# StartUSB for AVR™

## Manual

## Development System

*All MikroElektronika´s development systems represent irreplaceable tools for programming and developing microcontroller-based devices. Carefully chosen components and the use of machines of the last generation for mounting and testing thereof are the best guarantee of high reliability of our devices. Due to simple design, a large number of add-on modules and ready to use examples, all our users, regardless of their experience, have the possibility to develop their projects in a fast and efficient way.*

MikroElektronika
SOFTWARE AND HARDWARE SOLUTIONS FOR EMBEDDED WORLD ...making it simple

# StartUSB for AVR

The StartUSB for AVR is a miniature development system that enables you to experiment with the AT90USB162 microcontroller.

### Key features:

- Bootloader;
- Proto board;
- USB support;
- 5V power supply voltage.

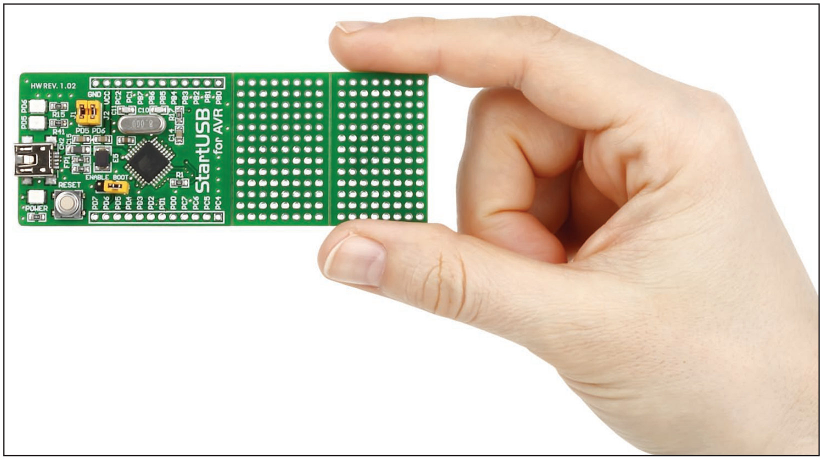


**Figure 1:** StartUSB for AVR development system

### How to connect the development system?

The StartUSB for AVR development system can be easily connected to a PC via a USB connector CN2, Figure 3. The microcontroller pins can be connected to a device via 1x12 connectors CN3 and CN4. An LED labelled POWER is used to indicate whether the board is turned on or off, whereas LEDs PD5 and PD6 can be configured to indicate the logic state on the PD5 and PD6 pins. In order to enable programming via bootloader place jumper J3 (ENABLE BOOT) on the board.

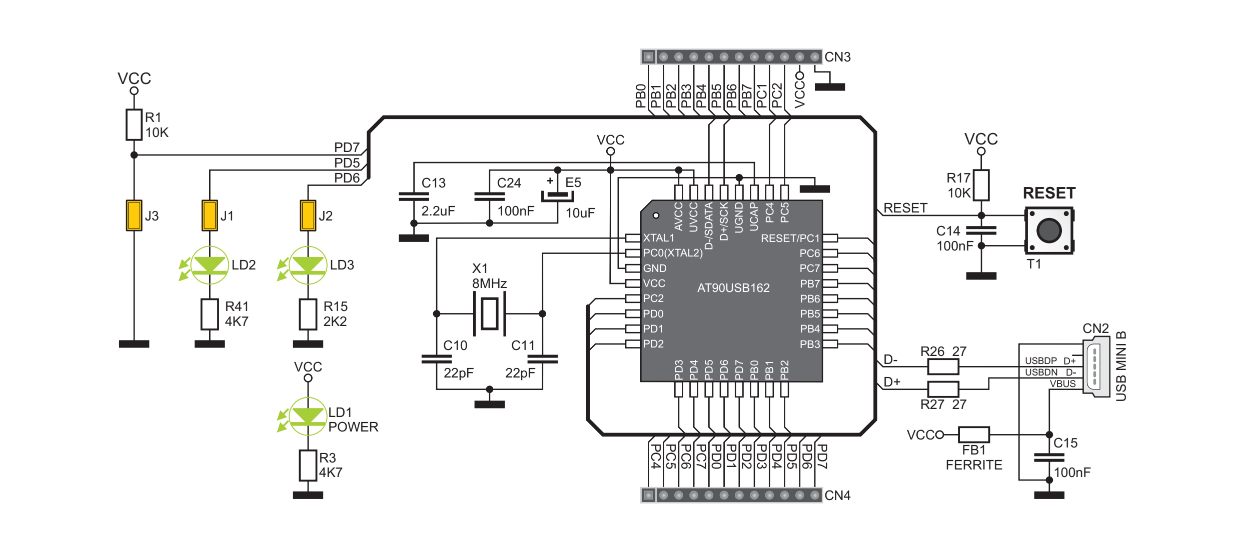


**Figure 2**: The StartUSB for AVR development system connection schematic

## How to program the microcontroller?

A .hex code is loaded into the microcontroller via the bootloader program stored in the microcontroller memory. The **Atmel FLIP** application is used to transfer the .hex code from the PC to the microcontroller.

Follow the steps below in order to program the microcontroller properly:

## STEP 1: Install Atmel FLIP

Download the **Atmel FLIP** program from Atmel's website at:
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=3886 (FLIP 3.4.2 for Windows (Java Runtime Environment included))

After download is completed double click on the FLIP installation icon

JRE_Flip_Installer_3.4.2
Atmel

Follow on-screen instructions and install the program on your PC.

## **STEP 2:** Connect StartUSB for AVR to your PC

Attach StartUSB for AVR to a PC via a USB cable, Figure 3. A PC will automatically start driver installation for a new hardware. If the driver installation fails go one step back and find driver manually in the Program files-Atmel-Flip folder.

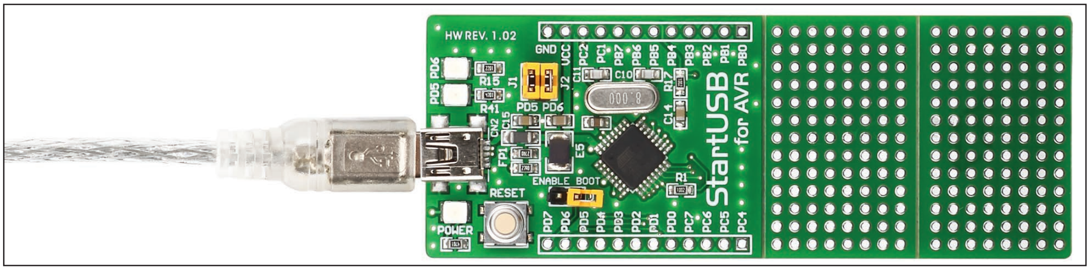
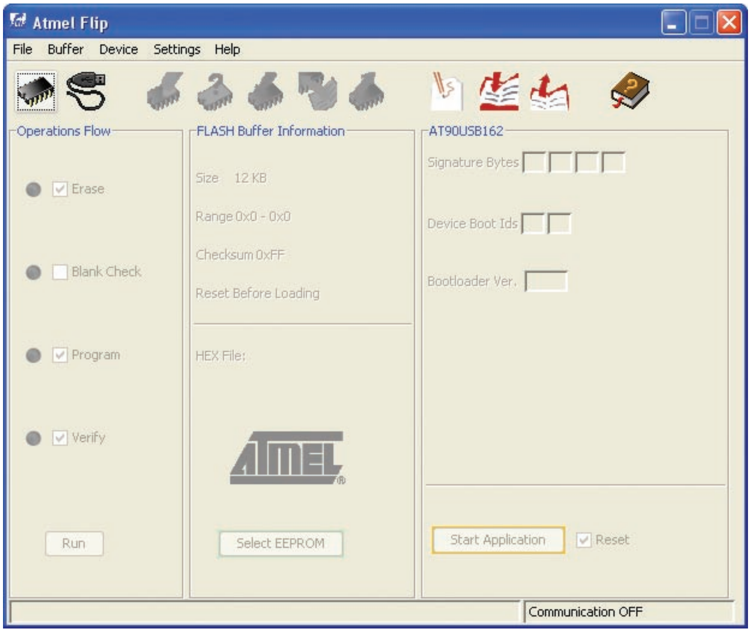


**Figure 3:** Connecting the system to a PC

## **STEP 3:** Start Atmel FLIP program

To start this program double click on the Atmel FLIP icon Flip 3.4.2 Shortcut 1 KB . The Atmel FLIP window will appear on the screen.



☑ Erase

□ Blank Check

☑ Program

☑ Verify

Size 12 KB

Range 0x0 - 0x0

Checksum 0xFF

Reset Before Loading

HEX File:

Signature Bytes

Device Boot Ids

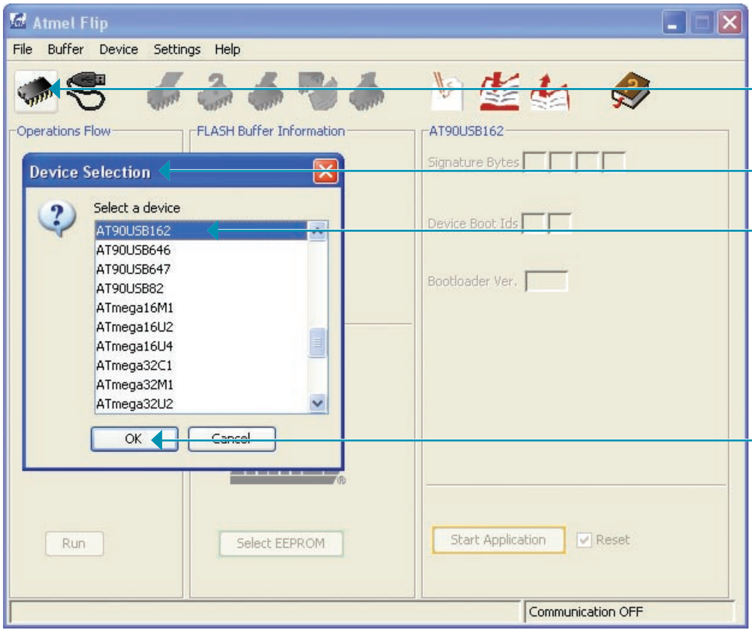
Bootloader Ver.

☑ Reset

Communication OFF

## STEP 4: Link the program with development system

After Atmel FLIP is started press **RESET** button on the StartUSB for AVR development board.

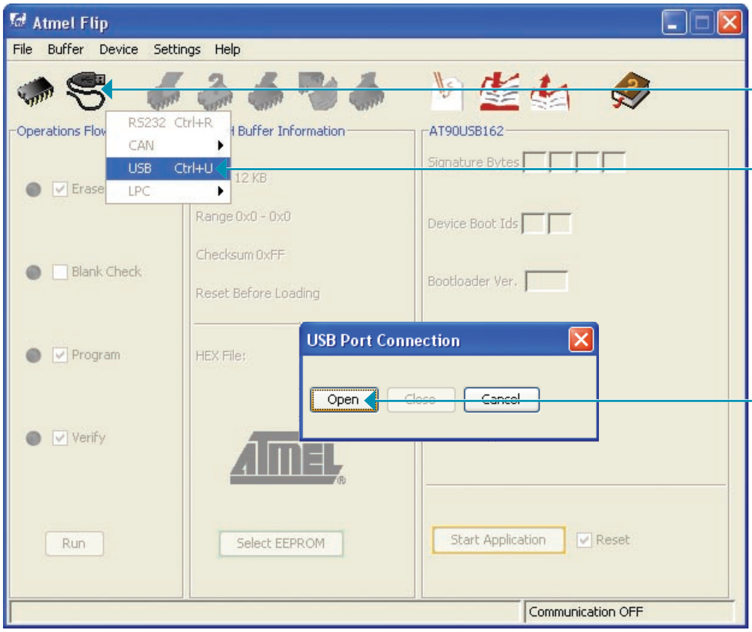


Click on the Select a Target Device button

The Device Selection window will open

Select AT90USB162 from the drop-down list

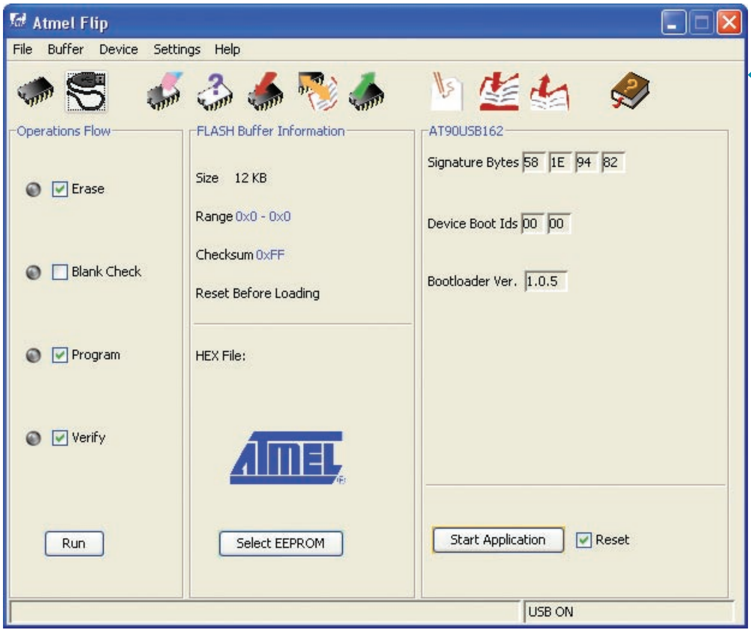
Click on the OK button



Click on the Select a Communication Medium icon. A drop down menu will appear

Select the USB option (You can also use keyboard shortcut Ctrl+U)

In the pop-up window click on the Open button



Operations Flow: ☑ Erase, □ Blank Check, ☑ Program, ☑ Verify

FLASH Buffer Information

Size 12 KB

Range 0x0 - 0x0

Checksum 0xFF

Reset Before Loading

HEX File:

AT90USB162

Signature Bytes 58 1E 94 82

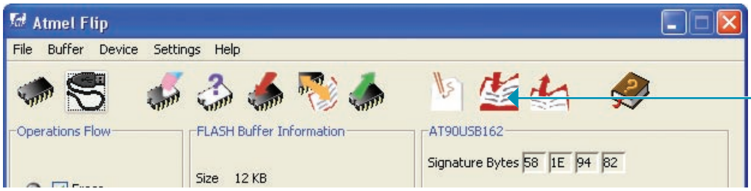
Device Boot Ids 00 00

Bootloader Ver. 1.0.5

☑ Reset

After the link is established all options and buttons will become active

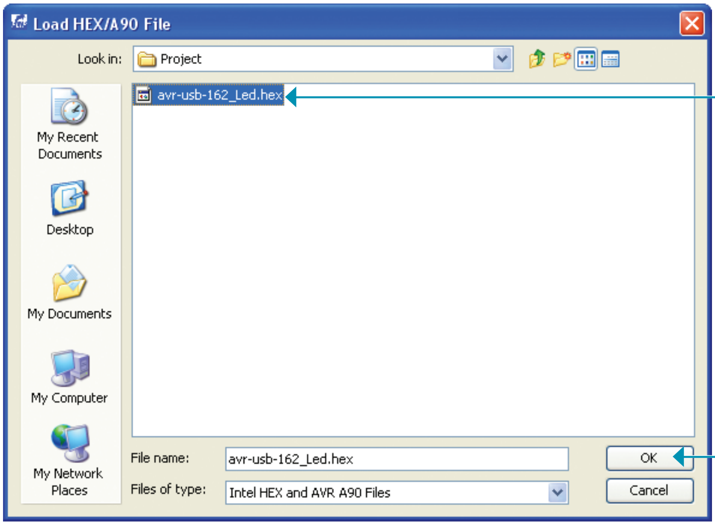
## STEP 5: Browse for .hex code



Size 12 KB

Signature Bytes 58 1E 94 82

Click on the Load HEX File icon



Look in: Project

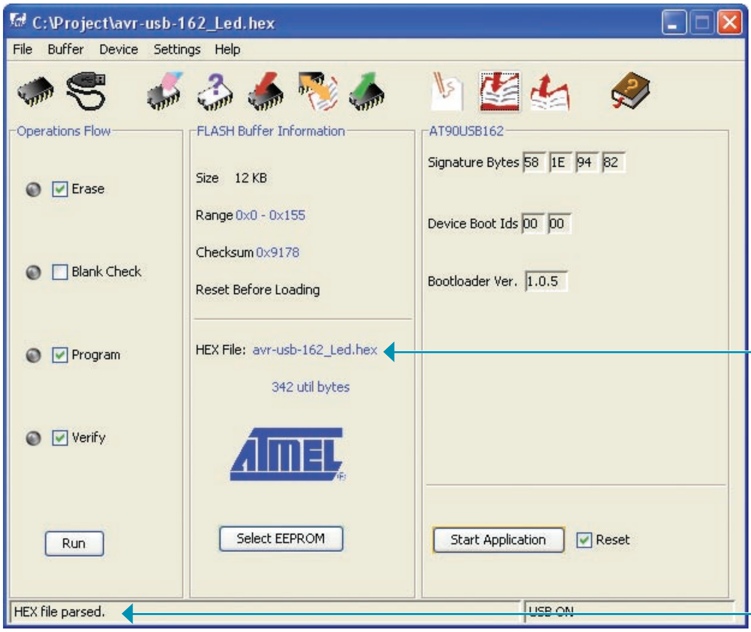
avr-usb-162_Led.hex

File name: avr-usb-162_Led.hex

Files of type: Intel HEX and AVR A90 Files

In the pop-up window browse for a .hex file

Click on the OK button



Operations Flow

- ☑ Erase
- □ Blank Check
- ☑ Program
- ☑ Verify

FLASH Buffer Information

Size 12 KB

Range 0x0 - 0x155

Checksum 0x9178

Reset Before Loading

HEX File: avr-usb-162_Led.hex

342 util bytes

AT90USB162

Signature Bytes 58 1E 94 82

Device Boot Ids 00 00

Bootloader Ver. 1.0.5

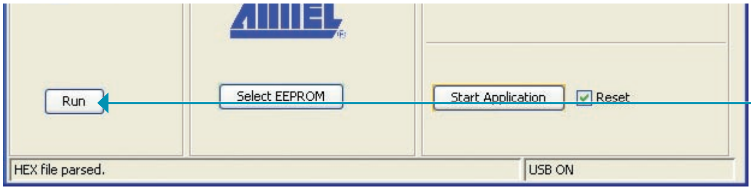
☑ Reset

HEX file parsed. USB ON

After the .hex file is loaded, an information about it will appear in the Atmel FLIP window

Information that .hex file is parsed

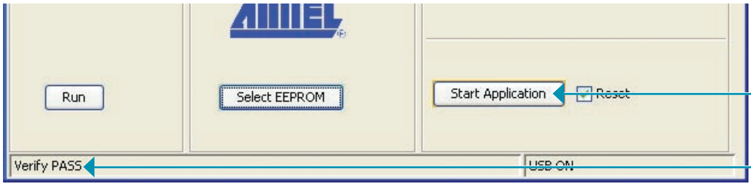
## **STEP 6:** MCU programming



☑ Reset

HEX file parsed. USB ON

Click on the Run button



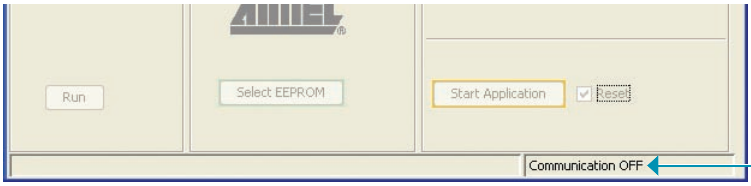
☑ Reset

Verify PASS USB ON

Click on the Start Application button

Information that .hex file verification is passed

After the MCU programming is completed, the Atmel FLIP window will become inactive.



☑ Reset

Communication OFF

Information that link with StartUSB for AVR is disabled (Communication OFF)

**NOTE:** When the MCU programming is finished, remove jumper J3 (ENABLE BOOT) from the StarUSB for AVR development system. Otherwise, the system will reenter bootloader mode .

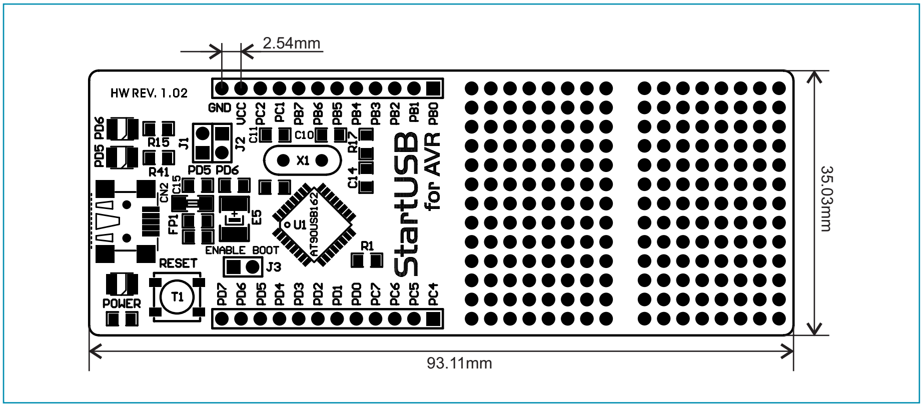


**Figure 4**: Dimensions of the StartUSB for AVR development system

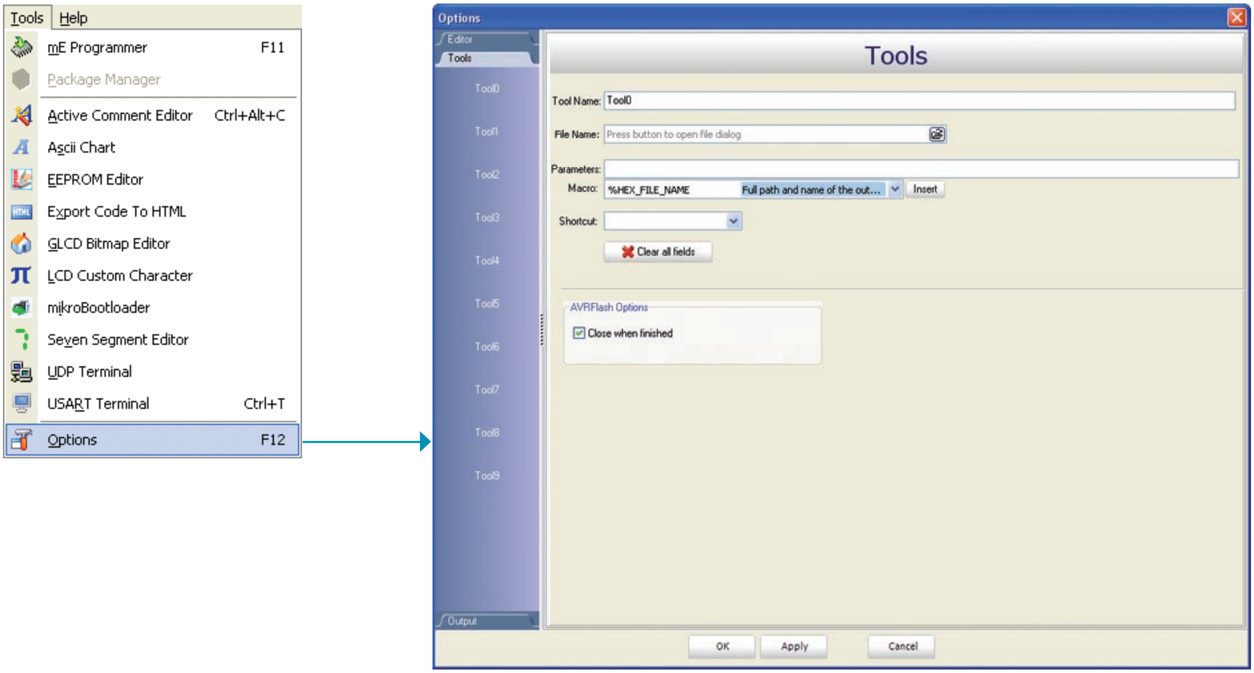
## Adding FLIP to mikroC PRO for AVR compiler

Follow the next few steps to add FLIP to mikroC PRO for AVR tools list. By doing so you can start programming of MCU on StartUSB for AVR directly from the compiler.

The process of adding FLIP tool to mikroBasic PRO for AVR and mikroPascal PRO for AVR is the same as for mikroC PRO for AVR.

### STEP 1: Open tools menu

In the compiler window select **Options** from the **Tools** menu or press F12 on keyboard.



Tool Name: Tool0

File Name: Press button to open file dialog

Parameters:

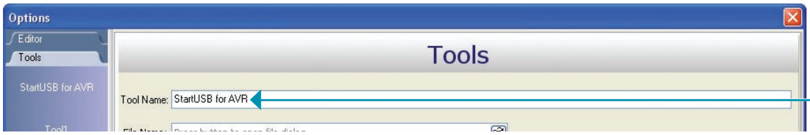
Macro: %HEX_FILE_NAME Full path and name of the out...

Shortcut:

☑ Close when finished

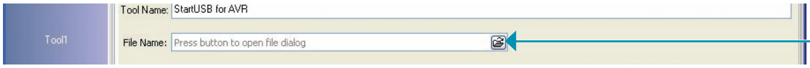
**Figure 5:** Options window

## STEP 2: Add tool parameters



Tool Name: StartUSB for AVR

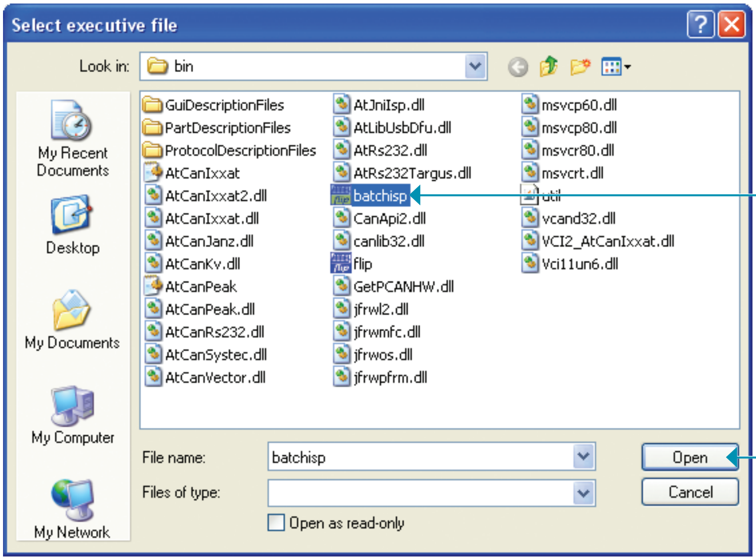
Rename Tool0 to StartUSB for AVR



Tool Name: StartUSB for AVR

File Name: Press button to open file dialog

Click on the open button

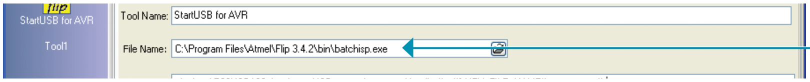


File name: batchisp

□ Open as read-only

Find batchisp in the C:\Program Files\Atmel\Flip 3.4.2\bin folder

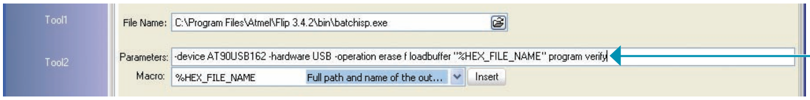
Click on the Open button



Tool Name: StartUSB for AVR

File Name: C:\Program Files\Atmel\Flip 3.4.2\bin\batchisp.exe

A path to batchisp.exe file will appear in the File Name box



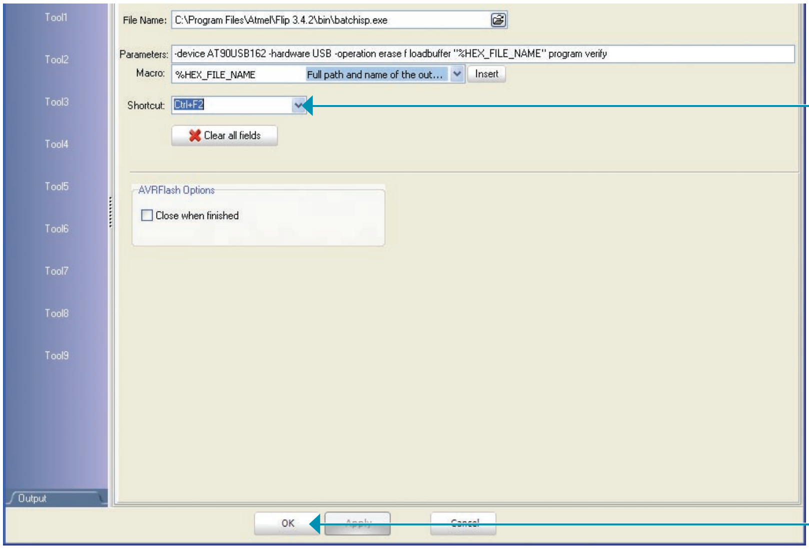
File Name: C:\Program Files\Atmel\Flip 3.4.2\bin\batchisp.exe

Parameters: -device AT90USB162 -hardware USB -operation erase f loadbuffer "%HEX_FILE_NAME" program verify

Macro: %HEX_FILE_NAME

In parameters text box type in command line path: **-device AT90USB162 -hardware USB -operation erase f loadbuffer "%HEX_FILE_NAME" program verify**

**NOTE**: If you copy command line path retype every quotation mark



File Name: C:\Program Files\Atmel\Flip 3.4.2\bin\batchisp.exe

Parameters: -device AT90USB162 -hardware USB -operation erase f loadbuffer "%HEX_FILE_NAME" program verify

Macro: %HEX_FILE_NAME Full path and name of the out... Insert

Shortcut: Ctrl+F2

AVRFlash Options

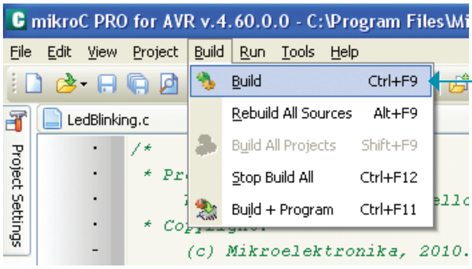
□ Close when finished

Select desired shortcut (in this case Ctrl+F2) from the drop down list

Click OK

## STEP 3: Compile source code

After the source code is written, select the Build option from the Build menu to compile it.

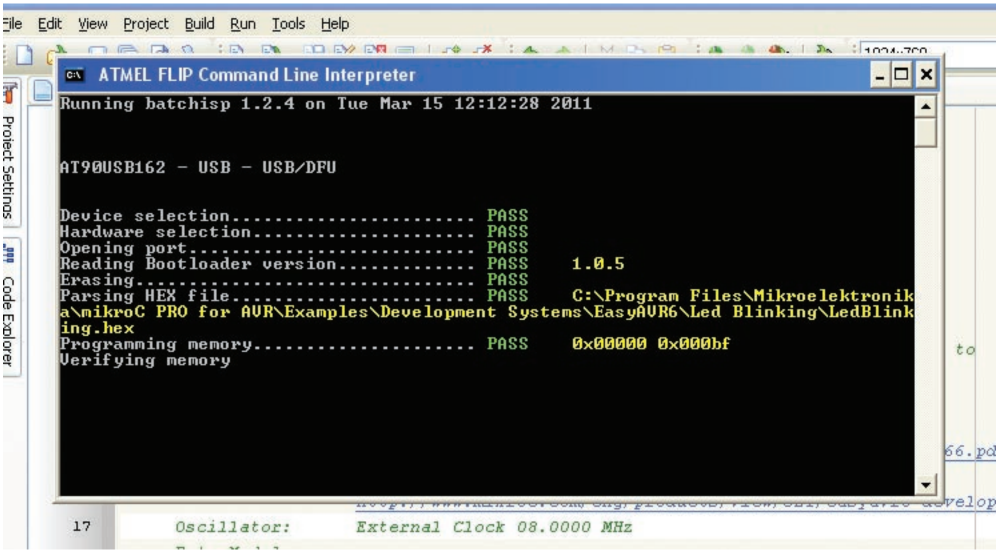

mikroC PRO for AVR v.4.60.0.0 - C:\Program Files\Mi
File Edit View Project Build Run Tools Help
Build Ctrl+F9
Rebuild All Sources Alt+F9
Build All Projects Shift+F9
Stop Build All Ctrl+F12
Build + Program Ctrl+F11
LedBlinking.c
Project Settings
/*
* Pr
* Co
(c) Mikroelektronika, 2010.

Click on the Build option or press Ctrl+F9 on keyboard

When the source code is compiled, the .hex file is generated and ready to be uploaded into the MCU.

## STEP 4: Uploade .hex file into MCU

To upload .hex file we will use StartUSB for AVR that was created in step 2. Just press Ctrl+F2 (or chosen shortcut) and .hex file will be automatically transferred into the MCU, Figure 6.



File Edit View Project Build Run Tools Help
ATMEL FLIP Command Line Interpreter
Project Settings
Code Explorer
17 Oscillator: External Clock 08.0000 MHz

```
Running batchisp 1.2.4 on Tue Mar 15 12:12:28 2011


AT90USB162 - USB - USB/DFU


Device selection........................ PASS
Hardware selection...................... PASS
Opening port............................ PASS
Reading Bootloader version.............. PASS    1.0.5
Erasing................................. PASS
Parsing HEX file........................ PASS    C:\Program Files\Mikroelektronik
a\mikroC PRO for AVR\Examples\Development Systems\EasyAVR6\Led Blinking\LedBlink
ing.hex
Programming memory...................... PASS    0x00000 0x000bf
Verifying memory
```

**Figure 6:** Uploading .hex file

After uploading is over, a command line window (figure 6) will disappear which means that .hex file is uploaded.