Contents

Table of Contents

Attiny Fuse Repair User Guide for Version 3.X Firmware

The ATTiny Fuse Repair Programmer will restore your corrupted ATTiny device to factory defaults less than one second without the use of a computer.

Simple to use, there are only two possible outcomes. Yellow LED (ERROR) or Green LED (Success). It just doesn't get easier than that.

No more throwing away locked microcontrollers, just fix them all in a snap!

In this version of firmware, we have added the ability to program fuses in devices that are not in the supported devices list. As long as the device you are trying to restore uses the same High Voltage Serial Programming method, the programmer will work for you.

Also available is the ability to skip the erase command, leaving your code intact. Just simply install jumper J1 when the power is disconnected from the programmer, then power on the programmer. The fuse restore programmer will write the factory default fuses to your attiny and skip the erase function at the same time.

Please follow the instructions carefully. We are not responsible for any damage caused if the programmer is misused.

To use the programmer, simply insert your ATTiny in the correct socket and connect power. Within one second either the green or the yellow led will illuminate. Green LED indicates success, Yellow LED indicates failure. Failure can be caused by a number of reasons.

1. **You do not have the correct power supply. This is 99% the cause of all failures. Please use a 13.5 - 16 Volt DC power supply with positive center on the plug. It is very important that you measure the output from your power supply using an accurate voltmeter! We are not responsible if you damage the programmer, or anything connected to it. Incorrect voltage or polarity may cause damage to you and/or the programmer!**

2. **You are not inserting the chip in the correct socket. Please see the instructions in Table 3 below.**

3. **The device signature cannot be read. Is the device damaged? Is it inserted correctly?**

4. **The device is not supported. Please check the device list (Table 3) below. If the programmer does not reprogram your attiny and there are no led's illuminated, perhaps you have a damaged power supply. Once power is connected, the red led will illuminate to indicate power has been applied. If it does not, you are not using the correct power supply. If you are getting errors continuously (yellow LED illuminated) you can connect a MAX232 circuit to the debug port as shown in Fig 1 below.**

5. **Do you have jumper J1 installed? If the lock bits have been programmed and J1 is installed, the programmer will fail. The attiny must be erased if the lock bits have been programmed. Please remove jumper J1 and power cycle the programmer, and try again.**

Please Remember: This device is not designed to circumvent LOCK BITS!

Mode 3 Advanced Mode

New in Version 3, mode 3 or "advanced mode". This mode works with a terminal on your PC. It allows reading and writing of the fuse bytes. Also included is a new feature that disables signature verification, allowing the user to program fuses on avr devices that are not in the device list (Table 3).

WARNING: We are not responsible for damage caused by programming non standard fuse values. KNOW WHAT YOU ARE DOING!

****The fuse programmer supplies the MAX232 with 5 Volts.**

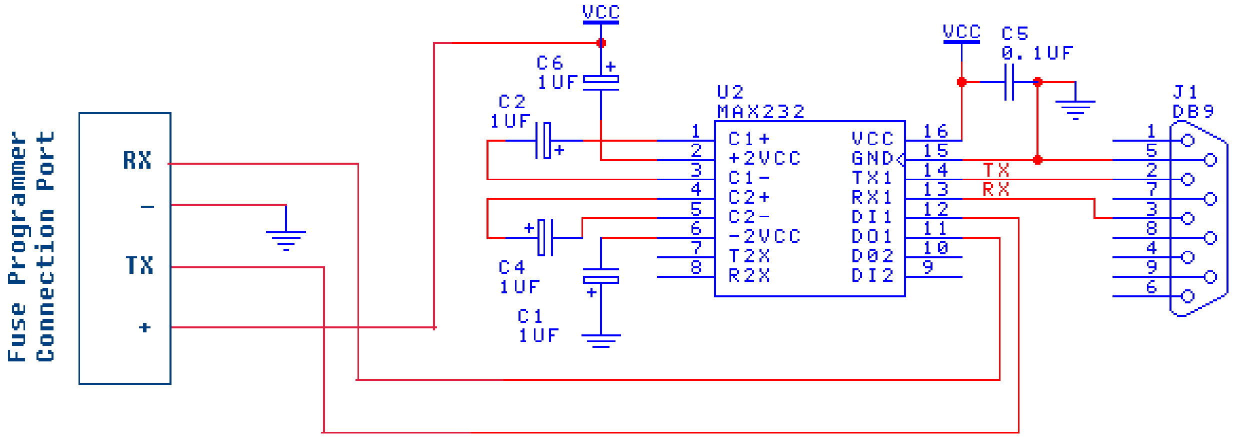


Fig 1. Connecting the Programmer to the Computer's Serial Port

When connected, the programmer will display fuse information or error messages to a terminal at 9600 baud.

Hyperterminal settings:
Baud Rate: 9600,
Data Bits: 8,
Stop Bits: 1,
Parity: None,
Flow Control: None,
Terminal Emulation: VT-100 (ANSI).

A low Cost Serial to TTL Converter can be found on our website here.

Attiny Fuse Repair Programmer	RS232toTTL Converter
RX	RX
TX	TX
-	GND
+	VCC

Table 1. Connecting the RS232toTTL module to the ATTiny Fuse Restore Programmer

If you do not have a serial port on your PC, the programmer can be connected to one of our low cost USBtoTTL adapters found on our website here. See Table 2 below for connection details.

Attiny Fuse Repair Programmer	USBtoTTL Converter
RX	TX
TX	RX
-	GND
+	5V

Table 2. Connecting the Programmer to the Computer's USB Port using the USBtoTTL.

The ATTiny Fuse Repair Programmer supports the following devices:

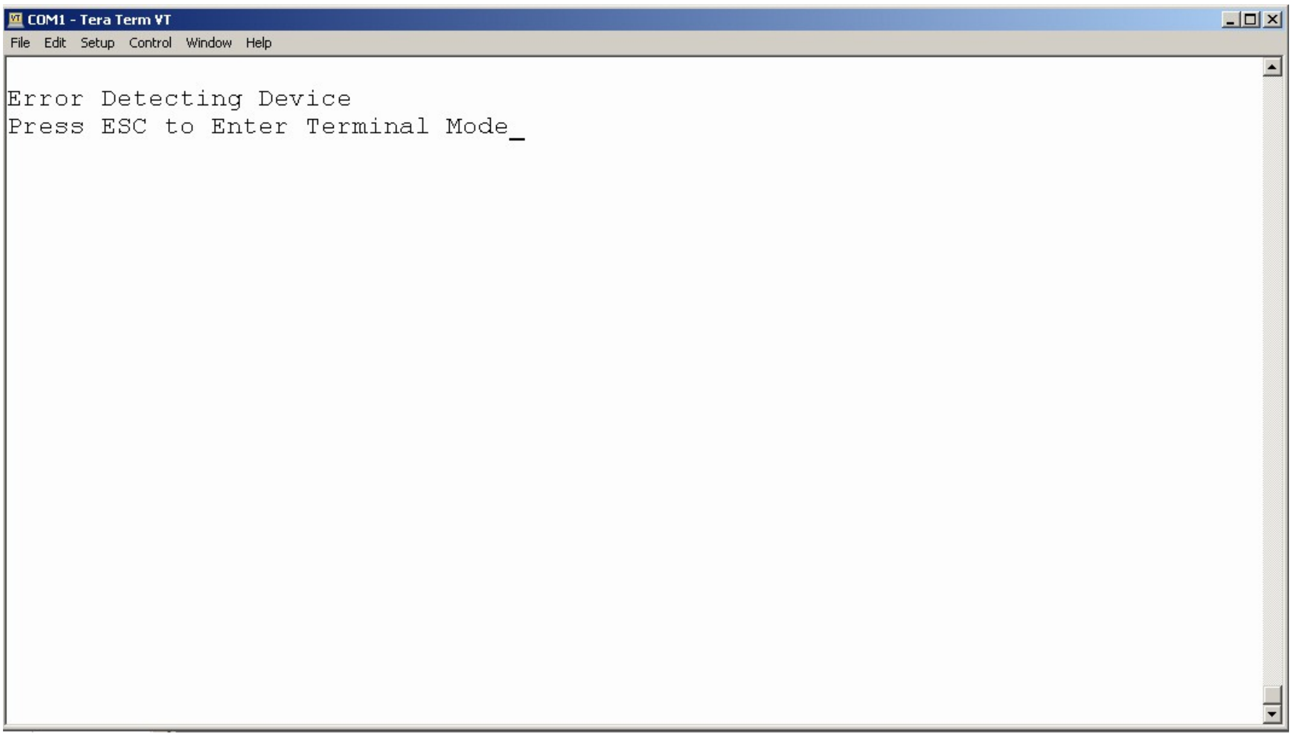
Socket U5	Socket U6
Attiny11***	ATtiny24
ATtiny12	ATtiny24V
ATtiny12V	ATtiny44
ATtiny12L	ATtiny44V
ATtiny13	ATtiny84
ATtiny13A	ATtiny84V
ATtiny13V	
ATtiny25	
ATtiny25V	
ATtiny45	
ATtiny45V	
Attiny85	
ATtiny85V	

Table3. Device List

*****While the ATtiny11 does not support ISP programming, this programmer will restore the fuses back to factory settings.**

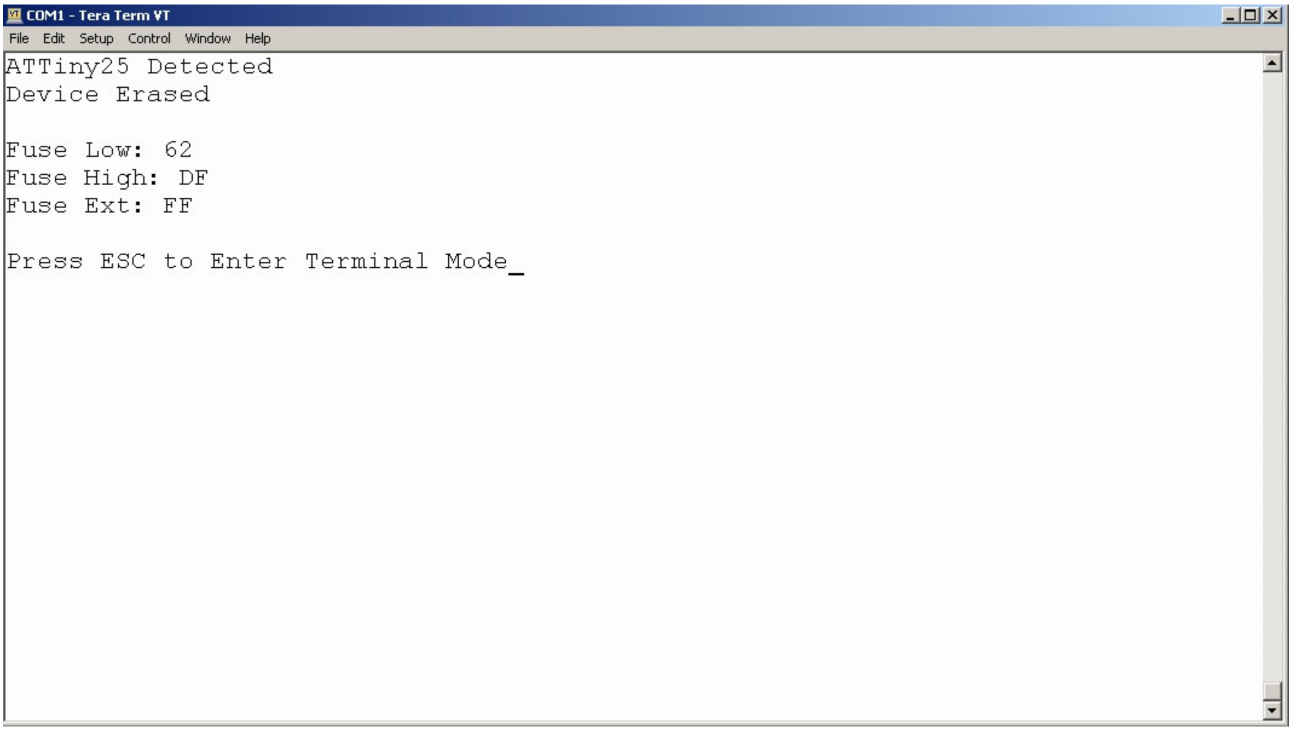
Using the Programmer

If the programmer is connected to the terminal, and it is powered up with no devices inserted, you will see a screen like this:



```
Error Detecting Device
Press ESC to Enter Terminal Mode_
```

If a device was inserted, you will see this message:

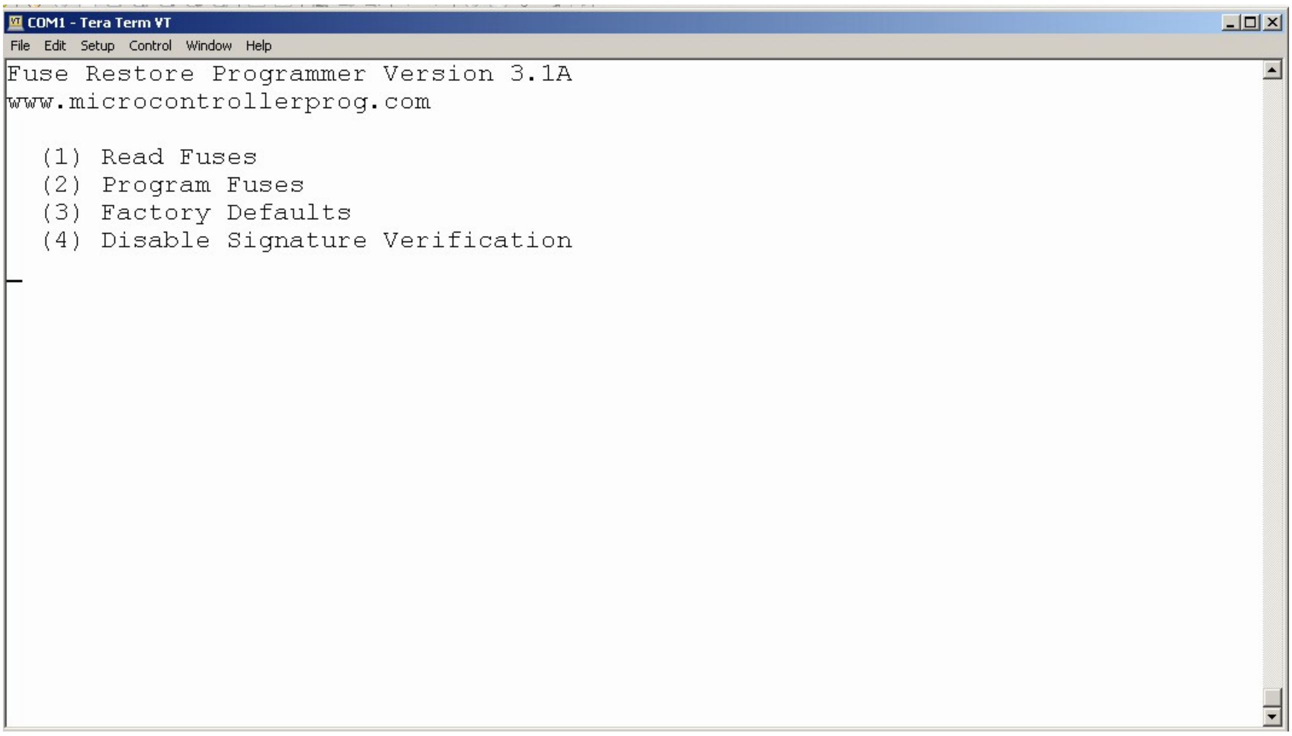


```
ATTiny25 Detected
Device Erased

Fuse Low: 62
Fuse High: DF
Fuse Ext: FF

Press ESC to Enter Terminal Mode_
```

Pressing ESC gets you into MODE3 (Advanced Mode). You will then see a screen like this:



```
Fuse Restore Programmer Version 3.1A
www.microcontrollerprog.com

  (1) Read Fuses
  (2) Program Fuses
  (3) Factory Defaults
  (4) Disable Signature Verification

_
```

You now have Four Options:

1. Read Fuses. Have you ever been curious to know what the fuse values are when your attiny device has been corrupted? Pressing "1" will show you the fuses. Then pressing any key will always return you back to the main menu.

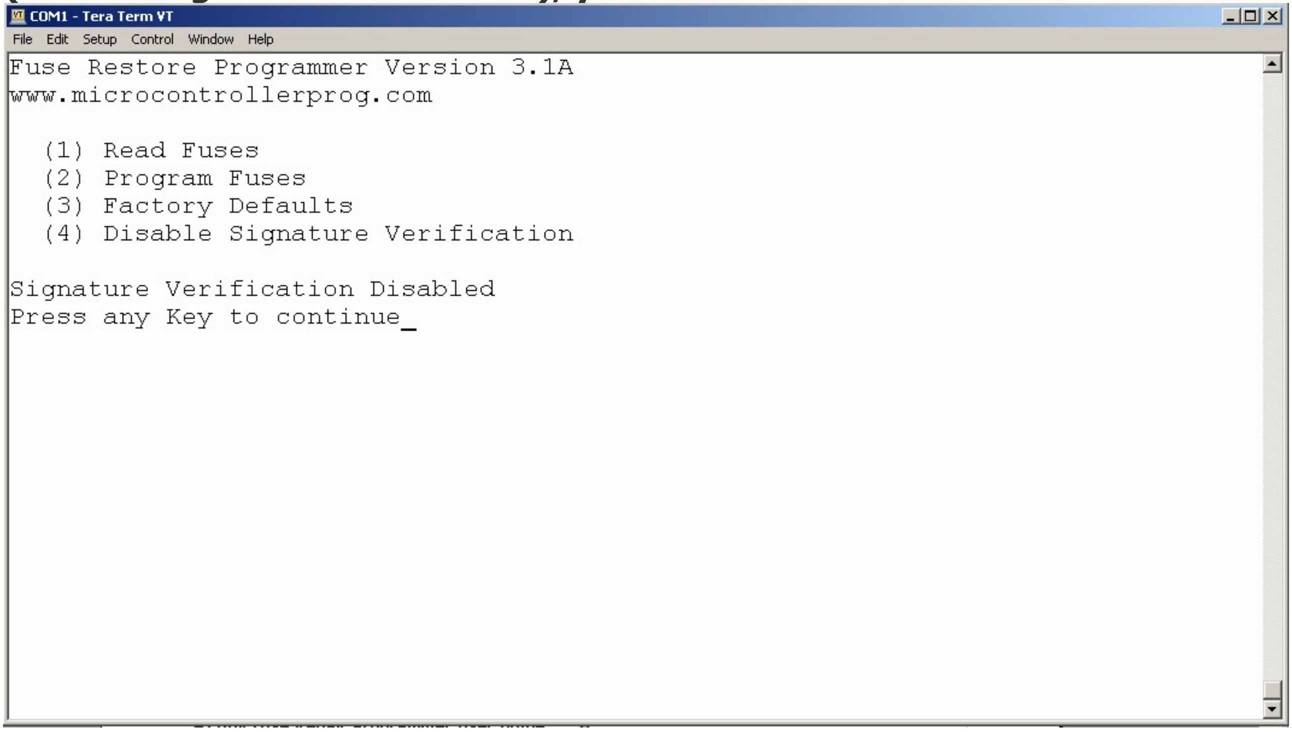
2. Pressing "2" will allow you to program the fuses on any of the devices in the device list. If you would like to program any device NOT in the device list, you must select "4" (Disable Signature Verification) before pressing "2", Program Fuses.

AGAIN, PLEASE Be careful here! NON Standard fuse values may cause unknown results preventing your device from entering programming mode, even with this programmer!!
BE SURE YOU KNOW WHAT YOU ARE DOING!!

If signature verification is disabled, the programmer does not know any information about the device it is trying to program. (i.e. The default fuse values, or whether or not the device has high, low or extended fuse bytes or all three).
When programming a device (i.e. a mature avr device such as the 90S series) you will be asked for all three fuse bytes. Even though the device has only one fuse byte, you must enter the same fuse byte for all three fuse bytes.

For example, you want to restore the SPIEN fuse on a AT90S2323 and disable the FSTRT fuse at the same time, enter hex value DF (not case sensitive) for all three values (high fuse, low fuse, and extended Fuse bytes). The device will ignore the extra fuse bytes. After programming, the fuse repair programmer will read back three fuse bytes. This mode does not work for restoring factory defaults since the information is unknown as to what the default values are based on the device's signature.

The only way to Enable Signature Verification (if you disabled it) is to restart the programmer. (unplug power and reconnect). Signature verification is always enabled by default. When pressing "4" (Disable Signature Verification), you will see a screen like this:



```
Fuse Restore Programmer Version 3.1A
www.microcontrollerprog.com

  (1) Read Fuses
  (2) Program Fuses
  (3) Factory Defaults
  (4) Disable Signature Verification

Signature Verification Disabled
Press any Key to continue_
```

3. Factory Defaults selection "3".
Pressing "3" will restore the fuses to factory defaults, same as in standalone mode. If signature verification was disabled, it will be re-enabled by selecting factory defaults.

You can exit Advanced mode by powering off the programmer.

Any questions? Email us at: microcontrollerprog@yahoo.com