How Lansitec Bluetooth Beacon/Tag/Sticker Communicate

User Manuals · https://www.lansitec.com/fr/docs/bluetooth-beacons-documentation/bluetooth-beacons-user-manuals/comment-communiquent-les-etiquettes-autocollantes-bluetooth-lansitec/

A common misunderstanding is that a Lansitec Bluetooth sticker actively connects to random Bluetooth devices and uses them to transmit its location. This is not the case.

Bluetooth communication works in different modes: devices can **advertise (broadcast data)**, **receive (scan for data)**, or remain **inactive (silent)**.

Initial Configuration (Bidirectional Communication)

When the sticker is first powered on, it enters a short configuration window (approximately the first 3 minutes). During this period:

- The sticker **both advertises and receives** data.
- It can be **connected to by a Bluetooth device** (typically a Bluetooth gateway, Bluetooth bridge or smartphone).
- Configuration is done via apps such as *LightBlue* or any app that supports iBeacon.
- A password is required for secure setup.

This stage is **bidirectional communication**:

- Beacon/tag/Sticker → Bluetooth device
- Bluetooth device → Beacon/tag/Sticker

Normal Operation (Broadcast Only)

After the initial configuration period, the sticker switches to a power-saving mode:

- It **only advertises (broadcasts)** data
- It **no longer receives** any incoming Bluetooth messages
- This significantly reduces battery consumption

At this point, communication becomes **unidirectional**:

Sticker → Bluetooth device

The sticker continuously broadcasts iBeacon messages at defined intervals. These messages include identifiers such as:

- **UUID** (used to filter relevant devices)
- **Major and Minor values** (used to uniquely distinguish individual stickers)
- **Battery level and signal information**

Byte Offset	Default Value	Description	Properties
0	0x02	Data length – 2 bytes	Constant Preamble
1	0x01	Data type – flags	Constant Preamble
2	0x06	LE and BR/EDR flag	Constant preamble
3	0x1A	Data length – 26 bytes	Constant preamble
4	0xFF	Data type – Manufacturer specific data	Constant preamble
5	0x4C	Manufacturer data	Constant preamble
6	0x00	Manufacturer data	Constant preamble
7	0x02	Manufacturer data	Constant preamble
8	0x15	Manufacturer data	Constant preamble
9	0xF2	Proximity UUID 1st byte	User UUID
10	0xA5	Proximity UUID 2nd byte	User UUID
11	0x2D	Proximity UUID 3rd byte	User UUID
12	0x43	Proximity UUID 4th byte	User UUID
13	0xE0	Proximity UUID 5th byte	User UUID
14	0xAB	Proximity UUID 6th byte	User UUID
15	0x48	Proximity UUID 7th byte	User UUID
16	0x9C	Proximity UUID 8th byte	User UUID
17	0xB6	Proximity UUID 9th byte	User UUID
18	0x4C	Proximity UUID 10th byte	User UUID
19	0x4A	Proximity UUID 11th byte	User UUID

20	0x83	Proximity UUID 12th byte	User UUID
21	0x00	Proximity UUID 13th byte	User UUID
22	0x14	Proximity UUID 14th byte	User UUID
23	0x67	Proximity UUID 15th byte	User UUID
24	0x20	Proximity UUID 16th byte	User UUID
25	AA	Major 1st byte	Major value
26	BB	Major 2nd byte	Major value
27	CC	Minor 1st byte	Major value
28	CC	Minor 2nd byte	Major value
29	0xB3	Signal power (Calibrated SSI@1m)	Signal power value

How Other Devices with the Sticker

Although the sticker broadcasts its signal openly, this does **not** mean all nearby devices will process it:

- Most Bluetooth devices (phones, laptops, etc.) will **ignore the signal** unless specifically programmed to look for it
- Only devices configured to scan for the sticker's **UUID** will process the data
- Even then, each sticker is uniquely identified by its **Major and Minor values**, allowing precise tracking

The sticker does **not** randomly connect to nearby devices or "use" them to relay its location. Instead:

- It simply **broadcasts its data.**
- Compatible devices or gateways **choose whether to listen and process it**
- Filtering (via UUID) ensures only relevant devices respond

All Lansitec Bluetooth gateways are designed to support this filtering and identification mechanism. Actually, you can filter all Bluetooth data in the 31 bytes package you want and send whatever data you want in the package. For detailed information please refer to the datasheet attached.