# OneTouch Glucometer
# BLE Protocol

| | |
|---|---|
| **Client:** | Appia Care Inc. |
| **Document:** | ONETOUCH-20-001-EN |
| **Prepared by:** | Santiago, IVULICH |
| | Tobías, LIFSCHITZ |
| **Date:** | 07/06/2020 |
| **Revision:** | B |

# 1.Contents


**Contents** **2**

**Scope** **3**

**Overview** **3**

Introduction 3

Document objectives 3

**Glucometer operation** **3**

GATT services and characteristics 3

**Packets** **4**

Decoded Packets 5

Time Get 5

Time Set 5

High Limit Get 5

High Limit Set 6

Low Limit Get 6

Low Limit Set 6

Total Record Count Get 6

Correct Record Count Get 7

Record Get By Index 7

Record Request By ID 7

Unknown Packets 8

UK 1 - R 8

UK 2 8

UK 3 - R 9

**References** **9**

# 2.Scope

The scope of this document is to decode the communication protocol used by a third-party commercial glucometer over Bluetooth Low Energy (BLE) technology. It is not within the scope of this document the determination of the statefulness of the protocol, as at the time of writing it's not known if the order of the packets affects the communication.

# 3.Overview

## 3.1.Introduction

The purpose of this document is to define and document the packets used to communicate with the Onetouch Select Plus Flex glucometer. A technique of reverse engineering was used observing the correlation between packets of the same length and knowledge on the field to decode the meaning of the unknown packets. This allowed to understand the communication protocol in a bottom up approach.

## 3.2.Document objectives

The key objective is to document the communication with the device, logging all the packets with the official app in the process. This will ease the development of the protocol during the implementation stage.

# 4.Glucometer operation

After scanning bluetooth communications, it was confirmed that the device does not implement the standard profile defined by Bluetooth SIG for healthcare devices which measure blood glucose levels [2].

Instead, the manufacturer designed the communication using a proprietary protocol running over an emulated serial port using BLE technology. Frequently called serial over BLE, it consists of an Attribute Protocol (ATT) made up of a single service containing one RX characteristic with only the notify/indicate permission, and one TX characteristic that only allows writing.

## 4.1.GATT services and characteristics

After scanning all services and characteristics exposed by the device using an Android tool[3] the result are presented below.

- Generic Access Profile | UUID: 0x1800

| Characteristic | UUID | Properties |
|---|---|---|
| Device Name | 0X2A00 | READ WRITE |
| Appearance | 0X2A01 | READ |
| Peripheral Preferred Connection Parameters | 0X2A04 | READ |

- Generic Attribute | UUID: 0x1801

| Characteristic | UUID | Properties |
|---|---|---|
| Service Changed | 0X2A05 | INDICATE |

- Device Information | UUID: 0x180A

| Characteristic | UUID | Properties |
|---|---|---|
| Manufacturer Name String | 0X2A29 | READ |
| Model Number String | 0X2A24 | READ |
| Serial Number String | 0X2A25 | READ |
| Software Revision String | 0X2A28 | READ |
| System ID | 0X2A23 | READ |

- Unknown Service 1 | UUID: 0xaf9df7a1-e595-11e3-96b4-0002a5d5c51b

| Characteristic | UUID | Description | Properties |
|---|---|---|---|
| Unknown Characteristic | 0x2902 | RX | NOTIFY |
| Unknown Characteristic | 0xaf9df7a2-e595-11e3-96b4-0002a5d5c51b | TX | WRITE, WRITE NO RESPONSE |

Note: 16 bit characteristic UUID is equal to 0x0000XXXX-0000-1000-8000-00805F9B34FB

The first detailed service is specified in BLE Core 4.0[4] and is the GAP service. GAP is an acronym for the Generic Access Profile, and it controls connections and advertising in Bluetooth. GAP is what makes the device visible to the outside world, and determines how two devices can (or can't) interact with each other.

The second service include characteristics specified in GATT Specification Supplement[5] which specifies standard characteristics not defined in Bluetooth Core Specification. In this case contains information detailed in the table above.

Finally the last service is vendor specific, reason why it's parsed as Unknown Service by the used tool, as well as their characteristics.

# 5.Packets

After making measurements and logging several interactions with the official app[6], and having sufficient data to analyze the following protocol was identified and decoded.

Because of limitations of BLE the messages are split in several blocks. This is why the first packet of a message starts with the total number of blocks. The following packets will have in the first byte

the index of the packet inside the message (e.g each packet of a long message divided in five packets will have in the first byte: 0x05 → 0x41 → 0x42 → 0x43 → 0x44 ).

Every packet is acknowledged with a byte that has 8 in its upper nibble and copies the low nibble of the first byte of the packet.
Examples:

Request:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 09 (9) | 00 | 04 | 20 (32) | 02 (2) | 03 | f9 | c3 |

Ack

| |
|---|
| 81 |

Request:

| | | | | |
|---|---|---|---|---|
| 42 | 00 | 03 | e3 | 3a |

Response:

| |
|---|
| 82 |

## 5.1.Decoded Packets

All packets are represented in hexadecimal numbers separated in bytes.

### 5.1.1.Time Get

Request the time of the device. Responds with the number of seconds since 2000-01-01.00:00.

Request:

| BLKS | - | LEN | PAD | DLM | CMD | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 09 | 00 | 04 | 20 | 02 | 03 | CS_L | CS_H |

Response:

| BLKS | - | LEN | PAD | DLM | - | TIMESTAMP | | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0c | 00 | 04 | 06 | TM_3 | TM_2 | TM_1 | TM_0 | 03 | CS_L | CS_H |

### 5.1.2.Time Set

Sets the device time, must be the number of seconds since 2000-01-01 00:00.

Request:

| BLKS | - | LEN | PAD | DLM | CMD | | TIMESTAMP | | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0d | 00 | 04 | 20 | 01 | TM_3 | TM_2 | TM_1 | TM_0 | 03 | CS_L | CS_H |

Response:

| BLKS | - | LEN | PAD | DML | - | DML | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 08 | 00 | 04 | 06 | 03 | 78 | c1 |

### 5.1.3.High Limit Get

Request the limit to display high glucose. Returns a two byte integer in little endian.

Request:

| BLKS | - | LEN | PAD | DLM | CMD | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0a | 00 | 04 | 0a | 02 | 0a | 03 | CS_L | CS_H |

Response:

| BLKS | - | LEN | PAD | DLM | - | High Limit | | PAD | PAD | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0c | 00 | 04 | 06 | HI_L | HI_H | 00 | 00 | 03 | CS_L | CS_H |

### 5.1.4.High Limit Set

Set the limit to display high glucose.

Request:

| BLKS | - | LEN | PAD | DLM | CMD | | | High Limit | | PAD | PAD | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0e | 00 | 03 | 0a | 01 | 0a | HI_L | HI_H | 00 | 00 | 03 | CS_L | CS_H |

Response:

### 5.1.5.Low Limit Get

Request the limit to display low glucose. Returns a two byte integer in little endian.

Request:

| BLKS | - | LEN | PAD | DLM | CMD | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0a | 00 | 04 | 0a | 02 | 09 | 03 | CS_L | CS_H |

Response:

| BLKS | - | LEN | PAD | DLM | - | Low Limit | | PAD | PAD | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0c | 00 | 04 | 06 | LO_L | LO_H | 00 | 00 | 03 | CS_L | CS_H |

### 5.1.6.Low Limit Set

Set the limit to display low glucose. Returns a two byte integer in little endian.

Request:

| BLKS | - | LEN | PAD | DLM | CMD | | | Low Limit | | PAD | PAD | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0e | 00 | 03 | 0a | 01 | 09 | LO_L | LO_H | 00 | 00 | 03 | CS_L | CS_H |

Response:

| BLKS | - | LEN | PAD | DLM | - | Low Limit | | PAD | PAD | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0c | 00 | 03 | 06 | LO_L | LO_H | 00 | 00 | 03 | CS_L | CS_H |

### 5.1.7.Total Record Count Get

Request the number of measurements including errors. Returns a two byte integer in little endian.

Request:

| BLKS | - | LEN | PAD | DLM | CMD | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0a | 00 | 04 | 0a | 02 | 06 | 03 | CS_L | CS_H |

Response:

| BLKS | - | LEN | PAD | DLM | PAD | Count | | PAD | PAD | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0c | 00 | 04 | 06 | CNT_L | CNT_H | 00 | 00 | 03 | CS_L | CS_H |

### 5.1.8.Correct Record Count Get

Request the number of correct measurements. Returns a two byte integer in little endian.

Request:

| BLKS | PAD | LEN | PAD | DLM | CMD | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 09 | 00 | 04 | 27 | 00 | 03 | CS_L | CS_H |

Response:

| BLKS | PAD | LEN | PAD | DLM | PAD | COUNT | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0a | 00 | 04 | 06 | CNT_L | CNT_H | 03 | CS_L | CS_H |

### 5.1.9.Record Get By Index

Request a measurement by its index. Returns its ID as a two byte integer in little endian, its Value as a two byte integer in little endian and a timestamp of seconds since 2000-01-01 00:00.

Request:

| BLKS | - | LEN | PAD | DLM | CMD | | INDEX | | PAD | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0c | 00 | 04 | 31 | 02 | IND_L | IND_H | 00 | 03 | CS_L | CS_H |

Response:

Block-1

| BLKS | - | LEN | PAD | DLM | - | INDEX | | PAD | ID | |
|---|---|---|---|---|---|---|---|---|---|---|
| 02 | 02 | 18 | 00 | 04 | 06 | IND_L | IND_H | 00 | ID_L | ID_H |

| TIMESTAMP | | | | VALUE | | ERROR CODE | | |
|---|---|---|---|---|---|---|---|---|
| TM_3 | TM_2 | TM_1 | TM_0 | VAL_L | VAL_H | ERR_1 | ERR_2 | ERR_3 |

Block-2

| - | ERROR CODE | | DLM | CHECKSUM | |
|---|---|---|---|---|---|
| 41 | ERR_4 | ERR_5 | 03 | be | d8 |

### 5.1.10.Record Request By ID

Request a measurement by its ID. Returns its value as a two byte integer in little endian and a timestamp of seconds since 2000-01-01 00:00.

Request

| BLKS | - | LEN | PAD | DLM | CMD | REC_ID | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0a | 00 | 04 | B3 | ID_L | ID_H | 03 | CS_L | CS_H |

Response

| BLKS | - | LEN | PAD | DLM | - | TIMESTAMP | | | | VALUE | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 13 | 00 | 04 | 06 | 75 | 190 | 106 | 38 | a8 | 00 |

| ERROR CODES | | | | | DLM | Checksum | |
|---|---|---|---|---|---|---|---|
| ERR_1 | ERR_2 | ERR_3 | ERR_4 | ERR_5 | 03 | CS_L | CS_H |

## 5.2.Unknown Packets

The following packets could not be decoded due to the rarity of appearance and complexity.

### 5.2.1.UK 1 - R

This packet is believed to be the offset of the available ids, as the devices stores only 500 measurements but can give 65536 unique ids. This can be confirmed with further testing, but at the time of writing this document the device holds only 60 measurements.

Packet :

| BLKS | - | LEN | PAD | DLM | CMD | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0a | 00 | 04 | 09 | 02 | 02 | 03 | c6 | 0f |

Response:

| BLKS | - | LEN | PAD | DLM | - | Always 0x00000000 | | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0c | 00 | 04 | 06 | 00 | 00 | 00 | 00 | 03 | e7 | f6 |

### 5.2.2.UK 2

Request:

| Block | - | LEN | PAD | DLM | CMD | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 0a | 00 | 04 | e6 | 02 | 08 | 03 | 09 | b0 |

Response:

| BLKS | - | LEN | PAD | DLM | - | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 03 | 02 | 2a | 00 | 04 | 06 | 41 | 00 | 44 | 00 |
| | | | | | | | | | |
| 42 | 00 | 38 | 00 | 42 | 00 | 46 | 00 | 39 | 00 |
| | | | | | | | | | |
| 41 | 36 | 00 | 34 | 00 | 41 | 00 | 37 | 00 | 41 |
| | | | | | | | | | |
| 00 | 31 | 00 | 30 | 00 | 43 | 00 | 41 | 00 | 00 |

| | | DLM | CHECKSUM | |
|---|---|---|---|---|
| 42 | 00 | 03 | c5 | a1 |

### 5.2.3.UK 3 - R

Request:

| BLKS | - | LEN | PAD | DLM | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 02 | 02 | 18 | 00 | 04 | 11 | 5a | c6 | 27 | 6f | 1c |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1c | f2 | 3f | 83 | e8 | 7f | 13 | c6 | 0c | 7b |

| | | | DLM | CHECKSUM | |
|---|---|---|---|---|---|
| 41 | e1 | 16 | 03 | 8c | 10 |

Response:

| BLKS | - | LEN | PAD | DLM | - | DLM | CHECKSUM | |
|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 08 | 00 | 04 | 06 | 03 | 78 | c1 |

# 6. References


[2] 11073-10417-2015 - IEEE Health informatics -- Personal health device communication Part 10417: Device Specialization -- Glucose Meter

[3] nRF Connect for Mobile.

[4] Bluetooth Core Specification

[5] GATT Characteristics | Bluetooth® Technology Website

[6] OneTouch Reveal® mobile app for Diabetes - Apps on Google Play