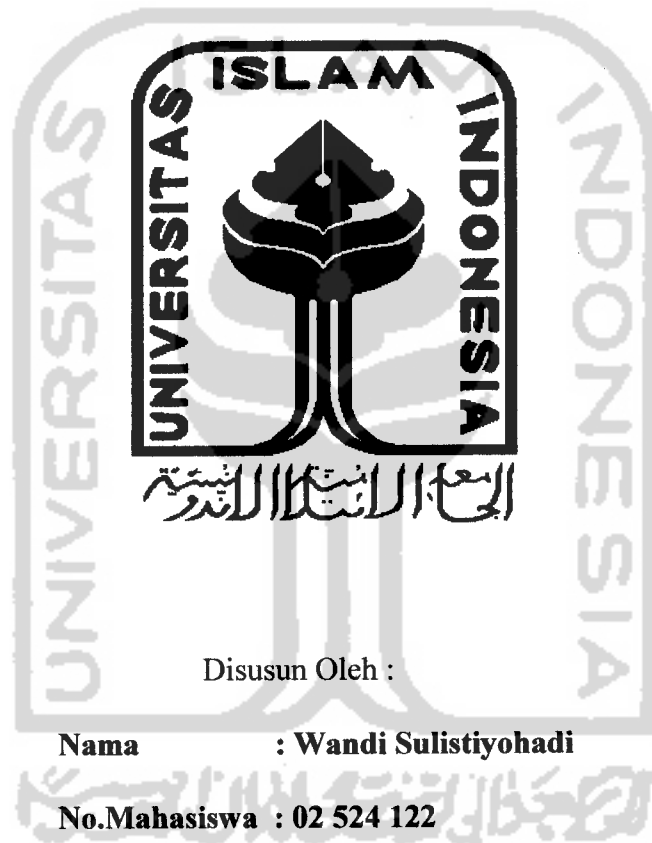


**PERANCANGAN DEKRIPTOR - SIMETRIS SINYAL AUDIO
BERBASIS MIKROKONTROLER ATMEGA 16**

TUGAS AKHIR

Diajukan sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Pada Jurusan Teknik Elektro
Fakultas Teknologi Industri Universitas Islam Indonesia



Disusun Oleh :

Nama : Wandi Sulistiyohadi

No.Mahasiswa : 02 524 122

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

2007

LEMBAR PENGESAHAN PEMBIMBING

PERANCANGAN DEKRIPTOR - SIMETRIS SINYAL AUDIO BERBASIS MIKROKONTROLER ATMEGA16

TUGAS AKHIR


Disusun oleh :

Nama : Wandi Sulistiyohadi

No. Mahasiswa : 02 524 122

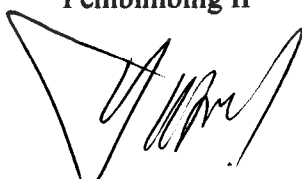
Yogyakarta, Agustus 2007

Pembimbing I



(Ir. Hj. Budi Astuti, MT)

Pembimbing II



(Yusuf Aziz Amrullah, ST)

LEMBAR PENGESAHAN PENGUJI
PERANCANGAN DECRYPTOR – SIMETRIS SINYAL AUDIO
BERBASIS ATMEGA 16

Oleh :

Nama : Wandi Sulistiyohadi

No. Mahasiswa : 02 524 122

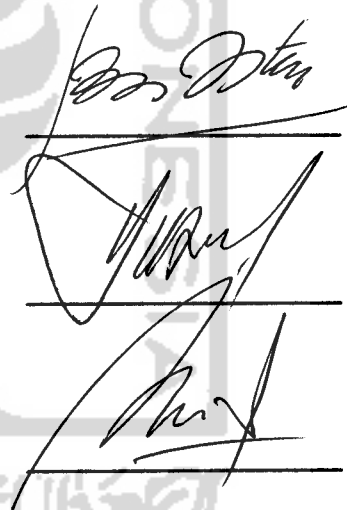
Telah dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat untuk
Memperoleh Gelar Sarjana Teknik Elektro
Fakultas Teknologi Industri Universitas Islam Indonesia
Yogyakarta, 15 Agustus 2007

Tim Penguji :

Ir. Hj. Budi Astuti, MT
Ketua

Yusuf Aziz Amrullah, ST
Anggota I

Dwi Ana Ratna Wati, ST
Anggota II



Mengetahui,
Ketua Jurusan Teknik Elektro
Fakultas Teknologi Industri
Universitas Islam Indonesia



(Eko Yuwono, ST, MSc)

ABSTRAK

Masalah keamanan dan kerahasiaan data merupakan salah satu aspek penting dari suatu informasi. Dalam hal ini sangat terkait dengan betapa pentingnya informasi tersebut dikirim dan diterima oleh orang yang berkepentingan. Informasi akan tidak berguna lagi apabila di tengah jalan informasi itu disadap atau dibajak oleh orang yang tidak berhak. Untuk mengantisipasi berbagai masalah kebocoran informasi rahasia, maka diperlukan sebuah sistem yang *real time*, untuk mengamankan informasi yang dikirimkan tersebut. Dekriptor merupakan kebalikan dari enkriptor yaitu mengembalikan informasi yang telah diacak menjadi bentuk informasi semula, Perancangan dekriptor simetris signal audio berbasis ATmega 16 merupakan sebuah alat yang dirancang untuk menterjemahkan sinyal yang dikirim oleh enkriptor yaitu berupa *chipper digital audio* untuk menjadi sinyal audio. Perancangan dekriptor simetris signal audio berbasis ATmega 16 mengimplementasikan sebuah logika sebagai kunci dan password untuk mendekripsi sinyal. Dalam perancangannya *chipper digital audio* digunakan sebagai sinyal masukan yang kemudian sinyal didekripsi dengan menggunakan dua kunci yaitu mode 1 dan mode 2 dan 2 buah *password* kemudian hasil keluaran dari proses dekripsi berupa sinyal digital audio selanjutnya sinyal diolah lagi oleh DAC agar menjadi sinyal aslinya yaitu berupa sinyal audio. Dari hasil penelitian sinyal *chipper digital audio* mampu didekripsi menjadi sinyal audio.



MOTTO

“Alloh akan meninggikan orang – orang yang beriman dan orang – orang yang diberi ilmu pengetahuan beberapa derajat”

(QS. Al Mujadalah : 11)

“Alloh tidak akan membebani seseorang kecuali sepadan dengan kemampuannya “

(QS. Al Baqarah : 286)

“Barang siapa yang menempuh jalan untuk mencari ilmu maka Alloh akan memudahkan baginya jalan menuju syurga”

(HR. Muslim & Abu hurairah)

“Bahwa tiada yang ia dapatkan, kecuali yang ia usahakan “

(Surat AN Najm : 39)

HALAMAN PERSEMBAHAN

Tugas akhir ini kupersembahkan untuk kedua orang tuaku yang telah memberikan seluruh apa yang dimilikinya demi masa depanku

“Ayahanda Tercinta (Sugeng), atas bimbingan, ketauladanan, pengorbanan, kesabaran dan do'a ayahanda adalah motivator utama dalam hidupku”

“Ibunda Tersayang (Yatimah), wujud kasih sayangmu, kesabaran, pengorbanan, ketabahan dan doa serta keselarasan hidup yang telah ibunda tunjukkan telah mendewasakanmu”

*“Adik-adikku tercinta (Wiwit, Nita, Inung, Sally) yang selalu membuat hidupku begitu bersemangat
Ulurkan terus tangan kita untuk selalu membahagiakan bapak dan ibu kita tercinta”*

“Seluruh keluarga besarku yang selalu memberikan do'a semangat dan dukungan”

*“Kekasihku tercinta yang selalu memberikan warna baru dalam hidupku
Doa, semangat, kesabaran, dan kedewasaanmu selalu menjadi motivasi dalam hidupku”*

KATA PENGANTAR



Assalamu'alaikum Wr. Wb.

Alhamdulillah, segala puji dan syukur ditunjukkan hanya bagi Allah SWT, sang Esa pemilik alam semesta. Semoga kesejahteraan diberikan bagi rosul-Nya, Muhammad SAW, atas rahmat dan taufik-Nya penulis dapat menyelesaikan tugas akhir dengan judul **“Perancangan Dekripsi simetrik Signal Audio Berbasis Mikrokontrol ATMEGA 16”** dengan lancar.

Adapun maksud dan tujuan penyusunan tugas akhir ini adalah untuk melengkapi salah satu syarat dalam menempuh gelar sarjana pada Jurusan Teknik Elektro Universitas Islam Indonesia, Yogyakarta. Disamping itu untuk menambah pengetahuan terhadap ilmu yang telah dipelajari di bangku perkuliahan untuk diterapkan di masyarakat.

Pada kesempatan ini dengan segala rasa syukur dan kerendahan hati penulis sampaikan ucapan terima kasih yang sebesar-besarnya kepada :

1. Alloh SWT dan junjungan Nabi Besar Muhammad SAW.
2. Bpk Tito Yuwono, ST, MSc., selaku Ketua Jurusan Teknik Elektro Universitas Islam Indonesia
3. Ibu Hj Budi Astuti, Ir, MT. Selaku dosen pembimbing I.
4. Bpk Yusuf Aziz Amirulloh, ST. Selaku dosen pembimbing II.
5. Segenap dosen dan karyawan Fakultas Teknologi Industri UII, yang bersedia dengan sabar membantu dan berbagi ilmunya.

6. Seluruh Staf TU dan Perpustakaan Jurusan yang telah banyak membantu segala kelancaran penulis..
7. Ayahanda dan Ibunda serta adik-adikku tercinta terima kasih atas semua kasih sayang, air mata, perhatian, nasehat, bimbingan, pengorbanan, keikhlasan, dan doa-doa malamnya semoga, alloh SWT membalas semuanya.
8. Kekasihku tercinta, terima kasih atas kasih sayang, doa, semangat, dan kesabarannya.
9. Seluruh mahasiswa jurusan Teknik Elektro UII, khususnya Angkatan 2002.
10. Teman-teman kost yang telah memberikan suasana baru dalam kehidupan anak-anak kost.
11. Seluruh saudaraku dan teman-teman yang telah memberi motivasi, dorongan serta doa bagi penulis.

Dalam penyelesaian Tugas Akhir ini penulis menyadari bahwa masih terdapat kekurangan dan kesalahan baik dalam penulisan maupun bentuk lainnya. Oleh karena itu, kritik dan saran yang bersifat membangun akan penulis terima dengan harapan penulisan berikutnya menjadi lebih baik.

Akhir kata penulis hanya dapat memanjatkan doa kehadiran Allah SWT, kiranya dapat melimpahkan rahmat dan kesejahteraan kepada semua pihak atas segala budi baik yang telah diberikan kepada penulis. Amin...

Wassalamu'alaikum Wr. Wb

Yogyakarta, Agustus 2007

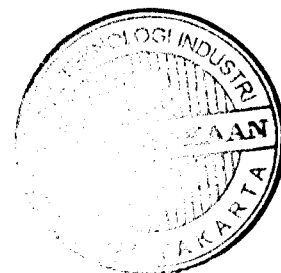
Wandi Sulistiyohadi



DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN PEMBIMBING.....	ii
HALAMAN PENGESAHAN PENGUJI.....	iii
ABSTRAK.....	iv
HALAMAN MOTTO.....	v
HALAMAN PERSEMBAHAN.....	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xiv
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah.....	2
1.4. Manfaat Penelitian.....	3
1.4. Tujuan Penelitian.....	3
1.5. Sistematika Penulisan.....	3
BAB II DASAR TEORI.....	5
2.1. Pengertian Deskripsi.....	5
2.2. Algoritma Deskripsi.....	6

2.2.1 Kriptografi.....	6
2.3. Mikrokontroler AVR ATmega 16.....	9
2.3.1. Fitur ATmega 16.....	11
2.3.2. Arsitektur ATmega 16.....	12
2.3.3. Konfigurasi Pin ATmega 16.....	13
2.3.4 Peta Memori.....	15
2.3.5. Status Register.....	17
2.3.6. Interupsi.....	18
2.3.7. Serial Peripheral Interface.....	21
2.4. Logika.....	23
2.4.1. Operasi NOT.....	24
2.4.2. Operasi AND.....	24
2.4.3. Operasi OR dan XOR.....	25
2.5. DAC.....	26
BAB III PERANCANGAN SISTEM.....	28
3.1. Perancangan Sistem.....	28
3.2. Perancangan Hardware.....	31
3.2.1. Perancangan Port Mikrokontroler ATmega 16.....	31
3.2.2. Perancangan Osilator dan reset.....	33
3.2.3 Perancangan Rangkaian DAC.....	34
3.2.4 Perancangan Rangkaian Op Amp.....	34
3.2.5 Perancangan Catu daya.....	35
3.3. Perancangan Perangkat Lunak.....	36



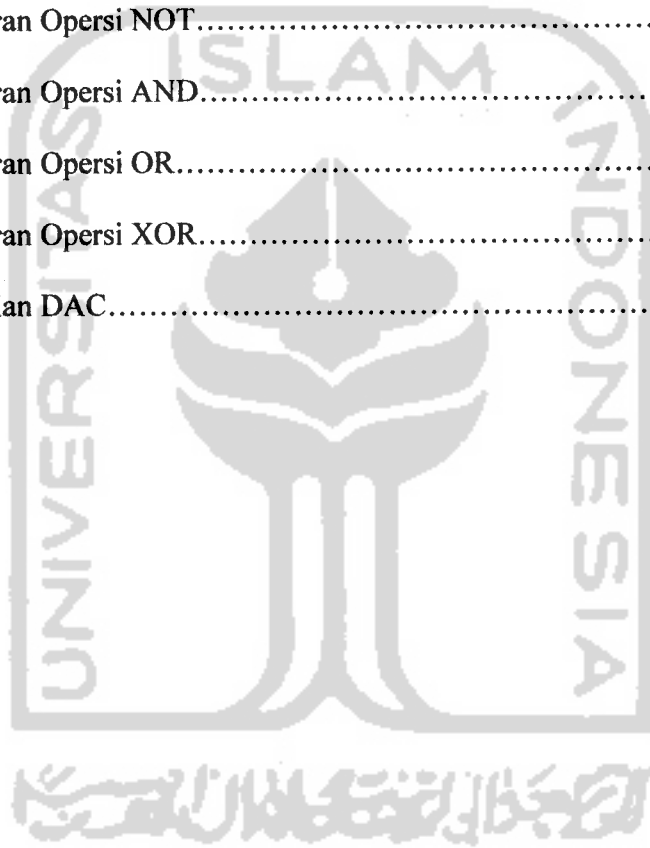
3.3.1 Perancangan Teknik Dekripsi.....	36
3.3.2. Program Utama.....	38
3.3.3. Program Dekripsi Mode 1.....	39
3.3.4. Program Dekripsi Mode 2.....	40
BAB IV PENGUJIAN DAN PEMBAHASAN.....	42
4.1. Pengujian DAC.....	42
4.2. Pengujian Dekripsi dengan Sinyal Audio.....	44
4.2.1 Pengujian dengan Mode Normal.....	44
4.2.2 Pengujian dengan Mode 1.....	44
4.2.3. Pengujian dengan Mode 2.....	45
4.3.. Pengujian dengan menggunakan AFG.....	47
4.3.1 Pengujian dengan Mode Normal.....	47
4.3.2. Pengujian dengan Mode 1.....	48
4.3.3. Pengujian dengan Mode 2.....	49
BAB V PENUTUP.....	51
5.1. Kesimpulan.....	51
5.2. Saran.....	52

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR TABEL

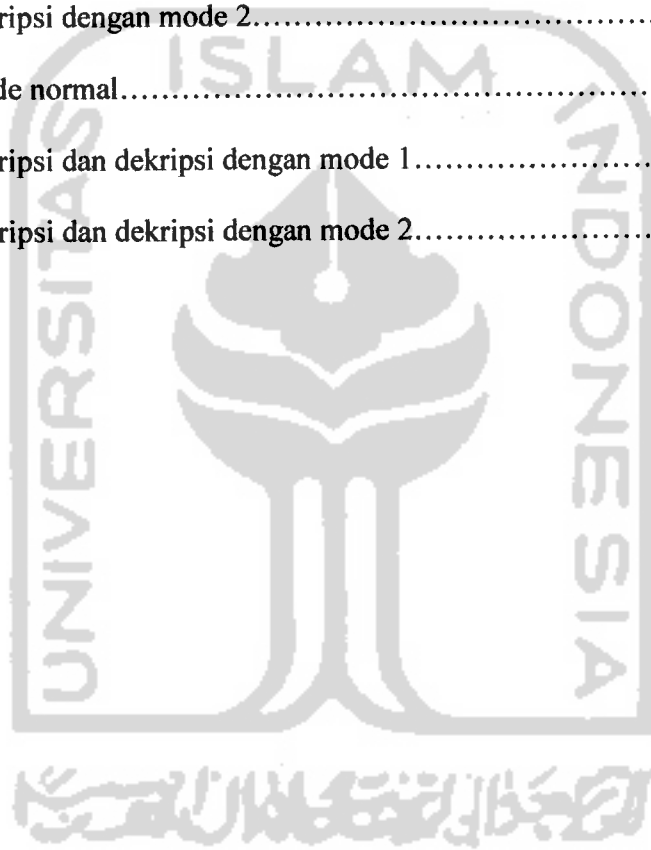
Tabel 2.1. Setting kondisi menyebabkan interupsi eksternal 1.....	19
Tabel 2.2. Setting kondisi menyebabkan interupsi eksternal 0.....	20
Tabel 2.3. Macam sumber Interupsi pada ATmega 16.....	21
Tabel 2.4. Konfigurasi pin master dan slave saat komunikasi SPI.....	23
Tabel 2.5. Nilai kebenaran Operasi NOT.....	24
Tabel 2.6. Nilai kebenaran Operasi AND.....	25
Tabel 2.7. Nilai kebenaran Operasi OR.....	25
Tabel 2.8. Nilai kebenaran Operasi XOR.....	25
Tabel 4.1 .Hasil Pengujian DAC.....	42



DAFTAR GAMBAR

Gambar 2.1 Proses Enkripsi dan Dekripsi Simetris.....	7
Gambar 2.2 Proses Enkripsi dan Dekripsi ASimetris.....	8
Gambar 2.3. Gambar Blok diagram fungsional ATmega 16.....	12
Gambar 2.4. Mikrokontrol AVR ATmega 16.....	15
Gambar 2.5 Konfigurasi memori data ATmega 16.....	16
Gambar 2.6 Memori program AVR ATmega 16.....	17
Gambar 2.7 Status Register ATmega 16.....	17
Gambar 2.8 Register MCUCR.....	19
Gambar 2.9. General Input Control Register.....	20
Gambar 2.10. Perawatan SPI Saat menggunakan 1 Slave.....	23
Gambar 2.11. Konfigurasi pin DAC 0808.....	26
Gambar 3.1. Diagram blok system Dekripsi Signal Audio.....	28
Gambar 3.2. Rangkaian LED.....	31
Gambar 3.3. Rangkaian SPI dan Mode.....	31
Gambar 3.4. Rangkaian Output.....	32
Gambar 3.5. Rangkaian Input.....	32
Gambar 3.6. Rangkaian osilator dan reset.....	33
Gambar 3.7. Rangkaian DAC.....	34
Gambar 3.8. Simbol penguat Op amp.....	35
Gambar 3.9. Rangkaian Catu daya.....	36
Gambar 3.10. Flowchart program utama.....	39
Gambar 3.10.Flowchart proses dekripsi mode 1.....	40

Gambar 3.11. Flowchart proses dekripsi mode 2.....	41
Gambar 4.1. Sinyal Audio tanpa enkripsi.....	44
Gambar 4.2. Sinyal audio tanpa dekripsi.....	44
Gambar 4.3. Sinyal enkripsi dengan mode 1.....	44
Gambar 4.4. Sinyal dekripsi dengan mode 1.....	45
Gambar 4.5. Sinyal enkripsi dengan mode 2.....	46
Gambar 4.6. Sinyal dekripsi dengan mode 2.....	46
Gambar 4.7. Sinyal mode normal.....	47
Gambar 4.8. Sinyal enkripsi dan dekripsi dengan mode 1.....	48
Gambar 4.9. Sinyal enkripsi dan dekripsi dengan mode 2.....	50



BAB I

PENDAHULUAN

1.1 Latar Belakang

Informasi adalah kekuatan tersendiri yang tak dapat diremehkan bagi yang dapat memanfaatkannya. Di saat Perang antara Jepang dengan Amerika, sangat terlihat bagaimana mereka saling memecahkan kata sandi lawannya. Pecahnya kata sandi akan menghasilkan informasi yang sangat penting bagi yang memperolehnya, bisa menjadi bencana bagi pemilik informasi tersebut.

Masalah keamanan dan kerahasiaan data merupakan salah satu aspek penting dari suatu informasi. Dalam hal ini sangat terkait dengan betapa pentingnya informasi tersebut dikirim dan diterima oleh orang yang berkepentingan. Informasi akan tidak berguna lagi apabila di tengah jalan informasi itu disadap atau dibajak oleh orang yang tidak berhak. Untuk mengantisipasi berbagai masalah kebocoran informasi rahasia, maka diperlukan sebuah sistem yang *real time*, untuk mengamankan informasi yang dikirimkan tersebut. Untuk keperluan pengamanan data tersebut dapat dilakukan sebuah metode yang dikenal sebagai metode Enskripsi data. Untuk menjalankan metode tersebut, maka diperlukan dekriptor.

Beberapa contoh aplikasi sistem dekripsi data yaitu dekripsi data pada Sistem komunikasi *satelite*, komunikasi pada ATM, komunikasi pada kepentingan Militer, komunikasi kepentingan Negara, atau komunikasi rahasia perusahaan tertentu, dan tentunya masih banyak aplikasi lainnya.

Seperti pengalaman negara maju yang ada, dengan berjalannya waktu, maka diharapkan akan meningkat pula sistem pengamanan yang ada. Mengingat pentingnya sistem ini untuk digunakan, baik sekarang maupun kelak dikemudian hari, maka akan sangat baik apabila segera dapat direalisasikan metode dekripsi yang ada. Selanjutnya, penelitian ini, diharapkan akan dapat memberikan inspirasi pada penelitian berikutnya. Nantinya diharapkan akan dapat dikembangkan menjadi penelitian yang berkelanjutan menuju kesempurnaan sistem.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat dirumuskan masalah :

1. Bagaimanakah membuat rancangan *Decryptor Simetrik Signal Audio* berbasis *Mikrokontroler ATmega16*.
2. Bagaimanakah cara pemrosesan sinyal agar sinyal bisa kembali seperti bentuk aslinya.

1.3. Batasan Masalah

Dalam penelitian ini diberikan beberapa batasan masalah sebagai berikut :

1. Mikrokontroler yang digunakan dalam penelitian ini adalah mikrokontroler AVR ATmega16.
2. Metode Dekripsi yang digunakan adalah metode Dekripsi simetris.
3. Jumlah mode dekripsi yang digunakan adalah 2 mode.
4. Data yang di dekripsi adalah *chipper digital audio*.

5. Pengujian yang diberikan pada sistem ini berupa *Chipeper digital audio* dengan frekuensi 1 KHz -3 KHz..
6. Mengingat dalam penelitian ini diperlukan kecepatan akses yang tinggi untuk menuju pada *real time*, maka tidak digunakan *display*.

1.4. Manfaat Penelitian

1. Memberi pemahaman dalam perncangan alat dengan menggunakan mikrokontrol ATmega16.
2. Sistem *Decryptor - Simetrik Sinyal Audio berbasis Mikrokontroler ATMEGA16* mampu menterjemahkan data sinyal yang telah di enkripsi.
3. Memberi kemampuan untuk menyimpan kerahasiaan suatu data.

1.5. Tujuan Penelitian.

Untuk mengantisipasi/melindungi berbagai masalah kebocoran informasi data yang sifatnya rahasia, baik itu informasi berupa teks maupun suara.

1.6. Sistematika Penulisan

Sistematika penulisan tugas akhir ini terdiri dari 5 bab bagian isi laporan, dengan penjelasan bab sebagai berikut

BAB I : Pendahuluan

Meliputi pembahasan Latar Belakang Masalah, Pemecahan Masalah, Patasan Masalah, Maksud dan Tujuan Penelitian, serta Sistematika Penulisan.

BAB II : Landasan teori

Dalam bab ini akan membahas semua teori yang diperlukan dalam perancangan hardware maupun software.

BAB III : Perancangan sistem

Bagian ini menjelaskan metode-metode perancangan yang digunakan, cara mensimulasikan rancangan dan pengujian sistem yang telah dibuat, pembagian fungsi kerja dalam diagram blok serta berisi lebih terperinci tentang apa yang telah disampaikan pada proposal Tugas Akhir ini.

BAB IV : Pengujian dan analisa data

Bab ini membahas tentang hasil pengujian dan analisis dari sistem yang dibuat dibandingkan dengan dasar teori sistem atau sistem yang lain yang dapat dijadikan sebagai pembandingan .

BAB V : Penutup

Bab ini berisi mengenai kesimpulan dari penelitian yang dilakukan serta saran untuk perbaikan sistem agar dalam pengembangannya didapat sistem yang lebih baik lagi.

BAB II

DASAR TEORI

2.1. Pengertian Dekripsi

Masalah keamanan merupakan salah satu aspek penting dari sebuah system informasi. Salah satu hal yang penting dalam komunikasi menggunakan komputer dan dalam jaringan komputer untuk menjamin keamanan pesan, data, ataupun informasi adalah enkripsi. Disini enkripsi dapat diartikan sebagai kode atau *chipper*. Sebuah sistem pengkodean menggunakan suatu tabel atau kamus yang telah didefinisikan untuk kata dari informasi atau yang merupakan bagian dari pesan, data, atau informasi yang di kirim. Sebuah *chipper* menggunakan suatu algoritma yang dapat mengkodekan semua aliran data (*stream*) bit dari suatu pesan asli (*plaintext*) menjadi cryptogram (*chipertext*) yang tidak di mengerti. Proses sebaliknya, untuk mengubah ciphertext menjadi *plaintext*, atau menterjemahkan kembali informasi yang di enkripsi menjadi sebuah informasi yang dapat dikenali, disebut dekripsi (*decryption*). Menurut terminologi yang lebih tepat untuk proses ini adalah “*decipher*”. *Cryptanalysis* adalah seni dan ilmu untuk memecahkan ciphertext tanpa bantuan kunci. *Cryptanalyst* adalah pelaku atau praktisi yang menjalankan cryptanalysis. *Cryptology* merupakan gabungan dari *cryptography* dan *cryptanalysis*.

2.2. Algoritma Dekripsi

2.2.1. Kriptografi

Kriptografi (*cryptography*) merupakan ilmu dan seni untuk menjaga pesan agar aman. Kriptografi (*Cryptography*) berasal dari bahasa Yunani yaitu “*Crypto*” berarti “*secret*” (rahasia) dan “*graphy*” berarti “*writing*” (tulisan). Para pelaku atau praktisi kriptografi disebut *cryptographers*. Sebuah algoritma kriptografik (*cryptographic algorithm*), disebut *cipher*, merupakan persamaan matematik yang digunakan untuk proses enkripsi dan dekripsi. Biasanya kedua persamaan matematik (untuk enkripsi dan dekripsi) tersebut memiliki hubungan matematis yang cukup erat.

Proses yang dilakukan untuk mengamankan sebuah pesan (yang disebut *plaintext*) menjadi pesan yang tersembunyi (disebut *ciphertext*) adalah enkripsi (*encryption*). *Ciphertext* adalah pesan yang sudah tidak dapat dibaca dengan mudah. Menurut ISO 7498-2, terminologi yang lebih tepat digunakan adalah “*encipher*”. Proses sebaliknya, untuk mengubah *ciphertext* menjadi *plaintext*, disebut dekripsi (*decryption*).

Untuk mengenkripsi dan mendekripsi data, Kriptografi menggunakan suatu algoritma (*chipper*) dan kunci (*key*). *Cipher* adalah fungsi matematika yang digunakan untuk mengenkripsi dan mendekripsi data. Sedangkan kunci merupakan sederetan bit yang diperlukan untuk mengenkripsi dan mendekripsi data.

Algoritma kriptografi modern tidak lagi mengandalkan keamanannya pada kerahasiaan algoritma tetapi kerahasiaan kunci. *Plaintext* yang sama bila disandikan dengan kunci yang berbeda akan menghasilkan *ciphertext* yang berbeda pula. Dengan demikian algoritma kriptografi dapat bersifat umum dan boleh diketahui oleh siapa saja, akan tetapi tanpa pengetahuan tentang kunci, data tersandi tetap saja tidak dapat

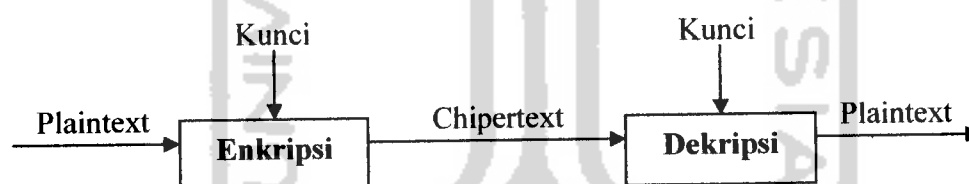
terpecahkan. Sistem kriptografi atau *Cryptosystem* adalah sebuah algoritma kriptografi ditambah semua kemungkinan plaintext, *ciphertext* dan kunci.

Berdasarkan kunci yang dipakai, algoritma kriptografi dapat dibedakan atas dua golongan, yaitu :

a. Kunci Simetris

Kunci Simetris adalah jenis kriptografi yang paling umum digunakan. Kunci untuk membuat pesan yang di sandikan sama dengan kunci untuk membuka pesan yang disandikan itu. Jadi pembuat pesan dan penerimanya harus memiliki kunci yang sama persis. Siapapun yang memiliki kunci tersebut termasuk pihak-pihak yang tidak diinginkan dapat membuat dan membongkar rahasia *ciphertext*. Contoh algoritma kunci simetris yang terkenal adalah DES (*Data Encryption Standard*).

Proses enkripsi-dekripsi algoritma kriptografi kunci simetris dapat dilihat pada gambar dibawah ini :



Gambar 2.1. Proses enkripsi dan dekripsi kunci simetris

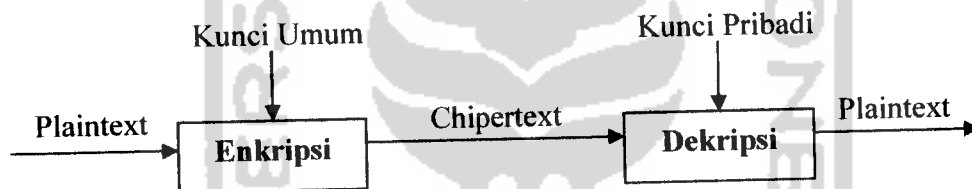
Algoritma kriptografi simetris dibagi menjadi 2 kategori yaitu algoritma aliran (*Stream Ciphers*) dan algoritma blok (*Block Ciphers*). Pada algoritma aliran, proses penyandiannya berorientasi pada satu bit atau satu *byte* data. Sedang pada algoritma blok, proses penyandiannya berorientasi pada sekumpulan bit atau *byte* data (per blok).



b. Kunci Asimetris

Kunci asimetris adalah pasangan kunci kriptografi yang salah satunya digunakan untuk proses enkripsi dan yang satu lagi untuk dekripsi. Semua orang yang mendapatkan kunci publik dapat menggunakannya untuk mengenkripsikan suatu pesan, data ataupun informasi, sedangkan hanya satu orang saja yang memiliki rahasia tertentu dalam hal ini kunci privat untuk melakukan pembongkaran terhadap sandi yang dikirim untuknya. Contoh algoritma terkenal yang menggunakan kunci asimetris adalah RSA.

Proses enkripsi-dekripsi algoritma kunci asimetris dapat dilihat pada gambar dibawah ini :



Gambar 2.2. Proses enkripsi dan dekripsi kunci simetris

Dasar matematis yang mendasari proses enkripsi dan dekripsi adalah relasi antara dua himpunan yaitu himpunan berisi elemen *plaintext* dan himpunan berisi elemen *ciphertext*. Enkripsi dan dekripsi merupakan fungsi tranformasi antara dua himpunan tersebut. Bila himpunan *plaintext* dinotasikan dengan P dan himpunan *ciphertext* dinotasikan dengan C, sedang fungsi enkripsi dinotasikan dengan E dan fungsi dekripsi dengan D maka proses enkripsi-dekripsi dapat dinyatakan dalam notasi matematis dengan $E(P) = C$ dan $D(C) = P$(2.1)

Karena proses enkripsi-dekripsi bertujuan memperoleh kembali data asal, maka :

$$D(E(P)) = P \dots\dots\dots(2.2)$$

Relasi antara himpunan *plaintext* dengan himpunan *ciphertext* harus merupakan fungsi korespondensi satu-satu (*one to one relation*). Hal ini merupakan keharusan untuk mencegah terjadinya ambiguitas dalam dekripsi yaitu satu elemen *ciphertext* menyatakan lebih dari satu elemen *plaintext*.

Pada metode kriptografi simetris atau konvensional digunakan satu buah kunci. Bila kunci dinotasikan dengan 'K' maka proses enkripsi-dekripsi metode kriptografi simetris dapat dinotasikan dengan :

$$E_k(P) = C \text{ dan } D_k(C) = P \dots\dots\dots(2.3)$$

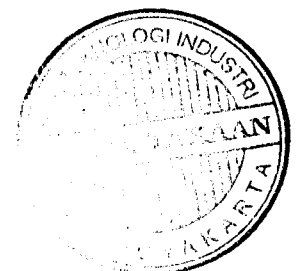
Dan keseluruhan sistem dinyatakan sebagai :

$$D_k(E_k(P)) = P \dots\dots\dots(2.4)$$

2.3 Mikrokontroler AVR ATmega 16

AVR merupakan sebuah mikrokontroler dengan arsitektur RISC (*Resource instruction set computer*) 8 bit yang dikeluarkan oleh Atmel Corporation, sampai saat ini mikrokontroler keluarga AVR adalah AT Tinyxx, AT Megaxxx, AT 90Sxxx.

AVR dikembangkan pertama kali oleh para peneliti dari Throncim, Norwegia sebelum mereka diakuisisi oleh Atmel di tahun 1995. Fitur inti AVR adalah 32 buah register identik 8-bit yang memiliki kemampuan dan sifat seperti sebuah *Accumulator*. AVR dikembangkan berdasarkan arsitektur Harvard, yang memisahkan antara memori memori program dan memori data. Keunggulan arsitektur Harvard adalah kecepatan eksekusi yang tinggi. Susunan intruksinya bertipe RISC, tetapi variasinya cukup banyak.



Dengan desain seperti itu, memungkinkan dihasilkan program yang efisien (sependek mungkin) dengan waktu eksekusi yang sangat cepat (kebanyakan dalam 1 *clock*) dengan panjang tiap intruksi rata-rata 16 bit.

Fitur yang dimiliki masing-masing keluarga AVR sedikit berbeda, perbedaan ini menyebabkan beberapa intruksi tidak kompatibel antar satu seri dengan seri yang lain, selain itu detail arsitekturnya juga sedikit berbeda, perbedaan tersebut diatas bukan suatu yang prinsipil, tetapi merupakan konsekuensi logis yang timbul dari perbedaan fitur masing-masing seri, seperti pada seri AT 90S1200 yang tidak memiliki fitur SRAM (*Static Random Acces Memori*) otomatis intruksi untuk melakukan akses ke SRAM seperti : *pop*, *push*, dan beberapa intruksi lainnya tidak didukung oleh seri AT 90S1200.

Arsitektur dan mikrokontroller jenis ini dirancang mendekati bahasa C. Bahasa pemrograman C merupakan bahasa level atas yang paling banyak dipakai untuk mikrokontrol, sehingga mikrokontrol model AVR yang dirancang dengan mempertimbangkan sifat-sifat pengkodean bahasa C sehingga kode yang dihasilkan bisa sekecil mungkin. Selain itu, dengan pertimbangan bahasa C adalah bahasa yang paling dekat dengan bahasa mesin.

Inti AVR mengkombinasikan banyak intruksi, disusun oleh 32 buah register yang bekerja dengan berbagai tujuan. Semua register tersebut langsung terhubung dengan ALU (*Arithmetic Logic Unit*), memberikan dua register yang independen untuk diakses dengan satu intruksi tunggal dalam satu *clock cycle*. Arsitektur yang diperoleh merupakan kode yang lebih efisien sehingga mencapai sepuluh kali lebih cepat dari pada mikrokontroler CISC konvensional.

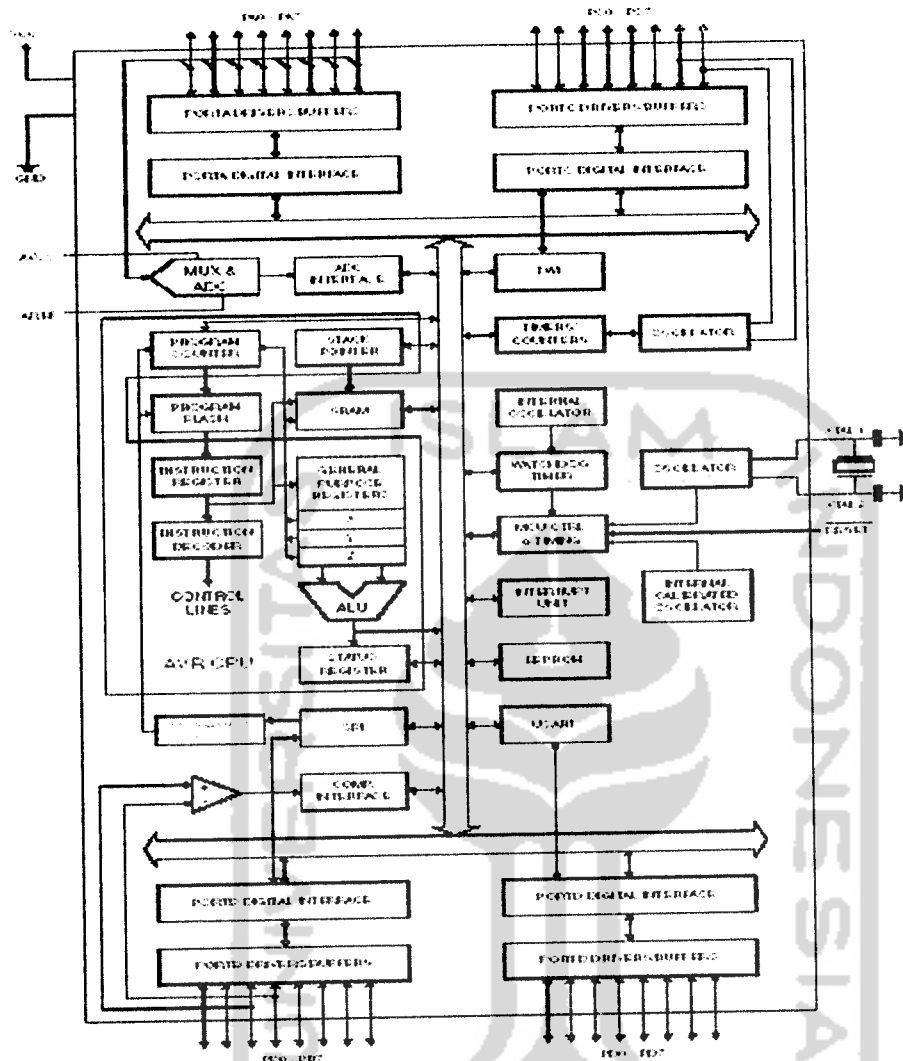
Konfigurasi pin pada mikrokontroler AVR ATmega 16 adalah sesuai dengan gambar 2.4. Jumlah pin pada mikrokontroler tersebut adalah 40 pin yang memiliki 4 port, yaitu port A, port B, port C dan port D dengan I/O lines sebanyak 32. Pada port A terdapat 8 *channel* ADC yang tersedia. ADC yang tersedia ini menggunakan mode 10 bit. X-TAL yang dapat digunakan dalam Mikrokontroler ini, yang direkomendasikan adalah hingga 16MHz. Penggunaan X-TAL yang melebihi dari rekomendasi tersebut, dapat dikategorikan sebagai *overclock*.

2.3.1. Fitur ATmega 16

Kapabilitas detail dari ATmega 16 adalah sebagai berikut :

1. Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MIPS.
2. Kapabilitas memori flash 16 KB, SRAM sebesar 1 Kbyte, dan EEPROM (*Elektrically Erasable Programmable Read Only Memory*) sebesar 512 byte.
3. ADC internal dengan fidelitas 10 bit sebanyak 8 *channel*.
4. Memiliki 131 instruksi yang sebagian besar bekerja dalam satu *clock cycle*.

2.3.2. Arsitektur ATMEGA16



Gambar 2.3. Blok diagram fungsional ATmega16

Dari gambar tersebut dapat dilihat bahwa ATmega16 memiliki beberapa bagian sebagai berikut :

1. Saluran I/O sebanyak 32 buah yaitu Port A, Port B, Port C, dan Port D.
2. ADC sebanyak 8 saluran / channel dengan mode 10 bit.
3. CPU terdiri atas 32 buah register.

4. Memori *Flash* sebesar 16 Kbyte dengan kemampuan *Read While Write*
5. EEPROM sebesar 512 byte.
6. X-TAL direkomendasikan hingga 16 MHz
7. *Whatchdog Timer* dengan osilator internal
8. Unit interupsi internal dan eksternal
9. Port antarmuka SPI
10. Antarmuka komparator analog
11. Port USART untuk komunikasi serial
12. Tiga buah *Timer/Counter* dengan kemampuan perbandingan.

2.3.3. Konfigurasi Pin ATmega 16

Konfigurasi pin ATmega 16 dapat dilihat pada Gambar 2.4

- a. Vcc digunakan untuk masukan tegangan.
- b. Reset digunakan untuk input reset. Pin ini dapat diaktifkan dengan *logic low* selama 50 ms. Dengan syarat itu, maka mikrokontroler dapat dalam kondisi reset.
- c. Ground merupakan pin ground
- d. Port A merupakan satu kelompok 8-bit *bi-directional I/O Port*. Jika A/D *Converter* tidak dapat digunakan, pin ini pada kondisi tri-state ketika terjadi reset, dapat diberi *pull-up* secara internal. Ketika pin PA0 hingga pin PA7 digunakan sebagai input dan secara eksternal dialiri daya rendah, maka akan menjadi sumber arus jika internal *pull-up* register aktif.
- e. Port B merupakan satu kelompok 8-bit *bi-directional I/O Port* dengan internal *pull-up* register (dipilih untuk tiap bit). Sebagai input port B yang secara eksternal

dialiri daya rendah akan menjadi sumber arus jika internal *pull-up* register aktif. Pin pada kondisi *tri-state* ketika terjadi *reset*.

- f. Port C merupakan satu kelompok 8-bit *bi-directional I/O Port* dengan internal *pull-up* register. Sebagai input port C yang secara eksternal dialiri daya rendah akan menjadi sumber arus jika internal *pull-up* register aktif. Pin pada kondisi *tri-state* ketika terjadi *reset*
- g. Port D merupakan kelompok 8-bit *bi-directional I/O Port* dengan internal *pull-up* register. INT0 – INT1 disediakan untuk operasi interupsi secara eksternal. Sebagai input, port D yang secara eksternal dialiri daya rendah akan menjadi sumber arus apabila *pull-up* register aktif. Pin ini pada kondisi *tri-state* ketika terjadi *reset*.
- h. XTAL 1 merupakan input dari *inverting* osilator
- i. XTAL 2 merupakan output dari *inverting* osilator A/D dengan Vcc.
- j. AVCC digunakan sebagai suplai tegangan untuk port A dan A/D *converter*. Apabila ADC tidak digunakan pin ini harus dikoneksikan dengan Vcc. Jika ADC digunakan, pin ini harus dikoneksikan ke Vcc melalui *low-pass filter*.
- k. AREF merupakan input referensi analog untuk A/D *converter*.
- l. AGND analog *ground*. Jika *board* memiliki bidang *ground* analog terpisah, pin ini sebaiknya dikoneksikan ke dalam bidang *ground* ini. Sebaliknya, dihubungkan dengan *ground*.

PCINT0 FB0	1	40	FA0 (AD00)
PCINT0 FB1	2	39	FA1 (AD01)
PCINT0 FB2	3	38	FA2 (AD02)
PCINT0 FB3	4	37	FA3 (AD03)
PCINT0 FB4	5	36	FA4 (AD04)
PCINT0 FB5	6	35	FA5 (AD05)
PCINT0 FB6	7	34	FA6 (AD06)
PCINT0 FB7	8	33	FA7 (AD07)
RESET	9	42	AREF
VCC	10	31	GND
GND	11	30	AGND
XTAL2	12	29	PC0 (TDS00)
XTAL1	13	28	PC1 (TDS01)
PC0 PD0	14	27	PC2 (TDS02)
PC1 PD1	15	26	PC3 (TDS03)
PC2 PD2	16	25	PC4 (TDS04)
PC3 PD3	17	24	PC5 (TDS05)
PC4 PD4	18	23	PC6 (TDS06)
PC5 PD5	19	22	PC7 (TDS07)
PC6 PD6	20	21	PC7 (TDS07)

Gambar. 2.4. Mikrokontrol AVR ATmega 16

2.3.4. Peta Memori

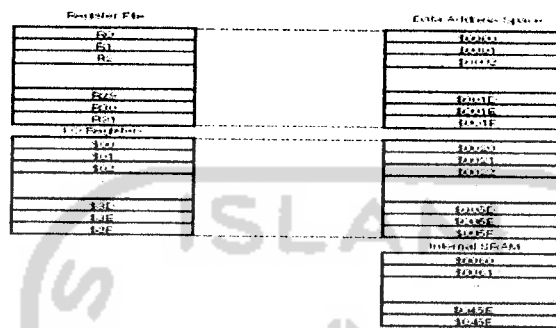
Pada mikrokontroler ATmega 16 terdapat dua bagian memori yaitu : memori data dan memori program. Memori data menempati suatu ruangan alamat yang terpisah dari memori program. Memori data dan memori program eksternal dapat dikombinasikan dengan cara menggabungkan sinyal RD dan PSEN melalui gerbang NAND dan keluarannya sebagai tanda baca ke memori data / program eksternal.

a. Memori data

Memori data terbagi menjadi 3 bagian, yaitu 32 buah register umum, 64 buah register I/O, dan 1024 byte SRAM Internal. Register keperluan umum menempati space data pada alamat dibawah, yaitu \$00 sampai \$1F. sementara itu, register khusus untuk menangani I/O dan control terhadap mikrokontroler menempati 64 alamat berikutnya, yaitu mulai dari \$20 hingga \$5F. Register tersebut merupakan register yang khusus

digunakan untuk mengatur fungsi terhadap berbagai peripheral mikrokontroler, seperti kontrol register, *timer/counter*, fungsi-fungsi I/O dan sebagainya.

Alamat memori berikutnya digunakan untuk SRAM 1 Kbyte, yaitu pada lokasi \$60 sampai dengan \$45F. Konfigurasi memori data ditunjukkan pada gambar dibawah ini.



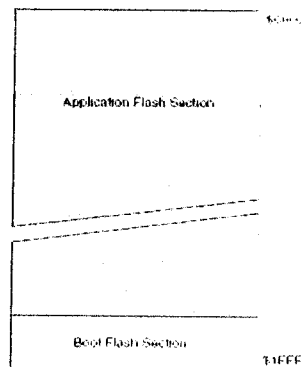
Gambar 2.5. konfigurasi Memori data ATmega16

ATmega16 juga memiliki memori data berupa EEPROM 8 bit sebanyak 512 byte. Alamat EEPROM dimulai dari \$0000 sampai \$1FFF.

EEPROM tetap dapat menyimpan data saat tidak dicatu daya dan juga dapat diubah saat program berjalan. Sangat berguna untuk menyimpan informasi, seperti nilai kalibrasi, nomer ID, dan juga *password*.

b. Memori program

Memori program yang terletak dalam *On-Chip In-System Flash Memori Reprogrammable* tersusun dalam word atau 2 byte karena setiap instruksi memiliki lebar 16-bit atau 32-bit. AVR ATmega16 memiliki 8 Kbyte X 16-bit *Flash memori* dengan alamat mulai dari \$0000 sampai \$1FFF. AVR ini memiliki 13 bit *Program Counter (PC)* sehingga mampu mengamati isi *Flash Memori Program*. Untuk keamanan perangkat lunak, ruang *Flash Memori Program* dibagi ke dalam dua bagian, bagian program sepatu dan bagian program aplikasi.



Gambar 2.6. Memori Program AVR ATmega16

2.3.5. Status Register (SREG)

Status Register adalah register berisi status yang dihasilkan pada setiap operasi yang dilakukan ketika suatu intruksi dieksekusi. SREG merupakan bagian dari inti CPU mikrokontroler.

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	0	0	0	

Gambar 2.7. Status Register ATmega16

a. Bit 7 – I : *Global Interrupt Enable*

Bit harus di set untuk mengaktifkan interupsi. Setelah itu, dapat mengaktifkan interupsi mana yang akan digunakan dengan cara mengaktifkan bit kontrol register yang bersangkutan secara individu. Bit akan di *clear* apabila terjadi suatu interupsi yang dipicu oleh hardware, dan bit tidak akan mengizinkan terjadinya interupsi, serta akan di set kembali oleh instruksi RETI.

b. Bit 6 – T : Bit Copy Storage

Intruksi BLD dan BST menggunakan bit-T sebagai sumber atau tujuan dalam operasi bit. Suatu bit dalam sebuah register GPR dapat disalin ke bit T menggunakan instruksi BST, dan sebaliknya bit-T dapat disalin kembali ke suatu bit dalam register GPR menggunakan instruksi BLD.

c. Bit 5 – H : *Half Carry Flag*

d. Bit 4 – S : *Sign Bit*

Bit-S merupakan hasil operasi EOR antara flag-N(negatif) dan flag V(komplemen dua *overflow*).

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

Bit berguna untuk mendukung operasi aritmatika

f. Bit 2 – N : *Negative Flag*

Apabila suatu operasi menghasilkan bilangan negatif, maka flag-N akan diset.

g. Bit 1 – Z : *Zero Flag*

Bit akan diset bila hasil operasi yang diperoleh adalah nol.

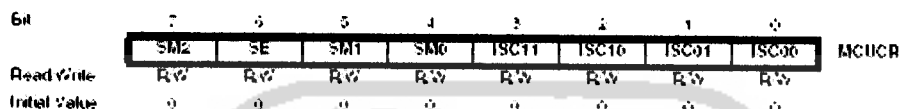
h. Bit 0 – C : *Carry Flag*

Apabila suatu operasi menghasilkan *carry*, maka bit akan diset.

2.3.6. Interupsi

Interupsi adalah suatu kejadian yang akan menghentikan sementara jalannya program yang sedang berjalan untuk menjalankan suatu subrutin interupsi tertentu. Setelah selesai dikerjakan maka program yang dihentikan tadi akan dilanjutkan kembali secara normal.

Pada AVR terdapat 3 pin untuk interupsi eksternal, yaitu INT0, INT1 dan INT2. Interupsi eksternal dapat dibangkitkan apabila terdapat perubahan logika atau logika 0 pada pin interupsi. Pengaturan kondisi keadaan yang menyebabkan terjadinya interupsi eksternal diatur oleh register MCUCR (MCU Control Register) yang terlihat pada ambar dibawah ini :



Gambar 2.8. Register MCUCR

Bit penyusunnya dapat dijelaskan berikut :

- a. Bit ISC11 dan ISC10 bersama-sama menentukan kondisi yang dapat menyebabkan interupsi eksternal pada pin INT1. Keadaan selengkapnya dapat dilihat pada Tabel di bawah ini.

Tabel 2.1. Beberapa setting kondisi menyebabkan Interupsi Eksternal 1

ISC11	ISC10	Deskripsi
0	0	Logika 0 pada pin INT1 menyebabkan interupsi
0	1	Perubahan logika pada pin INT1 menyebabkan interupsi
1	0	Perubahan kondisi 1 ke 0 pada pin INT1 menyebabkan interupsi
1	1	Perubahan kondisi 0 ke 1 pada pin INT1 menyebabkan interupsi

- b. Bit ISC01 dan ISC00 bersama-sama menentukan kondisi yang dapat menyebabkan interupsi eksternal pada pin INT0. Keadaan selengkapnya dapat dilihat pada Tabel dibawah ini.

Tabel 2.2 Beberapa *setting* Kondisi menyebabkan Interupsi Eksternal 0

ISC01	ISC00	Deskripsi
0	0	Logika 0 pada pin INT0 menyebabkan interupsi
0	1	Perubahan logika pada pin INT0 menyebabkan interupsi
1	0	Perubahan kondisi 1 ke 0 pada pin INT0 menyebabkan interupsi
1	1	Perubahan kondisi 0 ke 1 pada pin INT0 menyebabkan interupsi

Pemilihan pengaktifan interupsi eksternal diatur oleh register GICR (*General Interrupt Control Register*) yang terlihat seperti gambar berikut :

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	

Gambar 2.9. *General Interrupt Control Register*

Bit penyusun dapat dijelaskan sebagai berikut :

- Bit INT1 adalah bit yang mengaktifkan interupsi eksternal 1. Apabila bit tersebut diberi logika 1 dan bit-I pada SRE (status register) juga satu, maka interupsi eksternal 1 akan aktif.
- Bit INT0 adalah bit untuk mengaktifkan interupsi eksternal 0. Apabila bit tersebut diberi logika 1 dan bit-I pada SREG (status register) juga satu, maka interupsi eksternal 0 akan aktif.
- Bit INT2 adalah bit untuk mengaktifkan interupsi eksternal 2. Apabila bit tersebut diberi logika 1 dan bit-I pada SREG (status register) juga satu, maka interupsi eksternal 2 akan aktif.

Program interupsi dari masing-masing jenis interupsi eksternal akan dimulai dari vektor interupsi pada masing-masing jenis. Alamatnya dapat dilihat pada Tabel 2.3.

Tabel 2.3. Macam sumber interupsi pada ATmega16

Vector No.	Program Address ^h	Source	Interrupt Definition
1	\$0001 ^h	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

2.3.7. Serial Peripheral Interface (SPI)

Serial Peripheral Interface memungkinkan komunikasi sinkron berkecepatan tinggi antar-mikrokontroler ATmega16 atau antara ATmega16 dengan perangkat lain yang mendukung SPI. SPI memungkinkan untuk membuat aplikasi multiprocessor. Berikut fitur dari SPI ATmega16.

- *Full Duplex*, data transfer tak sinkron menggunakan 3 kabel
- Operasi master atau slave
- Data transfer awal LSB atau MSB
- Tujuh bit rate yang dapat diprogram
- Flag interupsi apabila transmisi data berakhir
- Flag proteksi untuk kegagalan penulisan
- *Wake-up fro idle mode*
- Dua kali kecepatan mode SPI master.



Antarmuka tersebut memungkinkan sebuah perangkat master berhak memulai dan mengendalikan komunikasi. Perangkat lain yang menerima dan mengirimkan data kembali ke master disebut slave.

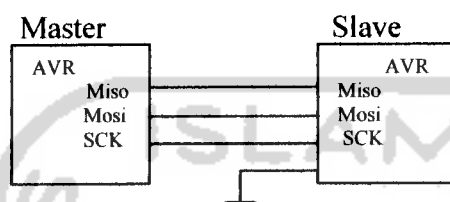
Inti dari komunikasi SPI adalah register geser 8 bit pada kedua piranti master dan slave, serta sinyal clock yang dibangkitkan oleh master. Misalnya, master ingin mengirimkan data A ke slave dan dalam waktu yang sama master menerima data B dari slave. Sebelum memulai komunikasi SPI, master meletakkan data A ke shift registernya dan B juga meletakkan atau B di shift register. Selanjutnya, master membagikan 8 pulsa clock sehingga data pada shift register master ditransferkan ke shift register slave, dan sebaliknya. Pada akhir pulsa, clock master telah menerima data B dan slave telah menerima data A. Oleh karena data diterima pada saat yang sama, maka SPI termasuk dalam komunikasi *full duplex*

Komunikasi dengan SPI membutuhkan 4 jalur sinyal, yaitu:

1. SCK (*Serial Clock*); yaitu sinyal clock yang menggeser bit yang hendak dituliskan kedalam register geser terima AVR lain, dan menggeser bit yang hendak dibaca dari register geser kirim AVR lain.
2. MOSI (*Master Out Slave In*); sinyal bit data serial yang hendak dituliskan dari master ke slave.
3. MISO (*Master In Slave Out*); sinyal bit data serial yang hendak dibaca dari slave ke master.
4. SS' (*Slave Select/ aktif rendah*); sinyal untuk memilih dan mengaktifkan slave.
5. SPI memungkinkan komunikasi dengan beberapa slave dengan satu master. Cara master memilih slave yang diinginkan untuk berkomunikasi adalah menggunakan

pin SS'. Jika pin SS' diset pada logika high, maka pin SPI slave berfungsi sebagai normal input dan tidak akan menerima data SPI masuk. Dilain pihak, apabila pin SS' berlogika *low*, maka SPI akan aktif.

6. Pada konfigurasi master, pin SS' harus diset sbagai output atau dapat berupa input, tetapi harus berlogika high.



Gambar 2.10. Perkawatan SPI saat menggunakan 1 Slave

Secara umum, konfigurasi master dan slave saat komunikasi SPI yang dilakukan adalah seperti table berikut.

Tabel 2.4. Konfigurasi pin master dan slave saat komunkasi SPI

Pin	Arah, Master SPI	Arah, Slave SPI
MOSI	Ditentukan User	Input
MISO	Input	Ditentukan User
SCK	Ditentukan User	Input
SS'	Ditentukan User	Input

2.4. Logika

Logika sangat penting dalam aspek kehidupan manusia. Dengan menggunakan logika kita lebih efektif dalam mengenal dan menghindari kesalahan penalaran. Adapun aplikasi logika salah satunya di bidang teknologi informasi dimana di bidang ini kita mengenal perangkat komputer serta bahasa pemrogramannya. Perangkat komputer

tersebut biasanya disebut *hardware* dan bahasa pemrograman biasa disebut *software*, dan baik *software* ataupun *hardware* banyak menggunakan prinsip-prinsip logika.

Dalam menerapkan suatu logika, terdapat beberapa operasi yang sering ditemui, seperti operasi negasi yaitu penyangkalan atau ingkaran. Ingkaran dari suatu pernyataan merupakan pernyataan baru dengan nilai kebenaran berlawanan dengan nilai pernyataan sebelumnya. Ingkaran dari suatu pernyataan diperoleh dengan cara memberi tambahan kata “tidak benar” di awal kalimat, atau dapat pula memberi tambahan kata “tidak” atau “bukan” pada pernyataan tersebut.

Operasi lainnya adalah operasi biner. Operasi biner adalah operasi antara dua pernyataan, operasi biner antara dua pernyataan tunggal akan menghasilkan suatu pernyataan majemuk. Ada 3 macam operasi biner yaitu NOT, AND, OR

2.4.1 Operasi NOT

Nilai yang dihasilkan dari operasi NOT adalah kebalikan dari nilai yang di kandung di dalamnya. jika nilai awal bernilai benar (1), maka setelah operasi not nilainya menjadi salah (0), begitu juga sebaliknya. Berikut ini adalah tabel yang menunjukkan hasil dari operasi NOT

Tabel 2.5. Nilai kebenaran operasi NOT

X	!X
1	0
0	1

2.4.2. Operasi AND

Operasi AND hanya akan menghasilkan nilai benar (1) jika semua operand nya bernilai benar, namun jika tidak, maka oprasi tersebut akan menghasilkan nilai salah (0) berikut ini adalah tabel yang menunjukkan hasil dari operasi AND.

Tabel 2.6. Nilai kebenaran operasi AND

X	Y	X & Y
1	1	1
1	0	0
0	0	0
0	1	0

2.4.3 Operasi OR dan XOR

Operasi OR hanya akan menghasilkan nilai salah (0) jika semua operannya bernilai salah, namun jika tidak maka operasi tersebut akan menghasilkan nilai 1 (benar), operasi OR bila di kombinasikan dengan kedua operasi logika diatas maka dapat menghasilkan operasi XOR (Eklusif OR), adapun hasil dari operasi ini akan bernilai benar (1) jika satu operand nya (bukan salah satu) bernilai benar, selain itu akan menghasilkan salah (0) dengan demikian, jika kedua operand nya bernilai benar (1), maka hasil dari operasi itu adalah ini adalah salah (0). Untuk lebih memahaminya, berikut ini adalah tabel yang menunjukkan operasi OR dan Operasi XOR.

Tabel 2.7. Nilai kebenaran operasi OR

X	Y	X + Y
1	1	1
1	0	1
0	0	0
0	1	1

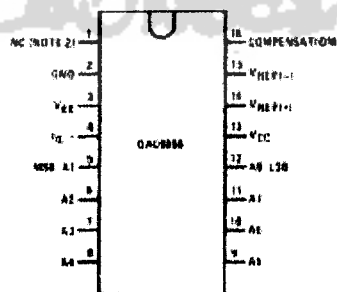
Tabel 2.8. Nilai kebenaran operasi XOR

X	Y	$X \oplus Y$
1	1	0
1	0	1
0	0	0
0	1	1

2.5. DAC

DAC (Digital Analog to Converter) adalah suatu rangkaian pengubah informasi dari digital ke analog. Rangkaian ini diperlukan pada saat suatu rangkaian digunakan sebagai alat kontrol pada suatu sistem rangkaian yang mengoperasikan parameter tegangan/arusnya dalam analog, misalnya peralatan pemanas(heater), pengatur putaran DC motor, speech synthesizer, sistem rekaman digital dan lain-lain. D/A Converter akan mengubah setiap konfigurasi logika pada input-inputnya ke dalam tegangan analog pada outputnya dengan perbandingan tertentu. DAC dapat digunakan untuk menyelesaikan berbagai macam permasalahan pembangkitan gelombang yang akurat, misalnya digital function generator, Digital Oscillator untuk transceiver sebagai pengganti PLL (Phase Lock Loop).

Salah satu contoh DAC yang sering kita jumpai dalam rangkaian – rangkaian elektronik misalnya DAC 0808, DAC 0808 adalah D/A converter yang mempunyai 8 bit input, dibangun dengan metode konversi rangkaian R-2R ladder. Input data biner IC ini kompatibel dengan gate TTL sehingga output port standar PIO, misalnya PPI8255 atau Z80PIO, dapat dihubungkan secara langsung. IC DAC 0808 ini mempunyai konfigurasi pin seperti pada gambar 2.11.



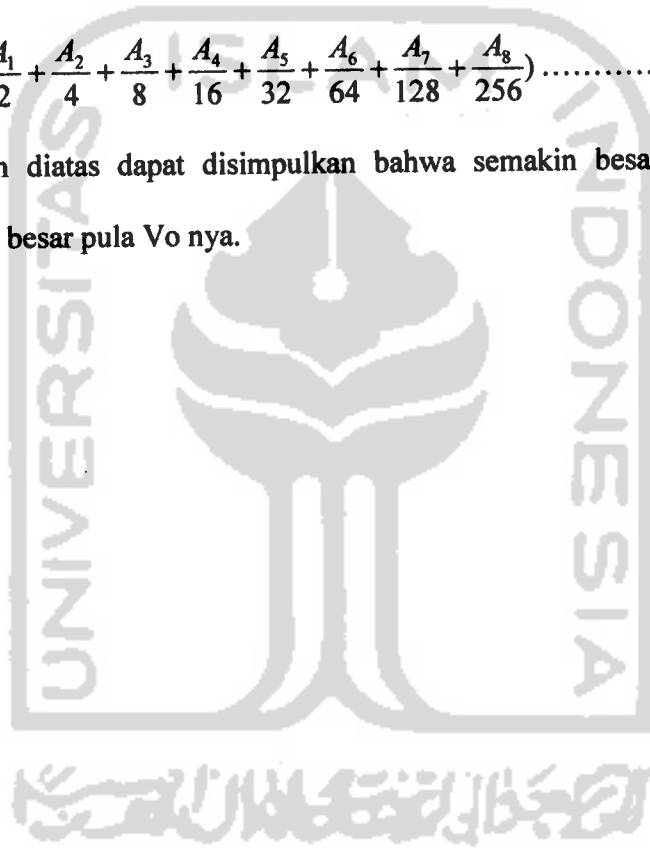
Gambar 2.11 : konfigurasi pin DAC 0808



DAC 0808 dipasang pada V_{cc} dengan tegangan +5v untuk memberikan referensi pada tegangan input yang berlevel TTL. Adapun jangkauan output ditentukan oleh $V_{ref}(+)$ dan $V_{ref}(-)$. Tegangan output DAC dapat diatur sesuai catu tegangan referensi yang diberikan. Pedoman tegangan output minimum juga dapat diatur dengan mengeset perbandingan tegangan pada referensi-referensinya. Sehingga dapat dituliskan persamaan rumus tegangan outputnya adalah :

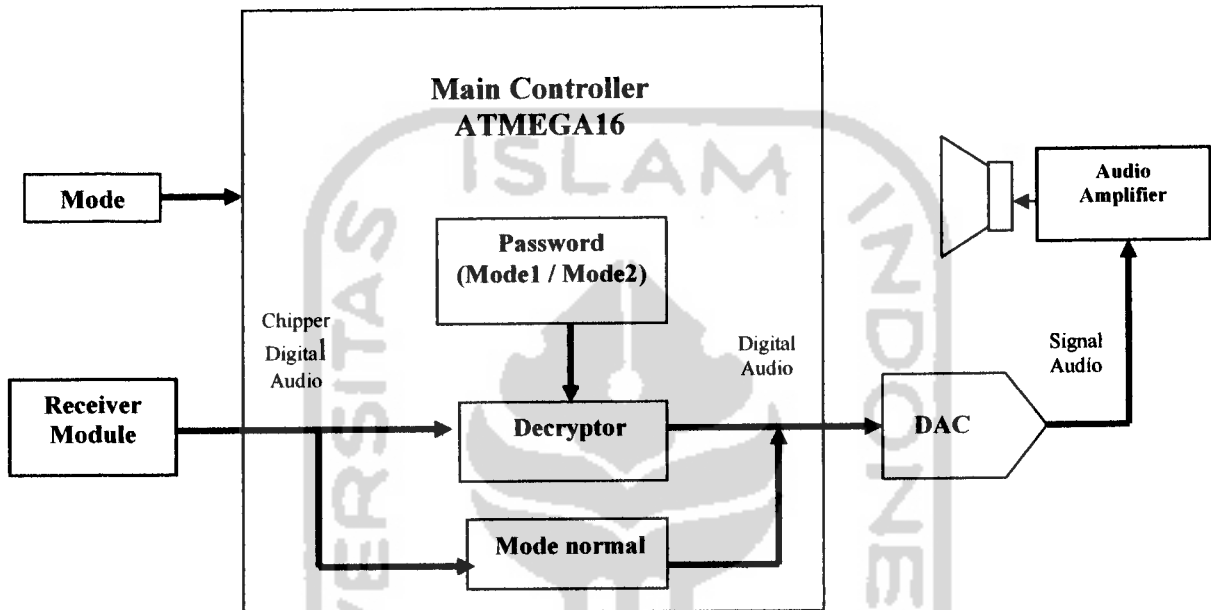
$$V_{out} = V_{ref} \left(\frac{A_1}{2} + \frac{A_2}{4} + \frac{A_3}{8} + \frac{A_4}{16} + \frac{A_5}{32} + \frac{A_6}{64} + \frac{A_7}{128} + \frac{A_8}{256} \right) \dots\dots\dots(2.5)$$

Dari persamaan diatas dapat disimpulkan bahwa semakin besar bit-bit yang diterima DAC semakin besar pula V_o nya.



BAB III
PERANCANGAN ALAT

3.1. Perancangan Sistem.



Gambar 3.1 Diagram Blok Sistem Dekripsi Signal Audio

Prinsip kerja dari rangkaian diatas pada awalnya sistem melakukan inisialisasi atau *setting* awal. Sistem dalam posisi mode normal dan menunggu tombol Mode 1 atau Mode 2 ditekan untuk mengawali proses dekripsi, apabila tombol Mode 1 ditekan, maka proses dekripsi akan dilakukan dengan dengan *fixed password* 1, Apabila tombol Mode 2 ditekan, maka proses dekripsi akan dilakukan dengan menggunakan *fixed password* 2, selanjutnya apabila akan dilakukan penghentian proses dekripsi, maka dapat dilakukan dengan menekan tombol normal. Proses dekripsi diawali dengan signal yang masuk

melalui *receiver module* yaitu berupa sinyal *Chipper Digital Audio* selanjutnya sinyal tersebut masuk ke sebuah mikrokontrol untuk di dekripsi dengan menggunakan sebuah kunci dan fix password. Hasil pengolahan dari sistem tersebut, akan menghasilkan sinyal digital audio. Selanjutnya, sinyal digital audio tersebut diolah dengan DAC menjadi sinyal audio, yaitu sesuai dengan sinyal aslinya sebelum signal tersebut dikirimkan.

3.1.1. Bagian-bagian Sistem

Berdasarkan diagram blok di atas, sistem dapat dibagi dalam beberapa bagian, yaitu

1. *Main Controller*

Main Controller sebagai pusat pemrosesan sinyal audio memerlukan kecepatan yang tinggi serta memiliki fungsi-fungsi kendali. Prosesor yang ideal untuk perancangan pengolahan sinyal adalah DSP, yaitu *Digital Signal Processor*, karena DSP memiliki minimal 32 bit. Hal ini tentu saja akan ikut mempengaruhi kecepatan akses data untuk menuju terwujudnya sistem yang *real time*.

Namun demikian, dalam sistem ini digunakan Mikrokontroler atmel AVR seri ATmega16, dengan pertimbangan bahwa mikrokontroler ini dikenal sebagai mikrokontroler dengan arsitektur terbaru, yaitu *RISC architecture*. Sehingga akan memiliki kecepatan yang lebih tinggi dari mikrokontroler generasi sebelumnya. Mikrokontroler 8-bit ini memiliki pemrosesan data dengan proses internal 16-bit. Sehingga mikrokontroler ini sudah akan memiliki kecepatan pemrosesan 2 kali dari kecepatan pemrosesan yang dimiliki mikrokontroler seri 89C51.

2. Mode

Mode adalah tombol yang digunakan untuk proses dekripsi, tombol ini berisi tiga buah mode yaitu tombol normal, mode 1 dan mode 2. Tombol normal adalah tombol untuk menghentikan proses dekripsi, sehingga sinyal yang masuk diloloskan langsung menuju DAC tanpa dilakukan proses dekripsi. Tombol Mode 1 adalah tombol perintah dekripsi dengan menggunakan *fixed password 1*. Dengan penekanan tombol ini, maka sistem akan melakukan dekripsi sinyal yang masuk. Tombol Mode 2 adalah tombol perintah dekripsi dengan menggunakan *fixed password 2*. Dengan penekanan tombol ini, maka sistem akan melakukan dekripsi sinyal yang masuk.

3. Receiver Module

Modul ini digunakan untuk menerima data yang telah dienkripsi untuk ditransfer ke Mikrokontroler agar data bisa diterjemahkan.

4. DAC

Digunakan untuk keperluan perubahan sinyal digital menjadi sinyal analog. Hasil pengolahan data *chipper digital audio* menjadi *digital audio* dimasukan ke input DAC, yang kemudian dikeluarkan dalam bentuk signal audio.

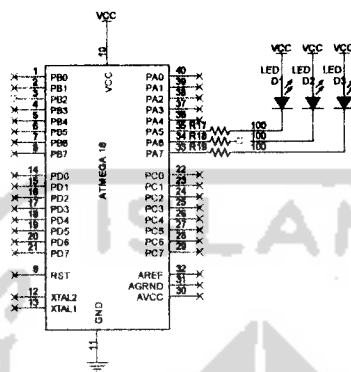
5. Audio Amplifier

Yaitu alat yang digunakan untuk memperkuat sinyal audio menjadi suara yang dengan tingkat suara yang dapat didengar oleh manusia melalui *speaker* atau *headphone*.

3.2 Perancangan Hardware

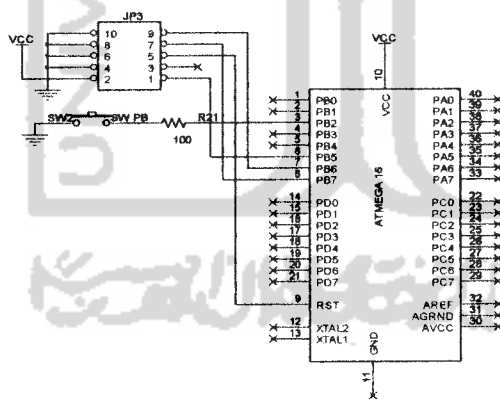
3.2.1. Perancangan port Mikrokontroler AT MEGA 16

Port A pada mikrokontroler ATmega16 digunakan sebagai output yang berupa nyala LED, dimana pin yang digunakan adalah port C.5 sampai port C.7



Gambar 3.2. Rangkaian LED

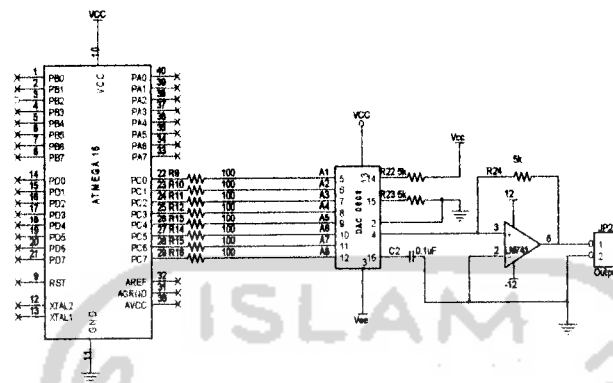
Port B pada mikrokontroler ATmega16 yang digunakan untuk saluran *Serial Peripheral Interface* (SPI) atau saluran pemrograman mikrokontroler adalah port B.5 sampai port B.7 dan sebagai saluran input untuk mode-mode adalah port B.2.



Gambar 3.3. Rangkaian SPI dan mode

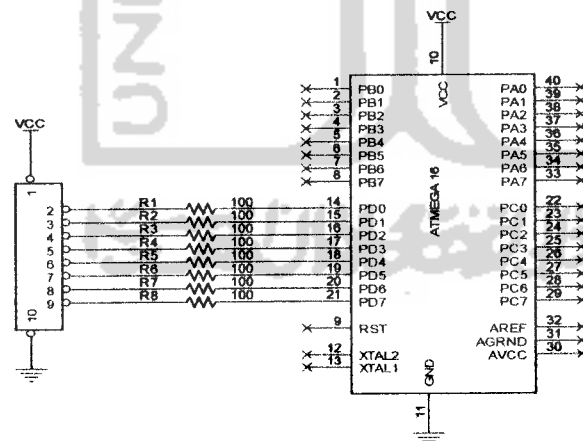
Port C pada mikrokontroler ATmega16 digunakan sebagai keluaran data yang telah di dekripsi, yaitu data yang berupa *signal digital audio*, yang kemudian dimasukkan ke

input DAC untuk menghasilkan keluaran dalam bentuk *signal audio*. Dimana pin yang digunakan adalah port C.0 sampai port C.7



Gambar 3.4. Rangkaian output

Port D pada mikrokontroler ATmega 16 digunakan sebagai input data atau untuk menerima data yang telah dienkripsi, yang kemudian data itu nantinya akan diproses secara dekripsi di dalam mikrokontroler. Dimana pin yang digunakan adalah port C.0 sampai port C.7

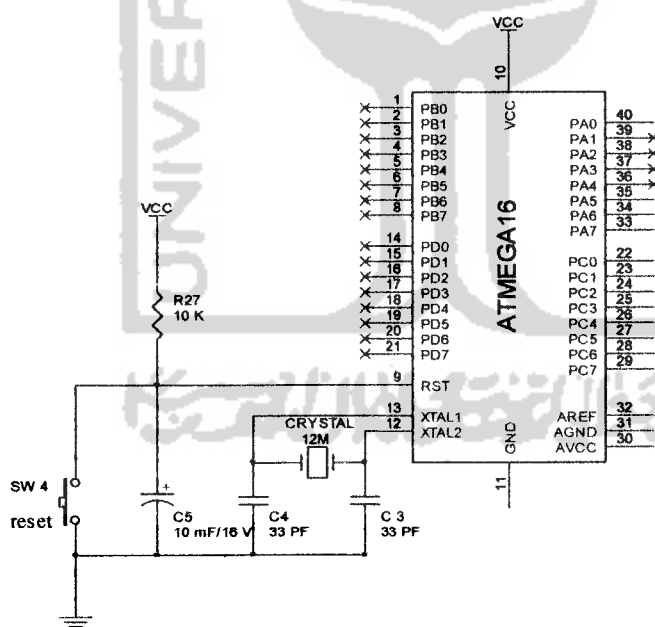


Gambar 3.5. Rangkaian input

3.2.2. Osilator dan Reset

Pada rangkaian *osilator* ini menggunakan pin 12 (XTAL1) dan pin 13 (XTAL2) diparalel, digunakan kristal 16 MHz seperti ditunjukkan Gambar 3.8. Menurut *datasheet* kristal yang bisa digunakan untuk mikrokontroler ATmega16 adalah 0 – 33 MHz dan dua kapasitor 30 ± 10 pF.

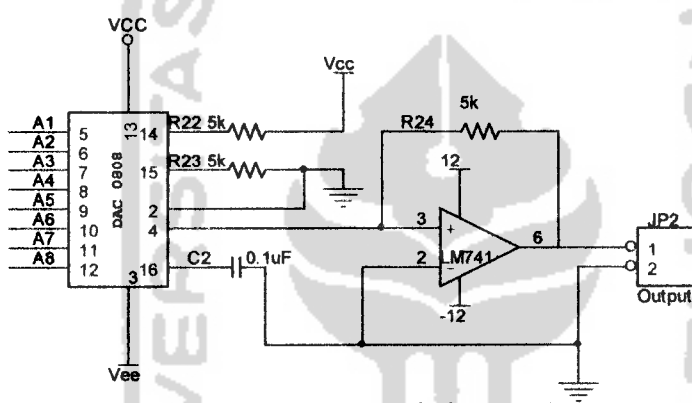
Pin 9 (RST) merupakan rangkaian *reset*, yang mana digunakan untuk menghentikan kerja mikrokontroler dengan kembali ke alamat 0000/reset. Rangkaian *reset* dapat dilihat pada Gambar 3.6. Untuk *mereset* mikrokontroler ATmega16 yaitu dengan memberikan logika tinggi pada pin *reset* (pin 9) mikrokontroler ATmega16, logika tinggi ini dibuat minimal dua *machine cycles* (24 *Oscilator periode*). Jika menggunakan XTAL 16 MHz maka logika tinggi minimal selama 2 μ s.



Gambar 3.6. Rangkaian osilator dan reset.

3.2.3. Rangkaian DAC

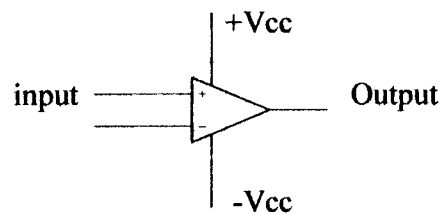
Fungsi dari DAC ini adalah untuk mengubah data digital yang dikirim oleh mikrokontroler yang berupa data digital menjadi data analog yang berupa tegangan atau arus yang besarnya sebanding dengan harga data digital tersebut. Rangkaian DAC ini masukannya berupa sinyal digital 8 bit dan mengubahnya menjadi sinyal analog berupa arus, untuk mengubah arus ini menjadi besaran tegangan maka dipakai Op-Amp dan sebuah resistor R_f yang berfungsi sebagai rangkaian pengubah arus ke tegangan.



Gambar 3.7. Rangkaian DAC

3.2.4. Rangkaian Operational Amplifier

Penguat operasional (*op-amp*) adalah suatu rangkaian terintegrasi yang berisi beberapa tingkat dan konfigurasi penguat differensial. Penguat operasional memiliki dua masukan dan satu keluaran serta memiliki penguat DC yang tinggi untuk dapat bekerja dengan baik, penguat operasional memerlukan tegangan catu yang simetris yaitu tegangan yang berharga positif (V^+) dan tegangan yang berharga negatif (V^-) terhadap tanah (*ground*) simbol penguat operasional ditunjukkan pada Gambar 3.8.



Gambar 3.8. Simbol penguat operasioanal (*Op-amp*)

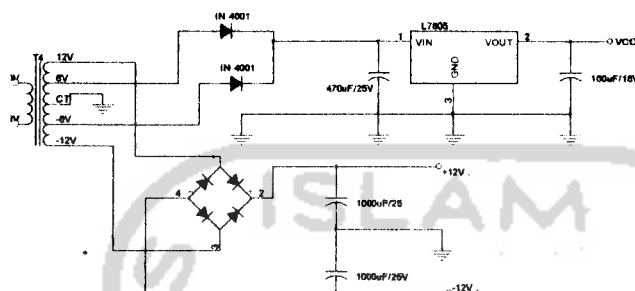
Penguat operasional banyak digunakan dalam berbagai aplikasi karena beberapa keunggulan yang dimilikinya, seperti penguatan yang tinggi, impedensi masukan yang tinggi, impedensi keluaran yang rendah, dan lain sebagainya

Kondisi ideal tersebut hanya merupakan kondisi teoritis, tidak mungkin dicapai dalam kondisi praktis. Tetapi para pembuat *op-amp* berusaha untuk membuat *Op-mp* yang memiliki karakteristik mendekati seperti diatas, karena itu sebuah *op-amp* yang baik harus memiliki karakteristik yang mendekati ideal.

3.2.5. Perancangan Catu Daya.

Rangkaian catu daya merupakan bagian yang sangat penting pada rangkaian ini karena tanpa adanya catu daya maka rangkaian ini tidak akan dapat bekerja. Catu daya digunakan sebagai penyedia sumber tegangan untuk keseluruhan sistem, sehingga dapat mempertahankan suatu level tegangan yang konstan yang sangat diperlukan dalam rangkaian catu daya, dengan demikian rangkaian catu daya pada tugas akhir ini menggunakan regulator tegangan (*voltage regulator*) yang berbentuk IC (*Integrated circuit*) yang mengandung sejumlah rangkaian untuk untuk referensi, alat pengontrol, koparator danm tegangan berlebih (*overload protection*). Jenis regulasi yang digunakan seri 78xx sebagai regulator tegangan tetap positif. Komponen ini biasanya

sudah dilengkapi dengan pembatas arus (*current limiter*) dan juga pembatas suhu (*thermal shut down*). Komponen ini hanya tiga pin dan dengan menambah beberapa komponen saja sudah dapat menjadi rangkaian catu daya yang teregulasi dengan baik. rangkaian catu daya dapat ditunjukkan pada gambar berikut.



Gambar 3.9. Rangkaian catu daya

Berhubung semua rangkaian menggunakan tegangan 5 volt maka catu daya pada rangkaian ini dapat diganti transformator 6 volt yang dihubungkan dengan IC 7805 yang berfungsi menurunkan tegangan menjadi 5 volt.

3.3. Perancangan Perangkat Lunak

3.3.1. Perancangan Teknik Dekripsi

Perancangan dekriptor simetrik sinyal audio berbasis ATmega 16 pada dasarnya merupakan fasilitas untuk mengkonversi *chipper digital audio* menjadi *digital audio*. Dalam tugas akhir ini, seperangkat parameter yang menentukan transformasi tertentu disebut suatu set kunci. Proses dekripsi ini diatur oleh dua buah kunci, secara umum kunci-kunci yang digunakan untuk proses dekripsi tidak perlu identik. Tergantung pada sistem yang digunakan.

Perancangan dekriptor simetris sinyal audio dalam tugas akhir ini terdiri dari 2 buah kunci, kedua buah kunci tersebut identik dengan logika XOR, kedua kunci tersebut

disimbolkan mode 1 sebagai kunci 1 dan mode 2 sebagai kunci 2. untuk proses dekripsinya digunakan dua buah *password* dalam perancangan system ini yaitu sebagaimana yang dapat dijelaskan secara matematis :

$$Y(n) = I(n) \oplus I(n-1) \dots \dots \dots (3.1)$$

$$Y(n) = I(n) \oplus I(n-2) \dots \dots \dots (3.2)$$

Berdasarkan *password* diatas yang sudah dijelaskan sehingga diperoleh :

- **Dekripsi mode 1:**

$$Y(n) = I(n) \oplus I(n-1) \dots \dots \dots (3.3)$$

Dimana $Y(n)$ = Data output

$I(n)$ = Data input

$I(n-1)$ = Data input sebelumnya n-1

- **Dekripsi mode 2 :**

$$Y(n) = I(n) \oplus I(n-2) \dots \dots \dots (3.4)$$

Dimana $Y(n)$ = Data output

$I(n)$ = Data input

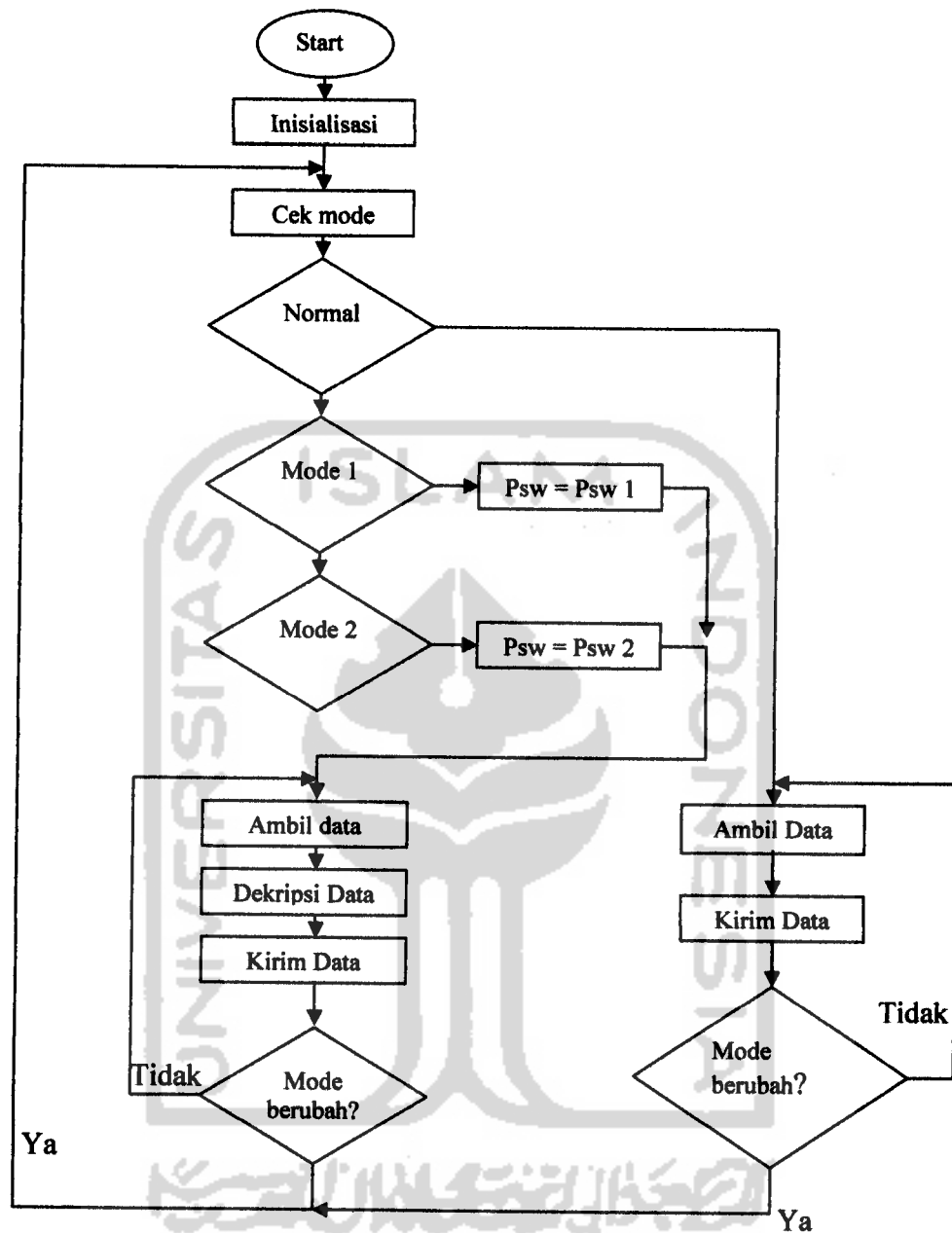
$I(n-2)$ = Data input sebelumnya n-2.

Dari persamaan diatas terdapat perbedaan password antara kunci 1 dan kunci 2, dimana perbedaan password untuk kunci 1 adalah data input di XOR dengan data input sebelumnya yang ke-1 dan untuk kunci 2 adalah data input di XOR dengan data input sebelumnya yang ke-2, dengan adanya perbedaan ini jelas mempengaruhi hasil keluaran dari masing-masing mode juga sedikit berebeda. Akan tetapi dalam hal ini tidak mempengaruhi perubahan bentuk sinyal.

Agar sistem bekerja dengan sebagaimana mestinya, maka diperlukan perangkat lunak yang mengatur kerja dari keseluruhan rangkaian. Pertama yang dibuat adalah diagram alir (*Flowchart*) dan kemudian dilakukan pembuatan program. Pembuatan program ditulis dalam bahasa C dengan menggunakan *Code Vision AVR* Mikrokontroler ATmega 16, dan program tersebut disimpan dalam memori flash. Pada mikrokontroler ATmega 16 terdapat memori program sebesar 16 kbyte flash dan memori data 1k byte RAM.

3.3.2. Program utama.

Program utama meliputi sub rutin-sub rutin dari tiap rangkaian. Dan mekanismenya adalah yang pertama di mulai, program akan melakukan inisialisasi, kemudian dilanjutkan dengan pengecekan tombol atau mode, jika mode = normal maka data langsung diambil tanpa ada proses dekripsi, kemudian data langsung dikirim, jika tombol tidak ditekan maka program akan berjalan terus - menerus pada mode normal, jika tombol ditekan maka program akan berhenti selama 200 ms, kemudian program berjalan lagi untuk langsung mengecek ke mode 1, selanjutnya mode 1 langsung mengambil data, kemudian data di proses secara dekripsi, setelah diproses data langsung dikirim, selanjutnya program akan berjalan terus menerus hingga tombol 2 ditekan, apabila tombol ditekan maka program akan berhenti selama 200 ms, kemudian program berjalan lagi untuk mengecek ke mode 2, selanjutnya mode 2 langsung mengambil data kemudian data langsung diproses secara dekripsi, kemudian langsung dikirim. Program akan terus-menerus mengalami pengulangan seperti yang telah dijelaskan. Berikut adalah diagram alir dari program utama.

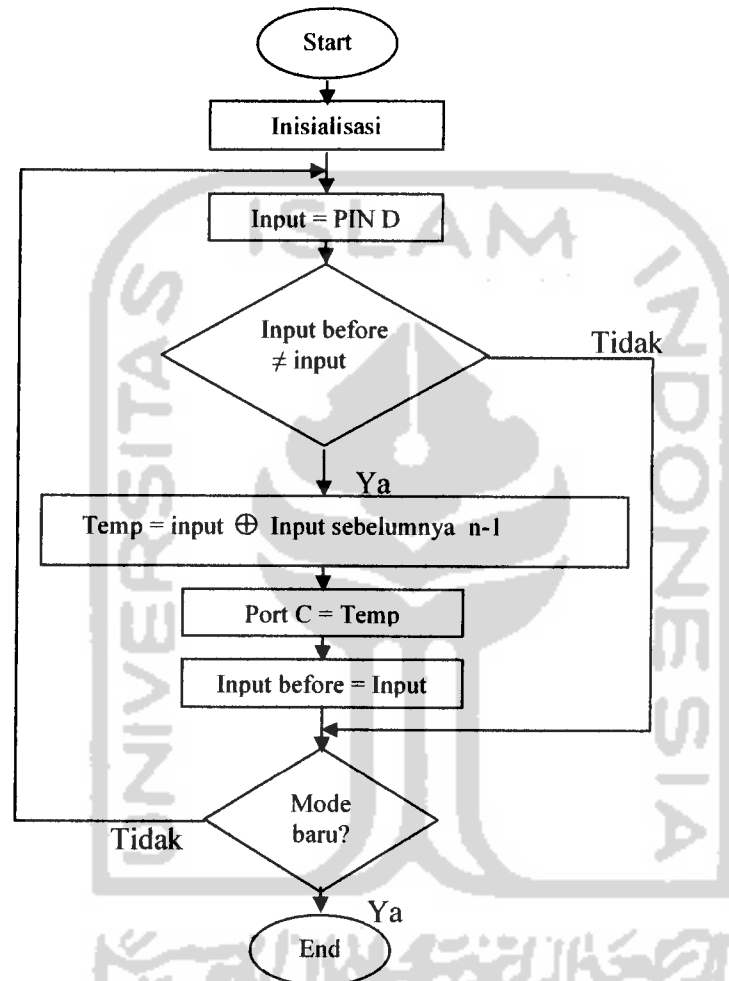


Gambar 3.10. Flowcahrt program utama.

3.3.3. Program dekripsi mode 1

Pada sub rutin proses dekripsi, program pertama kali program akan melakukan inisialisasi, selanjutnya program akan mengecek apakah input tidak sama dengan input sebelumnya, jika input sama dengan input sebelumnya maka data tidak diproses sebaliknya jika input tidak sama dengan input sebelumnya maka data akan

langsung diproses seperti yang terlihat dalam *flowchart* dibawah, kemudian setelah data diproses maka data akan dikeluarkan di port C, sehingga input sebelumnya sama dengan input, selanjutnya program akan terus-menerus mengalami pengulangan.

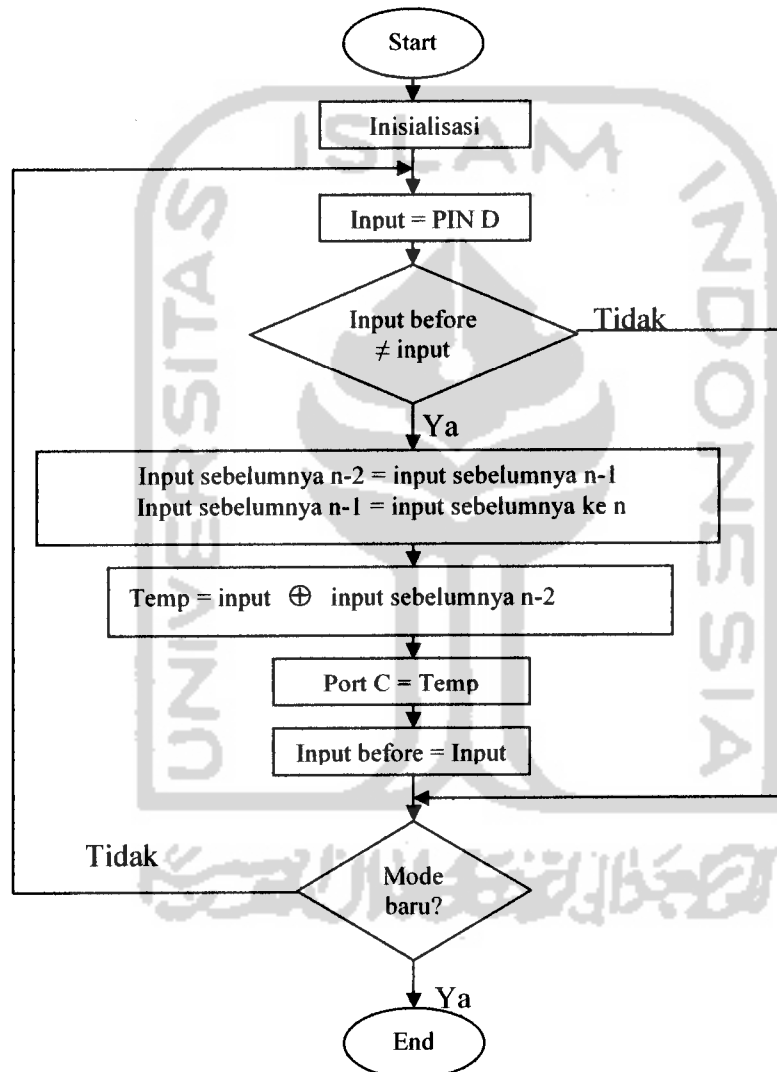


Gambar 3.11. *Flowchart* program dekripsi mode 1

3.3.4. Program dekripsi mode 2

Pada sub rutin proses dekripsi, program pertama kali program akan melakukan inisialisasi, selanjutnya program akan mengecek apakah input tidak sama dengan input sebelumnya (input sebelumnya yang ke-2 sama dengan input sebelumnya yang ke-1 dan input sebelumnya yang ke-1 sama dengan input), jika input sama dengan input

sebelumnya maka data tidak diproses sebaliknya jika input tidak sama dengan input sebelumnya maka data akan langsung diproses seperti yang terlihat dalam *flowchart* dibawah, kemudian setelah data diproses maka data akan dikeluarkan di port C, sehingga dua input sebelumnya sama dengan input, selanjutnya program akan terus-menerus mengalami pengulangan.



Gambar 3.12. *Flowchart* program dekripsi Mode 2.

BAB IV

PENGUJIAN DAN PEMBAHASAN

Untuk mengetahui apakah sistem yang dirancang bekerja dengan baik maka dilakukan pengujian, pengujian ini dilakukan tiap sistem yang meliputi pengujian DAC, pengujian sinyal dekripsi terhadap tiap-tiap mode, selain untuk mengetahui apakah sistem yang dirancang bekerja dengan baik, pengujian dilakukan untuk mengetahui tingkat ketepatan dari sistem.

4.1. Pengujian DAC

Pengujian DAC ini dilakukan untuk mengetahui apakah DAC tersebut bekerja dengan baik, dalam arti dapat mengubah data-data digital yang diberikan ADC menjadi tegangan analog.

Proses pengujian blok ini adalah sebagai berikut :

DAC diberi data dari mikrokontrol kemudian keluaran DAC diamati dengan voltmeter dan hasilnya dicatat. seperti terlihat dalam Tabel 4.1.

Tabel 4.1. Hasil pengujian DAC

Input (data)								Hasil pengukuran output (volt)	Hasil perhitungan output	Selisih (error)
D7	D6	D5	D4	D3	D2	D1	D0			
1	1	1	1	1	1	1	1	4,98	4,98	0
1	1	1	1	0	1	0	1	4,78	4,78	0
1	1	1	0	1	1	0	0	4,60	4,60	0
1	1	1	0	0	0	1	0	4,41	4,41	0
1	1	0	1	0	1	1	1	4,19	4,19	0
1	1	0	0	1	1	0	1	4,00	4,00	0
1	1	0	0	0	0	1	1	3,80	3,80	0

Lanjutan tabel 4.1

1	0	1	1	1	0	0	1	3,63	3,63	0
1	0	1	0	1	1	1	0	3,39	3,39	0
1	0	1	0	0	1	0	1	3,22	3,22	0
1	0	0	1	1	0	1	0	3,00	3,00	0
1	0	0	1	0	0	0	1	2,83	2,83	0
1	0	0	0	0	1	1	0	2,61	2,61	0
0	1	1	1	1	0	1	1	2,40	2,40	0
0	1	1	1	0	0	0	1	2,20	2,20	0
0	1	1	0	0	1	1	0	1,99	1,99	0
0	1	0	1	1	1	0	0	1,83	1,83	0
0	1	0	1	0	0	1	0	1,60	1,60	0
0	1	0	0	1	0	0	0	1,40	1,40	0
0	0	1	1	1	1	1	0	1,21	1,21	0
0	0	1	1	0	0	1	1	0,99	0,99	0
0	0	1	0	1	0	0	1	0,80	0,80	0
0	0	0	1	1	1	1	1	0,60	0,60	0
0	0	0	1	0	1	0	0	0,39	0,39	0
0	0	0	0	1	0	0	1	0,17	0,17	0
0	0	0	0	0	0	0	0	0	0	0

Dari tabel diatas dapat diambil kesimpulan bahwa DAC tersebut berfungsi dengan baik dan dapat mengkonversi data ke bentuk tegangan analog.

Misalnya data yang dikirim mikriontroller ke DAC dengan data 0 0 0 0 1 0 0 1, maka dapat kita peroleh tegangan outputnya sesuai dengan rumus :

$$V_{out} = V_{ref} \left(\frac{A_1}{2} + \frac{A_2}{4} + \frac{A_3}{8} + \frac{A_4}{16} + \frac{A_5}{32} + \frac{A_6}{64} + \frac{A_7}{128} + \frac{A_8}{256} \right)$$

Maka di hasilkan V_{out} sebagai berikut:

$$V_{out} : 5 \left(\frac{0}{2} + \frac{0}{4} + \frac{1}{8} + \frac{0}{16} + \frac{1}{32} + \frac{0}{64} + \frac{0}{128} + \frac{1}{256} \right)$$

: 0,17 Volt

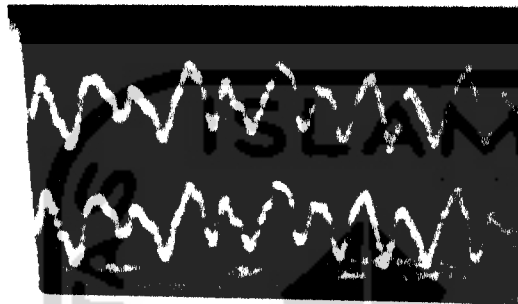
Sehingga pada data 0 0 0 0 1 0 0 1 dihasilkan keluaran sebesar 0,17 volt.



4.3. Pengujian dengan menggunakan Sinyal Audio

4.3.1 Pengujian sinyal dengan menggunakan mode normal

Pengujian ini bertujuan untuk mengamati sinyal masukan dan sinyal keluaran, apakah sinyal output sama dengan sinyal keluaran, pengujian dilakukan dengan menggunakan masukan berupa sinyal audio dan keluaran akan terlihat pada osiloskop.

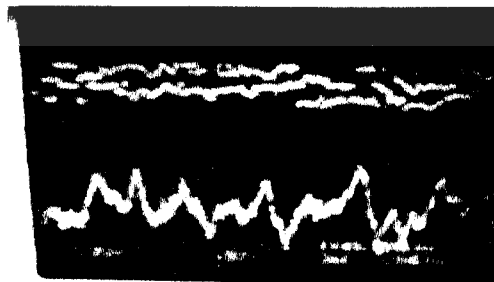


Gambar 4.1. Sinyal audio tanpa enkripsi dan dekripsi

Dari hasil pengujian ini diperoleh kesamaan antara kedua sinyal yaitu sinyal masukan dan sinyal keluaran, ini disebabkan karena pada pengujian ini tidak ada proses enkripsi dan dekripsi sehingga antara sinyal masukan dan keluaran mempunyai bentuk yang sama.

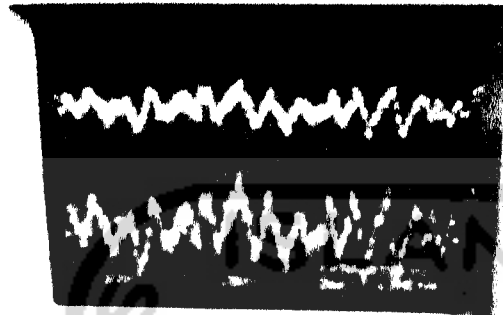
4.3.2. Pengujian sinyal dengan menggunakan mode 1

Pengujian ini dilakukan dengan cara yang sama seperti mode diatas, tetapi pada pengujian ini sebelum sinyal didekripsi terlebih dulu sinyal dienkripsi.



Gambar 4.2. Sinyal enkripsi dengan mode 1

Pada Gambar 4.2. diatas diperoleh sinyal dalam keadaan telah di enkripsi dengan menggunakan mode 1 dan *password* 1, hasilnya adalah berbeda dengan sinyal sebelumnya yaitu berupa sinyal acak, namun sinyal ini belum didekripsi.

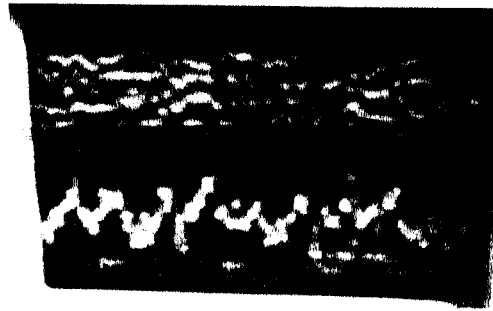


Gambar 4.3. Sinyal dekripsi dengan mode 1

Pada Gambar 4.3. diatas sinyal yang tadinya telah di enkripsi oleh mode 1 dan *password* 1 kemudian di dekripsikan dengan kunci dan *password* yang sama yaitu mode 1 dan *fix password* 1, kemudian setelah sinyal sinyal didekripsi sinyal langsung dikirim ke DAC untuk diolah menjadi sinyal seperti aslinya sehingga hasilnya dapat terlihat pada Gambar 4.3 diatas.

4.3.3. Pengujian sinyal dengan menggunakan mode 2

Pengujian ini dilakukan dengan cara yang sama seperti mode mode – mode sebelumnya yaitu dengan input sinyal audio dan output akan terlihat pada osiloskop, pada pengujian ini sebelum sinyal di dekripsi terlebih dulu sinyal di enkripsi.



Gambar 4.4. Sinyal enkripsi dengan mode 2

Hasil pengujian pada Gambar 4.4. di atas adalah sangat berbeda dengan mode sebelumnya, karena pada pengujian ini kunci dan *password* enkripsinya berbeda yaitu dengan menggunakan mode 2 dan *fix password 2* sehingga didapat pada gambar di atas yaitu sinyal acak dengan tingkat pengacakan yang lebih tinggi dari mode 1



Gambar 4.5. Sinyal dekripsi dengan mode 2

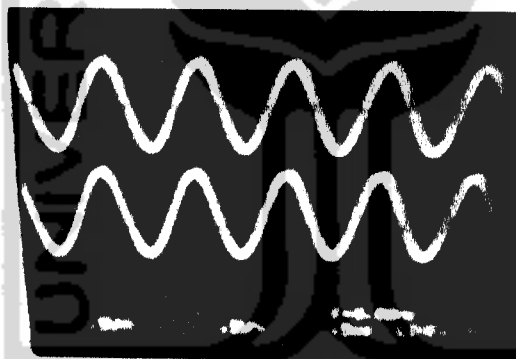
Pada Gambar 4.5. diatas sinyal telah di dekripsi dengan kunci dan *password* yang sama yaitu mode 2 dan *fix password 2* dan sinyal yang telah di dekripsi kemudian langsung dikirim ke DAC untuk diolah menjadi sinyal yang mirip seperti aslinya yaitu sinyal audio.

4.4. Pengujian sinyal dengan menggunakan AFG

Pengujian ini dilakukan untuk mengamati sinyal hasil dekripsi, dengan input yang besarnya antara 1 – 3 KHz dan keluaran terlihat pada osiloskop, dalam pengujian ini jika sinyal masukan tidak mengalami proses enkripsi maka sinyal keluaran akan terlihat mirip seperti bentuk sinyal aslinya yaitu bentuk sinyal sinus tetapi dalam hal ini sinyal keluarannya akan terlihat berupa titik sampling sehingga pada pengujian ini kita akan mengamati titik sampling/1 gelombangnya.

4.4.1 Pengujian sinyal dengan mode normal

Pengujian dilakukan dengan menggunakan AFG sebagai masukan, dengan besarnya frekuensi adalah 1 KHz dan keluaran akan terlihat pada osiloskop.


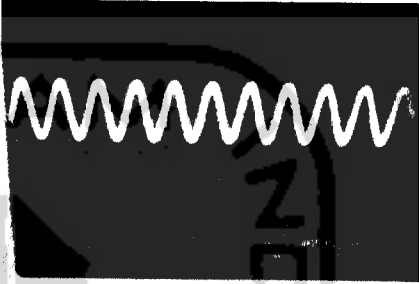
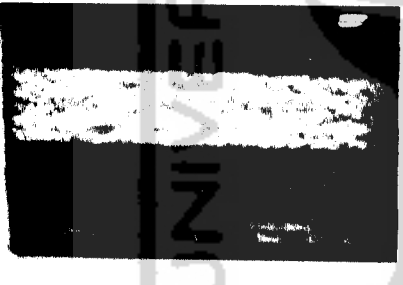
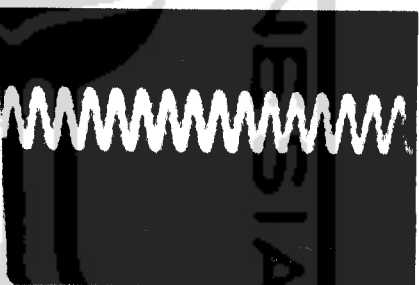
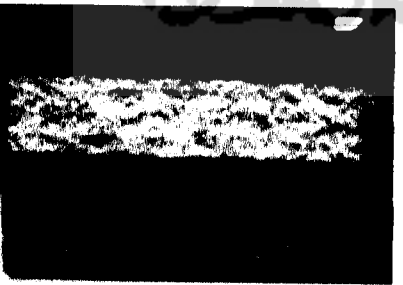
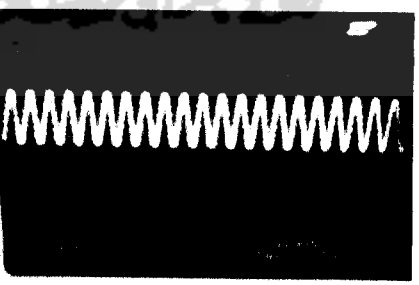


Gambar 4.6. Sinyal dekripsi mode normal

Pada Gambar 4.6. di atas diperoleh sinyal dalam bentuk yang sama persis, ini disebabkan karena dalam pengujian ini mode dalam keadaan normal sehingga tidak ada proses enkripsi dan dekripsi, sehingga di peroleh sinyal output yang sama seperti sinyal aslinya

4.4.2. Pengujian sinyal dengan menggunakan mode 1

Pengujian dilakukan masih dengan cara yang sama yaitu engan menggunakan AFG sebagai masukan dengan besarnya frekuensi sebesar 1 KHz dan keluaran akan terlihat pada osiloskop


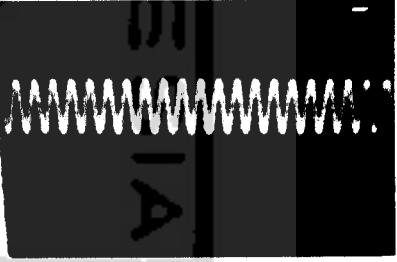
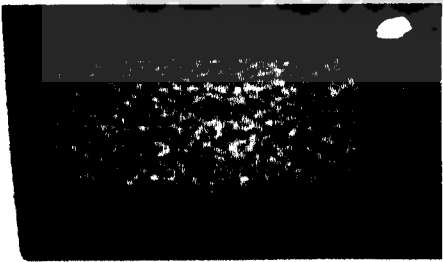

Frekuensi	Enkripsi mode 1	Dekripsi mode 1
1KHz	 V = 0.608 Volt	 V = 0.614 Volt
2 KHz	 V = 0.584 Volt	 V = 0.588 Volt
3 KHz	 V = 0.541 Volt	 V = 0.550 Volt

Gambar 4.7. Sinyal enkripsi dan dekripsi dengan mode 1

Mula – mula sebelum sinyal dikirim, sinyal terlebih dahulu diacak/dienkripsi dengan menggunakan mode 1 dan *password* 1 sehingga hasilnya terlihat dalam gambar 4.7. diatas. Setelah sinyal dienkripsi, kemudian sinyal dikembalikan lagi dengan menggunakan metode dekripsi yaitu dengan menggunakan mode 1 dan *fix password* 1, setelah di dekripsi sinyal langsung dikirim ke DAC untuk bisa dikembalikan lagi menjadi sinyal seperti aslinya, sehingga terlihat pada Gambar 4.8 diatas.

4.4.3. Pengujian sinyal dengan menggunakan mode 2

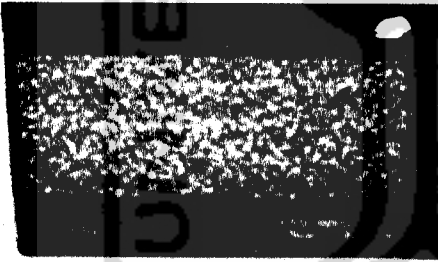
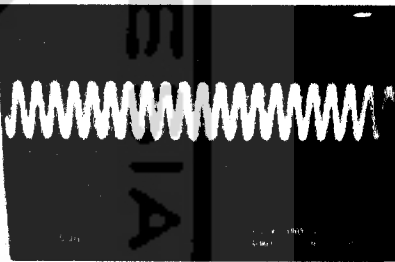
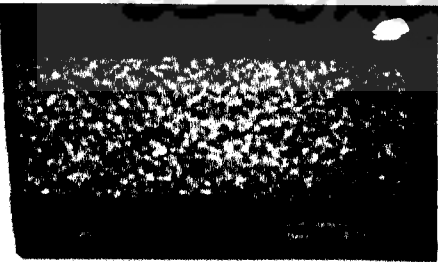
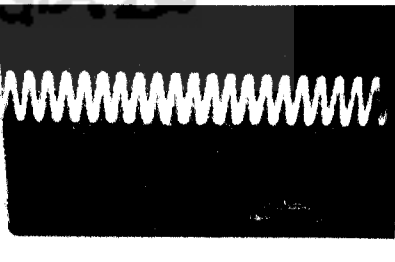
Pengujian dilakukan masih dengan cara yang sama yaitu engan menggunakan AFG sebagai masukan dengan besarnya frekuensi sebesar 1 KHz dan keluaran akan terlihat pada osiloskop.

Frekuensi	Enkripsi mode 1	Dekripsi mode 1
1 KHz	 <p>V = 0.607 Volt</p>	 <p>V = 0.612 Volt</p>
2 KHz	 <p>V = 0.604 Volt</p>	 <p>V = 0.586 Volt</p>

Mula – mula sebelum sinyal dikirim, sinyal terlebih dahulu diacak/dienkripsi dengan menggunakan mode 1 dan *password* 1 sehingga hasilnya terlihat dalam gambar 4.7. diatas. Setelah sinyal dienkripsi, kemudian sinyal dikembalikan lagi dengan menggunakan metode dekripsi yaitu dengan menggunakan mode 1 dan *fix password* 1, setelah di dekripsi sinyal langsung dikirim ke DAC untuk bisa dikembalikan lagi menjadi sinyal seperti aslinya, sehingga terlihat pada Gambar 4.7 diatas.

4.4.3. Pengujian sinyal dengan menggunakan mode 2

Pengujian dilakukan masih dengan cara yang sama yaitu engan menggunakan AFG sebagai masukan dengan besarnya frekuensi sebesar 1 KHz dan keluaran akan terlihat pada osiloskop.

Frekuensi	Enkripsi mode 1	Dekripsi mode 1
1 KHz	 <p>V = 0.607 Volt</p>	 <p>V = 0.612 Volt</p>
2 KHz	 <p>V = 0.604 Volt</p>	 <p>V = 0.586 Volt</p>

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan perancangan sistem, pengujian, dan hasil analisa yang di dapat maka dalam perancangan decriptor simetrik sinyal audio berbasis ATmega 16 ini dapat disimpulkan beberapa hal, yaitu :

1. Perancangan decriptor simetrik sinyal audio berbasis ATmega 16 berfungsi untuk mengembalikan sinyal yang telah dienkripsi atau diacak yaitu berupa *chipper digital audio* menjadi sinyal dalam bentuk aslinya yaitu sinyal audio.
2. Dalam perancangan ini terdapat dua buah kunci dan dua buah *password* untuk mendekripsi data sinyal yang telah dienkripsi.
3. Dengan memanfaatkan logika yang sederhana maka data sinyal yang telah di enkripsi dapat dekripsi menjadi data asli.
4. Dalam perancanganya DAC dapat di fungsikan untuk pengubahan sinyal digital audio menjadi sinyal audio analog .
5. Dari hasil pengujian sinyal *chipper* digital mampu di dekripsi menjadi sinyal digital audio, yang kemudian sinyal digital audio diolah lagi oleh DAC untuk menjadi sinyal seperti aslinya yaitu berupa sinyal audio, kemudian sinyal audio diperkuat oleh *op-amp* agar bisa didengar oleh penerima.

5.2. Saran

Untuk mengembangkan sistem dimasa yang akan datang maka dapat disarankan beberapa hal sebagai berikut :

1. Perancangna decriptor simetrik sinyal audio berbasis ATmega 16 dapat di kembangkan lagi dengan menggunakan input yang berbeda selain sinyal audio yaiu input berupa teks yang kemudian untuk outputnya akan tertampil di layar LCD.
2. Metode yang digunakan dapat dikembangkan lagi menjadi decriptor Asimetrik.
3. Dalam perancangan dekripsi kunci dan *password* yang di gunakan dapat dikembangkan lagi dengan ketentuan sendiri.



DAFTAR PUSTAKA

1. Budioko, Totok., 2005. *Belajar dengan mudah Pemrograman Bahasa C dengan SDCC [Small Device C Compiler] Pada Mikrokontroller AT 89X051 / AT 89C51 / 52 Teori simulasi dan aplikasi*, Yogyakarta : Gava Media.
2. Nurdiansyah, Yanuar., 2006, *Termometer Badan Dengan Output Suara Berbasis Mikrokontrol AT 89S51*, Yogyakarta.: Skripsi UII.
3. Proakis, John & Manolakkis, Dimitris., 1997. *Pemrosesan sinyal digital*, Jakarta : Penerbit Prehalindo.
4. Raharjo, Budi, 2006. *Pemrograman C++*, Bandung : Penerbit Informatika.
5. Wahana komputer Semarang, *Memahami Model Enkripsi dan Security Data* Yogyakarta : Penerbit Abdi.
6. Wardana, Lingga., 2006. *Belejar Sendiri Mikrokontrol AVR Seri ATMega 8535 Simulasi, Hardware dan Aplikasi*, Yogyakarta : Penerbit Andi.
7. Willa, Lukas, Drs, 2007. *Teknik Digital, Mikroprosesor, Mikrokomputer*, Bandung : Penerbit Informatika.
8. www.atmel.com/
9. www.dg_ware.com/
10. www.goole.com/

```
////////////////////////////////////  
//Program : Decryptor Data Audio  
//By : Wandi Sulistiyohadi  
//University : UII
```

```
//Mikrokontroler : Atmel ATmega16  
//X-tal : 16 MHz  
////////////////////////////////////
```

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

```
#define normal 0  
#define On 0  
#define Off 1
```

```
unsigned char OutputBefore, InputBefore, InputBefore1, InputBefore2;  
unsigned char temp, mode, Input;  
bit ModeBaru;
```

```
//Eksternal Interrupt 2 service routine  
Interrupt [EXT_INT2] void ext_int2_isr(void)  
{  
ModeBaru = 1  
}
```

```
Void main (void)  
{
```

```
////////////////////////////////////  
//
```

```
// Inisialisasi awal
```

```
////////////////////////////////////
```

```
// Inisialisasi input/Output Ports
```

```
// Port A initialization
```

```
// Func7=Out Func6= Out Func5= Out Func4= In Func3= In Func2= In Func1= In  
Func0=In
```

```
//State7=0 State6=0 State5=0 State4=T State3=T State2=T State1=T State0=T
```

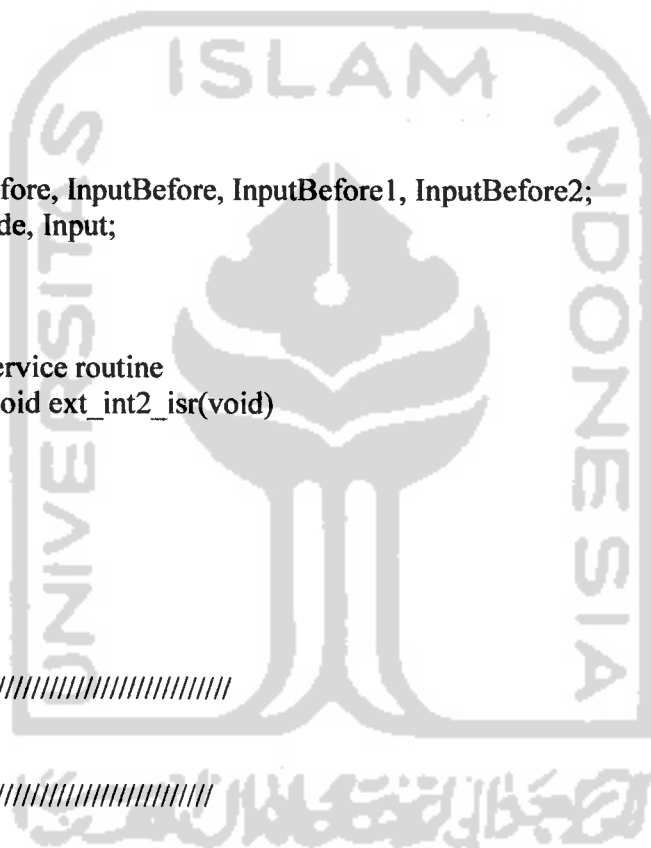
```
PORTA=0x00;
```

```
DDRA=0xE0;
```

```
// Inisialisasi Port B
```

```
// Semua sebagai Masukan
```

```
// State7=T State6=T State5=T State4=T State3=T State2=P State1=P State0=T
```



```

PORTB=0x06;
DDRB=0x00;

// Inialisasi Port C
// Semua sebagai Output
// Nilai awal 0
PORTC=0x00;
DDRC=0xFF;

// Inialisasi Port D
// Semua sebagai Inputan
// Semua pada kondisi tri state
PORTD=0x00;
DDRD=0x00;

// Inialisasi Eksternal Interrupt(s)
// INT0: Off
// INT1: Off
// INT2: On
// INT2 Mode: Falling Edge
GICR |=0x20;
MCUCR=0x00;
MCUCSR=0x00;
GIFR=0x20;

// Global enable interrupts
#asm("sei")

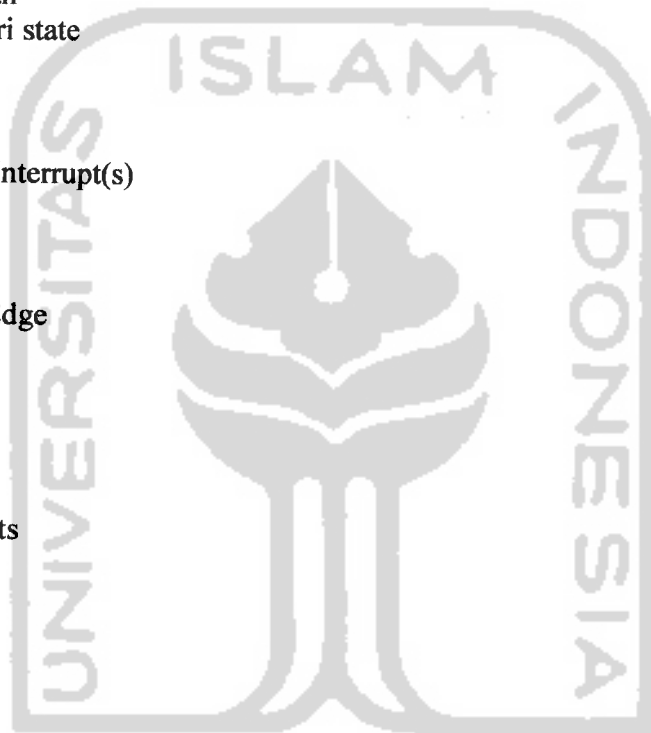
Mode = Normal;
While (1)
{

//Pengecekan Data Tombol
If((ModeBaru == 1)&(PINB.2 == 0)){

    While(1){
        Delay_ms(100);

        If(PINB.2 == 1)break;
    }
    Mode= Mode + 1;
}
If(Mode== 3) Mode = Normal;
ModeBaru = 0;

```



```
// Display Mode
```

```
    //delay_ms(200);
    If(Mode== Normal) {
        PORTA.7 = 0; //On;
        PORTA.6 = 1; //Off;
        PORTA.5 = 1; // Off;
        //delay_ms(200);
    }
    Else if(Mode==1) {
        PORTA.7 = 1; //Off;
        PORTA.6 = 0; // On;
        PORTA.5 = 1; // Off;
        //delay_ms(200);
    }
    Else if(Mode==2) {
        PORTA.7 = 1; //Off;
        PORTA.6 = 0; // On;
        PORTA.5 = 0; // On;
        //delay_ms(200);
    }
}
```

```
// Proses Dekripsi Data Audio
```

```
    If(Mode == 1) {

        //InputBefore/Password = 01
        While(1){
            Input = PIND;
            If(inputBefore != Input){
                Temp = Input ^ InputBefore ;
                //if(input == 0xFF)break;
                //if(k == 256)k = 0;
                //PORTD = 0xFF; //Mask output
                //PORTD = Input ^ RandomCaesar[k];
                //temp = temp ^ 0xAA;
            PORTC = temp;
            InputBefore = Input;
            }

            If(ModeBaru == 1) break;
        }
    }
}
```

```

Else if(Mode == 2){

    //InputBefore/Pasword = 02
    While(1){
        Input = PIND;

        If(inputBefore != Input){

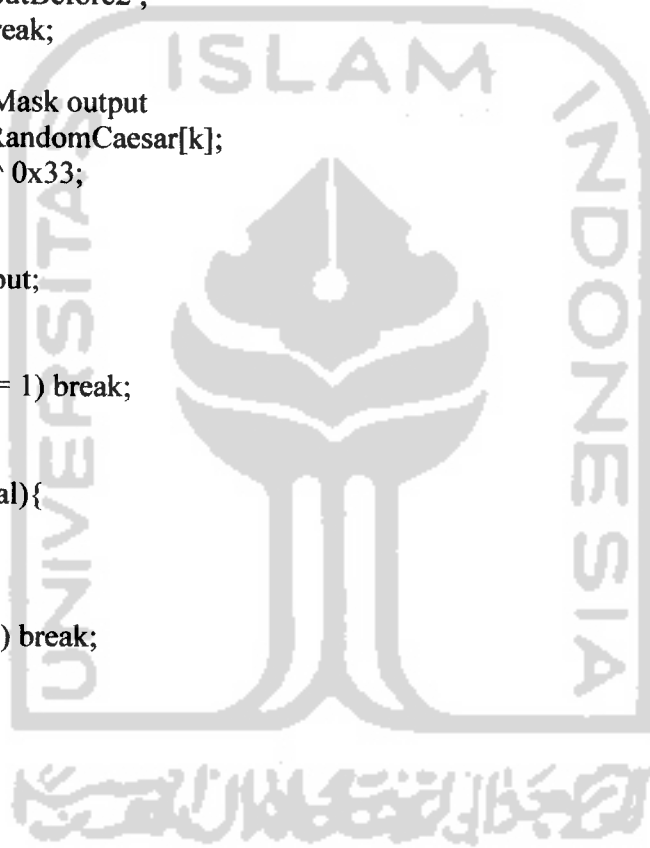
            InputBefore2 = InputBefore1;
            InputBefore1 = InputBefore;

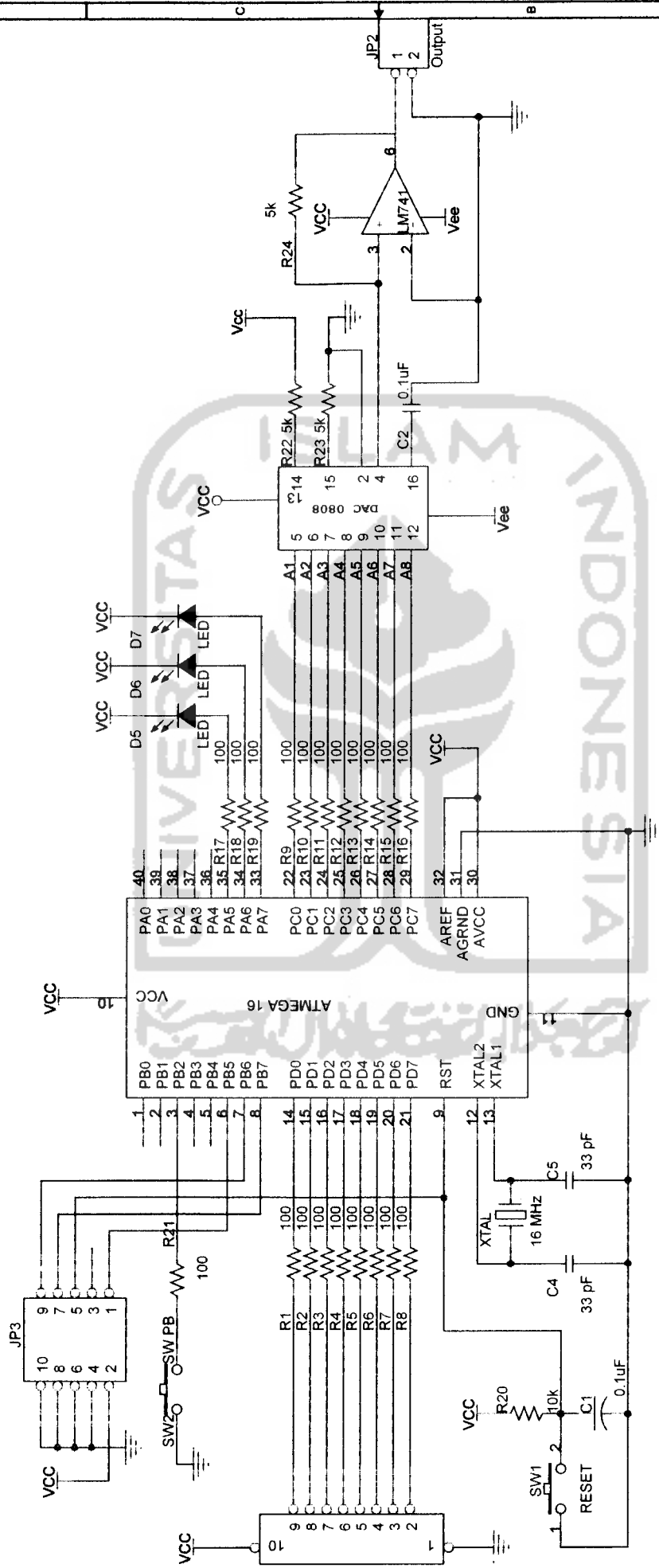
            Temp = Input ^ InputBefore2 ;
            //if(input == 0xFF)break;
            //if(k == 256)k = 0;
            //PORTD = 0xFF; //Mask output
            //PORTD = Input ^ RandomCaesar[k];
            //temp = temp ^ 0x33;
            PORTC = temp;

            InputBefore = Input;
        }

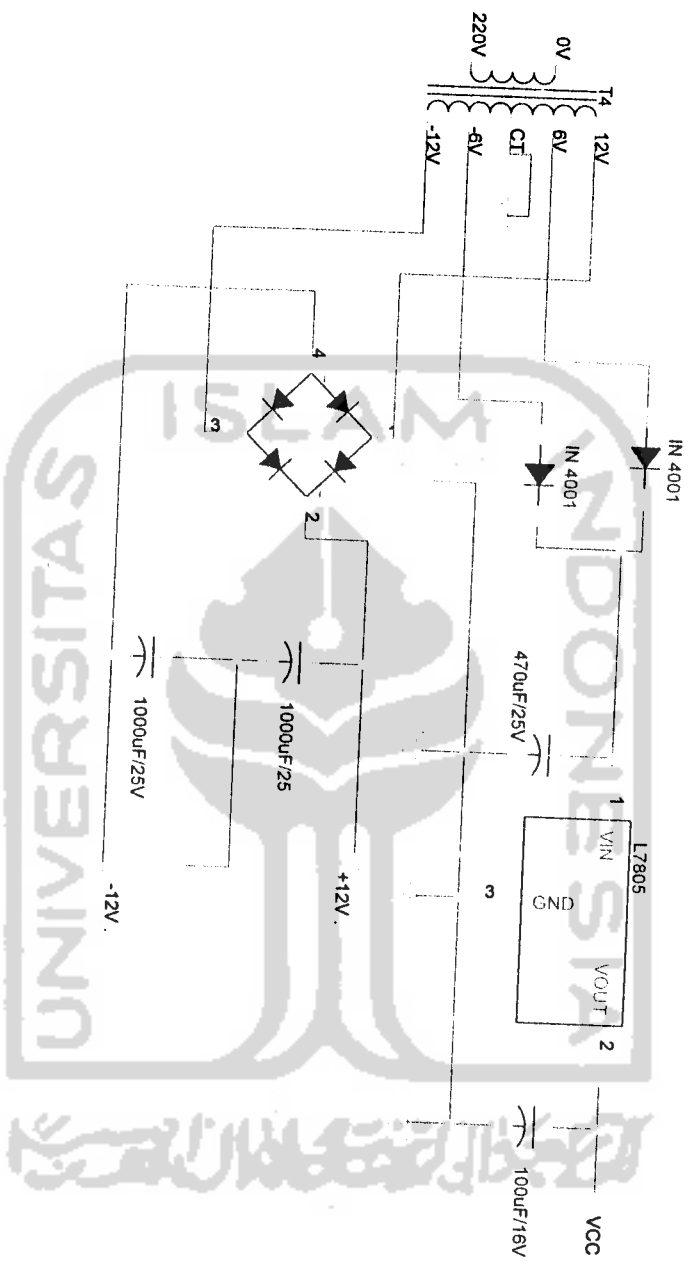
        If(ModeBaru == 1) break;
    }
}
Else if(Mode == Normal){
    While(1){
        Input = PIND;
        PORTC = Input;
        If(ModeBaru == 1) break;
    }
}

```





Title		RANGKAIAN DEKRIPSI	
Size	A	Document Number	Whandi Sulistiyohadi
Date:		Sheet	1 of 1



Title	Rangkaian catu daya
Size	Document Number