©ISTOCKPHOTO.COM/KEITH BISHOP

By Darshana Thomas, Ross McPherson,
Greig Paul, and James Irvine

BATTERIES—©ISTOCKPHOTO.COM/ST-DESIGN

The Internet of Things (IoT)—the connection of small smart sensors, actuators, and other devices to the Internet—is a key concept within the smart home. To ease deployment, such devices are often wireless and battery powered. An important issue is the wireless interface used. The ubiquity of Wi-Fi in homes today makes this an attractive option, but the relatively high power requirements of Wi-Fi conflict with the requirement for long battery life and low maintenance. Lower power alternatives, such as Bluetooth and ZigBee, have been proposed, but these have a much smaller installed base. In addition, many smart home products are currently available using 433-MHz technology.

Optimizing Power Consumption of Wi-Fi for IoT Devices

An MSP430 processor and an ESP-03 chip provide a power-efficient solution.

Digital Object Identifier 10.1109/MCE.2016.2590148
Date of publication: 22 September 2016

This article considers whether it is possible to reduce Wi-Fi power usage to the point where cheap Wi-Fi-based products can be used instead of other protocols. We undertake the power analysis of a wireless sensor with a system-on-a-chip (SOC) Wi-Fi module, with and without a separate microcontroller optimized for low-power usage, which can be used to switch the Wi-Fi module on and off. This article is an extension of previous work comparing Wi-Fi and 433-MHz devices, and it compares 433-MHz devices to the optimized Wi-Fi sensor. Finally, we consider the energy usage of Dynamic Host Configuration Protocol (DHCP), demonstrating that further energy savings can be made if the application handles Internet protocol (IP) addressing and presents a static IP address to the Wi-Fi module.

This article considers whether it is possible to reduce Wi-Fi power usage to the point where cheap Wi-Fi-based products can be used instead of other protocols.

INTRODUCTION

The IoT is a networking paradigm where small sensors, actuators, and other devices are connected to the Internet, either directly or through a hub unit. The devices then become accessible remotely and so provide flexibility for the user to control their electrical devices from anywhere with the availability of an Internet connection. Smart homes are a prime example of systems that utilize small sensor devices that relay their data to a centralized hub [10], where this information can then be redistributed to users or devices requiring data input. For example, a central heating system would benefit from temperature sensor readings.

SOC technology has allowed very complex wireless modules to be marketed at a low cost, which is encouraging its use. Figure 1 illustrates some IoT examples, and there are very many applications that can be envisaged. According to Gartner, the number of IoT devices will reach 6.4 billion this year [4].

Wi-Fi is a relatively complex wireless protocol, but recently, Wi-Fi modules have become available at low enough costs to facilitate their incorporation into IoT devices. This offers significant advantages, as many homes now have Wi-Fi hotspots, and this, combined with public-access hotspots in city centers, hotels, and transportation, means that Wi-Fi coverage is becoming ubiquitous. Using a Wi-Fi module allows the IoT device to have a direct connection to the Internet, and it can, for example, use a web service without requiring a hub. This simplifies both deployment and software development. However, the complexity of Wi-Fi means that it is far less power efficient than other more specialized wireless technologies designed for sensors. While low-power Wi-Fi is currently being standardized, the cheap Wi-Fi modules making low-cost IoT devices possible use current Wi-Fi technology.

Power consumption is a major constraint for IoT devices, since they are likely to depend on batteries. Some IoT devices can be powered from the mains, for example, a smart power socket or central heating controller. It is interesting to note that commercial examples of such devices often already incorporate Wi-Fi modules. However, mobile IoT devices, such as remote controls or smart buttons, are powered from batteries and use alternative wireless technology, such as Bluetooth low energy (BLE) or 433 MHz, and ZigBee offers another alternative as a wireless protocol specifically designed for low-power sensor applications.

The ubiquity of Wi-Fi means that it has great customer acceptance and ease of deployment—most customers simply have to switch it on within the home, rather than purchase additional hub units. Therefore, if Wi-Fi-based devices could have an acceptable battery life and reduced cost they would be of interest for consumers. However, not much research has been done on Wi-Fi-based IoT devices because Wi-Fi is generally considered to not be efficient enough in terms of power usage. For example, [16], [8], and [11] focused on the ZigBee or 433 solutions.

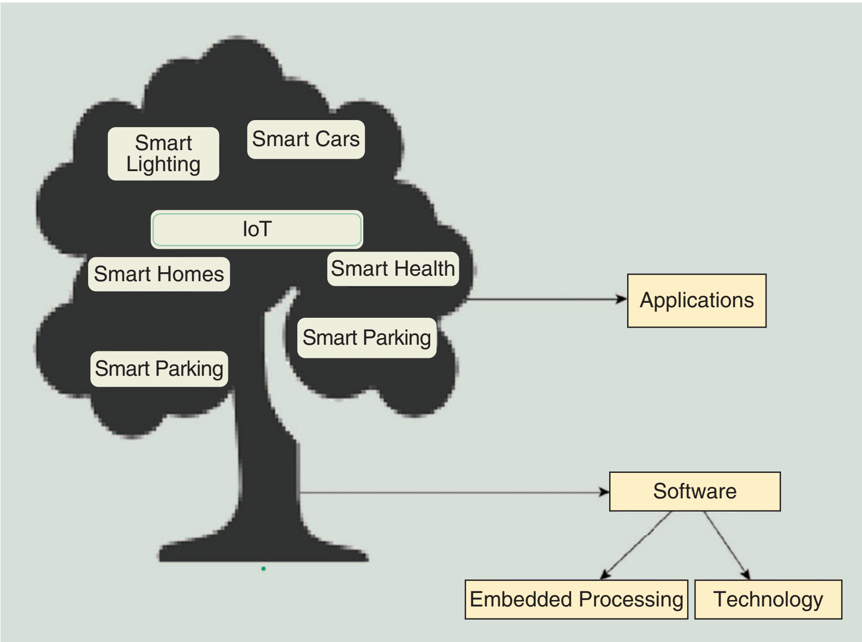

FIGURE 1. Some IoT applications.

The IoT is a networking paradigm whereby small sensors, actuators, and other devices are connected to the Internet, either directly or through a hub unit.

A comparison of the frequencies and estimated ranges is provided in Table 1. Table 2 details the capacity of each technology.

In addition to the ubiquity of Wi-Fi, it has other attractive features. In many applications, it has a longer range than alternative technologies, and with a complete Internet stack built into the module, it offers a plug-and-play option for service deployment.

This article considers the optimization of power consumption of an IoT device using a cheap commercial Wi-Fi module to see if it is practical to use such a module for a battery operated IoT device. An alternative low-power Wi-Fi device, CC3000, is also tested and compared with the low-cost ESP8266 module. The power consumption results obtained for the Wi-Fi inbuilt IoT device will show whether it is possible to replace the existing protocols with the Wi-Fi device.

WI-FI

Wi-Fi offers very high data rates—theoretically up to 600 Mb/s for the most commonly used 802.11n version controlled by Wi-Fi Alliance. A number of different versions are available with different operating frequencies and throughputs. The most widely adopted version currently deployed is 802.11n, which is compatible with early devices, albeit at lower speeds. The latest commercially available version is 802.11ac, offering higher speeds, but also the ability to support older devices. While useful for broadband access within the home, typical Wi-Fi data rates are rarely used to their full potential in sensor networks. However, the ability to support roaming and send large amounts of information in bursts is ideal for many applications. The range varies on implementation, but it can cover up to 200 m [12].

Wi-Fi offers security: both the authentication of devices and the encryption of transmitted data. Early versions of the standard were relatively insecure, but current devices implement Wi-Fi Protected Access II, which offers good security, especially given the very low amounts of data transmitted by IoT devices.

An interesting development in Wi-Fi is the Wi-Fi HaLow, or IEEE802.11ah. This is an extension to the existing Wi-Fi standard targeted at IoT applications, which uses lower frequencies, allows longer range, and has support for extended sleep cycles and other power saving features. However, it is only expected to be standardized this year, and the variations with the existing Wi-Fi mean that it will not share the advantages of compatibility with the existing deployed infrastructure. Our aim in this article is to investigate whether the currently deployed Wi-Fi is suitable for IoT devices.

Table 1. A comparison of protocols.

	Bluetooth	ZigBee	Wi-Fi
Frequencies	2.4 GHz	2.4 GHz, 868/915 MHz	2.4 GHz for 802.11 b/g/n
Range	10 m	Up to 100 m	250 m
Battery	Days	Years	Hours

RELATED WORK

There is a significant body of literature on the smart home concept and the protocols within the smart home. However, since Wi-Fi is normally discounted as requiring too much power, the focus has been on other wireless technologies.

Dongmei Yan et al. [17] discuss the implementation of ZigBee in smart home products. The approach and implementation mentioned in their work was, at the time, utilized within China's smart home industry. Later, the authors looked into power consumption of IoT devices within a smart home.

Karan Nair et al. [9] have discussed reducing power consumption for IoT wireless sensor networks (WSNs) using BLE. Their work discusses topologies used within the WSN followed by protocols such as ZigBee and Bluetooth. These authors compared the power consumption of ZigBee and NRF against BLE. Their research shows that ZigBee and NRF have high wakeup times resulting in more power consumption, whereas BLE connects faster and its wakeup time is much lower. This work does not consider real-time monitoring.

An extension to this work was considered by Artem Dementyev et al. [8], which discusses the disadvantage of using BLE in a cyclic sleep scenario. Their experiment transmitted an 8-B data packet at certain sleep intervals. The power consumption was compared against ZigBee and ANT. According to their results, ZigBee was efficient in a cyclic sleep scenario situation. The results were compared based on

Table 2. A comparison of protocols.

	Bluetooth	ZigBee	Wi-Fi
Raw Data Rate	1 Mb/s	250 Kb/s (2.4 GHz), 40 Kb/s (915 MHz), 20 Kb/s (868MHz)	11 Mb/s (2.4 GHz), 54 Mb/s (2.4 GHz or 5 GHz), 500 Mb/s (2.4 GHz and 5 GHz)

the reconnection time. Although BLE provided low-power consumption value, the time to reconnect after a cyclic sleep is much longer than ZigBee and ANT. This work has considered fixed packet sizes, and this could have an effect on the results. M.D Prieto et al. [11] looked into variable packet size. Their results look at power consumption, but other factors, such as processor, processing times, and the cost, have not been taken into consideration.

A number of surveys have been carried out on protocols such as ZigBee and BLE for WSNs, for example [13]. Kuor-Hsin Chang [7] discussed the suitability of BLE for the IoT. According to this work, BLE supports star networks and not mesh networks. Although work has been done for this aspect, it is still to be confirmed if BLE could support mesh networks. For BLE to be considered for the IoT, it has to be compatible with all types of topologies for flexibility of developers to research upon the existing work.

In their work [14], Kamlesh Sharma et al. discussed using an IoT system to reduce power consumption for devices or equipments used within a university campus. The idea is similar to the smart home concept in which ZigBee nodes would communicate to sink node and to a central server. The power consumption is reduced using this technique within the university. According to the work by Sharma et al. in which Wi-Fi enabled devices are developed, the central node could be avoided, resulting in less equipment and lower cost.

In the work by D. Thomas et al. [15], the power consumption of the aforementioned Wi-Fi chip was compared with a comparable 433-MHz amplitude modulation (AM) transmission system. This work discussed the use of the ESP chip coupled with a ATMega328P processor, which is commonly found within the popular Ardiuno Uno device [1]. This work is advanced in this article by using a much more capable processor—an MSP430 produced by Texas Instruments. The previous findings of Thomas et al. are used as a benchmark in this article.

THE EXPERIMENT

The CC3000 was one of the first highly integrated Wi-Fi modules to become available. Priced at significantly less than US$10, the CC3000 allowed Wi-Fi to be connected to small devices with very little additional design effort. The CC3000 supports 802.11g, but not 802.11n like newer devices. When launched, the manufacturer claimed that energy use per transmission was down to 3.6 μAh, which is very impressive. But this is under ideal conditions, and it does not consider the rest of the circuit, including a microcontroller, which is required.

A further reduction in costs came with the introduction of the Expressif ESP8266 chipset. These are built into a number of modules and can be obtained, with a microcontroller, for less than US$5, providing a truly integrated solution.

We considered two scenarios. In the first, the SOC capabilities of the Wi-Fi module were used to read the sensor and process the data, with no additional microcontroller. In the second scenario, a dedicated low-power microcontroller was used for data processing and controlling the Wi-Fi module, allowing it to be switched off completely to save power.

The ubiquity of Wi-Fi means that it has great customer acceptance and ease of deployment—most customers simply have to switch it on within the home, rather than purchase additional hub units.

DS18S20

The DS18S20 is a temperature sensor that operates over the one-wire interface. Each temperature sensor [2] device has its own unique code and is stored in the 64-bit, read-only memory of the device. The memory (scratchpad) of the device holds the 2 B of digital output data and retain the data by providing access to two of its access registers. The scratchpad memory consists of cyclic redundancy check (CRC) which verifies if the data received is error free. The scratchpad CRC calculates a value based on the data within the scratchpad and compares it with the read CRC. If both values match, then the data received has no error within it.

ESP-03

The Wi-Fi module used for building the prototype is the ESP-03, which is based on the ESP8266 [3]. The cost of the chip is low and has all the advantages needed to convert it into a Wi-Fi device by suitable coding pushed onto the chip. The pin layout of the chip is provided in Table 3.

Some points to be taken into consideration for this chip are that a few of the pins within the chip have to be high

Table 3. The chip pin connections.

ESP-03 PIN	Connections
CHPD	High
URXD	Receiving pin for programming
UTXD	Transmitting pin for programming
GND	Gnd
GPIO0	Low for programming
GPIO15	High
GPIO16	Input-output (I/O)
GPIO2	I/O
GPIO14	I/O
GPIO13	I/O
GPIO12	I/O

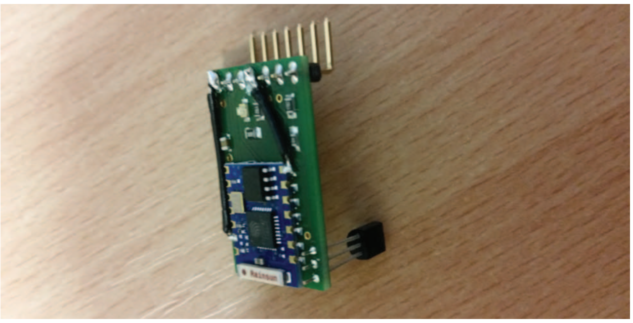

FIGURE 2. The ESP-03 prototype.

> Wi-Fi offers security: both the authentication of devices and the encryption of transmitted data.

and that a few of the pins have to be grounded all the time. The connection of this chip for the prototype is provided in Table 3.

FIRST SCENARIO

In the first scenario, a prototype (Figure 2) is built with ESP-03 and the DS18S20 [2] sensor. For the ESP-03 scenario, the temperature sensor is connected to a general-purpose input/output (GPIO) pin on the chip. The chip conducts the entire process of reading the temperature data, connecting, and transmitting. The pinouts for the prototype board are only used for programming. The temperature sensor transmits a fixed 7 B of data. The coding for the ESP is setup such that it looks for the sensor connected to the chip and takes the reading in degrees Celsius. The temperature sensor is turned to *low* within the code after taking the reading to reduce power consumption. Each time the code runs, the sensor goes to *high* to take the reading and then *low* during conversion. After the reading is obtained, the ESP connects to the server via a router and transmit the obtained temperature

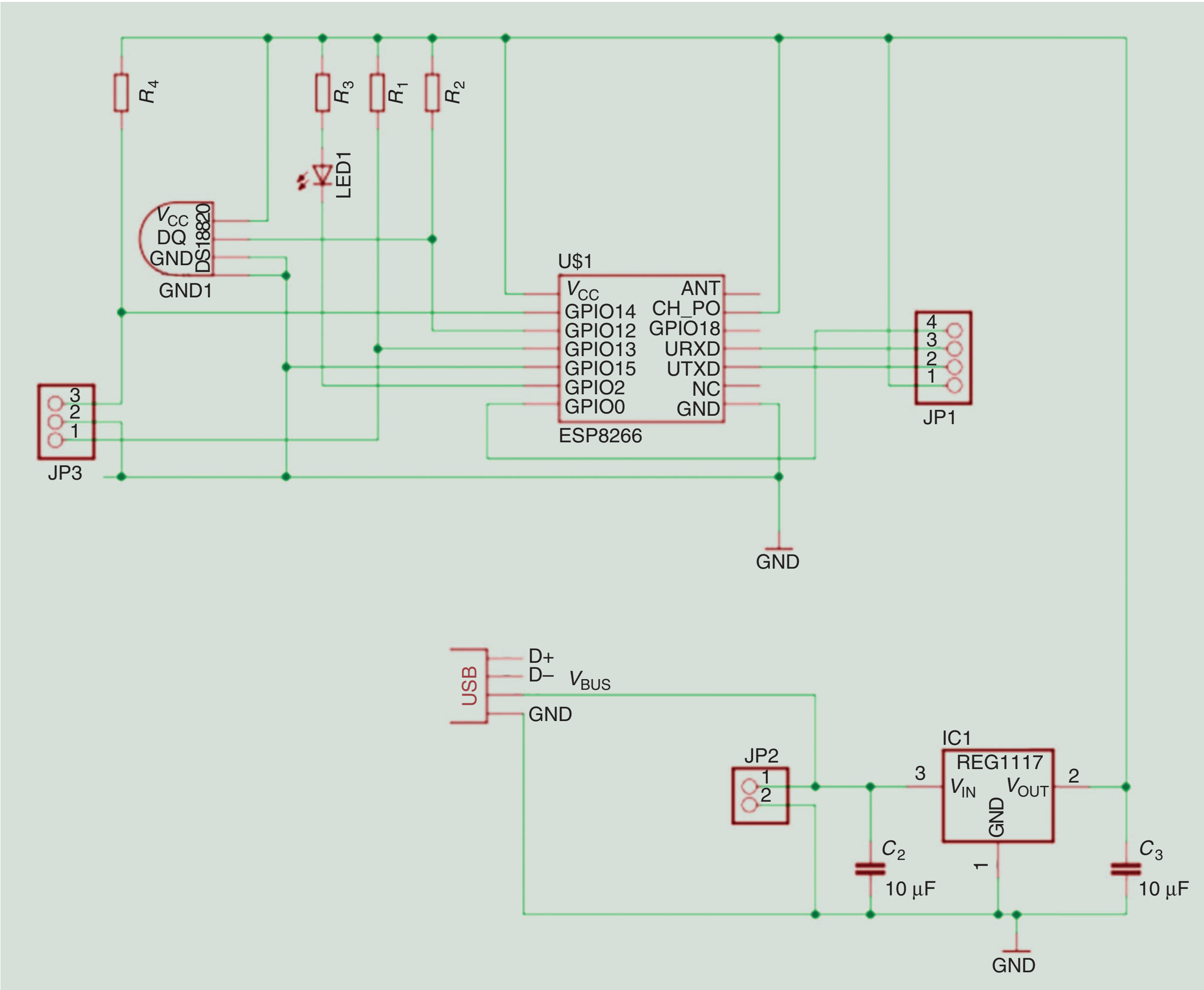

FIGURE 3. The ESP-03 prototype schematics.

In the first scenario, a prototype is built with ESP-03 and the DS18S20 sensor.

data. A Hypertext Transfer Protocol (HTTP) server was setup for recording data. As power consumption value has to be recorded, a power measuring device, Portapow [6], was used. The power reading was obtained in milliwatt hours. Power is measured for one transmission going up for different transmission times, in seconds. The schematics of this prototype are provided in Figure 3.

SECOND SCENARIO

The second prototype (Figure 4) is built with an MSP430 processor, an ESP-03 chip, and a temperature sensor (DS18S20) for temperature reading. In Figure 4, the difference for this prototype is the addition of an MSP430 [5] chip from the previous prototype; therefore, the figure illustrates the back of the prototype where the MSP chip is added. The top part of the prototype is the same as Figure 2. More pinouts are provided for this prototype (Figure 4) to add any additional components externally to this device if necessary for test purposes. The schematics of this prototype are provided in Figure 5.

In this prototype, the MSP430 reads the data and uses the ESP only for transmission. The data is transmitted with a 10-min delay. The outcome for this scenario is expected to show less power consumption than the first scenario setup (Table 4).

FIGURE 4. The MSP430 + ESP-03 prototype with lithium-thionyl chloride.

Table 4. The current measurements.

Deep-sleep ESP	15 uA
Temperature sensor current	15 mA
ESP + MSP deep sleep	10 uA

DISCUSSION OF THE RESULTS

Table 5 shows the energy usage per transmission for the CC3000, the ESP-03 alone (scenario 1), and the ESP-03 and MSP430 combination (scenario 2). It can be seen that in all

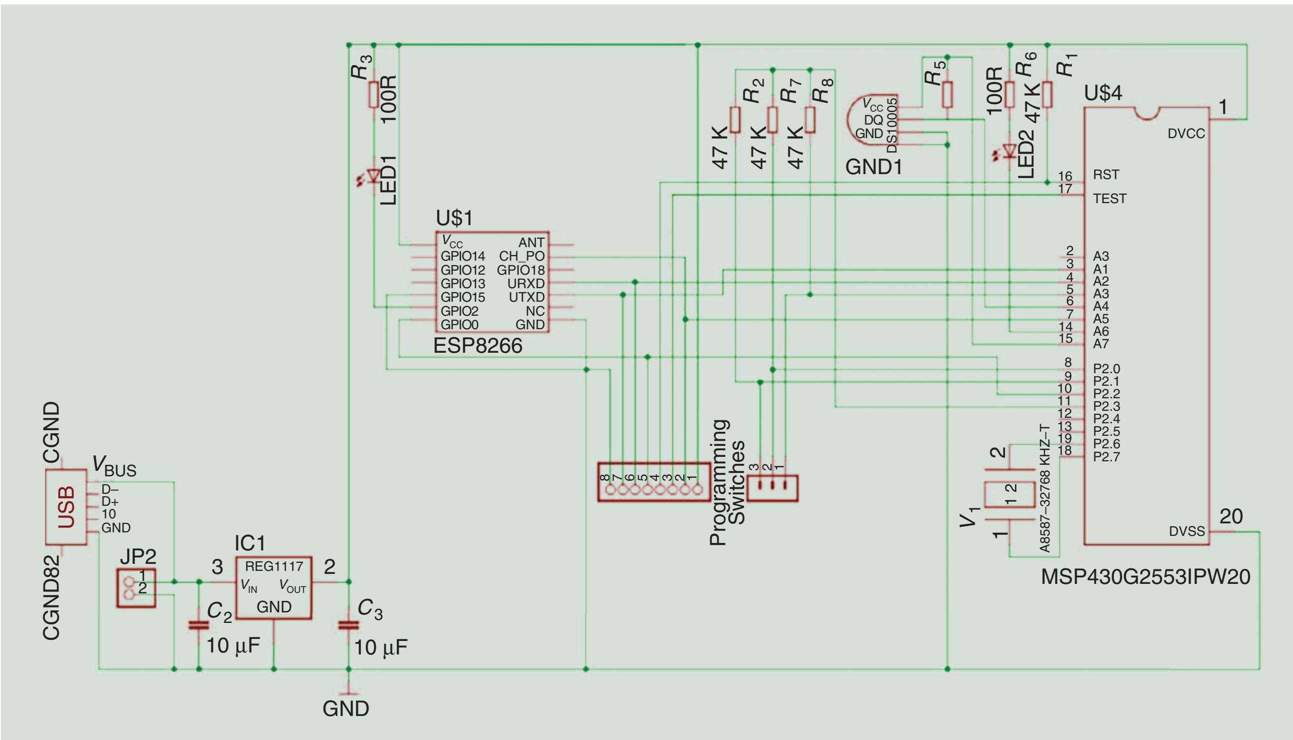

FIGURE 5. The ESP-03+ MSP430 prototype schematics.

> The second prototype is built with an MSP430 processor, an ESP-03 chip, and a temperature sensor (DS18S20) for temperature reading.

cases, as the transmission interval increases, so does the energy use: this is due to the energy consumed during the sleep cycle between transmissions. However, the energy use of the transmission itself dominates. The CC3000 has the poorest performance in terms of energy use, and it is significantly higher than the manufacturer claims. This is due to the fact that the energy use of the complete system is considered, and the CC3000 requires quite a significant microcontroller. The ESP-03, which is completely integrated, has a lower energy use. For long transmission intervals, the ESP-03 and MSP430 combination has the lowest energy use.

Table 5. The energy use per transmission.

Transmission Every	CC3000 (MWh)	ESP (MWh)	ESP+MSP (MWh)
10 min	1.268	0.285	0.331
30 min	1.277	0.297	0.333
1 h	1.289	0.315	0.337
2 h	1.316	0.350	0.344
3 h	1.343	0.421	0.359
4 h	1.370	0.563	0.389
1 day	2.039	1.131	0.508

For long-term fit-and-forget applications, lithium-thionyl chloride provides an excellent option with high energy density and low self-leakage. With 3.6 V and an ability to provide peak currents of tens of milliamps, no regulator is required for the ESP-03, which saves on energy use. Available in a range of capacities, the AA-size battery provides between 2,200 mAh and 2,500 mAh depending on the manufacturer. Allowing for leakage, such a battery would last for three years when transmitting every 4 h or for over ten years if averaging one transmission per day.

Figure 6 illustrates the power savings obtained by using MSP430 and ESP-03 together in more detail. If the device has to run for longer periods of time, the second scenario with the MSP430 is the better option. Using the ESP-03 alone uses more energy, as its power consumption in sleep mode exceeds that of the MSP430. If data is being sent frequently and the sleep period is short, the additional energy used by the MSP430 exceeds the savings obtained by switching the ESP-03 off; so, using the ESP-03 on its own would save power. Comparing the results, the ESP03 with the MSP (Figure 6) and the ESP with the ATmega processor (Figure 7), the MSP430 processor is better in terms of power savings. However, the majority of IoT devices installed in homes would have a relatively long period of time between transmissions.

In [15], it was discovered that with a low delay between transmissions the 433-MHz system used less energy than the ESP device. But for long periods of delay, the energy consumed by the ATMega328P processor during sleep mode was still high. This article extends this work and compares the results of using a MSP430 processor instead.

The results from [15] (Figure 7) have been compared with the results obtained for MSP430 with 433 MHz (Figure 8). Although MSP430 + ESP03 uses significantly more energy for an equal number of transmissions over a given period, it is worth noting that due to physical limitations of operating at a

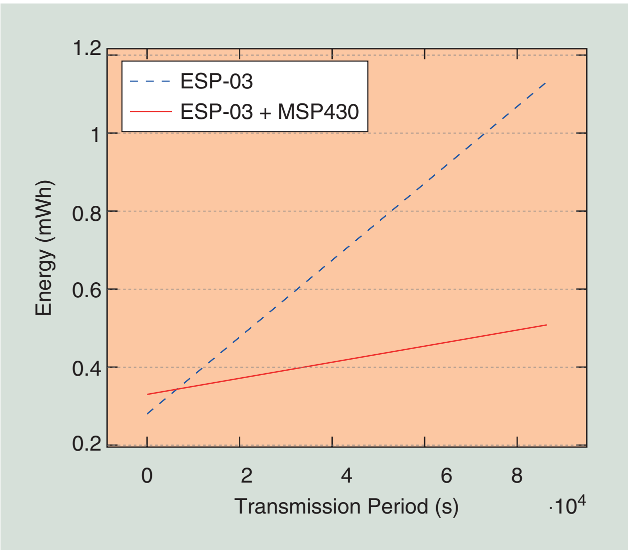

FIGURE 6. A comparison of ESP-03 versus MSP430.

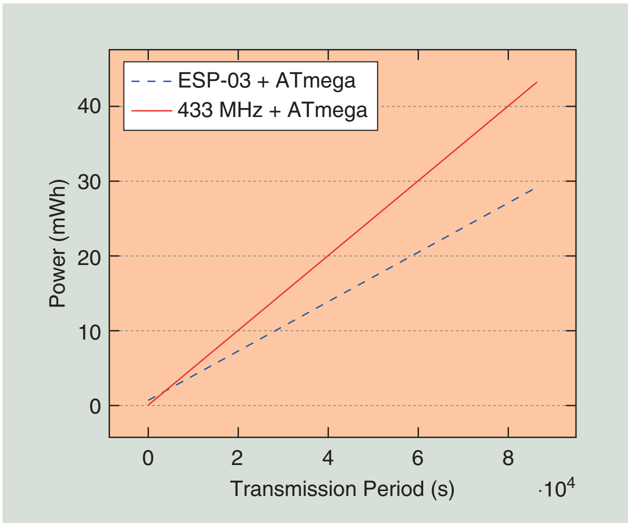

FIGURE 7. A comparison of Wi-Fi versus 433 MHz when using an ATmega chip [15].

higher frequency (2.4 GHz versus 433 MHz), coupled with the fact that to transmit data, the ESP-03 has to create the whole ISO stack (while the AM system can transmit on the equivalent to the physical layer), the energy requirements for the Wi-Fi system is always going to be higher. But the Wi-Fi system has significantly higher data rates; so, if a system required a lot of bandwidth to be transmitted, it would be more efficient to use the Wi-Fi system. Furthermore, since authentication is required when connecting to a secured wireless network, this means that only sensors that have been authorized to access and transmit on the network can provide information, and this provides more security. Whereas within a basic AM system, any received signal will be interrupted allowing the possibility of false signals.

IP ADDRESS ALLOCATION

Home networks and Wi-Fi networks often use dynamic IP address allocation for ease of management. IP addresses are allocated using DHCP. In order scenario, since the ESP-03 is switched off to save power between transmissions, an address has to be allocated whenever the device is switched on, i.e., for each transmission. Since the transmissions are short, the DHCP exchange, although short, has an impact on battery life.

DHCP is a very flexible protocol and could be used to send measurements without an overlaid Internet service, but the DHCP server would have to be configured to allow this. Since we want to use standard Internet protocols and reduce deployment issues, we would not want to use DHCP in this manner. However, in a home network, IP addresses are normally allocated for an extended period of time (for example, 24 h). For the MSP430-based device, the MSP430 can be used to store the allocated IP address between transmissions, so when the ESP-03 is switched on, it can use a static address from the MSP430. We have investigated the power consumption with static IP to investigate the reduction in power consumption through the saved DHCP exchange.

A graph comparing static and DHCP IP address allocation on both the standalone ESP-03 and the ESP-03 + MSP430 is shown in Figure 9. As shown in the graph, there is roughly 30% reduction in energy consumption when using static IP address allocation compared to DHCP on each transmission. This experiment showed that it was possible to further reduce the energy consumption of the device by using long DHCP leases but at some reduction in adaptability; so, it was decided to continue using DHCP for the remainder of the experiments.

The scenario that we have considered is for unidirectional communications from the sensor node to the Internet, which fits well with a sensor gathering readings or a smart button triggering an alert.

SENSOR ACTIVATION

The sensor used for this prototype had a standby current resulting in power used up by both the sensor and ESP during sleep mode. For the temperature sensor with the shutdown feature, the standby current of 750 nA, for the DS18s20, could be saved. In the long run, this would result in more power savings. Replacing the sensor with a no-standby current would be an option for the future to reduce power usage. The temperature sensor would be *high* while taking the reading and then turned to *low* within the code before it goes into sleep mode. This would reduce the power, and only the standby current would be used up.

BIDIRECTIONAL COMMUNICATIONS

The scenario that we have considered is for unidirectional communications from the sensor node to the Internet, which fits well with a sensor gathering readings or a smart button triggering

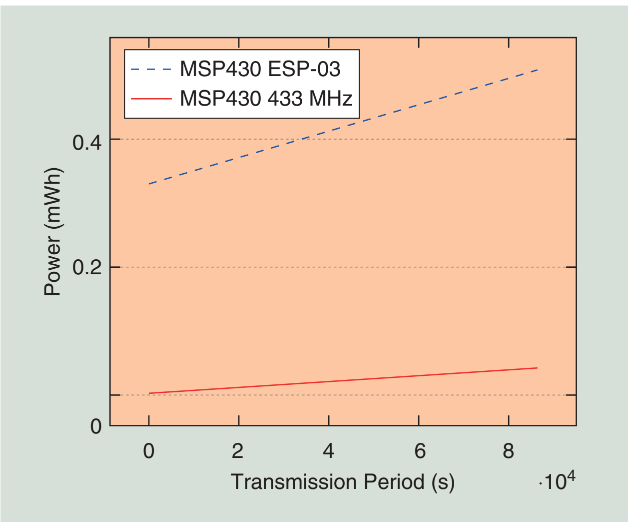

FIGURE 8. A comparison of Wi-Fi versus 433 MHz using MSP430.

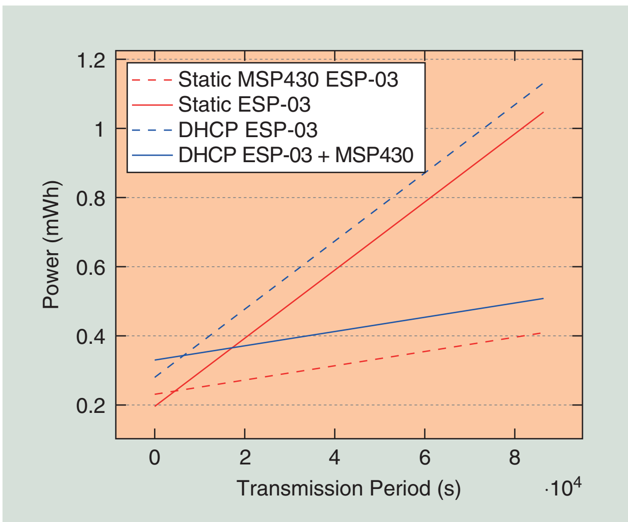

FIGURE 9. A comparison of static IP assignment versus DHCP.

When combined with a low-power processor such as the MSP430, the ESP-03 is power efficient for use in an IoT device, so Wi-Fi-based battery operated IoT devices are practical.

an alert. Bidirectional communications have significant implications on energy use because the reception circuitry has to be left on in listening mode. For Wi-Fi, bidirectional communication requires the device to be attached to the access point, which requires it to be active to receive beacon frames. The ESP8266 has a light sleep mode that has a timer to switch the central processing unit and radio circuitry off between beacons to save power, waking the chip up before the next beacon. However, while this offers significant reductions over keeping the chip active, the overall power usage remains in the 0.5–1-mA range, which is clearly far too high for long-term battery operation. Wi-Fi, therefore, does not offer a solution where asynchronous bidirectional communication is required.

If delays can be tolerated, for example, for updating configuration values, data can be sent to the sensor as part of the acknowledgment when the sensor sends data to the server, or the sensor can periodically poll the server even if there is no data to be sent. Our measurements already allow for an acknowledgment as part of the HTTP exchange—adding some additional bytes to send data would have very little effect on the energy usage.

CONCLUSIONS

Based on the results from this article, when combined with a low-power processor such as the MSP430, the ESP-03 is power efficient for use in an IoT device, so Wi-Fi-based battery operated IoT devices are practical. The results illustrated that using ESP-03 on its own is only efficient if it is to be used for a short period of time. If the ESP-03 was coupled with MSP430, then the power usage is reduced, and it is much more efficient when used for longer period of time. The power consumption results were then carried out using DHCP and a static IP to demonstrate the power saving by using a static IP instead if DHCP, which is used majority of the time to reduce complexity. The use of the processor MSP430 showed an increased power saving compared to the ATMEGA processor.

Using a lower voltage 433-MHz AM transmitter used less power than MSP430 + ESP03, but taking into account the low transmission range and security aspects, the MSP430 + ESP03 is a more flexible choice. In terms of range and security, Wi-Fi is better, and therefore it is much more effective for IoT devices.

ACKNOWLEDGMENTS

This work is supported by Treegreen and Strathclyde University in Glasgow, United Kingdom.

ABOUT THE AUTHORS

Darshana Thomas (darshana.thomas@strath.ac.uk) is a Ph.D. researcher in the Mobile Communications Group at the University of Strathclyde, Glasgow, United Kingdom.

Ross McPherson (ekb12174@uni.strath.ac.uk) is an undergraduate student undertaking the computer and electronic systems degree at the University of Strathclyde, Glasgow, United Kingdom.

Greig Paul (greig.paul@strath.ac.uk) is a Ph.D. researcher in the Mobile Communications Group at the University of Strathclyde, Glasgow, United Kingdom.

James Irvine (j.m.irvine@strath.ac.uk) is head of the Mobile Communications Group at the University of Strathclyde, Glasgow, United Kingdom.

REFERENCES

[1] Arduino. (2016). Arduino Uno & Genuino Uno. [Online]. Available: https://www.arduino.cc/en/main/arduinoBoardUno

[2] Maximum Integrated Products, Inc. (2015). DS18S20: High-Precision 1-Wire Digital Thermometer. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/DS18S20.pdf

[3] Espressif, Inc. (2016). ESP8266EX. [Online]. Available: http://espressif.com/en/products/hardware/esp8266ex/overview

[4] Gartner, Inc. (2015, Nov.). Gartner Says 6.4 Billion Connected "Things" Will Be in Use in 2016, Up 30 Percent From 2015. [Online]. Available: http://www.gartner.com/newsroom/id/3165317

[5] Texas Instruments. (2016). Mixed Signal Microcontroller. [Online]. Available: http://www.ti.com/lit/ds/symlink/msp430g2553.pdf

[6] Portapow. (2016). PortaPow Premium USB + DC Power Monitor. [Online]. Available: http://www.portablepowersupplies.co.uk/portapow-premium-usb-dc-power-monitor/

[7] K. H. Chang, "Bluetooth: a viable solution for IoT? [industry perspectives]," *IEEE Wireless Commun.*, vol. 21, no. 6, pp. 6–7, Dec. 2014.

[8] A. Dementyev, S. Hodges, S. Taylor, and J. Smith, "Power consumption analysis of Bluetooth Low Energy, ZigBee, and ANT sensor nodes in a cyclic sleep scenario," in *2013 IEEE Int. Wireless Symp.*, pp. 1–4.

[9] K. Nair, J. Kulkarni, M. Warde, Z. Dave, V. Rawalgaonkar, G. Gore, and J. Joshi, "Optimizing power consumption in IoT based wireless sensor networks using Bluetooth low energy," in *2015 Int. Conf. Green Computing and Internet of Things*, pp. 589–593.

[10] C. Perera, P. Jayaraman, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Dynamic configuration of sensors using mobile sensor hub in internet of things paradigm," in *2013 IEEE 8th Int. Conf. Intelligent Sensors, Sensor Networks, and Inform. Processing*, pp. 473–478.

[11] M. D. Prieto, B. Martinez, M. Montn, I. V. Guillen, X. V. Guillen, and J. A. Moreno, "Balancing power consumption in IoT devices by using variable packet size," in *2014 8th Int. Conf. Complex, Intelligent, and Software Intensive Syst.*, pp. 170–176.

[12] G. Reiter. (2014). Wireless connectivity for the Internet of Things. [Online]. Available: http://www.ti.com/lit/wp/swry010/swry010.pdf

[13] A. Sharma and G. S. Tewolde, "Considerations in low power wireless sensor networks," in *2015 IEEE Int. Conf. on Electro/Inform. Technol.*, pp. 626–631.

[14] K. Sharma and T. Suryakanthi, "Smart system: IoT for university," in *2015 Int. Conf. Green Computing and Internet of Things*, pp. 1586–1593.

[15] D. Thomas, R. McPherson, and J. Irvine, "Power analysis of local transmission technologies," presented at the 12th Conf. Ph.D. Res. Microelectron. and Electron., 2016.

[16] D. Yan and Z. Dan, "ZigBee-based smart home system design," in *2010 3rd Int. Conf. Advanced Comput. Theory and Eng.*, vol. 2, pp. 650–653.