



**TEKNIK  
ELEKTRO**

**MODUL PRAKTIKUM  
SISTEM MIKROPROSESOR**

**N a m a :** .....

**N I M :** .....

**FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM SULTAN AGUNG  
Jl. Raya Kaligawe km.4 (024) 6583584 Semarang  
[www.fti.unissula.ac.id](http://www.fti.unissula.ac.id)  
Maret 2017**

## KATA PENGANTAR

Assalamu'alaikum wr wb

Alhamdulillahrabbi'l'aalamin, ucapan syukur penulis sampaikan atas kenikmatan yang tiada tara yang selalu dilimpahkan oleh-Nya sehingga penulisan modul praktikum Sistem Mikroprosesor ini selesai. Sholawat dan salam semoga selalu tercurah kepada nabi Muhammad SAW.

Praktikum adalah salah satu jenis pendidikan yang dilaksanakan di-universitas. Diantara praktikum yang ada di Teknik Elektro Unissula adalah praktikum Sistem Mikroprosesor. Telah diketahui bersama bahwa sistem mikroprosesor pada periode ini memegang peranan penting dalam kehidupan. Dimulai dari alat komunikasi, alat-alat rumah tangga, alat keamanan, dan lain sebagainya berbasis teknologi mikroprosesor. Dalam rangka memberi bekal ilmu tersebut, maka disusunlah modul Praktikum Sistem Mikroprosesor dibawah Laboratorium Elektronika dan Kendali.

Penulis mengucapkan terima kasih kepada semua pihak yang telah membantu dan membaca, serta mempraktikkan modul ini. Semoga bermanfaat.

Wassalamu'alaikum wr wb.

Maret 2017  
Penulis

Bustanul Arifin, ST, MT

**PANDUAN PELAKSANAAN PRAKTIKUM**

1. Praktikum dilaksanakan hari ..... jam ..... - ..... wib.
2. Setelah melakukan praktikum, mahasiswa praktikan mengumpulkan laporan sementara pada kertas yang telah disediakan berupa jawaban tugas dan berupa file yang disimpan pada masing-masing komputer.
3. Mahasiswa praktikan melakukan asistensi 1 maksimal 2 hari setelah praktikum (hari .....) dan asistensi 2 maksimal 4 hari setelah praktikum (hari .....).
4. Mahasiswa praktikan dimungkinkan telah melakukan asistensi lebih dari 2 kali dan setelah disetujui oleh asisten praktikum dalam 1 minggu (sebelum melakukan praktikum modul selanjutnya).
5. Jika mahasiswa praktikan tidak melakukan asistensi secara berurutan dan sesuai jadwal, maka mahasiswa tersebut tidak diijinkan untuk melakukan asistensi berikutnya untuk modul yang bersangkutan.
6. Laporan akhir dikumpulkan paling lambat satu minggu setelah praktikum keseluruhan berakhir dalam bentuk tulisan tangan.

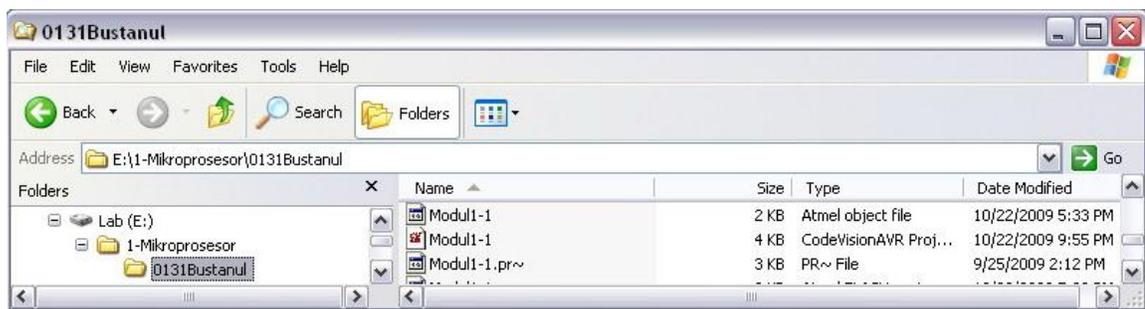
## PANDUAN PENYIMPANAN FILE

Aturan penyimpanan file hasil pada komputer adalah sebagai berikut:

1. Praktikan harus menyimpan file hasil praktikum di folder *1-Mikroprosesor* (telah tersedia dalam komputer) dengan ketentuan membuat folder dengan diberi nama empat digit terakhir NIM masing-masing mahasiswa diikuti nama depan

Contoh:

D:\\Laboratorium\\1-Mikroprosesor\\0131Bustanul\\Modul1-1



2. Penyimpanan diluar ketentuan seperti pada partisi utama yang berisi sistem operasi akan berakibat hilang jika komputer di restart karena masing-masing komputer dilengkapi dengan program *Deep Freeze*.
3. Hilangnya file hasil praktikum diluar aturan tempat penyimpanan menjadi tanggung jawab/resiko praktikan sendiri.

## TATA TERTIB

### **Tata Tertib Laboratorium:**

1. Berlaku sopan, santun dan menjunjung etika akademik dalam laboratorium.
2. Menjunjung tinggi dan menghargai staf laboratorium dan sesama pengguna laboratorium.
3. Menjaga kebersihan dan kenyamanan ruang laboratorium.
4. Peserta praktikum (praktikan) yang mengenakan pakaian/kaos oblong tidak boleh memasuki laboratorium dan/atau tidak boleh mengikuti praktikum.
5. Praktikan makan dan minum, membuat kericuhan selama kegiatan praktikum dan di dalam ruang laboratorium.
6. Dilarang memindahkan, mengganti dan menggunakan peralatan di laboratorium yang tidak sesuai dengan acara praktikum matakuliah yang diambil.
7. Praktikan yang telah menghilangkan, merusak atau memecahkan peralatan praktikum harus mengganti sesuai dengan spesifikasi alat yang dimaksud, dengan kesepakatan antara laboran, pembimbing praktikum dan kepala laboratorium. Persentase pengantian alat yang hilang, rusak atau pecah disesuaikan dengan jenis alat atau tingkat kerusakan dari alat.
8. Apabila praktikan tidak sanggup mengganti alat yang hilang, rusak atau pecah dikarenakan harga alat mahal atau alat tidak ada dipasaran, maka nilai penggantian ditetapkan atas kesepakatan antara ketua jurusan, pembimbing praktikum dan peserta praktikum.
9. Membaca, memahami dan mengikuti prosedur operasional untuk setiap peralatan dan kegiatan selama praktikum dan di ruang laboratorium.

### **Hal yang perlu diperhatikan oleh praktikan sebelum meninggalkan ruangan laboratorium:**

1. Pastikan komputer telah di-*shutdown* (dimatikan).
2. Matikan modul hardware praktikum.
3. Cabutlah kabel power komputer dari stop kontak.
4. Cabutlah kabel power modul hardware dari stop kontak.
5. Cabutlah semua kabel aplikasi pada masing-masing port mikrokontroler.
6. Rapiakan kabel modul hardware dengan cara menggulung dan mengikatnya.
7. Letakkan modul hardware dan kabel-kabel aplikasi pada meja dengan rapi.
8. Rapiakan laci keyboard, mouse, dan pad mouse.
9. Rapiakan kursi dengan cara mendekatkannya dengan meja komputer.
10. Jangan meninggalkan sampah apapun di dalam ruang praktikum (kertas, sobekan kertas, plastik, bungkus permen, dll).
11. Periksa barang-barang anda agar tidak ada yang ketinggalan (flashdisk, buku, pulpen, handphone).
12. Kehilangan barang-barang setelah meninggalkan tempat menjadi tanggung jawab praktikan sendiri.

**DAFTAR ISI**

1. Cover	i
2. Kata Pengantar	ii
3. Panduan Pelaksanaan Praktikum	iii
4. Panduan Penyimpanan File	iv
5. Tata Tertib	v
6. Daftar isi	vi
7. (Modul-1A) Aplikasi CV-AVR	1
8. (Modul-1B) Aplikasi Port	7
9. (Modul-2A) Aplikasi Led	9
10. (Modul-2B) Aplikasi For	12
11. (Modul-3A) Aplikasi Array, Prosedur, dan Fungsi	15
12. (Modul-3B) Aplikasi LCD	18
13. (Modul-4A) Aplikasi Push Button	20
14. (Modul-4B) Aplikasi Seven Segmnet	22
15. (Modul-5) Aplikasi Keypad	25
16. (Modul-6A) Aplikasi Sensor Ultrasonik	29
17. (Modul-6B) Aplikasi Motor DC	32
18. Daftar Pustaka	35

# APLIKASI CV-AVR

Modul-1A

## A. Tujuan

1. Mahasiswa dapat menjalankan program CodeVisionAVR
2. Mahasiswa dapat memahami cara pemrograman mikrokontroler

## B. Dasar Teori

### CV-AVR (CodeVision AVR)

CodeVision Alf and Vegard's Risc processor atau sering disingkat CVAVR adalah program yang berbasis pada bahasa C. Program ini merupakan salah satu alat bantu pemrograman (programming tool) yang bekerja dalam lingkungan pengembangan perangkat lunak yang terintegrasi (Integrated Development Environment, IDE). CV-AVR dilengkapi dengan source code editor, compiler, linker, dan dapat memanggil Atmel AVR Studio untuk debuggernya.

CodeVision AVR terdiri atas Generator Program Otomatis CodeWizardAVR yang memungkinkan pengguna untuk menulis dalam beberapa saat semua kode yang diperlukan untuk beberapa implementasi seperti dalam daftar berikut:

- Setup akses memori eksternal
- Identifikasi sumber reset chip
- Inisialisasi port untuk input maupun output
- Inisialisasi interupsi eksternal
- Inisialisasi timer dan counter
- Inisialisasi timer watchdog
- Inisialisasi UART dan interupsi yang dikendalikan oleh buffer komunikasi serial
- Inisialisasi komparator analog
- Inisialisasi ADC
- Inisialisasi interface SPI
- Bus I2C, sensor temperature LM75, thermometer/termostatDS1621 dan PCF8563, PCF8583, DS1302, inisialisasi RTC DS1307
- Inisialisasi sensor temperatur DS1820/DS18S20, 1 wire bus
- Inisialisasi modul LCD

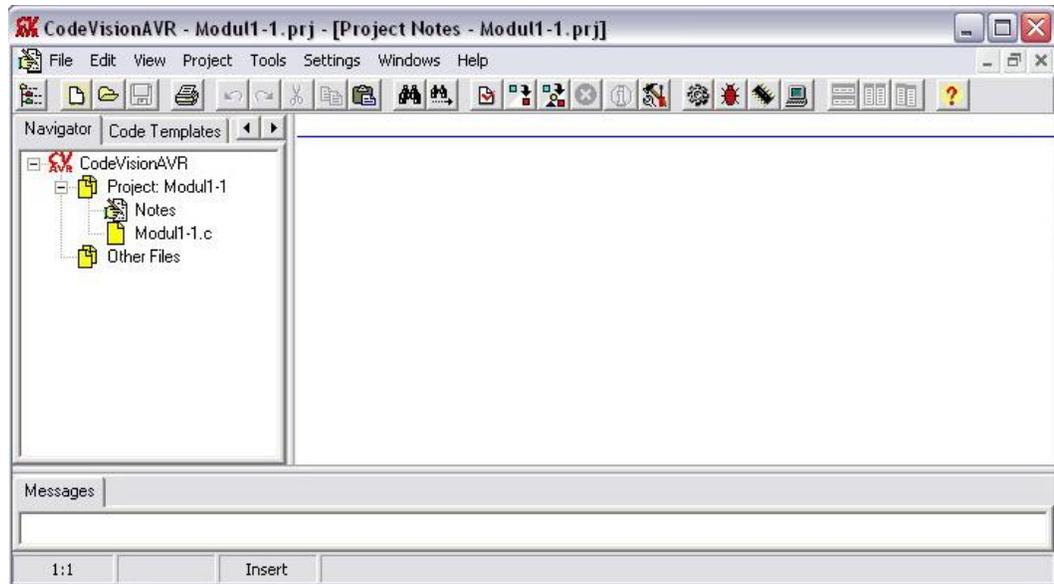
## C. Percobaan

### a) Alat yang diperlukan

1 unit komputer yang dilengkapi dengan software CVAVR

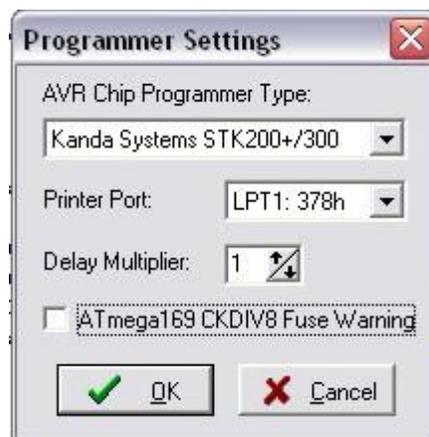
### b) Langkah Percobaan

- 1) Buka program CodeVisionAVR, sehingga tampak seperti pada Gambar 1.



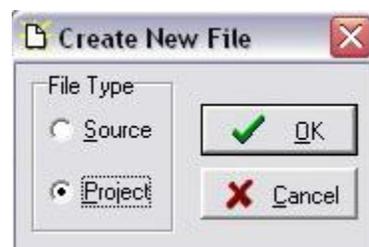
Gambar 1. Tampilan awal program CVAVR

- 2) Buka menu Setting, pilih Programmer. Lakukan perubahan sehingga menjadi tampak pada Gambar 2.



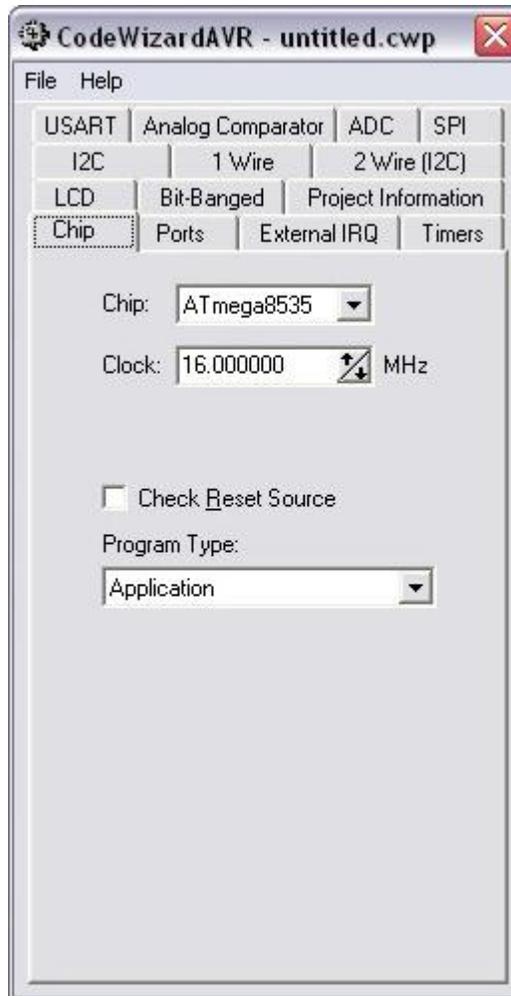
Gambar 2. Tampilan Programmer Setting

- 3) Buka menu File, kemudian pilih New sehingga akan muncul tampilan seperti pada Gambar 3.



Gambar 3. Tampilan pembuatan file baru

- 4) Pilih Project dan kemudian tekan tombol OK.
- 5) Akan muncul dialog untuk konfirmasi menggunakan program CodeWizardAVR. Pilih tombol Yes dan akan tampak seperti pada Gambar 4.

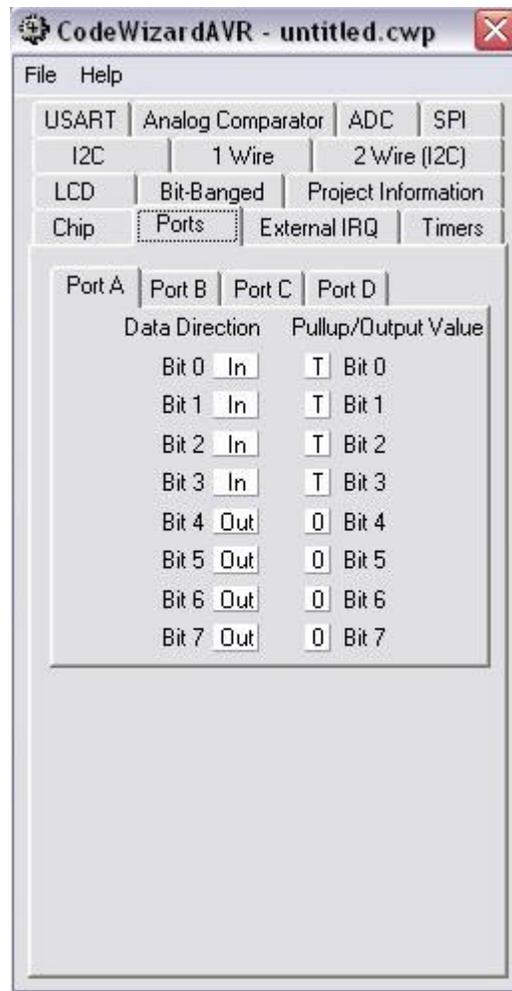


Gambar 4. Tampilan Project Wizard

- 6) Pilih ATmega8535 sesuai dengan hardware yang akan digunakan pada pilihan chip, dan sesuaikan nilai clocknya.
- 7) Setelah semuanya disesuaikan buka tab Ports untuk menentukan penggunaan masing-masing Port seperti ditunjukkan pada Gambar 5. Untuk pengaturan penggunaan port disesuaikan dengan Tabel 1.

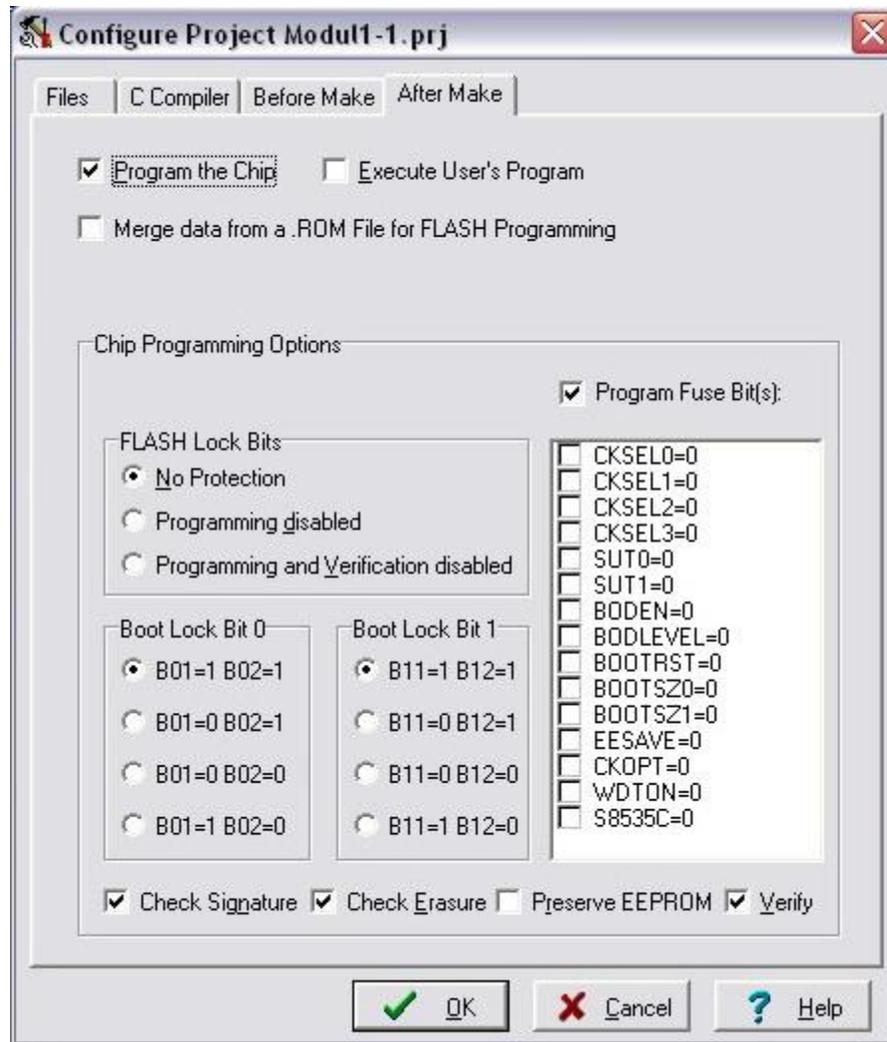
Tabel 1. Pengaturan Port I/O

	DDR bit = 1	DDR bit = 0
PORT bit =1	Output; High	Input; R pull-up
PORT bit=0	Output; Low	Input; Floating



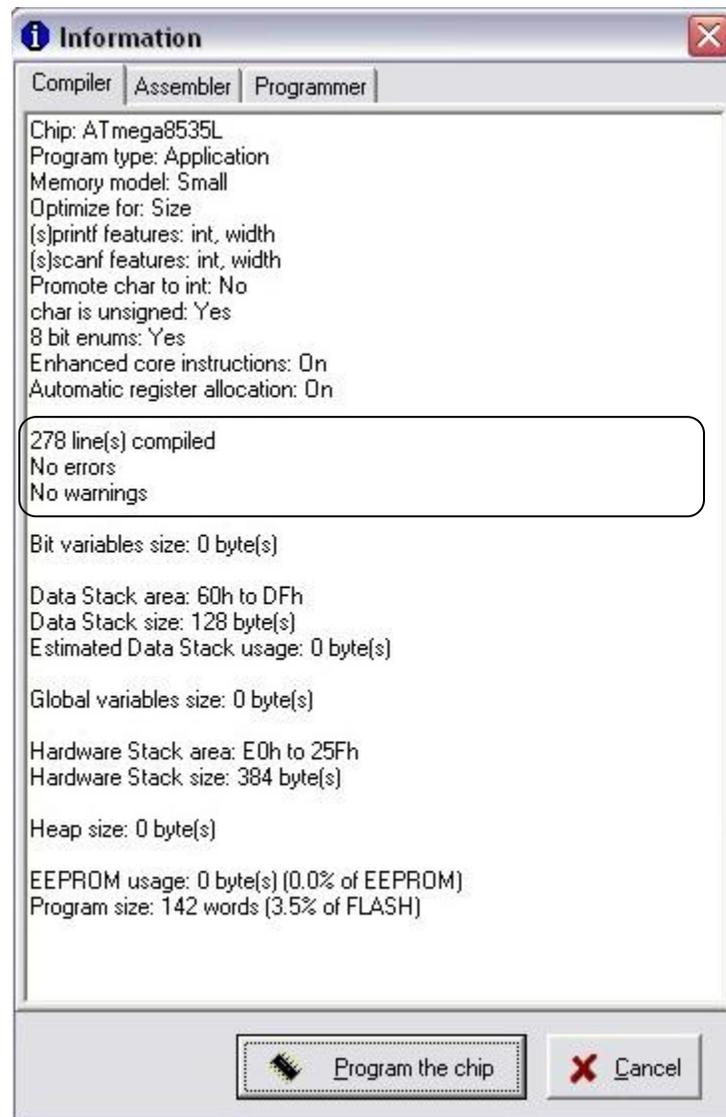
Gambar 5. Tampilan pengaturan input dan output

- 8) Buka menu file, pilih Generate, save, and exit. Buatlah folder dengan nama kelompok anda pada partisi D. Simpan masing-masing dengan nama Praktek-1.C (untuk file source C), Praktek-1.prj (untuk file project) dan Praktek-1.cwp (untuk Code Wizard Project).
- 9) Buka menu Project, pilih Configure, dan pilih After Make seperti tampak pada Gambar 6.



Gambar 6. Tampilan Configure Project

- 10) Untuk meng*compile* dengan memilih menu Project, dilanjutkan Make. Jika tidak terjadi kesalahan maka akan terlihat pada Gambar 7.



Gambar 7. Tampilan hasil kompilasi

11) Untuk mendownload program ke mikrokontroler, pilih menu *Program the cip*.

**c) Hasil Percobaan**

- 1) Amati yang terjadi pada modul hardware
- 2) Lihatlah file yang dihasilkan setelah di-compile dan di-make

**d) Tugas dan Pertanyaan**

- 1) Apakah yang dimaksud dengan meng-compile program?
- 2) Apakah yang dimaksud dengan DDR?
- 3) Apakah fungsi file yang berektensi Hex?
- 4) Bagaimanakah agar suatu sintak dijadikan menjadi komentar atau keterangan?
- 5) Apakah perbedaan antara source dengan project ketika membuat file baru?
- 6) Bagaimanakah struktur bahasa C dalam program CVAVR ini?

# APLIKASI PORT

Modul-1B

## A. Tujuan

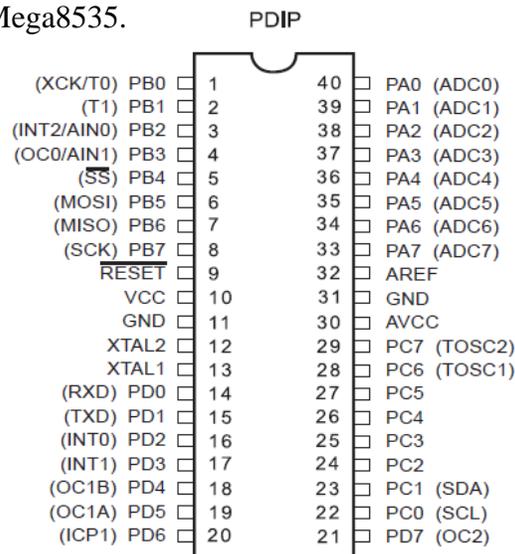
1. Mahasiswa dapat menjalankan program CodeVisionAVR
2. Mahasiswa dapat memahami cara pemrograman port-port pada mikrokontroler

## B. Dasar Teori

### Mikrokontroler ATmega8535

Mikrokontroler ATmega8535 adalah mikrokontroler jenis RISC (Reduced Instructions of Set Computing) yang berasal dari keluarga AVR (Alf and Vegard's Risc Processor). Mikrokontroler ini mempunyai arsitektur computer 8-bit dan semua instruksi dalam kode 16-bit (1 word). Keunggulan mikrokontroler ini dibandingkan dengan keluarga MCS-51 adalah pada kecepatannya. Hal ini dikarenakan intruksi dieksekusi dalam satu siklus clock.

ATmega8535 ini mempunyai 4 port yang disebut dengan Port-A, Port-B, Port-C, dan Port-D. Masing-masing port dapat digunakan sebagai saluran/kanal input dan output berjumlah 32 (masing-masing port 8-pin). Jumlah pin/kaki yang tersedia pada chip ini adalah 40 dengan catudaya +5 volt. Fasilitas yang ada pada mikrokontroler ini adalah ADC (Analog to Digital Converter) 10-bit sebanyak 8-kanal, 3 buah Timer/Counter, memori SRAM 512-byte, memori flash 8-kbyte, dan kecepatan maksimum 16-MHz. Memori data terbagi menjadi 3 bagian: (32 register umum [\$00-\$1F], 64 register I/O [\$20-5F], dan 512 SRAM Internal [\$060-025F]). Memori program terletak pada Flash PEROM tersusun dalam word (2-byte) karena setiap instruksi memiliki lebar 16-bit atau 32-bit mulai \$000 sampai \$FFF. Memori data berupa EEPROM 8-bit sebanyak 512-byte dengan alamat \$000-\$1FF. Gambar 8 menunjukkan konfigurasi pin-pin pada chip mikrokontroler ATmega8535.



Gambar 8. Tata letak pin-pin mikrokontroler ATmega8535

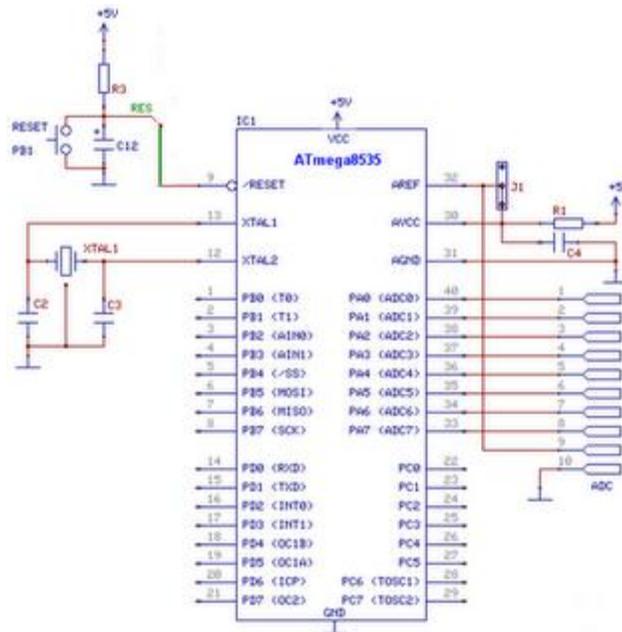
### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535

#### b) Gambar Rangkaian

Skema rangkaian sistem minimum AVR ATmega8535 ditunjukkan pada Gambar 9.



Gambar 9. Sistem minimum AVR ATmega8535

#### c) Langkah Percobaan

- 1) Buka program CodeVisionAVR.
- 2) Buatlah *project* baru dengan ketentuan DDRA sebagai output.
- 3) Beri nama *project* ini Praktek-2.
- 4) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.
- 5) Jika *tidak* terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara memilih *Program the chip*.

#### d) Hasil Percobaan

- 1) Amati yang terjadi pada modul hardware
- 2) Buatlah agar PORTA dan PORTB menjadi output, amati yang terjadi pada DDRA dan DDRB.

#### e) Tugas dan Pertanyaan

- 1) Berapakah PORT yang dimiliki oleh mikrokontroler ATmega8535?
- 2) Berapa pin-kah masing-masing PORT?
- 3) Bagaimanakah cara mensetting PORT menjadi input maupun output jika sudah terlanjur masuk ke dalam *project*?

## APLIKASI LED

Modul-2A

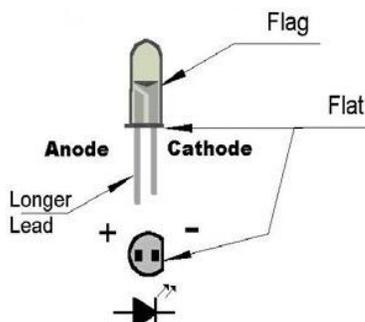
### A. Tujuan

1. Mahasiswa dapat menjalankan program CodeVisionAVR
2. Mahasiswa dapat memahami cara pemrograman aplikasi LED pada mikrokontroler

### B. Dasar Teori

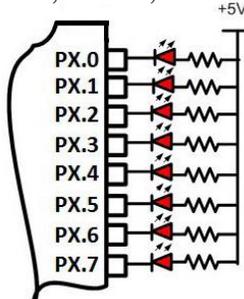
#### Light Emitting Diode (LED)

Led adalah salah satu jenis diode yang dapat menghasilkan cahaya. Led dapat menghasilkan cahaya berbagai macam warna tergantung dari jenisnya. Warna yang umum dihasilkan adalah merah, kuning, hijau, biru atau ungu, dan putih. Sama halnya dengan diode led mempunyai 2 kaki yaitu kaki anoda yang dihubungkan dengan tegangan positif dan kaki katoda yang dihubungkan dengan tegangan negatif. Untuk lebih mudah menghafalkannya sering digunakan singkatan KNAP (katoda negatif anoda positif). Adapun kaki dan simbol led dapat dilihat pada Gambar 10. Arus maksimum yang dapat diterima oleh Led adalah sekitar 20 mA (miliampere).



Gambar 10. Simbol dan kaki led

Pada aplikasi mikrokontroler yang digunakan kaki anoda dihubungkan dengan tegangan Vcc, dan kaki katoda dihubungkan dengan port mikrokontroler. Skema rangkaian aplikasi led diperlihatkan pada Gambar 10. Port X merupakan simbol Port yang bersangkutan, seperti Port A, Port B, Port C, ataupun Port D.



Gambar 11. Skema rangkaian led pada sistem mikrokontroler

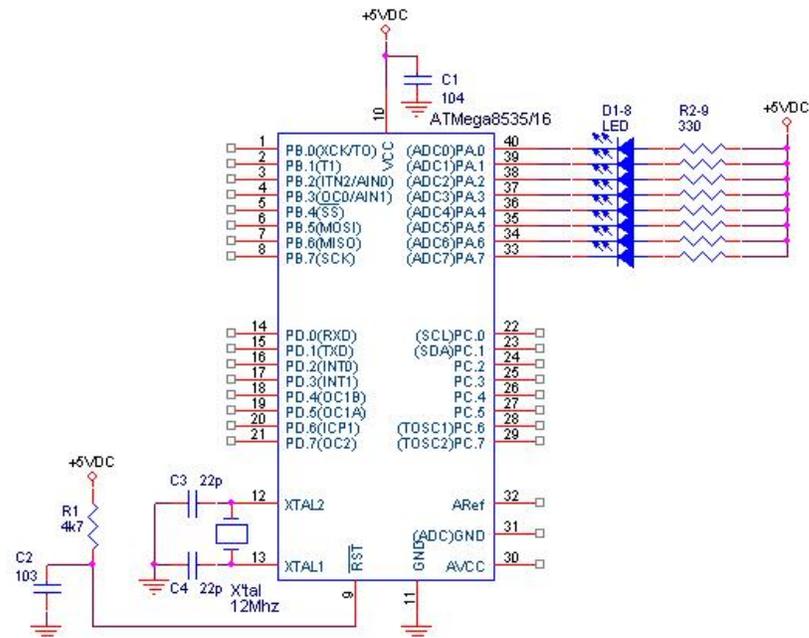
### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535 dengan aplikasi Led

#### b) Gambar Rangkaian

Skema rangkaian sistem minimum AVR ATmega8535 yang dihubungkan dengan aplikasi Led pada PORTA ditunjukkan pada Gambar 12.



Gambar 12. ATmega8535 dihubungkan dengan Led

#### c) Langkah Percobaan

- a) Hubungkan modul aplikasi Led pada Port-A mikrokontroler dengan kabel data.
- b) Bukalah program CodeVisionAVR
- c) Buatlah *project* baru dengan ketentuan DDRA sebagai output.
- d) Beri nama *project* ini Praktek-3.
- e) Ketiklah header delay sebagai berikut:

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

- d) Ketiklah program utama sebagai berikut.

```
while (1)
{
// Place your code here
PORTA=0b00001111;
delay_ms(500);
PORTA=0b11110000;
delay_ms(500);
};
```

- e) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.

- f) Jika *tidak* terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara memilih *Program the chip*.

**d) Hasil Percobaan**

- 1) Amati yang terjadi pada modul hardware.
- 2) Apa yang terjadi pada Led jika delay dihilangkan?
- 3) Gantilah delay dengan waktu 3000ms. Apa yang terjadi dengan Led?

**e) Tugas dan Pertanyaan**

- 1) Apakah perbedaan common anode dan common katode?
- 2) Sebutkan keunggulan common anode dibandingkan dengan common katode dalam aplikasi led!
- 3) Apakah perbedaan satuan ms dan us dalam aplikasi ini?

# APLIKASI FOR

Modul-2B

## A. Tujuan

1. Mahasiswa dapat menjalankan program CodeVisionAVR
2. Mahasiswa dapat memahami cara pemrograman aplikasi LED dengan menggunakan statement *for* pada mikrokontroler

## B. Dasar Teori

### Statement *while* dan statement *for*

Komputer yang telah diprogram mempunyai keunggulan jika dibandingkan dengan manusia dalam melaksanakan perintah pengulangan. Didalam algoritma pengulangan (repetition atau kadang sering disebut dengan loop) dapat dilakukan dalam jumlah tertentu sampai dengan kondisi tertentu telah dicapai. Dalam pemrograman yang berbasis bahasa C terdapat statement yang berfungsi untuk pengulangan yaitu statement *while* dan *for*. Statement *while* dan Statement *for* digunakan untuk melakukan pengulangan sampai dengan persyaratan yang telah ditentukan pada awal program tercapai. Bentuk baku *while* dalam pemrograman CVAVR adalah:

```
while(ekspresi)
{
    Pernyataan_1;
    Pernyataan_2;
}
```

Dalam bentuk baku *while* ini, ekspresi digunakan sebagai suatu persyaratan yang harus dipenuhi agar dapat mengeksekusi Pernyataan\_1 dan Pernyataan\_2. Sedangkan pengulangan dengan statement *for* dalam pemrograman CVAVR adalah:

```
for (ekspresi1; ekspresi2; ekspresi3)
{
    Pernyataan_1;
    Pernyataan_2;
    Pernyataan_3;
}
```

Ekspresi1 merupakan inisialisasi, ekspresi2 adalah persyaratan yang harus dipenuhi, dan ekspresi3 yaitu pernyataan yang dilakukan jika persyaratan ekspresi2 terpenuhi. Pernyataan\_1, Pernyataan\_2, dan Pernyataan\_3 dalam kurung merupakan perintah yang akan dilaksanakan/dieksekusi jika semua persyaratan ekspresi2 terpenuhi.

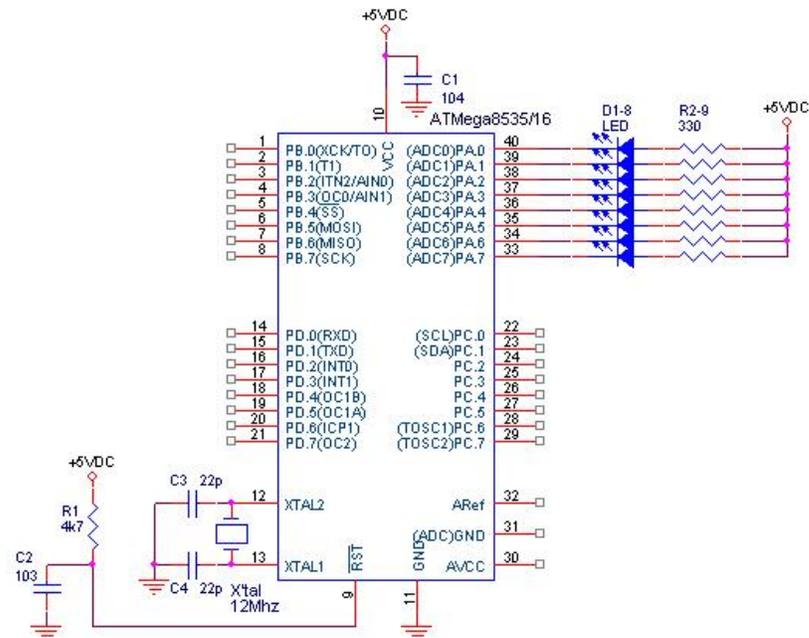
### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535 dengan aplikasi Led

#### b) Gambar Rangkaian

Skema rangkaian sistem minimum AVR ATmega8535 yang dihubungkan dengan aplikasi Led pada PORTA ditunjukkan pada Gambar 13.



Gambar 13. ATmega8535 untuk aplikasi statement for

#### c) Langkah Percobaan

- 1) Hubungkan modul aplikasi Led pada Port-A mikrokontroler dengan kabel data.
- 2) Bukalah program CodeVisionAVR
- 3) Buatlah *project* baru dengan ketentuan DDRA sebagai output.
- 4) Beri nama *project* ini Praktek-4.
- 5) Ketiklah header delay sebagai berikut:

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

- 6) Ketiklah deklarasi program sebagai berikut:

```
// Declare your global variables here
int i;
```

- 7) Ketiklah program utama sebagai berikut.

```
while (1)
{
// Place your code here
for (i=1; i<10; i++)
{
PORTB=0x0F;
delay_ms(500);
PORTB=0xF0;
```

```

        delay_ms(500);
    }
    delay_ms(10000);
}

```

- 8) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.
- 9) Jika *tidak* terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara memilih *Program the chip*.

#### d) Hasil Percobaan

- 1) Amati yang terjadi pada modul hardware.
- 2) Apa yang terjadi pada Led jika delay dihilangkan?
- 3) Gantilah delay dengan waktu 3000ms. Apa yang terjadi dengan Led?
- 4) Apa yang terjadi pada Led jika statement for (i=1; i<10; i++) diganti for (i=5; i<10; i++)
- 5) Buatlah agar LED menyala bergantian sebanyak tujuh kali.
- 6) Buatlah program untuk menyalakan LED secara bergantian sebanyak sepuluh kali tetapi dengan menggunakan sintak i--.

#### e) Tugas dan Pertanyaan

- 1) Jelaskan fungsi statement for dalam program tersebut!
- 2) Apa yang dimaksud dengan sintak i++ ?
- 3) Apakah yang dimaksud dengan PORTB=0xF0?
- 4) Apakah perbedaan penulisan 0b00001111 dengan penulisan 0x0F?

## APLIKASI ARRAY, PROSEDUR, dan FUNGSI

Modul-3A

### A. Tujuan

1. Mahasiswa dapat menjalankan program CodeVisionAVR
2. Mahasiswa dapat memahami pemrograman menggunakan array, *procedure*, dan *fungsi* pada pemrograman mikrokontroler

### B. Dasar Teori

#### Array (Larik)

Array ialah sebuah variabel yang menyimpan sekumpulan data yang memiliki tipe data sama. Dengan kata lain array dapat didefinisikan sebagai tipe data terstruktur yang terdapat dalam memori yang merupakan gabungan dari beberapa variabel sejenis serta memiliki jumlah komponen yang tetap. Jadi setiap data menempati lokasi atau alamat memori yang berbeda-beda yang sering disebut dengan register. Setiap elemen array dapat diakses melalui indeks di dalamnya. Yang perlu diperhatikan bahwa sesuai dengan kaidah ilmu digital bahwa indeks array selalu dimulai dari hitungan ke-0 (bukan hitungan ke-1). Gambar 14 menunjukkan pemetaan memori untuk array.

Reserved Space		Storing Values after Initialization	
	Number[0]	35	Number[0]
	Number[1]	20	Number[1]
	Number[2]	40	Number[2]
	Number[3]	57	Number[3]
	Number[4]	19	Number[4]

Gambar14. Register penyimpan data array

#### Prosedur dan Fungsi

Dalam pemrograman di CVAVR, program yang baik terdiri atas fungsi-fungsi. Khususnya dalam implementasi teknik pemrograman yang terstruktur, penulisan kode program dalam blok yang berulang-ulang sebaiknya dihindari. Hal lain yang sebaiknya tidak digunakan adalah penulisan pengarah proses (dikenal dengan *statement goto*). Istilah fungsi (*function*) dikenal dalam bahasa pemrograman umum, sedangkan dalam pemrograman Pascal dikenal istilah prosedur.

Prosedur sebenarnya merupakan salah satu bentuk khusus dari suatu fungsi. Prosedur didefinisikan sebagai suatu proses yang tidak mengembalikan nilai. Pada pemrograman dibedakan antara main program dan sub program. Prosedur dan fungsi termasuk dalam sub program. Sub program dapat dibagi menjadi sub program yang bebas, sub program yang memanggil sub program yang lain, dan sub program yang memanggil dirinya sendiri.

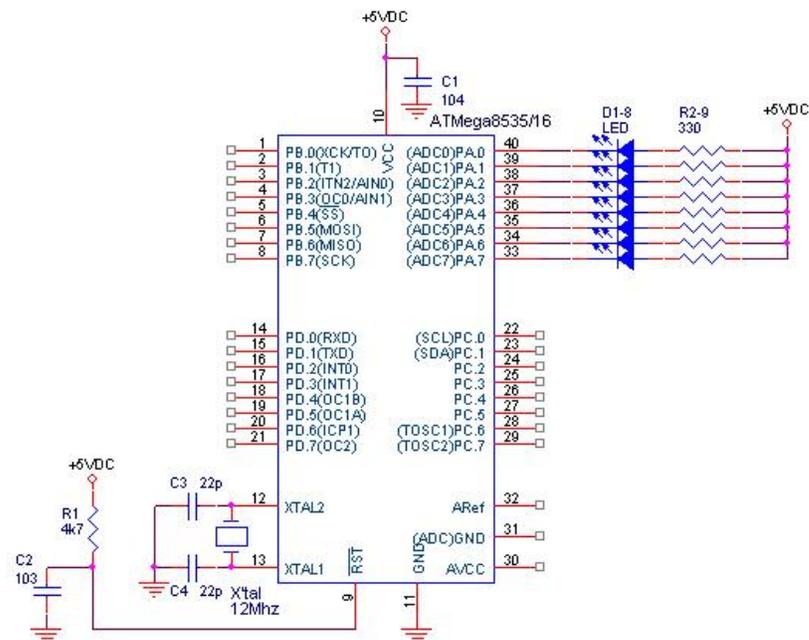
### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535 dengan aplikasi Led

#### b) Gambar Rangkaian

Skema rangkaian sistem minimum AVR ATmega8535 yang dihubungkan dengan aplikasi Led pada PORTA ditunjukkan pada Gambar 15.



Gambar 15. ATmega8535 untuk aplikasi statement for

#### c) Langkah Percobaan

- 1) Bukalah program CodeVisionAVR
- 2) Buatlah project baru dengan ketentuan DDRA sebagai Output.
- 3) Beri nama project ini Praktek-5
- 4) Ketiklah header delay sebagai berikut:

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

- 5) Ketiklah deklarasi program sebagai berikut:

```
// Declare your global variables here
char dataLED1[8]={0xFE,0xFD,0xFB,0xF7,0xEF,0xDF,0xBF,0x7F};
char dataLED2[8]={0x7F,0xBF,0xDF,0xEF,0xF7,0xFB,0xFD,0xFE};
char i;
char j;
```

- 6) Ketiklah *procedure* sebagai berikut:

```
void lampu1()
{
```

```

for(j=0; j<8; j++)
{
    for(i=0; i<8; i++)
    {
        PORTA=dataLED1[i];
        delay_ms(500);
    };
};

void lampu2()
{
    for(j=0; j<8; j++)
    {
        for(i=0; i<8; i++)
        {
            PORTA=dataLED2[i];
            delay_ms(500);
        };
    };
};

```

7) Program utama diketik sebagai berikut.

```

while (1)
{
    // Place your code here
    lampu1();
    PORTA=0xF0;
    delay_ms(1000);
    lampu2();
    PORTA=0x0F;
    delay_ms(1000);
};

```

- 8) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.
- 9) Jika tidak terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara menekan *Program the chip*.

**d) Hasil Percobaan**

- 1) Apa yang terjadi pada Led ?
- 2) Jelaskan maksud program Praktek-5

**e) Tugas dan Pertanyaan**

- 1) Apa yang dimaksud dengan *procedure* dan apa manfaatnya?
- 2) Apa yang dimaksud dengan *fungsi* dan apa manfaatnya?

## APLIKASI LCD

Modul-3B

### A. Tujuan

1. Mahasiswa dapat menjalankan program CodeVisionAVR
2. Mahasiswa dapat memahami cara pemrograman LCD pada mikrokontroler

### B. Dasar Teori

#### LCD (Liquid Crystal Display)

LCD merupakan perangkat output yang sering digunakan dalam dunia elektronik. Teknologi terkini untuk keperluan output sebagian besar sudah menggunakan LCD. Baik untuk keperluan televisi, layar ponsel, osiloskop, serta keperluan lain. Bentuk LCD ada berbagai macam baik mulai dari layar lebar, layar sedang, maupun layar kecil. Untuk aplikasi dipraktikum ini digunakan LCD 16x2. Maksudnya adalah LCD ini mempunyai 16 kolom dan 2 baris. LCD ini dapat digunakan dengan tegangan + 5 volt. Gambar 16 menunjukkan LCD 16x2.



Gambar 16. LCD jenis 16x2

### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535
- 3) 1 modul aplikasi LCD

#### b) Langkah Percobaan

- 1) Hubungkan modul LCD dengan Port-C mikrokontroler
- 2) Bukalah program CodeVisionAVR
- 3) Buatlah project baru dengan ketentuan LCD ke Port-C
- 4) Beri nama project ini Praktek-6
- 5) Ketiklah insialisasi LCD dengan program berikut:

```
lcd_gotoxy(0,0);
lcd_putsf("Praktikum");
lcd_gotoxy(3,1);
lcd_putsf("Mikroprosesor");
```

- 6) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.
- 7) Jika tidak terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara memilih *Program the chip*.

**c) Hasil Percobaan**

- 1) Apa yang terjadi pada LCD?
- 2) Buatlah tulisan pada baris pertama LCD “Fakultas Teknologi Industri” pada LCD baris kedua tulisan “Unissula Semarang”.
- 3) Amati apa yang terjadi pada LCD dan jelaskan kenapa terjadi hal demikian !

**d) Tugas dan Pertanyaan**

- 1) Berdasarkan setting pada CV-AVR sebutkan hubungan pin mikrokontroler dengan pin LCD.
- 2) Jelaskan maksud program Praktek-6

## APLIKASI PUSH BUTTON

Modul-4A

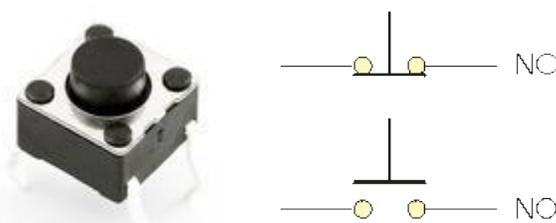
### A. Tujuan

1. Mahasiswa dapat menjalankan program CodeVisionAVR
2. Mahasiswa dapat memahami cara pemrograman Push Button pada mikrokontroler

### B. Dasar Teori

#### Tombol Push Button

Tombol push button merupakan rangkaian saklar yang berfungsi untuk menyambung dan memutus rangkaian. Yang membedakan tombol ini dengan tombol saklar biasa (toggle) adalah fungsinya yang tidak bersifat tetap. Hal ini menyebabkan keadaan akan menjadi kembali ke keadaan semula jika tombol sudah tidak ditekan. Ada dua jenis tombol ini, yaitu yang secara normal (tidak diapa-apakan) adalah dalam keadaan terbuka/tidak tersambung (normally open) dan yang secara normal kondisi saklar dalam keadaan tersambung/tertutup (normally close). Gambar 17 menunjukkan bentuk fisik dan simbol saklar NO serta NC.



Gambar 17. Bentuk fisik dan Simbol tombol push button NO-NC

### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535
- 3) 1 modul aplikasi tombol push button

#### b) Langkah Percobaan

- 1) Hubungkan modul LCD dengan Port-C mikrokontroler, Push Button dengan Port B
- 2) Bukalah program CodeVisionAVR
- 3) Buatlah project baru dengan ketentuan LCD ke Port-C, DDR-B sebagai Input, dan  $PORTB=0xFF$ .
- 4) Beri nama project ini Praktek-7
- 5) Ketiklah header delay sebagai berikut:

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

6) Program utama diketik sebagai berikut.

```

while (1)
{
  if(PINB.0==0)
  {lcd_gotoxy(0,1);
  lcd_putsf("Mahasiswa");
  delay_ms(10);
  }
  else if (PINB.1==0)
  {lcd_gotoxy(10,1);
  lcd_putsf("FTI");
  delay_ms(10);
  }
  else
  {lcd_gotoxy(0,1);
  lcd_putsf("      "); //beri 5 spasi
  }
};

```

- 7) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.
- 8) Jika tidak terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara memilih *Program the chip*.

**c) Hasil Percobaan**

- 1) Amatilah jika push button tidak ditekan apa yang akan muncul pada LCD?
- 2) Amatilah jika push button-0 (push button paling kanan) ditekan apa yang terjadi pada LCD?
- 3) Amati apa yang terjadi pada LCD jika push button-1 ditekan?
- 4) Jelaskan kenapa terjadi hal demikian !

**d) Tugas dan Pertanyaan**

- 1) Modifikasilah program diatas agar jika push button-0 ditekan pada koordinat baris kedua kolom pertama menghasilkan tulisan "tombol 0" pada LCD dst (sesuai dengan push button yang ditekan), dan jika push button ditekan hanya muncul tulisan "Silahkan tekan tombol" pada koordinat baris pertama dan kolom pertama
- 2) Jelaskan program yang anda buat !

# APLIKASI SEVEN SEGMENT

Modul-4B

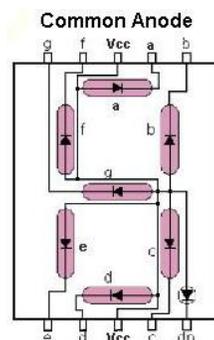
## A. Tujuan

1. Mahasiswa dapat menjalankan program CodeVisionAVR
2. Mahasiswa dapat memahami cara pemrograman seven segment pada mikrokontroler

## B. Dasar Teori

### Seven Segment (7-segment)

Seven-segment adalah gabungan lampu Led yang membentuk suatu angka desimal. Sesuai dengan namanya maka alat ini terdiri atas 7 ruas yang menjadi satu kesatuan. Akan tetapi seringkali seven-segment sendiri terdiri atas 8 lampu Led karena dilengkapi dengan tanda titik (dp=dot-point). Ada 2 jenis seven-segment yang umum yaitu common anode dan common catode. Pada praktikum ini digunakan common anode karena seven segment ini relatif lebih mudah didapatkan di pasaran. Gambar 18 menunjukkan skema rangkaian seven-segment yang dihubungkan dengan port mikrokontroler. Untuk bentuk-bentuk angka yang dihasilkan oleh seven-segment ditunjukkan pada Gambar 19. Angka-angka yang dihasilkan adalah berdasarkan pada kombinasi logika pada tiap-tiap kaki seven-segment. Rangkuman angka biner dan heksadesimal ditunjukkan pada Tabel 2.



Gambar 18. Skema rangkaian seven-segment common anode



Gambar 19. Angka-angka yang dihasilkan oleh seven-segment

Tabel 2. Kombinasi logika untuk tampilan seven-segment

Tmpln	dp	g	f	e	d	c	b	a	Heksa
0	1	1	0	0	0	0	0	0	C0
1	1	1	1	1	1	0	0	1	F9
2	1	0	1	0	0	1	0	0	A4
3	1	0	1	1	0	0	0	0	B0
4	1	0	0	1	1	0	0	1	99
5	1	0	0	1	0	0	1	0	92
6	1	0	0	0	0	0	1	0	82
7	1	1	1	1	1	0	0	0	F8
8	1	0	0	0	0	0	0	0	80
9	1	0	0	1	0	0	0	0	90

### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535
- 3) 1 modul aplikasi seven segment

#### b) Langkah Percobaan

- 1) Hubungkan kabel alamat aplikasi seven segment pada Port-B, kabel kendali seven segment pada Port-A mikrokontroler.
- 2) Bukalah program CodeVisionAVR
- 3) Buatlah project baru dengan ketentuan DDR-A=0xFF, PORTA=0xFF dan DDR-B=0xFF, PortB=0xFF.
- 4) Beri nama project ini Praktek-8
- 5) Ketiklah header delay sebagai berikut:

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

- 6) Ketiklah deklarasi program sebagai berikut:

```
unsigned char bil[10] = {0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};
int kiri, kanan, i;
```

- 7) Ketiklah program utama sebagai berikut:

```
kiri=0;
kanan=0;

for(kiri=0;kiri<2;kiri++)
{
    for(kanan=0;kanan<10;kanan++)
    {
        for(i=0;i<100;i++)
        {
```

```

        PORTA.0=1;
        PORTA.1=0;
        PORTB=bil[kiri];
        delay_ms(5);
        PORTA.0=0;
        PORTA.1=1;
        PORTB=bil[kanan];
        delay_ms(5);
    }
}
};

```

- 8) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.
- 9) Jika tidak terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara memilih *Program the chip*.

**c) Hasil Percobaan**

- 1) Amatilah hasil tampilan seven segment, apa yang terjadi?

**d) Tugas dan Pertanyaan**

- 1) Modifikasi program agar menghitung naik sampai dengan angka 49
- 2) Modifikasi program agar menghitung turun dari angka 99 sampai dengan 0
- 3) Modifikasi program agar tampilan hanya 1 display saja dan menghitung dari 0 ke 9 secara berulang-ulang.
- 4) Jelaskan maksud program Praktek-8

## APLIKASI KEYPAD

Modul-5

### A. Tujuan

- 1) Mahasiswa dapat menjalankan program CodeVisionAVR
- 2) Mahasiswa dapat memahami cara pemrograman keypad pada mikrokontroler

### B. Dasar Teori

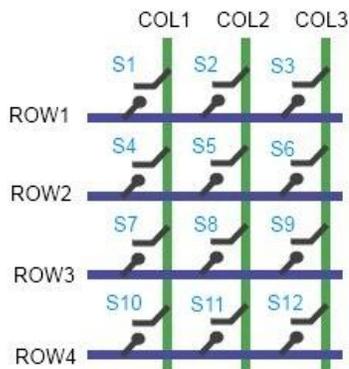
#### Keypad

Keypad merupakan salah satu jenis tombol switch/saklar. Untuk penggunaan tombol mandiri (satu tombol terhubung langsung ke port mikrokontroler yang berguna untuk satu aplikasi) mempunyai keterbatasan jumlah tombol yang dapat dipakai. Untuk mengatasi keterbatasan masalah pada pin mikrokontroler yang dihubungkan dengan tombol maka digunakanlah keypad.

Sistem kerja keypad adalah berdasarkan matriks. Ada beberapa jenis keypad, diantaranya adalah keypad matriks 3x4 dan 4x4. Dalam praktikum ini digunakan jenis keypad 3x4. Dengan teknik matriks dapat dilakukan penghematan. Untuk matriks 3x4 dapat digunakan aplikasi sebanyak 12 buah. Gambar 20 menunjukkan gambar matriks 3x4 dan skema pensaklarannya diperlihatkan pada Gambar 21.



Gambar 20. Jenis keypad matriks 3x4



Gambar 21. Skema pensaklaran keypad 3x4

### Statement Switch-case

Pemilihan proses biasanya menggunakan statement if. Statement ini didasarkan pada dua pilihan yang bisa terjadi. Jika pemilihan sudah sangat lebih banyak maka penggunaan statement if akan membuat bingung pengguna. Untuk mengatasi ini maka digunakanlah statement switch-case untuk menyederhanakan statement if-else yang bertingkat. Semua masalah yang bisa diselesaikan dengan statement switch hampir dipastikan dapat diselesaikan dengan statement if. Tetapi hal ini tidak berlaku sebaliknya. Struktur pemrograman menggunakan statement switch-case adalah:

```
Switch(kondisi)
{
    case konstanta1: Pernyataan_1; break
    case konstanta2: Pernyataan_2; break
    case konstanta3: Pernyataan_3; break
}
```

## C. Percobaan

### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535
- 3) 1 modul aplikasi keypad

### b) Langkah Percobaan

- 1) Hubungkan kabel alamat aplikasi seven segment pada Port-D, kabel kendali seven segment pada Port-C, dan hubungkan kabel data keypad pada Port-B mikrokontroler.
- 2) Bukalah program CodeVisionAVR
- 3) Buatlah project baru dengan ketentuan DDRC dan DDRD sebagai output, isi DDRB = 0xF0 dan PORTB=0xFF.
- 4) Beri nama project ini Praktek-9
- 5) Ketiklah header delay sebagai berikut:

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

- 6) Ketiklah *fungsi* program sebagai berikut:

```
unsigned char dt,dtkey;
```

```

void detek_key(void)
{
    PORTB.4=0;
    dt=(~PINB & 0x0F);
    switch (dt) {
        case 1: dtkey=0b11111001;
            break;
        case 2: dtkey=0b11111111;
            break;
        case 4: dtkey=0b11111111;
            break;
        case 8: dtkey=0b11111111;
            break;
    };

    PORTB.4=1;
    PORTB.5=0;
    dt=(~PINB & 0x0F);
    switch (dt) {
        case 1: dtkey=0b11111111;
            break;
        case 2: dtkey=0b10010010;
            break;
        case 4: dtkey=0b11111111;
            break;
        case 8: dtkey=0b11111111;
            break;
    };
    PORTB.5=1;
    PORTB.6=0;
    dt=(~PINB & 0x0F);
    switch (dt) {
        case 1: dtkey=0b11111111;
            break;
        case 2: dtkey=0b11111111;
            break;
        case 4: dtkey=0b10010000;
            break;
        case 8: dtkey=0b11111111;
            break;
    };
    PORTB.6=1;
}

```

7) Program utama diketik sebagai berikut.

```

while (1)
{
    // Place your code here
    detek_key();
    PORTC.6=1;
    PORTC.7=0;
    PORTD=dtkey;
    delay_ms(5);
}

```

- 8) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.
- 9) Jika tidak terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara memilih *Program the chip*.

**c) Hasil Percobaan**

- 1) Amatilah hasil tampilan seven segment, apa yang terjadi?
- 2) Apa yang terjadi pada seven segment jika keypad ditekan (tekan angka 1 s/d 0)?

**d) Tugas dan Pertanyaan**

- 1) Buatlah program agar seven segment menyala sesuai angka yang ditekan pada keypad.
- 2) Jelaskan maksud program Praktek-9

# APLIKASI SENSOR ULTRASONIK

Modul-6A

## A. Tujuan

- 1) Mahasiswa dapat menjalankan program CodeVisionAVR
- 2) Mahasiswa dapat memahami cara pemrograman sensor ultrasonik pada mikrokontroler

## B. Dasar Teori

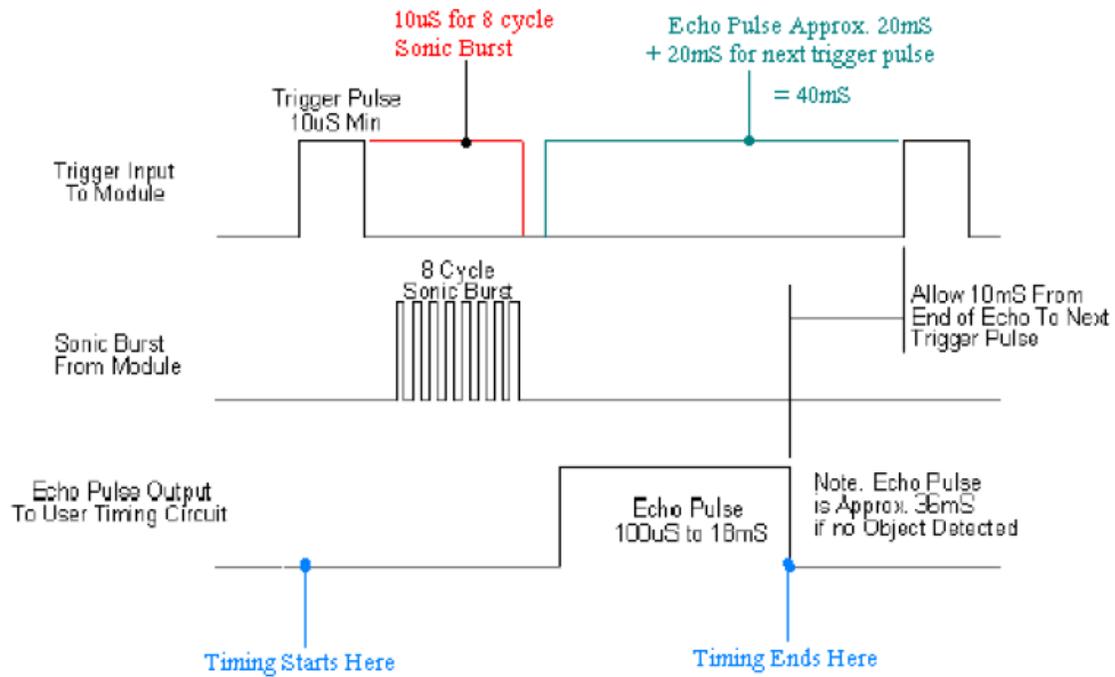
### Sensor Ultrasonik

Sensor jarak yang paling umum digunakan dalam dunia robotika adalah menggunakan teknologi ultrasonic. Salah satu jenis sensor ini adalah SRF04 seperti tampak pada Gambar 22. Berdasarkan data sheet yang dikeluarkan oleh produsennya sensor ini mampu mendeteksi jarak hingga 500 cm.



Gambar 22. Sensor ultrasonik SRF04

Sensor ini mempunyai 5-pin yang masing-masing terhubung ke tegangan +5v, pulsa pantulan (echo), pulsa pemisu (trigger), dan 1 pin lagi tidak dihubungkan (tidak terpakai). Cara pemrogramannya berdasarkan Gambar 23 yaitu diagram pewaktuan (timing diagram) yang telah disediakan di data sheetnya.



Gambar 23. Diagram pewaktuan sensor SRF04

### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535
- 3) 1 modul sensor ultrasonik SRF04

#### b) Langkah Percobaan

- 1) Hubungkan aplikasi ultrasonik pada Port-B dan aplikasi LCD ke Port-C mikrokontroler.
- 2) Bukalah program CodeVisionAVR
- 3) Buatlah project baru dengan ketentuan DDRB=0b00000001.
- 4) Beri nama project ini Praktek-10
- 5) Ketiklah header delay sebagai berikut:

```
#include <lcd.h>
#include <delay.h>
#include <stdio.h>
#define TRIGGER PORTB.0
#define ECHO PINB.1
unsigned int baca_jarak();
```

6) Ketiklah program utama sebagai berikut:

```
while (1)
{
// Place your code here
if (baca_jarak()<=1){lcd_gotoxy(0,0); lcd_putsf("kira2 10 cm");}
else if (baca_jarak()<=2) {lcd_gotoxy(0,0); lcd_putsf("kira2 20 cm");}
else if (baca_jarak()<=3) {lcd_gotoxy(0,0); lcd_putsf("kira2 30 cm");}
else if (baca_jarak()<=4) {lcd_gotoxy(0,0); lcd_putsf("kira2 40 cm");}
else {lcd_gotoxy(0,0); lcd_putsf("lebih jauh dari 40 cm");}
};
```

7) Buatlah *fungsi* sebagai berikut:

```
unsigned char jarak;
unsigned int baca_jarak()
{
jarak=0;
TRIGGER=0;
delay_us(10);
TRIGGER=1;
delay_us(20);
TRIGGER=0;
while (ECHO==0);
while (ECHO==1) {jarak++; delay_us(710);}
delay_ms(40);
return jarak;
}
```

8) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.

9) Jika tidak terjadi kesalahan, maka lakukan pemrograman ke chip dengan cara memilih *Program the chip*.

**c) Hasil Percobaan**

- 1) Amati sensor jika tidak terhalang oleh benda apapun !
- 2) Apa yang terjadi pada LCD jika sensor ultrasonik diberi halangan secara maju dan mundur?

**d) Tugas dan Pertanyaan**

- 1) Jelaskan maksud program Praktek-10

## APLIKASI MOTOR DC

Modul-6B

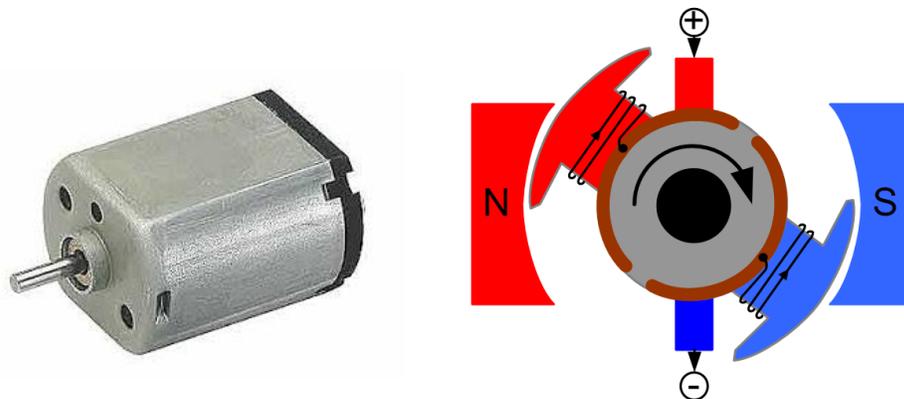
### A. Tujuan

- 1) Mahasiswa dapat menjalankan program CodeVisionAVR
- 2) Mahasiswa dapat memahami cara pemrograman motor dc dengan menggunakan mikrokontroler

### B. Dasar Teori

#### Motor DC

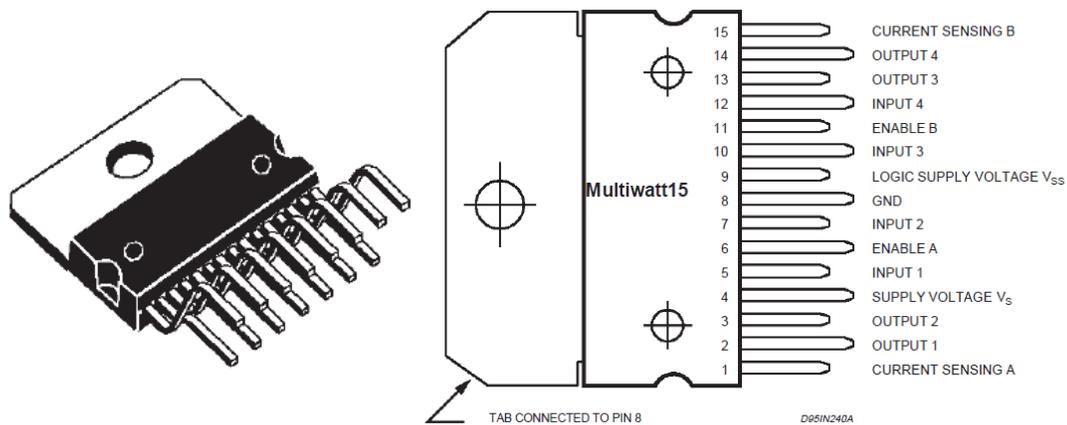
Piranti elektronik yang dapat mengubah energi listrik menjadi energi mekanik/gerak disebut dengan motor. Jenis motor ada 2 yaitu motor AC dan motor DC. Motor DC mempunyai keunggulan dibandingkan dengan motor AC disisi kemudahan pengendaliannya. Motor ini dapat diubah arah putarannya dengan membalik polaritasnya saja. Kecepatan motor DC dapat diatur salah satunya adalah dengan cara mengatur tegangan yang masuk ke dalamnya. Hal ini sering disebut dengan istilah pulse width modulation (PWM). Bagian motor terdiri atas stator (bagian yang diam) dan bagian rotor (bagian yang berputar). Salah satu jenis motor dan bagian-bagiannya ditunjukkan pada Gambar 24.



Gambar 24. Contoh satu jenis motor DC dan bagian-bagiannya

#### Driver Motor IC L298

Port-port mikrokontroler dapat difungsikan sebagai input maupun output. Ketika difungsikan sebagai output maka port yang bersangkutan akan menghasilkan suatu arus. Akan tetapi arus yang dihasilkan masih terlalu kecil jika digunakan untuk menggerakkan suatu motor. Oleh karena itu diperlukanlah sebuah driver yang berfungsi memperkuat arus. Driver dapat disusun dengan menggunakan rangkaian H-bridge transistor, mosfet, maupun IC. Salah satu jenis IC driver yang umum dipasaran adalah L298. Gambar 25 memperlihatkan IC driver L298 dan fungsi masing-masing pin.



Gambar 25. Bentuk fisik dan Susunan pin IC L298

IC ini mampu dibebani hingga 4A pada tegangan 6V – 46V. Dalam L298 terdapat dua rangkaian H-Bridge sehingga mampu mengendalikan 2 motor sekaligus dengan arus beban masing-masing 2 A.

### C. Percobaan

#### a) Alat yang diperlukan

- 1) 1 unit komputer dengan dilengkapi software CVAVR
- 2) 1 unit modul hardware mikrokontroler ATmega8535
- 3) 1 modul motor DC

#### b) Langkah Percobaan

- 1) Bukalah program CodeVisionAVR
- 2) Buatlah project baru dengan ketentuan Code wizard Timer1 diatur clock value (187.500 kHz) Mode (Fast PWM Top = 00FFH) Out A (non invt) Out B (non invt) dan PORTD sebagai output.
- 3) Beri nama project ini Praktek-11.
- 4) Ketiklah header delay sebagai berikut:

```
#include <delay.h>
#define kirplus PORTD.1
#define kirmin PORTD.2
#define kanplus PORTD.6
#define kanmin PORTD.3
#define lpwm OCR1B
#define rpwm OCR1A
```

- 5) Ketiklah program utama sebagai berikut:

```
while (1)
{
//Place your code here
kirplus=1;kirmin=0;
lpwm=0;rpwm=0;
delay_ms(1000);
kirplus=0;kirmin=1;
lpwm=250;rpwm=250;
```

6) Buatlah *fungsi* sebagai berikut:

```
void maju()
{
  kirplus=1;   kanplus=1;
  kirmin=0;    kanmin=0;
}

void mundur()
{
  kirplus=0;   kanplus=0;
  kirmin=1;    kanmin=1;
}

void belok_kanan()
{
  kirplus=1;   kanplus=0;
  kirmin=0;    kanmin=1;
}

void belok_kiri()
{
  kirplus=0;   kanplus=1;
  kirmin=1;    kanmin=0;
}

void stop()
{
  kirplus=0;   kanplus=0;
  kirmin=0;    kanmin=0;
}
```

- 7) Simpan, *compile* dan *make* program tersebut. Perhatikan informasi yang disampaikan.
- 8) Jika tidak terjadi kesalahan, maka lakukan pemrograman ke chip pada modul robot beroda dengan cara memilih *Program the chip*.
- 9) Pindahkan saklar pada robot beroda pada posisi on.

**c) Hasil Percobaan**

- 1) Amati yang terjadi pada motor robot beroda dalam keadaan diangkat dengan tangan.
- 2) Letakkan robot beroda pada lantai dan amati apa yang terjadi!

**d) Tugas dan Pertanyaan**

- 1) Modifikasilah program agar robot berjalan maju sejauh 100 cm dan kemudian mundur ke tempat semula.
- 2) Buatlah program agar robot beroda bergerak maju membentuk lintasan persegi dengan panjang sisi 50 cm.

## DAFTAR PUSTAKA

- Andrianto, G, 2008, *Pemrograman Mikrokontroler AVR Atmega16, Menggunakan Bahasa C*, Penerbit Informatika, Bandung.
- Budioko, T, 2005, *Belajar dengan Mudah dan Cepat Pemrograman Bahasa C dengan SDCC pada Mikrokontroler AT89X051/AT89C51/52 Teori, Simulasi dan Aplikasi*, Gava media, Yogyakarta
- Junaedi F, 2007, *Algoritma dan Pemrograman*, Penerbit Salemba Infotek, Jakarta.
- Keunsuk, Lee. 2002, *Application of The Devantec SRF04 Ultrasonic Rang Finder*.
- Munir, R, 2005, *Algoritma dan Pemrograman dalam bahasa Pascal dan C*, Edisi ke-3, Penerbit Informatika, Bandung
- Raharjo, B, 2007, *Pemrograman C++ Mudah dan Cepat Menjadi Master C*, Penerbit Informatika, Bandung.
- , 2006, *Data Sheet ATmega8535*, Atmel Corporation, <http://www.atmel.com/literature> diakses tanggal 20 Juli 2013.
- , 2003, *Code Vision AVR V1.23.9b User Manual*, Pavel Haiduc and HP Infotesh S.R.L, Romania, <http://www.hpinfotech.ro> diakses tanggal 19 Januari 2012.