

AVR Standalone ISP Chip Copier V1.3 User Guide

The AVR Standalone ISP Chip Copier will read the contents of any device in the device list below and store the data in memory to be used for programming another AVR at a later time.

If the lock bits have been programmed in the target device, the AVR Chip Copier will not show an error. HOWEVER, the contents read from the target will be random data and not accurate.

The AVR Chip Copier is shipped with default configuration settings. Configuration consists of Lock/BLLB bits, and programming options. Default configuration settings will read/write Flash and Eeprom memory only. There will be no fuse or lock bits reading/writing in the default configuration. If you wish to change the configuration, a [USBtoTTL](#) converter or an [RS232toTTL](#) converter is needed as well as a terminal program.

When configuration is complete, you will be ready to use the programmer as a Stand Alone Chip Copier simply by setting the jumpers J4 (see Fig 2, Fig 3 & Fig 4 below) and J1.

***We recommend [teraterm](#) as other terminal programs may not display the menu text correctly.

Step 1. Setting up Tera-Term

Download TeraTerm from here:

<http://sourceforge.jp/projects/ttssh2/releases/>

TeraTerm works with Windows XP, Windows Vista, and Windows 7. Once installed, the program must be configured correctly.

Set the com port number,
baud rate should be 9600,
Data Bits = 8,
Parity = none,
Stop Bits = 1,
Flow Control = none,
Transmit Delay:
0 msec/char
0 msec/line.
(See Fig.1 below).

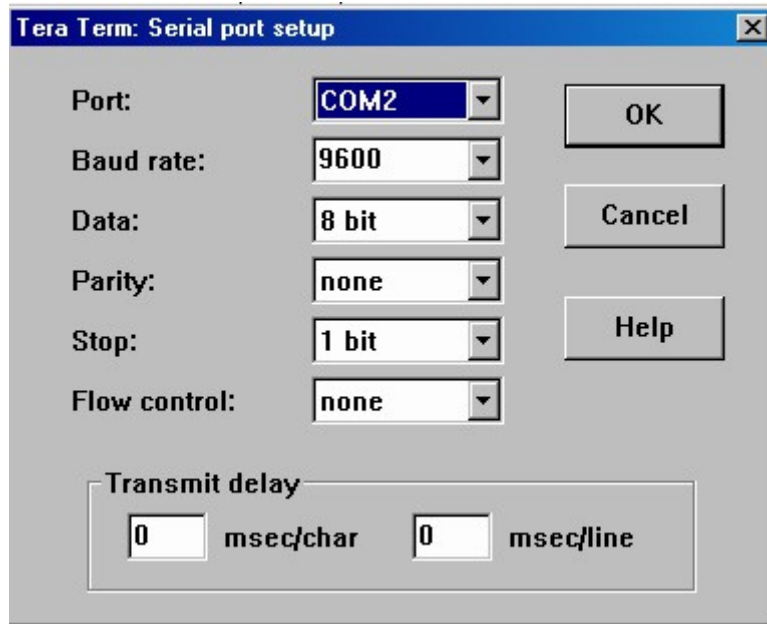


Fig.1

Step 2. Setting up Hardware.

A. Power Options

The Avr Chip Copier will operate correctly between 3.3 VDC and 5 VDC.

The programmer's power supply can be taken from the target (see Fig 2), or from a [USBtoTTL](#) converter (see Fig 3). If your conversion device does not supply power, or you are using an RS232toTTL converter, and the target also does not supply power, as a last resort power can be connected to the J5 pins (see Fig 4).

PLEASE BE AWARE that the programmer does not offer any regulation against over voltage or protection against short circuits or improper polarity. It is YOUR responsibility to connect an accurate power source, correctly. Failure to do so may destroy the programmer!



Fig 2.

Join programmer's power pin to target power (between 3 - 5 VDC).

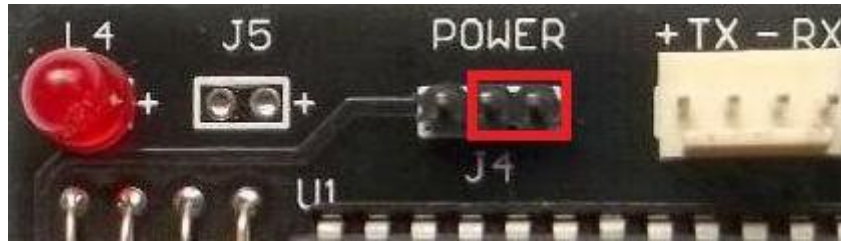


Fig 3.

Join programmer's power pin to USBtoTTL power supply.



Fig 4.

Supply power to J5. Applying power to J5 will supply the programmer with power only. If power is needed at the target and/or the conversion device, jumper J4 must be set accordingly. Typically you can join all three pins on J4 to join power pins on target, programmer, and conversion device together.

****Please observe correct polarity and voltage level (between 3 - 5 VDC).****

AGAIN THIS IS YOUR RESPONSIBILITY.

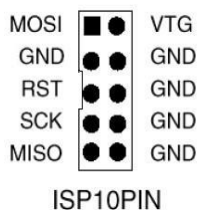
B. Connecting a Serial Interface

The only occasion when you will need to connect the programmer to your PC is to configure the programming options. The programmer outputs TTL voltage levels therefore a conversion device must be used as mentioned above. The 4 pin connector is used to communicate with a [USBtoTTL](#) converter or an [RS232toTTL](#) converter. The pins are labeled on the board. The pin description is as follows:

- "+" 3V or 5V input power to the programmer (positive)
- "-" Ground pin
- "TX" (input) connect to Transmit pin on the conversion device
- "RX" (output) connect to Receive pin on conversion device

C. Connecting to the Target

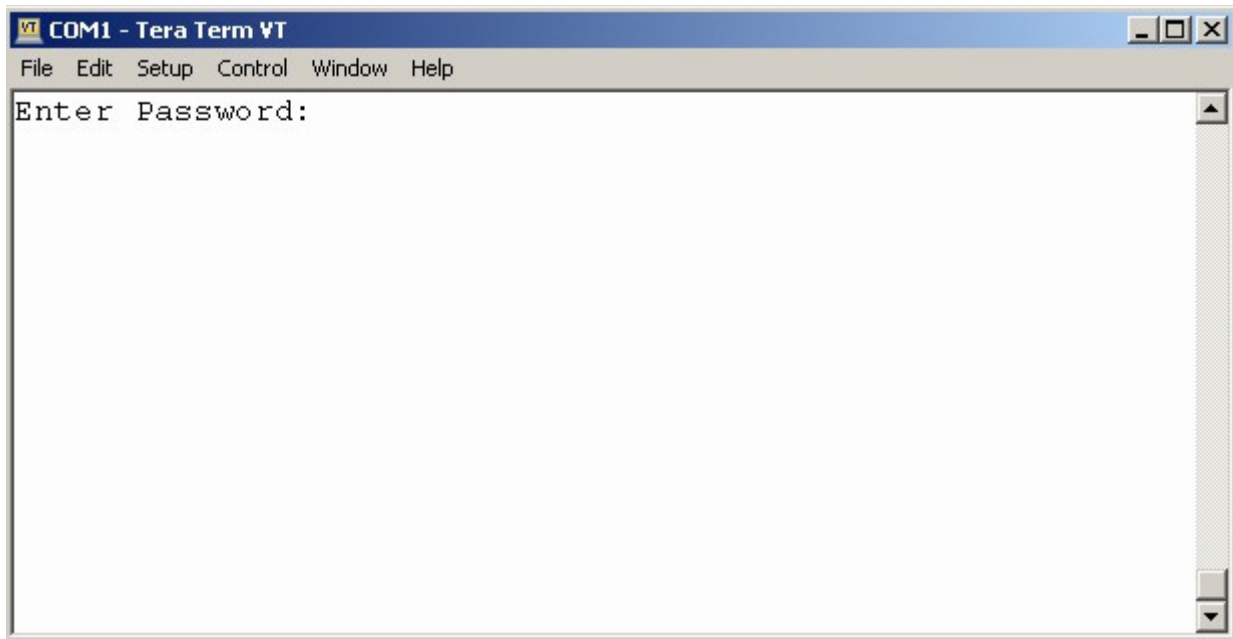
The Avr Chip Copier uses the ATMEL standard 10 pin ISP connector. Use the included ISP cable to connect to your target.



Step 3. Programmer Configuration.

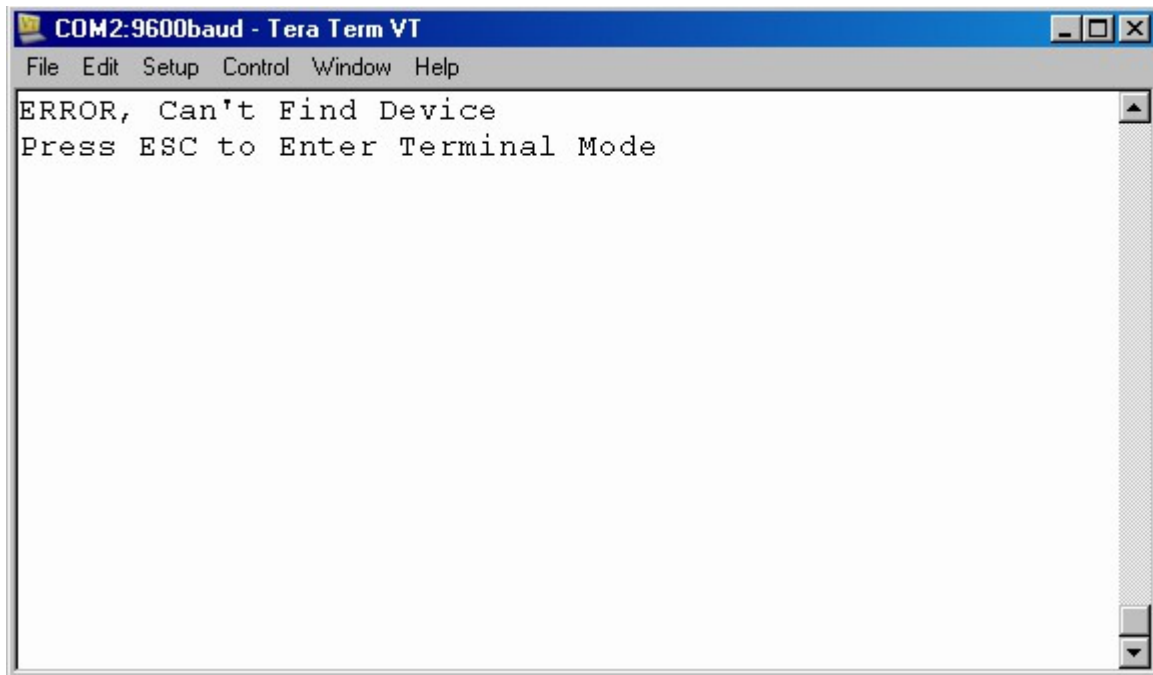
A. Ensure that the target is not connected. This is very important because the programmer will automatically begin to read/write the target (depending on jumper J1 setting) as soon as power is applied and will not enter the configuration menu.

B. Connect the programmer to a com port on your PC and start the terminal program. Apply power to the programmer. If everything is configured correctly you will see the following:

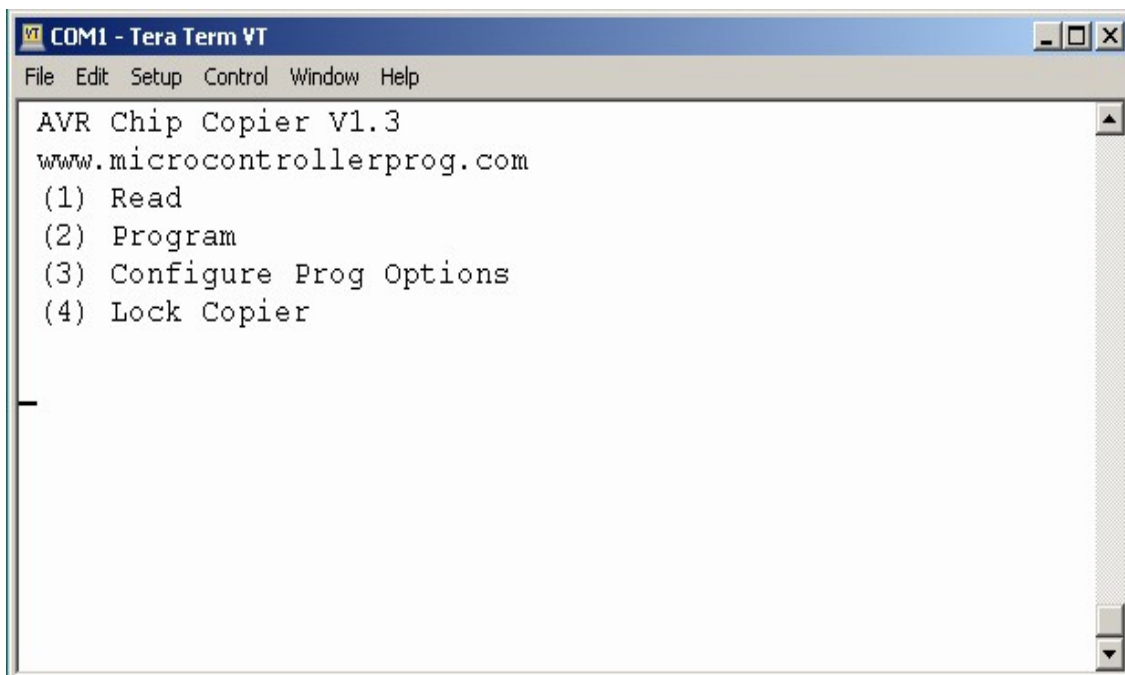


The password screen will only be displayed the very first time you power the AVR Chip Copier on. When the correct password is entered, you will never see this message again unless the AVR Chip Copier access is locked (see below). You will have three chances to enter the password. If the incorrect password is not entered within three tries, the AVR Chip Copier will remain locked until you reboot.

When the password has been accepted you will see a screen like this:

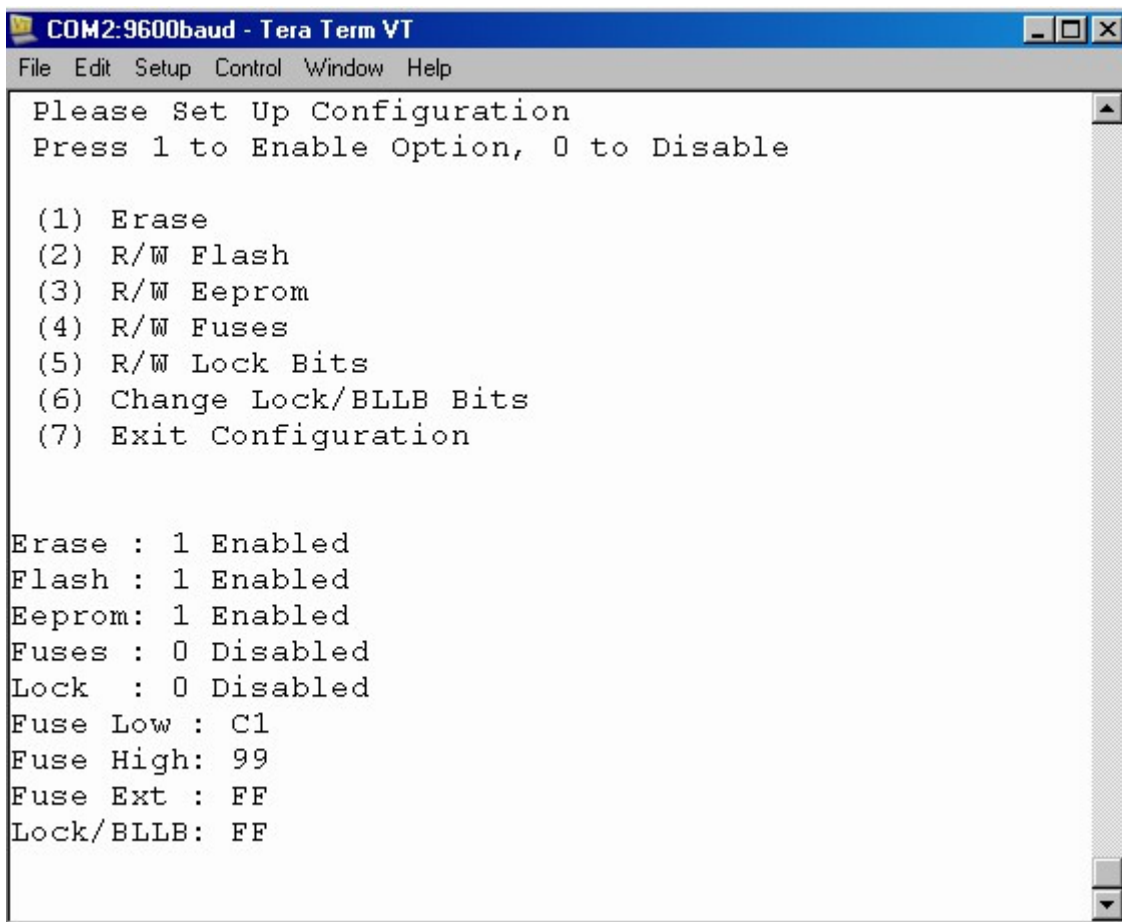


Pressing "ESC" will bring you to the configuration menu:



Now you have 4 options. By pressing 4, all access to the programmer is locked. To regain access the AVR Chip Copier must be restarted (disconnect & reconnect power). Now enter the correct password.

Next, configuration is necessary. Pressing 3 will bring up a screen like this:

A screenshot of a terminal window titled "COM2:9600baud - Tera Term VT". The window has a menu bar with "File", "Edit", "Setup", "Control", "Window", and "Help". The main text area displays a configuration menu. At the top, it says "Please Set Up Configuration" and "Press 1 to Enable Option, 0 to Disable". Below this is a list of seven options: (1) Erase, (2) R/W Flash, (3) R/W Eeprom, (4) R/W Fuses, (5) R/W Lock Bits, (6) Change Lock/BLLB Bits, and (7) Exit Configuration. At the bottom of the screen, the current configuration is displayed: Erase : 1 Enabled, Flash : 1 Enabled, Eeprom: 1 Enabled, Fuses : 0 Disabled, Lock : 0 Disabled, Fuse Low : C1, Fuse High: 99, Fuse Ext : FF, and Lock/BLLB: FF.

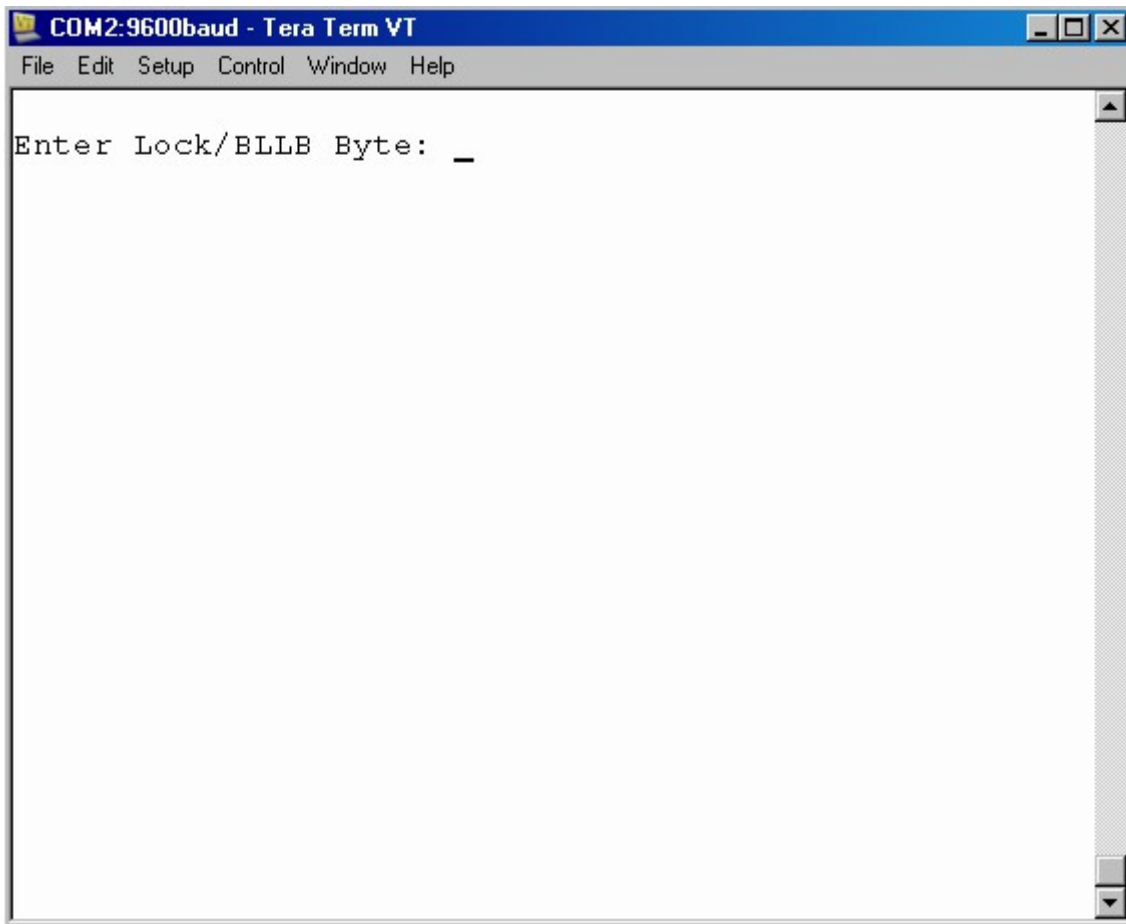
Now you have 7 options. At any time you can exit to the main menu by pressing 7 or by removing power to the programmer. Options 1 through 5 will establish which steps will be taken when the programmer is in read mode or write mode (depending on J1 setting, J1 off = read mode, J1 on = write mode).

To enable or disable any of the functions, a "1" or "0" must be pressed after the option is selected. For example, to skip erasing the device before programming, press 1, then 0 on your keyboard. The menu will immediately update the option you have selected and return to the configuration menu. To enable the erase option before programming, you will press 1, then 1.

The Avr Chip Copier will read the lock bits from the target and store them in memory until programming if the option (5) above is enabled. If it is necessary for the lock bits to be programmed

(secured) in write mode, the lock bits can be modified by selecting option (6).

Option 6 will bring up a screen like this:



Here you must enter an 8 bit hex value (2 digits, not case sensitive). The screen will then revert back to the configuration menu with the updated lock byte.

*****Please remember that if the target is read again, while the lock option is enabled, the modified lock byte will be overwritten.***** If the lock bits are to be changed from what is programmed in the target, do so after the target has been read.

*****A WARNING regarding fuse bits*****

The Avr Chip Copier will read the fuse values from the target and store them in memory until programming if the option above (4) is enabled. If you wish to enable the option please be 100% certain that the correct fuse bits are being displayed in the menu. If the option was disabled, then later enabled without reading the target, the stored values will be written to the device possibly locking you out of the target!!!

Please be certain that when the fuse option is enabled, you read the target before writing. This is very important!
(you could write a strange value to all fuses and lock yourself out of the target!)

The erase command does not change the fuse values, therefore with any future software updates to your target, the fuse writing option can be disabled to speed up the programming process.

To exit and return back to the main menu, press 7.

Step 4. Using Your Avr Chip Copier.

The target can be read or programmed in terminal mode from the main menu (while the programmer is connected to the PC) or in standalone mode (if jumper J1 is installed before the device is powered up). When reading or programming, the programmer will execute the options you have configured previously.

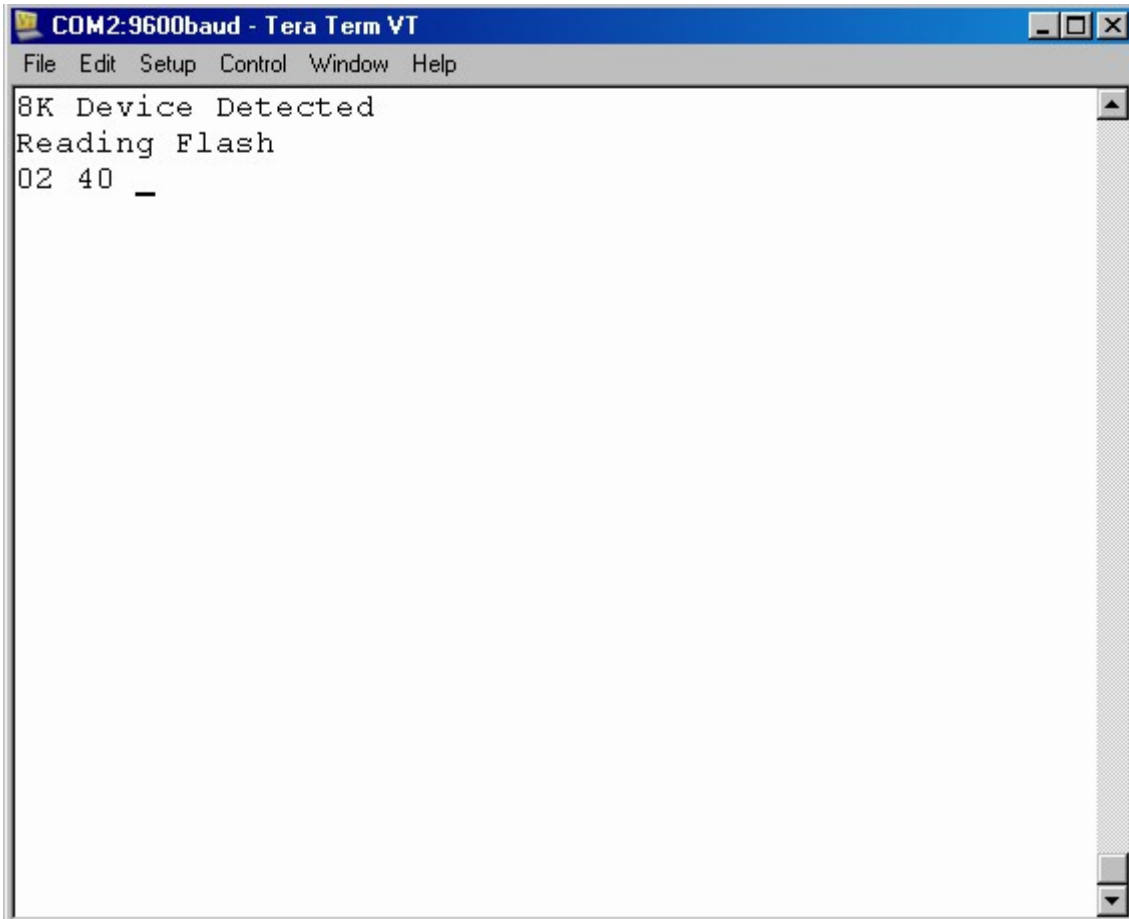
Steps required to copy an avr chip:

- A. Configure the Avr Chip Copier for read mode by removing jumper J1.
- B. Remove Jumper J1 to put the Avr Chip Copier in read mode.
- C. Install Jumper J4 correctly depending on where power will be taken from.

Once power is connected, the AVR Chip Copier will begin to read the contents of the target.

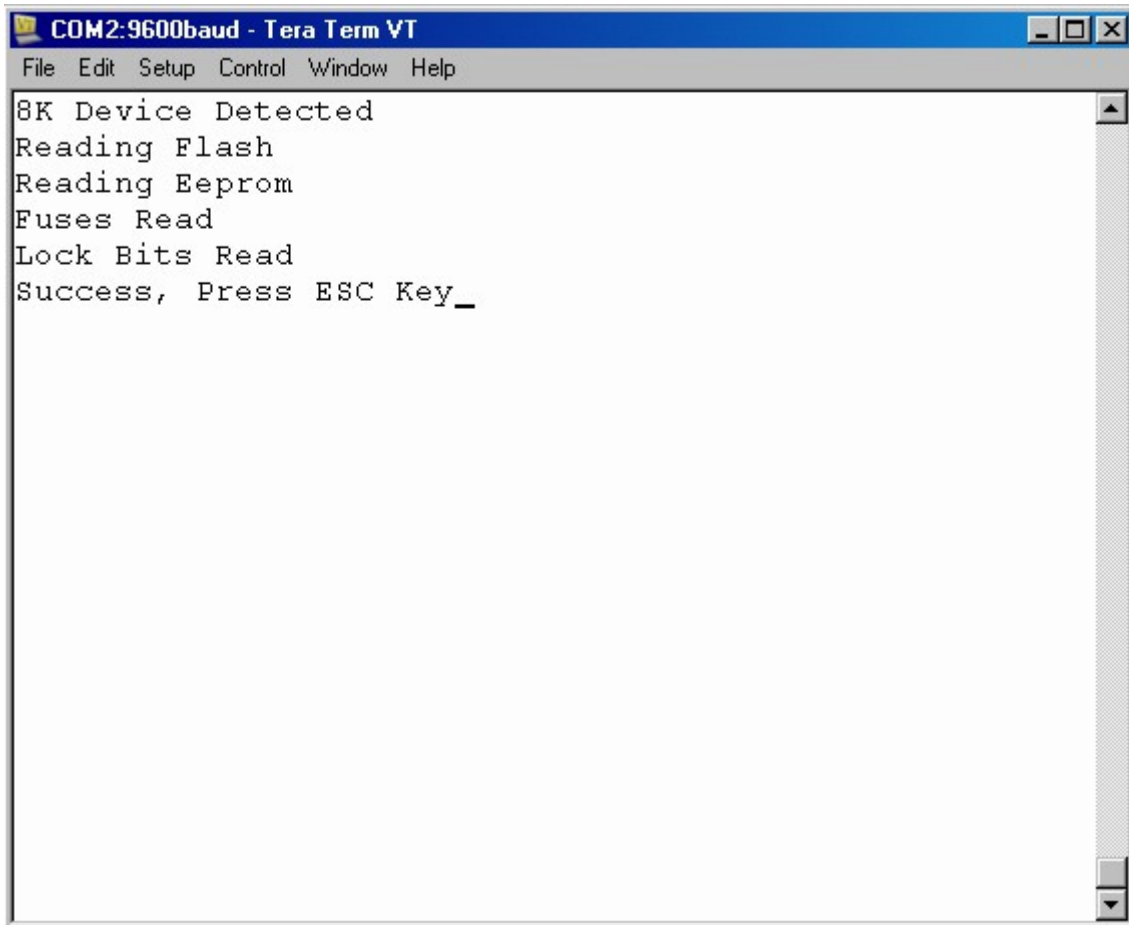
If the lock bits have been programmed in the target device, the AVR Chip Copier will not show an error. HOWEVER, the data read will be random and not accurate.

Here is a screen shot of the reading operation while connected to the PC (either by pressing (1) from the main menu or by powering up the device with jumper J1 removed):



While reading (in standalone mode or terminal mode) the yellow LED will illuminate indicating the programmer is busy reading. The reading address will be displayed on the screen. The address may not display correctly if a terminal program is being used other than Hyperterminal or TeraCopy. When complete, the green led will illuminate if there are no errors (in the case of errors, the red led will illuminate).

If successful, you will see this message:



D. Now you can enter the configuration menu if you wish to change the lock bits other than the value that was read from the target.

E. Remove jumper J4 from the AVR Chip Copier. (disconnect power)

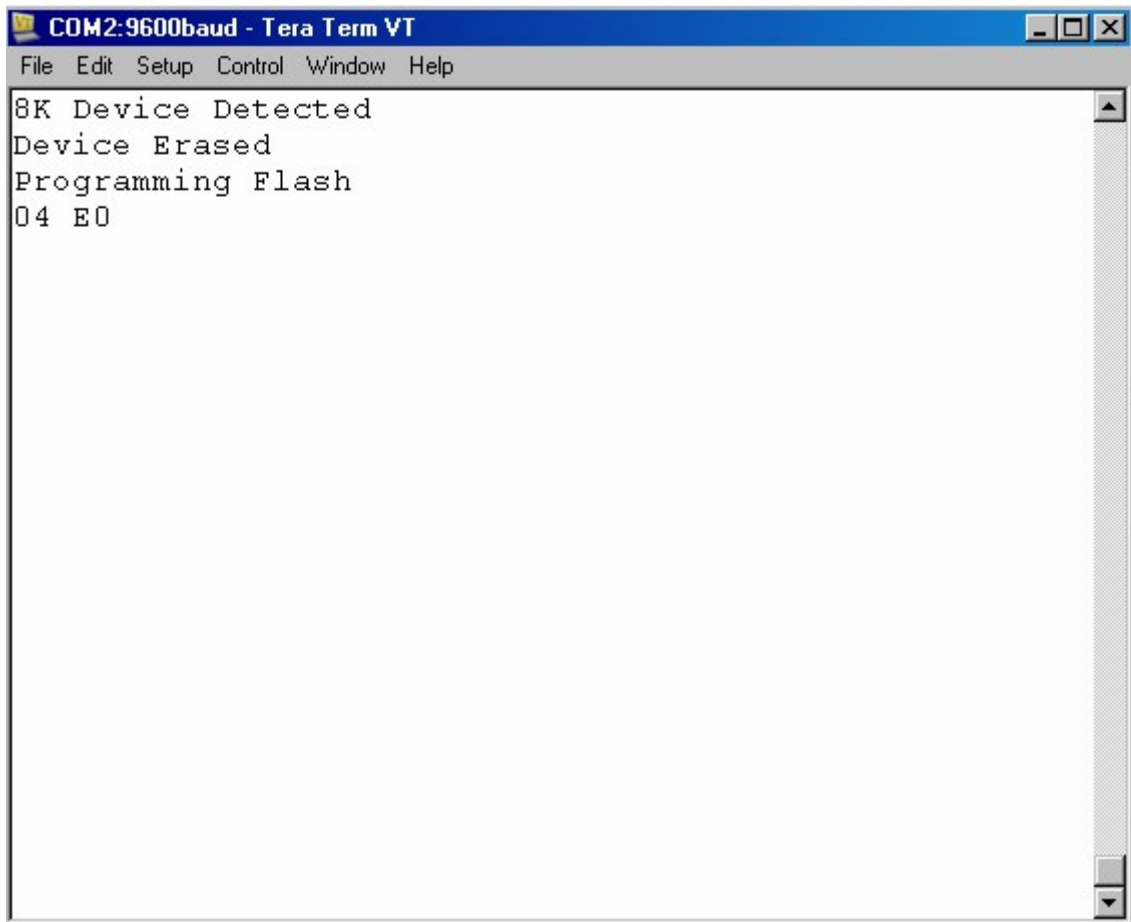
F. To program another avr, Install jumper J1.

G. Ensure that the Avr Chip Copier is connected to the target correctly.

F. Install jumper J4. (apply power)

The AVR Chip Copier will begin to program the target based on the options you have configured. In order to speed up the programming process, the Avr Chip Copier will not program the value 0xFF. It will however program every single byte of the eeprom (including the value 0xFF) in case the erase option is disabled and only the eeprom is being copied/programmed.

Here is a screen shot of the programming operation while connected to the PC (either by pressing (2) from the main menu or by powering up the device with jumper J1 installed):

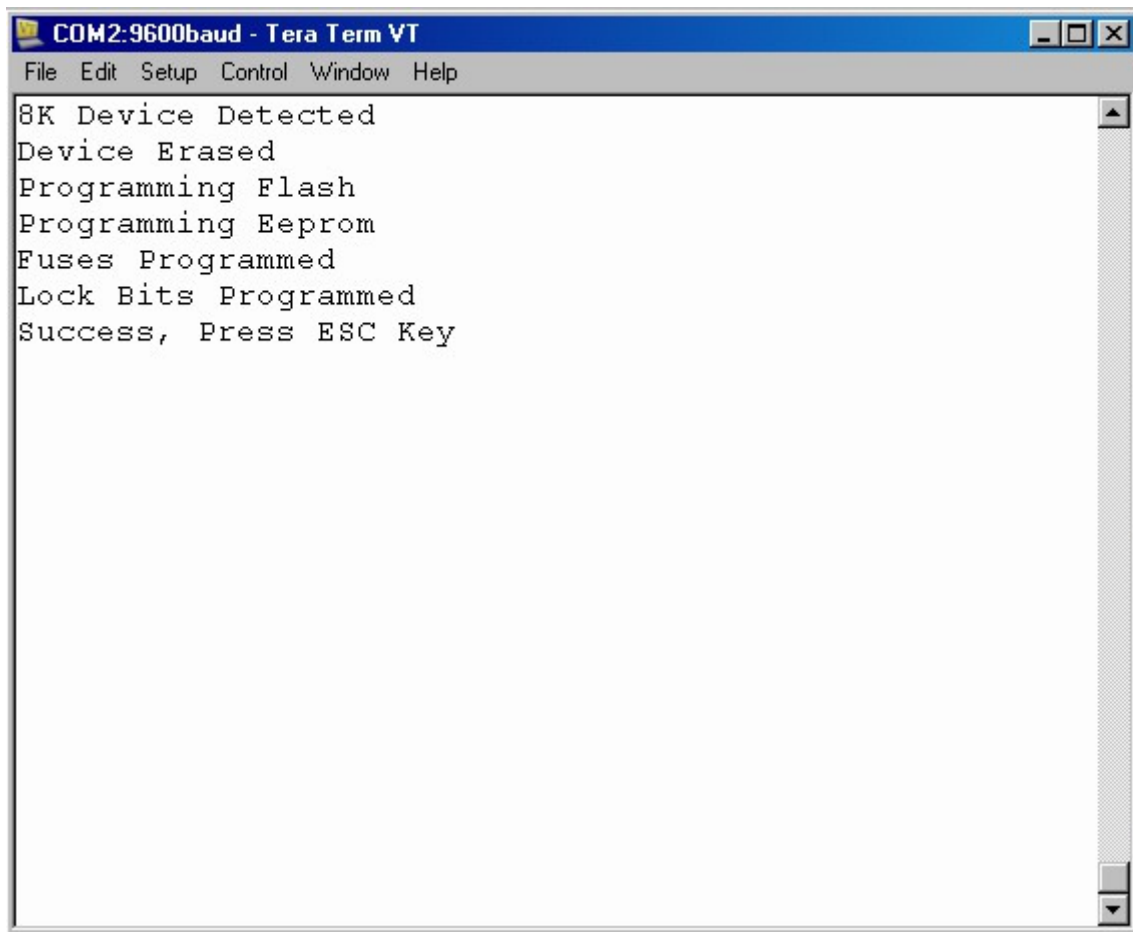


```
COM2:9600baud - Tera Term VT
File Edit Setup Control Window Help
8K Device Detected
Device Erased
Programming Flash
04 E0
```

While programming (in standalone mode or terminal mode) the yellow LED will illuminate indicating the programmer is busy programming.

The programming address will be displayed on the screen. The address may not display correctly if a terminal program is being used other than Hyperterminal or TeraCopy. The programmer will automatically verify every byte as it is written. When complete, the green led will illuminate if there are no verify errors (in the case of errors, the red led will illuminate).

If successful, you will see this message:



You will need to rely on the LED's to view programming status in standalone mode since there will be no terminal connected. At this point you can press any key to return to the main menu (if in terminal mode) or disconnect the power to the programmer if in standalone mode.

Device List for AVR Chip Copier V1.3

<u>ATmega Devices</u>		<u>ATtiny Devices</u>
ATmega8	ATmega324	ATtiny13
ATmega8L	ATmega324V	ATtiny13A
ATmega16	ATmega325	ATtiny13V
ATmega16L	ATmega325V	ATtiny24
ATmega32	ATmega3250	ATtiny24V
ATmega32L	ATmega3250	ATtiny26
ATmega48	ATmega325	ATtiny26L
ATmega48PB	ATmega325A	ATtiny261
ATmega48V	ATmega325P	ATtiny261V
ATmega88	ATmega325PA	ATtiny461
ATmega88PB	ATmega3250	ATtiny461V
ATmega88V	ATmega3250A	ATtiny861
ATmega8515L	ATmega3250P	ATtiny861V
ATmega8515	ATmega3250PA	ATtiny43U
ATmega8535	ATmega328	ATtiny44
ATmega8535L	ATmega328P	ATtiny44V
ATmega162	ATmega329	ATtiny48
ATmega162V	ATmega329A	ATtiny88
ATmega164	ATmega329P	ATtiny84
ATmega164A	ATmega329PA	ATtiny84V
ATmega164P	ATmega3290	ATtiny25
ATmega164PA	ATmega3290A	ATtiny25V
ATmega164V	ATmega3290P	ATtiny45
ATmega165A	ATmega3290PA	ATtiny45V
ATmega165P	ATmega64	ATtiny85
ATmega165PA	ATmega64L	ATtiny85V
ATmega169A	ATmega64A	ATtiny87
ATmega169P	ATmega640	ATtiny167
ATmega169PA	ATmega644	ATtiny2313
ATmega168	ATmega644A	ATtiny2313A
ATmega168PB	ATmega644P	ATtiny2313V
ATmega168V	ATmega644PV	ATtiny4313
ATmega164A	ATmega645P	
	ATmega6450P	

For support or any questions, please email:
microcontrollerprog@yahoo.com