
Protokoll

LEDs ansteuern

Systemtechnik Theorie
4CHIT 2016/17

Martin Wölfer

Note:
Betreuer: WEIJ

Version 0.2
Begonnen am 30. September
Beendet am 6. Oktober 2016

Inhaltsverzeichnis

1	Einführung	1
1.1	Ziele	1
1.2	Voraussetzungen	1
1.3	Aufgabenstellung	1
2	Vorbereitungen	2
2.1	AC6 Workbench installieren	2
2.1.1	Installer-File herunterladen	2
2.1.2	Installation	2
2.2	GCC-ARM-NONE-EABI installieren	3
3	Ergebnisse	3
3.1	Projekt erstellen	3
3.2	LED ansteuern	6
3.2.1	Programm schreiben	6
3.2.2	Programm kompilieren	6
3.2.3	Programm ausführen	6

1 Einführung

1.1 Ziele

- AC6 System Workbench installieren
- Mit der workbench das board ansteuern"
- Rote LED zum leuchten bringen

1.2 Voraussetzungen

- Installer Datei Workbench
- STM32F3DISCOVERY Board
- Extra Compiler: gcc-arm-none-eabi
- Grundwissen C

1.3 Aufgabenstellung

Es soll uns gelingen erste LEDs zum Leuchten zu bringen.

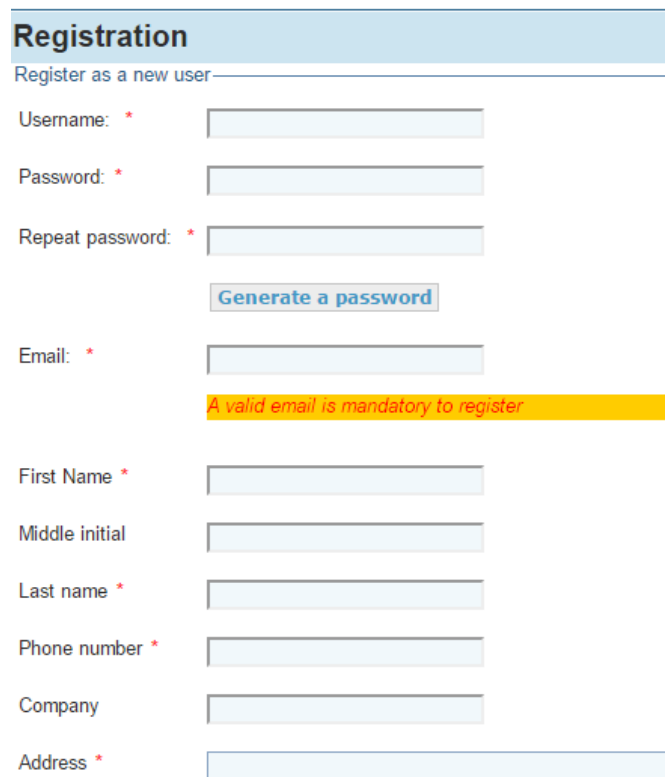
Schreibe ein entsprechendes Programm und dokumentiere dein Ergebnis mit ein paar Screenshots und einem Photo vom Board. Füge Foto und Screenshots zu einem Dokument zusammen. Gib dieses zusammen mit dem Programm hier ab.

2 Vorbereitungen

2.1 AC6 Workbench installieren

2.1.1 Installer-File herunterladen

Um das Installer File zu bekommen, muss man sich auf der Seite <http://www.openstm32.org/> registrieren:



The image shows a registration form titled "Registration" with the subtitle "Register as a new user". The form contains several input fields and a button:

- Username:** * (required) [input field]
- Password:** * (required) [input field]
- Repeat password:** * (required) [input field]
- Generate a password** [button]
- Email:** * (required) [input field]
- First Name:** * (required) [input field]
- Middle initial** [input field]
- Last name:** * (required) [input field]
- Phone number:** * (required) [input field]
- Company** [input field]
- Address:** * (required) [input field]

A yellow error message is displayed below the email field: "A valid email is mandatory to register".

Abbildung 1: Registration auf der OpenSTM32 Seite

2.1.2 Installation

Bei der Installation muss man lediglich den Anweisungen folgen, und erhält anschließend einen Ordner Ac6, welcher die Workbench, eine Eclipse Anwendung, beinhaltet.



Abbildung 2: Die STM32 Workbench auf Eclipse basierend

2.2 GCC-ARM-NONE-EABI installieren

Man braucht diesen extra Compiler um das Programm auf dem Board auszuführen. Herunterladen kann man diesen von: <https://launchpad.net/gcc-arm-embedded/+download>

3 Ergebnisse

3.1 Projekt erstellen

Um ein Projekt zu erstellen muss man rechtsklick drücken und ein neues C Projekt erstellen:

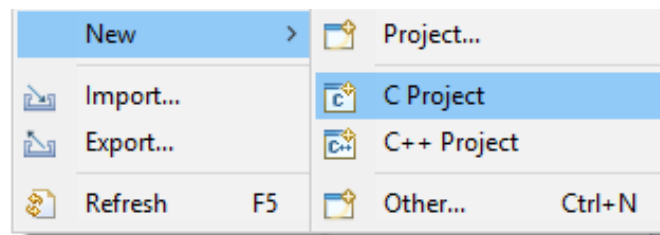


Abbildung 3: Neues C Projekt erstellen

Als nächstes muss man den Namen eingeben und bei Project type/Toolchains Ac6 STM32 MCU auswählen

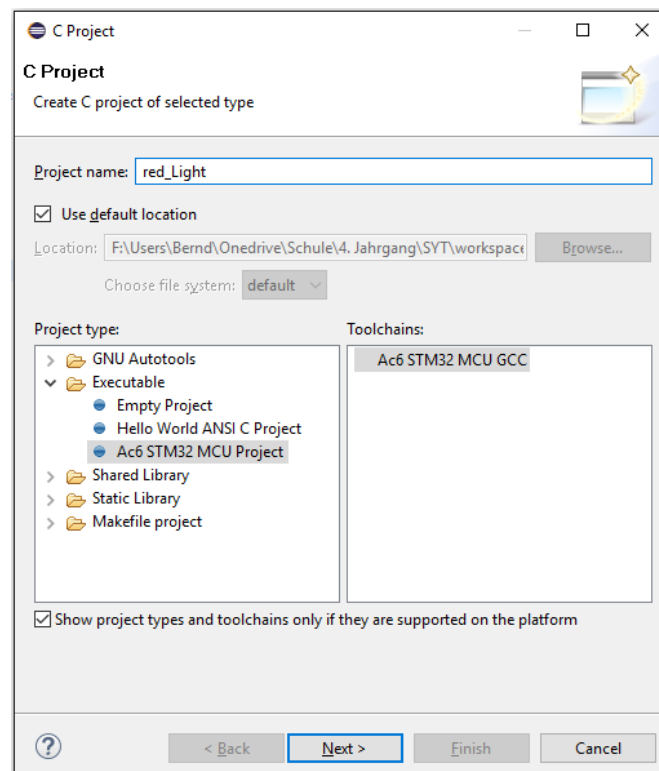


Abbildung 4: Richtige Optionen auswählen

Bei der nächsten Seite muss man überall Häkchen setzen

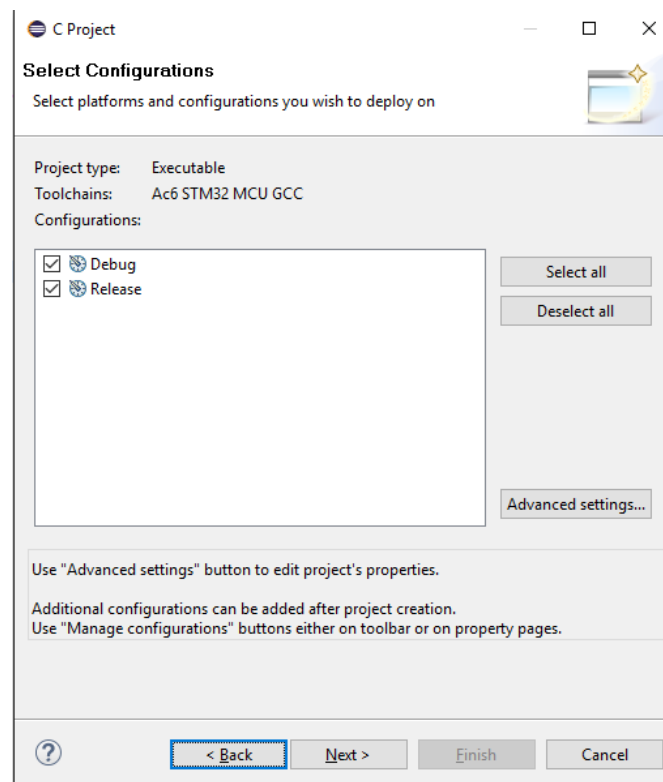


Abbildung 5: Häkchen setzen

Der nächste Schritt ist einfach unser Board auszuwählen: STM32F3DISCOVERY

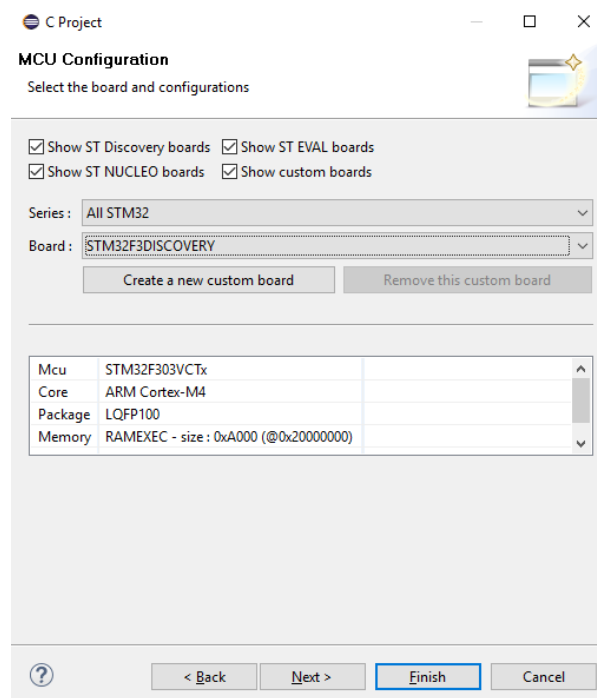


Abbildung 6: Richtiges Board auswählen

Der Prozess geht weiter indem man die Hardware Abstraction Layer (Cube HAL) Firmware runterlädt. Man muss allerdings keine weiteren Driver mehr hinzufügen.

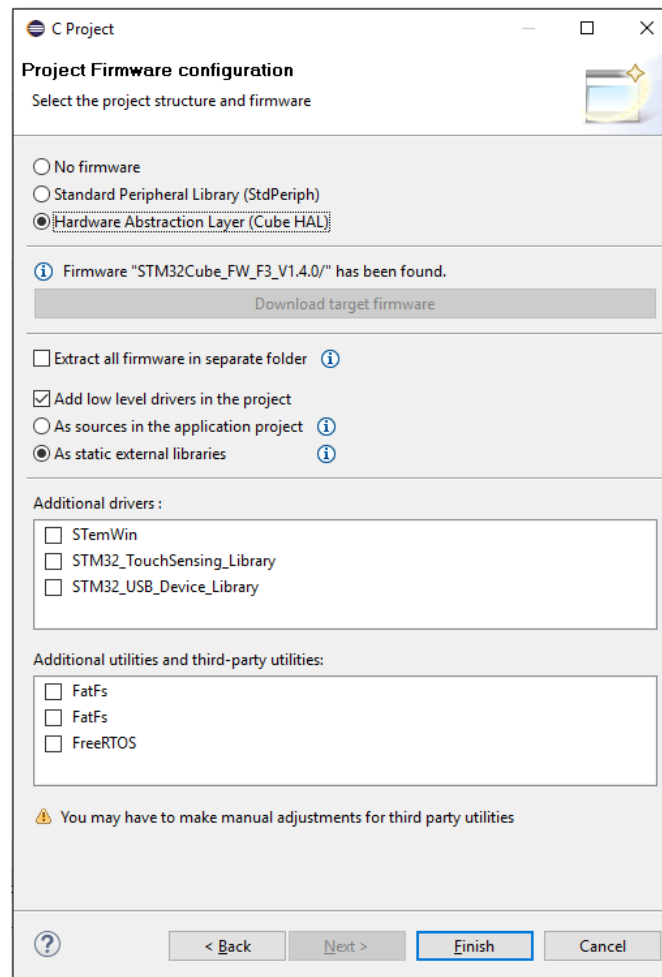


Abbildung 7: Firmware installieren

Wenn man nun auf Finish drückt sollte man so eine ähnliche Ordner Struktur erhalten

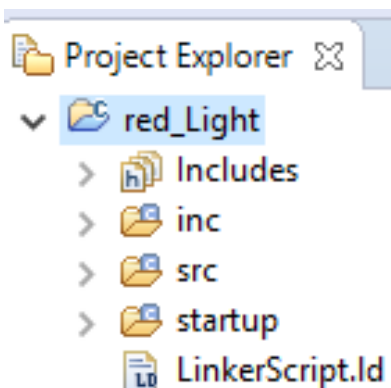


Abbildung 8: Ordnerstruktur

3.2 LED ansteuern

3.2.1 Programm schreiben

Man muss das File `main.c` im order `src` öffnen.

Man ersetzt dort den Sample Code:

```
1 int main(void) {
2     for(;;);
3 }
```

Mit folgendem Code:

```
1 int main(void) {
2     HAL_Init();
3     BSP_LED_Init(LED_RED);
4     BSP_LED_On(LED_RED);
5 }
```

Das einzige was dieser Code macht ist die Lampe Rot leuchten lassen.

3.2.2 Programm kompilieren

Um das Programm zu kompilieren muss man auf auf den Hammer drücken



Abbildung 9: Programm kompilieren

Und erhält dann folgenden output:

```
1 arm-none-eabi-objcopy -O binary "red_Light.elf" "red_Light.bin"
2 arm-none-eabi-size "red_Light.elf"
3 text      data      bss      dec      hex filename
4 1196      1076      1568      3840      f00 red_Light.elf
5 ' '
6
7 16:41:06 Build Finished (took 3s.557ms)
```

3.2.3 Programm ausführen

Um es auszuführen muss man zuerst eine kurze Änderung durchführen.

Man muss zuerst auf `Run Configurations...` drücken

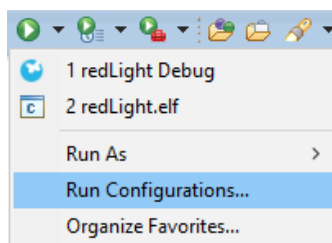


Abbildung 10: Run Configuration

Bei dem Fenster dass sich öffnet muss man nun auf Search Project... drücken und sein Projekt auswählen

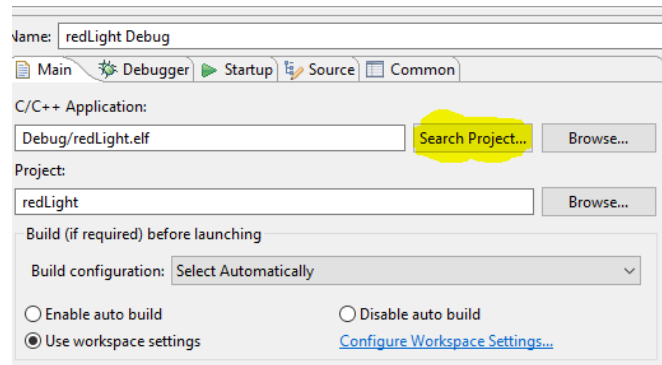


Abbildung 11: Search project....

Nun muss man nur mehr auf Apply drücken und das Programm schließlich ausführen.

Ob es passt sowohl in der Konsole betrachten:

```

1 xPSR: 0x61000000 pc: 0x2000002e msp: 0x2000a000
  stm32f3x.cpu: target state: halted
3 target halted due to breakpoint, current mode: Thread
  xPSR: 0x61000000 pc: 0x2000002e msp: 0x2000a000
5 verified 4476 bytes in 0.093752s (46.624 KiB/s)
  ** Verified OK **
7 ** Resetting Target **
  adapter speed: 1000 kHz

```

Als auch am Board:

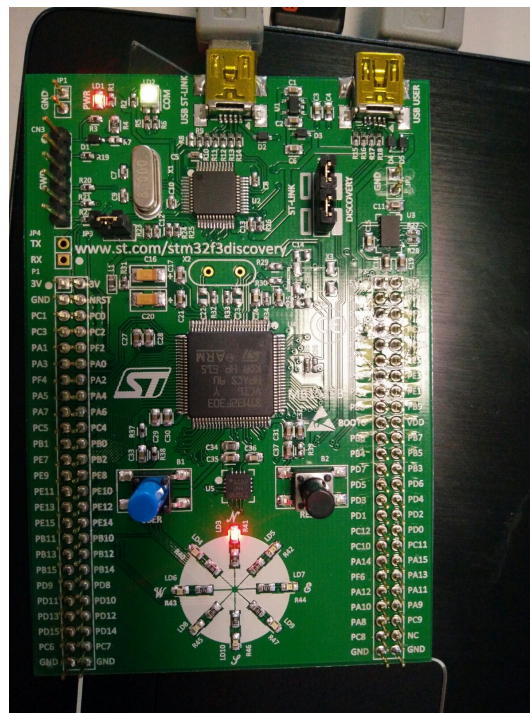


Abbildung 12: Board LED leuchtet rot

Abbildungsverzeichnis

1	Registration auf der OpenSTM32 Seite	2
2	Die STM32 Workbench auf Eclipse basierend	2
3	Neues C Projekt erstellen	3
4	Richtige Optionen auswählen	3
5	Häkchen setzen	4
6	Richtiges Board auswählen	4
7	Firmware installieren	5
8	Ordnerstruktur	5
9	Programm kompilieren	6
10	Run Configuration	6
11	Search project....	7
12	Board LED leuchtet rot	7