

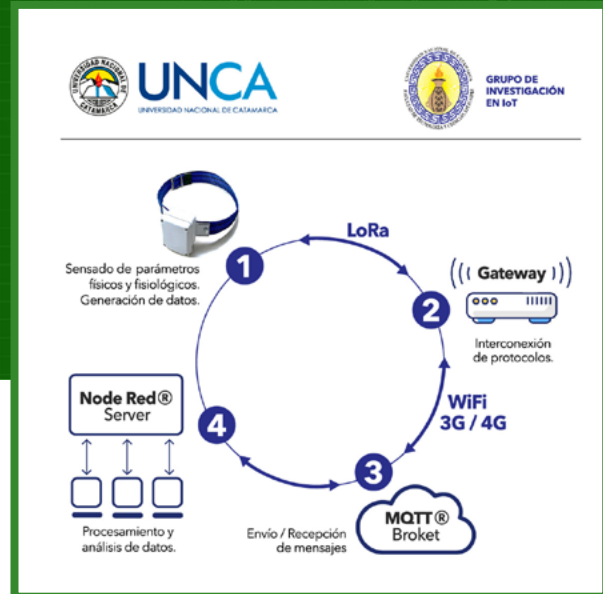
IoT aplicado a la ganadería extensiva

Marcos D. Aranda¹, Paola I. Beltramini¹, Jesús E. Cano¹, Luis D. Virragrán¹, Juan P. Moreno¹, Sergio H. Gallina¹, Oscar Ariel Herrera Conegliano²

1- Grupo de Investigación IoT, Facultad de Tecnología y Ciencias Aplicadas, Universidad Nacional de Catamarca, Argentina.

2- Instituto Nacional de Tecnología Agropecuaria INTA – EEA Catamarca, Argentina.

Contacto: markosdarioaranda@gmail.com
pbeltramini@tecno.unca.edu.ar
jesuseduardocano@hotmail.com
dvillagran@tecno.unca.edu.ar



juanpablomoreno@gmail.com
sgallina@tecno.unca.edu.ar
herrera.ariel@inta.gob.ar

RESUMEN

La ganadería en la región NOA es en términos generales extensiva, es decir que se realiza en amplias superficies donde los animales recorren grandes distancias en busca de pasturas y agua. En el presente trabajo se describe la tecnología IoT puesta a disposición de la ganadería a los efectos de mejorar la producción y rentabilidad, con sostenibilidad del territorio. El desarrollo llevado a cabo entre el INTA Catamarca y el Grupo de Investigación en Internet de las Cosas de la Facultad de Tecnología de la UNCA, hace uso de redes de bajo consumo y largo alcance para el monitoreo de animales. Se presenta el desarrollo de un nodo diseñado para obtener información acerca de la posición y estado biológico del ganado, la manera en la que resolvió el almacenamiento y la transmisión de dicha información mediante la combinación de LoRa y MQTT. Se logró un prototipo de excelentes prestaciones.

ABSTRACT

Advances in hardware development and technologies applied to the Internet of Things (IoT) have allowed the manufacture and deployment of networks of small wireless nodes, custom designed for different applications and that can be adapted to the needs of problems individuals. These wireless nodes are composed of a central processing unit or microcontroller, various sensors and communication modules that allow data to be collected from a physical environment, stored, and transmitted to a central server for processing. This article presents the development of a prototype node manufactured to obtain information about the position and biological status of cattle in the province of Catamarca, and the way in which it resolved the storage and transmission of the information collected by the node by combining LoRa and MQTT. A prototype of excellent performance was achieved.

INTRODUCCIÓN

El concepto de “Internet de las cosas” IoT (Internet of Things), entendido como la interconexión digital de objetos cotidianos con cualquier otro de su alrededor, se ha convertido en los últimos años en uno de los términos más populares en la industria tecnológica gracias al enorme número de dispositivos que cuentan con la posibilidad de interconectarse y a la necesidad de crear redes capaces de gestionar estos dispositivos [1].

Para atender las necesidades de IoT se han desarrollado diversas tecnologías Low-Power Wide Area Networks (LPWANs), que se caracterizan por ofrecer conectividad inalámbrica de largo alcance a un gran número de dispositivos (nodos), con menor consumo de energía y menor costo que las redes móviles, siendo su principal limitante el ancho de banda reducido. Entre las tecnologías LPWANs más utilizadas en IoT están: LoRaWAN, SigFox, In-genuRPMA, Weightless, DASH-7, nWave. [1]

Los nodos son pequeñas unidades conformadas por un microcontrolador, uno o varios sensores, un módulo de conversión A/D y acondicionamiento de señal, otro de energía, un módulo de comunicación y, en algunos casos, actuadores. Estos nodos tienen como función primaria la recolección y transmisión de datos para un control centralizado. Gracias a los avances en las técnicas de fabricación de componentes, desarrollando pequeños Sistemas Micro Electro Mecánicos (MEMS), es posible fabricar nodos de reducido tamaño (del orden del micrómetro al milímetro), bajo costo, comunicaciones seguras, movilidad en las comunicaciones y poco consumo.

La información recolectada por estos sensores MEMS, puede almacenarse en memorias, utilizarla para ejecutar acciones en tiempo real o enviarse a un servidor de red a través de un Gateway, para su posterior análisis y procesamiento. El Gateway representa un puente transparente que retransmite los mensajes recibidos. La comunicación entre nodos y Gateway se puede realizar a través de diferentes tecnologías de comunicación inalámbrica, que se extiende en diferentes canales de frecuencia y velocidad de datos, que dependen del intervalo de comunicación y la duración del mensaje.

Como surge de lo que se viene expresando, los sistemas de IoT tienen tres componentes fundamentales, el nodo y sus sensores, el mecanismo de comunicación nodo-gateway y el Gateway propiamente dicho que nos permite subir la información a la red.

En este artículo, se comenta la experiencia en el desarrollo de un nodo prototipo y su sistema de comunicación con el Gateway basado en LoRa, aplicado a un proyecto de ganadería de precisión en áreas extensivas que lleva adelante el INTA Catamarca.

JUSTIFICACION

El desarrollo del nodo surge de un Convenio de Vinculación Tecnológica celebrado entre el Grupo de Investigación en Internet de las Cosas (GIoT) de la Facultad de Tecnología y Ciencias Aplicadas (FTyCA) de la Universidad Nacional de Catamarca (UNCA) y el grupo de Ganadería de precisión del Instituto Nacional de Tecnología Agropecuaria (INTA-EEA Catamarca). Su objetivo es promover el desarrollo agropecuario, atendiendo de esta manera a los crecientes requerimientos del sector, principalmente en zonas áridas donde el animal debe recorrer largas distancias diarias para buscar alimento.

La ganadería en zonas áridas se realiza de manera extensiva, por lo que es una demanda regional actual, la necesidad de contar con información de la localización animal y sus patrones de desplazamiento tanto con fines de investigación como con fines productivos. Sin embargo, uno de los principales impedimentos para conseguir esta información es que los dispositivos que se consiguen en mercado son importados, tienen un costo elevado y la información que brindan es limitada.

Dado que este área se encuentra en constante evolución, en el país algunas empresas ofrecen el desarrollo de dispositivos similares. Sin embargo, en la provincia de Catamarca, no existen proveedores ni servicios técnicos capaces de dar soporte técnico inmediato a las necesidades del sector agropecuario local, lo que conlleva grandes demoras y costos extras.

Ante esta realidad actual y en el marco del citado convenio, se propuso diseñar y construir un prototipo de un dispositivo electrónico no invasivo de monitoreo y almacenamiento de datos (nodo), que permita obtener información acerca de la ubicación y el estado biológico del ganado.

En función de dichos objetivos y de común acuerdo entre los grupos participantes se fijaron algunos requerimientos de diseño, tales como:

- El primer prototipo del dispositivo (Nodo) permitirá realizar el monitoreo de algunos parámetros ambientales y fisiológicos del ganado (vacuno, bovino o caprino) necesarios para su estudio y obtener su localización geográfica.
- El dispositivo se montará en un collar en el cuello del animal, por lo que debe ser robusto.
- El costo final del nodo debe ser accesible para poder colocarlo a un número representativo de animales del rodeo.
- El equipo móvil debe tener una autonomía mínima de 30 días.
- La información recogida por el nodo se almacenará en memoria para asegurar su resguardo y posterior tratamiento.

- Para mejorar las prestaciones del nodo, y el acceso en tiempo real a la información recolectada por el mismo, se ofrece transmitir la información a la red global mediante comunicación inalámbrica para ser consultada mediante una aplicación del usuario. Esto permitirá monitorear a animales las 24 hs del día, sin importar su ubicación y actuar ante alguna contingencia.

ESPECIFICACIONES TÉCNICAS

En función de los requerimientos acordados y como parte del trabajo que viene desarrollando el grupo de investigación, se establecieron las especificaciones técnicas iniciales del proyecto:

- Cada nodo contendrá una unidad de procesamiento central con capacidad para recoger información de una serie de sensores y almacenarla en memoria (micro SD / EEPROM) para asegurar su resguardo.

- El nodo deberá recabar datos de los sensores cada 10 min, como mínimo.

- El nodo deberá ser autónomo, su alimentación se hará mediante baterías de bajo peso y deberán asegurar una duración de 30 días sin necesidad de recarga.

- Desde el punto de vista constructivo, su tamaño debe reducirse al menor posible y debe ser estanco categoría IP65 y resistente a golpes y vibraciones.

- Respecto a la transmisión de la información, la tecnología de comunicación debe ser de bajo consumo, la cobertura del enlace debe ser de largo alcance y se deberá disponer de acceso a la red global mediante WiFi o telefonía celular.

Para dar cumplimiento a estas especificaciones, se diseñó un primer prototipo de nodos sensores que trabajan de manera independiente y almacenan la información recogida en una memoria, a la que se podrá acceder localmente. La figura 1 ilustra el diagrama en bloque de un nodo

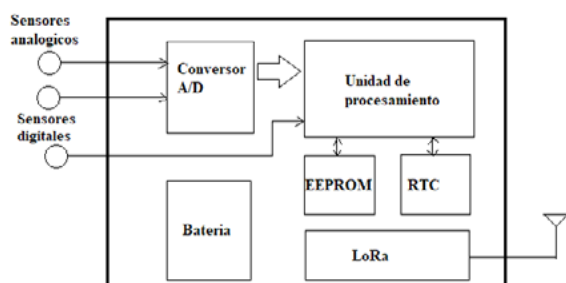


Figura 1: Diagrama en bloques del nodo.

De manera paralela, el equipo se abocó al estudio del sistema de comunicación LoRa el cual

surge como el más indicado para la transmisión de los datos en regiones donde no existen redes WiFi y la telefonía celular solo está disponible en puntos específico.

DESARROLLO DEL NODO

Hardware

Siguiendo el diagrama en bloques mostrado en la figura 1, a manera de resumen se comentan los componentes del nodo y sus funciones. Los sensores seleccionados en esta etapa fueron elegidos por su disponibilidad en el mercado nacional y su costo, pudiendo ser reemplazados en desarrollos futuros en función del análisis de los resultados de su funcionalidad por parte del INTA.

- Unidad de procesamiento: para realizar un prototipado más rápido se utilizó una placa Adafruit Feather M0 [2] que posee un procesador ARM-Cortex M0 e incorpora un módulo de radio RFM9x LoRa 868/915 MHz. Es liviana y de tamaño reducido (51 mm x 23 mm x 8 mm).

- Memoria EEPROM de 1024 Kbits: basada en la estructura del dato que se adoptó, esta memoria permite almacenar la información capturada por los sensores y el estado de la batería, durante un periodo superior a los 30 días con un tiempo entre muestras de 20 minutos.

- Sensor de posicionamiento global o Módulo GPS permite tomar datos de fecha y hora y la posición satelital del animal.

- Sensores de temperatura y humedad relativa: para medir la temperatura y humedad del ambiente en que se encuentre pastando el ganado y la temperatura corporal del animal.

- Banco de Baterías de litio-polímero de 3,7V para alimentar todos los componentes del nodo.

- Acelerómetro (opcional): detecta el movimiento de la cabeza del animal.

- Sensor flex (opcional): mide el Ritmo cardíaco del animal a partir del cambio de la resistencia a lo largo del sensor debido a la deformación o flexión del mismo.

Estos dos últimos sensores se implementarán en etapas futuras, posterior a la evaluación de este primer prototipo.

No fue necesaria la utilización de un reloj de tiempo real (RTC) ya que se accede a la información de la fecha y la hora local mediante la lectura del GPS y una simple adecuación por software al uso horario de Argentina.

En la figura 2 se muestra el diseño del nodo a nivel de prototipo, utilizado para las pruebas preliminares de adquisición, almacenamiento y transmisión de datos.

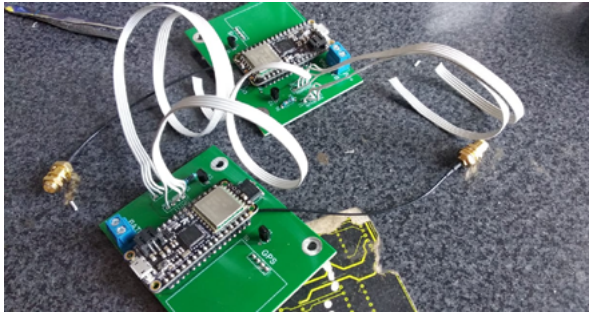


Figura 2: prototipo del nodo.

Firmware

Definidos los requerimientos funcionales y los elementos centrales del hardware, se desarrollaron los componentes del software mediante una metodología top-down, comenzando con las capas inferiores que trabajan directamente sobre el hardware. Así, se configuraron los drivers básicos de los componentes a utilizar, lo que en la figura 3 se denomina “capa 0”. Sobre esta capa, integrando la “capa 1”, se encuentran los archivos header o cabecera y finalmente como “capa 2” tenemos la aplicación que dará las funcionalidades generales del nodo. Este diseño permite: a) incorporar o reemplazar sensores y b) cambiar la funcionalidad del nodo.

Capa	Nombre del software										
2	Aplicación (APP_Nodo)										
1	Bateria.h	Led.h	RAM.h	Serial.h	DHT.h	GPS.h	LowPower.h	EEProm.h	LoRa.h		
0	Arduino.h				DHT.h	A_MLX30614.h	A_GPS.h	RTCZoo.h	LPower.h	A_EE.h	A_LoRa.h
HW	Adafruit Feather M0 + HW GPS + HW DHT + EEPROM + LoRa (RF)										

Figura 3: Estructura de capas del software

COMUNICACIÓN INALÁMBRICA: ESTÁNDARES Y PROTOCOLOS

LoRa® y LoRaWAN™

Para mejorar las prestaciones del desarrollo se decidió dotar al nodo con la capacidad de transmitir la información a la red global mediante comunicación inalámbrica.

La comunicación inalámbrica se puede realizar a través de diferentes estándares y protocolos, tales como los actuales redes GSM (Global System for Mobile), UTM (Universal Mobile Telecommunications System o Sistema Universal de Telecomunicaciones Móviles), WiFi, LoRa (Low Range), siendo ésta última la que mejor se ajusta a los requisitos y necesidades esenciales para la aplicación que hemos planteando (ver figura 4), ya que se centra en los requisitos claves de IoT tales como la comunicación bidireccional segura, la movilidad y los servicios de localización.

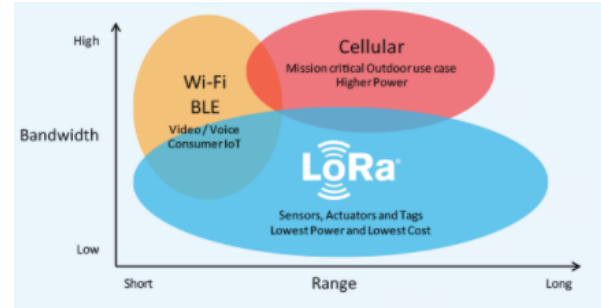


Figura 4: Ubicación de LoRa en relación a otras tecnologías inalámbricas

Fuente: Semtech

LoRa® (acrónimo de Long Range) es una técnica de modulación de radio, desarrollada entre 2008 y 2013 en Francia y adquirida y patentada por la compañía Semtech, quien comercializa solo los dispositivos. LoRa es la capa física de la red LPWAN, habilita el enlace de comunicación de largo alcance, mientras que LoRaWAN™ define el protocolo de comunicación y la arquitectura del sistema para la red (ver figura 5). Estas capas LoRaWAN son desarrolladas de forma abierta por una organización sin fines de lucro denominada Lora Alliance [3], [4]

La capa física LoRa utiliza la técnica de modulación de radio Chirp Spread Spectrum (CSS o espectro expandido) y permite un alcance amplio a baja potencia. Opera en las bandas ISM de 433, 868 o 915 MHz, dependiendo del país en el que esté funcionando el sistema. La carga útil de cada transmisión puede variar de 2255 octetos y la velocidad de datos puede alcanzar hasta 50 Kbps cuando se emplea la agregación de canales.

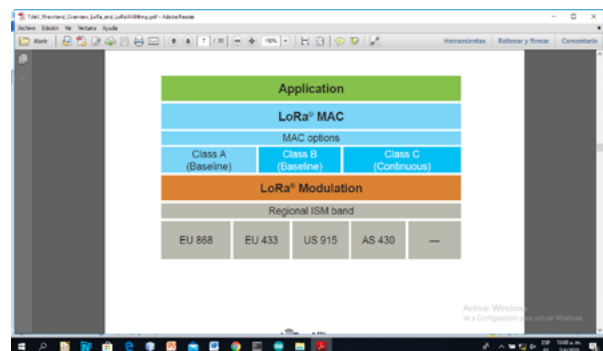


Figura 5: Capas LoRaWAN

Fuente: A technical overview of LoRa® and LoRaWAN™

LoRaWAN es un protocolo de red que usa la tecnología LoRa para comunicar y administrar dispositivos LoRa. Proporciona un mecanismo de control de acceso medio, que permite que muchos dispositivos finales puedan comunicarse con una puerta de enlace usando la modulación LoRa. Una red típica de LoRa es una topología en estrella, que

incluye tres tipos diferentes de dispositivos, los nodos o end device, gateways y el servidor. [4]

Los dispositivos LoRa y el protocolo abierto LoRaWAN® se han convertido en la tecnología elegida para aplicaciones de Internet de las Cosas (IoT), agricultura inteligente, medición inteligente, entre otras, ya que permiten comunicaciones inalámbricas a largo alcance, bajo consumo de energía (~100mW) y transmisión segura de datos.

Ventajas de la tecnología LoRa:

- Largo alcance (valores teóricos de 2 km en área urbana a 10 o 20 km en área rural), mayor que las redes celulares.
- Alta inmunidad a las interferencias
- Alta sensibilidad para recibir datos (-168dB)
- Basado en modulación “chirp”
- Bajo Consumo (hasta 10 años con una batería*)
- Baja transferencia de datos (hasta 255 bytes)
- Conexión punto a punto
- Frecuencias de trabajo: 868 MHz en Europa, 915 MHz en América, y 433 MHz en Asia
- La tecnología LoRa se puede conectar fácilmente a la infraestructura existente.
- Permite aplicaciones de IoT de bajo costo que funcionan con baterías.

La arquitectura básica de una red LoRaWAN es la siguiente: los nodos se comunican con el Gateway o puerta de enlace usando LoRa con LoRaWAN. El Gateway reenviará el paquete recibido (tramas LoRaWAN) a un servidor de red IP a través de WiFi, Ethernet, 3G o 4G celular (figura 6) [3].

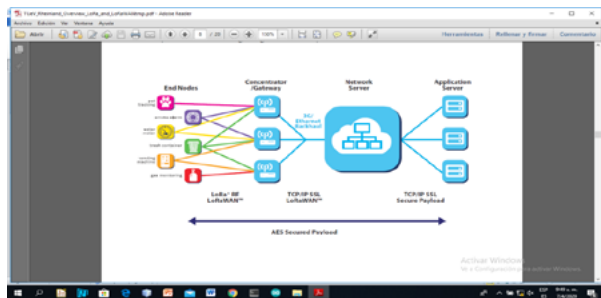


Figura 6: Topología de Red LoRaWAN

Fuente: A technical overview of LoRa® and LoRaWAN™

Un Gateway LoRa es simplemente un dispositivo que tiene un módulo de transmisión/Recepción LoRa y que redirige la información entre Internet y los nodos que se comuniquen con él y viceversa.

Para hacer viable una red en estrella de largo alcance, el Gateway debe tener una capacidad muy alta para recibir mensajes desde un gran número de nodos. La alta capacidad de red en LoRaWAN se logra utilizando la velocidad de datos adaptativa y utilizando un transceptor multicanal en el gateway para que se puedan recibir mensajes simultáneos en múltiples canales.

En nuestro desarrollo, por razones de costos, utilizamos un Gateway “Dragino LG02”, que posee las siguientes características [7]:

- Es una puerta de enlace LoRa de doble canal de código abierto.
- Puede conectarse a Internet por interfaz WiFi, puerto Ethernet y Puerto host USB.
- Puede admitir el protocolo LoRaWAN en frecuencia única y protocolo LoRa personalizado.
- Rango de Frecuencias LoRa: Band 1 (HF): 862 ~ 1020 Mhz; Band 2 (LF): 410 ~ 528 Mhz
- Alcance máximo en LoRa: 5 ~ 10 km. Área de densidad:> 500m
- Admite hasta 300 nodos
- Permite múltiples modos de trabajo como: modo LoRa repetidor, modo MQTT, modo Cliente TCP/IP, modo Servidor TCP/IP, lo que le permite adaptarse a diferentes requisitos de conexión IoT.

MQTT®, Eclipse Mosquitto™ y Node-RED

MQTT (Message Queuing Telemetry Transport) es un protocolo de comunicación M2M (machine-to-machine) de tipo messagequeue. Está basado en el protocolo TCP/IP como base para la comunicación. Fue creado en 1999 por el Dr. Andy Stanford-Clark de IBM y Arlen Nipper de Arcom (ahora Eurotech) como un mecanismo para conectar dispositivos empleados en la industria petrolera a servidores remotos. En el año 2014 pasó a ser un estándar abierto supervisado por la OASIS (Organization for the Advancement of Structured Information Standards) [5].

MQTT es un protocolo pensado para IoT, ya que está enfocado al envío de datos en aplicaciones donde se requiere muy poco ancho de banda y pocos recursos para su funcionamiento, como por ejemplo el envío de datos de sensores. Es bidireccional, ya que además de enviar datos de sensores, permite comunicarse con los dispositivos, es decir, enviar señales a sensores y actuadores, dispositivos móviles o micro controladores.

El funcionamiento del MQTT es un servicio de mensajería push con patrón publicador/suscriptor (pub-sub). En este tipo de infraestructuras los clientes se conectan con un servidor central denominado broker. Para filtrar los mensajes que son enviados a cada cliente, los mensajes se disponen en topics organizados jerárquicamente. Un cliente puede publicar un mensaje en un determinado topic (en nuestro caso, temperatura, posición, etc.). Otros clientes pueden suscribirse a este topic y el broker le hará llegar los mensajes suscritos (ver figura 7).

Los clientes inician una conexión TCP/IP con el broker, el cual mantiene un registro de los clientes conectados. Esta conexión se mantiene abierta hasta que el cliente la finaliza. Por defecto, MQTT emplea el puerto 1883 y el 8883 cuando funciona sobre TLS.

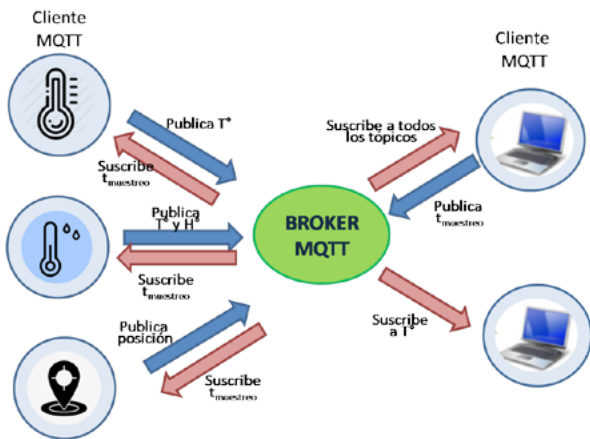


Figura 7: Arquitectura del modelo publicador/suscriptor

Un broker de mensajería es un programa que intercambia mensajes entre aplicaciones a través de algún medio de comunicación.

Eclipse Mosquitto es un broker de mensajes de código abierto (con licencia EPL / EDL) que implementa las versiones 5.0, 3.1.1 y 3.1 del protocolo MQTT. Mosquitto es liviano y es adecuado para usar en todos los dispositivos, desde computadoras de baja potencia hasta servidores completos [5].

El proyecto Mosquitto también proporciona una biblioteca C para la implementación en clientes MQTT. test.mosquitto.org. (IP: 5.196.95.208) es un host público de Mosquitto MQTT que funciona como un servidor /broker disponible públicamente.

Uno de los componentes más importantes del protocolo MQTT es la definición y tipología de los mensajes, ya que son una de las bases de la agilidad en la que radica su fortaleza. Cada mensaje consta de 3 partes (figura 8):

Always		Optional	Optional
Fixed Header		Optional Header	Payload
Control Header	Packet Length		
1 Byte	1-4 Bytes	0-Y Bytes	0-256Mbs

Figura 8: Estructura de un mensaje MQTT

- Cabecera fija. Ocupa 2 a 5 bytes, obligatorio. Consta de un código de control, que identifica el tipo de mensaje enviado y de la longitud del mensaje. La longitud se codifica en 1 a 4 bytes, de los cuales se emplean los 7 primeros bits y el último es un bit de continuidad.

- Cabecera variable. Opcional, contiene información adicional que es necesaria en ciertos mensajes o situaciones.

- Contenido (payload). Es el contenido real del mensaje. Puede tener un máximo de 256 Mb aunque en implementaciones reales el máximo es de 2 a 4 kB.

Como el broker sólo intercambia mensajes entre los dispositivos o aplicaciones, para acceder a la información se debe disponer de un software (que se instale en un nodo o en el mismo servidor que el broker) que edite y procese dicha información.

Node-RED es una herramienta de programación visual para conectar dispositivos de hardware, API y servicios en línea de formas nuevas e interesantes. Es un proyecto de la Fundación JS (<https://js.foundation/>) que se ha convertido en un estándar open-source para la gestión y procesamiento de datos en tiempo real en entornos de Industry 4.0, IOT, Marketing digital o sistemas de Inteligencia Artificial, entre otros, logrando simplificar los procesos entre productores y consumidores de información.[6]

Implementación

Se optó por una comunicación inalámbrica heterogénea. En la figura 9 se muestra esquemáticamente la topología adoptada.

Los nodos finales se conectan al Gateway mediante comunicación LoRa, y el gateway se conecta al exterior del sistema utilizando tecnología de comunicación inalámbrica WiFi/3G/4G, con protocolo MQTT. La información recogida por los nodos se envía vía Gateway al broker MQTT mosquitto, para que luego, a través de la consola de debug de Node Red se pueda tener acceso y visualización de la información.

Como se observa en la figura 9, se pueden establecer dos caminos de conexión, en el primero (UPLINK) o enlace ascendente, el sensor envía datos al Gateway a través de LoRa wireless, quien procesará estos datos y los reenvía al BrokerMQTT remoto a través de Internet, al cual se accede mediante la aplicación creada en Node Red. Para enlace descendente (DOWNLINK), la puerta de enlace (Gateway) sabrá y transmitirá los datos a la red LoRa local.

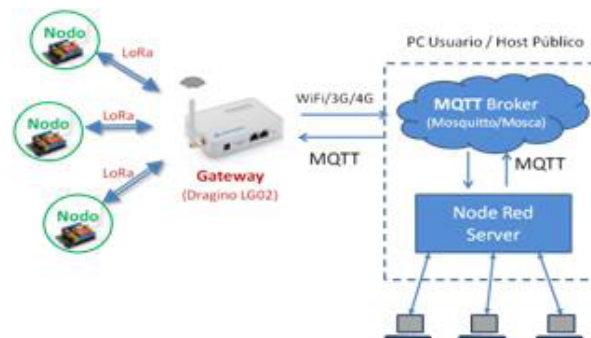


Figura 9: Arquitectura con LoRa – MQTT

En cuanto al contenido útil del mensaje, en nuestro caso, las pruebas se realizaron con el payload mostrado en la figura 10, donde la información se envía en el siguiente orden: ID_Productor/ ID_Animal / Fecha/ Hora/ Temperatura Ambiente/ Temperatura Corporal/ Humedad / Latitud / Longitud / Tensión de la Batería. Se utilizan la barra inclinada como carácter de separación.

ID P	ID A	Fecha	Hora	T A	T C	H A	Lat.	Long.	Bat.
5B	4B	8B	5B	4B	4B	4B	12B	12B	3B

Figura 10: Payload del nodo para control de ganado

Los primeros 9 bytes del paquete mostrado en la figura 10 se mantienen fijos para un nodo (animal) dado, se cargan manualmente y se guardan en la memoria flash de la placa Adafruit. Las lecturas de los sensores (a partir del byte 10) se guardan en la memoria EEPROM. Posteriormente se construye el payload que se transmiten via LoRa al Gateway, para continuar el camino precedentemente indicado.

En Node-Red se realiza la decodificación del mensaje, pudiendo acceder a los datos recibidos en la consola de Debug. Node-Red permite además crear una base de datos para el resguardo de la información, pudiendo de esta manera almacenar y procesar la información del nodo en tiempo real.



Figura 11: Interfaz del usuario.

Asimismo, se diseñó una interfaz gráfica de usuario (figura 11) que permite configurar la frecuencia de lectura y visualizar y guardar los datos almacenados en la memoria, generando un archivo .txt.

PRUEBAS EXPERIMENTALES

Para realizar las pruebas de funcionamiento del nodo, transmisión-recepción mediante LoRa y alcance de las comunicaciones, se instaló el gateway en una oficina de la Facultad de Tecnología y Ciencias Aplicadas de la UNCA, en un primer piso de las instalaciones, localizada en zona urbana, rodeada de edificaciones y arboleda.

Se programó y configuró el nodo para que realice la toma de datos de los sensores, los guarde en memoria y los transmita al Gateway. Si bien se estableció un período mínimo de toma datos de los sensores de 10 minutos, durante las pruebas experimentales y a efectos de determinar la funcionalidad del nodo, se fue variando la frecuencia de lectura a partir de un tiempo mínimo entre muestras de 1 minuto.

Las pruebas se realizaron en diferentes etapas. Por un lado, para verificar la comunicación nodo-gateway, para validar los datos recibidos en NodeRed con los almacenados en memoria y para realizar un análisis de la autonomía del nodo. Para ellos, en la primera etapa, se hicieron ensayos de comunicación manteniendo fija la distancia nodo-gateway y se efectuaron pruebas de conectividad y validación de las mediciones, no observándose pérdidas de información ni datos erróneos, siempre cotejando la información recibida en Node Red con la guardada en memoria (figura 12).

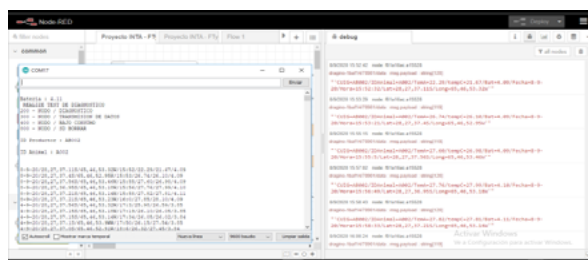


Figura 12: información recibida en Node Red vs Memoria Eeprom

Posteriormente, se fue variando la posición del nodo prototipo, siempre en el predio de la Facultad, a efectos de determinar el alcance del nodo. Cabe destacar que a pesar de no ser éstas las condiciones de funcionamiento normales del nodo, se logró una efectividad mayor al 80% a distancias cercanas a los 500 m en zonas urbanas.

En una segunda etapa de ensayo, se fueron modificando los períodos de lectura para verificar los cálculos realizados al determinar la autonomía del nodo (en capacidad de memoria y duración de la batería).

CONCLUSIONES

En esta etapa de desarrollo del prototipo, se destaca en primer término, el impacto positivo de la transferencia bidireccional de conocimiento entre la universidad y el sector productivo, representado en esta oportunidad por el INTA.

En cuanto a los logros como grupo de investigación, el presente trabajo permitió adquirir los conocimientos y el dominio de las comunicaciones LoRa y MQTT.

Respecto al prototipo desarrollado, se obtuvo un dispositivo que responde con éxito a los requerimientos iniciales. Las principales limitaciones encontradas se circunscribieron a la falta de disponibilidad en el mercado nacional y/o elevado costo de algunos elementos, como por ejemplo un Gateway LoRa outdoor de alta calidad. Por ello, se optó por trabajar con un equipo Dragino LG02 de excelentes prestaciones para las pruebas realizadas.

Se consiguieron mediciones de los parámetros físico, el almacenamiento y la transmisión de la información, con resultados satisfactorios respecto a la duración de la batería. La vida útil de la batería estará influenciada por la frecuencia de lectura y transmisión de la información proveniente de los sensores, entre otros.

A pesar de las limitaciones encontradas se han realizado ensayos en ciudad, logrando comunicaciones a distancias mayores a 500 m, lo que da cuenta de la robustez de la modulación LoRa, apropiada para operar en bandas no licenciadas.

Finalizada la etapa de prueba sobre un primer nodo prototipo, en etapas posteriores se espera realizar mediciones en el campo experimental INTA, a efectos de determinar la autonomía y fiabilidad del sistema en un ambiente real de operación.

Como trabajos a futuros se prevé el desarrollo de una base de datos en Node-Red que permita almacenar la información obtenida para su tratamiento posterior y una interfase de usuario que permitan visualizar en tiempo real los datos obtenidos, sin dejar de lado el respaldo de la información en memoria no volátil.

REFERENCIAS

- [1] Villagrán L. D., Gallina S. H., Aranda M. D., y otros. (2018). Nodos de control con tecnología LoRa y LoRaWan en desarrollos del manejo de rodeos. IX *Congreso de Microelectrónica aplicada (UEA2018)*, p.18.
- [2] Adafruit Industries, (2020). Adafruit Feather M0 Radio with LoRa Radio Module. <https://www.adafruit.com/product/3178>
- [3] SemtechLoRa® (2020). Recuperado de: <https://www.semtech.com/lora>
- [4] A technical overview of LoRa® and LoRaWAN™. (2020) Recuperado de: <https://lora-alliance.org>
- [5] Eclipse Mosquitto™ An open source MQTT broker (2020). Recuperado de: <https://mosquitto.org/>
- [6] Node-RED dashboard User Manual Getting started (2020). Recuperado de: <https://nodered.org/>
- [7] https://wiki.dragino.com/index.php?title=Main_Page