Technical Workshop

Academic High Altitude Conference

Ethan Harstad and Matthew Nelson

Stratospheric Ballooning Association

June 23-24, 2014

Outline

1. Introduction
   - What is mbed
   - Nucleo Development Board
   - An Example Program
2. mbed Basics
   - Your Own Program

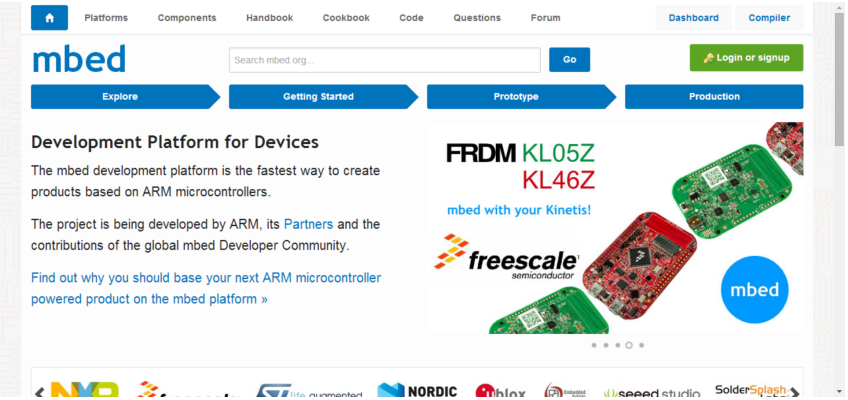
What is mbed

The mbed development platform is a collection of open source hardware and software to allow rapid ARM based prototyping.

- Professional online compiler lets you work from any computer
- Integrated version control system lets you easily find and use open source libraries
- CMSIS based APIs for core and peripheral functions let you work high level or bare metal
- Hardware abstraction layer insulates your application code from hardware changes

Essentially a high performance Arduino with highly integrated tools to save you time

Register on mbed



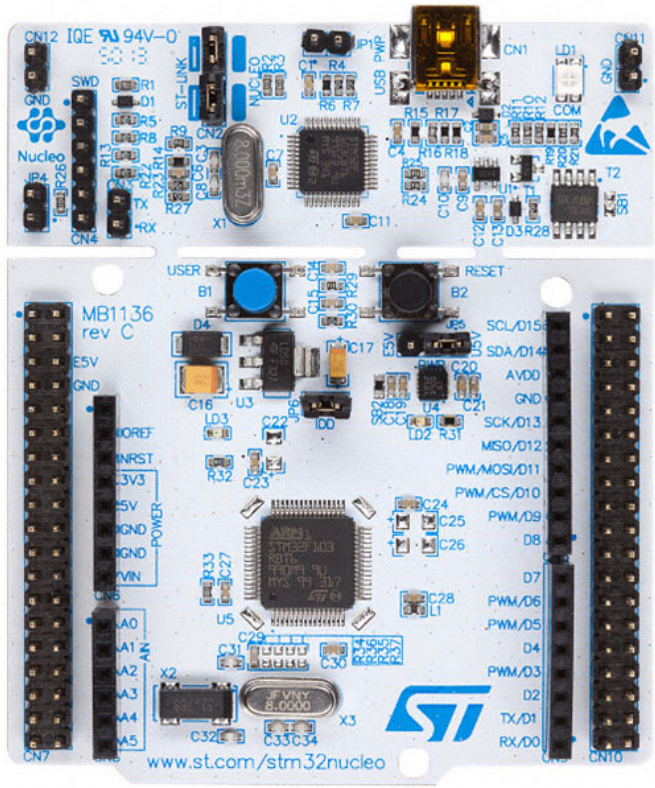
1. Navigate to `http://www.mbed.org`
2. Click the green login or signup button
3. Click the signup button
4. Follow the prompts
5. Confirm your e-mail address

Everyone should have an account on mbed. You can create a team to share programs between users in your organization.

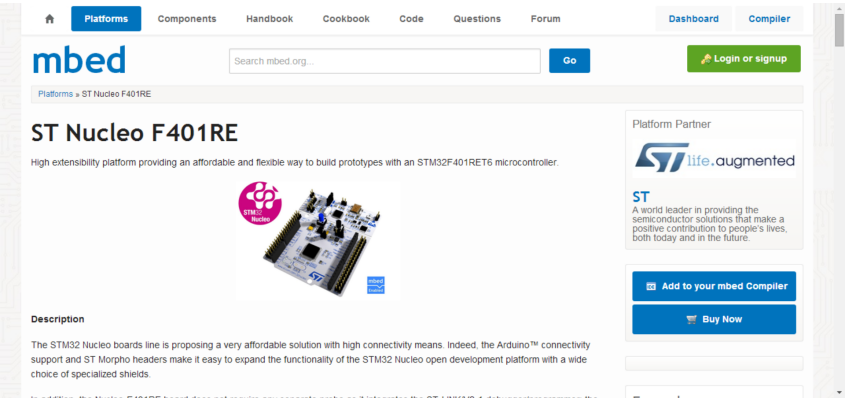
Nucleo Development Board

The Nucleo development board combines a USB programmer with a powerful STM32 processor and Arduino compatible headers

- ARM Cortex-M4 with FPU at 84 MHz
- 512 KBytes flash memory
- 12bit ADC at 2.4 Msps with up to 10 channels
- Up to 3xUART, 3xI2C, 4xSPI interfaces



Add Nucleo to Your Account

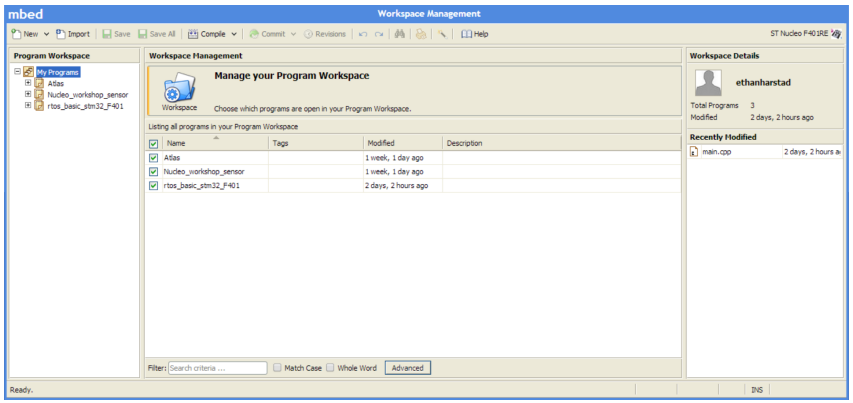


1. Connect your Nucleo to your computer
2. Open the external drive that connects
3. Open the mbed.htm file
4. Click "Add to Your mbed Compiler"

Notice

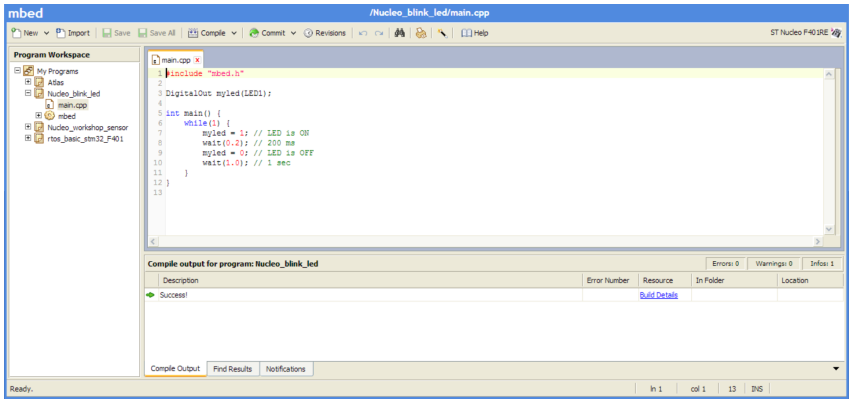
You only need to do this once!

An Example Program



1. Navigate to `http://www.mbed.org`
2. Click the Compiler button
3. Click New and select New Program
4. Choose "Blinky LED" as the program template
5. Name the program anything you desire and click OK

Compile and Upload Your Program



Tip

Set your browsers download location to the Nucleo to save time while debugging

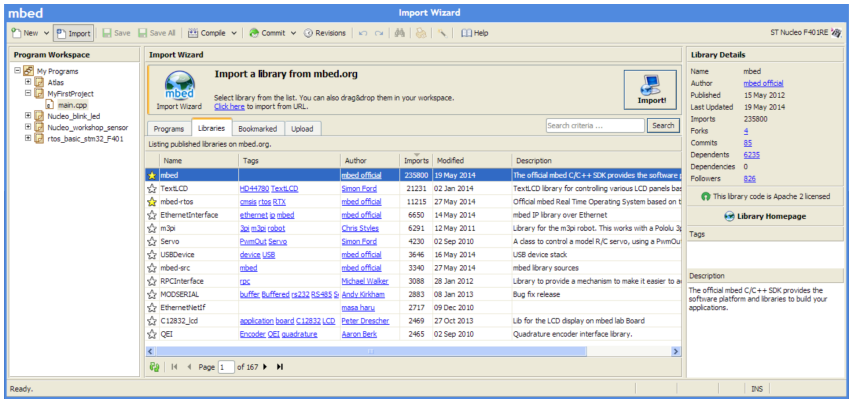
1. Click Compile
2. A file will be downloaded
3. Move this file to the Nucleo external drive
4. The LED will flash red/green while programming
5. When the‘ LED is solid green, your program has started successfully!

Outline

1 Introduction
- What is mbed
- Nucleo Development Board
- An Example Program

2 mbed Basics
- Your Own Program

Create an Empty Program



1. Create a new program with "Empty Program" as the template
2. Create a new file named "main.cpp"
3. Click Import and search for "mbed"

Notice

The mbed library is required for every mbed based program

Write Your First Program

Listing 1: main.cpp

```
1  #include "mbed.h"
2
3  DigitalOut led(LED1);
4
5  int main() {     // Program entry
6    while(1) {     // Main loop
7      led = 1;     // Turn on LED
8      wait(0.2);   // Wait 0.2 seconds
9      led = 0;     // Turn off LED
10     wait(0.8);   // Wait 0.8 seconds
11   }              // Repeat
12 }
```

Breaking Down main.cpp

1 `#include "mbed.h"`

`#include` directive is used to include another file, in this case "mbed.h". You will use this to include libraries in your program. You can also split your program into multiple files and combine them using the include directive.

3 `DigitalOut led(LED1);`

`DigitalOut` is a class from the mbed library that supports digital output. The library also provides `DigitalIn` and `DigitalInOut` for inputs and bidirection pins. This line creates an object named `led` that is attached to the `LED1` pin.

Breaking Down main.cpp

```
5  int main() {     // Program entry
```

This line declares a function called `main` that takes no parameters and returns an integer. Every program needs a function with this signature, it is used as the entry point to the program.

```
6    while(1) {     // Main loop
```

This is an example of a type of loop. It runs while the condition inside the parentheses is true, in the case, always. This loop will never terminate because a microcontroller program should never exit.

Breaking Down main.cpp

```
7       led = 1;      // Turn on LED
8       wait(0.2);    // Wait 0.2 seconds
9       led = 0;      // Turn off LED
10      wait(0.8);    // Wait 0.8 seconds
```

This is what makes the LED flash.
Writing a 1 to an output pin sets the pin high, and a 0 sets the pin low. `wait(sec)` is a function provided by the mbed library that halts the program.

Try It Yourself

Try changing the wait times or making your own patterns.

Outline

1 mbed Platform

What is mbed?

Outline

1 Open Session

What is mbed?