

GlacTracker Deployment Manual

Michael Jones

12/01/23

This manual is designed to help install and configure the software onto the various components of the glactracker device. This can be performed on the devices separately or after assembly of the MCU PCB but most steps are recommended to be performed before travelling to the site (aside from any steps that are required to be on site which will be marked as such). For hardware assembly advice, please refer to the hardware manual inside the design archive.

Contents

1	uBlox Configuration	2
1.1	Prerequisites	2
1.2	Basic configuration	2
1.3	Configuring the fixed RTCM3 coordinates - Base only - On-site location required	2
1.4	Saving the configuration - IMPORTANT	3
2	Thing+ Setup	3
2.1	Firmware Installation	3
2.2	Configuring the first alarm wakeup	4

1 uBlox Configuration

1.1 Prerequisites

These steps must be done to start configuring any parts of the GPS module

1. Install the U-Center software (for M9 boards and below?) found here to a computer or laptop
2. Plug in the uBlox module via its included USB-C port into the computer
3. Open the U-Center software

1.2 Basic configuration

This provides most of the configuration for the module (in fact, for the rover it provides the entire configuration)

1. Click the "tools" menu button at the top of U-Center, and then click on "Receiver Configuration"
2. Select the corresponding config file from the uBlox folder in the firmware
3. Click the Transfer File -> GNSS button and wait for the configuration to be transferred
4. The basic configuration is now in volatile memory on the module. To ensure the module remembers this configuration make sure to follow the steps to save the configuration (after you change any other settings you may want). **if you do not save the configuration, the device will lose all modified settings upon power loss**

1.3 Configuring the fixed RTCM3 coordinates - Base only - On-site location required

This section only applies to base station devices, and unfortunately must be done on site at actual deployment time because an accurate deployment location must be obtained from the device (unless you are omniscient). It describes how to configure the fixed coordinates of the base station's GPS module that is used to obtain the RTCM3 correction data.

1. Open the "Configuration view" dialog window
2. Under the "TMODE3" sub-menu, click on the "Survey In" radio button. This will start finding an accurate location (which may take a long time)
3. Once the accuracy of the given location is satisfactory, copy the survey in information somewhere where you can view it (if text copy doesn't work take a screenshot if necessary) making sure you at least have latitude, longitude and accuracy

4. Click the "Fixed mode" radio button to return the device to fixed RTCM3 mode
5. Enter the previously copied information into the fixed mode text boxes, making sure to put the correct information (latitude, longitude and accuracy) into the correct boxes (in particular, latitude and longitude may be swapped from how they appear in survey in mode)
6. **Save the configuration. If you do not do this it will lose its configuration on power loss**

1.4 Saving the configuration - IMPORTANT

Once you have configured the module, you must make sure to follow these steps to actually save the configuration, **as there is no auto-save functionality**.

1. Click the "Receiver" menu button at the top of U-Center, and then hover over the "Action" sub-menu
2. Click the "Save Config" button
3. The configuration is now safely saved and the module can be powered off safely

2 Thing+ Setup

2.1 Firmware Installation

1. Plug in the Sparkfun Thing+ via its included USB-Micro B port into a computer or laptop
2. The Thing+ should show up as a storage drive
3. If the storage drive contains a file with a .uf2 extension (typically CURRENT.UF2) then the CircuitPython core needs to be installed. If not, skip the following sub-steps.
 - (a) Download CircuitPython 7.3 stable (at time of writing) core for the Thing+ SAMD51 (can be found [here](#))
 - (b) Copy the downloaded file and paste it directly into the Thing+ storage drive that appeared when you plugged the Thing+ in (by default this is named THING-PLUS)
 - (c) The device should reboot with CircuitPython core installed (and no uf2 file in the virtual storage drive)
 - (d) Optional: you may now reformat the new storage drive as FAT32 with a different drive label if you wish. The CircuitPython bootloader installation will remain intact

4. Change the configuration file in the firmware src to configure the device. In particular, the `DEVICE_ID` field will determine if the device is a base or a rover, so make sure to set that correctly.
5. Copy the contents of the firmware src folder (including your new config) to the storage drive, replacing any files if necessary.
6. If the RTC has lost power at some point or has the incorrect alarm time, you will need to configure the first alarm time

2.2 Configuring the first alarm wakeup

This will set the time and date that the device will first start up at after being deployed. Requires a laptop/computer capable of running a serial monitor over usb, so if you wish to do this on-site make sure the laptop is prepared with this (Visual Studio Code with the CircuitPython extension is recommended for this, although any serial monitor capable of sending and receiving should work fine)

1. Change the `first_alarm.time` variable in the `Utility/set_first_alarm.py` file to the time that you want the first wakeup of the device to be at.
2. Optional: Disable the watchdog in the device config file (**make sure to re-enable it after the next step!!!**)
3. Press `Ctrl+C` in a serial monitor and paste the contents of the `Utility/set_first_alarm.py` file into the REPL that appears.
4. If the watchdog was disabled, re-enable it via the config on the device's storage
5. The alarm is now ready, assuming the RTC does not lose power.