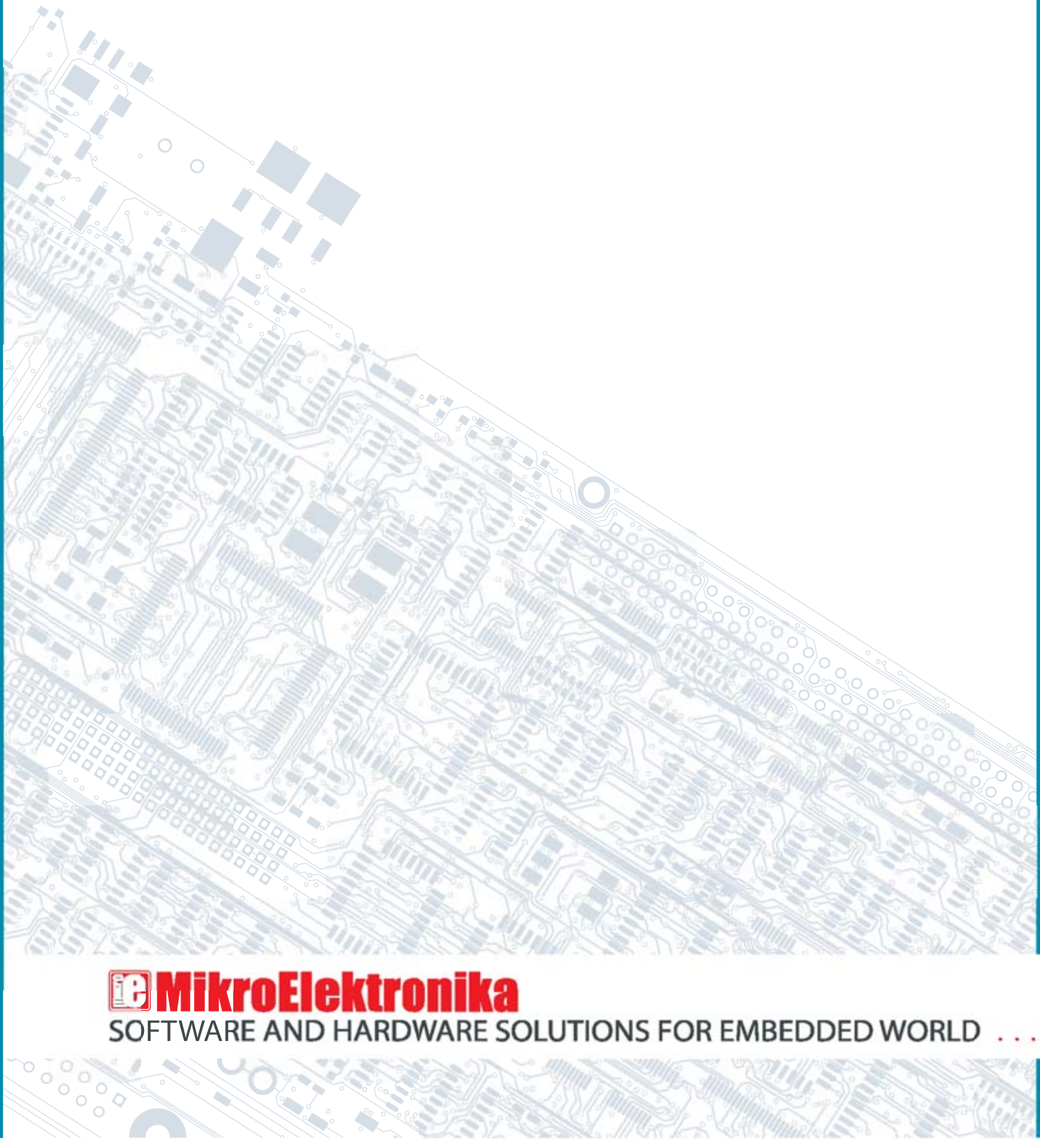
AVRflash™

User manual

AVRprog programmer is a high performance tool used for programming AVR microcontroller families from Atmel. The AVRflash program communicates to the microcontroller through a USB cable which is also used for powering the AVRprog programmer.



Program

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

TO OUR VALUED CUSTOMERS

I want to express my thanks to you for being interested in our products and for having confidence in mikroElektronika.
The primary aim of our company is to design and produce high quality electronic products and to constantly improve the performance thereof in order to better suit your needs.

Nebojsa Matic
General Manager

TABLE OF CONTENTS

1.0. Introduction to AVRprog™ Programmer

The *AVRprog* programmer is a great tool used for programming AVR® microcontrollers from Atmel®. Given that it is a low power consumption device, it is ideal for working with portable PCs. It is very popular tool among beginners and professional users alike, for its unique design and simplicity. In order to use this programmer, it is necessary to have the *AVRflash*™ program and appropriate driver, provided on the product CD, installed on your PC. The *AVRflash* program communicates to the microcontroller through a USB cable which is also used for powering the *AVRprog* programmer.

The *AVRprog* programmer is built into all AVR development systems designed by MikroElektronika. There is also a stand-alone *AVRprog* programmer used for programming AVR microcontrollers built into (soldered on) the target device.

The *AVRprog* programmer is built into all MikroElektronika's development systems designed for working with AVR microcontrollers.

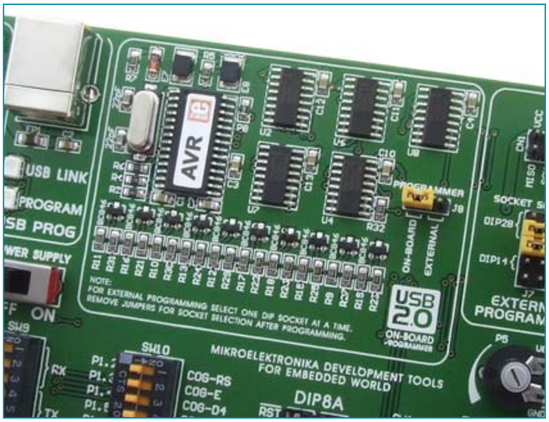


Figure 1-1: On-board *AVRprog* programmer

The *AVRprog* programmer is also available as a stand-alone device used for programming AVR microcontrollers built into (soldered on) the taget device.

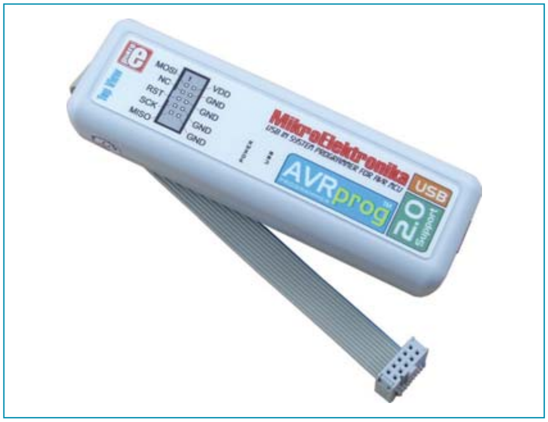
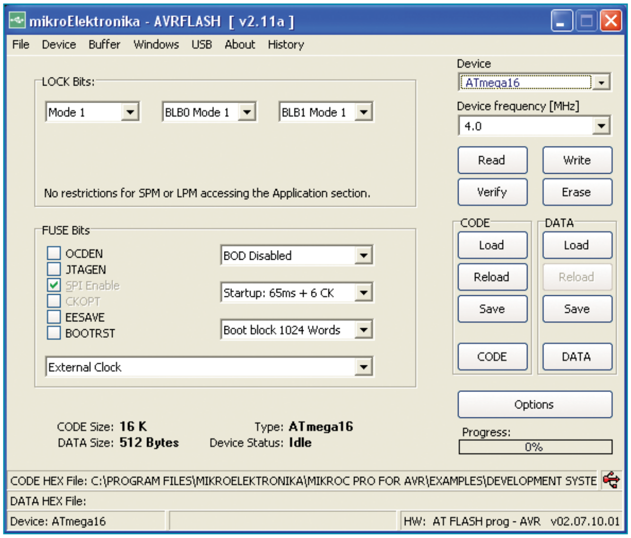


Figure 1-2: Stand-alone *AVRprog* programmer


mikroElektronika - AVRFLASH [v2.11a]
File Device Buffer Windows USB About History
Read
Write
Verify
Erase
CODE
DATA
Load
Load
Reload
Reload
Save
Save
CODE
DATA
Options

Device: ATmega16

Device frequency [MHz]: 4.0

LOCK Bits: Mode 1 | BLB0 Mode 1 | BLB1 Mode 1

No restrictions for SPM or LPM accessing the Application section.

FUSE Bits

- □ OCDEN
- □ JTAGEN
- ☑ SPI Enable
- □ CKOPT
- □ EESAVE
- □ BOOTRST

BOD Disabled

Startup: 65ms + 6 CK

Boot block 1024 Words

External Clock

CODE Size: 16 K
DATA Size: 512 Bytes
Type: ATmega16
Device Status: Idle

Progress: 0%

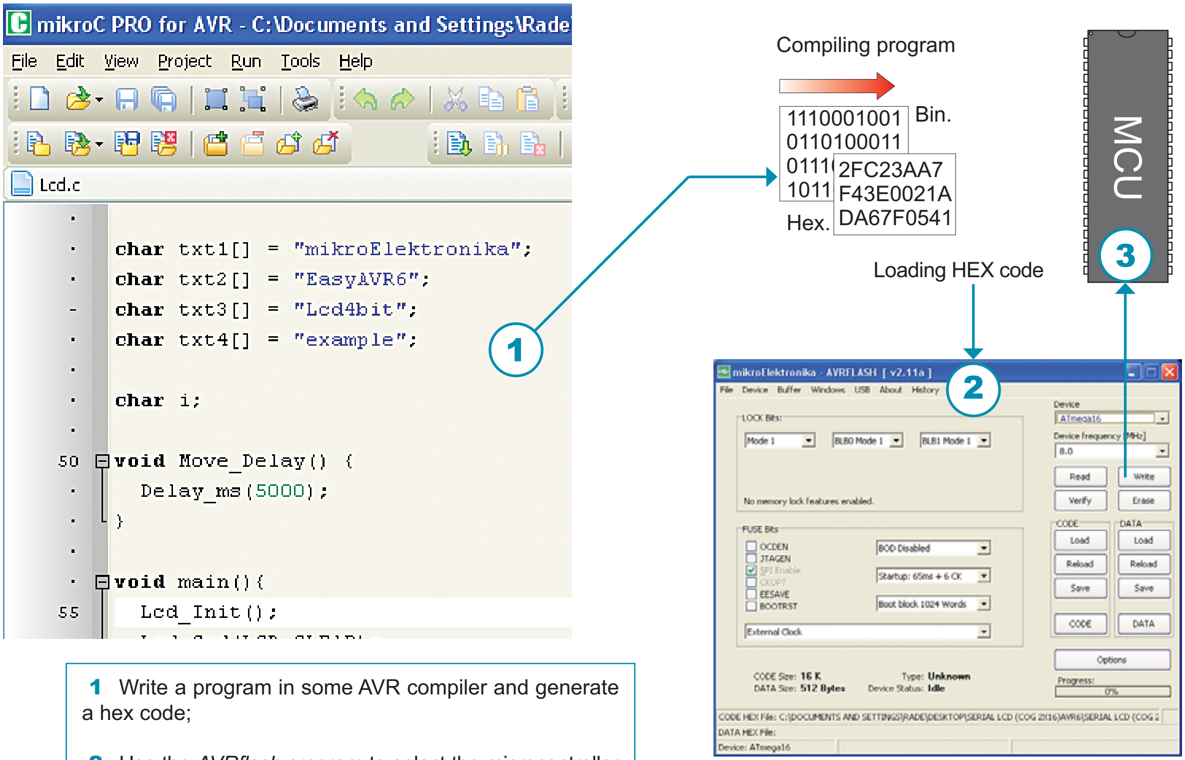
CODE HEX File: C:\PROGRAM FILES\MIKROELEKTRONIKA\MIKROC PRO FOR AVR\EXAMPLES\DEVELOPMENT SYSTE
DATA HEX File:
Device: ATmega16
HW: AT FLASH prog - AVR v02.07.10.01

The *AVRflash* program contains an option for selecting the microcontroller to be programmed. The latest version of this software with updated list of supported microcontrollers can be downloaded free of charge from our website at www.mikroe.com

Figure 1-3: *AVRflash* program is used for programming AVR microcontrollers from Atmel

2.0. Dumping Code into the Microcontroller

The programming process starts by writing a program in some AVR compiler such as *mikroC PRO for AVR™*, *mikroBASIC PRO for AVR™*, *mikroPASCAL PRO for AVR™* etc. Once the program has been written, it is necessary to compile it into appropriate format which can be loaded into the microcontroller. The program to be loaded into the microcontroller has the *.hex* extension. The last step is to write the generated .hex file into the microcontroller using the *AVRflash* program.



mikroC PRO for AVR - C:\Documents and Settings\Rade
File Edit View Project Run Tools Help
Lcd.c
1
Compiling program
1110001001
0110100011
0111
1011
Bin.
2FC23AA7
F43E0021A
DA67F0541
Hex.
Loading HEX code
MCU
3
2
mikroElektronika - AVRFLASH [v2.11a]
File Device Buffer Windows USB About History
Read
Write
Verify
Erase
CODE
DATA
Load
Reload
Save
Options

```
char txt1[] = "mikroElektronika";
char txt2[] = "EasyAVR6";
char txt3[] = "Lcd4bit";
char txt4[] = "example";

char i;

void Move_Delay() {
  Delay_ms(5000);
}

void main(){
  Lcd_Init();
```

Device: ATmega16
Device frequency (MHz): 8.0
LOCK Bits: Mode 1, BLB0 Mode 1, BLB1 Mode 1
No memory lock features enabled.
FUSE Bits:
□ OCDEN
□ JTAGEN
☑ SPI Enable
□ CKOPT
□ EESAVE
□ BOOTRST
BOD Disabled
Startup: 65ms + 6 CK
Boot block 1024 Words
External Clock
CODE Size: 16 K
DATA Size: 512 Bytes
Type: Unknown
Device Status: Idle
Progress: 0%
CODE HEX File: C:\DOCUMENTS AND SETTINGS\RADE\DESKTOP\SERIAL LCD (COG 2X16)AVR6\SERIAL LCD (COG 2
DATA HEX File:
Device: ATmega16

1 Write a program in some AVR compiler and generate a hex code;

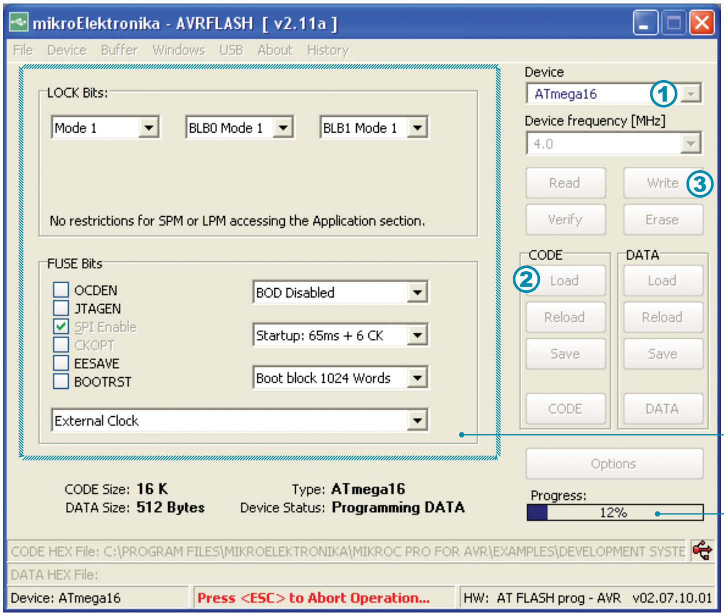
2 Use the *AVRflash* program to select the microcontroller to be programmed and load the hex code;

3 Click the *Write* button to dump code into the microcontroller.

On the right side of the *AVRflash* program's main window there are a number of options which will make the programming process easier. Positioned in the right bottom corner, the *Progress* bar is used for monitoring the programming process.

3.0. AVRflash Program’s Operation

The *AVRflash* program is easy to use as all the options necessary for its operation are provided in a simple window which will appear either by clicking on the AVRFLASH icon or automatically by starting the compiling process in the AVR compiler (*Build And Program* option).
The options used for setting configuration bits are provided on the left side of the window, whereas the options for loading HEX file into the programmer and the microcontroller are provided on the right side of the window. The layout of the left side of the window varies depending on the microcontroller’s type and configuration bits.



LOCK Bits: Mode 1 BLB0 Mode 1 BLB1 Mode 1

No restrictions for SPM or LPM accessing the Application section.

FUSE Bits:
□ OCDEN
□ JTAGEN
☑ SPI Enable
□ CKOPT
□ EESAVE
□ BOOTRST

BOD Disabled
Startup: 65ms + 6 CK
Boot block 1024 Words
External Clock

Device: ATmega16
Device frequency [MHz]: 4.0

CODE Size: 16 K Type: ATmega16
DATA Size: 512 Bytes Device Status: Programming DATA

Progress: 12%

CODE HEX File: C:\PROGRAM FILES\MIKROELEKTRONIKA\MIKROC PRO FOR AVR\EXAMPLES\DEVELOPMENT SYSTE
DATA HEX File:

Device: ATmega16 Press <ESC> to Abort Operation... HW: AT FLASH prog - AVR v02.07.10.01

Figure 3-1: *AVRflash* program’s main window

To load the program into the microcontroller, follow these steps:

1. Select the microcontroller to be programmed and the *AVRflash* will automatically set default parameters for working with the respective microcontroller
2. Click the *Load* (CODE) option to open the window to select the hex code to be loaded into the microcontroller
3. Click the *Write* option to start programming the microcontroller

Settings for configuration bits varies depending on the type of the microcontroller

The *Progress* bar shows the programming progress expressed in percentages

The *AVRflash* program enables a hex code, generated in some of the AVR compilers, to be loaded into the microcontroller. The hex code should be first loaded into the programmer’s buffer by clicking the *Load* option, then into the microcontroller by clicking the *Write* option within the programmer’s main window. The programming progress will be shown in the *Progress* bar in the bottom right corner of the same window.

In order to prevent a loaded code from being accidentally changed, LOCK bits are to be used. Depending on the protection level, one of the three available modes may be employed:

Mode 1 - protection disabled
Mode 2 - reprogramming disabled, code reading enabled
Mode 3 - full protection, both reprogramming and code reading disabled

FUSE bits are primarily used for selecting the microcontroller’s operating mode when it is turned on. These bits also have other functions such as selecting clock signal source, enabling the operation of built-in JTAG interface, selecting reset vector etc. Only experienced users are advised to change these bits. By selecting the microcontroller from the *Device* drop-down menu (Figure 3-1, no. 1), all relevant settings of FUSE bits will be automatically performed.

4.0. Software Installation

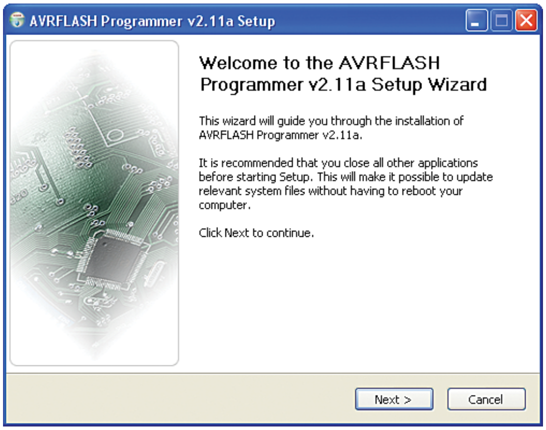
Before using the *AVRflash* program, it is necessary to install the appropriate driver. For more information on installing USB drivers consult the relevant manual.

Step 1: Start installation

Insert the product CD into your PC drive. After a few seconds, a list with all MikroElektronika's products will appear on the screen. To start the installation process of the *AVRflash* software, click on the setup icon provided in the *AVRflash* section on the product CD:

CD Drive:/zip/AVRFlash_setup.exe

You can also download the *AVRflash* program free of charge from our website. In this case the installation starts from your hard drive. A welcome window appears. Click *Next* to proceed.



AVRFLASH Programmer v2.11a Setup

Welcome to the AVRFLASH Programmer v2.11a Setup Wizard

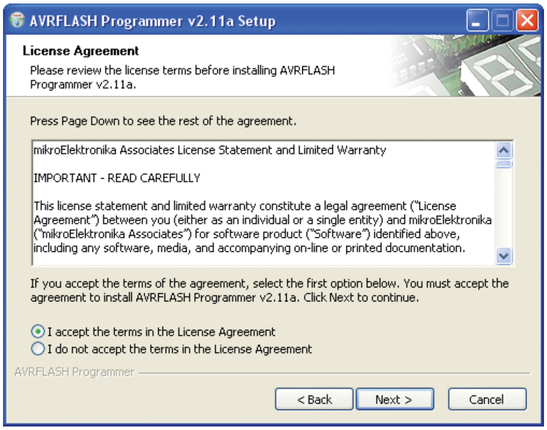
This wizard will guide you through the installation of AVRFLASH Programmer v2.11a.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

Step 2: Licence Agreement

Before you start the installation procedure, please review the licence agreement terms. To accept them, select the option *I accept the terms in the Licence Agreement* and click *Next*.



AVRFLASH Programmer v2.11a Setup

License Agreement

Please review the license terms before installing AVRFLASH Programmer v2.11a.

Press Page Down to see the rest of the agreement.

mikroElektronika Associates License Statement and Limited Warranty

IMPORTANT - READ CAREFULLY

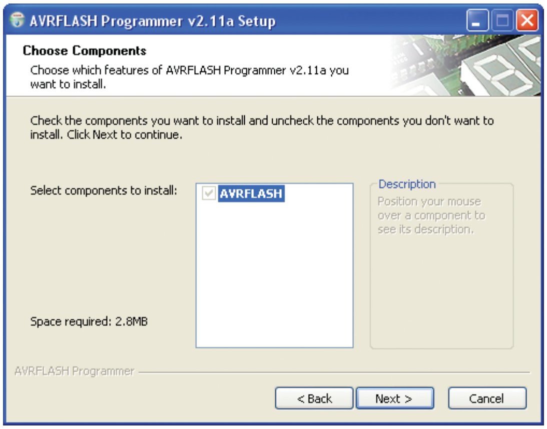
This license statement and limited warranty constitute a legal agreement ("License Agreement") between you (either as an individual or a single entity) and mikroElektronika ("mikroElektronika Associates") for software product ("Software") identified above, including any software, media, and accompanying on-line or printed documentation.

If you accept the terms of the agreement, select the first option below. You must accept the agreement to install AVRFLASH Programmer v2.11a. Click Next to continue.

● I accept the terms in the License Agreement
○ I do not accept the terms in the License Agreement

Step 3: Choose Components

To make your choice uncomplicated, this installation step offers you only one component to choose. Click *Next*.



AVRFLASH Programmer v2.11a Setup

Choose Components

Choose which features of AVRFLASH Programmer v2.11a you want to install.

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

Select components to install: ☑ AVRFLASH

Description

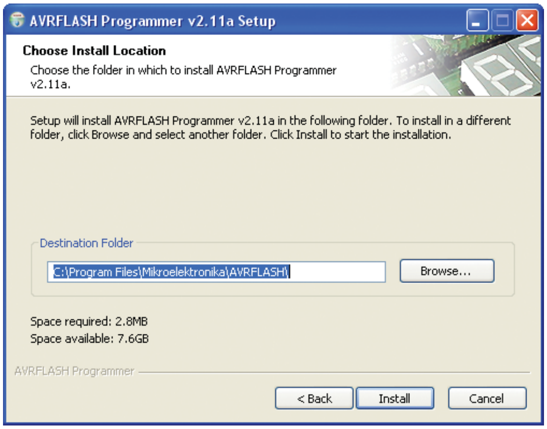
Position your mouse over a component to see its description.

Space required: 2.8MB

Step 4: Choose Installation Location

Next, you should specify the folder to install the *AVRflash* program in. If you wish to install it in a folder different from default, click *Browse* and select another folder on your hard disc. Then click *Next*. If you choose the default folder, the program will be installed on the following location:

C:\Program Files\Mikroelektronika\AVRFLASH\



AVRFLASH Programmer v2.11a Setup

Choose Install Location

Choose the folder in which to install AVRFLASH Programmer v2.11a.

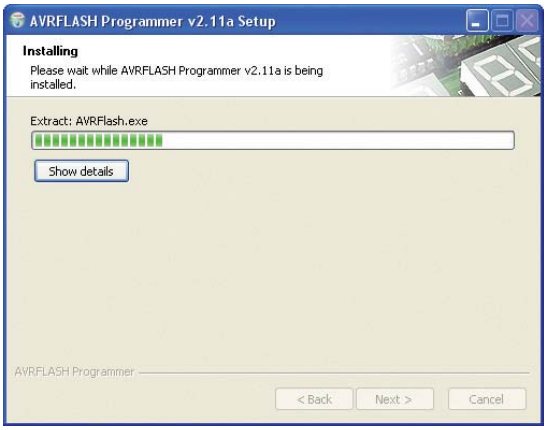
Setup will install AVRFLASH Programmer v2.11a in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.

Destination Folder: C:\Program Files\Mikroelektronika\AVRFLASH\

Space required: 2.8MB
Space available: 7.6GB

Step 5: Installation Details

The *AVRflash* program installation starts immediately. The installation progress will be shown on the screen. If you are interested in details about the installation, click the *Show details* button.



AVRFLASH Programmer v2.11a Setup

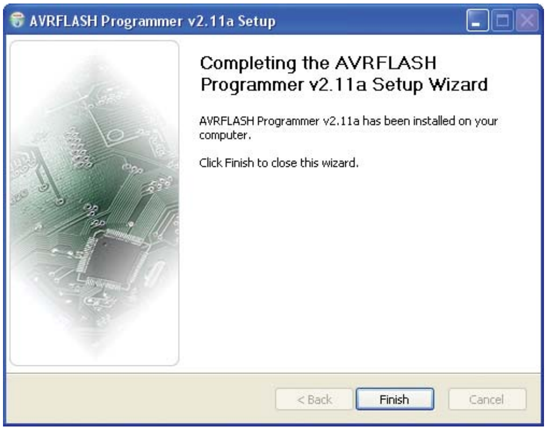
Installing

Please wait while AVRFLASH Programmer v2.11a is being installed.

Extract: AVRFlash.exe

Step 6: Completing Installation

Windows will inform you, as shown in figure on the right, that the *AVRflash* program has been successfully installed. Click *Finish* to complete the installation.



AVRFLASH Programmer v2.11a Setup

Completing the AVRFLASH Programmer v2.11a Setup Wizard

AVRFLASH Programmer v2.11a has been installed on your computer.

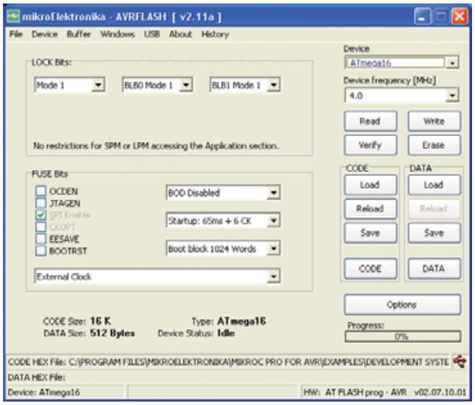
Click Finish to close this wizard.

5.0. Practical Example of Using AVRflash Program

After the software installation is complete, connect the programmer to your PC using a USB cable. The USB connection will be automatically established, which is indicated by the *USB LINK LED diode's* illumination.

Step 1: Start up the *AVRflash* program

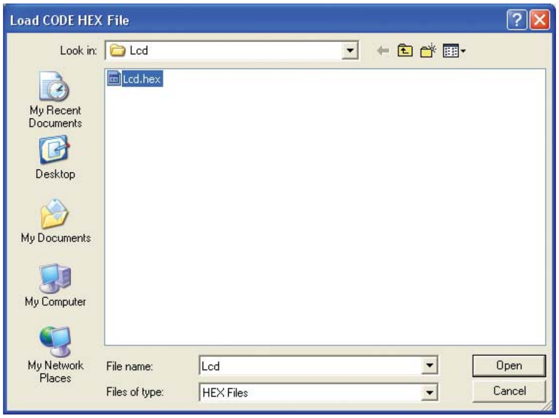
Start up the *AVRflash* program installed on your PC. Click the *Device* option in order to select the microcontroller to be programmed. The *AVRflash* program will automatically set default parameters for working with the respective microcontroller.



Device: ATmega16
Device frequency [MHz]: 4.0
LOCK Bits: Mode 1, BLB0 Mode 1, BLB1 Mode 1
No restrictions for SPM or LPM accessing the Application section.
FUSE Bits:
☐ OCDEN
☐ JTAGEN
☑ SPI Enable
☐ CKOPT
☐ EESAVE
☐ BOOTRST
BOD Disabled
Startup: 65ms + 6 CK
Boot block 1024 Words
External Clock
CODE Size: 16 K
DATA Size: 512 Bytes
Type: ATmega16
Device Status: Idle
Progress: 0%
CODE HEX File: C:\PROGRAM FILES\MIKROELEKTRONIKA\MIKROC PRO FOR AVR\EXAMPLES\DEVELOPMENT SYSTE
DATA HEX File:
Device: ATmega16
HW: AT FLASH prog - AVR v02.07.10.01

Step 2: Load a .hex file into the programmer's buffer

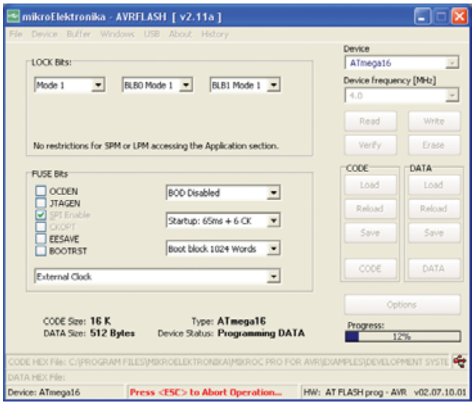
Click the *Load* button to open the *Open* window, as shown in figure on the right. Select the relevant file with the *.hex* extension and click the *Open* button. The file will be automatically loaded into the programmer's buffer.



Look in: Lcd
Lcd.hex
File name: Lcd
Files of type: HEX Files

Step 3: Load the hex code into the microcontroller

Click the *Write* button in the upper right corner of the main window to start programming the microcontroller. The programming progress will be shown in the bottom right corner of the same window.



Device: ATmega16
Device frequency [MHz]: 4.0
LOCK Bits: Mode 1, BLB0 Mode 1, BLB1 Mode 1
No restrictions for SPM or LPM accessing the Application section.
FUSE Bits:
☐ OCDEN
☐ JTAGEN
☑ SPI Enable
☐ CKOPT
☐ EESAVE
☐ BOOTRST
BOD Disabled
Startup: 65ms + 6 CK
Boot block 1024 Words
External Clock
CODE Size: 16 K
DATA Size: 512 Bytes
Type: ATmega16
Device Status: Programming DATA
Progress: 12%
CODE HEX File: C:\PROGRAM FILES\MIKROELEKTRONIKA\MIKROC PRO FOR AVR\EXAMPLES\DEVELOPMENT SYSTE
DATA HEX File:
Device: ATmega16
Press <ESC> to Abort Operation...
HW: AT FLASH prog - AVR v02.07.10.01

6.0. Keyboard Shortcuts and Command Line Parameters

Keyboard Shortcuts:

Alt-E	Erase microcontroller’s memory
Alt-B	Program memory blank check
Alt-W	Write hex code into AVR microcontroller
Alt-V	Verify loaded hex code
Alt-R	Read program memory
Alt-D	Change microcontroller type
Ctrl-S	Save .hex file
Ctrl-O	Open (Load) .hex file
Ctrl-R	Reload .hex file

Command Line:

The *AVRflash* program may also be activated from the command line, thus enabling you to use it from some other software, compiler etc. Here is a list of the command line parameters:

-w	Write to AVR microcontroller
-v	Verify
-e	Erase program from AVR microcontroller
-r	Read program from AVR microcontroller
-p	Microcontroller type (for example, ATmega16, AT90S2323 etc.)
-fc	Name of .hex file (FLASH) “[<name must be enclosed in quotation marks>]”
-fd	Name of data file (EEPROM) “[<name must be enclosed in quotation marks>]”
-LOCK:0x.....	Specify values of LOCK bits
-FUSE:0x.....	Specify valus of FUSE bits
-q	Close the *AVRflash* program after programming

Example 1:

AVRFlash.exe -w -pATMEGA16 -v -f”C:\somefile.hex”

This command is used for loading *C:\somefile.hex* into the ATmega16 microcontroller. This file is verified immediately after being loaded.

Example 2:

AVRFlash.exe -r -pATMEGA16

This command is used for reading the ATmega16 microcontroller’s program memory.

Example 3:

AVRFlash.exe -e -pATMEGA16

This command is used for erasing program from the ATmega16 microcontroller.

DISCLAIMER

All the products owned by MikroElektronika are protected by copyright law and international copyright treaty. Therefore, this manual is to be treated as any other copyright material. No part of this manual, including product and software described herein, may be reproduced, stored in a retrieval system, translated or transmitted in any form or by any means, without the prior written permission of MikroElektronika. The manual PDF edition can be printed for private or local use, but not for distribution. Any modification of this manual is prohibited.

MikroElektronika provides this manual 'as is' without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties or conditions of merchantability or fitness for a particular purpose.

MikroElektronika shall assume no responsibility or liability for any errors, omissions and inaccuracies that may appear in this manual. In no event shall MikroElektronika, its directors, officers, employees or distributors be liable for any indirect, specific, incidental or consequential damages (including damages for loss of business profits and business information, business interruption or any other pecuniary loss) arising out of the use of this manual or product, even if MikroElektronika has been advised of the possibility of such damages. MikroElektronika reserves the right to change information contained in this manual at any time without prior notice, if necessary.

HIGH RISK ACTIVITIES

The products of MikroElektronika are not fault – tolerant nor designed, manufactured or intended for use or resale as on – line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of Software could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). MikroElektronika and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

TRADEMARKS

The Mikroelektronika name and logo, the Mikroelektronika logo, mikroC, mikroC PRO, mikroBasic, mikroBasic PRO, mikroPascal, mikroPascal PRO, AVRflash, PICflash, dsPICprog, 18FJprog, PSOCprog, AVRprog, 8051prog, ARMflash, EasyPIC5, EasyPIC6, BigPIC5, BigPIC6, dsPIC PRO4, Easy8051B, EasyARM, EasyAVR5, EasyAVR6, BigAVR2, EasydsPIC4A, EasyPSoC4, EasyVR Stamp LV18FJ, LV24-33A, LV32MX, PIC32MX4 MultiMedia Board, PICPLC16, PICPLC8 PICPLC4, SmartGSM/GPRS, UNI-DS are trademarks of Mikroelektronika. All other trademarks mentioned herein are property of their respective companies.

All other product and corporate names appearing in this manual may or may not be registered trademarks or copyrights of their respective companies, and are only used for identification or explanation and to the owners' benefit, with no intent to infringe.

MikroElektronika

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

If you want to learn more about our products, please visit our website at www.mikroe.com

If you are experiencing some problems with any of our products or just need additional information, please place your ticket at www.mikroe.com/en/support

If you have any questions, comments or business proposals, do not hesitate to contact us at office@mikroe.com