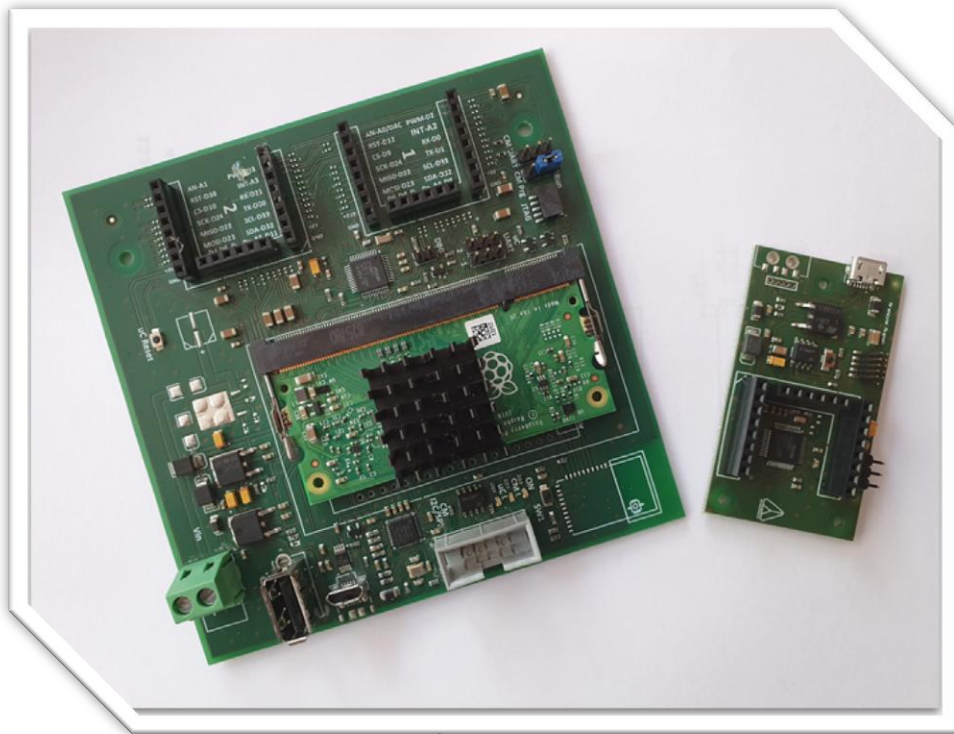




# User Guide

boards arancino.cc™ and arancino.cc™ mignon

Copyright © 2020 SmartMe.IO Srl  
November 2020 - Version 1.5



**INDEX**

INDEX ..... 2

INTRODUCTION ..... 3

    PRODUCTS..... 3

    EU DECLARATION OF CONFORMITY ..... 3

    APPLICATIONS..... 3

    DOCUMENT EDITION ..... 4

    WARNING AND GENERAL PRECAUTIONS ..... 4

    WARRANTY ..... 4

    LIMITATIONS AND DISCLAIMER OF THE WARRANTY ..... 4

ARANCINO.CC ARCHITECTURE ..... 5

    ARANCINO BOARD (vers. 1.0.2) ..... 6

    ARANCINO MIGNON BOARD (vers. 1.2)..... 13

    ARANCINO ADD-ON ..... 16

CONFIGURATION AND INSTALLATION ..... 18

    ARANCINO.CC INSTALLATION ..... 18

    IMPORT THE ARANCINO LIBRARY IN ARDUINO IDE ..... 19

    HOW TO FLASH THE ARANCINO.CC™ MICROCONTROLLER..... 20

    HOW TO FLASH ARANCINO.CC™ OS ON THE COMPUTE MODULE ..... 21

    ARANCINO CONFIGURATOR..... 22

THE ARANCINO.CC FIRMWARE..... 24

    ARANCINO.CC BOOTLOADER ..... 24

    BOOTLOADER FLASHING ..... 24

USE CASES..... 26

CONTACTS..... 27

## INTRODUCTION

*Dear Customer,*

*We thank you for choosing one of our products whose correct use guarantees solutions of the highest quality and reliability over time.*

In many applications it is useful to have continuous monitoring of internal and / or external environments for statistical purposes, safety, evaluation of effectiveness and optimization. **Arancino.cc™** is an innovative integrated system developed by SmartMe.IO® that can be used in applications ranging from simple temperature sensing to automotive applications, artificial intelligence, machine learning, neural networks, cloud, big data analysis, predictive maintenance, etc.

Arancino.cc finds a wide field of application in "Physical Computing", in order to create systems capable of interacting with the external environment through the use of hardware and software. The system interacts with the outside in a bidirectional way, according to the model of the Internet of Things: it acquires the quantities detected by the sensors, processes them through the microcontroller or remotely, and then performs the appropriate actions through the actuators.

The operating system used (called arancino OS™) is a Linux distribution designed to be used on embedded systems, effectively transforming the product into a real single-board computer, to which the microcontroller is added for real-time code execution .

The compute module used on the arancino.cc™ board is the “raspberry Pi CM3 +” on which the arancino OS™ operating system is generally already installed.

To program the firmware of the microcontroller you need to install the Arduino IDE development environment on your computer and configure it for the arancino.cc platform.

## PRODUCTS

- arancino.cc™ Board (1.0.2)
- arancino.cc™ Mignon Board (1.2)

## EU DECLARATION OF CONFORMITY

The above-mentioned products, placed on the European Community market by the Company, are in conformity with:

- the Directive 2014/30 / EU of the European Parliament and of the Council of 26 February 2014 on the harmonization of the laws of the Member States relating to electromagnetic compatibility (recast).
- the Directive 2011/65 / EU on the restriction of Certain Hazardous Substances in Electrical and Electronic Equipment.

## APPLICATIONS

- Internet of Things
- Cyber Physical Systems (CPS)
- Smart Environments
- Industrial
- Automotive
- Utilities

## DOCUMENT EDITION

I edition (version 1.5 - November 2020). Updated arancino.cc board configuration procedures.

I edition (version 1.4 - May 2020). Reference to the new website <https://arancino.cc/> .

I edition (version 1.3 - March 2020). Updated URL for installation on Arduino IDE.

I edition (version 1.2 - March 2020). Updated details on microcontroller flashing.

I edition (version 1.1 - February 2020). Arduino IDE installation procedures updated.

I edition (version 1.0 - January 2020).

## WARNING AND GENERAL PRECAUTIONS

Please carefully read this document before proceeding to installing the device and keep it in a safe place in good condition for future consultation and maintenance of the device.

- Make sure that the device has not been damaged during transport, the storage phase or the installation.
- The installation must be carried out in an appropriate way and place, in order to guarantee the correct functioning of the device.
- The installation and maintenance of the device must be carried out in full compliance with current safety regulations.

## WARRANTY

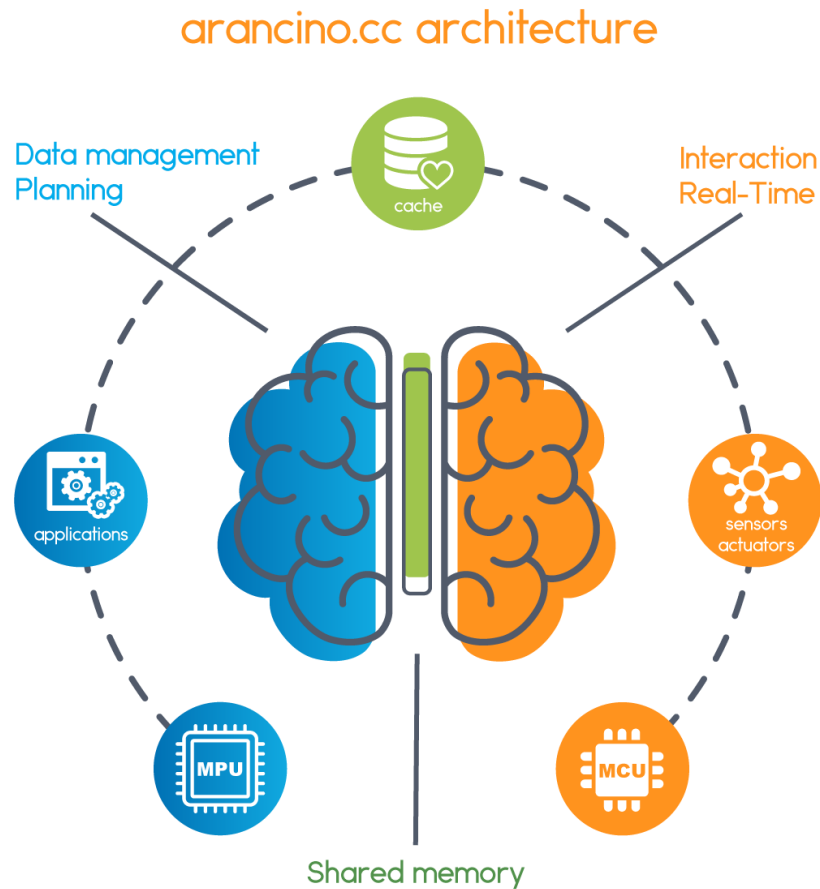
**12 months.** Any changes or alterations made to the equipment, if they have not been expressly approved **in writing** by the Manufacturer, void the warranty and the authorization for use by user.

## LIMITATIONS AND DISCLAIMER OF THE WARRANTY

In no event will the retailer be liable to any other party or person for the following:

- Damage by negligence in compliance with the general precautions.
- Personal damage or any damage caused by inappropriate use or negligence by the User.
- Unauthorized dismantling, repair or modification of the product by the User.
- Any problem, resulting in inconvenience, loss or damage, resulting from the system in combination with nearby devices.

## ARANCINO.CC ARCHITECTURE



*fig. 1: arancino architecture.cc*

**arancino.cc™** is an architecture developed by smartme.IO® based on the same concept of communication between the right and left hemisphere of the human brain.

The **left hemisphere** is dominant for the functions of calculation and logical and mathematical ability (**Data management, Planning**); the **right hemisphere**, on the other hand, is dominant in its ability to recognize faces, spatial abilities and images (**Real-time, Interaction**).

The activity of the two hemispheres is coordinated thanks to the continuous exchange of information that takes place through the **corpus callosum**, the element that connects them (**Shared memory**). The different specializations of the two hemispheres allow them to work together more effectively. The two parties work together and one takes "control of operations" as appropriate.

The arancino.cc™ architecture simplifies cloud-IoT interaction and facilitates the implementation of Cyber Physical Systems, and also leverages edge and fog computing and adapts seamlessly to artificial intelligence and machine learning solutions.

## ARANCINO BOARD (vers. 1.0.2)

The arancino.cc board (illustrated in fig. 2 with its protective case provided with housing for din bar) is made up of two main parts: microprocessor and microcontroller; through a simple module/click, the arancino microcontroller is able to communicate directly with the on-board Compute Module.



fig. 2: board arancino.cc front / back

The **microcontroller** takes care of making the communication between the microprocessor part and the sensors uniform, which according to the protocol used (SPI, UART, I2C, ...) allow to be interconnected in various ways. The microcontroller is directly linked to two connectors (connector 1 and connector 2) derived from mikroBUS (a standard for connectors created to simplify connections as it contains the main pins generally required by a board), to which further relative signals have been added to USB and GPIO connections, making these connectors a real extension of the mikroBUS, with which they are compatible.

This choice originates from the need to have immediately available numerous sensors, transducers and connection systems (GSM, LTE, RAM, etc.) and to be able to easily connect them to the board by mounting them directly on the arancino connector.

The microcontroller on the board is an Atmel SAM D21G18A, belonging to a family of 32-bit microcontrollers with ARM Cortex-M0 + core, equipped with 256 kB flash memory and 32KB SRAM memory. It allows you to work at a maximum operating frequency of 48 MHz (reaches 2.46 Coremark / MHz), and is characterized by low consumption.

The **microprocessor** part deals with high-level processing, management of network connectivity (WiFi, etc.) and programming of the microcontroller. The **Raspberry PI CM3 + Compute Module** is used; it's based on the Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.2GHz, 1GB LPDDR2 SDRAM, 8GB / 16GB / 32GB eMMC Flash memory processor, or Lite variant without eMMC Flash memory.

The arancino board has wireless connectivity and is equipped with **two arancino Connectors 1 and 2** that allow to manage externally various types of peripherals (UART, SPI, I2C, ADC, DAC, GPIO, USB Host 2.0).

**Arancino.cc™ Board (1.0.2) Technical Characteristics**

- Microcontroller
  - ARM Cortex M0 + running at 48MHz
  - 256kB Flash
  - 32kB SRAM
- I / O devices
  - Up to 32 GPIO
  - Up to 6x 350ksps 12-bit ADC with programmable gain
  - 1x 10-bit 350ksps DAC
  - 12 Channels DMA Controller
  - 12 Event System Channels
  - Programmable interrupt Controller
  - 32-bit Real Time Clock and calendar
  - 3x 24-bit Timer / Counter
  - Watchdog Timer (WDT)
  - 3x Full-Speed USB 2.0 ports
  - 2x I2C Interface
  - 2x SPI Interface
  - 1x I2S Interface
  - 2x UART
  - Expansions: 2x arancino.cc™ connectors

**Technical Characteristics of Arancino.cc-compatible Microprocessor**

- **Compute Module PI 1**
  - CPU: Broadcom BCM2835 @ 700 MHz
  - RAM: 512 MB LPDDR2
  - MEMORY: 4GB eMMC
- **Compute Module PI 3**
  - CPU: Broadcom BCM2837 @ 1.2GHz
  - RAM: 1 GB LPDDR2
  - MEMORY: 4 GB eMMC.
- **Compute Module PI 3+**
  - CPU: Broadcom BCM2837B0 Cortex A-53(ARM v8) 64bit SoC @ 1.2GHz
  - RAM: 1 GB LPDDR2
  - MEMORY: 8/16/32 GB eMMC.
- **Linkit Smart 7688**
  - CPU: MIPS24KEc @ 580MHz
  - RAM: 128 MB LPDDR2
  - MEMORY: 32 MB eMMC.
  - Slot microSD, host USB, pinout GPIO, I2C, I2S, SPI, UART, PWM, Ethernet port
  - Single Input/Single Output (1T1R) Wi-fi 802.11/b/g/n.

Let's analyze in detail the various components available on the arancino.cc board.

### SLOT 1 and 2

In the upper part of the board there are two connectors connected only to the microcontroller, with the exception of the two pins D + and D- of each connector, connected to the compute module. All other pins are GPIO type, configurable as communication channels.



fig. 3: arancino.cc connectors 1 and 2

The **arancino.cc™ Connector** is an expansion which is present on the arancino.cc™ boards. The purpose is to enable easy hardware expandability with a large number of standardized compact add-on boards, each one carrying a single sensor or compliant hardware module.

The following figures show the pin out of each of the two connectors:

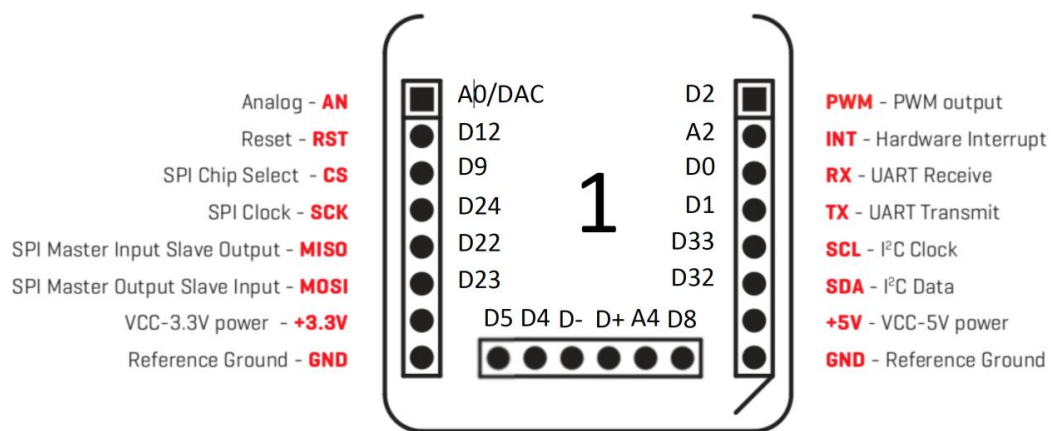


fig. 4: arancino connector 1



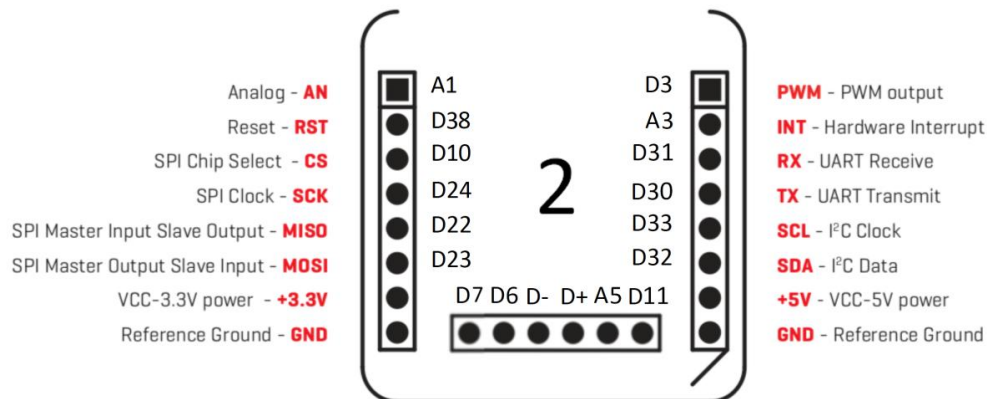


fig 5 arancino connector 2

On both connectors there are 4 SPI pins (Serial Peripheral Interface) used for synchronous serial communication between the embedded systems and the microcontroller; in particular we have:

- two D22 pins, one for each connector, of MISO type (Master Input Slave Output), used to send data to the master;
- two D23 pins, one for each connector, of the MOSI (Master Output Slave Input) type, used for sending data to peripherals;
- two D24 SCK pins (SPI Clock) that send the clock pulses to synchronize the transmission of the data generated by the master;
- two separate CS (Chip Select) pins, pin D9 in connector 1 and pin D10 in connector 2.

The D9 and D10 pin allow you to select two different devices through the same SPI channel: in practice, if you have 2 SPI devices connected to the board, via any GPIO, you can select the two devices in the communication channel via firmware.

Two different serial ports can be managed on the board, accessible either by the two Rx / Tx UART connectors (D0 and D1 pins of connector 1, D30 and D31 pins of connector 2) or by  $\mu$ UART pins placed in slot 4 located below connector 1.

Through the pins D32 SDA (Serial Data line) and D33 SCL (Serial Clock Line) of both connectors it is possible to manage different devices (selectable by an address) on the serial communication channel I<sup>2</sup>C.

The pins marked with the initial letter D (Digital) or A (Analog) are configurable for digital or analog signals.

In particular, we must pay attention to the fact that **D2** and **D10** pins (in the 1.2.0 version of the SAMD21 core) cannot be used simultaneously as PWM pins; they use the same timer, and a variation in the duty cycle of one also affects the duty cycle of the other pin.

From both connectors it is possible to take three different power supplies: a common ground (GND) and two direct current voltages of 5V and 3.3V for each connector.

From firmware there is also the possibility to select all connector pins as interrupt pins.

Connectors 1 and 2 respect the mikroe standard (2 parallel connectors of 8 pins each), therefore they are completely compatible with Mikroelektronika clicks (generally used in addition to click arancino). Compared to

the mikroe connector, the arancino Connector additionally provides a 6-pin connector characterized by two USB pins (D- and D +), and other GPIO pins marked by the letters D (digital) or A (analog).

### SLOT 3

On the left side of the board there is a **µC Reset button** strictly used to reset the microcontroller. Pressing it quickly for two consecutive times runs the bootloader of the microcontroller, which waits to receive the program to be executed.



### SLOT 4



In slot 4 there are:

- 2 µC UART connectors used for serial connection;
- 1 DBG connector for the debugging of the board.

### SLOT 5

In the slot next to connector 1 there are:

- a JTAG connector (2x5) used to load and program the microcontroller bootloader;
- a CM UART connector connected to the “com” port of the microcontroller;
- a CM Prg/Run connector that allows you to use the microcontroller in two different modes, running or programming.

Using the board in running mode, the microcontroller can also be programmed through the compute module.



## SLOT 6

In the central part of the board it is possible to apply a **raspberry PI compute module 3+ microprocessor** or a **Linkit Smart 7688** board in a mutually exclusive way.



The compute module is very compact: it has the typical dimensions of a DDR2 SoDIMM memory, and can be easily installed in the soDIMM socket mounted on the arancino board.

Generally a heat sink is mounted on the raspberry compute module (as shown in the figure above on the right) in order to lower the high temperatures (60-70 ° C) reached during the use of the board.

The Linkit Smart 7688 board features a 580Mhz Mips CPU, 32Mb flash memory and 128Mb DDR2 RAM memory. There is also a slot for microSD, a USB host and a pin out for GPIO, I2C, I2S, SPI, UART, PWM and ethernet connections.

In the central part of the slot there is also a crypto-chip, designed to encrypt the information that comes in input or output from the USB port.

## SLOT 7: POWER

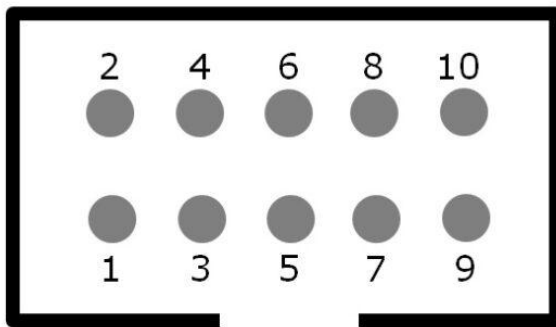


The board is powered by 5V DC voltage through a microUSB port or through a DC power jack. If the arancino.cc board is equipped with an OKI DCDC Converter 78SR-5/15, the jack could be powered in a wider voltage range between 7V and 36V. Once the board is powered, the branch chain supplies the various components of the board with all the necessary power supplies (1.8V - 3.3V and 5V and ground). In particular, the 1.8V power supply is mainly used by the compute module, while the remaining power supplies can be taken directly from the + 3.3V, + 5V and GND pins of both connectors 1 and 2 on the arancino.cc board.

The USB port between the power jack and the micro USB can instead be used in data mode, for example by connecting a mass memory, or an external modem for internet connection.

## SLOT 8

In slot 8, located below the microprocessor / linkit socket, there are a 2x5 connector, 1 reset button SW1 and three LED diodes:



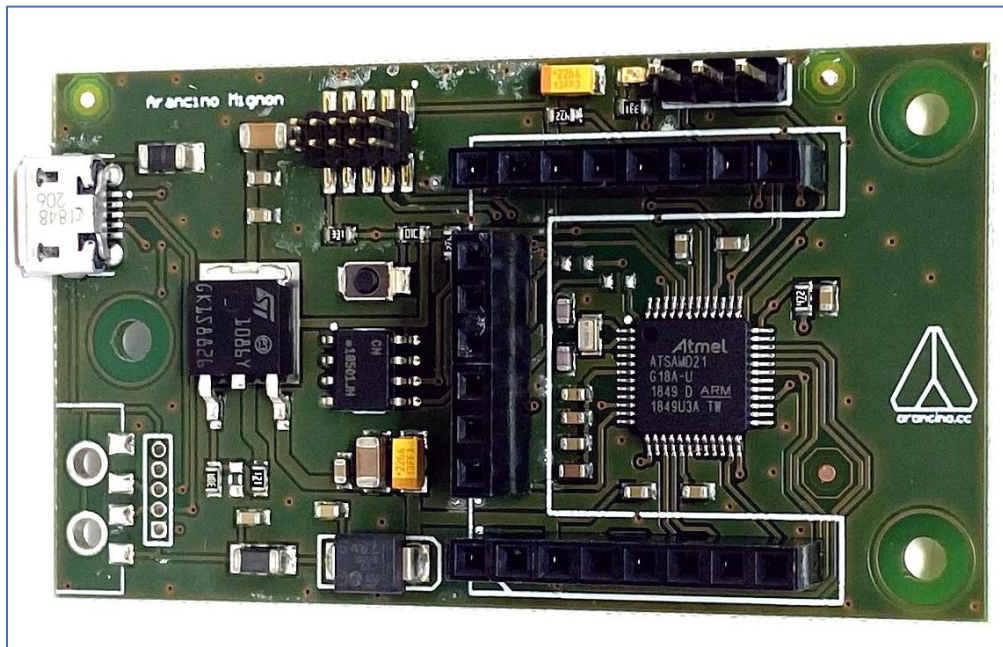
The pin-out of the (2x5) connector is shown in the figure on the left. It is related to the compute module and consists of the following pins (GPIO pins are fully user configurable):

- 1: 3,3V DC power supply;
  - 2: GPIO 8/SPI CEO N;
  - 3: GPIO 2/SDA1;
  - 4: GPIO 11/SPI SCLK;
  - 5: GPIO 3/SCL1;
  - 6: GPIO 9/SPI MISO;
  - 7: GPIO 26;
  - 8: GPIO 10/SPI MOSI;
  - 9: Ground;
  - 10: 5V DC power supply.
- the **reset button SW1** can be fully configurable via software; by default, holding it down for a couple of seconds, the board enters in ap-mode (it can be used as an access point);
  - **led ON**: when the board is powered, the led stays on;
  - **CML led**: when it is on the operating system is running mode;
  - **led μCL**: it depends by the microcontroller, and as the CML led, it can be managed by software; it blinks if it works in writing mode.

## ARANCINO MIGNON BOARD (vers. 1.2)

**Arancino.cc Mignon** is a programmable hardware platform through which it is possible to interface with a computer or other input/output devices and peripherals.

The arancino mignon board (illustrated in Fig.1 in version 1.2) is composed of two main parts: a microcontroller and an Arancino Connector, to which it is possible to connect an arancino module or a click mikroe.



*Fig12: board arancino.cc mignon*

The microcontroller on the board is an Atmel SAM D21G18A, belonging to a family of 32-bit microcontrollers with ARM Cortex-M0 + core, equipped with 256 kB flash memory and 32KB SRAM memory. It allows you to work at a maximum operating frequency of 48 MHz (reaches 2.46 Coremark / MHz), and is characterized by low consumption.

Inside there is a firmware (bootloader) capable of loading the program instructions (sketch) in the memory of the mignon board.

The microcontroller is directly connected to the arancino.cc Connector.

**Arancino Connector** represents an extension of the mikroBUS connector (a standard for connectors created to simplify connections, containing the main pins generally required by a board), to which additional signals relating to USB and GPIO connections have been added with which they are compatible.

This choice originates from the need to have immediately available numerous sensors, transducers, connection systems (GSM, LTE, RAM, etc.) and peripherals of various types (UART, SPI, I2C, ADC, DAC, GPIO, USB Host 2.0) and to be able to easily connect them to the arancino mignon board.

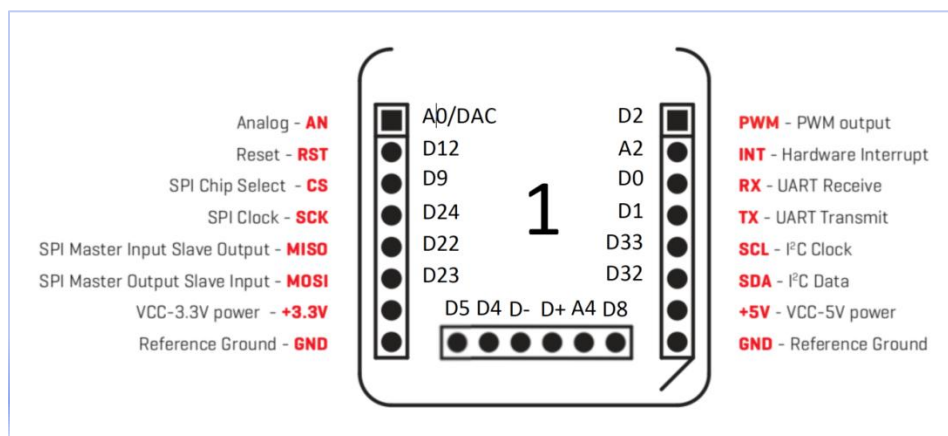
The aim is to enable easy hardware expandability with a large number of standardized compact add-on boards, each with a single sensor or compliant hardware module.

## Arancino.cc™ Mignon Board (1.2) Technical Characteristics

- Microcontroller
  - ARM Cortex M0 + running at 48MHz
  - 256kB Flash
  - 32kB SRAM
- I / O devices
  - Up to 18 GPIO
  - Up to 3x 350ksps 12-bit ADC with programmable gain
  - 1x 10-bit 350ksps DAC
  - 12 Channels DMA Controller
  - 12 Event System Channels
  - Programmable interrupt Controller
  - 32-bit Real Time Clock and calendar
  - 3x 24-bit Timer / Counter
  - Watchdog Timer (WDT)
  - 1x Full-Speed USB 2.0 port
  - 1x I2C interface
  - 1x SPI interface
  - 1x I2S interface
  - 2x UART
- Expansions
  - 1x arancino.cc™ Connector

### Arancino connector 1

Unlike the motherboard arancino.cc, on which there are 2 arancino connectors (1 and 2), on the arancino mignon board there is only connector 1, and in particular, it does not allow the USB connection on pins D+ and D- which therefore are unusable.



*pin out connector 1 Arancino*

Arancino Connector is directly connected to the microcontroller, and all its pins are GPIO type, configurable as communication channels.

The **4 SPI pins** (Serial Peripheral Interface) are used for the synchronous serial communication between the embedded systems and the microcontroller; in particular we have:

- a D22 pin of MISO type (Master Input Slave Output), used for sending data to the master;
- a MOSI (Master Output Slave Input) pin D23, used for sending data to peripherals;
- a D24 SCK pin (SPI Clock) which sends the clock pulses to synchronize the transmission of the data generated by the master;
- a D9 CS pin (Chip Select), which allows you to enable/disable the device connected on the SPI channel.

The serial ports connected to the board can be managed by connecting them either to the two Rx/Tx UART connectors (D0 and D1 pins of connector 1) or through the pins of the connector **μUART** placed on the side of the arancino connector.

Through D32 SDA (Serial Data line) and D33 SCL (Serial Clock Line) pins it is possible to manage the devices (selectable by an address) connected to the serial communication channel I<sup>2</sup>C.

The remaining pins marked with the initial letter D (Digital) or A (Analog) are configurable for digital or analog signals.

From both connectors it is possible to take three different power supplies: a common ground (GND) and two direct current voltages of 5V and 3.3V for each connector.

From firmware there is also the possibility to select all connector pins as interrupt pins.

**Arancino Connector** it complies with the mikroe standard (2 parallel connectors of 8 pins each), therefore it is completely compatible with the click modules of Mikroelektronika (generally used in addition to the arancino.cc add-on).

Compared to the mikroe connector, arancino Connector additionally is equipped with a 6-pin connector characterized by two D+ and D- pins related to the USB connection (useless on arancino mignon 1.2), and other GPIO pins marked by the letters D (digital) or A (Analog).

### Additional components of Arancino Mignon

In addition to the microcontroller and to the arancino connector, the following components are present on the board:

- a Full-speed 2.0 USB port used to power the arancino mignon board with a voltage of 5V in direct current;
- a JTAG connector (2x5) used to program the firmware (load and program the bootloader) of the microcontroller;
- an LDO (Low Drop Out) linear voltage regulator used to lower the voltage value from 5V to 3.3V;
- a crypto-chip capable of encrypting the information that enters or exits the USB port;
- **μC Reset button** used to reset the microcontroller; pressing it quickly for two consecutive times, it runs the microcontroller bootloader, which waits to receive the program to be executed.
- a UART connector connected to the COM port of the microcontroller for the serial connection;

Through the communication channels SPI, I<sup>2</sup>C, UART present on the arancino mignon board it is possible to establish a connection between the arancino mignon microcontroller and the microcontroller present on the arancino board. In addition, by connecting the two boards via USB, it is possible to connect arancino mignon to the compute module mounted on the arancino board.

## ARANCINO ADD-ON

Currently three **arancino.cc modules** are available, designed to be mounted on the Arancino Connectors present on the arancino and arancino mignon boards: two modules (Wifi Module, USB module) and a recipe (Weather Recipe).

### Arancino Wifi Module

Arancino Wifi module is a USB 2.0 module compliant with IEEE802.11 b/g/n Wireless Local Area Networks standards.

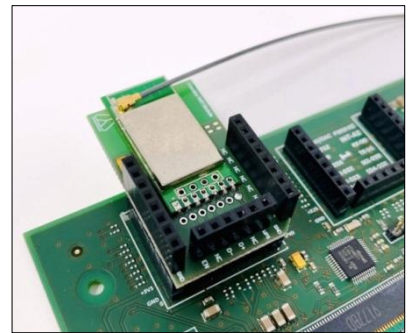
It is based on a RaLink RT5370N chip and is fully supported by the latest Linux Kernel versions. It can be mounted directly on an Arancino Connector and is powered by a continuous voltage of 3.3 Volts. It can operate in managed mode, ad-hoc mode or access point mode.

The module requires an external RF antenna connected to the MHF i-PEX socket.

The RT5370 is a BBP (Baseband Processor) chip made in CMOS technology.

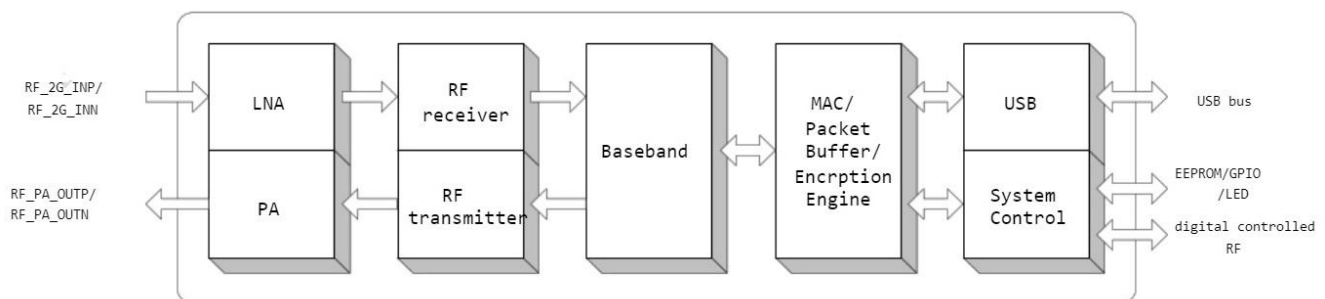
It is a SoC (Single on a Chip) on which the RF/PA/LNA and MAC modules are integrated, and operates at a frequency of 2.4 GHz in a temperature range between -10 ° and 70 ° C.

It supports PHY speeds of 150 Mbps in both transmission and receiving mode.



The optimized RF architecture complies with the WEP 64/128, WPA, WPA2, TKIP, AES, and WAPI security protocols, providing excellent performance and low power consumption.

The chip is also equipped with CISCO CCX and Multiple BSSID supports, and offers high quality multimedia service QoS-WMM, WMM-PS.



*RT5370: functional block diagram*

Through the Arduino IDE it is possible to install and program the module firmware, which will load at startup.



## Features of RT5370

- CMOS Technology with PA, LNA, RF, Baseband, and MAC Integrated.
- 1T1R Mode with 150Mbps PHY Rate for Both Transmit and Receiving.
- Legacy and High Throughput Modes
- 20MHz / 40MHz Bandwidth
- Reverse Direction Grant Data Flow and Frame Aggregation
- WEP 64/128, WPA, WPA2, TKIP, AES, WAPI
- QoS-WMM, WMM-PS
- WPS, PIN, PBC
- Multiple BSSID Support
- USB 2.0
- Cisco CCX Support
- Bluetooth Co-existence
- Low Power with Advanced Power Management
- Operating Systems - Windows XP 32/64, 2000, Windows 7, Vista 32/64, Linux, Macintosh
- Temperature range between -10 ° and 70 ° C

## Arancino USB Module

The arancino USB module allows you to interface the arancino and arancino mignon boards to a computer or to any device with a high-speed USB 2.0 port, bypassing the microcontroller.

The module can be powered at 3.3V from pin D5 of arancino Connector 1 or from pin D7 of the arancino Connector 2 on which it is mounted.



## Arancino Weather Recipe



Arancino Weather Recipe consists mainly of two RJ11 connectors used for connection to a weather station.

The rain gauge and anemometer of the weather station are connected to the RJ11 connectors.

Arancino Weather Recipe is powered with a continuous voltage of 3.3V directly from D5 pin of arancino Connector 1 or from D7 pin of Arancino connector 2 on which it is mounted.

## CONFIGURATION AND INSTALLATION

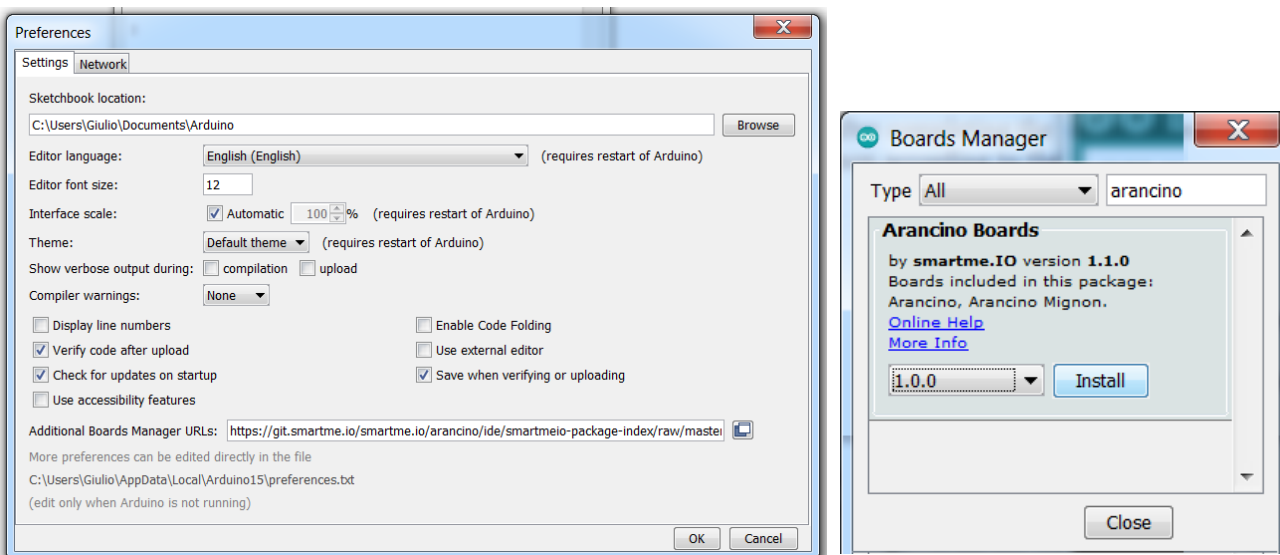
In this section we analyze the installation procedures necessary to the correct functioning and use of the arancino.cc and arancino.cc mignon boards.

### ARANCINO.CC INSTALLATION

To program the microcontroller of the arancino.cc and arancino.cc mignon boards it is necessary to use the development environment **Arduino IDE**, downloadable from [download page](#) of the official website arduino.cc, and then install it on your personal computer. The Arduino IDE development environment basically consists of an editor, a compiler, a loader and a serial monitor.

After completing the installation and starting the IDE, you need to install arancino.cc board or arancino.cc mignon according to the following procedure:

- File → Preferences → Additional Board Manager URLs
- add link: [https://git.smartme.io/smartme.io/arancino/ide/smartmeio-package-index/raw/master/package\\_smartmeio\\_index.json](https://git.smartme.io/smartme.io/arancino/ide/smartmeio-package-index/raw/master/package_smartmeio_index.json)



- from “Tools → Board: → Board Manager” wait for the download of the new Arancino Boards platform;
- typing "arancino" in the search bar at the top of the window;
- select and install the first item “Arancino Boards”, choose the desired version and press the “Install” button.

Once the installation is completed, the arancino boards installed in the IDE will be visible in the "board" section, as shown in fig.2.

## IMPORT THE ARANCINO LIBRARY IN ARDUINO IDE

**Arancino Library** is a library created for arancino.cc boards, written in Arduino Code programming language, which can be imported into the Arduino IDE development environment. The Arancino library is written to work also in SAMD21 microcontrollers, through the Arduino SAMD platform.

It uses the serial connection (through the SerialUSB library) to communicate with the Arancino Module, and through the Serial library it allows you to perform debugging operations.

**Arancino Module** is a daemon (background running program) that runs on the linux side of the arancino boards; it acts as an interface between the microcontrollers connected via serial port to the arancino.cc board, and can manage multiple connections at the same time. It is designed to run on the *Arancino OS* operating system, but can also run on Unix-based systems. It manages serial connections by default, but can be extended to other communication channels. Each connection must implement the *Arancino Cortex Protocol*.

The Arancino library allows you to import/export data to and from the Linux environment using the *Redis datastore* as database cache. In fact, the Arancino APIs are modeled on the standard Redis commands: arancino module interprets the API commands (begin (), get (), set (), ...) coming from the connected microcontrollers and sends them to the Redis datastore waiting for their execution.

To use the Arancino library it is necessary to download it in two ways:

### From a .zip file

1. Download the latest version of the **arancino-library repository**, a compressed file available in the section [SmartMe.IO Repository Management Site](#); it is possible to choose between two versions of libraries, the basic version “0.xx”, and the more complete version “1.xx”, which supports multitasking with FreeRTOS;
2. Open Arduino IDE and import the compressed folder (“Sketch → Include library → Add .ZIP Library...”).

The library will be visible at the end of the menu under “Sketch → Include Library”, under the section “Contributed libraries”. A collection of examples is also available under: “File → Examples → Arancino”.

### From Arduino Library Manager

1. Opening the Arduino IDE under “Sketch → Include Library → Manage Library”, the Library Manager window will be shown. By typing the word "arancino" in the text box, it will be possible to install the latest version of Arancino Library showed in the window.

## HOW TO FLASH THE ARANCINO.CC™ MICROCONTROLLER

1. Open Arduino IDE and select one of the arancino.cc boards connected to your computer, under the menu item Arancino Boards in the section: “Tools → Board: Arancino → Arancino Boards”:

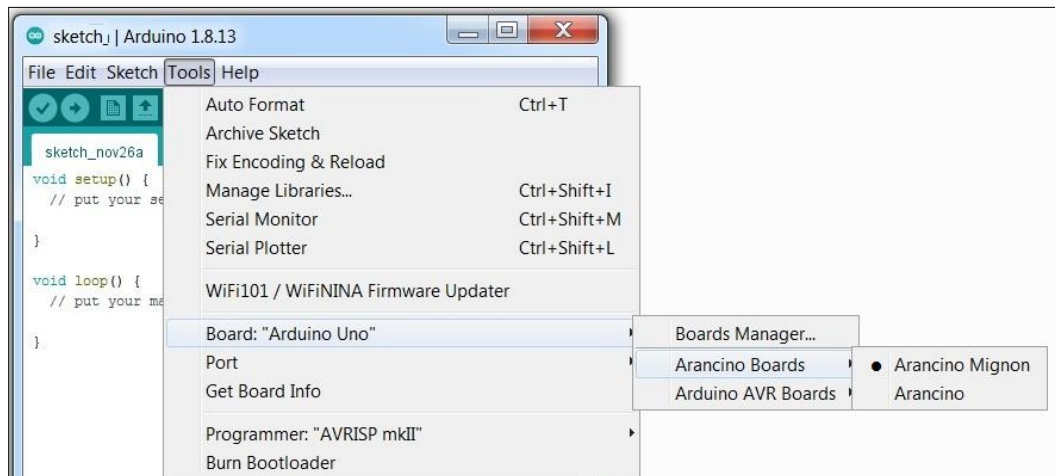
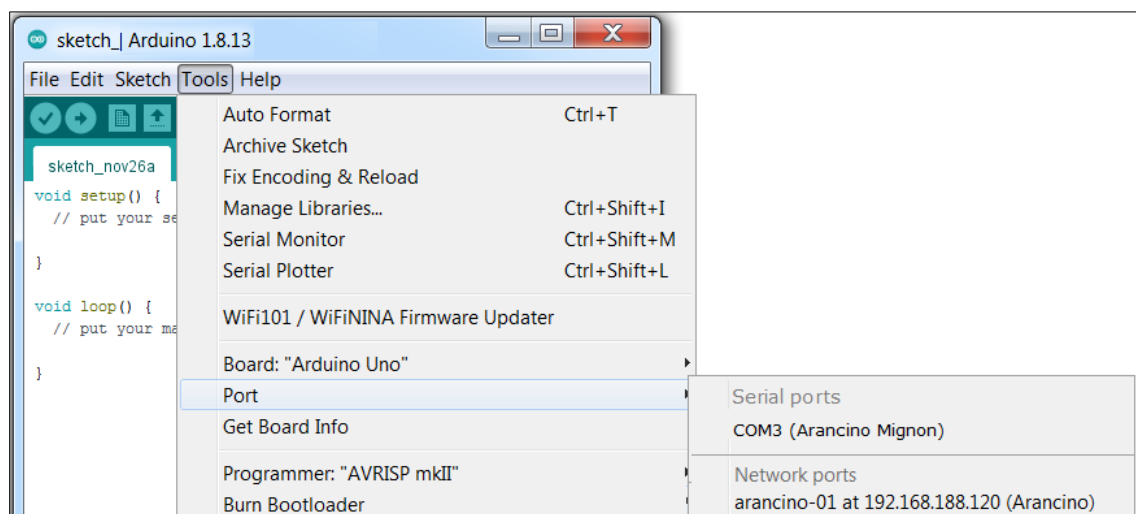


fig.2: arancino.cc boards installed in Arduino IDE

2. Based on the selected board:
  - a. **For arancino.cc™ Board:** with the Board connected to the network via a USB Ethernet or Wi-Fi adapter (in the latter case the network must be configured through the interface accessible once put in AP Mode by pressing the button on the board ), you can flash the sketch from the Arduino IDE under Tools → Port → Network Ports;



- b. For arancino.cc™ Mignon: select the USB port from “Tools → Port” and load the sketch.

## HOW TO FLASH ARANCINO.CC™ OS ON THE COMPUTE MODULE

The Compute Module used with the arancino.cc board is a **raspberry Pi CM3+** calculation module equipped with a 4GB eMMC memory module, which allows it to host the *arancino.cc™ OS* operating system. It can be flashed indiscriminately in Windows, Linux or MAC environment. Before flashing it is advisable to move the jumper placed in slot 5 of the board to “PRG”, install the raspberry drivers on your computer and use the **git** and **libusb** software.

In Windows environment, rpiboot software can be used to install the required drivers and boot tool, which can be downloaded from [https://github.com/raspberrypi/usbboot/raw/master/win32/rpiboot\\_setup.exe](https://github.com/raspberrypi/usbboot/raw/master/win32/rpiboot_setup.exe)

After the rpiboot installation is complete, a new USB mass storage drive will appear in Windows.

### Image writing:

- connect ArancinoHW
- install and run the utility [Raspberry PI Imager](#)<sup>1</sup>
- select the image of ArancinoOS
- select the drive related to the compute module
- start writing

At the end of the writing process turn off the board and move the jumper from PRG to RUN.

After writing the OS image, make sure J4 (USB SLAVE NOT ENABLE) is set to the disabled position and/or nothing is connected to the USB slave port. The IO board power cycle should result in the compute module booting from the OS image from the eMMC.

**For Windows and Linux:** follow the official Raspberry guide at the link:  
<https://www.raspberrypi.org/documentation/hardware/computemodule/cm-emmc-flashing.md>

### For macOS:

```
Install libusb using brew
brew install libusb
Install rpiboot (after installing libusb):
git clone --depth = 1 https://github.com/raspberrypi/usbboot
cd usbboot
make
sudo ./rpiboot
```

If the "Failed to claim interface" error is returned, first connect the ArancinoHW and then run `sudo ./rpiboot` again

---

<sup>1</sup> <https://www.raspberrypi.org/download/>

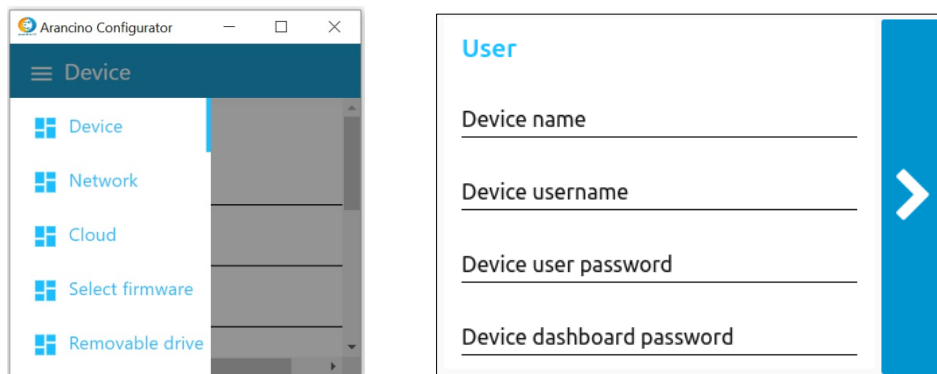
## ARANCINO CONFIGURATOR

**Arancino Configurator** is a software created by **smartme.IO®** useful to configure easily the parameters of an arancino.cc board equipped with a compute module.

First you need to download the software from the official website at:

<https://download.smartme.io/artifactory/webapp/#/artifacts/browse/tree/General/arancino-configurator>

When Arancino Configurator opens, a menu will appear with all configurable items: *Device*, *Network*, *Cloud*, *Select Firmware* and *Removable drive*.

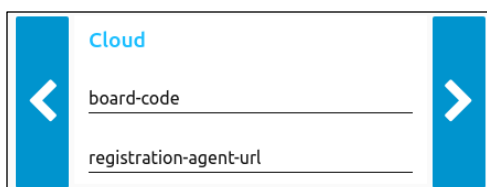
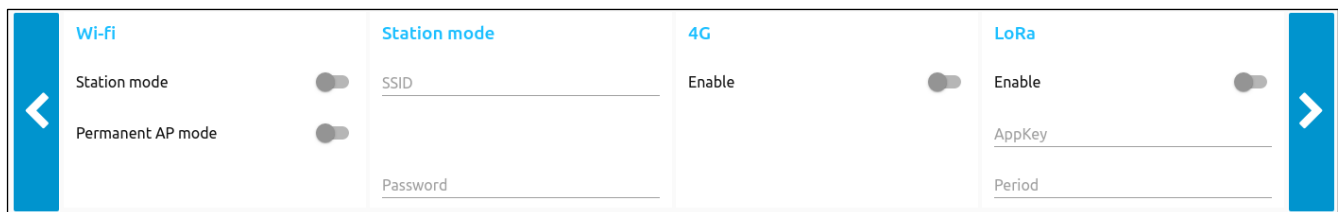


In the **Device** section it is necessary to enter:

- *Device Name*: name of the device with which the board can be reached within your LAN network;
- *Username*: username with which you can access the device console;
- *user password*: password to access the device console;
- *password dashboard*: password to be entered, together with the Device Name, in the web browser dashboard.

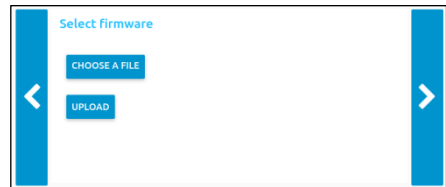
In the **Network** section it is possible to configure the desired network connection mode (Station mode, 4G or LoRa) by setting the relative parameters:

- *Station mode*: connects the device to the available wifi network;
- *Permanent AP mode*: to use the device as a Wifi Access Point, creating a wifi network to which other devices can connect;

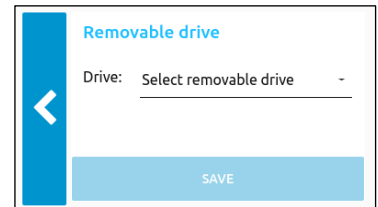


To configure the "board-code" and "registration-agent-url" fields it is necessary to request this information to smartme.IO by sending an email to the address: [info@smartme.io](mailto:info@smartme.io).

If you want to flash the firmware of the arancino.cc microcontroller, you can select the desired image file in the “Select Firmware” section and upload it.



Once all the required parameters have been entered, all that remains is to specify the path of the pendrive where to save the compressed file “arancino.conf.zip” containing the json configuration file of the arancino board.



Then it is necessary to power up the arancino.cc board and insert the pendrive with the configuration file created in its USB connector.

Inside the compute module of the arancino board there is a daemon that takes the configuration file from the pendrive, decompresses it by extracting the file “*arancino.conf.json*” and configures the work environment of arancino.cc board.

The configuration is complete when the compute module led starts blinking; after having extracted the pendrive from the USB port of the board, the system is restarted and arancino.cc is ready to be used.

## THE ARANCINO.CC FIRMWARE

The term "firmware" generally indicates the software installed on a protected area of a microcontroller (a Flash memory, ROM, PROM, EPROM) capable of controlling the correct functioning of the hardware of the entire system.

The **Firmware** of arancino.cc consists of the program that resides on the SAMD microcontroller, and it is divided into two different types of code:

- **bootloader**: the protected code executed after a system reset, capable of initializing the hardware and loading user-code into the microcontroller through serial/USB communication.
- **sketch**: the program written, compiled and transmitted by the user through the Arduino IDE installed on the computer.

Generally, the arancino boards are configured to program the microcontroller through the usb port of the pc. In the internal memory of the Atmel SAMD 21G18A microcontroller is usually pre-installed the bootloader needed to load the sketches.

## ARANCINO.CC BOOTLOADER

The bootloader is the code (a program) that is executed after a system restore of the arancino.cc board in order to initialize the hardware for the system boot.

The bootloader also allows you to load into the microcontroller, via serial communication, the sketches written in the Arduino IDE and compiled inside the flash memory of the microcontroller.

The bootloader resides in a secure, overwrite-free part of memory in order to preserve its integrity.

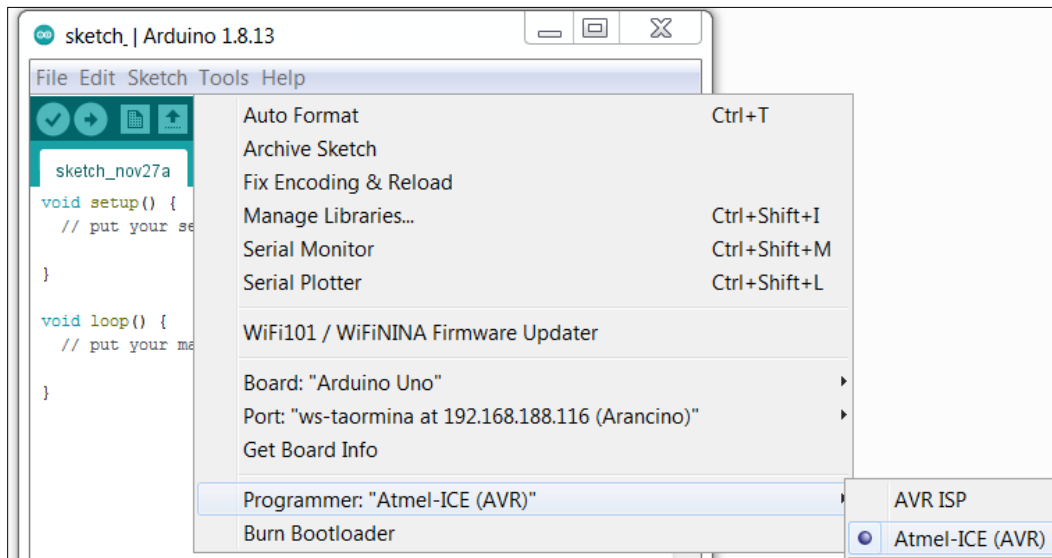
If after the board reset, the user needs to load a program (sketch written in Arduino IDE) from the computer connected through the USB port, the bootloader checks if the operation is allowed and manages it.

The SAMD21 microcontroller present on the arancino.cc board is generally supplied with a pre-installed bootloader; if the microcontroller is not flashed you must follow the procedure below.

## BOOTLOADER FLASHING

- Open the Arduino IDE
- Tools → Board → Board manager Search for "arancino" and if it is not present, follow the installation procedure described on page 17 ("[Arancino.cc installation](#)");
- Select the Arancino programming board
- Select the "Atmel-ICE" programmer under "Tools → Programmer AVRISP"
- Remove the cap from the JTAG connector (slot 5 of the arancino.cc board)
- Connect the programmer being careful to PIN 1 (indicated by a dot on the board, and by the red cable in the flat cable)
- Connect the programmer to the computer
- Power up the board
- From the Arduino IDE, select "Tools → Burn bootloader"





To verify that everything went well:

- Connect the Arancino board to the computer via serial link (see "UART (SAM3U)")
- Once logged in, type *lsusb* and check for the presence of the "Arduino SA" device with the following PID and VID:
  - Arancino Mignon: VID: 0x04D8 – PID: 0xECD9 – Product ID: "Arancino Mignon"
  - Arancino: VID: 0x04D8 – PID: 0xECDA ( visible only from Arancino OS side ) - Product ID: "Arancino"

## USE CASES

The arancino.cc architecture can be used in various fields of application, such as environmental detection, automotive, artificial intelligence, machine learning, neural networks, cloud, big data analysis, predictive maintenance. The arancino.cc™ architecture is equipped with the arancino.cc™ OS Open Source operating system, also developed by SmartMe.IO®.

### Environmental Detection System

The **SmartMe.IO SME-ES** environmental station allows the measurement of environmental parameters such as temperature, relative humidity, air pressure, air quality (PM1 / PM2.5 / PM10 particles) and weather (rain, wind direction and wind speed).

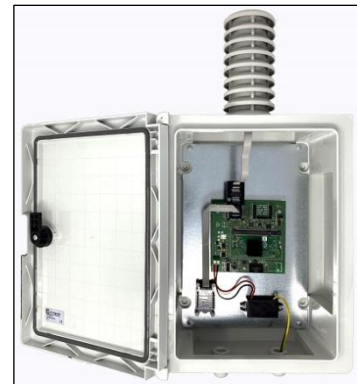
It is mainly based on the arancino.cc board equipped with the arancino.cc™ OS operating system, installed on the compute module mounted on the board. The SmartMe.IO SME-ES environment station is part of a Research and Development Project named [TOO\(L\)SMART](#).

SME-ES-01 uses WiFi, 3G / 4G / LTE and / or LoRa® (LoRaWAN™ Class A) connectivity,

The environmental survey station communicates with the outside via wifi or via LTE modem, connected to the USB port of the arancino board; the wifi connection can be made via wifi dongle or using a wifi click module connected to one of the two Arancino Connectors of the board.

The various necessary modules and sensors are connected to the arancino Connectors:

- add-on arancino weather recipe, consisting of two RJ45 ports to which the rain gauge and anemometer are connected to detect rain, wind direction and wind speed;
- LoRa module for long-distance radio frequency LoRa connectivity;
- atmospheric pressure, temperature and humidity sensor inside the control unit;
- sensors inside the Stevenson screen, used to detect external humidity and temperature and particulate data.



The data collected by the environmental survey station are sent to the smartme.IO cloud or the station manager and can be monitored by users by connecting to a dashboard via their device (computer, smartphone, tablet ...).

## Smart Camera

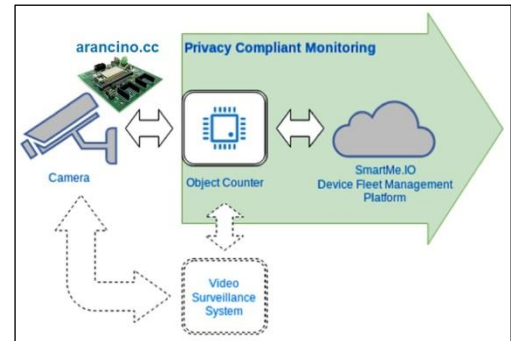
The Smart Camera created using the smartme.IO® arancino.cc™ architecture can be used in numerous fields of application: continuous monitoring of indoor and outdoor environments, counting of people and objects, density/congestion assessment for gatherings, queues, traffic, logistics, monitoring of road traffic and / or at traffic lights, gates, construction sites, and toll booths with dynamic identification of vehicle flows, monitoring for statistical purposes and evaluation of success for events in exhibitions, shopping centers/businesses, museums, conferences, showrooms, monitoring the use of personal protective equipment or risk situations in construction sites or industrial plants, complement to systems of Video-Surveillance.

Inside the smart camera there is an arancino.cc board equipped with a compute module (on which the Arancino.cc™ OS Operating System is installed), an arancino wifi module, an at-the-Edge processing unit and various devices for data transmission (wifi module, LoRa module, 4G usb token).

Through LoRa® connectivity, it provides as output a periodic report (5-30 minutes) with average, minimum and maximum measures of the period.

Through Deep Learning algorithms it processes the information received on the Edge from even pre-existing cameras, without recording or sending any image.

The smartmeIO® smart camera works as a new generation presence sensor, not limiting itself to signaling only the presence of objects, people or animals, but also their quantity (maximum, average and minimum number in the intervals) and, if required, their positions in the monitored scene.



## PicoGW

The PicoGW device is a gateway that operates on both cellular and LoRaWAN™ networks.

Its operation is based on the smartme.IO® arancino.cc™ architecture; inside the PicoGW there is an arancino.cc board on which a MIPS @ 580Mhz microprocessor is mounted (with 32Mb flash memories, 128Mb ram and microSD card), two arancino wifi modules, an embedded GNSS geolocation module, a LoRa module spread spectrum FSK GFSK with 860Mhz-1020Mhz frequency range, a 10/100 ethernet port and a 1T1R 802.11 b/g/n (2.4G) wifi module.



The smartme.IO® PicoGW is used in many fields of application: Internet Of Things (IoT), automatic meter reading, smart-cities, home and building automation, wireless alarm and security systems, machine to machine (M2M), industrial monitoring and control, systems long range irrigation.

## Motorsport

Arancino.cc technology is also used in the motorbike environment; thanks to the arancino.cc™ technology, the [“SIC – Stretto In Carena”](#) MotoStudent’s team deployed an advanced motorbike dashboard mounted on a racing motobike. Arancino.cc is used for telemetry, the geo-localization of the bike on the track, the management of the graphic dashboard and the real-time communication with the engine control unit (Engine control Unit ECU). The data acquired by the sensors in a specific instant of time and in a certain position of the motorcycle on the track, are processed in real-time and displayed on the dashboard display, in order to be consulted by the rider.



## Connected machines for intelligent manufacturing plants



Arancino.cc helps industries to make production processes “intelligent”, that is, safer and more sustainable, allowing the collection of data from a large number of IoT machines, tools and sensors. In this way, intelligent and interconnected production processes are created via WiFi 802.11 b/g/n (2.4G), Ethernet, LoRa®, 3G/4G/LTE and GEO location with incorporated GNSS connection. The resulting benefits are mainly evident in the automation and implementation of new technologies such as machine learning, artificial intelligence and cognitive analysis.

By accessing the services based on arancino.cc it will be possible to collect and process large amounts of data (big data), also bringing the artificial intelligence algorithms directly to the devices, especially in environments with network connection limits. This allows for real-time monitoring, ensuring safety and reliability. Arancino.cc has been implemented on an industrial machine with a conveyor belt in order to create a monitoring and prediction system capable of predicting possible malfunctions (such as breaking of the conveyor belt) of the machine.



## CONTACTS

SmartMe.IO Srl  
Via Osservatorio, 1  
98121 Messina (ME) - ITALY

Lab:  
SS 114 km 5,400  
c / o Today Multifunctional Center  
Tower B - floor 2 - 98125 Messina (ME) - ITALY

Engineering Department, C / da Di Dio, 1  
98166 S.Agata Village, Messina (ME) - ITALY

REA number: ME238676  
VAT / Fiscal Code: 03457040834  
Phone: +39 0907385454  
E-mail: [info@smartme.io](mailto:info@smartme.io)  
Web: <http://smartme.io/>