

# HVSP Fusebit Programmer

von guloshop.de



## Funktionsumfang

Willkommen beim HVSP Fusebit Programmer!

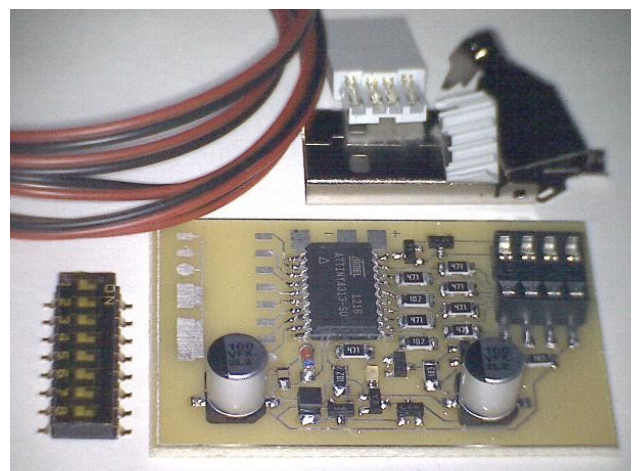
Wie der Name bereits verrät, können Sie damit Fuse-Bits (und sogar Lockbits) der Atmel-AVR-Typen verändern, die sich per HV (hohe Spannung) seriell programmieren lassen: ATtiny13A, ATtiny25, ATtiny45, ATtiny85, ATtiny24A, ATtiny44A, ATtiny84A und dazu kompatibel. Da Lock-Bits nicht einzeln gelöscht werden können, beherrscht der HVSP Fusebit Programmer auch die Chip-Erase-Funktion.

Die Mikrocontroller-Typen mit 8 Anschlussbeinen können direkt auf die Platine gesteckt werden, für den Anschluss der Typen mit 14 Anschlussbeinen ist ein einfacher optionaler Adapter erforderlich. Weiter hinten in diesem Dokument finden Sie eine Bauanleitung dazu.

## Lieferumfang

Vor dem Aufbau prüfen Sie bitte, ob alle Komponenten vorhanden sind:

- eine Platine mit aufgelöteten Bauteilen
- ein 8-poliger DIP-Schalter
- eine zweiadrige Anschlussleitung für die 5-V-Stromversorgung (Farbe Rot/Schwarz)
- ein USB-Stecker (Typ A), ebenfalls für die Stromversorgung
- das Dokument, das Sie gerade lesen

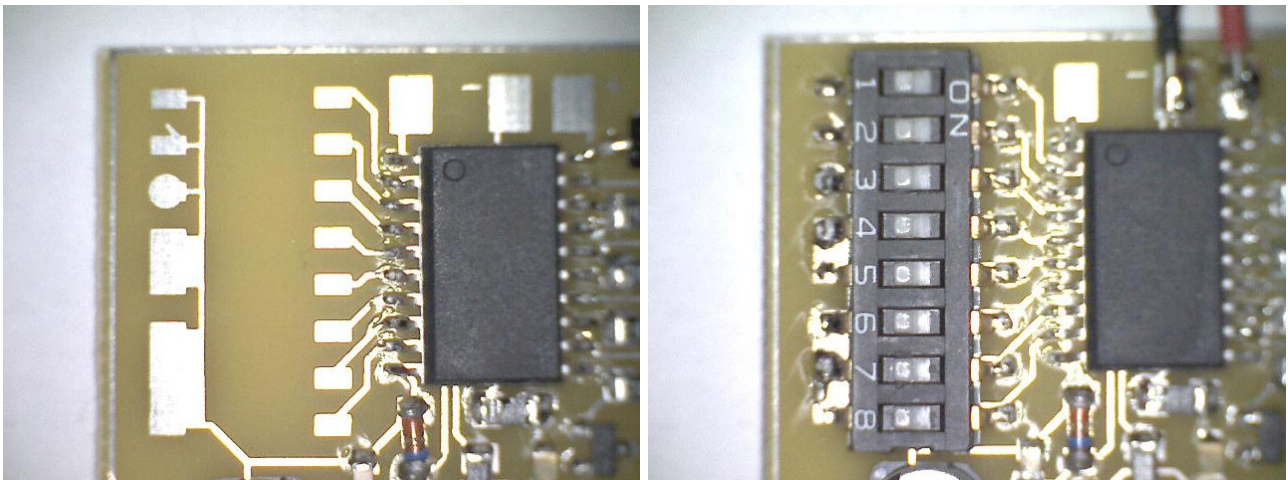


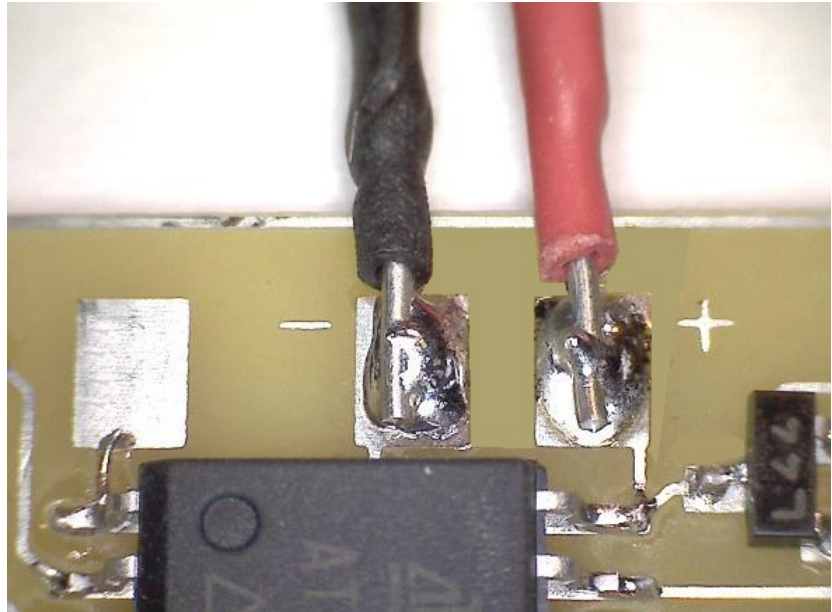
## Benötigte Werkzeuge

Für den Zusammenbau des HVSP Fusebit Programmiers brauchen Sie einen LötKolben, LötZinn und etwas LötErfahrung. Für das Abisolieren der Leitungen verwenden Sie am besten ein kleines Messer oder eine geeignete Zange. Später wird ein kleiner Schraubendreher sehr nützlich sein, wenn es darum geht, die kleinen Schiebeschalter zu bewegen und programmierte Mikrocontroller wieder aus der Fassung zu entfernen.

## Zusammenbau

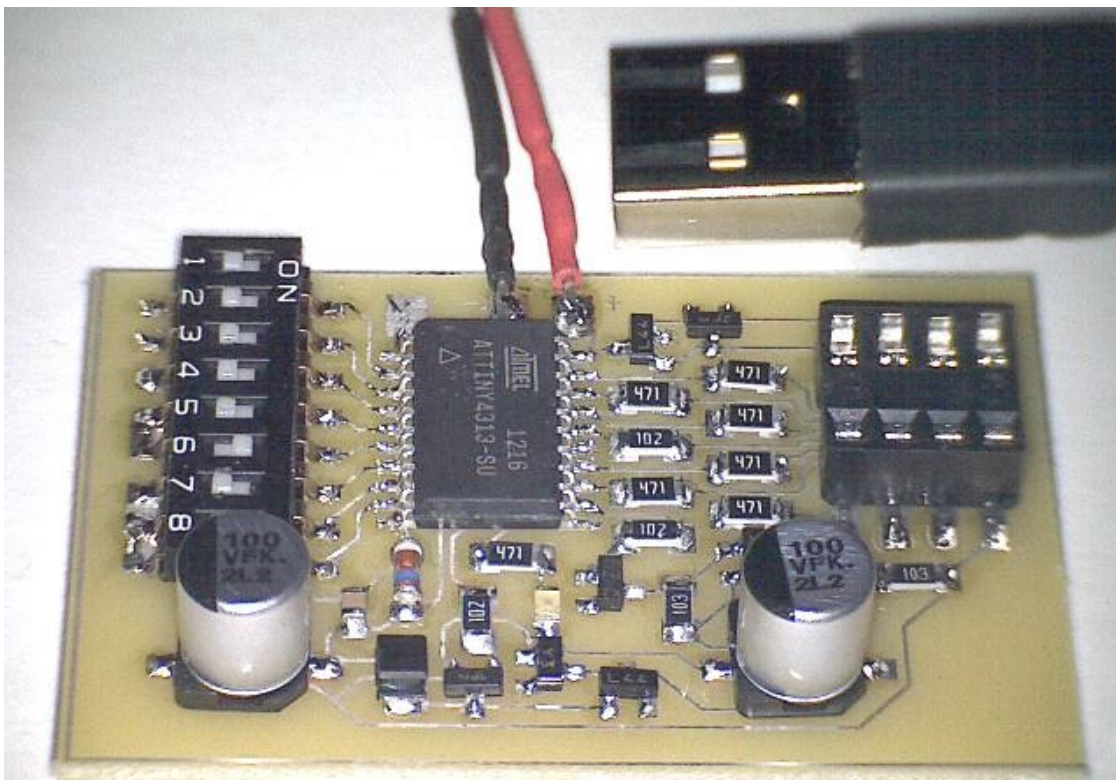
Setzen Sie den 8-poligen DIP-Schalter so auf die Platine, wie in der Abbildung zu sehen. Achten Sie darauf, dass sich Schalter Nummer 1 oben befindet (siehe Bild) und dass die 16 Anschlussbeinchen genau auf den zugehörigen Pads sitzen. Beim LötEBeginnen Sie am besten mit zwei gegenüber liegenden Anschlüssen und prüfen dann noch einmal, ob die Position des DIP-Schalters stimmt. Dann löten Sie die restlichen Anschlüsse an die Platine.





Als nächstes entfernen Sie die Isolierungen an beiden Enden der beiliegenden Anschlussleitung um jeweils ca. 2 Millimeter. Danach löten Sie die Anschlussleitung an den USB-Stecker. Achten Sie unbedingt auf die richtige Polung, die beiden Leiter dürfen nicht vertauscht werden!

Die anderen Enden der roten und der schwarzen Leitung löten Sie auf die Platine: die rote Leitung an den Plus-Anschluss, die schwarze Leitung an den Minus-Anschluss (siehe Bild). Der nicht beschriftete Anschluss links des Minus-Anschlusses wird nicht verwendet. Achten Sie beim Löten immer darauf, dass die angelöteten Leitungen wirklich nur die vorgesehenen Kontakte berühren, sonst besteht die Gefahr eines Kurzschlusses, der den Programmer oder die USB-Schnittstelle beschädigen kann. Hier der fertig zusammengebaute HVSP Fusebit Programmer:



# Funktionsweise

Bevor Sie den zu programmierenden Mikrocontroller in die Fassung stecken, stellen Sie sicher, dass der Programmer nicht angeschlossen, also stromlos ist.

Nach dem Einsetzen des Mikrocontrollers prüfen Sie bitte, dass er richtig herum in der Fassung sitzt: Pin 1, meist gekennzeichnet mit einem kleinen Kreis befindet sich dann an der Fassungsseite mit der Kerbe, das ist die Seite, die dem SMD-Mikrocontroller zugewandt ist, der auf die Programmer-Platine aufgelötet ist.

Je nach gewünschter Funktion müssen die kleinen DIP-Schalter eingestellt werden. In der Beschreibung verwenden wir die Bezeichnung 1 für "eingeschaltet" ("ON") und 0 für "ausgeschaltet" ("OFF"). Um Überraschungen zu vermeiden, empfiehlt es sich, nach jeder Verwendung die Schalter 2 und 3 zurück auf 0 zu schieben. In diesem Zustand werden die Fuse-Bytes des Mikrocontrollers nur gelesen und angezeigt, es wird nichts verändert. In den folgenden Abschnitten finden Sie die Bedeutung der Schalter im Einzelnen.

## Schalter 1: RESTART

Wenn Sie diesen Schalter von 0 auf ein oder von 1 auf 0 schieben, beginnt der Fusebit Programmer neu mit seine Arbeit. Das ist hilfreich, wenn Sie nach einem Programmiervorgang noch ein weiteres Bit verändern möchten oder ein anderes Fuse-Byte anzeigen lassen wollen. Natürlich können Sie den Programmer auch dadurch neu starten, dass Sie die Stromzufuhr kurz unterbrechen.

## Schalter 2 und 3: Funktion

Mit diesen beiden Schaltern bestimmen die sie Funktion des HVSP Fusebit Programemrs. Sie können ein noch auszuwählendes Fuse-Bit auf 1 oder 0 setzen, den Inhalt des Mikrocontrollers löschen oder einfach den Zustand eines der Fuse-Bits anzeigen lassen:

| Schalter 2 | Schalter 3 | Funktion          | Bedeutung   |
|------------|------------|-------------------|---|
| 0          | 0          | <b>DISPLAY</b>    | gewähltes Fuse-Bit und Fuse-Byte anzeigen lassen  |
| 0          | 1          | <b>CLEAR</b>      | gewähltes Fuse-Bit auf 0 ändern (Datenblätter nennen diesen Zustand meist "programmed")   |
| 1          | 0          | <b>SET</b>        | gewähltes Fuse-Bit auf 1 ändern (Datenblätter nennen diesen Zustand meist "unprogrammed") |
| 1          | 1          | <b>CHIP ERASE</b> | Mikrocontroller einschließlich möglicher Sperren löschen (Lock-Bits werden auf 1 gesetzt) |

Die Schalter 2 und 3 sollten Sie nach jedem Programmiervorgang wieder in die Stellung 0 zurückschieben. Damit vermeiden Sie ungewolltes Verändern von Fuse-Bits beim nächsten Start des Programmers.

### Schalter 4 und 5: Fuse-Byte

Damit wird das anzuzeigende oder zu ändernde Fuse-Byte festgelegt. Die Schalter 4 und 5 haben dabei folgende Bedeutungen:

| Schalter 4 | Schalter 5 | Byte               |
|------------|------------|--------------------|
| 0          | 0          | Fuse Low Byte      |
| 0          | 1          | Fuse High Byte     |
| 1          | 0          | Fuse Extended Byte |
| 1          | 1          | Lock Byte          |

Die genaue Bedeutung dieser Bytes finden Sie im Datenblatt Ihres Mikrocontrollers.

### Schalter 6 bis 8: Fuse-Bit

Mit diesem Schalter wird die Nummer des Bits festgelegt, das innerhalb des oben gewählten Fuse-Bytes angezeigt oder geändert werden soll. Die Schalter 6 bis 8 haben dabei folgende Bedeutungen:

| Schalter 6 | Schalter 7 | Schalter 8 | Bit Nummer                                 |
|------------|------------|------------|--|
| 0          | 0          | 0          | 0 (ganz rechts, auch als LSbit bezeichnet) |
| 0          | 0          | 1          | 1  |
| 0          | 1          | 0          | 2  |
| 0          | 1          | 1          | 3  |
| 1          | 0          | 0          | 4  |
| 1          | 0          | 1          | 5  |
| 1          | 1          | 0          | 6  |
| 1          | 1          | 1          | 7 (ganz links, auch als MSbit bezeichnet)  |

Die genaue Bedeutung der einzelnen Bits finden Sie im Datenblatt Ihres Mikrocontrollers.

### Einschalten

Wenn Sie sicher sind, dass sich alle DIP-Schalter in der gewünschten Stellung befinden, schließen Sie die Stromversorgung für den Programmer an – am einfachsten an die USB-Buchse eines PCs. Sie können auch ein USB-Steckernetzteil verwenden, wichtig ist aber in beiden Fällen, dass die Spannung ca. 5 Volt beträgt und 5,5 Volt auf keinen Fall überschreitet und die im nächsten Kapitel zusammengestellten Sicherheitshinweise beachtet werden. Sobald der Programmer läuft, können Sie die im Folgenden beschriebenen LED-Ausgaben beobachten.

## **Fehlersignal**

Falls eines der Fuse-Bits programmiert werden sollte und dieser Vorgang nicht zum gewünschten Ergebnis geführt hat, blinkt die LED für etwa 2 Sekunden in sehr schneller Folge.

Bitte beachten Sie, dass es nicht möglich ist, die Fuse-Bits eines gesperrten Mikrocontrollers zu verändern. Sollten sie es trotzdem versuchen, erscheint das Fehlersignal. Die Sperre kann nur durch ein "Chip Erase" beseitigt werden, womit dann aber auch der Inhalt des Programmspeichers gelöscht wird.

## **Fuse-Bit**

Unabhängig davon, ob ein Fehler aufgetreten ist oder nicht, der Zustand des selektierten Bits im gewählten Fuse-Byte wird per LED angezeigt: langes Leuchten für 1, sehr kurzes Leuchten für 0.

## **Fuse-Byte**

Anschließend zeigt der Programmer das gesamte selektierte Fuse-Byte an. Er beginnt dabei mit dem Bit Nummer 7, also dem ganz linken. Auch hier bedeutet ein langes LED-Leuchten 1 und ein sehr kurzes Leuchten 0.

## **Ausschalten und Neustart**

Nach dem Anzeigen von Fuse-Bit und Fuse-Byte schaltet sich der Programmer selbst ab. Neu starten können Sie ihn, indem Sie ihn kurz von der Stromversorgung trennen oder indem Sie den Schalter Nummer 1 umschalten (von 0 auf 1 oder von 1 auf 0).

## **Herausnehmen des Mikrocontrollers**

Bevor Sie den Mikrocontroller entnehmen, sollten Sie den Programmer von der Stromzufuhr trennen. Gehen Sie beim Herausziehen des Mikrocontrollers bitte vorsichtig vor! Die IC-Fassung ist oberflächenmontiert, das heißt, sie ist nicht mit der Platine verschraubt und kann bei sehr hoher mechanischer Beanspruchung von ihr wegreißen. Am besten, Sie nehmen einen Schraubendreher, schieben ihn zwischen Mikrocontroller und Fassung und hebeln den eingesteckten Mikrocontroller vorsichtig heraus.

## Sicherheitshinweise

Die folgenden Hinweise helfen Ihnen, Probleme bei der Benutzung des HVSP Fusebit Programmers zu vermeiden.

- Sorgen Sie für eine Betriebsspannung von 5 Volt Gleichspannung, niemals über 5,5 Volt.
- Entfernen Sie Mikrocontroller grundsätzlich mit einem Schraubendreher aus der IC-Fassung.

## Adapter für 14-Pin-Mikrocontroller

Diese Pins der 8-poligen Fassung müssen mit den jeweils folgenden Pins der 14-poligen Fassung verbunden werden:

|       |     |        |         |
|-------|-----|--------|---------|
| Pin 1 | mit | Pin 4  | (RESET) |
| Pin 2 | mit | Pin 2  | (SCI)   |
| Pin 3 | mit | Pin 11 | (PE2)   |
| Pin 4 | mit | Pin 12 | (PE1)   |
| Pin 4 | mit | Pin 13 | (PE0)   |
| Pin 4 | mit | Pin 14 | (GND)   |
| Pin 5 | mit | Pin 7  | (SDI)   |
| Pin 6 | mit | Pin 8  | (SII)   |
| Pin 7 | mit | Pin 9  | (SDO)   |
| Pin 8 | mit | Pin 1  | (VCC)   |

## Weitere Informationen

Zum verändern von Fuse- oder Lock-Bytes ist es natürlich wichtig zu wissen, welche Funktion welches der enthaltenen Bits besitzt. Sämtliche Details finden Sie in den jeweiligen Datenblättern:

<http://www.atmel.com/products/microcontrollers/avr/tinyAVR.aspx>

Bei Fragen finden Sie oft Hilfe in einschlägigen Internet-Communities. Besonders empfohlen seien:

<http://www.mikrocontroller.net/articles/AVR-Tutorial>

[http://www.rn-wissen.de/index.php/AVR\\_Assembler\\_Einf%C3%BChrung](http://www.rn-wissen.de/index.php/AVR_Assembler_Einf%C3%BChrung)

<http://www.mikrocontroller.net/forum/mikrocontroller-elektronik>

<http://www.roboternetz.de/community/forums/20-AVR-Hardwarethemen>

<http://www.roboternetz.de/community/forums/56-Assembler-Programmierung>