User guide

Principal Author/s	D Buma T Telkamp	-
Checked	Lacuna Space	-
Released	M Kalama	20th Oct 2020

Table of Content

1 Introduction

1.1 Welcome

Thank you for taking part in the Lacuna Space Technology Partner Programme, our first phase of third party trials open only to partners and supporters of the system. We hope you will be as excited as we are about the tremendous opportunities direct to satellite IoT technology offers.

Being part of the Lacuna Technology Early Access Partner allows you to get acquainted with the satellite connectivity services offered by Lacuna Space The program is intended for companies and individuals wishing to develop products and services based on Lacuna's connectivity services. With this document we hope to offer you a smooth introduction to our tools and services.

In this document we provide an introduction to the software (libraries) and tools we provide for make the use of the Lacuna Satellite Network as simple as using any terrestrial network.

In chapter 2 of this document we also provide a basic background on the key principles behind the operation of the Lacuna Satellite Network. The software provided will automatically extract most of the details of communicating with the satellite network. So although it is not mandatory or needed for basic operations as, we do think it will help you improve your understanding in the key opportunities and applications of our technology.

Please keep in mind that all information shared within the partner programme, as long as it has not been publicly released yet, is subject to the Non-Disclosure Agreement signed between your company and Lacuna Space Ltd.

We do welcome that you share your experiences on social media and are open to joint promotion, but we kindly ask you to first check with us before making anything public. For example in this phase we do not publicly release the source code yet and we prefer to keep confidential specific configuration data and performance measurements like signal strengths.

We hope you enjoy the ride and we very much welcome your feedback.

Best regards from the Lacuna team.

1.2 About the Lacuna Network

This document is intended as a guide for companies looking to develop solutions that incorporate Lacuna Space's satellite network. Our satellite network will extend the reach of conventional LoRaWAN® networks to areas with poor or non-existent connectivity. The Lacuna network is designed to work seamlessly with terrestrial networks but can also be used as stand alone. Lacuna Space has been developing its satellite based network since 2016 and has been deploying satellites that form gateways on the network since April 2019.

Although the network is not yet open to commercial service, we do provide stable global coverage for product development, testing and field trials. As more satellites will be added to the constellation, more functionality will be introduced to the network, including time and orbital parameter (almanac) updates to the devices using the Lacuna satellites.

1.3 About LoRaWAN®

LoRaWAN® is a Low Power Wide Area Network (LPWAN) protocol designed to send small messages at low data rates over very large distances using very low-power, normally using battery powered devices.

The physical layer of LoRaWAN® is implemented through low-cost off-the-shelf radio chips. Numerous organizations have developed such modules and these can be purchased through many outlets. Many sensors using LoRaWAN® and devices are available as commercial products to buy off-the-shelf.

1.4 Intended use and audience

This guide is intended for the beta testers of the Lacuna Network. It is assumed that partners accepted onto the Technology Partner Programme have basic working knowledge of embedded software design and the Arduino development platform and are familiar with LoRaWAN®, the networking protocol developed by the LoRa Alliance®. This document serves as a basic introduction to the key concepts and is complemented by the Lacuna user forum. On the user forum you can can get support and exchange experiences with other participants in the partner programme .

2 Lacuna System overview

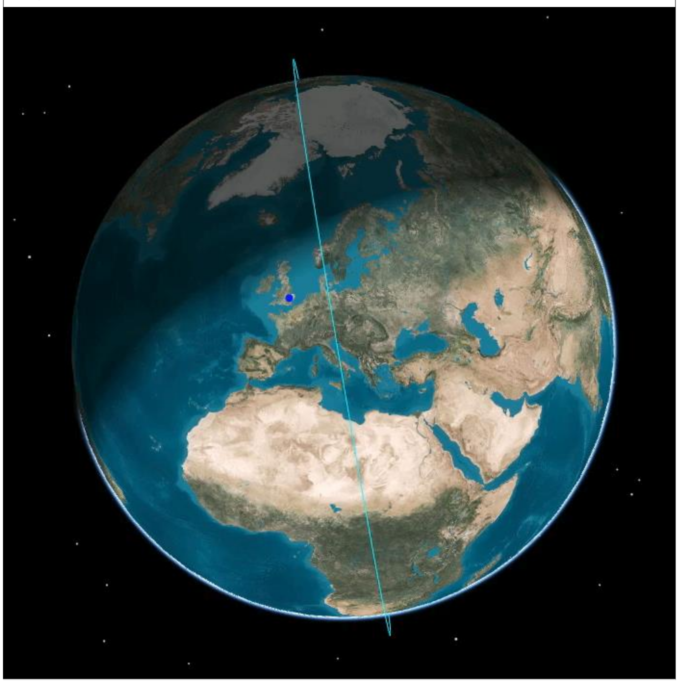
2.1 Introduction

In this chapter we provide you with the key concepts of the operation of the Lacuna Satellite Network. As mentioned before the software (libraries) and tools we provide will extract most of the specifics of using satellite communications compared to using terrestrial networks. We think it will help you improve your understanding in the key opportunities and applications of our technology

2.2 Satellite Network

Our satellites are so called Low Earth Orbit (LEO) satellites. The satellites are 500km above the surface of the Earth in an orbit that for most satellites runs over the North and South pole (polar orbit). Each satellite orbits the Earth in about 100 minutes which equates to about 15 orbits per day. As the Earth rotates during the orbits of the satellites, the orbits eventually map out the whole of the globe with no blank spots.

Figure 1: Polar orbit

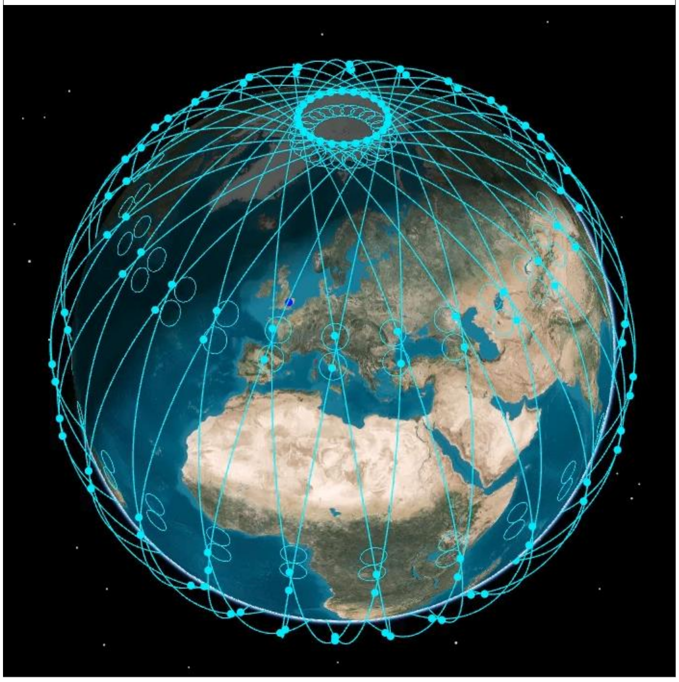


The revisit times, that is the time between each view of the satellite from a fixed position on the Earth (e.g. Lacuna's head office is shown in figure 1 as a blue dot), is among other things dependent upon the total number of satellites, their spacing and the size of their footprint. The footprint or the total ground area that a Lacuna satellite can see at one moment in time and is approximately 550 kilometres in diameter (see figure 2).

Figure 2: Satellite footprint

The coverage is better (lower revisit times) at the poles than at the equator where the satellites are spread out the most. Near the poles the tracks that the satellites cover with their footprint will overlap, whereas near the equator this will not be the case. The revisit times are becoming lower as more satellites are deployed and the user forum provides the latest updates on satellite launch and service start dates.

Figure 3: Multiple satellites in orbit



For the partner program we will primarily be making use of Lacuna's LS1 test satellite. In practice this means that for example in Northern Europe it is possible to transmit data every 2-3 days. This is also due to the current limitation that in Northern Europe the (North to South) passes are used for testing of the spacecraft itself. Other satellites do not have this limitation.

The approximate revisit times for LS1 for a number of locations are given in table 1. The column expected latency gives the expected time between time of transmission of information by a device/node and the delivery of this information to the customer backend (this parameter will be described in more detail in paragraph 2.3).

It is good to be aware that the pattern of the revisit times depends on the specific satellite orbits and do not work like a regular train schedule. Instead of having a pass at the same time each day, we know that the next satellite will pass within a given amount of time after the current pass. Exact satellite pass predictions for LS1 satellite for a specific location can be obtained with the Lacuna User Dashboard to which you should have access to or other publicly available online tools such as https://www.n2yo.com/passes/?s=44109.

Location	Approximate revisit time	Expected latency
Europe	2-3 days	6 hours
North America	1-2 days	11 hours
Australia	15-20 hours	30 mins
New-Zealand	8-12 hours	8 hours

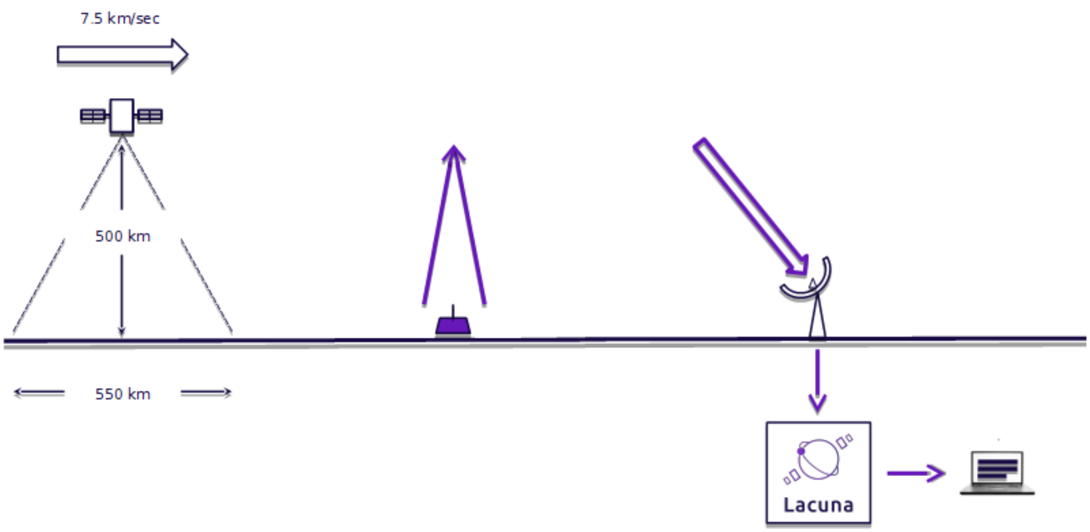
Table 1: Expected revisit times for LS1 trials

As the number of satellites in the Lacuna Network increases over time the revisit times will decrease accordingly to approximately every 6 hours and every 4 hours.

2.3 Principle of operation

A simplified diagram of the principle of operation is shown in figure 4.

Figure 4: Principle of operation



In the figure the satellite travels from left to right with a speed of 7.5 km per second. At a certain time the device node (purple box) will be inside the footprint of the satellite. As soon as this is the case the device node can start its transmission. When the satellite receives the information sent by the device node it temporarily stores it until it passes a ground station.

The satellite then transmits the stored information to the ground station, which in turn forwards it to Lacuna, which finally delivers the traffic to the customer. The direction of travel of information is from device node to satellite. It is (currently) not possible to transmit from satellite to device node (with the exception of Lacuna network specific service information).

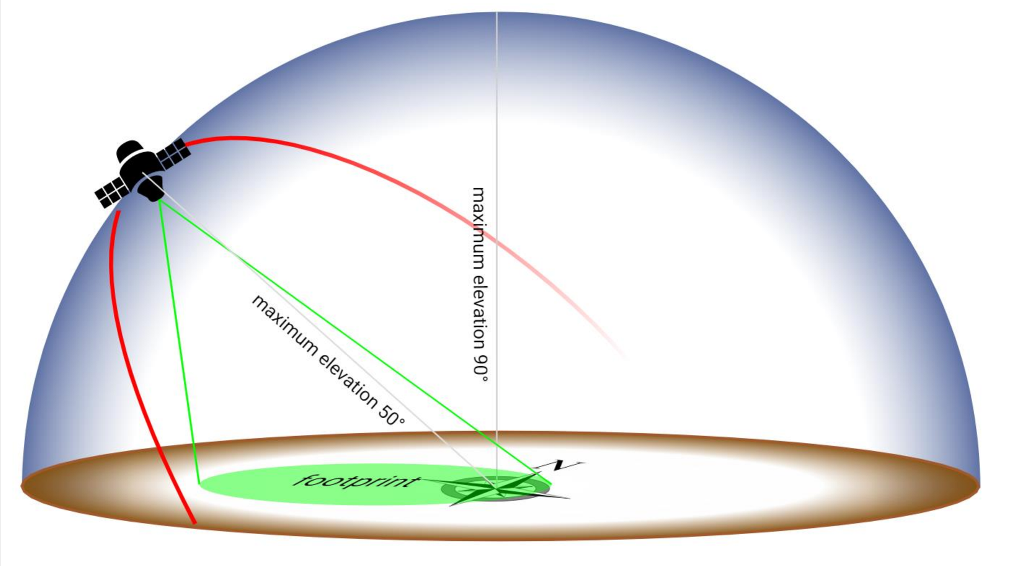
The latency between the transmission of information by the device node and reception by the customer is determined by the relative position of the device node with regard to the ground station(s). The further the satellite has to travel before it passes over a ground station the higher the latency will be.

During the current phase, for LS1, we make use of a single ground station located in Aalborg, Denmark. Expected latencies during the trial are given in table 1. As we continue to deploy new satellites we will also increase the number of ground stations in such a way as to minimize the average latency

2.4 Satellite pass elevation

For sake of simplicity figure 4 in the paragraph 2.3, showed the movement of the satellite in 2 dimensions. In reality the satellite during most cases will not pass straight over a device node (maximum elevation 90 degrees), but will rather be passing at a lower maximum elevation angle. This is shown in the figure below.

Figure 5: Satellite pass elevation



Seen from the device node a satellite will show up somewhere on the horizon (elevation 0) rise to a maximum height in the sky (maximum elevation) and disappear behind the horizon again (elevation 0). This cycle is very similar to sun rise – noon - sun set, except in this case it is not the earth orbiting the sun but the satellite that is orbiting the Earth. The most optimal situation is when the satellite passes at maximum elevation of 90 degrees (in other words it passes right over our heads).

This means that the antennas of both the device node as well as the satellite are perfectly aligned and the distance between the satellite and the device node is shortest possible. In practice most passes will be at a lower maximum elevation or not be visible above the horizon at all. A lower maximum elevation, means less optimal alignment of the antennas which in turn results in higher power losses. Based on our early trials with the current LS200 reference design, we have seen that in practice we need a pass with a minimum elevation of approximately 50 degrees for reliable transmission. This is no hard limit as we have in some situations picked up transmissions at elevations as low as 30 degrees. In less optimal situations the maximum elevation limit might be higher in case there is obstructions in the signal path such as buildings.

For future satellites this practical elevation limit is expected to be set lower as we are using improved antenna designs.

2.5 Transmission window

Because of the high speed at which the satellites are traveling, the window in which we can transmit information to the satellite is fairly short. The transmission window width depends on the elevation of the pass and ranges approximately in between 30-90 seconds. If we transmit outside of the transmission window the satellite will not be able to receive the transmission.

The transmission of a single message depends on the number of bytes contained in the message. This means that the transmission window is usually large enough to transmit multiple messages to the satellite during a single pass.

The most optimum time to send a message would normally be at the middle of the window as the satellite is at its maximum elevation. In practice however it is better to spread the transmission(s) within the transmission window.

In order to determine the transmission window of a satellite we need to know the location of the device node relative to the path travelled by the satellite. The former can be obtained by means of a GPS receiver. The path travelled by the satellite can be derived by feeding its orbit parameters (e.g. speed, inclination, etc) along with the current time into an algorithm. The algorithm we currently use for this at Lacuna Space is called SGP4. The orbit parameters are defined by a so called Two Line Element set (TLE).

We will not go into further detail here, but it is important to mention that it is recommended that the TLE information is updated every 1-2 months for optimum transmission.

In the current stage of the partner programme the TLE information is either updated online in the source code or via a terrestrial LoRaWAN connection to the things network.

Lacuna will soon provide a solution to distribute the TLE information updates directly from its satellite network. The TLE information will be broadcasted making use of the same protocol, therefore no hardware changes are needed to existing device nodes for this service.

During the partner programme there is no limit on the number of messages that can be send during a pass. The exact moment of transmission within the transmission window is not regulated. The example software provided with the LS200 reference design kits sends three messages during each pass. One message is sent exactly in the middle of the transmission window, also called Time of Closest Approach (or TCA). The other two messages are send equally spaced before and after the TCA.

Once Lacuna's commercial operations start, the amount of messages and the exact time to transmit will be governed by a software algorithm provided by Lacuna. The software algorithm will provide the application information on the intervals at which to transmit. Currently we are still investigating the type of services we will be offering, any input from the market is more than welcome.

3 LoRaWAN ecosystem

3.1 Introduction

As mentioned before, the Lacuna Satellite Network seamlessly integrates with the existing terrestrial LoRaWAN ecosystem. The satellites of the Lacuna Satellite Network can be compared to gateways in the sky. Device nodes can directly transmit to a satellite using the same hardware (with exception of the antenna) and the same low power requirements as they would transmitting to a terrestrial gateway.

Although there many similarities there are also differences between terrestrial and satellite LoRaWAN of which the most significant are:

- satellite LoRaWAN is unidirectional , i.e. device nodes can only transmit (uplink) to the satellite
- the latency is higher, as the messages transmitted by a device node are stored until the satellite passes a ground station
- the device node can only transmit when a satellite is passing over it

There is also a difference in the way the received data is processed in the backend. Where terrestrial gateways forward all their data to a network server, in the case of satellite LoRaWAN the data is first forwarded to the Lacuna backend which in turn forwards the data to the customer. The forwarding is in this case more intelligent than would be the case with a terrestrial gateway. The 'Lacuna gateway' performs certain functions that would normally be executed by the network server of which the most important is routing based on device address (range).

To simplify access for users, during the Lacuna Partner Programme data is generally forwarded to The Things Network (TTN), unless a different arrangement is established with the partner. In order to receive the data sent by the device node a free TTN account has to be setup (an existing account can also be used). Feel free to contact Lacuna to discuss possibilities about connecting to different servers.

3.2 Creating a device on TTN

For the purpose of testing the Lacuna Satellite Service during the partner programme it is necessary to setup a free account with The Things Network (TTN). The TTN account is used for registering the device node and creating an endpoint for delivering the transmitted data. In case you already have an account you can just skip this step and create an application for the device. An application serves as an endpoint of the data for one or more devices.

The Things Network provide detailed Documentation Guides and a series of How-To videos on YouTube.

https://www.thethingsnetwork.org/docs/

https://www.youtube.com/watch?v=JrNjY-pGuno&list=PLM8eOeiKY7JVwrBYRHxsf9p0VM_dVapXl

https://www.youtube.com/watch?v=JrNjY-pGuno

1. Create an account on the TTN

Go to: https://account.thethingsnetwork.org/register and create an account on the TTN.

2. Create an application

After following the steps in the registration procedure log in the TTN.

Next click on "Applications" in the toolbar on top and click on the "add application" button.

In the form that appears fill in a name for the "Application ID" field (eg. "lacuna_test") and add a description in the "Description" field. Leave the other fields in the form unchanged.

Next click on the "Add application" button.

3. Create a device

Click on "Applications" in the toolbar on top and click on name of the application created in the previous step. Next click on the "register device" button.

In the form that appears fill in a name for the "Device ID" field (eg. "lacuna_test"). Click on the button in front of the Device EUI field, which should now show "this field will be generated". Leave the other fields in the form unchanged.

Next click on the "Register" button.

4. Configure device

After registering the device, the TTN console will automatically direct to the device overview of the device just created. In this overview press the "Settings" button (next to "Overview" and "Data").

In the "settings" overview scroll down to "Activation Method" and click on the "ABP" button. Next uncheck the checkbox next to "Frame Counter Checks" Leave the other fields in the form unchanged.

Next click "Save".

5. Enabling storage integration

Click on “Applications” in the toolbar on top and click on name of the application just created. Next click on the “Integrations” button in the toolbar in the top of the screen.

In the “Integrations overview” click on the “add integration” button.

In the “Add integrations” overview click on the “Data Storage” button.

Next click the “Add integration” button.

If everything went ok we have now configured a device in TTN. The device is configured as an Access By Personalisation (ABP) type device, as for Over The Air Activation (OTAA) a two way connection would be necessary. The frame counter checks are disabled to facilitate debugging. In production framecounter checks should be enabled to prevent network replay attacks.

4 LS200 reference design kit

4.1 Introduction

As part of becoming a partner in the Lacuna Technology Partner Programme you have received the LS200 reference design kit. The LS200 is intended for giving partners hands on experience with the Lacuna Satellite network and servers as a reference for device manufacturers to create or adapt their own hardware for operation with the Lacuna network.

You can modify the LS200 reference design kit to your own wish and connect your own 3rd party hardware to it. We do however recommend first playing around with the kit and the software provided - as is - before making changes to it. This will help you familiarise with the specifics of the Lacuna Space Network and give you a good start for further exploration.

4.2 Description

The blue case you receive with the Lacuna Technology Partner Programme contains the LS200 reference design board with a 18650 rechargeable Li-ion battery. Both are attached to the case with Velcro tape, and can be removed and reattached. The case is waterproof (IP65) so it can be used outdoors.

The LS200 reference design board is shown below and is built around an STM32L476 MCU and an Ebyte E22-900M22S Transceiver. Also included with the kit is either a specially designed Lacuna Antenna from RFThings or an equivalent Qube 6010 antenna.

Figure 6: LS200 reference design

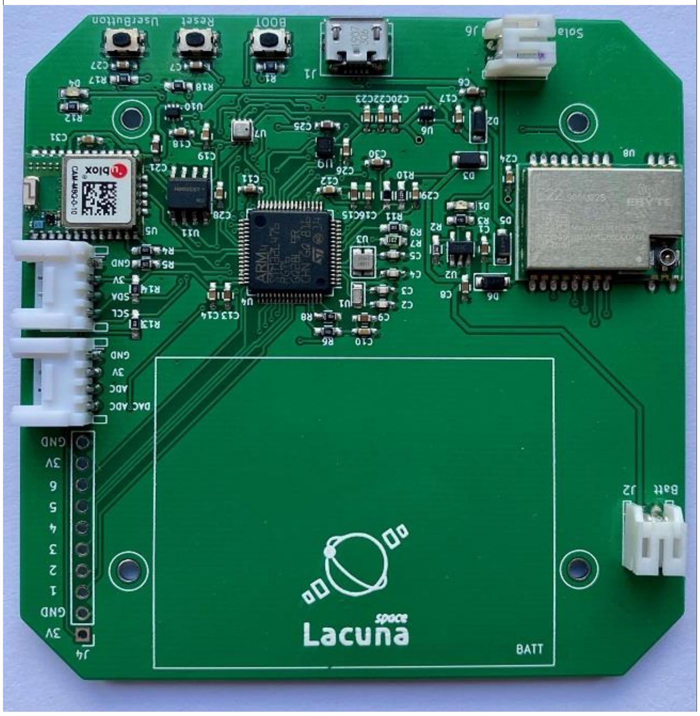


Figure 7: RFThings Lacuna antenna

The design contains a number of sensors and peripherals amongst which:

- a U-blox CAM-M8Q GPS receiver
- a BME280 combined temperature/humidity/pressure sensor
- a LSM303 combined triple axis accelerometer/magnetometer

- a ATECC608A secure element

The kit also offers a number of expansion connectors for connecting 3rd party hardware:

- a Grove I2C standard connector
- a Grove Analog standard connector
- a connector for connecting a rechargeable 18650 Li-ion battery
- a connector for connecting a solar Panel
- a break out port with additional 4 wire SPI bus and serial port

Of all the hardware used in the reference design only the transceiver chip and the directional antenna (which should be right hand circular polarized and have a gain of at least 2.1 Dbi) are mandatory. All other parts are optional for communicating with the Lacuna Satellite Network. Currently a GPS receiver is mandatory for non-stationary devices but this limitation will be removed in the future. Feel free to contact is for more information on this.

4.3 The software development

The toolchain you will use for developing your own software for the LS200 is based on the Arduino architecture as is easy to use for novice users and is readily available. To accommodate setting up the toolchain an Arduino board file for the LS200 hardware is also included. The libraries provided with the LS200 reference design kit contain all functionality needed to send messages to the Lacuna Satellite Network as well as to any terrestrial LoRaWAN network. Several examples are included with the library for easy testing with the Lacuna Satellite Network. In this description we assume you are within coverage of The Things Network (TTN) as examples also make use of the terrestrial network.

This is a beta hardware and software release, and we will likely send you updated versions during this phase. You will be notified through the development forum if a new software version is available. Therefore we recommend checking the forum regularly. Also on the forum and Dashboard we inform you on any scheduled or unscheduled service outages. The description in this document only gives generic instructions on installing the software toolchain (Arduino IDE). We refer to the respective manuals of the tools used for detailed usage instructions.

Installation of the software is straightforward and should only take 30-60 minutes. Even though the Lacuna library is currently targeted at the Arduino architecture it should be relatively easy to port it to other platforms.

4.4 Software subject to NDA

We like to remind you here that although it is our intention to make the software libraries open source, during the current phase of the partner programme the software falls under the NDA and may not be distributed to other parties. Feel free to contact Lacuna if you have questions regarding our future open sourcing policies.

5 Software installation

5.1 Install software development toolchain

Setup Arduino IDE

If it is not already installed on your computer you should install the Arduino IDE (Integrated Development Environment). The software and installation instructions can be found here:

https://www.arduino.cc/en/Main/Software

The version should be 1.6.8 or higher. The Arduino IDE allows you to edit and compile the source code of your programs. After successful compilation you can also use the Arduino IDE for downloading (programming) your software to the LS200. Furthermore the Arduino IDE facilitates debugging by allowing you to monitor the USB port of the LS200 for any debug messages generated by your source code. The arduino also contains tools for easily adding software libraries to your projects (Library Manager) and development boards (Board Manager), both of which will be use this description.

Setup LS200 hardware.in Arduino IDE

In order for the Arduino environment to be able to compile the code for the MCU used on the LS200 (STM32L476), the toolchain for this MCU needs to be installed first. This can be done via the board manager in the Arduino IDE:

1. *Start the Arduino IDE*
2. *Select "Preferences" from the "File" menu*
3. *Add the following line to "Additional Board Manager URL":*

   https://lacunaspace.github.io/arduino-STM32L4-Lacuna/package_STM32L4_Lacuna_boards_index.json

4. *Open the "Boards Manager" via the "Tools" menu by clicking:*

Tools → Board: '.....' → Boards Manager

5. *Select and install the latest version of "STM32L4 Boards (Lacuna)"*
6. Select the "Lacuna-LS200" board via the "Tools"menu by clicking:

Tools → Board → STM32L4 (Lacuna) → Lacuna-LS200

Install USB drivers for the LS200

As a final step for the installation we need to install the USB drivers for the LS200 board. The setup procedure is dependent on the operating system used. Installation instructions can be found here:

https://github.com/LacunaSpace/arduino-STM32L4/tree/lacuna

Follow the instructions under "OS Specific Setup" for your operating system (Linux/Windows).

5.2 Install Lacuna software libraries

After installing the Arduino IDE the Lacuna specific libraries can be installed in the appropriate directories. First obtain the latest version of the Lacuna Library from the Lacuna User Form via the following link:

https://forum.lacuna.space/t/liblacuna-releases/38

After downloading the file extract it into a temporary directory. After extracting the temporary directory will contain the following files:

- LibLacuna.zip → contains the source code and examples for LibLacuna
- SGP4-Library.zip → contains the source code and examples for SGP4 library
- LS200-schematic.pdf → contains the schematic of the LS200 reference design board
- Lacuna-TTN-Payload-Decoder.js →contains the javascript code for TTN for decoding the payload of the automated-satellite-link example.

The zip files for LibLacuna and SGP4 need to be extracted to the appropriate library folders created during the installation of the Arduino IDE. The exact path might differ depending on you operating system. For Windows systems the folder structure after exctraction should look something like:

```
My Documents\Arduino\libraries\LibLacuna\Src\..
My Documents\Arduino\libraries\LibLacuna\Examples\..
My Documents\Arduino\libraries\SGP4-Library-1\Src\..
My Documents\Arduino\libraries\SGP4-Library-1\Examples\..
```

The correct installation of the libraries can be checked by restarting the Arduino IDE and selecting the "File" menu under "Examples" there should now be a category for LibLacuna and SGP4.

5.3 Install libraries for LS200 sensors

In order to be able to configure and use the sensors available on the LS200 reference design board the correct software libraries need to be installed. The installation can be done easily via the Arduino IDE. Select the option “Manage Libraries” from the “Tools” menu and install the latest version of the following libraries:

- Sodaq_LSM303AGR
- Sparkfun BME280
- Sparkfun Ublox Arduino Library

Make sure you select these exact libraries as there are different variants of libraries for the same hardware. Failing to install or installing the wrong libraries will generate errors when compiling the examples.

6 Programming the LS200

- Make a reference copy of the automated satellite uplink code and copy device id, app eui and dev eui to the code. Make sure you keep this copy of the code so you always have a working starting point to fall back to!
- Upload the code to the LS200
- Check if:
  - serial console does not give error messages
  - the device connects to your local TTN gateway (you should see packets on the TTN console)
  - serial console if TLE data was successfully updated
  - serial console for time and date of the next available satellite pass/uplink opportunity
  - check if this corresponds with the data on the Lacuna dashboard
- Check your setup with a terrestrial gateway!
- Position device outside with direct line of sight to the sky (no major obstructions nearby)
- The device will wait for the satellite to pass and if everything is configured correctly will transmit to the satellite
- Check the TTN console for incoming data under your registered device and application. **Be mindful of the fact that the TTN console only logs data if you are online** (data delivery is done on the browser side, so if the console is not open or PC not on you will see no data). As we have explained above, please use the Data storage integration or any other Integration of choice from AWS to more specific tools: https://www.thethingsnetwork.org/docs/applications/integrations.html

- Keep the current configuration steps and freeze them (keep this instance in the Things Console, keep the code you used for the test saved as-is). You can fall back to this in case later on during development things don't work anymore.

7 Disclaimer

Participating in the Lacuna Technology Partner program is at the partner's own risk. Lacuna will introduce Service Level Agreements at the start of the commercial service but at this early stage, the services are provided on a 'best efforts' basis. Lacuna does not give any guarantee on (future) availability of the services offered. During this phase, Lacuna reserves the right to change the specifications for access to the service without prior notice (although in practice we will always strive to inform you through the established support channels). For example, this means that the software provided by Lacuna is subject to change as we continue to improve it through working with our Partners. The reference design provided might also need changes to improve future access to the services. Lacuna cannot indemnify participants for any cost they might incur during this programme, for example in developing prototypes.