

BAB II

KAJIAN PUSTAKA

2.1. Sterilisasi

Sterilisasi yaitu proses membunuh semua mikroorganisme termasuk spora bakteri pada benda yang telah didekontaminasi dengan tepat. Tujuan sterilisasi yaitu untuk memusnahkan semua bentuk kehidupan mikroorganisme patogen termasuk spora, yang mungkin telah ada pada peralatan kedokteran dan perawatan yang dipakai. Hal yang perlu dipertimbangkan dalam memilih metode sterilisasi yaitu sifat bahan yang akan disterilkan.

Terdapat beberapa cara untuk sterilisasi, yaitu:

2.1.1. Pemanasan

a. Dengan pemanasan kering

a) Pembakaran

Alat yang digunakan adalah lampu spiritus. Pembakaran dapat dilakukan dengan cara :

(a) Memijarkan

Pembakaran dengan cara ini hanya cocok untuk alat-alat logam (ose, pinset), yang dibiarkan sampai memijar. Dengan cara ini seluruh mikroorganisme, termasuk spora, dapat dibasmi.

(b) Menyalakan

Dapat diartikan suatu pelintasan alat gelas (ujung pinset, bibir tabung, mulut *erlenmeyer*) melalui nyala api. Cara ini merupakan hal darurat dan tidak memberikan jaminan bahwa mikroorganisme yang melekat pada alat dengan pasti terbunuh.

b) Dengan udara panas (*hot air oven*)

Cara ini menggunakan udara yang dipanaskan dan kering, serta berlangsung dalam sterilisator udara panas (*oven*). Pemanasan dengan udara panas digunakan untuk sterilisasi alat-alat laboratorium dari gelas misalnya : petri, tabung gelas, botol pipet juga untuk bahan-bahan minyak dan *powder* misalnya *talk*. Bahan dari karet, kain, kapas dan kasa tidak dapat disterilkan dengan cara ini.

b. Dengan pemanasan basah

- a) Dengan merebus
- b) Dengan uap air panas
- c) Dengan uap air panas bertekanan (*autoclave*)

2.1.2. Filtrasi

Sterilisasi dengan cara ini dilakukan dengan mengalirkan cairan atau gas pada saringan berpori kecil sehingga dapat menahan mikroorganisme dengan ukuran tertentu. [7]

Kegunaan:

- a. untuk sterilisasi media yang tidak tahan terhadap pemanasan, misalnya *Urea Broth* ataupun untuk sterilisasi vaksin, serum, enzim, vitamin.
- b. Meminimalkan kuman udara masuk untuk ruangan kerja secara aseptis. Virus seperti mikroorganisme tanpa dinding sel (mikroplasma) umumnya tidak dapat ditahan oleh filter.[7]

2.1.3. Penyinaran dengan sinar gelombang pendek (radiasi)

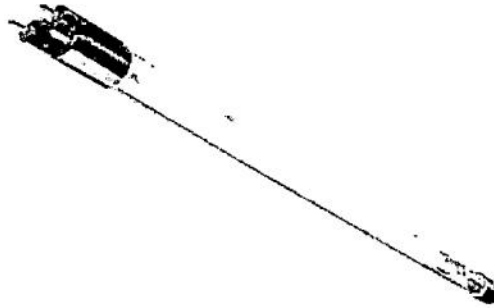
Sterilisasi dengan cara ini diperlukan jika sterilisasi panas maupun dinding tidak dapat dilakukan. Beberapa macam radiasi mengakibatkan letal terhadap sel-sel jasad renik dan mikroorganisme lain. Sinar x mempunyai daya penetrasi lebih besar dibanding dengan sinar *ultraviolet*. Sinar gamma mempunyai daya penetrasi lebih besar dibandingkan dengan sinar x dan digunakan untuk mensterilkan material yang tebal, misalnya bungkusan alat-alat kedokteran atau paket makanan. Sinar katoda biasa dipakai menghapus hama pada suhu kamar terhadap barang-barang yang telah dibungkus.[7]

2.1.4. Kimia

Merupakan cara sterilisasi dengan bahan kimia.

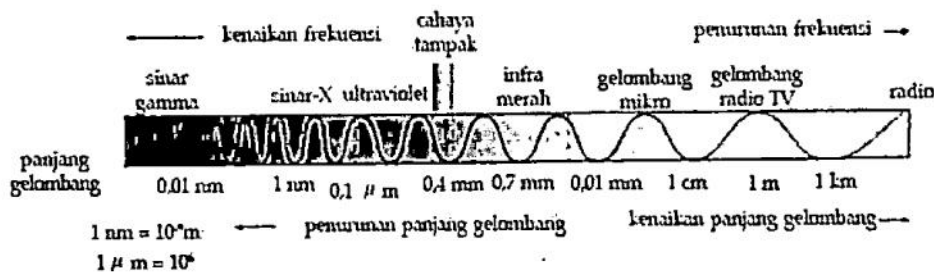
2.2. Sinar Ultraviolet

Di bawah ini adalah gambar dari lampu UV



Gambar 2.1. Lampu UV

Nama *Ultraviolet*, atau sering disingkat *UV* berasal dari bahasa latin 'ultra' dan 'violet'. *Violet* adalah warna dari panjang gelombang terpendek dari spektrum sinar tampak. Sinar *UV* memiliki panjang gelombang yang lebih pendek dari sinar *violet* tersebut (Gambar 2.2.). Matahari mengemisi radiasi *ultraviolet* dalam tiga panjang gelombang, *UV A*, *UV B* dan *UV C*, tapi 99% radiasi yang mencapai bumi hanya *UV A* dikarenakan adanya absorpsi oleh lapisan ozon di atmosfer (Wikipedia, 2007). [9]



Gambar 2.2. Spektrum gelombang elektromagnetik dan panjang gelombangnya.

Sinar *UV* memiliki panjang gelombang yang lebih pendek dari 400nm.

Sinar *UV* terbagi atas 3 jenis, yaitu:

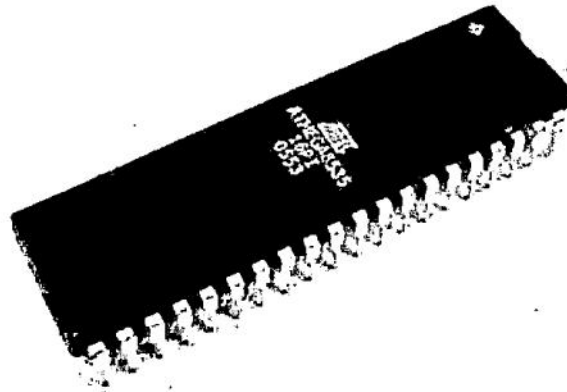
- a. *UV A*, umumnya dikenal sebagai *UV* gelombang panjang, *near-ultraviolet*, *black-light*, atau *Wood's light* dengan panjang gelombang 320-420 nm.
- b. *UV B*, juga dikenal sebagai *UV* gelombang sedang dengan panjang gelombang 280-320 nm
- c. *UV C*, sering direferensikan sebagai gelombang pendek, *far ultraviolet*, atau *germicidal UV* dengan panjang gelombang 180-280 nm.

Sinar *ultraviolet* mempunyai panjang gelombang 15-390 nm. Lampu sinar *ultraviolet* dengan panjang gelombang 260 – 270 nm, dimana sinar dengan panjang gelombang sekitar 265 nm mempunyai daya bakterisid yang tinggi. Lampu *ultraviolet* digunakan untuk mensterilkan ruangan, misalnya di kamar bedah, ruang pengisian obat dalam ampul dan flakon di industri farmasi, juga bisa digunakan diperusahaan makanan untuk mencegah pencemaran permukaan.[8]

Sterilisasi ruangan dengan sinar *ultraviolet* dapat dinilai keberhasilannya dengan mengukur kualitas udara ruangan. Indikator yang digunakan adalah angka kuman udara ruang. Menurut Kepmenkes Nomor 1204/Menkes/SK/X/2004 tentang persyaratan kesehatan lingkungan rumah sakit, ditetapkan bahwa angka kuman udara ruang operasi harus < 10 *CFU/M3*. [8]

2.3. Mikrokontroler AVR ATmega8535

Di bawah ini adalah gambar dari mikrokontroler ATmega8535



Gambar 2.3. Mikrokontroler AVR ATmega8535

Mikrokontroler *AVR* di buat berdasarkan *architecture RISC (Reduced Instruction Set Computing)* terbaru untuk meningkatkan kecepatan, ukuran program dan penggunaan catu daya. *AVR* telah berhasil menggabungkan *fast access register file* dan *single cycle intruction* dengan *32 register x 8 bit*. Dengan *32 register AVR* dapat mengeksekusi beberapa instruksi sekali jalan (*single cycle*). Enam dari *32 register* yang ada dapat digunakan sebagai *indirect address register pointer 16 bit* untuk pengalamatan *data space*, yang memungkinkan penghitungan alamat yang efisien. *AVR* mempunyai kecepatan dari *0-16 Mhz* bahkan *AVR* yang telah ditambahkan beberapa alat dapat mencapai kecepatan *20 Mhz*. *AVR* merupakan mikrokontroler yang sangat *powerful* dan efisien dalam *addressing code* karena *AVR* dapat mengakses program memori dan data memori.[2]

AVR secara umum terbagi dua jenis yaitu *high-end* dan *low-end*

performance untuk varian ATmega perbedaan dapat di lihat pada akhiran nomor seri setiap AVR seperti tipe ATmega8535 dan ATmega8535L setiap tipe yang berakhiran L merupakan versi *low-end* dari AVR, yang menjadi perbedaan mencolok hanyalah *operating voltage* dan *speed*. Secara umum, AVR dapat dikelompokkan menjadi 4 kelas, yaitu keluarga ATtiny, keluarga AT90xx, keluarga ATmega, dan AT86RFxx. Pada dasarnya yang membedakan masing-masing kelas adalah memori, peripheral, dan fungsinya. Dari segi arsitektur dan instruksi yang digunakan, mereka bisa dikatakan hampir sama.

Keistimewaan Mikrokontroler AVR ATmega8535:

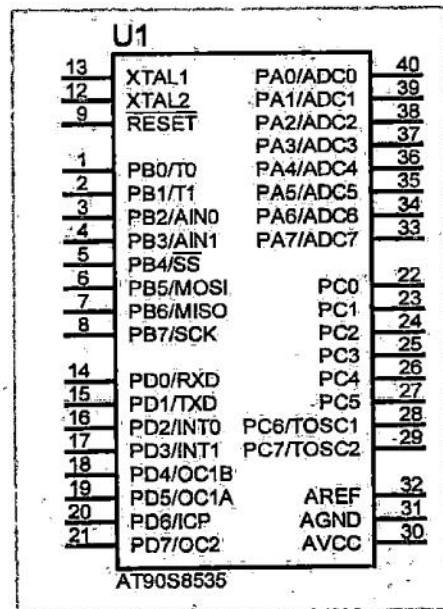
- 8 bit CPU sebagai pusat pengendalian aplikasi.
- Mempunyai 130 instruksi.
- 32 register umum yang terhubung dengan ALU (*Arithmetic Logic Unit*).
- Kemampuan memproses instruksi sampai 16 MIPS (*million instruction per second*) secara *bit per bit*.
- Memiliki 8 Kbyte untuk *Flash* dalam untuk menyimpan program dan dapat ditulis ulang hingga 10.000 kali.
- Memiliki 512 Bytes EEPROM dengan *endurance* : 100,000 *Write/Erase Cycles*.
- Memiliki 512 Bytes Internal SRAM (*Static Random Access Memory*) digunakan untuk menyimpan sementara data dari program *flash*.
- ADC (*Analog To Digital Converter*) internal dengan *fidelitas* 10 bit

sebanyak 8 channel.

- 32 jalur I/O (Input/Output) yang terpisah dalam empat port yaitu A, port B, port C, dan Port D.
- 16 bit timer/counter dan 8 bit timer/counter.
- Full Duplex Universal Synchronous Asynchronous Receiver Transmitter (USART).
- RTC (Real Time Clock) dengan osilator terpisah.
- SPI (Serial Peripheral Interface) untuk komunikasi serial yang memiliki kecepatan yang relatif tinggi pada jarak dekat.
- Enam pilihan mode sleep menghemat penggunaan daya listrik.
- Watchdog timer yang dapat diprogram dengan osilator internal.

Dalam penelitian ini digunakan Mikrokontroler AVR ATmega8535

dari Atmel, dimana konfigurasinya sebagai berikut :



Gambar 2.4. IC ATmega8535

Penjelasan dari masing-masing pin adalah sebagai berikut :

- 1) Pin 1 sampai 8 (*Port B*) merupakan *port* paralel 8 bit dua arah (*bidirectional*), yang dapat digunakan untuk berbagai keperluan (*general purpose*) dan (*special feature*).
- 2) Pin 9 (*Reset*) jika terdapat minimum *pulse* pada saat *active low*.
- 3) Pin 10 (*VCC*) dihubungkan ke *Vcc* (+5 Volt).
- 4) Pin 11 dan 31 (*GND*) dihubungkan ke *Vss* atau *Ground*.
- 5) Pin 12 (*XTAL 2*) adalah pin masukan ke rangkaian osilator *internal*. Sebuah osilator kristal atau sumber osilator luar dapat digunakan.
- 6) Pin 13 (*XTAL 1*) adalah pin keluaran ke rangkaian osilator *internal*. Pin ini dipakai bila menggunakan osilator kristal.
- 7) Pin 14 sampai 21 (*Port D*) adalah 8-bit dua arah (*bi-directional I/O port*) dengan *internal pull-up resistors* (memilih setiap *bit*) digunakan untuk *general porpouse* dan *special feature*.
- 8) Pin 22 sampai 29 (*Port C*) adalah 8-bit dua arah (*bi-directional I/O port*) dengan *internal pull-up resistors* (memilih setiap *bit*) digunakan untuk *general porpouse* dan *special feature*.
- 9) Pin 30 adalah *Avcc* pin penyuplai daya untuk *port A* dan *A/D converter* dan dihubungkan ke *Vcc*. Jika *ADC* digunakan maka pin ini dihubungkan ke *Vcc* dengan *low pas filter*.
- 10) Pin 32 adalah *A REF* pin yang berfungsi sebagai referensi untuk pin analog jika *A/D Converter* digunakan.
- 11) Pin 33 sampai 40 adalah 8-bit dua arah (*bi-directional I/O port*)

dengan internal *pull-up resistors* (memilih setiap *bit*) digunakan untuk *general purpose*.

Di dalam mikrokontroler, terdapat banyak fungsi-fungsi khusus.

Fungsi- fungsi khusus tersebut antara lain :

1) Interupsi

Interupsi adalah perintah untuk menjalankan satu program *subroutine* pada saat menjalankan program utama, sifat *subroutine* ini memaksa dan harus harus dilayani. Setelah program interupsi selesai, maka mikrokontroler akan kembali ke program awal yang ditinggalkan.

Jenis-jenis interupsi :

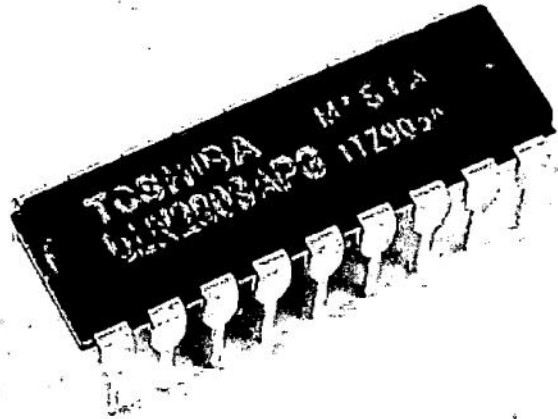
- a. Interupsi yang tidak dapat dihalangi (*Non Mascable interrupt*), misalnya *reset*.
- b. Interupsi yang dapat dihalangi (*Mascable interrupt*), misalnya *INT0, INT1, Timer1, Timer0, Serial port (internal)*.

2) *Timer / Counter*

Di dalam *chip AVR ATmega8535*, terdapat dua buah *timer/counter* 8 dan 16 *bit*. Supaya berfungsi sebagai *timer AVR* menggunakan *internal signal clock*, untuk berfungsi sebagai *counter AVR* menggunakan sinyal eksternal yang terdapat pada *port pin* yang berfungsi sebagai *timer* dan *counter*. [2]

2.4. IC ULN 2803

Di bawah ini adalah gambar IC ULN 2803

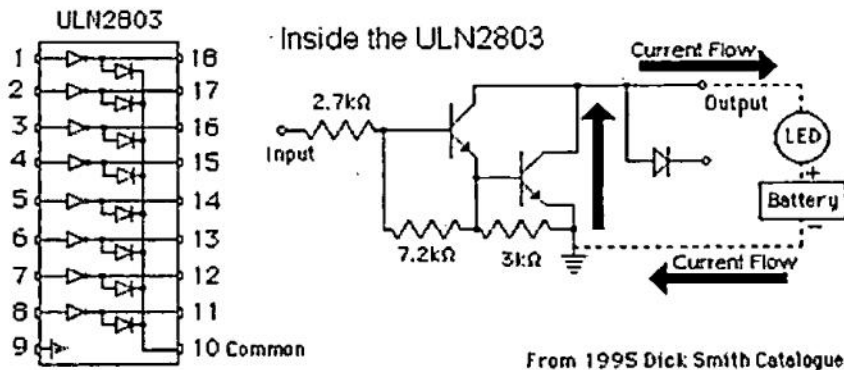


Gambar 2.5. IC ULN 2803

IC ULN2803 adalah IC yang didalamnya merupakan susunan transistor NPN yang terpasang secara *darlington* untuk mengendalikan sebuah beban yang terkontrol dan dapat menangani/mengalirkan arus sebesar 500 mA. Setiap ULN2803 terdapat delapan buah susunan *darlington* yang dapat bekerja secara individu atau terpisah sehingga beban yang dapat dipasang pada ULN2803 sebanyak 8 buah. ULN2803 dapat bekerja pada tegangan 50 volt. IC ULN2803 ini dapat digunakan untuk mengaktifkan beban yang terpasang dari sumber tegangan positif (VCC).[1]

Darlington adalah suatu bentuk penguat gandengan langsung yang sering digunakan, dimana emitor dari satu transistor dihubungkan dengan basis dari transistor berikutnya.

Aplikasi IC ULN2803 sebagai *driver* adalah untuk mengendalikan *relay*, lampu DC atau *LED*, dan untuk sistem pensaklaran yang lain. Pada tugas akhir kali ini penulis menggunakan IC ULN2803 sebagai *ic driver* untuk mengaktifkan *relay*. Berikut ini merupakan konfigurasi pin ULN2803 beserta susunan transistor *darlington* yang terdapat di dalamnya.[1]

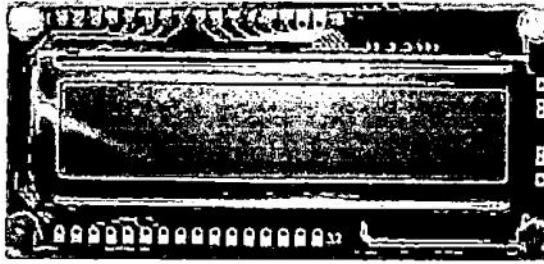


Gambar 2.6. Konfigurasi IC ULN 2803

2.5. LCD (Liquid Crystal Display)

LCD merupakan komponen *display* yang dapat menampilkan berbagai macam karakter. Jenis LCD ada berbagai macam, dan yang paling sering digunakan adalah LCD Karakter 16 x 2.[3]

Modul *LCD character* 16 x 2 dapat dengan mudah dihubungkan dengan mikrokontroler seperti AVR ATmega8535 atau sejenisnya. LCD yang akan di gunakan mempunyai lebar *display* 2 baris 16 kolom atau biasa disebut sebagai *LCD Character* 16 x 2, dengan 16 pin konektor, yang didefinisikan sebagai berikut:



Gambar 2.7. Modul LCD Karakter 16 x 2

Tabel 2.1. Pin dan Fungsi Modul LCD Karakter 16 x 2

| PIN | NAME | FUNCTION |
|------------|-------------|---------------------------------------------------------------------------------------------------|
| 1 | <i>VSS</i> | <i>Ground Voltage</i> |
| 2 | <i>VCC</i> | + 5V |
| 3 | <i>VEE</i> | <i>Contrast Voltage</i> |
| 4 | <i>RS</i> | <i>Register Select</i> 0 = <i>Instruction Register</i> 1 = <i>data Register</i> |
| 5 | <i>R/W</i> | <i>Read /Write, to choose write or read mode</i> 0 = <i>write mode</i> 1 = <i>read mode</i> |
| 6 | <i>EN</i> | <i>Enable</i> 0 = <i>start to lacht data to LCD character</i> 1 = <i>disable</i> |
| 7 | <i>DB0</i> | <i>LSB</i> |
| 8 | <i>DB1</i> | - |
| 9 | <i>DB2</i> | - |
| 10 | <i>DB3</i> | - |
| 11 | <i>DB4</i> | - |
| 12 | <i>DB5</i> | - |
| 13 | <i>DB6</i> | - |
| 14 | <i>DB7</i> | <i>MSB</i> |
| 15 | <i>BPL</i> | <i>Black Panel Light</i> |
| 16 | <i>GND</i> | <i>Ground Voltage</i> |

Display karakter pada LCD diatur oleh pin *EN*, *RS* dan *RW* dengan rincian sebagai berikut:

Jalur *EN* dinamakan *Enable*. Jalur ini digunakan untuk memberitahu *LCD* bahwa anda sedang mengirimkan sebuah data. Untuk mengirimkan

data ke *LCD*, maka melalui program *EN* harus dibuat logika *low* "0" dan set pada dua jalur kontrol yang lain *RS* dan *RW*. Ketika dua jalur yang lain telah siap, set *EN* dengan logika "1" dan tunggu untuk sejumlah waktu tertentu (sesuai dengan datasheet dari *LCD* tersebut) dan berikutnya set *EN* ke logika *low* "0" lagi.[2]

Jalur *RS* adalah jalur *Register Select*. Ketika *RS* berlogika *low* "0", data akan dianggap sebagai sebuah perintah atau instruksi khusus (seperti *clear screen*, posisi kursor dll). Ketika *RS* berlogika *high* "1", data yang dikirim adalah data *text* yang akan ditampilkan pada *display LCD*. Sebagai contoh, untuk menampilkan huruf "T" pada layar *LCD* maka *RS* harus diset logika *high* "1".

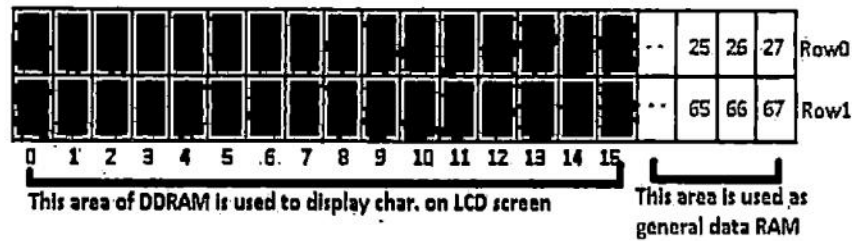
Jalur *RW* adalah jalur kontrol *Read/ Write*. Ketika *RW* berlogika *low* (0), maka informasi pada bus data akan dituliskan pada layar *LCD*. Ketika *RW* berlogika *high* "1", maka program akan melakukan pembacaan memori dari *LCD*. Sedangkan pada aplikasi umum pin *RW* selalu diberi logika *low* "0".

Memori *LCD* memiliki bagian-bagian sebagai berikut

2.5.1. *DDRAM (Display Data RAM)*

Memori *DDRAM* digunakan untuk menyimpan karakter yang akan ditampilkan. Semua teks yang kita tuliskan ke modul *LCD* adalah disimpan didalam memori ini, dan modul *LCD* secara berturutan membaca memori ini untuk menampilkan teks ke modul *LCD* itu sendiri.

DDRAM Address **00000000** For 2-Line LCD :

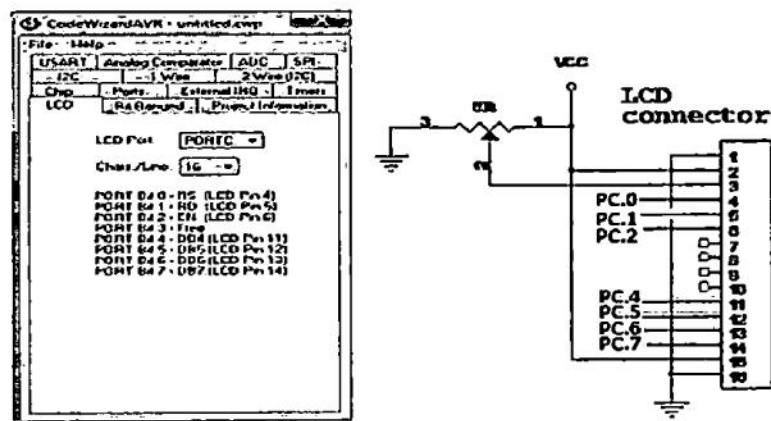


Gambar 2.8. Lokasi Memori *Display LCD* Karakter 16 x 2

Pada peta memori tersebut, daerah yang berwarna hijau (00 s/d 0F dan 10 s/d 1F) adalah *display* yang tampak. Sebagaimana yang anda lihat, jumlahnya sebanyak 16 karakter per baris dengan dua baris. Angka pada setiap kotak adalah alamat memori yang bersesuaian dengan posisi dari layar. Demikianlah karakter pertama di sudut kiri atas adalah menempati alamat 00h. Posisi karakter berikutnya adalah alamat 01h dan seterusnya.[2]

2.5.2. Inisialisasi

Sebelum kita dapat menggunakan modul *LCD*, kita harus melakukan inisialisasi dan mengkonfigurasikannya.



Gambar 2.9. Konfigurasi *Codevision* dan koneksi *hardware*

2.5.3. Fungsi LCD

Fungsi *LCD* terletak pada header *lcd.h* yang harus dimasukkan (*include*) sebelum digunakan. Sebelum melakukan *include* terlebih dahulu sebutkan *port* mikrokontroler yang akan dihubungkan ke *LCD*. Hal ini juga dapat dengan mudah dilakukan dengan menggunakan *CodeWizardAVR*. [2]

```
/* modul LCD dihubungkan dengan PORTC */
```

```
#asm .
```

```
    equ __lcd_port=0x15
```

```
#endasm /*
```

```
Sekarang fungsi LCD dapat di-include*/
```

```
#include <lcd.h>
```

Fungsi-fungsi untuk mengakses *LCD* diantaranya adalah :

a. *unsigned char lcd_init (unsigned char lcd_columns)*

Untuk menginisialisasi modul *LCD*, menghapus layar dan meletakkan posisi karakter pada baris ke-0 kolom ke-0. Jumlah kolom pada *LCD* harus disebutkan (misal 16). Cursor tidak ditampilkan. Nilai yang dikembalikan adalah 1 bila modul *LCD* terdeteksi, dan bernilai 0 bila tidak terdapat modul *LCD*. Fungsi ini harus dipanggil pertama kali sebelum menggunakan fungsi yang lain.

b. *void lcd_clear (void)*

Menghapus layar *LCD* dan meletakkan posisi karakter pada baris ke-0 kolom ke-0.

c. *void lcd_gotoxy (unsigned char x, unsigned char y)*

Meletakkan posisi karakter pada kolom ke-x baris ke-y. Nomor baris dan kolom dimulai dari nol.

d. *void lcd_putchar (char c)*

Menampilkan karakter pada *LCD*.

e. *void lcd_puts (char *str)*

Menampilkan string yang disimpan pada SRAM pada *LCD*. [2]

2.6. Relay

Relay adalah komponen elektronika berupa saklar elektronik yang digerakkan oleh arus listrik. Secara prinsip, *relay* merupakan tuas saklar dengan lilitan kawat pada batang besi (*solenoid*) di dekatnya. Ketika *solenoid* dialiri arus listrik, tuas akan tertarik karena adanya gaya magnet yang terjadi pada *solenoid* sehingga kontak saklar akan menutup. Pada saat arus dihentikan, gaya magnet akan hilang, tuas akan kembali ke posisi semula dan kontak saklar kembali terbuka. *Relay* biasanya digunakan untuk menggerakkan arus/tegangan yang besar (misalnya peralatan listrik 4 *ampere* AC 220 V) dengan memakai arus/tegangan yang kecil (misalnya 0.1 *ampere* 12 Volt DC). *Relay* yang paling sederhana ialah *relay* elektromekanis yang memberikan pergerakan mekanis saat mendapatkan energi listrik.

Secara sederhana *relay* elektromekanis ini didefinisikan sebagai berikut :

- a. Alat yang menggunakan gaya elektromagnetik untuk menutup (atau membuka) kontak saklar.

- b. Saklar yang digerakkan (secara mekanis) oleh daya/energi listrik.

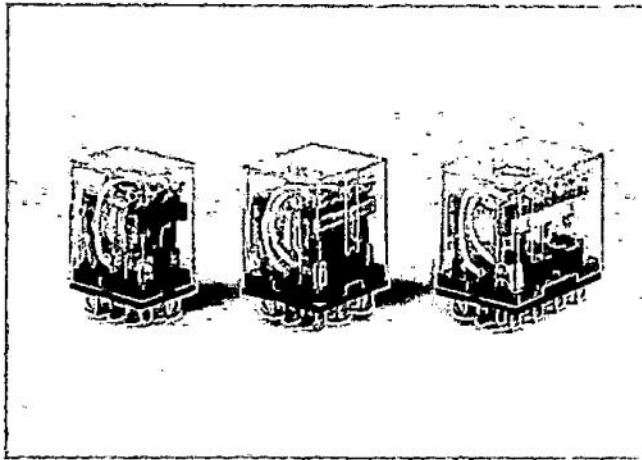
Dalam pemakaiannya biasanya *relay* yang digerakkan dengan arus DC dilengkapi dengan sebuah dioda yang di-paralel dengan lilitannya dan dipasang terbalik yaitu anoda pada tegangan (-) dan katoda pada tegangan (+). Ini bertujuan untuk mengantisipasi hentakan listrik yang terjadi pada saat *relay* berganti posisi dari *on* ke *off* agar tidak merusak komponen di sekitarnya.

Konfigurasi dari kontak-kontak *relay* ada tiga jenis, yaitu:

- a. *Normally Open (NO)*, apabila kontak-kontak tertutup saat *relay* dicatu
- b. *Normally Closed (NC)*; apabila kontak-kontak terbuka saat *relay* dicatu
- c. *Change Over (CO)*, *relay* mempunyai kontak tengah yang normal tertutup, tetapi ketika *relay* dicatu kontak tengah tersebut akan membuat hubungan dengan kontak-kontak yang lain.

Penggunaan *relay* perlu memperhatikan tegangan pengontrolnya serta kekuatan *relay* memutuskan arus/tegangan. Biasanya ukurannya tertera pada *body relay*. Misalnya *relay* 12 VDC/4 A 220 Volt, artinya tegangan yang diperlukan sebagai pengontrolnya adalah 12 Volt DC dan mampu men-switch arus listrik (maksimal) sebesar 4 ampere pada tegangan 220 Volt. Sebaiknya *relay* difungsikan 80% saja dari kemampuan maksimalnya agar aman, lebih rendah lagi lebih aman. *Relay* jenis lain ada yang namanya *reedswitch* atau *relay* lidi. *Relay* jenis ini berupa batang kontak terbuat dari

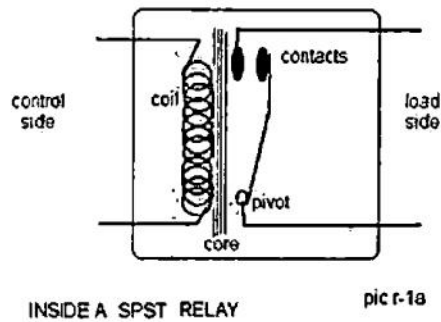
besi pada tabung kaca kecil yang dililitin kawat. Pada saat lilitan kawat dialiri arus, kontak besi tersebut akan menjadi magnet dan saling menempel sehingga menjadi saklar yang *on*. Ketika arus pada lilitan dihentikan medan magnet hilang dan kontak kembali terbuka (*off*).



Gambar 2.10. Relay

Cara kerja *Relay*:

Relay terdiri dari *Coil* dan *Contact*. *Coil* adalah gulungan kawat yang mendapat arus listrik, sedang *Contact* adalah sejenis saklar yang pergerakannya tergantung dari ada tidaknya arus listrik dicoil. *Contact* ada 2 jenis : *Normally Open* (kondisi awal sebelum diaktifkan *open*), dan *Normally Closed* (kondisi awal sebelum diaktifkan *close*). Secara sederhana berikut ini prinsip kerja dari *relay* : ketika *Coil* mendapat energi listrik (*energized*), akan timbul gaya elektromagnet yang akan menarik *armature* yang berpegas, dan *contact* akan menutup.[3]



Gambar 2.11. Cara Kerja *Relay*

2.7. *Hourmeter*

Hourmeter adalah piranti untuk mengukur seberapa lama unit tersebut bekerja atau jumlah jam operasi. Data *hourmeter* menjadi acuan terhadap analisis penggantian komponen-komponen kritis yang mengalami beban keausan.

Alat *Hourmeter* akan menghitung selama lampu *UV* dihidupkan, sehingga lamanya pemakaian lampu *UV* dapat tercatat oleh *Hour Meter* ini. Untuk menghentikan *counter Hourmeter*, cukup mematikan lampu *UV*, dikarenakan tidak adanya arus listrik maka *Hourmeter* akan berhenti menghitung waktu pemakaian. Mengingat bahwa *life time* dari lampu *UV* adalah 6000 jam.



Gambar 2.13. *Hourmeter*

2.8. *Blower*

Pengertian *blower* merupakan mesin atau alat yang digunakan untuk menaikkan atau memperbesar tekanan udara atau gas, yang akan dialirkan dalam suatu ruangan tertentu, selain itu juga dapat dimanfaatkan sebagai penghisap/pemvakum pada ruang tertentu. Di industri-industri kimia alat ini biasanya digunakan untuk mensirkulasikan gas-gas tertentu didalam tahap proses-proses secara kimiawi dikenal dengan nama *booster* atau *circulator*. [10]

Pada *blower*, cara kerja yang terjadi memiliki empat tahap, yaitu : -1. Akhir isapan, dimana udara dihisap sepenuhnya melalui lubang isap masuk kedalam ruang alur, isapan akan selesai setelah ruang alur tertutup seluruhnya oleh dinding rumah. 2. Awal kompresi, proses ini merupakan pertengahan proses kompresi, dimana *volume* udara dalam ruang alur sudah ada di tengah. 3. Akhir kompresi, dimana udara yang terkurung sudah

mencapai lubang keluar di ujung kanan atas dari rumah. 4. Pengeluaran, dimana udara yang dikurung dalam alur tadi telah dikeluarkan.[10]

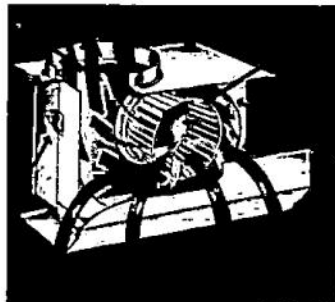
Secara umum *blower* dapat dibagi menjadi dua macam, yaitu:

a. *Positive Displacement Blower*

Pada jenis ini udara atau gas dipindahkan *volume per volume* dalam ruangan yang disebabkan adanya pergerakan elemen impeler yang berputar karena adanya pertambahan massa udara atau gas yang dipindahkan.

b. *Sentrifugal Blower*

Blower sentrifugal pada dasarnya terdiri dari satu *impeller* atau lebih yang dilengkapi dengan sudu – sudu yang dipasang pada poros yang berputar yang diselubungi oleh sebuah rumah (*casing*). Udara memasuki ruang *casing* secara horizontal akibat perputaran poros maka ruang pipa masuk menjadi vakum lalu udara dihembuskan keluar.



2.14. *Blower*