



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES INDUSTRIALES

“IMPLEMENTACIÓN DE UN DISPOSITIVO PARA MONITOREAR
VARIABLES METEOROLÓGICAS EN LA PISTA DE DESPEGUE DEL
CLUB PARAPENTE CHIMBORAZO CON UNA APP ANDROID”

TRABAJO DE TITULACIÓN

TIPO: PROYECTO TÉCNICO

Presentado para otorgar el grado académico de:

INGENIERO EN ELECTRÓNICA CONTROL Y REDES
INDUSTRIALES

AUTORES: JHONATAN RAÚL CORONEL QUISHPI

MARCOS DAVID ESPINOSA MACHUCA

TUTOR: ING. ISIDORO ENRIQUE TAPIA SEGARRA

Riobamba-Ecuador

2019

©2019, Jhonatan Raúl Coronel Quishpi; Marcos David Espinosa Machuca

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

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES INDUSTRIALES

El tribunal del trabajo de titulación certifica que: El trabajo de investigación: Dispositivo tecnológico “IMPLEMENTACIÓN DE UN DISPOSITIVO PARA MONITOREAR VARIABLES METEOROLÓGICAS EN LA PISTA DE DESPEGUE DEL CLUB PARAPENTE CHIMBORAZO CON UNA APP ANDROID” de responsabilidad de los señores JHONATAN RAUL CORONEL QUISHPI Y MARCOS DAVID ESPINOSA MACHUCA, ha sido minuciosamente revisado por los Miembros del Tribunal del trabajo de titulación, quedando autorizada su presentación.

FIRMA **FECHA**

Ing. Washington Gilberto Luna Encalada
**DECANO DE LA FACULTAD DE
INFORMÁTICA Y ELECTRÓNICA.**

Ing. Freddy Enrique Chávez Vázquez
**DIRECTOR DE LA ESCUELA DE
INGENIERÍA ELECTRÓNICA EN
CONTROL Y REDES INDUSTRIALES**

Ing. Isidoro Enrique Tapia Segarra
**DIRECTOR DEL TRABAJO DE
TITULACIÓN**

Ing. Wilson Armando Zuñiga Vinuesa
MIEMBRO DEL TRIBUNAL

Nosotros, Jhonatan Raúl Coronel Quishpi Y Marcos David Espinosa Machuca, somos responsables de las ideas, doctrinas y resultados expuestos en esta tesis y el patrimonio intelectual de la Tesis de Grado pertenece a la Escuela Superior Politécnica de Chimborazo.

Jhonatan Raúl Coronel Quishpi

Marcos David Espinosa Machuca

DEDICATORIA

El presente trabajo de titulación se lo dedico a mi familia por ser un apoyo incondicional tanto en los aspectos personales como en el proceso de formación, en especial a mis padres por ser un ejemplo de superación, humildad y éxito; a cada una de las personas que me brindaron un consejo o una palabra de aliento cuando más fue necesario.

Jhonatan

Primero que nada, quiero dedicar este logro al grande, a mi Dios todopoderoso que me ha guiado y brindado la sabiduría necesaria para cumplir esta gran meta de mi vida, y por supuesto a mis queridos padres, que desde un principio han sido mi motivo para llegar hasta aquí y enorgullecerlos, por toda la atención y paciencia que me han tenido día a día desde que comencé mis estudios sin importar los sacrificios que hagan, también quiero dedicar este éxito a mi hermano y hermana que desde lejos siempre han estado pendiente de mí y de mi carrera universitaria, a mi abuelita que desde el cielo se llena de alegría y gozo al verme consagrado como todo un profesional y a todo el resto de mi familia que de una u otra forma me han ayudado a seguir adelante y conseguir este objetivo.

Marcos

AGRADECIMIENTO

Mis agradecimientos más sinceros a mis padres Washington y Patricia por el sacrificio que hicieron estos duros años, por su apoyo incondicional y por sus buenos consejos especialmente a Mamá por ser el faro de la familia, el reflejo de lo que soy, mi paño de lágrimas y el refugio en esos días malos , a mis hermanos Antony y Estefani por apoyarme y darme ánimos, a mis abuelitos por transmitirme su sabiduría, experiencia y el afán de superación; a esa persona que estuvo conmigo desde el principio dándome días de felicidad, que confió más en mí que yo mismo y me hizo madurar y ser mejor persona; a mi compañero de este trabajo de titulación por eliminar tensiones y siempre mostrar una cara positiva a los problemas; a todas las personas que confiaron en mí y que de una u otra manera colaboraron con este logro.

Jhonatan

Quiero agradecer a mi Dios por alcanzar este logro tan importante en mi vida, a mis queridos padres Norman y Marisol por siempre estar apoyándome a seguir adelante con mi carrera y nunca ceder ante los problemas que se presenten y por los sacrificios de cada uno de ellos que los hacían con todo el amor del mundo solo para verme feliz, a mis hermanos Andrés y Diana que de una u otra manera estuvieron al pendiente en toda mi carrera y en ocasiones ayudándome a cumplir con las obligaciones de mi carrera, a mis compañeros y amigos que me brindaron momentos agradables e inolvidables en esta etapa de mi vida, a mi compañero de este proyecto que se convirtió en un hermano ayudándome a solucionar problemas tanto personales como de nuestra carrera y apoyándome incondicionalmente para cumplir con este objetivo y por supuesto a toda mi familia que siempre confiaron en mí, se los agradezco de corazón.

Marcos

TABLA DE CONTENIDO

RESUMEN.....	xvi
ABSTRACT.....	xvii
INTRODUCCIÓN	1
CAPÍTULO I.....	5
1. MARCO TEÓRICO	5
1.1 El parapente.....	5
1.2 Condiciones meteorológicas óptimas para practicar este deporte.....	6
1.3 Estaciones meteorológicas automáticas.....	7
1.4 Variables a medir en la estación meteorológica	7
1.4.1 <i>Temperatura del aire</i>	7
1.4.2 <i>Humedad relativa</i>	8
1.4.3 <i>Presión atmosférica</i>	8
1.4.4 <i>Velocidad y dirección del viento</i>	8
1.4.5 <i>Presencia de lluvia</i>	9
1.5 Descripción de los tipos sensores utilizados en la estación meteorológica	9
1.5.1 <i>Sensor de humedad relativa</i>	9
1.5.2 <i>Sensor de Temperatura</i>	10
1.5.3 <i>Sensor de presión atmosférica</i>	11
1.5.4 <i>Sensor detector de lluvia</i>	12
1.5.5 <i>Sensor de velocidad y dirección del viento</i>	12
1.6 Tarjetas controladoras.....	13
1.7 Transmisión de datos	14
1.7.1 <i>Red GPRS (Global Packet Radio System)</i>	14

1.8	Sistema fotovoltaico	15
<i>1.8.1</i>	<i>Energía Solar</i>	<i>16</i>
<i>1.8.2</i>	<i>Componentes de un sistema fotovoltaico.....</i>	<i>16</i>
<i>1.8.3</i>	<i>Elementos de Protección</i>	<i>18</i>
1.9	Almacenamiento de datos.....	19
1.10	Aplicaciones móviles	20
<i>1.10.1</i>	<i>Android</i>	<i>20</i>
<i>1.10.2</i>	<i>Entorno de desarrollo Android Studio</i>	<i>22</i>
 CAPÍTULO II		23
2.	MARCO METODOLÓGICO.....	23
2.1	Requerimientos del dispositivo	23
2.2	Diseño de la arquitectura del dispositivo	23
2.3	Diseño de las etapas del dispositivo	24
<i>2.3.1</i>	<i>Etapas de lectura y visualización de las variables meteorológicas</i>	<i>25</i>
<i>2.3.2</i>	<i>Etapas de comunicación y transmisión de datos</i>	<i>25</i>
<i>2.3.3</i>	<i>Etapas de alimentación y energía</i>	<i>26</i>
2.4	Selección del hardware del dispositivo y sensores.....	26
<i>2.4.1</i>	<i>Arduino</i>	<i>27</i>
<i>2.4.2</i>	<i>Sensor de lluvia</i>	<i>30</i>
<i>2.4.3</i>	<i>Sensores de velocidad y dirección del viento</i>	<i>31</i>
2.5	Selección del módulo de comunicación.....	32
<i>2.5.1</i>	<i>Módulo Shield</i>	<i>32</i>
2.6	Módulo LM2596.....	33
2.7	Módulo LCD 2004A	34
2.8	Diseño de la etapa de alimentación.....	35
<i>2.8.1</i>	<i>Energía diaria consumida.....</i>	<i>35</i>
<i>2.8.2</i>	<i>Dimensión del panel fotovoltaico</i>	<i>37</i>

2.8.3	<i>Dimensión de la batería</i>	38
2.8.4	<i>Dimensión Controlador de carga</i>	40
2.9	Esquema de conexiones.....	40
2.10	Software del dispositivo	42
2.10.1	<i>Requerimientos del software del dispositivo</i>	42
2.11	Software para la lectura, muestra y envío de datos	42
2.11.1	<i>Programación en el IDE de Arduino</i>	43
2.12	Software para el desarrollo de la aplicación.....	47
2.12.1	<i>Programación en Android Studio</i>	47
 CAPÍTULO III		49
3.	PRUEBAS Y RESULTADOS	49
3.1	Validación de sensores	49
3.1.1	<i>Validación de medición de temperatura</i>	50
3.1.2	<i>Validación de medición de presión atmosférica</i>	51
3.1.3	<i>Validación de medición de humedad relativa</i>	53
3.1.4	<i>Validación de medición de velocidad del viento</i>	54
3.1.5	<i>Validación de medición de la dirección del viento</i>	56
3.1.6	<i>Validación de medición de la lluvia</i>	58
3.2	Repetitividad de los sensores	59
3.3	Evaluación de envío de datos al servidor	61
3.4	Análisis del consumo de datos móviles	63
3.5	Análisis de costos	63
 CONCLUSIONES		66
 RECOMENDACIONES		67
 BIBLIOGRAFÍA		
 ANEXOS		

ÍNDICE DE TABLAS

Tabla 1-1:	Comparación entre tipos de sensores electrónicos de humedad relativa	9
Tabla 2-1:	Características de sensores de temperatura	10
Tabla 3-1:	Comparación entre los tipos de sensores de presión atmosférica.....	11
Tabla 4-1:	Comparación entre los tipos de señal entregada por los sensores de velocidad y dirección del viento.....	13
Tabla 5-1:	Comparación entre tarjetas controladoras	13
Tabla 6-1:	Características de las tecnologías inalámbricas.....	14
Tabla 7-1:	Características de las plataformas IoT	20
Tabla 1-2:	Características técnicas de las tarjetas controladoras	27
Tabla 2-2:	Características técnicas de los sensores de humedad	28
Tabla 3-2:	Características técnicas de los sensores de presión atmosférica y temperatura.....	29
Tabla 4-2:	Características técnicas de los sensores de lluvia.....	30
Tabla 5-2:	Características técnicas de los sensores de velocidad y dirección del viento	31
Tabla 6-2:	Características técnicas de los módulos de comunicación	33
Tabla 7-2:	Características técnicas del módulo LM2596.....	34
Tabla 8-2:	Características técnicas del módulo LCD 2004 A.....	35
Tabla 9-2:	Calculo de la energía total diaria consumida por el dispositivo	36
Tabla 10-2:	Características técnicas del panel fotovoltaico.....	38
Tabla 11-2:	Características técnicas de la batería solar	39
Tabla 12-2:	Características técnicas del controlador de carga	40
Tabla 1-3:	Comparación de los valores de temperatura medidos entre el HTC-1 y el BMP180	50

Tabla 2-3:	Comparación de los valores de presión atmosférica medidos entre el barómetro clásico de mercurio y el BMP180	52
Tabla 3-3:	Comparación de los valores de humedad relativa medidos entre el HTC-1 y el DHT11.....	53
Tabla 4-3:	Comparación de los valores de velocidad del viento medidos entre el SkyWatch Xplore 1 y el SY-FS2	55
Tabla 5-3:	Valores de la dirección del viento medidos con el SY-FS2	57
Tabla 6-3:	Medición de la presencia de lluvia con el módulo sensor FC-37	58
Tabla 7-3:	Repetitividad de los sensores evaluados.....	59
Tabla 8-3:	Resumen de datos enviados y recibidos en la plataforma IoT.....	63
Tabla 9-3:	Consumo de datos del dispositivo	63
Tabla 10-3:	Costo de la implementación del dispositivo.....	64

ÍNDICE DE FIGURAS

Figura 1-1	Partes del parapente.....	06
Figura 2-1	Placa para detección de lluvia	12
Figura 3-1	Elementos de una red GPRS	15
Figura 4-1	Panel Fotovoltaico.....	17
Figura 5-1	Modulo controlador de carga	18
Figura 6-1	Porta fusibles y fusibles.....	19
Figura 7-1	Resumen arquitectura Android.....	21
Figura 1-2	Esquema de la arquitectura del dispositivo	24
Figura 2-2	Etapa de lectura y visualización de las variables meteorológicas	25
Figura 3-2	Etapa de comunicación y transmisión de datos.....	26
Figura 4-2	Etapa de alimentación y energía.....	26
Figura 5-2	Arduino MEGA 2560.....	28
Figura 6-2	Sensor DHT11	29
Figura 7-2	Sensor BMP180.....	30
Figura 8-2	Modulo sensor YL-38	31
Figura 9-2	Sensor SY-FS2	32
Figura 10-2	Sensor SY-FX2.....	32
Figura 11-2	Modulo GPRS SIM900	33
Figura 12-2	Modulo LM2596	34
Figura 13-2	Modulo LCD 2004 A	34
Figura 14-2	Panel fotovoltaico.....	38
Figura 15-2	Batería Solar	39
Figura 16-2	Controlador de carga	40

Figura 17-2	Esquema de conexiones electrónica.....	41
Figura 18-2	Diagrama de flujo de lectura y visualización de las variables meteorológicas.....	44
Figura 19-2	Diagrama de flujo de comunicación y envío de información a la base de datos.....	46
Figura 20-2	Diagrama de flujo de comunicación de la aplicación con plataforma IoT	48
Figura 1-3	Dispositivo final implementado	49
Figura 2-3	HTC-1.....	50
Figura 3-3	Barómetro clásico de mercurio.....	52
Figura 4-3	SkyWatch Xplore 1	47
Figura 5-3	Aplicación de brújula	57
Figura 6-3	Datos procesados listos para el envío	61
Figura 7-3	Datos recibidos en ThingSpeak.....	62
Figura 8-3	Datos visualizados en la aplicación móvil.....	62

ÍNDICE DE ANEXOS

ANEXO A:	Datasheet de Arduino MEGA 2560
ANEXO B:	Datasheet del sensor DHT11
ANEXO C:	Datasheet del sensor BMP180
ANEXO D:	Datasheet del módulo sensor YL-38
ANEXO E:	Datasheet del sensor de velocidad del viento SY-FS2
ANEXO F:	Datasheet del sensor de velocidad del viento SY-FX2
ANEXO G:	Datasheet del módulo GPRS SIM900
ANEXO H:	Datasheet del regulador de voltaje LM2596
ANEXO I:	Datasheet del módulo LCD 2004A
ANEXO J:	Datasheet del panel fotovoltaico
ANEXO K:	Datasheet de la batería
ANEXO L:	Datasheet del controlador de carga
ANEXO M:	Código del programa de Arduino
ANEXO N:	Código de la programación en Android Studio
ANEXO O:	Interfaz de la aplicación
ANEXO P:	Datos recibidos en plataforma IoT

ÍNDICE DE ABREVIATURAS

°C:	Grados celsius
hPa:	Hectopascales
GPRS:	Servicio general de paquetes vía radio
Ic:	Corriente de carga
IoT:	Internet de las cosas
App:	Aplicación
LCD:	Pantalla de cristal liquido
P:	Potencia nominal
EDC:	Energía diaria consumida
H:	Tiempo de funcionamiento
W/H:	Watts por hora
ET:	Energía total diaria consumida.
η:	Eficiencia.
NT:	Número de paneles
HPS:	Horas de pico solar
Pp:	Potencia pico del panel fotovoltaico
Fg:	Factor global de perdidas
C:	Capacidad de la batería.
D:	Días de autonomía
Vbat:	Voltaje de la batería
Pd:	Máxima profundidad de descarga

RESUMEN

En este trabajo se diseñó e implementó un dispositivo para monitorear las variables meteorológicas en la pista de despegue del club parapente Chimborazo a través de una app Android. Las variables medidas fueron la temperatura, presión atmosférica, humedad relativa, lluvia, velocidad y dirección del viento, las cuales fueron procesadas con la tarjeta controladora Arduino MEGA 2560 para luego ser enviados mediante el módulo GPRS SIM 900 que permitió comunicarse a internet y guardar la información en la plataforma Thingspeak. A continuación, se diseñó una aplicación con sistema Android para mostrar los datos medidos con lo que se decidirá si existen las condiciones necesarias para volar. Para la etapa de alimentación se optó por el uso de la energía solar mediante un panel fotovoltaico de 100 W con una batería de 55 Ah y un controlador de carga permitiendo así tener una autonomía de trabajo de hasta 2 días cuando no haya presencia de luz solar. Analizando las pruebas realizadas al dispositivo se obtuvo como resultado que el 90,83% de los datos leídos llegan correctamente a la plataforma IoT y por lo tanto se pueden visualizar en la aplicación móvil. Este resultado es aceptable y representa un consumo de la red móvil de 270 Mb por lo que sería necesario el plan mínimo mensual ofrecido por el proveedor. Además, el costo final del dispositivo es de \$1117, lo que se considera como un costo aceptable teniendo en cuenta que los dispositivos similares en el mercado no brindan el servicio de la aplicación ni el de autonomía energética. Se recomienda investigar sobre la posibilidad de integrar un dispositivo móvil que permita ser acoplado en los equipos del piloto para poder expandir el monitoreo tanto al despegue como al aterrizaje y durante el vuelo.

PALABRAS CLAVE

<TECNOLOGIA Y CIENCIAS DE LA INGENIERIA> <CONTROL AUTOMATICO>
<DISPOSITIVO ASISTENTE DE DESPEGUE> <MONITOREAR VARIABLES METEOROLÓGICAS> <ARDUINO> < GLOBAL PACKET RADIO SERVICE (GPRS)>
<BASE DE DATOS> <APLICACIÓN MÓVIL (APP)> <SISTEMA FOTOVOLTAICO><GPRS SIM 900><ARDUINO MEGA 2560>

ABSTRACT

In this work a device was designed and used to monitor the weather variables on the runway of the Chimborazo paragliding club through an Android app. The measured variables were temperature, atmospheric pressure, relative humidity, rainfall, speed and wind direction, which were processed with the Arduino MEGA 2560 controller card and then sent through the GPRS SIM 900 module that allowed to communicate to internet and save the information on the Thingspeak platform. Next, an application with Android system was designed to show the measured data with what will be decided if there are necessary conditions to fly. For the feeding stage, we opted for the use of solar energy through a 100 W photovoltaic panel with a 55 Ah battery and a charge controller, thus allowing a working autonomy of up to 2 days when there is no sunlight. Analyzing the tests performed on the device, it was obtained that 90.83% of the read data arrive correctly to the IoT platform and therefore can be viewed in the mobile application. This result is acceptable and represents a consumption of the mobile network of 270 Mb, which would require the monthly minimum plan offered by the provider. In addition, the final cost of the device is \$ 1117, which is considered an acceptable cost taking into account that similar devices in the market do not provide the service of the application nor the energy autonomy. It is recommended to investigate the possibility of integrating a mobile device that allows it to be coupled in the pilot's equipment in order to expand the monitoring of both take-off, landing and during the flight.

KEYWORDS

<TECHNOLOGY AND SCIENCE OF ENGINEERING> <AUTOMATIC CONTROL> <ASSISTANT DEPARTURE DEVICE> <MONITORING METEOROLOGICAL VARIABLES> <ARDUINO> <GLOBAL PACKET RADIO SERVICE (GPRS) > <THINGSPEAK> <PHOTOVOLTAIC SYSTEM> <GPRS SIM 900> <ARDUINO MEGA 2560>

INTRODUCCIÓN

El parapente es una aeronave fabricada de tela y que utiliza sogas capaces de ser manipuladas para despegar y aterrizar. Este deporte extremo se practica en todas las regiones del territorio ecuatoriano, de acuerdo a la asociación ecuatoriana de pilotos y profesionales de parapente existen 11 lugares totalmente calificados para brindar este servicio, así como 57 pilotos certificados (Asociación Ecuatoriana de Pilotos y Profesionales de Parapente, 2019). En la provincia de Chimborazo, ciudad de Riobamba, se encuentra el “Club de parapente Chimborazo” ubicado en el sector de Tunshi a 400 magl (Meters Above ground level / Metros Sobre el Nivel del Mar) este es un lugar turístico que ofrece la posibilidad de tener la experiencia de volar a cualquier interesado con la compañía de un profesional calificado.

Para realizar esta actividad en condiciones óptimas hay que tener en cuenta diversas condiciones tanto técnicas como climatológicas, es necesario que existan un viento de hasta 15 km/h que sirva para realizar un despegue adecuado, sin embargo, en el rango entre 15 y 30 km/h todavía es posible realizar un despegue para pilotos experimentados, en velocidades mayores a los 55km/h se convierte en un vuelo realmente peligroso y no es aconsejable (Saez, 2016). Además, hay que tener en cuenta la dirección del viento ya que para permitir elevar el parapente con más suavidad es necesario que el viento sea de cara a la carrera previa al despegue, en caso contrario deberá realizarse una carrera más forzada para poder despegar.

A nivel mundial existen una variedad de estaciones meteorológicas que ayudan a la medición, recolección y transmisión de datos de ciertas variables climáticas, pero enfocadas al deporte del parapente no existe ninguna.

Tomando en consideración las necesidades analizadas en la práctica del parapente es necesario conocer de la manera más precisa ciertas condiciones meteorológicas para tener la seguridad de realizar un vuelo con éxito, en el “Club parapente Chimborazo” la implementación de una estación meteorológica con comunicación a una aplicación Android ayudaría a los pilotos y usuarios interesados en realizar este deporte ya que podríamos obtener la información climatológica en el lugar del despegue.

JUSTIFICACIÓN TEÓRICA

Para la realización de este proyecto nos enfocamos en lo que es la supervisión de las variables meteorológicas en el Club parapente Chimborazo ubicado en el sector Tunshi debido a que en

este lugar el deporte del parapente es muy recurrente todas las semanas y también porque se realizan campeonatos a nivel mundial de este deporte, por lo que es necesario visualizar las condiciones que tiene el clima en ese momento para tener la seguridad de realizar un despegue adecuado y así gozar una mejor experiencia al practicar este deporte extremo.

El club parapente Chimborazo es reconocido a nivel nacional como un atractivo turístico ya que cuenta con el récord de asistencia en un festival de vuelo con la presencia de 125 pilotos de parapente de 16 nacionalidades diferentes en el año 2018 esto implica un gran impacto turístico para el país y en especial a la provincia (Andrade Alban, 2019). Existen registros en los cuales se pueden observar que es un deporte extremo en un auge de crecimiento, lo que conlleva que cada año un mayor número de aficionados empiecen a realizar este deporte los cuales no tienen la suficiente experiencia y necesitan de información climatológica para evitar posibles accidentes.

Con este dispositivo se supervisará las variables meteorológicas presentes en el club de parapente Chimborazo con la finalidad de permitir a los usuarios tener acceso a estos datos, los cuales son de gran importancia para la práctica segura de este deporte, desde cualquier dispositivo Android. La comunicación se realizará mediante protocolo 3G/GPRS ya que en el sector no existe cobertura de internet por cable.

JUSTIFICACIÓN APLICATIVA

Dado el crecimiento exponencial de los aficionados a la práctica del parapente y del uso de Smartphones, se presentó la idea de crear un dispositivo capaz de brindarnos las características climatológicas de la pista de despegue desde una app Android.

El dispositivo realizará la medición de temperatura, presencia de lluvia, velocidad y dirección del viento, presión atmosférica y humedad relativa del aire con el fin de procesar estos datos mediante una tarjeta controladora y enviarlos a una base de datos con comunicación 3G/GPRS a la vez que podrán ser visualizados en una pantalla presente en el lugar de implementación. Una vez almacenados los datos el usuario a través de una aplicación Android tendrá acceso a ellos.

FORMULACIÓN DEL PROBLEMA

¿Cómo implementar un dispositivo para monitorear variables meteorológicas en la pista de despegue del club parapente Chimborazo con una app Android?

SISTEMATIZACIÓN DEL PROBLEMA

¿Qué mecanismos existen para el monitoreo de variables meteorológicas?

¿Qué mecanismos para el monitoreo de variables meteorológicas pueden ser utilizadas en la comunicación con una app Android?

¿Cuáles son los requerimientos del dispositivo a implementar?

¿Qué hardware y software cumplen con los requerimientos del prototipo?

¿Se logrará supervisar las variables meteorológicas en tiempo real desde la app Android?

OBJETIVO GENERAL

- Implementar un dispositivo para monitorear variables meteorológicas en la pista de despegue del club parapente Chimborazo con una app Android

OBJETIVOS ESPECÍFICOS

- Investigar las variables y condiciones meteorológicas necesarias para el despegue de un parapente.
- Diseñar un dispositivo de medición de variables meteorológicas en tiempo real con comunicación a una aplicación móvil.
- Definir los componentes físicos que deberán ser usados en la implementación del dispositivo.
- Evaluar el funcionamiento del dispositivo y de la exactitud de las mediciones

El presente proyecto consta de una investigación aplicada, que nos permitió conocer el funcionamiento de las estaciones meteorológicas y su aporte para practicar el parapente, así como de los principales medios de comunicación que ayudaron a la transmisión de la información a una app Android y una investigación bibliográfica que nos ayudó en la búsqueda de información sobre el tema del proyecto, a partir de artículos científicos, libros y repositorios públicos disponibles en internet. Los resultados de la investigación se recopilan en tres capítulos, el primero consta del

estudió del parapente con las condiciones necesarias para practicar este deporte, así como las estaciones meteorológicas autónomas existentes, las variables a medir y la descripción de cada elemento utilizado en el dispositivo. El segundo capítulo se presenta los requerimientos tanto de hardware como de software del dispositivo, realizando una selección de cada uno de los elementos y el dimensionamiento del panel solar, batería y controlador para la etapa de alimentación del sistema. En tercer capítulo se realiza las pruebas de validación de los sensores y la repetitividad de cada uno, se evalúa el envío y consumo de datos que llegan al servidor de Thingspeak y se analiza el costo del dispositivo final implementado.

CAPÍTULO I

1. MARCO TEÓRICO

El presente proyecto describe la implementación de una estación meteorológica automática en el sector de Tunshi, este dispositivo permitirá la medición de variables meteorológicas para leerlas en el lugar del montaje a través de una interfaz visual y desde una aplicación Android con conexión a internet. Estos parámetros serán leídos y registrados con la tarjeta controladora Arduino para realizar la adquisición de cada dato de los sensores, además un módulo de comunicación GPRS/GSM para enviar las mediciones a una base de datos a la cual podremos acceder desde una aplicación Android.

La estación consta de un conjunto de sensores controlados por una tarjeta con conexión a una base de datos, las variables podrán ser revisadas desde la aplicación Android y en una pantalla en el lugar de implementación. La aplicación se conectará a la base de datos a través de conexión a internet.

1.1 El parapente

Es un dispositivo de vuelo constituido por una estructura a modo de vela fabricado de una tela sintética especial conocida como Rip Stop. A esta se le une la silla mediante delgadas cuerdas sintéticas de kevlar o dyneema llamadas suspentes o cordines, el final consta de bandas unidas a dos anclajes que finalmente se unen al arnés utilizando mosquetones de seguridad. La vela toma la forma gracias a la presión que se ejerce dentro de los cajones y que ingresa por las bocas situados en la parte delantera o de ataque (Sáez, 2016).



Figura 1-1. Partes del parapente

Fuente: (Saez, 2016)

1.2 Condiciones meteorológicas óptimas para practicar este deporte

El primer aspecto a tomar en cuenta es la dirección del viento, para aterrizar y para despegar siempre a barlovento. Se entiende el término barlovento cuando se mira hacia el cuadrante del cual sopla el viento, recibiendo en la cara y pecho. Uno de los aspectos más importantes antes de tomar la decisión de realizar un despegue es analizar las condiciones exactas de dirección y velocidad del viento de acuerdo con las características del terreno. En ocasiones el viento se introduce a modo de impulso lo que puede producir que se sienta una falsa apreciación de la dirección e intensidad del viento en el lugar, por lo que se aumenta el peligro a la hora del despegue (Soriano, Valarezo y Hernández, 2016).

Para poder realizar un despegue en condiciones ideales el viento debe estar entre los 15 y 30 Km/h a barlovento. Con esta velocidad la vela del parapente se inflará sin apenas carrera por lo que resultará un despegue sencillo, en caso de ser una velocidad menor no existe la posibilidad de tener problemas técnicos, sin embargo, será necesaria una carrera de mayor distancia y con más trabajo hasta lograr que se infle la vela (Saez, 2016).

En resumen, podemos definir las siguientes condiciones climáticas como óptimas para realizar el despegue con un parapente.

- La velocidad del viento debe estar entre los 15 y 30 Km/h, para pilotos experimentados puede ser hasta los 40 Km/h. Si existe una velocidad mayor, no se recomienda el despegue ya que existe una gran probabilidad de que se descontrola la vela.
- La dirección del viento debe ser siempre a barlovento para que el despegue sea suave y a favor, nunca se debe realizar a contravento ya que además de dificultar la carrera también puede producir lo que se conoce como viento de cola que produce movimientos impredecibles de la vela.

- En caso de existir inestabilidad en la presión atmosférica también hay q tener precaución, ya que esto genera turbulencias.
- Por último, hay que tener en cuenta que la vela es sensible a lluvia o nieve por lo que se recomienda evitar estas situaciones en medida de lo posible, esto se debe a que la tela de la que está construida la vela puede absorber la humedad y variar su peso lo que podría provocar problemas en las maniobras.

1.3 Estaciones meteorológicas automáticas.

Es un dispositivo para obtener las características meteorológicas de un lugar en específico mediante el uso de sensores electrónicos. Estas lecturas después son procesadas con un microcontrolador y transmitida con sistemas de comunicación ya sean mediante cables o inalámbricos de manera autónoma. Una estación detecta, mide y transmite durante 24 horas al día y debe poseer independencia energética, ya sea con baterías de larga duración o paneles solares (Benítez y Armoa, 2016).

Las principales ventajas de la utilización de una estación meteorológica automática son:

- Mediciones de acuerdo con las necesidades del usuario.
- Posibilidad de realizar mediciones en lugares de difícil acceso.
- Confiabilidad en la medición de datos.
- Rapidez en la transmisión de las medidas.

1.4 Variables a medir en la estación meteorológica

En este proyecto es indispensable tener en claro cuáles son las variables que se necesita monitorear, para este deporte lo más importante es considerar las características del viento, presión atmosférica, presencia de lluvia, temperatura y humedad relativa en el lugar de despegue.

1.4.1 Temperatura del aire

Es una medida que resulta del intercambio de calor entre la atmosfera y la tierra, se mide en grados Celsius (°C) (Ayllón, 2013). Esta característica es tomada en cuenta para que los usuarios del

parapente puedan evitar temperaturas extremas ya sea de frio o de calor en el lugar del despegue el rango en la zona varía entre los 2 °C y 25 °C (Freemeteo.Ec, 2019).

1.4.2 Humedad relativa

Se entiende como la capacidad que tiene el aire de absorber la humedad, es decir la cantidad de vapor de agua que tiene el aire y se mide en %. Por lo tanto, a mayor humedad relativa mayor partículas de vapor de agua se encuentran en el aire (Ayllón, 2013). Es necesario considerar la humedad relativa ya que si es demasiado alta puede dañar los equipos de vuelo, en la pista de despegue existen registros desde el 60% de humedad hasta el 95% (Freemeteo.Ec, 2019).

1.4.3 Presión atmosférica

Es la fuerza que ejerce la atmosfera en una unidad de superficie y se expresa en bares (ba). Por lo tanto, se puede entender como el peso de una columna de aire en una unidad de superficie con una altura definida superiormente por el límite de la atmosfera (Ayllón, 2013). Esta variable es necesaria ya que la variación de esta presión significa una inestabilidad climatológica tanto en el viento como en las precipitaciones en el lugar de la implementación existen registros entre 700 hPa y los 710 hPa (Freemeteo.Ec, 2019).

1.4.4 Velocidad y dirección del viento

Primero debemos entender que el viento es el movimiento del aire en la atmosfera, este viento se puede caracterizar con su dirección, expresada en coordenadas geográficas, y su velocidad, medida en metros sobre segundo (m/s) (Ayllón, 2013). Sin duda alguna es una de las características más importantes en la práctica del parapente ya que depende de estas hacia donde realizar la carrera para el despegue y con qué intensidad, en el sector de Tunshi se registran valores hasta los 60 km/h (Freemeteo.Ec, 2019).

1.4.5 *Presencia de lluvia*

En caso de existir lluvia en el lugar del despegue, será poco recomendado el vuelo ya que esta puede producir daño en los equipos, así como dificultar las maniobras de los pilotos. Por lo tanto, hay q tener presente la presencia o ausencia de lluvia en la pista de despegue.

1.5 **Descripción de los tipos sensores utilizados en la estación meteorológica**

Para diseñar una estación meteorológica electrónica es necesario conocer los sensores que producen una señal la cual pueda ser procesada de manera electrónica, se entiende como sensor un dispositivo el cual recibe una señal o estímulo físico y responde con una variación eléctrica, resistiva, capacitiva, etc. Un sensor está formado por un transmisor, encargado de acondicionar la señal, y un transductor que es capaz de convertir un tipo de energía en otra normalizada ya sea de corriente (4-20 mA) o de voltaje (0-5 V) (Pallás Areny, 2007).

1.5.1 *Sensor de humedad relativa*

Para poder medir la humedad relativa de un lugar, existe una clasificación de sensores extensa, en la tabla 1-1 se van a comparar los sensores electrónicos.

Tabla 1-1: Comparación entre tipos de sensores electrónicos de humedad relativa

Características	Psicometría por bulbo húmedo/bulbo seco	Sensor de bloque de polímero resistivo	Sensores capacitivos
Composición	un par de termómetros eléctricos acoplados, uno de los cuales opera en estado húmedo	Sustrato cerámico aislante sobre el cual se deposita una grilla de electrodos.	Platos paralelos con electrodos porosos o con filamentos entrelazados en el sustrato.

Precisión	Alta en valores cercanos a 100% Baja cuando trabaja por debajo de 20%	Alta en valores cercanos a 100% Baja cuando trabaja por debajo de 20%	Alta en valores bajos de humedad. Presenta variaciones en valores mayores a 85%.
Desventajas	No trabaja a temperaturas inferiores a 0 °C. No puede trabajar en ambientes cerrados y pequeños.	Tiempo de respuesta variante. Trabajan con corriente alterna.	Soporta temperaturas muy altas. Sensibilidad muy alta. Tiempo de respuesta alto.

Fuente: (Kouro, 2001)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Analizando la tabla 1-1, se decidió optar por la familia de sensores capacitivos ya que cumple con los requerimientos del dispositivo. Es capaz de trabajar en un rango de 0 a 100% con un tiempo de respuesta alto y con una variación significativa de acuerdo con la variación de humedad.

1.5.2 Sensor de Temperatura

Para realizar una medición de la temperatura en el lugar de la implementación se realizó el estudio de dos tipos de sensores cuyas características se observan en la tabla 2-1.

Tabla 2-1: Características de sensores de temperatura

Características	Termocuplas	Termorresistencias
Rango en °C	-200 a 2800	-250 a 850
Velocidad de respuesta	Rápida	Media

Estabilidad	Largo periodo	Medio periodo
Salida	Lineal	No lineal
Precisión	Alta	Media

Fuente: (JMIndustrial, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Una vez analizada la tabla 2-1, se produjo la selección de la familia de sensores térmicos termo resistivos por su mayor precisión, linealidad que permite procesar mejor las mediciones y por su tiempo de respuesta más rápido.

1.5.3 Sensor de presión atmosférica

Existen diferentes principios bajo los cuales podemos realizar la medición de presión atmosférica, en la tabla 3-1 podemos apreciar los estudiados en este apartado con sus principales características.

Tabla 3-1: Comparación entre los tipos de sensores de presión atmosférica

Características	Piezo-Resistivos	Capacitivos	Piezoeléctricos
Construcción	Realiza cambio en la resistencia de elementos semiconductores cuando son sometidos tensiones mecánicas.	Se basa en la variación de capacidad de un capacitor con el desplazamiento de una de sus capas debido a la presión.	Son materiales cristalinos que producen una señal eléctrica debido a una deformación física causada por la presión.
Rango de medición (bar)	0.3 a 1000	0.05 a 40	0.1 a 600
Precisión	1	1	1
Sensibilidad a vibraciones	Baja	Media	Baja

Ventajas	Bajo costo Posibilidad de integrarlo en integrado	Solo trabaja en presiones bajas.	Mide presiones dinámicas
-----------------	--	----------------------------------	--------------------------

Fuente: (Torres, 2018, p. 2)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Analizando la tabla 3-1, se decidió optar por los sensores piezo-resistivos por su disponibilidad en el mercado y fácil adaptación a un integrado, además cumple con el rango de medición indicado en los requerimientos y tiene un costo bastante bajo.

1.5.4 Sensor detector de lluvia

Es un tipo de sensor capaz de detectar la presencia de lluvia basando en el principio de variación de la conductividad, poseen una construcción sencilla de varios hilos conductores separados entre sí que al entrar en contacto con el agua hacen contacto y empiezan la conducción como se puede observar en la figura 2-1 (Llamas, 2016).



Figura 2-1. Placa para detección de lluvia.

Fuente: (Llamas, 2016)

1.5.5 Sensor de velocidad y dirección del viento

Es necesario realizar la medición tanto de la velocidad como de la dirección del viento, para ello se utilizan tanto anemómetro como veleta con la capacidad de convertir la energía cinética del viento en energía eléctrica que puede ser procesada por una tarjeta de desarrollo. En la tabla 4-1 se observa las diferencias entre el tipo de señal entregada por el sensor.

Tabla 4-1: Comparación entre los tipos de señal entregada por los sensores de velocidad y dirección del viento

	Salida de corriente	Salida de Voltaje	Salida Frecuencia
Rango de señal	4-20 mA	0-5 V	0-100 Hz
Resolución	± 1 km/h	± 1 km/h	± 1.8 km/h

Fuente: (PCE, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Analizando la tabla 4-1, se optó por seleccionar un sensor con salida de voltaje ya que nos ofrece más facilidad a la hora de realizar el procesamiento con una tarjeta Arduino además de ofrecernos una precisión acorde a los requerimientos del dispositivo.

1.6 Tarjetas controladoras

Para poder leer y procesar las diferentes señales producidas por cada sensor es necesaria la utilización de una tarjeta controladora. Para seleccionar la más adecuada se analizó en la tabla 5-1 tres diferentes familias disponibles en el mercado con sus principales características.

Tabla 5-1: Comparación entre tarjetas controladoras

Características	Arduino	RaspBerry pi	Tessel
Alimentación	7-12 V	5V	5V
Tipo de lectura	Analógica/ digital	Digital	Digital
Corriente I/O	40 mA	2.5 A	17 mA
Reloj	16 MHz	1.2 GHz	48MHz

Fuente: (Arduino, 2019)(Raspberry, 2019)(Tessel, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Se decidió optar por la familia de tarjetas controladoras Arduino debido a que posee tanto terminales de lectura analógicos como digitales, lo que es de gran ayuda al momento de realizar el procesamiento de las señales a tratar.

1.7 Transmisión de datos

Una vez realizado el procesamiento de datos, esta información debe ser enviada y almacenada en un base de datos la cual se encuentra en la nube, para ello se procede a realizar un análisis de las posibles tecnologías a utilizar, en la tabla 6-1 se encuentran las características de las tecnologías inalámbricas analizadas.

Tabla 6-1: Características de las tecnologías inalámbricas

	Wifi	GPRS	Wi max
Radio de celda	0.01-0.1 Km	30 Km	1-15Km
Estándar	802.11	GPRS	802.16
Banda de transmisión	2.4 GHz 5 GHz	800 MHz 1800MHz 1900MHz	2.3 GHz 3.5 GHz
Tasa de transmisión	36 Mbps	22Kbps	75Mbps
Disponibilidad en lugar de aplicación	NO	SI	NO

Fuente: (Adrián de la Cámara, 2017)(Viloria Núñez, Peña y Garzón, 2009, p. 200-217)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

La forma de transmitir los datos obtenidos mediante la tarjeta controladora es muy importante y hay que tener en cuenta la mejor manera de acuerdo con las condiciones del entorno en donde se va a implementar el proyecto, por ello se decidió trabajar con tecnología GPRS debido a que el lugar de implementación es bastante aislado, de difícil acceso y no existe cobertura de las otras tecnologías, además cumple con la tasa de transmisión necesaria para el envío de los datos obtenidos.

1.7.1 Red GPRS (Global Packet Radio System)

También conocida como 2.5G, nos permite realizar una comunicación satelital, es decir sin ninguna conexión física entre dispositivos móviles. Se puede transmitir información a una velocidad de 144 kbps ya que incorpora la transmisión por paquetes. Una de las grandes ventajas de estos sistemas GPRS es su compatibilidad con redes de internet (Adrián de la Cámara, 2017).

1.8.1 Energía Solar

El sol es una fuente de energía constante y que con el avance de la tecnología se ha empezado aprovechar para obtener energía eléctrica. Estos procesos se aplican en lugares de difícil acceso donde no llega la red eléctrica convencional, elementos de exterior como iluminación, parquímetros o cercados eléctricos para ganado (Perpiñan Lamigueiro, 2018).

Las principales ventajas del aprovechamiento de esta energía:

- No contamina
- No necesita mantenimiento
- Instalación de larga duración
- No emite ruido
- Aguanta climas severos

1.8.2 Componentes de un sistema fotovoltaico

Los principales elementos de este tipo de sistemas se pueden reducir a paneles solares, baterías, reguladores y elementos de protección.

1.8.2.1 Panel fotovoltaico

Es un elemento capaz de generar energía eléctrica, a partir de la radiación solar, consiste en un conjunto de celdas unidas en serie y en paralelo, el número de celdas depende de la potencia que se quiere alcanzar (Tobajas Vásquez, 2015). Posee dos borneras, una positiva y una negativa, para poder conectar el panel al resto de la instalación, podemos observarlo en la figura 4-1.

Las partes principales que lo conforman son (Méndez Muñiz, 2012):

- Marco de aluminio: Sujeta y da estabilidad a la estructura.
- Cubierta de vidrio resistente y liviano que deje el paso total de la radiación.
- Cierre hermético de esponja
- Células fotovoltaicas
- Cara posterior con superficie plástica o metálica.

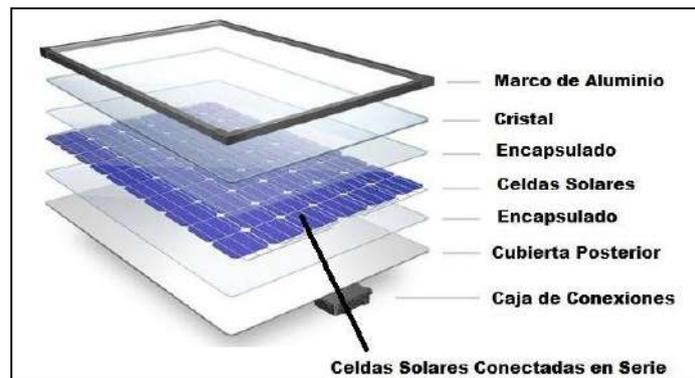


Figura 4-1. Panel Fotovoltaico

Realizado por: Coronel jhonatan, Espinosa Marcos 2019.

1.8.2.2 Baterías

Los paneles fotovoltaicos necesitan de acumuladores o baterías para mantener energía cuando no estén en operación. Su funcionamiento es similar a una batería de automóvil que se carga cuando el generador está activo. Existen dos tipos de baterías dependiendo del trabajo al que son destinadas, carga superficial y carga profunda (Perpiñan Lamigueiro, 2018).

- Carga superficial: Se trata de las baterías que realizan con normalidad descargas del 40% diaria y en días excepcionales hasta un 60% de descarga con lo que se garantiza la durabilidad de los materiales.
- Carga profunda: Este tipo de baterías trabajan normalmente con descargas del 60% y en excepciones hasta el 80%.

1.8.2.2.1 Carga y descarga de las baterías

Para que se puedan cargar las baterías debe existir una tensión de carga mayor a la del propio acumulador, además la transferencia de energía proporcionada por los paneles debe ser mayor que el consumo del dispositivo. La corriente de carga (I_c) deber no debe ser superior al 10% de la capacidad útil de la batería. Lo normal esta entre un 6% y 6.5%. Dependiendo de las necesidades de cada aplicación se podrán acumular baterías ya sea en paralelo para aumentar la capacidad de corriente o en serie para una mayor tensión (Perpiñan Lamigueiro, 2018).

1.8.2.3 Regulador

Es el elemento encargado de controlar el flujo de energía entre el panel fotovoltaico y la batería, se puede ver en la figura 5-1. Es el encargado de garantizar las siguientes funciones (Martín Jimenez, 2010):

- Proteger a las baterías tanto de la sobrecarga como de sobre descarga.
- Alarmar en caso de nivel de batería muy bajo.
- Debe desconectarse de la carga cuando alcance un nivel de descarga máximo.
- Desconexión hacia el panel fotovoltaico cuando la batería este en su capacidad máxima.



Figura 5-1. Modulo regulador.

Realizado por: Coronel jhonatan, Espinosa Marcos 2019.

1.8.3 Elementos de Protección

Como toda instalación eléctrica los elementos de protección son muy importantes para evitar daños en los equipos e incluso a los usuarios. Podemos clasificar como protecciones contra contactos directos y protecciones contra contactos indirectos.

1.8.3.1 Protecciones contra contactos directos:

El exceso de temperatura causa daños irreversibles en una instalación eléctrica, esto se produce tanto por cortocircuitos como por sobrecargas, siendo más dañino el cortocircuito ya que en un tiempo muy corto puede dañar por completo los equipos (Martín Jimenez, 2010). Para evitar estos

daños utilizamos los fusibles, se colocan en portafusiles como se puede ver en la figura 6-1 y los interruptores automáticos, el principio básico de estos elementos es que al detectar una corriente elevada y debido al efecto Joule aumenta la temperatura por lo que abren el circuito.



Figura 6-1. Portafusiles y fusibles

Realizado por: Coronel Jhonatan, Espinosa Marcos 2019.

1.8.3.2 Protecciones contra contactos indirectos:

En una instalación eléctrica pueden existir fugas eléctricas por fallas o por agentes externos como la humedad, mediante las cuales puede llegar la energía a eléctrica a los usuarios. Para evitar este tipo de fallas o detectarlas a tiempo se suele usar un interruptor diferencial que mide el paso de corriente eléctrica en un conductor y en la vuelta de el mismo, en caso de no ser iguales abre el circuito inmediatamente (Tobajas Vásquez, 2015).

1.9 Almacenamiento de datos

La selección de la base de datos es de suma importancia ya que nos debe permitir acceder a ella tanto desde la aplicación móvil como desde nuestra tarjeta de desarrollo, para ellos se optó por utilizar plataformas dedicadas a trabajar con IoT, en la tabla 6-1 se muestran las principales características de las diferentes plataformas IoT estudiadas.

Tabla 7-1: Características de las plataformas IoT

	ThingSpeak	Samsung Artik Cloud	aRest Framework
Numero de datos gratuitos	250 000 mensuales (cada 15 segundos)	100 000 mensuales (cada 10 minutos)	1000 mensuales
Dispositivos conectados	4	5	5
Tiempo histórico de acceso a datos	Ilimitado	3 meses	-

Fuente: (Loureiro Garrido, 2015, p. 5-8)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Tomando en consideración el número de datos enviados por nuestro dispositivo se decidió utilizar la plataforma ThingSpeak ya que nos permite almacenar mayor número de datos durante un tiempo ilimitado, además de que existen librerías propias para poder trabajar con tarjetas de desarrollo.

1.10 Aplicaciones móviles

Actualmente la gran mayoría de personas alrededor del mundo cuenta con un teléfono inteligente o Smartphone, es por ello que las aplicaciones móviles se han convertido en uno de los medios más importantes para la interacción con un usuario, especialmente el sistema operativo Android (Robledo Fernández, 2014).

1.10.1 Android

Android es un sistema operativo con bases en el Kernel de Linux, este conjunto de software consta del sistema operativo, middleware y aplicaciones. Una de las grandes ventajas de este SO es que esta desarrollado con licencia libre y código abierto, es decir cualquiera que desee puede tener acceso al código fuente. Otra ventaja es la existencia de un enfoque unificado a la hora de desarrollar aplicaciones móviles, por lo que es necesario desarrollar únicamente una aplicación

para Android y esta debe ser capaz de ejecutarse en cualquier dispositivo con SO Android (Santillán Rivadeneira y Maldonado Jaramillo, 2017).

1.10.1.1 Arquitectura de Android

Básicamente Android es una plataforma para dispositivos móviles que consta de un SO, librerías desarrolladas en lenguaje C y C++, un framework usado para el desarrollo de aplicaciones y aplicaciones básicas iniciales, esta estructura puede observarse en la figura 7-1.

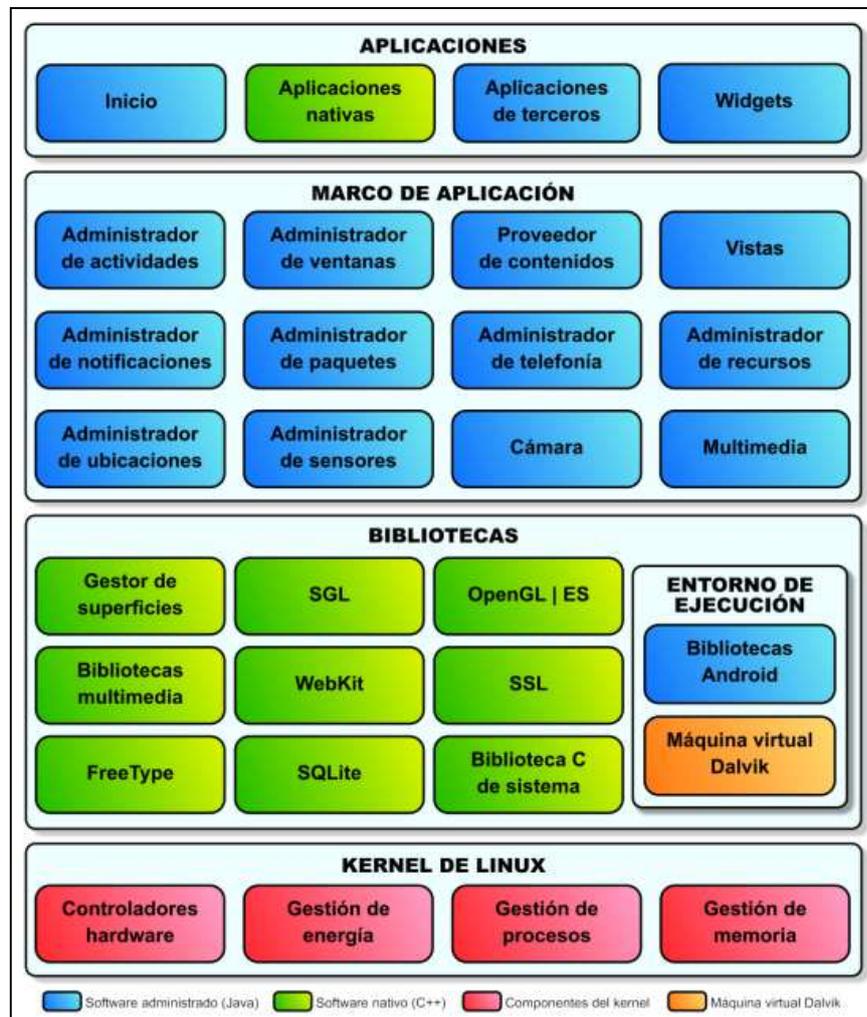


Figura 7-1. Resumen arquitectura Android.

Fuente: (Villa Yanez, 2016)

Las principales características son (Nolasco Valenzuela, 2013):

- Componentes básicos de cada aplicación son modulares, es decir se pueden sustituir fácilmente.

- Aplicaciones reutilizables y que se puedan añadir en otros dispositivos y aplicaciones.
- Se pueden utilizar bases de datos.
- Controla elementos hardware como: bluetooth, wi-fi, cámaras, etc.
- Desarrollo gratuito en base a un SDK muy cuidado.

1.10.2 Entorno de desarrollo Android Studio

Para empezar el desarrollo de una aplicación Android es necesario poseer herramientas y entorno para poder programar (Nolasco Valenzuela, 2013).

- Se necesita un JDK (Android Software Development Kit)
- Un IDE, Android Studio es el entorno de desarrollo oficial.
- Además de los componentes principales, los cuales deben permanecer actualizados, como el SDK, la versión de Android, Google Play Services entre otros.

Una vez se posee las herramientas necesarias se lleva a cabo la programación en lenguaje Java o .NET, las características de Android Studio en su versión actual 3.2.1 más destacable es la facilidad de realizar el diseño de la interfaz de usuario, un tipo de programación gráfica, en la que basta con arrastrar hacia el espacio de trabajo cada uno de los elementos que se desean utilizar (Santillán Rivadeneira y Maldonado Jaramillo, 2017). Además de poseer una interfaz bastante amigable con el desarrollador y muy intuitiva a la hora de programar.

CAPÍTULO II

2. MARCO METODOLÓGICO

En el siguiente capítulo se muestra el diseño del hardware y software del dispositivo, detallando cada una de las etapas las cuales lo conforman, los componentes utilizados con sus características y los esquemas de conexión eléctricos y electrónicos de los circuitos implementados.

2.1 Requerimientos del dispositivo

- Tener acceso a la información obtenida desde cualquier lugar con una App Android con conexión a internet.
- Ayudar a los usuarios de parapente en su despegue y la planificación de vuelo.
- Seleccionar sensores acordes a la tarjeta controladora utilizada.
- Proveer la información obtenida en el lugar de la implementación.
- Diseñar un dispositivo autosustentable energéticamente con la utilización de paneles solares.
- La estructura debe soportar vientos de hasta 60 km/h
- Resistencia de la estructura a la corrosión.
- Protección con una malla en un área de 2x2 m con base de cemento.
- Gabinete de conexiones para montar los equipos.

2.2 Diseño de la arquitectura del dispositivo

La concepción general del dispositivo se lo presenta en la figura 1-2, mostrando la detección y transmisión de las variables meteorológicas medidas con cada una de las etapas las cuales la conforman.

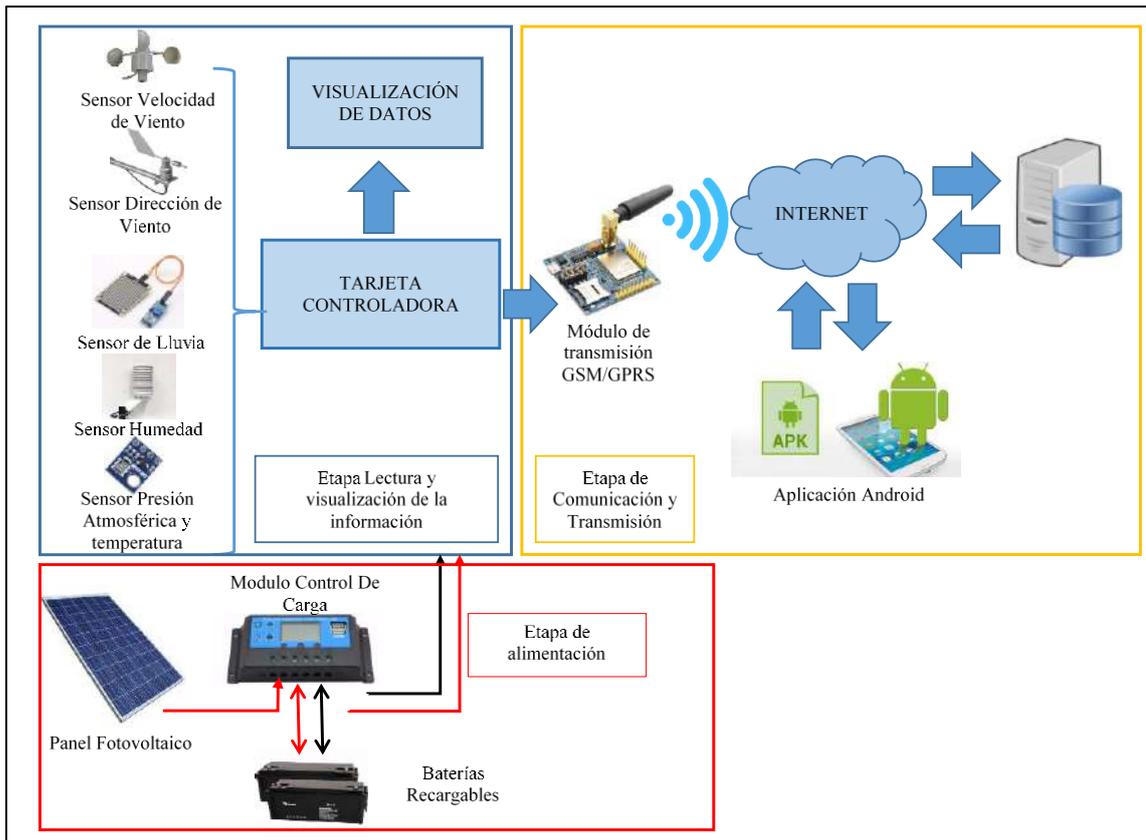


Figura 1-2. Esquema del dispositivo
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Con este dispositivo se permitirá a los usuarios el acceso a estos datos, los cuales son de gran importancia para la práctica óptima de este deporte, desde cualquier dispositivo Android. La comunicación se realizará mediante protocolo 3G/GPRS debido a la falta de cobertura de internet por cable en el sector.

El comportamiento del dispositivo se puede observar en la figura 1-2, el dispositivo realizará la medición de temperatura, velocidad y dirección del viento, presión atmosférica y humedad relativa del aire con el fin de procesar estos datos mediante una tarjeta microcontroladora y enviarlos a una base de datos con comunicación 3G/GPRS a la vez podrán ser visualizados en una pantalla presente en el lugar de implementación. Una vez almacenados los datos el usuario a través de una aplicación Android tendrá acceso a los datos guardados tanto en tiempo real como en un historial con lo cual se podrá decidir si existen las condiciones óptimas para volar. También el dispositivo se auto sustentará energéticamente con el uso de paneles solares.

2.3 Diseño de las etapas del dispositivo

A continuación, se muestran las etapas que conforman nuestro dispositivo a implementar.

2.3.1 Etapa de lectura y visualización de las variables meteorológicas

En la figura 2-2 se presenta el diagrama de bloques para la etapa de lectura y procesamiento de datos. La lectura de cada una de las variables se realizará mediante sensores como anemómetro, veleta, barómetro, sensor de humedad, temperatura y lluvia, las señales obtenidas serán procesadas con la tarjeta controladora Arduino Mega 2560 para su correcta interpretación, a su vez la información se enviará a una pantalla LCD ubicada en el lugar de la instalación.

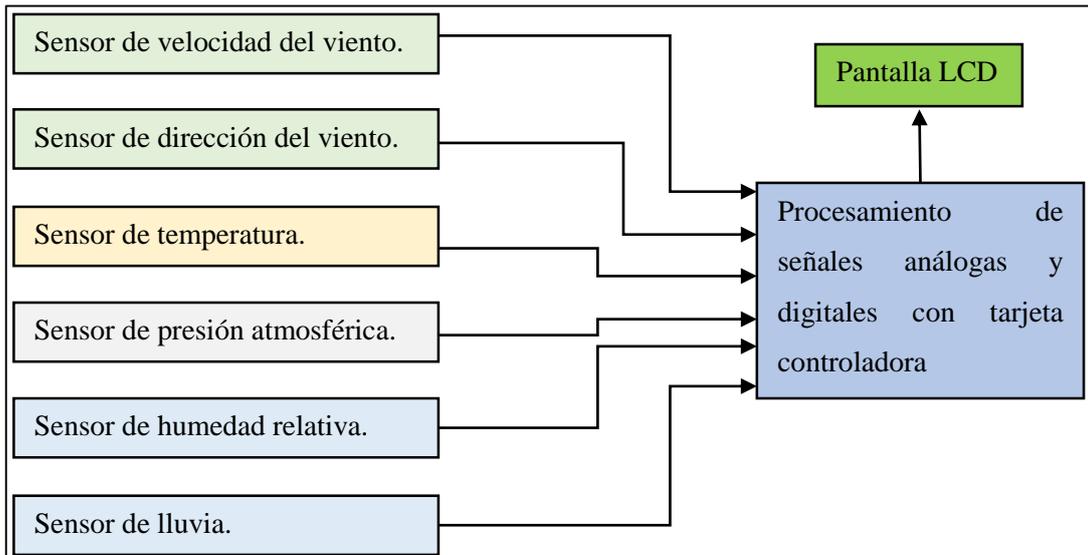


Figura 2-2. Etapa de lectura y visualización de las variables meteorológicas
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.3.2 Etapa de comunicación y transmisión de datos.

En la figura 3-2 se representa el diagrama de bloques correspondiente a la comunicación de nuestro dispositivo, para ello se dispone de un módulo GPRS SIM 900 el cual será el encargo de realizar la transmisión de los datos leídos hacia una base de datos. A través de una aplicación móvil se tendrá acceso a estas variables tanto en tiempo real como a un historial almacenado en la base de datos.

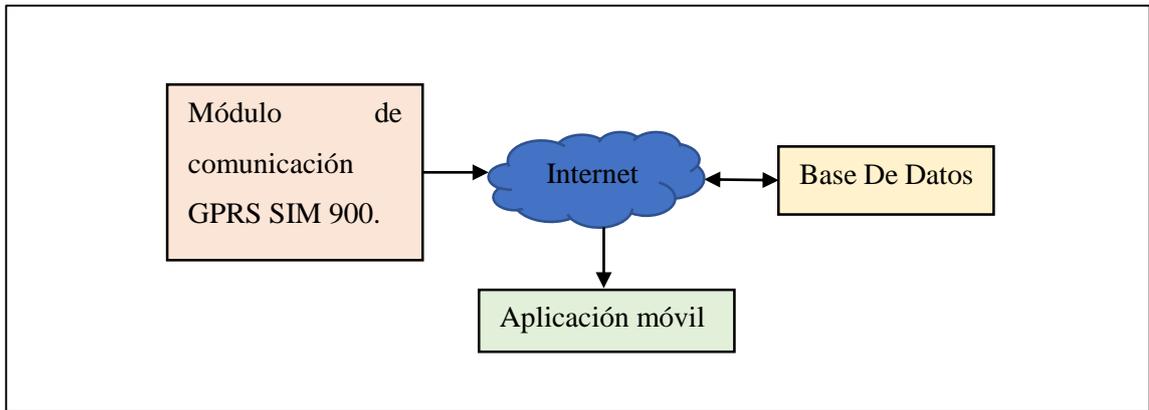


Figura 3-2. Etapa de comunicación y transmisión de datos.

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.3.3 Etapa de alimentación y energía.

En la figura 4-2 se observa la etapa encargada de alimentar todo nuestro circuito, con la característica de equipar un panel solar el cual será el principal alimentador de la batería que a su vez alimentará el resto de nuestro circuito. A través del módulo controlador se podrá recargar la batería en las horas en las cuales las celdas fotovoltaicas generen mayor energía.

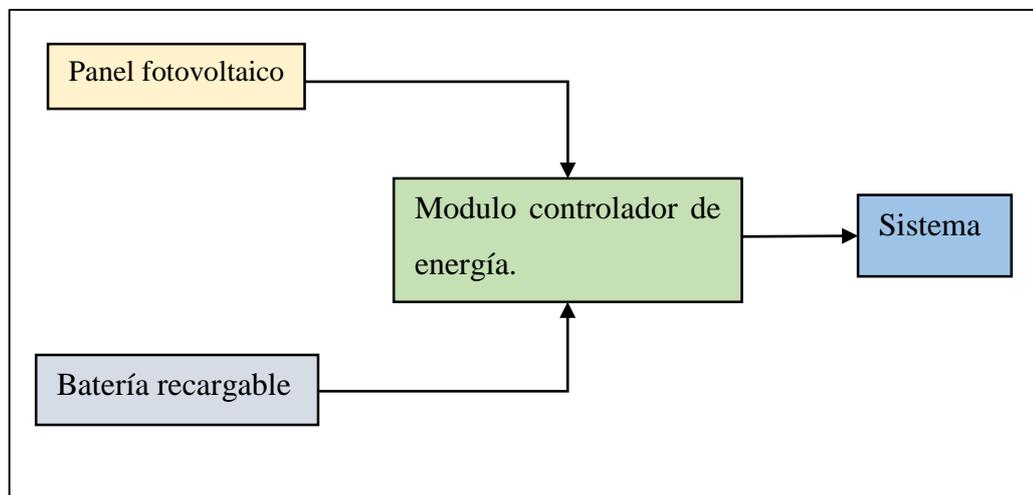


Figura 4-2. Etapa de alimentación y energía

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.4 Selección del hardware del dispositivo y sensores

En esta parte se muestra la selección tanto del hardware como de los sensores utilizados por el dispositivo.

2.4.1 Arduino

Como en el capítulo anterior se optó por el uso de Arduino vamos a realizar una comparación entre tres tarjetas controladoras de la misma familia, en donde los escogidos fueron el Arduino UNO, MEGA2560 y el DUE. A continuación, se presentan las características principales de los tres componentes para su respectiva selección.

Tabla 1-2: Características técnicas de las tarjetas controladoras

Características	Arduino UNO	Arduino MEGA 2560	Arduino DUE
Voltaje Entrada	7-12 V	7-12 V	7-12 V
Voltaje de Operación	5 V	5 V	3.3 V
Corriente de E/S	40 mA	40 mA	130 mA
Procesador	ATmega328P	Atmega2560	AT91SAM3X8E
Velocidad	16 MHz	16 MHz	84 MHz
Memoria Flash	32 Kb	256 Kb	512 Kb
I/O Analógicas	6	16	12
I/O Digitales	14	54	54

Fuente: (Arduino, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Con la tabla 1-2 se llegó a la conclusión que la mejor tarjeta controladora para el dispositivo es el Arduino MEGA 2560 mostrado en la figura 5-2, ya que consume menos energía para el desarrollo de la etapa de alimentación y con una memoria de procesamiento suficiente para lo requerido, con entradas y salidas digitales necesarias para la conexión de los demás equipos y, además de su fácil conectividad con los mismos.

Este dispositivo posee un microcontrolador ATmega2560, al cual se le adaptan entradas digitales y analógicas y salidas digitales y PWM. Esta tarjeta programable en un entorno de desarrollo donde se implementa el lenguaje processing/wiring, además tiene la posibilidad de realizar

comunicación serial mediante los UARTs (Lajara Vizcaino, 2014). En el anexo A se muestra el datasheet del mismo.



Figura 5-2. Arduino MEGA 2560
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.4.1.1 Sensor de humedad relativa

Para la selección del sensor de humedad se escogieron los sensores DHT11 y DHT22 que son de tipo capacitivo para la detección de humedad relativa. Ambos sensores usan un módulo de procesamiento para obtener una salida digital utilizando una librería propia para la conexión con Arduino haciéndolo más confiable y de una fácil programación. A continuación, se muestran las características principales de cada sensor para su selección.

Tabla 2-2: Características técnicas de los sensores de humedad

Características	DHT22	DHT11
Voltaje de Operación	3.3-6 V	3-5 V
Corriente de Operación	2.1 mA	2.5 mA
Rango De Humedad Relativa	0% a 100%	20% a 95% RH
Precisión	2% RH	4% RH
Resolución	0.1%	1%
Tiempo De Respuesta	2s	1s
Precio	\$12	\$6

Fuente: (Omniblug, 2018); (Electronilab, 2019c)
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 2-2 se observan las características de cada sensor, dando como resultado la selección del sensor DHT11 mostrado en la figura 6-2 debido a su bajo precio, aunque el DHT22 tiene

mejores rangos de medición, precisión y resolución, el DHT11 tiene las características necesarias para los requerimientos del lugar de la instalación. En el anexo B se indica el datasheet del mismo.

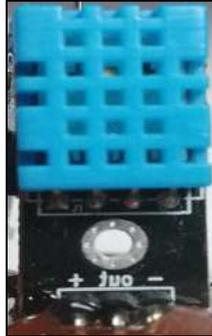


Figura 6-2. Sensor DHT11

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.4.1.2 Sensor de presión atmosférica y temperatura

En esta selección se optó por los sensores BMP180 y BME280 basados en la tecnología piezo-resistiva para darle mayor robustez, precisión alta y linealidad, haciéndolos muy estables a largo plazo. Ambos sensores son capaces de medir presión y temperatura, y utilizan interfaz I2C que ayudan a tener una fácil conexión con Arduino. A continuación, se muestran las características principales de cada sensor para su selección.

Tabla 3-2: Características técnicas de los sensores de presión atmosférica y temperatura

Características	BMP180	BME280
Voltaje de Operación	1.8-3.6 V	1.8-3.3 V
Corriente de Operación	3 uA	2.8 uA
Rango De Presión Atmosférica	300 hPa-1100 hPa	300 hPa – 1100 hPa
Resolución de Presión Atmosférica	0.01 hPa	0.16 Pa
Precisión de Presión Atmosférica	0,12 hPa	1 hPa
Rango De Temperatura	0° C a 65° C	-40 ° C a 85°C
Resolución de Temperatura	0.1 °C	0.01°C
Precisión de Temperatura	1° C	1°C
Precio	\$5	\$12

Fuente: (Electronilab, 2019b)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Con los datos de la tabla 3-2 se escogió el sensor BMP180 mostrado en la figura 7-2, debido a su bajo precio y también porque cumple con los requerimientos del lugar de la instalación del dispositivo haciéndolo más accesible en el mercado para obtenerlo con facilidad. En el anexo C se indica el datasheet del mismo.



Figura 7-2. Sensor BMP180
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.4.2 Sensor de lluvia

Para la selección del sensor de lluvia se comparó al sensor M-152 con el módulo sensor YL-38, ambos capaces de detectar la presencia de lluvia. A continuación, se muestran las características principales de cada sensor para su selección.

Tabla 4-2: Características técnicas de los sensores de lluvia

Características	YL-38	M-152
Voltaje de Operación	3.3-5 V	12 V
Corriente de Operación	15 mA	160 mA
Tipo De Salida	Digital y Analógica	Relé

Fuente:(Coba Robalino y Hernández Angulo, 2018)
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 4-2 el sensor escogido fue el YL-38 mostrado en la figura 8-2, ya que posee una salida digital y analógica permitiendo una fácil conexión con Arduino y también por su bajo consumo de energía. En el anexo D se indica el datasheet del mismo.



Figura 8-2. Módulo sensor YL-38
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.4.3 Sensores de velocidad y dirección del viento

Para la selección de los sensores de velocidad y dirección del viento se escogieron a los sensores SY-FS2 y FST200-201 para la velocidad y los sensores SY-FX2 y FST200-202 para la dirección ambos de fácil instalación, siendo capaces de soportar temperaturas y humedades altas. A continuación, se muestran las características principales de cada sensor para su selección.

Tabla 5-2: Características técnicas de los sensores de velocidad y dirección del viento

Características	SY-FS2	FST200-201	SY-FX2	FST200-202
Voltaje de Operación	12-24 V	12-36 V	12-24 V	12-30 V
Salida Análoga	0-5 V o 4-20 mA			
Rango de medición	0-30 m/s	0,5-50 m/s	0-360°	0-360°
Resolución	0,1 m/s	0,5 m/s	0,1 m/s	22,5°
Precisión	±3 %	±3 %	±3 %	±3 %
Precio	\$70	\$85	\$80	\$86

Fuente:(Trading, 2019); (Trading, 2019, p. 1-2); (Sensor, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Con los datos de la tabla 5-2 los sensores seleccionados fueron el SY-FS2 para la velocidad mostrado en la figura 9-2 y el SY-FX2 para la dirección mostrado en la figura 10-2, ya que son de bajo costo y cumplen con los requerimientos del lugar de la instalación del dispositivo. En el anexo E y anexo F se indican los datasheet de los mismos.



Figura 9-2. Sensor SY-FS2
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019



Figura 10-2. Sensor SY-FX2
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.5 Selección del módulo de comunicación

En esta parte se muestra la selección del módulo de comunicación utilizado por el dispositivo.

2.5.1 *Módulo Shield*

En el capítulo 1 se optó por el uso de la comunicación GPRS, ya que se requiere enviar la información a una base de datos vía internet desde el lugar de la instalación y como no existe ninguna red de wi-fi cercana, se escogió la red celular para conectarse a internet. Y para la selección del módulo adecuado se comparó a los módulos GPRS SIM900 y GPRS SIM808. A continuación, se muestran las características principales de cada módulo para su selección.

Tabla 6-2: Características técnicas de los módulos de comunicación

Características	SIM900	SIM808
Velocidad de transmisión	42.8/85.6 Kbps	85,6 Kbps
Ancho de banda	850, 900, 1800 y 1900 MHz	850, 900, 1800 y 1900 MHz
Voltaje	4.8 – 5.2 V	5 – 12 V
Corriente	50 – 450 mA	2 – 500 mA
Potencia de transmisión	2W a 850 o 900 MHz	2W a 850 o 900 MHz
Precio	\$38	\$45

Fuente: (SIMCom, 2013, p. 2)(SIMcom, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Con los datos indicados en la tabla 6-2 se escogió al módulo Shield GPRS SIM900 mostrado en la figura 11-2, ya que su precio es más bajo que el SIM808, aunque sus características sean similares. En el anexo G se indica el datasheet del mismo.



Figura 11-2: Módulo GPRS SIM900

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.6 Módulo LM2596

Este módulo es un tipo de regulador de voltaje capaz de brindar el voltaje y la corriente necesaria para alimentar al Arduino y al módulo GPRS soportando valores de 4,5 hasta 40 V de entrada con una excelente regulación en línea y bajos voltajes de rizado evitando el uso o el diseño de otras fuentes de alimentación. En la figura 12-2 se muestra el módulo LM2596 y en el anexo H se indica el datasheet del mismo.

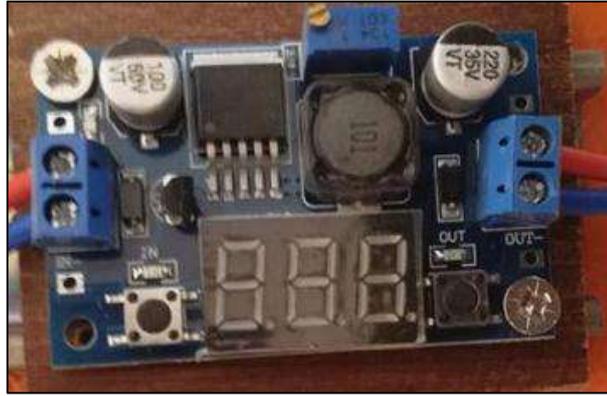


Figura 12-2: Módulo LM2596
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 7-2 se muestran las características principales del módulo LM2596.

Tabla 7-2: Características técnicas del módulo LM2596

Características	
Voltaje de Entrada	4,5-40 V
Voltaje de Salida	1,5-35 V (Ajustable)
Corriente de Operación	50mA
Corriente de Salida	3A

Fuente: (Electronilab, 2019a)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.7 Módulo LCD 2004A

Es un módulo display con un formato de 20 caracteres y 4 líneas de muy bajo consumo. Es adaptable con Arduino de fácil instalación y programación, adecuado para la muestra de las variables medidas por el dispositivo. En la figura 13-2 se muestra el módulo LCD 2004A y en el anexo I se indica el datasheet del mismo.



Figura 13-2: Módulo LCD 2004A
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 8-2 se muestran las características principales del módulo LCD 2004A.

Tabla 8-2: Características técnicas del módulo LCD 2004A

Características	
Voltaje de Entrada	5V
Corriente de Operación	4mA

Fuente: (Powertip, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.8 Diseño de la etapa de alimentación

Para el diseño de la etapa de alimentación con paneles fotovoltaicos se tomó en cuenta varios parámetros como: la energía diaria consumida por todos los elementos del dispositivo, las horas de sol disponible en el sector de la instalación, la dimensión del panel fotovoltaico, con el número de paneles a utilizar y su potencia, la dimensión de la batería, su capacidad, tiempo de autonomía y la dimensión del controlador de carga (Caluña Tisalema y Jordán Vargas, 2017).

2.8.1 Energía diaria consumida

Para el cálculo de la energía a consumir diariamente en nuestro dispositivo, se debe calcular la potencia nominal de cada elemento del dispositivo (Caluña Tisalema y Jordán Vargas, 2017); de acuerdo con la ecuación 1-2:

Ecuación 1-2: Potencia nominal

$$P = V \times I$$

En donde:

P: Potencia nominal

V: Voltaje de operación de cada dispositivo

I: Corriente consumida por cada dispositivo

Una vez calculada la potencia nominal de cada elemento, se indica el tiempo de funcionamiento de cada uno para el cálculo de la energía diaria consumida (Caluña Tisalema y Jordán Vargas, 2017); de acuerdo con la ecuación 2-2:

Ecuación 2-2: Energía diaria consumida

$$EDC = P \times H$$

En donde:

EDC: Energía diaria consumida

P: Potencia nominal.

H: Tiempo de funcionamiento.

Ahora sumamos la energía diaria consumida por cada elemento para obtener la energía total consumida por el dispositivo. En la tabla 9-2, se muestran los elementos utilizados por el dispositivo para el cálculo de la energía total consumida por el dispositivo.

Tabla 9-2: Cálculo de la energía total diaria consumida por el dispositivo

Elementos	Potencia nominal (W)	Tiempo de operación (H)	Energía diaria consumida (WH)
Arduino MEGA 2560	0,558	24	13,392
Sensor DHT11	0,0125	24	0,3
Sensor BMP180	0,0000099	24	0,0002376
Módulo sensor FC-37	0,075	24	1,8
Sensor SY-FS2	0,21	24	5,04
Sensor SY-FX2	0,21	24	5,04
Módulo GSM/GPRS SIM900	2,7	24	64,8
Módulo LM2596	0,6	24	14,4
Módulo LCD 2004A	0,02	24	0,48
		Energía Total Consumida	105,25

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

La energía total consumida por el dispositivo es de 105,25 WH, por lo que se consideró la eficiencia de la batería para el dispositivo, la cual debe ser de 12V y con una eficiencia (η) de 85%, con lo cual se procedió a realizar un nuevo cálculo para la energía total diaria consumida; de acuerdo con la ecuación 3-2:

Ecuación 3-2: Energía total diaria consumida

$$ET = \frac{EDC}{\eta}$$

En donde:

ET: Energía total diaria consumida.

EDC: Energía diaria consumida.

η : Eficiencia.

Mediante el cálculo anterior se obtuvo el valor de la Energía Total a consumir nuestro dispositivo diariamente, el cual da un valor de: 123,82 WH; En base a esto se debe seleccionar el panel solar adecuado para el sistema fotovoltaico.

2.8.2 *Dimensión del panel fotovoltaico*

Para calcular la dimensión del panel fotovoltaico se debe considerar las horas de pico solar (HPS) en el lugar de la instalación del dispositivo cuyo valor es de 3, ya que son los registros mínimos en temporada de invierno en todo el Ecuador. También se indica el uso de un panel fotovoltaico de 110W con un factor global de pérdidas de 0,65 que es el mínimo en el que se encuentra el panel y la energía total diaria consumida por el dispositivo (Caluña Tisalema y Jordán Vargas, 2017). Con todo esto se procede al cálculo del número de paneles necesarios a utilizar; de acuerdo con la ecuación 4-2:

Ecuación 4-2: Numero de paneles

$$NT = \frac{ET}{Pp * HPS * Fg}$$

En donde:

NT: Número de paneles

ET: Energía total diaria consumida.

HPS: Horas de pico solar

Pp: Potencia pico del panel fotovoltaico

Fg: Factor global de pérdidas

Según los cálculos realizados se obtuvo la necesidad de 0,58 paneles, por lo tanto, para el sistema fotovoltaico a instalar se debe utilizar una potencia mínima de 65W, pero como no existen paneles con este valor se optó por el uso del panel de 110W mostrado en la figura 14-2.



Figura 14-2: Panel fotovoltaico
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En cuanto a la colocación del panel fotovoltaico es recomendable colocarlo a un ángulo entre 5° hasta los 15°, por lo que se escogió un ángulo de 10° con dirección de Este a Oeste para tener un mejor aprovechamiento del sol en el lugar de la instalación (Moro Vallina, 2010). En el anexo J se indica el datasheet del mismo.

En la tabla 10-2 se muestran las características principales del panel fotovoltaico.

Tabla 10-2: Características técnicas del panel fotovoltaico

Características	
Potencia máxima	110W
Corriente máxima	5,97A
Voltaje máximo	18,43V
Corriente de corto circuito	6,48A
Voltaje de corto circuito	22,51V

Fuente: (Energy, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.8.3 *Dimensión de la batería*

Para la dimensión de la batería del dispositivo se debe tener en cuenta su voltaje nominal que son de 12V, así como la profundidad de descarga que se lo considera en un valor de 0,7, los días de autonomía requeridos por el dispositivo que en este caso son 2 y la energía total diaria consumida.

Al tener todos estos parámetros se procede al cálculo de la capacidad de la batería; de acuerdo con la ecuación 5-2:

Ecuación 5-2: Capacidad de la batería

$$C = \frac{D * ET}{V_{bat} * Pd}$$

En donde:

C: Capacidad de la batería.

ET: Energía total diaria consumida.

D: Días de autonomía

V_{bat}: Voltaje de la batería

Pd: Máxima profundidad de descarga

Con los cálculos realizados se obtuvo que la batería debe tener una capacidad de 29,48AH a 12V de acuerdo con las cargas a utilizar en el dispositivo, por lo cual se escogió una batería de igual voltaje con una capacidad de 55AH mostrado en la figura 15-2, ya que no se encontró baterías solares con la misma dimensión de lo calculado. En el anexo K se indica el datasheet del mismo.



Figura 15-2: Batería Solar

Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 11-2 se muestran las características principales de la batería solar.

Tabla 11-2: Características técnicas del batería solar

Características	
Voltaje nominal	12V
Capacidad nominal	55AH
Vida útil	12 años a 20°C

Fuente: (Ultracell, 2019, p. 1-2)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.8.4 Dimensión Controlador de carga

Para dimensionar el controlador de carga se considera la corriente máxima del panel solar que es de 5,97A dando como resultado la selección de un controlador capaz de soportar 6A mostrado en la figura 16-2. En el anexo L se indica el datasheet del mismo.



Figura 16-2: Controlador de carga
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 12-2 se muestran las características principales de la batería solar.

Tabla 12-2 Características técnicas del controlador de carga

Características	
Voltaje nominal	12V
Corriente máxima de consumo	8mA

Fuente:(Morningstar, 2019)

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

2.9 Esquema de conexiones

En la figura 17-2 se muestra el esquema de conexiones del hardware del dispositivo con sus respectivos sensores, dispositivo de comunicación, pantalla LCD, regulador de voltaje y el sistema de alimentación.

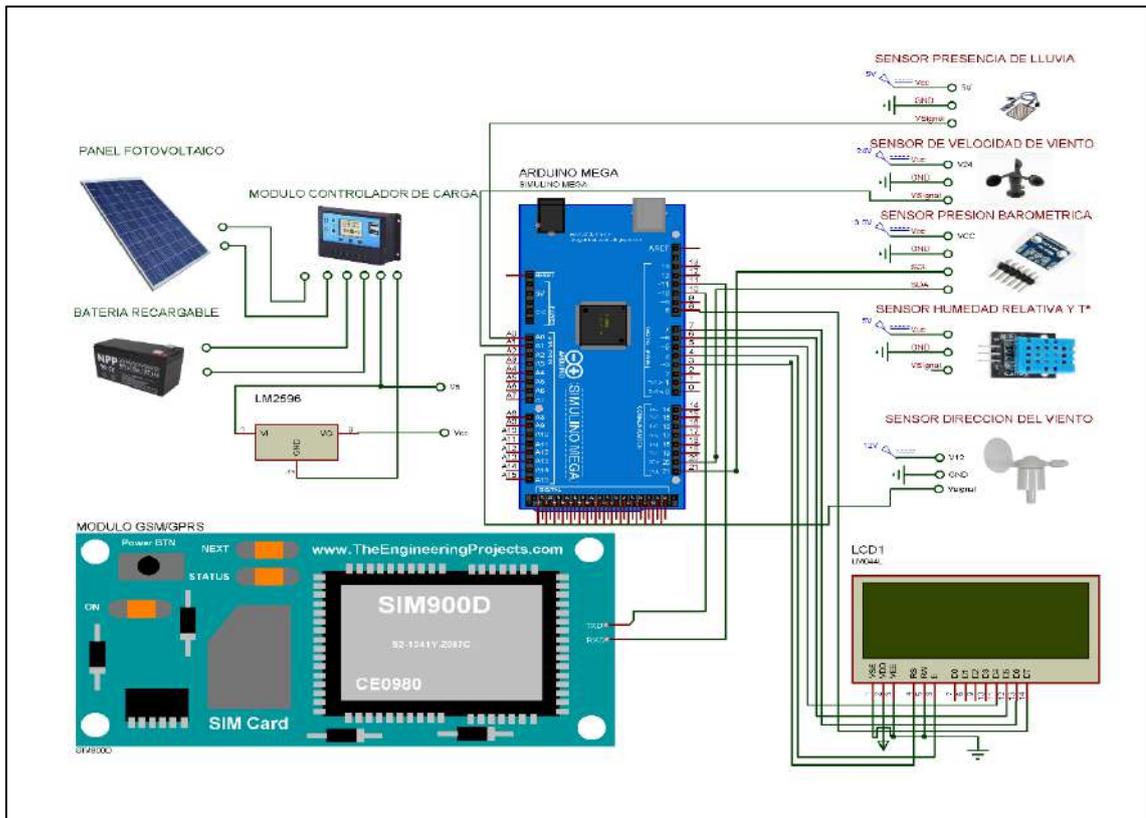


Figura 17-2 Esquema de conexión electrónica

Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

El esquema de conexiones del dispositivo se lo detallará a continuación:

- Para la conexión del módulo GSM/GPRS SIM900 y el Arduino MEGA 2560 se utilizaron los terminales digitales D7 y D8 del módulo y los terminales digitales D10 y D11 del Arduino para obtener una comunicación serial entre los dos, conectando D7 con D10 y D8 con D11 respectivamente.
- Los sensores de velocidad y dirección del viento se los alimentó con los 12V entregados por el controlador de carga y se los conectó al Arduino MEGA 2560 en los terminales análogos A1 y A2 respectivamente, utilizando la salida de 0-5 V de ambo sensores.
- El sensor DHT11 se lo alimentó con los 5V suministrados por el Arduino MEGA 2560 y se lo conectó al pin digital D2 del Arduino.
- El sensor BMP180 se lo alimentó con 3.3V que fueron suministrados por el Arduino MEGA 2560 y se conectó a los terminales digitales D20(SDA) y D21(SCL) del Arduino siendo estos los terminales para una comunicación I2C utilizados por el sensor de presión.
- Al módulo sensor FC-37 se lo alimentó con 5V suministrados por Arduino MEGA 2560 conectándolo con el pin análogo A0 del Arduino.
- Para la conexión de la pantalla LCD 2004A se conectaron los terminales 1(VSS), 3(VO), 5(RW) y 16(K) del LCD a GND, los terminales 2(VDD) y 15(A) a 5V y los terminales

4(RS), 6(E), 11(D4), 12(D5), 13(D6), 14(D7) a los terminales digitales D3, D4, D5, D6, D7 y D8 del Arduino MEGA 2560 respectivamente.

- En la etapa de alimentación, lo primero que se realizó fue la conexión de la batería de 12V al controlador de carga, seguido del panel fotovoltaico, en donde se conectó el terminal positivo a un breaker para la desconexión con el controlador y para la protección del dispositivo se colocó dos fusibles de 6A en cada terminal del panel en caso de una elevada corriente que sobrepase a la permitida. Por último, se realizó la conexión al regulador de voltaje LM2596 con una tensión de entrada de 12V para obtener una tensión de salida de 6V la cual ayudara a alimentar al Arduino MEGA 2560 y al módulo GSM/GPRS SIM900.

2.10 Software del dispositivo

En esta sección se muestra el software utilizado para la lectura, muestra y envío de los datos, como también el software para el desarrollo de la aplicación.

2.10.1 Requerimientos del software del dispositivo

- Ser de licencia gratuita.
- Ser compatible con las librerías utilizadas.
- Enviar las mediciones a la base de datos.
- Acceder a la información de la base de datos para mostrarla en la aplicación.

2.11 Software para la lectura, muestra y envío de datos.

El software utilizado para la programación en la plataforma Arduino es el IDE que se basa en C++, es un código el cual ofrece muchas librerías para la realización de distintas operaciones (Banzhi et al., 2007). Este software también incluye un monitor serial para enviar los datos simples de texto y desde la placa.

2.11.1 Programación en el IDE de Arduino

La programación se la dividió en dos partes, una para la lectura y muestra de las variables meteorológicas y la otra para la comunicación y envío de la información a la base de datos, el código completo se encuentra en el anexo M.

Para la lectura y muestra de las variables meteorológicas se realizó un diagrama de flujo mostrado en la figura 18-2 que se lo detallara a continuación:

- Inicio.
- Declarar las librerías para el uso del DHT11, BMP180 y el LCD
- Configurar los terminales para la conexión con el LCD
- Crear las variables para la lectura de la temperatura, presión, humedad, velocidad, dirección y lluvia.
- Iniciar el LCD y el BMP180.
- Realizar la lectura de los datos.
- Realizar la conversión de los datos.
- Mostrar los datos en la pantalla LCD.
- Fin.

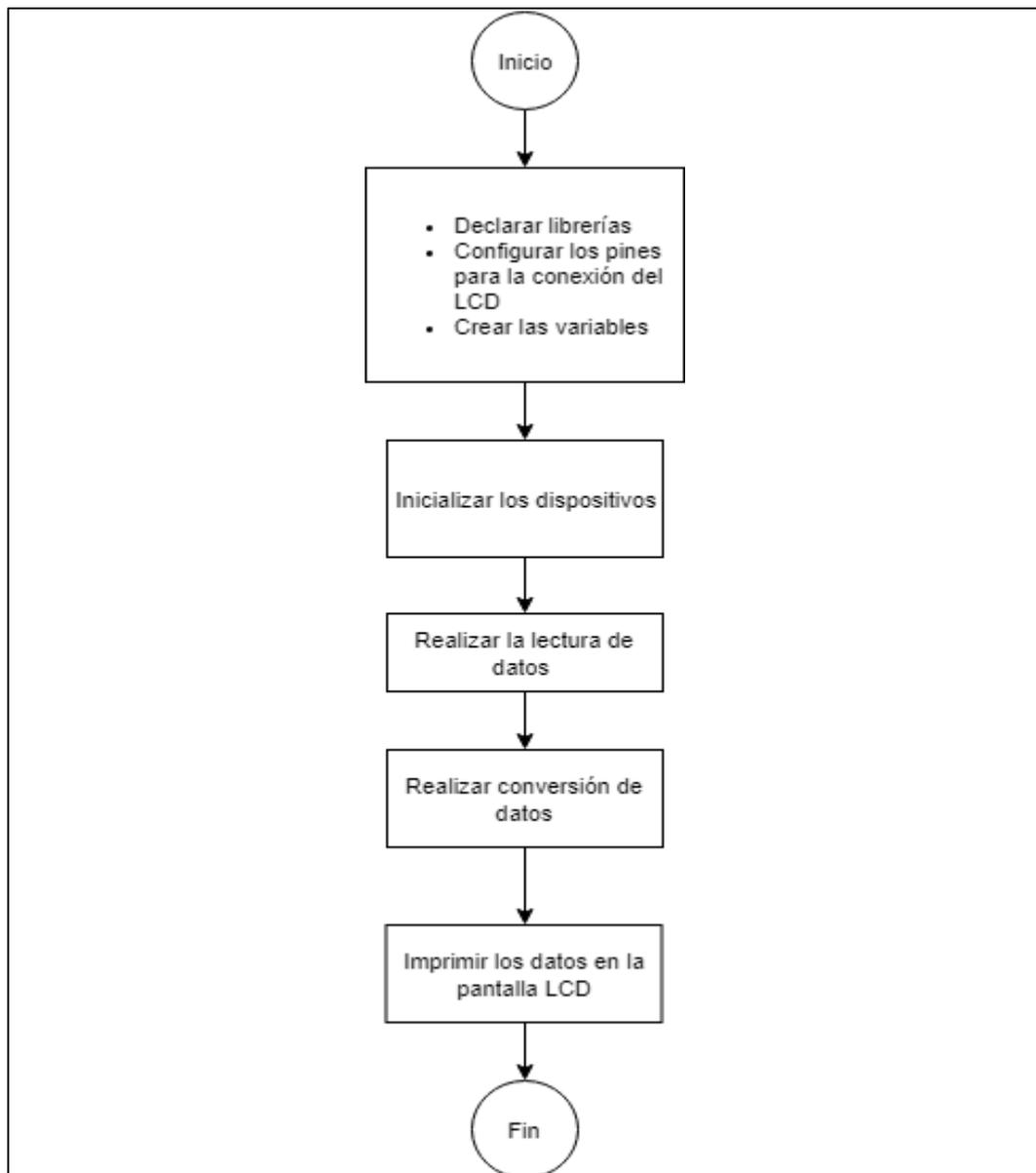


Figura 18-2. Diagrama de flujo de la lectura y visualización de las variables meteorológicas
 Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

Las librerías y funciones utilizadas en la programación de la lectura y muestra de los datos son las siguientes:

- **Lib. LiquidCrystal:** Con esta librería se crea un objeto un objeto que representa al módulo LCD para tener una fácil programación.
- **Lib. Wire:** Esta librería permite que el sensor BMP180 se comunique mediante el protocolo I2C.
- **Lib. Adafruit_BMP085:** Con esta librería se crea un objeto un objeto que representa al sensor BMP180 para tener una fácil programación.
- **Lib. dht11:** Con esta librería se crea un objeto que representa al sensor DHT11 para tener una fácil programación.

- **Func. setup:** En esta función se inicia el módulo LCD y el sensor BMP180.
- **Func. sensores:** En esta función se realiza la lectura y conversión de las señales obtenidas para luego ser mostradas en el módulo LCD.

Para la comunicación y envío de la información a la base de datos se realizó un diagrama de flujo mostrado en la figura 19-2 que se lo detallara a continuación:

- Inicio.
- Declarar las librerías para el uso del GPRS SIM900.
- Configurar los terminales para la comunicación serial.
- Establecer la velocidad del puerto serie.
- Esperar 10 segundos para conectarse a la red del operador.
- Consultar el estado de la conexión.
- Configurar un IP única.
- Configurar la APN, nombre de usuario y contraseña del operador.
- Realizar la activación de los datos inalámbricos del GPRS.
- Obtener un IP local.
- Establecer un indicador de los datos a enviar.
- Indicar la URL, puerto y tipo de conexión.
- Verificar la conexión con el servidor.
- Enviar los datos.
- Cerrar el contexto PDP del GPRS.
- Fin.

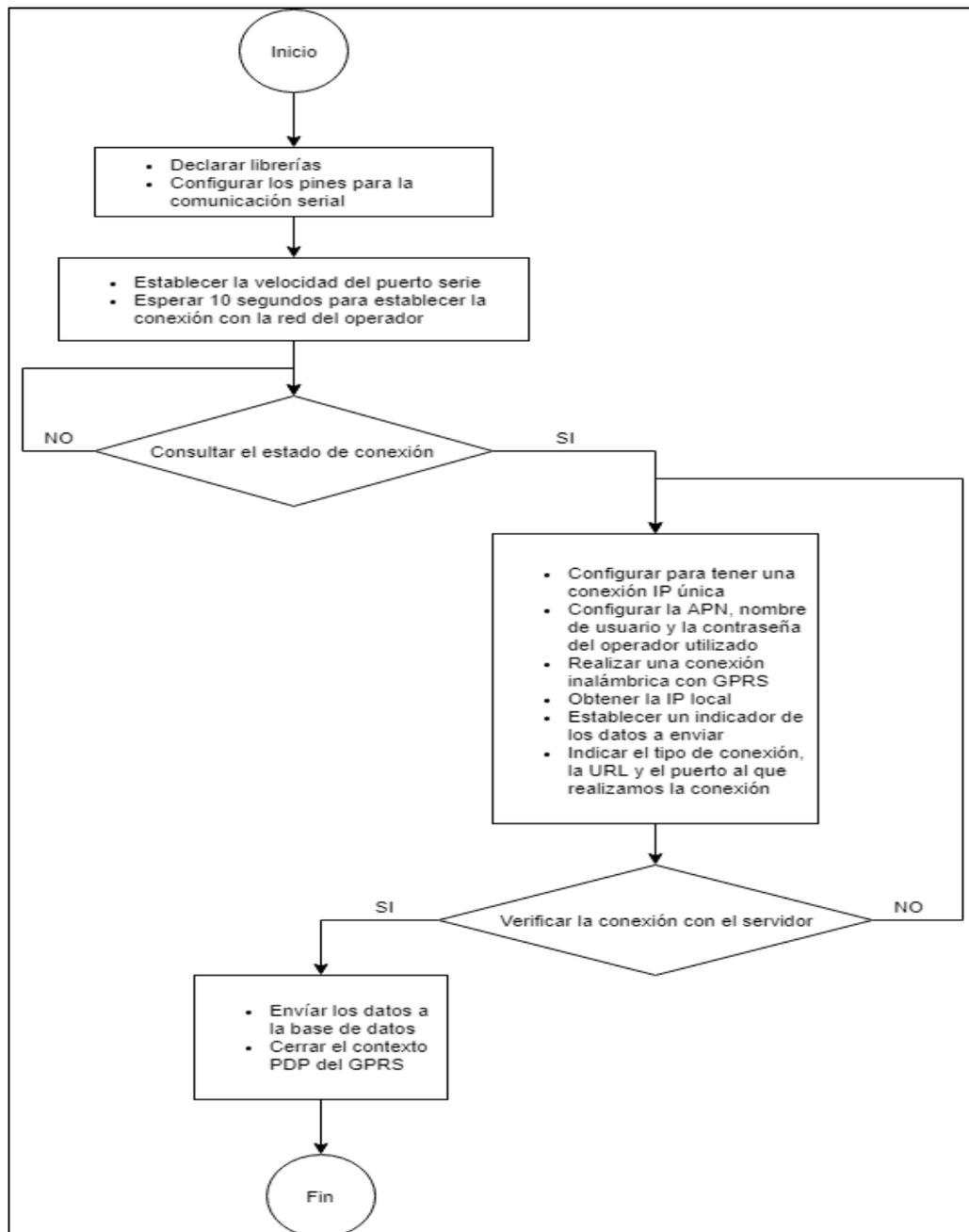


Figura 19-2. Diagrama de flujo de la comunicación y envío de información a la base de datos
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

Las librerías y funciones utilizadas para la comunicación y envío de la información a la base de datos son las siguientes:

- **Lib. SoftwareSerial:** Con esta librería se permite tener comunicación serial.
- **Func. Setup:** En esta función se configura la velocidad del puerto serie del Arduino y el GPRS.
- **Func. Loop:** En esta función se llama a la función comGPRS.
- **Func. comGPRS:** Esta función realiza la comunicación con la base de datos Thingspeak para enviar la información obtenida por la función sensores.

2.12 Software para el desarrollo de la aplicación

La aplicación se desarrolla en un entorno de trabajo llamado Android Studio, basado en el lenguaje de programación java, que brinda la facilidad de personalización grafica permite realizar de manera sencilla la comunicación deseada entre la aplicación y la base de datos.

2.12.1 Programación en Android Studio

La programación de la aplicación Android se puede dividir en dos procesos, el primero encargado de obtener los datos almacenados en el servidor y procesarlos para poder ser visualizados en la interfaz gráfica, dicho proceso se puede observar en el diagrama de flujo de la figura 20-2, y el segundo encargado de realizar el algoritmo que envía un mensaje una vez procesada la información el código completo se encuentra en el anexo N.

El diagrama de flujo de la figura 20-2 se lo detalla a continuación:

- Inicio
- Declaración de variables a utilizar
- petición XMLHTTP del tipo GET
- obtención de los datos
- Procesamiento de los datos obtenidos
- Verificación de la posibilidad del vuelo
- Mostrar datos procesados
- Mostrar consejo de vuelo
- Fin

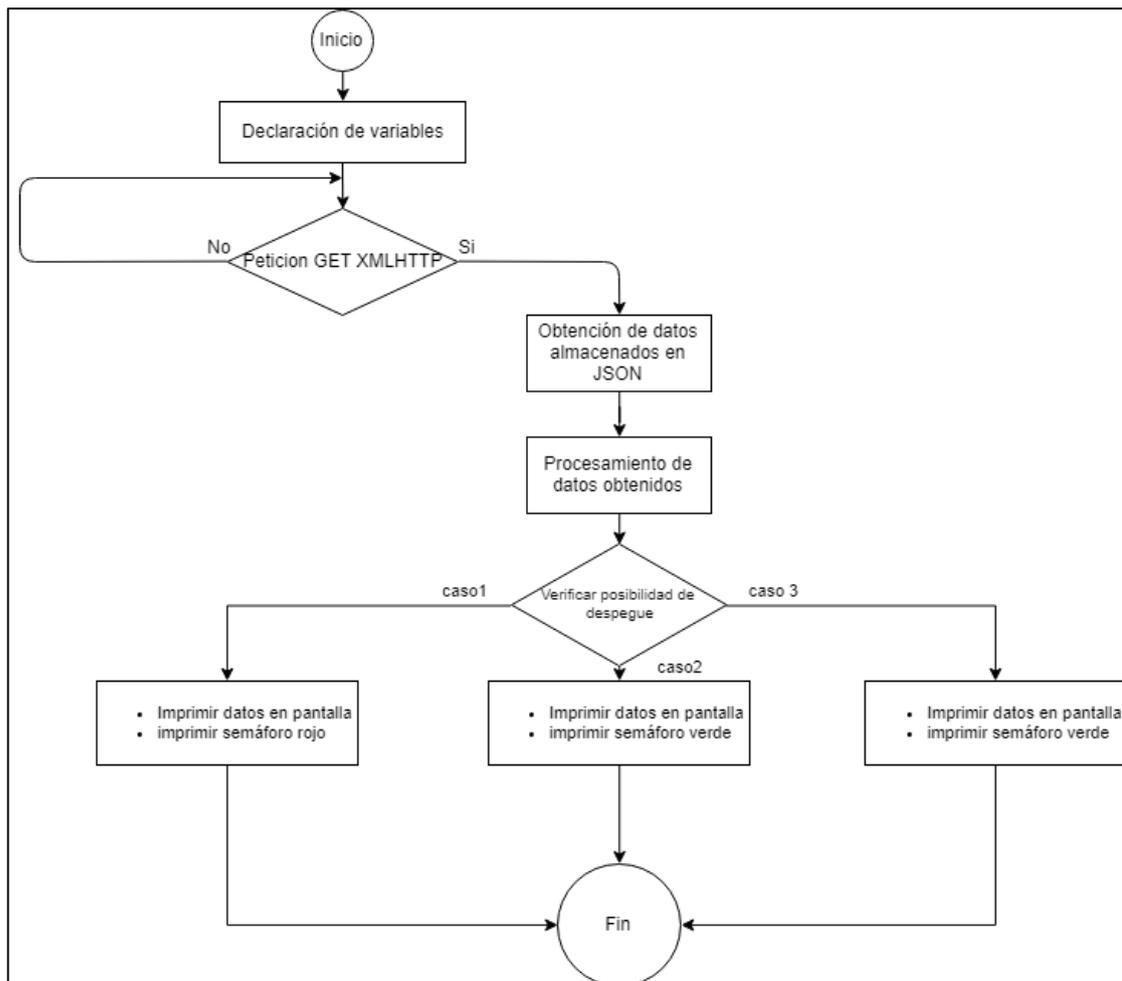


Figura 20-2. Diagrama de flujo de la comunicación de la aplicación con el servidor.
Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

A continuación, se muestra las funciones y librerías utilizadas en el desarrollo de la aplicación móvil para Android:

- **Lib Volley:** Es una librería creada para realizar peticiones Http de manera sencilla y directa.
- **Lib Json:** Esta librería nos permite reconocer archivos JSON, que es una manera sencilla de ordenar datos en matrices.
- **Func. obtenerDatosVolley:** Esta función está construida para realizar las peticiones Http y así obtener los elementos deseados del JSON almacenado en la nube y el procesamiento de los mismos.
- **Func. OnCreate:** En esta función se realiza la verificación de la posibilidad de vuelo y se crea un mensaje de advertencia basándose en las condiciones meteorológicas.

CAPÍTULO III

3. PRUEBAS Y RESULTADOS

En el presente capítulo se realizan las pruebas para la validación de los sensores utilizados y se verifica el envío de la información hacia la base de datos de Thingspeak, la publicación en la app Android y el análisis de costos del dispositivo implementado. En la figura 1-3 se muestra el dispositivo final implementado.

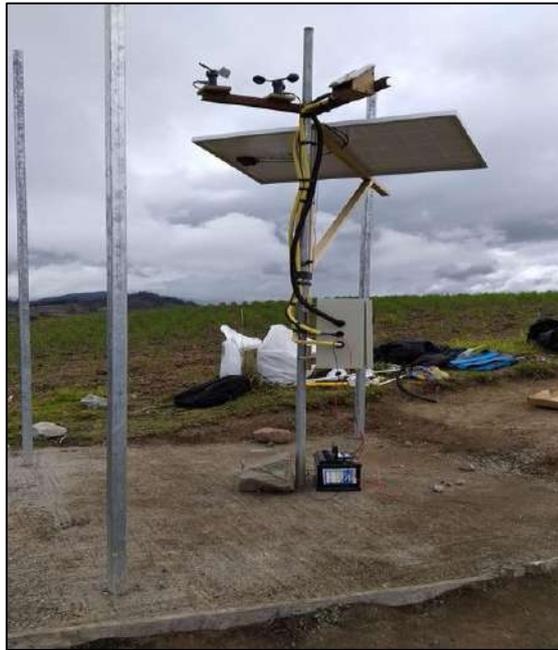


Figura 1-3. Dispositivo final implementado
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

3.1 Validación de sensores

En esta sección se presenta las validaciones de cada uno de los sensores utilizados en la implementación del dispositivo.

3.1.1 Validación de medición de temperatura

Para validar las mediciones de la temperatura del sensor BMP180, se lo comparó con el equipo HTC-1 mostrado en la figura 2-3 que tiene como rangos de medición de -10°C a 50°C , con una exactitud de $\pm 1^{\circ}\text{C}$ entre 0°C a 50°C y una resolución de 0.1°C . Se tomaron 20 muestras durante un día dando como resultados los datos obtenidos en la tabla 1-3.



Figura 2-3. HTC-1

Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

Tabla 1-3: Comparación de los valores de temperatura medidos entre el HTC-1 y el BMP180

Muestra	Equipo patrón HTC-1 ($^{\circ}\text{C}$)	BMP180 ($^{\circ}\text{C}$)	Error absoluto
1	19.8	19.2	0.6
2	19.7	19.3	0.4
3	20.3	19.8	0.5
4	20.5	19.8	0.7
5	20.3	20.3	0
6	20.1	20.6	0.5
7	19.7	20.2	0.5
8	19.4	19.0	0.4

9	19.2	19.1	0.1
10	18.8	19.0	0.2
11	18.8	18.8	0
12	18.8	18.5	0.3
13	18.7	18.4	0.3
14	18.1	18.1	0
15	17.9	18.5	0.6
16	17.7	18.2	0.5
17	17.5	17.8	0.3
18	17.6	17.5	0.1
19	17.4	17.2	0.2
20	17.2	17.6	0.4

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 1-3 se observa que el error máximo absoluto es de 0,7°C, siendo un valor que está dentro de los rangos de error permitidos por el HTC-1 y el BMP180, concluyendo que el sensor no incluye error al dispositivo, por lo que la precisión del sensor al medir la temperatura es válida.

3.1.2 Validación de medición de presión atmosférica

Para validar las mediciones de la presión atmosférica del sensor BMP180, se lo comparó con el barómetro clásico de mercurio ubicado en la estación meteorológica de la ESPOCH mostrado en la figura 3-3. Se tomaron 20 muestras durante 10 horas dando como resultados los datos obtenidos en la tabla 2-3.



Figura 3-3. Barómetro clásico de mercurio
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

Tabla 2-3: Comparación de los valores de presión atmosférica medidos entre el barómetro clásico de mercurio y el BMP180.

Muestra	Equipo patrón barómetro clásico de mercurio (hPa)	BMP180 (hPa)	Error absoluto
1	726.47	726.55	0.08
2	726.41	726.48	0.07
3	726.43	726.47	0.05
4	726.44	726.48	0.04
5	726.49	726.57	0.06
6	726.40	726.46	0.06
7	726.47	726.53	0.06
8	726.50	726.54	0.04
9	726.47	726.52	0.05
10	726.33	726.39	0.06
11	726.38	726.45	0.07
12	726.42	726.47	0.05

13	726.37	726.41	0.04
14	726.35	726.40	0.05
15	726.39	726.45	0.06
16	726.41	726.47	0.06
17	726.43	726.48	0.05
18	726.46	726.52	0.06
19	726.47	726.53	0.06
20	726.49	726.56	0.07

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 2-3 se observa un error máximo absoluto de 0.08 hPa, cuyo valor se encuentra en el rango de error permitido por el sensor BMP180, el cual no incluye error al dispositivo validando el correcto funcionamiento del sensor.

3.1.3 Validación de medición de humedad relativa

Para validar las mediciones de la humedad relativa del sensor DHT11, se lo comparó con el equipo HTC-1 mostrado en la figura 1-3 que tiene como rangos de medición de 10%RH a 99%RH, con una exactitud de $\pm 5\%$ y una resolución de 1%. Se tomaron 20 muestras durante un día dando como resultados los datos obtenidos en la tabla 3-3.

Tabla 3-3: Comparación de los valores de humedad relativa medidos entre el HTC-1 y el DHT11

Muestra	Equipo patrón HTC-1 (%)	DHT11 (%)	Error absoluto
1	95	92	3
2	89	87	2
3	91	90	1

4	92	94	2
5	82	82	0
6	75	75	0
7	81	82	1
8	87	90	3
9	95	94	1
10	91	89	2
11	88	89	1
12	86	85	1
13	81	81	0
14	78	78	0
15	83	81	2
16	88	87	1
17	91	89	2
18	94	92	2
19	90	91	1
20	88	87	1

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 3-3 se indica un error máximo absoluto del 3%RH, siendo un valor que está dentro de los rangos de error permitidos por el HTC-1 y el DHT11, por lo que no incluye error al dispositivo dando como válido el funcionamiento del sensor.

3.1.4 Validación de medición de velocidad del viento

Para validar las mediciones de la velocidad del viento del sensor SY-FS2, se lo comparó con el equipo Skywatch Xplorer 1 mostrado en la figura 4-3, que tiene como rangos de medición de 0 a

150 Km/h, con una precisión de $\pm 3\%$ y una resolución de 0,1 Km/h. Se tomaron 20 muestras durante una hora dando como resultados los datos obtenidos en la tabla 4-3.



Figura 4-3. Skywatch Xplorer 1
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

Tabla 4-3: Comparación de los valores de velocidad del viento medidos entre el Skywatch Xplorer 1 y el SY-FS2

Muestra	Equipo patrón Skywatch Xplorer 1 (Km/h)	SY-FS2 (Km/h)	Error absoluto
1	8,6	8,4	0,2
2	10,7	10,4	0,3
3	3,3	3,5	0,2
4	14,4	14,1	0,3
5	10,2	9,9	0,3
6	11,3	11,4	0,1
7	11,1	10,9	0,2
8	8,7	8,5	0,2

9	5,7	5,6	0,1
10	2,5	2,5	0
11	4,9	5,0	0,1
12	5,2	5,3	0,1
13	5,7	5,6	0,1
14	7,8	7,7	0,1
15	8,2	7,9	0,2
16	8,8	8,6	0,2
17	9,1	8,9	0,2
18	9,4	9,2	0,2
19	9,0	9,2	0,2
20	8,8	8,6	0,2

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 4-3 se observa un error máximo absoluto de 0,3 Km/h, estando en el rango de error permitido por el Skywatch Xplorer 1 y el SY-FS2, por lo que no incluye error al dispositivo dando como válido el funcionamiento del sensor.

3.1.5 Validación de medición de la dirección del viento

Para validar las mediciones de la dirección del viento del sensor SY-FX2, se basó en el voltaje de salida que nos entrega el sensor para determinar la posición. Con todo esto se realizó la tabla 5-3 donde se indica los rangos de voltaje en el que se encuentra cada dirección. Para verificar los resultados se comparó con una aplicación de brújula del teléfono Huawei p20 lite como se aprecia en la figura 4-3.

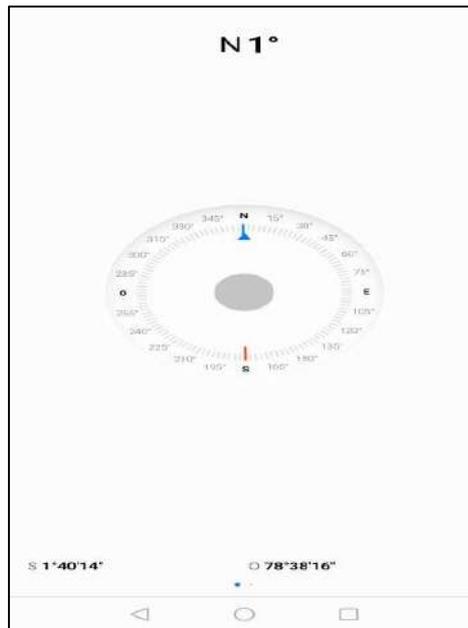


Figura 5-3. Aplicación de brújula
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

Tabla 5-3: Valores de la dirección del viento medidos con el SY-FS2

	Rango (V)	Dirección
1	4,70 – 0,31	NORTE
2	0,32 – 0,94	NORESTE
3	0,95 – 1,56	ESTE
4	1,57 – 2,19	SURESTE
5	2,20 – 2,81	SUR
6	2,82 – 3,44	SUROESTE
7	3,45 – 4,06	OESTE
8	4,07 – 4,69	NOROESTE

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 5-3 se muestran diferentes rangos de voltaje que son los que nos indicaran la dirección en la que se encuentre el sensor.

3.1.6 Validación de medición de la lluvia

Para validar si existe o no lluvia con el módulo sensor FC-37, se tomaron 20 muestras que ayudaron a obtener los valores analógicos en voltios del sensor para indicar el rango de cuándo se va a encontrar lloviendo o no. En la tabla 6-3 se muestran los valores tomados bajo una lluvia artificial causada por una manguera de agua y en condiciones ambientales sin lluvia.

Tabla 6-3: Medición de la presencia de lluvia con el módulo sensor FC-37

Valor	Tensión del módulo sensor FC-37 (V)	Estado
1	0	Con lluvia
2	0,25	Con lluvia
3	0,50	Con lluvia
4	0,75	Con lluvia
5	1,0	Con lluvia
6	1,25	Con lluvia
7	1,50	Con lluvia
8	1,75	Con lluvia
9	2,0	Con lluvia
10	2,25	Con lluvia
11	2,50	Con lluvia
12	2,75	Sin lluvia
13	3,0	Sin lluvia
14	3,25	Sin lluvia
15	3,50	Sin lluvia
16	3,75	Sin lluvia

17	4,0	Sin lluvia
18	4,25	Sin lluvia
19	4,50	Sin lluvia
20	4,75	Sin lluvia

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Con la tabla 6-3 se obtuvo los rangos de 0V a 2,74V para presencia de lluvia y de 2,75V a 4,75V sin presencia de lluvia dándonos como resultado la veracidad de las mediciones realizadas por el módulo sensor FC-37.

3.2 Repetitividad de los sensores

Para analizar la respuesta del dispositivo ante un estímulo continuo se realizó una prueba en la que se tomaron mediciones en un ambiente controlado durante un corto periodo de tiempo. La prueba consistió en tomar 20 muestras cada 1 minuto a los sensores de temperatura, presión atmosférica, humedad relativa, detector de lluvia, dirección y velocidad del viento. Para el detector de lluvia se utilizó un rociador de agua y para los sensores de dirección y velocidad del viento se usó un secador de cabello para realizar la prueba. La repetitividad es cuantificable, por lo que a través de la media (\bar{x}) y la desviación estándar (σ) se puede determinar el coeficiente de variación $C.V$, cuya ecuación es $C.V = (\sigma/\bar{x})*100$. A continuación, en la tabla 7-3 se muestra la repetitividad de cada sensor evaluado.

Tabla 7-3: Repetitividad de los sensores evaluados

Muestra	BMP180		DHT11	YL-38	SY-FS2	SY-FX2
	Temperatura (°C)	Presión (hPa)	Humedad (%)	Lluvia (V)	Velocidad (Km/h)	Dirección (V)
1	23.5	737.80	61	1.66	23.4	1.25
2	23.6	737.85	63	1.67	23.5	1.28
3	23.6	737.89	63	1.62	23.6	1.30

4	23.7	737.83	62	1.61	23.5	1.27
5	23.6	737.90	63	1.58	23.5	1.26
6	23.5	737.82	63	1.69	23.4	1.31
7	23.6	737.93	61	1.60	23.6	1.29
8	23.5	737.95	61	1.61	23.7	1.32
9	23.7	737.87	62	1.62	23.6	1.30
10	23.6	737.89	62	1.57	23.6	1.27
11	23.7	737.90	62	1.56	23.5	1.28
12	23.7	737.91	63	1.60	23.6	1.25
13	23.6	737.87	62	1.62	23.5	1.30
14	23.5	737.94	62	1.64	23.4	1.32
15	23.6	737.88	62	1.62	23.8	1.31
16	23.5	737.95	61	1.61	23.7	1.28
17	23.6	737.90	61	1.66	23.8	1.30
18	23.6	737.95	62	1.63	23.6	1.27
19	23.5	737.98	62	1.59	23.6	1.26
20	23.5	737.95	62	1.57	23.8	1.29
Media(\bar{x})	23.59	737.90	62	1.62	23.59	1.29
Desviación Estándar (σ)	0.072629195	0.047497367	0.707106781	0.033952172	0.123592070	0.021324868
Coefficiente de variación (C.V) %	0.307881284	0.006436829	1.140494808	2.095813086	0.52391721	1.653090543

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Con los datos de la tabla 7-3 se obtiene que el coeficiente de variación de la temperatura y presión del BMP180 son de 0.31% y 0.01% respectivamente, el de humedad del DHT11 es de 1.14%, el de lluvia del YL-38 es de 2.10%, de la velocidad del SY-FS2 es de 0.52% y el de la dirección del SY-FX2 es de 1.65%, cuyos valores presentan poca variabilidad menor al 10%, por lo que se concluye que las respuestas brindadas por el dispositivo son aceptables (Llamosa, Meza Contreras y Botero, 2007).

3.3 Evaluación de envío de datos al servidor

Se realizó un análisis para observar el comportamiento del envío de los datos medidos, para ellos se analizó una muestra en un periodo de 12 horas donde se enviaba un dato cada 2 minutos, en las figuras 6-3,7-3 y 8-3 se puede observar un ejemplo de cómo se realizó la prueba, para esta prueba se optó por utilizar un proveedor de servicio CNT ya que en el lugar de la implementación era el que brindaba mayor cobertura. En la tabla 8-3 se observa un resumen de los resultados obtenidos en la plataforma IoT. El listado completo de todos los datos receptados se puede observar en el ANEXO P.



Figura 6-3. Datos procesados listos para el envío
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

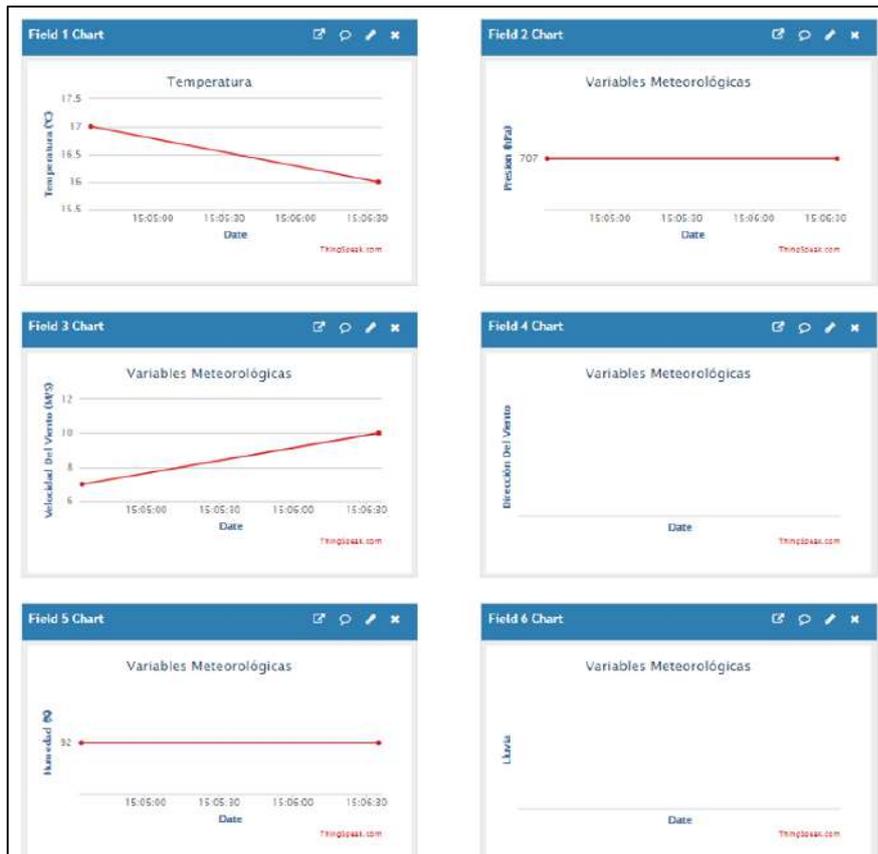


Figura 7-3. Datos recibidos en Thingspeak
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019

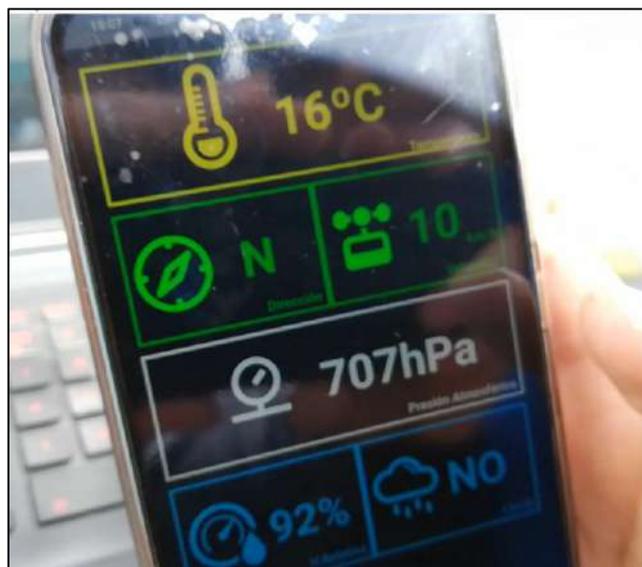


Figura 8-3. Datos visualizados en la aplicación móvil
Realizado Por: Coronel Jhonatan, Espinosa Marcos, 2019.

Tabla 8-3: Resumen de datos enviados y recibidos en la plataforma IoT

Datos Enviados	360
Datos Recibidos	327
Datos Perdidos	33

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

Como se puede observar en la tabla 8-3, en un periodo de 12 horas se enviaron 360 datos donde fueron recibidos correctamente 327 y tan solo se perdieron 33 dando un resultado del 90,83% de datos recibidos con éxito, las pérdidas se pudieron producir debido a fallos en la cobertura ofrecida por el proveedor, pero se considera mínima por lo que es totalmente aceptable para el correcto funcionamiento de nuestro dispositivo.

3.4 Análisis del consumo de datos móviles

El dispositivo está programado para enviar una trama de datos cada 2 minutos, cada trama tiene un consumo de bytes, el consumo mensual se detalla en la tabla 9-3.

Tabla9-3. Consumo de datos del dispositivo

Bytes por cada envío	Numero de envíos al día	Total consumo al día	Total Consumo mensual
12500	720	9000 Kb	270 Mb

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

3.5 Análisis de costos

Para analizar el costo total de la implementación, en la tabla 9-3 se detallaron los precios de todos los equipos y materiales utilizados por el dispositivo.

Tabla 10-3: Costo de la implementación del dispositivo

EQUIPO/MATERIAL	PRECIO (USD)
Arduino MEGA 2560	45
Sensor de humedad relativa DHT11	6
Sensor de presión y temperatura BMP180	5
Módulo sensor de lluvia YL-38	4
Sensor de velocidad del viento SY-FS2	70
Sensor de dirección del viento SY-FX2	80
Módulo GPRS SIM900	38
Módulo LM2596	9
Módulo LCD 2004A	18
Panel Fotovoltaico	100
Batería Solar	190
Controlador De Carga	45
Gabinete	27
Cables	15
Porta fusibles y fusibles	10
Breaker	5
Otros Materiales	50
Estructura	400
TOTAL	1117

Realizado por: Coronel Jhonatan, Espinosa Marcos, 2019

En la tabla 10-3 se puede apreciar que el costo total del dispositivo es de 1117 USD, que a diferencia de otros equipos de características similares tienen un 20% menos de costo, por lo que se considera un porcentaje aceptable en lo invertido teniendo en cuenta que ninguno de los equipos existentes en el mercado están enfocados a la práctica del parapente, además se puede acceder a esta información desde cualquier ubicación en la que nos encontremos y con una autonomía de hasta 12 años de vida gracias a su panel solar y batería.

CONCLUSIONES

- Se implementó un dispositivo para monitorear variables meteorológicas en la pista de despegue del club parapente Chimborazo, con la capacidad de leer las variables meteorológicas del lugar y enviar dichos datos procesados a una plataforma almacenada en la nube a los cuales podemos acceder desde la aplicación móvil desarrollada específicamente para este dispositivo, además posee una autonomía energética otorgada por un sistema fotovoltaico.
- Se verificó un error absoluto máximo de 0,7 °C y de 0,08 hPa en las mediciones realizadas con el sensor BMP180 el cual tiene un error de $\pm 1^\circ\text{C}$ y $\pm 0,12\text{hPa}$, un error absoluto máximo de 3% en la medición de humedad relativa del aire con el sensor DHT11 el cual tiene una exactitud de $\pm 4\%$, y en la velocidad se obtuvo un error absoluto máximo de 0,3 Km/h con el sensor SY-FS2 que posee un error de 3%, por lo tanto el dispositivo no incorpora error a los sensores.
- Se realizó la validación del sensor de dirección del viento utilizando una aplicación de brújula en el Huawei p20 lite, obteniendo como resultado un valor de voltaje (4.70-0.31 V) para la dirección norte, (0.32-0.94 V) para noroeste, (0.95-1.56 V) para este, (1.57-2.19 V) para sureste, (2.20-2.81 V) para sur, (2.82-3.44 V) para suroeste. (3.45-4.06 V) para oeste y (4.07-4.69 V) para Noroeste.
- Se estableció un rango para la calibración del sensor detector de lluvia YL-38 con el cual se obtuvo un rango para considerar que el clima es seco (2.75 V a 4.75 V) y climas lluviosos (0V a 2.74V), para esta prueba se utilizó un rociador para crear un efecto de lluvia artificial.
- El dispositivo posee una estabilidad aceptable ya que se obtuvo un coeficiente de variación menor al 10% basándonos en la repetitividad de las mediciones obtenidas por los sensores.
- Se verificó que el 90,83% de los datos enviados a la plataforma IoT llegan a su destino correctamente y en el tiempo establecido, este resultado se entiende como aceptable para nuestro dispositivo ya que los datos perdidos no son consecutivos por lo que no es considerable en su lectura. Además, para realizar el envío se evaluó el consumo de datos móviles obteniendo un total de 270Mb mensuales.
- El costo de implementación total del dispositivo es de \$1117 obteniendo una diferencia del 20% respecto a los equipos de características similares, por lo tanto, se considera un costo aceptable teniendo en cuenta que el dispositivo implementado cuenta con características extras a los disponibles en el mercado como la aplicación móvil, la autonomía energética y el enfoque hacia la práctica del parapente.

RECOMENDACIONES

- Realizar una investigación para poder incorporar a la aplicación móvil una lectura de datos durante el vuelo consiguiendo así asistencia tanto en el despegue como durante el vuelo y el aterrizaje.
- Enviar la información obtenida a servidores del INAMHI incorporando así las mediciones a la red nacional de observación hidrometeorológica.
- Implementar una red de dispositivos de medición meteorológica en otros lugares de vuelo a nivel nacional para obtener una red de información acerca de las condiciones meteorológicas.
- Integrar un sistema de visualización adherido a unas gafas que le permitan al piloto tanto visualizar los datos como de protección ante la fuerza del viento.
- Incorporar un algoritmo para la predicción del tiempo atmosférico en días posteriores, consiguiendo así poder planificar días de vuelo seguro en la pista de despegue con mayor anticipación.

BIBLIOGRAFÍA

Adrián De La Cámara, R., *Arduino + módulo GSM/GPRS: monitorización, automatización y gestión remota en un viñedo*. [en línea]. S.l.: Universitat Oberta de Catalunya. 2017. Disponible en: <http://hdl.handle.net/10609/65345>.

Andrade Alban, E., Registro de asistencia «Club parapente Chimborazo». . Riobamba: 2019.

ARDUINO, Arduino. [en línea]. 2019. Disponible en: <https://www.arduino.cc/>.

Asociación Ecuatoriana de Pilotos y Profesionales de Parapente, Asociación Ecuatoriana de Pilotos y Profesionales de Parapente. [en línea]. 2019. [Consulta: 5 febrero 2019]. Disponible en: <http://www.aeparapente.org/>.

Ayllón, T., *Elementos de Meteorología y Climatología*. segunda. Mexico D.F.: Trillas. 2013. ISBN 978-607-17-1561-6.

Banzi, M., Barragin, H., Cuartielles, D., Todd, T.I. Y Mellis, K.D., Arduino: Manual de Programación 2 Datos del documento original. [en línea]. 2007. [Consulta: 18 febrero 2019]. Disponible en: <http://www.arduino.cc/en/Booklet/HomePage> <http://www.wiring.org.co> <http://www.arduino.cc/en/Booklet/HomePage> <http://cslibrary.stanford.edu/101/http://creativecommons.org/licenses/by-nc-/>.

Benítez, M.S. Y Armoa, J., Estación Meteorológica. [en línea]. 2016. [Consulta: 4 junio 2018]. Disponible en: <http://www.pol.una.py/?q=node/165>.

Caluña Tisalema, A.M. Y Jordán Vargas, A.R., *Diseño E Implementación De Un Sistema De Monitoreo Climatológico vía Gprs Y Automatización De La Calibración Del Anillo Del Piranómetro En La Estación Meteorológica - Epoch* [en línea]. S.l.: ESPOCH. 2017. Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/6880/1/108T0199.pdf>.

Coba Robalino, J.L. Y Hernández Angulo, B.N., *IMPLEMENTACIÓN DE UN SISTEMA AUTÓNOMO INALÁMBRICO PARA LA ALIMENTACIÓN DE TILAPIA, BASADO EN SUPERVISIÓN DE VARIABLES CLIMÁTICAS Y CONTROL DE CALIDAD DE AGUA* [en línea]. S.l.: Escuela Superior Politécnica de Chimborazo. 2018. [Consulta: 18 febrero 2019]. Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/9245/1/108T0266.pdf>.

ELECTRONILAB, Módulo LM2596 Convertidor de Voltaje DC-DC Buck 1.25V-35V - Electronilab. [en línea]. 2019a. [Consulta: 18 febrero 2019]. Disponible en: <https://electronilab.co/tienda/modulo-lm2596-convertidor-de-voltaje-dc-dc-buck-1-25v->

35v/.

ELECTRONILAB, Sensor de Presión Barométrica - BMP180 - Electronilab. [en línea]. 2019b. [Consulta: 18 febrero 2019]. Disponible en: <https://electronilab.co/tienda/sensor-de-presion-barometrica-bmp180/>.

ELECTRONILAB, Sensor de Temperatura y Humedad DHT11 - Electronilab. [en línea]. 2019c. [Consulta: 18 febrero 2019]. Disponible en: <https://electronilab.co/tienda/sensor-de-temperatura-y-humedad-dht11/>.

ENERGY, E.C.O.G., Datasheet EGE-110P-36 Polycrystalline Photovoltaic module. , 2019.

FREEMETEO.EC, freemeteo.ec. [en línea]. 2019. Disponible en: <https://freemeteo.ec/eltiempo/?language=spanishar&country=ecuador>.

JMINDUSTRIAL, Diferencia entre los termopares y los RTD's. [en línea]. 2019. [Consulta: 18 febrero 2019]. Disponible en: https://www.jmi.com.mx/documento_literatura/Diferencia-termopares-RTD.pdf.

Kouro, S., *Automatización industrial: Sensores De Humedad* [en línea]. S.l.: Universidad Tecnica Federico Santa Maria. 2001. Disponible en: <https://docplayer.es/4600549-Automatizacion-industrial-sensores-de-humedad.html>.

Lajara Vizcaino, J.R., *Sistemas integrados con Arduino*. primera. Mexico D.F.: Alfaomega. 2014. ISBN 978-607-622-046-7.

Llamas, L., DETECTOR DE LLUVIA CON ARDUINO Y SENSOR FC-37 O YL-83. [en línea]. 2016. Disponible en: <https://www.luisllamas.es/arduino-lluvia/>.

Llamosa, L.E., Meza Contreras, L.G. Y Botero, M., ESTUDIO DE REPETIBILIDAD Y REPRODUCIBILIDAD UTILIZANDO EL MÉTODO DE PROMEDIOS Y RANGOS PARA EL ASEGURAMIENTO DE LA CALIDAD DE LOS RESULTADOS DE CALIBRACIÓN DE ACUERDO CON LA NORMA TÉCNICA NTC-ISO/IEC 17025. *Scientia et Technica Año XIII* [en línea], 2007. [Consulta: 19 enero 2019]. Disponible en: <http://revistas.utp.edu.co/index.php/revistaciencia/article/viewFile/5479/2817>.

Loureiro Garrido, R., Estudio Plataformas IoT. [en línea]. 2015. [Consulta: 5 febrero 2019]. Disponible en: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/42812/6/rloureiroTFC0615memoria.pdf>.

Martín Jimenez, J., *Sistemas solares fotovoltaicos: fundamentos, tecnologías y aplicaciones*. sexta. Barcelona-España: Reverté. 2010. ISBN 978-84-291-4424-6.

- Méndez Muñiz, J.M.**, *Energía solar fotovoltaica*. séptima. Madrid: Fundación Confemetal. 2012. ISBN 978-84-92735-77-8.
- MORNINGSTAR**, Datasheet controlador solar. , 2019.
- Moro Vallina, M.**, *Instalaciones Solares Fotovoltáicas*. Madrid: Paraninfo. 2010. ISBN 978-84-9732-776-3.
- Nolasco Valenzuela, J.S.**, *Desarrollo de aplicaciones móviles con Android*. 1. Lima-Peru: Macro EIRL. 2013. ISBN 978-612-304-106-9.
- OMNIBLUG**, Sensor de temperatura y humedad DHT11 - DHT22. [en línea]. 2018. [Consulta: 21 junio 2018]. Disponible en: <http://www.omniblug.com/sensor-temperatura-humedad-DHT11-DHT22.html>.
- PCE, I., PCE INSTRUMENTS**. [en línea]. 2019. [Consulta: 18 febrero 2019]. Disponible en: <https://www.pce-instruments.com/>.
- Perpiñan Lamigueiro, O.**, *Energía solar fotovoltaica* [en línea]. 1.9. S.l.: creative commons. 2018. Disponible en: <https://github.com/oscarperpinan/esf%0AEsta>.
- POWERTIP**, Datasheet LCD 2004-a. , 2019.
- Prieto Donate, F.**, Protocolo GPRS.pdf. [en línea]. Sevilla: 2007. Disponible en: <http://bibing.us.es/proyectos/abreproy/11372/fichero/Memoria/03+-+GPRS.pdf>.
- RASPBERRY, P.**, Raspberry Pi — Teach, Learn, and Make with Raspberry Pi. [en línea]. 2019. [Consulta: 18 febrero 2019]. Disponible en: <https://www.raspberrypi.org/>.
- Robledo Fernández, D.**, Desarrollo De Aplicaciones Para Android. [en línea]. 2014. Disponible en: https://books.google.com.ec/books?id=1wLXAWAAQBAJ&pg=PT135&dq=desarrollo+de+aplicaciones++para+android+II&hl=es&sa=X&redir_esc=y#v=onepage&q=desarrollo+de+aplicaciones+para+android+II&f=false.
- Saez, G.A.**, *Manual del parapentista* [en línea]. 5.4. Buenos aires argentina: s.n. 2016. Disponible en: <http://www.facl.es/sites/default/files/documentos/Librodigitaldemanualdeparapente.pdf>.
- Santillán Rivadeneira, M.J. Y Maldonado Jaramillo, D.A.**, *Desarrollo de una aplicación móvil para la estimulación cognitiva de adultos mayores que padecen alzheimer en fases ligera y moderada utilizando la plataforma android studio* [en línea]. S.l.: ESPOCH. 2017. Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/7369/1/18T00699.pdf>.

- Sensor, H.F.**, Pressure Sensor, Pressure Transmitter, Level Sensor-Hunan Firstrate Sensor Co., Ltd. [en línea]. 2019. Disponible en: <http://www.firstsensor.com.cn/>.
- SIMCOM**, SIM900 the GSM/GPRS Module for M2M Applications. *GSM / GPRS Module*, 2013. p. 2. DOI 10.1007/s11207-006-0068-7.
- SIMCOM**, SIM808_Hardware Design_V1.00. [en línea]. 2019. Disponible en: https://cdn-shop.adafruit.com/datasheets/SIM808_Hardware+Design_V1.00.pdf.
- Soriano, R., Valarezo, S. Y Hernández, R.**, El parapente: equipo básico y consejos de seguridad de un deporte de aventura. [en línea]. 2016. [Consulta: 4 junio 2018]. Disponible en: <https://guiafitness.com/el-parapente.html>.
- TESSEL**, Tessel. [en línea]. 2019. [Consulta: 18 febrero 2019]. Disponible en: <https://tessel.io/>.
- Tobajas Vásquez, M.C.**, *Instalaciones Solares Fotovoltaicas*. segunda. Bogota-Colombia: Ediciones de la U. 2015. ISBN 978-958-762-267-6.
- Torres, S.**, EL ABC DE LA AUTOMATIZACION: SENSORES DE PRESIÓN. [en línea]. 2018. [Consulta: 18 febrero 2019]. Disponible en: <http://www.aie.cl/files/file/comites/ca/abc/sensores-de-presion.pdf>.
- Trading, E.K.**, CH WIND DIRECTION SENSOR Model : SY-FX2. [en línea]. 2019. [Consulta: 4 febrero 2019]. Disponible en: http://www.ekt2.com/pdf/412_CH_WIND_DIRECTION_SENSOR.pdf.
- Trading, E.K.**, Wind Sensor (Pulse -T ype) Model : JL -FS2. , 2019. p. 1-2.
- ULTRACELL**, Datasheet Bateria Ucg 55-12. , 2019. p. 1-2.
- Villa Yanez, M.**, *Desarrollo De Una Aplicación Que Permita El Escaneo De Las Vulnerabilidades En Los Dispositivos Móviles Android Para Mitigar Los Problemas De Seguridad* [en línea]. S.l.: Escuela Superior Politecnica de Chimborazo. 2016. Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/4599/1/20T00677.pdf>.
- Viloria Núñez, C., Peña, J.C. Y Garzón, C.L.**, Análisis comparativo de tecnologías inalámbricas para una solución de servicios de telemedicina. *Ingeniería & Desarrollo. Universidad del Norte.*, 2009. p. 200-217. ISSN 0122-3461.

ANEXOS

ANEXO A: Datasheet de Arduino MEGA 2560

Arduino Mega2560 Rev3



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

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

Additional features coming with the R3 version are:

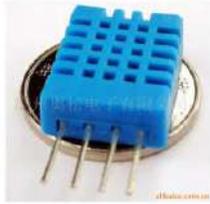
- ATmega16U2 instead 8U2 as USB-to-Serial converter.
- 1.0 pinout: added SDA and SCL pins for TWI communication placed near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board and the second one is a not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

Technical Specifications

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

ANEXO B: Datasheet del sensor DHT11

Digital-output relative humidity & temperature sensor/module – DHT11



Resistive-type humidity and temperature module/sensor

1. Feature & Application:

* Full range temperature compensated * Relative humidity and temperature measurement
 * Calibrated digital signal * Outstanding long-term stability * Extra components not needed
 * Long transmission distance * Low power consumption * 4 pins packaged and fully interchangeable

2. Description:

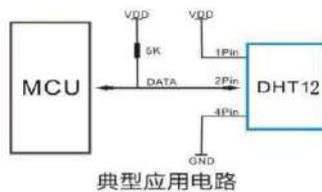
DHT11 output calibrated digital signal. It utilizes exclusive digital-signal-collecting-technique and humidity sensing technology, assuring its reliability and stability. Its sensing elements is connected with 8-bit single-chip computer.

Every sensor of this model is temperature compensated and calibrated in accurate calibration chamber and the calibration-coefficient is saved in OTP memory.

Small size & low consumption & long transmission distance(20m) enable DHT11 to be suited in all kinds of harsh application occasions. Single-row packaged with four pins, making the connection very convenient.

1

5. Typical application



3Pin=NULL, MCU=Microcomputer or single-chip computer

6. Operating specifications:

(1) Power and Pins

Power's voltage should be 3-5.5V DC. When power is supplied to sensor, don't send any instruction to the sensor within one second to pass unstable status. One capacitor valued 100nF can be added between VDD and GND for power filtering.

(2) Communication and signal

Single-bus data is used for communication between MCU and DHT11.

7. Electrical Characteristics:

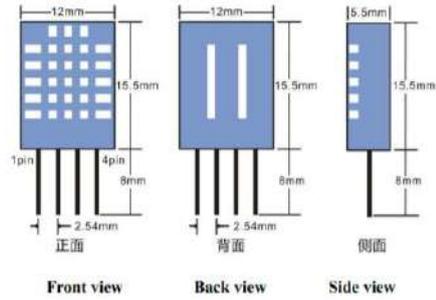
Item	Condition	Min	Typical	Max	Unit
Power supply	DC	3	5	5.5	V
Current supply	Measuring	0.5		2.5	mA
	Stand-by	100	Null	150	uA
	Average	0.2	Null	1	mA

3

3. Technical Specification:

Model	DH111	
Power supply	3-5.5V DC	
Output signal	digital signal via single-bus	
Sensing element	Polymer resistor	
Measuring range	humidity 20-90%RH;	temperature 0-50 Celsius
Accuracy	humidity $\pm 4\%$ RH (Max $\pm 5\%$ RH);	temperature ± 2.0 Celsius
Resolution or sensitivity	humidity 1%RH;	temperature 0.1 Celsius
Repeatability	humidity $\pm 1\%$ RH;	temperature ± 1 Celsius
Humidity hysteresis	$\pm 1\%$ RH	
Long-term Stability	$\pm 0.5\%$ RH/year	
Sensing period	Average: 2s	
Interchangeability	fully interchangeable	
Dimensions	size 12*15.5*5.5mm	

4. Dimensions: (unit---mm)



ANEXO C: Datasheet del sensor BMP180

BMP180

Digital, barometric pressure sensor

Bosch Sensortec



BOSCH
Invented for life

General description

The BMP180 is the new digital barometric pressure sensor of Bosch Sensortec, with a very high performance, which enables applications in advanced mobile devices, such as smart phones, tablet PCs and sports devices. It follows the BMP085 and brings many improvements, like the smaller size and the expansion of digital interfaces.

The ultra-low power consumption down to 3µA makes the BMP180 the leader in power saving for your mobile devices. BMP180 is also distinguished by its very stable behavior (performance) with regard to the independency of the supply voltage.

Technology and specification

Through its high relative accuracy of ±0.12 hPa (±1m) the BMP180 has become the most reliable sensor for precise applications, like indoor-navigation. The small size of 3.6 x 3.8mm² and the height of only 0.93mm makes it very suitable for the implementation in small mobile devices. The high absolute accuracy (please see parameter sheet beside) and a noise level down to 0.02 hPa (altitude of 17 cm) open new perspectives for applications in the sport devices.

The BMP180 is a sensor based on piezo-resistive MEMS technology for EMC robustness and high quality standards. The dies of the BMP180 are protected by a stable and thin LGA package with a metal lid. The package has seven optimized pins. The BMP180 can communicate directly with a microcontroller in the device through I²C or SPI as a variant.

BMP180 target applications

- ▶ Indoor navigation
- ▶ GPS-enhancement for dead-reckoning, slope detection, etc.
- ▶ Sport devices, e.g. altitude profile

- ▶ Weather forecast
- ▶ Vertical velocity indication (rise/sink speed)

Technical Data	BMP180
Pressure range	300 ... 1100 hPa
RMS noise expressed in pressure	0.06 hPa, typ. (ultra low power mode)
	0.02 hPa, typ. (ultra high resolution mode)
RMS noise expressed in altitude	0.06 hPa, typ. (ultra low power mode)
	0.02 hPa, typ. (ultra high resolution mode)
Relative accuracy pressure: V _{DD} = 3.3 V	950 ... 1050 hPa @ 25 °C ±0.12 hPa ±1.0 m
	700 ... 900 hPa 25 ... 40 °C ±0.12 hPa ±1.0 m
Absolute accuracy p=300 ... 1100hPa (T=0 ... +65°C,	Pressure: -4.0 ... +2.0hPa Temperature: -1 hPa (+/- 1 hPa), typ.
Average current consumption (1Hz data refresh rate)	3 µA, typ. (ultra-low power mode)
	32 µA, typ. (advanced mode)
Peak current	650 µA, typ.
Stand-by current	0.1 µA, typ.
Supply voltage V _{DDIO}	1.62 ... 3.6 V
Supply voltage V _{DD}	1.8 ... 3.6 V
Operation temp. range full accuracy	-40 ... +85 °C
	0 ... +65 °C
Pressure conv. time	-5 msec, typ. (std. mode)
I ² C data transfer rate	3.4 MHz, max.
Package type / pin no.	LGA / 7
Package dimensions	3.6 x 3.8 x 0.93 mm ³

ANEXO D: Datasheet del módulo sensor YL-38

RAIN SENSOR MODULE



Description

The rain sensor module is an easy tool for rain detection. It can be used as a switch when raindrop falls through the raining board and also for measuring rainfall intensity. The module features, a rain board and the control board that is separate for more convenience, power indicator LED and an adjustable sensitivity through a potentiometer.

The analog output is used in detection of drops in the amount of rainfall. Connected to 5V power supply, the LED will turn on when induction board has no rain drop, and DO output is high. When dropping a little amount water, DO output is low, the

switch indicator will turn on. Brush off the water droplets, and when restored to the initial state, outputs high level.

Specifications

- Adopts high quality of RF-04 double sided material.
- Area: 5cm x 4cm nickel plate on side,
- Anti-oxidation, anti-conductivity, with long use time;
- Comparator output signal clean waveform is good, driving ability, over 15mA;
- Potentiometer adjust the sensitivity;
- Working voltage 5V;
- Output format: Digital switching output (0 and 1) and analog voltage output AO;
- With bolt holes for easy installation;
- Small board PCB size: 3.2cm x 1.4cm;
- Uses a wide voltage LM393 comparator

Pin Configuration



1. **VCC**: 5V DC
2. **GND**: ground
3. **DO**: high/low output
4. **AO**: analog output

ANEXO E: Datasheet del sensor de velocidad del viento SY-FS2

Wind Sensor (Pulse - Type)
Model: JL - FS2



Features:
Wind sensor made of aluminum alloy material, the use of special precision die casting process, dimensional tolerances very small surface accuracy is very high, both through internal circuit protective treatment, the entire sensor with high strength, weather resistance, anti-corrosion and water resistance. Plug the cable connector for the military, has a good anti-corrosion, anti-corrosion properties, to ensure the long-term use of the instrument, in conjunction with internal bearings system to ensure the accuracy of the wind speed acquisition.
Circuit PCB using military grade A grade materials to ensure the quality and stability and electrical performance parameters; electronic components are imported industrial-grade chips, making the whole has a very reliable anti-electromagnetic interference capability, to ensure that the host -20C ~ 60C, humidity can work properly within 10% -95% range.
Wind sensor compact, flange base, carrying, convenient installation, beautiful appearance, high accuracy, wide range, good stability, low power consumption, data of good, long distance signal transmission, anti-interference ability The signal output forms, light aluminum alloy material quality, high strength.

Wiring:
Three-wire:
Red Line Supply positive
Black line Power negative
Yellow Line Signal line

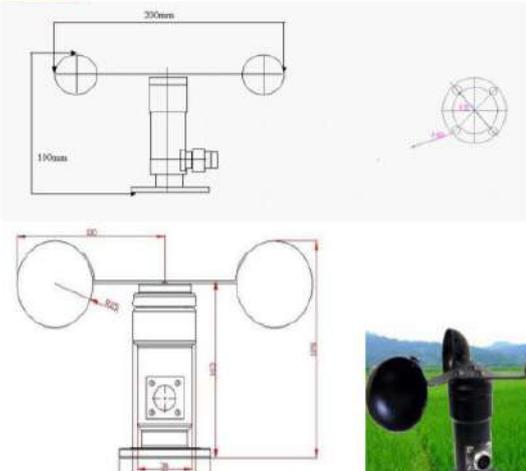
Specifications:

Type	Cable and outlet	
		Model
	Name	Universal wind sensor
Sensor	Signal output	Pulse signal output
	Sensor Style	Three cups of formula
	Start wind speed	0.4-0.8 m/s
	Resolution	0.1m/s
	Effective wind speed measurement range	0-30m/s or 0-60 m/s



General parameters	System error	± 3%
	Transmission distance	Greater than 1000m
	Transmission medium	Cable transmission
	Wiring	Three-wire
	Operating Temperature	-40 °C ~ 80 °C
	Supply Voltage	DC12-24V can be generic
	Power Consumption	MAX ≤ 0.2W
	Size	See size chart
	Weight	<1Kg

Dimensions:




Scope:
Can be widely used in greenhouses, environmental protection, weather stations, ships, docks, heavy machinery cranes, ports, terminals, cable cars, no need to measure wind speed and direction in place.



ANEXO F: Datasheet del sensor de velocidad del viento SY-FX2

CH WIND DIRECTION SENSOR
Model:SY-FX2



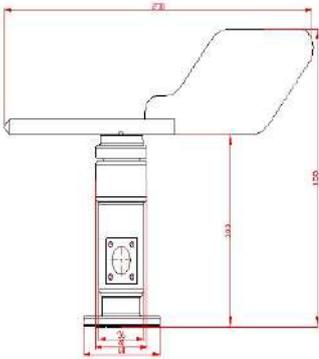
Description:
Wind direction sensor using aluminum alloy material, the use of special precision die-casting process dimensional tolerances very small surface accuracy is very high, the internal circuit have been protective treatment, the entire sensor has a high strength, weather resistance, anti-corrosion and water resistance.
Cable Connector To Military plug, Has a good anti-corrosion, anti-corrosion properties, long-term use can ensure that the instrument, in conjunction with internal bearings system to ensure the accuracy of the wind speed acquisition.
Circuit PCB using military grade A grade materials to ensure the quality and stability and electrical performance parameters; electronic components are imported industrial grade chip, making the overall with a very reliable anti-electromagnetic interference capability, to ensure that the host at -20° ~ 60°, can work within the normal range of 10% ~ 95% humidity.
Compact wind direction sensor, flange base, carry, quick and easy installation, attractive appearance, high accuracy, wide range, stable performance, low power consumption, the data information of good, long distance signal transmission, anti-interference ability Signal output forms, light aluminum alloy material quality, high strength.

Technical Parameters:

- Power: Supply Voltage: DC12-24V
- Signal Output: Voltage: 0-5v Current: 4-20mA
- Sensor Style: fin-type
- Start Wind Speed: 0.5 m/s
- Resolution: 0.1m/s
- System Error: $\pm 3\%$
- Transmission Distance: more than 1000m
- Wiring: Voltage: three-wire, Current: Three-wire
- Operating Temperature: -20°C ~ 80°C
- Power Consumption: Voltage- MAX $\leq 0.3W$, Current- MAX $\leq 0.7W$, Numeric MAX $\leq 0.3W$
- Weight: <1kg

EKT
ELECTRONICS KANSAS (INDIA) PVT. LTD.

Scope:
Can be widely used in greenhouses, environmental protection, weather stations, ships, dock, Heavy machinery, cranes, harbor, wharf, cable cars, no need to measure wind speed and direction in place.



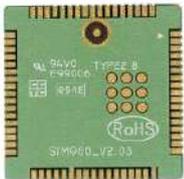
Made in china

EKT
ELECTRONICS KANSAS (INDIA) PVT. LTD.

ANEXO G: Datasheet del módulo GPRS SIM900

SIM900
The GSM/GPRS Module for M2M applications

SIM900 GSM/GPRS Module

The SIM900 is a complete Quad-band GSM/GPRS solution in a SMT module which can be embedded in the customer applications.

Featuring an industry-standard interface, the SIM900 delivers GSM/GPRS 850/900/1800/1900MHz performance for voice, SMS, Data, and Fax in a small form factor and with low power consumption. With a tiny configuration of 24mm x 24mm x 3 mm, SIM900 can fit almost all the space requirements in your M2M application, especially for slim and compact demand of design.

- SIM900 is designed with a very powerful single-chip processor integrating AMR326EJ-S core
- Quad - band GSM/GPRS module with a size of 24mmx24mmx3mm
- SMT type suit for customer application
- An embedded Powerful TCP/IP protocol stack
- Based upon mature and field-proven platform, backed up by our support service, from definition to design and production

SIM900
The GSM/GPRS Module for M2M applications

General features

- Quad-Band 850/ 900/ 1800/ 1900 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 22+
 - Class 1 (2 W @ 850/ 900 MHz)
 - Class 1 (1 W @ 1800/1900MHz)
- Dimensions: 24* 24 * 3 mm
- Weight: 3.4g
- Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
- SIM application toolkit
- Supply voltage range 3.4 ... 4.5 V
- Low power consumption
- Operation temperature: -30 °C to +60 °C

Specifications for fax

- Group 3, class 1

Specifications for data

- GPRS class 10; max. 85.6 kbps (downlink)
- PBCCH support
- Coding schemes CS 1, 2, 3, 4
- CSD up to 14.4 kbps
- USSD
- Non transparent mode
- PPP-stack

Specifications for SMS via GSM Pin Assignment

/ GPRS

- Point-to-point MO and MT
- SMS cell broadcast
- Text and PDU mode

Drivers

- MUX Driver

Specifications for voice

- Tricodec
 - Half rate (HR)
 - Full rate (FR)
 - Enhanced Full rate (EFR)

- Hands-free operation (Echo suppression)
- AMR, Half Rate(HR) Full Rate(FR)

Interfaces

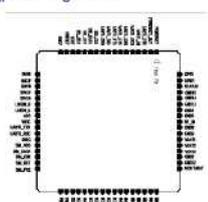
- Interface to external SIM 3V/ 1.6V
- analog audio interface
- RTC backup
- SPI interface
- Serial interface
- Antenna pad
- I2C
- GPIO
- PWM
- ADC

Compatibility

- AT cellular command interface

Approvals (in planning)

- CE
- FCC
- ROHS
- PTCRB
- GCF
- AT&T
- IC
- TA



More about SIM900 module, Please contact: Tel:+86 21 32523300
 Fax:+86 21 32523301
 Email:simcom@sim.com

ANEXO H: Datasheet del regulador de voltaje LM2596



LM2596

SNVS124D – NOVEMBER 1999 – REVISED MAY 2016

LM2596 SIMPLE SWITCHER® Power Converter 150-kHz 3-A Step-Down Voltage Regulator

1 Features

- 3.3-V, 5-V, 12-V, and Adjustable Output Versions
- Adjustable Version Output Voltage Range: 1.2-V to 37-V \pm 4% Maximum Over Line and Load Conditions
- Available in TO-220 and TO-263 Packages
- 3-A Output Load Current
- Input Voltage Range Up to 40 V
- Requires Only 4 External Components
- Excellent Line and Load Regulation Specifications
- 150-kHz Fixed-Frequency Internal Oscillator
- TTL Shutdown Capability
- Low Power Standby Mode, I_Q , Typically 80 μ A
- High Efficiency
- Uses Readily Available Standard Inductors
- Thermal Shutdown and Current-Limit Protection
- Create a Custom Design Using the LM2596 with the [WEBENCH Power Designer](#)

2 Applications

- Simple High-Efficiency Step-Down (Buck) Regulator
- On-Card Switching Regulators
- Positive to Negative Converter

3 Description

The LM2596 series of regulators are monolithic integrated circuits that provide all the active functions for a step-down (buck) switching regulator, capable of driving a 3-A load with excellent line and load regulation. These devices are available in fixed output voltages of 3.3 V, 5 V, 12 V, and an adjustable output version.

Requiring a minimum number of external components, these regulators are simple to use and include internal frequency compensation, and a fixed-frequency oscillator.

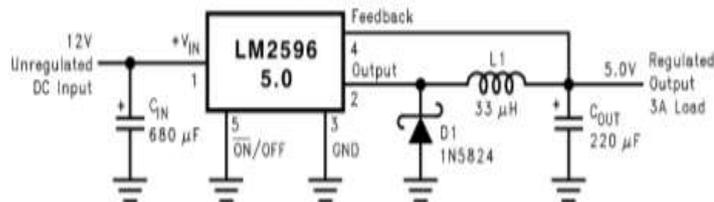
The LM2596 series operates at a switching frequency of 150 kHz, thus allowing smaller sized filter components than what would be required with lower frequency switching regulators. Available in a standard 7-pin TO-220 package with several different lead bend options, and a 7-pin TO-263 surface mount package.

Device Information⁽¹⁾

PART NUMBER	PACKAGE	BODY SIZE (NOM)
LM2596	TO-220 (7)	14.986 mm \times 10.16 mm
	TO-263 (7)	10.10 mm \times 8.89 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.

Typical Application



Copyright © 2016, Texas Instruments Incorporated

(Fixed Output Voltage Versions)

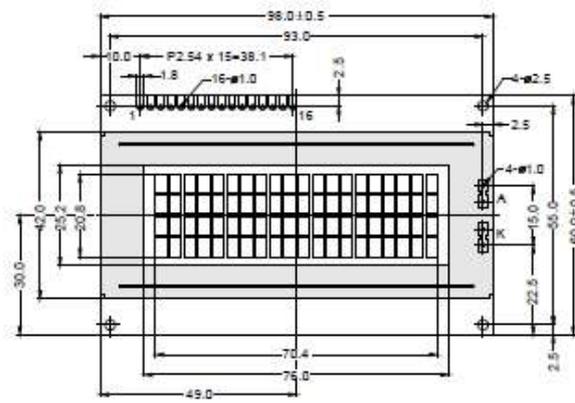
ANEXO I: Datasheet del modulo LCD 2004A

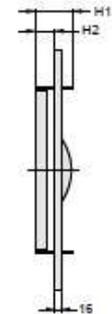


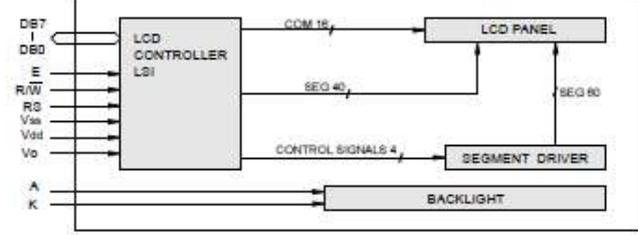
PC 2004-A

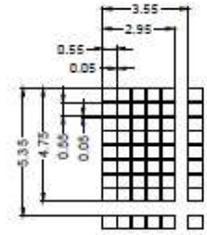


..... OUTLINE DIMENSION & BLOCK DIAGRAM









The tolerance unless classified $\pm 0.3\text{mm}$

MECHANICAL SPECIFICATION			
Overall Size	98.0 x 60.0	Module	H2 / H1
View Area	76.0 x 25.2	W / O B/L	5.0 / 9.6
Dot Size	0.55 x 0.55	EL B/L	5.0 / 9.6
Dot Pitch	0.60 x 0.60	LED B/L	8.7 / 13.3

PIN ASSIGNMENT		
Pin no.	Symbol	Function
1	Vss	Power supply(GND)
2	Vdd	Power supply(+)
3	Vo	Contrast Adjust
4	RS	Register select signal
5	R/W	Data read / write
6	E	Enable signal
7	DB0	Data bus line
8	DB1	Data bus line
9	DB2	Data bus line
10	DB3	Data bus line
11	DB4	Data bus line
12	DB5	Data bus line
13	DB6	Data bus line
14	DB7	Data bus line
15	A	Power supply for LED B/L (+)
16	K	Power supply for LED B/L (-)

ABSOLUTE MAXIMUM RATING					
Item	Symbol	Condition	Min.	Max.	Units
Supply for logic voltage	Vdd-Vss	25°C	-0.3	7	V
LCD driving supply voltage	Vdd-Vee	25°C	-0.3	13	V
Input voltage	Vin	25°C	-0.3	Vdd+0.3	V

ELECTRICAL CHARACTERISTICS									
Item	Symbol	Condition	Min.	Typical	Max.	Units			
Power supply voltage	Vdd-vss	25°C	2.7	—	5.5	V			
LCD operation voltage	Vop	Top	N	W	N	W	N	W	V
		-20°C	—	7.1	—	7.5	—	7.9	V
		0°C	5.1	—	5.4	—	6.1	—	V
		25°C	5	6.1	5.3	6.4	5.8	6.7	V
		50°C	4.4	—	4.7	—	5.5	—	V
		70°C	—	5.7	—	6	—	6.3	V
LCM current consumption (No B/L)	Idd	Vdd=5V	—	2.5	4	mA			
Backlight current consumption	LED/edge	VB/L=4.2V	—	—	—	mA			
	LED/array	VB/L=4.2V	—	260	—	mA			

REMARK

LCD option: STN, TN, FSTN
 Backlight Option: LED, EL Backlight feature, other Specs not available on catalog is under request.

ANEXO J: Datasheet del panel fotovoltaico

EGE ECO GREEN ENERGY

**EGE-110P-36
POLYCRYSTALLINE
PHOTOVOLTAIC MODULE**



- High efficiency** 4 busbars cells, with efficiency up to 18.8 %.
- Secure weather** IP65 junction box for long term weather endurance.
- Quality frame** High quality aluminum frame, resisting load up to 5400 Pa and wind pressure up to 2400 Pa.
- Smart glass** High transmissivity, low-iron tempered glass.

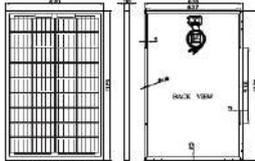
10 Years Product warranty

25 Years Output power warranty

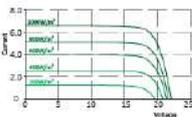
www.eco-greenenergy.com

EGE ECO GREEN ENERGY

Engineering drawings



IV curves



Electrical characteristics at STC*		General characteristics	
Power output (Pmax)	110 W	Cell type	Polycrystalline (156 × 157 mm)
Power output tolerance	±3 %	Number of cells	36
Panel efficiency	18.64 %	Dimensions	1120 × 670 × 30 mm
Maximum power voltage (Vmp)	18.49 V	Weight	8.4 kg
Maximum power current (Imp)	5.95 A	Glass	3.2 mm tempered glass
Open circuit voltage (Voc)	22.58 V	Frame	Anodized aluminium alloy
Short circuit current (Isc)	6.48 A	Junction box	IP65
*Standard test conditions: Irradiance: 1000 W/m² • cell temperature 25 °C • AM 1.5		Cable	0.5 m
		Connector	MC4 or MC4 compatible

Electrical characteristics at NOCT*		Packaging	
Power output (Pmax)	81 W	Per box	2 pcs
Maximum power voltage (Vmp)	17.08 V	Per 20ft GP container	940 pcs
Maximum power current (Imp)	4.76 A	Per 40ft GP container	1900 pcs
Open circuit voltage (Voc)	20.96 V	Per 40ft HQ container	2200 pcs
Short circuit current (Isc)	5.24 A		
*Nominal operating cell temperature: irradiance: 800 W/m² • ambient temperature: 20 °C • AM 1.5 • WIND SPEED: 1 m/s			

Operating conditions		Certification	
Operating temperature range	-45 °C → +85 °C		
Max system voltage	1000 V		
Max series fuse rating	20 A		
Max front load (ac snow)	5400 Pa		
Max back load (ac wind)	2400 Pa		

www.eco-greenenergy.com

ANEXO K: Datasheet de la batería

UCG 55-12
12V 55AH
Deep Cycle

Ultracell®
Quality in Every Language

UCG55-12



Physical Specification

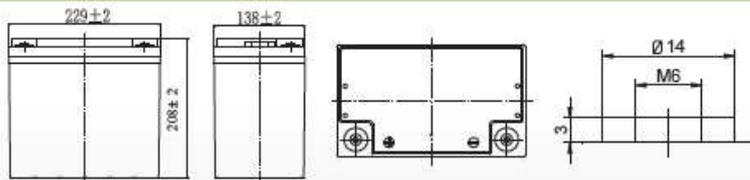
Part Number:	UCG55-12
Length:	229 ± 2 mm (9.02 inches)
Width:	138 ± 2 mm (5.43 inches)
Container Height:	208 ± 2 mm (8.2 inches)
Total Height (with terminal):	208 ± 2 mm (8.2 inches)
Approx Weight:	Approx 17kg (37.4lbs)

Specifications

	Normal Voltage	12V
	Normal Capacity (20HR)	55.0AH
Terminal Type	Standard Terminal	F9
	Optional Terminal	-
Container Material	Standard Option	ABS
	Flame Retardant Option (FR)	ABS(UL94-V0)
Rated Capacity	55AH	(20hr, 1.80V/cell, 25°C / 77°F)
	42.9AH	(5hr, 1.75V/cell, 25°C / 77°F)
	32.0 AH	(1hr, 1.60V/cell, 25°C / 77°F)
Max Discharge Current	550A (5 secs)	
Internal Resistance	Fully Charged at 25°C	Approx. 7m Ω
Discharge Characteristics	Operating Temp. Range	Discharge: -20 ~ 55°C (5 ~ 131°F)
		Charge: -10 ~ 55°C (14 ~ 131°F)
		Storage: -20 ~ 55°C (-4 ~ 131°F)
	Nominal Operating Temp. Range	25 ± 3°C (77 ± 5°F)
	Cycle Use	Initial Charging Current less than - Voltage 14.10V ~ 14.40V at 25°C (77°F) Temp. Coefficient -30mV/°C
	Standby Use	No limit on initial Charging Current Voltage 13.50V ~ 13.80V at 25°C (77°F) Temp. Coefficient -18mV/°C
Capacity affected by Temperature	40°C (104°F) 103%	
	25°C (77°F) 100%	
	0°C (32°F) 88%	
Design Floating Life at 20°C	12 Years	
Self Discharge	Ultracell batteries may be stored for up to 6 months at 25°C (77°F) and then a refresh charge is required. For higher temperatures the time interval will be shorter.	

Dimensions

F9 Terminal



ALL DATA IS SUBJECT TO CHANGE WITHOUT NOTICE

ISO9001 VdS CE R1

1

Ultracell (UK) Ltd | Vestry Business Park | Vestry Road | Liverpool | L30 1NY | United Kingdom
Tel: +44 (0) 151 523 2777 Fax: +44 (0) 151 523 0855 Email: info@ultracell.co.uk
www.ultracell.co.uk

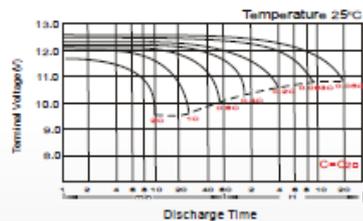
Constant Current Discharge (Amperes) at 25°C (77°F)

F.V/TIME	15min	30min	60min	2h	3h	4h	5h	6h	8h	10h	20h	48h	120h
1.60V/cell	74.3	51.3	32.0	18.7	13.2	10.4	8.69	7.43	5.78	5.15	2.81	---	---
1.65V/cell	72.6	50.2	31.5	18.6	13.1	10.3	8.64	7.37	5.72	5.14	2.81	---	---
1.70V/cell	69.9	48.7	30.7	18.4	13.0	10.2	8.58	7.32	5.72	5.13	2.80	1.23	---
1.75V/cell	67.7	47.5	30.1	18.2	12.9	10.2	8.53	7.26	5.67	5.09	2.75	1.22	0.51
1.80V/cell	63.8	45.8	29.2	17.7	12.5	9.85	8.25	7.04	5.49	5.06	2.75	1.22	0.51

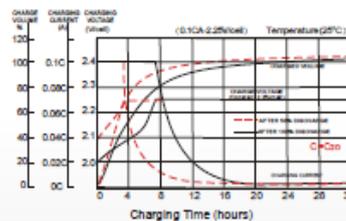
Constant Power Discharge (Watts) at 25°C (77°F)

F.V/TIME	15min	30min	60min	2h	3h	4h	5h	6h	8h	10h	20h	48h	120h
1.60V/cell	829	584	369	218	156	123	103	88.0	68.8	61.6	33.9	---	---
1.65V/cell	809	573	363	216	155	122	103	87.5	68.8	61.6	33.8	---	---
1.70V/cell	779	555	354	215	154	121	102	86.9	68.2	61.4	33.5	14.7	---
1.75V/cell	752	542	347	211	153	120	101	86.4	67.7	61.1	33.2	14.6	6.11
1.80V/cell	713	522	336	206	148	117	98.5	83.6	65.5	60.5	33.3	14.6	6.05

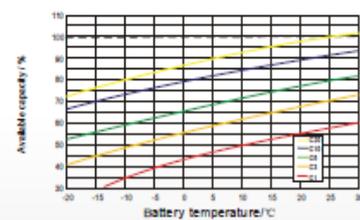
Discharge Characteristics



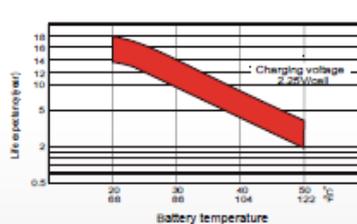
Float Charging Characteristics



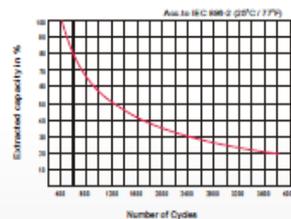
Temperature Effects in Relation to Battery Capacity



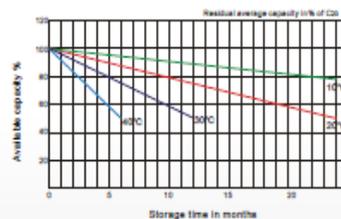
Effect of Temperature on Long Term Float Life



Cycle Life in Relation to Depth of Discharge



General Relation of Capacity VS. Storage Time



ALL DATA IS SUBJECT TO CHANGE WITHOUT NOTICE

ANEXO L: Datasheet del controlador de carga




SHS™ PARA SISTEMAS HOGAREÑOS DE ENERGÍA SOLAR DE HASTA 170 VATIOS

La necesidad alta calidad y confiabilidad de los equipos Morningstar ahora están disponibles en un controlador solar de bajo costo. El **Controlador SHS** es ideal para sistemas de electrificación rural con uno a tres paneles solares. Este controlador cumple con las especificaciones del Banco Mundial y proporciona muchas prestaciones y beneficios.





Beneficios de un Controlador Solar

Es importante incluir un controlador en un sistema de energía solar.

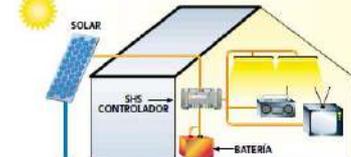
- Reduce el costo de reemplazo de baterías**
La desconexión por bajo voltaje prolonga la vida útil de la batería. Las cargas son desconectadas automáticamente cuando la batería está baja y son reconectadas después de recargada la batería. La carga completa incrementa la vida útil de la batería y la capacidad de almacenamiento. El controlador maximiza la cantidad de energía solar hacia la batería y evita que la misma se sobrecaliente.
- Proporciona información de utilidad**
Los LED muestran en pantalla información sobre el estado y el nivel de la batería. Lo cual ayuda al usuario a operar mejor el sistema de energía solar.

Ventajas del controlador SHS

El controlador SHS de Morningstar tiene múltiples ventajas en comparación con otros controladores:

- Bajo Costo**
El SHS fue específicamente diseñado para satisfacer las necesidades del mercado de electrificación rural. El bajo costo es resultado del uso de la última tecnología y de un alto volumen de fabricación.
- Alta confiabilidad**
Morningstar es un líder mundial en la producción de controladores de carga solar. Todos nuestros controladores son diseñados para una alta calidad y una alta confiabilidad. El controlador SHS tiene un bajo régimen de fallas y presta servicio por mucho tiempo.
- Protecciones electrónicas**
El SHS tiene fusibles electrónicos incluidos que no necesitan reemplazo. Ninguna conexión errónea durante la instalación dañará el controlador.
- Tropicalización**
La electrónica del SHS está protegida con una cubierta a prueba de humedad que minimiza el daño que podrían causar la humedad y los insectos que eventualmente pudieran arañar en el sistema.
- Facilidad de uso**
El controlador SHS es completamente automático y no necesita ajustes ni selecciones por parte del usuario.

SHS™ CONTROLADOR SOLAR

INFORMACIÓN TÉCNICA

CAPACIDADES NOMINALES (TODOS A 12 VOLTIOS)

SHS-6	100 W ó 6 A PARA EL SISTEMA DE CARGA SOLAR Y LA CARGA
SHS-10	170 W ó 10 A PARA EL SISTEMA DE CARGA SOLAR Y LA CARGA

• Punto de regulación	14.3 V	• Indicaciones de los LED	Verde: en carga Niveles de batería: verde, amarillo y rojo Rojo: advertencia de bajo voltaje y desconexión Los 3 LED destellando — indicación de errores.
• Desconexión por bajo voltaje	11.5 V	• Terminales	Para tamaños de cable de hasta 4 mm ²
• Reconexión por bajo voltaje	12.4 V	• Dimensiones	15.1 x 6.6 x 3.8 cm
• Tipo de carga	PWM Sine (modulación de ancho de pulso) 4 etapas: En bruto, PWM, por incremento o "Boost" y Rotando Carga compensada en temperatura	• Peso	115 g
• Protecciones electrónicas	Cortocircuito y exceso de corriente — sistema de carga solar y carga Polaridad inversa — en sistema de carga solar, en la carga y en la batería Corriente inversa por la noche Alto voltaje — en la carga Reyes — en sistema solar, en la carga y en la batería	• Consumo propio	8 mA máximo
• Tropicalización	Placa de circuito — recubrimiento según norma Terminales — protegidos contra corrosión	• Temperatura	-25°C a +50°C
		• Humedad	100% sin condensación
		• Encapsulado	IP 22
		• Cumple con normas de	CE, Banco Mundial

GARANTÍA: Período de dos años de garantía. Póngase en contacto con Morningstar o su distribuidor autorizado para conocer los términos completos.

DISTRIBUIDOR MORNINGSTAR AUTORIZADO:



1000 Washington Crossing Road
Washington Crossing, PA 19377-6144
Tel: +1 215-321-4657 Fax: +1 215-321-4658
E-mail: info@morningstarcorp.com
Website: www.morningstarcorp.com

MFG/CO. OF U.S. DISTRICTS LABOR 12/04/1994

ANEXO M: Código del programa de Arduino

24/2/2019

Estacion Meteorológica.html

```
#include <LiquidCrystal.h>
#include <Wire.h>
#include <BigNumbers.h>
#include <Adafruit_BMP085.h>
#include <dht11.h>
#include <SoftwareSerial.h>
#include <Steing.h>

#define DHT11_PIN 2 //Se define el pin D2 para lectura del DHT11

SoftwareSerial SIM900(10, 11); //Se configura los pines D10 y D11 (puertos seriales del Arduino) para la conexión con el GPRS SIM900
LiquidCrystal lcd(3,4,5,6,7,8); //Se configura los pines para la conexión con el LCD

//Se crea los objetos para la muestra de los datos de Temperatura, Presión, Humedad y Velocidad en un tamaño mas grande
BigNumbers bigT(&lcd);
BigNumbers bigP(&lcd);
BigNumbers bigH(&lcd);
BigNumbers bigV(&lcd);

Adafruit_BMP085 SPT; //Se crea de la variable para lectura del BMP180
dht11 DHT; //Se crea la variable para lectura del DHT11

//Se crea las variables a utilizar
int P;
int T;
int H;
int V;
double Vel;
double D;
double dir;
String Dir="";
int chk;
double LL;
double ll;
String ll="";

//Se crea los caracteres para la muestra del texto mas grande
byte car_cero[8] =
{
  000111,
  001111,
  001111,
  001111,
  001111,
  001111,
  001111,
  001111
};
byte car_uno[8] =
{
  011111,
  011111,
  011111,
  000000,
  000000,
  000000,
  000000,
  000000
};
byte car_dos[8] =
{
  011100,
  011100,
  011100,
  011100,
  011100,
  011100,
  011100,
  011100
};
byte car_tres[8] =
{
  001111,
  000111,
  000000,
  000000,
  000000,
  000000,
  000011,
  000111
};
byte car_cuatro[8] =
{
  000000,
  000000,
  000000,
  000000,
  000000,
  011111,
  011111,
  011111
};
byte car_cinco[8] =
{
  011110,
  011100,
  000000,
  000000,
  000000,
  000000,
  011000,
  011100
};
byte car_seis[8] =
```

file:///C:/Users/David/Downloads/Tesis/Estacion%20Meteorol%C3%B3gica.html

1/6

```

{
  B11111,
  B11111,
  B11111,
  B00000,
  B00000,
  B00000,
  B11111,
  B11111
};
byte car_siete[8] =
{
  B00000,
  B00000,
  B00000,
  B00000,
  B00000,
  B00000,
  B00111,
  B01111
};

void setup() { //Inicialización de los dispositivos

  TCCR2B = TCCR2B & 0b11111000 | 0x01;
  lcd.begin(20,4); // Se inicia la pantalla LCD

  //Iniciamos los objetos para los textos numéricos mas grandes
  bigP.begin();
  bigT.begin();
  bigH.begin();
  bigV.begin();

  lcd.clear();//Limpiamos el LCD

  SIM900.begin(19200);//Configuramos la velocidad del puerto serial del GPRS SIM900
  Serial.begin(19200);//Configuramos la velocidad del puerto serial del GPRS SIM900
  if (!SPT.begin()){
    Serial.println("Could not find a valid BMP085 sensor, check wiring!");
    while (1) {}
  }
  delay(10000);//Tiempo necesario para la conexión con la red del operador
}

void loop() { //Función Principal

  comGPRS();//Llama a la función comGPRS
  if(SIM900.available())//Verificamos si hay datos disponibles desde el SIM900
  Serial.write(SIM900.read());//Escritura de datos en monitor serial
}

void sensores(){ //Lectura y muestra de Variables

  chk = DHT.read(DHT11_PIN); //Leemos los datos del sensor
  H=DHT.humidity; //Lectura de la humedad
  P=SPT.readPressure()/100; //Lectura de presión del BMP180
  T=SPT.readTemperature(); //Lectura de temperatura del BMP180
  V=analogRead(1); //Lectura analoga de velocidad del viento
  Vel-(((V*5)/1024)*30/5)*3.0; //Conversion a KM/H
  D=analogRead(2); //Lectura analoga de direccion del viento
  dir-((D*5)/1024); //Conversion a voltios
  ll=analogRead(0); //Lectura de la lluvia
  ll-((ll*5)/1024); //Conversion a voltios

  //Mostramos la temperatura y la presión.
  bigP.displayLargeInt(P, 2, 11, false);
  bigT.displayLargeInt(T, 1, 4, false);

  lcd.setCursor(0, 1);
  lcd.print("TEMP: ");
  lcd.setCursor(14, 1);
  lcd.print("°C");
  lcd.setCursor(0, 3);
  lcd.print("PRE: ");
  lcd.setCursor(15, 3);
  lcd.print("hPa");

  delay(3000);
  lcd.clear();

  //Mostramos la direccion y la velocidad del viento
  bigV.displayLargeInt(Vel, 1, 11, false);

  lcd.setCursor(0, 3);
  lcd.print("VEL: ");
  lcd.setCursor(15, 3);
  lcd.print("KM/H");

  if (dir>4.70 || dir<0.31){
    Dir="N";
    NORTE();
  }

  if(dir>0.32 && dir<0.94){
    Dir="NE";
    NORESTE();
  }

  if(dir>0.95 && dir<1.56){
    Dir="E";
    ESTE();
  }
}

```

```

if(dir>1.57 && dir<2.19){
  Dir="SE";
  SURESTE();
}

if(dir>2.20 && dir<2.81){
  Dir="S";
  SUR();
}

if(dir>2.82 && dir<3.44){
  Dir="SO";
  SUROESTE();
}

if(dir>3.45 && dir<4.06){
  Dir="O";
  OESTE();
}

if(dir>4.07 && dir<4.69){
  Dir="NO";
  NO();
}

lcd.setCursor(0, 1);
lcd.print("DIR: ");

delay(3000);
lcd.clear();

//Mostramos la lluvia y la humedad
bigr.displayLargeInt(H, 1, 11, false);

lcd.setCursor(0, 3);
lcd.print("HUM: ");
lcd.setCursor(15, 3);
lcd.print("%");

if (ll>2.50){
  ll="NO";
  NO();
}
else {
  ll="SI";
  SI();
}

lcd.setCursor(0, 1);
lcd.print("LLU: ");

delay(3000);
lcd.clear();
}

void comGPRS(){ //Envío de Información desde el GPRS a Thingspeak

  sensores();
  SIM000.println("AT+CIPSTATUS");//Se consulta el estado de la conexión
  DS();
  sensores();
  SIM000.println("AT+CIPMUX=0");//Se configura el dispositivo para una conexión de IP única o múltiple 0=única
  sensores();
  DS();
  SIM000.println("AT+CSIT=\\"internet.cnt.net.ec\\","\\" \\" \\" \\"");//Se configura la APN, nombre de usuario y contraseña
  sensores();
  DS();
  SIM000.println("AT+CIICR");//Se realiza la activación de los datos inalámbricos del GPRS
  sensores();
  DS();
  SIM000.println("AT+CIFSR");//Se obtiene nuestra IP local
  sensores();
  DS();
  SIM000.println("AT+CIPSPRT=0");//Se establece un indicador 'x' al enviar los datos
  TYP();
  sensores();
  DS();
  SIM000.println("AT+CIPSTART=\\"TCP\\","\\"api.thingspeak.com\\","\\"80\\");//Se indica el tipo de conexión, URL o dirección IP y puerto al que realizamos
  sensores();
  DS();
  SIM000.println("AT+CIPSEND");//Se prepara el envío de los datos
  sensores();
  DS();
  String datos="GET https://api.thingspeak.com/update?api_key=8MEQUX9D7ZV8BV&field1=" + String(T);
  String datos1="%field2=" + String(P);
  String datos2="%field3=" + String(Ve1);
  String datos3="%field4=" + String(Dir);
  String datos4="%field5=" + String(H);
  String datos5="%field6=" + String(LL);
  SIM000.println(datos+datos1+datos2+datos3+datos4+datos5);//Se envía los datos al servidor Thingspeak
  sensores();
  DS();
  SIM000.println((char)26);
  sensores();
  SIM000.println();
  sensores();
  DS();
  SIM000.println("AT+CIPSHUT");//Se cierra el contexto PDP del GPRS SIM000
  sensores();
  DS();
}

```

```

}

void DS(){ //Muestra los datos que va entregando el GPRS SIM900

while(SIM900.available() != 0)
Serial.write(SIM900.read());

}

void TVP(){ //Escritura de las variables sensadas en el Monitor Serie

Serial.print("Temperatura: ");
Serial.print(T); //Imprime la temperatura
Serial.println(" °C");
Serial.print("Presión: ");
Serial.print(P); //Imprime la temperatura
Serial.println(" hpa");
Serial.print("Velocidad: ");
Serial.print(vw1); //Imprime la temperatura
Serial.println(" KM/H");
Serial.print("Dirección: ");
Serial.print(Dir); //Imprime la temperatura
Serial.println(" °");
Serial.print("Humedad: ");
Serial.print(H); //Imprime la temperatura
Serial.println(" %");
Serial.print("Lluvia: ");
Serial.print(LL); //Imprime la temperatura

}

void NO(){ //Creamos el caracter NO(NOROESTE) en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

//Letra N
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(2);
lcd.write(2);
lcd.setCursor(8, 1);
lcd.write(byte(0));
lcd.write(byte(0));
lcd.write(2);

//Letra O
lcd.setCursor(11, 0);
lcd.write(byte(0));
lcd.write(1);
lcd.write(2);
lcd.setCursor(11, 1);
lcd.write(byte(0));
lcd.write(4);
lcd.write(2);

}

void SI(){ //Creamos el caracter SI en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

//Letra S
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(6);
lcd.write(5);
lcd.setCursor(8, 1);
lcd.write(7);
lcd.write(6);
lcd.write(2);

//Letra I
lcd.setCursor(12, 0);
lcd.write(byte(0));
lcd.setCursor(12, 1);
lcd.write(byte(0));

}

void NORTE (){//Creamos el caracter N(NORTE) en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

```

```

//Letra N
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(2);
lcd.write(2);
lcd.setCursor(8, 1);
lcd.write(byte(0));
lcd.write(byte(0));
lcd.write(2);
}

void SUR (){//Creamos el caracter S(SUR) en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

//Letra S
lcd.setCursor(9, 0);
lcd.write(byte(0));
lcd.write(6);
lcd.write(5);
lcd.setCursor(8, 1);
lcd.write(7);
lcd.write(6);
lcd.write(2);
}

void ESTE (){//Creamos el caracter E(ESTE) en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

//Letra E
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(6);
lcd.write(6);
lcd.setCursor(8, 1);
lcd.write(byte(0));
lcd.write(4);
lcd.write(4);
}

void OESTE (){//Creamos el caracter O(OESTE) en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

//Letra O
lcd.setCursor(11, 0);
lcd.write(byte(0));
lcd.write(1);
lcd.write(2);
lcd.setCursor(11, 1);
lcd.write(byte(0));
lcd.write(4);
lcd.write(2);
}

void NORESTE (){//Creamos el caracter NE(NORESTE) en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

//Letra N
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(2);
lcd.write(2);
lcd.setCursor(8, 1);
lcd.write(byte(0));
lcd.write(byte(0));
lcd.write(2);
}

```

```

//Letra E
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(6);
lcd.write(6);
lcd.setCursor(8, 1);
lcd.write(byte(0));
lcd.write(4);
lcd.write(4);
}

void SURESTE (){//Creamos el caracter SE(SURESTE) en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

//LETRA S
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(6);
lcd.write(5);
lcd.setCursor(8, 1);
lcd.write(7);
lcd.write(6);
lcd.write(2);

//LETRA E
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(6);
lcd.write(6);
lcd.setCursor(8, 1);
lcd.write(byte(0));
lcd.write(4);
lcd.write(4);
}

void SURESTE (){//Creamos el caracter SO(SURESTE) en letras grandes

lcd.createChar(0,car_cero);
lcd.createChar(1,car_uno);
lcd.createChar(2,car_dos);
lcd.createChar(3,car_tres);
lcd.createChar(4,car_cuatro);
lcd.createChar(5,car_cinco);
lcd.createChar(6,car_seis);
lcd.createChar(7,car_siete);

//Letra S
lcd.setCursor(8, 0);
lcd.write(byte(0));
lcd.write(6);
lcd.write(5);
lcd.setCursor(8, 1);
lcd.write(7);
lcd.write(6);
lcd.write(2);

//Letra O
lcd.setCursor(11,0);
lcd.write(byte(0));
lcd.write(1);
lcd.write(2);
lcd.setCursor(11, 1);
lcd.write(byte(0));
lcd.write(4);
lcd.write(2);
}

```

ANEXO N: Código de la programación en Android Studio

```
File - C:\Users\pc\AndroidStudioProjects\PRUEBA_IOT\app\src\main\java\com\example\pruebas_iot\MainActivity.java
1 package com.example.pc.prueba_iot;
2
3 import android.content.Intent;
4 import android.os.Handler;
5 import android.support.design.widget.FloatingActionButton;
6 import android.support.v7.app.AppCompatActivity;
7 import android.os.Bundle;
8 import android.view.View;
9 import android.widget.ImageView;
10 import android.widget.TextView;
11
12 import com.android.volley.Request;
13 import com.android.volley.RequestQueue;
14 import com.android.volley.Response;
15 import com.android.volley.VolleyError;
16 import com.android.volley.toolbox.JsonObjectRequest;
17 import com.android.volley.toolbox.Volley;
18
19 import org.json.JSONArray;
20 import org.json.JSONException;
21 import org.json.JSONObject;
22
23 import java.util.Objects;
24
25 public class MainActivity extends AppCompatActivity {
26
27     private RequestQueue queue;
28     private TextView mTextView;
29     private TextView mTextView2;
30     private TextView mTextView3;
31     private TextView mTextView4;
32     private TextView mTextView5;
33     private TextView mTextView6;
34
35     public String Temp;
36     public String Pres;
37     public String VelV;
38     public String DirV;
39     public String HumR;
40     public String Lluv;
41     public String m_si="SI";
42     public String m_no="NO";
43
44
45     private int Value_VelV;
```

```

46
47     Handler myHandler = new Handler();
48     final Handler handler = new Handler();
49     FloatingActionButton inf;
50
51
52     @Override
53     protected void onCreate(Bundle savedInstanceState) {
54         super.onCreate(savedInstanceState);
55         setContentView(R.layout.activity_main);
56
57         mTextView=findViewById(R.id.textViewID);
58         mTextView2=findViewById(R.id.textViewID2);
59         mTextView3=findViewById(R.id.textViewID3);
60         mTextView4=findViewById(R.id.textViewID4);
61         mTextView5=findViewById(R.id.textViewID5);
62         mTextView6=findViewById(R.id.textViewID6);
63         inf=(FloatingActionButton)findViewById(R.id.
floatingActionButton);
64
65         inf.setOnClickListener(new View.OnClickListener()
{
66             @Override
67             public void onClick(View v) {
68                 Intent i = new Intent(
getApplicationContext(),map_parapente.class);
69                 startActivity(i);
70             }
71         });
72         queue = Volley.newRequestQueue(this);
73         myHandler.postDelayed(new Runnable() {
74             @Override
75             public void run() {
76                 obtenerDatosVolley();
77                 if(Objects.equals(Lluv,m_no)) {
78                     if (Value_VelV > 30) {
79
80                         ImageView semaforo = findViewById(
R.id.color_sem);
81
82                         semaforo.setImageResource(R.
drawable.sem_rojo);
83
84                         TextView mensaje1 = (TextView)
findViewById(R.id.textViewID7);

```

```

85
86         mensaje.setText("NO ESTA
PERMITIDO EL VUELO!! \n\n Se registra una velocidad
demasiado alta como para poder realizar el despegue en
condiciones optimas, espere a que disminuya la velocidad
del viento.");
87
88         } else if (Value_VelV >= 15) {
89             ImageView semaforo = findViewById
(R.id.color_sem);
90             semaforo.setImageResource(R.
drawable.sem_amarillo);
91
92             TextView mensaje = (TextView)
findViewById(R.id.textViewID7);
93
94             mensaje.setText("SE PUEDE
REALIZAR EL DESPEGUE CON PRECAUCIÓN!!\n\n Se registra una
velocidad moderada para realizar el despegue.");
95             } else if (Value_VelV <= 15) {
96
97             ImageView semaforo = findViewById
(R.id.color_sem);
98             semaforo.setImageResource(R.
drawable.sem_verde);
99
100            TextView mensaje = (TextView)
findViewById(R.id.textViewID7);
101
102            mensaje.setText("SE PUEDE
REALIZAR EL DESPEGUE!!\n\nSe registra una velocidad
optima para realizar el despegue, disfrute de su vuelo.")
;
103            )
104        }
105        else if(Objects.equals(Lluv,m_si)){
106            ImageView semaforo = findViewById(R.
id.color_sem);
107
108            semaforo.setImageResource(R.drawable.
sem_rojo);
109
110            TextView mensaje = (TextView)
findViewById(R.id.textViewID7);
111

```

```

112         mensaje.setText("NO ESTA PERMITIDO
EL VUELO!!\n\n Se registra lluvia en la pista de despegue
.");
113
114     }
115     handler.postDelayed(this,1000);
116 }
117 },0);
118
119
120
121 )
122
123
124
125     public void obtenerDatosVolley(){
126
127         String url = "https://api.thingspeak.com/channels
/625040/feeds.json?results=10";
128
129         JsonObjectRequest request=new JsonObjectRequest(
Request.Method.GET, url, null, new Response.Listener<
JSONObject>() {
130             @Override
131             public void onResponse(JSONObject response) {
132
133                 try {
134                     JSONArray mJSONArray = response.
getJSONArray("feeds");
135                     for(int i=0; i<mJSONArray.length();i
++) {
136
137
138                         JSONObject mJsonObteject =
mJSONArray.getJSONObject(i);
139                         Temp= mJsonObteject.getString("
field1");
140                         Pres= mJsonObteject.getString("
field2");
141                         VelV= mJsonObteject.getString("
field3");
142                         DirV= mJsonObteject.getString("
field4");
143                         HumR=mJsonObteject.getString("
field5");

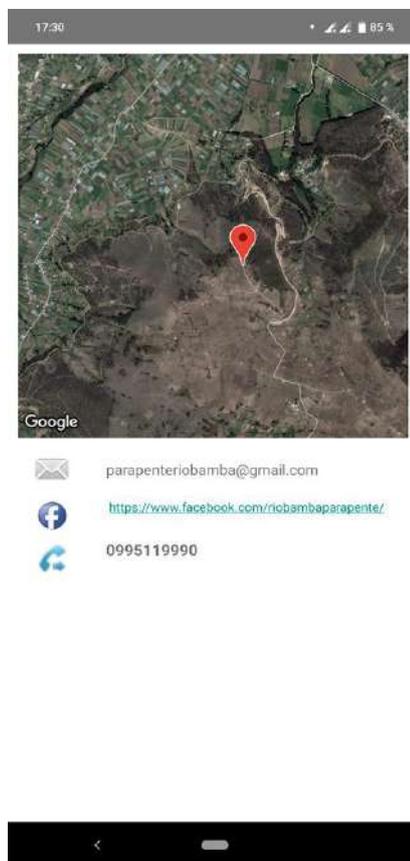
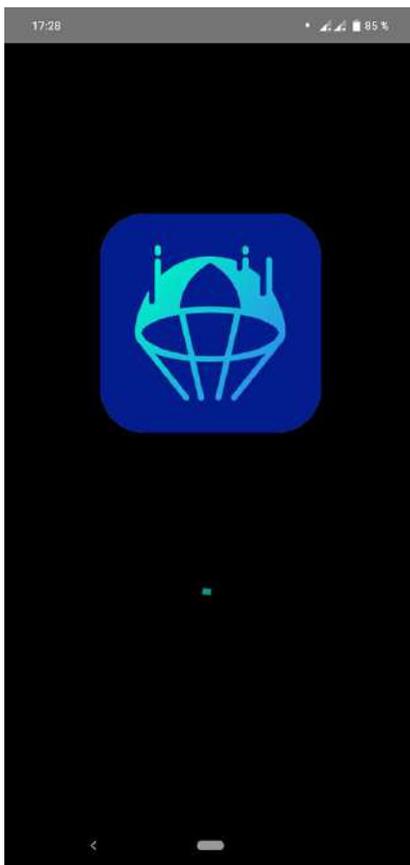
```

```

144         Lluv=mJsonObject.getString("
    field6");
145
146         //Value_Temp=Integer.parseInt(
    Temp.replaceAll("[\\D]", ""));
147         //Value_Pres=Integer.parseInt(
    Pres.replaceAll("[\\D]", ""));
148         //Value_VelV=Integer.parseInt(
    VelV.replaceAll("[\\D]", ""));
149
150         try{
151
152             Value_VelV = Integer.parseInt
    (VelV);
153
154
155         }catch (NumberFormatException nfe
    ){
156
157         }
158
159         mTextView.setText(Temp+"°C");
160         mTextView2.setText(Pres+"hPa")
    ;
161         mTextView3.setText(VelV);
162         mTextView4.setText(DirV);
163         mTextView5.setText(HumR+"%");
164         mTextView6.setText(Lluv);
165     }
166     } catch (JSONException e) {
167         e.printStackTrace();
168     }
169 }
170 }, new Response.ErrorListener() {
171     @Override
172     public void onErrorResponse(VolleyError error
    ){
173
174     }
175 }
176 );
177 queue.add(request);
178 }
179 }
180

```

ANEXO O: Interfaz de la aplicación



ANEXO P: Datos recibidos en plataforma IoT.

1	2018-12-03 17:26:35 UTC,73,19,737,106.42,NE,72,NO
2	2018-12-03 17:28:28 UTC,74,19,737,71.19,NE,72,NO
3	2018-12-03 17:30:23 UTC,75,21,737,54.84,S,94,NO
4	2018-12-03 17:32:20 UTC,76,19,737,54,S,75,NO
5	
6	2018-12-03 17:36:09 UTC,78,19,737,2,SO,74,NO
7	2018-12-03 17:38:02 UTC,79,19,737,2,SO,73,NO
8	2018-12-03 17:39:56 UTC,80,19,737,1,SO,74,NO
9	2018-12-03 17:41:51 UTC,81,19,737,1,SO,74,NO
10	2018-12-03 17:43:44 UTC,82,19,737,25,N,73,NO
11	2018-12-03 17:50:01 UTC,83,19,737,43,NE,69,NO
12	2018-12-03 17:51:55 UTC,84,19,737,43,NE,69,NO
13	2018-12-03 17:53:58 UTC,85,19,737,43,NE,70,NO
14	
15	2018-12-03 17:57:37 UTC,87,19,737,43,NE,70,NO
16	2018-12-03 17:59:31 UTC,88,19,737,43,NE,70,NO
17	2018-12-03 18:01:26 UTC,89,19,737,43,NE,70,NO
18	2018-12-03 18:03:18 UTC,90,19,737,42,NE,70,NO
19	2018-12-03 18:05:12 UTC,91,19,737,13,SE,69,NO
20	2018-12-03 18:07:07 UTC,92,19,737,13,SE,70,NO
21	2018-12-03 18:09:06 UTC,93,19,737,13,SE,70,NO
22	2018-12-03 18:10:54 UTC,94,19,737,13,SE,70,NO
23	2018-12-03 18:12:48 UTC,95,19,737,13,SE,93,NO
24	2018-12-03 18:14:42 UTC,96,19,737,13,SE,71,NO
25	
26	2018-12-03 18:18:30 UTC,98,19,736,13,SE,70,NO
27	2018-12-03 18:20:24 UTC,99,19,736,13,SE,93,NO

28	2018-12-03 18:22:19 UTC,100,19,736,13,SE,93,NO
29	2018-12-03 18:24:12 UTC,101,19,736,13,SE,93,NO
30	2018-12-03 18:26:06 UTC,102,19,736,14,SE,70,NO
31	2018-12-03 18:28:00 UTC,103,19,736,13,SE,93,NO
32	2018-12-03 18:29:54 UTC,104,19,736,13,SE,93,NO
33	2018-12-03 18:31:48 UTC,105,19,736,13,SE,93,NO
34	2018-12-03 18:33:42 UTC,106,19,736,13,SE,93,NO
35	2018-12-03 18:35:36 UTC,107,19,736,13,SE,93,NO
36	2018-12-03 18:37:30 UTC,108,19,736,13,SE,93,NO
37	2018-12-03 18:39:24 UTC,109,19,736,13,SE,93,NO
38	
39	2018-12-03 18:43:12 UTC,111,19,736,13,SE,71,NO
40	2018-12-03 18:45:06 UTC,112,19,736,13,SE,70,NO
41	2018-12-03 18:47:00 UTC,113,19,736,13,SE,75,NO
42	2018-12-03 18:48:54 UTC,114,19,736,13,SE,71,NO
43	2018-12-03 18:50:48 UTC,115,19,736,13,SE,70,NO
44	2018-12-03 18:52:42 UTC,116,19,736,13,SE,93,NO
45	2018-12-03 18:54:36 UTC,117,19,736,14,SE,93,NO

46	2018-12-03 UTC,118,19,736,13,SE,93,NO	18:56:30
47	2018-12-03 UTC,119,19,736,36,NE,70,NO	18:58:24
48	2018-12-03 UTC,120,19,736,36,NE,69,NO	19:00:18
49	2018-12-03 UTC,121,19,736,36,NE,69,NO	19:02:12
50	2018-12-03 UTC,122,19,736,36,NE,69,NO	19:04:06
51		
52	2018-12-03 UTC,124,19,736,36,NE,68,NO	19:07:54
53	2018-12-03 UTC,125,19,736,36,NE,68,NO	19:09:48
54	2018-12-03 UTC,126,19,736,36,NE,68,NO	19:11:41
55	2018-12-03 UTC,127,19,736,36,NE,67,NO	19:13:36
56	2018-12-03 UTC,128,19,736,36,NE,68,NO	19:15:29
57	2018-12-03 UTC,129,19,736,36,NE,68,NO	19:17:24
58	2018-12-03 UTC,130,19,736,36,NE,68,NO	19:19:18
59	2018-12-03 UTC,131,19,736,36,NE,67,NO	19:21:12
60	2018-12-03 UTC,132,19,735,36,NE,67,NO	19:23:05
61	2018-12-03 UTC,133,19,735,35,NE,67,NO	19:25:00
62	2018-12-03 UTC,134,19,735,36,NE,67,NO	19:26:54
63		
64		
65	2018-12-03 UTC,137,19,735,35,NE,67,NO	19:32:36

66	2018-12-03 UTC,138,19,735,35,NE,67,NO	19:34:29
67	2018-12-03 UTC,139,19,735,36,NE,67,NO	19:36:23
68	2018-12-03 UTC,140,19,735,36,NE,66,NO	19:38:17
69	2018-12-03 UTC,141,19,735,36,NE,66,NO	19:40:11
70	2018-12-03 UTC,142,19,735,36,NE,66,NO	19:42:05
71	2018-12-03 UTC,143,19,735,36,NE,65,NO	19:43:59
72	2018-12-03 UTC,144,19,735,36,NE,65,NO	19:45:53
73	2018-12-03 UTC,145,19,735,36,NE,64,NO	19:47:48
74	2018-12-03 UTC,146,19,735,36,NE,65,NO	19:49:41
75	2018-12-03 UTC,147,19,735,36,NE,65,NO	19:51:36
76	2018-12-03 UTC,148,19,735,36,NE,65,NO	19:53:29
77	2018-12-03 UTC,149,19,735,36,NE,65,NO	19:55:23
78	2018-12-03 UTC,150,19,735,36,NE,65,NO	19:57:17
79	2018-12-03 UTC,151,19,735,36,NE,65,NO	19:59:11
80	2018-12-03 UTC,152,19,735,36,NE,65,NO	20:01:05
81	2018-12-03 UTC,153,19,735,35,NE,64,NO	20:02:59
82	2018-12-03 UTC,154,19,735,36,NE,64,NO	20:04:53
83	2018-12-03 UTC,155,19,735,36,NE,65,NO	20:06:47
84	2018-12-03 UTC,156,19,735,36,NE,65,NO	20:08:41

85		
86	2018-12-03	20:12:29
	UTC,158,19,735,35,NE,65,NO	
87	2018-12-03	20:14:23
	UTC,159,19,735,36,NE,65,NO	
88	2018-12-03	20:16:16
	UTC,160,19,735,36,NE,65,NO	
89	2018-12-03	20:18:10
	UTC,161,19,735,36,NE,65,NO	
90	2018-12-03	20:20:05
	UTC,162,19,735,36,NE,65,NO	
91	2018-12-03	20:21:59
	UTC,163,19,735,36,NE,65,NO	
92	2018-12-03	20:23:53
	UTC,164,19,735,35,NE,65,NO	
93	2018-12-03	20:25:49
	UTC,165,19,735,36,NE,65,NO	
94	2018-12-03	20:27:40
	UTC,166,19,735,36,NE,65,NO	
95		
96	2018-12-03	20:31:28
	UTC,168,19,735,36,NE,65,NO	
97	2018-12-03	20:33:22
	UTC,169,19,735,35,NE,65,NO	
98	2018-12-03	20:37:11
	UTC,170,19,735,36,NE,65,NO	
99	2018-12-03	20:39:04
	UTC,171,19,735,36,NE,65,NO	
100	2018-12-03	20:40:58
	UTC,172,19,735,35,NE,65,NO	
101	2018-12-03	20:42:53
	UTC,173,19,735,35,NE,65,NO	
102	2018-12-03	20:44:46
	UTC,174,19,735,35,NE,65,NO	
103	2018-12-03	20:46:40
	UTC,175,19,735,35,NE,65,NO	
104		

105	2018-12-03	20:50:30
	UTC,177,19,735,36,NE,65,NO	
106	2018-12-03	20:52:22
	UTC,178,19,735,35,NE,65,NO	
107	2018-12-03	20:54:16
	UTC,179,19,735,36,NE,65,NO	
108	2018-12-03	20:56:10
	UTC,180,19,735,36,NE,65,NO	
109	2018-12-03	20:58:04
	UTC,181,19,735,36,NE,65,NO	
110	2018-12-03	20:59:58
	UTC,182,19,735,35,NE,66,NO	
111	2018-12-03	21:01:52
	UTC,183,19,735,35,NE,93,NO	
112	2018-12-03	21:03:46
	UTC,184,19,735,35,NE,93,NO	
113		
114	2018-12-03	21:07:34
	UTC,186,19,735,35,NE,93,NO	
115	2018-12-03	21:09:28
	UTC,187,19,735,35,NE,93,NO	
116	2018-12-03	21:11:22
	UTC,188,19,735,35,NE,93,NO	
117	2018-12-03	21:13:16
	UTC,189,19,735,35,NE,93,NO	
118	2018-12-03	21:15:10
	UTC,190,19,735,36,NE,93,NO	
119	2018-12-03	21:17:04
	UTC,191,19,735,36,NE,93,NO	
120	2018-12-03	21:18:58
	UTC,192,19,735,36,NE,93,NO	
121	2018-12-03	21:20:52
	UTC,193,19,735,36,NE,93,NO	
122	2018-12-03	21:22:46
	UTC,194,19,735,35,NE,93,NO	
123	2018-12-03	21:24:40
	UTC,195,19,735,35,NE,93,NO	
124		

125	2018-12-03 UTC,197,19,735,36,NE,93,NO	21:28:28
126	2018-12-03 UTC,198,19,735,35,NE,93,NO	21:32:15
127	2018-12-03 UTC,199,19,735,35,NE,93,NO	21:34:09
128	2018-12-03 UTC,200,19,735,35,NE,93,NO	21:36:05
129	2018-12-03 UTC,201,19,735,35,NE,93,NO	21:37:58
130	2018-12-03 UTC,202,19,735,35,NE,93,NO	21:39:52
131	2018-12-03 UTC,203,19,735,36,NE,93,NO	21:41:45
132	2018-12-03 UTC,204,19,735,68,SE,58,NO	21:43:39
133		
134	2018-12-03 UTC,206,19,735,68,SE,65,NO	21:47:27
135	2018-12-03 UTC,207,19,735,68,SE,93,NO	21:49:21
136	2018-12-03 UTC,208,19,735,68,SE,63,NO	21:51:16
137	2018-12-03 UTC,209,19,735,68,SE,62,NO	21:53:10
138	2018-12-03 UTC,210,19,735,68,SE,63,NO	21:55:03
139	2018-12-03 UTC,211,19,736,68,SE,63,NO	21:56:57
140	2018-12-03 UTC,212,19,736,68,SE,62,NO	21:58:51
141		
142	2018-12-03 UTC,214,19,736,68,SE,63,NO	22:02:39
143	2018-12-03 UTC,215,19,736,68,SE,63,NO	22:04:33
144	2018-12-03 UTC,216,19,736,68,SE,63,NO	22:08:21

145	2018-12-03 UTC,217,19,736,68,SE,63,NO	22:10:17
146	2018-12-03 UTC,218,19,736,68,SE,65,NO	22:12:09
147	2018-12-03 UTC,219,19,736,69,SE,65,NO	22:14:03
148	2018-12-03 UTC,220,19,736,68,SE,65,NO	22:17:51
149	2018-12-03 UTC,221,19,736,68,SE,65,NO	22:19:45
150	2018-12-03 UTC,222,19,736,68,SE,65,NO	22:21:39
151	2018-12-03 UTC,223,19,736,68,SE,65,NO	22:23:36
152		
153	2018-12-03 UTC,225,19,736,68,SE,65,NO	22:29:15
154	2018-12-03 UTC,226,19,736,68,SE,64,NO	22:31:09
155	2018-12-03 UTC,227,19,736,68,SE,64,NO	22:33:03
156	2018-12-03 UTC,228,19,736,68,SE,65,NO	22:34:57
157		
158	2018-12-03 UTC,230,19,736,68,SE,65,NO	22:38:44
159	2018-12-03 UTC,231,19,736,68,SE,65,NO	22:40:39
160	2018-12-03 UTC,232,19,736,68,SE,65,NO	22:42:47
161	2018-12-03 UTC,233,19,736,68,SE,65,NO	22:44:26
162	2018-12-03 UTC,234,19,736,68,SE,65,NO	22:46:20
163	2018-12-03 UTC,235,19,736,68,SE,65,NO	22:48:15
164	2018-12-03 UTC,236,19,736,69,SE,66,NO	22:50:08

165	2018-12-03 UTC,237,19,736,68,SE,65,NO	22:52:03
166	2018-12-03 UTC,238,19,736,69,SE,65,NO	22:53:57
167	2018-12-03 UTC,239,19,736,69,SE,65,NO	22:55:50
168	2018-12-03 UTC,240,19,736,69,SE,65,NO	22:57:45
169		
170	2018-12-03 UTC,242,19,736,69,SE,66,NO	23:01:33
171	2018-12-03 UTC,243,19,736,69,SE,66,NO	23:03:26
172	2018-12-03 UTC,244,19,736,69,SE,66,NO	23:05:20
173		
174	2018-12-03 UTC,246,19,736,69,SE,66,NO	23:09:08
175	2018-12-03 UTC,247,19,736,69,SE,66,NO	23:11:02
176	2018-12-03 UTC,248,19,736,69,SE,66,NO	23:12:56
177	2018-12-03 UTC,249,19,736,68,SE,66,NO	23:14:50
178	2018-12-03 UTC,250,19,736,68,SE,66,NO	23:16:44
179	2018-12-03 UTC,251,19,736,69,SE,66,NO	23:18:43
180	2018-12-03 UTC,252,19,736,68,SE,66,NO	23:20:32
181	2018-12-03 UTC,253,19,737,68,SE,66,NO	23:22:26
182	2018-12-03 UTC,254,19,737,68,SE,68,NO	23:24:20
183	2018-12-03 UTC,255,19,737,68,SE,66,NO	23:26:14
184	2018-12-03 UTC,256,19,736,68,SE,66,NO	23:28:10

185	2018-12-03 UTC,257,19,737,68,SE,66,NO	23:30:06
186		
187	2018-12-03 UTC,259,19,737,68,SE,66,NO	23:33:50
188	2018-12-03 UTC,260,19,737,68,SE,66,NO	23:35:44
189	2018-12-03 UTC,261,19,737,68,SE,67,NO	23:37:38
190	2018-12-03 UTC,262,19,737,68,SE,67,NO	23:39:32
191	2018-12-03 UTC,263,19,737,68,SE,67,NO	23:41:28
192	2018-12-03 UTC,264,19,737,68,SE,67,NO	23:43:20
193	2018-12-03 UTC,265,19,737,68,SE,67,NO	23:45:14
194	2018-12-03 UTC,266,19,737,68,SE,67,NO	23:47:08
195	2018-12-03 UTC,267,19,737,68,SE,67,NO	23:49:02
196	2018-12-03 UTC,268,19,737,69,SE,67,NO	23:50:56
197		
198	2018-12-03 UTC,270,19,737,68,SE,67,NO	23:56:38
199	2018-12-04 UTC,271,19,737,68,SE,67,NO	00:00:26
200	2018-12-04 UTC,272,19,737,68,SE,67,NO	00:02:19
201	2018-12-04 UTC,273,19,737,68,SE,67,NO	00:04:13
202	2018-12-04 UTC,274,19,737,68,SE,67,NO	00:06:08
203	2018-12-04 UTC,275,19,737,68,SE,67,NO	00:08:03
204	2018-12-04 UTC,276,19,737,68,SE,67,NO	00:09:55

205	2018-12-04 UTC,277,19,737,68,SE,67,NO	00:11:49
206	2018-12-04 UTC,278,19,737,68,SE,67,NO	00:13:44
207	2018-12-04 UTC,279,19,737,68,SE,67,NO	00:15:38
208	2018-12-04 UTC,280,19,737,68,SE,67,NO	00:19:25
209		
210	2018-12-04 UTC,282,19,737,68,SE,67,NO	00:23:13
211	2018-12-04 UTC,283,19,737,68,SE,67,NO	00:25:07
212	2018-12-04 UTC,284,19,737,68,SE,67,NO	00:27:01
213	2018-12-04 UTC,285,19,737,68,SE,67,NO	00:28:55
214	2018-12-04 UTC,286,19,737,68,SE,68,NO	00:30:50
215	2018-12-04 UTC,287,19,737,68,SE,68,NO	00:32:43
216	2018-12-04 UTC,288,19,737,68,SE,68,NO	00:34:37
217		
218	2018-12-04 UTC,290,19,737,68,SE,68,NO	00:38:25
219	2018-12-04 UTC,291,19,738,68,SE,68,NO	00:40:19
220	2018-12-04 UTC,292,19,738,68,SE,68,NO	00:42:13
221	2018-12-04 UTC,293,19,737,68,SE,68,NO	00:44:07
222	2018-12-04 UTC,294,19,738,68,SE,68,NO	00:46:01
223	2018-12-04 UTC,295,19,738,68,SE,68,NO	00:49:49
224	2018-12-04 UTC,296,19,738,68,SE,68,NO	00:51:42

225	2018-12-04 UTC,297,19,738,68,SE,68,NO	00:53:37
226	2018-12-04 UTC,298,19,738,68,SE,68,NO	00:55:31
227	2018-12-04 UTC,299,19,738,68,SE,68,NO	00:57:24
228	2018-12-04 UTC,300,19,738,68,SE,68,NO	00:59:19
229	2018-12-04 UTC,301,19,738,68,SE,68,NO	01:01:13
230	2018-12-04 UTC,302,19,738,68,SE,68,NO	01:03:06
231		
232	2018-12-04 UTC,304,19,738,68,SE,68,NO	01:06:55
233	2018-12-04 UTC,305,19,738,69,SE,68,NO	01:10:43
234	2018-12-04 UTC,306,19,738,69,SE,68,NO	01:14:31
235	2018-12-04 UTC,307,19,738,69,SE,68,NO	01:16:24
236	2018-12-04 UTC,308,19,738,69,SE,68,NO	01:18:19
237	2018-12-04 UTC,309,19,738,69,SE,68,NO	01:20:12
238	2018-12-04 UTC,310,19,738,69,SE,68,NO	01:22:06
239	2018-12-04 UTC,311,19,738,69,SE,68,NO	01:24:00
240	2018-12-04 UTC,312,19,738,69,SE,68,NO	01:25:54
241	2018-12-04 UTC,313,19,738,69,SE,68,NO	01:27:48
242		
243	2018-12-04 UTC,315,19,738,69,SE,68,NO	01:31:36
244	2018-12-04 UTC,316,19,738,69,SE,68,NO	01:33:30

245	2018-12-04 UTC,317,19,738,69,SE,68,NO	01:35:24
246	2018-12-04 UTC,318,19,738,69,SE,68,NO	01:37:18
247	2018-12-04 UTC,319,19,738,69,SE,68,NO	01:39:12
248	2018-12-04 UTC,320,19,739,69,SE,68,NO	01:41:06
249	2018-12-04 UTC,321,19,739,69,SE,68,NO	01:43:00
250	2018-12-04 UTC,322,19,739,69,SE,68,NO	01:44:54
251	2018-12-04 UTC,323,19,739,69,SE,68,NO	01:46:48
252		
253	2018-12-04 UTC,325,19,739,69,SE,68,NO	01:50:36
254	2018-12-04 UTC,326,19,739,69,SE,68,NO	01:52:30
255	2018-12-04 UTC,327,19,739,69,SE,68,NO	01:54:24
256	2018-12-04 UTC,328,19,739,69,SE,68,NO	01:56:18
257	2018-12-04 UTC,329,19,739,69,SE,68,NO	01:58:12
258	2018-12-04 UTC,330,19,739,69,SE,69,NO	02:00:06
259	2018-12-04 UTC,331,19,739,69,SE,69,NO	02:01:59
260	2018-12-04 UTC,332,19,739,69,SE,69,NO	02:03:54
261	2018-12-04 UTC,333,19,739,69,SE,69,NO	02:05:48
262	2018-12-04 UTC,334,19,739,69,SE,69,NO	02:07:41
263	2018-12-04 UTC,335,19,739,69,SE,69,NO	02:09:35

264	2018-12-04 UTC,336,19,739,69,SE,69,NO	02:11:29
265	2018-12-04 UTC,337,19,739,69,SE,69,NO	02:13:29
266		
267	2018-12-04 UTC,339,19,739,69,SE,69,NO	02:17:11
268	2018-12-04 UTC,340,19,739,69,SE,69,NO	02:19:06
269	2018-12-04 UTC,341,19,739,69,SE,69,NO	02:21:00
270	2018-12-04 UTC,342,19,739,69,SE,69,NO	02:22:54
271	2018-12-04 UTC,343,19,739,69,SE,69,NO	02:24:47
272	2018-12-04 UTC,344,19,739,69,SE,69,NO	02:26:41
273	2018-12-04 UTC,345,19,739,69,SE,69,NO	02:28:35
274	2018-12-04 UTC,346,19,739,69,SE,69,NO	02:30:29
275	2018-12-04 UTC,347,19,739,69,SE,69,NO	02:32:23
276	2018-12-04 UTC,348,19,739,69,SE,69,NO	02:34:17
277	2018-12-04 UTC,349,19,739,69,SE,69,NO	02:36:11
278	2018-12-04 UTC,350,19,739,69,SE,69,NO	02:38:06
279		
280	2018-12-04 UTC,352,19,739,69,SE,69,NO	02:43:47
281	2018-12-04 UTC,353,19,739,69,SE,69,NO	02:47:36
282	2018-12-04 UTC,354,19,739,69,SE,69,NO	02:49:29
283	2018-12-04 UTC,355,19,739,69,SE,69,NO	02:51:23

284	2018-12-04 UTC,356,19,739,69,SE,69,NO	02:53:17
285	2018-12-04 UTC,357,19,739,69,SE,69,NO	02:57:05
286	2018-12-04 UTC,358,19,739,69,SE,69,NO	02:58:59
287	2018-12-04 UTC,359,19,739,69,SE,69,NO	03:00:53
288	2018-12-04 UTC,360,19,740,69,SE,69,NO	03:02:47
289		
290	2018-12-04 UTC,362,19,739,69,SE,69,NO	03:08:29
291	2018-12-04 UTC,363,19,739,69,SE,69,NO	03:10:22
292	2018-12-04 UTC,364,19,740,69,SE,70,NO	03:12:17
293	2018-12-04 UTC,365,19,739,69,SE,70,NO	03:14:11
294	2018-12-04 UTC,366,19,740,69,SE,69,NO	03:16:10
295	2018-12-04 UTC,367,19,739,69,SE,70,NO	03:17:58
296	2018-12-04 UTC,368,19,739,69,SE,70,NO	03:19:52
297	2018-12-04 UTC,369,19,740,69,SE,70,NO	03:21:46
298		
299	2018-12-04 UTC,371,19,739,69,SE,70,NO	03:25:35
300	2018-12-04 UTC,372,19,740,69,SE,69,NO	03:27:28
301	2018-12-04 UTC,373,19,740,69,SE,69,NO	03:29:23
302	2018-12-04 UTC,374,19,740,69,SE,69,NO	03:31:16
303	2018-12-04 UTC,375,19,739,17,SE,69,NO	03:33:10

304	2018-12-04 03:35:05 UTC,376,19,740,0,N,69,NO	
305	2018-12-04 UTC,377,19,740,106,E,69,NO	03:38:52
306	2018-12-04 UTC,378,19,740,87,SE,69,NO	03:40:46
307	2018-12-04 UTC,379,19,739,87,SE,70,NO	03:42:40
308	2018-12-04 UTC,380,19,740,87,SE,70,NO	03:46:28
309		
310	2018-12-04 UTC,382,19,740,87,SE,71,NO	03:50:16
311	2018-12-04 UTC,383,19,740,87,SE,71,NO	03:52:10
312	2018-12-04 UTC,384,19,740,87,SE,70,NO	03:54:06
313	2018-12-04 UTC,385,19,740,87,SE,70,NO	03:55:58
314	2018-12-04 UTC,386,19,740,87,SE,70,NO	03:57:52
315	2018-12-04 UTC,387,19,740,87,SE,70,NO	03:59:46
316	2018-12-04 UTC,388,19,740,87,SE,70,NO	04:01:40
317	2018-12-04 UTC,389,19,740,87,SE,70,NO	04:03:34
318	2018-12-04 UTC,390,19,740,87,SE,70,NO	04:05:28
319	2018-12-04 UTC,391,19,740,87,SE,70,NO	04:07:23
320	2018-12-04 UTC,392,19,740,86,SE,70,NO	04:09:16
321		
322	2018-12-04 UTC,394,19,740,87,SE,70,NO	04:13:04
323	2018-12-04 UTC,395,19,740,87,SE,70,NO	04:14:59

324	2018-12-04 UTC,396,19,740,87,SE,70,NO	04:16:52
325	2018-12-04 UTC,397,19,740,87,SE,70,NO	04:18:51
326	2018-12-04 UTC,398,19,740,87,SE,70,NO	04:22:33
327	2018-12-04 UTC,399,19,740,86,SE,70,NO	04:24:27
328	2018-12-04 UTC,400,19,740,87,SE,70,NO	04:26:21
329	2018-12-04 UTC,401,19,740,86,SE,70,NO	04:28:15
330	2018-12-04 UTC,402,19,740,87,SE,70,NO	04:30:09
331		
332	2018-12-04 UTC,404,19,740,87,SE,70,NO	04:33:57
333	2018-12-04 UTC,405,19,740,87,SE,70,NO	04:35:51
334	2018-12-04 UTC,406,19,740,86,SE,69,NO	04:39:39
335	2018-12-04 UTC,407,19,740,87,SE,69,NO	04:41:33
336	2018-12-04 UTC,408,19,740,87,SE,70,NO	04:43:27
337	2018-12-04 UTC,409,19,740,86,SE,70,NO	04:45:21
338	2018-12-04 UTC,410,19,740,87,SE,70,NO	04:47:15
339	2018-12-04 UTC,411,19,740,87,SE,70,NO	04:49:09
340	2018-12-04 UTC,412,19,740,87,SE,69,NO	04:51:03
341	2018-12-04 UTC,413,19,740,87,SE,70,NO	04:52:57
342	2018-12-04 UTC,414,19,740,87,SE,69,NO	04:54:51

343	2018-12-04 UTC,415,19,740,87,SE,69,NO	04:56:45
344	2018-12-04 UTC,416,19,740,87,SE,69,NO	04:58:39
345	2018-12-04 UTC,417,19,740,86,SE,69,NO	05:00:33
346	2018-12-04 UTC,418,19,740,87,SE,68,NO	05:02:27
347	2018-12-04 UTC,419,19,740,87,SE,68,NO	05:04:21
348		
349	2018-12-04 UTC,421,19,740,87,SE,68,NO	05:08:09
350	2018-12-04 UTC,422,19,740,87,SE,68,NO	05:10:03
351	2018-12-04 UTC,423,19,740,87,SE,68,NO	05:11:57
352	2018-12-04 UTC,424,19,740,86,SE,68,NO	05:13:51
353	2018-12-04 UTC,425,19,740,86,SE,68,NO	05:15:45
354	2018-12-04 UTC,426,19,740,86,SE,68,NO	05:17:39
355		
356	2018-12-04 UTC,428,19,739,86,SE,68,NO	05:21:27
357	2018-12-04 UTC,429,19,740,86,SE,68,NO	05:23:20
358	2018-12-04 UTC,430,19,740,87,SE,68,NO	05:25:17
359	2018-12-04 UTC,431,19,739,87,SE,68,NO	05:27:08
360	2018-12-04 UTC,432,19,740,87,SE,68,NO	05:29:02