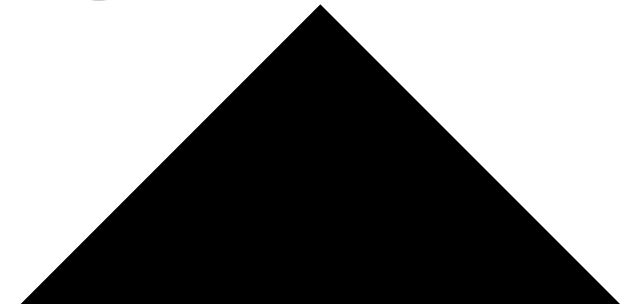


**Imp. Note.**

This is good to start but you need to update the firmware.
Look at page 11 for verifying compatibility of WIFI101 library and firmware.
If the check fails please open the pdf file "Arduino - FirmwareUpdater.pdf" file and read the highlights on the first 5 pages.



- Profile (<https://id.arduino.cc/>)
- Sign Out

Arduino MKR1000 Getting Started © LGPL

(<http://opensource.org/licenses/lgpl-license>)

An unofficial tutorial for getting started with the MKR1000 board as there are no official one yet.

Motivation

When i received the new Arduino MKR1000, i was surprised that there is no official getting started, tutorial, or support in the IDE. I decided to write a short getting started guide to avoid others to spend a lot of time searching peaces of information across internet :)

Moreover, when you google it nowadays, you find more links to the contest organized on h (https://www.hackster.io/challenges/arduino-microsoft-maker) than any other website providing a getting started to setup en environment and start enjoining the MKR1000

Arduino MKR1000

The MKR1000 is described in the official web site (https://www.arduino.cc/en/Main/ArduinoMKR1000) as a powerful board that combines the functionality of the Zero (https://www.arduino.cc/en/Main/ArduinoBoardZero) and the Wi-Fi Shield (https://www.arduino.cc/en/Main/ArduinoWiFiShield101).

The main information regarding the technical specifications for getting started are :

- Micro-controller - **SAMD21** Cortex-M0+ 32bit low power ARM MCU
- Board Power Supply - (USB/VIN) **5V**
- Supported Battery - Li-Po single cell, **3.7V, 700mAh minimum**
- Operating Voltage - **3.3V**

Setup the IDE

Get the latest Arduino IDE from here (https://www.arduino.cc/download.php?f=/arduino-nightly-windows.zip). We use in this tutorial the version 1.6.8 (nightly build) under windows 10. More details about the installation on other operating systems or versions are available this guides (https://www.arduino.cc/en/Guide/HomePage).



- Profile (https://id.arduino.cc/)
- Sign Out

Run the setup once installed and make sure that you install the drivers as illustrated in the Figure 1.

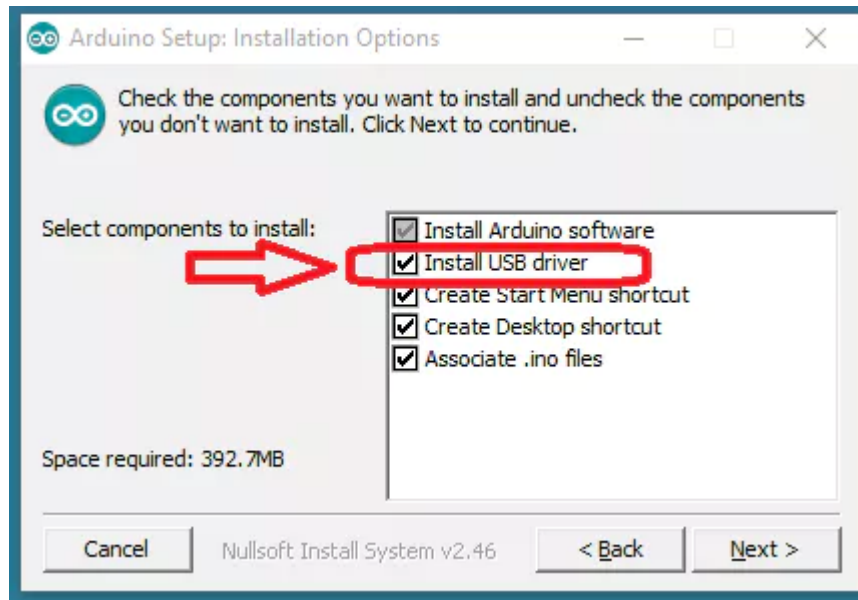
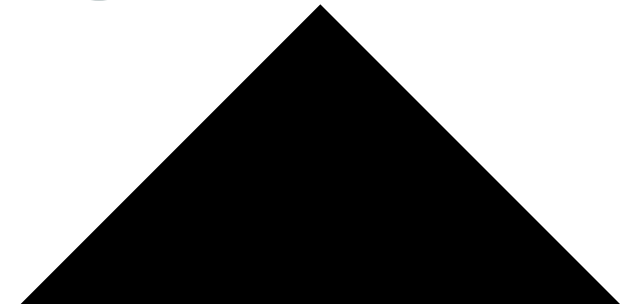


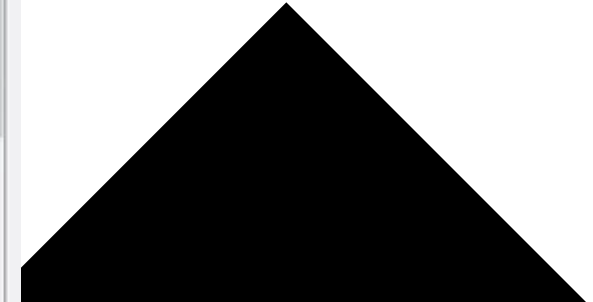
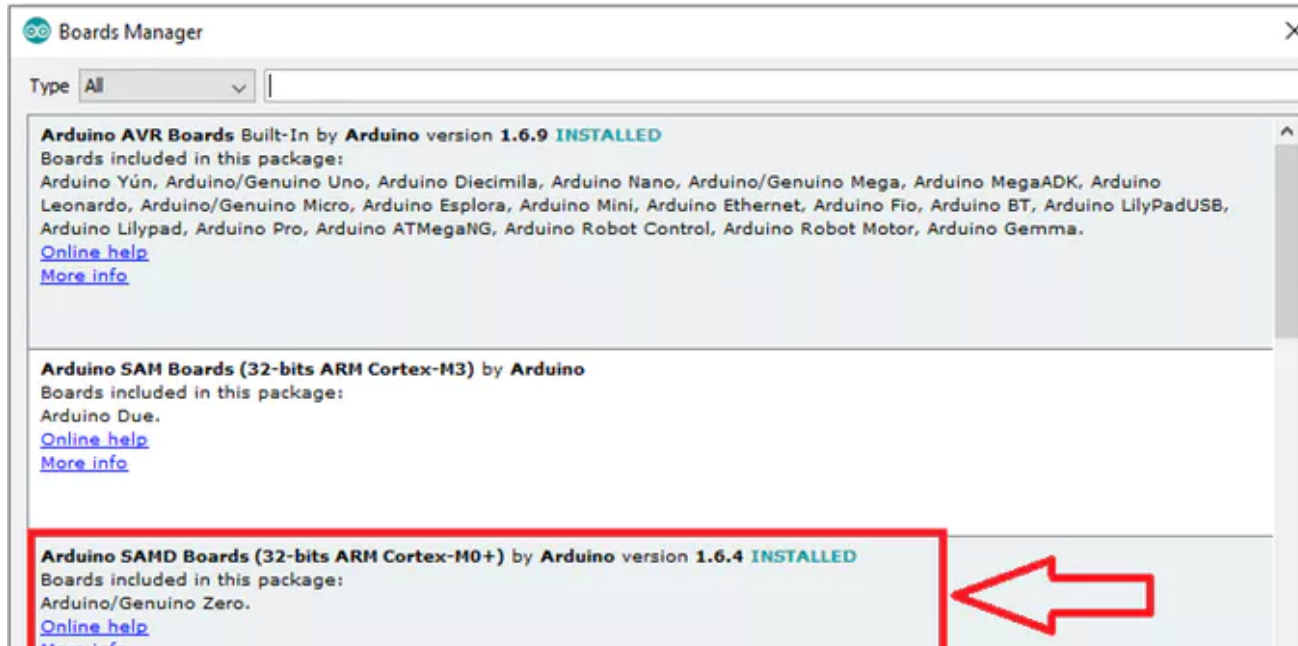
Figure 1



- Profile (<https://id.arduino.cc/>)
- Sign Out

Once installed, run the IDE and go to the menu under **Tools > Board**. You will be surprised as there is no board called MK1000. Don't worry, you can avoid that.

Under **Tools > Board**, go to the **Boards Manager**. You have to install the support for the **Arduino SAMD Boards (32bits ARM Cortex-M0+)** as illustrated in Figure 2. There is a button install that appears when you click on that board.



- Profile (<https://id.arduino.cc/>)
- Sign Out

When you go back to **Tools > Board**, You can now see that the Zero board and the MKR1000 was added to the available boards as illustrated in Figure 3.

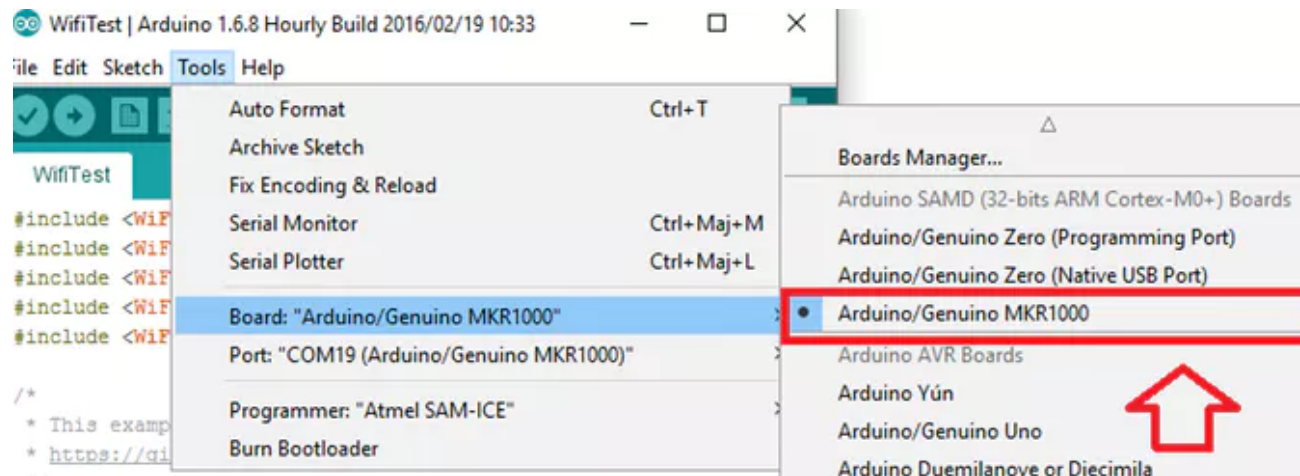
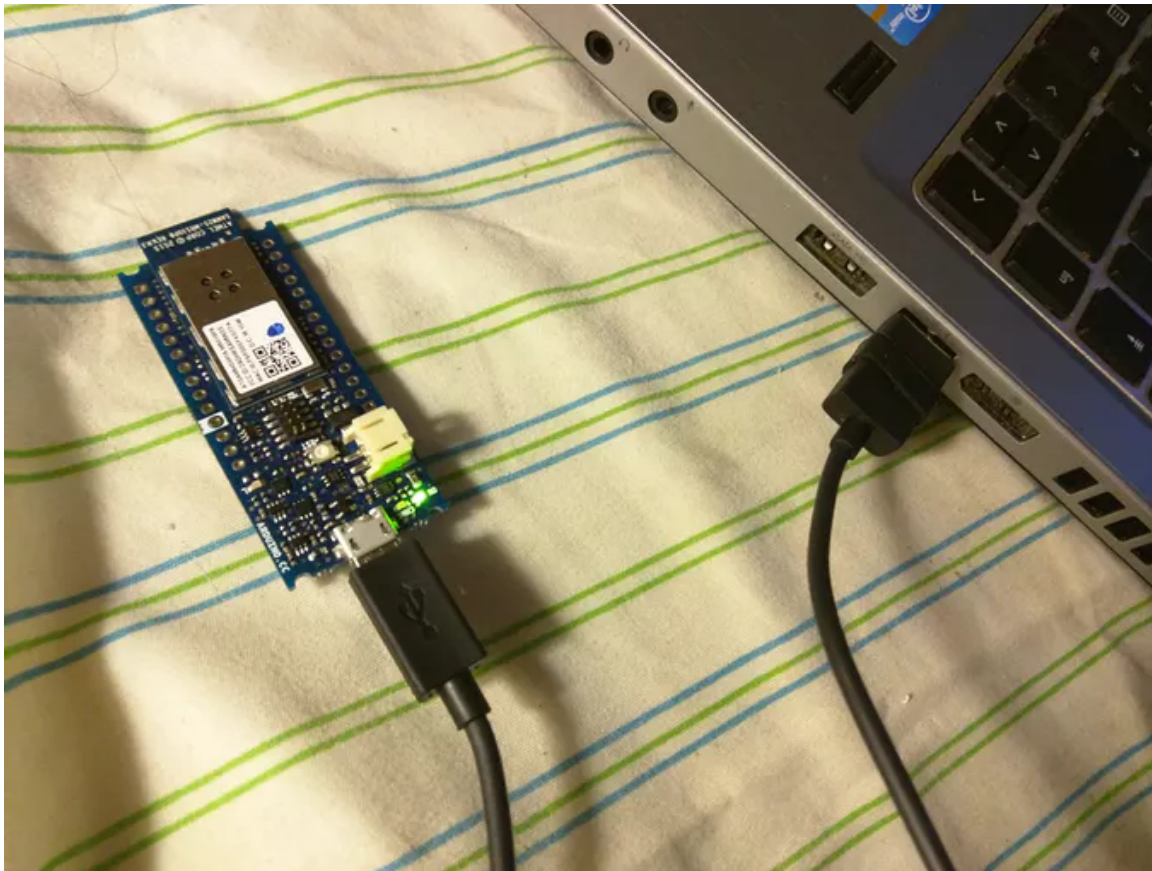


Figure 3

Now is time to connect your board and test the driver setup.

*Figure 4*

- Profile (<https://id.arduino.cc/>)
- Sign Out

Plug in the MKR1000 to your computer as illustrated in Figure 4, the driver should be installed automatically. To check if the driver is installed correctly and the board is ready. Run the device manager and you should see the device available under "COM PORTS" as illustrated in Figure 5.

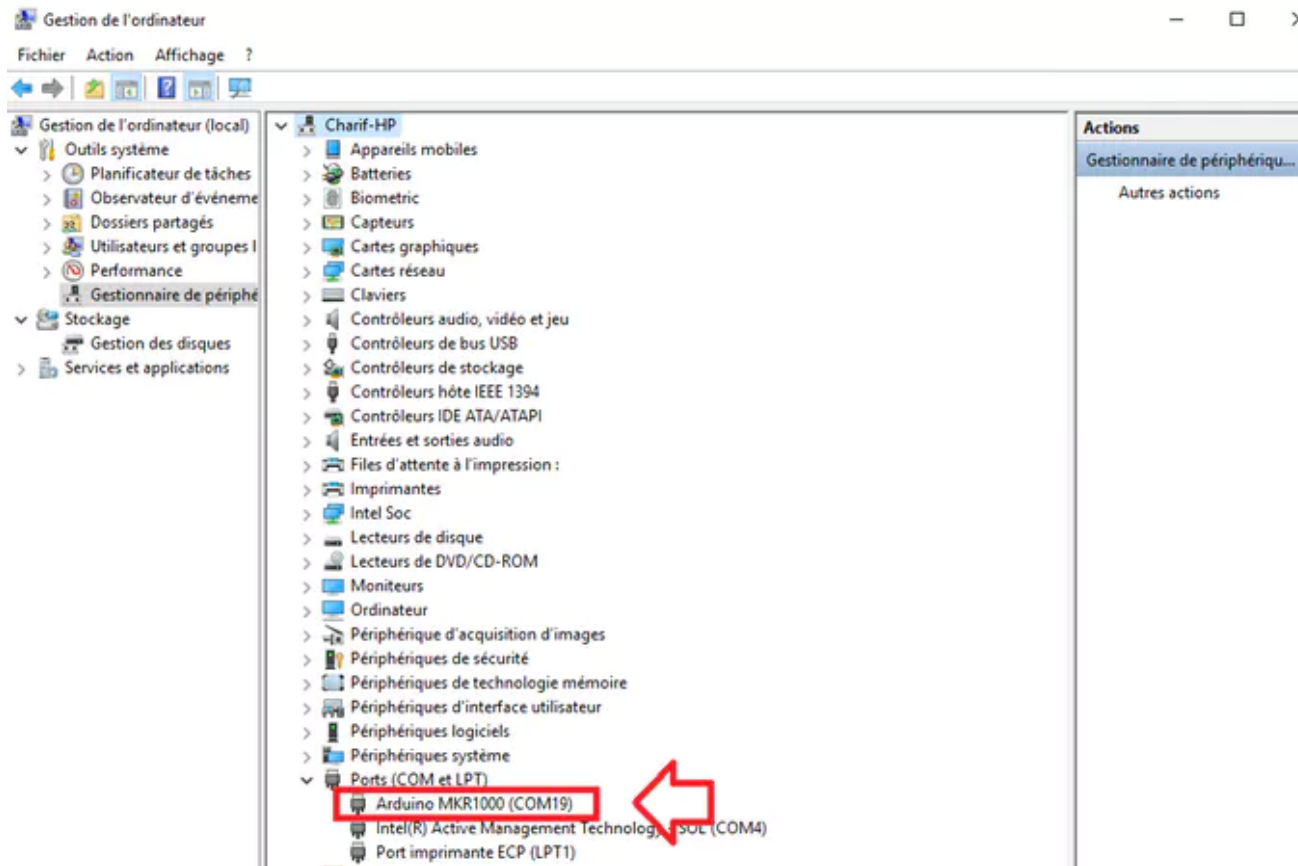
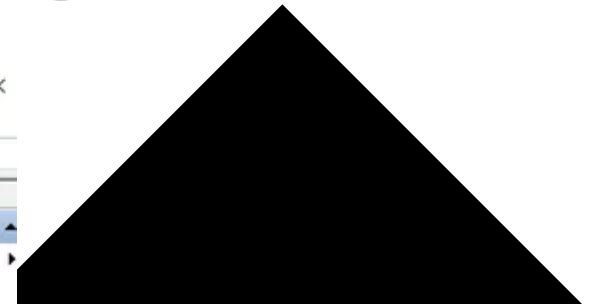
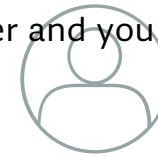


Figure 5



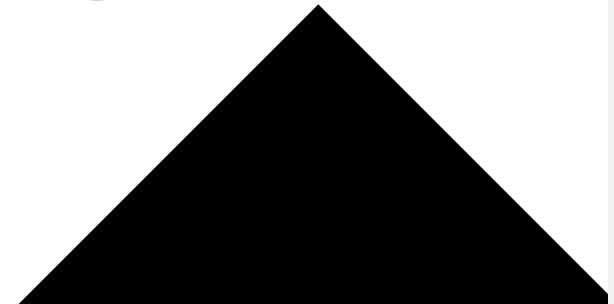
- Profile (<https://id.arduino.cc/>)
- Sign Out

Compile and upload your first MKR1000 sketch

Let's begin with the **Led blinking example**. We will use here the pin 6 instead of the 13 that come with the IDE example as illustrated bellow:

```
void setup() {
  pinMode(6, OUTPUT);
}

void loop() {
  digitalWrite(6, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(500);           // wait for 500ms
  digitalWrite(6, LOW);  // turn the LED off by making the voltage LOW
  delay(500);           // wait for 500ms
}
```



• Profile (https://id.arduino.cc/)

• Sign Out

To upload a sketch, choose **Arduino/Genuino MKR1000** from the Tools > Board menu in the Arduino IDE, and select the correct serial port from the Tools > Serial Port menu. In my case the port is COM19 as illustrated in the Figure 6.

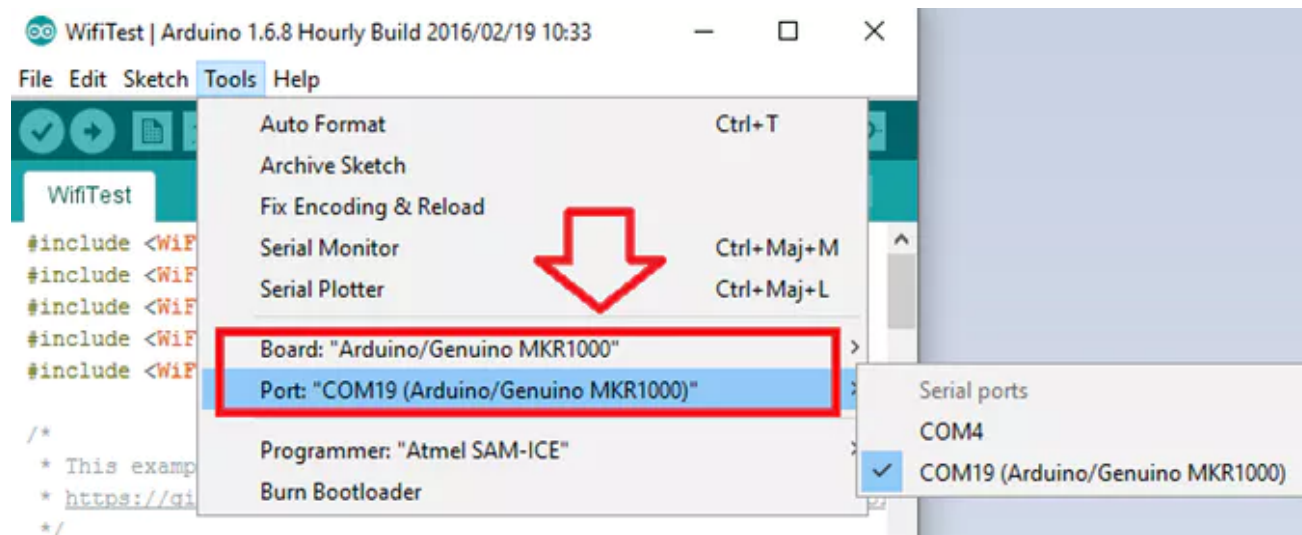


Figure 6

One uploaded, you should get something similar to the output on the Figure 7. The green Led should also blink each 500ms as illustrated in figure 8.



```

VintTest
void setup() {
  pinMode(6, OUTPUT);
}

void loop() {
  digitalWrite(6, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(500);           // wait for 500ms
}

```

Done uploading.

Sketch uses 8,156 bytes (3%) of program storage space. Maximum is 262,144 bytes.

Atmel SAMR device 0x10010005 found

Device : ATSAMR21G16A
 Chip ID : 10010005
 Version : v2.0 [Arduino:XYZ] Nov 24 2015 11:34:08
 Address : 0192
 Pages : 3968
 Page Size : 64 bytes
 Total Size : 248KB
 Pages : 1
 Lock Regions : 16
 Locked : none
 Security : false
 Boot Flash : true
 BOD : true
 BOR : true
 Arduino : FAST_CHIP_ERASE
 Arduino : FAST_MULTI_PAGE_WRITE
 Arduino : CAN_CHECKSUM_MEMORY_BUFFER

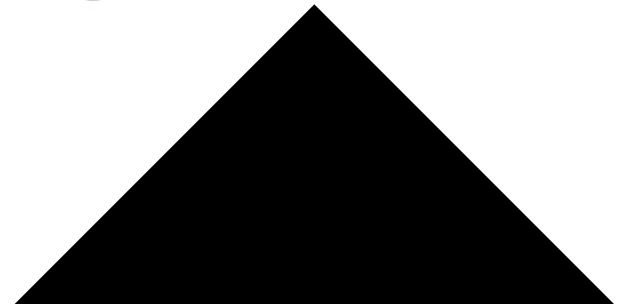
Erase flash
 done in 0.837 seconds

Write 8308 bytes to flash (130 pages)

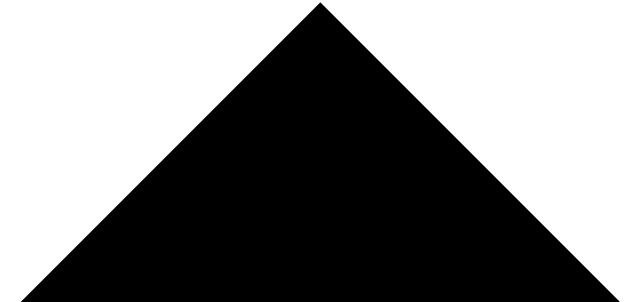
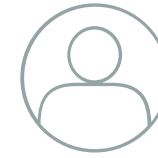
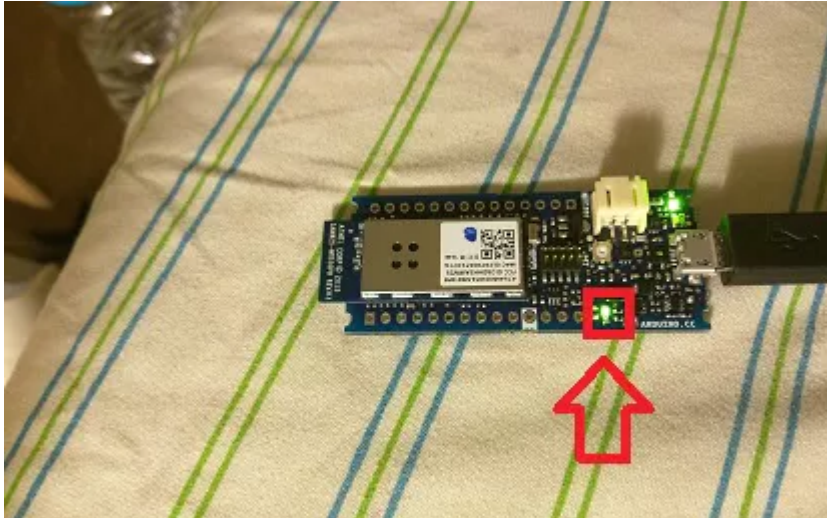
[=====] 49% (64/130 pages)
 [=====] 98% (128/130 pages)
 [=====] 100% (130/130 pages)
 done in 0.051 seconds

Verify 8308 bytes of flash with checksum.
 Verify successful
 done in 0.008 seconds
 CPU reset.

Figure 7



- Profile (<https://id.arduino.cc/>)
- Sign Out

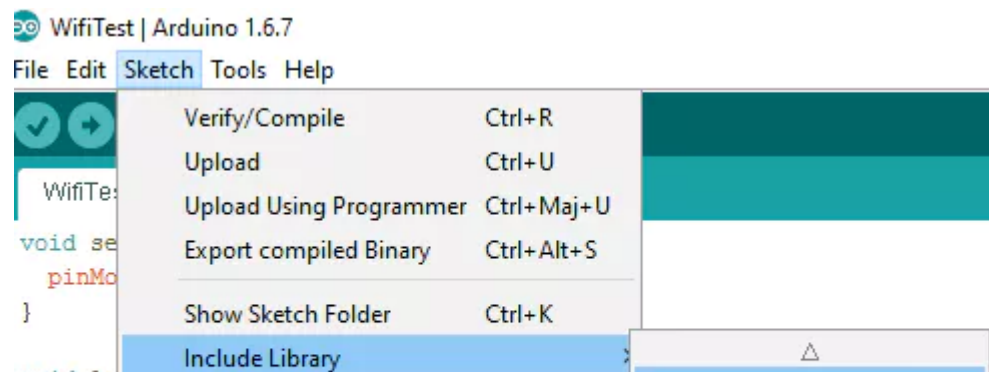


- Profile (<https://id.arduino.cc/>)
- Sign Out

One of the main features of the MKR1000 is its ability to access a WiFi network. To be able to use WiFi, you have to install the library first.

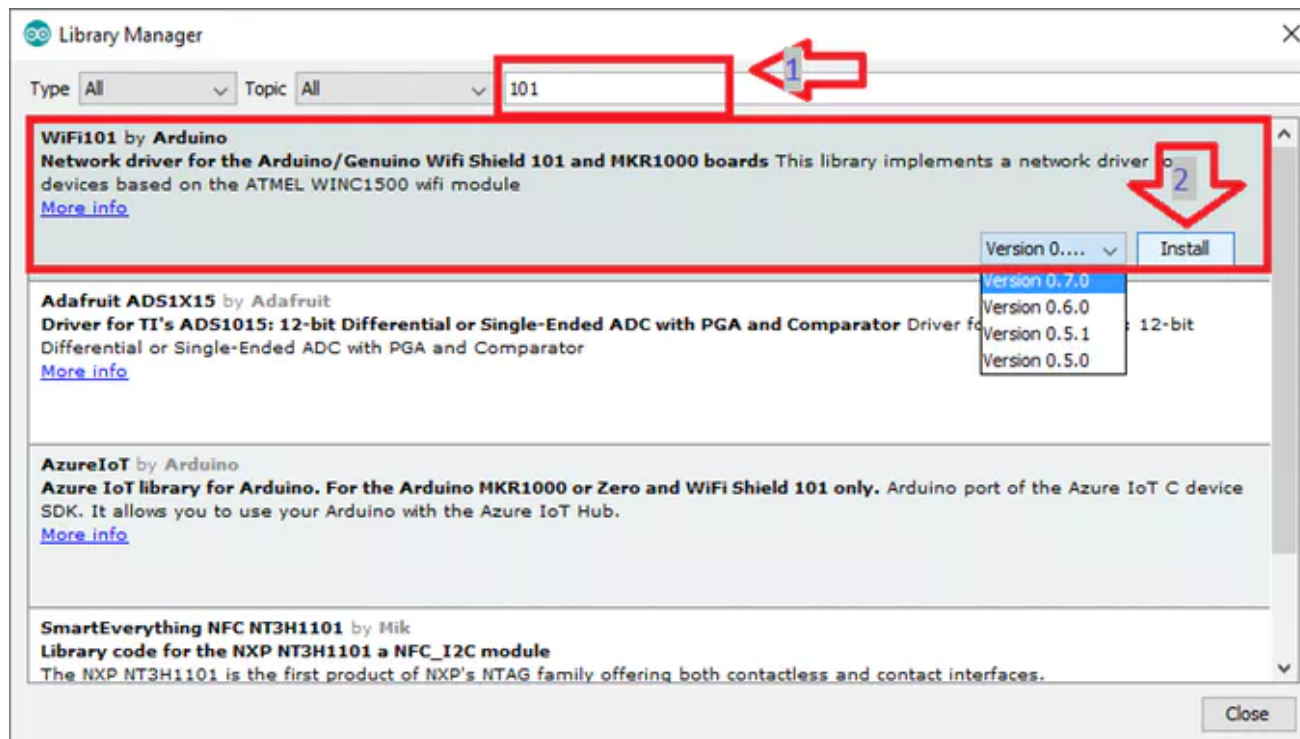
Install the WIFI 101 Library

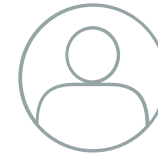
There is many way to install the wifi101 library (<https://github.com/arduino-libraries/WiFi101>) (you must use WiFi101 0.8.0) on the IDE. We propose in this tutorial to install this library using the Library manager. This method DO NOT WORK as is, it needs an extra manipulation described bellow while waiting for the library update. First go to **Sketch > Include Library > Manage Libraries** as illustrated in Figure 9.



Search "101" and install the WiFi101 library as illustrated in the Figure 10.

- Profile (<https://id.arduino.cc/>)
- Sign Out



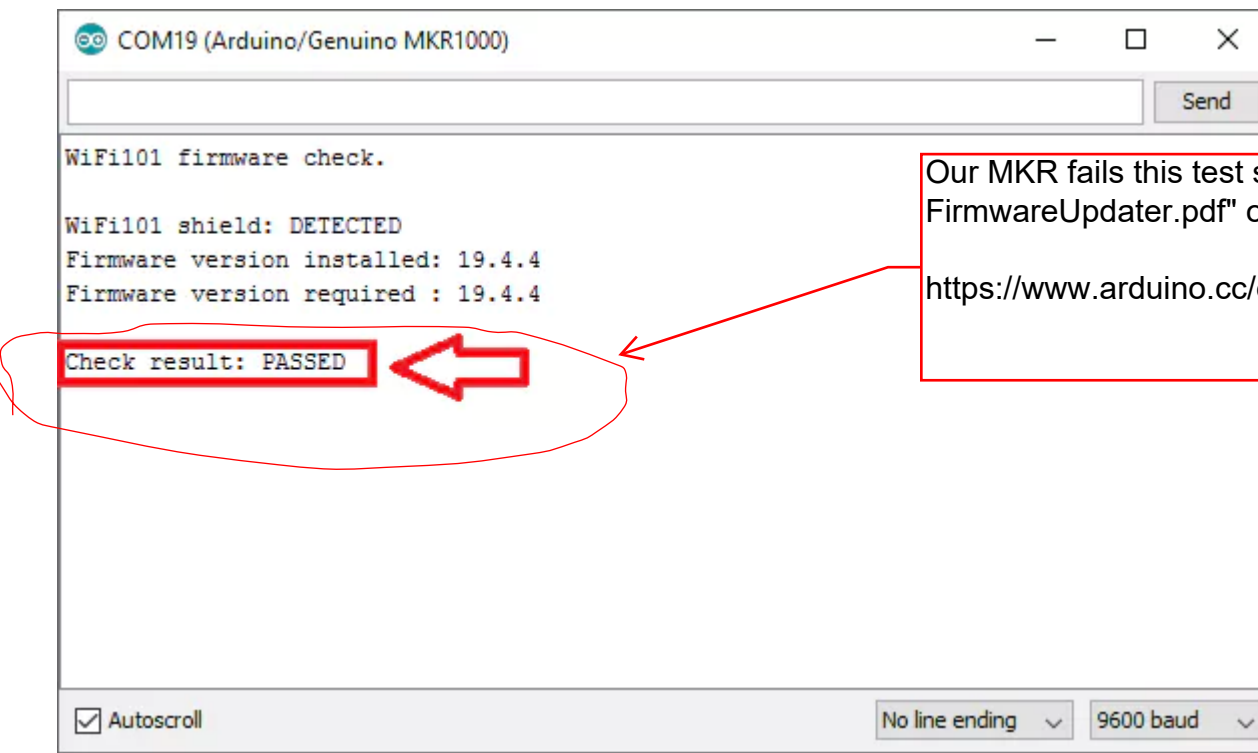


IF THE INSTALLED VERSION OF THE WIFI101 IS 0.7.0

Download the library from github (<https://github.com/arduino-libraries/WiFi101/archive/master.zip>) and open the folder "%userprofile%\documents\Documents\Arduino\libraries\WiFi101" and replace the content of the library with the content of the "WiFi101-master" folder in the downloaded zip

To check the WiFi101 library, open the Sketch located at **Examples > WiFi101 > CheckWifi101FirmwareVersion** illustrated in Figure 11.

- Profile (<https://id.arduino.cc/>)
- Sign Out



Our MKR fails this test so need that : "Arduino - FirmwareUpdater.pdf" or go to the webpage
<https://www.arduino.cc/en/Tutorial/FirmwareUpdater>

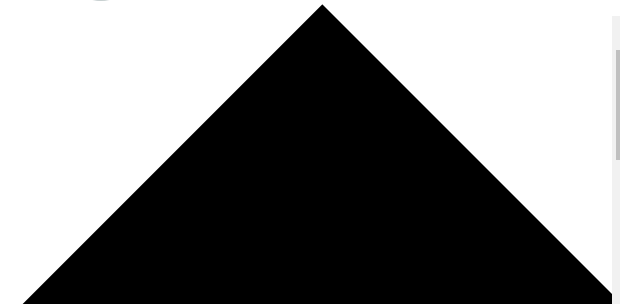
Figure 12

You can use the flowing code to start a web server that can turn on and off the MKR1000 Led, this code is an adaptation from the example from the WiFi101 library called **"SimpleWebServerWiFi"**. Once uploaded, you should see the address of the server in the Serial Monitor. Open it on any browser and you can enjoy executing the examples of the WiFi101 library.

```
#include <WiFi101.h>
#include <WiFiClient.h>
#include <WiFiServer.h>
#include <WiFiSSLClient.h>
#include <WiFiUdp.h>

/*
 * This example is modified from the original file
 * https://github.com/arduino-libraries/WiFi101/blob/master/examples/SimpleWebServerWiFi/SimpleWebServerWiFi.ino
 */
#include <SPI.h>
#include <WiFi101.h>

char ssid[] = "yourNetworkSSID"; // your network SSID (name)
char pass[] = "yourNetworkPassword"; // your network password
int keyIndex = 0; // your network key Index number (needed only for WEP)
int ledpin = 6;
bool val = true;
```



- Profile (https://id.arduino.cc/)
- Sign Out

Troubleshoot

As the MKR1000 still brand new, there is some issues.

There is a Topic (<https://forum.arduino.cc/index.php?topic=379236.15>) in Arduino forum about the wifi101 that may help. Another Topic (<http://forum.arduino.cc/index.php?topic=380708.0>) discuss the IDE related issues.

If you have issues with the Arduino/Genuino MKR1000 port, you can use the Zero port. Note that the official documentation (<https://www.arduino.cc/en/Guide/ArduinoZero>) of the Zero board commands using the Programming Port, However, it do not work for the MKR1000. So i recommend using the Native USB Port.

I would recommend to you to try 3 things if your device is not detected by windows:

- Verify that the USB cable that you are using supports data. The D+ and D- data lines are missing on some charge only cables. For that, try to connect your android device or another board using the USB cable and see if windows can detect it.
- Verify that the USB port in your computer. For that, simply try to plug device in some other USB port and wait some second to see if there is any change in the windows device manager. Sometimes unplugging the device and restarting windows resolve this issue.
- Verify that the driver is installed correctly. Open the **Device manager > Ports**, unplug the MKR1000 and plug it, if you see a new device appear that is not recolonized as MKR1000, **right click** on this device and click on **update the driver**. Click on **choose the driver from my computer** than choose the Arduino drivers folder. This should update the driver and detect the device as MKR1000.

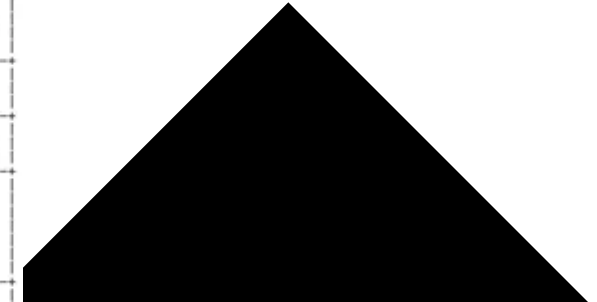
• Profile (https://id.arduino.cc/)

Sign Out

Advanced Pins description

An interesting pins description is illustrated in Figure13. This description was included in the code (then removed) from SAMD; It was used for their experiments on the MKR1000. The commit (<https://github.com/arduino/ArduinoCore-samd/commit/e6d19f7c06a53d9e9e67fb50e9f9dc2ca2e7b9f4>) is available on github SAMD repository (<https://github.com/arduino/ArduinoCore-samd/>).

Pin number	MKR Board pin	PIN	Notes	Peri. A EIC (EXTINT)	ADC (AIN)	Peripheral B AC (AIN)	PTC	DAC	Perip. C SERCOMx (x/PAD)	Perip. D SERCOMx (x/PAD)	Peri. E TCCx (x/IO)	Peri. F TCCx (x/IO)	Periph. G COM	Periph. H AC/GLCK
00	D0	PA22		•06			X10		3/00	5/00	• TC4/0	TCC0/4		GCLK_IO6
01	D1	PA23		•07			X11		3/01	5/01	• TC4/1	TCC0/5		GCLK_IO7
02	D2	PA10		10	•18		X02		0/02	2/02	•TCC1/0	TCC0/2	USB/SOF	GCLK_IO4
03	D3	PA11		11	•19		X03		0/03	2/03	•TCC1/1	TCC0/3	I2S/FS0	GCLK_IO5
04	D4	PB10		•10					4/02	4/02	• TC5/0	TCC0/4	I2S/MCK1	GCLK_IO4
05	D5	PB11		•11					4/03	4/03	• TC5/1	TCC0/5	I2S/SCK1	GCLK_IO5
06	D6	PA20		•04			X08		5/02	3/02		•TCC0/6	I2S/FS0	GCLK_IO4
07	D7	PA21		•05			X09		5/03	3/03		•TCC0/7	I2S/FS0	GCLK_IO5
08	MOSI	PA16		•00			X04		•1/00	3/00	•TCC2/0	TCC0/6		GCLK_IO2
09	SCK	PA17		•01			X05		•1/01	3/01	TCC2/1	TCC0/7		GCLK_IO3
10	MISO	PA19		03			X07		•1/03	3/03	• TC3/1	TCC0/3	I2S/SD0	AC/CMF1
11	SDA	PA08		NMI	•16		X00		•0/00	2/00	TCC0/0	TCC1/2	I2S/SD1	
12	SCL	PA09		09	•17		X01		•0/01	2/01	TCC0/1	TCC1/3	I2S/MCK0	
13	RX	PB23		07						•5/03				GCLK_IO1
14	TX	PB22		06						•5/02				GCLK_IO0
15	A0 / DAC0	PA02		02	•00		Y00	OUT		5/00				
16	A1	PB02		•02	•10		Y08			5/00				
17	A2	PB03		•03	•11		Y09			5/01				
18	A3	PA04		04	•04	00	Y02			0/00	•TCC0/0			
19	A4	PA05		05	•05	01	Y03			0/01	•TCC0/1			
20	A5	PA06		06	•06	02	Y04			0/02	TCC1/0			
21	A6	PA07		07	•07	03	Y05			0/03	TCC1/1		I2S/SD0	
22		PA24	USB N	12					3/02	5/02	TC5/0	TCC1/2	USB/TM	
23		PA25	USB P	13					3/03	5/03	TC5/1	TCC1/3	USB/DP	
24		PA18	USB ID	02			X06		1/02	3/02	TC3/0	TCC0/2		AC/CMF0
25	AREF	PA03		03	01		Y01							
26	ATVINC1501B SPI	PA12	WINC MOSI	12					•2/00	4/00	TCC2/0	TCC0/6		AC/CMF0
27		PA13	WINC SCK	13					•2/01	4/01	TCC2/1	TCC0/7		AC/CMF1
28		PA14	WINC SSN	14					2/02	4/02	TC3/0	TCC0/4		GCLK_IO0



- Profile (<https://id.arduino.cc/>)
- Sign Out

Use Microsoft Azure IoT

Steps are very well explained in this link:

Azure IoT Doc (<https://github.com/Azure/azure-iot-sdks/blob/master/readme.md>)

Do not hesitate to post comment on this tutorial if you need help.

COMMENTS

Write

Preview