



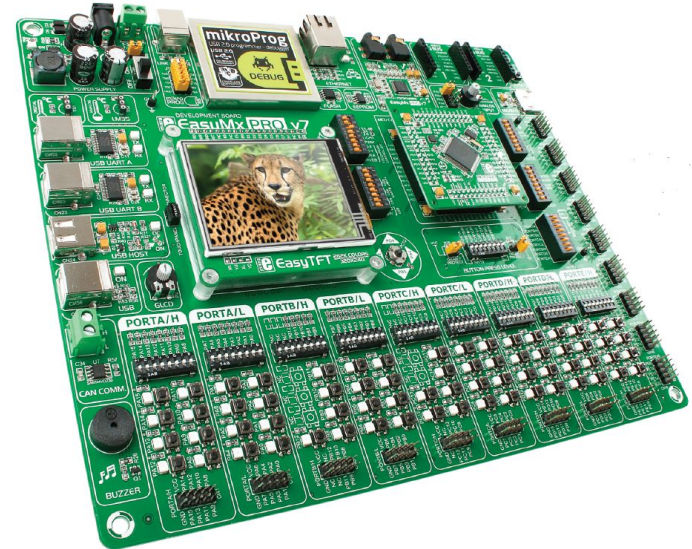
# EasyMx PRO v7 for STM32 ARM

Prepared by:  
Omar Samir

# Introduction

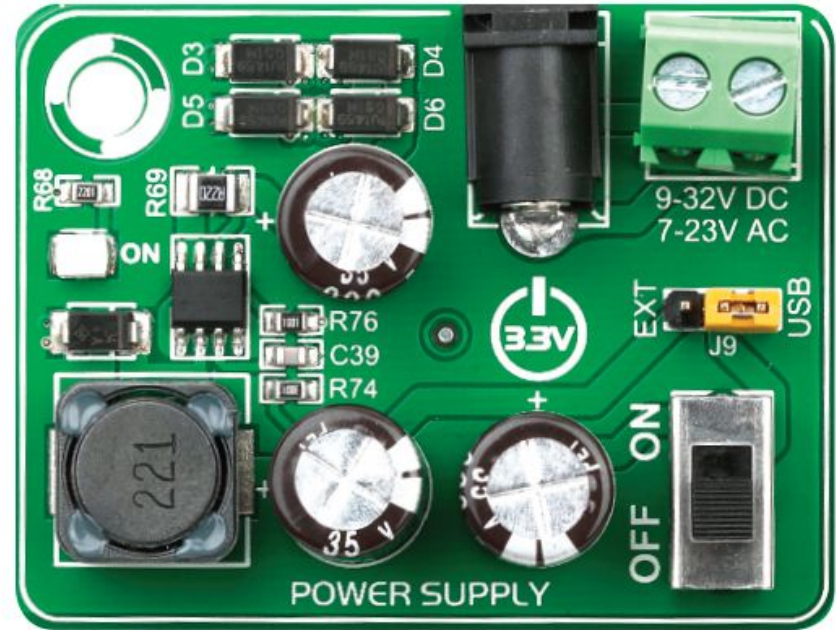
---

- ARM® Cortex™-M3 and Cortex™-M4 are increasingly popular microcontrollers
- A development board with as many peripherals on the board as possible
- Covers many internal modules
- On-board programmer and debugger



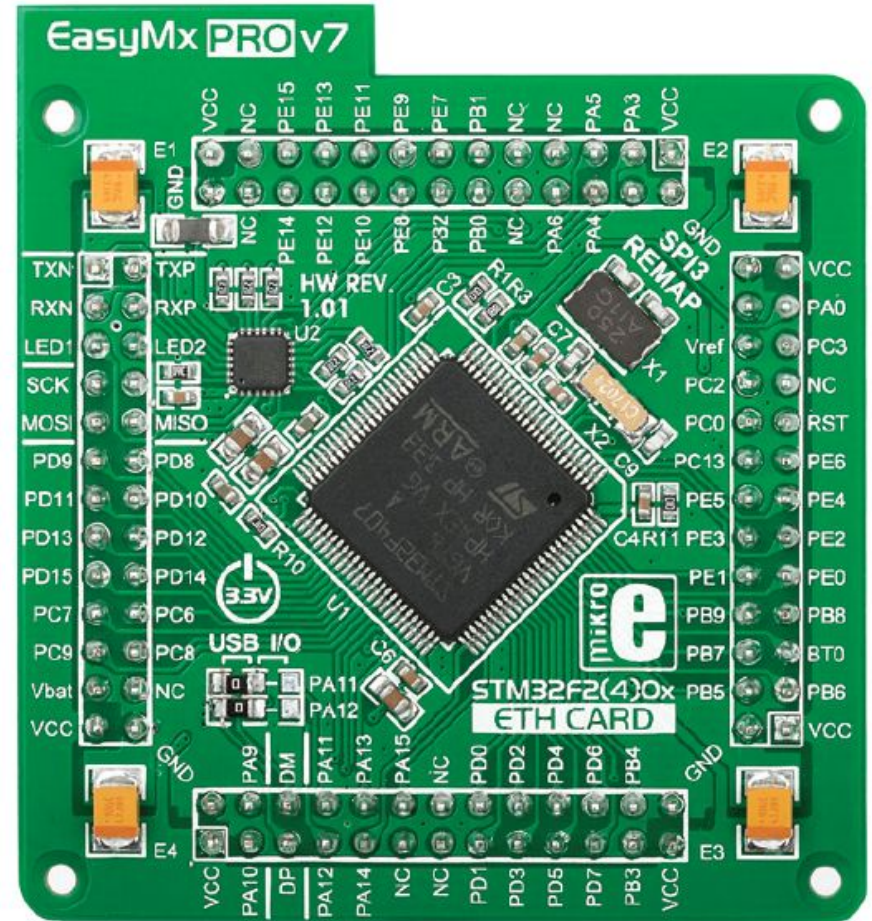
# Power Supply

Creates VCC 3.3 power supply



# MCU Card

- STM32F407VGT6
- 16 MHz default CPU clock
- 5 PORTS (A, B, C, D, E)



# MikroProg

- Fast programmer and debugger
- Enabled by placing the jumpers on the left side
- Takes the following pins for programming
  - PA13
  - PA14
  - PA15
  - PB3
  - PB4



# MikroC pro for ARM



- A compiler for ARM
- Produces hex file to be programmed on MCU
- Can be used for debugging
- Download:
  - <https://download.mikroe.com/setups/compilers/mikroc/arm/mikroc-arm-setup-v620.zip>

# Program our board



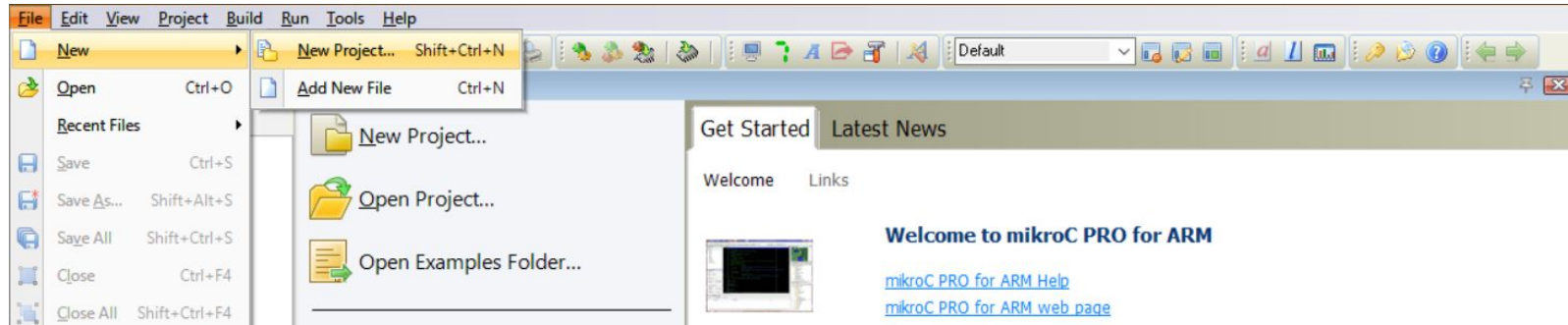
To program our board you need to do the following:

- Download and install mikroProg for STM32 drivers.
  - <https://download.mikroe.com/setups/drivers/mikroprog/arm/st-link-usb-drivers.rar>
- Download and install MikroC pro for ARM
  - While installing it will ask you to install MikroProg accept and install it as it is mandatory for programming
- Connect the board to your computer with the USB cable
- Power on the board

Having done these steps, your board is ready to be programmed!

# Start a new project in MikroC

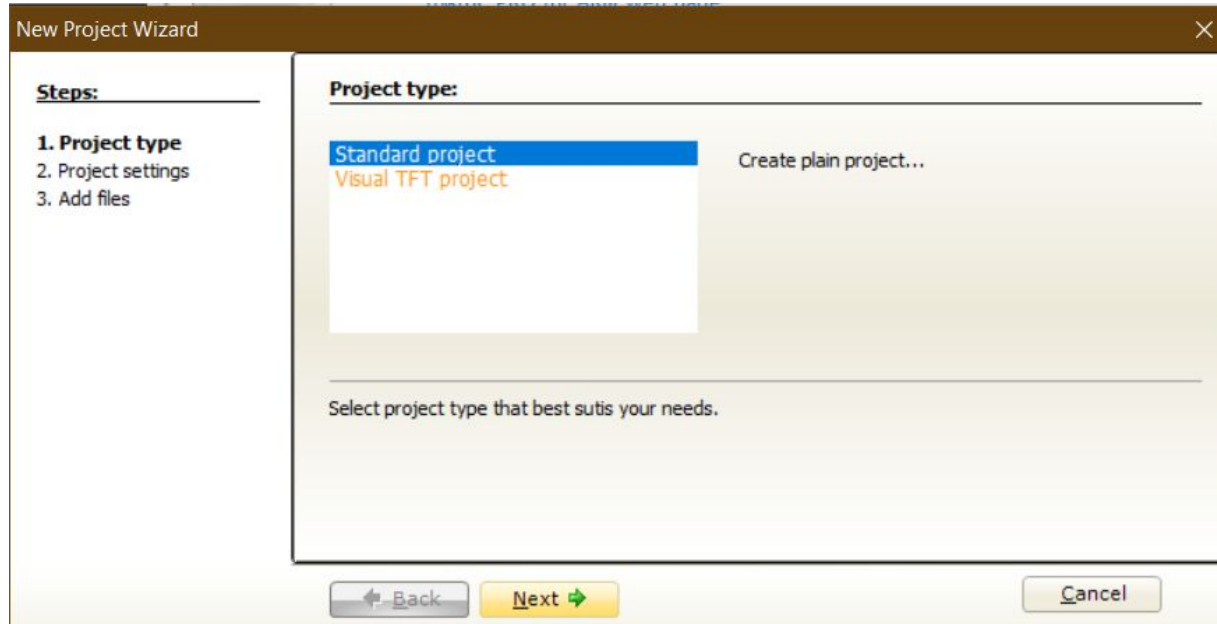
1. Open MikroC pro for ARM
2. Select File -> new -> new project





# Start a new project in MikroC

3. Choose standard project then click next



# Start a new project in MikroC

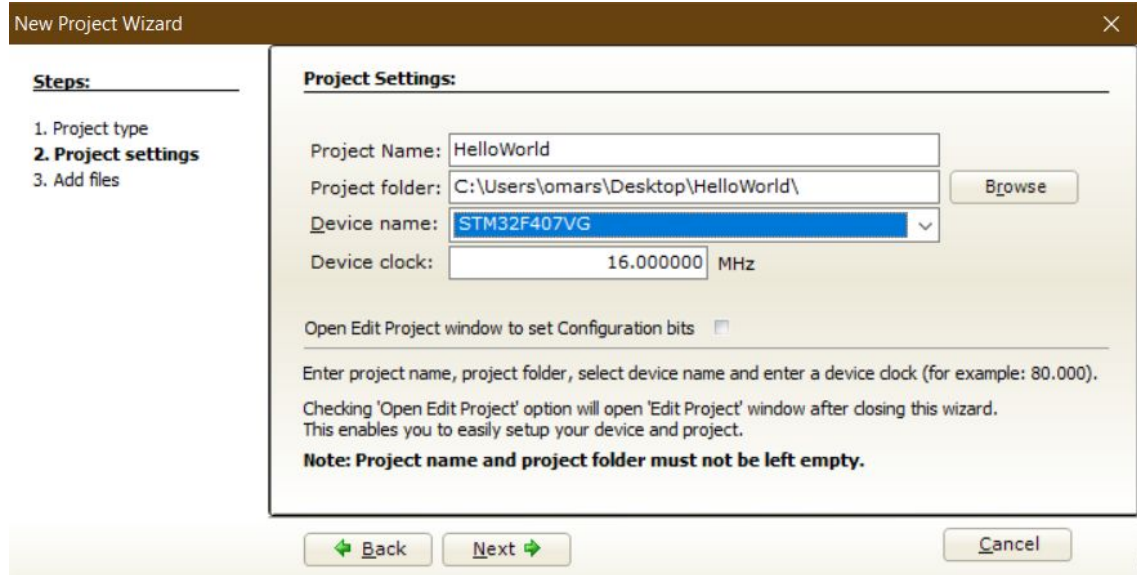
4. Type your project name in (Project Name)

5. Choose the location of the project in (Project folder)

6. Choose STM32F407VG in (Device Name)

7. Choose 16 MHz as our device clock frequency

8. Click Next



The screenshot shows the 'New Project Wizard' dialog box with a brown title bar and a close button. On the left, a 'Steps' list shows '1. Project type', '2. Project settings' (which is bolded and underlined), and '3. Add files'. The main area is titled 'Project Settings:' and contains several fields: 'Project Name' with the text 'HelloWorld', 'Project folder' with the path 'C:\Users\omars\Desktop\HelloWorld\' and a 'Browse' button, 'Device name' with a dropdown menu showing 'STM32F407VG', and 'Device clock' with a text box containing '16.000000' and a unit dropdown set to 'MHz'. Below these fields is a checkbox for 'Open Edit Project window to set Configuration bits' which is currently unchecked. A paragraph of text explains that the project name, folder, device name, and clock frequency must be entered. Another paragraph states that checking the 'Open Edit Project' option will open the 'Edit Project' window after the wizard closes. A bolded note at the bottom states: 'Note: Project name and project folder must not be left empty.' At the bottom of the dialog are three buttons: 'Back' with a left arrow, 'Next' with a right arrow, and 'Cancel'.

New Project Wizard

**Steps:**

1. Project type
- 2. Project settings**
3. Add files

**Project Settings:**

Project Name: HelloWorld

Project folder: C:\Users\omars\Desktop\HelloWorld\ Browse

Device name: STM32F407VG

Device clock: 16.000000 MHz

Open Edit Project window to set Configuration bits ☐

Enter project name, project folder, select device name and enter a device clock (for example: 80.000).

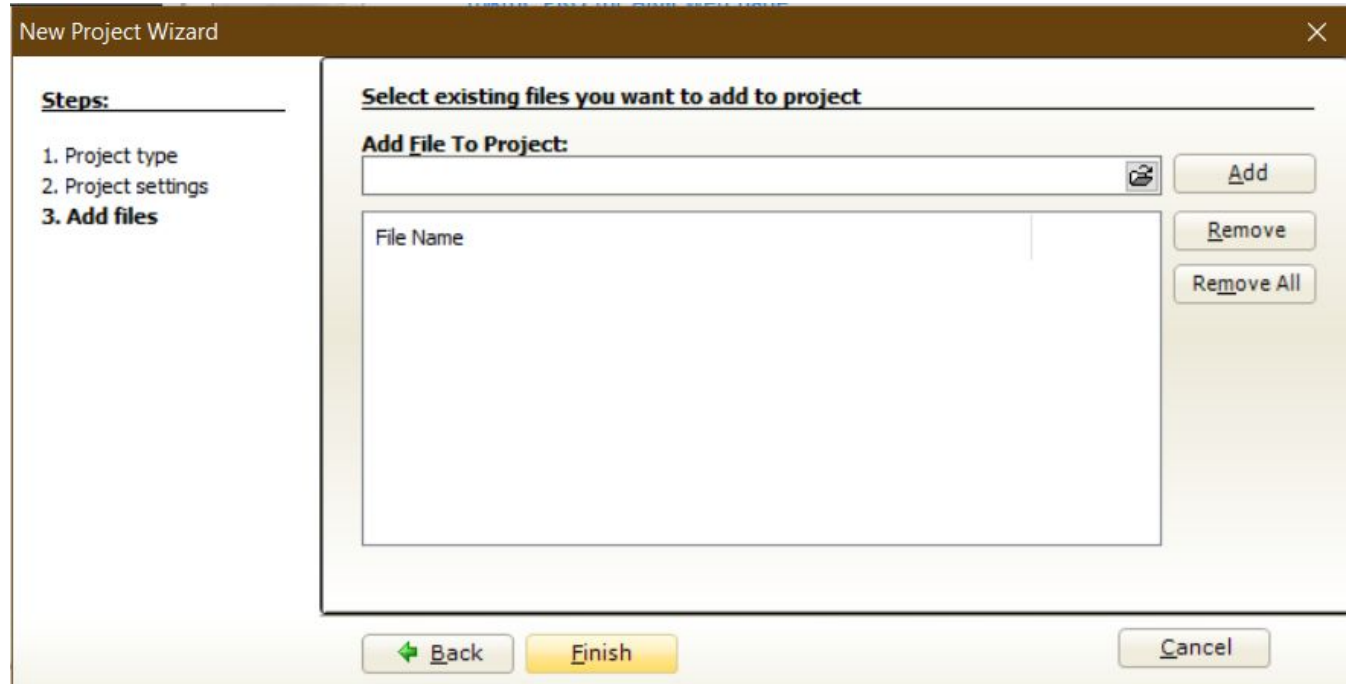
Checking 'Open Edit Project' option will open 'Edit Project' window after closing this wizard. This enables you to easily setup your device and project.

**Note: Project name and project folder must not be left empty.**

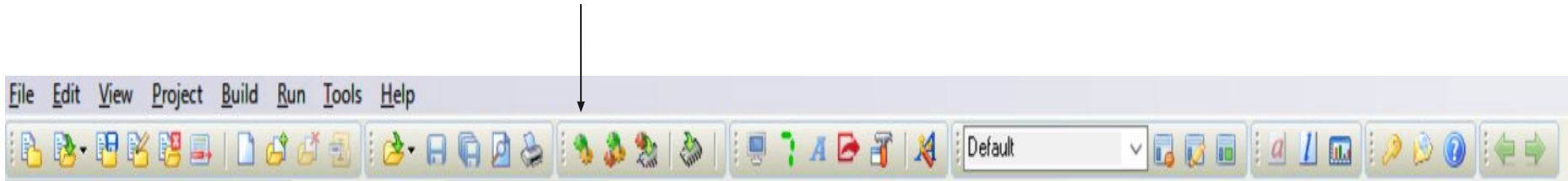
Back Next Cancel

# Start a new project in MikroC

## 9. Click Finish







# Build and Program

You need to be connected to the board and to power it on

