



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES INDUSTRIALES**

**“CONSTRUCCIÓN DE UN VEHÍCULO ACUÁTICO DE  
SUPERFICIE PARA SUPERVISIÓN EN PISCINAS  
CAMARONERAS EMPLEANDO UN PLANIFICADOR DE  
TRAYECTORIAS POR PUNTOS”**

**TRABAJO DE TITULACIÓN**

TIPO: DISPOSITIVO TECNOLÓGICO

Presentado para optar al grado académico de:

**INGENIERO EN ELECTRÓNICA, CONTROL Y REDES  
INDUSTRIALES**

**AUTORES: ESTÉVEZ MARÍN CARLOS ESTEBAN**

**RODAS MORALES JOSÉ ENRIQUE**

**TUTOR: ING. FAUSTO RAMIRO CABRERA AGUAYO**

Riobamba-Ecuador

2019

**©2019, Carlos Esteban Estévez Marín; José Enrique Rodas Morales**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el derecho de los autores.

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INGENIERÍA ELECTRÓNICA**

**ESCUELA DE INGENIERÍA ELECTRÓNICA, CONTROL Y REDES INDUSTRIALES**

El Tribunal del trabajo de titulación certifica que: El trabajo de investigación: Dispositivo tecnológico “CONSTRUCCIÓN DE UN VEHÍCULO ACUÁTICO DE SUPERFICIE PARA SUPERVISIÓN EN PISCINAS CAMARONERAS EMPLEANDO UN PLANIFICADOR DE TRAYECTORIAS POR PUNTOS”, de responsabilidad de los señores CARLOS ESTEBAN ESTÉVEZ MARÍN Y JOSÉ ENRIQUE RODAS MORALES, ha sido minuciosamente revisado por los Miembros del Tribunal del trabajo de titulación, queda autorizado su presentación.

**FIRMA**

**FECHA**

Dr. Washington Gilberto Luna Encalada

**DECANO DE LA FACULTAD DE  
INFORMÁTICA Y ELECTRÓNICA**

\_\_\_\_\_

\_\_\_\_\_

Ing. Freddy Enrique Chávez Vázquez

**DIRECTOR DE LA ESCUELA DE  
INGENIERÍA ELECTRÓNICA,  
CONTROL Y REDES INDUSTRIALES**

\_\_\_\_\_

\_\_\_\_\_

Ing. Fausto Ramiro Cabrera Aguayo

**DIRECTOR DEL TRABAJO DE  
TITULACIÓN**

\_\_\_\_\_

\_\_\_\_\_

Ing. José Luis Morales Gordon

**MIEMBRO DEL TRIBUNAL**

\_\_\_\_\_

\_\_\_\_\_

Nosotros, Carlos Esteban Estévez Marín Y José Enrique Rodas Morales, somos responsables de las ideas, doctrinas y resultados expuestos en esta tesis y patrimonio intelectual de la Tesis de Grado pertenece a la Escuela Superior Politécnica de Chimborazo.

Carlos Esteban Estévez Marín

José Enrique Rodas Morales

## **DEDICATORIA**

El presente trabajo investigativo lo dedico principalmente a Dios, por darme fuerza para continuar en este proceso de obtención uno de los anhelos más deseados en segundo lugar mis padres, por su amor, trabajo y sacrificio en todos estos años, gracias a ustedes he logrado llegar hasta aquí y convertirme en lo que soy. Ha sido un orgullo y el privilegio de ser su hijo, son los mejores padres en tercer lugar a mis hermanas por estar siempre presentes, acompañándome en las buenas y malas y por el apoyo moral, que me brindaron a lo largo de esta etapa de mi vida y finalmente todas las personas que me han apoyado y han hecho que el trabajo se realice con éxito en especial a aquellos que nos abrieron las puertas y compartieron sus conocimientos.

Carlos Esteban Estévez Marín

Dedico este proyecto a:

Únicamente a mi familia, por su esfuerzo y perseverancia de salir adelante día a día, y son ellos mi única motivación para lograr lo propuesto en mi vida.

José Enrique Rodas Morales

## **AGRADECIMIENTO**

Me van a faltar páginas para agradecer a las personas que se han involucrado en la realización de este trabajo, sin embargo, merecen reconocimiento especial mi Madre y mi Padre que con su esfuerzo y dedicación me ayudaron a culminar mi carrera universitaria y me dieron el apoyo suficiente para no decaer cuando todo parecía complicado e imposible.

A todos mis amigos, José, Pancho, Kevin, Zambo, Daya, Ana, Jhonatan, Marcos, Maggy y Taty que me ayudaron de una manera desinteresada, gracias infinitas por toda su ayuda y buena voluntad sin su apoyo no lo habría logrado.

Carlos Esteban Estévez Marín

Este camino de la formación académica no fue fácil y en primera instancia agradezco a mis maestros que han hecho de mi un profesional con conocimientos sólidos. A mis queridos amigos: Carlos, Pancho, Ana, Kevin, Zambo, Daya, Jhonatan, Marcos, Margarita y Taty, quienes me apoyaron durante toda mi vida universitaria en los buenos y malos momentos.

José Enrique Rodas Morales

## TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	xi
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE ANEXOS .....	xiv
ÍNDICE DE ABREVIATURAS.....	xv
RESUMEN.....	xvi
SUMMARY .....	xvii
INTRODUCCIÓN .....	1

## CAPÍTULO I

<b>1. MARCO TEÓRICO .....</b>	<b>4</b>
<b>1.1. Piscinas Acuícolas .....</b>	<b>4</b>
<b>1.2. Variables de Proceso .....</b>	<b>5</b>
<i>1.2.1. Oxígeno Disuelto .....</i>	<i>6</i>
<i>1.2.2. Temperatura.....</i>	<i>6</i>
<i>1.2.3. pH.....</i>	<i>7</i>
<i>1.2.4. Salinidad .....</i>	<i>7</i>
<b>1.3. Sensores de las variables a supervisar .....</b>	<b>8</b>
<i>1.3.1. Sensor de temperatura.....</i>	<i>8</i>
<i>1.3.2. Sensor de oxígeno .....</i>	<i>9</i>
<i>1.3.3. Sensor de pH.....</i>	<i>9</i>
<b>1.4. Tarjetas programables .....</b>	<b>10</b>
<i>1.4.1. Pixhawk.....</i>	<i>10</i>
<i>1.4.2. Arduino .....</i>	<i>11</i>
<i>1.4.3. Raspberry .....</i>	<i>11</i>
<b>1.5. Transmisión de datos.....</b>	<b>12</b>
<b>1.6. Base de datos .....</b>	<b>14</b>
<i>1.6.1. Plataformas IoT.....</i>	<i>15</i>
<b>1.7. Aplicaciones móviles.....</b>	<b>15</b>
<i>1.7.1. Android.....</i>	<i>16</i>
<i>1.7.1.1. Arquitectura.....</i>	<i>16</i>
<b>1.8. Vehículos subacuáticos no tripulados (ASV) .....</b>	<b>17</b>
<b>1.9. Trayectorias .....</b>	<b>18</b>

<b>1.9.1.</b>	<b>Trayectorias punto a punto .....</b>	<b>18</b>
<b>1.10.</b>	<b>MOTORES.....</b>	<b>19</b>
<b>1.10.1.</b>	<b>Servomotores.....</b>	<b>19</b>
<b>1.10.2.</b>	<b>Brushless.....</b>	<b>19</b>
<b>1.10.3.</b>	<b>Paso a paso.....</b>	<b>20</b>
<b>1.11.</b>	<b>Baterías.....</b>	<b>21</b>
<b>1.12.</b>	<b>GPS .....</b>	<b>21</b>
<b>1.13.</b>	<b>Radio transmisor .....</b>	<b>22</b>
<b>1.13.1.</b>	<b>Telemetría .....</b>	<b>22</b>
<b>1.13.2.</b>	<b>Comparativas emisoras.....</b>	<b>23</b>
<b>1.14.</b>	<b>Estructura.....</b>	<b>23</b>
<b>1.14.1.</b>	<b>Fibra de Vidrio.....</b>	<b>23</b>
<b>1.14.2.</b>	<b>Fibra de Carbono.....</b>	<b>24</b>
<b>1.15.</b>	<b>Proceso actual utilizado para medición de variables del agua en camaroneras ..</b>	<b>24</b>
<b>1.16.</b>	<b>Trabajos semejantes realizados en el Ecuador .....</b>	<b>25</b>

## CAPÍTULO II

<b>2.</b>	<b>MARCO METODOLÓGICO .....</b>	<b>26</b>
<b>2.1.</b>	<b>Requerimientos del sistema .....</b>	<b>26</b>
<b>2.2.</b>	<b>Concepción universal del sistema del prototipo ASV .....</b>	<b>27</b>
<b>2.3.</b>	<b>Diseño de la arquitectura del RASSC .....</b>	<b>28</b>
<b>2.3.1.</b>	<b>Etapa de supervisión.....</b>	<b>28</b>
<b>2.3.2.</b>	<b>Etapa de almacenamiento .....</b>	<b>29</b>
<b>2.3.3.</b>	<b>Etapa de desplazamiento .....</b>	<b>30</b>
<b>2.4.</b>	<b>Elección de dispositivos electrónicos para la construcción del RASSC .....</b>	<b>30</b>
<b>2.4.1.</b>	<b>Tarjetas de desarrollo .....</b>	<b>31</b>
<b>2.4.1.1.</b>	<b>Arduino mega 2560 .....</b>	<b>31</b>
<b>2.4.1.2.</b>	<b>Raspberry pi 3 B.....</b>	<b>31</b>
<b>2.4.1.3.</b>	<b>Pixhawk .....</b>	<b>32</b>
<b>2.4.2.</b>	<b>Sensores de temperatura, oxígeno disuelto y pH.....</b>	<b>33</b>
<b>2.4.2.1.</b>	<b>Temperatura .....</b>	<b>33</b>
<b>2.4.2.2.</b>	<b>Sensores de temperatura .....</b>	<b>34</b>
<b>2.4.2.3.</b>	<b>Oxígeno disuelto (DO).....</b>	<b>34</b>
<b>2.4.2.4.</b>	<b>Sensores de DO .....</b>	<b>35</b>
<b>2.4.2.5.</b>	<b>PH.....</b>	<b>36</b>

2.4.2.6.	<i>Selección del sensor</i> .....	37
2.4.3.	<b><i>Tarjeta de transmisión de datos</i></b> .....	38
2.4.4.	<b><i>Motor Brushless y módulo ESC</i></b> .....	39
2.4.4.1.	<i>Módulo ESC para tarjetas pixhawk</i> .....	40
2.4.5.	<b><i>Módulo GPS</i></b> .....	40
2.4.6.	<b><i>Mavlink de pixhawk</i></b> .....	41
2.4.7.	<b><i>Receptor para joystick</i></b> .....	42
2.4.8.	<b><i>Consumo energético del sistema</i></b> .....	42
2.4.8.1.	<i>Selección de batería</i> .....	43
2.5.	<b><i>Diseño estructura RASSC</i></b> .....	44
2.5.1.	<b><i>Referencia para el diseño del ASV</i></b> .....	44
2.5.2.	<b><i>Casco principal con tapa</i></b> .....	45
2.5.3.	<b><i>Flotadores</i></b> .....	46
2.6.	<b><i>Esquemas de conexiones del RASSC</i></b> .....	46
2.6.1.	<b><i>Diagrama de conexión de la etapa de supervisión</i></b> .....	46
2.6.2.	<b><i>Diagrama de conexión de la etapa de Transmisión</i></b> .....	48
2.6.3.	<b><i>Diagrama de conexión de la etapa de desplazamiento</i></b> .....	49
2.7.	<b><i>Diseño del software</i></b> .....	50
2.7.1.	<b><i>Requerimientos del software</i></b> .....	50
2.7.2.	<b><i>Softwares utilizados para la programación del RASSC</i></b> .....	51
2.7.2.1.	<i>Software para la tarjeta de adquisición de datos Arduino</i> .....	51
2.7.2.2.	<i>Software Python</i> .....	52
2.7.2.3.	<i>Software Android estudio</i> .....	52
2.7.3.	<b><i>Programación realizada para cada etapa del RASSC</i></b> .....	52
2.7.3.1.	<i>Diagrama de flujo del sistema de supervisión</i> .....	53
2.7.3.2.	<i>Bibliotecas etapa de supervisión</i> .....	54
2.7.3.3.	<i>Diagrama de flujo de la etapa de trasmisión</i> .....	55
2.7.3.4.	<i>Diagrama de flujo de la etapa de Desplazamiento</i> .....	57
2.7.4.	<b><i>App móvil para la presentación de información</i></b> .....	59
2.8.	<b><i>Especificaciones y configuraciones del RASSC</i></b> .....	61

### **CAPÍTULO III**

3.	<b><i>ANÁLISIS Y RESULTADOS</i></b> .....	62
3.1.	<b><i>Pruebas de la etapa de supervisión</i></b> .....	62
3.1.1.	<b><i>Caracterización del sensor de temperatura</i></b> .....	63

3.1.2.	<i>Caracterización del sensor de pH.....</i>	65
3.1.3.	<i>Caracterización del sensor de Oxígeno disuelto.....</i>	67
3.1.4.	<i>Prueba de estabilidad de la etapa de supervisión .....</i>	68
3.1.5.	<i>Caracterización de la comunicación entre las etapas de supervisión y transmisión</i>	70
3.2.	<b>Pruebas de la etapa de desplazamiento .....</b>	72
3.2.1.	<i>Caracterización del alcance máximo de telemetría .....</i>	72
3.2.2.	<i>Caracterización de las trayectorias por punto del planificador .....</i>	73
3.3.	<b>Análisis de costo .....</b>	74
<b>CONCLUSIONES.....</b>		77
<b>RECOMENDACIONES.....</b>		79
<b>BIBLIOGRAFÍA</b>		
<b>ANEXOS</b>		

## ÍNDICE DE TABLAS

<b>Tabla 1-1:</b> Parámetros calidad del agua y sus rangos aceptables.....	5
<b>Tabla 2-1:</b> Condiciones de temperatura (°C) .....	7
<b>Tabla 3-1:</b> Características sensores de temperatura.....	8
<b>Tabla 4-1:</b> Ventajas y desventajas de los sensores de temperatura.....	9
<b>Tabla 5-1:</b> Características de Arduinos.....	11
<b>Tabla 6-1:</b> Características plataformas IoT.....	15
<b>Tabla 7-1:</b> Características de las baterías recargables.....	21
<b>Tabla 8-1:</b> Emisoras.....	23
<b>Tabla 1-2:</b> Características de Pixhawk.....	33
<b>Tabla 2-2:</b> Condiciones de temperatura.....	33
<b>Tabla 3-2:</b> Comparativa de sensores de temperatura.....	34
<b>Tabla 4-2:</b> Concentración de oxígeno y efectos sobre el camarón.....	35
<b>Tabla 5-2:</b> Características Sensor de oxígeno.....	35
<b>Tabla 6-2:</b> Efectos del PH al camarón .....	37
<b>Tabla 7-2:</b> Comparativa de tarjetas de comunicación.....	38
<b>Tabla 8-2:</b> Consumo energético de cada elemento.....	43
<b>Tabla 9-2:</b> Especificaciones y configuraciones.....	61
<b>Tabla 1-3:</b> Mediciones realizadas entre el XL150 y el DS18B20.....	64
<b>Tabla 2-3:</b> Mediciones realizadas entre el XL150 y el E-201.....	66
<b>Tabla 3-3:</b> Mediciones realizadas entre el YSI A500 y el E-201.....	67
<b>Tabla 4-3:</b> Estabilidad de la etapa de supervisión .....	69
<b>Tabla 5-3:</b> Tabla de costos.....	75

## ÍNDICE DE FIGURAS

<b>Figura 1-1:</b> PIXHAWK.....	10
<b>Figura 2-1:</b> Modelos Rasberry Pi.....	12
<b>Figura 3-1:</b> Tecnologías totales.....	13
<b>Figura 4-1:</b> Arquitectura Android.....	16
<b>Figura 5-1:</b> SONOBOT, vehículo de supervisión.....	18
<b>Figura 6-1:</b> Trayectorias punto a punto.....	19
<b>Figura 7-1:</b> Toma de muestras y análisis.....	24
<b>Figura 1-2:</b> Concepción universal del sistema.....	27
<b>Figura 2-2:</b> Diagrama de bloques etapa de supervisión .....	28
<b>Figura 3-2:</b> Etapa de almacenamiento.....	29
<b>Figura 4-2:</b> Etapa de desplazamiento.....	30
<b>Figura 5-2:</b> Arduino Mega 2560.....	31
<b>Figura 6-2:</b> Raspberry Pi 3.....	32
<b>Figura 7-2:</b> Sensor Atlas Scientific EZO-DO.....	36
<b>Figura 8-2:</b> Sensor pH E-201.....	38
<b>Figura 9-2:</b> Motor brushless y ESC de 45A.....	39
<b>Figura 10-2:</b> Esquema de conexiones.....	40
<b>Figura 11-2:</b> Esquema de conexión GPS.....	41
<b>Figura 12-2:</b> Telemetría.....	41
<b>Figura 13-2:</b> Módulo de telemetría.....	42
<b>Figura 14-2:</b> Batería Li-Po.....	44
<b>Figura 15-2:</b> ASV C-CAT3.....	44
<b>Figura 16-2:</b> Casco principal.....	45
<b>Figura 17-2:</b> Tapa del casco.....	45

<b>Figura 18-2:</b> Flotadores.....	46
<b>Figura 19-2:</b> Esquema de conexión, etapa de supervisión.....	47
<b>Figura 20-2:</b> Esquema de conexión, etapa de transmisión.....	48
<b>Figura 21-2:</b> Conexión de periféricos a Pixhawk.....	50
<b>Figura 22-2:</b> Diagrama de flujo del sistema de supervisión.....	53
<b>Figura 23-2:</b> Cadena de caracteres enviada por puerto serial.....	54
<b>Figura 24-2:</b> Diagrama de flujo etapa de transmisión.....	56
<b>Figura 25-2:</b> HMI ThingSpeak.....	57
<b>Figura 26-2:</b> Diagrama de flujo etapa de desplazamiento.....	58
<b>Figura 27-2:</b> Diagrama de flujo App.....	59
<b>Figura 28-2:</b> App móvil.....	60
<b>Figura 29-2:</b> ThingSpeak base de datos .....	60
<b>Figura 1-3:</b> Prototipo final implementado.....	62
<b>Figura 2-3:</b> Equipo XL150, midiendo temperatura.....	63
<b>Figura 3-3:</b> Equipo XL150, midiendo pH.....	65
<b>Figura 4-3:</b> Equipo YSI A500.....	67
<b>Figura 5-3:</b> Comparación entre datos de almacenamiento interno y enviados a la nube.....	70
<b>Figura 6-3:</b> ThingSpeak y App móvil.....	71
<b>Figura 7-3:</b> Historial de recorrido del sistema RASSC.....	72
<b>Figura 8-3:</b> Distancia máxima de conectividad por telemetría.....	73
<b>Figura 9-3:</b> Way-points del planificador.....	74

## ÍNDICE DE ANEXOS

**ANEXO A:** EZO-DO Atlas-Scientific Datasheet

**ANEXO B:** Hoja de especificaciones Sensor de temperatura DS18B20

**ANEXO C:** Hoja de especificaciones Sensor pH E-201

**ANEXO D:** DATASHEET PIXHAWK

**ANEXO E:** Hoja de especificaciones De Arduino Mega 2560

**ANEXO F:** DATASHEET RASPBERRY PI 3

**ANEXO G:** GPS U-BLOX NEO 6M

**ANEXO H:** FLY-SKY fs-i6s

**ANEXO I:** Planos generados de Slide de Autodesk

**ANEXO J:** Construcción de la estructura

**ANEXO K:** Programación Etapa de supervisión

**ANEXO L:** Código Programación Android Studio

**ANEXO M:** Capturas Aplicación móvil

**ANEXO N:** Programación Etapa de Trasmisión

**ANEXO Ñ:** Archivo generado por la plataforma ThingSpeak

**ANEXO O:** Historial de variables en plataforma ThingSpeak

## ÍNDICE DE ABREVIATURAS

**DO:** Oxígeno disuelto.

**pH:** Potencial de hidrógeno.

**RASSC:** Robot acuático de superficie para supervisión de camaroneras.

**API:** Application programming interface.

**ASV:** Autonomous Surface Vehicle.

**GPS:** Sistema de Posicionamiento Global.

**2D:** Dos dimensiones.

**SQL:** Structured Query Language.

**RTD:** Resistencia temperatura detector.

**WIFI:** Wireless Fidelity.

**RF:** Radio frecuencia.

**WLAN:** Wireless Local Area Network.

**SI:** Sistema Informático.

**IoT:** Internet of Things.

**ESC:** Electronic Speed Controller.

**Li-Po:** Litio Polímero.

**HMI:** Human Machine Interface.

**USB:** Bus Universal en Serie.

**PPM:** Pulse Position Modulation.

**CAD:** Computer Aided Design.

## RESUMEN

El proyecto tuvo como finalidad generar un impacto favorable en el desarrollo del camarón, donde la variación drástica de parámetros produce efectos nocivos en la salud de la especie. La implementación de dicho proyecto se basa en un prototipo robot acuático de superficie, que sirvió para la supervisión de variables de la calidad del agua dentro de piscinas camaroneras. Se caracterizó por ser un vehículo acuático que puede ser trasladado a cualquier zona planificada ampliando la cobertura de la recolección de datos. Este a su vez, permite la supervisión de variables como: temperatura, oxígeno disuelto y pH. Su arquitectura consta de tres etapas: supervisión, transmisión y desplazamiento. La primera etapa está constituida por tres sensores: temperatura (ds18b20), pH (E-201-C) y oxígeno disuelto (DO-EZO-Atlas-Scientific), con una tarjeta programable Arduino Mega 2560 para lectura de señales análogas de estos, y la segunda por una Raspberry Pi 3, que a través del puerto serial toma los datos recolectados por el Arduino y sistema de posicionamiento global (GPS) para almacenarlos y posteriormente enviarlos a una base de datos. El desplazamiento está integrado por una tarjeta Pixhawk a la que se conectan varios periféricos como: GPS, motores brushless etc., que permitirán desplazarse al prototipo por los puntos planificados. La información obtenida es visualizada a través de una aplicación móvil que descarga los datos del servidor ThingSpeak. Según las pruebas realizadas se comprobó que supera los 100 m de alcance de telemetría y los sensores no presentan un error absoluto mayor a los determinados en las hojas de especificaciones del fabricante. Finalmente, se concluye que el prototipo es considerado una pequeña estación móvil de supervisión de calidad del agua, funcional y confiable. Se recomienda realizar un estudio más profundo con la base de datos generada para incluir reportes de mapeo 2D de las variables en cuestión.

**PALABRAS CLAVE:** <PISCINAS ACUÍCOLAS>, <CAMARÓN BLANCO (*PENAEUS VANNAMEI*)>, <SUPERVISIÓN DE VARIABLES ACUÍCOLAS DEL AGUA>, <CALIDAD DEL AGUA>, <VEHÍCULO AUTÓNOMO DE SUPERFICIE (ASV)>, <PIXHAWK (SOFTWARE-HARDWARE)>, <TRAYECTORIAS PUNTO A PUNTO>, <APLICACIÓN MÓVIL>.

## SUMMARY

The project had as purpose to generate a favourable impact in the shrimp growth, where the drastic variation of parameters produces harmful effects in the specie health. The implementation of the mentioned project is based on a surface aquatic robot prototype, that served for the supervision of water quality variables within the shrimp pools. It was characterized for being an aquatic vehicle which can be moved to any planned zone expanding the data collection coverage. This, in turn, allows the supervision of variables such as: temperature, dissolved oxygen, and pH. Its architecture consists of three stages: supervision, transmission and displacement. The first stage is constituted for three sensors: temperature (ds18b20), pH (E-201-C) and dissolved oxygen (DO-EZO-Atlas-Scientific), with a programmable card Arduino Mega 2560 for the reading of analogue signals of these, and the second by a Raspberry Pi 3, that through a serial port takes the collected data by the Arduino and global positioning system (GPS) to store them and later send them to a data base. The displacement is integrated by a Pixhawk card in which some peripherals are connected to such as: GPS, brushless motors, etc., which will allow to the prototype to displace by planned locations. The obtained information is displayed through a mobile application that downloads the data from the ThingSpeak server. According to the tests performed it was verified that it exceeds the 100 m of telemetry range and the sensors do not show any absolute errors greater to the determined in the maker specifications sheets. Finally, it is concluded that the prototype is considered a little mobile station of water quality supervision, functioning and reliable. It is recommended to realize a closer study with the data base generated to include 2D mapping reports of the variables in question.

**KEY WORDS:** <AQUACULTURE POOLS>, <WHITE SHRIMP (*PENAEUS VANNAMEI*)>, <WATER AQUACULTURE VARIABLES SUPERVISION>, <WATER QUALITY>, <AUTONOMOUS SURFACE VEHICLE (ASV)>, <PIXHAWK (SOFTWARE-HARDWARE)>, <POINT-TO-POINT TRAJECTORIES>, <MOBILE APPLICATION>.

## INTRODUCCIÓN

En la actualidad, dentro del Ecuador, el camarón, se posiciona en el segundo lugar, después del banano, como el producto no petrolero más grande de exportación. Esto se evidencia con lo ocurrido en el año 2014, cuando Ecuador exportó medio millón de libras de camarón, del cual se obtuvo un total de 1520 millones de dólares. (Trujillo, Rivera y Hardy, 2017)

En el mismo territorio, existen alrededor de unas 210.000 ha dedicadas al camarón; éstas se distribuyen a continuación: el 60% en Guayas, el 15% en El Oro, el 9% en Esmeraldas, el 9% se ubica en Manabí y finalmente el 7% se encuentra en Santa Elena. (Revista Líderes, 2017). El cultivo de camarón asemeja a la agricultura, debido a que, se compone de tres etapas: iniciando por siembra, seguido por el crecimiento y finaliza con la cosecha. De los 3 procesos el sector con más prioridad es la cosecha, etapa delegada de del empaquetado y control de calidad del producto.

La acuicultura intensiva es una forma de cultivo actualizada que se desarrolla con agilidad en muchos países. Recientemente, las personas ponen en práctica esta forma de siembra, debido a sus ventajas, con tiempo ilimitado, bajo consumo y un grado muy elevado de utilización de agua. La clave para el cultivo intensivo de *Penaeus vannamei* (camarón blanco) es conservar el equilibrio ecológico del agua a través de los dispositivos de medición. Comúnmente, trabajar con los medios tradicionales en el medio ambiente acuático es un trabajo pesado, más bien, una pérdida de tiempo.

Es por ello, que los sistemas de monitoreo del entorno acuático se apoyan en tecnología de fusión de información multisensorial, razón por ello, estos se convierten en uno de los medios más viables en el mercado. Los sistemas de monitoreo del ambiente acuático principalmente tienen como objetivo principal, controlar el valor de factores de agua como lo es la temperatura, el pH, el oxígeno disuelto (DO), la salinidad, y que estén ubicados en el rango permitido; es así como se proporciona un ecosistema acuático óptimo para el crecimiento de la especie. (Xiaojing, Ming y Jiang, 2009)

Por otro lado, pero en el mismo tema, estudiantes de la UTMACH han realizado proyectos sobre el control de temperatura mediante Arduino, sensor LM35 y una base de datos en SQL, en los cuales se efectúa la supervisión de un punto de toda la piscina acuícola en camaroneras de la provincia de El Oro, el sistema permite obtener datos sobre el control continuo de la temperatura del agua; confirmando que esta sea la conveniente para el desarrollo de la larva (camarón), evitando transmisión de enfermedades con cuidado óptimo sobre los niveles de producción. Dicha implementación no es la más factible, ya que la muestra tomada por el sensor es de un solo punto

fijo en el área de la piscina, por lo cual el presente proyecto tiene como finalidad ampliar el rango de medición para abarcar mayor área supervisada. (Novillo Vicuña y Vaca Simbala, 2015) Es por ello, que la construcción de un ASV permite el proceso de obtención de datos, el cual proporcionará beneficios a la industria camaronera al poseer un mayor control de las variables del crecimiento del camarón.

## **FORMULACIÓN DEL PROBLEMA**

¿Cómo construir un prototipo de vehículo acuático de superficie no tripulado para supervisión de producción acuícola en camaroneras?

## **SISTEMATIZACIÓN DEL PROBLEMA**

¿Cuáles son las variables de calidad del agua que intervienen en la camaronicultura?

¿Cuáles son las especificaciones que debe cumplir el ASV en la supervisión de las piscinas acuícolas?

¿Cuáles son los equipos hardware y software que permita adquirir parámetros del entorno de crecimiento del camarón?

¿El prototipo construido cumple con los requerimientos planteados en la investigación?

## **OBJETIVOS GENERALES**

- Construir un prototipo de vehículo acuático de superficie no tripulado para supervisión de producción acuícola en camaroneras.

## **OBJETIVOS ESPECIFICOS**

- Realizar un estudio de las variables de calidad del agua que intervienen en la camaronicultura.
- Diseñar el prototipo considerando los requerimientos mínimos para la supervisión de piscinas acuícolas.
- Identificar el equipo software y hardware que permita la adquisición de los parámetros del entorno de crecimiento del camarón.
- Realizar pruebas y analizar resultados del funcionamiento del ASV.

# CAPÍTULO I

## 1. MARCO TEÓRICO

En este apartado se detalla la información con argumentos de investigadores y obras publicadas en revistas científicas, además se presenta el estado del arte de la presente investigación.

### 1.1. Piscinas Acuícolas

Definir a la acuicultura, es mencionar la crianza de especies animales o vegetales que se desenvuelven en un medio acuático, en condiciones biológicas controladas. Los acuicultores, a través de la realización de diversas investigaciones, han logrado conocer sobre el proceso biológico de una variedad de especies, con la finalidad de ser incrustadas en el periodo comercial y posteriormente ser parte de la explotación industrial. (Bardach, 1990), (Carretero, 2002)

Es por ello, que se usa las piscinas acuícolas, que no son más que áreas donde se sitúa una cantidad adecuada de agua marina, la cual, va a acoger un ambiente apropiado, donde se simulan los procesos biológicos de las especies. Las piscinas camaroneras, se encuentran diseñadas y construidas en zonas litorales, con superficies de 1 a 10 ha y una profundidad de 1 a 1.2 m. (Martínez Córdova, 2009)

Por otro lado, para los piscicultores, los factores clave en la producción acuícola son: el manejo, inspección y tratamiento del agua dentro del estanques. Es imprescindible un estudio de los parámetros de calidad del agua para conocer los puntos críticos de los parámetros. La aclimatación de los estanques acuícolas corresponde a las propiedades físico, químicas y biológicas del agua, para establecer las variables principales que son: temperatura, salinidad, pH, oxígeno disuelto. (Lopez, 2006)

## 1.2. Variables de Proceso

Uno de los elementos principales para la acuicultura es el agua, éste es el medio que se utiliza para el desarrollo de los organismos acuícolas, y su eficacia dependerá en gran medida de su desarrollo. Razón por ello, es indispensable, saber seleccionar el agua que certifique que en ese aspecto no habrá ningún tipo de inconvenientes. (Martínez Córdova, 2015)

El desbalance de los niveles de pH y de la temperatura, aumenta la toxicidad del sulfuro de hidrógeno y a su vez del amoníaco. Para ello mantener los niveles balanceados de los parámetros de calidad de agua es imprescindible, tanto para la salud, como para el crecimiento de cuerpos de cultivo. Se encomienda mantener un seguimiento y un cálculo de los parámetros de calidad del agua en rutinas habituales.

Según (Aquafeed, 2018), los parámetros más significativos de calidad del agua son: oxígeno, pH, temperatura, salinidad, turbidez y compuestos de Nitrógeno. Estos a su vez, se describen con percepción de cómo estos parámetros se correlacionan entre sí. La Tabla 1-1 da un vistazo de los parámetros de calidad del agua con sus rangos aceptables.

**Tabla 1-1:** Parámetros de calidad de agua y sus rangos aceptables

<b>Parámetros</b>	<b>Rango aceptable</b>
<b>Temperatura</b>	20 a 32 °C
<b>Salinidad</b>	15 a 30 ppmil
<b>Oxígeno disuelto (DO)</b>	> 3 mg/l
<b>pH</b>	6 a 8 (dependiendo la especie)
<b>Alcalinidad (CaCO<sub>3</sub>)</b>	100 a 400 mg/l
<b>Dureza</b>	100 a 400 mg/l
<b>Nitrógeno amoniacal ionizado (NH<sub>4</sub><sup>+</sup>)</b>	0.2 a 2 mg/l
<b>Nitrógeno amoniacal no ionizado (NH<sub>3</sub>)</b>	< 0.1 mg/l
<b>Nitratos</b>	0.2 a 10 mg/l
<b>Nitritos</b>	< 0.3 mg/l
<b>Fósforo</b>	0.005 a 0.2 mg/l
<b>Sólidos suspendidos totales</b>	< 200 mg/l

Fuente:(Martínez Córdova, 2015)

Realizado por: Carlos Estévez, José Rodas, 2019.

### **1.2.1. Oxígeno Disuelto**

Mantener buenos niveles de DO en el agua es fundamental para una producción eficaz. El oxígeno (O<sub>2</sub>) tiene un dominio directo en la ingesta de alimentos, la tenacidad de la enfermedad y su metabolismo. Un nivel subóptimo es muy abrumador tanto para peces como para camarones. Es por ello, que se da la importancia de mantener el DO a niveles que van por encima de los 4.0 ppm.

En el nivel sub-óptimo, el organismo puede sobrevivir, pero en su desempeño general se ve el deterioro provocado, ya que, se produce un desgaste su energía, en moverse buceando sobre las capas superiores, donde se ubica un poco más de oxígeno (Martínez Córdova, 2015).

Cuando se alimenta a los peces y los camarones, la necesidad de oxígeno sube, esto ocurre debido al incrementado del gasto de energía (denominada también como acción dinámica específica). Para enfrentar este inconveniente, se presentan varias medidas: Una de ellas, es que los aireadores de rueda con paletas, estos permiten el rompiendo del agua en diminutas gotas y aumenta el contacto de la superficie con el aire. Los aireadores con aspirador permiten remitir aire dentro del agua a través de una entrada y un aspa; otra razón para la ventilación es el transporte de agua aireada a través del tanque de reserva (Aquafeed, 2018).

### **1.2.2. Temperatura**

Todos los seres vivos cuentan con un rango óptimo de temperatura, donde desarrollan sus funciones metabólicas; fuera de este funcionamiento, el desarrollo no es el más adecuado (Martínez Córdova, 2015) . Una de las consecuencias se basa en el metabolismo de camarones y en ocasiones de los peces; sus índices de alimentación e incluso el nivel de toxicidad de Amoníaco. La temperatura marca una ruta directo en los índices de respiración de la biota (consumo de O<sub>2</sub>) e influencia la disolución del O<sub>2</sub> (agua más cálida contiene menos O<sub>2</sub> que el agua más fresca).

La temperatura no puede ser vigilada desde una cisterna. Los animales acuáticos son aquellos que modifican la temperatura de sus cuerpos en relación al medioambiente, e incluso son sensibles a las variaciones de temperatura drásticas. La temperatura es diferente para cada especie y ésta se presenta en diferentes rangos y condiciones. (Tabla 2-1). Es por ello, que se debe adaptar a los camarones y peces progresivamente, es decir no transferirlos directamente de un tanque a otro (Aquafeed, 2018).

**Tabla 2-1:** Condiciones de temperatura (°C)

<b>Especies</b>	<b>Temperatura letal más baja</b>	<b>Temperatura preferida</b>	<b>Temperatura letal más alta</b>
Tilapia de Nilo	8 – 12	31 – 36	42
Bagre de Canal	9	22 – 29	37
Langostino tigre	14	25 – 30	36
<b>Camarón Blanco</b>	14	>20	40

Fuente:(Aquafeed, 2018)

Realizado por: Carlos Estévez, José Rodas, 2019.

### **1.2.3. pH**

EL pH es aquel que está relacionado con otros indicadores como la dureza, el oxígeno reducido, la alcalinidad, la productividad de fitoplancton, la potencia, entre otros. (Martínez Córdova, 2015) Es por ello, que es importante mantener el pH invariable a un nivel seguro porque esto perturba la desintegración de los alimentos e incluso otros procesos fisiológicos de los organismos.

Seguidamente a eso están las consecuencias, que pueden ser: crear estrés, aumentar la probabilidad de desarrollar enfermedades, bajar los niveles de producción y causar la muerte. Los signos de un pH subóptimo son: mucosidad aumentada en las agallas del pez, natación inusual, aletas raídas, daños en los ojos, así como también la falta de crecimiento del fitoplancton y del zooplancton. Los niveles óptimos de pH en la cisterna deben estar en el rango de 7.5 – 8.5. Las aguas de mediana alcalinidad están más apagadas y hay un grado menor de variación de pH (Aquafeed, 2018).

### **1.2.4. Salinidad**

Es aquella que representa la unión total de iones inorgánicos diluidos. Este juega un rol significativo para el desarrollo de organismos de cultivo a través de la osmorregulación de sales minerales del cuerpo en el agua adyacente. Para que el crecimiento de los organismos este en un rango adecuado de salinidad, el agua debe estar en un constante mantenimiento. En ocasiones la salinidad es demasiado elevada, las piscinas de los peces y camarones comenzarán a disipar agua al medioambiente.

Los cambios drásticos de salinidad alcanzan a alterar la fauna del fitoplancton y sus espesores de población, y llevar a desequilibrio del ecosistema. Debido a esto, no se recomienda reducir la salinidad por más de 5 ppt, a cada vez de intercambio de agua (Aquafeed, 2018).

La mayoría de los camarones comerciales son eurihalinos, es decir, resisten un rango más o menos alto de salinidad, sin embargo, se aprecian mejores resultados dentro de un rango más estrecho entre 15 y 35 ppmil (Martínez Córdova, 2015).

### 1.3. Sensores de las variables a supervisar

#### 1.3.1. Sensor de temperatura

Entre los sensores tradicionales encontramos los termistores, RTD y termopares que son confiables dada la aplicación a la que se los incluya, en la Tabla 3-1 se especifican sus principales características.

**Tabla 3-1:** Características sensores de temperatura.

<b>Tipo de sensor</b>	<b>Termistor</b>	<b>RTD</b>	<b>Termopar</b>
<b>Rango de temperatura</b>	De -100 °C a +325 °C	De -200 °C a +650 °C	De +200 °C a +1750 °C
<b>Precisión</b>	De 0.05 °C a 1.5 °C	De 0.1 °C a 1 °C	De 0.5 °C a 5 °C
<b>Estabilidad duradera a 100°C</b>	0.2 °C/año	0.05 °C/año	Variable
<b>Linealidad</b>	Exponencial	Relativamente lineal	No lineal
<b>Potencia requerida</b>	Voltaje o corriente constante	Voltaje o corriente constante	Autoalimentado
<b>Tiempo de respuesta</b>	Rápido: de 0.12 s a 10 s	Generalmente lento: de 1 s a 50 s	Rápido: de 0.1 s a 10 s
<b>Susceptibilidad al ruido eléctrico</b>	Raramente susceptible, solo alta resistencia	Raramente susceptible	Susceptible/Compensación de soldadura en frío
<b>Costo</b>	De bajo a moderado	Alto	Bajo

Fuente: (Schweber, 2019)

Realizado por: Carlos Estévez, José Rodas. 2019

**Tabla 4-1:** Ventajas y desventajas de los sensores de temperatura

<b>Tipo de sensor</b>	<b>Ventajas</b>	<b>Desventajas</b>
<b>Termopar</b>	Rango de temperatura Autoalimentado No se auto calienta Resistentes	Compensación de soldadura en frío Precisión Estabilidad Conductores de extensión de termopar
<b>RTD</b>	Precisión Estabilidad Linealidad	Error de resistencia de los conductores Tiempo de respuesta Resistencia a vibraciones
<b>Termistor</b>	Sensibilidad Precisión Costo y bajo costo	No linealidad Autocalentable Rangos limitados

Fuente: (Schweber, 2019)

Realizado por: Carlos Estévez, José Rodas. 2019

### **1.3.2. Sensor de oxígeno**

El sistema consiste en una célula de dos electrodos, el cual se compone de una membrana permeable al oxígeno y, por otro lado, un electrolito. Los electrodos son un ánodo de plata y un cátodo de un metal noble, generalmente suele ser de un material denominado: platino. El oxígeno se difunde a través de la membrana permeable hacia dentro del electrodo, en el cual se producen las resistencias (Aguas - Industria).

Es por ello, que se han determinado la existencia de dos tipos de sensores:

- Sensor polarográfico, se emplea un voltaje para consumir la medida.
- Sensor galvánico, donde la célula genera el propio voltaje para efectuar la medida.

### **1.3.3. Sensor de pH**

El pH por sus siglas hace referencia al potencial hidrógeno de una solución, define su alcalinidad. La escala de medición inicia desde el 0 y culmina en el 14, con los valores d que van del 0 al 6 ácido, y el 7 neutro; finalmente del 8 al 14 alcalinos.

La estructura se compone de un sensor de vidrio y un tubo de referencia. Una sonda de pH es aquella que mide la actividad de los iones de hidrógeno por medio de un proceso ocasionando una pequeña cantidad de tracción entre el sensor y el tubo de referencia. La tensión, por otro lado, es aquella que se mide como una corriente eléctrica, mientras el voltaje, generado en esta corriente, se configura en un valor de pH. De este modo se realizan variadas mediciones de una variedad de líquidos, semisólidos o sólidos (El Crisol, s.f.).

## 1.4. Tarjetas programables

### 1.4.1. Pixhawk

El Pixhawk no es más que un piloto automático en el módulo de alto rendimiento, el cual es ubicado en un ala fija, multi rotores, helicópteros, automóviles, barcos etc., e incluso a cualquier otra armazón robótica traslacional. Este está dirigido a la investigación de alto nivel, las necesidades de inclinados e industrias y combina la funcionalidad del PX4FMU + PX4IO. (PX4 Autopilot, s.f.)



**Figura 1-1: PIXHAWK**

**Realizado por:** Carlos Estévez, José Rodas, 2019.

### 1.4.2. Arduino

**Tabla 5-1:** Características de Arduinos

Característica de Arduino	UNO	Mega 2560	Leonardo	DUE
<b>Tipo de microcontrolador</b>	Atmega 328	Atmega 2560	Atmega 32U4	AT91SAM3X8E
<b>Velocidad de reloj</b>	16 MHz	16 MHz	16 MHz	84 MHz
<b>Pines digitales de E/S</b>	14	54	20	54
<b>Entradas analógicas</b>	6	16	12	12
<b>Memoria de programa (Flash)</b>	32 kb	256 kb	32 kb	512 kb
<b>Memoria de datos (SRAM)</b>	2 kb	8 kb	2.5 Kb	96 kb
<b>Memoria auxiliar (EEPROM)</b>	1 kb	4 kb	1 kb	0 kb
<b>Pines positivos</b>	1 de 5 V y 1 de 3.3 V	3 de 5 V y 1 de 3.3 V	1 de 5 V y 1 de 3.3 V	3 de 5 V y 1 de 3.3 V
<b>Pines negativos</b>	3 GND	5 GND	3 GND	5 GND
<b>Precio</b>	10	20	15	30

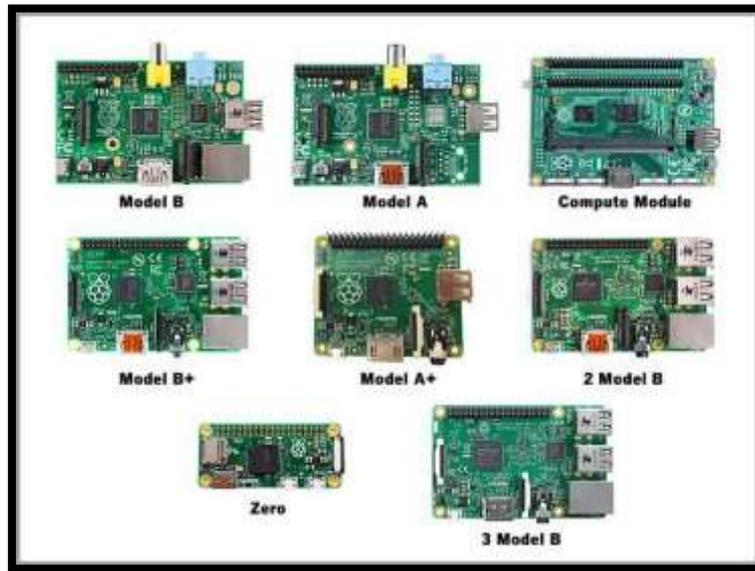
Fuente: (García González, 2015)

Realizado por: Carlos Estévez, José Rodas, 2019.

### 1.4.3. Raspberry

Representa un pequeño ordenador personal de bajo coste. Esta placa fue desarrollada en Reino Unido con el objetivo de incentivar al mundo de la informática por medio de la enseñanza de la programación en numerosos trabajos como robótica, realizar servidores, entre otras cosas.

Utiliza el sistema operativo Linux, existen varios modelos que se acoplan a los requerimientos de hardware y software para diseños de sistema a implementar. Consta de puertos tanto de entradas como de salidas, facilitando aplicaciones en las que se necesiten control como por ejemplo en el campo de la domótica (RASPBERRY PI, 2017). La Figura 2-1 muestra algunos modelos de tarjetas Raspberry Pi existentes en el mercado.

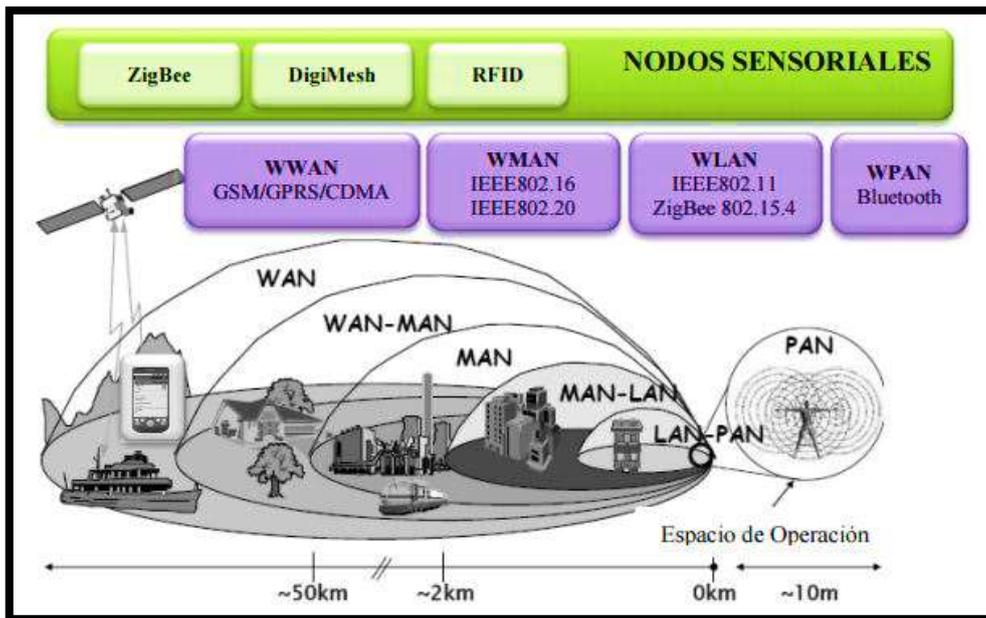


**Figura 2-1: Modelos Raspberry Pi**  
Fuente: (RASPBERRY PI, 2017)

### 1.5. Transmisión de datos

La transmisión de datos permite transmitir y recibir información entre dispositivos que se encuentren dentro de la red, esto es fundamental dentro de sensores inalámbricos, donde los datos obtenidos son transmitidos mediante ondas electromagnéticas, sin la necesidad de integrar un canal físico (Pico, 2012).

Las bandas de frecuencia mayormente utilizadas en redes sensoriales, que existen actualmente son Wi-Fi, Radio Frecuencia, Bluetooth (Fernández, et al. 2009). En su totalidad las tecnologías de comunicación se aprecian en la figura 3-1.



**Figura 3-1: Tecnologías totales**

Fuente: (PICO VALENCIA, Pablo, 2012)

### Wi-Fi

El Wi-Fi trabaja en una frecuencia de 2.4 GHz y está basado en el estándar IEEE 802.11, Presenta gran compatibilidad en estos tiempos y posee una cobertura geográfica expandible, fácil instalación y movilidad (Pico, 2012).

### Radio frecuencia (RF)

Trabaja entre frecuencias de 433 MHz y 2,4 GHz. Otras frecuencias son utilizadas como por ejemplo infrarrojos y en comunicaciones ópticas a través de láser (Fernández, et al. 2009).

Comúnmente utilizado en la televisión digital terrestre, por poseer una facilidad para atravesar obstáculo debido a su característica omnidireccional.

### Bluetooth

Trabaja en frecuencias de los 2.4 GHz, pero a diferencia del Wi-Fi esta posee un corto alcance dado que es utilizado esencialmente en redes de área personal o también llamadas WPAN, remplazando la conexión de cables al momento de transmitir información entre dispositivos que poseen esta tecnología (Pico, 2012).

## 1.6. Base de datos

Base de datos se denomina a la representación de los conjuntos de entes, es decir, es la instancia correspondiente a las diferentes entidades tipo del SI y de sus interrelaciones. Este conjunto estructurado de datos debe ser utilizado a manera compartida por muchos usuarios en una variedad de tipos. Según (Camps Paré et al., 2005) ,las bases de datos es la manera más usual al almacenar datos. Estos se pueden evidenciar desde grandes aplicaciones con multiusuarios, hasta los teléfonos inteligentes; e incluso, ciertos artefactos como agendas electrónicas, utilizan la base de datos para certificar la integridad de estos y proporcionar un mejoramiento en la labor tanto de usuarios como de los programadores que dan uso de ella.

Es por ello que, (Capacho y Nieto, 2017) menciona que la toma de decisiones apropiadas en las organizaciones se basa en informaciones acumuladas, las cuales deben cumplir ciertas características como, por ejemplo: deben estar estandarizadas, renovadas y con disponibilidad inmediata.

Como lo dijo Richard Roger, *“La programación para la nube se convertirá en la forma principal de construir la infraestructura de los servicios.”* (Rodger, 2012) . Razón por la cual, se utiliza nuevas tecnologías, como son las plataformas del Internet (IoT, por sus siglas en ingles), las cuales permiten acumular información en la nube, la cual sirve para brindarnos la disponibilidad de la información desde cualquier punto del universo.

### 1.6.1. Plataformas IoT

**Tabla 4-1:** Características plataformas IoT

	<b>ThingSpeak</b>	<b>Thingier.io</b>	<b>Samsung Artik Cloud</b>	<b>aRest Framework</b>
<b>Numero de datos gratuitos</b>	250 000 mensuales  (cada 15 segundos)	-	100 000 mensuales (cada 10 minutos)	1000 mensuales
<b>Numero de Dashboards</b>	8	4	6	4
<b>Dispositivos conectados</b>	4	30	5	5
<b>Tiempo histórico de acceso a datos</b>	Ilimitado	30 días	3 meses	-
<b>Compatibilidad con Hardware</b>	Alta	Media	Media	Media

Fuente: (Garrido, 2015)

Realizado por: Carlos Estévez, José Rodas, 2019.

### 1.7. Aplicaciones móviles

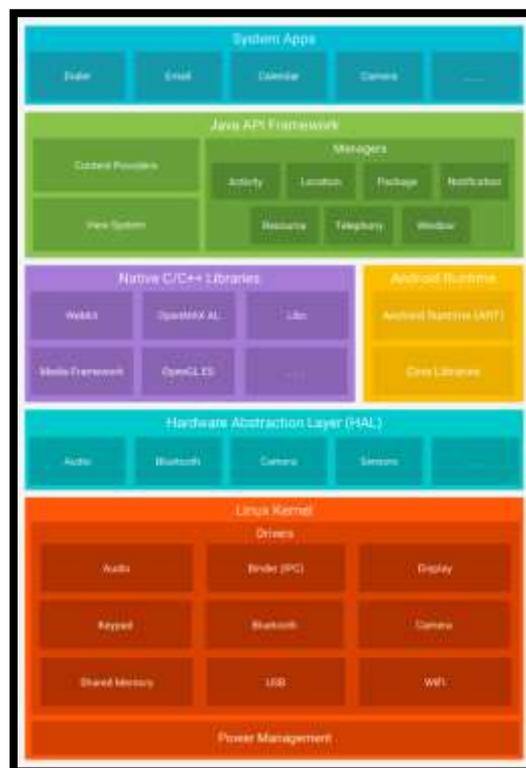
La telefonía móvil, durante la última década, ha generado una evolución en el mundo, gracias al desarrollo de equipos sofisticados. De tal manera que dicha evolución móvil ha logrado mejorar la calidad de vida de las personas, ya que, permiten brindar soluciones a las cosas cotidianas, a través del desarrollo de aplicaciones móviles, las cuales brindan un amplio servicio, el cual integra la comunicación con la información entre individuos (Nolasco Valenzuela, 2013).

### 1.7.1. Android

Actualmente se ha desarrollado una variedad de innovaciones tecnológicas, entre ellas se ubica el sistema operativo para dispositivos móviles, el cual fue apoyado en Linus Kernel 2.6; éste no es más que un conjunto de software, que contiene un sistema operativo, middleware y varias aplicaciones. (Nolasco Valenzuela, 2013) determina que uno de sus principales aspectos positivos es que posee una licencia libre y un código abierto; lo cual permite generar el interés por parte del consumidor de aplicaciones móviles. En los últimos años, Android, que es uno de los más grandes sistemas operativos, ha logrado cubrir localidades en un 93% del mercado.

#### 1.7.1.1. Arquitectura

El mismo autor menciona sobre la estructura Android, el cual se encuentra básicamente constituido por un sistema operativo, librerías (bajo C y C++), framework para el desarrollo de aplicaciones e incluye una suite de aplicaciones iniciales, esto se puede apreciar en la siguiente imagen. (Nolasco Valenzuela, 2013)



**Figura 4-1: Arquitectura android**

Fuente:(Google Developers, 2018)

Según (Nolasco Valenzuela, 2013) el diseño cuenta con las siguientes características:

- Los componentes básicos se pueden sustituir cómodamente por otros.
- Facilita el uso de base de datos.
- Permite la representación de gráficos 2D y 3D.
- Cuenta con un entorno de libre desarrollo mediante un SDK.
- Resiste un elevado número de formatos.

### **1.8. Vehículos subacuáticos no tripulados (ASV)**

Los drones o vehículos subacuáticos no tripulados (por sus siglas en inglés autonomous vehicles of Surface), hace referencia a cualquier vehículo cuya capacidad de maniobrar sobre el agua sin un manejo físico humano. Estos vehículos a su vez se dividen en dos categorías, vehículos subacuáticos operados a distancia (ROV), controlados por un ejecutor humano, y vehículos subacuáticos autónomos (AUV), sin intervención humana directa. Los vehículos autónomos subacuáticos (AUV) son vehículos robóticos programables, según su diseño, derivan, conducir o deslizarse a través del océano sin el control en tiempo real de los operadores humanos. Algunos (AUV) se comunican con los ejecutores continuamente a través de balizas acústicas submarinas para permitir un óptimo nivel de control.

Los (AUV) permite a los científicos, ampliar el campo de la experimentación dentro de un barco de superficie; mientras que en el vehículo no está recogiendo datos existe en otra parte de la superficie. Algunos de ellos, toman decisiones por sí solos, uno de estos son el cambio de su perfil de misión en función de los datos climáticos obtenidos a través de los sensores que están en marcha. (Woods Hole Oceanographic Institute, s.f.)



**Figura 5-1: SONOBOT, vehículo de supervisión.**

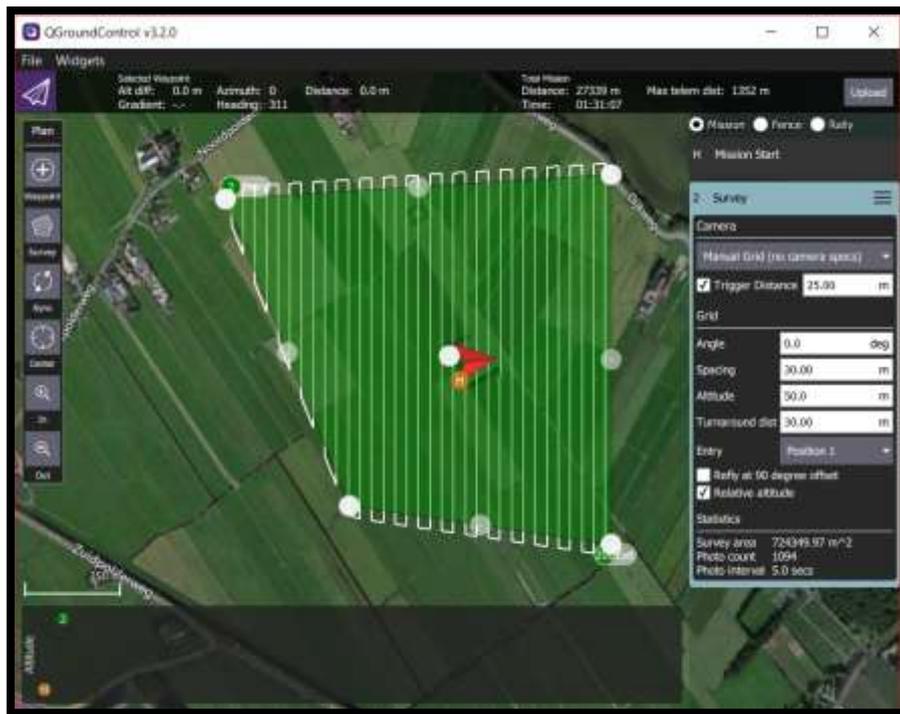
Fuente: («Sistema SONOBOT» 2019)

## **1.9. Trayectorias**

Es una ruta en donde unas distribuciones de tiempo son desarrolladas, por ejemplo, en términos de aceleraciones en cada uno de los diferentes puntos dados en una trayectoria (Siciliano et al., 2009). Así una línea trazada está vinculada con el tiempo del camino que es completado, esto depende de la rapidez con que se realice, el recorrido cambiará (Niku, 2010). Las diferentes trayectorias se escogen observando las restricciones físicas propias de las acciones, así como determinados razonamientos de calidad en la misma; tales como son la precisión y suavidad (Barrientos et al., 2007).

### ***1.9.1. Trayectorias punto a punto***

Es una metodología que consiste en establecer una serie de puntos en un área de trabajo dentro del dispositivo, donde el efector final, en este caso el (ASV), debe atravesar libremente en la trayectoria, seguida desde su configuración inicial. (Figura 6-1) (Kelly, Santibañez y Loría, 2005).



**Figura 6-1: Trayectorias punto a punto**  
 Realizado por: Carlos Estévez, José Rodas, 2019

## 1.10. MOTORES

### 1.10.1. Servomotores

Un servomotor es conocido como un motor eléctrico, que posee la posibilidad de alterar tanto la velocidad, como en la posición del eje giratorio. Sus características principales son:

- El par: que es la fuerza emitida en su eje. Éste también se denomina torque. Se suele expresar en Kg/cm, por ejemplo, 3Kg/cm.
- Velocidad: denominada velocidad angular o de rotación. (Tecnología, s.f.)

### 1.10.2. Brushless

Es un tipo de motor sin brochas. Lleva siempre agrupado un regulador de tensión o un variador, estribando del tipo de motor. (Quadrino, s.f.)

Según (Quadrino, s.f.) existen ciertas ventajas:

- Mayor eficacia (menor pérdida debido al calor).
- Mayor beneficio (duración alto de las baterías para una idéntica potencia).
- Menor peso para una idéntica potencia.
- Requieren un menor mantenimiento, debido a no poseer escobillas.
- Relación velocidad/par motor, es una cuasi-constante.
- Mayor fuerza para un tamaño idéntico.
- Excelente disipación de calor.
- Rango de rapidez elevado, debido a no poseer limitación mecánica.
- Menor ruido electrónico.

Por otro lado, ubica ciertas desventajas:

- Mayor costo de construcción.
- El control es por medio de un circuito costoso y complejo.
- Requiere un controlador electrónico (ESC's).

### ***1.10.3. Paso a paso***

Se conoce como paso a paso a un dispositivo electromecánico que convierte los pulsos eléctricos en desplazamientos angulares; es decir, es un dispositivo capaz de girar cierta suma de grados dependiendo de sus entradas de control. Los motores paso a paso son ideales para la cimentación de mecanismos, los cuales requieren movimientos exactos.

Su característica principal es el hecho de poder moverlos un paso a la vez por cada pulso aplicado. Este paso, en su mayoría, puede variar desde 90° hasta imperceptibles movimientos de 1.8°; es por ello, que se ha seleccionado comúnmente para moverse a deseo del usuario según la serie indicada por medio de un microcontrolador (EcuRed, 2018).

### 1.11. Baterías

Son elementos que almacenan energía por la reacción química en su interior generando una corriente eléctrica. Estas se dividen en dos categorías desechables y recargables. (Grafeno, 2016)

**Tabla 7-1:** Características de las baterías recargables

<b>Tipo</b>	<b>Pb-ácido</b>	<b>Ni-Cd</b>	<b>Ni-MH</b>	<b>Li-Ion</b>	<b>Li-Po</b>
<b>Aplicación</b>	Uso automotriz, industrial y doméstico	Juguetes, lámparas, equipo electrónico portátil	Productos electrónicos portátiles, vehículos de propulsión eléctrica, robots	Telefonía celular, computadoras, cámaras	Sistemas electrónicos de radio control
<b>Energía (Wh/kg)</b>	30 – 40	48 – 80	60 – 120	110 – 300	100 – 130
<b>Ciclos de recarga</b>	1000	500	1000	4000	5000
<b>Tiempo de carga (horas)</b>	8 – 16	10 – 14	2 – 4	2 – 4	1 – 1.5
<b>Tensión por celda (V)</b>	2 V	1.25 V	1.25 V	3.16 V	3.75 V
<b>Eficiencia energética (%)</b>	82.5	72.5	70.0	90.0	90.0

Realizado por: Carlos Estévez, José Rodas, 2019.

### 1.12. GPS

Los sistemas de navegación, normalmente, solucionan un problema muy antiguo en la historia de la humanidad. Este hace referencia a la necesidad de conocer sobre el punto de vista en la superficie terrestre. Sin esa capacidad, los movimientos dentro de la tierra deben apoyarse en

puntos de referencia distinguidos, y los movimientos marítimos deben circunscribirse a una franja de mar que sea visible.

En otras palabras, según el departamento de Guajira (2007), el GPS se trata de un sistema que permite computarizar las coordenadas de cualquier punto de la superficie de la tierra; a partir de la acogida de señales pronunciadas desde una constelación de satélites en órbita. En fin, su principal función es permitir al usuario conocer su ubicación en cualquier parte del mundo (GUAJIRA DEPARTAMENTO DE LA, 2007)

### **1.13. Radio transmisor**

En un sistema de comunicación, donde se plantea a la modulación como un paso importante, y lo define como el proceso de transmitir un mensaje (señal de banda base con baja frecuencia) a través de la señal del transmisor, dirigido al receptor sin cambiar sus peculiaridades, todo esto mediante el uso de una señal de alta frecuencia, que varía acorde a los valores instantáneos de la onda de baja frecuencia, permaneciendo su repetición y fase.(Electronics, 2015)

Por otro lado, un radiotransmisor de RC (TX) es un dispositivo que aprueba al piloto inspeccionar sobre el aparato, mediante ondas de radio. La señal obtenida es recibida por un receptor de radio (RX) ubicado en el aparato que está acoplado al interventor de vuelo FC.(Prometc, 2014)

#### **1.13.1. Telemetría**

La telemetría es una función muy útil que permite al RX enviar datos de vuelta al TX, como el RSSI, el voltaje de la batería o el amperaje utilizado, etc.

El RSSI (Radio Signal Strenght Indicator) es el indicador de la fuerza de la señal de radio.

### 1.13.2. Comparativas emisoras

**Tabla 8-1: Emisoras**

Nombre	Canales	Precio
<b>Flysky FS-i6s</b>	6 (expandible a 10)	56
<b>Spektrum DXe</b>	6	60
<b>Turnigy Evolution</b>	8	68
<b>Turnigy 9X</b>	8	73

Fuente:(Prometc, 2014)

Realizado por: Carlos Estévez, José Rodas, (2019).

## 1.14. Estructura

### 1.14.1. Fibra de Vidrio

El concepto fibra de vidrio hace referencia a una especie de entelado realizado a partir de diminutos hilos de vidrio entrelazados entre sí generando una malla o trama. El origen del término proviene del idioma inglés “fiber glass” y ha sido adoptado de forma casi textual en el idioma español.

Los hilos de vidrio se obtienen mediante el paso, en forma industrial, de vidrio líquido a través de una pieza resistente con pequeños orificios conocido como “espinerette”. Tras este proceso, se deja enfriar hasta adquirir una flexibilidad suficiente para construir una malla o tela.

A grandes rasgos, la fibra de vidrio es la resultante de mezclar la malla de vidrio con una resina epoxi la cual inicialmente es líquida para luego solidificar y dar la forma final. (Anmarkrud)

Las propiedades de la fibra de vidrio son:

- Elasticidad.
- Solidez.
- Resistencia, tanto a agentes químicos como al agua salada.

### 1.14.2. *Fibra de Carbono*

La fibra de carbono es un tejido muy resistente mecánicamente, duradero, flexible.

Se comercializa en forma de telas o fieltros. Sus principales características son su resistencia, su aspecto elegante y su peso liviano.

Este polímero es obtenido a partir de otro polímero llamado poliacrilonitrilo, el cual consiste en hebras muy finas de carbono (tan delgadas como el cabello humano) trenzadas, las cuales se tuercen y se agrupan continuamente para la formación de un hilo de varias hebras, se coloca sobre un molde y encima se le vierte una resina o plástico para pegar estos hilos tejidos y darles forma a sus diversas aplicaciones. (Fibra de Carbono, s.f.)

### 1.15. **Proceso actual utilizado para medición de variables del agua en camaroneras**

Actualmente la mayoría de las piscinas camaroneras efectúan el control periódico de las piscinas una vez por semana contratando un biólogo, que realiza la toma de muestras para realizar un análisis de laboratorio, es decir, dichas muestras son transportadas a un laboratorio (Figura 7-1) mediante una cadena de custodia para no cambiar o alterar las características fisicoquímicas de la misma. Donde las variables del agua más afectadas por transporte son la temperatura, pH y oxígeno disuelto, mientras que la salinidad no se ve perturbada por el transporte o tiempo hasta el análisis en laboratorio, es por ello, que la construcción del ASV es esencial, considerando que el dispositivo realiza la medición insitu.



**Figura 7-1: Toma de muestras y análisis.**

Realizado por: Carlos Estévez, José Rodas, 2019

### **1.16. Trabajos semejantes realizados en el Ecuador**

Estudiantes de la UTMACH han realizado proyectos sobre el control de temperatura mediante Arduino, sensor LM35 y una base de datos en SQL, en los cuales se efectúa la supervisión de un punto de toda la piscina acuícola en camaroneras de la provincia de El Oro, el sistema permite obtener datos sobre el control continuo de la temperatura del agua; confirmando que esta sea la conveniente para el desarrollo de la larva (camarón), evitando transmisión de enfermedades con cuidado óptimo sobre los niveles de producción (Novillo Vicuña y Vaca Simbala, 2015).

Dicha implementación no es la más factible, ya que la muestra tomada por el sensor es de un solo punto fijo en el área de la piscina, por lo cual el presente proyecto tiene como finalidad ampliar el rango de medición para abarcar mayor área supervisada. Es por ello, que la construcción de un ASV permite el proceso de obtención de datos, el cual proporcionará beneficios a la industria camaronera al poseer un mayor control de las variables del crecimiento del camarón.

## CAPÍTULO II

### 2. MARCO METODOLÓGICO

En este apartado se especifica el diseño de *hardware* y *software* del Robot acuático de superficie para supervisión de camaroneras RASSC, se define cada uno de los módulos que lo conforman, sus características técnicas y los esquemas de conexión con los dispositivos a utilizar.

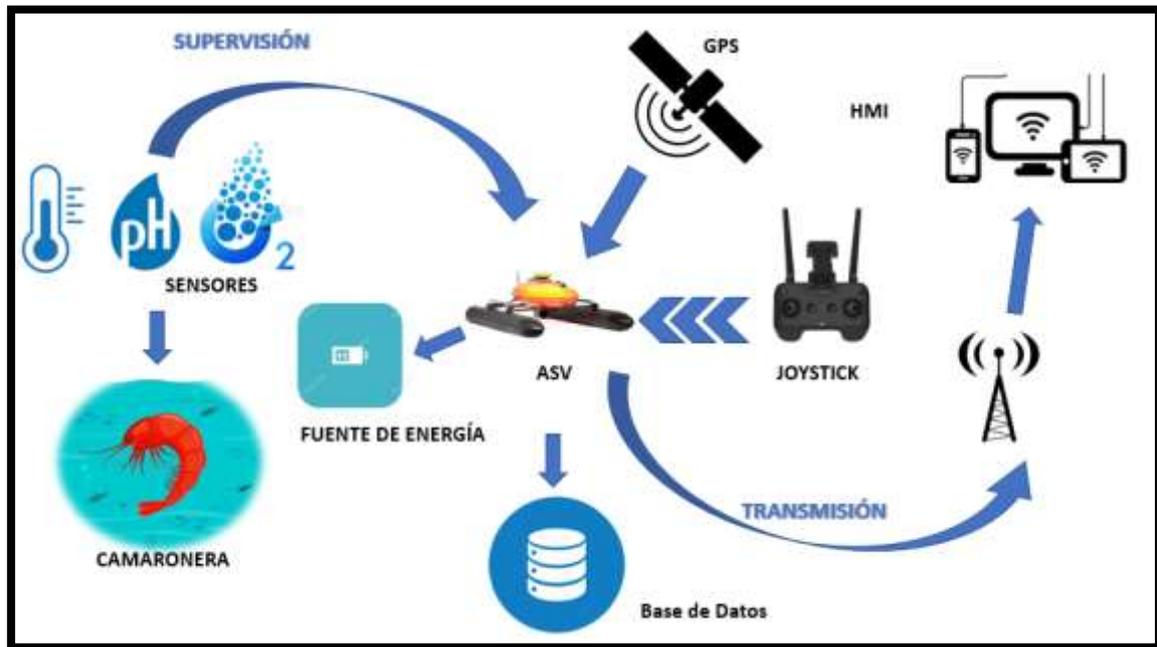
#### 2.1. Requerimientos del sistema

Basados en los estudios realizados en el capítulo anterior se plantean los requerimientos *hardware* para el dispositivo RASSC.

- Construir un prototipo de robot de superficie capaz de desplazarse en las piscinas de una hectárea.
- Seleccionar los sensores de PH, oxígeno disuelto y temperatura, conforme a las tablas de rangos de dichas variables para la especie de camarón blanco.
- Proveer la información censada dentro de la piscina camaronera.
- Proveer de autonomía energética al robot.
- El RASSC se desplazará mediante instrucciones a través de un mando de control y mediante un planificador de trayectorias punto a punto.
- El RASSC contara con un sistema de posicionamiento GPS.
- El rango de alcance de transmisión será máximo de 100m sin obstáculos.
- Visualizar en una HMI los datos obtenidos de los sensores.

## 2.2. Concepción universal del sistema del prototipo ASV

Se aprecia en la figura 1-2 el diseño *hardware* del dispositivo RASSC para la supervisión de las variables del agua, los elementos que forman parte de cada etapa: supervisión, almacenamiento, y desplazamiento, cumpliendo con los requerimientos del sistema.



**Figura 1-2. Concepción universal del sistema**

Realizado por: Carlos Estévez, José Rodas. 2018.

### Supervisión

La etapa de supervisión constituye en la adquisición de las variables de temperatura, pH y oxígeno disuelto a través de diferentes sensores sumergidos en las piscinas camaroneras, los datos obtenidos los procesará la tarjeta Arduino del RASSC donde se realizará las respectivas conversiones analógicas.

### Almacenamiento

En la segunda etapa, transmisión de datos, el módulo de comunicación la tarjeta Rasberry Pi 3 del RASSC trasmite hacia una pantalla HMI y a su vez a la nube que muestra los valores medidos periódicamente mientras se almacenan en la base de datos ThingSpeak.

## Desplazamiento

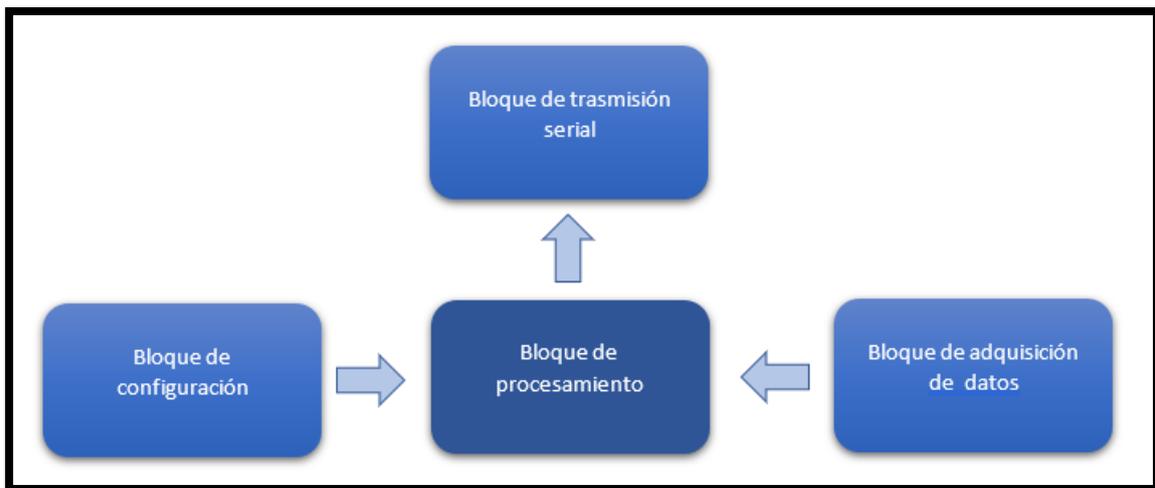
La tercera etapa, recepción del mando, el joystick envía a través de radio frecuencias las señales con los comandos para que el RASSC se desplace en la superficie.

El RASSC está integrado por una tarjeta de desarrollo PIXHAWK a la cual se conecta el módulo de motores, comunicación y GPS, alimentada por un banco de baterías tipo Li-Po.

### 2.3. Diseño de la arquitectura del RASSC

En esta sección se presenta el diseño de las 3 etapas que conforman al RASSC mediante diagramas de bloques y se realiza una breve explicación de la funcionalidad de cada una de ellas.

#### 2.3.1. Etapa de supervisión



**Figura 2-2: Diagrama de bloques etapa de supervisión**

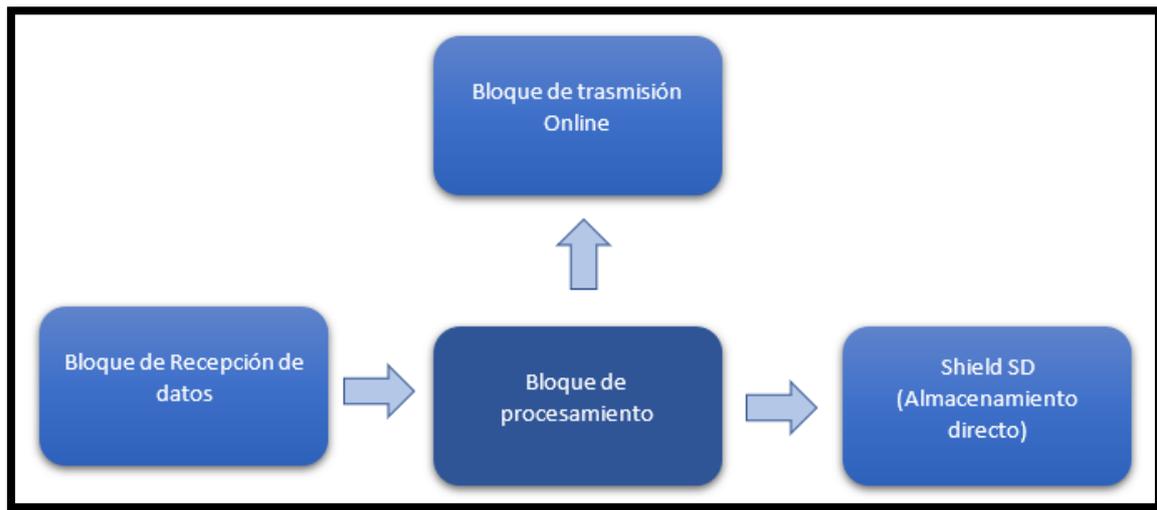
Realizado por: Carlos Estévez, José Rodas. 2019.

Como se aprecia en la figura 2-2, la etapa de supervisión cuenta con los bloques de adquisición, configuración y procesamiento de los sensores y a su vez el bloque de transmisión de datos.

Los datos obtenidos por los sensores de oxígeno disuelto, pH, temperatura son procesados mediante la tarjeta de desarrollo para convertirlos en valores numéricos, y así posteriormente ser transmitidos mediante el puerto serial a la etapa de Trasmisión e almacenamiento.

Este bloque está conformado de un Arduino Mega el cual dispone de entradas analógicas y digitales y 3 puertos seriales, el sensor de pH utiliza una entrada analógica de la misma manera el sensor de temperatura, el sensor de oxígeno disuelto utiliza el protocolo I2C el cual es conectado en un terminal serial TX y RX.

### 2.3.2. Etapa de almacenamiento



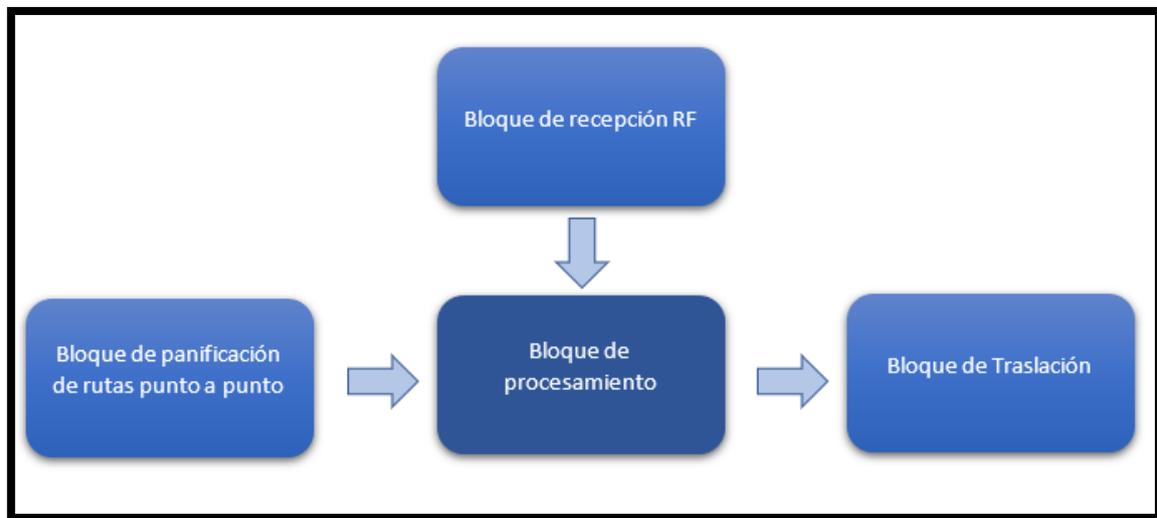
**Figura 3-2: Etapa de almacenamiento**

**Realizado por:** Carlos Estévez, José Rodas. 2019.

Como se aprecia en la figura 3-2, la etapa de almacenamiento cuenta con los bloques de recepción, almacenamiento directo y almacenamiento en la nube, cada uno de estos con su respectiva visualización.

Dicha etapa cuenta con un módulo de procesamiento Raspberry pi 3 B la cual posee un shield SD, para permitir almacenar la base de datos de forma directa y un módulo Wifi integrado, de tal forma que permite realizar la transmisión de los datos hacia la nube, en una plataforma IoT (ThingSpeak), donde se podrá visualizar y descargar los datos registrados en un determinado periodo de tiempo, a su vez el módulo de procesamiento realizara la función de interfaz de las trayectorias punto a punto.

### 2.3.3. Etapa de desplazamiento



**Figura 4-2: Etapa de desplazamiento**

Realizado por: Carlos Estévez, José Rodas. 2019.

Como se aprecia en la figura 4-2, la etapa de desplazamiento cuenta con los bloques de recepción, planificador de rutas y traslación.

Esta etapa cuenta con un módulo de procesamiento Pixhawk el cual recibe las señales por radio frecuencia, ya estas sean mediante el joystick o planificador de rutas, La tarjeta Pixhawk procesa dichas señales y activa los motores busheles para ejecutar la traslación al punto deseado o para realizar rutas predefinidas.

## 2.4. Elección de dispositivos electrónicos para la construcción del RASSC

En esta sección se presenta los principales elementos del RASSC, la justificación de la elección de los distintos componentes, sus características y los módulos que lo conforman, para mayor comprensión revisar datos técnicos de los dispositivos en el apartado de anexos.

### 2.4.1. Tarjetas de desarrollo

Mediante la comparativa de las distintas tarjetas de desarrollo realizada en el capítulo uno, las tarjetas a ser utilizadas son Arduino MEGA, Pixhawk y Raspberry pi 3 B.

#### 2.4.1.1. Arduino mega 2560

La tarjeta programable Arduino Mega 2560 que se aprecia en la figura 5-2 posee un microcontrolador ATmega 2560, cuenta con 54 entradas y salidas digitales, entre ellas 15 pueden ser utilizadas como salidas PWM, 16 entradas analógicas y 4 puertos serie y 4 puertos positivos y 5 GND lo cual es esenciales para el RASSC, considerando el tipo de conexión de los sensores de: oxígeno, pH y temperatura, adaptándose a los requerimientos.



**Figura 5-2: Arduino Mega 2560**  
Realizado por: Carlos Estévez, José Rodas. 2019.

#### 2.4.1.2. Raspberry pi 3 B

Para que el dispositivo RASSC sea un equipo embebido se realiza la selección de una Raspberry pi 3 B, dado que posee una gran compatibilidad para realizar la instalación de diversas herramientas como es el Qground, el cual permite interactuar con la configuración de la etapa de desplazamiento donde se realiza la calibración de los periféricos como GPS, Magnetómetro y acelerómetro, realiza la función de planificador de trayectorias por puntos para realizar rutas

automáticas, también posee integrado un Wifi el cual permite conectar al Smartphone con este dispositivo.



**Figura 6-2: Raspberry Pi 3**

**Realizado por:** Carlos Estévez, José Rodas. 2019.

#### 2.4.1.3. Pixhawk

En el capítulo anterior se realizó la comparativa de las diferentes tarjetas de desarrollo y para la aplicación a utilizar la tarjeta PIXHAWK es la que mejor se adapta a vehículos teledirigidos dado que cuenta con puertos integrados para control de motores, GPS, acelerómetro y radio frecuencia para recibir las señales de mando del joystick.

**Tabla 1-2:** Características de Pixhawk

<b>Descripción</b>	<b>Datos</b>
<b>Procesador</b>	<ul style="list-style-type: none"><li>• 32-bit ARM Cortex m4 core con FPU</li><li>• 168 MHz/256 KB RAM/ 2 MB Flash</li><li>• 32-bit failsafe coprocesador</li></ul>
<b>Sensores</b>	<ul style="list-style-type: none"><li>• MPU6000 como acelerómetro y giroscopio principal</li><li>• ST micro 16-bit como giroscopio secundario</li><li>• ST micro 14-bit acelerómetro/brújula</li></ul>
<b>Alimentación</b>	<ul style="list-style-type: none"><li>• Opción 1 → Utilizar la energía proveniente de los ESC</li><li>• Opción 2 → Utilizar cable de alimentación directo a placa</li><li>• Opción 3 → combinación de las dos</li></ul>
<b>Puertos</b>	<ul style="list-style-type: none"><li>• 5x UART puertos serie</li><li>• Spektrum DSM/DSM 2/DSM-X Conexión GPS</li><li>• Futaba S.BUS salida</li><li>• PPM sum señal</li><li>• RSSI (PWM) entrada</li><li>• I2C, SPI, 2x CAN, USB</li><li>• 3.3V y 6.6V ADC entradas</li></ul>
<b>Dimensiones</b>	<ul style="list-style-type: none"><li>• Anchura 50mm</li><li>• Altura 15.5mm</li><li>• Longitud 81.5 mm</li></ul>

Realizado por: José Rodas y Carlos Estévez. 2018

## 2.4.2. Sensores de temperatura, oxígeno disuelto y pH

### 2.4.2.1. Temperatura

La condición de temperatura para la especie de camarón blanco está detallada de la siguiente manera.

**Tabla 2-2:** Condiciones de temperatura

<b>Temperatura letal más baja</b>	<b>Temperatura preferida</b>	<b>Temperatura letal más alta</b>
14°C	>20°C	40°C

Realizado por: José Rodas y Carlos Estévez. 2018

El rango del sensor de temperatura deberá cubrir desde 10°C -50°C.

La precisión entre  $\pm 1^\circ\text{C}$ .

#### 2.4.2.2. Sensores de temperatura

**Tabla 3-2:** Comparativa de sensores de temperatura

<b>Características</b>	<b>LM35</b>	<b>DHT11</b>	<b>DS18B20</b>
<b>Voltaje de Operación</b>	4-30 V	3 - 5.5 V	3-5.5 V
<b>Rango De Temperatura</b>	-55 °C a 150 °C	0 °C a 50 °C	-10 °C a 85 °C
<b>Precisión</b>	$\pm 0.6^\circ\text{C}$	$\pm 2^\circ\text{C}$	$\pm 0.5^\circ\text{C}$
<b>Tiempo De Respuesta</b>	2s	1s	2s
<b>Tipo De Salida</b>	Análoga	Digital	Digital
<b>Precio</b>	\$3	\$6	\$10
<b>Disponible en sonda a prueba de agua</b>	NO	NO	SI

Realizado por: Carlos Estévez, José Rodas. 2019

Dado a su precisión de  $\pm 0.5^\circ\text{C}$  y su disponibilidad a ser sumergido y se encuentra dentro de los rangos de medición, se realiza la selección del DS18B20

#### 2.4.2.3. Oxígeno disuelto (DO)

El ciclo dinámico de oxígeno de los estanques fluctúa durante el día debido a la fotosíntesis y respiración del fitoplancton. un máximo de DO ocurrirá pasada la tarde debido al crecimiento de  $\text{O}_2$  durante el día a través de la fotosíntesis. Como el fitoplancton (algas microscópicas) usualmente consume la mayoría del  $\text{O}_2$  y debido a que la fotosíntesis no ocurre durante la noche, los niveles de DO declinan. Por lo tanto, los niveles de DO pueden ser afectados por cambios en el ambiente; un día nublado disminuirá la introducción del  $\text{O}_2$  de fotosíntesis al DO. Consecuentemente, altas temperaturas inusuales disminuirán la solubilidad del  $\text{O}_2$  en el agua, y por lo tanto un menor DO. Cuando un estanque está en “equilibrio” el DO no cambiará drásticamente. En la siguiente tabla se resume los efectos de las concentraciones de oxígeno sobre los camarones.

**Tabla 4-2:** Concentración de oxígeno y efectos sobre el camarón

Concentración de oxígeno disuelto	Efecto
Menor de 1 o 2 mg/L	Letal si la exposición dura más que unas horas.
2-3 mg/L	Crecimiento será lento si la baja de oxígeno disuelto se prolonga.
4-5 mg/L – saturación	Mejor condición para crecimiento adecuado
>5 mg/L – Super-saturación	Puede ser dañino si las condiciones existen por todo el estanque. Generalmente, no hay problema.

Realizado por: José Rodas y Carlos Estévez. 2018

Fuente:(«Balnova | Oxígeno en estanques de camarón» 2014)

#### 2.4.2.4. Sensores de DO

**Tabla 5-2:** Características Sensor de oxígeno

Características	EZO-DO TM	Puntaje	ODO W24-04	Puntaje
Voltaje de operación	3.3-5v	5	6.3 v DC	1
Corriente de trabajo	15mA	5	10-20 mA	5
Temperatura	-40 a 85 °C	5	-20 a 70° C	5
Rango de operación	0.01-100 + mg/L	5	0 a 50 mg/ L	4
Precisión	± 0.05mg/L	5	± 0.2 mg/L	4
Tiempo de respuesta	1 lectura por segundo	5	90% en 25 segundos	1
Protocolo de comunicación	I2C & UART	5	RS485	1
Acondicionar señal	Si	5	No	1
<b>Total</b>		<b>5</b>		<b>2.75</b>

Fuente: (Atlas Scientific LLC, 2019)

Realizado por: Carlos Estévez, José Rodas. 2019.

Como se aprecia en la tabla 5-2 el sensor de oxígeno atlas es el ganador, principalmente por su tarjeta de acondicionamiento, su rápida respuesta y su protocolo de comunicación, en la figura 7-2 se aprecia el sensor ganador.



**Figura 7-2: Sensor Atlas Scientific EZO-DO**  
Realizado por: Carlos Estévez, José Rodas. 2019

#### 2.4.2.5. PH

Es importante mantener un pH estable a un rango seguro porque esto afecta el metabolismo y otros procesos fisiológicos de los organismos de cultivo. Puede crear estrés, aumentar la susceptibilidad a enfermedades, disminuir los niveles de producción y causar un pobre crecimiento y aún muerte.

En la literatura científica se menciona que el rango óptimo de pH para el cultivo de camarón es de 7.5 en la mañana y 8.5 en la tarde.

**Tabla 6-2:** Efectos del PH al camarón

<b>Efecto</b>	<b>pH</b>
Punto de acidez letal	4
No reproducción	4-5
Crecimiento lento	5-6
Mejor crecimiento	6-9
Crecimiento lento	9-11
Punto letal de alcalinidad	11

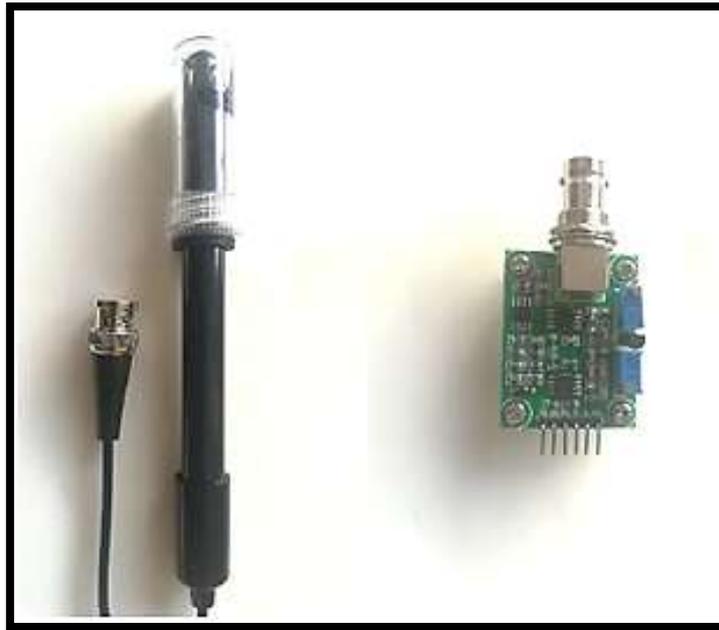
**Fuente:**(«Balnova | Oxígeno en estanques de camarón» 2014)

**Realizado por:** Carlos Estévez, José Rodas. 2019

#### 2.4.2.6. Selección del sensor

Se realiza la selección de un E-201 el cual se aprecia en la figura dado que se encuentra dentro de los requerimientos necesarios de la piscina contando con las siguientes características:

- Rango 0-12
- Precisión  $\pm 0.2$
- Tiempo de respuesta  $< 5s$
- Disponibilidad inmediata
- Señal Análoga
- Permite acondicionamiento de señal
- Bajo costo



**Figura 8-2: Sensor pH E-201**  
 Realizado por: Carlos Estévez, José Rodas. 2019

### 2.4.3. Tarjeta de transmisión de datos

**Tabla 7-2:** Comparativa de tarjetas de comunicación

	<b>Zigbee</b>	<b>Wi-Fi</b>	<b>Bluetooth</b>
RANGO	10-100 metros (+routers)	50-100 metros (+routers)	10-100 metros
BANDA ANCHA	250kbps	1Mbps	Hasta 54Mbps
TOPOLOGÍA DE RED	Ad-hoc, punto a punto estrella o malla	Punto a hub	Ad-hoc, redes muy pequeñas
COMPLEJIDAD (IMPACTO DE DISPOSITIVO Y APLICACIÓN)	Bajo	Alto	Alto
CONSUMO DE POTENCIA (OPCIONES DE BATERÍA Y VIDA)	Muy bajo (utilizar potencia es objetivo del diseño)	Alto	Medio
SEGURIDAD	128 AES, más seguridad en la capa		Cifrado de 64 y 128 bits

	de aplicación		
OTRA INFORMACIÓN	Los dispositivos pueden unirse a una red existente en menos de 30m	La conexión del dispositivo requiere de 3 a 5 segundos	La conexión del dispositivo requiere un máximo de 10 segundos
APLICACIONES TÍPICAS	Control industrial y monitorización, redes de sensores, automatización de edificios, control y automatización del hogar, juguetes, juegos	Conectividad LAN inalámbrica, acceso de banda ancha internet	La conectividad inalámbrica entre dispositivos como teléfonos, PDA, computadoras portátiles, auriculares

Realizado por: Carlos Estévez, José Rodas. 2019

Se selecciona la comunicación Wifi ya que cumple con los requerimientos de alcance, conexión entre dispositivos y facilidad para adaptar repetidores y extender el alcance.

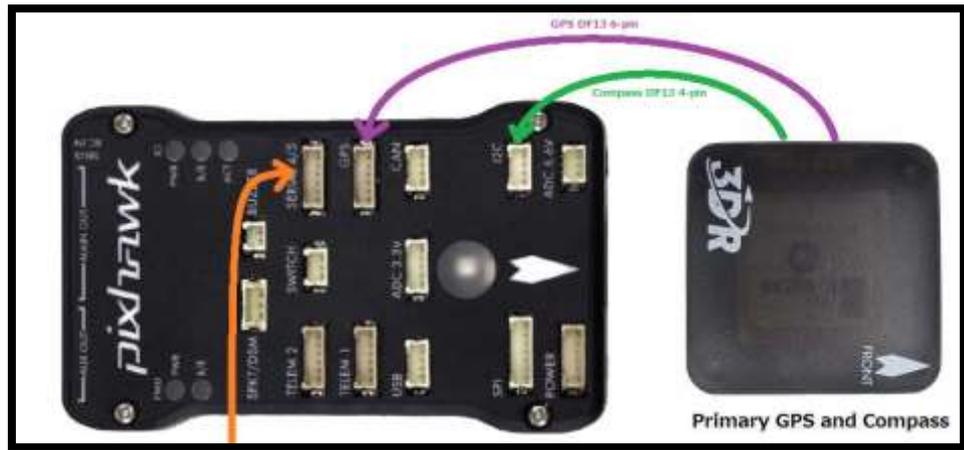
#### 2.4.4. Motor Brushless y módulo ESC



**Figura 9-2: Motor brushless y ESC de 45A.**

Realizado por: Carlos Estévez, José Rodas. 2019.





**Figura 11-2. Esquema de conexión GPS**  
 Realizado por: Carlos Estévez, José Rodas. 2019.

#### 2.4.6. Mavlink de pixhawk

Posee un alcance de 100 m sin dispositivos de antena direccional, así adaptándose de la mejor manera al sistema (Pixhawk) y cumpliendo los requerimientos de rango de alcance.



**Figura 12-2. Telemetría**  
 Realizado por: Carlos Estévez, José Rodas. 2019.

#### 2.4.7. Receptor para joystick

Llamados también PPM encoder, son los más utilizados en vehículos tele operados para la recepción de las señales de mando, y al poseer gran compatibilidad con la tarjeta programable (Pixhawk) al no necesitar ninguna modificación en el receptor R/C garantiza la precisión y estabilidad del sistema.



**Figura 13-2. Módulo de telemetría**  
Realizado por: Carlos Estévez, José Rodas. 2019.

#### 2.4.8. Consumo energético del sistema

En esta sección se analiza los consumos en corriente de todos los dispositivos implementados en el sistema RASSC, con lo que se determina la fuente de energía mínima (Batería) a partir de la descarga máxima que estos generan, como se muestra en la tabla 8-2.

**Tabla 8-2:** Consumo energético de cada elemento

Dispositivo	Voltaje (V)	Corriente (Ah)	Potencia (Wh)
Sensor EZO-OD	3.5	0.0135	0.04725
Sensor E-201	5	0.01	0.05
Sensor DS18B20	5	0.005	0.25
GPS	5	0.08	0.4
Pixhawk	5	0.250	1.25
Raspberry	5	0.800	4
Arduino Mega 2560	5	0.2	1
Motores Brushlles (x2)	11.1	5	55.5
<b>Total</b>		6.6585	62.49725

Realizado por: Carlos Estévez, José Rodas. 2019.

#### 2.4.8.1. Selección de batería

En el capítulo anterior, el estudio para la autonomía del robot se comparó diferentes tipos de batería y las baterías LiPo por su tasa de descarga alta y de gran capacidad en tamaño reducido y ligero son las óptimas para garantizar la autonomía del ASV.

Se realizó una división para la alimentación de los equipos de la siguiente manera: motores y electrónica.

La electrónica posee un consumo total máximo de 1.35 Ah por lo que para un funcionamiento óptimo se utiliza una batería de 1.4 sobredimensionando a su inmediato superior en venta.

El motor Brushlles posee un consumo de corriente máximo de 2.5 Ah, por ello se sobredimensiona a la más próxima en el mercado, siendo esta una batería LiPo de 3 A, cabe recalcar que será utilizada una batería para cada motor, para mejorar la distribución de peso en cada flotador y facilitar sus conexiones.

Se selecciona una batería con una tasa de descarga de 50C dado al pico máximo de corriente requerido por los motores el cual es de 45 A.



**Figura 14-2. Batería Li-Po**  
Realizado por: Carlos Estévez, José Rodas. 2019

## 2.5. Diseño estructura RASSC

El diseño del sistema se realizará mediante el software SolidWorks CAD que permite bosquejar las partes que componen al ASV como lo son: flotadores y casco principal.

### 2.5.1. Referencia para el diseño del ASV

Para el diseño estructural del robot se ha tomado de referencia el ASV GLOBAL C-CAT3, el mismo que cuenta con dos flotadores y un casco central.



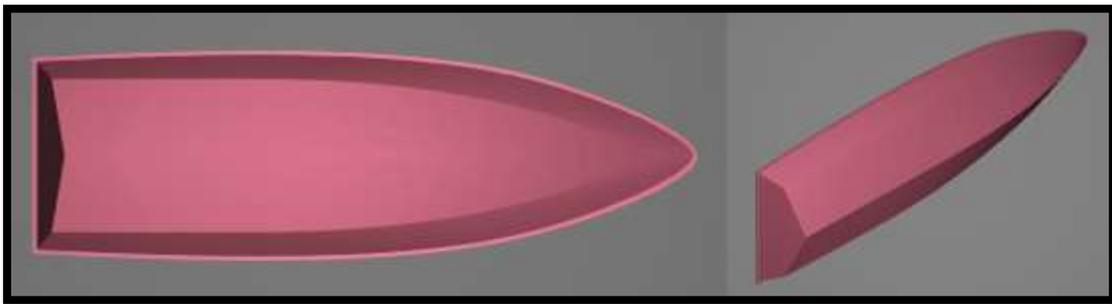
**Figura 15-2. ASV C-CAT3**  
Fuente: (NEWMAN, 2018)

Este tipo de diseño le permite a robot un desplazamiento con dirección diferencial con dos propulsores, además los flotadores garantizan un buen nivel de flotabilidad y estabilidad tanto para aguas calmadas o con pequeñas corrientes.

### 2.5.2. *Casco principal con tapa*

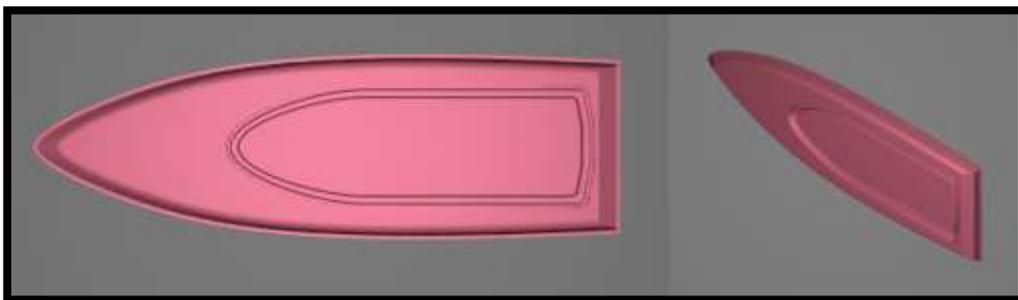
El casco corresponde a la estructura exterior que cubre e impermeabiliza el barco, este se compone de: copa, proa, babor, estribor que corresponden a la parte trasera, punta, lado izquierdo y lado derecho respectivamente.

En el diseño las dimensiones son: 789 mm de largo, 250 mm de ancho y 85 mm de altura.



**Figura 16-2. Casco principal**  
Realizado por: Carlos Estévez, José Rodas, 2019.

Para las dimensiones de la tapa se tiene: 789 mm de largo, 250 mm de ancho y 33 mm de altura.

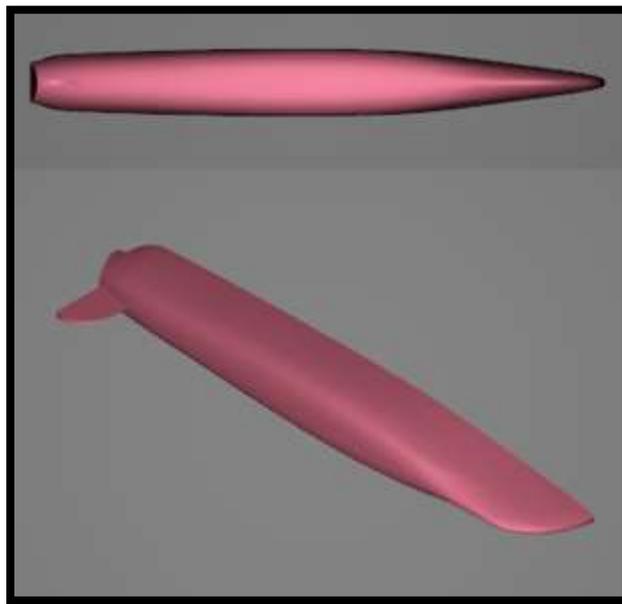


**Figura 17-2. Tapa del casco**  
Realizado por: Carlos Estévez, José Rodas, 2019.

### 2.5.3. Flotadores

Se denomina flotadores a los cuerpos que sostienen el casco principal para que no emerja, también dependiendo diseño que estos tengan permiten desplazar de forma más rápida al barco, ya que ayudan a romper la inercia del agua.

En el diseño se han determinado las siguientes dimensiones: 703 mm de largo, 83 mm de ancho y 138 mm de alto.



**Figura 18-2. Flotadores**  
Realizado por: Carlos Estévez, José Rodas, 2019.

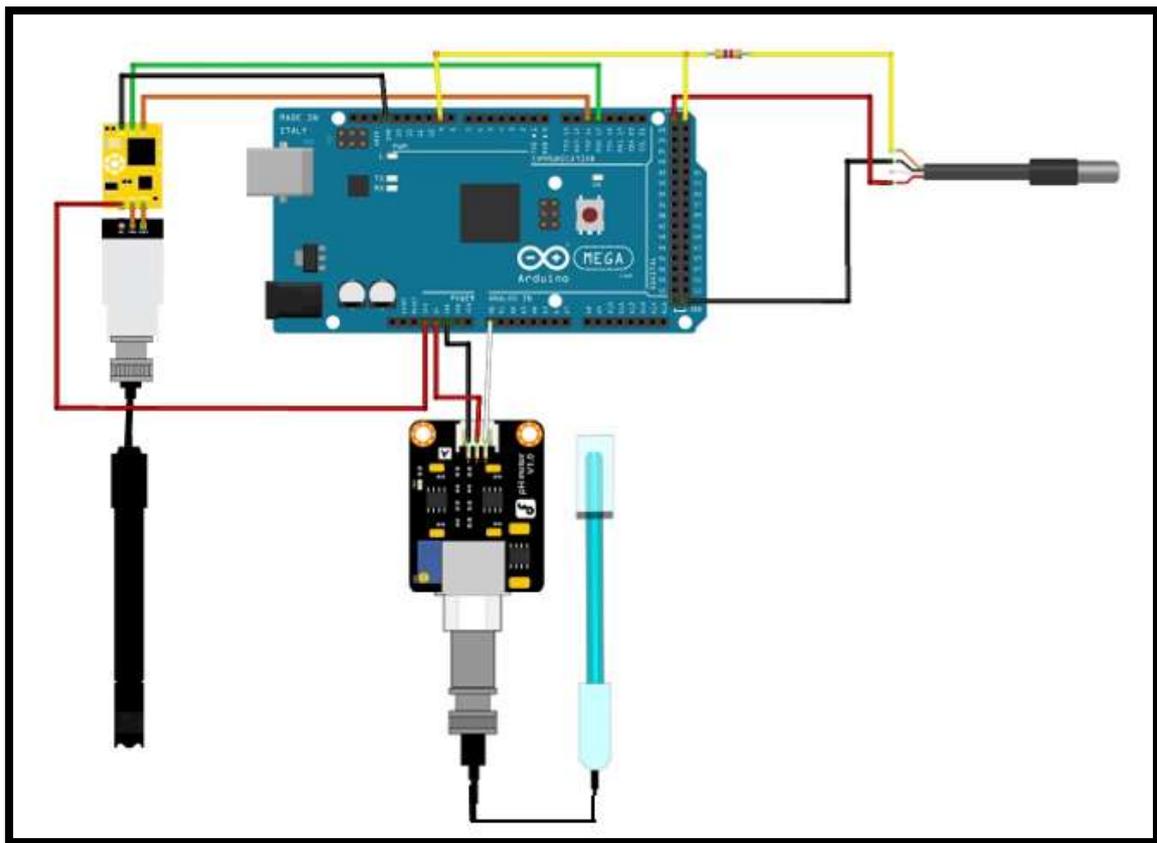
## 2.6. Esquemas de conexiones del RASSC

Mediante la presentación de todos los componentes que conforman al RASSC, se puede proseguir al detalle de las conexiones realizadas

### 2.6.1. Diagrama de conexión de la etapa de supervisión

En el módulo de supervisión se realiza las conexiones que se pueden apreciar en la figura 19-2. La tarjeta de desarrollo es el Arduino MEGA 2650 que posee conectado en sus terminales los sensores de oxígeno disuelto, pH y temperatura de la siguiente manera:

- El sensor de pH conectada su salida en el puerto análogo A0 del Arduino y una alimentación de 5V y GND.
- El sensor de temperatura posee puenteado mediante una resistencia de 4.7k ohmios los terminales señal y 5V, su salida se encuentra en el puerto 9 pwm del Arduino
- El sensor de oxígeno disuelto se encuentra conectado mediante comunicación UART al Arduino, esto quiere decir que los terminales TX y RX de la tarjeta de acondicionamiento son conectados a los terminales RX2 y TX2 respectivamente a su vez posee los puertos GND y PGND de la tarjeta de acondicionamiento conectados a GND del Arduino y su terminal VCC conectado a 5 V.
- La tarjeta Arduino es conectada al módulo de transmisión Raspberry Pi 3 mediante puerto USB. Todos los datos una vez acondicionados son transmitidas por puerto serial a la Raspberry Pi 3.

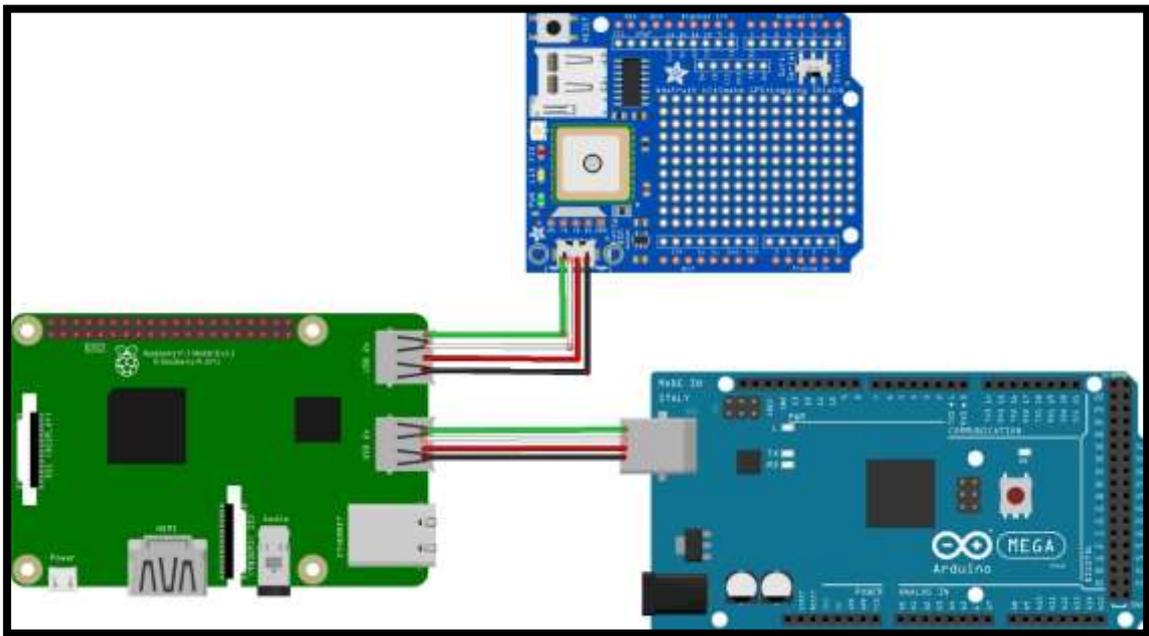


**Figura 19-2: Esquema de conexión, etapa de supervisión**  
 Realizado por: Carlos Estévez, José Rodas. 2019

### 2.6.2. Diagrama de conexión de la etapa de Transmisión

En el módulo de Trasmisión se realiza las conexiones que se pueden apreciar en la figura 20-2. La tarjeta de desarrollo es una Raspberry PI 3 la cual realiza la lectura del puerto serial del Arduino Mega. Se explicará el proceso a continuación:

- Se realiza la lectura serial del Arduino y se separa la cadena de caracteres en la tarjeta de procesamiento Rasberry Pi3.
- Se conecta el GPS mediante puerto USB y se realiza el acondicionamiento de este y a su vez se almacena la posición en variables.
- Se procede a continuación la trasmisión de los datos a la tarjeta SD y a su vez a la nube.
- Una vez subidos a la base de datos online ThingSpeak se pude tener acceso ya sea desde un ordenador mediante la página web o mediante la aplicación móvil.
- La aplicación móvil toma el último dato recibido en la plataforma.



**Figura 20-2: Esquema de conexión, etapa de transmisión**

Realizado por: Carlos Estévez, José Rodas. 2019

### ***2.6.3. Diagrama de conexión de la etapa de desplazamiento***

En el módulo de desplazamiento se realiza las conexiones que se pueden apreciar en la figura 21-2. La tarjeta de desarrollo es una Pixhawk la cual posee la función de tomar las señales de radio frecuencia para trasladar al RASSC o procesar las trayectorias punto a punto preestablecidas, a continuación se detalla brevemente las conexiones realizadas.

- Se realiza las conexiones del GPS y magnetómetro a la tarjeta para de esta manera pueda realizar las trayectorias punto a punto.
- Se conecta el pulsador a la tarjeta pixhawk como elemento de seguridad para armado y desarmado de los motores.
- El módulo de energía el cual realiza las lecturas del estado de la batería, a su vez regula y protege a la tarjeta de procesamiento
- Los controladores de motores ESC son alimentados cada uno con su propia batería para prolongar el funcionamiento de este, a su vez está conectado a la tarjeta pixhawk para en caso de fallo de la batería principal estas suministren energía a las demás etapas.



Requerimientos etapa de supervisión:

- Adquirir los datos mediante los sensores de temperatura, oxígeno disuelto y pH
- Convertir las señales analógicas de los sensores en mg/L, acidez del agua y °C respectivamente
- Enviar desde la etapa de supervisión mediante puerto serial la cadena, OD, pH, Temperatura para su almacenamiento y transmisión.

Requerimientos etapa de transmisión:

- Visualizar la información de Oxígeno disuelto, pH y temperatura, por medio de una HMI.
- Realizar una base de datos en la tarjeta raíz SD de la Raspberry pi3
- Transmitir los datos obtenidos a la nube en la plataforma ThingSpeak cada medio minuto.
- Crear una App móvil como un elemento de aviso para visualizar los datos de la nube durante el desplazamiento del RASSC.

Requerimientos etapa de desplazamiento:

- Establecer una comunicación bidireccional que permita conocer el nivel actual de energía y en caso de bajo voltaje cambiar a batería de emergencia.
- Realizar la traslación del RASSC mediante un joystick de radio frecuencia para trayectorias manuales
- Efectuar trayectorias punto a punto mediante el planificador Tower, desde un dispositivo móvil.

### **2.7.2. *Softwares utilizados para la programación del RASSC***

El encargado del buen funcionamiento de los módulos implementados son distintos *softwares* de desarrollo, los cuales poseen una programación de funciones estructuras. Se utilizaron los siguientes:

#### **2.7.2.1. *Software para la tarjeta de adquisición de datos Arduino***

Mediante esta herramienta de *software* libre, se realiza la programación de la tarjeta de adquisición de datos Arduino mega 2560 a través del IDE de Arduino, el cual simula un entorno

de programación C++. Los archivos creados mediante Arduino son llamados *sketch*, los cuales se conforman por instrucciones que trabajan en conjunto con bibliotecas (Martínez, 2013).

Consta de dos estructuras importantes: *void setup ()*, donde se ejecuta una sola vez las instrucciones planteadas instrucciones, comúnmente utilizada para realizar llamados de bibliotecas y declaraciones de variables. *void loop ()* como su nombre lo dice se encarga de hacer las repeticiones continuas de las instrucciones que posee escritas (García González, 2015)

#### 2.7.2.2. *Software Python*

Dado que este entorno de programación viene integrado en el sistema Operativo Linux de la Raspberry y cuenta con programación por consola, ofrece optimización de los recursos de memoria y de menor grado de complejidad en las líneas de código, ante otros entornos de programación como Java y C++. Además, posee librerías para la transmisión, recolección de datos y lectura de puerto serial.

#### 2.7.2.3. *Software Android estudio*

El software libre Android estudio, el cual está basado en una programación C++ y al poseer una gran compatibilidad con archivos JSON, lo propone como el *software* de programación a utilizar para el desarrollo de la Aplicación móvil.

La programación se realiza en dos entornos uno en línea de código llamada MainActivity, donde se realiza la declaración de variables, operaciones y recolección de datos, esto es desarrollado en lenguaje java. La otra, una interfaz gráfica denominada ActivityMain la cual permite realizar el diseño de la aplicación desarrollada en xml.

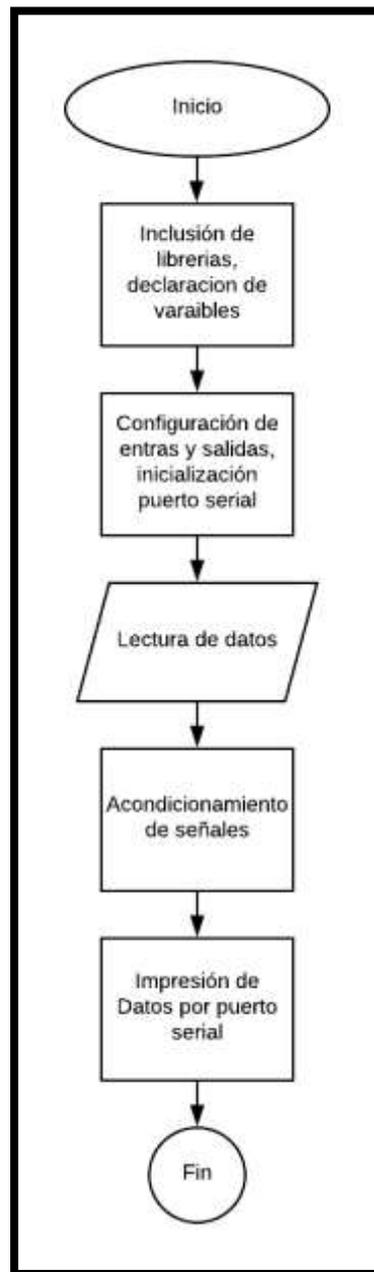
### 2.7.3. *Programación realizada para cada etapa del RASSC*

Los códigos realizados en los distintos *Softwares* se presentan en el apartado anexos. A continuación, se muestran los diagramas de flujo, funciones y bibliotecas utilizadas en cada uno de los *softwares*.

### 2.7.3.1. Diagrama de flujo del sistema de supervisión

La programación realizada se representa en el diagrama de flujo en la figura 22-2. Se realiza lo siguiente. Declaración de bibliotecas y variables:

- Se integran las bibliotecas respectivas las cuales permiten la detección y calibración del sensor.
- Se configura las entradas respectivas para los sensores de oxígeno, pH, GPS y temperatura.

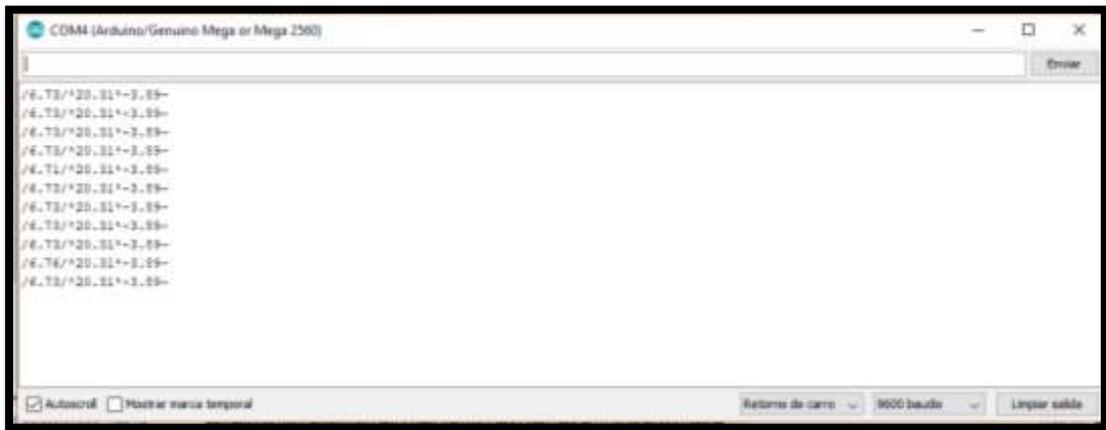


**Figura 22-2: Diagrama de flujo del sistema de supervisión**

Realizado por: Carlos Estévez, José Rodas. 2019.

Procedimiento:

- Se inicializa el puerto serial y el sensor de temperatura.
- Se realiza las lecturas de los sensores pH en el puerto análogo A0, Oxígeno en la comunicación UART Tx2 y Rx2, y el sensor de temperatura en el puerto 9, a continuación, se procede a la conversión de los distintos datos a sus respectivas unidades mg/L, acidez del agua y °C.
- Se continua con la transmisión los cuales se imprimen como una cadena de caracteres para ser transmitidos por puerto serial, como se observa en la figura 23-2.



**Figura 23-2: Cadena de caracteres enviada por puerto serial**

Realizado por: Carlos Estévez, José Rodas. 2019.

### 2.7.3.2. Bibliotecas etapa de supervisión

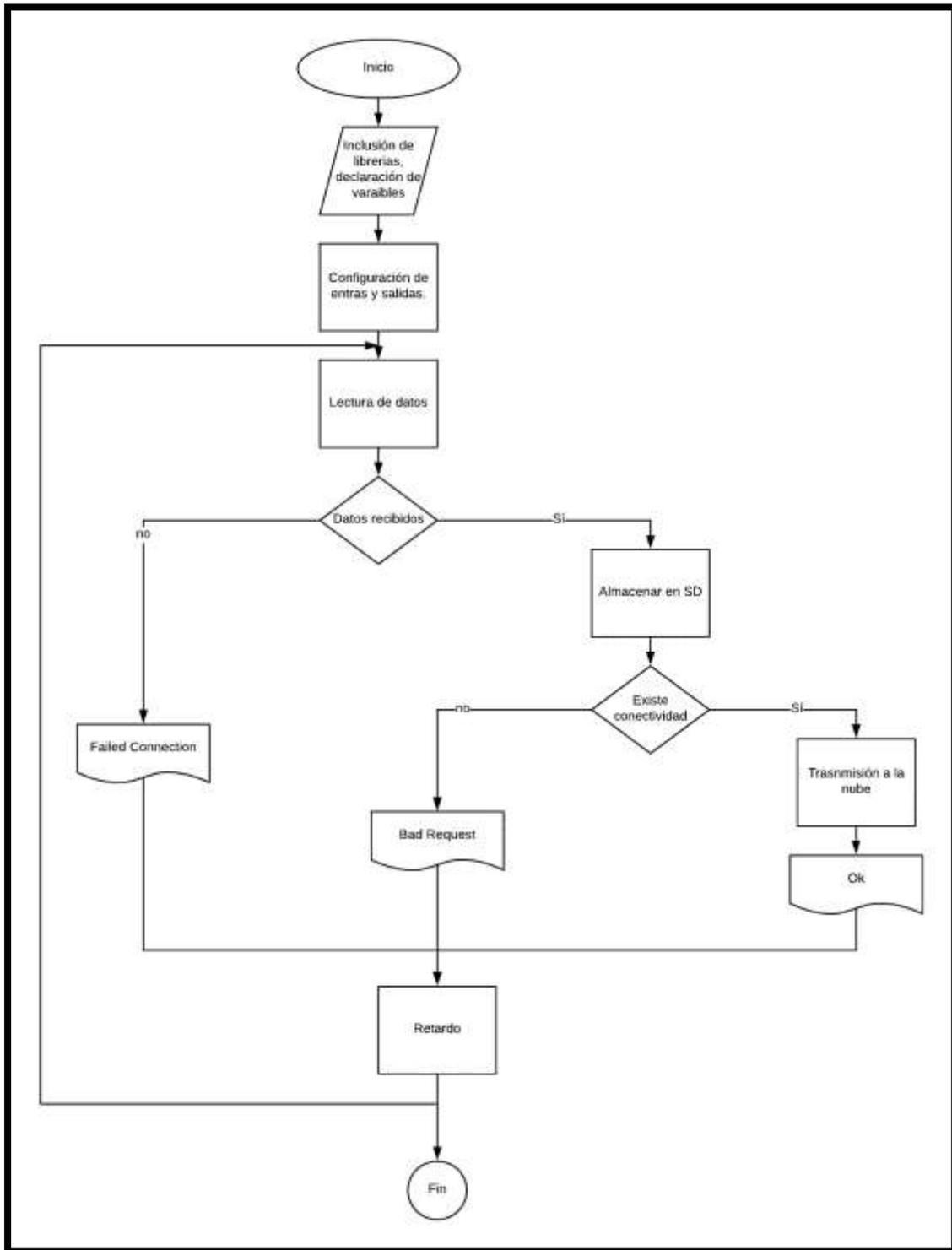
Las bibliotecas requeridas predefinidas, las cuales fueron incluidas se declaran a continuación:

- **OneWire.h:**
- Esta biblioteca permite transmitir y recibir información mediante un único cable, lo cual ayuda a detectar el sensor cuando existe más de uno y permite calibrarlo.
- **DallasTemperature.h:** permite reconocer la tarjeta integra del sensor (el número de serie) para su respectivo calibra miento

### 2.7.3.3. Diagrama de flujo de la etapa de transmisión

La programación realizada se representa en el diagrama de flujo en la figura 24-2. Se realiza lo siguiente.

- Instalación de bibliotecas en raspberry Pi 3 B, declaración de variables utilizadas en la programación.
- Configuración de puerto de entrada para realizar la lectura del puerto serial
- Inicialización de almacenamiento y transmisión de datos.



**Figura 24-2: Diagrama de flujo etapa de trasmisión**

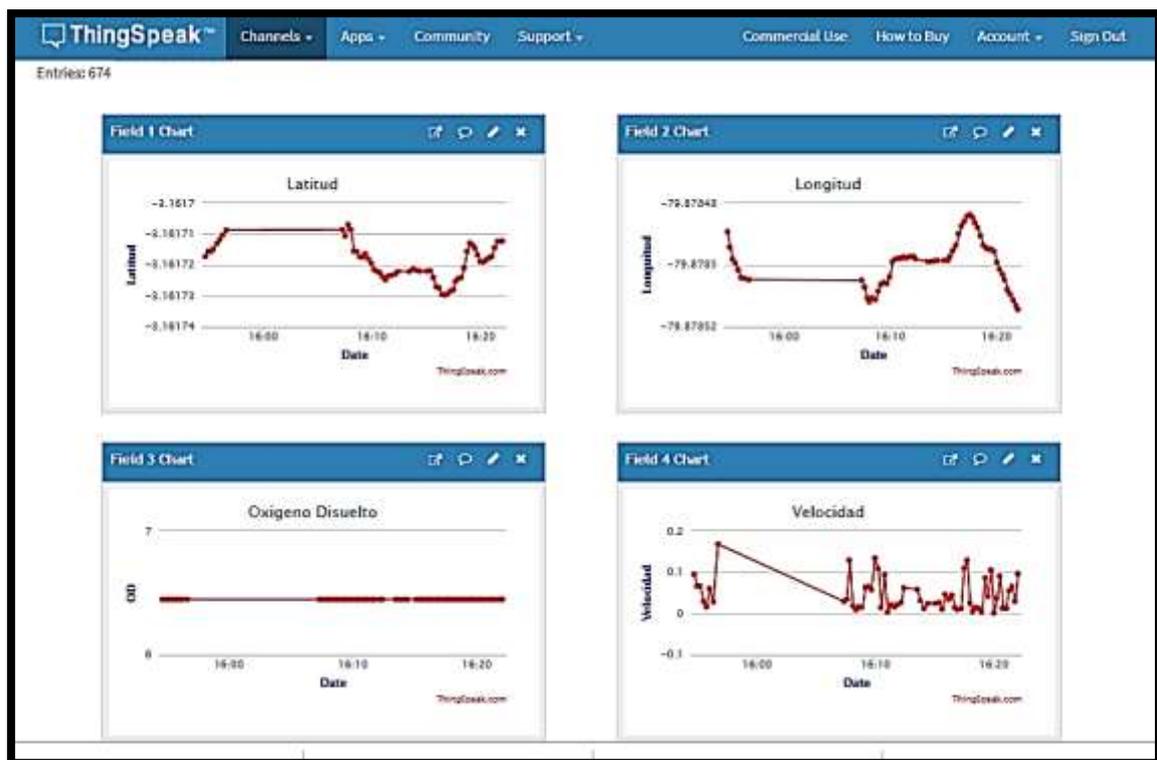
Realizado por: Carlos Estévez, José Rodas. 2019.

Procesos realizados:

- Recibir información desde la atapa de supervisión mediante puerto serial conectado de manera USB a la Raspberry Pi 3 B, para ello se configura el puerto serial de la Raspberry

mediante la línea de código “serial\_Arduino = serial. Serial('/dev/ttyACM1',9600)”

- Recoger de la cadena de caracteres los datos de los sensores obtenidos en la etapa de supervisión en nuevas variables dentro de la Raspberry.
- Almacenar dentro de la tarjeta raíz SD de la tarjeta de desarrollo
- Proceder a la transmisión de los variables a la base de datos en la plataforma ThingSpeak mediante la línea de texto `conn.request("POST", "/update", params, headers)`, donde en la variable `params` es donde se asignan las variables a transmitir. Una vez subidas a la nube se pueden apreciar cómo se observa en la figura 25-2.



**Figura 25-2: HMI ThingSpeak**

Realizado por: Carlos Estévez, José Rodas. 2019.

#### 2.7.3.4. Diagrama de flujo de la etapa de Desplazamiento

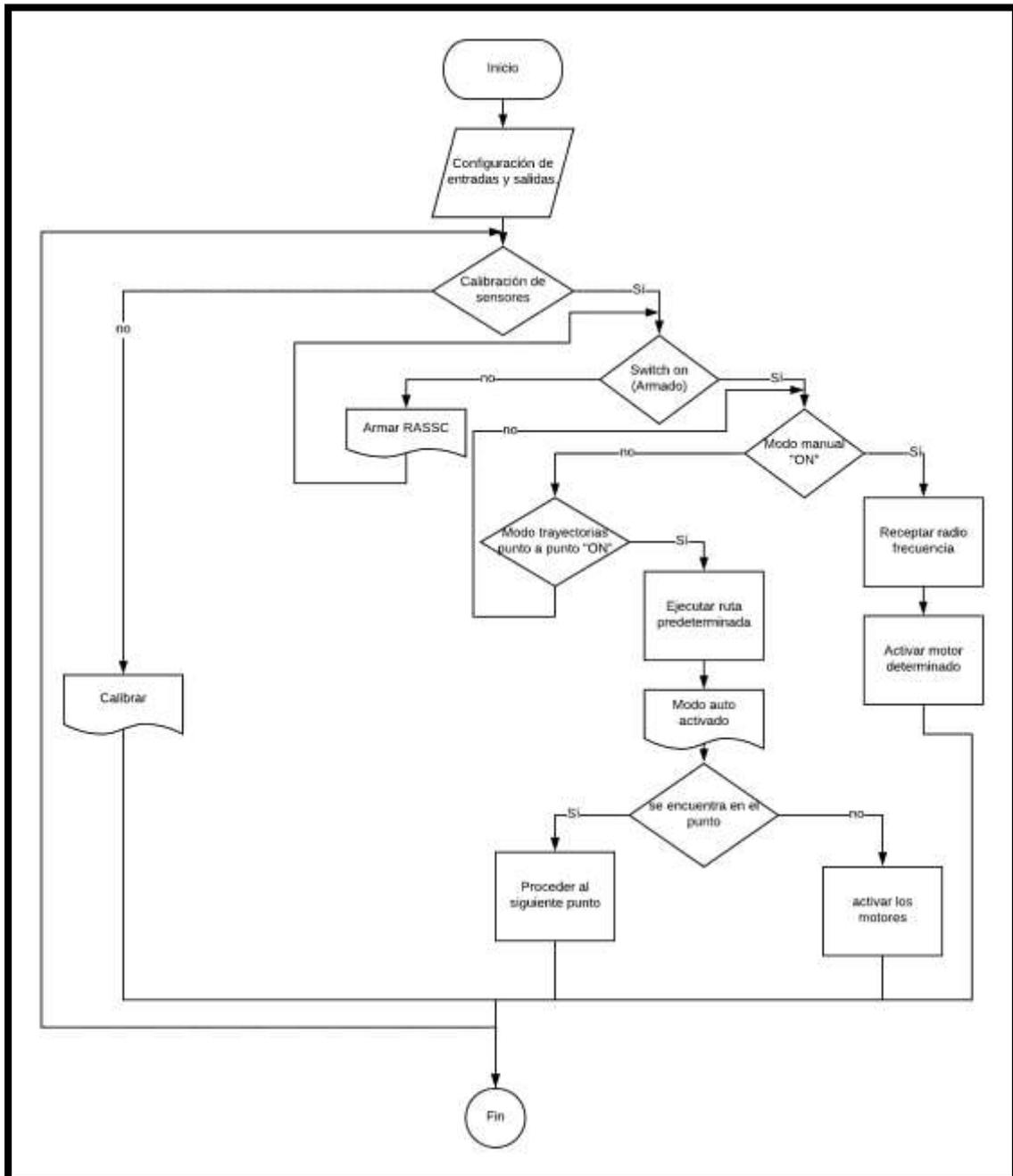
La programación realizada se representa en el diagrama de flujo en la figura 26-2. Se realiza lo siguiente:

Procesos realizados:

- Instalación del driver a utilizar en este caso Rover en su configuración Boat.
- Calibración de GPS, magnetómetro y joystick, estas son esenciales para la traslación del

RASSC dado que estas dan las respectivas órdenes a los motores para el movimiento.

- Configuración de los modos punto a punto al cual lo llamaremos automático y modo manual para realizar los cambios mediante radio frecuencia.
- Calibrar los niveles mínimos y máximos de voltaje de las baterías.

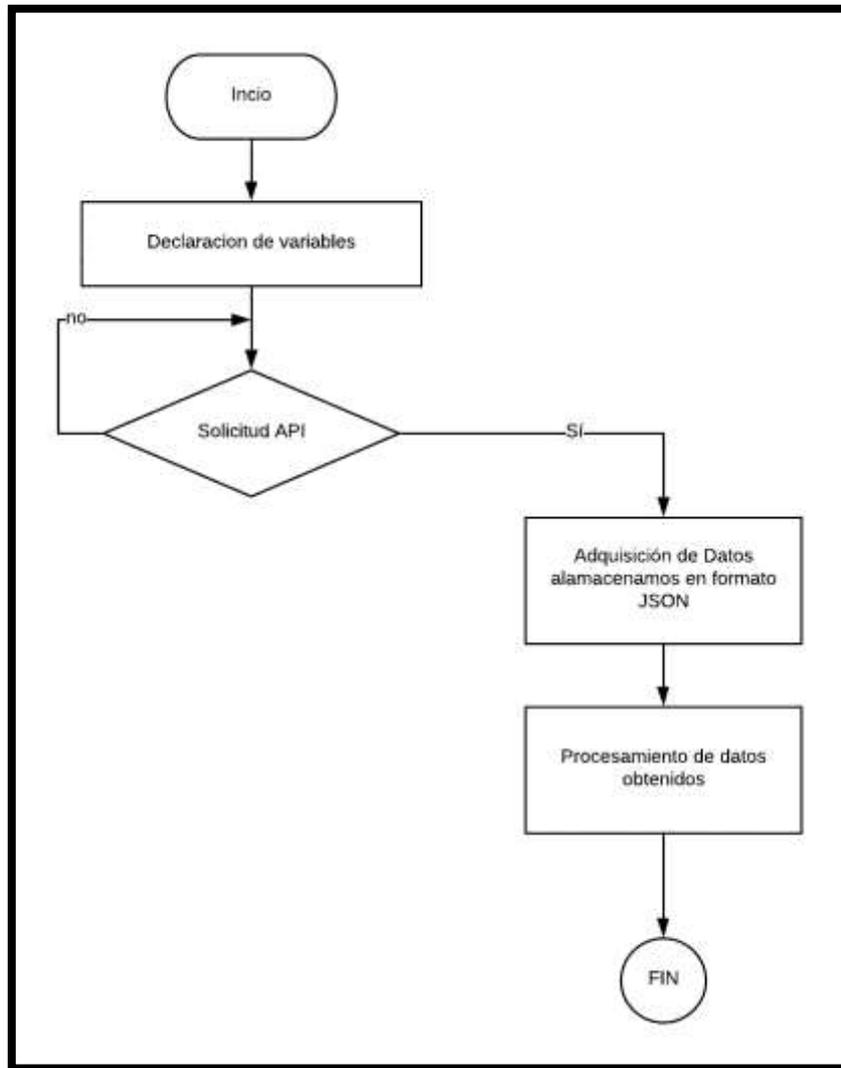


**Figura 26-2: Diagrama de flujo etapa de desplazamiento**

Realizado por: Carlos Estévez, José Rodas. 2019.

#### 2.7.4. App móvil para la presentación de información

RASSC posee una herramienta móvil desarrollada en Android Studio 2019, la cual permite visualizar la información transmitida a la nube mediante la plataforma ThingSpeak su diagrama de flujo se puede apreciar en la figura 27-2.



**Figura 27-2: Diagrama de flujo App**  
Realizado por: Carlos Estévez, José Rodas. 2019.

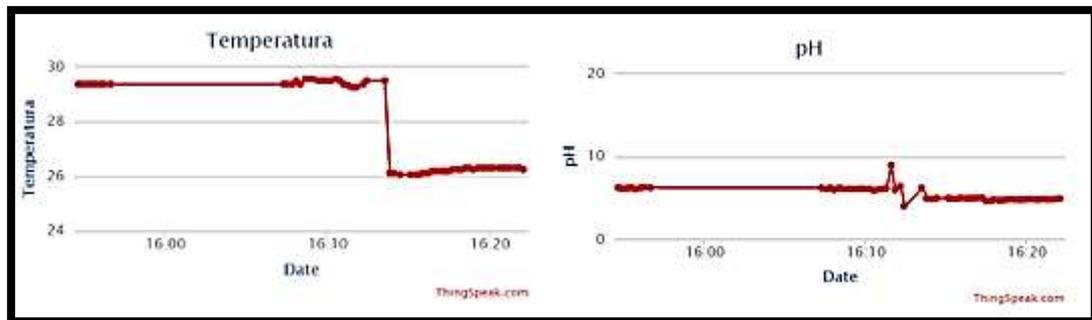
Permite observar 6 variables las cuales son latitud, longitud, Oxígeno disuelto, Velocidad, pH y temperatura como se puede apreciar en la figura 28-2, también permite descargar la base de datos actual colgada en la nube y brinda información sobre los rangos aconsejables de las variables de calidad del agua en camaroneras. La programación de este se agrega en anexos.



**Figura 28-2: App móvil**

Realizado por: Carlos Estévez, José Rodas. 2019.

A su vez en la plataforma ThingSpeak se puede apreciar las gráficas de los datos obtenidos en variación al tiempo como se observa en la figura 29-2.



**Figura 29-2: ThingSpeak base de datos**

Realizado por: Carlos Estévez, José Rodas. 2019.

## 2.8. Especificaciones y configuraciones del RASSC

**Tabla 9-2:** Especificaciones y configuraciones

<b>Módulos y sensores</b>	Estructura del vehículo	1 flota central y 2 flotadores de fibra de vidrio
	Sensor de oxígeno	Rango 0.01 – 100 mg/L Precisión de $\pm 0.05$ mg/L
	Sensor de PH	Rango 0 - 14 Precisión $\pm 0.2$
	Sensor de temperatura	Rango -10 – 85 °C Precisión $\pm 0.5$
	Fuente de alimentación	3 baterías Li-Po
	AutoPilot	Planificación y ejecución de trayectorias
	GPS	Estándar
	Joystick	10 canales
	Almacenamiento Interno	32GB
	Motores	Configuración diferencial. A prueba de agua
	App Móvil	Plataforma Android
<b>Operatividad</b>	Rango de telemetría	Hasta 140m en línea de vista
	Velocidad de operación	Óptima 1 m/s (3.6 km/h o 1.9 kn), Máxima 5 m/s (18 km/h o 9.7 kn)
	Tiempo de operación	1 hora a velocidad óptima
	Área de mapeo	Hasta una hectárea
<b>Dimensiones</b>	Largo	85 cm
	Ancho	60 cm
	Alto	30 cm
	Peso	10 kg

Realizado por: Carlos Estévez, José Rodas. 2019.

## CAPÍTULO III

### 3. ANÁLISIS Y RESULTADOS

En este apartado se realizan las pruebas de validación del prototipo RASSC, mediante experimentos de funcionamiento de los sensores comparados con dispositivos de características similares, validación del recorrido de las trayectorias planificadas y alcance máximo a nivel de conectividad de telemetría del joystick, estabilidad del sistema implementado y el análisis de costos del prototipo que se observa en la figura 1-3.



**Figura 1-3: Prototipo final implementado.**

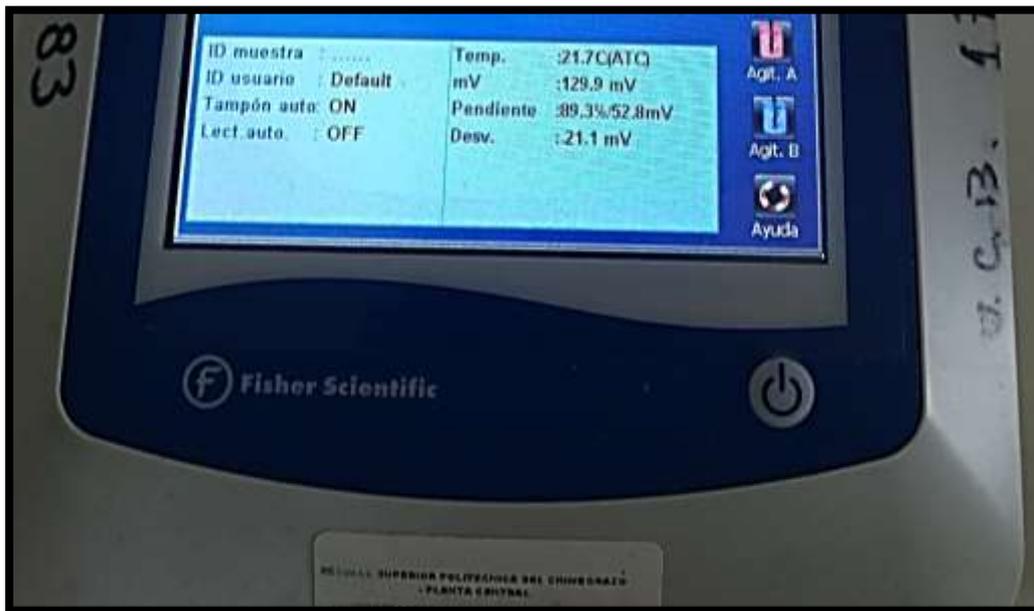
**Realizado por:** Carlos Estévez, José Rodas. 2019.

#### 3.1. Pruebas de la etapa de supervisión

En esta sección se muestra las validaciones de los sensores empleados en la etapa de supervisión, mediante pruebas de medición.

### 3.1.1. Caracterización del sensor de temperatura

Para validar el porcentaje de error que posee el sensor ds18b20, se comparó con el equipo Fisher Scientific XL150 mostrado en la Figura 2-3, que posee un rango de -5 a 105 °C con una precisión de  $\pm 0,2$  °C y una resolución de 0,1 °C. Se adquirieron 25 muestras durante un día, como se muestra en la Tabla 1-3.



**Figura 2-3: Equipo XL150, midiendo temperatura.**

Realizado por: Carlos Estévez, José Rodas. 2019.

**Tabla 1-3:** Mediciones realizadas entre el XL150 y el DS18B20.

<b>Número de muestra</b>	<b>Equipo XL150 (°C)</b>	<b>Sensor DS18B20 (°C)</b>	<b>Error absoluto</b>
1	20.1	20.33	0.23
2	20.7	20.51	0.19
3	20.9	20.92	0.02
4	21.8	21.76	0.04
5	23.2	23.40	0.20
6	22.9	22.67	0.23
7	22.3	22.28	0.02
8	24.0	24.32	0.32
9	24.6	24.56	0.04
10	24.9	24.57	0.33
11	25.2	25.69	0.49
12	23.6	23.24	0.36
13	24.7	24.22	0.48
14	25.5	25.33	0.17
15	24.4	24.67	0.27
16	22.5	22.51	0.01
17	19.8	19.56	0.24
18	19.4	19.77	0.37
19	18.9	18.85	0.05
20	18.7	18.88	0.18
21	17.1	17.12	0.02
22	17.3	17.66	0.36
23	18.5	18.45	0.05
24	19.9	19.93	0.03
25	19.6	19.88	0.28

Realizado por: Carlos Estévez, José Rodas, 2019.

Se observa en la Tabla 1-3 que el error absoluto máximo es de 0.49, concluyendo que el sistema no incorpora más error del que ya está establecido en el datasheet del sensor DS18B20 de  $\pm 0.5$  °C.

### 3.1.2. Caracterización del sensor de pH

Para validar el porcentaje de error que posee el sensor E-201, se comparó con el equipo Fisher Scientific XL150 mostrado en la Figura 3-3, que posee un rango de -2 a 20 pH con una precisión de  $\pm 0,002$  pH y una resolución de 0,01 pH. Se adquirieron 25 muestras durante un día, como se muestra en la Tabla 2-3.



**Figura 3-3: Equipo XL150, midiendo pH.**  
Realizado por: Carlos Estévez, José Rodas. 2019.

**Tabla 2-3:** Mediciones realizadas entre el XL150 y el E-201.

<b>Número de muestra</b>	<b>Equipo XL150 (pH)</b>	<b>Sensor E-201 (pH)</b>	<b>Error absoluto</b>
1	4.23	4.32	0.09
2	5.12	4.93	0.19
3	4.60	4.66	0.06
4	4.11	4.15	0.04
5	7.35	7.17	0.18
6	5.56	5.62	0.06
7	4.38	4.45	0.07
8	8.89	8.71	0.18
9	8.12	8.05	0.07
10	7.53	7.47	0.06
11	6.74	6.88	0.14
12	6.10	5.92	0.18
13	5.23	5.04	0.19
14	4.33	4.51	0.18
15	9.66	9.53	0.13
16	9.03	8.87	0.16
17	8.12	8.14	0.02
18	7.54	7.38	0.16
19	6.33	6.51	0.18
20	10.65	10.57	0.08
21	8.67	8.68	0.01
22	6.77	6.68	0.09
23	5.89	5.87	0.02
24	4.32	4.46	0.14
25	3.22	3.29	0.07

**Realizado por:** Carlos Estévez, José Rodas, 2019.

Se observa en la Tabla 2-3 que el error absoluto máximo es de 0.19, concluyendo que el sistema no incorpora más error del que ya está establecido en el datasheet del sensor E-201 de  $\pm 0.2$  °C.

### 3.1.3. Caracterización del sensor de Oxígeno disuelto

Para validar el porcentaje de error que posee el sensor EZO-DO de Atlas-Scientific, se comparó con el equipo el sensor referencial YSI A500 mostrado en la Figura 4-3, que posee un rango de 0 a 50 mg/L con una precisión de  $\pm 0,2$  mg/L y una resolución de 0,01 mg/L. Se adquirieron 25 muestras durante un día, como se muestra en la Tabla 3-3.



**Figura 4-3: Equipo YSI A500.**  
Realizado por: Carlos Estévez, José Rodas. 2019.

**Tabla 3-3:** Mediciones realizadas entre el YSI A500 y el E-201.

Número de muestra	Equipo YSI A500 (mg/L)	Sensor Atlas (mg/L)	EZO-OD Scientific	Error absoluto
1	5.34	5.32		0.02
2	4.35	4.30		0.05
3	5.36	5.35		0.01
4	5.35	5.31		0.04
5	7.39	7.39		0
6	5.37	5.36		0.01
7	5.39	5.38		0.01
8	4.41	4.42		0.01

<b>9</b>	3.41	3.39	0.02
<b>10</b>	5.41	5.39	0.02
<b>11</b>	5.45	5.42	0.03
<b>12</b>	5.70	5.69	0.01
<b>13</b>	3.42	3.40	0.02
<b>14</b>	5.55	5.56	0.01
<b>15</b>	4.55	4.55	0
<b>16</b>	4.35	4.34	0.01
<b>17</b>	6.46	6.48	0.02
<b>18</b>	3.33	3.33	0
<b>19</b>	5.05	5.07	0.02
<b>20</b>	4.51	4.52	0.01
<b>21</b>	6.23	6.20	0.03
<b>22</b>	7.11	7.11	0
<b>23</b>	6.10	6.11	0.01
<b>24</b>	4.75	4.76	0.01
<b>25</b>	4.68	4.65	0.03

Realizado por: Carlos Estévez, José Rodas. 2019.

Se observa en la Tabla 3-3 que el error absoluto máximo es de 0.05, concluyendo que el sistema no incorpora más error del que ya está establecido en el datasheet del sensor EZO-OD de  $\pm 0.05$  mg/L.

#### **3.1.4. Prueba de estabilidad de la etapa de supervisión**

Para realizar el análisis de la etapa de supervisión se realizó una prueba que consiste en tomar una muestra ya conocida su pH en este caso Agua destilada, esta se la coloco dentro de un cooler para mantener estables las condiciones de temperatura y oxígeno disuelto y se procedió a la toma de mediciones una cada minuto dentro de un periodo de 20 minutos los resultados obtenidos se muestran en la siguiente tabla.

**Tabla 4-3:** Estabilidad de la etapa de supervisión

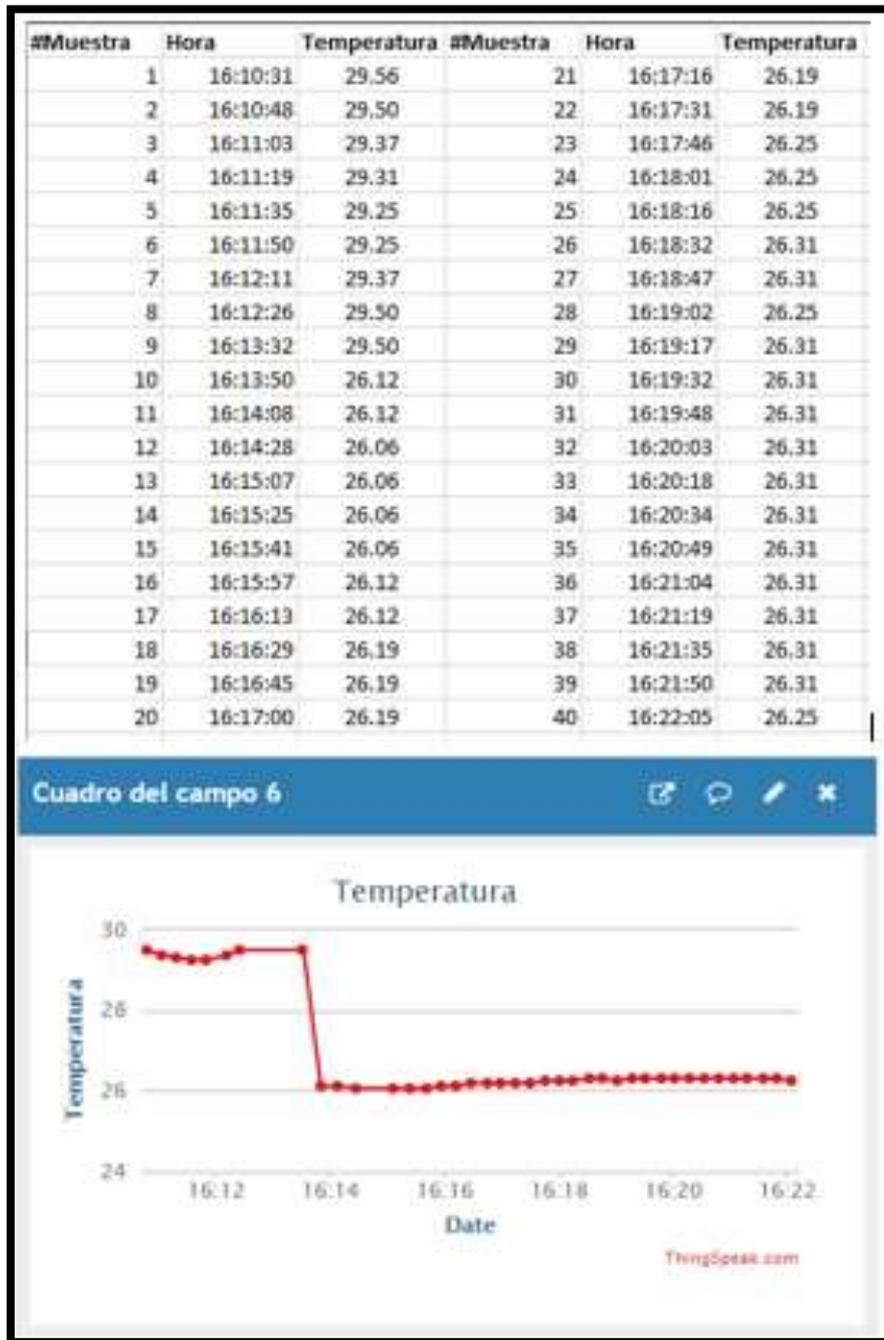
Muestra	DS18B20	E-120	EZO-OD
	Temperatura	pH	Oxígeno Disuelto
	(°C)	(pH)	(mg/L)
1	20,15	4,21	3,70
2	20,20	4,30	3,70
3	20,20	4,25	3,71
4	20,11	4,31	3,69
5	20,13	4,24	3,70
6	20,18	4,24	3,68
7	20,15	4,23	3,68
8	20,15	4,21	3,69
9	20,11	4,21	3,70
10	20,11	4,31	3,69
11	20,10	4,25	3,70
12	20,23	4,21	3,68
13	20,21	4,29	3,71
14	20,19	4,31	3,71
15	20,19	4,25	3,70
16	20,09	4,33	3,69
17	20,10	4,19	3,69
18	20,17	4,20	3,70
19	20,16	4,21	3,71
20	20,16	4,23	3,72
<b>Media (x)</b>	20,1545	4,249	3,6975
<b>Desviación Estándar (<math>\sigma</math>)</b>	0,040678618	0,042649736	0,010897247
<b>Coefficiente de variación (C.V) %</b>	0,201833925	1,003759384	0,294719334

Realizado por: Carlos Estévez, José Rodas. 2019.

Se puede apreciar en la tabla anterior los coeficientes de variación de cada sensor son los siguientes: 0.2% para el sensor de temperatura DS18B20, 1% para el sensor de pH E-201 y para el sensor de Oxígeno 0.29%, Representando una variación aceptable dado que es menor al 10% donde se considera una buena lectura.

### 3.1.5. Caracterización de la comunicación entre las etapas de supervisión y transmisión

Dicha prueba consistió en realizar mediciones durante 10 minutos, donde se toman 4 muestras por minuto, tomándose un total de 40 muestras. En la etapa de supervisión se transmitió mediante puerto serial a la etapa de transmisión, donde se realizó el almacenamiento interno en un archivo, siendo este comparado con lo recibido en la nube como se observa en la Figura 5-3.

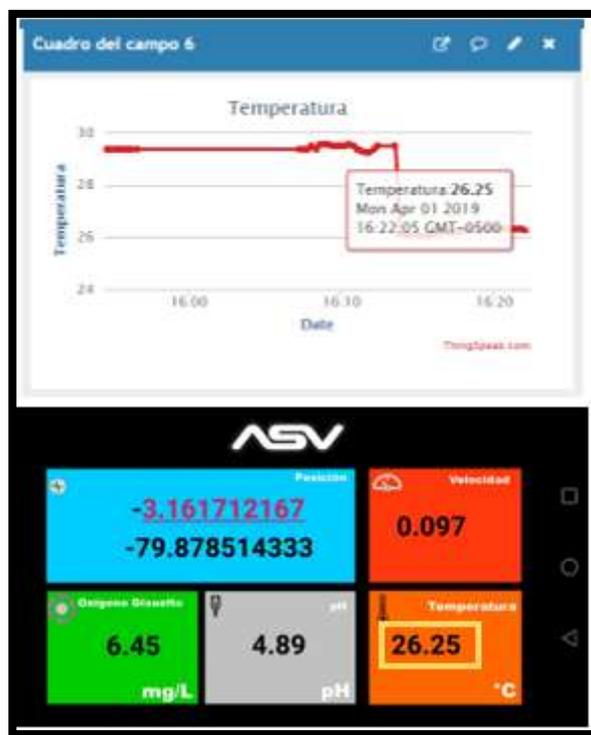


**Figura 5-3: Comparación entre datos de almacenamiento interno y enviados a la nube.**

Realizado por: Carlos Estévez, José Rodas. 2019.

Se comprobó que no existe perdidas en el almacenamiento directo por medio de USB, pero en el caso de la trasmisión de datos hacia la nube se recibieron 39 datos de 40 que equivale un error del 2.5%, siendo este un error aceptable.

También se realizó una prueba de conectividad entre la base de datos de la nube y la APP móvil desarrollada, se comprobó que cada dato recibido en el ThingSpeak era actualizado en la App como se observa en la figura 6-3.

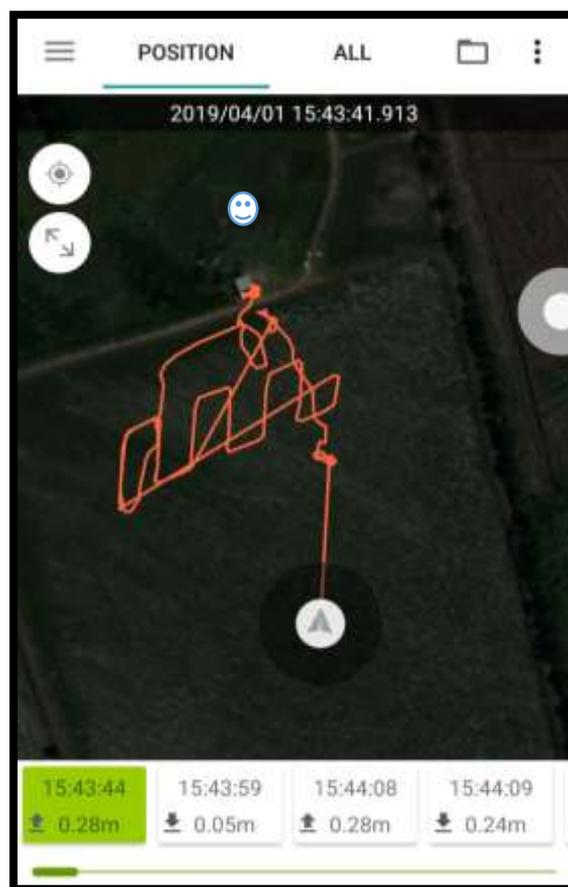


**Figura 6-3: ThingSpeak y App móvil**  
Realizado por: Carlos Estévez, José Rodas. 2019.

## 3.2. Pruebas de la etapa de desplazamiento

### 3.2.1. Caracterización del alcance máximo de telemetría

En esta sección se muestra las validaciones del alcance de la telemetría de la Pixhawk (bote en navegación) y el joystick (operador con app Tower), en línea recta, sin obstáculos mediante pruebas realizadas como se muestra en la Figura 7-3 y Figura 8-3.



**Figura 7-3: Historial de recorrido del sistema RASSC.**  
Realizado por: Carlos Estévez, José Rodas. 2019.



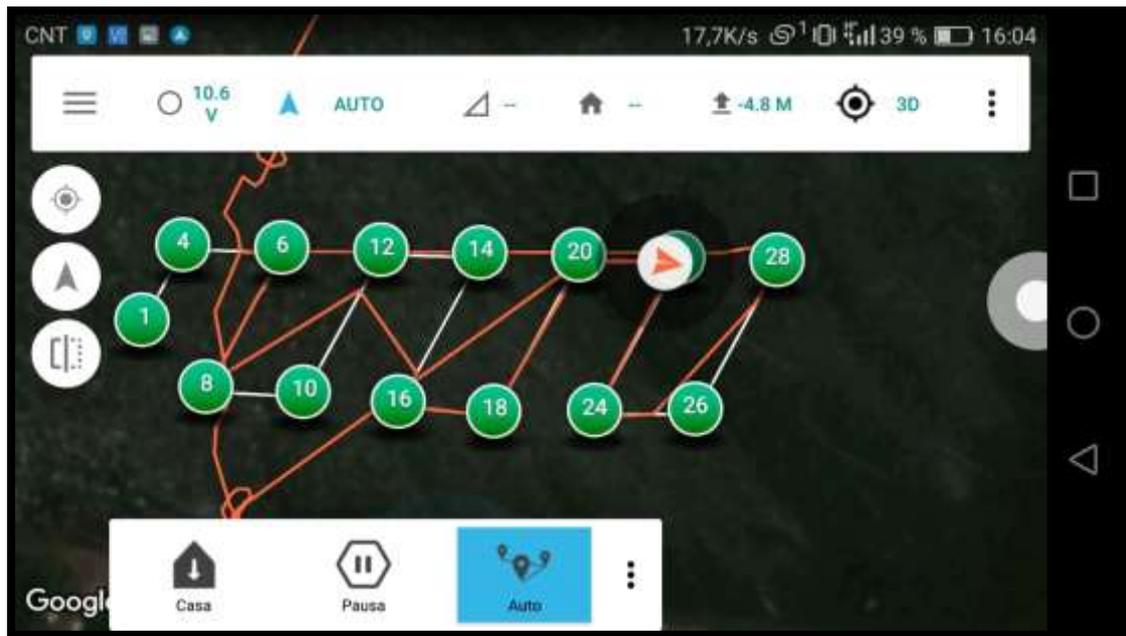
**Figura 8-3: Distancia máxima de conectividad por telemetría.**

Realizado por: Carlos Estévez, José Rodas. 2019.

Para esta prueba llevamos el sistema RASSC en modo manual hasta perder conectividad con el mismo y tomamos ese punto (1) como se aprecia en la Figura 8-3 y se determinó la distancia máxima a la cual podemos estar alejados del sistema. Con esto se comprobó que el sistema cumple con los 100 m requeridos, con un alcance máximo de 148.8 m.

### 3.2.2. Caracterización de las trayectorias por punto del planificador

En esta prueba se evalúa la capacidad del sistema RASSC para pasar por cada uno de los puntos señalados como way-points en el planificador, como se muestra en la figura 9-3. De tal manera que el porcentaje de efectividad se realiza tomando en cuenta el número de puntos por los que si cruzó el ASV del total de los planificados.



**Figura 9-3: Way-points del planificador**

Realizado por: Carlos Estévez, José Rodas. 2019.

Como se logra apreciar en la anterior figura el sistema RASSC consigue seguir las rutas trazadas por el planificador y los puntos por los que no logró pasar son 3 de 28 a una velocidad de 4 m/s, obteniendo un 10.7% de error. En la misma prueba a una velocidad de 1 m/s logró cruzar por 27 de los 28 puntos, logrando un 96.4% de efectividad.

Tiempo de ejecución de la trayectoria planificada:

A una velocidad de 1 m/s el tiempo de duración fue de aproximadamente 30 minutos, para la supervisión de una hectárea de la camaronera. Esto comprueba que el sistema RASSC tiene una autonomía para cubrir 2 misiones continuas de una hectárea.

### 3.3. Análisis de costo

En la siguiente tabla se puede apreciar el costo de cada uno de los elementos utilizados en el dispositivo RASSC.

**Tabla 5-3:** Tabla de costos

<b>MÓDULOS</b>	<b>COMPONENTES</b>	<b>CANTIDAD</b>	<b>COSTOS (\$)</b>
<b>Microcontroladores</b>	Arduino mega 2560	1	20
	Raspberry Pi 3	1	90
	Pixhawk 2.4.8	1	100
<b>Sensor OD</b>	EZO-DO Atlas Scientific	1	400
<b>Sensor pH</b>	E-201	1	50
<b>Sensor de temperatura</b>	DS18B20	1	10
<b>Motor Brushless y controlador ESC</b>	GoolRC	2	150
<b>Fuente Baterías LiPo</b>	Ovonic Air 3000 mAh	2	100
	Ovonic Air 1400 mAh	1	30
<b>GPS</b>	ublox NEO 6M	2	70
<b>Periféricos Pixhawk</b>	Power module	1	20
	Buzzer	1	10
	Telemetría	1	40
	Switch	1	10
	Cable OTG	1	5
<b>Regulador de voltaje</b>	LM2596	1	10
<b>Joystick</b>	Flysky fs-i6s	1	100
<b>Estructura</b>	Fibra de vidrio	1	300
	Hélices	2	20
	Moldes de yeso	3	100
	Moldes de madera	3	50
	Cardan y varilla	2	40
	Sujetadores	4	20
	Pernos M4	50	20
<b>Total</b>			<b>1765</b>

Realizado por: Carlos Estévez, José Rodas. 2019

El costo total de implementación del prototipo está valorado en \$1765, que comparado con sistemas de monitoreo multisensorial es significativamente menor con lo que respecta a su inversión inicial y mantenimiento.

## CONCLUSIONES

- Se construyó un dispositivo que posee una autonomía energética capaz de analizar piscinas camaroneras con una extensión de una hectárea, supervisar variables de calidad de agua dentro de las mismas, almacenar y transmitir las mediciones de los sensores a una base de datos a la cual se accede y visualiza mediante una aplicación móvil.
- El sistema RASSC dado que realiza planeación y ejecución de trayectorias entregando mediciones geo-referenciadas precisas de los sensores, lo convierte en un vehículo autónomo de superficie de reconocimiento hidrográfico.
- Se comprobó que el error absoluto máximo de temperatura es de  $0.49\text{ }^{\circ}\text{C}$  en la calibración del sensor DS18B20 comparado con el equipo XL150, considerado aceptable debido a que la variación  $\pm 0.5\text{ }^{\circ}\text{C}$  dentro de las piscinas no es considerado un cambio abrupto para el camarón.
- El error absoluto máximo es de 0.19 en la escala de pH en la calibración del sensor E-201 comprado con el equipo XL150, por lo cual se considera un error aceptable ya que el dispositivo no añade error al sensor que posee una presión de  $\pm 0.2$  en su hoja de especificaciones.
- El error absoluto máximo que presenta el sensor de oxígeno disuelto EZO-OD es de  $0.5\text{ mg/L}$ , lo cual se considera un error aceptable, ya que el dispositivo no añade error al sensor el cual presenta una presión de  $\pm 0.5$  en su hoja de especificaciones.
- El dispositivo RASSC presento un coeficiente de variación menor al 10% basándonos en la repetitividad en las mediciones de sus tres sensores, lo que representa una estabilidad aceptable.
- La totalidad de los datos medidos fueron almacenados en la memoria interna en un archivo xls y el 97% de los datos se transmitieron de manera efectiva al servidor de ThingSpeak.

- EL sistema RASSC en la conectividad por telemetría alcanza una distancia máxima de 148 m en línea de vista.
- La flotabilidad del sistema RASSC a la que fue diseñado, es satisfactoria, dado que permite un desplazamiento fluido y a su vez otorga cierto grado impermeabilidad a los componentes electrónicos dentro del casco principal. Cabe recalcar que en los flotadores los elementos internos son a prueba de agua.
- El sistema RASSC implementado por su flexibilidad y versatilidad en el control de las variables de calidad del agua, se considera como una mejor alternativa ante los sistemas de monitoreo multisensorial en los que por la extensión de las piscinas requieren una mayor inversión y mantenimiento en sensores para cubrir más puntos de supervisión.
- El costo total de implementación del prototipo está valorado en \$1765, logrando una diferencia del 40% entre el sistema RASSC y el ASV SONOBOT de Evo Logics, lo que se considera un porcentaje aceptable de inversión dadas las características y la aplicación a camaroneras con los respectivos sensores y la integración de la aplicación móvil.
- El sistema RASSC logró un 96% de efectividad en la prueba de trayectorias por puntos y demostró que tiene la autonomía suficiente para realizar misiones de una hectárea.

## RECOMENDACIONES

- Realizar una investigación más profunda con el fin de mostrar resultados de mapeo en 2D ya que la base de datos es compatible con software de sistemas de información geográfica como lo es ArcGIS.
- Realizar estudios de la variable de oxígeno disuelto para determinar la ubicación más apropiada de los aireadores que son indispensables para la oxigenación del agua.
- Integrar otros sensores al sistema RASSC de manera que las mediciones tomadas sean más versátiles como, por ejemplo, sensores para salinidad, dureza y sólidos suspendidos en el agua.
- Extender el alcance de la comunicación Wireless a través de una estación WLAN con una antena direccional de alta ganancia, con lo que se ampliaría el área de mapeo.
- El prototipo RASSC al ser un sistema versátil y flexible le permite enfocarse a diversos campos por lo que se recomienda estudiar una nueva aplicación para realizar batimetrías por sonar en presas de centrales hidroeléctricas midiendo la sedimentación de estas.

## BIBLIOGRAFÍA

**AQUAFEED.** *Monitoreo de la calidad de agua del estanque para mejorar la producción de camarones y peces* [en línea]. 2018. [Consulta: 7 enero 2019]. Disponible en: <http://www.aquafeed.co/monitoreo-de-la-calidad-de-agua-del-estanque-para-mejorar-la-produccion-de-camarones-y-peces/>.

**ATLAS SCIENTIFIC LLC.** *Dissolved Oxygen* [en línea]. 2019. [Consulta: 4 febrero 2019]. Disponible en: [https://www.atlas-scientific.com/product\\_pages/kits/do\\_kit.html](https://www.atlas-scientific.com/product_pages/kits/do_kit.html).

**BALNOVA.** *Oxígeno en estanques* [en línea]. 2014. [Consulta: 13 febrero 2019]. Disponible en: <https://www.balnova.com/oxigeno-en-estanques-de-camaron/>.

**BARDACH, J.** *Acuicultura: crianza y cultivo de organismos marinos y de agua dulce*. México: AGT, 1990. ISBN 968-463-022-0.

**BARRIENTOS, A., PEÑIN, L., BALAGUER, C. y ARACIL, R.** *Fundamentos de Robótica*. 2nd. España: Mc Graw Hill, 2007.

**CAMPS PARÉ, R., CASILLAS SANTILLÁN, L., COSTA, D., GINESTÀ, M., ESCOFET, C. y PÉREZ, O.** *Base de Datos* [en línea]. Barcelona-España: Universitat Oberta de Catalunya, 2005. pp. 5,8. ISBN 84-9788-269-5. Disponible en: <https://www.uoc.edu/masters/oficiales/img/913.pdf>.

**CAPACHO, J.R. y NIETO, W.** *Diseño de base de datos* [en línea]. Barranquilla-Colombia: Universidad del Norte, 2017. [Consulta: 15 febrero 2019]. pp. 1-3. Disponible en: <https://ebookcentral.proquest.com/lib/espochsp/reader.action?docID=5309026&ppg=6>.

**CARRETERO, I.** *Técnico en Piscifactorías*. Madrid - España: Cultural, 2002. ISBN 84-8055-

633-1.

**CHAUMET-LAGRANGE, M., LOEB, H. y YGORRA, S.** *Design of an autonomous surface vehicle (ASV)*, 2002. pp. I/120-I/125. DOI 10.1109/oceans.1994.363851.

**CHEN, M.C.M., ZHAN, Q.Z.Q. y CAI, S.C.S.** *Control System Design of an Autonomous Underwater Vehicle*, 2006. *2006 IEEE Conference on Robotics, Automation and Mechatronics*, pp. 495-499. DOI 10.1109/RAMECH.2006.252736.

**ECURED, MOTOR PASO A PASO** [en línea]. 2018. [Consulta: 20 febrero 2019]. Disponible en: [https://www.ecured.cu/Motor\\_paso\\_a\\_paso](https://www.ecured.cu/Motor_paso_a_paso).

**ELECTRONICS, RADIOFRECUENCIA** [en línea]. 2015. [Consulta: 20 enero 2019]. Disponible en: <https://www.elprocus.com/difference-between-pam-pwm-ppm/>.

**ERHARDT, T.**, En: fibra de carbono, *Tecnología Textil Básica 3: fibras sintéticas*. México: Trillas, 1991. pp. 44-45. ISBN 968-24-0800-8.

**EVOLOGICS GMBH, Evo Logics** [en línea]. 2019. [Consulta: 12 febrero 2019]. Disponible en: <https://evologics.de/sonobot-system>.

**FÁBREGAS, F.** *Aprender Raspberry Pi: con 100 ejercicios prácticos*. Barcelona-España: Marcombo, 2016. pp. 17,120-122. ISBN 978-84-267-2253-9.

**GARCÍA GONZÁLEZ, A.** *IDE Arduino* [en línea], 2015. [Consulta: 8 marzo 2019]. Disponible en: <http://panamahitek.com/el-setup-y-el-loop-en-arduino/>.

**GARRIDO, R.L.** *Internet de las cosas (IoT)* [en línea], 2015. [Consulta: 18 enero 2019]. Disponible en: <http://hdl.handle.net/10609/42812>

**GOOGLE DEVELOPERS.** *AndroidStudio* [en línea], 2018. [Consulta: 19 enero 2019].  
Disponible en: <https://developer.android.com/guide/platform/?hl=es-419>.

**GRAFENO, B.** *Tipos de baterías clasificación* [en línea], 2016. [Consulta: 14 febrero 2019].  
Disponible en: <http://bateriasdegrafenopara.com/tipos-de-baterias/>.

**DEPARTAMENTO DE LA GUAJIRA.** *Sistema de posicionamiento global* [en línea]. 2007.  
[Consulta: 15 enero 2019]. Disponible en:  
[ftp://gisweb.ciat.cgiar.org/DAPA/planificacion/GEOMATICA/GPS/GPS\\_Modulo.pdf](ftp://gisweb.ciat.cgiar.org/DAPA/planificacion/GEOMATICA/GPS/GPS_Modulo.pdf).

**JURADO LARREA, F.G.** *Análisis estadístico de las enfermedades del camarón: Caso Guayas*  
[en línea]. 2002. S.l.: ESPOL. Disponible en:  
<http://www.dspace.espol.edu.ec/xmlui/handle/123456789/4112>.

**LOPEZ, A.** *Crianza y Producción de Camarones*. Lima-Perú: Ripalme, 2006. pp. 66-71,77,80,88-89,92. ISBN 9972-840-31-X.

**MARTINEZ, A.** *Arduino* [en línea], 2013. [Consulta: 8 marzo 2019]. Disponible en:  
<http://ovtoaster.com/quesarduino/>.

**MARTÍNEZ CÓRDOVA, L.R.** *Alimentos y estrategias de alimentación para una acuacultura sustentable*. México. D.F.- México: AGT Editor, 2015. pp. 20-24,118-119. ISBN 978-607-518-078-6.

**MARTÍNEZ CÓRDOVA, R.,** *Camaronicultura Sustentable: Manejo y evaluación*. México: Trillas, 2009. pp. 25-26. ISBN 978-607-17-0036-0.

**NEWMAN, T.** *ASV C-CAT 3* [en línea], 2018. [Consulta: 23 febrero 2019]. Disponible en:  
<https://geo-matching.com/uploads/default/m/i/migrationtwtsr4.pdf>.

**NIKU, S.B.**, *Intriduction to Robotics: Analysis, Control, Aplications*. 2nd. S.l.: s.n., 2010.

**NOLASCO VALENZUELA, J.S.** *Desarrollo de Aplicaciones Móviles con Android*. Lima-Perú: Macro EIRL, 2013. pp. 15. ISBN 978-612-304-106-9.

**NOVILLO VICUÑA, J.P. y VACA SIMBALA, R.D.** *Implementación de Interfaz electrónica para la medición y registro de temperatura en una piscina camaronera mediante aplicación Web* [en línea] (tesis) UTMACH, Machala, Ecuador. 2015. [Consulta: 24 febrero 2019]. Disponible en: <http://repositorio.utmachala.edu.ec/handle/48000/5034>.

**PROMETEC**. Radio Transmisor RC[en línea], 2014. [Consulta: 21 enero 2019]. Disponible en: <https://www.prometec.net/elegir-transmisor-receptor-radio/>.

**RASPBERRY PI**. *Raspberry Pi 3*[en línea], 2017. Disponible en: <https://www.raspberrypi.org/>.

**REVISTA LÍDERES**. *Industria nacional de camarón reflotó con fuerza* [en línea], 2017. [Consulta: 8 enero 2019]. Disponible en: <https://www.revistalideres.ec/lideres/industria-nacional-camaron-refloto-fuerza.html>.

**RODGER, R.** *Desarrollo de Aplicacioners en la Nube para Dispositivos Móviles*. Madrid-España: Anaya Multimedia, 2012. ISBN 978-84-415-3188-8.

**SCHWEBER, B.**, *Detecte la temperatura con eficacia en aplicaciones de IoT usando tecnología de estado sólido* [en línea]. 2019. [Consulta: 26 Febrero 2019]. Disponible en: <https://www.digikey.com/es/articles/techzone/2019/jan/effectively-sense-temperature-iot-applications--solid-state-technology>

**SICILIANO, B., SCIAVICCO, L., VILLANI, L. y ORIOLO, G.**, *Robotics: Modelling, Planning and Control*. Springer Science & Business, Italia. 2010. DOI 10.1007/978-1-84628-642-1

**THIRUNAVUKKARASU, G., PATRICK, L., CHAMPION, B., CHUA, L., HUYNH, V.T. y JOORDENS, M.** *Desing and development of a low-cost Autonomous Surface Vehicle*, 2017. *2017 12th System of Systems Engineering Conference, SoSE 2017*, DOI 10.1109/SYSSOSE.2017.7994972.

**TRUJILLO, L., RIVERA, L. y HARDY, E.** *Estrategias Naturales para Mejorar el Crecimiento y la salud en los cultivos de camarón en Ecuador*, 2017. *Bionatura*,

**VILLAR, S.A., SOLARI, F.J., MENNA, B. V. y ACOSTA, G.G.** *Obstacle detection system design for an autonomous surface vehicle using a mechanical scanning sonar*, 2017. *2017 17th Workshop on Information Processing and Control, RPIC 2017*, vol. 2017-Janua, pp. 1-6. DOI 10.23919/RPIC.2017.8211620.

**WANG, J., GU, W. y ZHU, J.** *Design of an autonomous surface vehicle used for marine enviroment monitoring*, 2009. *Proceedings - International Conference on Advanced Computer Control, ICACC 2009*, DOI 10.1109/ICACC.2009.50.

**XIAOJING, S., MING, C. y JIANG, Y.** *Water Enviroment Monitoring System Based on Neitrl Networks for Shrimp Cultivation*, 2009. *IEEE*,

# ANEXOS

## ANEXO A: EZO-DO Atlas-Scientific Datasheet



The image shows the cover of the EZO-DO datasheet. It features a yellow background with the AtlasScientific logo at the top, which includes a stylized bird icon and the text 'AtlasScientific Environmental Robotics'. Below the logo, the version 'V 4.8' is indicated. The main title 'EZO-DO™' is prominently displayed in large white letters, with the subtitle 'Embedded Dissolved Oxygen Circuit' underneath. A table of specifications is on the left, and a top-down view of the circuit board is on the right. The board shows a central black diamond-shaped chip, various pins labeled GND, TX (SDA), RX (SCL), D.O., VCC, PRB, and PGND, and other components like a circular sensor and several resistors. A RoHS logo is in the bottom right corner, and a 'PATENT PROTECTED' notice is at the bottom. A footer note states 'This is an evolving document, check back for updates.'

**AtlasScientific**  
Environmental Robotics  
V 4.8

# EZO-DO™

Embedded Dissolved Oxygen Circuit

Reads	<b>Dissolved Oxygen</b>
Range	0.01 – 100+ mg/L 0.1 – 400+ % saturation
Accuracy	+/- 0.05 mg/L
Response time	1 reading per sec
Supported probes	Any galvanic probe
Calibration	1 or 2 point
Temperature, salinity and pressure compensation	Yes
Data protocol	UART & I <sup>2</sup> C
Default I <sup>2</sup> C address	97 (0x61)
Operating voltage	3.3V – 5V
Data format	ASCII

GND TX (SDA) RX (SCL)

D.O. VCC PRB PGND

EZO™

RoHS

PATENT PROTECTED

Written by Jordan Davis  
Designed by Frank Pines

This is an evolving document, check back for updates.

# Calibration theory

The most important part of calibration is watching the readings during the calibration process. It's easiest to calibrate the device in its default state (UART mode, continuous readings). Switching the device to PC mode after calibration **will not** affect the stored calibration. If the device must be calibrated in PC mode be sure to request readings continuously so you can see the output from the probe.

The Atlas Scientific EZO™ Dissolved Oxygen circuit, has a flexible calibration protocol, allowing for **single point** or **dual point** calibration.

## Calibrate first, compensate later.

Temperature, salinity and pressure compensation values have no effect on calibration.

## Single point calibration



1. Pull off and discard cap from the Dissolved Oxygen probe. *(only used to protect probe during shipping)*
2. Let the Dissolved Oxygen probe sit, exposed to air until readings stabilize (5–30 sec).
3. Calibrate using the command "Cal".
4. After calibration is complete, you should see readings **-9.09 – 9.1Xmg/L**  
*(only if temperature, salinity and pressure compensation are at default values)*

## Dual point calibration (optional)

Only perform this calibration if you require accurate readings **below 1.0 mg/L**

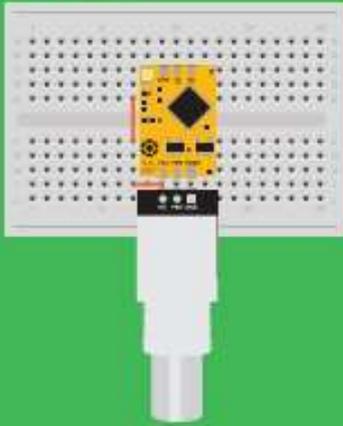
After you have calibrated using the command "Cal"



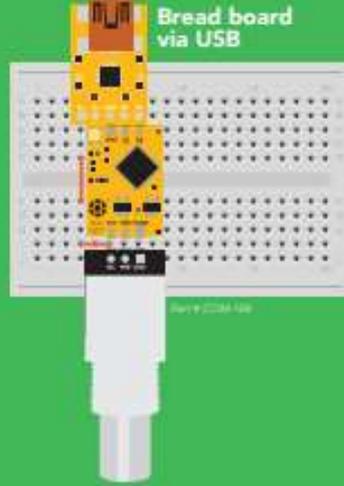
1. Stir probe in Zero D.O. calibration solution to remove trapped air, *(which could cause readings to go high)*.
2. Let the probe sit in Zero D.O. calibration solution until readings stabilize (0:10 – 1:30).
3. Calibrate using the command "Cal,0".

# ✓ Correct wiring

Bread board



Bread board via USB



Carrier board



USB carrier board

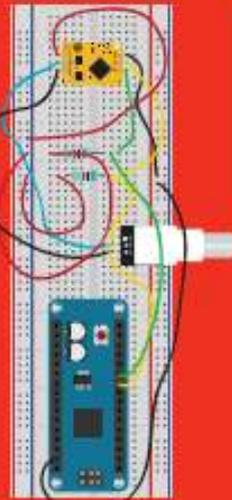


# X Incorrect wiring

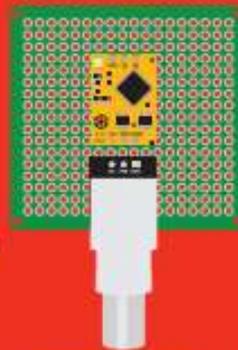
Extended leads



Sloppy setup

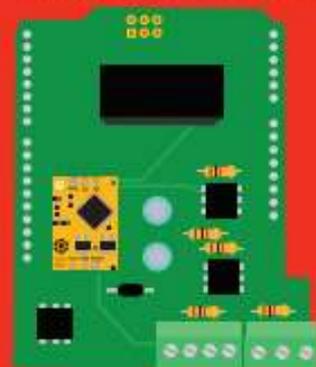


Perfboards or Protoboards



**NEVER**  
use Perfboards  
or Protoboards

\*Embedded into your device



\*Only after you are familiar  
with EZO™ circuits operation

# UART mode

8 data bits    no parity  
1 stop bit    no flow control

**Baud** 300  
1,200  
2,400  
**9,600 default**  
19,200  
38,400  
57,600  
115,200

**RX**  
Data in



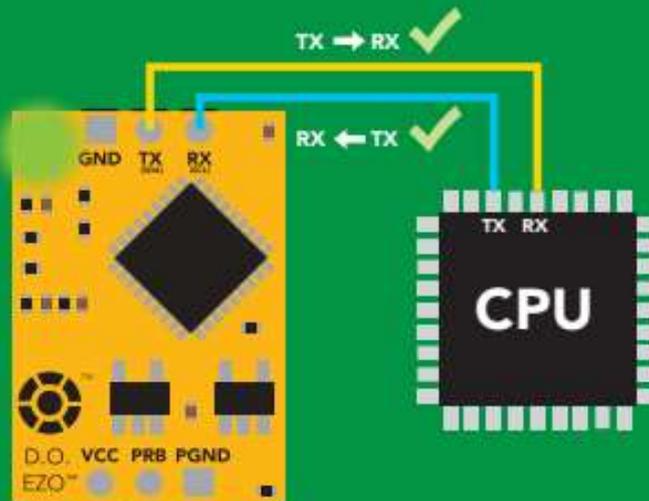
**TX**  
Data out



**Vcc** 3.3V – 5.5V



0V    Vcc    0V



## Data format

<b>Reading</b>	<b>D.O.</b>	<b>Data type</b>	<b>floating point</b>
<b>Units</b>	<b>mg/L &amp; (% sat)</b> <small>when enabled</small>	<b>Decimal places</b>	mg/L = 2 % sat = 1
<b>Encoding</b>	<b>ASCII</b>	<b>Smallest string</b>	<b>4 characters</b>
<b>Format</b>	<b>string</b> <small>(CSV string when % sat is enabled)</small>	<b>Largest string</b>	<b>16 characters</b>
<b>Terminator</b>	<b>carriage return</b>		

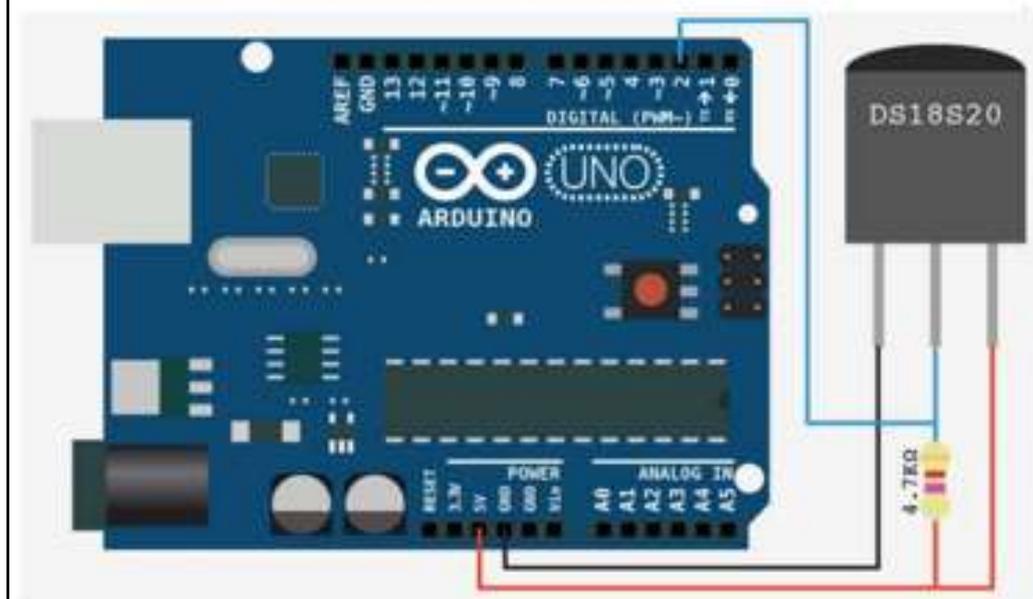
## ANEXO B: Hoja de especificaciones Sensor de temperatura DS18B20

### Sensor de temperatura DS18B20

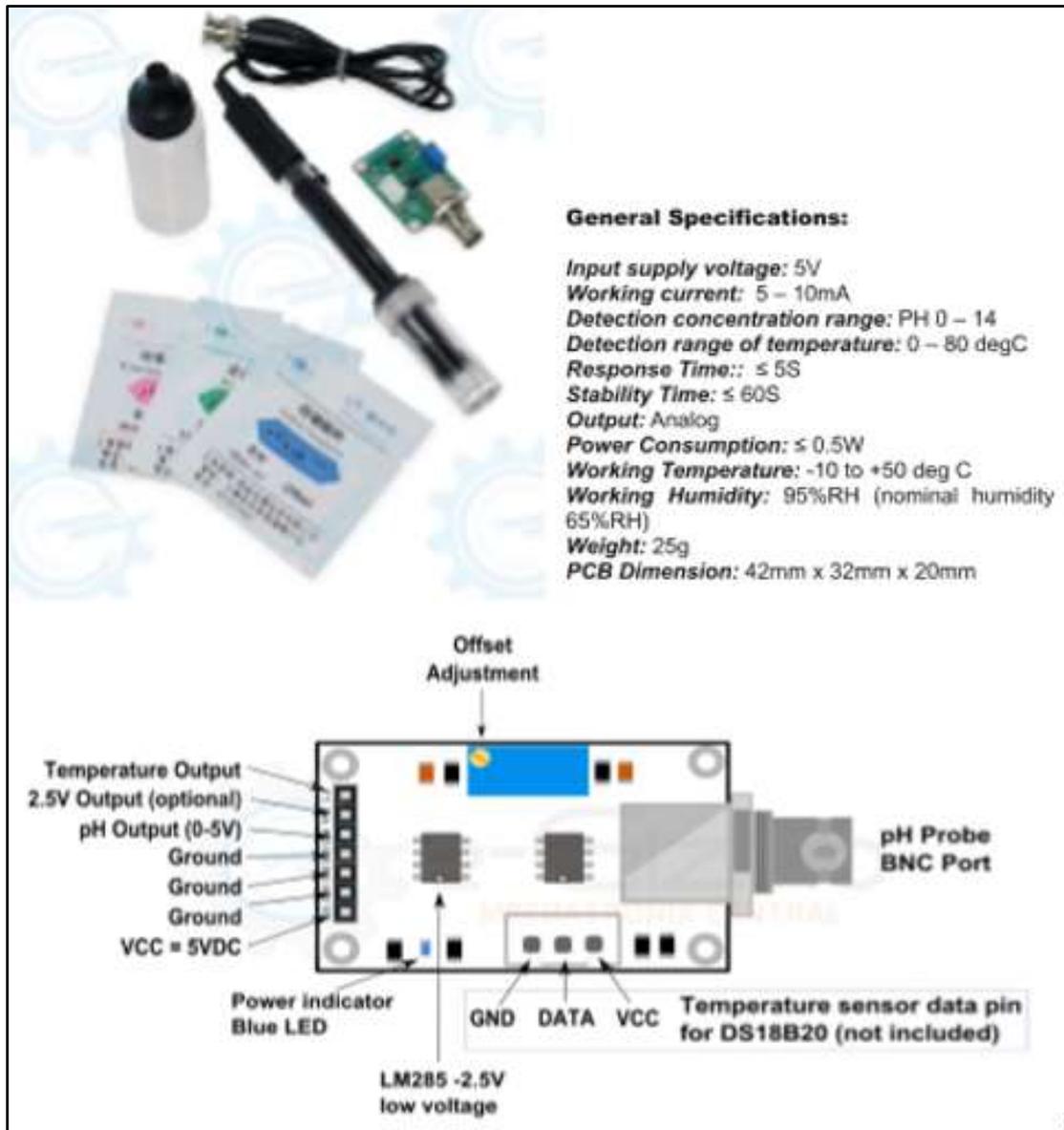


**Especificaciones:** Incluye sensor de temperatura DS18B20.

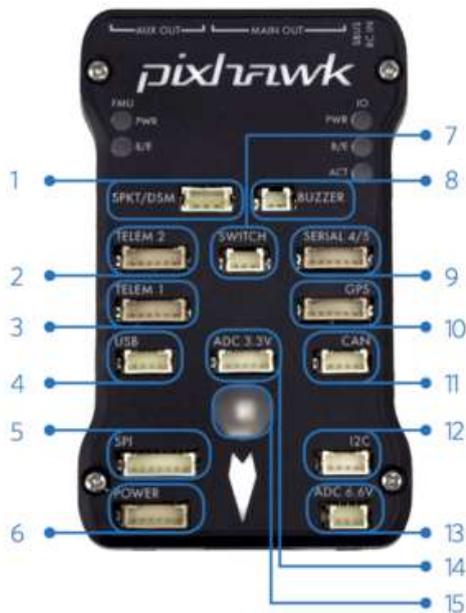
- ◊ Sensor Digital
- ◊ Resolución de 9 y 12 bits
- ◊ Rango de operación de -50 a 125 grados Centígrados
- ◊ Precisión +- 0.5 grados
- ◊ Protocolo OneWire



## ANEXO C: Hoja de especificaciones Sensor pH E-201



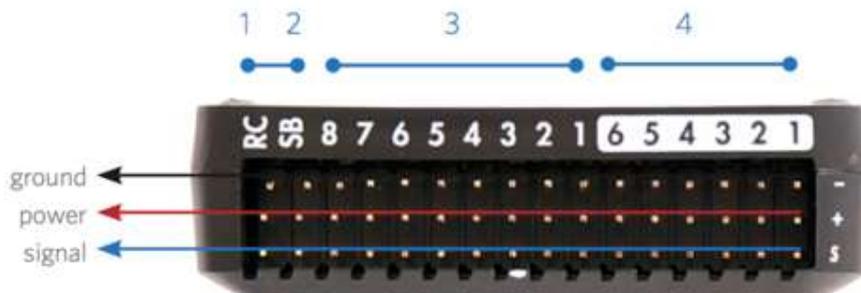
## ANEXO D: DATASHEET PIXHAWK



- 1 Spektrum DSM receiver
- 2 Telemetry (radio telemetry)
- 3 Telemetry (on-screen display)
- 4 USB
- 5 SPI (serial peripheral interface) bus
- 6 Power module
- 7 Safety switch button
- 8 Buzzer
- 9 Serial
- 10 GPS module
- 11 CAN (controller area network) bus
- 12 I<sup>2</sup>C splitter or compass module
- 13 Analog to digital converter 6.6 V
- 14 Analog to digital converter 3.3 V
- 15 LED indicator



- 1 Input/output reset button
- 2 SD card
- 3 Flight management reset button
- 4 Micro-USB port



- 1 Radio control receiver input
- 2 S.Bus output
- 3 Main outputs
- 4 Auxiliary outputs

## Características clave

- Sistema principal en chip: [STM32F427](#)
  - CPU: ARM<sup>®</sup> Cortex<sup>®</sup> M4 de 180 MHz con FPU de precisión simple
  - RAM: 256 KB SRAM (L1)
- Sistema-en-chip a prueba de fallos: STM32F100
  - CPU: 24 MHz ARM Cortex M3
  - RAM: 8 KB SRAM
- Wifi: ESP8266 externo
- GPS: U-Blox<sup>®</sup> 7/8 (Hobbyking<sup>®</sup>) / U-Blox 6 (Robótica 3D)
- Flujo óptico: [unidad de flujo PX4](#)
- Entradas de alimentación redundantes y failover automático.
- Interruptor de seguridad externo
- Indicador visual principal LED multicolor
- Indicador de audio piezoeléctrico multitono de alta potencia
- Tarjeta microSD para registro de alta velocidad durante largos períodos de tiempo

### Conectividad

- 1x I2C
- 1x CAN (2x opcional)
- 1x ADC
- 4x UART (2x con control de flujo)
- Consola 1x
- 8x PWM con anulación manual
- Entrada 6x PWM / GPIO / PWM
- Entrada S.BUS / PPM / Spektrum
- Salida S.BUS

## ANEXO E: Hoja de especificaciones De Arduino Mega 2560

### Arduino Mega2560 Rev3



The Arduino Mega 2560 is a microcontroller board based on the ATmega2560. 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 Uno, Duemilanove or Diecimila.

The Mega 2560 is an update to the [Arduino Mega](#), which it replaces.

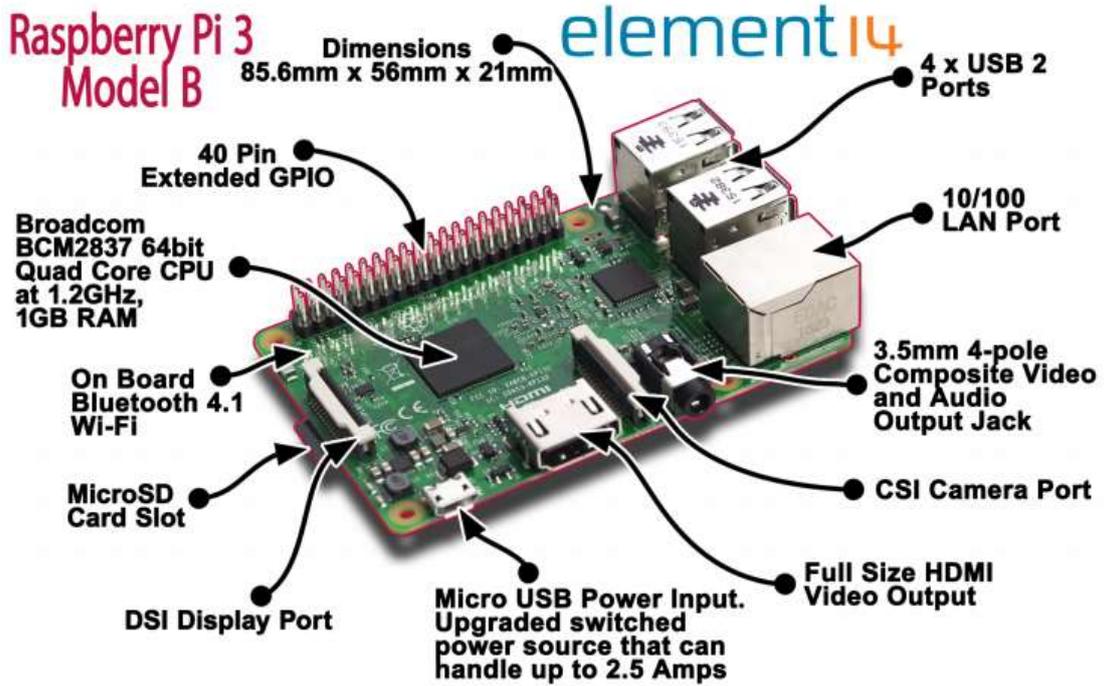
Additional features coming with the R3 version are:

- ATmega16U2 instead 8U2 as USB-to-Serial converter.
- 1.0 pinout: added SDA and SCL pins for TWI communication placed near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board and the second one is a not connected pin, that is reserved for future purposes.
- stronger RESET circuit.

#### Technical Specifications

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

ANEXO F: DATASHEET RASPBERRY PI 3



## Specifications

<b>Processor:</b>	Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4 GHz
<b>Memory:</b>	1GB LPDDR2 SDRAM
<b>Connectivity:</b>	<ul style="list-style-type: none"><li>■ 2.4 GHz and 5 GHz IEEE 802.11 b/g/n/ac wireless LAN, Bluetooth 4.2, BLE</li><li>■ Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)</li><li>■ 4 × USB 2.0 ports</li></ul>
<b>Access:</b>	Extended 40-pin GPIO header
<b>Video &amp; sound:</b>	<ul style="list-style-type: none"><li>■ 1 × full size HDMI</li><li>■ MIPI DSI display port</li><li>■ MIPI CSI camera port</li><li>■ 4 pole stereo output and composite video port</li></ul>
<b>Multimedia:</b>	H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics
<b>SD card support:</b>	Micro SD format for loading operating system and data storage
<b>Input power:</b>	<ul style="list-style-type: none"><li>■ 5V/2.5A DC via micro USB connector</li><li>■ 5V DC via GPIO header</li><li>■ Power over Ethernet (PoE)-enabled (requires separate PoE HAT)</li></ul>
<b>Environment:</b>	Operating temperature, 0–50 °C
<b>Compliance:</b>	For a full list of local and regional product approvals, please visit <a href="http://www.raspberrypi.org/products/raspberry-pi-3-model-b+">www.raspberrypi.org/products/raspberry-pi-3-model-b+</a>
<b>Production lifetime:</b>	The Raspberry Pi 3 Model B+ will remain in production until at least January 2023.

## ANEXO G: GPS U-BLOX NEO 6M

Parameter	Specification			
Receiver type	50 Channels GPS L1 frequency, C/A Code SBAS: WAAS, EGNOS, MSAS			
Time-To-First-Fix <sup>1</sup>		NEO-6G/Q/T	NEO-6M/V	NEO-6P
	Cold Start <sup>2</sup>	26 s	27 s	32 s
	Warm Start <sup>2</sup>	26 s	27 s	32 s
	Hot Start <sup>2</sup>	1 s	1 s	1 s
	Aided Starts <sup>3</sup>	1 s	<3 s	<3 s
Sensitivity <sup>4</sup>		NEO-6G/Q/T	NEO-6M/V	NEO-6P
	Tracking & Navigation	-162 dBm	-161 dBm	-160 dBm
	Reacquisition <sup>5</sup>	-160 dBm	-160 dBm	-160 dBm
	Cold Start (without aiding)	-148 dBm	-147 dBm	-146 dBm
	Hot Start	-157 dBm	-156 dBm	-155 dBm
Maximum Navigation update rate		NEO-6G/Q/M/T	NEO-6P/V	
		5Hz	1 Hz	
Horizontal position accuracy <sup>6</sup>	GPS	2.5 m		
	SBAS	2.0 m		
	SBAS + PPP <sup>7</sup>	< 1 m (2D, R50) <sup>8</sup>		
	SBAS + PPP <sup>7</sup>	< 2 m (3D, R50) <sup>9</sup>		
Configurable Timepulse frequency range		NEO-6G/Q/M/P/V	NEO-6T	
		0.25 Hz to 1 kHz	0.25 Hz to 10 MHz	
Accuracy for Timepulse signal	RMS	30 ns		
	99%	<60 ns		
	Granularity	21 ns		
	Compensated <sup>10</sup>	15 ns		
Velocity accuracy <sup>6</sup>		0.1m/s		
Heading accuracy <sup>6</sup>		0.5 degrees		
Operational Limits	Dynamics	≤ 4 g		
	Altitude <sup>10</sup>	50,000 m		
	Velocity <sup>10</sup>	500 m/s		

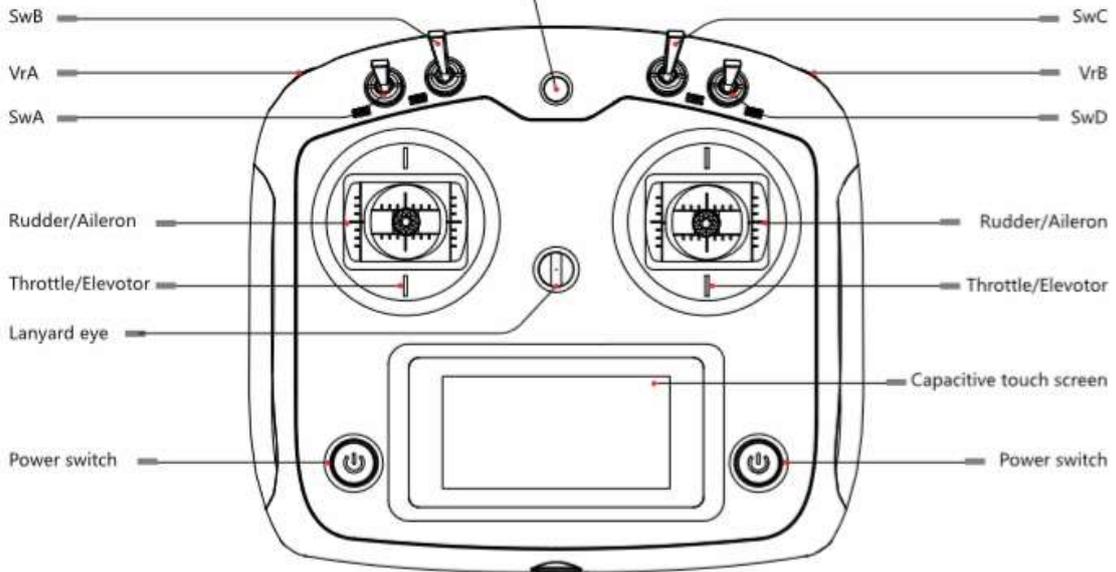
### 3.1 Absolute maximum ratings

Parameter	Symbol	Module	Min	Max	Units	Condition
Power supply voltage	VCC	NEO-6G	-0.5	2.0	V	
		NEO-6Q, 6M, 6P, 6V, 6T	-0.5	3.6	V	
Backup battery voltage	V_BCKP	All	-0.5	3.6	V	
USB supply voltage	VDDUSB	All	-0.5	3.6	V	
Input pin voltage	Vin	All	-0.5	3.6	V	
	Vin_usb	All	-0.5	VDDU SB	V	
DC current trough any digital I/O pin (except supplies)	Ipin			10	mA	
VCC_RF output current	ICC_RF	All		100	mA	
Input power at RF_IN	Prfin	NEO-6Q, 6M, 6G, 6V, 6T		15	dBm	source impedance = 50Ω, continuous wave
		NEO-6P		-5	dBm	
Storage temperature	Tstg	All	-40	85	°C	

# ANEXO H: FLY-SKY fs-i6s

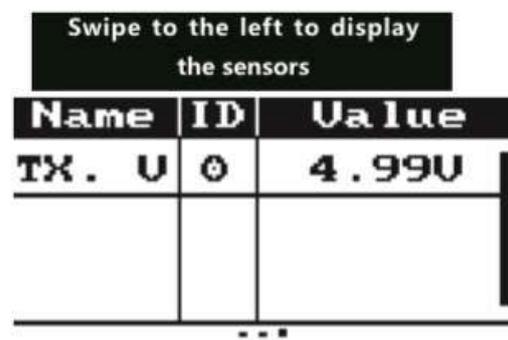
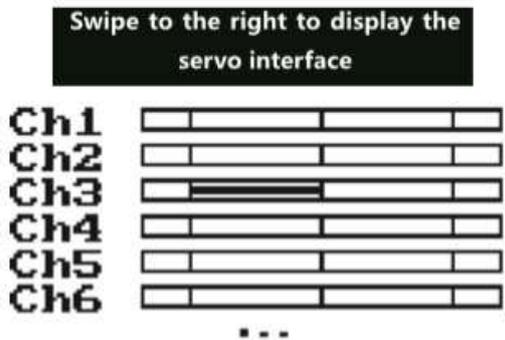
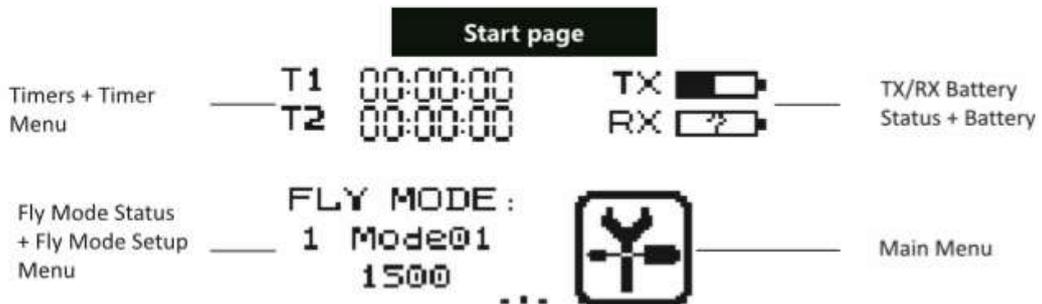
## 2.2 Transmitter Overview

Device holder mounting point

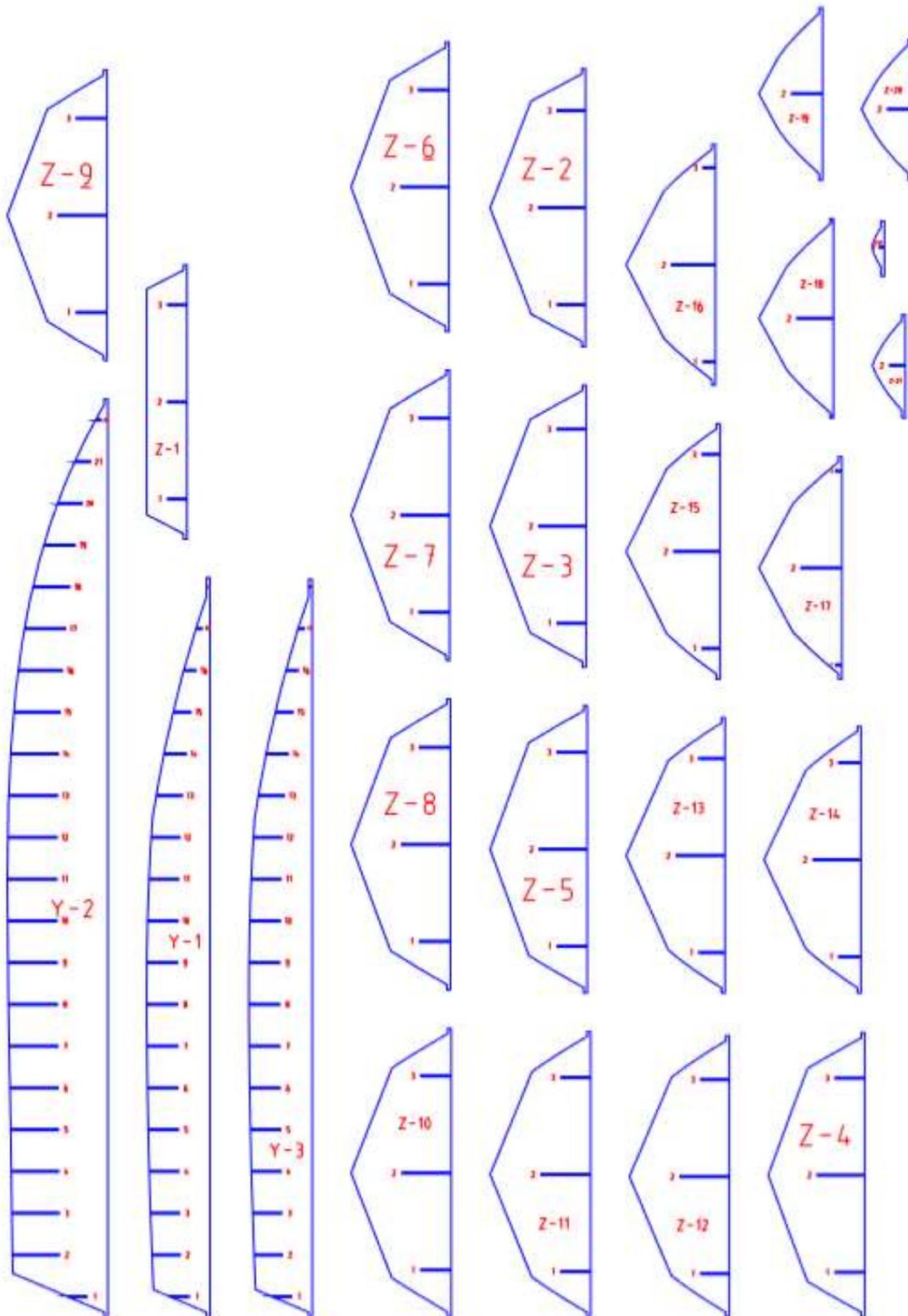


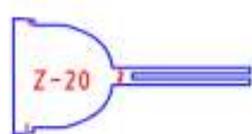
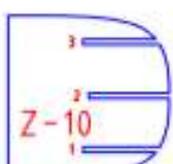
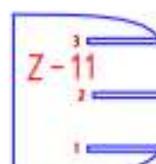
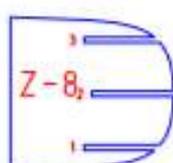
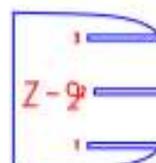
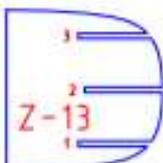
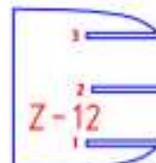
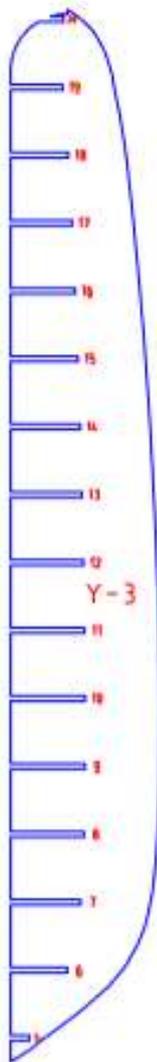
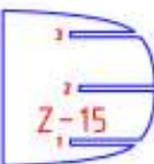
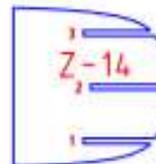
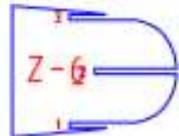
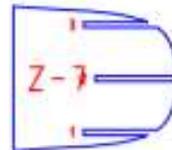
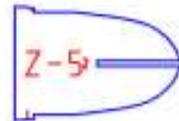
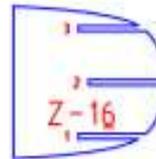
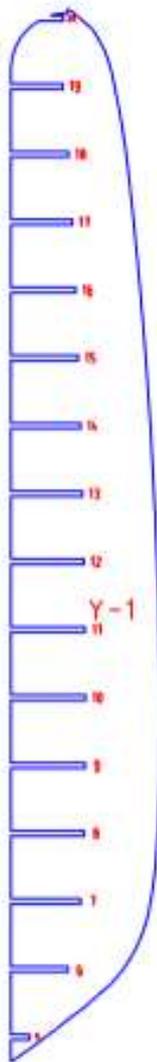
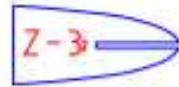
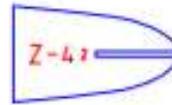
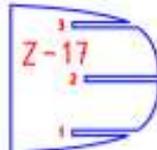
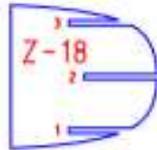
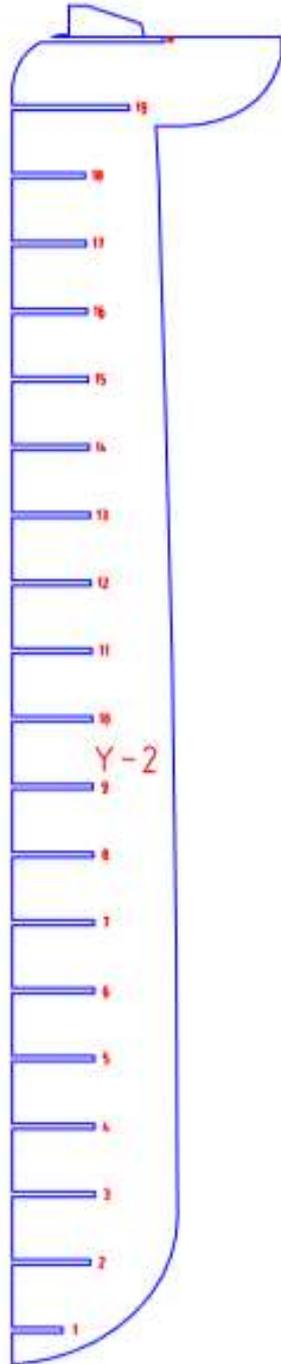
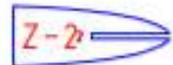
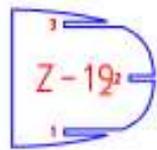
## 5.1 Home Screen

The main interface displays model.



**ANEXO I: Planos generados de Slide de Autodesk**





## ANEXO J: Construcción de la estructura

### Molde de madera



## Moldes de yeso y fibra de vidrio



**Acabado final Masillado y Pintado**



## ANEXO K: Programación Etapa de supervisión

```
#include <OneWire.h>
#include <DallasTemperature.h>

int ph_pin = A0; //This is the pin number connected to Po
int t_pin = A7;
// Pin donde se conecta el bus 1-Wire
const int pinDatosDQ = 9;

// Instancia a las clases OneWire y DallasTemperature
OneWire oneWireObjeto(pinDatosDQ);
DallasTemperature sensorDS18B20(&oneWireObjeto);

void setup() {
  Serial.begin(9600);
  //Serial.println("          PH");
  //Serial.println("ESPOCH");
  // Iniciamos el bus 1-Wire
  sensorDS18B20.begin();
}
```

## ANEXO L: Código Programación Android Studio

```
1 package com.example.myapplication;
2
3 import ...
23
24 public class MainActivity extends AppCompatActivity {
25
26     private RequestQueue queue;
27
28     private TextView mTextView;
29     private TextView mTextView2;
30     private TextView mTextView3;
31     private TextView mTextView4;
32     private TextView mTextView5;
33     private TextView mTextView6;
34
35     public String Lat;
36     public String Lat1;
37     public String Lon;
38     public String Lon1;
39     public String Alt;
40     public String Vel;
41     public String Ph;
42     public String Temp;
43     Button Odb, Phb, temb;
44
45     Handler myHandler = new Handler();
46     final Handler handler = new Handler();
47
48
49     @Override
50     protected void onCreate(Bundle savedInstanceState) {
51         super.onCreate(savedInstanceState);
```

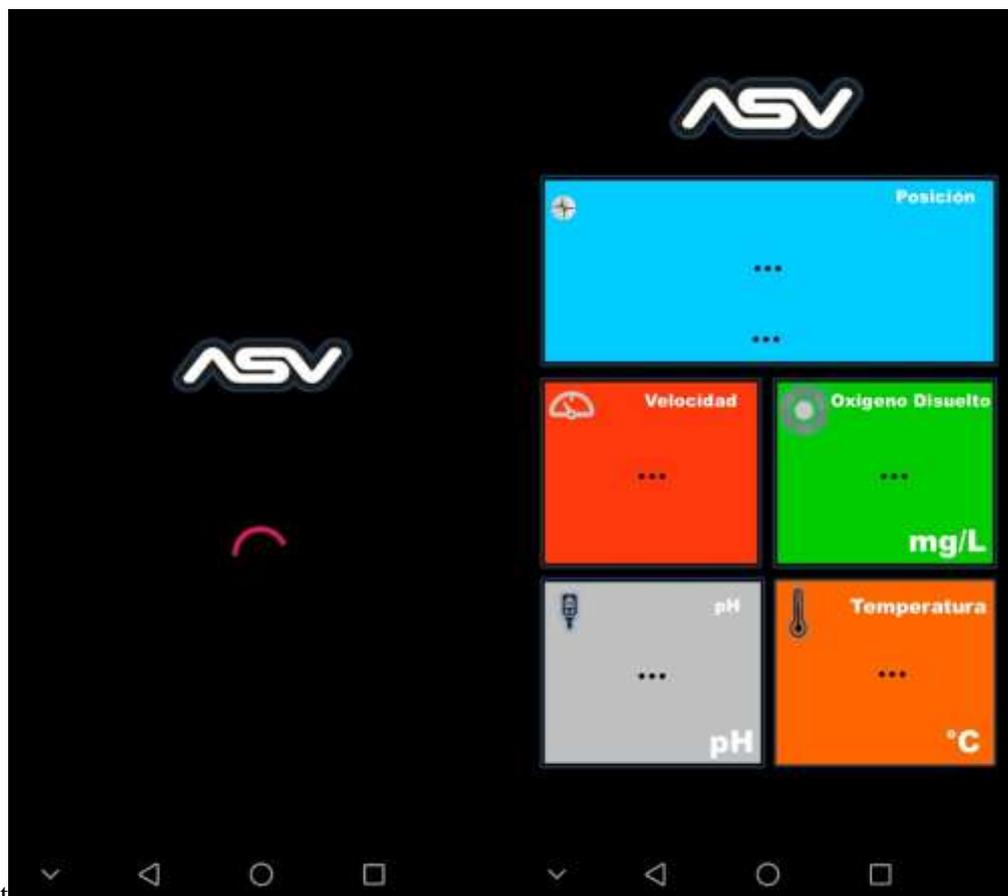
```

52
53 //Quita la barra de estado
54 requestWindowFeature(Window.FEATURE_NO_TITLE);
55 getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
56     WindowManager.LayoutParams.FLAG_FULLSCREEN);
57
58 //
59 setContentView(R.layout.activity_main);
60
61 //envia mediante botones a otra ventana
62 Odb=(Button)findViewById(R.id.button5);
63 Odb.setOnClickListener((v) - {
64     Intent intent=new Intent( packageContext: MainActivity.this, Od.class);
65     intent.putExtra( name: "Oxigenoy",Alt);
66     startActivity(intent);
67 });
68
69
70
71 Phb=(Button)findViewById(R.id.button6);
72 Phb.setOnClickListener((v) - {
73     Intent intent=new Intent( packageContext: MainActivity.this, Ph.class);
74     intent.putExtra( name: "Phv",Ph);
75     startActivity(intent);
76 });
77
78
79
80
81 temb=(Button)findViewById(R.id.button7);
82 temb.setOnClickListener((v) - {
83     Intent intent=new Intent( packageContext: MainActivity.this, Temp.class);
84     intent.putExtra( name: "Tempv",Temp);
85     startActivity(intent);
86 });
87
88
89
90
91 ///////////////////////////////////////////////////
92
93 mTextView=findViewById(R.id.textView);
94 mTextView2=findViewById(R.id.textView2);
95 mTextView3=findViewById(R.id.textView3);
96 mTextView4=findViewById(R.id.textView4);
97 mTextView5=findViewById(R.id.textView5);
98 mTextView6=findViewById(R.id.textView6);
99
100 queue = Volley.newRequestQueue( context: this);
101
102 myHandler.postDelayed(() - {
103     obtenerDatosVolley();
104     handler.postDelayed( R: this, delayMillis: 1000);
105 }, delayMillis: 0);
106
107
108
109
110
111
112
113
114 }
115
116
117 public void obtenerDatosVolley(){
118     String url = "https://api.thingspeak.com/channels/ 710046 /feeds.json?results= 10";
119
120 JSONObjectRequest request=new JSONObjectRequest(Request.Method.GET, url,  JsonRequest: null, (response) - {
121
122     try {
123         JSONArray mJSONArray = response.getJSONArray( name: "feeds");
124         for(int i=0; i<mJSONArray.length();i++) {

```

```
129
130
131     JSONObject mJsonObject = mJSONArray.getJSONObject(i);
132     Lat= mJsonObject.getString( name: "field1");
133     Lon= mJsonObject.getString( name: "field2");
134     Alt= mJsonObject.getString( name: "field3");
135     Vel= mJsonObject.getString( name: "field4");
136     Ph=mJsonObject.getString( name: "field5");
137     Temp=mJsonObject.getString( name: "field6");
138     //Lat1=Lat.substring(0,7);
139     //Lon1=Lon.substring(0,7);
140     mTextView.setText(Lat);
141     mTextView2.setText(Lon);
142     mTextView3.setText(Alt);
143     mTextView4.setText(Vel);
144     mTextView5.setText(Ph);
145     mTextView6.setText(Temp);
146
147
148
149     } catch (JSONException e) {
150     e.printStackTrace();
151
152
153     }, new Response.ErrorListener() {
154     @Override
155     public void onErrorResponse(VolleyError error) {
156
157
158     }
159     });
160     queue.add(request);
161
```

## ANEXO M: Capturas Aplicación móvil



## ANEXO N: Programación Etapa de Trasmisión

```
#!/usr/bin/python
import httplib, urllib
import threading, os
import random
from gps import *
import time

import socket
import serial

serial_Arduino = serial.Serial('/dev/ttyACM1',9600)
serial_Arduino.flushInput()

key = 'DL6V6IGKCDMH50NQ' # Thingspeak Write API Key

gpsd = gps(mode=WATCH_ENABLE|WATCH_NEWSTYLE)
print 'latitude\tlongitude\ttime utc\t\t\taltitude\tepv\tept\tspeed\tclimb' # '\t' = TAB to try and output the data in columns.

try:

    while True:
        report = gpsd.next() #

        if report['class'] == 'TPV':

            print getattr(report,'lat',0.0),"\t",
            print getattr(report,'lon',0.0),"\t",
            print getattr(report,'time',''),"\t",
            print getattr(report,'alt','nan'),"\t\t",
            print getattr(report,'epv','nan'),"\t",
            print getattr(report,'ept','nan'),"\t",
            print getattr(report,'speed','nan'),"\t",
            print getattr(report,'climb','nan'),"\t"
```

```

print getattr(report,'epv','nan'),"\t",
print getattr(report,'ept','nan'),"\t",
print getattr(report,'speed','nan'),"\t",
print getattr(report,'climb','nan'),"\t"
lat= getattr(report,'lat',0.0)
long= getattr(report,'lon',0.0)
alt= getattr(report,'alt','nan')
speed= getattr(report,'speed','nan')

sArduino = serial_Arduino.readline()
datos = sArduino.rstrip('\n')

ph=sArduino[0:5]
tem=sArduino[5:11]
oxi= 6.45
print('pH: ',ph)
print('Temp: ',tem)
print('OD: ',oxi)

params = urllib.urlencode({'field1': lat, 'field2': long,'field3': alt,'field4': speed, 'field5': ph,'field6': tem,'field7':oxi })
headers = {"Content-type": "application/x-www-form-urlencoded","Accept" : "text/plain"}
conn = urllib.HTTPConnection("api.thingspeak.com:80")

try:
    conn.request("POST", "/update", params, headers)
    response = conn.getresponse()
    print response.status, response.reason
    data = response.read()
    conn.close()
except:
    print "connection failed"

time.sleep(1)

except (KeyboardInterrupt, SystemExit): #when you press ctrl+c
    print "Done.\nExiting."
|

```

ANEXO Ñ: Archivo generado por la plataforma ThingSpeak

CSV Viewer Free

## Spreadsheet

A1

	A	B	C	D	E	F	G	H	I
1	created_at	entry_id	field1	field2	field3	field4	field5	field6	
2	2019-03-30 21:38:37 UTC	431	-3.322556667	-79.7701525	6.45	0.033	-3.47	23.31	
3	2019-03-30 21:38:52 UTC	432	-3.322557413	-79.770152454	6.45	0.016	-2.79	23.31	
4	2019-03-30 21:39:07 UTC	433	-3.322557547	-79.770152592	6.45	0.035	-2.95	23.31	
5	2019-03-30 21:39:22 UTC	434	-3.322558431	-79.770152274	6.45	0.039	-2.73	23.31	
6	2019-03-30 21:39:38 UTC	435	-3.322557333	-79.770152833	6.45	0.029	-2.60	23.31	
7	2019-03-30 21:39:53 UTC	436	-3.322557833	-79.7701525	6.45	0.023	-2.52	23.31	
8	2019-03-30 21:40:08 UTC	437	-3.322558753	-79.77015202	6.45	0.018	-2.57	23.31	
9	2019-03-30 21:40:23 UTC	438	-3.322559681	-79.770151946	6.45	0.01	-2.49	23.31	
10	2019-03-30 21:40:39 UTC	439	-3.322558797	-79.770152169	6.45	0.047	-2.41	23.31	
11	2019-03-30 21:40:54 UTC	440	-3.322558667	-79.770152667	6.45	0.022	-2.65	23.31	
12	2019-03-30 21:41:09 UTC	441	-3.322559298	-79.770152839	6.45	0.028	-2.43	23.31	
13	2019-03-30 21:41:25 UTC	442	-3.322559066	-79.770152992	6.45	0.042	-2.65	23.31	
14	2019-03-30 21:41:40 UTC	443	-3.322559502	-79.770152481	6.45	0.028	-2.27	23.31	
15	2019-03-30 21:41:55 UTC	444	-3.3225615	-79.770151333	6.45	0.015	-2.11	23.31	
16	2019-03-30 21:42:11 UTC	445	-3.32256398	-79.770150587	6.45	0.035	-2.00	23.31	
17	2019-03-30 21:42:27 UTC	446	-3.322564041	-79.770151966	6.45	0.02	-1.84	23.31	
18	2019-03-30 21:42:42 UTC	447	-3.322563996	-79.770152205	6.45	0.044	-2.25	23.31	
19	2019-03-30 21:42:57 UTC	448	-3.3225635	-79.770152	6.45	0.037	-2.25	23.31	
20	2019-03-30 21:43:14 UTC	449	-3.322563833	-79.770151833	6.45	0.027	-2.27	23.31	
21	2019-03-30 21:43:29 UTC	450	-3.322563164	-79.770152446	6.45	0.051	-2.16	23.31	
22	2019-03-30 21:43:44 UTC	451	-3.32256409	-79.770152553	6.45	0.02	-2.16	23.31	
23	2019-03-30 21:44:06 UTC	452	-3.322564113	-79.770152755	6.45	0.024	-2.16	23.31	
24	2019-03-30 21:44:21 UTC	453	-3.322564833	-79.770152	6.45	0.041	-2.14	23.31	
25	2019-03-30 21:44:37 UTC	454	-3.322566102	-79.770151467	6.45	0.009	-1.95	23.31	
26	2019-03-30 21:44:52 UTC	455	-3.322567315	-79.770151162	6.45	0.025	-2.03	23.31	
27	2019-03-30 21:45:07 UTC	456	-3.322567599	-79.770151396	6.45	0.028	-2.08	23.31	
28	2019-03-30 21:45:24 UTC	457	-3.322567833	-79.770151667	6.45	0.01	-1.78	23.31	
29	2019-03-30 21:45:39 UTC	458	-3.322568	-79.770151667	6.45	0.002	-1.92	23.31	
30	2019-03-30 21:45:54 UTC	459	-3.322567434	-79.77015336	6.45	0.031	-2.00	23.31	
31	2019-03-30 21:46:10 UTC	460	-3.322565198	-79.770153843	6.45	0.071	-2.06	23.31	
32	2019-03-30 21:46:25 UTC	461	-3.322564144	-79.770154212	6.45	0.054	-2.00	23.25	

Sheet1

## ANEXO O: Historial de variables en plataforma ThingSpeak

