Applications of LoRaWAN in SCADA Systems

Nijat Hasanov
Process Automation Engineering department
Baku Higher Oil School
Baku, Azerbaijan
nijat.hasanov.b0469@bhos.edu.az

Ali Pourmohammad
Process Automation Engineering department
Baku Higher Oil School
Baku, Azerbaijan
pourmohammad@bhos.edu.az

***Abstract*—Internet of Things (IoT) introduces a new term called "connected objects" which are devices that can communicate with each other and the servers over wireless networks. Although technologies like WIFI, Bluetooth, 3G and 4G are used when the power consumption and long-range connection are not critical, many emerging IoT applications require to operate on battery power and communicate over large distances. Therefore, new IoT connectivity family called LPWA (Low Power Wide Area) is introduced to the market and LoRaWAN is one of the few famous LPWA technologies used for low-power long-range applications. This paper explores possible applications of LoRaWAN in the oil & gas industry (SCADA Systems). Firstly, it discusses the LoRa modulation technique (its difference from LoRaWAN), and the LoRaWAN network architecture briefly. Then, it moves on to the explaining of the possible applications of LoRaWAN in SCADA systems. Moreover, it explains the details of building a simple LoRaWAN network based on the work done by the authors, which includes developing embedded software and building the circuits. Finally, the data from the experiments to test range and power of the signal are given and discussed at the end.**

Keywords— IoT, LPWA, LoRaWAN, LoRa, Oil & Gas industry, SCADA Systems

I. INTRODUCTION

Energy-efficient and cost-effective connectivity is crucial for low-power and devices and systems. Traditional communication methods have become insufficient considering their energy usage, cost, and communication range. There are new emerging technologies which are scalable and deployable for large-scale applications. LPWA networks are emerging in the market which means Low-Power Wide-Area. It is not a technology, but it is used to refer to any network that is used in communication of low-power devices and systems. In addition, LPWANs cover large distances unlike other low power networks such as Bluetooth or NFC. Trade-off is that LPWANs allow only small amounts of data to be transmitted. While LTE Advanced and upcoming 5G technologies are developed to have gigabits per second speeds, speed of transmission in LPWA networks is usually a few kilobits per second. However, LPWA technologies cover larger distances–hundreds of kilometers (figure 1) [1].

LPWA networks are not very useful for many consumer applications, which includes transmission of video, audio, or text messaging, due to its low bandwidth. They are exclusively intended for Internet of Things (IoT) devices and machine-to-machine communications. In short, LPWA networks are useful in cases where the device needs to send small packets of data to large distances while consuming as less power as possible. This feature are main differences of LPWA networks from the other wireless networks like Bluetooth, WIFI, Zigbee etc. [1]. Some other LPWA technologies as NB-IOT are under development which is also known as cellular LPWA. Active research and development continue for NB-IOT, LTE-M, 5G and similar networks but it is not clear that if they are suitable for long-term solutions. These networks are also not cross-compatible. The mesh networks such as Zigbee network are intended for medium ranges and does not offer long-range capabilities. As each node of a mesh network should receive and repeat RF signals, it is not battery efficient. Especially, mesh networks do not fit the requirements of LPWA network in a large scale. Finally, there are local RF networks like Bluetooth and NFC, but their range makes them useless for many long-range applications (figure 2) [2-5].

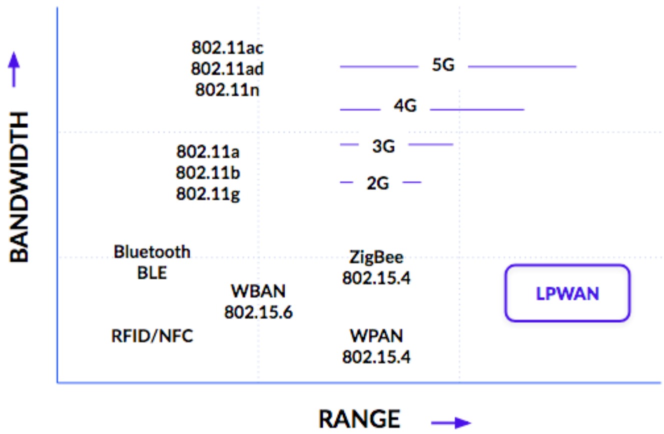

Fig. 1 Comparison of wireless networks in Range [1]

Feature	LoRaWAN	Narrow-Band	LTE Cat-1 2016 (Rel12)	LTE Cat-M 2018 (Rel13)	NB-LTE 2019(Rel13+)
Modulation	SS Chirp	UNB / GFSK/BPSK	OFDMA	OFDMA	OFDMA
Rx bandwidth	500 - 125 KHz	100 Hz	20 MHz	20 - 1.4 MHz	200 KHz
Data Rate	290bps - 50Kbps	100 bit/sec 12 / 8 bytes Max	10 Mbit/sec	200kbps – 1Mbps	~20K bit/sec
Max. # Msgs/day	Unlimited	UL: 140 msgs/day	Unlimited	Unlimited	Unlimited
Max Output Power	20 dBm	20 dBm	23 - 46 dBm	23/30 dBm	20 dBm
Link Budget	154 dB	151 dB	130 dB+	146 dB	150 dB
Batery lifetime - 2000mAh	105 months	90 months		18 months	
Power Efficiency	Very High	Very High	Low	Medium	Med high
Interference immunity	Very high	Low	Medium	Medium	Low
Coexistence	Yes	No	Yes	Yes	No
Security	Yes	No	Yes	Yes	Yes
Mobility / localization	Yes	Limited mobility, No loc	Mobility	Mobility	Limited Mobility No Loc

Fig. 2 Comparison of different LPWA networks [5]

978-1-7281-8566-8/20/$31.00 ©2020 IEEE

II. LoRaWAN Networks

LoRa is founded by a French startup and it is bought by Semtech that added MAC layer to LoRa physical layer to standardize and extend it onto internet networks. Architecture of the system and coomunication protocol is defined by LoRaWAN while long-range communication is made possible with LoRa physical layer. LoRaWAN encapsulates important wireless network functionalities like adaptive data rate optimization, end-to-end encryption, quality of service and etc. Battery lifetime, security and network capacity are mostly determined by protocol and the architecture of the network [1].

LoRa is a wireless modulation method used in physical layer to establish long range communication. It is common to come across Frequency Shift Keying (FSK) modulation in the physical layer of many legacy systems as it is one of the most power-efficient modulations. LoRa uses chirp spread spectrum modulation that increases the communication range significantly while maintaining low power consumption as FSK. Long range communication capability is the main advantage of LoRa technology. The range is heavily dependent on the obstructions between the nodes and gateways, but link budget of LoRa technology is higher than any other standard communication methods [2-5]. Traditional principles of Spread Spectrum are extended in LoRa modulation to minimize the required energy for transmission of the bits over the channel. To compute the Data Rate of communication, bandwidth (BW), Spreading Factor (SF) and Coding Rate (CR) is used. The most important variable affecting the quality of service is SF. Lower values of SF allows to achieve higher data rates and low airtime. Increasing SF will result in longer ranges of communication while reducing the Quality of Service. Table I presents relation among SF [values×(Chirps/Symbols)] and Demodulation SNR. The relationship between BW, Symbol Rate (RS) and SF is [6]:

$$RS = \frac{BW}{2^{SF}} \tag{1}$$

Another important issue for real-time applications is frame airtime. Transmission time for LoRa frame can be calculated using SF, BW and CR. It is the sum of time to transmit preamble and payload as:

$$T_{frame} = T_{preamble} + T_{payload} \tag{2}$$

To compute $T_{preamble}$, time to send one symbol and length of preamble should be added, as:

$$T_{sym} = \frac{1}{RS} \rightarrow T_{preamble} = (n_{preamble} + 4.25)T_{sym} \tag{3}$$

Payload size $n_{payload}$ defines the frame transmission time. Payload size is computed using Implicit Header (IH), Coding Rate (CR), Packet Length (PL) and Low Data Rate Optimization (DE). IH is 1 if the header is disabled. Predefined CRC and CR values are used. DE value is 1 if low data rate is set. Then, $n_{payload}$ and $T_{payload}$ are calculated as [7]:

$$n_{payload} = 8 + max(ceil\left[\frac{8PL-4SF+28+16CRC-20IH}{4(SF-2DE)}\right](CR+4), 0) \tag{4}$$

$$T_{payload} = n_{payload} \times T_{sym} \tag{5}$$

Equations (2-5) allow us to calculate airtime for a single LoRa frame. To sum up, frame airtime of LoRa packets are mostly dependent on SF. High SF values result in high airtime.

Usually, mesh network architecture is used by many existing networks. Individual nodes receive and forward the data from other nodes to increase communication range and cell size. However, it increases complexity, reduces capacity of network and battery lifetime because the nodes are receiving and forwarding irrelevant data for them. Therefore, the most sensible architecture to use is long range star for minimizing power consumption while enabling long range communication. Communication is asynchronous in LoRaWAN network and nodes transmit when they have data ready. This type of communication is called Aloha method. On the other hand, the nodes in cellular or other mesh networks must wake up and send data to synchronize and check messages. Synchronization is the number one factor that reduces the battery lifetime [7].

Recent studies comparing various LPWAN technologies show that LoRaWAN is 3-5 times better than other technologies [4]. End-devices in LoRaWAN network may serve different purposes, therefore, requirements differ for each node. Different device classes are available in LoRaWAN to optimize the operation of the end applications. Tradeoff between downlink latency and battery lifetime is changed in every class. In actuator-type applications, one of the most important factors is downlink communication latency. Three classes of communication profiles A, B and C are made available for LoRaWAN as figure 3. Class A is the most power efficient class among the three [1-7].

Security is one of the most important features of any LPWAN. Two layers of security are incorporated in LoRaWAN. One of them is for application and the other is for network. Authenticity of the end-node is ensured by the network security whereas the application security eliminates the access of network operator to the application data of the user. IEEE EUI64 identifier is used in the key exchange of AES security. Authentication and encryption is relied on a just single key in many systems while they are separated in LoRaWAN to authenticate packets and protect the integrity [1-7].

TABLE I. LoRa SF values [6]

Spreading Factor	Chirps/Symbol	Demodulation SNR
7	128	–7.5 dB
8	256	–10 dB
9	512	–12.5 dB
10	1024	–15 dB
11	2048	–17.5 dB
12	4096	–20 dB

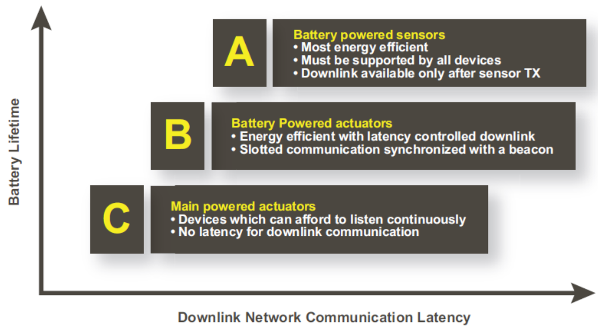

Fig. 3. Device classes in LoRaWAN [4]

978-1-7281-8566-8/20/$31.00 ©2020 IEEE

III. LoRaWAN Networks and SCADA Systems

SCADA (Supervisory Control and Data Acquisition) system is a central control system made up of I/O devices, network interfaces, communication equipment and software. SCADA is used for controlling and supervising industry processes, such as production, manufacturing, and development. Oil and gas, water distribution or electrical power systems can be examples where SCADA is used. SCADA system measures the variables in the process at regular intervals and checks sensors to ensure process is stable, so, minimum human interaction is required [8].

A SCADA station is a single PC or multiple PCs which is also called a server (master). Data collecting devices use process controllers (PLCs and RTUs) to communicate with field devices. Remote Terminal Units (RTUs) are mainly used for collecting data from the sensors in remote places (figure 4). They are spread out to wide geographical area. Therefore, wireless networks are widely used for establishing communication between data servers and RTUs. In some applications such as oil & gas or water & wastewater, wireless network has become the only option because of the remoteness of site. Signals from sensors are converted to digital data by RTUs and sent to the master. Usually, RTUs must operate on battery because there is no continuous power supply available in remote places. That is why, it is crucial for RTUs to use minimum energy, so they can stay powered for a long time. Selection of wireless network has a huge impact on the energy consumption of the RTU. Table II depicts two widely adopted networks for RTU communication and compares them with LoRaWAN [8-9].

Ethernet protocol offers bandwidth of more than 100Mbps and covers distances up to 8km. New versions of the ethernet wireless bridges use 5GHz signals. The communication is secured using 128-bit AES encryption. However, power consumption of the ethernet bridges are too high to operate on battery. Usually, 300-400mA current at 12V is required for an ethernet bridge. For example, monitoring a remote well in oil & gas industry does not require high speeds like 100Mbps, but there is a need for RTU to feed from battery. GSM network covers enormous ranges depending on the coverage of the network provider. Data rate of GSM is usually not more than a few megabits per second. This connection speed is ideal for streaming media but not required for sending a few bytes of sensor data to SCADA host.

The power consumption is relatively low compared to Ethernet bridges but not suitable for operating on battery [8]. LoRaWAN is offering better coverage than Ethernet protocol in rural environments. But it does not offer high performance in urban environments for the sake of saving power. Data rate is much lower than GSM or Ethernet but suitable for RTU applications. The most important point is that the power consumption of LoRaWAN allows it to operate on battery for years in remote places. With the rise of LPWA networks, reliability of LoRaWAN is improved constantly and it is already being implemented for industrial applications. Based on the comparison, it can be concluded that LoRaWAN satisfies the requirements of many SCADA applications better than the traditional networks.

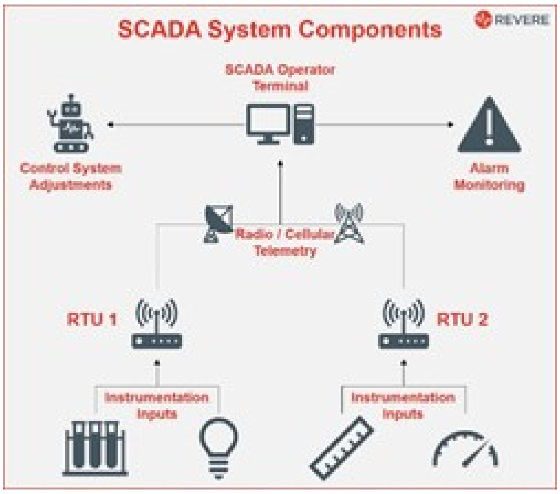

Fig. 4 Typical SCADA system [9]

TABLE II. COMPARISON OF WIRELESS SCADA NETWORKS AND LORAWAN [8]

	LoRaWAN	**Ethernet using wireless bridges**	**GSM**
Typ. Range	5km(urban) 15km(rural)	8 km	35km
Data Rate	50kbps	100Mbps	1Mbits+
Security	Yes	Yes	Yes
Power consumption	Low	Very high	High

IV. Implementations and Discussion on The Results

The LoRa hardware architecture includes [1]:

1. Battery or power plug are two options for the power supply
2. All device functionalities are controlled by a MCU which also implements the LoRaWAN stack
3. LoRa transceiver, antenna circuitry and the antenna are parts of the LoRa Radio
4. Various peripherals can be connected such as, temperature sensors or I/O devices

Couple of choices are available to make a LoRa device based on the production and design limitations [1]:

1. Design using LoRa chipset
2. Design using LoRa Module
3. Design using RF-MCU
4. Design using LoRa Modem
5. An existing device which has a LoRa modem.

Common software architecture of a device is given as figure 5.

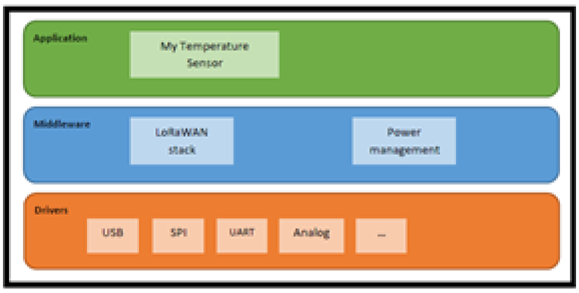

Fig. 5 LoRa device software architecture [1]

978-1-7281-8566-8/20/$31.00 ©2020 IEEE

The main tasks that considered in the implementations are:

1. Availability of the development kit for the selected architecture
2. Specifications for the design
3. Software development
4. Schematics
5. PCB design
6. Debugging the prototype v1
7. Manufacturing development and test software
8. Antenna design

Sensitivity of the antenna is critical point in LoRa technology and choosing the right antenna is one of the most important part in communication. 8.2cm antenna length is suitable for 868MHz band based on λ/4 and this calculation should be considered when working on mechanical design to eliminate close disturbances by placing the antenna correctly [1]. Three most common antenna architectures are:

1. OEM antenna that is tuned for specific frequencies of LoRa
2. dedicated antenna is the best way to get maximum results in sensitivity and transmission while this is the most expensive solution
3. Another option is a simple trace on the PCB. However, thickness and width of the copper must be considered carefully during the design.

The easiest solution is just a quarter wave antenna (λ/4). But repeatability during production and ensuring exact length in each product could be challenging. To calculate the hourly power consumption of the device, developer should know about these modes and estimate how much time will be spent in each of them. The worst-case scenario must be considered when doing calculations, for example Spreading Factor 12 is assumed during Tx. Expected battery lifetime and capacity of the battery can be estimated using this approach.

The easiest way to start developing device is a starter kit which includes development board and connectors for peripherals and sensors in addition to LoRa radio board. Starter kit enables us to quickly add connectivity to our device. LoRaWAN stack is already implemented in starter kit and a simple API is provided for sending messages. The most popular choice is using Arduino platform and shield to start developing.

The chosen hardware architecture mostly defines the software development phase. LoRa modem architecture allows the easiest software development in which all LoRaWAN stack is ready inside modem and communication management is done automatically. Only required step for the developer is to initialize the stack for sending and receiving the messages. In case the module architecture is used, LoRaWAN stack may be implemented by the manufacturer that fits the integrated MCU. The developer is provided with an API to setup and control the LoRaWAN stack using commands. If selected module does not have LoRaWAN stack or we use LoRa chipset integration, LoRaWAN stack should be implemented inside the software by the developer. To eliminate any deployment issues and make sure the device is working properly based on regulations, testing should be carried out for RF performance after the assembly.

The most important one of the tests is antenna matching and it can be done by using the following instructions:

1. One message is transmitted using three different channels at a selected output power. These three messages must be received by the receiver and measured RSSI/SNR should match the pre-calibrated limits.
2. One or more messages are received and RSSI is checked after reception by the tested device. The messages should be received successfully and the RSSI should match the pre-calibrated limits.

The tests above are used to make sure frequency tuning and antenna matching are successful.

As it is mentioned, LoRa device can be designed in a few different ways. LoRa module is the most user-friendly way to start building devices as the hardware part is already completed by the manufacturer. So, we should implement the software and LoRaWAN stack. In this project, an SX1276 module breakout board is used. Next step is selecting the antenna. The frequency plan for Azerbaijan can be found on the official regional parameters document in the website. It indicates that both 434MHZ and 868MHz frequency plans can be used. So SX1276 modules should be configured to either 434 or 868MHz frequency. In this project, 868MHz is used for communication. The next step is to select which MCU will be used to control the LoRa module. There are many options available, but the commonly used ones are STM32 family of microcontrollers. They provide many features compared to other microcontrollers while having less cost. However, the most important feature of STM32 family of microcontrollers is that they are much suitable for low power applications and low power series of STM32 consumes current up to nano amperes in deep sleep mode. So, STM32F103 development board is selected for this project. SX1276 module uses SPI connection and it should be connected to STM32f103 in the next step. To program the STM32 controller, it should be connected to a PC using an FTDI programmer. Although all the connections have been made, we need to know if the LoRa Node is transmitting any signal. So, we should use a RF Power Meter to see if the device is transmitting. The used RF Power Meter is a hand-held device in the range of 1MHz-8GHz with the power of -50dBm to 0dBm. After connecting it to the transmitter, it is possible to see the power of transmitted signals. Finally, The LoRa node is ready for programming by the PC (Figure 6).

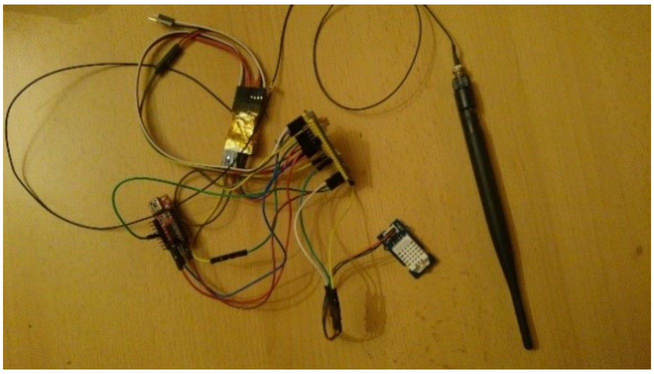

Fig. 6 The implemented LoRa node circuit

978-1-7281-8566-8/20/$31.00 ©2020 IEEE

A gateway is needed for a node to communicate with the server. A single channel LoRaWAN gateway based on ESP8266 and SX1276 has been implemented using an ESP8266 board, a STM32F103 board, a SX1276 module and an 868MHz antenna (figure 7). Using ESP8266 board is the cheapest way to create a LoRaWAN gateway. It can give any microcontroller access to Wi-Fi with its integrated TCP/IP protocol stack. This module has integrated low power 32-bit microprocessor and can be used as a main device to host the application or provide networking functionalities for other application.

As there are far more resources to program STM32F103 for building LoRaWAN node and gateway, it has been used Arduino IDE with STM32Duino package to program node and gateway. Software of the gateway is based on the libraries developed by Maarten Westenberg. This project is focused on the low-cost implementation; therefore, single channel gateway (one-channel/one-frequency) is used. As a result, only 868MHz signals will be received by the gateway after it is set up properly. As this project is set up for 868MHz frequency band, EU863-870 is defined in the programming file. CLASS "A" mode should be defined as the project is supposed to connect to The Things Network (TTN). SF is the next parameter to set. Because the gateway is single channel and single frequency, it can only listen to the messages in one SF only. So, the node should also be set for the same SF. There is a A_SERVER parameter that is used to activate Webserver. This Webserver can be opened in the browser to configure the gateway in runtime if the device is in the same Wi-Fi with gateway. Statistics of the last messages are also shown with the timestamps. At this point, gateway can be powered up and registered in TTN. 8 bytes of Gateway ID is required to register the gateway and should be entered in the Gateway EUI field in the TTN. After the registration is completed, gateway will be shown in the Gateways section of the TTN and if it is powered up, webpage will show its status as connected. Now the gateway is complete and waiting for the LoRaWAN packets to forward to TTN. Software of the node is based on the work of Thomas Telkamp and Matthijs Kooijman. The code contains IBM LMIC library which is modified to be used with SX1276 in the Arduino environment. The library supports:

- Sending uplink messages and managing duty cycling
- Encrypting and checking message integrity
- Receiving downlink packets
- Custom settings for frequencies and data rate
- Over-the-air activation (OTAA)

The code is working with ABP (Activation by Personalization) where session keys and device address should be predefined in the code. To register the device in TTN, it should be started by creating application in the portal. There is nothing special in this step and the application EUI is automatically generated by TTN. The next step is registering device in the application. Again, Device EUI and Application Key are automatically generated by TTN. Successful registration of the device will lead to a page where Device Address, Network Session Key and Application Session Key are generated.

Now the next step is copying these keys to the software of device, so TTN can recognize it. The most important point is that ABP activation should be selected under the Settings tab of the device in TTN. Otherwise, TTN will wait for OTAA Join Request which will not be sent by the node. Next important part of the code is where the transmission interval and the pin mapping are stated. Transmission interval can be limited due to regional limitations. And the pin mapping must be the same with physical pin connections (figure 8).

After all the settings are considered, node device can be powered up. However, as the gateway in this project is single channel, the node must use only one channel of the selected frequency. Selected channel can be found in the line 247 of the code. Figure 9 shows how to enable only channel zero and disable all the others in green square. To use all the channels, the commented code inside the red square should be used. Finally, the project is complete, and the node can be turned on after uploading the software. If the gateway is on and set up properly, TTN network will show green icon and elapsed time after the last message.

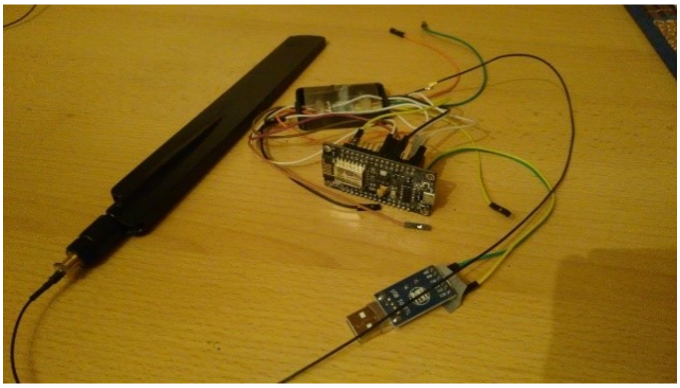

Fig. 7 The implemented Gateway circuit

```
// Schedule TX every this many seconds (might become longer due to duty
// cycle limitations).
const unsigned TX_INTERVAL = 5;

// Pin mapping
// Adapted for Feather M0 per p.10 of [feather]
const lmic_pinmap lmic_pins = {
.nss = PB0,
.rxtx = 0xFF,
.rst = 0xFF,
.dio = { PB1 , PB10, PB11 },
};
```

Fig. 8 Transmission interval and pin mapping section in code

```
    LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      // g-band

   int channel = 0;
   for(int i=0; i<9; i++) {
     if(i != channel) {
       LMIC_disableChannel(i);
     }
   }
//    LMIC_setupChannel(1, 868300000, DR_RANGE_MAP(DR_SF12, DR_SF7B), BAND_CENTI);      // g-band
//    LMIC_setupChannel(2, 868500000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      // g-band
//    LMIC_setupChannel(3, 867100000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      // g-band
//    LMIC_setupChannel(4, 867300000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      // g-band
//    LMIC_setupChannel(5, 867500000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      // g-band
//    LMIC_setupChannel(6, 867700000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      // g-band
//    LMIC_setupChannel(7, 867900000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      // g-band
//    LMIC_setupChannel(8, 868800000, DR_RANGE_MAP(DR_FSK,  DR_FSK),  BAND_MILLI);      // g2-band
    // TTN defines an additional channel at 869.525Mhz using SF9 for class B
    // devices' ping slots. LMIC does not have an easy way to define set this
    // frequency and support for class B is spotty and untested, so this
    // frequency is not configured here.
```

Fig. 9 Configuring channels of the node

978-1-7281-8566-8/20/$31.00 ©2020 IEEE

As the project is implemented, data should be collected to show the performance of the network. The data has been be collected for the approximately 100 meters range between node and the gateway. Airtime was very stable with 46.3ms. There was not any faulty data but sometimes a few packets were not delivered. Approximately one out of 20 packages were missed. The main problem was that downlink messages were never delivered. Therefore, this cheap network is not useful for consumer devices. There are strict timing constraints in LoRaWAN network and downlink messages should be delivered just in the correct receiving time window. Inaccuracies in the receive time window does not allow device to receive downlink messages. Therefore, OTAA activation does not work either as it requires server to respond to JOIN request. One option to increase possibility of device to receive messages is relaxing the timing constraint. Although it will increase the power consumption, some applications might sacrifice some power to receive messages. A few data points were missing randomly due to the inaccuracies of the devices.

Table III depicts measured current consumption of the node. Average battery life of the device can be estimated using these values as:

$$t_{lifetime} = \frac{C_{battery}}{I_{average}} \quad (6)$$

$$I_{avg} = \frac{I_1 \times t_1 + I_2 \times t_2 \dots I_n \times t_n}{t_{total}} \quad (7)$$

Assume a battery with 2400mAh capacity is used and end-node transmits one message, then sleeps half an hour. Equation 7 results in 0.07041 mA which is the average power consumption of the device per total cycle. The result of Equation 6 is 34086 hours (1420 days). Therefore, this end-node can work with a single 2400mAh battery for nearly 4 years while sending one byte of message to cloud every half an hour. Table IV shows specifications of the famous LPWA networks. The comparison results that LoRaWAN has average data rate and poor performance in urban settings. But it offers large coverage in remote and open places, has secure connection and the longest battery life. So, it is one of the best choices when it comes to industrial applications like SCADA.

TABLE III AVERAGE CURRENT CONSUMPTION OF THE LORA NODE

State	Duration (ms)	Current consumption (mA)
Wake up	168	23.5
Transmission	15	84.3
Wait 1st window	994	28
1st receive window	13.2	38.3
Wait 2nd window	50	28.2
2nd receive window	2	36.6
Sleep	Dependent on interval btw msgs	0.051

TABLE IV COMPARISON OF LONG-RANGE NETWORKS FOR [10]

Technology	Sigfox	LoRaWAN	LTE-M	NB-IoT
Max. Data Rate	600bps	50Kbps	1Mbps	106Kbps
Range	10km(urban), 50km(rural)	5km(urban), 15km(rural)	11km	15km
Security	No	Yes	Yes	Yes
Battery life	90 months	105 months	18 months	18 months

V. CONCLUSION

In this paper, it is discussed technical characteristics of LoRaWAN network and its possible applications in SCADA systems. Unlike traditional networks, LoRaWAN uses long range star topology and it allows network to have longer range while maintaining minimum power consumption. Effects of network architecture on the battery consumption and communication range is elaborated. Moreover, LoRaWAN network offers three different classes for the end-nodes to serve the requirements of different applications. As this paper is focused on implementing of LoRaWAN, network elements are discussed one-by-one. Embedded software is inherent part of the project, so effects of the architecture on the software development is detailed while giving information about some best practices. Final part of paper is devoted to creating a simple LoRaWAN network by explaining the selection of devices and tools. Individual parts of the circuit for LoRaWAN node and gateway are discussed and the most important parts of the software is clarified. Currently, the well-known cloud-server for "connected things" is The Things Network (TTN), so the tests are carried out using TTN. Results showed that there are two main issues because of hardware inaccuracies. Downlink messages are not delivered as LoRaWAN has strict timing rules which the hardware could not handle. Therefore, OTAA activation did not work but the network security is decreasing significantly without it. This problem can be tackled by increasing the receive window of end-device. Unfortunately, longer receive windows decrease the battery lifetime. Next problem is that a few uplink messages are not delivered because of the inaccuracies in host microcontrollers. Even in this case, more than 90% of the messages are delivered to the server. Finally, measured current consumption of the end-node is used to estimate its lifetime. Results showed that a device which sends one byte of message every half an hour can work on single 2400mAh battery for almost 4 years.

References

[1] N. Ducrot, D. Ray, A. Saadani, O. Hersent, G. Pop and G. Remond, "LoRa Device Developer Guide," Orange Connected Objects & Partnerships, Actility, 2016.

[2] L. Alliance, "A technical overview of LoRa and LoRaWAN," 2015.

[3] Semtech, WIRELESS & SENSING PRODUCTS, "SX1276/77/78/79 - 137 MHz to 1020 MHz Low Power Long Range Transceiver," 2019.

[4] K. Mekki, E. Bajic, F. Chaxel and F. Meyer, "A comparative study of LPWAN technologies for large-scale IoT deployment," ICT Express, 2019.

[5] B. Foubert and N. Mitton , "Long-Range Wireless Radio Technologies: A Survey," future internet, 14 January 2020.

[6] P. San Cheong, J. Bergs, C. Hawinkel and J. Famaey, "Comparison of LoRaWAN classes and their power consumption. Communications and Vehicular Technology (SCVT)," IEEE Symposium on, pp. 1-6, 2017.

[7] M. A. Ertürk, M. A. Aydın, M. T. Büyükakkaşlar and H. Evirgen, "A Survey on LoRaWAN Architecture, Protocol and," future internet, 17 October 2019.

[8] BENTEK SYSTEMS, "BENTEK SYSTEMS," SCADALİNK, [Online]. Available: https://www.scadalink.com/support/knowledge-base/wireless-scada/. [Accessed 3 May 2020].

[9] Elprocus, "Elprocus," [Online]. Available: https://www.elprocus.com/scada-systems-work/. [Accessed 5 May 2020].

978-1-7281-8566-8/20/$31.00 ©2020 IEEE