Start developing with ARM mbed

Tools and Workflow

June 2017

AVNET SILICA

mbed Tools Overview

mbed Tools

- Free core tools provide build, debug, test and collaboration workflows
- Third party partner industry tools support

mbed OS DVCS support

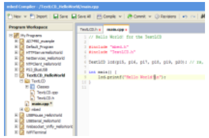
git mercurial

GitHub Atlassian Bitbucket

GitLab **developer.mbed.org**

mbed OS core tools

mbed CLI
Command Line Interface



mbed Compiler
Online IDE

mbed Greentea
Porting Testsuite and CI

mbed pyOCD
CMSIS-DAP Debug Library

mbed DAPLink
CMSIS-DAP Debug Firmware

mbed OS IDEs and toolchains

ARMKEIL
Microcontroller tools
ARM Compiler 5

IAR SYSTEMS

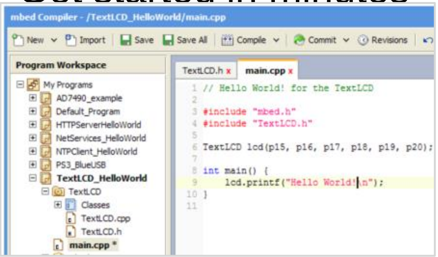
GCC

AC6 SW4STM32 System Workbench for STM32

Developing with mbed OS

Online

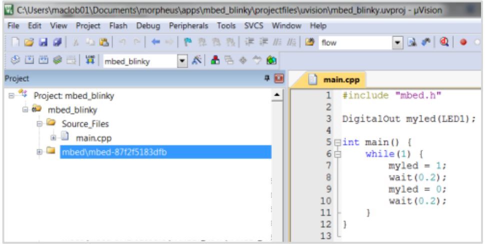
– Instant access to your lightweight C/C++ microcontroller development environment
– All required tools available online
– Get started in minutes



```
// Hello World! for the TextLCD

#include "mbed.h"
#include "TextLCD.h"

TextLCD lcd(p15, p16, p17, p18, p19, p20);

int main() {
    lcd.printf("Hello World!\n");
}
```

Desktop

– Use your favorite IDE to Code & Debug
– Command line build
– Automated Testing
– Common interface across multiple compilers



```
#include "mbed.h"

DigitalOut myled(LED1);

int main() {
    while(1) {
        myled = 1;
        wait(0.2);
        myled = 0;
        wait(0.2);
    }
}
```

```
C:\> mbed
```

All mbed tools focused on collaboration with:

- Software Version Control and reproducibility
- Control and update module dependencies
- Import / Publish Libraries & Applications

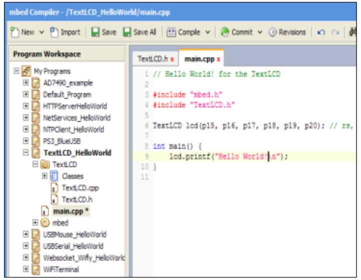
Managing the Source Code

Repo Hosting
mbed.com
GitHub
BitBucket
Private DVCS

import
add
update

publish

mbed Online IDE



```
// Hello World! for the TextLCD

#include "mbed.h"
#include "TextLCD.h"

TextLCD lcd(p15, p16, p17, p18, p19, p20); // rs,

int main() {
    lcd.printf("Hello World!\n");
}
```

Upload project

Download project

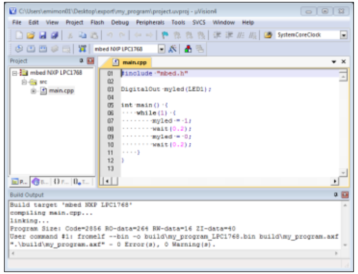
```
C:\> mbed
```

mbed CLI

export

sync

uVision, DS-5, IAR, AC6

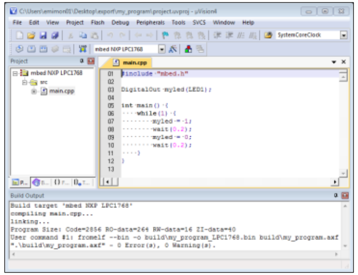


```
#include "mbed.h"

DigitalOut myled(LED1);

int main() {
    while(1) {
        myled = 1;
        wait(0.2);
        myled = 0;
        wait(0.2);
    }
}
```

```
Build target 'mbed NXP LPC1768'
compiling main.cpp...
linking...
Program Size: Code=2856 RO-data=264 RW-data=16 ZI-data=40
User command #1: fromelf --bin -o build\my_program_LPC1768.bin build\my_program.axf
".\build\my_program.axf" - 0 Error(s), 0 Warning(s).
```

Develop, Test, Debug

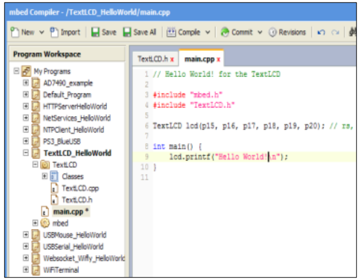
uVision, DS-5, IAR, AC6


C:\Users\emimon01\Desktop\export\my_program\project.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
SystemCoreClock
mbed NXP LPC1768
Project
mbed NXP LPC1768
src
main.cpp
Build Output

```
#include "mbed.h"

DigitalOut myled(LED1);

int main() {
    while(1) {
        myled = 1;
        wait(0.2);
        myled = 0;
        wait(0.2);
    }
}
```

```
Build target 'mbed NXP LPC1768'
compiling main.cpp...
linking...
Program Size: Code=2856 RO-data=264 RW-data=16 ZI-data=40
User command #1: fromelf --bin -o build\my_program_LPC1768.bin build\my_program.axf
".\build\my_program.axf" - 0 Error(s), 0 Warning(s).
```

mbed Online IDE



```
// Hello World! for the TextLCD

#include "mbed.h"
#include "TextLCD.h"

TextLCD lcd(p15, p16, p17, p18, p19, p20); // rs,

int main() {
    lcd.printf("Hello World!\n");
}
```

```
C:\> mbed
```

mbed CLI

CMSIS-DAP Debugging

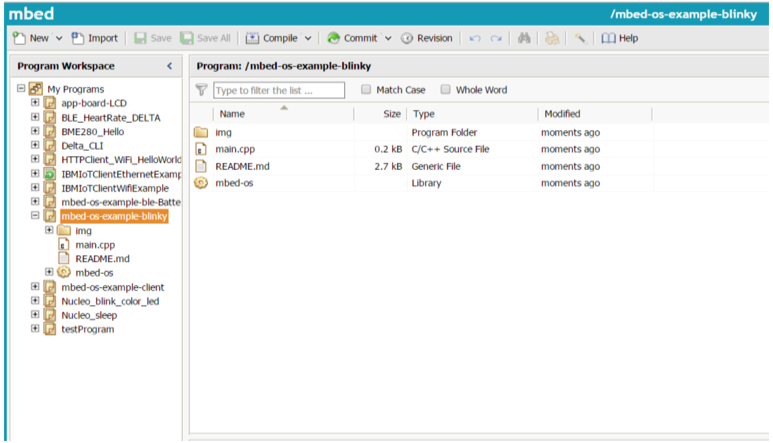
Rapid Prototyping
Drag & Drop
programming

Automated
Tests

mbed
Online Tools

mbed Online Compiler

- Provides a public lightweight online C/C++ IDE
- Uses the professional ARM Compiler 5 toolchain
- Focused on ease of use, reproducibility
- Build-in version control (DVCS) and memory usage table
- Import and export programs with common IDEs
- Web app based. Log in from anywhere - https://developer.mbed.org/compiler

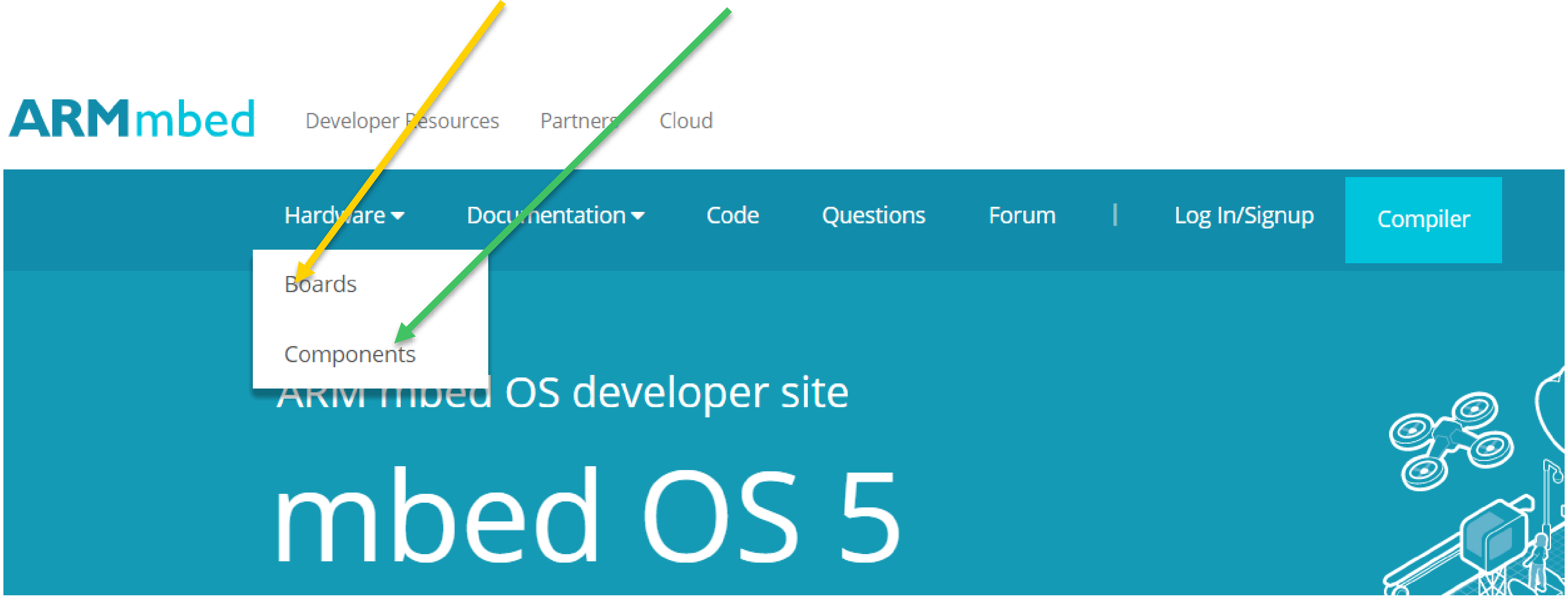


- My Programs
  - app-board-LCD
  - BLE_HeartRate_DELTA
  - BME280_Hello
  - Delta_CLI
  - HTTPClient_WiFi_HelloWorld
  - IBMIoTClientEthernetExamp
  - IBMIoTClientWifiExample
  - mbed-os-example-ble-Batte
  - mbed-os-example-blinky
    - img
    - main.cpp
    - README.md
    - mbed-os
  - mbed-os-example-client
  - Nucleo_blink_color_led
  - Nucleo_sleep
  - testProgram

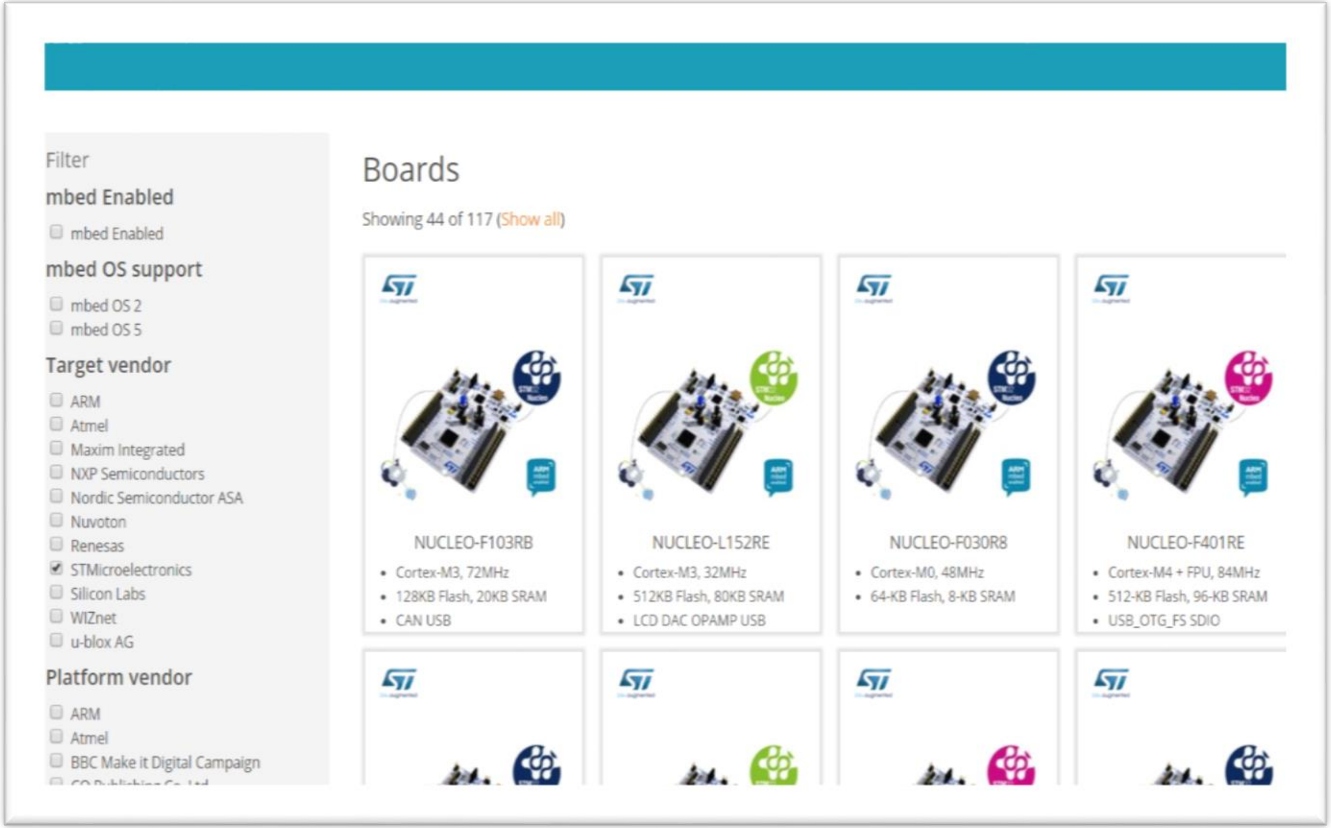
Name	Size	Type	Modified
img		Program Folder	moments ago
main.cpp	0.2 kB	C/C++ Source File	moments ago
README.md	2.7 kB	Generic File	moments ago
mbed-os		Library	moments ago

ARM mbed Boards & Components

Choose your favorite **board** and **components**.



Boards



Filter

mbed Enabled

□ mbed Enabled

mbed OS support

□ mbed OS 2
□ mbed OS 5

Target vendor

□ ARM
□ Atmel
□ Maxim Integrated
□ NXP Semiconductors
□ Nordic Semiconductor ASA
□ Nuvoton
□ Renesas
☑ STMicroelectronics
□ Silicon Labs
□ WIZnet
□ u-blox AG

Platform vendor

□ ARM
□ Atmel
□ BBC Make it Digital Campaign
□ CQ Publishing Co.,Ltd

Boards

Showing 44 of 117 (Show all)

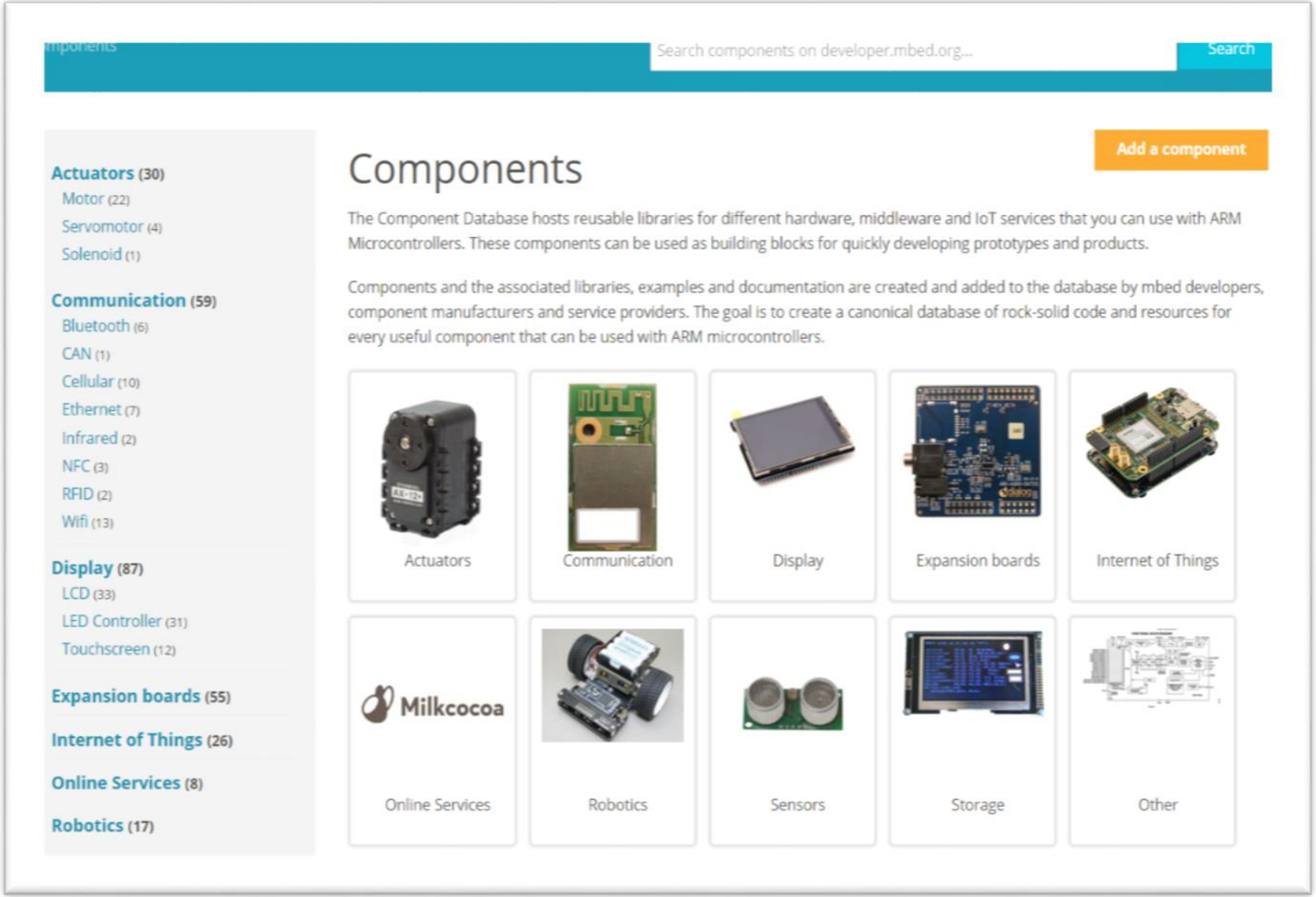
NUCLEO-F103RB
- Cortex-M3, 72MHz
- 128KB Flash, 20KB SRAM
- CAN USB

NUCLEO-L152RE
- Cortex-M3, 32MHz
- 512KB Flash, 80KB SRAM
- LCD DAC OPAMP USB

NUCLEO-F030R8
- Cortex-M0, 48MHz
- 64-KB Flash, 8-KB SRAM

NUCLEO-F401RE
- Cortex-M4 + FPU, 84MHz
- 512-KB Flash, 96-KB SRAM
- USB_OTG_FS SDIO

Components



Actuators (30)
- Motor (22)
- Servomotor (4)
- Solenoid (1)

Communication (59)
- Bluetooth (6)
- CAN (1)
- Cellular (10)
- Ethernet (7)
- Infrared (2)
- NFC (3)
- RFID (2)
- Wifi (13)

Display (87)
- LCD (33)
- LED Controller (31)
- Touchscreen (12)

Expansion boards (55)

Internet of Things (26)

Online Services (8)

Robotics (17)

Components

The Component Database hosts reusable libraries for different hardware, middleware and IoT services that you can use with ARM Microcontrollers. These components can be used as building blocks for quickly developing prototypes and products.

Components and the associated libraries, examples and documentation are created and added to the database by mbed developers, component manufacturers and service providers. The goal is to create a canonical database of rock-solid code and resources for every useful component that can be used with ARM microcontrollers.

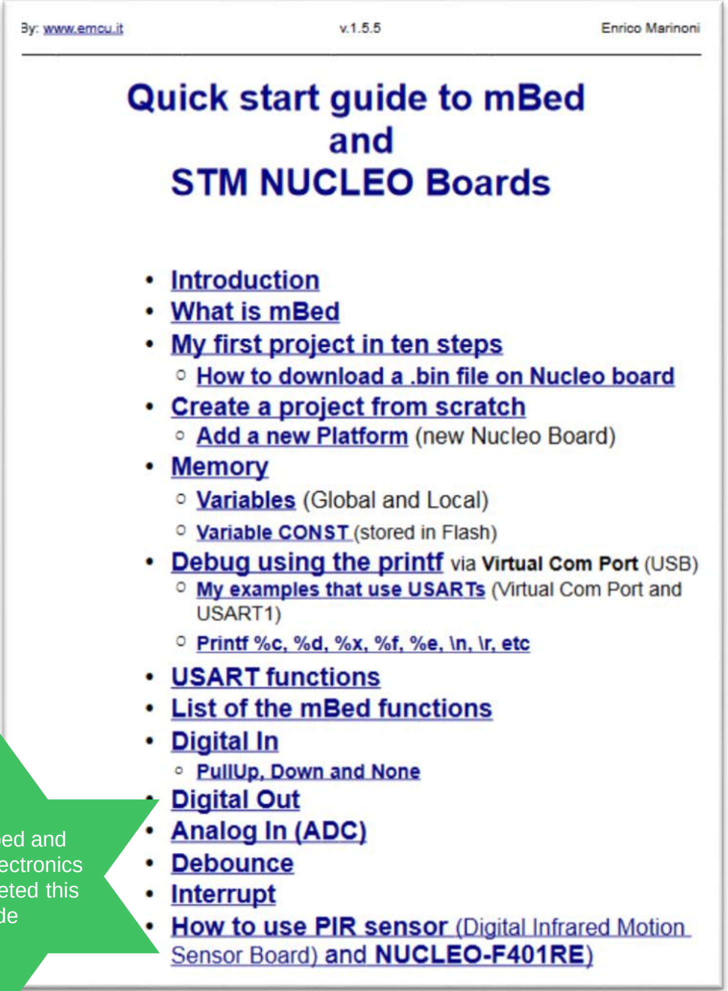
ARM mbed
Finding documentation

For more info see the:

- mbed **API** documentation that is here
- Fast and Effective Embedded Systems Desig
- Quick start guide to mBed and STM NUCLEO Boards
- Introduction to the mbed OS 5 handbook

Also see the links below:

- mbed home page
- The official mbed C/C++ SDK provides the software platform and libraries to build your applications• mBED compiler
- C++ Basics
- mbed and NUCLEO tutorials



By: www.emcu.it v.1.5.5 Enrico Marinoni

Quick start guide to mBed and STM NUCLEO Boards

- Introduction
- What is mBed
- My first project in ten steps
  - How to download a .bin file on Nucleo board
- Create a project from scratch
  - Add a new Platform (new Nucleo Board)
- Memory
  - Variables (Global and Local)
  - Variable CONST (stored in Flash)
- Debug using the printf via **Virtual Com Port** (USB)
  - My examples that use USARTs (Virtual Com Port and USART1)
  - Printf %c, %d, %x, %f, %e, \n, \r, etc
- USART functions
- List of the mBed functions
- Digital In
  - PullUp, Down and None
- Digital Out
- Analog In (ADC)
- Debounce
- Interrupt
- How to use PIR sensor (Digital Infrared Motion Sensor Board) and NUCLEO-F401RE)

ST Corporate Comm ha ritwittato la tua risposta · 20 mar 2015

EMCU @EMCU_IT
@ST_News @ARMmbed Quick start guide - mBed and Nucleo Boards is h...

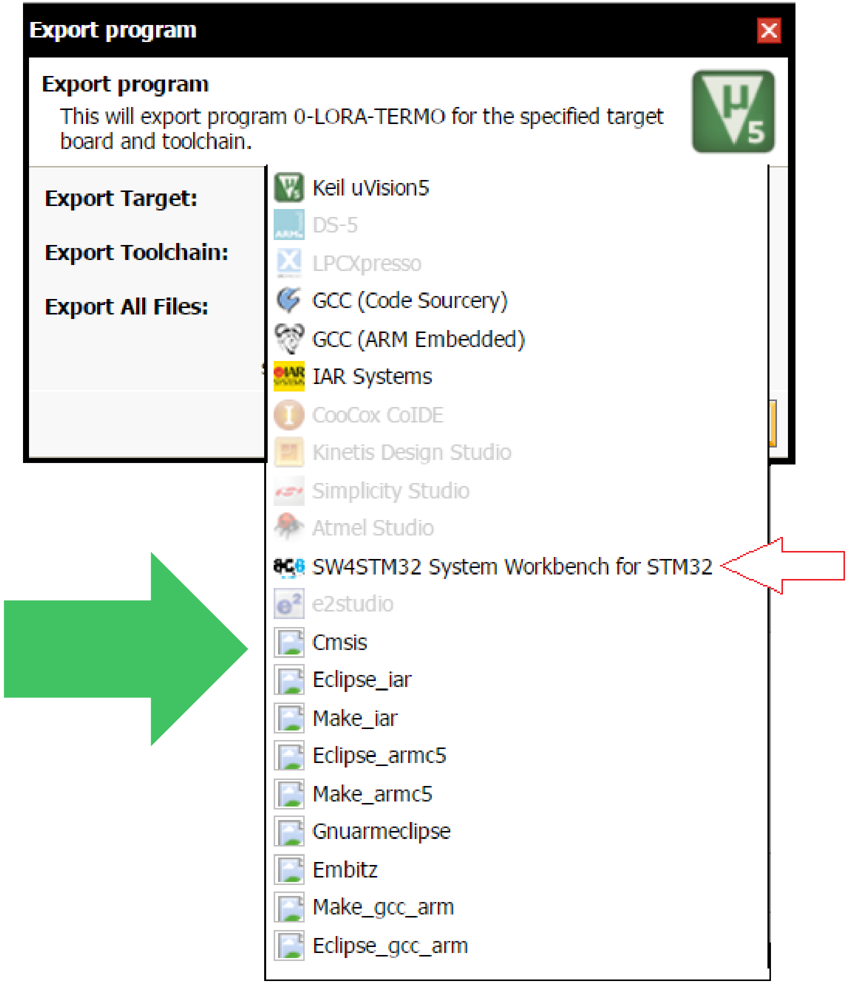
ARM mbed @ARMmbed · 18 feb 2015
RT @EMCU_IT: @ST_News @ARMmbed Quick start guide - #mbed and Nucleo Boards is here: bit.ly/1DCOSxa

Traduci dalla lingua originale: inglese

ARM mbed and STMicroelectronics has retweeted this guide

Exporting Project

mbed allows you to **export the program** to external IDE, that at the moment are:



Export program

Export program

This will export program 0-LORA-TERMO for the specified target board and toolchain.

Export Target:

Export Toolchain:

Export All Files:

- Keil uVision5
- DS-5
- LPCXpresso
- GCC (Code Sourcery)
- GCC (ARM Embedded)
- IAR Systems
- CooCox CoIDE
- Kinetis Design Studio
- Simplicity Studio
- Atmel Studio
- SW4STM32 System Workbench for STM32
- e2studio
- Cmsis
- Eclipse_iar
- Make_iar
- Eclipse_armc5
- Make_armc5
- Gnuarmeclipse
- Embitz
- Make_gcc_arm
- Eclipse_gcc_arm

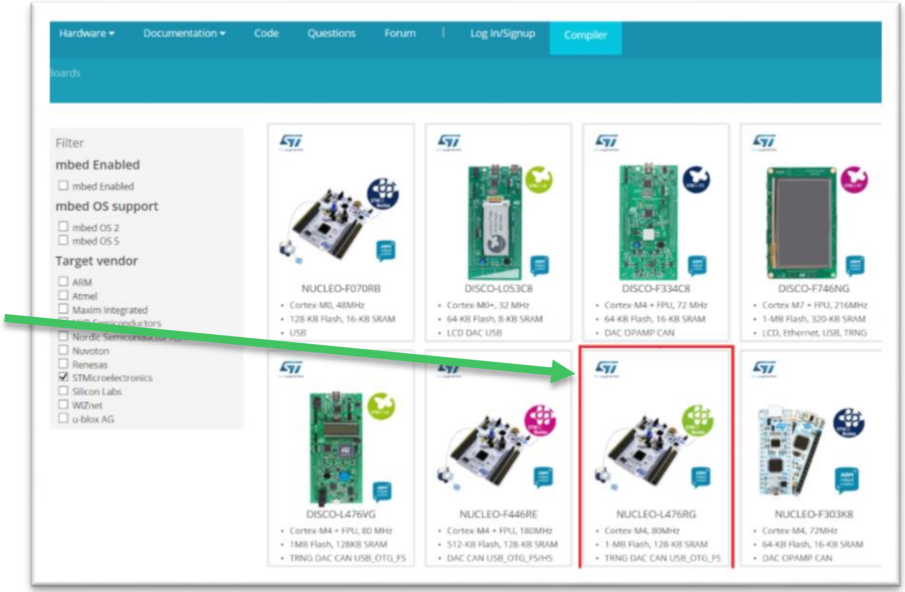
AC6

mbed
First Project

mbed
First Project In Ten Steps

1. Choose the **NUCLEO board** that you need to use from the mbed platform page.
Here there is the list of the **STM32-NUCLEO** boards.

2. We decided to use the **NUCLEO-L476RG** but you can choose what you want.



Filter

mbed Enabled

□ mbed Enabled

mbed OS support

□ mbed OS 2
□ mbed OS 5

Target vendor

□ ARM
□ Atmel
□ Maxim Integrated
□ NXP Semiconductors
□ Nordic Semiconductor ASA
□ Nuvoton
□ Renesas
☑ STMicroelectronics
□ Silicon Labs
□ WIZnet
□ u-blox AG

NUCLEO-F070RB
- Cortex-M0, 48MHz
- 128-KB Flash, 16-KB SRAM
- USB

DISCO-L053C8
- Cortex-M0+, 32 MHz
- 64-KB Flash, 8-KB SRAM
- LCD DAC USB

DISCO-F334C8
- Cortex-M4 + FPU, 72 MHz
- 64-KB Flash, 16-KB SRAM
- DAC OPAMP CAN

DISCO-F746NG
- Cortex M7 + FPU, 216MHz
- 1-MB Flash, 320-KB SRAM
- LCD, Ethernet, USB, TRNG

DISCO-L476VG
- Cortex-M4 + FPU, 80 MHz
- 1MB Flash, 128KB SRAM
- TRNG DAC CAN USB_OTG_FS

NUCLEO-F446RE
- Cortex-M4 + FPU, 180MHz
- 512-KB Flash, 128-KB SRAM
- DAC CAN USB_OTG_FS/HS

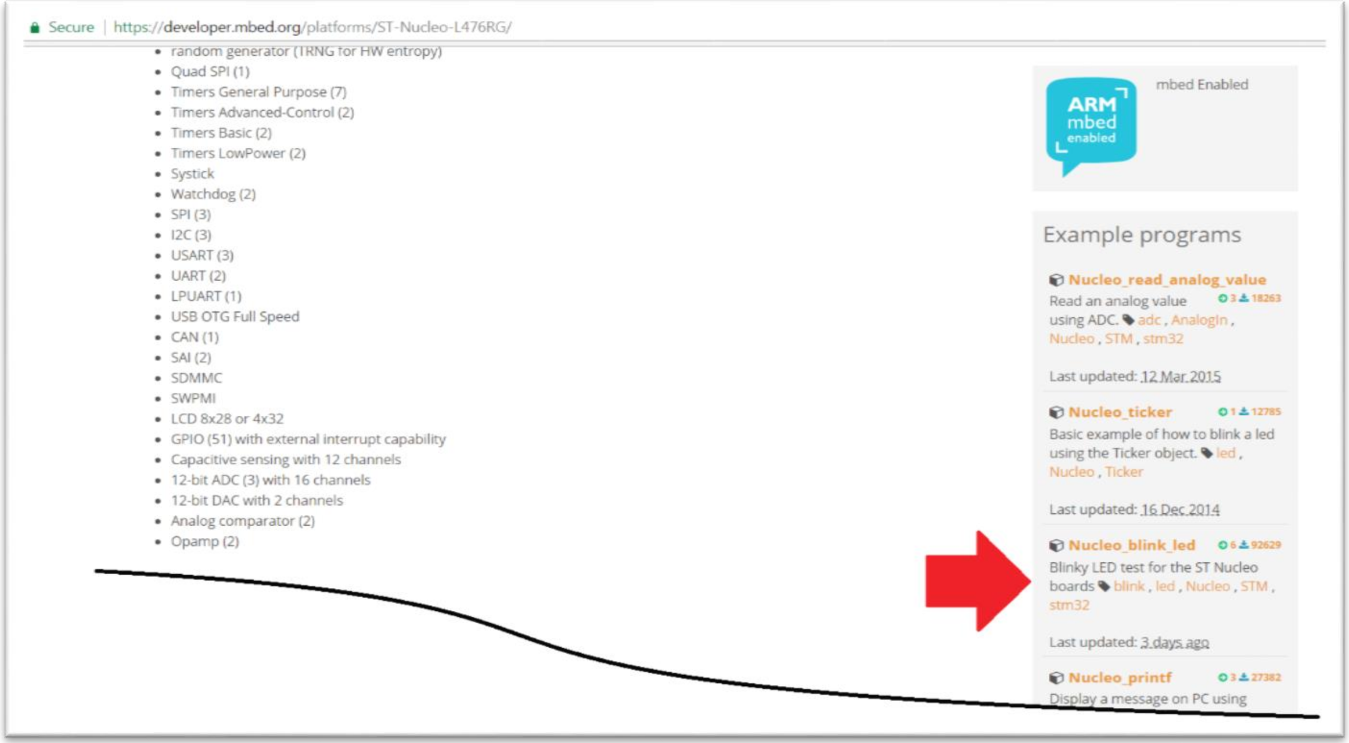
NUCLEO-L476RG
- Cortex-M4, 80MHz
- 1-MB Flash, 128-KB SRAM
- TRNG DAC CAN USB_OTG_FS

NUCLEO-F303K8
- Cortex-M4, 72MHz
- 64-KB Flash, 16-KB SRAM
- DAC OPAMP CAN

3. From the page that appears you have all the information regarding your NUCLEO board.

On the right of the page there are the Example programs, please choose the:

Nucleo_blink_led



Secure | https://developer.mbed.org/platforms/ST-Nucleo-L476RG/

- random generator (TRNG for HW entropy)
- Quad SPI (1)
- Timers General Purpose (7)
- Timers Advanced-Control (2)
- Timers Basic (2)
- Timers LowPower (2)
- Systick
- Watchdog (2)
- SPI (3)
- I2C (3)
- USART (3)
- UART (2)
- LPUART (1)
- USB OTG Full Speed
- CAN (1)
- SAI (2)
- SDMMC
- SWPMI
- LCD 8x28 or 4x32
- GPIO (51) with external interrupt capability
- Capacitive sensing with 12 channels
- 12-bit ADC (3) with 16 channels
- 12-bit DAC with 2 channels
- Analog comparator (2)
- Opamp (2)

mbed Enabled

Example programs

Nucleo_read_analog_value 3 18263

Read an analog value using ADC. adc , AnalogIn , Nucleo , STM , stm32

Last updated: 12 Mar 2015

Nucleo_ticker 1 12785

Basic example of how to blink a led using the Ticker object. led , Nucleo , Ticker

Last updated: 16 Dec 2014

Nucleo_blink_led 6 92629

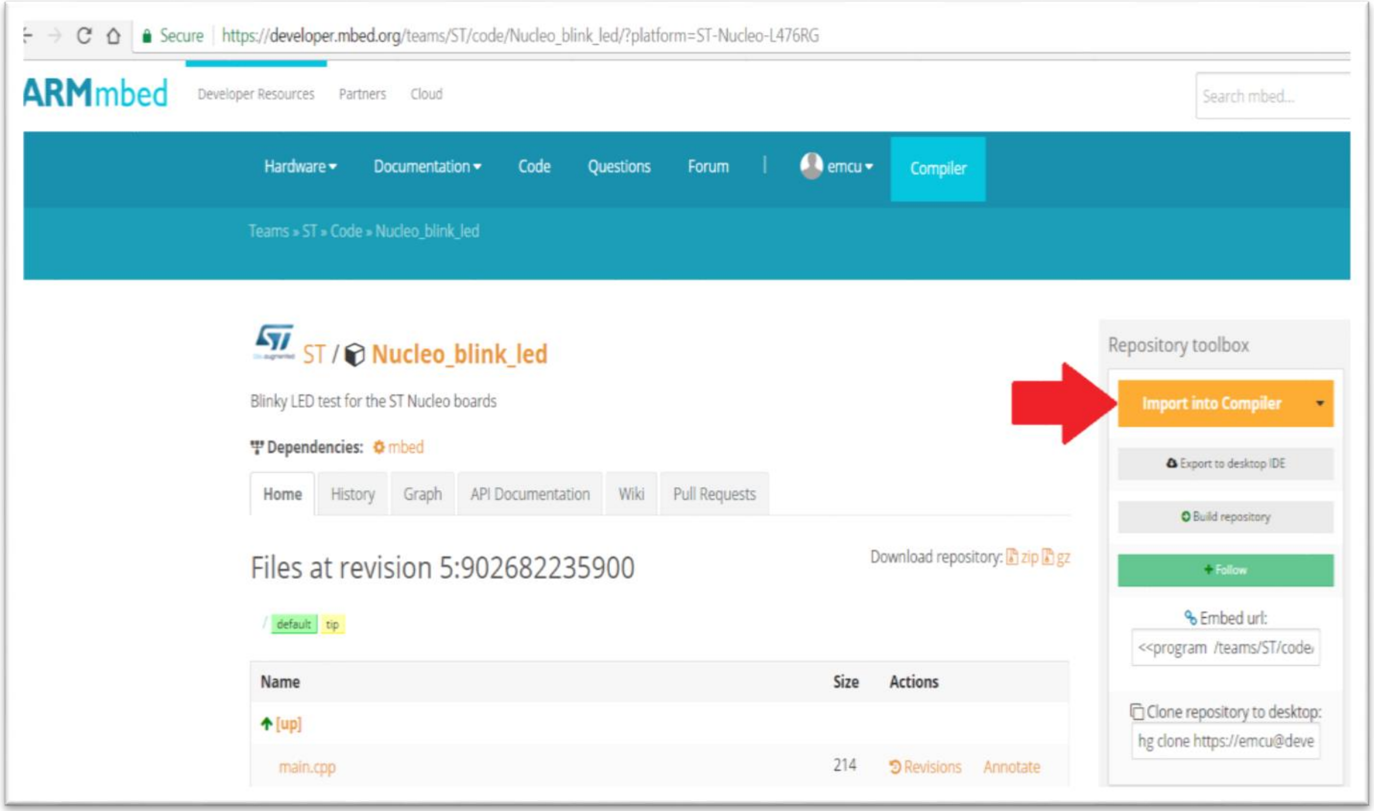
Blinky LED test for the ST Nucleo boards blink , led , Nucleo , STM , stm32

Last updated: 3 days ago

Nucleo_printf 3 27382

Display a message on PC using

4. A new page will be opened and from this page choose: **Import into Compiler**



Secure https://developer.mbed.org/teams/ST/code/Nucleo_blink_led/?platform=ST-Nucleo-L476RG
ARMmbed
Developer Resources
Partners
Cloud
Search mbed...
Hardware
Documentation
Code
Questions
Forum
emcu
Compiler
Teams » ST » Code » Nucleo_blink_led

ST / Nucleo_blink_led

Blinky LED test for the ST Nucleo boards

Dependencies: mbed

Files at revision 5:902682235900

Download repository: zip gz

/ default tip

Name	Size	Actions
[up]		
main.cpp	214	Revisions Annotate

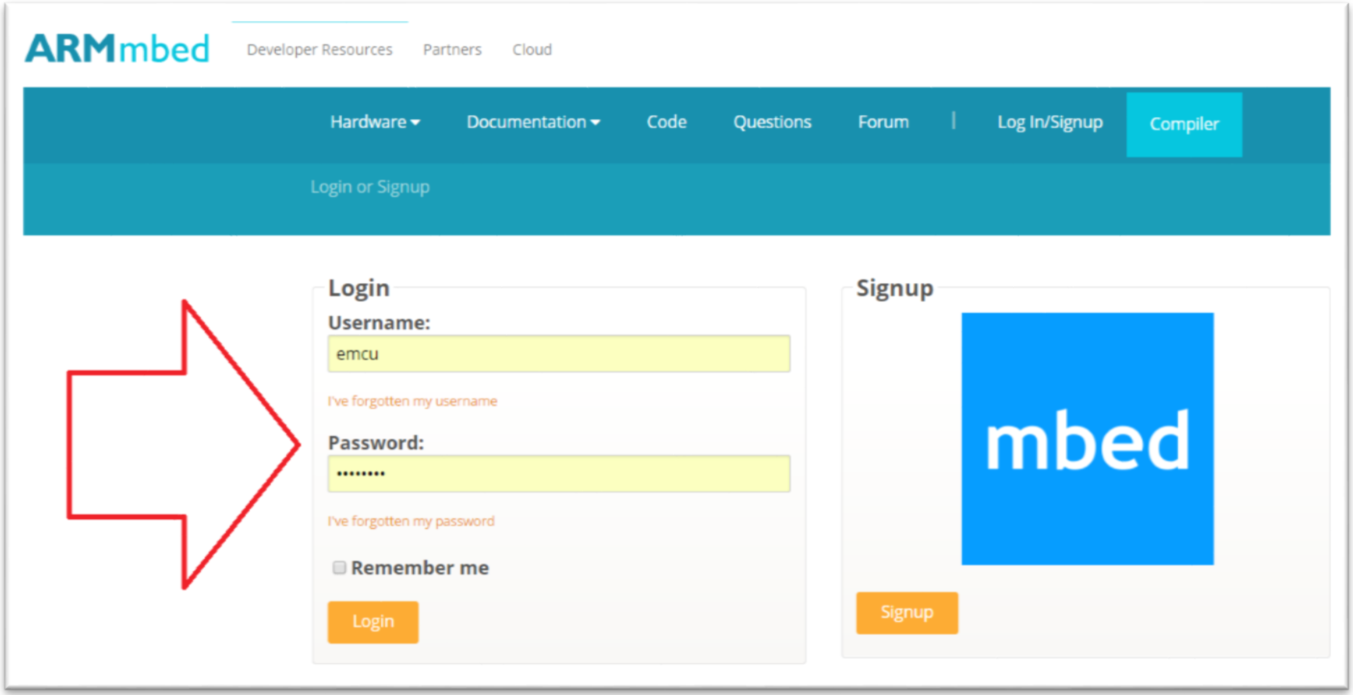
Embed url:

<<program /teams/ST/code/

Clone repository to desktop:

hg clone https://emcu@deve

From the new window that appear, choose login into your mbed account.



Login

Username: emcu

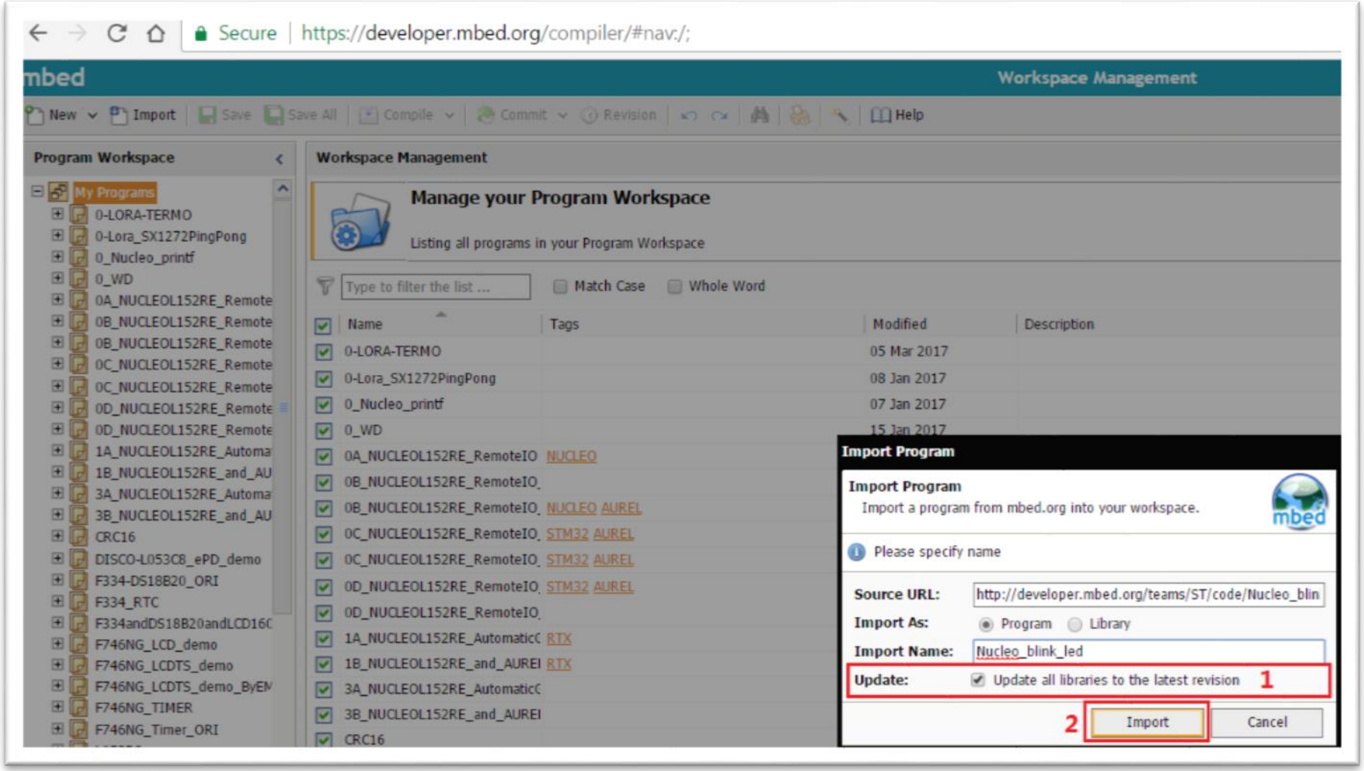
I've forgotten my username

Password: ••••••••

I've forgotten my password

□ Remember me

5. At this point the compiler will start and you must see something like below.
Please select: **Update all libraries to latest revision**
and select: **Import**



Manage your Program Workspace

Listing all programs in your Program Workspace

Type to filter the list ... □ Match Case □ Whole Word

☑ Name	Tags	Modified	Description
☑ 0-LORA-TERMO		05 Mar 2017	
☑ 0-Lora_SX1272PingPong		08 Jan 2017	
☑ 0_Nucleo_printf		07 Jan 2017	
☑ 0_WD		15 Jan 2017	
☑ 0A_NUCLEOL152RE_RemoteIO	NUCLEO		
☑ 0B_NUCLEOL152RE_RemoteIO_			
☑ 0B_NUCLEOL152RE_RemoteIO_	NUCLEO AUREL		
☑ 0C_NUCLEOL152RE_RemoteIO_	STM32 AUREL		
☑ 0C_NUCLEOL152RE_RemoteIO_	STM32 AUREL		
☑ 0D_NUCLEOL152RE_RemoteIO_	STM32 AUREL		
☑ 0D_NUCLEOL152RE_RemoteIO_			
☑ 1A_NUCLEOL152RE_AutomaticC	RTX		
☑ 1B_NUCLEOL152RE_and_AUREl	RTX		
☑ 3A_NUCLEOL152RE_AutomaticC			
☑ 3B_NUCLEOL152RE_and_AUREl			
☑ CRC16			

Import Program

Import Program
Import a program from mbed.org into your workspace.

Please specify name

Source URL: http://developer.mbed.org/teams/ST/code/Nucleo_blin
Import As: ● Program ○ Library
Import Name: Nucleo_blink_led
Update: ☑ Update all libraries to the latest revision 1

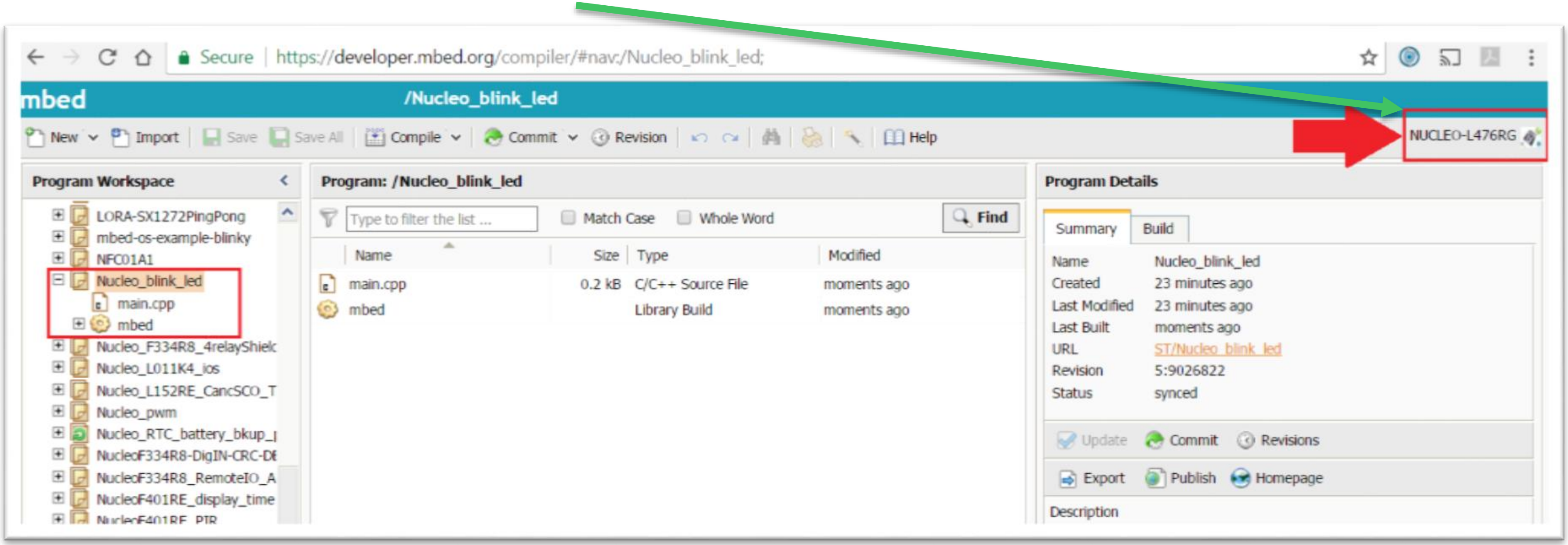
2 Import Cancel

6. At the end of the import procedure you must see something like below.

ATTENTION:

It is important that your board is **NUCLEO-L476RG**.

If it is wrong, press on the name of the board and choose the right Nucleo.



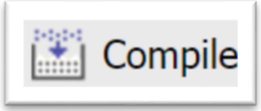
Program Workspace:

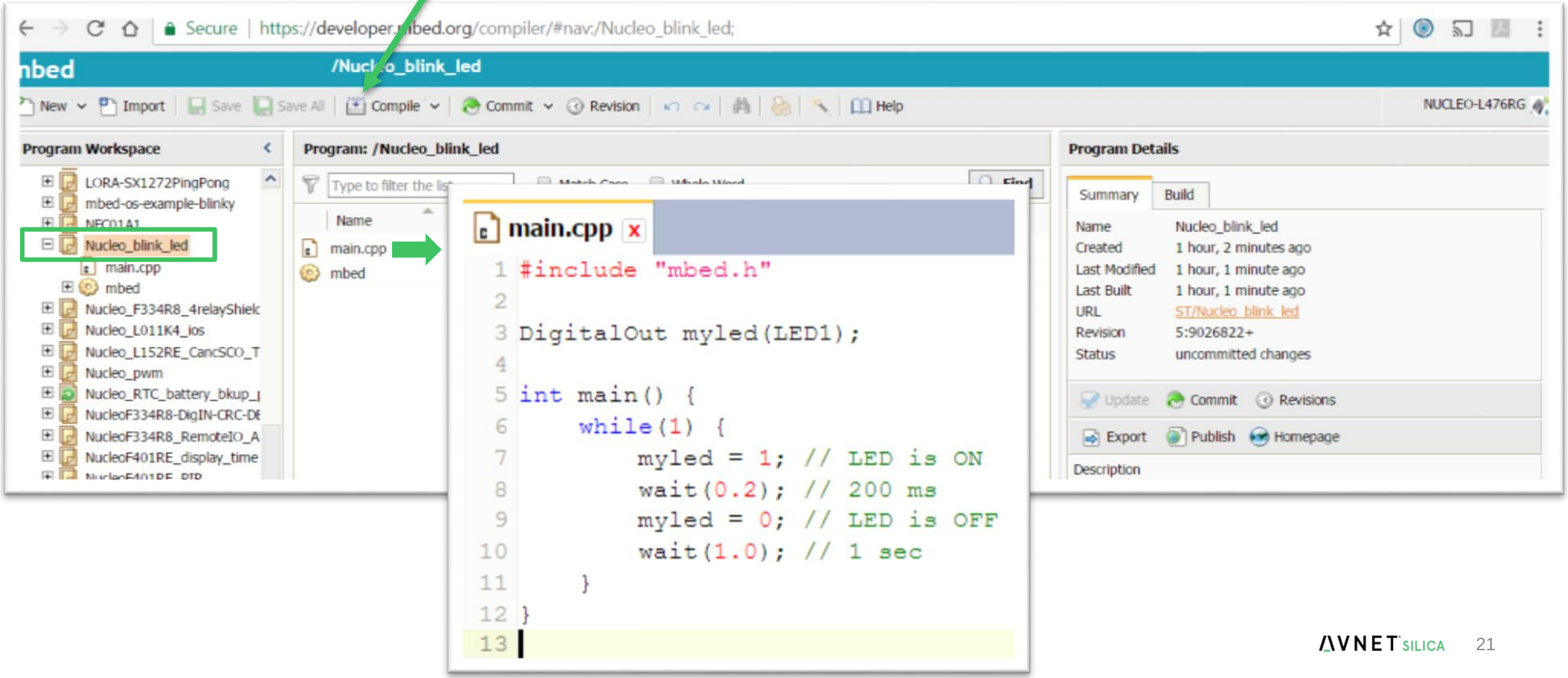
- LORA-SX1272PingPong
- mbed-os-example-blinky
- NFC01A1
- Nucleo_blink_led
  - main.cpp
  - mbed
- Nucleo_F334R8_4relayShielc
- Nucleo_L011K4_ios
- Nucleo_L152RE_CancSCO_T
- Nucleo_pwm
- Nucleo_RTC_battery_bkup_
- NucleoF334R8-DigIN-CRC-DE
- NucleoF334R8_RemoteIO_A
- NucleoF401RE_display_time
- NucleoF401RE_PIR

Name	Size	Type	Modified
main.cpp	0.2 kB	C/C++ Source File	moments ago
mbed		Library Build	moments ago

Name	Nucleo_blink_led
Created	23 minutes ago
Last Modified	23 minutes ago
Last Built	moments ago
URL	ST/Nucleo_blink_led
Revision	5:9026822
Status	synced

7. Now click on the **COMPILE** icon



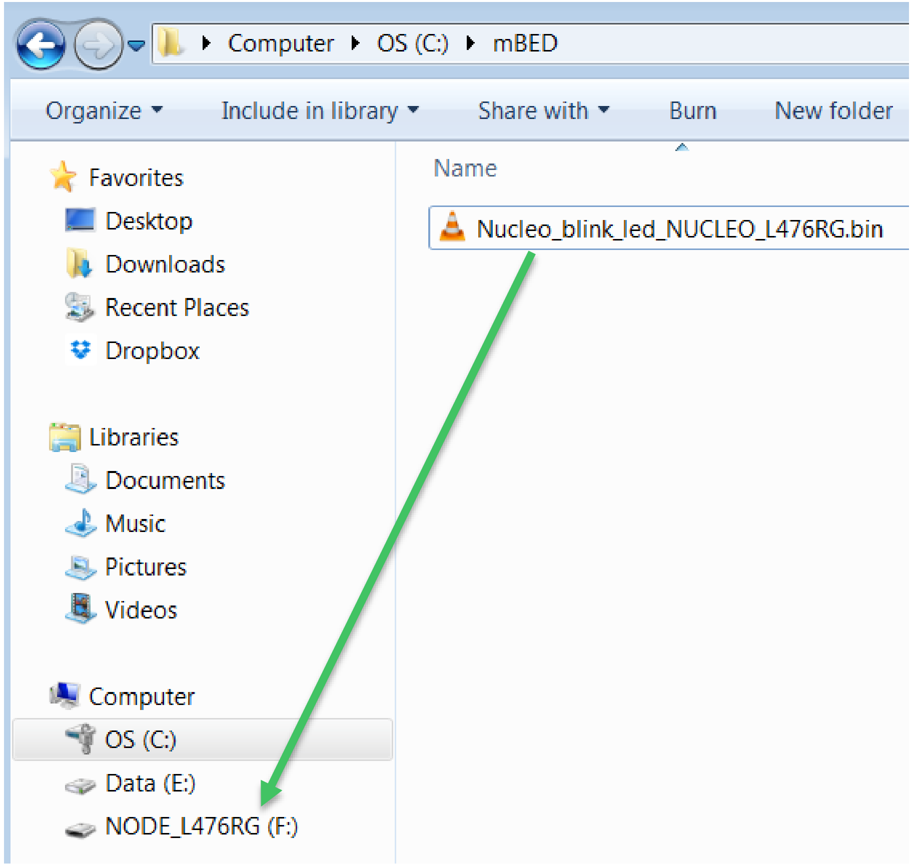


Name	Nucleo_blink_led
Created	1 hour, 2 minutes ago
Last Modified	1 hour, 1 minute ago
Last Built	1 hour, 1 minute ago
URL	ST/Nucleo_blink_led
Revision	5:9026822+
Status	uncommitted changes

```
#include "mbed.h"

DigitalOut myled(LED1);

int main() {
    while(1) {
        myled = 1; // LED is ON
        wait(0.2); // 200 ms
        myled = 0; // LED is OFF
        wait(1.0); // 1 sec
    }
}

```

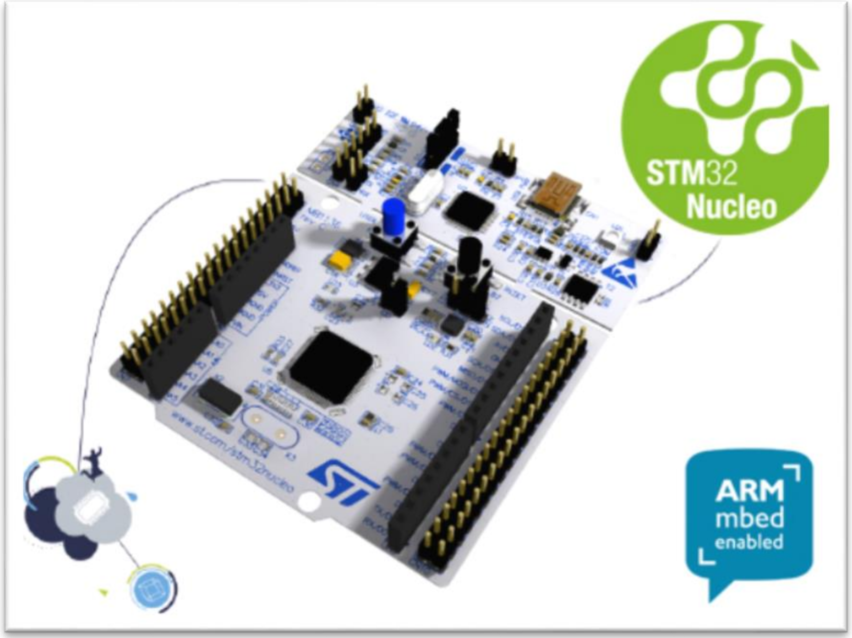
8. At the end of the compilation, mbed asks us where save the **bin** file. Choose a directory and save it.

9. Now to program your NUCLEO board is only necessary to **drag and drop the bin file on the NUCLEO board** icon.
In other words:
select the .bin file, drag it on the NUCLEO icon and release it.



10. Now you must see the green LED that blinking.

Congratulation, your first program is running.



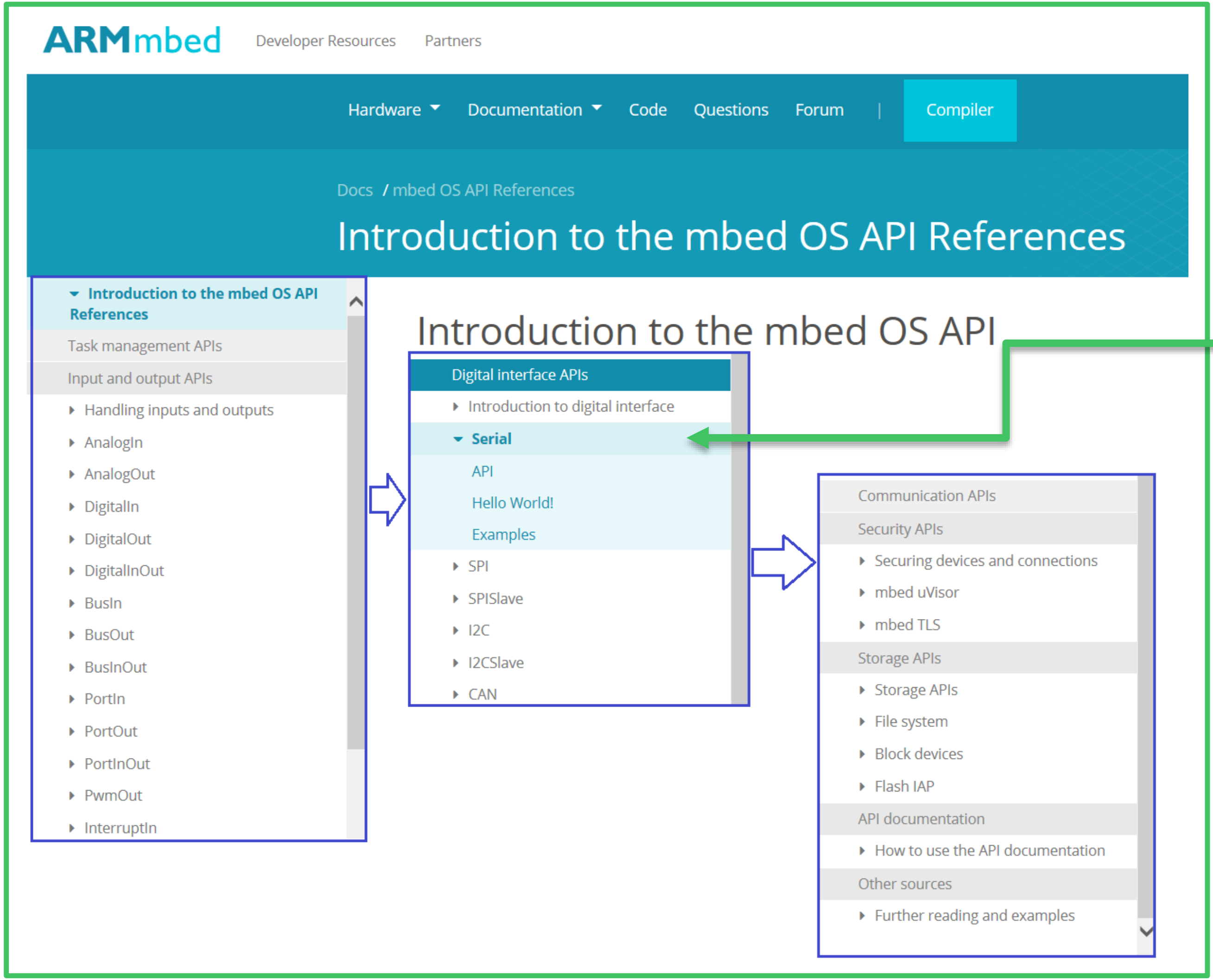
Debug using the **printf** via Virtual Com port (USB)

Debugging your programs with online tools require using **printf**.
To do this is very easy, follow the steps below.

Define the Virtual Com port to redirect the printf.
On NUCLEO boards I suggest to use the declaration shown below.
Serial pc(SERIAL_TX, SERIAL_RX);

By default the Virtual Comm configuration is:
9600-8-N-1 FlowControl None
More info are **here**.

Now you can use the **printf** to send to your PC the data that you want.
On PC I suggest to use **TeraTerm**.


ARMmbed   Developer Resources   Partners

Hardware ▾   Documentation ▾   Code   Questions   Forum | Compiler

Docs / mbed OS API References

Introduction to the mbed OS API References

- Introduction to the mbed OS API References
- Task management APIs
- Input and output APIs
  - Handling inputs and outputs
  - AnalogIn
  - AnalogOut
  - DigitalIn
  - DigitalOut
  - DigitalInOut
  - BusIn
  - BusOut
  - BusInOut
  - PortIn
  - PortOut
  - PortInOut
  - PwmOut
  - InterruptIn

Introduction to the mbed OS API

- Digital interface APIs
  - Introduction to digital interface
  - Serial
    - API
    - Hello World!
    - Examples
  - SPI
  - SPISlave
  - I2C
  - I2CSlave
  - CAN

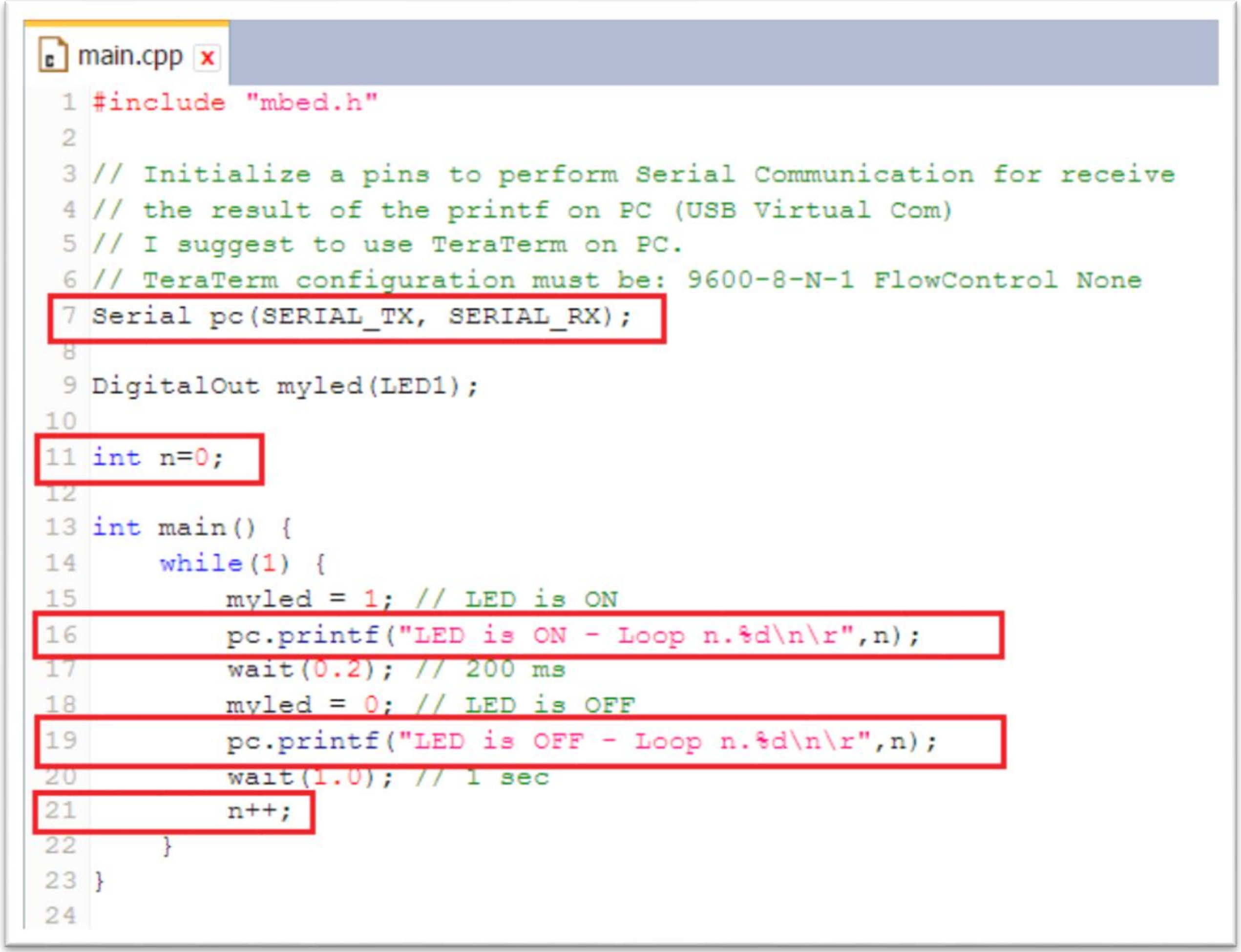
- Communication APIs
- Security APIs
  - Securing devices and connections
  - mbed uVisor
  - mbed TLS
- Storage APIs
  - Storage APIs
  - File system
  - Block devices
  - Flash IAP
- API documentation
  - How to use the API documentation
- Other sources
  - Further reading and examples

More info concerning **API** SERIAL port are **here**.

Take the previous program and insert the highlighted lines in the red rectangles.

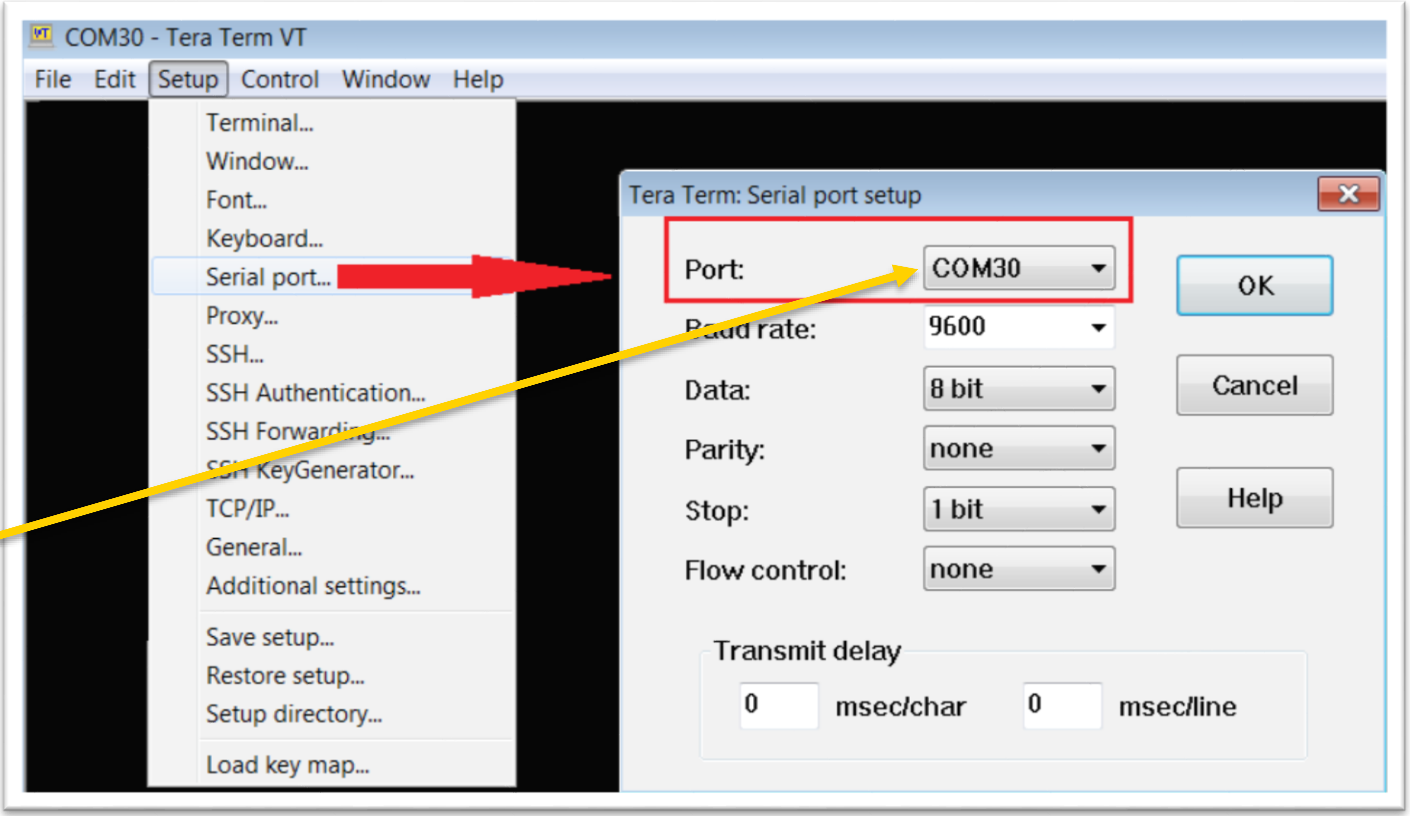
Compile the program and download it on the NUCLEO-L476RG

More info concerning **printf** and **serial port** are **here**.



```
#include "mbed.h"

// Initialize a pins to perform Serial Communication for receive
// the result of the printf on PC (USB Virtual Com)
// I suggest to use TeraTerm on PC.
// TeraTerm configuration must be: 9600-8-N-1 FlowControl None
Serial pc(SERIAL_TX, SERIAL_RX);

DigitalOut myled(LED1);

int n=0;

int main() {
    while(1) {
        myled = 1; // LED is ON
        pc.printf("LED is ON - Loop n.%d\n\r",n);
        wait(0.2); // 200 ms
        myled = 0; // LED is OFF
        pc.printf("LED is OFF - Loop n.%d\n\r",n);
        wait(1.0); // 1 sec
        n++;
    }
}
```

Open **TeraTerm** and configure it as shown here.



Port: COM30

Baud rate: 9600

Data: 8 bit

Parity: none

Stop: 1 bit

Flow control: none

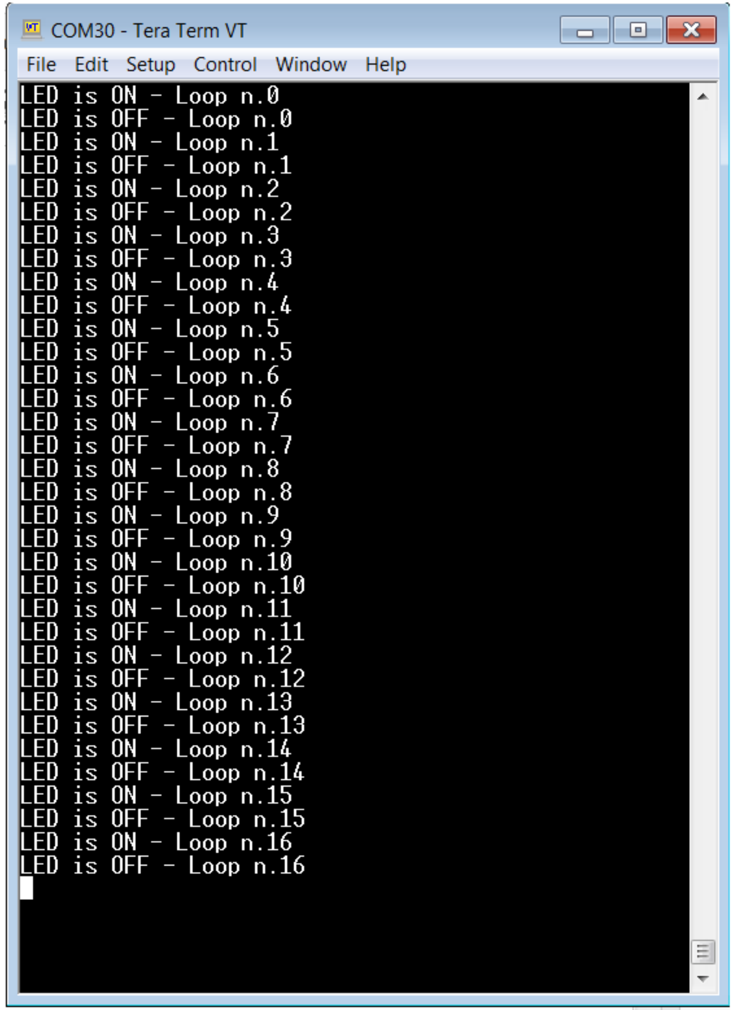
0 msec/char

0 msec/line

Choose the correct
COM PORT

On TeraTerm you must see something like as shown here.

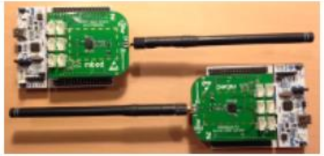
Congratulation, your second program is running.



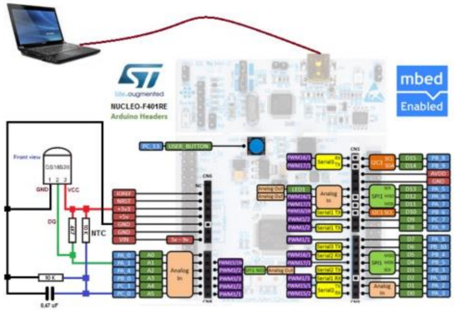
```
LED is ON - Loop n.0
LED is OFF - Loop n.0
LED is ON - Loop n.1
LED is OFF - Loop n.1
LED is ON - Loop n.2
LED is OFF - Loop n.2
LED is ON - Loop n.3
LED is OFF - Loop n.3
LED is ON - Loop n.4
LED is OFF - Loop n.4
LED is ON - Loop n.5
LED is OFF - Loop n.5
LED is ON - Loop n.6
LED is OFF - Loop n.6
LED is ON - Loop n.7
LED is OFF - Loop n.7
LED is ON - Loop n.8
LED is OFF - Loop n.8
LED is ON - Loop n.9
LED is OFF - Loop n.9
LED is ON - Loop n.10
LED is OFF - Loop n.10
LED is ON - Loop n.11
LED is OFF - Loop n.11
LED is ON - Loop n.12
LED is OFF - Loop n.12
LED is ON - Loop n.13
LED is OFF - Loop n.13
LED is ON - Loop n.14
LED is OFF - Loop n.14
LED is ON - Loop n.15
LED is OFF - Loop n.15
LED is ON - Loop n.16
LED is OFF - Loop n.16
```

mbed **examples**

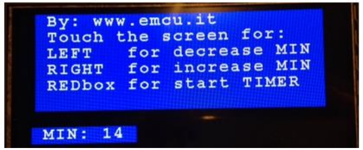
Lora kit and NUCLEO board is here



NUCLEO-F401RE + **DS18B20** + **Thermistor** is here

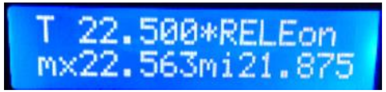


CountDown TIMER from 1 to 199 minutes is here



```
By: www.emcu.it
Touch the screen for:
LEFT    for decrease MIN
RIGHT   for increase MIN
REDbox  for start TIMER

MIN: 14
```

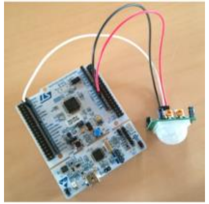
Home **thermostat** is here



```
T 22.500*RELEon
mx22.563mi21.875
```

Automatic sliding gate is here

How to use **PIR sensor** is here



Other examples are here

WWW.EMCU.EU

Special Thanks

Avnet Silica wishes to thank Enrico Marinoni for his help in supporting this event and ST Microelectronics products.

We recommend to check **www.emcu.eu** a great website, mantained by Enrico focusing on STM32 MCUs, PLM, WireLes and MEMS.

Thank you!