



Selamat Datang

Di Dunia Pengendali Robotika



Perencanaan pembelajaran

- TUJUAN
 - Agar Mahasiswa dapat memahami tentang Penggunaan IC Mikrokontroler ATMEGA 16/32/8535 sebagai komponen pengendali elektronika atau robotika
 - Mahasiswa dapat membuat downloader mikrokontroler
 - Mahasiswa dapat menggunakan software simulator baik berupa proteus, Top View. Read51



Mikrokontroler

MATERI POKOK

Pertemuan Ke-	Pokok Bahasan
1	PENGANTAR MIKROKONTROLER
2	PERANGKAT KERAS MIKROKONTROLER
3	MEMORY DAN REGISTER MIKROKONTROLER
4	BAHASA PEMOGRAMAN
5	SET INSTRUKSI MIKROKONTROLER
6	TEKNIK PEMOGRAMAN MIKROKONTROLER
7	Review Materi – Quiz Essay
8	UJIAN TENGAH SEMESTER (UTS) – Pilihan Ganda (OCR)
9	APLIKASI MIKROKONTROLER DENGAN SIMULASI PROTEUS
10	APLIKASI MIKROKONTROLER DENGAN SIMULASI PROTEUS
11	PRESENTASI TUGAS KELOMPOK
12	PRESENTASI TUGAS KELOMPOK
13	PRESENTASI TUGAS KELOMPOK
14	PRESENTASI TUGAS KELOMPOK
15	Review Materi



Sumber Referensi :

1. Heri Andrianto, 2008. Pemograman Mikrokontroler AVR ATMEGA 16 menggunakan Bahasa C. Penerbit Informatika Bandung
2. Ardi Winoto. 2008 .Mikrokontroler AVR Atmega8/16/32/8535 dengan Bahasa C. Penerbit Informatika Bandung.
3. M. Ary Heryanto, ST, Ir Wisnu Adi P, 2008, Pemograman Bahasa C untuk Mikrokontroler ATMEGA 8535, Penerbit Andi.

CATATAN :

1. Mahasiswa wajib mempunyai buku referensi tersebut.
2. Mahasiswa diharapkan mencari referensi tambahan dari jurnal-jurnal dan wajib memiliki datasheet IC Mikrokontroler Atmega 8 dan Atmega 16 (dapat dicari di Internet)



RENCANA PEMBELAJARAN

- Pertemuan 1 s/d 6 disampaikan dengan Metode Ceramah, Metode Diskusi dan Latihan Soal.
- Pada Pertemuan 9 s/d 10 belajar membuat simulasi dengan software simulator,
- Pada pertemuan 11 s/d 15 dilakukan presentasi per kelompok yang sudah dibagi kelompoknya.
- Dosen menentukan mahasiswa yang akan presentasi dari kelompok tersebut.
- Kelompok lain yang tidak mempresentasikan makalahnya harus membuat resume dan kesimpulan dari kelompok lain yang sedang presentasi.



Penilaian Nilai Tugas

- Tugas 1 : Bobot Nilai 40
- Tugas 2 : Bobot Nilai 40
- Quiz UTS Essay : Bobot Nilai 20

Jadi Total Nilai TUGAS : 100

- Tugas 4 : merupakan nilai UAS dengan bobot nilai 40%

Dengan kriteria penilaian:

- Presentasi 25 %
- Penguasaan Materi 50 %
- Makalah 25%



DESKRIPSI SINGKAT TUGAS

Pembentukan Kelompok disesuaikan dengan jumlah mahasiswa di dalam kelas, misal jika dalam 1 kelas terdapat 20 mahasiswa berarti dibentuk 10 kelompok terdiri dari 2 mahasiswa.



TUGAS I

Dikerjakan oleh perorangan

1. Membuat aplikasi mikrokontroler dengan software simulator lihat di pert 9.
2. Aplikasi bebas tetapi tidak boleh sama dengan yang lain.
3. Di Print out Listing programnya serta dikumpulkan ke dosen.



TUGAS II

Dikerjakan Perkelompok

1. Membuat Downloader Atmega / AVR baik Model Port Pararel atau USB (Pilih salah Satu) wajib membuat.
2. Membuat Aplikasi Mikrokontroler :
 - Membuat Aplikasi Led dan Pus Button
 - Membuat Aplikasi Seven segmen dan Pus Button (tombol).
 - Membuat aplikasi Seven segmen dengan sensor Infra Red atau LDR



- Membuat aplikasi motor DC / Motor Stepper
- Membuat aplikasi sensor dengan buzzer
- DII

Tugas kelompok di presentasikan pada
Pert.11s/d 15

Dosen yang menentukan tugas kelompok
untuk dibagikan ke mahasiswa, tugas ini
wajib di kerjakan oleh mahasiswa



PERTEMUAN

PENGANTAR MIKROKONTROLER ATMEGA 16/32/8535



Mikrokontroler

AMIK

Bina Sarana Informatika

Kode Mata Kuliah	: 701
Nama Mata Kuliah	: Mikrokontroler
Beban Kredit	: 4 SKS
Semester	: 3 (tiga)
Fakultas/Jurusan	: Teknik Komputer



MATERI KULIAH

- Dasar mikrokontroler ATMEGA 16/32/8535 meliputi : Pengantar Mikrokontroler ATMEGA 16/32/8535, perangkat keras, bahasa pemrograman, set instruksi, teknik pemrograman, program aplikasi, timer/counter dan Simulator

Software Pendukung

- Software Proteus (Bahasa C), Software Top View (ASM)
- Software Prog ISP (Ver 1.68), MIDE 51
- Software AVR Studio 4.18, PonyProg 2.05/2.06/2.07
- AVR CodenVision, AVR Evaluation
- Software tersebut dapat di download di:

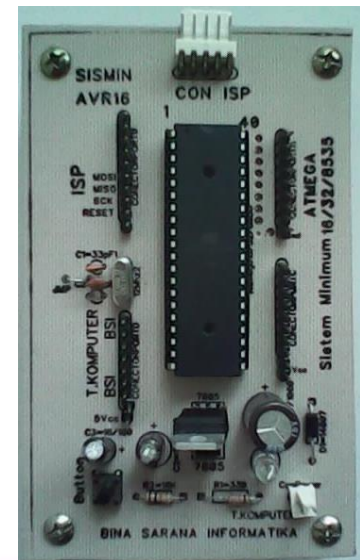
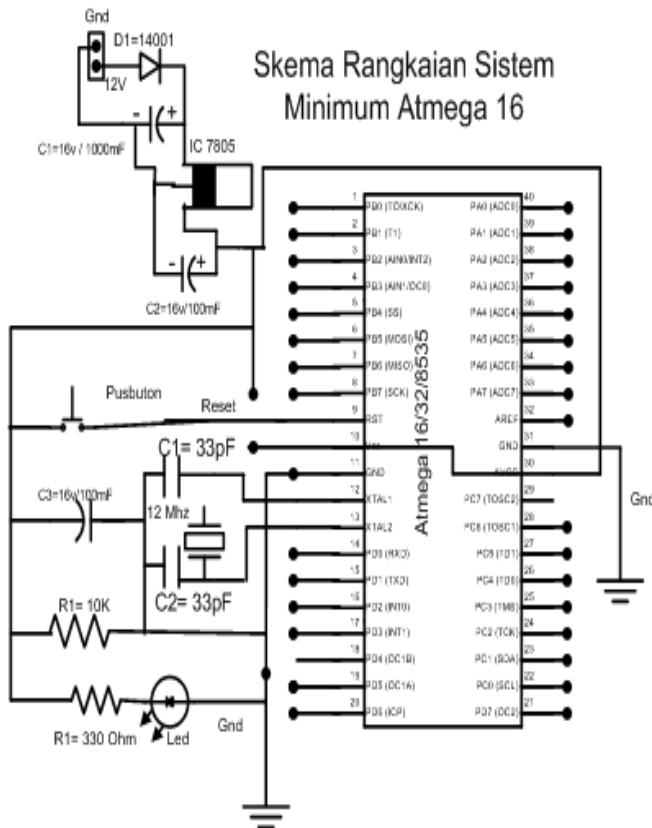
<http://www.kelas-mikrokontrol.com/download.html>, <http://www.atmel.com/avr>,
<http://www.hpinfotech.ro>, www.fischl.de/usbasp, <http://khazama.com>

Sistem Minimum ATmega

Rangkaian Sistem Minimum Mikrokontroler ATMEGA16/32/8535 memakai Software AVR Studio 4, PonyProg 2000, ISP Prog V.168

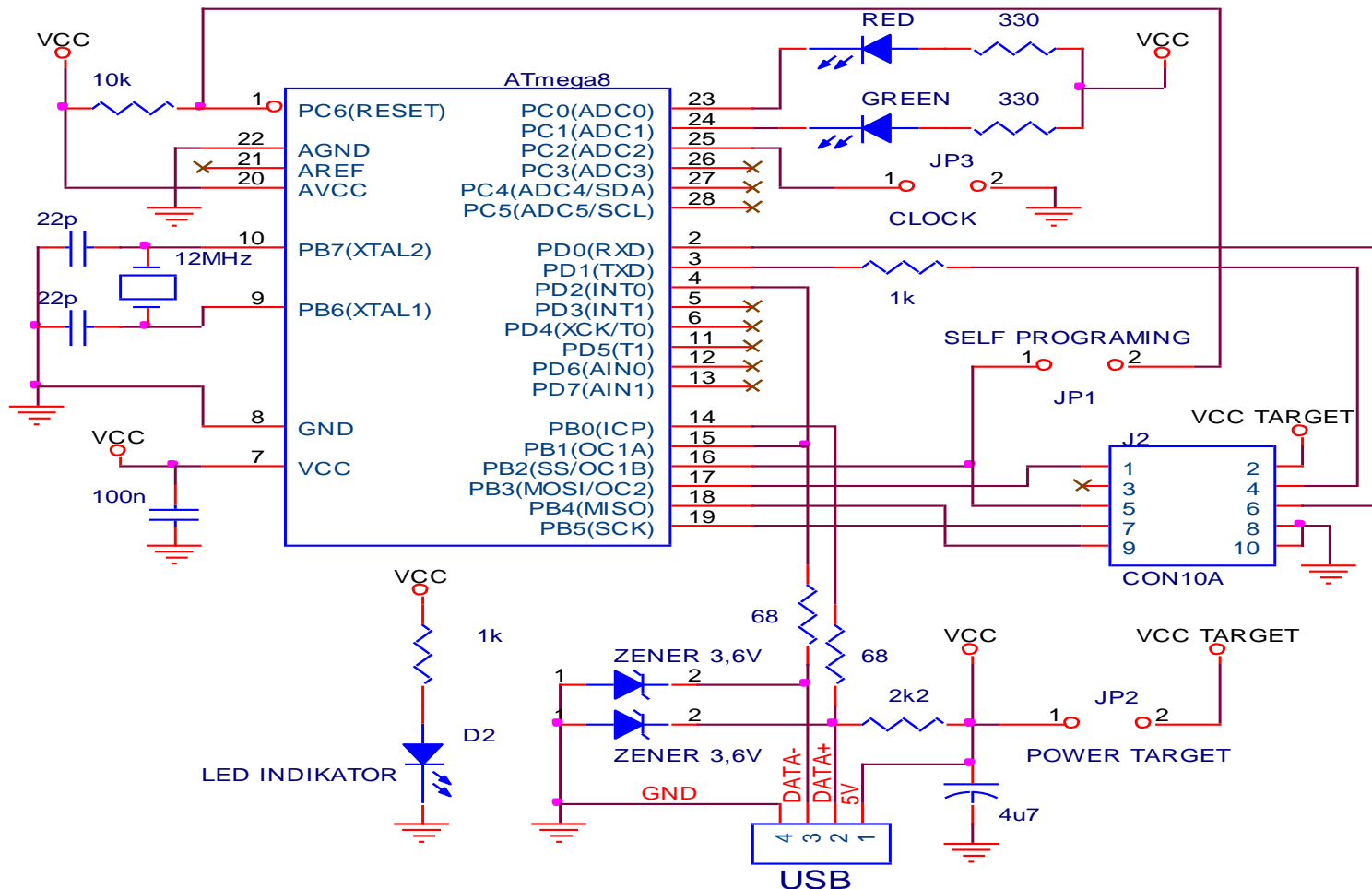
Daftar Komponen :

1. IC Mikrokontroler ATMEGA 16/32/5835
2. Soket IC 40 Pin
3. Resistor (R1) = 10K
4. Capacitor (C1&C2= 33pF atau 22pF
5. Xtal = 12 Mhz
6. Capacitor (C3) = 16v/100mF
7. PusButton
8. IC Regulator 7805
9. Resistor (R=330)
10. Dioda 14007
11. Led
12. Conector 8 pin dan 2 pin
13. PCB Fiber





Rangkaian Downloader Driver USB Mikrokontroler ATMEGA8 16PU



Dapat di download disini :

<http://www.fischl.de/usbasip/>

<http://elektro-kontrol.blogspot.com/p/avr-usb-downloader.html>

PROGRAM DIPLOMA III BINA SARANA INFORMATIKA



Daftar Komponen Mikrokontroler Atmega8 Versi USB :

1. Capacitor (C1, C2) = 22pF warna orange
2. Capacitor Elco (C3) = 10v/4,7 μ F
3. Capacitor (C4) = 100nF warna kuning
4. Dioda Zener (D1, D3) = 3V6
5. IC1 = ATMEGA8 16 PU/PI
6. LED1 = LED Red
7. LED2 = LED Green
8. Kristal (Q1) = 12Mhz
9. Resistor $\frac{1}{4}$ Watt (R5, R6) = 68 Ohm
10. Resistor $\frac{1}{4}$ Watt (R8) = 2,2 kOhm
11. Resistor $\frac{1}{4}$ Watt (R1, R2) = 330 Ohm
12. Resistor $\frac{1}{4}$ Watt (R7) = 10 kOhm
13. Resistor $\frac{1}{4}$ Watt (R3) = 1 kOhm
14. Conector (X1) = USB





Tahapan Pembuatan Tugas

1. Gambar Skema Rangkaian Downloader Port Pararel atau USB dengan Program Aplikasi Yang di kuasai Seperti:
 - ~ Diptrace
 - ~ Eagle Skematik
 - ~ Protelpilih salah satu saja
2. Buat PCB dengan Cara:
 - ~ Manual
 - ~ PCB Disaigner, Diptrace
3. Buat Listing Program Dengan software Win AVR, AVR Studio, CodeVision AVR, MIDE 51, (Pilih salah satu)
4. Isi Program IC Mikrokontroler Dengan ISP Prog. 1.68
5. Buat Slide Presentasi

<http://www.kelas-mikrokontrol.com>
<http://www.fischl.de/usbaspl/>



Pendahuluan

Pengertian Rangkaian Analog Rangkaian Digital

Rangkaian Analog

Jenis rangkaian elektronika yang dapat memproses signal yang bersifat kontinyu. Perubahan signal dalam rangkaian Analog ini adalah sedikit demi sedikit meliputi semua titik pada amplitudo signal maximum dan minimumnya

Rangkaian Digital

Jenis rangkaian elektronika yang dapat memproses signal yang dua kondisi yaitu kondisi tinggi atau rendah. Perubahan signal dalam rangkaian digital bersifat diskrit



Rangkaian digital dan mikroprosesor

Persamaan

Rangkaian digital/diskrit dan mikroprosesor merupakan perangkat keras yang mengolah sinyal dua (2) keadaan yaitu sinyal kondisi tinggi “1” dan rendah “0”.

Perbedaan

Rangkaian digital/diskrit tidak membutuhkan perangkat lunak untuk dapat beroperasi.

Sedangkan Mikroprosesor dapat beroperasi jika ada perangkat lunak melalui instruksi program yang di buat



Pengertian Mikroprosesor, Mikrokomputer Mikrokontroler

Sering terjadi kesalahan dalam memahami definisi mikroprosesor, mikrokomputer dan mikrokontroler. Ketiga istilah tersebut berbeda dan penting untuk mengerti tentang pengertian yang sesungguhnya.

Mikroprosesor

Mikroprosesor adalah CPU atau *Central Processing Unit* yang terdapat dalam satu chip. CPU di rancang dengan (MSI/LSI) chips yang terintegrasi dengan jumlah medium ataupun besar



Mikrokontroler yang di produksi Intel Microprosesor didalam sebuah Chip atau IC terintegrasi:

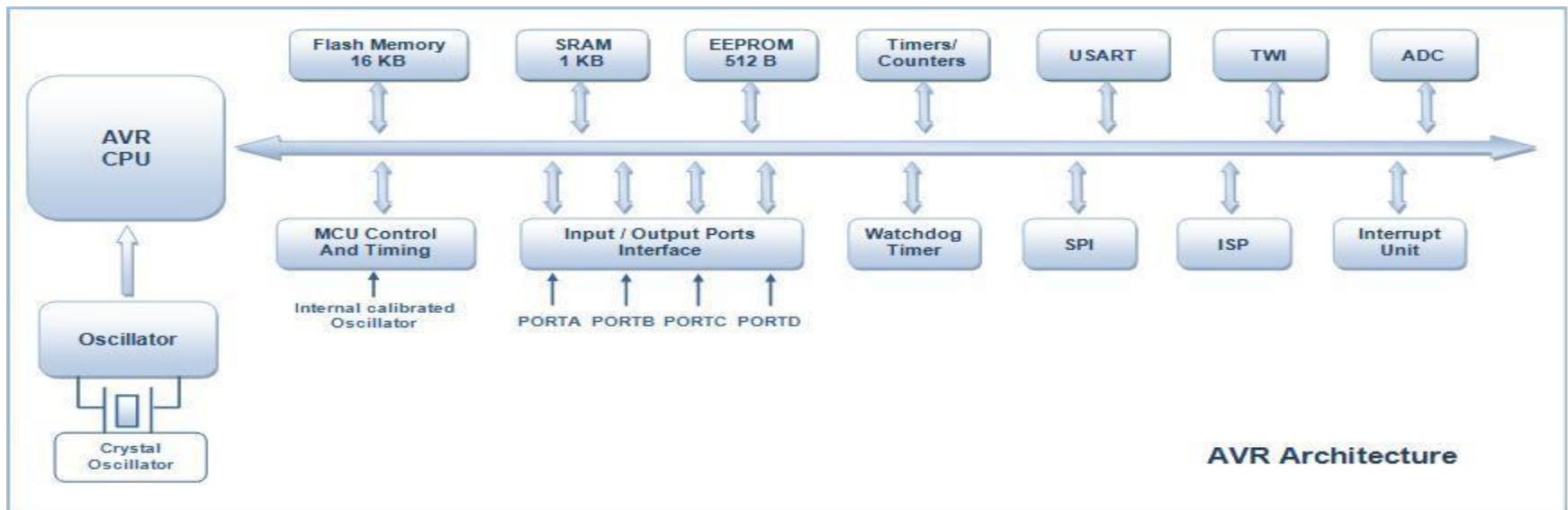
Aritmetic logic unit (ALU)

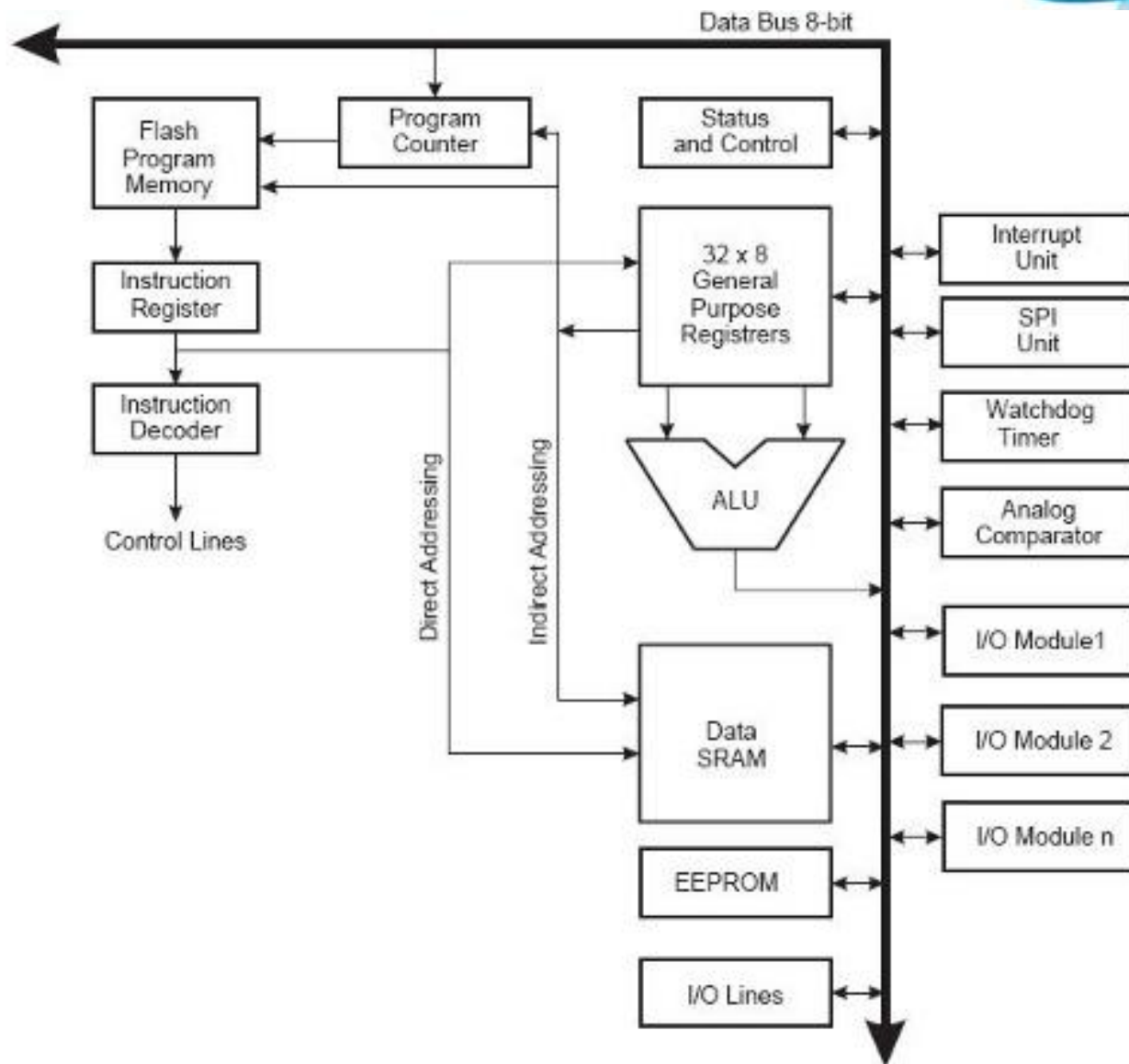
Instructions decoder

Registers

Bus control circuits

DII





Arsitektur Mikrokontroler ATMEGA



FUNGSI BLOCK DALAM MIKROPROSESOR

CONTROL UNIT

Tugas dari control unit adalah :

1. Mengatur dan mengendalikan alat-alat input dan output.
2. Mengambil instruksi-instruksi dari main memory.
3. Mengambil data dari main memory kalau diperlukan oleh proses.
4. Mengirim instruksi ke ALU bila ada perhitungan aritmatika atau perbandingan logika, serta mengawasi kerja ALU.
5. Menyimpan hasil proses ke main memory.



ARITHMETIC AND LOGIC UNIT (ALU)

- Tugas utama dari ALU adalah melakukan semua perhitungan aritmatika atau matematika yang terjadi sesuai dengan instruksi program.
- Selain itu ALU juga melakukan keputusan dari operasi logika sesuai dengan instruksi program.



REGISTER

Register merupakan simpanan kecil yang mempunyai kecepatan tinggi, lebih cepat sekitar 5 sampai 10 kali dibandingkan dengan kecepatan perekaman atau pengambilan data di main memory.

Register digunakan untuk menyimpan instruksi dan data yang sedang diproses oleh CPU, sedang instruksi-instruksi dan data lainnya yang menunggu giliran untuk diproses masih disimpan di main memory.



Microkomputer

Jika sebuah microprosesor dengan peralatan peripheral pendukungnya seperti circuit board, I/O peripheral, Memory (program ataupun data) di letakkan bersama dalam suatu tempat seperti komputer kecil yang di khusus di gunakan untuk akuisisi dan aplikasi process kontrol, maka yang seperti itu disebut sebagai mikrokomputer

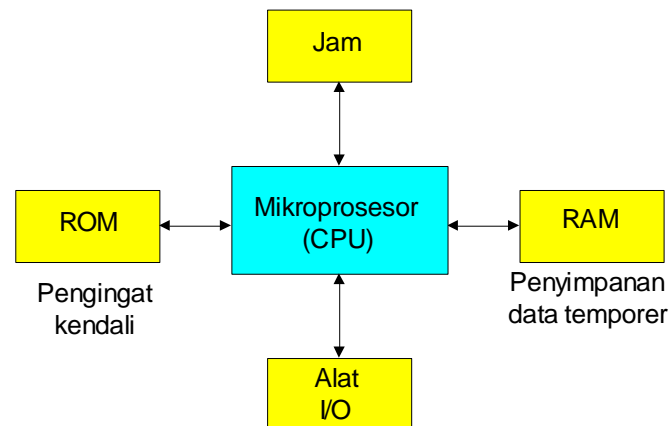


Diagram block mikrokomputer



Mikrokontroler

DEFINISI :suatu IC dengan kepadatan yang sangat tinggi, dimana semua bagian yang diperlukan untuk suatu kontroler sudah dikemas dalam satu keping,

Didalam chip atau IC mikrokontroler terintegrasi:

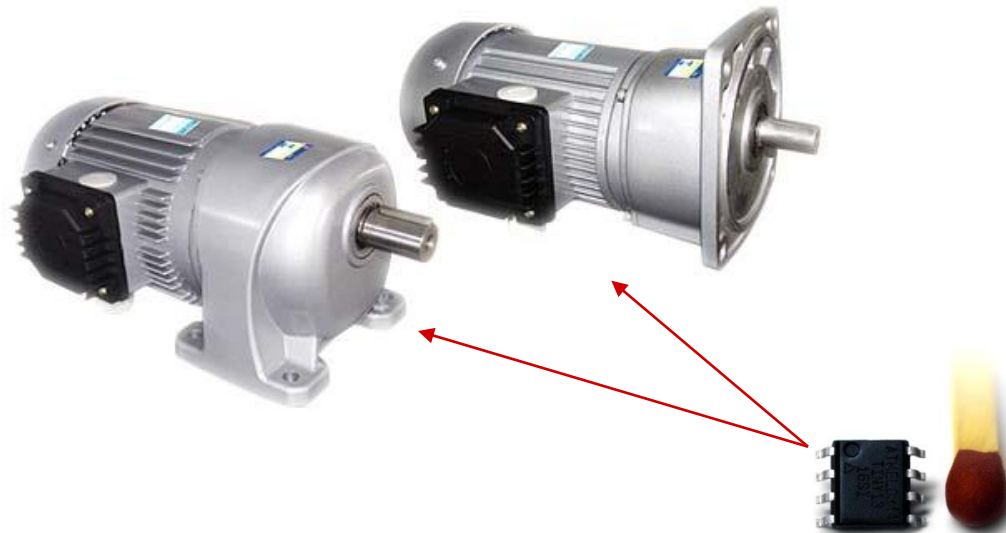
1. CPU
2. Peripheral pendukung berupa:
 - RAM
 - ROM
 - I/O ports
 - Timers
 - Serial port
 - Usb, dll



APA ITU MICROCONTROLLER?

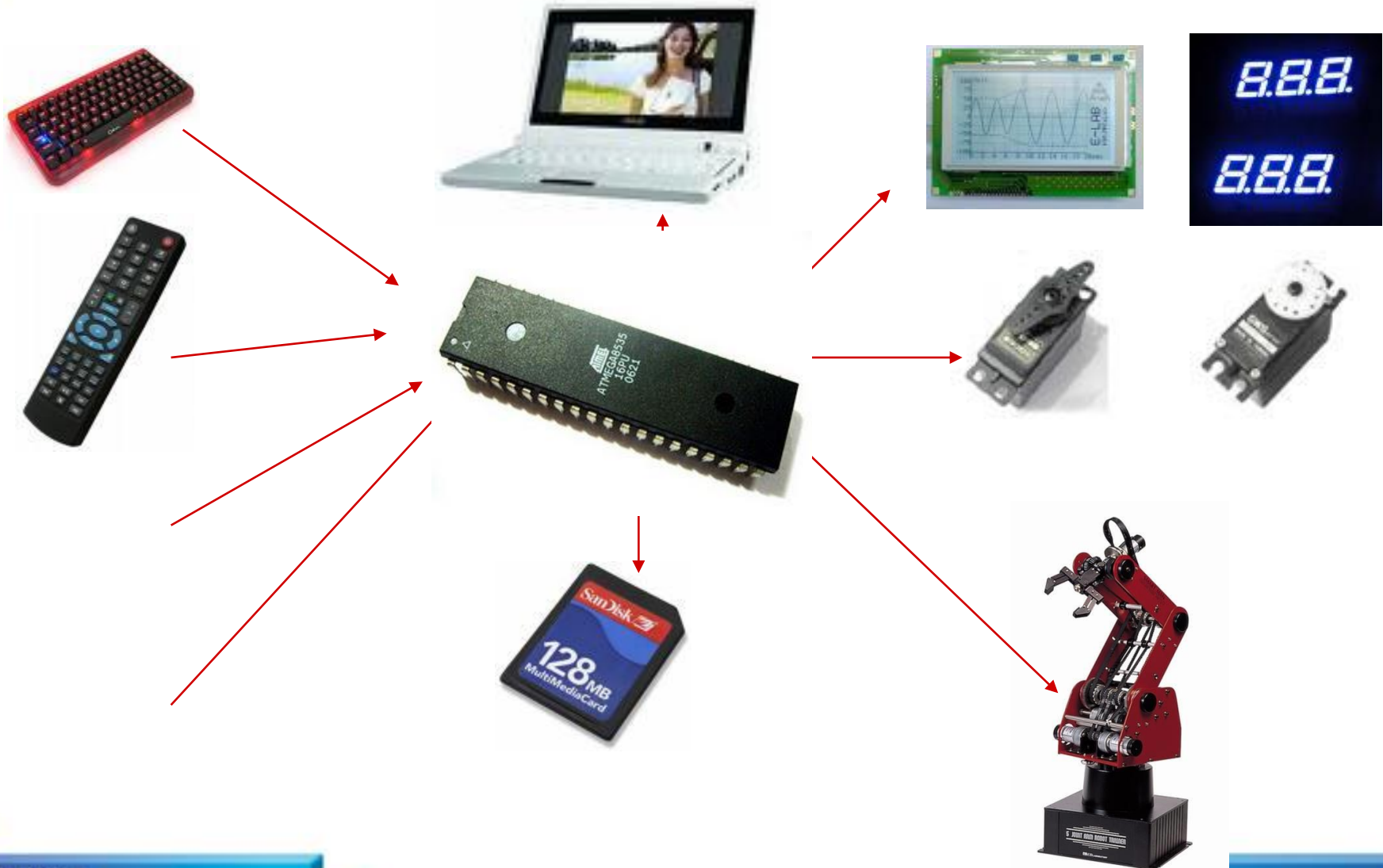
MICRO
KECIL

CONTROLLER
PENGATUR

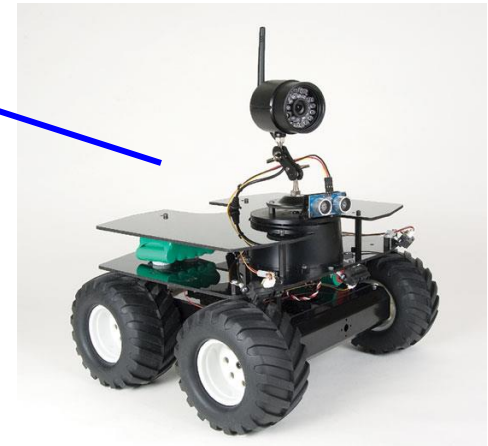
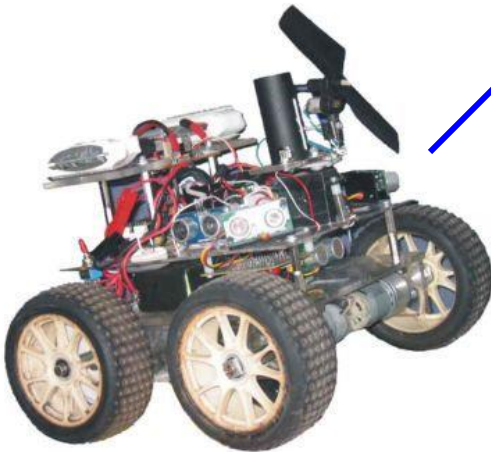
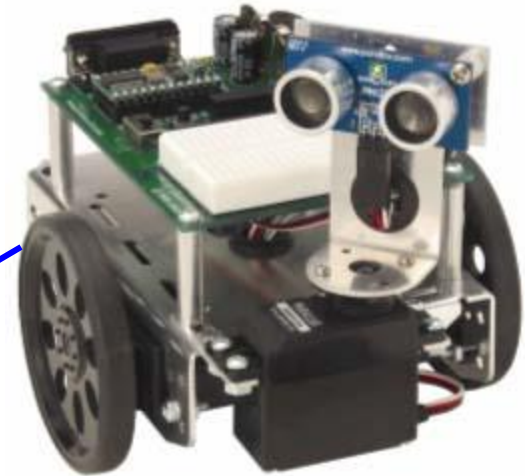
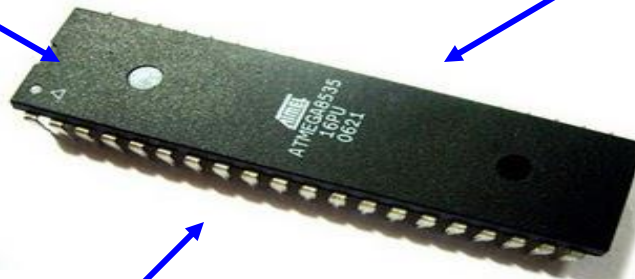
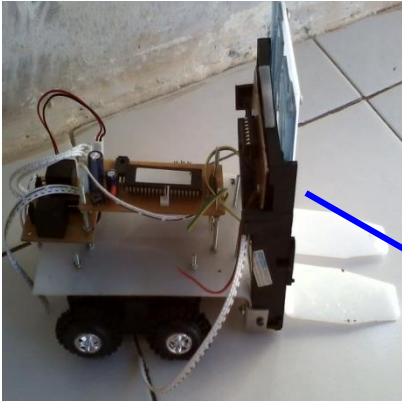




APA YANG BISA DI LAKUKAN?



Robotika





Texas Instruments adalah perusahaan semiconductor pertama yang membuat mikrokontroler dengan TMS 1000 series.

Microkontroler di gunakan untuk aplikasi microwave oven controller, industrial timer dan calculator

Jenisnya Mikrokontroler:

- ~ Motorola dengan seri 68xx
- ~ keluarga MCS51 yang diproduksi Atmel
- ~ Keluarga AVR
- ~ PIC dari Microchip
- ~ Zilog.



PERBANDINGAN MICROCONTROLLER

FITURE	MSC-51 89C2051	MSC-51 89S51	AVR ATMEGA 8515	AVR ATMEGA 8	AVR ATMEGA 32
PIN	20PIN	40PIN	40PIN	28PIN	40PIN
RAM	256byte	256 byte	512 byte Up to 64K	1 Kbyte	1 Kbyte
FLASH	4 Kbyte	4 Kbyte	8 Kbyte	8 Kbyte	32 Kbyte
EEPROM	NO	NO	512byte	1Kbyte	2Kbyte
ADC	NO	NO	NO	8 CH	8CH
BROWN DETECTOR	NO	NO	YES	YES	YES
PWM	NO	NO	YES	YES	YES
OSCILATOR	EXTERNAL	EXTERNAL	EXTERNAL/I NTERNAL	EXTERNAL/IN TERNAL	EXTERNAL/IN TERNAL
POWER ON RESET	EXTERNAL	EXTERNAL	EXTERNAL/I NTERNAL	EXTERNAL/IN TERNAL	EXTERNAL/IN TERNAL
HARGA (Rp)	15.000	20.000	25.000	15.000	30.000



Defenisi Lain Mikrokontroler

Mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus

Dengan mikrokontroler ini maka :

Sistem elektronik akan menjadi lebih ringkas, rancang bangun sistem elektronik akan lebih cepat karena sebagian besar dari sistem adalah perangkat lunak yang mudah dimodifikasi pencarian gangguan lebih mudah ditelusuri karena sistemnya yang kompak



Di dalam mikrokontroler terdapat mikroprosesor, BUS, clock/osilator, RAM, EPROM, timer dan port I/O seperti halnya mikrokomputer.

Yang membedakan adalah fungsi mikroprosesor pada mikrokontroler sangat sederhana dan terbatas sebagai pengontrol. Selain itu mikrokontroler bekerjanya menempel (***embedded***) pada suatu alat.

Seperti halnya mikrokomputer, mikrokontroler untuk beroperasi memerlukan program. Bahasa pemrograman yang dipakai adalah bahasa rakitan (*assembly*) yang sintaksis dan kompilasinya tergantung pabrik pembuat,



Jenis-jenis mikrokontroler

- Secara teknis hanya ada 2 jenis mikrokontroler yaitu: RISC dan CISC dan masing-masing mempunyai turunan/keluarga sendiri-sendiri.
- RISC (*Reduced Instruction Set Computer*) :
instruksi terbatas tapi memiliki fasilitas yang lebih banyak

CISC (*Complex Instruction Set Computer*) :
instruksi bisa dikatakan lebih lengkap tapi dengan fasilitas yang terbatas.



Memilih Mikrokontroler

Terdapat beberapa hal yang perlu dipertimbangkan dalam menentukan jenis mana yang akan dipergunakan dalam disain seperti:

1. Ketersediaan dan harga dari suatu development tools (*Programmer, Emulator dan Simulator*)
2. Ketersediaan dokumentasi (Ref. Manual, Application notes, dan buku lainnya).
3. Ketersediaan tempat bertanya.
4. Ketersediaan komponen OTP, Mask, Programmable.



Perbedaan mikroprosesor dengan mikrokontroler

Diantara perbedaannya adalah :

- ~ Dari segi arsitekturnya mikroprosesor merupakan single chip CPU, didalam chip mikrokontroler selain terdapat CPU juga terintegrasi ROM, RAM, Eprom, paraler interface, interupt controler, timer, dll (tergantung dari jenis mikrokontroler)
- ~ Dari segi aplikasinya, mikroprosesor hanya berfungsi sebagai CPU yang menjadi otak komputer, sedangkan mikrokontroler yang ukurannya relatif kecil umumnya diberikan untuk tugas-tugas yang berorientasi kontrol pada rangkaian yang membutuhkan jumlah komponen minimum dan konsumsi biaya rendah (low cost).



Sistem Minimum Mikrokontroler ATMEGA

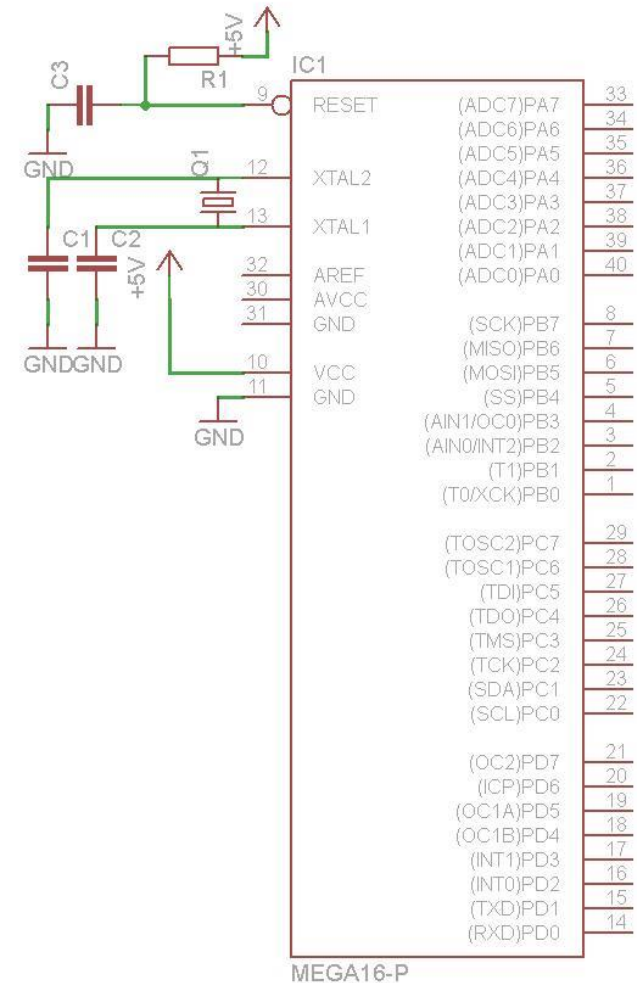
Untuk membuat sebuah sistem minimum dengan mikrokontroler ATMEGA dibutuhkan beberapa komponen yaitu :

IC mikrokontroler

Kristal (XTAL) 12 MHz ,

kapasitor 33pF & Resistor 10 Kohm

Sebagai sumber energi adalah sumber tegangan 5V DC

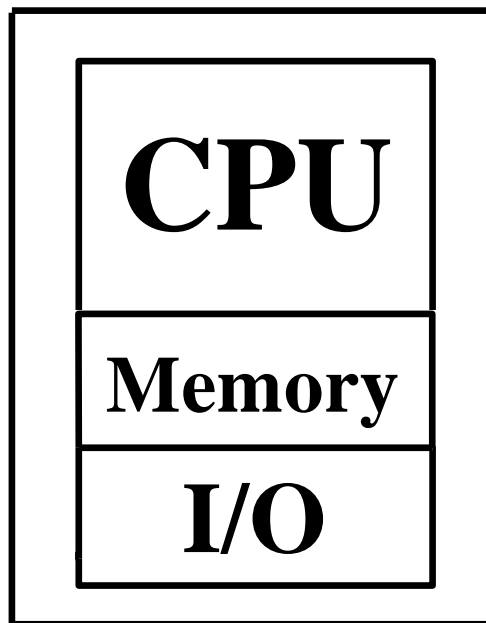


Gambar rangkaian Sistem Minimum

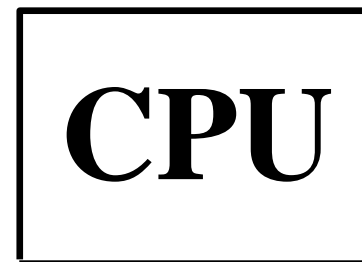


Mikrokontroler Vs Mikroprosesor

Mikrokontroler



Mikroprosesor

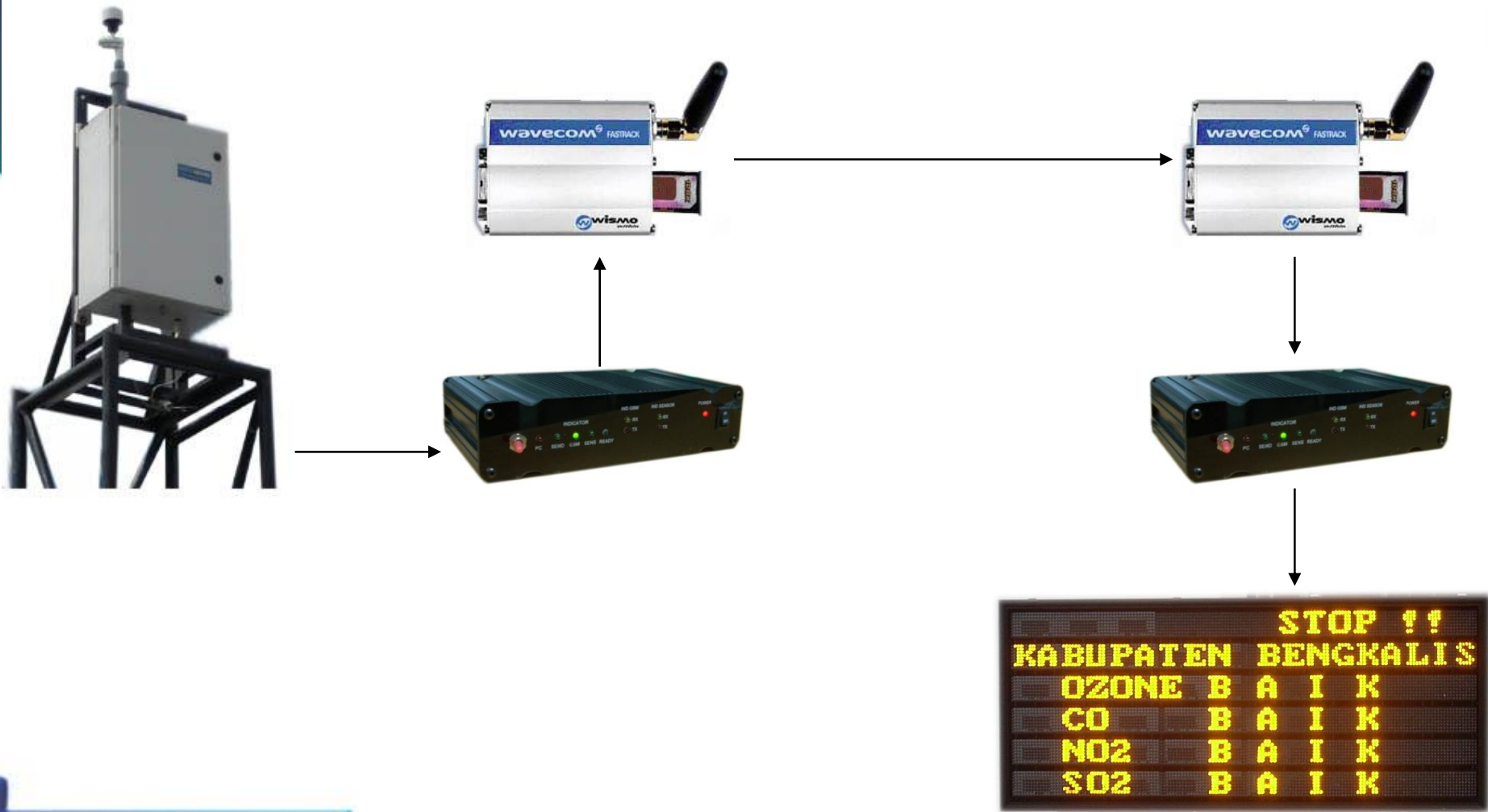




Aplikasi Mikrokontroler Industri



Aplikasi Mikrokontroler Dalam mengukur Kondisi Udara





Sampai Jumpa



PERTEMUAN

2

PERANGKAT KERAS MIKROKONTROLER ATMEGA 16/32/8535



Pendahuluan

Pada dasarnya mikrokontroler bukanlah ilmu pengetahuan yang baru, tetapi adalah hasil pengembangan dalam teknologi elektronika.

Jika dasar pengetahuan seperti :

Elektronika dasar

Rangkaian listrik

Rangkaian digital

Mikroprosesor

Dan bahasa rakitan

Semua bidang ilmu diatas akan sangat membantu dalam mempelajari mikrokontroler



Belajar Mikrokontroler Yang Efektif

Agar bisa memahami dan menguasai mikrokontroler dalam waktu yang relatif singkat maka diperlukan :

1. Pembuatan Downloadernya sebagai pengisi IC mikrokontroler (Download Program Ke IC)
2. Berlatih membuat program Aplikasi dengan software simulator (Cth. Proteus)
3. Memahami set instruksi perintah yang dapat di gunakan pada mikrokontroler.
4. Membuat eksperimen baik Led, Seven Segmen dll

Untuk mencapai hasil yang maksimal maka di butuhkan perangkat pengembang atau *development tools*



Perangkat pengembang mikrokontroler sbb:

Compiler (Penterjemah):

Berupa perangkat lunak karena mikrokontroler beroperasi dengan bahasa mesin, sedangkan manusia terbiasa memakai bahasa yang lebih tinggi seperti Bahasa Assembler, C/C++, Basic dll.

Atau manusia biasa sulit memahami bahasa mesin, maka dengan bantuan Compiler (Penterjemah) bahasa pemrograman diterjemahkan menjadi bahasa mesin.

Simulator:

Berupa perangkat lunak yang mensimulasikan kerja program ke dalam mikrokontroler

Dengan bantuan simulator seorang programmer dapat melihat hasil program yang di buat sebelum di tulis / di isi ke dalam IC mikrokontroler.



Emulator:

Berupa perangkat keras dan perangkat lunak
Suatu alat yang di hubungkan dengan PC.

Program yang sudah selesai di buat di tulis (Download)
ke emulator kemudian emulator akan bekerja sendiri
walaupun koneksi dilepas dari PC.

Jika terjadi kesalahan atau program yang di buat belum
sesuai dengan keinginan maka setelah program diperbaiki
dapat di tulis (Download) ulang ke emulator

catatan:

Emulator sering di sebut dengan target.



Programmer:

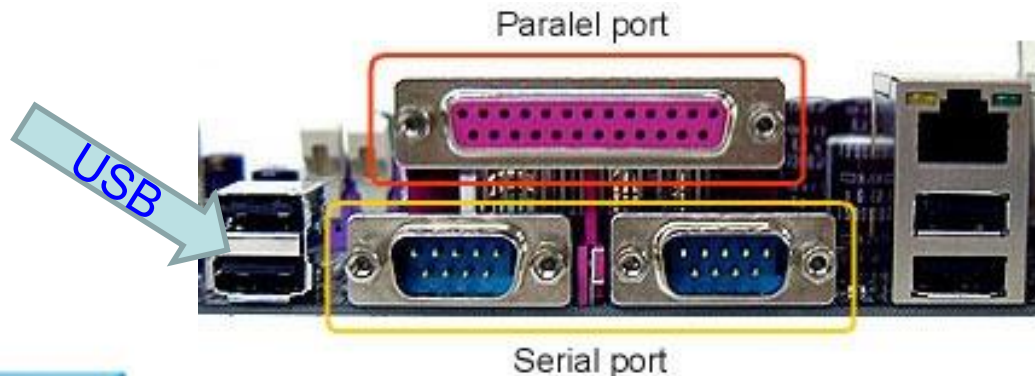
Berupa perangkat yang berfungsi untuk mengisi program yang dibuat kedalam mikrokontroler atau bisa di sebut sebagai interface antara mikrokontroler dengan Personal Computer atau Laptop.

Mikrokontroler Atmega 16/32/8535 dapat terhubung ke PC melalui :

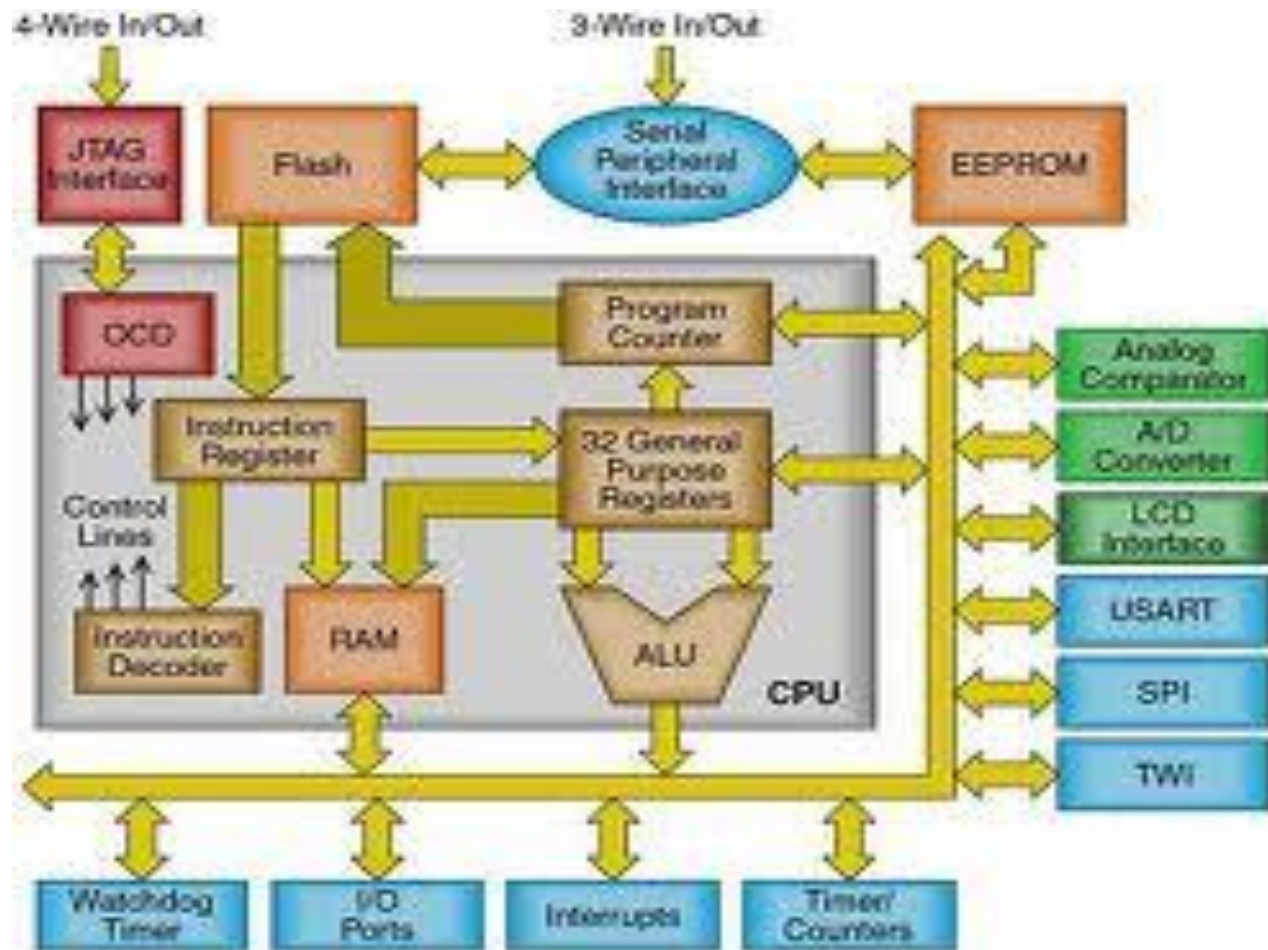
Port Serial = DB 9

Port Paralel = DB 25 (LPT)

atau dengan Universal Serial BUS (USB)

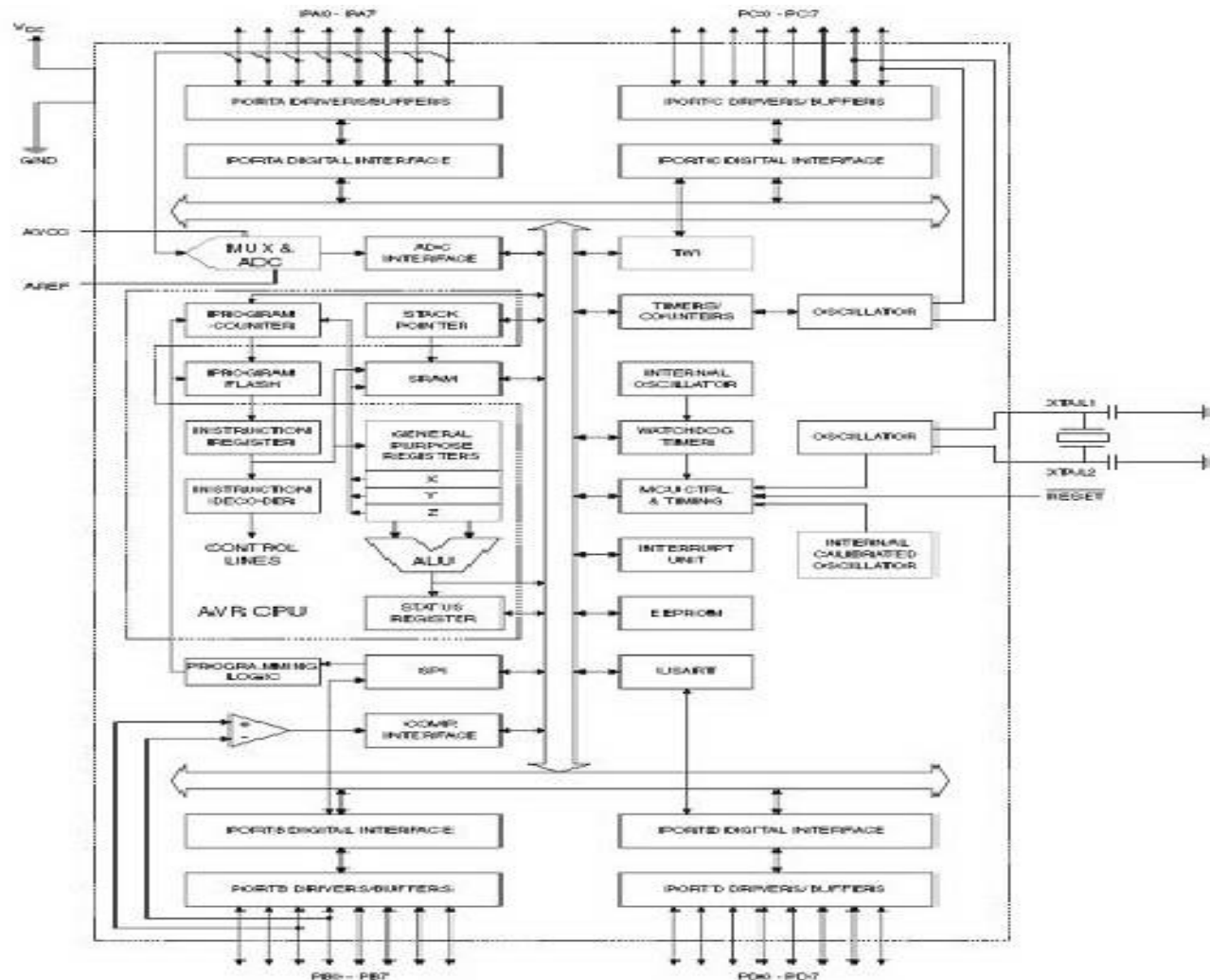


Arsitektur mikrokontroler Atmega 16/32/8535



Gambar Diagram Blok Mikrokontroler ATmega

Arsitektur mikrokontroler Atmega 16/32/8535



Gambar Arsitektur Mikrokontroler Atmega 16/32/8535



Fungsi Block mikrokontroler

CPU

- CPU singkatan dari *Central Prosesing Unit*

CPU ialah bagian yang paling penting dari suatu Mikrokontroler berfungsi untuk melakukan pemrosesan data.

Memory ATmega

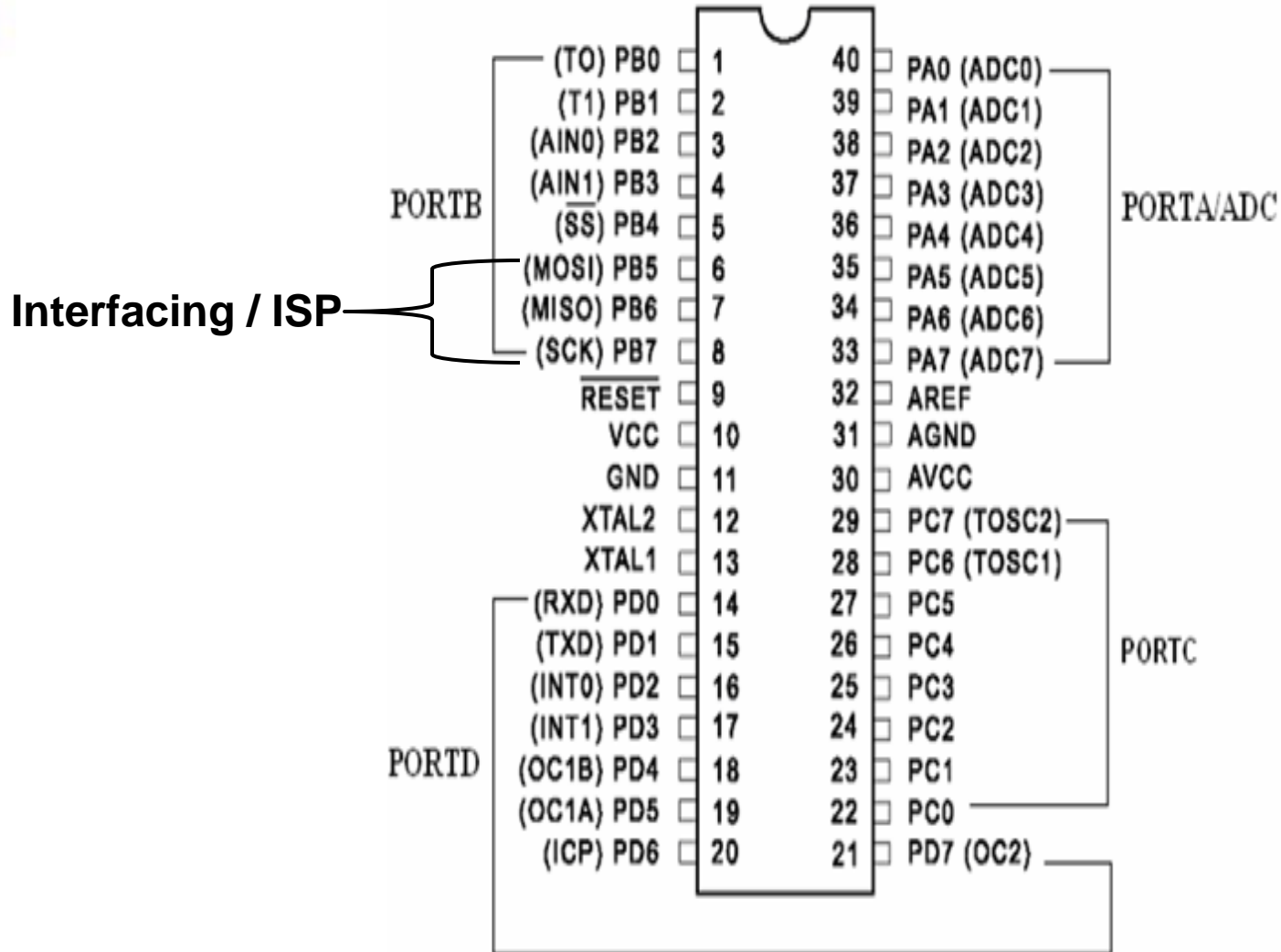
Memory Mikrokontroler ATmega dapat di akses dengan alamat 8 Bit . Pada Mikrokontroler Program kontrol disimpan di dalam ROM



Konfigurasi Pin Mikrokontroler ATmega

Walaupun IC dikemas dalam bentuk yang berbeda, tetapi fungsi kaki IC mempunyai persamaan.

Konfigurasi Pin mikrokontroler ATmega merupakan penjelasan fungsi masing-masing pin/kaki dari IC mikrokontroler



Nama Pin-pin ATmega



Adapun fungsi Pin pada mikrokontroler ATmega:

Port (TA 0-7)

Terdapat pada kaki 33 s/d 40 berfungsi sebagai Port I/O dua arah dan berfungsi khusus sebagai pin masukan ADC (merupakan Port Khusus Converter Analog ke Digital)

Port Pin	Fungsi
PA.0	PA0 ADC0 (ADC input channel 0)
PA.1	PA1 ADC1 (ADC input channel 1)
PA.2	PA2 ADC2 (ADC input channel 2)
PA.3	PA3 ADC3 (ADC input channel 3)
PA.4	PA4 ADC4 (ADC input channel 4)
PA.5	PA5 ADC5 (ADC input channel 5)
PA.6	PA6 ADC6 (ADC input channel 6)
PA.7	PA7 ADC7 (ADC input channel 7)



Port (TB 0-7)

Terdapat pada kaki 1 s/d 8 terdiri atas 8 bit, merupakan pin Input/Output dua arah dan berfungsi khusus sebagai pin Timer/Counter, komparator Analog dan SPI.

Port Pin	Fungsi
PB.0	T0 T1 (Timer/Counter 0 External Counter Input) XCK (USART External Clock Input/Output)
PB.1	T1 (Timer/Counter1 External Counter Input)
PB.2	AIN0 (Analog Comparator Positive Input / INT2 (External Interrupt 2 Input)
PB.3	AIN1 (Analog Comparator Negative Input / OCO (Timer/Counter Output Compare)
PB.4	SS (SPI Slave Select Input)
PB.5	MOSI (SPI Bus Master Input/Slave Output/sebagai interface ke PC)
PB.6	MISO (SPI Bus Master Input/Slave Output/ sebagai Interface ke PC)
PB.7	SCK (SPI Bus Serial Clock)



Port (TC 0-7)

Terdapat pada kaki 22 s/d 29 selain berfungsi sebagai Input/Output dua arah juga mempunyai fungsi khusus

Port Pin	Fungsi
PC.0	SCL (Two-Wire Serial Bus Clock Line)
PC.1	SDA (Two – Wire Serial Bus Data Input/Output Line)
PC.2	TCK (JTAG Test Clock)
PC.3	TMS (JTAG Test Mode Select)
PC.4	TDO (JTAG Test Data Out)
PC.5	TD1 (JTAG Test Data In)
PC.6	TOSC1 (Timer Oscillator Pin1)
PC.7	TOSC2 (Timer Oscillator Pin2)



Port (TD 0-7)

Terdapat pada kaki 14 s/d 21 berfungsi sebagai Port Input/Output dua arah juga, mempunyai fungsi khusus pada pin OC1A dan OC1B berfungsi sebagai PWM.

Port Pin	Fungsi
PD.0	RDX (<i>serial input port</i>)
PD.1	TDX (<i>serial output port</i>)
PD.2	_INT0 (<i>external interrupt 0</i>)
PD.3	_INT1 (<i>external interrupt 1</i>)
PD.4	OC1B (Timer/Counter1 output membandingkan output B)
PD.5	OC1A (Timer/Counter1 output membandingkan output A)
PD.6	_ICPI(Timer/Counter1 masukan menangkap Pin)
PD.7	_OC2 (timer / counter2 membandingkan mencocokkan output untuk PWM)



AREF (Kaki Pin = 32)

AREF merupakan pin masukan tegangan referensi converter Analog ke Digital (ADC).

AVCC (Kaki Pin = 30)

Merupakan pin masukan tegangan untuk Analog ke Digital (ADC).

VCC (Kaki Pin = 10)

Merupakan pin yang berfungsi sebagai masukan catu daya / tegangan ke mikrokontroler.



GND (Kaki Pin = 11 dan 31)

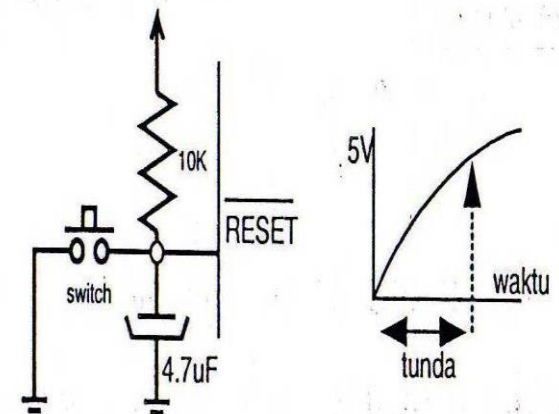
Merupakan kaki pin yang berfungsi sebagai Ground/Negatif ke mikrokontroler

GND (Kaki Pin = 11 dan 31)

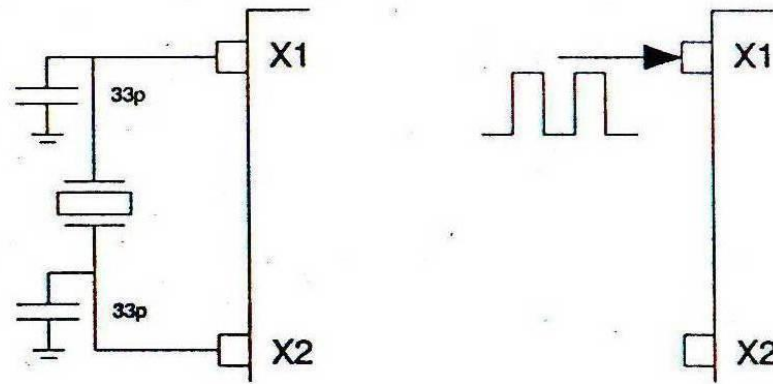
Merupakan kaki pin yang berfungsi sebagai Ground/Negatif ke Mikrokontroler

RST (Reset Kaki Pin = 9)

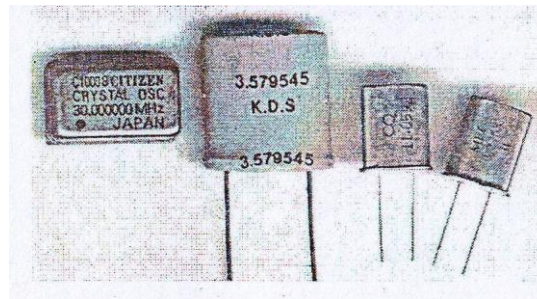
Terdapat pada kaki 9, dengan cara memberikan logic tinggi selama ± 2 siklus maka system dan memory internal pada mikrokontroler akan direset



Gambar rangkaian Reset



Output dari rangkaian osilator berbentuk pulsa dengan frekuensi 12-16 MHz.



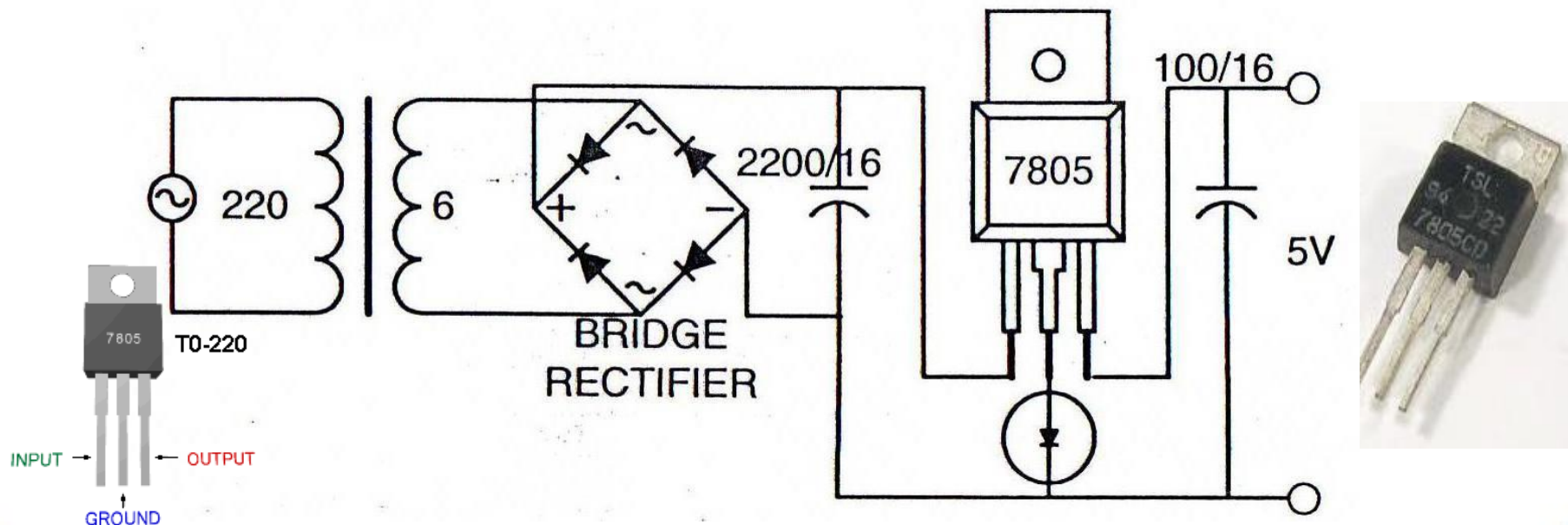
Gambar XTAL



Koneksi Power

Terdapat pada kaki 10, kaki 30 untuk tegangan positif dan kaki 11, kaki 31 untuk Ground (V_{ss}).

Sebagai sumber tegangan di gunakan power supply dengan keluaran 5 Volt DC.



Gambar Power Supply



Sampai Jumpa



PERTEMUAN

3

MEMORY, REGISTER, TIMER & COUNTER MIKROKONTROLER ATMEGA 16/32/8535

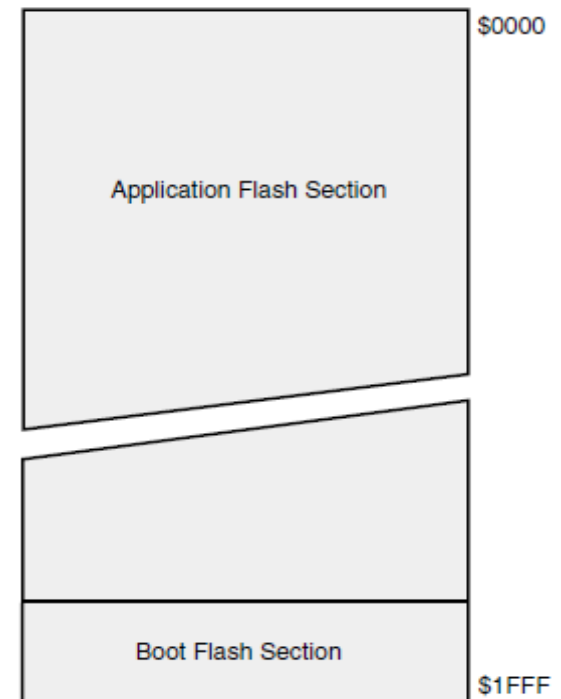


Memory

Program Memory, Data Memory dan EEPROM

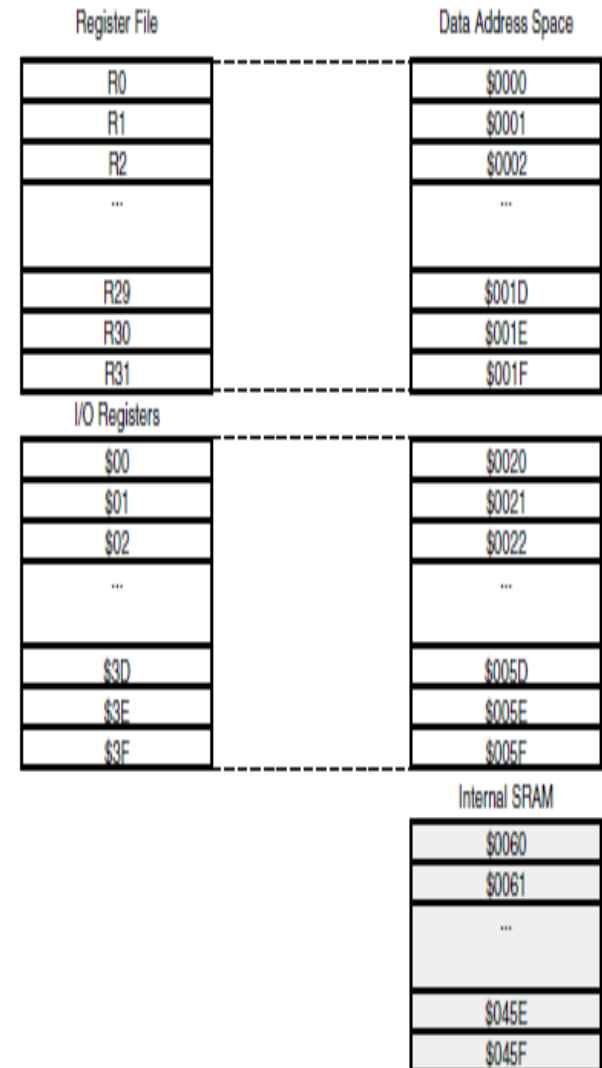
Memori yang terdapat pada Mikrokontroler dipisahkan menjadi 3 bagian yaitu *program memory* (memori program), *data memory* (memori data) dan Memori Data EEPROM yaitu :

1. Memori program berfungsi untuk menyimpan instruksi-intruksi program yang sifatnya hanya di baca/Read Only (ROM) dengan alamat \$0000 hingga \$1FFF.



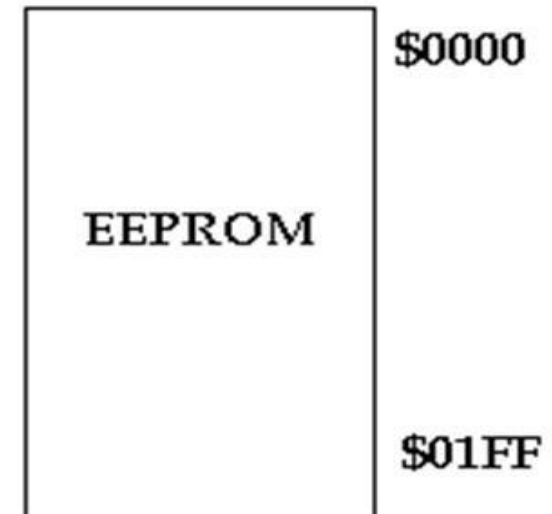


2. Memori Data berfungsi untuk keperluan program, memori data terbagi 3 bagian yaitu : 32 buah register umum, 64 buah register I/O dan 1 Kbyte SRAM internal. GPR(General Purpose Register Menepati alamat data yaitu : \$0000 s/d \$001F, Memori I/O menempati 64 alamat \$20 hingga \$5F. Memori I/O berkapasitas 1024 Byte merupakan Special Fungsi Register yang Khusus (SFR) dengan Alamat \$60 hingga \$45F.





3. Memori Data EEPROM 8 Bit (*Electrically Erasable Programmable read only Memory* (EEPROM): merupakan memori nonvolatile (tidak hilang ketika catu daya mati) yang menyerupai SRAM dalam fleksibilitas pemrogramannya. Memori jenis ini pada umumnya digunakan untuk menyimpan data–data permanen yang akan digunakan dalam program, jumlah EEPROM yang tersedia adalah 512 Byte dengan Alamat \$000 hingga \$1FF.





Register

General Purpose Register (GPR)

Seluruh instruksi operasi register dalam AVR memiliki akses langsung ke semua register. Kecuali untuk lima instruksi aritmatika-logika yang mengoperasikan register dengan konstanta (SBCI, SUBI, CPI, ANDI, dan ORI) dan LDI yang mengoperasikan pemuatan data konstan imediat. Instruksi-instruksi tersebut dioperasikan hanya pada separuh lokasi register terakhir GPR (R16 sampai R31). Instruksi untuk operasi umum seperti SBC, SUB, CP, AND, OR dan operasi lainnya yang mengoperasikan dua register atau satu register dapat dilakukan akses terhadap seluruh register.

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



Status Register (SREG)

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

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



SP (Stack Pointer)

Stack utamanya digunakan untuk menyimpan data sementara, untuk menyimpan variabel local dan untuk return address setelah interrupt/pemanggilan subrutin. Stack diimplementasikan mulai dari lokasi memori tertinggi ke lokasi memori terendah (PUSH)

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



Timer dan Counter

Didalam mikrokontroler AVR Atmega 16 terdapat tiga buah Timer / Counter 32 bit yang dapat difungsikan sebagai timer atau counter.

Timer 0 = 8 bit

Timer 1 = 16 bit

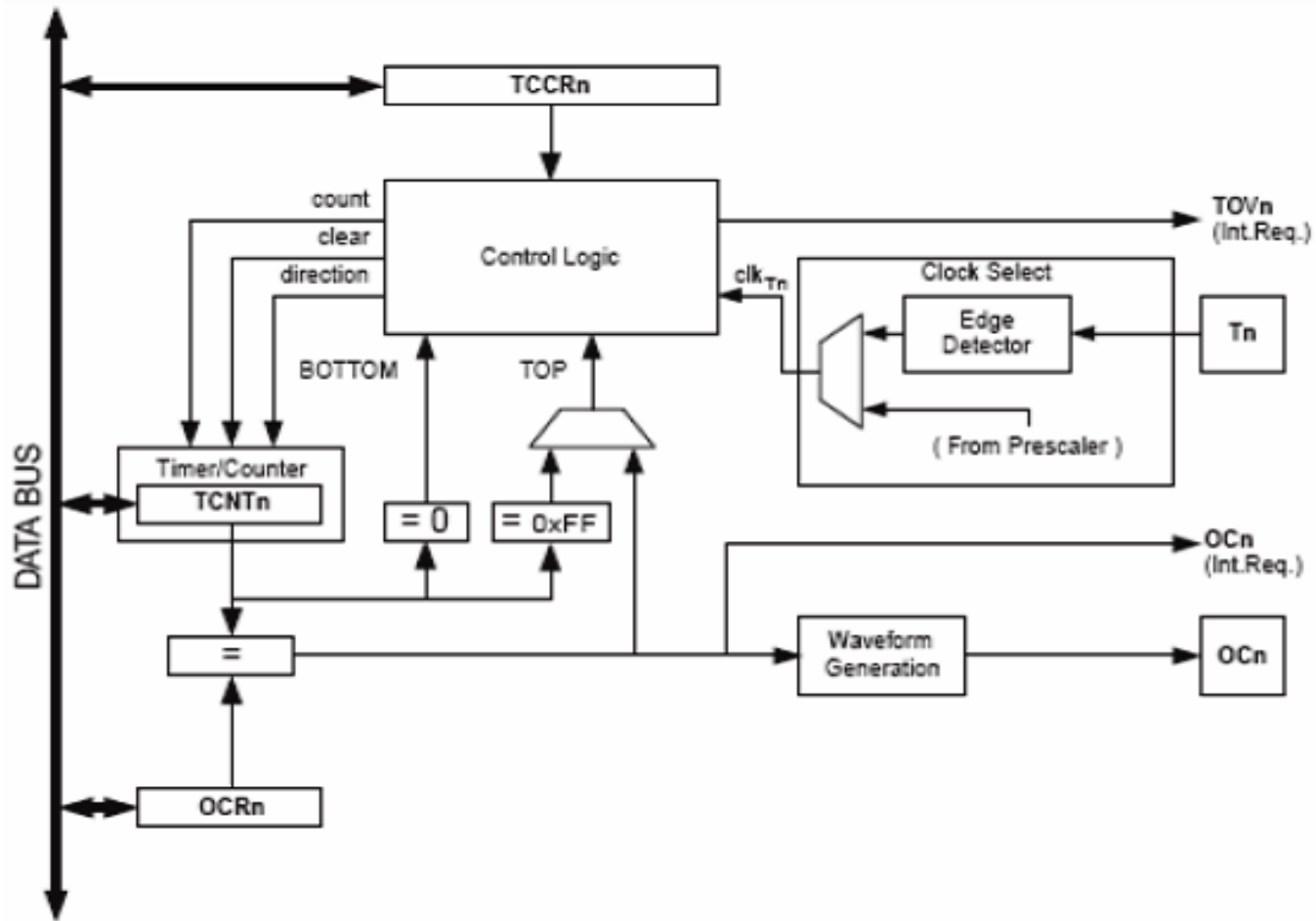
Timer 2 = 8 bit

Timer / Counter 8 bit

Memiliki multifungsi adapun fiturnya :

- Counter satu kanal
- Timer dinolkan saat Match compare (Autoreload)
- Phase Correct Pulse Width Modulator (PWM)
- Frekuensi generator
- 10 bit cloc prescaler
- interupsi timer / toimer overflow (T0Vn) dan Compare Match (0CFn)

Timer/Counter 8 bit dapat menghitung maksimal 255 (00-FF)



Gambar 1. Diagram Timer/Counter 8-bit



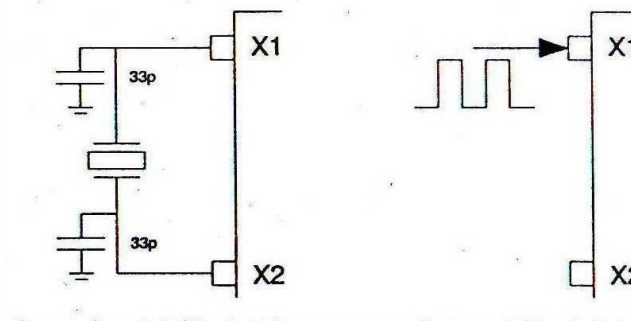
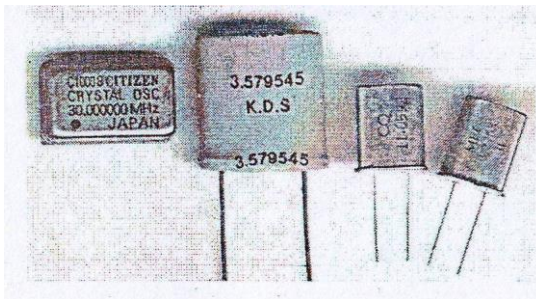
Timer/Counter 16 bit

Pada mode normal, TCNT1 akan menghitung naik dan membangkitkan interrupt timer/counter 1 ketika nilainya berubah dari 0xFFFF ke 0x0000. untuk menggunakan timer yang menghitung mundur cukup dengan memasukkan nilai yang diinginkan ke TCNT1 dan menunggu sampai terjadi interrupt, tetapi untuk timer yang menghitung maju nilainya harus $65536 - (\text{timer value})$.



Timer/ counter akan berfungsi sebagai counter apabila sinyal pemicu berasal dari clock eksternal.

Sebaliknya timer/ counter akan berfungsi sebagai timer jika sinyal pemicu berasal dari clock internal.
Lama waktu sebuah timer tergantung dari osilator yang dibangkitkan oleh X-tal yang digunakan.



Gambar X-tal dan Rangkaian Osilator



Pemakaian Timer Untuk Mengukur waktu

Fungsi utama timer adalah mengukur waktu. Pada saat sebuah timer digunakan waktunya akan satu setiap siklus mesin (siklus instruksi).

Karena X-tal yang digunakan sebagai pembangkit pulsa pada mikrokontroler ATMEGA adalah 12 Mhertz. Maka dapat dihitung berapa lama satu siklus

Frekuensi X-tal (f) = 12 MHz = 12.000.000 Hz

Siklus atau perioda (t) dimana:

$$t = 1/f$$

$$t = 1/(12.000.000)$$

$$t = 0,0000000833 \text{ detik}$$

$$t = 0.083 \text{ detik}$$



Sampai jumpa ?????



PERTEMUAN

4

BAHASA PEMOGRAMAN DAN FORMAT BILANGAN MIKROKONTROLER

ATMEGA 16/32/8535



Bahasa merupakan media komunikasi.

Manusia dapat berintegrasi dengan baik jika memahami apa yang diinginkan oleh orang lain.

Pada dasarnya bahasa yang di gunakan dapat di kelompokkan berdasarkan tahapan atau tingkat

1. Tingkat rendah yang digunakan anak-anak
2. Tingkat menengah berupa bahasa kebanyakan pergaulan atau sehari-hari
3. Bahasa tingkat tinggi seperti pada buku-buku ilmiah, Jurnal dan Artikel



Demikian juga pada peralatan yang berbasis mikroprosesor atau mikrokontroler akan dapat menjalankan suatu pekerjaan jika ada perintah (instruksi) yang sesuai pada Mikrokontroler.

Format bahasa pemrograman:

Sama seperti mikroprosesor agar CPU didalam mikrokontroler dapat mengerjakan suatu pekerjaan maka diperlukan program dalam bentuk perintah-perintah (instruksi)

Perintah-perintah tersebut di tulis dalam bentuk:

1. Bahasa mesin (bahasa tingkat rendah)
2. Bahasa Assembler/Assembly (bahasa tingkat menengah)
3. Basic, pascal, C++ dsb (bahasa tingkat tinggi)



Jenis bahasa yang digunakan pada Mikrokontroler

1. Bahasa mesin

Bahasa dalam bentuk biner yang disandikan menjadi code yang dikenal dengan *code mesin* biasanya dalam bentuk heksadesimal Hexa.

Semua perintah di masukan dalam bentuk kode mesin atau *Operational Code* (Op-Code) kode diterjemahkan mikrokontroler menjadi bentuk biner kemudian baru diproses.

Program pada CPU mikrokontroler yang disimpan didalam ROM merupakan bahasa mesin.



Umumnya perintah-perintah tersebut di tulis dalam bentuk bilangan hexa tetapi disimpan di dalam memory (ROM) dalam format bilangan biner contoh :

001110100b (74h)

11110000(F0h)

Bahasa mesin adalah bahasa tingkat rendah.

2. Bahasa Assembly (Assembler)

Bahasa yang di gunakan oleh programmer untuk membuat program sehingga dapat diproses oleh mikrokontroler. Bahasa assembler merupakan sandi yang merupakan singkatan kata dari kata-kata dalam bahasa inggris, contoh:

Mov dari Move

INC dari Increment

SJMP dari Short Jump, dll



2. Bahasa C

Penulisan program dalam bahasa C

`#include` merupakan file header atau library

`#define` merupakan definisi konstanta

`#if -#endif` merupakan kode program, dll

Contoh : `#include<mega16.h>` (merupakan lokasi standar file yang kita tentukan dari jenis ic yang digunakan).

`#define Max 100` (setiap kemunculan max diganti angka 100).

`#if` mengawali program

`#endif` (mengakhiri program)

Contoh `#define pembagi 15`
`#if (pembagi >0)`
`y=x/pembagi;`
`#endif`



Variabel

Variabel adalah untuk menyimpan dan mengakses data yang mewakili memori dalam mikrokontroler. Variable yang harus dideklarasikan (memberitahu Compiler) dengan **tipe data** beserta **nama variabel** yang akan digunakan, Bahasa C bersifat *case sensitif* dimana huruf kapital dan huruf kecil dibedakan.

Tiap tipe data mempunyai jangkauan bilangan yang dapat disimpan, hal ini akibat dari *byte* memori yang dipesan dan bentuk bilangan bertanda atau tidak. Misalnya *unsigned char* oleh *compiler* disediakan 1 *byte* memori RAM sehingga hanya bisa menampung bilangan dari 0 s.d. 255 sedangkan jika bertanda -128 s.d. 127.



Tipe data

Tipe data	byte	bit	Minimum	Maximum
Char	1	8	-128	127
signed char	1	8	-128	127
Unsigned char	1	8	0	255
Int	2	16	-32768	32767
Signed int	2	16	-32768	32767
unsigned int	2	16	0	65535
Long	4	32	-2147483648	2147483647
signed long	4	32	-2147483648	2147483647
unsigned long	4	32	0	4294967295
Float	4	32	1.28E-38	3.4E38



Penanganan Variabel

1. Huruf Penambahan variabel sebaiknya menggunakan huruf-huruf kecil dan huruf-huruf kapital sebaiknya digunakan untuk penanaman bilangan konstan. Panjang maksimal nama variabel 31 karakter.
2. Nama variabel sebaiknya menggunakan nama yang mewakili penggunaan variabel tersebut sehingga mudah untuk dipakai, dibaca dan dipahami. Nama variabel harus diawali dengan huruf tidak boleh dengan angka, jika diawali dengan angka gunakan *underscore*(_) sebelum angka ditulis



3. Gunakan tipe data sesuai dengan kebutuhan variabel (perkiraan bilangan yang akan disimpan) itu digunakan sehingga dapat menghemat ruangan SRAM mikrokontroler. Gunakan se-efisien mungkin SRAM, misalnya jangan menggunakan tipe “int” jika variabel yang kita gunakan hanya untuk menyimpan angka-angka dari 0 sampai 200 walaupun ole *compiler* tidak *error* namun tidak efisien karena “int” akan memakan 2 *byte* sedangkan *unsigned char/char* hanya memakan 1 *byte*.



Sistem Bilangan

Karena bermain dalam digital kita harus paham tentang bilangan, bilangan yang biasanya digunakan untuk pemograman yaitu : Desimal (13_D . Heksadesimal 05_H) dan Biner (0011_B)

1. Bilangan Desimal (Basis 10) adalah bilangan yang mempunyai sepuluh simbol angka 0 s/d 9. aturan penulisan langsung, seperti 23 tanpa dirubah
2. Bilangan Heksadesimal (basis 16) adalah bilangan yang mempunyai enam belas simbol yaitu : 0 s/d 9 dan A s/d F, penulisan 0x atau \$ misalnya : 0x2fe
3. Bilangan Biner (Basis 2) adalah bilangan yang mempunyai dua simbol 0 dan 1, penulisannya : 0b110001100



Konversi Bilangan

Konversi Biner ke Heksadesimal

Contoh : 0b10110001 = dibagi dua bagian yaitu (1011) dan (0001)

$$1011 = 8 + 0 + 2 + 1 = B$$

$$0001 = 0 + 0 + 0 + 1 = 1$$

$$\text{jadi} = 0b10110001 = 0xB1_{(H)}$$

Konversi Heksadesimal ke Biner

Contoh : 0x2F = dibagi dua bagian yaitu (2) dan (F)

$$2 = 0 + 0 + 2 + 0 = 0010 \text{ biner}$$

$$F = 8 + 4 + 2 + 1 = 1111 \text{ biner}$$

$$\text{jadi} = 0x2F_{(H)} = 0b00101111$$



Konversi Biner ke Desimal

Contoh : 0b10110001=

$$(1 \times 2^7) + (0 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (1 \times 2^0)$$

$$128 + 0 + 32 + 16 + 0 + 0 + 0 + 1 = 177$$

$$\text{jadi} = 0b10110001 = 177_{(D)}$$

Konversi Heksadesimal ke Desimal

Contoh : 0x2F= $(2 \times 16) + (F \times 16)$

$$0x2F = 32 + 15$$

$$\text{jadi} = 0x2F_{(H)} = 47_{(D)}$$



3. Bahasa Level Tinggi

Bahasa pemrograman tingkat tinggi yang di gunakan oleh mikrokontroler Atmega 16/32/8535 antara lain:

Basic

Pascal

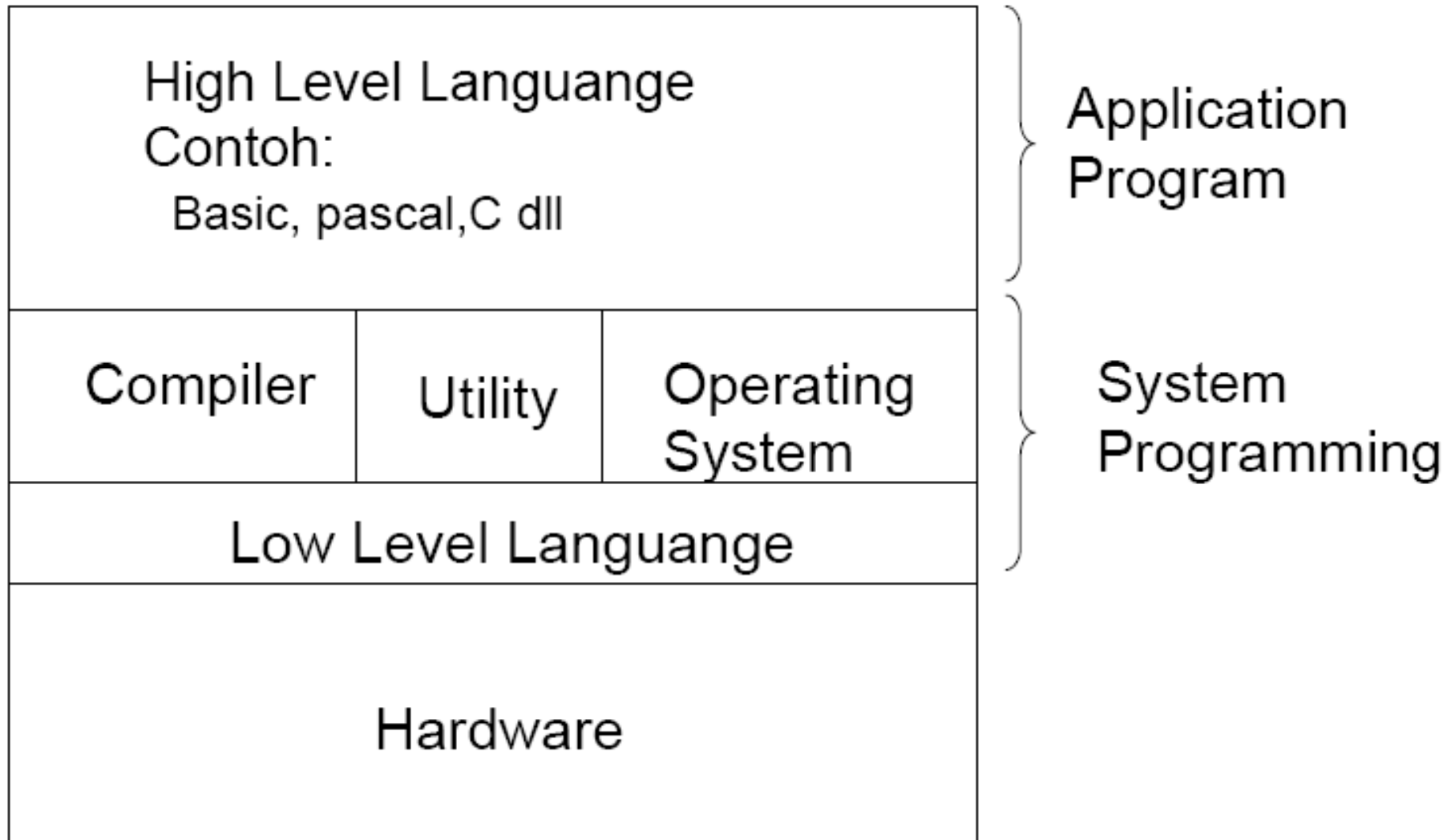
Bahasa C/C++ dan Java

Bahasa C paling banyak dikembangkan diantaranya:

Keil Compiler oleh Keil Corp

Small Device C Compiler oleh Sundeep Duta

Jika dalam penulisan program menggunakan bahasa assembler atau bahasa tingkat tinggi maka di butuhkan software yang dikenal dengan *Compiler* untuk menterjemahkan bahasa yang di mengerti manusia menjadi bahasa mesin yang dapat di mengerti Mikrokontroler



Gambar level bahasa pemograman



Sampai Jumpa !!!!



PERTEMUAN

5

SET INSTRUKSI MIKROKONTROLER ATMEGA 16/32/8535



Pendahuluan

Untuk Mendeklarasikan sebuah program atau instruksi maka diperlukan sebuah alogaritma dimana terdiri dari berbagai pengendali aliran program.

Pengendali Bersyarat

If(...) { ... }

Digunakan untuk mengecek satu kondisi untuk satu blok jawaban.

Cara penulisan:

```
if (uji_benar_atau_salah_)  
{ _eksekusi_blok_ini_jika_benar_ ;};
```



If (...) { ... } else { ... }

Digunakan untuk mengecek satu kondisi untuk dua blok jawaban.

Cara penulisan:

```
If (_uji_benar_atau_salah_)  
{_eksekusi_blok_ini_jika_benar_;}  
Else {_eksekusi_blok_ini_jika_salah_;};
```

While (...) { ... }

Digunakan untuk perulangan/looping/iterasi jika kondisi yang diuji bernilai benar.

Cara penulisan:

```
While (_uji_benar_atau_salah_)  
{_selama_benar_blok_ini_akan_selalu_dieksekusi_;};
```



If (...) { ... } else if (...) { ... } else { ... };

Digunakan untuk mengecek beberapa kondisi yang berkaitan.

Cara penulisan:

```
If (_uji_benar_atau_salah_)
{_eksekusi_blok_ini_jika_benar_;}
else if (_uji_benar_atau_salah_)
{ _eksekusi_blok_ini_jika_benar_;}
else if (_uji_benar_atau_salah_)
{ _eksekusi_blok_ini_jika_benar_;}
.....
else if (_uji_benar_atau_salah_)
{ _eksekusi_blok_ini_jika _benar_;}
else
{ _eksekusi_blok_ini_jika_salah_;;}
```




do { ... } while (...)

Digunakan untuk perulangan/looping/iterasi jika kondisi yang diuji bernilai benar. Perbedaannya dengan while tanpa do adalah blok dieksekusi dulu baru diuji.

Cara penulisan :

```
do
{ _selama_benar_blok_ini_akan_selalu_dieksekusi_; }
while ( _uji_benar_atau_salah_ );
```

for (... ; ... ; ...) { ... }

Digunakan untuk perulangan/looping/iterasi dengan kondisi dan syarat yang ditentukan.

Cara penulisan :

```
for ( _kondisi_awal ; _uji_kondisi_ ;
    _aksi_jika_benar )
{ _eksekusi_blok_ini_jika_benar_ ; }
```



switch

Digunakan untuk menguji satu variabel dengan beberapa jawaban konstanta.

Cara penulisan :

```
switch (_variabel_uji_) {  
  case _konstanta1_: _pernyataan1_;break;  
  case _konstanta2_: _pernyataan2_;break;  
  case _konstanta3_: _pernyataan3_;break;  
  .....  
  .....  
  case _konstantaN_: _pernyataanN_;break;  
  default: _pernyataan_jika_dalam_  
    semua_case_tidak_ada_;break;  
}
```



Pengendali Tanpa Bersyarat

Goto

Digunakan untuk melompat ke baris program tertentu (dalam satu fungsi) yang telah diberi label.

Cara penulisan :

Goto label ;

Contoh :

Lagi :

.....kode program.....

.....

. goto lagi;



Super loop :

Digunakan untuk memutar/mengulang aliran program tanpa henti dari awal blok hingga akhir blok. :

Cara penulisan:

```
While (1) {
```

```
.....
```

```
.....
```

```
}
```

atau

```
for ( ; ; ) {
```

```
.....
```

```
.....
```

```
}
```



Continue

Digunakan untuk melewati satu pernyataan atau blok jika dalam blok tersebut hanya memakai pernyataan continue, misalnya :

```
if (x==10) {continue;}
```

```
else { x++; } ;
```

jika x sam dengan 10 lanjutkan (keluar dari if). Jika tidak sama dengan 10 naikan nilai x.



Instruksi Aritmatika

Perintah-perintah perhitungan yang digunakan dalam bahasa C dapat di lihat pada tabel di bawah

Operator Aritmatika	Keterangan
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
%	Sisa Bagi (modulus)



Operator Kondisi

Operator Kondisi	Keterangan
<	Lebih kecil
<=	Lebih kecil atau sama dengan
>	Lebih besar
>=	Lebih besar atau sama dengan
==	Sama dengan
!=	Tidak sama dengan



Operator Bit

Operator Bit	Keterangan
~	Mebalik isi tiap bit
&	Bit AND
	Bit OR
^	Bit Exclusive OR
>>	Menggeser tiap bit ke kanan (Right Shift)
<<	Menggeser tiap bit kekiri (Left Shift)

Operator Logika

Operator Logika	Keterangan
!	<i>Boolean</i> NOT
&&	<i>Boolean</i> AND
	<i>Boolean</i> OR



Operator Assignment

Operator Assignment	Keterangan
=	Untuk memasukkan nilai
+=	Untuk menambah nilai dari keadaan semula
-=	Untuk mengurangi nilai dari keadaan semula
*=	Untuk mengalikan nilai dari keadaan semula
/=	Untuk melakukan pembagian terhadap bilangan semula
%=	untuk memasukkan nilai sisa bagi dari pembagian bilangan semula
<<=	Untuk memasukkan shift left
>>=	Untuk memasukkan shift right
&=	Untuk memasukkan bitwise AND
^=	Untuk memasukkan bitwise XOR
\=	Untuk memasukkan bitwise OR



Operator Aritmatika

Pada Perkalian

Contoh :

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

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

```
Void main()
```

```
{
```

```
    int bil1;
```

```
    int bil2;
```

```
    DDRB=0xff;
```

```
    PORTB=0xff;
```

```
    bil1=4;
```

```
    bil2=2;
```

```
    PORTB=bil1*bil2;
```

```
}
```



Pada Pembagian
Contoh :

```
#include <mega16.h>
#include <delay.h>
Void main()
{
    int bil1;
    int bil2;
    DDRB=0xff;
    PORTB=0xff;
    bil1=10;
    bil2=2;
    PORTB=bil1/bil2;
}
```



Pada Penjumlahan
Contoh :

```
#include <mega16.h>
#include <delay.h>
Void main()
{
    int bil1;
    int bil2;
    DDRB=0xff;
    PORTB=0xff;
    bil1=0x30;
    bil2=0x20;
    PORTB=bil1+bil2;
}
```



Pada Pengurangan
Contoh :

```
#include <mega16.h>
#include <delay.h>
Void main()
{
    int bil1;
    int bil2;
    DDRB=0xff;
    PORTB=0xff;
    bil1=0x30;
    bil2=0x20;
    PORTB=bil1-bil2;
}
```



Pada Modulus
Contoh :

```
#include <mega16.h>
#include <delay.h>
Void main()
{
    int bil1;
    int bil2;
    DDRB=0xff;
    PORTB=0xff;
    bil1=13;
    bil2=2;
    PORTB=bil1%bil2;
}
```




Dest (*ination*)/ *Operand* adalah tujuan: < dest> dan <source> adalah dua operand dengan beragam kombinasi penggalamatan

Tabel Instruksi Logika

Bit 1	Bit 2	Hasil Operasi AND	Hasil Operasi OR	Hasil Operasi XOR
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0



Sampai Jumpa ????



PERTEMUAN

6

TEKNIK PEMOGRAMAN MIKROKONTROLER ATMEGA 16/32/8535



TEKNIK PEMOGRAMAN

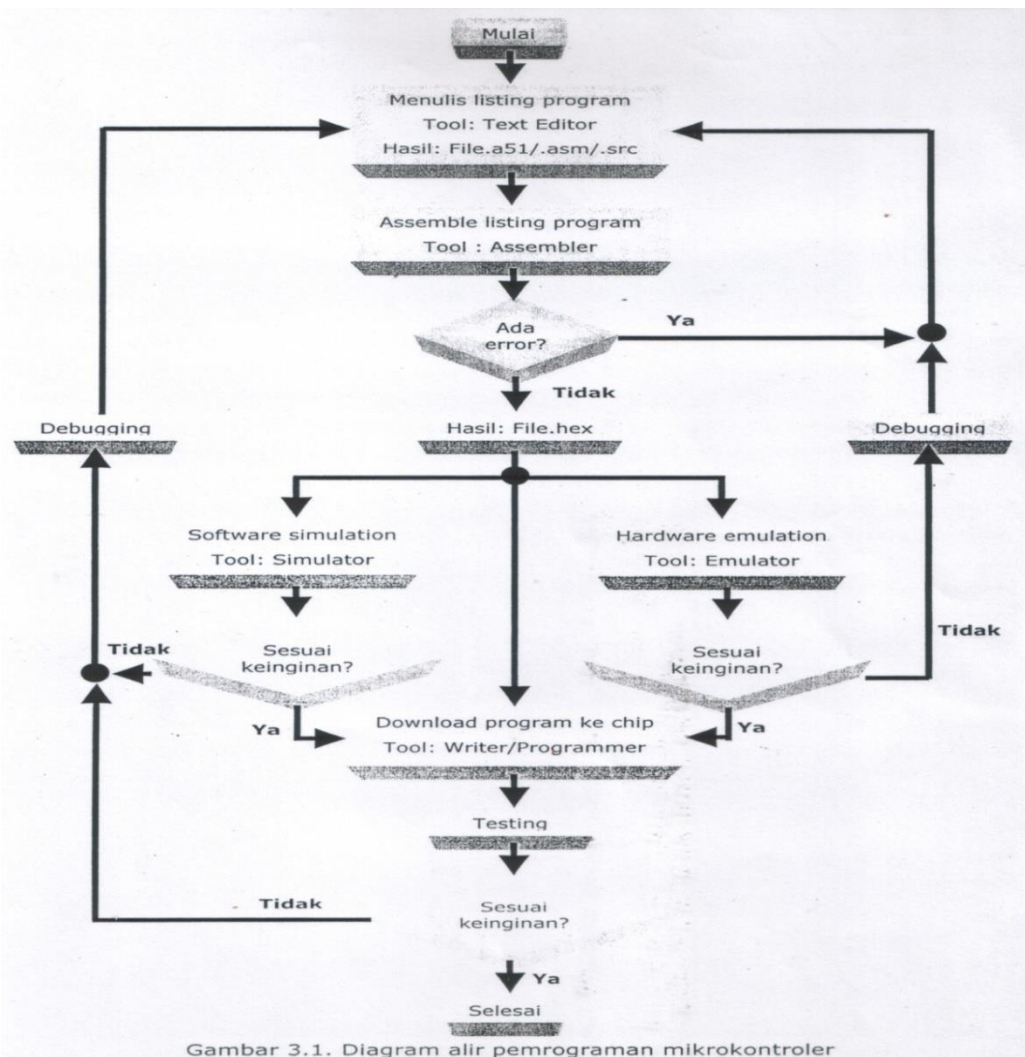
PEMOGRAMAN MIKROKONTROLER

Untuk dapat menjalankan suatu fungsi mikrokontroler ATmega membutuhkan perintah-perintah yang ditulis kedalam chip (IC).

Pembuatan program mikrokontroler Atmega bisa menggunakan dua program aplikasi:

- ~ *Codevision AVR* atau *AVR Studio* merupakan program aplikasi untuk penulisan listing program dengan format bahasa C.
- ~ *Proteus* merupakan program aplikasi untuk melakukan simulasi listing program dengan format hex
- ~ *ISP Programmer V1.68* merupakan program aplikasi untuk mengisikan program (*DownLoad*) program kedalam IC mikrokontroler.

TEKNIK PEMOGRAMAN



Gambar 3.1. Diagram alir pemrograman mikrokontroler

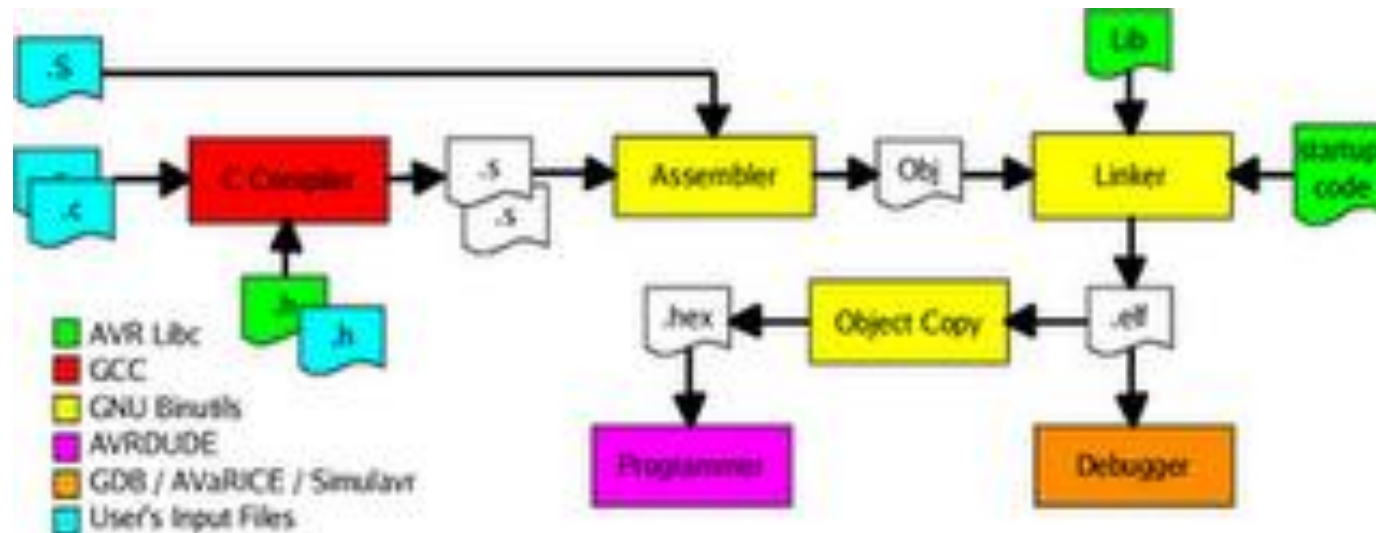
Alur Pembuatan Program Mikrokontroler



TEKNIK PEMOGRAMAN

PEMOGRAMAN DENGAN MELAKUKAN COMPILER

Software CodeVision AVR merupakan lembar kerja penulisan listing program dan juga dapat melakukan compiler atau terjemahan, codevision AVR sendiri merupakan kumpulan dari program yang berfungsi untuk mengubah bahasa C menjadi Hex (.Hex) intel.



Alur Compiler Program Mikrokontroler



Compiler C akan menghasilkan file dengan format ".asm" dari setiap file input ".c" yang kemudian akan diubah ke dalam file objek oleh **assembler**, di mana banyaknya file objek sama dengan file ".asm". Setelah proses tersebut dilakukan, **linker** akan bertugas untuk menyatukan file-file objek dan fungsi-fungsi yang bersilangan di antara file objek dan mengambil modul *library* C yang digunakan ke dalam satu file objek yaitu file ".ELF". Dengan menggunakan **avr-objcopy** file ".ELF" diubah menjadi file ".Hex"

Dari keempat proses tersebut (**compiler**, **assembler**, **linker**, dan **avr-objcopy**) oleh WinAVR dijadikan dalam satu file yang disebut "*makefile*", sedangkan untuk proses kompilasi "*makefile*" itu sendiri ditangani oleh "**make.exe**".

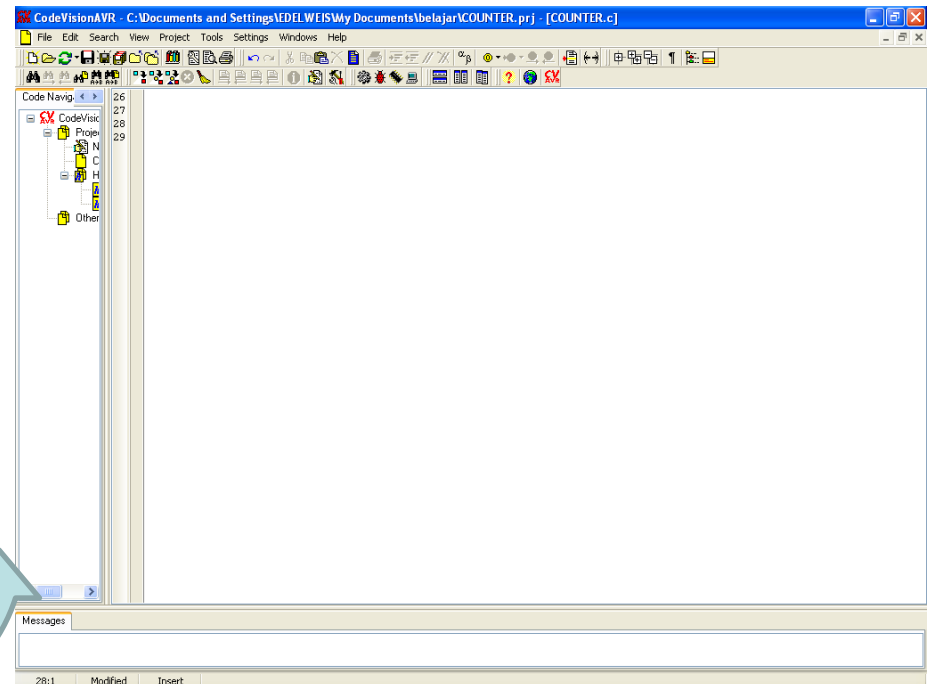
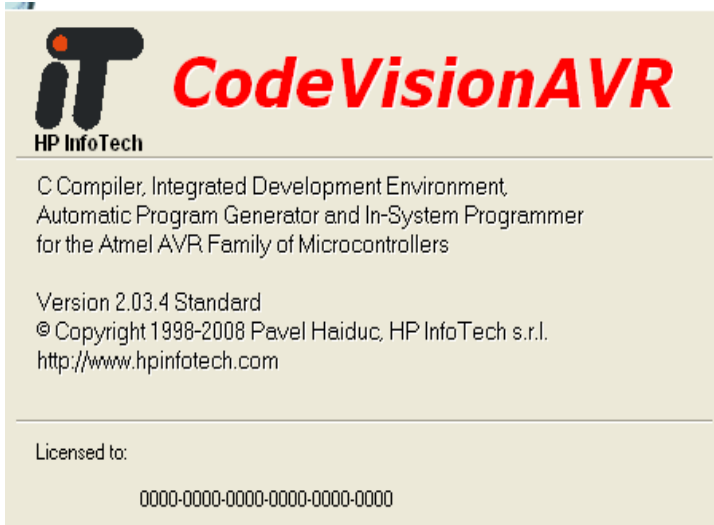


Proses Instalasi Codevision AVR

Codevision AVR merupakan software open source, yang dapat di download : <http://www.hpinfotech.ro>

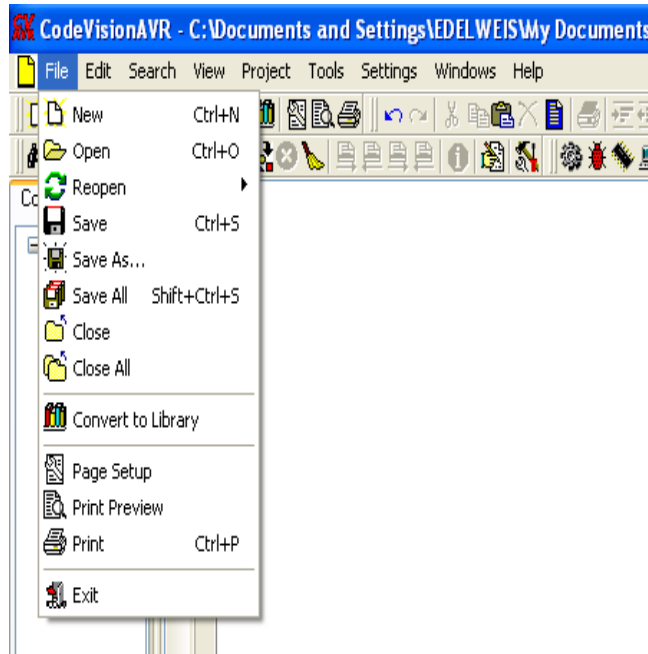


Setup
CodeVisionAVR C Compiler Setup

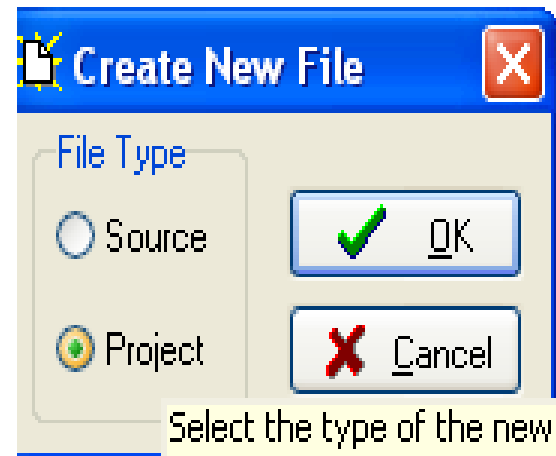


Lembar kerja Codevision AVR

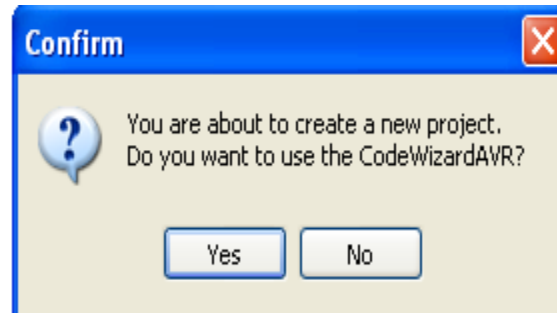
1. Klik New



2. Pilih Project lalu klik OK

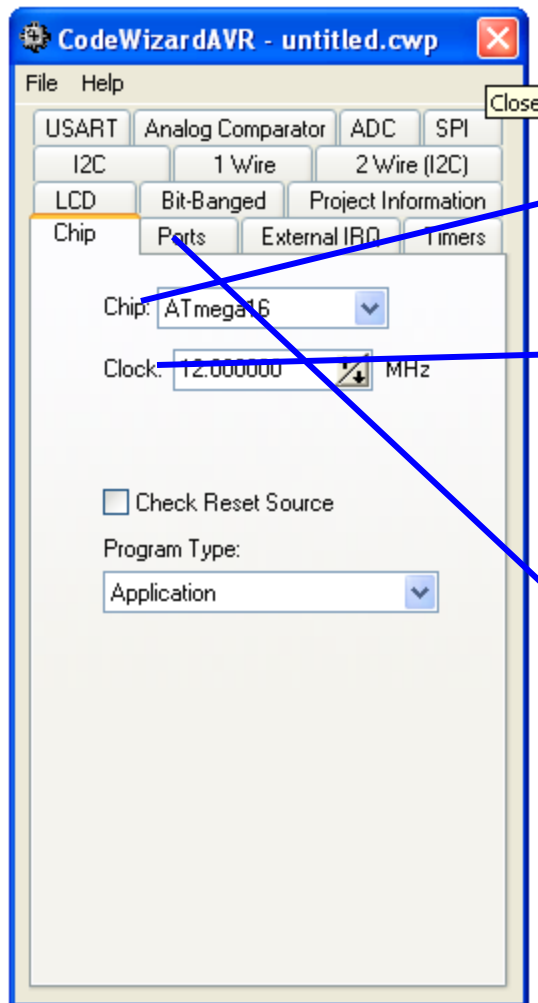


3. Klik Yes





4. Lakukan penyetingan



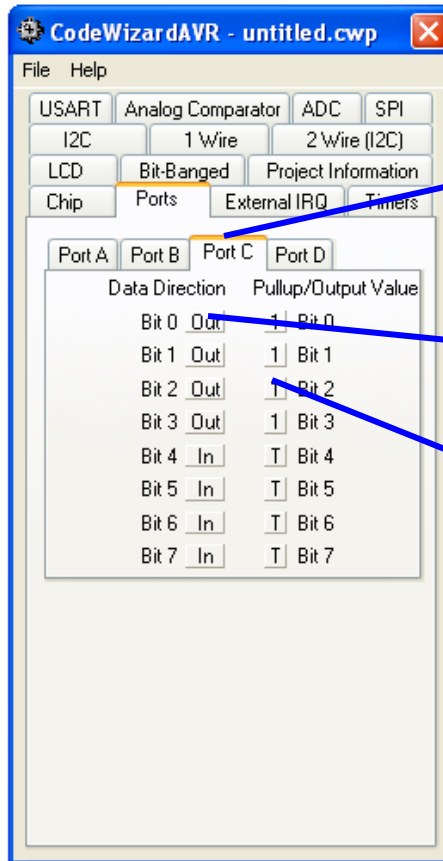
Pada Chip pilih atmega 16

Ubah nilai Clock menjadi
12.000000MHz

Klik Port untuk memilih kaki
port yang digunakan



5. Karena rangkaian yang di raknai di simulasi proteus menggunakan port C maka pilih port C



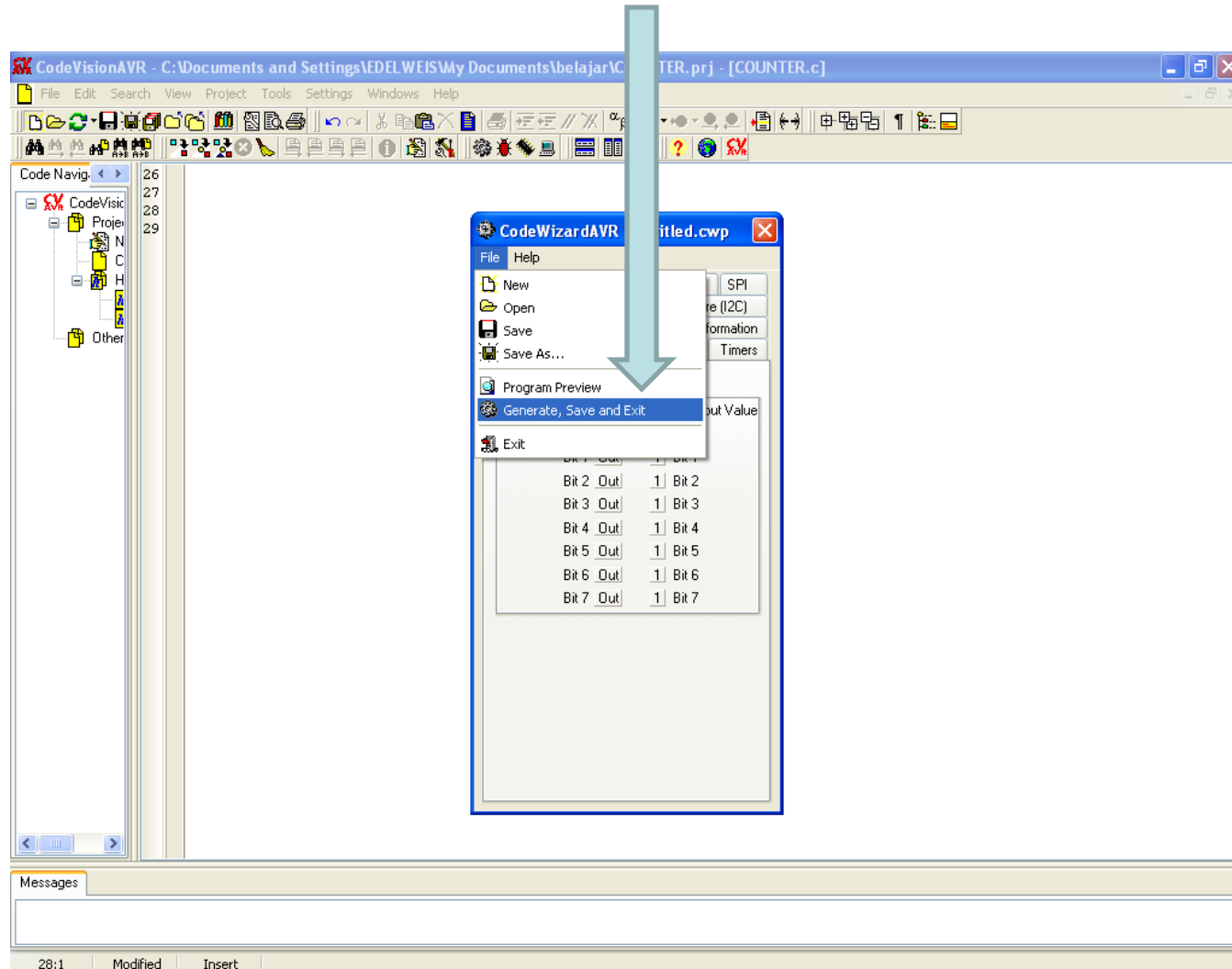
Bagian Bit In rubah menjadi
→ Out dengan klik,

Output yang 0 rubah menjadi
logika 1

**NB : jangan melakukan penyetingan
yang lainnya bila tidak diperlukan**

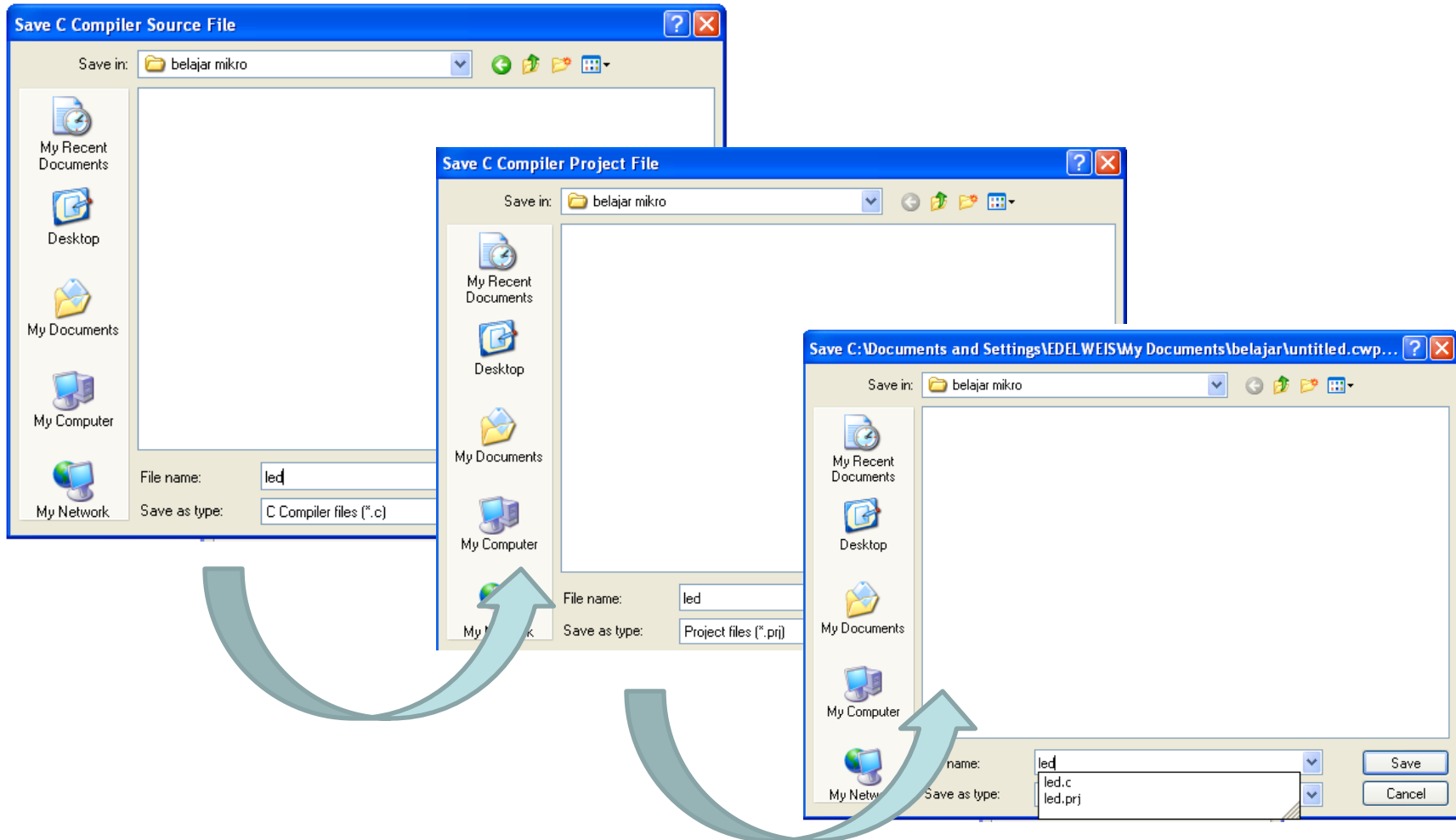


6. Lakukan penyimpanan dengan klik file. Pilih Generate, Save and Exit

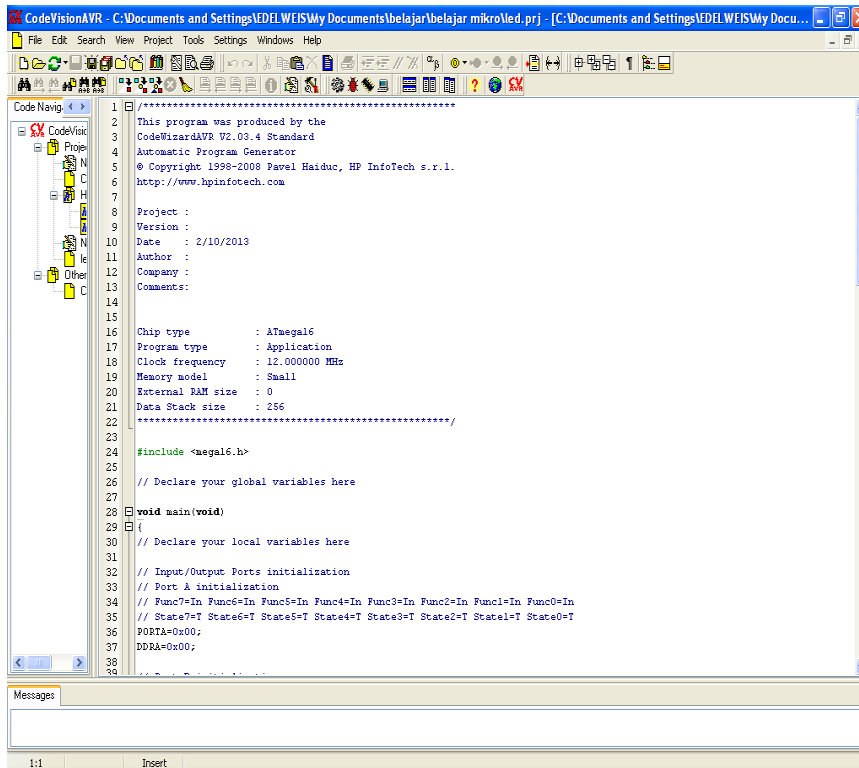




7. Penyimpanan sampai 3 kali dengan nama yang sama cth : LED



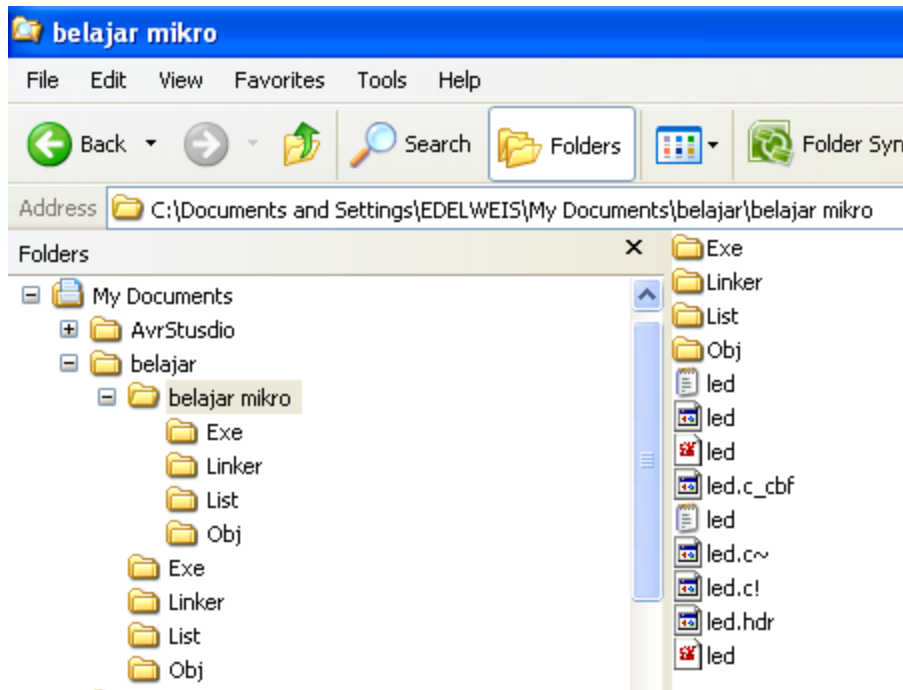
8. Hasil dari penyimpanan sampai 3 kali



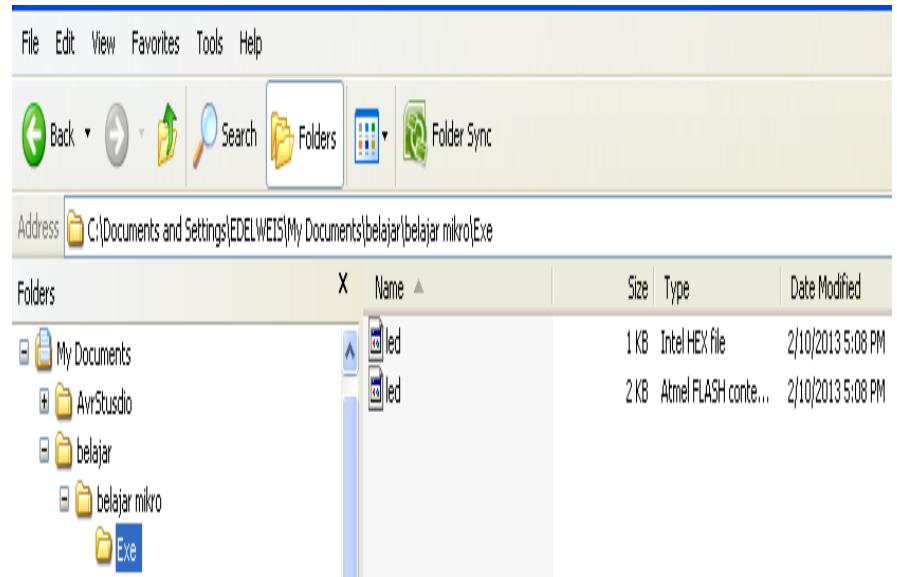
```
1 //*****
2 This program was produced by the
3 CodeWizardAVR V2.03.4 Standard
4 Automatic Program Generator
5 © Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.
6 http://www.hpinfo.tech.com
7
8 Project :
9 Version :
10 Date : 2/10/2013
11 Author :
12 Company :
13 Comments:
14
15
16 Chip type : ATmega16
17 Program type : Application
18 Clock frequency : 12.000000 MHz
19 Memory model : Small
20 External RAM size : 0
21 Data Stack size : 256
22
23
24 #include <mega16.h>
25
26 // Declare your global variables here
27
28 void main(void)
29 {
30 // Declare your local variables here
31
32 // Input/Output Ports initialization
33 // Port A initialization
34 // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
35 // State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
36 PORTA=0x00;
37 DDRA=0x00;
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
26
```




10. Hasil setelah di compile

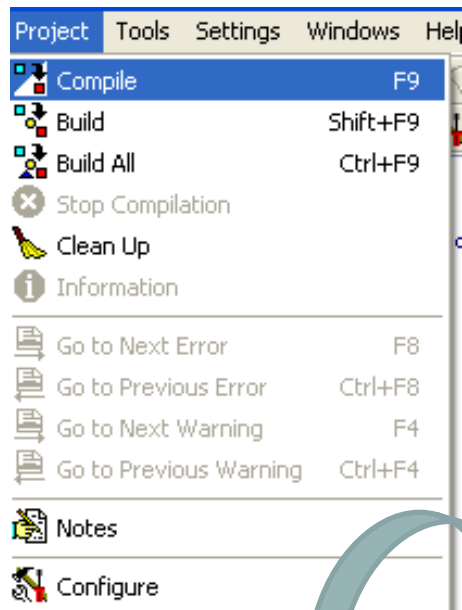


11. Led yang bertype hex

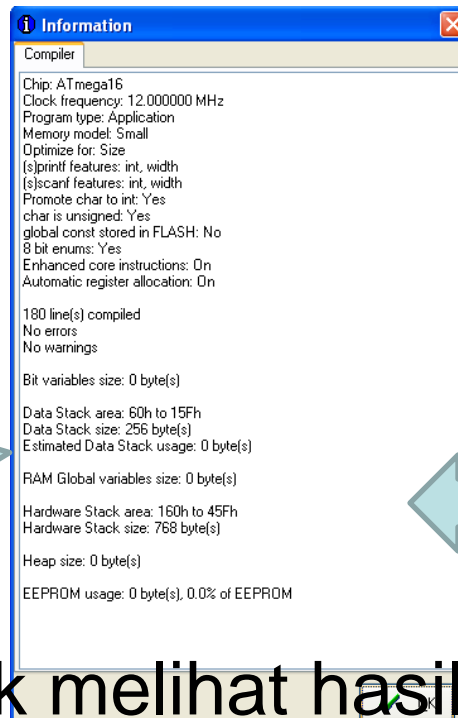
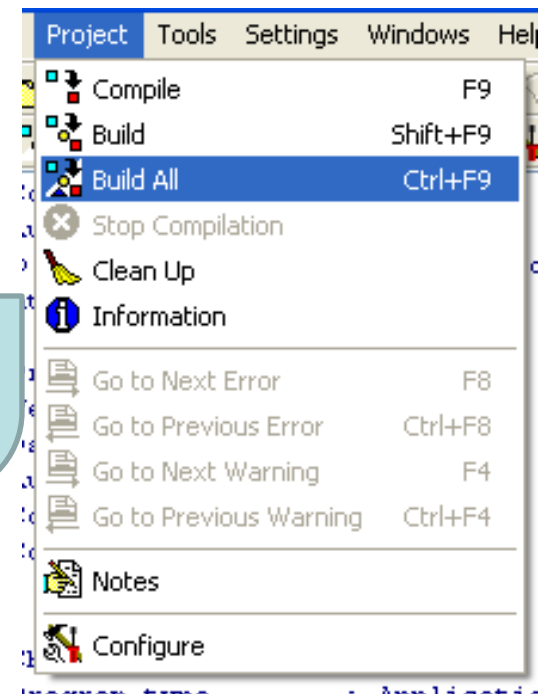




12. Klik Project, lalu pilih compile



13. Klik Project, Lalu pilih Build All



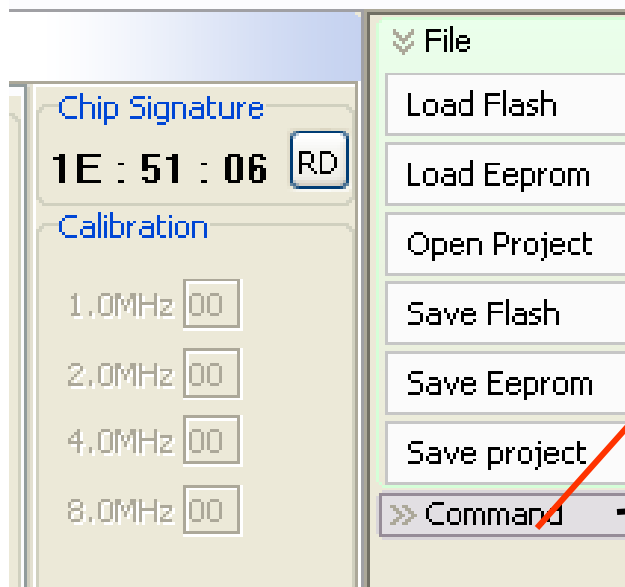
14. Information untuk melihat hasil compile apakah ada yang error



Software untuk dapat mendownload / mengisi program ke ic mikrokontroler Atmega dengan software ISP Prog V1.68.

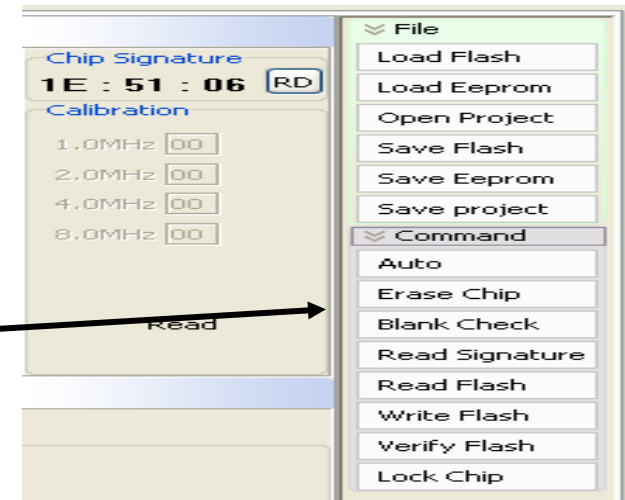
Cara menggunakannya :

1. Klik Program **ProgISP**  progisp



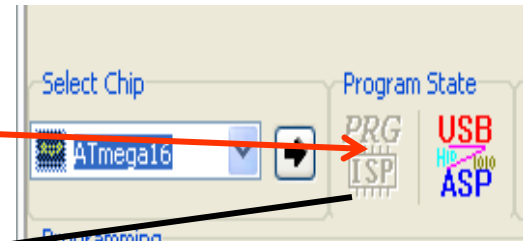
2. Klik **Command** Untuk membuka menu lainnya

Hasilnya



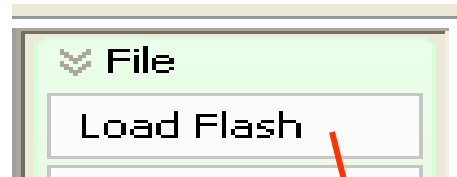


3. Bentuk Sebelum Di Pasang Downloader USB Ke Komputer IC/Laptop



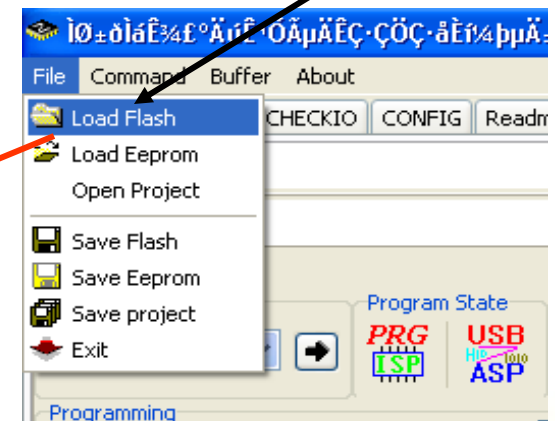
Hasilnya setelah di pasang

4. Klik Klik **Load Flash**



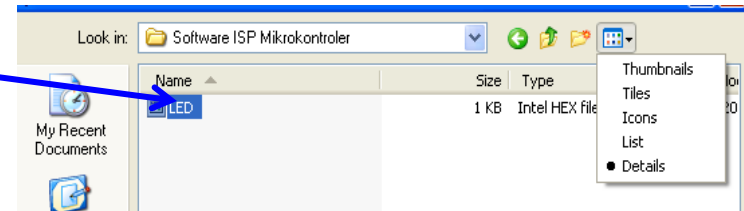
atau klik **File** lalu pilih **Load Flash**

untuk membuka file yang berupa (**Hex**).





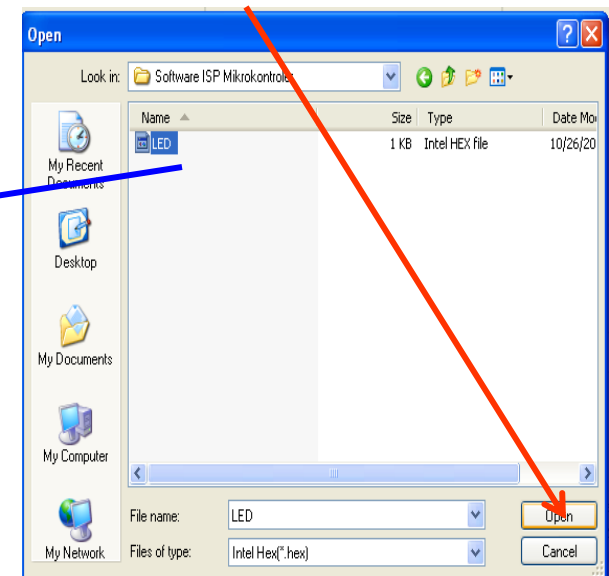
5. Klik file LED yang sudah di compiler atau berextensi (hex)



6. Klik Contoh Program Data **LED** lalu klik **Open**

Load Flash file C:\Documents and Settings\Administrator\My Documents\Software ISP Mikrokontroler\LED.HEX
A kind reminder:
Please click readme button and get yourself familiarized with the latest features of this software before you proceed to using it. Thank you!

Hasilnya





7. Klik **Write Flash**

Write Flash

1: Flash written succesfully
Load Flash file C:\Documents and Settings\Administrator\My Documents\Software ISP Mikrokontroler\LED.HEX
A kind reminder:
Please click readme button and get yourself familiarized
with the latest features of this software before you

8. Klik **Verify Flash**

Verify Flash

Flash Verify succesfully
1: Flash written succesfully
Load Flash file C:\Documents and Settings\Administrator\My Documents\Software ISP Mikrokontroler\LED.HEX
A kind reminder:
Please click readme button and get yourself familiarized

9. Jangan Klik **Lock Chip** bila tidak ingin mengunci program download **Verify Flash**

Lock Chip

10. Bila sudah berhasil atau succesfully dalam mendownload program ke dalam IC mikrokontroler, maka IC Siap dipakai dan lihat hasilnya.



Sampai Jumpa !!!!!



PERTEMUAN

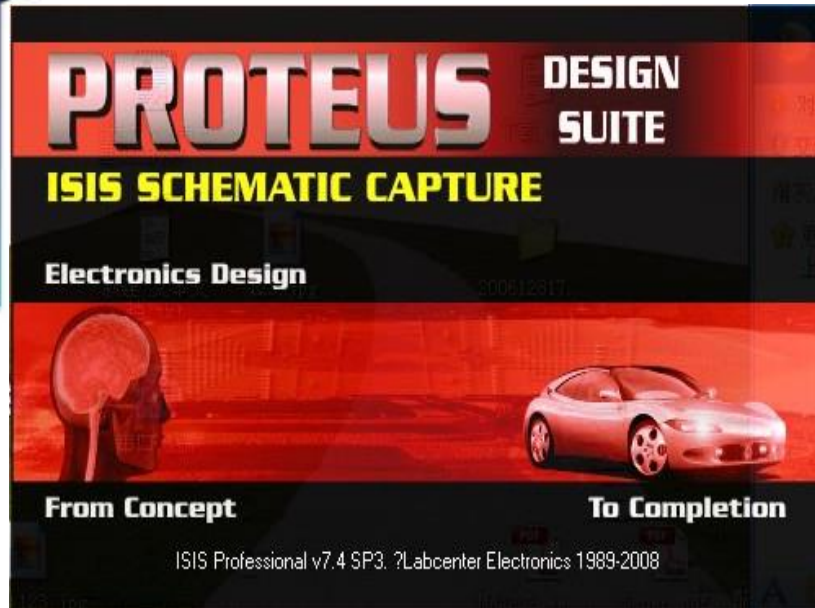
9

APLIKASI MIKROKONTROLER DENGAN SOFTWARE SIMULASI PROTEUS



Tujuan :

- Mahasiswa dapat memahami cara kerja program
- Mahasiswa dapat membuat aplikasi sederhana
- Mahasiswa dapat membuat aplikasi lebih dari satu jenis
- Program Simulasi dapat diterapkan pada alat yang dibuat



Proses Instalasi

Install Software Proteus

Install Software proteus dengan klik Setup, klik Next sampai selesai instalasi





Simulasi Mikrokontroler dengan Proteus

SP1.SP2,SP3, Vista dan Windows 7

Proteus merupakan software bersifat simulasi dengan merangkai rangkaian elektronika, baik rangkaian digital maupun mikrokontroler, Namun perlu diketahui untuk dapat menggunakan software proteus harus disesuaikan dengan sistem operasinya atau OS Windows,

Rangkaian aplikasi mikrokontroler yang kita buat untuk disimulasikan pada proteus, berupa program yang sudah di compier berupa hexsa (Format.Hex)

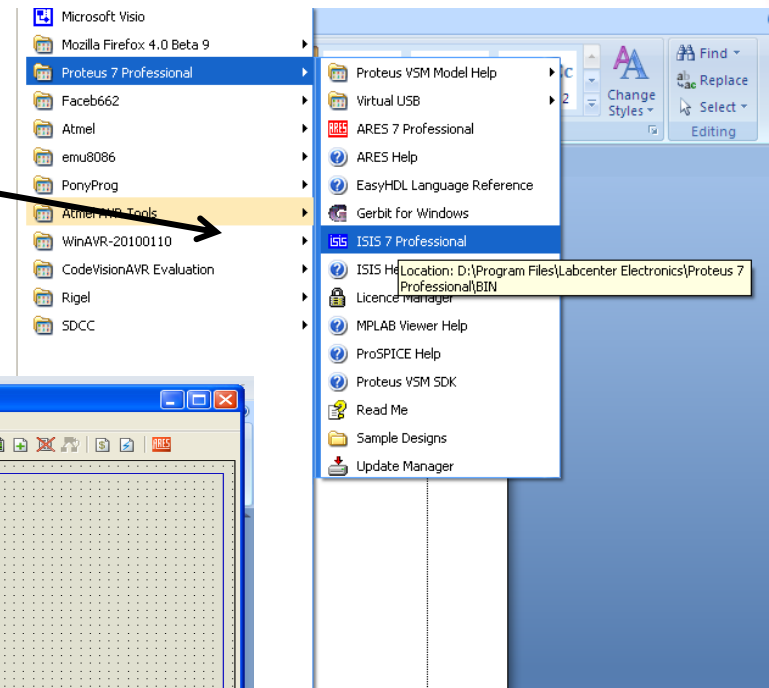
Proteus yang digunakan pada materi ini Proteus 7.4 SP3 dengan OS (sistem operasi) Windows SP1, SP2, SP3 dan Vista. Tidak bisa Windows 7



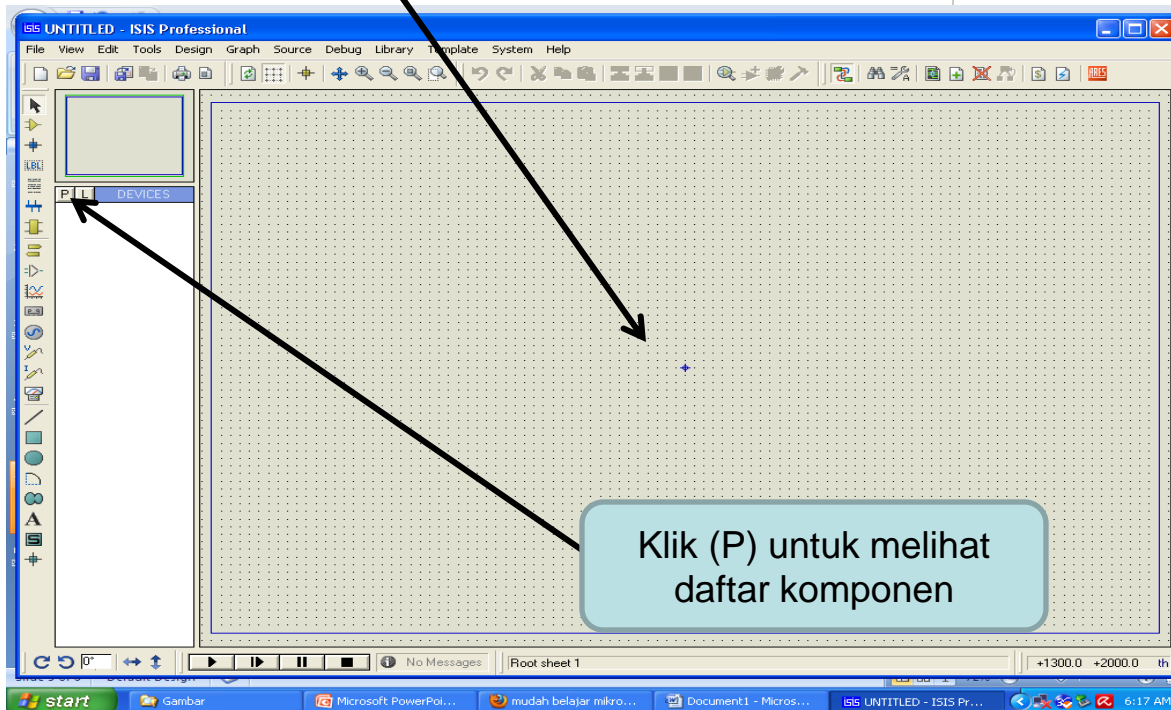
Cara menggunakan Software Proteus 7.4 SP3

Setelah selesai proses instalasi,

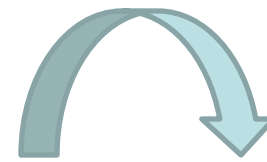
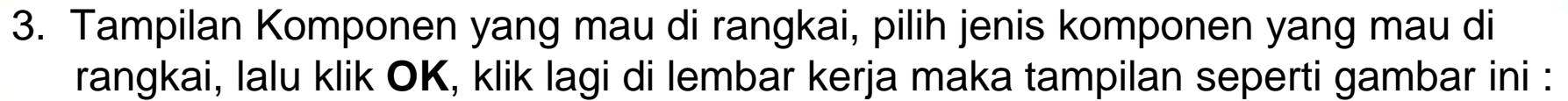
1. Klik ISIS Profesional



2. Lembar Kerja Proteus

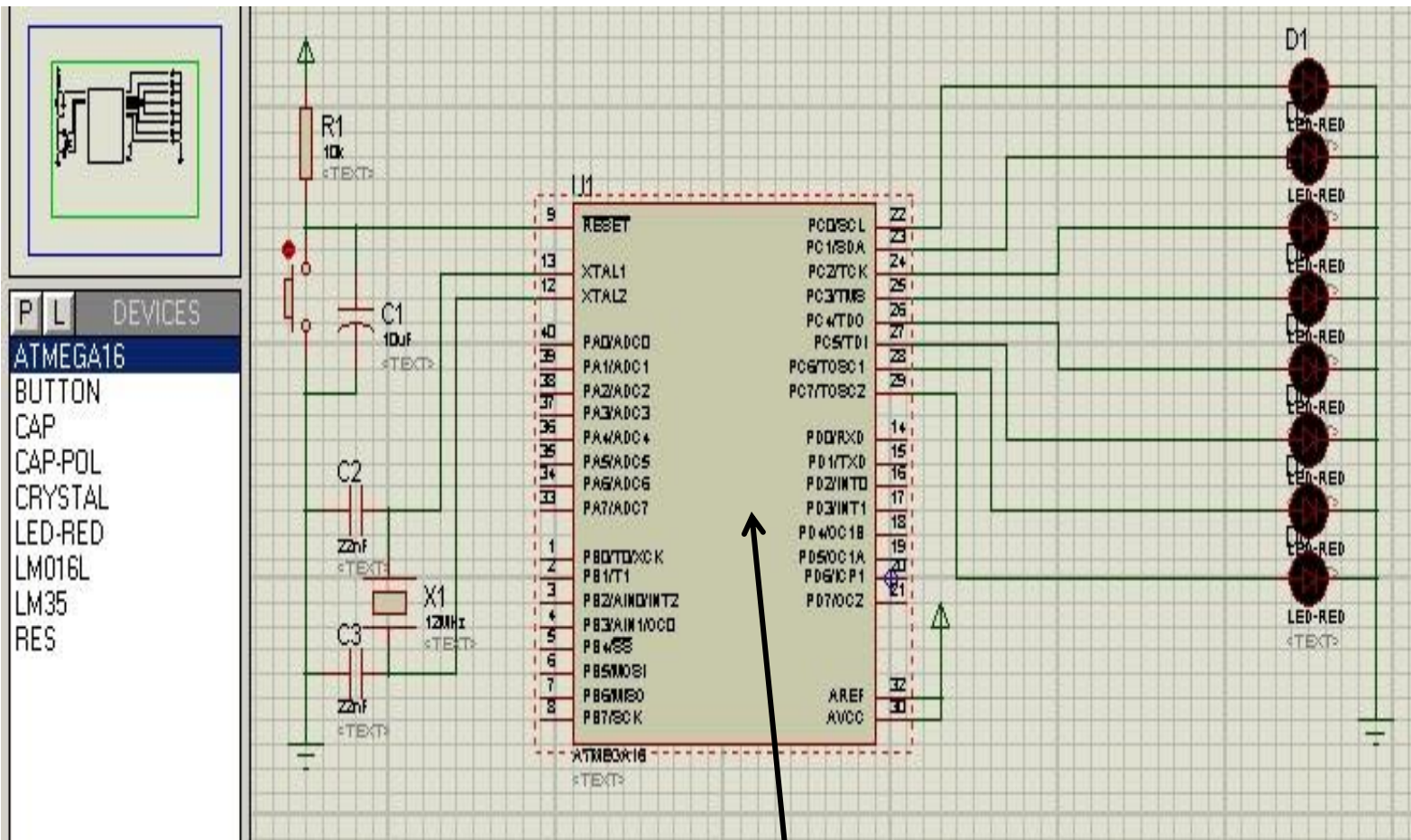


Klik (P) untuk melihat daftar komponen



Hasil





4. Setelah selesai dirangkai, klik kanan pada IC Mikrokontroler untuk memasukkan Program Hasil Compiler yang berupa Hex:



5. Tampilan Mengisi program ke IC Mikrokontroler yang berupa file type Hex seperti gambar ini , rubah CKSEL RC1MHz menjadi RC 8MHz,pada clock Frekuensi isi 12 Mhz, karena memakai Kristal 12 Mhz,

Edit Component

Component Reference: U1 Hidden: ☐

Component Value: ATMEGA32 Hidden: ☐

PCB Package: DIL40 Hide All

Program File: E:\Digital 2012\LED.HEX Hide All

BOOTRST (Select Reset Vector) (1) Unprogrammed Hide All

CKSEL Fuses: (0001) Int.RC 1M Hide All

Boot Loader Size: (00) 2048 words. Starts at 0x38(Hide All

SUT Fuses: (Default) Hide All

Advanced Properties:

Clock Frequency 12Mhz Hide All

Other Properties:

☐ Exclude from Simulation ☐ Attach hierarchy module

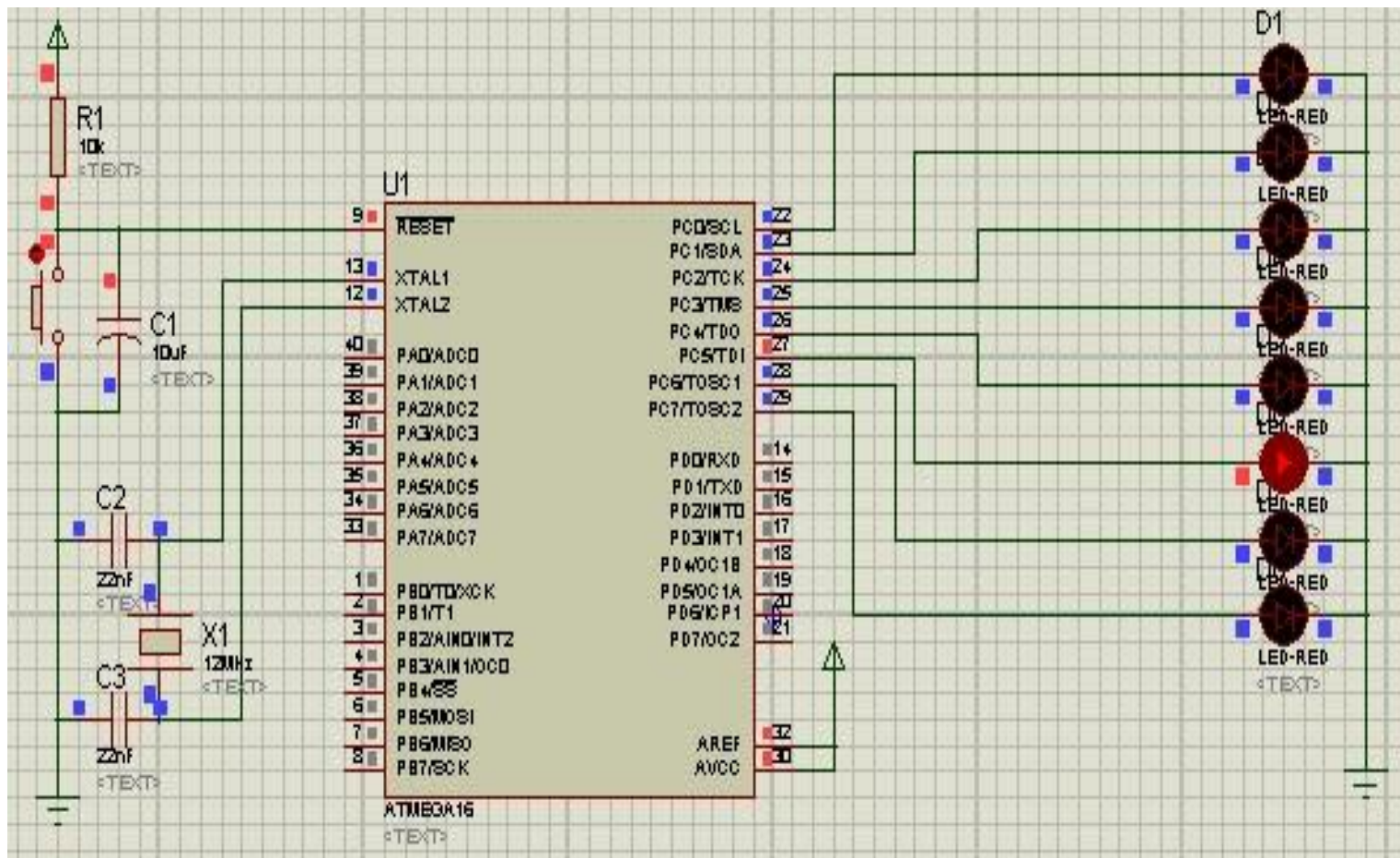
☐ Exclude from PCB Layout ☐ Hide common pins

☐ Edit all properties as text

Buttons: OK, Help, Data, Hidden Pins, Cancel

6. Klik Ok

Gambar hasil simulasi





Tugas Perorangan :

- Membuat Aplikasi LED Menyala dari kiri ke kanan
- Membuat Aplikasi Simulasi Seven Segmen
- Membuat Aplikasi Led dengan Tombol/Pus Buton
- Membuat Aplikasi Seven Segmen & Tombol/Pus Buton
- Membuat aplikasi Timer Counter
- Membuat Tampilan Tulisan di Layar LCD
- DII

NB : Pilih salah satu yang di kerjakan boleh sama jenis aplikasinya namun tidak boleh sama hasil aplikasi dan penulisan programnya



PERTEMUAN

10

**PRESENTASI TUGAS
KELOMPOK**



Tugas Kelompok

- Membuat aplikasi LED, Tombol /pus button dan sensor LDR
- Membuat aplikasi Seven Segmen, Tombol / pus button dan LED
- Membuat aplikasi Seven Segmen dengan Sensor Infra Red
- Membuat aplikasi Infra Red dengan Motor DC
- Membuat aplikasi Infra Red dengan Buzer

NB: Pembagian Tugas Kelompok Dosen yang mengajar membagikan



Bentuk Laporan

- Membuat Makalah dan Power point pada judul alat yang dibuat

Isi Laporan :

1. Teori Mikrokontroler dan teori komponen yang digunakan
 1. Prinsip kerja rangkaian yang di buat
 2. Listing program
 3. Penutup
- Mencantumkan DataSheet, Skema Rangkaian dan daftar komponen dan harga

NB: print out di kertas A4, min 10 s/d mak 20 lembar



Tahapan Presentasi:

1. Perwakilan Kelompok Menjelaskan Pembagian Tugas (Job Deskripsi)
2. Perwakilan Kelompok Menjelaskan Teknik Pembuatan Hardware (Rangkaian)
3. Perwakilan Kelompok Menjalankan listing Program IC Mikrokontroler Atmega Dengan Proteus Simulator.



Tahapan Presentasi:

5. Selesai Presentasi Anggota Kelompok Lain diberikan Kesempatan Untuk :
 - a. Mengajukan Pertanyaan
 - b. Memberikan Masukan
 - c. Memberikan Kritikan

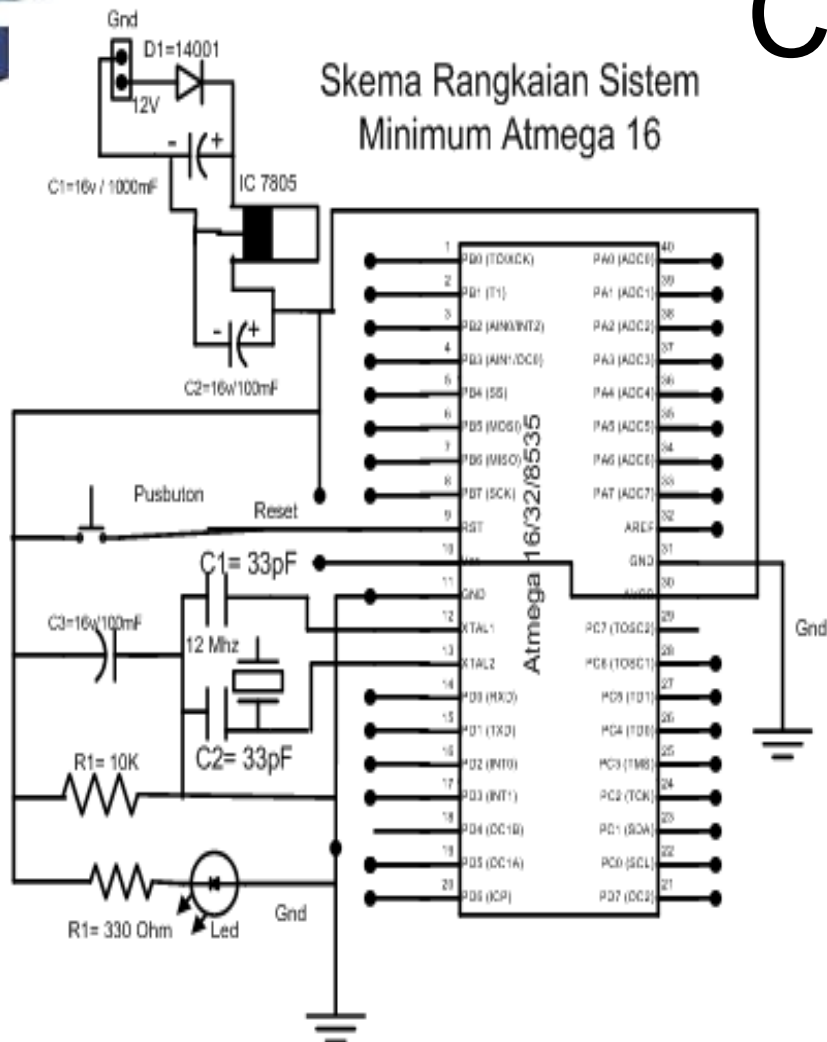


Keteria Penilaian

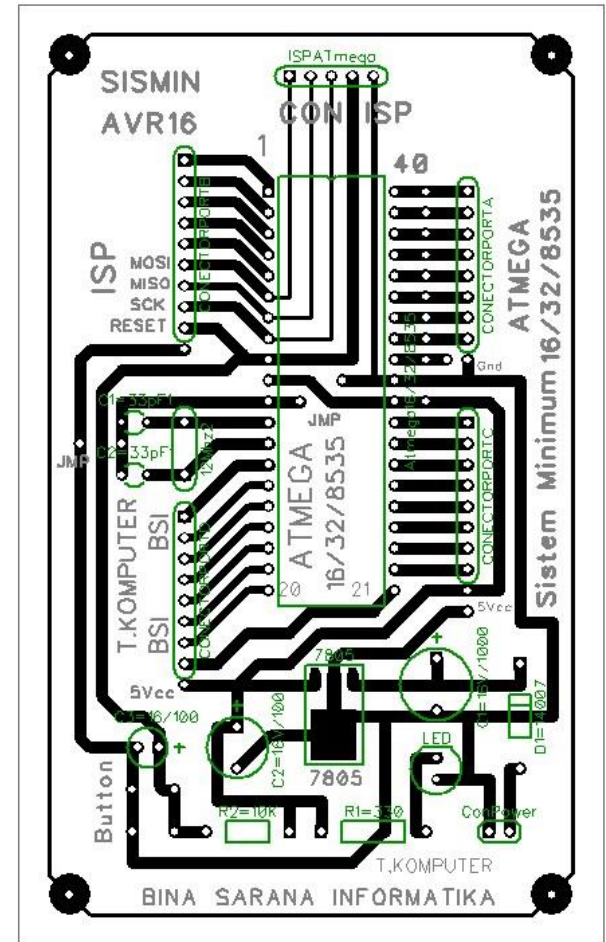
- Penulisan Laporan Makalah
- Bentuk Power Point
- Aplikasi yang digunakan / alat yang dibuat
- Keaktifan perorangan
- Tanya Jawab

NB : Bagi Mahasiswa yang tidak aktif hanya bergabung di kelompok diberi nilai rendah / tidak sama dengan mahasiswa yang aktif

Contoh Layout

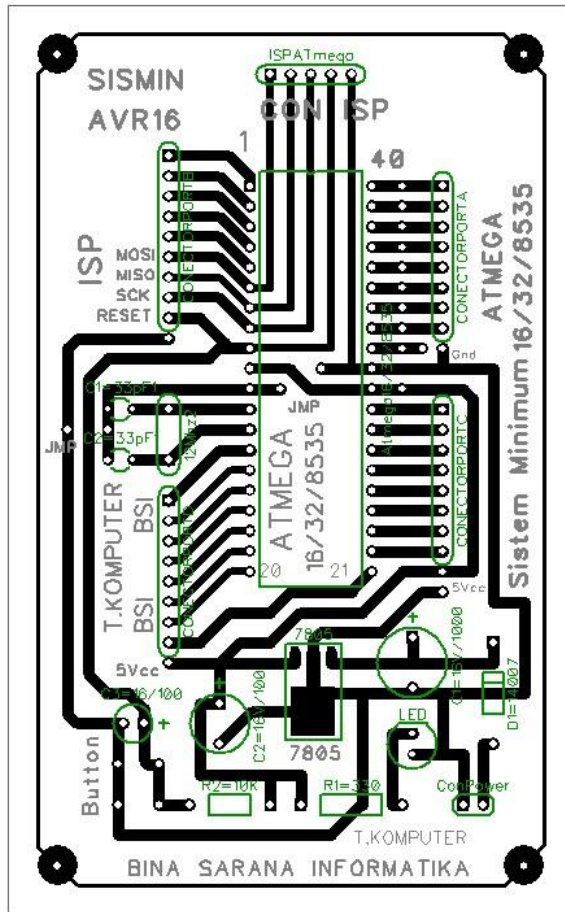


Skema Rangkaian Sistem Minimum ATmega



Layout ke PCB

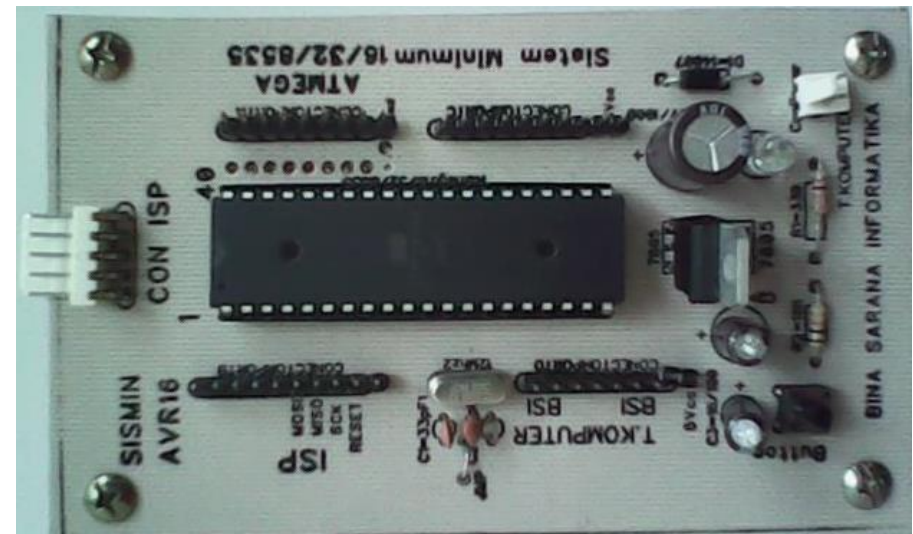
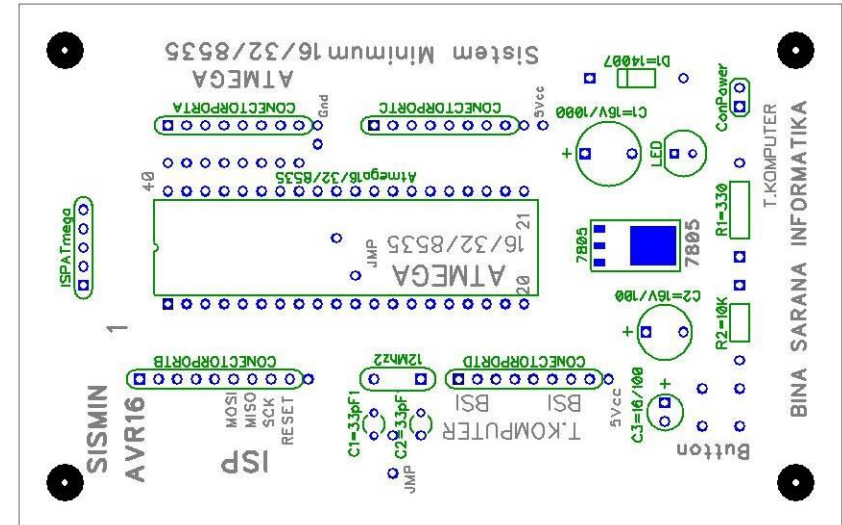
Contoh Layout



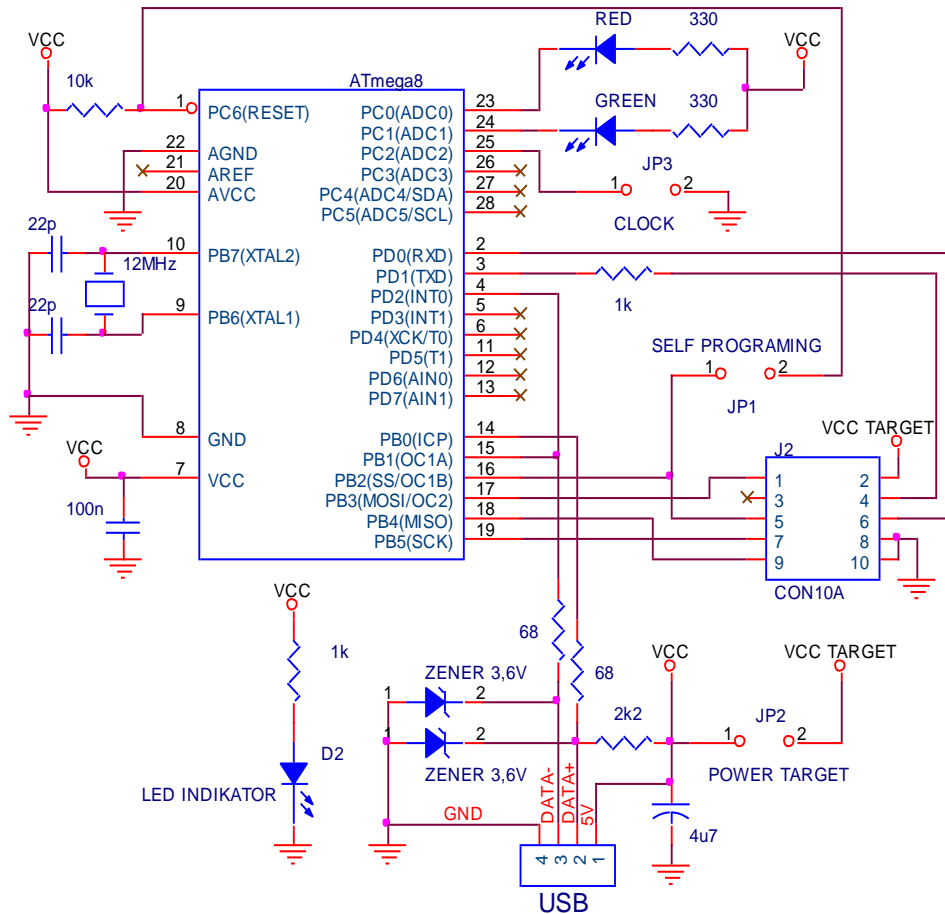
Jalur Rangkaian



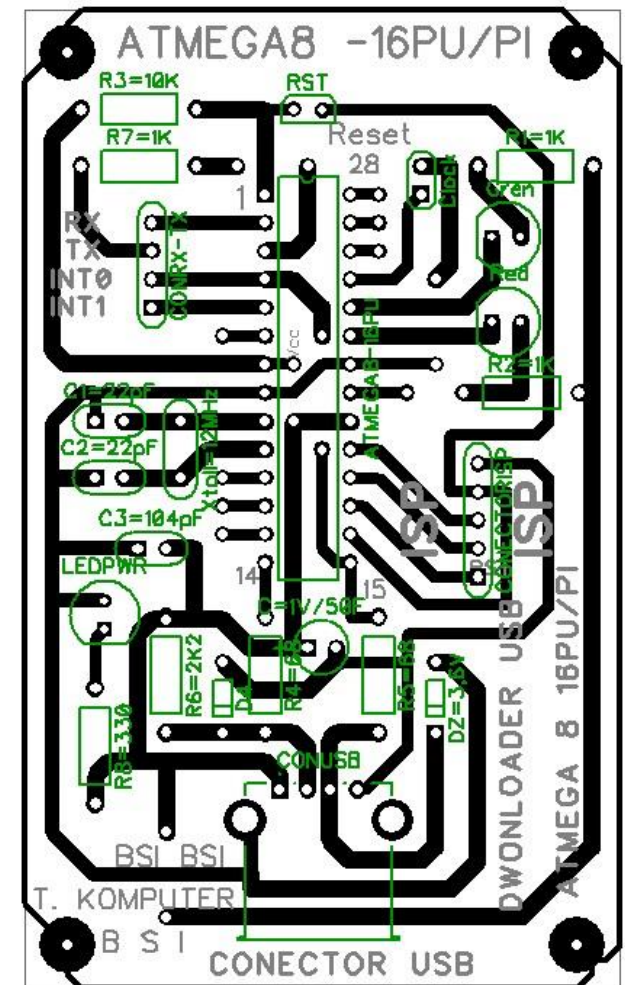
Tata Letak komponen



Contoh Layout

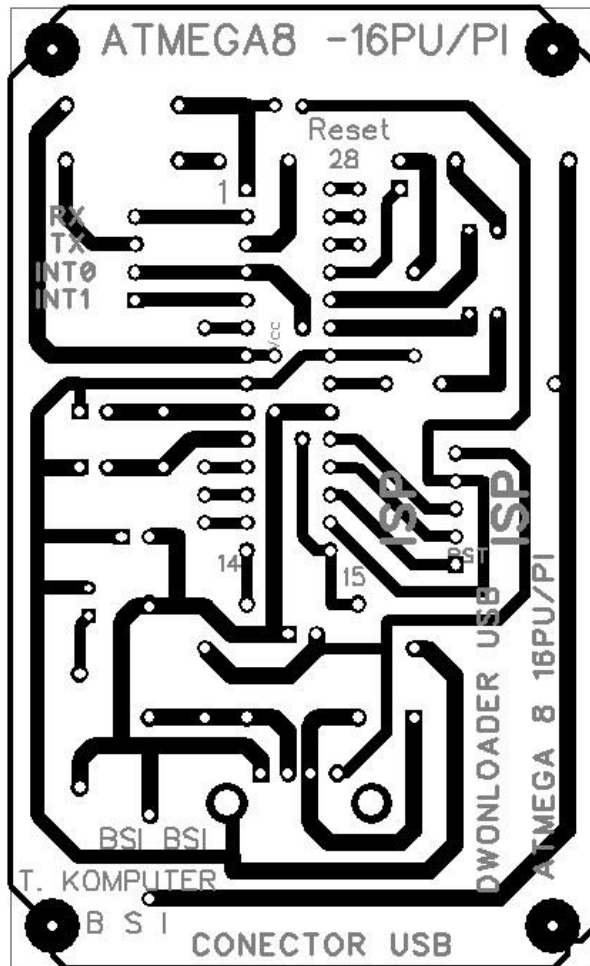


Skema Rangkaian
Downloader USB

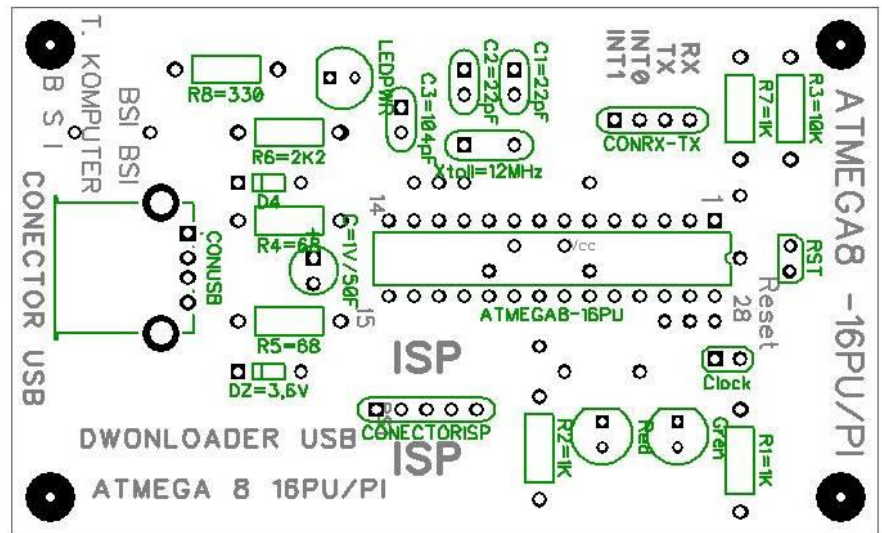


Layout ke PCB

Tata Letak komponen

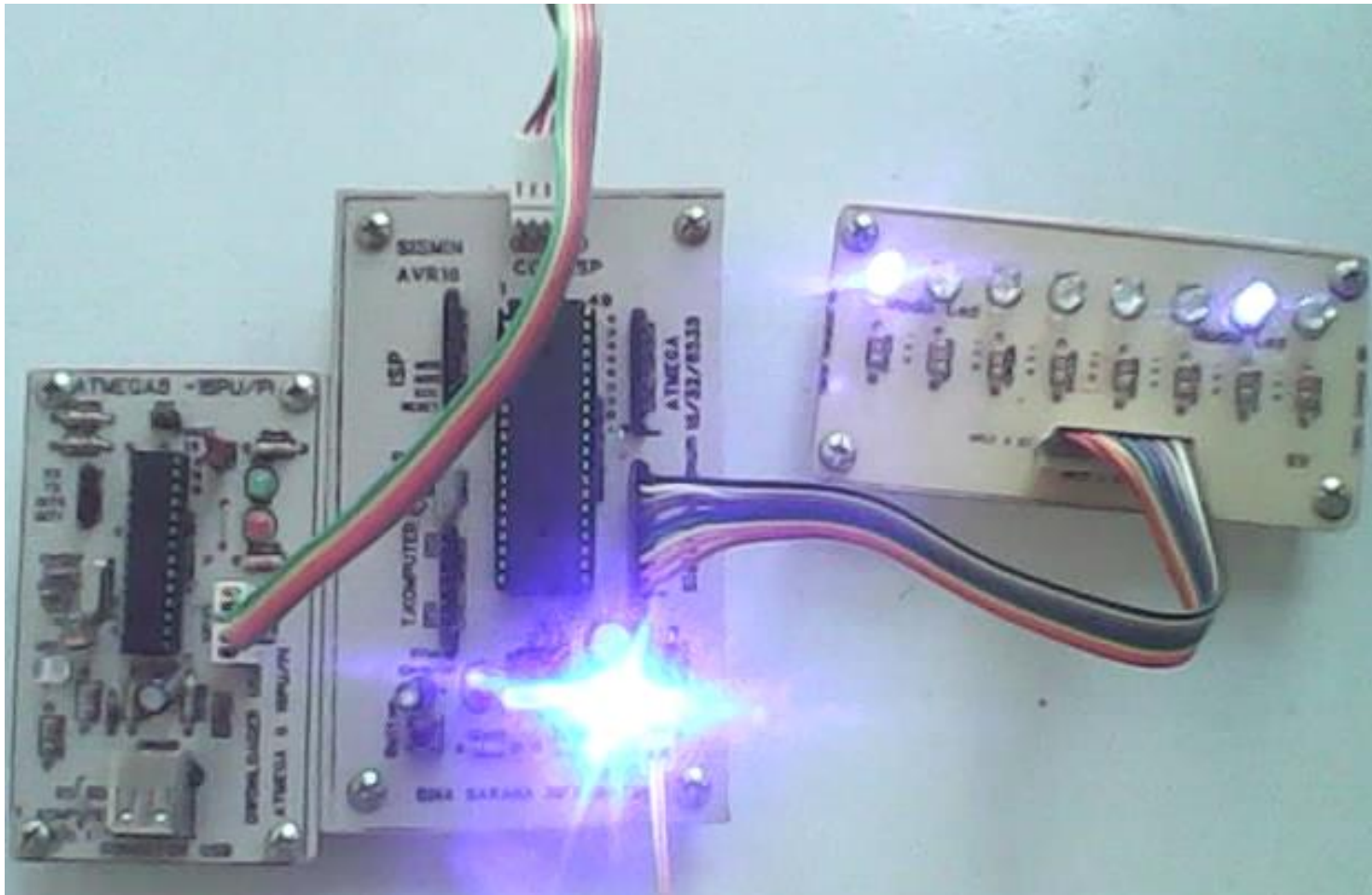


Jalur Rangkaian





Contoh Aplikasi Mikrokontroler



Blogger : winnerparluhutan.blogspot.com