

Programiranje AVR mikrokontrolera u razvojnom okruženju Eclipse IDE

Integrirano razvojno okruženje Eclipse IDE predstavlja moćan i savremen alat koji se koristi za razvoj aplikacija namenjenih izvršavanju na velikom broju različitih platformi. Sve većoj popularnosti ovog okruženja doprinosi veliki broj postojećih plug-in dodataka koji omogućavaju povezivanje sa različitim razvojnim softverskim modulima. Među mnoštvom podržanih opcija, omogućeno je programiranje AVR familije mikrokontrolera u programskom jeziku C. U ovom dokumentu biće ukratko opisan postupak instalacije i podešavanja softverskih modula koji su neophodni za programiranje AVR mikrokontrolera ATmega328p koji se nalazi na Arduino UNO ploči.

Svi alati koji će biti korišćeni u nastavku su open-source i omogućeno je njihovo besplatno preuzimanje od strane korisnika. Pošto je Eclipse cross-platform okruženje koje je razvijano u jeziku Java, za njegovo funkcionisanje je neophodna postojeća instalacija Jave. Ovde će biti opisan postupak instalacije i korišćenja Eclipse IDE pod operativnim sistemom Windows.

Preuzimanje i instalacija Eclipse IDE

Preuzimanje Eclipse razvojnog okruženja, obavlja se preko web stranice:

www.eclipse.org-downloads

Među ponuđenim verzijama, potrebno je odabrati “**Eclipse IDE for C/C++ developers**”. U zavisnosti od platforme, bira se 32-bitna ili 64-bitna verzija za odgovarajući operativni sistem. Nakon otpakivanja preuzete arhive, pokreće se izvršna verzija programa **Eclipse** koja se nalazi u otpakovanom direktorijumu.

Pokretanjem okruženja Eclipse, prvo se vrši izbor direktorijuma u kojem će biti smešteni svi izvorni kodovi i ostale datoteke koje sačinjavaju strukturu projekata koje korisnik bude razvijao. Takav direktorijum naziva se **Workspace** (radni prostor). Ukoliko više korisnika koristi istu mašinu, preporučljivo je da svaki korisnik definiše za sebe zaseban **Workspace** folder, kako bi se izbegle neželjene situacije koje bi mogle nastati usled korišćenja zajedničkog radnog prostora.

Instalacija AVR-GCC i AVR Dude

AVR-GCC je kompajler koji se koristi za prevođenje programa za AVR mikrokontrolere napisanih u programskom jeziku C. **AVR Dude** je pomoćni alat koji komunicira sa hardverskim programatorom i posredstvom kojeg se prevedeni program upisuje u programsku FLASH memoriju mikrokontrolera.

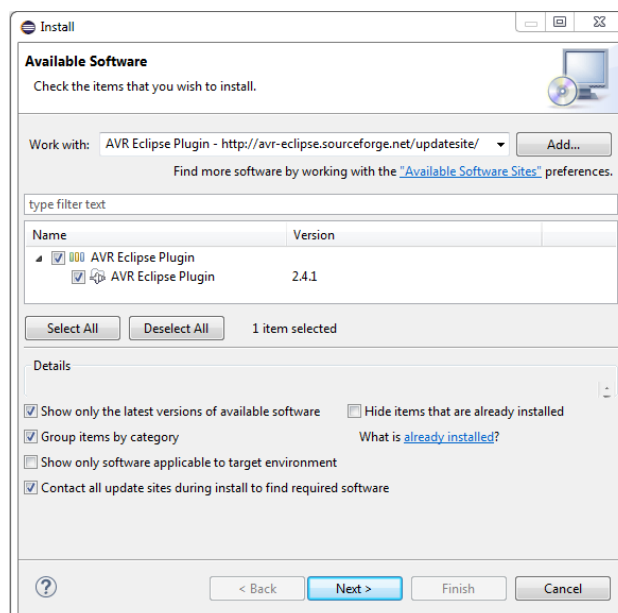
Pod operativnim sistemom Windows, oba ova alata se instaliraju u okviru programskog paketa **WinAVR**, koji je moguće besplatno preuzeti preko web stranice:

winavr.sourceforge.net/download.html

Instalacija AVR dodatka za Eclipse

Dodatak za AVR mikrokontrolere (**AVR plugin**) omogućava i u znatnoj meri olakšava rad sa ovom familijom mikrokontrolera u okviru razvojnog okruženja Eclipse IDE. Njegovom instalacijom omogućeno je kreiranje, kompajliranje i učitavanje C projekta koji će se izvršavati na željenom AVR mikrokontroleru.

Instalacija ovog dodatka vrši se izborom opcije **Help -> Install New Software** u okviru Eclipse okruženja. Po otvaranju prozora čiji izgled je prikazan na slici, u polju **Work with** potrebno je upisati link <http://avr-eclipse.sourceforge.net/updatesite/> i kliknuti **Add**. Po iskakanju prozora **Add Repository**, u polju **Name** dodeljuje se ime AVR dodatku, npr. **AVR Eclipse Plugin**.

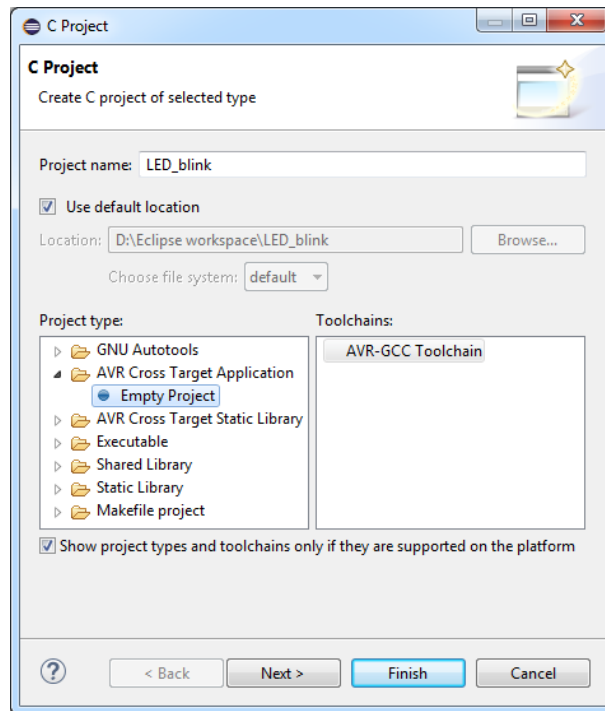


Nakon pojavljivanja opcije **AVR Eclipse Plugin**, potrebno je potvrditi odabir opcije klikom na kvadrat koji se nalazi sa leve strane, nakon čega sledi pritisak na taster **Next**. Praćenjem i potvrdom opcija koje će se pojavljivati u narednim prozorima, stiže se do opcije **Finish**, čime se okončava instalacija AVR dodatka. Da bi dodatak bio aktiviran, potrebno je restartovati Eclipse okruženje.

Instalacijom okruženja **Eclipse IDE** sa dodatkom **AVR plugin** i **WinAVR** programskog paketa, korisnik ima na raspolaganju sve softverske alate koji su mu potrebni za programiranje AVR mikrokontrolera u programskom jeziku C.

Kreiranje projekta

Pošto je razvojno okruženje instalirano i podešeno, naredni korak je kreiranje projekta. U okviru projekta, nalaziće se sve potrebne datoteke i podešavanja vezana za aplikaciju koja se razvija. U svrhu demonstracije, biće kreiran projekat za aplikaciju koja upravlja LED diodom tako što je naizmenično uključuje i isključuje.



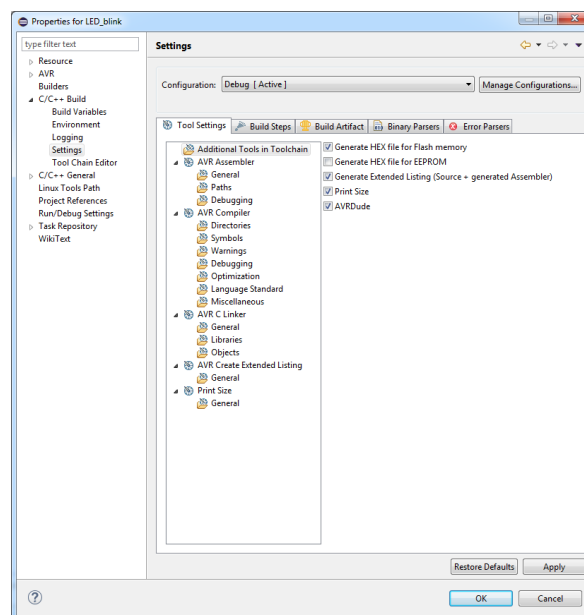
Novi prazan projekat kreira se izborom opcije **File -> New -> C Project**, nakon čega se projektu dodeljuje ime (npr. *LED_blink*). U polju **Project type** potrebno je izabrati opciju **AVR Cross Target Application/Empty project** i kliknuti **Next**. U sledećem prozoru potrebno je izostaviti opciju

Debug (za sada neće biti korišćena) i ostaviti izabranu samo opciju **Release**, nakon čega ponovo sledi **Next**. Naredni prozor omogućava izbor mikrokontrolera. U polju **MCU Type** treba izabrati **ATmega328P**, a u polje **MCU Frequency(Hz)** upisuje se vrednost radne frekvencije takta **16000000** (pošto je takt oscilatora na Arduino Uno ploči frekvencije 16Mhz). Klikom na taster **Finish**, projekat će biti kreiran u okviru radnog prostora (Workspace) izabranog prilikom pokretanja Eclipse IDE.

Podešavanje kompajlera i programatora

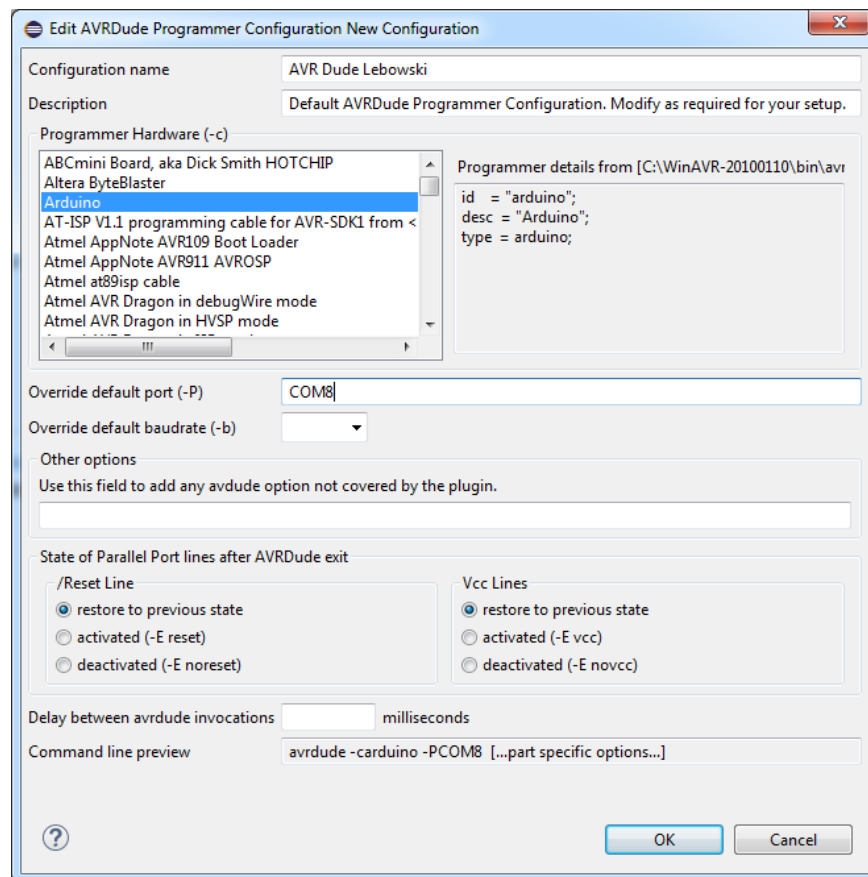
Ako su prethodni koraci uspešno obavljeni, unutar prozora **Project Explorer** pojaviće se projekat pod nazivom koji je definisao korisnik. Po kreiranju projekta, neophodno je još dodatno podesiti opcije u vezi sa kompajlerom. Ovim podešavanjima moguće je pristupiti desnim klikom na naziv projekta u okviru **Project Explorer**-a, a zatim izborom opcije **Properties** (na dnu liste). U prozoru koji potom iskoči bira se opcija **C/C++ Build -> Settings**. U okviru taba **Tool Settings**, među ponuđenim opcijama potrebno je izabrati :

- **Generate Hex File for Flash Memory** – ovim se uključuje kreiranje HEX datoteke koja sadrži kompajlirani (mašinski) kod, koji se upisuje u programsku FLASH memoriju mikrokontrolera
- **Print Size** – Nakon kompajliranja biće prikazana veličina programskog koda, odnosno procenat zauzeća programske (FLASH) i operativne (RAM) memorije
- **AVR Dude** – Za programiranje mikrokontrolera koristi se eksterna aplikacija AVR Dude



Preostale dve opcije mogu, ali i ne moraju biti izabrane, pošto nisu neophodne. Izbor se potvrđuje klikom na **Apply**.

Naredni korak je podešavanje programatora. U projektnim podešavanjima treba izabrati opciju **AVR -> AVR Dude**. Inicijalno ne postoji nijedna programatorska konfiguracija, pa je potrebno dodati novu klikom na taster **New**. U polju **Configuration name** zadaje se naziv nove konfiguracije (npr. **AVR Dude Lebowski**). Dalje, u polju **Programmer Hardware** potrebno je izabrati opciju **Arduino**, a u polju **Override default port (-P)** navodi se oznaka virtualnog serijskog porta koji je Windows enumerisao prilikom priključenja Arduino ploče na USB port (npr. **COM8**). Ostale opcije ostaju nepromenjene, nakon čega se pritiskom na **OK** potvrđuje konfiguracija programatora, a zatim se pritiskom na **Apply** potvrđuje njegov izbor.



Nakon ovih koraka, projekat je spreman za upotrebu. Finalnu proveru konfiguracije moguće je izvršiti pomoću opcije **AVR -> Target Hardware**, u okviru projektnih podešavanja. Ukoliko je sve podešeno kako treba, klikom na taster **Load From MCU**, bez obzira na to koji je mikrokontroler prethodno bio izabran prilikom kreiranja projekta, u polju **MCU Type** će se pojaviti **ATmega328P**, što je korektna oznaka mikrokontrolera koji se nalazi na Arduino UNO ploči.

Dodavanje izvornog koda

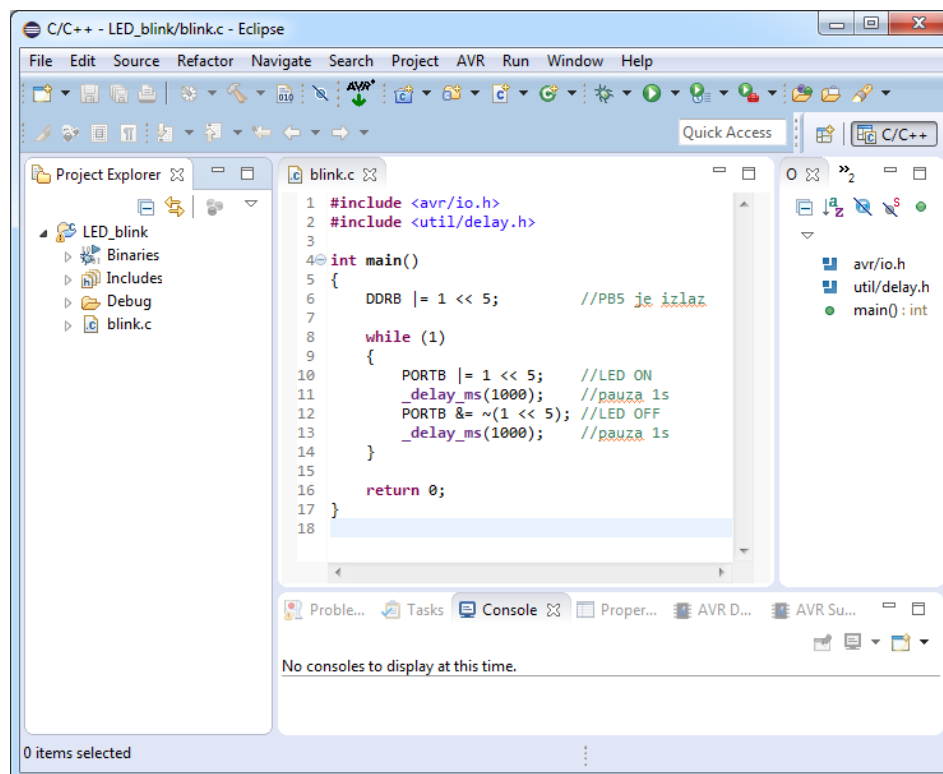
Kada je projekat uspešno kreiran i konfigurisan, datoteke koje sadrže izvorni kod programa se jednostavno dodaju pomoću opcija **New -> Source File** i **New -> Header File**. U primeru prikazanom na slici dodata je datoteka **blink.c**, u koju je dodat programski kod za upravljanje LED diodom koja se nalazi na Arduino ploči. Ukoliko je programski kod raspoređen na više datoteka, neophodno je da se u jednoj od njih nalazi implementacija funkcije *main*:

```
#include <avr/io.h>
#include <util/delay.h>


int main()
{
    DDRB |= 1 << 5;           //PB5 je izlaz

    while (1)
    {
        PORTB |= 1 << 5;      //LED ON
        _delay_ms(1000);      //pauza 1s
        PORTB &= ~(1 << 5);  //LED OFF
        _delay_ms(1000);      //pauza 1s
    }

    return 0;
}
```



Kompajliranje izvornog koda i programiranje mikrokontrolera

Nakon što je programski kod u programskom jeziku C dodat u okviru postojećeg projekta, potrebno je izvršiti njegovo kompajliranje (prevođenje) u mašinski kod. Proces kompajliranja započinje izborom opcije **Project -> Build All**¹, ili klikom na ikonicu . Ukoliko program ne sadrži sintaksne greške i ima odgovarajuću strukturu, krajnji rezultat kompajliranja je izvršna verzija programa u obliku datoteke sa ekstenzijom **.hex**. Ova datoteka sadrži niz mašinskih instrukcija koje se smeštaju u programsku FLASH memoriju posredstvom aplikacije *AVR Dude* i pomoćnog programa (*bootloader-a*), koji se nalazi unutar samog kontrolera.

¹ Korisno je uključiti opciju kojom se pre kompajliranja sve izvorne datoteke koje su menjane od poslednjeg kompajliranja automatski snimaju u projektni direktorijum (**Window -> Preferences -> General -> Workspace -> Save automatically before build**).