



PONTIFICIA
UNIVERSIDAD
CATÓLICA DE
VALPARAÍSO



Gustavo Alejandro Landaeta Baeza

Desarrollo de un Sistema de Riego Autónomo Escalable

Informe Proyecto de Título de Ingeniero Electrónico



**Escuela de Ingeniería Eléctrica
Facultad de Ingeniería**

Valparaíso, 04 de octubre de 2018



Desarrollo de un Sistema de Riego Autónomo Escalable

Gustavo Alejandro Landaeta Baeza

Informe Final para optar al título de Ingeniero Electrónico,
aprobada por la comisión de la
Escuela de Ingeniería Eléctrica de la
Facultad de Ingeniería de la
Pontificia Universidad Católica de Valparaíso
conformada por

Sr. David Alberto Velasco López
Profesor Guía

Sr. Francisco Javier Alonso Villalobos
Segundo Revisor

Sr. Sebastian Fingerhuth Massmann
Secretario Académico

Valparaíso, 04 de Octubre de 2018

Este trabajo está dedicado a mi madre, hermanos y amigos

Resumen

En este presente documento se dará conocer toda la información reunida para el desarrollo del informe final de “Proyecto de Título”, el cual representa la labor realizada en las asignaturas de “Proyecto 1” y “Proyecto 2”. El trabajo realizado tiene como principal objetivo el desarrollo de un sistema de riego autónomo y escalable, utilizando componentes electrónicos de rápido y mediano acceso con los que se pueda crear un sistema que trabaje por si solo en la toma de decisiones para la generación del riego, aportando un sistema de ahorro en los recursos hídricos.

En el primer capítulo, se detallaran los antecedentes generales reunidos para el desarrollo del proyecto, siguiendo con la problemática general que se pretende ayudar a resolver, siguiendo con el estado del arte referido a conceptos técnicos básicos relacionados en materia de riego, también abarcando el estado del arte de los implementos electrónicos principales a ocupar en el desarrollo del proyecto, finalizando con una propuesta clara de cómo se desea resolver el problema planteado siguiendo una pauta de objetivos principales y generales a realizar.

El segundo capítulo, es un resumen del protocolo de comunicación inalámbrico que ocupará el sistema, detallando aspectos técnicos importantes, como las bandas de frecuencias, tasa de transmisión de datos, distancia de alcance de la señal, consumo energético y sus topologías principales de trabajo.

El tercer capítulo, mostrará en detalle el proceso realizado para llegar al sistema de riego autónomo y escalable, partiendo por etapas básicas de prueba del sistema de comunicación entre transceptores, se comentará la programación más importante efectuada para una correcta comunicación entre equipos ya prototipados, finalizando con la etapa de pruebas entre dos y más dispositivos y poniendo a prueba la factibilidad en la generación de escalabilidad del sistema.

En el cuarto y quinto capítulo, se dará a conocer el desarrollo ya probado del sistema finalizando con la evaluación económica correspondiente detallando aspectos principales para la generación de una empresa que se dedique a la venta de este sistema como producto de mercado.

Palabras claves: protocolo de comunicación, XBee, Raspberry Pi, Arduino Uno, Sistema de riego autónomo, programación.

Abstract

The present document provides all the necessary information for the development of “Proyecto de Titulo” final report, which represents the work done in the courses “Proyecto 1” and “Proyecto 2”. The purpose of this work is the development of an autonomous and scalable irrigation system, which by means of electronic components of fast and median access, is able to create a decision-making system about irrigation, thus contributing on a saving system of hydric resources.

In the first chapter is included the general background that resulted in the creation of this project, following with the main problem that is intended to be solved, then with the state of art related to basic technical concepts on the irrigation matter. It's also included the state of art of the main electronic devices to be used in the development of the project, then ending with a clear proposal of how we would like to solve the stated problem following a guideline of main and general goals to pursue.

The second chapter is a summary of the main subject of this project, which is a wireless protocol for the communication of the system, given that is the most relevant issue so the communication between interconnected devices can work with no problems in the environment. In this part I present some fundamental aspects of the protocol.

The third chapter contains on detail the process to achieve the autonomous and scalable irrigation system, commencing with a basic testing of the communication system between transceivers. There will also be a few comments about the most important programming used for proper communication between already prototyped equipment, finishing with the testing between two or more devices and checking the feasibility on the generation of scalability of the system.

Finally, in the fourth and fifth chapter, I present the already proved development of the system, ending with the corresponding economic valuation and detailing the main aspects for the generation of a company dedicated to the sale of this system as a market product.

Key words: communication protocol, XBee, Raspberry Pi, Arduino Uno, Autonomous irrigation system, programming.

Índice general

Introducción.....	1
1 Antecedentes generales.....	4
1.1 Contexto del proyecto.	4
1.2 Problemática	4
1.3 Estado del arte referente al riego	5
1.3.1 Riego tecnificado	5
1.3.2 Desarrollo en el ámbito nacional de sistema de riego autónomos y escalables	6
1.3.3 Desarrollo en el ámbito internacional de sistemas de riego autónomos y escalables .	7
1.4 Estado del arte referente a dispositivos electrónicos a utilizar	9
1.4.1 Raspberry Pi	9
1.4.2 Arduino Uno R3	11
1.4.3 Módulo XBee S2C	11
1.5 Solución propuesta	12
1.6 Objetivo del Proyecto	12
1.6.1 Objetivo general.....	13
1.6.2 Obejtivos específicos.....	13
1.7 Conclusión del capítulo.....	13
2 Comunicación del sistema de nodos.....	14
2.1 Protocolo de comunicación	14
2.2 IEEE 802.15.4	14
2.3 Estándar ZibBee	16
2.3.1 Bandas de Operación	16
2.3.2 Nodos y topologías ZigBee	17
2.4 Conclusión del capítulo.....	18
3 Desarrollo de la solución propuesta	19
3.1 Configuración de módulos XBee	19
3.1.1 Software XCTU y configuración	19
3.1.2 Modo API para redes XBee	21
3.1.3 Pruebas entre módulos	22

3.2 Desarrollo del nodo de sensor transmisor.....	23
3.2.1 Desarrollo prototipo del Nodo de Sensor transmisor	23
3.3 Desarrollo del nodo actuador receptor.....	27
3.3.1 Desarrollo prototipo del nodo actuador receptor	27
3.4 Desarrollo del sistema coordinador de nodos	29
3.4.1 Aspectos técnicos para la implementación del programa administrador	29
3.4.2 Desarrollo de programa nodo coordinador	32
3.4.3 Desarrollo del programa básico	35
3.4.4 Escalabilidad del programa	38
3.5 Conclusión del capítulo.....	41
4 Desarrollo del sistema	42
4.1 Desarrollo del nodo transmisor	42
4.2 Desarrollo del nodo actuador	43
4.3 Procedimientos de Pruebas	44
4.4 Conclusión del capítulo.....	45
5 Evaluación económica	46
5.1 Estudio de mercado	46
5.1.1 Mercado proveedores	46
5.1.2 Mercado consumidores	47
5.1.3 Mercado competidores	47
5.2 Estudio económico	48
5.2.1 Inversión inicial	48
5.2.2 Financiamiento de la inversión Inicial	51
5.2.3 Depreciación de activos fijos.....	52
5.2.4 Costos de operación	53
5.2.5 Flujo de caja y cálculo del VAN	56
5.3 Conclusión del capítulo.....	58
Discusión y conclusiones.....	59
Bibliografía	61

Introducción

En Chile se estima que para el año 2030, entre las latitudes que están ubicadas de Arica a Chiloé ocurrirá un calentamiento promedio de 2 a 4 grados Celsius, con intensificación de aridez de la zona norte y avance paulatino del desierto hacia la zona sur. De este problema nacen distintas interrogantes. ¿Cómo se mitiga este problema?, ¿Qué soluciones se están implementando como sociedad?, ¿Cómo se optimizan los recursos? De acuerdo con estas problemáticas nace la necesidad de crear sistemas eficientes, de bajo costo y al alcance de los que más lo necesiten.

El avance de la tecnología ha permitido ir en pos del desarrollo y ahorro de procesos, tanto en el área industrial como en el sector agrícola, este último es donde países y organizaciones han puesto en marcha planes de mejora haciendo hincapié en el ahorro de agua, derivando a la confección y mejora de sistemas de riego más inteligentes, atribuyéndoles mayor grado de autonomía, mejora en la calidad de los materiales y más precisión a la hora de regar, trayendo un gran impacto en las mejoras de cultivos y ahorro de los recursos hídricos.

El estado de Chile, ha brindado gran parte de capital para subsidiar a pequeños y medianos agricultores gracias a la Ley de fomento de obras menores de riego y drenaje, esto con el fin de obtener mejoras en sus sistemas de riego, debido a las proyecciones que se esperan para el país en los próximos años se estiman aumentos en el desarrollo de sequias que afectaran al país de norte a sur.

Debido a estas problemáticas es que nace el proyecto de desarrollar un Sistema de Riego Autónomo Escalable, con elementos electrónicos de fácil y rápido acceso a conseguir en el mercado tanto nacional como internacional. Entre los factores a considerar en este tipo de proyectos son los factores técnicos y económicos, ya que se debe tomar en cuenta que tanto en el sector agrícola como en el sector privado no siempre se poseen grandes recursos de inversión, es por esto que gracias a la electrónica más actual se pueden recurrir a obtener sistemas más prácticos y reducidos en lo económico para el desarrollo de este tipo de proyectos. Dispositivos como Raspberry Pi y Arduino Uno son elementos de desarrollo simples y prácticos que ayudarán a la generación del sistema, enfocados para realizar todo tipo de proyectos tanto profesionales como personales, generando prototipados de excelente confiabilidad y desempeño al momento de trabajar.

Las características de los sistemas de riego con mayor tecnología requieren información de más de una variable del entorno en donde se esté aplicando, es debido a esto que el sistema de riego a proponer deberá contemplar el mapeo de más de una variable del entorno, siendo los rangos de temperatura y humedad las variables más básicas y necesarias en todo tipo de cultivo o sector con áreas verdes a controlar, para la implementación de esta tecnología es necesario generar nodos de sensores, los cuales a medida que van creciendo van creando elementos de una red. Gracias a las nuevas tecnologías es que se pueden generar redes de sensores inalámbricos las cuales permiten una implementación más efectiva y con menos impacto. Es por esto que para el desarrollo del proyecto se generará una red de sensores en donde un dispositivo coordinador será el encargado de administrar los nodos que recopilen datos en terreno y este los administre de tal manera que pueda generar señales a otros dispositivos indicándoles que es debido generar riego en ese entorno sensorizado.

La placa de desarrollo Raspberry Pi es un elemento clave en este proyecto, ya que como sistema embebido cuenta con casi todas las funcionalidades que cuenta un computador de escritorio, pero físicamente es de tamaño reducido, esto sin tener en cuenta la gran cantidad de periféricos que se asocian a un computador. En comparación con otras tarjetas de desarrollo, Raspberry Pi cumple con las características técnicas, precio y calidad que se requiere para casi cualquier tipo de proyecto, otro aspecto importante es la gran cantidad de información que se ha generado en la red debido al auge que han tenido este tipo de sistemas, en comparación a otras tarjetas de desarrollo Raspberry Pi tiene un gran poder de procesamiento de datos, capaz de realizar tareas de reconocimiento de patrones o análisis de imágenes al igual de su capacidad de guardar información, lo que permite generar pequeñas bases de datos para diversos programas. Raspberry Pi como cualquier otro computador utiliza sistema operativo, por lo que tiene un ambiente de trabajo el en cual pueden trabajar en distintos softwares y compatibilizar con distintos lenguajes de programación.

Arduino Uno es una plataforma de trabajo con un microcontrolador Atmel. Esta placa de circuito impreso es de bastante utilidad para el desarrollo de los nodos del sistema, tanto de la etapa de muestreo de variables, como de la activación del riego. Esta plataforma de prototipos open-source basada en hardware y software libre cumple con bastantes funcionalidades requeridas, lo que la hace en una de las plataformas favoritas al momento de empezar a adentrarse en el mundo de los dispositivos electrónicos.

Hoy en día los sistemas inteligentes se ven en la necesidad de mantener la comunicación entre sus dispositivos coexistentes, es por esto que los protocolos de comunicación resuelven la problemática de trasladar la información de un punto A hacia un punto B sin errores, con lo que se puede dar origen a crear sistemas cerrados en donde se puedan mantener variables registradas de entornos o procesos a controlar.

Uno de los aspectos más importantes a la hora de realizar la comunicación entre dispositivos es saber qué protocolo se utilizará y qué transceptor es el más adecuado. Es por esto que los módulos XBee, en especial el XBee S2C es de los más adecuados para generar un óptimo trabajo en la comunicación entre dispositivos, ya que al trabajar con el protocolo ZigBee, este es capaz de

generar redes inalámbricas con el alcance necesario para la implementación de los nodos relacionados al proyecto. Otro punto fundamental es el rendimiento del ahorro energético dependiendo su configuración, también son muy estables en periodos de trabajo. Estos dispositivos proporcionan un software que permite realizar su respectiva configuración, el cual permite modificar cada dispositivo según las necesidades de la red.

En base a todo lo expuesto en los párrafos anteriores, es que se sustenta el trabajo revelado en este informe, el cual consiste en desarrollar un sistema de riego autónomo y escalable, el cual se dividirá en tres tipos de módulos, el módulo coordinador, elemento principal que será el encargado de gestionar las entradas y salidas de la información todo esto generado con un programa el cual se encargará de enlazar los dispositivos entre sí y generar escalabilidad cuando se requiera la ampliación del sistema. El segundo módulo es el nodo de sensores, el cual es el encargado de muestrear las variables de temperatura y humedad y tomar la decisión correcta de cuando el suelo necesita humedad. Por último, el módulo que será el nodo receptor, el cual recibirá la orden del coordinador para activar el riego.

Parte importante de este trabajo en el área de la electrónica consiste en entender de forma exacta como trabaja la comunicación entre el XBee con el Arduino y el XBee con la Raspberry Pi. Importante también es el desarrollo del programa coordinador el cual mediante el lenguaje de programación Python y sus librerías correspondientes se logre crear un programa el cual pueda generar una plataforma básica de comunicación del sistema de riego y también lograr la integración de dispositivos nuevos para la generación de la escalabilidad del sistema.

1 Antecedentes generales

Hoy en día la necesidad de trabajar en soluciones de procesos de mejora continua ha gravitado en el desarrollo de mejoras en las tecnologías en todo tipo de implementaciones tanto industriales como agrícolas. Es por esto, que hoy gracias a los avances tecnológicos es posible innovar en implementaciones o acople de sistemas convencionales, derivándolos en una mayor autonomía, eficiencia, calidad y medio ambiente. En este primer capítulo, se entrega en detalle el procedimiento necesario para ir respondiendo a las incógnitas que afectan el desarrollo de este proyecto. Exponiendo los términos básicos en los cuales se basarán las ideas de la implementación del sistema para comprender el contexto de esta, para seguir con la problemática que se quiere abarcar en medida a la solución propuesta al desarrollo de proyecto.

1.1 Contexto del proyecto.

Este proyecto tiene como función principal estudiar las nuevas tecnologías que han ido emergiendo en el transcurso de los últimos años y que han ganado un espacio para el desarrollo de proyectos. Es por esto que la realización de este sistema generará el conocimiento adecuado para comprender el funcionamiento de protocolos de comunicación y la implementación de estos en sistemas de riego

1.2 Problemática

El cambio climático es uno de los problemas más graves, quizás el más grave, al que el ser humano enfrenta en la actualidad, es por esta razón que hoy en Chile y varios países del resto del mundo el problema del agua ha sido un tema fundamental en resolver.

En Chile se estima que para el año 2030, entre las latitudes que se ubican de Arica a Chiloé, ocurrirá un calentamiento promedio de 2 a 4 grados Celsius, con intensificación de aridez en la zona norte y avance paulatino del desierto hacia la zona sur, provocando la migración de la población en las zonas afectadas [1].

Es por esto que la agricultura y áreas verdes en algunas zonas de Chile se verían afectadas negativamente de forma gradual.

Chile supera los 12.000 millones de dólares anuales en exportaciones tanto forestales como agropecuarias, pero debido al efecto de las sequías por cada 1% de disminución en la capacidad productiva del territorio se pierden alrededor de 120 millones de dólares del PIB, estas pérdidas recaen intensamente en áreas rurales más pobres. Esto es sinónimo del efecto de pérdida de fertilidad del suelo, lo que genera un impacto en menores volúmenes de cosechas y baja calidad en el producto [2].

Gran parte de estos malos resultados se debe a las tecnologías tradicionales y deficientes ocupadas en los sistemas de riego y distribución del agua, como lo son los sistemas de riego por gravedad e inundación, no obstante, también repercute en un alto costo que hoy genera implementar sistemas con mayor tecnología y grado de autonomía.

1.3 Estado del arte referente al riego

Existen diferentes tipos de tecnologías utilizadas en los sistemas de riego, por tanto, se describirán tipos de sistemas ya utilizados tanto a nivel país como en el extranjero, esto para entender la elección de elementos principales que se ocuparon para generar el desarrollo del proyecto.

1.3.1 Riego tecnificado

El riego tecnificado si bien ha sido utilizado desde la antigüedad, este empieza a tomar mayor importancia ante la creciente necesidad de dar uso eficiente al agua a nivel agrícola, fomentado por las administraciones publicas de los países afectados por sus distintas geografías y/o cambios climáticos.

El riego tecnificado o la tecnificación de riego se refiere al aprovechamiento eficiente de los recursos acuíferos, esto a partir del uso adecuado de la tecnología en beneficio de la agricultura, a diferencia de los riegos convencionales, este tipo de riego se distingue por saber cuándo, cuánto y cómo regar, permitiendo la aplicación de fertilizantes y nutrientes de forma segura.

Actualmente los riegos tecnificados más utilizados alrededor del mundo son el riego por aspersión y el riego por goteo, esto se debe a su versatilidad, ya que pueden ser adaptados a cualquier tipo de suelo y a condiciones diversas. Asimismo, favorece el crecimiento y desarrollo de los cultivos, permitiendo alcanzar una eficiencia de alrededor del 95% en comparación a otros mecanismos de riego.

Ventajas

- Ahorro de agua.
- Posibilidad de regar cualquier tipo de topografía.
- Mejor control de malas hierbas.
- Aumento en la producción y calidad de los cultivos.
- Ahorro de mano de obra (sistemas automatizados o semiautomatizados).

Desventajas

- Costos de adquisición elevado.
- Delicados con agua en alto contenido de sales.
- Requiere capacitación a los usuarios.

1.3.2 Desarrollo en el ámbito nacional de sistema de riego autónomos y escalables

En Chile gracias a diversas políticas públicas se ha ido mejorando de forma acelerada la instauración de sistemas de riego más eficientes, esto debido a la ley N° 18.450 [2]. Ley de Fomento a la Inversión de Obras Menores de Riego y Drenaje, esta ley faculta al Estado de Chile para financiar anualmente un programa de construcción de obras menores de riego y drenaje, el que mediante un sistema de concurso público bonifica el porcentaje del costo de construcción. Desde su aplicación se puede afirmar que alrededor del 50% del aumento de la superficie de riego con sistema tecnificado en el país corresponde a aportes estatales [2]. Es importante destacar que esta ley permite potenciar sustantivamente la actividad de las organizaciones regantes y focalizar recursos en la recuperación de la calidad del riego en aguas contaminadas, el apoyo a la agricultura sustentable y fomentar el uso y generación de energías renovables no convencionales.

En Chile existe una variada gama de empresas que se dedican a implementar sistemas de riego avanzados tanto agrícolas como de áreas verdes, entre las cuales se pueden encontrar:

- Livn: Empresa Chilena de sistema de riego inteligente pensado para jardines públicos y privados, consta de un controlador de riego el cual va conectado a actuadores de bombas de distribución de agua, este controlador también está conectado a una red WIFI el cual por medio de una aplicación móvil es controlado y monitoreado, este controlador permite conectar 8 zonas de riego, pero se puede expandir instalando módulos que permiten llegar hasta 256 zonas de riego, este sistema de riego estima el ahorro de agua de hasta un 50% (Figura 1-1).
- Riego Palo Alto: Empresa dedicada al diseño, instalación y suministro de materiales para sistemas de riego tecnificado por goteo, microaspersión y aspersión, además cuenta con equipo especializado en proyectos de financiamiento mediante la LEY de Riego 18.450, así cada proyecto es seguido desde el comienzo hasta el pago de la bonificación adjudicada. En el área de riego inteligente esta empresa ha desarrollado un programa de riego con el propósito de gestionar un buen uso del agua, el que puede ser controlado y monitoreado a distancia por medio de un dispositivo móvil o una computadora (Figura 1-2), sus características principales son la escalabilidad de sectores a los que se puede adaptar conectando el sistema en paralelo, control del sector, hora y tiempo de riego, dispone de dos modelos agrícola y hogar, sistemas los cuales llevan una alta cantidad de sensores conectados para optimizar el riego.

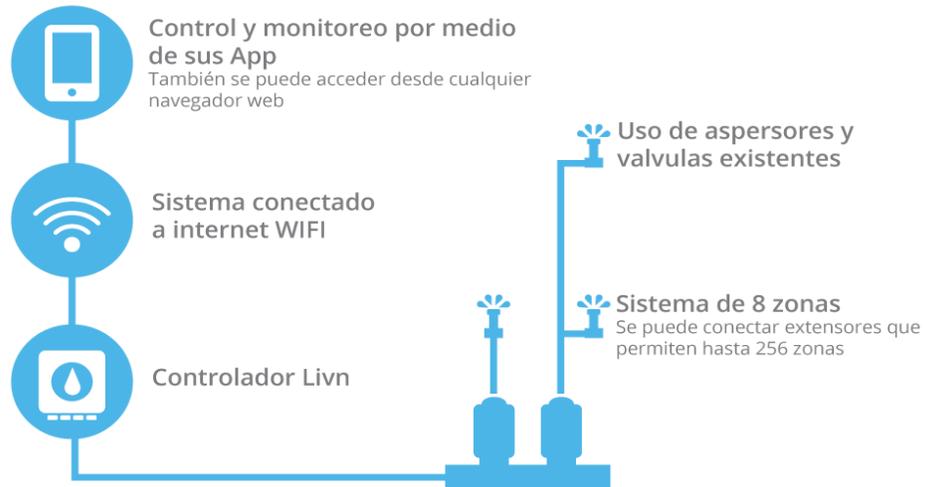


Figura 1-1: Diagrama Livn (fuente: <http://www.livn.garden>)

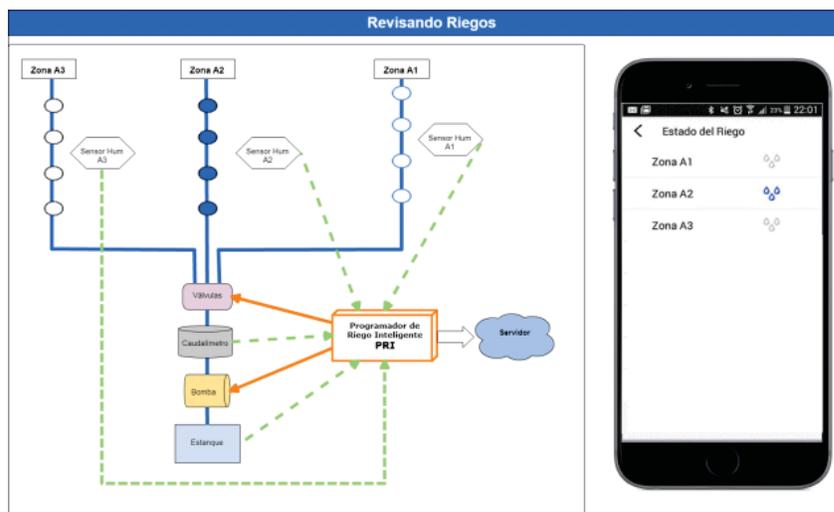


Figura 1-2: Diagrama Riego Palo Alto (fuente: <http://www.riegopalalto.cl>)

1.3.3 Desarrollo en el ámbito internacional de sistemas de riego autónomos y escalables

En el ámbito internacional, el desarrollo de variados sistemas de Riego Autónomo tiene una importante relevancia y son organizaciones las que han impulsado a la implementación de mejoras tecnológicas en este tipo de sistemas, es el caso de la Organización de las Naciones Unidas para la Alimentación [3] y la Agricultura con un gran número de países participantes entre las potencias mundiales como Estados Unidos, Francia, China, Reino Unido, Alemania entre otras.

También existen Ferias Internacionales, donde empresas de todo el mundo relacionadas con el tema van a exponer sus últimas mejoras en cuanto a sistemas de riego tanto inteligentes como autónomos, como por ejemplo la IRRIGATION SHOW [4].

Entre la gran cantidad de empresas y publicaciones relacionadas con sistemas de riego podemos encontrar:

- IDIS COMPANY: Compañía Española dedicada a la fabricación de tecnología de riego avanzado, dirigido a sistemas de riego tanto agrícola como de paisajismo, a través de su sistema INTELLIWATER. El sistema INTELLIWATER actúa sobre el riego de forma automática y en tiempo real, para conseguir condiciones hídricas óptimas que favorezcan el correcto desarrollo de las plantas. Este sistema ocupa tecnología Smart, que consiste en raíces electrónicas que detectan las necesidades de las plantas como lo es la falta de agua, ahorrando hasta un 40% de agua y componentes asociados a esto (abonos, fitosanitarios). El uso eficiente de este sistema se traduce a una mejor distribución del riego, ya que en los sectores donde se monitorea se conoce el estado hídrico de la tierra actuando cuando se requiera.
- Tevatronic Autonomous Irrigation: Empresa dedicada al perfeccionamiento de sistemas de riego tecnificado, su sistema de última tecnología hace que los cultivos sean totalmente autónomos en los aspectos de riego y fertilización, también promueve el aumento de la salud y productividad de los cultivos al mismo tiempo que reduce el uso de agua y fertilizantes, reduciendo de esta manera la contaminación del suelo. El núcleo de Tevatronic está en el controlador de interruptor de válvulas, el cual va conectado al servidor de una nube. Este interruptor recibe datos de sensores colocados en el suelo donde estén las plantas y mediante el algoritmo de su controlador calcula el tiempo que debe regar. Tevatronic ha demostrado su funcionalidad ahorrando hasta un 75% en agua y fertilizantes y aumentando los rendimientos de los cultivos hasta en un 25% [5] (Figura 1-3).

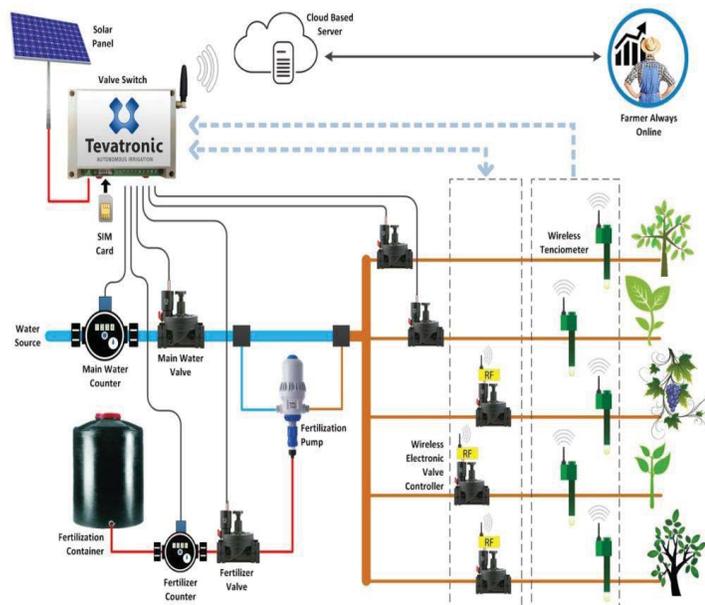


Figura 1-3: Esquema de Riego Autónomo Tevatronic (fuente: <http://www.tevatronic.net>)

- THRESHOLD TENSION AUTONOMOUS IRRIGATION (TTAI) OF OLIVE CONTROLS IRRIGATION EFFICIENCY, WATER USE EFFICIENCY AND FRUIT QUALIT: Esta publicación hecha por el Ph.D Isaac Klein plantea la idea que al colocar controladores de riego en Aceitunas, están pueden regarse autónomamente analizando con un dispositivo que controle el umbral de tensión, en efecto controlando una válvula de riego. Exponer parte del sistema a un conjunto de raíces controlando el umbral de riego mejora la eficiencia del uso del agua y la calidad de la fruta. El enfoque autónomo para el riego precisado en la conclusión de esta publicación resultó prometedor y tiene una ventaja de compilar las interacciones complejas, entre estos parámetros y naturaleza dinámica del consumo del agua en la planta [6].
- Sistema de riego autónomo basado en el Internet de las Cosas, Universidad Internacional de La Rioja, España 2016: Esta tesis de investigación para Master muestra la construcción de un sistema de riego autónomo ocupando elementos de Hardware y Software libre, implementando redes de sensores inalámbricos que permiten obtener la información de las variables agroclimáticas (humedad del suelo, temperatura ambiente, precipitaciones, etc.). Aplicando un sistema de aprendizaje de máquina (Machine Learning), para predicciones de clima [7] (Figura 1-4).



Figura 1-4: Internet de las cosas y la agricultura (fuente: <http://www.smartnet.com.co>)

1.4 Estado del arte referente a dispositivos electrónicos a utilizar

La tecnología a medida que avanza ha ido desarrollando dispositivos electrónicos para distintos fines, esto en base a la necesidad del ser humano de la mejora en procesos tanto productivos como del diario vivir, es por esto que en los últimos años han aparecido distintos tipos de dispositivos electrónicos que al desarrollarlos en conjunto pueden generar origen a distintas soluciones.

1.4.1 Raspberry Pi

Raspberry Pi es un ordenador de placa reducida de bajo costo, desarrollado en Reino Unido por la Fundación Raspberry Pi, con el objetivo de estimular la enseñanza de la ciencia en escuelas [8].

Esta placa soporta varios componentes necesarios en un ordenador común, esto le permite ser utilizado por muchas de las cosas que un pc de escritorio hace, como juegos, procesadores de texto y hojas de cálculo, también tiene la capacidad de reproducir video de alta definición, pero también cumple la versatilidad de poder adaptarle módulos, sensores y otros componentes electrónicos que cumplan con las características de compatibilidad.

En la actualidad esta computadora está siendo utilizada en muchas Universidades e Instituciones de educación superior, para proyectos de pregrado, se tiene como ejemplo de la ESCUELA DE INGENIERIA ELECTRICA PUCV entre uno de sus proyectos de tesis fue el “DISEÑO Y DESARROLLO DE EQUIPO DE APLICACIONES EN TELEMEDICINA”, el cual la Raspberry Pi se ocupó como plataforma para hacer toma de signos vitales, también podemos encontrar la tesis de la Universidad Nacional Autónoma de México “DISEÑO E INTEGRACIÓN DE UN SISTEMAS DE ADQUISICIÓN DE DATOS MEDIANTE EL USO DE ARDUINO Y RASPBERRY PI”, Diseñar e integrar un sistema de adquisición de datos mediante el uso de Raspberry Pi que se pueda controlar y monitorear mediante conexiones a internet y que sea útil al estudio de fenómenos físicos.

En papers, la revista en línea **International Journal of Advanced Research in Computer and Communication Engineering**, el documento que se titula “Machine Monitoring on Cloud using Raspberry Pi and Internet of Things”, el que detalla un nuevo enfoque en la monitorización de máquinas industriales y servidores utilizando Internet de las Cosas (IoT) implementadas en las Raspberry Pi (Figura 1-5), ya que por su estructura y modalidades de configuración, esta puede enviar directamente datos a un servidor de una nube. Cualquier tipo de maquina CNC (Control Numérico Computarizado) o industrial se puede conectar a una Raspberry Pi, recibiendo la información y enviándola a la nube, para así poder monitorear con el servidor de la nube datos de producción en tiempo real, el beneficio de este, es que se pueden monitorear remotamente el funcionamiento y nivel de producción de una línea de proceso o trabajo en particular.



Figura 1-5: Dispositivo Raspberry Pi (fuente: <https://www.prometec.net>)

1.4.2 Arduino Uno R3

ARDUINO UNO R3 (Figura 1-6) es una plataforma de prototipos de electrónica open-source, basada en hardware y software flexibles sencillos de ocupar. Las placas de ARDUINO UNO R3 son programadas a través de un entorno de desarrollo basado en un conjunto de Processing y Wiring.

El hardware consiste en una placa de circuito impreso, el cual es comandado por un microcontrolador ATmega328P-PU, pines digitales y analógicos de entrada/salida, la cual se pueden conectar una infinidad sensores electrónicos y Shields (placas de expansión).

Arduino UNO R3 en comparación a otros tipos de microcontroladores y plataformas disponibles en el mercado ofrece ventajas para profesores, estudiantes y aficionados. Entre las cuales se encuentra su precio accesible comparado con otras plataformas. Es un sistema multiplataforma ya que el software Arduino se ejecuta en sistemas operativos Windows, Macintosh, OSX y GNU/LINUX. Ofrece un entorno de programación simple y claro, también brinda hardware y software abierto y expandible.

Arduino Uno es utilizado en distintas Universidades tanto a nivel nacional como nivel internacional, con el fin de acercar al área de desarrollo electrónico a estudiantes relacionados con áreas de la tecnología misma. También este tipo de dispositivo es ampliamente ocupado por pequeñas y medianas empresas para el desarrollo de prototipos de trabajo, esto debido a la versatilidad que posee a la hora modificar prototipos de desarrollo.

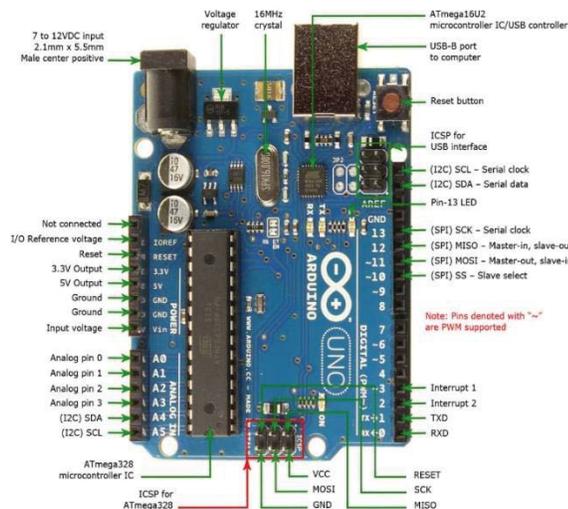


Figura 1-6: ARDUINO UNO R3 (fuente: <http://www.robomart.com>)

1.4.3 Módulo XBee S2C

El XBee S2C (Figura 1-7) es un módulo transceptor de radio frecuencia que utiliza el protocolo ZigBee. Este dispositivo de bajo costo permite realizar conexiones inalámbricas entre dispositivos

electrónicos. Trabaja a una frecuencia de 2.4GHz y permite crear redes punto a punto, punto a multipunto, broadcast y mesh.

Los dispositivos XBee son ocupados en distintos tipos de aplicaciones debido a su fácil instalación, ahorro energético, alto número de dispositivos conectados a una misma red y modificación de la posición de trabajo, los usos más típicos para estos transceptores son: Sensores de movimiento, lectura de instrumentos de servicio (luz, agua, gas, etc.), sistema de riego automático, domótica entre otras.



Figura 1-7: XBee S2C (fuente: <http://www.tinyosshop.com>)

1.5 Solución propuesta

La solución propuesta por el estudiante para la problemática tratada es implementar un sistema de riego autónomo y escalable con dispositivos electrónicos de fácil y rápido acceso con disponibilidad de comunicación inalámbrica en tiempo real, sensar parámetros adecuados para la activación de electroválvulas para el paso de agua, también la creación de un sistema administrador para que puedan los dispositivos comunicarse inalámbricamente. Este sistema se compone básicamente de 3 nodos distintos los cuales cumplen los siguientes roles:

- Nodo coordinador: Dispositivo encargado de administrar todos los nodos conectados a su misma red de comunicación
- Nodo de sensores: Dispositivo encargado en medir variables de temperatura y humedad, configurado con rangos de activación para enviar toma de decisiones a dispositivo coordinador
- Nodo receptor: Dispositivo encargado de la recepción de los datos provenientes del coordinador los cuales generan la orden de activación de electroválvula para el paso correspondiente de agua.

1.6 Objetivo del Proyecto

Los objetivos del proyecto se establecen en dos partes, en primera instancia un objetivo general y un objetivo específico.

1.6.1 Objetivo general

El objetivo general del proyecto es crear un sistema de riego autónomo y escalable, autónomo refiriéndose a la toma de decisiones en las fases de riego y escalable argumentando la disponibilidad de ir creciendo a medida que se solicite cubrir entornos más amplios.

1.6.2 Obejtivos específicos

- Revisar estado del arte sobre riego y dispositivos electrónicos a ocupar.
- Estudiar sobre el protocolo de comunicación para la implementación del sistema.
- Manejar lenguajes de programación para implementación del sistema.
- Diseñar un prototipo coordinador de sensorizado y de recepción para el sistema de riego.
- Análisis de resultados y conclusiones.

1.7 Conclusión del capítulo

La tecnología a raíz de sus grandes avances ha generado herramientas primordiales para la creación de prototipos de distintas clases, dado estas facultades es donde se encuentran dispositivos electrónicos que son sumamente adaptables a la generación de nuevos sistemas compatibles con necesidades de ahorro en recursos naturales. Gracias a leyes implementadas por el estado de Chile, hoy en día es posible generar instancias para la creación e implementación de sistemas basados en el ahorro de recursos naturales, es por esto que el sistema propuesto busca generar una solución en la optimización de los recursos hídricos.

2 Comunicación del sistema de nodos

La comunicación de datos es la particularidad de resolver el problema de llevar información de un punto A hacia un punto B sin la obtención de errores, utilizando redes con la codificación relacionada a su transmisión. En esto es donde se necesitan equipos transductores, que emitan y reciban información con sus respectivos convertidores encargados de la codificación y decodificación de la información que se requiere trasladar entre los diferentes puntos del sistema.

Para poder realizar la comunicación entre Nodos de Sensores, Nodo coordinador y Nodos receptores dada la solución propuesta, es necesario escoger el transceptor adecuado para los requerimientos técnicos a ocupar en el sistema. Es por esto que se ha escogido realizar la comunicación con el transceptor Xbee S2C, el cual ofrece un protocolo de comunicación adecuado para los requerimientos de escalabilidad del Sistema de Riego Autónomo Escalable.

2.1 Protocolo de comunicación

Un protocolo de comunicación se define como las reglas necesarias para la transmisión de información entre dos o más puntos de una misma red. Esta información transmitida puede ser por medio de cualquier tipo de variación correspondiente a una magnitud física. Trata de reglas o estándares que definen sintaxis, semántica y sincronismo de la comunicación. Los protocolos pueden ser implementados por hardware, software o una combinación de estos dos [9].

2.2 IEEE 802.15.4

El estándar IEEE 802.15.4 (Tabla 2-1) [10] es un protocolo de comunicaciones para redes inalámbricas de corto alcance, o redes de área personal (WPAN, Wireless Personal Area Network). Sus características principales son el bajo consumo y costo de los dispositivos, a costa de una baja tasa de transmisión de datos. Este estándar se ajusta a aplicaciones como redes de sensores, donde pueden existir una multitud de nodos que requieren solo un pequeño ancho de banda, pero a la vez necesitan la mayor autonomía energética posible [10].

El estándar define las capas Física y de Acceso al Medio (MAC), capa 1 y 2 en el modelo OSI respectivamente. De esta manera, no es una solución de comunicación completa, sino que requiere de la implementación de capas superiores.

Tabla 2-1: Características cuantitativas del estándar IEEE802.15.4

Segmento	Descripción
Bandas de Frecuencia	868 MHz (Europa, 1 canal) 915 MHz (USA, 10 canales con separación de 2 MHz) 2,4 GHz (Mundial, 16 canales con separación de 5 MHz)
Tasa de Transmisión de Datos	20 Kbps (banda 868 MHz) 40 Kbps (banda 915 MHz) 250 Kbps (banda 2,4 GHz)
Alcance entorno abierto	Especificación inicial original menor a 50 metros Especificación actualizada, más de 1000 metros
Tamaño de las direcciones, Subcapa MAC	64 bits, aunque solo se usan 16 una vez establecida la red

Entre las características de cuantitativas y características generales del estándar se pueden encontrar las siguientes:

- **Bajo consumo:** La filosofía del estándar, que apunta en la implementación de las redes de sensores, permite la fabricación de dispositivos que pueden permanecer en funcionamiento durante periodos de tiempo extensos con solo la necesidad de tener conectada una batería. Esto es gracias a la característica de la utilización de capa MAC de tan solo 16bits (disminuye el número de bits a transmitir), una vez establecida la red. Aunque las direcciones únicas a nivel mundial tienen 64 bits, al establecer la red se asignan direcciones cortas a cada uno de los dispositivos.
- **Reducidos costos de fabricación:** Otra cualidad esencial para las redes de sensores es la necesidad de un bajo costo unitario, puesto que estas redes pueden estar formadas por cientos e incluso miles de dispositivos. El estándar resulta simple de implementar y permite la comunicación con virtualmente ninguna infraestructura, lo que reduce el costo de la electrónica asociada.
- **CSMA-CA (Carrier Sense Multiple Acces – Collision Avoidance):** Implementa una escucha activa del medio para evitar colisiones entre unidades de datos (tramas o frames en este caso, tratándose de la capa 2) y la retransmisión (en momentos aleatorios) si estas colisiones ocurren. Esto es equivalente a lo utilizado en el estándar Ethernet.
- **Seguridad:** La subcapa MAC permite la realización de encriptación simétrica de las tramas, pero las llaves deben ser especificadas por capas de más arriba.
- **RFD (Reduced Function Device):** Este es un tipo de nodo de funcionalidad reducida, pueden comunicarse únicamente con los FFDs, por ende, necesitan muy pocos recursos de trabajo (tiempo y poder de procesamiento).
- **Topología:** Existe la posibilidad de establecer redes punto a punto o en estrella. Ambas requieren que al menos uno de los nodos asuma la función de coordinador, por lo cual únicamente puede ser un FFD. Este nodo coordinador guarda información relacionada con toda la estructura de red. La elección de la topología se deja a la capa de aplicación.

2.3 Estándar ZibBee

ZigBee es un estándar que define un conjunto de protocolos para el armado de redes inalámbricas a corta distancia y baja velocidad de datos [11], basado en el estándar IEEE 802.15.4 de redes inalámbricas de área personal (WPAN). Opera en las bandas de 868 MHz (Europa), 915 MHz (Estados Unidos) y 2.4 GHz (Todo el mundo) y puede transferir hasta 250Kbps.

Este estándar fue desarrollado por la ZigBee Alliance, alianza sin fines de lucro la que participan distintas empresas del rubro tecnológico de las cuales la mayoría son fabricantes de semiconductores. Esta organización nace en el año 2002, la cual desarrolla un protocolo que adopta al estándar IEEE 802.15.4 para sus 2 primeras capas, es decir la capa física (PHY) y la subcapa de acceso al medio (MAC) y agrega la capa de red y de aplicación (Figura 2-1).

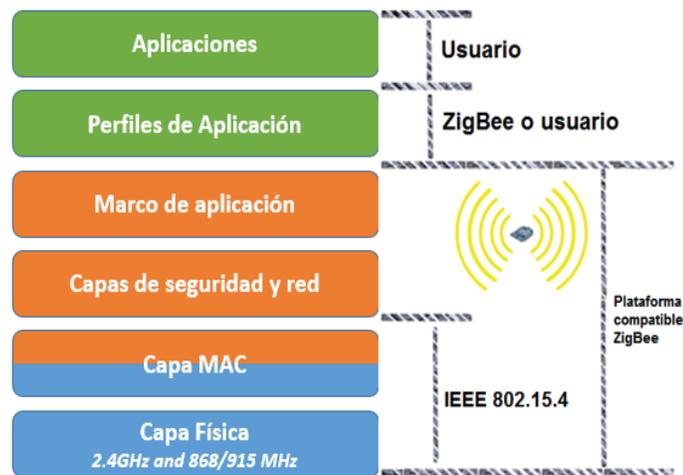


Figura 2-1:Capas ZigBee/802.15.4 (fuente: <https://www.certs.es>)

El estándar Zigbee fue diseñado en un principio pensando en las siguientes especificaciones:

- Bajo consumo para permitir el uso de baterías reducidas.
- Bajo costo de dispositivo.
- Bajo costo de mantenimiento.
- Alcance corto menor a 50 metros.
- Velocidad de transmisión bajo los 250Kbps.

2.3.1 Bandas de Operación

ZigBee opera en las bandas libres de 2.4GHz, 858MHz en Europa y 915MHz para Estados Unidos. En la Figura 2-2, se puede ver el espectro que ocupa en las bandas del protocolo 802 en el cual va incluida ZigBee, el cual abarca los rangos de WPAN (Redes inalámbricas de área personal) y WLAN (Redes inalámbricas de área locas), también notando su baja tasa de datos.

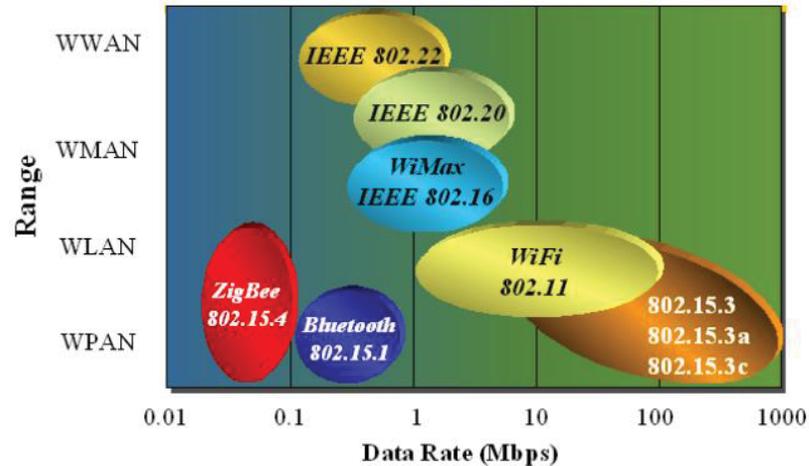


Figura 2-2: Clasificación de redes inalámbricas (fuente: <http://www.ebah.com.br>)

2.3.2 Nodos y topologías ZigBee

Las redes ZigBee, según la disposición de sus nodos y el tipo de implementación más adecuado permite tres tipos de topología de red: en malla, árbol o estrella, dependiendo de estas topologías son como se definen los tres tipos de nodos ocupados en la red ZigBee.

- **Nodo Coordinador:** Realiza funciones de inicio, control y enrutamiento, por lo que debe presentar memoria y capacidad de comunicación. El coordinador es el único en cada red, se sitúa en el centro de una red estrella o en la raíz de una red árbol. Ejerce un papel clave en las funciones de gestión de seguridad de las comunicaciones actuando como centro de confianza (Trust center).
- **Router:** Su función es la de gestionar las rutas de comunicación entre dispositivos. Este nodo es del tipo FFD pero no es coordinador. La utilidad de estos nodos es para extender la cobertura de la red y para aumentar la confiabilidad con la creación de rutas adicionales
- **Dispositivo Final:** Son los nodos que van conectados con sensores y/o actuadores, se caracterizan por ser los últimos nodos conectados a las diferentes topologías, transmiten o reciben datos, entre su característica principal es que pueden permanecer en estado sleep, lo cual optimiza el rendimiento del sistema de alimentación.

Cualquier nodo puede trabajar con uno de estos tipos de configuraciones, solo depende del tipo de configuración realizada y la infraestructura en la topología de red que se quiera implementar. Si bien existen nodos que vienen configurados con un solo tipo de función, la generalidad de estos dispositivos comerciales es que pueden ser modificados en su rol operativo.

ZigBee permite tres tipos de topologías de red:

- **Topología Estrella:** En este tipo de topología se configura un solo nodo coordinador de la red, el cual es el responsable de inicializar todos los dispositivos en la misma, el resto de los elementos conectados al coordinador se comportan como dispositivos esclavos. Cada

mensaje debe pasar por el nodo coordinador el cual enruta la información al correspondiente dispositivo.

- **Topología en árbol:** Esta topología está formada por varias subredes y en la cual el coordinador es el encargado de formar el primer clúster. Este tipo de topología permite que el coordinador pueda ser precedido por nodos routers que, de acuerdo a la ubicación del destinatario en la topología de la red, se enrutarán los paquetes. En esta topología los nodos esclavos no pueden comunicarse entre sí.
- **Topología Malla:** Es una extensión de la topología de comunicación entre pares (peer to peer), el nodo coordinador es el responsable de inicializar la red y elegir los parámetros de la red. Este tipo de topología permite a la red enrutar diferentes caminos la información y así mantener la comunicación entre los nodos.

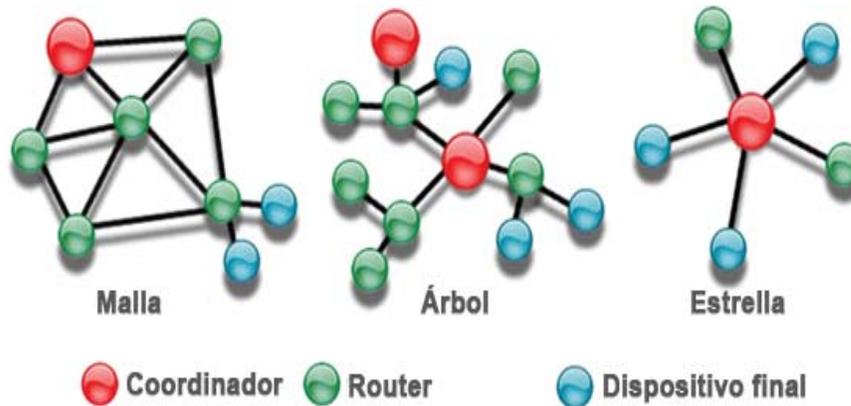


Figura 2-3: Topologías de redes ZigBee (fuente: [http:// www.logicbus.com.mx](http://www.logicbus.com.mx))

2.4 Conclusión del capítulo

El protocolo ZigBee es el estándar de comunicación que cumple con las expectativas de comunicación para la realización del proyecto, ya que brinda optimización en el ahorro energético y cumple con la distancia de alcance en la transferencia de datos. Otro aspecto importante es la configuración de sus topologías de trabajo, ya que estas pueden ser adaptadas dependiendo la necesidad en donde se distribuyan los nodos del sistema.

3 Desarrollo de la solución propuesta

En este capítulo se detallarán todos los aspectos necesarios para las primeras configuraciones de los dispositivos y cómo se programó cada uno de ellos. En esta primera etapa, se apreciará la configuración de los transductores XBee S2C para poder lograr la comunicación y armado de una red, para seguir con el desarrollo de cada dispositivo y detallar los elementos ocupados en cada uno de estos nodos.

3.1 Configuración de módulos XBee

Los módulos XBee que conforman una red de dispositivos interconectados deben estar adecuadamente configurados, ya que dependiendo de la forma en que se quiera trabajar con estos módulos es su tipo de configuración. La configuración de estos módulos se puede realizar de diversas maneras, trabajando en sistemas punto a punto como lo es el modo transparente que es el equivalente a un enlace serie virtual o a partir de tramas API las cuales sirven para trabajos punto a multipunto o redes tipos mesh que permiten enviar o recibir datos a través del envío de paquetes con estructura definida. Estas configuraciones se pueden realizar a través de una computadora, microcontrolador entre otras.

3.1.1 Software XCTU y configuración

La configuración inicial y el reconocimiento de trabajo de los dispositivos XBee de DIGI se puede realizar de manera comprensible con el software XCTU [12], programa distribuido de manera gratuita por él fabricante. En la web se pueden encontrar versiones para Windows, Mac OS y GNU/Linux. El software permite realizar configuraciones con todos los parámetros que se pueden modificar internamente en el dispositivo, entre lo más destacado de estas configuraciones es la de los propios pines (definirlos como entradas analógicas y como entradas y salidas digitales), también permite realizar modificaciones del firmware entre otros. Otro aspecto importante de este software es que permite establecer comunicación tanto en modo transparente como en tramas API, solo conectando los dispositivos al ordenador o conectando un dispositivo al ordenador y otro a un periférico, ya sea mediante Arduino, Raspberry Pi, entre otras plataformas.

Para las primeras pruebas de conexión entre dispositivos, el estudiante se basó en la guía entregada por DIGI [13]. En primera instancia se deben conectar los módulos con sus respectivos transceptores, luego abrir el software y hacer clic en la figura del dispositivo con la lupa, luego se

selecciona el puerto donde este o estén conectados los módulos, tras la selección del puerto se mostrará una ventana con opciones de cambio de parámetros del transceptor (Figura 3-1) los cuales se puede modificar la tasa de baudios, bit de datos, paridad entre otros. El fabricante recomienda no modificar estos parámetros y trabajar con el valor que viene dado por defecto.

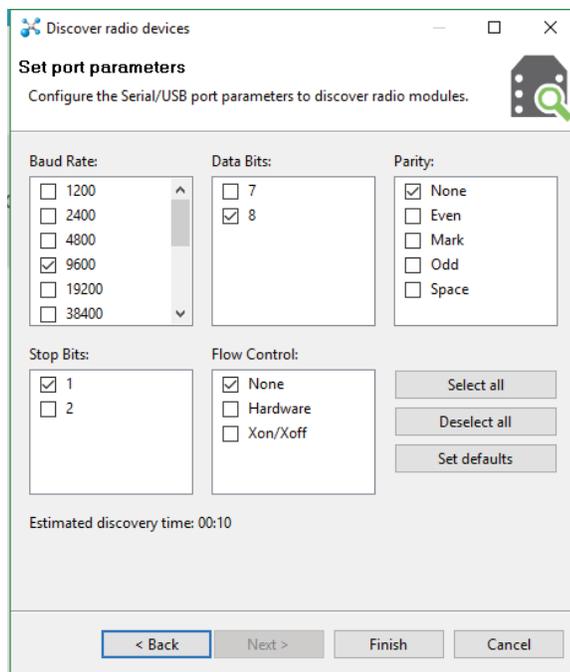


Figura 3-1:Parámetros configurables del transceptor

Luego del reconocimiento de los módulos conectados se procede a la configuración necesaria para cada módulo, para su correcto funcionamiento fue necesario configurar los parámetros que se muestran en la Tabla 3-1, en donde esta configuración fue realizada en modo API.

Tabla 3-1: Parámetros configuración modo API

	Coordinador	Router
PAN ID	2018	2018
CE	Enabled	Disabled
NI	Coordinador	Router
JV	Disabled	Enabled
AP	API enabled	API disabled

- PAN ID: Define la red a la que se conectará. Esto debe ser lo mismo para todos los dispositivos que quieran funcionar en la misma red.
- CE: Establece el dispositivo como coordinador.
- NI: Define el identificador de nodo, nombre cualquiera para módulo.
- JV: Verifica si existe un coordinador en el mismo canal para unirse a la red.

- AP: Habilita el modo API.

3.1.2 Modo API para redes XBee

El modo API proporciona una interfaz estructurada donde los datos se comunican a través de la interfaz serial en paquetes organizados y en un orden determinado. Esto permite establecer comunicación compleja entre dispositivos sin tener que definir su propio protocolo. Los paquetes de datos estructurados en modo API se le denominan frames. Estos frames tanto de entrada como de salida contienen una cierta cantidad de bytes los cuales definen operaciones o eventos dentro del módulo.

El modo API facilita la transmisión de datos a destinos múltiples, ya que para esto solo se requiere cambiar la dirección destino de la recepción en el frame API. Los receptores a su vez indican la dirección de origen, por ende, se tiene un mayor control de direccionamiento.

La estructura de los frames (Figura 3-2) muestra la organización a grandes rasgos en donde el primer byte en formato hexadecimal 0x7E, el cual denota el inicio de la trama, los dos siguientes bytes, la longitud de la trama, desde el cuarto hasta el penúltimo bytes, la estructura específica y el ultimo byte el checksum el cual se encarga de hacer que coincida la estructura de la trama o que si la trama contiene errores en la misma esta sea descartada.

Start delimiter	Length		Frame data								Checksum
1	2	3	4	5	6	7	8	9	...	n	n+1
0x7E	MSB	LSB	API-specific structure								Single byte

Figura 3-2: Estructura de frames (fuente: [13])

Para el proyecto se trabajará en transmisión de datos RF, por lo cual es interesante comprender el comportamiento de la interfaz serial. Primero cuando el módulo envía una solicitud de transmisión por RF, el módulo receptor emite a la interfaz una trama de indicador de recepción. La trama de estado de transmisión no se entrega hasta que el ACK (Mensaje de control a nivel MAC para confirmar entrega de paquetes) es recibido, la cual se muestra en la Figura 3-3. Los bytes marcados en color rojo indican patrones de comportamiento de la trama:

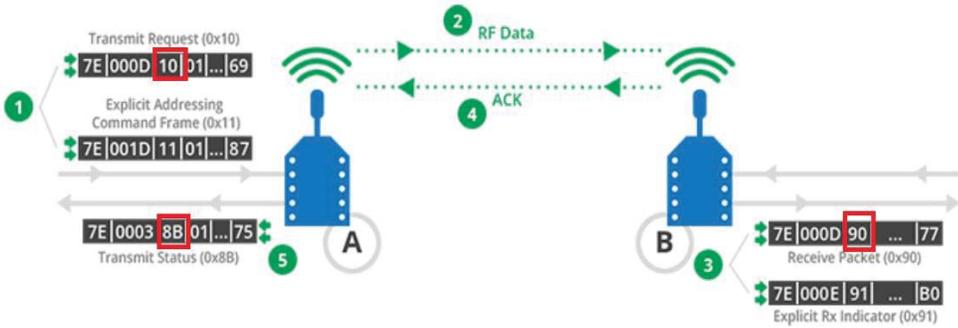


Figura 3-3: Interacción con la interfaz serial, solicitud de Transmisión RF (fuente: [13])

- 0x10 (Transmit Request): Transmite datos inalámbricos al destino especificado.
- 0x8B (Transmit Status): Indica el éxito o fracaso de la transmisión inalámbrica de datos.
- 0x90:(Receive Packet): Indica la recepción del paquete.

3.1.3 Pruebas entre módulos

Para las pruebas entre módulos XBee se necesitaron dos transeptores XBee con sus respectivos módulos conectados a una computadora. Un dispositivo trabajará como coordinador y otro dispositivo como Router, para la primera prueba se comprobará el envío de una trama en modo broadcast y modo directo, para esta primera prueba se debe seleccionar el módulo coordinador, seleccionar la consola y generar un frame. El software XCTU, proporciona un generador de frame (Figura 3-4), lo cual hace amigable el proceso de la generación de este, para un envío broadcast en el generador de frames se debe modificar la zona de 64-bit dest.address =00 00 00 00 00 00 FF FF, lo cual hace referencia que se debe enviar la trama tipo broadcast, en RF data se coloca el mensaje que para este caso será 01, ya que en hexadecimal hace referencia al 30 31 y en seguida se generará el checksum de forma automática.

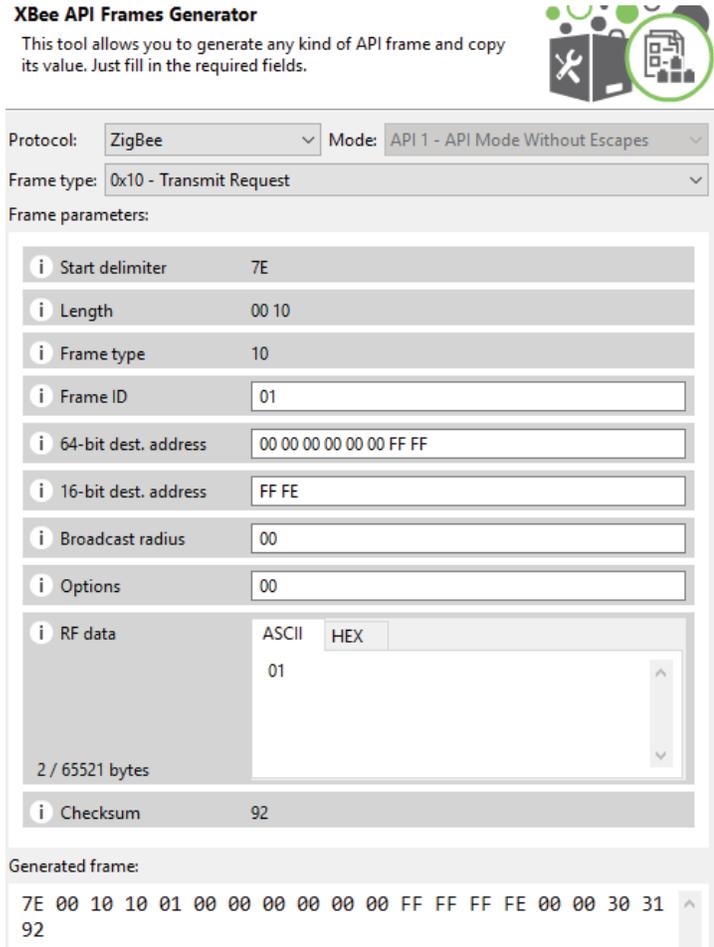


Figura 3-4:Frame generado por XCTU

Para la prueba de envío directo solo se modifica 64-bit dest.address =00 00 00 00 00 00 FF FF, en el cual la dirección debe ser igual a la MAC del dispositivo destinatario, para este caso 64-bit dest.address =00 13 A2 00 41 7B D2 90. Las pruebas con este cambio a la MAC directa arrojaron el mismo resultado que la prueba anterior (Figura3-5), por lo que se concluye que los dispositivos se pueden comunicar sin problemas.

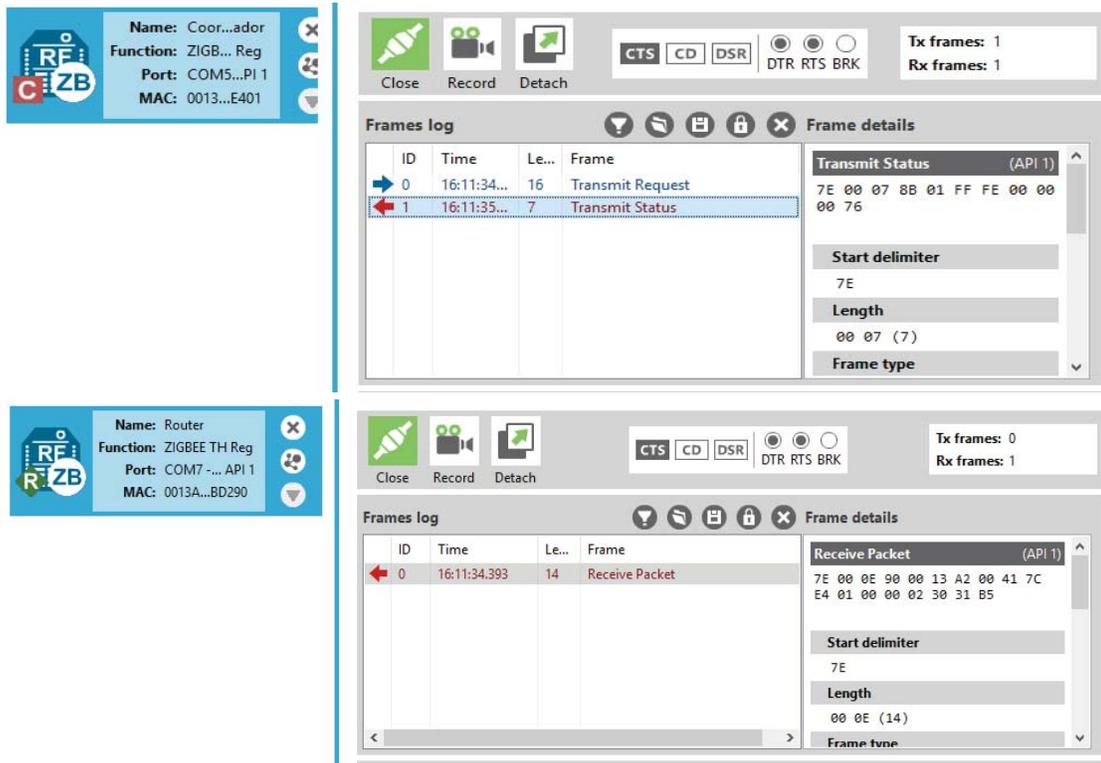


Figura 3-5: Respuestas de frames al enviar y recibir datos con MAC específica

3.2 Desarrollo del nodo de sensor transmisor

El nodo de sensor transmisor será el encargado de muestrear los parámetros necesarios del entorno para el funcionamiento óptimo del sistema. Este nodo sensor estará compuesto por los siguientes dispositivos:

- Arduino Uno R3: Encargado de gobernar los procesos de adquisición de datos de los sensores y enviar señal al módulo XBee.
- Módulo XBee: Encargado de enviar información mapeada.
- Batería litio: Encargado de mantener el equipo energizado.
- Panel Solar: Encargado de alimentar la batería.
- Sensor de Humedad: Encargado de sensar el factor de humedad de la tierra.
- Sensor de Temperatura: Encargado de sensar la temperatura del entorno.

3.2.1 Desarrollo prototipo del Nodo de Sensor transmisor

Para el desarrollo de nodo de sensor transmisor en su prototipo serán ocupados Arduino Uno,

XBee S2C, sensor de temperatura DS18B20 y sensor de humedad FC-20, los que estarán montados esquemáticamente como se aprecia en la Figura 3-6.

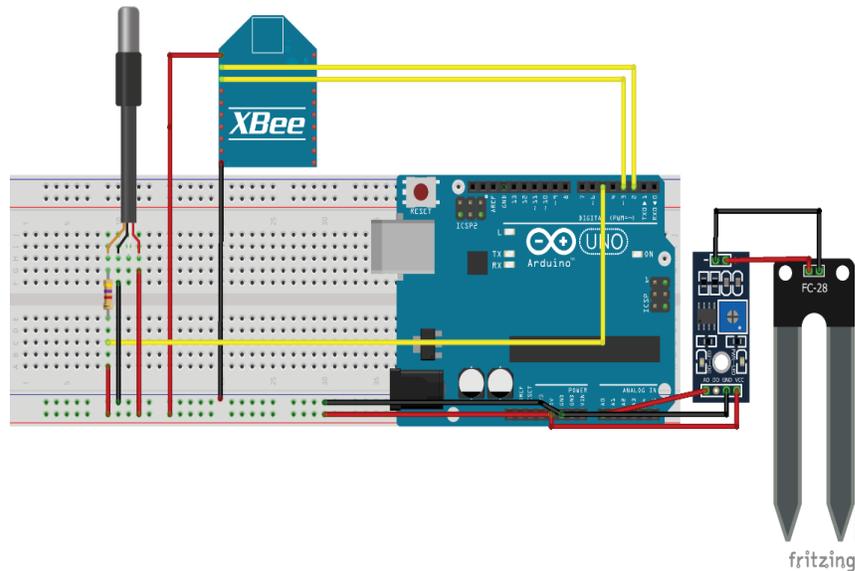


Figura 3-6:Esquema conexión de Nodo de Sensor

Para estas primeras pruebas (Figura3-7) en base a lo estudiado en materia agrícola se utilizarán parámetros de temperatura que debe estar entre los 10°C y los 28°C, los cuales al realizar la medición si se cumplen estos parámetros darán comienzo a la activación del sensor de humedad para realizar mediciones de estado de la tierra. Como este sensor tiene una conversión de 0 a 1023, significará que 0 es húmedo y 1023 seco, por ende, se ajusta a que 700 es un nivel de sequedad, si estas mediciones confirman que la tierra está seca o no tiene la humedad suficiente, se activará la función de enviar una trama al coordinador atribuyendo a que se necesita regar (Figura 3-8). La trama se formará byte por byte y será enviada al coordinador.

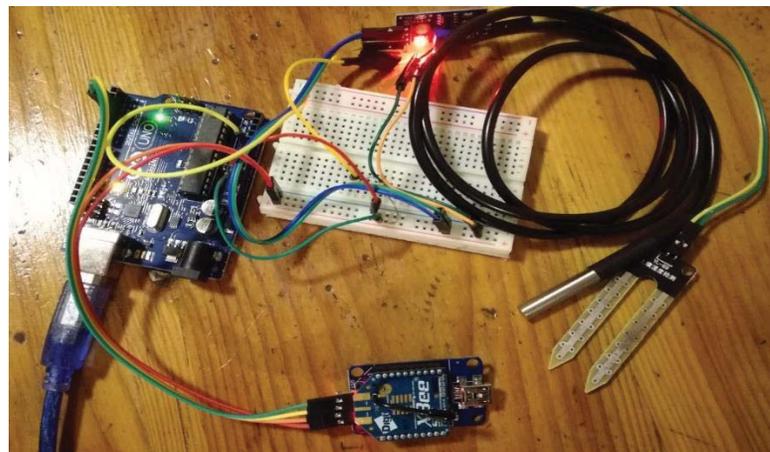


Figura 3-7:Prototipo Nodo de Sensor

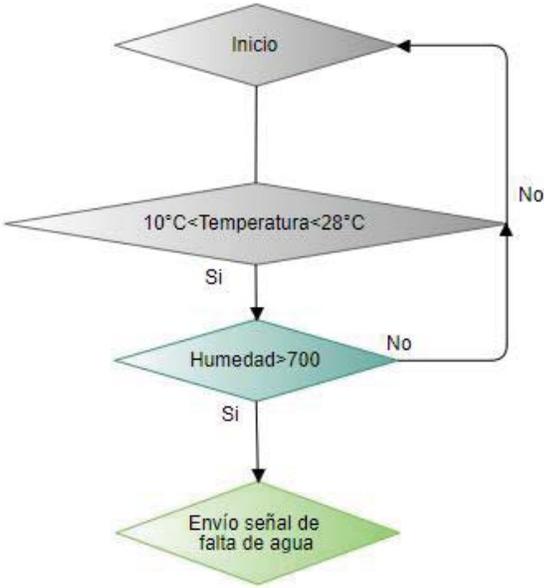


Figura 3-8:Diagrama de flujo condición de riego básica

Para estas primeras pruebas se decide verificar por monitor serial la temperatura y el estado de humedad, el cual cuando por monitor serial avise que se debe regar, deberá llegar un mensaje al coordinador, estas pruebas se realizaron a través de un Arduino y el software XCTU. En la Figura 3-9 se muestra que no es necesario el riego, por ende, no se genera ningún frame para enviar al coordinador, en cambio en la Figura 3-10 se retira el sensor de humedad de una zona húmeda a una seca y el programa genera un frame el cual es enviado al coordinador para informar que necesita ser regado.

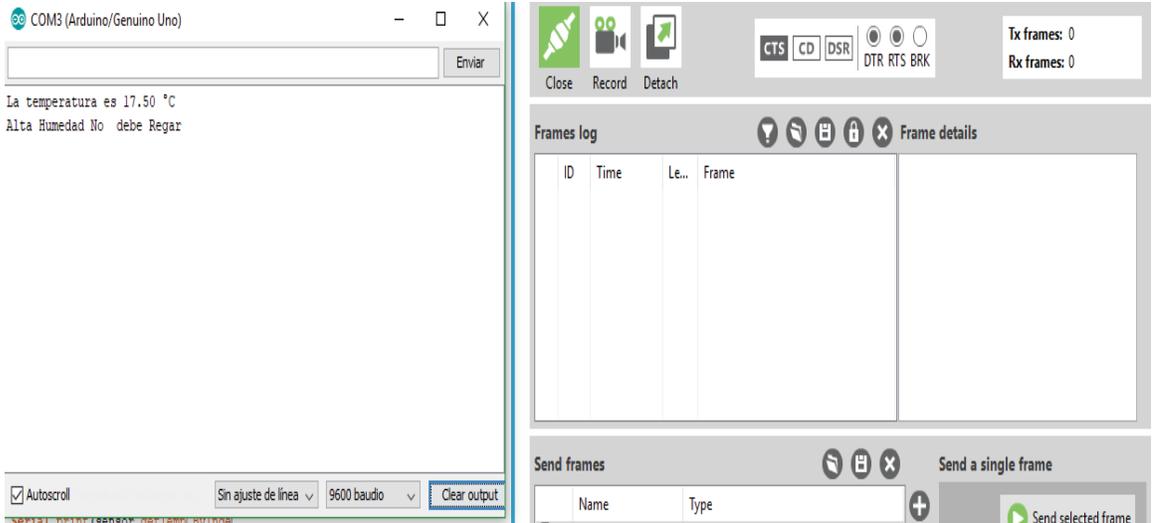


Figura 3-9:Estado que no se necesita riego, sin frame recibido

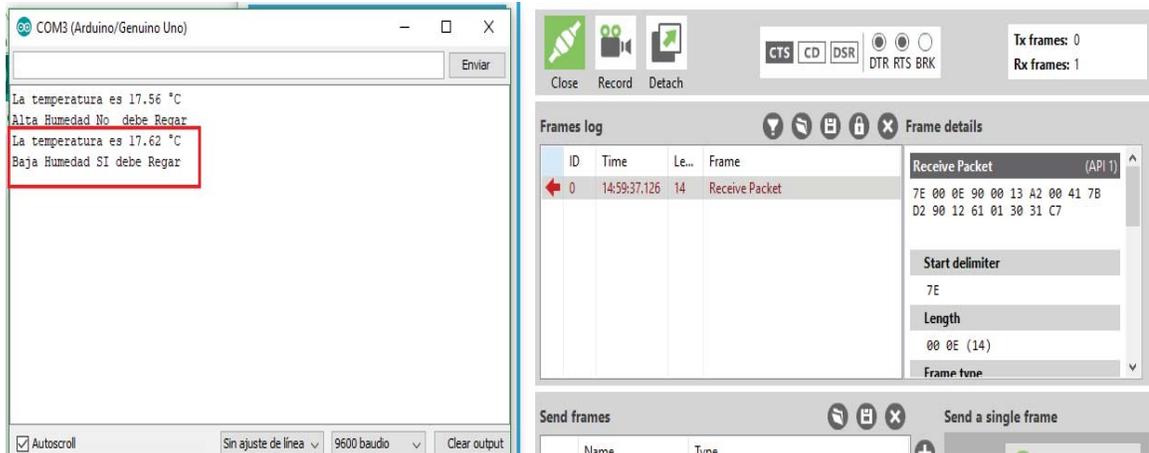


Figura 3-10: Estado en que necesita riego, frame recibido al coordinador

Es importante resaltar que el frame que se generará debe ir dirigido al coordinador, ya que para las siguientes etapas del proyecto este es el que deberá administrar todas las entradas y salidas. Para esta prueba el frame generado en el programa fue el siguiente: 7E 00 10 10 01 00 13 A2 00 41 7C E4 01 FF FE 00 00 31 31 38. Entre los bytes más importantes se pueden ver en primer lugar el 7E que es el byte delimitador la dirección MAC del coordinador y los bytes 31 31 que es el mensaje RF enviado desde el nodo transmisor (Listado 3-1).

Listado 3-1: Función de frame para envío a coordinador

```
XBee.begin(9600);
delay (3000);
XBee.write(0x7E); //byte de inicio
XBee.write((byte)0x0); //Longitud (MSB)
XBee.write(0x10); //longitud (LSB)
XBee.write(0x10); //Tipo de trama
XBee.write(0x01); //Secuencia del mensaje
XBee.write((byte)0x0);
XBee.write(0x13);
XBee.write(0xA2);
XBee.write((byte)0x0);
XBee.write(0x41);
XBee.write(0x7C);
XBee.write(0xE4);
XBee.write(0x01); // dirección destino
XBee.write(0xFF);
XBee.write(0xFE); //Reservado
XBee.write((byte)0x0); // Máximo radio de envío
XBee.write((byte)0x0); //Ajuste de ACK (si es 0x01 se
deshabilita)
XBee.write(0x31);
XBee.write(0x31); // Datos RF
XBee.write(0x38); // Checksum
```

3.3 Desarrollo del nodo actuador receptor

Este nodo será el encargado de recibir la trama de activación de riego enviada por el coordinador y generar la apertura de una electroválvula para el paso de flujo de agua. Este nodo dependiendo de la necesidad del usuario se puede configurar como nodo enrutador o como dispositivo final. Los componentes a utilizar serán los siguientes:

- Arduino Uno: Encargado de activar la electroválvula.
- Módulo XBee: Encargado de recibir orden de activación de electroválvula.
- Electroválvula 12V DC: Encargado de control de flujo de agua.
- Baterías (Para Nodo de control de riego y para electroválvula): Encargadas de alimentar microcontrolador, módulo XBee, relé y electroválvula.
- Panel Solar: Encargado de alimentar batería de microcontrolador con XBee.
- Modulo Relé: Encargado de activar el paso de la tensión para la electroválvula. Este módulo viene con transistor y diodo integrado para no permitir corrientes inversas.

3.3.1 Desarrollo prototipo del nodo actuador receptor

Para el desarrollo prototipo del nodo actuador receptor se ocuparán los siguientes componentes: Arduino Uno, Modulo XBee S2C, válvula solenoide 12VDC normalmente cerrada, un módulo relé 5VDC y una fuente de alimentación o batería de 12VDC, todos estos montados como en el esquema de la Figura 2-6.

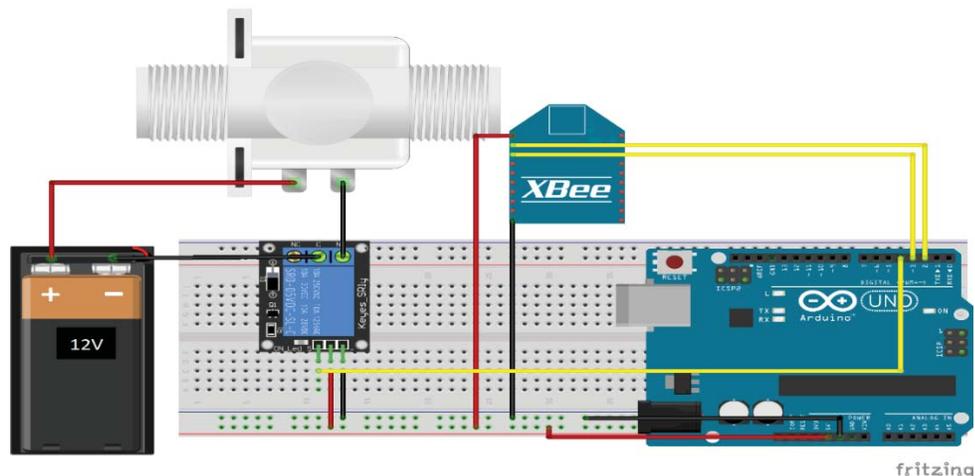


Figura 3-11: Esquema nodo actuador receptor

Para la implementación de este nodo actuador se configuro el módulo XBee en modo Router. En esta etapa del proyecto, se estudiaron diferentes formas enviar y recibir una trama adecuada para el funcionamiento de la válvula solenoide, dando el método más factible es generar un frame (7E 00 0F 10 01 00 13 A2 00 41 7B D2 90 FF FE 00 00 31 ED) con la MAC destino del nodo y un mensaje de activación, para esta activación será enviar un 1=0x31. Primero se tuvo que verificar que el frame que recibía el dispositivo Arduino fuera el mismo recibido en las pruebas con el software XCTU (7E 00 0D 90 00 13 A2 00 41 7C E4 01 00 00 01 31 E6) (Figura 3-12).

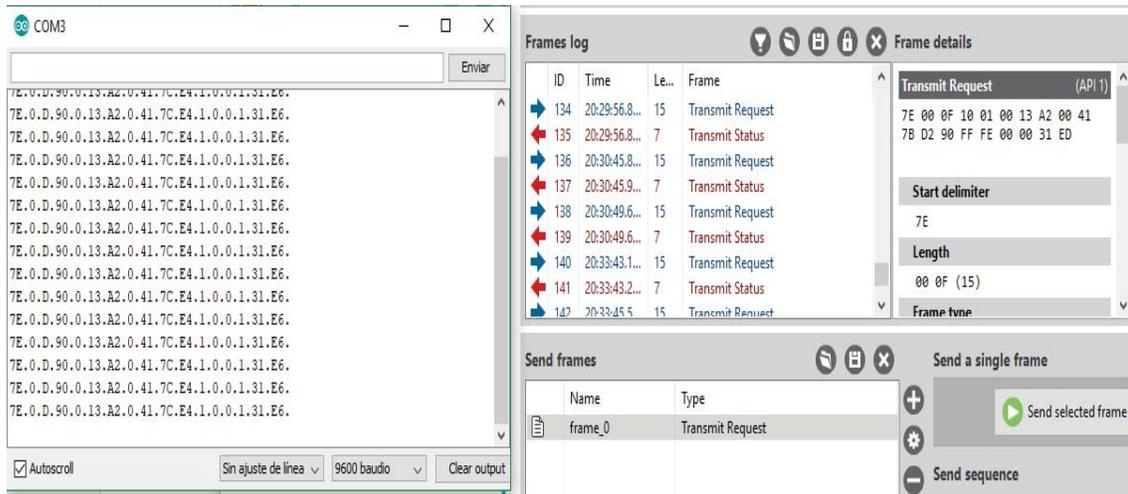


Figura 3-12: Prueba de lectura de frame receptor mediante monitor serial Arduino Uno

En las etapas de prueba se comprobó que el nodo receptor recibe solo mensajes con dirección a su propia MAC o mensajes broadcast, es por esto que la condición del microcontrolador para iniciar la habilitación de la electroválvula será testear la existencia del 1=0x31. Para esta etapa se programó el Arduino Uno (Listado 3-2) de forma que fuera capaz de captar el byte 0x31 y enviar un mensaje por el monitor serial que el riego está habilitado (Figura 3-13).

Listado 3-2: Parte importante de función nodo receptor

```

if(XBee.available(>18){
    if(XBee.read()==(0x7E))// Comienzo a leer después del primer byte
    {
        for(int i=0; i<14;i++){
            byte skip=XBee.read();//omito los bytes hasta llegar al byte
0x31
        }
        valor=XBee.read();
        if(valor ==0x31){ // si lee byte 0x31, iniciar riego
            Serial.print("EL BYTE ES ");
            Serial.print(valor,HEX);
            digitalWrite(Partida,HIGH); //activación de electroválvula
            digitalWrite(11,HIGH);//led de encendido
            Serial.println(" Activo Riego");
        }
        else {// si no leo byte 0x31, no hago nada
            Serial.print("EL BYTE ES ");
            Serial.print(valor,HEX);
            digitalWrite(Partida,LOW);
            Serial.println(" No activo Riego");
        }
    }
}

```

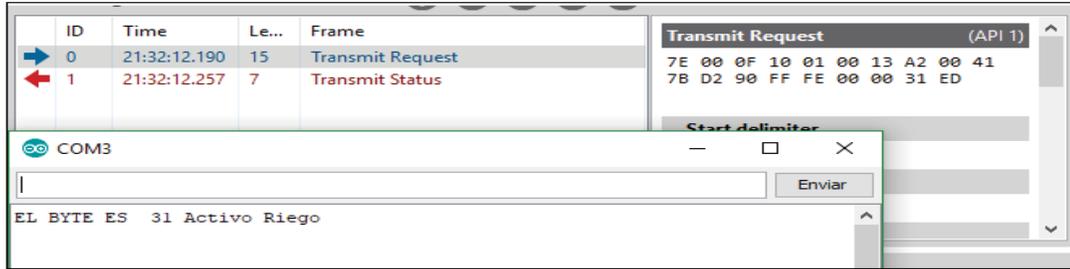


Figura 3-13: Activación de Riego confirmado por monitor serial

También se realizó la prueba de enviar un byte distinto para este caso un 0 =0x30 a la información requerida, para confirmar que no se activara el riego al recibir otro tipo de mensaje (Figura 3-14).

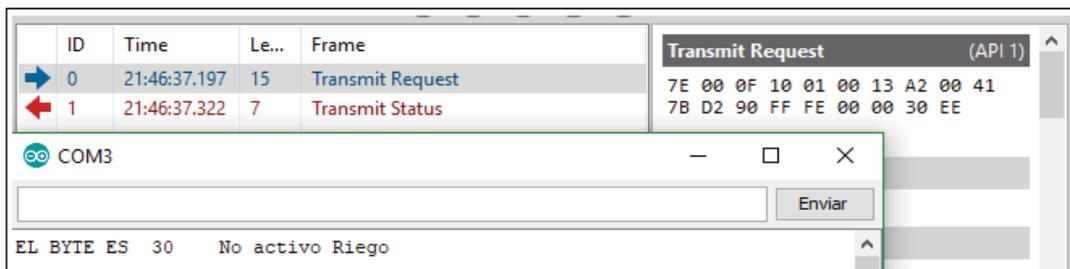


Figura 3-14: Prueba con byte distinto de confirmación de riego

3.4 Desarrollo del sistema coordinador de nodos

El nodo coordinador será el encargado de administrar la red de todo el sistema, en este se recibirá la trama de los nodos transmisores, los cuales irán enlazados con sus respectivos nodos receptores, generando el proceso de riego autónomo. Antes de comenzar con el desarrollo es necesario conocer ciertos aspectos técnicos sobre cómo se implementará la realización del programa administrador del sistema de Riego Autónomo Escalable. El dispositivo coordinador constará con los siguientes elementos:

- Raspberry Pi 3: Encargado de gestionar las decisiones y administrar el sistema completo.
- Transceptor XBee S2C: Encargado de recibir y enviar datos.
- Transformador: Encargado de la alimentación del sistema.
- Panel Solar: Encargado de abastecer la batería con carga.
- Batería Litio: Encargado de mantener el sistema encendido en caso de corte de energía.

3.4.1 Aspectos técnicos para la implementación del programa administrador

Para la administración y coordinación del Sistema de Riego Autónomo, es necesario ocupar un ordenador apto para lograr los requerimientos específicos al caso. Es por esto que el ordenador de placa reducida Raspberry Pi 3 Model B resulta conveniente a la hora de gestionar el proceso más importante del sistema de riego.

Para la generación del programa que administre y coordine los nodos se ocupará el sistema operativo Raspbian, basado en Linux el cual ofrece una variada gama de lenguajes ya preinstalados para ocupar en el desarrollo del proyecto.

3.4.1.1 Python

El programa administrador de nodos se realizará con el lenguaje de programación Python 2.7 (Incluido en Raspbian) el cual es un lenguaje interpretado, Python hace referencia a tener una filosofía de código legible lo que hace amigable al momento de generar la programación (Figura 3-15).



Figura 3-15: Python (fuente: <http://www.blog.aulaformativa.com>)

3.4.1.2 Instrucciones y librerías importantes

Python ofrece una gran cantidad de documentación gratuita [14], donde se puede encontrar todas sus funciones y librerías predeterminadas, no obstante, el estudiante tuvo que requerir a librerías externas para poder trabajar de mejor manera con la conexión de Raspberry Pi y módulo XBee, entre las instrucciones y funciones más importantes se pueden mencionar:

- PySerial: Esta librería externa encapsula el acceso para el puerto serie. Proporciona Backends para Python, disponible para Windows, OSX, Linux entre otros sistemas operativos. Con la instrucción llamada “serial” se selecciona automáticamente el Backends apropiado [15].
- Struct: Esta librería interpreta cadenas como datos binarios empaquetados, esta librería es ocupada para manejar datos binarios almacenados en archivos o conexiones de red, entre otras fuentes.
- Time: Este módulo proporciona varias funciones relacionadas con el tiempo y fechas de calendario de manera fácil y compleja, en ella se puede obtener fechas, hora, zona horaria.
- def: Es la función de la secuencia de sentencias que ejecuta una operación deseada, se pueden escribir los nombres que se deseen para las funciones siempre y cuando no use una palabra reservada en Python.
- serial.readline(): Instrucción para leer datos que están entrando por puerto serial.
- serial.write(): Instrucción para enviar datos desde el programa al puerto serial.

3.4.1.3 VNC

Para poder trabajar de forma cómoda y eficiente se utilizó el programa VNC, el cual es un programa de software libre basado en estructura cliente-servidor que permite observar y realizar acciones del ordenador servidor remotamente a través de un ordenador cliente. VNC no impone restricciones en el sistema operativo a ocupar entre servidor y cliente, lo que hace posible compartir la pantalla de una máquina con cualquier sistema operativo de admita VNC. Para realizar la conexión es necesario habilitar VNC Server en la Raspberry Pi e instalar VNC Viewer en cualquier ordenador que se quiere ocupar. Teniendo los dos programas funcionando se debe ingresar a VNC Server y obtener la dirección IP (Figura 3-16).



Figura 3-16: VNC Server desde Raspberry Pi

Ya obtenida la IP del servidor, se ingresa a VNC Viewer y se coloca la IP proporcionada por el servidor (para este caso 10.7.3.104), al verificar que corresponde la dirección se abre una ventana en donde se puede trabajar de forma remota con la Raspberry Pi (Figura 3-17).

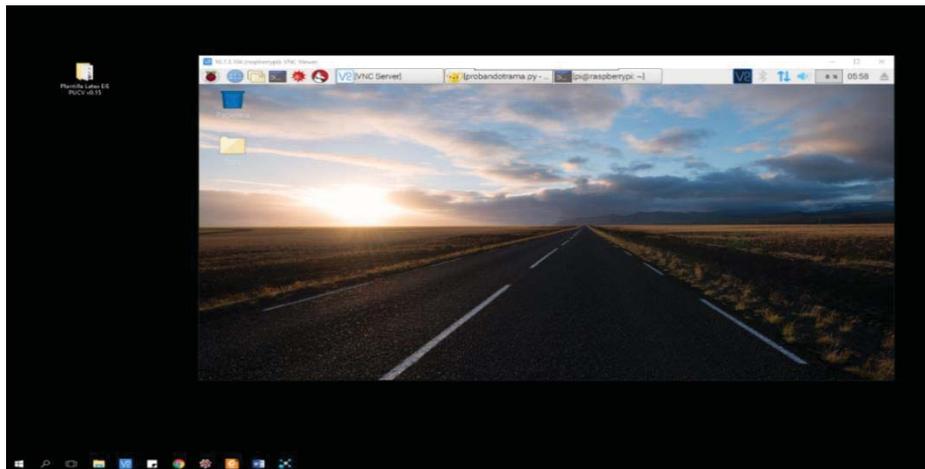


Figura 3-17: Visualización de pantalla remota Raspberry Pi por medio de VNC

3.4.2 Desarrollo de programa nodo coordinador

Con los elementos necesarios a ocupar en el desarrollo del sistema coordinador se fue probando en bloques de funciones programadas cada proceso involucrado en el armado de la red de nodos tanto receptores como transmisores. Los materiales para la conformación de este nodo Coordinador corresponden a una Raspberry Pi, XBee S2C, adaptador módulo XBee mini USB serial y transformador 5[V] 2[A] para la alimentación.

En esta parte se mostrará cada etapa del desarrollo del programa, logrando unir cada bloque de forma adecuada para generar un programa final que conecte todas las funciones necesarias para lograr el proceso requerido. En esta etapa para un trabajo óptimo se comunicó la Raspberry conectada a un XBee con una computadora y un dispositivo XBee más el software XCTU (Figura 3-18).

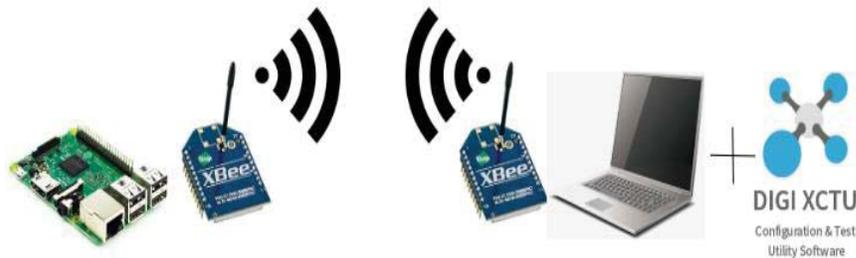


Figura 3-18: Esquema de conexión para pruebas

3.4.2.1 Transmisión de nodo coordinador

Para la etapa de programación se generó una función llamada Tx (Listado 3-3), en el cual se arma una trama ayudada por el software XCTU, esta trama es convertida en hexadecimal para que Python la genere como tal, luego es enviada por el puerto serial al módulo transceptor. Como prueba se generó enviar una trama desde la Raspberry Pi hacia el software XCTU (Figura 3-19).

Listado 3-3: Función transmisión de trama

```
import serial

ser=serial.Serial('/dev/ttyUSB0',9600)#Activo Puerto serial

def Tx():
    fTx1="7E001010010013A200417BD290FFFE00003131BC"
    #Frame a enviar
    Tx1=fTx1.decode("hex")#frame convertido a hexadecimal
    ser.write(Tx1)#Escribo frame por puerto serial
    print "paquete enviado"

Tx()
```

```

pi@raspberrypi: ~
Archivo Editar Pestañas Ayuda
pi@raspberrypi:~ $ python enviar.py
Paquete enviado
pi@raspberrypi:~ $

```

Figura 3-19: Ejecución de envío de trama por consola

ID	Time	Length	Frame
0	17:16:54.790	14	Receive Packet
1	17:17:03.904	14	Receive Packet
2	17:17:06.878	14	Receive Packet
3	17:17:07.685	14	Receive Packet

Receive Packet (API 1)
7E 00 0E 90 00 13 A2 00 41 7C E4 01 00 00 01 31 31 B5
Start delimiter
7E
Length
00 0E (14)

Figura 3-20: Recepción de la trama proveniente de Raspberry Pi

Al chequear por medio del software XCTU (Figura 3-20) se confirma que la función para generar una trama y enviarla a una fuente receptora a concluido sin problemas, ya que al verificar los bytes de la recepción estos nos muestran la dirección MAC desde donde se generó el mensaje y también se verifican el penúltimo y antepenúltimo byte el cual es el contenido del mensaje.

3.4.2.2 Recepción de tramas dirigidas al coordinador

Para la etapa de recepción de tramas por medio del nodo coordinador se realizaron pruebas mediante la comunicación entre Raspberry pi y el software XCTU, al igual que el procedimiento anterior se generó una trama por medio del software de DIGI, el cual tendrá que ser leído de forma completa, sin embargo, se definen los parámetros de recepción que al igual que en el desarrollo del nodo receptor deberá cumplir con los bytes a chequear para ser indexada la trama de forma completa.

La función para el bloque del programa se llamó rx (Listado 3-4), la estructura de esta función consiste en habilitar el puerto serial, luego este queda esperando la recepción de tramas definidas de un largo de 18 bytes los cuales al ser chequeados se indexan a la variable data_struc, esta variable aloja los bytes recibidos lo cuales son alojados en una tupla llamada frm, luego con la sentencia if se verifica que estén alojados en la posición 0 el byte 0x7E y en la posición 3 el byte

0x90 que corresponden al primer byte de mensaje y al byte de trama recibida, para finalizar imprimiendo todos los bytes de la trama (Figura 3-21).

Listado 3-4: Función para recepción de tramas

```
def rx():
    leer_frame=ser.readline(18)
    if len(leer_frame)==18:
        data_struct=struct.Struct('=BBBBBBBBBBBBBBBBBBBB')
        frm=data_struct.unpack(leer_frame)
        if frm[0]==0x7E and frm[3]==0x90:
            for x in frm[0:18]:
                print hex(x).upper()

            print "Recibido"
        else:
            print "Tengo problemas"
```

```
pi@raspberrypi:~ $ python probandotrama.py
0X7E
0X0
0XE
0X90
0X0
0X13
0XA2
0X0
0X41
0X7B
0XD2
0X90
0X10
0X3B
0X1
0X31
0X31
0XEE
Recibido
pi@raspberrypi:~ $
```

Figura 3-21: Bytes recibidos por la función rx

3.4.2.3 Confirmación de envío de tramas

Los módulos XBee al enviar un mensaje proporcionan un registro de estado de transmisión, este registro es conveniente agregar en el programa ya que generara la confirmación de que la orden de habilitar los nodos receptores fue realizada. Para realizar esta confirmación de envió del mensaje, se creó una función llamada chequeo que va de la mano con la función Tx, una de las principales características de esta función será leer los bytes que entrega el registro de estado y comparar que sean iguales ciertos bytes que indican la confirmación del chequeo, para este caso

es importante reconocer el byte 0x7E que indica el inicio de trama, el byte 0x8B que indica el estado de la transmisión y en la posición 8 el byte 0x00 que indica transmisión correcta (Listado 3-5).

Listado 3-5: Función para verificar que mensaje fue recibido

```
def chequeo(): #funcion para chequear estado envio de mensaje
    leer_check=ser.readline(11)
    if len(leer_check)==11:
        data_struct=struct.Struct('=BBBBBBBBBBBB')
        frm_chequeo=data_struct.unpack(leer_check)
        if frm_chequeo[0]==0x7E and frm_chequeo[3]==0x8B and
frm_chequeo[8]==0x00:
            print "mensaje enviado con éxito, riego activad
        else:
            print "tengo problemas"
```

3.4.3 Desarrollo del programa básico

El programa se desarrolló en base a las tres funciones vistas anteriormente, ya que la función básica del programa debe ser la recepción de tramas el envío y su respectivo chequeo de envío. Este programa debía contener ciertas características necesarias para un prototipo básico de riego como lo es saber la dirección MAC a la cual corresponde cada nodo emisor de datos ya que estos van enlazados con un nodo receptor (Frames de MAC) que es el que activa el riego para la zona sensorizada, el esquema de la Figura 3-22 muestra a grandes rasgos como es el funcionamiento del programa básico.

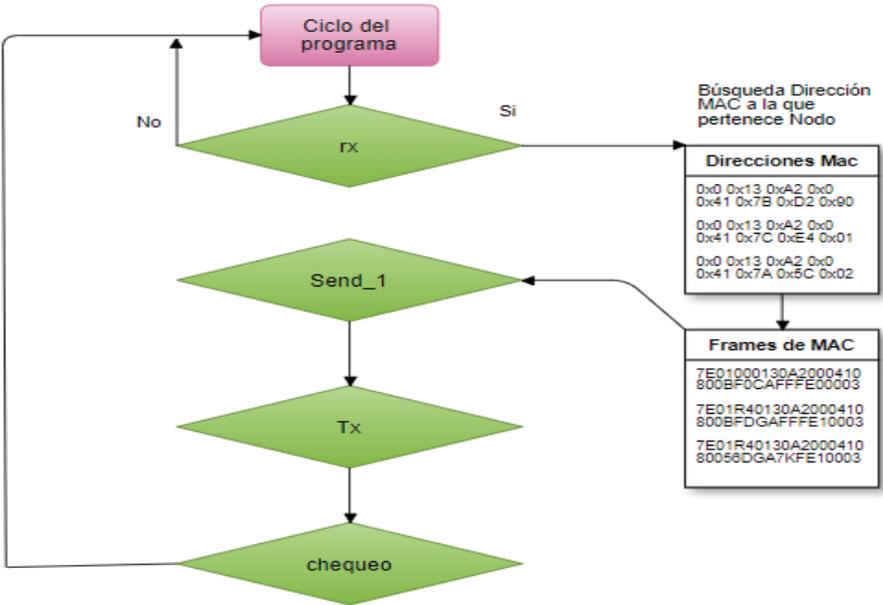


Figura 3-22: Diagrama básico de funcionamiento programa coordinador

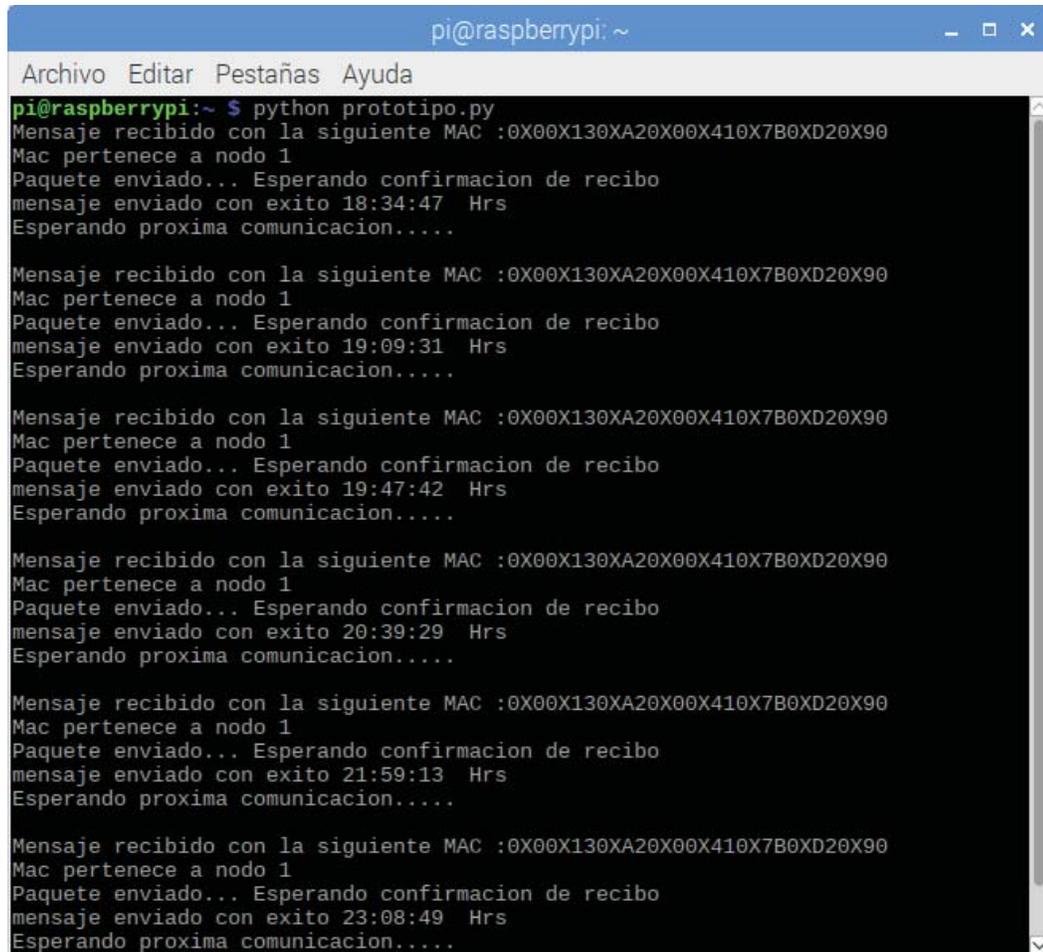
Se agregó una cuarta función llamada Send_1 (Listado 3-6), en su primera etapa esta función tiene en una lista todas las tramas posible a enviar a nodos receptores, para luego pasar a la etapa que consiste en comparar la dirección MAC que lee el puerto serial de un frame recibido con las direcciones MAC que tiene alojada en una lista del programa, con esta comparación sigue a una sentencia if seguido de sentencias elif, las cuales comparan si la dirección MAC es igual a alguna de las que tiene alojadas dentro del programa, al acusar que esta dirección es igual a alguna de las direcciones alojadas envía la trama necesaria al dispositivo receptor.

Listado 3-6: Función Send_1

```
def Send_1(str1):  
  
    if str1 == Tx1:  
        print "Mac pertenece a nodo 1, activar sector 1"  
        str2=li[0]  
        Tx(str2)  
        time.sleep(1)  
  
    elif str1==Mac[1]:  
        print "MAC pertenece a nodo 2, activar sector 2"  
        str2=li[1]  
        Tx(str2)  
        time.sleep(1)
```

3.4.3.1 Pruebas de comunicación entre dos dispositivos

Las pruebas de comunicación se realizaron entre software XCTU y Raspberry Pi, de esta manera se entiende de forma más clara el proceso de comunicación de todo el programa y logra de forma rápida generar envíos de datos compatibles con la recepción del nodo coordinador. Las primeras pruebas consistían en que el coordinador recibía un mensaje de un nodo, este buscaba la MAC del nodo de dónde provenía el mensaje y con respecto a esa información el programa enviaba una trama al receptor correspondiente. En primera instancia, se realizaron pruebas con dos dispositivos, pero para generar confianza en el programa se fue generando envíos de frames en tiempos aleatorios, esto para garantizar que el nodo coordinador siempre estuviera alerta de recibir un frame, para esto se agregó en el código la hora y la fecha en donde ocurría la comunicación entre dispositivos y así dar confianza en que el programa funciona de forma estable para comunicaciones futuras. Se generaron envíos de datos por alrededor de 6 horas, los cuales no se verificaron problemas de comunicación entre los dispositivos, esto genera confianza en que el dispositivo no tendrá problemas de comunicación esperando recibo de datos en tiempos prolongados (Figura 3-23).



```
pi@raspberrypi: ~
Archivo Editar Pestañas Ayuda
pi@raspberrypi:~ $ python prototipo.py
Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1
Paquete enviado... Esperando confirmacion de recibo
mensaje enviado con exito 18:34:47 Hrs
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1
Paquete enviado... Esperando confirmacion de recibo
mensaje enviado con exito 19:09:31 Hrs
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1
Paquete enviado... Esperando confirmacion de recibo
mensaje enviado con exito 19:47:42 Hrs
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1
Paquete enviado... Esperando confirmacion de recibo
mensaje enviado con exito 20:39:29 Hrs
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1
Paquete enviado... Esperando confirmacion de recibo
mensaje enviado con exito 21:59:13 Hrs
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1
Paquete enviado... Esperando confirmacion de recibo
mensaje enviado con exito 23:08:49 Hrs
Esperando proxima comunicacion....
```

Figura 3-23: Prueba de comunicación

3.4.3.2 Prueba de comunicación entre tres dispositivos

Para esta etapa del proyecto se realizaron pruebas con 3 dispositivos, para lo cual se modificaron parámetros del programa, para poder realizar pruebas más reales de funcionamiento, con más de un dispositivo transmisor y receptor, para estas pruebas se alojaron dos direcciones MAC de reconocimiento del programa, las cuales pertenecerían a los nodos transmisores y receptores a la vez, estos fueron configurados como Router y End device respectivamente, esto para generar cohesión entre la red generada con los dispositivos (Figura 3-24).

Se realizaron pruebas de envío de datos aleatorios, esto en relación a que tanto el Router como el End device funcionarían como emisores y receptores de los frames (Figura 3-25).

Con la demostración de que el programa es capaz de coordinar envíos y recepciones de mensajes entre emisores y receptores se da por finalizada la etapa de la implementación de un sistema básico de sensado y toma de decisiones para la implementación de un sistema de riego autónomo.

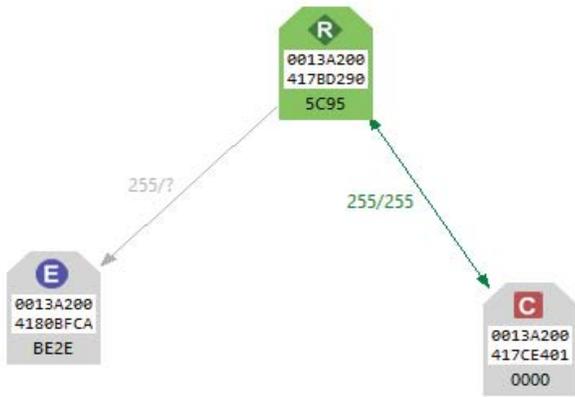


Figura 3-24: Red formada entre dispositivos

```
pi@raspberrypi:~ $ python prototipo.py
Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X800XBFOXCA
MAC pertenece a nodo 2, activar sector 2
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 02:June:2018 Hora: 15:27:52
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1, activar sector 1
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 02:June:2018 Hora: 15:28:22
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1, activar sector 1
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 02:June:2018 Hora: 15:28:26
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X800XBFOXCA
MAC pertenece a nodo 2, activar sector 2
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 02:June:2018 Hora: 17:42:37
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1, activar sector 1
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 02:June:2018 Hora: 17:42:58
Esperando proxima comunicacion....
```

Figura 3-25: Programa funcionando con dos dispositivos emisor y receptor

3.4.4 Escalabilidad del programa

El sistema de riego autónomo entre sus características debe ser escalable, es por tanto que al programa coordinador se le fueron agregando funciones para agregar direcciones MAC desconocidas y crear frames para dispositivos emisores y receptores nuevos, para la escalabilidad

de esta etapa se debe agregar en primera instancia un equipo receptor y luego el equipo transmisor, en esta etapa se desligo del programa la MAC y el frame generado para el dispositivo utilizado como End device, para poder lograr el acoplamiento del nuevo dispositivo generando la escalabilidad asociada al proyecto (Figura 3-26).

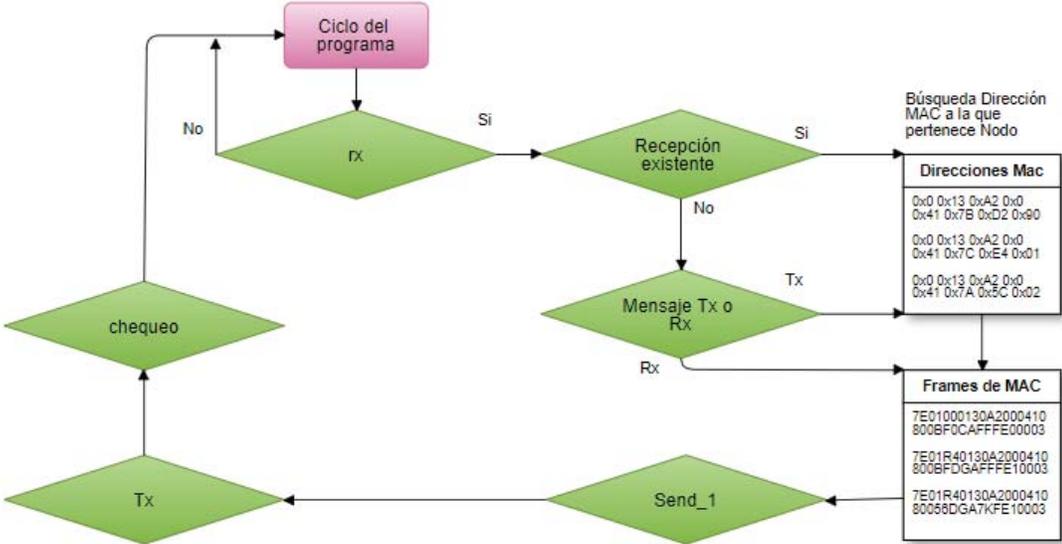


Figura 3-26: Funcionamiento del sistema escalable

3.4.4.1 Acoplamiento módulo receptor

Los equipos receptores que se acoplen deberán enviar una primera trama de datos, donde su característica principal será que RF data (posición 15 y 16 de la trama), será 0X31 y 0X30, lo cual el programa por medio de sus sentencias sabrá que es un dispositivo nuevo a añadir en la lista de tramas para enviar a los dispositivos receptores, el programa empaquetará la dirección MAC correspondiente al tranceptor, luego de esto el dispositivo enviara la MAC a una función llamada generar_frame el cual por medio de un algoritmo generará una trama con los datos correspondientes al dispositivo receptor, luego la función agregar_emisor, agregará esta trama en la última posición de la lista de las tramas alojadas y avisará por consola la trama generada y la lista con tramas dentro del sistema.

Para esta prueba de asociación de dispositivos, la trama que generó el sistema fue la siguiente: 7E001010010013A2004180BFCAFFFE0000313091, la cual quedó almacenada en el programa coordinador (Figura 3-27).

```

pi@raspberrypi:~ $ python prototipo2.py
Nuevo dispositivo receptor
MAC 0X41 0X80 0XBF 0XCA del receptor
Frame generado con nuevo dispositivo Receptor
7E001010010013A2004180BFCAFFFE0000313091
Se agrego correctamente dispositivo receptor
Lista de frames:
['7E001010010013A200417BD290FFFE00003131BC', '7E001010010013A2004180BFCAFFFE000
313091']

Esperando proxima comunicacion....

```

Figura 3-27: Prueba de acoplamiento de dispositivo receptor

3.4.4.2 Acoplamiento módulo receptor

Para la etapa del acoplamiento del módulo coordinador, el dispositivo emisor funciona igual que cualquier dispositivo emisor del sistema, la diferencia se genera cuando dispositivo coordinador al leer los datos enviados por el nuevo módulo emisor no reconoce la dirección MAC alojada en la lista del sistema, para esta etapa el programa envía esta MAC no reconocida a una sentencia la cual la enviará a la función agregar_mac. Esta función se encarga de agregar en la última posición de la lista las nuevas MAC reconocidas por el sistema, el programa mostrara por consola que esta nueva dirección es agregada y también mostrara la actualización de las listas de MAC dentro del sistema (Figura 3-28), además inmediatamente enviara un mensaje a su nuevo dispositivo enlazado el cual le pide que debe activarse.

```

La MAC no esta almacenada, se procede a acoplar Nodo
fecha: 06:June:2018 Hora: 00:28:02
Se procede a agregar la siguiente MAC: 0X00X130XA20X00X410X800XBF0XCA
Lista de direcciones MAC:
['0X00X130XA20X00X410X7B0XD20X90', '0X00X130XA20X00X410X800XBF0XCA']

MAC pertenece a nodo 2, activar sector 2
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 06:June:2018 Hora: 00:28:03
Esperando proxima comunicacion....

```

Figura 3-28: Nuevo dispositivo emisor detectado por programa

Con estas pruebas realizadas se comprueba que el programa adquiere escalabilidad deseada, con la característica que para poder agregar los nuevos dispositivos emisores y receptores es necesario indexar en primera instancia el dispositivo receptor para luego acoplar el dispositivo emisor. Para que sus datos de importancia queden guardados en el programa principal y coordinador del Sistema de Riego Autónomo Escalable (Figura 3-29).

```
Se agrego correctamente dispositivo receptor
Lista de frames:
['7E001010010013A200417BD290FFFE00003131BC', '7E001010010013A2004180BFCAFFFE0000
313091']

Esperando proxima comunicacion....

La MAC no esta almacenada, se procede a acoplar Nodo
fecha: 06:June:2018 Hora: 00:28:02
Se procede a agregar la siguiente MAC: 0X00X130XA20X00X410X800XBF0XCA
Lista de direcciones MAC:
['0X00X130XA20X00X410X7B0XD20X90', '0X00X130XA20X00X410X800XBF0XCA']

MAC pertenece a nodo 2, activar sector 2
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 06:June:2018 Hora: 00:28:03
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X7B0XD20X90
Mac pertenece a nodo 1, activar sector 1
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 06:June:2018 Hora: 00:44:07
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0X00X130XA20X00X410X800XBF0XCA
MAC pertenece a nodo 2, activar sector 2
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 06:June:2018 Hora: 00:44:13
Esperando proxima comunicacion....
```

Figura 3-29: Ejecución del programa con los dispositivos agregados

3.5 Conclusión del capítulo

Llevar a cabo el desarrollo del proyecto fue una tarea que implicó aprender y comprender bastantes conceptos teóricos y técnicos, los cuales a medida que se iba avanzando en el desarrollo y funcionamiento de los dispositivos fue adquiriendo conocimientos que hoy en día deben estar insertos en la formación básica de un futuro profesional.

Generar escalabilidad en un sistema congregó bastante tiempo de estudio y retroalimentación entre estudiante y profesor ya que fue la etapa más complicada al momento de desarrollar un programa que requiera este tipo de envergadura, por lo cual se generó un concepto de aprendizaje distinto a lo desarrollado en las primeras etapas de trabajo.

4 Desarrollo del sistema

Para la implementación de un Sistema de Riego es necesario realizar pruebas concretas de funcionamiento. En este capítulo se detallará el paso a paso del funcionamiento de una maqueta prototipo básica, en la cual se mostrará los materiales a utilizar y el funcionamiento tanto de dispositivo emisor y receptor, los cuales son los elementos con mayor porcentaje de estar expuestos a ambientes hostiles.

4.1 Desarrollo del nodo transmisor

Para el desarrollo del nodo transmisor se utilizaron los siguientes materiales:

- Arduino Uno
- XBee S2C
- Sensor Humedad FC-28
- Sensor de Temperatura DS18B20
- Placa PCB
- Caja de Protección eléctrica
- Cables
- Resistencia 4,7K[Ω]
- Transformador 7[V]-1[A]

Se realizaron todas las conexiones correspondientes en base al prototipo mostrado capítulo 3, por lo cual se procedió a soldar en la placa PCB los componentes a utilizar en el nodo de sensores (Figura 4-1). Luego de esto se procedieron a realizar distintas pruebas de comunicación y funcionamiento arrojando resultados positivos. Entre las pruebas fueron someter a funcionamiento en exterior los sensores y captar las señales tomadas por los sensores, también se hicieron pruebas de medición de rangos extremos, como lo es poner a temperaturas altas y bajas, lo mismo ocurrió con el sensor de humedad, pero con rangos para este sensor, lo cual para todos estos rangos de medida dieron resultados óptimos en el funcionamiento.



Figura 4-1: Nodo sensores transmisor en procesos de pruebas

4.2 Desarrollo del nodo actuador

Para el desarrollo del nodo actuador se utilizaron los siguientes materiales.

- Arduino Uno
- XBee S2C
- Relé 5[V] a 12[V]
- Electroválvula
- Placa PCB
- Caja de Protección eléctrica
- Cables
- Transformador 12[V]- 500m[A]
- Diodo LED



Figura 4-2: Nodo Actuador en proceso de prueba

Para la implementación de este prototipo también se realizaron las conexiones correspondientes, en donde la placa PCB utilizada es de menor tamaño ya que se requieren menos conexiones que el nodo de sensores (Figura 4-2), uno de los primeros problemas en las pruebas de este nodo fue la dirección de conexión de la electroválvula, ya que viene marcada la dirección de flujo de agua,

dato que al conectarla al revés esta produce cavitaciones dentro de la manguera de agua y electroválvula, esto debido a su sistema de bloqueo interno muy parecido a las electroválvulas que se utilizan en la neumática e hidráulica , lo cual puede provocar el deterioro y mal funcionamiento de esta.

Luego de chequear las pruebas de funcionamiento el nodo actuador está listo para ser utilizado.

4.3 Procedimientos de Pruebas

Para el procedimiento de pruebas se instaló el módulo coordinador en un lugar alejado a la zona de pruebas (Figura 4-3), pero mediante el programa VNC se puede interactuar y ver el funcionamiento del sistema, luego de esto se procedieron a instalar los nodos tanto transmisor como receptor. En la instalación de la electroválvula se ocuparon dos trozos de manguera con sus respectivos conectores y se fijaron con alambres para no tener problemas de fugas (Figura 4-4).



Figura 4-3: Sensores y módulo en terreno



Figura 4-4: Nodo actuador

El sensor de temperatura no tendría inconvenientes para habilitar el sensor de humedad puesto en tierra seca, por lo tanto, al detectar falta de agua procede a comunicarse con el nodo coordinador, genera el proceso para enviar la orden de activación al nodo actuador, por lo cual prende el led verde de la caja que significa que se activó el riego. Por cambios del programa para uso de prototipo deshabilitará en 30 segundos la electroválvula para corte de agua siempre y cuando el sensor de humedad no esté en presencia de falta de esta. Se verificó el funcionamiento del programa (Figura 4-5) sin obtener errores.

```

pi@raspberrypi ~
Archivo Editar Pestañas Ayuda
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 23:July:2018 Hora: 17:20:44
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0x00x130xA20x00x410x7B0xD20x90
Mac pertenece a nodo 1, activar sector 1
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 23:July:2018 Hora: 17:20:57
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0x00x130xA20x00x410x7B0xD20x90
Mac pertenece a nodo 1, activar sector 1
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 23:July:2018 Hora: 17:21:10
Esperando proxima comunicacion....

Mensaje recibido con la siguiente MAC :0x00x130xA20x00x410x7B0xD20x90
Mac pertenece a nodo 1, activar sector 1
Paquete de activacion de zona enviado... Esperando confirmacion de recibo
mensaje enviado con exito, riego activado
fecha: 23:July:2018 Hora: 17:23:45
Esperando proxima comunicacion....

```

Figura 4-5: Funcionamiento del programa en terreno

Con esta breve prueba se da por finalizado y comprobado el funcionamiento del Sistema de Riego Autónomo Escalable.

4.4 Conclusión del capítulo

El desarrollo de un prototipo de trabajo requiere de bastante tiempo de dedicación, es necesario realizar pruebas de ensayo y error debido a que siempre aparecen pequeños problemas de funcionamiento y se deben mitigar estos pequeños problemas para así obtener un sistema que sea confiable al momento de ser utilizado. Las pruebas realizadas en terreno sirvieron para comprender que si bien un proyecto se puede idealizar de una forma capaz de solucionar un problema sin tener errores, este debe ser ejecutado de forma cuidadosa en sus primeros pasos para poder responder a posibles eventualidades que se puedan generar en el transcurso del proceso.

5 Evaluación económica

En la última etapa del proyecto es necesario generar una evaluación económica, con el fin de determinar si es posible una futura puesta en marcha del proyecto. La realización e implementaciones futuras de alternativas distintas a sistemas de riego convencionales podrían ser capaces de generar una mejor alternativa de ejecución para empresas y personas, tanto en la parte de ahorro en gastos por agua y búsqueda en la eficiencia y optimización de los recursos. Por esto, se desarrollará una evaluación económica del sistema de riego como un producto de mercado con una visión de proyecto de 5 años.

5.1 Estudio de mercado

Es necesario realizar un estudio de mercado cuando se tiene la intención de ingresar con un producto nuevo o una mejora de este, ya que al observar el comportamiento de los consumidores se pueden generar ideas sobre la viabilidad comercial de la actividad económica. Dos factores importantes que conocer en este tipo de estudios serán la estrategia de la empresa y el análisis del consumidor.

La estrategia de la empresa se basa en el rumbo que debe tomar esta, ya sean objetivos de venta, confianza en el producto, recursos y estudios de la competencia. Es recomendable para toda nueva empresa optar por dos estrategias posibles.

- Liderazgo en costo: Mantenerse en los primeros lugares de la lista a nivel competitivos en materia de costos.
- Diferenciación: Crear valor agregado sobre el producto ofrecido para que este sea percibido en el mercado como único ya sea en diseño, imagen, atención al cliente, entre otras.

El análisis al consumidor se puede verificar sobre varias aristas, ya sea a través de encuestas a la población de interés, análisis estadísticos que posean instituciones públicas o privadas, o bien a través de expertos en el área a la cual se enfocará la venta del insumo o servicio de la empresa.

5.1.1 Mercado proveedores

En un análisis económico es importante realizar un estudio del mercado de proveedores de los dispositivos electrónicos esenciales a ocupar en la implementación del producto, ya que la

empresa constantemente requerirá productos para mantener un mínimo de stock en su bodega de almacenamiento. Para esto es necesario contar con la información actualizada de precios y stock de componentes, tiempos y costos de envío, para evaluar la o las alternativas más convenientes ya sea con proveedores nacionales o internacionales, en este último también contar con la información de los costos de impuestos aduaneros.

Entre los posibles proveedores tanto fuera como dentro del país podemos encontrar (Tabla 5-1):

Tabla 5-1: Proveedores

Nacionales	Internacionales
MCIelectronics	eBay
MaxElectrónica	Mouser
ALLIED electronics&automation	Aliexpress
Tectronix	Amazon

5.1.2 Mercado consumidores

El mercado de consumidores se agrupará en todas las posibles empresas y/o personas que estén interesadas en la adquisición del producto. El análisis de mercado para los futuros posibles consumidores puede realizarse en una forma sistemática utilizando las llamadas 6 O's [16]:

- Objeto de compra: Identificar qué es lo que compra el consumidor.
- Ocupantes del mercado: Quién está en el mercado de consumidores (quiénes adquieren el producto).
- Organización de compra: Identificar quien participa en la decisión de la compra.
- Ocasión de compra: Cuándo compran los consumidores.
- Objeto de compra: Por qué compran los consumidores.
- Operaciones de compra: Cómo compran los consumidores.

Las características del proyecto estarán dirigidas a todo tipo de personas, empresas agrícola o entidades que requieran automatizar sus áreas verdes, no obstante, se empeñará a dirigir el proyecto a pequeños y medianos agricultores con el fin de postular a subsidios que entrega el estado por medio de INDAP [17] (Instituto de desarrollo agropecuario) para riego tecnificado.

5.1.3 Mercado competidores

El mercado competidor está formado por el conjunto de empresas que en la actualidad satisfacen total o parcialmente las necesidades de los potenciales consumidores del proyecto, el estudio de estos competidores tiene como propósito fundamental el conocer el funcionamiento de empresas de similares características, tanto en aspectos de balances, gestión, marketing entre otras.

En Chile existen una gran variedad de empresas relacionadas con la venta de productos o servicios de riego tecnificado entre los cuales podemos encontrar (Tabla 5-2):

Tabla 5-2: Empresas dedicadas a sistemas de riego en Chile

Empresas Competidoras	
Autoriego	Olivos
Agrolar	Riego Palo Alto
Agroservicios Capurro	Green IQ
Riego Chile	Livn

La mayoría de estas empresas ofrecen la instalación de sistemas de riego programados en rangos de horas enfocados al sector agrícola, en cambio solo una empresa ofrece autonomía de riego, pero dedicado solo en áreas verdes.

5.2 Estudio económico

Una de las fases más importantes antes de empezar cualquier proyecto es realizar un estudio de viabilidad económica, para verificar la factibilidad de la puesta en marcha y proyección en el tiempo de cualquier tipo de proyecto. Es por esto que se realizará el siguiente estudio económico con el fin de estimar una proyección de rentabilidad que obtendrá el proyecto. Para la realización de esto se analizarán aspectos relevantes, como lo es la inversión inicial, cálculos de costos de operación los cuales se dividen en costos de producción y administrativos. Así como otro factor de gran importancia será el análisis del flujo de caja proyectado, el cálculo del valor actual neto (VAN) y la tasa interna de retorno (TIR).

El contexto de este análisis será la formación de una empresa dedicada a ofrecer el servicio de mejoras a los sistemas de riego, tanto para el sector agrícola y personas o instituciones que necesiten automatizar el riego de sus áreas verdes.

La empresa ofrecerá desde un sistema básico de automatización inteligente de riego a sistemas con más cantidad de dispositivos enlazados, incluyendo la característica principal de que a medida que el cliente necesite expandir sectores donde necesita sensorizar y regar, se puedan adquirir los elementos transmisores y receptores correspondientes para la escalabilidad de sus requerimientos. La empresa también ofrecerá la instalación de los dispositivos en terreno y modificación de ajustes técnicos que requiera el cliente.

5.2.1 Inversión inicial

En la inversión inicial se asociará la suma de todos los costos necesarios para poder iniciar una actividad de trabajo, donde estos gastos serán de suma importancia para la puesta en marcha del proyecto. Dentro de los costos necesarios se pueden considerar: instalaciones, maquinaria, equipos, arriendo de oficina, muebles, componentes electrónicos como materia prima del producto y una cantidad de dinero para gastos extras en caja.

Las inversiones que se deben tomar en cuenta antes de la puesta en marcha del proyecto se clasifican en tres tipos:

- Activos Fijos o Tangibles
- Activos Nominales o Intangibles
- Capital de Trabajo

Los activos fijos son aquellos utilizados en el proceso de transformación de insumos o apoyo a la operación de explotación de la empresa necesaria para que esta pueda desarrollar funciones

Para este proyecto se considerarán:

- Insumos electrónicos: Son los que permitirán manipular, medir y desarrollar el sistema a vender (Tabla 5-3).

Tabla 5-3: Insumos electrónicos

Insumos Electrónicos	Costo Unidad	Cantidad	Costo Total
Fuente de poder regulable	\$20.000	2	\$40.000
Multímetro digital	\$16.000	2	\$32.000
Herramientas de trabajo	\$200.000	1	\$200.000
Osciloscopio digital	\$200.000	1	\$200.000
Programador Xbee + cable	\$7.000	3	\$21.000
Total sin IVA			\$414286
Total con IVA			\$493.000

- Insumos computacionales: Son los costos asociados a una eventual oficina de atención al público y clientes, el cual requerirá contar con computadores más impresoras, para efectos de entrega de cotizaciones y registros importantes, además de accesorios extras que agilicen el trabajo de oficina (Tabla 5-4).

Tabla 5-4: Insumos computacionales

Insumos computacionales	Costo Unidad	Cantidad	Costo Total
Notebook	\$300.000	2	\$600.000
Impresora multifuncional	\$100.000	1	\$100.000
Accesorios	\$30.000		\$30.000
Total sin IVA			\$613.445
Total con IVA			\$730.000

Mobiliario: En los gastos del mobiliario se incluirá todo lo asociado a la organización del área física de trabajo del personal, ya que se propone tener una oficina de atención al público en conjunto con un taller de pruebas armado y servicio técnico. Es necesario contemplar gastos en muebles y todo lo relacionado con organización del servicio, para generar una confiabilidad como futura empresa y generar profesionalismo con el cliente (Tabla 5-5).

Tabla 5-5: Mobiliario

Mobiliario	Costo Unidad	Cantidad	Costo Total
Escritorio oficina	\$80.000	2	\$160.000
Escritorio servicio técnico	\$50.000	2	\$100.000
Sillas	\$50.000	6	\$300.000
Repisas y Organizadores	\$150.000		\$150.000
Total sin IVA			\$596.639
Total con IVA			\$710.000

Para Activos nominales o intangibles corresponderán a servicios o derechos necesarios para la puesta en marcha del proyecto, estas se dividirán en gastos de organización y gastos por incorporación de servicios.

- Gastos de organización: Estos gastos se estiman mediante consultas a un contador auditor. Los servicios que presentarán son la realización del balance anual, el cual se realizara al 30 de diciembre de cada año, además del pago de impuestos mensuales y pago físico PPM (Pago Provisorio Mensual), estos servicios se realizarán durante el tiempo que se estima sea la duración del proyecto. Los costos que deben ir asociados en este apartado son los asociados a la adquisición de talonarios de boletas, libros contables y facturas de ventas (Tabla 5-6).

Tabla 5-6: Gastos contables

Gastos contables	Costo unidad	Cantidad	Total
Libro compra	\$1.000	1	\$1.000
Libro venta	\$1.200	1	\$1.200
Libro mayor	\$2.000	1	\$2.000
Libro balance	\$4.500	1	\$4.500
Boletas de venta	\$2.500	4	\$10.000
Total sin IVA			\$15.714
Total con IVA			\$18.700

- Gastos por servicios adicionales y arriendo: La incorporación de servicios de comunicación son parte del gasto más el pago de arriendo de local con mes de garantía incluido (Tabla 5-7).

Tabla 5-7:Detalle Servicios Adicionales y Arriendo

Servicios Iniciales	Valor
Servicio de Internet y telefonía	\$32.000
Arriendo local	\$250.000
Total Con IVA	\$282.000

En la última etapa de la inversión se considera la inversión en capital de trabajo. Esto se entiende a los activos circulantes o disponibles en la empresa con los que se cuenta para hacer frente a las obligaciones diarias o que estimen convenientes para su desarrollo diario, a corto o largo plazo. Este capital representa los recursos con los que cuenta la empresa para sus operaciones iniciales en cuanto a capital de trabajo.

- Capital de trabajo: Capital en dinero para uso de gastos necesarios iniciales y emergencias a cubrir en general. Para esto se calculará por el modelo del saldo óptimo de efectivo (5-1) [18]. Se estima que el capital óptimo de trabajo es el proyectado por la estimación de las primeras ventas del año, ya que la empresa necesita una cantidad elevada de dinero para la iniciación de su actividad de forma completa, entre estas compras se contempla el primer lote de materias primas para el armado de los dispositivos y otros gastos extras. Las cantidades de costo efectivo por unidad estimadas son las analizadas en la sección 5.2.4, mientras que el desembolso anual se estima entre lo que se gastará del costo total de unidades más los gastos administrativos (Tabla 5-15) y la tasa de interés será del 15%. Segundo lo calculado se necesitan \$8.000.000 aproximados de capital de trabajo.

$$C = \sqrt{\frac{2BT}{i}} \quad (5-1)$$

En donde:

- C= Capital óptimo de trabajo.
- B=Costo de hacer efectivo un valor.
- T=Desembolso anual.
- i: Tasa de interes.

El resultado final de todos los factores para la inversión inicial dan la suma de \$10.233.700 (Tabla5-8).

Tabla 5-8: Inversión Inicial Total

Inversión Inicial	Valor
Activos fijos	\$1.933.000
Activos nominales	\$300.700
Capital de trabajo	\$8.000000
Total	\$10.233.700

5.2.2 Financiamiento de la inversión Inicial

El proyecto puede ser financiado por medio de capital propio o con préstamo o crédito de terceros. Se estimará un préstamo de \$9.000.000 a pagar en 3 años a una entidad bancaria, este prestamos será personal, ya que los bancos en sus condiciones de préstamos realizan estos a empresas que lleven más de un año de funcionamiento. Se simuló un prestamos con el total de la

inversión inicial total, en donde se escogió el prestamos con la tasa de interés y el CAE (Carga Anual Equivalente) más bajo. Este prestamos se simulo al pago de 36 cuotas iguales con una tasa de interés del 1,68%, un CAE del 20,75%, el valor de la cuota será igual a \$339.315. Lo que por año se pagara una cuota de \$4.071.780 (Tabla 5-9).

Tabla 5-9:Detalle del Crédito

Detalle del Crédito	Valor
Monto del crédito liquido	\$9.000.00
Gastos notariales	\$1.940
Impuesto	\$72.596
Monto total del crédito	\$9.074.536
36 cuotas iguales	\$339.315
Costo total del crédito	\$12.215.351

5.2.3 Depreciación de activos fijos

La depreciación de un activo fijo corresponde el valor que va sufriendo un bien producto de su uso o desgaste. En medida que avance el tiempo el servicio decrece el valor contable de dicho activo.

La vida contable de un activo depreciable da por inicializado cuando este es comprado y empieza la explotación económica de él, esto hasta la fecha en que cumple su depreciación, el método más utilizado para la depreciación es el de la línea Recta (5-2).

$$DI = \frac{(Ci - Vr)}{N} \quad (5-2)$$

En donde:

- N= Vida útil en años
- Ci= Costo inicial
- Vr= Valor residual
- DI= depreciación anual

Tabla 5-10: Depreciación activos fijos

Activos Fijos	Valor con IVA	Vida Útil	Valor Residual	Depreciación Anual
2 Fuente de poder regulable	\$40.000	4	\$5.000	\$8.750
2 Multímetro digital	\$32.000	4	\$4.000	\$7.000
Herramientas de trabajo	\$200.000	4	\$50.000	\$37.500
Osciloscopio digital	\$200.000	4	\$50.000	\$37.500
2 Notebook	\$600.000	4	\$100.000	\$125.000
Impresora multifuncional	\$100.000	4	\$30.000	\$17.500

Accesorios	\$30.000	3	\$0	\$7.500
2 Escritorio oficina	\$160.000	5	\$20.000	\$28.000
2 Escritorio servicio técnico	\$100.000	5	\$17.000	\$16.600
6 Sillas	\$300.000	5	\$17.000	\$56.600
Artículos oficina	\$150.000	3	\$0	\$50.000
Total con IVA				\$391.950

Para efectos de cálculo se aproximará a la suma de \$392.000 (Tabla5-10).

5.2.4 Costos de operación

También llamados costos de producción son los gastos necesarios para mantener un proyecto, línea de procesamiento o un equipo en funcionamiento durante el periodo de existencia. Dentro de estos se deben analizar los costos del producto y los costos de periodo.

- **Costos de producción o costos del producto:** Es el valor del conjunto de bienes y esfuerzo en que se va a incurrir, lo que debe consumir el centro de producción para la obtención de producto terminado con las condiciones para ser entregado al área de ventas. Este costo será lo que genere producir el o los elementos a vender. Estos costos se dividirán en tres. Primero será el costo del sistema coordinador y los siguientes para los sistemas transmisores y receptores. El costo del dispositivo coordinador tendrá un valor de \$86.026 (Tabla 5-11), el dispositivo transmisor tendrá un costo de \$50.000 (Tabla 5-12) y el dispositivo receptor un valor de \$64.00 (Tabla 5-13).

Tabla 5-11: Detalle Dispositivo Coordinador

Componente	Descripción	Cantidad	Precio Unitario	Bruto	Neto
Raspberry Pi	Computador	1	\$27.000	\$27.000	\$32.130
Micro SD 16 GB	Memoria	1	\$7.000	\$7.000	\$8.330
Xbee S2C	Transceptor	1	\$13.990	\$13.990	\$16.648
Adaptador Xbee	Adaptador	1	\$859	\$859	\$1.022
Case + disipador	Protector	1	\$3.250	\$3.250	\$3.868
Transformador	Transformador	1	\$2.320	\$2.320	\$2.761
Batería 5V-2Ah	Batería	1	\$10.530	\$10.530	\$12.531
Panel solar 10W	Panel	1	\$7.242	\$7.242	\$8.618
Cables	Cables	10	\$10	\$100	\$119
Total				\$72.291	\$86.026

Tabla 5-12: Detalle Dispositivo Transmisor

Componente	Descripción	Cantidad	Precio Unitario	Bruto	Neto
Xbee S2C	Transceptor	1	\$13.990	\$13.990	\$16.648
Arduino Uno	Microcontrolador	1	\$3.600	\$3.600	\$4.284

Adaptador Xbee	Adaptador	1	\$859	\$859	\$1.022
Case Protector	Case	1	\$1.000	\$1.000	\$1.190
FC-28	Sensor humedad	4	\$250	\$1.000	\$1.190
	Sensor				
DS18B20	temperatura	1	\$3.500	\$3.500	\$4.165
Panel Solar	Panel	1	\$7.242	\$7.242	\$8.618
Batería					
Arduino	Batería	1	\$10.530	\$10.530	\$12.531
Cables	Cables	10	\$10	\$100	\$119
Total				\$41.821	\$49.767

Tabla 5-13: Detalle Dispositivo Receptor

Componente	Descripción	Cantidad	Precio Unitario	Bruto	Neto
Xbee S2C	Transceptor	1	\$13.990	\$13.990	\$16.648
Arduino Uno	Microcontrolador	1	\$3.600	\$3.600	\$4.284
Adaptador Xbee	Adaptador	1	\$859	\$859	\$1.022
Case Protector	Case	1	\$1.000	\$1.000	\$1.190
válvula 12V	Electroválvula	1	\$5.000	\$5.000	\$5.950
Relé 5-12 VDC	Relé	1	\$1.000	\$1.000	\$1.190
Panel Solar	Panel	1	\$7.242	\$7.242	\$8.618
Batería Arduino	Batería	1	\$10.530	\$10.530	\$12.531
Batería elec-valv	Batería	1	\$10.300	\$10.300	\$12.257
Cables	Cables	10	\$10	\$100	\$119
Total				\$53.621	\$63.809

Estos valores corresponden a los costos de los componentes para el montaje del dispositivo, en tanto el KIT básico del sistema de riego constará de:

- Módulo Coordinador: En el cual vendrá dentro de la caja una Raspberry Pi 3 Model B con una tarjeta de memoria de 16 GB, un XBee S2C con su respectivo módulo de conexión, Transformador para alimentación y sistema de batería con panel solar para mitigar cortes de energía eléctrica.
- Módulo transmisor: Este módulo vendrá en una caja donde se dispondrá de un Arduino Uno, un XBee S2C para comunicación, un sistema de panel solar con batería para alimentación, un sensor de temperatura para el entorno y cuatro sensores de humedad con alcance máximo de 10 (mts) de distancia.
- Modulo receptor: Este módulo vendrá dentro de una caja que por dentro tendrá un Arduino Uno, módulo relé, módulo XBee S2C, electroválvula, sistema de alimentación para modulo y electroválvula.

A este Kit además de su contenido básico se le podrán agregar dispositivos emisores y receptores para generar mayor volumen a cubrir dependiendo las necesidades del cliente, estos módulos

emisores y receptores agregados tendrán las mismas características técnicas de los módulos que vendrán en un kit básico. El costo redondeado del Kit sera de \$200.000 y de la expansión de equipos transmisor y receptor tendrá un costo de \$114.000.

Se estimará un volumen de producción que aumentará en un 10% anual aproximado en los Kits. El volumen de venta de los Kits más la expansión se proyectará abarcando el 3% de los dineros otorgados por subsidios de riegos según las estimaciones de la dirección de gastos del Gobierno de Chile [19] y en menor porcentaje se considera la venta a privados (Tabla 5-14).

Tabla 5-14: Costos de Producción por Año

Concepto	Año 1	Año 2	Año 3	Año 4	Año 5
Kit básico	55	61	66	71	77
Total KIT	\$11.000.000	\$12.200.000	\$13.200.000	\$14.200.000	\$15.400.000
Trasmisor y Receptor	20	30	35	40	45
Total TX y RX	\$2.280.000	\$3.420.000	\$3.990.000	\$4.560.000	\$5.130.000
Total (costo* volumen)	\$13.280.000	\$15.620.000	\$17.190.000	\$18.760.000	\$20.530.000

- Costos del periodo: Estos costos son los que están determinador a los ingresos durante un determinado periodo, por lo tanto, no se incluye como parte integral de los inventarios, estos costos son causa adicional con el fin de poder vender el producto o servicio, estos se dividen en costos de administración y costos de distribución. En estos costos entra el sueldo del personal, al igual que los trámites contables y servicios telefónicos entre otros.

Tabla 5-15: Costo de Administración

Concepto	Costo Mensual	Costo Anual
Jefe de Operaciones	\$600.000	\$7.200.000
Técnico Informático	\$400.000	\$4.800.000
IVA y PPM	\$50.000	\$600.000
Balance anual		\$200.000
Arriendo Local	\$250.000	\$3.000.000
Telefonía e Internet	\$32.000	\$384.000
Luz	\$25.000	\$300.000
Agua	\$5.000	\$60.000
Plan celular	\$30.000	\$360.000
Contador	\$150.000	\$1.800.000
Total		\$18.704.000

Para la venta del kit básico se realizará la estimación de precio con la ecuación del punto de equilibrio (5-3), en donde se le agregará un 10% de utilidades al producto en venta (Tabla 5-16), mientras que los módulos transmisores y receptores se utilizará la ecuación (5-4), con un margen de ganancia del 25%.

$$P = \frac{CF}{X} + CVu \quad (5-3)$$

$$P = \frac{\text{Costo}}{1 - \%Ganancia} \quad (5-4)$$

En donde:

- P=Precio Unitario
- CF= Costo fijo total
- X=Volumen de producción
- CVu= Costo variable unitario

Tabla 5-16: Calculo precio Venta Kit Básico

	Año 1	Año 2	Año 3	Año 4	Año 5
Volumen producción Kit	55	61	66	71	77
Costos de administración	\$18.704.000	\$18.704.000	\$18.704.000	\$18.704.000	\$18.704.000
Costos variables por producción	\$11.000.000	\$11.000.000	\$12.000.000	\$12.000.000	\$13.000.000
Precio venta unitario	\$540.073	\$506.623	\$483.394	\$463.437	\$442.909
Venta con utilidad	\$594.080	\$557.285	\$531.733	\$509.780	\$487.200

El valor de venta del dispositivo Transmisor con Receptor se redondeará en \$152.000

5.2.5 Flujo de caja y cálculo del VAN

Se entiende por VAN (Valor Actualizado Neto) al procedimiento que permite calcular el valor presente de un determinado número de flujos de caja futuros, originados por una inversión [20]. La metodología consiste en descontar al momento actual todos los flujos de cajas futuros o en determinar la equivalencia en el tiempo 0 de los flujos de efectivo en el futuro que genera un proyecto y comparar este resultado con el desembolso inicial. Según formula el VAN se puede calcular como (5-5):

$$VAN = -A + \left[\frac{Q_1}{(1+K)^1} + \frac{Q_2}{(1+K)^2} + \dots + \frac{Q_n}{(1+K)^n} \right] \quad (5-5)$$

Siendo:

- A = Inversión Inicial
- Qn = Flujo de caja
- N = Número de años
- K = Tipo de actualización o descuento (interés)

Tabla 5-17: Flujos de caja

Flujo de caja	Año 0	Año 1	Año 2	Año 3	Año 4	Año 5
Ingresos por venta Coordinador		\$32.674.400	\$33.994.400	\$35.094.400	\$36.194.400	\$37.514.400
Ingresos por Venta TX y RX		\$3.040.000	\$4.560.000	\$5.320.000	\$6.080.000	\$6.840.000
Costo Fijos (-)		\$18.704.000	\$18.704.000	\$18.704.000	\$18.704.000	\$18.704.000
Costos Variables Coordinador (-)		\$11.000.000	\$12.200.000	\$13.200.000	\$14.200.000	\$15.400.000
Costo Variable TX y RX (-)		\$2.280.000	\$3.420.000	\$3.990.000	\$4.560.000	\$5.130.000
Utilidad bruta		\$3.730.400	\$4.230.400	\$4.520.400	\$4.810.400	\$5.120.400
Depreciación (-)		\$392.000	\$392.000	\$392.000	\$392.000	\$392.000
Utilidad antes de impuesto		\$3.410.400	\$3.910.400	\$4.200.400	\$4.490.400	\$4.800.400
Impuesto (-)		\$647.976	\$742.976	\$798.076	\$853.176	\$912.076
Utilidad neta		\$2.762.424	\$3.167.424	\$3.402.324	\$3.637.224	\$3.888.324
Inversión Inicial	-\$10.233.700					
Crédito	\$ 9.000.000					
Amortización crédito		\$ 4.071.780	\$ 4.071.780	\$ 4.071.780		
Flujos de caja	-\$1.233.700	-\$1.367.676	-\$962.676	-\$727.776	\$3.578.904	\$3.830.004

Si en el resultado del VAN este da como resultado mayor a 0, se considera que el proyecto es rentable, por el contrario, si es menor a 0 el proyecto no es rentable. Otra opción es cuando este es igual a 0, esto da a entender que solo el proyecto recuperara la inversión inicial, por lo tanto, no va a generar beneficios y perdidas.

Tabla 5-18: Resultados flujo de caja

15%	TRMA
321.000	VAN
19%	TIR

Con los flujos de caja calculados y una TRMA del 15% se calculó el VAN el cual arrojó un valor positivo, lo cual significa que el proyecto en su idealización como tal es rentable en base a los supuestos calculados, también se puede observar que el proyecto es rentable en su totalidad ya que los flujos de caja cubren los costos involucrados en el desarrollo de este.

La TIR indica que el proyecto es d, ya que a mayor TIR más rentable es el proyecto para este tipo de casos.

5.3 Conclusión del capítulo

En este análisis económico se comprendió la importancia de tener el control de todos los precios dada todas las aristas de la conformación de un proyecto, tanto como analizar el mercado de competidores, público objetivo y los recursos necesarios para la inicialización de la empresa así lo es también el dinero que se necesita para llevar a cabo un negocio y mantener un orden y control de los gastos y ganancias adquiridas para que la empresa funcione sin tener desorden contable.

Discusión y conclusiones

Los avances tecnológicos han permitidos mejorar y automatizar muchos procesos que hace años atrás eran impensados, es por esto, que es de mucha importancia seguir realizando desarrollos tecnológicos en post a la vanguardia y necesidades de las personas y entorno.

El cambio climático es un proceso que en los últimos años ha ido generando preocupación en el medio ambiente, el efecto que causa esta problemática ha tenido repercusión en la mayoría de los países del mundo, no ajeno a esto Chile también ha sufrido notorios cambios en su entorno, es por esto, que el minimizar los efectos y aprovechar los recursos naturales es de suma importancia en el sistema en el que vivimos actualmente, en especial el aprovechamiento de los recursos hídricos, los que se han visto precarizados poco a poco en diferentes zonas.

El objetivo principal del proyecto fue la creación de un sistema de riego autónomo escalable dada la problemática a considerar. Con la realización de este proyecto se tuvo la oportunidad de conocer y trabajar en tecnologías que están siendo cada día más utilizadas entre personas que se están involucrando en el mundo de la electrónica, estas nuevas alternativas tienen la cualidad de que el usuario no debe ser un experto en programación ni en electrónica, pero a medida que va desarrollando habilidades con estos elementos puede ir expandiendo su conocimiento y generar nuevos prototipos.

Entre los aspectos fundamentales en el desarrollo del proyecto fue escoger el dispositivo transceptor de comunicación entre más de dos nodos, ya que estos requerían de un protocolo que fuera adecuado para el armado de la topología necesaria y escalabilidad requerida. Es por esto que los transceptores XBee fueron los seleccionados para la implementación del sistema, ya que el protocolo ZigBee que ellos utilizan rinde de forma óptima en materias de topologías de uso y diseño de implementación. Si bien fue un desafío el aprendizaje y comprensión sobre cómo se comunicaban estos módulos, a medida que se iban realizando pruebas se fue desarrollando un autoaprendizaje y conocimiento en la forma de ir testeando las diferentes maneras de comunicación en tramas que ofrece para generar el software XCTU. Otros puntos importantes de comentar son la compatibilidad que ofrecen estos módulos para funcionar con otros sistemas embebidos y/o microcontroladores. El punto en contra de estos módulos es la no compatibilidad con transceptores de otros fabricantes que trabajen con el protocolo ZigBee.

Generar la escalabilidad del sistema requirió de bastante tiempo de estudio, ya que fue la parte más difícil de implementar en el sistema, pero gracias a que Python ofrece toda su documentación de manera abierta y gratuita se logró generar un programa con la escalabilidad deseada. Es importante mencionar que a raíz de cada etapa de trabajo se fue generando una expansión de ideas para llevar a cabo futuros procesos, y además comprender que la implementación de una red de trabajo está implicada en un gran número de aplicaciones.

Si bien, el desarrollo del prototipo no utilizó una gran cantidad de elementos, si se pudo demostrar y comprobar el funcionamiento íntegro del sistema, tanto como sistema básico a sistema escalable. El desarrollo del prototipo también contrajo conocimientos técnicos aplicados ya que se tuvo que generar la instancia de realizar un prototipo de trabajo que demostrase que todo lo planteado a medida que se desarrollaron las etapas del sistema, eran funcionales en el tiempo.

Se logró abarcar todos los objetivos planteados con respecto a la pauta de trabajo, haciendo énfasis en el objetivo principal que era crear un sistema de riego de función autónoma y con posibilidad de escalamiento, como se puede observar en el desarrollo del informe. También se logró generar que este sistema funcionara acorde a los datos recopilados en terreno y mediante un sistema coordinador actuara conforme a otros sistemas de riego estudiados.

Este proyecto propone generar trabajos futuros, tanto para sistemas de riego como sistemas de redes de sensores, ya que todo lo que pasa por la red es información, que siendo bien utilizada puede servir para mejorar estadísticas y ejecutar procesos de trabajo. Un desarrollo futuro siguiendo la pauta de cómo fue realizado el proyecto del sistema de riego puede ser la conformación de una base de datos para obtener estadísticas de gastos de agua con respecto a la calidad de los cultivos y obtener datos en tiempo real a través de dispositivos móviles. Además, se pueden generar una gran cantidad de trabajos que requieran sensorizado de parámetros y accionamientos electromecánicos que en la actualidad son muy utilizados en la industria.

Finalmente, con respecto al proyecto se puede comentar que este tipo de sistemas son de bastante utilidad para ayudar con el ahorro de los recursos hídricos, pero siempre se debe seguir en búsqueda de la mejora en este tipo de trabajos, ya que se puede generar un valor agregado a la propuesta inicial.

Bibliografía

- [1] Equipo de la Unidad de Cambio Climático y Servicios Ambientales, Corporación Nacional Forestal, «Estrategia Nacional de Cambio Climático y Recursos Vegetacionales,» Jaeel Moraga y Angelo Artori, Santiago de Chile, 2016.
- [2] M. d. Agricultura, *APRUEBA NORMAS PARA EL FOMENTO DE LA INVERSION PRIVADA EN OBRAS DE RIEGO Y DRENAJE*, Chile: Biblioteca del Congreso Nacional de Chile - Legislación Chilena, 2013.
- [3] «Organización de las Naciones Unidas para la alimentación y la agricultura,» [En línea]. Available: <http://www.fao.org/home/es/>. [Último acceso: 5 Octubre 2017].
- [4] I. Association, «irrigation,» [En línea]. Available: <https://www.irrigation.org/2018Show>. [Último acceso: 20 septiembre 2018].
- [5] «tevatronic,» [En línea]. Available: <http://tevatronic.net/>. [Último acceso: 7 Octubre 2017].
- [6] P. Isaac Klein, *THRESHOLD TENSION AUTONOMOUS IRRIGATION (TTAI) OF OLIVE CONTROLS IRRIGATION EFFICIENCY, WATER USE EFFICIENCY AND FRUIT QUALITY*, International Society for Horticultural Science, 2011.
- [7] J. A. C. Silva, «Sistema de Riego Autónomo Basado en Internet de las Cosas,» Universidad Internacional de La Rioja, Neiva-Huila, 2016.
- [8] «Raspberry Pi,» [En línea]. Available: www.raspberrypi.org. [Último acceso: 7 Septiembre 2017].
- [9] H. K. - A. Willig, *Protocols and Architectures for Wireless Sensor Networks*, Londres: John Wiley & Sons, Ltd, 2005.

- [10] «IEEE802,» [En línea]. Available: <http://www.ieee802.org/15/pub/TG4.html>. [Último acceso: 30 Noviembre 2017].
- [11] «ZigBee Alliance,» [En línea]. Available: <http://www.zigbee.org>. [Último acceso: 16 Abril 2018].
- [12] D. I. Inc., «digi.com,» [En línea]. Available: <https://www.digi.com/products/xbee-rf-solutions/xctu-software/xctu>. [Último acceso: 15 Abril 2018].
- [13] ©. 2. D. I. Inc, «XBee® Zigbee® Mesh Kit Radio Frequency (RF) Module».
- [14] Python Software Foundation, «www.python.org/,» [En línea]. Available: <https://docs.python.org/2/index.html>. [Último acceso: 3 Mayo 2018].
- [15] © Copyright 2001-2017, Chris Liechti. Revisión 22d39007., «<http://pyserial.readthedocs.io/>,» [En línea]. Available: <http://pyserial.readthedocs.io/en/latest/pyserial.html#features>. [Último acceso: 27 Abril 2018].
- [16] J. Carreto, «<http://upromkteinv.blogspot.com/>,» [En línea]. Available: <http://upromkteinv.blogspot.com/2007/07/anlisis-del-mercado-de-consumidores.html>. [Último acceso: 2 Julio 2018].
- [17] G. d. Chile, «www.odepa.gob.cl/,» [En línea]. Available: <https://www.odepa.gob.cl/odepaweb/servicios-informacion/Instrumentos/fichas/f-31.html>. [Último acceso: 2 Julio 2018].
- [18] «Aula de la economica,» [En línea]. Available: <http://www.auladeeconomia.com/>. [Último acceso: 23 Junio 2018].
- [19] G. d. Chile, «Dirección de Presupuestos del Gobierno de Chile,» [En línea]. Available: <http://www.dipres.gob.cl/597/w3-multipropertyvalues-14572-24043.html>. [Último acceso: 13 Julio 2018].
- [20] G. B. Urbina, Fundamentos de la ingeniería económica, Mexico: Mc Granw Hill, 2007.

A Códigos de Nodos

En esta sección se presentarán todos los códigos utilizados en el dispositivo coordinador, emisor y receptor.

A.1 Código nodo coordinador

Código realizado en lenguaje de programación Python.

```
#librerias
import serial
import struct
import time

#Direcciones MAC de los transceptores emisores
fNrxl="0x0 0x13 0xA2 0x0 0x41 0x7B 0xD2 0x90" #direccion mac TX 1

fNrxl=" ".join(fNrxl.split())

Tx1=fNrxl.upper()

#tramas a enviar a dispositivos receptores
trama_1="7E001010010013A2004180BFCAFFFE0000313190"

#inicio del programa

ser= serial.Serial('/dev/ttyUSB0',9600)

Mac=[Tx1]

li=[trama_1]

def rx():

    leer_frame=ser.readline(18)
    if len(leer_frame)==18:
        data_struct=struct.Struct('=BBBBBBBBBBBBBBBBBBBB')
        frm=data_struct.unpack(leer_frame)

        if frm[0]==0x7E and frm[3]==0x90 and frm[15]==0x31 and
frm[16]==0x31:#frame de nodo transmisor
```

```

        str1=""
        for x in frm[4:12]:
            str1+=str(hex(x)).upper()

        if str1 in Mac:

            print "Mensaje recibido con la siguiente MAC
:"+str1

            Send_1(str1)
            time.sleep(2)
        else:
            print "La MAC no esta almacenada, se procede a
acoplar Nodo "
            print "fecha:",time.strftime("%d:%B:%Y"),"
Hora:",time.strftime("%H:%M:%S")
            agregar_mac(str1)

        elif frm[0]==0x7E and frm[3]==0x90 and frm[15]==0x31 and
frm[16]==0x30: #trama de centinela receptor

            str1=""
            for x in frm[8:12]:
                str1+=str(hex(x)).upper()+" "
            print "Nuevo dispositivo receptor"
            print "MAC",str1,"del receptor"
            generar_frame(str1)

    #funcion(generar_frame) para generar el frame del dispositivo
receptor nuevo
    #anido lo valores que siempre va a ocupar la trama, luego los
convierto en una lista a cada uno
    #los recorro por separado y los convierto a numero entero, los
sumo, luego quedo con los 8bits
    # de la suma total para poder sacar la diferencia de generar el
checksum    ff-valor de 8bits
def generar_frame(str1):

    frame_ck="10 01 00 13 A2 00 "+str1+"FF FE 00 00 31 3"
    frm_ck=frame_ck.replace('0X','') #elimino elemento 0X de str1
    frm_ck1=frm_ck.split()
    suma=0
    for n in frm_ck1:
        s=int(n,16)
        suma=suma+s
        b=hex(suma)
        c=str(b)
        c=c[3:]
        D=int(c,16)
        Checksum=255-D
        E=hex(Checksum)
        E=E[2:].upper()
        ch_sum=frm_ck+" "+E
        agregar_emisor(ch_sum)

def agregar_emisor(ch_sum):
    ch_sum="7E 00 10 "+ch_sum

```

```

ch_sum1="".join(ch_sum.split())
print "Frame generado con nuevo dispositivo Receptor"
print ch_sum1
li.append(ch_sum1)
print "Se agrego correctamente dispositivo receptor"
print "Lista de frames: "
print li, "\n"

def Send_1(str1):

    if str1 == Tx1:
        print "Mac pertenece a nodo 1, activar sector 1"
        str2=li[0]
        Tx(str2)
    elif str1==Mac[1]:
        print "MAC pertenece a nodo 2, activar sector 2"
        str2=li[1]
        Tx(str2)

    elif str1==Mac[2]:
        print "MAC pertenece a nodo 2, activar sector 2"
        str2=li[2]
        Tx(str2)

def Tx(str2):

    Tx_1=str2.decode("hex")
    ser.write(Tx_1)

    print "Paquete de activacion de zona enviado... Esperando
confirmacion de recibo"
    time.sleep(1)
    chequeo()

def chequeo(): #funcion para chequear el estado del envio de
mensaje
    leer_check=ser.readline(11)
    if len(leer_check)==11:
        data_struct=struct.Struct('=BBBBBBBBBBBB')
        frm_chequeo=data_struct.unpack(leer_check)
        if frm_chequeo[0]==0x7E and frm_chequeo[3]==0x8B and
frm_chequeo[8]==0x00:
            print "mensaje enviado con exito, riego activado"
            print " fecha:",time.strftime("%d:%B:%Y"),"
Hora:",time.strftime("%H:%M:%S")

        else:
            print "tengo problemas"

def agregar_mac(str1): # con esta funcion agrego direcciones MAC no
adjuntas

```

```

#provenientes de nodos transmisores
Mac.append(str1)
print "Se procede a agregar la siguiente MAC: "+str1
print "Lista de direcciones MAC: "
print Mac , "\n"
Send_1(str1)

def main():
    while True:
        rx()
        print "Esperando proxima comunicacion...."
        print ""

main()

```

A.2 Código nodo transmisor

Código para Arduino

```

#include <Printers.h>
#include<SoftwareSerial.h>
#include <OneWire.h>//Libreria Sensor de temperatura
#include <DallasTemperature.h> //Libreria conversos de temperatura
#define Temp 5 // Defino Sensor digital de sensor de temperatura

int temperatura;
int humedad=0;

OneWire ourWire (Temp);
DallasTemperature sensor(&ourWire); //
SoftwareSerial XBee(3,2);//cambio puerto serial XBee TX=3 y RX=2

void setup ()
{
    Serial.begin (9600);

}
void enviar(){// Funcion para el envio de la trama indicando que
debe activar el riego
    XBee.begin(9600);
    delay(3000);
    XBee.write(0x7E); //byte de inicio
    XBee.write((byte)0x0); //Longitud (MSB)
    XBee.write(0x10); //longitud (LSB)
    XBee.write(0x10); //Tipo de trama
    XBee.write(0x01); //Secuencia del mensaje

```

```

XBee.write((byte)0x0);
XBee.write(0x13);
XBee.write(0xA2);
XBee.write((byte)0x0);
XBee.write(0x41);
XBee.write(0x7C);
XBee.write(0xE4);
XBee.write(0x01);
// direccion destino
XBee.write(0xFF);
XBee.write(0xFE);//REservador
XBee.write((byte)0x0); // Maximo radio de envio
XBee.write((byte)0x0);//Ajuste de ACK (si es 0x01 se desabilita)
XBee.write(0x31);
XBee.write(0x31);// Datos RF
XBee.write(0x38); // Checksum
  delay(10000);
}
void loop()
{

  sensor.requestTemperatures(); //preparo el sensor de temperatura
  temperatura=sensor.getTempCByIndex(0);
  if((temperatura<28) &&(temperatura>10)){ //La temperatura tiene
que ser menor a 28° y mayor a 10°

  Serial.print("La temperatura es ");
  Serial.print(sensor.getTempCByIndex(0));
  Serial.println(" °C ");
  humedad= analogRead(A0);
  if(humedad>700){
    digitalWrite(7,HIGH);
    Serial.println( "Baja Humedad SI debe Regar");
    enviar();

  }
  else{digitalWrite(7,LOW);
  Serial.println( "Alta Humedad No debe Regar");
  delay(10000);
}
}
  else{
    digitalWrite(8,HIGH);
    Serial.print("La temperatura es ");
    Serial.print(sensor.getTempCByIndex(0));
    Serial.println(" °C ");
    delay(1000);

  }

}
}

```

A.3 Código nodo receptor

Código para Arduino.

```
#include <SoftwareSerial.h>
#define Partida 10
SoftwareSerial XBee(3,2);
int valor=0;

void setup ()
{
  XBee.begin(9600);
  delay(3000);
  XBee.write(0x7E); //byte de inicio
  XBee.write((byte)0x0); //Longitud (MSB)
  XBee.write(0x10); //longitud (LSB)
  XBee.write(0x10); //Tipo de trama
  XBee.write(0x01); //Secuencia del mensaje
  XBee.write((byte)0x0);
  XBee.write(0x13);
  XBee.write(0xA2);
  XBee.write((byte)0x0);
  XBee.write(0x41);
  XBee.write(0x7C);
  XBee.write(0xE4);
  XBee.write(0x01);
  // direccion destino
  XBee.write(0xFF);
  XBee.write(0xFE); //REservador
  XBee.write((byte)0x0); // Maximo radio de envio
  XBee.write((byte)0x0); //Ajuste de ACK (si es 0x01 se desabilita)
  XBee.write(0x31);
  XBee.write(0x30); // Datos RF
  XBee.write(0x39); // Checksum
  delay(10000);
  pinMode(11,OUTPUT); // agregado
  pinMode(Partida,OUTPUT);
  Serial.begin(9600);
  XBee.begin(9600);
}

void loop()
{
  if(XBee.available()>18){
    if(XBee.read()==(0x7E)// Comienzo a leer despues del primer
byte
    ){
      for(int i=0; i<14;i++){
        byte skip=XBee.read();//omito los bytes que no me sirven
hasta llegar al byte 0x31
      }
      valor=XBee.read();
      if(valor ==0x31){ // si lee byte 0x31, iniciar riego
        Serial.print("EL BYTE ES");
        Serial.print(valor,HEX);
      }
    }
  }
}
```

```
digitalWrite(Partida,HIGH);
digitalWrite(11,HIGH);//agregado
Serial.println(" Activo Riego");
digitalWrite(Partida,LOW);
digitalWrite(11,LOW);
}
else { // si no leo byte 0x31, no hago nada
Serial.print("EL BYTE ES ");
Serial.print(valor,HEX);
digitalWrite(Partida,LOW);
Serial.println(" No activo Riego");
digitalWrite(11,LOW);//agregado
}
}
}
```