

Review

# Power Consumption Analysis of Bluetooth Low Energy Commercial Products and Their Implications for IoT Applications

Eduardo Garcia-Espinosa <sup>1,2,\*</sup>, Omar Longoria-Gandara <sup>1</sup>, Ioseth Pegueros-Lepe <sup>1,2</sup> and Arturo Veloz-Guerrero <sup>1</sup>

- <sup>1</sup> Department of Electronics, Systems and Computer Science, Instituto Tecnológico de Estudios Superiores de Occidente, San Pedro Tlaquepaque 45604, Mexico; olongoria@iteso.mx (O.L.-G.);  
ioseth.i.pegueros.lepe@intel.com (I.P.-L.); arturo.veloz@gmail.com (A.V.-G.)
- <sup>2</sup> Intel Guadalajara Design Center, Zapopan 45019, Mexico
- \* Correspondence: e.garcia-espinosa@ieee.org; Tel.: +52-33-1645-2250

Received: 13 October 2018; Accepted: 28 November 2018; Published: 4 December 2018



**Abstract:** Internet of things (IoT) has become a very important business segment for the industry since it is estimated that 50 billion devices will be interconnected by 2020, where 30% of them are forecasted to use Bluetooth low energy as the enabling communication protocol. IoT involves smart devices that can be connected to the cloud in order to share and process all types of data collected through sensors. Moreover, 100% of smartphones and tablets shipped in 2018 include Bluetooth low energy, which will allow the interoperation with the upcoming 5G standard. As a result, different low-power hardware platforms have become available to either build prototypes, or implement full IoT solutions. One fundamental aspect while choosing a development platform is to analyze the power consumption of its components to build an optimal low-power application and extend the battery life. In this work, we present current consumption measurements of four Bluetooth low energy commercial platforms configured as central and as peripheral devices, these measurements are aimed to provide the data that are not available in the datasheets and which is needed to select, in an optimal way, the proper connection parameters for the protocol.

**Keywords:** Bluetooth low energy; Internet of things; power consumption

---

## 1. Introduction

Internet of things (IoT) is a new paradigm of the Industrial Internet Consortium focused on enabling the adoption of internet technologies, which includes different kinds of devices: Computers, tablets, smartphones, cyber-physical systems, and everyday objects. These devices include embedded sensors that share information between each other and can be controlled via the Internet [1]. With this in mind, a regular object, such as a lamp or a coat with embedded electronics such as sensors will become a smart object. Therefore, an IoT environment will use these smart objects as the main components to enable businesses, industries, and consumers to connect to the Internet for sharing and analyzing data. Given these points, the estimation is that 50 billion devices will be connected by 2020 [1], thus posing considerable market opportunities for those companies investing in IoT [2].

Consequently, different vendors have made hardware platforms available, both for production and prototyping that allow the implementation of IoT applications. One example of a prototyping platform is the Arduino 101 board, based on an Intel® Curie™ system on chip (SoC), which includes an Intel® Quark™ microcontroller unit (MCU) and a BLE module, among other components. Other examples of prototyping platforms are the BLE Texas Instruments® (TI) CC2540 and the Nordic® NRF51822 SoC.

It is important to note that the Bluetooth low energy (BLE) protocol is widely used as the wireless communication protocol for IoT applications and is included in the majority of the hardware platforms available in the market as well as in smartphones, tablets and laptops. Another key point is that the smartphones are used by several IoT applications as a gateway for Internet access. As a result, and despite the fact that the fifth generation (5G) network will contribute by itself to the connection of billions of devices in the future, the BLE based applications can still coexist and use the 5G network as a gateway.

The BLE protocol defines a network called piconet [3], formed by a central device (CD) or master and one or more peripheral devices (PD) or slaves. In this case, the CD can be a smartphone or a personal computer, while a PD would be a sensor device. In the piconet, the CD and PD are bound together in the advertising state and once they are connected, they will exchange data packets in a periodical time interval, which is known as a connection event. In general, a CD battery can be easily charged, and the power consumption of a PD becomes a critical parameter, since such devices are required to operate continuously using coin cell batteries. In other use cases, both the CD and PD would need to operate using coin cell batteries, and therefore there is the need to be aware of the power consumption of the platform to prevent the battery drain [3]. As an illustration, during the advertising and connection events, a BLE device transitions different states before the transmission of data takes place, such as wake-up, receive data, transmit data, processing, and sleeping states. Therefore, instead of the peak current, the average current is required to determine the battery life of a BLE device [4].

With this in mind, since the BLE protocol was developed to operate with coin cell batteries that may last from months to several years depending on the application [3], there is the need to understand the power consumption of the available BLE commercial products to implement an optimal IoT application. Moreover, such information is either not available in the datasheets or the data are scarce and does not provide a design guideline for using different connection parameters in an optimal way.

In this work, we describe the required setup to measure the current consumption of four BLE commercial platforms, focusing on the consumption of the system while the platform is being configured as a BLE CD and as a PD. To the best of our knowledge, this work is the first to show the current consumption of a CD. Additionally, a guide for IoT designers is provided to calculate the battery life for several connection intervals that will help to deliver a much more dependable low energy IoT system.

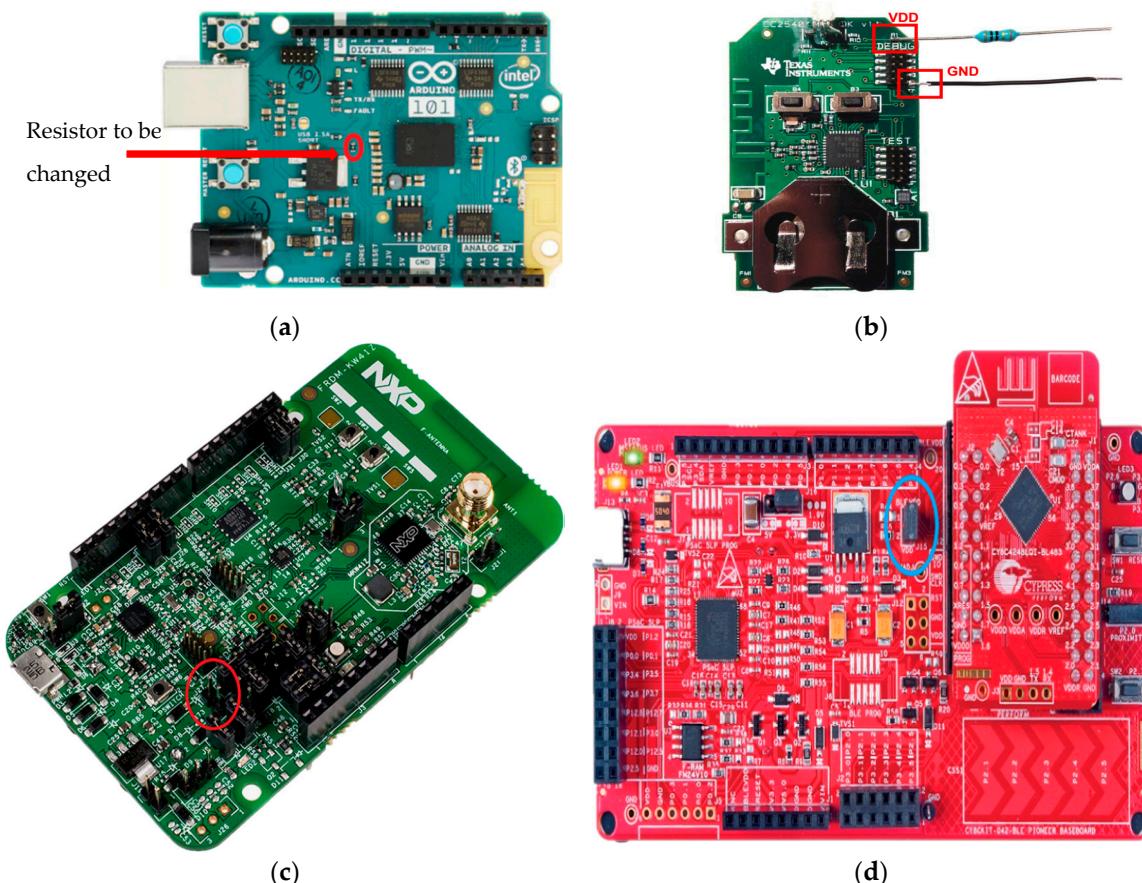
## 2. Materials and Methods

### *Experimental Setup*

Four BLE commercial platforms were used to perform the power measurements: Arduino 101 (Intel A-101), Texas Instruments (TI) CC2540, NXP FRDM-KW41Z, and Cypress Semiconductor CY8CKIT-042-BLE-A.

In order to perform the measurements, the first step is to perform a hardware modification in the platforms in order to measure the average current. Such modifications are as follows:

- Intel A-101 requires to replace a zero-ohm resistor (R58) by a  $10\ \Omega$  resistor, as shown in Figure 1a.
- The TI CC2540 requires to add a  $10\ \Omega$  resistor, as shown in Figure 1b. The figure is taken from Reference [4].
- FRDM-KW41Z can be powered by a coin cell, in which case, a  $10\ \Omega$  resistor can be placed in the jumper J27, as shown in Figure 1c.
- CY8CKIT-042-BLE-A requires a  $10\ \Omega$  resistor in jumper J15 as shown in Figure 1d.



**Figure 1.** The four platforms used in the experiments and the required hardware modifications: (a) Intel A-101, red circle shows the location of the resistor to be changed; (b) TI CC2540; (c) NXP FRDM-KW41Z, red circle shows the location of the jumper where the resistor can be placed; and (d) Cypress CY8CKIT-042-BLE-A, where the blue circle shows the location of the jumper where the resistor can be placed.

The next step is to load the application software in both the CD and PD. In this case, the software configures the devices to perform the advertising event, followed by the connection event.

In order to get familiar with the software stack from each platform, the following setup was used along with the built-in examples provided in the software stack for each platform:

1. A smartphone configured as the CD, with the CySmart application from Cypress and one CY8CKIT-042-BLE-A platform configured as PD
2. A smartphone configured as the CD, with the IoT Toolbox application from NXP and one FRDM-KW41Z platform configured as PD
3. An Intel A-101 configured as the CD and one CC2540 and another Intel A-101 boards were used as PDs in a one-to-one configuration
4. A CY8CKIT-042-BLE-A configured as a CD and the rest of the platforms as PDs
5. Finally, two FRDM-KW41Z platforms were used, one configured as a CD and the other one as a PD.

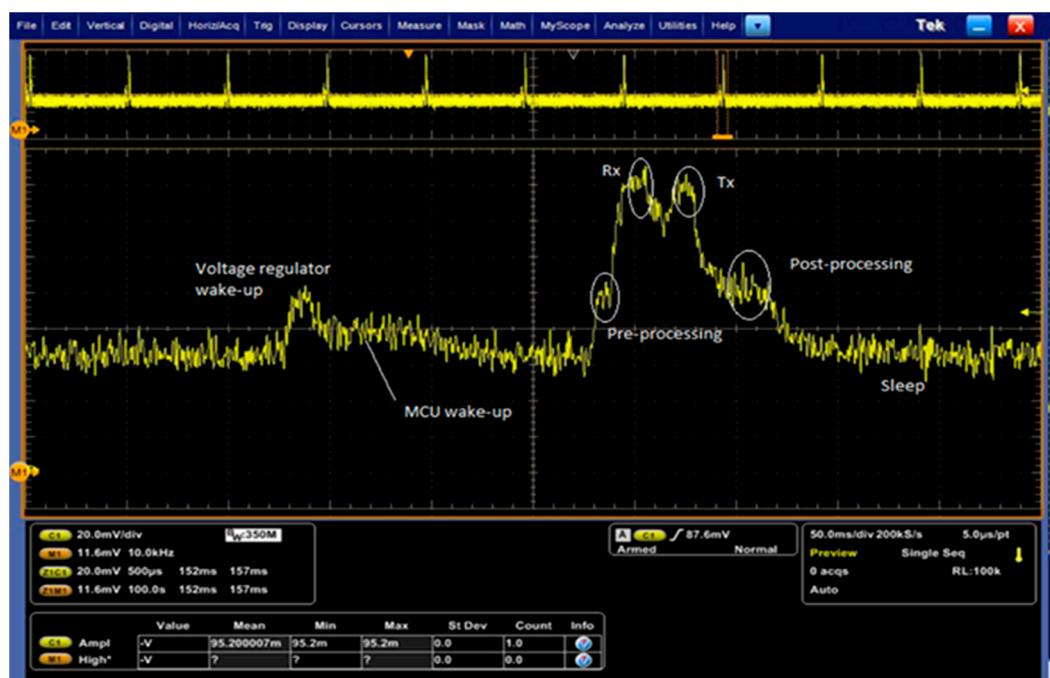
Table 1 shows the software stack version from each platform used in the experiments.

**Table 1.** Software stack version used to program the boards.

Platform	Software Stack Version
Intel A-101	Arduino 1.8.5
TI CC2540	BLE-Stack 1.4.2.2
Cypress CY8CKIT-042-BLE-A	PSoC Creator 4.2
NXP FRDM-KW41Z	MCUXpresso IDE v10.1.1

To perform the measurements, a MSO5104 Tektronix oscilloscope was used to capture the different states of an advertisement and connection events. These states, which are shown in Figure 2 and described below, are related to the operational cycle of the MCU and the radio of the devices. It is important to note that the current consumption will vary depending on each state, thus the need to measure the average current through the time duration of the advertising and connection events. The states are described as follows:

- The *Wake-up* state, where a low power MCU should remain in sleep mode most of the time and the spike shown in Figure 2, indicates the wake up of the MCU to perform tasks related to the system initialization.
- The *Pre-processing* state, where the MCU performs tasks to prepare the radio to transmit (Tx) and receive (Rx) data packets.
- The *Rx* state, where the system performs a receiving operation.
- The *Tx* state, where the system performs a transmitting operation.
- The *Post-processing* state, where the MCU processes the received packets and sets the timer for the next connection event.

**Figure 2.** Connection event states for Intel A-101.

In order to calibrate the Oscilloscope and to test the setup configuration of each platform, the TI CC2540 platform was used to replicate the results presented in Reference [4], thus being able to corroborate the measurements with the TI application note.

### 3. Results

The obtained waveforms are shown in Figures 2 and 3. As explained in Section 1, the first step for a PD is to send advertising packets in the three specified channels for such purpose: 37, 38, and 39 [3]. In this case, the PD is advertising its presence for a CD wanting to connect to it, thus the CD is scanning the advertising packets in the same channels. Once the CD finds a PD, it will send a connection request and if the PD acknowledges such a request, a connection is established and periodic connection events, set by a connection interval parameter, will be set to transmit data.



**Figure 3.** Advertising event waveform for Intel A-101.

Figure 2 shows the typical waveform of a connection event, which is basically the same for all the platforms, and Figure 3 shows the advertising event waveforms for a CD and PD. Important to note is that the waveforms presented in this section are just related to the Intel A-101 since the rest of the platforms' waveforms are similar. In the case of the Intel A-101, the current consumption data for CD and PD are not available in the datasheets of the Intel® Curie™ SoC. Similarly, the CD current consumption data obtained for the rest of the platforms (except the TI CC2540, which does not support a CD configuration) are not presented in the datasheets nor in any application note from the vendors. In the case of the PD obtained data, the results are similar to the ones presented by each vendor's application notes [4–6].

The advertising and scanning event waveforms from a PD and CD are shown in Figure 3. These waveforms can be used to estimate the current consumption given by

$$I_{on} = \frac{\sum_{i=1}^{N_s} (T_{Si} I_{Si})}{T_{on}} \quad (1)$$

where  $I_{on}$  is the average current that can be calculated while the device is awake in a connection event  $T_{on}$  and the time  $T_{Si}$  spent in each state with its corresponding current consumption  $I_{Si}$ .  $N_s$  represents the number of states and  $i$  is the index of the state [7]. The total average current  $I_{avg}$  during a connection interval  $T_{ci}$ , considering the sleep current  $I_{sleep}$ , while the system is in an idle state, can be computed by

$$I_{avg} = \frac{I_{on} T_{on} + I_{sleep} (T_{ci} - T_{on})}{T_{ci}} \quad (2)$$

Subsequently, the battery lifetime in hours is given by

$$T_{bat} = \frac{C_{bat}}{I_{avg}} \quad (3)$$

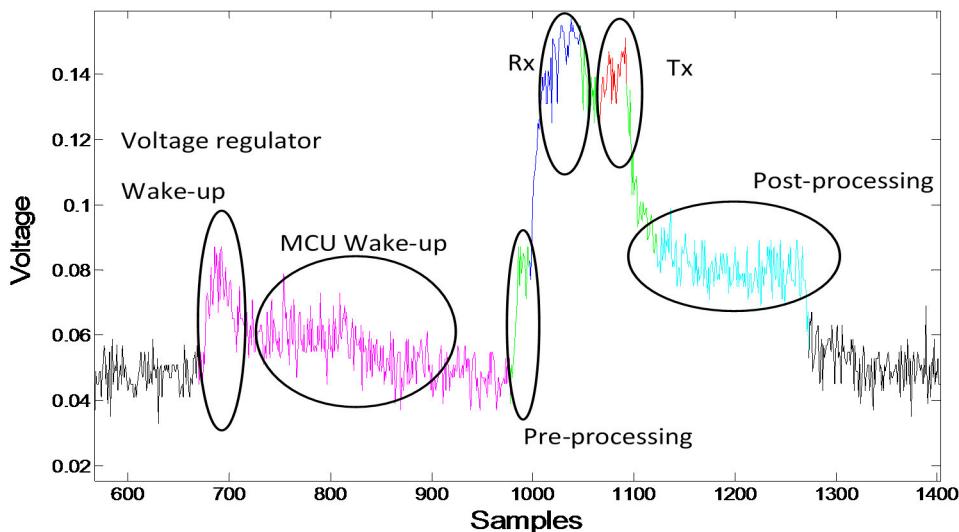
where  $C_{bat}$  is the battery capacity in mAh.

Using (2), the overall power consumption is given by

$$P = V \cdot I_{avg} \quad (4)$$

where  $V$  is the supplied voltage.

Figure 4 shows the current consumption of a connection event processed with Matlab to obtain the average current values, which are presented in Table 2, for each state and with the platforms configured as a PD.



**Figure 4.** Matlab waveform of a peripheral connection event.

**Table 2.** Current consumption comparison between the platforms configured as a PD using a connection interval of 100 ms.

State	$I_{avg}$ (mA) CC2540	$I_{avg}$ (mA) Intel A-101	$I_{avg}$ (mA) Cypress CY8CKIT-042-BLE-A	$I_{avg}$ (mA) NXP FRDM-KW41Z
Wake-up	5.69	0.774	2.11	3.73
Pre-processing	8.80	2.56	2.28	2.34
Rx	19.37	8.99	13.83	5.63
Tx	23.81	8.93	15.97	8.54
Rx-Tx	10.79	8.68	15.81	7.55
Post-processing	7.89	2.25	5.71	5.02
Total average current	0.279	0.067	0.141	0.118
Expected battery life (days)	35	145	69	82

The current consumption of a connection event for a platform configured as CD is shown in Table 3.

**Table 3.** Current consumption comparison between the platforms configured as a CD using a connection interval of 100 ms.

State	$I_{avg}$ (mA) Intel A-101	$I_{avg}$ (mA) Cypress CY8CKIT-042-BLE-A	$I_{avg}$ (mA) NXP FRDM-KW41Z
Wake-up	0.517	0.194	0.304
Pre-processing	3.95	0.573	0.704
Rx	9.02	1.48	1.97
Tx	7.97	1.8	1.48
Tx-Rx	7.37	1.39	1.77
Post-processing	2.14	0.63	1.03
Total average current	0.089	0.018	0.036
Expected battery life (days)	109	514	267

The average power consumption of a connection event is shown in Table 4.

**Table 4.** Power consumption of the platforms configured as PD and CD using a connection interval of 100 ms and a 3V power supply.

Configuration	CC2540 (mW)	Intel A-101 (mW)	Cypress CY8CKIT-042-BLE-A (mW)	NXP FRDM-KW41Z (mW)
PD	0.837	0.201	0.423	0.354
CD	N/A	0.267	0.054	0.108

Extrapolating the values for different connection intervals, an approximation of the battery life associated to a connection interval for each platform configured as a PD is shown in Table 5. The calculation of the battery life is based on a CR-2032 battery, which has a typical capacity of 235 mAh.

**Table 5.** Current consumption and expected battery life for each platform configured as PD for different connection intervals.

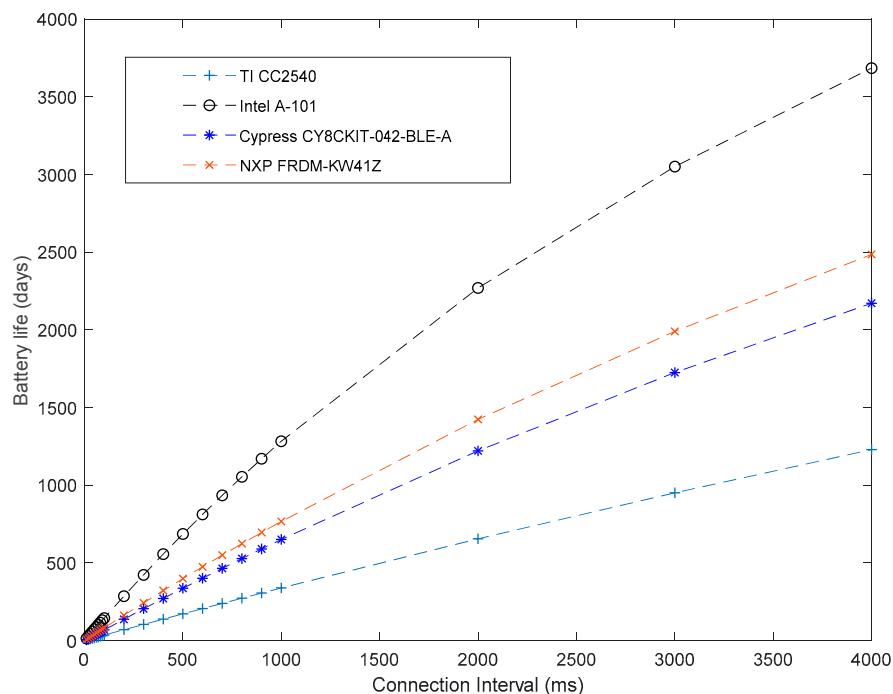
Connection Interval (ms)	$I_{avg}$ (mA)				Battery Life (in Days)			
	TI CC2540	Intel A-101	Cypress CY8CKIT-042-BLE-A	NXP FRDM-KW41Z	TI CC2540	Intel A-101	Cypress CY8CKIT-042-BLE-A	NXP FRDM-KW41Z
10	2.78	0.66	1.404	1.176	3.5	14	7	8
50	0.558	0.130	0.281	0.236	17	73	35	41
100	0.279	0.067	0.141	0.118	35	145	69	82
300	0.093	0.023	0.047	0.04	104	423	205	243
500	0.056	0.014	0.029	0.024	172	686	337	399
800	0.035	0.009	0.018	0.015	273	1054	528	624
1000	0.028	0.007	0.015	0.012	339	1283	651	767
2000	0.014	0.004	0.008	0.006	655	2269	1221	1423
3000	0.01	0.003	0.005	0.004	951	3050	1725	1990
4000	0.007	0.002	0.004	0.003	1229	3684	2171	2485

Similarly, an approximation of the battery life associated to a connection interval for each platform configured as a CD is shown in Table 6.

**Table 6.** Current consumption and expected battery life for each platform configured as CD for different connection intervals.

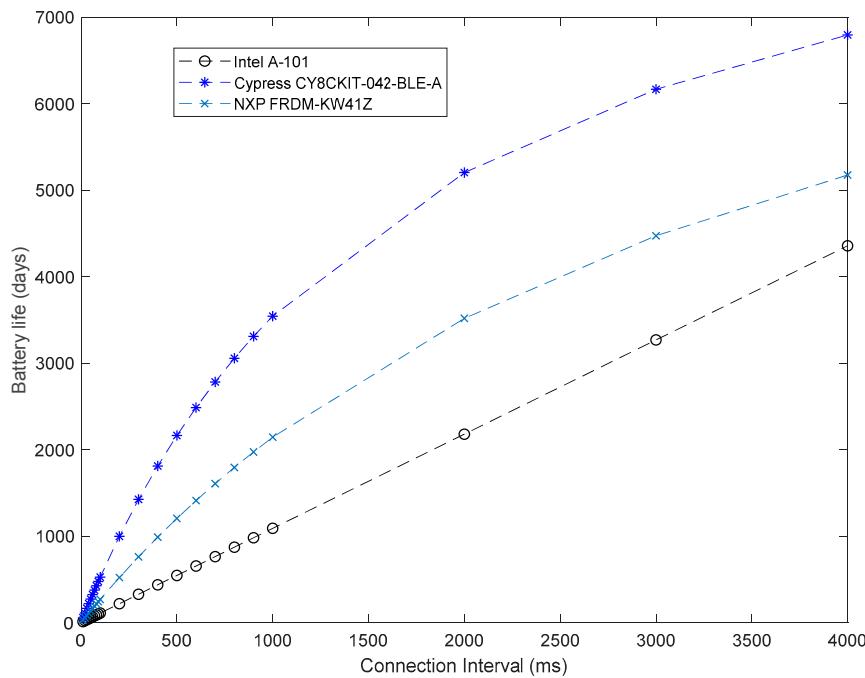
Connection Interval (ms)	$I_{avg}$ (mA)			Battery Life (in Days)		
	Intel A-101	Cypress CY8CKIT-042-BLE-A	NXP FRDM-KW41Z	Intel A-101	Cypress CY8CKIT-042-BLE-A	NXP FRDM-KW41Z
10	0.892	0.177	0.357	11	55	27
50	0.179	0.036	0.072	54	270	135
100	0.089	0.018	0.036	109	525	267
300	0.029	0.006	0.012	326	1424	760
500	0.019	0.004	0.008	544	2163	1204
800	0.011	0.003	0.005	871	3056	1793
1000	0.008	0.002	0.004	1089	3543	2144
2000	0.004	0.001	0.0027	2178	5204	3518
3000	0.002	0.0015	0.0021	3268	6167	4473
4000	0.001	0.0014	0.0018	4357	6796	5176

The connection interval versus battery life for all the platforms configured as PD, as shown in Figure 5.



**Figure 5.** Connection interval vs. battery life in PD mode.

Similarly, the connection interval versus battery life for all the platforms configured as CD is shown in Figure 6.



**Figure 6.** Connection interval vs. battery life in CD mode.

#### 4. Discussion

When comparing the obtained results for each platform, it is noted that the Intel A-101 has the lower power consumption when configured as a PD. However, Cypress CY8CKIT-042-BLE-A has a better performance when configured as a CD. In any case, different factors might affect the power consumption of the platforms. For instance, the MCU plays a significant role, since depending on the application, it might be performing different tasks such as controlling other peripherals. Moreover, the software and firmware stacks will also play a significant role, since they need to provide the means to implement the required low-power modes. As an example, Cypress and NXP platforms provide detailed application notes [5,6] and software callbacks to build a low power application by enabling deep low power modes in the platform for a better battery performance.

Another key point is that the BLE specification defines services and characteristics for a device to transfer data through the Generic Attribute Profile (GATT) [8]. In this case, a profile describes the use case and the roles that a CD and PD will have. In general, a PD will be connected only to one CD at a time meaning that once a CD discovers a PD, the device will stop advertising and will only be available to transfer data to one master device in every connection interval. Thus, it is relevant to understand the power consumption and the expected battery life for different connection interval values, which are required to be defined depending on the data transfer needed for a given application, as specified in a profile. A list of defined BLE profiles can be consulted in Reference [9].

Moreover, the relevance of the power consumption of a BLE CD is minimized since it is assumed that a smartphone, a tablet or a computer, which are typically used as CDs, can be charged more easily than a PD. While this can be true for different applications, there might be other use cases, such as in the smart farming, where this assumption is incorrect. As an example, sensors can be placed across a crop field to sense different variables such as humidity, lightning, temperature, amount of rain, among others, to detect a crop disease. Thus, the sensors (PDs) as well as the CD need to be powered by batteries that might not be easily replaced. Therefore, in order to provide a general design guideline for an optimal implementation of a BLE based application, the power consumption and expected battery life metrics for both the CD and PD, becomes relevant.

Additionally, the obtained data for a CD is not available in the datasheets and has not been reported elsewhere in the literature. Therefore, it is important to obtain these measurements to

evaluate which platform can be used for a given IoT application. There will be different tradeoffs to take into account while choosing a platform such as which MCU is implemented in the platform, the available system memory, the system peripherals, analog modules, communication interfaces, and price among others, but one key element is the software stack that allows the user to control the low power states of the device.

Equally important is to understand the implications of using a coin cell battery when designing IoT applications based on BLE. For example, the BLE protocol was designed to remain in a sleep state for the majority of the time and only to wake up to transmit data or to maintain a connection [10]. During the sleep state, the devices will typically consume few  $\mu$ A since the MCU and peripherals, including the BLE radio, are turned off. However, when a device wakes up, the current peak drawn by the BLE radio waking up and performing Rx and Tx operations, will typically consume above 10 mA by a few micro-seconds (~15 mA in Intel A-101, ~23 mA in the Cypress CY8CKIT-042-BLE-A, ~17 mA in the NXP FRDM-KW41Z and ~18 mA in TI CC2540). This current peak is known as high current pulse (HCP) [11]. In this case, the typical continuous current drain specified in a coin cell battery (~190  $\mu$ A in a CR2032) [12], will be surpassed. In other words, the battery capacity will start degrading when the load current increases and the battery voltage will start dropping, as shown in Figure 3 from Reference [11]. In such results, the battery capacity follows the typical value specified in Reference [12] when the current is 500  $\mu$ A, but when the current draw is 3 mA, the battery capacity decreases to ~150 mAh. Another key point to consider is the functional end point (FEP), defined in Reference [11], which is the minimum voltage level required by the platform circuitry to operate. Thus, an IoT application will need to take into account these factors to either select the appropriate battery or to forecast the duration of it for an application. In other words, the effective battery capacity will need to be calculated based on the requirements of the application in terms of the FEP and the duty cycle of the HCP. With this in mind, the HCP duty cycle becomes relevant when a short advertising or connection interval is set. For example, consider a 10 ms connection interval and a HPC of 30 mA with a duration of 1 ms. In this scenario, the device will be in sleep mode for 9 ms and will wake up at the 10th ms with a current consumption peak of 30 mA. When this load occurs every 10 ms, the battery capacity is heavily impacted, as can be seen in Figure 5a of Reference [11] where the battery capacity drops to 150 mAh at 2.0 V of FEP.

Important to realize is that the current peaks in the BLE will have a duration between one and 1.5 ms, considering the time it takes for the radio to wake up and the time that the Rx and Tx operations are performed. According to Reference [11], the battery life estimation can be calculated by modifying Equation (3) by

$$T_{bat} = \frac{C_{corrected}}{I_{avg}} \quad (5)$$

where  $C_{corrected}$  is the corrected battery capacity estimated by the test results performed in Reference [11]. Following such test results and their recommendation, the  $C_{corrected}$  parameter should be used when lower values of connection intervals are required, mainly from 7.5 to 25 ms since for longer time intervals, the battery will have enough time to recover. Thus, Table 7 shows the updated values for the platforms configured as a PD and with a corrected battery capacity of 190 mAh, which was obtained as the mean value from the test results and curves obtained in Reference [11].

**Table 7.** Current consumption and expected battery life for each platform configured as PD for lower connection intervals and a corrected battery capacity of 190 mAh.

Connection Interval (ms)	$I_{avg}$ (mA)				Battery Life (in Days)			
	TI CC2540	Intel A-101	Cypress CY8CKIT-042-BLE-A	NXP FRDM-KW41Z	TI CC2540	Intel A-101	Cypress CY8CKIT-042-BLE-A	NXP FRDM-KW41Z
7.5	3.43	1.12	1.87	1.56	2.3	6	4.2	5
10	2.57	0.66	1.40	1.17	3	8.4	5.6	6.7
15	1.72	0.51	0.93	0.78	4.6	12	8.5	10
20	1.29	0.35	0.70	0.58	6	16.7	11.2	13.4
25	1.06	0.23	0.56	0.47	7.6	21	14	16.8

Estimating the battery life for IoT applications depends on several other parameters apart from the peak current caused by the wake up and Rx and Tx operations, such as data packet size, MCU operations and peripherals required for a given application. In this case, the connection parameters of the BLE protocol, such as connection interval and slave latency, need to be fine-tuned to achieve the required performance when using low values of a connection interval. Additionally, when designing a low power application, the processing of information and the use of peripherals should be performed in a way that the current peak does not increase while performing the Rx and Tx operation.

## 5. Conclusions

The main contribution of this work is the measurement of current and power consumption of different commercial platforms configured as a central device. Similar measurements have been performed for the TI CC2540, the Cypress CY8CKIT-042-BLE-A, and the NXP FRDM-KW41Z. However, the data provided in such reports are only related to the PD, while this work also includes the measurements for a CD. These measurements are useful to achieve a dependable design when the CD needs to be powered by batteries that might not be easily replaced. The experimental results show that the Intel A-101 outperforms the rest of the platforms when configured as a PD, while the Cypress CY8CKIT-042-BLE-A CC2540 BLE platform shows better results when configured as a CD. In general, Cypress and NXP platforms have a better documentation, code examples and development software stacks, which are key points to implement an optimal application. In addition of the power consumption measurements, a guide for IoT designers is provided with current and battery life data as functions of several connection intervals to help designers deliver much more dependable low energy IoT system. Further steps such as measuring the power consumption with applications demanding different data throughput, are required to provide a full design guideline for implementing an optimal BLE IoT application.

**Author Contributions:** Conceptualization, E.G.-E., O.L.-G. and A.V.-G.; Methodology, E.G.-E. and O.L.-G.; Software, E.G.-E. and I.P.-L.; Experimentation, I.P.-L.; Validation, E.G.-E. and I.P.-L.; Formal Analysis, E.G.-E., O.L.-G. and A.V.-G.; Investigation, E.G.-E. and O.L.-G.; Resources, O.L.-G.; Data Curation, I.P.-L.; Writing—Original Draft Preparation, E.G.-E.; Writing—Review & Editing, O.L.-G. and A.V.-G.; Visualization, O.L.-G. and A.V.-G.; Supervision, O.L.-G. and A.V.-G.; Project Administration, O.L.-G.; Funding Acquisition, O.L.-G.

**Funding:** This research was funded by CONACYT grant number [91355-399073].

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

## References

- Swan, M. Sensor Mania! The Internet of Things, Wearable Computing, Objective Metrics, and the Quantified Self 2.0. *J. Sens. Actuator Netw.* **2012**, *1*, 217–253. [[CrossRef](#)]

2. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [[CrossRef](#)]
3. Gomez, C.; Oller, J.; Paradells, J. Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology. *Sensors* **2012**, *12*, 11734–11753. [[CrossRef](#)]
4. AN092—Measuring Bluetooth Low Energy Consumption. Available online: <http://www.ti.com/lit/an/swra347a/swra347a.pdf> (accessed on 10 December 2017).
5. AN92584—Designing for Low Power and Estimating Battery Life for BLE Applications. Available online: <http://www.cypress.com/documentation/application-notes/an92584-designing-low-power-and-estimating-battery-life-ble> (accessed on 1 May 2018).
6. NXP Semiconductors—Application Note AN5272 2016 MKW40Z Power Consumption Analysis. Available online: <https://www.nxp.com/> (accessed on 15 August 2018).
7. Afonso, J.A.; Maio, A.J.F.; Simoes, R. Performance Evaluation of Bluetooth Low Energy for High Data Rate Body Area Networks. *Wirel. Pers. Commun.* **2016**, *90*, 121–141. [[CrossRef](#)]
8. Tosi, J.; Taffoni, F.; Santacatterina, M.; Sannino, R.; Formica, D. Performance Evaluation of Bluetooth Low Energy: A Systematic Review. *Sensors* **2017**, *17*, 2898. [[CrossRef](#)] [[PubMed](#)]
9. GATT Specifications Bluetooth Technology Website. Available online: <https://www.bluetooth.com/specifications/gatt> (accessed on 2 September 2018).
10. The Bluetooth Special Interest Group. *Specification of the Bluetooth System, Covered Core Package*; Version: 5.0; The Bluetooth Special Interest Group: Kirkland, WA, USA, 2016.
11. High Pulse Drain Impact on CR2032 Coin Cell Battery Capacity. Available online: <https://m.eet.com/media/1121454/c0924post.pdf> (accessed on 28 October 2018).
12. Energizer CR2032 Product Datasheet. Available online: <http://data.energizer.com/pdfs/cr2032.pdf> (accessed on 28 October 2018).



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).