XVI Jornadas de Ingeniería Telemática
**JITEL 2023**
La Salle -- Universitat Ramon Llull

*Actas de las XVI Jornadas de Ingeniería Telemática (JITEL 2023), Barcelona (España), 8-10 de noviembre de 2023.*

# Converting a Weather Station into a LoRaWAN-enabled Device

Jorge Navarro-Ortiz, Natalia Chinchilla-Romero, Felix Delgado-Ferro,
Juan Jose Ramos-Munoz, Juan M. Lopez-Soler
Departamento de Teoría de la Señal, Telemática y Comunicaciones,
Universidad de Granada
C/ Periodista Daniel Saucedo Aranda, s/n
{jorgenavarro, nataliachr, felixdelgado, jjramos, juanma}@ugr.es

**This paper presents a method for transforming an affordable device, such as a weather station, into a LoRaWAN-enabled device. For this purpose, a low-cost node with an ESP32 microcontroller and a SX1276 LoRa modem was employed which, in addition to provide LoRaWAN connectivity, is able to capture and decode the transmissions from the weather station in the 868 MHz band. A proof of concept was carried out to verify the success of this approach.**

***Keywords*** **– LoRaWAN, weather station, RTL-433, IoT**

## I. Introduction

According to a report by Internet of Things (IoT) Analytics, the number of IoT connections is projected to rise from 7 billion in 2018 to 22 billion by 2025 [1]. In 2019, Low Power Wide Area Networks (LPWANs) accounted for 231 million of these IoT connections [2]. LPWANs are wireless networks designed specifically for IoT, enabling the connection of numerous low-cost devices. They offer long-range communication capabilities while transmitting small amounts of traffic and consuming minimal energy.

Currently, there are four dominant LPWAN technologies available in the market: LoRaWAN, Sigfox, NB-IoT, and LTE-M. Among these, LoRaWAN stands out as a highly promising solution that meets the requirements for long-range and low-power communication. It supports bit rates ranging from 250 bps to 11 kbps within the non-licensed ISM band. LoRaWAN follows an open standard, defining the medium access control (MAC) layer and network topology. Moreover, it has been deployed in 157 countries worldwide and is supported by the LoRa Alliance, a non-profit association consisting of over 500 members [3]. LoRaWAN is intended for applications that send few messages per day, e.g., logistics, smart farming, environmental monitoring, smart cities, smart grids, etc.

To get a device to generate environmental data and send it through LoRaWAN we have two possibilities. On the one hand, you could build a device, e.g., based on Arduino, that includes different types of sensors (e.g., temperature, humidity, wind, precipitation, light, etc.) and uses LoRaWAN natively. On the other hand, you could use a low-cost device that includes all these sensors and implement a gateway between the original device's wireless technology and LoRaWAN. This last approach is the one used in this paper, which can even be cheaper (e.g. the Bresser weather station [4] is available for under 100 euros, whereas assembling a weather station using Arduino [5] might incur costs exceeding 150 euros (excluding expenses tied to solar energy components)).

Thus, the objective of this article is to enable commercial products to function as LoRaWAN devices inexpensively, thereby achieving the inherent advantages of this technology (wide coverage with reduced energy cost and security in the LoRaWAN part). This will allow developers to generate LoRaWAN traffic with real data, which can be useful in research or for creating demonstrators.

The rest of the paper is organized as follows. Section II makes a brief description of LoRaWAN. The characteristics of the weather station are described in Section III, while Section IV explains how to turn it into a LoRaWAN device. Section V summarizes the proof of concept carried out, and Section VI concludes the paper.

## II. LoRaWAN Overview

LoRa, the physical layer of a LoRaWAN network, is a modulation technique developed by Cycleo in 2009 and later acquired by Semtech in 2012. It operates using chirp spread spectrum modulation. One of the advantages of LoRa modulation is its receiver design, which has low complexity due to the timing and frequency offsets between the transmitter and receiver being equivalent. In this modulation scheme, the data signal is modulated onto a chirp signal that varies its frequency over time. The data

This work is licensed under a Creative Commons 4.0 International License (CC BY-NC-ND 4.0)

rate can vary based on the spectral bandwidth (BW, typically 125 KHz), the spreading factor (SF) employed (ranging from 7 to 12), and the coding rate (ranging from 1 to 4) following Eq. (1).

$$R_b = SF \times \left(\frac{BW}{2^{SF}}\right) \times \left(\frac{4}{4+CR}\right) \quad (1)$$

where the spreading factor (*SF*) is the first term in the equation, the symbol rate (*Rs*) is represented by the second term (symbols per second), and the third term is dependent on the coding rate (*CR*), which can vary between 1 and 4. Assuming a fixed bandwidth and coding rate, it is observed that as the spreading factor increases, the data rate decreases and the Time over Air (*ToA*) becomes higher.

The LoRaWAN open standard defines the network architecture and depicts the communication protocol that enables connectivity among all entities within the network. Fig. 1 illustrates a typical LoRaWAN network configuration.

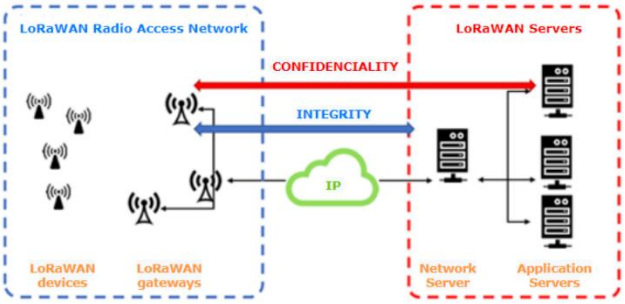



Fig. 1. LoRaWAN network architecture.

In this network configuration, a star-of-stars topology is employed. It consists of one or more gateways, also known as concentrators, which serve as intermediaries for relaying downlink and uplink traffic between a diverse range of end-nodes. These end-nodes can either be mobile or fixed at specific locations. On one hand, the gateways establish connections with the end-nodes using single-hop LoRa or FSK connections. On the other hand, the gateways utilize standard IP connections to communicate with the central network server. The network server assumes the responsibility of managing the traffic between each end-device and its associated application server. In addition, LoRaWAN provides integrity using a Message Integrity Code (*MIC*) between the end-device and the network server, using a Network Session Key (*NwkSKey*). The transmission between end-devices and the network server is encrypted using an Application Session Key (*AppSKey*). This guarantees the security of communications and allows the application owner to contract the usage of gateways and network server from an infrastructure operator, i.e., the application operator and the infrastructure operator can be different.

## III. Weather Station Description

We own two low-cost weather stations, a Bresser professional 7-in-1 Wi-Fi Weather Station with Light Intensity and UV Measurement Function, located at the roof of the Higher Technical School of Computer and Telecommunication Engineering (ETSIIT) of the University of Granada (see Fig. 2), and a Bresser 7-in-1 ClimateConnect Tuya Smart Home Weather Station, located at the roof of the house of one member of the research team (see Fig. 3). Both weather stations include the same sensors: temperature, humidity, wind speed, wind direction, rainfall, UV and light intensity. The first weather station can be found for around 280 euros, and the second for around 100 euros (depending on promotions).

Both weather stations have a similar behavior. Both devices measure data, which are encoded with Pulse Code Modulation (PCM, with a pulse width of 124 μs) and then transmitted using the Frequency Shift Keying (FSK) modulation on the 868 MHz band. Data is transmitted every 12 seconds for UV, light intensity, wind speed and wind direction, and every 24 seconds for temperature, humidity and rain data.

Once the measurements have been sent to the display over 868 MHz, the display is able to post-process these data or forward them to different web weather services such as Weather Underground, Weathercloud or Awekas. This allows us to have beautiful dashboards with the current stats and to aggregate data or compute more complex calculations, e.g., for weather forecasting. These



Fig. 2. Weather station (and display) at the roof ot the ETSIIT.



Fig. 3. Weather station (and display) at one house.

This work is licensed under a Creative Commons 4.0 International License (CC BY-NC-ND 4.0)

features depend on the display and not on the weather station so, in our case, the web weather services are only available for the first weather station.

## IV. Becoming a LoRaWAN-Enabled Device

Converting a Bresser 7-in-1 weather station to a LoRaWAN-enabled device is done in two steps. First, we have to capture and decode the data sent over the native wireless technology. Second, we have to forward this data through a LoRaWAN device.

Capturing and decoding is based on code from RTL-433 [6], adapted to the ESP32 microcontroller [7]. RTL-433 is software that allows capturing and decoding many devices using MC (Manchester Code), PCM, RZ (Return to Zero), PPM (Pulse Position Modulation), PWM (Pulse Width Modulation), DMC (Differential Manchester Code), etc., and transmission using FSK, OOK (On-Off Keying) and ASK (Amplitude Shift Keying) modulations. RTL-433 initially supported capture by RTL-SDR devices [8] and decoding by devices that used the 433 MHz band, hence its name, but now it also supports the usual bands (315, 345, and 915 MHz).

Since we want the final prototype to be low-cost, we have opted for a device based on ESP32 (which can be easily programmed using Arduino, LUA, MicroPython, etc.) that includes a LoRa chip. In particular, due to its availability, we have used a Heltec Wireless Stick mote [9], which costs less than 20 euros and has an ESP32 together with a SX1276 LoRa chip [10]. In addition to being a LoRa modem, SX1276 supports FSK, GFSK, MSK, GMSK and OOK modulations. This will allow us to capture and, with the appropriate software, decode 868 MHz transmissions from devices such as the weather stations we are using in this paper before sending the data over LoRaWAN with the same chip.

The code was developed using Arduino and allows the ESP32 to choose the period to read the data and send it via LoRaWAN. After each transmission, the device enters deep-sleep mode the rest of the time. This has two advantages. On the one hand, its energy consumption is reduced. On the other hand, every time the device wakes up it performs a reset, which improves its stability. In the latter sense, watch dog timers have also been used to eliminate the possibility of the microcontroller getting frozen. Fig. 4 shows the pseudocode for the ESP32 microcontroller.

In addition, since web weather services are only available for the weather stations with expensive displays, we have implemented an application that reads data from the LoRaWAN application server using its MQTT API and forwards them to Weather Underground, Weathercloud or Awekas using their corresponding APIs.

It should be noted that security is covered in the LoRaWAN part, but not around the weather station since it transmits unencrypted data over 868 MHz.

## V. Proof-of-Concept

Our proof of concept uses a Pycom PyGate [11] as the LoRaWAN gateway and the ChirpStack framework [12] for the network and application servers. ChirpStack has different APIs to manage the different entities, including the use of MQTT (Message Queuing Telemetry Transport)

```
setup()
    lastTimeWDTReset = currentTime
    lastTransmission = -1
    dataAvailable = False
    sleepReq = True

    AddWatchdogTimer()

       if firstBoot == True:
          ResetStats()
       InitializeLoRaWAN()
       InitializeSensors()

loop()
    ResetWatchdogTimer()
    LoRaWAN.loop()
    Sensors.loop()
    checkGoingToDeepSleep()

LoRaWAN::loop()
    If dataAvailable
        LoRaWAN.sendWeatherData()
    If LoRaWAN.txComplete()
        dataAvailable = False
        sleepReq = True

Sensors::loop()
    If time > (lastTransmission + txPeriod)
        dataAvailable = ReadWeatherData()

checkGoingToDeepSleep()
    If sleepReq
        timeToSleep = computeTimeToNextTx()
        ESP.deepSleep(timeToSleep)
```

Fig 4. Pseudocode for the ESP32 microcontroller.

to query the received messages. We have also tested the usage of The Things Network [13] as LoRaWAN platform, including both the network and application servers (see the steps in [14]). The layout of the proof of concept can be observed in Fig. 5.

For the visualization of the received data, we selected the Grafana tool. For data storage, we used the InfluxDB database, which is specialized in time series. We employed Telegraf as an MQTT-InfluxDB gateway, which acts as an MQTT subscriber and stores the received data in InfluxDB. An example of this visualization can be found in Fig. 6.

The repositories with our implementation are publicly available [15][16][17].

## VI. Conclusions

This paper presents a method for transforming an affordable device, such as a weather station, into a LoRaWAN-enabled device. To achieve this, we utilized the Heltec Wireless Stick, which costs less than 20 euros and incorporates an ESP32 microcontroller and a LoRa SX1276 modem. The LoRa chip's built-in FSK demodulation and other modulations eliminate the need for additional hardware to capture and decode the weather station's messages. This approach is valid for other devices transmitting on the 868 MHz band.

Furthermore, we conducted a proof of concept that involved a LoRaWAN test network comprising a PyGate gateway and the ChirpStack platform. Data visualization was achieved using Grafana, and we integrated the capability to send this data to web weather services like WeatherUnderGround, WeatherCloud, and AWEKAS.

This work is licensed under a Creative Commons 4.0 International License (CC BY-NC-ND 4.0)

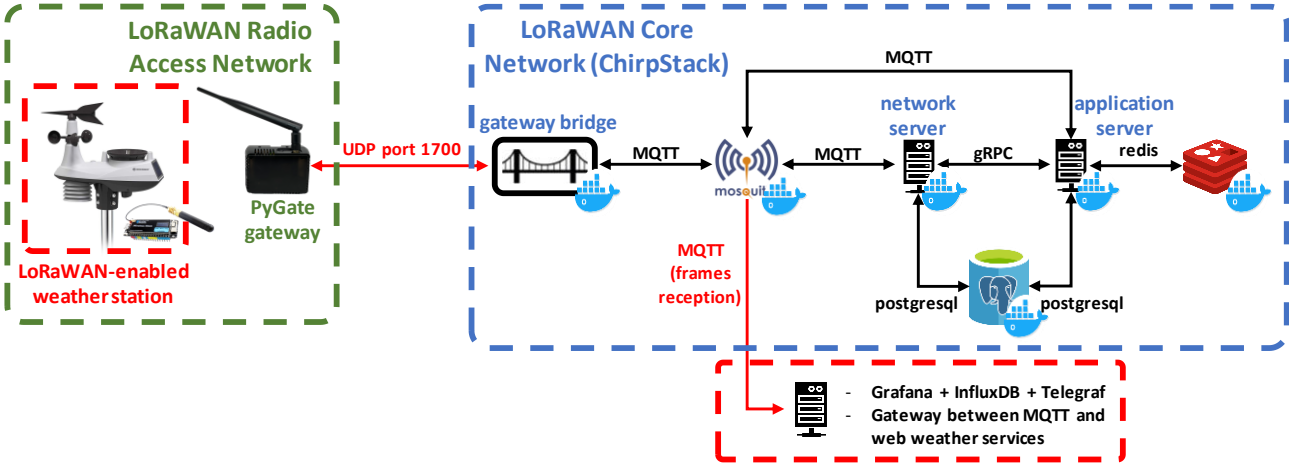



Fig. 5. Proof of concept.

## ACKNOWLEDGMENTS

This research was partially funded by the Andalusian Knowledge Agency (project B-TIC-568-UGR20), the Spanish Ministry of Science and Innovation (projects PID2019-108713RB-C53 and PID2022-137329OB-C43), the Spanish Ministry of Economic Affairs and Digital Transformation (project TSI-063000-2021-28) and the Spanish Ministry of Universities (FPU Grant Number: 20/02621).




Fig. 6. Grafana dashboard with some stats from the weather station.

## REFERENCES

[1] IoT Analytics. LPWAN Market Report 2018–2023. Available online: https://iot-analytics.com/lpwan-market-report-2018-2023-new-report/ (accessed on 30th May 2023).

[2] IoT Analytics. 5 Things to Know about the LPWAN Market in 2020. Available online: https://iot-analytics.com/5-things-to-know-about-the-lpwan-market-in-2020/ (accessed on 30th May 2023).

[3] LoRa Alliance. Available online: https://lora-alliance.org/ (accessed on 30th May 2023).

[4] BRESSER 7-in-1 ClimateConnect Tuya Smart Home Weather Station. Available online: https://www.bresser.de/en/Weather-Time/BRESSER-7-in-1-ClimateConnect-Tuya-Smart-Home-Weather-Station.html (accessed on 30th May 2023).

[5] Solar Powered WiFi Weather Station V3.0. Available online: https://www.instructables.com/Solar-Powered-WiFi-Weather-Station-V30/ (accessed on 30th May 2023).

[4] RTL-433 generic data receiver. Available online: https://github.com/merbanan/rtl_433 (accessed on 30th May 2023).

[5] ESP32, a feature-rich MCU with integrated Wi-Fi and Bluetooth connectivity for a wide range of applications. Available online: https://www.espressif.com/en/products/socs/esp32 (accessed on 30th May 2023).

[6] RTL-SDR and software defined radio news and projects. Available online: https://www.rtl-sdr.com/ (accessed on 30th May 2023).

[7] Heltec Wireless Stick. Available online: https://heltec.org/project/wireless-stick/ (accessed on 30th May 2023).

[8] Semtech's SX1276. Available online: https://www.semtech.com/products/wireless-rf/lora-connect/sx1276 (accessed on 30th May 2023).

[9] Pycom's PyGate. Available online: https://docs.pycom.io/tutorials/expansionboards/pygate/ (accessed on 30th May 2023).

[10] ChirpStack, open-source LoRaWAN Network Server. Available online: https://www.chirpstack.io/ (accessed on 30th May 2023).

[11] The Things Network. Available online: https://www.thethingsnetwork.org/ (accessed on 30th May 2023).

[12] LoRaWAN testbed repository. Available online: https://github.com/jorgenavarroortiz/testbed-LoRaWAN (accessed on 30th May 2023).

[13] BresserWeatherSensorReceiver with Bresser 7-in-1 decoder. Available online: https://github.com/jorgenavarroortiz/BresserWeatherStationLoRaWAN (accessed on 30th May 2023).

[14] Platform to show weather station data in Grafana. Available online: https://github.com/jorgenavarroortiz/weatherstation-mqtt-grafana (accessed on 30th May 2023).

[15] LoRaWAN network using Pycom gateway (PyGate) and ChirpStack platform. Available online: https://github.com/jorgenavarroortiz/testbed-LoRaWAN (accessed on 30th May 2023).

This work is licensed under a Creative Commons 4.0 International License (CC BY-NC-ND 4.0)