

CB-OLS425 CB-OLS426 THROUGHPUT AND LATENCY OPTIMIZATIONS APPLICATION NOTE

Document Revision

Document number: 23330818

Release: Mar 28, 2014 12:23

Document version: 1

Copyright © 2014 u-blox AG. The contents of this document can be changed by u-blox AG without prior notice and do not constitute any binding undertakings from u-blox AG. u-blox AG is not responsible under any circumstances for direct, indirect, unexpected damage or consequent damage that is caused by this document. All rights reserved. All brand and product names are trademarks or service marks of their respective owners.

1 Abstract

The throughput and latency between an OLS425/426 and a Bluetooth low energy device with central role are affected by, among other things, the connection parameters *Connection interval* and *Connection latency*. These parameters are ultimately controlled by the central. When the OLS425/426 is used with an OBS421 as central unit, the user can choose these parameters by issuing ATS commands on the OBS421. But if the OLS425/426 is used together with another central, for example a iPhone or Android unit, it is this central that decides which parameters to use. This application note describes how to attempt to control these parameters from the OLS425/426.

2 Table of contents

- 1 [Abstract](#)
- 2 [Table of contents](#)
- 3 [Related documents](#)
- 4 [Connection parameters](#)
 - 4.1 [Connection interval](#)
 - 4.2 [Connection latency](#)
 - 4.3 [Connection timeout](#)
- 5 [How to affect connection parameters from the OLS425/426](#)
 - 5.1 [Connection Update Request](#)
 - 5.2 [S-register configuration](#)
 - 5.3 [Example: iPad mini](#)

3 Related documents

1. [Bluetooth Low Energy Serial Port Adapter - Getting Started](#)
2. [cB-OLS425 cB-OLS426 Firmware](#)
3. Bluetooth Accessory Design Guide Lines for Apple Products, <https://developer.apple.com/hardware/drivers/bluetoothdesignguidelines.pdf>

4 Connection parameters

4.0.0.0.1 Connection interval

The *Connection Interval* determines how often the peripheral (OLS425/426) and the central should communicate.

Example: If the connection interval is 100 ms, then the peripheral will communicate with the central every 100 ms. This means data can be sent every 100 ms in any direction. A lower connection interval means higher throughput and shorter latency, but at the same time consumes more power and vice versa.

4.0.0.0.2 Connection latency

The *Connection Latency* decides how many connection intervals the peripheral is allowed to avoid communication with the central. Default value is 0, meaning the peripheral have to communicate with the central at every connection interval.

Example: If the connection latency is set to 10 it means that the peripheral can miss 10 connection intervals before it has to communicate with the central, thus being able to reduce it's power consumption by going into deep sleep. This of course affects the throughput, but only in the direction from central to peripheral, since the peripheral is still *allowed* to communicate at every connection interval.

4.0.0.0.3 Connection timeout

The connection timeout is the time from the last exchange of data until the link is considered lost. The central will not try to reconnect to the peripheral until this time has passed, so for example in a situation where the peripheral moves in and out of range a lot this value should be set low.

5 How to affect connection parameters from the OLS425/426

5.0.0.0.4 Connection Update Request

The connection parameters are decided by the central and sent to the peripheral in the connection request. But, the peripheral can try to change the parameters of an existing connection by sending a *Connection Update Request*. A *Connection Update Request* contains new values for Connection Interval, Connection Latency and Connection Timeout. When this is received by the central, the central decides if it should use the new parameters or not and sends an answer back to the peripheral.

5.0.0.0.5 S-register configuration

When a connection is established, the connection parameters are checked against a list of s-register to determine if a *Connection Update Request* should be sent from the OLS425/426. The default setting is to accept the connection parameters decided by the central.

A *Connection Update Request* is sent from the OLS425/426 to the central if any of the following conditions are met:

- Connection interval is less than Accepted Connection Interval Min (S-reg 6011)
- Connection interval is more than Accepted Connection interval Max (S-reg 6012)
- Connection latency is less than Accepted Connection latency Min (S-reg 6013)
- Connection latency is more than Accepted Connection latency Max (S-reg 6014)
- Connection timeout is less than Accepted Connection timeout Min (S-reg 6015)
- Connection timeout is more than Accepted Connection timeout Max (S-reg 6016)

The values used in the connection update request are also configured through S-registers:

Connection Interval min (S-reg 6004)
Connection Interval max (S-reg 6005)
Connection latency (S-reg 6006)
Connection timeout (S-reg 6000)

As stated above, the parameters in the *Connection Update Request* can be accepted or ignored by the central. For Apple products, accepted connection parameters can be found in <https://developer.apple.com/hardware/drivers/bluetoothdesignguidelines.pdf>.

5.0.0.0.6 Example: iPad mini

The default connection parameters used by the iPad mini when connecting to an OLS425/426 are:

- Connection Interval 30 ms
- Connection Latency 0
- Connection Timeout 720 ms

These default values can be different on different central devices, this is just an example.

To increase the throughput we want to set the Connection Interval to 20 ms, which is the lowest accepted value according to <https://developer.apple.com/hardware/drivers/bluetoothdesignguidelines.pdf>.

To accomplish this, we issue the following AT command to the OLS425/426:

ATS6005=16

This sets our desired *Connection Interval Max* to 20 ms (The parameter value is multiplied by 1.25 to get milliseconds). This value will be used in the *Connection Update Request*.

To make sure the Connection Update Request is sent we issue the following AT command:

ATS6012=16

This sets the *Accepted Connection Interval Max* to 20 ms as well, which means that after a connection is successfully established a *Connection Update Request* will be sent to the central (since 30 ms is above the *Accepted Connection Interval Max*).