



R-IoT Wireless Latency Tests

Hardware Tests and Data Analysis
[2020-02-01]

PLUX Wireless Biosignals S.A.

email: plux@plux.info

web: <http://www.plux.info>

Headquarters

Zona Industrial das Corredouras, Lt. 14 – 1º

2630-369 Arruda dos Vinhos

Portugal

tel.: +351 263 978 572

fax: +351 263 978 902

Lisbon Office

Av. 5 de Outubro, nº 79 – 2º

1050-059 Lisboa

Portugal

tel.: +351 211 956 542

fax: +351 211 956 542

Version	Author	Date	Page
1.0	William Primett	2020-02-19	2 / 10

Modifications Applied to the Document

Version	Author	Pages	Date	Description
1.0	William Primett	All	2020-02-19	Document Creation

Table of Contents

Introduction	4
A An overview of the BITalino R-IoT	4
1 Device Overview	4
2 Wireless Communication	4
B Experimental Protocol	5
1 Experiment Objectives	5
2 Method	5
2.1 Reference Signal and Timestamps	5
2.2 Multiple Device Setup	6
2.3 Device Firmware	6
C Results & Discussion	6
1 Conditions	6
2 Single Device Test	6
2.1 Data Analysis	6
3 Multiple Device Test – 4 Devices	7
3.1 Configuration	7
3.2 Data Analysis	7
4 Discussion	9
4.1 Setup Considerations	9
4.2 Latency Assessment	9
Bibliography	10

Introduction

The BITalino R-IoT board was released in 2017 as an adaptation of the module originally introduced by IRCAM (<http://ismm.ircam.fr/riot/>). It is designed to sense motion amongst other physiological signals, as it is compatible with the standard range of BITalino sensor modules (<https://plux.info/15-sensors>). The BITalino R-IoT intends to be a robust tool that's suitable for wearable applications and performance environments given its compact form factor and ability to transmit data over a Wi-Fi network via Open Sound Control (OSC) messages. This infrastructure for wireless communication facilitates the monitoring of multiple devices simultaneously, which can be exploited in interaction environments with multiple users, or in use-cases that require sensors to be applied different parts of the body. With a focus on real-time interaction, we consider the latency that occurs when multiple R-IoT modules are setup to communicate to a single host computer. In this report, we will outline an experimental protocol to assess the time taken for the module to react to a stimulus signal and consider how this is affected when more devices are included in the network.

A An overview of the BITalino R-IoT

1 Device Overview

The BITalino R-IoT module is the 7th generation of IRCAM's wireless sensor digitizing unit, an essential tool linking motion sensing, gestural recognition and Live Performance Arts. Since 2001, IRCAM aimed to provide a low latency, high data-rate & resolution and stage compatible devices, capable of streaming gestural information to the computer from multiple performers.

The BITalino R-IoT embeds a 9-axis digital IMU sensor (LSM9DS1) [1] featuring a 3-axis accelerometer, a 3-axis gyroscope and a 3-axis magnetometer, allowing for instance the onboard computation of the absolute orientation of the module in space. The sensor is attached to the SPI port to sample the 16-bit motion data at high speed. In addition to this, there are two 12-bit ADC inputs, compatible with the BITalino sensor modules.

To ease the reading of this document, we will refer from now to the BITalino R-IoT as R-IoT. More information about the device can be found in the official documentation¹.

2 Wireless Communication

Out of the box, the R-IoT is programmed to transmit data over Open Sound Control (OSC). This is a communication protocol for communication among computers, sound synthesizers, and other multimedia devices that is optimized for modern networking technology.

Features:

- Open-ended, dynamic, URL-style symbolic naming scheme
- Symbolic and high-resolution numeric argument data
- Pattern matching language to specify multiple recipients of a single message
- High resolution time tags
- "Bundles" of messages whose effects must occur simultaneously

For further information please visit <http://opensoundcontrol.org/>.

With the default configuration, each device will send an array of 21 float values every 5 milliseconds. This array includes motion data from the 9-axis IMU, as well as the analog inputs when are coupled with an OSC message address that includes the ID number of the device. We can assign individual ID numbers to each device in the network to separate the incoming data within the host computer.

¹ BITalino https://bitalino.com/docs/MANUAL_RIOT_V1.2.pdf

Version	Author	Date	Page
1.0	William Primett	2020-02-19	4 / 10

B Experimental Protocol

1 Experiment Objectives

In our tests, we calculate the time taken for the host computer to receive a response to a stimulus which changes in the analog input on the R-IoT device. The stimulus signal is distributed to multiple devices which are included in the network to see the effect this has on latency. The test is run for a duration of several minutes to assess how the latency deviates over time.

2 Method

2.1 Reference Signal and Timestamps

The stimulus is originally triggered from the host computer, which sends single byte messages to a teensy 3.2² board via USB serial. Upon receiving a message from the host computer, the teensy board will output either 3.3 or 0 volts that is put through a voltage divider before reaching the analog input of the R-IoT device. The R-IoT then sends OSC packets (containing the analog input value) back to the host computer via a TP-Link MR3020 Wi-Fi router. The stimulus signal is set to switch every second, resulting in an oscillation of 0.5 Hz.

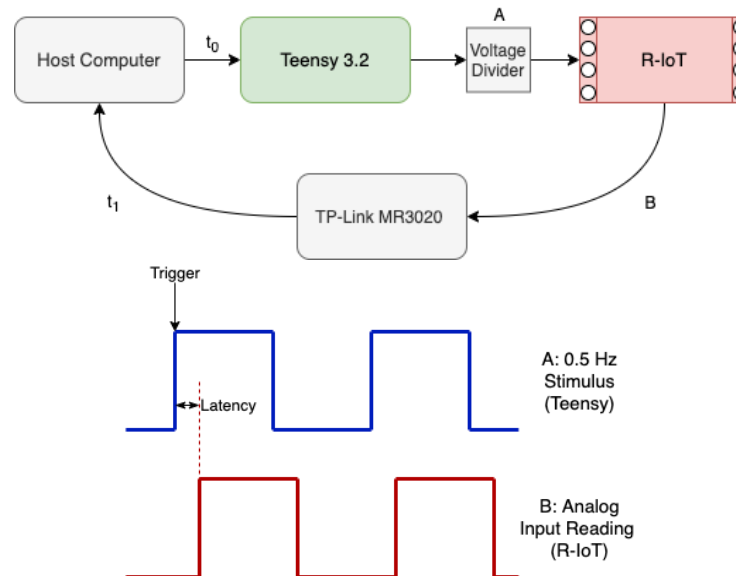


Figure 1 Experimental Architecture for Latency Tests

A timestamp is logged at t_0 , the moment the trigger message is sent from the host computer. Another is logged at t_1 , when the change is received in the incoming OSC message. From here, we define the latency as $t_1 - t_0$ for each instance of a trigger message.

This approach is partially inspired by the closed-loop latency tests performed by Sebastian Madgwick for the x-OSC device [2], where a square wave is fed into the analog input of the device and outputted into a frequency counter to measure latency. We acknowledge that this method is not as accurate due to the small latency that occurs between the host computer and the Teensy, which is covered in section [section]. However, we consider this a low-cost solution that can be more easily replicated for the user's specific conditions and environment.

² <https://www.pjrc.com/store/teensy32.html>

Version	Author	Date	Page
1.0	William Primett	2020-02-19	5 / 10

2.2 Multiple Device Setup

To test multiple devices simultaneously, we feed to the same reference signal from the Teensy to many R-IoTs. The devices are configured to stream data through the same Wi-Fi router, where the data is addressed by the ID number of each device.

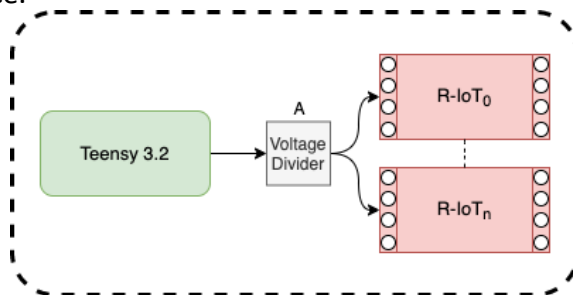


Figure 2 Feeding Reference Signal to Multiple R-IoTs

2.3 Device Firmware

Each device is flashed with the default firmware, which implements algorithms to compute the absolute orientation of the device from the raw IMU data, proving quaternions and Euler angles every time a data packet is transmitted. For this reason, our tests can be considered to account for the latency that occurs when retrieving motion data.

C Results & Discussion

1 Conditions

For these tests, the device(s) were first placed less than 1 metre away from the router which was connected to the host via an ethernet connection. From the test, we extracted 248 triggers form a total duration of approximately 8 minutes and 15 seconds. We then repeated the tests with the device(s) placed 5 metres away from the router.

2 Single Device Test

2.1 Data Analysis

For the duration of the test, we accumulate the latencies for every trigger and calculate a minimum, maximum and mean average latency. The results are given in Table 1. The results span between 6 to 11 milliseconds. When we repeat the test at greater distance, we see a greater range of values and a higher average, as shown in

Device ID	Maximum Latency (ms)	Minimum Latency (ms)	Mean Latency (ms)
0	20.603	7.077	10.702

Table 2 and Figure 3. Over the duration of the acquisitions, we could not observe any increasing/decreasing trend in the results.

Table 1 Latency results for one device (1 metre)

Device ID	Maximum Latency (ms)	Minimum Latency (ms)	Mean Latency (ms)
0	11.019	6.605	8.859

Table 2 Latency results for one device (5 metres)

Device ID	Maximum Latency (ms)	Minimum Latency (ms)	Mean Latency (ms)
0	20.603	7.077	10.702

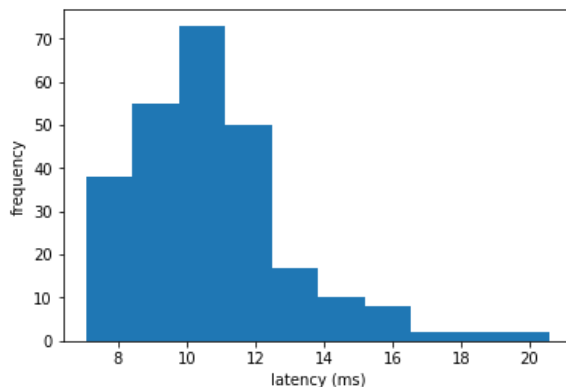


Figure 3 Histogram of latency values (5 metres)

3 Multiple Device Test – 4 Devices

3.1 Configuration

To test multiple devices, we pair each R-IoT to the same router and separate the incoming data according to the ID number of the device. All devices are sending data to the same port number. Other conditions are kept the same as the previous test.

3.2 Data Analysis

3.2.1 Absolute Latency

The minimum, maximum and mean latencies for the individual devices are shown in Table 3. We can see the maximum and mean latency has increased in all instances when more devices are added to the network. The results are fairly consistent between each device aside from number 2, which doesn't reach as high of a maximum. From all 992 triggers, the latency durations ranged from 6.55 to 17.31 milliseconds, and the mean average was calculated as 10.03 milliseconds. In Figure 4 and Figure 5, we show these plotted as a histogram.

When we repeat the test at 5 metres, we calculate a small increase in the average latency and a higher maximum amongst all the devices.

Table 3 Latency results for four devices (1 metre)

Device ID	Maximum Latency (ms)	Minimum Latency (ms)	Mean Latency (ms)
0	16.243	6.705	10.011
1	17.201	6.657	10.007
2	14.877	6.550	9.998
3	17.305	6.868	10.130

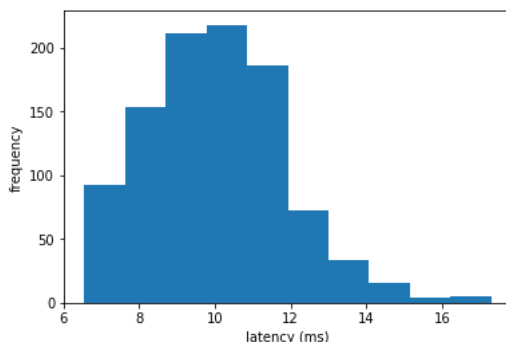


Figure 4 Histogram of absolute latencies (1 metre)

Table 4 Latency results for four devices (5 metres)

Device ID	Maximum Latency (ms)	Minimum Latency (ms)	Mean Latency (ms)
0	27.992	6.720	11.023
1	27.559	7.248	11.916
2	27.990	6.857	11.263
3	26.346	6.837	10.830

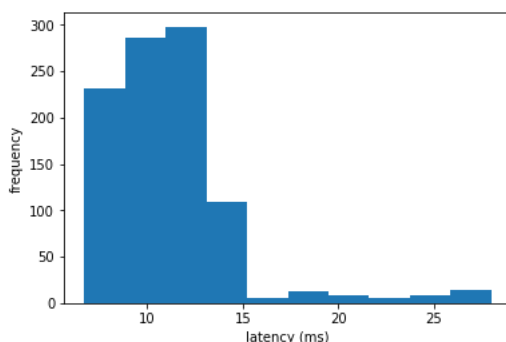


Figure 5 Histogram of absolute latencies (5 metres)

3.2.2 Relative Latency

As well as the response time between each individual device and the host computer, we are also interested in the difference in latency of the received message between all the devices in the network. This is important to enable data synchronisation between sources. For each trigger, we calculate the difference between the first and the last response out of the 4 devices. From our first acquisition, the mean average latency difference was calculated as 3.22 milliseconds with a maximum 8.96. The data is plotted as a histogram to assess the likelihood of these latency differences occurring. When we increase the distance, we find the maximum relative latency goes up to 20.148 milliseconds, and we calculate an increased average of 5.925 milliseconds.

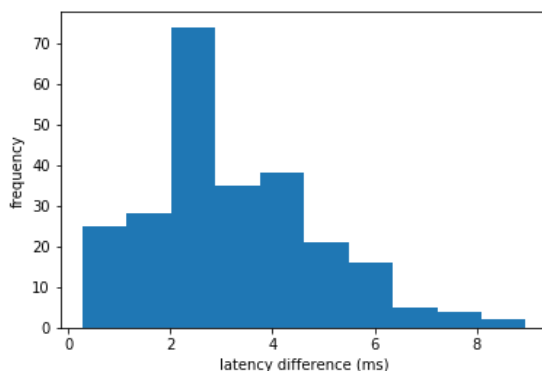


Figure 6 Histogram of latency differences (1 metre)

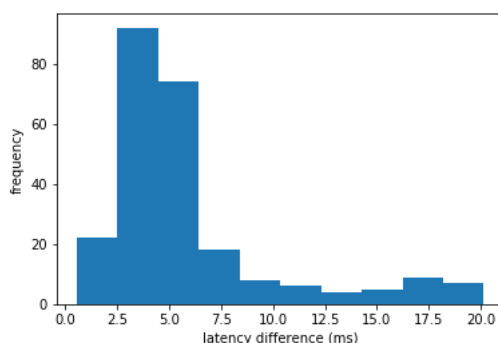


Figure 7 Histogram of latency differences (5 metres)

4 Discussion

4.1 Setup Considerations

There are a couple of factors regarding wireless configuration that are worth noting. First is that all the devices were all sending to the same port number (8888). We did this for convenience and ease of programming. However, separating ports for each device may have an effect of performance [3]. Secondly, we set a WPA-2 password on the network which uses more bandwidth.

Another factor that is briefly addressed in the document is the added latency between the host computer and the Teensy 3.2, connected by USB. As the communication was dependent on single byte messages, we assume that this is relatively insignificant to the final results. Some more information can be found in the following article³.

4.2 Latency Assessment

From the results, we conclude the following factors to have an impact on the latency between the R-IoT and the host computer. Having more devices in the network will increase the average latency and maximum latency. Additionally, increasing the distance between the R-IoT(s) and the Wi-Fi router will also increase the average latency. When we tested four devices at 5 metres, we observed a few instances of the latency being much higher than the average, compared to when we tested at a short distance. We also found that the distance would affect the difference in latency between devices, which would have an impact on data synchronization.

³ <http://neophob.com/2011/04/serial-latency-teensy-vs-arduino/>

Version	Author	Date	Page
1.0	William Primett	2020-02-19	9 / 10

Bibliography

- [1] STMicroelectronics, “LSM9DS1 iNEMO inertial module: 3D accelerometer, 3D gyroscope, 3D magnetometer,” 2014. [Online].
- [2] S. Madgwick and T. Mitchell, “x-OSC: A Versatile Wireless I/O Device For Creative/Music Applications,” 2013.
- [3] P. W. B. S.A., “BITalino R-IoT User Manual,” 2020. [Online]. Available: https://bitalino.com/docs/MANUAL_RIOT_V1.2.pdf.

Version	Author	Date	Page
1.0	William Primett	2020-02-19	10 / 10