Barcode scanning via Bluetooth

COBI.wms supports using barcode scanners that connect to the Android device through **Bluetooth 4.0** or later, also called Bluetooth Low Energy, Bluetooth LE, or shortly **BLE**.

There are some challenges involved when using a BLE scanner. The main problem is that there is no standardized Bluetooth specification for transmitting barcode-scan data. For this reason, most BLE scanners pretend to be a keyboard (**HID device**) and send the scanned barcodes as a series of key-press events.

For this reason, we strongly recommend using an Android device with an integrated barcode scanner instead of connecting an external scanner via Bluetooth. Nonetheless, this page describes the challenges involved and possible workarounds.

Using scanners that operate as a HID device

If you want to use a BLE scanner in HID mode, you must pair the device in the Android Settings (outside of COBI.wms), make sure that it's registered as a keyboard, and then within COBI.wms you must enable the **app option** called **“Barcode scan per keyboard input.”**

While this should work for plain (non-GS1) barcodes, it might not work with GS1 barcodes such as GS1 DataBar, GS1-128, GS1 DataMatrix, and GS1 QR Code.

The reason for this is complex. GS1 barcodes contain a special symbol called the “Function Code 1” (**FNC1**) which both appears as the beginning of the barcode to indicate that it's a GS1 barcode, and may appear in various places within the barcode to separate data fields. For example, if there is a batch number field followed by a serial number field, and the batch number does not use all of the 20 digits that are allowed by the GS1 standard, then it must be ended with FNC1 before the next field (serial number) begins.

The problem is: the FNC1 symbol does not correspond to any letter, digit or other key on a keyboard, and therefore the scanner in HID mode may not transmit FNC1 symbols, which makes supporting GS1 barcodes with that scanner impossible.

Some scanners may nevertheless send the FNC1 symbol as a special key-combination (for example using the CTRL key). This is known to work with BLE scanners from the brand **Inateck** (tested with the model **BCST-40**). But please note that COBISOFT cannot guarantee that these devices are suitable for industrial use.

In COBI.wms, there is a workaround to this problem using pseudo-GS1 barcodes which are not standards-compliant. The semicolon symbol (**;**) can be used in places where the FNC1 would normally appear. For example, if you encode the following text as a Code 128 barcode or a DataMatrix code, then COBI.wms will recognized it as a GS1 barcode:

<html><pre> ;24012345;10BATCHXYZ;21SERIALXYZ </pre></html>

As you see, the text begins with a semicolon, which is to indicate that it's a GS1 code. Later, it uses

semicolons to end the data fields 240 (item code) and 10 (batch number).

If you use this workaround with a one-dimensional barcode format such as Code 128, it might result in a very long barcode which may be difficult to scan. The reason is that special symbols like the semicolon are difficult to encode in Code 128 and produce a much longer barcode than if the FNC1 symbol was used.

If you use a two-dimensional encoding like DataMatrix or QR Code then it should work fine. But please note that this workaround is a specialization for COBI.wms only, and it will not work with any other barcode scanning system unless it is programmed specifically to recognize these “fake” GS1 codes.

Using scanners with a custom GATT Service

In Bluetooth LE terminology, **GATT** is an abbreviation for Generic Attribute Profile and refers to a system that allows BLE devices (e.g. barcode scanner) and host device (e.g. Android tablet) to exchange any kind of specialized data by defining a set of **Services** and **Characteristics** for data exchange.

If a BLE scanner supports a specialized GATT Service/Characteristic for transmitting barcode scan events, then COBI.wms might be able to support it without needing to rely on HID mode.

To make use of this mode of operation, you have to pair the BLE device from within COBI.wms by opening the Bluetooth screen and finding the device there.

So far, there is only one brand of BLE scanner supported by COBI.wms in this way, which is **Eyoyo**. (Tested with the **EY-R30 2D** finger scanner model.) Sadly this model does not transmit the FNC1 symbol explained in the previous section, which makes supporting GS1 barcodes impossible, and the model might not be suitable for industrial use.

From:
https://docs.cobisoft.de/wiki/ - **COBISOFT Documentation**

Permanent link:
https://docs.cobisoft.de/wiki/cobi.wms/bluetooth_scanning

Last update: **2022/05/16 10:41**