

Dalam menjalankan setiap instruksi, kecepatan sebuah mikrokontroler ditentukan oleh frekuensi *clock* komponen kristal yang digunakan. Sinkronisasi antara frekuensi *clock* pada program dan frekuensi kristal sangatlah penting, mikrokontroler tidak akan dapat bekerja dengan baik sesuai instruksi apabila frekuensi *clock* tidak sinkron. Umumnya mikrokontroler menggunakan eksternal kristal, namun ada banyak juga mikrokontroler yang memiliki internal kristal. Salah satu contoh adalah mikrokontroler ATMEGA8535 yang dipakai untuk aplikasi ini. Mikrokontroler ATMEGA8535 pada setiap DT-AVR Low Cost Micro System telah diseting agar sesuai dengan program *tester*, yakni menggunakan kristal eksternal 4Mhz. Apabila mikrokontroler pada DT-AVR Low Cost Micro System hendak diganti dengan yang baru, maka diperlukan pengaturan *fusebits* nya untuk bekerja pada frekuensi kristal eksternal 4Mhz. Hal ini dikarenakan setiap mikrokontroler yang baru telah terprogram secara *default* menggunakan kristal internal 1Mhz. Aplikasi kali ini akan menjelaskan seting *fusebits* untuk pemilihan frekuensi *clock* dari eksternal kristal maupun internal kristal menggunakan CodeVisionAVR dan Bascom-AVR.

Modul yang diperlukan dalam aplikasi ini adalah:

- 1 DT-AVR Low Cost Micro System
- 1 DT-HiQ AVR USB ISP / DT-HiQ AVR In System Programming

Adapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 1
Blok Diagram AN177

Hubungan antara DT-AVR Low Cost Micro System dengan DT-HiQ AVR In System Programming dan DT-HiQ AVR USB ISP adalah sebagai berikut:

DT-AVR Low Cost Micro System (J16)	DT- HiQ AVR ISP	DT- HiQ AVR USB ISP
MOSI (Pin 1)	MOSI (Pin 1)	MOSI (Pin 1)
VCC (Pin 2)	VCC (Pin 2)	VTG (Pin 2)
LED (Pin 3)	LED (Pin 3)	LED (Pin 3)
GND (Pin 4)	GND(Pin 4)	GND(Pin 4)
RST AVR (Pin 5)	RST AVR (Pin 5)	Reset (Pin 5)
GND (Pin 6)	GND (Pin 6)	GND (Pin 6)
SCK (Pin 7)	SCK (Pin 7)	SCK (Pin 7)
GND (Pin 8)	GND (Pin 8)	GND (Pin 8)
MISO (Pin 9)	MISO (Pin 9)	MISO (Pin 9)
GND (Pin 10)	GND (Pin 10)	GND (Pin 10)

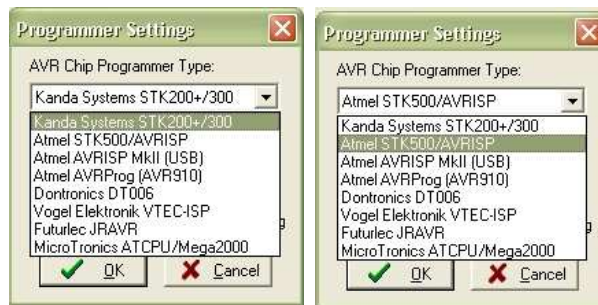
Tabel 1

Hubungan DT-AVR Low Cost Micro System Dengan DT-HiQ AVR In System Programming dan DT-HiQ USB AVR ISP

Setelah semua koneksi terhubung dengan benar, bukalah program CodeVisionAVR dan Bascom-AVR. Berikut akan dijelaskan bagaimana penggunaan dua program tersebut untuk seting *fusebits*.

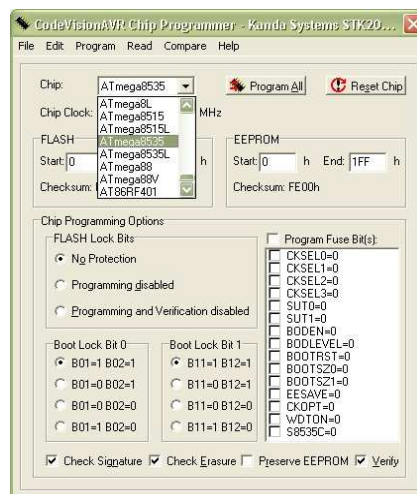
1. CodeVisionAVR

- Hal pertama yang dilakukan setelah program dibuka adalah melakukan *programmer setting*. Untuk masuk *programmer setting* klik **Settings|Programmer** pada *menu bar* CodeVisionAVR. Disini terdapat pilihan untuk memilih AVR Chip Programmer Type atau bisa dikatakan tipe koneksi yang digunakan untuk menghubungkan antara komputer dan DT-AVR Low Cost Micro System. Untuk DT-HiQ AVR In System Programming tipe yang digunakan adalah Kanda System STK200+/300 karena menggunakan antarmuka parallel. Sedangkan untuk DT-HiQ AVR USB In System Programming yang menggunakan antarmuka serial, tipe yang digunakan adalah Atmel STK500/AVRISP. Kemudian klik **OK**.



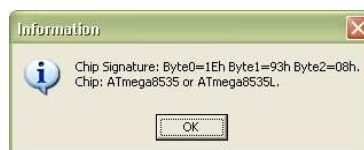
Gambar 2
AVR Chip Programmer Type

- Kemudian pilih **Tools|chip programmer** pada *menu bar* CodeVisionAVR atau tekan **Shift+F4** pada *keyboard* untuk menampilkan CodeVisionAVR Chip Programmer. Disini terdapat banyak pilihan untuk jenis mikrokontroler yang dipakai. Untuk DT-AVR Low Cost Micro System, mikrokontroler yang dipakai adalah ATMEGA8535. Pilih jenis mikrokontroler yang sesuai pada CodeVisionAVR Chip Programmer.



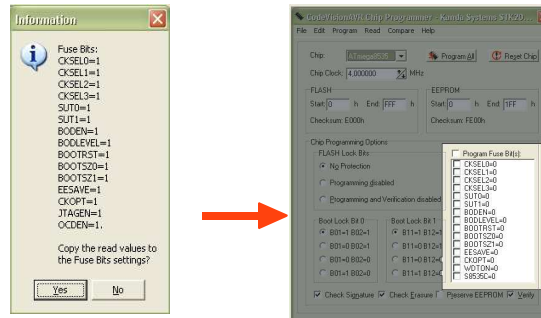
Gambar 3
CodeVisionAVR Chip Programmer

- Masih di CodeVisionAVR Chip Programmer, langkah selanjutnya adalah melakukan *checking* apakah mikrokontroler berhasil diakses atau tidak. Pada *menu bar* CodeVisionAVR Chip Programmer, klik **Read|Chip Signature**. Jika pembacaan mikrokontroler berhasil maka akan muncul informasi seperti gambar 4.



Gambar 4
Pembacaan Mikrokontroler

- Setelah pembacaan mikrokontroler berhasil, langkah selanjutnya adalah pembacaan *fusebits*. Untuk menampilkan *fusebit* klik **Read|Fuse Bit(s)** pada *menu bar CodeVisionAVR Chip Programmer*. Klik **Yes** untuk menyalinkan konfigurasi ke dalam menu pengaturan *fusebits*. Pilih **NO** untuk keluar dan tidak menyalinkan konfigurasi ke dalam menu pengaturan *fusebits*.



Gambar 5
Pembacaan Fusebits

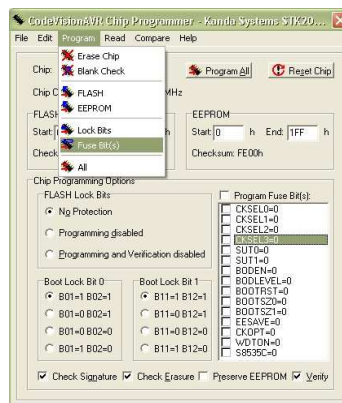
- Ada beberapa parameter yang perlu diperhatikan dalam pengaturan *fusebits*, diantaranya adalah CKSEL, SUT, BODEN, BODLEVEL, BOOTRST, BOOTSZ0, BOOTSZ1, EESAVE, CKOPT, JTAGEN dan OCDEN. Namun pengaturan *fusebits* untuk pemilihan jenis kristal eksternal atau kristal internal, hanya satu parameter yang dipakai, yaitu: CKSEL. Pengaturan nilai *fusebits* pada setiap mikrokontroler berbeda, nilai tersebut sesuai dengan *datasheet* masing-masing mikrokontroler. Untuk ATMEGA8535 nilai setiap bit pada CKSEL untuk pemilihan kristal dapat dilihat pada tabel 2.

Device Clocking Option	CKSEL3..0
External Crystal/Ceramic Resonator	1111 - 1010
External Low-frequency Crystal	1001
External RC Oscillator	1000 - 0101
Calibrated Internal RC Oscillator	0100 - 0001
External Clock	0000

Note: 1. For all fuses "1" means unprogrammed while "0" means programmed.

Tabel 2
Device Clocking Options

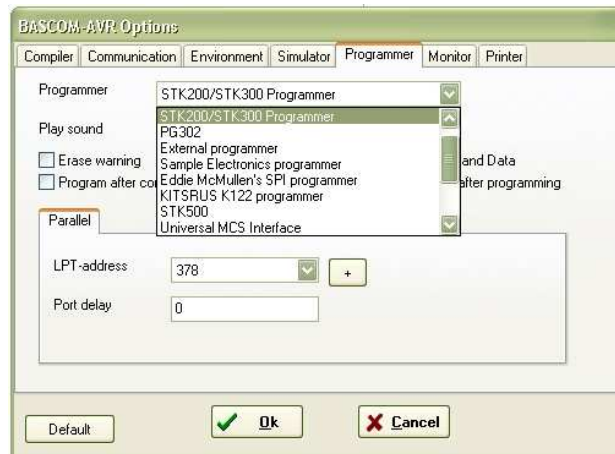
CKSEL memiliki empat bit data dalam bilangan biner, diantaranya: CKSEL3, CKSEL2, CKSEL1, CKSEL0. Nilai "1" pada *fusebit* menunjukkan bahwa *fusebit* tidak diprogram (*unprogrammed*) atau kalau dalam CodeVisionAVR berarti tidak dicentang. Sedangkan nilai "0" berarti diprogram (*programmed*). Sebagai contoh apabila akan menggunakan frekuensi kristal internal 1Mhz sebagai frekuensi kerja, maka nilai CKSEL3 =0; CKSEL2=0;CKSEL1=0; CKSEL =1. Untuk mengubah nilai CKSEL, lihat pada jendela CodeVisionAVR *Chip Programmer* sebelah kanan bawah. Beri tanda centang pada CKSEL3, CKSEL2 dan CKSEL1. Ini menunjukkan CKSEL tersebut terprogram dan bernilai "0". Kemudian klik **Program|Fusebit(s)** pada *menu bar CodeVisionAVR Chip Programmer* untuk menuliskan konfigurasi kedalam *fusebits* mikrokontroler.



Gambar 6
Pemrograman Fusebits

2. Bascom-AVR

- Pengaturan *fusebits* pada Bascom-AVR pada dasarnya tidak jauh berbeda dengan CodeVisionAVR, karena parameter yang digunakan dalam pengaturan sama. Perbedaannya hanya pada metode pemrogramannya. Setelah program Bascom-AVR dibuka, hal yang pertama perlu dilakukan adalah pemilihan tipe koneksi yang digunakan untuk menghubungkan komputer dan DT-AVR Low Cost Micro System. Caranya klik **Options| Programmer** pada *menu bar*. Pilih STK200/300 Programmer jika koneksi yang digunakan adalah DT-HiQ AVR In System Programming. Pilih Atmel STK500 jika koneksi yang digunakan adalah DT-HiQ AVR USB ISP.



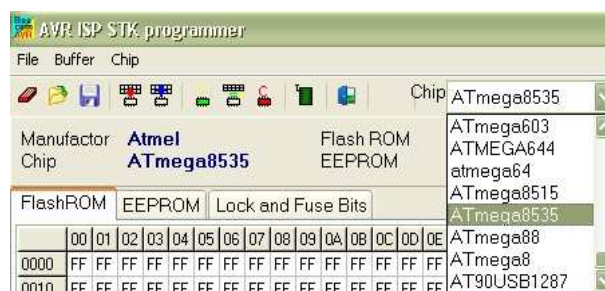
Gambar 7
Bascom-AVR Options

- Langkah berikutnya adalah memilih jenis mikrokontroler yang terdapat pada jendela AVR ISP STK Programmer. Untuk membuka jendelanya, klik ikon *programchip* seperti yang ditunjukkan oleh gambar 7 atau dengan menekan F4 pada *keyboard*.



Gambar 8
Program Chip

Kemudian akan muncul tampilan seperti gambar 9, pada kotak *Chip* pilih jenis mikrokontroler yang dipakai yaitu ATMEGA8535.



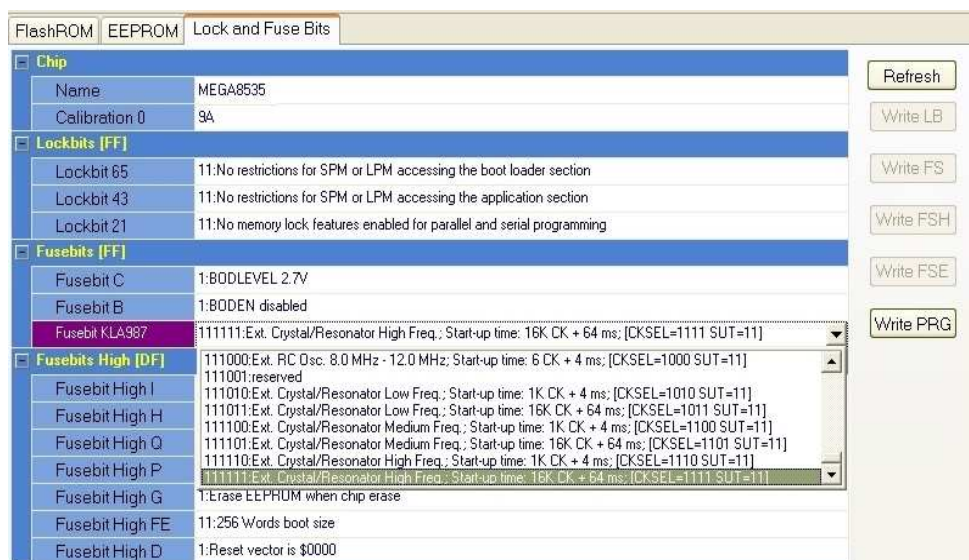
Gambar 9
AVR ISP STK

- Pilih menu **Chip|Identify** untuk menjalankan perintah identifikasi chip. Fungsi ini berguna untuk mengetahui jenis mikrokontroler dan untuk memastikan mikrokontroler dikenali oleh program. Jika identifikasi berhasil, maka akan muncul tampilan seperti pada gambar 10.



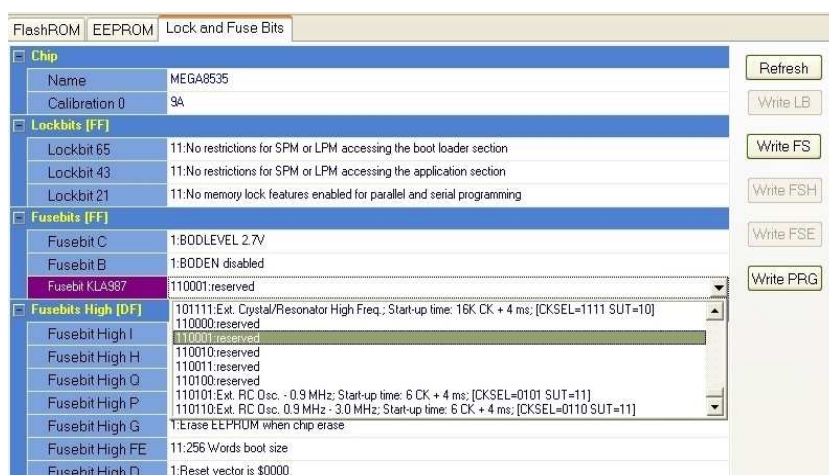
Gambar 10
Identifikasi Chip

- Untuk mengetahui nilai dari *fusebits*, klik tab **Lock And Fusebits**.



Gambar 11
Pangaturan Fusebits

Gambar 11 menunjukkan bahwa kristal yang dipakai adalah kristal eksternal. Untuk mengubahnya, klik pilihan yang ada dan sesuai dengan kebutuhan. Jika ingin menggunakan kristal internal 1MHz, salah satu pilihan bisa dilihat dalam gambar 12. Setelah seting selesai, klik tombol **Write FS** untuk menuliskan konfigurasi *fusebits* kedalam mikrokontroler.



Gambar 12
Pemrograman Fusebits

Selamat berinovasi!

All trademarks, trade names, company names, and product names are the property of their respective owners. All softwares are copyright by their respective software publishers and/or creators.