Wie kommt das Weiche in das Harte

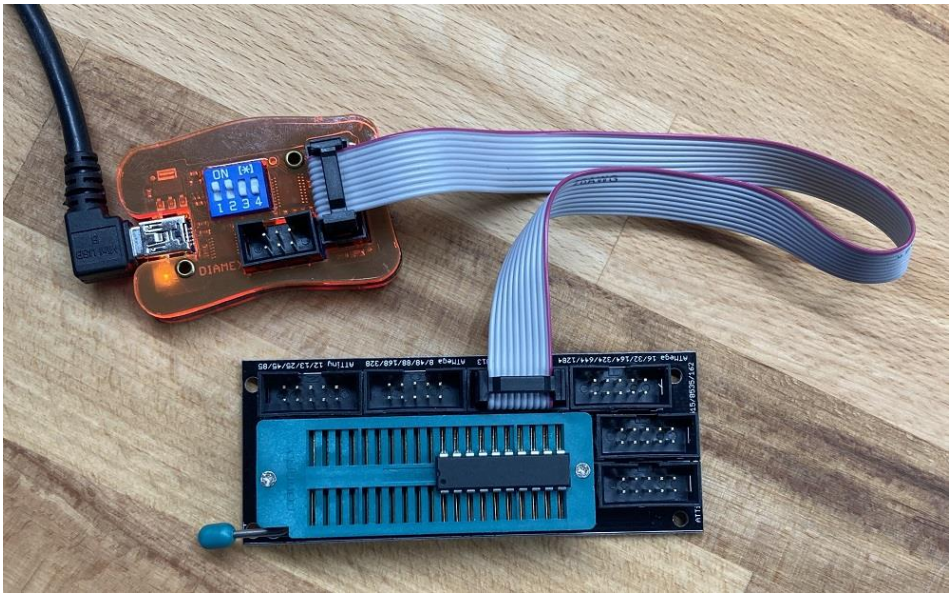
Ich habe in der Vergangenheit ein paar Projekte auf Basis von Microchip AVR (8Bit-) Controller (MCU) vorgestellt. Immer wieder wurde die Frage an mich herangetragen:

„Wie bekomme ich die Software auf den Controller?“

Es gibt viele Wege die Software auf die MCU zu bringen. Es gibt etliche Software-Tools und unzählige Programmer (Hardware, die die Verbindungung zw PC und MCU herstellt) und Progrmmieradapter. Nicht alle Software-Tools können alle Programmer bedienen und umgekehrt. Jedes Tool, jeder Programmer hat seine spezifischen Vor- und Nachteile.
Hier will ich eine Kombi vorstellen und beschreiben.

Die Hardware

Als Programmer empfehle ich einen „STK500 Programmer“.



Ein gutes Preis-Leistungs-Verhältnis bietet die Firma Diamex

Der DIAMEX PROG-S2 kann AVR MCUs und auch andere MCUs flashen.
https://www.diamex.de/dxshop/USB-ISP-Programmer-fuer-AVR-STM32-LPC-Cortex-Prog-S2

Zusammen mit dem „AVR-Schwenkhebler“ hat man die Hardware schon komplett um AVRs im DIL-Gehäuse zu flashen.
https://www.diamex.de/dxshop/Schwenkhebel-Modul-fuer-fast-alle-AVR-Controller-im-DIL-Gehaeuse

Die Software

Auch bei den AVR-Flash-Tools gibt es eine vielfältige Auswahl mit ihren jeweiligen Vor- und Nachteilen.
Unter Windows 10 die mMn einfachste Möglichkeit (die ich auch selbst nutze) ist das in Bascom integrierte Flash-Tool.
Bascom von MCS Electronics ist ein sehr leistungsfähiger Basic Compiler für 8-Bit-AVRs.
Für den Privatgebrauch ist die Version „BASCOM-AVR Demo“ kostenlos.
https://www.mcselec.com/index.php?option=com_docman&task=cat_view&gid=99&Itemid=54

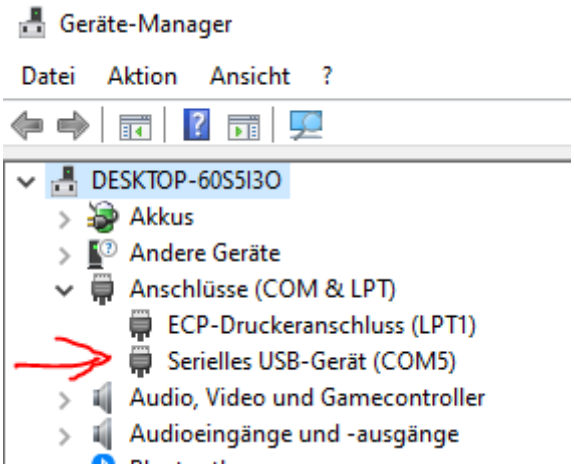
Der Bascom Flasher bietet eine Grafische Oberfläche und bringt den Treiber für STK500 Programmer gleich mit.

Installation von Bascom unter Windows 10

Einfach Bascom herunterladen und der Setupprozedur folgen. Das wars schon.

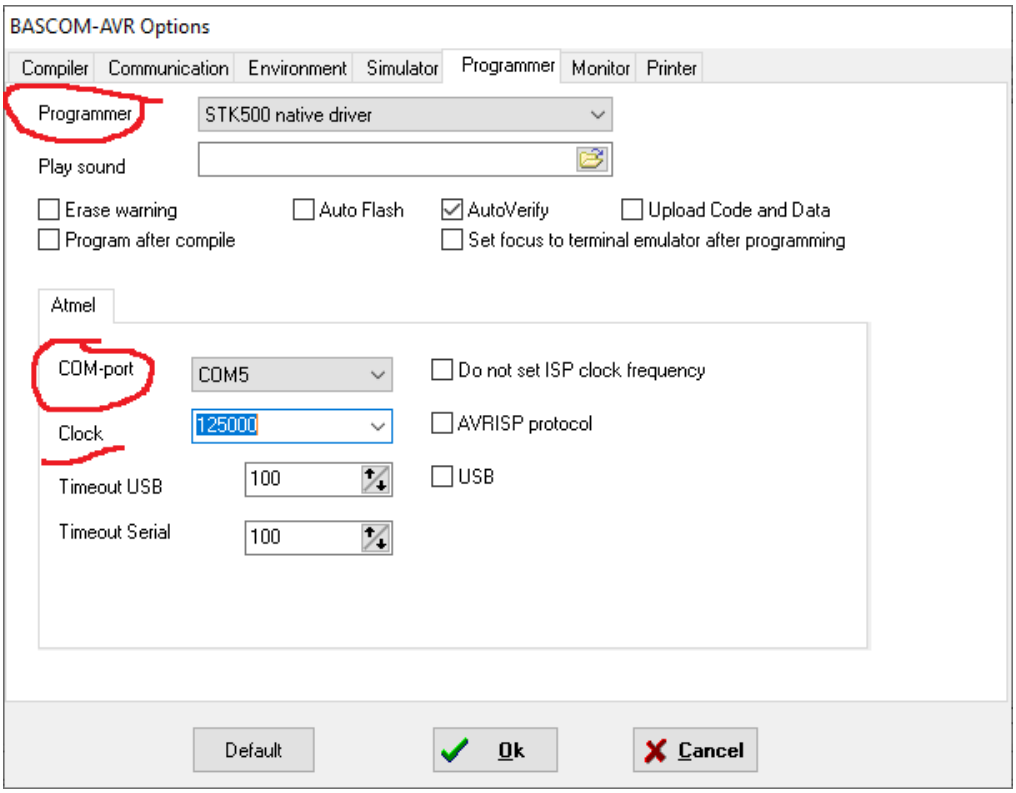
STK500 Programmer einrichten

Programmer an den USB-Port des PC stecken. Der Programmer wird dann im Gerätemanager von Windows als COM-Port angezeigt (hier COM5).



Bascom

In Bascom die Einstellungen für den Programmer vornehmen.
Unter Options – Programmer den Programmer und COM-Port (siehe Gerätemanager) auswählen.

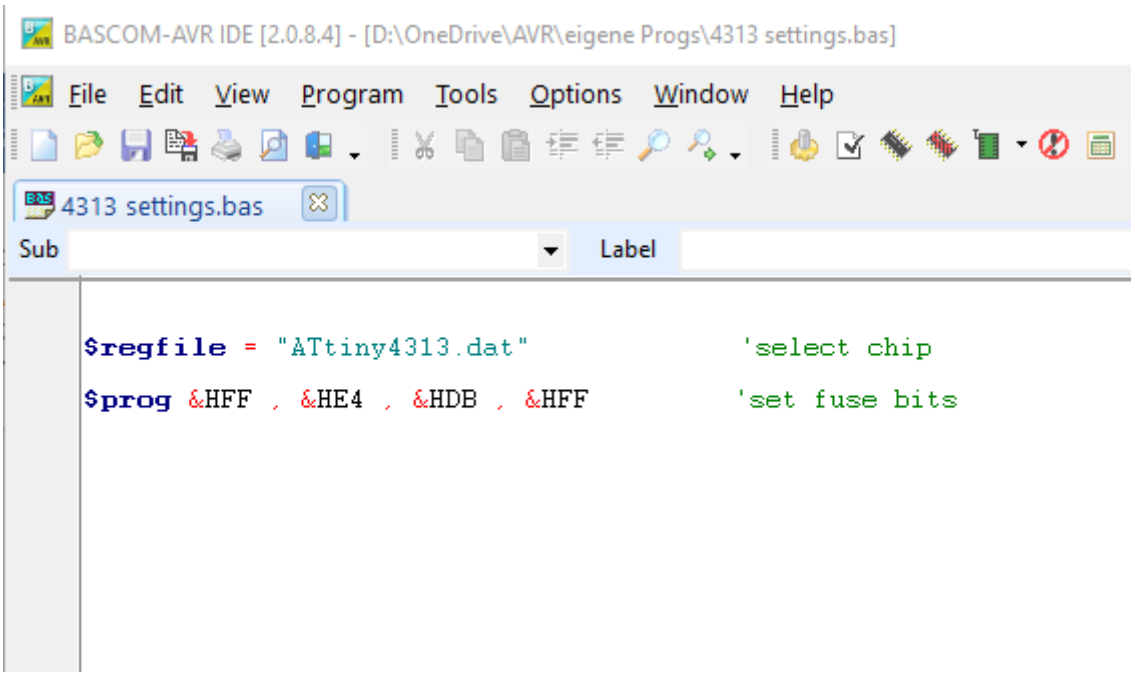


Einstellungen für den AVR und Fusebits

Bevor man einen AVR flashen kann, muss man Bascom mitteilen welchen AVT Typ man bespaßen möchte. Bei der Gelegenheit übergeben wir auch die Einstellungen für die „Fusebits“. Die Fusebits steuern uA die Taktfrequenz des AVR.
Dazu mit File – New ein neues Programm erstellen.
Beispiel für einen ATTiny4313 und 8MHz interner Takt

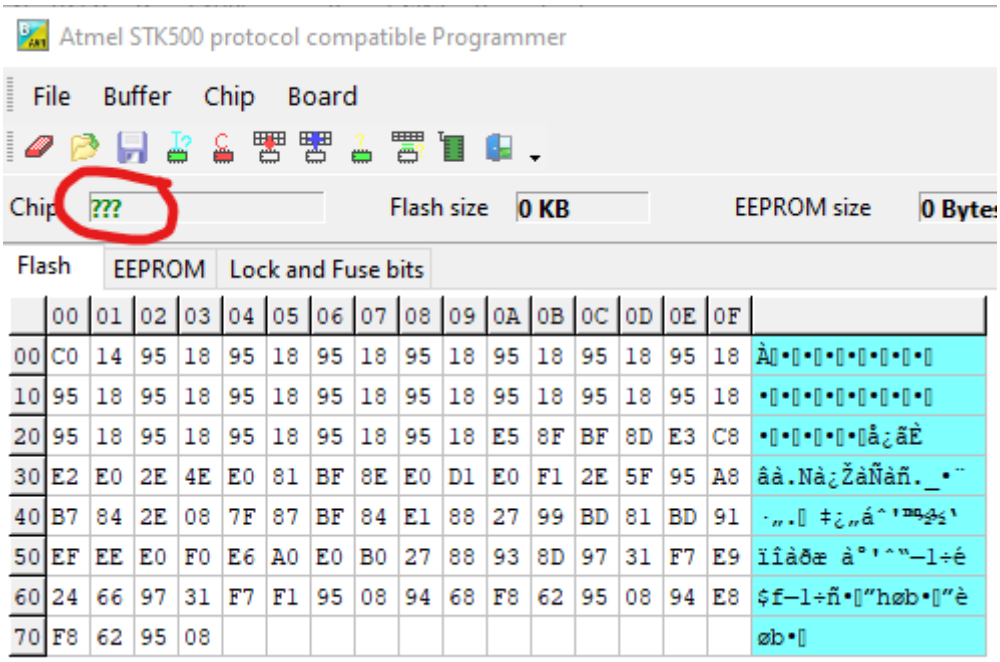
```
‘settings für SBUS-Switch
$regfile = "ATtiny4313.dat"          'select chip
$prog &HFF , &HE4 , &HDB , &HFF      'set fuse bits
```

Sieht in Bascom dann so aus:



Vor dem Kompilieren muss man dann mit File – Save As speichern und kann das Programm beim nächsten Mal einfach wieder laden.
Das Programm dann mit File – Compile kompilieren.
Jetzt die Fusebits mit Program – Send to Chip auf den AVR übertragen.

Damit öffnet sich das Fenster für das Flash-Tool.



	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00	C0	14	95	18	95	18	95	18	95	18	95	18	95	18	95	18	À••••••••••
10	95	18	95	18	95	18	95	18	95	18	95	18	95	18	95	18	••••••••••••
20	95	18	95	18	95	18	95	18	95	18	E5	8F	BF	8D	E3	C8	••••••••å¿ãÈ
30	E2	E0	2E	4E	E0	81	BF	8E	E0	D1	E0	F1	2E	5F	95	A8	âà.Nà¿ŽàÑàñ._•¨
40	B7	84	2E	08	7F	87	BF	84	E1	88	27	99	BD	81	BD	91	·„.• ‡¿„á^'™½¼'
50	EF	EE	E0	F0	E6	A0	E0	B0	27	88	93	8D	97	31	F7	E9	ïîàðæ à°'^"—1÷é
60	24	66	97	31	F7	F1	95	08	94	68	F8	62	95	08	94	E8	$f—1÷ñ•"høb•"è
70	F8	62	95	08													øb••

Mit Chip – Identify prüfen wir jetzt die Verbindung PC – Programmer – AVR. Im oben markierten Bereich muss dann der korrekte ACR Typ stehen.

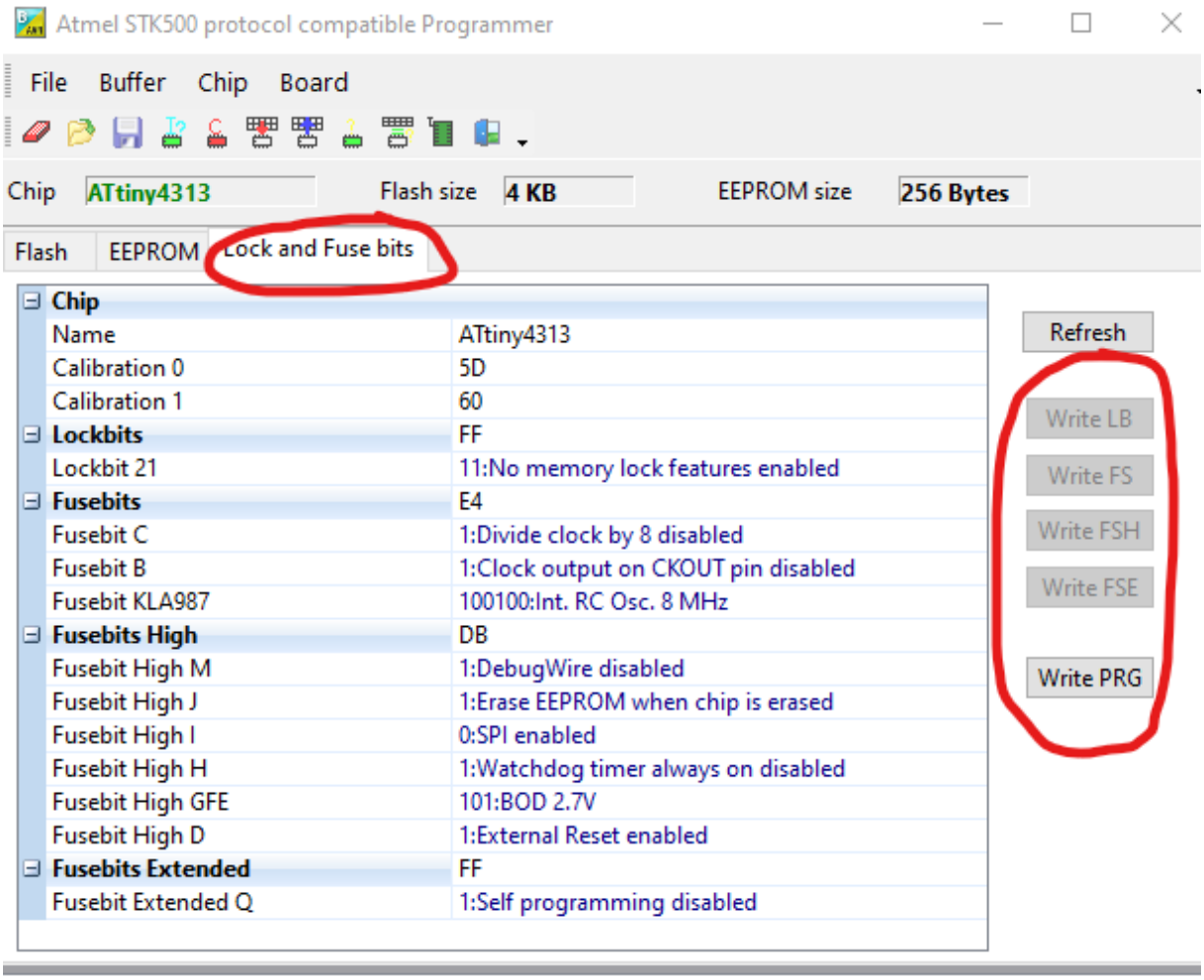
Jetzt mit Chip – Autoprogramm die Fusebits auf dem AVR setzen. Das Fenster schliesst sich am Ende automatisch.

Was haben wir bis jetzt?

- Das Flash-Tool weiss welcher Programmer verwendet wird und wie er erreichbar ist.
- Die Verbindung zum AVR funktioniert.
- Die Fusebits wurden auf dem AVR gesetzt.

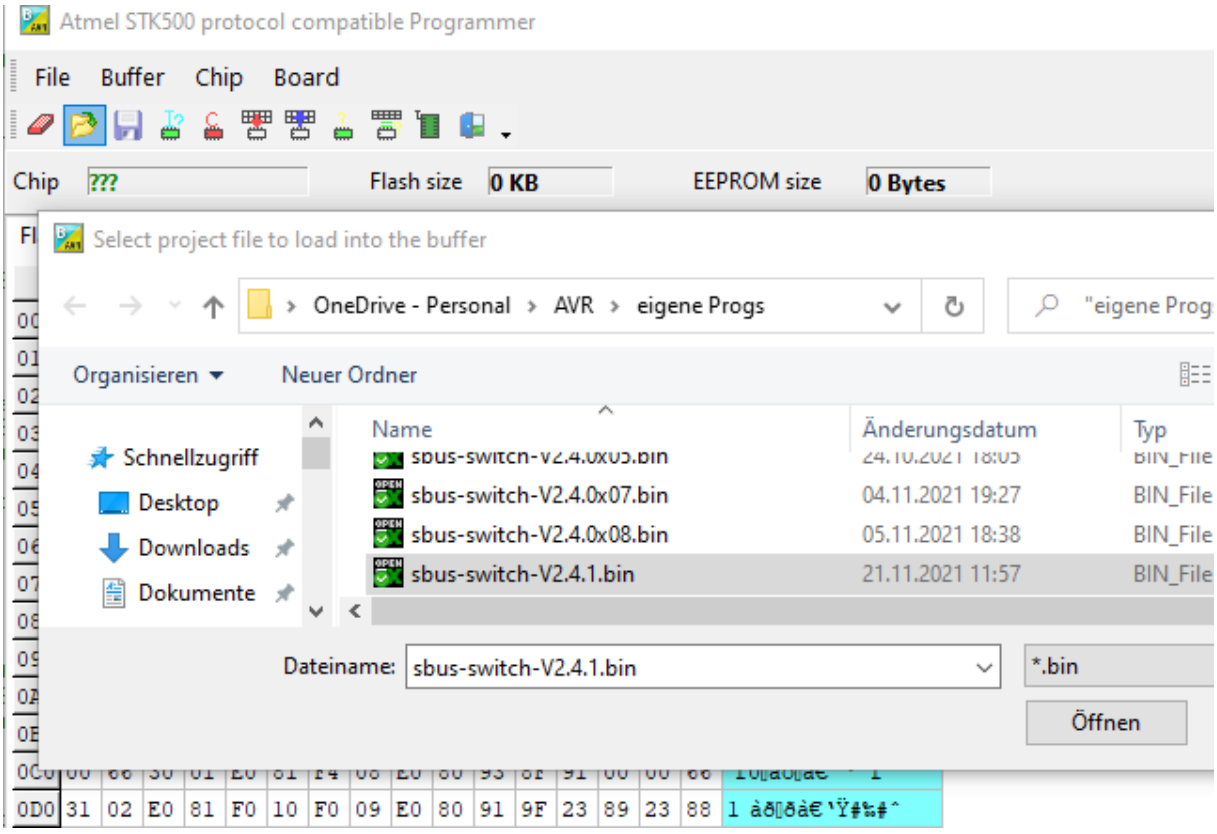
Optional

Man kann die Fusebits auch manuell einstellen. Die Erste Zeile ($regfile = ….) ist dann trotzdem nötig. Hier sollte man wissen was man tut, da man sich schnell den eigenen Ast absägt und dann keine Verbindung mehr zum AVR hat.
Dazu im Flash-Tool auf Lock and Fuse bits gehen und nach Änderung auf der rechten Seite die entsprechenden Write-Buttons klicken



Binary flashen

Jetzt noch das Binary auf den AVR flashen.
Dazu wieder das Flash-Tool mit Program – Send to Chip öffnen.
Im Flash-Tool mit Buffer – Load from File das Binary auswählen



und mit Chip – Autoprogramm auf den AVR flashen.
Das war's!!!
Jetzt haben wir einen fertigen AVR, der z.B. nur noch auf den Socket des SBUS-Switch gesteckt werden muss.
Wenn man das ein paar Mal gemacht hat, dann ist das eine Sache von einer Minute.

Wichtig zu wissen:
Ein Binary ist immer für einen bestimmten AVR-Typ erstellt. Man kann ein Binary, welches für einen ATTiny4313 erstellt wurde, nicht auf einen ATMega8 flashen. Auch nicht wenn es sich um sehr ähnliche Typen handelt. Z.B. ein Binary für einen ATTiny2313 auf einen 4313 flashen.