



Zielstellung

Bei den ersten Schritten in die Programmierung des mySTM32 Board light mit SiSy soll Ihnen dieser Schnelleinstieg helfen. Damit können Sie ein erstes Programm für ihren neuen STM32 Mikrocontroller erstellen und testen.

Das Ziel ist es, eine LED auf der Platine des mySTM32 Board light in kurzen Zeitabständen aufleuchten zu lassen und damit ein „Blinklicht“ zu erzeugen.

Voraussetzungen

Für die Bearbeitung der Aufgaben benötigen Sie folgende Software und Hardware:

Software

- SiSy ab Version 3.75 (DEMO unter www.sisy.de Download)

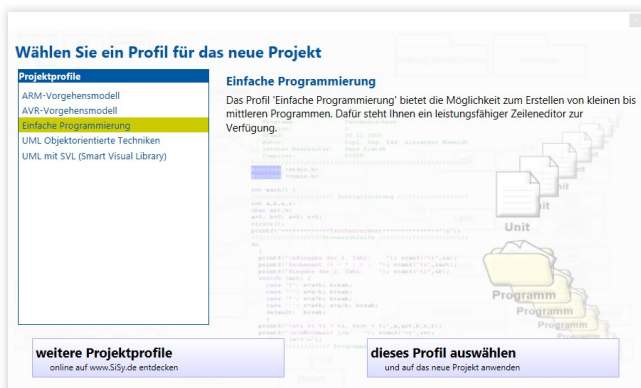
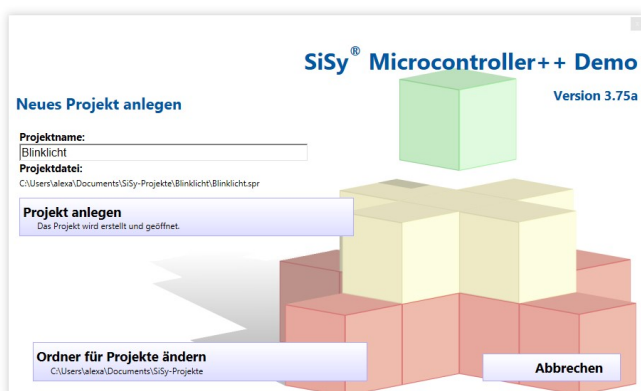
Hardware

- 1 mySTM32Board light
- 1 Mini USB-Kabel
- 1 kleines Patchkabel

Sie finden ausführliche Beschreibungen, Hinweise, Hilfen, Beispiele und Vorlagen im online Tutorial unter www.mySTM32.de.

1. Ein neues Projekt anlegen

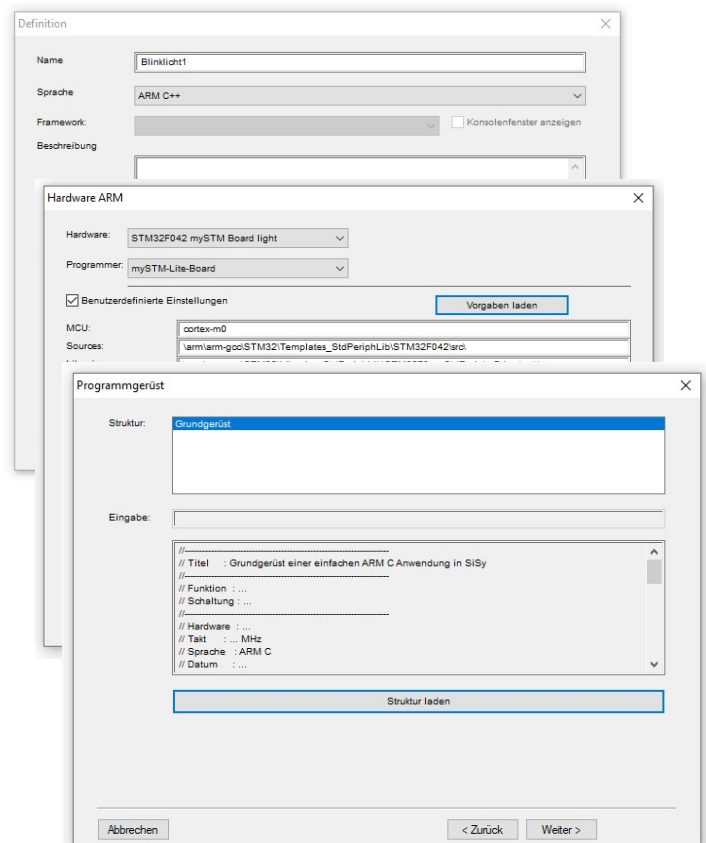
Starten Sie SiSy und wählen „Neues Projekt erstellen“, vergeben Sie den Projektnamen „Blinklicht“ und bestätigen Sie mit „Projekt anlegen“. Wählen Sie das Profil „Einfache Programmierung“ aus.



Es öffnet SiSy LibStore und stellt Vorlagen für die weitere Arbeit zur Verfügung. Wir arbeiten an dieser Stelle ohne Vorlagen und kehren zu SiSy zurück indem wir den LibStore-Dialog schließen.

2. Kleines C-Programm anlegen

Erstellen Sie ein Programm für den STM Mikrocontroller, indem Sie per Drag & Drop aus der Objektbibliothek ein Objekt „kleines Programm“ in das Diagrammfenster ziehen. In dem aufgeblendeten Dialogfenster vergeben Sie den Namen „Blinklicht1“. Der Datei- und Programmname wird dabei automatisch vergeben. Wählen Sie die Sprache ARM C++. Im nächsten Dialogfenster wählen Sie die verwendete Hardware aus, in unserem Fall „STM32F042 mySTM Board light“ (ohne HAL) und als Programmierer „mySTM-Lite-Board“. Im anschließenden Diagrammfenster markieren Sie „Grundgerüst“ und betätigen die Schaltfläche „Struktur laden“. Damit sind alle Einstellungen erledigt und ein Programm-Grundgerüst steht zur Verfügung. Zur Kontrolle können Sie alle Einstellungen jederzeit aufrufen, indem Sie das Objekt „kleines Programm“ markieren und mittels rechter Maustaste „Definieren“ auswählen.



Das geladene Programmgerüst steht Ihnen nun uneingeschränkt für die Weiterverarbeitung zur Verfügung und Sie können mit der Programmierung beginnen. Verbinden Sie das mySTM32 Board light über ein Mini-USB Kabel mit Ihrem PC und benutzen Sie ein kleines Patchkabel und Verbinden Pin B0 mit der roten LED

3. Quellcode erstellen

Das Ausgabegerät (LED) soll vom Port B gesteuert werden. Die Realisierung erfolgt über Pin B0. Ergänzen Sie die Programmvorlage mit nachfolgend aufgeführtem Quellcode.

```

//-----
// Titel      : Beispiel Blinky
//-----
// Funktion   : lässt die rote LED blinken
// Schaltung  : LED an GPIO Port B Pin B0
//-----
// Hardware  : STM32F042 Board light
// Takt      : 48 MHz
// Sprache   : ARM C
// Datum     : heute
// Version   : 1
// Autor     : ich
//-----
#include <stdint.h>
#include <stdlib.h>
#include "hardware.h"

void initApplication()
{
    SysTick_Config(SystemCoreClock/100);
    // weitere Initialisierungen durchführen

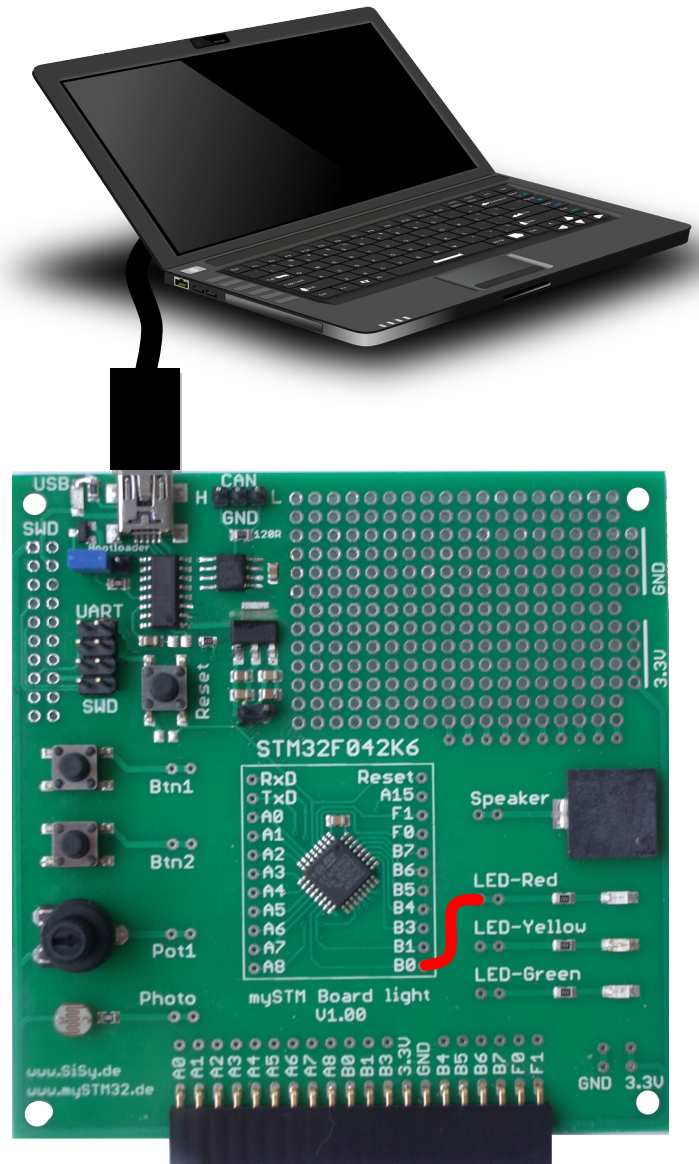
    /* GPIOD Takt einschalten */
    RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOD, ENABLE);

    /* Konfiguriere GPIO Port B0 */
    RCC_AHBPeriphClockCmd(RCC_AHBPeriph_GPIOB, ENABLE);
    GPIO_InitTypeDef led;
    led.GPIO_Mode = GPIO_Mode_OUT;
    led.GPIO_OType = GPIO_OType_PP;
    led.GPIO_Pin = GPIO_Pin_0;
    led.GPIO_PuPd = GPIO_PuPd_NOPULL;
    led.GPIO_Speed = GPIO_Speed_2MHz;
    GPIO_Init(GPIOB, &led);
}

int main(void)
{
    SystemInit();
    initApplication();
    do {
        GPIO_ToggleBits(GPIOB, GPIO_Pin_0);
        waitMs(200);
    } while (true);
    return 0;
}

extern "C" void SysTick_Handler(void)
{
    // Application SysTick default 10ms
}

```



4. Kompilieren und Linken

Der eingegebene Quellcode muss nun in Maschinencode für den Prozessor übersetzt werden. Wählen Sie dazu die Schaltflächen „Kompilieren“ und „Linken“. Bei fehlerfreier Übersetzung liegt das Programm unter dem Namen „Blinklicht.hex“ vor und kann auf den FLASH-Programmspeicher des Prozessors gebrannt werden.

5. Programm übertragen

Das Board mySTM32 Board light verfügt über einen Bootloader. Der Prozessor muss also nicht für die Programmierung aus dem System entfernt werden, um ihn in einem gesonderten Programmiergerät zu brennen, sondern kann direkt programmiert werden.

Verbinden Sie das Board mit dem Programmierkabel an dem USB-Port Ihres Rechners. Zum Übertragen des Programms wählen Sie die Schaltfläche „Brennen“. In Abhängigkeit Ihrer Konfiguration erhalten Sie im Ausgabefenster eine entsprechende Meldung.

6. Mikrocontrollerlösung testen

Zum Testen des Programms ist es erforderlich, dass das Board angeschlossen ist. Das Programm startet automatisch und die LED auf Ihrem Board blinkt.

Hinweis:

Ein ausführliches Tutorial finden Sie unter
www.mySTM32.de