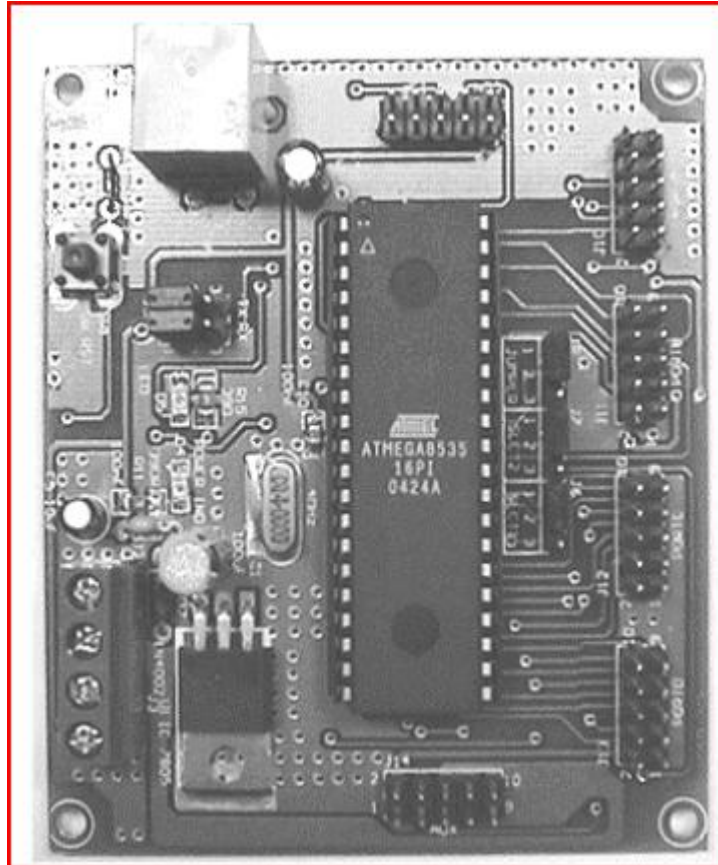


Workshop

MIKROKONTROLER ATMEGA16

DI SMA Muhammadiyah BANTUL



Jurusan Teknik Elektro
Fakultas Teknik
Universitas Muhammadiyah Yogyakarta
2013

MIKROKONTROLER ATMEGA16

Apa itu Mikrokontroler?



Komputer hadir dalam kehidupan manusia baru 50 tahun terakhir, namun efeknya sangat besar dalam merubah kehidupan manusia, bahkan melebihi penemuan manusia lainnya seperti radio, telepon, automobil, dan televisi. Begitu banyak aplikasi memanfaatkan komputer, terutama dalam pemanfaatan kemampuan chip mikroprosesor di dalamnya yang dapat melakukan komputasi sangat cepat, dapat bekerja sendiri dengan diprogram, dan dilengkapi memori untuk menyimpan begitu banyak data. Seiring dengan perkembangan zaman, semakin luaslah kebutuhan akan kemampuan seperti yang dimiliki oleh komputer, sehingga menyebabkan munculnya terobosan-terobosan baru yang salah satunya adalah dibuatnya chip mikrokontroler.

Mikrokontroler adalah single chip computer yang memiliki kemampuan untuk diprogram dan digunakan untuk tugas-tugas yang berorientasi kontrol. Mikrokontroler datang dengan dua alasan utama, yang pertama adalah kebutuhan pasar (market need) dan yang kedua adalah perkembangan teknologi baru. Yang dimaksud dengan kebutuhan pasar adalah kebutuhan yang luas dari produk-produk elektronik akan perangkat pintar sebagai pengontrol dan pemroses data. Sedangkan yang dimaksud dengan perkembangan teknologi baru adalah perkembangan teknologi semikonduktor yang memungkinkan pembuatan chip dengan kemampuan komputasi yang sangat cepat, bentuk yang semakin mungil, dan harga yang semakin murah.

APLIKASI MIKROKONTROLER

Karena kemampuannya yang tinggi, bentuknya yang kecil, konsumsi dayanya yang rendah, dan harga yang murah maka mikrokontroler begitu banyak digunakan di dunia. Mikrokontroler digunakan mulai dari mainan anak-anak, perangkat elektronik rumah tangga, perangkat pendukung otomotif, peralatan industri, peralatan telekomunikasi, peralatan medis dan kedokteran, sampai dengan pengendali robot serta persenjataan militer. Terdapat beberapa keunggulan yang diharapkan dari alat-alat yang berbasis mikrokontroler (microcontroller-based solutions) :

- Keandalan tinggi (high reliability) dan kemudahan integrasi dengan komponen lain (high degree of integration)
- Ukuran yang semakin dapat diperkecil (reduced in size)
- Penggunaan komponen dipersedikit (reduced component count) yang juga akan menyebabkan biaya produksi dapat semakin ditekan (lower manufacturing cost)
- Waktu pembuatan lebih singkat (shorter development time) sehingga lebih cepat pula dijual ke pasar sesuai kebutuhan (shorter time to market)
- Konsumsi daya yang rendah (lower power consumption)

Overview Mikrokontroler AVR



Mikrokontroler AVR merupakan salah satu jenis arsitektur mikrokontroler yang menjadi andalan Atmel. Arsitektur ini dirancang memiliki berbagai kelebihan dan merupakan penyempurnaan dari arsitektur mikrokontroler-mikrokontroler yang sudah ada.

Berbagai seri mikrokontroler AVR telah diproduksi oleh Atmel dan digunakan di dunia sebagai mikrokontroler yang bersifat low cost dan high performance. Di Indonesia, mikrokontroler AVR banyak dipakai karena fiturnya yang cukup lengkap, mudah untuk didapatkan, dan harganya yang relatif terjangkau.

Antar seri mikrokontroler AVR memiliki beragam tipe dan fasilitas, namun kesemuanya memiliki arsitektur yang sama, dan juga set instruksi yang relatif tidak berbeda. Berikut tabel perbandingan beberapa seri mikrokontroler AVR buatan Atmel.

Seri	Flash (KBytes)	RAM (Bytes)	EEPROM (KBytes)	Pin I/O	Timer 16-bit	Timer 8-bit	UART	PWM	ADC 10-bit	SPI	ISP
ATmega8	8	1024	0.5	23	1	1	1	3	6/8	1	Ya
ATmega8535	8	512	0.5	32	2	2	1	4	8	1	Ya
ATmega16	16	1024	0.5	32	1	2	1	4	8	1	Ya
ATmega162	16	1024	0.5	35	2	2	2	6	8	1	Ya
ATmega32	32	2048	1	32	1	2	1	4	8	1	Ya
ATmega128	128	4096	4	53	2	2	2	8	8	1	Ya
ATtiny12	1	-	0.0625	6	-	1	-	-	-	-	Ya
ATtiny2313	2	128	0.125	18	1	1	1	4	-	1	Ya
ATtiny44	4	256	0.25	12	1	1	-	4	8	1	Ya
ATtiny84	8	512	0.5	12	1	1	-	4	8	1	Ya

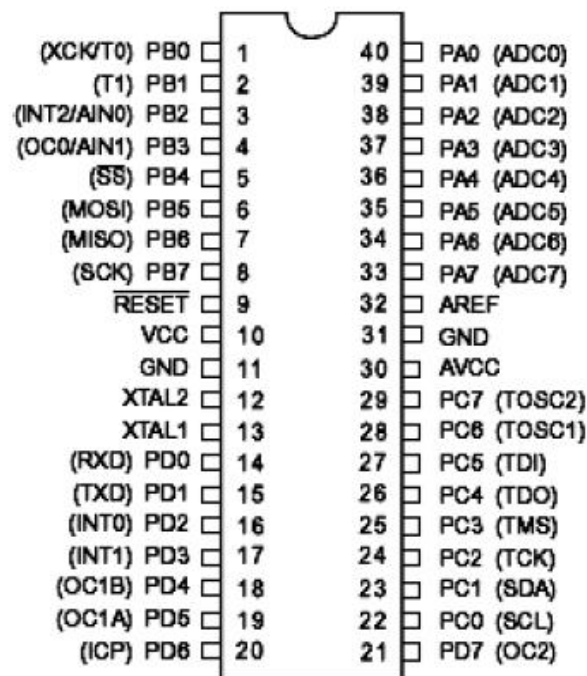
Keterangan:

- Flash adalah suatu jenis Read Only Memory yang biasanya diisi dengan program hasil buatan manusia yang harus dijalankan oleh mikrokontroler
- RAM (Random Acces Memory) merupakan memori yang membantu CPU untuk penyimpanan data sementara dan pengolahan data ketika program sedang running
- EEPROM (Electrically Erasable Programmable Read Only Memory) adalah memori untuk penyimpanan data secara permanen oleh program yang sedang running
- Port I/O adalah kaki untuk jalur keluar atau masuk sinyal sebagai hasil keluaran ataupun masukan bagi program
- Timer adalah modul dalam hardware yang bekerja untuk menghitung waktu/pulsa
- UART (Universal Asynchronous Receive Transmit) adalah jalur komunikasi data khusus secara serial asynchronous
- PWM (Pulse Width Modulation) adalah fasilitas untuk membuat modulasi pulsa
- ADC (Analog to Digital Converter) adalah fasilitas untuk dapat menerima sinyal analog dalam range tertentu untuk kemudian dikonversi menjadi suatu nilai digital dalam range tertentu
- SPI (Serial Peripheral Interface) adalah jalur komunikasi data khusus secara serial secara serial synchronous
- ISP (In System Programming) adalah kemampuan khusus mikrokontroler untuk dapat diprogram langsung dalam sistem rangkaiannya dengan membutuhkan jumlah

pin yang minimal

Mengenal Mikrokontroler ATMEGA16

AVR merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur *RISC (Reduced Instruction Set Computer)*. Hampir semua instruksi dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register general-purpose, timer/counter fleksibel dengan mode *compare*, *interrupt internal* dan *eksternal*, serial *UART*, *programmable Watchdog Timer*, dan *mode power saving*, ADC dan PWM internal. AVR juga mempunyai *In-System Programmable Flash on-chip* yang memungkinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial *SPI*. ATmega16. ATmega16 mempunyai *throughput* mendekati 1 MIPS per MHz membuat disainer sistem untuk mengoptimasi konsumsi daya versus kecepatan proses.



Gambar 1 Pin-pin ATmega16 kemasan 40-pin

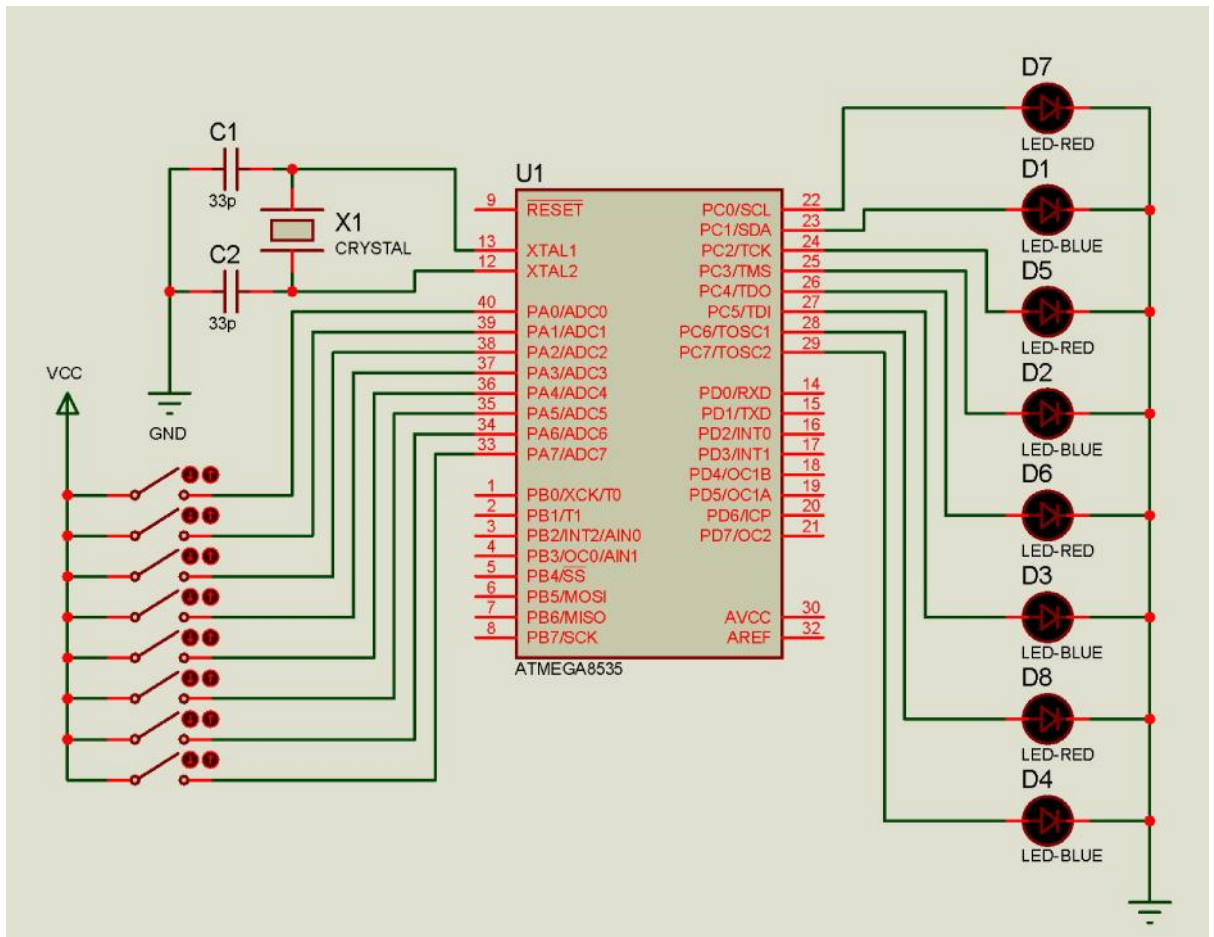
Pin-pin pada ATmega16 dengan kemasan 40-pin DIP (*dual inline package*) ditunjukkan oleh gambar 1. Guna memaksimalkan performa, AVR menggunakan arsitektur *Harvard* (dengan memori dan bus terpisah untuk program dan data)[1].

Port sebagai input/output digital

ATmega16 mempunyai empat buah port yang bernama *PortA*, *PortB*, *PortC*, dan *PortD*[2]. Keempat port tersebut merupakan jalur *bidirectional* dengan pilihan *internal pull-up*. Tiap port mempunyai tiga buah register bit, yaitu *DDxn*, *PORTxn*, dan *PINxn*. Huruf 'x' mewakili

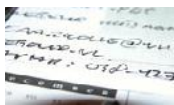
nama huruf dari port sedangkan huruf 'n' mewakili nomor bit. Bit DDxn terdapat pada I/O address DDRx, bit PORTxn terdapat pada I/O address PORTx, dan bit PINxn terdapat pada I/O address PINx. Bit DDxn dalam register DDRx (*Data Direction Register*) menentukan arah pin. Bila DDxn diset 1 maka Px berfungsi sebagai pin output. Bila DDxn diset 0 maka Px berfungsi sebagai pin input. Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin input, maka resistor *pull-up* akan diaktifkan. Untuk mematikan resistor *pull-up*, PORTxn harus diset 0 atau pin dikonfigurasi sebagai pin output. Pin port adalah *tri-state* setelah kondisi reset. Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin output maka pin port akan berlogika 1. Dan bila PORTxn diset 0 pada saat pin terkonfigurasi sebagai pin output maka pin port akan berlogika 0. Saat mengubah kondisi port dari kondisi *tri-state* (DDxn=0, PORTxn=0) ke kondisi *output high* (DDxn=1, PORTxn=1) maka harus ada kondisi peralihan apakah itu kondisi *pull-up enabled* (DDxn=0, PORTxn=1) atau kondisi *output low* (DDxn=1, PORTxn=0).

Biasanya, kondisi *pull-up enabled* dapat diterima sepenuhnya, selama lingkungan impedansi tinggi tidak memperhatikan perbedaan antara sebuah *strong high driver* dengan sebuah *pull-up*. Jika ini bukan suatu masalah, maka bit PUD pada register SFIOR dapat diset 1 untuk mematikan semua *pull-up* dalam semua port. Peralihan dari kondisi *input dengan pull-up* ke kondisi *output low* juga menimbulkan masalah yang sama. Kita harus menggunakan kondisi *tri-state* (DDxn=0, PORTxn=0) atau kondisi *output high* (DDxn=1, PORTxn=0) sebagai kondisi transisi.



Pengantar Pemrograman C dengan Code Vision AVR

Pemrograman Bahasa C untuk AVR



Bahasa C luas digunakan untuk pemrograman berbagai jenis perangkat, termasuk mikrokontroler. Bahasa ini sudah merupakan high level language, dimana memudahkan programmer menuangkan algoritmanya. Untuk mengetahui dasar bahasa C dapat dipelajari sebagai berikut [3].

1. Struktur penulisan program

```
#include < [library1.h] > // Opsional
#include < [library2.h] > // Opsional

#define [nama1] [nilai] ; // Opsional
#define [nama2] [nilai] ; // Opsional
```

```
[global variables] // Opsional

[functions] // Opsional

void main(void) // Program Utama
{
[Deklarasi local variable/constant]
[Isi Program Utama]
}
```

2. Tipe data

- char : 1 byte (-128 s/d 127)
- unsigned char : 1 byte (0 s/d 255)
- int : 2 byte (-32768 s/d 32767)
- unsigned int : 2 byte (0 s/d 65535)
- long : 4 byte (-2147483648 s/d 2147483647)
- unsigned long : 4 byte (0 s/d 4294967295)
- float : bilangan desimal
- array : kumpulan data-data yang sama tipenya.

3. Deklarasi variabel & konstanta

- Variabel adalah memori penyimpanan data yang nilainya dapat diubah-ubah.
Penulisan : [tipe data] [nama] = [nilai] ;
- Konstanta adalah memori penyimpanan data yang nilainya tidak dapat diubah.
Penulisan : const [nama] = [nilai] ;
- Tambahan:
Global variabel/konstanta yang dapat diakses di seluruh bagian program.
Local variabel/konstanta yang hanya dapat diakses oleh fungsi tempat dideklarasikannya.

4. Statement

Statement adalah setiap operasi dalam pemrograman, harus diakhiri dengan [;] atau [}]. Statement tidak akan dieksekusi bila diawali dengan tanda [//] untuk satu baris. Lebih dari 1 baris gunakan pasangan [/*] dan [*/]. Statement yang tidak dieksekusi disebut juga comments / komentar [4].

Contoh:

```
suhu=adc/255*100; //contoh rumus perhitungan suhu
```

5. Function

Function [5] adalah bagian program yang dapat dipanggil oleh program utama.

Penulisan :

```
[tipe data hasil] [nama function]([tipe data input 1],[tipe data input 2])
{
[statement] ;
}
```

6. Conditional statement dan looping

if else : digunakan untuk penyeleksian kondisi

```
if ( [persyaratan] ) {
[statement1];
[statement2];
}
else {
[statement3];
[statement4];
}
```

for : digunakan untuk looping dengan jumlah yang sudah diketahui

```
for ( [nilai awal] ; [persyaratan] ; [operasi nilai] ) {
[statement1];
[statement2];
}
```

while : digunakan untuk looping jika dan selama memenuhi syarat tertentu

```
while ( [persyaratan] ) {
[statement1];
[statement2];
}
```

do while : digunakan untuk looping jika dan selama memenuhi syarat tertentu, namun min 1 kali

```
do {
[statement1];
[statement2];
}
while ( [persyaratan] )
```

switch case : digunakan untuk seleksi dengan banyak kondisi


```

switch ( [nama variabel] ) {
case [nilai1]: [statement];
break;
case [nilai2]: [statement];
break;
}

```

7. Operasi logika dan biner

- Logika
AND : &&
NOT : !
OR : ||
- Biner
AND : &
OR : |
XOR : ^
Shift right : >>
Shift left : <<
Komplemen : ~

8. Operasi relasional (perbandingan)

- Sama dengan : ==
- Tidak sama dengan : !=
- Lebih besar : >
- Lebih besar sama dengan : >=
- Lebih kecil : <
- Lebih kecil sama dengan : <=

9. Operasi aritmatika

- + , - , * , / : tambah, kurang, kali, bagi
- += , -= , *= , /= : nilai di sebelah kiri operator di tambah/kurang/kali/bagi dengan nilai di sebelah kanan operator
- % : sisa bagi
- ++ , -- : tambah satu (increment) , kurang satu (decrement)
- Contoh :
a = 5 * 6 + 2 / 2 - 1 ; maka nilai a adalah 30
a *= 5 ; jika nilai awal a adalah 30, maka nilai a = 30x5 = 150.
a += 3 ; jika nilai awal a adalah 30, maka nilai a = 30+5 = 33.
a++ ; jika nilai awal a adalah 5 maka nilai a = a+1 = 6.
a-- ; jika nilai awal a adalah 5 maka nilai a = a-1 = 4.

Contoh Pemrograman AVR Dengan Codevision

Tentang CodeVisionAVR

CodeVisionAVR merupakan sebuah cross-compiler C, Integrated Development Environment (IDE), dan Automatic Program Generator yang didesain untuk mikrokontroler buatan Atmel seri AVR. CodeVisionAVR dapat dijalankan pada sistem operasi Windows 95, 98, Me, NT4, 2000, dan XP. Cross-compiler C mampu menerjemahkan hampir semua perintah dari bahasa ANSI C, sejauh yang diijinkan oleh arsitektur dari AVR, dengan tambahan beberapa fitur untuk mengambil kelebihan khusus dari arsitektur AVR dan kebutuhan pada sistem embedded.

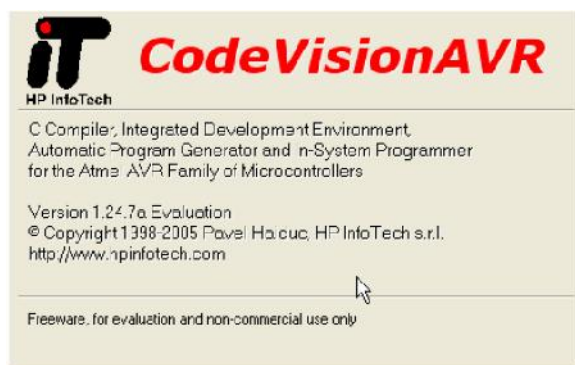
File object COFF hasil kompilasi dapat digunakan untuk keperluan debugging pada tingkatan C, dengan pengamatan variabel, menggunakan debugger Atmel AVR Studio. IDE mempunyai fasilitas internal berupa software AVR Chip In-System Programmer yang memungkinkan Anda untuk melakukan transfer program kedalam chip mikrokontroler setelah sukses melakukan kompilasi/assembly secara otomatis. Software In-System Programmer didesain untuk bekerja dengan Atmel STK500/AVRISP/AVRProg, Kanda Systems STK200+/300, Dontronics DT006, Vogel Elektronik VTEC-ISP, Futurlec JRAVR dan MicroTronics ATCPU/Mega2000 programmers/development boards [6]–[10].

Menjalankan Codevision AVR

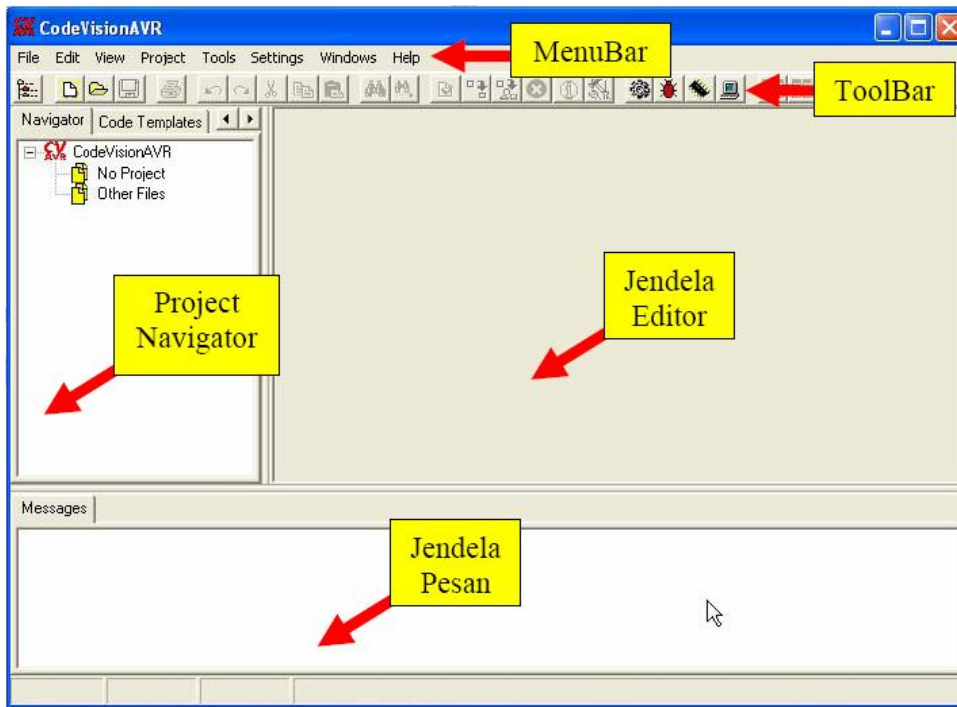
Jalankan aplikasi CodeVisionAVR dengan cara melakukan klik ganda pada *shortcut* ikon CodeVisionAVR yang terbentuk pada Desktop [11]–[14].




Sebuah tampilan akan muncul sebagai berikut :



Beberapa detik kemudian akan muncul tampilan :



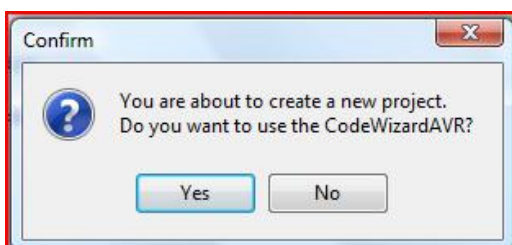
Langkah-langkah untuk membuat project baru (menyalakan LED di Port B) dengan menggunakan codeVisionAVR adalah sebagai berikut :

1. Pilih menu [File] → [New] atau mengklik toolbar  kemudian akan muncul kotak dialog :

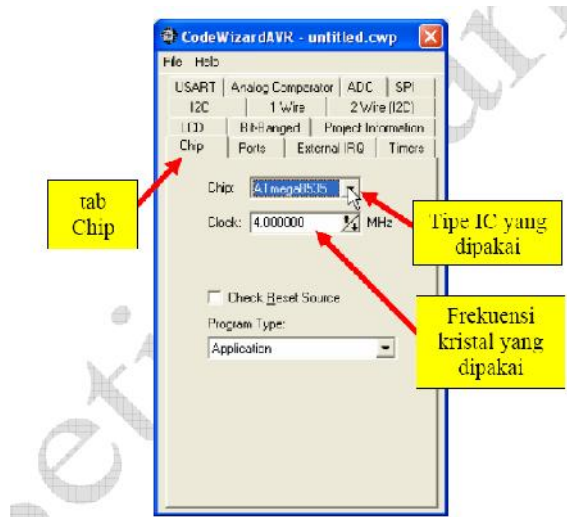


Jika kita menginginkan untuk membuat sebuah project baru maka pilih [File Type] → Project, terus tekan OK [15]–[21]

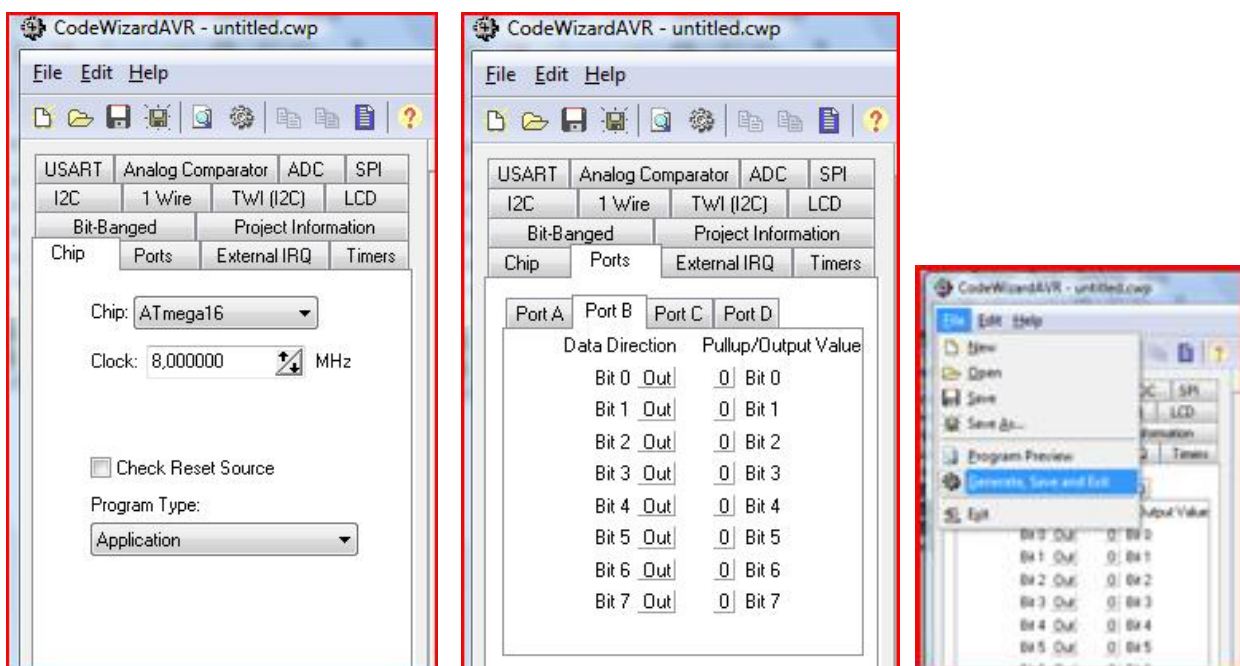
2. Kemudian muncul kotak dialog apakah akan menggunakan CodeWizard AVR untuk mempermudah membuat kerangka program, pilih YES.



3. Maka akan muncul tampilan sebagai berikut :



4. Pilih chip yang digunakan adalah AT Mega16 dengan clock 11,520000 MHz, kemudian pilih tab Ports, lakukan setting untuk PORTB sebagai output, kemudian pilih File Generate, Save and Exit, simpan source file, project file dan codewizard project.



5. Tambahkan beberapa perintah ke dalam kerangka program yang sudah dibuat, untuk menyalakan LED pada PORTB.

- a. Tambahkan `#include<delay.h>`
- b. Pada main program sisipkan perintah berikut :

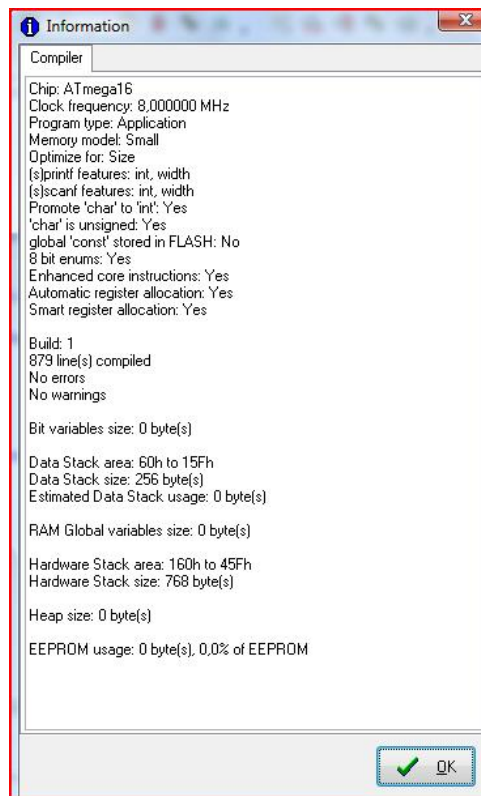

```
while (1)
{
    //Place your code here
```

```



PORTB = 0x0f; // nyalakan led PB.0 – PB.3
delay_ms(100);
PORTB=0xf0; // nyalakan led PB.4 – PB.7
Delay_ms(100);
}

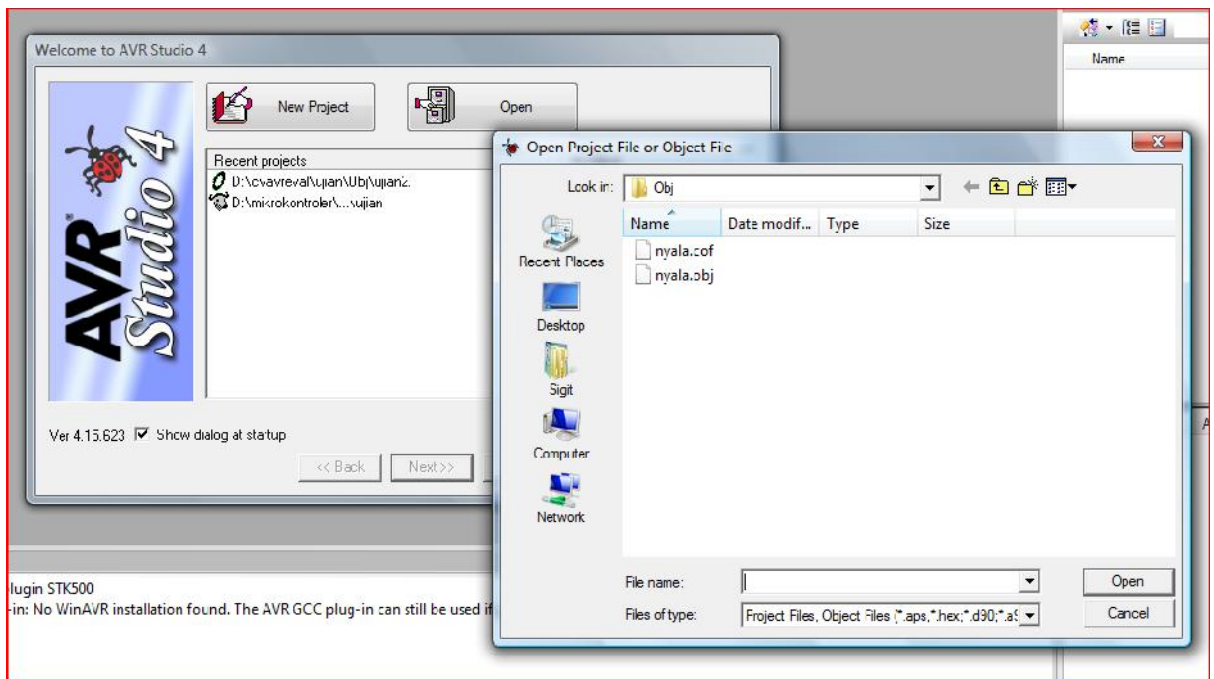
```

6. Kemudian klik tab Project → Compile , untuk memeriksa apakah ada kesalahan. Tampilan yang nampak adalah sebagai berikut :

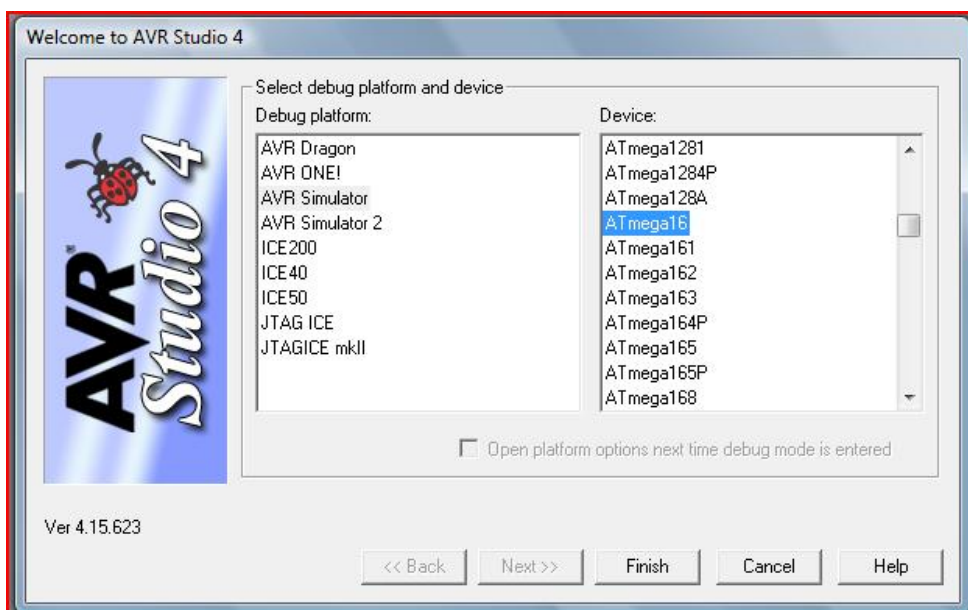


Kemudian tekan OK.

7. Kemudian klik Project → Build All . Jika tidak ada kesalahan maka kita bisa lanjutkan ke pemrograman chip atau disimulasikan ke AVR Studio.
8. Klik Tools → debugger atau klik toolbar  , dengan syarat kita sdh mensetting debugger yang digunakan adalah AVR Studio 4 dengan klik  dan mencari lokasi AVR Studio 4 .exe disimpan.
9. Setelah jendela AVR Studio terbuka maka kita pilih menu OPEN dan cari file berekstensi cof yg telah kita simpan (hasil eksekusi CodeVision).



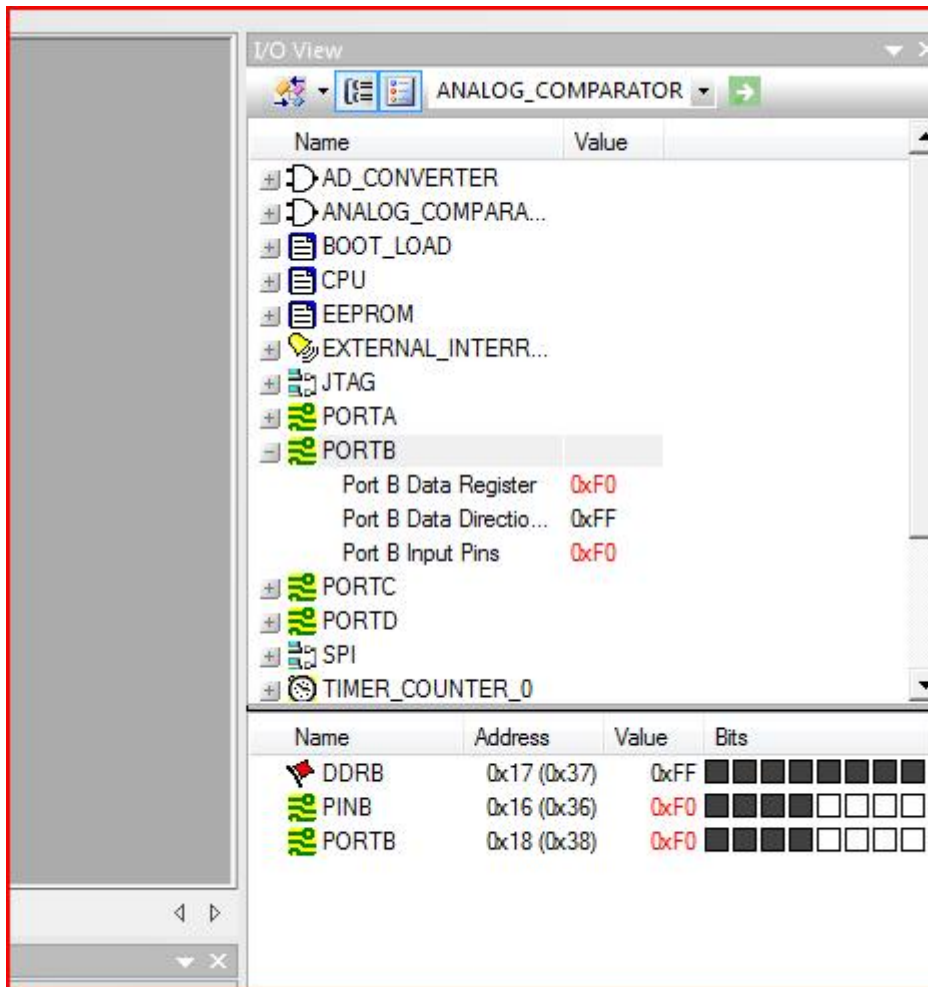
Kemudian file cof tersebut kita pilih (sorot) kemudian klik Open. Kemudian klik SAVE. Akan tampil jendela sebagai berikut :



Pilih AVR Simulator dan Device : Atmega16 , kemudian klik Finish.

Untuk mensimulasikan klik Debug → Run (F5) atau step by step dengan menekan F11.

Tampilan eksekusi program sebagai berikut :



Untuk di program langsung ke board sistem minimum AVR ATmega 16 akan dijelaskan pada labsheet berikutnya.

Daftar Pustaka

- [1] I. Iswanto, *Antarmuka Port Pararel dan Port Serial dengan Delphi 6 Compatible Sistem Operasi Windows*. Gava Media, 2008.
- [2] Iswanto, *Design dan Implementasi Sistem Embedded Mikrokontroler ATMEGA8535 dengan Bahasa Basic*. Yogyakarta: Gava Media, 2008.
- [3] I. Iswanto, N. M. Raharja, and A. Subardono, "Sistem Peringatan Dini Tanah Longsor Berbasis Atmega8535," in *Seminar Nasional Informatika 2009 (semnasIF 2009)*, 2009, pp. 53–57.
- [4] I. Iswanto, *Belajar Sendiri Mikrokontroller AT90S2313 Dengan Basic Compiler*. Andi Publisher, 2009.
- [5] D. B. Prasetya, Iswanto, and R. T. A. Sadad, "Implementasi Mikrokontroler Sebagai Pengendali Kapasitor Untuk Perbaikan Faktor Daya Otomatis pada Jaringan Listrik," *SEMESTA Tek.*, vol. 13, no. 2, pp. 181–192, 2010.
- [6] ISWANTO, A. JAMAL, and F. SETIADY, "Implementasi Telepon Seluler sebagai Kendali Lampu Jarak Jauh," *J. Ilm. Semesta Tek.*, vol. 14, no. 1, pp. 81–85, 2011.
- [7] A. N. N. Chamim and Iswanto, "Implementasi Mikrokontroler Untuk Pengendalian Lampu Dengan Sms," in *Prosending Retii 6.*, 2011, pp. 30–34.
- [8] M. yusvin Mustar, R. Tsaqif As Sadad, and Iswanto, "IMPLEMENTASI ROBOT TANK MENGGUNAKAN KAMERA CCTV WIRELLES BERBASIS MIKROKONTROLER ATmega 8535L," in *SEMINAR NASIONAL ke 6 Tahun 2011 : Rekayasa Teknologi Industri dan Informasi*, 2011, pp. 24–29.
- [9] N. M. Raharja and Iswanto, "Monitoring Sistem Pengendalian Suhu dan Saluran Irigasi Hidroponik pada Greenhouse Berbasis Web," *Pros. Semin. Ilm. Nas. Komput. dan Sist. Intelijen (KOMMIT 2012)*, vol. 7, no. September, pp. 303–310, 2012.
- [10] R. T. A. Sadad and Iswanto, "Implementasi mikrokontroler untuk deteksi drop tegangan pada instalasi sederhana," in *Prosiding Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT 2012)*, 2012, pp. 368–376.
- [11] I. Iswanto and N. M. Raharja, "Sistem monitoring dan peringatan dini tanah longsor," in *Simposium Nasional RAPI IX 2010*, 2010, pp. 54–62.
- [12] I. Ambar Tri Utomo, Ramadani Syahputra, "Implementasi Mikrokontroller Sebagai Pengukur Suhu Delapan Ruangan," *J. Teknol.*, vol. 4, no. 2, pp. 153–159, 2011.
- [13] R. T. A. SADAD, ISWANTO, and J. A. SADAD, "Implementasi Mikrokontroler Sebagai Pengendali Lift Empat Lantai," *J. Ilm. SEMESTA Tek.*, vol. 14, no. 2, pp. 160–165, 2011.
- [14] R. T. A. Sadad and Iswanto, "Peranan Teknologi Solar Cell dalam Peningkatan Daya Saing Usaha Kecil dan Menengah," *SEMESTA Tek.*, vol. 14, no. 1, pp. 58–63, 2011.
- [15] ISWANTO and H. MUHAMMAD, "WEATHER MONITORING STATION WITH REMOTE RADIO FREQUENCY WIRELESS COMMUNICATIONS," *Int. J. Embed. Syst. Appl.*, vol. 2, no. 3, pp. 99–106, 2012.
- [16] S. Suropto and Iswanto, "DESAIN AND IMPLEMENTATION OF FM RADIO WAVES AS DISTANCE MEASURING AC VOLTAGE," *Int. J. Mob. Netw. Commun. Telemat.*,

vol. 2, no. 5, pp. 13–24, 2012.

- [17] A. Wahyudianto, Iswanto, and A. N. N. Chamim, “ALAT PENGONTROL LAMPU MENGGUNAKAN REMOTE TV UNIVERSAL,” in *SEMINAR NASIONAL ke 8 Tahun 2013*, 2013, pp. 112–116.
- [18] R. an T. A. Sadad, Iswanto, R. I. F. An, and T. A. S. Sadad, “Remote Measurement System Ground Shift with GSM,” *Int. J. Mob. Netw. Commun. Telemat. (IJMNCT)*, vol. 3, no. 3, pp. 65–72, 2013.
- [19] H. Muhammad and Iswanto, “EGT 10 Design and Application For Position,” *Int. J. Mob. Netw. Commun. Telemat. (IJMNCT)*, vol. 3, no. 3, pp. 1–8, 2013.
- [20] A. Mustaqim, A. Nuryanto, T. Mulyana, M. . A. Ramadhan, Muholidin, and Iswanto, “JUKNYI (TUNJUK BUNYI) SEBAGAI ALAT BANTU TUNA NETRA DALAM PEMILU,” in *SEMINAR NASIONAL ke 8 Tahun 2013: Rekayasa Teknologi Industri dan Informasi*, 2013, pp. 108–111.
- [21] I. Iswanto and R. D. Setiawan, “Power Saver with PIR Sensor,” *J. Control Instrum.*, vol. 4, no. 3, pp. 26–34, 2013.