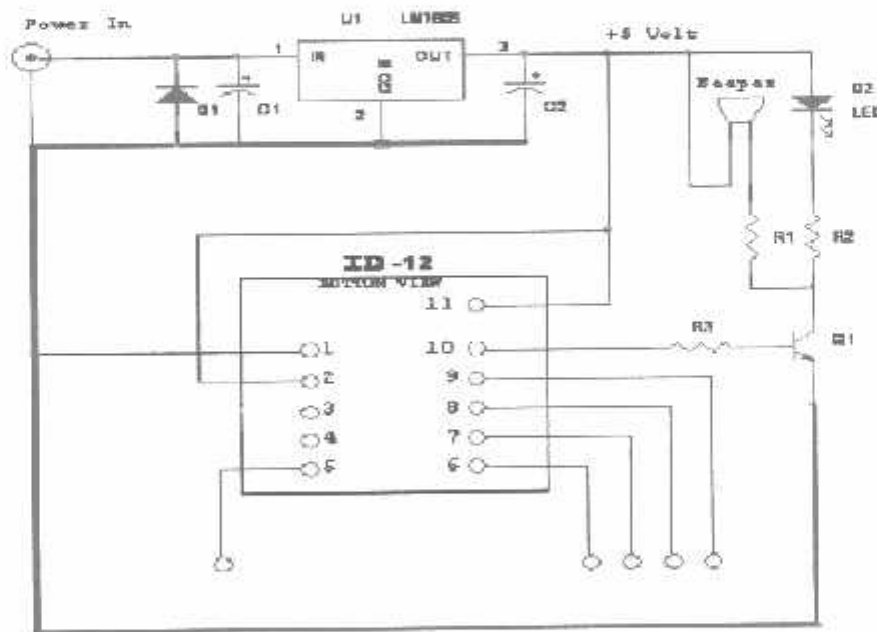
3.2 Perancangan Sistem Perangkat Keras

3.2.1 Tag RFID

Tag RFID adalah alat yang dibuat dari rangkaian elektronika dan antena yang terintegrasi di dalam rangkaian tersebut. Rangkaian elektronik dari Tag RFID umumnya memiliki memori sehingga Tag RFID ini mempunyai kemampuan untuk menyimpan data. Memori pada Tag RFID dibagi menjadi sel-sel. Beberapa sel menyimpan data Read Only, misalnya serial number yang unik yang disimpan pada saat Tag RFID tersebut diproduksi. Sel lain pada Tag RFID mungkin juga dapat ditulis dan dibaca secara berulang. Tag yang digunakan adalah Tag Card Pasif yaitu Tag RFID yang catu dayanya diperoleh dari medan listrik yang dihasilkan oleh RFID Reader, rangkaianannya lebih sederhana, dan harganya jauh lebih murah, ukurannya kecil, dan lebih ringan. Kelemahannya adalah Tag RFID hanya dapat mengirimkan informasi dalam jarak yang dekat.

3.2.2 Rangkaian Modul RFID

Jenis RFID Reader yang digunakan adalah tipe ID-12 buatan Innovative Electronics.

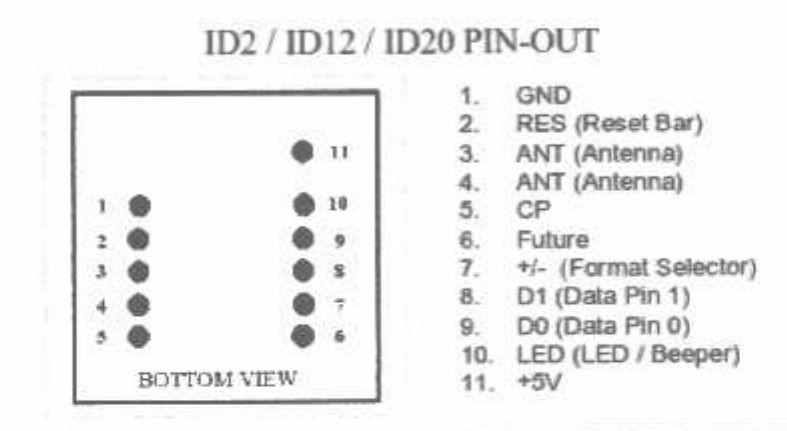


Gambar 3.2 Rangkaian Modul RFID Reader

Keterangan komponen :

R1 = 100R

R2 = 1K
 R3 = 1K
 C1 = 100 μ F 16V
 C2 = 100 μ F 10V
 Beeper = 2,7-3.5 KHz 100R
 D1 = 1N4001
 D2 = Green LED
 U1 = LM7805
 Q1 = UTC8050 (NPN)
 ID-12 = ID Innovations ID-12



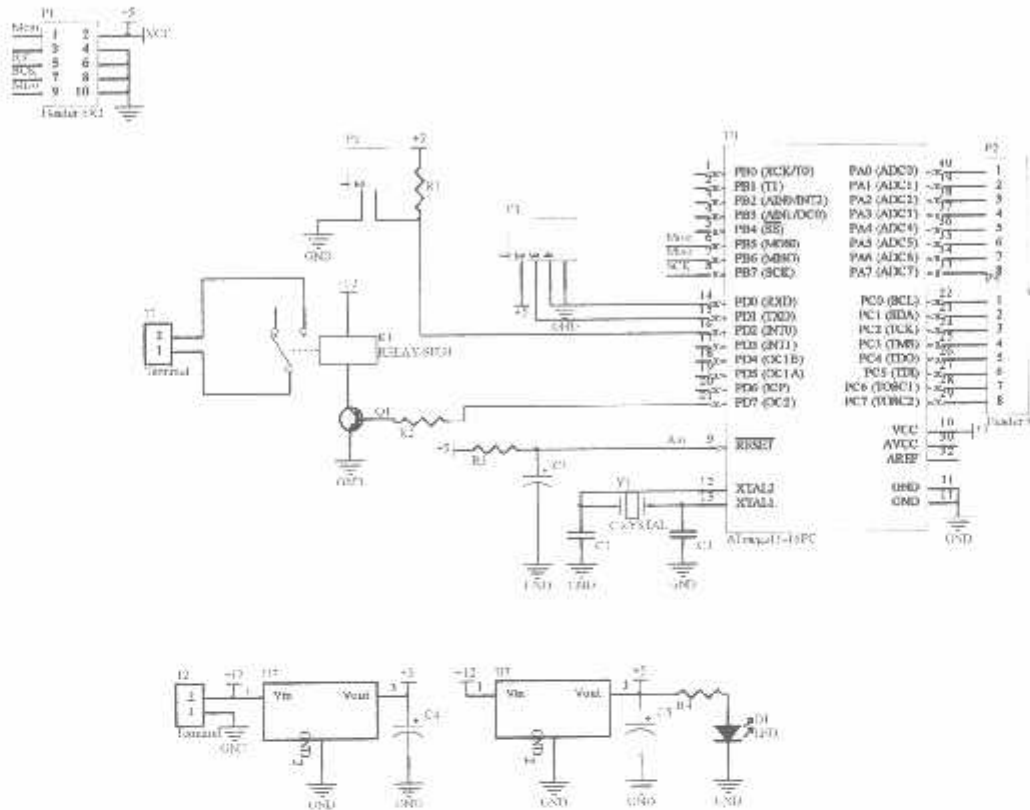
Gambar 3.3 Konfigurasi Pin RFID Reader

Spesifikasi Modul RFID :

1. Berbasis RFID reader ID-12 dengan frekuensi 125 kHz untuk kartu berformat EM401/sejenis dan memiliki jarak baca 12 cm.
2. Kompatibel dengan varian RFID reader lainnya, antara lain : ID-2, ID-10 dan ID-20.
3. Mendukung varian RFID reader/writer, antara lain : ID-2RW, ID-12RW dan ID-20RW.
4. Mendukung format data ASCII (UART TTL/ RS-232), Wiegand26, maupun Magnetic ABA Track2 (Magnet Emultion).
5. Dilengkapi dengan buzzer sebagai indikator baca, serta LED sebagai indikator tulis.
6. Tersedia jalur komunikasi serial UART RS-232 dengan konektor RJ11.
7. Tegangan input catu daya 9-12 VDC (J2)

3.2.3 Perancangan Mikrokontroler

Jenis mikrokontroler yang digunakan pada sistem ini adalah Atmega16, yang memiliki saluran I/O sebanyak 32 buah, yaitu Porta A, Port B, Port C, dan Port D serta memiliki ADC 10 bit sebanyak 8 saluran. Mikrokontroler dalam sistem ini berfungsi untuk menerima dan mengelola data yang dikirim dari RFID Reader setelah membaca data pada ID Tag.

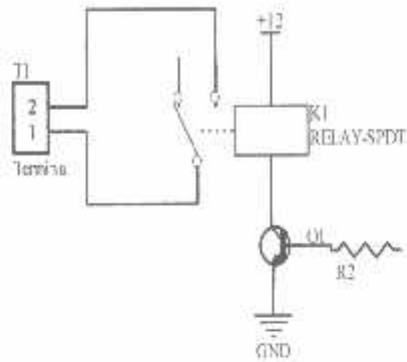


Gambar 3.4 Rangkaian Minimum Sistem Mikrokontroler

3.2.4 Perancangan Driver Relay

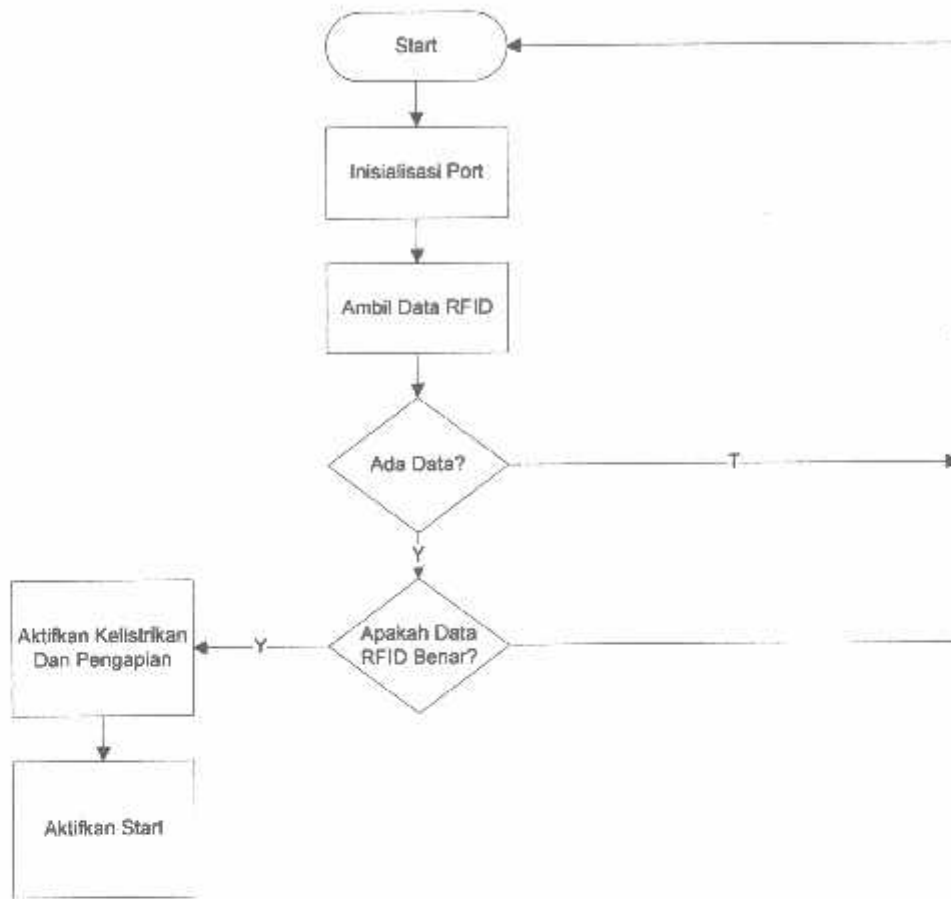
Rangkaian driver relay berasal dari rangkaian mikrokontroler. Terdapat 8 jalur keluaran dari unit sistem control alat, di gunakan 1 jalur yang dihubungkan ke relay. Tegangan keluaran dari mikrokontroler dihubungkan ke kaki basis transistor Q1 dan Q2 melalui resistor pembatas arus R2. Dimana jika masukan pada kaki basis bertegangan tinggi maka transistor Q1 akan menghantar arus melalui relay, sehingga saklar relay akan otomatis menutup dan arus listrik pada CDI sepeda motor akan terhubung dan sepeda motor dapat dinyalakan. Sedangkan jika masukan pada kaki

basis transistor Q1 bertegangan nol, transistor Q1 akan menyumbat sehingga arus tidak akan mengalir melalui relay. Dimana saklar relay akan terbuka dan arus listrik pada CDI sepeda motor tidak akan terhubung, sehingga sepeda motor tidak akan bisa dinyalakan.



Gambar 3.5 Rangkaian Driver Relay

3.3 Diagram Alir Sistem



Gambar 3.6 Flowchart Sistem

BAB IV
PEMBUATAN DAN PENGUJIAN SISTEM

Untuk mengetahui apakah tujuan-tujuan dari pembuatan alat ini telah terlaksana atau tidak, perlu dilakukan pengujian dan analisa terhadap alat yang dibuat. Dan sebagai bagian yang tak terpisahkan adalah adanya proses evaluasi sehingga akan dapat dilakukan langkah-langkah positif guna membawa alat ini ke arah yang lebih baik.

Pengujian yang dilakukan terdiri dari dua tahap, tahapan-tahapan tersebut adalah sebagai berikut :

1. Pengujian unit sistem blok RFID sistem.
2. Pengujian modul RFID Reader, pengujian tersebut terdiri dari :
 - a. Pengujian jarak pembacaan RFID Reader terhadap tag RFID
 - b. Pengujian posisi tag RFID ketika pembacaan terjadi
 - c. Pengujian pembacaan multi tag
3. Pengujian perangkat lunak
4. Pengujian sistem secara keseluruhan.

4.1 Pengujian modul RFID

Dalam uji coba ini, modul akan diuji tanpa menghubungkan ke rangkaian minimum mikrokontroler, CDI sepeda motor dan RFID reader tidak difungsikan untuk membaca tag RFID.

Tabel 4.1 Pengujian Modul RFID Tanpa Difungsikan

Komponen	Status	Keterangan
Led Power Supply	Menyala	Sesuai rancangan
Led Clock	Berkedip	Sesuai rancangan
Buzzer	Off	Sesuai rancangan

Dari hasil pengujian modul RFID pada saat modul tidak difungsikan LED power supply akan menyala, sedangkan LED clock akan berkedip dan buzzer tidak berbunyi menandakan bahwa tidak ada proses pembacaan yg dilakukan oleh RFID Reader.

4.2 Pengujian Jarak Pembacaan RFID Reader Terhadap Tag RFID Dari Berbagai Arah

Dalam modul RFID ID-12 Innovations, kemampuan baca RFID Reader memiliki kemampuan jarak pembacaan maksimalnya mencapai 12 cm, oleh karenanya dalam uji coba ini akan diukur kemampuan maksimum jarak baca RFID Reader terhadap tag RFID. Dalam hal ini pengujian pembacaan dicoba dari berbagai arah.

4.2.1 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Atas

Pengujian dilakukan dengan posisi tag berada di atas RFID Reader. Uji coba ini bertujuan untuk mengetahui berapa jarak pendeteksian RFID Tag Card yang dapat dilakukan oleh RFID Reader. Pengujian dilakukan dengan mendekatkan RFID Tag Card ke RFID Reader dengan jarak tertentu dan kemudian diukur oleh mistar ukur. Apabila RFID Tag Card terdeteksi oleh RFID Reader maka buzzer pada rangkaian akan berbunyi.

Tabel 4.2 Jarak pengukuran RFID Reader Terhadap Tag RFID Dari Arah Atas

Jarak (cm)	Kartu 1	Kartu 2	Kartu 3	Kartu 4	Kartu 5
12	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
11	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
8	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
7	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
6	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
4	Terdeteksi	Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
3	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi
2	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi
1	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi

Dari tabel 4.2 maka dapat diambil kesimpulan bahwa semakin kecil jarak antara RFID Reader dengan tag, maka nilai kesalahan pembacaan akan semakin kecil seperti ditunjukkan oleh tabel diatas. Dan didapatkan besar jarak antara reader dan tag pada arah atas mempunyai akurasi pembacaan sampai 100% pada jarak antara 0-3 cm.

4.2.2 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Kanan

Pengukuran jarak baca dengan posisi tag dengan bagian tepi tag sebelah kanan reader.

Tabel 4.3 Jarak pengukuran RFID Reader Terhadap Tag RFID Dari Arah Kanan

Jarak (cm)	Kartu 1	Kartu 2	Kartu 3	Kartu 4	Kartu 5
12	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
11	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
8	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
7	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
6	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
3	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
2	Terdeteksi	Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
1	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi

Dari tabel 4.3 maka dapat diambil kesimpulan bahwa semakin kecil jarak antara RFID Reader dengan tag, maka nilai kesalahan pembacaan akan semakin kecil seperti

ditunjukkan oleh tabel diatas. Dan didapatkan besar jarak antara reader dan tag pada arah kanan RFID Reader mempunyai akurasi pembacaan sampai 100% pada jarak antara 0-1 cm.

4.2.3 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Kiri

Pengukuran hasil percobaan dilakukan dengan posisi tag berada pada bagian tepi sebelah kiri RFID Reader.

Tabel 4.4 Jarak pengukuran RFID Reader Terhadap Tag RFID Dari Arah Kiri

Jarak (cm)	Kartu 1	Kartu 2	Kartu 3	Kartu 4	Kartu 5
12	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
11	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
8	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
7	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
6	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
3	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
2	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
1	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi

Dari tabel 4.4 maka dapat diambil kesimpulan bahwa semakin kecil jarak antara RFID Reader dengan tag, maka nilai kesalahan pembacaan akan semakin kecil seperti ditunjukkan oleh tabel diatas. Dan didapatkan besar jarak antara reader dan tag dari

arah kiri RFID Reader mempunyai akurasi pembacaan sampai 100% pada jarak antara 0-1 cm.

Dari hasil percobaan yang telah dilakukan, maka didapat kesimpulan bahwa arah pembacaan Tag RFID yang paling maksimal adalah berada pada bagian atas RFID Reader. Hal ini dipengaruhi oleh pola radiasi dari antena RFID.

Dimana pembacaan Tag ID belum maksimal, karena pada data sheet RFID Reader harus mampu membaca Tag ID pada jarak 12 cm, namun pada pengujian alat RFID hanya mampu membaca Tag ID pada jarak 0 – 4 cm.

4.3 Pengujian Pembacaan Multi Tag

Pada pengujian ini, jarak baca yang dipakai adalah 2 cm pada arah atas. Pengujian ini dibatasi hanya pada pembacaan tiga tag secara bersamaan dimana posisi tagnya berada pada garis sejajar dan bukan ditindih.

Tabel 4.5 Hasil Pengujian Multi Tag

Percobaan	Single Tag	Double Tag	Triple Tag
1	Terbaca	Tidak Terbaca	Tidak Terbaca
2	Terbaca	Terbaca	Tidak Terbaca
3	Terbaca	Tidak Terbaca	Tidak Terbaca
4	Terbaca	Tidak Terbaca	Tidak Terbaca
5	Terbaca	Terbaca	Tidak Terbaca

Dari tabel 4.5 maka dapat diambil kesimpulan bahwa pembacaan tag hanya dapat dilakukan pada satu tag dalam satu waktu, apabila beberapa tag berada pada jarak baca RFID maka tag dengan jarak paling dekatlah yang akan diidentifikasi oleh reader.

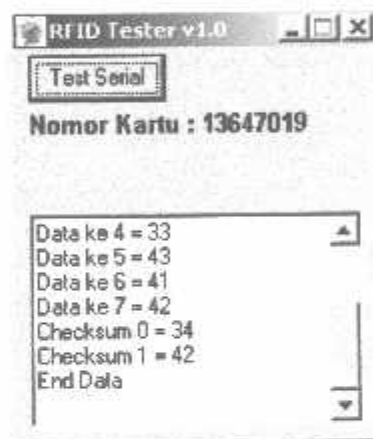
4.4 Pengujian Pembacaan Kode Tag RFID

Dalam pengujian kali ini, akan diuji pembacaan kode yang terdapat dalam tag RFID dengan menggunakan RFID Tester v1.0. Ada lima tag RFID yang diuji dimana dalam pengujian ini bertujuan untuk mengetahui apakah program dapat membaca dan membedakan tag.



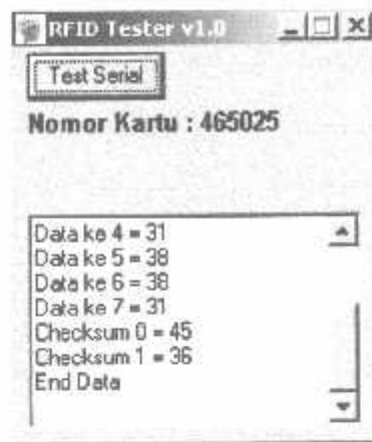
Gambar 4.1 Hasil Pengujian Tag Pertama

Dari pengujian tag yg pertama ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.



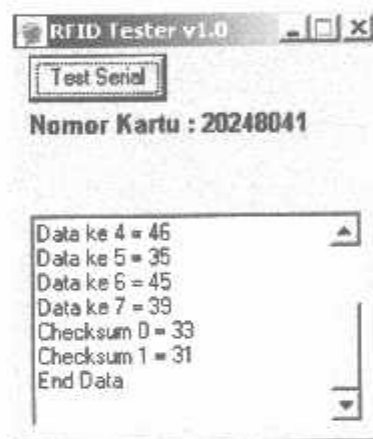
Gambar 4.2 Hasil Pengujian Tag Kedua

Dari pengujian tag yg kedua ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.



Gambar 4.3 Hasil Pengujian Tag Ketiga

Dari pengujian tag yg ketiga ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.



Gambar 4.4 Hasil Pengujian Tag Keempat

Dari pengujian tag yg keempat ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.



Gambar 4.5 Hasil Pengujian Tag Kelima

Dari pengujian tag yg kelima ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.

Sehingga dari percobaan diatas dapat disimpulkan bahwa program yang dibuat dapat berjalan dengan lancar dalam sistem pembacaan dan tanpa mengalami error sama sekali.

4.5 Pengujian Sistem Secara Keseluruhan

Pada pengujian ini dilakukan dengan dua tahapan, dimana tahapan terdiri dari :

1. Pengujian keseluruhan dengan menggunakan Tag RFID yang terdata didalam program, terdiri dari satu tag RFID.
2. Pengujian keseluruhan dengan menggunakan Tag RFID yang tidak terdata didalam program, terdiri dari empat Tag RFID tidak terdata.
- 3.

Tabel 4.6 Hasil Tag Terdata dan Tidak Terdata

Percobaan	Jenis tag	Start
1	Terdata	Ya
2	Tidak Terdata	Tidak
3	Tidak Terdata	Tidak
4	Tidak Terdata	Tidak
5	Tidak Terdata	Tidak

Pengujian ini dilakukan setelah pembuatan hardware dan software selesai dan program sudah didownload keadalam mikrokontroler. Tujuan pengujian ini untuk

mengetahui apakah alat yang dirancang sudah berjalan sesuai dengan yang diinginkan atau belum. Adapun yang akan diuji adalah pembacaan Tag ID dengan RFID Reader. Pengujian ini dilakukan untuk mengecek apakah perintah yang diinputkan sudah sesuai dengan output.

Pada percobaan pertama yaitu dengan menggunakan tag RFID yg tidak terdata diprogram. Maka didapatkan bahwa sepeda motor tidak dapat dihidupkan walaupun kunci kuntak sudah berada pada posisi on.

Pada percobaan yang kedua yaitu dengan menggunakan tag RFID yang sudah terdata dalam program. Maka didapat kesimpulan bahwa sepeda motor dapat dinyalakan karena data yang ada terbaca oleh RFID reader sesuai dengan yang terdata dalam program.

BAB V

PENUTUP

Dari perancangan dan pengujian alat, dapat ditarik kesimpulan dan saran mengenai skripsi sebagai berikut :

5.1 Kesimpulan

1. Dengan menggunakan program BASCOM-AVR penulis dapat membuat program, sistem kendali lebih mudah karena program ini cukup lengkap dengan adanya simulator untuk LED dan LCD sehingga penulis bisa melihat program yang dibuat didalam simulasi dan sebelum didownloadkan kedalam mikrokontroler.
2. Penggunaan alat dilakukan dengan cara menghubungkan perangkat lunak (Software) dan perangkat keras (Hardware) sehingga dapat dilihat kemampuan program dalam mengendalikan mikrokontroler.
3. Pembacaan Tag ID belum maksimal, karena pada data sheet RFID Reader harus mampu membaca Tag ID pada jarak 12 cm, namun pada pengujian alat RFID hanya mampu membaca Tag ID pada jarak 0 – 4 cm.
4. Bahwa arah pembacaan Tag RFID yang paling optimal adalah pada bagian atas RFID. Hal ini dipengaruhi oleh pola radiasi dari antenna RFID.
5. Pembacaan tag hanya dapat dilakukan pada satu tag dalam satu waktu, apabila beberapa tag dalam range baca maka tag dengan jarak paling dekatlah yang akan diidentifikasi oleh reader.

5.2 Saran

1. Sebelum mendownload program kedalam mikrokontroler, sebaiknya compile terlebih dahulu program untuk memastikan program sudah berjalan dengan benar.
2. Diharapkan kepada pembaca dan semua orang yang berminat dengan keamanan sepeada motor agar dapat mengembangkan program untuk dapat lebih mengoptimalkan kerja dari sistem kendali.

Penulis sangat mengharapkan sumbangan ilmu berupa kritik dan saran dari pembaca, karena penulis sadar keterbatasan kemampuan sehingga diharapkan masukan agar dapat lebih berkembang lagi dan mengharapkan kedepan ada yang

mengembangkan alat ini jauh lebih baik. Dan semoga karya ini dapat bermanfaat bagi kita semua.

DAFTAR PUSTAKA

- [1]. *AVR AT Teknik Antar muka dan Pemograman Mikrokontroller AT MEGA 16* yogyakarta: Penerbit ANDI
- [2]. *Belajar Sendiri Mikrokontroller AVR Seri AT Mega 16* yogyakarta: penerbit ANDI
- [3]. Alfian, Moh. Yusuf : *Pengembangan Modul Praktikum, Pengaturan Otomatis :* Proyek Akhir D4 PENS-ITS, 2007
- [4]. Atmel Corp. Application Note. Datasheet of Atmega 16 datasheet
- [5]. Innovative Electronics. Manual RFID Starter Kit, 2007
- [6]. Innovative Electronics. ID Series Datasheet, 10 Februari 2004
- [7]. Innovative Electronics. ID Series Datasheet, 1 Maret 2005

D
A
F
7
A
R

L
A
M
P
9
R
A
N

Listing Program

\$regfile = "m16def.dat"

\$crystal = 8000000

\$baud = 9600

Relay Alias Portd.7

Config Relay = Output

Reset Relay

Dim S As String * 16

Dim Temps As String * 2

Dim Datl(8) As Long

Dim DalsAs String * 10

Dim A As Byte

Dim SttAs Byte

Dim I As Byte

Dim P As Byte

Dim Buff(16) As Byte

Dim D As Byte

Dim Al As Long

Dim Templ(8) As Long

Dim SstrAs Byte

ConstAddre = 1

Datl(8) = 268435456

Datl(7) = 16777216

Datl(6) = 1048576

Datl(5) = 65536

Datl(4) = 4096

Datl(3) = 256

Datl(2) = 16

Datl(1) = 1

```

ReadeepromSttr ,Addre
If Sttr = 1 Then
    Set Relay
Else
    Reset Relay
End If

Do
    Input S

Dats = "" : A = 0 : Stt = 0
For I = 1 To Len(s)
    Temps = Mid(s , I , 1)
    P = Asc(temps)
    If Stt = 1 Then
        Incr A
        If A > 2 And A < 11 Then
            Dats = Dats + Temps
        End If
    End If
    If P = 2 Then Stt = 1
    Buff(i) = Asc(temps)
Next

A = 0
For I = Len(dats) To 1 Step - 1
    Temps = Mid(dats , I , 1)
    If Temps = "A" Then Temps = "10"
    If Temps = "B" Then Temps = "11"
    If Temps = "C" Then Temps = "12"
    If Temps = "D" Then Temps = "13"
    If Temps = "E" Then Temps = "14"
    If Temps = "F" Then Temps = "15"
Incr A

```

```
D = Val(temps)
Templ(i) = Dat1(a) * D
Next
A1 = 0
For I = 1 To 8
    A1 = A1 + Templ(i)
Next I
Wait 2
Print "Nomerkartu = "; A1
If A1 = 268575304 Then
    Relay = Not Relay
    If Sstr = 1 Then
        Sstr = 0
    Else
        Sstr = 1
    End If
    WriteepromSstr ,Addre
End If

Loop
End                                     'end program
```



PERMOHONAN PERSETUJUAN SKRIPSI

Yang Beranda Tangan Dibawah Ini:

Nama : DAULAT KRISTON NAINGGOLAN
 NIM : 0217057
 Semester :
 Fakultas : (Teknologi Industri)
 Jurusan : (Teknik Elektro S-1)
 Konsentrasi : **TEKNIK ENERGI LISTRIK**
TEKNIK ELEKTRONIKA
TEKNIK KOMPUTER DAN INFORMATIKA
TEKNIK KOMPUTER
TEKNIK TELEKOMUNIKASI

Alamat :

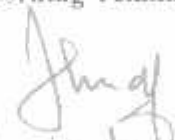
Dengan ini kami mengajukan permohonan untuk mendapatkan persetujuan untuk membuat SKRIPSI Tingkat Sarjana. Untuk melengkapi permohonan tersenut, bersama ini kami lampirkan persyaratan-persyaratan yang harus dipenuhi.

Adapun persyaratan- persyaratan pengambilan SKRIPSI adalah sebagai berikut:

- | | |
|--|---------|
| 1. Telah melaksanakan semua praktikum sesuai dengan konsentrasinya | (.....) |
| 2. Telah lulus dan menyerahkan laporan Praktek Kerja | (.....) |
| 3. Telah lulus seluruh mata kuliah keahlian (MKB)sesuai konsentrasinya | (.....) |
| 4. Telah menempuh matakuliah > 134 sks dengan IPK > 2 dan tidak ada nilai E | (.....) |
| 5. Telah mengikuti secara aktif kegiatan seminar Skripsi yang diadakan Jurusan | (.....) |
| 6. Memenuhi persyaratan administrasi | (.....) |

Demikian permohonan ini untuk mendapatkan penyelesaian lebih lanjut dan atas perhatiannya kami ucapkan terima kasih.

Telah diteliti kebenarannya data tersebut diatas
 Recording Teknik Elektro S-1


 (Purni Wardayanti)

Disetujui
 Ketua Prodi Teknik Elektro S-1


 M. Ibrahim Ashari, ST, MT
 NIP. P. 1030100358

Malang, 30 - 06 - 2013

Permohon


 (D. KRISTON NAINGGOLAN)

Mengetahui
 Dosen Wali


 (.....)

Catatan:

Bagi mahasiswa yang telah memenuhi persyaratan mengambil SKRIPSI agar membuat proposal dan mendapat persetujuan dari Ketua Prodi T. elektro S-1

IP 381 / = 2.76
 138



ampiran : 1 (satu) berkas
Pembimbing Skripsi

epada : Yth. Bapak/Ibu **Irmalia Suryani Faradisa, ST, MT**
Dosen Teknik Elektro S-1
ITN Malang

Yang bertanda tangan dibawah

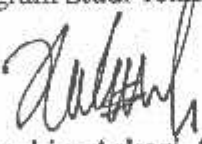
Nama : **DAULAT KRISTON NAINGGOLAN**
Nim : **0217057**
Jurusan : **Teknik Elektro S-1**
Konsentrasi : **Teknik Elektronika**

Dengan ini mengajukan permohonan, kiranya Bapak/Tbu bersedia menjadi Dosen Pembimbing untuk penyusunan Skripsi dengan judul :


"PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR MENGGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16"

Demikian permohonan kami buat dan atas kesediaan Bapak kami ucapkan terima kasih.

Mengetahui
Ketua Program Studi Teknik Elektro S-1


M. Ibrahim Ashari, ST, MT
NIP.P. 1030100358

Hormat Kami


DAULAT KRISTON NAINGGOLAN
NIM. 0217057



ampiran : 1 (satu) berkas
Pembimbing Skripsi

epada : Yth. Bapak/Ibu **Yuli Wahyuni, ST, MT**
Dosen Teknik Elektro S-1
ITN Malang

Yang bertanda tangan dibawah


Nama : **DAULAT KRISTON NAINGGOLAN**
Nim : **0217057**
Jurusan : **Teknik Elektro S-1**
Konsentrasi : **Teknik Elektronika**

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing untuk penyusunan Skripsi dengan judul :


"PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR MENGGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16"

Demikian permohonan kami buat dan atas kesediaan Bapak kami ucapkan terima kasih.

Mengetahui
Ketua Program Studi Teknik Elektro S-1


M. Ibrahim Ashabi, ST, MT
NIP.P. 1030100358

Hormat Kami


DAULAT KRISTON NAINGGOLAN
NIM. 0217057



PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : DAULAT KRISTON NAINGGOLAN
Nim : 0217057
Semester : XIV (Empatbelas)
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika

Dengan ini menyatakan bersedia/tidak bersedia*) Membimbing skripsi dari mahasiswa tersebut, dengan judul :

**" PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR
MENGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16"**

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Hormat Kami

Yuli Wahyuni, ST, MT

NIP.P. 1031200456




Catatan :

Setelah disetujui agar formulir ini Diserahkan mahasiswa/i yang bersangkutan kepada jurusan untuk diproses lebih lanjut

*) Coret yang tidak perlu



**BERITA ACARA SEMINAR PROPOSAL SKRIPSI
PROGRAM STUDI TEKNIK ELEKTRO S-1
Konsentrasi : Teknik Elektronika**

1.	Nim	0217057		
2.	Nama	DAULAT KRISTON NAINGGOLAN		
3.	Konsentrasi Jurusan	Teknik Elektronika		
4.	Jadwal Pelaksanaan	Waktu	Tempat	
	20 Juni 2013	09.00	III.1.5	
5.	Judul proposal yang diseminarkan Mahasiswa	PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR MENGGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16		
6.	Perubahan judul yang diusulkan oleh Kelompok Dosen Keahlian			
7.	Catatan :			
8.	Catatan :			
	Persetujuan judul Skripsi			
	Disetujui, Dosen Keahlian I	Disetujui, Dosen Keahlian II	Disetujui, Dosen Keahlian III	
	(.....)	(.....)	(.....)	
	Mengetahui, Ketua Program Studi Teknik Elektro S-1  M. Ibrahim Ashari, ST, MT NIP. P 1030100358	Disetujui, Calon Dosen Pembimbing ybs		
		Pembimbing I	Pembimbing II	
		 (.....)	 (.....)	



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunfling), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa : Daulat Kriston Nainggolan
NIM : 02.17.057
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : PERANCANGAN DAN PEMBUATAN ALAT
PENGAMANAN SEPEDA MOTOR MENGGUNAKAN
RFID BERBASIS MIKROKONTROLER ATMEGA 16

Dipertahankan dihadapan tim penguji skripsi jenjang Strata Satu (S-1) pada:

Hari : Selasa
Tanggal : 20 Agustus 2013
Dengan Nilai : 77 (B+) *rs*

PANITIA UJIAN SKRIPSI

Ketua Prodi Teknik Elektro S-1

Ibrahim Ashari, ST, MT
NIP. P. 1030100358

Sekretaris Jurusan Teknik Elektro S-1

Dr. Eng Aryuanto, ST, MT
NIP. Y. 1030800417

DOSEN PENGUJI

Dosen Penguji I

Bambang Prio Hartono, ST, MT
NIP. Y. 102840082

Dosen Penguji II

Ibrahim Ashari, ST, MT
NIP. P. 1030100358



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
 FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
 PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
 BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
 Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Formulir Perbaikan Skripsi

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : DAULAT KRISTON NAINGGOLAN
 NIM : 02.17.057
 Jurusan : Teknik Elektro S-1
 Konsentrasi : Teknik Elektronika
 Judul Skripsi : PERANCANGAN DAN PEMBUATAN ALAT
 PENGAMANAN SEPEDA MOTOR MENGGUNAKAN
 RFID BERBASIS MIKROKONTROLER ATMEGA 16

No.	Dosen Penguji	Tanggal	Uraian	Paraf
1	Dosen Penguji I		1. Abstrak 2. Photo Alat	
2	Dosen Penguji II			

Disetujui oleh :

Dosen Penguji I

Bambang Prio Hartono, ST, MT
 NIP. Y. 102840082

Dosen Penguji II

Ibrahim Ashari, ST, MT
 NIP. P. 1030100358

Mengetahui :

Dosen Pembimbing I

Irmalia Suryani Faradisa, ST, MT
 NIP. P. 1020000265

Dosen Pembimbing II

Yuli Wahyuni, ST, MT
 NIP. P. 1031200456

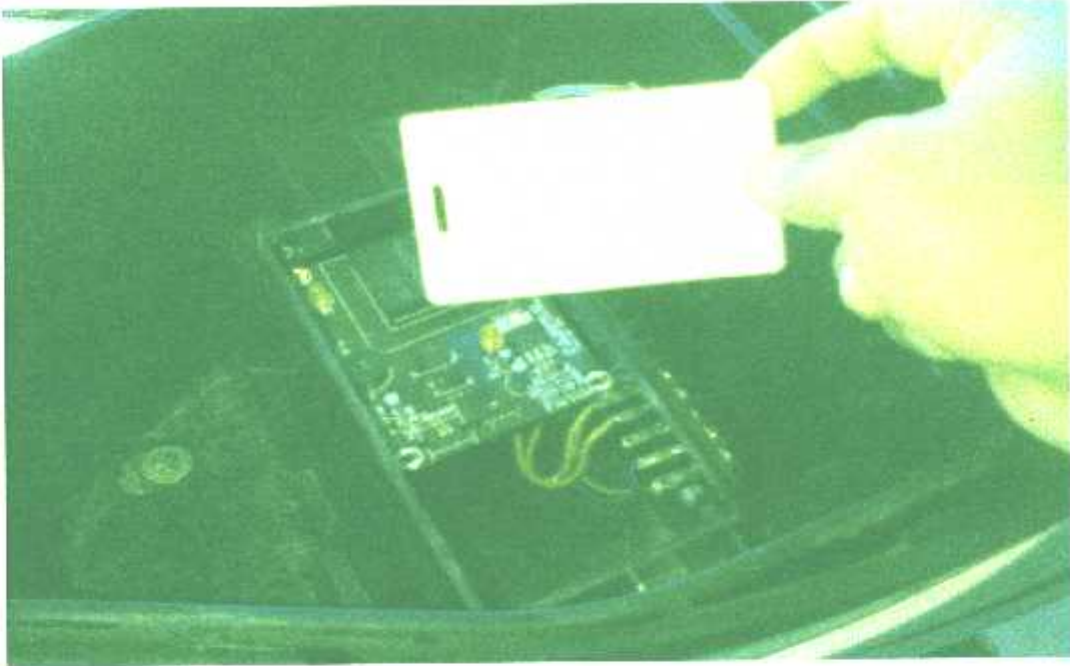


FOTO SAAT PENGUJIAN ALAT II



GAMBAR KESELURUHAN

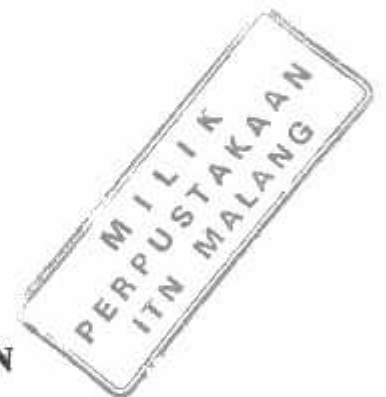
**PERANCANGAN DAN PEMBUATAN ALAT PENGAMANAN
SEPEDA MOTOR MENGGUNAKAN RFID
BERBASIS MIKROKONTROLER ATMEGA 16**

SKRIPSI



Disusun Oleh :

**DAULAT KRISTON NAINGGOLAN
NIM: 02.17.057**



**PROGRAM STUDI TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2013**

LEMBAR PERSETUJUAN

**PERANCANGAN DAN PEMBUATAN ALAT PENGAMANAN
SEPEDA MOTOR MENGGUNAKAN RFID
BERBASIS MIKROKONTROLER ATMEGA 16**

SKRIPSI

Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik

Disusun Oleh :

DAULAT KRISTON NAINGGOLAN

NIM : 02.17.057

Mengetahui,

Ketua Jurusan Teknik Elektro S-1



M. Ibrahim Ashari, ST, MT
NIP.P.1030100358

Diperiksa dan Disetujui

Dosen Pembimbing I

Dosen Pembimbing II

Irmalia Suryani Faradisa, ST, MT
NIP.P. 1030000365

Yuli Wahyuni, ST, MT
NIP.P. 1031200456

**PROGRAM STUDI TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2013**

SURAT PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini :

Nama : **Daulat Kriston Nainggolan**
NIM : **02.17.057**
Program Studi : **Teknik Elektro S-1**
Konsentrasi : **Teknik Elektronika S-1**

Dengan ini menyatakan bahwa Skripsi yang saya buat adalah hasil karya sendiri, tidak merupakan plagiasi dari karya orang lain. Dalam Skripsi ini tidak memuat karya orang lain, kecuali dicantumkan sumbernya sesuai dengan ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat, dan apabila di kemudian hari ada pelanggaran atas surat pernyataan ini, saya bersedia menerima sanksinya.

Malang, 11 September 2013

Yang membuat Pernyataan,



Daulat Kriston Nainggolan

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Tuhan Yang Maha Mulia atas berkah dan rahmat-Nya penulis dapat menyelesaikan tugas skripsi yang berjudul :

“PERANCANGAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR MENGGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16”

Penulisan skripsi ini disusun untuk memenuhi syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro S-1 Konsentrasi Teknik Elektronika di Institut Teknologi Nasional Malang. Terima kasih yang mendalam penulis haturkan kepada kedua orang tua penulis “P. Nainggolan dan T. Br. Silaban” yang telah berjasa membentuk kepribadian penulis, khususnya memberikan inspirasi yang kuat serta dukungan dan doa yang tiada henti. Terimakasih juga penulis sampaikan kepada :

1. Bapak Ir. Soeparno Djiwo, MT selaku rektor ITN Malang.
 2. Bapak Ir. Anang Subardi, MT selaku Dekan Fakultas Teknologi Industri ITN Malang.
 3. Bapak M. Ibrahim Ashari, ST, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
 4. Bapak Dr. Eng. Aryunto Soetedjo, ST, MT selaku Sekretaris Jurusan Teknik Elektro S-1 ITN Malang.
 5. Ibu Irmalia Suryani Faradisa, ST, MT selaku Dosen Pembimbing I.
 6. Ibu Yuli Wahyuni, ST, MT selaku Dosen Pembimbing II.
 7. Ibu Puji Handayani selaku Recording Jurusan Teknik Elektro ITN Malang.
-

**PERANCANGAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA
MOTOR MENGGUNAKAN RFID
BERBASIS MIKROKONTROLER ATMEGA 16**

Daulat Kriston Nainggolan, NIM 02.17.057
pakkatnabalau@yahoo.co.id

**Dosen Pembimbing : Irmalia Suryani Faradisa, ST, MT dan Yuli Wahyuni, ST,
MT**

ABSTRAK

Dewasa ini banyak orang yang meninggalkan sepeda motor di areal parkir atau dimana saja. Tetapi mereka tidak mengetahui bahwa sepeda motornya menjadi sasaran para pencuri. Sering pemiliknya terlambat mengetahui bahwa sepeda motornya telah hilang karena alat pengaman yang ada pada umumnya mereka pasang dapat dengan mudah dihobol pencuri. Dengan kemajuan teknologi seperti sekarang penulis mencoba membuat alat yang berfungsi untuk mengamankan sepeda motor.

Pemulis mencoba membuat modul keamanan sepeda motor menggunakan RFID yang berbasis mikrokontroler AVR Atmega 16, yang berfungsi sebagai unit pengontrol alat yang dibuat. Sistem ini berfungsi sebagai pengganti kunci konvensional. Sistem keamanan sepeda motor ini dapat memberikan keamanan karena alat ini berfungsi jika RFID Tag yang dimasukkan dalam keadaan benar, maka motor tidak bisa menyala kalau data RFID Tagnya berbeda.

Sistem ini terdiri atas perangkat lunak dan perangkat keras. perangkat keras terdiri dari RFID Reader dan RFID Tag sebagai inputan perangkat lunak terdiri dari bahasa basic dengan BASCOM-AVR IDE yang diisikan dalam mikrokontroler sebagai pemroses data.

Sistem ini tidak dapat bekerja jika secara bersamaan ada lebih dari satu RFID Tag dideteksi sekaligus oleh RFID Reader. Dimana tujuan pembuatan system ini untuk memberikan alternatif pengamanan.

Kata kunci: Mikrokontroler, RFID, Pengamanan Sepeda Motor

8. Keluarga besar Op. Betram Nainggolan : Kel. Risma Nainggolan, Kel. Ana Nainggolan, Kel. Bidicoff Nainggolan, Kel. Ernita Nainggolan dan John Nainggolan.
9. Keponakan-keponakanku : Gabe, Bagas, Luisa, Enzo, Bertram, Benheart dan Rafael.
10. Sahabat-sahabat penulis yang ada di Malang, akka halak hita.
11. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa skripsi ini masih jauh dari sebuah kesempurnaan, untuk itu kritik dan saran yang membangun dari pembaca sangat penulis harapkan untuk perbaikan penelitian ini.

Akhir kata penulis mohon maaf kepada semua pihak jika dalam proses pembuatan skripsi ini penulis melakukan kesalahan baik yang disengaja maupun tidak disengaja.

Semoga Allah Bapa selalu memberkati dan memberi jalan yang terbaik bagi kita semua. Amin.

Malang, September 2013

Penulis
(Daulat Kriston Nainggolan)

DAFTAR ISI

Lembar Persetujuan	i
Pernyataan Originalitas.....	ii
Abstrak	iii
Kata Pengantar	iv
Daftar Isi	vi
Daftar Tabel	viii
Daftar Gambar.....	ix
BAB I Pendahuluan	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metode Penelitian	2
1.6 Sistematika Penulisan	3
BAB II Landasan Teori	4
2.1. Radio Frequency Identification (RFID)	4
2.2. RFID Reader	6
2.3. RFID Tag (Transponder)	6
2.4. Mikrokontroler ATMEGA 16	8
2.4.1 Arsitektur ATMEGA 16	9
2.4.2 Konfigurasi ATMEGA 16	10
2.5 Relay	11
BAB III Perancangan Sistem	14
3.1 Blok Diagram Rangkaian	14
3.2 Perancangan Sistem Perangkat Keras	15
3.2.1 Tag RFID	15
3.2.2 Perancangan Rangkaian Modul RFID Reader	15
3.2.3 Perancangan Rangkaian Mikrokontroler	17
3.2.4 Perancangan Driver Relay	17
3.3 Diagram Alir Sistem	19
BAB IV Pembuatan Dan Pengujian Sistem	20
4.1 Pengujian Modul RFID	20

4.2 Pengujian Jarak Pembacaan RFID Reader Terhadap Tag RFID Dari Berbagai Arah	21
4.2.1 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Atas	21
4.2.2 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Kanan	22
4.2.3 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Kiri	23
4.3 Pengujian Pembacaan Multi Tag	24
4.4 Pengujian Pembacaan Kode Tag RFID	24
4.5 Pengujian Sistem Secara Keseluruhan	27
BAB V Penutup	29
5.1 Kesimpulan	29
5.2 Saran	29
Daftar Pustaka	30
Daftar Lampiran	31

DAFTAR TABEL

4.1. Pengujian Modul RFID Tanpa Difungsikan	21
4.2. Jarak Pengukuran RFID Reader Terhadap Tag RFID Pada Arah Atas	22
4.3. Jarak Pengukuran RFID Reader Terhadap Tag RFID Pada Arah Kanan	23
4.4. Jarak Pengukuran RFID Reader Terhadap Tag RFID Pada Arah Kiri	24
4.5. Hasil Pengujian Multi Tag	25
4.6. Hasil Tag Terdata dan Tidak Terdata	28

DAFTAR GAMBAR

2.1. Data Transfer RFID Reader dan Tag RFID	6
2.2. Penggolongan RFID Tag	8
2.3. Konfigurasi ATMEGA 16	11
2.4. Rangkaian Darlington	13
3.1. Blok Diagram	15
3.2. Rangkaian Modul RFID Reader	16
3.3. Konfigurasi Pin RFID Reader	17
3.4. Rangkaian Minimum Sistem Mikrokontroler	18
3.5. Rangkaian Relay	19
3.6. Flowchart Sistem	20
4.1. Hasil Pengujian Tag Pertama	26
4.2. Hasil Pengujian Tag Kedua	26
4.3. Hasil Pengujian Tag Ketiga	27
4.4. Hasil Pengujian Tag Keempat	27
4.5. Hasil Pengujian Tag Kelima	28

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Teknologi memegang peran penting di era modernisasi seperti pada saat ini, dimana teknologi telah menjadi bagian yang tidak dapat dipisahkan dalam kehidupan sehari – hari. Sehingga mendorong manusia untuk menciptakan sebuah alat yang lebih aman sehingga membantu mempermudah dan mempercepat manusia dalam menyelesaikan pekerjaan dengan hasil yang sebaik-baiknya. Dimana kita bisa mengambil contoh permasalahan yang terdapat pada pengamanan sepeda motor.

Dewasa ini sepeda motor telah menjadi alat transportasi yang paling banyak digunakan di jalanan di Indonesia, dikarenakan harga sepeda motor terjangkau bagi semua kalangan masyarakat. Umumnya untuk pengamanan pabrik hanya memberikan kunci konvensional. Pencurian sepeda motor sering kita dengar baik secara langsung maupun lewat media massa. Dalam hal pengamanan tambahan sepeda motor sudah sering kita temukan penggunaan alat tambahan seperti penggunaan remote kontrol, maupun dengan menggunakan jasa pengamanan seperti satpam atau petugas parkir.

Dalam hal ini penulis mencoba memberikan opsi lain dalam pengamanan sepeda motor dengan menggunakan RFID (Radio Frequency Identification). RFID sering kita temukan pemakaiannya seperti kartu ATM, KTP Elektronik, kunci kamar hotel, dll. Aplikasi ini menggunakan mikrokontroler sebagai pengontrol utama.

Alat ini menggunakan RFID sebagai pengamanan tambahan, karena kunci konvensional sepeda motor masih digunakan. RFID reader digunakan sebagai pembaca, sedangkan Tag RFID digunakan sebagai kartu identitas atau pengenalan ketika hendak menyalakan sepeda motor. Tag RFID berisi kode-kode yang unik, dimana setiap tag berisi kode yg berbeda-beda. Proses pengamanan sepeda motor ini dimulai dengan menyalakan kunci konvensional sepeda motor ke posisi on, kemudian tag RFID didekatkan ke reader. Reader akan membaca data yang ada pada tag, apabila data pada tag tidak terdata maka motor tidak akan bisa dinyalakan. Sedangkan ketika reader membaca tag dan datanya sesuai, mikrokontroler akan memberikan perintah kepada relay untuk menghubungkan arus pada CDI.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, telah diambil permasalahan yang akan dibahas dalam perancangan dan pembuatan alat ini adalah:

1. Bagaimana membuat dan menempatkan RFID Reader pada sepeda motor.
2. Bagaimana menghubungkan RFID Reader dengan mikrokontroler.
3. Bagaimana membuat instalasi hardware dari mikrokontroler.
4. Bagaimana membuat software pada mikrokontroler ATMEGA 16.

1.3 Tujuan

Tujuan dari penulisan skripsi ini adalah:

1. Untuk merancang dan membuat alat pengamanan sepeda motor menggunakan RFID berbasis mikrokontroler ATMEGA 16.
2. Mempelajari prinsip kerja mikrokontroler ATMEGA 16.
3. Mengaplikasikan teori yang telah didapat dibangku kuliah untuk membuat suatu alat yang sesuai teori yang telah didapat.

1.4 Batasan Masalah

Agar permasalahan mengarah sesuai dengan tujuan yang diharapkan, maka pembahasan dibatasi oleh hal-hal sebagai berikut :

1. Tidak membahas tentang struktur dan jaringan instalasi listrik sepeda motor.
2. Alat akan dikontrol menggunakan mikrokontroler AT MEGA 16.
3. Hanya membahas perancangan dan pembuatan model pengamanan sepeda motor menggunakan RFID dengan mikrokontroler AT MEGA 16.

1.5 Metode Penelitian

Adapun metode penelitian yang digunakan adalah sebagai berikut:

1. Studi Literatur

Pengumpulan data yang dilakukan dengan mencari bahan-bahan kepustakaan dan referensi dari berbagai sumber sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan objek penelitian.

2. Analisa Kebutuhan Sistem

Data dan informasi yang telah diperoleh akan dianalisa agar didapatkan

kerangka global yang bertujuan untuk mendefinisikan kebutuhan sistem di mana nantinya akan digunakan sebagai acuan perancangan sistem.

3. Perancangan dan Implementasi

Berdasarkan data dan informasi yang telah diperoleh serta analisa kebutuhan untuk membangun sistem ini, akan dibuat rancangan kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat dan diimplementasikan kedalam sistem.

4. Eksperimen dan Evaluasi

Pada tahap ini, sistem yang telah selesai dibuat akan diuji coba, yaitu pengujian berdasarkan fungsionalitas program, dan akan dilakukan koreksi dan penyempurnaan program jika diperlukan.

1.6 Sistematika Penulisan

Untuk mempermudah dan memahami pembahasan penulisan skripsi ini, maka sistematika penulisan disusun sebagai berikut :

Bab I : Pendahuluan

Berisi Latar Belakang, Rumusan Masalah, Tujuan Penelitian, Pembatasan Permasalahan, Metode Penelitian dan Sistematika Penulisan.

Bab II : Tinjauan Pustaka

Berisi tentang landasan teori mengenai permasalahan yang berhubungan dengan penelitian yang dilakukan.

Bab III : Perancangan Sistem

Dalam bab ini berisi mengenai analisa kebutuhan sistem baik software maupun hardware yang diperlukan untuk membuat kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat.

Bab IV : Pembuatan dan Pengujian Sistem

Berisi tentang implementasi dari perancangan sistem yang telah dibuat serta pengujian terhadap sistem tersebut.

Bab V : Penutup

Merupakan bab terakhir yang memuat intisari dari hasil pembahasan yang berisikan kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk pengembangan penulisan selanjutnya.

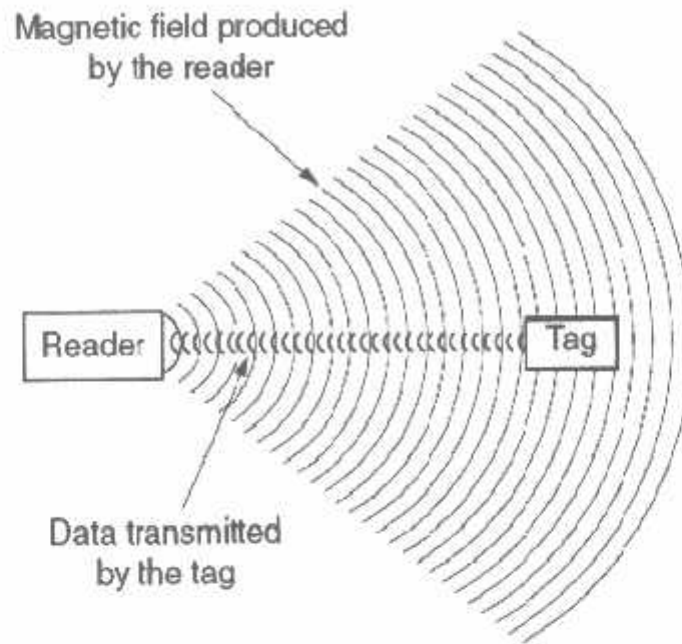
BAB II LANDASAN TEORI

2.1 Radio Frequency Identification (RFID)

Radio Frequency Identification (RFID) adalah proses identifikasi seseorang atau objek dengan menggunakan frekuensi transmisi radio. Radio frequency identification (RFID) menggunakan frekuensi radio untuk membaca informasi dari sebuah devais kecil yang disebut tag atau transponder (Transmitter + Responder). Tag RFID akan mengenali diri sendiri ketika mendeteksi sinyal dari devais yang kompatibel, yaitu pembaca RFID (RFID Reader).

RFID merupakan teknologi yang berfungsi untuk melakukan deteksi dan identifikasi terhadap suatu obyek melalui data yang ditransmisikan melalui frekuensi radio. Sistem tersebut minimal memerlukan sebuah tag (yang berfungsi sebagai transponder), sebuah reader (yang berfungsi sebagai interrogator) dan sebuah antena (yang berfungsi sebagai coupling device). Reader biasanya terhubung dengan sebuah host computer atau perangkat lainnya yang memiliki kecerdasan untuk memproses lebih lanjut tag data dan memutuskan untuk mengambil suatu tindakan. Salah satu elemen penting pada RFID adalah data transfer. Data transfer terjadi ketika terjadi hubungan antara sebuah tag dengan sebuah reader, yang dikenal dengan coupling, melalui antena baik yang terpasang pada tag tersebut maupun pada.

Coupling pada kebanyakan sistem-sistem RFID menggunakan metode magnetic (inductive) atau electromagnetic (backscatter). Metode yang digunakan tersebut bergantung pada harga, ukuran, kecepatan, dan jangkauan pembacaan serta keakuratan. Pada umumnya komunikasi antara sebuah tag dengan sebuah reader terjadi melalui sebuah physical principle yang dikenal sebagai sebuah backscatter modulation. Pada proses ini, sebuah reader mengirimkan sinyal kepada sebuah tag, dan tag akan menanggapi dengan memantulkan sebagian dari energi ini kembali ke reader. Hal ini dapat diilustrasikan dari gambar 2.1.



Gambar 2.1 Data Transfer RFID Reader dan Tag RFID

Dalam memilih RFID tersebut, terdapat beberapa hal yang harus diperhatikan, yaitu:

1. Jenis tag yang digunakan.
2. Jenis reader yang dipakai.
3. Frekuensi operasional dari sistem.
4. Jarak antara reader dan tag yg diinginkan.

RFID terbagi dari dua komponen, yaitu :

1. RFID Tag
Alat yang menyimpan informasi untuk identifikasi objek.
2. RFID Reader
Alat yang kompatibel dengan RFID Tag yang berkomunikasi secara wireless dengan RFID Tag.

2.2 RFID Reader

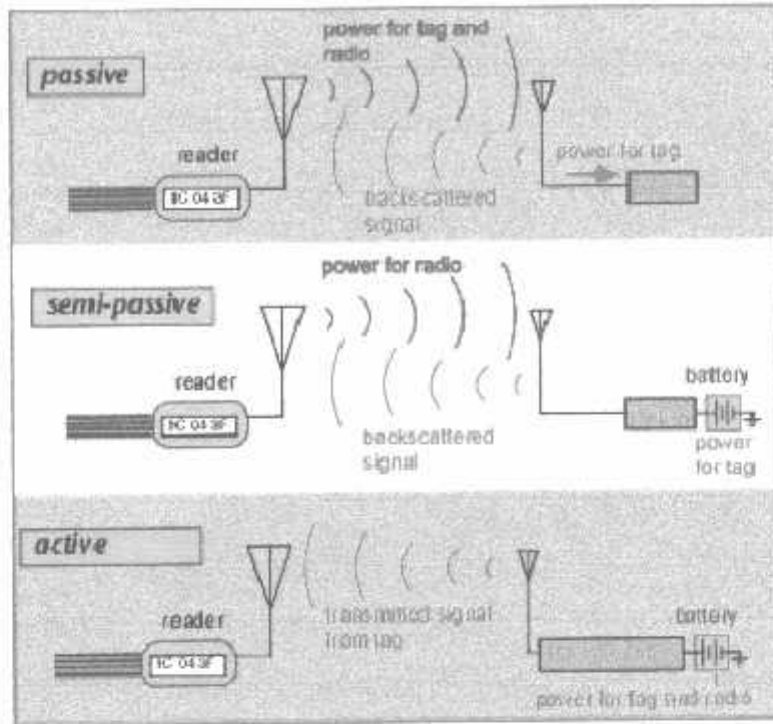
RFID reader memiliki high frequency module (transmitter dan receiver), control module dan juga coupling element (coil dan microwave antenna), yang berfungsi membangkitkan sinyal untuk mengaktifkan RFID tags, sehingga dapat melakukan pengiriman dan penerimaan data. Ketika sebuah RFID tags melewati medan elektromagnetik RFID reader, maka RFID tags tersebut akan mendeteksi sinyal pengaktifan dari reader, dan mengirimkan sinyal balik untuk pemrosesan data yang telah tersimpan dalam memori tag sebagai respon. RFID reader kemudian menterjemahkan data yang dikirimkan oleh RFID tags tersebut sesuai dengan kebutuhan. Proses pembacaan kode-kode data yang terdapat pada RFID tags dilakukan menggunakan gelombang radio, sehingga proses pengidentifikasian menjadi lebih mudah.

2.3 RFID Tag (Transponder)

Tag ID adalah devais yang dibuat dari rangkaian elektronika dan antena yang terintegrasi di dalam rangkaian tersebut. Rangkaian elektronik dari tag RFID umumnya memiliki memori sehingga tag ini mempunyai kemampuan untuk menyimpan data. Memori pada tag dibagi menjadi sel-sel. Beberapa sel menyimpan data Read Only, misalnya serial number yang unik yang disimpan pada saat tag tersebut diproduksi. Sel lain pada RFID mungkin juga dapat ditulis dan dibaca secara berulang.

Tag ini bekerja saat antena mendapatkan sinyal dari reader dan sinyal tersebut akan dipantulkan lagi, sinyal pantul ini biasanya sudah ditambahkan dengan data yang dimiliki tag tersebut. RFID tag ukurannya dapat berbeda-beda, umumnya ukurannya kecil.

Berdasarkan catu daya tag, tag RFID dapat digolongkan menjadi 3 bagian seperti gambar di bawah:



Gambar 2.2 Penggolongan RFID Tag

➤ Tag Aktif

Tag yang catu dayanya diperoleh dari batere, sehingga akan mengurangi daya yang diperlukan oleh RFID Reader. Dan tag dapat mengirimkan informasi dalam jarak yang lebih jauh.

Kelemahan dari tag tipe ini adalah harga yang mahal dan ukurannya yang lebih besar karena lebih kompleks. Semakin banyak fungsi yang dapat dilakukan oleh tag, maka rangkaianannya akan semakin kompleks dan ukurannya semakin besar.

➤ Tag Pasif

Tag yang catu dayanya diperoleh dari medan yang dihasilkan oleh pembaca RFID. Rangkaianannya lebih sederhana, harganya jauh lebih murah, ukurannya kecil dan lebih ringan.

Kelemahannya adalah tag hanya dapat mengirimkan informasi dalam jarak yang dekat dan pembaca RFID harus menyediakan daya tambahan untuk tag RFID.

➤ Tag Semi Pasif

Tag pasif yang menggunakan energi dari baterai untuk keperluan rangkaian, tetapi tidak untuk menghasilkan sinyal. Ketika baterai digunakan untuk memberikan energi pada sensor, maka ini disebut sensor tag. Tag ini cenderung lebih kecil dan lebih murah jika dibandingkan dengan tag aktif, juga memiliki fungsi lebih jika dibandingkan dengan tag pasif karena lebih banyak energi yang tersedia untuk kegunaan lain.

2.4 Mikrokontroler ATMEGA 16

Mikrokontroler dewasa ini menjadi booming dikalangan bidang pendidikan, terutama perguruan tinggi. Mikrokontroler dengan merek ATMEL dengan tipenya meliputi ATtiny, AT90 dan Atmega. Masing-masing tipe memiliki fitur yang berbeda-beda dengan tujuan aplikasi yang spesifik.

Mikrokontroler sebagai sebuah "one chip solution" pada dasarnya adalah rangkaian terintegrasi (Integrated Circuit-IC) yang telah mengandung secara lengkap berbagai komponen pembentuk sebuah komputer. Berbeda dengan penggunaan mikroprosesor yang masih memerlukan komponen luar tambahan seperti RAM, ROM, Timer, dan sebagainya untuk sistem mikrokontroler, tambahan komponen diatas secara praktis hampir tidak dibutuhkan lagi. Hal ini disebabkan semua komponen penting tersebut telah ditanam bersama dengan sistem prosesor ke dalam IC tunggal mikrokontroler bersangkutan. Dengan alasan itu sistem mikrokontroler dikenal juga dengan istilah populer The Real Computer On a Chip (komputer utuh dalam keping tunggal), sedangkan sistem mikroprosesor dikenal dengan istilah yang lebih terbatas yaitu Computer On a Chip (komputer dalam keping tunggal).

Berdasarkan fungsinya, mikrokontroler secara umum digunakan untuk menjalankan program yang bersifat permanen pada sebuah aplikasi yang spesifik (misal aplikasi yang berkaitan dengan pengontrolan dan monitoring). Sedangkan program aplikasi yang dijalankan pada sistem mikroprosesor biasanya bersifat sementara dan berorientasi pada pengolahan data. Perbedaan fungsi kedua sistem diatas secara praktis mengakibatkan kebutuhan minimal yang harus dipenuhi juga akan berbeda (misal ditinjau dari kecepatan detak operasi, jumlah RAM, panjang register, dan lain sebagainya). Hampir tidak dapat disangkal, dewasa ini akan sukar dijumpai seseorang yang masih menggunakan komputer dengan mikroprosesor berbasis 8 atau 16 bit (misal mikroprosesor 8088 dan 8086 produk perusahaan Intel).

Mengapa demikian? Salah satu alasannya, perangkat lunak komputer yang beredar saat ini umumnya mensyaratkan kecepatan CPU yang sangat tinggi (dalam orde Mega bahkan GigaHz) serta memori dengan kapasitas sangat besar (dalam orde MegaByte) yang mana hal tersebut tidak mungkin dapat dipenuhi oleh sistem microprocessor lama tersebut. Sedangkan untuk sistem mikrokontroler, program yang dijalankan biasanya tidak memerlukan sumber daya sebanyak dan sebesar itu. Untuk aplikasi kontrol sederhana dan tingkat menengah, mikrokontroler yang digunakan cukup berbasis 4 sampai 8 bit. Mikrokontroler dengan ukuran lebih besar (misal 16 dan 32 bit) umumnya hanya digunakan untuk aplikasi-aplikasi khusus pada bidang pengolahan citra atau bidang kontrol yang memerlukan kepresisian tinggi.

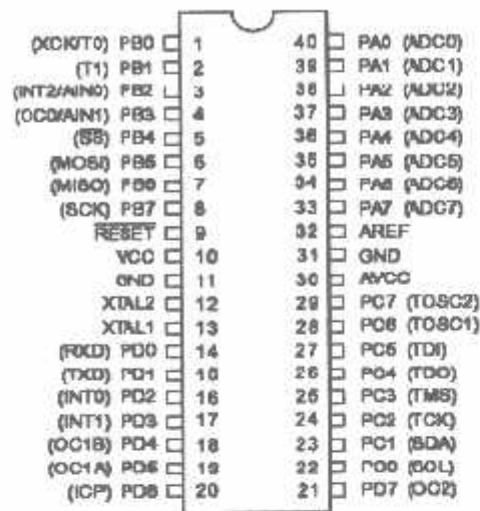
2.4.1 Arsitektur ATMEGA 16

Mikrokontroler ini menggunakan arsitektur Harvard yang memisahkan memori program dari memori data, baik bus alamat maupun bus data, sehingga pengaksesan program dan data dapat dilakukan secara bersamaan (concurrent). Secara garis besar mikrokontroler ATMEGA 16 terdiri dari:

1. Frekuensi clock maksimum 16 Mhz.
2. Kapasitas Flash memori 8 Kbyte dengan kemampuan read while write.
3. EEPROM (Electrically Erasable Program-mable Read Only Memory) sebesar 512 Byte yang dapat diprogram saat operasi.
4. SRAM internal sebesar 1 Kilo byte.
5. Jalur I/O 32 buah, yang terbagi dalam Port A, Port B, Port C dan Port D.
6. CPU 8 bit yang terdiri dari 32 buah register.
7. Interrupt internal maupun eksternal.
8. Port komunikasi SPI (Serial Pheripheral Interface)
9. Komunikasi serial standar USART dengan kecepatan maksimal 2,5 Mbp.
10. Analog komparator.
11. Watchdog timer dengan osilator internal.
12. Timer/counter sebanyak 3 buah.
13. Analog to Digital Converter (ADC) 10 bit sebanyak 8 input.

2.4.2 Konfigurasi ATMEGA 16

Konfigurasi pin-pin pada ATMEGA 16 ditunjukkan pada gambar 2.1 sebagai berikut:



Gambar 2.3 Konfigurasi ATMEGA 16

Konfigurasi pin mikrokontroler ATMEGA 16 terdiri dari 40 pin. Dari gambar di atas dapat terlihat ATMEGA 16 memiliki 8 pin untuk setiap portnya, port A, port B, port C dan port D.

Deskripsi Mikrokontroler ATMEGA 16

- VCC
 - Sebagai tegangan penyuplai.
- Ground
 - Sebagai ground
- Port A (PA7 ... PA0)
 - Port A sebagai input analog ke A/D konverter. Port A juga sebagai 8-bit bi-directional port I/O, jika A/D konverter tidak digunakan. Pin-pin port dapat menyediakan resistor-resistor internal pull-up. Ketika port PA0...PA7 digunakan sebagai input dan pull eksternal yang rendah akan menjadi sumber arus jika resistor-resistor pull-up diaktifkan. Pin-pin port A adalah tri-state ketika kondisi reset menjadi aktif sekalipun clock tidak aktif.
- Port B (PB7 ... PB0)

Port B adalah port I/O 8-bit bi-directional dengan resistor-resistor internal pull-up. Buffer output port B mempunyai karakteristik drive yang simetris dengan kemampuan keduanya sink dan source yang tinggi. Sebagai input, port B yang mempunyai pull eksternal yang rendah akan menjadi sumber arus jika resistor-resistor pull-up diaktifkan. Pin-pin port B adalah tri-state ketika kondisi reset menjadi aktif sekalipun clock tidak aktif.

➤ Port C (PC7 ... PC0)

Port C adalah port I/O 8-bit bi-directional dengan resistor-resistor internal pull-up. Buffer output port C mempunyai karakteristik drive yang simetris dengan kemampuan keduanya sink dan source yang tinggi. Sebagai input, port C yang mempunyai pull eksternal yang rendah akan menjadi sumber arus jika resistor-resistor pull-up diaktifkan. Pin-pin port C adalah tri-state ketika kondisi reset menjadi aktif sekalipun clock tidak aktif. Jika antarmuka JTAG enable, resistor-resistor pull up pada pin-pin PC5(TDI), PC3(TMS), PC2(TCK) akan diaktifkan sekalipun terjadi reset.

➤ Port D (PD7 ... PD0)

Port D adalah port I/O 8-bit bi-directional dengan resistor-resistor internal pull-up. Buffer output port D mempunyai karakteristik drive yang simetris dengan kemampuan keduanya sink dan source yang tinggi. Sebagai input, port D yang mempunyai pull eksternal yang rendah akan menjadi sumber arus jika resistor pull-up diaktifkan.

➤ RESET (Reset input)

➤ XTAL1 (Input oscillator)

➤ XTAL2 (Output oscillator)

➤ AVCC

Pin penyedia tegangan untuk Port A dan konverter A/D.

➤ AREF

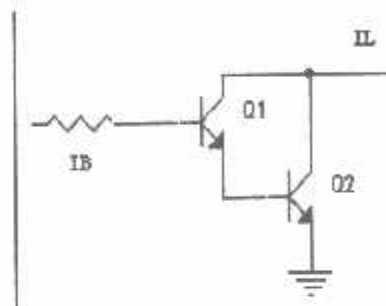
Pin referensi analog untuk konverter A/D.

2.5 Relay

Relay merupakan salah satu komponen output yang paling sering digunakan baik pada industri, otomotif, ataupun peralatan elektronika lainnya. Berfungsi sebagai pemutus dan penghubung aliran arus ke sistem pengapian dan kelistrikan dan alarm.

Relay adalah sebuah saklar auto-mekanik. Relay terdiri dari kumparan yang dapat menimbulkan medan magnet jika dialiri arus dan akan menarik beberapa kontakannya.

Ada 2 macam relay berdasarkan tegangan untuk menggerakkan koilnya yaitu AC dan DC. Pada umumnya relay DC dengan tegangan koil 12V DC, memerlukan arus sekitar 20-30mA. Karena itu tidak dapat langsung menghubungkan output suatu IC logic (TTL/CMOS) atau peripheral lain seperti μC 89C51, PPI 82C55 dengan relay karena I_{OHmax} (arus maximum yang dikeluarkan pada saat logic '1') atau I_{OLmax} (arus maximum yang mampu dibenamkan pada saat logic '0') tidak cukup besar. Karena itu perlu digunakan driver untuk penguat arus yang biasanya berupa transistor dalam konfigurasi Darlington seperti Gambar 2.4 berikut ini.



Gambar 2.4 Rangkaian Darlington

Rangkaian tersebut juga dilengkapi dengan suppression diode. Dioda ini berfungsi untuk mencegah kickback yaitu transient yang terjadi pada koil relay (beban induktif) saat relay dimatikan. Bila terjadi perubahan arus yang cukup besar dalam satuan waktu yang sangat cepat maka tegangan balik ini menjadi sangat besar, dan dapat mengakibatkan kerusakan pada transistor.

Problem lain yang sering terjadi pada kontak relay adalah loncatan bunga api listrik yang dapat memperpendek umur kontak. Bunga api ini terutama terjadi pada beban induktif seperti motor, solenoid, dll. Untuk mencegah hal ini dapat digunakan MOV (Metallic Oxide Varistor) yang dipasang secara paralel dengan kontak. Varistor bersifat seperti resistor dengan nilai resistansinya tergantung pada tegangan. Ketika kontak terbuka, beban induktif menghasilkan tegangan balik yang cukup besar akibat perubahan medan magnet.

Pada saat ini ('protective state') nilai resistansi varistor menjadi sangat kecil dan arus akan mengalir melalui MOV, sehingga transient dapat diredam. Pada saat keadaan normal resistansi MOV sangat besar dan hanya menarik arus yang sangat kecil, bunga api juga menyebabkan sinyal Radio Frequency Interference (RFI) yang dapat mengganggu peralatan – peralatan sensitif.

Karena itu komponen peredam transient seperti MOV sangat diperlukan terutama pada beban induktif. Parameter-parameter penting Varistor :

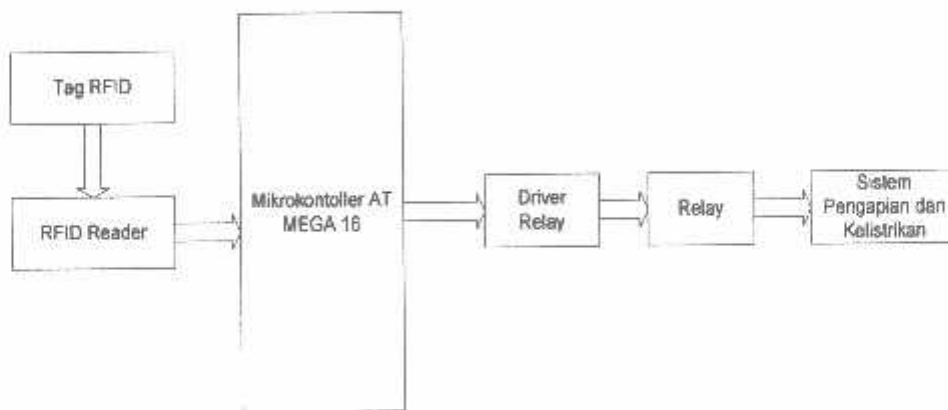
- Tegangan Varistor (tegangan breakdown):
Tegangan pada varistor yang diukur pada arus tertentu (0,1 mA atau 1 mA) selama waktu tertentu.
- Tegangan maksimum yang diperbolehkan:
Tegangan maksimum pada varistor agar tetap pada keadaan normal.

BAB III PERANCANGAN SISTEM

Perancangan sistem pengamanan sepeda motor ini terbagi atas dua bagian, yaitu perancangan hardware dan perancangan software. Perancangan hardware terbagi atas perancangan sistem control, perancangan unit masukan, perancangan dan unit keluaran. Sedangkan perancangan software terdiri dari perancangan program basic (BASCOM).

3.1 Blok Diagram Rangkaian

Pada bagian ini penulis akan membahas tentang alat yang meliputi diagram blok dan realisasi rangkaian. Komponen dalam alat ini dirangkai berdasarkan prinsip kerja masing-masing komponen. Diagram blok rangkaian sebagai berikut:



Gambar 3.1 Blok Diagram

Cara Kerja Kunci Pengaman Sepeda Motor Menggunakan RFID

Adapun cara kerja rangkaian dari kunci kendaraan bermotor menggunakan RFID adalah sebagai berikut ini.

Kartu tag ID dan RFID yang digunakan adalah sebagai media input utama yang berfungsi sebagai pengaman kendaraan bermotor. Kartu tag ID berisikan kode-kode spesifik atau unik dimana setiap kartu tag ID memiliki kode-kode yang berbeda-beda. Hal inilah yang dimanfaatkan sebagai kunci untuk menghidupkan atau memutuskan kabel saklar dan busi serta alaram yang terpasang. Untuk menghidupkan kontak kendaraan bermotor kita terlebih dahulu mengidentifikasi kartu tag ID ke RFID, jika

kartu tag ID memiliki kode yang sesuai dengan input pada mikrokontroller maka mikrokontroller akan memberikan output pada relay menyalakan motor.

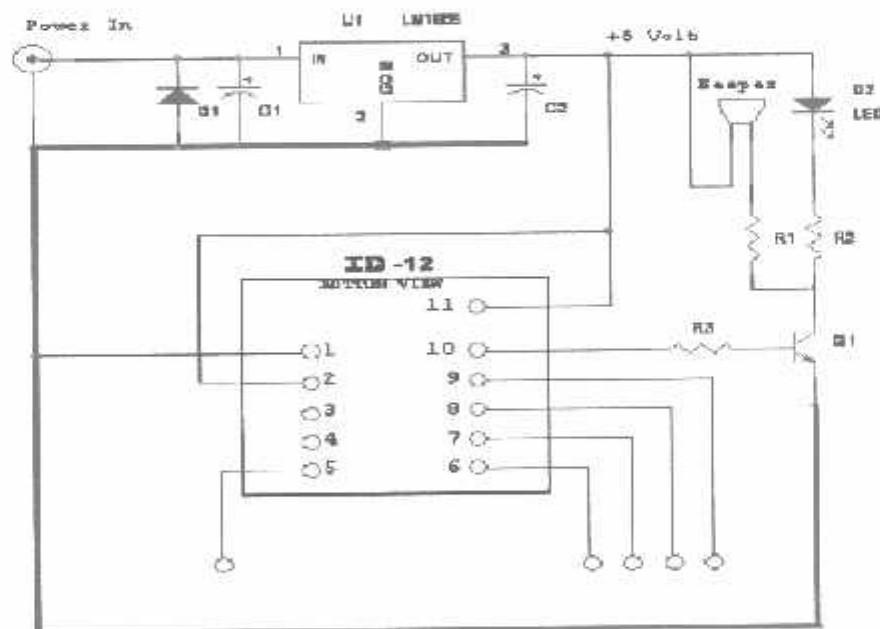
3.2 Perancangan Sistem Perangkat Keras

3.2.1 Tag RFID

Tag RFID adalah alat yang dibuat dari rangkaian elektronika dan antena yang terintegrasi di dalam rangkaian tersebut. Rangkaian elektronik dari Tag RFID umumnya memiliki memori sehingga Tag RFID ini mempunyai kemampuan untuk menyimpan data. Memori pada Tag RFID dibagi menjadi sel-sel. Beberapa sel menyimpan data Read Only, misalnya serial number yang unik yang disimpan pada saat Tag RFID tersebut diproduksi. Sel lain pada Tag RFID mungkin juga dapat ditulis dan dibaca secara berulang. Tag yang digunakan adalah Tag Card Pasif yaitu Tag RFID yang catu dayanya diperoleh dari medan listrik yang dihasilkan oleh RFID Reader, rangkaianannya lebih sederhana, dan harganya jauh lebih murah, ukurannya kecil, dan lebih ringan. Kelemahannya adalah Tag RFID hanya dapat mengirimkan informasi dalam jarak yang dekat.

3.2.2 Rangkaian Modul RFID

Jenis RFID Reader yang digunakan adalah tipe ID-12 buatan Innovative Electronics.

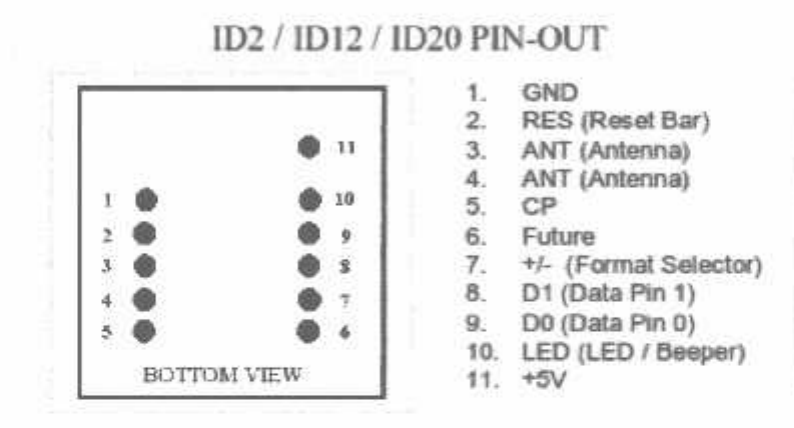


Gambar 3.2 Rangkaian Modul RFID Reader

Keterangan komponen :

R1 = 100R

R2 = 1K
 R3 = 1K
 C1 = 100 μ F 16V
 C2 = 100 μ F 10V
 Beeper = 2,7-3.5 KHz 100R
 D1 = 1N4001
 D2 = Green LED
 U1 = LM7805
 Q1 = UTC8050 (NPN)
 ID-12 = ID Innovations ID-12



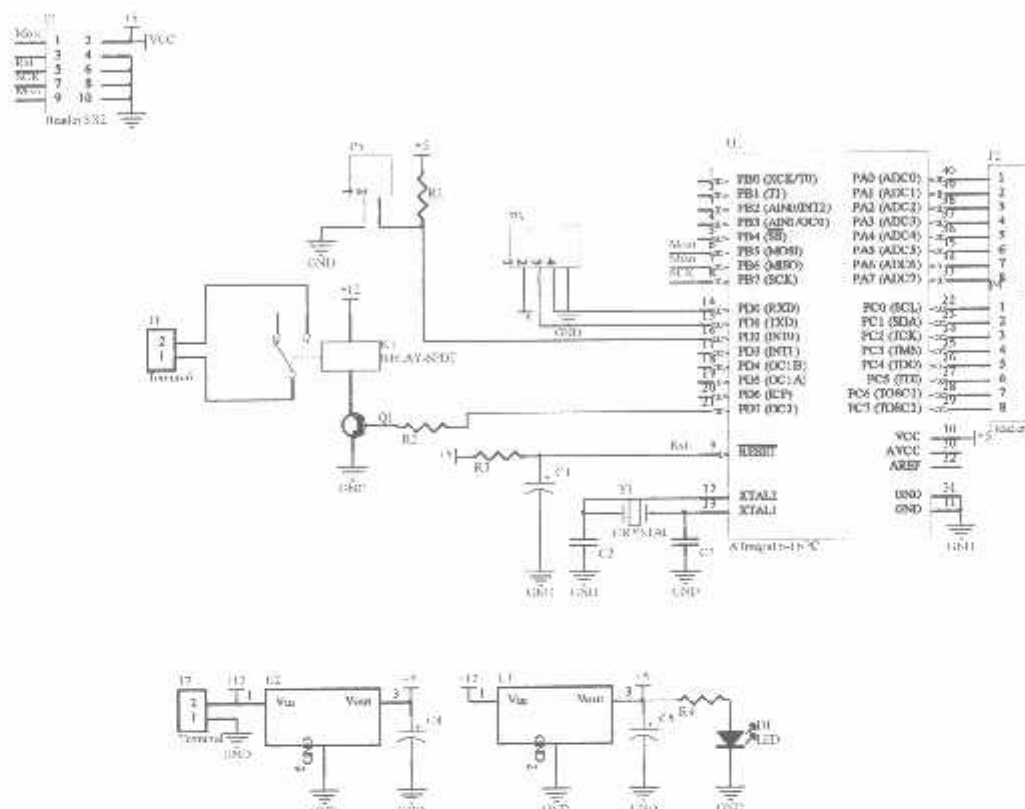
Gambar 3.3 Konfigurasi Pin RFID Reader

Spesifikasi Modul RFID :

1. Berbasis RFID reader ID-12 dengan frekuensi 125 kHz untuk kartu berformat EM401/sejenis dan memiliki jarak baca 12 cm.
2. Kompatibel dengan varian RFID reader lainnya, antara lain : ID-2, ID-10 dan ID-20.
3. Mendukung varian RFID reader/writer, antara lain : ID-2RW, ID-12RW dan ID-20RW.
4. Mendukung format data ASCII (UART TTL/ RS-232), Wiegand26, maupun Magnetic ABA Track2 (Magnet Emultion).
5. Dilengkapi dengan buzzer sebagai indikator baca, serta LED sebagai indikator tulis.
6. Tersedia jalur komunikasi serial UART RS-232 dengan konektor RJ11.
7. Tegangan input catu daya 9-12 VDC (J2)

3.2.3 Perancangan Mikrokontroler

Jenis mikrokontroler yang digunakan pada sistem ini adalah Atmega16, yang memiliki saluran I/O sebanyak 32 buah, yaitu Porta A, Port B, Port C, dan Port D serta memiliki ADC 10 bit sebanyak 8 saluran. Mikrokontroler dalam sistem ini berfungsi untuk menerima dan mengelola data yang dikirim dari RFID Reader setelah membaca data pada ID Tag.

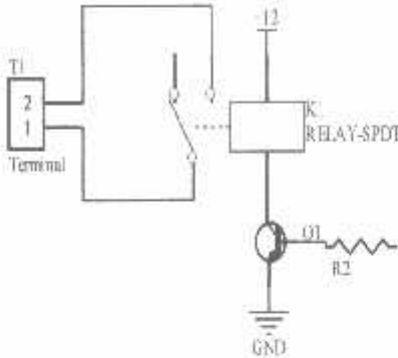


Gambar 3.4 Rangkaian Minimum Sistem Mikrokontroler

3.2.4 Perancangan Driver Relay

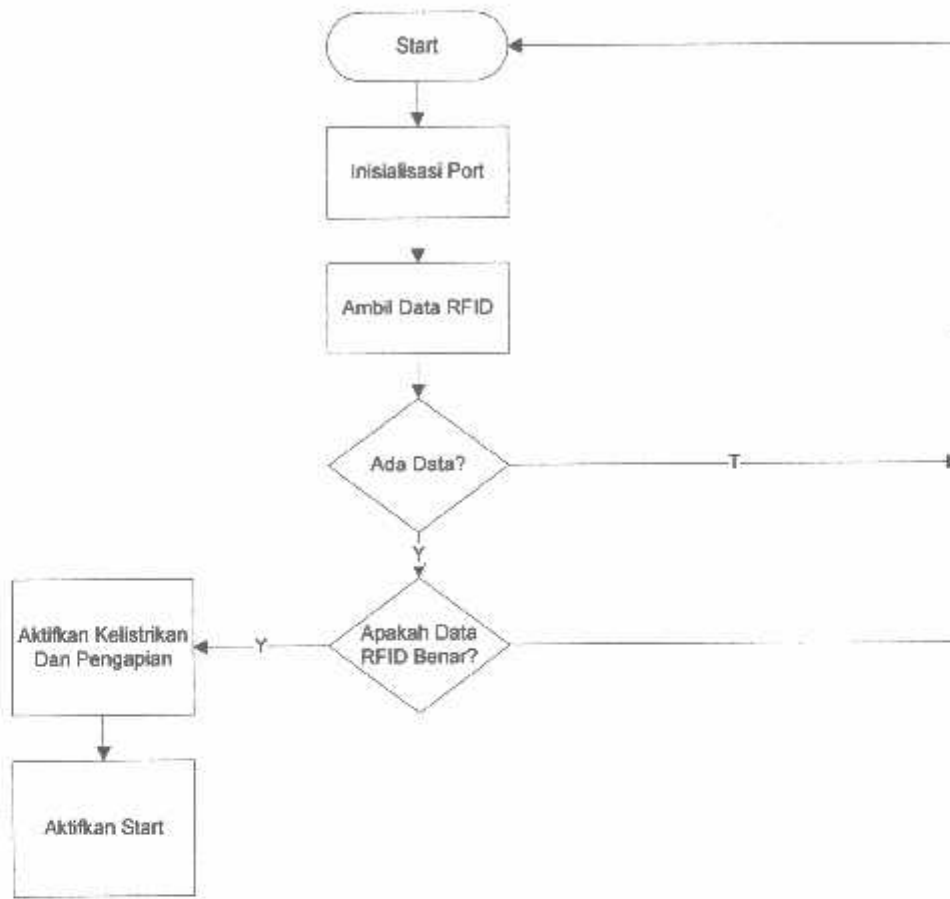
Rangkaian driver relay berasal dari rangkaian mikrokontroler. Terdapat 8 jalur keluaran dari unit sistem control alat, di gunakan 1 jalur yang dihubungkan ke relay. Tegangan keluaran dari mikrokontroler dihubungkan ke kaki basis transistor Q1 dan Q2 melalui resistor pembatas arus R2. Dimana jika masukan pada kaki basis bertegangan tinggi maka transistor Q1 akan menghantar arus melalui relay, sehingga saklar relay akan otomatis menutup dan arus listrik pada CDI sepeda motor akan terhubung dan sepeda motor dapat dinyalakan. Sedangkan jika masukan pada kaki

basis transistor Q1 bertegangan nol, transistor Q1 akan menyumbat sehingga arus tidak akan mengalir melalui relay. Dimana saklar relay akan terbuka dan arus listrik pada CDI sepeda motor tidak akan terhubung, sehingga sepeda motor tidak akan bisa dinyalakan.



Gambar 3.5 Rangkaian Driver Relay

3.3 Diagram Alir Sistem



Gambar 3.6 Flowchart Sistem

BAB IV PEMBUATAN DAN PENGUJIAN SISTEM

Untuk mengetahui apakah tujuan-tujuan dari pembuatan alat ini telah terlaksana atau tidak, perlu dilakukan pengujian dan analisa terhadap alat yang dibuat. Dan sebagai bagian yang tak terpisahkan adalah adanya proses evaluasi sehingga akan dapat dilakukan langkah-langkah positif guna membawa alat ini ke arah yang lebih baik.

Pengujian yang dilakukan terdiri dari dua tahap, tahapan-tahapan tersebut adalah sebagai berikut :

1. Pengujian unit sistem blok RFID sistem.
2. Pengujian modul RFID Reader, pengujian tersebut terdiri dari :
 - a. Pengujian jarak pembacaan RFID Reader terhadap tag RFID
 - b. Pengujian posisi tag RFID ketika pembacaan terjadi
 - c. Pengujian pembacaan multi tag
3. Pengujian perangkat lunak
4. Pengujian sistem secara keseluruhan.

4.1 Pengujian modul RFID

Dalam uji coba ini, modul akan diuji tanpa menghubungkan ke rangkaian minimum mikrokontroler, CDI sepeda motor dan RFID reader tidak difungsikan untuk membaca tag RFID.

Tabel 4.1 Pengujian Modul RFID Tanpa Difungsikan

Komponen	Status	Keterangan
Led Power Supply	Menyala	Sesuai rancangan
Led Clock	Berkedip	Sesuai rancangan
Buzzer	Off	Sesuai rancangan

Dari hasil pengujian modul RFID pada saat modul tidak difungsikan LED power supply akan menyala, sedangkan LED clock akan berkedip dan buzzer tidak berbunyi menandakan bahwa tidak ada proses pembacaan yg dilakukan oleh RFID Reader.

4.2 Pengujian Jarak Pembacaan RFID Reader Terhadap Tag RFID Dari Berbagai Arah

Dalam modul RFID ID-12 Innovations, kemampuan baca RFID Reader memiliki kemampuan jarak pembacaan maksimalnya mencapai 12 cm, oleh karenanya dalam uji coba ini akan diukur kemampuan maksimum jarak baca RFID Reader terhadap tag RFID. Dalam hal ini pengujian pembacaan dicoba dari berbagai arah.

4.2.1 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Atas

Pengujian dilakukan dengan posisi tag berada di atas RFID Reader. Uji coba ini bertujuan untuk mengetahui berapa jarak pendeteksian RFID Tag Card yang dapat dilakukan oleh RFID Reader. Pengujian dilakukan dengan mendekatkan RFID Tag Card ke RFID Reader dengan jarak tertentu dan kemudian diukur oleh mistar ukur. Apabila RFID Tag Card terdeteksi oleh RFID Reader maka buzzer pada rangkaian akan berbunyi.

Tabel 4.2 Jarak pengukuran RFID Reader Terhadap Tag RFID Dari Arah Atas

Jarak (cm)	Kartu 1	Kartu 2	Kartu 3	Kartu 4	Kartu 5
12	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
11	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
8	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
7	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
6	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
4	Terdeteksi	Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
3	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi
2	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi
1	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi

Dari tabel 4.2 maka dapat diambil kesimpulan bahwa semakin kecil jarak antara RFID Reader dengan tag, maka nilai kesalahan pembacaan akan semakin kecil seperti ditunjukkan oleh tabel diatas. Dan didapatkan besar jarak antara reader dan tag pada arah atas mempunyai akurasi pembacaan sampai 100% pada jarak antara 0-3 cm.

4.2.2 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Kanan

Pengukuran jarak baca dengan posisi tag dengan bagian tepi tag sebelah kanan reader.

Tabel 4.3 Jarak pengukuran RFID Reader Terhadap Tag RFID Dari Arah Kanan

Jarak (cm)	Kartu 1	Kartu 2	Kartu 3	Kartu 4	Kartu 5
12	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
11	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
8	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
7	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
6	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
3	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
2	Terdeteksi	Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
1	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi

Dari tabel 4.3 maka dapat diambil kesimpulan bahwa semakin kecil jarak antara RFID Reader dengan tag, maka nilai kesalahan pembacaan akan semakin kecil seperti

ditunjukkan oleh tabel diatas. Dan didapatkan besar jarak antara reader dan tag pada arah kanan RFID Reader mempunyai akurasi pembacaan sampai 100% pada jarak antara 0-1 cm.

4.2.3 Pengujian Jarak Baca RFID Reader Terhadap Tag RFID Dari Arah Kiri

Pengukuran hasil percobaan dilakukan dengan posisi tag berada pada bagian tepi sebelah kiri RFID Reader.

Tabel 4.4 Jarak pengukuran RFID Reader Terhadap Tag RFID Dari Arah Kiri

Jarak (cm)	Kartu 1	Kartu 2	Kartu 3	Kartu 4	Kartu 5
12	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
11	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
8	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
7	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
6	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
5	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
4	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
3	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
2	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
1	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi	Terdeteksi

Dari tabel 4.4 maka dapat diambil kesimpulan bahwa semakin kecil jarak antara RFID Reader dengan tag, maka nilai kesalahan pembacaan akan semakin kecil seperti ditunjukkan oleh tabel diatas. Dan didapatkan besar jarak antara reader dan tag dari

arah kiri RFID Reader mempunyai akurasi pembacaan sampai 100% pada jarak antara 0-1 cm.

Dari hasil percobaan yang telah dilakukan, maka didapat kesimpulan bahwa arah pembacaan Tag RFID yang paling maksimal adalah berada pada bagian atas RFID Reader. Hal ini dipengaruhi oleh pola radiasi dari antena RFID.

Dimana pembacaan Tag ID belum maksimal, karena pada data sheet RFID Reader harus mampu membaca Tag ID pada jarak 12 cm, namun pada pengujian alat RFID hanya mampu membaca Tag ID pada jarak 0 – 4 cm.

4.3 Pengujian Pembacaan Multi Tag

Pada pengujian ini, jarak baca yang dipakai adalah 2 cm pada arah atas. Pengujian ini dibatasi hanya pada pembacaan tiga tag secara bersamaan dimana posisi tagnya berada pada garis sejajar dan bukan ditindih.

Tabel 4.5 Hasil Pengujian Multi Tag

Percobaan	Single Tag	Double Tag	Triple Tag
1	Terbaca	Tidak Terbaca	Tidak Terbaca
2	Terbaca	Terbaca	Tidak Terbaca
3	Terbaca	Tidak Terbaca	Tidak Terbaca
4	Terbaca	Tidak Terbaca	Tidak Terbaca
5	Terbaca	Terbaca	Tidak Terbaca

Dari tabel 4.5 maka dapat diambil kesimpulan bahwa pembacaan tag hanya dapat dilakukan pada satu tag dalam satu waktu, apabila beberapa tag berada pada jarak baca RFID maka tag dengan jarak paling dekatlah yang akan diidentifikasi oleh reader.

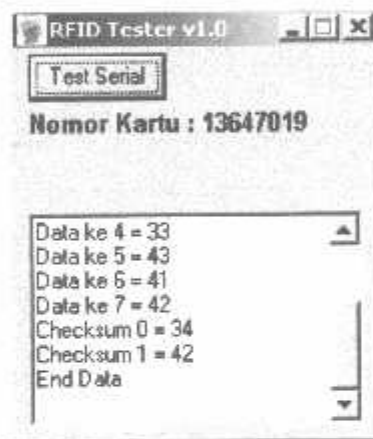
4.4 Pengujian Pembacaan Kode Tag RFID

Dalam pengujian kali ini, akan diuji pembacaan kode yang terdapat dalam tag RFID dengan menggunakan RFID Tester v1.0. Ada lima tag RFID yang diuji dimana dalam pengujian ini bertujuan untuk mengetahui apakah program dapat membaca dan membedakan tag.



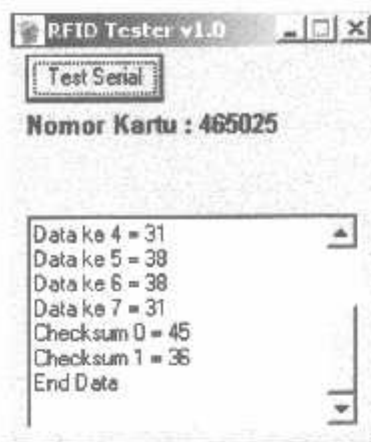
Gambar 4.1 Hasil Pengujian Tag Pertama

Dari pengujian tag yg pertama ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.



Gambar 4.2 Hasil Pengujian Tag Kedua

Dari pengujian tag yg kedua ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.



Gambar 4.3 Hasil Pengujian Tag Ketiga

Dari pengujian tag yg ketiga ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.



Gambar 4.4 Hasil Pengujian Tag Keempat

Dari pengujian tag yg keempat ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.



Gambar 4.5 Hasil Pengujian Tag Kelima

Dari pengujian tag yg kelima ini didapat hasil bahwa RFID Reader dapat membaca dengan benar data yang ada didalam Tag RFID.

Sehingga dari percobaan diatas dapat disimpulkan bahwa program yang dibuat dapat berjalan dengan lancar dalam sistem pembacaan dan tanpa mengalami error sama sekali.

4.5 Pengujian Sistem Secara Keseluruhan

Pada pengujian ini dilakukan dengan dua tahapan, dimana tahapan terdiri dari :

1. Pengujian keseluruhan dengan menggunakan Tag RFID yang terdata didalam program, terdiri dari satu tag RFID.
2. Pengujian keseluruhan dengan menggunakan Tag RFID yang tidak terdata didalam program, terdiri dari empat Tag RFID tidak terdata.
- 3.

Tabel 4.6 Hasil Tag Terdata dan Tidak Terdata

Percobaan	Jenis tag	Start
1	Terdata	Ya
2	Tidak Terdata	Tidak
3	Tidak Terdata	Tidak
4	Tidak Terdata	Tidak
5	Tidak Terdata	Tidak

Pengujian ini dilakukan setelah pembuatan hardware dan software selesai dan program sudah didownload keadalam mikrokontroler. Tujuan pengujian ini untuk

mengetahui apakah alat yang dirancang sudah berjalan sesuai dengan yang diinginkan atau belum. Adapun yang akan diuji adalah pembacaan Tag ID dengan RFID Reader. Pengujian ini dilakukan untuk mengecek apakah perintah yang diinputkan sudah sesuai dengan output.

Pada percobaan pertama yaitu dengan menggunakan tag RFID yg tidak terdata diprogram. Maka didapatkan bahwa sepeda motor tidak dapat dihidupkan walaupun kunci kuntak sudah berada pada posisi on.

Pada percobaan yang kedua yaitu dengan menggunakan tag RFID yang sudah terdata dalam program. Maka didapat kesimpulan bahwa sepeda motor dapat dinyalakan karena data yang ada terbaca oleh RFID reader sesuai dengan yang terdata dalam program.

BAB V

PENUTUP

Dari perancangan dan pengujian alat, dapat ditarik kesimpulan dan saran mengenai skripsi sebagai berikut :

5.1 Kesimpulan

1. Dengan menggunakan program BASCOM-AVR penulis dapat membuat program, sistem kendali lebih mudah karena program ini cukup lengkap dengan adanya simulator untuk LED dan LCD sehingga penulis bisa melihat program yang dibuat didalam simulasi dan sebelum didownloadkan kedalam mikrokontroler.
2. Penggunaan alat dilakukan dengan cara menghubungkan perangkat lunak (Software) dan perangkat keras (Hardware) sehingga dapat dilihat kemampuan program dalam mengendalikan mikrokontroler.
3. Pembacaan Tag ID belum maksimal, karena pada data sheet RFID Reader harus mampu membaca Tag ID pada jarak 12 cm, namun pada pengujian alat RFID hanya mampu membaca Tag ID pada jarak 0 – 4 cm.
4. Bahwa arah pembacaan Tag RFID yang paling optimal adalah pada bagian atas RFID. Hal ini dipengaruhi oleh pola radiasi dari antena RFID.
5. Pembacaan tag hanya dapat dilakukan pada satu tag dalam satu waktu, apabila beberapa tag dalam range baca maka tag dengan jarak paling dekatlah yang akan diidentifikasi oleh reader.

5.2 Saran

1. Sebelum mendownload program kedalam mikrokontroler, sebaiknya compile terlebih dahulu program untuk memastikan program sudah berjalan dengan benar.
2. Diharapkan kepada pembaca dan semua orang yang berminat dengan keamanan sepeca motor agar dapat mengembangkan program untuk dapat lebih mengoptimalkan kerja dari sistem kendali.

Penulis sangat mengharapkan sumbangan ilmu berupa kritik dan saran dari pembaca, karena penulis sadar keterbatasan kemampuan sehingga diharapkan masukan agar dapat lebih berkembang lagi dan mengharapkan kedepan ada yang

mengembangkan alat ini jauh lebih baik. Dan semoga karya ini dapat bermanfaat bagi kita semua.

DAFTAR PUSTAKA

- [1]. *AVR AT Teknik Antar muka dan Pemograman Mikrokontroller AT MEGA 16* yogyakarta: Penerbit ANDI
- [2]. *Belajar Sendiri Mikrokontroller AVR Seri AT Mega 16* yogyakarta: penerbit ANDI
- [3]. Alfian, Moh. Yusuf : *Pengembangan Modul Praktikum, Pengaturan Otomatis :* Proyek Akhir D4 PENS-ITS, 2007
- [4]. Atmel Corp. Application Note. Datasheet of Atmega 16 datasheet
- [5]. Innovative Electronics. Manual RFID Starter Kit, 2007
- [6]. Innovative Electronics. ID Series Datasheet, 10 Februari 2004
- [7]. Innovative Electronics. ID Series Datasheet, 1 Maret 2005

D
A
F
7
A
R

L
A
M
P
9
R
A
N

Listing Program

\$regfile = "m16def.dat"

\$crystal = 8000000

\$baud = 9600

Relay Alias Portd.7

Config Relay = Output

Reset Relay

Dim S As String * 16

Dim Temps As String * 2

Dim Datl(8) As Long

Dim DatsAs String * 10

Dim A As Byte

Dim SttAs Byte

Dim I As Byte

Dim P As Byte

Dim Buff(16) As Byte

Dim D As Byte

Dim Al As Long

Dim Templ(8) As Long

Dim StrrAs Byte

ConstAddre = 1

Datl(8) = 268435456

Datl(7) = 16777216

Datl(6) = 1048576

Datl(5) = 65536

Datl(4) = 4096

Datl(3) = 256

Datl(2) = 16

Datl(1) = 1

ReadepromSstr ,Addre

If Sstr = 1 Then

Set Relay

Else

Reset Relay

End If

Do

Input S

Dats = "" : A = 0 : Stt = 0

For I = 1 To Len(s)

Temps = Mid(s , I , 1)

P = Asc(temps)

If Stt = 1 Then

Incr A

If A > 2 And A < 11 Then

Dats = Dats + Temps

End If

End If

If P = 2 Then Stt = 1

Buff(i) = Asc(temps)

Next

A = 0

For I = Len(dats) To 1 Step - 1

Temps = Mid(dats , I , 1)

If Temps = "A" Then Temps = "10"

If Temps = "B" Then Temps = "11"

If Temps = "C" Then Temps = "12"

If Temps = "D" Then Temps = "13"

If Temps = "E" Then Temps = "14"

If Temps = "F" Then Temps = "15"

Incr A

```
D = Val(temps)
Templ(i) = Dat1(a) * D
Next
  AI = 0
  For I = 1 To 8
    AI = AI + Templ(i)
  Next I
  Wait 2
  Print "Nomerkartu = " ; AI
  If AI = 268575304 Then
    Relay = Not Relay
    If Sstr = 1 Then
      Sstr = 0
    Else
      Sstr = 1
    End If
    WriteepromSstr ,Addre
  End If

Loop
End                                     'end program
```



PERMOHONAN PERSETUJUAN SKRIPSI

Yang Beranda Tangan Di bawah Ini:

Nama : DAULAT KRISTON NAINGGOLAN
 NIM : 0217057
 Semester :
 Fakultas : Teknologi Industri
 Jurusan : Teknik Elektro S-I
 Konsentrasi : TEKNIK ENERGI LISTRIK
TEKNIK ELEKTRONIKA
TEKNIK KOMPUTER DAN INFORMATIKA
TEKNIK KOMPUTER
TEKNIK TELEKOMUNIKASI

Alamat :


Dengan ini kami mengajukan permohonan untuk mendapatkan persetujuan untuk membuat SKRIPSI Tingkat Sarjana. Untuk melengkapi permohonan tersenut, bersama ini kami lampirkan persyaratan-persyaratan yang harus dipenuhi.

Adapun persyaratan- persyaratan pengambilan SKRIPSI adalah sebagai berikut:

- | | |
|--|---------|
| 1. Telah melaksanakan semua praktikum sesuai dengan konsentrasinya | (.....) |
| 2. Telah lulus dan menyerahkan laporan Praktek Kerja | (.....) |
| 3. Telah lulus seluruh mata kuliah keahlian (MKB)sesuai konsentrasinya | (.....) |
| 4. Telah menempuh matakuliah > 134 sks dengan IPK > 2 dan tidak ada nilai E | (.....) |
| 5. Telah mengikuti secara aktif kegiatan seminar Skripsi yang diadakan Jurusan | (.....) |
| 6. Memenuhi persyaratan administrasi | (.....) |

Demikian permohonan ini untuk mendapatkan penyelesaian lebih lanjut dan atas perhatiannya kami ucapkan terima kasih.

Telah diteliti kebenarannya data tersebut diatas
Recording Teknik Elektro S-I


 (Purni Wardayanti)

Malang, 20 - 06 - 2013

Pemohon

 (D. KRISTON NAINGGOLAN)

Disetujui
Ketua Prodi Teknik Elektro S-I


M. Ibrahim Ashari, ST, MT
NIP. P. 1030100358

Mengetahui
Dosen Wali


 (.....)

Catatan:

Bagi mahasiswa yang telah memenuhi persyaratan mengambil SKRIPSI agar membuat proposal dan mendapat persetujuan dari Ketua Prodi T. elektro S-I

1. IP 381 / = 2.76
 2. 138



ampiran : 1 (satu) berkas
Pembimbing Skripsi

epada : Yth. Bapak/Ibu Irmalia Suryani Faradisa, ST, MT
Dosen Teknik Elektro S-1
ITN Malang

Yang bertanda tangan dibawah

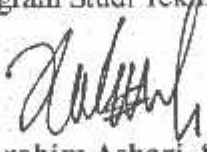
Nama : DAULAT KRISTON NAINGGOLAN
Nim : 0217057
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing untuk penyusunan Skripsi dengan judul :


"PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR MENGGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16"

Demikian permohonan kami buat dan atas kesediaan Bapak kami ucapkan terima kasih.

Mengetahui
Ketua Program Studi Teknik Elektro S-1


M. Ibrahim Ashari, ST, MT
NIP.P. 1030100358

Hormat Kami


DAULAT KRISTON NAINGGOLAN
NIM. 0217057

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

	7	0	Addr.	
General Purpose Working Registers	R0		\$00	
	R1		\$01	
	R2		\$02	
	...			
	R13		\$0D	
	R14		\$0E	
	R15		\$0F	
	R16		\$10	
	R17		\$11	
	...			
	R26		\$1A	X-register Low Byte
	R27		\$1B	X-register High Byte
	R28		\$1C	Y-register Low Byte
	R29		\$1D	Y-register High Byte
	R30		\$1E	Z-register Low Byte
	R31		\$1F	Z-register High Byte

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.

16 X-register, Y-register and Z-register

The registers R26..R31 have some added functions to their general purpose usage. These registers are 16-bit address pointers for indirect addressing of the Data Space. The three indirect address registers X, Y, and Z are defined as described in Figure 5.

Figure 5. The X-register, Y-register, and Z-register



In the different addressing modes these address registers have functions as fixed displacement, automatic increment, and automatic decrement (see the Instruction Set Reference for details).

Stack Pointer

The Stack is mainly used for storing temporary data, for storing local variables and for storing return addresses after interrupts and subroutine calls. The Stack Pointer Register always points to the top of the Stack. Note that the Stack is implemented as growing from higher memory locations to lower memory locations. This implies that a Stack PUSH command decreases the Stack Pointer. If software reads the Program Counter from the Stack after a call or an interrupt, unused bits (15:13) should be masked out.

The Stack Pointer points to the data SRAM Stack area where the Subroutine and Interrupt Stacks are located. This Stack space in the data SRAM must be defined by the program before any subroutine calls are executed or interrupts are enabled. The Stack Pointer must be set to point above \$60. The Stack Pointer is decremented by one when data is pushed onto the Stack with the PUSH instruction, and it is decremented by two when the return address is pushed onto the Stack with subroutine call or interrupt. The Stack Pointer is incremented by one when data is popped from the Stack with the POP instruction, and it is incremented by two when data is popped from the Stack with return from subroutine RET or return from interrupt RETI.

The AVR Stack Pointer is implemented as two 8-bit registers in the I/O space. The number of bits actually used is implementation dependent. Note that the data space in some implementations of the AVR architecture is so small that only SPL is needed. In this case, the SPH Register will not be present.

Bit	15	14	13	12	11	10	9	8	
	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SPH
	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	



Instruction Execution Timing

This section describes the general access timing concepts for instruction execution. The AVR CPU is driven by the CPU clock clk_{CPU} , directly generated from the selected clock source for the chip. No internal clock division is used.

Figure 6 shows the parallel instruction fetches and instruction executions enabled by the Harvard architecture and the fast-access Register File concept. This is the basic pipelining concept to obtain up to 1 MIPS per MHz with the corresponding unique results for functions per cost, functions per clocks, and functions per power-unit.

Figure 6. The Parallel Instruction Fetches and Instruction Executions

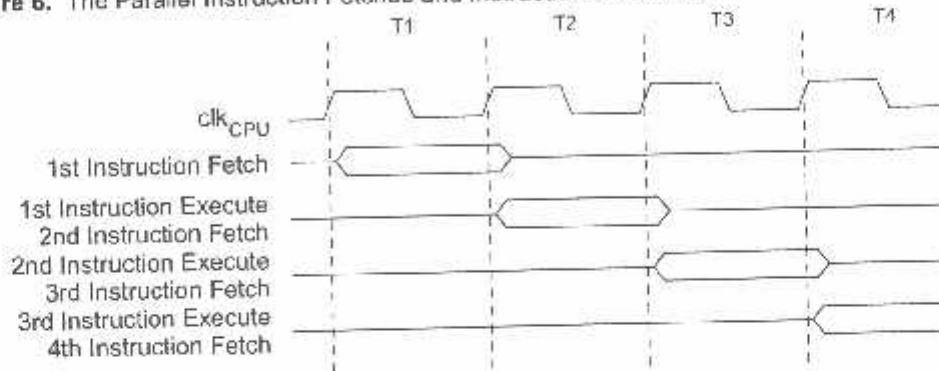
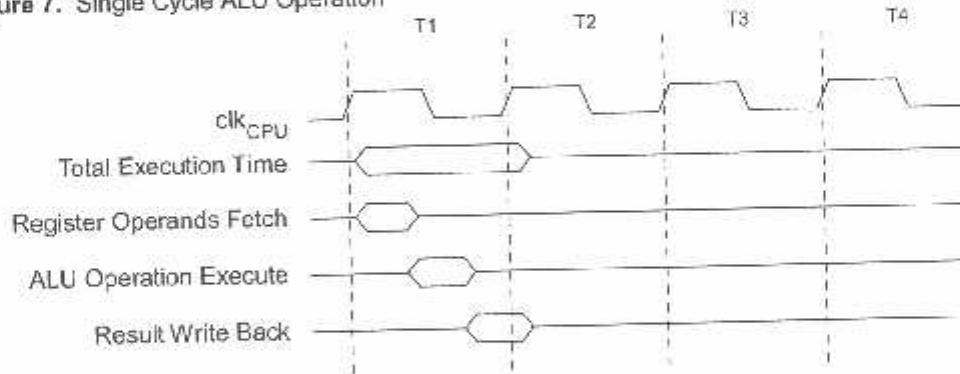


Figure 7 shows the internal timing concept for the Register File. In a single clock cycle an ALU operation using two register operands is executed, and the result is stored back to the destination register.

Figure 7. Single Cycle ALU Operation



Reset and Interrupt Handling

The AVR provides several different interrupt sources. These interrupts and the separate reset vector each have a separate program vector in the program memory space. All interrupts are assigned individual enable bits which must be written logic one together with the Global Interrupt Enable bit in the Status Register in order to enable the interrupt. Depending on the Program Counter value, interrupts may be automatically disabled when Boot Lock bits BLB02 or BLB12 are programmed. This feature improves software security. See the section "Memory Programming" on page 259 for details.

The lowest addresses in the program memory space are by default defined as the Reset and Interrupt Vectors. The complete list of vectors is shown in "Interrupts" on page 45. The list also determines the priority levels of the different interrupts. The lower the address the higher is the priority level. RESET has the highest priority, and next is INT0 – the External Interrupt Request

0. The Interrupt Vectors can be moved to the start of the Boot Flash section by setting the IVSEL bit in the General Interrupt Control Register (GICR). Refer to "Interrupts" on page 45 for more information. The Reset Vector can also be moved to the start of the boot Flash section by programming the BOOTRST Fuse, see "Boot Loader Support – Read-Write-Write Self-Programming" on page 246.

When an interrupt occurs, the Global Interrupt Enable I-bit is cleared and all interrupts are disabled. The user software can write logic one to the I-bit to enable nested interrupts. All enabled interrupts can then interrupt the current interrupt routine. The I-bit is automatically set when a Return from Interrupt instruction – RETI – is executed.

There are basically two types of interrupts. The first type is triggered by an event that sets the Interrupt Flag. For these interrupts, the Program Counter is vectored to the actual Interrupt Vector in order to execute the interrupt handling routine, and hardware clears the corresponding Interrupt Flag. Interrupt Flags can also be cleared by writing a logic one to the flag bit position(s) to be cleared. If an interrupt condition occurs while the corresponding interrupt enable bit is cleared, the Interrupt Flag will be set and remembered until the interrupt is enabled, or the flag is cleared by software. Similarly, if one or more interrupt conditions occur while the Global Interrupt Enable bit is cleared, the corresponding Interrupt Flag(s) will be set and remembered until the global interrupt enable bit is set, and will then be executed by order of priority.

The second type of interrupts will trigger as long as the interrupt condition is present. These interrupts do not necessarily have Interrupt Flags. If the interrupt condition disappears before the interrupt is enabled, the interrupt will not be triggered.

When the AVR exits from an interrupt, it will always return to the main program and execute one more instruction before any pending interrupt is served.

Note that the Status Register is not automatically stored when entering an interrupt routine, nor restored when returning from an interrupt routine. This must be handled by software.

When using the CLI instruction to disable interrupts, the interrupts will be immediately disabled. No interrupt will be executed after the CLI instruction, even if it occurs simultaneously with the CLI instruction. The following example shows how this can be used to avoid interrupts during the timed EEPROM write sequence.

Assembly Code Example
<pre> in r16, SREG ; store SREG value cli ; disable interrupts during timed sequence sbi EECR, EEWE ; start EEPROM write sbi EECR, EEWE out SREG, r16 ; restore SREG value (I-bit) </pre>
C Code Example
<pre> char *SREG; *SREG = SREG; /* store SREG value */ /* disable interrupts during timed sequence */ _cli(); EECR = (1<<EEWE); /* start EEPROM write */ EECR = (1<<EEWE); SREG = *SREG; /* restore SREG value (I-bit) */ </pre>

When using the SEI instruction to enable interrupts, the instruction following SEI will be executed before any pending interrupts, as shown in this example.

Assembly Code Example

```
sei ; set global interrupt enable
sleep ; enter sleep, waiting for interrupt
; note: will enter sleep before any pending
; interrupt(s)
```

C Code Example

```
_SEI(); /* set global interrupt enable */
_SLEEP(); /* enter sleep, waiting for interrupt */
/* note: will enter sleep before any pending interrupt(s) */
```

Interrupt Response Time

The interrupt execution response for all the enabled AVR interrupts is four clock cycles minimum. After four clock cycles the program vector address for the actual interrupt handling routine is executed. During this four clock cycle period, the Program Counter is pushed onto the Stack. The vector is normally a jump to the interrupt routine, and this jump takes three clock cycles. If an interrupt occurs during execution of a multi-cycle instruction, this instruction is completed before the interrupt is served. If an interrupt occurs when the MCU is in sleep mode, the interrupt execution response time is increased by four clock cycles. This increase comes in addition to the start-up time from the selected sleep mode.

A return from an interrupt handling routine takes four clock cycles. During these four clock cycles, the Program Counter (two bytes) is popped back from the Stack, the Stack Pointer is incremented by two, and the I-bit in SREG is set.



AVR ATmega16 Memories

This section describes the different memories in the ATmega16. The AVR architecture has two main memory spaces, the Data Memory and the Program Memory space. In addition, the ATmega16 features an EEPROM Memory for data storage. All three memory spaces are linear and regular.

In-System Reprogrammable Flash Program Memory

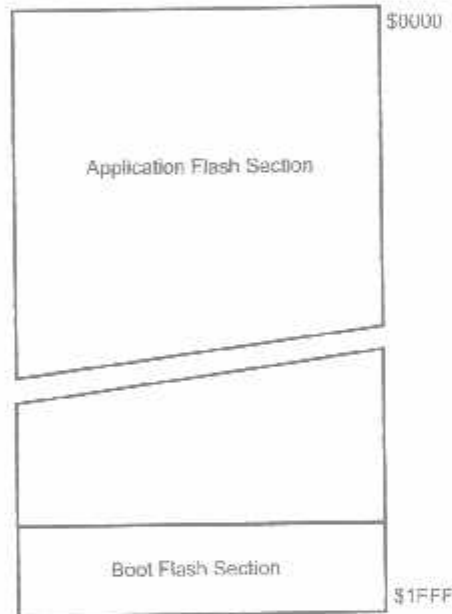
The ATmega16 contains 16 Kbytes On-chip In-System Reprogrammable Flash memory for program storage. Since all AVR instructions are 16 or 32 bits wide, the Flash is organized as 8K × 16. For software security, the Flash Program memory space is divided into two sections, Boot Program section and Application Program section.

The Flash memory has an endurance of at least 10,000 write/erase cycles. The ATmega16 Program Counter (PC) is 13 bits wide, thus addressing the 8K program memory locations. The operation of Boot Program section and associated Boot Lock bits for software protection are described in detail in "Boot Loader Support – Read-While-Write Self-Programming" on page 245. "Memory Programming" on page 259 contains a detailed description on Flash data serial downloading using the SPI pins or the JTAG interface.

Constant tables can be allocated within the entire program memory address space (see the LPM – Load Program Memory Instruction Description).

Timing diagrams for instruction fetch and execution are presented in "Instruction Execution Timing" on page 13.

Figure 8. Program Memory Map



SRAM Data Memory

Figure 9 shows how the ATmega16 SRAM Memory is organized.

The lower 1120 Data Memory locations address the Register File, the I/O Memory, and the internal data SRAM. The first 96 locations address the Register File and I/O Memory, and the next 1024 locations address the internal data SRAM.

The five different addressing modes for the data memory cover: Direct, Indirect with Displacement, Indirect, Indirect with Pre-decrement, and Indirect with Post-increment. In the Register File, registers R26 to R31 feature the indirect addressing pointer registers.

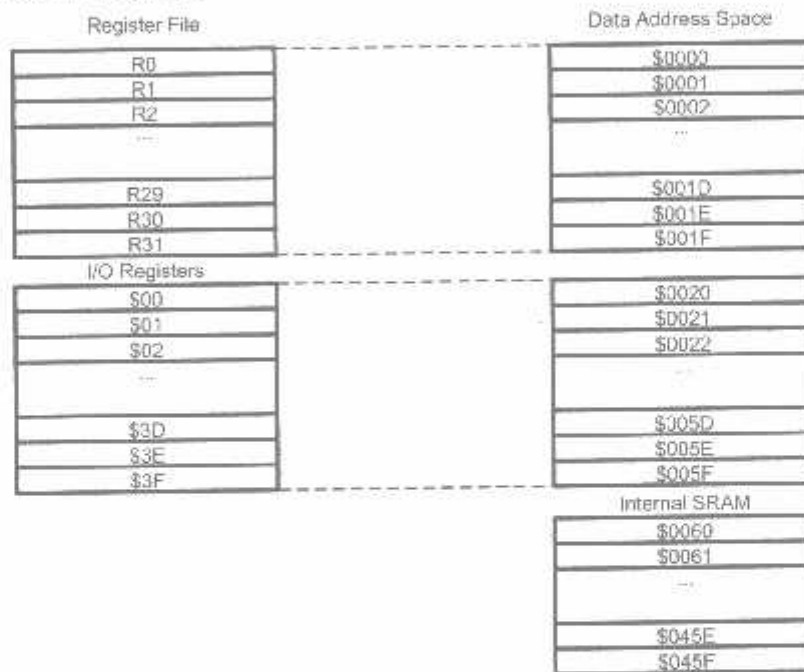
The direct addressing reaches the entire data space.

The Indirect with Displacement mode reaches 63 address locations from the base address given by the Y-register or Z-register.

When using register indirect addressing modes with automatic pre-decrement and post-increment, the address registers X, Y, and Z are decremented or incremented.

The 32 general purpose working registers, 64 I/O Registers, and the 1024 bytes of internal data SRAM in the ATmega16 are all accessible through all these addressing modes. The Register File is described in "General Purpose Register File" on page 11.

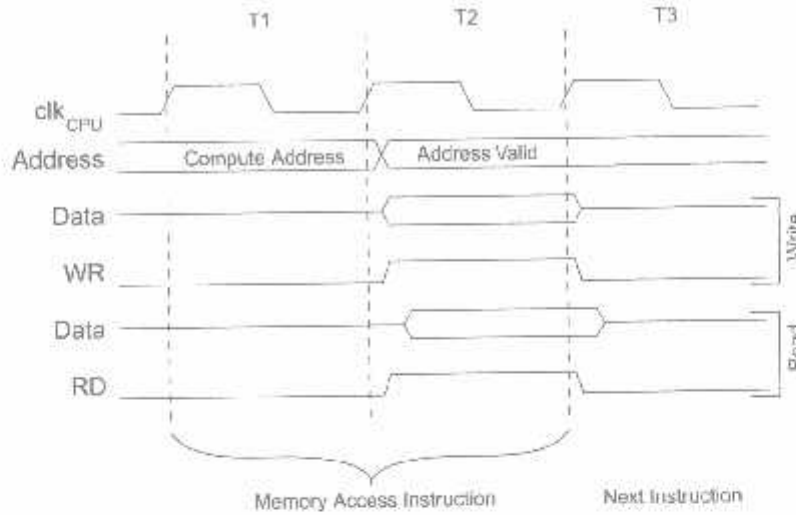
Figure 9. Data Memory Map



Data Memory Access Times

This section describes the general access timing concepts for internal memory access. The internal data SRAM access is performed in two clk_{CPU} cycles as described in Figure 10.

Figure 10. On-chip Data SRAM Access Cycles



EEPROM Data Memory

The ATmega16 contains 512 bytes of data EEPROM memory. It is organized as a separate data space, in which single bytes can be read and written. The EEPROM has an endurance of at least 100,000 write/erase cycles. The access between the EEPROM and the CPU is described in the following, specifying the EEPROM Address Registers, the EEPROM Data Register, and the EEPROM Control Register.

For a detailed description of SPI, JTAG, and Parallel data downloading to the EEPROM, see page 273, page 278, and page 262, respectively.

EEPROM Read/Write Access

The EEPROM Access Registers are accessible in the I/O space.

The write access time for the EEPROM is given in Table 1. A self-timing function, however, lets the user software detect when the next byte can be written. If the user code contains instructions that write the EEPROM, some precautions must be taken. In heavily filtered power supplies, V_{CC} is likely to rise or fall slowly on Power-up/down. This causes the device for some period of time to run at a voltage lower than specified as minimum for the clock frequency used. See "Preventing EEPROM Corruption" on page 22 for details on how to avoid problems in these situations.

In order to prevent unintentional EEPROM writes, a specific write procedure must be followed. Refer to the description of the EEPROM Control Register for details on this.

When the EEPROM is read, the CPU is halted for four clock cycles before the next instruction is executed. When the EEPROM is written, the CPU is halted for two clock cycles before the next instruction is executed.

The EEPROM Address Register – EEARH and EEARL

Bit	15	14	13	12	11	10	9	8	
	-	-	-	-	-	-	-	EEAR8	EEARH
	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
Read/Write	R	R	R	R	R	R	R	R/W	
Initial Value	0	0	0	0	0	0	0	X	
	X	X	X	X	X	X	X	X	

- **Bits 15..9 – Res: Reserved Bits**

These bits are reserved bits in the ATmega16 and will always read as zero.

- **Bits 8..0 – EEAR8..0: EEPROM Address**

The EEPROM Address Registers – EEARH and EEARL – specify the EEPROM address in the 512 bytes EEPROM space. The EEPROM data bytes are addressed linearly between 0 and 511. The initial value of EEAR is undefined. A proper value must be written before the EEPROM may be accessed.

The EEPROM Data Register – EEDR

Bit	7	6	5	4	3	2	1	0	
	MSB							LSB	EEDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits 7..0 – EEDR7..0: EEPROM Data**

For the EEPROM write operation, the EEDR Register contains the data to be written to the EEPROM in the address given by the EEAR Register. For the EEPROM read operation, the EEDR contains the data read out from the EEPROM at the address given by EEAR.

The EEPROM Control Register – EECR

Bit	7	6	5	4	3	2	1	0	
	-	-	-	-	EERIE	EEMWE	EEWE	EERE	EECR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	X	0	

- **Bits 7..4 – Res: Reserved Bits**

These bits are reserved bits in the ATmega16 and will always read as zero.

- **Bit 3 – EERIE: EEPROM Ready Interrupt Enable**

Writing EERIE to one enables the EEPROM Ready Interrupt if the I bit in SREG is set. Writing EERIE to zero disables the interrupt. The EEPROM Ready interrupt generates a constant interrupt when EEWE is cleared.

- **Bit 2 – EEMWE: EEPROM Master Write Enable**

The EEMWE bit determines whether setting EEWE to one causes the EEPROM to be written. When EEMWE is set, setting EEWE within four clock cycles will write data to the EEPROM at the selected address. If EEMWE is zero, setting EEWE will have no effect. When EEMWE has been written to one by software, hardware clears the bit to zero after four clock cycles. See the description of the EEWE bit for an EEPROM write procedure.



• Bit 1 – EEW: EEPROM Write Enable

The EEPROM Write Enable Signal EEW is the write strobe to the EEPROM. When address and data are correctly set up, the EEW bit must be written to one to write the value into the EEPROM. The EEMWE bit must be written to one before a logical one is written to EEW, otherwise no EEPROM write takes place. The following procedure should be followed when writing the EEPROM (the order of steps 3 and 4 is not essential):

1. Wait until EEW becomes zero.
2. Wait until SPEN in SPMCR becomes zero.
3. Write new EEPROM address to EEAR (optional).
4. Write new EEPROM data to EEDR (optional).
5. Write a logical one to the EEMWE bit while writing a zero to EEW in EECR.
6. Within four clock cycles after setting EEMWE, write a logical one to EEW.

The EEPROM can not be programmed during a CPU write to the Flash memory. The software must check that the Flash programming is completed before initiating a new EEPROM write. Step 2 is only relevant if the software contains a Boot Loader allowing the CPU to program the Flash. If the Flash is never being updated by the CPU, step 2 can be omitted. See "Boot Loader Support – Read-While-Write Self-Programming" on page 246 for details about boot programming.

Caution: An interrupt between step 5 and step 6 will make the write cycle fail, since the EEPROM Master Write Enable will time-out. If an interrupt routine accessing the EEPROM is interrupting another EEPROM Access, the EEAR or EEDR register will be modified, causing the interrupted EEPROM Access to fail. It is recommended to have the Global Interrupt Flag cleared during all the steps to avoid these problems.

When the write access time has elapsed, the EEW bit is cleared by hardware. The user software can poll this bit and wait for a zero before writing the next byte. When EEW has been set, the CPU is halted for two cycles before the next instruction is executed.

• Bit 0 – EERE: EEPROM Read Enable

The EEPROM Read Enable Signal – EERE – is the read strobe to the EEPROM. When the correct address is set up in the EEAR Register, the EERE bit must be written to a logic one to trigger the EEPROM read. The EEPROM read access takes one instruction, and the requested data is available immediately. When the EEPROM is read, the CPU is halted for four cycles before the next instruction is executed.

The user should poll the EEW bit before starting the read operation. If a write operation is in progress, it is neither possible to read the EEPROM, nor to change the EEAR Register.

The calibrated Oscillator is used to time the EEPROM accesses. Table 1 lists the typical programming time for EEPROM access from the CPU.

Table 1. EEPROM Programming Time

Symbol	Number of Calibrated RC Oscillator Cycles ⁽¹⁾	Typ Programming Time
EEPROM write (from CPU)	8448	8.5 ms

Note: 1. Uses 1 MHz clock, independent of CKSEL Fuse setting.

The following code examples show one assembly and one C function for writing to the EEPROM. The examples assume that interrupts are controlled (for example by disabling interrupts globally) so that no interrupts will occur during execution of these functions. The examples



also assume that no Flash Boot Loader is present in the software. If such code is present, the EEPROM write function must also wait for any ongoing SPM command to finish.

Assembly Code Example

```
EEPROM_write:
    ; Wait for completion of previous write
    sbic EECR, EEWE
    rjmp EEPROM_write
    ; Set up address (r18:r17) in address register
    out EEARH, r18
    out EEARL, r17
    ; Write data (r16) to data register
    out EEDR, r16
    ; Write logical one to EEMWE
    sbi EECR, EEMWE
    ; Start eeprom write by setting EEWE
    sbi EECR, EEWE
    ret
```

C Code Example

```
void EEPROM_write(unsigned int uiAddress, unsigned char ucData)
{
    /* Wait for completion of previous write */
    while(EECR & (1<<EEWE))
        ;
    /* Set up address and data registers */
    EEAR = uiAddress;
    EEDR = ucData;
    /* Write logical one to EEMWE */
    EECR |= (1<<EEMWE);
    /* Start eeprom write by setting EEWE */
    EECR |= (1<<EEWE);
}
```



The next code examples show assembly and C functions for reading the EEPROM. The examples assume that interrupts are controlled so that no interrupts will occur during execution of these functions.

Assembly Code Example

```
EEPROM_read:
    ; Wait for completion of previous write
    sbic EECR, EEWE
    rjmp EEPROM_read
    ; Set up address (r16:r17) in address register
    out EARL, r16
    out EARH, r17
    ; Start eeprom read by writing EERE
    sbi EECR, EERE
    ; Read data from data register
    in r16, EEDR
    ret
```

C Code Example

```
unsigned char EEPROM_read(unsigned int uiAddress)
{
    /* Wait for completion of previous write */
    while(EECR & (1<<EEWE))
        ;
    /* Set up address register */
    EAR = uiAddress;
    /* Start eeprom read by writing EERE */
    EECR |= (1<<EERE);
    /* Return data from data register */
    return EEDR;
}
```

EEPROM Write During Power-down Sleep Mode

When entering Power-down Sleep mode while an EEPROM write operation is active, the EEPROM write operation will continue, and will complete before the Write Access time has passed. However, when the write operation is completed, the Oscillator continues running, and as a consequence, the device does not enter Power-down entirely. It is therefore recommended to verify that the EEPROM write operation is completed before entering Power-down.

Preventing EEPROM Corruption

During periods of low V_{CC} , the EEPROM data can be corrupted because the supply voltage is too low for the CPU and the EEPROM to operate properly. These issues are the same as for board level systems using EEPROM, and the same design solutions should be applied.

An EEPROM data corruption can be caused by two situations when the voltage is too low. First, a regular write sequence to the EEPROM requires a minimum voltage to operate correctly. Secondly, the CPU itself can execute instructions incorrectly, if the supply voltage is too low.



EEPROM data corruption can easily be avoided by following this design recommendation:

Keep the AVR RESET active (low) during periods of insufficient power supply voltage. This can be done by enabling the internal Brown-out Detector (BOD). If the detection level of the internal BOD does not match the needed detection level, an external low V_{CC} Reset Protection circuit can be used. If a reset occurs while a write operation is in progress, the write operation will be completed provided that the power supply voltage is sufficient.

I/O Memory

The I/O space definition of the ATmega16 is shown in "Register Summary" on page 331.

All ATmega16 I/Os and peripherals are placed in the I/O space. The I/O locations are accessed by the IN and OUT instructions, transferring data between the 32 general purpose working registers and the I/O space. I/O Registers within the address range \$00 - \$1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions. Refer to the Instruction Set section for more details. When using the I/O specific commands IN and OUT, the I/O addresses \$00 - \$3F must be used. When addressing I/O Registers as data space using LD and ST instructions, \$20 must be added to these addresses.

For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.

Some of the Status Flags are cleared by writing a logical one to them. Note that the CBI and SBI instructions will operate on all bits in the I/O Register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers \$00 to \$1F only.

The I/O and Peripherals Control Registers are explained in later sections.

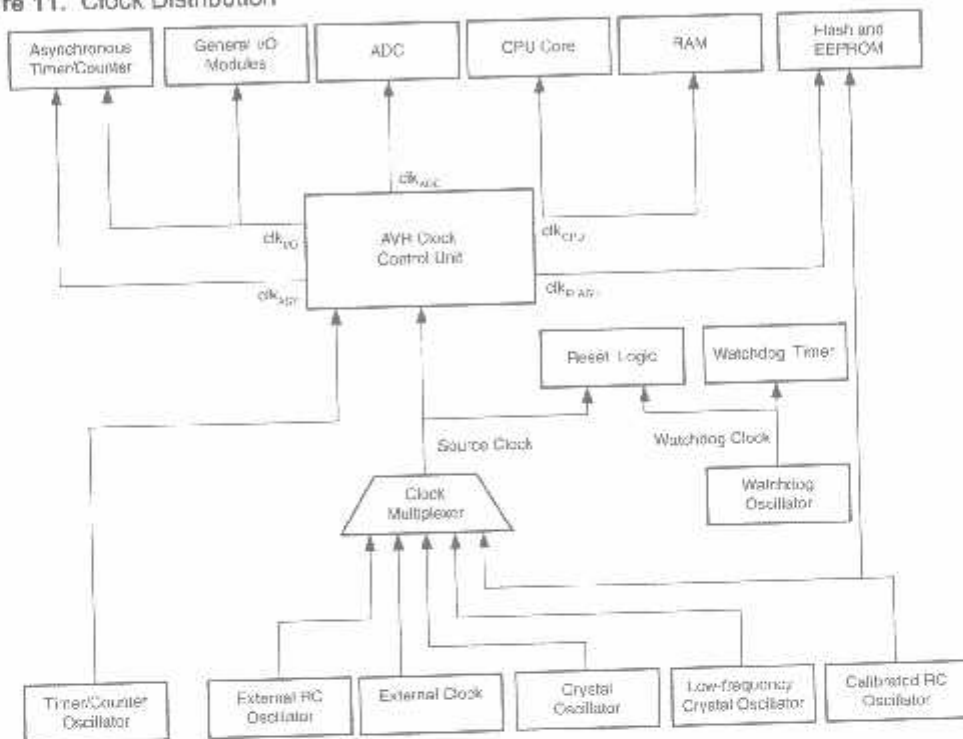


System Clock and Clock Options

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.



ADC Clock – clk_{ADC}

The ADC is provided with a dedicated clock domain. This allows halting the CPU and I/O clocks in order to reduce noise generated by digital circuitry. This gives more accurate ADC conversion results.

Clock Sources

The device has the following clock source options, selectable by Flash Fuse bits as shown below. The clock from the selected source is input to the AVR clock generator, and routed to the appropriate modules.

Table 2. Device Clocking Options Select⁽¹⁾

Device Clocking Option	CKSEL3..0
External Crystal/Ceramic Resonator	1111 - 1010
External Low-frequency Crystal	1001
External RC Oscillator	1000 - 0101
Calibrated Internal RC Oscillator	0100 - 0001
External Clock	0000

Note: 1. For all fuses "1" means unprogrammed while "0" means programmed.

The various choices for each clocking option is given in the following sections. When the CPU wakes up from Power-down or Power-save, the selected clock source is used to time the start-up, ensuring stable Oscillator operation before instruction execution starts. When the CPU starts from Reset, there is as an additional delay allowing the power to reach a stable level before commencing normal operation. The Watchdog Oscillator is used for timing this real-time part of the start-up time. The number of WDT Oscillator cycles used for each time-out is shown in Table 3. The frequency of the Watchdog Oscillator is voltage dependent as shown in "ATmega16 Typical Characteristics" on page 299.

Table 3. Number of Watchdog Oscillator Cycles

Typ Time-out ($V_{CC} = 5.0V$)	Typ Time-out ($V_{CC} = 3.0V$)	Number of Cycles
4.1 ms	4.3 ms	4K (4,096)
65 ms	69 ms	64K (65,536)

Default Clock Source

The device is shipped with CKSEL = "0001" and SUT = "10". The default clock source setting is therefore the 1 MHz Internal RC Oscillator with longest startup time. This default setting ensures that all users can make their desired clock source setting using an In-System or Parallel Programmer.

Crystal Oscillator

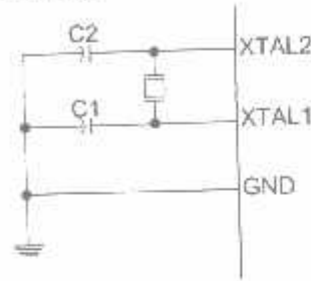
XTAL1 and XTAL2 are input and output, respectively, of an inverting amplifier which can be configured for use as an On-chip Oscillator, as shown in Figure 12. Either a quartz crystal or a ceramic resonator may be used. The CKOPT Fuse selects between two different Oscillator amplifier modes. When CKOPT is programmed, the Oscillator output will oscillate with a full rail-to-rail swing on the output. This mode is suitable when operating in a very noisy environment or when the output from XTAL2 drives a second clock buffer. This mode has a wide frequency range. When CKOPT is unprogrammed, the Oscillator has a smaller output swing. This reduces power consumption considerably. This mode has a limited frequency range and it can not be used to drive other clock buffers.

For resonators, the maximum frequency is 8 MHz with CKOPT unprogrammed and 16 MHz with CKOPT programmed. C1 and C2 should always be equal for both crystals and resonators. The optimal value of the capacitors depends on the crystal or resonator in use, the amount of stray capacitance, and the electromagnetic noise of the environment. Some initial guidelines for



choosing capacitors for use with crystals are given in Table 4. For ceramic resonators, the capacitor values given by the manufacturer should be used.

Figure 12. Crystal Oscillator Connections



The Oscillator can operate in three different modes, each optimized for a specific frequency range. The operating mode is selected by the fuses CKSEL3..1 as shown in Table 4.

Table 4. Crystal Oscillator Operating Modes

CKOPT	CKSEL3..1	Frequency Range (MHz)	Recommended Range for Capacitors C1 and C2 for Use with Crystals (pF)
1	101 ¹	0.4 - 0.9	-
1	110	0.9 - 3.0	12 - 22
1	111	3.0 - 8.0	12 - 22
0	101, 110, 111	1.0 ≤	12 - 22

Note: 1. This option should not be used with crystals, only with ceramic resonators.

The CKSEL0 Fuse together with the SUT1..0 Fuses select the start-up times as shown in Table 5.

Table 5. Start-up Times for the Crystal Oscillator Clock Selection

CKSEL0	SUT1..0	Start-up Time from Power-down and Power-save	Additional Delay from Reset ($V_{CC} = 5.0V$)	Recommended Usage
0	00	258 CK ¹	4.1 ms	Ceramic resonator, fast rising power
0	01	258 CK ¹	65 ms	Ceramic resonator, slowly rising power
0	10	1K CK ²	-	Ceramic resonator, BOD enabled
0	11	1K CK ²	4.1 ms	Ceramic resonator, fast rising power
1	00	1K CK ²	65 ms	Ceramic resonator, slowly rising power
1	01	16K CK	-	Crystal Oscillator, BOD enabled
1	10	16K CK	4.1 ms	Crystal Oscillator, fast rising power
1	11	16K CK	65 ms	Crystal Oscillator, slowly rising power

- Notes:
1. These options should only be used when not operating close to the maximum frequency of the device, and only if frequency stability at start-up is not important for the application. These options are not suitable for crystals.
 2. These options are intended for use with ceramic resonators and will ensure frequency stability at start-up. They can also be used with crystals when not operating close to the maximum frequency of the device, and if frequency stability at start-up is not important for the application.

Low-frequency Crystal Oscillator

To use a 32.768 kHz watch crystal as the clock source for the device, the Low-frequency Crystal Oscillator must be selected by setting the CKSEL Fuses to "1001". The crystal should be connected as shown in Figure 12. By programming the CKOPT Fuse, the user can enable internal capacitors on XTAL1 and XTAL2, thereby removing the need for external capacitors. The internal capacitors have a nominal value of 36 pF.

When this Oscillator is selected, start-up times are determined by the SUT Fuses as shown in Table 6.

Table 6. Start-up Times for the Low-frequency Crystal Oscillator Clock Selection

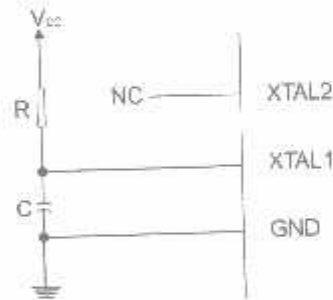
SUT1..0	Start-up Time from Power-down and Power-save	Additional Delay from Reset (V _{CC} = 5.0V)	Recommended Usage
00	1K CK ¹⁾	4.1 ms	Fast rising power or BOD enabled
01	1K CK ¹⁾	65 ms	Slowly rising power
10	32K CK	65 ms	Stable frequency at start-up
11	Reserved		

Note: 1. These options should only be used if frequency stability at start-up is not important for the application.

External RC Oscillator

For timing insensitive applications, the external RC configuration shown in Figure 13 can be used. The frequency is roughly estimated by the equation $f = 1/(3RC)$. C should be at least 22 pF. By programming the CKOPT Fuse, the user can enable an internal 36 pF capacitor between XTAL1 and GND, thereby removing the need for an external capacitor. For more information on Oscillator operation and details on how to choose R and C, refer to the External RC Oscillator application note.

Figure 13. External RC Configuration



The Oscillator can operate in four different modes, each optimized for a specific frequency range. The operating mode is selected by the fuses CKSEL3..0 as shown in Table 7.

Table 7. External RC Oscillator Operating Modes

CKSEL3..0	Frequency Range (MHz)
0101	0.1 - 0.9
0110	0.9 - 3.0
0111	3.0 - 8.0
1000	8.0 - 12.0

When this Oscillator is selected, start-up times are determined by the SUT Fuses as shown in Table 8.

Table 8. Start-up Times for the External RC Oscillator Clock Selection

SUT1..0	Start-up Time from Power-down and Power-save	Additional Delay from Reset ($V_{CC} = 5.0V$)	Recommended Usage
00	18 CK	–	BOD enabled
01	18 CK	4.1 ms	Fast rising power
10	18 CK	65 ms	Slowly rising power
11	6 CK ⁽¹⁾	4.1 ms	Fast rising power or BOD enabled

Note: 1. This option should not be used when operating close to the maximum frequency of the device.

Calibrated Internal RC Oscillator

The Calibrated Internal RC Oscillator provides a fixed 1.0 MHz, 2.0 MHz, 4.0 MHz, or 8.0 MHz clock. All frequencies are nominal values at 5V and 25°C. This clock may be selected as the system-clock by programming the CKSEL Fuses as shown in Table 9. If selected, it will operate with no external components. The CKOPT Fuse should always be unprogrammed when using this clock option. During Reset, hardware loads the calibration byte into the OSCCAL Register and thereby automatically calibrates the RC Oscillator. At 5V, 25°C and 1.0 MHz, 2.0 MHz, 4.0 MHz or 8.0 MHz Oscillator frequency selected, this calibration gives a frequency within $\pm 3\%$ of the nominal frequency. Using calibration methods as described in application notes available at www.atmel.com/avr it is possible to achieve $\pm 1\%$ accuracy at any given V_{CC} and Temperature. When this Oscillator is used as the Chip Clock, the Watchdog Oscillator will still be used for the Watchdog Timer and for the reset time-out. For more information on the pre-programmed calibration value, see the section "Calibration Byte" on page 261.

Table 9. Internal Calibrated RC Oscillator Operating Modes

CKSEL3..0	Nominal Frequency (MHz)
0001 ⁽¹⁾	1.0
0010	2.0
0011	4.0
0100	8.0

Note: 1. The device is shipped with this option selected.

When this Oscillator is selected, start-up times are determined by the SUT Fuses as shown in Table 10. XTAL1 and XTAL2 should be left unconnected (NC).

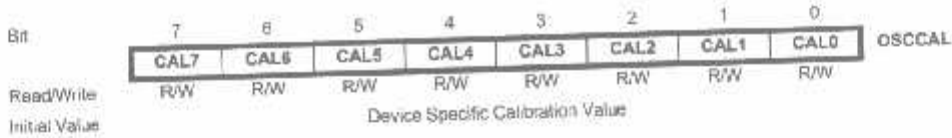
Table 10. Start-up Times for the Internal Calibrated RC Oscillator Clock Selection

SUT1..0	Start-up Time from Power-down and Power-save	Additional Delay from Reset ($V_{CC} = 5.0V$)	Recommended Usage
00	6 CK	–	BOD enabled
01	6 CK	4.1 ms	Fast rising power
10 ⁽¹⁾	6 CK	65 ms	Slowly rising power
11		Reserved	

Note: 1. The device is shipped with this option selected.



Oscillator Calibration Register – OSCCAL



• Bits 7..0 – CAL7..0: Oscillator Calibration Value

Writing the calibration byte to this address will trim the Internal Oscillator to remove process variations from the Oscillator frequency. This is done automatically during Chip Reset. When OSCCAL is zero, the lowest available frequency is chosen. Writing non-zero values to this register will increase the frequency of the Internal Oscillator. Writing \$FF to the register gives the highest available frequency. The calibrated Oscillator is used to time EEPROM and Flash access. If EEPROM or Flash is written, do not calibrate to more than 10% above the nominal frequency. Otherwise, the EEPROM or Flash write may fail. Note that the Oscillator is intended for calibration to 1.0 MHz, 2.0 MHz, 4.0 MHz, or 8.0 MHz. Tuning to other values is not guaranteed, as indicated in Table 11.

Table 11. Internal RC Oscillator Frequency Range.

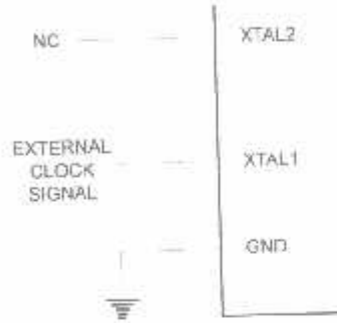
OSCCAL Value	Min Frequency in Percentage of Nominal Frequency (%)	Max Frequency in Percentage of Nominal Frequency (%)
\$00	50	100
\$7F	75	150
\$FF	100	200



External Clock

To drive the device from an external clock source, XTAL1 should be driven as shown in Figure 14. To run the device on an external clock, the CKSEL Fuses must be programmed to "0000". By programming the CKOPT Fuse, the user can enable an internal 36 pF capacitor between XTAL1 and GND.

Figure 14. External Clock Drive Configuration



When this clock source is selected, start-up times are determined by the SUT Fuses as shown in Table 12.

Table 12. Start-up Times for the External Clock Selection

SUT1..0	Start-up Time from Power-down and Power-save	Additional Delay from Reset ($V_{CC} = 5.0V$)	Recommended Usage
00	6 CK	—	BOD enabled
01	6 CK	4.1 ms	Fast rising power
10	6 CK	65 ms	Slowly rising power
11	Reserved		

When applying an external clock, it is required to avoid sudden changes in the applied clock frequency to ensure stable operation of the MCU. A variation in frequency of more than 2% from one clock cycle to the next can lead to unpredictable behavior. It is required to ensure that the MCU is kept in reset during such changes in the clock frequency.

Timer/Counter Oscillator

For AVR microcontrollers with Timer/Counter Oscillator pins (TOSC1 and TOSC2), the crystal is connected directly between the pins. No external capacitors are needed. The Oscillator is optimized for use with a 32.768 kHz watch crystal. Applying an external clock source to TOSC1 is not recommended.

Note: The Timer/Counter Oscillator uses the same type of crystal oscillator as Low-Frequency Oscillator and the internal capacitors have the same nominal value of 36 pF.



Power Management and Sleep Modes

Sleep modes enable the application to shut down unused modules in the MCU, thereby saving power. The AVR provides various sleep modes allowing the user to tailor the power consumption to the application's requirements.

To enter any of the six sleep modes, the SE bit in MCUCR must be written to logic one and a SLEEP instruction must be executed. The SM2, SM1, and SM0 bits in the MCUCR Register select which sleep mode (Idle, ADC Noise Reduction, Power-down, Power-save, Standby, or Extended Standby) will be activated by the SLEEP instruction. See Table 13 for a summary. If an enabled interrupt occurs while the MCU is in a sleep mode, the MCU wakes up. The MCU is then halted for four cycles in addition to the start-up time, it executes the interrupt routine, and resumes execution from the instruction following SLEEP. The contents of the Register File and SRAM are unaltered when the device wakes up from sleep. If a Reset occurs during sleep mode, the MCU wakes up and executes from the Reset Vector.

Figure 11 on page 24 presents the different clock systems in the ATmega16, and their distribution. The figure is helpful in selecting an appropriate sleep mode.

MCU Control Register – MCUCR

The MCU Control Register contains control bits for power management.

Bit	7	6	5	4	3	2	1	0	
	SM2	SE	SM1	SM0	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

• Bits 7, 5, 4 – SM2..0: Sleep Mode Select Bits 2, 1, and 0

These bits select between the six available sleep modes as shown in Table 13.

Table 13. Sleep Mode Select

SM2	SM1	SM0	Sleep Mode
0	0	0	Idle
0	0	1	ADC Noise Reduction
0	1	0	Power-down
0	1	1	Power-save
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Standby ¹
1	1	1	Extended Standby ¹

Note: 1. Standby mode and Extended Standby mode are only available with external crystals or resonators.

• Bit 6 – SE: Sleep Enable

The SE bit must be written to logic one to make the MCU enter the sleep mode when the SLEEP instruction is executed. To avoid the MCU entering the sleep mode unless it is the programmers purpose, it is recommended to write the Sleep Enable (SE) bit to one just before the execution of the SLEEP instruction and to clear it immediately after waking up.



Idle Mode

When the SM2..0 bits are written to 000, the SLEEP instruction makes the MCU enter Idle mode, stopping the CPU but allowing SPI, USART, Analog Comparator, ADC, Two-wire Serial Interface, Timer/Counters, Watchdog, and the interrupt system to continue operating. This sleep mode basically halts clk_{CPU} and clk_{FLASH} , while allowing the other clocks to run.

Idle mode enables the MCU to wake up from external triggered interrupts as well as internal ones like the Timer Overflow and USART Transmit Complete interrupts. If wake-up from the Analog Comparator interrupt is not required, the Analog Comparator can be powered down by setting the ACD bit in the Analog Comparator Control and Status Register – ACSR. This will reduce power consumption in Idle mode. If the ADC is enabled, a conversion starts automatically when this mode is entered.

ADC Noise Reduction Mode

When the SM2..0 bits are written to 001, the SLEEP instruction makes the MCU enter ADC Noise Reduction mode, stopping the CPU but allowing the ADC, the External Interrupts, the Two-wire Serial Interface address watch, Timer/Counter2 and the Watchdog to continue operating (if enabled). This sleep mode basically halts clk_{IO} , clk_{CPU} , and clk_{FLASH} , while allowing the other clocks to run.

This improves the noise environment for the ADC, enabling higher resolution measurements. If the ADC is enabled, a conversion starts automatically when this mode is entered. Apart from the ADC Conversion Complete interrupt, only an External Reset, a Watchdog Reset, a Brown-out Reset, a Two-wire Serial Interface Address Match Interrupt, a Timer/Counter2 interrupt, an SPM/EEPROM ready interrupt, an External level interrupt on INT0 or INT1, or an external interrupt on INT2 can wake up the MCU from ADC Noise Reduction mode.

Power-down Mode

When the SM2..0 bits are written to 010, the SLEEP instruction makes the MCU enter Power-down mode. In this mode, the External Oscillator is stopped, while the External interrupts, the Two-wire Serial Interface address watch, and the Watchdog continue operating (if enabled). Only an External Reset, a Watchdog Reset, a Brown-out Reset, a Two-wire Serial Interface address match interrupt, an External level interrupt on INT0 or INT1, or an External interrupt on INT2 can wake up the MCU. This sleep mode basically halts all generated clocks, allowing operation of asynchronous modules only.

Note that if a level triggered interrupt is used for wake-up from Power-down mode, the changed level must be held for some time to wake up the MCU. Refer to "External Interrupts" on page 68 for details.

When waking up from Power-down mode, there is a delay from the wake-up condition occurs until the wake-up becomes effective. This allows the clock to restart and become stable after having been stopped. The wake-up period is defined by the same CKSEL Fuses that define the reset time-out period, as described in "Clock Sources" on page 25.

Power-save Mode

When the SM2..0 bits are written to 011, the SLEEP instruction makes the MCU enter Power-save mode. This mode is identical to Power-down, with one exception:

If Timer/Counter2 is clocked asynchronously, that is, the AS2 bit in ASSR is set, Timer/Counter2 will run during sleep. The device can wake up from either Timer Overflow or Output Compare event from Timer/Counter2 if the corresponding Timer/Counter2 interrupt enable bits are set in TIMSK, and the Global Interrupt Enable bit in SREG is set.

If the Asynchronous Timer is NOT clocked asynchronously, Power-down mode is recommended instead of Power-save mode because the contents of the registers in the Asynchronous Timer should be considered undefined after wake-up in Power-save mode if AS2 is 0.

This sleep mode basically halts all clocks except clk_{ASY} , allowing operation only of asynchronous modules, including Timer/Counter2 if clocked asynchronously.



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 _{FLASH}	clk _{IO}	clk _{ADC}	clk _{ASY}	Main Clock Source Enabled	Timer Osc. Enabled	INT2 INT1 INT0	TWI Address Match	Timer 2	SPM / EEPROM Ready	ADC	Other I/O
Idle			X	X	X	X	X ⁽¹⁾	X	X	X	X	X	X
ADC Noise Reduction				X	X	X	X ⁽²⁾	X ⁽³⁾	X	X	X	X	
Power Down								X ⁽³⁾	X				
Power Save					X ⁽²⁾		X ⁽²⁾	X ⁽³⁾	X	X ⁽²⁾			
Standby ⁽¹⁾						X		X ⁽³⁾	X				
Extended Standby ⁽²⁾					X ⁽²⁾	X	X ⁽²⁾	X ⁽³⁾	X	X ⁽²⁾			

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

Minimizing Power Consumption	There are several issues to consider when trying to minimize the power consumption in an AVR controlled system. In general, sleep modes should be used as much as possible, and the sleep mode should be selected so that as few as possible of the device's functions are operating. All functions not needed should be disabled. In particular, the following modules may need special consideration when trying to achieve the lowest possible power consumption.
Analog to Digital Converter	If enabled, the ADC will be enabled in all sleep modes. To save power, the ADC should be disabled before entering any sleep mode. When the ADC is turned off and on again, the next conversion will be an extended conversion. Refer to "Analog to Digital Converter" on page 204 for details on ADC operation.
Analog Comparator	When entering Idle mode, the Analog Comparator should be disabled if not used. When entering ADC Noise Reduction mode, the Analog Comparator should be disabled. In the other sleep modes, the Analog Comparator is automatically disabled. However, if the Analog Comparator is set up to use the Internal Voltage Reference as input, the Analog Comparator should be disabled in all sleep modes. Otherwise, the Internal Voltage Reference will be enabled, independent of sleep mode. Refer to "Analog Comparator" on page 201 for details on how to configure the Analog Comparator.
Brown-out Detector	If the Brown-out Detector is not needed in the application, this module should be turned off. If the Brown-out Detector is enabled by the BODEN Fuse, it will be enabled in all sleep modes, and hence, always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption. Refer to "Brown-out Detection" on page 40 for details on how to configure the Brown-out Detector.
Internal Voltage Reference	The Internal Voltage Reference will be enabled when needed by the Brown-out Detector, the Analog Comparator or the ADC. If these modules are disabled as described in the sections above, the internal voltage reference will be disabled and it will not be consuming power. When turned on again, the user must allow the reference to start up before the output is used. If the reference is kept on in sleep mode, the output can be used immediately. Refer to "Internal Voltage Reference" on page 42 for details on the start-up time.
Watchdog Timer	If the Watchdog Timer is not needed in the application, this module should be turned off. If the Watchdog Timer is enabled, it will be enabled in all sleep modes, and hence, always consume power. In the deeper sleep modes, this will contribute significantly to the total current consumption. Refer to "Watchdog Timer" on page 42 for details on how to configure the Watchdog Timer.
Port Pins	When entering a sleep mode, all port pins should be configured to use minimum power. The most important thing is then to ensure that no pins drive resistive loads. In sleep modes where the both the I/O clock (clk_{IO}) and the ADC clock (clk_{ADC}) are stopped, the input buffers of the device will be disabled. This ensures that no power is consumed by the input logic when not needed. In some cases, the input logic is needed for detecting wake-up conditions, and it will then be enabled. Refer to the section "Digital Input Enable and Sleep Modes" on page 54 for details on which pins are enabled. If the input buffer is enabled and the input signal is left floating or have an analog signal level close to $V_{CC}/2$, the input buffer will use excessive power.



**JTAG Interface and
On-chip Debug
System**

If the On-chip debug system is enabled by the OCDEN Fuse and the chip enter Power down or Power save sleep mode, the main clock source remains enabled. In these sleep modes, this will contribute significantly to the total current consumption. There are three alternative ways to avoid this:

- Disable OCDEN Fuse.
- Disable JTAGEN Fuse.
- Write one to the JTD bit in MCUCSR.

The TDO pin is left floating when the JTAG interface is enabled while the JTAG TAP controller is not shifting data. If the hardware connected to the TDO pin does not pull up the logic level, power consumption will increase. Note that the TDI pin for the next device in the scan chain contains a pull-up that avoids this problem. Writing the JTD bit in the MCUCSR register to one or leaving the JTAG fuse unprogrammed disables the JTAG interface.



System Control and Reset

Resetting the AVR

During Reset, all I/O Registers are set to their initial values, and the program starts execution from the Reset Vector. The instruction placed at the Reset Vector must be a JMP – absolute jump – instruction to the reset handling routine. If the program never enables an interrupt source, the Interrupt Vectors are not used, and regular program code can be placed at these locations. This is also the case if the Reset Vector is in the Application section while the Interrupt Vectors are in the Boot section or vice versa. The circuit diagram in Figure 15 shows the reset logic. Table 15 defines the electrical parameters of the reset circuitry.

The I/O ports of the AVR are immediately reset to their initial state when a reset source goes active. This does not require any clock source to be running.

After all reset sources have gone inactive, a delay counter is invoked, stretching the Internal Reset. This allows the power to reach a stable level before normal operation starts. The time-out period of the delay counter is defined by the user through the CKSEL Fuses. The different selections for the delay period are presented in "Clock Sources" on page 25.

Reset Sources

The ATmega16 has five sources of reset:

- **Power-on Reset.**
The MCU is reset when the supply voltage is below the Power-on Reset threshold (V_{POT}).
- **External Reset.**
The MCU is reset when a low level is present on the $\overline{\text{RESET}}$ pin for longer than the minimum pulse length.
- **Watchdog Reset.**
The MCU is reset when the Watchdog Timer period expires and the Watchdog is enabled.
- **Brown-out Reset.**
The MCU is reset when the supply voltage V_{CC} is below the Brown-out Reset threshold (V_{BOT}) and the Brown-out Detector is enabled.
- **JTAG AVR Reset.**
The MCU is reset as long as there is a logic one in the Reset Register, one of the scan chains of the JTAG system. Refer to the section "IEEE 1149.1 (JTAG) Boundary-scan" on page 228 for details.



Figure 15. Reset Logic

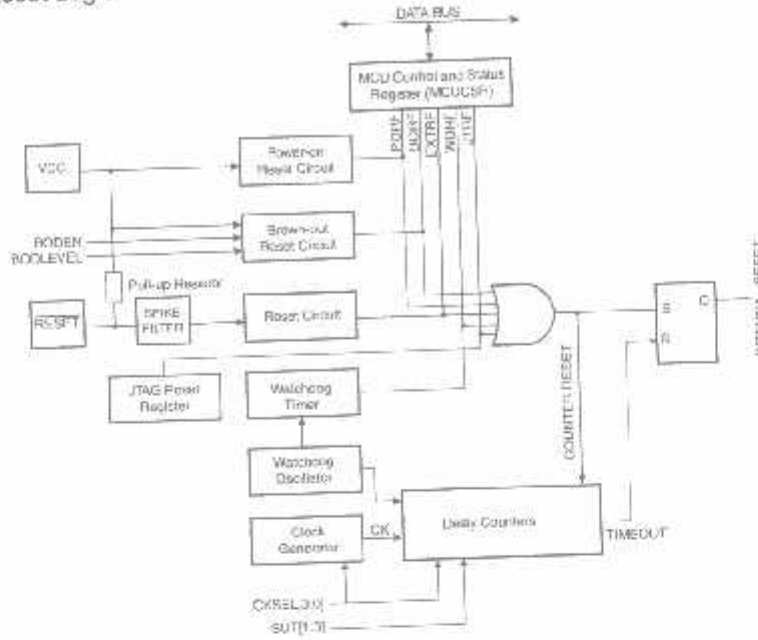


Table 15. Reset Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Units
V_{POT}	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_{BOT}	Brown-out Reset Threshold Voltage ²⁾	BODLEVEL = 1	2.5	2.7	3.2	V
		BODLEVEL = 0	3.6	4.0	4.5	
t_{BCD}	Minimum low voltage period for Brown-out Detection	BODLEVEL = 1		2		μs
		BODLEVEL = 0		2		
V_{HYST}	Brown-out Detector hysteresis			50		mV

- Notes:
1. The Power-on Reset will not work unless the supply voltage has been below V_{POT} (falling).
 2. V_{BOT} may be below nominal minimum operating voltage for some devices. For devices where this is the case, the device is tested down to $V_{CC} = V_{BOT}$ 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.



Power-on Reset

A Power-on Reset (POR) pulse is generated by an On-chip detection circuit. The detection level is defined in Table 15. The POR is activated whenever V_{CC} is below the detection level. The POR circuit can be used to trigger the Start-up Reset, as well as to detect a failure in supply voltage.

A Power-on Reset (POR) circuit ensures that the device is reset from Power-on. Reaching the Power-on Reset threshold voltage invokes the delay counter, which determines how long the device is kept in RESET after V_{CC} rise. The RESET signal is activated again, without any delay, when V_{CC} decreases below the detection level.

Figure 16. MCU Start-up, $\overline{\text{RESET}}$ Tied to V_{CC}

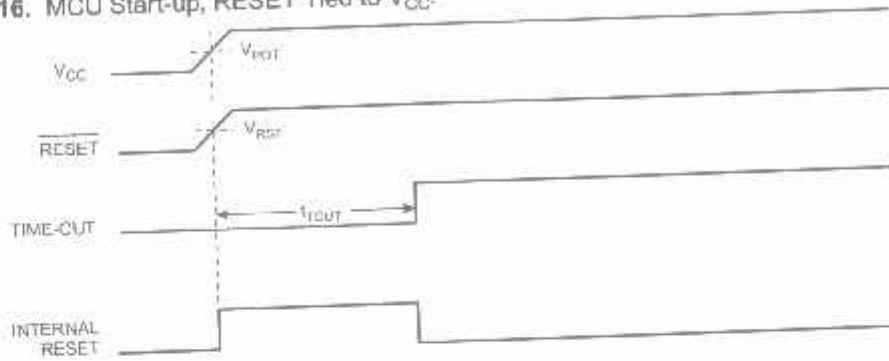
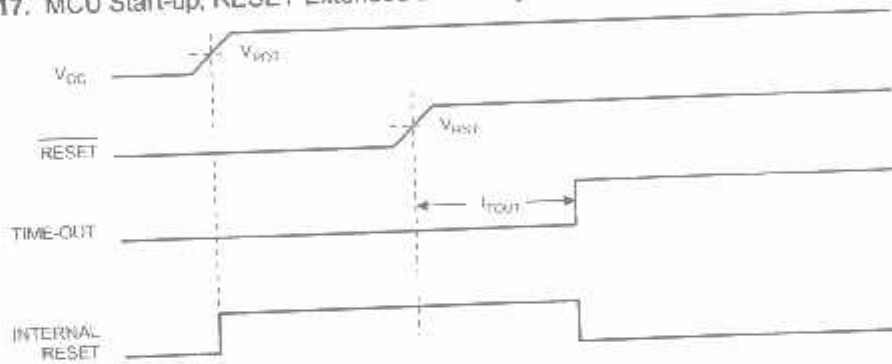


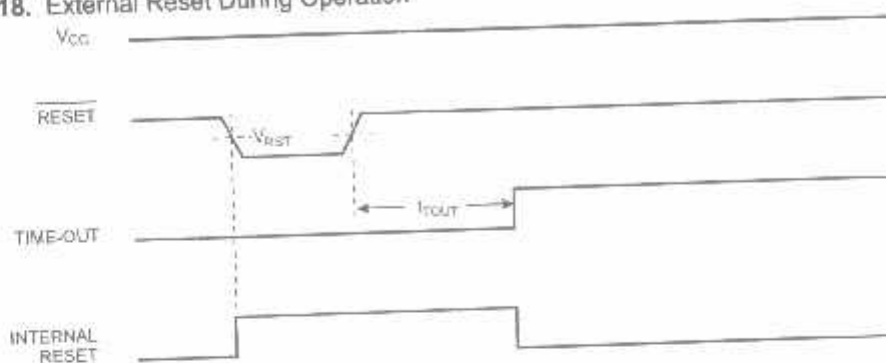
Figure 17. MCU Start-up, $\overline{\text{RESET}}$ Extended Externally



External Reset

An External Reset is generated by a low level on the $\overline{\text{RESET}}$ pin. Reset pulses longer than the minimum pulse width (see Table 15) will generate a reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a reset. When the applied signal reaches the Reset Threshold Voltage – V_{RST} – on its positive edge, the delay counter starts the MCU after the Time-out period t_{TOUT} has expired.

Figure 18. External Reset During Operation



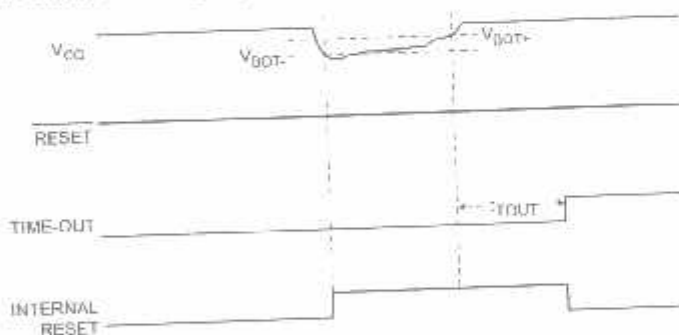
Brown-out Detection

ATmega16 has an On-chip Brown-out Detection (BOD) circuit for monitoring the V_{CC} level during operation by comparing it to a fixed trigger level. The trigger level for the BOD can be selected by the fuse BODLEVEL to be 2.7V (BODLEVEL unprogrammed), or 4.0V (BODLEVEL programmed). The trigger level has a hysteresis to ensure spike free Brown-out Detection. The hysteresis on the detection level should be interpreted as $V_{\text{BOT+}} = V_{\text{BOT}} + V_{\text{HYST}}/2$ and $V_{\text{BOT-}} = V_{\text{BOT}} - V_{\text{HYST}}/2$.

The BOD circuit can be enabled/disabled by the fuse BODEN. When the BOD is enabled (BODEN programmed), and V_{CC} decreases to a value below the trigger level ($V_{\text{BOT-}}$ in Figure 19), the Brown-out Reset is immediately activated. When V_{CC} increases above the trigger level ($V_{\text{BOT+}}$ in Figure 19), the delay counter starts the MCU after the Time-out period t_{TOUT} has expired.

The BOD circuit will only detect a drop in V_{CC} if the voltage stays below the trigger level for longer than t_{BOD} given in Table 15.

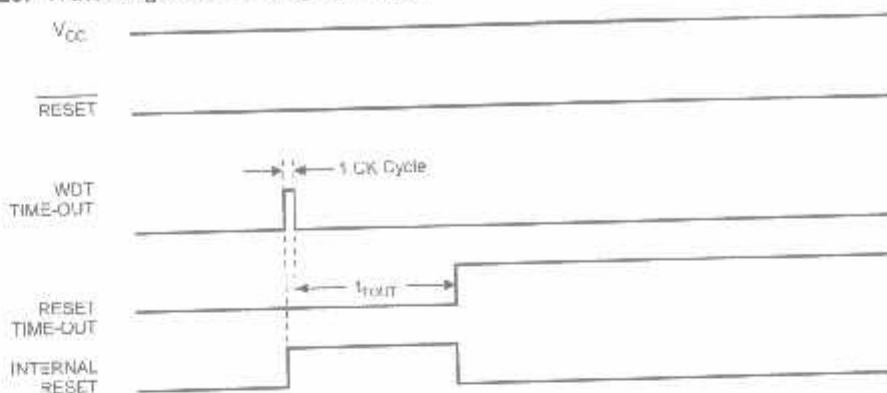
Figure 19. Brown-out Reset During Operation



Watchdog Reset

When the Watchdog times out, it will generate a short reset pulse of one CK cycle duration. On the falling edge of this pulse, the delay timer starts counting the Time-out period t_{TOUT} . Refer to page 42 for details on operation of the Watchdog Timer.

Figure 20. Watchdog Reset During Operation



The MCU Control and Status Register provides information on which reset source caused an MCU Reset.

Bit	7	6	5	4	3	2	1	0	
	JTD	ISCZ	-	JTRF	WDRF	BORF	EXTRF	PORF	MCUCSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initia Value	0	0	0						See Bit Description

- **Bit 4 – JTRF: JTAG Reset Flag**

This bit is set if a reset is being caused by a logic one in the JTAG Reset Register selected by the JTAG instruction AVR_RESET. This bit is reset by a Power-on Reset, or by writing a logic zero to the flag.

- **Bit 3 – WDRF: Watchdog Reset Flag**

This bit is set if a Watchdog Reset occurs. The bit is reset by a Power-on Reset, or by writing a logic zero to the flag.

- **Bit 2 – BORF: Brown-out Reset Flag**

This bit is set if a Brown-out Reset occurs. The bit is reset by a Power-on Reset, or by writing a logic zero to the flag.

- **Bit 1 – EXTRF: External Reset Flag**

This bit is set if an External Reset occurs. The bit is reset by a Power-on Reset, or by writing a logic zero to the flag.

- **Bit 0 – PORF: Power-on Reset Flag**

This bit is set if a Power-on Reset occurs. The bit is reset only by writing a logic zero to the flag.

To make use of the Reset Flags to identify a reset condition, the user should read and then reset the MCUCSR as early as possible in the program. If the register is cleared before another reset occurs, the source of the reset can be found by examining the Reset Flags.



Internal Voltage Reference

Voltage Reference Enable Signals and Start-up Time

ATmega16 features an internal bandgap reference. This reference is used for Brown-out Detection, and it can be used as an input to the Analog Comparator or the ADC. The 2.56V reference to the ADC is generated from the internal bandgap reference.

The voltage reference has a start-up time that may influence the way it should be used. The start-up time is given in Table 16. To save power, the reference is not always turned on. The reference is on during the following situations:

1. When the BOD is enabled (by programming the BODEN Fuse).
2. When the bandgap reference is connected to the Analog Comparator (by setting the ACBG bit in ACSR).
3. When the ADC is enabled.

Thus, when the BOD is not enabled, after setting the ACBG bit or enabling the ADC, the user must always allow the reference to start up before the output from the Analog Comparator or ADC is used. To reduce power consumption in Power-down mode, the user can avoid the three conditions above to ensure that the reference is turned off before entering Power-down mode.

Table 16. Internal Voltage Reference Characteristics

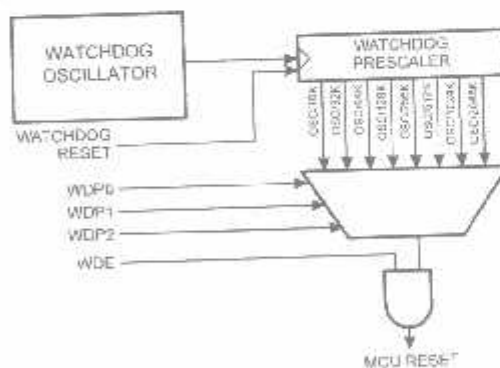
Symbol	Parameter	Min	Typ	Max	Units
V_{BG}	Bandgap reference voltage	1.15	1.23	1.4	V
t_{BG}	Bandgap reference start-up time		40	70	μ s
I_{BG}	Bandgap reference current consumption		10		μ A

Watchdog Timer

The Watchdog Timer is clocked from a separate On-chip Oscillator which runs at 1 MHz. This is the typical value at $V_{CC} = 5V$. See characterization data for typical values at other V_{CC} levels. By controlling the Watchdog Timer prescaler, the Watchdog Reset interval can be adjusted as shown in Table 17 on page 43. The WDR – Watchdog Reset – instruction resets the Watchdog Timer. The Watchdog Timer is also reset when it is disabled and when a Chip Reset occurs. Eight different clock cycle periods can be selected to determine the reset period. If the reset period expires without another Watchdog Reset, the ATmega16 resets and executes from the Reset Vector. For timing details on the Watchdog Reset, refer to page 41.

To prevent unintentional disabling of the Watchdog, a special turn-off sequence must be followed when the Watchdog is disabled. Refer to the description of the Watchdog Timer Control Register for details.

Figure 21. Watchdog Timer



Watchdog Timer Control Register – WDTCR

Bit	7	6	5	4	3	2	1	0	
	-	-	-	WDTOE	WDE	WDP2	WDP1	WDP0	WDTCR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits 7..5 – Res: Reserved Bits**

These bits are reserved bits in the ATmega16 and will always read as zero.

- **Bit 4 – WDTOE: Watchdog Turn-off Enable**

This bit must be set when the WDE bit is written to logic zero. Otherwise, the Watchdog will not be disabled. Once written to one, hardware will clear this bit after four clock cycles. Refer to the description of the WDE bit for a Watchdog disable procedure.

- **Bit 3 – WDE: Watchdog Enable**

When the WDE is written to logic one, the Watchdog Timer is enabled, and if the WDE is written to logic zero, the Watchdog Timer function is disabled. WDE can only be cleared if the WDTOE bit has logic level one. To disable an enabled Watchdog Timer, the following procedure must be followed:

1. In the same operation, write a logic one to WDTOE and WDE. A logic one must be written to WDE even though it is set to one before the disable operation starts.
2. Within the next four clock cycles, write a logic 0 to WDE. This disables the Watchdog.

- **Bits 2..0 – WDP2, WDP1, WDP0: Watchdog Timer Prescaler 2, 1, and 0**

The WDP2, WDP1, and WDP0 bits determine the Watchdog Timer prescaling when the Watchdog Timer is enabled. The different prescaling values and their corresponding Timeout Periods are shown in Table 17.

Table 17. Watchdog Timer Prescale Select

WDP2	WDP1	WDP0	Number of WDT Oscillator Cycles	Typical Time-out at $V_{CC} = 3.0V$	Typical Time-out at $V_{CC} = 5.0V$
0	0	0	16K (16,384)	17.1 ms	16.3 ms
0	0	1	32K (32,768)	34.3 ms	32.5 ms
0	1	0	64K (65,536)	68.5 ms	65 ms
0	1	1	128K (131,072)	0.14 s	0.13 s
1	0	0	256K (262,144)	0.27 s	0.26 s
1	0	1	512K (524,288)	0.55 s	0.52 s
1	1	0	1,024K (1,048,576)	1.1 s	1.0 s
1	1	1	2,048K (2,097,152)	2.2 s	2.1 s



The following code example shows one assembly and one C function for turning off the WDT. The example assumes that interrupts are controlled (for example by disabling interrupts globally) so that no interrupts will occur during execution of these functions.

Assembly Code Example

```
WDT_off:
  ; Reset WDT
  WDR
  ; Write logical one to WDTOR and WDR
  in  r16, WDTCR
  ori r16, (1<<WDTOR) | (1<<WDR)
  out WDTCR, r16
  ; Turn off WDT
  ldi r16, (0<<WDR)
  out WDTCR, r16
  ret
```

C Code Example

```
void WDT_off(void)
{
  /* Reset WDT */
  _WDR();
  /* Write logical one to WDTOR and WDR */
  WDTCR |= (1<<WDTOR) | (1<<WDR);
  /* Turn off WDT */
  WDTCR = 0x00;
}
```

Interrupts

This section describes the specifics of the interrupt handling as performed in ATmega16. For a general explanation of the AVR interrupt handling, refer to "Reset and Interrupt Handling" on page 13.

Interrupt Vectors in ATmega16

Table 18. Reset and Interrupt Vectors

Vector No.	Program Address ⁽¹⁾	Source	Interrupt Definition
1	\$000	RESET	External Pin, Power-on Reset, Brown-out Reset, Watchdog Reset, and JTAG AVR Reset
2	\$002	INT0	External Interrupt Request 0
3	\$004	INT1	External Interrupt Request 1
4	\$006	TIMER2 COMP	Timer/Counter2 Compare Match
5	\$008	TIMER2 OVF	Timer/Counter2 Overflow
6	\$00A	TIMER1 CAPT	Timer/Counter1 Capture Event
7	\$00C	TIMER1 COMPA	Timer/Counter1 Compare Match A
8	\$00E	TIMER1 COMPB	Timer/Counter1 Compare Match B
9	\$010	TIMER1 OVF	Timer/Counter1 Overflow
10	\$012	TIMER0 OVF	Timer/Counter0 Overflow
11	\$014	SPI, STC	Serial Transfer Complete
12	\$016	USART, RXC	USART, Rx Complete
13	\$018	USART, UDRE	USART Data Register Empty
14	\$01A	USART, TXC	USART, Tx Complete
15	\$01C	ADC	ADC Conversion Complete
16	\$01E	EE_RDY	EEPROM Ready
17	\$020	ANA_COMP	Analog Comparator
18	\$022	TWI	Two-wire Serial Interface
19	\$024	INT2	External Interrupt Request 2
20	\$026	TIMER0 COMP	Timer/Counter0 Compare Match
21	\$028	SPM_RDY	Store Program Memory Ready

- Notes:
1. When the BOOTRST Fuse is programmed, the device will jump to the Boot Loader address at reset, see "Boot Loader Support - Read-While-Write Self-Programming" on page 246.
 2. When the IVSEL bit in GICR is set, Interrupt vectors will be moved to the start of the Boot Flash section. The address of each Interrupt Vector will then be the address in this table added to the start address of the Boot Flash section.

Table 19 shows Reset and Interrupt Vectors placement for the various combinations of BOOTRST and IVSEL settings. If the program never enables an interrupt source, the Interrupt Vectors are not used, and regular program code can be placed at these locations. This is also the case if the Reset Vector is in the Application section while the Interrupt Vectors are in the Boot section or vice versa.



Table 19. Reset and Interrupt Vectors Placement⁽¹⁾

BOOTRST	IVSEL	Reset address	Interrupt Vectors Start Address
1	0	\$0000	\$0002
1	1	\$0000	Boot Reset Address + \$0002
0	0	Boot Reset Address	\$0002
0	1	Boot Reset Address	Boot Reset Address + \$0002

Note: 1. The Boot Reset Address is shown in Table 100 on page 257. For the BOOTRST Fuse "1" means unprogrammed while "0" means programmed.

The most typical and general program setup for the Reset and Interrupt Vector Addresses in ATmega16 is:

Address	Labels	Code	Comments
\$000		jmp RESET	; Reset Handler
\$002		jmp EXT_INT0	; IRQ0 Handler
\$004		jmp EXT_INT1	; IRQ1 Handler
\$006		jmp TIM2_COMP	; Timer2 Compare Handler
\$008		jmp TIM2_OVP	; Timer2 Overflow Handler
\$00A		jmp TIM1_CAPT	; Timer1 Capture Handler
\$00C		jmp TIM1_COMPA	; Timer1 CompareA Handler
\$00E		jmp TIM1_COMPB	; Timer1 CompareB Handler
\$010		jmp TIM1_OVP	; Timer1 Overflow Handler
\$012		jmp TIM0_OVP	; Timer0 Overflow Handler
\$014		jmp SPI_STC	; SPI Transfer Complete Handler
\$016		jmp USART_RXC	; USART RX Complete Handler
\$018		jmp USART_UDRE	; UDR Empty Handler
\$01A		jmp USART_TXC	; USART TX Complete Handler
\$01C		jmp ADC	; ADC Conversion Complete Handler
\$01E		jmp EE_RDY	; EEPROM Ready Handler
\$020		jmp ANA_COMP	; Analog Comparator Handler
\$022		jmp TWI1	; Two-wire Serial Interface Handler
\$024		jmp EXT_INT2	; IRQ2 Handler
\$026		jmp TINC_COMP	; Timer0 Compare Handler
\$028		jmp SPM_RDY	; Store Program Memory Ready Handler
\$02A	RESET:	ldi r16,high(RAMEND)	; Main program start
\$02B		out SPH,r16	; Set Stack pointer to top of RAM
\$02C		ldi r16,low(RAMEND)	
\$02D		out SPL,r16	
\$02E		sei	; Enable interrupts
\$02F		rjmp xxx	
...	



When the BOOTRST Fuse is unprogrammed, the Boot section size set to 2 Kbytes and the IVSEL bit in the GICR Register is set before any interrupts are enabled, the most typical and general program setup for the Reset and Interrupt Vector Addresses is:

Address	Labels	Code	Comments
\$000	RESET:	ldi r16,high(RAMEND)	; Main program start
\$001		out SPH,r16	; Set Stack Pointer to top of RAM
\$002		ldi r16,low(RAMEND)	
\$003		out SPL,r16	
\$004		sei	; Enable interrupts
\$005		<instr> xxx	
;			
.org \$1C02			
\$1C02		jmp EXT_INT0	; IRQ0 Handler
\$1C04		jmp EXT_INT1	; IRQ1 Handler
...	;
\$1C28		jmp SPM_RDY	; Store Program Memory Ready Handler

When the BOOTRST Fuse is programmed and the Boot section size set to 2 Kbytes, the most typical and general program setup for the Reset and Interrupt Vector Addresses is:

Address	Labels	Code	Comments
.org \$002			
\$002		jmp EXT_INT0	; IRQ0 Handler
\$004		jmp EXT_INT1	; IRQ1 Handler
...	;
\$028		jmp SPM_RDY	; Store Program Memory Ready Handler
;			
.org \$1C00			
\$1C00	RESET:	ldi r16,high(RAMEND)	; Main program start
\$1C01		out SPH,r16	; Set Stack Pointer to top of RAM
\$1C02		ldi r16,low(RAMEND)	
\$1C03		out SPL,r16	
\$1C04		sei	; Enable interrupts
\$1C05		<instr> xxx	

When the BOOTRST Fuse is programmed, the Boot section size set to 2 Kbytes and the IVSEL bit in the GICR Register is set before any interrupts are enabled, the most typical and general program setup for the Reset and Interrupt Vector Addresses is:

Address	Labels	Code	Comments
.org \$1C00			
\$1C00		jmp RESET	; Reset handler
\$1C02		jmp EXT_INT0	; IRQ0 Handler
\$1C04		jmp EXT_INT1	; IRQ1 Handler
...	;
\$1C28		jmp SPM_RDY	; Store Program Memory Ready Handler
;			
\$1C2A	RESET:	ldi r16,high(RAMEND)	; Main program start
\$1C2B		out SPH,r16	; Set Stack Pointer to top of RAM
\$1C2C		ldi r16,low(RAMEND)	
\$1C2D		out SPL,r16	
\$1C2E		sei	; Enable interrupts
\$1C2F		<instr> xxx	

Moving Interrupts Between Application and Boot Space

General Interrupt Control Register – GICR

The General Interrupt Control Register controls the placement of the Interrupt Vector table.

Bit	7	6	5	4	3	2	1	0	
	INT1	INT0	INT2	-	-	-	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R/W	R	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

• Bit 1 – IVSEL: Interrupt Vector Select

When the IVSEL bit is cleared (zero), the Interrupt Vectors are placed at the start of the Flash memory. When this bit is set (one), the interrupt vectors are moved to the beginning of the Boot Loader section of the Flash. The actual address of the start of the Boot Flash section is determined by the BOOTSZ Fuses. Refer to the section "Boot Loader Support – Read-While-Write Self-Programming" on page 246 for details. To avoid unintentional changes of Interrupt Vector tables, a special write procedure must be followed to change the IVSEL bit:

1. Write the Interrupt Vector Change Enable (IVCE) bit to one.
2. Within four cycles, write the desired value to IVSEL while writing a zero to IVCE.

Interrupts will automatically be disabled while this sequence is executed. Interrupts are disabled in the cycle IVCE is set, and they remain disabled until after the instruction following the write to IVSEL. If IVSEL is not written, interrupts remain disabled for four cycles. The I-bit in the Status Register is unaffected by the automatic disabling.

Note: If Interrupt Vectors are placed in the Boot Loader section and Boot Lock bit BLB02 is programmed, interrupts are disabled while executing from the Application section. If Interrupt Vectors are placed in the Application section and Boot Lock bit BLB12 is programmed, interrupts are disabled while executing from the Boot Loader section. Refer to the section "Boot Loader Support – Read-While-Write Self-Programming" on page 246 for details on Boot Lock bits.

• Bit 0 – IVCE: Interrupt Vector Change Enable

The IVCE bit must be written to logic one to enable change of the IVSEL bit. IVCE is cleared by hardware four cycles after it is written or when IVSEL is written. Setting the IVCE bit will disable interrupts, as explained in the IVSEL description above. See Code Example below

Assembly Code Example

```
Move_interrupts:
; Enable change of interrupt vectors
ldi r16, (1<<IVCFE)
out GICR, r16
; Move interrupts to boot Flash section
ldi r16, (1<<IVSEL)
out GICR, r16
ret
```

C Code Example

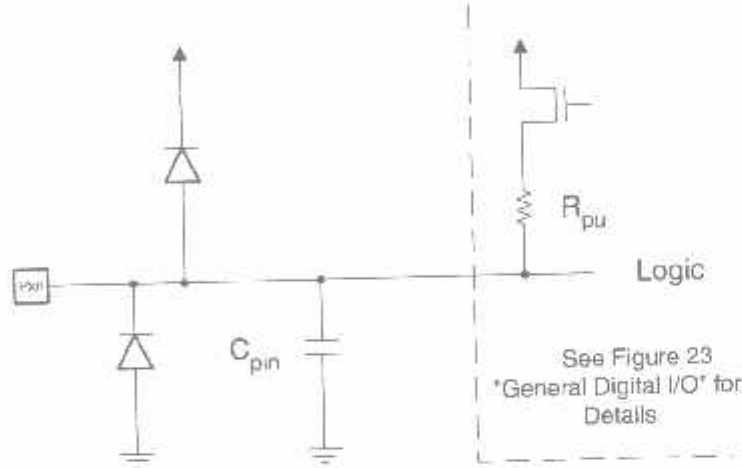
```
void Move_interrupts(void)
{
/* Enable change of interrupt vectors */
GICR = (1<<IVCFE);
/* Move interrupts to boot Flash section */
GICR = (1<<IVSEL);
}
```

I/O Ports

Introduction

All AVR ports have true Read-Modify-Write functionality when used as general digital I/O ports. This means that the direction of one port pin can be changed without unintentionally changing the direction of any other pin with the SBI and CBI instructions. The same applies when changing drive value (if configured as output) or enabling/disabling of pull-up resistors (if configured as input). Each output buffer has symmetrical drive characteristics with both high sink and source capability. The pin driver is strong enough to drive LED displays directly. All port pins have individually selectable pull-up resistors with a supply-voltage invariant resistance. All I/O pins have protection diodes to both V_{CC} and Ground as indicated in Figure 22. Refer to "Electrical Characteristics" on page 281 for a complete list of parameters.

Figure 22. I/O Pin Equivalent Schematic



All registers and bit references in this section are written in general form. A lower case "x" represents the numbering letter for the port, and a lower case "n" represents the bit number. However, when using the register or bit defines in a program, the precise form must be used, that is, PORTB3 for bit no. 3 in Port B, here documented generally as PORTxn. The physical I/O Registers and bit locations are listed in "Register Description for I/O Ports" on page 66.

Three I/O memory address locations are allocated for each port, one each for the Data Register – PORTx, Data Direction Register – DDRx, and the Port Input Pins – PINx. The Port Input Pins I/O location is read only, while the Data Register and the Data Direction Register are read/write. In addition, the Pull-up Disable – PUD bit in SFIOR disables the pull-up function for all pins in all ports when set.

Using the I/O port as General Digital I/O is described in "Ports as General Digital I/O" on page 50. Most port pins are multiplexed with alternate functions for the peripheral features on the device. How each alternate function interferes with the port pin is described in "Alternate Port Functions" on page 55. Refer to the individual module sections for a full description of the alternate functions.

Note that enabling the alternate function of some of the port pins does not affect the use of the other pins in the port as general digital I/O.

Ports as General Digital I/O

The ports are bi-directional I/O ports with optional internal pull-ups. Figure 23 shows a functional description of one I/O-port pin, here generically called Pxn.





PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : **DAULAT KRISTON NAINGGOLAN**
Nim : **0217057**
Semester : **XIV (Empatbelas)**
Jurusan : **Teknik Elektro S-1**
Konsentrasi : **Teknik Elektronika**

Dengan ini menyatakan bersedia/tidak bersedia*) Membimbing skripsi dari mahasiswa tersebut, dengan judul :

" PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR MENGGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16"

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Hormat Kami

Irmalia Suryani Faradisa, ST, MT
NRP 1030000365

*) Coret yang tidak perlu



ampiran : 1 (satu) berkas
Pembimbing Skripsi

kepada : Yth. Bapak/Ibu **Yuli Wabyuni, ST, MT**
 Dosen Teknik Elektro S-1
 ITN Malang

Yang bertanda tangan dibawah

Nama : **DAULAT KRISTON NAINGGOLAN**
Nim : **0217057**
Jurusan : **Teknik Elektro S-1**
Konsentrasi : **Teknik Elektronika**


Dengan ini mengajukan permohonan, kiranya Bapak/Tbu bersedia menjadi Dosen Pembimbing untuk penyusunan Skripsi dengan judul :

**"PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR
MENGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16"**


Demikian permohonan kami buat dan atas kesediaan Bapak kami ucapkan terima kasih.

Mengetahui

etua Program Studi Teknik Elektro S-1


M. Ibrahim Ashari, ST, MT
NIP.P. 1030100358

Hormat Kami


DAULAT KRISTON NAINGGOLAN
NIM. 0217057



PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : **DAULAT KRISTON NAINGGOLAN**
Nim : **0217057**
Semester : **XIV (Empatbelas)**
Jurusan : **Teknik Elektro S-1**
Konsentrasi : **Teknik Elektronika**

Dengan ini menyatakan bersedia/tidak bersedia*) Membimbing skripsi dari mahasiswa tersebut, dengan judul :

**" PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR
MENGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16"**

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Hormat Kami

Yuli Wahyuni, ST, MT

NIP.P. 1031200456

Catatan :

Setelah disetujui agar formulir ini Diserahkan mahasiswa/i yang bersangkutan kepada jurusan untuk diproses lebih lanjut

*) Coret yang tidak perlu



BERTITA ACARA SEMINAR PROPOSAL SKRIPSI
PROGRAM STUDI TEKNIK ELEKTRO S-1
Konstraksi : Teknik Elektronika

1.	Nim	0217057		
2.	Nama	DAULAT KRISTON NAINGGOLAN		
3.	Konstraksi Jurusan	Teknik Elektronika		
4.	Jadwal Pelaksanaan	Waktu	Tempat	
		20 Juni 2013	09.00	III.1.5
5.	Judul proposal yang diseminarkan Mahasiswa	PERENCANAAN DAN PEMBUATAN ALAT PENGAMANAN SEPEDA MOTOR MENGGUNAKAN RFID BERBASIS MIKROKONTROLER ATMEGA 16		
6.	Perubahan judul yang diusulkan oleh Kelompok Dosen Keahlian			
7.	Catatan			
8.	Catatan :			
	Fersetujuan judul Skripsi			
	Disetujui, Dosen Keahlian I	Disetujui, Dosen Keahlian II	Disetujui, Dosen Keahlian III	
	(.....)	(.....)	(.....)	
	Mengetahui, Ketua Program Studi Teknik Elektro S-1 <i>M. Ibrahim Ashari</i> M. Ibrahim Ashari, ST, MT NIP. P 1030106358	Disetujui, Calon Dosen Pembimbing ybs		
		Pembimbing I	Pembimbing II	
		<i>[Signature]</i> (.....)	<i>[Signature]</i> (.....)	



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting). Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa : Daulat Kriston Nainggolan
NIM : 02.17.057
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : PERANCANGAN DAN PEMBUATAN ALAT
PENGAMANAN SEPEDA MOTOR MENGGUNAKAN
RFID BERBASIS MIKROKONTROLER ATMEGA 16

Dipertahankan dihadapan tim penguji skripsi jenjang Strata Satu (S-1) pada:

Hari : Selasa
Tanggal : 20 Agustus 2013
Dengan Nilai : 77 (B+) *rs*

PANITIA UJIAN SKRIPSI

Ketua Prodi Teknik Elektro S-1

Ibrahim Ashari, ST, MT
NIP. P. 1030100358

Sekretaris Jurusan Teknik Elektro S-1

Dr. Eng Aryuanto, ST, MT
NIP. Y. 1030800417

DOSEN PENGUJI

Dosen Penguji I

Bambang Prio Hartono, ST, MT
NIP. Y. 102840082

Dosen Penguji II

Ibrahim Ashari, ST, MT
NIP. P. 1030100358



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
 FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
 PROGRAM PASCASARJANA MAGISTER TEKNIK

T. BNI (PERSERO) MALANG
 BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
 Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Formulir Perbaikan Skripsi

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : DAULAT KRISTON NAINGGOLAN
 NIM : 02.17.057
 Jurusan : Teknik Elektro S-1
 Konsentrasi : Teknik Elektronika
 Judul Skripsi : PERANCANGAN DAN PEMBUATAN ALAT
 PENGAMANAN SEPEDA MOTOR MENGGUNAKAN
 RFID BERBASIS MIKROKONTROLER ATMEGA 16

No.	Dosen Penguji	Tanggal	Uraian	Paraf
1	Dosen Penguji I		1. Abstrak 2. Photo Alat	
2	Dosen Penguji II			

Disetujui oleh :

Dosen Penguji I

Bambang Prio Hartono, ST, MT
 NIP. Y. 102840082

Dosen Penguji II

Ibrahim Ashari, ST, MT
 NIP. P. 1030100358

Mengetahui :

Dosen Pembimbing I

Irmalia Suryani Faradisa, ST, MT
 NIP. P. 1020000265

Dosen Pembimbing II

Yuli Wahyuni, ST, MT
 NIP. P. 1031200456

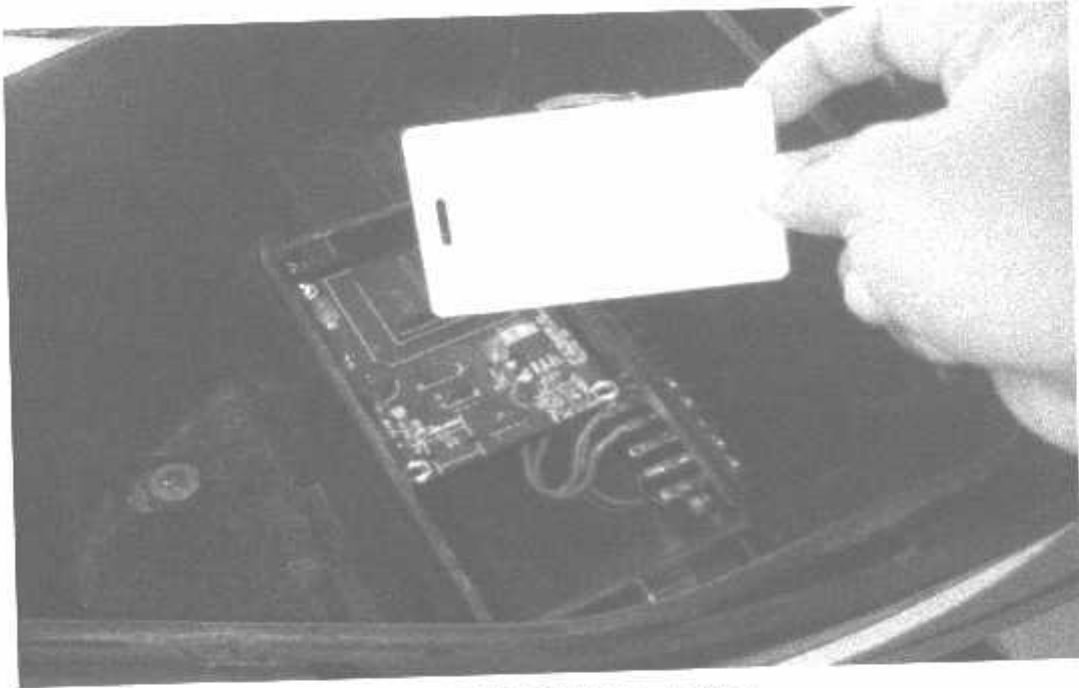


FOTO SAAT PENGUJIAN ALAT II



GAMBAR KESELURUHAN

LAMPIRAN FOTO ALAT

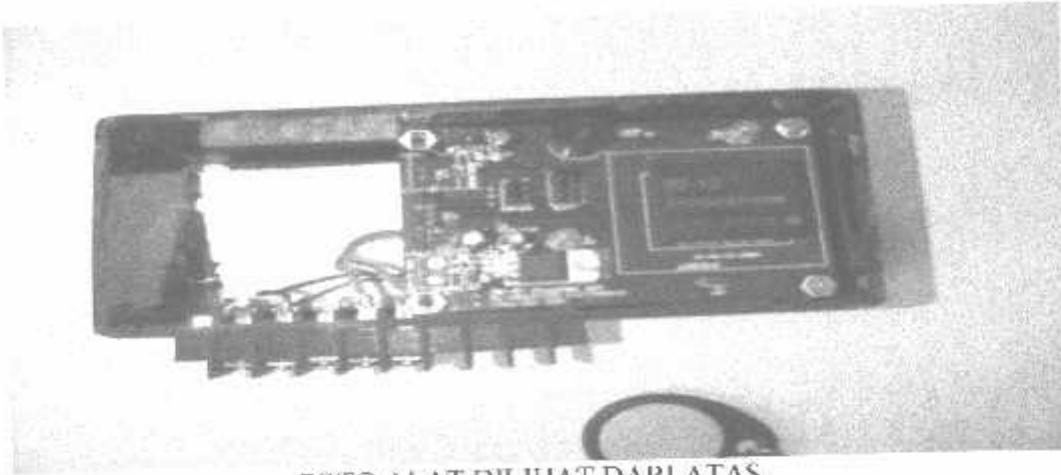


FOTO ALAT DILIHAT DARI ATAS



FOTO ALAT DILIHAT DARI SAMPING



FOTO SAAT PENGUJIAN ALAT I

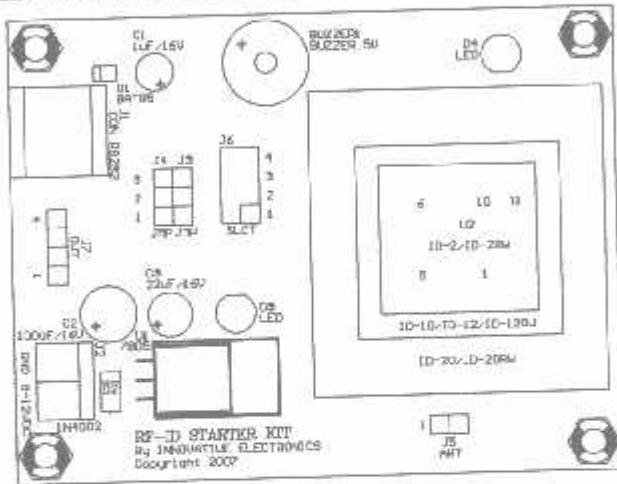
RFID Starter Kit

RFID Starter Kit merupakan suatu sarana pengembangan RFID berbasis reader tipe ID-12 yang telah dilengkapi dengan jalur komunikasi RS-232 serta indikator buzzer dan LED. Modul ini dapat digunakan dalam aplikasi mesin absensi RFID, RFID access controller, dsb.

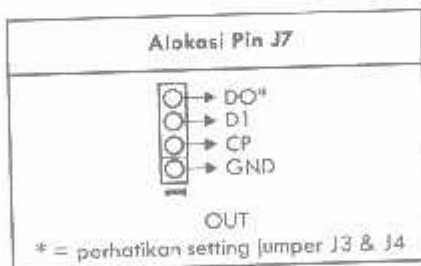
Spesifikasi

1. Berbasis RFID reader ID-12 dengan frekuensi kerja 125 kHz untuk kartu berformat EM4001/sejenis dan memiliki jarak baca maksimal 12 cm.
2. Kompatibel dengan varian RFID reader lainnya, antara lain ID-2, ID-10, dan ID-20.
3. Mendukung varian RFID reader/writer, antara lain ID-2RW, ID-12RW, dan ID-20RW.
4. Mendukung format data ASCII (UART TTL/RS-232), Wiegand26, maupun Magnetic ABA Track2 (Magnet Emulation).
5. Dilengkapi dengan buzzer sebagai indikator baca, serta LED sebagai indikator tulis.
6. Tersedia jalur komunikasi serial UART RS-232 dengan konektor RJ11.
7. Tegangan input catu daya 9 - 12 VDC (J2).

Tata Letak dan Setting Jumper



Antena dapat dihubungkan ke J5 untuk RFID reader only atau reader/writer yang memerlukan antena eksternal, seperti ID-2 dan ID-2RW.



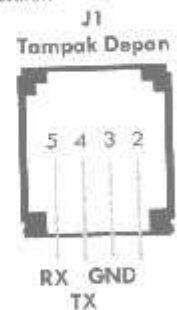
J7 hanya digunakan untuk RFID reader only dalam mode UART TTL (ASCII), Wiegand26, dan Magnet Emulation. J7 tidak boleh digunakan pada mode lain.

Pengaturan jumper J3, J4, dan J6 harus disesuaikan dengan jenis RFID (reader only atau reader/writer) serta format data RFID reader yang akan digunakan.

Setting Jumper J3, J4, & J6		
		RFID reader only dengan format data UART RS-232 (ASCII).
		RFID reader only dengan format data UART TTL (ASCII).
		RFID reader only dengan format data Wiegand26
		RFID reader only dengan format data Magnet Emulation
		RFID reader/writer dengan antarmuka UART RS 232.

Adapun hubungan antara komputer dengan RFID Starter Kit adalah "Straight" dengan konfigurasi sebagai berikut:

COM port Komputer DB9	RFID Starter Kit J1
RX (pin 2)	RX (pin 5)
TX (pin 3)	TX (pin 4)
GND (pin 5)	GND (pin 3)



J1 hanya digunakan untuk RFID reader only dalam mode UART RS-232 (ASCII) dan RFID reader/writer. Pada mode lain, J1 tidak boleh digunakan dan kabel tidak boleh terhubung.

Isi CD

1. Contoh Aplikasi dan Program Testing.
2. Datasheet RFID Reader ID-12.
3. Manual RFID Starter Kit.
4. Website Innovative Electronics

Prosedur Testing

Prosedur testing berikut akan menguji jalur komunikasi RS-232 dan RFID reader ID-12 dalam mode ASCII.

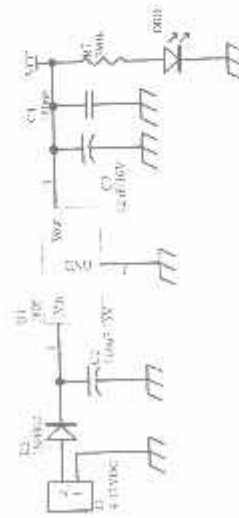
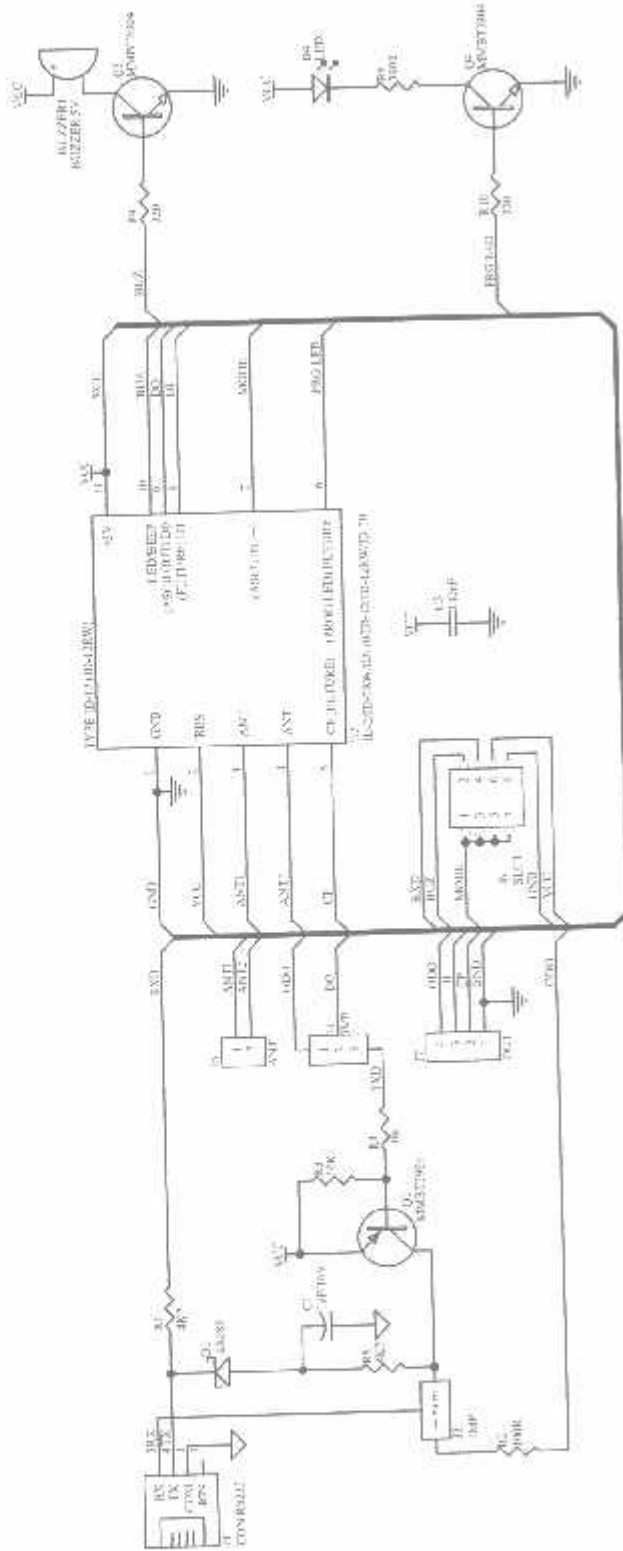
Langkah-langkah testing:

- Atur RFID Starter Kit agar RFID reader ID-12 bekerja pada mode UART RS-232 (ASCII), yaitu jumper J3 & J4 pada posisi 2-3 dan jumper J6 pada posisi 4.
- Hubungkan RJ11 (J1) RFID Starter Kit ke COM port komputer menggunakan kabel serial.
- Hubungkan catu daya 9 VDC ke terminal J2 RFID Starter Kit.

- Jalankan program RFID1.exe, lalu pilih COM port yang sesuai.
- Nyalakan catu daya, lalu dekatkan RFID transponder ke RFID reader. Pada program RFID1.exe akan muncul nomor ID dari RFID transponder tersebut.

* Terima Kasih atas kepercayaan Anda menggunakan produk kami, bila ada kesulitan, pertanyaan atau saran mengenai produk ini silahkan menghubungi technical support kami:

Support@innovativeelectronics.com

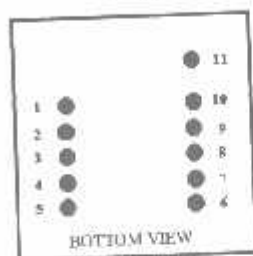


ID SERIES DATASHEET Feb 10, 2004

ID-2 / ID-12 / ID-20

The ID2, ID12 and ID20 are similar to the ID0, ID10 and ID15 MK(ii) series devices, but they have extra pins which allow Magnetic Emulation output to be included in the functionality. The ID-12 and ID-20 come with internal antennas, and have read ranges of 12+ cm and 16+ cm, respectively. With an external antenna, the ID-2 can deliver read ranges of up to 25 cm. All three readers support ASCII, Wiegand26 and Magnetic ABA Track2 data formats.

ID2 / ID12 / ID20 PIN-OUT



1. GND
2. RES (Reset Bar)
3. ANT (Antenna)
4. ANT (Antenna)
5. CP
6. Future
7. +/- (Format Selector)
8. D1 (Data Pin 1)
9. D0 (Data Pin 0)
10. LED (LED / Beeper)
11. +5V



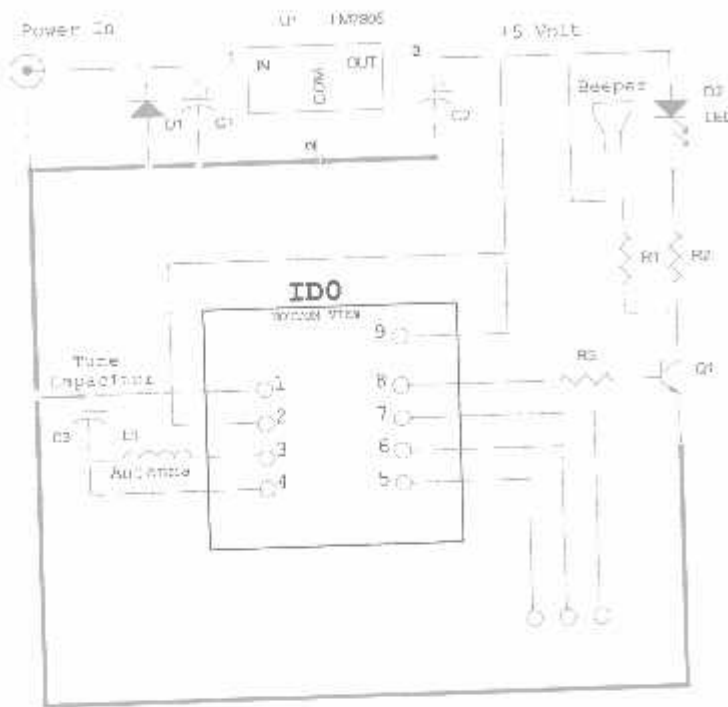
Operational and Physical Characteristics

Parameters	ID-2	ID-12	ID-20
Read Range	N/A (no internal antenna)	12+ cm	16+ cm
Dimensions	21 mm x 19 mm x 6 mm	26 mm x 25 mm x 7 mm	40 mm x 40 mm x 9 mm
Frequency	125 kHz	125 kHz	125 kHz
Card Format	EM 4001 or compatible	EM 4001 or compatible	EM 4001 or compatible
Encoding	Manchester 64-bit, modulus 64	Manchester 64-bit, modulus 64	Manchester 64-bit, modulus 64
Power Requirement	5 VDC @ 13mA nominal	5 VDC @ 30mA nominal	5 VDC @ 65mA nominal
I/O Output Current	+/-200mA PK	-	-
Voltage Supply Range	+4.6V through +5.4V	+4.6V through +5.4V	+4.6V through +5.4V

Pin Description & Output Data Formats

Pin No.	Description	ASCII	Magnet Emulation	Wiegand26
Pin 1	Zero Volts and Tuning Capacitor Ground	GND 0V	GND 0V	GND 0V
Pin 2	Strap to +5V	Reset Bar	Reset Bar	Reset Bar
Pin 3	To External Antenna and Tuning Capacitor	Antenna	Antenna	Antenna
Pin 4	To External Antenna	Antenna	Antenna	Antenna
Pin 5	Card Present	No function	Card Present	No function
Pin 6	Future	Future	Future	Future
Pin 7	Format Selector (+/-)	Strap to GND	Strap to Pin 10	Strap to +5V
Pin 8	Data 1	CMOS	Clock	One Output
Pin 9	Data 0	TTL Data (inverted)	Data	Zero Output
Pin 10	3.1 kHz Logic	Beeper / LED	Beeper / LED	Beeper / LED
Pin 11	DC Voltage Supply	+5V	+5V	+5V

Circuit Diagram for the ID0



COMPONENT LIST

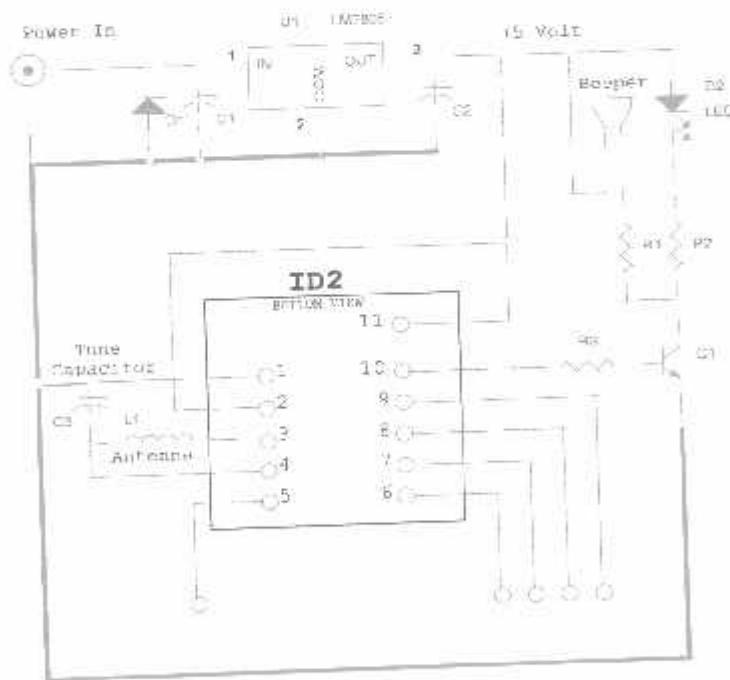
- R1 = 100R
- R2 = 1K
- R3 = 1K
- C1 = 100uF 16V
- C2 = 100uF 10V
- C3 = 1nF COG 100V *
- Beeper = 2.7-3.5KHz 100R
- D1 = 1N4001
- D2 = GREEN LED
- U1 = LM7805
- Q1 = U1C8050 (NPN)
- L1 = 640Uh

ID0 = ID Innovations ID0

* Please Note the ID0 has an internal tuning capacitor of 1.5nF and this makes the total tuning capacity - 2.5nF

The 3.1Khz Beeper Logic is centered for most Beeper in range 2.7-3.5Khz

Circuit Diagram for the ID2



COMPONENT LIST

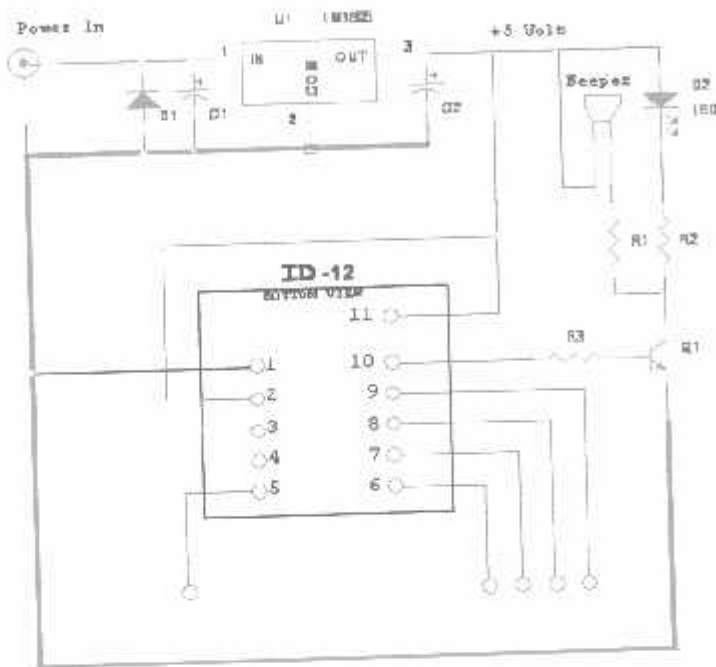
- R1 = 100R
- R2 = 1K
- R3 = 1K
- C1 = 100uF 16V
- C2 = 100uF 10V
- C3 = 1nF COG 100V *
- Beeper = 2.7-3.5KHz 100R
- D1 = 1N4001
- D2 = GREEN LED
- U1 = LM7805
- Q1 = U1C8050 (NPN)
- L1 = 640Uh

ID2 = ID Innovations ID2

* Please Note the ID2 has an internal tuning capacitor of 1.5nF and this makes the total tuning capacity - 2.5nF

The 3.1Khz Beeper Logic is centered for most Beeper in range 2.7-3.5Khz

Circuit Diagram for the ID-12



COMPONENT LIST

R1 = 100R
 R2 = 1K
 R3 = 1K
 C1 = 100uF 16V
 C2 = 100uF 10V
 Beeper = 2.7-3.5KHz 100R
 D1 = 1N4001
 D2 = GREEN LED
 U1 = LM7805
 Q1 = UTC8050 (NPN)
 ID2 = ID Innovations ID2

* Please Note the ID2 has an internal tuning capacitor of 1.5nF and this makes the total tuning capacity = 2.5nF

The 3.1KHz Beeper Logic is centered for most Beeper in range 2.7-3.5KHz

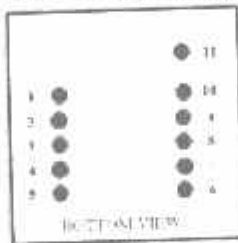
ID SERIES DATASHEET MAR 01, 2005

ID-2/ID-12 Brief Data

The ID2, ID12 and ID20 are similar to the obsolete ID0, ID10 and ID15 MK(ii) series devices, but they have extra pins that allow Magnetic Emulation output to be included in the functionality. The ID-12 and ID-20 come with internal antennas, and have read ranges of 12+ cm and 16+ cm, respectively. With an external antenna, the ID-2 can deliver read ranges of up to 25 cm. All three readers support ASCII, Wiegand26 and Magnetic ABA Track2 data formats.



ID2 / ID12 / ID20 PIN-OUT



1. GND
2. RES (Reset Bar)
3. ANT (Antenna)
4. ANT (Antenna)
5. CP
6. Future
7. +/- (Format Selector)
8. D1 (Data Pin 1)
9. D0 (Data Pin 0)
10. LED (LED / Beeper)
11. +5V

Operational and Physical Characteristics

Parameters	ID-2	ID-12	ID-20
Read Range	N/A (no internal antenna)	12+ cm	16+ cm
Dimensions	21 mm x 19 mm x 6 mm	26 mm x 25 mm x 7 mm	40 mm x 40 mm x 9 mm
Frequency	125 kHz	125 kHz	125 kHz
Card Format	EM 4001 or compatible	EM 4001 or compatible	EM 4001 or compatible
Encoding	Manchester 64-bit, modulus 64	Manchester 64-bit, modulus 64	Manchester 64-bit, modulus 64
Power Requirement	5 VDC @ 13mA nominal	5 VDC @ 30mA nominal	5 VDC @ 85mA nominal
I/O Output Current	+/- 200mA PK	-	-
Voltage Supply Range	+4.6V through +5.4V	+4.6V through +5.4V	+4.6V through +5.4V

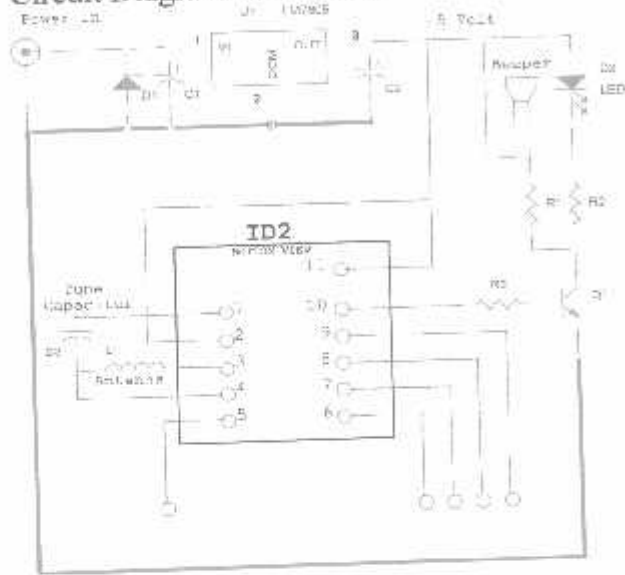
Pin Description & Output Data Formats

Pin No.	Description	ASCII	Magnet Emulation	Wiegand26
Pin 1	Zero Volts and Tuning Capacitor Ground	GND 0V	GND 0V	GND 0V
Pin 2	Strap to +5V	Reset Bar	Reset Bar	Reset Bar
Pin 3	To External Antenna and Tuning Capacitor	Antenna	Antenna	Antenna
Pin 4	To External Antenna	Antenna	Antenna	Antenna
Pin 5	Card Present	No function	Card Present	No function

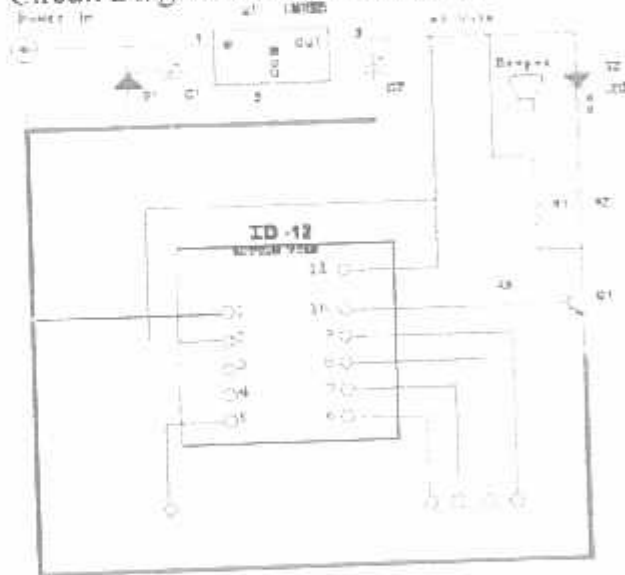
Pin 6	Future	Future	Future	Future
Pin 7	Format Selector (+/-)	Strap to GND	Strap to Pin 10	Strap to +5V
Pin 8	Data 1	CMOS	Clock *	One Output *
Pin 9	Data 0	TTL Data (inverted)	Data *	Zero Output *
Pin 10	3.1 kHz Logic	Beeper / LED	Beeper / LED	Beeper / LED
Pin 11	DC Voltage Supply	+5V	+5V	+5V

* Requires 4K7 Pull-up resistor to +5V

Circuit Diagram for the ID2

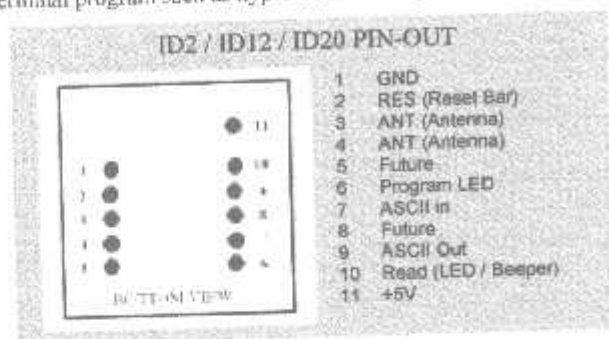


Circuit Diagram for the ID-12/ID20



ID-2RW/ID-12RW Brief Data

The ID2-RW, ID12-RW and ID15-RW are a new series of Read/Write modules for the Temec Q5 tag. It has full functionality including password. They contain built-in algorithms to assist customers programming the popular Sokymat Unique type tag. Password protection is allowed. Control is via a host computer using a simple terminal program such as hyper terminal or Qmodem.



Operational and Physical Characteristics

Parameters	ID-2RW	ID-12RW	ID-20RW
Read Range	N/A (no internal antenna)	12+ cm (Unique Format)	15+ cm (Unique Format)
Dimensions	21 mm x 19 mm x 8 mm	26 mm x 25 mm x 7 mm	40 mm x 40 mm x 9 mm
Frequency	125 kHz	125 kHz	125 kHz
Card Format	Temec Q5555	Temec Q5555	Temec Q5555
Read Encoding	Manchester modulus 64	Manchester modulus 64	Manchester modulus 64
Power Requirement	5 VDC @ 13mA nominal	5 VDC @ 30mA nominal	5 VDC @ 50mA nominal
I/O Output Current	-/-200mA PK.	-	-
Voltage Supply Range	+4.6V through +5.4V	+4.6V through +5.4V	+4.6V through +5.4V
Coil Detail	L - 0.6mH - 1.5mH, Q - 15-30	-	-

Description

A simple terminal program such as Qmodem or Hyper-terminal can be used to send commands to the module. The blocks are individually programmable. The command interface is simple to use and easily understood. The programmer also has two types of internal reader. One of these is provided to read Sokymat 'Unique' type tag configuration. The module does not require a MAX232 type chip interface. The module does not need an RS232 interface such as a MAX232 IC. The input pin7 goes to the computer through a 4k7 resistor and the output goes to the computer through a 100R resistor.

DATA FORMATS

Output Data Structure – ASCII

STX (02h)	DATA (10 ASCII)	CHECK SUM (2 ASCII)	CR	LF	ETX (03h)
-----------	-----------------	---------------------	----	----	-----------

[The 1byte (2 ASCII characters) Check sum is the "Exclusive OR" of the 5 hex bytes (10 ASCII) Data characters.]

Output Data Structure – Wiegand26

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
P	E	E	E	E	E	E	E	E	E	E	E	E	O	O	O	O	O	O	O	O	O	O	O	O	O	P
Even parity (E)													Odd parity (O)													

P – Parity start bit and stop bit

Output Data Magnetic ABA Track2

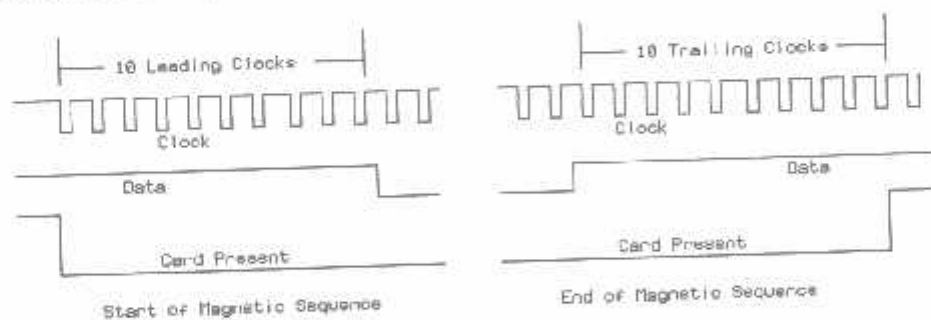
10 Leading Zeros	SS	Data	ES	LCR	10 Ending Zeros
------------------	----	------	----	-----	-----------------

[SS is the Start Character of 11010, ES is the end character of 11111, LRC is the Longitudinal Redundancy Check.]

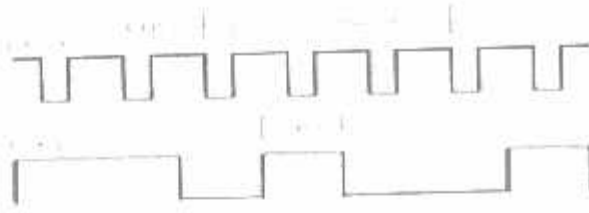
Magnetic Emulation Waveforms



Start and End Sequences For Magnetic Timing

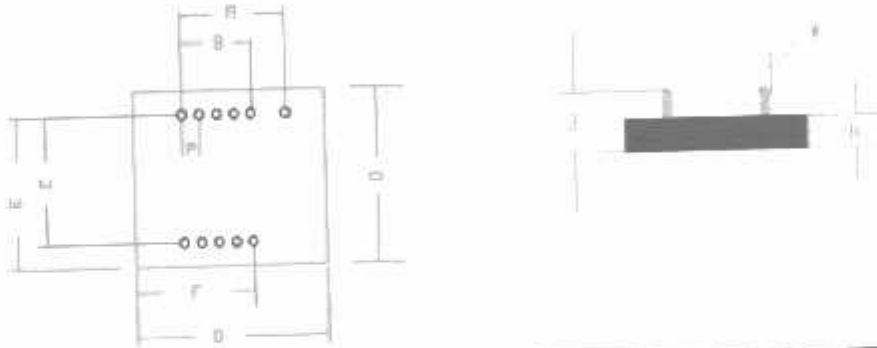


DATA TIMINGS FOR MAGNETIC EMULATION



The magnetic Emulation Sequence starts with the Card Present Line going active (down). There next follows 10 clocks with Zero '0' data. At the end of the 10 leading clocks the start character (11010) is sent and this is followed by the data. At the end of the data the end character is sent followed by the LCR. Finally 10 trailing clocks are sent and the card present line is raised.
 The data bit duration is approximately 330uS. The approximate clock duration is 110uS. Because of the symmetry data can be clocked off either the rising or falling edge of the clock.

Dimensions (Top View) (mm)



	ID-0/ID-2wr			ID-10/ID-12wr			ID-15/ID-20wr		
	Nom.	Min.	Max.	Nom.	Min.	Max.	Nom.	Min.	Max.
A	12.0	11.6	12.4	12.0	11.5	12.4	12.0	11.6	12.4
B	8.0	7.6	8.4	8.0	7.6	8.4	8.0	7.6	8.4
C	15.0	14.6	15.4	15.0	14.6	15.4	15.0	14.6	15.4
D	20.5	20.0	21.5	25.3	24.9	25.9	40.3	40.0	41.0
E	18.5	18.0	19.2	20.3	19.8	20.9	27.8	27.5	28.6
F	14.0	13.0	14.8	16.3	15.8	16.9	22.2	21.9	23.1
G	22.0	21.6	22.4	26.4	26.1	27.1	38.5	38.2	39.2
P	2.0	1.8	2.2	2.0	1.8	2.2	2.0	1.8	2.2
H	5.92	5.85	6.6	6.0	5.8	6.5	6.8	6.7	7.0
J	9.85	9.0	10.5	9.9	9.40	10.5	9.85	9.4	10.6
W	0.66	0.62	0.67	0.66	0.62	0.67	0.66	0.62	0.67

Note - measurements do not include any burring of edges.
 NOTICE - Innovated Devices reserve the right to change these specifications without prior notice.

Designing Coils for ID2

The recommended Inductance is 1.08mH to be used with an internal tuning capacitor of 1n5. In general the bigger the antenna the better, provided the reader is generating enough field strength to excite the tag. The ID-2 is relatively low power so a maximum coil size of 15x15cm is recommended if it is intended to read ISO cards. If the reader is intended to read glass tags the maximum coil size should be smaller, say 10x10cm.

There is a science to determine the exact size of an antenna but there are so many variables that in general it is best to get a general idea after which a degree of 'Try it and see' is unavoidable.

If the reader is located in a position where there is a lot of heavy interference then less range cannot be avoided. In this situation the coil should be made smaller to increase the field strength and coupling.

It is difficult to give actual examples of coils for hand winding because the closeness and tightness of the winding will significantly change the inductance. A professionally wound coil will have much more inductance than a similar hand wound coil.

For those who want a starting point into practical antenna winding it was found that 63 turns on a 120mm diameter former gave an inductance of 1.08mH. For those contemplating adding an additional tuning capacitor it was found that 50 turns on a 120mm diameter former gave 700uH. The wire diameter is not important.

Anybody who wishes to be more theoretical we recommend a trip to the Microchip Website where we found an application sheet for Loop Antennas.

<http://www.microchip.com/dsw/enr/AppNotes/00831b.pdf>

The Tuning Capacitor

It is recommended that the internal 1n5 capacitor is used for tuning, however a capacitor may be also be added externally. The combined capacitance should not exceed 2n7. Do not forget that the choice of tuning capacitor can also substantially affect the quality of your system. The Id12 is basically an ID2 with an internal antenna. The loss in an ID12 series antenna is required to be fairly high to limit the series current. A low Q will hide a lot of the shortcomings of the capacitor, but for quality and reliability and repeatability the following capacitors are recommend.

Polypropylene	Good Readily available. Ensure AC voltage at 125kHz is sufficient.
COG/NPO	Excellent. Best Choice
Silver Mica	Excellent but expensive
Polycarbonate	Good Readily available. Ensure AC voltage at 125kHz is sufficient.

Voltage Working.

A capacitor capable of withstanding the RMS voltage at 125kHz MUST be chosen. The working voltage will depend on the coil design. I suggest the designer start with rugged 1n5 Polypropylene 630v capacitor to do his experiments and then come down to a suitable size/value. The capacitor manufacturer will supply information on their capacitors. Do not simply go by the DC voltage. This means little. A tolerance of 2% is preferable. A tolerance of 5% is acceptable.

Fine Tuning

We recommend using an oscilloscope for fine-tuning. Connect the oscilloscope to observe the 125kHz AC voltage across the coil. Get a sizeable piece of ferrite and bring it up to the antenna loop. If the voltage increases then you need more inductance (or more capacitance). If the voltage decreases as you bring the ferrite up to the antenna then the inductance is too great. If you have no ferrite then a piece of aluminum

sheet may be used for testing in a slightly different way. Opposing currents will flow in the aluminum and it will act as a negative inductance. If the 125kHz AC voltage increases as the aluminum sheet approaches the antenna then the inductance is too high. Note it may be possible that the voltage will first maximize then decrease. This simply means that you are near optimum tuning. If you are using ferrite then the coil is a little under value and if you are using an aluminum sheet then the coil is a over under value.

ID Innovations

Advanced Digital Radio Technology

----*Better by Design*

Features

- High-performance, Low-power Atmel[®] AVR[®] 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single-clock 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 Lock 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 Lock 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

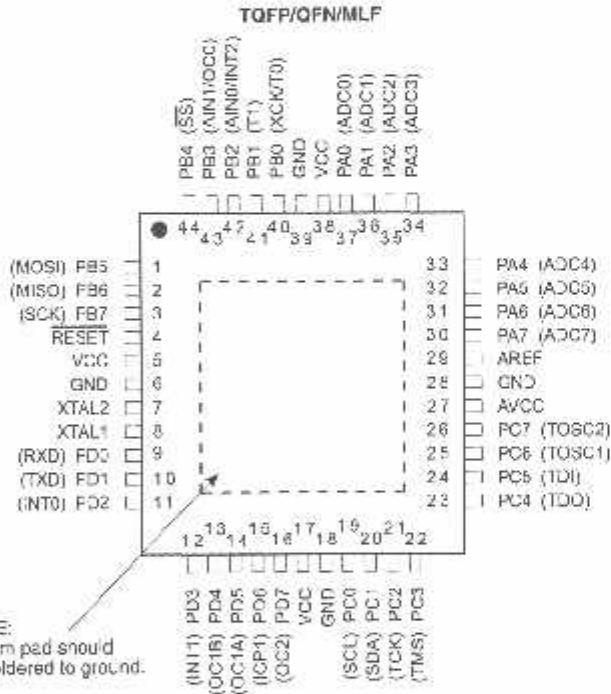
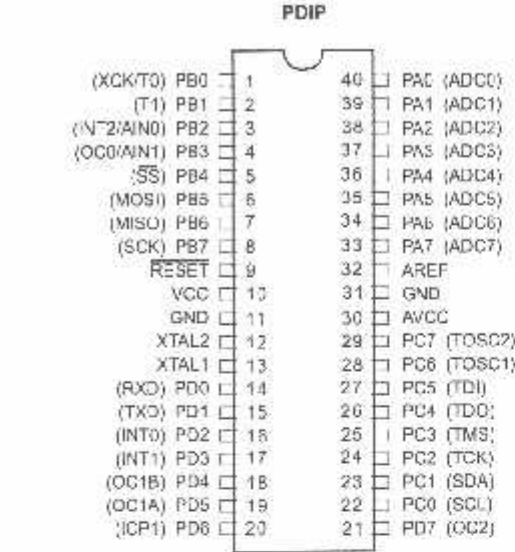
ATmega16
ATmega16L

Rev 2468T-AVR-07/10



Pin Configurations

Figure 1. Pinout ATmega16



Disclaimer

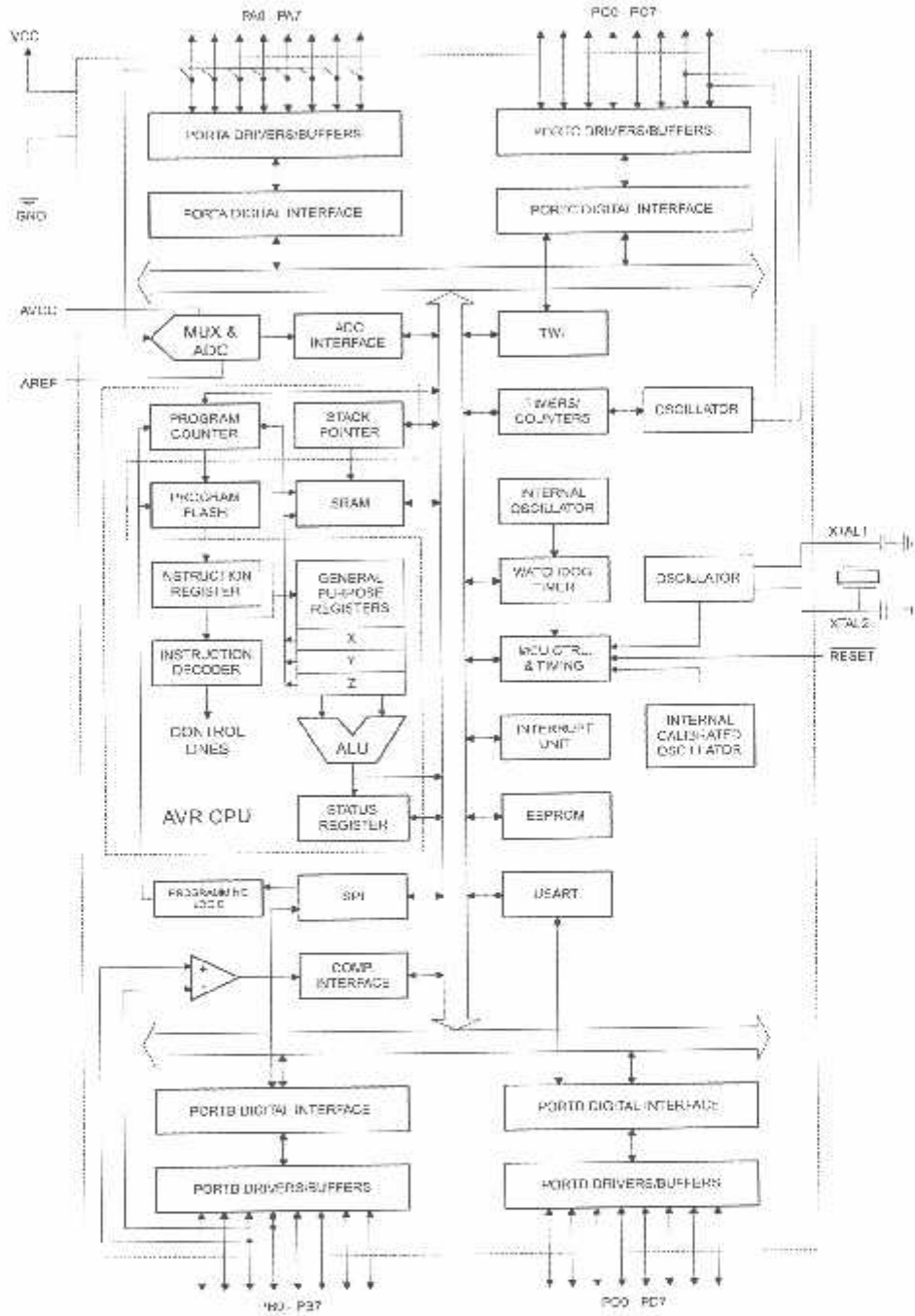
Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

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



The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega16 provides the following features: 16 Kbytes of In-System Programmable Flash Program memory with Read-While-Write capabilities, 512 bytes EEPROM, 1 Kbyte SRAM, 32 general purpose I/O lines, 32 general purpose working registers, a JTAG interface for Boundary-scan, On-chip Debugging support and programming, three flexible Timer/Counters with compare modes, Internal and External Interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, an 8-channel, 10-bit ADC with optional differential input stage with programmable gain (TQFP package only), a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and six software selectable power saving modes. The Idle mode stops the CPU while allowing the USART, Two-wire interface, A/D Converter, SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next External Interrupt or Hardware Reset. In Power-save mode, the Asynchronous Timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except Asynchronous Timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run.

The device is manufactured using Atmel's high density nonvolatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega16 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The ATmega16 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

Pin Descriptions

CC	Digital supply voltage.
ND	Ground.
Port A (PA7..PA0)	Port A serves as the analog inputs to the A/D Converter. Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B (PB7..PB0)

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B also serves the functions of various special features of the ATmega16 as listed on page 58.

Port C (PC7..PC0)

Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PC5(TDI), PC3(TMS) and PC2(TCK) will be activated even if a reset occurs.

Port C also serves the functions of the JTAG interface and other special features of the ATmega16 as listed on page 61.

Port D (PD7..PD0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port D also serves the functions of various special features of the ATmega16 as listed on page 63.

RESET

Reset Input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 38. Shorter pulses are not guaranteed to generate a reset.

XTAL1

Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting Oscillator amplifier.

AVCC

AVCC is the supply voltage pin for Port A and the A/D Converter. It should be externally connected to V_{CC} , even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter.

AREF

AREF is the analog reference pin for the A/D Converter.



Resources

A comprehensive set of development tools, application notes and datasheets are available for download on <http://www.atmel.com/avr>.

Data Retention

Reliability Qualification results show that the projected data retention failure rate is much less than 1 PPM over 20 years at 85°C or 100 years at 25°C.

About Code Examples

This documentation contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C Compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C Compiler documentation for more details.



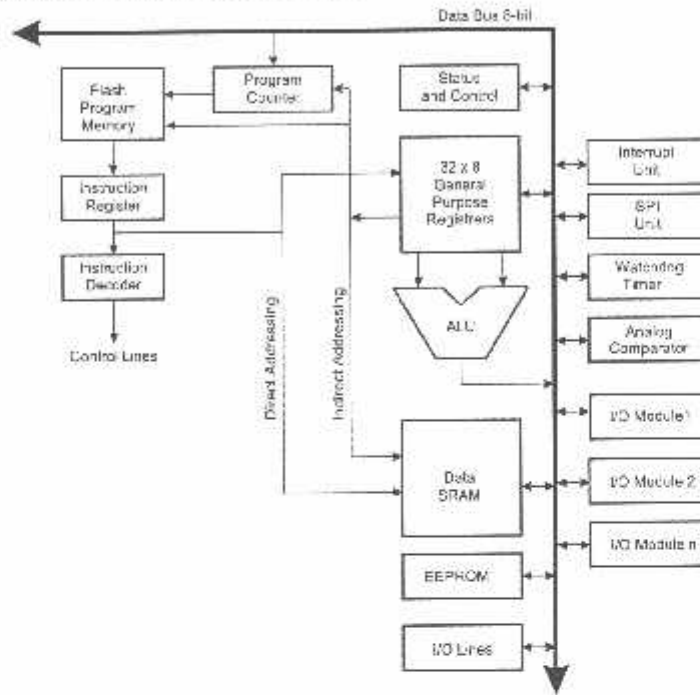
VR CPU Core

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.

Program flow is provided by conditional and unconditional jump and call instructions, able to directly address the whole address space. Most AVR instructions have a single 16-bit word format. Every program memory address contains a 16-bit or 32-bit instruction.

Program Flash memory space is divided in two sections, the Boot program section and the Application Program section. Both sections have dedicated Lock bits for write and read/write protection. The SPM instruction that writes into the Application Flash memory section must reside in the Boot Program section.

During interrupts and subroutine calls, the return address Program Counter (PC) is stored on the Stack. The Stack is effectively allocated in the general data SRAM, and consequently the Stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the SP in the reset routine (before subroutines or interrupts are executed). The Stack Pointer SP is read/write accessible in the I/O space. The data SRAM can easily be accessed through the five different addressing modes supported in the AVR architecture.

The memory spaces in the AVR architecture are all linear and regular memory maps.

A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the Status Register. All interrupts have a separate interrupt vector in the interrupt vector table. The interrupts have priority in accordance with their interrupt vector position. The lower the interrupt vector address, the higher the priority.

The I/O memory space contains 64 addresses for CPU peripheral functions as Control Registers, SPI, and other I/O functions. The I/O Memory can be accessed directly, or as the Data Space locations following those of the Register File, \$20 - \$5F.

ALU – Arithmetic Logic Unit

The high-performance AVR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations between general purpose registers or between a register and an immediate are executed. The ALU operations are divided into three main categories – arithmetic, logical, and bit-functions. Some implementations of the architecture also provide a powerful multiplier supporting both signed/unsigned multiplication and fractional format. See the "Instruction Set" section for a detailed description.

Status Register

The Status Register contains information about the result of the most recently executed arithmetic instruction. This information can be used for altering program flow in order to perform conditional operations. Note that the Status Register is updated after all ALU operations, as specified in the Instruction Set Reference. This will in many cases remove the need for using the dedicated compare instructions, resulting in faster and more compact code.

The Status Register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt. This must be handled by software.

The AVR Status Register – SREG – is defined as:

Bit:	7	6	5	4	3	2	1	0	
	I	T	H	S	V	N	Z	C	SREG
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	C	0	0	

• Bit 7 – I: Global Interrupt Enable

The Global Interrupt Enable bit must be set for the interrupts to be enabled. The individual interrupt enable control is then performed in separate control registers. If the Global Interrupt Enable Register is cleared, none of the interrupts are enabled independent of the individual interrupt enable settings. The I-bit is cleared by hardware after an interrupt has occurred, and is set by the RETI instruction to enable subsequent interrupts. The I-bit can also be set and cleared by the application with the SEI and CLI instructions, as described in the instruction set reference.



- **Bit 6 – T: Bit Copy Storage**

The Bit Copy instructions BLD (Bit Load) and BST (Bit Store) use the T-bit as source or destination for the operated bit. A bit from a register in the Register File can be copied into T by the BST instruction, and a bit in T can be copied into a bit in a register in the Register File by the BLD instruction.

- **Bit 5 – H: Half Carry Flag**

The Half Carry Flag H indicates a Half Carry in some arithmetic operations. Half Carry is useful in BCD arithmetic. See the "Instruction Set Description" for detailed information.

- **Bit 4 – S: Sign Bit, $S = N \oplus V$**

The S-bit is always an exclusive or between the Negative Flag N and the Two's Complement Overflow Flag V. See the "Instruction Set Description" for detailed information.

- **Bit 3 – V: Two's Complement Overflow Flag**

The Two's Complement Overflow Flag V supports two's complement arithmetics. See the "Instruction Set Description" for detailed information.

- **Bit 2 – N: Negative Flag**

The Negative Flag N indicates a negative result in an arithmetic or logic operation. See the "Instruction Set Description" for detailed information.

- **Bit 1 – Z: Zero Flag**

The Zero Flag Z indicates a zero result in an arithmetic or logic operation. See the "Instruction Set Description" for detailed information.

- **Bit 0 – C: Carry Flag**

The Carry Flag C indicates a carry in an arithmetic or logic operation. See the "Instruction Set Description" for detailed information.