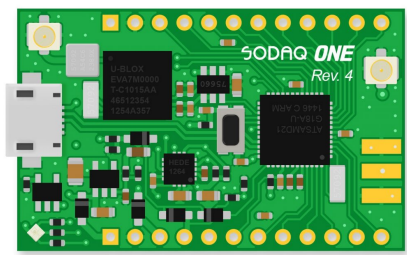


Overview



Version 2 of the SODAQ ONE is coming soon! See the differences [here](#)

This board is based on our succesful [Autonomo](#), a proven 32 bits Arduino compatible platform. It has a solar charge controller and runs on a LiPo or permanent battery.

Next we added a GPS module. We chose the U-blox Eva 7M. It's not only very small, but with it's assisted GPS feature it can get a fix in just seconds.

We also included a low power Accelerometer/Magnetometer chip. This means we can keep the board in sleep until it moves or has a change in magnetic field. This is crucial for developing low power devices.

Say you want to develop a bicycle tracker. You would like know the position of the bike, but only when it has moved. So keep the device in deep sleep until you detect motion. If the motion continues for a while, the bicycle may have changed position so you only then switch on the GPS to get a new reading and send this new location over the LoRa network. This way you make most efficient use of you battery capacity by only using the GPS when really needed.

The Magnetometer extends the range of possibilities. You could now use the board to develop a parking sensor (fit the device in the street, if a car is parked this changes the magnetic field). Or you can mount it to a fence and use the compass direction to determine if the gate is open or not.

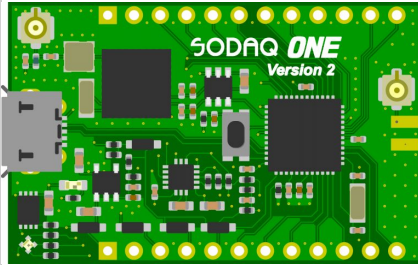
The board is so small (40x25 mm) you can fit it inside almost anything. To ease the development we have fitted two rows of headers so you can use it on a breadboard. Why? Well it has 14

free I/O lines that you can use for your own purpose. Whether you want to build a weather station, control street lights or get a signal when you have received (snail) mail. It's all possible with the SODAQ ONE.

Here's an overview of the features:

Microcontroller	ATSAMD21G18, 32-Bit ARM Cortex M0+
Compatibility	Arduino M0 Compatible
Size	40 x 25 mm
Operating Voltage	3.3V
I/O Pins	14, All can be used for digital and analog with PWM, UART, SPI and TWI (I2C)
Analog Output Pin	10-bit DAC
External Interrupts	Available on all pins
DC Current per I/O pin	7 mA
Flash Memory	256 KB
SRAM	32KB
EEPROM	Up to 16KB by emulation
Clock Speed	48 MHz
Power	5V USB power and/or 3.7 LiPo battery
Charging	Solar charge controller, up to 500mA charge current
LED	RGB LED
LoRa	Microchip RN2483 Module
GPS	uBlox EVA 7M
Accelerometer/Magneto	LSM303D(ONE V1) / LIS3DE(ONE V2)
USB	MicroUSB Port

ONE vs ONE v2



The components we use to make the ONE v2 are more power efficient.

The GPS module we updates from the U-blox EVA 7M to the U-blox EVA 8M.

The accelerometer/magnetometer has been replaced by an accelerometer which can be used in a very low power mode and wake up the board by an interrupt.

The RF path is shorter, this results in better antenna performance. To make the RF path shorter we have removed the line for the 2nd frequency, we don't support the 433Mhz anymore.

We added a LoRa reset line from the microcontroller to the RN module. If the RN module isn't responding anymore, the microcontroller can do a hard reset to the RN module.

The SODAQ ONE and ONE V2 are pin compatible!

The Universal Tracker software is compatible with V1 and V2.

Summary

ONE	ONE v2
GPS EVA 7M	GPS EVA 8M
LoRa 868/433 or 915	LoRa 868 or 915
Accelerometer and Magnetometer (LSM303D)	Low power accelerometer with interrupt lines (LIS3DE)
No lora reset line	LoRa reset line

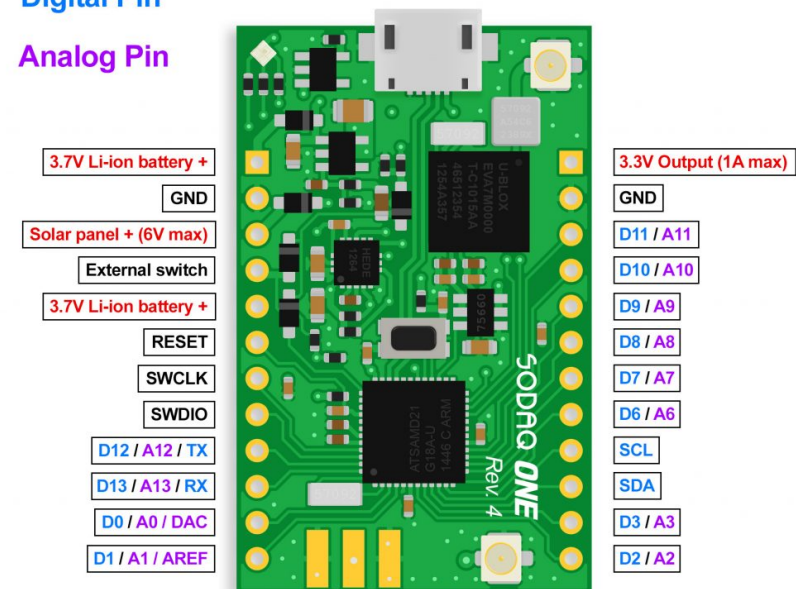
Getting started with the SODAQ ONE

To start you off, here are the external pin descriptions of the SODAQ ONE. Every header pin will be described here, so you know exactly how to address the pin in the Arduino IDE. *(click to open in a new window)*

Power connection

Digital Pin

Analog Pin



External pin descriptions will get you started with the basics, but to fully utilize the board, take a look at the internal pin descriptions. These will allow you to use all the functions that are built into the SODAQ ONE. *(click to open in a new window)*

LoRaONE internal pin descriptions

GPS Switch D18 OUTPUT

Set this pin to HIGH to turn the GPS module on.
Set this pin to LOW to conserve energy.

GPS Timepulse D17 INPUT

This pin will turn HIGH every second, synchronized to GPS time signal.
When there is no GPS fix, this pin will not provide accurate timing.

User Button D19 INPUT

This pin is high by default. Pin will be pulled low when its pressed.

Power Enable D22 OUTPUT

When running on a battery, and without USB connected,
the regulator that supplies power has to be enabled manually.
This can be done with this IO pin, set it to HIGH to enable power.

External Switch D23 INPUT

The LoRaONE supports an external switch (such as a simple button)
that can be connected between the battery and the external switch pin.
When the switch closes, you will be able to sense this on pin D23.

LED D14 D15 D16 OUTPUT

Set a pin to LOW in order to turn on a LED,
setting a pin to HIGH turns the LED off.

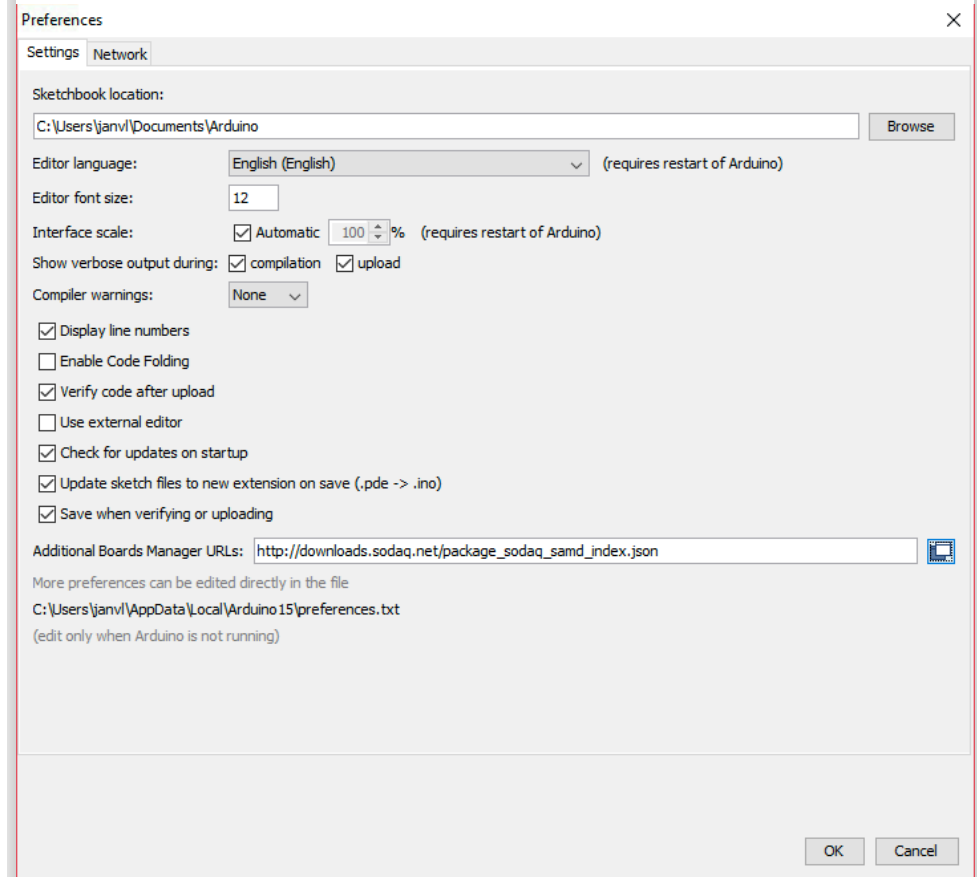
Setting up your Arduino IDE

In order to start programming for the SODAQ ONE, make sure you have the latest Arduino IDE. [Click here to go to the Arduino downloads page.](#)

When your Arduino IDE is installed, it might be necessary to restart your computer. In some cases, if you don't restart your computer, the IDE will not launch.

When your IDE is all started, we should install the SODAQ ONE board files. Click on *File > Preferences* and at the bottom you should see 'Additional Boards Manager URLs'. This is where you need to paste the following URL:

```
http://downloads.sodaq.net/package_sodaq_samd_index.json
```

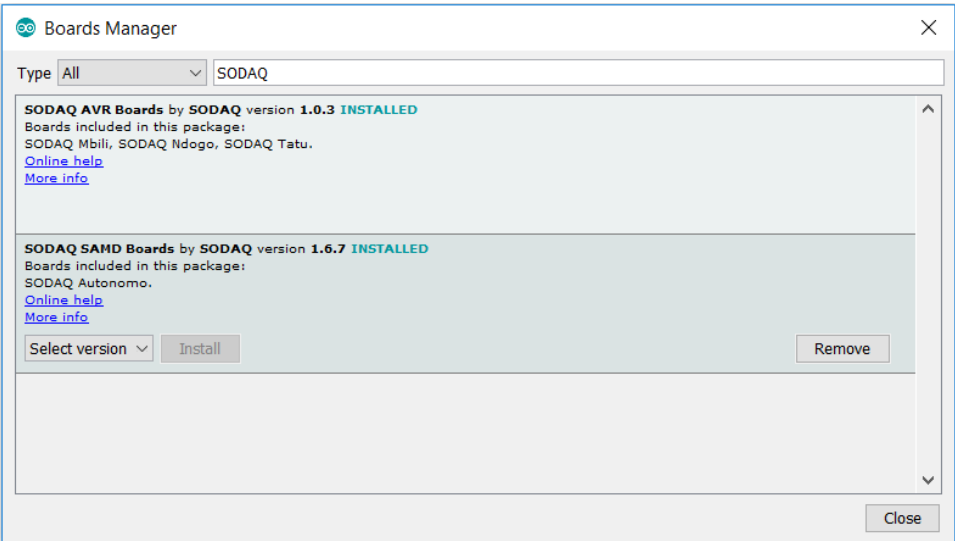


When you have pasted the URL, click 'OK' and you're ready for the next step.

Click on *Tools > Board > Boards Manager...*

Scroll all the way to the bottom, you should see **SODAQ SAMD Boards**. Click on it, and install

the latest version. It should now download some files and install the required board files for the SODAQ ONE.



We’re almost there! Now go to *Tools > Board* again and select the *SODAQ ONE* at the bottom. You’re now set up to start programming with the SODAQ ONE!

Handy pin definitions!

Now that you’re using the SODAQ ONE board files, you’ll be able to use our handy pin definitions. Let’s say you want to use the red LED, but you forgot that pesky pin number, you can just address the led with *LED_RED* inside your Arduino IDE. Our pin definitions allow you to address by name instead of pin number.

Here are all the pin definitions for the SODAQ ONE:

Pin description	Pin number	Definition
Red LED	D14	LED_RED
Green LED	D15	LED_GREEN

Blue LED	D16	LED_BLUE
Power Enable	D22	ENABLE_PIN_IO
GPS Enable	D18	GPS_ENABLE
External Switch Sense	D23	SWITCH_SENS
Push Button	D19	BUTTON
GPS Timepulse	D17	GPS_TIMEPULSE
MISO (SPI)	D8	
SS (SPI)	D9	
MOSI (SPI)	D10	
SCK (SPI)	D11	

Base Orientation and antenna connection

If you want to use the pins: 2/3, 6/7 and 8/9 you first have to set pin D11 high.

RX / TX	Serial
Enable power on 2/3 6/7 8/9	D11
Power On / Off*	Left / Right

*note: The on/off switch only works when there is no usb connected and the switched pin is not set high in software

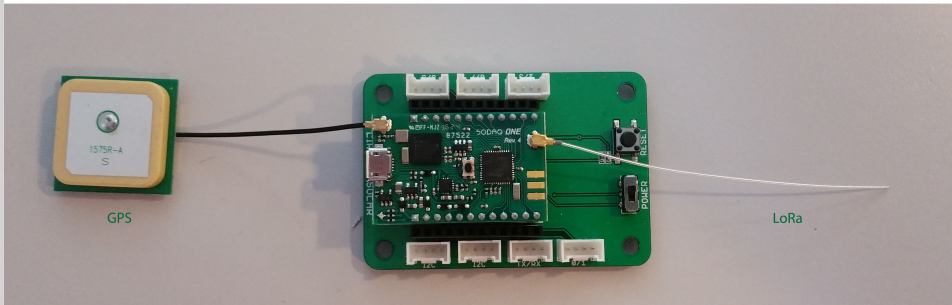
```
#define enablePin 11

void setup() {
  // put your setup code here, to run once:
```

```
pinMode(enablePin, OUTPUT);
digitalWrite(enablePin, HIGH);
}

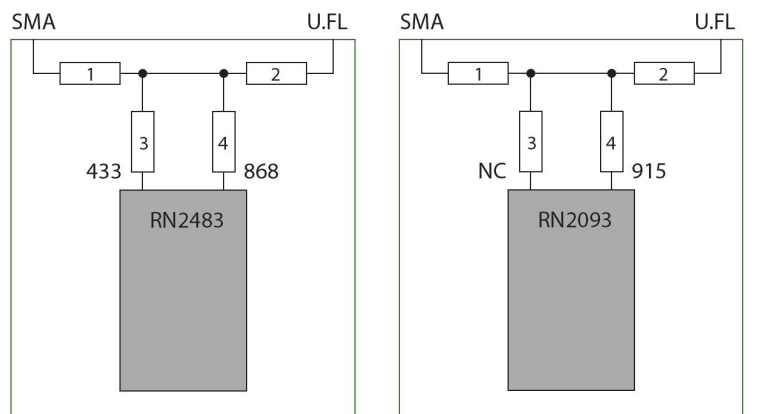
void loop() {
  // put your main code here, to run repeatedly:

}
```



In the first batch all zero-ohm resistors were soldered on. The SODAQ ONE works fine, but we found that you can improve the antenna performance with only two zero-ohm resistors (by removing 1 and 3).

Schematic of the resistors:



Check the video about how to remove the resistors:

SODAQ ONE removal of resistors to improve signal strength



Bootloader mode

To save power it is possible to disable the usb connection. In the tracker code the usb is disabled after the boot menu, this means you only have 30 seconds to upload a new sketch. If the reset button (base board) is pressed twice within a second the current sketch will not start and the SODAQ ONE will go into bootloader mode and is expecting a new sketch. On your computer you will see a different com port.