

ESCUELA SUPERIOR POLITECNICA DEL LITORAL
Facultad de Ingeniería en Electricidad y Computación



“DISEÑO Y SIMULACIÓN DE UNA SOLUCIÓN IOT
PARA MONITOREO REMOTO DE PARÁMETROS DE
CALIDAD DE AGUA EN PISCINAS CAMARONERAS.”

EXAMEN DE GRADO

Previo a la obtención del título de:

MAGISTER EN TELECOMUNICACIONES

Presentado por:

DIEGO FERNANDO YUNGÁN GONZÁLEZ

GUAYAQUIL – ECUADOR

AÑO: 2020

AGRADECIMIENTOS

A Dios primeramente por ser mi impulso y fortaleza para alcanzar y conquistar logros en todos los ámbitos de mi vida, a la Ing. Jacqueline Yungán por su asesoría en temas relacionados con los parámetros de calidad de agua requeridos para la óptima operación de piscinas camaroneras, a cada uno de los docentes de la Maestría en Telecomunicaciones de ESPOL por su excelencia en la impartición de conocimientos, a la Dra. María Antonieta Álvarez por su efectiva gestión en la coordinación de este proceso de graduación, y al Máster Jorge Brito evaluador el presente trabajo.

DEDICATORIA

Dedico este trabajo a Dios, mi Padre celestial, a mis padres terrenales Marco e Inés, a mi hermano David, al resto de mi familia entre ellos a quienes ya no están presentes conmigo físicamente en la tierra, pero están presentes en el cielo, en mi memoria y en mi corazón, y a todos mis leales amigos, llevan un lugar especial en mi vida y sé que este triunfo también es especial para ustedes.

TRIBUNAL DE EVALUACIÓN

Jorge Brito C.

M.Sc. Jorge Brito

PROFESOR EVALUADOR

PhD. María Antonieta Alvarez

PROFESOR EVALUADOR

DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, me corresponde exclusivamente; y doy mi consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



Diego Fernando Yungán González

RESUMEN

El proyecto descrito en el presente documento se basa en el diseño y simulación de una solución IoT útil para el sector camaronero en el ámbito del poder monitorear de manera remota los parámetros de calidad de agua de un grupo piscinas camaroneras.

Esta solución IoT se ha diseñado para poder llegar a operar en un grupo de 8 piscinas camaroneras cuya extensión total es de 121,66 hectáreas. Cada piscina camaronera tendrá ubicado en la cercanía de sus compuertas de salida de agua sensores capaces de captar los niveles de los parámetros que describen la buena o mala calidad de agua en estas piscinas, estos parámetros son los siguientes:

- A. Temperatura del agua.
- B. Concentración de Oxígeno(O₂).
- C. Potencial de Hidrogeno (pH).
- D. Concentración de Amonio (NH₄).

Debido a la gran extensión de este grupo de piscinas camaroneras, la tecnología a utilizar para este diseño serpa la tecnología LoRa Wan, la cual es una tecnología de bajo consumo de potencia y largo alcance en cuanto a distancias de comunicación.

En cada piscina los datos tomados por los sensores serán procesados por un microcontrolador para a través de un módulo LoRa Wan ser enviados a un LoRa Gateway el cual cuenta conectividad a internet para así mediante la utilización de soluciones Cloud poder observar de manera remota la información captada por los sensores.

Para simular lo antes propuesto se hará uso de la herramienta de simulación Cisco Packet Tracer en la cual se configurará una red sensorial capaz de captar los parámetros relacionados a la calidad de agua en cada piscina de cría de camarones.

En vista a que los sensores de esta red son de uso especializado y en su mayoría no se encuentran disponibles entre los dispositivos simulables de Cisco Packet; varios de ellos serán emulados mediante la modificación de programación de sensores existentes en el simulador; la operación de estos sensores puede ser programada tanto en lenguaje Python como en JavaScript.

Los sensores mencionados anteriormente estarán interconectados a un microcontrolador (MCU) los cuales tendrán comunicación inalámbrica con un Gateway para tener conectividad hacia internet.

La funcionalidad de cada microcontrolador (MCU) de esta simulación será programado en lenguaje Python para recibir de forma adecuada los valores captados por cada sensor dentro de la simulación, además los MCUs serán programados para que los valores obtenidos en esta simulación sean exteriorizados hacia la solución Cloud PaaS (Platform as a Service) denominada Thing Speak de la compañía MathWorks, en donde se hará storing de los valores obtenidos en la simulación y a su vez dichos valores podrán ser visualizados y analizados a través de channels de la plataforma de Thing Speak ya sean estos privados o públicos desde cualquier parte del mundo por quien posea acceso a dichos channels o tópicos IoT. En otras palabras, los valores obtenidos dentro la simulación podrán ser vistos fuera de la simulación mediante la plataforma de Thing Speak.

El comportamiento de los parámetros a sensor será programado en el apartado “Environments” de Cisco Packet Tracer y se crearán ambientes adicionales a los que vienen por defecto en el simulador a fin de que el escenario de simulación sea lo más cercano al escenario de una piscina camaronera.

ÍNDICE GENERAL

RESUMEN	vi
ÍNDICE GENERAL	viii
ÍNDICE DE FIGURAS	x
ÍNDICE DE TABLAS	xiii
ABREVIATURAS.....	xiv
CAPÍTULO 1	14
1. INTRODUCCIÓN.....	14
1.1. Descripción del Problema	14
1.2. Justificación de la propuesta.....	14
1.3. Objetivos.....	15
1.3.1. Objetivo General	15
1.3.2. Objetivos Específicos.....	15
CAPÍTULO 2	17
2. MARCO TEÓRICO.....	17
2.1.1. Parámetros importantes en la calidad del agua para Piscinas Camaroneras.	18
2.5.1. Duración de baterías en redes LoraWan.....	25
2.5.2. Seguridad en redes LoraWan	26
2.5.3. Capacidad de redes LoraWan.....	26
2.5.4. Frecuencias de operación de redes LoraWan	27
2.5.5. Clases de Dispositivos usados en de redes LoraWan.....	29
2.5.6. Modulación CSS en redes LoRaWAN.....	31
2.5.7. Parámetros Característicos las de redes LoraWan.....	33
2.5.8. Impacto del Spreading Factor en Transmisiones LoraWan.....	42
CAPÍTULO 3	43
3. DISEÑO Y SIMULACIÓN DE LA SOLUCIÓN IoT	43

3.1.	Diseño de la solución IoT.....	43
3.1.1.	Reconocimiento del espacio de operación de la solución IoT.	43
3.1.2.	Elección de la tecnología a utilizar en el diseño de la solución	44
3.1.3.	Ubicación de los elementos de la red.	44
3.1.4.	Arquitectura de la red.	46
3.1.5.	Presupuesto de enlace.	46
3.1.6.	Cálculo de zona de Fresnel.	50
3.1.7.	Diagrama de Bloques y esquemas del sistema.	52
3.1.8.	Descripción de los componentes de los LoRa End Points.	56
3.1.9.	Descripción de los componentes de los LoRa GateWays.	60
3.2.	Simulación de la solución IoT	61
3.2.1.	Simulación de enlaces en Radio Mobile.	61
3.2.2.	Simulación de la solución IoT en Cisco Packet Tracer.	66
	Configuración de Ambientes en Cisco Packet Tracer.	66
	Configuración de canales y campos en la plataforma ThingSpeak.	67
	Programación de MCUs.	69
	Programación de Sensores Especializados.	73
	CAPÍTULO 4	75
4.	ANÁLISIS DE RESULTADOS	75
4.1.	Resultados obtenidos en los cálculos de diseño de la red.	75
4.2.	Resultados mostrados en plataforma ThingSpeak a través de la simulación en Packet Tracer.	78
	Comparativa entre los valores sensados y los valores recibidos en ThingSpeak durante la ejecución de la simulación	80
	CONCLUSIONES Y RECOMENDACIONES	83
	BIBLIOGRAFÍA	84

ÍNDICE DE FIGURAS

FIGURA 2.1 – CAMARONERAS EN EL ECUADOR [3].....	17
FIGURA 2.2 – MEDICIÓN MANUAL DE PARAMETROS DE CALIDAD DE AGUA [11]	18
FIGURA 2.3 – INTERNET OF THINGS [12]	20
FIGURA 2.4 – PLATFORM AS A SERVICE [13].....	22
FIGURA 2.5 – Tecnología LoRa [22].....	23
FIGURA 2.6 - Capas LoRa y LoraWAN [22].....	24
FIGURA 2.7 – Esquema de operación usual de redes LoRaWAN [22].....	25
FIGURA 2.8 Esquema de trabajo de dispositivos clase A [24]	30
FIGURA 2.9 Esquema de trabajo de dispositivos clase B [24]	30
FIGURA 2.10 Esquema de trabajo de dispositivos clase C [24]	31
FIGURA 2.11 Grafica en el tiempo de un Up-Chirp [24]	32
FIGURA 2.12 Grafica Frecuencia Vs Tiempo de un UpChirp y un DownChirp [24]	32
FIGURA 2.13 Grafica Frecuencia Vs Tiempo de una señal modulada CSS [24]	33
FIGURA 2.14 Aumento de tiempo de simbolo por incremento de SF [21]	37
FIGURA 2.15 Sensibilidad del receptor ante incremento de SF [24].....	39
FIGURA 2.16 Formatos de Paquete LoraWAN [24]	39
FIGURA 3.1 – Grupo de Piscinas Camaroneras para el diseño de la solución IoT	43
FIGURA 3.2 – Ubicaciones de LoRa Gateway y LoRa End Points.....	45
FIGURA 3.3 Enlaces End Points – LoRa Gateways	52
FIGURA 3.4 – Esquema de la solución IoT.....	52
FIGURA 3.5 – Diagrama de bloques de la solución IoT	52
FIGURA 3.6 – Esquema del Transmisor.....	53
FIGURA 3.7 – Diagrama de bloques del Transmisor.....	53
FIGURA 3.8 – Esquema del Receptor	54

FIGURA 3.9 – Diagrama de bloques del Receptor	55
FIGURA 3.10 – Sensor Digital de Temperatura Adafruit Waterproof DS18B20 [15].	56
FIGURA 3.11 – Sensor Gravity de oxígeno disuelto [16]	57
FIGURA 3.12 – Sonda Atlas Scientific Detectora de Niveles de pH [17].	58
FIGURA 3.13 – Sensor Libelium de niveles de NH4 [18].	58
FIGURA 3.14 – Shield Dragino LoRa [19].	59
FIGURA 3.8 – Simulación de Link 1	61
FIGURA 3.9 – Simulación de Link 2	62
FIGURA 3.10 – Simulación de Link 3	62
FIGURA 3.11 – Simulación de Link 4	63
FIGURA 3.12 – Simulación de Link 5	63
FIGURA 3.13 – Simulación de Link 6	64
FIGURA 3.14 – Simulación de Link 7	64
FIGURA 3.15 – Simulación de Link 8	65
FIGURA 3.16 – CONFIGURACIÓN DE PARÁMETROS	67
FIGURA 3.17 – ESTABLECIMIENTO DE CANALES EN THINGSPEAK	68
FIGURA 3.18 – CONFIGURACIÓN DE CAMPOS EN THINGSPEAK	69
FIGURA 3.19 – Topología de le Red Sensorial	70
FIGURA 3.20 – Conexión MCU - Sensores	70
FIGURA 3.21 – FRAGMENTO DE PROGRAMACIÓN DE MCUs	72
FIGURA 3.22 – FRAGMENTO DE PROGRAMACIÓN DE MCUs	72
FIGURA 3.23 – FRAGMENTO DE PROGRAMACIÓN DE SENSOR ESPECIALIZADO.	73
FIGURA 4.1 – Simulación del Enlace 6	77
FIGURA 4.2 – RESULTADOS EN THINGSPEAK PARA PISCINA 1	78

FIGURA 4.3 – RESULTADOS EN THINGSPEAK PARA PISCINA 2	79
FIGURA 4.4 – RESULTADOS EN THINGSPEAK PARA PISCINA 3	79
FIGURA 4.5 – VALORES CAPTADOS POR SENSORES EN PISCINA1.	80
FIGURA 4.6 – VALORES CAPTADOS POR SENSORES EN PISCINA2.	81
FIGURA 4.7 – VALORES CAPTADOS POR SENSORES EN PISCINA3.	82

ÍNDICE DE TABLAS

Tabla 1: Niveles Adecuados de Parámetros de Calidad de Agua [9]	19
Tabla 2: Disminución de tasa de datos por aumento de SF.	36
Tabla 3: Límites de SNR por aumento de SF	37
Tabla 4: Sensibilidad del Receptor por aumento de SF.....	38
Tabla 5: Distancias entre End Points y Lora Gateway	46
Tabla 6: Pérdidas en Espacio Libre entre End Points y Lora Gateway.....	48
Tabla 7: Pérdidas por el modelo de trayectoria de 3 rayos	49
Tabla 8: Pérdidas Totales	50
Tabla 9: Radios de Zona de Fresnel.....	51
Tabla 10: Resumen de resultados del capítulo 3	76
Tabla 11: Diferencia entre Link Budget y Pérdidas	76

ABREVIATURAS

IoT	INTERNET OF THINGS
PaaS	PLATFORM AS A SERVICE
MCU	MICRO CONTROLLER UNIT
IP	INTERNET PROTOCOL
LoRa	LONG RANGE
CSS	CHIRP SPREAD SPECTRUM
LPWAN	LOW POWER WIDE AREA NETWORK
AES	ADVANCED ENCRYPTION STANDARD
SF	SPREADING FACTOR
ISM	INDUSTRIAL SCIENTIFIC MEDICAL
CRC	CYCLIC REDUNDANCY CHECK

CAPÍTULO 1

1. INTRODUCCIÓN

1.1. Descripción del Problema

La calidad del agua en la cría de camarones es de vital importancia debido a que los camarones son criaturas muy delicadas, susceptibles a sufrir estrés cuando las condiciones de su ambiente son adversas; en condiciones de estrés estas especies tienden a enfermarse, a no crecer de la forma adecuada y hasta llegar a morir [9]. Por ello es necesario e imperativo un constante monitoreo de los parámetros que definen la buena o mala calidad del agua en las piscinas camaroneras para que los acuicultores que administran las mismas puedan tomar las acciones pertinentes de forma oportuna para así evitar tener bajo crecimiento de las crías, crías enfermas o las pérdidas definitivas de las mismas [7].

En el país aún no está muy difundido el uso de las tecnologías IoT para monitoreo remoto 24/7 de los parámetros que describen la buena o mala calidad del agua para la cría de camarones; de hecho, estas mediciones se siguen haciendo de forma manual lo cual implica una escasa efectividad para la prevención de los problemas consecuentes con la mala calidad del agua [3]. La presente propuesta se suma como un inicio a una solución que busca llenar este vacío existente en la operación de la cría de camarones en el país.

1.2. Justificación de la propuesta.

Dado que la calidad de agua se suma importancia en la cría de camarones, es muy importante que se desarrollen diseños de soluciones tecnológicamente viables para que los parámetros que describen la calidad del agua pueden ser constantemente monitoreados y así las empresas camaroneras puedan tener la

necesaria capacidad de reacción cuando el ambiente no es el adecuado en dichas piscinas y aplicar los correctivos que se requieran, logrando de esta forma aumentar la eficiencia de la producción disminuyendo pérdidas por enfermedades o muertes de los camarones.

El diseño de una red de sensores debe involucrar como punto importante la fiabilidad de información y optimización de recursos, puntos los cuales son reunidos por la tecnología LoRa (Long Range) la cual permite que sus dispositivos se puedan desempeñar con bajo consumo de energía, grandes rangos de comunicación y bajo costo económico frente a otras tecnologías.

1.3. Objetivos.

1.3.1. Objetivo General.

Diseñar y Simular una solución IoT mediante el uso una red sensorial especializada en captar los valores de los parámetros más importantes relacionados a la calidad del agua en piscinas de cría de camarón con la finalidad de que dichos parámetros puedan estar disponibles para su monitoreo remoto a cualquier hora del día.

1.3.2. Objetivos Específicos.

- Programar el comportamiento de sensores simulados para captar los parámetros relacionados con la calidad del agua en piscinas camaroneras.
- Realizar un estudio de la propagación de la señal enviada por los LoRa EndPoints que estarán en las piscinas camaroneras.
- Configurar el comportamiento a través del tiempo de los parámetros ambientales que tendrán lugar en las piscinas camaroneras simuladas para que la información captada sean lo más cercana a la realidad.

- Exteriorizar la información captada por los sensores en la simulación hacia la solución Cloud Thing Speak con la finalidad de poder monitorear cada parámetro a través del tiempo.

CAPÍTULO 2

2. MARCO TEÓRICO

2.1. Sector Camaronero en el Ecuador

El sector camaronero en el país es uno de los principales motores de la producción e impulsores de la economía nacional, siendo junto al petróleo y el banano uno de los sectores que más divisas atraen a la nación, una muestra de ello es que el sector camaronero del Ecuador desde enero a mayo del presente año generó ventas al exterior estimadas en 1.663 millones de dólares americanos, generando empleos para miles de personas [1].



FIGURA 2.1 – CAMARONERAS EN EL ECUADOR [3]

Es un hecho que el sector camaronero es sin duda uno de los sectores más importantes de la economía nacional representando un 18% de las exportaciones no petroleras del Ecuador [2], por ello debe ser impulsado y tecnificado haciendo uso de las tecnologías disponibles para así optimizar recursos de producción y minimizar las pérdidas. En la Figura 2.1 podemos apreciar un conjunto de piscinas camaroneras que operan en el golfo de Guayaquil.

2.1.1. Parámetros importantes en la calidad del agua para Piscinas Camaroneras.

En la cría continental de camarones la calidad del agua es un punto en extremo importante debido a que es el medio ambiente en el que los tales subsisten, se alimentan, y se desarrollan.

Las inversiones realizadas para la producción de camarones son montos muy importantes [3] los cuales experimentan un factor de riesgo en el caso de que la producción se llegue a perder o no llegue a alcanzar los objetivos propuestos; es por ello por lo que es de vital importancia el monitorear, analizar y controlar que los parámetros que reflejan la calidad del agua se encuentren en los niveles óptimos para la producción. En la Figura 2.2 podemos apreciar una toma de mediciones de parámetros de calidad de agua de forma manual en una piscina camaronera durante operaciones de aeración.



FIGURA 2.2 – MEDICIÓN MANUAL DE PARAMETROS DE CALIDAD DE AGUA [11]

A continuación, en la Tabla 1 se mostrará cuáles son los parámetros más importantes que definen la buena o mala calidad del agua para la cría de camarones y los niveles en los cuales la producción es la más óptima [9].

Parámetro	Unidad	Niveles Adecuados
Temperatura del agua.	C	18 - 28
Concentración de Oxígeno(O ₂).	mg/L	5 - 15
Potencial de Hidrogeno (PH).		7 - 9
Concentración de Nitrógeno. (Amonio Ionizado NH ₄)	mg/L	0.2 - 2
Concentración de sulfuro. (Sulfato S ₀₄)	g/L	0.5 - 3
Concentración de Dióxido de Carbono (CO ₂)	mg/L	1 - 10
Concentración de Cloro (Cl ⁻)	g/L	2 - 20
Concentración de calcio (Ca ²⁺)	g/L	0.1 - 0.5
Concentración de magnesio (Mg ²⁺)	g/L	0.1 - 1.5
Concentración de sodio (Na ⁺)	g/L	2 - 11
Concentración de Potasio (K ⁺)	g/L	0.1 - 0.4
Concentración de Bicarbonato (HCO ₃ ⁻)	g/L	0.075 - 0.3
Concentración de Carbonato	mg/L	0 - 20
Concentración de Silicio	mg/L	2 - 20
Concentración de Hierro	mg/L	0.05 - 0.5
Concentración de Cobre (Cu ²⁺).	mg/L	0.005 - 0.1

Tabla 1: Niveles Adecuados de Parámetros de Calidad de Agua [9]

Los siguientes son los parámetros que captaran los sensores en la solución que se diseña en el presente trabajo.:

- A. Temperatura del agua.
- B. Concentración de Oxígeno(O₂).
- C. Potencial de Hidrogeno (pH).
- D. Concentración de Amonio (NH₄).

2.2. El Internet de las Cosas.

El Internet de las cosas (IoT) se ha convertido en una de las nuevas tendencias en el mercado de soluciones tecnológicas.

Sin temor a exagerar podemos decir que IoT es una tecnología importante con aplicaciones en muchos campos. IoT tiene sus raíces

los costos operativos; además los edificios inteligentes también pueden usar sensores para monitorear su salud estructural.

Las ciudades inteligentes usan sensores para monitorear el tráfico de peatones y vehículos y pueden integrar datos de edificios inteligentes.

Los vehículos utilizan sensores en red para monitorear el estado del vehículo y proporcionar desempeño mejorado, menor consumo de combustible y menores emisiones de gases.

Los sistemas médicos conectan una amplia gama de sensores de monitorización de pacientes que pueden estar ubicados en sus hogares, en vehículos de emergencia, en el consultorio del médico o en el hospital.

2.4. Soluciones Cloud PaaS.

PaaS es un tipo de solución cloud denominado como Platform as a Service o Plataforma como servicio; siendo un entorno de desarrollo e implementación que se opera íntegramente desde la nube, este tipo de entornos o plataformas con sus recursos pueden soportar desde aplicaciones sencillas basadas en Cloud hasta aplicaciones empresariales ampliamente sofisticadas y habilitadas para la nube [4]. Adicional al uso del almacenamiento y utilización de otros recursos informáticos, los usuarios estarán habilitados para operar un set de herramientas prediseñadas para personalizar desarrollar y probar sus aplicaciones propias. La Figura 2.4 esquematiza los campos de acción de las soluciones PaaS.

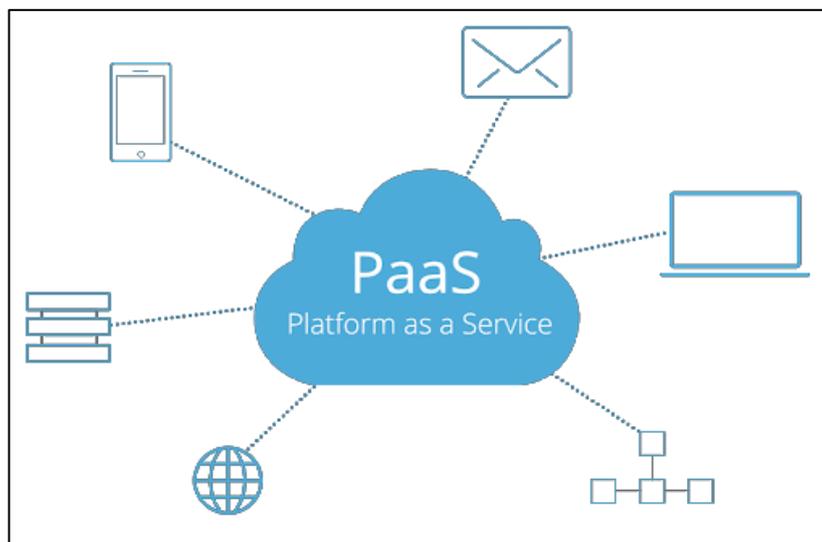


FIGURA 2.4 – PLATFORM AS A SERVICE [13]

PaaS brinda un entorno o una plataforma con herramientas para desarrollar, probar y alojar aplicaciones dentro del mismo entorno.

Este tipo de soluciones o servicios habilita a las organizaciones y usuarios a concentrarse en el desarrollo, y despreocuparse por la infraestructura subyacente a sus proyectos.

Los proveedores de este tipo de servicio se encargan de gestionar la seguridad, el software de servidor los sistemas operativos, y los respaldos copias de seguridad [5].

Además, da facilidades para la colaboración aún si los equipos operan en forma remota.

2.5. Tecnología LoRa Wan

LoRa hace referencia a un esquema de capa física o de modulación inalámbrica útil para conseguir largos enlaces de comunicación. LoRa se basa en la utilización de modulación Chirp Spread Spectrum (CSS), manteniendo las mismas características de bajo consumo de potencia que el esquema de modulación FSK pero aumentando ampliamente el rango de comunicación. Chirp Spread Spectrum ha sido utilizado anteriormente en comunicaciones de tipo militar y de tipo espacial

durante varias décadas por las largas distancias de comunicación que se logran conseguir además de la robustez antes las interferencias.

La gran ventaja de la tecnología LoRa reside en su largo alcance, un solo LoRa Gateway o estación base podría cubrir ciudades enteras dependiendo de sus características de propagación, es decir la cantidad y tipos de obstrucciones que puedan existir en un determinado lugar.



FIGURA 2.5 – Tecnología LoRa [22].

LoRaWAN se define como el protocolo de comunicación y la arquitectura para red teniendo mayor influencia en la determinación de vida útil de la batería de un nodo final, la capacidad de la red, la seguridad y la calidad del servicio [22].

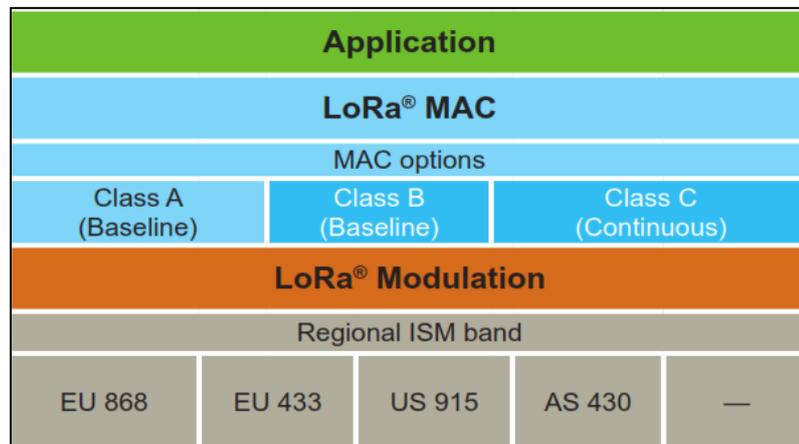


FIGURA 2.6 - Capas LoRa y LoraWAN [22].

Tal como podemos ver en la Figura 2.6 Lora tiene que ver con la capa física o la modulación CSS para la transmisión de la señal y LoRaWAN tiene que ver con la capa superior en la cual ese gestiona la capacidad de red, la vida útil de la batería del nodo final y la seguridad de la comunicación tal como se mencionó anteriormente [22].

Los elementos que comúnmente podemos encontrar en una red Lora son los Lora End Point o nodos finales Lora y también los denominados LoRa Gateway. Un Lora End Point puede estar conformado por sensores conectados a un microcontrolador el cual se conecta a un módulo LoRa con su respectiva antena. Existen en el mercado LoRa Gateway que están listos para ser programados, ubicados y utilizados, pero también los LoRa Gateway pueden ser contruidos mediante el uso de microcontroladores y módulos Lora con su respectiva antena.

La FIGURA 2.7 describe el esquema común de desempeño de una red LoraWan la cual estaría conformada por las siguientes cuatro partes:

- Lora End Nodes
- Lora Gateways.
- Network Server.
- Application Server.

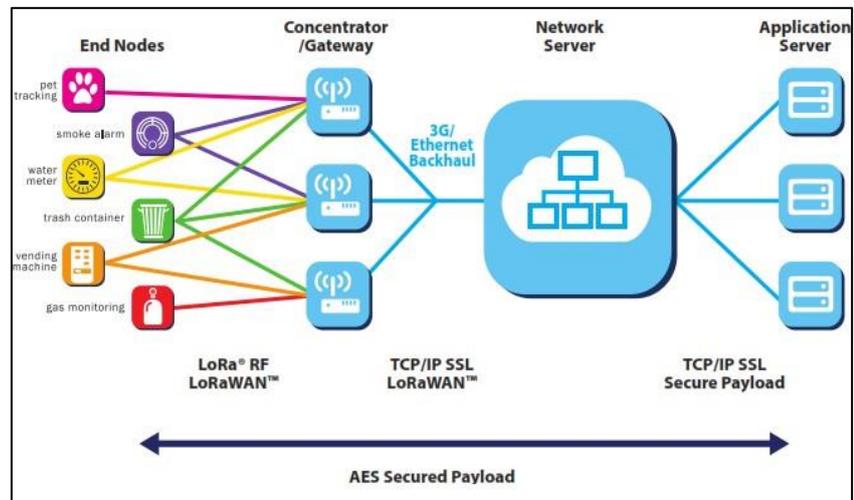


FIGURA 2.7 – Esquema de operación usual de redes LoRaWAN [22]

En una red LoRaWAN, los End Points no siempre están asociados un Gateway específico. En su lugar, los datos transmitidos por un End Point son recibidos usualmente por múltiples Gateways, realizando cada uno de ellos un reenvío del o los paquetes recibidos desde el nodo final hacia la solución Cloud utilizada como Network Server.

El Network Server o servidor de red es la etapa que se encarga de gestionar y filtrar los paquetes recibidos redundantes, realizar controles de seguridad y enviar la información al Application Server para que la información esté disponible para su visualización.

2.5.1. Duración de baterías en redes LoraWan.

Los End Points en una red LoRaWAN trabajan en modo asíncrono y estableciendo comunicación solamente cuando tienen datos listos para ser enviados ya sea por la intervención de un evento o por programación.

2.5.2. Seguridad en redes LoraWan.

Es de extrema importancia para cualquier LPWAN la incorporación de seguridad. LoRaWAN hace uso de 2 capas de seguridad: una capa para la red y otra capa para la aplicación.

La capa de seguridad de red busca garantizar la autenticidad de los End Nodes en la red, en cambio la capa de seguridad de aplicación busca garantizar solamente el usuario final cuenta con acceso a los datos transmitidos. Se utiliza encriptación AES (Advanced Encryption Standard) con el intercambio de llaves (Key) usando un identificador IEEE EUI64.

2.5.3. Capacidad de redes LoraWan.

Para que una red de largo alcance en topología estrella sea viable, su puerta de enlace o Gateway debe tener la capacidad de recibir mensajes de altos volúmenes de nodos finales.

La alta capacidad una red LoRaWAN es alcanzada por medio de la utilización de tasas de datos adaptables y mediante el uso de un transceiver multi-módem multi-canal en los Gateways para que se logre recibir mensajes simultáneos en múltiples canales. [24]

Los factores críticos para la capacidad efectiva son el número de canales concurrentes, la tasa de datos la cual está relacionada con el tiempo en el aire de los mensajes, la

la longitud de la carga útil de los datos y la frecuencia con la que transmitirán los nodos finales.

Dado que LoRa está basado en modulación de espectro ensanchado, las señales transmitidas son prácticamente ortogonales entre si cuando diferentes Spreading Factors (SF) son utilizados. A medida que hay cambios en el Spreading Factor, la tasa de datos en la transmisión también cambia. Los LoRa Gateway aprovechan esta propiedad al así poder recibir

múltiples tasas de datos diferentes al mismo tiempo en el mismo canal de comunicación.

Si un End Node tiene un buen enlace y está cerca de un Gateway, no hay razón para que utilice siempre la tasa de datos más baja y utilice el espectro disponible por más tiempo que el necesario. Al aumentar la tasa de datos, el tiempo en el aire de la transmisión se acorta dando más espacio para que otros nodos finales puedan transmitir. La tasa de datos adaptable ayuda además a optimizar la vida útil de la batería de los

Nodos finales. Para que la tasa de datos adaptable funcione, tanto el enlace ascendente como el descendente deben ser simétricos.

Lo antes mencionado refleja que las redes LoraWan son redes de gran capacidad y escalabilidad.

2.5.4. Frecuencias de operación de redes LoraWan.

Las especificaciones en cuanto al uso de frecuencias para operación de redes LoRaWAN varía ligeramente de una región a otra según las diferentes asignaciones regionales de espectro y requisitos reglamentarios. De manera general podemos decir que la tecnología LoRaWAN opera en las bandas ISM (Industrial Scientific Medical) de 433, 868 y 915 MHz trabajando con los Duty Cycles definidos en las normativas de cada región.

En Estados Unidos, Canadá, Singapur, Israel, Australia y Sudamérica se hace uso de la banda ISM de 915MHz [20], continuación se detallan los canales de dicha banda con sus respectivos anchos de banda y spreading factor definidos.

Plan de Frecuencias US902-928:

Comunicación ascendente:

903.9 - SF7BW125 to SF10BW125

904.1 - SF7BW125 to SF10BW125

904.3 - SF7BW125 to SF10BW125

904.5 - SF7BW125 to SF10BW125

904.7 - SF7BW125 to SF10BW125

904.9 - SF7BW125 to SF10BW125

905.1 - SF7BW125 to SF10BW125

905.3 - SF7BW125 to SF10BW125

904.6 - SF8BW500

Comunicación Descendente:

923.3 - SF7BW500 to SF12BW500 (RX1)

923.9 - SF7BW500 to SF12BW500 (RX1)

924.5 - SF7BW500 to SF12BW500 (RX1)

925.1 - SF7BW500 to SF12BW500 (RX1)

925.7 - SF7BW500 to SF12BW500 (RX1)

926.3 - SF7BW500 to SF12BW500 (RX1)

926.9 - SF7BW500 to SF12BW500 (RX1)

927.5 - SF7BW500 to SF12BW500 (RX1)

923.3 - SF12BW500 (RX2)

Plan de Frecuencias AU915-928:

Comunicación ascendente:

916.8 - SF7BW125 to SF12BW125

917.0 - SF7BW125 to SF12BW125

917.2 - SF7BW125 to SF12BW125

917.4 - SF7BW125 to SF12BW125

917.6 - SF7BW125 to SF12BW125

917.8 - SF7BW125 to SF12BW125

918.0 - SF7BW125 to SF12BW125

918.2 - SF7BW125 to SF12BW125

917.5 - SF8BW500

Comunicación ascendente:

923.3 - SF7BW500 to SF12BW500 (RX1)

923.9 - SF7BW500 to SF12BW500 (RX1)

924.5 - SF7BW500 to SF12BW500 (RX1)

925.1 - SF7BW500 to SF12BW500 (RX1)

925.7 - SF7BW500 to SF12BW500 (RX1)

926.3 - SF7BW500 to SF12BW500 (RX1)

926.9 - SF7BW500 to SF12BW500 (RX1)

927.5 - SF7BW500 to SF12BW500 (RX1)

923.3 - SF12BW500 (RX2)

2.5.5. Clases de Dispositivos usados en de redes LoraWan.

Cada Lora End Point puede ser utilizado en diversas áreas de aplicaciones las cuales pueden tener diferentes requerimientos. LoRaWAN utiliza diferentes clases de dispositivos en base al uso que tendrán los mismos en la red LoRaWAN. A continuación, se detallan las diferentes clases de dispositivos finales que podemos encontrar en la tecnología LoRaWAN:

Dispositivos Clase A

Son dispositivos finales que tienen comunicación bidireccional permitiendo que luego de que una transmisión de enlace ascendente es efectuada haya dos ventanas de recepción de enlace descendente de corta duración, esto lo podemos apreciar claramente en la Figura 2.8. De este modo las

comunicaciones del servidor en cualquier otro momento tendrán que esperar hasta que el próximo enlace ascendente programado tome lugar [24].

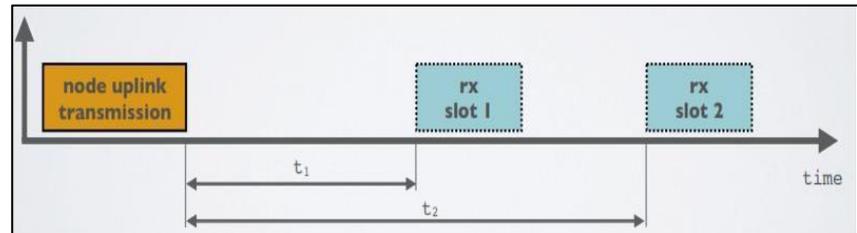


FIGURA 2.8 Esquema de trabajo de dispositivos clase A [24]

Dispositivos Clase B

Son dispositivos finales bidireccionales con ventanas de recepción programadas además de las ranuras de recepción de los dispositivos Clase A, figura 2.9, es decir que los dispositivos de Clase B establecen ventanas de recepción adicionales en tiempos los cuales son programados. Para que el dispositivo final abra su ventana de recepción en la hora programada, recibe una señal sincronizada en el tiempo desde el Gateway, esto permite que el servidor de red sepa cuándo el dispositivo final está en modo de escucha [24].

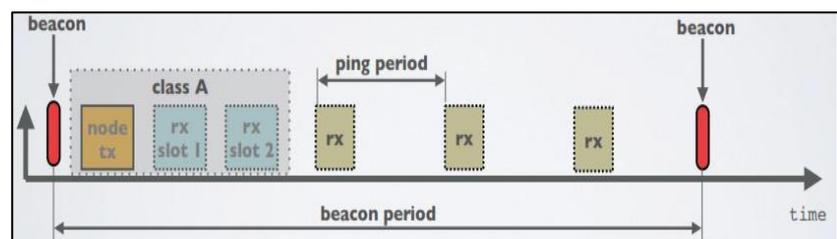


FIGURA 2.9 Esquema de trabajo de dispositivos clase B [24]

Dispositivos Clase C

Estos son dispositivos finales bidireccionales con ventanas de recepción máximas, es decir, tienen ventanas de recepción abiertas casi continuamente, y solo son cerradas durante la transmisión. Esto es apreciable en la Figura 2.10

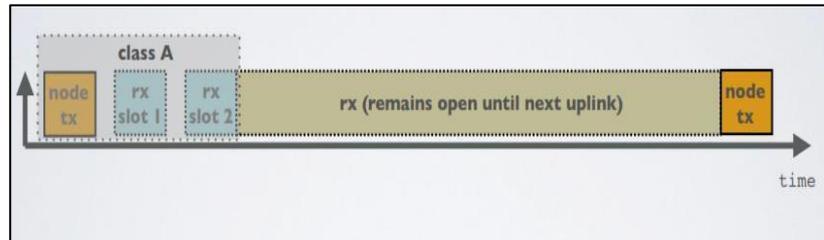


FIGURA 2.10 Esquema de trabajo de dispositivos clase C [24]

2.5.6. Modulación CSS en redes LoRaWAN.

LoRa es un esquema de modulación de espectro ensanchado patentado que se basa en Chirp Spread Spectrum (CSS) el cual es una técnica de espectro extendido que utiliza banda ancha lineal pulsos de chirrido de frecuencia modulada para codificar información.

Las técnicas de espectro ensanchado son métodos mediante los cuales una señal se difunde deliberadamente en el dominio de la frecuencia. Por ejemplo, una señal se transmite en ráfagas cortas, saltando entre frecuencias en una secuencia pseudoaleatoria.

Un chirp, a menudo llamado señal de barrido, es un tono en el que la frecuencia aumenta (upchirp) Figura 2.11 o disminuye (down-chirp) con el tiempo.

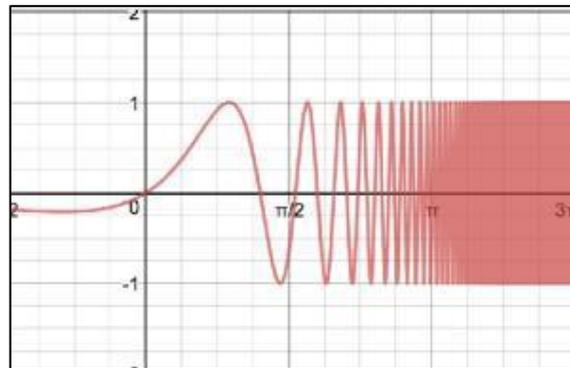


FIGURA 2.11 Grafica en el tiempo de un Up-Chirp [24]

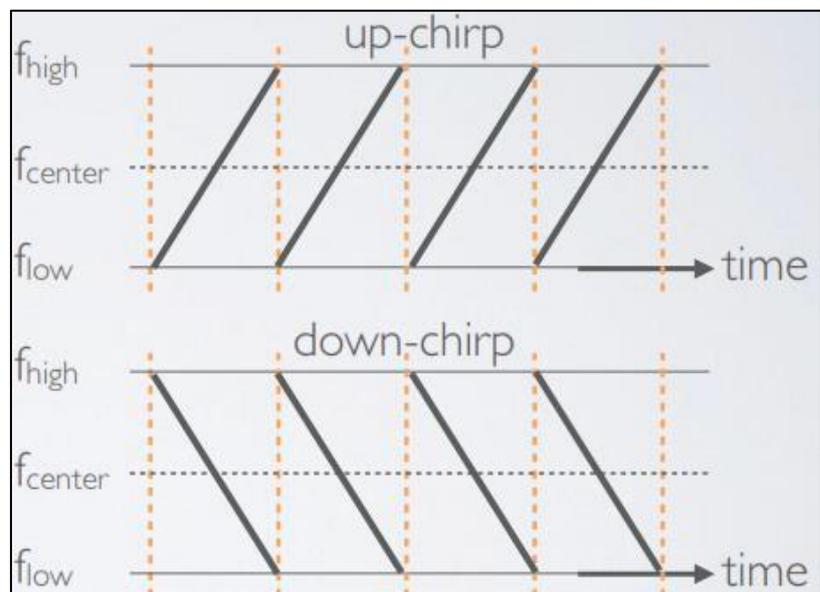


FIGURA 2.12 Grafica Frecuencia Vs Tiempo de un UpChirp y un DownChirp [24]

La Figura 2.12 muestra el comportamiento en el tiempo de un Down-Chirp y un Up-Chirp, mientras que en la Figura 2.13 podemos observar de forma visual una señal modulada CSS notando los valores que toman en frecuencia a través del tiempo representando así la información que es transmitida mediante las distintas formas que pueden tomar los Chirps.

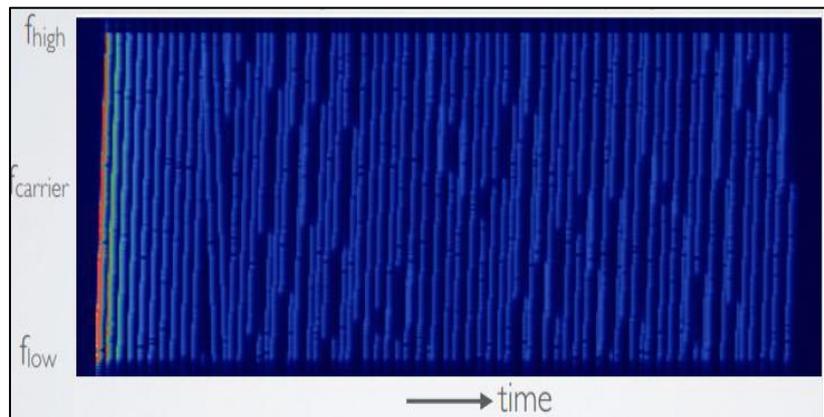


FIGURA 2.13 Grafica Frecuencia Vs Tiempo de una señal modulada CSS [24]

2.5.7. Parámetros Característicos las de redes LoraWan.

A continuación, se especificarán algunos de los parámetros que son característicos de las redes LoRaWAN y qué tienen que ser considerados según el caso de uso de estas de estas redes.

Tiempo en el Aire (ToA: Time on Air)

Es la cantidad de tiempo que la antena del transmisor está energizada y transmitiendo datos [21].

Ciclo de Trabajo (Duty cycle)

Es la proporción de tiempo durante el cual un componente, dispositivo o sistema está operando [22]. El ciclo de trabajo se puede expresar como una relación o como un porcentaje.

Por ejemplo, para un determinado valor del ToA de una transmisión con ciclo de trabajo es del 1%, el dispositivo tendrá que esperar un tiempo equivalente a 99 veces el ToA para poder transmitir nuevamente.

El ciclo de trabajo es definido por la regulación de cada región en la banda de frecuencias correspondiente.

SPREADING FACTOR (SF)

Es la cantidad de bits con el cual podemos representar la información que va a ser transmitida a través de la red LoRaWAN. Por ejemplo, si estuviéramos interesados en representar el número decimal 95 como un símbolo binario requeriríamos de una cantidad de 7 bits para poder representarlo de la siguiente manera 1011111, de manera sencilla podríamos decir que el Spreading Factor para dicha representación es igual a 7.

En las transmisiones LoraWAN el Spreading Factor puede ir desde 7 hasta 12 [22].

FORWARD ERROR CORRECTION (FEC)

La corrección de errores hacia adelante (FEC) es el proceso en el que se agregan bits de corrección de errores a los datos transmitidos.

Estos bits redundantes ayudan a restaurar los datos cuando los datos se corrompen por interferencias. Si se agregan más bits de corrección de errores, más fácilmente se pueden corregir los datos [24].

Sin embargo, al agregar más bits de corrección de errores, se transmiten más datos, lo que reduce la vida útil de la batería.

CODING RATE (CR)

El Coding Rate o la tasa de codificación se refiere a la proporción de bits transmitidos que realmente transportan información.

Valores de tasa de codificación permitidos por LoRa: CR = 4/5, 4/6, 4/7 o 4/8 [21].

CHIP RATE (Tasa de Chip)

La unidad de medida del ancho de banda (BW) es el Hertz (Hz), el cual representa el número de vibraciones o ciclos de onda por segundo. El valor numérico del ancho de banda es

intercambiable con la tasa de chip, es decir: $BW \rightarrow R_C$, la unidad de medida de la tasa de chips es: chips /s.

SYMBOL RATE (Tasa de Símbolos)

La tasa de símbolos en transmisiones LoRaWAN es calculada de la siguiente manera:

$$R_S = \frac{BW}{2^{SF}} \quad (2.1)$$

Donde BW es el ancho de banda del canal de comunicación y SF es el Spreading Factor de la transmisión. Su unidad de medida son los símbolos por segundo.

La tasa de Chip siempre será mayor que la tasa de símbolos.

DATA RATE (Tasa de Datos)

La tasa de datos en transmisiones LoRaWAN es calculada de la siguiente manera [21]:

$$R_D = \frac{BW}{2^{SF}} \cdot \frac{4}{4+CR} \quad (2.2)$$

Donde BW es el ancho de banda del canal de comunicación, SF es el Spreading Factor de la transmisión y CR es el Coding Rate el cual puede ser un valor entero entre 1 y 4. Su unidad de medida son los bits por segundo.

Podemos notar que la velocidad de la transmisión LoRaWAN depende directamente del valor del Spreading Factor, es decir, que si el Spreading Factor aumenta, la tasa de datos disminuye, esto se implica que habrá un mayor tiempo en el aire para la transmisión, por ende esquemas con Spreading Factor mayores se usan para transmisiones a distancias mayores;

mientras que, cuando el Spreading Factor es menor, la tasa de datos aumenta y el tiempo en el aire disminuye, siendo este el esquema usado para transmisiones hacia distancias más cortas.

Podemos notar lo antes descrito en la Tabla 2 donde se muestran las tasas de datos obtenidas al variar los valores Spreading Factor de 7 a 12, manteniendo fijo el valor del ancho de banda a 125 KHz y el valor del coding rate en 1.

SF	DATA RATE (Kbps)
7	5,47
8	3,13
9	1,76
10	0,98
11	0,54
12	0,29

Tabla 2: Disminución de tasa de datos por aumento de SF.

SYMBOL DURATION (Tiempo de Símbolo)

El tiempo de símbolo es calculada de la siguiente manera [24]:

$$T_s = \frac{2^{SF}}{BW} \quad (2.3)$$

Donde BW es el ancho de banda del canal de comunicación y SF es el Spreading Factor de la transmisión.

Una descripción general de la duración de los símbolos con respecto a diferentes valores de Spreading Factor es la que podemos observar en la Figura 2.14 donde notamos que, si el SF aumenta en uno, la duración del símbolo se duplica.

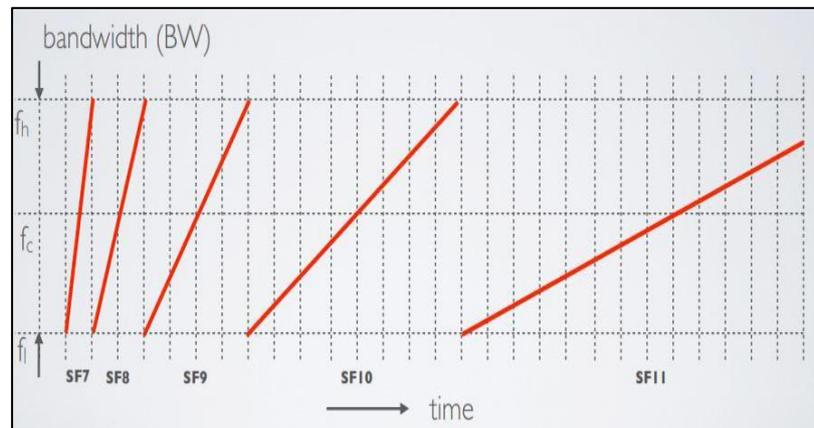


FIGURA 2.14 Aumento de tiempo de símbolo por incremento de SF [21]

LÍMITE DE SNR

Para cada Spreading Factor hay un límite SNR, si se alcanza este límite, el receptor no podrá demodular la señal [21].

En la tabla 3, se puede encontrar el límite de SNR para cada SF:

SF	Símbolos	Límite de SNR (dB)
7	128	-7,5
8	256	-10
9	512	-12,5
10	1024	-15
11	2048	-17,5
12	4096	-20

Tabla 3: Límites de SNR por aumento de SF

Cabe notar que, si el SF aumenta en 1, el límite SNR cambia en -2,5 dB.

SENSIBILIDAD DEL RECEPTOR

La sensibilidad del receptor varía directamente en función del ancho de banda del canal de transmisión, la figura de ruido del receptor y el límite la relación señal ruido; esto podemos ver a través de la ecuación 2.4:

$$S = -174 + 10\log(Bw) + NF + SNR \text{ limit} \quad (2.4)$$

Donde, la sensibilidad del receptor (S) es medida en dBm.

El ancho de banda (BW) es medido en Hz.

La Figura de ruido del receptor (NF) en dB, siendo este un valor fijo para un determinado hardware.

Para chipsets transceptores de nodo final LoRa SX1272 y SX1276 la figura de ruido es 6 dB [24].

En la tabla 4 se muestra como conforme el SF aumenta, el receptor se vuelve más sensible y por ende es capaz de captar señales más débiles, esto es utilizado para transmisiones a largas distancias.

SF	Límite de SNR (dB)	Sensibilidad del Receptor (dBm)
7	-7,5	-124,53
8	-10	-127,03
9	-12,5	-129,53
10	-15	-132,03
11	-17,5	-134,53
12	-20	-137,03

Tabla 4: Sensibilidad del Receptor por aumento de SF

Lo detallado en la tabla 4 también es presentado de forma visual en la figura 2.15 donde podemos notar que al aumentar el SF, el receptor tiene un comportamiento de mayor sensibilidad para la captación de señales.



FIGURA 2.15 Sensibilidad del receptor ante incremento de SF [24]

Si la distancia entre el nodo final y el LoRa Gateway aumenta, la señal se debilita y por lo tanto, se necesita un SF mayor para que una sensibilidad del receptor más baja sea capaz de demodular la señal recibida. Los valores de SF varían desde 7 cuando el nodo final está cerca de una LoRa Gateway a 12 cuando el nodo final está lejos del LoRa Gateway.

FORMATO DEL PAQUETE LORA

Hay dos tipos de modos de formato de paquete LoRa tal como se puede observar en la Figura 2.16.

El modo de encabezado explícito incluye un encabezado corto que contiene información sobre la longitud de la carga útil, la tasa de codificación y si se utiliza un CRC en el paquete.

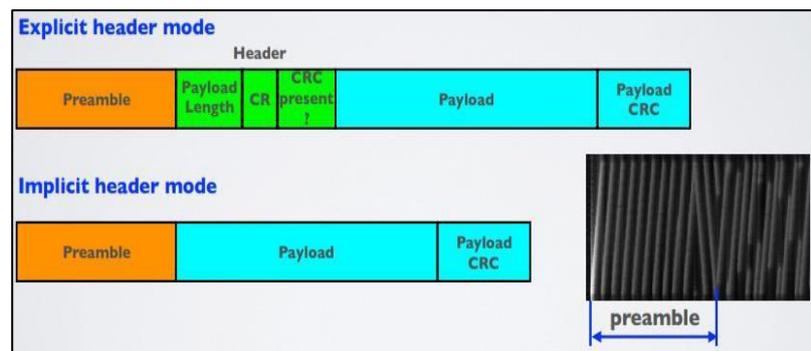


FIGURA 2.16 Formatos de Paquete LoraWAN [24]

CRC significa Cyclic Redundancy Check y se utiliza para detectar errores en datos digitales.

El modo de encabezado implícito donde la carga útil, la tasa de codificación (CR) y la presencia de CRC son fijos. En este modo, el encabezado se elimina del paquete, lo que reduce tiempo de transmisión. En este caso, la longitud de la carga útil, la tasa de codificación de errores y la presencia del CRC de carga útil debe configurarse manualmente en ambos lados del enlace de radio.

El preámbulo se utiliza para detectar el inicio del paquete por parte del receptor.

El encabezado (solo en el modo de encabezado explícito) es el modo de funcionamiento predeterminado [24].

CALCULO DEL TIEMPO EN EL AIRE

El tiempo en el aire es igual al tiempo el paquete y esto puede ser calculado en la siguiente manera [24]:

$$T_{\text{paquete}} = T_{\text{preámbulo}} + T_{\text{payload}} \tag{2.5}$$

Tanto el tiempo de preámbulo como el tiempo de payload medido en segundos, siendo estos calculados en la siguiente forma [24]:

$$T_{\text{preámbulo}} = (T_s \cdot r \cdot 8 \cdot 12) + 4.25 \cdot T_s \tag{2.6}$$

$$T_{\text{payload}} = T_s \cdot (8 + 2 \cdot x \cdot (i \cdot (P - 4) + 16 - 20) / 4 \cdot (2 - 2 \cdot x) \cdot (2 + 4), 0)) \tag{2.7}$$

Donde (Ts) es el tiempo del símbolo en segundos.

PL es la carga útil en bytes.

CRC (activado = 1, desactivado = 0). Para LoRaWAN elCRC predeterminado es 1.

En cuento al encabezado (inhabilitado $H = 1$ [implícito], habilitado $H = 0$ [explícito]). Para el encabezado LoRaWAN está habilitado: $H = 0$

LowDataRateOptimize (Habilitado $DE = 1$, deshabilitado $DE = 0$)

Tasa de codificación ($CR = 1,2,3$ o 4). Para LoRaWAN el CR predeterminado es 1 .

LOW DATA RATE OPTIMIZATION

Dada la duración potencialmente larga del paquete en SF altos, la opción Low Data Rate Optimization se puede configurar para mejorar la robustez de la transmisión a variaciones de frecuencia durante la transmisión y recepción del paquete.

Cuando la optimización de baja velocidad de datos está habilitada, aumenta la solidez del enlace LoRa a estas bajas velocidades de datos.

El uso Low Data Rate Optimization es obligatorio cuando la duración del símbolo supera los 16 ms. Hay que tener en cuenta que tanto el transmisor como el receptor deben tener la misma configuración para LowDataRateOptimize [24].

LowDataRateOptimize está habilitado para ancho de banda 125 kHz y $SF > = 11$.

ADAPTIVE DATA RATE

A continuación, se detalla una explicación simplificada de cómo funciona Adaptive Data Rate.

El servidor de red recopila los 20 datos de transmisiones de enlace ascendente más recientes como la tasa de datos y relación señal / ruido desde un nodo final.

De los 20 mensajes de enlace ascendente recibidos, el servidor de red toma la SNR máxima, valor que ahora se llama SNRmedido y la tasa de datos correspondiente.

El servidor de red calcula el margen de la siguiente manera:

$$r_{medido} = SNR_{medido} - SNR_{requisito} -$$

$$M_{medido} \quad (2.8)$$

Si el margen es grande, significa que el nodo final usa demasiada batería innecesaria para poder para transmitir mensajes de enlace ascendente [24].

2.5.8. Impacto del Spreading Factor en Transmisiones LoraWan.

En base a lo visto en los puntos anteriores podemos notar que si aumenta el SF en 1 tendremos los siguientes resultados:

- La duración del símbolo se duplicaría en comparación con el SF anterior.
- La tasa de bits se reduciría aproximadamente a la mitad en comparación con el SF anterior.
- El tiempo en el aire (ToA) o la cantidad de tiempo que la antena del transmisor está energizada y transmitiendo datos aumenta, lo que significa que la distancia del alcance será mayor.

Los dispositivos LoRa utilizan un SF más alto cuando la señal es débil o hay mucha interferencia. El uso de un SF más alto significa un tiempo en el aire (ToA) más largo.

Si un Lora End Point está más lejos de un LoRa Gateway, la señal se debilita y, por lo tanto, necesita un SF más alto.

CAPÍTULO 3

3. DISEÑO Y SIMULACIÓN DE LA SOLUCIÓN IoT

3.1. Diseño de la solución IoT

3.1.1. Reconocimiento del espacio de operación de la solución IoT.

Para diseñar una solución telecomunicaciones es importante conocer y considerar todas las características del medio o el lugar en el cual funcionaria la solución para en base a toda la información recolectada poder escoger los modelos matemáticos que más se apeguen ambiente en el cual operará nuestra solución y así realizar los cálculos correspondientes, análisis y toma de decisiones.

Para la solución descrita a continuación se escogió un grupo de 8 piscinas camaroneras ubicadas en el cantón Guayaquil en el kilómetro 3.5 de la avenida Las Esclusas, el área total aproximada que abarcan estas 8 piscinas camaroneras es de 121.66 hectáreas tal como se muestra en la figura 3.1.

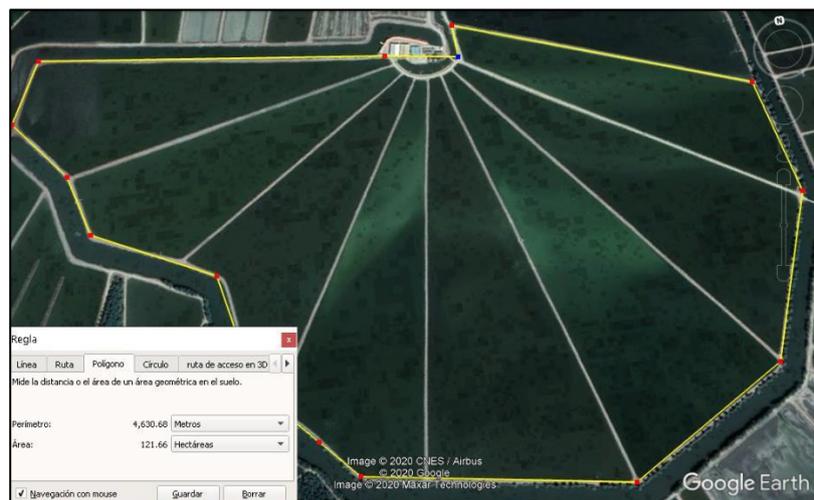


FIGURA 3.1 – Grupo de Piscinas Camaroneras para el diseño de la solución IoT

Cabe recalcar que los ambientes de piscinas camaroneras son lugares bastante despejados y que no presentan mayores obstrucciones en la línea de vista entre los dispositivos de la red a diseñar, lo cual es bastante favorable para el diseño y operación de soluciones IoT inalámbricas, debido a las grandes masas de agua que se encuentran en la zona se debe tomar en cuenta la utilización de un modelo de propagación lo más cercano posible a este ambiente.

3.1.2. Elección de la tecnología a utilizar en el diseño de la solución.

Como ya se mencionó anteriormente debido a la extensa área que abarca este grupo de piscinas camaroneras se ha optado por hacer uso de la tecnología LoRaWAN la cual es una tecnología de comunicaciones a largo alcance que puede establecer coberturas de distancias inclusive mayores a 15 km en campos abiertos sin obstrucciones, además de que es una tecnología muy eficiente en cuanto al uso de la energía garantizando así que las baterías que llegaren a energizar los elementos de la red pueda funcionar de manera prolongada por varios años.

3.1.3. Ubicación de los elementos de la red.

En la figura 3.2 puede observar las ubicaciones en las cuales se propone deberían estar tanto el Lora GetWay como los como cada uno de los ocho Lora End Points que estarían ubicados en las respectivas piscinas.

La ubicación propuesta para el LoRa GateWay se justifica en que es un punto central, equidistante a las piscinas camaroneras, dónde se encuentra una bodega de alimentos, así como también medicamentos para las crías, juntamente con una pequeña oficina con acceso a internet el cual sería útil para brindar conectividad al dispositivo Lora GateWay.

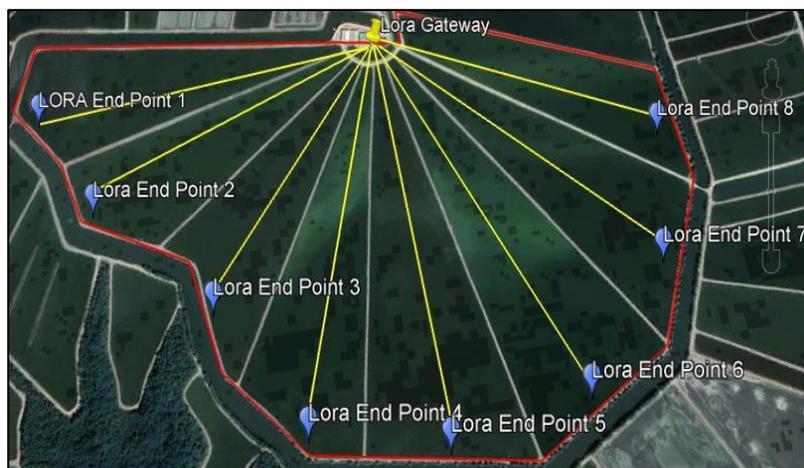


FIGURA 3.2 – Ubicaciones de LoRa Gateway y LoRa End Points.

La ubicación de los LoRa End Points en los extremos externos de cada piscina se justifica en el hecho de que en esa área de las piscinas se encuentran ubicadas las compuertas de salida de agua. Normalmente las áreas cercanas a las entradas de agua de las piscinas son las áreas con mejor calidad de agua y por ende mayor contenido de oxígeno y las áreas cercanas a las salidas de agua de las piscinas son las áreas con menor contenido de oxígeno, por ello se propone realizar las mediciones en esas zonas ya que es más probable que allí las valoraciones de concentración de oxígeno lleguen a sus puntos críticos (menores a 5 mg/l).

Las piscinas camaroneras en nuestro medio por lo general se diseñan y construyen con profundidades de aproximadamente 1.2 metros, los camarones al ser especies bentónicas normalmente se encuentran la parte más profunda de la piscina, por ello se recomienda ubicar los sensores que irán conectados a los LoRa End Points a una profundidad de al menos de entre 80 centímetros y 1 metro ya que es allí donde se da el consumo de oxígeno por parte de los camarones.

La tabla 5 a continuación detalla cada una de las distancias que se dan entre el dispositivo Lora GateWay y cada uno de los Lora End Points que se ubicarían en las respectivas piscinas camarones.

Distancias EndPoints - LoRa GateWay			
Punto Inicial	Punto Final	Nombre del Link	Distancia (m)
LoRa Gateway	LoRa End Point 1	Link 1	865
LoRa Gateway	LoRa End Point 2	Link 2	756
LoRa Gateway	LoRa End Point 3	Link 3	684
LoRa Gateway	LoRa End Point 4	Link 4	838
LoRa Gateway	LoRa End Point 5	Link 5	874
LoRa Gateway	LoRa End Point 6	Link 6	925
LoRa Gateway	LoRa End Point 7	Link 7	742
LoRa Gateway	LoRa End Point 8	Link 8	752

Tabla 5: Distancias entre End Points y Lora Gateway

3.1.4. Arquitectura de la red.

En cuanto a la arquitectura de esta red, como la gran mayoría de soluciones IoT que hacen uso de la tecnología LoRaWAN. Se ha adoptado la arquitectura en estrella la cual está caracterizada por un dispositivo intermedio o central (Lora Gateway) el cual tiene comunicación con dispositivos extremos (Lora End Points).

Esta arquitectura brinda la posibilidad de escalar el tamaño de la red conforme a las a los requerimientos que pueden surgir a través del tiempo, además de que esta topología aporta en cuanto al minimizar fallos a nivel de todo el sistema ya que si alguno de los nodos finales llegase a fallar el resto de los nuevos seguirían operando.

3.1.5. Presupuesto de enlace.

Para el cálculo del presupuesto de enlace máximo en Redes LoRaWAN se debe considerar los valores de la potencia de

transmisión (Pt) y el valor de la sensibilidad del receptor (S) tal como se lo describe en la ecuación 3.1.

$$P_t - L_{total} = S \quad (3.1)$$

El esquema de modulación LoRa es muy robusto ante las interferencias, pero puede llegar a ser vulnerable al ruido por los bajos valores de transmisión que se manejan, por ello se debe calcular la sensibilidad del receptor en base a los niveles más críticos de SNR manejables por la tecnología LoRa, siendo este valor -20dB. Para el cálculo de la sensibilidad haremos uso de la ecuación 2.4

$$S = -174 + 10\log(Bw) + NF + SNR \text{ limit} \quad (2.4)$$

Conociendo que:

BW= 125KHz

NF= 6dB para chips Lora SX1275

SNRlimit= -20dB

Tenemos que:

$$S = -174 + 10\log(125000) + 6 - 20$$

$$S = -137 \text{ dBm}$$

Ya que conocemos el valor de la máxima sensibilidad del receptor podemos calcular en Link Budget máximo sabiendo que la potencia transmitirse 20 dbm.

$$P_t - L_{total} = S \quad (3.1)$$

$$20 - L_{total} = -137$$

$$L_{total} = 157$$

Con este resultado del presupuesto máximo de enlace podemos notar desde el transmisor hasta el receptor se podría admitir hasta 157dB de perdida sin que existan problemas en la demodulación de la señal recibida.

Para el cálculo de pérdidas en el espacio libre se hace uso de la ecuación 3.2

$$L_{FS} = 32.45 + 20 \log(D) + 20 \log(f)$$

(3.2)

Donde “D” la distancia del transmisor al receptor en kilómetros y “f” es la frecuencia de transmisión en megahercios.

Sabiendo que la frecuencia de operación en 915 MHz y las distancias de separación entre los nodos finales y el LoRa GateWay varían entre 684 y 925 metros, a continuación, la tabla 6 mostrará los resultados del desarrollo de los cálculos de pérdidas de espacio libre para cada uno de los ocho enlaces.

Perdidas de Espacio Libre			
Nombre del Link	Distancia (Km)	Frecuencia (MHz)	L_FS (dB)
Link 1	0,865	915	90,42
Link 2	0,756	915	89,25
Link 3	0,684	915	88,38
Link 4	0,838	915	90,14
Link 5	0,874	915	90,51
Link 6	0,925	915	91,00
Link 7	0,742	915	89,09
Link 8	0,752	915	89,20

Tabla 6: Perdidas en Espacio Libre entre End Points y Lora Gateway

A esta pérdida por espacio libre le debe ser sumada además las pérdidas que se podrían obtener por factores del propio entorno siendo en este caso son los propios de las piscinas camaroneras los cuales son ambientes con poca obstrucción, pero con grandes cantidades de dando lugar a la refracción de las ondas propagadas.

En la superficie de estas masas de agua casi todo el tiempo están presente los denominados ductos de evaporación los cuales generan refracción de las señales que son propagadas. Por lo ello, se ha considerado un modelo de trayectoria de tres rayos el cual contempla dichos efectos, tomando en

consideración las ondas atrapadas en un ducto troposférico, la señal reflejada sobre la superficie de los cuerpos de agua y la señal con línea de vista directa (LOS) [25]. Este modelo obedece a la ecuación 3.3 descrita a continuación.

$$P_{\text{B0}} \{ (\frac{\lambda}{4\pi d})^2 [2(1 + \Delta)]^2 \} \tag{3.3}$$

Donde:

$$\Delta = 2 \sin(\frac{2\pi h_t h_r}{\lambda}) \sin(\frac{2\pi (h_e - h_t)(h_e - h_r)}{\lambda}) \tag{3.4}$$

De estas ecuaciones indicamos que d es la distancia en entre transmisor y receptor ht y hr son las alturas del transmisor y del receptor respectivamente, con alturas de 1m y 10m para este diseño, y he es la altura de la capa del ducto troposférico, el cual varía entre 20 y 40 dependiendo de los meses del año.

La tabla 7 a continuación mostrará los resultados cálculo que pérdidas por el modelo de trayectoria de 3 rayos para cada uno de los enlaces.

Perdidas por modelo de Tres Rayos				
Nombre del Link	Distancia (Km)	Lamda (m)	Delta	L 3 Ray (dB)
Link 1	0,865	0,328	0,70	19,79
Link 2	0,756	0,328	1,66	14,72
Link 3	0,684	0,328	-0,73	33,61
Link 4	0,838	0,328	0,92	18,45
Link 5	0,874	0,328	0,29	22,28
Link 6	0,925	0,328	0,28	22,85
Link 7	0,742	0,328	1,14	16,45
Link 8	0,752	0,328	-0,56	30,38

Tabla 7: Perdidas por el modelo de trayectoria de 3 rayos

A continuación, la tabla 8 muestra los resultados totales de las pérdidas generadas tanto por espacio libre como por el modelo de trayectoria de 3 rayos.

Sumatoria de Perdidas			
Nombre del Link	L_FS (dB)	L 3 Ray (dB)	Perdida Total (dB)
Link 1	90,42	19,79	110,21
Link 2	89,25	14,72	103,97
Link 3	88,38	33,61	121,99
Link 4	90,14	18,45	108,60
Link 5	90,51	22,28	112,79
Link 6	91,00	22,85	113,85
Link 7	89,09	16,45	105,53
Link 8	89,20	30,38	119,58

Tabla 8: Pérdidas Totales

Como podemos notar, ninguno de los enlaces tiene perdidas mayores a máximo Link Budget (157dB), por lo que podemos indicar que la señal sería capaz de llegar desde los nodos finales hasta el LoRaGateWay con la suficiente intensidad de señal para tener comunicación optima.

3.1.6. Cálculo de zona de Fresnel.

La zona de Fresnel es una de forma elíptica alrededor de la línea de visión directa entre el nodo final y la puerta de enlace la cual agrupa los diversos haces que son transmitidos en el medio inalámbrico. Cualquier obstáculo dentro de este volumen puede debilitar la señal transmitida incluso aunque haya una línea de visión directa entre el extremo nodo y la puerta de enlace.

El radio máximo de la zona de Fresnel está ubicado a la mitad de la distancia entre el nodo final y el Gateway se calcula según lo indica la ecuación 3.5:

$$r = 8.657 \sqrt{\frac{D}{f}} \quad (3.5)$$



Donde D es la distancia entre punto y punto en kilómetros y f es la frecuencia de operación en GHz.

Como regla general, la zona de Fresnel debe estar siempre libre de obstrucciones, pero esto puede ser poco práctico, por lo que se dice que más allá del 40% de bloqueo, la pérdida de señal será significativa. Esta estimación puede ser calculada según lo describe la ecuación 3.6.

$$r(40\%) = \frac{8.657 \cdot D^{0.6}}{\sqrt{f}} \quad (3.6)$$



La tabla 9 presenta los resultados el radio de la zona de Fresnel considerando 0% de obstrucción y 40% de obstrucción para cada uno de los ocho enlaces.

Nombre del Link	Distancia (Km)	Radio de Zona de Fresnel (m)	Radio de Zona de Fresnel (m) al 40% de Obs
Link 1	0,865	8,42	6,52
Link 2	0,756	7,87	6,10
Link 3	0,684	7,48	5,80
Link 4	0,838	8,28	6,42
Link 5	0,874	8,46	6,55
Link 6	0,925	8,70	6,74
Link 7	0,742	7,80	6,04
Link 8	0,752	7,85	6,08

Tabla 9: Radios de Zona de Fresnel

En base a los resultados obtenidos y en base a una simulación efectuada en el software Radio Mobile, para este diseño se define que los LoRa End Points estén ubicados a una altura de un metro mientras que el LoRa GateWay esté ubicado a una altura de diez metros. Los resultados de la simulación en Radio Mobile se mostrarán más adelante en el presente documento.

En la figura 3.3 se puede apreciar varios de los enlaces de que tendrían lugar en las piscinas camaroneras, donde se puede apreciar el contorno de la zona de Fresnel y la línea directa de vista con las alturas indicadas en el párrafo anterior. Podemos observar que la zona de Fresnel es obstruida en una pequeña

porción la cual no representa afectaciones de acuerdo con la simulación efectuada.

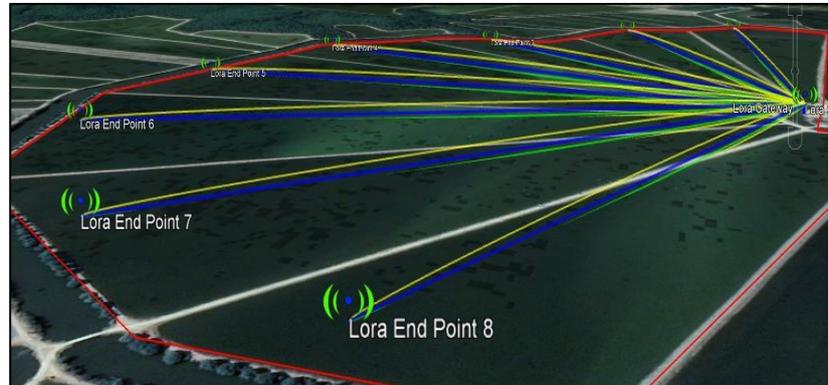


FIGURA 3.3 Enlaces End Points – LoRa Gateways

3.1.7. Diagrama de Bloques y esquemas del sistema.

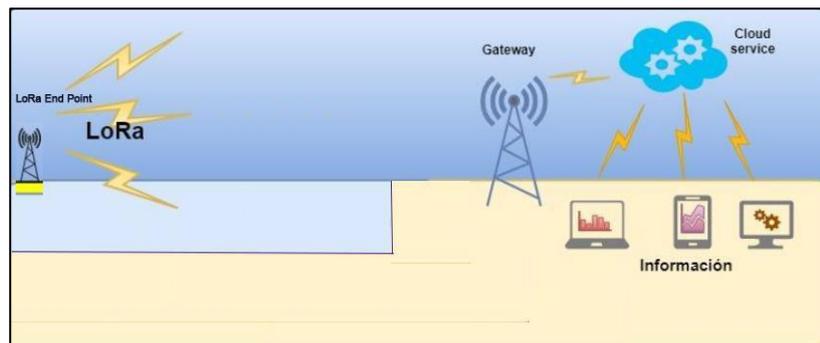


FIGURA 3.4 – Esquema de la solución IoT

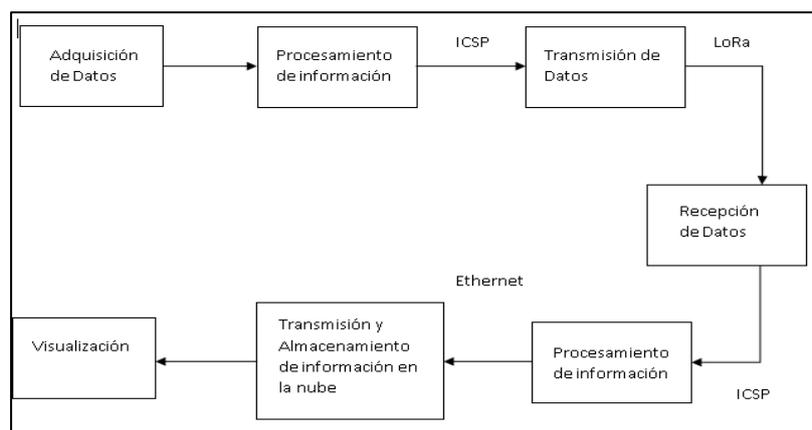


FIGURA 3.5 – Diagrama de bloques de la solución IoT

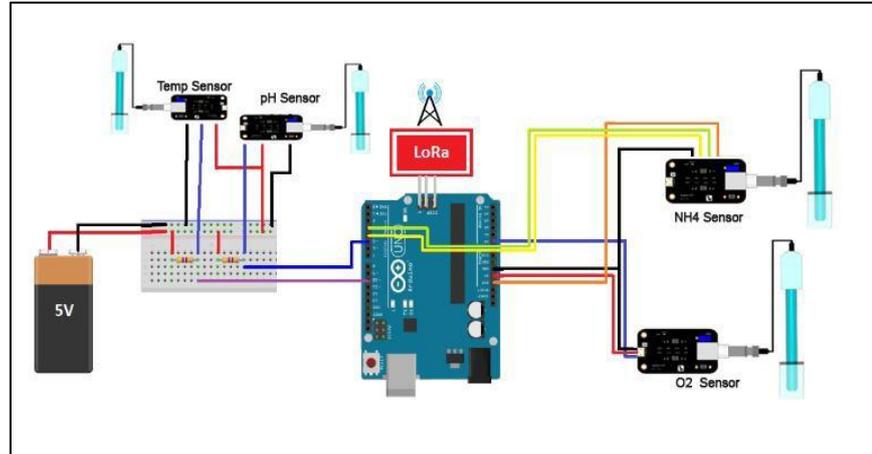


FIGURA 3.6 – Esquema del Transmisor

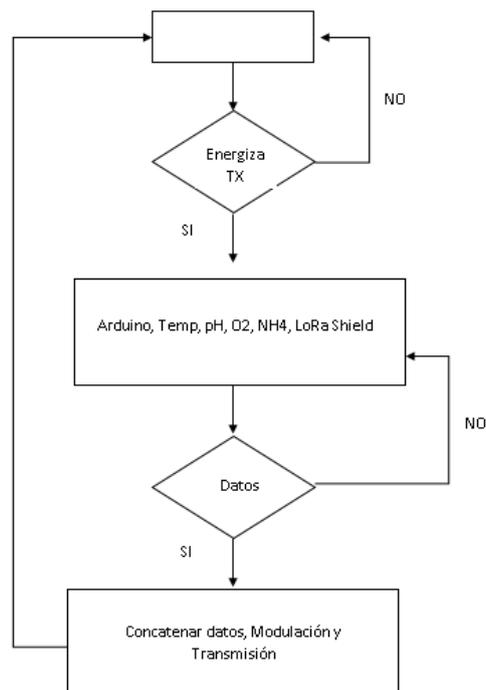


FIGURA 3.7 – Diagrama de bloques del Transmisor

Energiza TX: Siempre que la fuente de alimentación en el Transmisor opere sin problemas el sistema deberá operar

adecuadamente, si no es así se, entrara en estado de fuera de servicio.

Arduino, Temp, pH, O2 MH4, LoRaShield: Tanto el Arduino como sus módulos en estado de operación juntamente con los sensores.

Datos: si algún sensor no opera adecuadamente, no captará datos, no afectaría a los demás, y volvería a operar una vez se haya reparado o reemplazado.

Concatenar datos, modulación y transmisión: siempre que se generen datos, Arduino estará encargado de efectuar la concatenación de los mismos para que luego esa data será preparada y modulada por LoRa para la posterior transmisión.

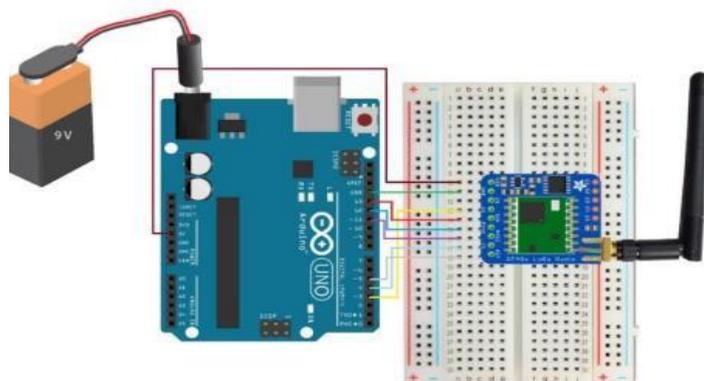


FIGURA 3.8 – Esquema del Receptor

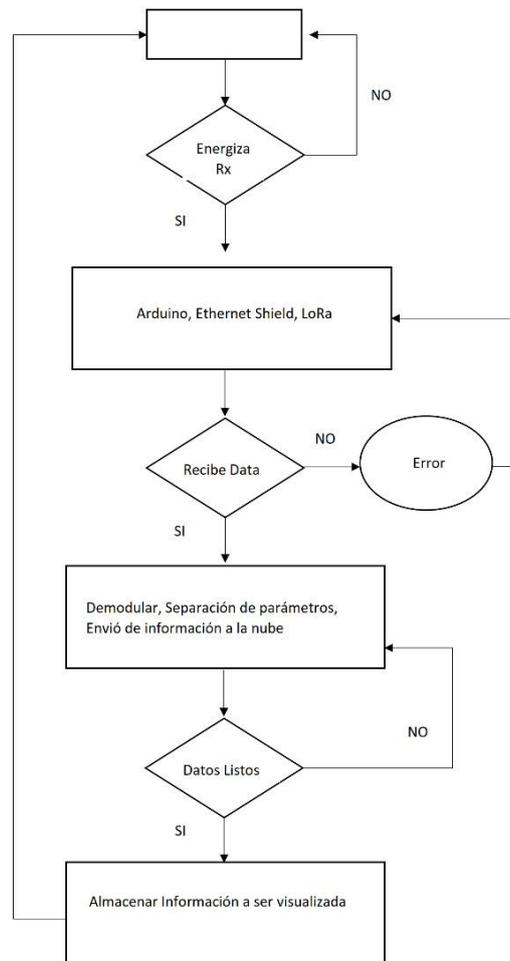


FIGURA 3.9 – Diagrama de bloques del Receptor

Energiza Rx: Siempre que la fuente de alimentación en el receptor opere sin problemas el sistema deberá operar adecuadamente, si no es así se, entrara en estado de fuera de servicio.

Arduino, Ethernet Shield, LoRa: Tanto el Arduino como sus módulos en estado de operación.

Recibe Data: Si los datos no arriban al receptor, se activaría una señal alertando un error para así tomar todas las medidas necesarias para corregirlo, caso contrario se da continuidad al proceso de recepción.

Demodular, Separación de parámetros, Envió de información a la nube: La señal es recibida y demodulada por el LoRa Shield para que luego Arduino realice la separación de parámetros que serán subidos a la nube vía ethernet.

Datos listos: Se realiza la verificación de si los datos están preparados para ser enviados.

Almacenar información a ser visualizada: La data que ha sido enviada estará a la disponibilidad de los usuarios finales del sistema.

3.1.8. Descripción de los componentes de los LoRa End Points.

En la presente parte de este documento se procede a detallar los diferentes componentes sugeridos para la implementación de los LoRa End Points.

Sensor Digital de Temperatura Adafruit Waterproof DS18B20



FIGURA 3.10 – Sensor Digital de Temperatura Adafruit Waterproof DS18B20 [15].

Este sensor digital de la marca Adafruit es capaz de captar temperaturas de entre -50°C y 125°C , cuenta con una gran precisión con un grado de tolerancia al error de $\pm 0,5^{\circ}\text{C}$ y pueden proporcionar hasta 12 bits de precisión en su convertidor digital a analógico integrado, es útil en sistemas de 3.0-5.0V [15].

El sensor cuenta con 3 terminales, dos de ellas para alimentación y un terminal para comunicación de datos. El Sensor se encuentra protegido por una capsula de acero inoxidable por lo que es resistente al ambiente las piscinas camaroneras al estar sumergido perennemente. Su apariencia física se puede observar en la Figura 3.10

Sensor Gravity de oxígeno disuelto compatible con Arduino



FIGURA 3.11 – Sensor Gravity de oxígeno disuelto [16].

Este producto se usa para hacer mediciones del oxígeno disuelto en el agua, para reflejar la calidad de esta, este sensor es compatible con los microcontroladores Arduino. Se aplica ampliamente en muchas aplicaciones de calidad del agua, como el monitoreo del medio ambiente y la acuicultura.

Está compuesto por una sonda galvánica, no necesita tiempo de polarización y permanece disponible en cualquier momento. Cuenta con un rango de detección de 0 a 20 mg / L y puede ser sumergido hasta dos metros [16]. La apariencia física de este sensor puede ser apreciada en la Figura 3.11.

Sonda Atlas Scientific Detectora de Niveles de pH

La sonda de pH Atlas Scientific de grado de consumidor proporciona lecturas precisas en un amplio rango de pH. Esta sonda ha sido diseñada para funcionar en aplicaciones

comunes de lectura de pH. Esta sonda funcionará bien para la mayoría de las aplicaciones generales de detección de pH [17].



FIGURA 3.12 – Sonda Atlas Scientific Detectora de Niveles de pH [17].

Entre las principales características sonda detectora de niveles de pH tenemos:

Rango: 2 - 13

Precisión: $\pm 0,1$

Tiempo de respuesta: 95% en 4 s

Rango de temperatura: 1 - 60 ° C

Sensor Libelium de niveles de NH₄



FIGURA 3.13 – Sensor Libelium de niveles de NH₄ [18].

Es muy importante captar el amonio ionizado en las piscinas camaroneras, ya que el mismo es tóxico para los camarones puede provocar directamente la muerte. El amonio ionizado se acumula cuando hay un equilibrio poco saludable de desechos

o alimentos no consumidos en la o las piscinas camaroneras [9]. Entre las características del sensor mostrado en la figura 3.13 tenemos [18]:

Ion: Ion de amonio (NH_4^+)

Sensibilidad: -54 ± 5

Temperatura ($^{\circ}\text{C}$): 5-50

Rango lineal: 0,09 - 9000 mg / L

Arduino

Arduino es un tipo de plataforma electrónica basado en software y hardware libre que implementa el lenguaje Processing/Wiring. Es capaz de recibir en sus entradas señales digitales y analógicas para ser procesadas, existen en el mercado un inmenso número de módulos y sensores compatibles con Arduino lo que viabiliza la implementación de prototipos complejos y útiles en diversos campos [26].

Existen varias versiones de Arduino, cada uno de ellos se diferencia por sus características y capacidades, para el diseño de estos LoRa End Points se recomienda emplear Arduino UNO.

Shield Dragino LoRa



FIGURA 3.14 – Shield Dragino LoRa [19].

El LoRa Shield de Dragino es un transceptor de largo alcance, basado en una biblioteca de código abierto. El LoRa Shield permite el envío de datos alcanzando rangos bastante largos con tasas de datos bajas.

Este LoRa Shield está basado en los chips SX1276 y SX1278 para uso en redes de sensores inalámbricos de tipo profesional, como medición inteligente, sistemas de riego, ciudades inteligentes, automatización de edificios, etc.

Utiliza modulación LoRa la cual está patentada de Semtech, este LoRa Shield puede alcanzar una sensibilidad de alrededor de -148dBm. Esta alta sensibilidad en combinación con su amplificador de potencia de +20 dBm brinda un Link Budget líder en la industria, lo que lo convierte en óptimo para el uso en cualquier aplicación en la que se requiera rango o robustez [19]. La apariencia física de este LoRa Shield se muestra en la figura 3.14.

3.1.9. Descripción de los componentes de los LoRa GateWays.

En cuanto a los componentes del Lora GateWay tenemos un microcontrolador Arduino en su versión UNO y un Lora Shield Dragino. Estos componentes ya fueron detallados ampliamente en el punto anterior, la diferencia entre LoRaGateway y un LoRa End point es que el LoRaGateway está programado para reenviar la información que ha recibido por parte de los End Points hacia una plataforma cloud para que los datos pueden ser presentados al usuario final.

3.2. Simulación de la solución IoT

3.2.1. Simulación de enlaces en Radio Mobile.

En la presente parte de este trabajo se mostrarán los resultados obtenidos al simular los radioenlaces constatando esta forma que los mismos tendrían buena operatividad en cuanto a intensidad de la señal a nivel de recepción además de que podemos verificar que las alturas que fueron definidas para colocar tanto los nodos finales como el LoRa GateWay dan lugar a que el impacto por obstrucción en la zona de Fresnel no sea perjudicial para cada uno de los enlaces.

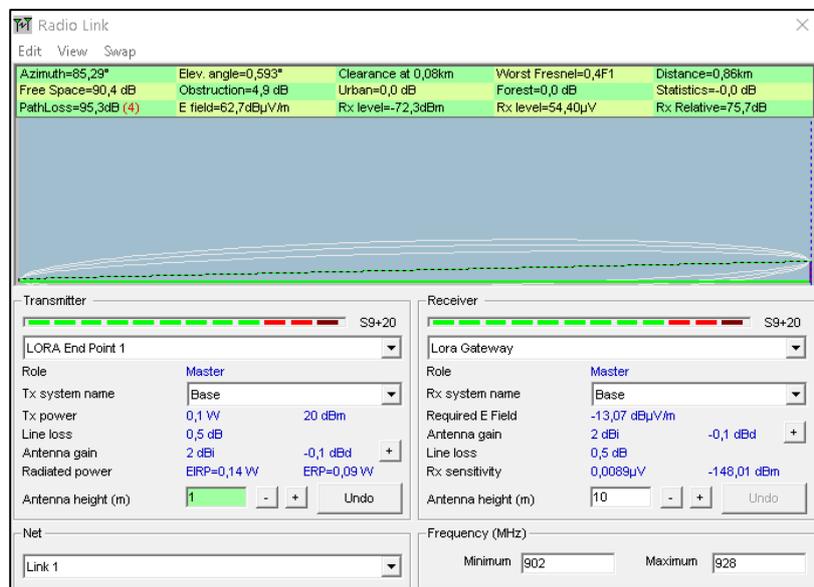


FIGURA 3.8 – Simulación de Link 1

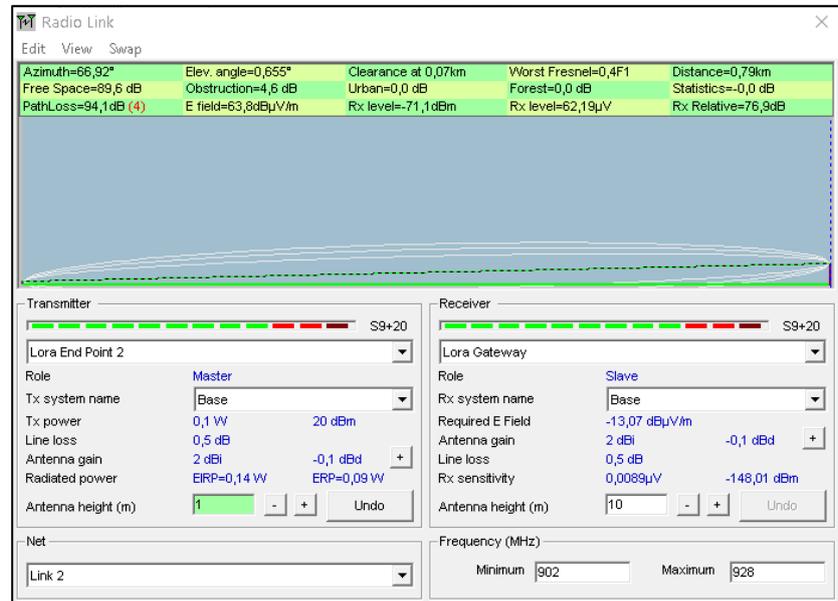


FIGURA 3.9 – Simulación de Link 2

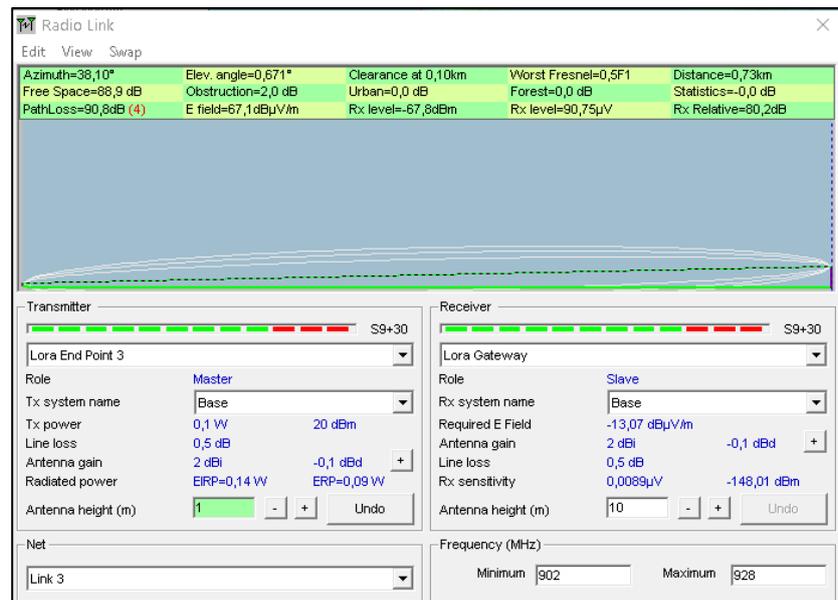


FIGURA 3.10 – Simulación de Link 3

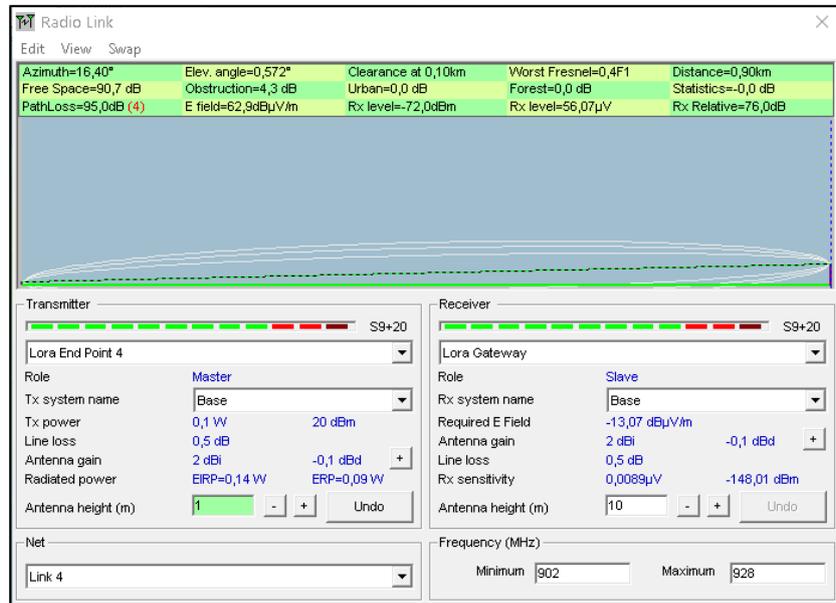


FIGURA 3.11 – Simulación de Link 4

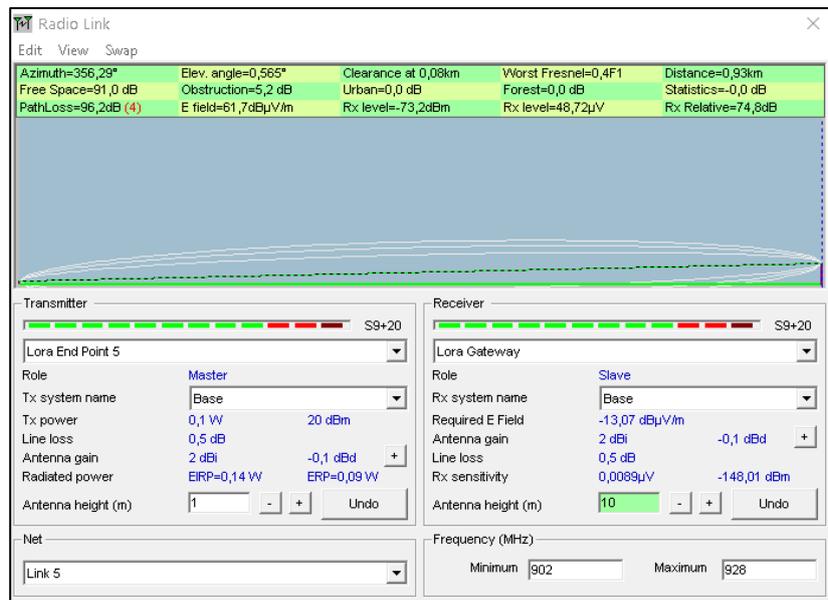


FIGURA 3.12 – Simulación de Link 5

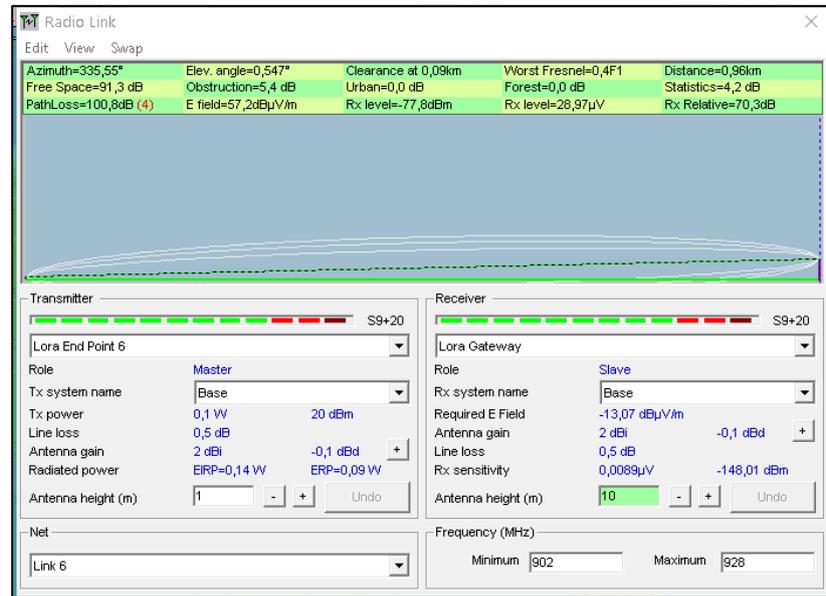


FIGURA 3.13 – Simulación de Link 6

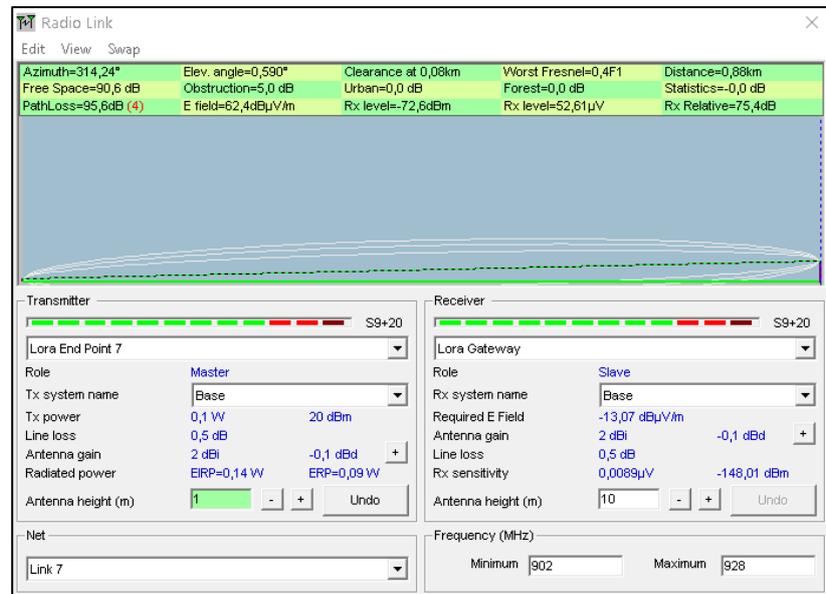


FIGURA 3.14 – Simulación de Link 7

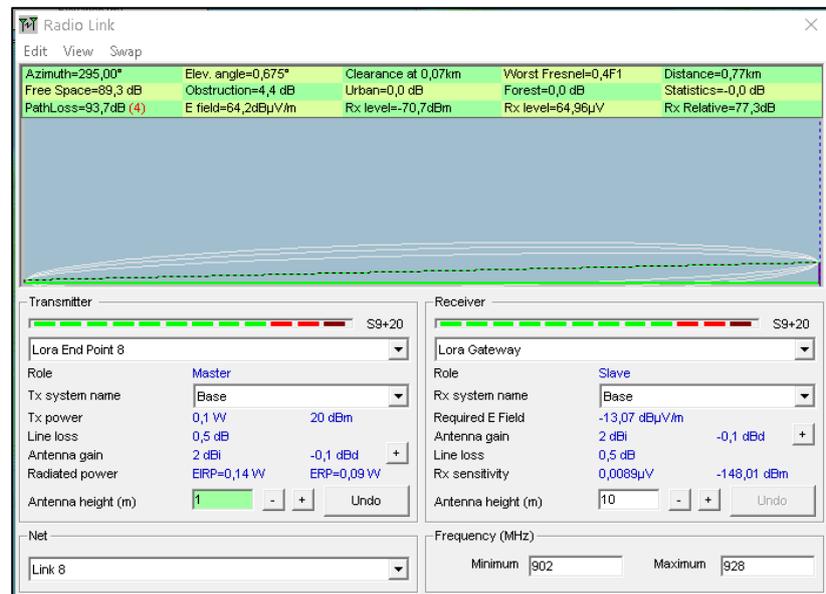


FIGURA 3.15 – Simulación de Link 8

En cuanto a los resultados obtenidos podemos notar que los niveles de recepción mostrados en el simulador están representados por el código S9 + 20, lo cual quiere decir que el margen entre la potencia recibida y la sensibilidad del receptor estaría en valores entre 39dB y 49dB [14], estos son resultados bastante satisfactorios ya que podemos notar que el sistema cuenta con la suficiente robustez para operar en los ambientes de piscinas camaroneras.

3.2.2. Simulación de la solución IoT en Cisco Packet Tracer.

La presente parte de este trabajo detallará las diferentes configuraciones y programaciones que se debieron realizar en el simulador Cisco Packet Tracer y en la plataforma ThingSpeak para poder simular la solución IoT y a su vez lograr exteriorizar los valores obtenidos de los parámetros a través de la plataforma cloud ThingSpeak.

Configuración de Ambientes en Cisco Packet Tracer.

Para la simulación de esta solución IoT en Cisco Packet se debió configurar en Tracer los ambientes o parámetros que deben ser captados por los sensores. Por tratarse de una solución específica para una problemática puntual no todos los ambientes o parámetros necesarios para esta simulación se encuentran disponibles o ya configurados en la plataforma de Cisco Packet Tracer, por ello se procedió a la respectiva creación de nuevos parámetros para la simulación, estos nuevos parámetros fueron creados definiendo sus valores iniciales, sus unidades de medición y a su vez su comportamiento en el transcurrir de un día simulado.

En la Figura 3.16 muestra el detalle de ciertos parámetros creados y configurados con sus valores iniciales y su comportamiento al transcurrir un día simulado.

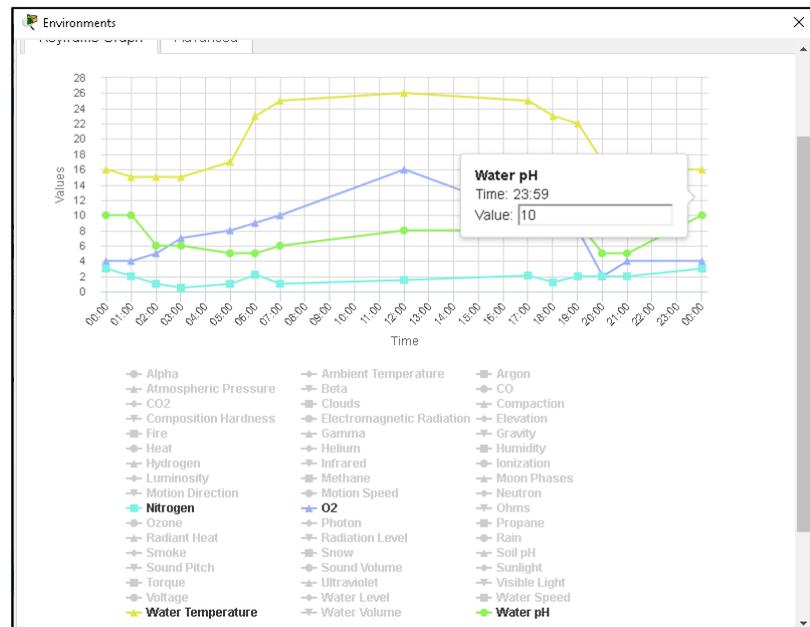


FIGURA 3.16 – CONFIGURACIÓN DE PARÁMETROS

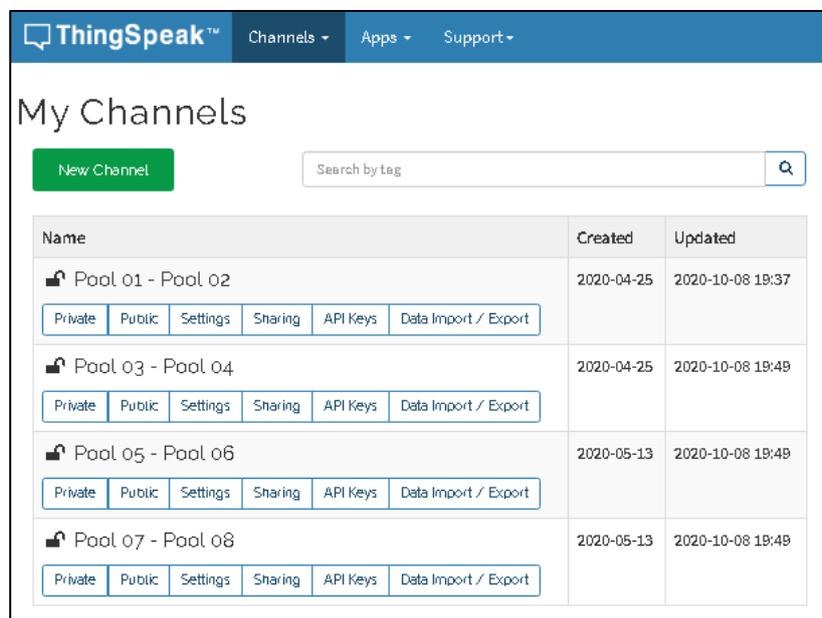
Configuración de canales y campos en la plataforma ThingSpeak.

ThingSpeak es una solución o un servicio de plataforma de análisis de Internet de las cosas creado por la compañía MathWorks, los cuales son los creadores de Simulink y MATLAB. ThingSpeak permite a sus usuarios visualizar, agregar y analizar flujos de datos en tiempo cuasi real en la nube [10]. Esta solución es capaz de proporcionar visualizaciones instantáneas de los datos publicados por los equipos o dispositivos conectados a la plataforma. ThingSpeak es de gran utilidad para el aceleramiento del desarrollo de sistemas del Internet de las cosas, especialmente en aquellos que necesitan de análisis.

Esta herramienta es de uso muy práctico ya que nos permite crear sistemas de IoT sin la necesidad de configurar servidores ni desarrollar software orientado a web.

Para la utilización de esta herramienta se requiere realizar el registro de usuario y contraseña; en este caso se procedió a registrar una cuenta de uso libre o gratuito la cual nos dará acceso a la utilización de cuatro canales los cuales recibirán la información captada por los sensores para ser mostrados mediante la plataforma de visualización de Thing Speak.

La Figura 3.17 a continuación muestra la configuración de los canales de ThingSpeak efectuada para esta simulación.



Name	Created	Updated
Pool 01 - Pool 02 Private Public Settings Sharing API Keys Data Import / Export	2020-04-25	2020-10-08 19:37
Pool 03 - Pool 04 Private Public Settings Sharing API Keys Data Import / Export	2020-04-25	2020-10-08 19:49
Pool 05 - Pool 06 Private Public Settings Sharing API Keys Data Import / Export	2020-05-13	2020-10-08 19:49
Pool 07 - Pool 08 Private Public Settings Sharing API Keys Data Import / Export	2020-05-13	2020-10-08 19:49

FIGURA 3.17 – ESTABLECIMIENTO DE CANALES EN THINGSPEAK

Cada canal nos da acceso a la utilización tipo 8 campos, donde cada campo representa un parámetro a ser captado por algún sensor, Además cada canal en ThingSpeak nos brinda un API KEY el cual es un código alfanumérico de 16 caracteres el cual podemos utilizar introduciéndolo en nuestro código de programación en los MCUs para que la información recibida por los sensores suba hacia los canales ThingSpeak que hayamos configurado. La figura 3.18 a continuación muestra la configuración del canal “Pool 01 - Pool 02” el cual contiene ocho

campos, entre los cuales recibirán la medición de los parámetros de: temperatura, concentración de oxígeno, nivel de pH, concentración de Amonio NH4 en el agua de las piscinas 1 y 2.

The screenshot shows the 'Channel Settings' interface in ThingSpeak. At the top, there are navigation links for 'Channels', 'Apps', and 'Support'. The main content area displays the following information:

- Percentage complete:** 50%
- Channel ID:** 1048556
- Name:** Paal 01 - Paal 02
- Description:** Calidad de Agua de Piscina 1 y Piscina 2
- Fields:** A list of 8 fields, each with a name and a checked checkbox:
 - Field 1: T[°C] - Paal 01
 - Field 2: O2 [mg/L] - Paal 01
 - Field 3: PH - Paal 01
 - Field 4: NH4 [mg/L] - Paal 01
 - Field 5: T[°C] - Paal 02
 - Field 6: O2 [mg/L] - Paal 02
 - Field 7: PH - Paal 02
 - Field 8: NH4 [mg/L] - Paal 02

FIGURA 3.18 – CONFIGURACIÓN DE CAMPOS EN THINGSPEAK

Programación de MCUs.

Una vez creados los canales de ThingSpeak juntamente con sus respectivos campos se procede a armar la red en la simulación la cual está compuesta por un Gateway ubicado en un punto central, el cual mantiene comunicación inalámbrica con los MCU's los cuales están conectados a sensores los cuales captan los parámetros de interés en el agua de las piscinas camaroneras (Concentración de O2, Nivel de pH, Concentración de Amonio NH4, y Temperatura) configurado la,

la Figura 3.19 muestra la topología establecida en la simulación para la red sensorial.

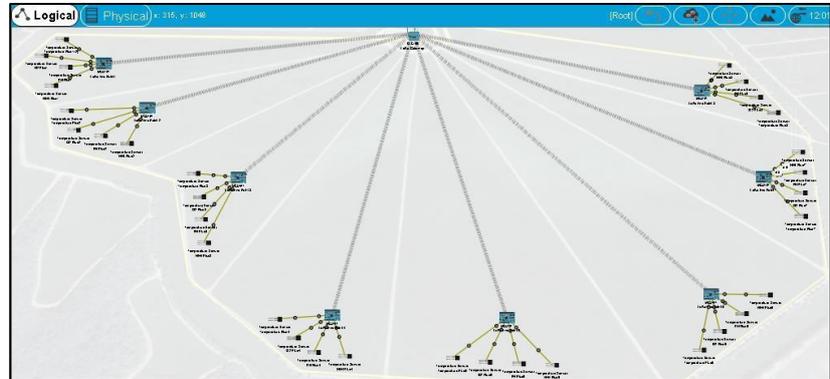


FIGURA 3.19 – Topología de le Red Sensorial

La Futura 3.20 muestra la conexión que tienen los MCU's con los sensores simulados como se detalló anteriormente cada MCU tiene conexión inalámbrica con el GateWay de la simulación.

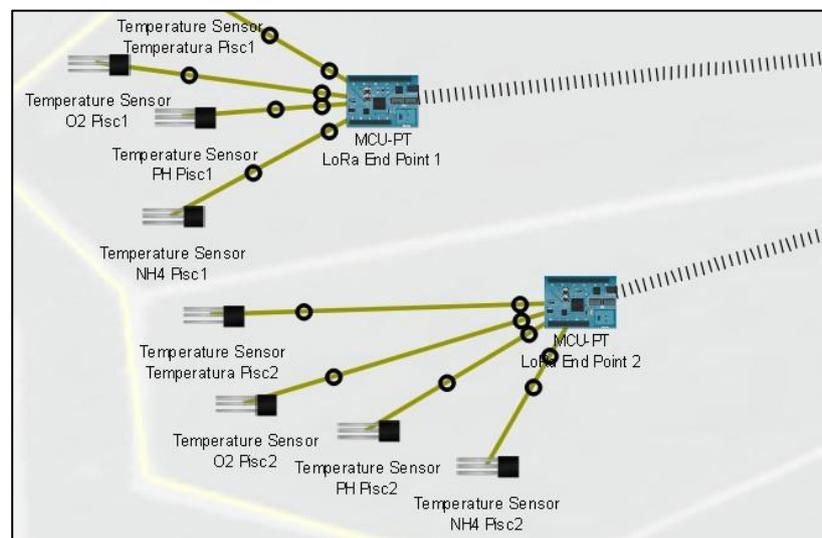


FIGURA 3.20 – Conexión MCU - Sensores

Una vez que realizada la conexión de cada elemento en la simulación se procede a programar los MCU's para que puedan recibir los valores captados por los sensores y que a su vez los mismos puedan ser reenviados hacia la plataforma Cloud Thing Speak, esto es posible ya que contamos con el API Key que ThingSpeak nos proporciona para cada canal.

Las Figuras 3.21 y 3.22 muestran fragmentos de la programación donde los MCUs mediante la cual se logra ingresar el API KEY y a su vez se definen variables de tipo string para almacenar las URL que apuntan a los diferentes campos del canal correspondiente.

Mediante el código de programación de estos dispositivos se efectúan lecturas de las entradas analógicas de los mismos a través de la utilización de la función AnalogRead() para que cada valor tomado pueda ser re-mapeado o preparado para ser enviado hacia la plataforma de ThingSpeak mediante los URL que se han generado para cada campo.

```

1 from realhttp import *
2 from gpio import *
3 from time import *
4
5 #Poner aqui el API key de Thingspeak
6 key = "8UKX2MN3CP360I4J"
7 url_a0 = "https://api.thingspeak.com/update?api_key="+key+"&field1="
8 url_a1 = "https://api.thingspeak.com/update?api_key="+key+"&field2="
9 url_a2 = "https://api.thingspeak.com/update?api_key="+key+"&field3="
10 url_a3 = "https://api.thingspeak.com/update?api_key="+key+"&field4="
11
12 def onHTTPDone(status, data):
13     print("status: " + str(status))
14     print("data: " + data)
15
16 def remap( x, oMin, oMax, nMin, nMax ):
17
18     #range check
19     if oMin == oMax:
20         print "Warning: Zero input range"
21         return None
22
23     if nMin == nMax:
24         print "Warning: Zero output range"

```

FIGURA 3.21 – FRAGMENTO DE PROGRAMACIÓN DE MCUs

```

pinMode(A0, IN)
pinMode(A1, IN)
pinMode(A2, IN)
pinMode(A3, IN)

while True:
    r0 = analogRead(A0)
    ra0 = remap(r0,0,1023,-100,100)

    r1 = analogRead(A1)
    ra1 = remap(r1,0,1023,-100,100)

    r2 = analogRead(A2)
    ra2 = remap(r2,0,1023,-100,100)

    r3 = analogRead(A3)
    ra3 = remap(r2,0,1023,-100,100)

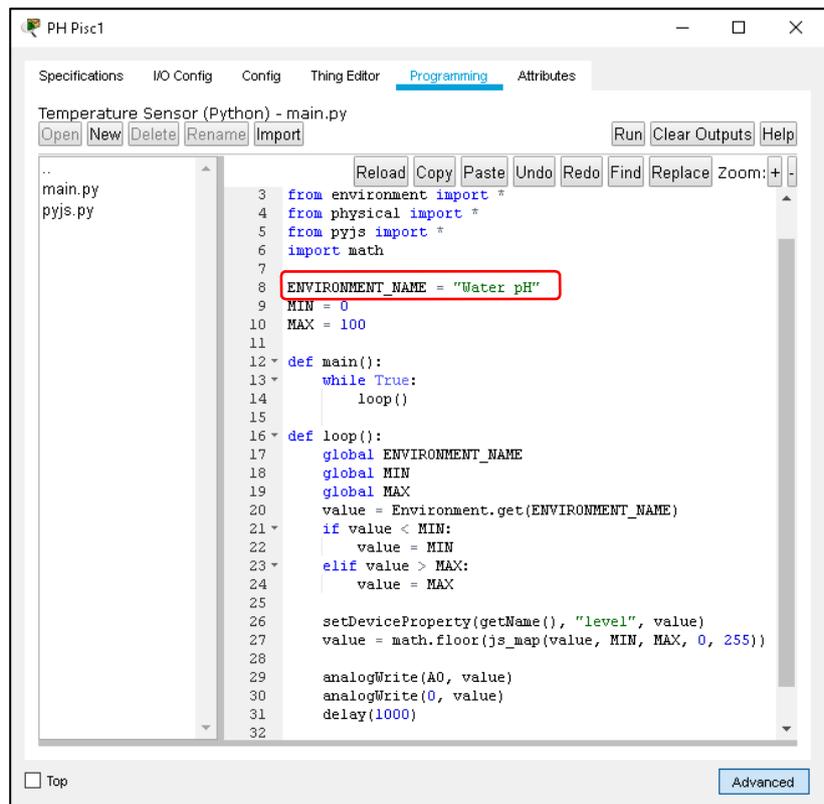
    url = url_a0 + str(ra0)
    print("Enviando: "+url)
    http.get(url)
    sleep(16)

```

FIGURA 3.22 – FRAGMENTO DE PROGRAMACIÓN DE MCUs

Programación de Sensores Especializados.

Debido que en esta simulación en particular los sensores utilizados son sensores especializados, no todos los sensores requeridos se encuentran disponibles en Cisco Packet Tracer, por ello se ha procedido a modificar de programación de sensores existentes para poder lograr la captación de los valores arrojados por los parámetros que no se encontraban disponibles en la plataforma de Cisco Packet Tracer. En la Figura 2.23 se aprecia un fragmento de la programación de un sensor para la medición del nivel de pH en el agua.



```

3  from environment import *
4  from physical import *
5  from pyjs import *
6  import math
7
8  ENVIRONMENT_NAME = "Water pH"
9  MIN = 0
10 MAX = 100
11
12 def main():
13     while True:
14         loop()
15
16 def loop():
17     global ENVIRONMENT_NAME
18     global MIN
19     global MAX
20     value = Environment.get(ENVIRONMENT_NAME)
21     if value < MIN:
22         value = MIN
23     elif value > MAX:
24         value = MAX
25
26     setDeviceProperty(getName(), "level", value)
27     value = math.floor(js_map(value, MIN, MAX, 0, 255))
28
29     analogWrite(A0, value)
30     analogWrite(0, value)
31     delay(1000)
32

```

FIGURA 3.23 – FRAGMENTO DE PROGRAMACIÓN DE SENSOR ESPECIALIZADO.

La programación de cada uno de los sensores empieza realizando una lectura del valor del parámetro correspondiente haciendo uso de la instrucción `Environment.get()`. en la Figura

3.23 se observa cómo se realiza el sensado del parámetro "Water pH" el cual es aquel que engloba todo el comportamiento de este parámetro alrededor del día simulado; se ejecutan instrucciones dentro de un lazo infinito para la toma de los niveles del parámetro, se lo evalúa en rangos máximos y mínimos para luego en la evaluación proceder a escribir dicho valor en la salida analógica la cual estará directamente conectada a la entrada analógica del MCU.

CAPÍTULO 4

4. ANALISIS DE RESULTADOS

4.1. Resultados obtenidos en los cálculos de diseño de la red.

En la presente parte de su trabajo se analizarán cada uno los resultados obtenidos a través de los respectivos cálculos de dimensionamiento y el diseño de la red sensorial IoT bajo la utilización de la tecnología LoRaWAN para poder monitorear los parámetros que representan la calidad del agua en piscinas camaroneras.

En los cálculos efectuados en el capítulo 3 de este trabajo notamos que la sensibilidad mínima que pueden que se puede tener a nivel de recepción en nuestra solución IoT es de -137 dBm, dando esto lugar a que se cuente con un presupuesto enlace máximo de 157dB, todo esto considerando que las transmisiones se dan en los canales LoRaWAN con ancho de banda de 125 KHz, utilizando en nuestro sistema transceivers que operan con el chipset SX1276 cuya figura de ruido es de 6 dB y cuyas frecuencias de operación van desde 137 MHz a 1020 MHz, teniendo en cuenta además que estos dispositivos LoRaWAN pueden manejar límites de relación señal ruido de hasta -20dB.

La Tabla 10 presenta un resumen de los resultados obtenidos en cuanto al cálculo de pérdidas por espacio libre, las pérdidas por la aplicación del modelo de trayectoria de 3 rayos y también el cálculo el radio máximo de la zona de Fresnel para cada uno de los enlaces. Luego la presentación de la tabla haremos un contraste entre los resultados obtenidos en cuanto a las pérdidas y el presupuesto de enlace máximo calculado para el diseño de esta solución.

Resumen de Resultados						
Nombre del Link	Distancia (Km)	L_FS (dB)	L 3 Ray (dB)	Pérdida Total (dB)	Radio de Zona de Fresnel (m)	Radio de Zona de Fresnel (m) con 100% OT
Link 1	0,865	90,42	19,79	110,21	8,42	6,52
Link 2	0,756	89,25	14,72	103,97	7,87	6,10
Link 3	0,684	88,38	33,61	121,99	7,48	5,80
Link 4	0,838	90,14	18,45	108,60	8,28	6,42
Link 5	0,874	90,51	22,28	112,79	8,46	6,55
Link 6	0,925	91,00	22,85	113,85	8,70	6,74
Link 7	0,742	89,09	16,45	105,53	7,80	6,04
Link 8	0,752	89,20	30,38	119,58	7,85	6,08

Tabla 10: Resumen de resultados del capítulo 3

Por inspección simple podemos notar que la pérdida más grande de todas las obtenidas (121,99dB) es muy inferior al presupuesto enlace máximo calculado anteriormente (157dB) esto nos da la garantía de que la señal contaría con la suficiente intensidad para poder llegar desde los LoRa End Points hacia el LoRa Gateway.

En la tabla 11 podemos observar a detalle las diferencias que hay entre el presupuesto máximo de enlace y las pérdidas totales calculadas.

Resumen de Resultados			
Nombre del Link	Link Budget Max(dB)	Pérdida Total (dB)	Diferencia (dB)
Link 1	157	110,21	46,79
Link 2	157	103,97	53,03
Link 3	157	121,99	35,01
Link 4	157	108,60	48,40
Link 5	157	112,79	44,21
Link 6	157	113,85	43,15
Link 7	157	105,53	51,47
Link 8	157	119,58	37,42

Tabla 11: Diferencia entre Link Budget y Perdidas

Una característica importante de las redes LoRaWAN es que operan bajo esquemas los cuales buscan la optimización del consumo de energía, en este caso cómo podemos observar existe un excedente de energía ya que hemos considerando el límite de la relación señal ruido lo cual implica la utilización de un SF de 12, los esquemas de operación de LoRaWAN al notar un escenario como el presente, optan por adaptarse al mismo, reduciendo el valor del Spreading Factor, teniendo como consecuencia que la tasa de datos incremente así como también el tiempo de símbolo el tiempo en el aire disminuye, generando un menor consumo de energía para la transmisión.

A continuación, analizaremos el enlace de más larga distancia el cual es aquel que está conformado por el End Point ubicado en la piscina 6 y el LoRa Gateway, para ello nos hemos apoyado en el uso del simulador Radio Mobile tal como se puede observar en la figura 4.1

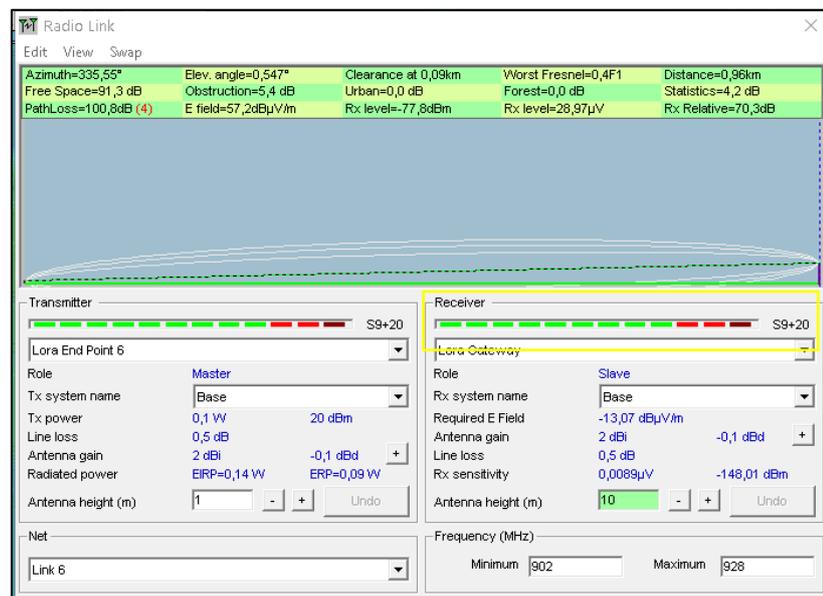


FIGURA 4.1 – Simulación del Enlace 6

En cuanto a los resultados obtenidos podemos notar que los niveles de recepción mostrados en el simulador están representados por el código S9 + 20, lo cual quiere decir que el margen entre la potencia recibida y la sensibilidad del receptor estaría en valores entre 39dB y 49dB [14],

estos son resultados bastante satisfactorios ya que podemos notar que el sistema cuenta con la suficiente robustez para operar en los ambientes de piscinas camaroneras, confirmando además que la elección de las alturas tanto de los LoRa End Points(1) como la del Lora Gateway (10m) son las adecuadas.

4.2. Resultados mostrados en plataforma ThingSpeak a través de la simulación en Packet Tracer.

Una vez ejecutada la simulación se ha procedido a verificar que en efecto los valores o la información llegue de manera apropiada a la plataforma de Thing Speak para su continuo monitoreo. Como se esperaba y según lo indican las políticas de Thing Speak para las licencias gratuitas de uso de la plataforma, se ha podido enviar información hacia esta plataforma Cloud cada 15 segundos alternándose entre cada uno de los parámetros, para este tipo de aplicaciones el hecho de que la transmisión de datos sea cada 15 segundos no es una gran limitante.

La Figura 4.2 muestra el comportamiento de los parámetros a través del tiempo en cuanto a la temperatura, concentración de oxígeno, nivel de pH, y concentración de Amonio NH4 en el agua de la piscina 1.



FIGURA 4.2 – RESULTADOS EN THINGSPEAK PARA PISCINA 1.

La figura 4.3 muestra el comportamiento de los parámetros a través del tiempo en cuanto a la temperatura, concentración de oxígeno, nivel de pH, y concentración de Amonio NH_4 en el agua de la piscina 2.



FIGURA 4.3 – RESULTADOS EN THINGSPEAK PARA PISCINA 2.

La figura 4.4 muestra el comportamiento de los parámetros a través del tiempo en cuanto a la temperatura, concentración de oxígeno, nivel de pH, y concentración de Amonio NH_4 en el agua de la piscina 3.



FIGURA 4.4 – RESULTADOS EN THINGSPEAK PARA PISCINA 3.

Comparativa entre los valores sensados y los valores recibidos en ThingSpeak durante la ejecución de la simulación.

En la presente parte de este documento se procede a realizar una comparación entre los valores visualizados en la plataforma ThingSpeak contra los valores leídos por los sensores los cuales han llegado a los MCUs. La primera comparativa será entre la figura 4.2 y la figura 4.5 en las cuales podemos observar que efectivamente en ambas figuras en sus últimas mediciones los niveles de concentración de oxígeno en el agua de la piscina 1 están en el rango de los 8 miligramos por litro, así como el nivel de pH está en alrededor de 5, como la concentración de amonio NH4 está en 0.43 miligramos por litro, tanto así como la temperatura en el agua en la piscina 1 está por alrededor de los 27° centígrados en el ambiente simulado.

```

LoRa End Point 1
Specifications Physical Config Programming Attributes
Thingspeak (Python) - main.py
Open New Delete Rename Import Stop Clear Outputs Help
main.py Reload Copy Paste Undo Redo Find Replace Zoom: + -
data: 95
Enviando: https://api.thingspeak.com/update?
api_key=QLEXR71JM9EYUHT2&field4=0.387390224751
status: 200
data: 96
Enviando: https://api.thingspeak.com/update?
api_key=QLEXR71JM9EYUHT2&field1=26.4705296944
status: 200
data: 97
Enviando: https://api.thingspeak.com/update?
api_key=QLEXR71JM9EYUHT2&field2=7.47799790134
status: 200
data: 98
Enviando: https://api.thingspeak.com/update?api_key=QLEXR71JM9EYUHT2&field3=5
status: 200
data: 99
Enviando: https://api.thingspeak.com/update?
api_key=QLEXR71JM9EYUHT2&field4=0.431220717555
status: 200
data: 100

```

FIGURA 4.5 – VALORES CAPTADOS POR SENSORES EN PISCINA1.

La segunda comparativa es entre la figura 4.3 y la figura 4.6 en las cuales podemos observar que en sus últimas mediciones los niveles de concentración de oxígeno en el agua de la piscina 2 están en el

rango de los 12,25 miligramos por litro, así como el nivel de pH está en alrededor de 7, como la concentración de amonio NH4 está en 2,17 miligramos por litro, tanto así como la temperatura en el agua en la piscina 2 está por alrededor de los 28° centígrados en el ambiente simulado.

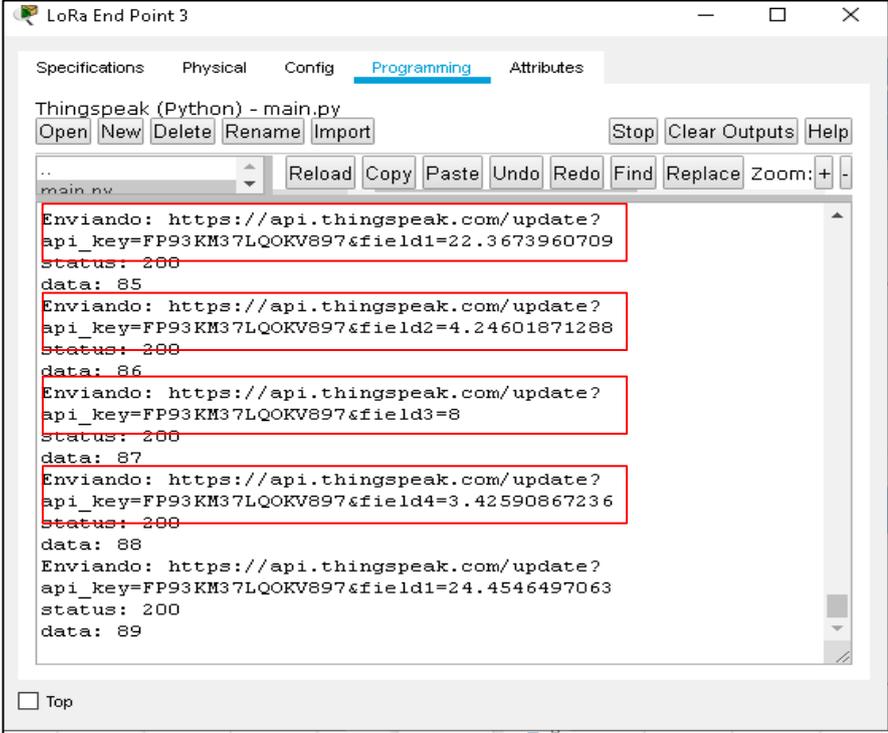
```

LoRa End Point 2
Specifications Physical Config Programming Attributes
Thingspeak (Python) - main.py
Open New Delete Rename Import Stop Clear Outputs Help
main.py 45
api_key=QLEXR71JM9EYUHT2&field8=2.17576650842
status: 200
data: 146
Enviando: https://api.thingspeak.com/update?
api_key=QLEXR71JM9EYUHT2&field5=28.1916094902
status: 200
data: 147
Enviando: https://api.thingspeak.com/update?
api_key=QLEXR71JM9EYUHT2&field6=12.2585324711
status: 200
data: 148
Enviando: https://api.thingspeak.com/update?
api_key=QLEXR71JM9EYUHT2&field7=7
status: 200
data: 149
Enviando: https://api.thingspeak.com/update?
api_key=QLEXR71JM9EYUHT2&field8=2.17749030811
status: 200
data: 150
 Top

```

FIGURA 4.6 – VALORES CAPTADOS POR SENSORES EN PISCINA2.

Proseguimos con la comparativa entre la figura 4.4 y la figura 4.7 en las cuales podemos observar que en sus últimas mediciones los niveles de concentración de oxígeno en el agua de la piscina 3 están en el rango de los 4 miligramos por litro, así como el nivel de pH está en alrededor de 8, como la concentración de amonio NH4 está en 3,4 miligramos por litro, tanto así como la temperatura en el agua en la piscina 3 está por alrededor de los 22° centígrados en el ambiente simulado.



The screenshot shows the 'Programming' tab of the 'LoRa End Point 3' interface. It displays a Python script for sending data to the Thingspeak API. The script consists of five sequential requests, each enclosed in a red box. Each request includes the URL, API key, field number, status, and data value.

```
Enviando: https://api.thingspeak.com/update?  
api_key=FP93KM37LQOKV897&field1=22.3673960709  
status: 200  
data: 85  
Enviando: https://api.thingspeak.com/update?  
api_key=FP93KM37LQOKV897&field2=4.24601871288  
status: 200  
data: 86  
Enviando: https://api.thingspeak.com/update?  
api_key=FP93KM37LQOKV897&field3=8  
status: 200  
data: 87  
Enviando: https://api.thingspeak.com/update?  
api_key=FP93KM37LQOKV897&field4=3.42590867236  
status: 200  
data: 88  
Enviando: https://api.thingspeak.com/update?  
api_key=FP93KM37LQOKV897&field1=24.4546497063  
status: 200  
data: 89
```

FIGURA 4.7 – VALORES CAPTADOS POR SENSORES EN PISCINA3.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- Se logró programar el comportamiento de sensores especializados dentro de la simulación para la captación de parámetros relacionados con la calidad del agua en piscinas camaroneras.
- Se cumplió con configurar la simulación del comportamiento a través del tiempo de los parámetros ambientales que tendrán lugar en las piscinas camaroneras para que sean lo más cercanos a la realidad.
- Se ha logrado exteriorizar la información captada por los sensores en la simulación hacia la solución Cloud Thing Speak con la finalidad de poder monitorear cada parámetro a través del tiempo, siendo dicha información bastante precisa con respecto a lo receptado por los sensores en la simulación.
- Se han obtenido resultados satisfactorios en un estudio de la propagación realizado para esta solución IoT.

Recomendaciones

- En caso de que este diseño se llegase a prototipar se recomienda la utilización de sensores especializados con niveles de protección IP68 ya que estarían sumergidos en las piscinas de forma permanente.
- Se recomienda utilizar el rango de frecuencias de 915 a 928MHz, ya en el Ecuador el ente regulador ha establecido que la banda de 895 a 915MHz sea utilizada para telefonía móvil.

BIBLIOGRAFÍA

- [1] “Revista Acuicultura – Cámara Nacional de Acuicultura,” Cna-ecuador.com, 2011. [Online]. Available: <https://www.cna-ecuador.com/revista-acuicultura/>. [Accessed: 1-Sep-2020].
- [2]. “El camarón se convierte en el primer producto de exportación superando al banano | Pesca Ecuador,” Acuiculturaypesca.gob.ec, 2020. [Online]. Available: <http://acuiculturaypesca.gob.ec/subpesca3703-el-camaron-se-convierte-en-el-primer-producto-de-exportacion-superando-al-banano.html#:~:text=Cifras%20reveladas%20por%20la%20C%3%A1mara,Camposano%2C%20presidente%20de%20la%20CNA>. [Accessed: 14-Ago-2020].
- [3]. “Las camaroneras, motor económico de la ciudad puerto,” El Universo, 15-Jul-2015.[Online].Available:<https://www.eluniverso.com/noticias/2015/07/15/nota/5019508/camaroneras-motor-economico-ciudad-puerto>. [Accessed: 14-Ago-2020].
- [4]. “¿Qué es PaaS? Plataforma como servicio | Microsoft Azure,” Microsoft.com, 2020. [Online]. Available: <https://azure.microsoft.com/es-es/overview/what-is-paas/>. [Accessed: 20-Ago-2020].
- [5]. “Modelos de servicio de cloud IaaS PaaS SaaS,” Ibm.com, 2019. [Online]. Available: <https://www.ibm.com/es-es/cloud/learn/iaas-paas-saas>. [Accessed: 20-Ago-2020].
- [6]. D. Serpanos and M. Wolf, “Internet-of-Things (IoT) Systems, 1st ed”. Cham: Springer International Publishing, 2018, pp. 1-22.
- [7]. C. E Boyd,. “Pond Aquaculture Water Quality Management”. KluwerFisheries Academic Publishers, 2000, pp. 700,
- [8]. “IoT Courses,” Networking Academy, 02-Feb-2018. [Online]. Available: <https://www.netacad.com/es/courses/iot>. [Accessed: 01-Ago-2020].

- [9]. C. E. Boyd, "Consideraciones sobre la calidad del agua y del suelo en cultivos de camarón". Department of Fisheries and Allied, Auburn University, 2000, pp. 5 – 6.
- [10]. "Subcategories," Mathworks.com, 2020. [Online]. Available: <https://la.mathworks.com/help/thingspeak/>. [Accessed: 02-Ago-2020].
- [11]. "Product - Jala," Jala, 11-Sep-2020. [Online]. Available: <https://jala.tech/product/>. [Accessed: 14-Sep-2020].
- [12]. Prensa CambioDigital OnLine, "IoT: Qué necesitan saber los profesionales de la red | CambioDigital OnLine," Cambiodigital-ol.com, 12-Dec-2018. [Online]. Available: <https://cambiodigital-ol.com/2018/12/iot-que-necesitan-saber-los-profesionales-de-la-red/>. [Accessed: 1-Sep-2020].
- [13]. ["Revista Gerencia - En 2022 habrá el doble de proveedores PaaS," Wwww.emb.cl, 2020. [Online]. Available: <http://www.emb.cl/gerencia/noticia.mvc?nid=20190522p2&ni=>. [Accessed: 14-Sep-2020].
- [14] "EA4FSI-28T1 :: Manual de Radio Mobile - Obtención del mapa de cobertura," Ipellejero.es, 2020. [Online]. Available: http://www.ipellejero.es/radiomobile/RM_08.php. [Accessed: 03-Oct-2020].
- [15] Hackster.io, "Hackster.io," Hackster.io, 2020. [Online]. Available: <https://www.hackster.io/adafruit/products/adafruit-waterproof-ds18b20-digital-temperature-sensor/specs>. [Accessed: 03-Oct-2020].
- [16] Hackster.io, "Hackster.io," Hackster.io, 2020. [Online]. Available: <https://www.hackster.io/dfrobot/products/gravity-analog-dissolved-oxygen-sensor-meter-kit-for-arduino/specs>. [Accessed: 04-Oct-2020].
- [17] "Consumer Grade pH Probe," Atlas Scientific, 2020. [Online]. Available: <https://atlas-scientific.com/probes/consumer-grade-ph-probe/>. [Accessed: 01-Oct-2020].
- [18] "Ammonium Ion (NH₄⁺) Sensor Probe [PRO] - Sensors - Waspote - Shop," Cooking-hacks.com, 2020. [Online]. Available: <https://www.cooking->

- hacks.com/ammonium-ion-nh4-sensor-probe-pro.html. [Accessed: 24-Sep 2020].
- [19] "Lora Shield - Wiki for Dragino Project," Dragino.com, 2017. [Online]. Available: https://wiki.dragino.com/index.php?title=Lora_Shield#What_is_the_Dragino_LoRa_Shield. [Accessed: 22-Sep-2020].
- [20] "Frequency Plans," The Things Network, 19-Aug-2020. [Online]. Available: <https://www.thethingsnetwork.org/docs/lorawan/frequency-plans.html>. [Accessed: 01-Oct-2020].
- [21] "What is LoRa? | Semtech LoRa Technology | Semtech," Semtech.com, 2020. [Online]. Available: <https://www.semtech.com/lora/what-is-lora>. [Accessed: 22-Sep-2020].
- [22] "LoRaWAN® Specification v1.0.3 | LoRa Alliance®," Lora-alliance.org, 2020. [Online]. Available: <https://lora-alliance.org/resource-hub/lorawan-specification-v103>. [Accessed: 24-Sept-2020].
- [23] "RP2-1.0.1 LoRaWAN® Regional Parameters | LoRa Alliance®," Lora-alliance.org, 2020. [Online]. Available: <https://lora-alliance.org/resource-hub/rp2-101-lorawanr-regional-parameters-0>. [Accessed: 25-Sept -2020].
- [24] R. Lie, "Mobilefish.com - LoRa/LoRaWAN tutorial.," Mobilefish.com, 2019. [Online]. Available: https://www.mobilefish.com/developer/lorawan/lorawan_quickguide_tutorial.html. [Accessed: 26-Sep-2020].
- [25] Y. Lee, F. Dong, and Y. Meng, "Near Sea-Surface Mobile Radiowave Propagation: Measurements and Modeling."
- [26] "Arduino Pro," Arduino.cc, 2018. [Online]. Available: <https://www.arduino.cc/pro>. [Accessed: 01-Oct-2020].