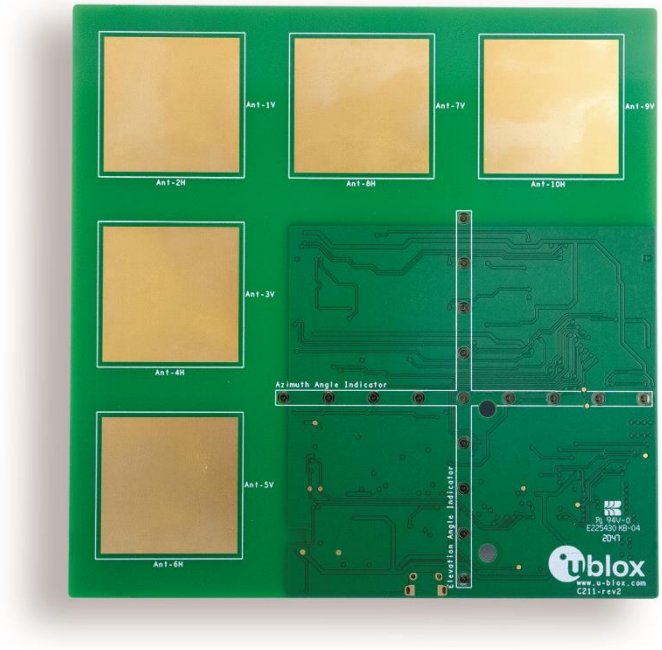
S

XPLR-AOA explorer kits

Bluetooth indoor direction finding

User guide



Abstract

This document describes the XPLR-AOA explorer kits for evaluating Bluetooth 5.1 direction finding and indoor positioning use cases using Angle-of-Arrival methodology. It serves as a practical guide that explains how the u-blox modules and software included in the kit are used to explore direction-finding tags and anchor nodes. The kit contents, setup, configuration, and operation are described.

UBX-21004616 - R09
C1 - Public

Document information

Title	**XPLR-AOA explorer kits**	
Subtitle	Bluetooth indoor direction finding	
Document type	User guide	
Document number	UBX-21004616	
Revision and date	R09	1-Jun-2022
Disclosure restriction	C1 - Public	

This document applies to the following products:

Product name
NINA-B4
XPLR-AOA-1
XPLR-AOA-2

Contents

1 Product description

A Bluetooth "tag" is a small, thin device that can attached to any object to track its whereabouts. An "anchor node" calculates the position of the tag. Bluetooth tags can attach to keys, wallets, purses, and other personal property.

u-blox direction finding solutions that leverage this Bluetooth direction finding technology can be evaluated using two separate explorer kits, as described in Table 1.

Model	Ordering code	Description
XPLR-AOA-1	XPLR-AOA-1	Bluetooth 5.1 direction finding explorer kit for evaluating Bluetooth 5.1 direction finding using Angle-of-Arrival methodology. The kit includes a single C211 application board and one C209 tag: • C211 application boards include a NINA-B411 module and an antenna array that represents the anchor node for direction finding, using Angle-of-Arrival methodology. u-connectLocate direction finding software installed on NINA-B411 delivers angle information for tracked tags to a listening host. • C209 tags are based on the open CPU NINA-B406 module variant. The tags run on custom tag software that advertise Eddystone beacons with appended with Constant Tone Extensions (CTE). The CTE data is used by C211 application board to calculate the position of the C209 tag.
XPLR-AOA-2	XPLR-AOA-2	Bluetooth 5.1 direction finding explorer kit for evaluating and developing indoor positioning use cases, using Angle-of-Arrival methodology. The kit includes a four C211 application boards, four C209 tags and positioning engine client software: • C211 application boards include a NINA-B411 module and an antenna array that represents the anchor node for direction finding using Angle-of-Arrival methodology. u-connectLocate direction finding software installed on NINA-B411 delivers angle information for tracked tags to a listening host. • C209 tags are based on the open CPU NINA-B406 module variant. The tags run on custom tag software that advertise Eddystone beacons with appended with Constant Tone Extensions (CTE). The CTE data is used by C211 application board to calculate the position of the C209 tag.

Table 1: u-blox direction-finding explorer kits and ordering codes

For further information about the positioning engine client software for Windows, see also the indoor positioning guide [2].

Figure 1: u-blox direction-finding explorer kits comprising C211 application board(s) and C209 tag(s)

1.1 Kit includes

XPLR-AOA-1 direction finding explorer kit for out-of-the-box AoA evaluation includes:

- C211 antenna board with NINA-B411 Bluetooth LE module
- C209 tag with NINA-B406 Bluetooth LE module
- u-connectLocate direction finding software (from u-blox.com)
- C209 tag software example (from GitHub)

XPLR-AOA-2 indoor positioning explorer kit for out-of-the-box evaluation of indoor positioning includes:

- Four C211 antenna boards with NINA-B411 module
- Four C209 tags with NINA-B406 module
- u-connectLocate direction finding software (from u-blox.com)
- C209 tag software example (from GitHub)
- Positioning engine software example to run on a PC

☞ The C211 antenna board included in the initial versions of the XPLR-AOA-1 and XPLR-AOA-2 hardware kits featured a LED array that indicated the direction towards the C209 tag. Due to constraints in component availability, later versions of these kits are built without this LED array. This component is not essential for using the kits as they were originally intended.

1.2 Evaluation software

Several evaluation packages are available:

- u-connectLocate delivers angle information for tracked tags to a listening host.
  Customers install u-connectLocate software on the NINA B411 module mounted on the C211 integration board. C211 integration boards are delivered with bootloader software only.
- s-center software Bluetooth and Wi-Fi evaluation software provides a powerful and easy-to-use tool for evaluating, configuring, and testing u-blox short range modules
- Sample positioning-engine client that runs on a local Windows workstation (XPLR-AOA-2 only)

1.3 System requirements

- PC with USB interface
- Operating system: Windows 7 onwards, Linux, or Mac OS

2 Quick start guide

Use this to set up the C211 anchor and C209 tags included in the XPLR-AOA kits.

2.1 C211 anchor setup

Use the following procedure to set up the C211 anchor boards.

1. Connect the anchor to your computer using a USB-A to Micro-B cable. Drivers for the onboard FTDI USB-to-UART converter are normally installed automatically, but otherwise download and install these directly from the FTDI website [20]. The serial port is visible in the device manager when it is successfully installed.
2. Flash the u-connectLocate software using the bundled (Windows only) `newtmgr` executable or s-center [7]. Alternatively, use the Newt Manager Guide [10] to install the `newtmgr` application tool for MAC OS, Linux, and Windows. See also Flashing.
3. Start a terminal emulator (Tera Term or similar) and open the port with the settings described in Configuring the board.
4. Restart the board using the Reset button. A `+STARTUP` event is shown in your terminal emulator when the board is ready to accept AT commands and deliver direction finding angle calculation events. See also Angle calculation event +UUDF.

2.2 C209 tag set up

Use the following procedure to set up the C209 tags.

1. Connect the anchor to your computer using a USB-A to Micro-B cable. Drivers for the onboard FTDI USB-to-UART converter are normally installed automatically, but otherwise download and install these directly from the FTDI website [20]. The serial port is visible in the device manager when it is successfully installed.
2. Flash the C209 tag software available from the C209 GitHub repo [13] using command line tools or s-center [7], as described in Software and flashing. After the tag is flashed, the LED on the C209 tag blinks blue rapidly to show that the tag is now sending direction-finding enabled advertisements. You should now see angle calculation events over the UART of the C211.
3. You can now switch to power the C209 tag using a CR2032 battery (not included).

2.3 Wireless anchor-to-host setup

To set up wireless communication from the C211 anchor to a host, see EVK-ODIN-W2 wireless gateway configuration. The setup requires one EVK-ODIN-W2 per C211 anchor node.

3 Bluetooth direction finding

Bluetooth direction finding provides a relatively inexpensive and flexible approach to developing location-related applications for both in indoor and outdoor environments.

Examples of applications for which Bluetooth direction finding technology is most suitable include:

- Asset tracking
- Navigation
- Wayfinding
- Proximity/Direction detection

3.1 Technology

Bluetooth direction finding can be implemented using two different methods, Angle of Arrival (AoA) and of Departure (AoD).

In each case, protocol-specific control information and user data, transmitted as Bluetooth Protocol Data Units (PDU), are appended with direction-finding data known as Constant Tone Extension (CTE). This additional direction-finding data is appended to the end of the packages, as shown in Figure 2.

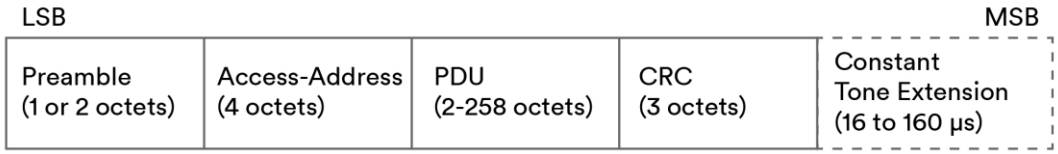


Figure 2: Bluetooth PDU with Constant Tone Extension

3.1.1 Angle of Arrival (AoA)

In AoA systems, the receiver has an antenna array with multiple antennas. The receiver calculates the phase shift between these antennas to detect the direction of the tag that it is tracking. An overview of a system using this method of direction finding is shown in Figure 3.

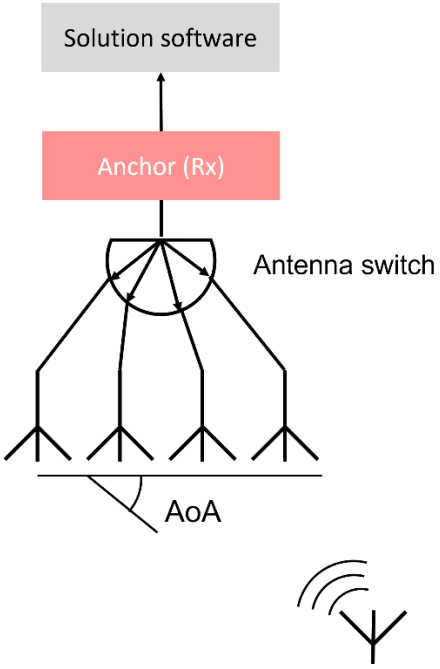


Figure 3: Angle of Arrival (AoA) system architecture

3.1.2 Angle of Departure (AoD)

In AoD systems, the transmitter has multiple antennas. The receiver calculates the phase difference between these antennas to determine the direction to the transmitter. By using this data in combination with angle data from other transmitters, the receiver can estimate its position. An overview of a system using this method of direction finding is shown in in Figure 4.

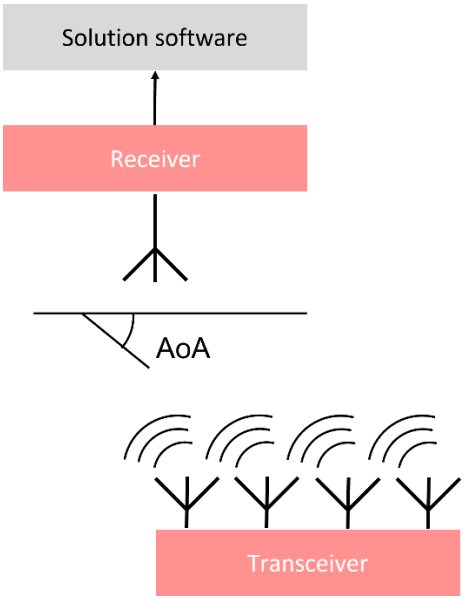


Figure 4: Angle of Departure (AoD) system architecture

3.1.3 Angles of measurement

In both AoA and AoD systems, the reported angles of the azimuth and elevation measurements are compared against a reference plane, as shown in Figure 5.

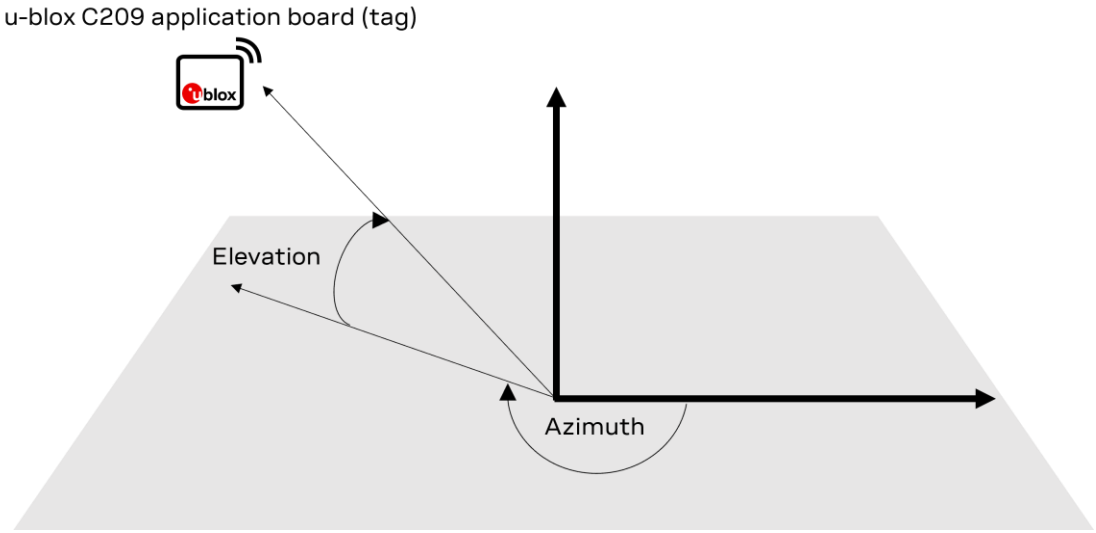


Figure 5: Azimuth and elevation angles

For further information about Bluetooth direction finding, see the u-blox webinar “Bluetooth for High Precision Indoor Positioning” available on the u-blox webinar page [8] and Bluetooth SIG technical overview [1].

4 XPLR-AOA anchor nodes and tags

4.1 Introduction

Both XPLR-AOA-1 and XPLR-AOA-2 explorer kits include both Bluetooth anchor nodes and tags.

Although this chapter generally describes these nodes and tags in the context of the XPLR-AOA-1 direction finding system, the concepts it describes are equally applicable to the XPLR-AOA-2 indoor positioning explorer kit. For information about the XPLR-AOA-2 positioning engine, setup, and configuration, see also the indoor positioning application note [2].

4.2 Overview

u-blox direction finding solutions are comprised of C211 anchor nodes C209 tags, as shown in Figure 6. See also Kit includes. u-blox direction-finding solution supports the Angle of Arrival (AoA) methodology.

C211 anchor nodes are based on NINA-B4 modules that include support for direction finding. Anchor nodes are based on NINA-B411 u-connectXpress functionality, whereas C209 tags are based on NINA-B406 open CPU architecture. For more information about these short-range Bluetooth modules, see also the respective data sheets [14][15] and product pages [2].

Figure 6 shows several anchor nodes and a host that uses the combined information from the anchor nodes to calculate the position of the tag. For simple direction finding one anchor node is sufficient.

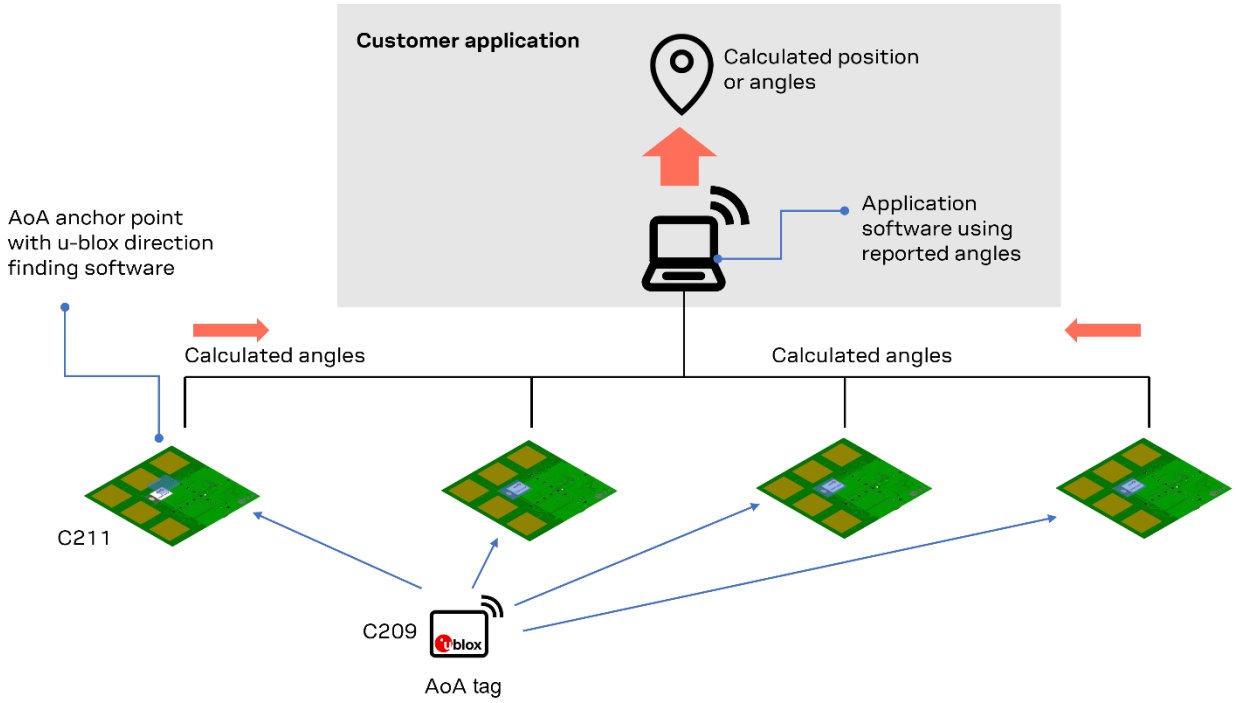


Figure 6: XPLR-AOA direction-finding solution showing four anchor points tracking a single tag

4.3 C211 anchor nodes

4.3.1 Overview

C211 application boards are equipped with a NINA-B411 module and an antenna array. These boards fulfill the role of the anchor node in the XPLR-AOA direction-finding solutions. Anchor nodes run AT command-based u-connectLocate direction finding software, which delivers AoA data for tracked tags to a listening host. Data is transmitted to the host as events over the NINA-B411 UART interface. See also Anchor node AT commands.

- C211 has an array of five antennas that are used to detect the phase shift for the direction finding – in both horizontal and vertical levels.
- C211 also has an LED array in the form of a cross. The LEDs indicate the direction of tracked beacons. In instances where the anchor node tracks several beacons, the LED array tracks the first C209 tag that is discovered.

☞ Some product variants do not include the LED cross on the C211 application board, which is part of the XPLR-AOA kits:

- XPLR-AOA-1-00 C211 has an active LED array
- XPLR-AOA-1-01 C211 does not have an active LED array
- XPLR-AOA-2-00 C211 has an active LED array
- XPLR-AOA-2-01 C211 does not have an active LED array

Figure 7 shows an overview image of the C211 antenna side.

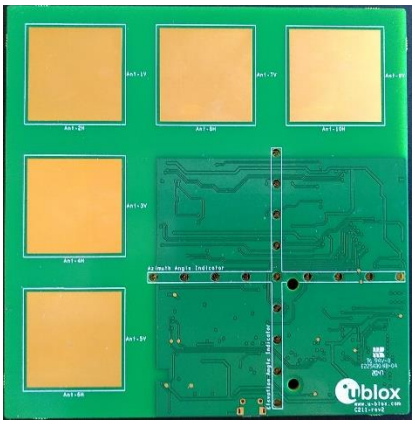


Figure 7: C211 antenna side

The C211 board dimensions are 115 (h) x 114 (w) mm.

☞ C211 schematics are proprietary and not shared.

4.3.2 Connectors

4.3.2.1 UART

The UART connection must be configured for USB connection with jumpers connecting pins 3 to 5 and 4 to 6 on connector J5, as shown in Figure 8.

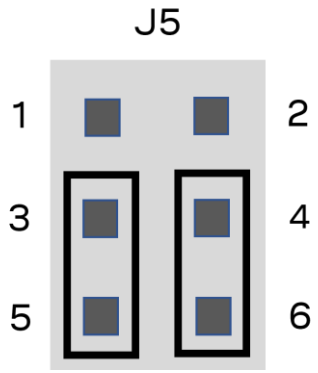


Figure 8: Jumper settings for UART connection over USB

4.3.2.2 Buttons

The C211 board has three buttons on one edge of the board:

- RESET: Resets (reboot) the board
- SW1: Initiates factory reset of the board settings: Hold down SW1 during board reset and hold for 10 seconds to trigger a factory reset.
- SW2: Enters the software download mode: Hold down SW2 during board reset to enter the download mode. See also Flashing.

4.3.2.3 Arduino interface

C211 boards contain an Arduino compatible interface that can be used to connect, for example, an ODIN-W2 EVK to enable wireless communication over UDP. For further information about UDP, see the u-connectXpress software user guide [17].

The Arduino connectors, J1, J3 and J4, are shown in Figure 9.

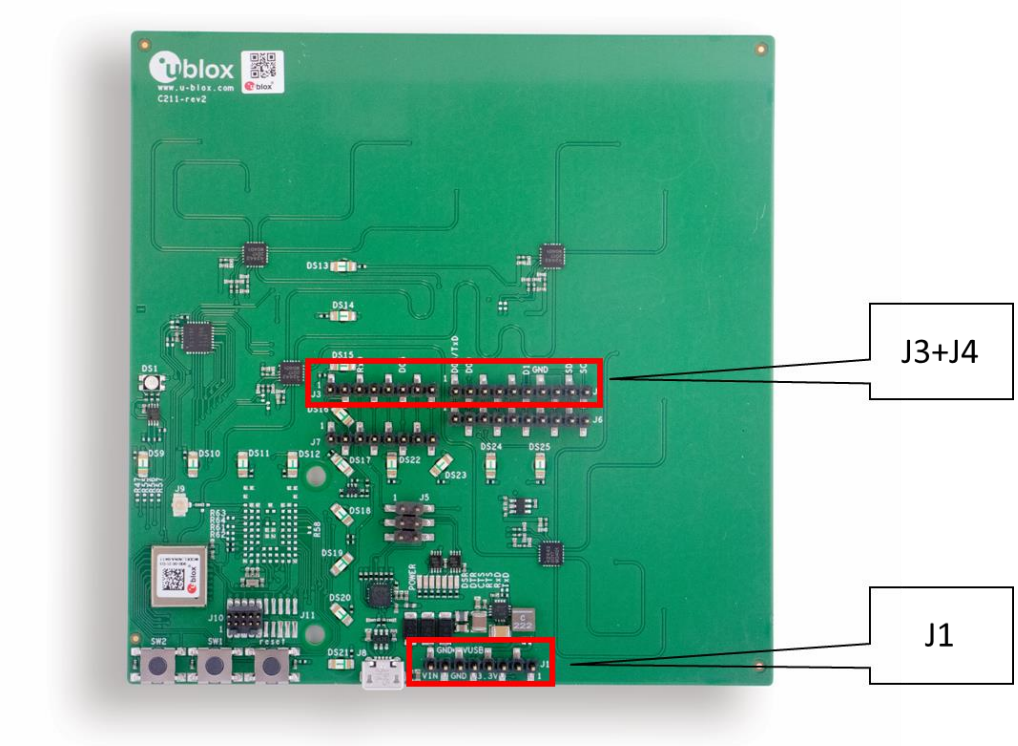


Figure 9: C211 with Arduino connectors marked

4.3.2.3.1 Pinout

The pinout of the Arduino compatible connectors is described in EVK-ODIN-W2 wireless gateway configuration.

Connector	Pin	Name	Description
J1	1	NC	Not connected
	2	IOREF	IO reference voltage
	3	RESET	Reset
	4	3V3	Regulated 3.3 V net. This net is supplied by the board and is always powered as long as a power source is connected.
	5	5V0	5 V supply
	6	GND	Ground
	7	GND	Ground
	8	VIN	External power supply via ODIN-W2
J3	1	NC	Not connected
	2	NC	Not connected
	3	RXD	Can be connected to NINA-B411 GPIO_22/UART_TXD by populating jumper J5 pin [1-3]
	4	NC	Not connected
	5	NC	Not connected
	6	D5	Not Connected. Can be connected to NINA-B411 GPIO_32 by populating position R68.
	7	NC	Not connected
	8	NC	Not connected
J4	1	TXD/D8	Can be connected to NINA-B411 GPIO_23/UART_RXD by populating jumper J5 pin [2-4] Not Connected. Can be connected to NINA-B411 GPIO_33 by populating position R70.
	2	D9	Not Connected. Can be connected to NINA-B411 GPIO_46 by populating position R69.

Connector	Pin	Name	Description
	3	NC	Not connected
	4	NC	Not connected
	5	NC	Not connected
	6	D13	Not Connected. Can be connected to NINA-B411 GPIO_45 by populating position R67
	7	GND	Ground
	8	NC	Not connected
	9	SDA	Not Connected. Can be connected to NINA-B411 GPIO_4/I2C SDA by populating position R66.
	10	SCL	Not Connected. Can be connected to NINA-B411 GPIO_5/I2C SDL by populating position R65.

Table 2: Pin out of Arduino connectors

4.3.2.3.2 Connecting EVK-ODIN-W2 for wireless communication

When connecting an EVK-ODIN-W2 evaluation kit for wireless communication, you need to redirect the TX channel from the C211 anchor to EKV-ODIN-W2 by modifying the jumpers on the J5 connector, as shown in Figure 10. This connects the UART TX pin on the C211 to UART1 RX on the EVK-ODIN-W2. See also the EVK-ODIN-W2 user guide [16].

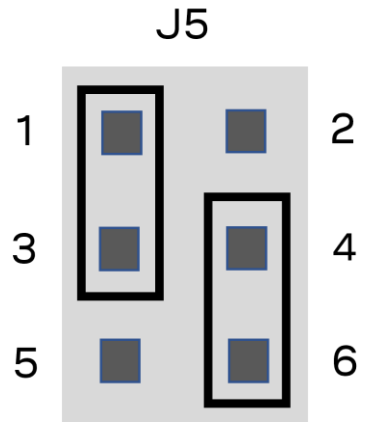


Figure 10: Redirecting C211 UART TX to Arduino connector

4.3.2.3.3 Power supply

When mounting an ODIN-W2 EVK, the application board is powered through the ODIN-W2 EVK USB contact. No other power supply to the C211 board is needed.

4.3.2.4 EVK-ODIN-W2 wireless gateway configuration

For C211 wireless communication though an EVK-ODIN-W2, the EVK must first be configured as a wireless UDP gateway. In this configuration the EVK can:

- Connect to a Wi-Fi network
- Automatically connect to the UDP server at "`server_ip`" and "`udp_port`"
- Startup in data mode
- Set the UART to 1 Mbps with no flow control

1. Enter the following commands to setup EVK-ODIN-W2 as a wireless UDP gateway:

☞ Replace the "ssid", "password", "server_ip" and "udp_port" with suitable values for your network.

```
AT+UWSC=0,0,1
AT+UWSC=0,2,"ssid"
AT+UWSC=0,5,2
AT+UWSC=0,8,"password"
AT+UWSC=0,100,2
AT+UWSC=0,107,0
AT+UWSC=0,300,0
AT+UWSC=0,301,1
AT+UWSCA=0,1
AT+UWSCA=0,3
AT+UDDRP=0,"udp://server_ip:udp_port/",2
AT+UMSM=1
AT+UMRS=1000000,2,8,1,1,0
AT&W
AT+CPWROFF
```

☞ The above settings start up the module in data mode and change the UART settings to 1 Mbps with no flow control. If you need to modify settings, open the COM port in s-center with 1 Mbps speed and no flow control, and then press the "AT Mode" button to get back into command mode. For more information, see the u-connectXpress AT commands manual [5].

☞ Note that some commands shown above are not valid in all ODIN-W2 software versions. If any of these commands return an error, use s-center to upgrade your ODIN-W2 EVK to software version 8.0 or later. See also the s-center user guide [18].

For other configuration possibilities see the u-connectXpress AT commands manual [5] and u-connectXpress software user guide [17].

2. After the configuration it is necessary to remove jumpers J13 and J22, as shown in Figure 13. These jumpers select UART1/UART3 on the EVK-ODIN-W2 and direct the UART signals to the Arduino-compatible interface.

Figure 11: UART selection jumpers to be removed on ODIN-W2 EVK

3. On C211, set the UART baud rate and other interface settings to 1 Mbps, with no flow control:

```
AT+UMRS=1000000,0
AT&W
AT+CPWROFF
```

4.3.3 Flashing

Other than the bootloader, C211 boards are delivered without software.

4.3.3.1 Command line flashing

Follow the procedure outlined below to flash the board over the UART connection.

1. Download the u-connectLocate software container from [19].
2. Flash the u-connectLocate software using the bundled (Windows only) `newtmgr` executable. Alternatively, use the Newt Manager Guide [10] to install the newtmgr application tool for MAC OS, Linux, and Windows.
3. Use Newt Manager to install u-connectLocate on the NINA-B411 module:

```
newtmgr --conntype=serial --connstring="COMXX,baud=115200" image upload <binary
image>
```

4. Press the reset button to reset the application board or reset it with `newtmgr`:

```
newtmgr --conntype=serial --connstring="COMXX,baud=115200" reset
```

☞ As there is no software to boot during the initial startup, the NINA-B411 module automatically enters the software download mode. In future updates of NINA-B41 u-connectLocate software the bootloader must be manually set in software download mode by pressing the SW2 button during board reset. The board is reset by inserting the USB cable or by clicking the RESET button. See also Figure 15.

4.3.3.2 Flashing using s-center

The C211 anchors can also be flashed over the UART using s-center [7].

☞ This requires s-center 6.1 or later.

Follow the flashing procedure outlined below when using s-center:

1. Download and unzip the u-connectLocate software container as described in Command line flashing.
2. Open s-center on the correct COM port once the module is in software download mode.
3. Select **Software Update.**
4. In the “Software Update” dialogue select the binary file in the software container, as shown in Figure 12. Make sure the flashing speed is set to 115200.
5. Press **Update**.

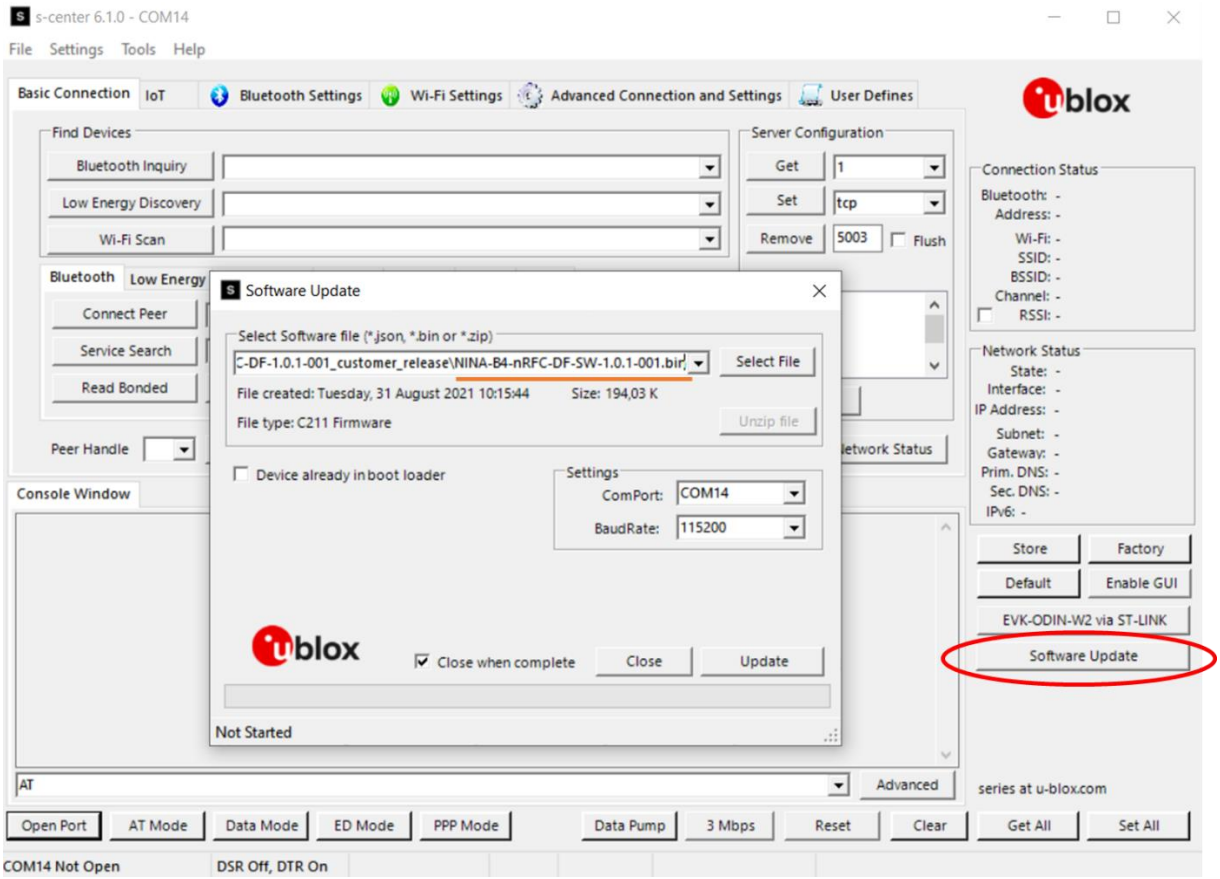


Figure 12: Updating the software on the C211 board using s-center

4.3.4 Configuring the board

When connecting to the USB port on the C211, a serial port (COM port on Windows) is available on the host. Connect the COM port to a terminal emulator or use the s-center tool [7] to initially configure the port settings:

- 115200 kbps
- 8 data bits, no parity, 1 stop bit (8N1)
- Flow control enabled using RTS/CTS

Having configured the COM port, configure the C211 using AT commands. The appropriate AT commands for configuring direction finding through the C211 anchor nodes are described in Appendix A.

☞ Although s-center does not support specific direction-finding AT commands using buttons, it is possible to configure the COM port as a terminal for use with AT commands.

Due to the large amount of data received over the UART it may be advisable to increase the baud rate over the UART interface. In the following command example, the baud rate is set to 1 Mbps using the command `AT+UMRS`.

```
AT+UMRS=1000000,1
AT&W
AT+CPWROFF
```

4.4 C209 tags

4.4.1 Overview

C209 tags are based on the open CPU NINA-B406 module variant. The tags run on custom tag software that advertises as an Eddystone beacon with a Constant Tone Extension added to the advertising packets. This CTE is used by the u-connectLocate direction finding software that runs on the C211 application board to calculate the Angle of Arrival.

The namespace included in the transmitted Eddystone-UID beacon is `0x4E494E412D4234544147` and the instance id is based on the MAC address of the NINA-B406 found on the module QR code label. See also Restore Eddystone Instance ID.

C209 tags are powered by a single CR2032 battery (not included) or through the USB connector.

C209 tags includes a versatile sensor node that comprises several sensors for use in a multitude of different applications. The on-board sensors include:

- LIS2DW12 accelerometer
- APDS-9306 ambient light sensor
- BME280 humidity, pressure, and temperature sensor

The main components of any C209 tag are shown in Figure 13.

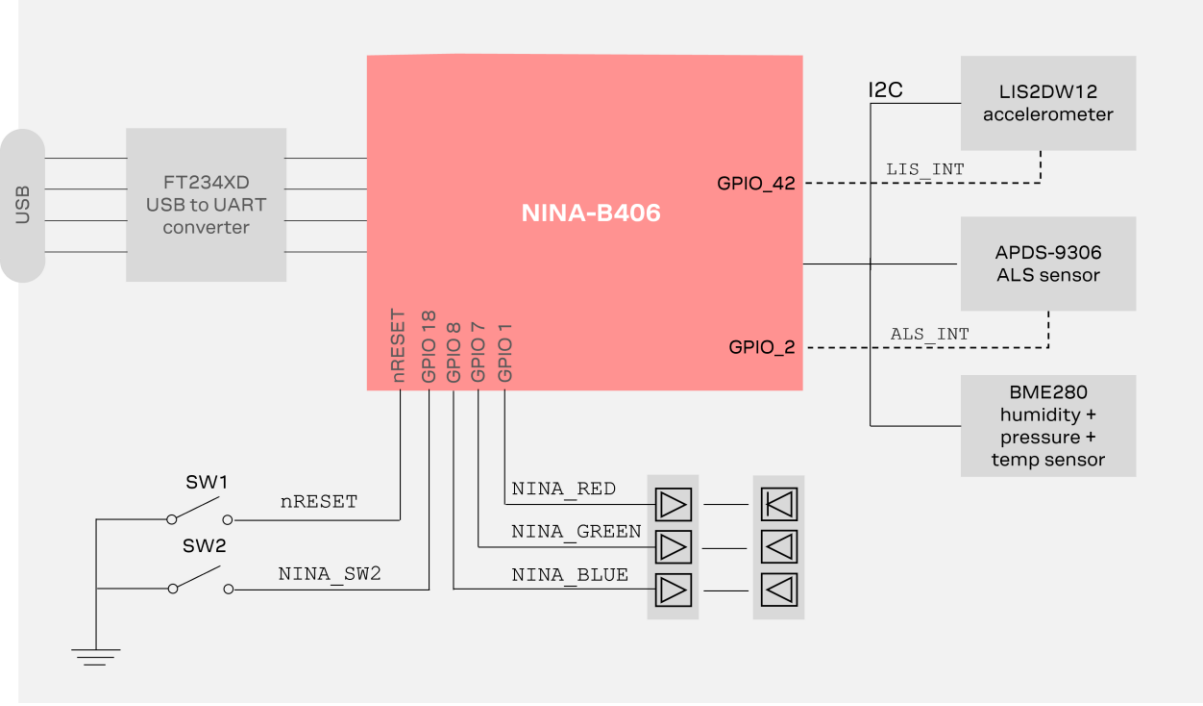


Figure 13: C209 main functional components

The important pin definitions on the C209 application board are described in Table 3.

No.	Name	I/O	Description	Remarks
GPIO_1	RED	O	RED system status signal	Active low
GPIO_2	LIS_INT	I	Interrupt signal from Ambient Light Sensor	
GPIO_7	GREEN	O	GREEN system status signal	Active low
GPIO_8	BLUE	O	BLUE system status signal	Active low
GPIO_18	SWITCH_2	I	Switch_2 button	Active low

No.	Name	I/O	Description	Remarks
GPIO_20	UART_RTS	O	UART request to send control signal	Used only when hardware flow control is enabled
GPIO_21	UART_CTS	I	UART clear to send control signal	Used only when hardware flow control is enabled
GPIO_22	UART_TXD	O	UART data output	
GPIO_23	UART_RXD	I	UART data input	
GPIO_42	LIS_INT	I	Interrupt signal from accelerometer	

Table 3: Important pin definitions on the C209

For more information about programming the module, see also the NINA-B4 system integration manual [6] and NINA-B40 data sheet [14]. See also C209 schematics.

☞ Although the sensors on the C209 application board are not used in the latest direction-finding tag software from u-blox, the sensors can be utilized in any customer application.

4.4.2 Software and flashing

C209 tags are delivered with bootloader software only and do not include the tag software (available from Github [13]) needed for the device to advertise Bluetooth beacons to C211 anchor nodes.

The bootloader on C209 tags is different than that supplied on C211anchor nodes, and the nrfutil [11] flashing tool is needed to install software via UART and the boot loader.

u-blox C209 tag software can be installed using the DFU bootloader or an external debugger. Precompiled packages for each installation option are available from the u-blox C209 software repository [13].

4.4.2.1 Installing tag software with the DFU bootloader over the command line

Install C209 tag software with the pre-flashed DFU bootloader on NINA-B406:

1. Download and install the nrfutil flashing tool from the Nordic Github repository [12].
2. Download and unpack the `c209_aoa_tag_for_dfu_boot.zip` file from the u-blox C209 software repository using this direct link, or go to the "Releases" section via the main GitHub repository page [13]. The unpacked file includes the `app.zip` file used for the installation.
3. Press and hold the SW2 button on the C209 while resetting the board (by inserting the USB cable or clicking the RESET button) to set the bootloader in "download" mode. See also Figure 15.
4. Use the following command to install the C209 tag software using nfutil:

```
nrfutil dfu serial -pkg app.zip -p COMXX -b 115200 -fc 1
```

☞ Replace `COMXX` with the appropriate COM port for your system.

4.4.2.2 Installing tag software with the DFU bootloader using s-center

The C209 tags can also be flashed over the UART using s-center [7].

☞ This requires s-center 6.1 or later.

Follow the flashing procedure outlined below when using s-center:

1. Download and unzip the u-connectLocate sofware container, as described in Installing tag software with the DFU bootloader over the command line.
2. Open s-center on the correct COM port once the module is in SW download mode.
3. Select **Software Update**.
4. In the "Software Update" dialogue, select the `app.zip` file from the software container, as shown in Figure 14. Make sure the flashing speed is set to 115200.
5. Press **Update**.

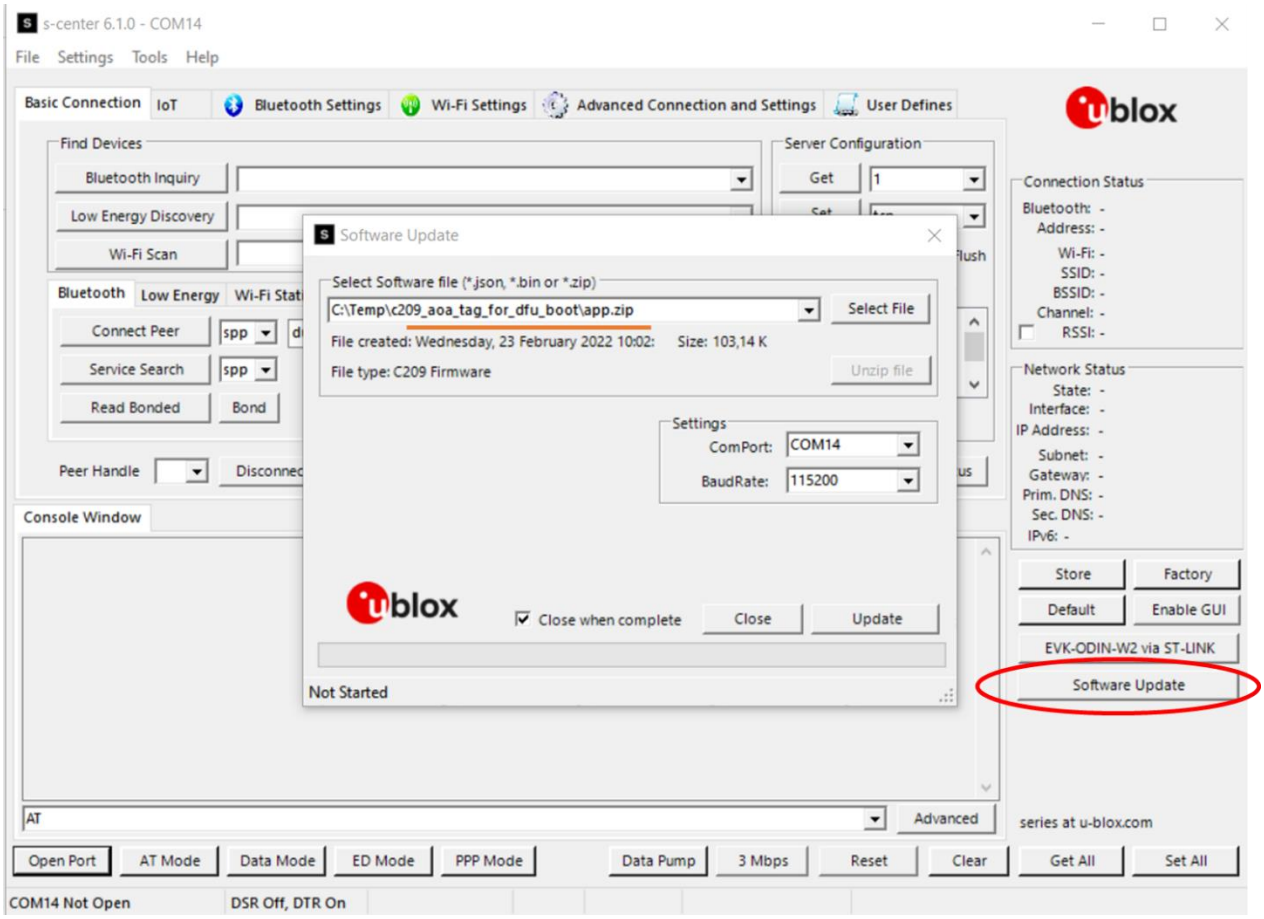


Figure 14: Updating the SW on the C209 board via s-center

4.4.2.3 Installing tag software using a debugger

C209 tag software can also be installed using a debugger connected through the J10 10-pin contact on the PCB. Install C209 tag software with the pre-flashed DFU bootloader on NINA-B406:

1. Download the `c209_aoa_tag_no_boot_required.zip` from the u-blox C209 software repository [13].
2. Unzip the downloaded file.
3. Connect the debugger to the 10-pin contact on the C209.
4. Flash the hex file from the zip using your debugger.

5 System setup

5.1 Anchor node configuration

By default, the C211 Anchor Point comes pre-configured to track all u-blox tags. The u-blox tags advertise with the Eddystone namespace `0x4E494E412D4234544147`, which is default on the C211 to track. So, if you are only using the u-blox supplied XPLR-AOA kit, no configuration is needed.

Each anchor node can be configured with beacons to track. An example for how to set up the anchor to track two tags is shown below:

```
AT+UDFFILT=1,2,”6E616D65737061636578”
At+UDFFILT=2,2,“CCF9578E0D8A”,”CCF9578E0D8B”
AT+UDFENABLE=1 (Tracking is enabled by default, so this is optional)
```

These commands set up the anchor to track the two tags with the given MAC addresses in the Eddystone name space (`6E616D65737061636578`) used by the tags.

The sequence described above reflects the most simplistic use case. Further configuration is possible using the `AT+UDFCFG` command. See also Configuration direction finding +UDFCFG.

The settings shall be saved using the `AT&W` command, followed by a restart (`AT+CPWROFF`).

5.2 Tag configuration

5.2.1 Configuring advertising interval

Press the SW2 button on the C209 to change the advertising interval. The default interval at startup is 20 milliseconds

When SW2 is pressed, the advertising interval cycles through [20, 100, 1000] milliseconds. For best performance, it is advisable to use the default 20 ms advertising interval.

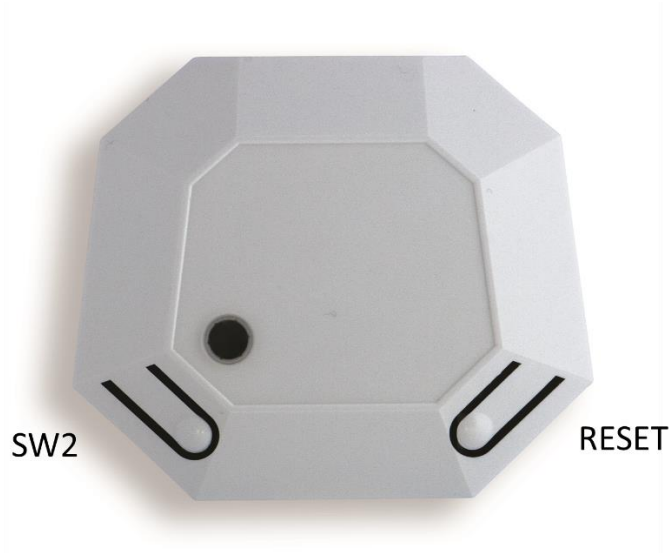


Figure 15: C209 with cover and buttons indicated

C209 tags are simply configured over the UART interface using AT commands.

The AT commands that can be used to configure the interface are described in Table 4.

Command	Description
AT+UMLA=1	Read Local MAC address
AT+GMM	Read the model identifier, NINA-B4-TAG
AT+TXPWR=<valid_tx_power>	<valid_tx_power> can be one of: -40, -30, -20, -16, -12, -8, -4, 0, 2, 3, 4, 5, 6, 7, 8 (dBm)

Table 4: C209 AT commands

☞ For +TXPWR to take effect the software must be reset, by pressing the reset button or power cycling the module. The configuration is persistently stored in flash.

All commands are echoed and responded with either \r\nOK\r\n or \r\nERROR\r\n, and should be terminated with \r.

☞ The UART interface is available for 10 seconds after reset with the settings 115200 bps, with no flow control.

5.2.2 Enabling/disabling advertising

Press and hold the C209 button for ~3 seconds to enable/disable advertising. If the tag is advertising, press and hold the button to stop it advertising. The onboard RGB LED blinks blue when advertising is enabled. The blinking interval correlates to the advertising interval.

5.3 Restore Eddystone Instance ID

All u-blox modules are delivered with a u-blox MAC address, which is written into the UICR register.

If the MAC address is accidentally erased, for example during a reflash of the software, this information is lost. If this happens, the Eddystone Instance ID transmitted in the C209 advertising beacon will not match the MAC address of the module. To correct this:

1. Scan the QR code on the module label. The information in the code includes a code that includes the MAC address (shown here in bold): H85(**CCF9578E0D89**)0400.
2. While having a debugger connected to the SWD interface of the board enter these commands to reinstate the MAC address **CCF9578E0D89** into the UICR of the module:

   ```
   nrfjprog --memwr 0x10001080 --val 0x8E57F9CC
   nrfjprog --memwr 0x10001084 --val 0xFFFF890D
   ```

   Another example for which the MAC address given in the scan code is **0123456789AB**:

   ```
   nrfjprog --memwr 0x10001080 --val 0x67452301
   nrfjprog --memwr 0x10001084 --val 0xFFFFAB89
   ```

3. After writing the MAC address, reset the module. The module now transmits the correct Eddystone Instance ID for the C209 tag.

6 Running system

Once the system is setup, each anchor reports, over the serial port connection, angle calculation events (+UUDF) when it detects a beacon:

```
+UUDF:CCF9578E0D8A,-42,20,0,-43,37,"CCF9578E0D89","",15869
+UUDF:CCF9578E0D8B,-41,10,4,-42,38,"CCF9578E0D89","",15892
+UUDF:CCF9578E0D8A,-42,-10,2,-43,39,"CCF9578E0D89","",15921
…
```

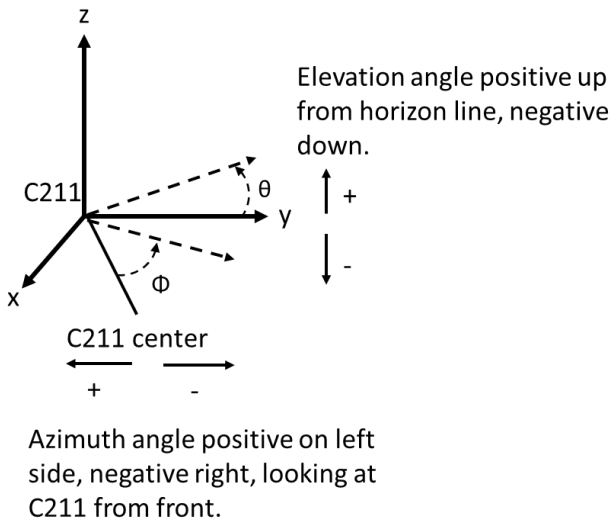
The data reported in this event can be used to estimate a position of the tracked beacon.

The parameters of the +UUDF event are (in order from left to right):

- Eddystone instance ID
- RSSI of 1st polarization
- Azimuth angle
- Elevation angle
- Reserved parameter
- The advertising channel where the advertisement was found. The advertisement channel is one of the normal Bluetooth low energy advertisement channels (37, 38 or 39)
- Anchor ID as set by AT+UDFCFG tag 4. For more details, see Appendix A.3.
- User defined strings as set by AT+UDFCFG tag 2. For more details. See also Configure direction finding +UDFCFG.
- Timestamp

For a detailed description of the +UUDF event parameters, see also Angle calculation event +UUDF.

As in the angle calculation events shown above, the beacon is moving from one side of the anchor to the other as the azimuth angle, given as the third parameter in the command (shown in bold), moves from a positive value (20) to a negative value (-10). The geometric relationships of the azimuth values and the orientation of the angles in relation to the C211 board are represented in the scientific diagram shown in Figure 16.



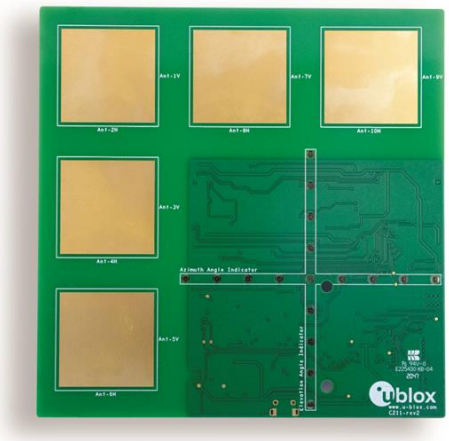


Figure 16: C211 azimuth angles.

The optimal distance between anchor nodes and tags is approximately 2–15 m.

The angle events can also be visualized in the s-center [7] support application using the “IoT->Angle of Arrival” tab, as shown in Figure 17.

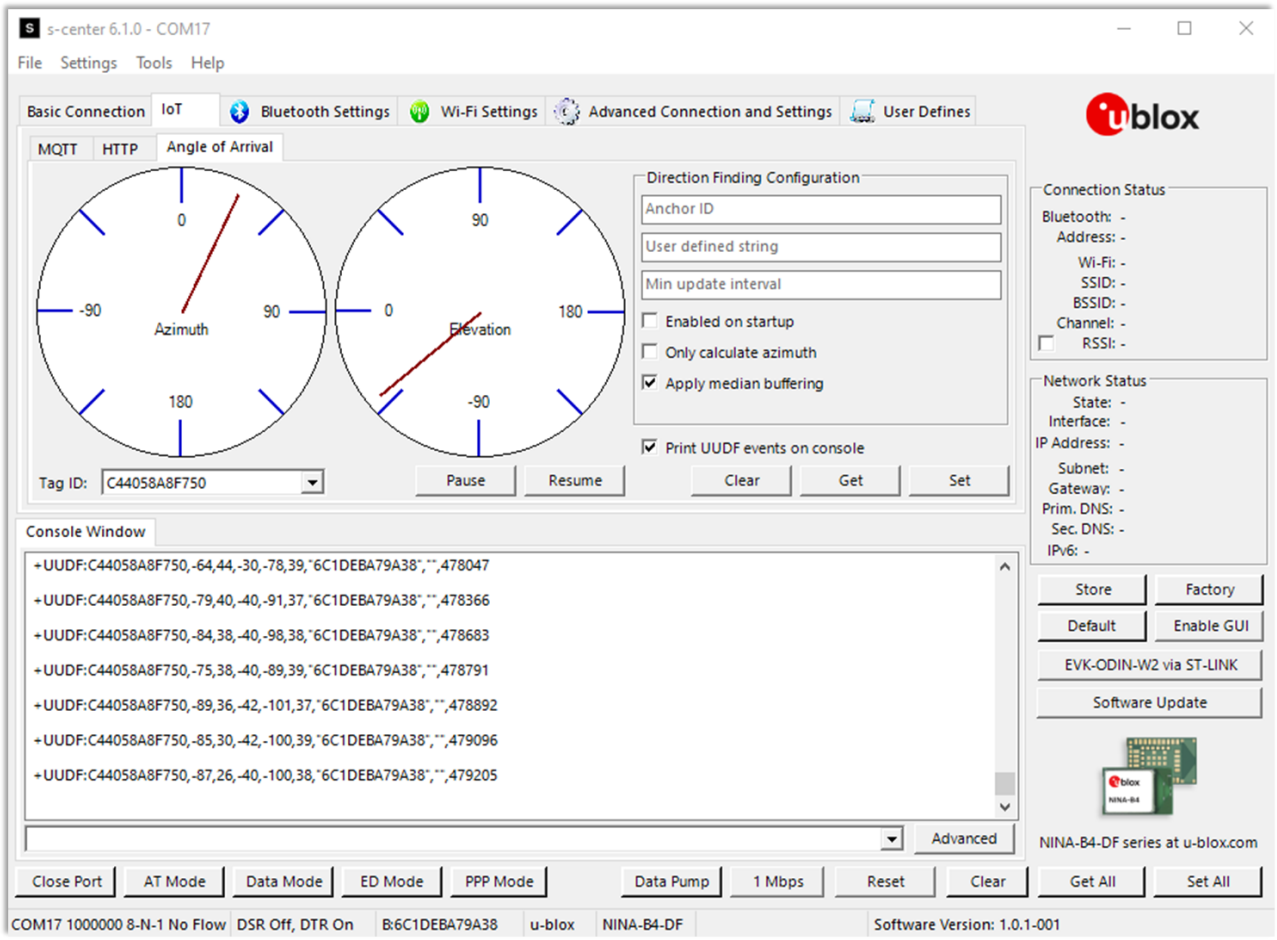


Figure 17: s-center with Angle of Arrival tab

Appendix

A Anchor node AT commands

A.1 Direction finding enable +UDFENABLE

+UDFENABLE				
Modules	NINA-B41X-40B			
Attributes	**Syntax**	**Settings saved**	**Can be aborted**	**Response time**
	Full	No	No	-

A.1.1 Description

AT Command	Description
AT+UDFENABLE=<enabled>	Start or stop angle calculations during runtime

A.1.2 Syntax

Response	Description
OK	Successful write response
ERROR	Error Response

A.1.3 Defined values

Parameter	Type	Description
Enabled	Integer	0: Disabled 1: Enabled

A.2 Direction finding filter +UDFFILT

+UDFFILT				
Modules	NINA-B41X-40B			
Attributes	**Syntax**	**Settings saved**	**Can be aborted**	**Response time**
	Full	Yes	No	-

A.2.1 Description

Configure a filter to decide which tags to track. This command is used to configure the filter to either track all devices with a specific namespace, or individual tags with a certain namespace and instance id. Currently, only filter types 1,2 (EDDYSTONE) are supported.

AT command	Description
AT+UDFFILT=<filter_type>,<action>[,<option_val1> ,<option_val2>,…<option_valXX>]]]	Set the tag filter for tracked tags
AT+UDFFILT=<filter_type>	Read the current filter for the specified <filter_type>

A.2.2 Syntax

Response	Description
+UDFFILT:<filter_type>,<option_val1>,.. OK	Read response
OK	Successful write response
Error	Error response

A.2.3 Defined values

Parameter	Type	Description
filter_type	Enumerator	Filter type, see description in table below
action	Enumerator	1: clear filter 2: append to filter
option_val		Filter values, see description in table below

Filter type	Description	Option values	Option Type	Optional
1	Eddystone namespace	option_val1: Eddystone namespace id. Only one namespace can be set.	String (10 HEX chars)	Yes
2	Eddystone instance id	option_val2: Eddystone instance id. Up to 100 instance ids can be set.	String (6 HEX chars)	Yes

A.2.4 Notes

This setting takes effect immediately. Use the command &W and +CPWROFF to store the configuration to startup database. Maximum filter length is 100.

☞ All hexadecimal data needs to be quoted, e.g., AT+UDFFILT=2,2,"0011223344FF".

A.3 Configure direction finding +UDFCFG

+UDFCFG				
Modules	NINA-B41X-40B			
Attributes	**Syntax**	**Settings Saved**	**Can be aborted**	**Response Time**
	Full	Yes	No	-

A.3.1 Description

This command is used to configure the direction-finding algorithm and the anchor output.

AT Command	Description
AT+UDFCFG=<param_tag>,<param_value>	Write config
AT+UDFCFG?	Read all config options
AT+UDFCFG=<param_tag>	Read individual configuration

A.3.2 Syntax

Response	Description
+UDFCFG:<param_tag>,<param_value> … OK	Read response
OK	Successful write response
ERROR	Error Response

A.3.3 Defined values

Param. tag	Min value/ length	Max value/ length	Default value	Type	Description
1	0	10000	1	Integer	Minimum interval between +UUDF events for each tag in milliseconds. +UUDF events may arrive at a smaller interval if multiple tags are tracked. This setting is used if the host cannot handle the rate of +UUDF events generated or when debugging to get less outputs. In a real scenario, it is better to just throw away the +UUDF events on host if they cannot be processed.
2	0	30	""	String	User defined string that can be set to any value. For example, it can be useful to set a GPS position (longitude/latitude) of the anchor for use in a positioning engine. Leave blank if not needed.
3	0	1	1	Integer	Angle calculations enabled at startup. This setting makes the anchor output +UUDF events at startup – without the need to call +UDFENABLE.
4	0	30	MAC address	String	Anchor ID. This sets the Anchor ID field in the +UUDF event. It can be useful when the host just blindly forwards the +UUDF to a server.
5	0	1	1	Integer	Configure if the anchor is to calculate both azimuth and elevation angles. 0: Only the azimuth angle is calculated and output in +UUDF event. 1: Both azimuth and elevation calculated and output in +UUDF event. This is useful if only azimuth angle is used as the disabling elevation angle speeds up the calculation time to make the anchor more efficient.
6	0	1	0	Integer	Antenna board used: 0: C211 1: CoreHW Core Patch antenna 2: ANT-B10 ☞ Valid from u-connectLocate version 1.2. ☞ For the ANT-B10 board the default is configured to 2.
7	0	1	0	Integer	Use CoreHW output format instead of +UUDF u-blox format.
8	0	1	1	Integer	Apply median buffering of output angle. It is advisable to keep this enabled.

☞ For the direction-finding configuration to take effect, use the commands &W and +CPWROFF to store the configuration to the startup database.

A.4 Angle calculation event +UUDF

+UUDF				
Modules	NINA-B41X-40B			
Attributes	**Syntax**	**Settings saved**	**Can be aborted**	**Response time**
	Full	No	No	-

A.4.1 Description

Unsolicited response code for an angle calculation event.

A.4.2 Syntax

Response	Description
`+UUDF:<ed_instance_id>,<rssi_pol1>,<angle_azimuth>,<angle_elevation>,<rssi_pol2>,<channel>,<anchor_id>,<user_defined_str> ,<timestamp_ms>`	Angle calculation event

A.4.3 Defined values

Parameter	Type	Description
`ed_instance_id`	Byte_Array	6-byte Eddystone instance id
`rssi_pol1`	Integer	RSSI of polarization 1
`angle_azimuth`	Integer	Azimuth angle in range -90 to 90 °
`angle_elevation`	Integer	Elevation angle in range -90 to 90 °
`<not used>`	Integer	Reserved for future use
`channel`	Integer	Channel from which the packet angle was calculated
`anchor_id`	String	The value set by `+UDFCFG` param_tag 4
`user_defined_str`	String	The value set by `+UDFCFG` param_tag 2
`timestamp_ms`	Integer	Time since boot in milliseconds

☞ Note that the RSSI values reported in the angle calculation events only give a very rough indication of distance between anchor and tag. As RSSI values are prone to many error sources like reflections of radio signals, these values should not be used to estimate position.

A.5 RS232 setting +UMRS

+UUDF				
Modules	NINA-B41X-40B			
Attributes	**Syntax**	**Settings saved**	**Can be aborted**	**Response time**
	Full	Yes	No	-

A.5.1 Description

Read and set RS232 settings.

A.5.2 Syntax

AT Command	Description
`AT+UMRS?`	Reads current RS232 settings from the module.
`AT+UMRS=<baud_rate>[,<flow_control>]`	Applies new RS232 settings.

A.5.3 Defined values

Parameter	Type	Description
baud_rate	Integer	Factory default value 115200 Valid baud rates: 19200, 28800, 31250, 38400, 56000, 57600, 76800, 115200, 230400, 250000, 460800, 921600, 100000
flow_control	Integer	0: CTS/RTS not used. 1 (factory default): CTS/RTS used for flow control

A.5.4 Notes

This command requires a store (AT&W) and restart (AT+CPWROFF) to take effect.

A.6 Other supported AT commands

- Attention AT
- Manufacturer identification AT+GMI
- Model identification AT+GMM
- Software identification ATI9
- Software version identification AT+GMR
- Local address AT+UMLA
- Store current configuration AT&W
- Module switch off AT+CPWROFF
- Set to factory defined configuration AT+UFACTORY
- Enter FW update mode AT+UFWUPD (Only mode 0 and baud config supported.)
- Startup event +STARTUP

For detailed information about the AT commands listed above, line termination character and so on, see also the u-connectXpress AT commands manual [5].

B Glossary

Abbreviation	Definition
AoA	Angle of Arrival
AoD	Angle of Departure
CTE	Constant Tone Extension
RSSI	Received Signal Strength Indication
UICR	User Information Configuration Register

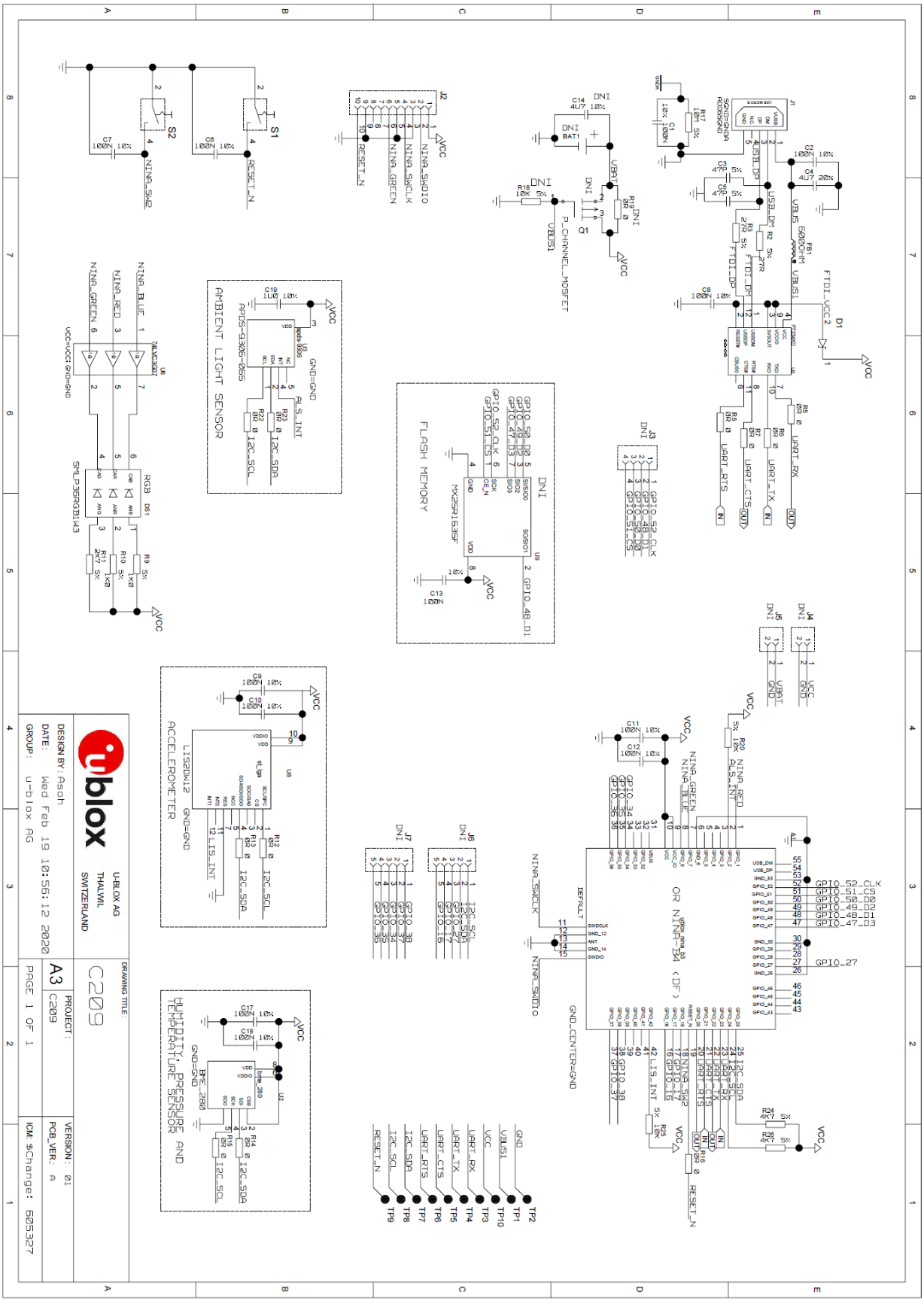
Table 5: Explanation of the abbreviations and terms used

C Limitations

The u-connectLocate software (1.2 and higher) supports tracking of up to 20 individual tags.

Tags that do not send any data for more than five seconds are considered as idle and are removed from the list of tracked tags. Newly identified tags are automatically added to the list of tracked tags.

D C209 schematics



Related documentation

[1] Bluetooth Direction Finding: A Technical Overview
https://www.bluetooth.com/bluetooth-resources/bluetooth-direction-finding/
[2] Bluetooth indoor positioning application note, UBX-21006395
[3] NINA-B41 product page, https://www.u-blox.com/en/product/nina-b41-series-u-connect
[4] NINA-B40 product page, https://www.u-blox.com/en/product/nina-b40-series-open-cpu
[5] u-connectXpress AT commands manual, UBX-14044127
[6] NINA-B4 system integration manual, UBX-19052230
[7] s-center, https://www.u-blox.com/en/product/s-center
[8] u-blox webinars, https://www.u-blox.com/en/webinar
[9] newtmgr download: https://mynewt.apache.org/latest/newtmgr/install/install_windows.html
[10] Newt Manager Guide: https://mynewt.apache.org/latest/newtmgr/index.html
[11] https://infocenter.nordicsemi.com/index.jsp?topic=%2Fug_nrfutil%2FUG%2Fnrfutil%2Fnrfutil_intro.html
[12] https://github.com/NordicSemiconductor/pc-nrfutil/releases
[13] https://github.com/u-blox/c209-aoa-tag
[14] NINA-B40 series data sheet, UBX-19049405
[15] NINA-B41 series data sheet, UBX-20035327
[16] EVK-ODIN-W2 user guide, UBX-16007132
[17] u-connectXpress software user guide, UBX-16024251
[18] s-center user guide, UBX-16012261
[19] https://www.u-blox.com/en/product/u-connectlocate
[20] FTDI website https://ftdichip.com/drivers/

☞ For product change notifications and regular updates of u-blox documentation, register on our website, www.u-blox.com.

Revision history

Revision	Date	Name	Comments
R01	05-Mar-2021	mape	Initial release
R02	28-Jun-2021	mape	Revised document title to reflect product scope including XPLR-AOA-x explorer kits. Added product description chapter and included hardware information describing anchors and tags. Introduced NINA-B411 and updated AT commands for u-connectLocate software.
R03	17-Sep-2021	mape	Corrected description of the RS232 command `AT+UMRS` in Anchor node AT commands. Improved C209 Software and flashing procedures and corrected link to the C209 GitHub repository,
R04	28-Oct-2021	mape	Added optimal distance anchor to tag in Running system. Added description of product variants for C211 in Overview. Added information about C211 Buttons. Corrected filter type example in Anchor node configuration.
R05	10-Jan-2022	mape	Added note about RSSI values to +UUDF events. Corrected information about optimal distance tag – anchor. Various minor improvements.
R06	28-Feb-2022	mape	Added Quick start guide and link to u-connectLocate software. Updated figure showing C209 main functional components in Overview.
R07	31-Mar-2022	mape	Added updates for u-connectLocate v1.2. Updated Limitations chapter. Added s-center flashing to Flashing and Installing tag software with the DFU bootloader. Updated UDFCFG tag 6. Described the Angle of Arrival tab in s-center.
R08	7-Apr-2022	mape	Added information about different XPLR-AOA-1 variants. Added.Ordering information.
R09	1-Jun-2022	mape	Removed ordering code information duplicated in Product description. Added note in Kit includes to describe the absence of LED arrays in later product versions. Updated contact information.

Contact

For further support and contact information, visit us at www.u-blox.com/support.