

Poglavlje 1

Programiranje AVR mikrokontrolera u razvojnom okruženju *Eclipse*

1.1 Uvod

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 dodataka (eng. *plugins*) koji omogućavaju povezivanje sa različitim razvojnim softverskim modulima. Među mnoštvom podržanih opcija, omogućeno je programiranje AVR familije mikrokontrolera u programskom jeziku C. U okviru ovog poglavlja biće ukratko opisan postupak instalacije i podešavanja softverskih modula koji su neophodni za programiranje AVR mikrokontrolera *ATmega328P* koji se nalazi na *Arduino UNO* platformi.

Svi alati koji će biti korišćeni u nastavku su alati otvorenog koda (eng. *open-source*) i omogućeno je njihovo besplatno preuzimanje od strane korisnika. Budući da je *Eclipse* zapravo *cross-platform* okruženje koje je razvijano u programskom jeziku Java, za njegovo funkcionisanje je neophodna postojeća instalacija Jave. U okviru ovog poglavlja će biti opisan postupak instalacije i korišćenja *Eclipse* razvojnog okruženja pod operativnim sistemom *Windows*. Vrlo sličnu proceduru je potrebno sprovesti i u slučaju nekog drugog operativnog sistema.

1.2 Preuzimanje i instalacija Eclipse IDE

Preuzimanje Eclipse razvojnog okruženja je moguće izvršiti putem sledećeg linka:

<https://www.eclipse.org/downloads/packages/>

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 odgo-

varajući operativni sistem. Nakon otpakivanja preuzete arhive, pokreće se izvršna verzija programa *Eclipse* koja se nalazi u otpakovanom direktorijumu.

Pokretanjem razvojnog okruženja, 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 se naziva *Workspace*, odnosno radni prostor. Ukoliko više korisnika koristi istu mašinu, preporučljivo je da svaki korisnik definiše zaseban *Workspace* folder, kako bi se izbegle neželjene situacije koje bi mogle nastati usled korišćenja zajedničkog radnog prostora.

1.3 Instalacija AVR-GCC i AVR Dude

AVR-GCC predstavlja skup razvojnih alata (eng. *toolchain*) koji se koriste za prevođenje programa za AVR mikrokontrolere napisanih u programskom jeziku C. Sa druge strane, *AVR Dude* je pomoćni alat koji komunicira sa hardverskim programatorom i posredstvom kog se prevedeni program upisuje u programsku (eng. *flash*) memoriju mikrokontrolera. Detaljno objašnjenje ovih i ostalih alata u okviru *AVR-GCC toolchain*-a dato je u Poglavlju 12.

Pod operativnim sistemom *Windows*, oba alata se instaliraju u okviru programskog paketa *WinAVR*, koji je moguće besplatno preuzeti putem narednog linka:

winavr.sourceforge.net/download.html

Napomena: Ukoliko se koristi *Windows 10* operativni sistem potrebno je, pored navedene instalacije, izvršiti dodatne korake koji podrazumevaju sledeće:

1. u instalacionom folderu na adresi `\WinAVR-20100110\utils\bin` je potrebno locirati datoteku *msys-1.0.dll*;
2. preuzeti novu, zamensku datoteku na sledećem linku:

<http://www.madwizard.org/download/electronics/msys-1.0-vista64.zip>

3. zameniti postojeću datoteku sa preuzetom datotekom.

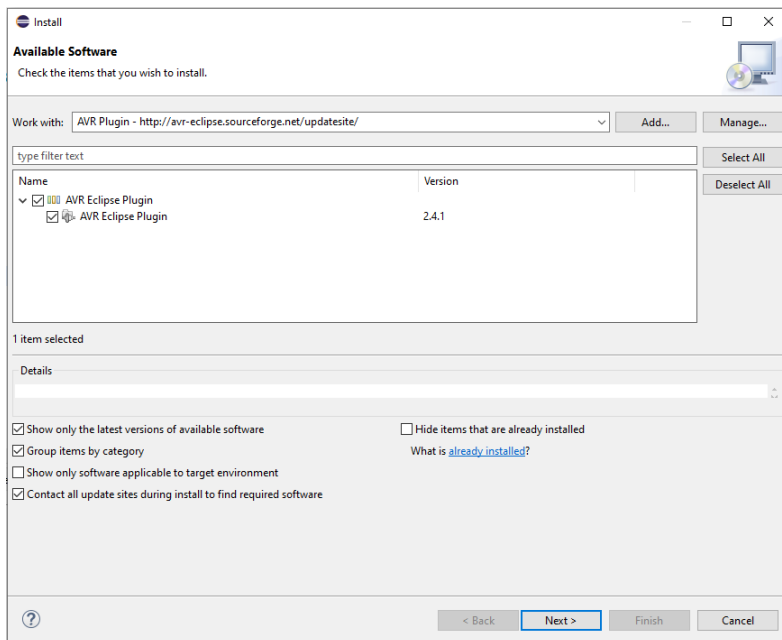
1.4 Instalacija AVR dodatka za Eclipse

Dodatak za AVR mikrokontrolere (*AVR plugin*) omogućava i u znatnoj meri olakšava rad sa ovom familijom mikrokontrolera u okviru *Eclipse* razvojnog okruženja. 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 naredni 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*.



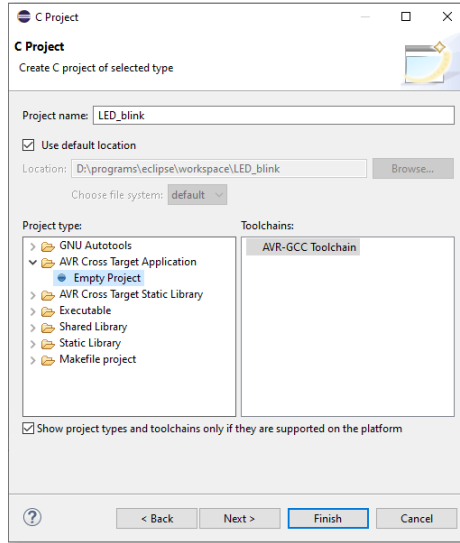
Slika 1.1: Instalacija AVR dodatka za Eclipse

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 pojavljiviti u narednim prozorima, stiže se do opcije *Finish*, čime se okončava instalacija AVR dodatka. Kako bi dodatak bio aktiviran, potrebno je restartovati Eclipse okruženje.

Instalacijom razvojnog okruženja Eclipse 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.

1.5 Kreiranje projekta

Nakon što je razvojno okruženje instalirano i podešeno, naredni korak podrazumeva kreiranje projekta. U okviru projekta, nalaziće se sve potrebne datoteke i podešavanja u vezi sa aplikacijom koja se razvija. U cilju demonstracije, biće kreiran projekat za aplikaciju koja upravlja ugrađenom diodom na Arduino ploči, tako što je naizmenično uključuje i isključuje u odgovarajućem vremenskom intervalu.



Slika 1.2: Kreiranje projekta

Novi, prazan projekat kreira se izborom opcije *File* → *New* → *Project* → *C Project*, nakon čega se projektu dodeljuje ime (na primer LED_blink). U polju *Project type* potrebno je izabrati opciju *AVR Cross Target Application/Empty project* i kliknuti *Next*. Zatim, u sledećem prozoru, potrebno je izostaviti opciju *Debug* i ostaviti izabranu samo opciju *Release*, nakon čega ponovo sledi pritisak na *Next*. Naredni prozor omogućava izbor mikrokontrolera. U polju *MCU Type* treba izabrati *AT-mega328P*, a u polje *MCU Frequency(Hz)* upisuje se vrednost radne frekvencije takta koja iznosi 16000000, budući da je takt oscilatora na Arduino Uno platformi frekvencije 16MHz. Pritiskom na taster *Finish*, projekat će biti kreiran u okviru radnog prostora izabranog prilikom pokretanja Eclipse razvojnog okruženja.

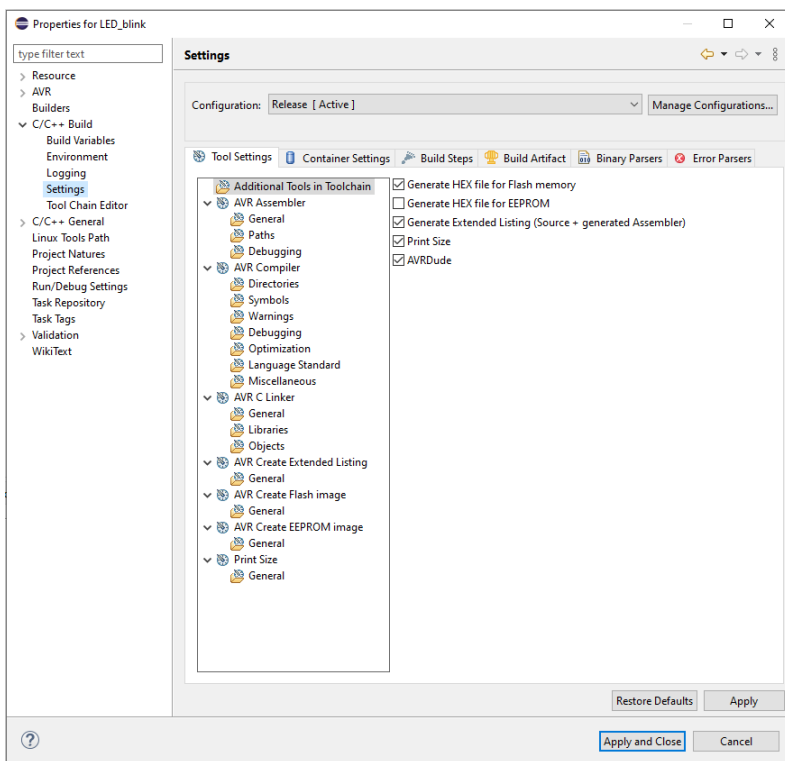
1.6 Podešavanje kompajlera i programatora

Ukoliko su prethodni koraci uspešno izvršeni, unutar prozora *Project Explorer* pojaviće se projekat pod nazivom koji je definisao korisnik. Po kreiranju projekta, neophodno je, dodatno, podesiti opcije u vezi sa kompajlerom. Ovim podešavanjima moguće je pristupiti desnim klikom na naziv projekta u okviru *Project Explorer* prozora, a zatim izborom opcije *Properties* (na dnu liste). U prozoru koji se potom pojavljuje, potrebno je izabrati opciju *C/C++ Build* → *Settings*. U okviru kartice *Tool Settings*, među ponuđenim opcijama, potrebno je izabrati sledeće stavke:

- **Generate Hex File for Flash Memory** – čime se uključuje kreiranje *HEX* datoteke koja sadrži kompajlirani (mašinski) kod, koji se upisuje u

programsku (eng. *flash*) memoriju mikrokontrolera;

- **Print Size** – čime se nakon kompajliranja prikazuje veličina programskog koda, odnosno procenat zauzeća programske i operativne memorije;
- **AVR Dude** – čime se za programiranje mikrokontrolera koristi se prethodno instalirana, eksterna aplikacija *AVR Dude*.

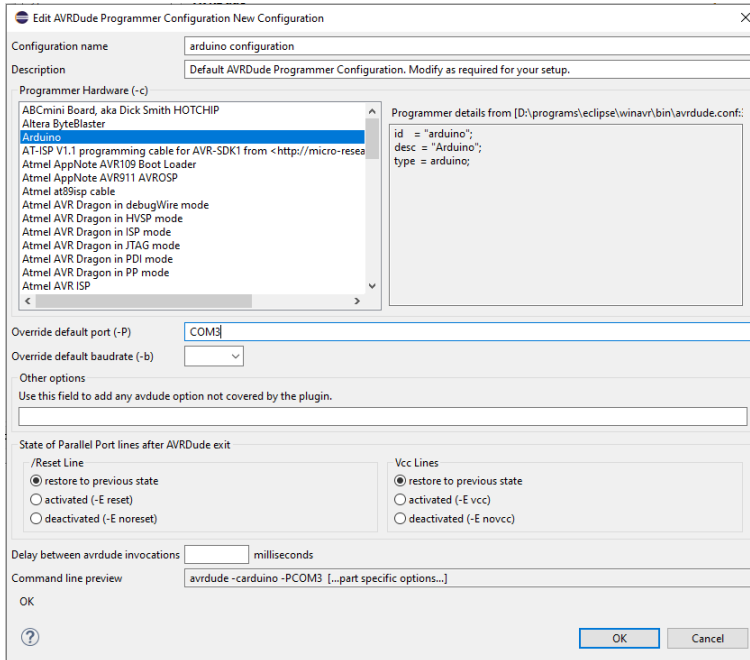


Slika 1.3: Podešavanje projekta

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

Naredni korak podrazumeva podešavanje programatora. U projektnim podešavanjima treba izabrati opciju *AVR* → *AVR Dude*. Inicijalno ne postoji nijedna programatorska konfiguracija, pa je potrebno dodati novu pritiskom na taster *New*. U polju *Configuration name* zadaje se naziv nove konfiguracije (npr. *AVR Dude*). Dalje, u polju *Programmer Hardware* potrebno je izabrati opciju *Arduino*, a u polju *Override default port (-P)* navesti oznaku virtuelnog 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.



Slika 1.4: Podešavanje programatora

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 na ispravan način, pritiskom 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.

1.7 Dodavanje izvornog koda

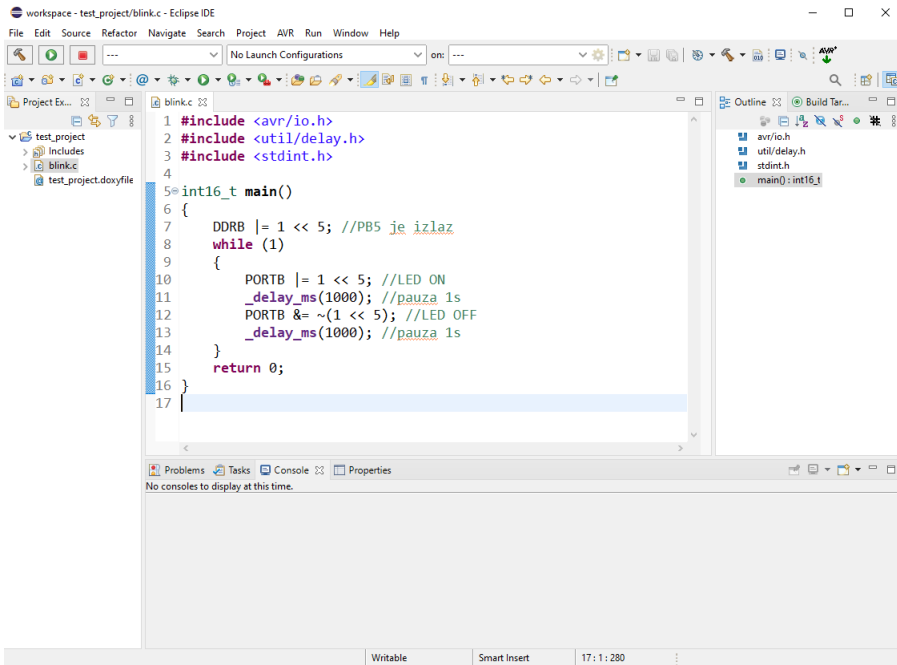
Kada je projekat uspešno kreiran i konfigurisan, datoteke koje sadrže izvorni kod programa se jednostavno dodaju pomoću opcija *New* → *Source File* i *New* → *Header File*. U primeru, prikazanom na slici 1.5, u projekat je uključena datoteka *blink.c*, prikazana na listingu 1.1. U okviru ove datoteke je dodat kod za upravljanje diodom koja se nalazi na Arduino ploči. U slučaju da je 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>
#include <stdint.h>

int16_t 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;
}
```

Listing 1.1: Izvorna datoteka blink.c



Slika 1.5: Dodavanje izvornog koda

1.8 Kompajliranje izvornog koda i programiranje mikrokontrolera

Nakon što je kod napisan 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*, ili pritiskom na ikonicu "čekića". 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 formatiranih na odgovarajući način, koje se smeštaju u programsku memoriju posredstvom aplikacije *AVR Dude* i pomoćnog programa (*bootloader-a*), koji se nalazi unutar samog mikrokontrolera.