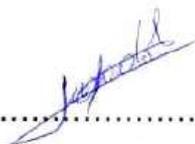


APROBACIÓN DEL TUTOR

En mi calidad de tutor de investigación sobre el tema: “RED INALÁMBRICA DE SENSORES (WSN) DE MONITOREO DE LA CALIDAD DEL AGUA PARA ESTANQUES DE TRUCHAS” de la señorita Paola Graciela Santos Benavides, estudiante de la Carrera de Ingeniería Electrónica y Comunicaciones de la Facultad de Tecnologías de la Información, Telecomunicaciones e Industrial de la Universidad Técnica de Ambato, considero que el informe de investigación reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato Junio, 2019

EL TUTOR



.....
Ing. Marco Jurado, Mg.

AUTORÍA

El presente trabajo de investigación titulado “RED INALÁMBRICA DE SENSORES (WSN) DE MONITOREO DE LA CALIDAD DEL AGUA PARA ESTANQUES DE TRUCHAS” es absolutamente original, auténtico y personal en tal virtud, el contenido, efectos legales y académicas que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, Junio 2019



.....
Paola Graciela Santos Benavides

C.C: 0503697823

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, Junio 2019



.....
Paola Graciela Santos Benavides

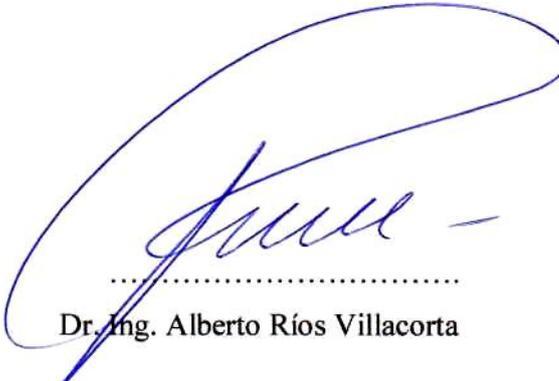
C.C: 0503697823

APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Mg. Geovanni Brito y Dr. Ing. Alberto Ríos Villacorta, revisó y aprobó el Informe Final del trabajo de graduación titulado “RED INALÁMBRICA DE SENSORES (WSN) DE MONITOREO DE LA CALIDAD DEL AGUA PARA ESTANQUES DE TRUCHAS” presentado por la señorita Paola Graciela Santos Benavides de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.


.....
Ing. Mg. Elsa Pilar Urrutia Urrutia
PRESIDENTA DEL TRIBUNAL


.....
Ing. Mg. Geovanni Brito


.....
Dr. Ing. Alberto Ríos Villacorta

DEDICATORIA

*A Dios y la Virgen María que cada día
me brindaron sus bendiciones cada día.
Mis padres Esperanza y Rodrigo por su infinito
apoyo, para que día a día cumpla
mis metas y objetivos.
Mi tía Soledad que ha sido como mi abuela,
mi segunda madre que siempre me ha cuidado,
inculcando principios y valores.
Querida hermanita que nunca te llegue a
conocer pero se que estas en el cielo
siempre cuidándome.
Mi familia que de una u otra manera siempre
formaron parte de mi vida.*

Paola Santos Benavides

AGRADECIMIENTO

*Mis padres que siempre me permitieron cumplir
mis sueños y metas.*

*Mis hermanos William, Mauricio y Juan Carlos,
quienes siempre han formado parte de mi vida
buenos y malos momentos.*

*La persona muy especial en mi vida Freddy Núñez
quien siempre me apoyó en toda mi carrera
en esos momento más difíciles de mi vida
me levantó.*

*Mis amigos quienes nos
apoyamos en toda la carrera universitaria.*

*A mi tutor, Ing. Marco Jurado quien siempre
con sus consejos y su apoyo académico.*

*A los propietarios del Criadero “El Chaparral” por
permitir realizar el proyecto de investigación en
sus instalaciones.*

Paola Santos Benavides

ÍNDICE GENERAL

APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR.....	iv
APROBACIÓN DE LA COMISIÓN CALIFICADORA	v
DEDICATORIA.....	vi
AGRADECIMIENTO	vii
RESUMEN.....	xv
ABSTRACT	xvi
GLOSARIO DE TÉRMINOS	xvii
INTRODUCCIÓN.....	xix
CAPÍTULO I.....	1
EL PROBLEMA.....	1
1.1. Tema de Investigación.....	1
1.2. Planteamiento del Problema.....	1
1.3. Delimitación:	3
1.3.1. Delimitación de contenidos:	3
1.3.2. Delimitación espacial:	3
1.3.3. Delimitación temporal:	3
1.4. Justificación.....	3
1.5. Objetivos	5
1.5.1. Objetivo General:	5
1.5.2. Objetivos Específicos:	5
CAPÍTULO II.....	6
MARCO TEÓRICO.....	6
2.1. Antecedentes Investigativos.....	6

2.2.	Fundamentación teórica	10
2.2.1.	Acuicultura	10
2.2.2.	Piscicultura	11
2.2.3.	Trucha Arco Iris	13
2.2.4.	Sistema de cultivo en estanques.....	17
2.2.5.	Calidad del agua.....	23
2.2.6.	Red de sensores inalámbricos (WSN).....	29
2.2.7.	Tecnologías y protocolos de comunicación para WSN.....	33
2.2.8.	Plataformas de Hardware Libre	34
2.2.9.	Computadores de Placas Únicas	37
2.2.10.	Sensores	38
2.2.11.	Base de datos (BD).....	40
2.2.12.	Protocolo para transferencia simple de correo.....	41
2.2.13.	Free - PDF.....	42
2.3.	Propuesta de Solución.....	42
CAPÍTULO III		43
METODOLOGIA.....		43
3.1.	Modalidad de la Investigación	43
3.2.	Recolección de Información.....	44
3.3.	Procesamiento y Análisis de Datos.....	44
3.4.	Desarrollo del Proyecto.....	44
CAPÍTULO IV		46
DESARROLLO DE LA PROPUESTA.....		46
4.1.	Introducción	46
4.2.	Factibilidad.....	47
4.2.1.	Factibilidad técnica	47
4.2.2.	Factibilidad económica.....	47

4.2.3.	Factibilidad bibliográfica.....	48
4.3.	Fundamentación científico-técnica.....	48
4.3.1.	Estado actual.....	48
4.3.2.	Requerimientos.....	51
4.4.	Diseño de circuitos.....	62
4.5.	Flujogramas de procesos.....	74
4.6.	Instalación del sistema operativo en la Raspberry Pi.....	76
4.7.	Instalación del servidor Apache.....	78
4.8.	Instalación de MySQL.....	78
4.9.	Instalación de PHP.....	79
4.10.	Instalación de PhpMyAdmin.....	80
4.11.	Base de datos en PhpMyAdmin.....	83
4.12.	Instalación de PHPMailer.....	86
4.13.	Diseño de la interfaz gráfica de usuario (GUI).....	87
4.14.	Implementación del prototipo.....	89
4.15.	Pruebas de los parámetros de la calidad del agua.....	93
4.16.	Pruebas de comunicación.....	103
4.17.	Prueba de envío de información a la base de datos.....	104
4.18.	Generación del reporte.....	105
4.19.	Presupuesto.....	107
CAPÍTULO V.....		111
CONCLUSIONES Y RECOMENDACIONES.....		111
5.1.	Conclusiones.....	111
5.2.	Recomendaciones.....	112
Bibliografía.....		114
ANEXOS.....		121

ÍNDICE DE FIGURAS

Fig. 2.1: Trucha Arco Iris.....	14
Fig. 2.2: Primera etapa de la Trucha Arco Iris	16
Fig. 2.3: Etapa – alevines	16
Fig. 2.4: Etapa – juveniles.....	17
Fig. 2.5: Etapa final.....	17
Fig. 2.6: Partes de un estanque piscícola (vista superior)	18
Fig. 2.7: Partes de un estanque piscícola (perfil longitudinal)	19
Fig. 2.8: Partes del sistema de abastecimiento de agua	19
Fig. 2.9: Dique del estanque piscícola	20
Fig. 2.10: Sección transversal de un dique.....	20
Fig. 2.11: Diseño del desagüe del agua.....	20
Fig. 2.12: Diseño de un estanque sumergido.....	21
Fig. 2.13: Diseño de un estanque de presa	22
Fig. 2.14: Diseño de un estanque de derivación.....	23
Fig. 2.15: Anatomía de la trucha Arco Iris.....	26
Fig. 2.16: Escala del nivel de pH.....	27
Fig. 2.17: Componentes de WSN Gateway y Nodos Distribuidos.....	29
Fig. 2.18: Topologías de WSN tipo estrella	31
Fig. 2.19: Topologías de WSN tipo árbol	31
Fig. 2.20: Topologías de WSN tipo malla.....	32
Fig. 2.21: Formato del paquete.....	34
Fig. 2.22: Placa Arduino Nano	36
Fig. 2.23: Placa Arduino Mega.....	37
Fig. 2.24: Ethernet Shield.....	37
Fig. 2.25: Raspberry Pi 2 modelo B+	38
Fig. 2.26: Sensor de temperatura.....	39
Fig. 2.27: Sensor de pH.....	39
Fig. 2.28: Sensor de turbidez.....	39
Fig. 2.29: Sensor de caudal basado en ondas ultrasónicas.....	40
Fig. 4.1: Ubicación del criadero “El Chaparral”	48
Fig. 4.2: Sistema de oxigenación en el criadero	49

Fig. 4.3: Cauce del río Yanayacu.....	49
Fig. 4.4: Desaguadero del criadero "El Chaparral"	50
Fig. 4.5: Canal de salida de agua del estanque	51
Fig. 4.6: Diagrama general del sistema de monitoreo de la calidad del agua.	52
Fig. 4.7: Esquema eléctrico del sensor de temperatura DS18B20.....	62
Fig. 4.8: Esquema eléctrico del sensor de pH SEN0161	63
Fig. 4.9: Gráfica de la ecuación de conversión entre ADC y pH medido.....	64
Fig. 4.10: Gráfica de la ecuación de conversión entre ADC y pH medido	65
Fig. 4.11: Esquema eléctrico del sensor de turbidez SEN0189.....	66
Fig. 4.12: Gráfica de la ecuación de conversión entre ADC y NTUs.....	66
Fig. 4.13: Gráfica de la ecuación de conversión entre ADC y NTUs.....	68
Fig. 4.14: Esquema eléctrico del sensor ultrasónico HC-SR04	68
Fig. 4.15: Esquema eléctrico del módulo de comunicación NRF 24L01	71
Fig. 4.16: Esquema electrico del primer nodo sensor.....	72
Fig. 4.17: Esquema eléctrico del segundo nodo sensor	73
Fig. 4.18: Diagrama de flujo de la red inalámbrica de los sensores	74
Fig. 4.19: Diagrama de flujo del prototipo.....	75
Fig. 4.20: Página de la Fundación Raspberry.....	76
Fig. 4.21: Inicio del sistema operativo	77
Fig. 4.22: Instalación del servidor Apache.....	78
Fig. 4.23: Instalación de MySQL	79
Fig. 4.24: Instalación del servidor Apache.....	80
Fig. 4.25: Instalación del servidor Apache.....	81
Fig. 4.26: Configuración de PhpMyAdmin.....	81
Fig. 4.27: Instalación de phpMyAdmin	82
Fig. 4.28: Base de datos en phpMyAdmin	83
Fig. 4.29: Base de datos “monitoreo”	83
Fig. 4.30: Tabla para el nodo sensor 1 “entrada”	84
Fig. 4.31: Columnas de la tabla “entrada”	84
Fig. 4.32: Base de datos “piscina1”	84
Fig. 4.33: Columnas de la tabla “piscina1”	85
Fig. 4.34: Base de datos “piscina2”	85
Fig. 4.35: Columnas de la tabla “piscina2”	85

Fig. 4.36: Repositorio de la librería PHPMailer	86
Fig. 4.37: Correo usando una cuenta Gmail	87
Fig. 4.38: Diagrama de operación de interfaz de usuario	88
Fig. 4.39: Interfaz del proyecto	88
Fig. 4.40: Interfaz del proyecto	89
Fig. 4.41: Circuito impreso del Nodo 1	89
Fig. 4.42: Circuito impreso del Nodo 2 y 3.....	90
Fig. 4.43: Soldadura de los elementos	90
Fig. 4.44: Conexión de los componentes electrónicos al nodo sensor	91
Fig. 4.45: Esquema eléctrico del sistema de monitoreo.....	92
Fig. 4.46: Analisis de valores de temperatura	94
Fig. 4.47: Analisis de valores de temperatura	95
Fig. 4.48: Análisis de los valores de pH	97
Fig. 4.49: Análisis de los valores de pH	99
Fig. 4.50: Análisis de los valores de pH	100
Fig. 4.51: Gráfica del comportamiento del sensor de turbidez.....	101
Fig. 4.52: Gráfica del comportamiento del sensor ultrasónico.....	102
Fig. 4.53: Ubicación de cada punto de comunicación	103
Fig. 4.54: Ruta de enlace de comunicación.....	103
Fig. 4.54: Perfil de elevación del terreno	104
Fig. 4.55: Prueba del envío de la información desde Arduino a la base de datos	105
Fig. 4.56: Reporte de la calidad del agua en archivo PDF.....	107

ÍNDICE DE TABLAS

Tabla 2.1: Principales características de los tipos de piscicultura	13
Tabla 2.2: Clasificación Taxómica de la trucha arco iris	14
Tabla 2.3: Propiedades nutricionales de la trucha Arco Iris	15
Tabla 2.4: Comportamiento de la trucha en función al nivel de la temperatura	24
Tabla 2.5: Comportamiento de la trucha en función al nivel de turbidez del agua. ..	26
Tabla 2.6: Comportamiento de la trucha en función al pH del agua	27
Tabla 2.7: Clasificación de las estructuras de medición	29
Tabla 4.1: Sensor DS18D20 vs PT-1000	53
Tabla 4.2: Sensor SEN-0161 vs Sensor E-201 de pH.....	54
Tabla 4.3: Sensor SEN0189 vs Sensor LTH serie T.....	55
Tabla 4.4: Sensor ultrasónico HC-SR04 vs Sensor K-0135	56
Tabla 4.5: Análisis de plataformas de hardware libre para la adquisición de datos...57	
Tabla 4.6: Análisis de los módulos de comunicación.....	58
Tabla 4.7: Análisis técnico de las computadoras de bolsillo.....	60
Tabla 4.8: Distribuciones GNU/Linux para arquitectura ARM	61
Tabla 4.9: Valores del ADC y de pH.....	65
Tabla 4.10: Valores del ADC y NTUs	67
Tabla 4.11: Valores de muestras de tiempo que recorre un objeto.....	69
Tabla 4.12: Análisis de valores de temperatura.....	93
Tabla 4.13: Promedio de valores de temperatura	95
Tabla 4.14: Analisis de valores de pH	96
Tabla 4.15: Análisis de valores de pH	98
Tabla 4.16: Análisis de valores de turbidez	99
Tabla 4.17: Análisis de valores de turbidez	100
Tabla 4.18: Análisis de valores de caudal.....	101
Tabla 4.19: Análisis de valores de turbidez	102
Tabla 4.20: Presupuesto del diseño del sistema	108
Tabla 4.21: Presupuesto para la implementación del sistema	109
Tabla 4.22: Presupuesto total del proyecto de investigación	110

RESUMEN

El presente proyecto propone un sistema electrónico para el monitoreo de la calidad del agua para estanques de truchas en el criadero “EL CHAPARRAL” de la ciudad de Salcedo mediante una WSN (Red Inalámbrica de Sensores) y un servidor web, con el objetivo de solucionar la problemática que existe en el criadero de peces, debido a que no existe un control de los parámetros de la calidad del agua como lo recomienda la FAO (Organización de las Naciones Unidas para la Alimentación y la Agricultura). Por esa razón se implementó un sistema electrónico para el monitoreo basado en una WSN para estanques de truchas que tiene el objetivo monitorear el nivel de temperatura, turbidez, pH y caudal. Estos parámetros son muy importantes en la calidad del agua para un crecimiento y desarrollo óptimo de las truchas.

El sistema de monitoreo consta de un nodo sensor inalámbrico en la piscina de alevines y otro en la piscina de juveniles, que permite la medición de los parámetros de la calidad del agua, estas variables son determinadas por sensores analógicos y digitales, además cuenta con un nodo sensor en el canal de derivación en el cual se ubica el sensor de caudal. A través de una topología punto-multipunto se envía los datos desde los nodos esclavos colocados en cada piscina hacia el nodo maestro, este los lleva a un servidor que esta compuesto de una Raspberry Pi mediante el modelo cliente – servidor para el monitoreo de los parámetros del criadero de truchas por medio de la utilización de software libre y tecnologías de bajo costo.

La base de datos se encuentra almacenada en la Raspberry Pi para que el encargado del criadero visualice los parámetros de la calidad del agua en la interfaz gráfica de usuario (GUI) en tiempo real, además generar un reporte en un archivo PDF del comportamiento de las variables monitorizadas de la calidad del agua, lo cual permitirá tomar decisiones precisas con el fin de obtener un producto de calidad. Finalmente, se genera una alerta vía correo electrónico cuando los parámetros detectados se encuentren fuera del rango normal permitiendo mejorar el hábitat del criadero.

ABSTRACT

The present project proposes an electronic system for the monitoring of water quality for trout ponds in the "EL CHAPARRAL" hatchery in the city of Salcedo through a WSN (Wireless Sensor Network) and a web server, with the objective of solving the problem that exists in the fish hatchery, because there is no control of water quality parameters as recommended by FAO (Food and Agriculture Organization of the United Nations). For this reason, an electronic system was implemented for the monitoring based on a WSN for trout ponds that has the objective of monitoring the temperature, turbidity, pH and flow level. These parameters are very important in water quality for optimal growth and development of trout. The monitoring system consists of a wireless sensor node in the fry pool and another in the juvenile pool, which allows the measurement of water quality parameters, these variables are determined by analog and digital sensors, and it also has a sensor node in the bypass channel in which the flow sensor is located. Through a point-multipoint topology the data is sent from the slave nodes placed in each pool to the master node, this takes them to a server that is composed of a Raspberry Pi through the client-server model for the monitoring of the parameters of the trout farm through the use of free software and low-cost technologies. The database is stored in the Raspberry Pi so that the keeper of the farm can visualize the parameters of the water quality in the graphical user interface (GUI) in real time, as well as generate a report in a PDF file of the behavior of the monitored variables of water quality, which will allow to make precise decisions in order to obtain a quality product. Finally, an alert is generated via email when the parameters detected are outside the normal range allowing to improve the habitat of the hatchery.

GLOSARIO DE TÉRMINOS

ADC: Analog to Digital Converter (Convertidor Analógico a Digital)

AIX: Advanced Interactive eXecutive

ARM: Arquitectura RISC de 32 bits para computadoras de bolsillo.

BD: Base de datos

CENIAC: Centro de Investigaciones Acuícolas

DINARA: Dirección Nacional de Recursos Acuáticos

FAO: Organización de las Naciones Unidas para la Alimentación y la Agricultura

FPDF: Free- Portable Document Format (Libre-Formato de Documento Portátil)

GFSK: Gaussian Frequency Shift Keying (Modulación por Desplazamiento de Frecuencia Gaussiana)

GIF: Graphics Interchange Format (Formato de Intercambio de Gráficos)

GMSK: Gaussian minimum shift keying (Modulación por Desplazamiento Mínimo Guassiano)

GND: Ground (Tierra)

GUI: Graphical User Interface (Interfaz Gráfica de Usuario)

HDMI: High Definition Multimedia Interface (Interfaz Multimedia de Alta Definición)

HMI: Human Machine Interface (Interfaz Humano Máquina)

HTML: HyperText Markup Language (Lenguaje de Marcas de Hipertexto)

IEEE: Institute of Electrical and Electronics Engineers (Instituto de Ingenieros Eléctricos y Electrónicos)

ICSP: In-Circuit Serial Programming (Programación serial en circuito de corriente alterna)

IoT: Internet of Things (Internet de las Cosas)

JPEG: Joint Photographic Experts Group (Grupo Conjunto de Expertos en Fotografía)

LAMP: Linux, Apache, MySQL y PHP

LXDE: Lightweight X11 Desktop Environment

MySQL: Sistema de gestión de bases de datos relacional desarrollado bajo licencia dual NodeMCU

NRF: Dispositivo basado en el chip NRF24 de la empresa Nordic que ofrecen una solución de bajo coste para aplicaciones de RF.

NTU: Unidad de Turbidez Nefelométrica

OD: Oxígeno Disuelto

OLTP: OnLine Transaction Processing (Procesamiento de Transacciones)

PHP: Hypertext Preprocessor (Preprocesador de Hipertexto)

pH: Pondus Hydrogenii (Potencial Hidrógeno)

PLC: Programmable Logic Controller (Controlador Lógico Programable)

PWM: Pulse-Width Modulation (Modulación Por Ancho De Pulsos)

RD: Radio Frecuencia

SD: Secure Digital

SGBD: Sistema Gestor de Base de Datos

SMP: Symmetric Multi-Processing (Multiprocesamiento Simétrico)

SMTP: Protocolo para transferencia simple de correo

SPI: Serial Peripheral Interface (Interfaz Periférica Serial)

TCP/IP: Transmission Control Protocol/Internet Protocol (Protocolo de Control de Transmisión/Protocolo de Internet)

TSS: Sólidos Suspendidos Totales

TTL: Transistor-Transistor Logic (Lógica Transistor A Transistor)

UART: Universal Asynchronous Reciver-Transmitter (Receptor-Transmisor Asíncrono Universal)

USB: Universal Serial Bus (Bus Universal en Serie)

VCC: Voltage Continuous Current (Voltaje de Corriente Continua)

WiFi: Wireless Fidelity (Fidelidad Inalámbrica)

WSN: Wireless Sensor Network (Redes De Sensores Inalámbricos)

INTRODUCCIÓN

El criadero el Chaparral ubicado en la provincia de Cotopaxi cantón San Miguel de Salcedo, barrio Yanayacu, esta zona se encuentra ubicada en la parte oriente a 12 Km de la ciudad de Salcedo a una altitud de 2480 m.s.n.m, de ambiente cálido la cual está destinada a la crianza de trucha y la conservación del medio ambiente, aprovechando las condiciones hídricas y climáticas del lugar para realizar la actividad piscícola.

El presente proyecto de investigación se conforma por sistema de monitoreo basado en una red inalámbrica de sensores en el criadero El Chaparral con el objetivo de monitorear los parámetros de la calidad del agua de un estanque de truchas, el cual la fuente de abastecimiento de agua es un punto muy significativo para el éxito de las operaciones piscícolas, que hoy en día el agua constituye un elemento fundamental para la producción de peces, es necesario tomar en consideración diversos aspectos biológicos en su ambiente natural, esto con lleva a la monitorización de la temperatura, turbidez, pH y caudal para evitar problemas en el crecimiento, desarrollo, la aparición de enfermedades y disminuir el índice de mortalidad.

El proyecto de investigación se encuentra estructurado de la siguiente manera:

En el Capítulo I, se detalla la problemática actual que existe en el criadero “EL CHAPARRAL” del cantón Salcedo en cual no existe un control de los parámetros de la calidad del agua para la crianza de truchas como lo recomienda la FAO, la justificación y los objetivos correspondientes para el desarrollo del proyecto.

En el Capítulo II, se analiza previamente de los antecedentes investigativos relacionados con la calidad del agua al igual que investigaciones sobre monitoreo basado en redes inalámbricas de sensores (WSN), en este capítulo se desarrolla el marco teórico basado en el tema de investigación.

En el Capítulo III, se especifica las diferentes técnicas de investigación que se utilizaron para la obtención de la información y se describe de manera general las etapas para el desarrollo del proyecto.

En el Capítulo IV, se describe el proceso mediante el cual se recolecto la información necesaria para el diseño y desarrollo del proyecto, describiendo paso a paso las etapas del sistema de monitorización de la calidad del agua para truchas.

En el Capítulo V contiene las conclusiones y recomendaciones obtenidas durante el desarrollo e implementación del prototipo bajo el tema Red Inalámbrica de Sensores (WSN) de monitoreo de la calidad del agua para estanques de truchas.

CAPÍTULO I

EL PROBLEMA

1.1. Tema de Investigación

RED INALÁMBRICA DE SENSORES (WSN) DE MONITOREO DE LA CALIDAD DEL AGUA PARA ESTANQUES DE TRUCHAS.

1.2. Planteamiento del Problema

La Organización de las Naciones Unidas para la Alimentación y la Agricultura (FAO), define como un aspecto muy importante para el criadero de truchas, el agua por eso; esta debe cumplir condiciones de calidad como: el nivel de temperatura, pH y turbidez. Este organismo señala que para determinar la calidad del agua se debe tomar en cuenta la temperatura del agua, parámetro que regula el crecimiento de los peces, ya que estos no tienen capacidad propia para regular su temperatura corporal. Si los niveles de temperatura están por debajo de los recomendados el crecimiento del pez es lento, a temperaturas más altas el desarrollo es más rápido pero dificulta el engorde. El nivel de turbidez puede impactar a los ecosistemas acuáticos, afectando la fotosíntesis, respiración y la reproducción de la vida acuática. Finalmente el nivel de pH ayuda a la prevención de erosiones en la piel. Mientras tanto en lo que respecta al caudal del agua que se encuentra en el canal de derivación, el mismo que dirige hacia las piscinas del criadero de truchas, debe ser medido para conocer el caudal óptimo y disponible porque en temporadas de invierno incrementa la creciente del río Yanayacu, incluso cuando abren las compuertas en la Central Pucara – Pisayambo el nivel del agua incrementa del río ocasionando inundaciones a las piscinas del criadero. [1]

Ecuador está progresando en el sector acuícola y las principales dificultades son la falta de capacitación, escaso financiamiento, escasa asistencia técnica y poco control en la calidad del agua. Debido a la geografía del país, este cuenta con un gran potencial acuícola para la cría y cultivo de especies bióticas durante todo el año gracias a ciertas condiciones favorables que no existen en países como Argentina, Chile, Paraguay y Uruguay debido a los cambios climáticos marcado por las cuatro estaciones del año. El Ministerio de Agricultura, Ganadería, Acuicultura y Pesca tiene 47 técnicos de la Subsecretaría de Acuicultura a nivel nacional, pero la asistencia técnica no es suficiente para tener un control de la calidad y cantidad del agua periódicamente en cada criadero. Eso hace que Cotopaxi no tenga un crecimiento económico en el sector piscícola como Azuay que produce 190 *ton/año* de trucha. [2], [3]

El Centro de Investigaciones Acuícolas (CENIAC), realizó el Primer Censo Piscícola de Producción de Trucha en las Zonas Norte, Centro y Sur de la Región Interandina, en el cual se determina que Cotopaxi es la segunda provincia con menos criaderos a nivel nacional, esto se debe a que existe 1 técnico y 9 operarios en los 6 cantones de la provincia. El volumen de producción mínimo corresponde a 1.152 m^3 y una producción de 22.8 *ton/año* de trucha a nivel nacional del Ecuador. El criadero “El Chaparral” del cantón Salcedo realiza el control del agua que ingresa a los estanques de las truchas en forma manual sin tomar en cuenta los principales parámetros del agua que ayudan al desarrollo del pez además de no contar con ningún registro de calidad del agua haciendo que el índice de mortalidad sea muy alto (entre 3 a 15 mensuales) provocando en muchos de los casos que el criadero no recupere la inversión inicial y que el producto final no satisfaga las expectativas del cliente. Además no cuentan con operarios quienes ayuden a tomar muestras de la calidad del agua diariamente, tampoco con registros de estanques de truchas que recomienda la FAO, causando pérdidas económicas y afectando la imagen del criadero, con lo cual no se contribuye en el aspecto socio-económico de la provincia de Cotopaxi. [2]

1.3. Delimitación:

1.3.1. Delimitación de contenidos:

Área académica:	Comunicaciones
Línea de investigación:	Tecnologías de Comunicación
Sublínea de investigación:	Comunicaciones Inalámbricas

1.3.2. Delimitación espacial:

La investigación se realizó en la Ciudad de San Miguel de Salcedo, Barrio Yanayacu, en el criadero “El Chaparral”.

1.3.3. Delimitación temporal:

La presente investigación se desarrolló en el periodo Septiembre 2018 – Febrero 2019 de acuerdo a lo establecido en el Reglamento de Graduación para Obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

1.4. Justificación

La Acuicultura aporta grandes posibilidades de mejoramiento de la calidad de la alimentación de los seres humanos, la carne de pescado es una fuente de proteína de alta calidad con un 30% menos de grasa con respecto a las carnes rojas. Además aporta al aumento de la productividad agropecuaria, rentabilidad, generación de oportunidades de empleo y desarrollo de la comunidad. La fuente de agua es un factor importante para esta actividad, tanto en cantidad como calidad que debe estar libre de pesticidas y tóxicos.

Al no disponer de un sistema de monitorización en el criadero “El Chaparral” se realiza una medición manual de la temperatura del agua que regula el crecimiento de las truchas, su rango es de 13 a 18°C, pero su valor óptimo debe ser alrededor de 15°C. La trucha requiere de una fuente de agua cristalina y pura, por ello es necesario el control del nivel de turbidez para evitar la disminución del oxígeno que puede afectar en la respiración. El control del nivel de pH en el hábitat ayuda a la prevención de enfermedades en la piel y

evitar contagiar a la camada que debe cumplir con el nivel óptimo de 7 que se le conoce como el “punto neutro”, si este es menor el agua es ácida y si este supera al nivel óptimo se considera como alcalino. La cantidad del agua es de mucha importancia debido a que está relacionado directamente con el nivel de oxígeno que necesitan los peces. Esto va a ayudar en la producción piscícola, contribuyendo a los conciudadanos a mejorar el proyecto del Buen Vivir que es ofrecer un producto para “Mejorar la calidad de vida de la población”. Además, con un nivel óptimo de caudal se evita inundaciones en los estanques y pérdidas económicas para el criadero. [1], [4], [5]

El sistema de monitorización propuesto permite al encargado del criadero disponer de un reporte diario, semanal, mensual o anual del comportamiento de los parámetros que intervienen en el estanque de las truchas. Para la obtención de la buena calidad de peces, garantizando los derechos de la naturaleza y promover la sostenibilidad ambiental territorial y global, además asegurar la soberanía y eficiencia de los sectores estratégicos para la transformación industrial y tecnológica. [4]

La implementación del sistema propuesto permite el correcto monitoreo del agua de los estanques de truchas con el objetivo de garantizar que el producto posea las condiciones de estadía adecuadas, durante el tiempo que dure su proceso de inicio, levante y engorde se tendrá como beneficio entregar una herramienta tecnológica que facilite el monitoreo de esta especie de pez para así poder obtener una información detallada en lo que se refiere a la calidad del agua. Disminuyendo procesos repetitivos de control, que tiene costos adicionales que se pueden evitar. Se obtiene una reducción en los niveles de infecciones en los peces adquiridas por distintas vías, además previene una transmisión de dichas infecciones y problemas hacia el ser humano haciendo más higiénico y optimizado el proceso de crianza.

En la propuesta los beneficiarios directos son los dueños criadero “El Chaparral” porque se realiza la monitorización diaria de la temperatura, nivel de turbidez, nivel de pH del agua y nivel de caudal, ahorrando el tiempo y dinero. Los beneficiarios indirectos son los consumidores quienes adquieren un producto de buena calidad.

El proyecto de investigación fue factible debido al aporte de la Universidad Técnica de Ambato, a través de los laboratorios donde se realizó las pruebas de comunicación de la red de sensores, el acoplamiento de señales y el diseño de la electrónica del sistema; el criadero “El Chaparral” está dispuesto a colaborar para tomar muestras y realizar las pruebas del prototipo. Este sistema de monitoreo se implementó con tecnologías inalámbricas de bajo costo, equipos y materiales de fácil adquisición, plataformas de hardware libre, protocolos de comunicación, además se utilizó distribuciones de código abierto en el servidor y hardware libre en los nodos de la red para obtener un prototipo de bajo costo.

Se utilizó un sistema microcontrolado en cada uno de los nodos sensores, el cual permite adquirir las variables analógicas del agua utilizando sensores, las señales son digitalizadas y a través de un protocolo de comunicación inalámbrica se traslada los datos hacia un servidor formado por una base de datos y un gestor de páginas web encargado de almacenar, gestionar, presentar e informar al encargado el estado de la calidad del agua de los estanques de truchas, a través de una interfaz web en el sitio y vía correo electrónico remotamente previa solicitud del encargado.

1.5. Objetivos

1.5.1. Objetivo General:

- Implementar una red inalámbrica de sensores (WSN) de monitoreo de la calidad del agua para estanques de truchas.

1.5.2. Objetivos Específicos:

- Analizar las características de la crianza de truchas en el criadero el “El Chaparral”
- Determinar las variables a monitorear en el agua del estanque de truchas.
- Diseñar un sistema de monitoreo de la calidad del agua para estanques de truchas.

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes Investigativos

Una vez realizada la investigación en los repositorios digitales, se han encontrado algunos trabajos similares al trabajo investigativo planteado.

En 2012, China, en el Paper desarrollado por los autores Dong He, Li-Xin Zhang, bajo el tema “El sistema de monitoreo de la calidad del agua basado en WSN”, realizaron un sistema para un departamento de protección ambiental en un área particular donde los requisitos de la calidad del agua son indispensables. El sistema de monitoreo consiste en una red de monitoreo inalámbrico de la calidad del agua y el centro de datos remoto. La plataforma de hardware utiliza un microprocesador CC2430 como núcleo del nodo. La red del sensor está construida de acuerdo con la transmisión inalámbrica ZigBee. WSN prueba la calidad del agua y envía los datos a Internet con la ayuda de la GPRS DTU que tiene un protocolo TCP / IP incorporado. A través de Internet, Remote Data Center obtiene los datos de calidad del agua en tiempo real, y luego analiza, procesa y registra los datos. El departamento de protección ambiental puede proporcionar información en tiempo real a aquellas industrias que dependen de las condiciones regionales de calidad del agua, como industrial, de plantas y acuicultura. Lo más importante es que el trabajo puede ser más eficiente y menos costoso. [6]

En 2013, Ecuador, en el proyecto realizado por Gallardo Cristian, bajo el tema “Implementación de un sistema para el control y monitoreo de niveles de agua, temperatura y oxigenación de un proyecto piscícola en la crianza y reproducción de truchas en la Finca San Nicolás”, se implementó un sistema eficiente y controlado en la reproducción de truchas en su etapa de fecundación, eclosión y alevinaje, en este sistema se puede visualizar y monitorear el estado real de las variables que están involucradas en el proceso como la temperatura el nivel del agua y también saber el estado de nuestros actuadores en el sistema como bombas oxigenadores, aireadores, mediante de un HMI y a su vez acondicionadas con plataformas de interfaces de comunicación PC Access y Lab VIEW. [7]

En 2014, Perú, en el Paper desarrollado por los autores Milton Ríos, Ricardo Yauri, John K. Rojas y Viky C. Camarena, bajo el tema “Red de sensores inalámbricos para la medición de parámetros de calidad del agua usada en la crianza de peces amazónicos”, proponen sistema para la adquisición, envío y monitoreo de los parámetros de calidad del agua en estanques de crianza de peces amazónicos. La red se implementa bajo el estándar IEEE 802.15.4 y los datos se envían a partir de nodos con sensores de temperatura, oxígeno disuelto, conductividad y pH (potencial de hidrógeno). El control de la adquisición de lecturas obtenidas desde cada nodo sensor se realiza a través de un sistema embebido que está implementado en un microcontrolador de bajo consumo y un dispositivo de transmisión inalámbrica para enviar los datos de los sensores. En la mejora del rendimiento de la batería que energiza el nodo sensor se utiliza un mecanismo de control de la energía para la operación del nodo sensor. Además aprovechando la energía solar en los ambientes de crianza se utilizan celdas solares para la carga de la batería. La configuración de los nodos sensores se realiza de forma inalámbrica usando el estándar Bluetooth mediante una aplicación en un dispositivo móvil. [8]

En 2014, China, en el Artículo Científico realizado por los autores Yanhua Chen y Guoping Hou, bajo el tema “Sistema de monitoreo basado en WSN para la industria acuícola”, proponen un sistema para monitorear los parámetros de la calidad del agua en términos de temperatura del agua, valor de pH y concentración de oxígeno disuelto. Este

sistema consta de una red de sensores inalámbricos y una computadora superior. La solución de este sistema está diseñada en utilizar la red de sensores inalámbricos y las tecnologías ZigBee. Las arquitecturas de hardware de los nodos y los circuitos relacionados están diseñados de acuerdo con los principios de diseño del hardware. Se realizaron experimentos para probar el rendimiento y la función del sistema propuesto, y los resultados experimentales muestran que el sistema de monitoreo propuesto funciona correctamente para monitorear los parámetros de la calidad del agua. [9]

En 2016, Indonesia, en el Artículo científico desarrollado por los autores Taufik Ibnu Salim, Triya Haiyunnisa y Hilman Syaeful Alam, bajo el tema “Diseño e implementación del monitoreo de la calidad del agua para la acuicultura de anguilas”, realizaron el sistema mediante un computador de arquitectura ARM 32 bits. Este dispositivo monitorea la calidad del agua de la acuicultura. Algunos parámetros de calidad del agua que se utilizan en este monitoreo son oxígeno disuelto (OD), acidez (pH) y temperatura. Las pruebas se llevan a cabo en dos etapas, son sensores que leen la estabilidad y el rendimiento del monitoreo del dispositivo. La estabilidad del sensor de prueba se desarrolló en el laboratorio y la monitorización del dispositivo de prueba se llevó a cabo en los acuarios utilizando la aireación de microburbujas. Como resultado de este proyecto se obtuvo un dispositivo que se puede usar para monitorear la calidad del agua en el laboratorio, acuarios y la acuicultura de anguilas. Todas las características de este dispositivo de monitoreo pueden funcionar de manera adecuada y sin problemas. [10]

En 2016, Japón, en el Paper desarrollado por los autores Tomoaki Kageyama, Masashi Miura, Akihiro Maeda, Akihiro Mori y Sang-Seok, bajo el tema “Una plataforma de red de sensores inalámbricos para el control de la calidad del agua”, proponen un sistema que consiste en varios sensores para la detección de propiedades físicas y químicas del agua, y sistemas que incluyen el módulo de comunicación inalámbrica, el módulo de potencia y la interfaz entre los sensores y el módulo de comunicación inalámbrica. Como resultado de este proyecto se ha obtenido la fabricación de un nodo y la adquisición de datos medidos en tiempo real que son transmitidos desde el nodo del sensor y confirmar que la plataforma propuesta fue operada con éxito en un entorno real. Significa que la plataforma

WSN utilizada en el sistema propuesto es flexible, útil para la creación rápida de prototipos de un nuevo sistema WSN y una solución de bajo costo. Además, la plataforma WSN se puede aplicar no solo a la supervisión de la calidad del agua sino también a otras áreas que utilizan el sistema WSN. [11]

En 2017, México, en el Artículo Científico realizado por los autores Cesar Encinas, Erica Ruiz, Joaquin Cortez y Adolfo Espinoza, con el tema “Diseño e implementación de un sistema distribuido IoT para el monitoreo de la calidad del agua en la acuicultura”, desarrollaron un prototipo y un sistema de monitoreo distribuido con las variables más importantes dentro de la calidad del agua de la acuicultura. Esto es de gran importancia porque la acuicultura es un área de tecnología rezagada en comparación con otras áreas como la agricultura. Por lo tanto, es importante resolver los problemas que se encuentran en esta área con el apoyo de la tecnología. Entre los problemas se encuentra el lento tiempo de respuesta en el cuidado de la calidad del agua, el desperdicio de recursos y pérdidas. El sistema propuesto en este trabajo monitorea la calidad del agua en base a las redes inalámbricas de sensores y Internet of Things (IoT). Esta información es importante para el desarrollo de esta área, ya que permite compartir las diferentes condiciones en la cría de organismos acuáticos entre diferentes criadores y organizaciones. Como resultado de esta investigación se desarrolló un sistema de bajo costo, bajo consumo de energía, escalable, versátil, distribuido, móvil y preciso. [12]

En 2017, Ecuador, en el Proyecto de Investigación realizado por Freddy Núñez, con el tema “Sistema de monitoreo y control del alumbrado público mediante una red de sensores RF”, implementó un prototipo de una WSN para el monitoreo del servicio de alumbrado público, donde se realiza una predicción mediante el modelo empírico Log-Normal del alcance máximo de los dispositivos NRF24101 PA LNA para un BER menor al 10% determina un alcance máximo de 550 metros a una velocidad de transmisión de 250 kbps. Este dispositivo según el fabricante posee un alcance de un kilómetro para velocidades de transmisión de 250 kbps. [13]

2.2. Fundamentación teórica

2.2.1. Acuicultura

La acuicultura abarca el cultivo de organismos acuáticos que incluye peces, moluscos, crustáceos y plantas acuáticas en áreas continentales o costeras, que implica por un lado la intervención en el proceso de crianza para mejorar la producción y por el otro la propiedad individual o empresarial del stock cultivado. [14]

En forma global el término acuicultura reúne a todas aquellas acciones que tienen por objeto la producción, el crecimiento y comercialización de organismos acuáticos animales o vegetales de aguas dulces, salobres o saladas. Implica el control de las diferentes etapas de desarrollo hasta la cosecha, proporcionando a los organismos los medios adecuados para su crecimiento y engorde. [15]

Ventajas de la acuicultura como actividad productiva

La acuicultura brinda grandes posibilidades de explotación debido a la existencia en todo el mundo de cientos de millones de hectáreas de aguas aptas para su desarrollo, detallando a continuación las siguientes ventajas:

a) Mejoramiento de la calidad de la alimentación de los seres humanos:

El mejoramiento de la calidad de la alimentación de los seres humanos depende de la fuente de proteínas que consuman, como es el pescado que contiene altas albúminas similares al pollo y superior a la de la carne roja. Las proteínas del pescado contienen todos los aminoácidos esenciales y al igual que las proteínas de la leche, los huevos y la carne de mamíferos tienen un valor biológico muy alto que ayudan a fortalecer el organismo del ser humano. El pescado eviscerado contiene cerca de un 30% menos de grasa que las carnes rojas. Estudios científicos han demostrado que las dietas a base de pescado en los niños ayuda a buen desarrollo y crecimiento , además de fortalecer su sistema inmunológico, en caso de los adultos ayuda a prevenir y disminuir los niveles de colesterol en la sangre, los casos de anemia y enfermedades coronarias. [15]

b) Integración y aumento de la productividad agropecuaria:

La integración y aumento de la productividad agropecuaria se refiere a reutilizar el agua de los estanques para limpieza de corrales así como para el riego de plantaciones, las que a su vez podrán ser utilizadas como alimento de los animales de granja. Por otra parte el limo acumulado en el fondo de los estanques constituye un excelente abono para la huerta. [15]

c) Aumento de la rentabilidad:

El aumento de la rentabilidad se refiere al bajo consumo de energía que se necesita para el mantenimiento de sus funciones vitales (respiración, locomoción y excreción entre otras) lo que permite destinar un mayor porcentaje del alimento consumido al crecimiento, significando una mayor producción de carne. [15]

d) Generación de oportunidades de empleo:

Generación de oportunidades de empleo en las pequeñas unidades acuícolas productivas como los emprendimientos mayores requieren de actividades que pueden involucrar la participación del núcleo familiar, o bien la incorporación de personal externo para todas o cada una de las etapas del proceso productivo. [15]

e) Desarrollo de la comunidad:

El desarrollo de la comunidad, se forja un escenario que mejora la calidad de vida de las personas. Cualquier emprendimiento exige contar con determinados servicios, por lo tanto la actividad generará progresos en la zona (luz eléctrica, alcantarillado, etc.) [15]

2.2.2. Piscicultura

La piscicultura es una técnica que se ocupa de la crianza y multiplicación de los peces, en plena libertad o en depósitos de agua cerradas, ya que estos se pueden convertir en estanques naturales o artificiales. [16]

Es la actividad más significativa dentro del volumen de la producción acuícola. Según la Dirección Nacional de Recursos Acuáticos – DINARA la piscicultura alcanza su auge a

partir de las décadas de 1960 y 1970, años en los cuales se obtuvo un gran desarrollo científico y tecnológico. [15]

Ventajas de la piscicultura:

Las ventajas de la piscicultura se describen a continuación:

- El costo de los peces se reduce, debido a que resulta más costoso llegar a los ríos para capturarlos, conservarlos y llevarlos al mercado.
- Los estanques pueden ser construidos en terrenos que no son aptos para la agricultura o ganadería, siempre que exista un suministro de agua.
- Permite el control casi absoluto de las condiciones de vida del pez y su pesca, lo que hace que se pueda incrementar la calidad.
- Se puede evitar la existencia de depredadores, combatir a los parásitos y limpiar con facilidad los estanques, lo cual permite tener peces de buena calidad para consumo humano. [17]

Clasificación de la piscicultura:

La piscicultura se clasifica según la administración o las actividades del piscicultor, de acuerdo a la tabla 2.1

Tabla 2.1: Principales características de los tipos de piscicultura, [18]

Tipo de piscicultura Características	Piscicultura extensiva	Piscicultura intensiva	Piscicultura super-intensiva	
			Sistema japonés (TANAKA)	Jaulas
Lugar	Embalses Lagos pequeños	Lagunas piscícolas	Lagunas artificiales con agua suficiente desviada del arroyo	En jaulas fijadas en ríos, canales o embalses.
Numero de siembra de peces	100 – 800 / ha	1200 – 5000 / ha	10 -50 / m ²	50 – 100 / m ²
Abono y fertilización	Ningún tipo	A menudo	Ningún tipo	Ningún tipo
Alimentación	Nada Muy poca	Diariamente	Solamente artificial completa (proteínas + minerales + vitaminas)	Solamente artificial completa o el alimento que entra en la jaula.
Cosecha	50 – 500 kg/ha	1000/2500 kg/ha	8 – 40 kg / m ²	40– 80 kg / m ²
Trabajo en general	Poco	Diariamente	Diariamente	Diariamente

2.2.3. Trucha Arco Iris

La trucha arco iris conocida como el nombre científico *Oncorhynchus mykiss*, perteneciente a la familia Salmonidae, de acuerdo a la clasificación Taxómica de la tabla 2.2. [19].

Tabla 2.2: Clasificación Taxómica de la trucha arco iris, [19]

Reino	Animal
Phylum	Chordata
Subphylum	Vertebrata
Superclase	Pisces
Clase	Osteichthyes
Subclase	Actinopterygii
Orden	Salmoniformes
Familia	Salmonidae
Genero	Oncorhynchus
Especie	Mykiss
Nombre científico	Oncorhynchus mykiss
Nombre común	Trucha arco iris

Este pez recibe la denominación de “Arco Iris” por los puntos multicolor de su piel como se observa en la figura 2.1, es una de las principales especies de cría en agua dulce. [20]



Fig. 2.1: Trucha Arco Iris, [20]

La trucha arco iris es una especie resistente que tolera un amplio abanico de entornos y manipulación. Este pez puede ocupar diversos hábitats, pasando de agua dulce a salada y viceversa, o permanecer en lagos. El crecimiento y la maduración dependen de la temperatura del agua y del alimento. [19]

Propiedades del consumo de la trucha arco iris

La trucha es un pescado semigraso que aporta proteínas de alto valor biológico, de manera que contiene los aminoácidos esenciales para el cuerpo del ser humano, como se muestra en la tabla 2.3. [21] , [22]

Tabla 2.3: Propiedades nutricionales de la trucha Arco Iris [21] , [22]

Propiedades	Características	Porcentaje / 100 g
Proteínas	Son aminoácidos esenciales para el desarrollo de músculos del cuerpo del ser humano.	15 g – 21 g
Calorías	Son la fuente principal de energía para el organismo de los seres vivos.	90 kcal – 133 kcal
Cenizas	Se refiere a los minerales presentes en los alimentos.	1,2 g – 1,25 g
Grasas	Son vitales como fuente de energía y protegen los órganos como el corazón y los riñones.	2,70 g - 3,1 g

Beneficios de la carne de trucha

Los beneficios para el ser humano de consumir la carne de trucha son:

- Tiene alto contenido en ácidos grasos OMEGA 3, los cuales ayudan a prevenir enfermedades cardiovasculares.
- Debido al aporte en grasas es bajo ayuda a disminuir el sobrepeso y la obesidad, manteniendo una dieta balanceada.
- La concentración de vitaminas contribuye al desarrollo de huesos y dientes. [21]

Etapas de desarrollo de las truchas

El desarrollo de las truchas arco iris se describe en 4 etapas:

1. Ova:

La primera etapa denominada Ova son los huevos fecundados que entran en estado de incubación y luego eclosionan para convertirse en larvas, como se observa en la figura 2.2.



Fig. 2.2: Primera etapa de la Trucha Arco Iris, [20]

2. Alevines:

Los alevines son peces que aproximadamente en el rango desde 3 cm a 14 cm como se observa en la figura 2.3, que están listos para su distribución, es una de las etapas más importantes porque su velocidad desarrollo depende de ésta.



Fig. 2.3: Etapa – alevines, [14]

3. Juveniles:

Los juveniles se encuentran en el cuarto mes desde su eclosión, a partir de esta etapa los peces son más resistentes a enfermedades lo que disminuye su mortalidad, como se observa en la figura 2.4.



Fig. 2.4: Etapa – juveniles, [14]

4. Trucha de engorde:

En la etapa de trucha de engorde, estas se encuentran listas para su distribución y comercialización, como se observa en la figura 2.5.



Fig. 2.5: Etapa final, [14]

2.2.4. Sistema de cultivo en estanques

El sistema de cultivo en estanques se basa en la explotación de estanques de agua dulce que mantienen e intercambian el líquido y producción de peces para ser comercializados. El diseño y construcción adecuada de estos estanques son parte esencial para la reproducción de estos peces. [23]

Estanques

Los estanques son estructuras diseñadas y construidas bajo especificaciones que permiten el cultivo eficiente de los peces, además depende de la topografía, calidad del terreno, la cantidad de agua por manejar y según estos parámetros se determina las dimensiones de los estanques. Son empleados con mucha frecuencia en donde se tiene cerca ríos, lagos, embalses y represas que tienen buenas condiciones para el cultivo de la trucha. En las figuras 2.6 y 2.7 se muestran las partes de un estanque piscícola.

Características:

Las características principales para la instalación de los estanques son:

- Paredes del estanque o dique que retienen el agua.
- Tuberías o canales permiten que el agua entre y salga.
- Dispositivos de regulación del agua que controlan el nivel o caudal del agua.
- Presencia de olas débiles que favorezcan la oxigenación del agua.
- Profundidad de los estanques es de 3.5 a 4 metros aproximadamente.
- Ausencia de plantas acuáticas y algas en el sitio de los estanques.
- Cuerpo de agua no contaminado. [1], [23]

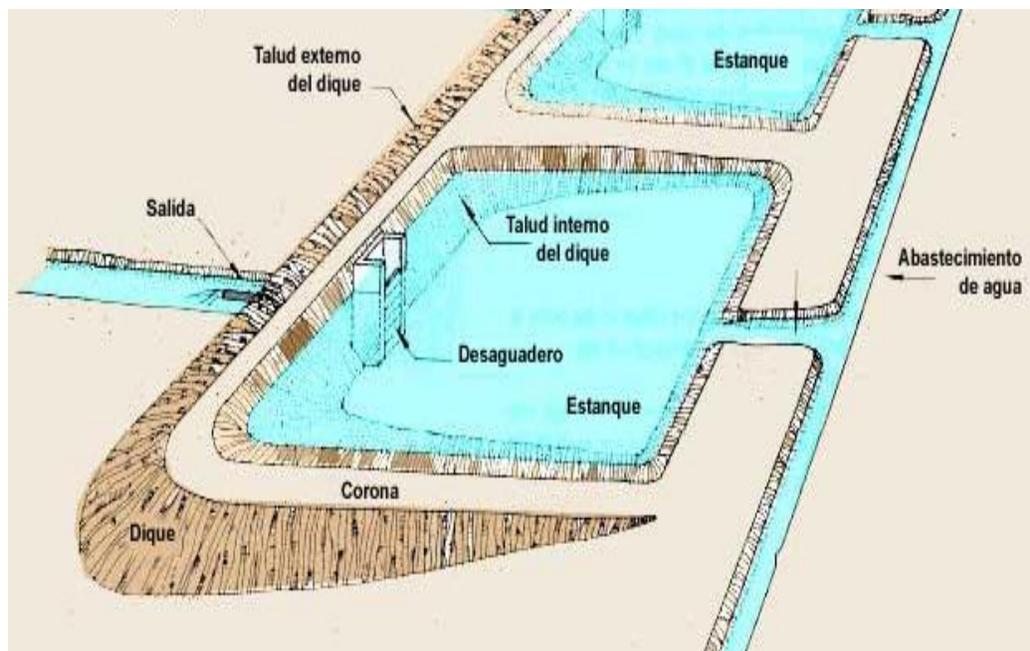


Fig. 2.6: Partes de un estanque piscícola (vista superior), [23]

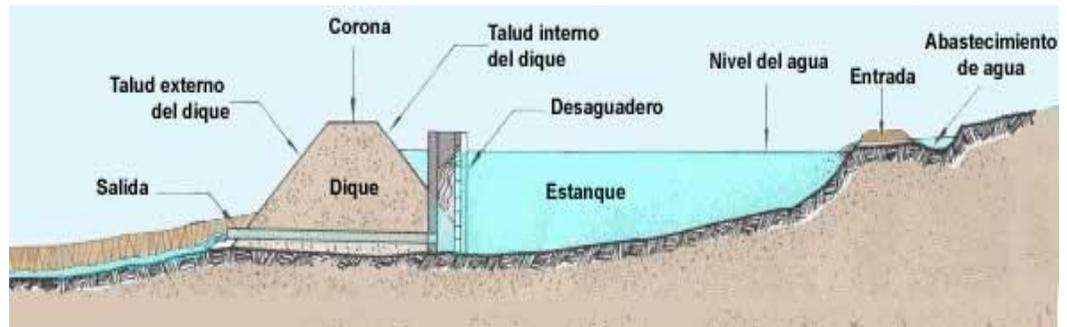


Fig. 2.7: Partes de un estanque piscícola (perfil longitudinal), [23]

Las partes de un estanque piscícola se detallan a continuación:

Sistema de abastecimiento de agua

El sistema de abastecimiento de agua es el conjunto de tuberías, instalaciones y accesorios destinados para conducir el líquido desde un arroyo o río hacia un estanque. En el sistema de abastecimiento de agua consiste en tres componentes, como se observa en la figura 2.8:

- Bocatoma
- Canal de derivación
- Ingreso de agua al estanque

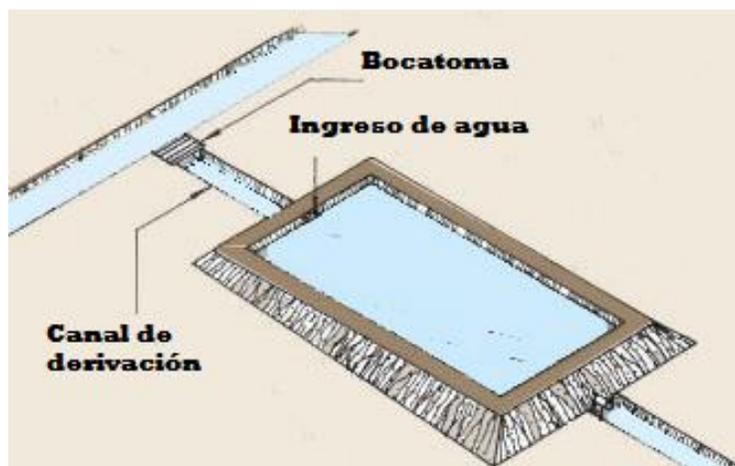


Fig. 2.8: Partes del sistema de abastecimiento de agua, [23]

Dique

El dique es una pared trapezoidal que se construye para cercar el agua del estanque de los peces como se observa en la figura 2.8.

Las partes de un dique son tres, que se puede apreciar en la figura 2.9 y 2.10:

- Ancho de la cima o corona
- Talud externo e interno
- Ancho de la base

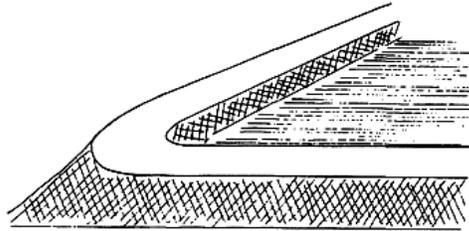


Fig. 2.9: Dique del estanque piscícola, [18]

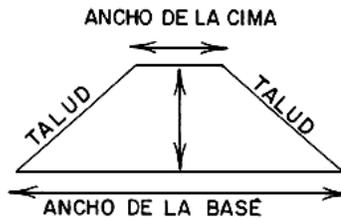


Fig. 2.10: Sección transversal de un dique, [18]

Canal de desagüe

El canal de desagüe o salida del agua de un estanque permite la eliminación de agua excedentes o simplemente la circulación del agua, como se observa en la figura 2.11

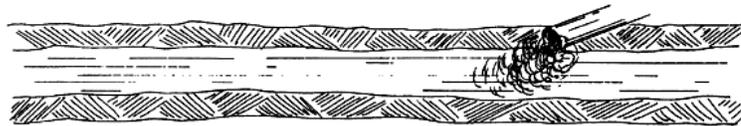


Fig. 2.11: Diseño del desagüe del agua, [18]

Clasificación de los estanques

La clasificación de los estanques según la fuente de abastecimiento de agua, se detalla de la siguiente manera:

a) Estanque sumergido

El estanque sumergido se alimenta directamente de agua subterránea o precipitaciones, que esta generalmente por debajo del nivel de la tierra, como se observa en la figura 2.12.



Fig. 2.12: Diseño de un estanque sumergido, [23]

Ventajas:

- No necesita dique, a no ser para evitar las inundaciones.
- No hace falta una mesa de agua para el abastecimiento.
- Se requiere poca preparación para la construcción.

Desventajas:

- El nivel del agua puede variar según la estación del tiempo.
- Se requiere el trabajo de excavación.
- Dificultad de gestión del estanque.
- El abastecimiento del agua no es regulado, por lo que se necesita bombeo.

b) Estanque de presa

El estanque de presa se alimenta directamente de un manantial construyendo en el fondo de un valle una presa, como se observa en la figura 2.13.

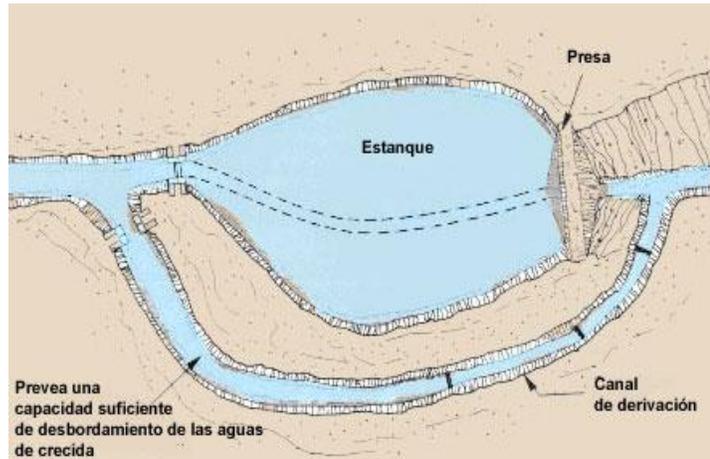


Fig. 2.13: Diseño de un estanque de presa, [23]

Ventajas:

- El diseño es sencillo cuando se trata de pequeñas corrientes.
- Los costos de construcción son bajos, a no ser que haya problemas para evitar las inundaciones.
- La producción natural del producto puede ser mayor debido al abastecimiento del agua.

Desventajas:

- No hay regulación del abastecimiento del agua.
- No se puede vaciar completamente el estanque.
- Forma y dimensiones es irregular.

c) Estanque de derivación

El estanque de derivación se alimenta indirectamente por la fuerza de gravedad o mediante el bombeo, el caudal se regula a través de la toma de agua, como se observa en la figura 2.14.

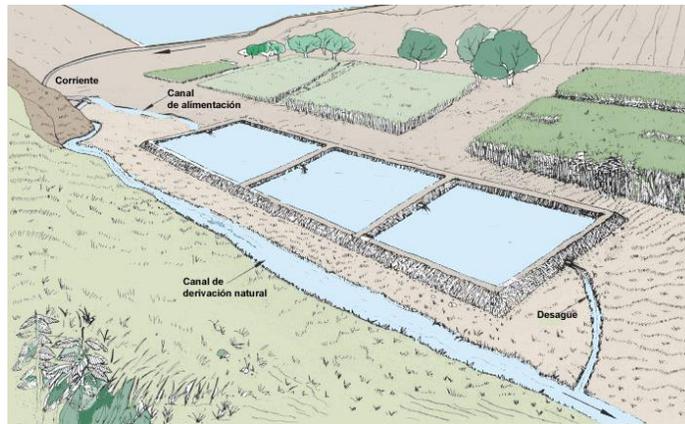


Fig. 2.14: Diseño de un estanque de derivación, [23]

Ventajas:

- Facilidad de control del abastecimiento de agua.
- Se puede vaciar por completo el estanque.
- Su diseño y forma se puede elegir.

Desventajas:

- Los costos de construcción son más elevados que los estanques en presa.
- Se requiere estudios topográficos.

Según su unidad de producción son de varios tipos como: [1]

- Estanque circular
- Estanque rectangular
- Estanques cuadrado
- Estanque ortogonal

2.2.5. Calidad del agua

Los principales parámetros de la calidad y cantidad del agua se detallan a continuación:

Temperatura (°C)

La temperatura es una magnitud física que refleja la cantidad de calor, ya sea de un cuerpo, de un objeto o del ambiente. Está vinculada a la noción de frío (menor temperatura) y caliente (mayor temperatura) que hace referencia al nivel del calor de un cuerpo. La temperatura corporal y la temperatura del entorno del cuerpo adquieren una importancia fundamental para la vida y la salud. [24]

Se utilizan tres unidades de medida como: [24]

- Grados Fahrenheit (°F) del sistema inglés,
- Grados Kelvin (K) del sistema Internacional
- Grados Celsius (°C) unidad derivada de los Kelvin.

Los peces son organismos poiquiloterms cuya temperatura corporal depende del medio en que viven. Por lo tanto la temperatura es la variable más importante y determinante para el cultivo, y a su vez la más difícil de controlar. Cada especie posee un rango óptimo para el crecimiento y desarrollo normal. Conocer las variaciones de la temperatura en forma estadística a lo largo del día con un mínimo de tres registros; en horas de la mañana, luego al medio día y al final de la tarde para obtener el comportamiento de la temperatura diariamente. [15]

El rango permisible de la temperatura del agua para el crecimiento de truchas fluctúa entre 11 a 18 °C, pero el rango óptimo se encuentra alrededor de 15°C, a temperaturas menores del rango óptimo se prolonga el crecimiento, y a temperaturas superiores del rango óptimo el desarrollo es más rápido pero dificulta el engorde debido al estrés que puede generar enfermedades en los peces, de acuerdo a la tabla 2.4. [25]

Tabla 2.4: Comportamiento de la trucha en función al nivel de la temperatura. [25]

Temperatura °C	1 - 10	11 - 13	14 - 16	17 - 18
Consecuencia	Muere	Crecimiento lento	Crecimiento óptimo Buena incubación y reproducción	Estrés Bajo contenido de O ₂

Turbidez (NTU)

La turbidez es una manifestación de la consecuencia óptica provocado por la interferencia y dispersión de los rayos luminosos que traspasan un cuerpo de agua y puede ser provocada por una extensa diversidad de compuestos en suspensión con difusión con distinta forma y tamaño, desde dispersiones coloidales hasta partículas gruesas. [26]

La turbidez es causada por partículas suspendidas en el agua, generalmente arrastradas desde el suelo o de la vegetación adyacente, así como de organismos planctónicos, que pueden generar una disminución en la absorción de oxígeno por parte de las truchas, puesto que sus branquias se ven afectadas presentando problemas branquiales y que pueden dar origen a infecciones, debido a que cuando las branquias de los peces son expuestas al contacto con las partículas suspendidas, se irritan fácilmente ya que se dificulta el pasó del oxígeno a través de ellas. En términos de productividad, la turbidez causa una reducción en la tasa de crecimiento de las truchas. Por otro lado la turbidez impide la visión de los peces para hallar su alimento cuando se dirigen a él con su vista. [5]

Las unidades de medición se expresan: [26]

- FTU (Unidad de Turbidez de la Formazina)
- NTU (Unidad Nefelométrica de Turbidez)
- JTU (Unidad de Turbidez Jackson)

Las truchas captan el oxígeno disuelto en el agua mediante las branquias, el mismo que es trasferido a la sangre, luego llega al corazón y este bombea al torrente sanguíneo, de acuerdo a la figura 2.15; cuando el oxígeno disminuye a valores críticos, generalmente se observa a los peces en la superficie intentando aspirar aire por ende, pueden presentar signos de asfixia. Los peces necesitan el rango óptimo de turbidez aproximadamente de 5 a 30 NTU, como se muestra en la tabla 2.15. [15], [25]

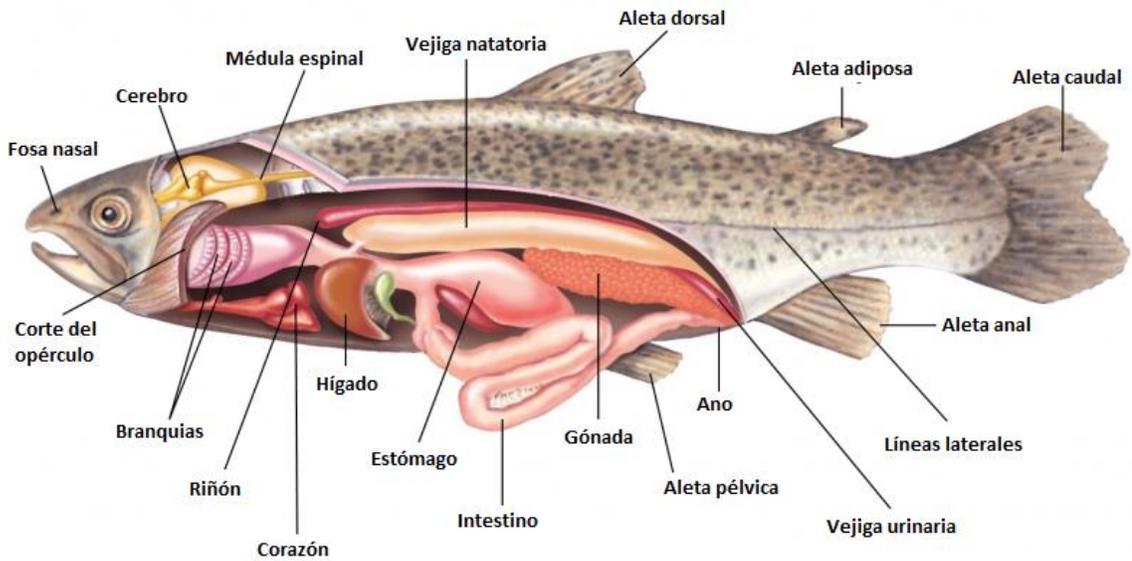


Fig. 2.15: Anatomía de la trucha Arco Iris

Tabla 2.5: Comportamiento de la trucha en función al nivel de turbidez del agua. [27], [28], [29]

Turbidez (NTU)	<3	3 -5	5-30	>30
Consecuencia	<p>Estanque demasiado claro.</p> <p>Productividad inadecuada.</p> <p>Crecen plantas acuáticas.</p>	<p>El fitoplancton es escaso.</p>	<p>Estanque en buenas condiciones.</p>	<p>Estanque demasiado turbio.</p> <p>Baja concentración de oxígeno disuelto.</p>

Nivel de potencial de hidrógeno (pH)

El potencial de hidrógeno es una medida de la acidez o alcalinidad de una sustancia. El pH se expresa a través de una escala que va de 0 a 14 como se visualiza en la figura 2.16. El valor 7 corresponde a un agua neutra, si su pH es inferior a este valor es ácida, y de ser superior es alcalina. Es importante que se conozca no sólo el valor de pH, sino la estabilidad o inestabilidad del mismo, ya que cambios bruscos de pH son perjudiciales

para las especies debido a que puede presentarse enfermedades o afecta en su crecimiento. [15]

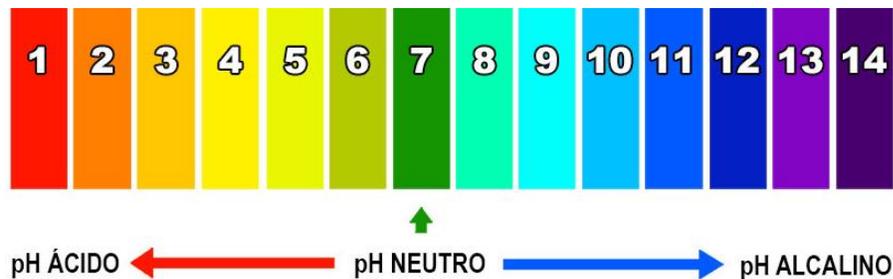


Fig. 2.16: Escala del nivel de pH, [13]

El potencial de hidrogeno actúa como regulador de la actividad metabólica de la trucha, en efecto el nivel neutro es 7, porque la excesiva variación de este parámetro sería perjudicial en el cultivo trucha, si el nivel de pH es menor a 6.5 puede producir hemorragias en las branquias de las truchas y causar muertes, de acuerdo a la tabla 2.6. [25]

Tabla 2.6: Comportamiento de la trucha en función al pH del agua. [25]

pH	4.0 – 5.0	5.1 – 6.5	6.6 – 7.9	8.0 – 10.0
Consecuencia	Mucho estrés Crecimiento lento	Estrés Crecimiento lento	Óptimo desarrollo	Crecimiento lento Muere

Caudal

El caudal es la cantidad de agua que lleva una corriente hacia un canal o cañería que llega a los estanques en un cierto tiempo. [25]

Las unidades de medición del caudal corresponden a un volumen por unidad de tiempo:

- Litros por segundo (L/s)
- Litros por minuto (L/min)
- Litros por hora (L/h)
- Metros cúbicos por hora (m³/h)

Para el desarrollo del cultivo de las truchas es necesario tener en cuenta el volumen del agua a ser utilizado para la infraestructura y relacionado directamente con el nivel de oxígeno disponible para la trucha. Es necesario tener un registro de las renovaciones de agua por hora, la cual tendrá una equivalencia en biomasa producida, a mayor cantidad de agua, mayor será la producción de truchas. [1], [25]

Métodos para la medición: [30]

Entre los métodos más utilizados para la medición de caudales se encuentran los siguientes:

1. Método del flotador

El método del flotador se utiliza en los canales, acequias y solo se obtiene un resultado aproximado.

2. Método volumétrico

El método volumétrico permite medir pequeños caudales de agua en surcos de riego o pequeñas acequias.

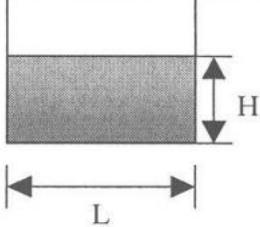
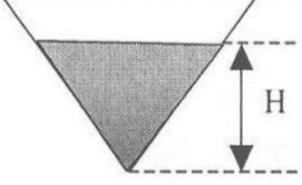
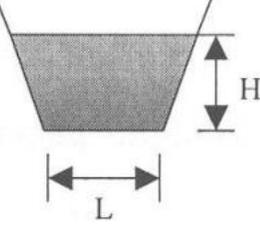
3. Método de la trayectoria

El método de la trayectoria se utiliza el nivel del caudal en tuberías y bombas, tiene una aproximación aceptable debido a su rápida y fácil operación.

4. Estructuras para medición

Existe una gran cantidad de sistemas para la medición del caudal que se clasifican según su forma, como se aprecia en la tabla 2.7:

Tabla 2.7: Clasificación de las estructuras de medición. [30]

		
<p>Vertedero rectangular: Indica el ancho de las cresta “L” y la carga de agua “H”.</p>	<p>Vertedero triangular: Muestra la carga de agua “H”, tiene un ángulo de 90°</p>	<p>Vertedero trapezoidal: Indica el ancho de la cresta “L” y la carga de agua “H”.</p>

2.2.6. Red de sensores inalámbricos (WSN)

Una red de sensores inalámbricos (WSN) consiste en dispositivos autónomos que se encuentran para monitorear condiciones físicas o ambientales. Un sistema WSN incorpora un Gateway y nodos distribuidos como se visualiza en la figura 2.17. El protocolo inalámbrico que se seleccione depende de los requerimientos de la aplicación. Algunos de los estándares disponibles incluyen radios de 2.4 GHz basados en los estándares IEEE 802.15.4 o IEEE 802.11 (Wi-Fi) o radios propietarios, los cuales son regularmente de 900 Mhz. [31]

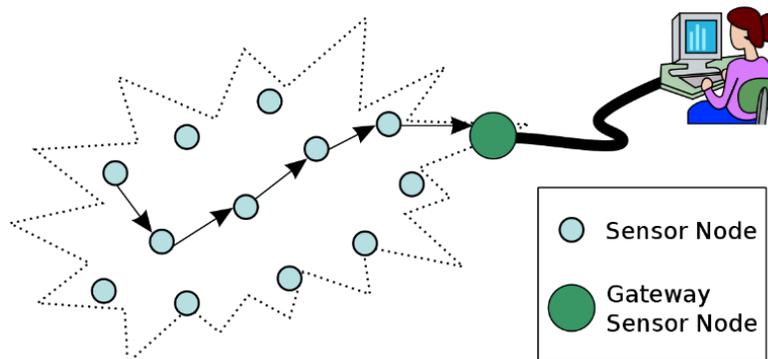


Fig. 2.17: Componentes de WSN Gateway y Nodos Distribuidos. [22]

Parámetros de una red de sensores inalámbricos

Los parámetros principales de una red de sensores inalámbricos son:

- Tiempo de vida
- Cobertura de la red
- Coste y facilidad de instalación
- Tiempo de respuesta
- Precisión y frecuencia de las mediciones
- Seguridad

Los parámetros principales de los nodos sensores:

- Flexibilidad
- Robustez
- Seguridad
- Capacidad de comunicación
- Capacidad de computación
- Facilidad de sincronización
- Tamaño y coste
- Gasto de energía

Topologías de red de sensores inalámbricos

Los tipos de topologías de red de sensores inalámbricos son:

- **Topología de estrella:** La topología de estrella está formada por un nodo final y un Gateway donde cada nodo final se conecta directamente al Gateway, de acuerdo a la figura 2.18. [31]

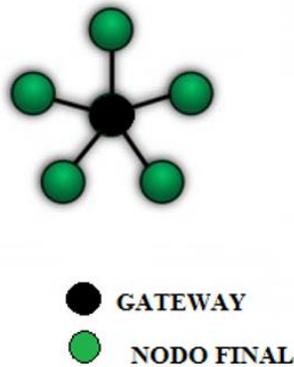


Fig. 2.18: Topologías de WSN tipo estrella

Fuente: La investigación, [31]

- Topología de árbol:** La topología de árbol está formada por un Gateway, nodo router y nodo final donde cada nodo se conecta a un nodo de mayor jerarquía y al nodo Gateway, los datos son ruteados desde el nodo de menor jerarquía en el árbol hasta el Gateway, como se visualiza en la figura 2.19. [31]

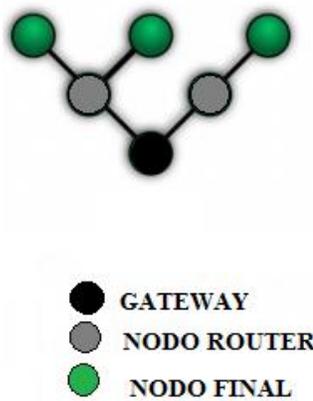


Fig. 2.19: Topologías de WSN tipo árbol

Fuente: La investigación, [31]

- Topología tipo malla:** La topología tipo malla tiene como característica conectar a múltiples nodos en el sistema y pasar los datos por el camino disponible de mayor confiabilidad. En enlace malla es referido como un ruteador, como se muestra en la 2.20. [31]

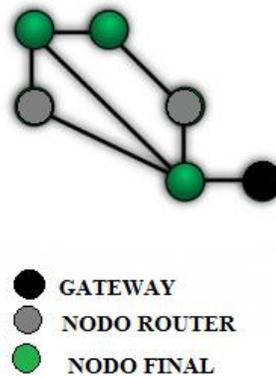


Fig. 2.20: Topologías de WSN tipo malla

Fuente: La investigación, [31]

Aplicaciones:

Las aplicaciones de las redes de sensores son diversas: [32]

- Área militar

Monitoreo de los glaciares, cambio climático, monitoreo del nivel mar, etc.

- Salud

Provisión de interfaces para discapacitados, monitoreo integrado de paciente, monitoreo de estructuras, diagnóstico de enfermedades, etc.

- Agricultura

Algunos ejemplos son el monitoreo de condiciones ambientales que afectan los cultivos, detección de incendios en bosques, detección de inundaciones, estudio de contaminación, etc.

- Automoción

Control de tráfico, distribución del tráfico, etc.

- Domótica

Automatización del hogar, ahorro energético, confort, seguridad, comunicación, etc.

2.2.7. Tecnologías y protocolos de comunicación para WSN

Bluetooth (IEEE 802.15.1)

Bluetooth es un sistema diseñado como una red inalámbrica de área personal, su principal aplicación es la conexión de dispositivos a una computadora personal. Se han realizado prototipos de redes de sensores basadas en Bluetooth, los nodos organizados en picoredes con un nodo maestro y un máximo de siete nodos esclavos activos. El maestro elige la secuencia de hopping que deben seguir los esclavos. Puede haber varios nodos esclavos en estado pasivo en la picored, el maestro interroga los nodos esclavos activos continuamente. [33]

WiFi (IEEE 802.11g)

WiFi es un protocolo de salto-único para redes ad-hoc que provee algunas funcionalidades de ahorro de energía, en general está orientado a altas tasas de transmisión, y los transceptores disponibles requieren una cantidad mayor de energía. Se requiere que los nodos estén permanentemente escuchando al medio, ya que podrían tener que recibir un frame en cualquier momento. [33]

ZigBee (IEEE 802.15.4)

ZigBee con el protocolo subyacente 802.15.4 incluye multi-salto, lo que implica que un mensaje puede utilizar varios saltos en las ondas de radio para llegar a su destino. Los nodos no tienen asignados intervalos específicos de tiempo, sino que han de competir para acceder al canal. Esto permite el acceso de más usuarios al medio inalámbrico, pero introduce incertidumbre en el sistema, ya que la demora y el consumo de energía aumentan cuando un nodo está esperando su turno. [34]

Enhanced Shockburst

Enhanced Shockburst es un protocolo básico que admite la comunicación bidireccional de paquetes de datos que incluye el almacenamiento, el reconocimiento y la retransmisión automática de paquetes perdidos. [35]

Formato del paquete:

En la figura 2.21, se muestra el contenido del paquete con el protocolo Enhanced Shockburst:

- **Preamble:** el preámbulo es una secuencia de bits para sincronizar el demodulador del receptor al flujo de bits entrantes.
- **Address:** es la dirección del receptor para asegurar que los datos son detectados y recibidos por el receptor correcto, esto resulta útil en el caso de que se utilicen varios dispositivos receptores nRF24L01+.
- **Packet Control Field:** la longitud de los datos son de 9 bits, contiene una identificación del paquete que indica si los datos son nuevos o retransmitidos y un bit que indica si se desea o no el paquete de acuse de recibo ACK.
- **Payload:** es el contenido que desea enviar el usuario, es decir los datos de 1 a 32 bytes.
- **Cyclic Redundancy Check (CRC):** es un mecanismo de detección de errores conocido como verificación por redundancia cíclica. [36]

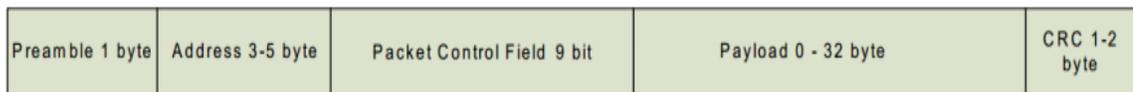


Fig. 2.21: Formato del paquete, [35]

2.2.8. Plataformas de Hardware Libre

Las plataformas de hardware libre son elementos tangibles cuyas especificaciones y diagramas esquemáticos son de acceso público y estos pueden ser estudiados, modificados, recreados y distribuidos por cualquier persona. Esto permite experimentar en las plataformas de hardware libre, modificando el código fuente o el esquema original y logrando crear distintas aplicaciones para una misma plataforma.

Ventajas

Las ventajas del Hardware son las siguientes: [37]

- Independencia tecnológica, permite a las naciones no depender de ninguna otra que le provea los recursos necesarios para su desarrollo.
- Favorece la calidad del hardware, a los estándares abiertos y que sean más económicos.
- El trabajo colaborativo sobre los diseños permite la reutilización y la adaptación de los mismos.
- Disminuye los costos y tiempos de diseño en sus trabajos.
- Se libera a los productores de los propietarios de alianzas globales.

Desventajas

Las desventajas del hardware libre se detallan a continuación: [37]

- Los diseños son específicos y únicos, dificultando su reproducción.
- La reproducción implica primero la prueba de los componentes, para luego fabricar el dispositivo, lo que eleva los costos.
- La disponibilidad de los chips, haciéndose dificultoso para muchos países en vías de desarrollo.
- La fabricación de hardware lleva implícito infraestructura de diseño, simulación, producción e implementación, al contrario de lo que se da en el software libre.

Arduino

Arduino es una plataforma de hardware libre, la cual consta en una placa con diversas entradas y salidas que permiten una gran variedad de aplicaciones, dentro de estas se incluyen la robótica, automatización, electrónica básica y el uso del procesador como medio de interpretar o comunicar información de hardware a un ordenador. [38]

Ventajas:

Las ventajas del Arduino respecto a otros sistemas son: [39]

- Asequible: Las placas Arduino son más asequibles comparadas con otras plataformas de microcontroladores.

- **Multi-Plataforma:** El software de Arduino funciona en los sistemas operativos Windows, Macintosh OSX y Linux. La mayoría de los entornos para microcontroladores están limitados a Windows.
- **Entorno de programación simple y directa:** El entorno de programación de Arduino es fácil de usar para principiantes y lo suficientemente flexible para los usuarios avanzados.
- **Software ampliable y de código abierto:** El software Arduino está publicado bajo una licencia libre y preparada para ser ampliado por programadores experimentados.

Arduino Nano

El Arduino Nano es una pequeña placa, completa y fácil de usar basado en el ATmega328P, consta de catorce pines digitales de entrada y salida de los cuales seis proporcionan salida PWM, ocho pines de entrada analógica, un cristal de cuarzo de 16 MHz, una conexión USB Mini-B, un encabezado ICSP y un botón de RESET. En la figura 2.22, se aprecia la placa del Arduino Nano. (Anexo A) [40]

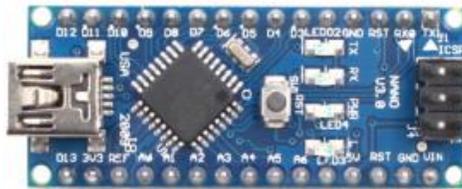


Fig. 2.22: Placa Arduino Nano, [39]

Arduino Mega

El Arduino Mega es una placa construida con un microcontrolador modelo ATmega2560 mostrado en la figura 2.23, tiene 54 pines digitales de entrada y salida de los cuales 15 se pueden ser utilizadas como salidas PWM, 16 entradas analógicas, 4 UARTs, conexión USB, conector ICSP, jack de alimentación y botón de RESET. Esta placa es compatible con la mayoría de las shields diseñados para Arduino. (Anexo B) [41]



Fig. 2.23: Placa Arduino Mega, [40]

Ethernet Shield

La tarjeta ethernet shield está basada en un chip ethernet Wiznet W5100 que se provee de una pila de red IP capaz de soportar los protocolos TCP y UDP, además soporta hasta cuatro conexiones de sockets simultáneos y tiene un conector ethernet estándar RJ45 que permite una conexión Ethernet, como se indica en la figura 2.24. (ANEXO C)

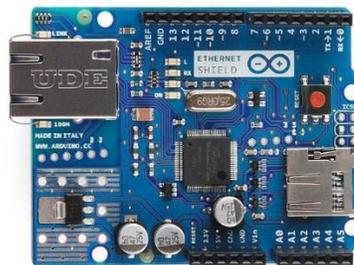


Fig. 2.24: Ethernet Shield, [40]

2.2.9. Computadores de Placas Únicas

Una computadora de placa única está integrada en un solo circuito, los cuales se utilizan en entornos industriales o sistemas embebidos como controladores e interfaces de control para su uso en controladores lógicos programables (PLC). [40]

Raspberry Pi

El Raspberry Pi es un computador de placa única de bajo costo desarrollado por la Fundación Raspberry Pi en 2009, enfocado al uso libre en el nivel educativo. Se encuentra disponible 7 modelos diferentes: Modelo A, Modelo B, Modelo A+, Modelo B+, Pi 2 Modelo B, Pi Zero y Pi 3 Modelo B. Los modelos A y A+ sólo disponen de 1 puerto USB, el modelo B de 2 puertos y las modelos B+ y Pi 2 de 4 puertos USB.

Todo lo que se necesita para empezar a trabajar con la Raspberry Pi es un teclado, mouse USB, una fuente de alimentación, una tarjeta SD para el sistema operativo y un monitor o un televisor que sirvan de pantalla. En la figura 2.25, se muestra una Raspberry Pi 2 Modelo B+. [42] (ANEXO D)



Fig. 2.25: Raspberry Pi 2 modelo B+, [40]

Aplicaciones

Las aplicaciones de Raspberry Pi se pueden mencionar son las siguientes:

- Sistemas de videovigilancia
- Domótica
- Robótica
- IoT
- Servidor web para alojamiento

2.2.10. Sensores

Los sensores son dispositivos con la capacidad de adquirir valores de magnitudes físicas del medio y transformar en variables eléctricas, a continuación se describe cada uno de los sensores:

Sensor de temperatura del agua

El sensor de temperatura del agua es un termómetro digital que permite medir el nivel de temperatura que se encuentra un líquido, su diseño es completamente impermeable. Su punta es de acero inoxidable para soportar el agua salada y es resistente a la corrosión, como se indica en la figura 2.26.

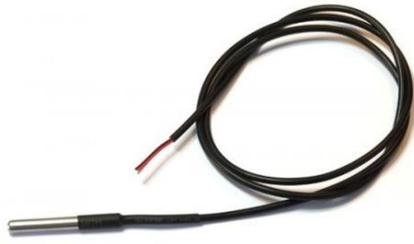


Fig. 2.26: Sensor de temperatura, [42]

Sensor de pH

El sensor de pH mide el nivel de acidez o alcalinidad de una solución, es decir, indica la concentración de iones hidronio (H_3O^+) presentes en determinadas sustancias. En la figura 2.27 se muestra el sensor que permite la medición del nivel de pH de un líquido gracias a la placa controladora que ofrece un analógico proporcional a la medición. El controlador tiene un potenciómetro multivuelta que permita la correcta calibración de la sonda. [43]



Fig. 2.27: Sensor de pH, [42]

Sensor de turbidez

El sensor de turbidez mide la cantidad de luz que es dispersada cuando se dirige a una muestra de agua. Las unidades utilizadas son las unidades de turbidez nefelométricas (UNTs). Los medidores se llaman nefelómetros o turbidímetros, como se observa en la figura 2.28. [44]



Fig. 2.28: Sensor de turbidez, [46]

Sensor de caudal

El sensor de caudal o gasto determina la velocidad del flujo de líquidos que se encuentra basado en ultrasonidos, que generalmente están alimentados eléctricamente y es posible encontrar según su principio de medición de Efecto Doppler y de tiempo de tránsito, como se indica en la figura 2.29. [45]

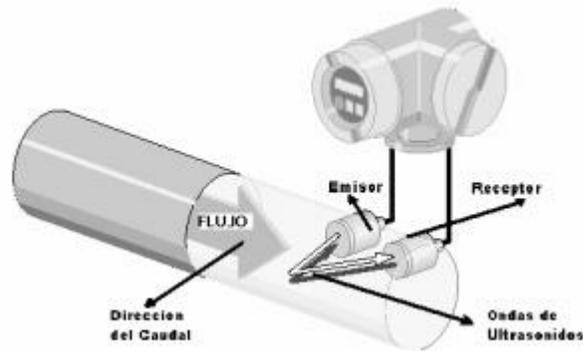


Fig. 2.29: Sensor de caudal basado en ondas ultrasónicas, [47]

2.2.11. Base de datos (BD)

Base de datos es un conjunto estructurado de datos que permite representar necesidades de almacenamiento que asegura la integridad y facilita la labor tanto de usuarios como de los programadores. [46]

Oracle

Oracle es la primera base de datos, es una herramienta cliente/servidor para la gestión de datos de alta potencia y su elevado precio hace que solo las grandes empresas y multinacionales utilicen este sistema de gestión.

Una BD Oracle tiene una estructura física y una estructura lógica:

- La estructura física se corresponde a los ficheros del sistema operativo.
- La estructura lógica está formada por los tablespaces y los objetos de un esquema de BD. [47]

DB2

DB2 Base de datos universal fue construido en base a dos productos como: AIX y Common Server, que para propósitos generales que incluye funciones avanzadas para el mercado de servidores de bases de datos, con soporte de hardware SMP y OLTP. DB2 Parallel Edition fue desarrollado para soportar aplicaciones de gran escala, como Data Warehousing y Data Mining. [48]

MySQL

MySQL es un sistema gestor de bases de datos de código abierto fiable y eficiente, disponible para aplicaciones comerciales, como de entretenimiento debido a su facilidad de uso y alto grado de estabilidad. [46]

Características:

Las principales características de MySQL son: [46]

- La base de datos MySQL está desarrollada en C/C++.
- Está optimizado para equipos de múltiples procesadores.
- Es muy destacable su velocidad de respuesta.
- Se puede utilizar como cliente-servidor o incrustado en aplicaciones.
- Soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes para poder optimizar el SGBD(Sistema Gestor de Base de Datos) a cada caso concreto.
- Su administración se basa en usuarios y privilegios.
- Es altamente confiable en cuanto a estabilidad se refiere.

2.2.12. Protocolo para transferencia simple de correo

STMP es un protocolo estándar que permite la transferencia de correo de un servidor a otro mediante una conexión punto a punto, el funcionamiento es en línea y está encapsulado en una trama TCP/IP. [49]

Puertos del protocolo:

Los puertos SMTP son: [49]

- **25 y 587:** proporciona la conectividad del cliente con el servicio de transporte en la parte delantera de la función de servidor de acceso al cliente (CAS).
- **25, 465 y 475:** utilizados por el servicio de transporte de buzón de correo.
- **7641:** garantiza la entrega de paquetes de datos en el mismo orden, en que fueron enviados.

2.2.13. Free - PDF

FPDF es una clase desarrollada en PHP para realizar documentos en PDF, dinámicamente a partir de los scripts PHP de manera gratuita para cualquier propósito y ser modificada de acuerdo a las necesidades de cada cliente [50].

Las características principales de FPDF son:

- Elección de la unidad de medida, formato de página y márgenes
- Gestión de cabeceras y pies de página
- Salto de página automático
- Salto de línea y justificación del texto automáticos
- Admisión de imágenes (JPEG, PNG y GIF)
- Colores
- Enlaces
- Admisión de fuentes TrueType, Type1 y codificación
- Compresión de página

2.3.Propuesta de Solución

El desarrollo de una red inalámbrica de sensores (WSN) de monitoreo de la calidad del agua para estanques de truchas, se obtuvo diariamente un registro del nivel de temperatura, pH, turbidez y caudal, para prevenir enfermedades o parásitos, las cuales pueden afectar la producción y pueden ser dañinos para el consumidor final.

CAPÍTULO III

METODOLOGIA

3.1. Modalidad de la Investigación

El presente proyecto se obtuvo las siguientes modalidades de la investigación:

Investigación aplicada:

El presente proyecto se considera como una investigación aplicada, porque se realizó la implementación física del sistema de monitoreo en el criadero “El Chaparral” y el cumplimiento de los objetivos.

Investigación documental/bibliográfica:

Se realizó una investigación documental/bibliográfica utilizando fuentes de información primaria y secundaria como libros, revistas, manuales ,artículos científicos y la base de datos bibliográfica, que aportan con el diseño, estudio de los parámetros, resultados experimentales y tecnologías para el desarrollo del proyecto.

Investigación de campo:

La investigación de campo fue ejecutada debido a la necesidad de obtener y recopilar información actual del criadero de truchas, la calidad del agua, que fue útil para el desarrollo del proyecto.

Investigación experimental:

La investigación experimental se utilizó para elaborar un prototipo temporal, el cual muestre en forma aproximada el comportamiento de los circuitos electrónicos, los enlaces inalámbricos de los nodos con el objetivo de depurar los errores que se puedan presentar para evitar los posibles fallos en el sistema que será implementados en los estanques de truchas.

3.2. Recolección de Información

Para la investigación bibliográfica se determinó cuáles son las topologías de red soportadas en redes inalámbricas de sensores con el fin de implementar un sistema de monitoreo de la calidad del agua, para lo cual se acudió a fuentes de información como artículos científicos. Además para la investigación de campo se utilizó registros de estanques para recoger información de la calidad agua para evaluar el estado actual de los estanques de truchas.

3.3. Procesamiento y Análisis de Datos

Una vez analizados los diferentes parámetros que determinó la calidad de agua, se presentaran los datos de forma sistemática a fin de determinar los sensores apropiados para la adquisición de estos parámetros así como la selección de un protocolo que soporte topologías inalámbricas de redes de sensores en función del alcance, fiabilidad y costos, a fin de implementar un sistema robusto, fiable y de bajo costo.

Mediante el siguiente procedimiento:

- Depuración de información y parámetros recopilados.
- Procesamiento de información.
- Optimización de información, parámetros y algoritmos.
- Diseño e implementación de un sistema de monitoreo de la calidad del agua para estanques de truchas.
- Pruebas de la red inalámbrica de sensores.
- Corrección de errores del hardware y software.

3.4. Desarrollo del Proyecto

- Investigación de los métodos de control utilizados para determinar la calidad del agua del criadero “El Chaparral” de la ciudad de Salcedo.
- Determinación del mecanismo de control de la cantidad del agua, destinada a la crianza de las truchas.

- Estudio de los parámetros técnicos que intervienen en la calidad del agua de los estanques de truchas.
- Análisis técnico para determinar las variables que serán monitoreadas de acuerdo a los criterios establecidos por la FAO.
- Delimitación de los límites críticos de las variables que evalúan la calidad del agua para el sistema propuesto.
- Definición de las tecnologías de sensorización más eficientes para la adquisición de los parámetros mencionados en el manual de la FAO.
- Determinación del protocolo de comunicación a utilizar en la WSN en función del alcance, el costo y la eficiencia.
- Diseño de la red inalámbrica de sensores para la adquisición, procesamiento y transmisión de los datos recogidos en los estanques de crianza.
- Diseño de un sistema de presentación de datos y generación de alertas.
- Implementación del sistema propuesto en el criadero “El Chaparral”.
- Ejecución de pruebas de funcionamiento y corrección de errores a fin de calibrar el sistema.
- Elaboración del informe escrito del proyecto.

CAPÍTULO IV

DESARROLLO DE LA PROPUESTA

4.1. Introducción

El presente proyecto de investigación está enfocado en la monitorización de la calidad del agua para los estanques de truchas en el Criadero “El Chaparral”, como es el control periódico de las variables físicas que son la temperatura, el nivel de turbidez, el nivel de pH y la cantidad del recurso hídrico, estas condiciones de calidad del agua son aspectos muy importantes para un proyecto piscícola, según lo señala la Organización de las Naciones Unidas para la Alimentación y Agricultura (FAO). Las condiciones de la calidad del agua intervienen directamente con el desarrollo de las truchas y que esta aporta grandes beneficios en la salud de los seres humanos.

En Ecuador, la producción acuícola está progresando, sin embargo, presenta ciertas dificultades como son la falta de capacitación, problemas de financiamiento, escasa asistencia técnica y poco control en la calidad del agua. El Criadero “El Chaparral” actualmente no dispone de un sistema de monitorización de la calidad el agua para los estanques de las truchas, por lo que se debe revisar diariamente la temperatura del agua que regula el crecimiento de las truchas, su rango es de 13 a 18°C, pero su valor óptimo debe ser alrededor de 15°C. La trucha requiere de una buena cantidad de oxígeno, por ello es necesario conocer el nivel de turbidez del agua debe ser mínimo para evitar que los niveles de oxígeno disuelto (OD) disminuyan. El control del nivel de pH en el hábitat ayuda a la prevención de enfermedades en la piel y evitar contagiar a la camada que debe cumplir con el nivel óptimo de 7 que se le conoce como el “punto neutro”.

La cantidad del agua es de mucha importancia debido a que está relacionado directamente con el nivel de oxígeno que necesitan los peces para su desarrollo y crecimiento, además evita inundaciones en los estanques de las truchas debido a que de manera repentina incrementa el nivel del caudal del río Yanayacu causando pérdidas económicas.

El sistema de monitoreo implementado en el criadero “EL CHAPARRAL” consta de un nodo sensor inalámbrico en la piscina de alevines, el segundo en la piscina de juveniles y el tercer nodo sensor ubicado en el canal de derivación, mediante la topología punto-multipunto se envía los datos desde los nodos esclavos ubicados en cada piscina hacia el nodo maestro ubicado en la oficina del criadero de truchas.

El sistema de monitorización de la calidad del agua propuesto permitió al encargado del criadero disponer de un reporte diario, semanal, mensual o anual del comportamiento de los parámetros que intervienen en el control del recurso hídrico. Garantizando a los consumidores un producto de calidad y contribuir en el aspecto socio-económico de la provincia de Cotopaxi.

4.2. Factibilidad

La factibilidad del proyecto de investigación se encuentra conformada por tres tipos de factibilidad que se detallan a continuación:

4.2.1. Factibilidad técnica

El desarrollo del proyecto es técnicamente factible, debido a que los materiales que se requiere para la construcción de sistema de monitoreo se pueden encontrar con facilidad en las tiendas electrónicas a nivel nacional.

4.2.2. Factibilidad económica

La factibilidad económica es posible, debido a que el investigador solventó con los gastos de los dispositivos que se requieran para la implementación del sistema de monitoreo.

4.2.3. Factibilidad bibliográfica

El desarrollo del proyecto bibliográficamente es factible, debido a que existe la información necesaria, en libros, revistas científicas, tesis y publicaciones técnicas sobre el tema de investigación.

4.3. Fundamentación científico-técnica

Para el desarrollo del sistema de monitorización de la calidad del agua hace referencia en los siguientes numerales:

4.3.1. Estado actual

El criadero “El Chaparral” propiedad de la Sra. Norma Correa con C.I. 0501484174 ubicado en el Barrio Yanayacu de la Ciudad de Salcedo como se muestra en la figura 4.1, este criadero realiza el control del agua de los estanques de forma manual, la oxigenación se obtiene mediante la constante circulación del agua que existe entre cada estanque como se observa en la figura 4.2, de acuerdo a las condiciones hídricas del lugar el flujo de agua es constante todo el tiempo, pero se desconoce si los parámetros de temperatura, pH y turbidez del agua cumplen con los rangos permisibles que intervienen en el crecimiento y desarrollo de las truchas.

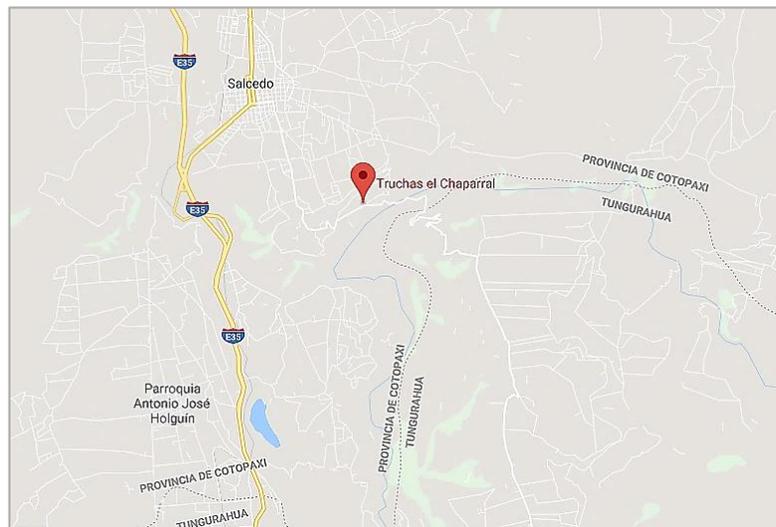


Fig. 4.1: Ubicación del criadero “El Chaparral”

Fuente: Google maps



Fig. 4.2: Sistema de oxigenación en el criadero

Fuente: Investigador

Este criadero no cuenta con ningún registro de calidad del agua que recomienda la FAO. Además, no dispone con operarios quienes ayuden a tomar muestras de la calidad del agua diariamente o alertar inconvenientes provocados por el nivel del agua del río Yanayacu, en la figura 4.3 se observa el cauce del río.



Fig. 4.3: Cauce del río Yanayacu

Fuente: Investigador

Actualmente, el criadero “El Chaparral” mediante una derivación del río Yanayacu abastece de líquido a los estanques de truchas. Este sistema de abastecimiento de agua tiene un desaguadero metros antes de llegar al estanque como se indica en la figura 4.4, la función del desaguadero es prevenir de inundaciones a las piscinas, mediante una compuerta manual.



Fig. 4.4: Desaguadero del criadero "El Chaparral"

Fuente: Investigador

Este criadero cuenta con dos piscinas de alevines, dos piscinas de juveniles y 1 piscina de engorde, cuando el agua entra al estanque, el vertedero tiene una altura, es decir, una pequeña caída de agua para aumentar la oxigenación como se muestra en la figura 4.5. La salida del agua se encuentra ubicada en la parte inferior del estanque, para que el agua tenga mayor circulación, la limpieza de los estanques se realiza cuando el agua se pone de un color oscuro para sacar las heces y restos de alimento que sirven como abono para la vegetación del lugar.



Fig. 4.5: Piscinas del criadero de truchas

Fuente: Investigador

4.3.2. Requerimientos

Para la implementación del sistema de monitorización se requiere el diseño, que está destinado al monitoreo de las variables físicas de la calidad del agua para estanques de truchas en el Criadero “El Chaparral”, es necesario la interacción de los nodos sensores y un servidor LAMP, por lo que se ha desarrollado un esquema general del proyecto.

En la figura 4.6 muestra el diseño general con el cual trabaja el sistema de monitoreo:

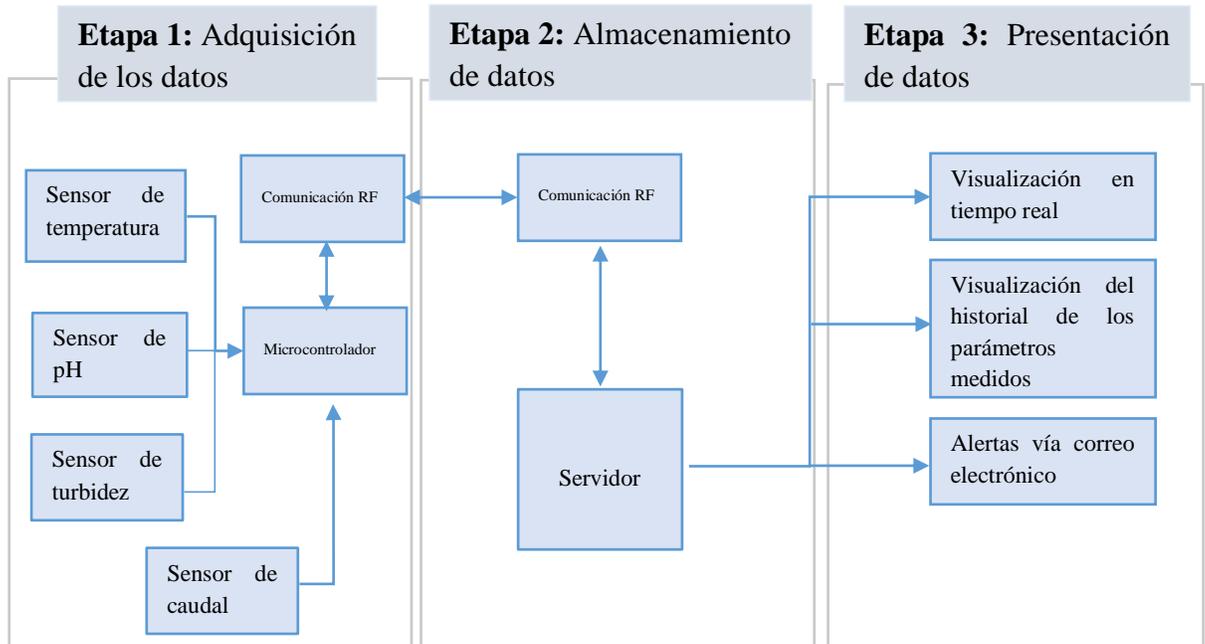


Fig. 4.6: Diagrama general del sistema de monitoreo de la calidad del agua.

Fuente: Investigador

Etapa de adquisición de datos

La etapa de adquisición de datos está formada por los sensores que se encargan de la lectura de las variables, acondicionamiento de señal en un rango de voltaje y corriente soportado por el microcontrolador, los sensores que se utiliza son: el sensor de pH que determina el nivel de acidez o alcalinidad del agua en cada una de las piscinas, el sensor de temperatura determina el nivel calorífico del agua. El sensor de turbidez determina el nivel de partículas suspendidas en el agua y el caudal requerido que se realiza mediante un sensor ultrasónico. Las unidades que se desean medir son programadas en el microcontrolador central y cada nodo sensor consta de un módulo de comunicación, el cual proporciona el medio de transmisión de datos hacia el servidor para el respectivo monitoreo.

Análisis de sensor de temperatura

Para obtener el nivel de temperatura del agua, se realiza un análisis técnico entre dos sensores mediante parámetros técnicos que se detallan en la tabla 4.1:

Tabla 4.1: Sensor DS18D20 vs PT-1000

	DS18B20	PT-1000
Parámetros técnicos		
Alimentación	3.0 a 5.5 V	3.3 a 5.5 V
Rango de temperatura utilizable	-55 a 125 °C (-67 °F a +257 °F)	-200°C a 850°C
Resolución	9 hasta 12 bits Programable	0.001 V/ °C
Capacidad	Multidrop	Ninguno
Conector	Ninguno	BNC
Intervalo de medición	750 ms (máx)	13 s
Precisión	±0.5 °C	±0.15°C
Tubo de acero inoxidable	6 mm * 35 mm	6mm * 81cm
Longitud del cable	90 cm	81 cm
Sumergible	Si	Si
Costo	\$3.50	\$68.00

Fuente: Investigador basado en [51], [52]

Para obtener el nivel de temperatura del agua, se utiliza el sensor DS18B20 porque es un elemento electrónico sumergible, posee características digitales en el cual su resolución es programable y con capacidad multidrop que simplifica las aplicaciones de detección de temperatura.

Análisis del sensor de pH

Para obtener el nivel de pH en el agua de los estanques de truchas, se debe monitorear si el líquido es ácido o alcalino del agua de los estanques de truchas se realiza un análisis entre dos sensores de pH, como se detalla sus especificaciones en la tabla 4.2:

Tabla 4.2: Sensor SEN-0161 vs Sensor E-201 de pH

Parámetros técnicos	SEN-0161	E-201
		
Alimentación	5 V	5 V
Rango de concentración detectable	0 - 14 pH	0 - 14 pH
Rango de temperatura de detección	0 - 60 °C	0 - 50 °C
Resistencia interna	$\leq 250 \text{ m}\Omega$	$\leq 250 \text{ m}\Omega$
Error alcalino	Ninguno	$\leq 15 \text{ mV}$
Punto cero	$7 \pm 0.5 \text{ pH}$	$7 \pm 0.5 \text{ pH}$
Pendiente porcentual teórico (PTS)	$\geq 98.5 \%$	$\geq 97 \%$
Tiempo de respuesta	$\leq 1 \text{ min}$	$\leq 2 \text{ min}$
Precisión	$\pm 0.1 \text{ pH (25}^\circ\text{C)}$	Ninguno
Conector	BNC	BNC/Y
Ajuste de ganancia	Potenciómetro	Ninguno
Tamaño del módulo	43mm*32mm	160mm*12mm
Costo	\$65.00	\$50.00

Fuente: Investigador basado en [43], [53]

Para medir del nivel de pH, se utiliza el sensor SEN-0161 que es un dispositivo robusto que toma medidas periódicamente, su tiempo de respuesta que es menor a 1 min y dispone un potenciómetro de calibración. Mientras que el sensor de pH E-201 tiene error alcalino que hace a este dispositivo se vuelva sensible a los materiales alcalinos con valor de pH mayores a 9, y tiempo de respuesta supera 1 min.

Análisis del sensor de turbidez

Para obtener el nivel turbidez en el agua, se analiza la cantidad de partículas suspendidas en el agua que absorben el calor de la luz de la solar, haciendo que las aguas turbias se vuelvan más calientes, y reduciendo así la concentración de oxígeno en el agua [54]. Las truchas no resisten aguas contaminadas aumentando su índice de mortalidad. Se realizó un análisis entre dos sensores de turbidez, que se detalla en la tabla 4.3:

Tabla 4.3: Sensor SEN0189 vs Sensor LTH serie T

Parámetros técnicos	SEN0189	LTH serie T
		
Alimentación	5 V	24 V
Corriente	30 mA	20 mA
Tiempo de respuesta	500 ms	450 ms
Temperatura de funcionamiento	-30 a 80 °C	0 a 85 °C
Temperatura de almacenamiento	-10 a 80 °C	Ninguno
Tamaño	30mm x 20mm x 12mm	35mm x 22mm x 8mm
Tipo de salida	Analógica 0 a 5V	Analógica 4 a 20 mA
Peso	30 g	40g

Fuente: Investigador basado en [55], [56]

Para la obtención del nivel de turbidez, se realiza a través del sensor SEN0189 por su compatibilidad con plataformas de hardware libre, menor consumo de energía y el menor peso debido a que el nodo sensor debe flotar en el estanque de truchas. Además, este sensor permite ajustar el umbral para aumentar la resolución en el ADC del microcontrolador.

Análisis del sensor de caudal

Actualmente, un sensor de caudal para medir una gran cantidad de agua de un arroyo o canal no existe en el mercado, pero se puede utilizar métodos como del flotador, volumétrico, trayectoria o estructuras de medida. El método de estructuras para la medición de agua es más eficiente y exacto debido a que utilizan estructuras especiales, como son los vertederos. El vertedero rectangular es muy utilizado, debido a la precisión de la lectura que corresponde aproximadamente al 97% de exactitud. [25]

La altura se obtiene mediante el nivel de agua, para el cual se analiza dos tipos de sensores, que se detalla en la tabla 4.4:

Tabla 4.4: Sensor ultrasónico HC-SR04 vs Sensor K-0135

Parámetros técnicos	Ultrasónico HC-SR04	K-0135
		
Alimentación	5 V	3 a 5 V
Corriente	15 mA	< 20 mA
Frecuencia	40 Hz	Ninguno
Área de detección	Calculable en función del ángulo	40mm*16mm
Rango de detección	2 cm a 400 cm	En función de la longitud del sensor
Ángulo de medición	15 grados	Ninguno
Señal de entrada del disparador	10 us pulsos TTL	Ninguno
Señal de salida de eco	La señal de nivel TTL y el rango en proporción	Ninguno
Tamaño del sensor	45mm*20mm*15mm	65mm*20mm*8mm
Costo	\$2.50	\$4.00

Fuente: Investigador basado en [57], [58]

Para obtener el valor del caudal, se utiliza el sensor HC-SR04 porque tiene mayor área de detección incluso se puede calcular en función del ángulo y tiene mayor exactitud la señal de salida de eco del sensor ultrasonido.

Análisis de las placas controladoras

La etapa de adquisición de datos recibe las tramas de datos individuales de cada nodo sensor, y las transforma en una trama general que es transportada a la etapa de procesamiento de datos, donde está alojado el servidor. Se realizó el análisis de las plataformas de hardware libre Arduino nano y uno para adquisición de los datos de los sensores, en el cual se detallan los parámetros en la tabla 4.5:

Tabla 4.5: Análisis de plataformas de hardware libre para la adquisición de datos

Parámetros técnicos	Arduino nano	Arduino uno
		
Voltaje de operación	5 V	5 V
Voltaje de entrada	7 – 12 V	7 – 12 V
Voltaje de salida	6 – 20 V	6 – 20 V
Corriente DC por pin I/O	40 mA	20 mA
Consumo de energía	19 mA	19 mA
Microcontrolador	ATmega328	ATmega328
Pines digitales I/O	22 (6 son PWM)	20
Pines de entrada analógicos	8	6
Velocidad de reloj	16 MHz	16 MHz
Memoria flash	32 KB 2 KB usados para bootloader	32 KB 2 KB usados para bootloader
SRAM	2 KB	2 KB
EEPROM	1 KB	1 KB
Convertidor de USB a serie	FTDI 232	Atmega16U2
Tamaño	18 mm * 45 mm	65mm*20mm*8mm
Peso	7 g	25 g
Costo	\$6.50	\$10.95

Fuente: Investigador basado en [59]

Para la adquisición de los datos de los sensores se requiere un microcontrolador, en efecto el Arduino nano y uno tienen similares características, es por esa razón que se analiza el peso y tamaño, seleccionando al Arduino nano; debido que el nodo sensor debe ser un componente muy compacto, el cual forma parte de un nodo que debe estar sumergido en las piscinas de truchas.

Análisis de la comunicación RF

Para el desarrollo del proyecto se requiere un módulo de comunicación de largo alcance, debido a que las piscinas de truchas se encuentran aproximadamente a unos 122 m de la casa del dueño, el análisis de los módulos se detalla en la tabla 4.6.

Tabla 4.6: Análisis de los módulos de comunicación

Parámetros técnicos	NRF 24L01+	Bluetooth v2.0
		
Voltaje de operación	1.9 V a 3.6 V	3.1 V a 4.2 V
Interfaz de comunicación	SPI	UART
Velocidad de TX	250 kbps y 2 Mbps Configurable	2 Mbps
Nodos por subred	255	8
Consumo	11,3 mA Sleep 26uA	<40 mA Sleep <1mA
Frecuencia de trabajo	2,4 GHz a 2,525 GHz	2,4 GHz a 2,48GHz
Alcance	1000 m	20 m
Retransmisión de paquetes	Automática	Automática
Modulación	GMSK	GFSK
Tamaño	15 mm * 29 mm	12,7 mm * 27 mm
Costo	\$15.99	\$7.99

Fuente: Investigador basado en [60], [61]

Para la comunicación se selecciona al módulo de radiofrecuencia NRF 24L01, porque la velocidad de transmisión es configurable dependiendo de la distancia. Además tiene una interfaz de comunicación SPI más rápida que UART permitiendo la transmisión de datos a alta velocidad, esta interfaz abarca una gran cantidad de dispositivos como esclavos. El alcance es un parámetro principal por la distancia que se requiere para la transmisión de los datos de cada nodo sensor.

Etapas de almacenamiento de datos

La etapa de almacenamiento de datos consta de servidor basado en software libre, el cual tiene la función de recibir los datos adquiridos por los sensores de los diferentes nodos, almacenarlos en una base para que estos se encuentren disponibles y sean consultados en la siguiente etapa que corresponde a la etapa de presentación de datos.

Análisis de las mini computadoras

El presente proyecto propone un sistema de monitoreo de la calidad de agua de los estanques de truchas, por lo que el servidor se puede ejecutar en una mini computadora que permita almacenar la base de datos y los componentes que permite al encargado del criadero monitorear el estado de los parámetros de la calidad del agua. Las mini computadoras denominadas “computadoras de bolsillo” preferido por los desarrolladores por sus amplias características es la denominada Raspberry Pi. En la tabla 4.7 se realizó un análisis técnico entre las diferentes computadoras de bolsillo que el mercado ofrece a los desarrolladores.

Tabla 4.7: Análisis técnico de las computadoras de bolsillo

Parámetros técnicos	Raspberry Pi 2	Orange Pi 2G
		
Arquitectura central	Quad-core ARM Cortex-A7	ARM Cortex-A5
CPU	900 MHz	1.6 GHz
GPU	Broadcom Video Core IV 250 MHz OpenGL ES 2.0	ARM Mali-400 MP2 GPU
Núcleos	4	2
Memoria	1 GB LPDDR2	256MB LPDDR2
Consumo	5 V, 2 A	5 V, 2 A
Ethernet	10/100 BaseT	10/100 Ethernet RJ45
Salida de video	HDMI (rev 1.3 & 1.4)	LCD
Salida de audio	3.5mm jack, HDMI	3.5 mm Jack FM, SPEAK
GPIO	Pin 40	(1x3) pin UART
USB 2.0	4	1 USB HOST 1 USB OTG
Cámara	Pin 15 (CSI-2)	CSI
Tamaño	85 x 56 x 17mm	67mm × 42mm
Ranura para tarjeta de memoria	Micro SD	Micro SD
Costo	\$42	\$35

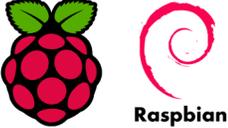
Fuente: Investigador basado en [62], [63]

Se selecciona a Raspberry Pi 2 por que trabaja con una arquitectura ARM de 32 bits, tiene una memoria de 1 GB y 4 núcleos en el procesador, además este dispositivo cuenta con 4 puertos USB.

El Raspberry Pi 2 se trata de una computadora destinada a desarrolladores y de hardware abierto, GNU/Linux lidera en número de distros compatibles con esta minicomputadora,

por lo que dependiendo del uso se debe elegir el sistema operativo, el análisis técnico de las distribuciones GNU/Linux ARM 32 bits se muestran en la tabla 4.8.

Tabla 4.8: Distribuciones GNU/Linux para arquitectura ARM

	Raspbian	Fedora remix
Parámetros técnicos		
Licencia	GPL, GPLv2 y otras libres	GPL, GPLv2 y otras libres
Modelo de desarrollo	Software libre y de código abierto	Software libre y de código abierto
Núcleo	Linux	Linux
Tipo de núcleo	Monolítico	Monolítico
Interfaz gráfica	LXDE y minimal	GNOME y minimal
Plataformas soportadas	ARMv6, ARMv7	ARMv6, ARMv7
Sistema de gestión de paquetes	dpkg	rpm
Basada	Debian	Fedora
Soporte	Si	Si

Fuente: Investigador basado en [64]

Se selecciona a la distribución Raspbian por su sistema de paquetes dpkg porque posee una serie de herramientas para construir un paquete.

Etapas de presentación de datos

La etapa de presentación de datos se utiliza tecnologías de actualización asíncrona de contenido web, para generar una presentación dinámica de los datos obtenidos de los estanques; debe ser una interfaz gráfica amigable con el usuario que al mismo tiempo genere reportes efectivos para la toma de decisiones acertadas que permitan un mejor control en la crianza del pez.

4.4. Diseño de circuitos

De acuerdo a la selección de sensores y tecnologías, se realiza el diseño de los circuitos, para el desarrollo del prototipo de la Red Inalámbrica de Sensores (WSN) de monitoreo de la calidad del agua para estanques de truchas para el Criadero “El Chaparral” del Barrio Yanayacu, del Cantón Salcedo.

a) Circuito del sensor de temperatura del agua

El sensor de temperatura del agua DS18B20 consta de 3 pines, que son los siguientes: Data, Vdd y Gnd como se indica en el anexo E. El sensor se alimenta desde arduino nano a través de los pines Vdd y Gnd, también se conecta Data al pin 2 del arduino, incluso se debe conectar entre los pines Vdd y Data una resistencia de 4.7K Ω , como se muestra en la figura 4.7.

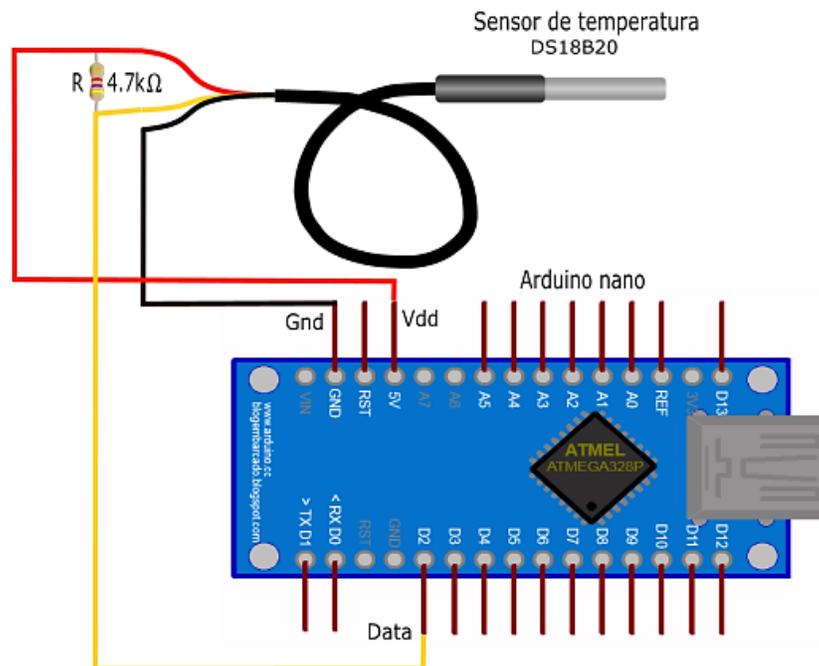


Fig. 4.7: Esquema eléctrico del sensor de temperatura DS18B20

Fuente: Investigador

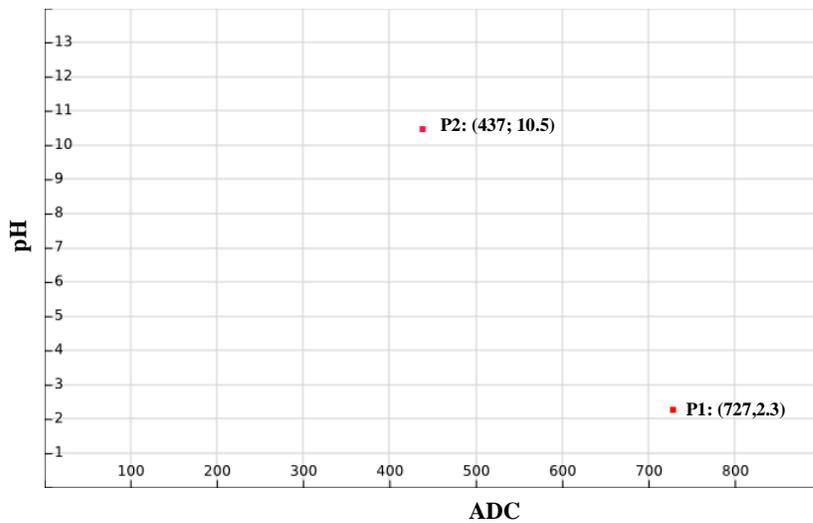


Fig. 4.9: Gráfica de la ecuación de conversión entre ADC y pH medido

Fuente: Investigador

Utilizando la fórmula de la ecuación de la recta:

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad (1)$$

$$m(x - x_1) = y - y_1 \quad (2)$$

Para obtener el valor de la pendiente se reemplaza en la ecuación (1):

$$P_1 = (437; 10.5)$$

$$P_2 = (727; 2.3)$$

$$m = \frac{2.3 - 10.5}{727 - 437} = \frac{-8.2}{290} = -0.028275862068$$

Luego, se reemplaza en la ecuación (2) para obtener la fórmula general:

$$y - y_1 = m(x - x_1)$$

$$y = m(x - x_1) + y_1$$

$$y = -0.028275862068(x - 437) + 10.5$$

$$y = -0.028275862068x + 12.356551724137 + 10.5$$

$$y = -0.028275862068x + 22.856551724137 \quad (3)$$

La ecuación (3) permite obtener una linealización de los valores ADC convertidos en valores de pH, que se observa en la tabla 4.9 y se gráfica en la figura 4.10:

Tabla 4.9: Valores del ADC y de pH

N°	X (ADC)	Y (pH)
1	565,00	6,88
2	560,00	7,02
3	574,00	6,63
4	566,00	6,85
5	542,00	7,53
6	541,00	7,56

Fuente: Investigador



Fig. 4.10: Gráfica de la ecuación de conversión entre ADC y pH medido

Fuente: Investigador

c) Circuito del sensor de turbidez

El sensor de turbidez SEN0189, consta de 3 pines, que son los siguientes: Out, Vcc y Gnd mostrado en la figura 4.11, y sus características técnicas de dicho sensor se encuentra en el anexo G. El sensor se alimenta desde arduino nano a través de los pines Vcc y Gnd, a continuación se conecta Out al pin A1 del arduino.

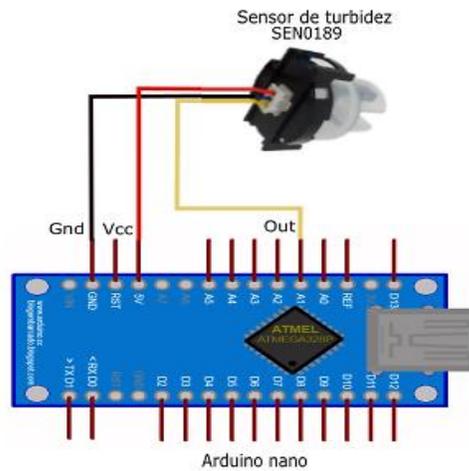


Fig. 4.11: Esquema eléctrico del sensor de turbidez SEN0189

Fuente: Investigador

El sensor de turbidez es lineal, por lo que se toma dos puntos que permite deducir la ecuación para convertir el valor del ADC a NTUs, utilizando el agua de un reservorio con un valor de turbidez = 17 NTUs y agua lluvia con un valor de turbidez = 3 NTUs se obtiene los valores del ADC 659 y 719 respectivamente, estos puntos son utilizados para la ecuación de la recta que permite obtener un valor de pH. En la figura 4.12, se obtiene la representación gráfica de conversión entre el valor del ADC y NTUs.

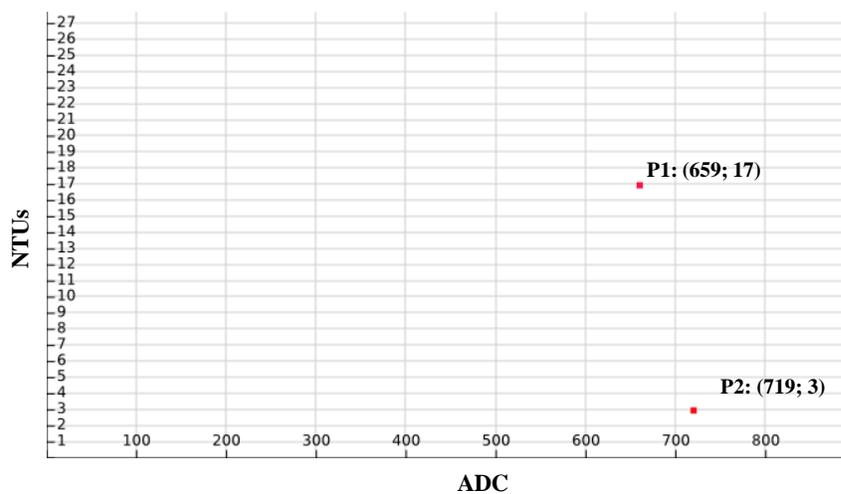


Fig. 4.12: Gráfica de la ecuación de conversión entre ADC y NTUs

Fuente: Investigador

Utilizando la fórmula de la ecuación de la recta:

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad (1)$$

$$m(x - x_1) = y - y_1 \quad (2)$$

Para obtener el valor de la pendiente se reemplaza en la ecuación (1):

$$P_1 = (659; 17)$$

$$P_2 = (719; 3)$$

$$m = \frac{3 - 17}{719 - 659} = -\frac{14}{60} = -0.2333$$

Luego, se reemplaza en la ecuación (2) para obtener la fórmula general:

$$y - y_1 = m(x - x_1)$$

$$y = m(x - x_1) + y_1$$

$$y = -0.2333(x - 659) + 17$$

$$y = -0.2333x + 170.7666 \quad (4)$$

La ecuación (3) permite obtener una linealización de los valores ADC convertidos en valores de turbidez, que se observa en la tabla 4.10 y se gráfica en la figura 4.13:

Tabla 4.10: Valores del ADC y NTUs

N°	X (ADC)	Y (NTUs)
1	592,00	32,65
2	585,00	34,29
3	619,00	26,35
4	659,00	17,02
5	628,00	24,25
6	664,00	15,86

Fuente: Investigador

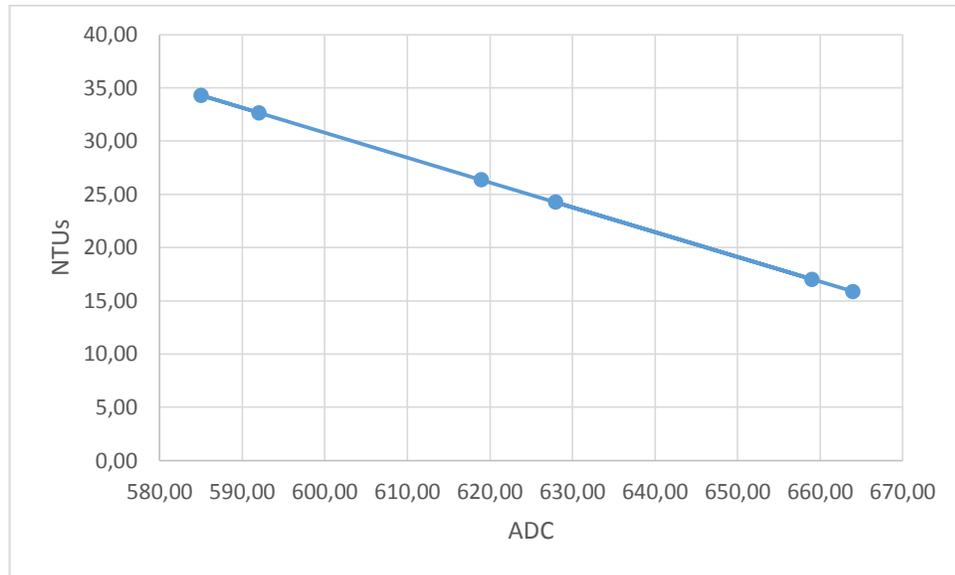


Fig. 4.13: Gráfica de la ecuación de conversión entre ADC y NTUs

Fuente: Investigador

d) Circuito del sensor ultrasónico

El sensor de ultrasónico HC-SR04 consta de 4 pines, que son los siguientes: Vcc, Trig, Echo y Gnd, como se detalla en el anexo H. En la figura 4.14, se muestra el sensor que se alimenta desde arduino nano a través de los pines Vcc y Gnd, a continuación se conecta Trig al pin D8 del arduino, Echo al pin D7.

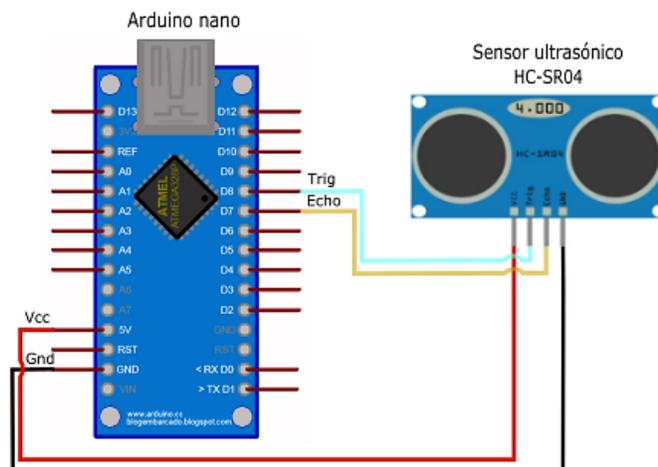


Fig. 4.14: Esquema eléctrico del sensor ultrasónico HC-SR04

Fuente: Investigador

Para calcular la fórmula del caudal se ha tomado muestras de tiempo que recorre un objeto una distancia determinada, utilizando el método del flotador:

Datos:

Distancia promedio: 3 m

Ancho de la acequia: 0.42 m

Profundidad de la acequia: 0.60 m

Tabla 4.11: Valores de muestras de tiempo que recorre un objeto

Muestra	Tiempo
T1	11 s
T2	12 s
T3	12 s
T4	13 s

Fuente: Investigador

Utilizando la ecuación de la media aritmética:

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} \quad (5)$$

Para obtener un tiempo promedio se reemplaza en la ecuación (5):

$$\bar{x} = \frac{\sum_{i=1}^4 x_i}{4}$$

$$\bar{x} = \frac{11 + 12 + 12 + 13}{4} = \frac{48}{4} = 12s$$

Velocidad de la superficie del agua se obtiene mediante la ecuación (6):

$$V_s = \frac{d}{t} \quad (6)$$

$$V_s = \frac{d}{t} = \frac{3 \text{ m}}{12 \text{ s}} = 0.25 \text{ m/s}$$

Velocidad media del agua se obtiene mediante la ecuación (7) :

$$V_m = V_s * \text{Coeficiente de corrección} \quad (7)$$

$$\text{Coeficiente de corrección} = 0.85$$

$$V_m = 0.25 \text{ m/s} * 0.85 = 0.2125 \text{ m/s}$$

Para obtener la fórmula del caudal se reemplaza el valor de la ecuación (7) :

$$Q = V_m * \text{Ancho de la acequia} * \text{Profundidad de la acequia}$$

$$Q = 0.2125 \text{ m/s} * 0.42 \text{ m} * \text{Profundidad de la acequia}$$

$$Q = (0.2125 \text{ m/s} * 0.42 \text{ m} * \text{Profundidad de la acequia}) * 1000 \text{ L} \quad (8)$$

e) Circuito de comunicación NRF 24L01+

El módulo de comunicación NRF 24L01+ consta de 8 pines, que son los siguientes: Vcc, Gnd, SCK, MISO, MOSI, SCN y CE como se detalla en el anexo I. Los pines del módulo de comunicación se conecta de la siguiente manera: el pin Vcc del NRF 24L01+ al Out del regulador de voltaje 1117, CE al D9, SCN al D10, MOSI al D11, MISO al D12 y SCK al D13 del arduino; se debe conectar el pin Gnd del regulador y del NRF al arduino y finalmente el Vin del regulador 1117 al Vin del arduino; como se indica en la figura 4.15.

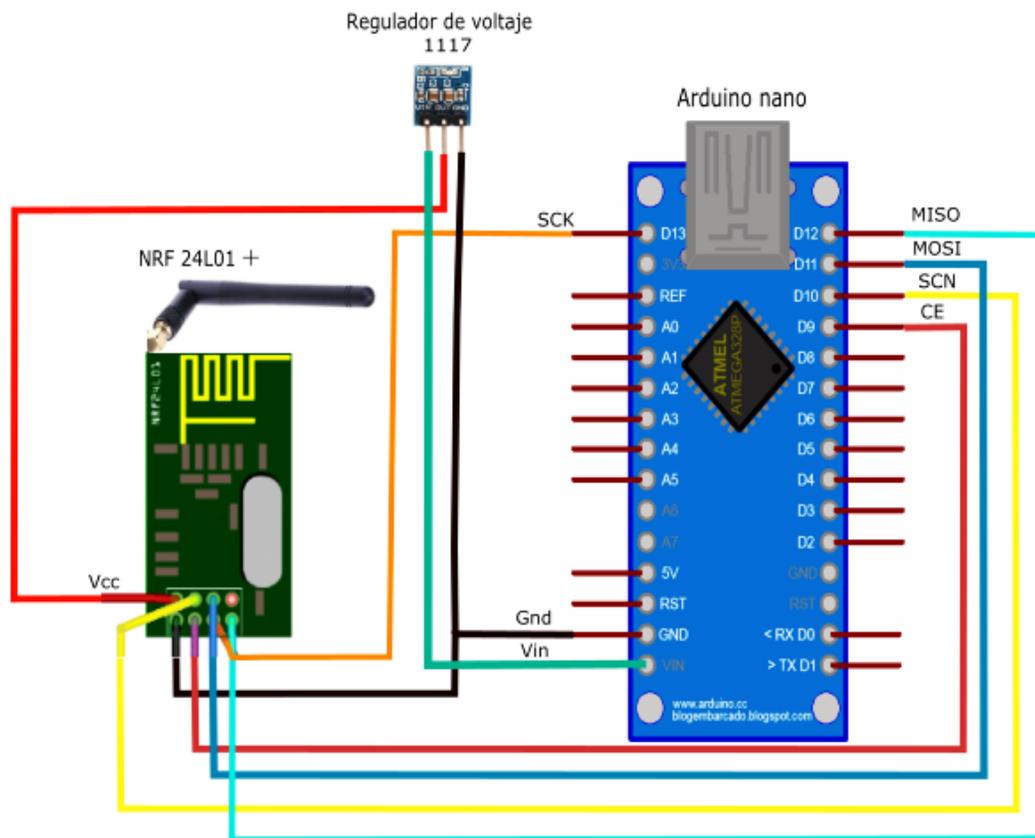


Fig. 4.15: Esquema eléctrico del módulo de comunicación NRF 24L01

Fuente: Investigador

f) Diseño de los nodos sensores

El diseño de los nodos sensores consta de dos partes:

1. El primer diseño esta formado por un arduino nano, un regulador de voltaje 1117 y el módulo de comunicación NRF 24L01+, además de un sensor ultrasónico que permite la medición del caudal del agua que ingresan a las piscinas de truchas y el nivel de temperatura, que se visualiza en la figura 4.16.

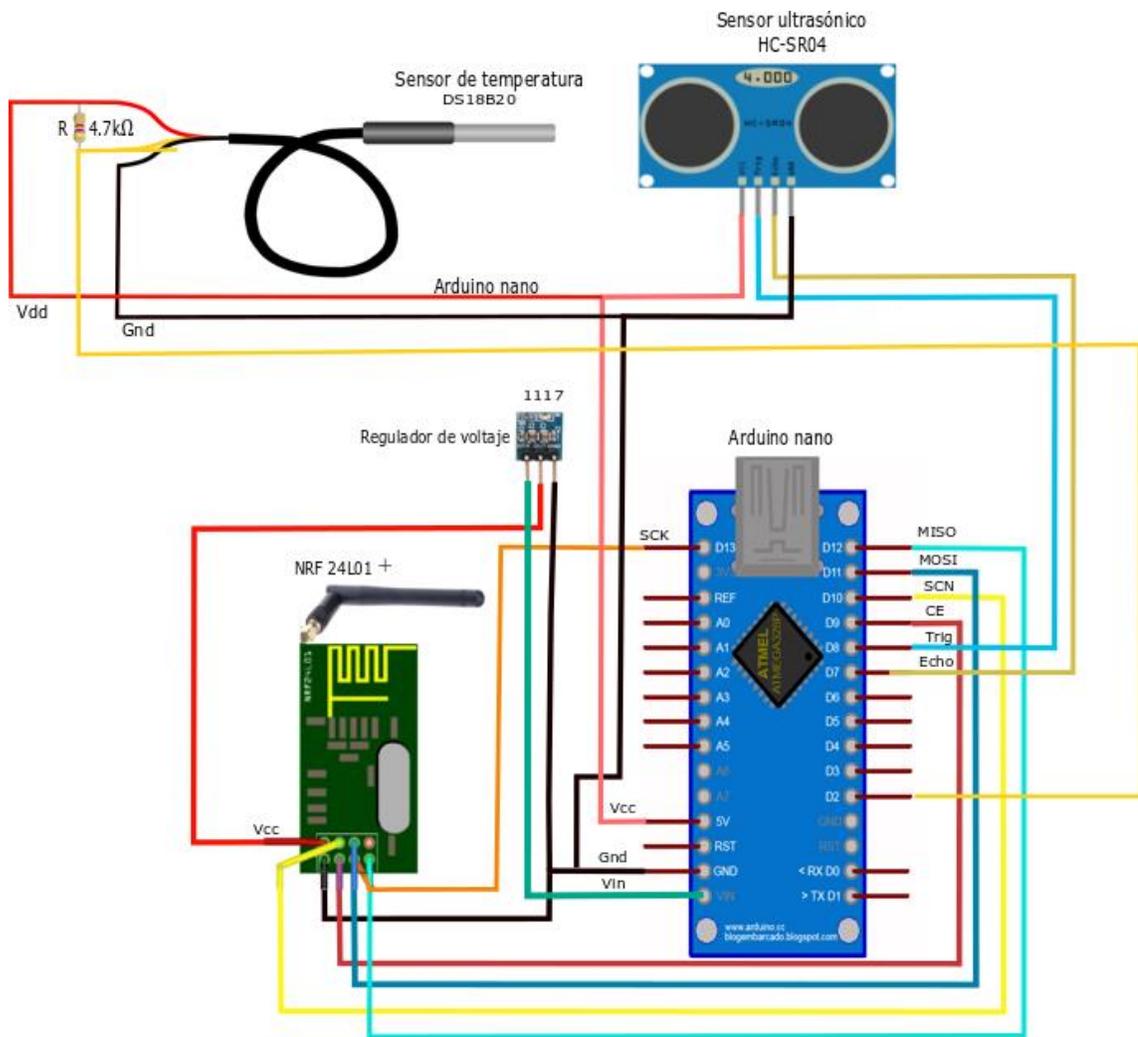


Fig. 4.16: Esquema eléctrico del primer nodo sensor

Fuente: Investigador

2. El segundo diseño, mostrado en la figura 4.17 se encuentra conformado por un arduino nano, un regulador de voltaje 1117 y el módulo de comunicación NRF 24L01+, además de un sensor de temperatura DS18B20, un sensor de turbidez SEN0189 y sensor de pH.

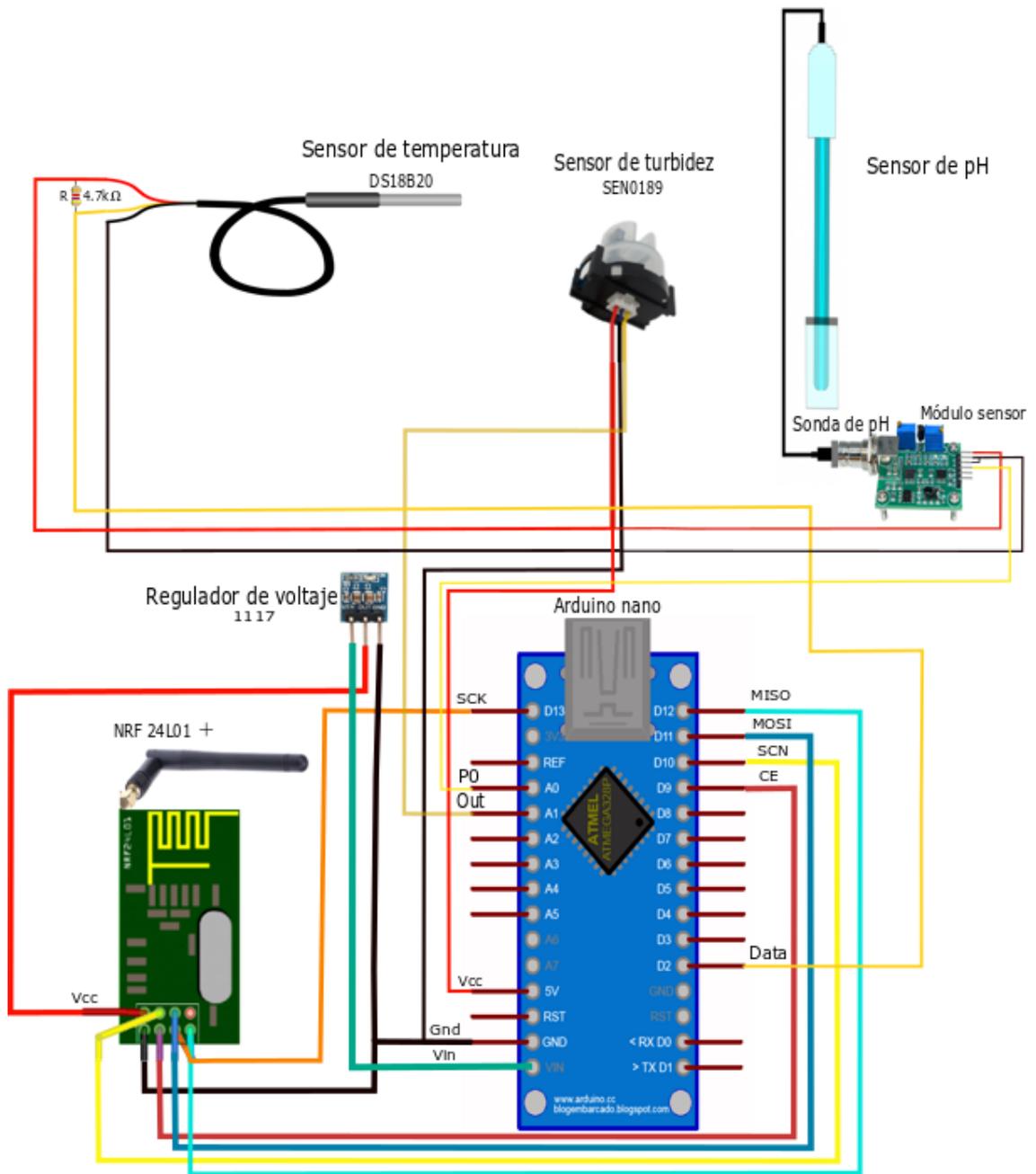


Fig. 4.17: Esquema eléctrico del segundo nodo sensor

Fuente: Investigador

4.5. Flujogramas de procesos

Para el proceso de la lectura de los valores de pH, temperatura, turbidez y nivel de caudal se empleó el diagrama de flujo de la figura 4.18, que permite tomar muestras cada seis horas y enviar lo datos registrados mediante el protocolo Shockburst a la base de datos.



Fig. 4.18: Diagrama de flujo de la red inalámbrica de los sensores

Fuente: Investigador

Para el inicio del servidor se emplea el siguiente diagrama de flujo de la figura 4.19. El diagrama permite la actualización de la base de datos, monitoreo de las variables en un rango adecuado, en el cual se realiza la notificación mediante E-mail y la visualización en una interfaz gráfica de usuario.

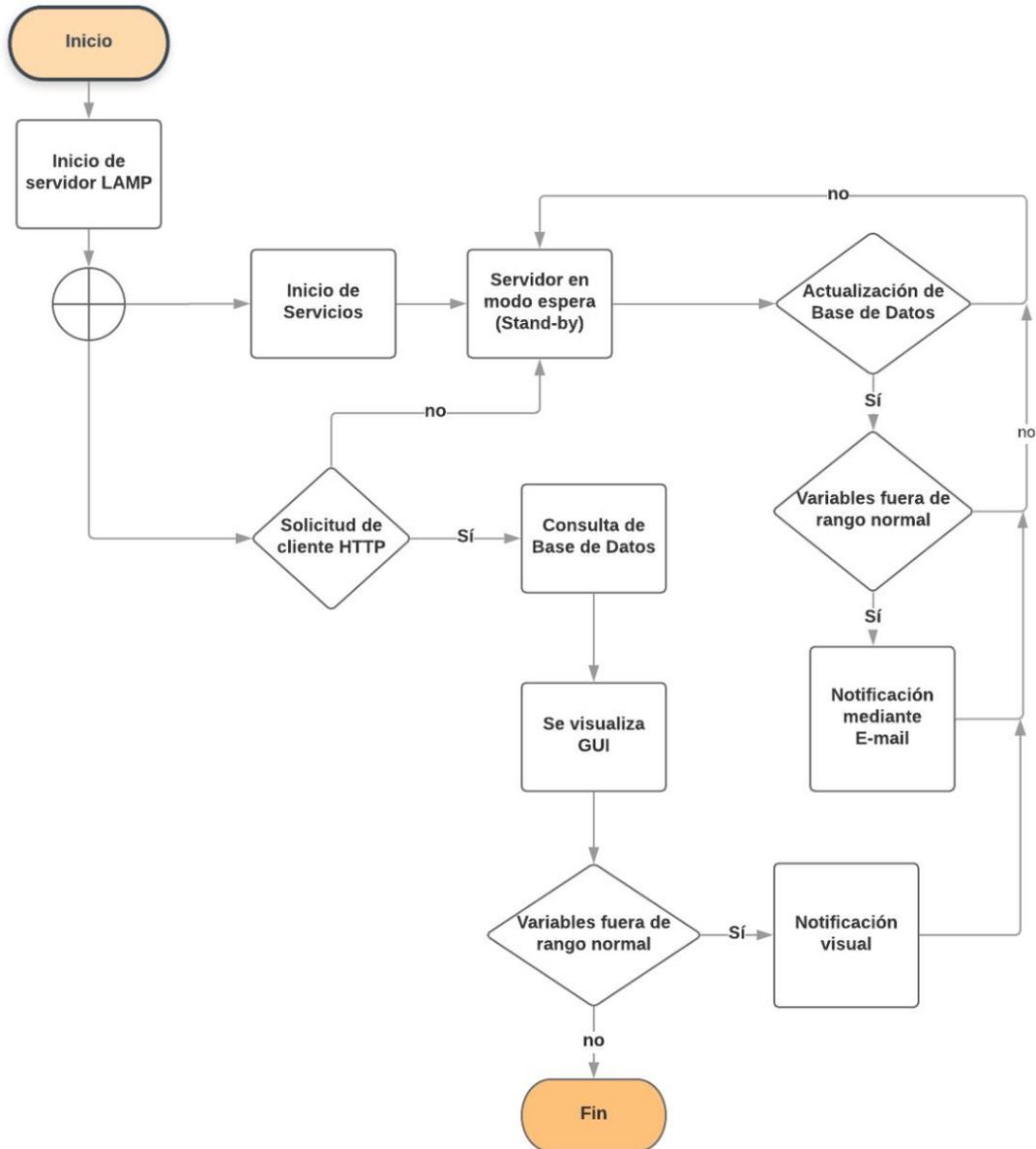


Fig. 4.19: Diagrama de flujo del prototipo

Fuente: Investigador

4.6. Instalación del sistema operativo en la Raspberry Pi

La instalación del sistema operativo oficial para la Raspberry Pi es Raspbian, que se puede descargar de la página oficial de la Fundación Raspberry, que se indica en la figura 4.20.

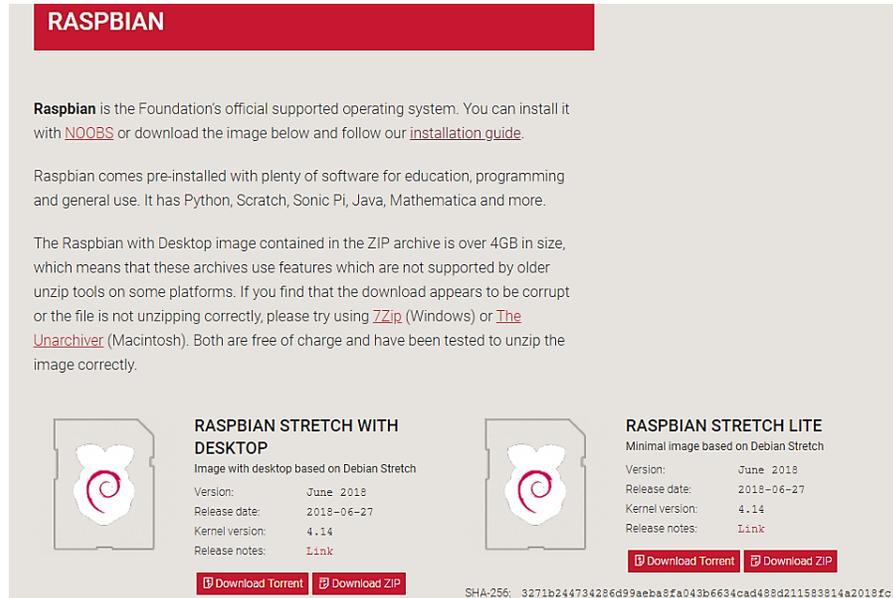


Fig. 4.20: Página de la Fundación Raspberry

Fuente: Investigador

Una vez descargado la imagen del sistema operativo de la página oficial, se procede a la instalación mediante el uso de una tarjeta micro SD de 16 GB y de clase 10, es primordial que la tarjeta SD se encuentre completamente vacía. Una vez descargada la imagen se procede a descomprimirla utilizando en siguiente comando:

```
$ sudo unzip 2018-06-27-raspbian-stretch-lite.zip
```

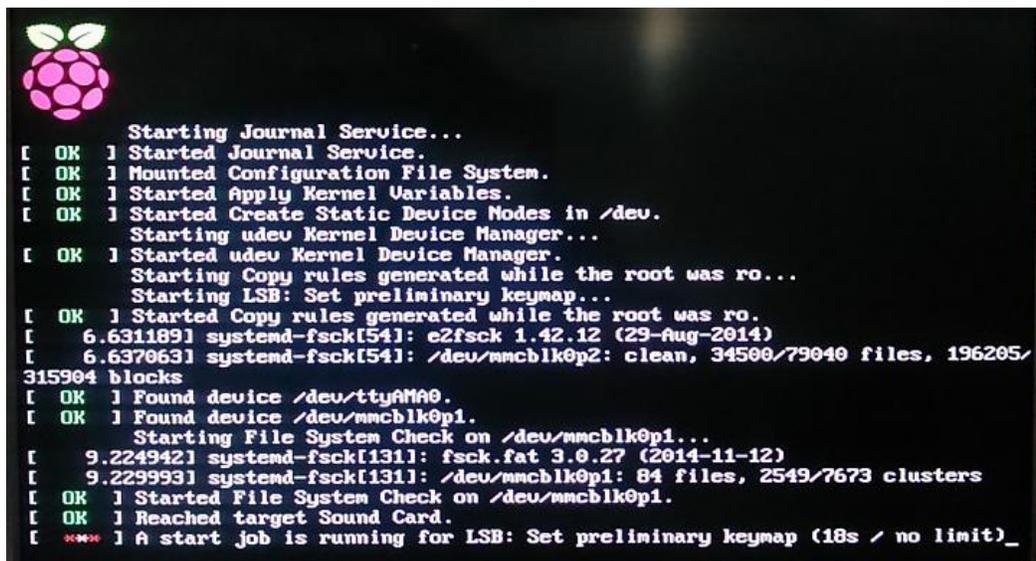
Luego, se obtiene un archivo 2018-06-27-raspbian-stretch-lite.img. Se procede a insertar la tarjeta microSD en el computador. Para la instalación de sistema operativo Raspbian por medio de consola, primero conocer la localización de la tarjeta mediante el comando:

```
df-h
```

Como resultado de este comando se obtiene el fichero /dev/sdb1. Entonces se procede a escribir la imagen en tarjeta con el siguiente comando:

```
$sudo dd bs=1M if=2018-06-27-raspbian-stretch-lite.img of=/dev/sdb1
```

Una vez concluida la instalación del sistema operativo en la tarjeta, se introduce la microSD en la Raspberry Pi, luego se conecta a la pantalla a través del puerto HDMI. Es necesario conectar la fuente de alimentación para que se encienda la Raspberry Pi B y el inicio del sistema operativo, como se observa en la figura 4.21.



```
Starting Journal Service...
[ OK ] Started Journal Service.
[ OK ] Mounted Configuration File System.
[ OK ] Started Apply Kernel Variables.
[ OK ] Started Create Static Device Nodes in /dev.
Starting udev Kernel Device Manager...
[ OK ] Started udev Kernel Device Manager.
Starting Copy rules generated while the root was ro...
Starting LSB: Set preliminary keymap...
[ OK ] Started Copy rules generated while the root was ro.
[ 6.631189] systemd-fsck[54]: e2fsck 1.42.12 (29-Aug-2014)
[ 6.637063] systemd-fsck[54]: /dev/mmcblk0p2: clean, 34500/79040 files, 196205/
315904 blocks
[ OK ] Found device /dev/ttyAMA0.
[ OK ] Found device /dev/mmcblk0p1.
Starting File System Check on /dev/mmcblk0p1...
[ 9.224942] systemd-fsck[131]: fsck.fat 3.0.27 (2014-11-12)
[ 9.229993] systemd-fsck[131]: /dev/mmcblk0p1: 84 files, 2549/7673 clusters
[ OK ] Started File System Check on /dev/mmcblk0p1.
[ OK ] Reached target Sound Card.
[ *** ] A start job is running for LSB: Set preliminary keymap (18s / no limit)_
```

Fig. 4.21: Inicio del sistema operativo

Fuente: Investigador

El sistema operativo Raspbian requiere una actualización de los paquetes y versiones disponibles, se recomienda realizar este proceso antes de la instalación de cualquier programa en la Raspberry Pi B mediante el comando:

```
$sudo apt-get update
```

4.7. Instalación del servidor Apache

La instalación del servidor apache requiere la ejecución del siguiente comando:

\$sudo apt-get install apache2

Para la verificar si la instalación fue exitosa, se ingresa al navegador <http://localhost> y se visualizará la siguiente figura 4.22.

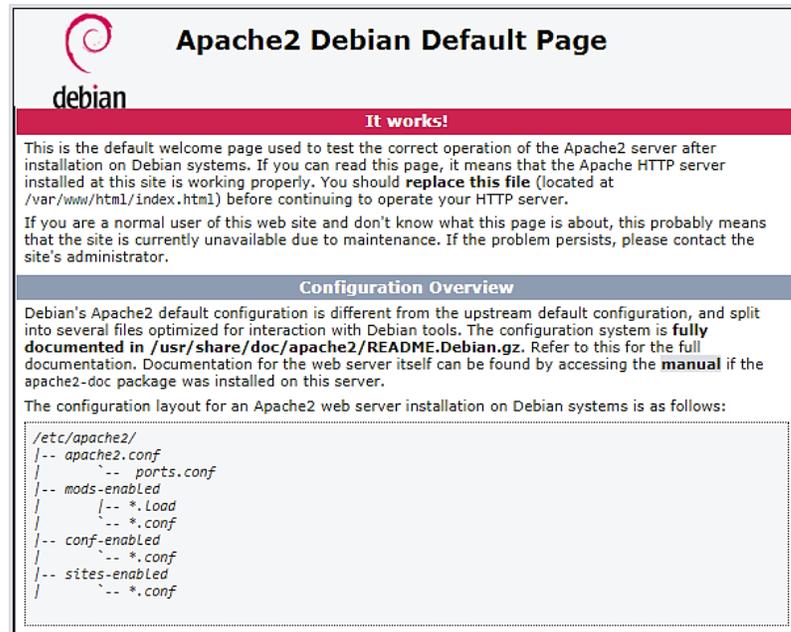


Fig. 4.22: Instalación del servidor Apache

Fuente: Investigador

4.8. Instalación de MySQL

MySQL es un sistema de gestión de base de datos, que se encarga y facilita el acceso a las bases de datos donde se almacenará la información de los parámetros de la calidad del agua, para la instalación se requiere el siguiente comando:

\$sudo apt-get install mysql-server-php5 mysql

Durante el proceso de instalación de MySQL, se debe ingresar una contraseña y luego confirmarla para completar la instalación, como se observa en la figura 4.23.

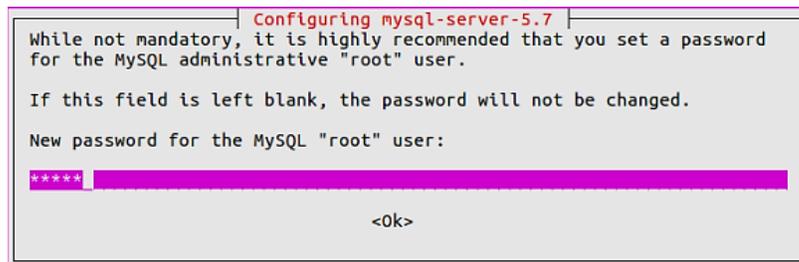


Fig. 4.23: Instalación de MySQL

Fuente: Investigador

Una vez terminada la instalación, se debe ejecutar los siguientes comandos para crear la estructura de directorios de MySQL donde se almacena la información.

```
$ sudo mysql_install_db  
$ sudo mysql_secure_installation
```

4.9. Instalación de PHP

Para la instalación de PHP se ejecuta el siguiente comando:

```
pi@raspberrypi:/ $ sudo apt-get install libapache2-mod-php5
```

Este comando se configura automáticamente para integrarse en Apache, creando los archivos necesarios en la carpeta de los módulos disponibles de apache y los enlaces necesarios en la carpeta de módulos habilitados. Para conectar a bases de datos MySQL desde PHP, se requiere el siguiente comando:

```
$sudo apt-get install php5-mysql
```

A continuación se ingresa al archivo html y se crea un archivo info.php

```
pi@raspberrypi:~ $ cd /var/www/html/  
pi@raspberrypi:/var/www/html $ sudo nano info.php  
<?php phpinfo(); ?>
```

Para comprobar la instalación de PHP, se ingresa la dirección en el navegador: <http://192.168.0.3t/info.php>, en la figura 4.24 se puede visualizar la correcta instalación del servidor apache.

192.168.0.3/info.php

PHP Version 5.6.38-0+deb8u1 	
System	Linux raspberrypi 4.9.35-v7+ #1014 SMP Fri Jun 30 14:47:43 BST 2017 armv7l
Build Date	Sep 20 2018 05:27:11
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-gd.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-mcrypt.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysql.ini, /etc/php5/apache2/conf.d/20-readline.ini
PHP API	20131108
PHP Extension	20131226
Zend Extension	220131226
Zend Extension Build	API220131226.NTS
PHP Extension Build	API20131226.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, mcrypt.*, mdecrypt.*
<small>This program makes use of the Zend Scripting Language Engine: Zend Engine v2.8.0, Copyright (c) 1998-2016 Zend Technologies with Zend OPcache v7.0.8-dev, Copyright (c) 1999-2016, by Zend Technologies</small> 	

Fig. 4.24: Instalación del servidor Apache

Fuente: Investigador

4.10. Instalación de PhpMyAdmin

Para la instalación de PhpMyAdmin se debe ejecutar el siguiente comando en el terminal:

\$ sudo apt-get install phpmyadmin

En el proceso de la instalación se visualizará un mensaje que es mostrado en la figura 4.25, en el cual se deberá seleccionar el servidor web “apache2” y luego “ok”.

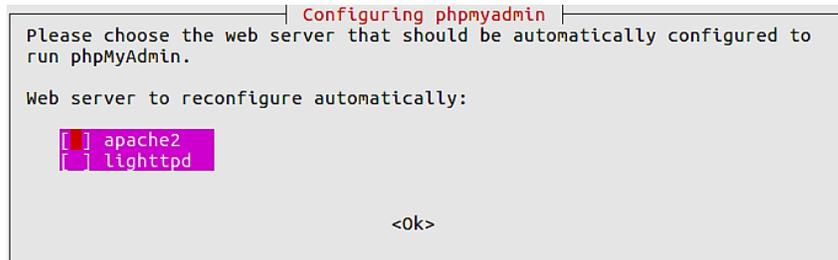


Fig. 4.25: Instalación del servidor Apache

Fuente: Investigador

A continuación en la figura 4.26, se selecciona “Yes” para configurar la base de datos para PhpMyAdmin mediante “dbconfig-common”, luego se solicita el ingreso de una contraseña para la base de datos.

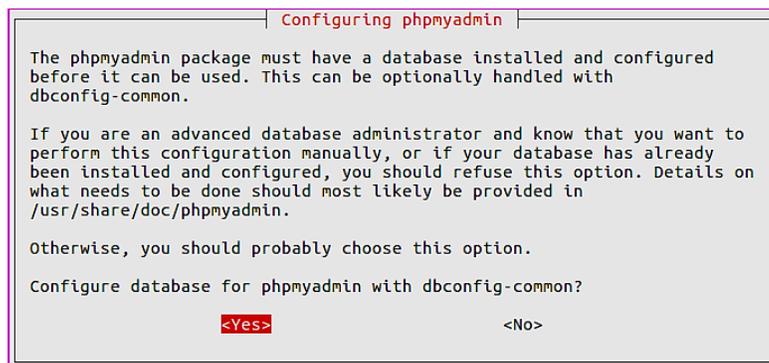


Fig. 4.26: Configuración de PhpMyAdmin

Fuente: Investigador

Además, se debe realizar las siguientes configuraciones para la base de datos y se ingresa a la terminal el comando para obtener acceso al archivo:

apache2.conf

Al final del archivo se debe insertar la sentencia:

include /etc/phpmyadmin/apache.conf

sudo nano /etc/apache2/apache2.conf

Para reiniciar el servicio se ingresa el comando:

sudo /etc/init.d/apache2 restart

Ingresar al fichero php.ini mediante el siguiente comando:

sudo nano /etc/php5/apache2/php.ini

Una vez en el fichero, se inserta la siguiente línea de código “extension=mywql.so”. Finalmente, para la actualizar los comandos insertados se debe reiniciar la Raspberry Pi 2 mediante el siguiente comando:

```
sudo reboot
```

Se crea una carpeta con el nombre conf.d

```
sudo mkdir /etc/apache2/conf.d
```

```
sudo ln -s /etc/phpmyadmin/apache.conf/etc/apache2/conf.d/phpmyadmin.conf
```

Para actualizar el servidor Apache2 se inserta en la terminal el comando:

```
sudo /etc/init.d/apache2 reload
```

Para ingresar a la base de datos se ingresa en el navegador la siguiente dirección <http://192.168.0.3/phpmyadmin>, que se visualiza en la figura 4.27.

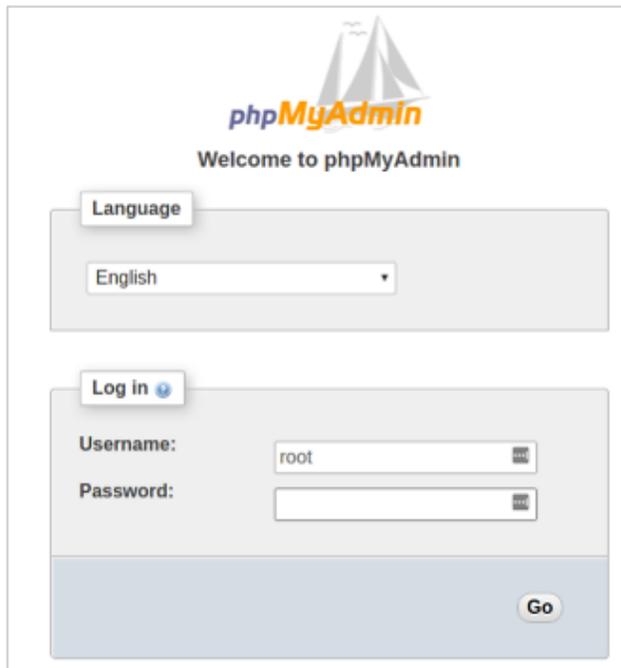


Fig. 4.27: Instalación de phpMyAdmin

Fuente: Investigador

4.11. Base de datos en PhpMyAdmin

Para ingresar a la base de datos en PhpMyAdmin, insertar el “usuario” y la “contraseña”, como se observa en la figura 4.28.

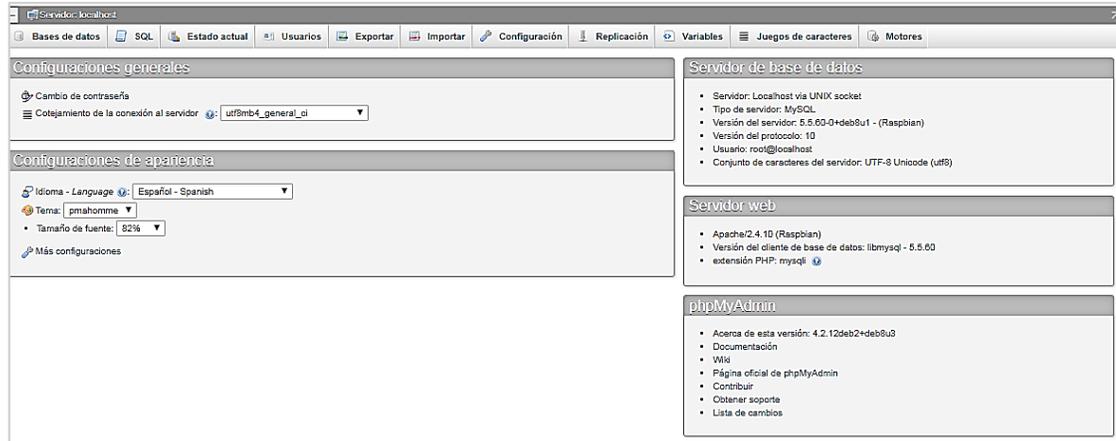


Fig. 4.28: Base de datos en phpMyAdmin

Fuente: Investigador

Para crear una base de datos se inserta un nombre “monitoreo”, como en la figura 4.29, donde se almacena la información de los sensores para la calidad del agua en los estanques de truchas.



Fig. 4.29: Base de datos “monitoreo”

Fuente: Investigador

A continuación, se selecciona la base de datos creada “monitoreo” y se inserta tres tablas que representan cada nodo sensor.

a) Nodo sensor 1 (Canal de derivación)

En el nodo sensor 1 ubicado en el canal de derivación consta del sensor ultrasónico, por lo que se crea una tabla de 1 columna con el nombre “entrada”, como se indica en la figura 4.30.

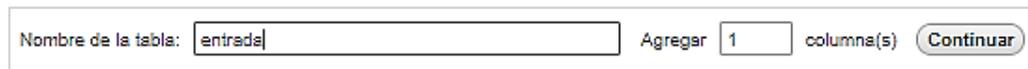


Fig. 4.30: Tabla para el nodo sensor 1 “entrada”

Fuente: Investigador

Una vez creada la tabla, se configura las columnas con el nombre del sensor “caudal” en donde se almacena la información obtenida por este en la tabla, como se observa en la figura 4.31.



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
1	id	int(11)			No	Ninguna	AUTO_INCREMENT	Cambiar, Eliminar, Primaria
2	caudal	float			No	Ninguna		Cambiar, Eliminar, Primaria

Fig. 4.31: Columnas de la tabla “entrada”

Fuente: Investigador

b) Nodo sensor 2 (Piscina 1)

En el nodo sensor 2 ubicado en la piscina 1 consta del sensor de pH, turbidez y temperatura; en la figura 4.32 se crea la tabla “piscina1” con 5 columnas.



Fig. 4.32: Base de datos “piscina1”

Fuente: Investigador

A continuación, se debe configurar las columnas con los nombres de los sensores “ph”, “temperatura” y “turbidez” más dos columnas que corresponde al “id” y la “fecha” como se indica en la figura 4.33.



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
1	id	int(11)			No	Ninguna	AUTO_INCREMENT	Cambiar Eliminar Primaria
2	ph	float			No	Ninguna		Cambiar Eliminar Primaria
3	temperatura	float			No	Ninguna		Cambiar Eliminar Primaria
4	turbidez	float			No	Ninguna		Cambiar Eliminar Primaria
5	fecha	timestamp			No	CURRENT_TIMESTAMP		Cambiar Eliminar Primaria

Fig. 4.33: Columnas de la tabla “piscina1”

Fuente: Investigador

c) Nodo sensor 3 (Piscina 2)

En el nodo sensor 3 ubicado en la piscina 2 consta del sensor de pH, turbidez y temperatura; en la figura 4.34 se crea la tabla “piscina2” que se encuentra conformado por 5 columnas.



Nombre de la tabla: Agregar columna(s)

Fig. 4.34: Base de datos “piscina2”

Fuente: Investigador

Además, en la figura 4.35 se debe configurar las columnas con los nombres de los sensores “ph”, “temperatura” y “turbidez” más dos columnas que corresponde al “id” y la “fecha”.



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
1	id	int(11)			No	Ninguna	AUTO_INCREMENT	Cambiar Eliminar Primaria
2	ph	float			No	Ninguna		Cambiar Eliminar Primaria
3	temperatura	float			No	Ninguna		Cambiar Eliminar Primaria
4	turbidez	float			No	Ninguna		Cambiar Eliminar Primaria
5	fecha	timestamp			No	CURRENT_TIMESTAMP		Cambiar Eliminar Primaria

Fig. 4.35: Columnas de la tabla “piscina2”

Fuente: Investigador

4.12. Instalación de PHPMailer

Para la instalación de PHPMailer se requiere descargar la librería del repositorio de GitHub, como se indica en la figura 4.36. La librería PHPMailer consta de 2 archivos `class.phpmailer.php` y `class.smtp.php`.

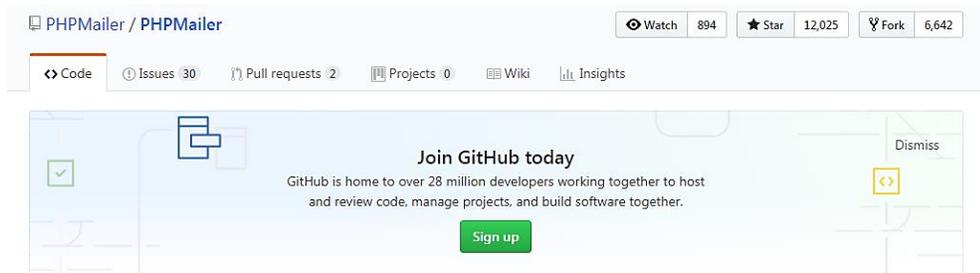


Fig. 4.36: Repositorio de la librería PHPMailer

Fuente: Investigador

```
1 <?php
2 include "class.phpmailer.php";
3 include "class.smtp.php";
4
5 $email_user = "el.chaparral.salcedo@gmail.com";
6 $email_password = "Ytjda3toG";
7 $the_subject = "Phpmailer Monitoreo";
8 $address_to = " el.chaparral.salcedo@gmail.com ";
9 $from_name = " Ytjda3toG ";
10 $phpmailer = new PHPMailer();
11
12 // ----- datos de la cuenta de Gmail -----
13 $phpmailer->Username = $email_user;
14 $phpmailer->Password = $email_password;
15 //-----
16 // $phpmailer->SMTPDebug = 1;
17 $phpmailer->SMTPSecure = 'ssl';
18 $phpmailer->Host = "smtp.gmail.com"; // GMail
19 $phpmailer->Port = 465;
20 $phpmailer->IsSMTP(); // use SMTP
21 $phpmailer->SMTPAuth = true;
22
23 $phpmailer->setFrom($phpmailer->Username,$from_name);
24 $phpmailer->AddAddress($address_to); // recipients email
```

25	
26	\$phpmailer->Subject = \$the_subject;
27	\$phpmailer->Body .= "<h1 style='color:#3498db;'>ALARMA!!!VALOR FUERA DE PARAMETROS PERMITIDOS, PECES EN PELIGRO!</h1>";
28	\$phpmailer->Body .= "<p>Mensaje personalizado</p>";
29	\$phpmailer->Body .= "<p>Fecha y Hora: ".date("d-m-Y h:i:s")."</p>";
30	\$phpmailer->IsHTML(true);
31	
32	\$phpmailer->Send();
33	?>

Finalmente, se obtiene el correo con el estado de los sensores del proyecto como se indica en la figura 4.37.

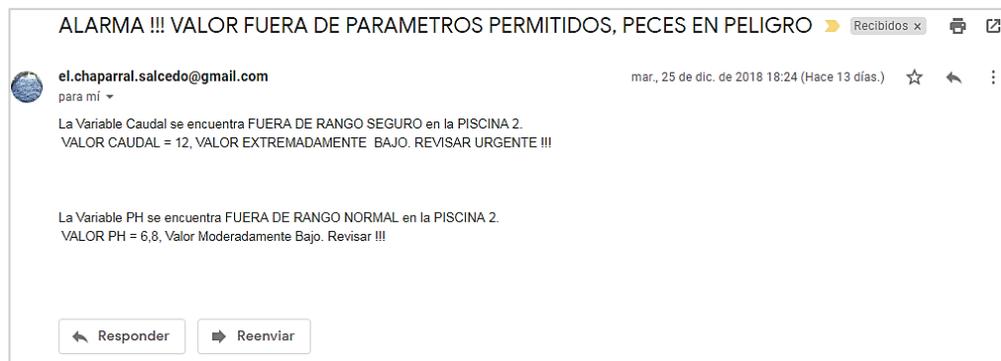


Fig. 4.37: Correo usando una cuenta Gmail

Fuente: Investigador

4.13. Diseño de la interfaz gráfica de usuario (GUI)

El diseño de la interfaz gráfica de usuario, se programa en Adobe Dreamweaver y se guarda en diferentes archivos con extensiones según el lenguaje de programación como php, html, ccs y js estos archivos se guardan en el servidor en la ruta /var/www/html desde donde se ejecutan cuando haya una solicitud de un cliente.

Una vez creada la base de datos donde se almacenan las variaciones de los parámetros de la calidad del agua, se requiere que estos datos sean presentados de una manera amigable con el usuario, a continuación se muestra en la figura 4.38 la forma en la que trabajan las diferentes tecnologías WEB que se utilizó para la creación de la interfaz de usuario.

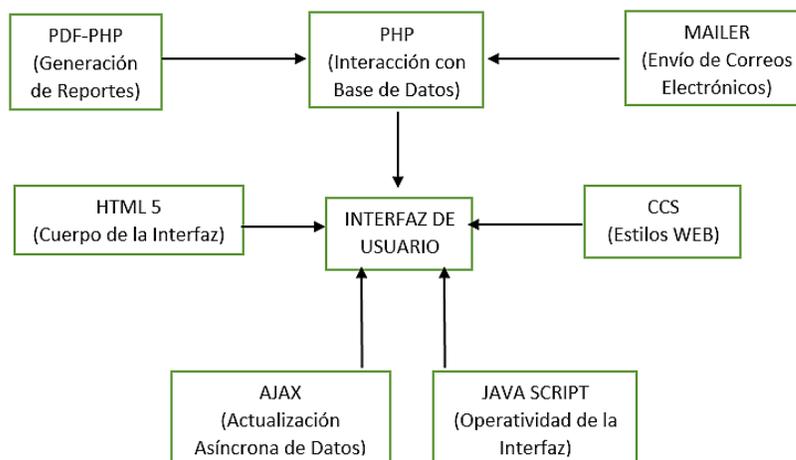


Fig. 4.38: Diagrama de operación de interfaz de usuario

Fuente: Investigador

A continuación se presenta la interfaz en la que el usuario monitorea los parámetros de calidad del agua, como se indica en la figura 4.39.

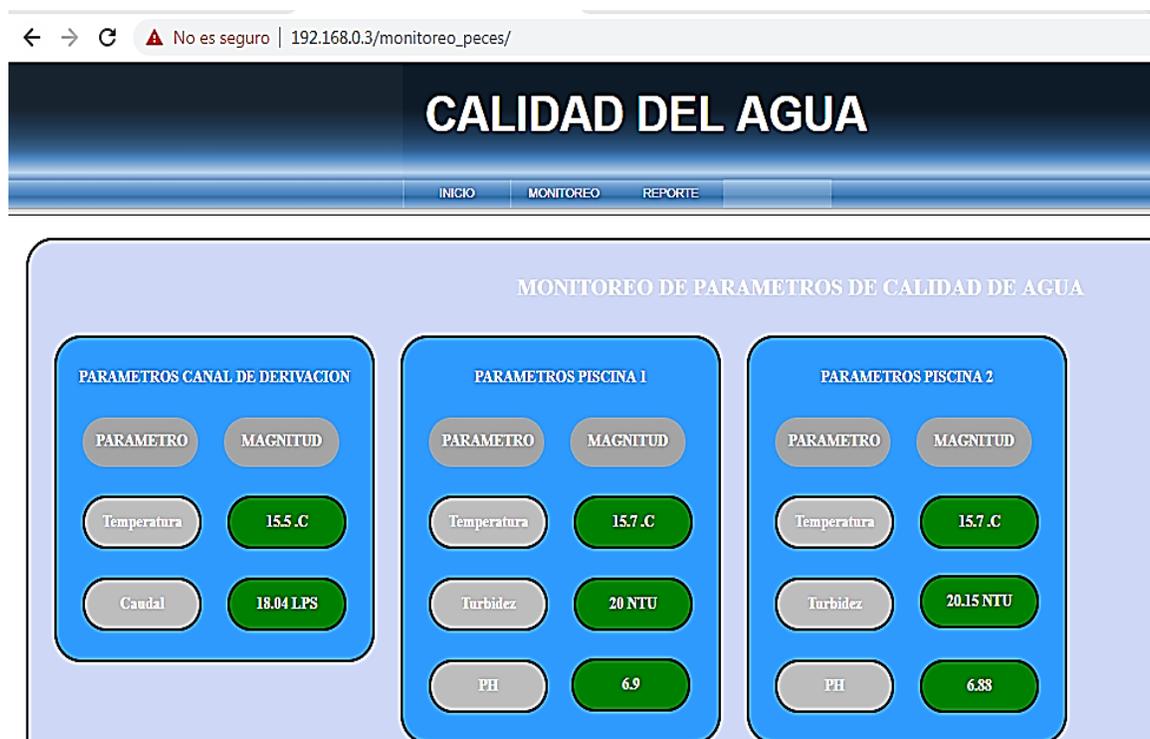


Fig. 4.39: Interfaz del proyecto

Fuente: Investigador



Fig. 4.40: Interfaz del proyecto

Fuente: Investigador

4.14. Implementación del prototipo

La implementación del prototipo se realiza mediante la interacción de competentes electrónicos como son plataformas de hardware libre, computadora de placa única y sensores. En la figura 4.41 y 4.42 se puede observar los diseños de las placas realizado en Inkscape, que permite la conexión de los elementos y dispositivos para cada nodo sensor.

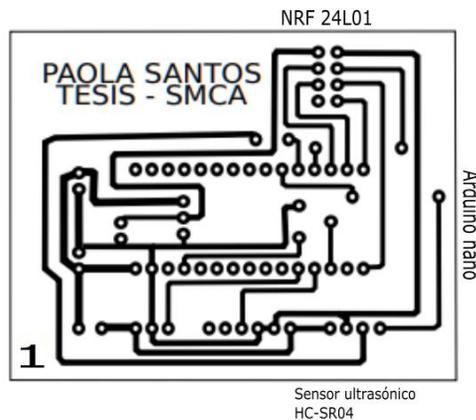


Fig. 4.41: Circuito impreso del Nodo 1

Fuente: Investigador

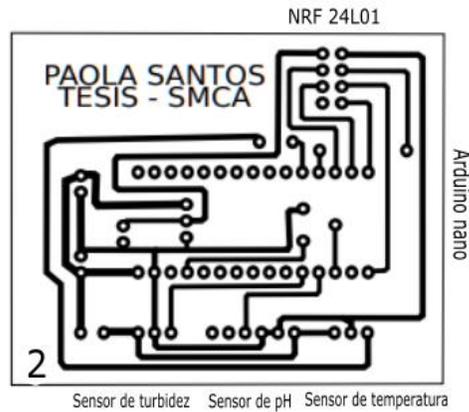


Fig. 4.42: Circuito impreso del Nodo 2 y 3

Fuente: Investigador

Una vez realizado la impresión en papel couche, se procede a realizar el diseño de PCB (Printed Circuit Board) con el método de la plancha de cada uno de los circuitos diseñados de cada nodo sensor, luego se realiza las perforaciones con un taladro para ubicar los componentes y finalmente puedan ser soldados, como se indica en la figura 4.43.



Fig. 4.43: Soldadura de los elementos

Fuente: Investigador

Luego, se coloca los componentes en cada nodo sensor para sus posteriores pruebas como se muestra en la figura 4.44.

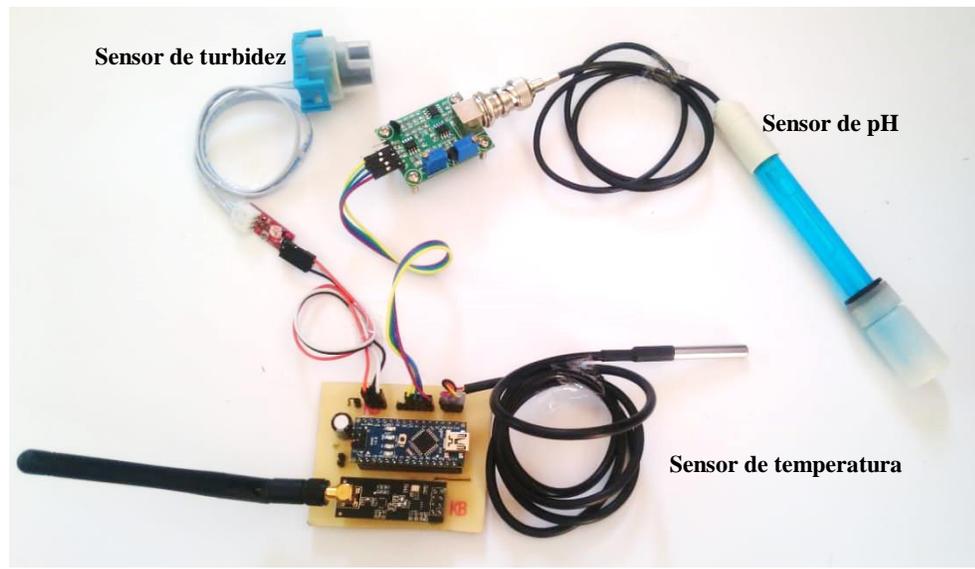


Fig. 4.44: Conexión de los componentes electrónicos al nodo sensor

Fuente: Investigador

Implementación de la red inalámbrica de sensores de monitoreo de la calidad del agua.

La implementación de la red inalámbrica de sensores de monitoreo de la calidad del agua de estques de truchas en el Criadero “EL CHAPARRAL”, se conecta los tres nodos sensores, el Arduino nano, la tarjeta Shield Ethernet, el Router y la Raspberry Pi 2, como se muestra en la figura 4.45.

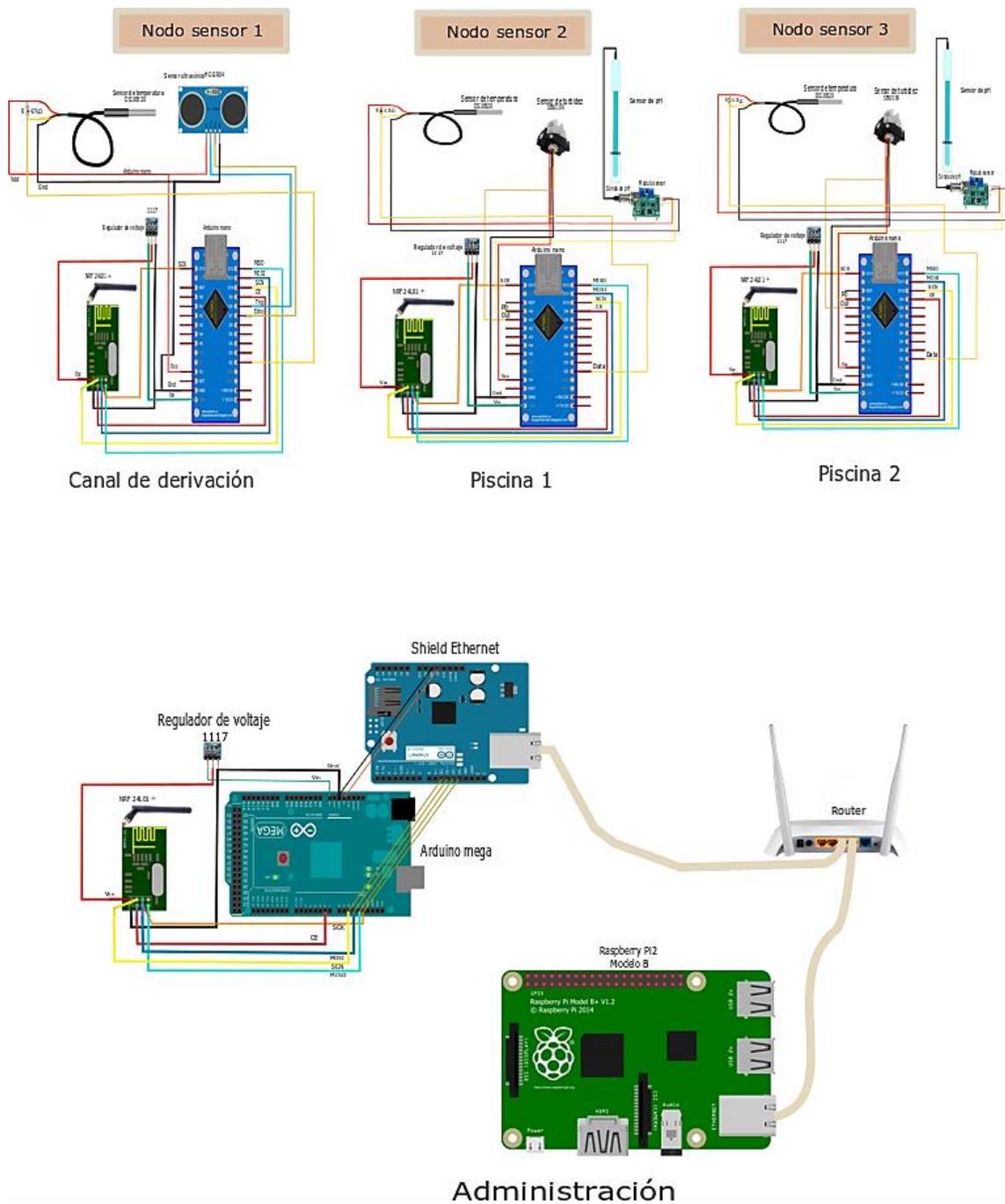


Fig. 4.45: Esquema eléctrico del sistema de monitoreo

Fuente: Investigador

4.15. Pruebas de los parámetros de la calidad del agua

Las pruebas de los parámetros de la calidad del agua se desarrollan de la siguiente manera:

4.15.1. Prueba de los valores de temperatura

Para la prueba de envío de los valores de temperatura se inicia el programa de Arduino que permite observar los datos obtenidos del sensor de temperatura. Las mediciones del prototipo y del termómetro se toman cada seis horas.

Tabla 4.12: Análisis de valores de temperatura

N°	Fecha / Hora	Termómetro Ferrara	Prototipo	Error Absoluto	Error Relativo
1	15-11-2018 / 06:00	13,16	13,66	0,50	3,80
2	15-11-2018 / 06:10	13,16	13,56	0,40	3,04
3	15-11-2018 / 06:20	13,41	13,89	0,48	3,58
4	15-11-2018 / 06:30	14,09	14,31	0,22	1,56
5	15-11-2018 / 06:40	14,09	14,31	0,22	1,56
6	15-11-2018 / 06:50	14,44	14,75	0,31	2,15
7	15-11-2018 / 07:00	15,04	14,61	0,43	2,86
8	15-11-2018 / 12:00	15,34	15,75	0,41	2,67
9	15-11-2018 / 12:10	15,34	15,69	0,35	2,28
10	15-11-2018 / 12:20	15,34	15,69	0,35	2,28
11	15-11-2018 / 12:30	15,35	15,75	0,40	2,61
12	15-11-2018 / 12:40	15,71	16,01	0,30	1,91
13	15-11-2018 / 12:50	15,81	16,52	0,71	4,49
14	15-11-2018 / 13:00	15,81	16,34	0,53	3,35
15	15-11-2018 / 18:00	15,73	16,75	1,02	6,48
16	15-11-2018 / 18:10	14,90	15,22	0,32	2,15
17	15-11-2018 / 18:20	14,75	15,13	0,38	2,58
18	15-11-2018 / 18:30	14,75	15,75	1,00	6,78
19	15-11-2018 / 18:40	14,88	15,69	0,81	5,44
20	15-11-2018 / 18:50	15,07	14,75	0,32	2,12
21	15-11-2018 / 19:00	15,11	14,13	0,98	6,49
				0,50	3,34

Fuente: Investigador

En la tabla 4.12 se analiza que el prototipo presenta un porcentaje de error en la medición de 3,34% en comparación al termómetro Ferrara, la medición presenta un error absoluto promedio de 0,50 por lo que el prototipo se aproxima al valor real obtenido del termómetro con un valor del 96,66% de fiabilidad con respecto a la temperatura.

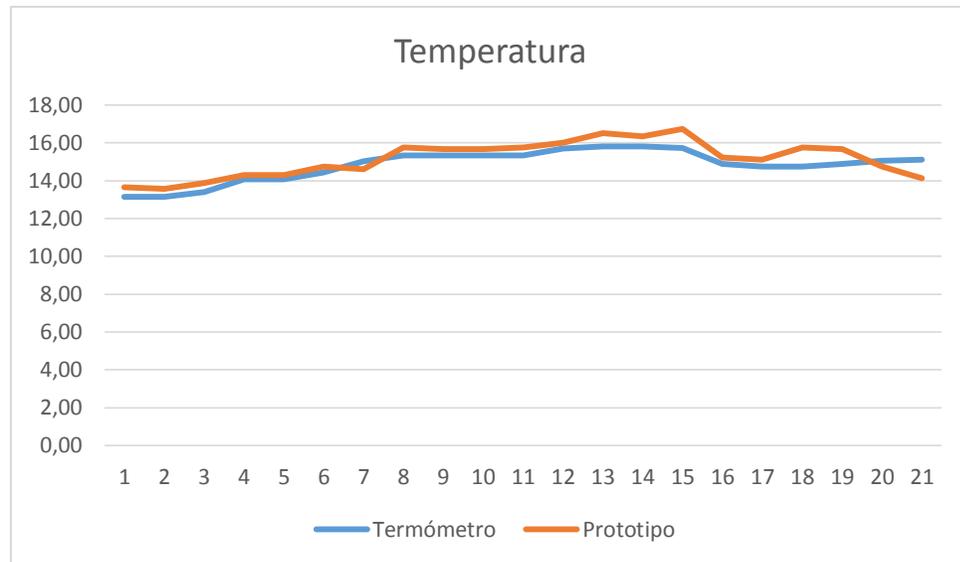


Fig. 4.46: Analisis de valores de temperatura

Fuente: Investigador

En la figura 4.46 se visualiza la comparación de los valores de temperatura obtenida del termómetro y del prototipo.

En la tabla 4.13 se analiza los valores promedios de temperatura monitoreados durante once semanas y en la figura 4.47 se visualiza los valores de temperatura obtenida del prototipo.

Tabla 4.13: Promedio de valores de temperatura

N°	Semanas	Prototipo
1	Semana 1	15,13
2	Semana 2	15,17
3	Semana 3	15,09
4	Semana 4	15,11
5	Semana 5	15,09
6	Semana 6	15,08
7	Semana 7	15,07
8	Semana 8	14,96
9	Semana 9	14,94
10	Semana 10	14,86
11	Semana 11	14,94

Fuente: Investigador

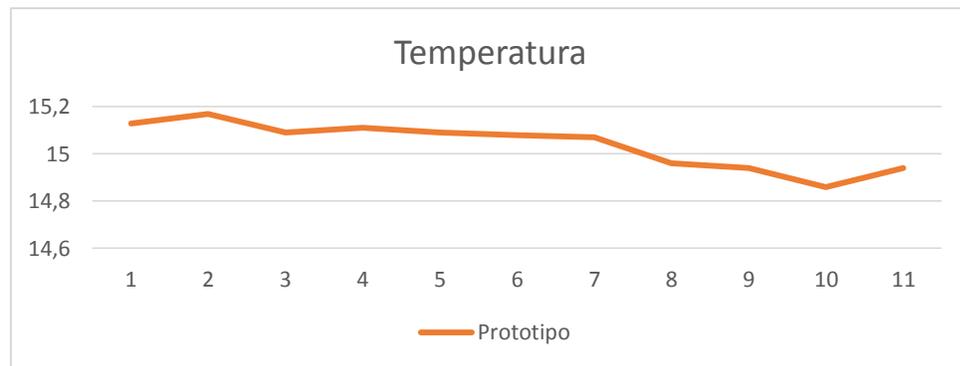


Fig. 4.47: Analisis de valores de temperatura

Fuente: Investigador

4.15.2. Prueba de los valores del pH

Para la prueba de envío de los valores del pH se ejecuta el programa desde Arduino, el mismo que permite observar los datos obtenidos del sensor de pH. Las mediciones del prototipo y del termómetro digital se realizan cada seis horas.

Tabla 4.14: Analisis de valores de pH

N°	Fecha/Hora	Tiras de papel de pH	Prototipo	Error Absoluto	Error Relativo
1	14-11-2018 / 06:00	6,00	7,06	1,06	17,67
2	14-11-2018 / 06:10	6,00	7,04	1,04	17,33
3	14-11-2018 / 06:20	6,00	7,08	1,08	18,00
4	14-11-2018 / 06:30	6,00	7,15	1,15	19,17
5	14-11-2018 / 06:40	7,00	7,25	0,25	3,57
6	14-11-2018 / 06:50	6,00	7,45	1,45	24,17
7	14-11-2018 / 07:00	6,00	7,46	1,46	24,33
8	14-11-2018 / 12:00	6,00	7,48	1,48	24,67
9	14-11-2018 / 12:10	6,00	7,15	1,15	19,17
10	14-11-2018 / 12:20	6,00	7,14	1,14	19,00
11	14-11-2018 / 12:30	6,00	7,15	1,15	19,17
12	14-11-2018 / 12:40	6,00	7,18	1,18	19,67
13	14-11-2018 / 12:50	7,00	7,53	0,53	7,57
14	14-11-2018 / 13:00	6,00	7,06	1,06	17,67
15	14-11-2018 / 18:00	7,00	7,48	0,48	6,86
16	14-11-2018 / 18:10	6,00	7,49	1,49	24,83
17	14-11-2018 / 18:20	6,00	7,24	1,24	20,67
18	14-11-2018 / 18:30	6,00	7,14	1,14	19,00
19	14-11-2018 / 18:40	6,00	7,00	1,00	16,67
20	14-11-2018 / 18:50	7,00	6,98	0,02	0,29
21	14-11-2018 / 19:00	7,00	6,98	0,02	0,29
				0,98	16,18

Fuente: Investigador

Los resultados de la tabla 4.14 se determina que el prototipo presenta un porcentaje de error en la medición de 16,18% debido a que los valores de las tiras de papel de pH son exactos y acarrea muchos decimales, la medición presenta un error absoluto promedio de 0,98 por lo que el prototipo se aproxima al valor real obtenido por las tiras de pH, con una fiabilidad de 83,82.

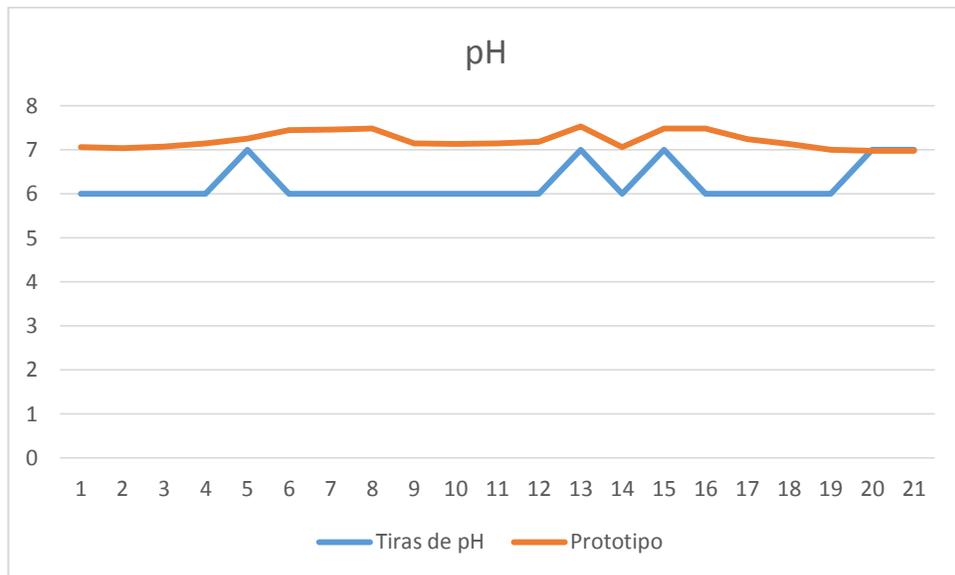


Fig. 4.48: Análisis de los valores de pH

Fuente: Investigador

En la figura 4.48 se visualiza la comparación de los valores de pH obtenidos por las tiras de papel de pH y del prototipo.

Una vez analizado los datos con el margen de error absoluto se procede a reducir el error en los valores de pH, para obtener datos más precisos se realiza el análisis entre dos puntos para obtener la fórmula de la recta, dando como resultado la ecuación (3) mediante esta se reduce el error a través de la modificación del algoritmo del software Arduino:

Tabla 4.15: Análisis de valores de pH

N°	Fecha/Hora	Tiras de pH	Prototipo	Error Absoluto	Error Relativo
1	28-11-2018 / 06:00	6,00	6,48	0,48	8,00
2	28-11-2018 / 06:10	6,00	6,48	0,48	8,00
3	28-11-2018 / 06:20	6,00	6,51	0,51	8,50
4	28-11-2018 / 06:30	6,00	6,48	0,48	8,00
5	28-11-2018 / 06:40	6,00	6,48	0,48	8,00
6	28-11-2018 / 06:50	6,00	6,48	0,48	8,00
7	28-11-2018 / 07:00	6,00	6,45	0,45	7,50
8	28-11-2018 / 12:00	6,00	6,45	0,45	7,50
9	28-11-2018 / 12:10	6,00	6,48	0,48	8,00
10	28-11-2018 / 12:20	6,00	6,51	0,51	8,50
11	28-11-2018 / 12:30	6,00	6,48	0,48	8,00
12	28-11-2018 / 12:40	6,00	6,48	0,48	8,00
13	28-11-2018 / 12:50	6,00	6,45	0,45	7,50
14	28-11-2018 / 13:00	6,00	6,42	0,42	7,00
15	28-11-2018 / 18:00	6,00	6,42	0,42	7,00
16	28-11-2018 / 18:10	6,00	6,45	0,45	7,50
17	28-11-2018 / 18:20	6,00	6,45	0,45	7,50
18	28-11-2018 / 18:30	6,00	6,51	0,51	8,50
19	28-11-2018 / 18:40	6,00	6,51	0,51	8,50
20	28-11-2018 / 18:50	6,00	6,42	0,42	7,00
21	28-11-2018 / 19:00	6,00	6,42	0,42	7,00
				0,47	7,79

Fuente: Investigador

Los resultados de la tabla 4.15 se determina que el prototipo presenta un porcentaje de error en la medición de 7,79% debido a los niveles definidos por las tiras de papel de pH, la medición presenta un error absoluto promedio de 0,47 por lo que el prototipo se aproxima al valor real obtenido por las tiras del pH, con una fiabilidad de 92,21.

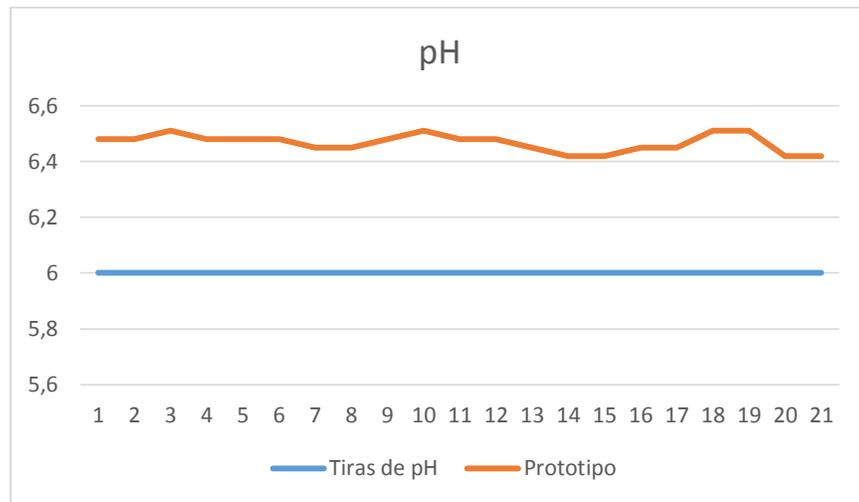


Fig. 4.49: Análisis de los valores de pH

Fuente: Investigador

En la figura 4.49 se visualiza la comparación de los valores de pH obtenidos de las tiras de papel de pH y del prototipo.

En la tabla 4.16 se analiza los valores promedios de pH monitoreados durante once semanas y en la figura 4.50 se visualiza los valores de temperatura obtenida del prototipo.

Tabla 4.16: Análisis de valores de pH

N°	Semanas	Prototipo
1	Semana 1	6,61
2	Semana 2	6,64
3	Semana 3	6,69
4	Semana 4	6,72
5	Semana 5	6,73
6	Semana 6	6,67
7	Semana 7	6,63
8	Semana 8	6,70
9	Semana 9	6,73
10	Semana 10	6,68
11	Semana 11	6,69

Fuente: Investigador

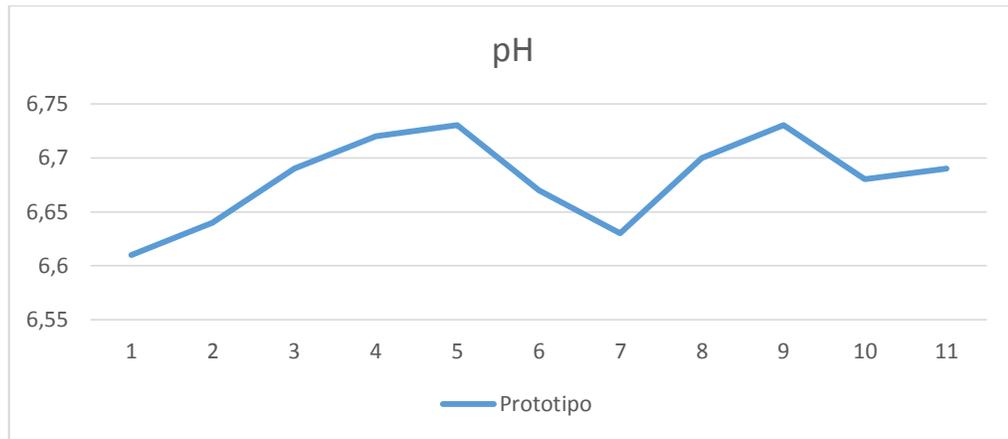


Fig. 4.50: Análisis de los valores de pH

Fuente: Investigador

4.15.3. Prueba de los valores de turbidez

Para la prueba de envío de los valores de turbidez, se ejecuta el programa desde Arduino que permite observar los datos obtenidos del sensor de turbidez, la medición de los valores se realizó cada seis horas, como se observa en la tabla 4.17

Tabla 4.17: Análisis de valores de turbidez

N°	Semanas	Prototipo
1	Semana 1	5,15
2	Semana 2	5,30
3	Semana 3	5,30
4	Semana 4	5,67
5	Semana 5	5,67
6	Semana 6	5,67
7	Semana 7	5,67
8	Semana 8	5,89
9	Semana 9	6,17
10	Semana 10	6,22
11	Semana 11	6,17

Fuente: Investigador

En la figura 4.51 se visualiza el comportamiento de los datos obtenidos por el sensor turbidez, no se puede realizar una comparación con un equipo homologado debido a que su costo es muy elevado y el presupuesto no abarca la compra de este dispositivo.

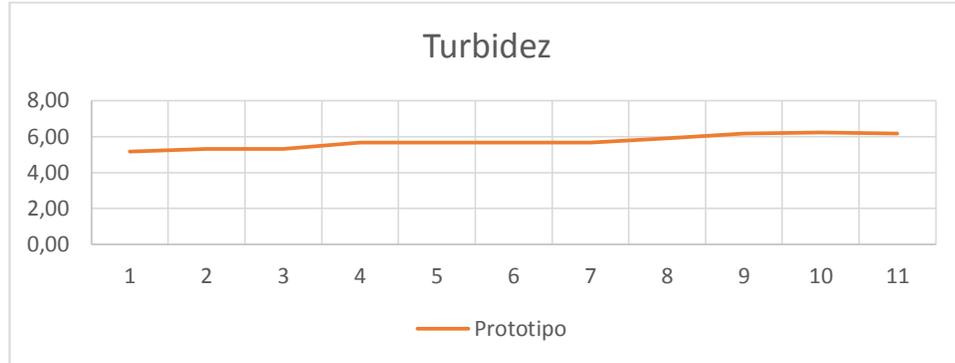


Fig. 4.51: Gráfica del comportamiento del sensor de turbidez

Fuente: Investigador

4.15.4. Prueba de los valores de caudal

Para la prueba de envío de los valores de caudal, se ejecuta el programa desde Arduino que permite observar los datos obtenidos del sensor ultrasónico, la medición de los valores se realiza cada quince minutos, como se observa en la tabla 4.18.

Tabla 4.18: Análisis de valores de caudal

N°	Semanas	Prototipo
1	Semana 1	15,51
2	Semana 2	15,52
3	Semana 3	15,52
4	Semana 4	15,50
5	Semana 5	15,51
6	Semana 6	15,51
7	Semana 7	15,73
8	Semana 8	15,55
9	Semana 9	15,57
10	Semana 10	15,60
11	Semana 11	15,60

Fuente: Investigador

En la figura 4.52 se visualiza el comportamiento de los datos obtenidos por el sensor ultrasónico, no se puede realizar una comparación con un equipo homologado debido a que su costo es muy elevado y el presupuesto no abarca la compra de este dispositivo.

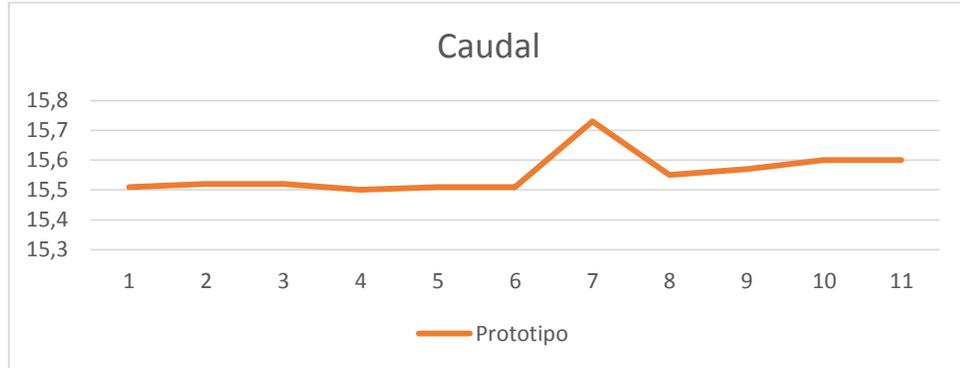


Fig. 4.52: Gráfica del comportamiento del sensor ultrasónico

Fuente: Investigador

Una vez analizado los datos del prototipo con otros equipos se realiza un margen de error general para obtener el promedio de error del sistema de monitoreo para estanques de truchas:

Tabla 4.19: Análisis de valores de turbidez

	Temperatura	pH	Total
Error Absoluto	0,50	0,47	0,49
Error Relativo	3,34	7,79	5,57
Fiabilidad			94,42

Fuente: Investigador

En la tabla 4.19 se obtuvo un promedio de error absoluto de 0,49, un promedio de error relativo de 5,57 con una fiabilidad del sistema de 94,42%.

4.16. Pruebas de comunicación

Las pruebas de comunicación se realizó entre el nodo maestro ubicado en la oficina del criadero y nodos esclavos ubicados en cada una de las piscinas de tuchas, la distancia de entre los dos puntos es de 122 metros como se observa en la figura 4.53 y 4.54.

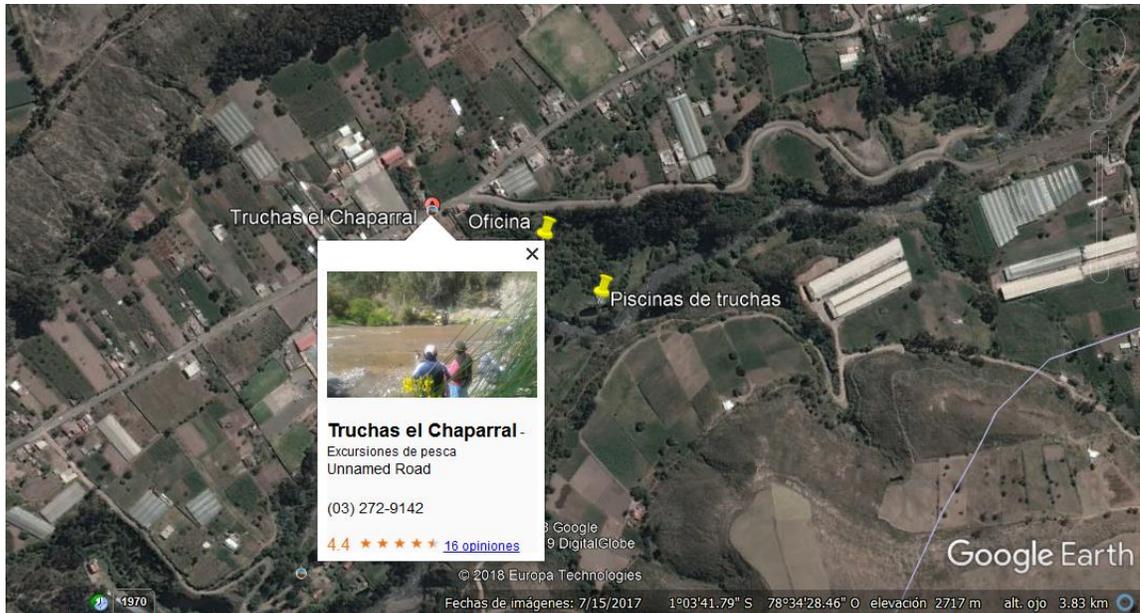


Fig. 4.53: Ubicación de cada punto de comunicación

Fuente: Google Earth



Fig. 4.54: Ruta de enlace

Fuente: Google Earth

El perfil topográfico del trayecto Oficina - Piscinas de truchas obtenido por Google Earth se observa en la figura 4.55, por la situación geográfica se determina que cuenta con las condiciones necesarias para realizar el enlace.



Fig. 4.55: Perfil de enlace

Fuente: Google Earth

4.17. Prueba de envío de información a la base de datos

En el archivo “enviodatos” se agrega el código correspondiente para enviar información al servidor. Con la Raspberry Pi y el Arduino conectados a la red por medio del Router, se realiza la prueba de envío de información comprobando que los datos enviados desde Arduino lleguen a la base de datos. La figura 4.56 muestra la prueba para el envío de información desde Arduino a la base de datos.

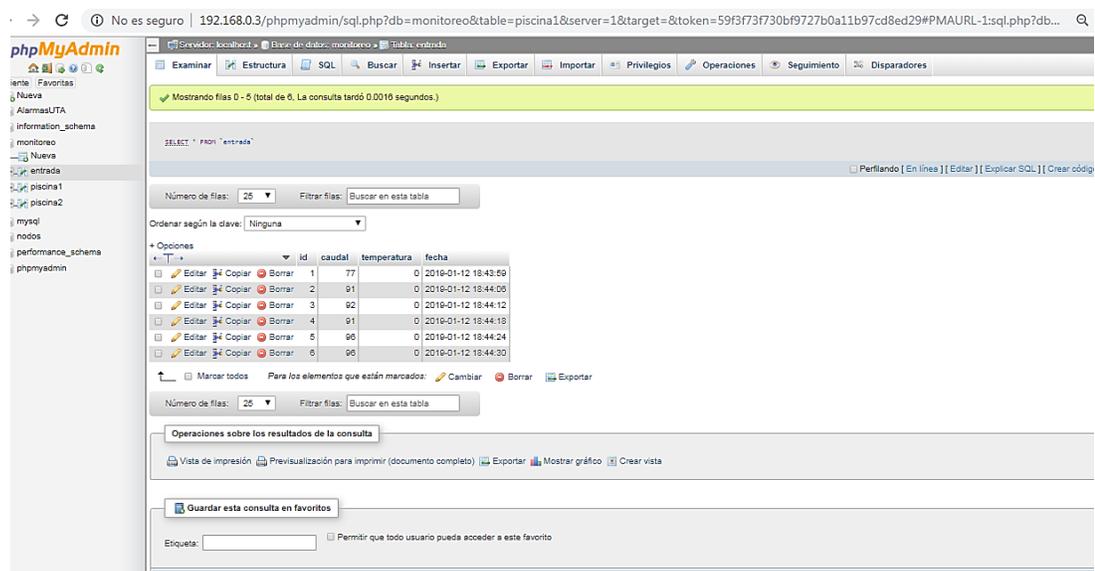


Fig. 4.56: Prueba del envío de la información desde Arduino a la base de datos

Fuente: Investigador

4.18. Generación del reporte

Para generar el reporte en un archivo PDF, se requiere descargar la librería FPDF desde la página www.fpdf.org, se debe establecer la conexión entre la base de datos con el servidor [65], por lo que se debe crear un script para la conexión llamando `conexion.php`:

```

1 <?php
2 //servidor, usuario de base de datos, contraseña del usuario, nombre de base de
3 datos
4 $mysqli = new mysqli("localhost","root","password","mexico");
5 if(mysqli_connect_errno()){
6     echo 'Conexion Fallida : ', mysqli_connect_error();
7     exit();
8 }
9 ?>

```

Luego, se crea una clase que contenga funciones de FPDF que es utilizada como plantilla para el encabezado y pie de página llamado “`formato.php`”

```

1 <?php
2     require 'fpdf/fpdf.php';
3     class PDF extends FPDF
4     {
5         function Header()
6         {
7             $this->Image('images/logochaparral.png', 5, 5, 30 );
8             $this->SetFont('Arial','B',15);
9             $this->Cell(30);
10            $this->Cell(120,10, ' REPORTE GENERAL DE
11 CALIDAD DEL AGUA ',0,0,'C');
12            $this->Ln(20);
13        }
14        function Footer() {
15            $this->SetY(-15);
16            $this->SetFont('Arial','T', 8);
17            $this->Cell(0,10, 'Pagina '.$this->PageNo().'/{nb}',0,0,'C'
18 );
19        }
20    }
21 ?>
22

```

Finalmente, se crea un script que permite crear el reporte en un archivo PDF.

```

1 <?php
2     require 'fpdf/fpdf.php';
3     class PDF extends FPDF
4     {
5         function Header()
6         {
7             $this->Image('images/logo.png', 5, 5, 30 );
8             $this->SetFont('Arial','B',15);
9             $this->Cell(30);
10            $this->Cell(120,10, ' REPORTE GENERAL DE
11 CALIDAD DEL AGUA ',0,0,'C');
12            $this->Ln(20);
13        }
14        function Footer() {
15            $this->SetY(-15);
16            $this->SetFont('Arial','T', 8);
17            $this->Cell(0,10, 'Pagina '.$this->PageNo().'/{nb}',0,0,'C' );
18        }
19    }
20 ?>

```

Para obtener el “REPORTE GENERAL DE CALIDAD DEL AGUA” que se indica en la figura 4.57, se genera desde la pantalla de monitoreo en el botón llamado “Reporte”, este archivo se descarga en formato PDF.

CALIDAD DEL AGUA

INICIO MONITOREO **REPORTE**



REPORTE GENERAL DE CALIDAD DEL AGUA

REPORTE DE PARAMETROS DE PISCINA 1

MUESTRA N.	PH	TEMPERATURA	TURBIDEZ	FECHA/HORA
1	6.7	14	45	2019-01-06 15:47:53
2	6.6	13.8	46	2019-01-06 15:48:15
3	6.9	13.8	45	2019-01-06 15:48:34
4	6.7	13.8	45	2019-01-06 15:48:49
5	6.8	14	45	2019-01-06 15:49:13
6	6.8	14	45	2019-01-06 15:49:34
7	6.6	14	45	2019-01-06 15:50:08

Firma del Responsable

Fig. 4.57: Reporte de la calidad del agua en archivo PDF

Fuente: Investigador

4.19. Presupuesto

El presupuesto para la implementación del prototipo red inalámbrica de sensores (WSN) de monitoreo de la calidad del agua para estanques de truchas, se divide en el presupuesto en diseño y construcción. El presupuesto del diseño se considera el total de las horas empleadas, por lo que se considera el análisis del salario básico de un

ingeniero en electrónica y Comunicaciones establecido por el Ministerio del Trabajo que corresponde a 858 dólares mensuales [66]. Se considera un promedio de 21 días laborables por año, mediante la ecuación (9) se obtiene el salario por día:

$$Salario_{diario} = \frac{Salario_{mensual}}{Dias_{laborables}} \quad (9)$$

$$Salario_{diario} = \frac{858}{21}$$

$$Salario_{diario} = 40.86 \text{ dólares}$$

Para obtener el valor de una hora de trabajo se utiliza la ecuación (10), en donde un día laboral está constituido por ocho horas.

$$Salario_{hora} = \frac{Salario_{diario}}{Horas_{laborables}} \quad (10)$$

$$Salario_{hora} = \frac{40.86}{8}$$

$$Salario_{hora} = 5.11 \text{ dólares}$$

Aproximadamente, se estima noventa horas de investigación empleadas para el diseño, simulación, pruebas de funcionamiento e implementación, se obtiene el presupuesto del proyecto de investigación indicado en la tabla 4.20:

Tabla 4.20: Presupuesto del diseño del sistema

Ítem	Descripción	Unidad	Cantidad	Valor Unitario	Valor Total
1	Diseño	Horas	90	5,11	459,90

Fuente: Investigador

En la tabla 4.21 se describe los dispositivos empleados en la construcción del prototipo.

Tabla 4.21: Presupuesto para la implementación del sistema

Ítem	Descripción	Unidad	Cantidad	Valor Unitario	Valor Total
1	Sensor de pH	1	2	46,43	92,86
2	Sensor de temperatura	°C	3	2,81	8,43
3	Sensor de turbidez	NTU	2	19,64	39,28
4	Arduino mega	1	1	23,00	23,00
5	Arduino nano	1	3	8,83	26,49
6	NRF 24L01	1	4	6,42	25,68
7	Raspberry Pi 2B+	1	1	32,00	32,00
8	Fuente 9V	V	1	6,00	6,00
9	Fuente 5V	V	4	5,00	20,00
10	Router LAN	1	1	22,00	22,00
11	Sensor ultrasónico	1	1	8,00	8,00
12	Baquelita	1	1	3,00	3,00
13	Reguladores de voltaje	V	4	1,50	6,00
14	Capacitores cerámicos 103	1	4	0,05	0,20
15	Potenciómetro 5K	Ohm	4	0,25	1,00
16	Diodo 1N4007	1	4	0,10	0,40
17	Capacitores 470uF - 16V	1	4	0,25	1,00
18	Cloruro férrico	1	2	0,70	1,40
19	Filas de espadines	1	7	0,35	2,45
20	Cables de energía eléctrica	1	4	2,10	8,40
21	Cables de red	1	3	2,00	6,00
22	Cables para protoboard	1	10	0,20	2,00
23	Zócalos	1	7	0,35	2,45
24	Resistencias 4.7kΩ	Ohm	3	0,15	0,45
25	Impresiones en papel couche	1	4	0,40	1,60
26	Case de plástico	1	3	3,50	10,50
				Subtotal	308,52
				Iva (12%)	42,07
				Total	350,59

Elaborador por: Investigador

Una vez realizado el presupuesto para el diseño y para la construcción se obtiene el presupuesto total en la tabla 4.22:

Tabla 4.22: Presupuesto total del proyecto de investigación

Ítem	Descripción	Unidad	Cantidad	Valor Unitario	Valor Total
1	Diseño	Horas	90	5,11	459,90
2	Construcción	Dólares	1	350,59	350,59
				Total	810,49

Fuente: Investigador

Finalmente, el presupuesto total del proyecto de investigación bajo el tema red inalámbrica de sensores (WSN) de monitoreo de la calidad del agua para estanques de truchas es de 810,49 dólares, este valor es bajo en comparación a otros a equipos homologados que solo realizan el monitoreo de pH, temperatura o turbidez que tienen un valor aproximado de 100 a 800 dólares; ya que no existe un equipo que realice el monitoreo de la calidad del agua de cuatro parámetros a la vez. Pero si se realiza la implementación de esta red inalámbrica de sensores en otro criadero de truchas el costo sería bajo debido a que el presupuesto de diseño ya no se considera y estaría disponible a un costo accesible en el país.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

- En el año 2003, el criadero de truchas el Chaparral realizó estudios previos al líquido del río Yanayacu para determinar la calidad del agua y la factibilidad de la construcción de los estanques de truchas, desde aquella fecha no se han ejecutado estudios posteriores sobre el análisis de los parámetros de la calidad del agua, lo cual es crucial para el desarrollo óptimo del pez. El presente proyecto determinó la importancia de contar con un sistema de monitoreo de la calidad del agua para prevenir al encargado del criadero sobre el nivel óptimo de temperatura, pH, turbidez y caudal permitiendo actuar rápidamente en las piscinas de crianza para la corrección de estos parámetros evitando que los peces sufran lesiones, lo cual permite obtener un producto de calidad.
- El análisis de los parámetros de la calidad de agua para el cultivo de peces se determina mediante las recomendaciones descritas por la FAO, que involucra aspectos fundamentales tales como temperatura, pH, turbidez y caudal que deben ser monitoreadas con el fin de obtener condiciones óptimas de higiene en cada uno de los estanques de truchas para prevenir la proliferación de bacterias, en el cual se tiene como resultado que la temperatura en un rango óptimo varía entre 14°C y 16°C para el crecimiento y desarrollo de las truchas, mientras que la turbidez se mantiene en un rango de 5 a 30 NTUs, el nivel de pH se encuentra en el rango óptimo de 6,60 a 7,90. Finalmente, el rango óptimo del caudal se encuentra entre 16 a 35 L/s para evitar

inundaciones en los estanques de truchas, estos parámetros se ven alterados en época lluviosa por el aumento de caudal en el río el cual arrastra sedimentos a las piscinas alterando el nivel de turbidez y afectando al resto de parámetros de la calidad del agua para estanques de truchas.

- Mediante dispositivos inalámbricos que operan en la banda de 2,4 GHz, plataformas de hardware libre, sensores y demás componentes electrónicos se implementó una WSN para recolectar los datos en forma individual de cada piscina de crianza de peces, además se implementó una interfaz gráfica de usuario donde el responsable monitorea, genera reportes del estado del agua que ingresa a los estanques de truchas. Se utilizó Google Apps – Gmail como servidor de correo electrónico como un sistema de alertas para el encargado del criadero cuando los valores de temperatura, pH, turbidez y caudal estén fuera del rango óptimo.

5.2. Recomendaciones

- El productor debe contar con toda la información de las buenas prácticas del manejo acuícola, como llevar un registro de la calidad del agua, a fin de que sirva como base para prevenir enfermedades que puedan afectar el desarrollo y crecimiento de las truchas, por lo que se recomienda se realice la implementación de un sistema de monitoreo, utilizando sensores apropiados y calibrados correctamente con modelos matemáticos incluso con soluciones y equipos certificados, para que la información proporcionada por cada uno de ellos sea correcta y de esta manera asegurar que la información almacenada en la base de datos, sea verídica debido a que el agua está en constante movimiento y relacionada con los cambios climáticos que sufre la fuente de agua.
- Se recomienda usar mallas en cada entrada de agua en cada piscina para evitar que hojas secas, pequeños palos ingresen al hábitat de los peces para disminuir el nivel de turbidez, además en el canal de derivación construir un estanque en el cual el agua de

ingresa del río Yanayacu repose con el objetivo de que impurezas del río reposen al interior del estanque y el agua ingrese al hábitat sea libre de partículas que pueden afectar la visión y branquias de las truchas.

- Se recomienda que para posteriores investigaciones se analice la utilización de un sensor de caudal para exteriores, el cual este apto a cambios climáticos, además utilizar filtros para el ruido cuya magnitud perjudica los sistemas de comunicaciones del prototipo.

Bibliografía

- [1] FAO, Organización de las Naciones Unidas para la Alimentación y la Agricultura, «Manual Práctico para el Cultivo de la Truchas Arcoíris,» 2014. [En línea]. Available: <http://www.fao.org/3/a-bc354s.pdf>. [Último acceso: 02 Octubre 2017].
- [2] MAGAP, Ministerio de Agricultura, Ganadería, Acuicultura y Pesca, «Estado actual y proyección de la acuicultura continental en el Ecuador,» 2014. [En línea]. Available: <http://acuicultura.espe.edu.ec/wp-content/uploads/2016/05/1-Estado-y-Proyecci%C3%B3n-de-la-Acuicultura-Ecuatoriana-Alejandro-de-la-Roche.pdf>. [Último acceso: 15 Noviembre 2017].
- [3] H. Muñoz y V. Vandeweerd, «PNUD América Latina y el Caribe: Una superpotencia de biodiversidad,» 2010. [En línea]. Available: http://www.latinamerica.undp.org/content/dam/undp/library/Environment%20and%20Energy/biodiversity/Latin-America-and-the-Caribbean---A-Biodiversity-Superpower--Policy_Brief_SPANISH.pdf. [Último acceso: 27 Octubre 2017].
- [4] República del Ecuador Consejo Nacional de Planificación, «Plan Nacional Para El Buen Vivir,» 2009. [En línea]. Available: http://www.planificacion.gob.ec/wp-content/uploads/downloads/2012/07/Plan_Nacional_para_el_Buen_Vivir.pdf. [Último acceso: 27 Octubre 2017].
- [5] G. Aquino, «Manual básico para el cultivo de trucha arco iris (*Oncorhynchus mykiss*),» Enero 2008. [En línea]. Available: https://www.researchgate.net/publication/320934783_Manual_basico_para_el_cultivo_de_trucha_arco_iris_Oncorhynchus_mykiss_GEM_TIES_Cuencas_Sanas_y_Modos_de_Vida_Sustentable_Series_de_Manuales_de_Capacitacion. [Último acceso: 21 Noviembre 2018].
- [6] D. He y L.-X. Zhang, «The Water Quality Monitoring System Based on WSN,» 17 Mayo 2012. [En línea]. Available: <http://ieeexplore.ieee.org/document/6201666/>. [Último acceso: 22 Noviembre 2017].
- [7] C. Gallardo, «Implementación de un sistema para el control y monitoreo de niveles de agua, temperatura y oxigenación de un proyecto piscícola en la crianza y reproducción de truchas en la Finca San Nicolás,» Marzo 2013. [En línea]. Available: <https://repositorio.espe.edu.ec/handle/21000/6151>. [Último acceso: 27 Noviembre 2017].

- [8] Milton Ríos, Ricardo Yauri, John K. Rojas, Viky C. Camarena, «Red de sensores inalámbricos para la medición,» 2014. [En línea]. Available: <https://cct-uni.pe/telcon/2014/files/papers/T-19.pdf>. [Último acceso: 14 Noviembre 2017].
- [9] Y. Chen y G. Hou, «WSN-based monitoring system for factory aquaculture,» 23 Noviembre 2014. [En línea]. Available: <http://ieeexplore.ieee.org/document/6933600/>. [Último acceso: 27 Noviembre 2017].
- [10] T. I. Salim, T. Haiyunnisa y H. S. Alam, «Design and implementation of water quality monitoring for eel fish aquaculture,» 27 Marzo 2016. [En línea]. Available: <http://ieeexplore.ieee.org/document/7886720/>. [Último acceso: 29 Noviembre 2017].
- [11] T. Kageyama, M. Miura, A. Maeda, A. Mori y S.-S. Lee, «A wireless sensor network platform for water quality monitoring,» 09 Enero 2016. [En línea]. Available: <http://ieeexplore.ieee.org/document/7808887/>. [Último acceso: 29 Noviembre 2017].
- [12] C. Encinas, E. Ruizy, J. Cortez y A. Espinoza, «Design and implementation of a distributed IoT system for the monitoring of water quality in aquaculture,» 8 Junio 2017. [En línea]. Available: <http://ieeexplore.ieee.org/document/7943540/>. [Último acceso: 29 Noviembre 2017].
- [13] F. Nuñez, «Sistema de monitoreo y control del alumbrado público mediante una red de sensores RF,» 2017.
- [14] Organización de las Naciones Unidas para la Alimentación y la Agricultura, «FAO Orientaciones Técnicas para la pesca responsable,» 2013. [En línea]. Available: <http://www.fao.org/3/a-i1917s.pdf>. [Último acceso: 12 Diciembre 2017].
- [15] Dirección Nacional de Recursos Acuáticos - Departamento de Acuicultura, 2010. [En línea]. Available: http://www.mgap.gub.uy/sites/default/files/multimedia/1959_manual.pdf. [Último acceso: 12 Diciembre 2017].
- [16] F. Balaguer, Industrias agrícolas: tratado de las que se explotan en España, Madrid: Eduardo Cuesta, 1977.
- [17] ILCE, «III Piscicultura,» Biblioteca General, [En línea]. Available: http://bibliotecadigital.ilce.edu.mx/sites/ciencia/volumen2/ciencia3/090/html/sec_7.html. [Último acceso: 12 Abril 2018].
- [18] FAO, «Piscicultura,» [En línea]. Available: <http://www.fao.org/docrep/field/009/ag196s/AG196S01.htm#ch1>. [Último acceso: 12 Abril 2018].

- [19] M. Calero y J. Villavicencio, «UDLA,» 2016. [En línea]. Available: <http://dspace.udla.edu.ec/bitstream/33000/6391/1/UDLA-EC-TIAG-2016-29.pdf>. [Último acceso: 13 Marzo 2018].
- [20] Eurostat, «Trucha *Oncorhynchus Mykiss*,» [En línea]. Available: https://ec.europa.eu/fisheries/sites/fisheries/files/docs/body/trout_es.pdf. [Último acceso: 12 Marzo 2018].
- [21] «Sierra y Selva exportadora,» [En línea]. Available: <http://www.sierraexportadora.gob.pe/wp-content/uploads/2017/01/FICHA-TRUCHA-fin.pdf>. [Último acceso: 12 Abril 2018].
- [22] J. García, F. Núñez y O. Chacón, «SCIELO,» 2014. [En línea]. Available: <http://www.scielo.org.mx/pdf/hbio/v14n1/v14n1a3.pdf>. [Último acceso: 12 Abril 2018].
- [23] FAO. [En línea]. Available: http://www.fao.org/fishery/static/FAO_Training/FAO_Training/General/x6708s/x6708s01.htm. [Último acceso: 10 Abril 2018].
- [24] S. Quast y O. Kimberger, «DRAGER,» Dragerwerk AG, 2016. [En línea]. Available: <https://www.draeger.com/Library/Content/t-core-bk-9101301-es-1604-1.pdf>. [Último acceso: 28 Diciembre 2017].
- [25] P. Ghezzi, J. Requejo y S. González, «FONDEPES Fondo Nacional de Desarrollo Pesquero,» Octubre 2014. [En línea]. Available: https://www.fondepes.gob.pe/src/manuales/MANUAL_TRUCHA.pdf. [Último acceso: 24 Abril 2018].
- [26] E. Tenelema, «Repositorio,» 01 Noviembre 2004. [En línea]. Available: <http://dspace.esPOCH.edu.ec/bitstream/123456789/6255/1/236T0240.pdf?fbclid=IwAR0zAuoQt7u9U3zLbHQAFpB0u8RR48YKgKBe9hVA0UrrdIG2gCrP9RFCBek>. [Último acceso: 28 Diciembre 2017].
- [27] T. Romero, «Asociación Cubana de Producción Animal (ACPA),» Abril 2009. [En línea]. Available: <http://www.actaf.co.cu/revistas/Revista%20ACPA/2009/REVISTA%2004/22%20CALIDAD%20AGUA.pdf>. [Último acceso: 08 Octubre 2018].
- [28] N. C. Oficial, «Requisitos de calidad del agua para diferentes usos,» [En línea]. Available: https://ciperchile.cl/pdfs/11-2013/norovirus/NCh1333-1978_Mod-1987.pdf. [Último acceso: 06 Octubre 2018].
- [29] L. Marcó, C. Azario y G. C., «Higiene y Sanidad Ambiental,» 2004. [En línea]. Available: [http://www.salud-publica.es/secciones/revista/revistaspdf/bc510156890491c_Hig.Sanid.Ambient.4.72-82\(2004\).pdf](http://www.salud-publica.es/secciones/revista/revistaspdf/bc510156890491c_Hig.Sanid.Ambient.4.72-82(2004).pdf). [Último acceso: 06 Octubre 2018].

- [30] M. Bello y M. Pino, «Medición de Presión y Caudal,» 2000. [En línea]. Available: <http://www2.inia.cl/medios/biblioteca/boletines/NR25635.pdf>. [Último acceso: 22 Abril 2018].
- [31] N. Instruments, «¿Qué es una Red de Sensores Inalámbricos?,» 2009. [En línea]. Available: <http://www.ni.com/white-paper/7142/es/>. [Último acceso: 11 Diciembre 2017].
- [32] C. Sebastián, «PFC Escuela Técnica Superior De Ingeniería Informática,» Septiembre 2010. [En línea]. Available: <https://riunet.upv.es/bitstream/handle/10251/8592/PFC%20-%20DESARROLLO%20DE%20APLICACIONES%20BASADAS%20EN%20WSN.pdf>. [Último acceso: 2018 Abril 15].
- [33] J. Garbarino, «Protocolos para redes inalámbricas de sensores,» 07 Noviembre 2011. [En línea]. Available: <http://materias.fi.uba.ar/7500/Garbarino.pdf>. [Último acceso: 16 Enero 2018].
- [34] «Wireless Sensor Network,» [En línea]. Available: <http://www.mfbarcell.es/conferencias/wsn.pdf>. [Último acceso: 16 Enero 2018].
- [35] N. Semiconductor, «Enhanced ShockBurst User Guide,» [En línea]. Available: https://devzone.nordicsemi.com/documentation/nrf51/4.3.0/html/group__esb__users__guide.html. [Último acceso: 04 Enero 2018].
- [36] I. Palacios, «Repositorio Universidad de Cuenca,» 2016. [En línea]. Available: <http://dspace.ucuenca.edu.ec/bitstream/123456789/24200/1/tesis.pdf>. [Último acceso: 04 Enero 2018].
- [37] «Ciencias informáticas,» EcuRed, [En línea]. Available: https://www.ecured.cu/Hardware_libre. [Último acceso: 31 Julio 2018].
- [38] A. Jossué, «Utilización de Plataformas de Hardware Libre y Computadoras de Placa Única para la Intrusión no Autorizada a Redes y Equipos,» 2015. [En línea]. Available: http://premios.eset-la.com/universitario/pdf/utilizacion_de_plataformas_de_hardware_libre_para_la_intrusion_no_authorized_a_redes.pdf. [Último acceso: 31 Julio 2018].
- [39] F. Hernando, «Open Hardware,» [En línea]. Available: http://jeuazarru.com/wp-content/uploads/2014/10/Open_Hardware.pdf. [Último acceso: 31 Julio 2018].
- [40] «Arduino,» 2018. [En línea]. Available: <https://store.arduino.cc/usa/arduino-nano>. [Último acceso: 30 Julio 2018].
- [41] «Arduino,» [En línea]. Available: <https://store.arduino.cc/usa/arduino-mega-2560-rev3>. [Último acceso: 31 Julio 2018].

- [42] ETech, «Raspberry Pi Una inversión en el futuro de la ingeniería,» RS Auras, 2012. [En línea]. Available: http://etech.designspark.info/ELE_0050_eTech%2010/ELE_0050_eTech_ES/pubData/source/ELE_0050_eTech_ES.pdf. [Último acceso: 01 Abril 2018].
- [43] DFRobot, «PH meter(SKU: SEN0161),» 27 Marzo 2018. [En línea]. Available: [https://www.dfrobot.com/wiki/index.php/PH_meter\(SKU:_SEN0161\)](https://www.dfrobot.com/wiki/index.php/PH_meter(SKU:_SEN0161)). [Último acceso: 28 Mayo 2018].
- [44] «Folleto Informativo Turbidez,» [En línea]. Available: https://www.waterboards.ca.gov/water_issues/programs/swamp/docs/cwt/guidance/3150sp.pdf. [Último acceso: 06 Diciembre 2018].
- [45] J. López, «Repositorio - Universidad de San Carlos de Guatemala,» Septiembre 2011. [En línea]. Available: http://biblioteca.usac.edu.gt/tesis/08/08_3300_C.pdf. [Último acceso: 14 Diciembre 2018].
- [46] L. Casillas, M. Ginesta y Ó. Pérez, «Base de datos,» Mayo 2005. [En línea]. Available: <http://www.uoc.edu/masters/oficiales/img/913.pdf>. [Último acceso: 04 Enero 2018].
- [47] «Oracle,» [En línea]. Available: <https://iessanvicente.com/colaboraciones/oracle.pdf>. [Último acceso: 16 Enero 2018].
- [48] S. Vallejos, «DB2 v.9.5,» [En línea]. Available: <http://exa.unne.edu.ar/informatica/SO/SofiaDB2.pdf>. [Último acceso: 16 Enero 2018].
- [49] R. Gracia, W. Calderon y J. Álvarez, Universidad de Córdoba, [En línea]. Available: <https://ingsistemasunicor.files.wordpress.com/2015/09/protocolo-smtp-simple-mail-transfer-protocol.pdf>. [Último acceso: 04 Enero 2018].
- [50] «FPDF Library,» FPDF, [En línea]. Available: <http://www.fpdf.org/>. [Último acceso: 02 Enero 2019].
- [51] Dfrobot, «Waterproof DS18B20 Digital Temperature Sensor (SKU:DFR0198),» 14 Julio 2017. [En línea]. Available: [https://www.dfrobot.com/wiki/index.php/Waterproof_DS18B20_Digital_Temperature_Sensor_\(SKU:DFR0198\)](https://www.dfrobot.com/wiki/index.php/Waterproof_DS18B20_Digital_Temperature_Sensor_(SKU:DFR0198)). [Último acceso: 28 Mayo 2018].
- [52] A. S. LLC, «PT-1000 Temperature Kit,» 10 Octubre 2017. [En línea]. Available: https://www.atlas-scientific.com/_files/_datasheets/_probe/PT-1000-probe.pdf. [Último acceso: 28 Mayo 2018].
- [53] Dfrobot, «PH composite electrode manual,» [En línea]. Available: <http://image.dfrobot.com/image/data/SEN0161/PH%20composite%20electrode%20manual.pdf>. [Último acceso: 28 Mayo 2018].

- [54] «Turbidez (JTU),» [En línea]. Available: <http://www.ucm.es/data/cont/docs/952-2015-02-14-turbidez%20f.pdf>. [Último acceso: 29 Mayo 2018].
- [55] Dfrobot, «Turbidity sensor SKU: SEN0189,» 25 Mayo 2017. [En línea]. Available: https://www.dfrobot.com/wiki/index.php/Turbidity_sensor_SKU:_SEN0189. [Último acceso: 28 Mayo 2018].
- [56] «LTH Electronics,» LTH_3851, [En línea]. Available: http://lth.co.uk/wp-content/uploads/2011/downloads/DS_S-AND-T-SERIES.pdf. [Último acceso: 28 Mayo 2018].
- [57] E. Freaks, «Ultrasonic Ranging Module HC - SR04,» Amazon AWS, [En línea]. Available: <https://halckemy.s3.amazonaws.com/.../jjrG8vF7jAwCZIXqO66S.pdf>. [Último acceso: 30 Mayo 2018].
- [58] D. KEYES, «Water Sensor Module User's Manual,» [En línea]. Available: http://www.fecegypt.com/uploads/dataSheet/1480850810_water.pdf. [Último acceso: 01 Junio 2018].
- [59] Newark, «Arduino Nano,» [En línea]. Available: <http://www.farnell.com/datasheets/1682238.pdf>. [Último acceso: 01 Junio 2018].
- [60] «NRF24L01+,» [En línea]. Available: <http://bibing.us.es/proyectos/abreproy/12159/fichero/3.pdf>. [Último acceso: 02 Junio 2018].
- [61] «Electrónicos Caldas,» [En línea]. Available: <https://www.electronicoscaldas.com/modulos-rf/452-modulo-bluetooth-hc-05.html>. [Último acceso: 02 Junio 2018].
- [62] «Raspberry Pi 2,» [En línea]. Available: <https://cdn-shop.adafruit.com/pdfs/raspberrypi2modelb.pdf>. [Último acceso: 01 Junio 2018].
- [63] S. X. Software, «Orange Pi 2G,» [En línea]. Available: <https://www.electronicdatasheets.com/datasheet/Orange%20Pi%202G-IOT%20User%20Manual.pdf>. [Último acceso: 02 Junio 2018].
- [64] «Raspbian,» [En línea]. Available: <http://www.raspbian.org/RaspbianDocumentation>. [Último acceso: 03 Junio 2018].

- [65] «Curso de PHP y MySQL 9: Reporte en PDF,» Códigos de programación, 22 Enero 2017. [En línea]. Available: <https://codigosdeprogramacion.com/2017/01/22/pdf-en-php-parte-3-reporte-en-pdf-con-mysql/>. [Último acceso: 02 Enero 2019].
- [66] D. Jimenez, «Universidad Técnica de Ambato,» April 2018. [En línea]. Available: <http://repositorio.uta.edu.ec/jspui/handle/123456789/27796>. [Último acceso: 05 Enero 2019].
- [67] iiap, [En línea]. Available: <http://www4.congreso.gob.pe/comisiones/1999/ciencia/cd/iiap/iiap1/TEXT002.htm>. [Último acceso: 11 Enero 2018].
- [68] «EVILNAPSIS,» Web Development, 24 Febrero 2017. [En línea]. Available: <https://evilnapsis.com/2017/02/24/enviar-emails-con-la-libreria-phpmailer-y-configurar-gmail/>. [Último acceso: 29 Octubre 2018].
- [69] D. Chaparro, «LAMP: Linux, Apache MySQL y PHP/Perl,» 05 Abril 2016. [En línea]. Available: <http://viejo.dchaparro.net/doc/lamp.pdf>. [Último acceso: 03 Junio 2018].
- [70] J. Sánchez y S. Román, «Universidad de Salamanca (España),» Abril 2013. [En línea]. Available: <http://hidrologia.usal.es/temas/Aforos.pdf>. [Último acceso: 29 Mayo 2018].

ANEXOS

Anexo A

En el Anexo A se indica las características técnicas del Arduino Nano



Arduino Nano



Product Overview

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.0) or ATmega168 (Arduino Nano 2.x). It has more or less the same functionality of the Arduino Duemilanove, but in a different package. It lacks only a DC power jack, and works with a Mini-B USB cable instead of a standard one. The Nano was designed and is being produced by Gravitech.

Index

Technical Specifications

Page 2

How to use Arduino
Programming Environment, Basic Tutorials

Page 6

Terms & Conditions

Page 7



radiospares **RADIONICS**





Technical Specification



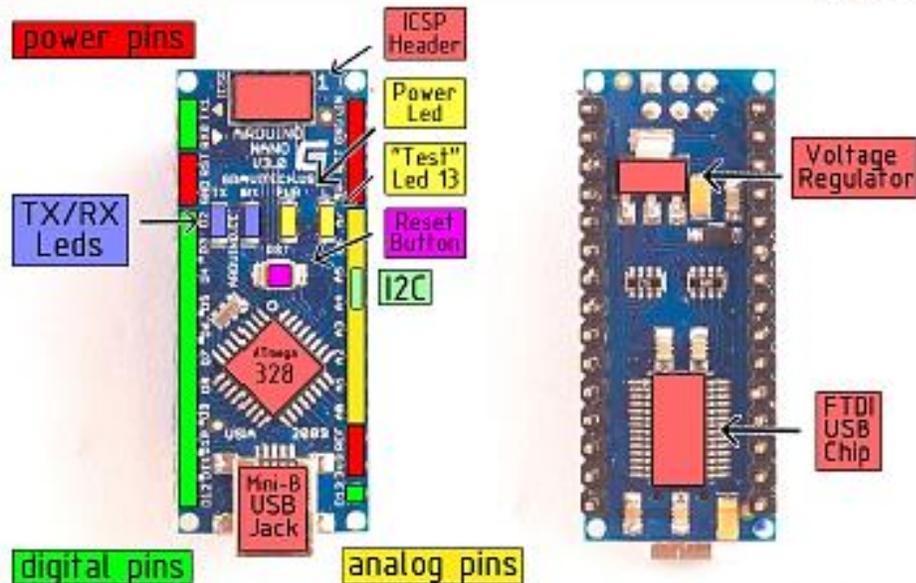
Arduino Nano 3.0 (ATmega328): [schematic](#), [Eagle files](#).

Arduino Nano 2.3 (ATmega168): [manual](#) (pdf), [Eagle files](#). Note: since the free version of Eagle does not handle more than 2 layers, and this version of the Nano is 4 layers, it is published here unrouted, so users can open and use it in the free version of Eagle.

Summary

Microcontroller	Atmel ATmega168 or ATmega328
Operating Voltage (logic level)	5 V
Input Voltage (recommended)	7-12 V
Input Voltage (limits)	6-20 V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	8
DC Current per I/O Pin	40 mA
Flash Memory	16 KB (ATmega168) or 32 KB (ATmega328) of which 2 KB used by bootloader
SRAM	1 KB (ATmega168) or 2 KB (ATmega328)
EEPROM	512 bytes (ATmega168) or 1 KB (ATmega328)
Clock Speed	16 MHz
Dimensions	0.73" x 1.70"

the board



radiospares

RADIONICS



Power

The Arduino Nano can be powered via the Mini-B USB connection, 6-20V unregulated external power supply (pin 30), or 5V regulated external power supply (pin 27). The power source is automatically selected to the highest voltage source.

The FTDI FT232RL chip on the Nano is only powered if the board is being powered over USB. As a result, when running on external (non-USB) power, the 3.3V output (which is supplied by the FTDI chip) is not available and the RX and TX LEDs will flicker if digital pins 0 or 1 are high.

Memory

The ATmega168 has 16 KB of flash memory for storing code (of which 2 KB is used for the bootloader); the ATmega328 has 32 KB, (also with 2 KB used for the bootloader). The ATmega168 has 1 KB of SRAM and 512 bytes of EEPROM (which can be read and written with the [EEPROM library](#)); the ATmega328 has 2 KB of SRAM and 1 KB of EEPROM.

Input and Output

Each of the 14 digital pins on the Nano can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial:** 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.
- **External Interrupts:** 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM:** 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI:** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- **LED:** 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Nano has 8 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **I²C:** 4 (SDA) and 5 (SCL). Support I²C (TWI) communication using the [Wire library](#) (documentation on the Wiring website).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega168 ports](#).



radiospares **RADIONICS**



Communication

The Arduino Nano has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega168 and ATmega328 provide UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An FTDI FT232RL on the board channels this serial communication over USB and the [FTDI drivers](#) (included with the Arduino software) provide a virtual com port to software on the computer. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the FTDI chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Nano's digital pins.

The ATmega168 and ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. To use the SPI communication, please see the ATmega168 or ATmega328 datasheet.

Programming

The Arduino Nano can be programmed with the Arduino software ([download](#)). Select "Arduino Diecimila, Duemilanove, or Nano w/ ATmega168" or "Arduino Duemilanove or Nano w/ ATmega328" from the Tools > Board menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega168 or ATmega328 on the Arduino Nano comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Nano is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the FT232RL is connected to the reset line of the ATmega168 or ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Nano is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Nano. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.



radiospares **RADIONICS**





How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](#) for the latest instructions. <http://arduino.cc/en/Guide/HomePage>

Linux Install

Windows Install

Mac Install

Once you have downloaded/unzipped the arduino IDE, you'll need to install the FTDI Drivers to let your PC talk to the board. **First Plug the Arduino to your PC via USB cable.**

Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

**File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink**

Once you have your skeeth you'll see something very close to the screenshot on the right.

In **Tools>Board** select **Arduino NANO** and with the AtMEGA you're using (probably 328)

Now you have to go to **Tools>SerialPort** and select the right serial port, the one arduino is attached to.

```

int ledPin = 13; // LED connected to digital pin 13

// The setup() method runs once, when the sketch starts

void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

void loop() {
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(1000);                // wait for a second
  digitalWrite(ledPin, LOW);  // set the LED off
  delay(1000);                // wait for a second
}

```



radiospares

RADIONICS



Anexo B

En el Anexo B se observa las características técnicas del Arduino Mega 2560



Product Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Index

Technical Specifications	Page 2
How to use Arduino Programming Enviroment, Basic Tutorials	Page 6
Terms & Conditions	Page 7
Enviromental Policies half sqm of green via Impatto Zero®	Page 7



radiospares

RADIONICS



Technical Specification

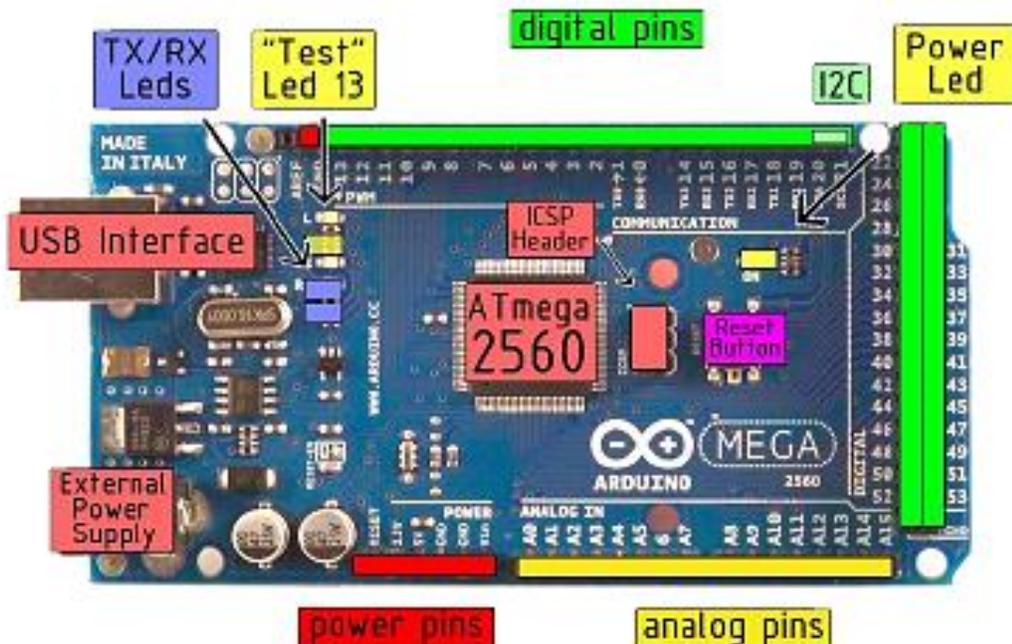


EAGLE files: [arduino-mega2560-reference-design.zip](#) Schematic: [arduino-mega2560-schematic.pdf](#)

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

the board



radiospares

RADIONICS



Power

The Arduino Mega2560 can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the ATmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 54 digital pins on the Mega can be used as an Input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial:** 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts:** 2 (Interrupt 0), 3 (Interrupt 1), 18 (Interrupt 6), 19 (Interrupt 4), 20 (Interrupt 3), and 21 (Interrupt 2). These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM:** 0 to 13. Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI:** 50 (MISO), 51 (MOSI), 52 (SS), 53 (SS). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Duemilanove and Diecimila.
- **LED:** 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- **I²C:** 20 (SDA) and 21 (SCL). Support I²C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I²C pins on the Duemilanove.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e., 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and [analogReference\(\)](#) function.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.



radiospares

RADIONICS



Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Mega's digital pins.

The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation on the Wiring website](#) for details. To use the SPI communication, please see the ATmega2560 datasheet.

Programming

The Arduino Mega2560 can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Arduino Mega comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.



radiospares **RADIONICS**



Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Mega has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics and Shield Compatibility

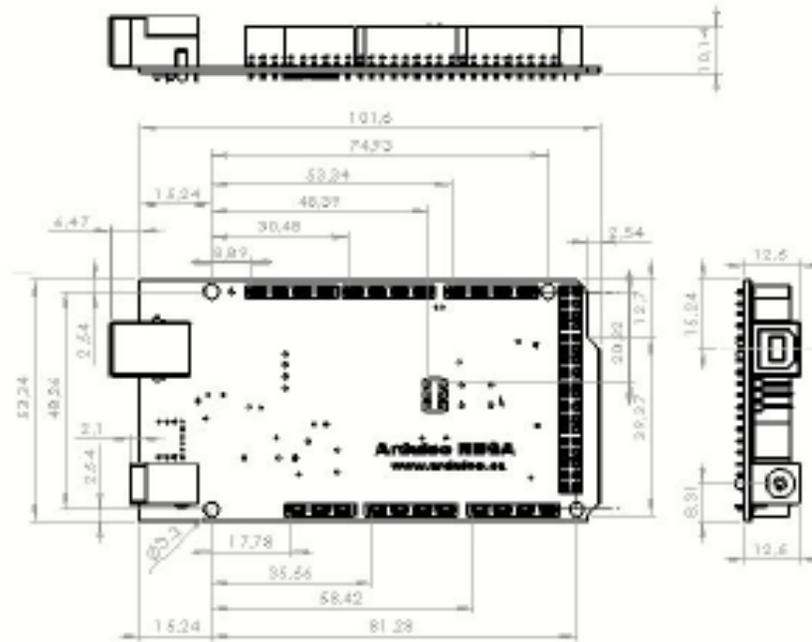
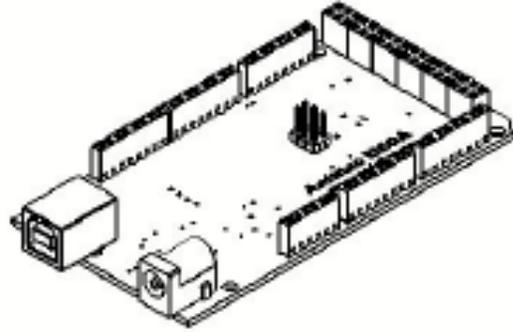
The maximum length and width of the Mega PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega is designed to be compatible with most shields designed for the Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega and Duemilanove / Diecimila. Please note that I²C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).



RADIOSPARES **RADIONICS**



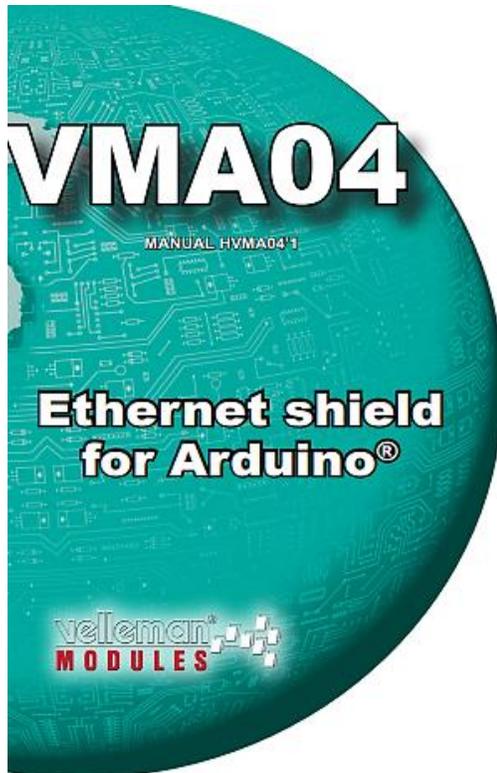


radiospares RADIONICS



Anexo C

En el Anexo C se indica las características técnicas de Ethernet Shield de Arduino



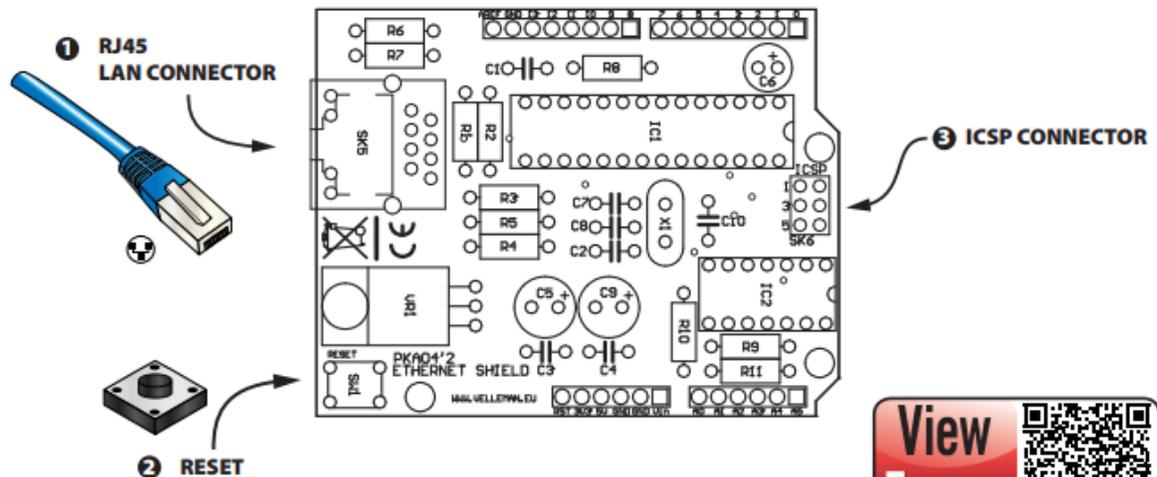
Configure your Arduino™ as a simple web server or let it get data from the worldwide web.

Features

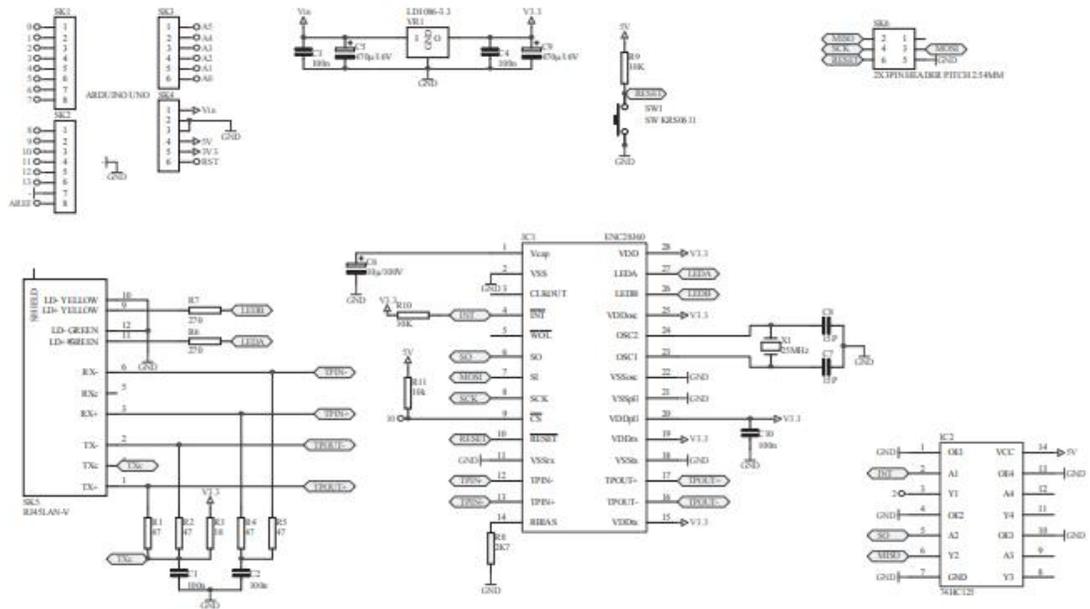
- For use with Arduino Uno™, Arduino Mega™
- Based on Microchip ENC28J60
- IEEE 802.3 compatible Ethernet controller
- Integrated MAC & 10BASE-T PHY
- SPI interface
- With RJ45 connector

Specifications

- Max. clock speed: 20MHz
- Transmit/receive buffer: 8kB
- Dimensions: 68 x 53mm / 2.67 x 2.08"



DOWNLOAD SAMPLE CODE FROM KA04 PAGE ON WWW.VELLEMAN.BE



Anexo D

En el Anexo D se indica las características técnicas Raspberry Pi 2 Model B



Raspberry Pi



Raspberry Pi 2, Model B

Product Name Raspberry Pi 2, Model B

Product Description The Raspberry Pi 2 delivers 6 times the processing capacity of previous models. This second generation Raspberry Pi has an upgraded Broadcom BCM2836 processor, which is a powerful ARM Cortex-A7 based quad-core processor that runs at 900MHz. The board also features an increase in memory capacity to 1Gbyte.

Specifications

Chip	Broadcom BCM2836 SoC
Core architecture	Quad-core ARM Cortex-A7
CPU	900 MHz
GPU	Dual Core VideoCore IV® Multimedia Co-Processor Provides Open GL ES 2.0, hardware-accelerated OpenVG, and 1080p30 H.264 high-profile decode Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure
Memory	1GB LPDDR2
Operating System	Boots from Micro SD card, running a version of the Linux operating system
Dimensions	85 x 56 x 17mm
Power	Micro USB socket 5V, 2A

Connectors:

Ethernet	10/100 BaseT Ethernet socket
Video Output	HDMI (rev 1.3 & 1.4)
Audio Output	3.5mm jack, HDMI
USB	4 x USB 2.0 Connector
GPIO Connector	40-pin 2.54 mm (100 mil) expansion header: 2x20 strip Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines
Camera Connector	15-pin MIPI Camera Serial Interface (CSI-2)
JTAG	Not populated
Display Connector	Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane
Memory Card Slot	Micro SDIO

Anexo E

En el Anexo E se indica las características técnicas del sensor de temperatura DS18B20

Click [here](#) for production status of specific part numbers.

DS18B20

Programmable Resolution 1-Wire Digital Thermometer

General Description

The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. In addition, the DS18B20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply.

Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus. Thus, it is simple to use one microprocessor to control many DS18B20s distributed over a large area. Applications that can benefit from this feature include HVAC environmental controls, temperature monitoring systems inside buildings, equipment, or machinery, and process monitoring and control systems.

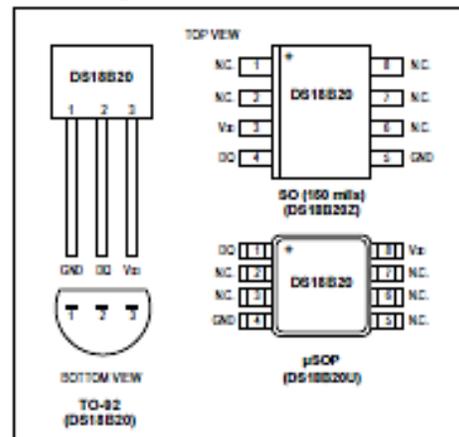
Applications

- Thermostatic Controls
- Industrial Systems
- Consumer Products
- Thermometers
- Thermally Sensitive Systems

Benefits and Features

- Unique 1-Wire[®] Interface Requires Only One Port Pin for Communication
- Reduce Component Count with Integrated Temperature Sensor and EEPROM
 - Measures Temperatures from -55°C to $+125^{\circ}\text{C}$ (-67°F to $+257^{\circ}\text{F}$)
 - $\pm 0.5^{\circ}\text{C}$ Accuracy from -10°C to $+85^{\circ}\text{C}$
 - Programmable Resolution from 9 Bits to 12 Bits
 - No External Components Required
- Parasitic Power Mode Requires Only 2 Pins for Operation (DQ and GND)
- Simplifies Distributed Temperature-Sensing Applications with Multidrop Capability
 - Each Device Has a Unique 64-Bit Serial Code Stored in On-Board ROM
- Flexible User-Definable Nonvolatile (NV) Alarm Settings with Alarm Search Command Identifies Devices with Temperatures Outside Programmed Limits
- Available in 8-Pin SO (150 mils), 8-Pin μSOP , and 3-Pin TO-92 Packages

Pin Configurations



[Ordering Information](#) appears at end of data sheet.

1-Wire is a registered trademark of Maxim Integrated Products, Inc.

Absolute Maximum Ratings

Voltage Range on Any Pin Relative to Ground.....	-0.5V to +6.0V	Storage Temperature Range.....	-55°C to +125°C
Operating Temperature Range.....	-55°C to +125°C	Solder Temperature.....	Refer to the IPC/JEDEC J-STD-020 Specification.

These are stress ratings only and functional operation of the device at these or any other conditions above those indicated in the operation section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

DC Electrical Characteristics

(-55°C to +125°C; $V_{DD} = 3.0V$ to $5.5V$)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS	
Supply Voltage	V_{DD}	Local power (Note 1)	+3.0		+5.5	V	
Pullup Supply Voltage	V_{PU}	Parasite power	+3.0		+5.5	V	
		Local power					V_{DD}
Thermometer Error	t_{ERR}	-10°C to +85°C			±0.5	°C	
		-30°C to +100°C					±1
		-55°C to +125°C					±2
Input Logic-Low	V_{IL}	(Notes 1, 4, 5)	-0.3		+0.8	V	
Input Logic-High	V_{IH}	Local power	+2.2		The lower of 5.5 or	V	
		Parasite power	+3.0		$V_{DD} + 0.3$		
Sink Current	I_L	$V_{DD} = 0.4V$	4.0			mA	
Standby Current	I_{DDS}	(Notes 7, 8)		750	1000	nA	
Active Current	I_{DD}	$V_{DD} = 5V$ (Note 9)		1	1.5	mA	
DQ Input Current	I_{DD}	(Note 10)		5		µA	
Drift		(Note 11)		±0.2		°C	

Note 1: All voltages are referenced to ground.

Note 2: The Pullup Supply Voltage specification assumes that the pullup device is ideal, and therefore the high level of the pullup is equal to V_{PU} . In order to meet the V_{IH} spec of the DS18B20, the actual supply rail for the strong pullup transistor must include margin for the voltage drop across the transistor when it is turned on; thus: $V_{PU_ACTUAL} = V_{PU_IDEAL} + V_{Vt(transistor)}$.

Note 3: See typical performance curve in Figure 1. Thermometer Error limits are 3-sigma values.

Note 4: Logic-low voltages are specified at a sink current of 4mA.

Note 5: To guarantee a presence pulse under low voltage parasite power conditions, V_{ILMAX} may have to be reduced to as low as 0.5V.

Note 6: Logic-high voltages are specified at a source current of 1mA.

Note 7: Standby current specified up to +70°C. Standby current typically is 3µA at +125°C.

Note 8: To minimize I_{DD} , DQ should be within the following ranges: $GND \leq DQ \leq GND + 0.3V$ or $V_{DD} - 0.3V \leq DQ \leq V_{DD}$.

Note 9: Active current refers to supply current during active temperature conversions or EEPROM writes.

Note 10: DQ line is high ("high-Z" state).

Note 11: Drift data is based on a 1000-hour stress test at +125°C with $V_{DD} = 5.5V$.

AC Electrical Characteristics—NV Memory(-55°C to +125°C; $V_{DD} = 3.0V$ to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
NV Write Cycle Time	t_{WR}			2	10	ms
EEPROM Writes	N_{EEWR}	-55°C to +55°C	50k			writes
EEPROM Data Retention	t_{EEDR}	-55°C to +55°C	10			years

AC Electrical Characteristics(-55°C to +125°C; $V_{DD} = 3.0V$ to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Temperature Conversion Time	t_{CONV}	9-bit resolution			93.75	ms
		10-bit resolution	(Note 12)		187.5	
		11-bit resolution			375	
		12-bit resolution			750	
Time to Strong Pullup On	t_{SPON}	Start convert T command issued			10	μs
Time Slot	t_{SLOT}	(Note 12)	60		120	μs
Recovery Time	t_{REC}	(Note 12)	1			μs
Write 0 Low Time	t_{LOW0}	(Note 12)	60		120	μs
Write 1 Low Time	t_{LOW1}	(Note 12)	1		15	μs
Read Data Valid	t_{RDV}	(Note 12)			15	μs
Reset Time High	t_{RSTH}	(Note 12)	480			μs
Reset Time Low	t_{RSTL}	(Notes 12, 13)	480			μs
Presence-Detect High	t_{PDHGH}	(Note 12)	15		60	μs
Presence-Detect Low	t_{PDLOW}	(Note 12)	60		240	μs
Capacitance	C_{INDUT}				25	μF

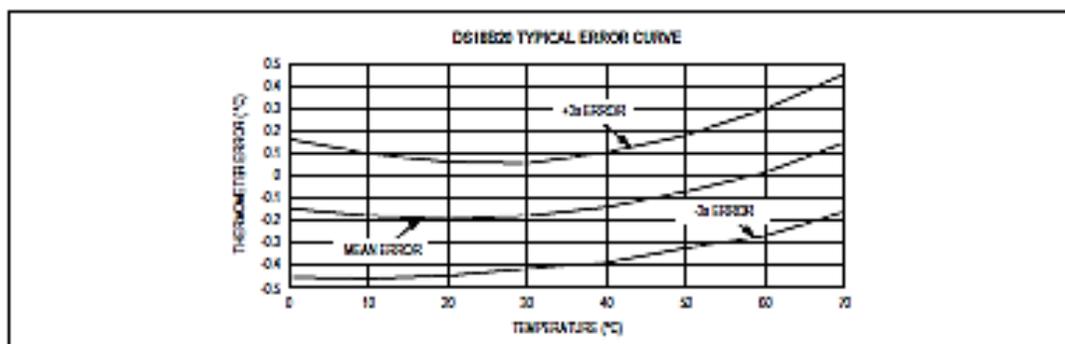
Note 12: See the timing diagrams in [Figure 2](#).Note 13: Under parasite power, if $t_{RSTL} > 960\mu s$, a power-on reset can occur.

Figure 1. Typical Performance Curve

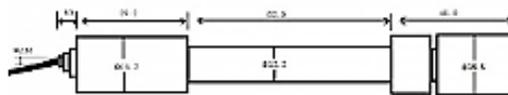
Anexo F

En el Anexo F se detalla las características técnicas del sensor de pH SEN161

Introduction

Need to measure water quality and other parameters but haven't got any low cost pH meter? Find it difficult to use with Arduino? Here comes an analog pH meter, specially designed for Arduino controllers and has built-in simple, convenient and practical connection and features. It has an LED which works as the Power Indicator, a BNC connector and PH2.0 sensor interface. You can just connect the pH sensor with BNC connector, and plug the PH2.0 interface into any analog input on Arduino controller to read pH value easily.

Specification



SEN161 dimension

- Module Power: 5.00V
- Circuit Board Size: 43mm×32mm
- pH Measuring Range: 0-14
- Measuring Temperature: 0-60 °C
- Accuracy: ± 0.1 pH (25 °C)
- Response Time: ≤ 1 min
- pH Sensor with BNC Connector
- PH2.0 Interface (3 foot patch)
- Gain Adjustment Potentiometer
- Power Indicator LED

Precautions

- Before and after use of the pH electrode every time, you need to use (pure)water to clean it.
- The electrode plug should be kept clean and dry in case of short circuit.
- **Preservation:** Electrode reference preservation solution is the 3N KCL solution.
- Measurement should be avoided staggered pollution between solutions, so as not to affect the accuracy of measurement.
- Electrode blub or sand core is defiled which will make PTS decline, slow response. So, it should be based on the characteristics of the pollutant, adapted to the cleaning solution, the electrode performance recovery.

- Electrode when in use, the ceramic sand core and liquid outlet rubber ring should be removed, in order to make salt bridge solution to maintain a certain velocity.

NOTE: Differences between the probes, SEN0161 and SEN0169

Their usages/ specifications are almost the same. The differences locates at

Long-firing Operation: SEN0169 supports, while SEN0161 NOT, i.e. you can not immerse SEN0161 in water for Continuous Testing.

Life Span: In 25 °C, pure water, do Continuous Testing with them both, SEN0169 can work two years, while SEN0161 can only last for 6 months. And just for reference, if put them in turbid, strongly acid and alkali solution, 25°C, the life span would drop to one year (SEN0169), 1 month(or shorter, SEN0161).

Temperature, pH, turbidity of the water effect the probe life span a lot.

Waterproof: You can immerse the whole probe SEN0169 into the water, while you can only immerse the front part of the probe SEN0161, the electrode glass bulb, into water, the rear part, from the white shell to the cable, MUST NOT be under water.

Strongly Acid and Alkali: SEN0169 are preferred for strongly acid and alkali test. And if your testing range is usually within pH6~8, then SEN0161 is capable for that.

pH Electrode Characteristics

The output of pH electrode is Millivolts, and the pH value of the relationship is shown as follows (25 °C):

VOLTAGE (mV)	pH value	VOLTAGE (mV)	pH value
414.12	0.00	-414.12	14.00
354.96	1.00	-354.96	13.00
295.80	2.00	-295.80	12.00
236.64	3.00	-236.64	11.00
177.48	4.00	-177.48	10.00
118.32	5.00	-118.32	9.00
59.16	6.00	-59.16	8.00
0.00	7.00	0.00	7.00

Anexo G

En el Anexo G se indica las características técnicas del sensor de turbidez



Turbidity sensor SKU: SEN0189



Contents

- [1 Introduction](#)
- [2 Specification](#)
- [3 Connection Diagram](#)
- [4 Examples](#)

Introduction

The turbidity sensor detects water quality by measuring the levels of turbidity. It uses light to detect suspended particles in water by measuring the light transmittance and scattering rate, which changes with the amount of total suspended solids (TSS) in water. As the TSS increases, the liquid turbidity level increases.

Turbidity sensors are used to measure water quality in rivers and streams, wastewater and effluent measurements, control instrumentation for settling ponds, sediment transport research and laboratory measurements.

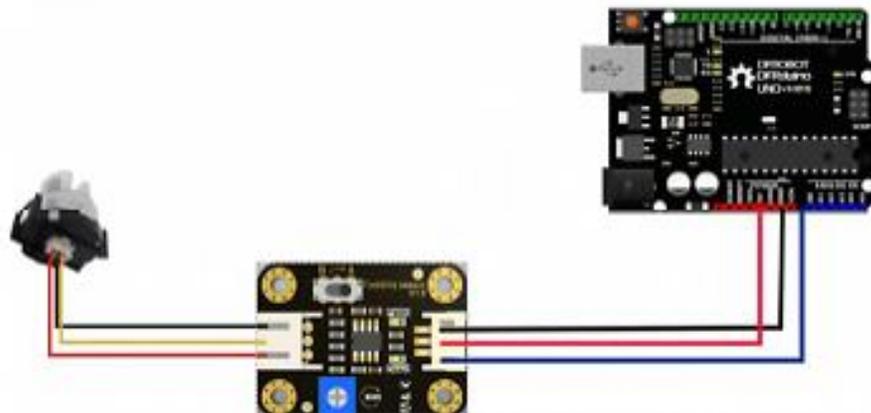
This sensor provides analog and digital signal output modes. The threshold is adjustable when in digital signal mode. You can select the mode according to your MCU.

 Note: The top of probe is not waterproof.

Specification

- Operating Voltage: 5V DC
- Operating Current: 40mA (MAX)
- Response Time : $\sim 500\text{ms}$
- Insulation Resistance: 100M (Min)
- Output Method:
 - Analog output: 0-4.5V
 - Digital Output: High/Low level signal (you can adjust the threshold value by adjusting the potentiometer)
- Operating Temperature: 5°C~90°C
- Storage Temperature: -10°C~90°C
- Weight: 30g
- Adapter Dimensions: 38mm*28mm*10mm/1.5Inches *1.1Inches*0.4Inches

Connection Diagram



Anexo H

En el Anexo H se observa las características técnicas del sensor ultrasónico



SENSOR ULTRASONIDO

Los sensores ultrasónicos usan sonar para determinar la distancia de un objeto como los murciélagos o delfines lo hacen, ofrecen un excelente rango de detección sin contacto, su operación no es afectada por la luz del sol o materiales oscuros, aunque materiales acústicamente blandos son difíciles de detectar.

Características:

Voltaje de alimentación: +5VDC

Corriente en espera: <2mA

Corriente de trabajo: 15mA

Ángulo eficaz: <15°

Rangos de distancia: 2cm a 400cm ó 1" a 13ft

Resolución: 0.3cm

Ángulo de medida: 30°

Ancho de pulso de disparo (Trigger Input Pulse Width): 10us se aplica pulso de (10us en Trig)

Eco (Echo): salida del sensor.

Frecuencia de ultrasonido: 40KHz

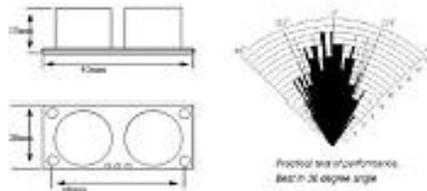
PINES DE CONEXION	
	<p>VCC = +5VDC</p> <p>Trig = Trigger input of Sensor, entrada del pulso de disparo para iniciar la medición.</p> <p>Echo = Echo output of Sensor, pin de eco, que retorna un pulso proporcional a la distancia que rebota el sonido.</p> <p>GND = GND</p>

E-Mail: info@microkitselectronica.com - www.microkitselectronica.com

Calle: 41 No. 51 15, local 141, Teléfono: 232 77 84,

Centro Comercial Paseo Bolívar, Carabobo con los Huesos, Medellín, Antioquia, Colombia.

Dimensiones y ángulos de desempeño:



Operación

Para iniciar la medición el pin Trig, el de la señal de disparo (trigger) debe recibir un pulso de cinco voltios por al menos diez micro segundos, esto indicará al sensor que transmita una ráfaga de ocho ciclos ultrasónicos a 40KHz y espere por la ráfaga reflejada cuando el sensor detecta la señal de ultrasonido en el receptor enviará una señal en alto (5v) por el pin Echo, esta señal tendrá un retardo (ancho) proporcional a la distancia, así que para obtener la distancia se debe medir el Ton (tiempo en alto en el pin Echo)



Tiempo que tarda la señal en volver es igual al ancho del pulso en el pin Echo en microsegundos, para determinar la distancia a partir de este tiempo:

$$\text{Distancia en centímetros} = \frac{\text{tiempo}}{58}$$

$$\text{Distancia en pulgadas} = \frac{\text{tiempo}}{148}$$

O se puede utilizar la velocidad del sonido, la cual es 340m/s

Anexo I

En el Anexo I se detalla las características técnicas del módulo de comunicación 24L01+



nRF24L01+

Single Chip 2.4GHz Transceiver

Preliminary Product Specification v1.0

Key Features

- Worldwide 2.4GHz ISM band operation
- 250kbps, 1Mbps and 2Mbps on air data rates
- Ultra low power operation
- 11.3mA TX at 0dBm output power
- 13.5mA RX at 2Mbps air data rate
- 900nA in power down
- 26µA in standby-I
- On chip voltage regulator
- 1.9 to 3.6V supply range
- Enhanced ShockBurst™
- Automatic packet handling
- Auto packet transaction handling
- 6 data pipe MultiCeiver™
- Drop-in compatibility with nRF24L01
- On-air compatible in 250kbps and 1Mbps with nRF2401A, nRF2402, nRF24E1 and nRF24E2
- Low cost BOM
- ±80ppm 16MHz crystal
- 5V tolerant inputs
- Compact 20-pin 4x4mm QFN package

Applications

- Wireless PC Peripherals
- Mouse, keyboards and remotes
- 3-in-1 desktop bundles
- Advanced Media center remote controls
- VoIP headsets
- Game controllers
- Sports watches and sensors
- RF remote controls for consumer electronics
- Home and commercial automation
- Ultra low power sensor networks
- Active RFID
- Asset tracking systems
- Toys

1.2 Block diagram

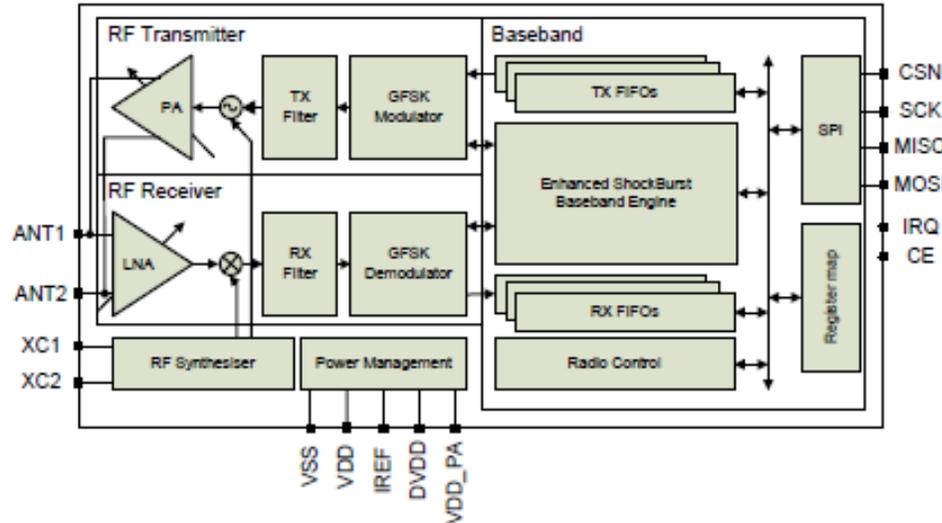


Figure 1. nRF24L01+ block diagram

2.2 Pin functions

Pin	Name	Pin function	Description
1	CE	Digital Input	Chip Enable Activates RX or TX mode
2	CSN	Digital Input	SPI Chip Select
3	SCK	Digital Input	SPI Clock
4	MOSI	Digital Input	SPI Slave Data Input
5	MISO	Digital Output	SPI Slave Data Output, with tri-state option
6	IRQ	Digital Output	Maskable interrupt pin. Active low
7	VDD	Power	Power Supply (+1.9V - +3.6V DC)
8	VSS	Power	Ground (0V)
9	XC2	Analog Output	Crystal Pin 2
10	XC1	Analog Input	Crystal Pin 1
11	VDD_PA	Power Output	Power Supply Output (+1.8V) for the internal nRF24L01+ Power Amplifier. Must be connected to ANT1 and ANT2 as shown in Figure 29 .
12	ANT1	RF	Antenna interface 1
13	ANT2	RF	Antenna interface 2
14	VSS	Power	Ground (0V)
15	VDD	Power	Power Supply (+1.9V - +3.6V DC)
16	IREF	Analog Input	Reference current. Connect a 22kΩ resistor to ground. See Figure 29 .
17	VSS	Power	Ground (0V)
18	VDD	Power	Power Supply (+1.9V - +3.6V DC)
19	DVDD	Power Output	Internal digital supply output for de-coupling purposes. See Figure 29 .
20	VSS	Power	Ground (0V)

Table 1. nRF24L01+ pin function

Anexo J

En el Anexo J se detalla el algoritmo para el prototipo del nodo 1

```
#include <RF24Network.h>
#include <RF24.h>
#include <SPI.h>
#include <OneWire.h>
#include <DallasTemperature.h>

RF24 radio(9,10);          // nRF24L01(+) radio attached using Getting Started
board

RF24Network network(radio);    // Network uses that radio

int id=1;
const uint16_t this_node = 01;    // Address of our node in Octal format
const uint16_t other_node = 00;    // Address of the other node in Octal format

const unsigned long interval = 2000; //ms // How often to send 'hello world to the other
unit

unsigned long last_sent;          // When did we last send?
unsigned long packets_sent;       // How many have we sent already
const int pinDatosDQ = 8;
  OneWire oneWireObjeto(pinDatosDQ);
  DallasTemperature sensorDS18B20(&oneWireObjeto);
  float ntu;

struct payload_t {                // Structure of our payload
  float ph;
  float temperatura;
  float oxigeno;
  float caudal;
  float id;
};
void setup(void)
{
```

```

Serial.begin(57600);
  sensorDS18B20.begin();
Serial.println("RF24Network/examples/helloworld_tx/");
SPI.begin();
radio.begin();
network.begin(/*channel*/ 90, /*node address*/ this_node);
radio.setDataRate(RF24_250KBPS);

pinMode(A0, INPUT);
pinMode(A1, INPUT);
pinMode(A2, INPUT);
}

void loop() {

  network.update();

  float ph=leer_ph();
  float temperatura=leer_temperatura();
  float oxigeno=leer_turbidez();
  float caudal=0;
  delay(10);
  enviar(ph, temperatura, oxigeno, caudal, id);
  delay(5000);
}

void enviar(float ph, float temperatura, float oxigeno, float caudal, int id)
{

  Serial.print("Sending...");
  delay(10);
  payload_t payload = { ph, temperatura, oxigeno, caudal, id};
  RF24NetworkHeader header(/*to node*/ other_node);
  bool ok = network.write(header,&payload,sizeof(payload));
  if (ok)
    Serial.println("ok.");

  else
    Serial.println("failed.");
}

```

```

}

float leer_ph()
{
  float ph=(-0.029710144)*(analogRead(A0))+(21.63492754);
  Serial.print(" ph=");
  Serial.print(ph);
  return(ph);
}

float leer_temperatura()
{
  sensorDS18B20.requestTemperatures();
  float temperatura=sensorDS18B20.getTempCByIndex(0);
  Serial.print(" Temperatura=");
  Serial.print(temperatura);
  return(temperatura);
}

float leer_turbidez()
{
  float turbidez;
  float adc=analogRead(A1);
  ntu = 0;
  int muestras=10;
  for(int i=0;i<muestras;i++)
  {
    ntu =ntu + (-0.2121*analogRead(A1)+12 o 6.7666);
  }
  //float voltaje = 0.00488*analogRead(A1);
  ntu=ntu/muestras;
  //Serial.print("ADC= ");// Print the result in the serial monitor.
  //Serial.print(adc);
  Serial.print(" NTU=");
  Serial.println(ntu);
  delay(2000);// Take 1 reading per second.
  return(ntu);
}

```

Anexo K

En el Anexo K se detalla el algoritmo para el prototipo del nodo 2

```
#include <RF24Network.h>
#include <RF24.h>
#include <SPI.h>
#include <NewPing.h>

/*Aqui se configuran los pines donde debemos conectar el sensor*/
#define TRIGGER_PIN 7
#define ECHO_PIN 6
#define MAX_DISTANCE 200

/*Crear el objeto de la clase NewPing*/
NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);
RF24 radio(9,10);          // nRF24L01(+) radio attached using Getting Started
board

RF24Network network(radio);    // Network uses that radio

int id=2;
const uint16_t this_node = 02;    // Address of our node in Octal format
const uint16_t other_node = 00;    // Address of the other node in Octal format

const unsigned long interval = 2000; //ms // How often to send 'hello world to the other
unit

unsigned long last_sent;          // When did we last send?
unsigned long packets_sent;      // How many have we sent already

struct payload_t {                // Structure of our payload
    unsigned long caudal;
    unsigned long id;
};
void setup(void)
{

    Serial.begin(57600);
```

```

Serial.println("RF24Network/examples/helloworld_tx/");
SPI.begin();
radio.begin();
network.begin(/*channel*/ 90, /*node address*/ this_node);
radio.setDataRate(RF24_250KBPS);

}

void loop() {

network.update();

float caudal=leer_caudal();

    delay(10);
    enviar(caudal, id);
    delay(5000);

}

void enviar(float caudal, int id)
{

Serial.print("Sending... ");
delay(10);
payload_t payload = {caudal, id};
RF24NetworkHeader header(/*to node*/ other_node);
bool ok = network.write(header,&payload,sizeof(payload));
if (ok)
    Serial.println("ok.");

else
    Serial.println("failed.");

}

float leer_caudal()
{
    delay(1000);
    // Obtener medicion de tiempo de viaje del sonido y guardar en variable uS

```

```
float uS = sonar.ping_median();  
float distancia = (uS / US_ROUNDTRIP_CM);  
float caudal = 0.2125*42*(60-distancia);  
return(caudal);  
}
```

Anexo L

Algoritmo programado para el monitoreo en tiempo de la calidad del agua

```
<script type="text/javascript"  
src="http://ajax.googleapis.com/ajax/libs/jquery/1.8.2/jquery.min.js"></script>  
<script type="text/javascript">
```

```
function obtener_umbral_Omax()  
{  
    var Omax = document.getElementById("Omax").value;  
    return Omax;  
}
```

```
function obtener_umbral_Omin()  
{  
    var Omin = document.getElementById("Omin").value;  
    return Omin;  
}
```

```
function obtener_umbral_Ommax()  
{  
    var Ommax = document.getElementById("Ommax").value;  
    return Ommax;  
}
```

```
function obtener_umbral_Ommin()  
{  
    var Ommin = document.getElementById("Ommin").value;  
    return Ommin;  
}
```

```
function obtener_umbral_T_max()  
{  
    var tempmax = document.getElementById("tempmax").value;  
    return tempmax;  
}
```

```
function obtener_umbral_T_min()  
{  
    var tempmin = document.getElementById("tempmin").value;
```

```

        return tempmin;
    }

function obtener_umbral_T_mmax()
{
    var tempmmax = document.getElementById("tempmmax").value;
    return tempmmax;
}

function obtener_umbral_T_mmin()
{
    var tempmmin = document.getElementById("tempmmin").value;
    return tempmmin;
}

function obtener_umbral_C_max()
{
    var caudalmax = document.getElementById("caudalmax").value;
    return caudalmax;
}

function obtener_umbral_C_min()
{
    var caudalmin = document.getElementById("caudalmin").value;
    return caudalmin;
}

function obtener_umbral_C_mmax()
{
    var caudalmmax = document.getElementById("caudalmmax").value;
    return caudalmmax;
}

function obtener_umbral_C_mmin()
{
    var caudalmmin = document.getElementById("caudalmmin").value;
    return caudalmmin;
}

function obtener_umbral_P_max()

```

```

{
    var PHmax = document.getElementById("PHmax").value;
    return PHmax;
}

function obtener_umbral_P_min()
{
    var PHmin = document.getElementById("PHmin").value;
    return PHmin;
}

function obtener_umbral_P_mmax()
{
    var PHmmax = document.getElementById("PHmmax").value;
    return PHmmax;
}

function obtener_umbral_P_mmin()
{
    var PHmmin = document.getElementById("PHmmin").value;
    return PHmmin;
}

function obtener_parametros()
{
    var oxigenoP2 = $.ajax({
        url:
'http://192.168.0.3/monitoreo_peces/archivos/consultasql.php?a=0&q=12',
        dataType: 'text',
        async: false
    }).responseText;
    document.getElementById("O2").innerHTML=oxigenoP2+" NTU";
    oxigenoP2 = parseFloat(oxigenoP2);
    if(oxigenoP2>obtener_umbral_Omax())
        document.getElementById("O2").className = "MINIMO";
    else
if(oxigenoP2<=obtener_umbral_Omax()&&oxigenoP2>obtener_umbral_Ommax())
        document.getElementById("O2").className = "PASADO";
}

```

```

else
if(oxigenoP2<=obtener_umbral_Omax())&&oxigenoP2>obtener_umbral_Omin())
    document.getElementById("O2").className = "NORMAL";
else
if(oxigenoP2<=obtener_umbral_Omin())&&oxigenoP2>obtener_umbral_Omin())
    document.getElementById("O2").className = "PASADO";

else
    document.getElementById("O2").className = "MINIMO";

var oxigenoP3 = $.ajax({
    url: 'http://192.168.0.3/monitoreo_peces/archivos/consultasql.php?a=0&q=22',
    dataType: 'text',
    async: false
}).responseText;
document.getElementById("O3").innerHTML=oxigenoP3+" NTU";
oxigenoP3 = parseFloat(oxigenoP3);
if(oxigenoP3>=obtener_umbral_Omax())
    document.getElementById("O3").className = "MINIMO";
else
if(oxigenoP3<obtener_umbral_Omax())&&oxigenoP3>obtener_umbral_Omax())
    document.getElementById("O3").className = "PASADO";
else
if(oxigenoP3<obtener_umbral_Omax())&&oxigenoP3>obtener_umbral_Omin())
    document.getElementById("O3").className = "NORMAL";
else
if(oxigenoP3<obtener_umbral_Omin())&&oxigenoP3>obtener_umbral_Omin())
    document.getElementById("O3").className = "PASADO";
else
    document.getElementById("O3").className = "MINIMO";

var temperaturaP1 = $.ajax({
    url: 'http://192.168.0.3/monitoreo_peces/archivos/consultasql.php?a=0&q=2',
    dataType: 'text',
    async: false
}).responseText;
document.getElementById("T1").innerHTML=temperaturaP1+" .C";
temperaturaP1 = parseFloat(temperaturaP1);

```

```

    if(temperaturaP1>=obtener_umbral_T_max())
        document.getElementById("T1").className = "MINIMO";
    else
if(temperaturaP1<obtener_umbral_T_max()&&temperaturaP1>obtener_umbral_T_mma
x())
        document.getElementById("T1").className = "PASADO";
    else
if(temperaturaP1<obtener_umbral_T_mmax()&&temperaturaP1>obtener_umbral_T_m
min())
        document.getElementById("T1").className = "NORMAL";
    else
if(temperaturaP1<obtener_umbral_T_mmin()&&temperaturaP1>obtener_umbral_T_mi
n())
        document.getElementById("T1").className = "PASADO";
    else
        document.getElementById("T1").className = "MINIMO";

var temperaturaP2 = $.ajax({
    url: 'http://192.168.0.3/monitoreo_peces/archivos/consultasql.php?a=0&q=11',
    dataType: 'text',
    async: false
}).responseText;
document.getElementById("T2").innerHTML=temperaturaP2+" .C";
temperaturaP2 = parseFloat(temperaturaP2);
if(temperaturaP2>=obtener_umbral_T_max())
    document.getElementById("T2").className = "MINIMO";
else
if(temperaturaP2<obtener_umbral_T_max()&&temperaturaP2>obtener_umbral_T_mma
x())
        document.getElementById("T2").className = "PASADO";
    else
if(temperaturaP2<obtener_umbral_T_mmax()&&temperaturaP2>obtener_umbral_T_m
min())
        document.getElementById("T2").className = "NORMAL";
    else
if(temperaturaP2<obtener_umbral_T_mmin()&&temperaturaP2>obtener_umbral_T_mi
n())
        document.getElementById("T2").className = "PASADO";
    else
        document.getElementById("T2").className = "MINIMO";

```

```

var temperaturaP3 = $.ajax({
    url: 'http://192.168.0.3/monitoreo_peces/archivos/consultasql.php?a=0&q=21',
    dataType: 'text',
    async: false
}).responseText;
document.getElementById("T3").innerHTML=temperaturaP3+" .C";
temperaturaP3 = parseFloat(temperaturaP3);
if(temperaturaP3>=obtener_umbral_T_max())
    document.getElementById("T3").className = "MINIMO";
else
if(temperaturaP3<obtener_umbral_T_max()&&temperaturaP3>obtener_umbral_T_mma
x())
    document.getElementById("T3").className = "PASADO";
else
if(temperaturaP3<obtener_umbral_T_mmax()&&temperaturaP3>obtener_umbral_T_m
min())
    document.getElementById("T3").className = "NORMAL";
else
if(temperaturaP3<obtener_umbral_T_mmin()&&temperaturaP3>obtener_umbral_T_mi
n())
    document.getElementById("T3").className = "PASADO";
else
    document.getElementById("T3").className = "MINIMO";

var caudalP1 = $.ajax({
    url: 'http://192.168.0.3/monitoreo_peces/archivos/consultasql.php?a=0&q=1',
    dataType: 'text',
    async: false
}).responseText;
document.getElementById("C1").innerHTML=caudalP1+" LPS";
caudalP1 = parseFloat(caudalP1);
if(caudalP1>=obtener_umbral_C_max())
    document.getElementById("C1").className = "MINIMO";
else
if(caudalP1<obtener_umbral_C_max()&&caudalP1>obtener_umbral_C_mmax())
    document.getElementById("C1").className = "PASADO";
else
if(caudalP1<obtener_umbral_C_mmax()&&caudalP1>obtener_umbral_C_mmin())
    document.getElementById("C1").className = "NORMAL";

```

```

else
if(caudalP1<obtener_umbral_C_mmin())&&caudalP1>obtener_umbral_C_min())
    document.getElementById("C1").className = "PASADO";
else
    document.getElementById("C1").className = "MINIMO";

var PHP2 = $.ajax({
    url: 'http://192.168.0.3/monitoreo_peces/archivos/consultasql.php?a=0&q=10',
    dataType: 'text',
    async: false
}).responseText;
document.getElementById("P2").innerHTML=PHP2+" ";
PHP2 = parseFloat(PHP2);
if(caudalP1>=obtener_umbral_P_max())
    document.getElementById("P2").className = "MINIMO";
else if(PHP2<obtener_umbral_P_max())&&PHP2>obtener_umbral_P_mmax())
    document.getElementById("P2").className = "PASADO";
else if(PHP2<obtener_umbral_P_mmax())&&PHP2>obtener_umbral_P_mmin())
    document.getElementById("P2").className = "NORMAL";
else if(PHP2<obtener_umbral_P_mmin())&&PHP2>obtener_umbral_P_min())
    document.getElementById("P2").className = "PASADO";
else
    document.getElementById("P2").className = "MINIMO";

var PHP3 = $.ajax({
    url: 'http://192.168.0.3/monitoreo_peces/archivos/consultasql.php?a=0&q=20',
    dataType: 'text',
    async: false
}).responseText;
document.getElementById("P3").innerHTML=PHP3+" ";
PHP3 = parseFloat(PHP3);
if(PHP3>=obtener_umbral_P_max())
    document.getElementById("P3").className = "MINIMO";
else if(PHP3<obtener_umbral_P_max())&&PHP3>obtener_umbral_P_mmax())
    document.getElementById("P3").className = "PASADO";
else if(PHP3<obtener_umbral_P_mmax())&&PHP3>obtener_umbral_P_mmin())
    document.getElementById("P3").className = "NORMAL";    else
if(PHP3<obtener_umbral_P_mmin())&&PHP3>obtener_umbral_P_min())
    document.getElementById("P3").className = "PASADO";
else

```

```

        document.getElementById("P3").className = "MINIMO";

    }

setInterval(obtener_parametros,2000);

</script>
<html>
<head>
<title>AJAX</title>
<style "type=text/css">
    body {color: black;background-color: white;}
    div {background-color: black; color: white; border: 2px solid black; padding:
15px; margin: 15px; height: auto;
    width: 100px; border-radius: 30px; text-align: center; float: left; font-
weight:bold;}
    .cabecera {background-color: #BDBDBD; clear: left;}
    .formulario {background-color: #2E9AFE; width: auto;}
    .oculto {background-color: transparent; border: none;}
    .NORMAL {background-color: green;}
    .PASADO {background-color: orange;}
    .MINIMO {background-color: red;}
    .ETIQUETAS {background-color: #A4A4A4; border:none;}
    input, #boton {width: 80px; border-radius: 5px; font-weight:bold;text-align:
center;}
    #boton:hover{border: 3px solid black;}
    #principal {background-color: #CED8F6;}
</style>

</head>
<body>
<div id = "principal" class="formulario">
<h2>MONITOREO DE PARAMETROS DE CALIDAD DE AGUA</h2>

<!FORMULARIO ENTRADA DE AGUA>
<div id="Fagua" class="formulario">
<p>PARAMETROS CANAL DE DERIVACION </p>

```

```
<div class="ETIQUETAS">
PARAMETRO
</div>
<div class="ETIQUETAS">
MAGNITUD
</div>
<div class="cabecera">
Temperatura
</div>
<div id="T1" class="NORMAL">
N/A
</div>
<div class="cabecera">
Caudal
</div>
<div id="C1" class="NORMAL">
N/A
</div>
</div>
```

```
<!FORMULARIO PISCINA 1>
<div id="Ftemperatura" class="formulario">
<p>PARAMETROS PISCINA 1</p>
<div class="ETIQUETAS">
PARAMETRO
</div>
<div class="ETIQUETAS">
MAGNITUD
</div>
<div class="cabecera">
Temperatura
</div>
<div id="T2" class="NORMAL">
N/A
</div>
<div class="cabecera">
Turbidez
</div>
<div id="O2" class="NORMAL">
```

N/A
</div>
<div class="cabecera">
PH
</div>
<div id="P2" class="NORMAL">
N/A
</div>
</div>

<!FORMULARIO PISCINA 2>
<div id="Fph" class="formulario">
<p>PARAMETROS PISCINA 2</p>
<div class="ETIQUETAS">
PARAMETRO
</div>
<div class="ETIQUETAS">
MAGNITUD
</div>
<div class="cabecera">
Temperatura
</div>
<div id="T3" class="NORMAL">
N/A
</div>
<div class="cabecera">
Turbidez
</div>
<div id="O3" class="NORMAL">
N/A
</div>
<div class="cabecera">
PH
</div>
<div id="P3" class="NORMAL">
N/A
</div>
</div>

```

<div class="formulario">
<p>CONFIGURACION Y PRUEBA</p>
<div class="ETIQUETAS">
PARAMETRO
</div>
<div class="ETIQUETAS">
MAXIMO
</div>
<div class="ETIQUETAS">
OPTIMO +
</div>
<div class="ETIQUETAS">
OPTIMO -
</div>
<div class="ETIQUETAS">
MINIMO
</div>

<div class="cabecera">
POTENCIAL HIDROGENO (PH)
</div>
<div id="umbralmax" class="oculto">
<input id="PHmax" type="number" onChange="cambia_umbral()" value="10"
onMouseOver="cambiaIntervalo(0)" onMouseOut="cambiaIntervalo(5000)">
</div>
<div id="umbralmed1" class="oculto">
<input id="PHmmax" type="number" onChange="cambia_umbral()" value="7.9"
onMouseOv$>
</div>
<div id="umbralmed2" class="oculto">
<input id="PHmmin" type="number" onChange="cambia_umbral()" value="6.6"
onMouseOv$>
</div>
<div id="umbralmin" class="oculto">
<input id="PHmin" type="number" onChange="cambia_umbral()" value="5.1"
onMouseOv$
</div>
</div>

```

```

<div class="cabecera">
TEMPERAT (0C)
</div>
<div id="umbramax2" class="oculto">
<input id="tempmax" type="number" onChange="cambia_umbral()" value="18"
onMouseOver="cambiaIntervalo(0)" onMouseOut="cambiaIntervalo(5000)">
</div>
<div id="umbramed1" class="oculto">
<input id="tempmmax" type="number" onChange="cambia_umbral()" value="16"
onMouseOv$>
</div>
<div id="umbramed2" class="oculto">
<input id="tempmmin" type="number" onChange="cambia_umbral()" value="14"
onMouseOv$>
</div>
<div id="umbramin2" class="oculto">
<input id="tempmin" type="number" onChange="cambia_umbral()" value="11"
onMouseOv$
</div>
</div>

<div class="cabecera">
CAUDAL (LPS)
</div>
<div id="umbralmax3" class="oculto">
<input id="caudalmax" type="number" onChange="cambia_umbral()" value="20"
onMouseOver="cambiaIntervalo(0)" onMouseOut="cambiaIntervalo(5000)">
</div>
<div id="umbralmmax3" class="oculto">
<input id="caudalmmax" type="number" onChange="cambia_umbral()" value="13"
onMouseOv$>
</div>
<div id="umbralmin3" class="oculto">
<input id="caudalmmin" type="number" onChange="cambia_umbral()" value="10"
onMouseOv$>
</div>
<div id="umbralmin3" class="oculto">
<input id="caudalmin" type="number" onChange="cambia_umbral()" value="6"
onMouseOv$
</div>

```

```
</div>
```

```
<div class="cabecera">
```

```
TURBIDEZ (NTU)
```

```
</div>
```

```
<div id="umbralmax4" class="oculto">
```

```
<input id="Omax" type="number" onChange="cambia_umbral()" value="12"  
onMouseOver="cambiaIntervalo(0)" onMouseOut="cambiaIntervalo(5000)">
```

```
</div>
```

```
<div id="umbralmin4" class="oculto">
```

```
<input id="Ommax" type="number" onChange="cambia_umbral()" value="9"  
onMouseOv$>
```

```
</div>
```

```
<div id="umbralmin4" class="oculto">
```

```
<input id="Ommin" type="number" onChange="cambia_umbral()" value="6"  
onMouseOv$>
```

```
</div>
```

```
<div id="umbralmin4" class="oculto">
```

```
<input id="Omin" type="number" onChange="cambia_umbral()" value="4.6"  
onMouseOv$>
```

```
</div>
```

```
</div>
```

```
<div class="cabecera">
```

```
PROBAR SISTEMA
```

```
</div>
```

```
<div id="boton" onClick="boton_manual()" onMouseOver="cambiaIntervalo(0)"  
onMouseOut="cambiaIntervalo(5000)" >OFF</div>
```

```
</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

Anexo M

Algoritmo programado para el ingreso de datos en la base MySQL

```
<?php
$conexion = mysql_connect('localhost', 'root', 'Ytjda3toGe') or die ("No se pudo
conectar a la base de datos");
mysql_select_db('nodos', $conexion) or die ("No se encontro la base de datos");

$vo = $_GET['vo'];
$v1 = $_GET['v1'];
$v2 = $_GET['v2'];
$v3 = $_GET['v3'];
$v4 = $_GET['v4'];

switch($vo)
{
case 1:
$conconsultar=mysql_query("SELECT * FROM LSP1 ORDER BY id DESC LIMIT 0,1",
$conexion);
$valor=mysql_result($conconsultar,0,"vida_util");
mysql_free_result($conconsultar);
$v1 = $v1 + $valor;
$sql = "INSERT INTO LSP1(vida_util, I_promedio, Alarmas) VALUES ('$v1','$v2',
'$v3')";
break;

case 2:
$conconsultar=mysql_query("SELECT * FROM LSP2 ORDER BY id DESC LIMIT 0,1",
$conexion);
$valor=mysql_result($conconsultar,0,"vida_util");
mysql_free_result($conconsultar);
$v1 = $v1 + $valor;
$sql = "INSERT INTO LSP2(vida_util, I_promedio, Alarmas) VALUES ('$v1','$v2',
'$v3')";
break;

case 3:
$conconsultar=mysql_query("SELECT * FROM LSP3 ORDER BY id DESC LIMIT 0,1",
$conexion);
$valor=mysql_result($conconsultar,0,"vida_util");
```

```

mysql_free_result($consultar);
$v1 = $v1 + $valor;
$sql = "INSERT INTO LSP3(vida_util, I_promedio, Alarmas) VALUES ('$v1','$v2',
'$v3')";
break;

case 100:
$consultar=mysql_query("SELECT * FROM LLD1 ORDER BY id DESC LIMIT 0,1",
$conexion);
$valor=mysql_result($consultar,0,"vida_util");
mysql_free_result($consultar);
$v1 = $v1 + $valor;
$sql = "INSERT INTO LLD1(vida_util, I_promedio, Alarmas, atnum) VALUES
('$v1','$v2', '$v3','$v4')";
break;

case 101:
$consultar=mysql_query("SELECT * FROM LLD2 ORDER BY id DESC LIMIT 0,1",
$conexion);
$valor=mysql_result($consultar,0,"vida_util");
mysql_free_result($consultar);
$v1 = $v1 + $valor;
$sql = "INSERT INTO LLD2(vida_util, I_promedio, Alarmas, atnum) VALUES
('$v1','$v2', '$v3','$v4')";
break;
}

mysql_query($sql, $conexion);
mysql_close($conexion);
?>

```

Anexo N

Algoritmo programado para generación de reporte en PDF

php

```
include 'plantilla.php';
require 'conexion.php';

$query = "SELECT id, caudal, temperatura, fecha FROM entrada ";
$resultado = $mysqli->query($query);

$pdf = new PDF();
$pdf->AliasNbPages();
$pdf->AddPage();

$espacio="          ";
$pdf->MultiCell(180,8,$espacio);
$id_piscina="          REPORTE DE PARAMETROS DEL CANAL DE
ENTRADA";
$pdf->SetFont('Times','B',16);
$pdf->MultiCell(180,8,$id_piscina);
$pdf->MultiCell(180,8,$espacio);

$pdf->SetFillColor(232,232,232);
$pdf->SetFont('Arial','B',12);
$pdf->Cell(50,6,'MUESTRA N.',1,0,'C',1);
$pdf->Cell(30,6,'CAUDAL',1,0,'C',1);
$pdf->Cell(40,6,'TEMPERATURA',1,0,'C',1);
$pdf->Cell(70,6,'FECHA/HORA',1,1,'C',1);
$pdf->SetFont('Arial','',10);

while($row = $resultado->fetch_assoc())
{
    $pdf->Cell(50,6,utf8_decode($row['id']),1,0,'C');
    $pdf->Cell(30,6,$row['caudal'],1,0,'C');
    $pdf->Cell(40,6,utf8_decode($row['temperatura']),1,0,'C');
    $pdf->Cell(70,6,utf8_decode($row['fecha']),1,1,'C');
}

$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,"Firma del Responsable");
$pdf->MultiCell(180,8,$espacio);
```

```

$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,"_____");

$query2 = "SELECT id, ph, temperatura, turbidez, fecha FROM piscina1 ";
$resultado2 = $mysqli->query($query2);

$pdf->AddPage();
$pdf->MultiCell(180,8,$espacio);
$id_piscina2="          REPORTE DE PARAMETROS DE PISCINA 1";
$pdf->SetFont('Times','B',16);
$pdf->MultiCell(180,8,$id_piscina2);
$pdf->MultiCell(180,8,$espacio);

$pdf->SetFillColor(232,232,232);
$pdf->SetFont('Arial','B',12);
$pdf->Cell(40,6,'MUESTRA N.',1,0,'C',1);
$pdf->Cell(20,6,'PH',1,0,'C',1);
$pdf->Cell(40,6,'TEMPERATURA',1,0,'C',1);
$pdf->Cell(30,6,'TURBIDEZ',1,0,'C',1);
$pdf->Cell(50,6,'FECHA/HORA',1,1,'C',1);
$pdf->SetFont('Arial','',10);

while($row = $resultado2->fetch_assoc())
{
    $pdf->Cell(40,6,utf8_decode($row['id']),1,0,'C');
    $pdf->Cell(20,6,$row['ph'],1,0,'C');
    $pdf->Cell(40,6,utf8_decode($row['temperatura']),1,0,'C');
    $pdf->Cell(30,6,utf8_decode($row['turbidez']),1,0,'C');
    $pdf->Cell(50,6,utf8_decode($row['fecha']),1,1,'C');
}

$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,"Firma del Responsable");
$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,"_____");

$query3 = "SELECT id, ph, temperatura, turbidez, fecha FROM piscina2 ";
$resultado3 = $mysqli->query($query3);

```

```

$pdf->AddPage();
$pdf->MultiCell(180,8,$espacio);
$id_piscina3="                REPORTE DE PARAMETROS DE PISCINA 2";
$pdf->SetFont('Times','B',16);
$pdf->MultiCell(180,8,$id_piscina3);
$pdf->MultiCell(180,8,$espacio);

$pdf->SetFillColor(232,232,232);
$pdf->SetFont('Arial','B',12);
$pdf->Cell(40,6,'MUESTRA N.',1,0,'C',1);
$pdf->Cell(20,6,'PH',1,0,'C',1);
$pdf->Cell(40,6,'TEMPERATURA',1,0,'C',1);
$pdf->Cell(30,6,'TURBIDEZ',1,0,'C',1);
$pdf->Cell(50,6,'FECHA/HORA',1,1,'C',1);
$pdf->SetFont('Arial','',10);

while($row = $resultado3->fetch_assoc())
{
    $pdf->Cell(40,6,utf8_decode($row['id']),1,0,'C');
    $pdf->Cell(20,6,$row['ph'],1,0,'C');
    $pdf->Cell(40,6,utf8_decode($row['temperatura']),1,0,'C');
    $pdf->Cell(30,6,utf8_decode($row['turbidez']),1,0,'C');
    $pdf->Cell(50,6,utf8_decode($row['fecha']),1,1,'C');
}
$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,"Firma del Responsable");
$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,$espacio);
$pdf->MultiCell(180,8,"_____");

    $mysqli = new mysqli("localhost","root","Ytjda3toGe","monitoreo");
    mysqli_query($mysqli, "TRUNCATE TABLE entrada");
mysqli_query($mysqli, "TRUNCATE TABLE piscina1");
mysqli_query($mysqli, "TRUNCATE TABLE piscina2");
    $pdf->Output();
?>

```

Anexo O

```
<?php
```

```
$tmax1=16;
```

```
$tmax2=18;
```

```
$tmin1=14;
```

```
$tmin2=11;
```

```
$cmax1=13;
```

```
$cmax2=20;
```

```
$cmin1=10;
```

```
$cmin2=6;
```

```
$pmax1=7.9;
```

```
$pmax2=10;
```

```
$pmin1=6.6;
```

```
$pmin2=5.1;
```

```
$zmax1=9;
```

```
$zmax2=12;
```

```
$zmin1=6;
```

```
$zmin2=4.6;
```

```
$caudal = $_GET['c'];
```

```
$temperatura = $_GET['t'];
```

```
$ph = $_GET['p'];
```

```
$turbidez = $_GET['z'];
```

```
$nodo = $_GET['i'];
```

```
$estado="";
```

```
$variable1="";
```

```
$variable2="";
```

```
$variable3="";
```

```
$variable4="";
```

```
$cuerpo="";
```

```
if ($caudal > $cmax2 || $caudal < $cmin2 || $temperatura>$tmax2 ||
```

```
$temperatura<$tmin2 || $ph>$pmax2 || $ph<$pmin2 || $turbidez>$zmax2 ||
```

```
$turbidez<$zmin2)
```

```
$estado=' ALARMA !!! VALOR FUERA DE PARAMETROS PERMITIDOS, PECES  
EN PELIGRO ';
```

```
else $estado= ' ALERTA !!! VALORES FUERA DE PARAMETROS OPTIMOS,  
REVISAR PISCINAS ';
```

```

if ($caudal > $cmax1 && $caudal <= $cmax2)
{
$variable1='La Variable Caudal se encuentra FUERA DE RANGO NORMAL en la
PISCINA '$nodo.'.
  VALOR CAUDAL = '$caudal.', Valor Moderadamente Alto. Revisar !!!';
}elseif ($caudal > $cmin2 && $caudal <= $cmin1)
{
$variable1='La Variable Caudal se encuentra FUERA DE RANGO NORMAL en la
PISCINA '$nodo.'.
  VALOR CAUDAL = '$caudal.', Valor Moderadamente Bajo. Revisar !!!';
}elseif ($caudal > $cmax2)
{
$variable1='La Variable Caudal se encuentra FUERA DE RANGO SEGURO en la
PISCINA '$nodo.'.
  VALOR CAUDAL = '$caudal.', VALOR EXTREMADAMENTE ALTO. REVISAR
URGENTE !!!';
}elseif ($caudal <= $cmin2)
{
$variable1='La Variable Caudal se encuentra FUERA DE RANGO SEGURO en la
PISCINA '$nodo.'.
  VALOR CAUDAL = '$caudal.', VALOR EXTREMADAMENTE BAJO. REVISAR
URGENTE !!!';
}
}

```

```

if ($temperatura > $tmax1 && $temperatura <= $tmax2)
{
$variable2='La Variable Temperatura se encuentra FUERA DE RANGO NORMAL en
la PISCINA '$nodo.'.
  VALOR TEMPERATURA = '$temperatura.', Valor Moderadamente Alto. Revisar !!!';
}elseif ($temperatura > $tmin2 && $temperatura <= $tmin1)
{
$variable2='La Variable Temperatura se encuentra FUERA DE RANGO NORMAL en
la PISCINA '$nodo.'.
  VALOR TEMPERATURA = '$temperatura.', Valor Moderadamente Bajo. Revisar !!!';
}elseif ($temperatura > $tmax2)
{

```

```
$variable2='La Variable Temperatura se encuentra FUERA DE RANGO SEGURO en la PISCINA '.$nodo.'
```

```
  VALOR TEMPERATURA = '.$temperatura.', VALOR EXTREMADAMENTE ALTO. REVISAR URGENTE !!!';
```

```
}elseif ($temperatura <= $tmin2)
```

```
{
```

```
$variable1='La Variable Caudal se encuentra FUERA DE RANGO SEGURO en la PISCINA '.$nodo.'
```

```
  VALOR CAUDAL = '.$caudal.', VALOR EXTREMADAMENTE BAJO. REVISAR URGENTE !!!';
```

```
}
```

```
if ($ph > $pmax1 && $ph <= $pmax2)
```

```
{
```

```
$variable3='La Variable PH se encuentra FUERA DE RANGO NORMAL en la PISCINA '.$nodo.'
```

```
  VALOR PH = '.$ph.', Valor Moderadamente Alto. Revisar !!!';
```

```
}elseif ($ph > $pmin2 && $ph <= $pmin1)
```

```
{
```

```
$variable3='La Variable PH se encuentra FUERA DE RANGO NORMAL en la PISCINA '.$nodo.'
```

```
  VALOR PH = '.$ph.', Valor Moderadamente Bajo. Revisar !!!';
```

```
}elseif ($ph > $pmax2)
```

```
{
```

```
$variable3='La Variable PH se encuentra FUERA DE RANGO SEGURO en la PISCINA '.$nodo.'
```

```
  VALOR PH = '.$ph.', VALOR EXTREMADAMENTE ALTO. REVISAR URGENTE !!!';
```

```
}elseif ($ph <= $pmin2)
```

```
{
```

```
$variable3='La Variable PH se encuentra FUERA DE RANGO SEGURO en la PISCINA '.$nodo.'
```

```
  VALOR PH = '.$ph.', VALOR EXTREMADAMENTE BAJO. REVISAR URGENTE !!!';
```

```
}
```

```
if ($sturbidez > $zmax1 && $sturbidez <= $zmax2)
```

```
{
```

```

$variable4='La Variable Turbidez se encuentra FUERA DE RANGO NORMAL en la
PISCINA '.$nodo.'.
  VALOR TURBIDEZ = '.$turbidez.', Valor Moderadamente Alto. Revisar !!!';
}elseif ($turbidez > $zmin2 && $turbidez <= $zmin1)
{
$variable4='La Variable Turbidez se encuentra FUERA DE RANGO NORMAL en la
PISCINA '.$nodo.'.
  VALOR TURBIDEZ = '.$turbidez.', Valor Moderadamente Bajo. Revisar !!!';
}elseif ($turbidez > $zmax2)
{
$variable4='La Variable TURBIDEZ se encuentra FUERA DE RANGO SEGURO en la
PISCINA '.$nodo.'.
  VALOR TURBIDEZ = '.$turbidez.', VALOR EXTREMADAMENTE ALTO.
REVISAR URGENTE !!!';
}elseif ($turbidez <= $zmin2)
{
$variable4='La Variable TURBIDEZ se encuentra FUERA DE RANGO SEGURO en la
PISCINA '.$nodo.'.
  VALOR TURBIDEZ = '.$turbidez.', VALOR EXTREMADAMENTE BAJO.
REVISAR URGENTE !!!';
}

if($variable1=="")
$cuero=".$variable2.'

'.$variable3.'

'.$variable4;
elseif($variable2=="")
$cuero=".$variable1.'

'.$variable3.'

'.$variable4;
if($variable3=="")
$cuero=".$variable1.'

'.$variable2.'

'.$variable4;

```

```
if($variable4=="")
$cuero=".$variable1.'
```

```
.$variable2.'
```

```
.$variable3;
```

```
require '/usr/share/php/libphp-phpmailer/class.phpmailer.php';
require '/usr/share/php/libphp-phpmailer/class.smtp.php';
$mail = new PHPMailer;
$mail->setFrom('monitoreo.piscinas.peces@gmail.com');
$mail->addAddress('el.chaparral.salcedo@gmail.com');
$mail->Subject = $estado;
$mail->Body = $cuero;
$mail->IsSMTP();
$mail->SMTPSecure = 'tls';
$mail->Host = 'tls://smtp.gmail.com';
$mail->SMTPAuth = true;
$mail->Port = 587;
```

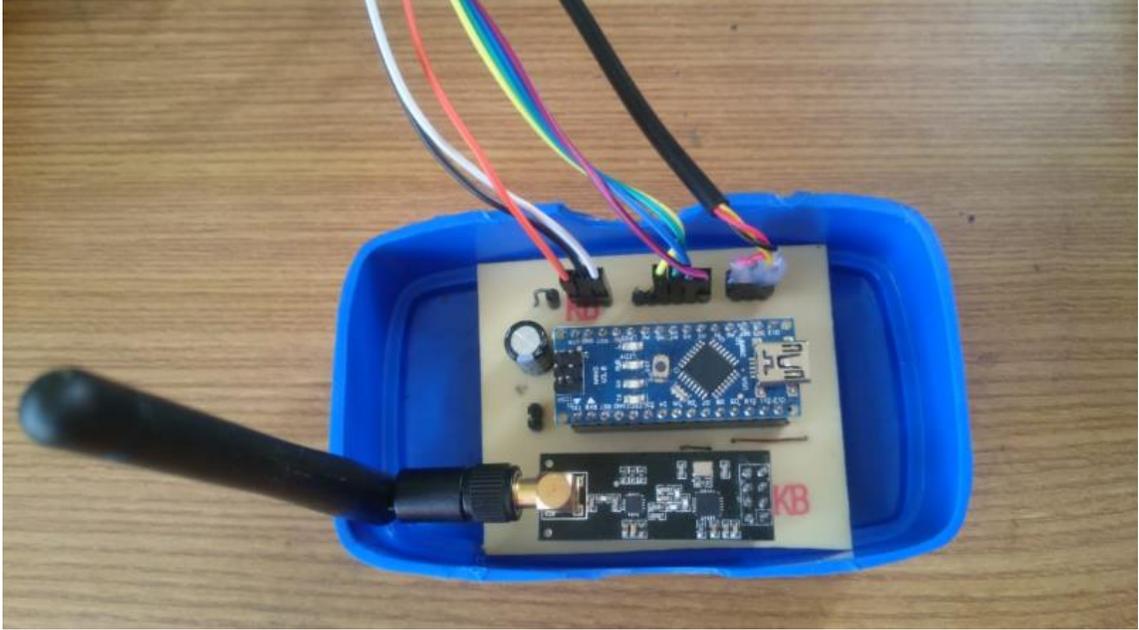
```
//Set your existing gmail address as user name
$mail->Username = 'el.chaparral.salcedo@gmail.com';
```

```
//Set the password of your gmail address here
```

```
$mail->Password = 'Ytjda3toGe';
if(!$mail->send()) {
    echo 'Email is not sent.';
    echo 'Email error: '. $mail->ErrorInfo;
} else {
    echo 'Email has been sent.';
}
?>
```

Anexo P

Construcción del nodo sensor 1 y 2



Estructura del nodo sensor 1 y 2



Pruebas del nodo sensor 1



Adquisición de valores de temperatura y pH en la piscina de truchas



Pruebas de los nodos sensores en las piscinas de truchas



Pruebas de los nodos sensores en las piscinas de truchas



Ubicación del nodo sensor en la piscina de alevines

