



**Implementación de una red LPWAN para el control de consumo de agua potable utilizando tecnología LoRa.**

Borja Pozo, Alberto Gerardo

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica y Telecomunicaciones

Ing. Montoya Lara, Luis Hernán Mgs.

15 de septiembre del 2021

15/9/21 15:59

tesis\_antiplagio

## Informe de originalidad

---

NOMBRE DEL CURSO  
RESULTADOS TESIS

NOMBRE DEL ALUMNO  
ALBERTO GERARDO BORJA POZO

NOMBRE DEL ARCHIVO  
ALBERTO GERARDO BORJA POZO - tesis\_Alberto

SE HA CREADO EL INFORME  
15 sept 2021

### Resumen

Fragmentos marcados	7	4 %
Fragmentos citados o entrecorridos	8	3 %

#### Coincidencias de la Web

upiles	6	3 %
udistrital.edu.co	4	2 %
ucuenca.edu.ec	2	2 %
semak.com.ar	3	1 %

1 de 15 fragmentos

Fragmento del alumno **MARCADO**

LoRa emplea una velocidad de datos (data rate) variable, manejando factores de dispersión (spreading factor SF) ortogonales, permitiendo al diseñador del sistema intercambiar la velocidad de datos por...

#### Mejor coincidencia en la Web

LoRa emplea una velocidad de datos (data rate) variable, manejando factores de dispersión (spreading factor SF) ortogonales, permitiendo al diseñador del sistema intercambiar la velocidad de datos por...

Evaluación de pasarela LoRa/LoRaWAN en entornos urbanos <https://riunet.uv.es/bitstream/handle/10251/109791/Moya%20-%20Evaluaci%C3%B3n%20de%20pasarela%20LoRa/LoRaWAN%20en%20entornos%20urbanos.pdf?sequence=1&isAllowed=y>

2 de 15 fragmentos

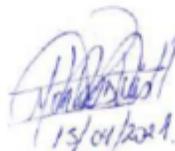
Fragmento del alumno **CREADO**

La naturaleza de banda ancha permite compensar una mejor relación señal/ruido (SNR) lo que permite a LoRa demodular las señales incluso cuando está a 19,5 dB por debajo del piso de

#### Mejor coincidencia en la Web

1. La naturaleza de banda ancha permite compensar una mejor relación señal/ruido (SNR) lo que permite a LoRa demodular las señales incluso cuando está a 19,5 dB por debajo del piso de ruido. (2)(5)(8)

Evaluación de pasarela LoRa/LoRaWAN en entornos urbanos <https://riunet.uv.es/bitstream/handle/10251/109791/Moya%20-%20Evaluaci%C3%B3n%20de%20pasarela%20LoRa/LoRaWAN%20en%20entornos%20urbanos.pdf?sequence=1&isAllowed=y>



Ing. MONTOYA LARA, LUIS HERNAN  
DIRECTOR

1/4



**DEPARTAMENTO DE ELÉCTRICA, ELÉTRÓNICA Y  
TELECOMUNICACIONES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

### **CERTIFICACIÓN**

Certifico que el trabajo de titulación, **“Implementación de una red LPWAN para el control de consumo de agua potable utilizando tecnología LoRa”** fue realizado por el señor **Borja Pozo, Alberto Gerardo** el cual ha sido revisado y analizado en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 15 de septiembre de 2021

Firma:

**Montoya Lara, Luis Hernán**

C. C. 1715480412



**DEPARTAMENTO DE ELÉCTRICA, ELÉTRÓNICA Y  
TELECOMUNICACIONES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

**RESPONSABILIDAD DE AUTORÍA**

Yo, **Borja Pozo, Alberto Gerardo**, con cédula de ciudadanía n° 020189651-1, declaro que el contenido, ideas y criterios del trabajo de titulación: *“Implementación de una red LPWAN para el control de consumo de agua potable utilizando tecnología LoRa”* es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

**Sangolquí, 15 de septiembre de 2021**

Firma

**Borja Pozo, Alberto Gerardo**

C.C.: 020189651-1



**DEPARTAMENTO DE ELÉCTRICA, ELÉCTRÓNICA Y  
TELECOMUNICACIONES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

**AUTORIZACIÓN DE PUBLICACIÓN**

Yo, **Borja Pozo, Alberto Gerardo**, con cédula de ciudadanía n° 020189651-1, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Título: "Implementación de una red LPWAN para el control de consumo de agua potable utilizando tecnología LoRa"** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

**Sangolquí, 15/09/2021**

Firma

**Borja Pozo, Alberto Gerardo**

C.C.: 0201896511

## **DEDICATORIA**

Dedico de manera especial mi tesis a mis padres, por haberme apoyado incondicionalmente durante toda mi vida. Su paciencia, su amor y sacrificio son la fortaleza que tengo para salir adelante y alcanzar mis metas.

**Alberto Gerardo Borja Pozo**

## **AGRADECIMIENTO**

Primero agradezco a Dios por la salud, la vida y la oportunidad de poder alcanzar este objetivo de ser profesional. Agradezco a mis padres por todo su apoyo en las decisiones que tomo en mi vida, también un agradecimiento especial a mi director de tesis Ing. Luis Montoya por la paciencia, ayuda y dedicación para poder culminar este proyecto. A mi familia porque siempre están presentes en todos los momentos de mi vida y finalmente a todas las personas que hicieron posible que pueda alcanzar este logro tan importante para mí.

**Alberto Gerardo Borja Pozo**

## INDICE DE CONTENIDOS

Reporte de verificación de similitud .....	2
Certificado del director del proyecto .....	3
Responsabilidad de Autoría .....	4
Autorización de Publicación .....	5
Dedicatoria .....	6
Agradecimiento .....	7
Índice de tablas .....	11
Índice figuras .....	12
Resumen.....	15
Abstract.....	16
<b>Capítulo I</b> .....	<b>17</b>
<b>Introducción</b> .....	<b>17</b>
<i>Antecedentes</i> .....	17
<i>Justificación e Importancia</i> .....	17
<i>Alcance del Proyecto</i> .....	18
<i>Objetivos</i> .....	19
<b>Capítulo II</b> .....	<b>21</b>
<b><i>Marco Teórico</i></b> .....	<b>21</b>
<b><i>Redes de largo alcance y bajo consumo de potencia</i></b> .....	<b>21</b>
<b><i>LoRa</i></b> .....	<b>23</b>
<i>Características</i> .....	23
<i>Modulación LoRa</i> .....	24
<b><i>Protocolo LoRaWAN</i></b> .....	<b>25</b>
<i>Canales y velocidades de transmisión</i> .....	27
<b><i>Clasificación de los dispositivos LoRa</i></b> .....	<b>28</b>
<i>Clase A</i> .....	28

Clase B.....	29
Clase C.....	30
<b>Internet de las cosas</b> .....	30
<i>Características del IoT</i> .....	31
<b>Servidores de IoT</b> .....	32
<b>Dispositivo de FrontEnd</b> .....	33
<b>Capítulo III</b> .....	34
<b>Descripción general del proyecto</b> .....	34
<i>Bloque de control</i> .....	35
<i>Bloque de TX/RX</i> .....	35
<i>Bloque de visualización</i> .....	36
<b>Módulos que componen el proyecto</b> .....	38
<b>Análisis de recursos</b> .....	42
<b>Ventajas del producto</b> .....	45
<b>Capítulo IV</b> .....	47
<b>Proceso de ensamblado del hardware</b> .....	47
<b>Proceso de ensamblado del software</b> .....	50
<b>Proceso de calibración</b> .....	58
<i>Pasos para calibrar el prototipo</i> .....	61
<b>Cálculo de la duración de la batería</b> .....	65
<b>Implementación</b> .....	67
<i>Ensamblado del sistema</i> .....	67
<i>Configuración del Gateway LoRa</i> .....	72
<i>Procedimiento para registrar el Gateway LoRa en TTN</i> .....	76
<i>Creación de una aplicación en TTN</i> .....	78
<i>Personalización del formato del mensaje recibido en TTN</i> .....	80
<i>Procedimiento para integrar Ubidots con TTN</i> .....	80
<i>Creación de la aplicación en Ubidots</i> .....	81

<i>Pruebas de funcionamiento</i> .....	85
<b>Capítulo V</b> .....	88
<i>Análisis de Resultados</i> .....	88
<i>Análisis por horas del consumo de agua del día 17 de febrero del 2021</i> .....	88
<i>Análisis del consumo por día</i> .....	89
<i>Cobertura del prototipo</i> .....	90
<b>Capítulo VI</b> .....	92
<i>Conclusiones</i> .....	92
<i>Trabajos Futuros</i> .....	93

**ÍNDICE DE TABLAS**

Tabla 1 <i>Canales para la banda de frecuencia de 868 MHz y 433 MHz – LoRa WAN</i> .....	28
Tabla 2 <i>Descripción técnica de los elementos utilizados en el proyecto</i> .....	39
Tabla 3 <i>Elementos utilizados en la construcción del sistema</i> .....	42
Tabla 4 <i>Precio para la producción en masa del sistema</i> .....	44
Tabla 5 <i>Cantidad de pulsos obtenidas por el prototipo en 20 L de agua</i> .....	62
Tabla 6 <i>Consumo diario de agua en dos semanas de monitoreo</i> .....	89

## ÍNDICE FIGURAS

Figura 1 <i>Características de LoraWAN de pendiente de la Región</i> .....	26
Figura 2 <i>Acceso al medio en clase A de LoRaWAN</i> .....	29
Figura 3 <i>Acceso al medio en la clase B LoRaWAN</i> .....	30
Figura 4 <i>Acceso al medio en la clase C LoRaWAN</i> .....	30
Figura 5 <i>Servidor para aplicaciones IoT de uso libre</i> .....	32
Figura 6 <i>Gateways LoRa conectados a TTN a nivel mundial</i> .....	32
Figura 7 <i>Diagrama general de bloques del funcionamiento del sistema</i> .....	34
Figura 8 <i>Esquema electrónico de conexión para los bloques 1 y 2</i> .....	36
Figura 9 <i>Diagrama de las etapas de funcionamiento del sistema</i> .....	37
Figura 10 <i>Esquema del circuito en bloques simplificado</i> .....	47
Figura 11 <i>Headers hembra</i> .....	48
Figura 12 <i>Diseño electrónico de conexión del sistema</i> .....	48
Figura 13 <i>Ubicación de los headers hembra y cables en la baquelita</i> .....	49
Figura 14 <i>Librerías usadas en Arduino</i> .....	50
Figura 15 <i>Instanciación de objetos</i> .....	51
Figura 16 <i>Constantes y variables empleadas en la programación</i> .....	51
Figura 17 <i>Declaración del payload y credenciales de TTN</i> .....	51
Figura 18 <i>Intervalo de tiempo de la transmisión Lora</i> .....	52
Figura 19 <i>Pines digitales usados por el módulo Lora</i> .....	52
Figura 20 <i>Función que realiza el envío de mensajes</i> .....	52
Figura 21 <i>Extracto de la función onEvent</i> .....	53
Figura 22 <i>Evento EV_TXCOMPLETE</i> .....	54
Figura 23 <i>Función flujo</i> .....	54
Figura 24 <i>Extracto de la función setup()</i> .....	54
Figura 25 <i>Extracto 1 de la función loop()</i> .....	55
Figura 26 <i>Extracto 2 de la función loop()</i> .....	56
Figura 27 <i>Función LCD_2004</i> .....	57
Figura 28 <i>Función envio</i> .....	57
Figura 29 <i>Laboratorio de la empresa Hidrocentro</i> .....	58
Figura 30 <i>Equipo que controla la cantidad de litros de agua que se suministra al prototipo</i> .....	59
Figura 31 <i>Tanque de agua de 20L</i> .....	59
Figura 32 <i>Barómetro para medir la presión del agua</i> .....	60
Figura 33 <i>Prototipo conectado a la tubería del laboratorio</i> .....	61

Figura 34 <i>Medida en metros cúbico obtenida por el prototipo cuando circularon 20 L de agua</i> .....	63
Figura 35 <i>Sensor óptico en 20 litros de agua</i> .....	63
Figura 36 <i>Medidor patrón en 20L de agua</i> .....	63
Figura 37 <i>Medida en metros cúbico obtenida por el prototipo cuando circularon 100 L de agua</i> .....	64
Figura 38 <i>Sensor óptico en 100 litros de agua</i> .....	65
Figura 39 <i>Valor de la corriente de carga del dispositivo</i> .....	66
Figura 40 <i>Distribución de elementos del sistema en la caja</i> .....	67
Figura 41 <i>Instalación de tubería dentro de la caja del proyecto</i> .....	67
Figura 42 <i>LCD 20x04 colocado en la caja de proyectos</i> .....	68
Figura 43 <i>Distribución de los elementos electrónicos en la caja de proyectos</i> .....	69
Figura 44 <i>Prototipo instalado en la tubería de agua potable del domicilio</i> .....	69
Figura 45 <i>Conexión e instalación inalámbrica del gateway LoRa</i> .....	70
Figura 46 <i>Instalación final del sistema de control de consumo de agua potable</i> .....	71
Figura 47 <i>Dirección IP del Dragino (Gateway Lora)</i> .....	72
Figura 48 <i>Interfaz de usuario del Dragino</i> .....	72
Figura 49 <i>Pestaña “Network” opción “Wireless”</i> .....	73
Figura 50 <i>Proceso de “Scan” para identificar las redes Wireless cercanas</i> .....	73
Figura 51 <i>Contraseña de la red Wireless</i> .....	74
Figura 52 <i>Configuración del servidor de red</i> .....	74
Figura 53 <i>Configuración del canal de recepción</i> .....	75
Figura 54 <i>Configuración del canal de transmisión</i> .....	75
Figura 55 <i>Consola de TTN</i> .....	76
Figura 56 <i>Registro del Gateway Lora en TTN</i> .....	76
Figura 57 <i>Campos necesarios para registrar el Gateway Lora en TTN</i> .....	77
Figura 58 <i>Estado de conexión del Gateway Lora</i> .....	77
Figura 59 <i>Creación de una aplicación en TTN</i> .....	78
Figura 60 <i>Registro del nodo final dentro de la aplicación creada</i> .....	78
Figura 61 <i>Método de activación ABP del nodo final</i> .....	79
Figura 62 <i>Función que decodifica los mensajes recibidos en TTN</i> .....	80
Figura 63 <i>Integración de Ubidos con TTN</i> .....	80
Figura 64 <i>Widgets disponibles en Ubidos</i> .....	81
Figura 65 <i>Agregación de widgets en Ubidos</i> .....	82
Figura 66 <i>Variable On/Off</i> .....	82
Figura 67 <i>Programar la ejecución del evento cuando cambie el estado de la variable On/Off</i> .....	83

Figura 68 <i>Acciones que se realizan cuando se ejecuta el evento</i> .....	84
Figura 69 <i>Ventana con las opciones de los widgets y variables asignadas</i> .....	84
Figura 70 <i>Pestaña Device de Ubidots</i> .....	85
Figura 71 <i>Listado de dispositivos registrados en Ubidots</i> .....	86
Figura 72 <i>VARIABLES empleadas en Ubidots</i> .....	86
Figura 73 <i>Ventana de información para envío de archivos y correo electrónico</i> .....	87
Figura 74 <i>Gráfica del consumo acumulado y consumo por minuto</i> .....	88
Figura 75 <i>Consumo acumulado y consumo diarios de agua potable en el domicilio</i> .....	90
Figura 76 <i>Área de cobertura del prototipo utilizando la tecnología lora</i> .....	91

## RESUMEN

Las redes LPWAN en la actualidad han sido de gran ayuda en el diseño de soluciones y aplicaciones para el Internet de las Cosas y más aún en tiempos de pandemia y distanciamiento social. En marzo del 2020 el estado ecuatoriano decretó un estado de excepción y una cuarentena como medida para contrarrestar la propagación del Covid-19. Durante la cuarentena, se prohibió la libre movilización de las personas y se impulsó el teletrabajo, pero ocasionó la paralización de actividades como la lectura de los medidores de agua y electricidad.

Este proyecto consiste en el desarrollo teórico-práctico de una solución para el análisis del desempeño de una red LPWAN con tecnología LoRa mediante la implementación de un medidor inteligente (smart meter) que permita realizar el control y cálculo de consumo de agua potable de forma remota en la ciudad de Quito, con el fin de conocer las variables y parámetros óptimos para la implementación de una red LPWAN basada en LoRa. Del mismo modo permitir la visualización en tiempo real del consumo de agua a través de una plataforma web permitiendo el ahorro de recursos humanos para realizar la medición del consumo de agua.

El control de consumo y la visualización de la información en la plataforma web permitirán a los administradores tomar decisiones en lo que respecta al cierre y apertura de válvulas de suministro de agua potable, es decir, permitirá el corte del agua potable de forma remota. Para ello se utilizará *Ubidots*, una plataforma IoT que ofrece almacenamiento de datos y la integración de widgets de acuerdo a las variables recibidas.

### PALABRAS CLAVES:

- **MEDIDOR INTELIGENTE**
- **LORA**
- **LORAWAN**
- **LPWAN**
- **UBIDOTS**

### **ABSTRACT**

LPWAN networks today have been of great help in designing solutions and applications for the Internet of Things and even more so in times of pandemic and social distancing. In March 2020, the Ecuadorian state decreed a state of exception and a quarantine as a measure to counteract the spread of Covid-19. During the quarantine, the free mobilization of people was prohibited and teleworking was promoted, but it caused the paralysis of activities such as the reading of water and electricity meters.

This project consists of the theoretical-practical development of a solution for the analysis of the performance of an LPWAN network with LoRa technology through the implementation of a smart meter that allows the control and calculation of drinking water consumption remotely in the city of Quito, in order to know the optimal variables and parameters for the implementation of an LPWAN network based on LoRa. In the same way, it allows real-time visualization of water consumption through a web platform, saving human resources to measure water consumption.

The control of consumption and the visualization of the information on the web platform will allow administrators to make decisions regarding the closing and opening of drinking water supply valves, that is, it will allow the cut-off of drinking water remotely. For this, Ubidots will be used, an IoT platform that offers data storage and the integration of widgets according to the variables received.

### **KEYWORDS:**

- **SMART METER**
- **LORA**
- **LORAWAN**
- **LPWAN**
- **UBIDOTS**

## Capítulo I

### Introducción

#### *Antecedentes*

El agua es un recurso natural no renovable, esencial para la vida, la productividad y salud de las personas. El desperdicio y mal uso de este recurso afecta no solamente el bienestar de las personas sino la economía de las familias y de los países ya que se invierten más recursos para producir mayor cantidad de agua potable con el aumento de demanda de este servicio.

En una vivienda se requiere de un monitoreo y control dado que se pueden producir fallas internas en las tuberías o derivaciones clandestinas, lo cual requiere de un cierre de forma local o remota de emergencia para poder solucionar estos inconvenientes. Este control se podría realizar en cada uno de los hogares con la instalación de dispositivos que permitan realizar este trabajo de forma automática y remota haciendo uso de redes y tecnologías inalámbricas.

Los medidores inteligentes y la lectura remota de los mismos se realizan a través de una red de transmisión. Esta red debe ser robusta para evitar pérdida de paquetes, debe tener gran cobertura y distancias de transmisión y debe consumir baja potencia para evitar el cambio de baterías en los medidores. Todas las características antes mencionadas las reúnen las redes LPWAN, del acrónimo en inglés Low Power Wide Area Network, lo que en español significa Redes de Área Amplia de Baja Potencia. Dentro de estas redes se destaca la tecnología LoRa.

#### *Justificación e Importancia*

Durante los meses de pandemia muchos ciudadanos recibieron sus planillas con valores elevados y diferentes a los que habían pagado en meses anteriores a la pandemia. Este

problema generó molestias y reclamos en las diferentes empresas municipales y alcaldías que gestionan el suministro y cobro de agua potable.

En el mes de junio, y con la finalidad de garantizar el proceso de facturación, se reanudó la lectura real de los medidores de forma habitual en la ciudad de Quito, es decir, un trabajador se acerca al medidor, verifica la lectura, y emite el comprobante de consumo y pago (EL UNIVERSO, 2020). Esto conlleva a la exposición de los trabajadores e incrementa el riesgo de contagio de Covid-19.

Las tecnologías de la Información y comunicación ofrece una alternativa a solucionar estos problemas, específicamente en el marco de las ciudades inteligentes como es el despliegue de una red de área extendida y bajo consumo de potencia – LPWAN (Alliance L, 2015) interconectando dispositivos electrónicos que permitan realizar el cálculo del consumo de agua potable de forma remota y la transmisión de la información a una plataforma de visualización (Mogrovejo Narvaez, 2019).

Mediante la implementación y evaluación de esta red se pretende obtener los datos de consumo de agua potable en tiempo real, cortar o habilitar el servicio debido a falta de pago, daño o mantenimiento como una alternativa a la lectura tradicional de medidores.

### *Alcance del Proyecto*

Para el control del consumo de agua potable se implementará un prototipo de control de flujo de agua basado en Arduino. Este prototipo será instalado al interior de la vivienda. El dispositivo contará con una pantalla de visualización de flujo de caudal y consumo de forma local mediante un LCD y una batería de respaldo para su funcionamiento.

Para la etapa de corte y reactivación del servicio se utilizará una electroválvula que permita el flujo de agua o la interrupción de esta. Esta etapa estará conectada al mismo sistema de alimentación de la etapa de monitoreo al igual que al respaldo de energía.

Se implementará una red LPWAN, específicamente, redes que usen el protocolo LORAWAN como una solución al control de consumo de agua potable de la vivienda mediante el análisis de métricas de desempeño de red.

Para la visualización de la información y datos de consumo se utilizará un servidor y aplicación web orientados al internet de las cosas. En la aplicación web se realizará un tratamiento de la información para presentar los datos de consumo de forma gráfica, acceso a historiales de consumo por semana, por día y por mes.

Con el objetivo de garantizar el funcionamiento continuo del prototipo se implementará un sistema de energía de respaldo basado en un banco de baterías recargable.

Finalmente, se presentará un documento final donde se presente una fundamentación teórica de las redes LPWAN, la metodología utilizada para el desarrollo del proyecto, la propuesta y desarrollo de la solución.

### *Objetivos*

#### **General**

Implementar una red LPWAN para el control de consumo de agua potable utilizando tecnología LoRa.

#### **Específicos**

- Realizar un estudio del estado del arte sobre la tecnología LoRa.

- Dimensionar los dispositivos electrónicos a utilizar en la construcción del dispositivo de control de flujo de agua.
- Diseñar e implementar un circuito para la adquisición de los datos sobre el consumo de agua potable.
- Configurar los equipos de radio, nodo y Gateway LoRa para la transmisión de los datos de consumo de agua.
- Implementar una red “LoRa” con topología en estrella entre el nodo y Gateway que se comunica con el servidor.
- Desarrollar un aplicativo Web con conexión al servidor para realizar el monitoreo y almacenamiento de información enviada por el Gateway.
- Validar el funcionamiento del sistema a través de un protocolo de pruebas.

## Capítulo II

### *Marco Teórico*

Las redes de larga distancia y bajo consumo de potencia se han ido convirtiendo en una solución para el crecimiento de las ciudades inteligentes o Smart Cities y el Internet de las Cosas o IoT. Dentro de las redes LPWAN se destaca LoRa / LoRaWAN por su simplicidad, costo y cobertura.

En esta sección se presenta una breve descripción de las redes LoRa / LoRaWAN, sus características y ventajas.

### *Redes de largo alcance y bajo consumo de potencia*

La tecnología LPWAN (Low Power Wide Area Network) es una red de baja potencia y de área amplia, la cual es un protocolo de transporte inalámbrico de datos, que en la actualidad es uno de los protocolos para implementación de IoT.

En el ámbito de IoT (Internet of Things), las LPWANs están sobresaliendo, y captando el interés de toda la comunidad tecnológica. Las LPWAN se caracterizan por ofrecer conectividad inalámbrica de largo alcance, baja potencia y bajo coste, tales como sensores alimentados mediante baterías. Asimismo, las LPWAN presentan limitaciones significativas en cuanto a ancho de banda, tasa de mensajes y tamaño de las tramas.

Las tecnologías LPWAN pueden proporcionar conectividad a un gran número de dispositivos. Con menor consumo de energía, mayor alcance y menor coste que las redes móviles, se cree que las LPWAN conquistarán las comunicaciones de la IoT.

Al desarrollar una red se pretende dar a conocer las herramientas de suma importancia que se ha dado en ciudades inteligentes, ya que equipos de radio frecuencia es solo utilizado en ciertos ámbitos tecnológicos, pero el aporte como tal permite dar a conocer que una red diseñada e implementada es capaz de comunicar e interactuar con sensores a una larga distancia de supervisión. La interoperabilidad entre dispositivos inteligentes es una de las ventajas claras para solucionar e implementar varios repetidores y tratar de comunicar punto a punto varios sensores inalámbricos que me permitirán entrar en varios ámbitos.

LoRa se ha posicionado como las principales tecnologías dentro de las LPWAN, gracias a los servicios de comunicación bidireccionales, la facilidad de interoperabilidad entre sensores sin necesidad de instalaciones complejas y la libertad que ofrece a usuarios, desarrolladores y empresas para el despliegue de IoT.

LoRa, desarrollada por Cycleo en 2010, es una tecnología de capa física, la misma que dos años más tarde fue adquirida por Semtech, mientras que LoRaWAN es una especificación de red propuesta por la LoRa Alliance en 2015 que ofrece una capa MAC basada en la modulación LoRa.

Por tal motivo, es importante distinguir la diferencia entre LoRa y LoRaWAN. Por un lado, LoRa es una modulación basada en una técnica de modulación de espectro expandido, CSS (cuyas siglas en inglés, Chirp Spread Spectrum), con integración a la técnica de corrección de errores, FEC (cuyas siglas en inglés, Forward Error Correction). Esta tecnología usa el ancho de banda total del canal para transmitir una señal mediante broadcast, haciéndolo robusto para canalizar ruido. LoRaWAN es un protocolo de control de acceso de medios, MAC (cuyas siglas en inglés, Media Access Control), diseñado para permitir que dispositivos de baja potencia se

comunicuen con aplicaciones conectadas a Internet a través de LPWAN. Es totalmente bidireccional y fue diseñado para garantizar la fiabilidad y seguridad.

### *LoRa*

LoRa se refiere a la capa física de la modulación inalámbrica que nos permite lograr un enlace de largo alcance, para ello usa una técnica de modulación basada en técnicas de espectro ensanchado similar a FSK y una variación de Chirp Spread Spectrum (CSS). (Murillo, Angel, Chunga, & David, 2017)

LoRa emplea una velocidad de datos (data rate) variable, manejando factores de dispersión (spreading factor SF) ortogonales, permitiendo al diseñador del sistema intercambiar la velocidad de datos por el rango o potencia, a fin de optimizar el rendimiento de la red en un ancho de banda constante; opera en los anchos de banda ISM (Industrial, Scientific, and Medical) (USA: 915MHz, EU: 433MHz y 868MHz). La naturaleza de banda ancha permite compensar una mejor relación señal/ruido (SNR) lo que permite a LoRa demodular las señales incluso cuando está a 19,5 dB por debajo del piso de ruido (Quimbita & Andrés, 2018).

### *Características*

#### **Ancho de banda escalable**

Esta modulación es escalable en ancho de banda y frecuencia. A diferencia de los esquemas existentes de modulación de banda estrecha o de banda ancha, LoRa puede adaptarse fácilmente a cualquiera de los dos modos de operación con sólo unos cuantos cambios de registro de configuración sencillos.

#### **Alta robustez**

Dado que el periodo de símbolos en LoRa puede ser más largo que duración de cada conmutación de los sistemas FHSS (Frequencyhopping spread spectrum), le proporciona una inmunidad excelente a los 12 mecanismos de interferencia de pulsos AM; pudiendo obtener receptor selectivo fuera de canal de 90 dB y un co-canal de rechazo de más de 20 dB.

Lo vuelve ideal para ambientes urbanos debido al ancho de banda de los pulsos chirp por lo cual resiste a los multi-trayectos y al desvanecimiento.

### **Capacidad de red mejorada**

La modulación de SemTech LoRa emplea factores de propagación ortogonales que permiten transmitir múltiples señales de propagación al mismo tiempo y en el mismo canal sin degradación mínima de la sensibilidad del receptor. Las señales moduladas con diferentes factores de propagación aparecen como ruido en el receptor destino y pueden ser tratadas como tales.

Una de las propiedades las interesantes es la de discriminar linealmente entre errores de tiempo y frecuencia. LoRa es una modulación ideal para la localización por lo que sus aplicaciones prácticas están más enfocadas al alcance y localización en tiempo real.

La modulación LoRa puede tolerar mecanismos de interferencia de ráfaga de niveles de potencia arbitrarios hasta un 30% de la longitud del símbolo con menos de 6 dB de degradación de sensibilidad.

### *Modulación LoRa*

Muchos sistemas inalámbricos usan modulación FSK, porque es muy eficiente para lograr baja potencia. Sin embargo LoRa está basada en la Modulación CSS (Chirp Spread Spectrum) que ha sido utilizada para comunicaciones espaciales y militares por décadas debido

a larga distancias de comunicación que se puede lograr y robustez a la interferencia. LoRa es la primera en implementar esta modulación de forma comercial (Ordoñez & Yamiith, 2017).

### *Protocolo LoRaWAN*

Es el encargado de satisfacer los requerimientos aplicados al internet de las cosas enfocados a un largo alcance, bajo consumo de energía y comunicación bidireccional y de esta manera permitir que los objetos se comuniquen de manera más sencilla (Murillo, Angel, Chunga, & David, 2017)

Alguno de los servicios de la que ofrece este protocolo son los siguientes:

- Tiempo de vida de la batería: Solo transmiten cuando los datos están listos es decir la mayoría de tiempo está en reposo.
- Capacidad de la red: Los gateways tienen la capacidad de recibir información de un elevado número de nodos ya que son multicanales y la red tienen una tasa de datos adaptativa, esto en conjunto permite recibir mensajes simultáneos en múltiples canales.
- Seguridad: LoRaWAN posee dos capas de seguridad una para la red en donde se verifica la autenticidad del nodo y otra para la aplicación donde se garantiza que nadie más que el usuario final tengan acceso a los datos. Se utiliza encriptación AES (Advanced Encryption Standard) con cambio de clave utilizando un identificador IEEE EUI64.

LoRa realiza el ensanchamiento del espectro generando señales chirp que continuamente varían en frecuencia. Una ventaja de este método es que los offsets de tiempo y frecuencia entre el transmisor y el receptor son equivalentes, reduciendo en gran medida la complejidad del diseño del receptor. El ancho de banda de frecuencia del

chirp es equivalente al ancho de banda espectral de la señal (Murillo, Angel, Chunga, & David, 2017).

- Rangos de frecuencia: LoRa trabaja en la banda de frecuencia ISM (Industrial, Scientific and Medical), son bandas libres es decir que no necesitan de licencia y son destinadas para la investigación y estudio. Dependiendo de la región la banda de frecuencias para LoRa varía, en Europa esta banda va desde los 863 MHz hasta los 870 MHz con ocho canales elegidos arbitrariamente y cada uno con un ancho de banda de 0,3 MHz, en Estados Unidos, Israel, Singapur, Australia, Canadá y otros la banda ISM para LoRa va desde los 902MHz a 928MHz con 13 canales y cada uno tiene un ancho de banda de 2.16 MHz (Murillo, Angel, Chunga, & David, 2017)

La comunicación entre los nodos y los Gateways se extiende en diferentes canales de frecuencia y velocidades de datos. La selección de la velocidad de datos es una compensación entre el intervalo de comunicación y la duración del mensaje. Debido a la tecnología Spread Spectrum, las comunicaciones con diferentes velocidades de datos no interfieren entre sí y crean un conjunto de canales "virtuales" que aumentan la capacidad del gateway. Las velocidades de datos de LoRaWAN oscilan entre 0,3 kbps y 50 kbps. Para maximizar tanto la duración de la batería de los dispositivos finales como la capacidad total de la red, el servidor de red LoRaWAN gestiona la velocidad de datos y la salida de RF para cada dispositivo final individualmente mediante un esquema de velocidad de datos adaptativa (ADR) (Ordoñez & Yamiith, 2017).

### **Figura 1**

### Características de LoRaWAN de pendiendo de la Región

	Europe	North America	China	Korea	Japan	India
Frequency band	867-869MHz	902-928MHz	470-510MHz	920-925MHz	920-925MHz	865-867MHz
Channels	10	64 + 8 + 8	In definition by Technical Committee			
Channel BW Up	125/250kHz	125/500kHz				
Channel BW Dn	125kHz	500kHz				
TX Power Up	+14dBm	+20dBm typ (+30dBm allowed)				
TX Power Dn	+14dBm	+27dBm				
SF Up	7-12	7-10				
Data rate	250bps- 50kbps	980bps-21.9kpbs				
Link Budget Up	155dB	154dB				
Link Budget Dn	155dB	157dB				

Nota. Recuperado de: (Ordoñez & Yamiith, 2017).

#### Canales y velocidades de transmisión

La separación de los canales físicos utilizando el mecanismo del SF es muy eficiente. Por lo tanto, dos paquetes enviados al mismo tiempo en el mismo canal utilizando diferentes SF no interferirán entre sí y se recibirán correctamente. En otras palabras, un nodo puede transmitir por cualquier canal y empleando cualquier bit rate (velocidad de datos), siempre que se lleve a cabo las siguientes condiciones (Quimbita & Andrés, 2018):

- Para cada transmisión el nodo debe cambiar de canal de forma pseudo-aleatoria.
- El nodo debe estar por debajo del máximo ciclo de transmisión y del tiempo máximo de transmisión en relación a la sub-banda empleada, además de seguir las regulaciones locales.

Los módulos LoRaWAN, disponen de:

- 16 canales para: 433MHz y 868MHz (Europa)
- 72 canales para: 900MHz.

**Tabla 1**

*Canales para la banda de frecuencia de 868 MHz y 433 MHz – LoRa WAN*

Número de canal	Parámetros	Bandas de frecuencia	
		868 MHz	433 MHz
0	Frecuencia	868100 MHz	433175MHz
	Ciclo de trabajo	0.33%	0.33%
	Data Rate	0-5	0-5
	Estado	On	On
1	Frecuencia	868300 MHz	433375MHz
	Ciclo de trabajo	0.33%	0.33%
	Data Rate	0-5	0-5
	Estado	On	On
2	Frecuencia	868500 MHz	433575MHz
	Ciclo de trabajo	0.33%	0.33%
	Data Rate	0-5	0-5
	Estado	On	On
3-15	Frecuencia	868325 kHz - 869750 kHz	868325 kHz - 869750 kHz
	Ciclo de trabajo	*	*
	Data Rate	0-5	0-7
	Estado	Off	Off
$*T_{off} = \frac{\text{Time on air}}{\text{DutyCycle subband}} - \text{TimeOnAir}$			

### *Clasificación de los dispositivos LoRa*

Existen 3 clases de dispositivos que manejan LoRaWAN definidos como A, B y C, aunque todos los dispositivos implementan al menos la funcionalidad de la clase A.

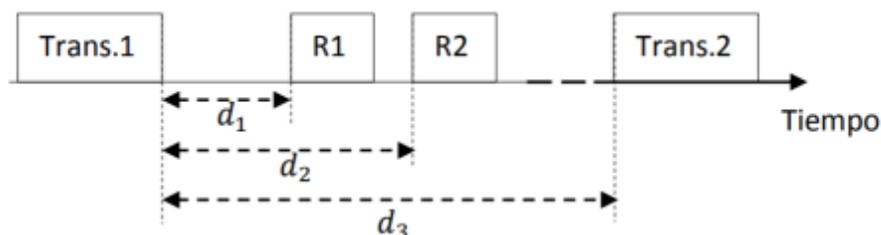
#### *Clase A*

Los nodos de la Clase A permiten comunicaciones bidireccionales por las cuales la transmisión de enlace ascendente de nodo es seguida por dos ventanas cortas de recepción de enlace descendente. La ranura de transmisión programada por el dispositivo final se basa en sus

propias necesidades de comunicación con una pequeña variación basada en una base de tiempo aleatoria (tipo de protocolo ALOHA) (Ordoñez & Yamiith, 2017).

**Figura 2**

*Acceso al medio en clase A de LoRaWAN.*



*Nota.* Recuperado de: (Ordoñez & Yamiith, 2017).

La Figura 1 representa el acceso al medio en la clase A de LoRaWAN, que se utiliza durante el procedimiento de activación, y durante el intercambio de datos (con temporizadores diferentes). Después de una transmisión, se abre una primera ventana de recepción. Si no se detecta ningún preámbulo durante la primera ventana de recepción, o si se detecta una transmisión para otro nodo, se abre la segunda ventana de recepción. Independientemente del hecho de que se reciba o no un preámbulo, el canal (así como otros canales de la misma subbanda) no se puede utilizar durante una duración, dependiendo del ciclo útil (Ordoñez & Yamiith, 2017).

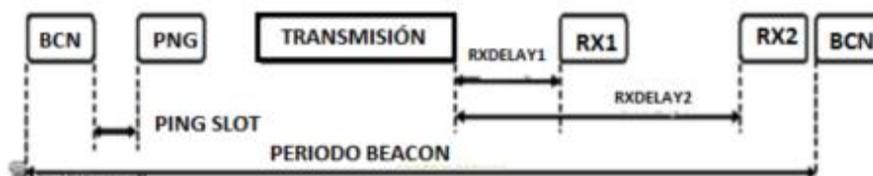
### *Clase B*

Dispositivos finales bidireccionales con ranuras de recepción programadas (Clase B): Los dispositivos finales de la Clase B permiten más ranuras de recepción. Además de las ventanas de recepción aleatoria de clase A, los dispositivos de clase B abren ventanas de recepción adicionales a horas programadas. Para que el dispositivo final abra la ventana de recepción a la

hora programada, recibe una baliza sincronizada de tiempo desde la puerta de enlace. Esto permite al servidor saber cuándo el dispositivo final está escuchando (Ordoñez & Yamiith, 2017).

**Figura 3**

*Acceso al medio en la clase B LoRaWAN*



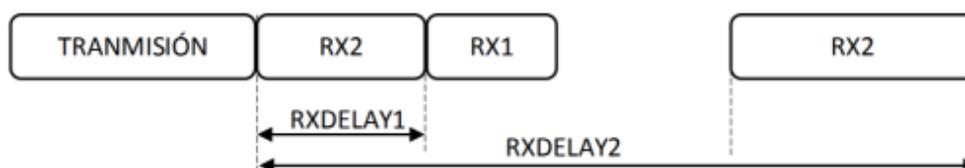
*Nota.* Recuperado de: (Ordoñez & Yamiith, 2017).

### *Clase C*

Los dispositivos finales de la clase C tienen ventanas de recepción abiertas casi continuamente, sólo cerradas cuando se transmiten. El dispositivo final de Clase C utiliza más energía para funcionar que la Clase A o la Clase B, pero ofrece la menor latencia para la comunicación entre el servidor y el dispositivo final.

**Figura 4**

*Acceso al medio en la clase C LoRaWAN*



*Nota.* Recuperado de: (Ordoñez & Yamiith, 2017).

### *Internet de las cosas*

Se define como el desarrollo de un primer Internet enfocado en las personas con la disposición para unir datos con personas, procesos y objetos. Tomando en cuenta sensores,

redes de comunicaciones avanzadas y procesos analíticos fundamentados en Big Data, lo que permitirá generar aplicaciones que favorecerán la vida de las personas, enriquecer la educación, fortificar ciudades, edificios y redes eléctricas, incrementarán la seguridad de los datos y la eficiencia de las empresas (Cuzme Rodríguez, 2015).

### *Características del IoT*

A continuación, se detallan las características más relevantes:

- Interconectividad: Cualquier dispositivo puede ser vinculado a Internet (Vermesan & Friess, 2014).
- Servicios relacionados con los objetos: Servicios como por ejemplo salvaguardar la privacidad y cohesión entre la correlación de los objetos físicos y los entornos virtuales asociados (Vermesan & Friess, 2014).
- Heterogeneidad: Son capaces de relacionarse con otros dispositivos o plataformas de servicio por medio de distintas redes (Vermesan & Friess, 2014).
- Cambios dinámicos: Los dispositivos modifican su estado de manera dinámica (Vermesan & Friess, 2014).
- Gran escala: La cantidad de dispositivos que pueden controlarse es escalable (Vermesan & Friess, 2014).
- Seguridad: Los dispositivos IoT poseen protocolos de seguridad que fortalecen la información personal, nodos finales, redes y los datos (Vermesan & Friess, 2014).

### *Servidores de IoT*

Son servidores, de código abierto, de uso libre y pagados que permiten implementar servicios y componentes para la creación de redes inalámbricas. Para el caso de LoRa, las redes a implementar son las LPWAN.

Estos servidores cuentan con una interfaz web amigable cuyo uso esta orientado al inventario de equipos dentro de la infraestructura de red, administrar usuarios, aplicaciones, visualización de datos, entre otros. Con estos servidores lo que se pretende hacer es generar la visualización web de las bicicletas en un mapa o transferirlas a una app móvil. El servidor LoRa utilizado en este proyecto es The Things Network – TTN. Este servidor proporciona un conjunto de herramientas para administración de dispositivos finales, gateways y aplicaciones orientado a redes LoRa. Ofrece un alta seguridad en el manejo de datos mediante una encriptación robusta end-to-end. Actualmente TTN opera con alrededor de miles de gateways en todo el mundo como se muestra en la Figura 6.

### **Figura 5**

*Servidor para aplicaciones IoT de uso libre*



*Nota.* Recuperado de: (TTN, 2020).

### **Figura 6**

*Gateways LoRa conectados a TTN a nivel mundial.*



*Nota.* Recuperado de: (TTN, 2020).

### *Dispositivo de FrontEnd*

Los dispositivos FrontEnd son aquellos en los que los usuarios visualizan la información entregada por los nodos en la red LoRa. Estos dispositivos pueden ser computadores o teléfonos inteligentes. Si el usuario accede a la información mediante un computador personal o de escritorio, lo deberá hacer a través de la aplicación web, en cambio, si lo hace desde un celular, se deberá descargar una aplicación móvil para la visualización.

### Capítulo III

En este capítulo se describe de forma general la propuesta del proyecto para posteriormente en el siguiente capítulo describir la implementación, y pruebas de funcionamiento del mismo. Se empieza por una breve descripción del funcionamiento del proyecto utilizando un diagrama de bloques para una mejor comprensión. En segundo lugar, se presentan los aspectos técnicos del sistema, seguidamente de un análisis de los recursos utilizados y finalizando con una enumeración de las ventajas del producto

#### *Descripción general del proyecto*

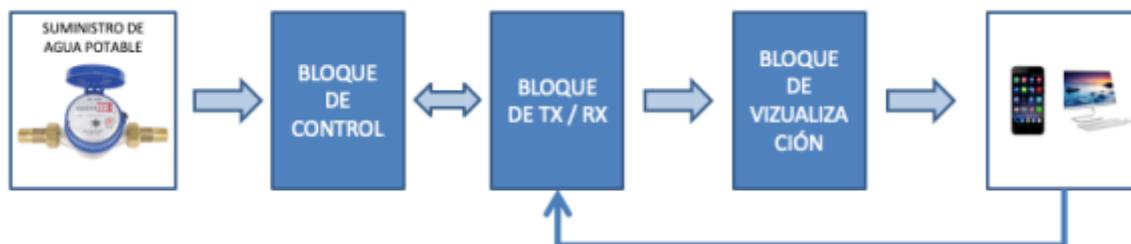
Para el desarrollo del presente proyecto se lo dividió en 5 fases:

- La primera fase consiste en una recopilación bibliográfica referente a redes LPWAN, redes LORA, casos de uso y dispositivos utilizados.
- En la segunda fase se implementó los dispositivos de medición y lectura, se realizaron las pruebas de funcionamiento del sistema de medición de consumo de agua.
- En la fase 3 se implementó la red y sus diferentes elementos como nodos finales y gateway para la transmisión y recepción de las lecturas de consumo.
- Durante la fase 4 se realizó las pruebas de campo y la evaluación de la red.
- Finalmente, la fase número 5 fue dedicada netamente para la redacción de resultados

En la Figura 7 se ilustra el funcionamiento del sistema a través de bloques generales de funcionamiento.

#### **Figura 7**

*Diagrama general de bloques del funcionamiento del sistema.*



### *Bloque de control*

El bloque de control está conectado al suministro de agua y es el encargado de abrir y cerrar una electroválvula permitiendo el flujo o interrupción de agua. Adicional en este bloque se instalará un medidor de flujo o caudal para determinar la cantidad de agua que se está consumiendo. Los datos se los obtendrán mediante una tarjeta Arduino y se dispondrá de un LDC para la visualización de información.

### *Bloque de TX/RX*

Este bloque es el encargado de transmitir los datos obtenidos por la tarjeta Arduino en el bloque anterior. La tecnología utilizada para la comunicación inalámbrica es LoRa. Este bloque cuenta con una comunicación de dos vías con el bloque de control ya que en la etapa de recepción se obtienen las órdenes de cierre o apertura de la electroválvula.

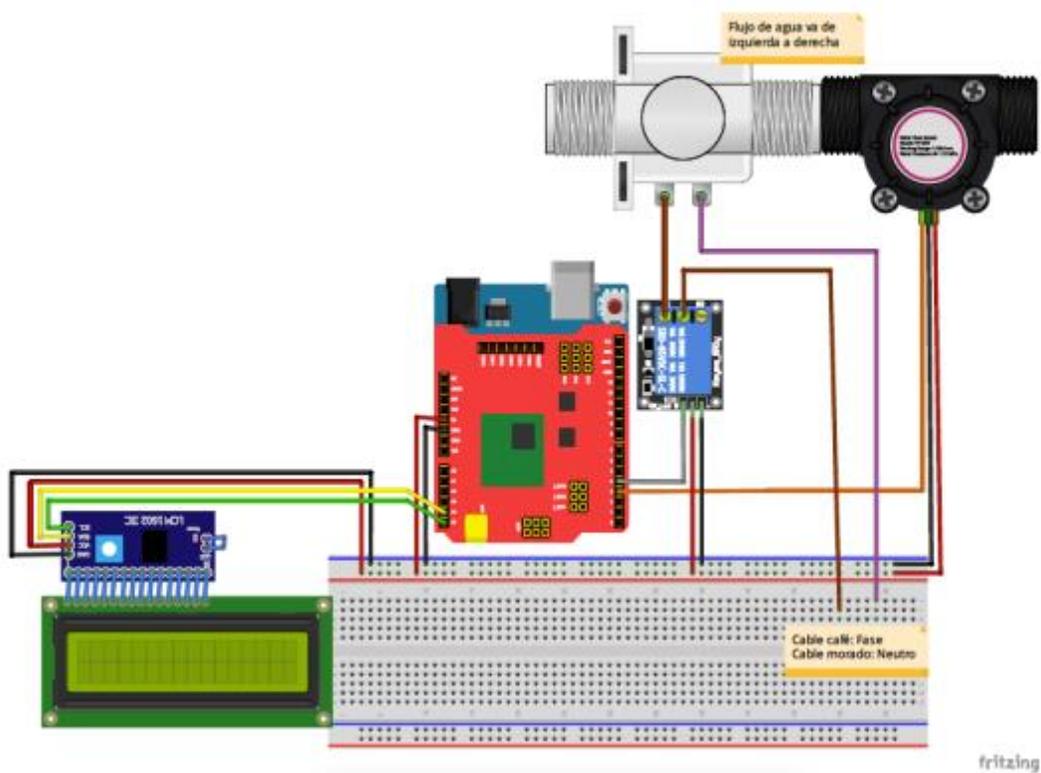
En la Figura 8, se puede observar el hardware o elementos electrónicos y su conexión para el funcionamiento de los bloques 1 y 2 descritos anteriormente. Los materiales utilizados para la transmisión en tecnología LoRa es el shield Dragino que opera a 915 MHz y es compatible con Arduino Uno.

### *Bloque de visualización*

En este bloque se genera el análisis de la información. La información se almacena en una base de datos en la nube dedicada a soluciones orientadas al Internet de las Cosas. Este es un servidor web y el acceso se lo puede hacer por medio de un teléfono inteligente o por un computador.

### **Figura 8**

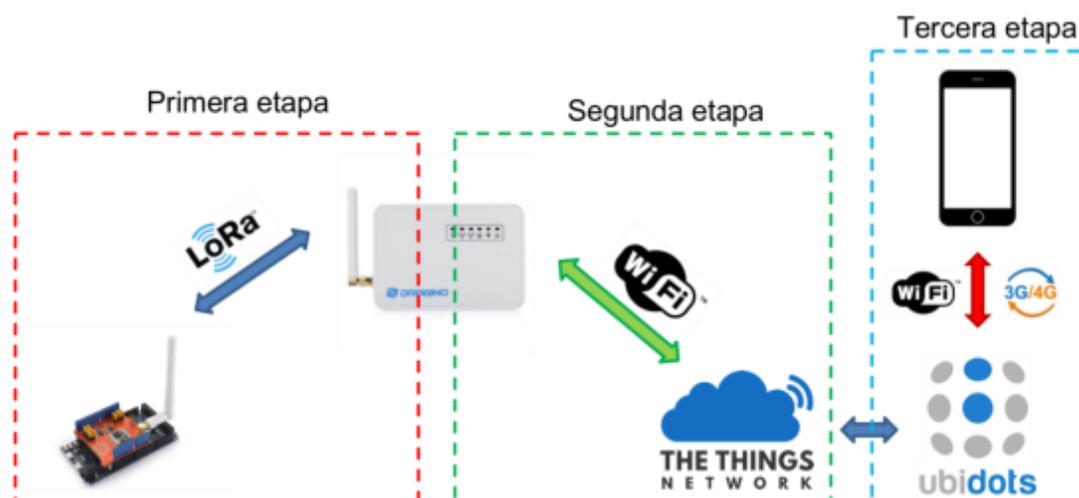
*Esquema electrónico de conexión para los bloques 1 y 2.*



En la Figura 9, se puede observar un diagrama de funcionamiento del sistema. Basados en este diagrama y en los bloques de funcionamiento explicados anteriormente se puede realizar la siguiente descripción.

**Figura 9**

*Diagrama de las etapas de funcionamiento del sistema*



La primera etapa se encarga de la recolección de la información proveniente del sistema de cálculo de consumo y control de agua, es decir, de los sensores y electroválvula instalados y la recepción del mismo en el gateway LoRa. La característica principal de esta etapa es la transmisión de la información por medio tecnología LoRa, para esto se utilizó el Shield LoRa para Arduino. Cabe recalcar que LoRa trabaja a nivel de la segunda capa del modelo de referencia OSI, es decir, en la capa física para los enlaces.

Una vez el mensaje de consumo de agua llega al gateway, este lo procesa y lo encamina a la pasarela IoT. En este proyecto se utiliza una pasarela de uso libre y gratuito TTN – The Things Network. Para la transmisión de los datos desde el gateway al servidor TTN se utiliza la tecnología WiFi.

En la pasarela IoT (TTN) se configuran los dispositivos LoRa finales y el gateway para poder realizar una lectura y escritura sobre los mismos. En esta pasarela se configura también el servidor IoT Ubidots para la visualización de datos y estadísticas de consumo.

En la etapa tres, se realiza la presentación de la información de consumo. En este servidor IoT se pueden configurar la forma de presentar los datos que se reciben, es decir, se configura botones, formas gráficas de visualización y tiempos de muestreo, es decir, se configura el periodo de consulta de los datos. A este servidor se lo puede acceder por medio de un smartphone o por un computador. En términos de tecnología, se puede acceder por medio de redes WiFi o por redes de telefonía celular como 3G o 4G

#### *Módulos que componen el proyecto*

Para una mejor identificación de los módulos que componen el proyecto, se lo separó en 2 etapas, la primera encargada del censado, procesamiento y visualización de la información y la segunda encargada de la transmisión y recepción de los datos hacia y desde la plataforma web. De esta manera, se tienen los siguientes módulos:

##### Primera etapa:

- Sensor de flujo de caudal YF-S201
- Electroválvula de bola
- Arduino UNO
- Relé
- Módulo adaptador I2C PCF8574
- LCD 20x04
- Banco de batería recargable de 4000 mAH.
- Módulo de reloj Ds3231

Segunda etapa:

- Shield LoRa Dragino SX127x
- Gateway LoRa Dragino LG02

Las características técnicas de cada uno de los elementos utilizados se detallan en la

Tabla 2.

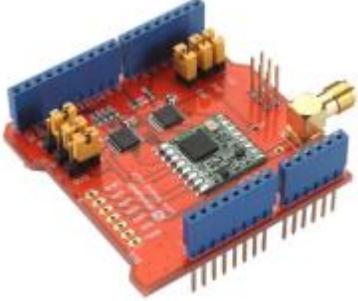
**Tabla 2**

*Descripción técnica de los elementos utilizados en el proyecto*

Nombre	Gráfico del dispositivo	Características
Sensor experimental de flujo de caudal <b>YF-S201</b>		<ul style="list-style-type: none"> <li>• Voltaje de funcionamiento 5V – 18V DC.</li> <li>• La salida es una onda cuadrada de pulso.</li> <li>• Permite medir flujos en el rango de 1 a 30 litros por minuto.</li> <li>• 450 pulsos por cada litro</li> <li>• Factor de conversión 7.5</li> <li>• Rosca externa de ½ pulgada.</li> <li>• Presión de trabajo máximo de 1.75 Mpa o 17 bares.</li> <li>• Precisión <math>\pm 10\%</math></li> </ul>
Electroválvula de bola		<ul style="list-style-type: none"> <li>• Voltaje de funcionamiento 9V-24V DC.</li> <li>• Tamaño de tubería ½ pulgada.</li> <li>• Roscado NPT</li> <li>• La apertura y cerrado de la válvula es acuerdo al tiempo que se suministre la energía.</li> <li>• No consume energía una vez que se cierra o abre completamente la válvula.</li> <li>• Potencia 2 Watts</li> <li>• Presión 2MPa</li> </ul>

Nombre	Gráfico del dispositivo	Características
Relé 5V - 110V/10A		<ul style="list-style-type: none"> <li>• Módulo relé de un canal.</li> <li>• Voltaje de alimentación y control de 5V DC.</li> <li>• Salida 10 A/110 VAC.</li> <li>• Corriente de activación de 200mA</li> <li>• 1 led indicador.</li> <li>• Tensión máxima en DC de 30V y 10A</li> <li>• Tensión máxima en AC de 120V y de 10A.</li> </ul>
Módulo adaptador LCD <b>I2C PCF8574</b>		<ul style="list-style-type: none"> <li>• Permite manejar un LCD de 16x2 o 20x4 utilizando solamente los pines SDA y SCL de los display.</li> <li>• Comunicación serial I2C. Dirección 0x3F.</li> <li>• Cuenta con un jumper para controlar la luz de fondo del display.</li> <li>• Tiene un potenciómetro para controlar y regular el contraste del display.</li> </ul>
LCD 20x04		<ul style="list-style-type: none"> <li>• Pantalla alfanumérica de 4 filas y 20 columnas.</li> <li>• Luz de fondo tipo LED de color azul regulable al igual que el contraste de los caracteres.</li> <li>• Interfaz de comunicación paralela y serial de 8 y 4 bits respectivamente.</li> <li>• Compatible con el controlador I2C PCDF8574.</li> <li>• Controlador interno Hd44780.</li> <li>• Voltaje de funcionamiento 5VDC.</li> </ul>

Nombre	Gráfico del dispositivo	Características
Banco de batería recargable de 4000 mAh		<ul style="list-style-type: none"> <li>• Una batería interna de polímero de litio.</li> <li>• Voltaje de funcionamiento y carga 5.1 VDC.</li> <li>• Potencia eléctrica de 45 W</li> <li>• Tiempo de carga 6.5 horas.</li> <li>• Carga eléctrica de la batería 4000 mAh.</li> <li>• Dimensiones: 15 x 7.3 x 3 cm.</li> <li>• Cuenta con protección ante cortocircuitos.</li> </ul>
Módulo de reloj <b>Ds3231</b>		<ul style="list-style-type: none"> <li>• Módulo reloj de muy alta precisión y de bajo costo.</li> <li>• Comunicación I2C con velocidad máxima de Tx de 400kHz</li> <li>• Oscilador integrado de cristal</li> <li>• Batería integrada CR2032</li> <li>• Voltaje de funcionamiento 3,3 – 5V DC</li> <li>• Capacidad de memoria de 32 KB</li> </ul>
Arduino UNO		<ul style="list-style-type: none"> <li>• Cuenta con un microcontrolador Atmega328P.</li> <li>• Voltaje de funcionamiento de 5 VDC. Con un rango de entrada entre 7 y 12 VDC con un voltaje máximo de 6 a 20 VDC.</li> <li>• 14 pines digitales de entrada y salida.</li> <li>• 6 pines de entrada analógica.</li> <li>• 20 mA de corriente continua en cada pin.</li> <li>• Memoria flash de 32 KB.</li> <li>• Memoria SRAM de 2KB.</li> <li>• Memoria EEPROM de 1KB.</li> <li>• 16 MHz de velocidad de reloj.</li> </ul>

Nombre	Gráfico del dispositivo	Características
Shield LoRa Dragino <b>SX127x</b>		<ul style="list-style-type: none"> <li>• Shield diseñado para usarlo sobre Arduino UNO, Leonardo, Mega y DUE.</li> <li>• Compatible con 3.3V o 5V de la tarjeta Arduino.</li> <li>• Basado en el chip SX1276.</li> <li>• Sensibilidad máxima de -148dBm.</li> <li>• Banda de frecuencia de 915 MHz.</li> <li>• Modulación FSK, GFSK, MSK, GMSK, LoRaTM y OOK.</li> </ul>
Gateway LoRa Dragino <b>LG02</b>		<ul style="list-style-type: none"> <li>• Cuenta con un procesador basado en Linux de 400MHz.</li> <li>• Memoria Flash de 16MB.</li> <li>• Memoria RAM de 64MB.</li> <li>• Interfaz WiFi 802.11 b/g/n.</li> <li>• Voltaje de funcionamiento de entrada de 12VDC.</li> <li>• 2 interfaces inalámbrica LoRa full-dúplex.</li> <li>• Banda de frecuencia de 915 MHz.</li> <li>• Bajo consumo de potencia.</li> <li>• Configuración y administración del dispositivo por medio de Web, SSH, LAN o WiFi.</li> <li>• Soporta hasta 300 nodos.</li> </ul>

### *Análisis de recursos*

En el mercado aún no existen medidores comerciales de caudal que permitan tener un control del consumo interno de agua de un domicilio o residencia, es por eso que se desarrolló un prototipo con tecnología LoRa, prácticamente nuevo en el mercado, donde se obtuvo muy buenos resultados. A continuación, se detalla en la Tabla 3 todos los recursos utilizados en el proyecto.

**Tabla 3**

*Elementos utilizados en la construcción del sistema*

<b>NOMBRE</b>	<b>PRECIO</b>	<b>CANTIDAD</b>	<b>TOTAL</b>
Puerta de enlace LoRa IoT DRAGINO LG02 de doble canal	\$ 147,60	1	\$ 147,60
Dragino Lora Shield 915Mhz	\$ 50,40	1	\$ 50,40
Módulo Arduino UNO	\$ 15,00	1	\$ 15,00
Batería portable de 4000A/h	\$ 25,00	1	\$ 25,00
Computador i7 - 8GB RAM - 500HD SSD	\$ 700,00	1	\$ 700,00
Electroválvula de bola 12 VDC	\$ 70,00	1	\$ 70,00
Sensor de flujo de agua Yf-s201c	\$ 10,00	2	\$ 20,00
LCD 20x04	\$ 15,00	2	\$ 30,00
Módulo adaptador I2C PCF8574	\$ 2,50	1	\$ 2,50
Módulo de reloj Ds3231	\$ 4,00	1	\$ 4,00
Protoboard 830 Puntos 1 Regleta	\$ 4,00	3	\$ 12,00
Caja Plástica impermeable para exteriores	\$ 20,00	1	\$ 20,00
Modulo Relé 5v 1 Canal 110-220vac 10a Optoacoplado	\$ 2,00	4	\$ 8,00
			<b>\$ 1.104,50</b>

Se empleó la tecnología LoRa, ya que un medidor de agua se encuentra por lo general fuera de las residencias o domicilios, por lo que la solución es cubrir la distancia que separa el medidor de la residencia y básicamente el Gateway de salida al Internet o puerta de enlace de la empresa que provee el servicio de Internet. Se realizó pruebas de conexión entre el módulo y Gateway muy efectivas.

Se utiliza el módulo Arduino UNO ya que es muy amigable y versátil lo cual ayudó a la programación del sensor de caudal y el control de la electroválvula, así como la visualización de la información en el LCD indispensables en este proyecto.

En el lector y sensor de consumo de agua se incorporó el LCD, en una caja muy liviana hermética e impermeable para su uso en exteriores. Dado que es un dispositivo que debe funcionar las 24 horas del día se incorporó una batería portátil de 4000mAh recargables, lo cual permite realizar la lectura y control de consumo de agua potable con un respaldo de energía de 10 horas (El cálculo de las horas se lo muestra en el siguiente capítulo).

Se utiliza la computadora para la programación de los algoritmos para la lectura, control de la electroválvula y transmisión de información mediante LoRa, así como la toma de datos a través de la pasarela de datos IoT - TTN (The Things Network), en tiempo real.

El desarrollo de este proyecto y las pruebas realizadas para la evaluación de resultados tomaron aproximadamente 80 horas, lo que en conjunto con el computador son los rubros más caros en el desarrollo del proyecto, llegando a ocupar el 72.16% del valor total del prototipo.

Si el producto finalmente se logra comercializar, los rubros de mano de obra y el computador ya no se contemplarían en la producción ya que no es necesario desarrollar algoritmos, diseño y las pruebas de funcionamiento, solamente se grabarían los ya desarrollados y probados en cada sistema a vender, es decir, el costo final de producción por cada prototipo a vender sería de 368 dólares aproximadamente, es decir, el valor de producción se reduce en un 75,57%. Los valores de producción se muestran en la Tabla 4:

**Tabla 4**

*Precio para la producción en masa del sistema*

NOMBRE	PRECIO	CANTIDAD	TOTAL
Puerta de enlace LoRa IoT DRAGINO LG02 de doble canal	\$ 147,60	1	\$ 147,60
Dragino Lora Shield 915Mhz	\$ 50,40	1	\$ 50,40
Módulo Arduino UNO	\$ 15,00	1	\$ 15,00

Batería portable de 4000A/h	\$	25,00	1	\$	25,00
Impresión 3D de la carcasa	\$	30,00	1	\$	30,00
Electroválvula de bola 12 VDC	\$	15,00	1	\$	15,00
Sensor de flujo de agua Yf-s201c	\$	10,00	2	\$	20,00
LCD 20x04	\$	15,00	2	\$	30,00
Módulo adaptador I2C PCF8574	\$	2,50	1	\$	2,50
Módulo de reloj Ds3231	\$	4,00	1	\$	4,00
Protoboard 830 Puntos 1 Regleta	\$	4,00	3	\$	12,00
Modulo Relé 5v 1 Canal 110-220vac 10a Optoacoplado	\$	2,00	4	\$	8,00
Modulo Relé 5v 1 Canal 110-220vac 10a Optoacoplado	\$	2,00	4	\$	8,00
				<b>\$</b>	<b>367,50</b>

### *Ventajas del producto*

El poder conocer el consumo actual de consumo de agua y tener un control en el suministro del mismo generan varias ventajas, entre las que se describen:

- Ahorro económico: Al poder cerrar y abrir una electroválvula y con eso permitir el flujo de agua es un aspecto importante al momento de evitar desperdicios, fugas o robo de agua en un domicilio.
- Visualización de estadísticas de consumo: esto permite al usuario saber los días en que la familia consume más agua y le puede permitir buscar una solución o analizar el porqué del aumento del consumo.
- Funcionamiento continuo: al contar con un banco de baterías de respaldo para su funcionamiento, se puede tener un registro continuo del consumo y poder generar mucho mejor las estadísticas.
- Comunicación a largas distancias: al utilizar la tecnología LoRa se puede establecer una comunicación a grandes distancias entre el gateway y el nodo

final, es decir, se puede tener el control no solamente de domicilios en lugares urbanos, al contrario, se puede tener y desplegar esta solución en entornos rurales cubriendo largas distancias. Esto posibilita su uso en granjas, invernaderos y fincas

## Capítulo IV

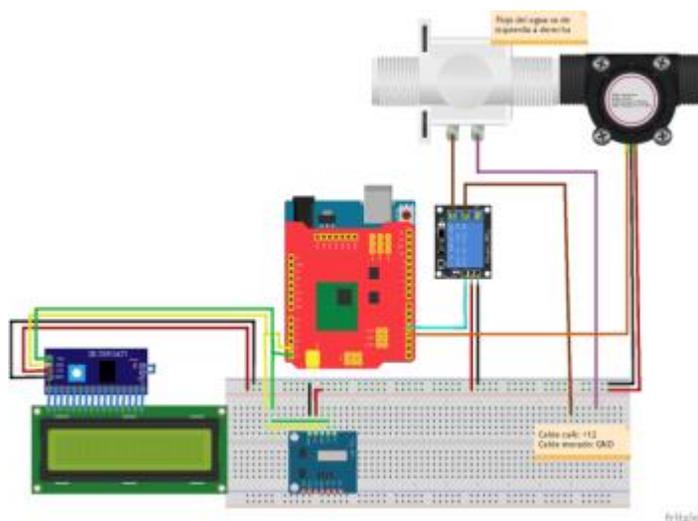
En este capítulo se describen los procesos de construcción del sistema de control de consumo interno de agua potable, así como las pruebas realizadas para la calibración del mismo y el análisis de resultados de dichas pruebas.

### *Proceso de ensamblado del hardware*

En la Figura 10 se detalla el esquema del circuito que se implementó en una baquelita perforada ideal para circuitos electrónicos. En la cual se usaron headers hembra que facilitan las conexiones de los componentes empleados como, por ejemplo: DS3230, LCD20x4, sensor de flujo YF-S201 y relé.

### **Figura 10**

*Esquema del circuito en bloques simplificado*



El primer paso es cortar el header hembra según la cantidad a usar, esto depende de la cantidad de pines macho de cada uno de los componentes anteriormente mencionados. A continuación, se detalla la cantidad de headers hembra para cada componente:

- 6 headers para el DS3230.
- 2 headers para la fuente de alimentación (5V DC).
- 2 headers para la alimentación del sensor de flujo YF-S201.
- 2 headers para la alimentación del relé.
- 4 headers para el LCD20X4.

**Figura 11**

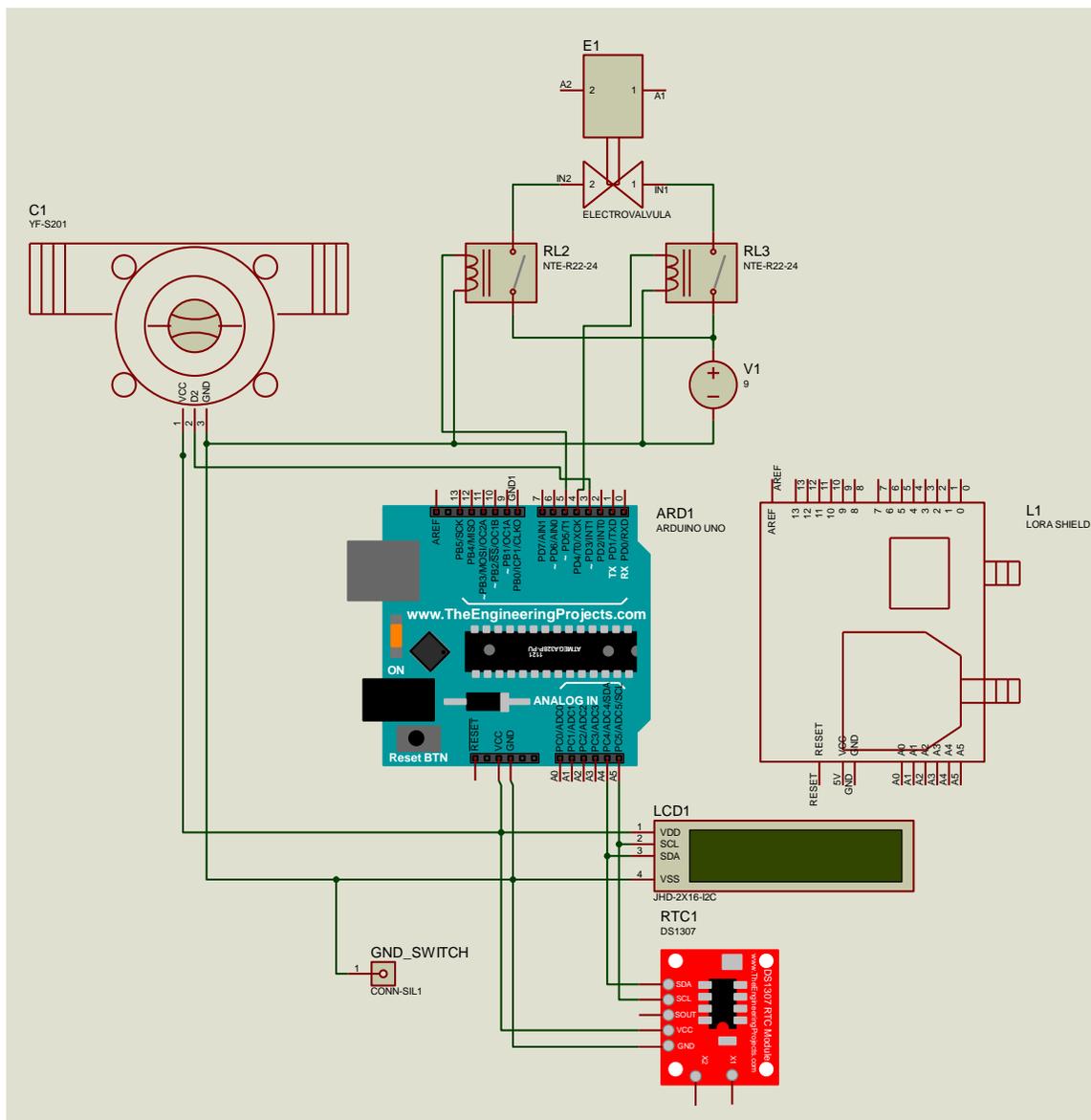
*Headers hembra*



El segundo paso es ubicar los headers hembra ya cortados al tamaño que se necesita en la baquelita y luego cortar los cables que permitan conectar estos componentes siguiendo el esquemático de la Figura 12.

**Figura 12**

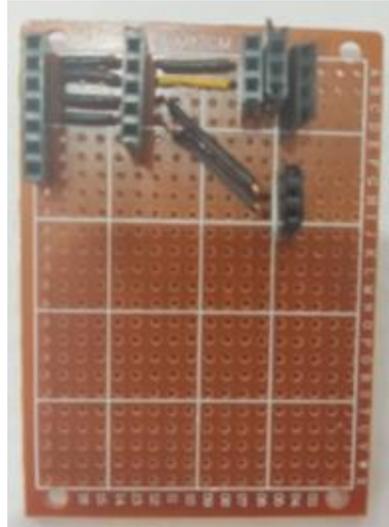
### Diseño electrónico de conexión del sistema



En la Figura 13 se observa la ubicación de los headers y cables en la baquelita para posteriormente proceder a soldarlos.

**Figura 13**

### *Ubicación de los headers hembra y cables en la baquelita*



El último paso es soldar los headers y cables ya ubicados en la baquelita, para ello empleamos un cautín. Se calienta la superficie y se coloca estaño teniendo mucho cuidado de cortocircuitar los componentes contiguos.

### *Proceso de ensamblado del software*

Para realizar el presente proyecto se utilizó las siguientes librerías, que se pueden observar en la Figura 14, las cuales facilitarán el manejo del LCD20X4, DS3231 RTC, módulo Lora y la librería que permite guardar datos en la EEPROM.

### **Figura 14**

#### *Librerías usadas en Arduino*

```
#include <LCD.h>
#include <LiquidCrystal_I2C.h> //librería para LCD
#include <Wire.h> // libreria para interfaz I2C
#include <lmic.h> //librerías para la tx (lora)
#include <hal/hal.h>
#include <SPI.h>
#include <RTClib.h> //libreria para el manejo del modulo RTC
#include <EEPROM.h> //librería para guardar datos en la eeprom
```

Luego se procede a instanciar dos objetos, uno para el LCD y otro para el módulo RTC. Esta instanciación nos facilitará la inicialización de ambos dispositivos.

### Figura 15

*Instanciación de objetos*

```
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
RTC_DS3231 rtc;      // crea objeto del tipo RTC_DS3231
```

A continuación, se declara dos constantes: “control\_valvula” y “sensor”, los cuales representan los pines digitales en donde se encuentran conectados la electroválvula y el sensor de flujo YF-S201 respectivamente. También se declara variables empleadas para el cálculo del caudal y el consumo en metros cúbicos, una variable para almacenar el valor recibido del enlace downlink y variable que lleva el conteo de mensaje enviados (uplink).

### Figura 16

*Constantes y variables empleadas en la programación*

```
const int control_valvula=4; // Relay
const int sensor=3; //Sensor de flujo YF-S201
int litros_Hora; // Variable que almacena el caudal (L/hora)
volatile int pulsos = 0; // Variable que almacena el número de pulsos
unsigned long tiempoAnterior = 0; // Variable para calcular el tiempo transcurrido
unsigned long pulsos_Acumulados = 0; // Variable que almacena el número de pulsos acumulados
float litros; // // Variable que almacena el número de litros acumulados
float metros_cubicos=0; //Variable que almacena los metros cúbicos
int valor_recibido=48; //Variable que recibe el dato downlink
unsigned int count = 0; //Contador de los mensajes de lora
```

Posteriormente se define el tamaño del payload que va enviar el nodo al Lora Gateway, el cuál es de 10 bytes. También se define las credenciales de TTN (NWKSKEY y APPSKEY) para acceder a la aplicación creada en dicha plataforma y el identificador del nodo (DEVADDR).

### Figura 17

### Declaración del payload y credenciales de TTN

```
static uint8_t mydata[10] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00};
static const PROGMEM ul_t NWSKEY[16] = { 0xFE, 0x0C, 0xDF, 0xF5, 0x43, 0x12, 0x50, 0x62, 0xAA, 0x68, 0x07, 0xC6, 0xD6, 0xB5, 0x76, 0xC9 };
static const ul_t PROGMEM APPSKEY[16] = { 0x78, 0x57, 0x07, 0x9C, 0x73, 0xC1, 0x6C, 0x56, 0xDF, 0xA0, 0xCC, 0x70, 0x3F, 0x0E, 0xC7, 0xB0 };
static const u4_t DEVADDR = 0x2601105E;
```

En la Figura 18 se establece el intervalo de tiempo que se realiza la transmisión del payload desde el nodo al Lora Gateway.

### Figura 18

*Intervalo de tiempo de la transmisión Lora*

```
/* Schedule TX every this many seconds (might become longer due to duty
   cycle limitations).*/
const unsigned TX_INTERVAL = 6; //intervalo de tiempo de tx lora segundos
```

En la Figura 19 se declara los pines digitales usado para gestionar la comunicación entre el Arduino y el módulo Lora.

### Figura 19

*Pines digitales usados por el módulo Lora*

```
const lmic_pinmap lmic_pins = {
  .nss = 10,
  .rxtx = LMIC_UNUSED_PIN,
  .rst = 9,
  .dio = {2, 6, 7},
};
```

En la Figura 19 se observa la función `do_send` encargada verificar si hay una transmisión o recepción de un mensaje en proceso, caso contrario realiza el proceso de envío del mensaje e imprime información de la frecuencia uplink.

### Figura 20

### Función que realiza el envío de mensajes

```

void do_send(osjob_t* j) {
    // Check if there is not a current TX/RX job running

    if (LMIC.opmode & OP_TXRXPEND) {
        Serial.println("OP_TXRXPEND, not sending");
    } else {
        envio();
        // Prepare upstream data transmission at the next possible time.
        // LMIC_setTxData2(1,datasend,sizeof(datasend)-1,0);
        LMIC_setTxData2(1, mydata, sizeof(mydata), 0);
        Serial.println("Packet queued");
        Serial.print("LMIC.freq:");
        Serial.println(LMIC.freq);
        Serial.println("Receive data:");
    }

    // Next TX is scheduled after TX_COMPLETE event.
}

```

En la Figura 21 se observa la función `onEvent` que posee una serie de eventos que pueden ocurrir al usar el módulo Lora. Dentro del cual se debe destacar el evento que se muestra en la Figura 22 se ejecuta luego de una transmisión realizada, en donde se abre una ventana para recepción y realiza el proceso para decodificarla y asignarla a la variable "valor\_recibido".

### Figura 21

*Extracto de la función `onEvent`*

```

void onEvent (ev_t ev) {
    Serial.print(os_getTime());
    Serial.print(": ");
    Serial.println(ev);
    switch (ev) {
        case EV_SCAN_TIMEOUT:
            Serial.println(F("EV_SCAN_TIMEOUT"));
            break;
        case EV_BEACON_FOUND:
            Serial.println(F("EV_BEACON_FOUND"));
            break;
        case EV_BEACON_MISSED:
            Serial.println(F("EV_BEACON_MISSED"));
            break;
    }
}

```

**Figura 22***Evento EV\_TXCOMPLETE*

```

case EV_TXCOMPLETE:
  Serial.println(F("EV_TXCOMPLETE (includes waiting for RX windows)"));
  if (LMIC.dataLen) {
    // data received in rx slot after tx
    Serial.print(F("Data Received: "));
    Serial.println(LMIC.dataLen);
    Serial.println();
    uint8_t result;
    if(LMIC.dataLen==1) //Verifica si el mensaje de downlink es 1 byte
    {
      result = LMIC.frame[LMIC.dataBeg + 0];
      valor_recibido=result;           //recibe el mensaje de downlink
    }
  }
  // Schedule next transmission
  os_setTimedCallback(&sendjob, os_getTime() + sec2osticks(TX_INTERVAL), do_send);
  break;

```

En la Figura 23 se muestra la función flujo que realiza el proceso de acumular los pulsos generados por el sensor de flujo.

**Figura 23***Función flujo*

```

void flujo()
{
    pulsos++; //Incrementa en una unidad el número de pulsos
}

```

En la Figura 24 se muestra un extracto de la función setup() , donde se lee el valor de metros cúbicos almacenados en la EEPROM. Se estable los pines de Arduino (control, válvula y sensor) como salida y entrada respectivamente. Luego se establece el estado inicial de control, válvula en "LOW" (0), se inicializa el LCD y se comprueba que esté conectado el módulo RTC. Por último, se habilita las interrupciones para calcular el consumo de agua y se inicializa el módulo LoRa

**Figura 24**

### Extracto de la función `setup()`

```

void setup() {
  Serial.begin(9600);
  metros_cubicos=EEPROM.get(0,metros_cubicos); //leer el último valor almacenado
  pinMode(control_valvula,OUTPUT);
  pinMode(sensor, INPUT_PULLUP); // Pin digital como entrada con conexión PULL-UP interna
  digitalWrite(control_valvula,LOW); //Estado inicial
  lcd.begin(20,4); // Inicializo el LCD 20X4
  if (! rtc.begin()) { // si falla la inicializacion del modulo
    Serial.println("Modulo RTC no encontrado !"); // muestra mensaje de error
    while (1); // bucle infinito que detiene ejecucion del programa
  }
  interrupts(); // Habilito las interrupciones
  // Interrupción INTO, llama a la ISR llamada "flujo" en cada flanco de subida en el pin digital 2
  attachInterrupt(digitalPinToInterrupt(sensor), flujo, RISING);
  tiempoAnterior = millis(); // Guardo el tiempo que tarda el ejecutarse el setup
  Serial.println("Conexión con TTN y envío de datos a Ubidots");
#ifdef VCC_ENABLE
  // For Pinoccio Scout boards
  pinMode(VCC_ENABLE, OUTPUT);
  digitalWrite(VCC_ENABLE, HIGH);
  delay(1000);
#endif
  // LMIC init
  os_init();
}

```

En la Figura 25 se muestra un extracto de la función `loop()`, donde se realiza cálculo del caudal y consumo en litros del agua obtenidos del sensor de flujo. También se establece que cada 5 min se guarde el consumo de agua en la memoria EEPROM con el objetivo de obtener un respaldo en caso de pérdida de energía. Los datos calculados como el caudal y consumo de agua son mostrados mediante la función `LCD_2004()`.

### Figura 25

#### Extracto 1 de la función `loop()`

```

void loop() {
  DateTime fecha = rtc.now();
  if(millis() - tiempoAnterior > 1000)
  {
    // Realizo los cálculos
    tiempoAnterior = millis(); // Actualizo el nuevo tiempo
    pulsos_Acumulados += pulsos; // Número de pulsos acumulados
    litros_Hora = (pulsos * 60 / 7.5); // Q = frecuencia * 60/ 7.5 (L/Hora)
    litros = pulsos_Acumulados*1.0/450; // Cada 450 pulsos son un litro
    metros_cubicos=metros_cubicos+litros*0,001; //Convertir de litros a metros cúbicos
    if(fecha.minute()%5 ==0 && fecha.second()==1) //identifica si los minutos son un múltiplo de 5 min
    {
      EEPROM.put(0,metros_cubicos); //Guardar el valor en memoria
      lcd.clear();
      lcd.print("Guardando dato...");
      delay(3000);
    }
    pulsos = 0; // Pongo nuevamente el número de pulsos a cero

    // Llamada a la función que muestra los resultados en el LCD 2004
    LCD_2004();
  }
}

```

En la Figura 26 se establece que si la variable “valor\_recibido=48” se cierra la electroválvula y si “valor\_recibido=49” se abre la electroválvula. Por último, se añade un reseteo automático cada primer día del mes a las 8:00:00 para evitar que la variable “metros\_cubicos” sobrepase el dimensionamiento programado.

### Figura 26

*Extracto 2 de la función loop().*

```

if(valor_recibido==48){
  //Cierra la electroválvula
  digitalWrite(control_valvula,LOW);
}
if (valor_recibido==49){
  //Abre la electroválvula
  digitalWrite(control_valvula,HIGH);
}
//Reseteo mensual
if (fecha.day()==1 && fecha.hour()==8 && fecha.minute()==0 && fecha.second()==0)
{
  metros_cubicos=0;
  delay(2000);
}
os_runloop_once();

```

En la Figura 27 se muestra el contenido de la función LCD\_2004 que se encarga de mostrar en el LCD20x4, una cabecera con el nombre del sensor usado, la fecha y hora, el caudal y metros cúbicos de agua que se consume

Figura 27

Función LCD\_2004

```

void LCD_2004()
{
    // Imprimo la cabecera del sistema
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(">CAUDALIMETRO YF-S201");
    lcd.setCursor(0,1);
    //Fecha y hora actual
    DateTime fecha = rtc.now();
    lcd.print(fecha.day(),DEC);
    lcd.print("/");
    lcd.print(fecha.month(),DEC);
    lcd.print("/");
    lcd.print(fecha.year(),DEC);
    lcd.print(" ");
    lcd.print(fecha.hour(),DEC);
    lcd.print(":");
    lcd.print(fecha.minute(),DEC);
    lcd.print(":");
    lcd.print(fecha.second(),DEC);
    lcd.setCursor(0,2);
    // Imprimo el caudal en L/hora
    lcd.print("-> ");
    lcd.print(litros_Hora, DEC);
    lcd.print(" L/Hora");
    lcd.setCursor(0,3);
    // Imprimo el número de litros acumulados
    lcd.print("-> ");
}

```

En la Figura 28 se muestra el contenido de la función envio(), encargada de convertir los datos del caudal y consumo de agua a una cantidad entera y posteriormente realizar un desplazamiento binario para asignarlo a cada posición mydata cada una de 1byte.

Figura 28

### *Función envio*

```

void envio()
{
  //variables para codificar los datos a enviar mediante lora
  int16_t caudal_agua;
  int16_t metros_3;
  Serial.print(F("NO."));
  Serial.print(count);
  Serial.println(F("      #####"));
  Serial.print("Caudal actual:");
  Serial.println(litros_Hora);
  Serial.print("Consumo total en m^3");
  Serial.println(litros);
  Serial.println(F("#####   "));
  count++;
  //Se convierte los datos de decimales a enteros
  caudal_agua = litros_Hora;
  metros_3=metros_cubicos*100;
  mydata[0] = caudal_agua>>8;
  mydata[1] = caudal_agua;
  mydata[2] = metros_3>>8;
  mydata[3] = metros_3;
}

```

### *Proceso de calibración*

Para la calibración del prototipo se necesitó asistir al laboratorio de la empresa Hidrocentro, el cual es especializado para calibrar medidores de agua potable. Este laboratorio cuenta con equipos que permiten modificar la presión y la cantidad de litros de agua que circularán por el prototipo. Además, cuentan con tanques de 20L y 100L, donde se acumula el agua usada en las pruebas. Por último, posee medidores patrones para comparar las medidas con el prototipo.

**Figura 29**

Laboratorio de la empresa Hidrocentro



Figura 30

Equipo que controla la cantidad de litros de agua que se suministra al prototipo



Figura 31

*Tanque de agua de 20L*



**Figura 32**

*Barómetro para medir la presión del agua*

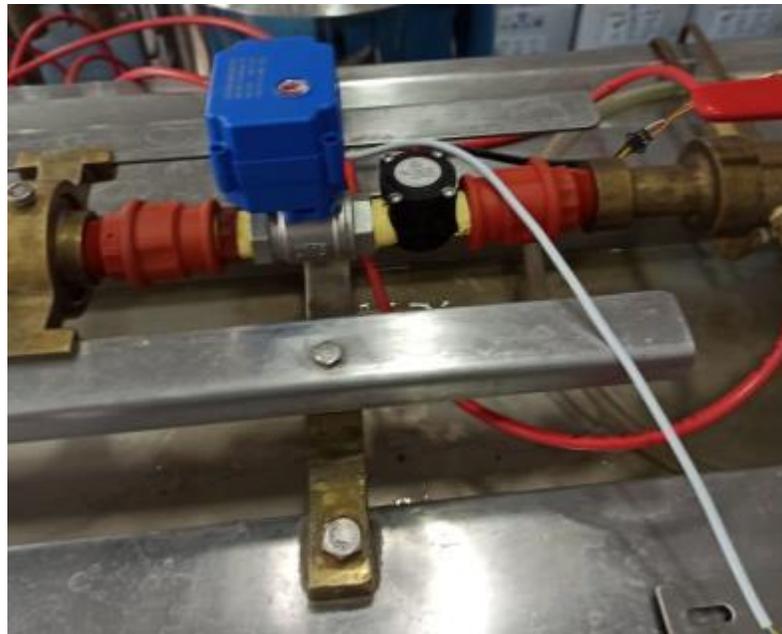


### *Pasos para calibrar el prototipo*

El primer paso es colocar al prototipo unos adaptadores de  $\frac{3}{4}$  a  $\frac{1}{2}$  con la finalidad de facilitar la conexión del prototipo a la tubería del laboratorio. Esto se puede observar en la Figura 33.

### **Figura 33**

*Prototipo conectado a la tubería del laboratorio*



Segundo paso, se planificó realizar 15 pruebas, en donde se hace circular 20 litros de agua en cada una de ellas. El objetivo de estas pruebas es obtener un promedio de la cantidad de pulsos que genera un litro de agua. Para ello, luego de finalizar una prueba se anota en una tabla la cantidad de pulsos generados por los 20 litros de agua.

Tercer paso, se resetea el Arduino para encerrar las variables del consumo de agua, caudal y la cantidad de pulsos. Luego se depura el agua del tanque para volver a usar la misma

agua en la siguiente prueba. Cabe recalcar que el agua no se desperdicia en el transcurso de las pruebas realizadas.

Cuarto paso, es realizar el paso 2 y 3 hasta finalizar las 15 pruebas planteadas al inicio. Es decir, hasta obtener 15 mediciones del número de pulsos generados por los 20 litros de agua. En la Tabla 5 se muestra los datos obtenidos en las 15 pruebas realizadas. Para obtener la cantidad de pulsos en 1L de agua se divide la cantidad de pulsos en 20L de agua para 20.

**Tabla 5**

*Cantidad de pulsos obtenidas por el prototipo en 20 L de agua*

<b>Prueba N°</b>	<b>Cantidad de pulsos en 20 L</b>	<b>Cantidad de pulsos en 100 L</b>
1	8604	430,20
2	8728	436,40
3	8633	431,65
4	8685	434,25
5	8775	438,75
6	8616	430,80
7	8878	443,90
8	8717	435,85
9	8867	443,35
10	8611	430,55
11	8850	442,50
12	8856	442,80
13	9142	457,10
14	8826	441,30
15	8585	429,25
<b>Promedio del # de pulsos en 1L</b>		<b>437,91</b>

Como se puede observar en la Tabla 5 el promedio de la cantidad de pulsos que genera 1L de agua es 437,91. Este valor se lo modifica en el script de Arduino para calibrar al prototipo.

Una vez cargado el nuevo script en el Arduino se procede a realizar una prueba con 20L de agua obteniendo  $0.020\text{m}^3=20\text{L}$ , esto se puede en la Figura 34.

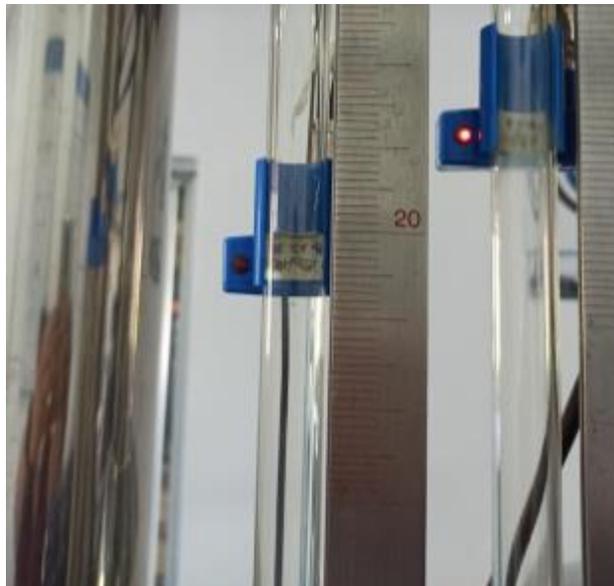
**Figura 34**

*Medida en metros cúbico obtenida por el prototipo cuando circularon 20 L de agua*



**Figura 35**

*Sensor óptico en 20 litros de agua*



**Figura 36**

*Medidor patrón en 20L de agua*



Finalmente se procedió a realizar una prueba con 100 L de agua obteniendo en el prototipo  $0.099\text{m}^3=99\text{L}$ , esto se puede observar en la Figura 37.

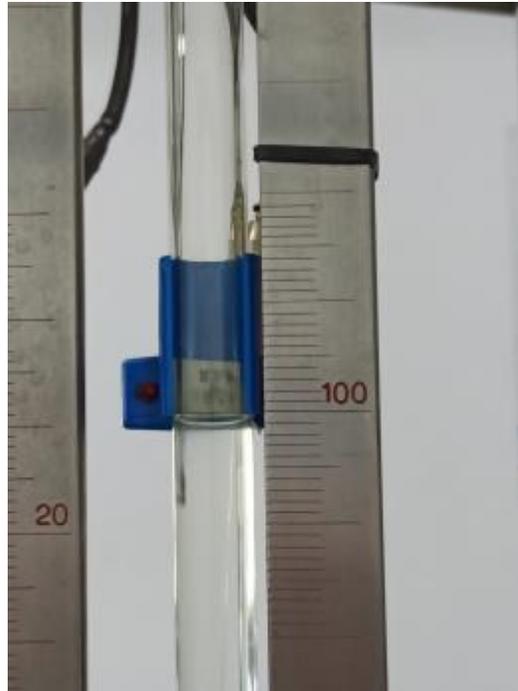
**Figura 37**

*Medida en metros cúbico obtenida por el prototipo cuando circularon 100 L de agua*



**Figura 38**

*Sensor óptico en 100 litros de agua*



#### *Cálculo de la duración de la batería*

La duración de la autonomía del dispositivo dependerá de la carga de la batería. Para el proyecto se utilizó una batería recargable de 4000 mAh a 5VDC. Mediante la Ecuación 1 se puede determinar la duración del sistema funcionando solamente con el banco de batería

$$Duración\ de\ la\ batería\ (h) = \frac{capacidad\ de\ la\ batería\ [mAh]}{corriente\ de\ carga\ [mA]} \quad (1)$$

Para determinar la corriente de carga del dispositivo, se utilizó una pinza amperimétrica para conocer la corriente que circula en el equipo cuando se está enviando un dato y la válvula está activada. De la Figura 39 se obtiene el valor de 400mA como corriente de carga del dispositivo.

**Figura 39**

*Valor de la corriente de carga del dispositivo*



Ya con todos los valores, se puede determinar la duración de la batería, reemplazando los mismo en la **¡Error! No se encuentra el origen de la referencia..**

$$\text{Duración de la batería (h)} = \frac{4000 \text{ [mAh]}}{400 \text{ [mA]}}$$

$$\text{Duración de la batería (h)} = 10 \text{ h}$$

Cabe mencionar que este resultado es un estimado y se lo considera bajo condiciones ideales, es decir, que la batería este cargada al 100% y que los dispositivos internos no estén averiados.

## *Implementación*

### *Ensamblado del sistema*

Para la instalación del sistema se utilizó una caja plástica de polietileno resistente al agua para trabajos industriales y que estén en exteriores. Dentro de la caja se procedió en primer lugar a colocar los elementos que integran el sistema de control para determinar la mejor distribución de los mismos tal y como se indica en la Figura 40.

#### **Figura 40**

*Distribución de elementos del sistema en la caja.*



Una vez determinada la distribución de los elementos se procedió a perforar la caja para colocar la tubería plástica que permitirá el tránsito del agua hacia la residencia desde el medidor de agua potable instalado por la empresa pública EMMAPS. Fuera de la caja se instaló dos uniones universales de media pulgada que permitirá la conexión del sistema a la tubería interna del domicilio. La Figura 41 presenta la tubería instalada dentro de la caja de proyectos.

#### **Figura 41**

*Instalación de tubería dentro de la caja del proyecto.*



Al igual que la tubería, se realizó una perforación para colocar el LCD 20x04 en la parte frontal de la caja para facilitar la visualización de los datos de consumo, tal y como se puede observar en la Figura 42.

**Figura 42**

*LCD 20x04 colocado en la caja de proyectos*



Como siguiente paso se colocaron los elementos electrónicos restantes como el Arduino UNO y su shield LoRa, el relé, el circuito de reloj y la batería de respaldo para dotar de

autonomía al sistema en caso de falla o corte de energía en el domicilio. La colocación de los elementos se puede observar en la Figura 43.

**Figura 43**

*Distribución de los elementos electrónicos en la caja de proyectos*



Ya con los elementos ensamblados se procedió a instalar el prototipo en la tubería interna del domicilio. En este punto además de la tubería se debe asegurar una conexión eléctrica de los elementos para el funcionamiento del dispositivo, así como la recarga continua de una batería. La instalación del sistema en la red de agua potable del domicilio se puede observar en la Figura 44.

**Figura 44**

*Prototipo instalado en la tubería de agua potable del domicilio.*



Al mismo tiempo se debe ubicar el gateway LoRa con línea de vista con el sistema de control y consumo. Para el gateway se debe asegurar una fuente de alimentación continua de energía y salida al Internet, sea esta de forma inalámbrica o alámbrica. Para este proyecto se utilizó una conexión ethernet para la salida al Internet. La Figura 45, muestra la instalación eléctrica del gateway LoRa.

**Figura 45**

*Conexión e instalación inalámbrica del gateway LoRa*



La Figura 46 muestra como quedó finalmente la conexión de los módulos que conforman el sistema de consumo de agua potable para el domicilio. La instalación del Gateway

se lo realizó en el techo de las gradas de acceso al domicilio para evitar posible manipulación o acceso a personal no calificado y el módulo de transmisión y recepción se encuentra en la entrada sobre unos soportes de madera que se observan en la Figura 44.

**Figura 46**

*Instalación final del sistema de control de consumo de agua potable.*



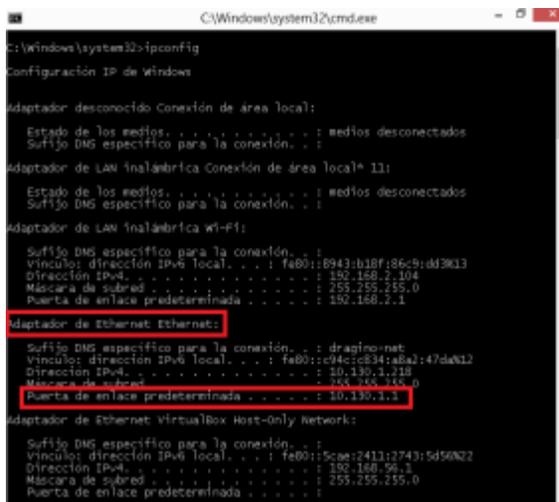
Finalmente, con los equipos instalados se puede ya proceder a la configuración del gateway y su aplicación en Ubidots.

## Configuración del Gateway LoRa

Se conecta el cable de red a la interfaz LAN del Dragino (Gateway Lora) y el otro extremo al computador. Para la configuración se ingresa al cmd y se digita el comando "ipconfig", para obtener la dirección IP del Dragino. Esto se puede observar en la Figura 47.

**Figura 47**

*Dirección IP del Dragino (Gateway Lora)*



```

C:\Windows\system32\cmd.exe
C:\Windows\system32>ipconfig

Configuración IP de Windows

Adaptador desconocido Conexión de área local:
    Estado de los medios . . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . . :

Adaptador de LAN inalámbrica Conexión de área local* 11:
    Estado de los medios . . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . . :

Adaptador de LAN inalámbrica Wi-Fi:
    Sufijo DNS específico para la conexión. . . :
    Vínculo dirección IPv6 local . . . : f480::0943:b18f:86c9:d3013
    Dirección IPv4 . . . . . : 192.168.2.104
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . : 192.168.2.1

Adaptador de Ethernet Ethernet:
    Sufijo DNS específico para la conexión. . . : dragino-net
    Vínculo dirección IPv6 local . . . : f480::c94c:c34:a8a2:47da612
    Dirección IPv4 . . . . . : 10.130.1.218
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . : 10.130.1.1

Adaptador de Ethernet VirtualBox Host-Only Network:
    Sufijo DNS específico para la conexión. . . :
    Vínculo dirección IPv6 local . . . : f480::5cae:2411:2743:5d5022
    Dirección IPv4 . . . . . : 192.168.56.1
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . :
  
```

Esta dirección IP, se digita en el navegador para poder ingresar a la interfaz de usuario del dispositivo y así poder configurarlo.

**Figura 48**

*Interfaz de usuario del Dragino*

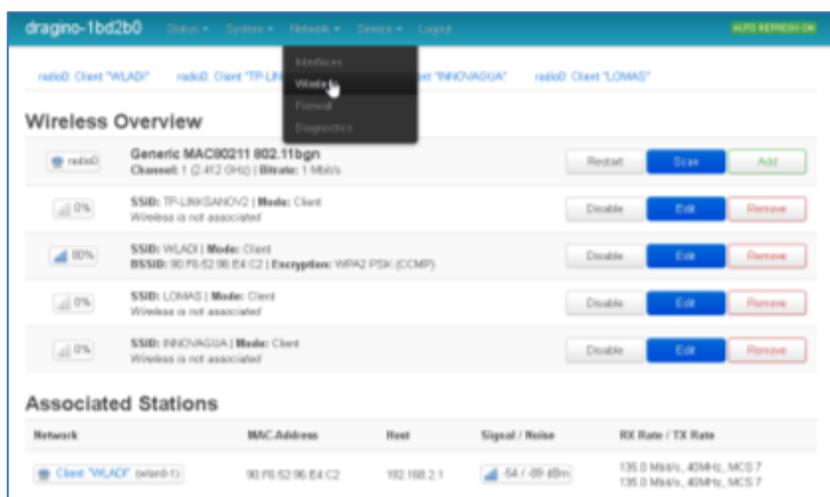


Una vez dentro de la interfaz de usuario del dispositivo, se realiza la siguiente configuración:

- Vincular el Gateway Lora al router del domicilio, para que los datos recibidos por la interfaz de radio frecuencia sean enviados al Internet. Para ello, se debe ir a la pestaña Network opción Wireless y dar clic en el botón Scan. Esto provoca que el dispositivo comience a buscar las redes Wireless cercanas. Luego de identificar la red objetivo del domicilio se da clic en “Join Network”, se ingresa la contraseña de la red Wireless y se da clic en el botón Submit.

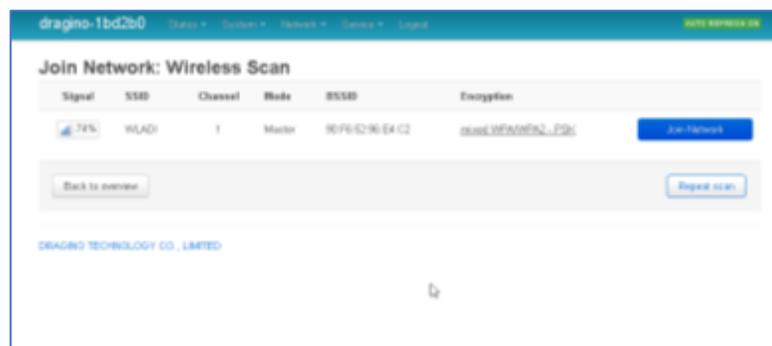
**Figura 49**

*Pestaña “Network” opción “Wireless”*



**Figura 50**

*Proceso de “Scan” para identificar las redes Wireless cercanas*



**Figura 51**

*Contraseña de la red Wireless*

The screenshot shows the 'Joining Network: "WLADI"' configuration page in the dragnio-1bd2b0 interface. The page has a blue header with navigation links: Home, Settings, Network, Services, and Logout. The main content area is titled 'Joining Network: "WLADI"' and includes the following sections:

- Replace wireless configuration:** A checkbox that is checked, with a note: 'Check this option to delete the existing network from this node.' Below it is a 'WPA passphrase' field with a dropdown arrow.
- Name of the new network:** A text input field containing 'wladi'. A note below states: 'The allowed characters are: A-Z, a-z, 0-9 and \_'.
- Create / Assign firewall zone:** A dropdown menu showing 'none', 'wan', 'LoRa', 'lora', and 'MICHUAGA'. A note below says: 'Choose the firewall zone you want to assign to this interface. Select unspecified to remove the interface from the associated zone or fill out the create field to define a new zone and attach the interface to it.'

At the bottom, there are two buttons: 'Back to your results' and 'Save'. The footer reads 'DRAGNO TECHNOLOGY CO., LIMITED'.

- Configurar el acceso del dispositivo al servidor de red The Things Network (TTN). Para realizar este procedimiento se debe ingresar a la pestaña "Service" en la opción "LoRaWAN Gateway". Una vez dentro realiza la configuración detallada en la Figura 52.

**Figura 52**

*Configuración del servidor de red*

The screenshot shows the 'LoRa Gateway Settings' configuration page in the dragnio-1bd2b0 interface. The page has a blue header with navigation links: Home, Settings, Network, Services, and Logout. The main content area is titled 'LoRa Gateway Settings' and includes the following sections:

- LoRaWAN Server Settings:** A list of configuration fields:
  - IoT Service:** A dropdown menu set to 'LoRaWAN/LoRa Gateway'.
  - Debug Level:** A dropdown menu set to 'LoRa message output'.
  - Service Provider:** A dropdown menu set to 'The Things Network'.
  - Server Address:** A dropdown menu set to 'th-mcu-gw-west'.
  - Server Port:** A text input field containing '1701'.
  - Gateway ID:** A text input field containing '460411u2005'.
  - Mail Address:** A text input field containing 'dragno\_tech@protonmail.com'.
  - Latitude:** A text input field containing '20.73'.
  - Longitude:** A text input field containing '114.25'.
  - RadioMode:** A dropdown menu set to 'A for 01, 0 for 01'.
  - Radio Power (dBm):** A text input field containing 'comp 5 + 20.000'.

**Figura 53**

*Configuración del canal de recepción*

The screenshot displays the configuration page for 'dragino-1bd2b0'. At the top, there are navigation tabs: 'Status', 'System', 'Network', and 'Service'. Below these, the 'Radio Power (Unit: dBm)' is set to 'range 5 ~ 20 dBm'. The main section is titled 'Channel 1 Radio Settings' and includes the following parameters:

- RadioA Frequency (Unit: Hz): 915100000
- RadioA Spreading Factor: SF7
- RadioA Coding Rate: 4/5
- RadioA Signal Bandwidth: 125 kHz
- RadioA Preamble Length: 8 (with a note: Length range: 6 ~ 65536)
- RadioA LoRa Sync Word: 52 (with a note: Value 52(0x34) for LoRaWAN)
- Encryption Key: Encryption Key

**Figura 54**

*Configuración del canal de transmisión*

The screenshot displays the configuration page for 'dragino-1bd2b0'. At the top, there are navigation tabs: 'Status', 'System', 'Network', 'Service', and 'Logout'. Below these, the 'Encryption Key' is set to 'Encryption Key'. The main section is titled 'Channel 2 Radio Settings' and includes the following parameters:

- RadioB Frequency (Unit: Hz): 923300000
- RadioB Spreading Factor: SF9
- RadioB Coding Rate: 4/5
- RadioB Signal Bandwidth: 125 kHz
- RadioB Preamble Length: 8 (with a note: Length range: 6 ~ 65536)
- RadioB LoRa Sync Word: 52 (with a note: Value 52(0x34) for LoRaWAN)
- Encryption Key: Encryption Key

### Procedimiento para registrar el Gateway LoRa en TTN

Se ingresa a la pestaña “Console” y en luego a “Gateway” como se observa en la Figura 55.

**Figura 55**

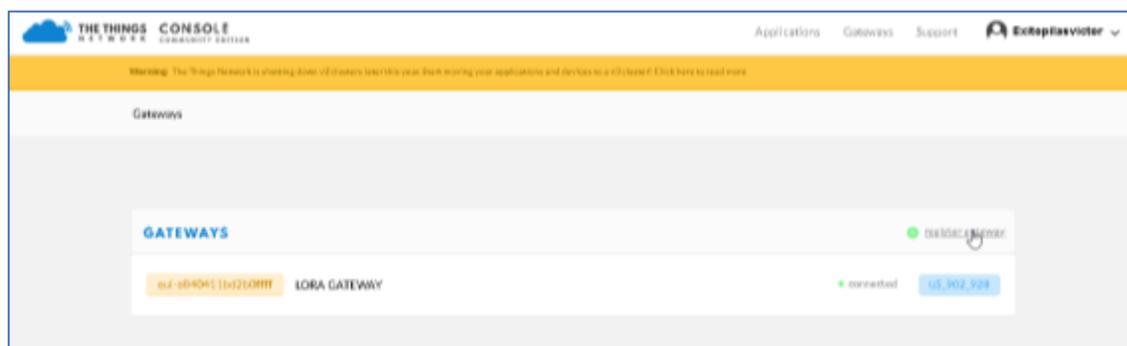
Consola de TTN



Luego dar clic en “register gateway”.

**Figura 56**

Registro del Gateway Lora en TTN

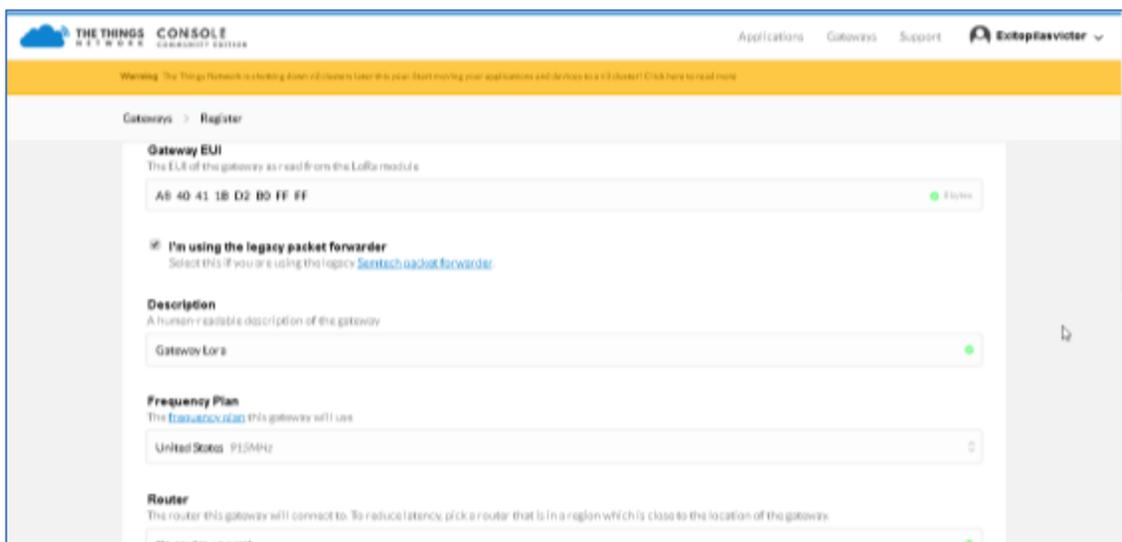


Para registrar el Gateway Lora en TTN se ingresa en Gateway EUI la dirección MAC del dispositivo, en Description se coloca un nombre para identificar al Gateway, en Frequency Plan

se escoge United States 915 MHz y en Router se selecciona ttn-router-us-west. Finalmente dar clic en el botón Register Gateway.

**Figura 57**

*Campos necesarios para registrar el Gateway Lora en TTN*

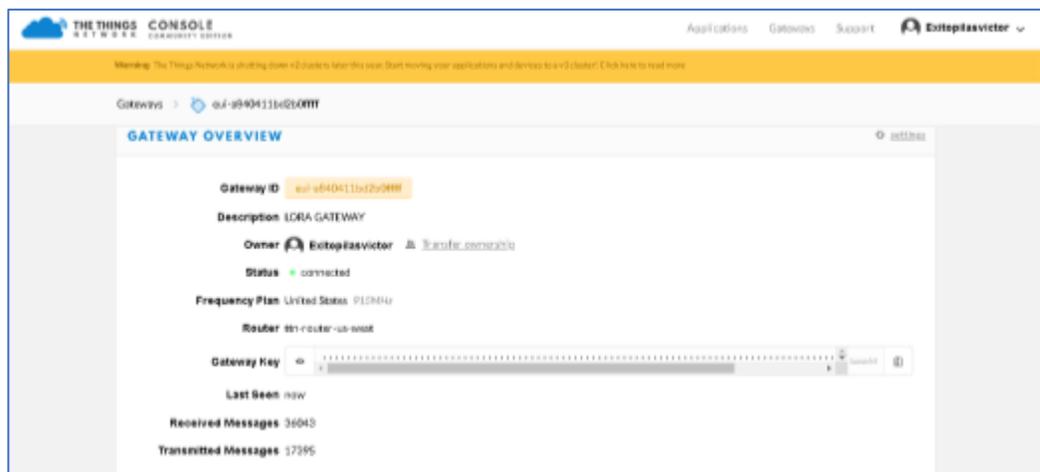


The screenshot shows the 'Register Gateway' form in the TTN Console. The form includes the following fields and options:

- Gateway EUI:** A text input field containing 'AS 40 41 1B D2 B5 FF FF' with a green checkmark icon.
- I'm using the legacy packet forwarder:** A checked checkbox with a link to 'Semtech packet forwarder'.
- Description:** A text input field containing 'Gateway Lora' with a green checkmark icon.
- Frequency Plan:** A dropdown menu showing 'United States 915MHz'.
- Router:** A dropdown menu showing 'ttn-router-us-west' with a green checkmark icon.

**Figura 58**

*Estado de conexión del Gateway Lora*



The screenshot shows the 'Gateway Overview' page for a specific gateway. The gateway is identified by the ID 'eui-a840411d42b900ff'. The page displays the following information:

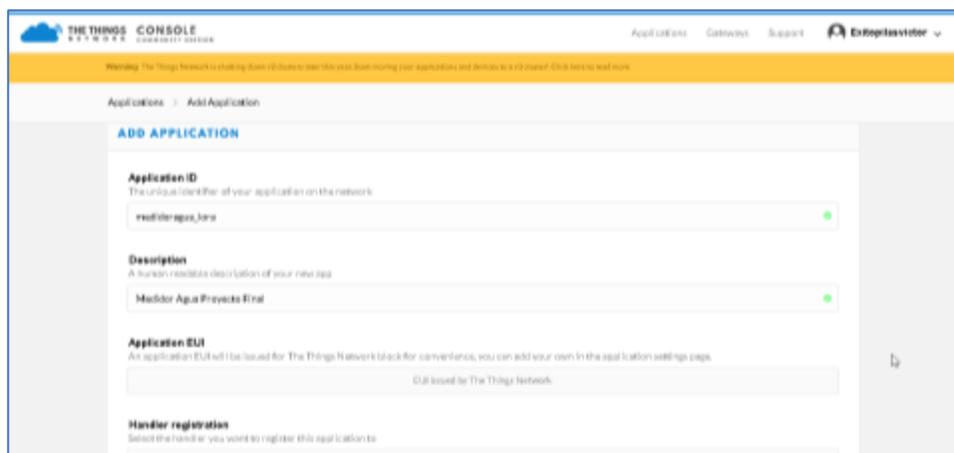
- Gateway ID:** eui-a840411d42b900ff
- Description:** LORA GATEWAY
- Owner:** Ecitepliasvictor
- Status:** connected
- Frequency Plan:** United States 915MHz
- Router:** ttn-router-us-west
- Gateway Key:** A masked field with a 'Show/Hide' button.
- Last Seen:** now
- Received Messages:** 36043
- Transmitted Messages:** 17395

### Creación de una aplicación en TTN

Luego de registrar el Gateway Lora en TTN, se procede a crear una aplicación. Para ello se debe ingresar a la pestaña “Applications” y luego en “add application”. A continuación, se le asigna un Application ID, el cuál debe ser único para cada aplicación que se desee crear, en Description se coloca un identificador para la aplicación y finalmente dar clic en el botón add application.

#### Figura 59

##### Creación de una aplicación en TTN



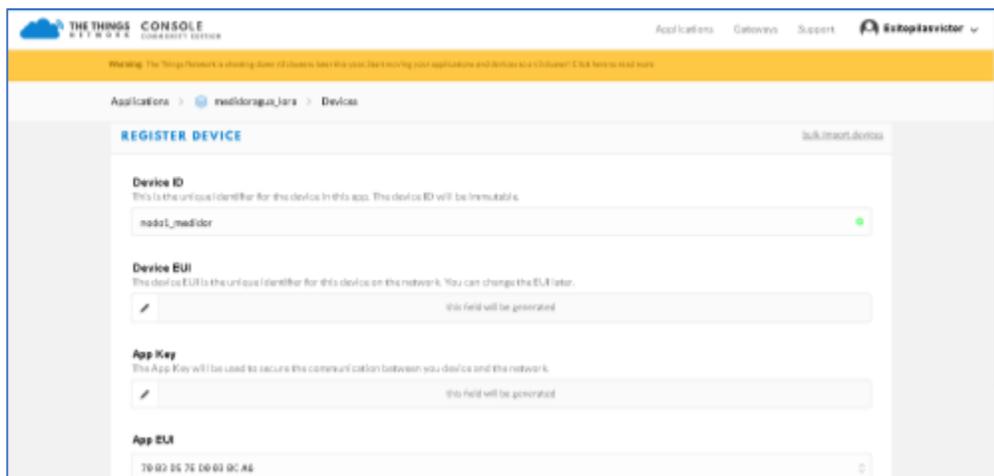
The screenshot shows the 'Add Application' page in the TTN Console. The page has a yellow header with the TTN logo and navigation links for 'Applications', 'Gateways', and 'Support'. Below the header, there is a breadcrumb trail 'Applications > Add Application' and a blue 'ADD APPLICATION' button. The form contains the following fields:

- Application ID:** The unique identifier of your application on the network. The value entered is 'mudbor\_agua\_lore'.
- Description:** A human-readable description of your new app. The value entered is 'Mudbor Agua Proyecto Final'.
- Application EUI:** An application EUI will be issued for The Things Network back for convenience, you can add your own in the application settings page. The value entered is 'EUI based by The Things network'.
- Handler registration:** Select the handler you want to register this application to.

Dentro de la aplicación creada se debe registrar el nodo final, para ello se dirige a la opción “register device”. En Device ID se le asigna un nombre que permita identificar de manera única al nodo final, luego dar clic en generate Device EUI y App Key, para que TTN genere estos campos automáticamente. Finalmente dar clic en Register.

#### Figura 60

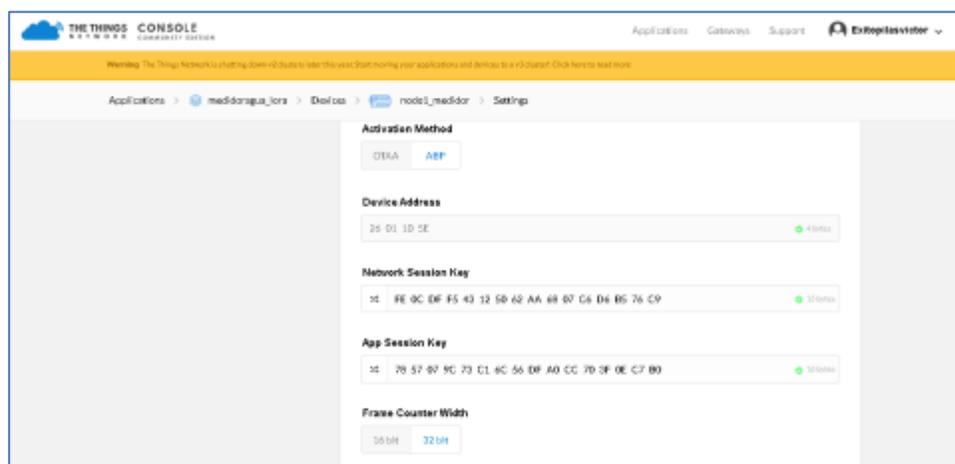
### Registro del nodo final dentro de la aplicación creada



Con el nodo final registrado, se debe dirigir a la pestaña “Settings” y seleccionar como método de activación ABP. A continuación, TTN genera automáticamente los campos Device Address, Network Session Key y App Session Key. Finalmente dar clic en el botón “save” para guardar los cambios realizados.

**Figura 61**

*Método de activación ABP del nodo final*

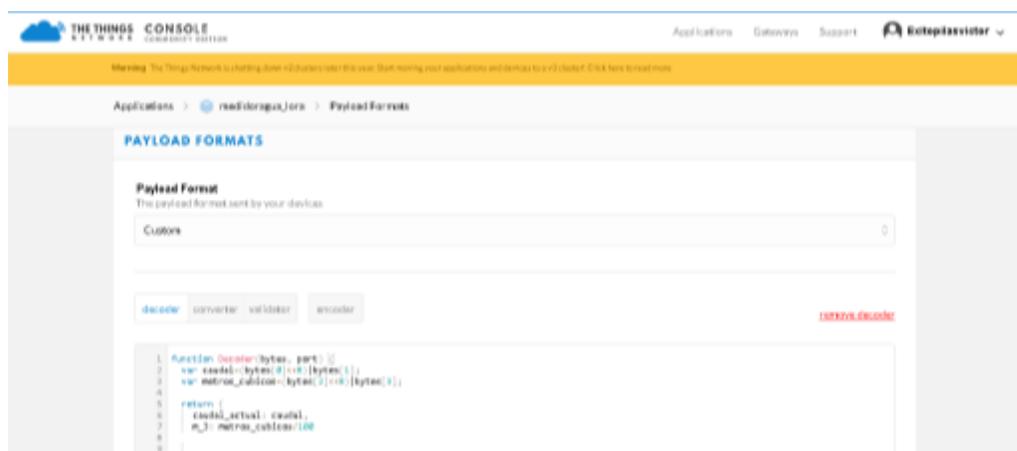


### *Personalización del formato del mensaje recibido en TTN*

Para personalizar el formato de los mensajes recibidos, se debe acceder a la aplicación creada y luego a Payload Formats. Posteriormente se crea una función para decodificar los mensajes recibidos y convertirlo en un formato entendible por el usuario, para ello se realiza el proceso inverso empleado en el Arduino. Este proceso se puede observar en la **¡Error! No se encuentra el origen de la referencia.62.**

**Figura 62**

*Función que decodifica los mensajes recibidos en TTN*

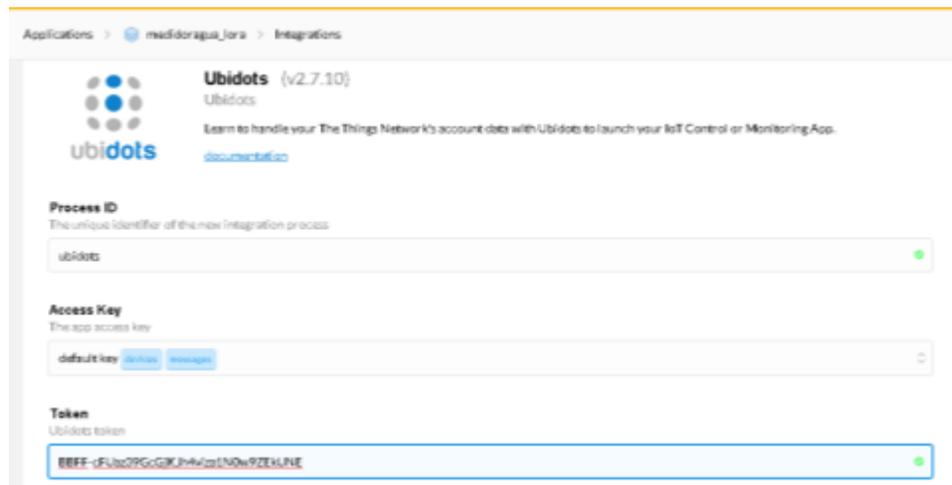


### *Procedimiento para integrar Ubidots con TTN*

Para ello en la aplicación creada en TTN, se debe ingresar en la pestaña “Integrations”, luego en “add integration” y seleccionar Ubidots. Una vez dentro se le asigna un identificador único al Process ID, en Access Key se seleccionan la opción por defecto y en Token se coloca el Token generado por Ubidots. Finalmente dar clic en el botón add integration como se observa en la Figura 63.

**Figura 63**

## Integración de Ubidots con TTN

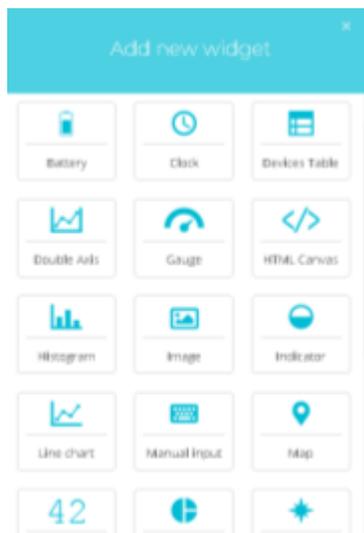


### Creación de la aplicación en Ubidots

En Ubidots se debe agregar widgets adecuados para la visualización de los datos. Por lo tanto, se debe dar clic en el botón “+” y se abre un menú con todos los widgets disponibles como se puede observar en la Figura 64.

**Figura 64**

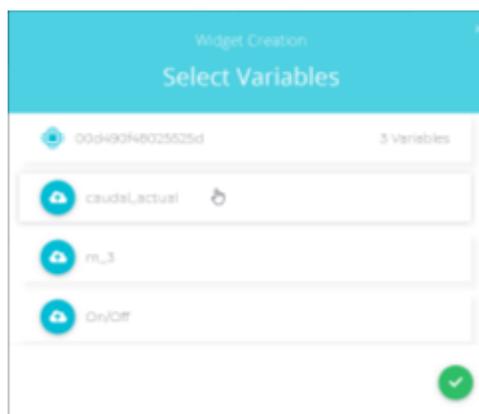
*Widgets disponibles en Ubidots*



Se selecciona para la variable del caudal de agua el widget Gauge, luego se da clic en el botón “add variables”. Finalmente seleccionar el dispositivo y la variable que se desea representar como se observa en la Figura 65.

**Figura 65**

*Agregación de widgets en Ubidots*

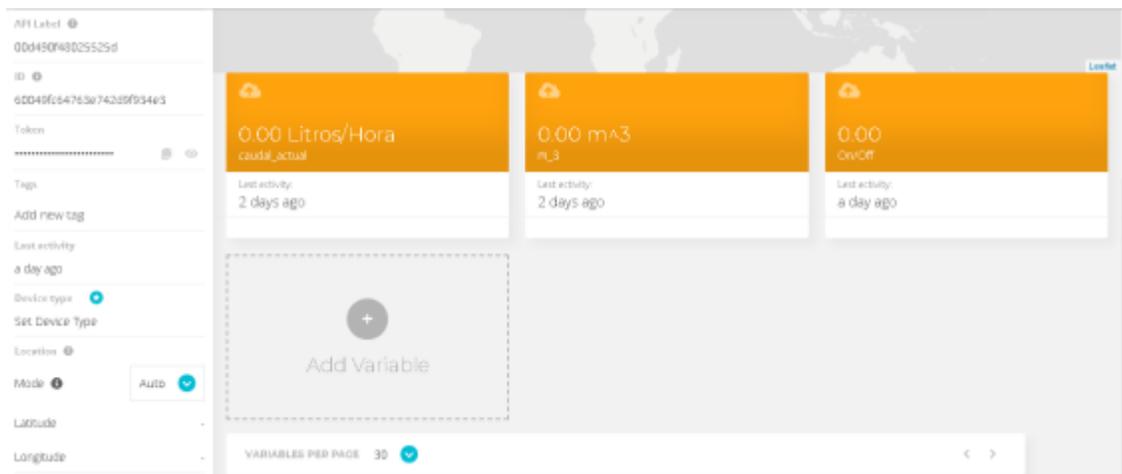


Este proceso se realiza para la otra variable que representa el consumo de agua, para el cual se utilizó el widget tank para su representación. Además, se agrega widgets que grafican las variables de caudal y consumo respecto al tiempo.

Por último, se procede a crear una variable cuya función es controlar la apertura y cierre de la electroválvula mediante mensajes enviados desde el Gateway Lora al nodo final. Para llevar a cabo esto se procede a ingresar a Device, luego seleccionar el dispositivo y dar clic en “add variable”. Esta variable es del tipo “raw”, luego se le asigna el nombre de “On/Off” como se puede observar en la Figura 66.

**Figura 66**

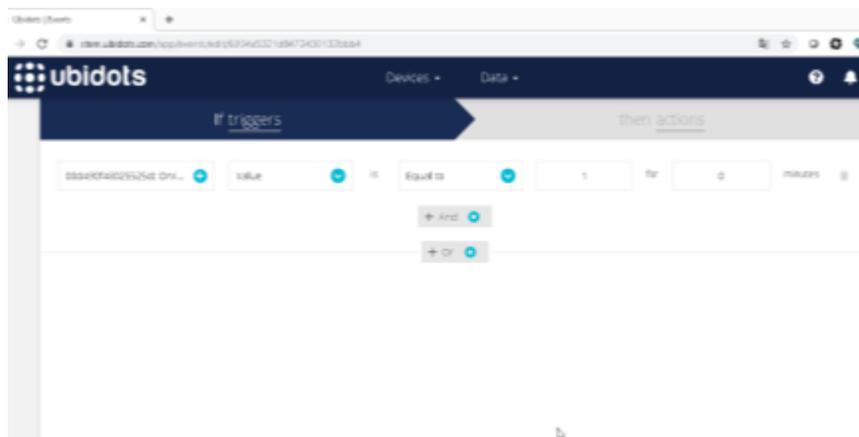
### Variable On/Off



Posteriormente se debe ingresar en Data, luego en events y dar clic en “new event”. Este evento se ejecutará cuando se presione el botón apertura/cierre de la electroválvula en Ubidots. Por lo tanto, se selecciona la variable “On/Off” anteriormente creada y se lo compara con 1 (On) y en el tiempo se coloca 0 minutos como se puede observar en la Figura 67.

**Figura 67**

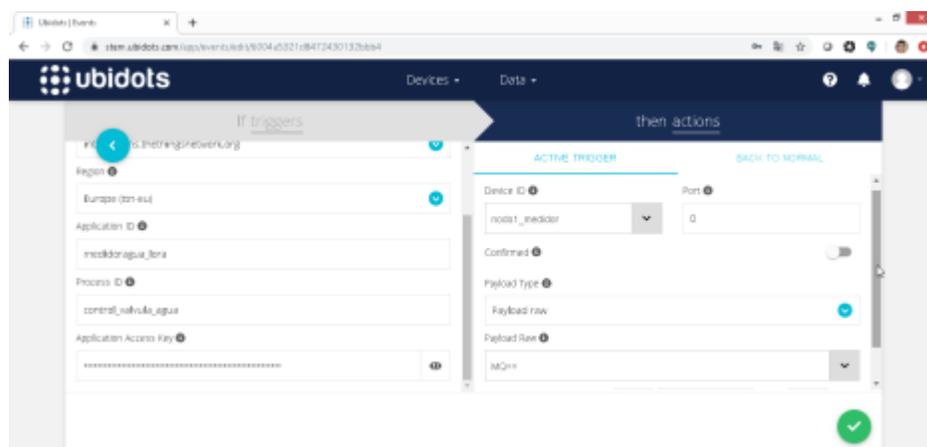
*Programar la ejecución del evento cuando cambie el estado de la variable On/Off*



En la parte de las acciones que se deben ejecutar cuando se lance el evento, se debe seleccionar la opción TTN Downlink. Una vez dentro se rellena los campos de Application ID, Process ID, Application Access Key y Device ID. Estos campos son proporcionados por TTN, en port se coloca “0” y en payload raw se coloca “MQ==” que representa un “1” codificado en base 64. Cuando el botón regrese a su estado inicial, es decir cerrar la electroválvula se coloca “MA==” que representa “0” codificado en base 64. Esto se puede observar en la Figura 68.

**Figura 68**

*Acciones que se realizan cuando se ejecuta el evento*

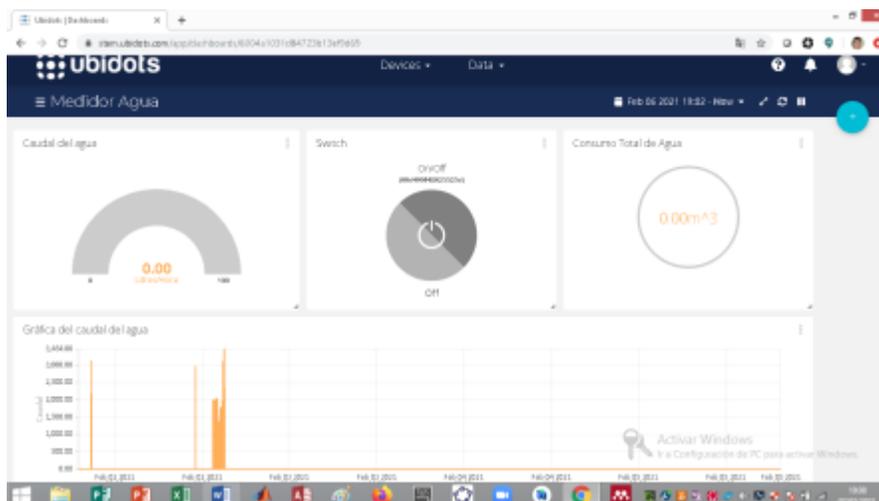


Se lleva a cabo los mismos pasos para crear un evento, pero en las acciones que se deben ejecutar se selecciona correo electrónico. Esto permitirá notificar al usuario cuando el consumo de agua sobrepase una cierta cantidad de metros cúbicos establecidos.

Finalmente se debe agregar el widget “switch” y asignarle la variable “On/Off”. El resultado de las acciones mencionadas anteriormente se observa en la Figura 69.

**Figura 69**

*Ventana con las opciones de los widgets y variables asignadas.*

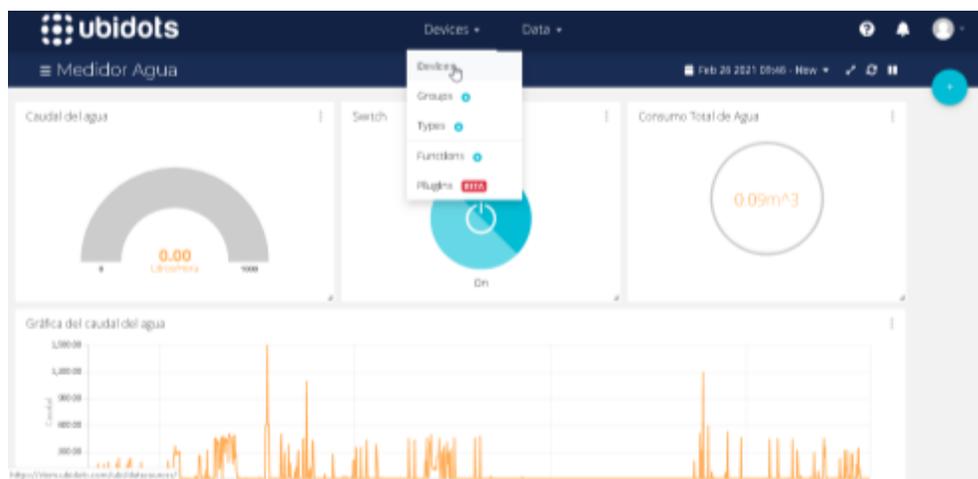


### *Pruebas de funcionamiento*

Los resultados fueron obtenidos de dos semanas de monitorización del prototipo, desde el 16 de febrero al 28 de febrero. Para analizar los datos se procede a descargar los mismos en el periodo anteriormente mencionado. Para ello se ingresa a Ubidots en la pestaña Device y luego en la opción Device, como se observa en la Figura 70.

### **Figura 70**

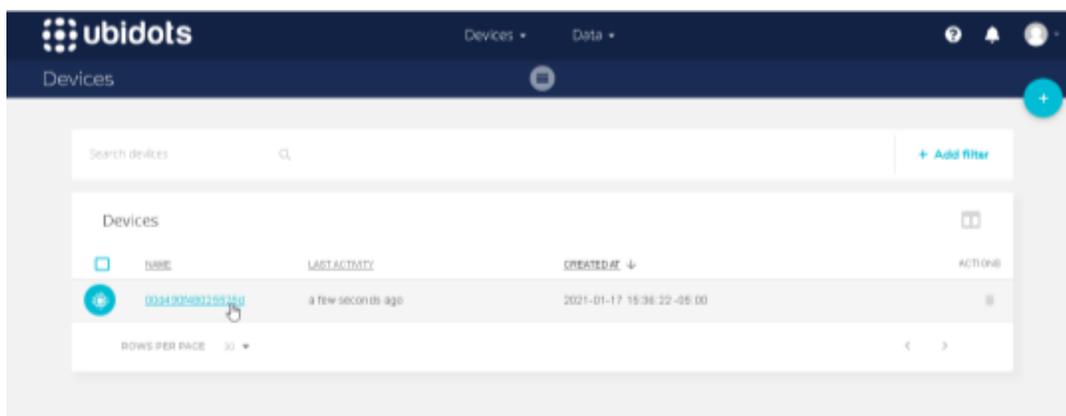
*Pestaña Device de Ubidots*



A continuación, se selecciona el dispositivo correspondiente al medidor de agua, como se observa en la Figura 71.

**Figura 71**

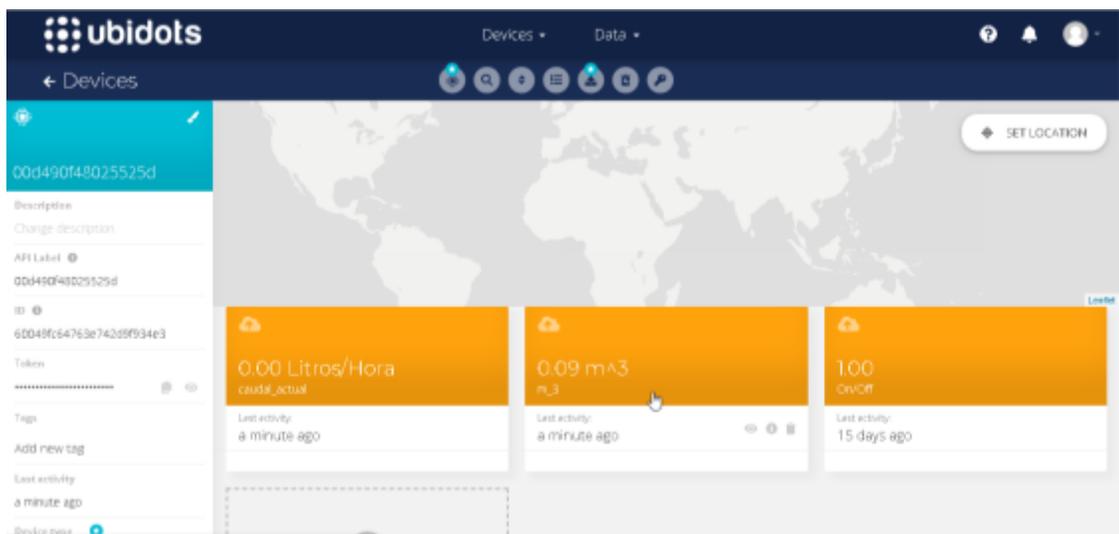
*Listado de dispositivos registrados en Ubidots*



Luego se selecciona la variable de la cual se quiere descargar la información para su análisis. En este caso se selecciona la variable m3 que corresponde al consumo de agua.

**Figura 72**

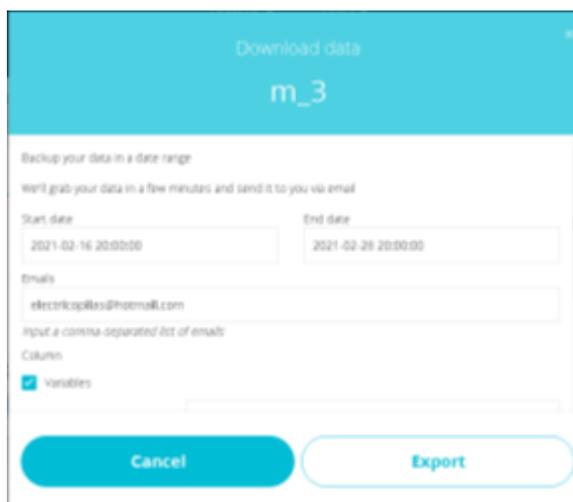
*Variables empleadas en Ubidots*



Finalmente, dar clic en Export Variable Data y a continuación se abre una ventana donde se ingresa el rango de tiempo y un correo electrónico, al cual se enviará el link de descarga para el archivo.

**Figura 73**

*Ventana de información para envío de archivos y correo electrónico*



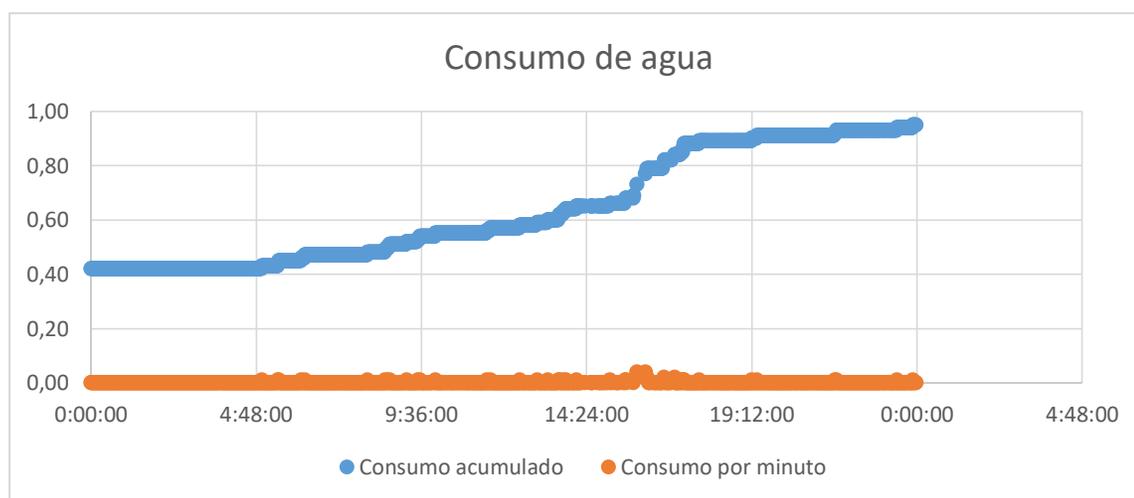
The image shows a dialog box titled "Download data" for variable "m\_3". The dialog contains the following fields and options:

- Start date:** 2021-02-16 20:00:00
- End date:** 2021-02-28 20:00:00
- Emails:** electricopilas@hotmail.com
- Columns:**  Variables

At the bottom, there are two buttons: "Cancel" and "Export".

El archivo descargado es un documento de Excel donde se detalla información de la fecha, hora y consumo de agua.

## Capítulo V

*Análisis de Resultados**Análisis por horas del consumo de agua del día 17 de febrero del 2021***Figura 74***Gráfica del consumo acumulado y consumo por minuto*

Como se puede observar en la Figura 74, en horas de la madrugada, es decir, desde media noche hasta las 5 AM el consumo acumulado de agua se mantiene constante. Esto se debe a que en ese lapso de tiempo nadie se encuentra utilizando el agua por motivos que generalmente en ese horario los habitantes del domicilio se encuentran descansando y a partir de las 5 AM en adelante se observa que el consumo acumulado de agua se va incrementando, por motivos que los huéspedes empiezan a realizar sus actividades diarias como: cepillarse los dientes, ducharse, preparar los alimentos, entre otros.

La gráfica de la Figura 74, corresponde al día 17 de febrero del 2021, en la cual se obtiene que el consumo de agua durante todo el día fue de 0.53m<sup>3</sup>. Además, se visualiza que se produce un mayor consumo de agua desde las 15:52:26 hasta las 16:07:29, en la cual se ha consumido 0.04m<sup>3</sup> lo que equivale a 40 litros

#### *Análisis del consumo por día*

A continuación, se muestra Tabla 6 que contiene información sobre el consumo diario de agua durante las dos semanas de monitorización del prototipo.

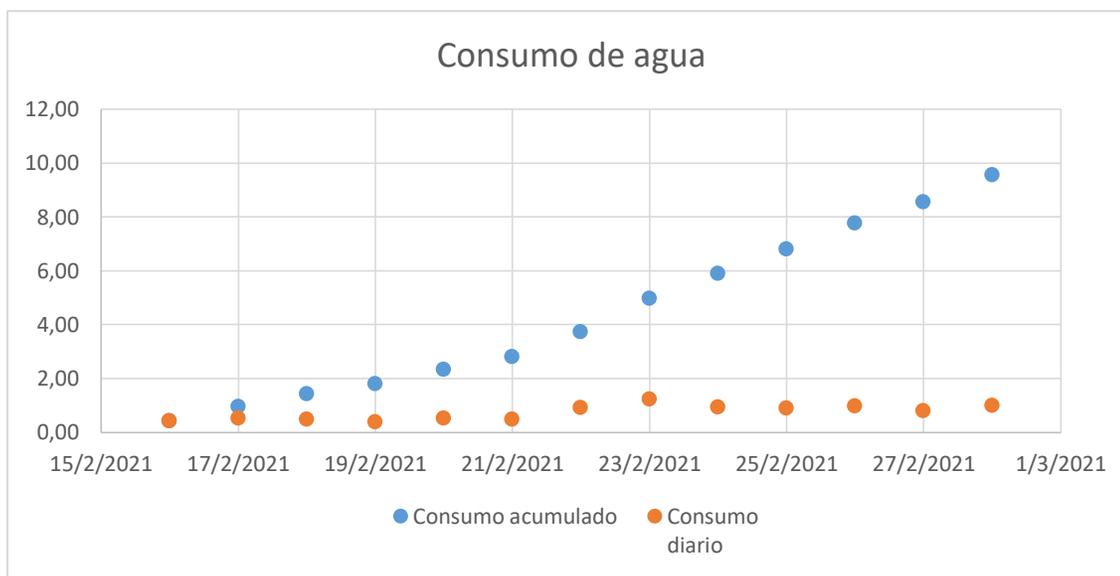
**Tabla 6**

*Consumo diario de agua en dos semanas de monitoreo*

Día	Hora	Consumo acumulado	Consumo diario
16/02/2021	23:59:29	0,42	0,42
17/02/2021	23:59:09	0,95	0,53
18/02/2021	23:59:03	1,43	0,48
19/02/2021	23:59:57	1,81	0,38
20/02/2021	23:59:27	2,33	0,52
21/02/2021	23:59:39	2,81	0,48
22/02/2021	23:59:51	3,73	0,92
23/02/2021	23:59:24	4,97	1,24
24/02/2021	23:59:20	5,90	0,93
25/02/2021	23:59:20	6,80	0,90
26/02/2021	23:59:25	7,77	0,97
27/02/2021	23:59:34	8,56	0,79
28/02/2021	20:22:41	9,55	0,99
Total			<b>9,55</b>

**Figura 75**

*Consumo acumulado y consumo diarios de agua potable en el domicilio.*



En la Figura 75 se puede observar el consumo acumulado y diario de agua durante las dos semanas de monitorización. En la cual se identifica que el día de mayor consumo de agua fue 23 de febrero del 2021 con un consumo de 1.24m<sup>3</sup>. También se puede visualizar que el consumo de agua acumulado tiene una tendencia aproximadamente lineal.

#### *Cobertura del prototipo*

Para la realización de las pruebas de cobertura del prototipo, se procede a instalar el Gateway Lora en la residencia del autor del presente trabajo de titulación. La residencia está ubicada en la ciudad de Quito en la dirección Cristobal de Acuña Oe3-320 y Ruiz de Castilla esquina. Además, el Gateway Lora se instaló a una altura aproximada de 8 m con la finalidad de obtener línea de vista entre el Gateway y el prototipo.

Luego se utiliza la funcionalidad de GPS que posee el módulo Lora/GPS Shield para enviar la ubicación del prototipo a la página de Ubidots. A continuación, se procede a la

movilización del prototipo por el barrio Santa Isabel y los barrios aledaños, para determinar la cobertura que posee la tecnología Lora.

En la Figura 76 se muestra el área de cobertura obtenido luego haber recorrido las calles y avenidas del barrio Santa Isabel y de los barrios aledaños. Obteniendo como resultado un área de cobertura  $0.77\text{km}^2$  con un perímetro de  $3.95\text{ Km}$  y un retardo promedio en la recepción de cada paquete de 19 s.

**Figura 76**

*Área de cobertura del prototipo utilizando la tecnología lora*



## Capítulo VI

### *Conclusiones*

Se determinó de manera correcta los elementos electrónicos para la monitorización y control interno de un domicilio, así como los componentes y tecnología para la transmisión y recepción de datos basados en un estudio del arte de la tecnología LoRa y sus aplicaciones en el Internet de las Cosas – IoT y ciudades inteligentes o Smart Cities.

Se implementó un respaldo de energía basado en un banco de baterías de 4000mAh, el cual dota de autonomía al sistema por 10 horas en condiciones ideales. Las dimensiones de la batería no implicaron un problema al momento de implementar y montar el sistema en la caja de proyectos.

La aplicación web, así como su configuración se desarrolló correctamente y de una forma amigable para el usuario. Su uso gratuito y renderización permite utilizarla sobre cualquier dispositivo inteligente lo que significó un ahorro de recursos económicos y de tiempo al no tener que desarrollar una aplicación móvil.

El desarrollo de los algoritmos para el monitoreo, cálculo de consumo y control de agua potable, transmisión y recepción de datos se lo realizó en bloques y con funciones para su correcta implementación y comprensión permitiendo un funcionamiento óptimo del dispositivo.

Se realizó la calibración del dispositivo a través de un laboratorio especializado en medidores de agua. La calibración se la realizó con el objetivo de determinar el número de pulsos que genera el sensor de caudal por cada litro de agua para tener un cálculo más real de flujo de agua.

Finalmente se validó el funcionamiento del dispositivo en un domicilio, logrando generar estadísticas diarias y semanales del consumo interno de un hogar determinando el valor promedio en metros cúbicos de agua que consume una familia.

### *Trabajos Futuros*

Desarrollo de una aplicación web para el registro automático de la lectura de medidores y reporte de novedades como manipulación, ruptura o medidores violentados.

Diseño y desarrollo de una pasarela IoT para evitar el salto de datos entre TTN y Ubidots. Para tener un mejor manejo y administración de la información.

Desarrollo de un Gateway LoRa que permita reducir costos en la implementación de redes LoRa para el despliegue de aplicaciones.

## Referencias

- Haro Vilaña, I. (2019). *IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO Y CONTROL AUTOMÁTICO DE RIEGO PARA INVERNADEROS MEDIANTE TECNOLOGÍA LORA CON ESP32*. TRABAJO DE TITULACIÓN, UNIVERSIDAD TECNOLÓGICA ISRAEL, CIENCIAS DE LA INGENIERIA, Quito.
- Mogrovejo Narvaez, D. (2019). *MONITOREO DE VARIABLES INDUSTRIALES BASADAS EN PROTOCOLO DE COMUNICACIÓN LoRa*. TRABAJO DE TITULACIÓN DE PREGRADO, UNIVERSIDAD TECNOLÓGICA ISRAEL, CIENCIAS DE LA INGENIERIA, Quito.
- Rose, K., Eldridge, S., & Chapin, L. (15 de Octubre de 2015). *La internet de las Cosas — Una breve reseña*. Obtenido de Internet Society: <https://www.internetsociety.org/es/resources/doc/2015/iot-overview>
- COM, G. (2015). *Una solución integrada en tu edificio*. Obtenido de <http://www.gmcom.com/cabelado-estructurado.html>
- Ferdaous, C. (2019). *LoRaWAN for IOT/AI: Applications in support of Societal Challenges*. Tunis.
- Reyes, C. R., Vaca, H. P., Calderón, M. P., Montoya, L., & Aguilar, W. G. (2017). *MilNova: an approach to the IoT solution based on model-driven engineering for the military health monitoring*. *CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, 1-5.

**Adams, W. (2009). IEEE IP over Broadband Access in Support of Convergence. *IEEE*.**

**McDermott-Wells, P. (2005). What is Bluetooth? *IEEE Potentials*, 23(5), 33-35.**

**Alliance., W.-F. (2020). *Low power, long range Wi-Fi® for IoT* . Obtenido de <https://www.wi-fi.org/discover-wi-fi/wi-fi-halow>**

**Alliance, L. (2015). *LoRaWAN ¿What is it? Technical Marketing Workgroup. Technical Marketing Workgroup 1.0.***

**Vangelista, L., Zanella, A., & Zorzi, M. (2015). Long-Range IoT Technologies: The Dawn of LoRa. . *EAI International Conference on Future access enablers of ubiquitous and intelligent infrastructures*, 51-58.**

**DIGIKEY. (2021). *DIGI-KEY ELECTRONICS*. Obtenido de Calculador de la vida útil de una batería: <https://www.digikey.com/es/resources/conversion-calculators/conversion-calculator-battery-life>**