

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA SEDE
CONCEPCIÓN “REY BALDUINO DE BÉLGICA”



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

Respaldo de energía y monitoreo de variables a través de IOT para
planta Quiriquina.

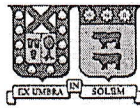
*Trabajo de Titulación para optar al Título Profesional de Ingeniero en Control e
Instrumentación industrial.*

Autor:

Franco Fabiano Ferranti Pereira

Profesor guía:

Israel Figueroa Pedreros



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): Memoria o trabajo de título Tesis de Postgrado

Título del trabajo: Respaldo de energía y monitoreo de variables a través de IOT Para Planta Guiriquina

Nombre del candidato(a): Franco Fabiano Ferranti Pereira

Carrera / Grado: Ingeniería en Control e Instrumentación Industrial

Campus: Concepción Departamento: Electrónica e Informática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, ISRAEL FIGUEROA PEDREZOS, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución.

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL (marcar una opción)

El trabajo **NO contiene** información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (**embargo**) por (**marcar una opción**):

6 meses 12 meses 2 años 3 años 5 años 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 06/01/2026 Firma: 

Estudiante o Candidato(a):

Fecha: 06/01/2026 Firma: 

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

Agradecimientos

Quiero expresar mi mas sincero agradecimiento a todas las personas que de alguna u otra forma contribuyeron en mi desarrollo profesional a lo largo de mi carrera.

En primer lugar, agradezco profundamente a mi pareja e hijo, Catalina Gajardo y Santino Ferranti. por su apoyo incondicional, paciencia y comprensión durante todo este proceso especialmente en los momentos de mayor exigencia académica, convirtiéndose en una fuente de motivación e inspiración, su siempre presente apoyo, amor y cariño fueron fundamentales para no rendirme y seguir adelante ante las adversidades.

A mi familia por su apoyo incondicional brindado durante cada etapa de mi formación académica.

A mi profesor guía Don Israel Figueroa Pedreros por su constante disposición, orientación y valioso apoyo en la creación de esta memoria.

A la Universidad Técnica Federico Santa María, institución que me brindo la oportunidad de formarme profesionalmente, entregándome las herramientas académicas y recursos necesarios para el desarrollo de mis estudios.

Índice

1	INTRODUCCION DEL PROBLEMA.....	8
1.1	IMPORTANCIA DEL PROBLEMA	9
1.2	MEDIOS ACTUALES DE LA PLANTA ANTE EL PROBLEMA	9
1.2.1	FALENCIAS DE LOS MEDIOS ACTUALES.....	10
2	Tecnologías asociadas	11
2.1	Grupos electrógenos y sistemas auxiliares.....	11
2.1.1	¿Qué es un grupo electrógeno?	11
2.1.2	Sistema Auxiliares	11
2.1.3	Puesta en marcha automática.....	13
2.2	Descripción de la IOT	14
2.2.1	Como se lleva a cabo la IOT	14
2.2.2	Importancia de la IOT	15
2.3	Tecnologías en sistemas de control IOT	15
2.3.1	Sensores y actuadores	15
2.3.2	Controladores y Microcontroladores	17
2.3.3	Protocolos de Comunicación	19
2.3.4	Tecnologías de Conectividad en IOT	20
2.3.5	Plataformas en la nube	21
2.3.6	Herramientas de programación	23
3	Solución.....	24
3.1	Objetivos	24
3.2	Posibles soluciones que cumplan con el objetivo	24

3.3	Plan de acción para lograr los objetivos planteados.	25
3.4	Diagrama de flujo Solución	27
3.5	Diseño del circuito	27
3.6	Programación del Circuito	32
3.7	Plataforma en la nube THINGSPEAK	33
4	Planificación de trabajo	34
4.1	Planificación del Prototipo	34
4.2	Tiempos de Desarrollo del Prototipo	34
4.2.1	Diseño y definición de arquitectura.....	34
4.2.2	Diseño ensamblaje físico y cableado inicial	35
4.2.3	Desarrollo del Programa principal.....	35
4.2.4	Incorporación del software con el hardware	35
4.2.5	Puesta en marcha y finalización del Prototipo.....	35
4.3	Costos de la elaboración del prototipo	37
5	Resultados y Conclusión	39
5.1	Resultado obtenido	39
5.2	Conclusión	39

Siglas y Simbología

Siglas:

- IOT:** Internet Of Things / Internet de las Cosas
- RPM:** Revoluciones Por Minuto
- ATS:** Automatic Transfer Switch / Interruptor de Transferencia Automática
- UPS:** Uninterruptible Power Supply / Sistema de Alimentación Ininterrumpida
- SPI:** Serial Peripheral Interface / Interfaz periférica Serial
- CS:** Chip Select / Selección de Chip
- SCK:** Serial Clock / Reloj Serial
- SO:** Serial Output / Salida Serial
- DO:** Data Output / Salida de Datos
- PLC:** Programmable Logic Controller / Controlador lógico Programable
- SRAM:** Static Random Acces Memory / Memoria de Acceso Aleatorio Estática
- ROM:** Read Only Memory / Memoria de Solo Lectura
- ADC:** Analog to Digital Converter / Convertidor Análogo Digital
- SAR:** Successive Approximation Register / Registro de Aproximación Sucesiva
- DAC:** Digital to Analog Converter / Convertidor Digital a Analógico
- MQTT:** Message Queuing Telemetry Transport / Message Queuing Telemetry Transport
- HTTP:** HyperText Transfer Protocol / Protocolo de Transferencia de Hipertexto
- CoAP:** Constrained Application Protocol / Protocolo de Aplicación para Dispositivos Restringidos
- LAN:** Local Área Network / Red de Área Local

TV: Televisión / Television

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

UDP: User Datagram Protocol / Protocolo de Datagrama de Usuario

DTLS: Datagram Transport Layer Security / Seguridad de la Capa de Transporte para Datagramas

CSS: Chirp Spread Spectrum / Espectro Ensanchado por Chirp

Resumen

Hoy en día el avance de las tecnologías ha permitido desarrollar sistemas inteligentes capaces de solucionar diversos problemas de la vida cotidiana tanto como en el mundo industria, una de estas tecnologías es Internet de las cosas (IOT) que nos permite controlar, monitorear y automatizar procesos en tiempo real a través de redes de comunicación.

Dentro de una gran variedad de dispositivos capaces de implementar el Internet de las cosas se destaca uno de los microcontroladores de última generación como el **ESP32** el cual integra conectividad inalámbrica como Wifi, una alta capacidad de procesamiento y un bajo consumo energético, convirtiéndose en una herramienta clave para implementar soluciones de monitoreo y control a distancia.

Bajo este contexto en este trabajo de título se diseñara un prototipo de sistema de control para el respaldo del suministro eléctrico en la planta repetidora de telecomunicaciones isla Quiriquina usando el ESP32 y un conjunto de sensores con el fin de poder controlar la partida automática de un generador auxiliar en el caso de un corte de suministro y a la vez poder monitorear variables del generador como nivel de combustible, temperatura y Potencia suministrada a la planta a través de IOT siendo estas variables monitoreadas de forma remota en la radio estación marítima de Talcahuano mediante dashboards en tiempo real.

ALCANCE DEL PROYECTO

- I. El alcance de este proyecto se enfocó en el análisis, diseño y propuesta de solución técnica que asegure la continuidad del suministro eléctrico en la planta receptora.
- II. Evaluación de alternativas de respaldo energético con sistema transferencia automática.
- III. Evaluación y diseño de control mediante sensores con tecnología IOT para monitoreo en tiempo real y accesibilidad en cualquier lugar.
- IV. Evaluación económica – financiera del proyecto.
- V. Este proyecto se limitará al diseño y validación técnica mediante simulaciones en laboratorio, quedando fuera del alcance la implementación física a escala real.

RESTRICCIONES DEL PROYECTO

- I. Disponibilidad de recursos y alcance físico: Diseño y validación se realizarán con equipos y e instrumentos disponibles solo en laboratorios, lo que deja fuera la implementación del sistema en la planta.
- II. Tiempo de ejecución: el proyecto deberá completarse dentro del período académico establecido, lo que restringe la posibilidad de realizar pruebas prolongadas o estudios comparativos de largo plazo.
- III. Acceso a información: la disponibilidad de datos técnicos de la infraestructura de la planta, obligando a trabajar con información de referencia y con supuesto.

- IV. Seguridad de la información: no se podrá trabajar con la red dentro de la planta debido a la seguridad de la red Wan armada.

1 INTRODUCCION DEL PROBLEMA

La planta Receptora Quiriquina es una instalación perteneciente a la Radio estación marítima de Talcahuano ubicada en la Isla Quiriquina, bahía de Concepción la cual cumple con el propósito de funcionar como una planta repetidora de telecomunicaciones para dicha radio estación.

La planta Receptora Quiriquina enfrenta una situación crítica en lo referente a la continuidad del suministro eléctrico, debido a diversos factores propios de una infraestructura insular y fallas en la red de distribución tales como:

- **Cortes de Poder por mantenimiento:** Debido a que la planta receptor está ubicada en territorio insular, la red de suministro eléctrico que alimenta dicha planta es dependiente de la Escuela de grumetes Alejandro Navarrete Cisterna la cual es abastecida mediante una línea submarina proveniente de la península de Tumbes, lo que provoca dependencia del correcto funcionamiento de la red en esa área local lo que nos deja vulnerable a cualquier corte de suministro por mantenimiento o falla en dicha área.
- **Cortes de poder por fallas en la Escuela de Grumetes:** Debido a que la línea de alimentación que llega desde la península de Tumbes hasta la subestación eléctrica ubicada en la escuela de grumetes deja vulnerable a la planta ante cualquier falla o corte de poder provocado por la escuela.
- **Cortes de poder por Flora, Fauna y Condiciones medioambientales:** Debido a que la planta receptora se encuentra ubicada en la parte mas alta de la Isla Quiriquina, la distancia que tiene que recorrer la línea de alimentación de la escuela hasta la planta se ve vulnerable ante la Flora silvestre debido a que en el territorio insular hay un exceso de arboles que sumados a unas condiciones medio ambientales como vientos fuertes provocan la caída de dichos arboles sobre las líneas de alimentación. También la fauna

silvestre propios de la isla como aves de gran tamaño al posarse arriba del tendido eléctrico o transformadores de la planta provocan cortes y fallas prolongadas.

Dichas fallas y mantenimientos del suministro eléctrico mencionadas anteriormente dejan a la planta receptora con un alto margen de incertidumbre debido a que dichas fallas no siempre logran ser subsanadas a la brevedad debido a diversos factores.

1.1 IMPORTANCIA DEL PROBLEMA

Debido a que la Radio Estación Marítima de Talcahuano presta servicios de Telecomunicaciones en el ámbito marítimo 24 horas 7 días a la semana esta no puede permitirse estar sin enlace durante todo el tiempo que duran los cortes del suministro eléctrico y/o esperar que el personal perteneciente a la escuela de grumetes tenga la disponibilidad de poner en marcha el grupo electrógeno de emergencia.

1.2 MEDIOS ACTUALES DE LA PLANTA ANTE EL PROBLEMA

La planta Receptora Quiriquina cuenta con una infraestructura antigua y bastante reducida, sin embargo, a 4 metros de esta planta hay una caseta que alberga un grupo electrógeno auxiliar para poder suministrar de energía eléctrica la planta ante posibles cortes de poder, este grupo electrógeno este compuesto por un motor Lister Petter modelo TS-2 adjunto aun generador monofásico Stamford, modelo UCL-164 de 220 Volt, 32.7 Amper 50HZ 9KVA, 1500 RPM.



Figura 1.1: Grupo electrógeno ubicado en Planta Receptora Quiriquina.

1.2.1 FALENCIAS DE LOS MEDIOS ACTUALES

- A pesar de que la planta cuenta con un grupo electrógeno destinado a garantizar el suministro eléctrico en caso de fallas, carece de un sistema de puesta en marcha automática, esta condición obliga a que el encendido y transferencia de energía deban realizarse manualmente, requiriendo la presencia física de un operador perteneciente a la radio estación el cual debido a que tiene que desplazarse de la radio estación en Talcahuano hasta la Isla mediante remolcadores con horarios limitados, retrasa la respuesta ante emergencias y afecta la continuidad del servicio.
- El grupo electrógeno que abastece a la planta no dispone de un sistema de monitoreo de variables críticas, tales como nivel de combustible, temperatura o potencia entregada. Esta falta de supervisión en tiempo real genera incertidumbre en los operadores de la radio estación marítima respecto al correcto funcionamiento del equipo y las horas

efectivas de operación, dificultando la planificación de mantenimientos y recargas de combustible oportunas debido a que este tiene una autonomía máxima de 35 horas.

2 Tecnologías asociadas

2.1 Grupos electrógenos y sistemas auxiliares

2.1.1 ¿Qué es un grupo electrógeno?

Un grupo electrógeno es un sistema diseñado para producir y suministrar de energía eléctrica a través de un sistema electromecánico de forma autónoma, esto es posible mediante la conversión de energía mecánica generada por un motor de combustión interna (diésel o gasolina) a energía eléctrica mediante un generador acoplado al eje del motor de combustión interna, esto transforma la energía mecánica en electricidad utilizable.



Figura 2.1: Grupo electrógeno Lister Petter.

2.1.2 Sistema Auxiliares

Los sistemas auxiliares ante caída de suministro eléctrico son un conjunto de equipos pensados para garantizar una continuidad del suministro eléctrico en instalaciones donde la interrupción de este puede generar consecuencias operacionales graves y económicas, estas tienen como misión

principal asegurar el funcionamiento sin interrupción de los sistemas eléctricos esenciales, manteniendo así una estabilidad y disponibilidad de energía ante alguna emergencia o falla.

Estos sistemas están diseñados para entrar en operación de forma automática o manual ante algún requerimiento, hay una gran variedad de estos sistemas de los cuales se destacan:

- **Grupos electrógenos:** Sistema que transforma la energía mecánica en energía eléctrica mediante el movimiento del eje de un generador a través de un motore de combustión interna, este tiene como ventaja la capacidad de entregar un alto nivel de potencia requerida por la instalación y mediante a otros dispositivos puede ser conmutado o activado de forma automática.
- **Fuente de alimentación ininterrumpida:** Es un dispositivo eléctrico/ electrónico capaz de almacenar energía en baterías recargables que se mantienen constantemente cargadas hasta que se detecte la ausencia del suministro eléctrico, este tiene como ventaja que conmuta de forma instantánea protegiendo a los equipos conectados a este de cambios bruscos.
- **Sistemas híbridos:** Este sistema combina 2 o más fuentes de suministro eléctrico como lo pueden ser las energías renovables eólica o solar, con la finalidad de cubrir gran parte de la demanda de consumo eléctrico, la ventaja de este sistema es que si el suministro eléctrico principal falla queda el segundo sistema activo alimentándola carga, además de que los sistemas de energía renovables cuentan con un banco de batería generalmente que van cargando mientras no se utiliza toda la capacidad.

2.1.3 Puesta en marcha automática

Los controladores automáticos y sistema de transferencia automática (ATS) están diseñados para un funcionamiento autónomo de los grupos electrógenos, ya que permiten detectar ausencia o fallas en la red de suministro eléctrico principal e iniciar la puesta en marcha del generador al mismo tiempo que transfieren la carga de forma automática, esto garantiza la continuidad del suministro eléctrico sin necesidad de que el usuario los tenga que accionar. Estos sistemas de puesta en marcha automática están compuestos de dos partes la cuales son:

- **Controladores automáticos:** Es un dispositivo electrónico que puede gestionar y garantizar las funciones del generador, este tiene como finalidad automatizar la secuencia de arranque y parada del generador como también que funcione dentro de los parámetros establecidos.



Figura 2.2: Controlador Deep Sea Electronics DSE7420.

- **Sistema de transferencia automática:** Son tableros con dispositivos electromecánicos o electrónicos en algunos casos encargados de transferir la carga eléctrica en caso de falla de una red principal A a una red de respaldo B como podría ser un generador, banco de baterías(UPS) o sistema de suministro eléctrico híbrido y viceversa en el caso de que la red principal volviera a funcionar, su funcionamiento se basa en la supervisión constante de del suministro eléctrico principal y al detectar una

anomalía o caída fuera de los límites establecidos transferir la carga a al sistema de respaldo.



Figura 2.3: Sistema de transferencia automática con interruptores Compact NSX y Compact NS.

2.2 Descripción de la IOT

El internet de las cosas (IOT) es una red tecnológica que nos permite la interconexión de objetos físicos mediante el internet, permitiéndonos mediante controladores, sensores y softwares obtener, procesar y enviar datos de un entorno físico, sistema o proceso que se esté realizando con la finalidad de poder tomar decisiones relevantes.

2.2.1 Como se lleva a cabo la IOT

Esta tecnología se puede realizar mediante el trabajo en conjunto de sensores y actuadores que puedan obtener variables y datos tanto analógicos como digitales pudiendo ser así analizados y procesadas por controladores o microcontroladores que mediante distintas tecnologías de conectividad como Wifi, Celular, Bluetooth, Lora, puedan subir estos datos a la nube donde serán almacenados y procesados.

2.2.2 *Importancia de la IOT*

La gran importancia de la tecnología IOT hoy en día para optimizar y automatizar procesos es que nos permiten mejorar la eficiencia debido a que podemos supervisar los datos y procesos que están ocurriendo en tiempo real mediante a un monitoreo continuo y a distancia del lugar físicos de donde se llevan a cabo estos procesos.

2.3 **Tecnologías en sistemas de control IOT**

2.3.1 *Sensores y actuadores*

Los sensores y actuadores son dispositivos que nos permiten detectar y medir datos de variables físicas y ambientales (temperatura, humedad, voltaje, presión, nivel de líquidos etc.) los sensores actúan como la capa de percepción haciendo la interacción entre el mundo físico y el sistema digital, los sensores pueden entregar datos tanto de forma análoga como digital (estados altos y bajos). Por otra parte, los actuadores son dispositivos encargados de ejecutar una acción física en respuesta a alguna señal de control proveniente de algún controlador, capaces de transformar una señal eléctrica en movimiento mecánico que pueda interactuar con su entorno. Un ejemplo de estos serían los:

- **Sensores de Temperatura:** Dispositivo capaz de detectar los cambios de temperatura en el ambiente o equipos y las transforma en una señal eléctrica capaz de ser interpretada por un sistema de control. Es bastante utilizado en la industria en una alta variedad de tipos con el fin de detectar temperatura en los procesos y poder tomar una respectiva acción con respecto al sistema en el que se utilice.



Figura 2.4: Modulo sensor de temperatura Max6675.

Modulo Max 6675 fabricado por Maxim Integrated / Analog Device está compuesto por un amplificador de entrada para la termocupla con el fin de acondicionar la señal de la

termocupla, Circuito de compensación de unión fría que asegura que el valor final que entrega el chip sea la temperatura real del punto caliente de la termocupla, un conversor análogo-digital de 14 bits que convierte la señal ya compensada a un valor digital quedando almacenado en registro internos como bit de temperatura, bit de estado y bit de detección de termocupla abierta y una interfaz serial tipo SPI para ser leída por el microcontrolador, CS(Chip Select), SCK(Serial Clock), SO/DO(Serial Data Output).

- **Sensor de Nivel:** Componente capaz de medir la altura o cantidad de liquido dentro de un contenedor, permitiendo mediante cálculos y su variación en respuesta eléctrica poder saber el nivel en que se encuentra de lleno un estanque (combustible, agua u otros fluidos).



Figura 2.5: Sensor de Nivel de combustible.

- **Sensor de efecto Hall:** Dispositivo capaz de medir la intensidad de corriente eléctrica que se Encuentra presente en un conductor sin necesidad de tener contacto directo, utiliza el efecto hall, este principio que genera una tensión de acuerdo con el campo magnético creado por la corriente, estos sensores son ampliamente utilizados porque permiten medir la corriente alterna tanco como la corriente continua de forma segura y fiable.



Figura 2.6: Modulo sensor de efecto Hall.

- **Contactores y Relés:** Los contactores y relés son considerados interruptores controlados de forma eléctrica mediante la carga de una bobina permitiendo mediante un campo magnético establecer contacto entre dos o más terminales activando y desactivando circuitos de baja, mediana o alta potencia, se utilizan generalmente en circuitos de aislación para separar la etapa de control con la etapa de Fuerza.



Figura 2.7: Contactor trifásico. **Figura 2.8:** relé monofásico 8 pines.

2.3.2 Controladores y Microcontroladores

Los controladores y microcontroladores son dispositivos compuestos de circuitos diseñados para coordinar funcionamiento de uno o mas elementos de un sistema, poder interpretar datos que recibe de sus sensores o periféricos y almacenar dichos datos en su memoria interna donde el procesador pueda acceder con el fin de ejecutar instrucciones previamente programadas. Dentro de los controladores y microcontroladores se destacan los siguientes:

- **Controladores Lógicos Programables (PLC):** Son sistemas electrónicos digitales diseñados para poder controlar y automatizar procesos industriales. A diferencia de los microcontroladores los PLC poseen una arquitectura más robusta, capaz de realizar operaciones en condiciones más adversas como ruidos eléctricos, vibraciones, humedad o altas temperaturas todo esto sin perder su fiabilidad y eficacia.



Figura 2.9: PLC Siemens S7- 1200.

- Microcontroladores:** Los microcontroladores son circuitos integrados que combinan una unidad de procesamiento, memoria y en algunos casos módulos de comunicación inalámbrica esenciales para proyectos IOT (wifi, Bluetooth) en un solo chip con el fin de programar su funcionamiento. Estos están orientados principalmente a un sistema de trabajo de bajo costo y consumo eléctrico, no siendo muy recomendado para ambientes industriales al menos que sean robustecidos para este tipo de ambientes.



Figura 2.10: Microcontrolador ESP32.

El microcontrolador ESP32 fabricado por Espressif Systems posee un núcleo Xtensa LX6 de 32 bits puede tener uno o dos núcleos hasta de 240 MHz, una memoria interna de 520KB de SRAM y 448 de ROM, un oscilador interno de 8MHz, cuenta con unos periféricos de 34 GPIO

programables 2 ADC SAR de 12 bits y hasta 18 canales, DAC 2 salidas de 8 bits, también posee conectividad inalámbrica como Wifi 802.11 (2.4GH) Bluetooth 4.2 + Bluetooth Low Energy, su voltaje de alimentación es de 2.3 a 3.6V ideal usar 3.3V.

Debido a todo esto y su posibilidad de conectividad por wifi lo hace perfecto para la implementación de proyecto en IOT.

2.3.3 Protocolos de Comunicación

Los protocolos de comunicación son un conjunto de normas pautadas que permite tanto a los dispositivos como usuarios comunicarse a través de paquetes de datos de una dirección a otra. El intercambio de información en IOT es entre los dispositivos y objetos hacia la nube sin usar cableado sino el espacio libre, dentro de los cuales se destacan los siguientes:

- **MQTT: Transporte de telemetría de cola de mensajes**, Es un protocolo de comunicación muy ligero diseñado para funcionar en redes con poco ancho de banda o conectividad inestable con dispositivos de poca potencia, su modelo de comunicación es publicador/suscriptor donde el servidor general el cual denominamos bróker actúa como intermediario entre los dispositivos, convirtiéndolo así en un protocolo eficiente y optimizado para redes inestables, este también se basa en el modelo TCP/IP lo cual lo hace accesible para correr sobre wifi, redes celulares y LoraWan permitiendo conectar cualquier dispositivo a aplicaciones en la nube sin problemas de compatibilidad.
- **HTTP: Protocolo de transferencia de hipertexto** es un protocolo de comunicación bastante utilizado en internet por su compatibilidad universal con servicios web y aplicaciones en la nube, su modelo es cliente/servidor donde el cliente realiza una solicitud y el servidor responde con la información solicitada, este protocolo esta diseñado para sistemas con buena conectividad sin embargo su uso en IOT se a extendido por su simplicidad, soporte generalizado y una muy accesible integración con infraestructura ya existentes, sin embargo su estructura con cabeceras extensas y su modelo basado en solicitudes del cliente incrementa su tráfico en red.

- **CoAP: Protocolo de aplicación restringida** es un protocolo diseñado específicamente para dispositivos con recursos muy limitados, es similar a el protocolo HTTP debido a que transfiere documentos completos pero con una estructura mucho mas ligera y eficiente optimizada para redes de baja potencia y alta latencia, esta protocolo a diferencia de HTTP opera sobre el protocolo UDP lo que reduce el consumo significativamente en el ancho de banda, en términos de seguridad utiliza DTLS (Datagram transport layer security) para ofrecer confidencialidad, autenticación e integridad en la transmisión de datos, su diseño orienta a la simplicidad y bajo consumo lo hace una opción bastante adecuada para aplicaciones IOT.

2.3.4 Tecnologías de Conectividad en IOT

Las tecnologías de conectividad en el mundo de la IOT son más bien los medios por los cuales se nos permite transmitir datos entre dispositivos y plataformas en la nube, haciendo posible la comunicación entre sí, existen diversas tecnologías, de las cuales más se destacan las inalámbricas como lo son:

- **Wifi:** El wifi es una de las tecnologías de telecomunicaciones inalámbrica que permite intercambiar datos entre dispositivos o conectarse a puntos de acceso en internet, el funcionamiento de este es mediante ondas electromagnéticas con una frecuencia que va entre los 2.4 GHz y los 5 GHz para transmitir datos a redes locales (LAN), este fácilmente puede comunicarse con dispositivos como computadoras, TV, celulares, microcontroladores con modulo wifi.
- **Redes 5G:** El funcionamiento de las redes 5G consiste en la transmisión de datos a través de ondas electromagnéticas que operan en frecuencias muy altas llegando incluso a bandas milimétricas, las antenas 5G dividen la señal en muchas pequeñas subfrecuencias para enviar más información a la vez, usan varias antenas en paralelo enfocando la señal directamente hacia cada usuario en vez de emitirla en todas direcciones.

- **Bluetooth:** Es una tecnología de conectividad de comunicación inalámbrica para corto alcance (100 metros aproximadamente) que transmite datos mediante ondas electromagnéticas en la banda de 2.4Ghz, su funcionamiento se basa en la conexión entre distintos dispositivos mediante una técnica llamada salto en frecuencia que cambia rápidamente el canal de transmisión muchas veces por segundo evitando así interferencias, estos datos se envían en paquetes utilizando el modelo Maestro- Esclavo.
- **LoRa:** La tecnología de conectividad LoRa funciona mediante una modulación llamada Chirp Spread Speactrum (Ccss) o modulación de espectro ensanchado esta modulación varía su frecuencia de forma continua para resistir interferencias propias de la comunicación inalámbrica manteniéndose estable a kilómetros de distancia, su banda esta entre los 433MHz a los 915MHz enviando pequeños datos a baja velocidad teniendo como prioridad el alcance y bajo consumo energético

2.3.5 *Plataformas en la nube*

- La nube es el espacio virtual o servidor en donde se almacenan y analizan los datos, se accede a través de internet estos están ubicados en centro de datos permitiendo al usuario acceder a archivos y aplicaciones mediante cualquier dispositivo IOT permitiendo visualizar información en tiempo real.

ThingsPeak : Es una plataforma de IOT de código abierto que permite a los usuarios recopilar analizar y visualizar datos de sensores y otros dispositivos conectados a internet en tiempo real ofreciendo muchos planes de suscripción. Para poder utilizarla se deben seguir los siguientes pasos:

- **Crear una cuenta en ThinSpeak:** Se debe crear una cuenta ingresando a <https://thingspeak.mathworks.com/> y ingresar al banner de “Comienza gratis” y crear una cuenta mediante un correo electrónico, se puede realizar con el mismo correo institucional debido a que ramos anteriores ya se posee una cuenta MathWorks.



Figura 2.11: Creación de cuenta Thingspeak.

- **Crear un nuevo canal:** Después de crear una cuenta, lo primero que se visualiza es un banner que indica la creación de un “Nuevo Canal” al presionar nos dirige a la creación de nuestro canal y asignación de nombre a las variables que queremos tener.

Figura 2.12: creación del canal.

Se debe tener presente que el nombre asignado a los campos o variables deben ser idénticos a los nombres de las variables utilizadas en el código de Arduino que queremos ver reflejadas.

- Crear dispositivo MQTT: Una vez ya creado el canal con las correspondientes variables en el banner principal esta un banner que dice “Dispositivos” el cual hay que crear para que nuestro canal se pueda comunicar con nuestro controlador.



Figura 2.13: Dispositivo MQTT.

- Programas en Arduino IDE: Una vez seguido todos los pasos anteriores, a través de la programación presentada en el anexo A se puede ver como realizar la comunicación entre la plataforma y el ESP32 dándonos así los grafico para poder visualizar las variables deseadas.

2.3.6 Herramientas de programación

Las herramientas de programación son software con el propósito de diseñar, configurar y gestionar operaciones de dispositivos electrónicos como microcontroladores o controladores lógicos programables, estas herramientas nos facilitan la escritura en código, incorporar gráficos y programación visual también crear flujos de datos con el fin de automatizar procesos, hay una gran cantidad de herramientas de programación en la actualidad y cada una con un lenguaje de programación que la caracteriza como pueden ser Arduino IDE, Tia Portal, Node-Red, Phyton, Visual Studio Code entra otras.

3 Solución

3.1 Objetivos

- **Objetivo General**

- Diseñar y proponer un sistema de Control que permita garantizar la continuidad del suministro eléctrico en la Planta Receptora ante un corte de red monofásica en la isla y a la vez que permita monitorear localmente variables de funcionamiento de la planta durante un corte de suministro eléctrico.

- **Objetivos Específicos**

- Elaborar un diseño de puesta en marcha automática del grupo electrógeno.
- Modelar un prototipo de sistema de control mediante IOT para el monitoreo de variables entregadas por el grupo electrógeno y su entorno.

3.2 Posibles soluciones que cumplan con el objetivo

- **Sistema híbrido fotovoltaico:**

El sistema híbrido fotovoltaico podría ser ideal para solucionar la problemática que sufre la planta receptora Quiriquina debido a que consiste en la instalación de paneles solares y equipos de almacenamiento de energía sobre o alrededor de las estaciones, abasteciendo de energía la planta, lo cual erradicaría la intermitencia del suministro eléctrico en caso de un corte, adaptando la planta con una fuente de energía autónoma limpia y disponible durante la mayor parte del día debido a estar ubicada en un entorno insular como la Isla Quiriquina donde existe una alta exposición le brindaría a la planta una mayor continuidad operativa, menor desgaste de los sistemas actuales.

Sin embargo, aunque el sistema fotovoltaico aporta autonomía, presenta una serie de limitaciones que se deben considerar antes de una posible instalación, como lo serían la dependencia total de las condiciones climáticas pudiéndose presentar nieblas costeras, temporadas de cielo nublado en invierno lo que dificulta la previsión de generación diaria.

Altos costos iniciales en la implementación debido a que los paneles, estructuras, controladores MPPT y equipos de conversión requieren una alta inversión.

- **Sistemas de alimentación ininterrumpidas:**

La implementación de UPS es altamente beneficiosa para resolver la problemática de intermitencia del suministro eléctrico debido a que un banco de baterías funciona como un respaldo inmediato capaz de entregar energía de manera estable y continua incluso cuando existen caídas repentinas del suministro.

Las UPS como respaldo también puede cumplir con el rol fundamental de estabilizar el voltaje y mitigar picos de señales no deseadas siendo extremadamente útil para evitar daños en los equipos de telecomunicación que posee la planta y dándole mayor calidad al suministro que no siempre es uniforme.

Sin embargo este sistema también presenta desventajas importantes de evaluar ante una implementación como lo serían altos costos de adquisición y mantenimiento, degradación con el tiempo y la temperatura, dependencia del suministro principal para la carga de estas debido a que a pesar de poder mantener la planta energizada por varias horas, ante cortes de suministro muy prolongados puede existir la posibilidad de que se terminen descargando antes de que vuelva la red de alimentación principal degradando las baterías internas de este sistema.

3.3 Plan de acción para lograr los objetivos planteados.

En este proyecto se analizaron los diversos tipos de posibles soluciones a la gran problemática que enfrenta la Planta Receptora Quiriquina como lo son la implementación de sistemas híbridos y/o sistemas de alimentación ininterrumpidas, los cuales resultaban ser económicamente poco

favorables y con resultados que no eran capaces de subsanar por completo la problemática, debido a esto se escogió el método de implementar el control para el generador auxiliar mediante un microcontrolador debido a que ofrece una solución económica, flexible y de código abierto, frente a opciones mucho más costosas con controladores automáticos o tableros ya diseñados para una acción en concreto sin posibilidad de agregar mejoras, gracias a que el ESP32 permite personalizar nuestra secuencia de arranque y monitoreo de variables podemos moldearlo a nuestros requerimientos específicos en el área.

también a diferencia de los otros métodos el ESP32 nos permite tener conectividad a una nube y poder visualizar nuestras variables en tiempo real para tomar acciones pertinentes sobre el estado de la planta, permitiendo llevar un registro histórico del funcionamiento del generador y a la vez poder predecir fallas o mantenimientos que este requiera por sus horas de uso, evitando también la incertidumbre de no conocer en tiempo real la cantidad de combustible que va quedando en el generador, por todo esto el ESP32 es la herramienta ideal para este problema.

Esta solución se llevará a cabo diseñando un prototipo de tablero de transferencia automática el cual cumplirá la función de conmutar el generador en caso de un corte de poder y a la vez subir información como el estado del generador (on,off), nivel de combustible, temperatura y corriente suministrada a la planta. Todo esto lo llevará a cabo un microcontrolador el ESP32 el cual es capaz de interpretar las entradas de datos provenientes de los sensores y a la vez mediante Wifi subir a internet los datos para ser interpretados en la plataforma de la nube Thingspeak a su vez mediante relevadores de distintas envergaduras accionar la partida del generador auxiliar de la planta receptora, una vez el generador se encuentre operando realizar el cambio de red de suministro mediante contactores y en el momento de que regrese el poder principal ser capaz de regresar la alimentación de la red como fuente principal y apagar el generador auxiliar mediante la bomba de combustible.

3.4 Diagrama de flujo Solución



Figura 3.1: Diagrama de flujo funcionamiento del sistema IOT.

3.5 Diseño del circuito

En la figura 3.2 se puede presenciar el diseño del circuito para el tablero este cuenta con el Esp32 como controlador principal del tablero el cual recibe señales de los relés monofásicos de 8 pines que se encuentran en la parte superior estos relés funcionan como relés detectores de fase los cuales se accionan al detectar voltaje en las borneras ubicada en la esquina inferior derecha al ser accionadas conmutan los conectores y dejan pasar 3.3V volts del microcontrolador a las entras de este mismo generando así una señal lógica que el ESP32 puede interpretar como si hay

o no alimentación de la red y/o el generador auxiliar.

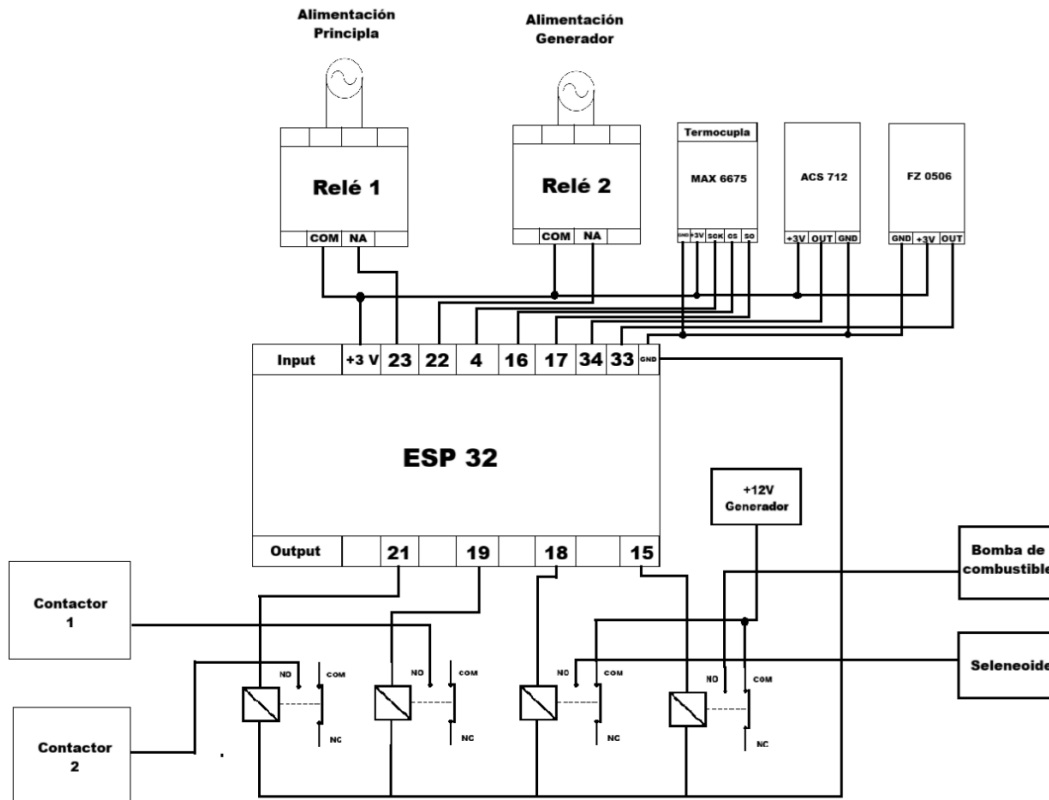


Figura 3.2: Esquema Circuito completo.

Mediante el módulo de 8 relés que está ubicado abajo del Microcontrolador se puede dar el accionamiento tanto a la bomba de combustible del generador (relé n°6), al contacto de encendido del generador (relé n°5) el cual tendrá intervalos de 3 segundos de encendido y 5 segundos de reposo con la finalidad de no ahogar el generador en caso de que no encendiera en el tercer intento se dejara de dar arranque al generador.



Figura 3.3: Solenoide 12V activador de bomba de combustible.

El microcontrolador en el caso de requerirlo podría mediante el relé n°6 del módulo de 8 relés accionar el solenoide de 12V el cual de forma mecánica se contrae dando así el libre flujo de combustible al Generador auxiliar para poder ser encendido.

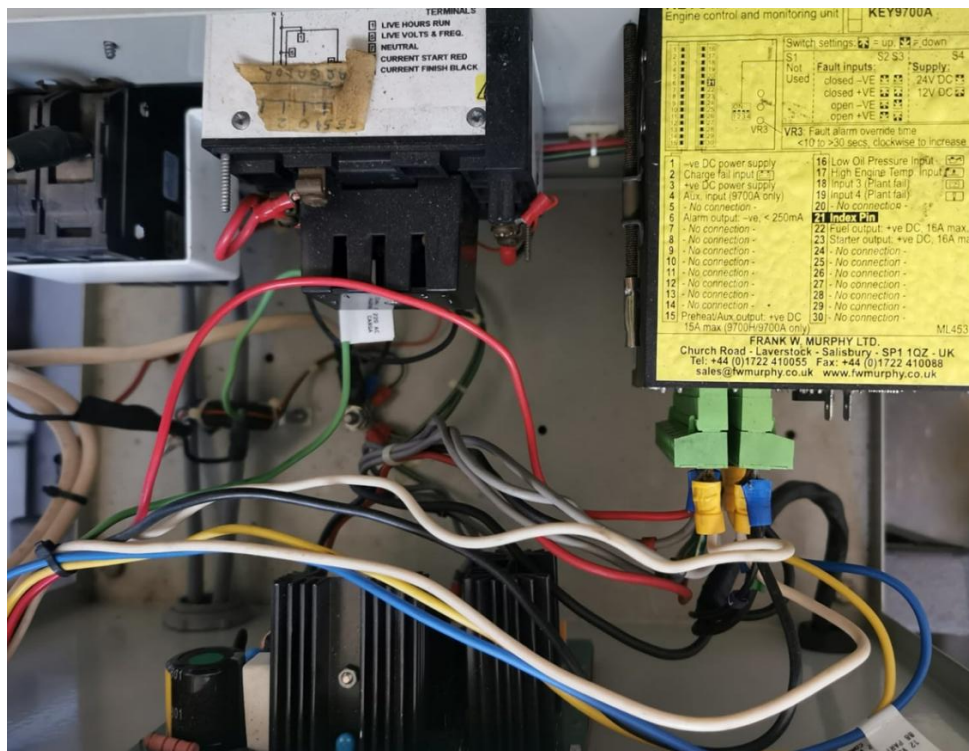


Figura 3.4: bornes de conexión eléctrica Generador auxiliar.

El microcontrolador de ser requerido por algún corte de poder sería capaz de encender el generador a través del relé nº5 del módulo de 8 relés que se conectaría entre al positivo de la tarjeta de carga del generador y el borne de conexión nº23 del Generador, dando contactos de 3 segundos con descansos de 5 segundos hasta completarse los 3 intentos.

El módulo de 8 relés también conmuta los 2 contactores ubicados a la derecha, el contactor de la izquierda es utilizado para la alimentación principal de la red y el contactor de la derecha se utiliza para la alimentación proveniente del generador auxiliar estos 2 contactores son activados mediante sus respectivas bobinas al relé nº7 y 8 del módulo de relés.

El circuito cuenta con una batería de 5V recargable externa que es cargada con un cargador de batería conectado a la alimentación, con la finalidad de mantener la alimentación de el tablero en caso de un corte de poder permitiendo así que el Esp32 sea capaz de ejecutar el accionamiento del generador auxiliar en caso de tener ausencia de suministro eléctrico.

El microcontrolador junto con los sensores presentados en la figura 3.5 es capaz de recolectar e interpretar variables físicas del generador como su temperatura, nivel de combustible y corriente que suministra a la planta, cabe resaltar de que debido a que este circuito es un prototipo son demostradas mediante sensores de bricolaje para su presentación y no sensores mas robustos que puedan soportar condiciones mas adversas.

El microcontrolador mediante este circuito es capaz tanto de detectar fallas en la alimentación principal y conmutar el generador auxiliar es capaz de tomar datos de variables ubicadas en la sala del generador y por medio de internet comunicarse con un servidor en la nube como en este caso sería Thingspeak y de forma local poder analizar las variables del generador en la Radio estación marítima de Talcahuano, cabe destacar que el microcontrolador ante cualquier imprevisto prioriza el accionamiento del generador auxiliar y/o el regreso del suministro eléctrico antes que la transmisión de datos mediante internet.

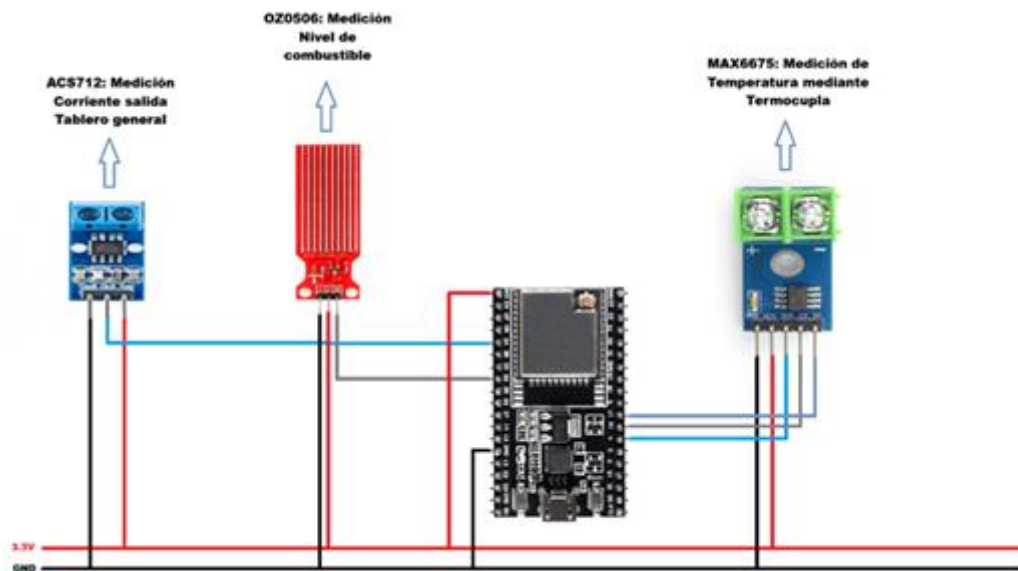


Figura 3.5: Esquema circuito de monitoreo.

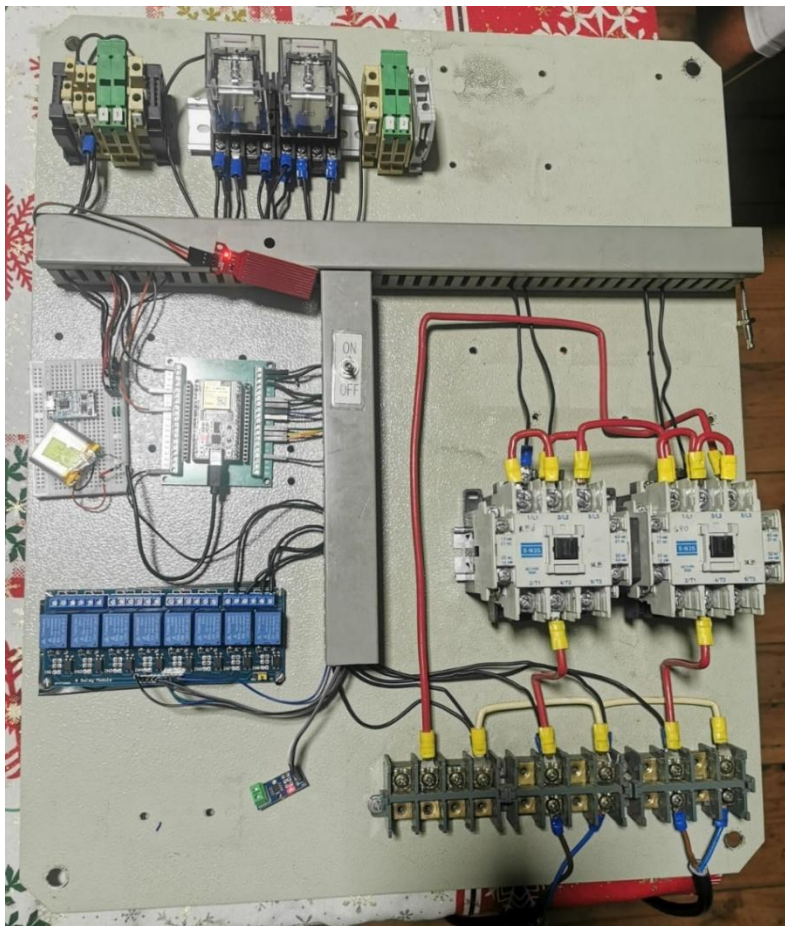


Figura 3.6: Prototipo puesta en marcha.

3.6 Programación del Circuito

Toda la programación del microcontrolador ESP32 se puede ver en el anexo A, cabe resaltar de que para poder ser visualizada y ejecutada se debe contar con los siguientes requisitos.

- Contar con la herramienta de programación **Arduino IDE**
- Descargar las librerías -Max6675 de Rob Tillar: para uso del Max6675
 - ThingSpeak de MathWorks: librería de ThingSpeak
 - ACS712 de Rob Tillar: para el uso del ACS712

El motivo de la utilización de estas librerías es facilitar el uso de hardware y protocolos sin tener que escribir todo el código desde el principio, proveer funciones ya probadas y optimizadas para

ahorrar tiempo en el desarrollo de este, mejorar la legibilidad del programa al abstraer detalles complejos para estos sensores y conexión a la nube de ThingsPeak

3.7 Plataforma en la nube THINGSPEAK

Una vez realizada la comunicación de el microcontrolador con la plataforma en Thingspeak se pueden ver reflejado los siguientes dashboards en la figura 3.7 y 3.8 con las variables que estamos midiendo en tiempo real.

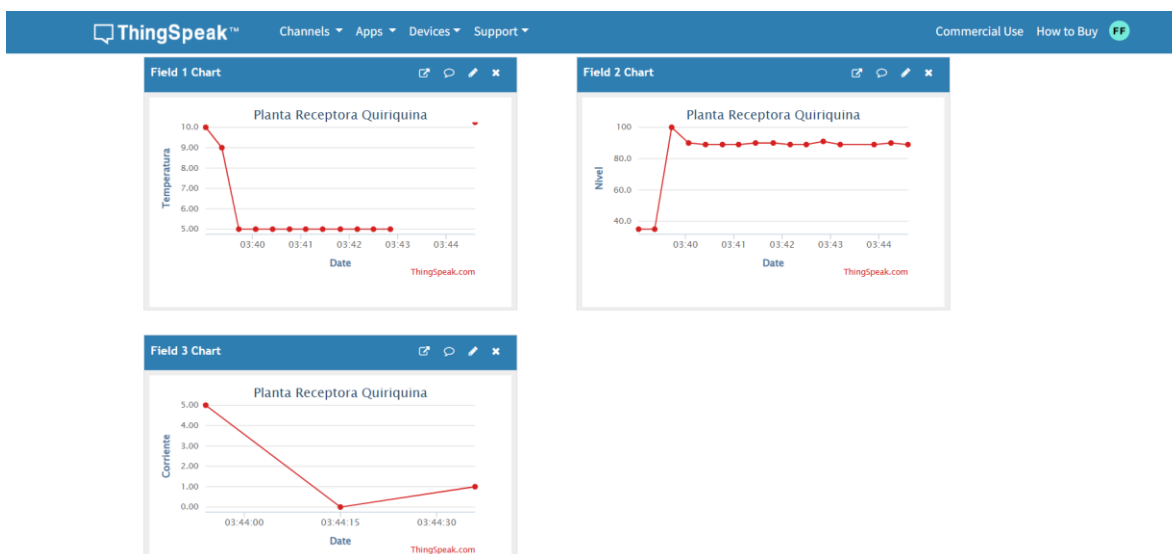


Figura 3.7: Grafico primario entregado por ThingsPeak.

ThingsPeak nos deja realizar adaptaciones a los gráficos que entrega por defecto para una lectura más cómoda y adecuada a la variable que se está leyendo como se ve reflejado en la figura 3.8.

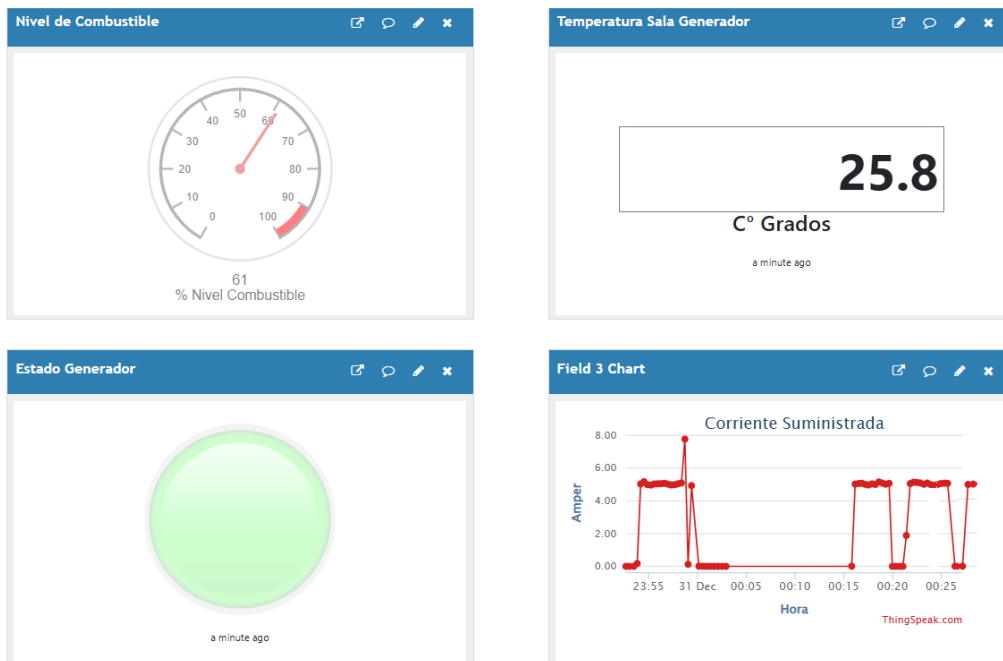


Figura 3.8: Grafico adaptado a gusto del operador.

4 Planificación de trabajo

4.1 Planificación del Prototipo

Una vez ya analizado el problema en la planta receptora Quiriquina y habiendo elegido entre las posibles soluciones para atacar esta problemática, la idea de realizar el prototipo de propuesta de solución mediante un microcontrolador debe ser evaluados factores como los tiempos, costos, recursos y riesgos para asegurar la correcta implementación del sistema.

4.2 Tiempos de Desarrollo del Prototipo

4.2.1 Diseño y definición de arquitectura

En esta fase se establece los requerimientos técnicos del prototipo como la selección del microcontrolador a utilizar como núcleo del sistema y los instrumentos que integraran este tablero como los sensores, relés, fuentes de alimentación, cableado y plataforma en la nube a utilizar. Esta fase el proyecto entre el análisis y la correcta elección de componentes a utilizar puede durar aproximadamente **1 semana**.

4.2.2 Diseño ensamblaje físico y cableado inicial

En esta fase una vez ya seleccionados los componentes a utilizar se procede a la adquisición de componentes para el realizado del prototipo lo cuales debido a su dificultad de adquisición pueden demorarse aproximadamente de una a una semana y media en llegar o conseguirse en tiendas comerciales o envíos externos. Una vez ya teniendo todos los componentes se procede al ensamblado en el panel del tablero distinguiendo la ubicación más adecuada de cada componente y su correcto cableado. Este tiempo fue mayor de lo esperado debido a el correcto ajuste en el cableado, pruebas de continuidad y adecuación de niveles de señal también se realizaban pequeñas modificaciones mediante se iba diseñando debido a todo este esta etapa duro en su totalidad aproximadamente **3 semanas**.

4.2.3 Desarrollo del Programa principal

En esta fase se diseño el programa el cual iba a ser incorporado en el microcontrolador para el correcto accionamiento del cambio de alimentación a la vez como el monitoreo de variables de Nivel, Corriente y Temperatura con su correcta comunicación con la plataforma en la nube de ThingSpeak. Esta etapa duro en su totalidad aproximadamente **2 días**.

4.2.4 Incorporación del software con el hardware

Una vez finalizado el desarrollo del programa principal se le carga este al ESP32 y se le realizan pruebas de funcionamiento al tablero principal con el fin de verificar que el código cargado en el microcontrolador sea adecuado para la correcta medición de las variables y accionamiento de los relés y contactores. En esta etapa también se detectaron inconsistencias entre las lecturas de sensores y accionamientos de los actuadores lo que implica a su vez ajustes dentro de la programación, todo esto tuvo un tiempo de ejecución de **2 semanas**.

4.2.5 Puesta en marcha y finalización del Prototipo

La puesta en marcha se realizo una vez ya incorporado el software con el hardware se verifica que el prototipo cumplía con los requisitos planteados, se registran configuraciones operativas, diagramas y parámetros para ser registrado en el informe. Este proceso duro aproximadamente **2 días**. Logrando así finalizar el prototipo con resultados positivos y cumpliendo los objetivos planteados al principio de este proyecto, como se puede ver reflejado en la figura 4.1 y 4.2.

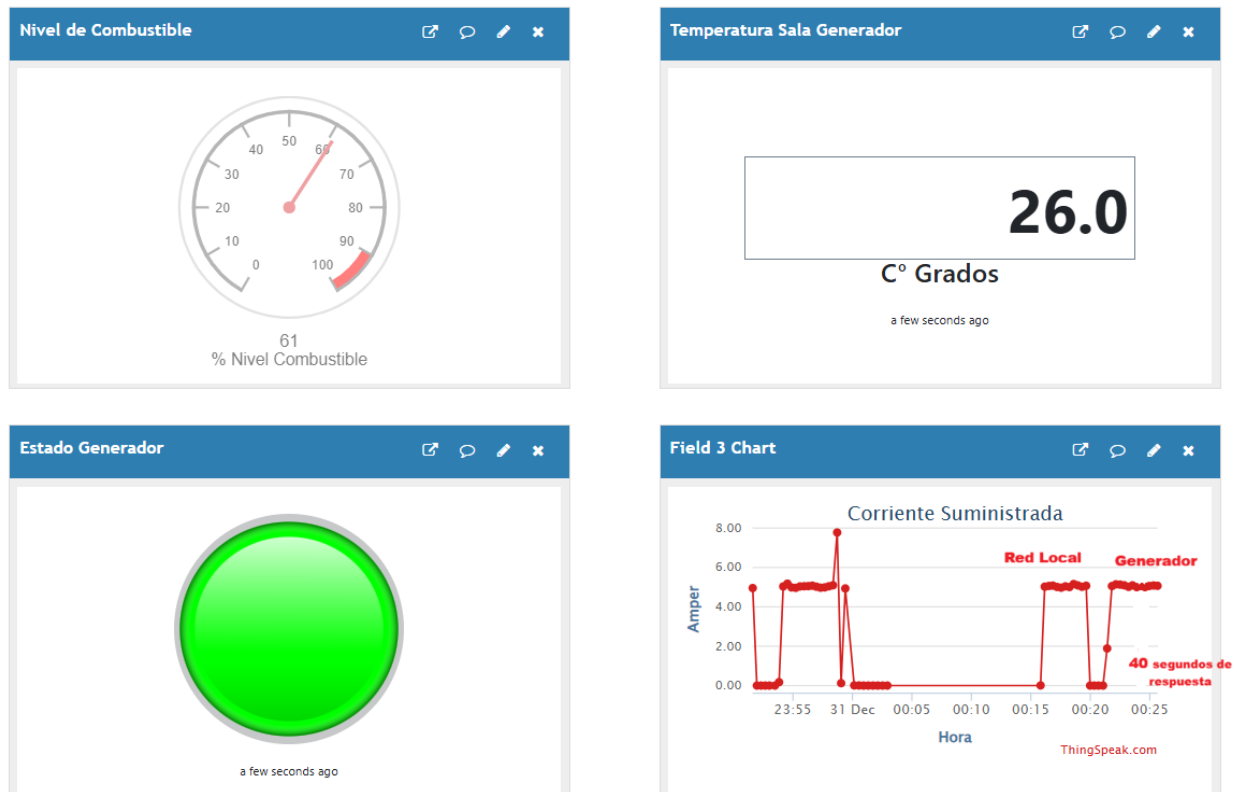


Figura 4.1: Cambio de red local a generador auxiliar con 40 segundos de retardo.

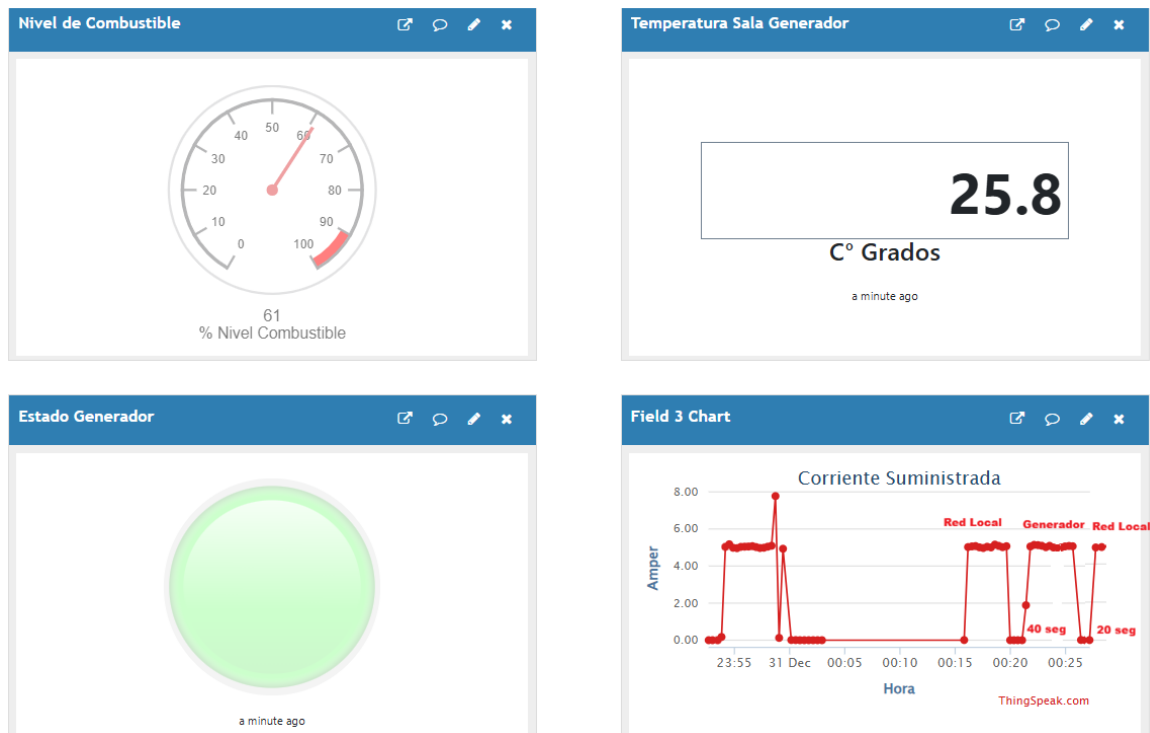


Figura: 4.2: Cambio de red local a generador auxiliar y nuevamente red local.

El tiempo total de diseño, ensamblaje y puesta en marcha llevo alrededor de 6 semanas y 4 días, teniendo presente que estos tiempos son acotados debido al horario disponible por el estudiante pudiendo haber variaciones.

4.3 Costos de la elaboración del prototipo

La identificación de costos para el desarrollo, implementación y puesta en marcha de este prototipo permiten determinar la viabilidad técnica y económica evidenciando los recursos utilizados en su ejecución.

- **Costos directos del prototipo:**

Componentes	Cantidad	Costo unitario CLP	Costo total CLP
ESP32	01 unidades	\$11500	\$11500
Módulo de 8 relé	01 unidades	\$9230	\$9230
Sensor de temperatura Max6675	01 unidades	\$7890	\$7890
Batería de litio 3.7v 1000 mAh	01 unidades	\$5500	\$5500
Sensor de corriente Acs712	01 unidades	\$4190	\$4190
Sensor de nivel fz0506	01 unidades	\$2300	\$2300
Mitsubishi S-N25 AC-220V Contactor	02 unidades	\$41530	\$83060
Base de expansión ESP32	01 unidades	\$3000	\$3000

Armario Metálico 500 x 400 x 250 mm Ekoline	01 unidades	\$83653	\$83653
Canaleta Ranurada 25x25mm 2 metros Color Gris	01 unidades	\$4446	\$4446
Mini Relé Universal 8 Pines 2NA-2NC 220V Ekoline	02 unidades	\$11833	\$23666
Base P/Mini Relé Universal 8 Pines Ekoline	02 unidades	\$3885	\$7770
Riel Din Simétrico 1 m Ekoline	01 unidades	\$2395	\$2395
Borne de Conexión Riel Din DC2.5 0.5/2.5mm2 Gris Biriél Degson	10 unidades	\$435	\$4350
Cable THHN 14 AWG Negro (100 Mts)	01 x 12 metros	\$38590	\$4630
Cable THHN 10 AWG Negro (100 Mts)	01 x 2 metros	\$94775	\$1896

Terminales tipo Horquilla	100 unidades	\$4000	\$4000
Total	-	-	\$263476 CLP

5 Resultados y Conclusión

5.1 Resultado obtenido

El resultado obtenido en el proyecto desarrollado cumplió completamente los objetivos planteados, demostrando que un sistema basado en un microcontrolador como el Esp32 puede resolver de manera satisfactoria el problema existente en la Planta receptora Quiriquina logrando así ejecutar secuencias de forma automática de transferencia eléctrica y monitoreo continuo de una manera viable, flexible y adaptable a necesidades reales de la planta reduciendo la dependencia de intervención humana logrando así que en caso de algún corte del suministro eléctrico el operador tenga que trasladarse físicamente a prender el generador. También mediante la integración de la plataforma en la nube ThingSpeak se confirma que es posible visualizar en línea las variables críticas del generador, adoptando información clave para una operación segura y eficiente.

Se logro también evidenciar que este sistema a diferencia de un ATS o un Controlador Deep Sea la solución planteada es económica, permite modificaciones de forma libre y puede ampliarse a nuevos sensores o funciones.

5.2 Conclusión

Desde la experiencia adquirida durante el desarrollo de este proyecto se logro concluir que la elaboración de este prototipo permitió validar técnicamente la solución propuesta, sino que también represento un aprendizaje significativo en cuanto a la integración de instrumentación, programación C++ y tecnologías IOT aplicadas al control de sistemas eléctricos y electrónicos habiendo logrado que el prototipo ejecutara correctamente la detección de falta de alimentación para la planta, el accionamiento correcto del generador auxiliar y el monitoreo de variables en

tiempo real. Permitiendo confirmar que todo esto era posible utilizando un ESP32 de manera eficiente y económica.

Sin embargo cabe resaltar que a pesar de el buen desempeño obtenido en las pruebas de laboratorio, para ser llevado e incorporado a un entorno real específicamente dentro de la planta Quiriquina es necesario remplazar los sensores de bricolaje utilizados por sensores mas robustos capaces de soportar los factores como humedad, vibraciones, variaciones térmicas y posibles interferencias electromagnéticas, con la finalidad de garantizar mediciones más precisas y confiables.

Este proyecto no solo cumplió con los objetivos propuestos también sirvió para afianzar competencias practicas relevantes para el ejercicio profesional y comprender mejor los desafios que se encuentran presente en el mundo del control e instrumentación industrial.

Bibliografía

Espressif Systems. (s.f.). ESP32 Series – Documentación.

https://documentation.espressif.com/esp-dev-kits/en/latest/esp32/esp32-devkitc/user_guide.html?q=%20esp32%20Devkitc%20v4%20

MathWorks. (s.f.). ThingSpeak - Documentación.

https://thingspeak.mathworks.com/pages/learn_more

Tillaart, R. (s.f.). MAX6675 Arduino Libreria -Repositorio GitHub.

<https://github.com/RobTillaart/MAX6675>

Tillaart, R. (s.f.). ACS712 Arduino Libreria -Repositorio GitHub.

<https://github.com/RobTillaart/ACS712>

MathWorks. (s.f.). ThingSpeak Arduino Libreria -Repositorio GitHub.

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

Geekflare Team. (s.f.). Protocolos de comunicación IoT.

<https://geekflare.com/es/mqtt-vs-coap-vs-http/>

IBM. (s.f.). Internet de las cosas (IoT). IBM.

<https://www.ibm.com/mx-es/think/topics/internet-of-things>

Mokosmart. (s.f.). Guía de tecnologías de redes IoT. Mokosmart.

<https://www.mokosmart.com/es/iot-networking-technologies-guide/>

Analog Devices. (2003). MAX6675- Hoja de datos. Analog Devices.

<https://www.analog.com/media/en/technical-documentation/data-sheets/max6675.pdf>

Allegro Microsystems. (2013). ACS712- Hoja de datos. Allegro Microsystems.

<https://cdn.sparkfun.com/assets/4/a/a/0/8/ACS712.pdf>

ANEXO A

En este Anexo se adjunta la programación utilizada en Arduino IDE para el ESP32.

```
//Proyecto de título Planta Receptora Quiriquina

//-----Librerías -----
-----
#include <WiFi.h>
#include <ThingSpeak.h>
#include <max6675.h>
#include "secrets.h"
#include <ACS712.h>

//-----definición de variables-----
-----
char ssid[]=SECRET_SSID;
char pass[]=SECRET_PASS;
WiFiClient client;
unsigned long myChannelNumber = SECRET_CH_ID;
const char *myWriteAPIKey = SECRET_WRITE_APIKEY;

byte relay_1= 23;
byte relay_2= 22;
byte contactor_1= 19;
byte contactor_2= 21;
byte gen_ON= 18;
byte gen_OFF= 15;

int intentos = 0;

// >>>> VARIABLE DE ESTADO REAL DEL GENERADOR <<<<<
int estadoGenerador = 0; // 0 = APAGADO | 1 = ENCENDIDO

int sck=4;
int cs=16;
int so=17;

float datos_n=33;
const int ACS_712 = 34;
float Temperatura;
float Nivel;
float mVperAmp = 100;

MAX6675 termopar(sck,cs,so);
ACS712 sensorACS(ACS_712, 3.3, 4095, mVperAmp);

//-----Conexion al WIFI-----

unsigned long ultimoIntentoWiFi = 0;
```

```

unsigned long intervaloWiFi = 5000;

void Wificonexion() {
  if (WiFi.status() == WL_CONNECTED) return;

  unsigned long ahora = millis();
  if (ahora - ultimoIntentoWiFi >= intervaloWiFi) {
    Serial.print("Intentando conectar WiFi a: ");
    Serial.println(ssid);
    WiFi.begin(ssid, pass);
    ultimoIntentoWiFi = ahora;
  }
}

void setup() {
  pinMode(relay_1, INPUT);
  pinMode(relay_2, INPUT);
  pinMode(contactor_1, OUTPUT);
  pinMode(contactor_2, OUTPUT);
  pinMode(gen_ON, OUTPUT);
  pinMode(gen_OFF, OUTPUT);

  analogReadResolution(12);
  analogSetPinAttenuation(ACS_712, ADC_11db);
  sensorACS.autoMidPoint(50);

  Serial.begin(115200);
  while(!Serial){;}

  WiFi.mode(WIFI_STA);
  ThingSpeak.begin(client);
  delay(1000);
}

void loop() {
  //-----Revisar y conectar a Wifi-----
  Wificonexion();

  //-----Sistema de control Grupo electrógeno-----
  --
  if (digitalRead(relay_1) == HIGH && digitalRead(relay_2) == LOW)
  {
    digitalWrite(contactor_1,LOW);
    digitalWrite(contactor_2,HIGH);
    digitalWrite(gen_OFF,HIGH);
    intentos = 0;
    estadoGenerador = 0;
  }
  else if (digitalRead(relay_1) == LOW && digitalRead(relay_2) == LOW)
  {
    digitalWrite(contactor_1,HIGH);
    digitalWrite(contactor_2,HIGH);
  }
}

```

```

digitalWrite(gen_OFF,LOW);

if (intentos < 3) {
  while (digitalRead(relay_2) == LOW && intentos < 3) {

    Serial.print("Intento de arranque #");
    Serial.println(intentos + 1);

    digitalWrite(gen_ON, LOW);
    delay(5000);
    digitalWrite(gen_ON, HIGH);

    Serial.println("Esperando estabilización...");
    delay(3000);

    intentos++;

    if (digitalRead(relay_2) == HIGH) {
      estadoGenerador = 1;
      break;
    }
  }
}
else {
  Serial.println("Generador no arrancó después de 3 intentos.");
  estadoGenerador = 0;
  delay(3000);
}
}
else if (digitalRead(relay_2) == HIGH && digitalRead(relay_1) == LOW)
{
  digitalWrite(contactor_1,HIGH);
  digitalWrite(contactor_2,LOW);
  digitalWrite(gen_ON,HIGH);
  digitalWrite(gen_OFF,LOW);
  intentos = 0;
  estadoGenerador = 1;
}
else if (digitalRead(relay_2) == HIGH && digitalRead(relay_1) == HIGH)
{
  digitalWrite(contactor_1,HIGH);
  digitalWrite(contactor_2,HIGH);
  digitalWrite(gen_ON,HIGH);
  digitalWrite(gen_OFF,HIGH);
  intentos = 0;
  estadoGenerador = 0;
}

//-----Lectura de Variable Nivel -----
datos_n = analogRead(33);
Nivel = map(datos_n, 0, 2115, 0, 100);

```

```
//----- Lectura de Variable Corriente -----  
float Frecuencia = sensorACS.mA_AC(50);  
float Corriente = Frecuencia / 1000.0;  
  
//-----Lectura de Variable Temperatura -----  
  
Temperatura = termopar.readCelsius();  
  
//-----Envió de Variables a Nube ThingSpeak-----  
  
ThingSpeak.setField(1, Temperatura);  
ThingSpeak.setField(2, Nivel);  
ThingSpeak.setField(3, Corriente);  
ThingSpeak.setField(4, estadoGenerador);  
  
int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);  
  
//----- Verificación Monitor Serial-----  
  
Serial.print("Corriente: ");  
Serial.print(Corriente);  
Serial.print(" A | Estado Generador: ");  
Serial.println(estadoGenerador);  
  
//-----Verificación de Comunicación-----  
  
if (x == 200) {  
    Serial.println("Actualización del canal exitosa.");  
} else {  
    Serial.println("Error ThingSpeak HTTP " + String(x));  
}  
  
delay(20000);  
}
```