

# Výuková laboratorní sestava seminář pro učitele

## Programování kitů v ARM MBED

Petr Kůrka  
([kurkape6@fel.cvut.cz](mailto:kurkape6@fel.cvut.cz))

Katedra měření

Fakulta elektrotechnická

České vysoké učení  
technické v Praze

# Úvod – co je MBED

- Programovací prostředí pro ARM kity
- Přístupné online
  - Nutnost připojení k internetu
  - Bez instalace softwaru
- Pro většinu kitů stejný program, minimální odlišnosti
- Možnost využití již připravených programů, knihoven  
*„Proč dělat něco, co už někdo udělal před námi?“*
- Jazyk C++

# Výčet podporovaných kitů

- <https://os.mbed.com/platforms/>

arm MBED

Mbed OS

Mbed Cloud

Partner Portal

Search...



OS Home

Hardware

Code

Documentation

Questions

Forum

Compiler

kurkape6

Boards

Filter

Mbed Enabled

☐ Mbed Enabled

Mbed OS support

☐ Mbed OS 2

☐ Mbed OS 5.4

☐ Mbed OS 5.5

☐ Mbed OS 5.6

Target vendor

☐ AnalogDevices

☐ ARM

☐ Avnet Silica

☐ Maxim Integrated

☐ Nordic Semiconductor ASA

☐ Nuvoton

☐ NXP Semiconductors

☐ Realtek

☐ Renesas

☐ Silicon Labs

☐ STMicroelectronics

☐ Toshiba

☐ WIZnet

Platform vendor

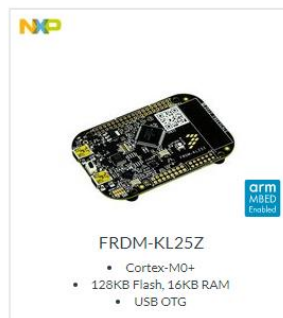
☐ AnalogDevices

☐ ARM

☐ Avnet Silica

☐ BBC Make it Digital Campaign

## Boards



1. 12. 2017

Výuková laboratorní sestava - seminář pro učitele

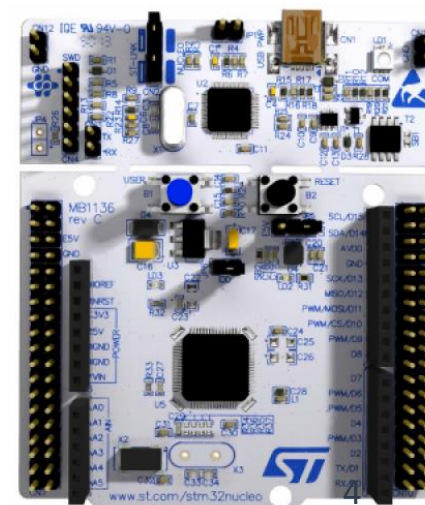
3

# Jednoduchá ukázka STM32F303

```
#include "mbed.h"
```

```
DigitalOut myOut(PA_5); // inicializace vystupu
```

```
int main() {  
    while(1) {  
        myOut = 1;           // nekonečná smyčka  
        wait(0.2);           // zapnutí pinu  
        myOut = 0;           // čekáme 200 ms  
        myOut = 0;           // Vypnutí pinu  
        wait(1.0);           // čekání 1 s  
    }  
}
```

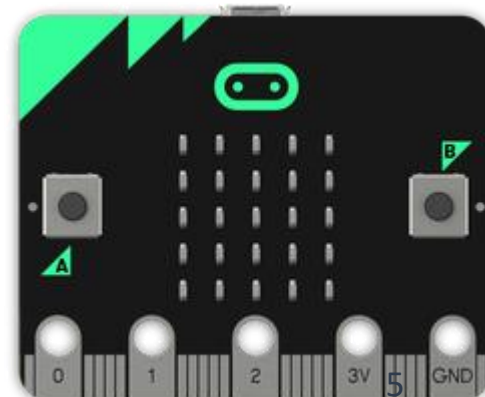


# Jednoduchá ukázka BBC micro:bit

```
#include "mbed.h"
```

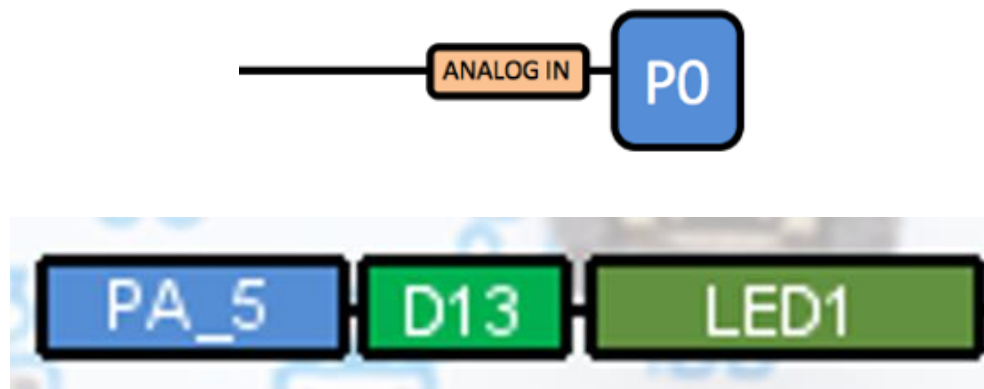
```
DigitalOut myOut(P0);    // inicializace vystupu
```

```
int main() {  
    while(1) {            // nekonečná smyčka  
        myOut = 1;        // zapnutí pinu  
        wait(0.2);        // čekáme 200 ms  
        myOut = 0;        // Vypnutí pinu  
        wait(1.0);        // čekání 1 s  
    }  
}
```

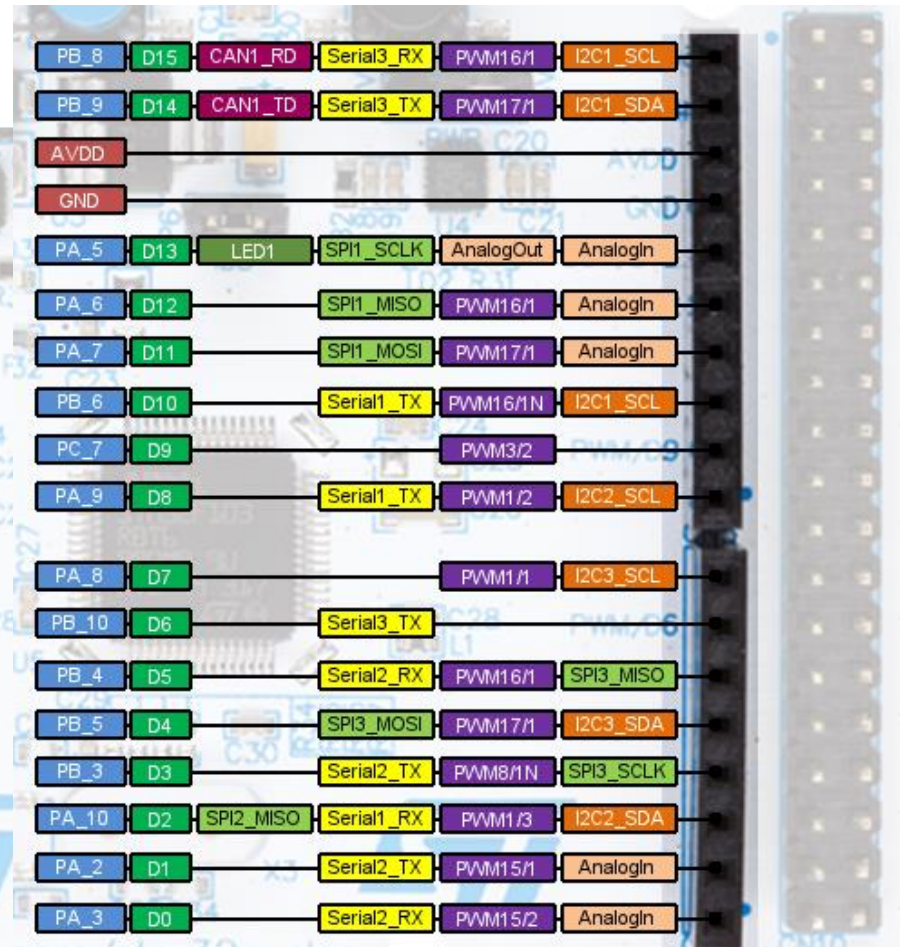
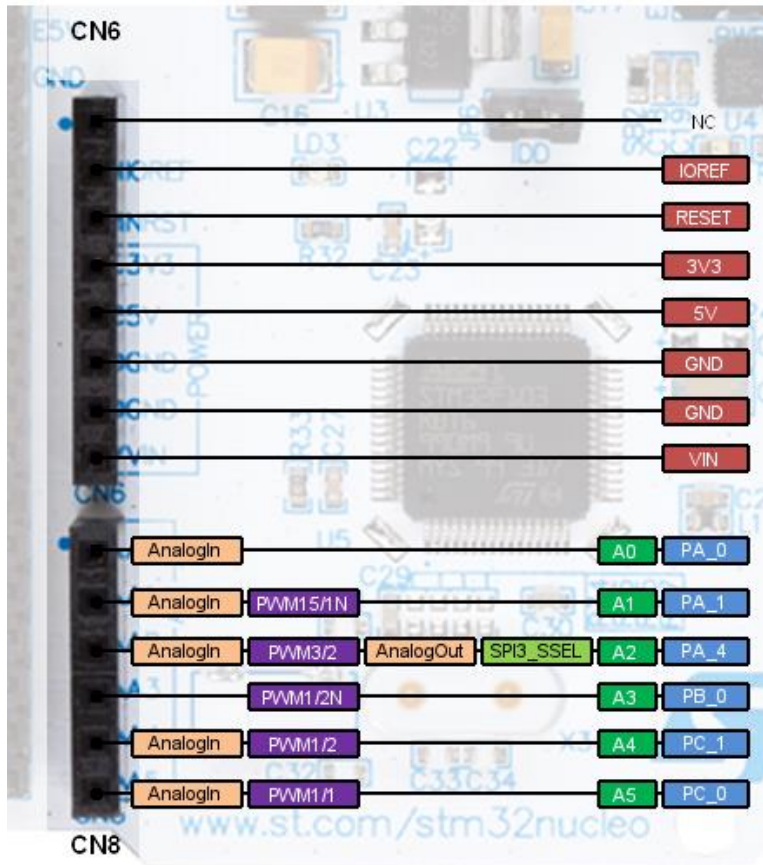


# Odlišnosti u různých kitů

1. `DigitalOut myOut(P0); //BBC`
  2. `DigitalOut myOut(PA_5); //STM32F303`
- Pojmenování pinů -> PINOUT



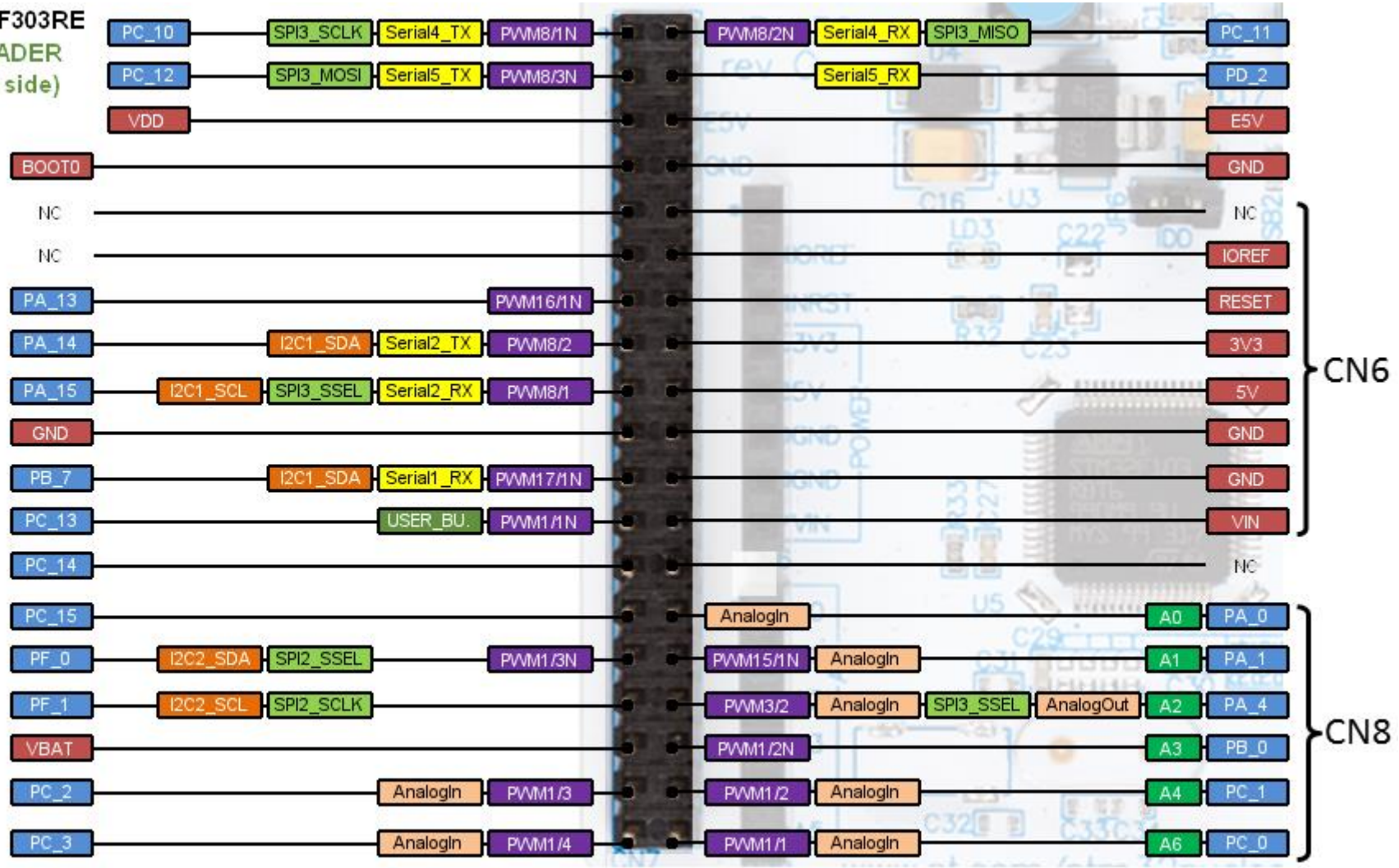
# Pinout STM32F303 Nucleo





# Pinout STM32F303 Nucleo

NUCLEO-F303RE  
CN7 HEADER  
(top left side)





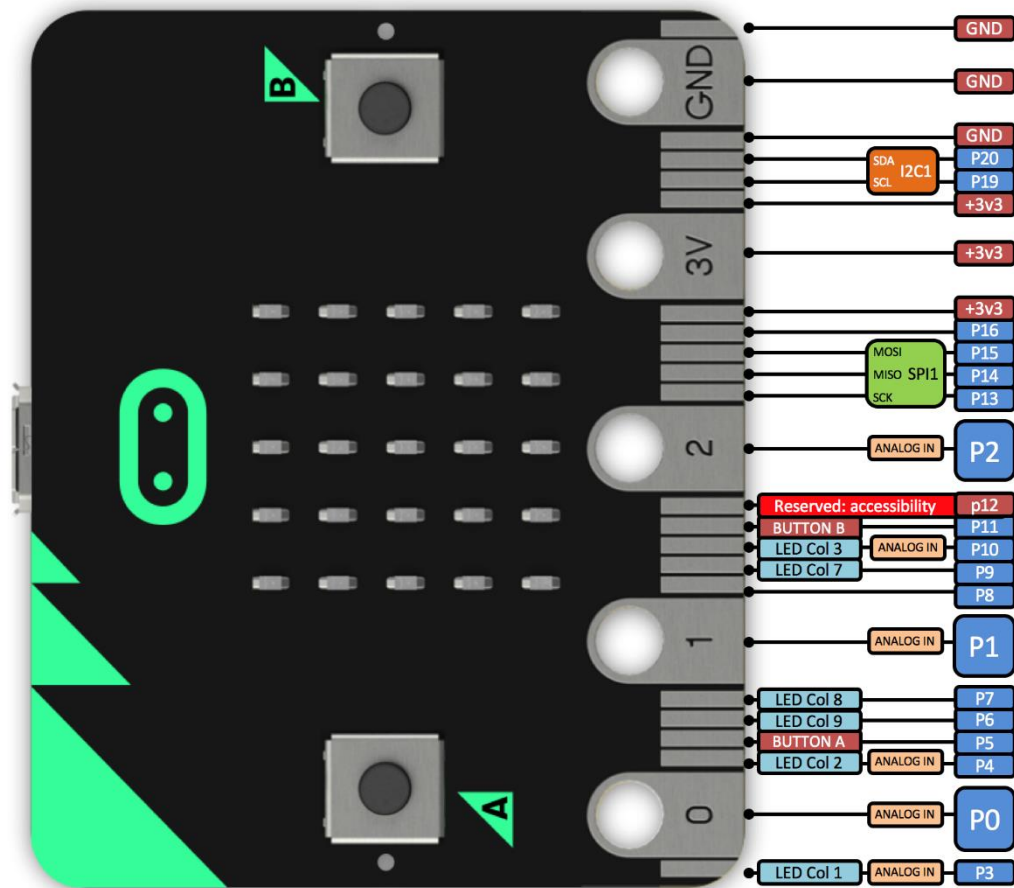
# Pinout STM32F303 Nucleo

NUCLEO-F303RE

CN10 HEADER  
(top right side)



# Pinout BBC micro:bit

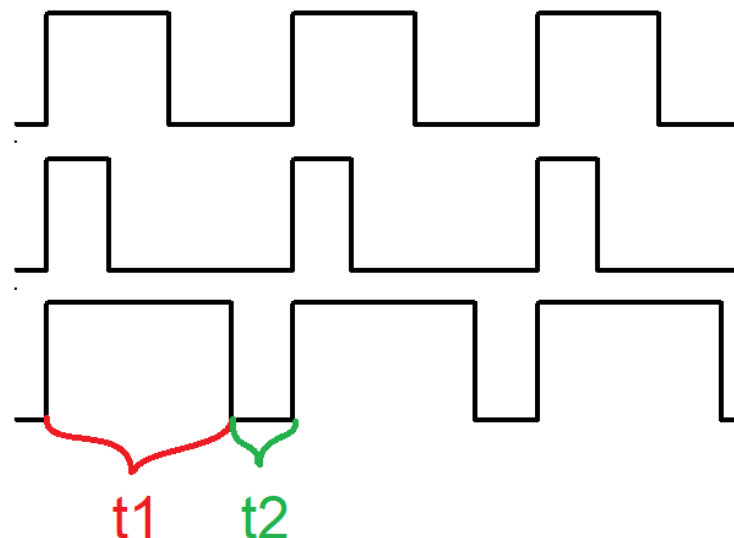


# Funkce kitů - Digital OUT, IN

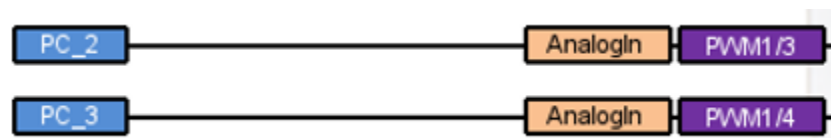
- Výstup
  - Nastavení pinu do stavu (1 nebo 0)
  - Fyzicky nastavení napětí na pinu (0V nebo 3,3V)
    - Některé procesory (0V nebo 5V)
- Vstup
  - Čtení stavu na pinu (0 nebo 1)
- V programu:
  - `DigitalOut myOutput(PA_5) ;`
  - `DigitalIn myInput(PA_5) ;`

# Funkce kitů - PWM

- Generování digitálního signálu
  - Nastavujeme střidu, frekvenci
- Použití:
  - Proměnná intenzita svítící LED
  - Reprodukory
  - Řízení rychlosti motoru
- V programu:
  - **PwmOut** **oznaceni (PC\_5)** ;
- Pin musí funkci podporovat

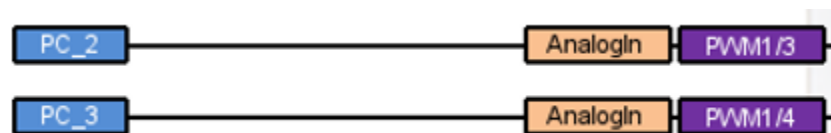


- Střída =  $t1/(t1+t2)$
- Frekvence =  $1/(t1+t2)$



# Funkce kitů – Analog OUT, IN

- Výstup
  - Nastavení libovolného napětí na pinu
  - **0 V až 3,3 V (5 V)** desetinným číslem 0 až 1
- Vstup
  - Čtení stavu na pinu (0 až 1)
- Příklady: Voltmetr, generátor sinusového signálu...
- V programu:
  - **AnalogOut myOutput (PA\_5) ;**
  - **AnalogIn myInput (PA\_5) ;**
- Pin musí funkci podporovat



# Funkce kitů – Sériová komunikace

- Možnost přímé komunikace a obsluhy z počítače
  - Použití aplikace pro terminál
- Standardy: UART, SPI, I2C...
- Dva piny: přijímací (RX), vysílací (TX)
- Pin musí funkci podporovat



```
C:\Users\juhoka\Downloads\psftp.exe
psftp: no hostname specified; use "open host.name" to connect
psftp> open 192.168.91.150
login as: root
root@192.168.91.150's password:
Remote working directory is /root
psftp>
```

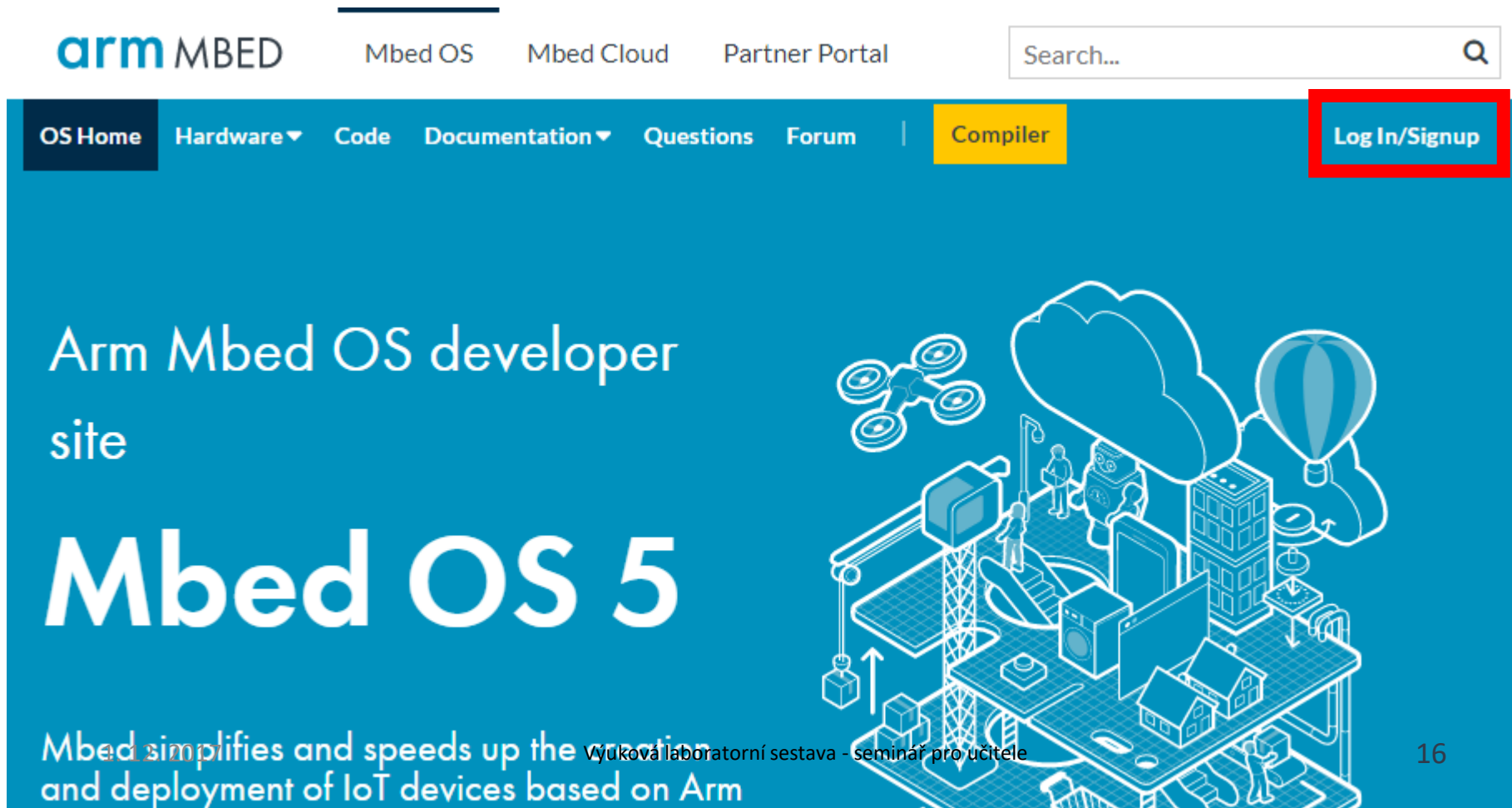


# Funkce prostředí

- Procedurální programování (C++)
  - Rozvedeme v pozdější části prezentace
- Funkce reálného času
  - Wait (čekání)
    - Program čeká po daný čas
  - Timer (stopky)
    - Program měří čas do události
  - Ticker (budík)
    - Program provede akci v daný čas
  - Detailnější popis na [os.mbed.com](https://os.mbed.com)

# Registrace nového uživatele - návod

- <http://www.developer.mbed.org/>



arm MBED Mbed OS Mbed Cloud Partner Portal Search...

OS Home Hardware Code Documentation Questions Forum Compiler Log In/Signup

Arm Mbed OS developer site

# Mbed OS 5

Mbed simplifies and speeds up the creation and deployment of IoT devices based on Arm

Výuková laboratorní sestava - seminář pro učitele

[Login or Signup](#)

## Login

Username:

[I've forgotten my username](#)

Password:

[I've forgotten my password](#)☐ Remember me

Login

## Signup



Signup

Create an os.mbed.com account

## Signup

Enter your email address:

[I already have an account!](#)

Choose a username:

Choose a password:

Confirm your password:


First name:

Last name:

Country:

Are you human? :

☐ Nejsem robot

  
reCAPTCHA  
Ochrana soukromí - Smluvní podmínky

☐ I agree to Arm's [Terms and Conditions of Use](#). (required)

Arm will process your information in accordance with the Account Registration section of our [Privacy Policy](#).

☐ By ticking this box you indicate your consent to receiving marketing communications from Arm in accordance with our Privacy Policy. Please visit our [Subscription Center](#) to manage your marketing preferences or unsubscribe from further communications.

## Summary

*You are about to...*

 Create an Mbed user account

An account will be set up for you, giving you access to the Mbed website and resources.

Arm Mbed OS developer site

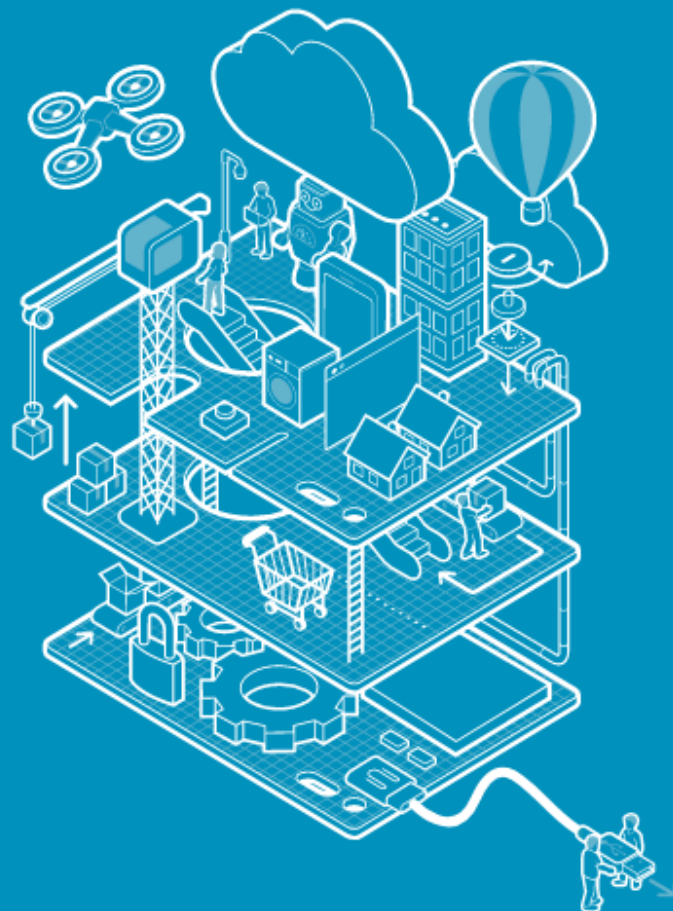
# Mbed OS 5

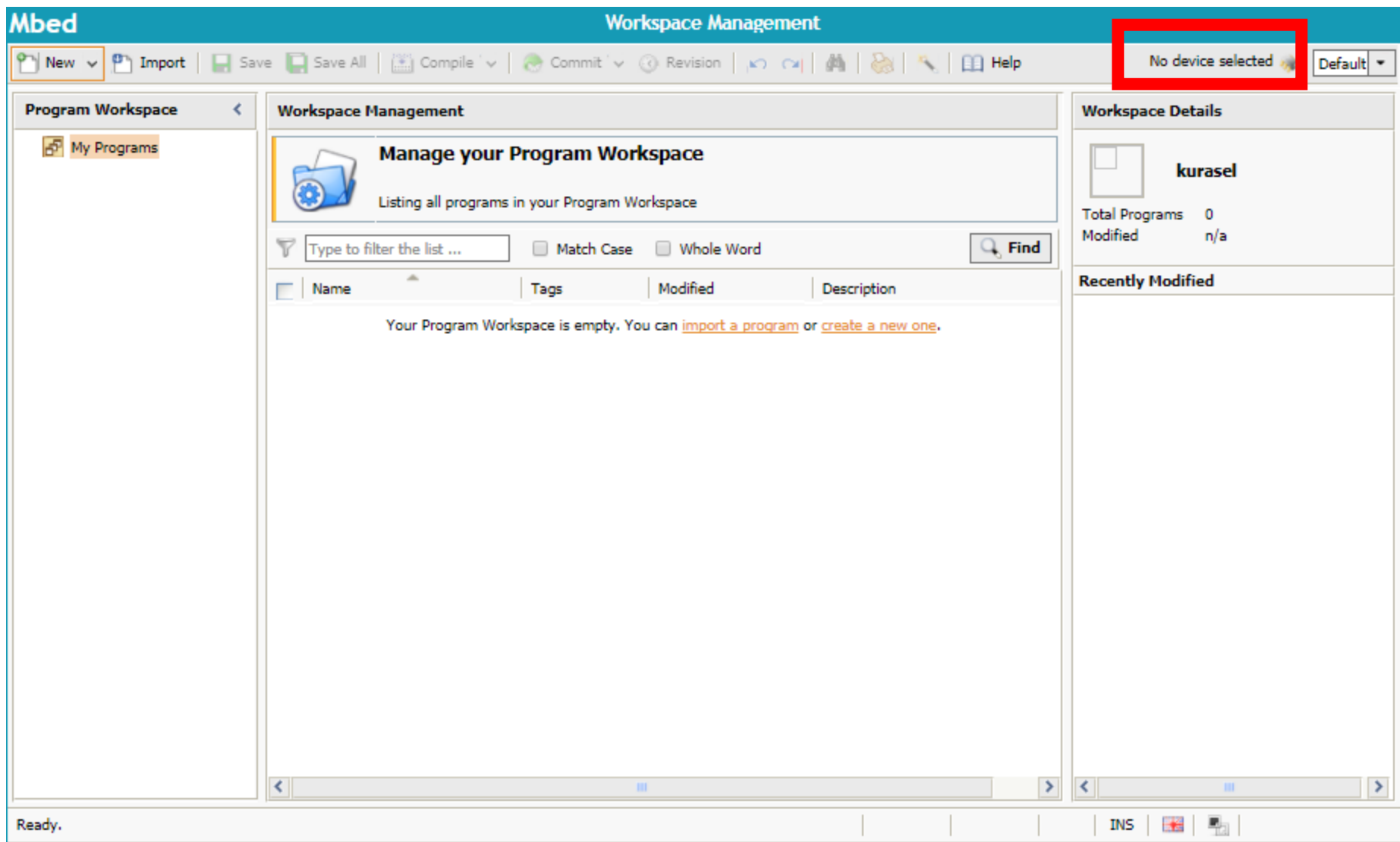
Mbed simplifies and speeds up the creation and deployment of IoT devices based on Arm microcontrollers.

The project is being developed by Arm, its Partners and the contributions of the global Arm Mbed Developer Community.

[Get started »](#)[Latest release »](#)

**326,158** compilations in the last 7 days







Your Program Workspace is empty. You can [import a program](#) or [create a new one](#).

### Select a Platform



**You haven't selected any platform yet**

Pick one from the list below and click the "Select Platform" button to select it.



Select  
Platform

Description

Pinout

N/A

More Info

### Your registered platforms



Add  
Platform

## Boards

## Filter

## Mbed Enabled

- ☐
- Mbed Enabled

## Mbed OS support

- ☐
- Mbed OS 2
- 
- ☐
- Mbed OS 5.4
- 
- ☐
- Mbed OS 5.5
- 
- ☐
- Mbed OS 5.6

## Target vendor

- ☐
- ARM
- 
- ☐
- Avnet Silica
- 
- ☐
- Maxim Integrated
- 
- ☐
- Nordic Semiconductor ASA
- 
- ☐
- Nuvoton
- 
- ☐
- NXP Semiconductors
- 
- ☐
- Realtek
- 
- ☐
- Renesas
- 
- ☐
- Silicon Labs
- 
- ☐
- STMicroelectronics
- 
- ☐
- Toshiba
- 
- ☐
- WIZnet

## Platform vendor

- ☐
- ARM
- 
- ☐
- Avnet Silica
- 
- ☐
- BBC Make it Digital
- 
1. 12. 2017 Campaign
- 
- ☐
- community contributors

## Boards



mbed LPC1768

- Cortex-M3, 96MHz
- 512KB Flash, 32KB RAM



mbed LPC11U24

- Cortex-M0, 48MHz
- 32KB Flash, 8KB RAM



Seeedstudio-Arch

- Cortex-M0, 48MHz
- 32KB Flash, 8KB RAM

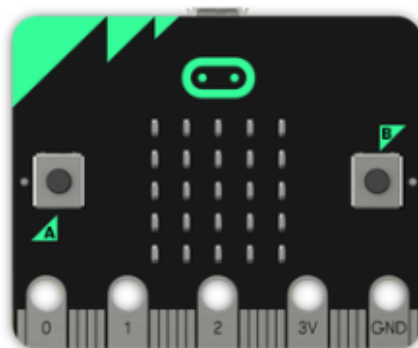


FRDM-KL25Z

- Cortex-M0+
- 128KB Flash, 16KB RAM
- USB OTG

# BBC micro:bit

The BBC micro:bit is a pocket-sized, codable computer that allows anyone to get creative with technology. Made possible through a major partnership with 31 organisations, a micro:bit has been given to every 11 or 12 year old child in year 7 or equivalent across the UK, for free.



The official website for the BBC micro:bit is <http://www.microbit.co.uk>.

If you're keen to try using the micro:bit with mbed, then start with this platform page to add it to your compiler (on the right hand side), then check out the activity on the [micro:bit team page](#).

If you're new to mbed, there's a [video showing your first example here](#)

The BBC micro:bit is based on the mbed HDK. The target MCU is a Nordic nRF51822 with 16K RAM, 256K Flash. As well as the nRF51822 there's also an onboard accelerometer and magnetometer from Freescale.

## Table of Contents

1. Pinout
2. micro:bit Device Abstraction Layer (DAL)
3. Getting Started Video
4. Features
5. mbed Libraries
6. Firmware

To compile a program for this board using Mbed CLI, use `nrf51_microbit` as the target name.

## Board Partner



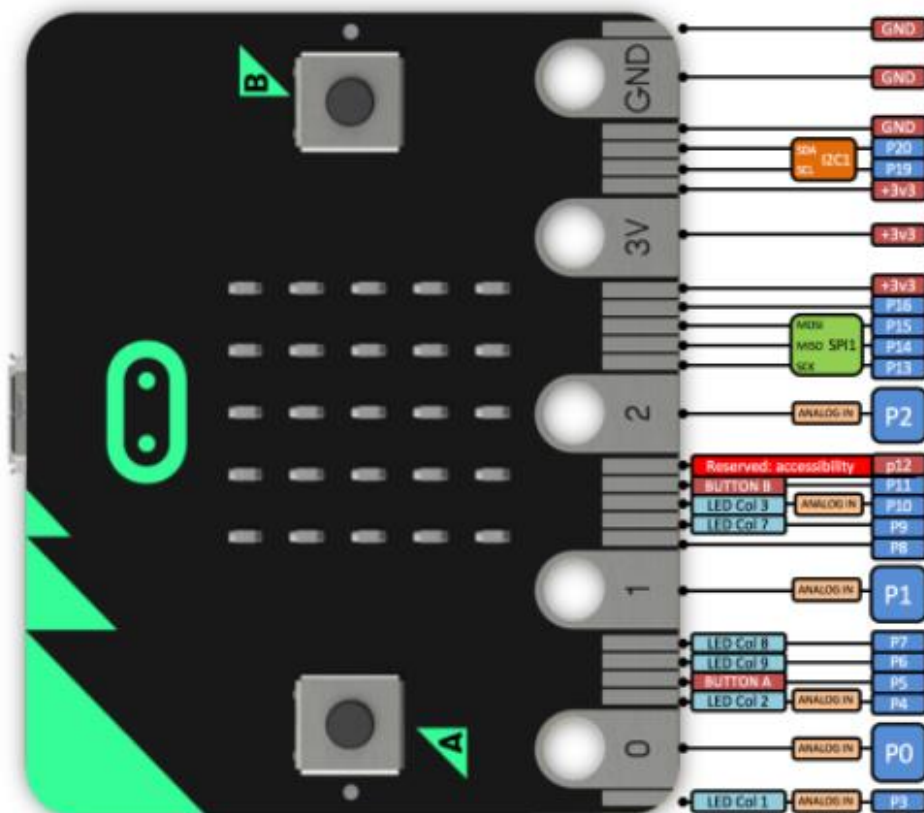
## BBC

The BBC are co-ordinating the micro:bit project

## Silicon Partner



# Pinout



7. micro:bit availability

## Nordic Semiconductor

Nordic Semiconductor is a fabless semiconductor company specializing in ultra low-power wireless SoCs and connectivity devices for the 2.4 GHz ISM band, with ultra-low power performance and cost being the main focus areas

[+ Add to your Mbed Compiler](#)

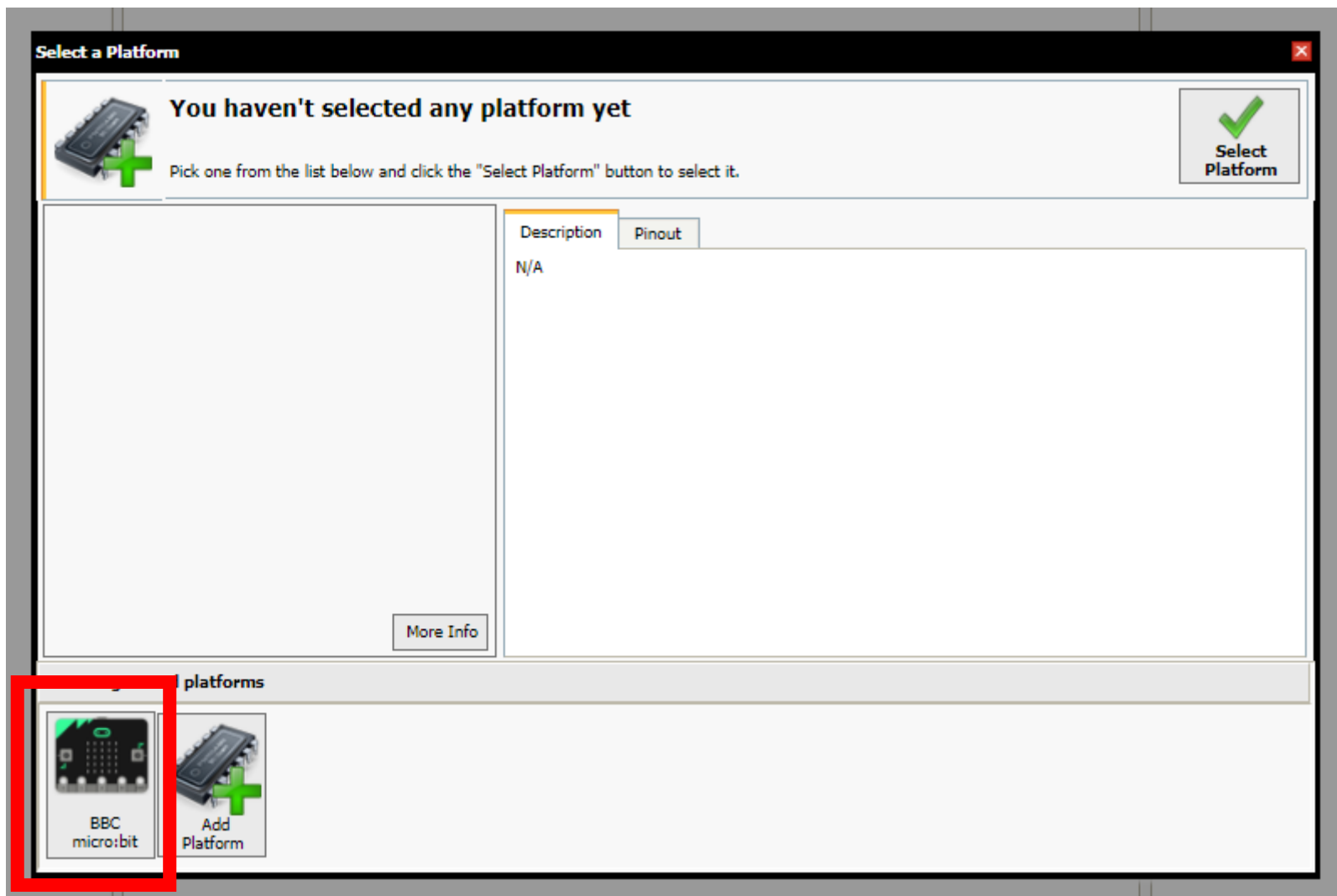
[Buy Now](#)

[+ Follow](#)



Mbed Enabled  
Mbed OS 2

## micro:bit Device Abstraction Layer (DAL)



# Práce s kompilátorem

Mbed /303\_Ticker/main.cpp

New Import Save Save All Compile Commit Revision Help

Volba kitu NUCLEO-F303RE

Program Workspace

- My Programs
  - 303\_Blink
    - blik.cpp
    - Ticker303.cpp
  - 303\_Ticker
    - main.cpp
    - mbed
- Classes
  - AnalogueIn

Projekty

- BusOut
- CANMessage
- CAN
- CThunk
- Callback
- CircularBuffer
- CriticalSectionLock
- DeepSleepLock
- DigitalInOut
- DigitalIn
- DigitalOut
- DirHandle
- Ethernet
- FileHandle
- FileSystemHandle
- FileSystemLike
- FlashIAP
- I2CSlave
- I2C

Hlavní okno – kód programu

```
1 // Toggle the blinking led after 5 seconds
2
3 #include "mbed.h"
4
5 Ticker timer;
6 DigitalOut led1(PB_0);
7 DigitalOut led2(PC_0);
8
9 int flip = 0;
10
11 void attime() {
12     flip = !flip;
13 }
14
15 int main() {
16     timer.attach(&attime, 5);
17     while(1) {
18         if(flip == 0) {
19             led1 = !led1;
20         } else {
```

Compile output for program: 303\_Ticker

Verbose Errors: 0 Warnings: 0 Infos: 1

Description	Error Number	Resource	In Folder	Location
Success!		<a href="#">Build Details</a>		

Výpisy událostí

Compile Output Find Results Notifications

Ready 12. 2017

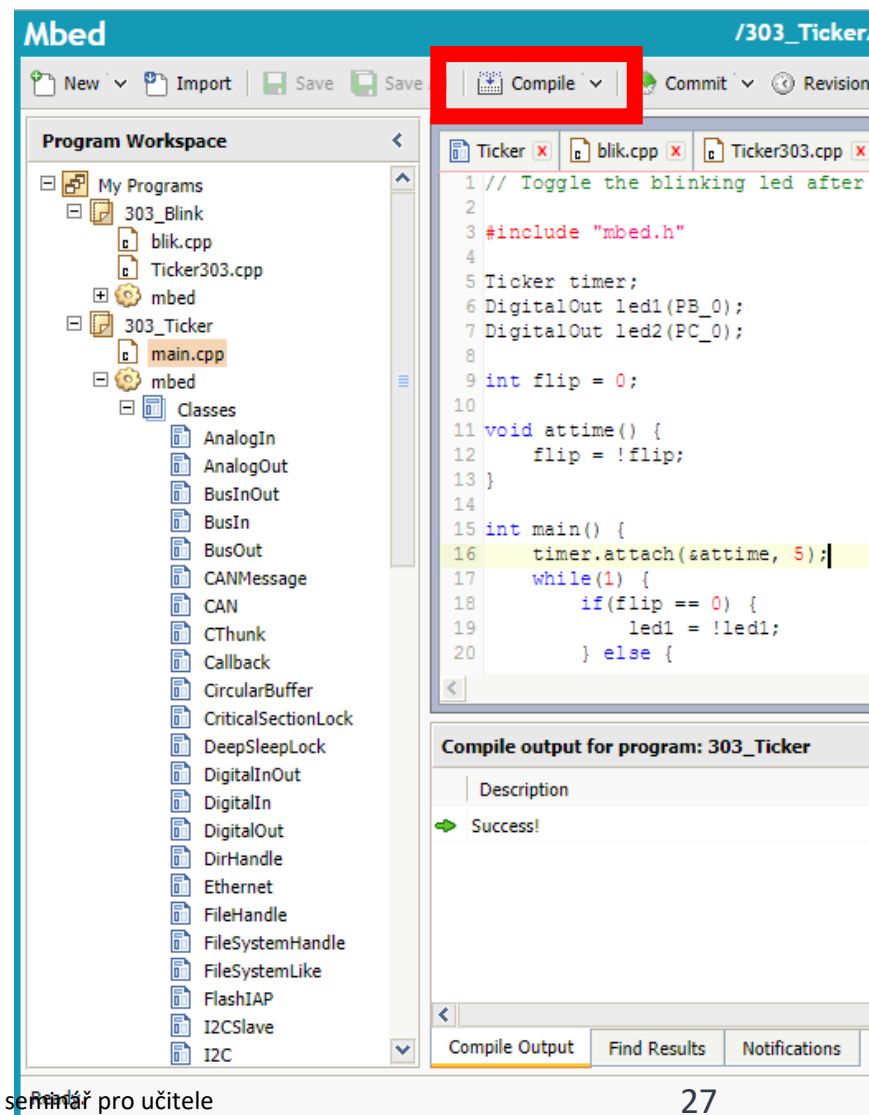
Výuková laboratorní sestava - seminář pro učitele col 30 25 INS

26



# Práce s kompilátorem

- Hotový program:
  1. Kompilovat
  2. Soubor se přeloží do .bin a stáhne se
  3. Přes USB kopírovat do kitu jako na flash disk
  4. Program se sám nahraje
  5. Program běží 😊



# Základy programování

- Vykonávání programu
  - Od začátku funkce **main()** { }
  - Deklarace proměnných před main – globální, v blocích – lokální
- Proměnné
  - Celá čísla
    - Inicializace: **int x;**
    - Zápis: **x = 15;**
  - Desetinná – **float** nebo **double**

# Základy programování

- Smyčky
  - **while (podmínka) {kód; }**
    - kód se vykonává pokud je splněna podmínka
  - **for (vstupní\_příkaz; podmínka; příkaz\_na\_konci\_každé\_smyčky) {kód; }**
    - Používána pro cyklus o daném počtu opakování
- Podmínky
  - **if (podmínka1) {kód při splnění1;} else if (podmínka2) {kód při splnění2;} else {kód při nesplnění;}**
- Čekací funkce
  - **wait(čas v sek.)**
  - udání času desetinným číslem

# Digitální výstupy, vstupy

- Výstupy
  - Třída DigitalOut
  - Inicializace:
    - `DigitalOut oznaceni(nazev_pinu, vychazi_stav);`
  - Ovládání:
    - `oznaceni = 1;` nebo `oznaceni = 0;`
  - Čtení výstupu:
    - `promenna = oznaceni;`

# Digitální výstupy, vstupy

- Vstupy
  - Třída DigitalIn
  - Inicializace:
    - **DigitalIn oznaceni (nazev\_pinu, (mód) ) ;**
    - mód – volitelný PullUp, PullDown, PullNone, OpenDrain
  - Čtení:
    - **Promenna = oznaceni ;**
  - Tlačítka aktivní v 0!!!

# Příklad 1. – ovládání Digitálního výstupu (pinu) STM32F303 Nucleo

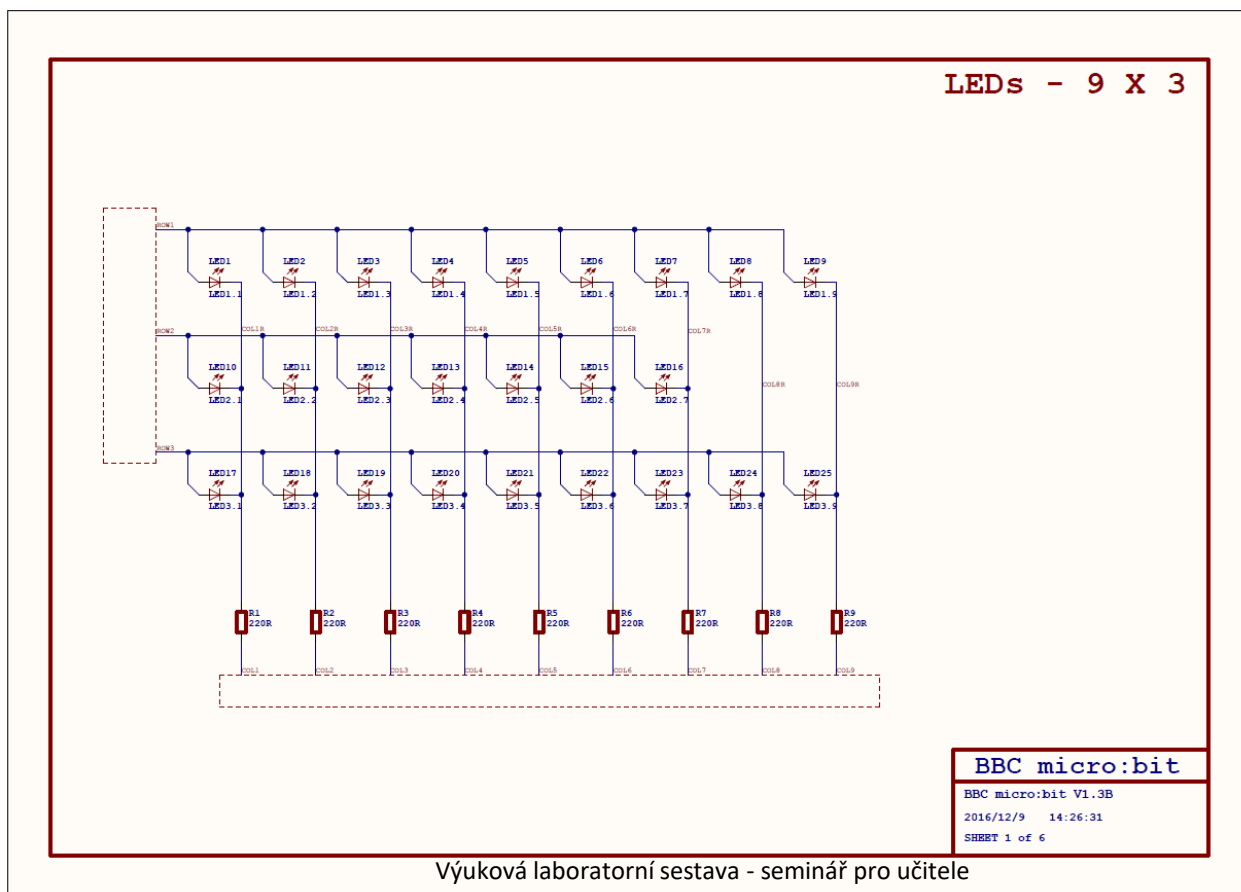
- Inicializace výstupu:
  - `DigitalOut myOut(PA_5);`
  - Na Pinu PA\_5 je shodou okolností LED dioda
    - stav pinu tedy vidíme díky diodě přímo na kitu
- V nekonečné smyčce:

```
myOut = 1; // aktivuje pin
wait(0.2); // čekání, pin aktivní
myOut = 0; // deaktivace pinu
wait(1.0); // čekání
```



# Příklad 2. – blikání BBC micro:bit

- Inicializace 2 výstupů
  - Matice 5x5 LED je řízena aktivací řádku a sloupce



## Příklad 2. – blikání BBC micro:bit

```
DigitalOut slupec0(P0_4, 1);
DigitalOut radek0(P0_13, 1);

int main() {
    while(1) {
        slupec0 = 0;           //Rozsviceni
LED
        wait(0.2);            //200 ms
        slupec0 = 1;          //zhasnuti LED
        wait(0.5);            //500ms
    }
}
```

# Příklad 3. – Digitální vstup STM32F303 Nucleo

- Inicializace:

- `DigitalIn mybutton(PC_13) ;`

- PC\_13 odpovídá user tlačítku



- Stisknuté tlačítko odpovídá: `mybutton = 0`

- Program:

```
while(1) {  
    if (mybutton == 0) {  
        myled = !myled;  
        wait(0.2);  
    }  
}
```

# Analogové výstupy

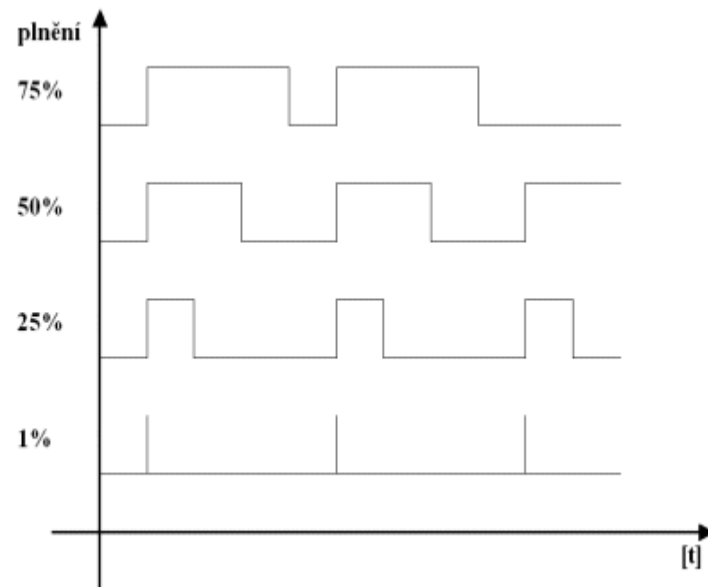
- Analogový výstup
  - Inicializace:
    - **AnalogOut oznaceni (nazev\_pinu) ;**
  - Ovládání:
    - **oznaceni = 0,15;**
      - Desetinné číslo 0 – 1
- Ukázkový příklad 4\_analog.cpp (triviální voltmetr pro BBC micro:bit)

# Analogové vstupy

- Analogový vstup
  - Inicializace analogového vstupu:
    - **AnalogIn oznaceni (nazev\_pinu) ;**
  - Čtení analogového vstupu:
    - **oznaceni.read() ;**
      - načte desetinnou hodnotu 0-1
    - **oznaceni.read\_u16() ;**
      - načte celé číslo 0 – 65535

# PWM – pulsně šířková modulace

- Forma digitálního výstupu
  - Místo `DigitalOut` je `PwmOut`
- Nastavení periody a střídý
- Inicializace:
  - `PwmOut oznaceni (pin) ;`
- Nastavení periody:
  - `oznaceni.period_us (period) ;`
    - Perioda v  $\mu\text{s}$
- Nastavení střídý:
  - `pwm.write (float_strida) ;`
    - Desetinné číslo 0 - 1



# Sériová linka - UART (Ukázkový program 0\_UART\_demo.cpp)

- Inicializace sériové linky:
  - **Serial pc(USBTX,USBRX,9600) ;** – nastav sériovou linku k PC 9600baud
- Zápis dat:
  - **printf("format",promenne)** – zápis zprávy
- Čtení dat
  - **scanf("format",&promenne)** – načtení zprávy o definovaném formátu
- Formát udáván formátovým řetězcem
  - %f desetinné číslo
  - %d celé číslo
  - %s obecný řetězec znaků

# Závěr - Motivační příklad pro STM32F303

- Napište program, kde LED(PA\_5) svítí pouze tehdy, když je zmáčknuté tlačítko(PC\_13)