Proposal and Assessment of Algorithms for Power Consumption Reduction in Wireless Networks with E-Ink Displays

Diego Sánchez-de-Rivera[1], Ramón Alcarria[2], Diego Martín[1], Gabriel Huecas[1]

[1] Department of Telematics Systems Engineering, Technical University of Madrid, Spain
[2] Department of Topographic Engineering and Cartography, Technical University of Madrid, Spain
{diegosanchez, ralcarria, dmartin, gabriel}@dit.upm.es

Abstract

Heterogeneous battery powered devices in the new Internet of Things (IoT) paradigm enable low computer resource elements to wirelessly and seamlessly communicate with other components. However, battery performance is currently a key issue, influenced by multiple factors such as data transmission time and use of computational resources. There are some battery-saving solutions optimizing sleep periods and compressing transmitted data to reduce the power impact, but they are not optimized for state of the art display technologies such as E-Ink. This paper proposes algorithms involving time synchronization and compression techniques, adapted from well-known classic RLE and Huffman algorithms, and evaluated their performance in a scenario in which small E-Ink display devices are used to implement a low-power wireless transmission. A study of the selected algorithms leads to personalize different parameters for an optimal adaptation and the life-time of the battery is tested in order to detect battery improvements over the selected algorithms.

Keywords: Low-energy, Battery optimization, Wake-up algorithms, Electronic paper

1 Introduction

Low-energy devices are growing in the new paradigm of the wireless self-aware hardware systems. In the actual era of electronics, one of the main goals is to maximize the interoperability between hardware components and achieve a reasonable performance in the case of wireless devices. We can distinguish two types of hardware devices related to the importance of them in a certain architecture. A first type of devices, performing main functions with complex hardware elements, need to be always on and plugged into a continuous power source in order to maintain the control of a system. These devices frequently work with wireless communications to control and manage other smaller devices with more limited resources and often only powered by a limited source of energy, such a battery. The available energy in these devices is a key element that it must be controlled exhaustively to minimize unnecessary current leaks and increase the run-time.

Focusing on low-energy transmission, the low resource devices try to exploit the communication in order to drain as little energy as possible. Low-energy communications depend almost entirely on the power and the duration of the transmission, and taking them into account, the consumption of the energy can be optimized in both ways. Transmission power is responsible for assuring the discoverability of the device in their range of operation and this is why the wireless devices need a sleep state to reduce and often disconnect the wireless interface. Otherwise, the amount of time that a transmission is being performed is easily connected to a software aspect, because it is directly related to the quantity of the data being transmitted through the link. Both of them are the scope of this analysis.

Keeping a wireless device discoverable all the time can be avoided to increase the running time on battery with no power source connected. To achieve that goal, this paper proposes the addition of different wake-up algorithms to adapt the sleep routine in order to reduce the time in which the device is discoverable but trying to carry on the job without crucial delays.

There are several software techniques to reduce the information data that is needed to carry a message. The compression is a method that takes a non-optimized data flow and reduces the size of it by increasing the complexity of the stream with a codification. Then the information is sent and received in the destination device, where in order to recover the original data, it has to be decoded.

The main objective of this paper is to perform a comparison of different wake-up algorithms against diverse types of sources and a power impact analysis for each compression algorithm presented in this work, including the option of the raw deliver to figure out which algorithm allows a more optimal use of the battery. An analysis of delay times in data transmission

---

[*] Corresponding Author: Diego Sánchez-de-Rivera; E-mail: diegosanchez@dit.upm.es

DOI: 10.3966/160792642018121907007

to the devices and a discussion discerning and deciding between sending large data over the wireless communication or sending a smaller quantity and use more computational effort in the data decompression, will be the final conclusion that we are aiming for this research project.

The structure of the paper is as follows: Section 2 describes related work in power consumption reduction and Section 3 describes the labeling architecture for retail. Section 4 proposes timing methodologies whereas Section 5 presents the compression algorithms compared in this paper. Section 6 describes the experimental scenarios and Section 7 shows the results. Finally, Section 8 describes some conclusions and future works.

2 Backgrounds

Low-power systems have been relevant in the last years due to the increased interest in energy aware devices. IoT related topics focus on how to propagate connectivity into devices with great mobility possibilities. Often, a reduced consumption is needed in order to provide longer autonomy and avoid maintenance procedures. To minimize operating power and maximizing the total battery life-time in a scenario where the wireless transmission is a key factor, low energy protocols, as well as procedures to reduce the transmitted data, have been considered to perform the final implementation in this paper.

Several wireless communication technologies have been developed to meet the recent IoT paradigm. Bluetooth, ZigBee and the newer version of Bluetooth defined as Bluetooth LE (low energy) are the most famous wireless standards nowadays. In [1] a comparison is made in terms of applications, power consumption, data rate and data encryption. Also, the advantages and disadvantages of each implementation are identified.

Wireless protocols like the new version of the Bluetooth technology are now bringing energy consumption reduction to a greater extent. Combining the energy aware wireless protocols with compression algorithms in the source has been analyzed in [3], obtaining relevant results in reducing the time involved in the final transmission. However, this study lacks the impact analysis of the decoding process in the destination device. Optimizing connection between master and slave devices is proposed in [5] with an infrastructure using a hybrid topology to reduce power consumption in wireless sensor network. This allows the device to remain more time in sleep mode in which the power drain is reduced.

Data compression algorithms such as [2] applies in wireless sensor networks to effectively reduce the quantity of the information transmitted through the air. Both lossy and lossless scenario approaches, exploiting the temporal correlation of sensor data can be found in [4, 15]. One problem found in literature emerges when the device is not a sensor, in charge of generating the information, but an actuator, in charge of representing the information received, and therefore the device has to decode and present it. Soft computing algorithms such as [7-8] try to reduce the computing load in both, master and slave devices.

Algorithm studies involving data compression in wireless devices often focus on video transmission, a review [6] of the state of the art in energy efficient techniques are presented and discussed in resource constrained systems and it concludes that there is a two-group classification, energy-aware transmission systems and energy-aware [14] compression systems, which we are trying to unify in a lossless scenario. Electronic paper (E-Paper) [16] intends to display information with minimal energy consumption, as it can be disconnected from the power source without losing the displayed information, low energy communications and computational energy aware methods [17] are key to an effective power saving strategies.

Wireless sensor networks relay on improving battery life of sensors by minimizing the transmission of information [13, 15] through the wireless channel. Optimizing the compression for efficient power allocation in multi-dimensional sensor clusters [9] is analyzed and a mathematical model is developed to achieve a balanced between the transmission power and the compression power consumption.

3 Architecture Definition

The system developed for this comparison experiment has been developed in the research group laboratory, it is described in [10] and presented in Figure 1. It is composed of different modules including a portable, low-powered device in which the measurements take place. In this paper, we will control how data size and wireless time utilization affects the battery life-time duration in our device. Our scenario includes a battery powered display device using an electronic paper display for representing black and white images. The display device is connected by a wireless connection for a reliable image transmission and used to perform the testing and validation of our proposal. This will allow comparing between which methods improves the battery life-time.

A computer with a compatible wireless technology is used to connect the display devices when a connection is ready to be done. The computer is configured to allow changing the transmission algorithm and the display device will recognize the proper algorithm to use by processing a small header at the beginning of each transmission.

Our scenario has been designed to perform dynamic screen updates in a low-power environment, where sending image data through a wireless channel is the

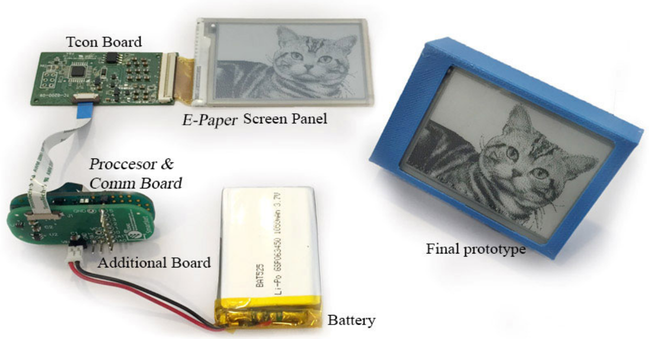

Figure 1. Evaluation prototype

most power consuming process. In previous work, the research group has proven that low power elements can help the realization of low-energy devices that do not need expensive maintenance and integrate wireless communications to facilitate the deployment. We now use the previously developed infrastructure to reveal whether or not, optimizing the communications, allow increasing the battery life-time in low resource devices.

After performing several tests with different wireless technologies, choosing Bluetooth LE as the main technology has been determined by the newest improvements in power consumption management offered by the low energy version of the protocol. It simplifies the sleeping process of the device and provides enough throughput to a successful data transmission. This includes the ability to modify the connection interval and lease time to provide consistent results through all test cases and allow the replication of the experiment with other devices.

Other options considered were the ZigBee and Wi-Fi technologies. ZigBee is based on the 802.15.4 standard and designed for a low transmission performance at low data rate and Wi-Fi is based on the popular 802.11 standard, which is designed for transmissions with high energy usage and high transfer rate. Bluetooth LE is built on top of the 802.15.1 Bluetooth layer as this latest version incorporates specific protocols to control the use of energy and adapt it in a more optimal form to control the transmission rate and consumption required in a concrete context. In our experiment, a low-power e-paper based display will be used as the main actuating element, and the content of the display transmitted by Bluetooth LE.

The prototype device is based in a crowdsourcing project [21] that developed an Arduino [20] compatible board with LE Bluetooth that permits building applications using the Arduino development platform. Using the same board schema used in official board allows us to program the prototype as if it was a simple Arduino sketch with Bluetooth wireless functions. The LM311 Bluetooth chip is connected to the main processor through a specific protocol that allows us access to the serial port of the Atmel processor using predefined functions. In addition, we have direct access to the communication buses in the main processor; this allows connecting elements that allow us to evaluate the sending and reception capabilities of the device, such as connecting the e-paper display to the board. As explained in [10], the main device is complemented with an additional PCB in order to provide additional functions and capabilities to interact with the e-paper display. The hardware used in the experiment has several limitations in terms of processing capacity.

Specifically, our test unit has the following specifications: ATMEGA328P microcontroller running at 8MHz a 32KB Flash memory and a limiting 2KB of SRAM. This limits us the dynamic capability of receiving and sending information due the limited space to work with received or sent data. Working with display screens, the size of the panel is a key element to consider when we have to select the memory and the processor components.

The display screen has a dimension of 264x176 pixels with a unique value between black or white for each pixel, thus requiring only one bit per pixel to represent the screen. This leads us to an array of 46464 pixels, which are the required 5808 bytes to store the total panel in memory.

The display used in the experiment is developed by Pervasive Displays, and offers a monochromatic pixel screen of 264x176 pixels using the electronic ink technology. The screen is driven by a timing interface board in charge of the final procedures of power up, power down and the refreshing process. The timing interface board offers an SPI interface to our main processor and requires a 3.3V power source. To satisfy all the connections between the prototype board and the display module, an additional display has been developed, implementing a specific SPI port and a configurable current switch to allow the complete disconnection of the screen in case of energy saving scenarios.

In the practical aspect, the resources of the hardware are limited even more once all libraries needed for the wireless functions are loaded. A blank sketch ended up with the available memory shown in Table 1.

Table 1. Memory availability

	Total size (Bytes)	Available space (Bytes)
Flash memory	3200	20295
SRAM memory	2000	1250

This memory availability discards all block compressors such as JPEG or LZ, because in these algorithms an additional buffer to decompress the received block tiles is needed, and makes impossible the generation of the decompressed image file in the SRAM memory. All algorithms must be compatible with sequential de-codification to be able to send chunks of compressed data to the device and process it before the next data slice reaches it. In this manner, SRAM memory is never full and the program runs without errors.

Another aspect of the data transmission requirements is the order in which the information has to be processed and sent to the display. The timing controller is very restrictive in this aspect because the screen only performs a complete refresh of the screen by passing the whole array of pixels. This implies the demand of all pixel information every time we want the display to change. The pixel data have to be sent in line packets, one line at a time and in rigorous order. This also eliminates the possibility of block processing, as it would be impossible to generate the following lines without decompressing all the image data. Leastwise, timing controller allows waiting for an unlimited period of time the following line packet thus it is possible to receive one section at a time.

The previous limitation leaves the compression algorithm analysis for those that satisfy all the requirements, which we have reduced to three compression algorithms: Raw, RLE-Bit and Huffman-KOpt.

To perform the experiments, we have based our investigation in a research project focused on providing new display solutions in large facilities such stores or warehouses. The low powered displays will show information about the corresponding item located near the device. The manager program generates this information automatically and configured to update the display whenever necessary.

Figure 2 shows an example image of our scenario. In this case, it represents the information related to an item in a grocery store.

Figure 2. Example of a displayed image

4 Timing Concerns

To perform a successful process, the devices have to take into account the timing requisites that the global system requires. These requisites can be imposed by the global quality of service of the scenario, defined as the maximum delays that the system can tolerate. In this concrete scenario, a tag device is required to show the information related to a product at a certain time.

A system requirement is to change this information as the current information can vary just as the price of an item can rise unexpectedly. An autonomous system is considered to work in real time if these changes occur at the same time that they are produced, but once battery powered devices are in charge of the update process, the techniques used to conserve battery power may induce delays and unwanted results in the final displays. In a wireless scenario, devices often put themselves to sleep to avoid unnecessary energy consumption. In this state, the device cannot be reached by the global coordinator, or central node, and therefore are undetected by the system. Bluetooth LE supports different levels of sleep modes to improve energy requirements.

In this paper, authors provide methods to adapt the sleep algorithm of an individual device to accomplish the maximum permitted delay in a various set of model scenarios. As we explained in the architecture definition, the devices in this experiment are autonomous displays with a battery powered e-paper screen managed by an administrator program and driven by a distributor node antenna, which is always on and connected to a power source.

Opposite to a well know scenario and the previously studied wireless sensor networks (WSN) case [22], the devices don’t need to inform to their control node in a scheduled way, neither the sensor mesh needs to be monitored as a previous work taunt to provide complex algorithms to detect the best way to improve energy in the total network by calculating the sensor devices that need to be powered and working in a certain time. Often these calculations are based on location information and battery status of all sensor network, to work as a whole and provide great results in WSN. The objective of a WSN differs with our scenario due to the main job of the wireless devices, whose needs to be updated from the central node and not the central node updated from the devices.

The administrator of the system can configure different ways of information update mechanism, in this manner, devices should adapt their sleep cycles to provide an efficient use of battery energy by waking up when the next screen update is available at the distributor node. It should be noted that the screen update does occur when measuring the timing algorithms, but they are not taking into account in the delay calculations or next wake-up periods.

Time synchronization between controller and connected device is a key factor to perform a successful synchronization at the expected times. In the solutions proposed, the clock signal at the controller is readjusted in each connection to avoid any undesirable delay due clock signal variation in a long-term situation.

Authors defined different methods that can be implemented in the devices to avoid unwanted power leaks due unwanted wireless communications.

4.1 Time Based Wake-up

When a periodic update is scheduled, devices are configured with a timing delay “td” period between each wake-up procedure and connection to the central node.. The maximum delay “d” in this algorithm will be directly related to the configured interval, as a new display update won´t reach the device until the delay is consumed and next event reached. Static delay implies a uniform power consumption. A lineal relation between number of wireless connection and power consumption is implied. Figure 3 shows a timing progression of an update generation example, following this method.

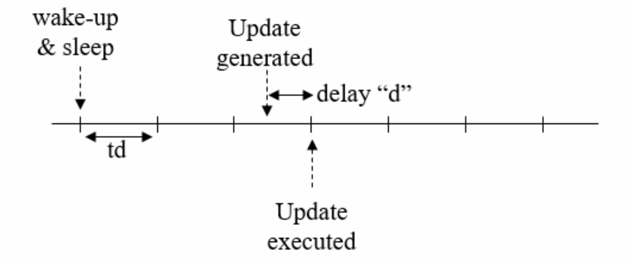

Figure 3. Fixed time events

With this method, “td” value equals the maximum time that the device will wait until asking for new updates, defined as “tmax” hereinafter.

4.2 Dynamic Time Scheduling

When a system administrator is unaware of when the update process could be generated, the time based wake-up method can fail in meeting the scenario requisites. Real time processes, such as frequent screen refresh rates, also require low delay premises that affect with an increase of the power consumption rate. Authors propose several wake-up algorithms that adapt the sleep time of the device based on the previous events. In this manner, the device automatically adapt a different pattern if the behavior of the system changes. These wake-up algorithms aims to reduce the average delay time between update procedures, trying to resemble the performance of the time based wake-up method, but losing the obligation of defining a concrete time value as well as automatize the generation of the duty cycle ranges.

All algorithms have a maximum time “tmax” defined in in order to narrow the maximum tolerable delay, forcing the device to wake-up and connect to the central node to check for unseen updates. Events can integrate the correct time interval based on a previously fixed event, but this requires the controller to know the scheduled next connection to inform the device. Algorithm 1 is presented next, where the device is always updated with a scheduled event and a defined “tmax” is included to avoid large delays in case of an unscheduled update.

Algorithm 1. Scheduled wake-up

```
if tmax | | next scheduled time
    check coordinator for unseen updates
    update tmax
    if unseen updates
      for each unseen update in central node do
        receive update and refresh screen
        receive next scheduled time
      end for
    end if
    update next scheduled time
end if
```

In this case, the device adapts seamlessly to a predefined update plan, thus it can be very affected if the planed update is delayed a certain time, as can be seen in Figure 4. The device in that case must wait until “tmax” is reached and the delay produced equals the “tmax” time value, which can be long in a battery saving strategy.

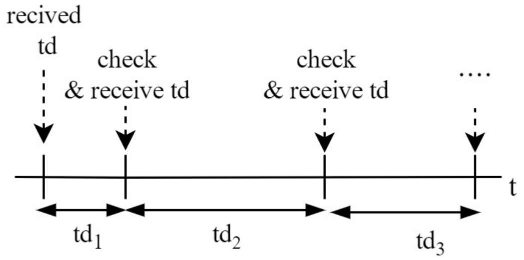

Figure 4. Scheduled wake-up

Algorithm 2 explains a wake-up algorithm focused on a device with no information about the next predicted connection. In a total autonomous operation, the device needs to preserve battery in a conservative manner by only knowing the previous update connections.

Algorithm 2. Increment time connections

```
if tmax | | calculated time
    check coordinator for unseen updates
    update tmax
    if unseen updates
      for each unseen update in central node do
        receive update and refresh screen
        receive next scheduled time
      end for
    end if
    next calculated time=calculateTime(actualTime)
end if
```

The algorithm now adapts the delay time between each connection check increasingly between each connection attempt. Increasing the time delay “td” between interval connections helps the final objective of keeping the device in a sleep state as long as possible, thus the devices are not updated within a real time window. The initial "td1" is updated with a predefined function each time until a "tmax" is reached, which will be the maximum time the device can be in the sleep state. In this paper, a simple *two times* the previous value formula is used. As indicated, this is defined by the administrator. With this approximation, the device will benefit continuous updates by keeping the delay "d" low in opposition to longer update intervals, which will have longer delay times between the update generated moment and the update execution time in the next connection try. Once a successful connection with a new update occurs, the device starts again with a new increasing algorithm to calculate the time between wake-up procedures. Figure 5 shows an example of a device with this wake-up algorithm.

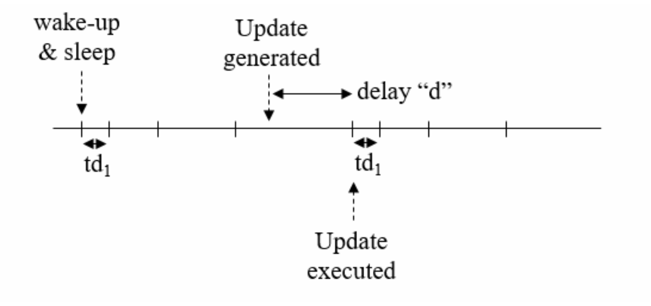

Figure 5. Increasing time delay

5 Compression Methodologies

In this chapter, implementations of different methodologies for image transmission are explained. We will describe how the information is modified to optimize the use of the wireless transmission using the selected compression methods in our specific scenario. We use the display screen to validate the selected compression algorithms and prove the correctness of the result visualization.

The main objective of this experiment is to measure and compare the advantage of wirelessly transmitting compressed data and processing it in destiny than a raw data transmission using more time in the transmission process and expecting more energy spending. We defined that the compression algorithms have to be lossless, so we can assume the validity of the received data in any of the following scenarios.

The selected methods are now explained:

5.1 Raw

The raw method sends the information entirely through the wireless communication channel with no modification. The display data are formed by putting together all the pixel information line by line and sending them in packets over the Bluetooth connection. Optimizing the transmission for a Bluetooth LE connection is made in order to ease and maximize the throughput of the communication channel.

Note that we can consider the Bluetooth connection as a zero error channel, thus once the information is correctly received, the device does not modify the data and starts relaying it to the display device by powering on the additional PCB. Once the display is refreshed, the device powers off the display and enters sleep mode until the next update display event.

The Bluetooth LE data are optimized by default for packets of 64 Bytes, this way we divide the display information in slices of 64 Bytes of information to be transmitted. In our scenario, the total size of the display is fixed because of the screen size selected, therefore, in the raw method, we can establish the total number of packets to be sent: 90 packets of 64 bytes and one additional of 48 bytes to carry all pixel information. To adapt other screen sizes to be transferred by the Bluetooth channel, the stream has to be sliced in packets of 64 bytes.

As there is no compression of the image data, we can assume this method will induce the large quantity of information to be transmitted and therefor the time spent with the wireless transmission powered on will be the most. To avoid other methods that may produce a large image data to be transmitted, the server will use this raw method as the safe option in the case of other methods fail to compress the image data. This feature is disabled in the experiments as the transmitted images will be the same for all scenarios.

5.2 RLE-Bit

As the hardware limitations and the procedures to interact with the screen are strict in terms of data ordering, compression and decompression processes have to be in a sequential form, a variation of the RLE compression method has been developed to achieve a compatible and lossless compression algorithm for the transmitted images.

We modify the basic RLE algorithm defined in [11] to simplify the generation of sequential data. The compression algorithm starts by identifying the first bit value of the stream. This will be the first bit of our compressed result. Once the first bit is written, then we count the number of same value consecutive bits and concatenate it converted in a bit codification. Note that in order to correctly identify the count number of the same bits, we limit the maximum decimal value to 127 to fill out a complete byte packet: 1 bit for defining the bit level of the stream and 7 bits for the decimal count of the same bits.

Let assume we have the following stream of bits: "0000 11111 0 111111 000000 1111111", the compressed HEX stream will be: "04 15 01 16 06 17". A graphical representation can be seen in Figure 6.

In the previous example, the compressed result is larger than the original stream, this is because the need of defining a fixed packet size in each chunk of contiguous bits. To ease the decoding, we have chosen an 8 bit packet, thus for a successful compression of

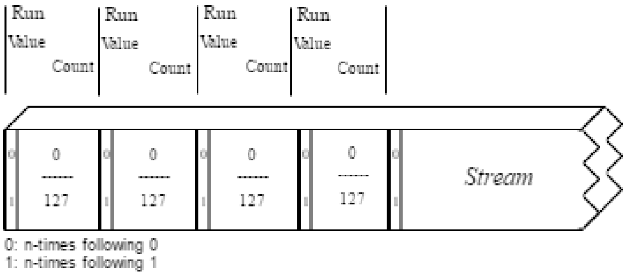

Figure 6. RLE-Bit algorithm graphical representation

the original data, the continuous same bit stream needs to be at least of an 8 bit consecutive stream.

This fits well in our image scenario, the retail images are often made by text and few images, and this is why the large white or black spaces in the image help the generation of a continuous same bit stream that is appropriate for this algorithm.

In our scenario, the original size of the image is 5808 bytes, each pixel is represented by a bit in the stream, building the data set to be processed by our RLE-Bit algorithm. After the compression, the final size average of our 100 sample images is 2204 bytes with a standard deviation fixed in 178, obtaining a compression rate of 2,64:1. The resulting data can be sent by wireless transmission using less time than the original stream but consequently, it will require a process time in the device before it can be relayed to the display.

5.3 Huffman-KOpt

Implementing a dictionary based algorithm is challenging for this scenario due to the limited memory available in the display device and the small size of the selected display. Algorithms in which dynamic dictionaries have to be generated based on the original data have to be carefully selected because the original image size is close to the overall capacity of the device, and the memory allocation requirements for an adaptive algorithm code would often be larger than the image itself in certain cases.

This is why a previous work has to be done in order to determine the optimal values to adapt a Huffman coding to this scenario. In particular, as the image is composed basically with black text and white background, we would try to obtain an optimal Huffman coding to take advantage of the described scenario and avoid the requirements of a low-resources device. Although in our scenario, using a common code for all generated images is reasonably, because the similarity between all generated images a code generation for each image would fit better in a generic system.

Given a set of random images, we can study how the length of the symbols to be coded (K) can affect the final size of the compressed data. As the K grows up, the complexity of the code increases due the quantity of the symbols to be generated. Particularly, the total number of symbols will be 2^K, and every symbol must have the corresponding code. The algorithm implemented is described in [18].

Huffman compressor is configured to generate a canonical code [19] to be sent along with the compressed data at the beginning of the transmission, this way the code is generated optimally for each image in each refresh of the display. The canonical code is a very compact code representation which only stores the bit length of the symbol coding in a sequential order. In this manner, the decoding algorithm can be reduced and simplified so that it is computationally efficient.

We will study how a different values of K lengths, can vary the final data size to be transmitted. As the information is sliced in larger fragments, the information to encode turns into a more complex code. Figure 7 shows the final compressed size of the image to be transmitted related to the length of the K value. The compression rate increases due the reduction of the slices in which the message is divided. The original size of the image is 5808 KB.

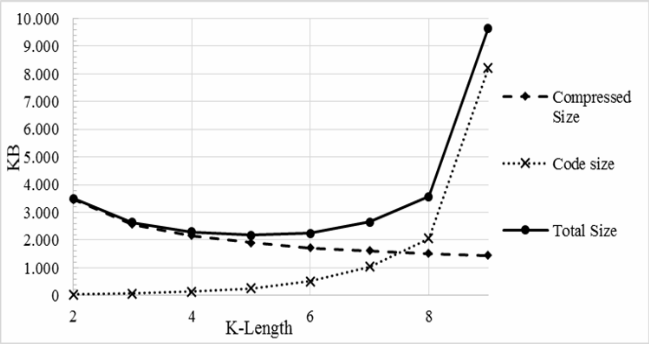

Figure 7. K-Length comparison

Code also get complex as the K value increases. The size of the code to be stored is 2^K different symbols and, using a canonical code, coding a symbol uses a byte size representation for each one if the selected K value is less than 8 bits. If the K is more than 9, a two byte length is needed in order to be capable of representing longer values, this is why a big increase in code size is shown in Figure 7. This study leads to select a K value of 5 as it allows a better compression rate, it is optimized to use byte words in memory and the maximum size of the code fits with a margin in the SRAM memory of the device.

Device limitations and the total amount of data transmitted are restricting higher K values, as the coding algorithm will not benefit the increased compression rate against total transmitted code.

6 Experiments Implementation

In this paper, two main aspects are involved; for the first experiment, a scenario where devices are assumed plenty functional is deployed and the timing algorithms are evaluated in a simulation experiment. These two experiments are uncorrelated to each other and conclusions will be extracted for each proposal.

The main objective of the first experimental scenario is to determine the differences between proposed timing algorithms in terms of updates delay and real time capabilities as well as to determine the maximum delay that each algorithm incorporates. In the second experiment, authors try to determine whether sending a compressed data stream through a wireless low power transmission and then using more time of data processing in the device, increases the battery life-time of a wireless device.

6.1 Delay Times Evaluation

In order to evaluate the proposed timing algorithms, a simulated scenario is implemented to provide different sources capable of generating update processes. The need to provide several sources behavior relays on the objective of differentiating the more adequate algorithm for a specific application.

As there are various possible scenarios in a real situation, authors have distinguished three types of source behavior related to the update scheduling generation: a random notification generator, a scheduled generator and a time fixed source of updates.

All of them are tested against the proposed algorithms in order to evaluate the average delay time produced in the screen update process.

The random notification generator is seemed as an unknown source that produces screen updates with a random output, not knowing the time in which the update will be generated. A second scheduled source is added to the simulation with a prearranged update plan, containing different wait times between updates. As the plan is known previously, the source knows when the next update will be so that information can be transmitted to the devices and used for a better wake-up planning. The third simulated source is driven by a fixed update plan, with a time based update plan, with a selectable time between each generated update.

Authors resume the specific configurable values to each source to create the simulation experiment as indicated in Table 2.

Table 2. Source parameters

	Random source	Scheduled source	Fixed update plan
Avg. time between updates	2 hours	2 hours	2 hours
Minimum wait value	1 minute	1 minute	-
Maximum wait value	4 hours	4 hours	-
Generation method	randomly	randomly, knowing the next update time	Fixed

Devices are configured with a “tmax” of 1 hour, to provide a real world situation and narrow the different algorithms adaptation in an hourly based window.

6.2 Compression Algorithms Evaluation

To evaluate the proposed algorithms, the experiment comprises measurements of the power consumption in each scenario. The devices developed in the research laboratory allow measuring the battery voltage in order to detect the remaining energy in a certain time. The process to evaluate the different scenarios are decomposed as indicated in the next subsections.

6.2.1 Context

To resolve the second evaluation, we performed a run-time test for each compression algorithm in order to evaluate its energy consumption. The devices were deployed with new fully-charged batteries with an initial voltage of 4.2V. The batteries used in the experiment are 1.050 mAh Li-Po. These batteries operate in a voltage range of 4.2 V to 3.3 V, in which it is considered empty. We monitored the battery voltage in order to compare the energy consumption between the experiments.

To adapt the experiments as close as possible to a real scenario, we will use the images generated by the grocery store program to test our different algorithms. A hundred image labels were selected and extracted to be compressed by our algorithms and we compared the performance of them. All results showed in the paper are the average of the results obtained applying the same algorithms to the extracted images. Also, the standard deviation is shown.

To characterize the discharge curve of the battery, the state of charge (SOC) is extracted from the linear mapping of the relation between the open-circuit voltage (VOC) and SOC proposed by H. Rahimi-Eichi et al. [12].

Each compression algorithm was used to transmit and refresh each display device 500 times. Once the experiment finished the batteries were measured. The experiment was repeated 5 times and in order to synthesize the results, the medians of the data are presented. Raw compression method has been chosen to represent the standard result and the others will be related to it in order to evaluate the results.

6.2.2 Plan

The experiments were divided into three phases: Devices preparation, algorithm testing and measurement and data evaluation.

Devices preparation. The first phase is dedicated to the preparation and the initialization of the devices. First, the devices were plugged into a power source in order to be initialized and configured with the appropriate program in which the different decompression algorithms are implemented. Once the devices were ready to receive data, the power source is switched and the batteries start to power them. Afterwards, the devices are ready in standby mode waiting for incoming display data.

Environment variables have been taken into account by controlling the room temperature and humidity at the same level across all tests. Batteries were charged using the same methods in every case and tested before each measure to verify any malfunction.

Algorithm testing and measurement. After devices preparation, this phase aims to measure the behavior of the proposed algorithms. The devices started with a full charged battery and connected to the gateway in order to start receiving display updates. A total of 500 times was each algorithm tested, each time the data was sent using the selected algorithm, the display refreshed and an idle sleep time of 10 seconds was added. Then, the procedure starts again and at the end of the final iteration, the display was turned off by disconnecting the battery and the measurement was performed with a calibrated voltmeter in an open voltage circuit design. All of the procedures were performed under the same environmental conditions of temperature and humidity.

The information about the electric charge of the battery [23] was measured using a remotely controlled multimeter connected by USB. The experimental setup can be seen in Figure 8.

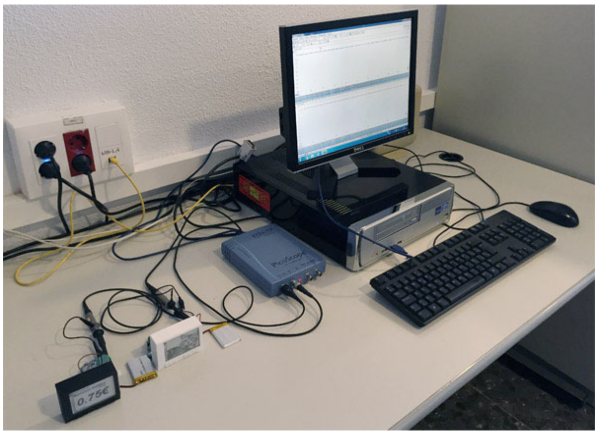

Figure 8. Test setup

Data evaluation. The information collected when all the experiments were completed, was analyzed to synthesize the information and present it, taking special attention to eliminate any outliers or measurement errors.

6.2.3 Data Collection

The information collected aims to compare the performance of the proposed algorithms for data compression. Figuring out what compression is the least energy consuming option and how it affects the transmission time or the process time of the information. Balancing those variables we try to establish a relation between battery life and the cost of the data manipulation process.

Registering the number of display changes completed and the final battery voltage after the display updates in each compression mode, we achieve a formal comparison between them.

7 Results

This section presents the results of the measurements executed in the experiments. Results are shown following the procedures explained in the previous section.

Figure 9 compares the results obtained in the simulation of the average delay time implied in each wake-up algorithm proved against the different sources defined in the experiment definition.

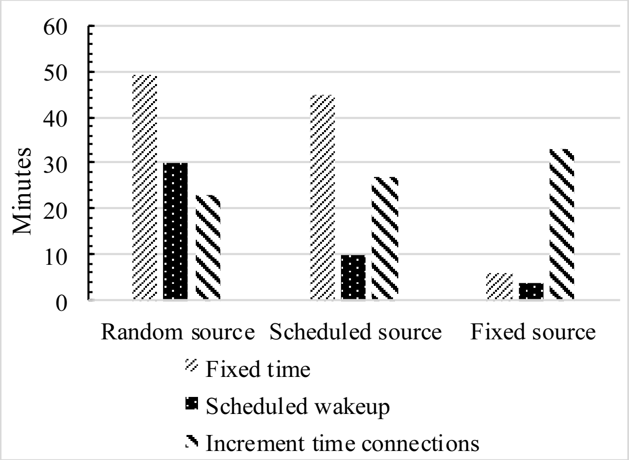

Figure 9. Wake-up algorithms comparison

As can be seen in the results, each wake-up algorithm adapts differently to the update generator sources. Regarding the first source, as it is programmed randomly, the fixed timed algorithm generates the worst delay time case, produced by the poor adaptation and synchronization between the devices and the update source. Scheduled wake-up and increment time connections algorithms work similar due to the lack of a scheduled planning into the source, and consequently the scheduled wake-up relays the wake-up timing to the increment time connections method. Scheduled source adapts better to the scheduled wake-up algorithm, since it is using the information of the following update to calculate the next wake-up instant. Last, a fixed source adapts almost analogous to the fixed algorithm and scheduled wake-up, which is because the knowledge of the following update as well as the fixed algorithm is configured to wake-up at the same rate that is the update source.

Related to the second experiment, Table 3 summarizes the voltages obtained in both, previous and after, instant moments of the experiment for each compression method. Some considerations should be given to Table 3.

Table 3. Battery voltages

	Before (V)	After (V)	MD
Raw	4.20	3,863	0.034
RLE-Bit	4.20	3,912	0.032
Huffman-KOpt	4.20	3,897	0.049

The voltage was measured at the moment of initiating the experiment to assure there was no degradation in the voltage after the charging process. The batteries were conserved in a same location under the same conditions after the preparation procedure, and then equally connected to the prototype devices. In the experiment the devices were constantly refreshing their displays with the corresponding transmission time in each step. The authors predefined the number of changes based on previous experiments and the prevision of the expected battery life in a normal operation mode.

The experiments were repeated 5 times in order to detect potential variations. Figure 10 shows the final voltage of each compression algorithms with their respective maximum deviation. We can see that there is a minimal deviation in each experiment, most probably related to slight differences in the charging process of the battery itself.

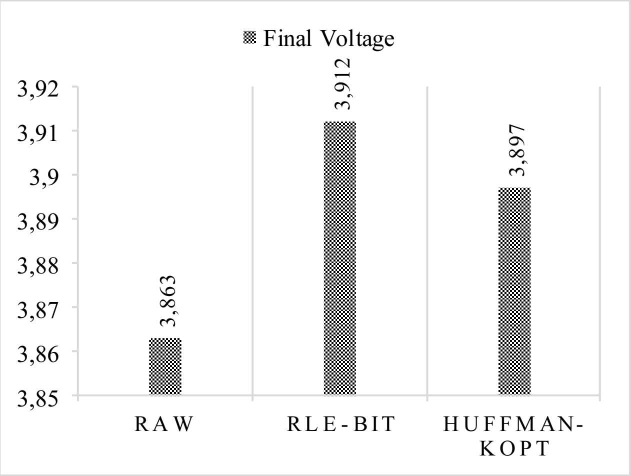

Figure 10. Final voltage comparison (V)

As we can see in the results, the algorithms used in the transmission affects the remaining battery of the devices after a succession of display refreshing processes. As long as the information is more compressed, the less time is necessary to keep powered on the wireless interface of the devices, and in consequence, less battery is consumed. In a standard scenario, with the RAW mode established, the remaining battery voltage averages 3.863V, which based on the discharge curve of a Li-Po battery [12], fixes the remaining battery percentage at 64%. With this measure, an improvement has been done with the addition of the additional compression algorithms proposed in this work.

Following, the result values in the case of the RLE-Bit, it is clear an increase in the remaining battery with an average of 3,912V, which implies a remaining 70% of battery, and a 6% improvement over the RAW method. In case of Huffman-KOpt, the results also reflect an improvement over RAW transmission, but less than RLE-Bit, this is due the bigger average data size and also the complex decompression algorithm. The final battery voltage in the last scenario was 3,897V, which corresponds to a 68% of remaining battery and a 4% improvement over the RAW method.

8 Conclusions

In this paper, a proposal and assessment of different algorithms for wake-up methodologies and data compression are presented and compared in order to reduce battery consumption in wireless devices. Selecting a correct wake-up algorithm based on the final application is a key factor to optimally improving the battery utilization in a low power device. In addition, results show that sending large data is less efficient that seeding a compressed stream and then using some computational effort to decompress the original information. Two compression algorithms have been tested against the raw method proving that a reduction in the data sent over the wireless channel and reducing the time that it is powered on, increases the battery life-time regardless the over-processing added in the decompression process. Algorithms used in this paper have been adapted to provide an optimal compression of the source image data.

Discerning between data manipulation procedures allows a more optimal use of the battery and processing capabilities in low-resource devices and it is a key factor to improve the battery life-time efficiency and extract the maximum performance of a device. It is clear that there must be a balance between the power need for the data compression and the data transmission.

In this particular test scenario, a concrete experiment has been carried on to provide a battery life-time increment for E-Ink display devices. The images implied in the wireless transmission process have similar characteristics in terms of the content displayed that allow a great compression rate using low-resource algorithms with simple implementations.

Future works involve the usage of more complex algorithms that may support data regeneration and lossy transmission. These algorithms will demand more processing power than the algorithms used in this research and results may vary if the device needs more processing time to reconstruct the original data.

Acknowledgments

This work is funded by the Ministry of Economy and Competitiveness through SEMOLA project

(TEC2015-68284-R) and from the Autonomous Region of Madrid through MOSI-AGIL-CM project (P2013/ICE-3019), co-funded by EU Structural Funds FSE and FEDER).

References

[1] J.-S. Lee, M.-F. Dong, Y.-H. Sun, A Preliminary Study of Low Power Wireless Technologies: ZigBee and Bluetooth Low Energy, *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, Auckland, New Zealand, 2015, pp. 135-139.

[2] R. Sharma, A Data Compression Application for Wireless Sensor Networks using LTC Algorithm, *2015 IEEE International Conference on Electro/Information Technology (EIT)*, Dekalb, IL, 2015, pp. 598-604.

[3] T. Ma, M. Hempel, D. Peng, H. Sharif, A Survey of Energy-efficient Compression and Communication Techniques for Multimedia in Resource Constrained Systems, *IEEE Communications Surveys & Tutorials*, Vol. 15, No. 3, pp. 963-972, Third Quarter, 2013.

[4] N. I. Mondal, S. U. Zaman, A. A. Masud, J. Alam, Comparisons of Maximum System Lifetime in Diverse Scenarios for Body Sensor Networks, *2008 11th International Conference on Computer and Information Technology (ICCIT)*, Khulna, Bangladesh, 2008, pp. 73-78.

[5] K. Nair, J. Kulkarni, M. Warde, Z. Dave, V. Rawalgaonkar, G. Gore, J. Joshi, Optimizing Power Consumption in Iot Based Wireless Sensor Networks Using Bluetooth Low Energy, *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, Noida, India, 2015, pp. 589-593.

[6] E. G. Pereira, R. Pereira, Video Encoding and Streaming Mechanisms in IoT Low Power Networks, *2015 3rd International Conference on Future Internet of Things and Cloud (FiCloud)*, Rome, Italy, 2015, pp. 357-362.

[7] Y. Nimmagadda, K. Kumar, Y.-H. Lu, Energy-efficient Image Compression in Mobile Devices for Wireless Transmission, *2009 IEEE International Conference on Multimedia and Expo (ICME)*, New York, NY, 2009, pp. 1278-1281.

[8] R. Xu, Z. Li, C. Wang, P. Ni, Impact of Data Compression on Energy Consumption of Wireless-networked Handheld Devices, *Proceedings of 23rd International Conference on Distributed Computing Systems*, Providence, Rhode Island, 2003, pp. 302-311.

[9] A. Kumar, B. Singh, Optimizing Compression for Power Efficient Wireless Multimedia Sensor Networks, *2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, Madurai, India, 2015, pp. 1-4.

[10] D. S. de Rivera, R. Alcarria, D. M. de Andres, B. Bordel, T. Robles, An Autonomous Information Device with E-paper Display for Personal Environments, *2016 IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, 2016, pp. 139-140.

[11] A. Banerjee, A. Halder, An Efficient Image Compression Algorithm for Almost Dual-color Image Based on K-means Clustering, Bit-map Generation and RLE, *2010 International Conference on Computer and Communication Technology (ICCCT)*, Allahabad, Uttar Pradesh, India, 2010, pp. 201-205.

[12] H. Rahimi-Eichi, F. Baronti, M.-Y. Chow, Online Adaptive Parameter Identification and State-of-charge Coestimation for Lithium-polymer Battery Cells, *IEEE Transactions on Industrial Electronics*, Vol. 61, No. 4, pp. 2053-2061, April, 2014.

[13] N. Kimura, S. Latifi, A Survey on Data Compression in Wireless Sensor Networks, *2005 International Conference on Information Technology: Coding and Computing (ITCC)*, Las Vegas, NV, 2005, pp. 8-13.

[14] S. Puthenpurayil, R. Gu, S. S. Bhattacharyya, Energy-aware Data Compression for Wireless Sensor Networks, *2007 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'07)*, Honolulu, HI, 2007, pp. II-45-II-48.

[15] T. Szalapski, S. Madria, Toward Energy Efficient Multistream Collaborative Compression in Wireless Sensor Networks, *2014 International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)*, Miami, FL, 2014, pp. 124-133.

[16] J. A. Rogers, Z. Bao, K. Baldwin, A. Dodabalapur, B. Crone, V. R. Raju, V. Kuck, H. Katz, K. Amundson, J. Ewing, P. Drzaic, Paper-like Electronic Displays: Large-area Rubber-stamped Plastic Sheets of Electronics and Microencapsulated Electrophoretic Inks, *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 98, No. 9, pp. 4835-4840, April, 2001.

[17] V. Raghunathan, C. Schurgers, S. Park, M. B. Srivastava, Energy-aware Wireless Microsensor Networks, *IEEE Signal Processing Magazine*, Vol. 19, No. 2, pp. 40-50, March, 2002.

[18] D. A. Huffman, A Method for the Construction of Minimum-redundancy Codes, *Proceedings of the IRE*, Vol. 40, No. 9, pp. 1098-1101, September, 1952.

[19] Y. Nekritch, Decoding of Canonical Huffman Codes with Look-Up Tables, *Proceedings DCC 2000- Data Compression Conference*, Snowbird, UT, 2000, p. 566.

[20] Arduino, https://www.arduino.cc/

[21] Bean – Punch Through, https://punch through.com/bean/

[22] G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, Energy Conservation in Wireless Sensor Networks: A Survey, *Ad Hoc Networks*, Vol. 7, No. 3, pp. 537-568, May, 2009.

[23] M. Coleman, C. K. Lee, C. Zhu, W. G. Hurley, State-of-charge Determination From EMF Voltage Estimation: Using Impedance, Terminal Voltage, and Current for Lead-acid and Lithium-ion Batteries, *IEEE Transactions on Industrial Electronics*, Vol. 54, No. 5, pp. 2550-2557, October, 2007.

Biographies

Diego Sánchez-de-Rivera holds a B.Sc and a M.S. degree in Telecommunication Engineering from Technical University of Madrid. Currently, he is a Ph.D. student at the E.T.S.I Telecommunications of the Technical University of Madrid. His research interests are Internet Of Things, Sensor Networks, and Web Development.

Ramón Alcarria received his M.S. and Ph.D. degrees in Telecommunication Engineering from the Technical University of Madrid in 2008 and 2013 respectively. Currently, he is an assistant professor at the E.T.S.I Topography of the Technical University of Madrid. His research interests are Service Architectures, Sensor Networks, Service Composition and Prosumer Environments.

Diego Martín received his doctoral degree in 2012, holds a B.Sc in Computer Engineering and an M.S. in Computer Science from the Department of Informatics at the Carlos III University of Madrid, Spain. His main research areas are Software Process Improvement, Knowledge Management, Reutilization and Prosumer Environments.

Gabriel Huecas obtained its Ph.D. on Telecommunication Engineering (1995), at the Universidad Politécnica de Madrid. Technical University of Madrid. From 1989 assistant researcher at Dpto. de Ingeniería Telemática (DIT-UPM). Currently, he is associate professor at Technical University of Madrid.