



	<b>UNIVERSIDAD SURCOLOMBIANA GESTIÓN SERVICIOS BIBLIOTECARIOS</b>					   		
	<b>CARTA DE AUTORIZACIÓN</b>					SC-7304-1      SA-CERE 597520      OS-CER 597555		
<b>CÓDIGO</b>	<b>AP-BIB-FO-06</b>	<b>VERSIÓN</b>	<b>1</b>	<b>VIGENCIA</b>	<b>2014</b>	<b>PÁGINA</b>	<b>1 de 2</b>	

Neiva, 01 de febrero de 2022

Señores

CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN

UNIVERSIDAD SURCOLOMBIANA

Ciudad

El (Los) suscrito(s):

Albert Plaza Sanchez, con C.C. No. 1077866932,

Leyder Badillo Valdes, con C.C. No. 1079509544,

Autor(es) de la tesis y/o trabajo de grado titulado **DISEÑO E IMPLEMENTACIÓN DE PROTOTIPO PARA MONITOREO Y RECOLECCIÓN DE DATOS APÍCOLAS MEDIANTE EL USO DE COMUNICACIÓN LPWAN** presentado y aprobado en el año 2021 como requisito para optar al título de pregrado;

Autorizo (amos) al CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN de la Universidad Surcolombiana para que, con fines académicos, muestre al país y el exterior la producción intelectual de la Universidad Surcolombiana, a través de la visibilidad de su contenido de la siguiente manera:

- Los usuarios puedan consultar el contenido de este trabajo de grado en los sitios web que administra la Universidad, en bases de datos, repositorio digital, catálogos y en otros sitios web, redes y sistemas de información nacionales e internacionales “open access” y en las redes de información con las cuales tenga convenio la Institución.
- Permita la consulta, la reproducción y préstamo a los usuarios interesados en el contenido de este trabajo, para todos los usos que tengan finalidad académica, ya sea en formato Cd-Rom o digital desde internet, intranet, etc., y en general para cualquier formato conocido o por conocer, dentro de los términos establecidos en la Ley 23 de 1982, Ley 44 de 1993, Decisión Andina 351 de 1993, Decreto 460 de 1995 y demás normas generales sobre la materia.
- Continúo conservando los correspondientes derechos sin modificación o restricción alguna; puesto que, de acuerdo con la legislación colombiana aplicable, el presente es un acuerdo jurídico que en ningún caso conlleva la enajenación del derecho de autor y sus conexos.

De conformidad con lo establecido en el artículo 30 de la Ley 23 de 1982 y el artículo 11 de la Decisión Andina 351 de 1993, “Los derechos morales sobre el trabajo son propiedad de los autores”, los cuales son irrenunciables, imprescriptibles, inembargables e inalienables.

Vigilada Mineducación

La versión vigente y controlada de este documento, solo podrá ser consultada a través del sitio web Institucional [www.usco.edu.co](http://www.usco.edu.co), link Sistema Gestión de Calidad. La copia o impresión diferente a la publicada, será considerada como documento no controlado y su uso indebido no es de responsabilidad de la Universidad Surcolombiana.



UNIVERSIDAD SURCOLOMBIANA  
GESTIÓN SERVICIOS BIBLIOTECARIOS



CARTA DE AUTORIZACIÓN

CÓDIGO

AP-BIB-FO-06

VERSIÓN

1

VIGENCIA

2014

PÁGINA

2 de 2

EL AUTOR/ESTUDIANTE:

Firma:

EL AUTOR/ESTUDIANTE:

Firma:

EL AUTOR/ESTUDIANTE:

Firma:

\_\_\_\_\_






EL AUTOR/ESTUDIANTE:

Firma:

\_\_\_\_\_

Vigilada Mineducación

La versión vigente y controlada de este documento, solo podrá ser consultada a través del sitio web Institucional [www.usco.edu.co](http://www.usco.edu.co), link Sistema Gestión de Calidad. La copia o impresión diferente a la publicada, será considerada como documento no controlado y su uso indebido no es de responsabilidad de la Universidad Surcolombiana.

	UNIVERSIDAD SURCOLOMBIANA GESTIÓN SERVICIOS BIBLIOTECARIOS					   	
	DESCRIPCIÓN DE LA TESIS Y/O TRABAJOS DE GRADO						
CÓDIGO	AP-BIB-FO-07	VERSIÓN	1	VIGENCIA	2014	PÁGINA	1 de 3

**TÍTULO COMPLETO DEL TRABAJO:** DISEÑO E IMPLEMENTACIÓN DE PROTOTIPO PARA MONITOREO Y RECOLECCIÓN DE DATOS APÍCOLAS MEDIANTE EL USO DE COMUNICACIÓN LPWAN

**AUTOR O AUTORES:**

Primero y Segundo Apellido	Primero y Segundo Nombre
Plaza Sanchez	Albert
Badillo Valdes	Leyder

**DIRECTOR Y CODIRECTOR TESIS:**

Primero y Segundo Apellido	Primero y Segundo Nombre
Molina Mosquera	Johan Julián

**JURADOS:**

Primero y Segundo Apellido	Primero y Segundo Nombre
Bravo Obando	Martin Diomedes
Quintero Polania	Jesús David

**PARA OPTAR AL TÍTULO DE:** Especialista en Gerencia Tributaria

**FACULTAD:** Economía y Administración

**PROGRAMA O POSGRADO:** Especialización en Gerencia Tributaria

**CIUDAD:** Neiva      **AÑO DE PRESENTACIÓN:** 2021      **NÚMERO DE PÁGINAS:** 53






**TIPO DE ILUSTRACIONES** (Marcar con una X):

Diagramas\_\_\_ Fotografías\_\_\_ Grabaciones en discos\_\_\_ Ilustraciones en general\_\_\_ Grabados\_\_\_  
Láminas\_\_\_ Litografías\_\_\_ Mapas\_\_\_ Música impresa\_\_\_ Planos\_\_\_ Retratos\_\_\_ Sin ilustraciones\_\_\_ Tablas  
o Cuadros \_\_\_

**SOFTWARE** requerido y/o especializado para la lectura del documento:

Vigilada Mineducación

La versión vigente y controlada de este documento, solo podrá ser consultada a través del sitio web Institucional [www.usco.edu.co](http://www.usco.edu.co), link Sistema Gestión de Calidad. La copia o impresión diferente a la publicada, será considerada como documento no controlado y su uso indebido no es de responsabilidad de la Universidad Surcolombiana.

	<b>UNIVERSIDAD SURCOLOMBIANA</b> <b>GESTIÓN SERVICIOS BIBLIOTECARIOS</b>					   	
	<b>DESCRIPCIÓN DE LA TESIS Y/O TRABAJOS DE GRADO</b>						
<b>CÓDIGO</b>	<b>AP-BIB-FO-07</b>	<b>VERSIÓN</b>	<b>1</b>	<b>VIGENCIA</b>	<b>2014</b>	<b>PÁGINA</b>	<b>2 de 3</b>

**MATERIAL ANEXO:**

**PREMIO O DISTINCIÓN** *(En caso de ser LAUREADAS o Meritoria):*






**PALABRAS CLAVES EN ESPAÑOL E INGLÉS:**

<u>Español</u>	<u>Inglés</u>	<u>Español</u>	<u>Inglés</u>
1. Aplicativo web	Web application	6. Monitoreo	Monitoring
2. Base de datos	Database	7. Microcontrolador	Microcontroller
3. LPWAN	LPWAN	8. Medición	Measurement
4. Colmena	Hive	9.	
5. IoT	IoT	10.	

**RESUMEN DEL CONTENIDO:** (Máximo 250 palabras)

El Huila es un departamento que goza con gran variedad de flora y fauna además de esto se ha visto afectado en cambios climáticos y destrucción de ella debido a las represas que has sido construidas por el hombre, las abejas con indispensables para la restauración de la naturaleza.

El presente documento tiene el objetivo de exponer generalmente un prototipo para monitoreo de datos apícolas usando comunicación de baja potencia para la realización de un proyecto aplicado de ingeniería hacia el agro. Donde básicamente se cubre la necesidad del apicultor de viajar constantemente para revisar físicamente el estado de la colmena. Por esto nace también como una forma de estudiar a la población de abejas y su miel en una colmena por medio del censado de variables como temperatura, humedad y peso de las abejas y miel. El proyecto incluye la descripción de la instrumentación que define los procesos de lectura de todos los sensores desde el microcontrolador, la organización de la estructura para el proceso de censado, el envío de información, la codificación de todos los datos obtenidos para ser enviados a través del sistema de comunicación de baja potencia a una base de datos online donde es posible revisar en cualquier dispositivo con acceso a internet.

	UNIVERSIDAD SURCOLOMBIANA GESTIÓN SERVICIOS BIBLIOTECARIOS					   	
	DESCRIPCIÓN DE LA TESIS Y/O TRABAJOS DE GRADO						
CÓDIGO	AP-BIB-FO-07	VERSIÓN	1	VIGENCIA	2014	PÁGINA	3 de 3

**ABSTRACT:** (Máximo 250 palabras)

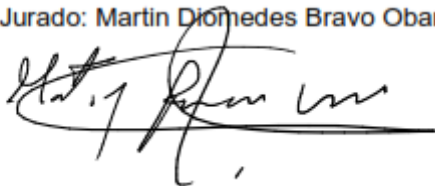
Huila is a department that enjoys a great variety of flora and fauna, in addition to this, it has been affected in climatic changes and destruction of it due to dams that have been built by man, bees are essential for the restoration of nature.

The present document has the objective of generally exposing a prototype for monitoring beekeeping data using low-power communication to carry out an applied agricultural engineering project. Where basically the need for the beekeeper to travel constantly to physically check the state of the hive is covered. For this reason, it was also born as a way of studying the population of bees and their honey in a hive by means of the census of variables such as temperature, humidity and weight of the bees and honey. The project includes the description of the instrumentation that defines the reading processes of all the sensors from the microcontroller, the organization of the structure for the census process, the sending of information, the coding of all the data obtained to be sent through from the low-power communication system to an online database where it is possible to check on any device with internet access.

APROBACION DE LA TESIS

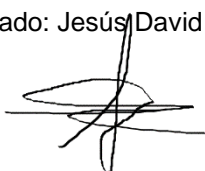
Nombre Jurado: Martin Diomedes Bravo Obando

Firma:



Nombre Jurado: Jesús David Quintero Polania

Firma:



Vigilada Mineducación

La versión vigente y controlada de este documento, solo podrá ser consultada a través del sitio web Institucional [www.usco.edu.co](http://www.usco.edu.co), link Sistema Gestión de Calidad. La copia o impresión diferente a la publicada, será considerada como documento no controlado y su uso indebido no es de responsabilidad de la Universidad Surcolombiana.

**DISEÑO E IMPLEMENTACIÓN DE PROTOTIPO PARA MONITOREO Y  
RECOLECCIÓN DE DATOS APÍCOLAS MEDIANTE EL USO DE  
COMUNICACIÓN LPWAN**

**ALBERT PLAZA SÁNCHEZ  
LEYDER BADILLO VALDES**

**UNIVERSIDAD SURCOLOMBIANA  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE ELECTRÓNICA  
NEIVA HUILA**

**2021**  
**DISEÑO E IMPLEMENTACIÓN DE PROTOTIPO PARA MONITOREO Y  
RECOLECCIÓN DE DATOS APÍCOLAS MEDIANTE EL USO DE  
COMUNICACIÓN LPWAN**

**ALBERT PLAZA SÁNCHEZ  
LEYDER BADILLO VALDES**

**Trabajo de grado presentado como requisito para optar al Título de Ingeniero  
Electrónico.**

**Director**  
**JOHAN JULIÁN MOLINA MOSQUERA**  
**MSC. TELECOMUNICACIONES MÓVILES**

**UNIVERSIDAD SURCOLOMBIANA**  
**FACULTAD DE INGENIERÍA**

**PROGRAMA DE INGENIERÍA DE ELECTRÓNICA  
NEIVA HUILA  
2021**

Nota de aceptación

---

---

---

---

---

---

---

\_\_\_\_\_  
Firma del director

\_\_\_\_\_  
Firma del jurado

\_\_\_\_\_  
Firma del jurado



Neiva, Octubre de 2021

## DEDICATORIA

*Dedico este proyecto de grado a todas las personas que siempre fueron un incondicional apoyo, a mi hermana Leidy Catalina por brindarme una ayuda para iniciar con la carrera y siempre estar presta en apoyarme, a mi madre Flor Angela por motivarme para aprender más y con su amor de madre estar al tanto de cada necesidad de mi vida, a mi padre Alberto por estar pendientes de mí y hacer lo mejor para cubrir mis necesidades, a mi tía Azucena quien me acogió y brindo su amor y guía como a un hijo más, a mis primos que no les importo incomodarse para brindarme parte de su espacio para iniciar por este camino universitario prestos a ayudar en todo momento. A mi amigo Jesús David Bonelo y su familia, a Leyder mi compañero de tesis quien entiende los retos que hemos tenido, a Sebastián Pastrana quien es un amigo con quien se puede contar en todo momento que con su sabiduría y enseñanzas despejo muchas dudas, y especialmente a mi amada Natalia y mi Hijo Samuel quienes son el pilar de mi vida siempre manteniéndome firme a culminar mi carrera*

**Albert Plaza Sánchez**

Agradezco a Dios por la permitirme culminar una de mis metas junto a las personas que quiero como lo son mis padres Maria Oveida Valdes Idrobo y Amaury Badillo Mosquera a mis hermanos Amaury Badillo Valdes y Darwin Badillo Erazo, también debo tener en cuenta a todos aquellos que estuvieron en mi camino como lo son docentes, administrativos y compañero de carrera en la Universidad Surcolombiana de igual forma quiero resaltar la presencia de mis compañeros más cercanos como lo es Albert Plaza, Juan Sebastián Pastrana, David Arango y Juan Diego Lozada que de una u otra forma me sirvieron para no quedar en el camino y seguir avanzado hasta lograr obtener el título de pregrado como ingeniero.

**Leyder Badillo Valdés**

## AGRADECIMIENTOS

Principalmente un agradecimiento especial a Dios por regalarnos la vida y la sabiduría diariamente, a nuestras familias por todo su amor e incondicionalidad, gracias a la Universidad Surcolombiana y a la facultad de ingeniería por tener todos los docentes más preparados para ayudar a subir cada peldaño de sabiduría, a los profesores de ingeniería electrónica los cuales hicieron parte muy importante en nuestra formación como personas y como futuros ingenieros electrónicos. A nuestro director Julián Molina por su apoyo y paciencia en el transcurso del proyecto, a nuestros jurados de proyecto de grado Jesús David Quintero y Martin Diomedes Obando por su conocimiento y recomendaciones. A Don Alberto Plaza por darnos la idea y brindarnos su conocimiento y recursos para desarrollar correctamente este proyecto. A todos los compañeros y personas que son ayudaron a realizar ese sueño. ¡Mil gracias!

*"El único modo de hacer un gran trabajo es amar lo que haces!"*  
**Steve Jobs**

*"La inteligencia consiste no sólo en el conocimiento, sino también en la destreza de aplicar los conocimientos en la práctica".*

**Aristóteles**  
**TABLA DE CONTENIDO**  
**Pág.**

## Contenido

<b>CONTENIDO</b>	<b>6</b>
<b>GLOSARIO</b>	<b>10</b>
<b>ACRÓNIMOS</b>	<b>10</b>
<b>DEFINICIONES</b>	<b>11</b>
<b>RESUMEN</b>	<b>13</b>
<b>ABSTRACT</b>	<b>14</b>
<b>1. INTRODUCCIÓN</b>	<b>15</b>
<b>2. JUSTIFICACIÓN</b>	<b>16</b>
<b>3. OBJETIVOS</b>	<b>17</b>
<b>3.1 OBJETIVO GENERAL</b>	<b>17</b>
<b>3.2 OBJETIVOS ESPECÍFICOS</b>	<b>17</b>
<b>4. MARCO TEÓRICO</b>	<b>18</b>
<b>4.1 ANTECEDENTES</b>	<b>18</b>
<b>4.2 CONTEXTUALIZACIÓN DEL PROBLEMA</b>	<b>20</b>
4.2.1 ABEJAS MELÍFERAS	20
4.2.2 CICLO DE VIDA	21
4.2.3 SEGURIDAD Y MANEJO DE LA COLMENA	23
<b>4.3. COMUNICACIONES INALÁMBRICAS</b>	<b>24</b>
<b>4.4 MICROCONTROLADOR</b>	<b>24</b>
4.4.1 ESP32	25
4.4.2 ESP8266	26
4.4.3 Modulo HX711	27
<b>5. VARIABLES A MEDIR</b>	<b>27</b>
<b>5.1 TEMPERATURA</b>	<b>27</b>
<b>5.2 HUMEDAD RELATIVA</b>	<b>28</b>
<b>5.3 MASA</b>	<b>28</b>
<b>5.4 FLUJO</b>	<b>28</b>
<b>6. SENSORES</b>	<b>28</b>
<b>6.1 SENSOR DE TEMPERATURA</b>	<b>28</b>
<b>6.2 SENSOR DE HUMEDAD</b>	<b>28</b>
<b>6.3 SENSOR FOTOELÉCTRICO</b>	<b>28</b>
6.3.1 Modelo reflectivo	29
6.3.2. Modelo de barrera	29
6.3.3. Modelo retro reflectivo	29
<b>7. SENSORES UTILIZADOS PARA LAS VARIABLES MEDIDAS POR EL PROTOTIPO</b>	<b>30</b>
<b>7.1 TEMPERATURA, HUMEDAD, PESO, FLUJO</b>	<b>30</b>
7.1.1 Sensor DHT11	30
7.1.2 Celda de carga 50kg	31
7.1.3 Sensor de proximidad	32
<b>8. DISEÑO Y DESARROLLO SOFTWARE</b>	<b>33</b>
<b>8.1 PROGRAMACIÓN Y CALIBRACIÓN DE CIRCUITOS INTEGRADOS Y SENSORICA</b>	<b>33</b>
8.1.1 Circuitos Integrados	33
8.1.2 Sensorica	35
8.1.2.1 Sensor Temperatura y Humedad DHT11	35
8.1.2.2 Sensor de Flujo FC-51	37
8.1.2.3 Sensor de Peso (Celda de carga de 50Kg)	39
<b>8.2 COMUNICACIÓN CON LA API THINGSPEAK Y BASE DE DATOS EN LA NUBE</b>	<b>42</b>

8.2.1 ThingSpeak.....	42
8.2.2 Canal en ThingSpeak .....	42
8.3 ESQUEMÁTICO DEL FUNCIONAMIENTO COMPLETO DEL PROTOTIPO .....	52
9. DISEÑO Y DESARROLLO DEL HARDWARE.....	52
9.1 DISEÑO DE LA TARJETA DE CIRCUITO IMPRESO - PRINTED CIRCUIT BOARD (PCB) .....	52
9.2 MODULO TTGO .....	53
9.3 MODULO ESP8266.....	57
9.4 MODULO LECTOR MICRO SD .....	58
9.5 ETAPA DE ALIMENTACIÓN.....	59
9.5.1 Conversión de la energía solar .....	59
9.5.2 Diseño etapa de alimentación.....	60
9.5.2.1 Paneles fotovoltaicos .....	62
9.5.2.2 Batería .....	63
9.5.2.3 Regulador LM2596 .....	63
9.6 UBICACIÓN ESPACIAL DE LA COLMENA .....	64
10. RESULTADOS Y ANÁLISIS .....	65
11. BIBLIOGRAFÍA .....	77
12. ANEXOS.....	80
12.1 COSTOS.....	80
12.2 CÓDIGO PROGRAMADO EN EL ESP32 (COLMENA) .....	81
12.3 CÓDIGO PROGRAMADO EN EL ESP32 (CASA).....	86

## LISTA DE TABLAS

	<b>Pág.</b>
TABLA 1 FUNCIONES QUE PUEDEN SER UTILIZADAS EN EL HX711 .....	38
TABLA 2 - TIPO DE DOCUMENTACIÓN ARDUINO .....	50

TABLA 3 - GANANCIA EN DBI PARA LAS FRECUENCIAS DE OPERACIÓN DE LA ANTENA EN EL PLANO H Y PLANO E1 .....	54
TABLA 4 - CONSUMO DE DISPOSITIVOS EMPLEADOS .....	59
TABLA 5 - DATOS OBTENIDOS EN LA SD .....	65
TABLA 6 - DATOS OBTENIDOS EN LA SD DESPUÉS DE LA LIMPIEZA DE DATOS .....	69

## LISTA DE FIGURAS

	<b>Pág.</b>
FIGURA 1 - LAS CASTAS DE LAS ABEJAS MELÍFERAS: (A) REINA, (B) OBRERA, (C) ZÁNGANO .....	21
FIGURA 2- CICLO DE DESARROLLO DIARIO DE LAS DOS CASTAS FEMENINAS Y EL ZÁNGANO .....	22
FIGURA 3 - ESQUEMA DE LA ARQUITECTURA VON NEUMANN .....	24
FIGURA 4 - ESQUEMA DE LA ARQUITECTURA HARVARD .....	24
FIGURA 5 - CONFIGURACIÓN DE PINES ESP32 .....	25
FIGURA 6 - CONFIGURACIÓN DE PINES Esp8266 .....	26
FIGURA 7 - CONFIGURACIÓN DE PINES HX711 .....	26
FIGURA 8 - MODELO REFLECTIVO .....	28
FIGURA 9 - MODELO DE BARRERA .....	28
FIGURA 10 - MODELO RETRO REFLECTIVO .....	29
FIGURA 11 - SENSOR DHT11 .....	30
FIGURA 12 - CELDA DE CARGA .....	31
FIGURA 13 - SENSOR DE PROXIMIDAD INFRARROJO .....	31
FIGURA 14 - DIAGRAMA SISTEMA DE COMUNICACIÓN DEL PROYECTO .....	32
FIGURA 15 - ESQUEMA DE SEÑALES DEL MICROCONTROLADOR CON EN EL SENSOR DHT11 .....	34
FIGURA 16 - ESQUEMA FUNCIONAL DE SEÑALES EN EL SENSOR DHT11 .....	35
FIGURA 17 - ESQUEMA FUNCIONAL DE SEÑALES TRANSMITIDAS POR EL SENSOR DHT11 .....	35
FIGURA 18 - ESQUEMA DE APLICACIÓN TÍPICO PERIFÉRICO DEL SENSOR DHT11 .....	36
FIGURA 19 - RESPUESTA DE UN AMPLIFICADOR COMPARADOR BÁSICO .....	37
FIGURA 20 - DIAGRAMA CONEXIONES ELÉCTRICAS PARA EL SENSOR FC-51 .....	37
FIGURA 21 - DIAGRAMA DE BLOQUES DEL CIRCUITO HX711 .....	38
FIGURA 22 - SEÑALES ENVIADAS POR EL HX711 A LA PLACA DE DESARROLLO .....	40
FIGURA 23 - INTERFAZ DE CREACIÓN DE USUARIO EN THINGSPEAK .....	42
FIGURA 24 - INTERFAZ DE USUARIO CON CONTROL DE LA PLATAFORMA .....	43
FIGURA 25 - INTERFAZ DE CREACIÓN DE CANAL .....	43
FIGURA 26 - INTERFAZ DE VISUALIZACIÓN DEL CANAL.....	45
FIGURA 27 - INTERFAZ OPCIONES DE CANAL .....	46
FIGURA 28 - CÓDIGO DE EJEMPLO COMUNICACIÓN CON THINGSPEAK DESDE TARJETA DE DESARROLLO ESP8266 .....	49
FIGURA 29 - ESQUEMÁTICO PROTOTIPO PARA MONITOREO .....	51
FIGURA 30 - CIRCUITO DE DISTRIBUCIÓN DE ALIMENTACIÓN .....	51
FIGURA 31 - MODULO TTGO LORA32 UTILIZADO .....	52
FIGURA 32 - ANTENA PARA LA TRANSMISIÓN DEL TTGO LORA32 .....	53
FIGURA 33 - PLANO DE RADIACION TTGO LORA32 .....	53
FIGURA 34 - PATRÓN DE RADIACIÓN PARA LA ANTENA LORA32 EN EL PLANO H .....	55
FIGURA 35 - PATRÓN DE RADIACIÓN PARA LA ANTENA LORA32 EN EL PLANO E1 .....	55
FIGURA 36 - MONTAJE COMPLETO PROTOTIPO DE MONITOREO .....	57
FIGURA 37 - MODULO LECTOR MICRO SD .....	57
FIGURA 38 - ESQUEMA ETAPA DE ALIMENTACIÓN .....	58
FIGURA 39 - ESQUEMA DE CORRIENTE .....	60
FIGURA 40 - PANEL SOLAR UTILIZADO .....	61

FIGURA 41 - REFERENCIA BATERIA .....	62
FIGURA 42 - REGULADOR UTILIZADO .....	62
FIGURA 43 - LOCALIZACIÓN GEOGRÁFICA COLMENA .....	63
FIGURA 44 - MAPA SOLAR .....	64
FIGURA 45 - MONTAJE EN COMPLETO Y FUNCIONAMIENTO. ....	64
FIGURA 46 - GRAFICO DE TEMPERATURA DENTRO DE LA COLMENA .....	66
FIGURA 47 - GRÁFICO DE HUMEDAD EN LA COLMENA .....	66
FIGURA 48 - GRÁFICO DE FLUJO RELATIVO EN LA COLMENA .....	66
FIGURA 49 - GRÁFICO DE PESO DE LA COLMENA .....	67
FIGURA 50 - GRÁFICO DE TEMPERATURA EXTERNA .....	67
FIGURA 51 - GRÁFICO DE HUMEDAD EXTERNA .....	67
FIGURA 52 - GRÁFICO RSSI .....	68
FIGURA 53 - GRÁFICO CONTADOR AUTOMÁTICO PARA GUÍA .....	68
FIGURA 54 - GRÁFICO DE TEMPERATURA INTERNA EN LA COLMENA DESPUÉS DE LA LIMPIEZA DE DATOS .....	69
FIGURA 55 - GRÁFICO DE HUMEDAD INTERNA EN LA COLMENA DESPUÉS DE LA LIMPIEZA DE DATOS .....	69
FIGURA 56 - GRÁFICO DE FLUJO RELATIVO EN LA COLMENA DESPUÉS DE LA LIMPIEZA DE DATOS .....	70
FIGURA 57 - GRÁFICO DE PESO EN LA COLMENA DESPUÉS DE LA LIMPIEZA DE DATOS .....	70
FIGURA 58 - GRÁFICO DE TEMPERATURA EXTERNA DESPUÉS DE LA LIMPIEZA DE DATOS .....	70
FIGURA 59 - GRÁFICO DE LA HUMEDAD EXTERNA DESPUÉS DE LA LIMPIEZA DE DATOS .....	71
FIGURA 60 - GRÁFICO DE RSSI DESPUÉS DE LA LIMPIEZA DE DATOS .....	71
FIGURA 61 - GRÁFICO DE CONTADOR AUTOMÁTICO PARA GUÍA DESPUÉS DE LA LIMPIEZA DE DATOS .....	71
FIGURA 62 - GRÁFICO DE TEMPERATURA SD Y TEMPERATURA TS .....	72
FIGURA 63 - GRÁFICO DE HUMEDAD SD Y HUMEDAD TS .....	73

## **GLOSARIO ACRÓNIMOS**

**GB:** (Gigabyte) Es una unidad de almacenamiento de información.

**GHz:** (Gigahertz) Es una unidad de frecuencia.

**GO/DONE:** (A/D Conversion Status bit) Bit de estado de conversión A/D.

**GPU:** (Graphics Processing Unit) Unidad de procesamiento gráfico.

**HD:** (High Definition) Alta definición.

**IP:** (Internet Protocol) Es un número que identifica, de manera lógica y jerárquica, a una Interfaz en red.

**IoT:** (Internet of Things) Internet de las cosas.

**IPv4:** (Internet Protocol version 4) El Protocolo de Internet versión 4. **LSB:**

(Least significant bit) Es la posición del bit en un número entero binario. mAh:  
Miliamperio-hora.

**MCU:** Microcontrolador.

**MEMS:** (Microelectromechanical Systems) Sistemas Microelectromecánicos.

**MISO:** (Master Input Slave Output) Salida de datos del Esclavo y entrada al Maestro.

**MOSI:** (Master Output Slave Input) Salida de datos del Maestro y entrada al Esclavo.

**MHz:** (Megahertz) Es una unidad de medida de la frecuencia.

**SYNC:** (synchronize) Sincronizar.

## DEFINICIONES

**Microcontrolador:** Un microcontrolador es un dispositivo electrónico capaz de llevar a cabo procesos lógicos. Estos procesos o acciones son programados en lenguaje ensamblador por el usuario, y son introducidos en este a través de un programador. Esto suena un poco complicado, pero sólo es un resumen de 3 líneas. A lo largo de este tutorial veremos algunas de las reglas y trucos de este lenguaje complicado por su sencillez<sup>1</sup>

**Sistema Electrónico de Comunicaciones:** Es transferir información de un lugar a otro. Por consiguiente, se puede decir que las comunicaciones electrónicas son la transmisión, recepción y procesamiento de información entre dos o más lugares, mediante circuitos electrónicos. La fuente original de información puede estar en forma analógica (continua), como por ejemplo la voz humana o la música, o en forma digital (discreta), como por ejemplo los números codificados binariamente o los códigos alfanuméricos.<sup>2</sup>

**Telemetría:** Procedente de las palabras tele y metron, la considera como una tecnología que permite la medición de las características de un objeto (como la temperatura ambiente, o la presión existente en una tubería) y transportar los resultados a una estación distante donde son desplegados, guardados y analizados, señalando, además que la gran utilidad de la Telemetría es la de la obtención de datos distribuidos.<sup>3</sup>

**Teledetección:** La teledetección o percepción remota es una disciplina científica que integra un amplio conjunto de conocimientos y tecnologías empleadas en la observación, análisis e interpretación de fenómenos terrestres y planetarios. La etimología de la palabra refiere a la adquisición de información sobre un objeto o fenómeno sin hacer contacto físico con el mismo, por tanto, contrasta la observación directa o in-situ.<sup>4</sup>

**Sensor:** Los sensores son aparatos que son capaces de transformar magnitudes físicas o químicas, llamadas variables de instrumentación, en magnitudes eléctricas. Las variables de instrumentación dependen del tipo de sensor y pueden ser, por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, pH, etc.<sup>5</sup>

---

<sup>1</sup> Paul Aguayo S., paguayo@olimex.cl. " INTRODUCCIÓN AL MICROCONTROLADOR", 2014. Paper.

<sup>2</sup> TOMASI, Wayne. Sistemas de comunicaciones electrónicas- 4 Edición. Pearson Educacion, 2015.

<sup>3</sup> DAZA, H., VEGA, O. Telemetría aplicada al monitoreo y control de la válvula La Maria PK165, 2014, Facultad de Ciencias e Ingeniería, Universidad de Manizales.

<sup>4</sup> Wikipedia teledetección. [en línea]. Consultado: ](03 de Marzo de 2021) Disponible en: <https://es.wikipedia.org/wiki/Teledetecci%C3%B3n>

<sup>5</sup> Mecafenix, I. [en línea]. Consultado:(11 de Marzo de 2021). Tipos de sensores y sus características. Obtenido de <https://www.ingmecafenix.com/automatizacion/sensores/>



**Colmenas Langstroth:** desde su primera construcción en 1852, la primera colmena movilista ha sufrido algunos cambios de medidas y técnicos, estos segundos para poder admitir algunos accesorios posteriores. Es de crecimiento vertical por el uso de alzas, eso quiere decir, que el crecimiento de la colmena en la temporada apícola, es hacia arriba. Su característica principal es la idéntica medida entre los cajones para la cría y los de producción.<sup>6</sup>

**Apicultura en Colombia:** La apicultura colombiana tiene sus inicios durante la conquista española cuando fue introducida al continente la abeja apis melífera, especie que debido a su alta producción de miel desplazó a las abejas nativas de forma racional por los indígenas para la obtención de miel, cera y otros productos. Posterior a la introducción de la abeja Abeja apis melífera, la apicultura se desarrolló de forma rústica por un amplio periodo hasta la segunda mitad del siglo XIX, periodo al cual se le atribuyen los primeros esfuerzos de modernización de la actividad, aunque de forma incipiente.<sup>7</sup>

---

<sup>6</sup> Jordi Besora Magem “INFORME TÉCNICO PARA LA CONSTRUCCIÓN DE UNA COLMENA Y PORTANÚCLEO TIPO LANGSTROTH” (2017), Informe tecnico

<sup>7</sup> Revista bolivariana, Bogota “Aportes para el análisis del sector Apícola Colombiano”,2013

## **RESUMEN**

### **TÍTULO:**

DISEÑO E IMPLEMENTACIÓN DE PROTOTIPO PARA MONITOREO Y RECOLECCIÓN DE DATOS APÍCOLAS MEDIANTE EL USO DE COMUNICACIÓN LPWAN.

### **AUTORES:**

ALBERT PLAZA SÁNCHEZ  
LEYDER BADILLO VALDES

### **PALABRAS CLAVES:**

Aplicativo web, base de datos, LPWAN, colmena, IoT, monitoreo, microcontrolador, medición, API.

### **DESCRIPCIÓN:**

El presente documento tiene el objetivo de exponer un prototipo para monitoreo de datos apícolas usando comunicación de baja potencia (LPWAN) para la realización de un proyecto aplicado de ingeniería hacia el agro. Donde básicamente se cubre la necesidad del apicultor de viajar constantemente para revisar físicamente el estado de la colmena. Por esto nace también como una forma de estudiar a la población de abejas y su miel en una colmena por medio del sensado de variables como temperatura, humedad y peso de las abejas y miel. El proyecto incluye la descripción de la instrumentación que define los procesos de lectura de todos los sensores desde el microcontrolador, la organización de la estructura para el proceso de sensado, el envío de información, la codificación de todos los datos obtenidos para ser enviados a través del sistema de comunicación de baja potencia (LPWAN) a una base de datos online donde es posible revisar en cualquier dispositivo con acceso a internet.

## **ABSTRACT**

### **TITLE:**

DESIGN AND IMPLEMENTATION OF PROTOTYPE FOR MONITORING AND COLLECTION OF BEEKEEPING DATA THROUGH THE USE OF LPWAN COMMUNICATION.

### **AUTHORS:**

ALBERT PLAZA SÁNCHEZ  
LEYDER BADILLO VALDES

### **KEYWORDS:**

Web application, database, LPWAN, hive, IoT, monitoring, microcontroller, measurement, API.

### **DESCRIPTION:**

The objective of this document is to present a prototype for monitoring beekeeping data using low power communication (LPWAN) for the realization of an applied engineering project towards agriculture. Where basically the need for the beekeeper to travel constantly to physically check the state of the hive is covered. For this reason, it was also born as a way of studying the population of bees and their honey in a hive through the sensing of variables such as temperature, humidity and weight of the bees and honey. The project includes the description of the instrumentation that defines the reading processes of all the sensors from the microcontroller, the organization of the structure for the sensing process, the sending of information, the coding of all the data obtained to be sent through from the low power communication system (LPWAN) to an online database where it is possible to check on any device with internet access.

## 1. INTRODUCCIÓN

La tecnología está cada vez más cerca de nuestra vida, teniendo cada día más innovación para facilitar los procesos que se realizan cotidianamente. Por esto también se entiende que las personas más alejadas de la tecnología como las personas que viven en áreas rurales necesitan también un apoyo para sus trabajos diarios es así que se está implementando proyectos específicos y estratégicos para mejorar la calidad de vida de estas personas y mejorar procesos o calidades de cultivos

En este caso conociendo la importancia de las abejas, las cuales son responsables de polinizar hasta el 90% de cultivos agrícolas y silvestres,<sup>8</sup> es vital reconocer los contratiempos o problemas que pueden tener como cambios climáticos, fumigaciones, poca tolerancia o miedo por parte de las personas y demás casos que llevan a la extinción de dichos insectos en diferentes lugares “un día están y cuando vamos a revisarlas tiempo después ya no hay ninguna” son las palabras del apicultor con el que se trabajó otros problemas en algunos casos sin respuesta concreta es la falta de conocimiento del por qué se puede tener o no un poco producción de miel en la época de cosecha, además de esto es necesario realizar estudios específicos sobre comportamiento de diferentes razas y entornos. Para este inconveniente una de las soluciones implementadas consiste en la detección, registro y monitoreo de múltiples variables físicas que influyen en el comportamiento, salubridad, bienestar y productividad de las abejas.

Pero surge otra situación; el registro puede tornarse tortuoso, en primer lugar, por la cantidad de variables que puedan y deban ser registradas y porque dichos registros deben ser tomados con tiempos de lectura diarios o de horas, convirtiendo esta labor en titánica si se pretende hacer manualmente y/o por una única persona. Aunado a lo anterior, la toma de algunos datos como la población, la temperatura de la colmena o la cantidad de miel suelen ser procesos invasivos y estresantes para las abejas y tediosos para el apicultor al requerir la apertura física de la colmena para la verificación visual.

Buscando solución para atender las necesidades expuestas por el apicultor y para tener una base para futuros estudios sobre el comportamiento de las abejas se implementa el prototipo para el monitoreo de estas variables físicas a la mayor distancia posible de personas para evitar alterar el comportamiento natural y la intrusión a la colmena, digitalizando los datos para visualizarlos en cualquier parte con acceso a internet.

---

<sup>8</sup> Marta García García, “La polinización en los sistemas de producción agrícola: revisión sistemática de la literatura”, Artículo, IDESIA Chile, Volumen 34, N°3. Páginas 51-66 junio, 2016

## 2. JUSTIFICACIÓN

La gran evolución de la electrónica junto a las tecnológicas ha permitido cambiar de una manera eficaz la forma en que se realizan trabajos cotidianos. La rama de las comunicaciones ha tenido un importante desarrollo en mecanismos de interacción para el hombre, desarrollando nuevas herramientas cada vez más avanzadas para dar satisfacción a las necesidades de servicios de conectividad que son cada vez más exigentes. La relación constante del hombre y la tecnología ha permitido enlazar una conexión con la naturaleza que brinda la posibilidad de efectuar un monitoreo en diversos lugares sin afectar su conservación, pudiendo minimizar la intervención del hombre al obtener una adquisición de datos de manera remota. En la actualidad, las universidades y el estado pretenden impulsar proyectos basados en el desarrollo sostenible, implementando ideas necesarias para la preservación de la biodiversidad. Los Objetivos de Desarrollo Sostenible, también conocidos como los Objetivos Globales, son una llamada por parte de las Naciones Unidas a todos los países del mundo para afrontar los grandes desafíos a los que se enfrenta la humanidad y garantizar que todas las personas tengan las mismas oportunidades y una buena calidad de vida sin comprometer el planeta. De esta forma, el uso de energías alternativas en el desarrollo del proyecto se convierte en una de las mejores opciones <sup>9</sup>.

En este proyecto de investigación se busca diseñar un prototipo de telemetría que permita monitorear datos adquiridos correspondientes a variables de gran importancia en colmenas apícolas, la selección de estas variables y sus respectivos sensores serán un paso determinante para la ejecución del proyecto, teniendo en cuenta el tiempo de funcionamiento, tipo de procesamiento para cada variable, consumo de potencia y requerimientos de instrumentación. Usar nuevas y eficientes formas de comunicación que tengan un gran alcance y bajo consumo de potencia; por ello, Lora WAN se convierte en la tecnología elegida ya que cumple con estos requisitos, permitirá registrar cada variable a una tasa de medición según sea requerida y enviar dichas variables en forma de datos a una plataforma web para que pueda ser visualizada y registrada por el usuario.

En diferentes lugares del mundo se están implementando tecnologías IoT enmarcados en el concepto de ciudades inteligentes; basados en esto, se pretende implementar en el sector rural dichos avances, incluyendo fuentes de energías alternativas. En la actualidad es muy necesario un sistema de monitoreo constante para el sector apícola en el Huila, pues permitiría también la realización de nuevos estudios sobre el comportamiento de esta especie en sitios específicos y comparación entre ellos. De esta manera, se abre un camino hacia la implementación de nuevos mecanismos o estrategias para reducir la mortalidad de

---

<sup>9</sup> Iberoamericanos OEI, [en línea], consultado: (17 de Mayo de 2021) disponible en: <https://www.oei.es/historico/decada/accion.php?accion=25>

las abejas, tan importantes en la preservación de la naturaleza.<sup>10</sup> inteligente de las luces del semáforo para el incremento del flujo vehicular en determinado sector.

### **3. OBJETIVOS**

#### **3.1 OBJETIVO GENERAL**

Diseñar e implementar un prototipo de telemetría basados en la tecnología de redes de baja potencia y área amplia para el monitoreo apícola en el sector del Huila.

#### **3.2 OBJETIVOS ESPECÍFICOS**

- Adaptar e implementar sensores de temperatura, humedad, flujo relativo y peso en una colmena ubicada en una zona rural del departamento del Huila.
- Configurar y acondicionar el funcionamiento de un servidor como interfaz para la visualización de las variables medidas.
- Cuantificar el consumo de potencia de los dispositivos a utilizar.
- Monitorear las variables de temperatura, humedad, flujo relativo (ingreso y salida de abejas) y peso que influyen en la colmena.
- Analizar y comparar los datos monitoreados en la colmena (obtenidos de forma local) con los datos que son recibidos en el servidor (en la nube).
- Validar el funcionamiento del prototipo de telemetría.

---

<sup>10</sup> RIVERA, M. Á. [en línea]. Consultado: (11 de Marzo 2021). Disponible en: <https://repository.javeriana.edu.co/bitstream/handle/10554/12407/BarraganRiveraMiguelAngel2014.pdf;jsessionid=6B74012D5673FE8C15753BD7E4461EF5?sequence=1>

## 4. MARCO TEÓRICO

### 4.1 ANTECEDENTES

La apicultura es el arte de criar abejas para aprovechar sus productos <sup>11</sup>, como la miel, polen, propóleos y jalea real. En Colombia esta actividad agropecuaria está creciendo, y cuenta con la presencia de un gran número de apicultores cuyos productos provenientes de las abejas son comercializados en el mercado nacional. Para ampliar las ventajas que ésta puede ofrecer al sector apícola es necesario difundir conocimiento y aprovechar la tecnología, siempre de la mano con los distintos grupos de interés para contribuir al bienestar, desarrollo y productividad de los apicultores y sus comunidades.

Asia es el principal continente productor de miel; seguido por América donde sobresalen Brasil, México y Argentina como el segundo productor a nivel mundial. Colombia ocupa el puesto 70 en la producción global de miel, según la FAO (Food and Agriculture Organization por sus siglas en inglés).<sup>12</sup> En la actualidad en Colombia la Universidad de Antioquia, la Universidad de Tolima y la Universidad Nacional de Colombia,<sup>13</sup> realizan investigaciones en el área apícola. En la Universidad Nacional de Colombia el Grupo de Apicultura (GAUN), busca contribuir al desarrollo de la producción apícola mediante la formación y capacitación de profesionales, el desarrollo de proyectos de producción, la investigación y transferencia de tecnología a nivel regional y nacional.

Piotr Medrzycki, Fabio Sgolastra, Laura Bortolotti, Gherardo Bogo, Simone Tosi, Erica Padovani, Claudio Porrini y Anna Gloria Sabatini en The International bee research association en el año 2009, desarrollo "Influence of brood rearing temperature on honey bee development and susceptibility to poisoning by pesticides", en el cual mostraron los resultados que influye la disminución de la temperatura de la cría de 2°C en relación a los efectos sobre la mortalidad larvaria y adulta, además de la susceptibilidad a la intoxicación por pesticidas, las larvas de

---

<sup>11</sup> Definición apicultura [en línea] consultado: (4/Abril/2012) <[lema.rae.es/drae/?val=apicultura](http://lema.rae.es/drae/?val=apicultura)>.

<sup>12</sup> Laverde Rodríguez Jairo César, Agenda Prospectiva De Investigación Y Desarrollo Tecnológico Para La Cadena Productiva De Las Abejas Y La Apicultura En Colombia Con Énfasis En Miel De Abejas, Ministerio de Agricultura y Desarrollo Rural, Colombia, 2010, pp. 15, 25

<sup>13</sup> XI encuentro colombiano de apicultura y seminario apícola de los andes xiv jornada apícola universitaria, Bogota, 2018

la abeja fueron criadas a dos temperaturas distintas: 35°C (óptimo) y 33°C (subóptimo), desde 12h después de la incubación hasta los 15 días de edad, el dimetoato que es un tipo de insecticida fue administrado a las larvas y a la abeja adulta, donde la longevidad fue el parámetro medido, concluyendo que las abejas adultas que se derivaban de la cría mantenida a temperatura subóptima tienen una aptitud más baja y son más susceptibles a la intoxicación por pesticida <sup>14</sup>.

Armands Kviesis en el Department of Computer Systems, Faculty of Information Technology, Latvia University of Agriculture, en el año 2016, desarrollo la "Application of neural networks for honey bee colony state identification", En este artículo, los autores proponen un método para la detección del estado de colonias de abejas (inicio del período de cría de cría y enjambres) mediante redes neuronales con aprendizaje supervisado. Esto se puede lograr mediante la inspección de los datos de temperatura y el desarrollo de algoritmos para cada estado de colonia de abejas o mediante la aplicación de redes neuronales. Las redes neuronales se utilizan ampliamente para diversas tareas, incluidas las relacionadas con la clasificación y el procesamiento de datos <sup>15</sup>.

Vyacheslav G. Rybin, en Department of Computer Aided Design, Saint Petersburg Electrotechnical University "LETI" en el año 2017 desarrollo "Embedded DataAcquisition System for Beehive Monitoring", en el cual desarrollaron un sistema de adquisición de datos integrado para el monitoreo automatizado de la colmena. Se proporciona una descripción de los subsistemas de sensores construidos. La solución propuesta adquiere la temperatura, la humedad y el peso de la colmena al referir estos datos a la aplicación móvil a través de una red inalámbrica. El sistema también realiza un análisis de los ruidos de abejas recolectados con redes neuronales artificiales para reconocer las tuberías de abejas e informa al apicultor para evitar el enjambre no deseado. Se representan los esquemas y los algoritmos del sistema propuesto, así como los resultados experimentales obtenidos <sup>1617</sup>.

Aldis Pecka, Vitalijs Osadcuks, en Faculty of Engineering, Latvia University of Agriculture, Latvia, en el año 2017, desarrollo "Development of Internet of Things concept for Precision Beekeeping", esta investigación presenta el diseño conceptual del enfoque de Internet of Thing para la apicultura. Además, ha desarrollado un sistema de monitoreo de la temperatura de la colonia de abejas basado en la red local de sensores inalámbricos junto con la interfaz externa GSM / GPRS para la comunicación basada en paquetes con el servidor remoto en Internet

---

<sup>14</sup> Journal of Apicultural Research "Influence of brood rearing temperature on honey bee development and susceptibility to poisoning by pesticides", 2010

<sup>15</sup> Armands Kviesis, Aleksejs Zacepins, "Application of neural networks for honey bee colony state identification", Carpathian control conference (ICCC) , 2016 pp. 413-417

<sup>16</sup> Vyacheslav G. Rybin "Embedded DataAcquisition System for Beehive Monitoring", Saint-Petersburg, 2017

<sup>17</sup> Aldis Pecka, Vitalijs Osadcuks "Development of Internet of Things concept for Precision Beekeeping", International Carpathian Control Conference (ICCC), 2017



## 4.2 CONTEXTUALIZACIÓN DEL PROBLEMA

En Colombia y en el mundo existen muchos grupos de apicultores que se preocupan por la supervivencia de las abejas, pues estas son muy importantes para el desarrollo del hombre; sin ellas, el 60% de las frutas y verduras que hoy se consumen dejarían de producirse al no ser polinizadas. De las 100 especies de cultivos que abastecen el 90% de los alimentos del mundo, las abejas polinizan más del 70% de ellos. Además, polinizan más de 25.000 especies de plantas con flores. Sin estos insectos la actividad agrícola entraría en una inminente crisis.<sup>18</sup> El planeta está pasando por un fuerte cambio climático que afecta la supervivencia de las abejas, principalmente por fuertes temperaturas e incendios.<sup>19</sup> En Colombia, además de las heladas y las intensas lluvias que obligan a las abejas a no salir de sus colmenas, y por ende no polinizar las plantas ni conseguir su propio alimento, existen otros factores que atentan contra su existencia, pues según los reportes a la cadena apícola, la principal causa de su mortalidad es intoxicación por el uso de agroquímicos. No es claro si se trata de una combinación de malas prácticas, excesos de dosis en aplicaciones agrícolas, productos no autorizados o incluso posibles actos criminales, pues no existen investigaciones detalladas en el país al respecto, pero sí existen reportes en algunos lugares en el Huila, en los cuales se evidencia un uso inadecuado de productos agroquímicos; en este departamento se ha comenzado a hacer polinización manual, ya que la productividad de cultivos de pasifloras ha disminuido a gran escala.<sup>20</sup>

Es por esto, que es necesario para los apicultores tener un sistema con el cual puedan monitorear en tiempo real las principales variables de una colmena, tales como flujo de los insectos, temperatura y humedad; considerando que estas están en lugares con mucha vegetación o floración a largas distancias de los apicultores. Es imprescindible también, tener un registro de los datos para predecir la eficiencia de las cosechas.

### 4.2.1 ABEJAS MELÍFERAS

Las abejas melíferas son insectos sociales con un sistema de castas de individuos. En cada colonia hay una hembra reproductora llamada reina como se muestra en la Figura 1.1a, cuyas estructuras reproductivas maduran como resultado de una dieta

especial alimentada durante las etapas larvarias. En la cima de su fertilidad, una abeja reina puede poner hasta 2,000 huevos por día. Los huevos fertilizados se

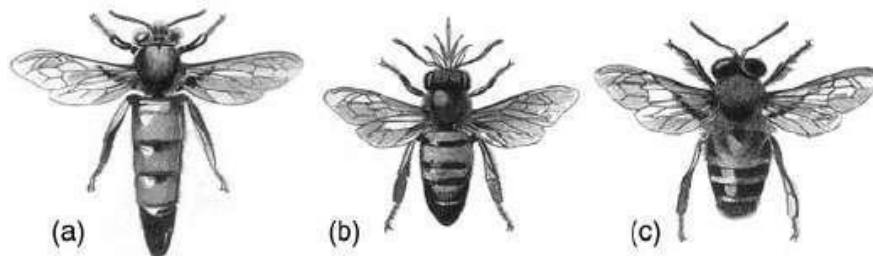
<sup>18</sup> Camargo, M. E.[en línea] En defensa de las abejas. Consultado: (24 de Mayo de 2021). Disponible en: [www.rcnradio.com](http://www.rcnradio.com): <https://www.rcnradio.com/opinion/en-defensa-de-las-abejas>

<sup>19</sup> ONU, N. [en línea] Consultado: (19 de Marzo de 2021). Disponible en: <https://news.un.org/>.  
Obtenido de <https://news.un.org/es/story/2016/02/1351301>

<sup>20</sup> Minagricultura. [en línea]. Plan Nacional de Desarrollo 2014-2018 Todos por un nuevo país. Consultado: (02 de Marzo de 2021) Disponible en:  
<https://www.minagricultura.gov.co/>:<https://www.minagricultura.gov.co/planeacion-control-gestion/Gestin/Plan%20de%20Acci%C3%B3n/PLAN%20NACIONAL%20DE%20DESARROLLO%202014%20-%202018%20TODOS%20POR%20UN%20NUEVO%20PAIS.pdf>

convierten en abejas obreras, y las que no son fertilizadas en machos llamados zánganos <sup>18</sup>.

Las abejas obreras se muestra en la Figura 1.1b, cuidan de la progenie de la reina, limpian la colmena, buscan comida en forma de polen, néctar, y producen miel. El papel que desempeñan las abejas obreras en la colonia está determinado en parte por su edad. Cuando la abeja obrera emerge recientemente, permanece dentro de la colmena, limpiando celdas, alimentando larvas y produciendo panales y miel. A medida que madure, se convertirá en una abeja en busca de néctar y polen. La única función de la abeja zángano macho que se muestra en la Figura 1.1c es aparearse con una reina. En general, una colmena saludable tiene mucho menos zánganos que obreras, y representan solo el 5 % de las abejas en la colmena .



*Figura 1 - Las castas de las abejas melíferas: (a) reina, (b) obrera, (c) zángano* fuente:

a historyof innovation in bee

#### 4.2.2 CICLO DE VIDA

El ciclo de vida de las abejas melífera pasa por cuatro etapas: huevo, larva, pupa y adulto. De la reina pone los huevos, que son de color blanco perla, cilíndricos, ovalados y alargados de entre 1.3 y 1.8 mm de largo, pueden desarrollarse como

---

<sup>18</sup> G. Kritsky, The Quest for the Perfect Hive: A History of Innovation in Bee Culture, 2010.

obreras o zánganos, dependiendo de si los huevos están fertilizados o no. Después de 3 días, una larva emerge del huevo, que será alimentada progresivamente por las abejas nodrizas, durante los primeros 3 días con jalea real, y después con una mezcla de miel y polen. Solo las larvas de la cría reina reciben jalea real durante la etapa larval completa. Al final de esta etapa, durante la cual las células están sin tapar, las larvas giran un capullo y se transforman en pupas después de que los trabajadores adultos han tapado las células. Durante la etapa pupal, la cutícula se oscurece gradualmente, y los músculos y órganos internos cambian a sus formas adultas, antes de que tenga lugar la muda final a la etapa adulta. Finalmente, el adulto comienza a quitar el tapón de la celda de adentro hacia afuera usando sus mandíbulas y emerge de la celda, despliega sus alas y antenas, y comienza sus actividades. La metamorfosis del huevo al adulto es más corta para las reinas, demora tan solo 16 días en *Apis mellifera* y más larga para los zánganos con una duración de 24 días. Para las obreras, se necesitan aproximadamente 21 días para desarrollarse del huevo a la etapa adulta, en la Figura 1.2 se muestra el desarrollo diario del ciclo de vida de las castas de abejas melíferas <sup>19</sup>.

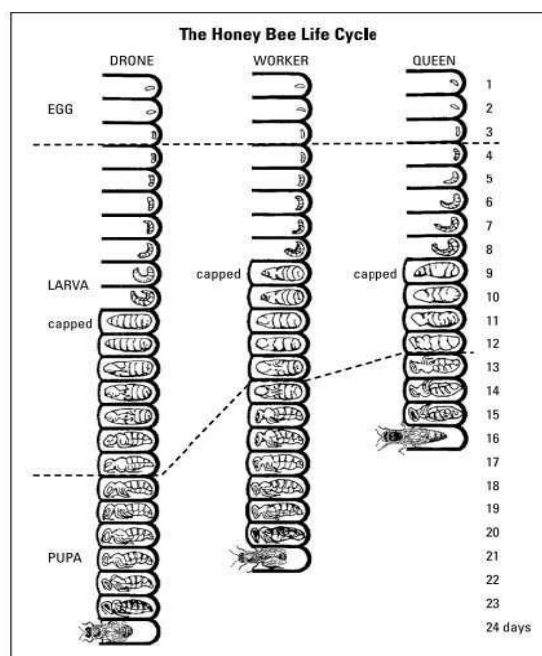


Figura 2- Ciclo de desarrollo diario de las dos castas femeninas y el zángano <sup>20</sup> fuente:

4th ed. John Wiley & Sons

<sup>19</sup> R. Gupta, W. Reybroeck, J. van Veen, and A. Gupta, Eds., *Beekeeping for poverty alleviation and livelihood security: Vol.1: Technological aspects of beekeeping*. Springer Science+Business Media, 2014. Pp 64

<sup>20</sup> H. Blackiston, *Beekeeping For Dummies*, 4th ed. John Wiley & Sons, Inc., 2017.pp 48

### 4.2.3 SEGURIDAD Y MANEJO DE LA COLMENA

La apertura de las colmenas presenta una fuente de estrés para las abejas, lo cual afecta directamente la producción de éstas. Por lo anterior los apicultores evitan en lo posible la apertura de la colonia para no perturbar a sus habitantes. El cuándo, el por qué y durante cuánto se abre una colmena es de gran importancia para llevar un control adecuado de las granjas apícolas y su producción. Ya que esto sólo se hace para realizar monitoreo de rutina.<sup>21</sup>

Existen tres peligros distintos de una exposición prolongada:

- Enfriamiento de la cría.
- Desorganización de la colonia
- Pillaje

Los dos últimos peligros indican al apicultor que ha tenido abierta la colmena por periodos prolongados. El enfriamiento de la cría pasará desapercibido en un primer momento. Se apreciará en días sucesivos al aparecer muertas larvas y ninfas<sup>22</sup>

Una colmena en producción genera beneficios a los apicultores, la posibilidad de perder una colonia en producción y la necesidad de restaurarla genera sobrecostos para el apicultor, además de la obvia pérdida en la producción. La posibilidad del hurto de una o varias colonias es una amenaza que los productores apícolas deben enfrentar.

Desafortunadamente, los apicultores no pueden costear frecuentes inspecciones a las colmenas, además las inspecciones manuales incomodan a las abejas. Lo anterior hace que la información disponible sea escasa; más aún, la información obtenida en las inspecciones manuales no es adicionada a un sistema de información, lo que impide un seguimiento de las condiciones de la colmena. Problemas en las colonias usualmente son descubiertos muy tarde para brindar una solución adecuada, y en muchos casos se pierde valiosa información al no ser registrada.

Según el encuentro de apicultores, realizado en Medellín en mayo de 2008, Colombia cuenta con aproximadamente 120.000 colmenas la mayoría son de tipo Langstroth, y menos de la mitad de los apicultores llevan algún tipo de registro de su producción. Su apicultura está basada en la recolección de enjambres (colonias en tránsito por evasión o por reproducción) y la captura de colonias silvestres (colonias en nidos de abejas ya establecidos), sin implementar ningún tipo de

---

<sup>21</sup> Henríquez Jiménez Félix, Paricio Núñez Enrique, Manual Práctico de Apicultura para las Islas Canarias, consejería de Agricultura y Pesca, 1989. pp. 52

<sup>22</sup> Mazzuferi, V. S. Avalos.. Metamorfosis de Insectos. SERIE DIDACTICA. CUADERNILLO 2. Cátedra Zoología Agrícola. Fac. de Cs. Agrop. U.N.C 1997

mejoramiento genético de este material biológico, factor negativo para la producción de miel.

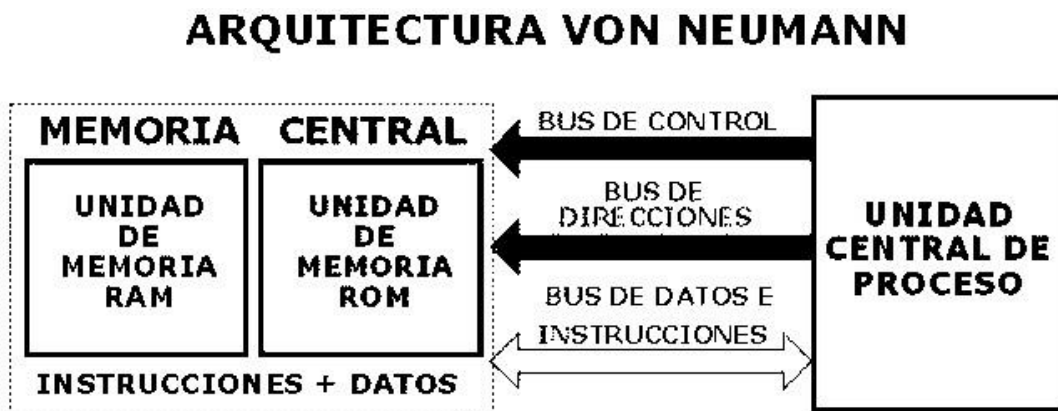
### 4.3 COMUNICACIONES INALÁMBRICAS

La comunicación inalámbrica es la transferencia de información o energía entre dos o más puntos que no están conectados por un conductor eléctrico, se propagan en banda libre o privada para transmitir, es decir, sin la necesidad de cables para la conexión entre el emisor y el receptor <sup>23</sup>.

### 4.4 MICROCONTROLADOR

Un microcontrolador es un dispositivo electrónico capaz de llevar a cabo procesos lógicos para desempeñar una tarea específica, dicha tarea debe ser programada por el usuario a través de un lenguaje de programación. Dentro de los microcontroladores existen dos arquitecturas básicas de hardware Von Neumann y Harvard.

**Von Neumann:** Se caracteriza por tener una memoria única para los datos y las instrucciones del programa. A dicha memoria se accede a través de un sistema de buses único (control, direcciones y datos), en la Figura2.1 se muestra la arquitectura del esquema Von Neumann<sup>24</sup>



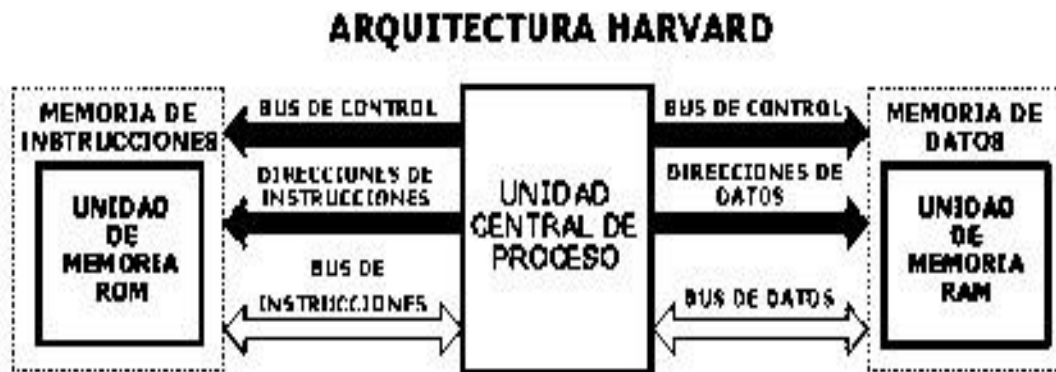
*Figura 3 - Esquema de la arquitectura Von Neumann*

**Harvard:** Este modelo tiene la unidad central de proceso (CPU) conectada a dos memorias, una con las instrucciones y otra con los datos, por medio de dos buses diferentes. Una de las memorias contiene solamente las instrucciones del programa

<sup>23</sup> Ali, D. Jayakody, Y. Chursin, S. Affes, and S. Dmitry, "Recent advances and future directions on underwater wireless communications," Archives of Computational Methods in Engineering, Jan. 2019.

<sup>24</sup> R. digital de la Facultad de Ingeniería UNAM, [En Línea.] "Microcontroladores," consultado (2 de Febrero 2021) Disponible en: <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/760/4/A4.pdf>.

(Memoria de Programa), y la otra sólo almacena datos (Memoria de Datos), en la Figura 2.2 se aprecia la arquitectura Harvard básica de hardware<sup>27</sup>.



*Figura 4 - Esquema de la arquitectura Harvard*

#### 4.4.1 ESP32

Es la denominación de una familia de chips SoC de bajo costo y consumo de energía, con tecnología Wi-Fi y Bluetooth de modo dual integrada. El ESP32 emplea un microprocesador Tensilica Xtensa LX6 en sus variantes de simple y doble núcleo e incluye interruptores de antena, balun de radiofrecuencia, amplificador de potencia, amplificador receptor de bajo ruido, filtros, y módulos de administración de energía. El ESP32 fue creado y desarrollado por Espressif Systems y es fabricado por TSMC utilizando su proceso de 40 nm. La plataforma ESP32 permite el desarrollo de aplicaciones en diferentes lenguajes de programación, frameworks, librerías y recursos diversos. Los más comunes a elegir son: Arduino (en lenguaje C++), Esp-idf (Espressif IoT Development Framework) desarrollado por el fabricante del chip, Simba Embedded Programming Platform (en lenguaje Python), RTOS's (como Zephyr Project, Mongoose OS, NuttX RTOS), MicroPython, LUA, Javascript (Espruino, Duktape, Mongoose JS), Basic. Al trabajar dentro del entorno Arduino podremos utilizar un lenguaje de programación conocido y hacer uso de un IDE sencillo de utilizar, además de hacer uso de toda la información sobre proyectos y librerías disponibles en internet. La comunidad de usuarios de Arduino es muy activa y da soporte a plataformas como el ESP32.

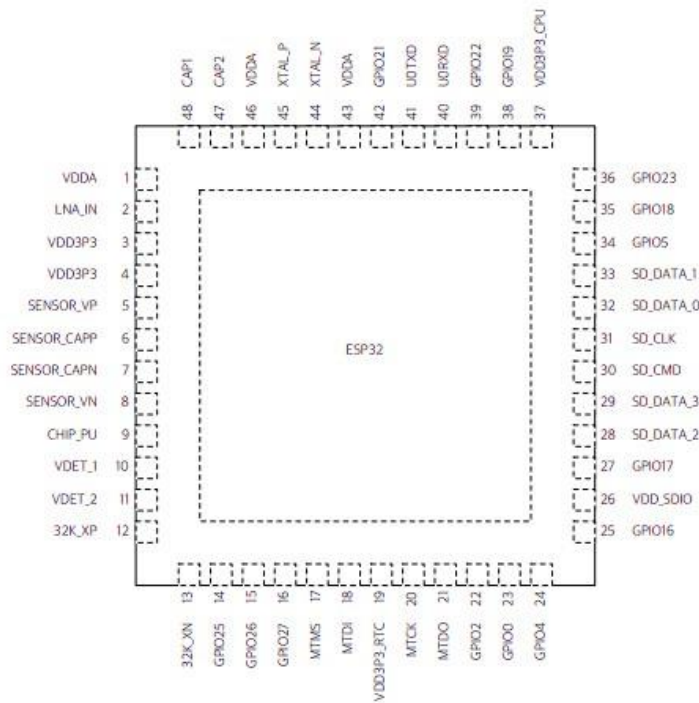


Figura 5 - Configuración de pines ESP32 <sup>25</sup>

#### 4.4.2 Esp8266

Es un microcontrolador con procesador Tensilica Xtensa LX106 80 MHz (160 MHz overck.) de 64KB de baja potencia. Con frecuencia de 80MHz, los dispositivos logran el rendimiento de la CPU se supera mas de un millón de instrucciones por segundo (MIPS) por megahertz, lo que permite al diseñador del sistema optimizar el consumo de energía frente a la velocidad de procesamiento. permite a otros microcontroladores conectarse a una red inalámbrica Wi-Fi y realizar conexiones simples con TCP/IP usando comandos al estilo Hayes.<sup>26</sup>

<sup>25</sup> All Datasheet [en línea] consultado: (4 de Febrero de 2021) Disponible: <https://www.alldatasheet.com/datasheet-pdf/pdf/1243003/ESPRESSIF/ESP32.html>

<sup>26</sup> Iotbytes [en línea] consultado (5 de Febrero de 2021) Disponible en <https://iotbytes.wordpress.com/esp8266-pinouts/>



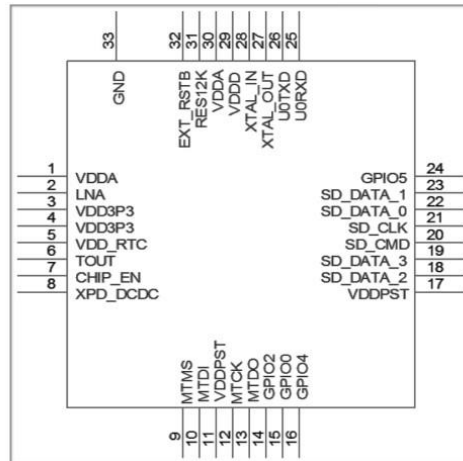


Figura 6 - Configuración de pines Esp8266 <sup>29</sup>

#### 4.4.3 Modulo HX711

El chip HX711 posee internamente la electrónica para la lectura del puente de Wheatstone formado por la celda de carga y también un conversor ADC de 24 bits. Se comunica con el microcontrolador por medio de un protocolo de tipo serial mediante 2 pines (Clock y Data).

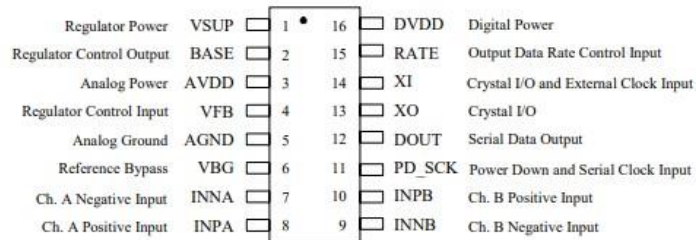


Figura 7 - Configuración de pines HX711 <sup>27</sup>

## 5. VARIABLES A MEDIR

Son aquellos fenómenos naturales a los cuales se les dará un valor lineal para tómalas como objetos de estudio.

### 5.1 Temperatura

Magnitud física que expresa el grado o nivel de calor de los cuerpos o del ambiente, y cuya unidad en el sistema internacional es el kelvin (K).[14]

<sup>27</sup> Sparkfun [en línea] Consultado: (2 de Marzo de 2021) Disponible en: [https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711\\_english.pdf](https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf)



## 5.2 Humedad relativa

Expresión porcentual de la cantidad de vapor de agua presente en el aire con respecto a la máxima posible para unas condiciones dadas de presión y temperatura.<sup>28</sup>

## 5.3 Masa

Magnitud física que expresa la cantidad de materia de un cuerpo, medida por la inercia de este, que determina la aceleración producida por una fuerza que actúa sobre él, y cuya unidad en el sistema internacional es el kilogramo (kg).<sup>29</sup>

## 5.4 Flujo

Acción y efecto de fluir.<sup>30</sup>

# 6. SENSORES

Los sensores son piezas clave en los proyectos con microcontroladores, permiten cuantificar una señal externa y enviarla al microcontrolador. Además, existen diferentes familias de sensores y cada uno de ellos tiene un uso concreto.<sup>31</sup>

## 6.1 SENSOR DE TEMPERATURA

Los sensores de temperatura convierten una magnitud física en una resistencia o tensión eléctrica. El uso de sensores de temperatura es muy amplio. Sea que se trate de la temperatura ambiental en la casa o en la oficina o la temperatura precisa de un material en proceso de ebullición, la medición de temperatura en el ámbito privado o industrial es muy importante. Los sensores de temperatura usan diferentes efectos físicos para convertir la temperatura en una magnitud eléctrica.<sup>32</sup>

## 6.2 SENSOR DE HUMEDAD

Los sensores de humedad convierten la magnitud física de humedad del aire en una señal normalizada y tiene un amplio ámbito de aplicación y sirve para determinar la humedad relativa [%] y absoluta [g/m<sup>3</sup>] con una gran precisión.<sup>34</sup>

## 6.3 SENSOR FOTOELÉCTRICO

Son dispositivos que detecta la presencia o alguna característica en particular de un objeto mediante luz visible o no visible. Se pueden aplicar para detectar

---

<sup>28</sup> Española, R. A. (s.f.). [en línea] Humedad Relativa. Consultado (7 de Febrero de 2021) Disponible en: <https://dle.rae.es/>: <https://dle.rae.es/humedad?m=form#G3gHYnG>

<sup>29</sup> Española, R. A. (s.f.). [en línea] Masa. Consultado (5 de Marzo 2021) Disponible: <https://dle.rae.es/>: <https://dle.rae.es/masa?m=form>

<sup>30</sup> Española, R. A. (s.f.). [en línea] Flujo. Consultado (5 de Marzo 2021) Disponible: <https://dle.rae.es/>: <https://dle.rae.es/flujo?m=form>

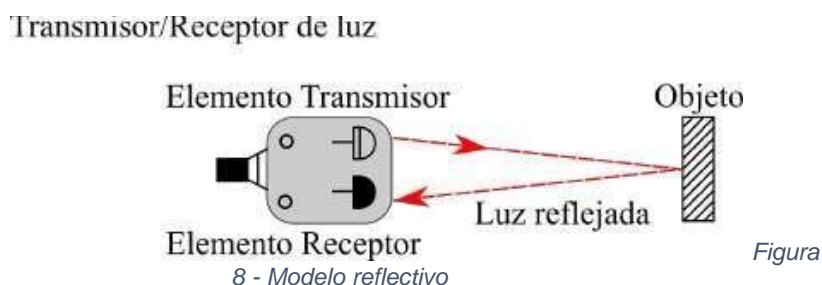
<sup>31</sup> Microcontroladores, [En Línea.] "Sensores con microcontroladores," Consultado (6 de Marzo de 2021) Disponible: <https://conmicrocontroladores.com/sensores/>.

<sup>32</sup> CE-Ibérica-S.L., [En Línea.] "Equipos de medida - balanzas - regulación y control," Consultado (3 de Marzo 2021) Disponible: <https://www.pce-iberica.es/instrumentos-de-medida/sistemas/>.

presencia, tamaño, color y brillo de objetos. Existen tres modos de configuración fotoeléctrica.<sup>33</sup>

### 6.3.1 Modelo reflectivo

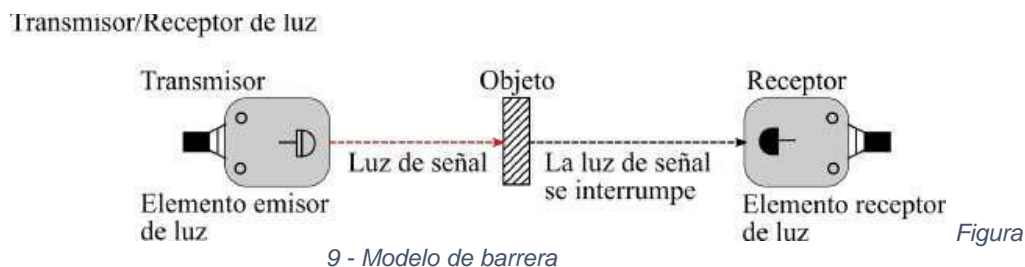
El modelo reflectivo de la Figura 1.22 el emisor de luz y los elementos receptores están contenidos en una sola carcasa. El sensor recibe la luz reflejada desde el objeto.<sup>35</sup>



Fuente: Equipos de medida - balanzas - regulación y control

### 6.3.2 Modelo de barrera

El modelo de barrera de la Figura 1.23 el transmisor y el receptor están separados, cuando el objeto se encuentra entre el transmisor y el receptor, se interrumpe la luz y manda una alerta.<sup>35</sup>



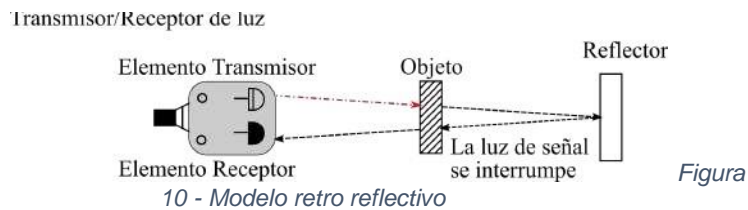
Fuente: Equipos de medida - balanzas - regulación y control

### 6.3.3 Modelo retro reflectivo

El modelo retro reflectivo de la Figura 1.24 el emisor de luz y los elementos receptores están contenidas en un mismo recinto. La luz del elemento emisor incide en el reflector y regresa al elemento receptor de luz. Cuando hay un objeto presente, se interrumpe la luz.<sup>35</sup>

---

<sup>33</sup> EYENCE, [En Línea.] "Fundamentos del sensor: Guía de sensores para fábricas clasificados por principios," Consultado: (8 Marzo 2021) Disponible: <https://www.keyence.com.mx/ss/products/sensor/sensorbasics/photoelectric/info/>



Fuente: Equipos de medida - balanzas - regulación y control

## 7. SENSORES UTILIZADOS PARA LAS VARIABLES MEDIDAS POR EL PROTOTIPO

### 7.1 Temperatura, Humedad, Peso, Flujo

#### 7.1.1 Sensor DHT11

El DHT11 es un sensor de humedad relativa y temperatura de bajo costo y de media precisión a un bajo precio. La salida suministrada es de tipo digital utilizando solamente 1 pin de datos.

Temperatura y Humedad (DHT 11):

Aplicaciones:

- Monitoreo de temperatura
- Monitores humedad

Datos técnicos:

- Voltaje de Operación: 3V - 5V DC
- Rango de medición de temperatura: 0 a 50 °C
- Precisión de medición de temperatura:  $\pm 2.0$  °C
- Resolución Temperatura: 0.1°C
- Rango de medición de humedad: 20% a 90% RH.
- Precisión de medición de humedad: 5% RH. □ Resolución Humedad: 1% RH
- Tiempo de sensado: 1 seg.
- Interface digital: Single-bus (bidireccional)
- Modelo: DHT11 □ Dimensiones: 16\*12\*5 mm □ Peso: 1 gr.
- Carcasa de plástico celeste

Pines:

- 1- Alimentación: +5V (VCC)
- 2- Datos (DATA)
- 3- No Usado (NC)
- 4- Tierra (GND)



11 - Sensor DHT11

Figura

Fuente: Fotografiado aportada por los tesistas

### 7.1.2 Celda de carga 50kg

Celda de carga, es un transductor que puede medir el peso o la fuerza y la convierte en una señal eléctrica, mediante un dispositivo mecánico, la fuerza que se desea medir deforma el sensor y convierte el desplazamiento o deformación en señales eléctricas. Aplicaciones:

- Báscula digital

Datos técnicos:

- Peso máximo 50 Kg
- Sensibilidad  $1.0 \pm 0.1$  □ Voltaje de entrada  $\leq 10$
- Rango de temperatura de operación  $0 - +50$

Pines:

- Negro
- Rojo
- Blanco



Figura 12 - Celda de carga

Fuente: Fotografiado aportada por los tesistas

### 7.1.3 Sensor de proximidad

Este sensor es utilizado para tener un estimado de flujo de abejas en la colmena.

**Sensor infrarrojo:** Transmisor y un receptor infrarrojo, el emisor emite una señal infrarroja que se refleja en un obstáculo la cual es recibida por el receptor, un led se enciende cuando se detecta un obstáculo. la sensibilidad puede ser ajustada mediante un potenciómetro.

Aplicaciones:

- Detector de obstáculos
- Seguidores de línea
- Detector de proximidad Datos técnicos:
- Voltaje: 3.3 ~ 5 VDC
- Rango de detección: 2 ~ 30 cm
- Angulo de detección: 35°
- Salida digital: 0 o 5 VDC
- Dimensiones: 32 x 14 mm

Pines:

- 1 Vcc
- 2 Gnd
- 3 Out



Figura 13 - Sensor de proximidad infrarrojo

Fuente: Fotografiado aportada por los tesistas

## 8. DISEÑO Y DESARROLLO SOFTWARE

### 8.1 PROGRAMACIÓN Y CALIBRACIÓN DE CIRCUITOS INTEGRADOS Y SENSORICA

#### 8.1.1 Circuitos Integrados

Los circuitos integrados se escogieron basándose en la revisando de la hoja de datos o técnica, analizando sus configuraciones que determinaron la viabilidad en el proyecto.

El principal reto se dio en el tipo de conexiones disponibles a nivel local debido a que el protocolo a usar para el envío de datos recolectados al dispositivo conectado a internet se realizaría vía LPWAN para ser exactos mediante la tecnología LoRa, los tipos más factibles fueron: Mediante conexión serial, conexión bluetooth, conexión wifi que son los tipos de conexión local permitidos por el esp32 y el esp8266, la conexión de uso fue la wifi por varios factores, el primero y más importante era la compatibilidad entre dispositivos, el segundo fue que la conexión serial del esp8266 se utilizaría para almacenar los datos en una SD y por último potencia de señal dado que el bluetooth no logro superar una de las barreras de prueba como lo fue una pared de concreto sin embargo el wifi no tubo problema además de la distancia alcanzada por el wifi es mayor que la que logra alcanzar el bluetooth, una vez se superó y se tuvo claro los protocolos de uso se procede a la investigación y prueba de códigos para mirar la mayor eficiencia que presenten cada uno de ellos con los parámetros que nos eran necesarios como lo son el tipo de sensores, recolección de datos y compatibilidad con la recepción y envío de datos.

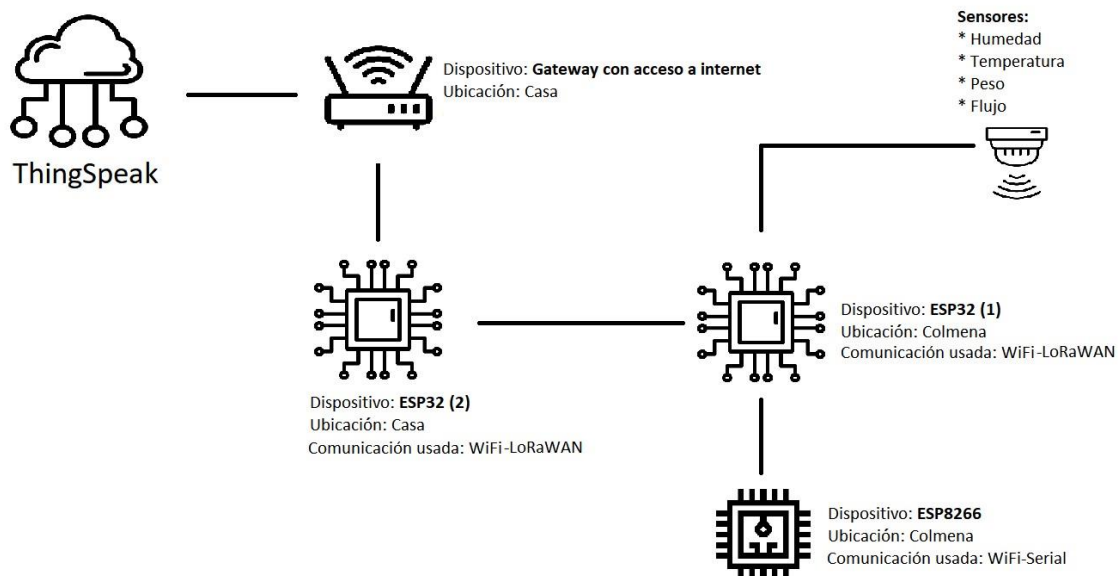


Figura 14 - Diagrama sistema de comunicación del proyecto

Fuente: Realizado por los tesistas

La programación se divide en 3 dispositivos y 6 configuraciones, para los dispositivos se presentan:

Dispositivo: ESP32 (1)

Ubicación: Colmena

Comunicación usada: WiFi-LoRaWAN

Instrucciones: El dispositivo esta programado para realizar la lectura de las variables como lo son humedad, temperatura, peso y flujo, mediante los sensores que se encuentran ubicados en la parte interna de la colmena, una vez obtenido dicha información se realiza un envío continuo y encriptado por medio de la comunicación LoRaWAN para de esta forma ser recibido por el ESP32(2), estos datos a su vez son enviado mediante WiFi la misma información al ESP8266.

Dispositivo: ESP32 (2)

Ubicación: Casa

Comunicación usada: WiFi-LoRaWAN

Instrucciones: El dispositivo es programado para recibir los datos enviados por el ESP32 (1) y una vez obtenida esta información enviarla a la nube a una API llamada ThingSpeak, todo esto se lleva acabo con una configuración que suministra la API como lo son validación de registro y puertos de acceso a plataforma.

Dispositivo: ESP8266 (3)

Ubicación: Colmena

Comunicación usada: WiFi-Serial

Instrucciones: El dispositivo esta configurado para realizar una petición mediante WiFi utilizando el protocolo http al ESP32 (1), en la cual extrae la información obtenida de los sensores para ser archivada o guardada en una Micro SD que se encuentra conectada a este dispositivo mediante una conexión serial.

Las 6 configuraciones que se mencionan son:

- 1) Configurar un usuario en ThingSpeak para de esta forma lograr crear un canal ya sea privado o público que permita archivar los datos obtenidos en la nube.
- 2) Configurar un canal en ThingSpeak que permita el acceso mediante los API Keys.
- 3) Configurar el router con usuario y contraseña deseadas para que el ESP32 (1) con la programación ya mencionada tenga acceso a internet.
- 4) Configurar el Key entre los dispositivos ESP32(1) Y ESP32 (2), para que de esta forma los datos sean obtenidos solo por el ESP32 (2) y no cualquiera que utilice la frecuencia de 915MHz.
- 5) Configuración de la red WiFi que permitirá la conexión en los dispositivos ESP32(1) y ESP8266.
- 6) Configurar los puertos de lectura en el ESP32 (1) para cada uno de los sensores que se desea utilizar.

### 8.1.2 Sensorica

Para la implementación del sistema se seleccionaron los sensores más adecuados para la creación del prototipo. Teniendo en cuenta el costo, eficiencia y sensibilidad. Además de estudiar cuales son las variables que son más útiles para los apicultores al momento de querer saber de las colmenas y su comportamiento; se seleccionan las variables: Temperatura, Humedad, Flujo y Peso.

#### 8.1.2.1 Sensor Temperatura y Humedad DHT11

El primer sensor es el **DHT11**, el cual mide la variación de temperatura y humedad en el medio, el **DHT11** no utiliza una interfaz serial estándar como I2C, SPI o 1Wire (en realidad utiliza algo similar a esta ultimo). En cambio, requiere su propio protocolo para comunicarse a través de un solo hilo. Afortunadamente el **protocolo de comunicación del DHT11 es simple** y puede implementarse tranquilamente usando los pines de I/O en una IDE de Arduino. La IDE de Arduino debe iniciar la comunicación con el DHT11 manteniendo la línea de datos en estado bajo durante al menos 18 ms. Luego el DHT11 envía una respuesta con un pulso a nivel bajo (para indicar su presencia) de 80 uS y luego deja “flotar” la línea de datos por otros 80 uS.

En la figura de abajo, el pulso de inicio enviado por el microcontrolador esta coloreado en rojo, mientras que la respuesta desde el sensor esta coloreada en azul.

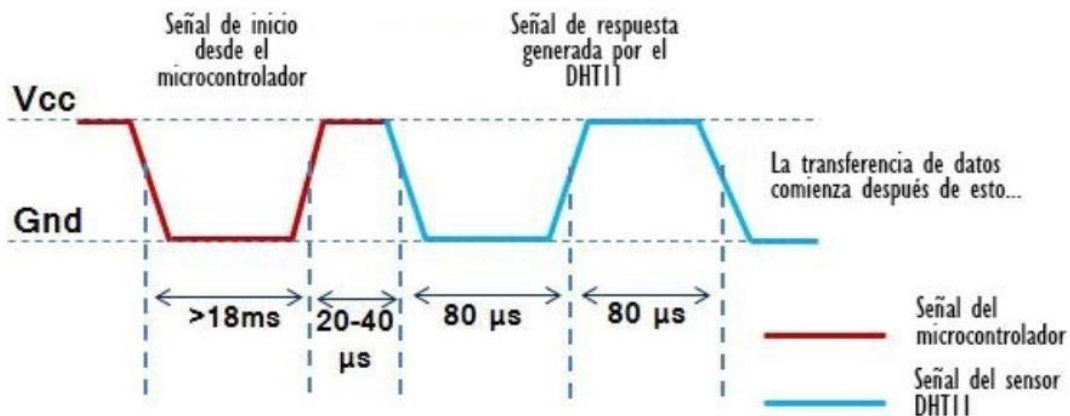


Figura 15 - Esquema de señales del microcontrolador con en el sensor DHT11

Fuente: Datasheet DHT11

Mediante una codificación de datos está basada en un esquema de ancho de pulso (se toma en cuenta el ancho del estado alto): Un pulso ancho representa un 1 lógico, un pulso corto representa un 0 lógico.



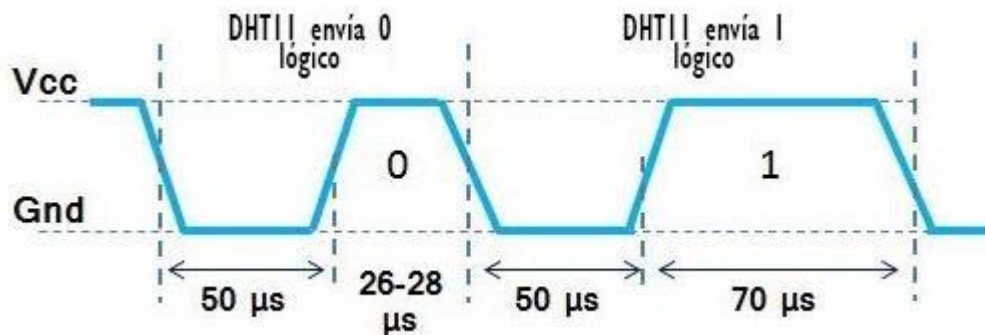


Figura 16 - Esquema funcional de señales en el sensor DHT11

Fuente: Datasheet DHT11

Todos los bits comienzan con un pulso bajo de 50 uS. Las librerías de comunicación con el DHT11 aprovechan este pulso para la sincronización. Luego viene un pulso alto que varía según el estado lógico o el valor del bit que el DHT11 desea transmitir:

- Se utilizan pulsos de 26-28 microsegundos para un "0" □ Se utilizan pulsos de 70 microsegundos para un "1"

En la siguiente ilustración observamos el inicio de una comunicación con el DHT11 o DHT22, resumiendo lo que hemos hablado con anterioridad:

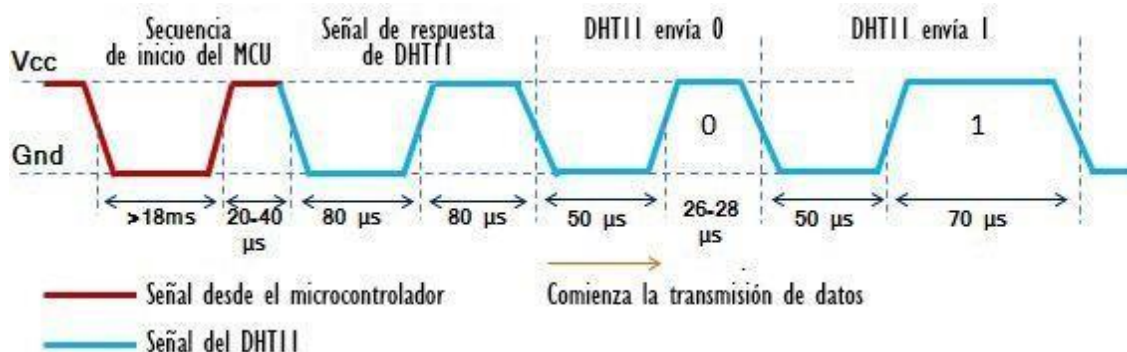
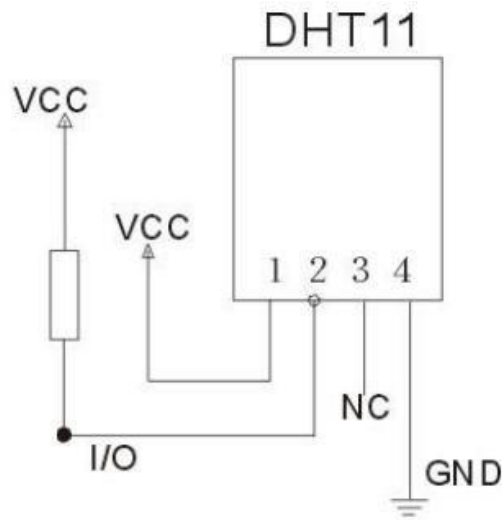


Figura 17 - Esquema funcional de señales transmitidas por el sensor DHT11

Fuente: Datasheet DHT11

Una transmisión completa comienza como ya describimos y se compone de 40 bits (5 bytes) que incluyen todos los datos que el sensor puede proporcionar. En cuanto a los datos que se transmiten, su interpretación es como sigue:

- El primer byte que recibimos es la parte entera de la humedad relativa (RH).  
□ El segundo byte es la parte decimal de la humedad relativa (no se utiliza en el DHT11, siempre es 0).
- El tercer byte es la parte entera de la temperatura.
- El cuarto byte es la parte decimal de la temperatura (no se utiliza en el DHT11, siempre es 0).
- El ultimo byte es la suma de comprobación (checksum), resultante de sumar todos los bytes anteriores.



*Figura 18 - Esquema de Aplicación Típico Periférico del sensor DHT11*

**Fuente:** Datasheet DHT11

### 8.1.2.2 Sensor de Flujo FC-51

Este módulo sensor infrarrojo emisor y receptor el cual se puede adaptar a luz ambiente y distancia de detección a través de un potenciómetro que viene incluido en la board, dicha distancia se encuentra comprendida entre 2cm – 8cm, con un ángulo de detección de 35°. Estos infrarrojos emiten un haz de luz infrarrojo, cuando el haz de luz se encuentra con un obstáculo rebota y es recibido por el receptor que se encarga de recibir la cantidad de luz rebotada, este fenómeno se usa para identificar si hay o no un obstáculo o también si hay una línea negra o blanca, ya que el negro no refleja la luz y el blanco refleja toda la luz. La señal captada por estos sensores es acondicionada mediante un circuito comparador, esto se ve reflejado mediante un LED indicador de color verde, para la programación del sensor IR FC-51 se utilizan los niveles lógicos presentados alto o bajo, de esta forma se permite detectar el paso de las abejas, cuando la abeja está en el lugar se estimó

un tiempo de ingreso o salida de 1.5sg y el contador suma +1 de esta forma determinamos una cantidad aproxima de abejas que ingresan o salen de la colmena esto se hace debido a que los parámetros de sensibilidad están ubicados en la placa del sensor mediante el potenciómetro, el diseño de software se limita a un alto o un bajo de la señal de entrada al **ESP32**, lo anterior se da gracias a la configuración de la placa con su amplificador comparador, como se muestra en la Figura X, la respuesta de funcionamiento ante algunas interrupciones en el sensor.

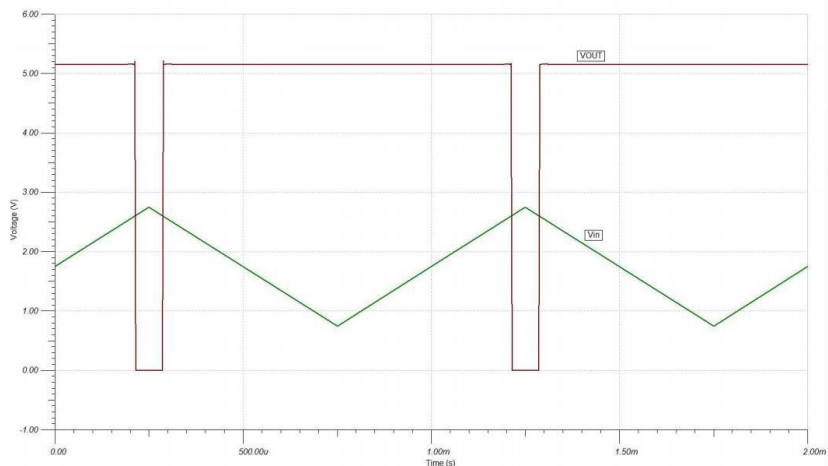


Figura 19 - Respuesta de un amplificador comparador básico

Fuente: DataSheet LMx93-N

Para la implementación de lectura de la interrupción se presenta las siguientes conexiones.

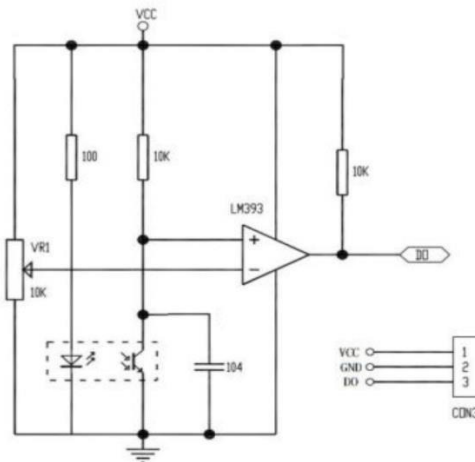


Figura 20 - Diagrama conexiones eléctricas para el sensor FC-51

Fuente: DataSheet FC-51

### 8.1.2.3 Sensor de Peso (Celda de carga de 50Kg)

La Celda es muy usada en básculas digitales, siendo que la señal eléctrica de salida es del orden de unos pocos milivoltios es ideal usarlo con el Módulo HX711 para poder tener la información digital y procesarla en alguna tarjeta de desarrollo Arduino, ESP32, ESP8266 y/o Raspberry, en la Figura X se presenta el diagrama de bloques del circuito denominado HX711, en la cual presenta los acoples y trama de datos de 24bits que se envía a la tarjeta de desarrollo, se presenta el voltaje de alimentación suministrado, los reguladores, el MUX usado para de esa forma realizar el AC/DC, se presenta un oscilador que lleva el tiempo en que se envía la trama de datos, de esta forma mediante un CLK sincronizado en la tarjeta de desarrollo y el HX711, se obtienen los bits necesarios que presentan qué tan deformados se encuentran las celdas de carga y de esa forma usar el dato digital suministrado para ser enviado a la nube y almacenado en la tarjeta SD.

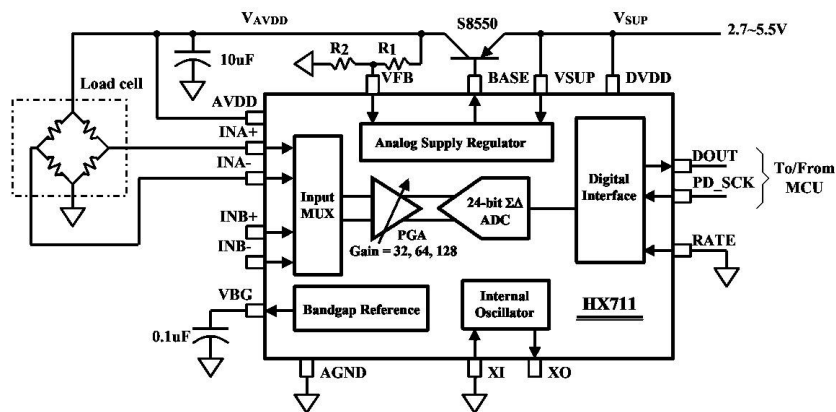


Figura 21 - Diagrama de bloques del circuito HX711

Fuente: Datasheet HX711

Pin #	Name	Function	Description
1	VSUP	Power	Regulator supply: 2.7 ~ 5.5V
2	BASE	Analog Output	Regulator control output (NC when not used)
3	AVDD	Power	Analog supply: 2.6 ~ 5.5V
4	VFB	Analog Input	Regulator control input (connect to AGND when not used)
5	AGND	Ground	Analog Ground
6	VBG	Analog Output	Reference bypass output
7	INA-	Analog Input	Channel A negative input
8	INA+	Analog Input	Channel A positive input
9	INB-	Analog Input	Channel B negative input
10	INB+	Analog Input	Channel B positive input
11	PD_SCK	Digital Input	Power down control (high active) and serial clock input
12	DOUT	Digital Output	Serial data output
13	XO	Digital I/O	Crystal I/O (NC when not used)
14	XI	Digital Input	Crystal I/O or external clock input, 0: use on-chip oscillator
15	RATE	Digital Input	Output data rate control, 0: 10Hz; 1: 80Hz
16	DVDD	Power	Digital supply: 2.6 ~ 5.5V

*Tabla 1 Funciones que pueden ser utilizadas en el HX711*

Fuente: Datasheet HX711

Para entender de forma detallada el funcionamiento del Driver HX711 se referencia el código usado por el ensamblador para la lectura del transductor (Celda de carga)

### Reference Driver (Assembly)

```

/*-----*/
Call from ASM:      LCALL  ReaAD
Call from C:      extern unsigned long ReadAD(void);

                .
                .
                unsigned long data;
                data=ReadAD();
                .
                .
/*-----*/
PUBLIC      ReadAD
HX711ROM    segment code
rseg        HX711ROM

sbit        ADD0 = P1.5;
sbit        ADSK = P0.0;
/*-----*/
OUT:  R4, R5, R6, R7    R7=>LSB
/*-----*/

ReadAD:
CLR        ADSK          //AD Enable (PD_SCK set low)
SETB       ADD0          //Enable 51CPU I/O
JB         ADD0,$         //AD conversion completed?
MOV        R4,#24

ShiftOut:
SETB       ADSK          //PD_SCK set high (positive pulse)
NOP
CLR        ADSK          //PD_SCK set low
MOV        C,ADD0        //read on bit
XCH        A,R7          //move data
RLC        A
XCH        A,R7
XCH        A,R6
RLC        A
XCH        A,R6
XCH        A,R5
RLC        A
XCH        A,R5
DJNZ       R4,ShiftOut    //moved 24BIT?
SETB       ADSK
NOP
CLR        ADSK
RET
END

```

Una vez se entiende el lenguaje de bajo nivel usado podemos detallar la configuración de envío de datos en el lenguaje de programación C, donde realiza la lectura utilizando la trama de 24Bits.

### Reference Driver (C)

```
//-----
sbit ADD0 = P1^5;
sbit ADSK = P0^0;
unsigned long ReadCount(void) {
    unsigned long Count;
    unsigned char i;
    ADD0=1;
    ADSK=0;
    Count=0;
    while (ADD0);
    for (i=0;i<24;i++) {
        ADSK=1;
        Count=Count<<1;
        ADSK=0;
        if (ADD0) Count++;
    }
    ADSK=1;
    Count=Count^0x800000;
    ADSK=0;
    return (Count);
}
```

Teniendo claro que la trama de 24Bits será suministrada a la tarjeta de desarrollo se debe conocer de qué forma se realiza el envío, para ello se toma la Figura X la cual referencia las señales en bits de salida como DOUT, el cual envía desde el MSB (El bit más significativo o de mayor peso) al LSB (El bit menos significativo o de menor peso); los PD\_SCK (Power Down and Serial Clock Input ) hacen referencia al apagado e ingreso de la trama de envío serial que genera el HX711 mediante el cual se comunica con la placa de desarrollo, como se puede observar se manejan 3 medidas simultaneas las cuales se reinician al momento de terminar el envío de la trama.

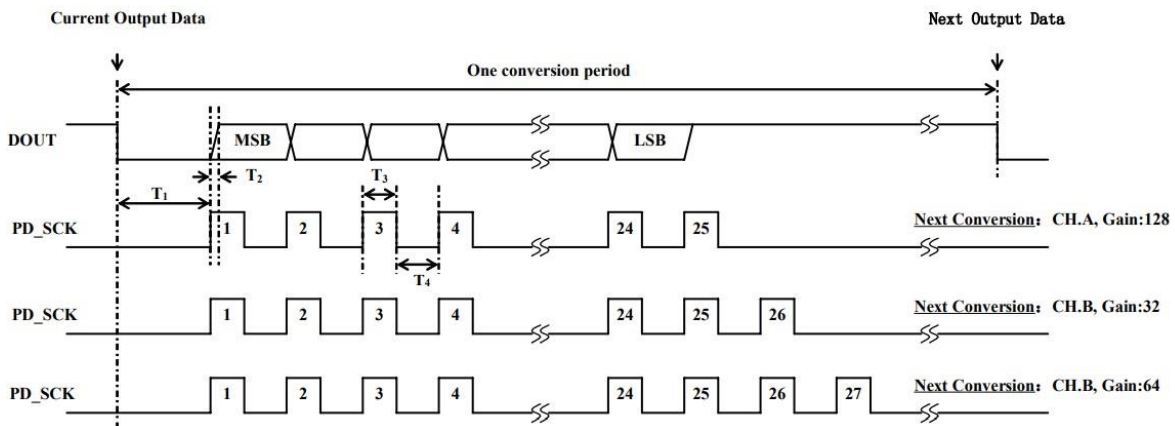


Figura 22 - Señales enviadas por el HX711 a la placa de desarrollo

Fuente: Datasheet HX711

Para lograr obtener los datos se utiliza la librería HX711, la cual permite la decodificación de los 24Bits enviados, pero para ello se debe hacer la conexión física de la entrada de datos y el CLK, una vez realizado esto se debe realizar una calibración de las celdas de carga la cual se realiza utilizando masas patrón debidamente certificadas y la conexión a un computador mediante la comunicación serial de nuestras placas de desarrollo, lo normal de una celda de carga se encuentra con un factor de más o menos 10.000 sin embargo para cada arreglo de celdas este valor cambia, en el caso que se presenta en este documento el factor de calibración se encontró en los 22.450, también cabe resaltar que este valor fue el que menos variación nos presentó en pesos altos, dado que el factor varía dependiendo del rango de uso, si son valores que no superan los 40Kg en el caso específico de este documento el factor no superó los 10.000, pero para los valores que deseábamos lo supero.

## **8.2 COMUNICACIÓN CON LA API THINGSPEAK Y BASE DE DATOS EN LA NUBE**

### **8.2.1 ThingSpeak**

Es una plataforma de IoT que utiliza canales para almacenar datos enviados desde aplicaciones o dispositivos. Con la configuración descrita en Configuraciones de canal, usted crea un canal y luego envía y recupera datos hacia y desde el canal. Puede hacer públicos sus canales para compartir datos, puede crear un canal y actualizar su feed, actualizar un canal existente, borrar un feed de canal y eliminar un canal.

Para utilizar nuestro canal o canales con API REST se debe entender el funcionamiento y utilidad que nos presenta. La transferencia de estado representacional (REST) es un estilo arquitectónico diseñado como un modelo de solicitud-respuesta que se comunica a través de HTTP. ThingSpeak, una plataforma de IoT, utiliza las llamadas a la API REST GET, POST, PUT y DELETE para crear y eliminar canales, leer y escribir datos de canales y borrar los datos de un canal. Un navegador web o un cliente envía una solicitud al servidor, que responde con datos en el formato solicitado. Los navegadores web utilizan esta interfaz para recuperar páginas web o enviar datos a servidores remotos.

[https://www.mathworks.com/help/thingspeak/channels-and-chartsapi.html#update\\_feed](https://www.mathworks.com/help/thingspeak/channels-and-chartsapi.html#update_feed)

### **8.2.2 Canal en ThingSpeak**

Un canal es una herramienta pública o privada que brinda ThingSpeak con la finalidad de obtener un espacio de información y datos relacionados al proyecto de cualquiera de los usuarios que utilicen esta plataforma de IoT.



**Crear canal:** para la creación de un canal debemos ser usuarios de ThingSpeak lo cual lo podemos hacer desde la URL

<https://thingspeak.com/login?skipSSOCheck=true> en la opción “créate one”

The image shows the 'Create MathWorks Account' form on the ThingSpeak website. The form includes fields for 'Email Address' (with the example 'ejemploemail@gmail.com'), 'Location' (a dropdown menu set to 'United States'), 'First Name', and 'Last Name'. There are 'Continue' and 'Cancel' buttons at the bottom. To the right of the form is a diagram illustrating the ThingSpeak architecture. It shows 'SMART CONNECTED DEVICES' sending data to a cloud labeled 'DATA AGGREGATION AND ANALYTICS ThingSpeak'. This cloud then connects to a 'MATLAB' interface, which is labeled 'ALGORITHM DEVELOPMENT SENSOR ANALYTICS'.

Figura 23 - Interfaz de creación de usuario en ThingSpeak

Fuente: Captura tomada por los autores

Ya creado el usuario de la plataforma ThingSpeak el reconocimiento del entorno es fundamental en la parte superior de la interfaz se encuentra las opciones de “Channels” – Canales, la cual desplegará una serie de opciones que incluyen My Channels, Watched Channels, Public Channels, los cuales permitirán ver los canales creados por el usuario ya sean públicos o privados, también permite realizar un rastreo de canales vistos por el usuario que pertenecen a otros usuarios, por ultimo en esa pestaña permite realizar la visualización y búsqueda de otros canales creados por distintos usuarios que hacen públicos, seguido tenemos “Apps”- Aplicaciones la cual despliega una serie de opciones que nos permite entender o indagar sobre otras aplicaciones que tiene esta plataforma como lo son: Todas las aplicaciones, análisis MATLAB, Visualización de MATLAB, Complementosm ThigsTweet, Control del tiempo, Reaccionar, Replicar, ThingHTTP, seguido de esto nos encontraremos con la pestaña de Devices, en esta no permite la conexión de dispositivos con protocolo de conexión MQTT por ultimo en los desplegables de control de usuario tenemos la ayuda que nos suministra la plataforma.



MQTT: es un protocolo de red ligero de publicación-suscripción que transporta mensajes entre dispositivos. El protocolo generalmente se ejecuta sobre TCP / IP; sin embargo, cualquier protocolo de red que proporcione conexiones bidireccionales ordenadas y sin pérdidas puede admitir MQTT.

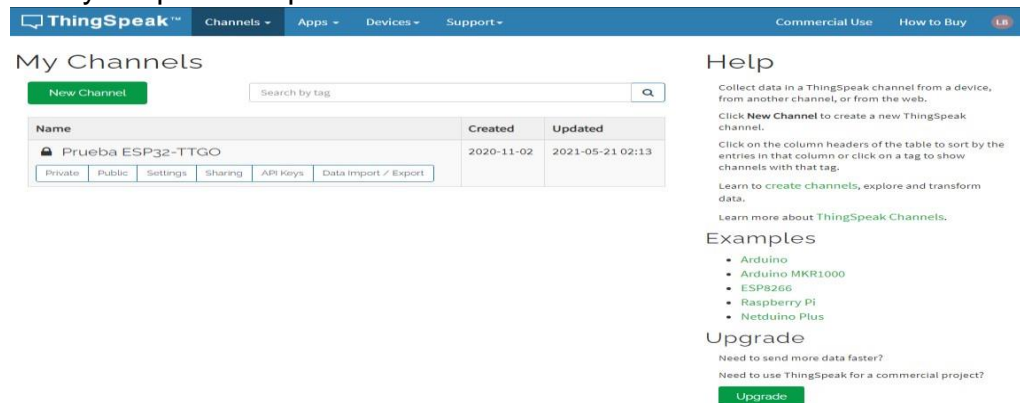
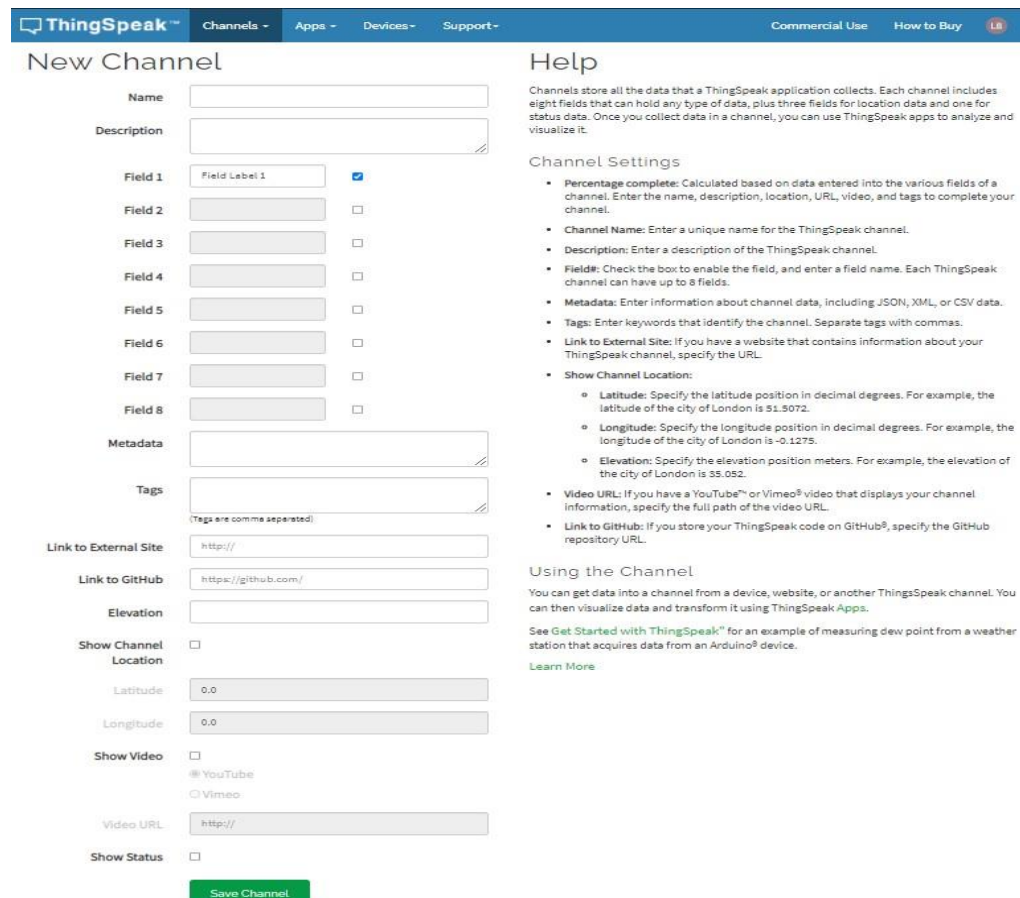


Figura 24 - Interfaz de usuario con control de la plataforma

Fuente: Captura tomada por los autores

En la Figura 24 se observa las opciones de manipular canales ya creados o permite la creación de nuevos.



*Figura 25 - Interfaz de creación de canal*

Fuente: Captura tomada por los autores

En la figura 25 se observa los datos que se requieren para la creación de canales con los distintos Feed los cuales equivalen a las variables que se van a usar en el proyecto para el este caso se encuentra, Temperatura, Humedad, Peso, Flujo y un contador que permite relacionar valores enviados con los valores recibidos. En la configuración de dichos canales tenemos que:

- **Porcentaje completado:** Calculado en base a los datos ingresados en los distintos feeds de un canal. Ingresar el nombre, la descripción, la ubicación, la URL, el video y las etiquetas para completar tu canal.
- **Nombre del canal:** ingrese un nombre único para el canal ThingSpeak.
- **Descripción:** introduzca una descripción del canal ThingSpeak.
- **Número de campo:** marque la casilla para habilitar el campo e ingrese un nombre de campo. Cada canal de ThingSpeak puede tener hasta 8 campos.
- **Metadatos:** introduzca información sobre los datos del canal, incluidos datos JSON, XML o CSV.
- **Etiquetas:** ingrese palabras clave que identifiquen el canal. Separa las etiquetas con comas.
- **Enlace a un sitio externo:** si tiene un sitio web que contiene información sobre su canal ThingSpeak, especifique la URL.
- **Mostrar ubicación del canal:**
  - **Latitud:** especifique la posición de latitud en grados decimales. Por ejemplo, la latitud de la ciudad de Londres es 51,5072.
  - **Longitud:** especifique la posición de la longitud en grados decimales. Por ejemplo, la longitud de la ciudad de Londres es 0,1275.
  - **Elevación:** especifique los metros de posición de elevación. Por ejemplo, la elevación de la ciudad de Londres es 35,052.
- **URL del video:** si tiene un video de YouTube <sup>TM</sup> o Vimeo® que muestra la información de su canal, especifique la ruta completa de la URL del video.
- **Enlace a GitHub:** si almacena su código ThingSpeak en GitHub®, especifique la URL del repositorio de GitHub.

En el listado anterior se realiza la presentación de las configuraciones generales del canal, una vez creado el canal se obtendrán cuadros de información relacional como se presentan en la Figura 25

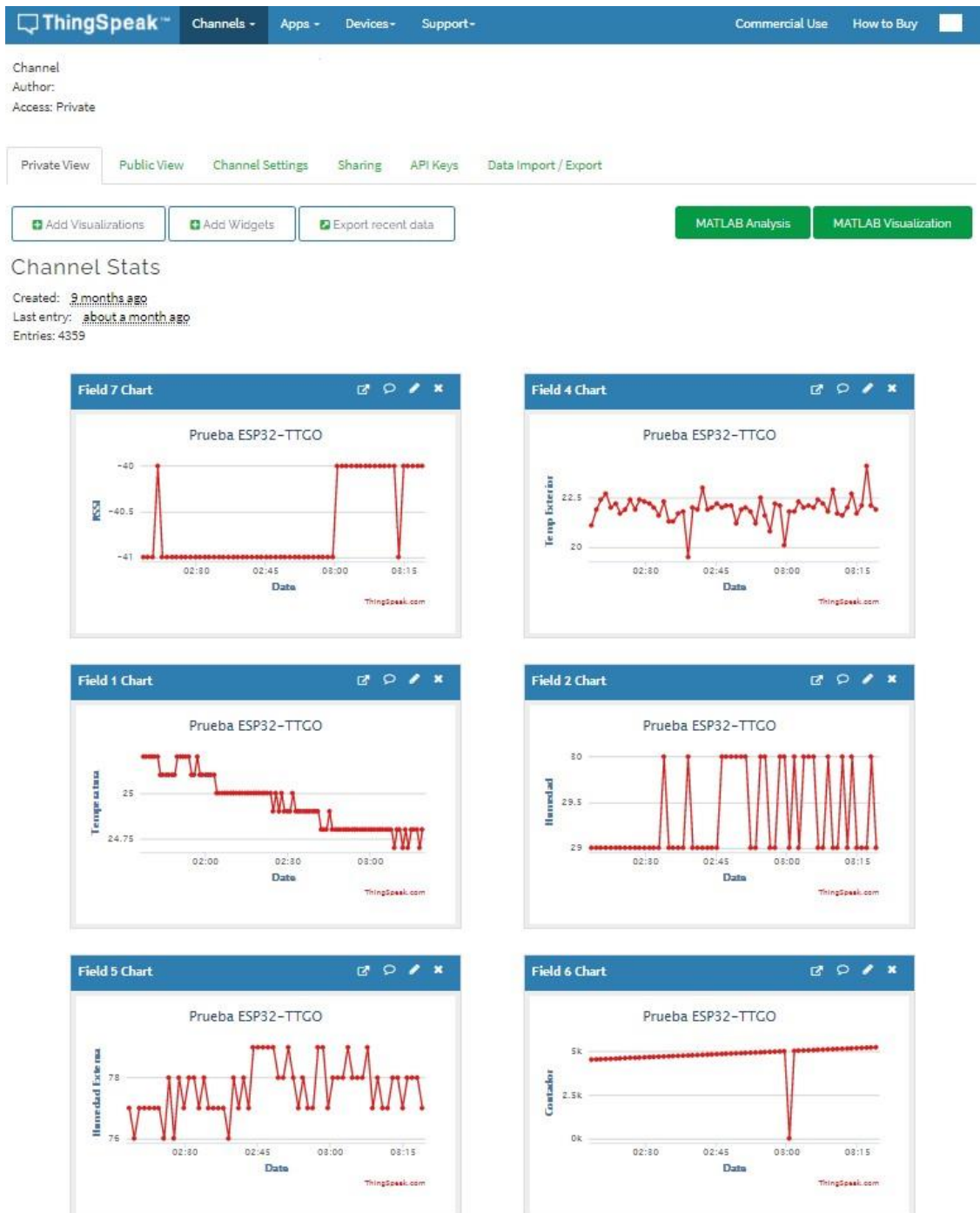


Figura 26 - Interfaz de visualización del canal

Fuente: Captura tomada por los autores

La Figura 26. hace referencia a algunos de los datos recibidos por el canal durante un lapso de tiempo del funcionamiento en toma de muestras y ubicado en la zona de instalación propuesta por los tesisistas y el apicultor tutor.

En la misma interfaz se presentan opciones de revisión del canal si se encuentra público o privado además de la configuración propia del canal, la opción de

compartir el canal mediante una URL pero los dos mas importantes en este proyecto se encuentra en la parte final de la fila, los cuales son los API Keys y Data Import/Export, primero se revisa los API Keys

ThingSpeak™ Channels Apps Devices Support Commercial Use How to Buy LB

Access: Private

Private View Public View Channel Settings Sharing API Keys Data Import / Export

### Write API Key

Key

[Generate New Write API Key](#)

### Read API Keys

Key

Note

[Save Note](#) [Delete API Key](#)

[Add New Read API Key](#)

### Help

API keys enable you to write data to a channel or read data from a private channel. API keys are auto-generated when you create a new channel.

### API Keys Settings

- **Write API Key:** Use this key to write data to a channel. If you feel your key has been compromised, click [Generate New Write API Key](#).
- **Read API Keys:** Use this key to allow other people to view your private channel feeds and charts. Click [Generate New Read API Key](#) to generate an additional read key for the channel.
- **Note:** Use this field to enter information about channel read keys. For example, add notes to keep track of users with access to your channel.

### API Requests

**Write a Channel Feed**

```
GET https://api.thingspeak.com/update?api_key=SW92JUZH3GZE9826&field=
```

**Read a Channel Feed**

```
GET https://api.thingspeak.com/channels/1214923/feeds.json?api_key=
```

**Read a Channel Field**

```
GET https://api.thingspeak.com/channels/1214923/fields/1.json?api_key=
```

**Read Channel Status Updates**

```
GET https://api.thingspeak.com/channels/1214923/status.json?api_key=
```

[Learn More](#)

Figura 27 - Interfaz opciones de canal

La Figura 27. presenta las Llaves (Keys) de ingreso al canal mediante la API la cual permite escribir o leer información contenida en el canal, en la parte derecha de la imagen se encuentras URL con el comando GET el cual permitiría el ingreso para la Escritura de los Feed del canal, Lectura de los Feed del Canal, La lectura del canal en general y por último el estado en el que se encuentra el canal.

Los API Keys en el caso del proyecto se utilizará el Write Keys que permite acceder al canal y mediante la librería que nos facilita el desarrollador se envía la información correspondiente de las variables leídas por las tarjetas de desarrollo usadas en dicho proyecto, para ellos se presenta la biblioteca y opciones suministradas mediante la librería de ThingSpeak para la IDE de Arduino con soporte a las tarjetas de desarrollo ESP32 y ESP8266.

Esta biblioteca permite a un Arduino u otro hardware compatible escribir o leer datos hacia o desde ThingSpeak, una plataforma de datos abierta para Internet de las cosas con análisis y visualización de MATLAB.

Aquí se encuentran ejemplos específicos de hardware. Pero para darle una idea de los ejemplos de uso para escribir y leer con un ESP8266 se muestran a continuación. La documentación completa también se muestra a continuación. ThingSpeak ofrece almacenamiento y análisis de datos gratuitos de datos numéricos o alfanuméricos con marca de tiempo. Los usuarios pueden acceder a ThingSpeak visitando <http://thingspeak.com> y creando una cuenta de usuario de ThingSpeak.

ThingSpeak almacena datos en canales. Los canales admiten un número ilimitado de observaciones con marca de tiempo (considérelas como filas en una hoja de cálculo). Cada canal tiene hasta 8 campos (considérelas como columnas en una hoja de cálculo). Vea este [video](#) para obtener una descripción general.

Los canales pueden ser públicos, donde cualquiera puede ver los datos, o privados, donde solo el propietario y los usuarios seleccionados pueden leer los datos. Cada canal tiene una clave API de escritura asociada que se utiliza para controlar quién puede escribir en un canal. Además, los canales privados tienen una o más claves de API para controlar quién puede leer desde el canal privado. No se requiere una clave API para leer de los canales públicos. Cada canal puede tener hasta 8 campos. Se crea un campo de forma predeterminada.

Puede visualizar y realizar análisis en línea de sus datos en ThingSpeak utilizando la versión incorporada de MATLAB, o usar la versión de escritorio de MATLAB para obtener una visión histórica más profunda.

Visite <https://www.mathworks.com/hardware-support/thingspeak.html> para obtener más información.

Las bibliotecas y ejemplos de dispositivos de partículas se pueden encontrar aquí: <https://github.com/mathworks/thingspeak-particle>

**Instalación de la Librería en el IDE de Arduino:** En el IDE de Arduino, elija Sketch / Incluir biblioteca / Administrar bibliotecas. Haga clic en la biblioteca ThingSpeak de la lista y haga clic en el botón Instalar.

1. Descargue el archivo ZIP (a continuación) en su máquina.
2. En el IDE de Arduino, elija Sketch / Incluir biblioteca / Agregar biblioteca Zip
3. Navegue hasta el archivo ZIP y haga clic en Abrir

### Lista de compatibilidad de Hardware

- Arduino / Genuino o compatible con WiFi Shield
- Arduino / Genuino o compatible con WiFi Shield 101 (use la biblioteca WiFi101 versión 0.13.0 o anterior).
- Arduino / Genuino o compatible con Ethernet Shield
- Arduino / Genuino o compatible con MKR ETH Shield
- Arduino MKR1000
- Arduino MKR1010
- Arduino VIDOR 4000
- Arduino GSM 14000
- Arduino Uno WiFi Rev2
- Arduino Yún (Rev1 y Rev2)
- Programación ESP8266 directamente (probado con SparkFun ESP8266 Thing - Dev Board y módulo NodeMCU 1.0)
- ESP8266 a través de comandos AT
- ESP32 (probado con SparkFun ESP32 Thing)

La biblioteca incluye varios ejemplos organizados por tipo de tablero para ayudarlo a comenzar. Se puede acceder a ellos en el menú Ejemplos> ThingSpeak del IDE de Arduino.

- **ReadField:** lectura de un canal público y un canal privado en ThingSpeak.
- **WriteSingleField:** escribir un valor en un solo campo en ThingSpeak.
- **WriteMultipleFields:** escribir valores en varios campos y estados en una transacción con ThingSpeak.
- **ReadMultipleFields:** lectura de valores de varios campos, estado, ubicación, marca de tiempo creada en un canal público en ThingSpeak
- **SecureConnect:** usar las funciones anteriores y conectarse de forma segura a ThingSpeak.

En este caso, lea desde un canal público y un canal privado con un ESP8266. El canal público es la temperatura (F) en la sede de MathWorks. El canal privado es un contador que se incrementa.

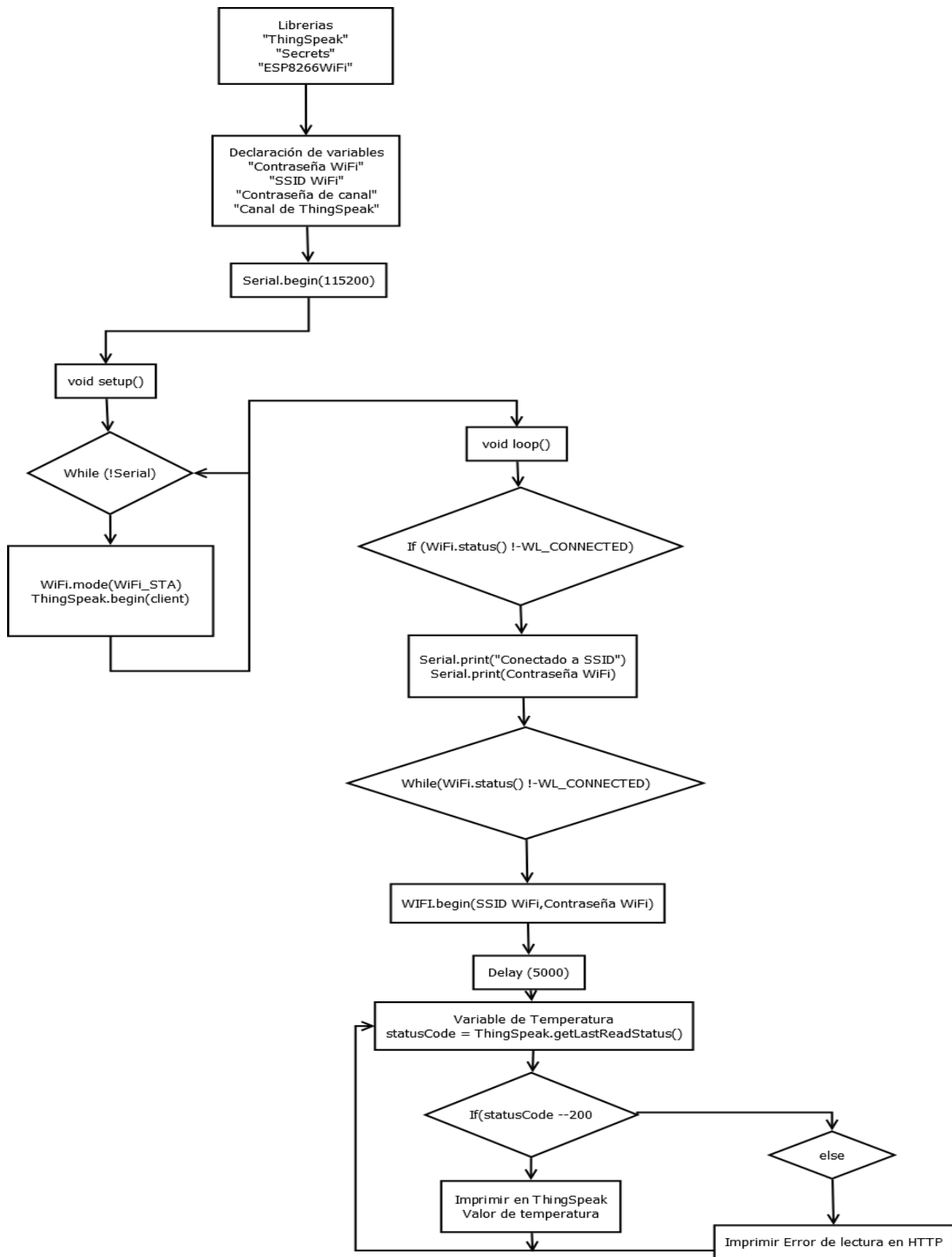


Figura 28 - código de ejemplo comunicación con ThingSpeak desde tarjeta de desarrollo ESP8266

Fuente: Captura tomada por los autores



Para el uso del código presentado en la Figura 28, se debe tener en cuenta lo presentado en el cuadro inferior el cual referencia los parámetros de uso más comunes en códigos utilizados en las distintas placas de desarrollo para lograr escribir la información deseada en un canal de ThingSpeak.

Parameter	Type	Description
channelNumber	unsigned long	Channel number
field	unsigned int	Field number (1-8) within the channel to write to.
value	int	Integer value (from -32,768 to 32,767) to write.
	long	Long value (from -2,147,483,648 to 2,147,483,647) to write.
	float	Floating point value (from -999999000000 to 999999000000) to write.
	String	String to write (UTF8 string). ThingSpeak limits this field to 255 bytes.
	const char *	Character array (zero terminated) to write (UTF8). ThingSpeak limits this field to 255 bytes.
writeAPIKey	const char *	Write API key associated with the channel. If you share code with others, do not share this key

*Tabla 2 - Tipo de documentación arduino*

Fuente: Documentación de ThingSpeak para IDE de Arduino

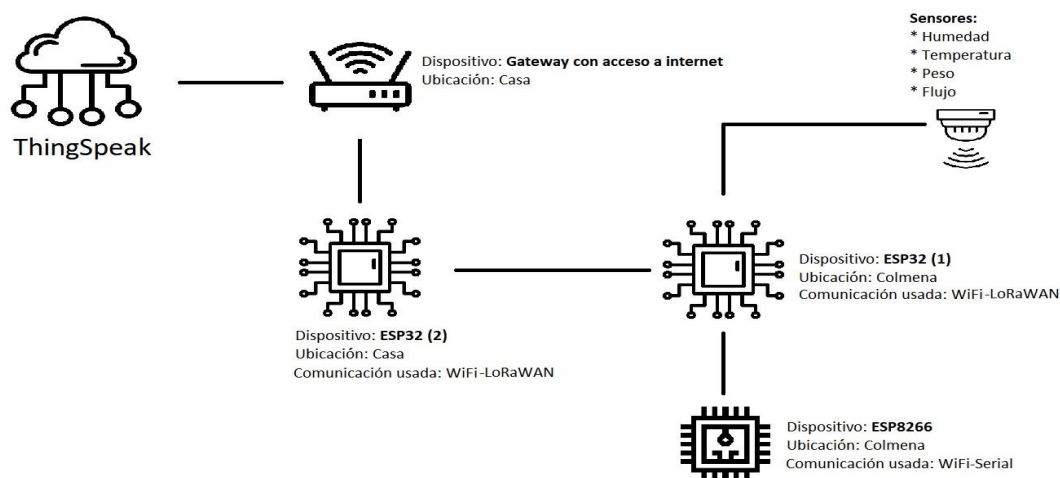
URL: <https://github.com/mathworks/thingspeak-arduino#documentation>

Se da lugar para resaltar que no es el único cuadro de información, pero si es el utilizado en el proyecto, caso de querer obtener mayor información el link se encuentra disponible en la parte inferior de cuadro.



### 8.3 ESQUEMÁTICO DEL FUNCIONAMIENTO COMPLETO DEL PROTOTIPO

La figura 32 resume el procedimiento completo desde el punto de origen en el que son censadas las variables. Los datos son recibidos y enviados a los dispositivos encargados de Gateway para ser almacenados en la SD y transmitidos a través de la red de internet, facilitando el acceso a los datos, la recuperación y el registro de ellos a la API tanto a dispositivos como a sitios web



29 - Esquemático Prototipo para monitoreo

Figura

Fuente: Elaborado por los Investigadores

## 9. DISEÑO Y DESARROLLO DEL HARDWARE

### 9.1 DISEÑO DE LA TARJETA DE CIRCUITO IMPRESO - PRINTED CIRCUIT BOARD (PCB)

El desarrollo de hardware en este proyecto es muy sencillo teniendo en cuenta que los micros ya vienen listos en sus propias PCB para los cuales solo fue necesario realizar una pequeña vácueta para las conexiones de masas y energía de los módulos.

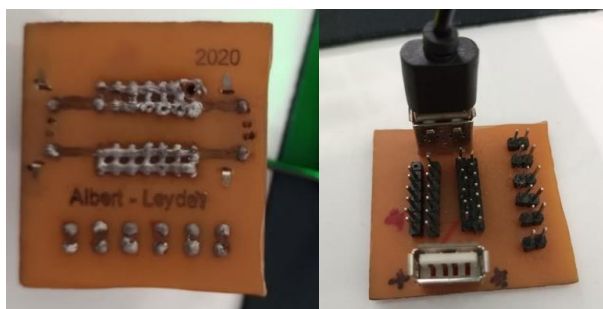


Figura 30 - Circuito de distribución de alimentación

Fuente: Elaborado por los Investigadores

## 9.2 MODULO TTGO

Cómo es mencionado anteriormente el módulo TTGO tiene un esp32 y un Shield de comunicación inalámbrica on board el cual con sus puertos seriales permite la conexión Lora y Wifi al mismo tiempo, por eso utilizando el esp32 se realiza la conexión de los sensores infrarrojos, temperatura y humedad por medio de los puertos digitales. Después este es conectado mediante Wifi al Modulo Arduino Nodemcu V3 Esp8266 que está configurado y programado para almacenar los datos censados por el TTGO y el peso de la colmena en una memoria micro SD. El TTGO a su vez envía los datos por medio de la conexión Lora a una distancia de 1.5Km utilizando la frecuencia libre de 915MHz a otro TTGO que recibe la información y la sube directamente a la API utilizada por medio de Wifi.

La efectividad de la transmisión a larga distancia es posible al diseño del dispositivo Lora y su antena la cual debido a su patrón de radiación y ganancia, cumpliendo así con los requerimientos para la red de comunicación.

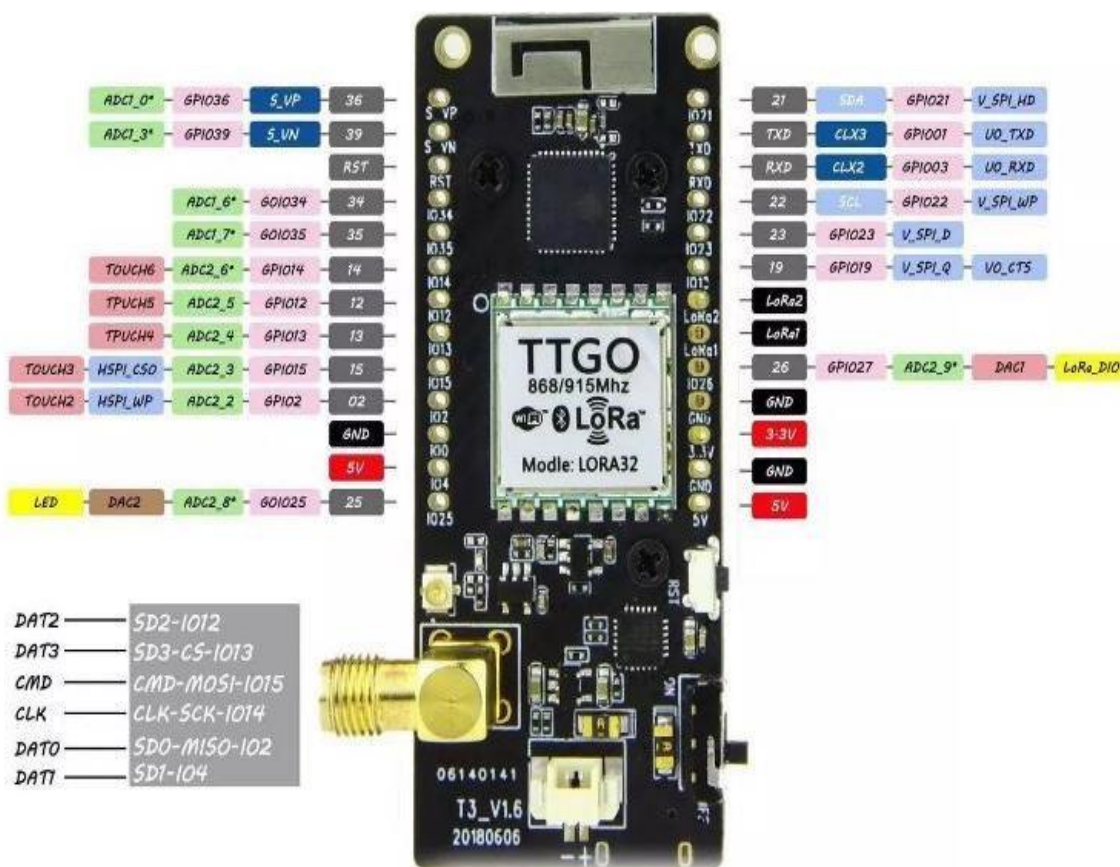


Figura 31 - Modulo TTGO Lora32 utilizado

Fuente: <https://altronics.cl/ttgo-lora32-t3-esp32-915mhz>

### Datos técnicos Antena:

Frecuencia (MHz): 868 – 915

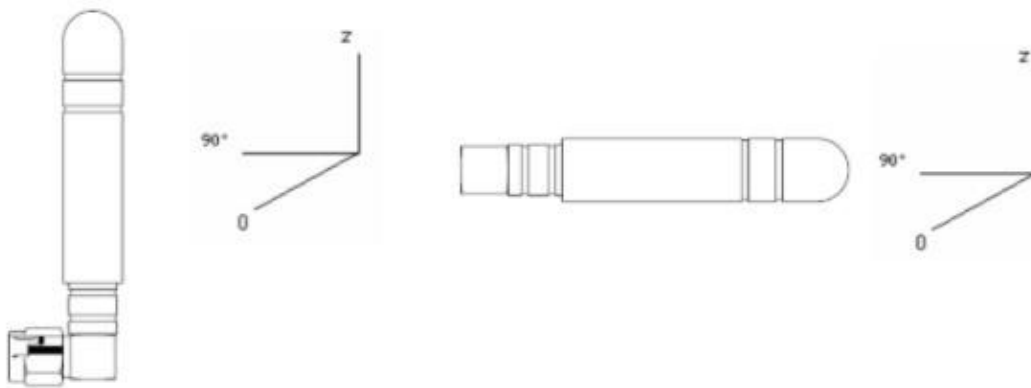
Ganancia: 3 dBi

Conector: SMA macho



*Figura 32 - Antena para la transmisión del TTGO Lora32*

Fuente: <https://uelectronics.com/producto/ttgo-lora32-v2-1-915mhz/>



*Figura 33 - Plano de radiacion TTGO Lora32*

Fuente: <https://www.sigmaelectronica.net/producto/ant-lora32/>

Frecuencia (MHZ)	Plano H			Plano E1		
Unidad (dBi)	Máximo	Mínimo	Promedio	Máximo	Mínimo	Promedio
868	1.52	-5.61	-1.64	-0.45	-10.65	-4.09
880	-0.6	-9.54	-2.94	-1.88	-12.53	-4.42
915	-0.42	-4.52	-2.06	-1.36	-10.19	-4.47

*Tabla 3 - Ganancia en dBi para las frecuencias de operación de la antena en el plano H y plano E1*

Fuente: Hoja de Datos de la Antena

De la figura y tabla anteriores, si la antena se ubica en dirección paralela al eje z (plano H) y teniendo en cuenta la frecuencia de operación para la red Lora (915 MHz) se tiene una ganancia promedio de un poco más de 3 dBi. Esta ganancia está determinada respecto a una antena isotrópica (ideal, con un patrón de propagación que forma una esfera perfecta y ganancia de 0 dBi).

Para calcular qué tan fuerte es la antena (transmisión-recepción) respecto a una antena isotrópica en una escala lineal se tiene la siguiente ecuación:

$$G = (10^{\frac{G(dBi)}{10}}) = (10^{\frac{-3}{10}})$$

$$G = 0,43$$

La fuerza de transmisión y recepción de la antena es aproximadamente 0.43 veces comparada con la antena isotrópica en el plano H.

La siguiente figura ilustra el patrón de radiación de la antena para tres de sus frecuencias de operación respecto al patrón de radiación de una antena isotrópica en el plano H, se puede observar que el patrón de radiación para 915 Mhz estaría enmarcado aproximadamente entre líneas roja y verde. De esta manera, se puede afirmar que la antena radía de la misma manera en todas las direcciones.

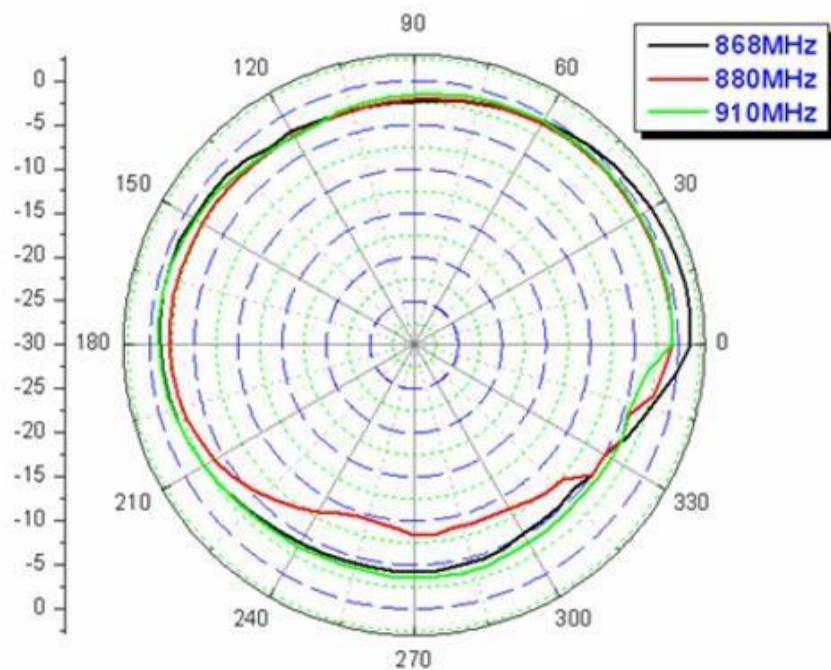


Figura 34 - Patrón de radiación para la antena Lora32 en el plano H

Fuente: Lora Antenna-868-915.pdf

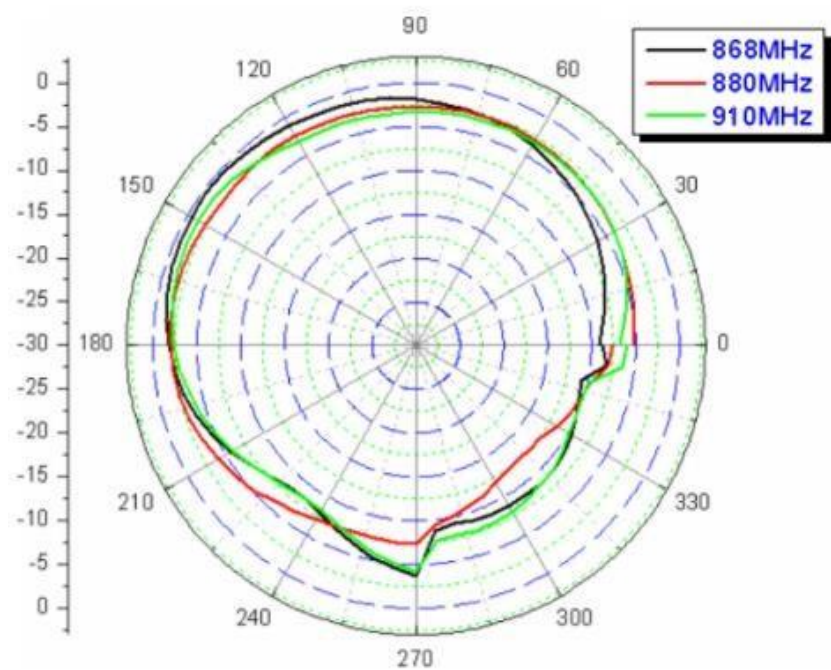


Figura 35 - Patrón de radiación para la antena Lora32 en el plano E1

Fuente: Lora Antenna-868-915.pdf

Para el plano E1, el patrón de radiación de la antena muestra que el lóbulo se hace más grande y uniforme en sólo la mitad del plano, es decir, hay una mayor directividad. Para transmitir o recibir usando este plano se requiere de conocer en qué dirección se encuentran las antenas de estación base para una mayor eficiencia.

La potencia de transmisión configurada es de 20 dBm, la ecuación para determinar su valor en mW es la siguiente:

$$P(mW) = (1W * \frac{10^{\frac{P(dBm)}{10}}}{1000}) = (1W * \frac{10^{\frac{20}{10}}}{1000})$$

$$P = 0,216mW$$

Este será el consumo del dispositivo cada vez que se envíe un mensaje, aproximadamente cada 11 minutos, durante la mayor parte del tiempo el dispositivo entra en modo de consumo muy bajo (Corriente de 120mA).<sup>34</sup>

### 9.3 MODULO esp8266

El ESP8266 tiene potentes capacidades a bordo de procesamiento y almacenamiento que le permiten integrarse con sensores y dispositivos específicos de aplicación a través de sus GPIOs con un desarrollo mínimo y carga mínima durante el tiempo de ejecución. Su alto grado de integración en el chip permite una circuitería externa mínima, y la totalidad de la solución, incluyendo el módulo está diseñado para ocupar el área mínima en un PCB, por esto fue utilizado y configurado como se describe anteriormente para recibir y almacenar los datos obtenidos por el TTGO y el peso de la colmena en una micro SD, se optó por utilizar debido a la fácil configuración en comunicación por wifi con el TTGO pues en conexiones seriales o directas por cables presentaban errores y pérdida de datos. Aparte de esto cuenta con la independencia de código pues si llega a presentar un fallo puede reiniciarse fácilmente sin interferir con el funcionamiento del TTGO por su independencia de código.

---

<sup>34</sup> [http://www.salleurl.edu/semipresencial/ebooks/ebooks/ebook\\_teorias\\_antenas.pdf](http://www.salleurl.edu/semipresencial/ebooks/ebooks/ebook_teorias_antenas.pdf)



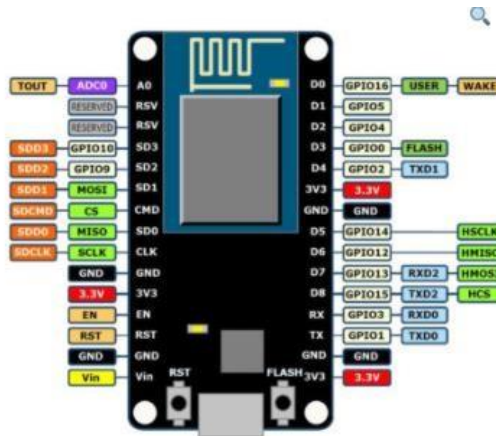


Figura 36 - Montaje Completo Prototipo de monitoreo

Fuente: <https://electronilab.co/tienda/nodemcu-board-de-desarrollo-con-esp8266-wifi-y-lua/>

## 9.4 MODULO LECTOR MICRO SD

Este módulo es encargado de enlazar la micro SD con el esp8266 para almacenar los datos obtenidos por los sensores. Esta configuración es realizada a través del chip LVC125A y un regulador de voltaje ASM1117 que en conjunto forman parte del módulo.

### Características

Voltaje de funcionamiento: 4.5 ~ 5.5V

VCC;Actual: 0.2 ~ 200mA

Nivel eléctrico Interfaz: 3.3V / 5V; A bordo 3.3V circuito regulador de voltaje

Soporta Micro SD de hasta 2 GB, Micro SDHC de hasta 32GB

Conduce: GND, VCC, MISO, MOSI, SCK, CS

Interfaz estándar SPI, 4 agujeros de montaje M2 tornillos 2.2mm.

Dimensiones: 1.81 in x 0.94 in x 0.28 in (4.6 cm x 2.4 cm x 0.7 cm) Peso:

0,18 oz (5 g)

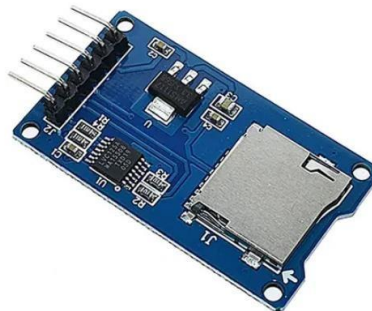


Figura 37 - Modulo lector micro SD

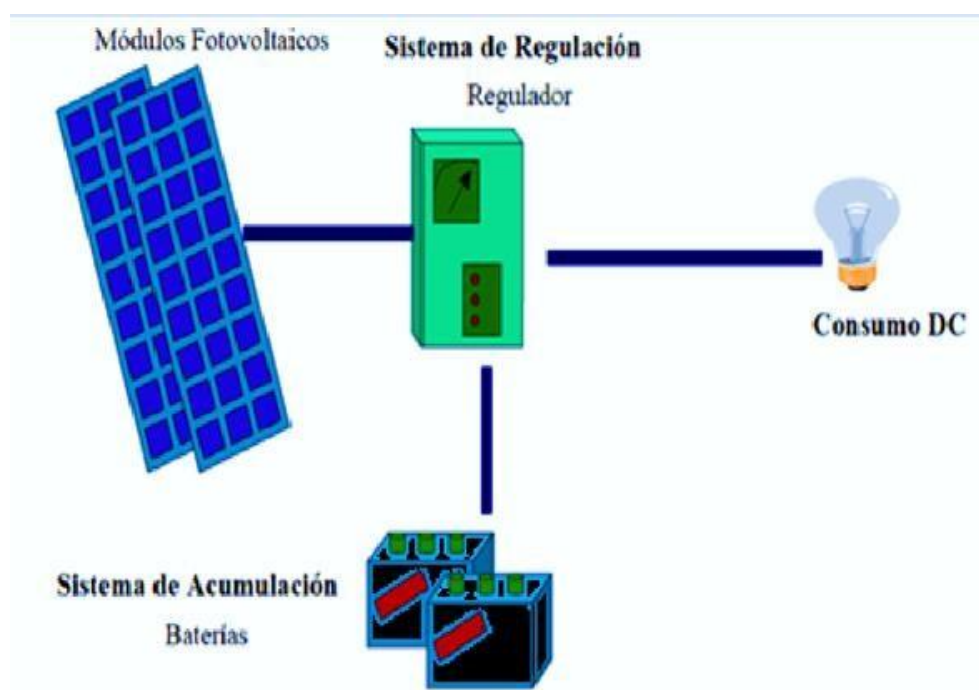
Fuente: <https://www.luisllamas.es/tarjeta-micro-sd-arduino/>

## 9.5 ETAPA DE ALIMENTACIÓN

### 9.5.1 Conversión de la energía solar

Para la conversión de la energía solar se va a usar un panel fotovoltaico es decir, vamos a utilizar un panel solar, de tal manera que éste pueda cumplir con las especificaciones de potencia que va a consumir el sistema como tal, teniendo en cuenta que dicha energía proveniente del sol pueda ser almacenada en una batería, ésta guarda la suficiente energía, cuando por alguna razón el panel no esté funcionando pueda soportar el consumo potencial mientras se arregla el posible daño que pueda tener el panel solar por otro lado, es necesario regular dicha tensión eléctrica, para ello se va a seleccionar un regulador de carga solar la cual se va poner entre el campo fotovoltaico y el campo de batería y básicamente se va encargar de controlar el flujo de energía que circula entre ambos equipos.

Para esto se tiene el siguiente diagrama:



*Figura 38 - Esquema etapa de alimentación*

Fuente: modificado por los investigadores



### 9.5.2 Diseño etapa de alimentación

Lo primero es el cálculo de la etapa de potencias basándonos en los datasheet de cada componente utilizado que necesitara la energía fotovoltaica.

Componente	Voltaje(V)	Corriente(mA)	Potencia(mW)
Sensor DHT11	5	2.5	12.5
Sensor DHT11	5	2.5	12.5
Celda de carga	5	5	25
Celda de carga	5	5	25
Celda de carga	5	5	25
Celda de carga	5	5	25
TTGO	3.3	70	231
Antena			216
Esp8266	3.3	250	825
Sensor Proximidad	3	10	30
Sensor Proximidad	3	10	30
<b>Total</b>		<b>365</b>	<b>1457</b>

*Tabla 4 - Consumo de dispositivos empleados*

De acuerdo a la pérdida de corriente en los elementos que vamos a utilizar se ha decidido que son cerca de un 15% por factores externos al sistema, por tanto, la corriente total se va a multiplicar por 1,15 es decir que la corriente total que el sistema va a necesitar es por tanto  $365\text{mA} \times 1,15 = 638,75\text{ mA}$  es decir menos de 1A. Ahora si tomamos como base para el sistema un voltaje de 12V debemos utilizar un panel solar cuyo funcionamiento sea mayor a los 12V y 1A que necesita el sistema, por tanto, el panel solar que vamos a utilizar es elegido de acuerdo a sus especificaciones técnicas descritas en la selección paneles fotovoltaicos que corresponda a un voltaje de 18V y un corriente de 1,67A.

Teniendo en cuenta dicho factor de seguridad del 15%, el sistema va a consumir  $1161\text{mW} \times 1,15 = 2032\text{mW}$ , se aconseja doblar la potencia total para prolongar el tiempo de funcionamiento sin energía solar o en días de poca radiación que es un poco menos de 5000mW, es decir, 5W cada hora.  
Por tanto,

$$\text{Consumo potencia diario} = 5\text{ W} * 24\text{ horas} = 120\text{ Wdia}$$

por otro lado, la batería que elegiremos será de 12V y 18 A de acuerdo a las especificaciones técnicas de nuestros componentes, por tanto,

$$Potencia\ batería = 12V * 18Ahora = 216\ Whora$$

Ahora vamos a obtener la autonomía que nos va a ofrecer la batería cuando el panel solar por diferentes circunstancias no se encuentre funcionando, por tanto,

$$Autonomía = \frac{216Whora}{5W} = 43,2\ horas$$

Ahora vamos a justificar el regulador que escogimos:

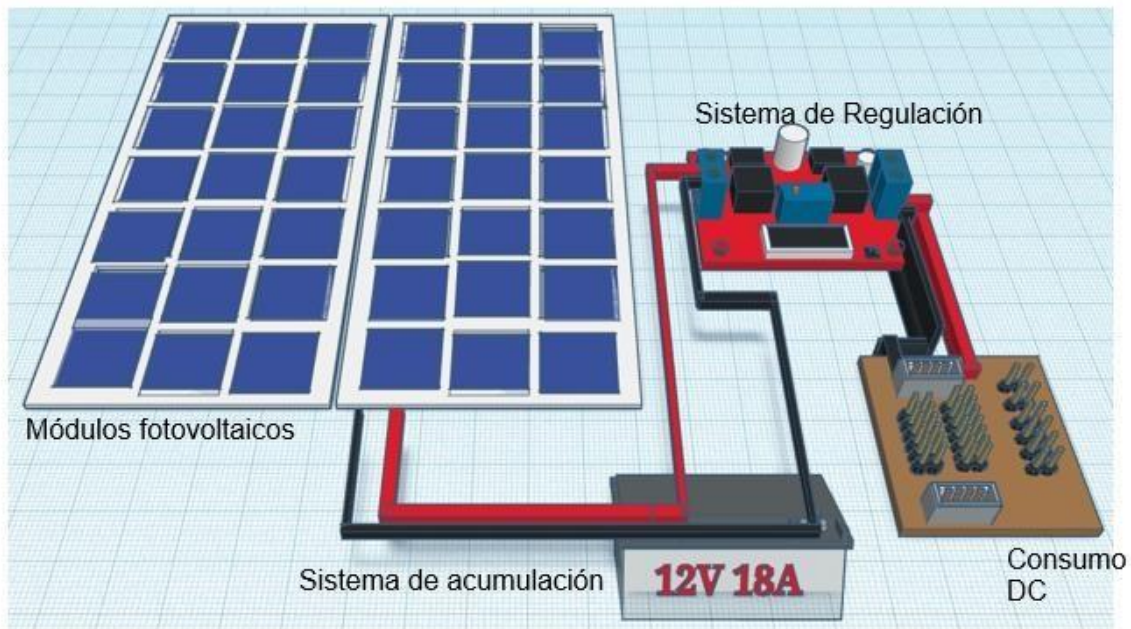


Figura 39 - Esquema de corriente

Fuente: elaborado por los investigadores

$$Corriente\ proveniente\ del\ panel\ solar = 1,67A$$

Observando que la corriente proveniente del panel solar es de 1,67A de acuerdo a las especificaciones técnicas del panel solar y de acuerdo a la corriente de salida a la carga de 0,365A y con una batería de 12V debemos utilizar un regulador que esté por encima de estos valores de corriente que va a soportar dicho regulador, por tanto, con las respectivas especificaciones del regulador que se va a utilizar que corresponde a una corriente de 5A y salida de 3A.

### 9.5.2.1 Paneles fotovoltaicos

Panel Solar 30w Policristalino (18 V - 1.667 A) Psp30w



*Figura 40 - Panel solar utilizado*

Fuente: foto realizada por los investigadores

Especificaciones técnicas:

- Potencia máxima: 30 vatios. - Voltaje: 18 voltios. (Ajustable 14v-18v) - Corriente: 1.667 Amperios.
- Voltaje con circuito abierto: 23.1 voltios.
- Tecnología: Policristalino.
- Número de celdas: 36 - Temperatura de funcionamiento: -40 +85 grados centígrados.
- Equipo para exterior, puede estar expuesto al sol y al agua.
- Producto garantizado, vida útil del panel solar 25 años.
- 67x35x2.5cm-2.8kg

### 9.5.2.2 Batería

Batería Sellada 12 Voltios 18 Amperios / 20hrs 12v 18ah: Batería Sellada Fuli Battery 12V-18AH Ref. FL12180GS



Referencia	Tipo	Voltaje nominal (VDC)	Capacidad Amp/h	Tipo de Terminal	Largo (mm)	Ancho (mm)	Alto total (mm)	Peso (kg)
FL640GS	VRLA	6	4	F1	70	47	107	0,72
FL1250HR	VRLA	12	5	F1	90	70	107	1,75
FL1275GS	VRLA	12	7,5	F1	151	65	101	2,32
FL1290HR	VRLA	12	9	F1	151	65	101	2,7
FL12120GS	VRLA	12	12	F1	151	98	101	3,8
FL12180GS	VRLA	12	18	I - M5	181	77	167	5,5
FL12260GS	VRLA	12	26	I - M5	166	175	125	8,1
FL12350GS	VRLA	12	35	I - M5	195	130	164	11,2
FL12400GS	VRLA	12	40	I - M5	197	165	170	12,97
FL12550GS	VRLA	12	55	I - M6	230	138	213	18
FL12800GS	VRLA	12	80	I - M6	258	169	212	21,7
FL12900GS	VRLA	12	90	I - M6	306	169	213	26,7
FL121000GS	VRLA	12	100	I - M8	330	172	222	31,5

Figura 41 - Referencia batería

Fuente: <https://spectronic.com.co/wp-content/uploads/2019/09/FULIBATTERYGS105-FL12180.pdf>

### 9.5.2.3 Regulador LM2596

LM2596 es un regulador de voltaje por modulación de ancho de pulso, por tratarse de una fuente de alimentación conmutada su eficiencia es significativamente mayor en comparación con los populares reguladores lineales de tres terminales, especialmente con tensiones de entrada superiores.

El módulo LM2596 es capaz de conducir una corriente de hasta 3A. Maneja una carga con excelente regulación de línea y bajo voltaje de rizado. Este dispositivo está disponible con voltaje de salida ajustable.

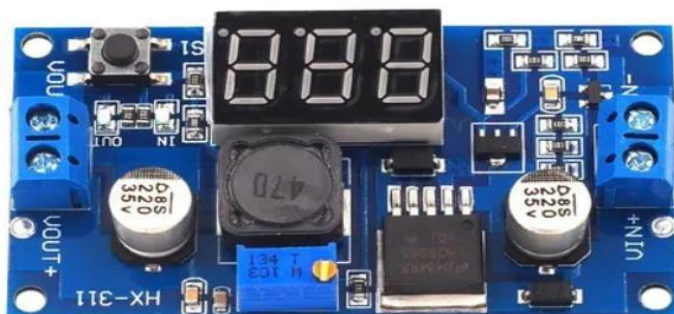




Figura 42 - Regulador utilizado

Fuente: foto realizada por los investigadores

## 9.6 Ubicación espacial de la colmena

El ideal ambiente de una colmena de abejas se basa en que no debe ser un lugar ruidoso y sin malos olores. Es evidente que no se puede ubicar en un lugar urbanístico, tanto por el bienestar de las abejas como para la gente que puede circular cerca de una colmena. En este sentido, la colmena se ubicará en una zona rural, en donde no haya mucho ruido de automóviles, camiones o demás que no dejen fluir el normal comportamiento de las abejas y la colmena.

En este caso es ubicada en una vereda cercana a Garzón llamada Majo, la colmena con el prototipo es ubicada a 1.5 Km del TTGO Gateway

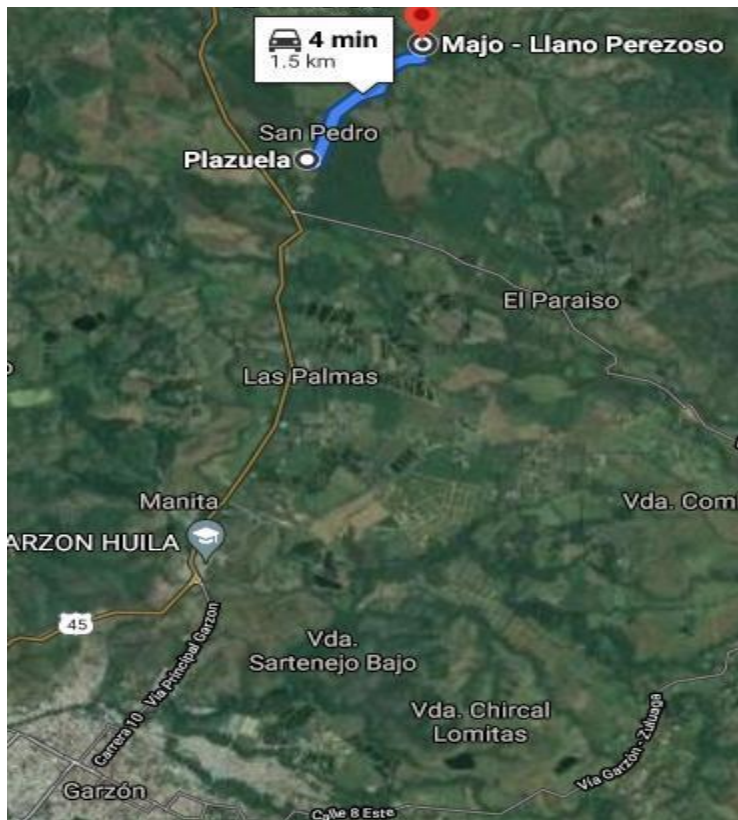


Figura 43 - Localización geográfica colmena

Fuente: Google maps

En cuanto al uso de las celdas solares para la recarga de la batería, la vereda recibe entre 4.0 y 4.5 KWh/m<sup>2</sup> de radiación solar, como se muestra en el siguiente mapa solar, lo que es ideal para el uso de las celdas y se pueda recargar correctamente la batería

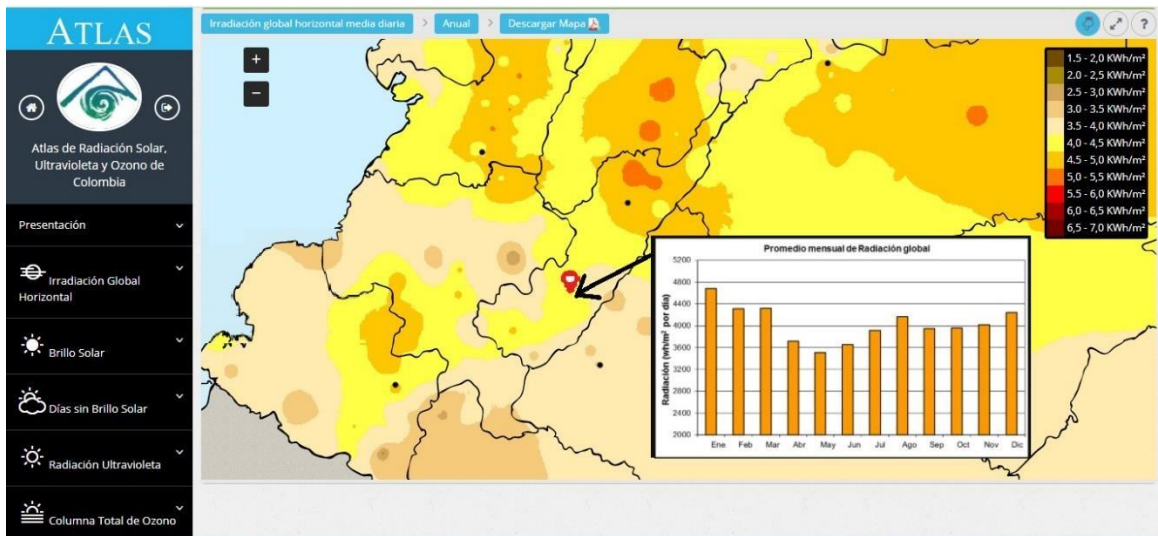


Figura 44 - Mapa Solar

Fuente: Atlas



Figura 45 - Montaje en completo y funcionamiento.

Fuente: Elaborado por los Investigadores

## 10. RESULTADOS Y ANÁLISIS

El Módulo de Monitoreo Apícola fue ensamblado y dispuesto sobre una colmena apícola para realizar pruebas en laboratorio. Anteriormente cada uno de los elementos fue probado de forma individual; sin embargo, se realizaron prueba con todos los elementos del sistema. El sistema tomó muestras continuamente de forma automática y al mismo tiempo mostrándolas en la API seleccionada.

Las dos salidas fueron probadas al tiempo, simulando el desplazamiento de varias abejas en un instante dado. Todos los movimientos fueron satisfactoriamente registrados. Al mismo tiempo que se realizaban las pruebas de simulación de ingreso y salida de varias abejas, se modificaron los estados de los sensores de proximidad. Cada evento fue registrado y almacenado, dentro de las pruebas realizadas, la colmena fue abierta, cerrada y manipulada (movida en diferentes direcciones). Tras realizar los procedimientos anteriores se verificó que el sistema hubiese registrado todos los eventos, que la información fuese visualizada en la pantalla del TTGO y que los datos fueron almacenados y enviados al servidor según el requerimiento.

Como paso a seguir, se llevo el sistema a campo y se tuvo en funcionamiento sin tener abejas, para poder verificar su eficiencia energética antes de comenzar el monitoreo apícola. Sin tener problemas en el envío y recepción de datos en días con poca radiación solar, se llevo a cabo el traspaso de las abejas a nuestro prototipo donde se empezó a almacenar y registrar los datos de todos los sensores, teniendo así los resultados obtenidos a continuación.

Los datos se dividieron en dos archivos de excel, unos que son almacenados en la SD y los que son almacenados en el servidor, esto con el fin de realizar comparaciones entre si y verificar el funcionamiento o perdida de estos según las variaciones climáticas (evidentes en la temperatura y humedad) y el RSSI, después estos datos fueron analizados con Python y sus herramientas de análisis de datos para obtener los archivos mas limpios y poder realizar mejores análisis y conclusiones.

Se agrego también un contador para tener una referencia de sincronía con los datos almacenados y enviados.

```
feeds.describe()
```

	entry_id	Temperatura	Humedad	Temp_Ext	Hum_Ext	Peso	Flujo	RSSI	Contador
<b>count</b>	4359.000000	4359.000000	4359.000000	4359.000000	4359.000000	4359.000000	4.359000e+03	4359.000000	4.359000e+03
<b>mean</b>	2180.000000	33.773526	16.957330	24.996306	70.980500	14.584331	1.895874e+07	-41.326910	9.504131e+06
<b>std</b>	1258.479241	1.593606	2.480491	4.722386	14.806379	1.963545	1.847943e+07	1.416253	9.250055e+06
<b>min</b>	1.000000	-2.100000	-16.000000	0.000000	0.000000	10.000000	7.800000e+01	-115.000000	3.800000e+01
<b>25%</b>	1090.500000	33.000000	15.000000	21.800000	64.000000	12.800000	1.194956e+06	-42.000000	6.092065e+05
<b>50%</b>	2180.000000	33.900000	17.000000	23.800000	73.000000	15.300000	1.671633e+07	-42.000000	8.390653e+06
<b>75%</b>	3269.500000	34.600000	19.000000	26.900000	82.000000	16.200000	3.150077e+07	-41.000000	1.579769e+07
<b>max</b>	4359.000000	36.800000	24.000000	47.800000	92.000000	16.600000	5.513216e+07	-39.000000	2.759544e+07

*Tabla 5 - Datos obtenidos en la SD*

Fuente: generada en jupyter por los investigadores

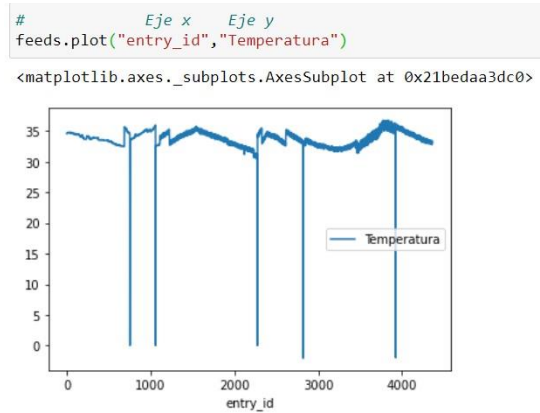


Figura 46 - Grafico de temperatura dentro de la colmena

Fuente: generada en jupyter por los investigadores

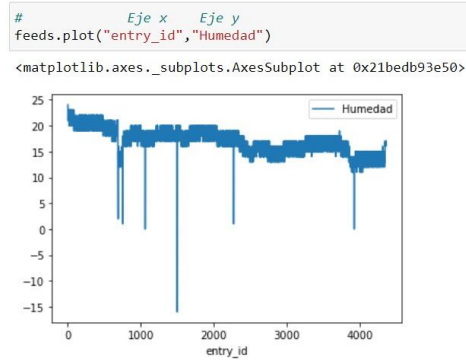


Figura 47 - Gráfico de humedad en la colmena

Fuente: generada en jupyter por los investigadores

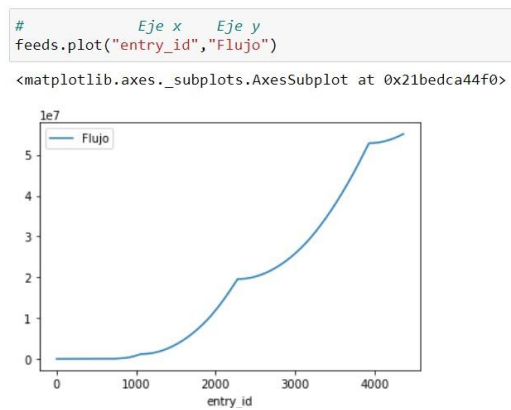


Figura 48 - Gráfico de flujo relativo en la colmena

Fuente: generada en jupyter por los investigadores



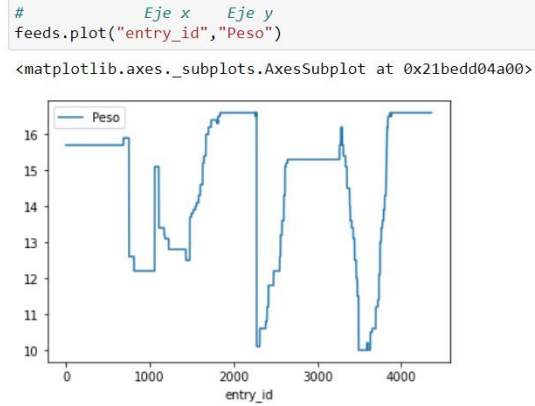


Figura 49 - Gráfico de peso de la colmena

Fuente: generada en jupyter por los investigadores

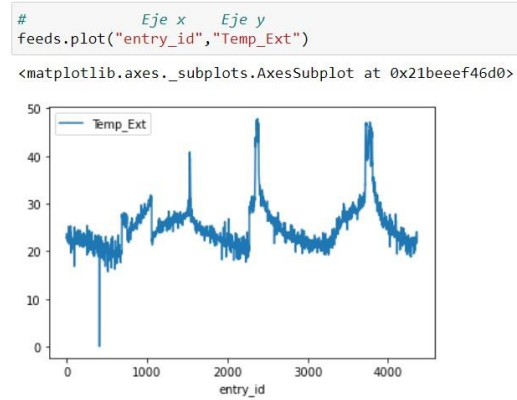


Figura 50 - Gráfico de temperatura externa

Figura

Fuente: generada en jupyter por los investigadores

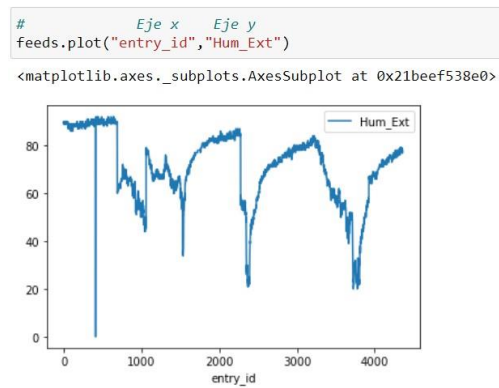
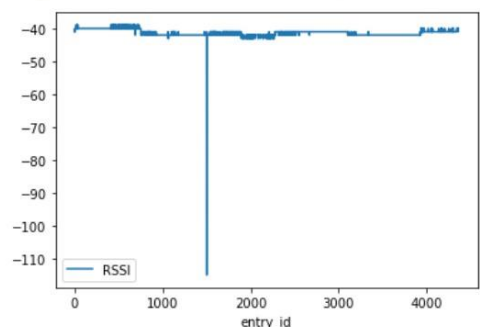


Figura 51 - Gráfico de humedad externa

Figura

Fuente: generada en jupyter por los investigadores

```
# Eje x Eje y
feeds.plot("entry_id", "RSSI")
<matplotlib.axes._subplots.AxesSubplot at 0x21bede0ef40>
```



52 - Gráfico RSSI

Figura

Fuente: generada en jupyter por los investigadores

```
# Eje x Eje y
feeds.plot("entry_id", ["Contador"])
<matplotlib.axes._subplots.AxesSubplot at 0x21beddb6d0>
```

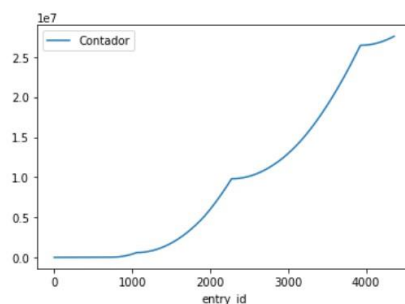


Figura 53 - Gráfico contador automático para guía

Fuente: generada en jupyter por los investigadores

Aplicando un análisis estadístico inicial a nuestra base de datos obtenemos que:

Valores máximos en las variables:

(36.8, 24, 55132161, 16.6, 47.8, 92, -39, 27595436)

Valores mínimos en las variables:

(-2.1, -16, 78, 10.0, 0.0, 0, -115, 38)

Variables (Temperatura, Humedad, Flujo, Peso, Temp\_Ext, Hum\_Ext, RSSI, Contador)

Analizando estos valores almacenados, se obtuvieron datos fuera de los rangos normales, estos datos no coinciden con las variables del lugar, por esto para realizar los análisis comparativos es necesario primero realizar un tratamiento o depuración de datos erróneos y datos no enviados, se realiza en jupyter con la ayuda de la

librería numpy y pandas para buscar y eliminar las filas correspondientes a datos que tengan valores “0” o “none”.

Después de la depuración tenemos datos con valores reales dentro de los parámetros de variación aplicamos nuevamente el análisis estadístico inicial y obtenemos los siguientes gráficos de resultados:

```
dt.describe()
```

	entry_id	Temperatura	Humedad	Temp_Ext	Hum_Ext	Peso	Flujo	RSSI	Contador
<b>count</b>	4351.000000	4351.000000	4351.000000	4351.000000	4351.000000	4351.000000	4.351000e+03	4351.000000	4.351000e+03
<b>mean</b>	2180.921397	33.812572	16.983682	25.000988	71.002758	14.584693	1.897053e+07	-41.309354	9.510042e+06
<b>std</b>	1258.487277	1.079244	2.368560	4.710365	14.776713	1.963449	1.847902e+07	0.871728	9.249847e+06
<b>min</b>	1.000000	30.600000	10.000000	15.700000	20.000000	10.000000	7.800000e+01	-43.000000	3.800000e+01
<b>25%</b>	1092.500000	33.000000	15.000000	21.800000	64.000000	12.800000	1.196446e+06	-42.000000	6.099525e+05
<b>50%</b>	2181.000000	33.900000	17.000000	23.800000	73.000000	15.300000	1.674520e+07	-42.000000	8.405090e+06
<b>75%</b>	3270.500000	34.600000	19.000000	26.900000	82.000000	16.200000	3.152548e+07	-41.000000	1.581005e+07
<b>max</b>	4359.000000	36.800000	24.000000	47.800000	92.000000	16.600000	5.513216e+07	-39.000000	2.759544e+07

Tabla 6 - Datos obtenidos en la SD después de la limpieza de datos

Fuente: generada en jupyter por los investigadores

```
dt.plot("entry_id", "Temperatura")
```

<matplotlib.axes.\_subplots.AxesSubplot at 0x21bf18729d0>

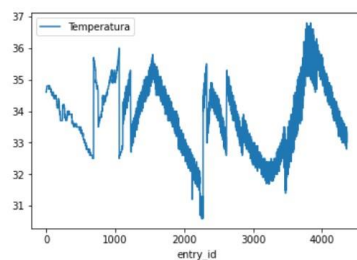


Figura 54 - Gráfico de temperatura interna en la colmena después de la limpieza de datos

Fuente: generada en jupyter por los investigadores

```
dt.plot("entry_id", "Humedad")
```

<matplotlib.axes.\_subplots.AxesSubplot at 0x21bf1bcd340>

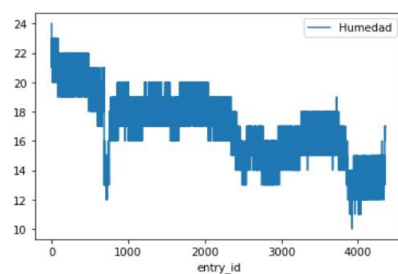


Figura 55 - Gráfico de humedad interna en la colmena después de la limpieza de datos

Fuente: generada en jupyter por los investigadores

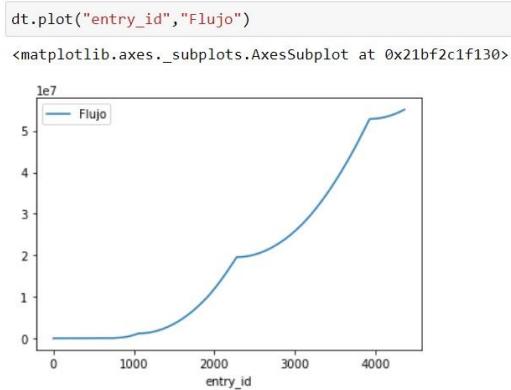


Figura 56 - Gráfico de flujo relativo en la colmena después de la limpieza de datos

Fuente: generada en jupyter por los investigadores

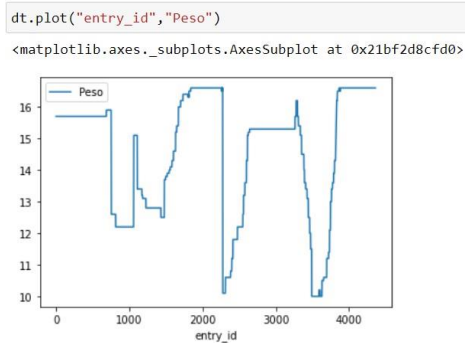


Figura 57 - Gráfico de peso en la colmena después de la limpieza de datos

Fuente: generada en jupyter por los investigadores

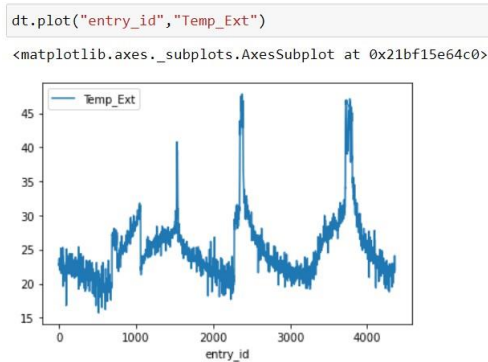


Figura 58 - Gráfico de temperatura externa después de la limpieza de datos

Fuente: generada en jupyter por los investigadores

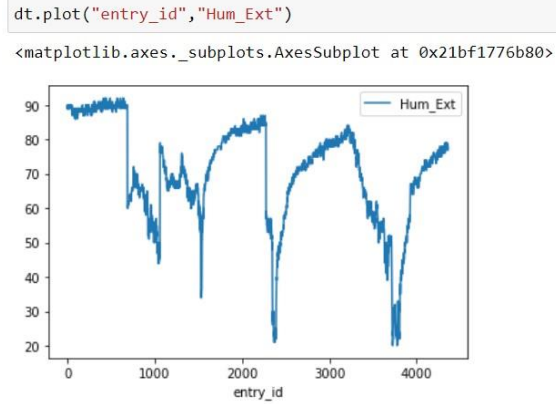
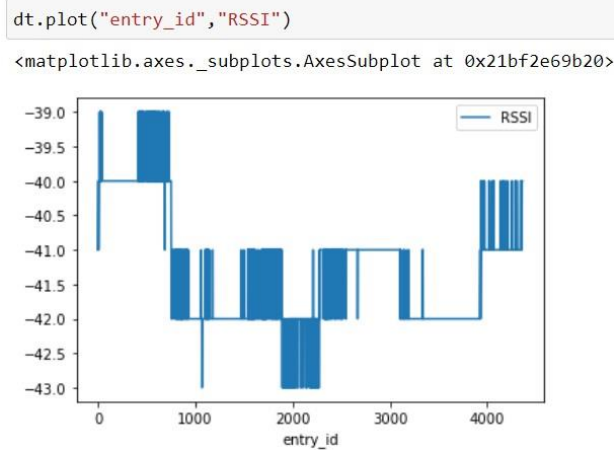


Figura 59

- Gráfico de la humedad externa después de la limpieza de datos

Fuente: generada en jupyter por los investigadores



Figura

60 - Gráfico de RSSI después de la limpieza de datos

Fuente: generada en jupyter por los investigadores

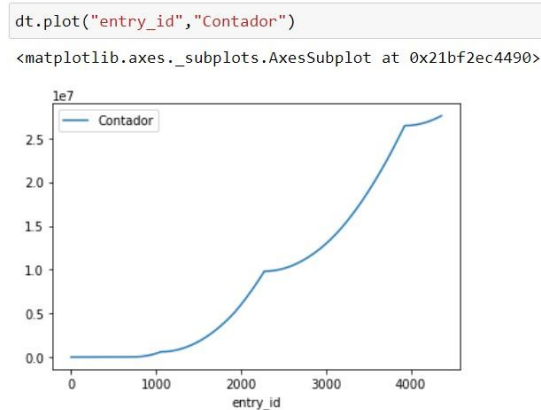


Figura 61 - Gráfico de contador automático para guía después de la limpieza de datos

Fuente: generada en jupyter por los investigadores

Datos estadísticos iniciales obtenidos:

Variables (Temperatura, Humedad, Flujo, Peso, Temp\_Ext, Hum\_Ext, RSSI, Contador)

Valores máximos en las variables con los datos limpios

(36.8, 24, 55132161, 16.6, 47.8, 92, -39, 92, 27595436)

Valores mínimos en las variables con los datos limpios

(30.6, 10, 78, 10.0, 15.7, 20, -43, 20, 38)

En la Figura 56 que corresponde al flujo relativo se puede observar un comportamiento lineal semi exponencial, esto se puede deducir que es por la sensibilidad de los sensores de ultrasonido y la organización de las abejas, que a pesar de haber realizado pruebas de laboratorio para la calibración de estos no son sensores adecuados para este tipo de medición

Haciendo un análisis entre las temperaturas y humedad almacenadas en la SD comparadas con las enviadas al servidor visibles en las figuras 62 y 63 respectivamente, se infiere que en el servidor llegaron menos datos, pues se denotan los datos sin valor usualmente por errores de medición o errores de comunicación entre los sensores y el dispositivo TTGO, por errores de puerto y conexión. Aparte de estos datos nulos o "0" tenemos datos faltantes, que no llegaron correctamente al servidor por lo cual son nulos o "0" también.

A pesar de tener un RSSI bastante bajo y poco variante en la toma de datos, podemos atribuir estos fallos a la comunicación LORA o por las sincronías de tiempo de recepción y envío de datos.

```
# Eje x Eje y
dt.plot("entry_id",["Temperatura_TS", "Temperatura"])
<matplotlib.axes._subplots.AxesSubplot at 0x1dd1c4b3130>
```

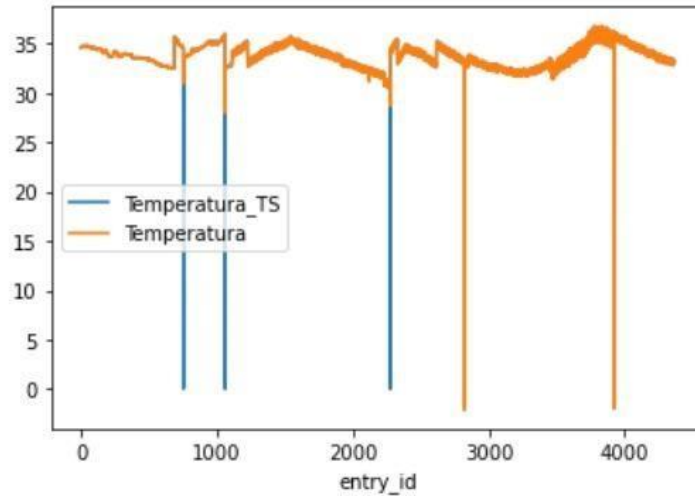
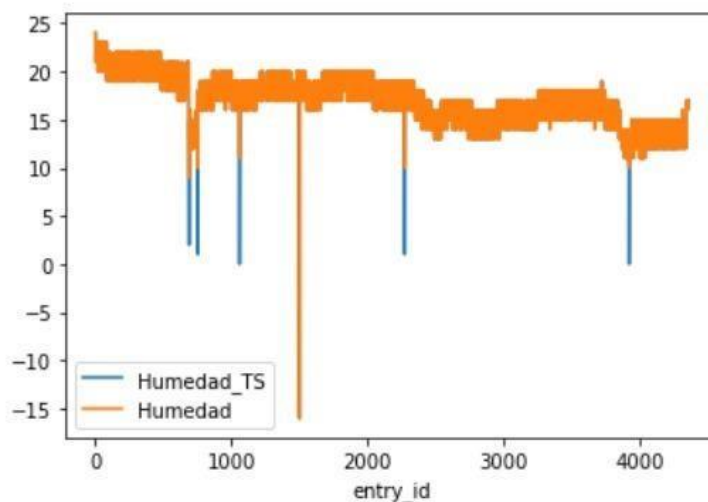


Figura 62 - Gráfico de temperatura SD y temperatura TS  
Fuente: generada en jupyter por los investigadores

```
# Eje x Eje y
dtt.plot("entry_id",["Humedad_TS", "Humedad"])
<matplotlib.axes._subplots.AxesSubplot at 0x1dd1c587220>
```



63 - Gráfico de humedad SD y humedad TS

Figura

Fuente: generada en jupyter por los investigadores

## 7. CONCLUSIONES

Se realizó el diseño y se implementó el prototipo de monitoreo obteniendo así resultados muy prometedores, con equipo de bajo costo y de bajo consumo eléctrico, pudiendo ser a su vez sustentable energéticamente, demostrando que puede funcionar por varios días sin la necesidad de revisiones de personal. Los datos adquiridos son datos muy reales comparados con otros prototipos similares en condiciones de igualdad climática y poblacional de las abejas.

La tecnología utilizada LORA es de gran ventaja para este tipo de proyectos relacionados con IoT, pues tenemos un ancho de banda libre que ofrece una seguridad de transmisión con el menor número de fallos en largos tiempos de funcionamiento, un RSSI bajo asegurando datos correctos, muy poco requerimiento energético, para envío y recepción de mensajes. Con todas estas ventajas logramos obtener un funcionamiento continuo con una eficiencia de aproximadamente el 99.84% de los datos evaluados útiles.

En otros países están implementando diferentes tecnologías para la producción apícola. Colombia tiene muy poco desarrollo en dicha área que en estos momentos está apostándole a su mejoramiento viendo los resultados de otros países con menos biodiversidad, con la implementación de estos prototipos los apicultores pueden monitorear variables de una manera más fácil en una gran cantidad de colmenas, ubicadas en diferentes regiones sin perturbar el trabajo de las abejas, ahorrando tiempo y manteniendo la producción de sus diferentes productos. En ocasiones los apicultores deben enfrentarse a cambios de clima repentinos que afectan las visitas para revisar las colmenas, también se observan grandes disminuciones de población de abejas por cultivos envenenados cercanos, estos son síntomas de alarma que podrían observar gracias al prototipo antes de tener una gran pérdida de producción.

Se encontró que el sector de la colmena con el prototipo tiene constantes fumigaciones a cultivos de lulo en una zona cercana, y estaba afectando dicha población de abejas, la cual pudo ser reubicada a tiempo salvando la colmena donde se encuentra el prototipo y las que están a su alrededor.

Los datos son revisados con facilidad por el apicultor y con la facilidad de ver datos obtenidos día a día en cualquier dispositivo con acceso a internet además los datos son almacenados para futuras investigaciones sobre comportamientos de población apícola.

En consumo de potencia es muy óptimo demostrando un funcionamiento continuo inclusive en los días con temperaturas bajas censadas.



Gracias al almacenamiento de datos se pudo hacer la correlación de estos comparándolos uno a uno por el contador guía demostrando la eficiencia del funcionamiento ya antes mencionada.

## **8. RECOMENDACIONES**

Instalar el prototipo en más colmenas para tener una red en mayada de comunicación para lograr la menos perdida de datos posible.

Optimizar la programación de códigos de los dispositivos de recepción y medición para tener una sincronía de envío de datos más exacta y disminuir la perdida de información.

El reemplazo del sensor de ultrasonido para lograr una medida real del flujo relativo de las abejas en la colmena.

Realizar una aplicación donde se puedan tener los datos depurados de malas mediciones, con interfaz gráfica más amigable con el apicultor, para poder comparar y visualizar dichos datos limpios.

Mejorar los materiales utilizados para tener un prototipo de fácil instalación en cualquier colmena que ya tenga abejas o que este en producción.

## 11. BIBLIOGRAFÍA

Paul Aguayo S., paguayo@olimex.cl. " INTRODUCCIÓN AL MICROCONTROLADOR", 2014. Paper.

TOMASI, Wayne. Sistemas de comunicaciones electrónicas. 4 Edición. Pearson Educacion, 2015.

DAZA, H., VEGA, O. Telemetría aplicada al monitoreo y control de la válvula La Maria PK165, 2014, Facultad de Ciencias e Ingeniería, Universidad de Manizales.

Wikipedia teledetección. [en línea]. Consultado: ](03 de Marzo de 2021)  
Disponible en: <https://es.wikipedia.org/wiki/Teledetecci%C3%B3n>

Mecafenix, I. [en línea]. Consultado:(11 de Marzo de 2021). Tipos de sensores y sus características. Obtenido de  
<https://www.ingmecafenix.com/automatizacion/sensores/>

Jordi Besora Magem "INFORME TÉCNICO PARA LA CONSTRUCCIÓN DE UNA COLMENA Y PORTANÚCLEO TIPO LANGSTROTH" (2017), Informe tecnico

Revista bolivariana, "Aportes para el análisis del sector Apícola Colombiano" Bogotá,2013

Marta García García, "La polinización en los sistemas de producción agrícola: revisión sistemática de la literatura",Articulo, IDESIA Chile, Volumen 34, N°3. Páginas 51-66, 2016

Iberoamericanos OEI, [en línea], consultado: (17 de Mayo de 2021) disponible en:  
<https://www.oei.es/historico/decada/accion.php?accion=25>

RIVERA, M. Á. [en línea]. Consultado: (11 de Marzo 2021).Disponible en:  
<https://repository.javeriana.edu.co/bitstream/handle/10554/12407/BarraganRiveraMiguelAngel2014.pdf;jsessionid=6B74012D5673FE8C15753BD7E4461EF5?sequence=1>

Definición apicultura [en línea] consultado: (4/Abril/2012)  
<lema.rae.es/drae/?val=apicultura>.

Laverde Rodríguez Jairo César, Agenda Prospectiva De Investigación Y

Desarrollo Tecnológico Para La Cadena Productiva De Las Abejas Y La Apicultura En Colombia Con Énfasis En Miel De Abejas, Ministerio de Agricultura y Desarrollo Rural, Colombia, 2010, pp. 15, 25

XI encuentro colombiano de apicultura y seminario apícola de los andes xiv jornada apícola universitaria, Bogotá, 2018

Journal of Apicultural Research "Influence of brood rearing temperature on honey bee development and susceptibility to poisoning by pesticides", 2010

Armands Kviesis, Aleksejs Zacepins, "Application of neural networks for honey bee colony state identification", Carpathian control conference (ICCC) , 2016 pp. 413417

Vyacheslav G. Rybin "Embedded DataAcquisition System for Beehive Monitoring", Saint-Petersburg, 2017

Aldis Pecka, Vitalijs Osadcuks "Development of Internet of Things concept for Precision Beekeeping", International Carpathian Control Conference (ICCC), 2017

Camargo, M. E.[en línea] En defensa de las abejas. Consultado: (24 de Mayo de 2021). Disponible en: [www.rcnradio.com](http://www.rcnradio.com):  
<https://www.rcnradio.com/opinion/endefensa-de-las-abejas>

ONU, N. [een línea] Consultado: (19 de Marzo de 2021). Disponible en: <https://news.un.org/>. Obtenido de <https://news.un.org/es/story/2016/02/1351301>

Minagricultura. [en línea]. Plan Nacional de Desarrollo 2014-2018 Todos por un nuevo país. Consultado: (02 de Marzo de 2021) Disponible en:  
<https://www.minagricultura.gov.co/https://www.minagricultura.gov.co/planeacioncontral- gestion/Gestin/Plan%20de%20Acci%C3%B3n/PLAN%20NACIONAL%20DE%20DESARROLLO%202014%20-%202018%20TODOS%20POR%20UN%20NUEVO%20PAIS.pdf>

G. Kritsky, The Quest for the Perfect Hive: A History of Innovation in Bee Culture, 2010.

R. Gupta, W. Reybroeck, J. van Veen, and A. Gupta, Eds., Beekeeping for poverty alleviation and livelihood security: Vol.1: Technological aspects of beekeeping. Springer Science+Business Media, 2014. Pp 64

H. Blackiston, Beekeeping For Dummies, 4th ed. John Wiley & Sons, Inc., 2017. Pp48

H. Blackiston, Beekeeping For Dummies, 4th ed. John Wiley & Sons, Inc., 2017.pp 48

Henríquez Jiménez Félix, Paricio Núñez Enrique, Manual Práctico de Apicultura para las Islas Canarias, consejería de Agricultura y Pesca, 1989. pp. 52

Mazzuferi, V. S. Avalos.. Metamorfosis de Insectos. SERIE DIDACTICA. CUADERNILLO 2. Cátedra Zoología Agrícola. Fac. de Cs. Agrop. U.N.C 1997

Ali, D. Jayakody, Y. Chursin, S. Affes, and S. Dmitry, "Recent advances and future directions on underwater wireless communications," Archives of Computational Methods in Engineering, Jan. 2019.

R. digital de la Facultad de Ingeniería UNAM, [En Linea.] "Microcontroladores,"consultado (2 de Febrero 2021) Disponible en: <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/760/4/A4.pdf>.

All Datasheet [en línea] consultado: (4 de Febrero de 2021) Disponible: <https://www.alldatasheet.com/datasheet-pdf/pdf/1243003/ESPRESSIF/ESP32.html>

Iotbytes [en línea] consultado (5 de Fevbrero de 2021) Disponible en <https://iotbytes.wordpress.com/esp8266-pinouts/>

Sparkfun [en línea] Consultado: (2 de Marzo de 2021) Disponible en: [https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711\\_english.pdf](https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf)

Española, R. A. (s.f.). [en línea] Humedad Relativa. Consultado (7 de Febrero de 2021) Disponible en: <https://dle.rae.es/humedad?m=form#G3gHYnG>

Española, R. A. (s.f.).[en línea] Masa. Consultado (5 de Marzo 2021) Disponible: [https://dle.rae.es/: https://dle.rae.es/masa?m=form](https://dle.rae.es/:https://dle.rae.es/masa?m=form)

Española, R. A. (s.f.). [en línea] Flujo. Consultado (5 de Marzo 2021) Disponible: [https://dle.rae.es/: https://dle.rae.es/flujo?m=form](https://dle.rae.es/:https://dle.rae.es/flujo?m=form)

Microcontroladores, [En Linea.] "Sensores con microcontroladores," Consultado ( 6 de Marzo de 2021) Disponible: <https://conmicrocontroladores.com/sensores/>.

CE-Ibérica-S.L., [En Línea.] “Equipos de medida - balanzas - regulación y control,” Consultado (3Marzo 2021) Disponible: <https://www.pce-iberica.es/instrumentos-demedida/sistemas/>.

EYENCE, [En Línea.] “Fundamentos del sensor: Guía de sensores para fábricas clasificados por principios,” Consultado: (8 Marzo 2021) Disponible: <https://www.keyence.com.mx/ss/products/sensor/sensorbasics/photoelectric/info/>.

## 12. ANEXOS

### 12.1 COSTOS

Valor estimado para la elaboración del prototipo de monitoreo

Descripción	Costo (USD)
Panel solar	30
Bateria	25
Regulador de voltaje	13
Celdas de carga	15
Sensor DHT11 (Temperatura y Humedad)	3
Sensor ultrasonido	3
Módulos TTGO	20
Módulo esp8266	2
Impresión PCB	3

Micro SD	5
Cableado	3
Caja de protección	10
<b>COSTO TOTAL</b>	<b>133</b>

## 12.2 Código programado en el ESP32 (Colmena)

```

#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_Sensor.h>
#include <DHT.h>
// #include <SD.h>

// LORA SENDER
// Libraries for LoRa
#include <SPI.h>
#include <LoRa.h>

// define the pins used by the LoRa transceiver module
#define SCK 5
#define MISO 19
#define MOSI 27
#define SS 18
#define RST 23
#define DIO0 26

#define BAND 915E6

#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels

// Declaration for an SSD1306 display connected to I2C (SDA, SCL pins)

```

```
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
```

```
#define DHTPIN 25 // Digital pin connected to the DHT sensor
const int sensorInte = 34; const int sensorInte1 = 35;
int c = 0; int
c1 = 0; int
ct = 0; int tt
= 0;
// Uncomment the type of sensor in use:
#define DHTTYPE DHT11 // DHT 11
```

```
DHT dht(DHTPIN, DHTTYPE);
```

```
void setup() {
  Serial.begin(115200);
  Serial.println("LoRa Sender Test");
  dht.begin();
  {
    Serial.println("Falta, sd o no esta bien conectada \n");
    //programa encerrado
  }
  return;
}
Serial.println("SD Iniciada correctamente.");
*/
pinMode(sensorInte , INPUT); //definir pin como entrada
pinMode(sensorInte1 , INPUT); //definir pin como entrada

if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
  Serial.println(F("SSD1306 allocation failed")); for(;;);
}
delay(200);
display.clearDisplay();
display.setTextColor(WHITE);

//SPI LoRa pins
SPI.begin(SCK, MISO, MOSI, SS);
//setup LoRa transceiver module
LoRa.setPins(SS, RST, DIO0);
LoRa.setSyncWord(0xF5); //Palabra clave para recepcion
if (!LoRa.begin(BAND)) {
  Serial.println("Starting LoRa failed!");
  while (1);
}
```

```

}

void loop() {
  delay(5000);

  int value = 0;
  value = digitalRead(sensorInte); //lectura digital de pin  if
  (value == LOW) {
    Serial.println("Optointerruptor activado");
    c++;    delay(1200);
  }

  int value1 = 0;
  value1 = digitalRead(sensorInte1); //lectura digital de pin
  if (value1 == LOW) {
    Serial.println("Optointerruptor activado");
    c1++;
    delay(1200);
  }
  ct=c1+c;

  //read temperature and humidity
  float t = dht.readTemperature();
  float h = dht.readHumidity();  if
  (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
  }
  Serial.println("T: ");
  Serial.print(t);
  Serial.println(" H: ");
  Serial.print(h);
  Serial.println(" F: ");
  Serial.println(ct);

  LoRa.beginPacket();
  LoRa.print(t);
  LoRa.println("/");
  LoRa.print(h);
  LoRa.println("/");
  LoRa.print(ct);
  LoRa.println("/");
  LoRa.print(tt);

```



```
LoRa.println("/");
```

```
LoRa.endPacket();
```

```
// clear display  
display.clearDisplay();  
display.setTextSize(1);  
display.setCursor(0,0);  
display.print("LORA-SENDER Api");
```

```
// display temperature  
display.setTextSize(1);  
display.setCursor(0,10);  
display.print("Temp: ");  
display.setTextSize(1);  
display.setCursor(50,10);  
display.print(t); display.print("  
"); display.setTextSize(1);  
display.cp437(true);  
display.write(167);  
display.setTextSize(1);  
display.print("C");
```

```
// display humidity  
display.setTextSize(1);  
display.setCursor(0, 20);  
display.print("Hum: ");  
display.setTextSize(1);  
display.setCursor(50, 20);  
display.print(h);  
display.print(" %");
```

```
//display flujo  
display.setCursor(0, 30);  
display.print("Flu: ");  
display.setTextSize(1);  
display.setCursor(50, 30);  
display.print(ct);  
display.display();
```

```
//display Autores  
display.setCursor(0, 40);  
display.print("Albert");  
display.setTextSize(1);  
display.setCursor(64, 40);
```

```

display.print("Leyder");
display.display();
  tt = tt+1;
}

```

Código programado en el ESP8266

```

#include <SPI.h>
#include <SD.h>
#include "HX711.h"

```

```

HX711 scale(5, 4); //DOUT-D1, CLK- D2

```

```

float calibration_factor = 22450; // this calibration factor is adjusted according to my
load cell float units;
float ounces;

```

```

File myFile;

```

```

/*#define DHTPIN 4
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
*/ int c =
0;

```

```

void setup() {
  // Open serial communications and wait for port to open:
  Serial.begin(115200);
  // dht.begin();
  scale.set_scale();
  scale.tare(); //Reset the scale to 0

```

```

  long zero_factor = scale.read_average(); //Get a baseline reading

```

```

  Serial.print("Initializing SD card...");

```

```

  if (!SD.begin(15)) {
    Serial.println("initialization          failed!");
    return;
  }
  Serial.println("initialization done.");

```

```

}

```

```

void loop() {

    // open the file. note that only one file can be open at a time, //
    so you have to close this one before opening another.
    myFile = SD.open("test.csv", FILE_WRITE);

    // if the file opened okay, write to it:  if
    (myFile) {
        String dataString = "";
        Serial.print("\n");
        Serial.print(units);
        Serial.print(" kg");
        //float t = dht.readTemperature();
        Serial.print("Peso: ");
        Serial.println();    dataString
        += String(units);    dataString
        += ",";    dataString +=
        String(c);
        Serial.println(dataString);
        myFile.println(dataString);
        // close the file:
        myFile.close();
        Serial.println("done.");
    } else {
        // if the file didn't open, print an error:
        Serial.println("error opening test.txt");
    }
    c      =      c+1;
    delay(5000);
}

```

### 12.3 Código programado en el ESP32 (Casa)

```

//Library separador
#include<Separador.h>
Separador s;
//Libraries for LoRa
#include <SPI.h>
#include <LoRa.h>

//Libraries for OLED Display
#include <Wire.h>
#include <Adafruit_GFX.h>

```

```

#include <Adafruit_SSD1306.h>
#include <DHT.h>

//define the pins used by the LoRa transceiver module
#define SCK 5
#define MISO 19
#define MOSI 27
#define SS 18
#define RST 23
#define DIO0 26

//433E6 for Asia
//866E6 for Europe
//915E6 for North America
#define BAND 915E6

#define DHTPIN 25
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

#include "ThingSpeak.h"
#include <WiFi.h>

char ssid[] = "Api_LoRa"; // your network SSID (name) char
pass[] = "152215960393"; // your network password
//int keyIndex = 0; // your network key Index number (needed only for WEP)
WiFiClient client;

unsigned long myChannelNumber = 1214923; const char
* myWriteAPIKey = "SW92JUZH3GZE9B2G";
String myStatus = "";

#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);

String LoRaData;

void setup() {

  //initialize Serial Monitor
  Serial.begin(115200);
  Serial.println("LoRa Receiver Test");

```

```

    if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3c)) { // Address 0x3C for
128x32
        Serial.println(F("SSD1306 allocation failed"));
        for(;;); // Don't proceed, loop forever
    }
    display.clearDisplay();
    display.setTextColor(WHITE);
    display.setTextSize(1);
    display.setCursor(0,0);
    display.print("LORA RECEIVER ");
    display.display();

    //SPI LoRa pins
    SPI.begin(SCK, MISO, MOSI, SS);
    //setup LoRa transceiver module
    LoRa.setPins(SS, RST, DIO0);
    LoRa.setSyncWord(0xF5); //Palabra clave para recepcion if
(!LoRa.begin(BAND)) {
        Serial.println("Starting LoRa failed!");
        while (1);
    }
    Serial.println("LoRa Initializing OK!");
    display.setCursor(0,10);
    display.println("LoRa Initializing OK!");
    display.display();
    delay(100);

    WiFi.mode(WIFI_STA);
    ThingSpeak.begin(client); // Initialize ThingSpeak

    if(WiFi.status() != WL_CONNECTED){
        Serial.print("Attempting to connect to SSID: ");
        while(WiFi.status() != WL_CONNECTED){
            WiFi.begin(ssid, pass); // Connect to WPA/WPA2 network. Change this line if
            using open or WEP network
            Serial.print(".");
            delay(5000);
        }
    }
    else {
        ESP.restart();
    }
}

```

```

    dht.begin();
}

void loop() { //try to
  parse packet
  int packetSize = LoRa.parsePacket();

  while (packetSize){
    //received a packet
    Serial.print("Received packet ");
    Serial.println("\nConnected.");
    //read packet
    while (LoRa.available()) { LoRaData = LoRa.readString();
    Serial.print(LoRaData); float tc = dht.readTemperature();
    //Sensor exterior TTGO Casa float hc = dht.readHumidity();
    //Sensor exterior TTGO Casa
    Serial.print("Temperature: ");
    Serial.print(tc);
    Serial.print(" *C ");
    String t = s.separa(LoRaData, '/', 0);
    String h = s.separa(LoRaData, '/', 1);
    String f = s.separa(LoRaData, '/', 2);
    String con = s.separa(LoRaData, '/', 3);
    float ti = t.toFloat();
    float hi = h.toFloat(); int
    fi = f.toInt();
    int tt = con.toInt(); //tener encuesta que es el contador
    //print RSSI of packet int
    rssi = LoRa.packetRssi();
    Serial.print(" with RSSI ");
    Serial.println(rssi);

    // set the fields with the values
    ThingSpeak.setField(1, ti);
    ThingSpeak.setField(2, hi);
    ThingSpeak.setField(3, fi);
    ThingSpeak.setField(4, tc);
    ThingSpeak.setField(5, hc);
    ThingSpeak.setField(6, tt);
    ThingSpeak.setField(7, rssi); myStatus
    = String("Datos enviados");
    ThingSpeak.setStatus(myStatus);
    ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
  }
}

```

```

    // Display information
display.clearDisplay();
display.setCursor(0,0);
display.print("RECEIVER T/H/F");
// display temperature
display.setTextSize(1);
display.setCursor(0,10);
display.print("T ");
display.setTextSize(1);
display.setCursor(10,10);
display.print(ti);
display.setTextSize(1);
display.cp437(true);
    display.write(167); // Para que los grados °C
    display.print("C");
display.setCursor(60, 10);
display.print(tc);
display.setTextSize(1);
display.cp437(true);
display.write(167);
    display.print("C");

```

```

    // display humidity
display.setTextSize(1);
display.setCursor(0, 20);
display.print("H");
display.setCursor(10,20);
display.print(hi);
display.print("%");
display.setCursor(60,20);
display.print(hc);
display.print("%");

```

```

    //display flujo
display.setTextSize(1);
display.setCursor(0, 30);
display.print("Flu: ");
display.setCursor(30,30);
    display.print(fi);

```

```

    //display t exterior
display.setTextSize(1);
display.setCursor(0, 40);
    display.print(tt);

```

```
    //display RSSI
    display.setCursor(0,50);
    display.print("RSSI:");
    display.setCursor(30,50);
    display.print(rssi);

    display.display();
    delay(60000);
  }
}
```