

RS41 Ham Use Project

Umprogrammierung der Vaisala RS41 Wettersonden auf 70cm Amateurfunkband (APRS und RTTY)

Quellcode:

<https://github.com/df8oe/RS41HUP>

Hardware:

Eine Vaisala RS41 Wettersonde ;-) und einen STLINK-V2 Programmer:

Entweder das Original von ST oder ein vereinfachter USB-Dongle z.B. "WINGONEER ST-Link V2"

Benutzte Programme für Windows:

Entwicklungsumgebung:

<https://www.wyzbee.com/download/Utilities/Software/ColIDE-1.7.8.exe>

Compiler:

https://launchpad.net/gcc-arm-embedded/5.0/5-2016-q3-update/+download/gcc-arm-none-eabi-5_4-2016q3-20160926-win32.exe

Programmiersoftware:

<http://www.st.com/en/development-tools/stm32cubeprog.html> (link ganz unten)

Schritt 1:

Kabel basteln und Software installieren:

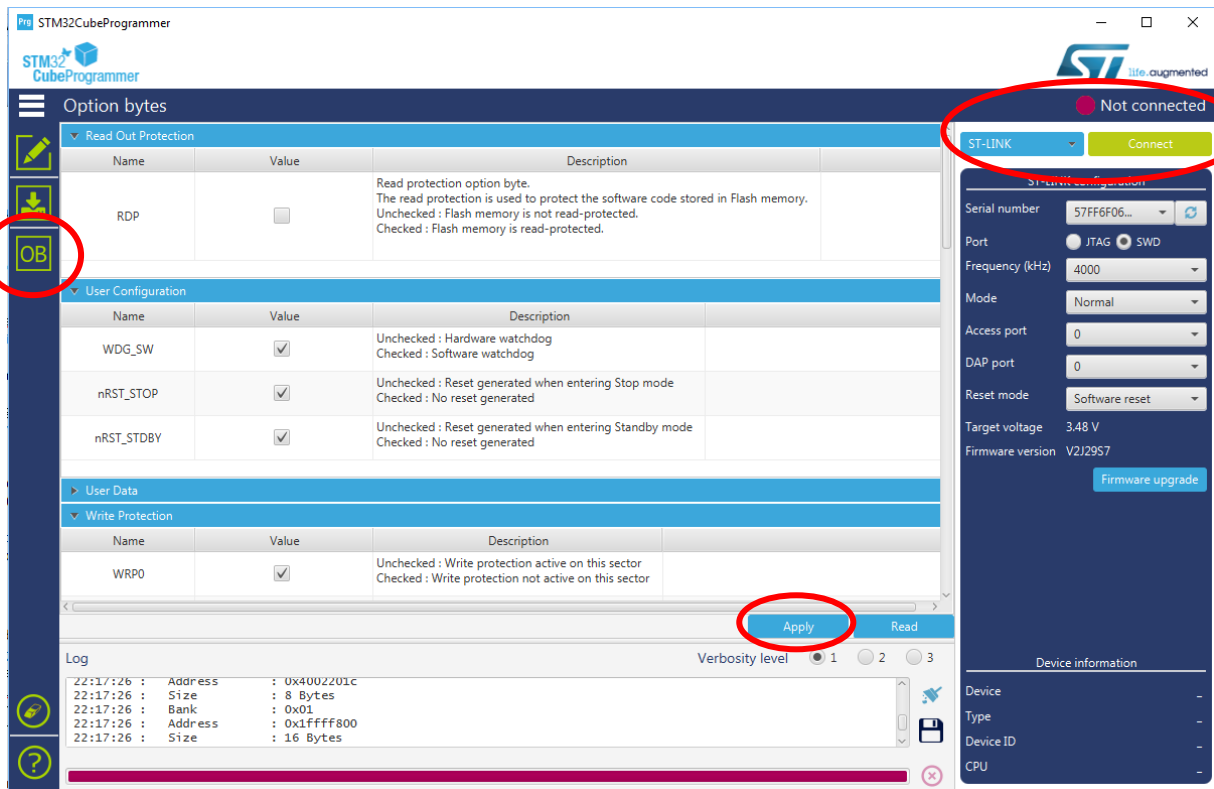
- Zum Programmieren der Sonde müssen 3 Pins der Sonde mit dem Programmiergerät verbunden werden: Masse, SWClock und SWData.
- Pinbelegung der Sonde ist im Quellcode unter Docs zu finden. Pinbelegung des Programmiergerätes ist unterschiedlich und im zugehörigen Handbuch zu finden.
- Man kann auch noch den 3.3V Pin verbinden, dann versorgt das Programmiergerät die Sonde während des Programmierens. und geht sofort an, wenn die Sonde angeschlossen wird.
- Software installieren, Programmiergerät an den Computer anschließen, mit der Sonde verbinden und die Sonde einschalten, falls man den 3.3V Pin nicht benutzt.

Schritt 2:

Erstmal wird Vaisala ausgetrickst. Mit der Programmiersoftware "CubeProg" werden die passenden Option Flags richtig eingestellt:

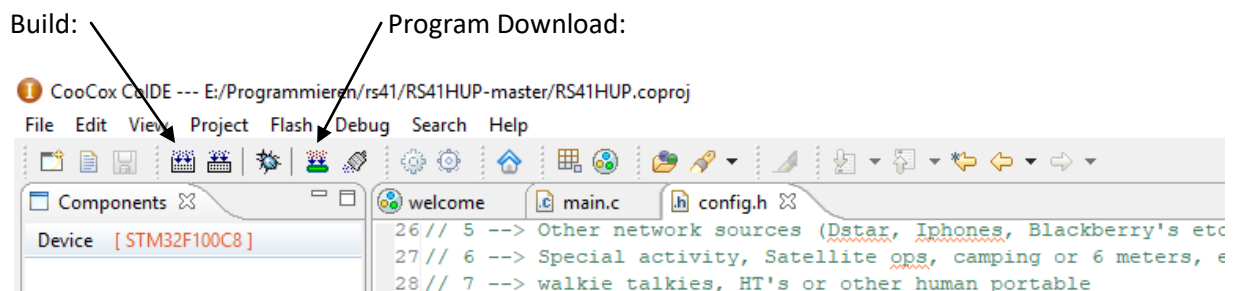
- Oben rechts:
 - Blaues Feld: ST Link auswählen
 - Grünes Feld: Connect auswählen (der rote Punkt oben drüber wechselt auf grün)
- Links "OB" für Option Bytes auswählen:
 - **Read Out Protection:** Haken raus und unten Mitte blaues Feld: *Apply* klicken. Das bewirkt dass man überhaupt was ändern kann. Alle anderen Einstellungen kann man dann gemeinsam anwenden.
 - **User Configuration:** Alle Haken rein, damit nicht ständig resettet wird
 - **Write Protection:** Alle Haken rein, damit der gesamte Speicher beschrieben werden kann
 - Änderungen speichern wieder mit "Apply"
- Nach jedem "Apply" sollten eine oder mehrere Meldungen aufploppen dass die Bytes erfolgreich gesetzt wurden.

Ab jetzt ist die CPU frei für alle weiteren Schritte.



Schritt 3:

- In der Entwicklungsumgebung "CoIDE" das Projekt "RS41HUP.coproj" aus dem Quellcodeordner öffnen .
- in der Datei "config.h" gewünschte Einstellungen wie Frequenz, Sendestärke, Rufzeichen usw. einstellen.
- Compilieren mit F7 oder dem entsprechenden Knopf in der Menüleiste (Build). Beim ersten Mal wird nach dem Speicherort des Compilers gefragt. Bei mir z.B. C:\Programme\GNU Tools ARM Embedded\5.4 2016q3\bin
- fertiges Programm auf die Sonde übertragen: In der Menüleiste ->Flash->Program Download klicken.



Wenn in den Statusmeldungen unten steht:

Erase: Done
Program: Done
Verify: Done

hat alles geklappt. Die grüne LED leuchtet und die Rote LED blinkt bei jeder Übertragung.

Viel Spass damit!