

AVR USB PROGRAMMER (810210)

User Manual



About the User Guide

AVR USB programmer 810210 is a high power low cost programmer with a USB interface for makes it easier to burn programs in microcontrollers. The programmer uses a firmware-only USB driver; no special USB controller is needed. It is also officially included and supported in WinAVR.

PROPRIETARY NOTICE

This document contains proprietary information furnished for evaluation purposes only; except with the express written permission of Technophilia, such information may not be published, disclosed, or used for any other purpose. You acknowledge and agree that this document and all portions thereof, including, but not limited to, any copyright, trade secret and other intellectual property rights relating thereto, are and at all times shall remain the sole property Technophilia and that title and full ownership rights in the information contained herein and all portions thereof are reserved to and at all times shall remain with Technophilia. You acknowledge and agree that the information contained herein constitutes a valuable trade secret of Technophilia. You agree to use utmost care in protecting the proprietary and confidential nature of the information contained herein.

AVR USB Programmer (810210)

User manual

Technophilia

www.technophilia.co.in

Features

- Supports 62 AVR microcontrollers
- Compatible with Win AVR IDE
- can also be use with AVR dude directly
- Programmer powered by USB bus
- Conforms to USB power requirements, including sleep
- Supports USB v1.1 protocol and USB v2.0 compatible
- Direct USB drivers -- not virtual serial port
- Microsoft approved drivers for real plug and play
- Supports Win 98SE/2000/XP/ME
 - ** Win 95 and early Win98 require Microsoft USB update
- Programs target devices from 1.8V to 5V
- Multiple programmers from one PC
- Automatic program start on insertion

Specifications

It supports the following micro controllers

1. ATMEGA6450
2. ATMEGA3250
3. ATMEGA645
4. ATMEGA325
5. AT90USB1287
6. AT90USB1286
7. AT90USB647
8. AT90USB646
9. ATTINY84
10. ATTINY44
11. ATTINY24
12. ATMEGA2561
13. ATMEGA2560
14. ATMEGA1281
15. ATMEGA1280
16. ATMEGA640
17. ATTINY85
18. ATTINY45

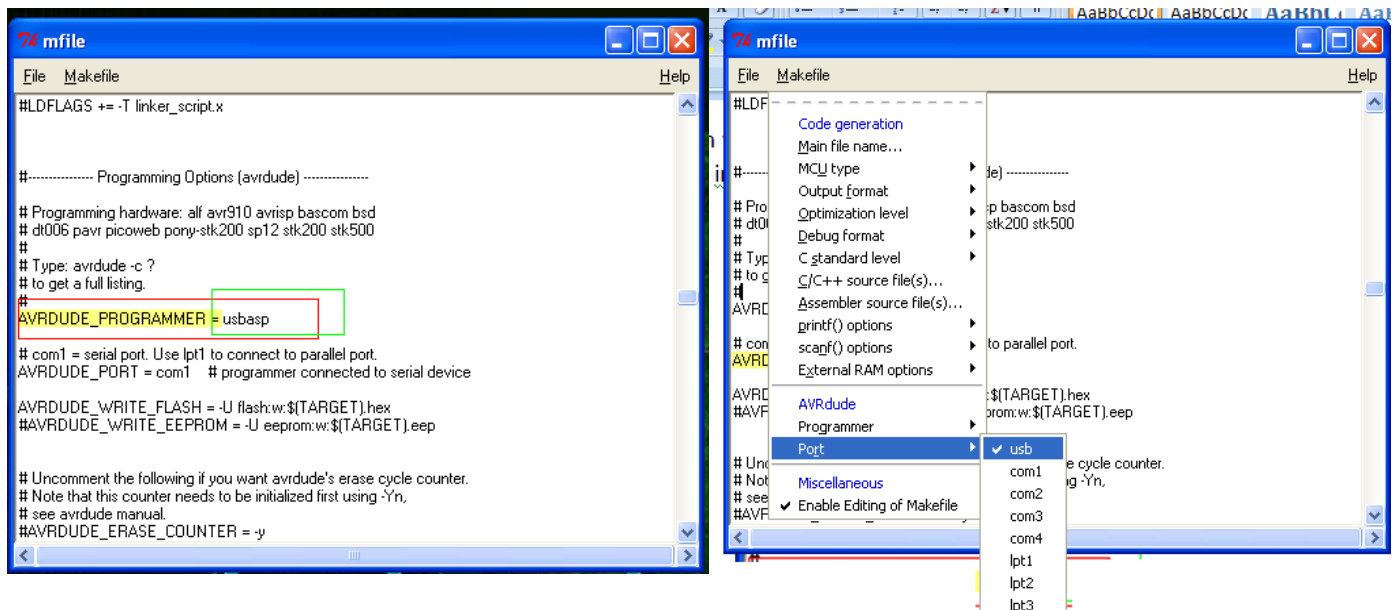
19. ATTINY25
20. AT90PWM3
21. AT90PWM2
22. ATTINY2313
23. ATMEGA168
24. ATMEGA88
25. ATMEGA48
26. ATTINY861
27. ATTINY461
28. ATTINY261
29. ATTINY26
30. ATMEGA8535
31. ATMEGA8515
32. ATMEGA8
33. ATMEGA161
34. ATMEGA32
35. ATMEGA6490
36. ATMEGA49
37. ATMEGA3290
38. ATMEGA329
39. ATMEGA169
40. ATMEGA163
41. ATMEGA162
42. ATMEGA644
43. ATMEGA324
44. ATMEGA164
45. ATMEGA16
46. AT90CAN128
47. ATMEGA128
48. ATMEGA64
49. ATMEGA103
50. AT90S8535
51. AT90S8515
52. AT90S4434
53. AT90S4433
54. AT90S2343
55. AT90S2333
56. AT90S2313
57. AT90S4414
58. AT90S1200
59. ATTINY15
60. ATTINY13
61. ATTINY12
62. ATTINY11

How to use it

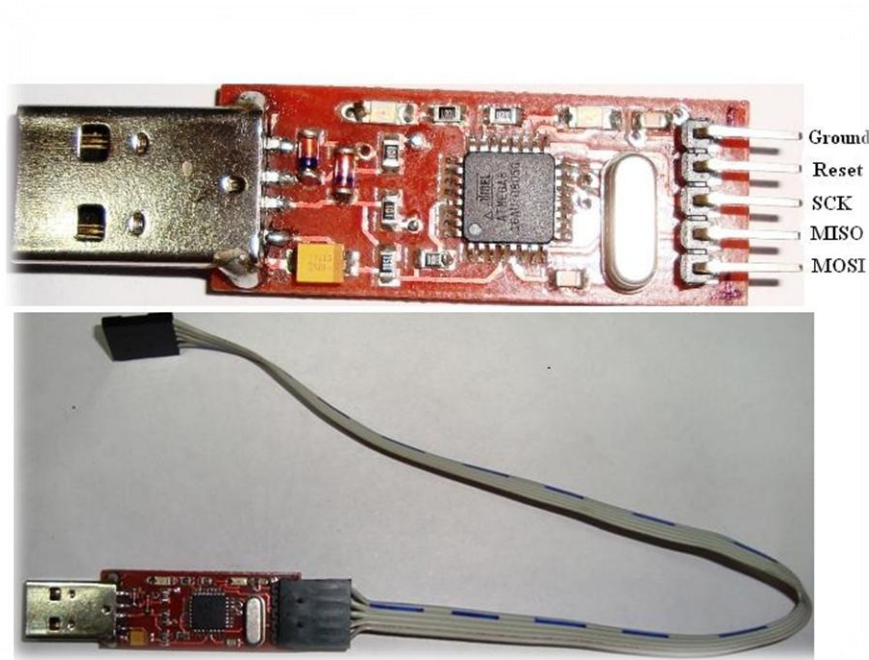
1. Install Win-AVR in your PC
2. Install AVR dude-5.5 in your PC
3. Connect this device to the PC's USB Port
4. It will give you a pop-up window "Found new hardware wizard"
Here locate and install your driver manually given on the CD

To use it with WINAVR do the following steps

1. When you are creating make file for win AVR in the programmer option edit the make file as "usbasp" in the port option select USB then save it



2. Connect your programmer into your micro controller board as per the pins diagram mentioned bellow



3. To compile your program go to tools->Make all

4. To load your code in your uc go to tools->program

if your code is downloaded successfully it will give you the message in the following format

```
*****  
*****
```

Creating load file for EEPROM: main.eep

```
avr-objcopy -j .eeprom --set-section-flags=.eeprom="alloc,load" \  
--change-section-lma .eeprom=0 --no-change-warnings -O ihex main.elf main.eep || exit 0
```

c:\WinAVR-20070525\bin\avr-objcopy.exe: there are no sections to be copied!

```
avrdude -p atmega16 -P usb -c usbasp -U flash:w:main.hex  
found 5 busses
```

avrdude: AVR device initialized and ready to accept instructions

Reading | ##### | 100% 0.02s

avrdude: Device signature = 0x1e9403

avrdude: NOTE: FLASH memory has been specified, an erase cycle will be performed

To disable this feature, specify the -D option.

avrdude: erasing chip

avrdude: reading input file "main.hex"

avrdude: input file main.hex auto detected as Intel Hex

avrdude: writing flash (226 bytes):

User manual

Writing | ##### | 100% 0.05s

```
avrdude: 226 bytes of flash written
avrdude: verifying flash memory against main.hex:
avrdude: load data flash data from input file main.hex:
avrdude: input file main.hex auto detected as Intel Hex
avrdude: input file main.hex contains 226 bytes
avrdude: reading on-chip flash data:
```

Reading | ##### | 100% 0.06s

```
avrdude: verifying ...
avrdude: 226 bytes of flash verified
```

avrdude done. Thank you.

> Process Exit Code: 0

> Time Taken: 00:01

```
*****
*****
```

If there is any problem you may get the following information on the screen

```
*****
*****
```

Creating load file for EEPROM: main.eep

```
avr-objcopy -j .eeprom --set-section-flags=.eeprom="alloc,load" \
--change-section-lma .eeprom=0 --no-change-warnings -O ihex main.elf main.eep || exit 0
c:\WinAVR-20070525\bin\avr-objcopy.exe: there are no sections to be copied!
avrdude -p atmega16 -P usb -c usbasp -U flash:w:main.hex
found 5 busses
```

```
avrdude: error: programm enable: target doesn't answer. 1
avrdude: AVR device initialized and ready to accept instructions
```

Reading | ##### | 100% 0.02s

```
avrdude: Device signature = 0x000000
avrdude: Yikes! Invalid device signature.
Double check connections and try again, or use -F to override
this check.
```

avrdude done. Thank you.

make.exe: *** [program] Error 1

> Process Exit Code: 2

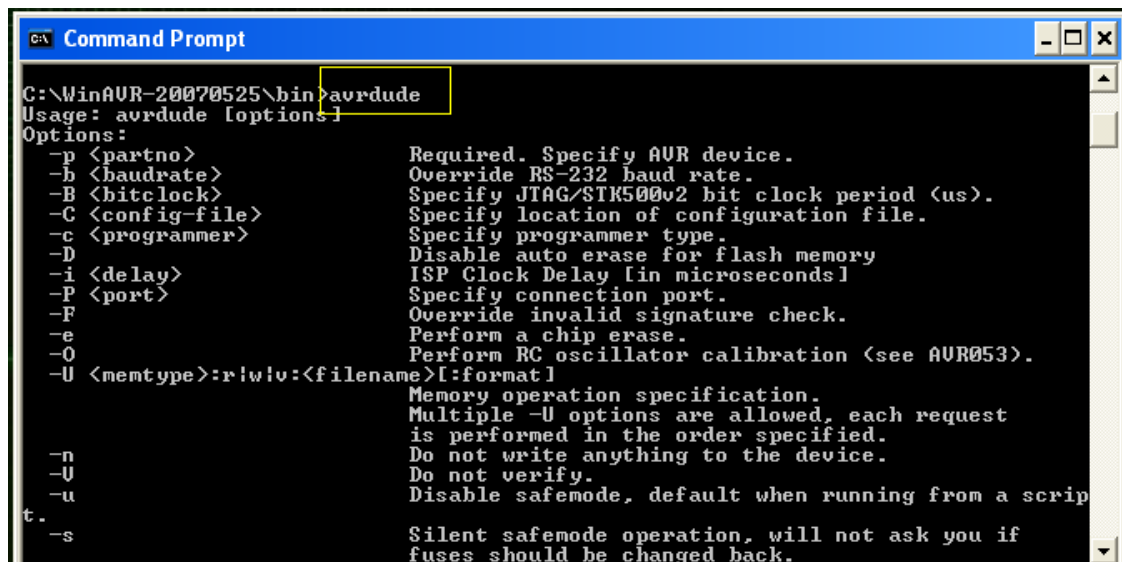
> Time Taken: 00:01

```
*****
*****
```

To use it with AVR DUDE do the following steps

Here you can explore its more features by using avrdude

1. Go to command prompt and enter into the following directory
C: WinAVR-20070525\bin>
2. To get information about the commands do following.



```
Command Prompt
C:\WinAVR-20070525\bin>avrdude
Usage: avrdude [options]
Options:
  -p <partno>           Required. Specify AVR device.
  -b <baudrate>         Override RS-232 baud rate.
  -B <bitclock>         Specify JTAG/STK500v2 bit clock period (us).
  -C <config-file>      Specify location of configuration file.
  -c <programmer>       Specify programmer type.
  -D                   Disable auto erase for flash memory
  -i <delay>            ISP Clock Delay [in microseconds]
  -P <port>             Specify connection port.
  -F                   Override invalid signature check.
  -e                   Perform a chip erase.
  -O                   Perform RC oscillator calibration (see AVR053).
  -U <memtype>[:r|w|v:<filename>[:format]]
                        Memory operation specification.
                        Multiple -U options are allowed, each request
                        is performed in the order specified.
  -n                   Do not write anything to the device.
  -U                   Do not verify.
  -u                   Disable safemode, default when running from a scrip
t.
  -s                   Silent safemode operation, will not ask you if
                        fuses should be changed back.
```

3. To check your connectivity or device, go to the terminal mode by typing the following command.

You can also check different status of your uC here

Note-here we have used an uC of ATmega-16


```

C:\ Command Prompt - avrdude -p m16 -c usbaspp -t
found 5 busses
avrdude: error: programm enable: target doesn't answer. 1
avrdude: AVR device initialized and ready to accept instructions
Reading ! ##### ! 100% 0.02s
avrdude: Device signature = 0x000000
avrdude: Yikes! Invalid device signature.
        Double check connections and try again, or use -F to override
        this check.

avrdude done. Thank you.

C:\WinAUR-20070525\bin>avrdude -p m16 -c usbaspp -t
found 5 busses
avrdude: AVR device initialized and ready to accept instructions
Reading ! ##### ! 100% 0.02s
avrdude: Device signature = 0x1e9403
avrdude>

```

4. In terminal mode you can do the following operations, type *quit* and press enter to exit from terminal mode

```

C:\ Command Prompt
avrdude: Device signature = 0x1e9403
avrdude> r eeprom
>>> r eeprom
0000 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff | .....|
0010 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff | .....|
0020 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff | .....|
0030 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff | .....|

avrdude> r flash
>>> r flash
0000 0c 94 49 00 0c 94 00 00 0c 94 00 00 0c 94 00 00 | öI. ö.. ö.. ö..|
0010 0c 94 00 00 0c 94 00 00 0c 94 79 00 0c 94 80 00 | ö.. ö.. öy. ö..|
0020 0c 94 00 00 0c 94 00 00 0c 94 00 00 0c 94 00 00 | ö.. ö.. ö.. ö..|
0030 0c 94 00 00 0c 94 00 00 0c 94 00 00 0c 94 00 00 | ö.. ö.. ö.. ö..|

avrdude> r hfuse
>>> r hfuse
0000 c9 |. |

avrdude> r lfuse
>>> r lfuse
0000 ff |. |

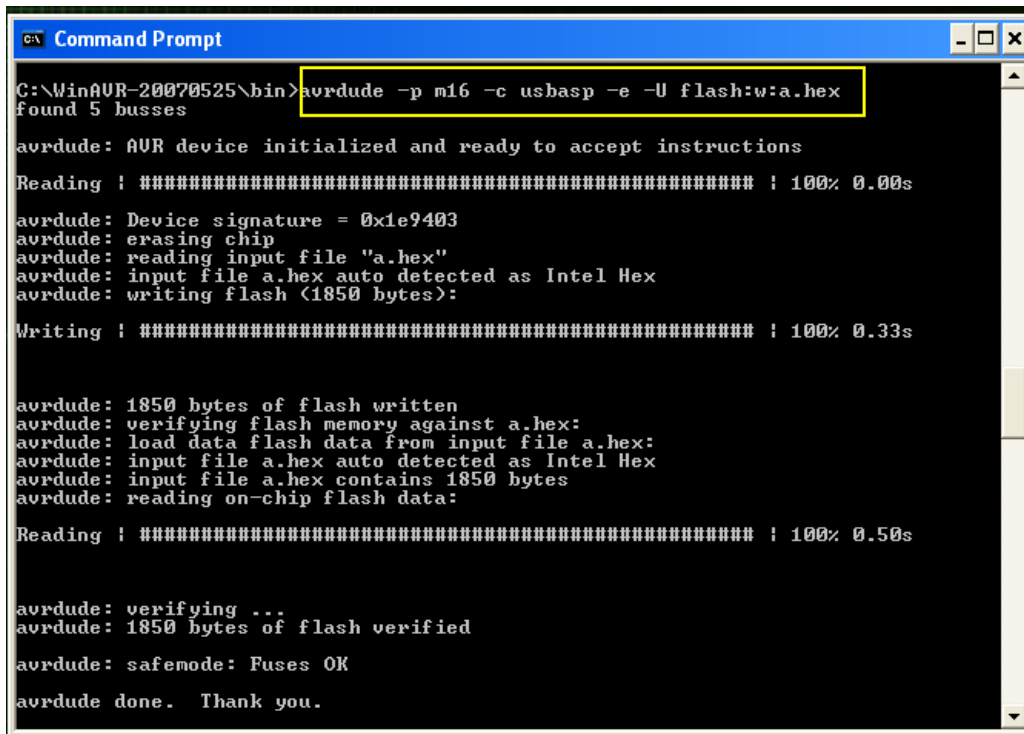
avrdude> quit
>>> quit

avrdude: safemode: Fuses OK
avrdude done. Thank you.

C:\WinAUR-20070525\bin>

```

5. To program your uC paste your hex file in the bin folder mentioned here then type the following command.



```

C:\WinAUR-20070525\bin>avrdude -p m16 -c usbasp -e -U flash:w:a.hex
found 5 busses

avrdude: AVR device initialized and ready to accept instructions

Reading : ##### : 100% 0.00s

avrdude: Device signature = 0x1e9403
avrdude: erasing chip
avrdude: reading input file "a.hex"
avrdude: input file a.hex auto detected as Intel Hex
avrdude: writing flash (1850 bytes):

Writing : ##### : 100% 0.33s

avrdude: 1850 bytes of flash written
avrdude: verifying flash memory against a.hex:
avrdude: load data flash data from input file a.hex:
avrdude: input file a.hex auto detected as Intel Hex
avrdude: input file a.hex contains 1850 bytes
avrdude: reading on-chip flash data:

Reading : ##### : 100% 0.50s

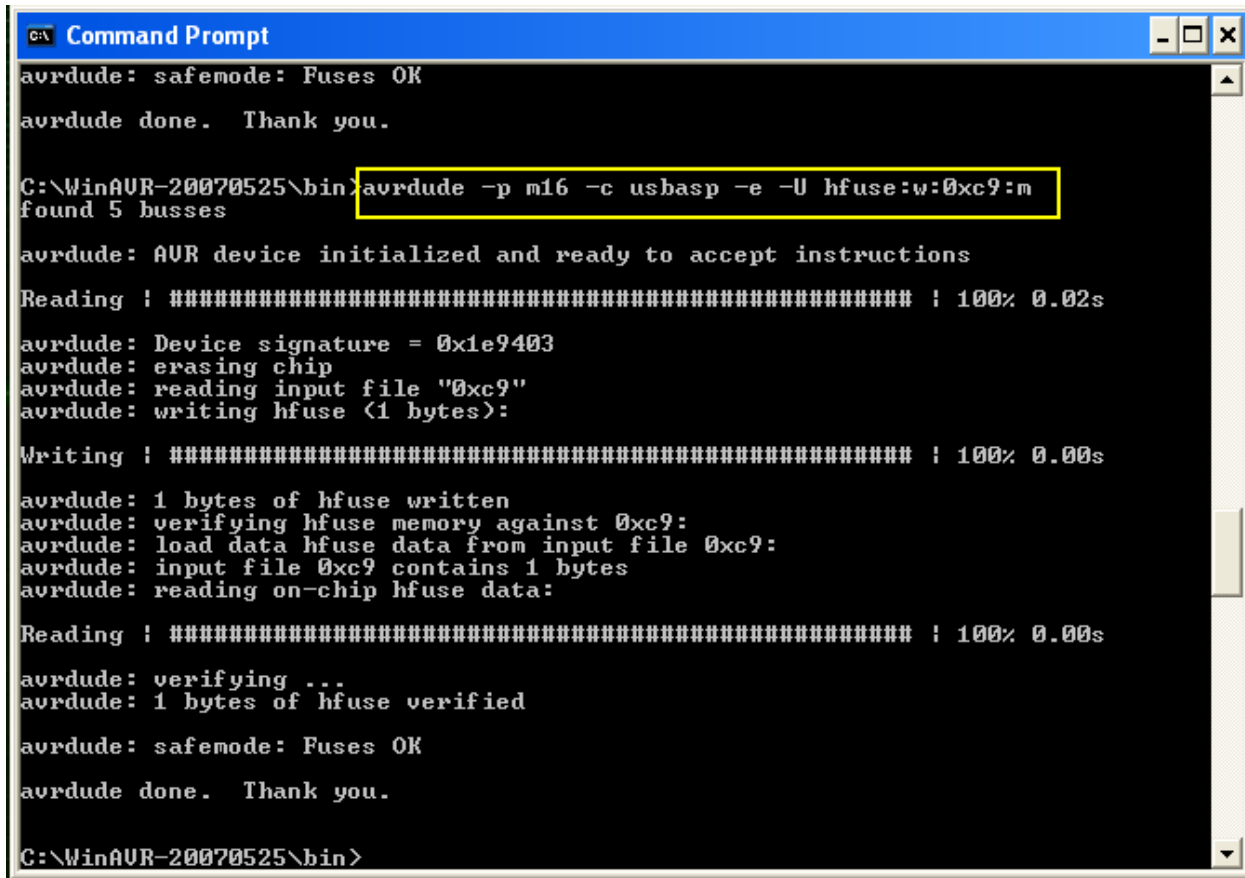
avrdude: verifying ...
avrdude: 1850 bytes of flash verified

avrdude: safemode: Fuses OK

avrdude done. Thank you.

```

6. To set the fuse bits follow the procedure given below.



```

avrdude: safemode: Fuses OK

avrdude done. Thank you.

C:\WinAUR-20070525\bin>avrdude -p m16 -c usbasp -e -U hfuse:w:0xc9:m
found 5 busses

avrdude: AVR device initialized and ready to accept instructions

Reading : ##### : 100% 0.02s

avrdude: Device signature = 0x1e9403
avrdude: erasing chip
avrdude: reading input file "0xc9"
avrdude: writing hfuse (1 bytes):

Writing : ##### : 100% 0.00s

avrdude: 1 bytes of hfuse written
avrdude: verifying hfuse memory against 0xc9:
avrdude: load data hfuse data from input file 0xc9:
avrdude: input file 0xc9 contains 1 bytes
avrdude: reading on-chip hfuse data:

Reading : ##### : 100% 0.00s

avrdude: verifying ...
avrdude: 1 bytes of hfuse verified

avrdude: safemode: Fuses OK

avrdude done. Thank you.

C:\WinAUR-20070525\bin>

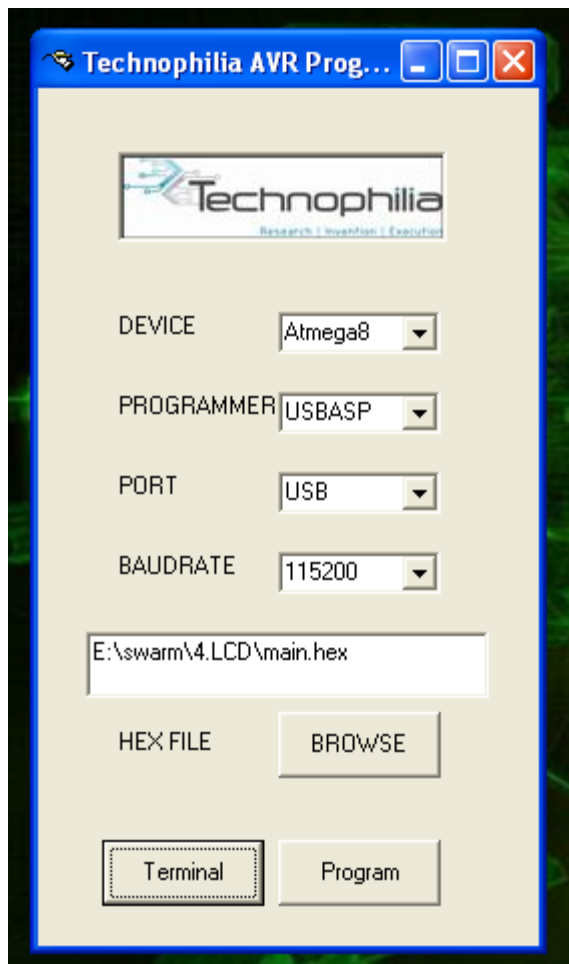
```

Technophilia's AVR programmer.

This programmer can easily used with Technophilia's programmer software easily.

The advantages of it are.

- Easy graphics user interface
- Not necessary to browse the hex code again and again
- You can open multiple windows at a time with different hex file
- It stores previously browsed hex file
- You can access most of the features of the controller by entering in to the terminal mode



To load the code click on browse select the hex file click on program.

Important Note

This device comes with a jumper connected backside which allows you to interface your programmer with a newly purchased microcontroller or with the microcontroller which works at a clock speed of below 8Mhz, but it reduces your program transfer speed if you are using a micro controller with clock speed of greater than 8 Mhz or a micro controller which is configured for working at a clock speed of greater than 8 Mhz than it is better to remove that jumper for achieving high program transfer speed.

You can again connect it if you are going to use a newly purchased micro controller or a low speed micro controller.

If you have any doubt please mail us at

info@technophilia.co.in

To know more about this programmer, go to the following links

www.technophilia.co.in

www.avrfreaks.net

www.avrbeginners.net

To download winAVR, go to the following link

www.winavr.sourceforge.net

