

DESARROLLO DE UN SISTEMA A BORDO PARA LA MEDICIÓN DE LAS
VARIABLES IMPLÍCITAS DEL CICLO DE CONDUCCIÓN PARA
MOTOCICLETAS EN LA CIUDAD DE BUCARAMANGA, COLOMBIA

SANTIAGO URIBE JAIMES
RAY STEVENSON AMAYA MALDONADO

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA MECATRÓNICA
BUCARAMANGA
2022

DESARROLLO DE UN SISTEMA A BORDO PARA LA MEDICIÓN DE LAS
VARIABLES IMPLÍCITAS DEL CICLO DE CONDUCCIÓN PARA
MOTOCICLETAS EN LA CIUDAD DE BUCARAMANGA, COLOMBIA

SANTIAGO URIBE JAIMES
RAY STEVENSON AMAYA MALDONADO

Trabajo de grado presentado como requisito para optar al título de:
INGENIERO MECATRÓNICO

DIRECTORA: MEng. MSc. JESSICA GISSELLA MARADEY LAZARO.
CODIRECTOR: PhD. JOSÉ IGNACIO HUERTAS CARDOZO

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA MECATRÓNICA
BUCARAMANGA
2022

DEDICATORIA

El presente trabajo se lo dedico a mi madre Luz Esperanza Maldonado y a mi padre Reinaldo Amaya por su incesable apoyo durante este gran camino de formación profesional y personal, a mis hermanas Zeidylix Blanco y Karina Amaya por su apoyo incondicional en los momentos donde más lo he necesitado, a mi compañero de proyecto Santiago Uribe por ser el complemento de un gran equipo de trabajo, a mi amada Angel Acevedo por ser mi sostén en los momentos más difíciles y estar a mi lado en los momentos más felices, y a todos mis compañeros y demás personas que me acompañaron durante esta etapa de mi vida.

-Ray Stevenson Amaya Maldonado

Dedico este trabajo principalmente a Dios, por haberme dado la vida y permitirme el haber llegado a este momento tan importante en mi formación integral. A mi madre Luz Stella Jaimes y mi padre Omar Uribe Martinez por ser los pilares más importantes y demostrarme por siempre su cariño y apoyo incondicional. También a toda mi familia y amistades por haber sido mi apoyo a lo largo de toda mi carrera universitaria y a lo largo de mi vida. A todas las personas especiales que me acompañaron en esta etapa, aportando a mi formación tanto profesional y como ser humano.

-Santiago Uribe Jaimes

AGRADECIMIENTOS

Nuestra gratitud a nuestra directora MsC. Jessica Gissella Maradey por su dedicación, orientación en el camino y apoyo proporcionado durante el desarrollo de este proyecto.

Agradecemos a nuestro codirector PhD. José Ignacio Huertas por aportarnos sus conocimientos en el área y por su orientación durante la ejecución del proyecto.

Gracias a la Red Latinoamericana de Investigación en Energía y vehículos (RELIEVE) y al Tecnológico de Monterrey por su vinculación y aporte para los instrumentos de medición utilizados.

Agradecemos a nuestro compañero José Miguel Mendoza por su amable colaboración y acompañamiento con la motocicleta durante el monitoreo de las pruebas de ruta, gracias por su tiempo y paciencia recorriendo la ciudad.

Agradecemos al MsC. Sergio Andrés Ardila por su orientación para el desarrollo de la calibración del flujómetro con el fluido de interés.

Agradecemos al MsC. Oscar Sebastián Serrano por su colaboración en la creación de las gráficas del diagrama de frecuencia de aceleración y velocidad (SAFD) y las diferentes herramientas para el análisis de los datos recopilados durante las pruebas dinámicas.

-Ray Stevenson y Santiago.

CONTENIDO

	Pág
1. INTRODUCCIÓN.....	15
2. ESTADO DEL ARTE	19
3. MARCO TEÓRICO	22
3.1. FUNDAMENTOS SOBRE EL FUNCIONAMIENTO DE LAS MOTOCICLETAS	22
3.1.1. Sistema de dosificación de combustible.....	23
3.1.1.1. Dosificación por medio de Carburador.....	24
3.1.2. Subsistemas presentes en una motocicleta	26
3.1.3. ¿Cómo medir el consumo de combustible de una motocicleta?	26
3.2. ¿CÓMO SE PUEDE MEDIR LAS REVOLUCIONES DEL MOTOR DE UNA MOTOCICLETA?.....	32
3.3. ¿CÓMO MEDIR LA VELOCIDAD DE LA MOTOCICLETA DURANTE UN RECORRIDO?.....	35
3.3.1. Precisión y normativa legal	37
3.4. ¿CÓMO TOMAR LA POSICIÓN DE LA MOTOCICLETA DURANTE UN RECORRIDO?.....	38
3.5. ¿CÓMO ALMACENAR Y PROCESAR LA INFORMACIÓN RECOLECTADA DE LA MOTOCICLETA DURANTE UN RECORRIDO?	39
3.5.1. Tarjeta de desarrollo NodeMCU ESP8266.....	41
3.5.2. Plataformas IoT.....	42
3.5.3. ¿Cómo funciona Blynk?	44
3.5.4. Arquitectura de Blynk	44
3.5.5. Protocolo de comunicación – Blynk.....	45
3.5.6. Almacenamiento de datos – Protocolo HTTPS	46
3.5.7. Google Apps Script	46
4. OBJETIVOS.....	47
4.1. OBJETIVO GENERAL.....	47
4.2. OBJETIVOS ESPECÍFICOS.....	47
5. METODOLOGÍA Y DESARROLLOS.....	48
5.1. SELECCIÓN DE ALTERNATIVAS:	48
5.2. MEDICIÓN DE CONSUMO DE COMBUSTIBLE.....	50
5.2.1. Calibración del Flujómetro para Gasolina Convencional.....	52
5.2.2. Instalación del flujómetro en línea.....	68

5.2.3.	Uniones hidráulicas	68
5.2.4.	Ubicación	69
5.2.5.	Conexiones eléctricas	70
5.2.6.	Código fuente	71
5.3.	MEDICIÓN DE VELOCIDAD ANGULAR DEL MOTOR	71
5.3.1.	Caracterización de señal:	72
5.3.2.	Diseño de Circuito Electrónico:	72
5.3.3.	Conexiones eléctricas	77
5.3.4.	Código fuente	78
5.4.	MEDICIÓN DE POSICIÓN Y VELOCIDAD	79
5.4.1.	Conexiones eléctricas	79
5.4.2.	Código fuente	80
5.5.	DESARROLLO DE LA APLICACIÓN MÓVIL PARA VISUALIZACIÓN	81
5.5.1.	Interfaz	81
5.5.2.	Código fuente	84
5.6.	DESARROLLO DE BASE DE DATOS EN LA NUBE	86
5.6.1.	Configuración Google Apps Script	87
5.6.2.	Configuración NodeMCU ESP8266	90
5.7.	DISEÑO DE TARJETA DE CIRCUITO IMPRESO PCB	91
5.8.	DISEÑO DE LA CARCASA DEL DISPOSITIVO	93
5.8.1.	Dimensiones y Ubicación	93
5.8.2.	Materiales	95
5.9.	INTEGRACIÓN DE LOS SUBSISTEMAS	96
5.9.1.	Esquema eléctrico	96
6.	RESULTADOS Y EVIDENCIAS	99
6.1.	CONECTIVIDAD	99
6.2.	PRUEBAS ESTÁTICAS	101
6.2.1.	Consumo de Combustible	101
6.2.2.	Velocidad Angular del Motor	102
6.2.3.	Posición y Velocidad	103
6.2.4.	Sistema Integrado	106
6.3.	PRUEBAS DINÁMICAS	108
6.3.1.	Sistema Integrado	108

6.4.	PROCEDIMIENTO DE INSTALACIÓN, ENCENDIDO Y APAGADO	111
6.4.1.	Procedimiento de Puesta en Servicio del Sistema	112
6.4.2.	Procedimiento de desconexión del sistema	113
6.5.	RESULTADOS DE LAS PRUEBAS DINÁMICAS	114
6.5.1.	Ruta Cañaveral – Ciudadela Real de Minas	115
6.5.2.	Ruta Ciudadela Real de Minas - Estoraques	119
7.	CONCLUSIONES	123
8.	RECOMENDACIONES y trabajos futuros	125
9.	CRONOGRAMA	126
10.	RECURSOS	127
11.	REFERENCIAS	128
12.	ANEXOS.....	134

LISTA DE FIGURAS

	Pag
Figura 1. Principio de funcionamiento del carburador.....	24
Figura 2. Componentes de un sistema de inyección electrónico	25
Figura 3. Características Yamaha YBR 125.	28
Figura 4. Sistema de encendido eléctrico en motocicleta	35
Figura 5. Sistema GPS	38
Figura 6. Interfaz Blynk App [Obtenida de https://blynk.io/	44
Figura 7. Arquitectura de Blynk.....	45
Figura 8. Metodología en V.....	48
Figura 9. Principio de funcionamiento del SLF3S-1300S.....	51
Tomada de Liquid flow concentration sensor SLF3C-1300F - more than flow: the thermal conductivity feature [35]	51
Figura 10. Principio de Funcionamiento del Sistema de Calibración	53
Figura 11. Sistema Husillo Tuerca	54
Figura 12. Delay vs Velocidad Angular.	54
Figura 13. Delay vs flujo de combustible	55
Figura 14. Velocidad angular vs flujo de combustible	55
Figura 15. Racores acoplados al flujómetro.....	56
Figura 16. Conexiones del sistema de bombeo.....	57
Figura 17. Asignamiento de pines.....	57
Figura 18. Conexiones del flujómetro con Arduino	58
Figura 19. Banco de calibración real.....	61
Figura 20. Error relativo porcentual vs velocidad angular real.	62
Figura 21. Flujo estimado vs flujo real.	65
Figura 22. Ajuste cuadrático.	66
Figura 23. Error relativo vs flujo de combustible real.	67

Figura 24. Dimensiones del SLF3-1300F en [mm].....	68
Figura 25. Diseño CAD del racor	69
Figura 26. Racor maquinado.....	69
Figura 27. CAD del SLF3S-1300F	70
Tomado de Sensirion [34].....	70
Figura 28. Esquema de conexiones eléctricas.	70
Figura 30. Esquemático de la señal de filtrado.	73
Figura 31. Esquemático de la etapa de filtrado y disparo.	75
Figura 32. Señal de pulsos y señal transformada.....	76
Figura 33. Esquemático de etapas de filtrado, disparo y regulación de la señal. ..	77
Figura 34. Diagrama electrónico de conexiones del circuito adaptador de señal de pulsos.	78
Figura 35. Conexiones eléctricas para el GPS.	79
Figura 36. Registro en la plataforma de Blynk.	81
Figura 37. Registro de Dispositivos en el entorno de Blynk.....	82
Figura 38. Widgets añadidos a la aplicación móvil	83
Figura 39. Configuración de los Widgets	83
Figura 40. Introducción de Credenciales de Red.....	85
Figura 46. Base de Datos en Google Sheets.....	87
Figura 47. Configuración de Google Apps Script.....	87
Figura 48. Spreadsheet ID.	88
Figura 49. Implementación del programa de Google Apps Script.....	89
Figura 50. ID de la implementación.	90
Figura 51. Esquema de conexiones en la Placa de Circuito Impreso PCB.....	91
Figura 52. Representación 3D de la Placa de Circuito Impreso PCB.	92
Figura 53. Placa de Circuito Impreso.....	92
Figura 54. Modelo 3D explosionado del sistema integrado [Ver anexo 11].	94
Figura 55. Modelo 3D de la carcasa del dispositivo.....	94
Figura 56. Impresión 3D de la carcasa del dispositivo.....	96
Figura 57. Esquema eléctrico de alimentación de energía del sistema.	97

Figura 58. Esquema de conexiones eléctricas del sistema.	97
Figura 59. Sistema Integrado.....	98
Figura 60. Esquema general de conexiones del sistema de monitoreo.....	98
Figura 61. Mapa de cobertura de red móvil de Claro Colombia en Bucaramanga.	100
Tomada de Nperf [38].	100
Figura 62. Tiempo de subida a la base de datos.	100
Figura 63. Consumo instantáneo de combustible visualizado en la aplicación móvil.	101
Figura 64. Velocidad angular calculada vs. Velocidad angular real en rpm.	102
Figura 65. Error relativo vs. Velocidad angular real.	102
Figura 66. Validación de velocidad angular del motor en rpm.	103
Figura 67. Validación de posición real con el sistema integrado.....	104
Figura 68. Validación de velocidad con la herramienta GPS Speedometer.....	104
Figura 69. Velocidad real vs Velocidad estimada.	105
Figura 70. Error relativo de velocidad vs Velocidad real.	105
Figura 71. Validación del sistema integrado en pruebas estacionarias.	106
Figura 72. Validación de datos recolectados en la aplicación durante pruebas estáticas.....	106
Figura 73. Resultados de subida de datos a la base de datos en Google Sheets.	107
Figura 74. Ubicación del sistema de monitoreo en la motocicleta	111
Figura 75. Procedimiento de puesta en marcha del sistema de monitoreo [ver anexo 12].....	112
Figura 76. Procedimiento de desconexión del sistema de monitoreo [Ver anexo 13].	113
Figura 77. Aplicación de visualización durante pruebas dinámicas.	115
Figura 78. Ruta monitoreada Cañaveral – Ciudadela Real de Minas en Google Maps.....	116

Figura 79. Curvas de velocidad y aceleración en ruta Cañaveral-Real de minas.	116
Figura 80. Curvas de velocidad angular y velocidad en ruta Cañaveral-Real de minas.	117
Figura 81. Curvas de altitud y pendiente en ruta Cañaveral-Real de minas.	117
Figura 82. Curvas de velocidad angular y Consumo de combustible en ruta Cañaveral-Real de minas.	118
Figura 83. SAFD de la ruta Cañaveral-Real de minas..	118
Figura 84. Ruta monitoreada Real de Minas – Estoraques en Google Maps.	119
Figura 85. Curvas de velocidad y aceleración en ruta Real de minas - Estoraques.	120
Figura 86. Curvas de velocidad angular y velocidad en ruta Real de minas - Estoraques.....	120
Figura 87. Curvas de altitud y pendiente en ruta Real de minas - Estoraques. ..	121
Figura 88. Curvas de velocidad angular y consumo de combustible en ruta Real de minas - Estoraques.	121
Figura 89. Curvas de velocidad y aceleración en ruta Real de minas - Estoraques.	122

LISTA DE TABLAS

Pág

Tabla 1. Parque automotor en Colombia entre los años 2002 al 2016.	16
Tabla 2. Parque automotor agosto de 2021 en Bucaramanga, Colombia	17
Tabla 3. Emisiones [Ton] año 2016 en Colombia.	18
Tabla 4. Resumen de proyectos de referencia.	21
Tabla 5. Cálculo del consumo instantáneo de combustible promedio de las motocicletas carburadas más vendidas en el 2020 en Colombia.	29
Tabla 6. Alternativas de solución para medición de consumo instantáneo de combustible.....	30
Tabla 6. Continuación	31
Tabla 6. Continuación	32
Tabla 7. Alternativas de solución para medición de velocidad angular.....	34
Tabla 8. Alternativas de solución para medición de velocidad.....	36
Tabla 8. Continuación.....	37
Tabla 9. Alternativas de solución para la selección de la unidad de procesamiento.	40
Tabla 9. Continuación.....	41
Tabla 10. Alternativas de solución de Plataformas IoT	43
Tabla 11. Estructura de protocolo de comunicación Blynk	45
Tabla 12. Matriz QFD.....	49
Tabla 12. Continuación.....	49
Tabla 12. Continuación.....	50
Tabla 12. Continuación.....	50
Tabla 13. Puntajes ponderados para selección de alternativa de solución para medición de consumo de combustible.....	51
Tabla 14. Incertidumbre combinada.....	63
Tabla 15. Continuación.....	65
Tabla 16. Comparativa de volumen	66

LISTA DE ANEXOS

	Pág
Anexo 1. Código Arduino Sistema Bombeo	134
Anexo 2. Código Arduino Flujómetro	136
Anexo 3. Código Google Script.....	139
Anexo 4. Código NodeMCU.....	141
Anexo 5. Diagrama Electrónico MOBD-IoT	148
Anexo 6. Layout banco de calibración	149
Anexo 7. Plano de racor	150
Anexo 8. Plano de caja	151
Anexo 9. Plano de tapa.....	152
Anexo 10. Plano de botón.....	153
Anexo 11. Plano general explosionado.....	154
Anexo 12. Procedimiento de puesta en marcha del sistema	155
Anexo 13. Procedimiento de desconexión del sistema.....	156

1. INTRODUCCIÓN

Las variables implícitas del ciclo de conducción se encuentran estrechamente relacionadas con la cantidad de emisiones que producen los vehículos, en el caso de los automóviles estos se encuentran equipados con una serie de sensores de presión, de caudal, de temperatura e instrumentación que permite medir la velocidad del vehículo y las revoluciones del motor, inclusive en los últimos años estos vehículos poseen módulos GPS que permiten recolectar los datos de altimetría y posición en tiempo real. No obstante, las motocicletas de cilindradas menores a 250 cc suelen utilizar carburador y no poseen toda esta instrumentación puesto que elevaría el coste de fabricación, y por consecuencia su precio de venta lo cual no es factible debido a que uno de los factores por los cuales muchas personas eligen las motocicletas es por su costo reducido respecto a otros vehículos como motocicletas de mayores cilindradas o automóviles. Esta dificultad en las motocicletas más nuevas del mercado y de cilindradas mayores no se presenta, sin embargo, la mayor parte de las motocicletas en países como Colombia no poseen estos sensores y existen serias dificultades al estimar las emisiones que puedan llevar al desarrollo de medidas que permitan tener un mayor control de la contaminación presente.

En Colombia existe un amplio sector en el mercado que pertenece a las motocicletas, según cifras del parque automotor [3] [4] para el año 2017 existían 7.964.558 motocicletas correspondientes al 57,6% del parque móvil, elevándose para junio del año 2021 a 9.734.208 unidades siendo esto el 59% de todos los vehículos registrados en el RUNT. Para el caso de la ciudad de Bucaramanga, en abril de 2021 había en circulación 462.996 motocicletas abarcando el 60.8% según cifras de la dirección de tránsito de Bucaramanga [5][6]. Sin embargo, los estándares legales vigentes en Colombia exigen que las motocicletas importadas o ensambladas en el país a partir de enero de 2021 cumplan con el estándar Euro 3 [7], a pesar de ello las motocicletas más comercializadas en el país en el año 2020 [8] sugieren que ocho de las primeras 10 son motocicletas carburadas, lo cual acarrea problemas para poder controlar las emisiones ya que estos motores tienen mayores complicaciones para obtener la mezcla estequiométrica exacta para mayor eficiencia.

Tabla 1. Parque automotor en Colombia entre los años 2002 al 2016.

Año	Automóvil y otros		Motocicleta		Total parque automotor
	Parque automóviles,	Participación	Parque motocicleta	Participación	
2002	2,229,047	57.6%	1,276,481	32.7%	3,892,273
2003	2,316,458	57.2%	1,354,860	33.0%	4,068,725
2004	2,401,393	56.6%	1,454,860	33.5%	4,245,177
2005	2,523,796	55.4%	1,626,813	34.3%	4,553,756
2006	2,670,231	52.7%	1,978,982	35.7%	5,071,465
2007	2,873,081	50.2%	2,390,479	39.0%	5,721,027
2008	3,108,512	47.9%	2,886,935	41.8%	6,483,291
2009	3,293,214	46.7%	3,245,643	44.5%	7,047,322
2010	3,459,018	45.8%	3,572,133	46.1%	7,553,765
2011	3,731,688	45.0%	4,017,404	47.3%	8,297,001
2012	4,021,978	43.9%	4,542,925	48.4%	9,156,984
2013	4,209,594	43.2%	4,911,347	49.6%	9,737,987
2014	4,596,355	36.9%	5,835,944	53.8%	11,108,298
2015	4,816,788	40.3%	6,684,836	55.8%	11,963,838
2016	4,962,265	38.4%	7,251,297	56.2%	12,909,738

Tomado de ANDI [7.]

La motocicleta es el medio de transporte preferido por trabajadores y estudiantes debido a su fácil accesibilidad puerta a puerta y su costo económico. Además, es un vehículo atractivo para distancias cortas de viaje en ciudades congestionadas. Por el contrario, la motocicleta trae consigo algunos problemas como la alta participación en el tráfico, participación en accidentes, un índice mayor de violación en las normas de tránsito y especialmente alta emisión de contaminación del aire en las ciudades congestionadas como es el caso del proyecto en Bucaramanga, Colombia [4]. En la actualidad se han presentado nuevas tecnologías en la planeación y construcción de motores, así como en el desarrollo de energías alternativas como es el caso de motocicletas eléctricas con el fin de disminuir las emisiones.

En los últimos diez años más del 50 por ciento del parque automotor en Colombia, como se observa en la Tabla 1, ha sido dominado por las motocicletas comparado con vehículos del tipo familiar, pues se ve superado por más del 20 por ciento; El anterior efecto se presenta de la misma forma como se puede ver en la Tabla 2, en la ciudad de Bucaramanga donde para abril del año 2021, del total de 760.747 vehículos, 462.996 representan el uso de motocicletas, a partir de la información dada se podría dimensionar la cantidad de emisiones de gases producidas por este tipo de vehículo, esta información se presenta en la Tabla 3, cabe resaltar que son las de mayor producción de CO y HC con un 36% y 30% respectivamente.

Tabla 2. Parque automotor agosto de 2021 en Bucaramanga, Colombia

Vehículo	Cantidad
Motocicletas	492,996
Automóviles	162,031
Camionetas	65,090
Camperos	23,519
Camiones	16,685
Buses:	3,582
Busetas	3,570
Otros (cuatrimoto, motocarro, tractocamión, volqueta):	23,273

Tomado de Vanguardia [4].

Teniendo en cuenta el alto grado de porcentaje de uso del tipo de vehículo mencionado en los párrafos anteriores en el parque automotor de Colombia, se plantea la siguiente pregunta ¿Existe un dispositivo masivo capaz de medir el consumo de combustible bajo condiciones reales de operación en las motocicletas? Actualmente no hay un dispositivo desarrollado de bajo costo en Colombia para la medición de combustible real en motocicletas, ya que el tipo de motor de la mayoría se alimenta de combustible por medio de carburador y no de inyección electrónica, lo que representa un reto directo al momento de poder instrumentar estos motores para la medición exacta del combustible utilizado. Otro motivo por el cual tampoco se ha desarrollado un dispositivo de este estilo se debe a que el cilindraje de estas motocicletas es menor a los 250 cc, con lo cual el consumo de combustible pasa desapercibido para los dueños de estos vehículos, creando poca conciencia de ahorro y cuidado del medio ambiente en cuanto a emisiones de gases contaminantes.

Tabla 3. Emisiones [Ton] año 2016 en Colombia.

	PARQUE 2016	EMISIONES TONELADAS AÑO				
		C02	CO	NOX	HC	MP 2,5
	12,302,000	61,228,000	3,312,000	556,000	393,000	10,000
Automóvil	23%	12%	23%	5%	24%	0%
Taxi	3%	11%	7%	11%	11%	0%
Campero y camioneta	11%	16%	31%	10%	28%	0%
Carga	5%	42%	2%	59%	6%	80%
Pax	1%	10%	0%	12%	2%	20%
Motos	57%	8%	36%	2%	30%	0%
Hasta 20 años	79%	71%	57%	63%	53%	30%
>20 años	21%	29%	43%	37%	47%	70%

Tomado de ANDEMOS [19].

Adicionalmente, no existe en el mercado un sistema instrumental capaz de recolectar los datos de los parámetros implícitos de una motocicleta, que permita la caracterización en campo de estilos de conducción y la evaluación de consumo y emisiones, pues estas mediciones se realizan habitualmente en laboratorios equipados con dinamómetros de chasis lo cual entrega resultados por lo menos un 60% menores que en condiciones reales de carretera según estudios recientes [1][2].

La motivación del proyecto se da con lo anterior descrito, planteando así el diseño y construcción de un dispositivo semi-industrial el cual se pueda llevar a bordo en motocicletas carburadas, permitiendo medir variables como lo son la posición, velocidad, RPM, consumo de combustible durante tiempos de conducción prolongados en carretera.

2. ESTADO DEL ARTE

El estudio de los ciclos de conducción es una tarea útil para poder analizar el impacto ambiental y económico que deja en una zona determinada el uso de vehículos y motocicletas, pues el consumo de combustible y la velocidad del vehículo están estrechamente relacionados con estos factores previamente mencionados, Tzeng y Cheng [9] en 1998 recolectaron y analizaron los datos provenientes de cuarenta y cinco motocicletas en la ciudad de Taipei utilizando velocímetros magnéticos para determinar la velocidad, además del tiempo a bordo y el origen y destino del recorrido. El caso de la detección de los parámetros implícitos del ciclo de conducción en motocicletas no es una tarea sencilla debido a que un gran número de estos vehículos no poseen los sensores pertinentes, Chen et al [10] en 2003 desarrollaron un sistema capaz de simular las condiciones de carretera tanto en ciudad como en zona rural en Taiwan y recopilar los datos de consumo de combustible, velocidad promedio y emisiones utilizando un dinamómetro de chasis, encontrando un consumo aproximadamente un 30% mayor en las ciudades respecto a las zonas rurales. Aunque un gran número de estos vehículos no cuentan con esta instrumentación, algunos actualmente presentan dispositivos que registran a bordo la posición y velocidad del vehículo mediante GPS, así como datos del estado del motor en tiempo real. Estos datos incluyen las RPM, el kilometraje, posición del pedal del acelerador, productos químicos y el consumo instantáneo de combustible. Beusen et al [11] en 2009 recopilaron datos durante 10 meses para 10 conductores en condiciones reales permitiendo realizar un análisis del estilo de conducción individual y posteriormente una evaluación el impacto del curso de eco-driving que educa hacia una conducción más eficiente y de esta manera fomentar la reducción de combustible y emisión de gases contaminantes.

El estilo de conducción puede dividirse característicamente en dos categorías: el no agresivo y el agresivo. Entender y reconocer los eventos de conducción que entran en dichas categorías pueden ayudar a los mejorar los sistemas de seguridad de los vehículos, así como estudiar e implementar estrategias de reducción de consumo y emisiones. El comportamiento agresivo de conducción es actualmente una de las principales causas de accidentes de tráfico en Estados Unidos. Johnson et al [12] en 2011 proponen un novedoso sistema que utiliza la deformación temporal dinámica del tiempo (DTW) y una fusión de sensores basada en los sensores disponibles en los smartphones. El método que utilizan se diferencia sobre el reconocimiento de los patrones de conducción al fusionar datos y realizar el proceso completamente en un smartphone.

El sistema de inyección ha ido evolucionando y perfeccionándose con el paso de los años por lo que actualmente existen una variedad o modalidades como es la inyección electrónica o fuel injection, el sistema multipunto o la inyección directa,

entre otras. Además, debido a la existencia de motocicletas carburadas que aun en su mayoría circulan, hace que sea una necesidad actualizar su sistema de dosificación de combustible por un control electrónico para obtener resultados eficientes y un reducir el impacto ambiental. De esta manera Arévalo S. et al [13] en 2014 diseñaron e implementaron un sistema de inyección electrónico de combustible para un motor de combustión interna de motocicleta logrando demostrar de manera practica la mejora del rendimiento de combustible con el sistema creado, llegando a un 50% de ahorro de gasolina, a pesar de que éste podría variar dependiendo de otras condiciones. La medición de combustible fue llevada a cabo midiendo la diferencia de combustible al finalizar los recorridos en motocicleta, adicionalmente utilizaron los pulsos entregados a la ECU cada vez que ocurre una revolución del eje del motor para medir las RPM y un sensor de posición para determinar la ruta.

El desarrollo de dispositivos de medición a bordo de las motocicletas es una estrategia utilizada para poder estimar con precisión el consumo de combustible y el impacto ambiental que involucra este tipo de vehículos en la vida cotidiana, Satiennam et al [14] en el 2017, diseñaron un sistema a bordo capaz de recolectar los datos de consumo de combustible en una motocicleta de inyección por carburador utilizando el flujómetro eléctrico SLQ-HC60, así como un módulo GPS, un analizador de gases de exhosto y un sensor de velocidad en la rueda trasera. Demostrando una alta correlación entre la velocidad instantánea y la tasa de aceleración con el consumo de combustible y las emisiones de CO₂, HC y NO_x. También demuestra que soluciones como zonas exclusivas para motocicletas podría mejorar el flujo del tráfico y por consiguiente una reducción en el combustible utilizado y el impacto ambiental.

La medición del consumo de combustible usando modelos de laboratorio exige la medición de múltiples datos en condiciones adecuadas para obtener buenas predicciones del comportamiento en el mundo real, por lo tanto, se hace necesaria la creación de un dispositivo móvil capaz de tomar muestras en pista reales. Nguyen et al [15] en el 2020 desarrollaron un sistema capaz de recolectar los datos en tiempo real de consumo de combustible y velocidad con una tasa de refresco de 1 Hz en una motocicleta de inyección electrónica. El sistema utilizó los pulsos generados en la UCE por cada inyección de combustible para estimar el consumo en lugar de un flujómetro, consiguiendo el desarrollo del dispositivo capaz de recolectar los datos y almacenarlos permitiendo el estudio de estos para la caracterización de estilos de conducción en condiciones reales.

La interconexión entre los vehículos y las tecnologías de recolección de datos para conocer los estilos de conducción son una tendencia que crece con el paso del tiempo, entre los principales retos se encuentra el uso de datos y la privacidad de estos. Kaiser et al [16] desarrollaron en 2020 una propuesta de sistema que es capaz de recolectar los datos necesarios para la caracterización de estilos de conducción y subirlos a la nube de manera segura dividido en varios niveles de

privacidad como los datos de uso público, uso limitado, uso anónimo y uso privado y desarrollaron una aplicación móvil capaz de mostrar al usuario la información de su estilo de conducción y almacenarlos en la nube.

Tabla 4. Resumen de proyectos de referencia.

Autores	Año	País	Técnica de desarrollo	Principales contribuciones
Tzeng y Cheng	1998	Taiwan	Medición de parámetros en laboratorio.	Técnicas para medición de velocidad y posición en motocicletas.
Chen et al.	2003	Taiwan	Medición de parámetros en entorno simulado.	Técnica de medición de velocidad con sensores magnéticos en las ruedas.
Beusen et al.	2009	Bélgica	Recopilación de datos utilizando sensores propios del vehículo.	Estrategia de recopilación de datos para estudiar el estilo de conducción a bordo de un vehículo.
Jhonson et al.	2011	Estados Unidos	Tratamiento de datos usando un smartphone.	Modelo de recopilación de datos utilizando algoritmos con los sensores de un smartphone.
Arévalo et al.	2014	Ecuador	Medición de parámetros en condiciones reales.	Sistema de recolección de parámetros como RPM, gases de escape, consumo de combustible y posición.
Satiennam et al.	2017	Tailandia	Medición de parámetros en condiciones reales.	Sistema de monitoreo utilizando un flujómetro para medición de combustible, módulo GPS, y sensor de velocidad en la rueda delantera.
Nguyen et al.	2020	Vietnam	Medición, recolección y almacenamiento de parámetros.	Sistema de medición de velocidad, consumo de combustible en tiempo real en una motocicleta de inyección electrónica y tratamiento de datos usando plataformas móviles.
Kaiser et al.	2020	República Checa	Desarrollo de un sistema telemático para mostrar los datos de estilos de conducción	Modelo de desarrollo de una aplicación móvil que brinde seguridad para el tratamiento de datos.

3. MARCO TEÓRICO

Los estilos de conducción se suelen monitorear teniendo en cuenta diferentes parámetros que describen diferentes funcionalidades del vehículo, entre estos parámetros de interés se encuentran el consumo instantáneo de combustible, la velocidad del vehículo, la velocidad angular del motor y la posición para poder rastrear la ruta recorrida.

Normalmente estas mediciones para automóviles se pueden realizar aprovechando los sensores que incluye el mismo vehículo y cuyos datos se encuentran accediendo a la ECU, sin embargo, para las motocicletas más comerciales en Colombia, como lo son las motocicletas carburadas, la recolección de estos datos debe realizarse instalando un sistema instrumentado especial para ello. El reto del presente proyecto es el diseño y la implementación de un sistema instrumentado que pueda ser sencillo, asequible y al alcance de los usuarios con precios reducidos.

3.1. FUNDAMENTOS SOBRE EL FUNCIONAMIENTO DE LAS MOTOCICLETAS

El propósito de los motores de combustión interna es la producción de potencia mecánica a partir de la energía química contenida en el combustible, a diferencia de los motores de combustión externa, la energía es obtenida oxidando en combustible al interior de una recámara [17]. El motor de combustión interna está presente en vehículos y diversas máquinas y usa combustibles como gasolina, diésel, etanol, gas, entre otros.

Conocer el funcionamiento de los motores de combustión interna es parte fundamental al momento de realizar la medición de combustible consumido por estos durante su operación. Los motores presentes en las motocicletas más recientes suelen ser motores de cuatro tiempos, esto indica que existe un ciclo que consta de cuatro momentos en los cuales ocurre una determinada acción, estos son:

- Tiempo de admisión: La mezcla de aire y combustible ingresan a la cámara de combustión por medio de la válvula de admisión.
- Tiempo de compresión: La mezcla se comprime manteniendo las válvulas de admisión y escape cerradas.
- Tiempo de combustión: Este tiempo consta de dos partes, la explosión y la expansión. La explosión es iniciada por la chispa eléctrica de la bujía, esta

explosión genera una gran presión sobre el cilindro el cual desciende y genera el trabajo del motor.

- Tiempo de escape: Aquí los gases residuales son expulsados del motor a través de la válvula de escape.

Los motores también suelen clasificarse según el combustible que utilizan, para el caso de vehículos pesados, aplicaciones navales y transporte terrestre se utilizan motores diésel debido a que genera unas altas temperaturas derivado de la alta compresión que posee, según el ciclo diésel. Por su parte en las motocicletas se utilizan motores de gasolina, estos motores se componen de: Bloque, cigüeñal, biela, pistón, culata, válvulas y aunque existen hoy en día distintas maneras de realizar la inyección de combustible utilizando microelectrónica para mejorar la eficiencia del motor. Las motocicletas más vendidas en Colombia para el año 2021 aún utilizan carburador como sistema de inyección de combustible y la inyección electrónica de combustible está relegada a motocicletas de alta cilindrada [3].

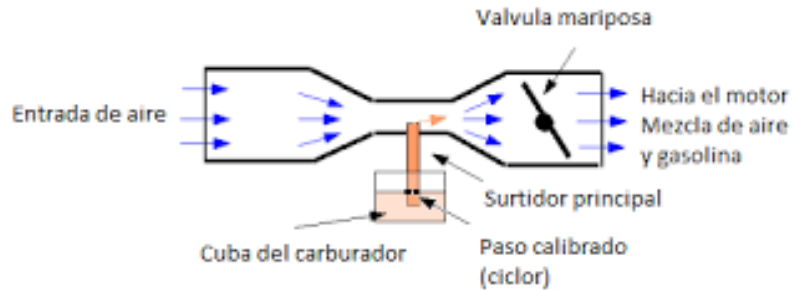
3.1.1. Sistema de dosificación de combustible

Los motores de combustión interna se componen de dos subsistemas encargados de dos tareas específicas, el primero de ellos es la cámara de combustión, este es un cilindro fijo, cerrado por el extremo superior con la culata y al interior de ella se desliza el pistón; el segundo es el sistema de alimentación el cual consta de un depósito o tanque, una bomba de combustible y un dispositivo que dosifica el combustible y lo vaporiza en las proporciones adecuadas, las cuales para el caso de la gasolina es de 1 gramo de gasolina por cada 14,7 gramos de aire, esta proporción es denominada mezcla estequiométrica.

3.1.1.1. Dosificación por medio de Carburador

Este sistema aprovecha el descenso del pistón en la fase de admisión, en la cual aspira aire hacia el interior del cilindro debido a la diferencia de presión que se genera.

Figura 1. Principio de funcionamiento del carburador.



Tomado de *Internal Combustion Engine Handbook* [26].

El aire, al pasar por el carburador, absorbe una pequeña cantidad del combustible que proviene de la cuba, la cual posee una válvula que permite el ingreso de combustible al carburador procurando siempre tener un mismo nivel de líquido que permita la presencia de combustible para atomizar. El combustible recorre la manguera de admisión que conecta la válvula de admisión a la salida del depósito de combustible con el carburador, se almacena en la cuba y se surte hacia la cámara de combustión. Esta cantidad de combustible ocasionalmente no llega a ser exactamente la mezcla estequiométrica y para tomar esta lectura, se debe utilizar como apoyo la tabla estequiométrica, que indica la apertura del tornillo de mezcla.

Con este sistema de alimentación la medición del combustible puede realizarse precisamente en la manguera de admisión que conecta el depósito con el carburador [14].

3.1.1.2. Dosificación por medio de inyección electrónica

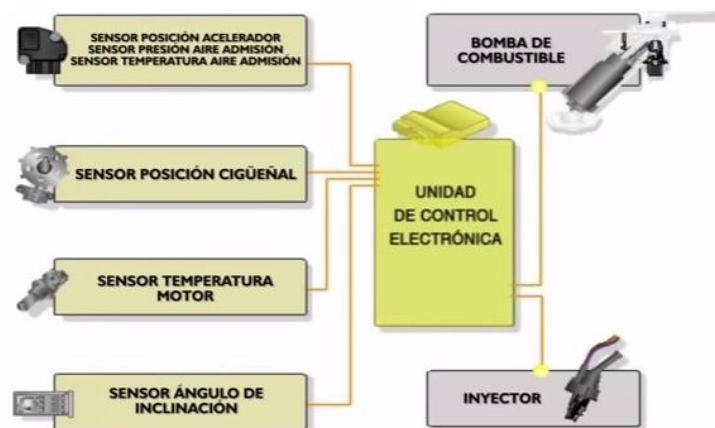
En este sistema, el aire y la gasolina siguen caminos independientes, el sistema consta de tres partes fundamentales: la primera un grupo de sensores para conocer los valores de todos los parámetros que influyen en la producción de la mezcla; la segunda, la unidad electrónica que decide la cantidad e incluso el momento óptimo para la inyección de la gasolina, la tercera y la última está formada por los elementos que directamente realizan el aporte de carburante.

Los sensores presentes en un sistema de inyección electrónica son:

- Sensor MAP: Permite medir la presión de aire en el conducto de admisión.
- Sensor de posición de mariposa o TPS: Indica la posición del acelerador.
- Sensor de temperatura del aire: Indica la temperatura del aire que ingresa por la admisión
- Sensor de temperatura del motor: Informa a la computadora el estado del motor (frío o caliente) para compensar la mezcla.
- Sensor RPM: Indica el punto muerto superior del motor y el régimen en el cual este se encuentra.
- Sonda de oxígeno o sonda lambda: Mide la cantidad de oxígeno en los gases de escape para comprobar permanentemente y compensar la cantidad de combustible utilizado.
- Sensor de ángulo de inclinación.

El aire aspirado en la admisión es filtrado y medido, de acuerdo con esto y con los datos recopilados de los sensores previamente descritos se inyecta gasolina pulverizada a presión constante, abriendo el inyector electromagnético el tiempo necesario en milisegundos, el cual es calculado por la unidad central o ECU.

Figura 2. Componentes de un sistema de inyección electrónico



Tomado de Revista Moto [27].

3.1.2. Subsistemas presentes en una motocicleta

La motocicleta es una máquina que se compone de diferentes subsistemas que integrados cumplen el objetivo principal, el cual es transportar. Estos subsistemas son:

- Sistema de distribución: Aquí cada cilindro toma el combustible y expulsa los gases residuales por las válvulas del cabezal o válvulas deslizantes.
- Sistema de encendido: Los motores inician el proceso de combustión dentro de los cilindros por medio de un autotransformador de alta tensión que está sincronizado con el tiempo de compresión en cada uno de los cilindros. La chispa al interior de la cámara es iniciada por medio de las bujías, las cuales contienen unos electrodos ligeramente distanciados algunas décimas de milímetros y que al ocurrir el impulso eléctrico inflaman el combustible.
- Sistema de refrigeración: Este sistema varía según la aplicación del motor, pues para el caso de las motocicletas la refrigeración se realiza con aire, mientras que para las aplicaciones navales se realiza con agua marina y para los automóviles se utilizan refrigerantes diferentes al agua ya que no generar sedimentos ni daños en los empaques debido a que poseen un punto de ebullición mayor al del agua.
- Sistema de arranque: Debido a la resistencia cinemática propia de los componentes del motor, se necesita una fuente primigenia de movimiento en el cigüeñal, a diferencia de los motores a vapor, los motores de combustión interna no producen un par de torsión inicial, con lo cual se utilizan principalmente dos sistemas uno de ellos es la incorporación de un motor eléctrico que se desacopla del cigüeñal una vez arranca el motor, por otro lado existe el sistema de arranque manual girando el cigüeñal por medio de una cadena o un pedal como es el caso de algunas motocicletas.

3.1.3. ¿Cómo medir el consumo de combustible de una motocicleta?

Para la medición del combustible en motocicletas se debe hacer especial énfasis en el sistema de alimentación, para el caso de las motocicletas que utilizan sistema de inyección electrónica, la medición del consumo se puede hacer extrayendo de la ECU directamente esta información pues este sistema es mucho más preciso y conociendo los pulsos generados por cada descarga de combustible se puede conocer la cantidad consumida durante un intervalo de interés. Para el caso de las motocicletas que utilizan carburador, Satiennam et al [14] en el 2017, proponen la instalación del flujómetro SLQ-HC60 en la manguera que conecta el depósito de combustible con el carburador, esta solución permite una medición altamente

precisa y con una respuesta rápida. Otra propuesta de solución es realizada por Arévalo S. et al [13], quienes midieron la diferencia de combustible al conocer la cantidad de combustible al iniciar y al finalizar un recorrido determinado, si bien esta manera no asume una medición en tiempo real, es un método no invasivo que permite el funcionamiento normal de la motocicleta al no intervenir con el sistema de inyección.

Conociendo estos conceptos es posible plantear una serie de alternativas de solución basados en los siguientes criterios:

Las motos más vendidas en el 2020 en Colombia según el medio informativo El Carro Colombiano [8], son:

- Bajaj Boxer CT 100 AHO
- AKT AK125 NKDR
- Honda CB125F
- Yamaha XTZ125
- Yamaha GPD150-A (NMAX)
- Yamaha FZN150D-6 (FZ-S)
- Honda XR150L
- Victory One
- Honda CB 110
- Yamaha T115FI (T115FL-5)

De las cuales todas a excepción de la Yamaha GPD150-A, la Yamaha FZN150D-6 y la Yamaha T115FI utilizan carburador, por lo tanto, es de interés monitorear las variables de interés en el presente proyecto.

La velocidad de los vehículos en Colombia depende de diferentes factores como puede ser el momento del día hora pico u hora valle. Según Bonells et al [23], la motocicleta es el medio de transporte que alcanza una mayor velocidad llegando a ser en promedio una máxima de 43 km/h en recorridos medios de entre 5 y 10 km de longitud en zonas urbanas y en cifras globales llega a ser un 30% mayor que la de un automóvil.

Para fines de este proyecto se utiliza la motocicleta Yamaha YBR 125 debido a la disponibilidad inmediata, esta motocicleta es carburada y sus características son:

Figura 3. Características Yamaha YBR 125.

 YBR 125Z	
Motor	
Tipo	Monocilíndrico, 4T, SOHC, Refrigerado p/aire
Cilindrada	124 cc
Diámetro x carrera	54 x 54 mm
Relación de compresión	10.0:1
Lubricación	Cárter húmedo
Alimentación	Carburador
Embrague	Multidisco en baño de aceite
Encendido	CDI
Arranque	Eléctrico
Transmisión	5 velocidades
Transmisión final	Cadena
Chasis	
Suspensión delantera	Horquilla telescópica
Recorrido delantero	120 mm
Suspensión trasera	Basculante
Recorrido trasero	105 mm
Freno delantero	Disco simple de 245 mm x 4 mm
Freno trasero	Tambor de 130 mm
Neumático delantero	2.75-18 42P
Neumático trasero	90/90-18 57P
Dimensiones	
Longitud total	1970 mm
Ancho total	745 mm
Alto total	1090 mm
Distancia entre ejes	1290 mm
Distancia mínima del suelo	160 mm
Peso en orden de marcha (c/aceite y combustible)	121 Kg
Capacidad de combustible	13 Lts
Cap. depósito aceite (Cambio periódico)	1 Lt

Tomado de Yamaha [28].

3.1.3.1. Rango de medida

El rango de medida se refiere a los valores y la escala en la cual se encuentra la variable de interés para poder seleccionar un sensor adecuado para el proceso, en el caso del consumo de combustible este se puede calcular con el dato del consumo promedio de cada motocicleta de las más comerciales en Colombia para poder tener una referencia.

$$FP \left[\frac{ml}{min} \right] = \frac{V \left[\frac{km}{min} \right]}{CP \left[\frac{km}{ml} \right]} \quad (1)$$

FP = Flujo Promedio

V = Velocidad de Referencia (Promedio o máxima)

CP = Consumo de Combustible Promedio.

Las velocidades de referencia utilizadas son 43 km/h y 86 km/h tomando como base el estudio de Bonells et al [23].

Tabla 5. Cálculo del consumo instantáneo de combustible promedio de las motocicletas carburadas más vendidas en el 2020 en Colombia.

Medición de Consumo de Combustible			
Velocidad Media [km/h]	43	Velocidad Media [km/min]	0.72
Velocidad máxima media [km/h]	86	Velocidad máxima media [km/min]	1.43
Referencia	Consumo promedio [km/l]	Flujo de combustible promedio [ml/min]	Flujo máximo promedio de consumo [ml/min]
Bajaj Boxer CT 100	70	10.2	20.5
AKT AK125 NKD	33	21.6	43.2
Honda CB125F	30	23.6	47.2
Yamaha XTZ125	32	22.4	44.8
Honda XR150L	45	15.9	31.9
Victory One	53	13.6	27.1
Honda CB 110	60	11.9	23.9
Yamaha YBR 125	38	18.9	37.7
Promedio		17.3	34.5

De acuerdo con ese cálculo, el sensor debe permitir la medición de hasta por lo menos 80 ml/min para poder abarcar un rango de velocidades amplio, así como una mayor cantidad de pasajeros o una altura sobre el nivel del mar diferente de la altura de referencia para cada motocicleta.

Utilizando toda la información previamente descrita se desarrolla una tabla de alternativas posibles de solución.

Tabla 6. Alternativas de solución para medición de consumo instantáneo de combustible.

REFERENCIA	DESCRIPCIÓN	TECNOLOGÍA UTILIZADA	RANGO DE OPERACIÓN	INTERFAZ DE COMUNICACIÓN
Flujómetro de mililitros para hidrocarburos SLQ-HC60 [41]	Sensor de hidrocarburos no invasivo montado en línea, con respuesta inferior a 50 ms y sensible a flujos bajos	Sensor térmico de alta precisión con compensación de temperatura y linearización por chip interno.	0-80 ml/min	Analog 0-5V
Flujómetro microfluídico de medios aislados SLI-2000 FMK [42]	Sensor de flujo no invasivo, montado en línea con rango de mediciones desde ul hasta ml y tiempo de respuesta de 40 ms.	Sensor térmico de alta precisión con compensación de temperatura y linearización por chip interno.	0-80 ml/min	I2C, RS-485, USB
Flujómetro compacto para bajas tasas de flujo de líquido LS32-1500 [43]	Sensor de flujo mecánicamente robusto no invasivo, montado en línea con rango de mediciones desde en el rango de los mililitros y tiempo de respuesta de 40 ms bidireccional.	Sensor de tubos rectos alineados.	0-40 ml/min	I2C

Tabla 6. Continuación

REFERENCIA	DESCRIPCIÓN	TECNOLOGÍA UTILIZADA	RANGO DE OPERACIÓN	INTERFAZ DE COMUNICACIÓN
Flujómetro compacto para aplicaciones de alto volumen SLF3S-1300F [34]	Sensor de flujo compacto no invasivo montado en línea con respuesta menor a 20 ms y flujo de hasta 40 ml/min y kit de damping para suprimir vibraciones del medio.	Sensor térmico de alta precisión con compensación de temperatura.	0-40 ml/min	I2C
Flujómetro de líquido compacto para aplicaciones OEM LG16-100D [44]	Sensor de flujo no invasivo, montado en línea con rango de mediciones de hasta 80 ml/min para hidrocarburos y resolución del orden de los nanómetros, bidireccional	Sensor capilar de vidrio recto y accesorios terminales. No hay obstrucciones en la ruta del flujo.	0-80 ml/min	I2C
Flujómetro de turbina Digital Mini NPT VBESTLIFE Turbine Flowmeter [45]	Sensor de diseño modular, alta precisión, tamaño compacto y rendimiento estable. Fácil de reemplazar y mantener los componentes internos.	Sensor proporcional a la velocidad de giro de la turbina de sensado.	15-120 L/min	Analog 0-5V

Tabla 6. Continuación

REFERENCIA	DESCRIPCIÓN	TECNOLOGÍA UTILIZADA	RANGO DE OPERACIÓN	INTERFAZ DE COMUNICACIÓN
Medidor Ultrasónico de Flujo de Combustible de Bajo Flujo FlowSonic® LF [46]	Está diseñado para las condiciones de flujo de combustible de volumen ultra bajo y adaptado para las vibraciones que se encuentran en los motores de automóviles de carretera de alta eficiencia de hoy en día y	Sensor térmico de alta precisión.	0-4000 ml/min	CAN, TTL pulse 0-5 V, Analog 0-10 V

3.2. ¿CÓMO SE PUEDE MEDIR LAS REVOLUCIONES DEL MOTOR DE UNA MOTOCICLETA?

Conocer las revoluciones del motor es importante cuando se requiere información sobre el estado funcional del motor. Toda caracterización de un motor implica necesariamente conocer la velocidad angular de este, normalmente medidas en revoluciones por minuto, y de esta manera determinar el régimen del motor.

Las revoluciones por minuto de un motor son la medida de cuántas rotaciones completas realiza el cigüeñal en este lapso de tiempo, y el número de veces que cada pistón sube y baja en su cilindro. La rotación del motor es generada en conjunto con el movimiento lineal que realiza el pistón en su cilindro. Ahora bien, los pistones son los encargados de mover los gases. El vacío que generan en la cámara de combustión llama a la mezcla a su interior para luego comprimirla y aprovechar la fuerza de la explosión. Este movimiento vertical es el que empuja el cigüeñal a través de las bielas y lo que posteriormente se convierte en la energía que mueve el vehículo.

Existen múltiples factores por los cuales el consumo de gasolina puede verse afectado, como la altitud, la densidad del aire, bujías ineficientes, entre otras. No

obstante, investigadores realizan estudios [14] para encontrar maneras de mejorar el rendimiento del motor. Un elemento fundamental y objeto de estudio para el buen funcionamiento del motor es el sensor de RPM. Detecta las revoluciones a las que gira el cigüeñal del motor, y envía la información a la unidad de control del sensor para su procesamiento.

El principio de funcionamiento radica en la detección de las variaciones magnéticas que se desarrollan entre el captador del sensor y el elemento de giro, que puede variar según el tipo de sensor. En motocicletas con inyección electrónica la información captada es traducida a una señal eléctrica y enviada a la ECU para que determine el caudal de combustible que debe ser inyectado en el instante correcto.

Utilizando toda la información previamente descrita se plantean las posibles alternativas de solución.

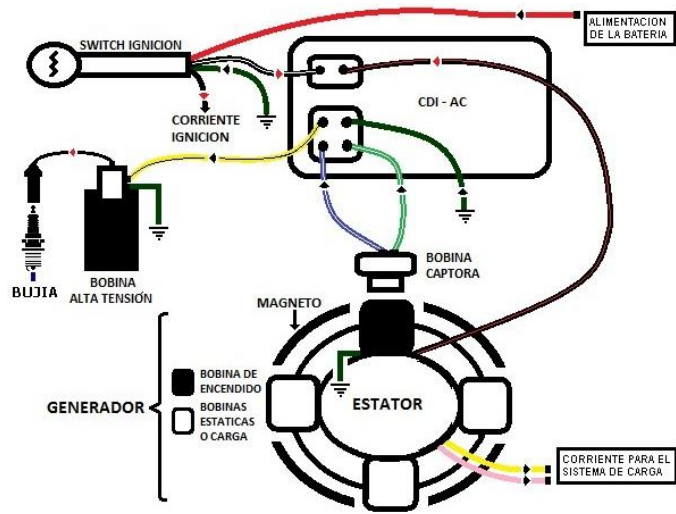
- Capturar directamente las señales emitidas por el sensor velocímetro de efecto hall. Este fenómeno físico se caracteriza por la aparición de un campo eléctrico en un material conductor o semiconductor por el cual circula una corriente, este material es atravesado perpendicularmente por un campo magnético el cual genera una diferencia de potencial. La diferencia de potencial son pulsos realizados por la rotación del eje en movimiento rotatorio los cuales representan un circuito cerrado o abierto dependiendo de la aparición del campo magnético.
- Convertir el movimiento rotatorio en una señal eléctrica utilizando un encoder para ser leída por una unidad de procesamiento y usar esta información para enviar un comando para una función particular. Los encoders utilizan diferentes tecnologías para generar la señal, algunas de ellas son: mecánica, magnética, óptica y de resistencia.
- Captar la señal de pulsos de un tacómetro digital comercial externo instalado en la motocicleta y con ello realizar un procesamiento de señal.

Tabla 7. Alternativas de solución para medición de velocidad angular.

NOMBRE	DESCRIPCIÓN	ALIMENTACIÓN
Kit medidor de tacómetro digital Trail Tech 752-118 [47]	Las características incluyen un gráfico grande de RPM y lectura de velocidad, luces LED programables que parpadean cuando es el momento de cambiar de marchas o durante un evento de sobre temperatura.	Funciona con sistemas eléctricos regulados de CA o de 12 V CC
Tacómetro 12RMP Velocímetro Digital Universal LCD [48]	Este velocímetro universal es adecuado para todas las motocicletas con carburador DC 12V, y este tacómetro para motocicleta adecuado es para ruedas de 8-22 pulgadas. Función múltiple en una: velocímetro general, odómetro, tacómetro, indicador de nivel de aceite, indicador izquierdo y derecho, marcha, batería, indicador de temperatura del agua.	12 V DC
Sensor de Efecto hall Winston CS-SM12-3002NB [49]	Sensor de alta precisión de ubicación repetida con diversificadas estructuras exteriores. Buen rendimiento de anti interferencias y cuenta con alta frecuencia de encendido-apagado. Tiene un amplio rango de voltaje, a prueba de polvo, a prueba de vibraciones, a prueba de agua y a prueba de aceite. Con protección contra cortocircuitos y protección de conexión invertida.	6-36 V DC
Encoder rotatorio Omron E6b2-cwz6c 1000 Pr [50]	Encoder Omron rotatorio con resolución 500 pulsos/rotación y fases de salida A,B y Z. Las diferencias entre las fases de salida son $90^{\circ} \pm 45^{\circ}$ entre A y B.	5-24 V DC

Por último, las motocicletas carburadas que poseen encendido eléctrico, como los casos de estudio presentados, incluyen en su sistema de arranque un motor eléctrico que genera la fuerza motriz necesaria para romper la inercia, el cual una vez encendida la motocicleta se desacopla del cigüeñal abriéndole paso al motor de combustión interna para cumplir su función motriz y empezar su ciclo de funcionamiento.

Figura 4. Sistema de encendido eléctrico en motocicleta



De esta manera el motor se convierte en un generador de corriente trifásica, del cual se desprenden dos fases para alimentar el sistema de carga y una para alimentar la bujía, pero adicionalmente existe una cuarta señal captada por una pequeña bobina denominada bobina pulsante o captora, el cual es un sensor que reacciona a la variación de campo magnético que genera la rotación de un imán permanente anclado al cigüeñal. De esta manera se puede enviar una señal pulsante que sincroniza todo el sistema y aporta la información de las revoluciones del motor para que la bujía se accione en el momento adecuado. Por lo tanto, la bobina pulsante se convierte en la mejor opción para poder tomar el dato de la velocidad angular del motor.

3.3. ¿CÓMO MEDIR LA VELOCIDAD DE LA MOTOCICLETA DURANTE UN RECORRIDO?

Se contempla la variable de interés como velocidad media (también llamada velocidad promedio) que es el cociente del espacio recorrido por un objeto entre el tiempo que este tarda en cubrir la trayectoria. Existen distintas maneras de capturar el dato de velocidad en la motocicleta:

- Acople de un odómetro de bicicleta: Todos los odómetros tienen como objetivo la medición de distancias, es decir, toma las vueltas que da una rueda de acuerdo con el diámetro conocido y por lo tanto la circunferencia. De esta forma la distancia recorrida por el objeto móvil, que rueda sobre ella, es función del número de vueltas que haya girado. Es posible evaluar la velocidad con el tiempo empleado en la trayectoria por medio de una unidad de procesamiento electrónico.

- Acople de sensor magnético o de efecto Hall: Generalmente la medición de velocidad en motocicletas es usando sensores inductivos o magnéticos acoplados en el eje de la rueda delantera del vehículo. La señal emitida por dichos sensores es procesada y convertida en unidades de velocidad en los tableros.
- GPS: Capturando el tiempo y la información de localización, una unidad GPS puede calcular rápidamente la velocidad relativa del objeto, basado en cuánta distancia ha recorrido en un determinado intervalo de tiempo. En consecuencia con ello, es posible capturar la velocidad de la motocicleta con un módulo GPS y transmitirla a una unidad electrónica de procesamiento.

Utilizando la información previamente descrita se desarrolla una tabla de posibles alternativas de solución.

Tabla 8. Alternativas de solución para medición de velocidad.

NOMBRE	DESCRIPCIÓN	ALIMENTACIÓN
Odómetro Velocímetro de Bicicleta Sunding SD-563B [51]	Monitorea la velocidad, la distancia y el tiempo de conducción de las bicicletas. El odómetro trae su propio sensor magnético de velocidad.	Batería Lipo no recargable
Sensor magnético odómetro Universal [52]	Sensor adecuado para tacómetro velocímetro de motocicleta con cable de 1.5 m de longitud de 3 hilos de alto rendimiento con dos imanes.	12 VDC
Velocímetro digital Universal Sport Moto [53]	Tablero universal para funciones originales que trae cada moto, por lo tanto, la moto debe adaptado a su sistema eléctrico: las direccionales, sensor de nivel de gasolina, de marcha, temperatura, luces altas, entre otras. El velocímetro trae su propio sensor magnético de velocidad envueltos en plásticos abs totalmente impermeables.	12 V DC
ELING Velocímetro GPS digital ELING TLG85BS [54]	Medidor de velocidad LCD Odómetro ajustable con antena GPS El medidor tiene las ventajas de alta precisión, confirmación de ubicación rápida y tiempo de arranque corto.	12 - 24 VDC

Tabla 8. Continuación.

NOMBRE	DESCRIPCIÓN	ALIMENTACIÓN
Modulo GPS Neo 6m [55]	El módulo se comunica a través de puerto serial UART con eeprom incluido, además incluye antena de 25mm x 25mm. Frecuencia de actualización de 5Hz	3-5 V DC
Modulo GPS Beitian BN-220 [36]	Se comunica por medio de puerto serial UART. Es un módulo dual del receptor del glonass de los gps, equipado del flash, no tiene ninguna brújula. Es compatible con arduino, control de vuelo y PC. Dimensiones 22x20x6 mm	3-5 V DC

3.3.1. Precisión y normativa legal

Los velocímetros de fábrica del vehículo muestran una velocidad mayor a la velocidad real, lo cual asegura que el conductor mantenga una velocidad menor a la de los límites establecidos en la carretera. Así mismo, esto sucede bajo unas normas que gobiernan este aspecto en el sector automotriz.

Según el reglamento No. 39 de la Comisión Económica para Europa de las Naciones Unidas [25], en toda la Unión Europea se establece que la velocidad indicada por el velocímetro nunca deberá ser inferior a la velocidad real con un margen de error máximo sobre la velocidad real del 10% + 4 km/h.

En otros países, como Estados Unidos y Japón, se rigen bajo el estándar J1226 de la Sociedad de Ingenieros de Automoción (SAE, en inglés). [24], cuya norma es más estricta pues se admite una desviación máxima de solo el 4%. Estos reglamentos son asumidos por la industria automotriz y son totalmente asociados directamente a la seguridad vial.

Los servicios prestados por GPS son velocímetros posicionales, es decir, se basan en la distancia que el vehículo ha recorrido desde la última medición. Los cálculos de velocidad por sistema GPS son más precisos que los velocímetros del vehículo, ya que el GPS no se ve afectado por factores como el tamaño de las ruedas o las relaciones de transmisión. Todo depende, desde luego, de la calidad de la señal que el GPS recibe del satélite, pero con los cálculos del promedio de desplazamiento, cualquier error es mínimo.

3.4. ¿CÓMO TOMAR LA POSICIÓN DE LA MOTOCICLETA DURANTE UN RECORRIDO?

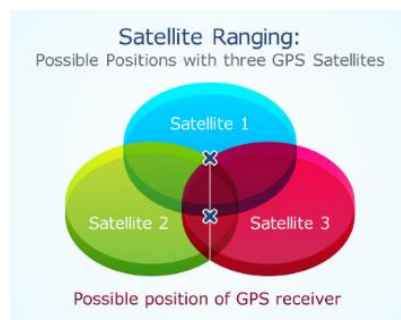
La posición de la motocicleta es una de las variables de interés para el análisis de los ciclos de conducción debido a que permite tomar registros de la ruta recorrida, además abre campo para analizar la relación entre las características del terreno, por ejemplo, si es una vía en ascenso o descenso o si es carretera o vía rural.

La tecnología GPS (Global Positioning System) consiste en una constelación de 24 satélites en órbita que transmiten información precisa de tiempo y posición a estaciones terrestres, sin embargo, estas estaciones no transfieren información de vuelta al satélite GPS, solamente utilizan la información recibida para realizar cálculos matemáticos. El receptor GPS es el encargado de calcular la posición con un método matemático conocido como trilateración, utilizando la información transmitida por los satélites.

La trilateración es una técnica geométrica para determinar la posición de un objeto conociendo su distancia a tres puntos de referencia. A diferencia de la más conocida técnica de triangulación, en la que se miden ángulos y distancias, en la trilateración sólo se utilizan distancias. Cada uno de los satélites de la red completa dos órbitas a la tierra en 24 horas y envía una señal, parámetros orbitales e información horaria únicos.

Un solo satélite no puede proporcionar mucha información sobre la ubicación. Cuando un satélite envía una señal, crea un círculo con un radio medido desde el módulo GPS al satélite. Un segundo satélite, crea un segundo círculo, y la ubicación se reduce a uno de los dos puntos donde los círculos convergen. Con un tercer satélite, la ubicación del receptor se puede determinar definitivamente. Esto quiere decir que entre más satélites se encuentre sincronizado el GPS, mayor es la exactitud de las coordenadas de posición, solo depende de la tecnología y capacidades técnicas del módulo receptor [20][21].

Figura 5. Sistema GPS



Tomado de GeoTab [29].

Dicha información obtenida y procesada por el dispositivo GPS se traduce en latitud, longitud, altitud, velocidad, distancia, tiempo, entre otros. De esta manera, la posición de la motocicleta se toma por medio de un módulo GPS recolectando datos de ruta mediante la unidad electrónica de procesamiento.

3.5. ¿CÓMO ALMACENAR Y PROCESAR LA INFORMACIÓN RECOLECTADA DE LA MOTOCICLETA DURANTE UN RECORRIDO?

Actualmente dentro de las tecnologías de vanguardia se encuentra la Industria 4.0, la cual se caracteriza por la introducción de múltiples tecnologías como *Big Data*, *Cybersecurity*, *Aumentaty Reality*, entre otras. Una de ellas es el Internet de las Cosas (*IoT – Internet of Things*). Se estima que existen alrededor de 27.000 millones dispositivos conectados a Internet. Según el portal web Organizational Projects Management [30] para el año 2030 la proyección es que 125 mil millones de objetos estarán conectados a este servicio. Esto indica que se llevarán a cabo miles de proyectos relacionados al IoT.

El Internet de las cosas es una red que interconecta objetos físicos/dispositivos mediante el Internet, a través de la utilización de sistemas o aplicaciones embebidas en los dispositivos que permiten la conectividad. Asimismo, estos sistemas facilitan la programación de eventos de forma remota en los dispositivos.

Existen en el mercado distintas placas de desarrollo electrónico que permiten aplicar las diferentes tecnologías de la industria 4.0 contribuyendo a la evolución y aprendizaje de las mismas. Dentro de las más conocidas se encuentran las desarrolladas por Arduino, Raspberry, Microchip, entre otras.

Utilizando toda la información previamente descrita se desarrolla una tabla de posibles alternativas de solución.

Tabla 9. Alternativas de solución para la selección de la unidad de procesamiento.

NOMBRE	DESCRIPCIÓN	TECNOLOGÍA UTILIZADA	ALIMENTACIÓN
Arduino MKR1000	Tarjeta de desarrollo basada en el microcontrolador SAMD21 Cortex-M0+, combina la funcionalidad del Zero y el shield WiFi. Cuenta con 8 pines digitales, 7 pines analógicos de entrada y 1 de salida.	Microcontrolador or SAMD21 Cortex®-M0+ 32bit	5 v - 3.3 v
Arduino MKR WiFi 1010	Placa pequeña, robusta y potente con conectividad WiFi y Bluetooth con facilidad de acceso a Arduino Cloud. Cuenta con 8 pines digitales, 7 pines analógicos de entrada y 1 de salida.	Microcontrolador or SAMD21 Cortex®-M0+ 32bit	3.3 v
Arduino UNO Rev3	Placa basada en el microcontrolador ATmega328P. Tiene 14 pines de entrada/salida digital, 6 entradas analógicas, un cristal de 16Mhz, conexión USB, conector jack de alimentación, terminales para conexión ICSP y un botón de reseteo	Microcontrolador or ATmega328P.	5 v - 3.3 v
Arduino Mega 2560 Rev3	Tarjeta de desarrollo open-source construida con un microcontrolador Atmega2560. tiene 54 pines de entradas/salidas digitales, 16 entradas análogas, 4 UARTs (puertos serial por hardware),	Microcontrolador or Atmega2560	5-20 V
Arduino Nano	Arduino Nano es una placa de desarrollo de tamaño compacto, completa y compatible con protoboards, basada en el microcontrolador ATmega328P. Tiene 14 pines de entrada/salida digital, 6 entradas analógicas, un cristal de 16Mhz, conexión Mini-USB, terminales para conexión ICSP y un botón de reseteo.	Microcontrolador or ATmega328P	6-20 VDC
Arduino Leonardo	El Arduino Leonardo es una placa de desarrollo basada en el microcontrolador ATmega32u4. Tiene 20 pines de entrada/salida digital, un cristal de 16Mhz, conexión micro USB, conector DC de alimentación, conector ICSP y botón de reinicio.	Microcontrolador or ATmega32u4	7-20 VDC

Tabla 9. Continuación.

NOMBRE	DESCRIPCIÓN	TECNOLOGÍA UTILIZADA	ALIMENTACIÓN
Raspberry Pi 4	Raspberry Pi 4 Modelo B es el último producto en la popular gama de ordenadores Raspberry Pi. Las características clave de este producto incluyen un procesador de cuatro núcleos de 64 bits de alto rendimiento, soporte de pantalla dual en resoluciones de hasta 4K a través de un par de puertos micro-HDMI, decodificación de vídeo de hardware hasta 4Kp60, 4GB de RAM.	Broadcom BCM2711 quad-core Cortex-A72 (ARM v8) 64-bit SoC	5 V DC
NodeMCU V3	El NodeMcu es un kit de desarrollo de código abierto basado en el popular chip (ESP8266), que utiliza el lenguaje de programación Lua para crear un ambiente de desarrollo propicio para aplicaciones que requiera conectividad Wifi de manera rápida.	ESP8266	3.3 – 5 VDC

3.5.1. Tarjeta de desarrollo NodeMCU ESP8266

NodeMCU es una placa open hardware por lo que su diseño es de libre distribución y utilización. El cerebro de la placa es el chip WiFi ESP8266 que trabaja bajo el protocolo TCP/IP, modelo altamente usado para acceder a internet. Las características del ESP8266 se extraen en la tarjeta de desarrollo NodeMCU y juntas forman el firmware autónomo apto en aplicaciones IoT. Una característica importante de la placa es la amplia disponibilidad de pines GPIO y de puertos habilitados para diferentes protocolos de comunicación como UART, I2C, I2S y SPI.

Con la popularización y evolución de la NodeMCU ESP8266, los creadores adicionaron más bibliotecas para disponibilidad de todos los desarrolladores de la placa, como la posibilidad de utilizar el protocolo MQTT de Contiki, y la posibilidad de utilizar el muy conocido IDE Arduino como editor de código fuente.

La evolución de la NodeMCU es muy amplia en proyectos IoT, tanto que se tienen varias maneras de monitorear los datos que se captan a través de los sensores. La manera más básica requiere tener el dispositivo conectado a un ordenador para la visualización de la información utilizando la comunicación serial. Sin embargo, se puede mostrar en una pantalla LCD, lo que permite desconectar la placa del

computador y desplazarlo a cualquier lugar. Pero existe una tercera opción, quizás la más idónea, para poder supervisar los datos y la información en tiempo real; utilizar una plataforma para proyectos IoT.

Todos los proyectos en IoT tienen algo en común, se dedican a compartir la información a través de plataformas enfocadas en recibir y transmitir datos. Según Mendoza y Marín [31], en un sistema basado en plataformas en la nube intervienen tres elementos principales:

- El dispositivo de monitoreo conectado.
- La plataforma IoT.
- Los dispositivos que permiten la visualización y el tratamiento de la información en la plataforma IoT.

Cada uno de estos sistemas se trata por separado y se comunican entre ellos a través de protocolos de comunicación. Estos protocolos son estándar para que, independientemente de la plataforma, se puedan comunicar. Algunos de los protocolos más utilizados y conocidos son HTTP, MQTT y CoAP.

3.5.2. Plataformas IoT

Las plataformas IoT son la base sobre la cual los dispositivos se interconectan y generan sus propios entornos. Según la empresa desarrolladora de proyectos IoT *link-labs* [32], una plataforma IoT es el software que conecta hardware, puntos de acceso y redes de datos a lo que generalmente suele ser la aplicación con la que interactúa el usuario. Las diferentes plataformas IoT se clasifican dependiendo del costo y el sector al que va orientado:

- El primer grupo, son las plataformas que están orientadas a startups y empresas pequeñas. También están incluidas todas aquellas que permiten el uso de manera gratuita, pero con limitaciones en cuanto al número de datos enviados y de dispositivos conectados.
- El segundo grupo engloba a plataformas que también ofrecen servicios gratuitos o versiones de prueba, pero están más centradas en ofrecer servicios globales a sistemas basados en IoT.
- En el tercer grupo están las plataformas que ofrecen las grandes empresas y corporaciones como Google, Amazon, Microsoft, IBM, entre otros. Están orientadas sobre todo al sector industrial y a grandes proyectos IoT, donde se ven involucrados cientos o miles de dispositivos.

Utilizando toda la información previamente descrita se desarrolla una tabla de posibles plataformas IoT por utilizar.

Tabla 10. Alternativas de solución de Plataformas IoT.

NOMBRE	DESCRIPCIÓN
Samsung Artik Cloud	Artik Cloud es la apuesta de Samsung por el sector del IoT. No solo se trata de una plataforma en la nube, también se puede adquirir hardware, el dispositivo Artik 1020. Esto no quiere decir que se debe tener un módulo Artik para comunicar con la nube, se puede hacer con cualquier dispositivo compatible.
aRest Framework	aRest Framework está centrado en dar soporte a placas como Arduino, ESP8266 y Raspberry Pi. Es una solución completa para crear aplicaciones con servicios RESTful. Soporta comunicaciones WiFi, Ethernet, Bluetooth y Serial. Incluye librerías para Arduino y también del lado del servidor, para controlar las comunicaciones.
Blynk	: Es una plataforma IoT para Android y iOS de control de sistemas desarrollados con Arduino y Raspberry Pi. Este sistema no está limitado a ningún hardware específico. Solo se requiere que el proyecto con Arduino o Raspberry Pi esté conectado a internet por medio de Wifi, Ethernet o un ESP8266. De esta manera, podrá conectarse con el servidor online o local de Blynk y controlar todo el sistema.
Arduino Cloud:	Quizás sea la más sencilla para utilizar en proyectos IoT con Arduino puesto que es totalmente compatible con cualquier placa de Arduino que tenga conectividad. Las limitaciones son evidentes, no tiene una API para consultar los datos desde otras aplicaciones, no tiene histórico y no representa la información en tiempo real. Según Arduino, muy pronto irán incorporando estas funcionalidades.
ThingSpeak:	Es la apuesta de de MathWorks, los creadores de MathLabs y está enfocado exclusivamente a la construcción de aplicaciones del IoT. Permite almacenar datos, visualizarlos y exponerlos a otras APIs.

La elección de la plataforma IoT a utilizar se realiza con base en su rendimiento, eficacia y facilidad de implementación en el proyecto bajo un punto de vista objetivo y rentable. Por tal motivo, se llega a la conclusión de que Blynk es la mejor alternativa para el desarrollo del proyecto.

3.5.3. ¿Cómo funciona Blynk?

Blynk es una plataforma gratuita IoT independiente del hardware y basada en la nube que permite a los usuarios conectar más de 400 modelos de hardware con la nube Blynk segura y de código abierto a través de WiFi, redes móviles 2G-5G o Ethernet. Esta plataforma de código abierto se puede implementar en servidores locales, en entornos de nube como AWS o alojada por el propio Blynk.

Una de las facilidades de trabajar con Blynk es que sus librerías están disponibles en varias plataformas de desarrollo como Arduino. De tal modo, se asegura la comunicación entre los recursos del hardware y el servidor. Las funciones de la aplicación incluyen visualización de datos del sensor por medio de gráficos, control remoto del equipo, administración de dispositivos, controles de reglas, notificaciones y muchos más widgets funcionales tales como deslizadores, botones, entre otros.

Figura 6. Interfaz Blynk App [Obtenida de <https://blynk.io/>]



Tomado de Blynk [33]

3.5.4. Arquitectura de Blynk

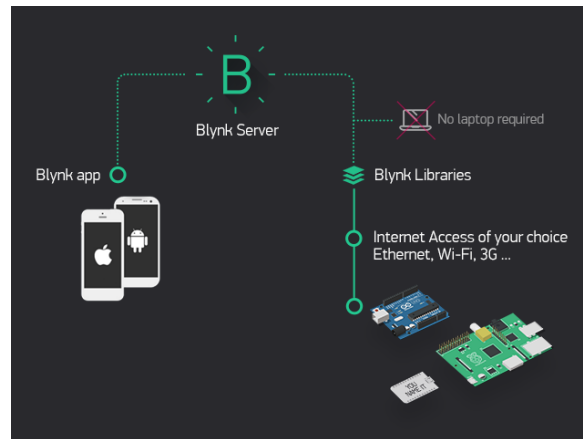
Según el portal web oficial de Blynk, el funcionamiento de la plataforma incluye los siguientes componentes:

- *Blynk app*: Permite crear interfaces para los proyectos utilizando varios widgets previamente mencionados y está disponible para las plataformas Android e iOS.
- *Blynk Server*: Responsable de todas las comunicaciones entre el dispositivo móvil que está ejecutando la aplicación Blynk y el hardware. La posibilidad de ser código abierto proporciona facilidad de manejar muchos dispositivos. Se

puede utilizar la nube de Blynk o ejecutar un servidor Blynk privado de forma local.

- *Blynk Libraries*: Permiten la comunicación con el servidor y procesan todos los comandos entrantes y salientes de la app Blynk y el hardware.

Figura 7. Arquitectura de Blynk



Tomado de Blynk [33]

3.5.5. Protocolo de comunicación – Blynk

Según el portal web oficial de Blynk tiene su propio protocolo para transmisión de datos (Hardware – Servidor Blynk), el cual consta de dos partes:

- Encabezado:
 - Comando del protocolo (1 byte)
 - ID del mensaje (2 bytes)
 - Longitud del cuerpo del mensaje (2 bytes)
- Cuerpo: *String* que puede ser hasta 2^{15} bytes

En conclusión, Blynk transfiere mensajes binarios con la siguiente estructura:

Tabla 11. Estructura de protocolo de comunicación Blynk

Comando	ID del mensaje	Longitud/Estado	Cuerpo
1 byte	2 bytes	2 bytes	Variable (0- 2^{15} bytes)

Tomado de Blynk.io

De esta manera el mensaje es siempre "1 byte + 2 bytes + 2 bytes + Longitud del cuerpo del mensaje".

3.5.6. Almacenamiento de datos – Protocolo HTTPS

Almacenar datos en la nube es guardar información de manera segura en una base de datos alojada en un servidor remoto interconectado a la red de internet evitando guardar los datos y archivos en el disco duro y memorias de un computador u otro dispositivo de procesamiento como la NodeMCU. Algunos ejemplos de las herramientas que prestan el servicio de alojamiento de información son Dropbox, Google Drive y One Drive, entre otros.

El envío de los datos obtenidos por Hardware a la nube es posible con la NodeMCU ESP8266 gracias al protocolo HTTPS. Este método realiza una petición HTTP a través de una conexión TLS. Al hacerlo, los datos que se envían de ida y vuelta entre el ordenador o dispositivo y el servidor están encriptados y protegidos. El chip ESP8266 soporta el protocolo con la clase *HTTPSRedirect* proveniente de la librería *WiFiClientSecure*.

HTTPSRedirect utiliza la información de la cabecera en la respuesta del servidor a una solicitud HTTP tipo GET o POST, y sigue la URL redirigida haciendo otra solicitud GET adecuada. La oportunidad que brinda la NodeMCU ESP8266 al utilizar la clase *HTTPSRedirect* es trabajar directamente con muchos servicios web de Google de manera gratuita como lo es con Calendar, Gmail y GoogleSheets sin necesidad de servicios secundarios como IFTTT, ThingSpeak, entre otros.

3.5.7. Google Apps Script

Para comunicar directamente desde la NodeMCU a los servicios de Google es necesario crear un script. El script se desarrolla en Google Apps Script, cuya plataforma permite realizar programas con múltiples funcionalidades que se ejecutan en un entorno definido por Google. Estos programas permiten comunicarse con fuentes de datos, tanto del propio entorno de Google como externas.

Según el portal web de consultoría TIC beServices [33], dentro de los beneficios que trae trabajar con Google Apps Script, se encuentra:

- Aprendizaje rápido: Se basa en el lenguaje popular JavaScript, presente en la mayoría de los sitios web actuales.
- Automatizar procesos internos: Apps Script de Google permite dejar de realizar manualmente muchas tareas mecánicas, ahorrando tiempo y recursos.
- No requiere contratación de un servidor: La plataforma de Google se encarga del alojamiento y la ejecución de los scripts.

4. OBJETIVOS

4.1. OBJETIVO GENERAL

Desarrollar un sistema instrumental para el monitoreo segundo a segundo de las variables de operación de motocicletas (posición, velocidad, RPM, consumo de combustible) durante tiempos prolongados en motocicletas carburadas de cilindradas menores a 250 cc.

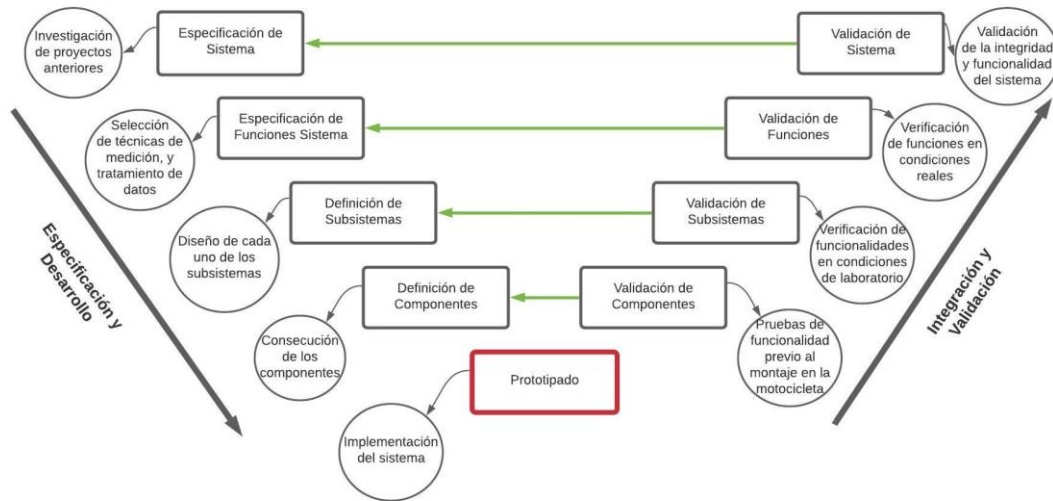
4.2. OBJETIVOS ESPECÍFICOS

- Identificar la instrumentación para monitorear en tiempo real las variables de operación de motocicletas carburadas de cilindradas menores a 250 cc.
- Seleccionar la instrumentación para monitorear en tiempo real las variables de operación de motocicletas carburadas de cilindradas menores a 250 cc.
- Diseñar a escala semi-industrial un sistema instrumental que monitoree y registre la posición, velocidad, RPM y consumo de combustible de una motocicleta carburada de cilindrada menor a 250 cc.
- Implementar a escala semi-industrial un sistema instrumental que monitoree y registre la posición, velocidad, RPM y consumo de combustible de una motocicleta carburada de cilindradas menores a 250 cc.
- Desarrollar una aplicación móvil que permita la visualización y almacenamiento en la nube de los datos obtenidos a partir de una ruta monitoreada.
- Validar el funcionamiento del dispositivo a partir de las mediciones obtenidas mediante pruebas de laboratorio y pruebas de ruta.

5. METODOLOGÍA Y DESARROLLOS

La metodología elegida para el desarrollo del presente proyecto es la metodología en V, ya que esta permite llevar una secuencia ordenada de desarrollo en cada una de las partes que forman el proyecto de manera individual y específica y validar el diseño para verificar que se cumplan los objetivos propuestos.

Figura 8. Metodología en V.



5.1. SELECCIÓN DE ALTERNATIVAS:

La matriz de QFD (Quality Function Deployment) es una herramienta de análisis sistemático que permite discriminar las propiedades de un producto a partir de las necesidades del cliente para mejorar la calidad del producto.

Se desarrolla una Matriz de QFD con el objetivo de seleccionar las alternativas de solución previamente propuestas basados en los siguientes requisitos funcionales y requisitos del cliente:

- Requisitos funcionales:
 - ¿Cómo medir la velocidad angular del motor en rpm?
 - ¿Cómo medir la velocidad de la motocicleta?
 - ¿Cómo medir la posición de la motocicleta?
 - ¿Cómo medir el consumo de combustible de la motocicleta?
 - ¿Cómo procesar, transmitir y visualizar la información capturada en ruta?

- Requisitos del cliente:

- Permitir la recolección, procesamiento y visualización de datos de ruta.
- Mostrar un resumen de datos recolectados durante una ruta.
- Almacenar los datos a la nube automáticamente.
- Conservar integridad del dispositivo y fácil instalación en la motocicleta.

Tabla 12. Matriz QFD.

1: low, 5: high Customer importance rating	Desired direction of improvement (↑,0,↓)	RPM				Velocidad			
	Functional Requirements (How) →	Sensor inductivo	Sensor de efecto Hall	Tacómetro	Encoder	Velocímetro	Módulos GPS	Odómetro	Sensor de efecto Hall
	Customer Requirements - (What) ↓								
5	Recolectar los datos de ruta	3	3	9	1	3	9	3	3
5	Procesar los datos capturados.								
5	Transmitir los datos monitoreados para su almacenamiento.								
4	Mostrar resumen de los datos de ruta.								
5	Subir los datos a la base de datos en la nube.								
3	Preservar la integridad ante condiciones climáticas y perturbaciones externas	3	3	9	1	9	9	9	3
5	Almacenar los datos en la nube.								
	Technical importance score	24	24	72	8	42	72	42	24
	Importance %	19%	19%	56%	6%	23%	40%	23%	13%
	Priorities rank	2	2	1	4	2	1	2	4

Tabla 12. Continuación.

1: low, 5: high Customer importance rating	Desired direction of improvement (↑,0,↓)	Posición	Consumo		Distancia	
	Functional Requirements (How) →	Módulos GPS	Flujómetro	Diferencial de volumen	Odómetro	Módulos GPS
	Customer Requirements - (What) ↓					
5	Recolectar los datos de ruta	9	9	3	3	9
5	Procesar los datos capturados.					
5	Transmitir los datos monitoreados para su almacenamiento.	9				9
4	Mostrar resumen de los datos de ruta.					
5	Subir los datos a la base de datos en la nube.					
3	Preservar la integridad ante condiciones climáticas y perturbaciones externas	9	3	3	9	9
5	Almacenar los datos en la nube.					
	Technical importance score	117	54	24	42	117
	Importance %	100%	69%	31%	26%	74%
	Priorities rank	1	1	2	2	1

Tabla 12. Continuación.

1: low, 5: high Customer importance rating	Desired direction of improvement (↑,0,↓)	Procesamiento							Transmisión		
	Functional Requirements (How) →	Arduino	Raspberry Pi	NodeMCU	FPGA	PIC	SOC	PC	Módulo Bluetooth	Módulo WiFi	Módulo GSM
	Customer Requirements - (What) ↓										
5	Recolectar los datos de ruta	9	9	9	3	3	3	1			
5	Procesar los datos capturados.	9	9	9	9	9	9	9			
5	Transmitir los datos monitoreados para su almacenamiento.	9	9	9	3	3	3	9	3	3	9
4	Mostrar resumen de los datos de ruta.	1	1	9	1	1	1	1			
5	Subir los datos a la base de datos en la nube.	9	9	9	3	3	3	9	1	3	9
3	Preservar la integridad ante condiciones climáticas y perturbaciones externas	3	3	9	3	9	9	3	9	9	9
5	Almacenar los datos en la nube.										
	Technical importance score	193	193	243	103	121	121	153	47	57	117
	Importance %	20%	20%	25%	11%	12%	12%	14%	21%	26%	53%
	Priorities rank	2	2	1	6	4	4	4	3	2	1

Tabla 12. Continuación.

1: low, 5: high Customer importance rating	Desired direction of improvement (↑,0,↓)	Visualización			Plataforma IoT				
	Functional Requirements (How) →	Smartphone	Tablet	PC	Samsung Artik Cloud	aRest Framework	Arduino Cloud	ThingSpeak	Blynk
	Customer Requirements - (What) ↓								
5	Recolectar los datos de ruta								
5	Procesar los datos capturados.				3	3	3	3	9
5	Transmitir los datos monitoreados para su almacenamiento.				9	9	9	9	9
4	Mostrar resumen de los datos de ruta.	9	9	3	3	3	3	3	3
5	Subir los datos a la base de datos en la nube.				3	3	9	3	9
3	Preservar la integridad ante condiciones climáticas y perturbaciones externas	9	9	1					
5	Almacenar los datos en la nube.				9	9	9	9	9
	Technical importance score	63	63	15	132	132	162	132	192
	Importance %	45%	45%	11%	18%	18%	22%	18%	26%
	Priorities rank	1	1	3	3	3	2	3	1

5.2. MEDICIÓN DE CONSUMO DE COMBUSTIBLE

La matriz QFD muestra que la alternativa que soluciona los requerimientos del cliente y, por lo tanto, que cumple apropiadamente con el objetivo de monitoreo en tiempo real es la instalación de un flujómetro montado en línea.

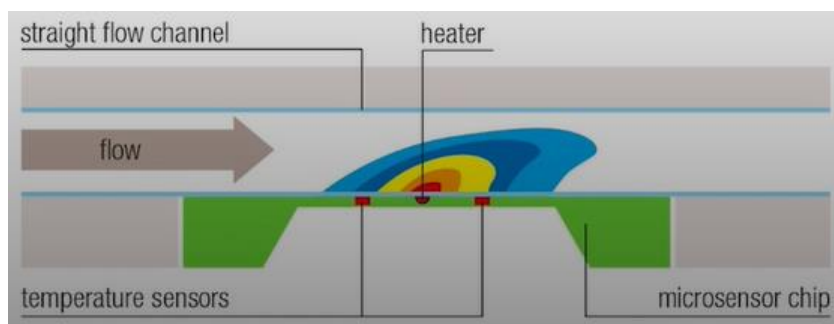
A partir de las alternativas previamente presentadas se desarrolla una tabla ponderada para evaluar las propuestas de flujómetros encontrados en el mercado.

Tabla 13. Puntajes ponderados para selección de alternativa de solución para medición de consumo de combustible.

Nombre	Rango de operación	Comunicación	Tolerancia a ruido	Facilidad de instalación	Precio	Puntaje	Ranking
SLQ-HC60	1	0.9	0.2	0.6	0	53	8
SLI-2000 FMK	1	1	0.2	0.6	0.3	60	5
LS32-1500	0.8	1	0.3	0.5	0.6	61	4
SLS-1500	0	1	0.5	0.5	0.6	42	9
SLF3S-1300F	0.8	1	0.9	0.7	0.9	85	1
LG16-100D	1	1	0.4	0.6	0.8	75	2
VBESTLIFE Turbine Flowmeter	0	0.9	1	0.2	1	57	6
US211M	0	0.9	0.9	0.2	1	54.5	7
FlowSonic® LF	0.6	0.9	1	0.3	0.5	66.5	3
Peso (0-100)	30	10	25	15	20		

El flujómetro con una mayor calificación es el flujómetro Sensirion SLF3S-1300S, el cual es un flujómetro de bajo flujo que debido a su tamaño y facilidad de instalación se puede instalar en la motocicleta, sin embargo, el fabricante vende sus productos calibrados únicamente para agua y alcohol isopropílico, pero el flujómetro tiene alta compatibilidad con gasolina convencional [34], por lo tanto, se puede recalibrar ya que este es un sensor que mide el flujo en función de sus propiedades termodinámicas utilizando una pareja de sensores de temperatura uno previo y otro posterior a un calentador y obtener una lectura a partir del gradiente de temperatura y otras propiedades implícitas del fluido.

Figura 9. Principio de funcionamiento del SLF3S-1300S.



Tomada de Liquid flow concentration sensor SLF3C-1300F - more than flow: the thermal conductivity feature [35]

Es por ello que en primer lugar se debe realizar una recalibración, para ello el fabricante entrega una guía audiovisual [34] para aplicar una curva de corrección a la medida del flujómetro para obtener un valor real.

5.2.1. Calibración del Fluviómetro para Gasolina Convencional

El procedimiento de calibración para un fluido diferente consiste en seleccionar el rango de medida deseado y bombear el fluido a diferentes flujos constantes dentro del rango por un tiempo determinado y capturar el dato leído por el sensor y el valor bombeado. Una vez completado se tabulan los valores reales y medidos para obtener una curva de ajuste que al aplicar como factores de corrección permita al flujómetro entregar una medida correcta.

La calibración requiere:

- Bomba de micro fluido
- Conexiones hidráulicas
- Dos placas Arduino
- Fluviómetro SLF3S-1300F
- Cableado electrónico

Se realiza la búsqueda de alternativas para bombear el fluido a una tasa constante por un tiempo determinado, siendo las dos más destacadas las siguientes:

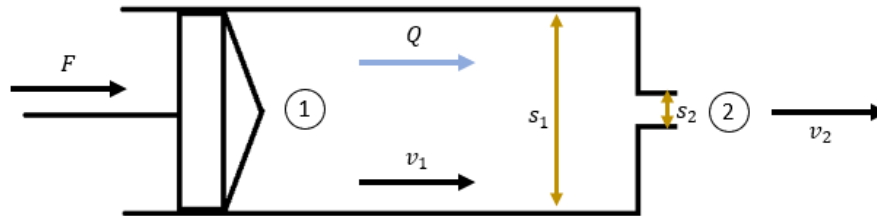
- Bomba peristáltica: Es una bomba hidráulica de desplazamiento positivo que utiliza un mecanismo de rodillos que comprimen la manguera al girar y crean un vacío que desplaza al fluido. La principal desventaja de este sistema es la complejidad del control de un flujo de salida y la condición de flujo pulsante generada por el mismo mecanismo de la bomba.
- Bomba de Infusión: Es una bomba hidráulica comúnmente utilizada para suministrar medicamentos hacia el sistema vascular del paciente de manera progresiva y controlada. Utiliza un mecanismo que genera una presión mecánica en una jeringa que desplaza el fluido a través de un tubo, la principal desventaja de este sistema es su elevado precio en el mercado.

Sin embargo, utilizando el principio de funcionamiento de la bomba de infusión, se realiza el diseño de un sistema de bombeo de tipo bomba de infusión capaz de controlar el flujo de gasolina hacia el flujómetro por un determinado tiempo.

5.2.1.1. Principio de funcionamiento:

El principio de funcionamiento de una bomba de infusión se encuentra en desplazar un fluido que fluye a través de una tubería de sección transversal constante, este es el caso de una jeringa inyectora, cuyo cilindro posee una sección transversal constante y al desplazar el émbolo a una velocidad constante se consigue un flujo constante directamente proporcional a la velocidad del émbolo.

Figura 10. Principio de Funcionamiento del Sistema de Calibración



Por principio de continuidad el flujo en 1 debe ser igual el flujo en 2:

$$Q_1 = Q_2 \quad (1)$$

$$v_1 S_1 = v_2 S_2 \quad (2)$$

Donde,

- Q , es el flujo.
- v , es la velocidad del fluido.
- S , es la sección transversal de la tubería.

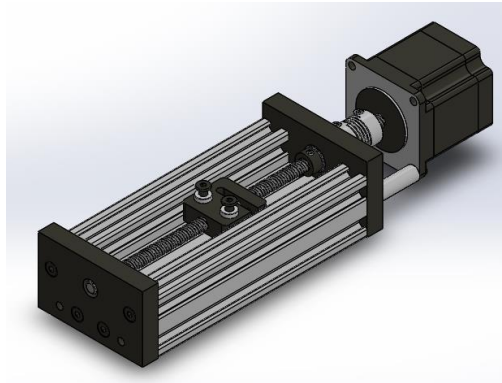
Si se conoce la sección transversal del cilindro de la jeringa es posible calcular el flujo conociendo la velocidad del émbolo.

5.2.1.2. Sistema motriz:

Para generar el movimiento rectilíneo constante necesario para el sistema se debe utilizar un dispositivo capaz de entregar un movimiento con la menor cantidad de variaciones y evitando condiciones de empuje por pulsos, con esto una solución adecuada es el mecanismo de movimiento de las máquinas CNC, las cuales utilizan un motor paso a paso, el cual genera movimientos graduales del eje con alta precisión y velocidad controlada, adicionalmente, estos incorporan una la función de micro paso programada en el controlador lo cual permite un control mucho más refinado.

El movimiento circular se debe convertir en movimiento rectilíneo, este mecanismo es llamado mecanismo husillo-tuerca, el cual es un tornillo sin fin que, al girar, desplaza una tuerca que está ensamblada a una base móvil que porta la carga.

Figura 11. Sistema Husillo Tuerca



La velocidad del motor de pasos se controla utilizando la variación del parámetro *delay* en el programa del microcontrolador encargado de este sistema. La medición de velocidad se realiza utilizando un encoder óptico rotatorio fijado al tornillo sin fin, el encoder se encarga de enviar una señal cada vez que un haz de luz atraviesa una de sus ranuras y se puede calcular el tiempo entre cada interrupción, de esta manera se puede conocer la frecuencia de giro del motor y, por lo tanto, la velocidad angular.

Los resultados se evidencian en las siguientes graficas:

Figura 12. Delay vs Velocidad Angular.

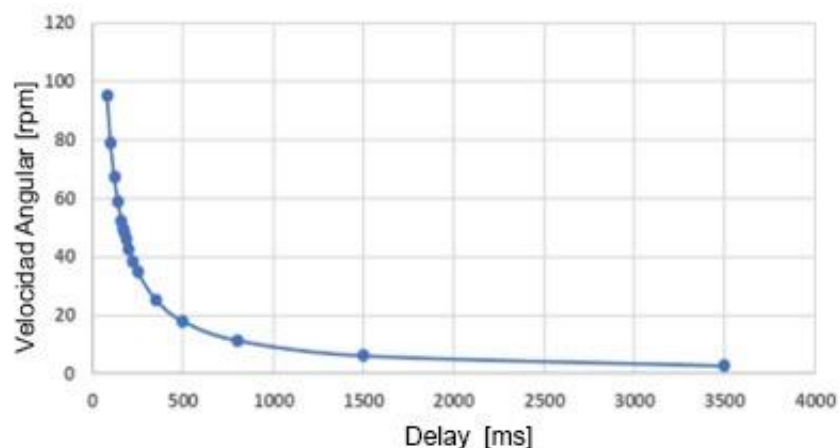


Figura 13. Delay vs flujo de combustible

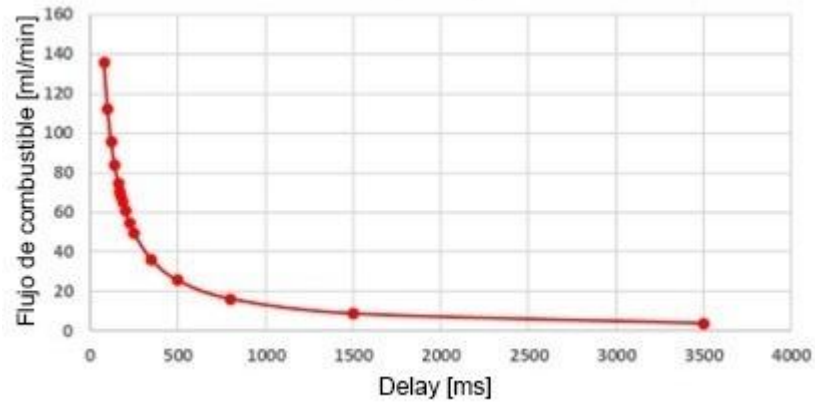
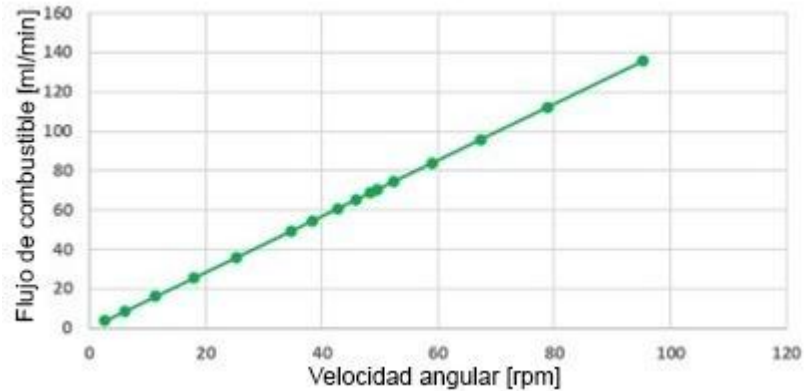


Figura 14. Velocidad angular vs flujo de combustible



La relación de velocidad lineal del carro de empuje del sistema viene dada por la expresión:

$$V_l = \omega * \varepsilon * \#_{ent} \quad (3)$$

Donde,

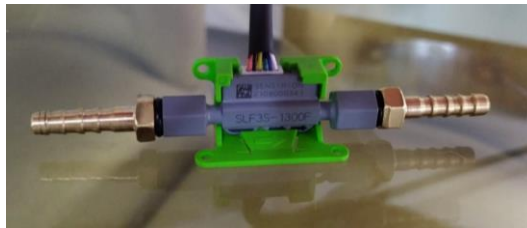
- V_l , es la velocidad lineal del carro en cm/min.
- ω , es la velocidad angular del motor de pasos en rev/min.
- ε , es el paso del tornillo sin fin.
- $\#_{ent}$, es el número de entradas del tornillo sin fin.

5.2.1.3. Conexiones Hidráulicas:

La jeringa utilizada es una jeringa de vidrio de 100 ml de capacidad, esta jeringa no tiene sistema de sellado en vacío con un polímero de alta fricción, en su lugar utiliza un sellado micrométrico que permite su hermeticidad sin aumentar su fricción.

El flujómetro utiliza rosca interna fina de ¼” para sus conexiones hidráulicas, sin embargo, la tubería de salida del sistema de bombeo posee un diámetro diferente, en este caso se realiza una adecuación utilizando mangueras y racores neumáticos para permitir el sello entre las diferentes secciones de tubería.

Figura 15. Racores acoplados al flujómetro.

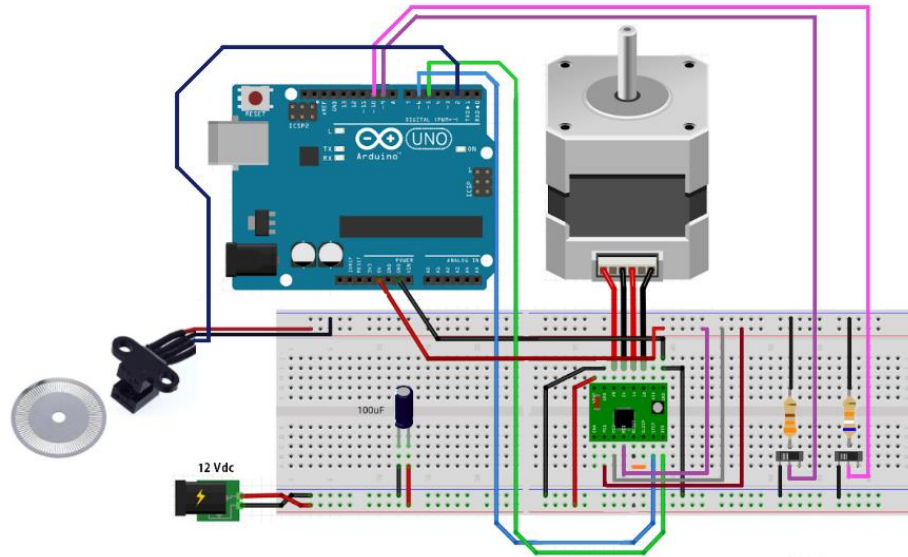


5.2.1.4. Electrónica:

El sistema se divide en dos partes principales, un circuito para controlar el sistema de bombeo y un circuito exclusivo para la comunicación con el flujómetro y la recopilación de datos.

- Sistema de Bombeo: El circuito del sistema de bombeo consiste en un Arduino UNO que controla el motor paso a paso NEMA17 a través de un driver. El control de la dirección y velocidad de giro se realiza con el driver A4988, el cual soporta el microstepping para motores bipolares como el que está en uso. La utilidad de dicha característica ayuda a la suavidad de la revolución por paso, permitiendo velocidades angulares no pulsantes. Para medir la velocidad angular del motor paso a paso se utiliza un encoder óptico rotatorio con una rueda perforada de 20 ranuras fijada al tornillo sin fin. Adicionalmente, el circuito tiene control manual que consta de dos switches que permiten el encendido y giro del motor respectivamente. El circuito queda, por lo tanto:

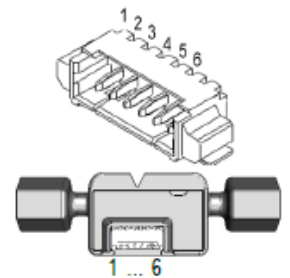
Figura 16. Conexiones del sistema de bombeo.



- Comunicación con Flujómetro: El flujómetro se alimenta de entre 3.3 y 3.6 volts por sus especificaciones técnicas, sin embargo, aunque el Arduino permite salida de tensión de 3.5 volts, las instrucciones enviadas utilizando el protocolo I2C son de 5 volts, para lo cual se utiliza un convertidor de nivel lógico bidireccional BSS138, las conexiones eléctricas de los pines del flujómetro vienen dadas por el fabricante.

Figura 17. Asignamiento de pines

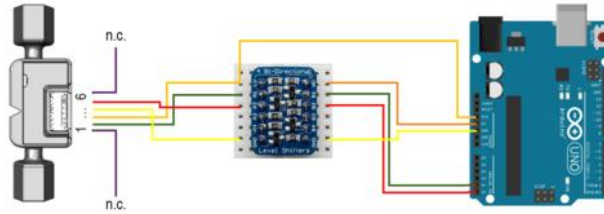
Pad	Description	Comments
1	n.c.	Nonfunctional, connect to GND or leave floating
2	SDA (data)	Serial data, bidirectional
3	VDD	Supply voltage
4	GND	Ground
5	SCL (clock)	Serial clock, bidirectional
6	n.c.	Nonfunctional, connect to GND or leave floating



Tomado de Sensirion [34].

El circuito, por tanto, es.

Figura 18. Conexiones del flujómetro con Arduino



5.2.1.5. Código fuente Sistema de bombeo

- Sistema de bombeo: Se programan los accionamientos de las bobinas ejecutando los movimientos de cada micropaso, de acuerdo con el estado de los switches de control [Ver código completo en anexo 1].

```
void loop() {
  estado = digitalRead(dirButn);
  if (estado == HIGH) {
    digitalWrite(dirPin, LOW);
  }
  else {
    digitalWrite(dirPin, HIGH);
  }

  while (digitalRead(pare) == HIGH) {
    for (int x = 0; x < stepsPerRevolution; x++) {
      if (digitalRead(pare) == LOW) {
        break;
      }
      digitalWrite(stepPin, HIGH);
      delayMicroseconds(delayTime);
      digitalWrite(stepPin, LOW);
      delayMicroseconds(delayTime);
      checkDir();
    }
  }
}
```

Se realiza un método que cuenta las señales de pulsos del sensor infrarrojo del encoder rotatorio y mediante el cálculo del tiempo entre cada pulso se obtiene la velocidad angular del motor stepper.

```

void counter() {
  if ( digitalRead (encoder_pin) && (micros() - debounce > 100) ) {
    debounce = micros(); // Almacena el tiempo para comprobar que no
    contamos el rebote que hay en la señal.
    timeold = millis(); // Almacenamos el tiempo actual.
    float t = (timeold - time18D) / 1000;
    rpm = 18 / (t * 6);
    time18D = timeold;
    Serial.print("Revoluciones: ");
    Serial.print(rpm, DEC);
    Serial.print("rpm ");
    flujo = rpm * 0.8 * PI * 1.488 * 1.488 / 4;
    Serial.print(", Flujo: ");
    Serial.print(flujo, 2);
    Serial.print("ml/min");
    Serial.println("");
  }
  else ;
}

```

5.2.1.6. Código fuente Comunicación con Flujómetro

Sensirion proporciona los códigos creados por sus desarrolladores en su portal web [37][34] de manera practica y entendible a todos los usuarios que utilizan sus productos en proyectos.

El SLF3S-1300F al comunicarse por medio del protocolo I2C, por lo tanto, requiere de la librería *Wire* incluida en el IDE de Arduino permitiendo la conexión entre el Arduino y el flujómetro. El código contiene constantes por default que son necesarias en todo el algoritmo de lectura y entrega de datos. [Ver código completo en anexo 2]

```

#include <Wire.h> // Arduino library for I2C

const int ADDRESS = 0x08; // Sensor I2C Address
const float SCALE_FACTOR_FLOW = 500.0; // Scale Factor for flow rate
measurement
const float SCALE_FACTOR_TEMP = 200.0; // Scale Factor for temperature
measurement
const char *UNIT_FLOW = " ml/min"; //physical unit of the flow rate
measurement
const char *UNIT_TEMP = " deg C"; //physical unit of the temperature
measurement
float add_flow = 0.0;
float flow_total = 0.0;
float cont = 0.0;

```

El algoritmo entrega una medición directa del sensor, es decir, todas las configuraciones están por defecto.

```

sensor_flow_value = Wire.read() << 8;
sensor_flow_value |= Wire.read();
sensor_flow_crc   = Wire.read();
sensor_temp_value = Wire.read() << 8;
sensor_temp_value |= Wire.read();
sensor_temp_crc   = Wire.read();
aux_value         = Wire.read() << 8;
aux_value         |= Wire.read();
aux_crc           = Wire.read();

```

5.2.1.7. Modelo matemático

Utilizando el principio de funcionamiento y el mecanismo previamente descritos, es posible obtener un modelo matemático para estimar el flujo de gasolina entregado al flujómetro.

$$Q = V_l * A \quad (4)$$

$$A = \pi r^2 \quad (5)$$

Donde,

- r , es la sección circular.

Reemplazando la ecuación 3 y 4 en la ecuación 5.

$$Q = \omega * \varepsilon * \#_{ent} * \pi r^2 \quad (6)$$

Siendo:

- $r = 0.744 [cm]$
- $\varepsilon = 2 \left[\frac{mm}{rev} \right]$.
- $\#_{ent} = 4$

$$Q = 13.91\omega \quad (7)$$

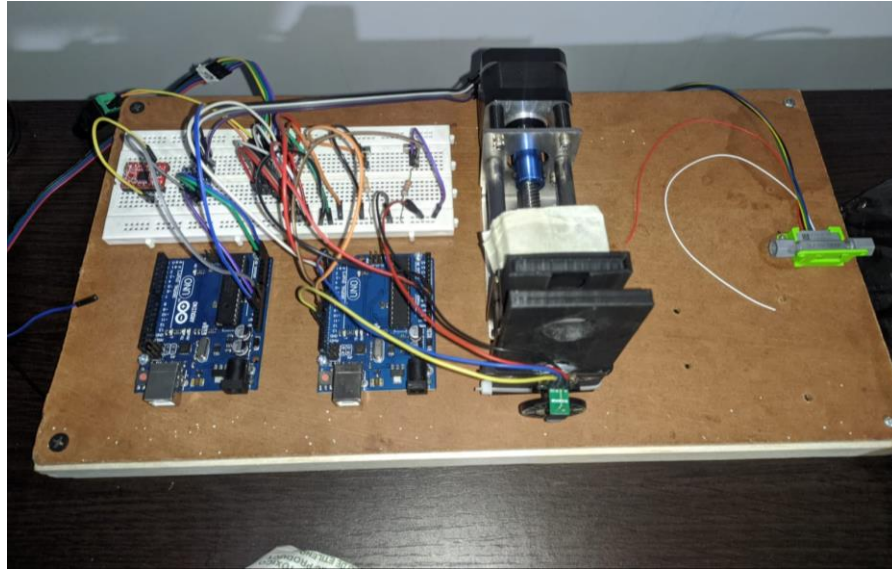
5.2.1.8. Layout

Los diferentes elementos del sistema se ubican fijos sobre una estructura plana con tres partes claramente diferenciadas, una sección de control y electrónica, una sección motriz y la sección del flujómetro, el cual se ubica estratégicamente para prevenir fugas y descargar el fluido en un depósito [Ver anexo 6].

5.2.1.9. Construcción del banco de Calibración

Se realiza la construcción del banco de calibración.

Figura 19. Banco de calibración real.



5.2.1.10. Modelo de Incertidumbre del banco de Calibración

Se realiza un modelo de incertidumbres utilizando la ley de propagación de incertidumbres a partir del modelo matemático del sistema con la ecuación 6 utilizando la guía ISO/IEC para la expresión de incertidumbres en mediciones [56].

$$\mu_i = \frac{\sigma}{\sqrt{N}} \quad (8)$$

$$\mu_{nom} = \sqrt{\mu_{sist}^2 + \mu_{inst}^2 + \mu_{alea}^2 + \mu_c^2} \quad (9)$$

$$\mu_c = \sqrt{\frac{\sum_{i=1}^N (n_i - 1) \mu_i^2}{n_N - N}} \quad (10)$$

Donde:

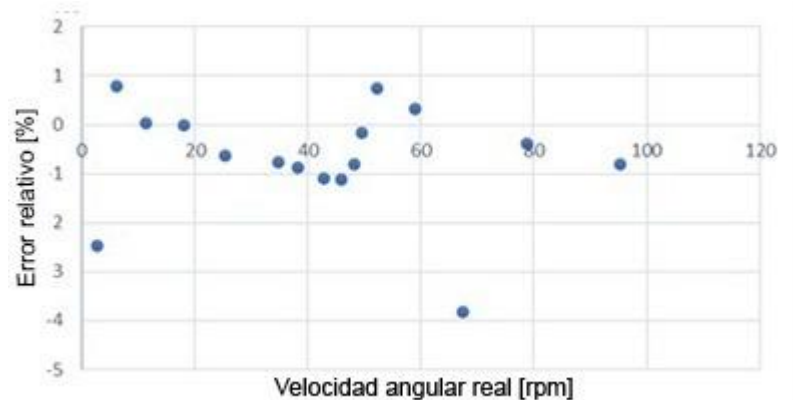
- n_i es el tamaño de la muestra
- N es la cantidad de medidas
- σ es la desviación estándar típica de medida de cada muestra
- μ_c es la incertidumbre combinada
- μ_c es la incertidumbre estándar de medida

La ley de propagación de las incertidumbres se plantea como la ecuación 11.

$$\mu = \sqrt{\sum_{i=1}^N \left(\frac{\partial f}{\partial x_i}\right)^2 \mu_{nom}^2(x_i)} \quad (11)$$

El error de medida de la velocidad angular se puede calcular utilizando un dispositivo para validar las medidas, en este caso se utiliza un tacómetro profesional UNI-T UT372, y los resultados obtenidos se encuentran en la siguiente tabla:

Figura 20. Error relativo porcentual vs velocidad angular real.



Se encuentra un error relativo promedio de -0.69% relacionado con la precisión del dispositivo.

La incertidumbre combinada de la medición de velocidades angulares se realiza monitoreando el comportamiento del sistema bajo diferentes Set Points, con una muestra de 431 datos divididos en 19 conjuntos.

$$N = 431$$

Tabla 14. Incertidumbre combinada.

Vel. Angular [rpm]	σ	μ_i	$\mu_i^2(n_i - 1)$
1.6	0.02698721	0.013494	0.000546
2.5	0.04967972	0.02484	0.001851
3.9	0.23991355	0.097944	0.028779
6.6	0.3436439	0.10867	0.035427
9.7	0.44149641	0.156093	0.073095
12.2	0.8479285	0.226618	0.154068
16.6	1.33491623	0.344674	0.3564
19.7	1.40591495	0.331377	0.329433
22.8	0.43318968	0.108297	0.035185
27.6	2.39599025	0.53576	0.861115
32.3	2.46534905	0.49307	0.729354
39.4	3.1876215	0.563497	0.952587
42.2	2.95463134	0.530667	0.844824
48.4	3.78099761	0.597828	1.072196
56.9	4.10274145	0.625662	1.17436
68.3	4.23205202	0.760099	1.733251
77.2	3.83136271	0.584277	1.02414
83	3.72123077	0.567482	0.966109
88	4.5554094	0.860891	2.223402

Se encuentra una incertidumbre combinada de 0.1717 utilizando la ecuación 10.

$$\mu_c = 0.1717$$

Aplicando la ley de propagación de la incertidumbre al flujo volumétrico medido, se tiene que:

$$f(x) = Q = \omega * \varepsilon * \#_{ent} * \pi r^2$$

$$\frac{\partial Q}{\partial \omega} = \varepsilon * \#_{ent} * \pi r^2 = 0.2 * 4 * \pi * 1.505^2 = 5.69; \sigma_{nom}(\omega) = \sqrt{\mu_c^2} = 0.1717$$

$$\frac{\partial Q}{\partial \varepsilon} = \omega * \#_{ent} * \pi r^2 = 4 * \pi * 1.505^2 = 27.8; \sigma_{nom}(\varepsilon) = \sqrt{\mu_{inst}^2} = \frac{0.012}{1.64} = 0.0073$$

$$\frac{\partial Q}{\partial \#_{ent}} = 0$$

$$\frac{\partial Q}{\partial r} = \omega * \varepsilon * \#_{ent} * 2\pi r = 0.2 * 4 * 2 * \pi * 1.505 = 7.47; \sqrt{\mu_{inst}^2} = 0.0073$$

Utilizando la ecuación 11, se obtiene:

$$\mu = \sqrt{5.69^2 * 0.1717^2 + 27.8^2 * 0.0073^2 + 7.47^2 * 0.0073^2}$$

$$\mu = 0.98 \frac{ml}{min}$$

Debido a que el error de medida se encuentra ubicado sobre el 5% se puede asumir una distribución normal con un nivel de confiabilidad de 95%.

$$U = \mu * k_{@95\%} \quad (12)$$

$$U = 1.92 \frac{ml}{min}$$

Se encuentra una incertidumbre expandida de $\pm 1.92 \frac{ml}{min}$, donde la incertidumbre se define en la guía ISO/IEC para la expresión de Incertidumbre en mediciones utilizando la ley de propagación de incertidumbres [58].

5.2.1.11. Resultados de Calibración

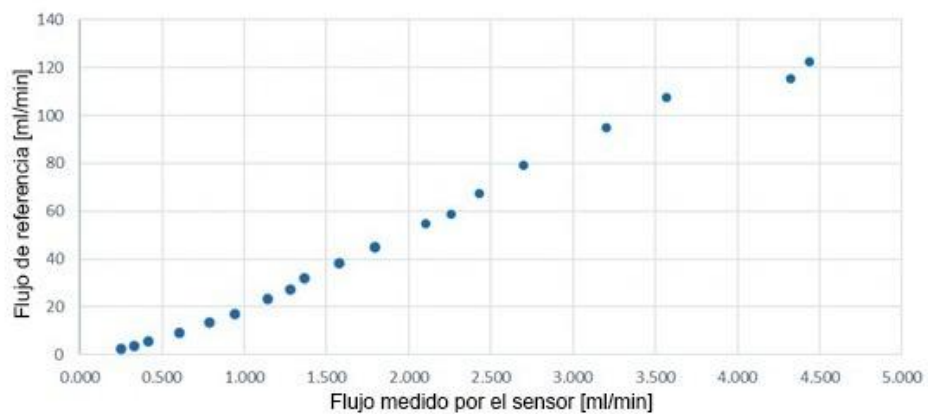
El rango de flujo entregado por el sistema de bombeo se establece entre 0 y 120 ml/min, para cubrir todo el rango teórico de consumo instantáneo de combustible de una motocicleta convencional.

Se realiza toma de datos partiendo de 2 [ml/min] hasta 120 [ml/min] por un periodo de tiempo de 4 segundos, obteniendo:

Tabla 15. Continuación.

Flujo Real [ml/min]	Flujo medido [ml/min]
2.313802894	0.252
3.516746616	0.328
5.44758324	0.414
13.53207351	0.790
16.9831009	0.943
23.09390671	1.143
27.36101562	1.280
31.80420963	1.368
38.42623225	1.576
45.00068348	1.796
54.77370035	2.106
58.68497906	2.256
67.26093481	2.430
79.16257314	2.700
95.01364868	3.202
107.3892024	3.566
115.4445968	4.321
122.3781419	4.439

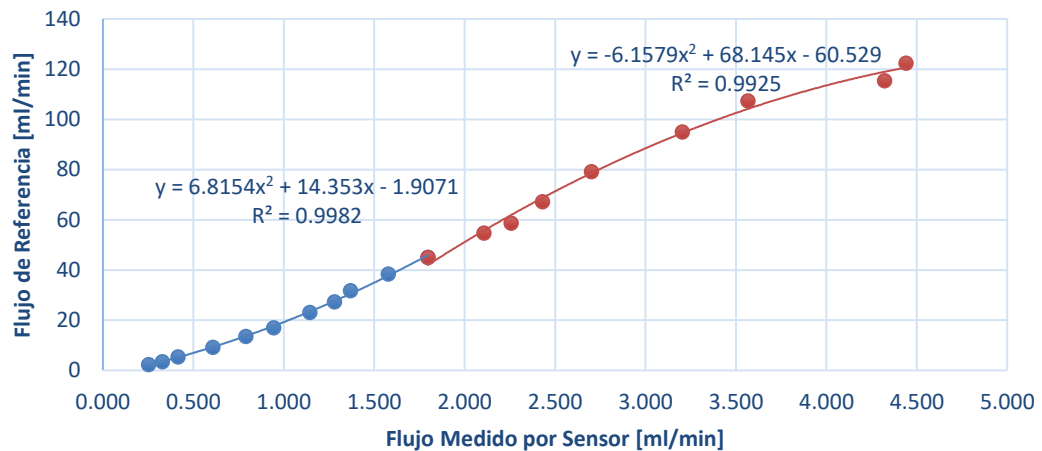
Figura 21. Flujo estimado vs flujo real.



A partir de esto se realiza ajuste de curva para conseguir la función que describe más precisamente el comportamiento del flujómetro con gasolina convencional.

Se encuentra que, seccionando la curva en dos funciones, una función para cubrir los flujos en el rango entre 0 y 40 ml/min y otra para flujos mayores a 40 ml/min. Se consiguen dos funciones de segundo orden con un valor de R2 superior al 0.9925 y error relativo medio de 2%, adicionalmente el flujómetro permite mediciones bidireccionales, pero estas se suprimen debido a que el flujo es unidireccional.

Figura 22. Ajuste cuadrático.



$$Q(q) = \begin{cases} 0, \forall q < 0 \\ 6.8154q^2 + 14.353q - 1.9071, \forall 0 \leq q \leq 40 \\ -6.1579q^2 + 68.145q - 60.529, \forall q > 40 \end{cases} \quad (13)$$

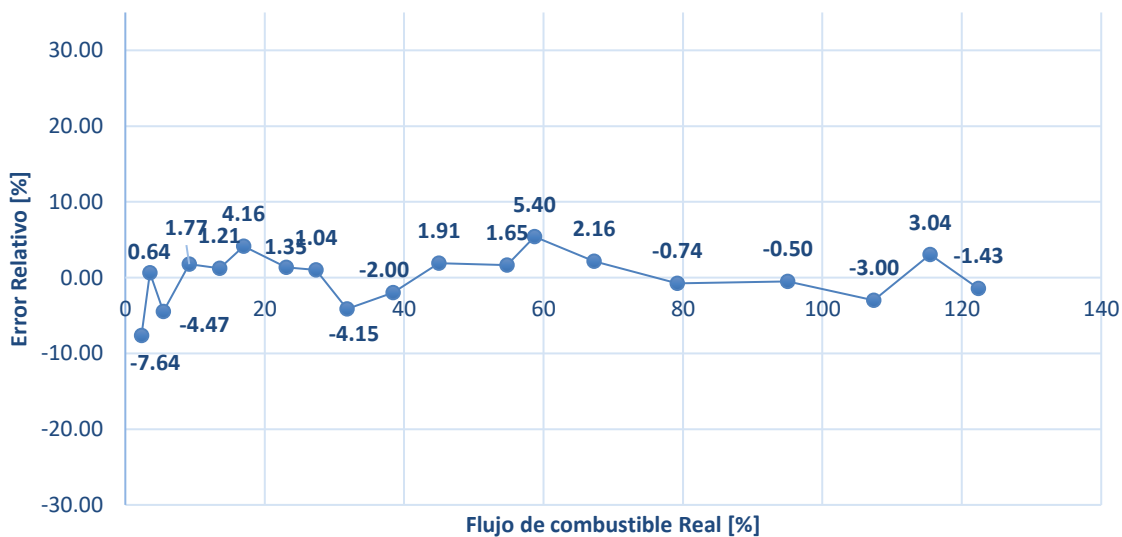
Tabla 16. Comparativa de volumen

Volumen Real [ml]	Salida del Sensor [ml]	Salida Corregida [ml]	Error Relativo [%]
2.313802894	0.252	2.136982332	-7.64
3.516746616	0.328	3.53913657	0.64
5.44758324	0.414	5.203816862	-4.47
9.170853308	0.608	9.333622007	1.77
13.53207351	0.790	13.69647938	1.21

Tabla 16. Continuación.

Volumen Real [ml]	Salida del Sensor [ml]	Salida Corregida [ml]	Error Relativo [%]
23.09390671	1.143	23.40577781	1.35
27.36101562	1.280	27.64440597	1.04
31.80420963	1.368	30.48534466	-4.15
38.42623225	1.576	37.65838991	-2.00
45.00068348	1.796	45.85888881	1.91
54.77370035	2.106	55.6748908	1.65
58.68497906	2.256	61.85321033	5.40
67.26093481	2.430	68.71370509	2.16
79.16257314	2.700	78.57899839	-0.74
95.01364868	3.202	94.54049477	-0.50
107.3892024	3.566	104.1679055	-3.00
115.4445968	4.321	118.9554398	3.04
122.3781419	4.439	120.6335089	-1.43

Figura 23. Error relativo vs flujo de combustible real.

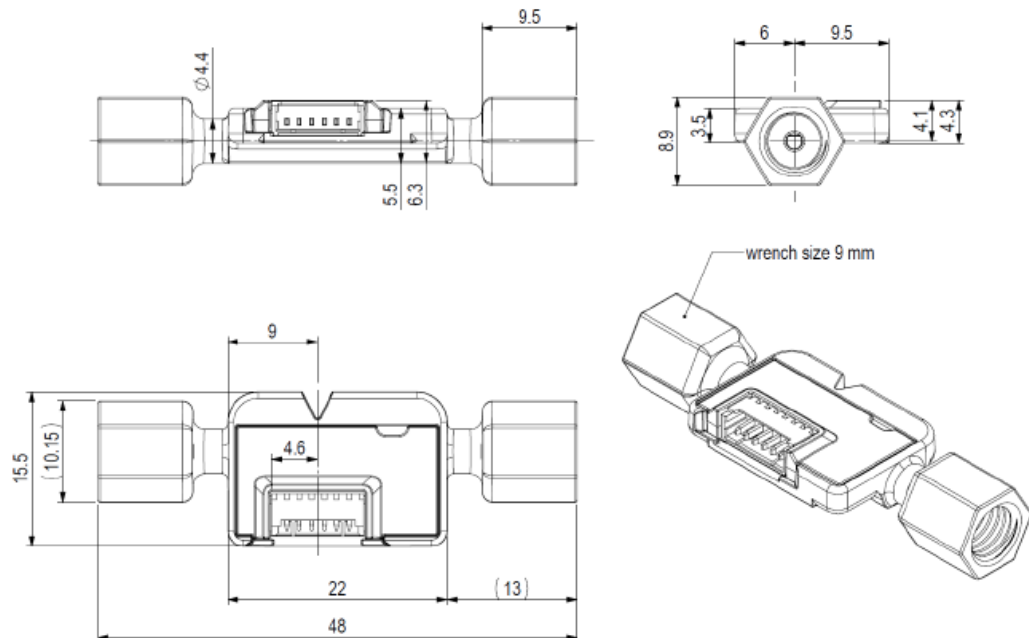


5.2.2. Instalación del flujómetro en línea

Como se definió, el flujómetro debe ir instalado en la línea que comunica el depósito de combustible con el carburador de la motocicleta.

La manguera tiene un diámetro interno de 5 mm y tiene 15 centímetros de longitud, sin embargo, la descarga de combustible se realiza por gravedad, por lo cual se puede realizar una extensión de la longitud de la manguera e instalar el flujómetro.

Figura 24. Dimensiones del SLF3-1300F en [mm].



Tomado de Sensirion [34].

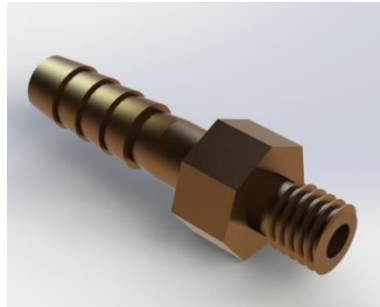
5.2.3. Uniones hidráulicas

Con el objetivo de prevenir posibles fugas de combustible que pudiesen provocar fallas en el circuito interno del flujómetro, es necesario adaptar las conexiones hidráulicas teniendo en cuenta que el flujómetro posee las siguientes dimensiones:

- Conectores: Rosca interna fina 1/4" de fondo plano
- Profundidad de 9 mm
- Diámetro interno de 4.4 mm

Con lo cual se diseña un racor de manguera en SolidWorks [Ver anexo 7].

Figura 25. Diseño CAD del racor



Se maquina la pieza en bronce siendo este un metal maleable con una dureza de entre 70 a 200 HB, lo cual no genera desgaste mecánico en la rosca interna del flujómetro cuyo material es PPS. Se agrega una junta tórica u O-ring para asegurar sello completo en las conexiones hidráulicas.

Figura 26. Racor maquinado.



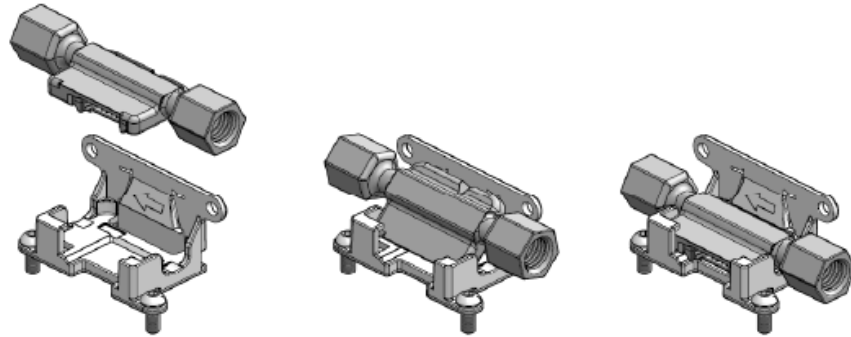
5.2.4. Ubicación

El flujómetro debe ser instalado sobre una superficie plana, rígida, con temperatura estable y bajo riesgo de daño por calor y contaminación química, además, debe permitir con facilidad la conexión-desconexión del mismo.

Una zona adecuada para este propósito que cumple las especificaciones es sobre la cinta aseguradora de la batería, pues esta zona no tiene contacto directo con el motor lo cual mitiga el riesgo de daño por exposición al calor y es una superficie estable.

De esta manera, la instalación del sensor únicamente requiere insertar el flujómetro en la montura de fijación.

Figura 27. CAD del SLF3S-1300F



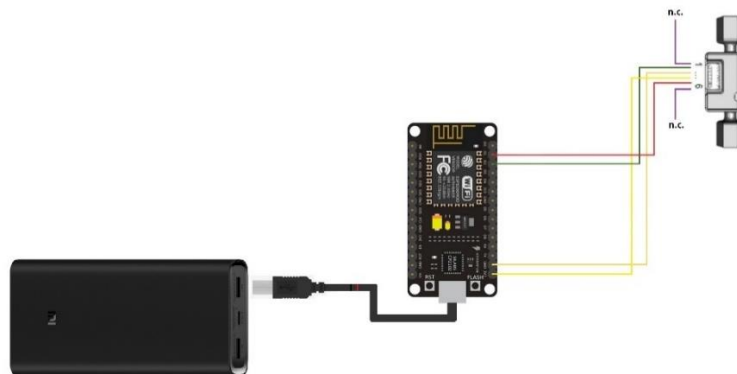
Tomado de Sensirion [34].

5.2.5. Conexiones eléctricas

El flujómetro posee un conector Molex 1.25mm de 6 pines, sin embargo, los pines de los extremos no tienen ningún uso, por lo tanto, se inhabilitan y se realiza la extensión del cable con una terminal Molex 2.54 mm de 4 pines estándar para conectar al dispositivo.

- La alimentación de 3.3 voltios se obtiene a partir de las terminales de 3.3 volts de salida del sistema embebido NodeMCU ESP8266 utilizado como centro de procesamiento del sistema.
- Los pines SDA y SCL se conectan a los pines GPIO05 y GPIO04 habilitados para comunicación por protocolo I2C.

Figura 28. Esquema de conexiones eléctricas.



5.2.6. Código fuente

La posibilidad de que la NodeMCU se programe en el mismo IDE de Arduino y se puedan utilizar todas las librerías del mismo, permite ejecutar el mismo código desarrollado en la sección de 5.2.1.6 con el SLF3S-1300F. De este modo, se utiliza la librería *Wire* permitiendo la conexión por I2C entre la NodeMCU y el flujómetro. Así mismo, el código contiene constantes por default que son necesarias en todo el algoritmo de lectura y entrega de datos.

Posteriormente se realiza el ajuste de corrección obtenido en la calibración [Ver código en anexo 4].

```
scaled_flow_value = ((float) signed_flow_value) / SCALE_FACTOR_FLOW;

if (scaled_flow_value < 0) {
    fuel_flow = 0.0;
} else if (scaled_flow_value >= 0.0 && scaled_flow_value <= 40) {
    fuel_flow = 6.8154 * scaled_flow_value * scaled_flow_value + 14.353
    * scaled_flow_value - 1.9071;
} else if (scaled_flow_value > 40) {
    fuel_flow = -3.9493 * scaled_flow_value * scaled_flow_value +
    52.444 * scaled_flow_value - 37.27;
}
```

5.3. MEDICIÓN DE VELOCIDAD ANGULAR DEL MOTOR

La alternativa de solución con un mayor impacto sobre el problema de medir las revoluciones por minuto del motor de la motocicleta es un tacómetro, sin embargo, no existe disponibilidad actual en el mercado de un transmisor de esta señal que permita la lectura por parte de una unidad de procesamiento como un microcontrolador.

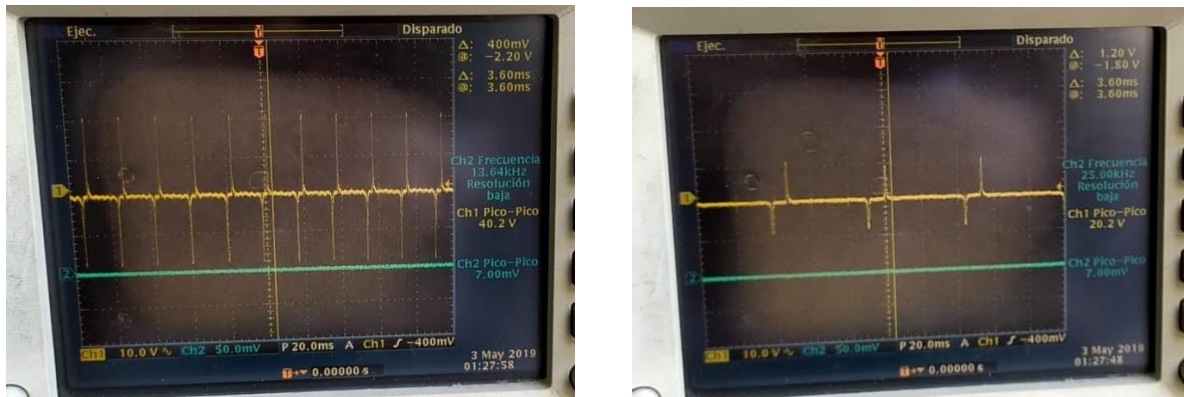
Es por ello que se opta por el diseño de un circuito electrónico capaz de tomar alguna señal disponible en la motocicleta y convertirla en una señal útil para la lectura de la velocidad angular del motor.

Como se describió previamente la señal proveniente de la bobina captora de la motocicleta es una señal eléctrica de un sensor ubicado en el cigüeñal del motor capaz de enviar un pulso hacia el CDI para sincronizar la señal de disparo de la bujía cada vez que el eje del motor da una vuelta completa.

5.3.1. Caracterización de señal:

Se realiza una caracterización de la señal proveniente de la bobina pulsante de la motocicleta utilizando multímetro y osciloscopio.

Figura 29. Señal de la bobina pulsante



La señal capturada es una señal periódica con un pulso negativo y otro positivo de la misma amplitud ambos la cual aumenta con la propia frecuencia entre 10 y 40 voltios, además se encuentra un rizado constante de alta frecuencia y una tensión de entre 3 volts.

La frecuencia de la señal coincide con la frecuencia de giro del cigüeñal del motor de la motocicleta, por lo tanto, es posible utilizar esta señal para conseguir el dato de velocidad angular del motor.

5.3.2. Diseño de Circuito Electrónico:

Tabla 17. Características de la señales de velocidad angular.

	Señal de Entrada	Señal de Salida
Forma	Pulsante	Cuadrada
Amplitud [v]	10-40	0-3.3
Frecuencia [Hz]	0-200 Hz	0-200 Hz
Periodo [ms]	0-5	0-5

El circuito debe transformar la señal de entrada proveniente de la bobina pulsante de la motocicleta en una señal cuadrada de 0 a 3.3 volts con la misma frecuencia y una duración suficiente para que el microcontrolador pueda detectar el cambio de nivel.

5.3.2.1. Filtrado

Debido a que la señal de entrada posee un rizado considerable de por lo menos tres volts a alta frecuencia se hace necesario el uso de un filtro pasa altas capaz de atenuar las componentes de alta frecuencia.

Este filtro debe poseer una frecuencia de corte mayor a la frecuencia mayor de la señal de revoluciones del motor con una tolerancia establecida para no generar atenuación de la parte útil de la señal proveniente de la bobina captora.

$$f_{corte} = 200 \text{ Hz}$$

Sin embargo, se debe agregar una tolerancia de por lo menos el 50% en dado caso que la señal cambie con las condiciones, con lo cual la frecuencia de corte debe ser de 300 Hz

Con esto el filtro pasa altas queda conformado de la siguiente forma:

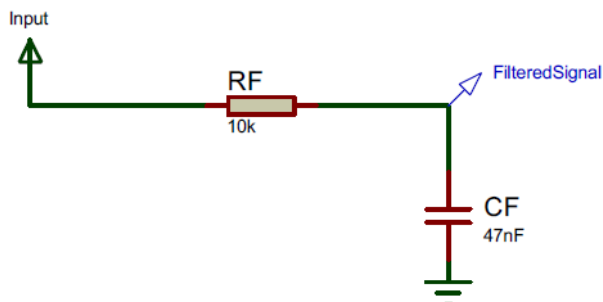
$$R_f = 10 \text{ k}\Omega$$

$$C_f = \frac{1}{2\pi R f_c} \quad (14)$$

$$C_f = \frac{1}{2\pi(10000)(300)} = 53 \text{ nF}$$

Con lo cual se opta por un condensador comercial de un valor cercano a 53 nF; Se selecciona un condensador cerámico puesto que estos atenúan en mayor medida las inductancias parásitas que pueden presentarse.

Figura 30. Esquemático de la señal de filtrado.



5.3.2.2. Disparo

Aprovechando la propiedad de los transistores que permite utilizarlos como un switch electrónico capaz de permitir o no que se entregue una señal de referencia por medio de una corriente de base que controla el disparo se desarrolla el circuito.

Un transistor BJT posee dos estados de interés, el estado de corte y el estado de saturación, estos dos estados permiten o no la corriente entre las terminales colector-emisor por medio de una señal de accionamiento en la base que suele ser una señal de corriente.

La señal de tensión a la salida del filtro debe ser, por lo tanto, convertida a una señal de corriente capaz de alternar entre el estado de corte y saturación del transistor. Por lo tanto, se selecciona el transistor BJT NPN BC337, el cual posee una ganancia variable entre 100 y 630.

Se diseña un circuito que utiliza la señal de la bobina pulsante ya filtrada y la convierte en una señal de corriente que conmute al transistor y entregue una señal de disparo cuadrada entre dos niveles lógicos de 3.3 volts.

$$R_c = 1k\Omega$$

$$V_{cc} = 3.3 [v]$$

Cuando $v_{CE} = 0$, utilizando la ley de Ohm se obtiene,

$$i_c = \frac{3.3}{1k} = 3.3mA$$

$$i_b = \frac{i_c}{\beta} \quad (15)$$

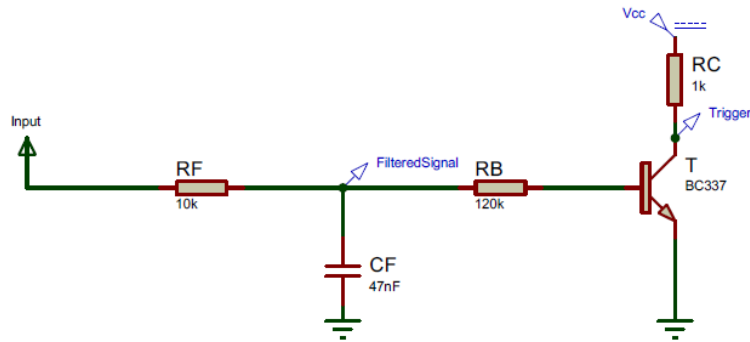
$$i_b = \frac{3.3mA}{100} = 33\mu A$$

Se desea que el transistor conmute siempre que la señal de la bobina sea mayor que cuatro volts para evitar falsos disparos.

$$R_b = \frac{v_b}{i_b} \quad (16)$$

$$R_b = \frac{4}{33\mu A} = 120k\Omega$$

Figura 31. Esquemático de la etapa de filtrado y disparo.



5.3.2.3. Regulación

La regulación de la señal se realiza con el objetivo de preservar la estabilidad e integridad del microcontrolador, pues, si bien el circuito de conversión entrega una señal leíble, esta conmuta hacia un estado bajo una vez ocurre un disparo de entrada y puede ser susceptible a falsos disparos, para esto se agrega un temporizador NE555 configurado como monoestable.

Este circuito es capaz de mantener una señal de salida en estado bajo y conmutar a estado alto por un periodo de tiempo configurable o cual permite establecer un rango de medición de la velocidad angular del motor. El tiempo de duración de la señal de salida en estado alto está determinado por la siguiente expresión:

$$T = 1.1RC \quad (17)$$

Se desea cubrir todo el rango de velocidades para el motor de la motocicleta, es por ello que se establece un valor sobre los 15000 RPM, para mayor comodidad se agrega un tiempo de tolerancia y se seleccionan los componentes de la siguiente manera

$$RPM_{corte} = 20000$$

$$R = 5600$$

$$t = \frac{60}{20000} = 3ms \rightarrow t_{lim} = 2.5ms$$

Reemplazando los datos en la ecuación 17, se obtiene:

$$C = \frac{0.0025}{1.1 \times 5600} = 405nF$$

$$C = 390nF$$

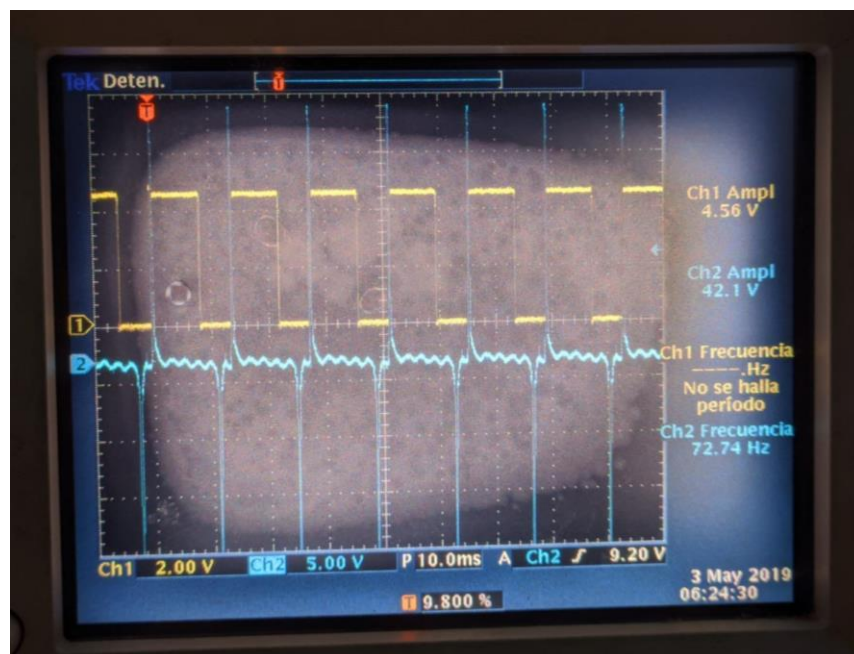
Para el uso con temporizadores se recomienda utilizar condensadores de poliéster debido a su baja pérdida de capacitancia y alta estabilidad.

Con este circuito es posible obtener una señal estable, segura y leíble por parte del sistema embebido NodeMCU con el cual se programa una interrupción en el pin D5 de la placa, el cual captura el cambio de nivel bajo a alto y temporiza el tiempo entre ocurrencias.

Una vez ocurre un disparo, el circuito transforma esa señal en una onda cuadrada que conmuta a estado alto durante 2.5 ms y vuelve a nivel bajo, el microcontrolador es capaz de detectar el flanco de subida de la señal e inicializar un temporizador que cuenta el tiempo entre cada interrupción y realiza el siguiente cálculo sabiendo que cada vuelta del cigüeñal se traduce en una conmutación y el tiempo se cuenta en milisegundos,

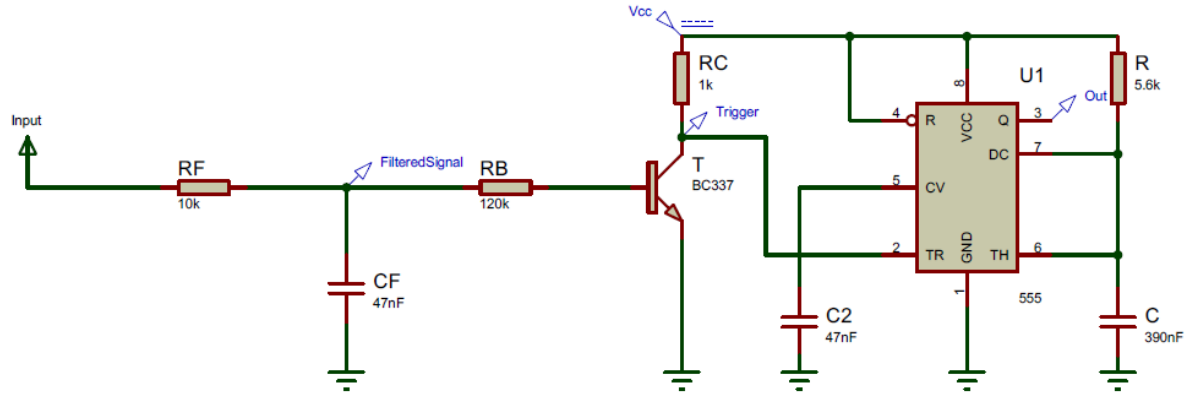
$$RPM_{real} = \frac{1}{t_{inicial} - t_{final}} \times 60000 \quad (18)$$

Figura 32. Señal de pulsos y señal transformada.



Por lo tanto, todo el circuito adaptador de señal proveniente de la bobina pulsante es:

Figura 33. Esquemático de etapas de filtrado, disparo y regulación de la señal.



5.3.2.4. Instalación en la Motocicleta

La bobina pulsante de la motocicleta se comunica con el CDI por medio de dos cables, un cable de datos y un cable de tierra. El cable de datos es un cable calibre 18 AWG al cual es posible instalarle un conector de autodesforre o roba corrientes para sacar una terminal de empalme rápido tipo faston hembra para facilidad de instalación del dispositivo.

La tierra de todo el circuito electrónico es común, por lo tanto, es necesario cablear la masa de la motocicleta y dado que esta es común a la carcasa del motor, de esta manera con un cable con terminal tipo pinza se puede crear la tierra común con el circuito electrónico.

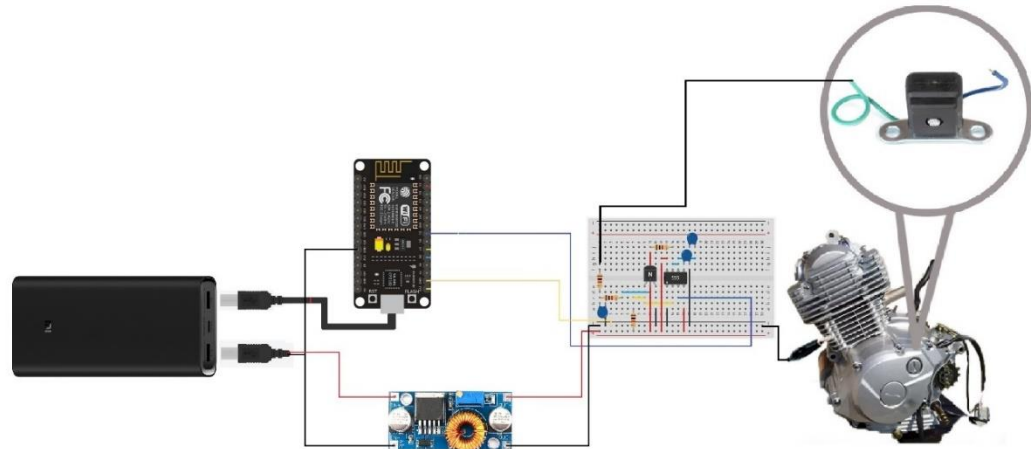
5.3.3. Conexiones eléctricas

La batería utilizada para alimentar el sistema es una batería de litio con dos salidas de tensión de 5 voltios reguladas por un microchip, con estos 5 voltios se alimenta al sistema embebido que controla a todo el sistema por medio del conector microUSB, sin embargo, debido a que las características de lectura de los pines de la placa nodeMCU son de 3.3 voltios máximo, el circuito adaptador de señal debe regular la tensión aplicada en Vcc a 3.3 voltios.

Para disminuir el nivel de tensión se utiliza un módulo regulador de tensión XL4005, el cual regula la tensión de salida con un dimmer propio del módulo.

La señal proveniente de la bobina pulsante es captada, procesada y se entrega como un pulso de 0-3.3 voltios a la placa NodeMCU que realiza el cálculo de la velocidad angular del motor en revoluciones por minuto.

Figura 34. Diagrama electrónico de conexiones del circuito adaptador de señal de pulsos.



5.3.4. Código fuente

La señal proveniente de la bobina pulsante es de alta frecuencia, por consiguiente, requiere de ejecuciones rápidas para captar y transformar la frecuencia en unidades de velocidad angular. El programa lee la señal externa sin importar el segmento de código que esté ejecutando y se realiza una interrupción incluso si está a mitad de un delay().

Por lo tanto, la interrupción permite pausar la ejecución de todo el programa principal para atender la interrupción en el pin D5 (GPIO14), y se ejecute otro método llamado calculateRPM() .

```
pinMode(14, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(14), calculateRPM, RISING);

ICACHE_RAM_ATTR void calculateRPM() {
  if ( (micros() - debounce > 100) ) {
    debounce = micros();
    timeold = millis(); // Almacenamos el tiempo actual.
    float t = (timeold - time18D) / 1000;
    rpm = 360 / (t * 6);
    rpm_filter = alpha * rpm + (1.0 - alpha) * rpm_filter;
    time18D = timeold;
  }
}
```

La interrupción está configurada para activarse en modo RISING, es decir, para todos los flancos de subida de la señal de entrada [Ver código en anexo 4].

5.4. MEDICIÓN DE POSICIÓN Y VELOCIDAD

Con el fin de obtener las dos variables de manera objetiva, efectiva y con una alta confiabilidad se opta por trabajar con el módulo GPS Beitian BN-220 utilizado para el control de vuelo en drones. El dispositivo obtiene las coordenadas de posicionamiento rápidamente gracias a sus capacidades técnicas. Se destaca la tecnología de conectividad RS-232, la incorporación de memoria Flash para su configuración y la antena pasiva ubicada al costado de todo el chip integrado.

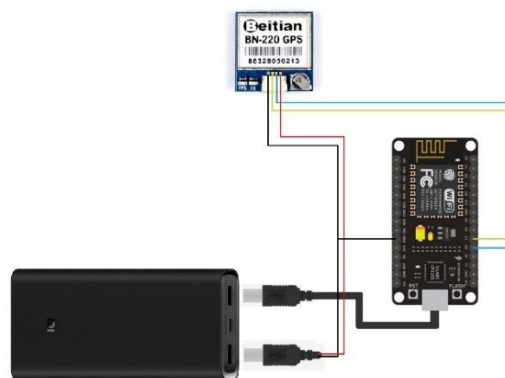
Los 72 canales de búsqueda que posee el Beitian BN-220 permiten tomar señal rápidamente de varios satélites que orbitan la tierra. Identificando los puntos en el espacio (latitud, longitud y altitud) se extrae información de velocidad, posición y pendiente. El dispositivo cuenta con configuraciones estándar por defecto, de manera que, es apto para utilizar en cualquier tarjeta de desarrollo con microcontrolador, como es el caso en la NodeMCU.

5.4.1. Conexiones eléctricas

El GPS posee un conector Molex 1.25mm de 4 pines, correspondientes a su alimentación y transmisión de datos.

- La alimentación de 5 voltios se obtiene directamente de la fuente de poder de todo el sistema.
- El transmisión y recepción de datos con la NodeMCU lo realiza mediante el protocolo UART. Esta comunicación en serie es a través de los pines TX y RX que se conectan a los pines GPIO13 y GPIO15 habilitados para comunicar por este tipo de protocolo.

Figura 35. Conexiones eléctricas para el GPS.



5.4.2. Código fuente

Se utiliza la librería TinyGPS++ que proporciona los métodos necesarios que reciben datos del GPS Beitian BN-220. Además, solo se toman en cuenta las variables de estudio (latitud, longitud, altitud y velocidad) [Ver código en anexo 4].

```
#include <TinyGPS++.h>
TinyGPSPlus gps;
static const int RXPin = 13, TXPin = 15;
static const uint32_t GPSPin = 9600;
float spd;
float latitude;
float longitude;
float altitud;
unsigned int move_index = 1;
float lastLong, lastLat;
```

Después de estar declaradas las variables de importancia, se extraen los datos con la clase “gps”.

```
latitude = (gps.location.lat());
longitude = (gps.location.lng());
altitud = (gps.altitude.meters());
spd = (gps.speed.kmph());
```

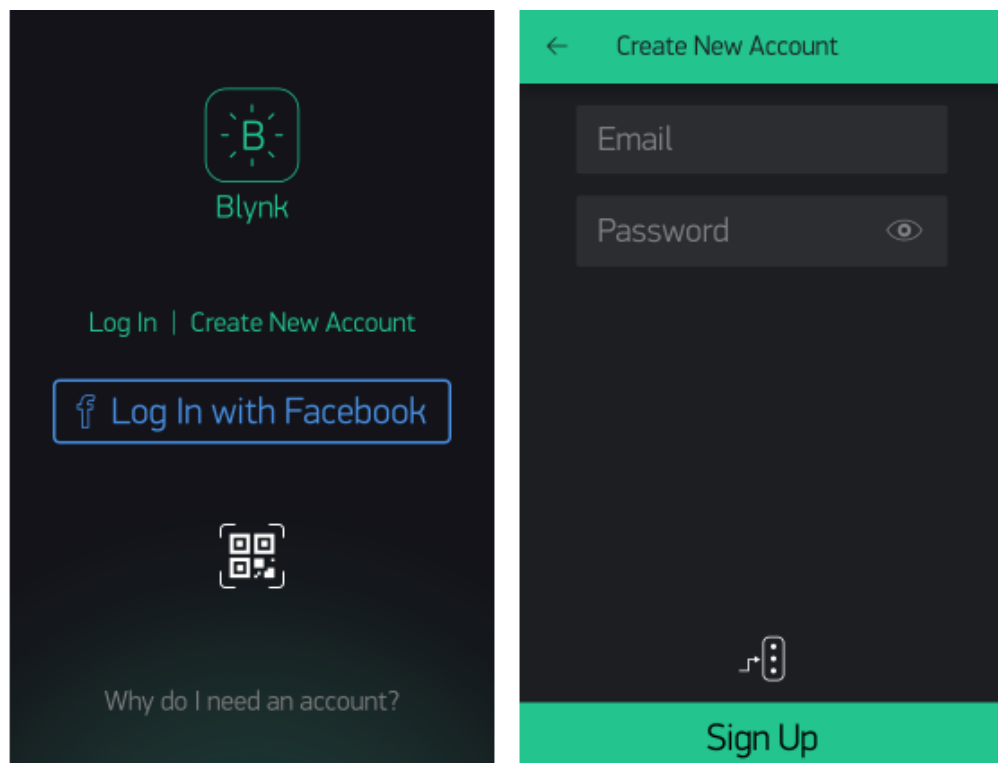
5.5. DESARROLLO DE LA APLICACIÓN MÓVIL PARA VISUALIZACIÓN

Como parte de los objetivos planteados, se desarrolla una aplicación móvil utilizando la plataforma de Blynk para la visualización de los datos de ruta en tiempo real.

5.5.1. Interfaz

Crear una aplicación en Blynk es simple, como se mencionó previamente, la plataforma IoT está disponible para Android y iOS de forma gratuita. Se parte de la instalación de la aplicación y creación de la cuenta.

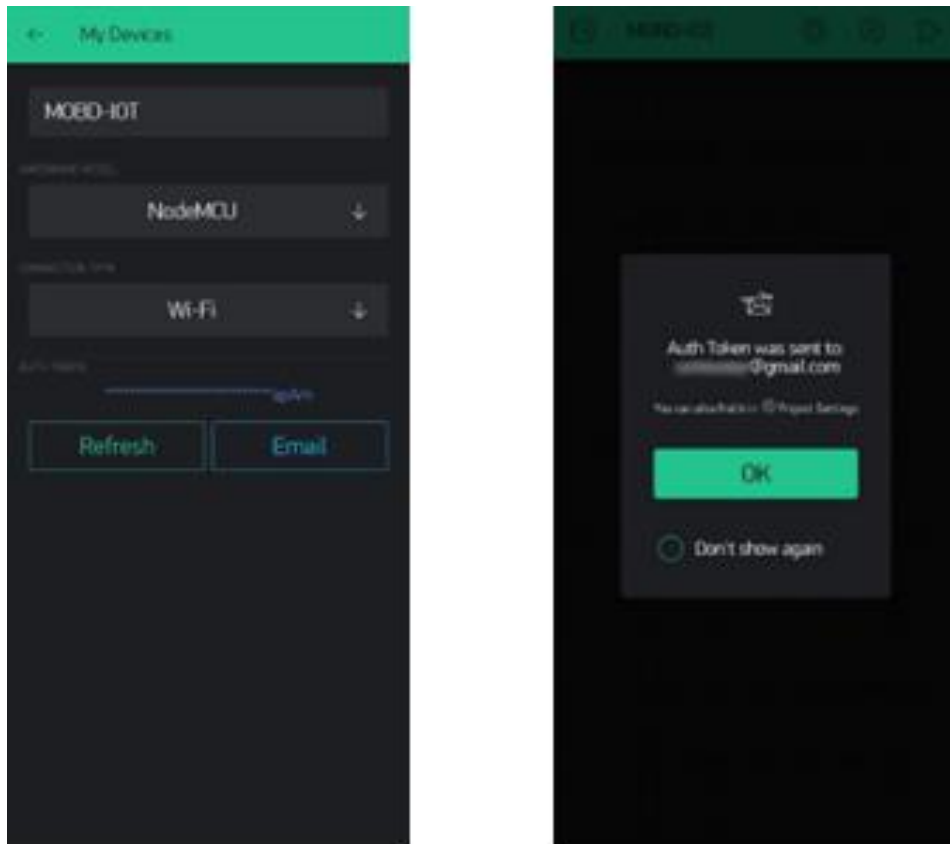
Figura 36. Registro en la plataforma de Blynk.



En la parte superior de la app se toca el símbolo “+” para crear el proyecto. Se le asigna el nombre al proyecto para posteriormente configurarlo según los detalles del hardware que se utiliza; en este caso se trabaja con la placa NodeMCU que se conecta utilizando red inalámbrica WiFi.

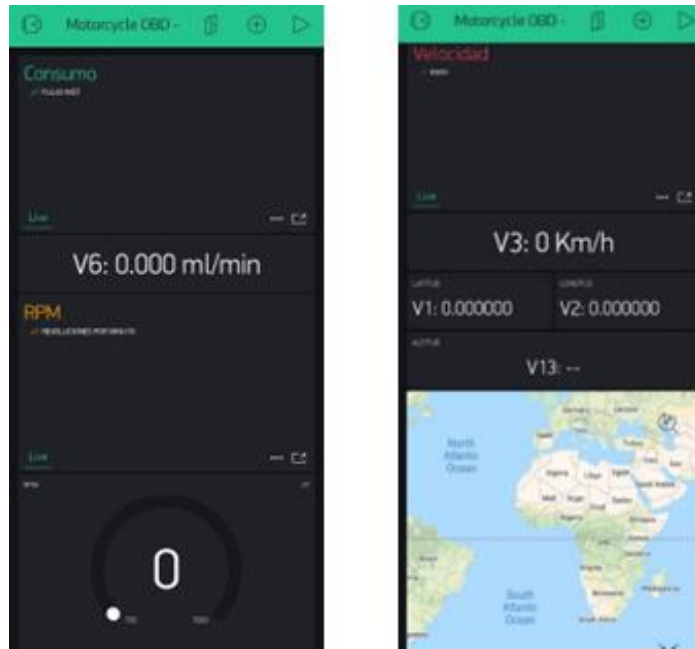
Una vez creado el proyecto, Blynk envía un mail con un “auth token” equivalente a una clave importante de autenticación e identificación que vincula la app con el hardware.

Figura 37. Registro de Dispositivos en el entorno de Blynk.



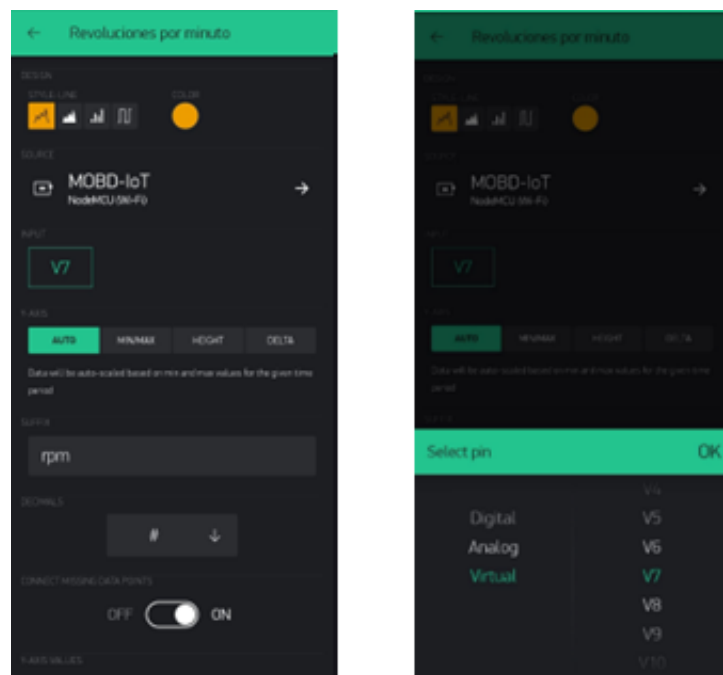
Blynk posee múltiples widgets para diseñar la interfaz, pueden ser de tipo display de control, notificaciones, entre otros. Es importante aclarar que cada uno tiene un costo de “energía”. Blynk brinda de forma gratuita 2000 unidades que se consumen a medida que se agregan elementos. Sin embargo, en caso de ser necesario, adicionar “energía” tiene un valor en USD. Los widgets utilizados son: labels, gauge, superchart (gráficas en tiempo real) y mapa.

Figura 38. Widgets añadidos a la aplicación móvil



La configuración de cada widget es sencilla e intuitiva para la modificación de sus funciones. Además, se tiene en cuenta el pin virtual que se asigna en Blynk debe coincidir con el asignado en la NodeMCU por código.

Figura 39. Configuración de los Widgets



5.5.2. Código fuente

Programar y enlazar Blynk con la NodeMCU, se dispone de dos librerías principales *BlynkSimpleEsp8266* y *WifiManager* correspondientes a la comunicación con la plataforma y el WiFiProvisioning [Ver código en anexo 4].

5.5.2.1. Configuración WiFi Provisioning

Esta funcionalidad es el medio por el cual la placa puede conectarse a cualquier red WiFi con las credenciales de red requeridas.

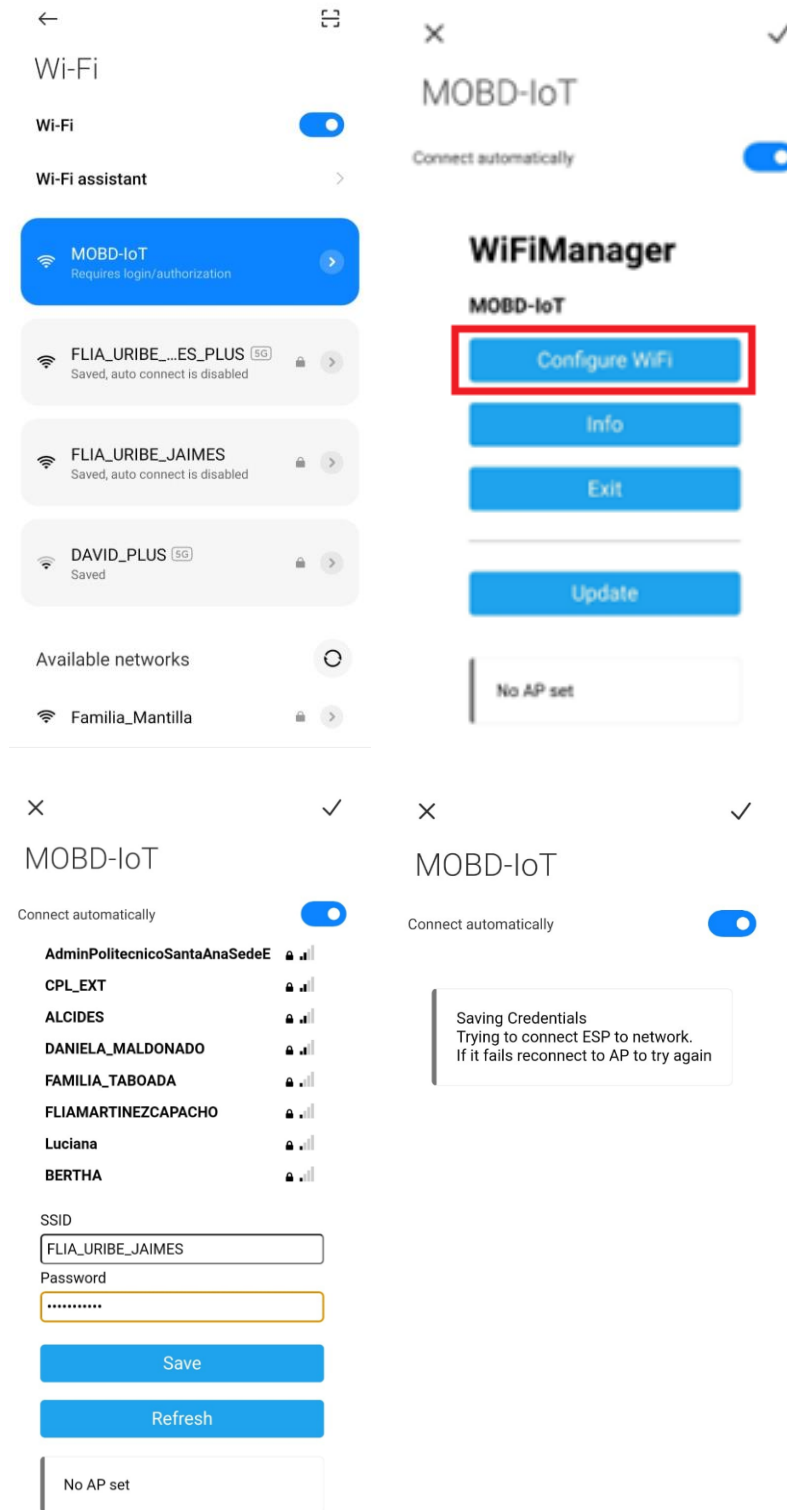
```
#include <WiFiManager.h>
```

Con la librería *WifiManager* se establece la NodeMCU como un punto de acceso (AP, Access Point en inglés) titulado "MOBD-IoT". Este AP fija la conexión inalámbrica a internet.

```
Serial.begin(115200);  
WiFiManager wifiManager;  
wifiManager.erase();  
wifiManager.autoConnect("MOBD-IoT", "");
```

El punto de acceso es generado y se visualiza en las redes de WiFi disponibles. Con el smartphone se le brindan las credenciales de red a la cual se quiere conectar la placa.

Figura 40. Introducción de Credenciales de Red



5.5.2.2. Configuración Blynk

La librería *BlynkSimpleEsp8266* se encuentra disponible en la página web oficial de Blynk [docs.blynk.cc], y habilita diferentes funciones, entre esas, para ejecutar la transmisión de datos.

```
#include <BlynkSimpleEsp8266.h>
```

Como se mencionó anteriormente, el pin virtual que fue asignado en la interfaz Blynk debe coincidir con el ingresado en código.

```
// Write RPM
Blynk.virtualWrite(V7, String(int(rpm)));

// Write LONGITUDE & LATITUDE & DISTANCE
Blynk.virtualWrite(V1, String(latitude, 7));
Blynk.virtualWrite(V2, String(longitude, 7));
Blynk.virtualWrite(V13, String(altitud, 7));

// Write SPEED
Blynk.virtualWrite(V3, int(sp));

// Write Flow
Blynk.virtualWrite(V6, String(fuel_flow, 3));
```

Por último, Blynk es gratuito para un uso básico y no comercial. El uso comercial o funciones más avanzadas como crear una app independiente para su publicación en AppStore o PlayStore a partir de un proyecto, están disponibles sólo en planes pago

5.6. DESARROLLO DE BASE DE DATOS EN LA NUBE

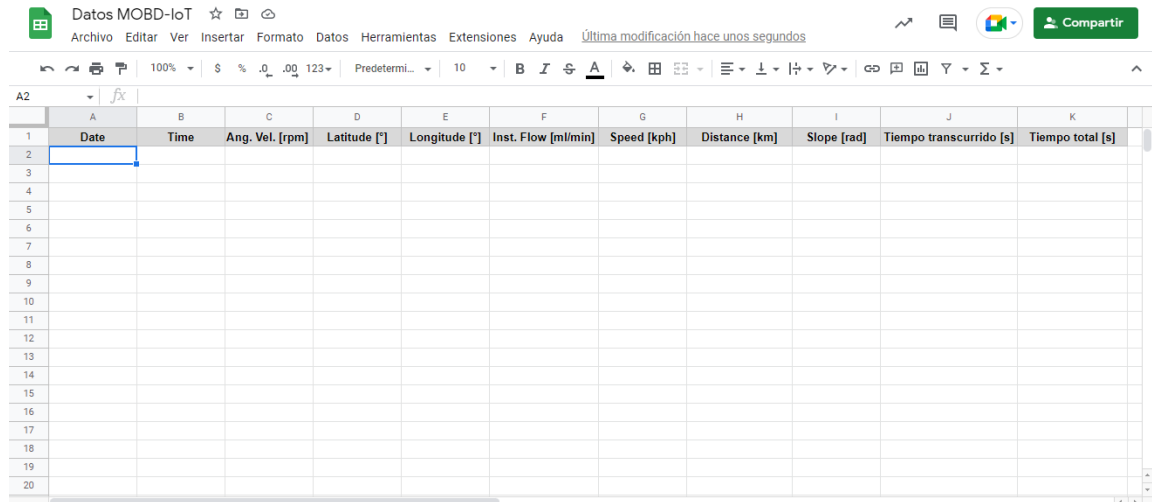
La gran posibilidad de que la nodeMCU pueda enviar datos a los servicios de GoogleSheets de manera gratuita y directa, es gracias al protocolo HTTPS. Como se mencionó previamente, para hacer uso de dichos servicios es necesaria la librería *WiFiClientSecure* que trae consigo todo el algoritmo útil para proyectos de este tipo. Así mismo, para trabajar con los servicios de Google es necesario desarrollar un programa en Apps Script para conectarse con la herramienta SpreadSheets.

El desarrollo de la base de datos se divide en la creación de la base de datos en los servicios de Google y en el envío de los datos provenientes de la placa.

5.6.1. Configuración Google Apps Script

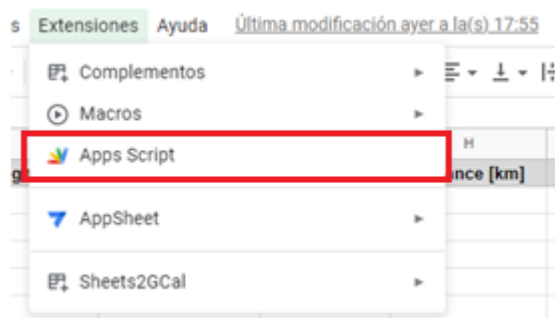
Inicialmente se crea el archivo que funcionara como base de datos. Para esto se tiene en cuenta la ubicación en Drive donde queda almacenado.

Figura 46. Base de Datos en Google Sheets



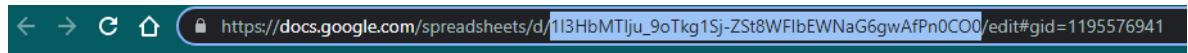
Después de creado el archivo de SpreadSheets se abre la plataforma Apps Script.

Figura 47. Configuración de Google Apps Script



En el desarrollo del programa en el entorno de Apps Script se tiene en cuenta el SpreadSheet ID, es decir, un fragmento del enlace de la hoja de cálculo creada:

Figura 48. SpreadSheet ID.



```
function doGet(e) {
  Logger.log( JSON.stringify(e) );
  var result = 'OK';
  if (e.parameter == 'undefined') {
    result = 'No Parameters';
  }
  else {
    var sheet_id = '1I3HbMTIju_9oTkg1Sj-
ZSt8WFibEWNaG6gwAfPn0CO0'; // Spreadsheet ID
    var sheet = SpreadsheetApp.openById(sheet_id).getActiveSheet();
    var newRow = sheet.getLastRow()+1;
    var rowData = [];
    var Curr_Date = new Date();
    rowData[0] = Curr_Date; // Date in column A
    var Curr_Time = Utilities.formatDate(Curr_Date, "Asia/Jakarta", 'HH:mm:ss');
    rowData[1] = Curr_Time; // Time in column B
```

Se tienen en cuenta todas las variables de interés (velocidad angular, latitud, longitud, altitud, flujo y velocidad) para ser guardadas en la base de datos [Ver anexo 3]:

```
switch (param) {
  case 'rpm':
    rowData[2] = value; // Temperature in column C
    result = 'RPM Written on column C';
    break;
  case 'lat':
    rowData[3] = value; // Humidity in column D
    result += ' ,Latitude Written on column D';
    break;
  case 'lon':
    rowData[4] = value; // Humidity in column D
    result += ' ,Longitude Written on column D';
    break;
  case 'alt':
    rowData[5] = value; // Humidity in column D
    result += ' ,Altitud Written on column D';
    break;
  case 'flo':
```

```

    rowData[6] = value; // Humidity in column D
    result += ' ,Flow Written on column D';
    break;
case 'spe':
    rowData[7] = value; // Humidity in column D
    result += ' ,Speed Written on column D';
    break;
default:
    result = "unsupported parameter";
}

```

Posteriormente se realiza la implementación del script como aplicación web, teniendo en cuenta que el acceso tiene que ser a cualquier usuario para no generar problemas a futuro.

Figura 49. Implementación del programa de Google Apps Script.

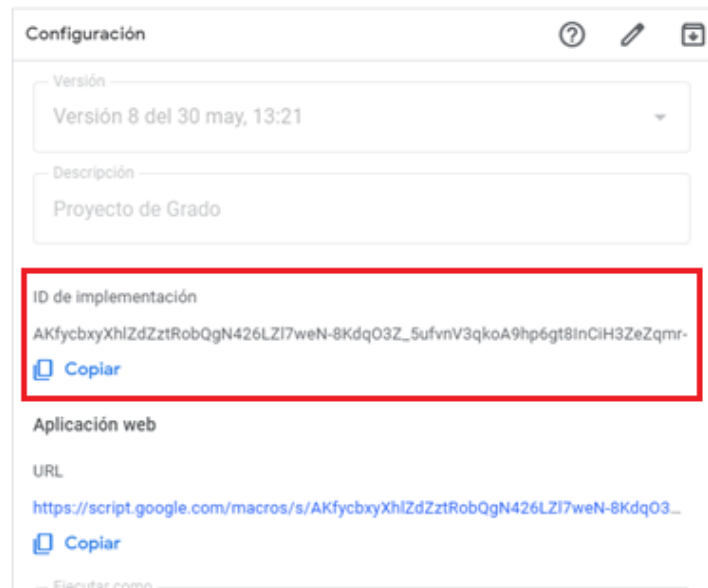
Nueva implementación

Seleccionar tipo	Configuración
Aplicación web	<p>Descripción</p> <p>Nueva descripción</p> <p>MOBD-IoT Project</p> <p>Aplicación web</p> <p>Ejecutar como</p> <p>Yo (suribe62@unab.edu.co)</p> <p>La ejecución de la aplicación web se autorizará con los datos de tu cuenta.</p> <p>Quién tiene acceso</p> <p>Cualquier usuario</p> <p>También se puede usar como biblioteca. Más información</p>

Cancelar Implementar

La ID de implementación del script se tiene en cuenta para el envío de datos desde la NodeMCU:

Figura 50. ID de la implementación.



5.6.2. Configuración NodeMCU ESP8266

La facilidad de trabajar con la librería *WiFiClientSecure* es la ejecución de todos los servicios del protocolo HTTPS, que a su vez enlaza directamente con el script de Google creado con la ID de implementación.

```
const char* host = "script.google.com";
const int httpsPort = 443;
WiFiClientSecure client; //--> Create a WiFiClientSecure object.

String GAS_ID = "AKfycbxyXhlZdZztRobQgN426LZl7weN-8KdqO3Z_5ufvnV3qkoA9hp6gt8InCiH3ZeZqmr-";
```

El objeto *HTTPSRedirect* proveniente de la librería utiliza la información de la ID de implementación en la respuesta del servidor a una solicitud *HTTP* tipo *GET*, y sigue la URL redirigida haciendo otra solicitud *GET* adecuada cada vez que las variables de interés tengan un dato.

```

void sendData(float latitude, float longitude, float altitud, int spd,
float rpm, float flow) {
    if (!client.connect(host, httpsPort)) {
        return;
    }
    String string_latitude = String(latitude, DEC);
    String string_altitud = String(altitud, DEC);
    String string_longitude = String(longitude, DEC);
    String string_spd = String(spd, DEC);
    String string_rpm = String(rpm, DEC);
    String string_flow = String(flow, DEC);

    String url = "/macros/s/" + GAS_ID + "/exec?rpm=" + string_rpm +
"&lat=" + string_latitude + "&lon=" + string_longitude + "&alt=" +
string_altitud + "&flo=" + string_flow + "&spe=" + string_spd;

    client.print(String("GET ") + url + " HTTP/1.1\r\n" +
        "Host: " + host + "\r\n" +
        "User-Agent: BuildFailureDetectorESP8266\r\n" +
        "Connection: close\r\n\r\n");
}

```

5.7. DISEÑO DE TARJETA DE CIRCUITO IMPRESO PCB

Con la determinación de todos los elementos electrónicos se realiza el diseño de la tarjeta de circuito impreso en el software Proteus [Ver anexo 5]. Para añadirle calidad al proyecto, se realiza el diseño bajo las normas IPC 2220. Aplicando la norma se le aporta robustez, confiabilidad y rendimiento a la tarjeta impresa, facilitando el ensamblaje y conexiones soldadas.

Figura 51. Esquema de conexiones en la Placa de Circuito Impreso PCB.

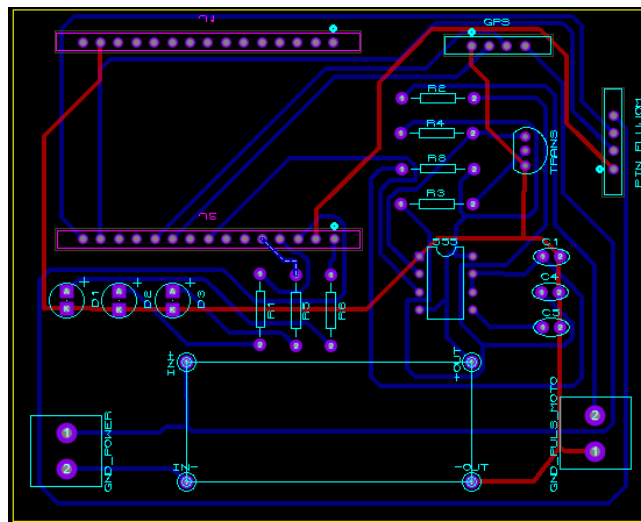
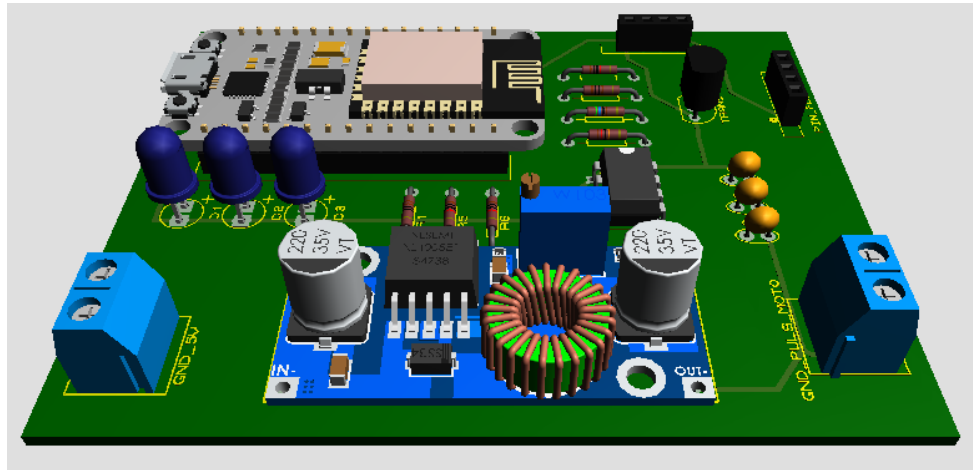
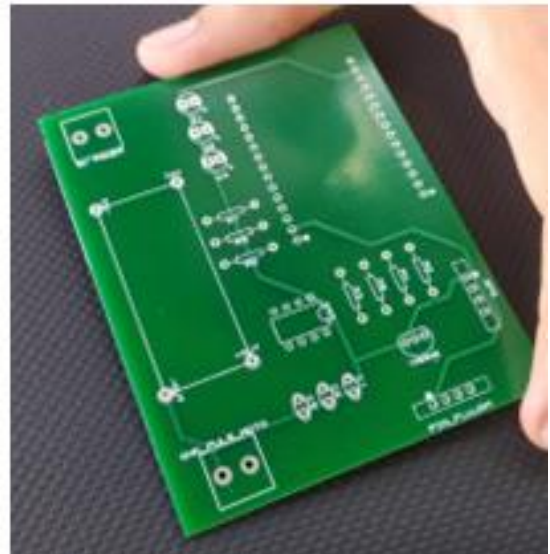
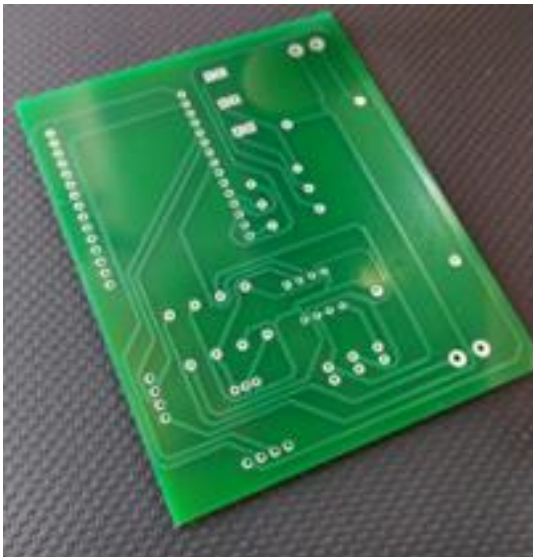


Figura 52. Representación 3D de la Placa de Circuito Impreso PCB.



La distribución de los elementos electrónicos es realizada con base en la facilidad de conexión y desconexión de todos los módulos. Como resultado del diseño se presenta una tarjeta con enrutamiento a doble cara y dimensiones 73 x 90 [mm]. La PCB final se muestra a continuación:

Figura 53. Placa de Circuito Impreso



5.8. DISEÑO DE LA CARCASA DEL DISPOSITIVO

El dispositivo debe contar con una carcasa que permita la facilidad de instalación, protección contra condiciones climáticas adversas y portabilidad, cumpliendo con los principales propósitos de una cubierta: estética, funcionalidad y protección encontrando un balance adecuando entre estos tres, se realiza el diseño de una cubierta o carcasa que debe contener y asegurar toda la circuitería, batería y conexiones internas del dispositivo.

Las consideraciones de diseño generales son:

- La carcasa debe proteger al dispositivo de pequeñas caídas, de resbalamiento de su superficie.
- El aseguramiento debe ser redundante, esto quiere decir que si un sistema falla, es posible percatarse y corregir previo al daño.
- La cubierta es fácil de remover
- El material de la carcasa no se dañará después de remover algunas pocas veces o durante la operación normal.
- La carcasa no le resta funcionalidad a ninguna característica básica del dispositivo (impedir conexión inalámbrica).
- La carcasa no debe incluir piezas móviles fáciles de perder que le resten funcionalidad al equipo.
- Debe evitar el uso de materiales peligrosos en su manufactura.

5.8.1. Dimensiones y Ubicación

Las dimensiones a considerar son las siguientes:

- Batería= 113 mm x 65 mm x 13 mm
- PCB = 73 mm x 90 mm x 30 mm
- GPS = 22 mm x 20 mm x 6 mm
- Conector 4p = 5.75mm x 12.5mm x 7mm
- Conector 2p = 5.75mm x 10 mm x 7mm

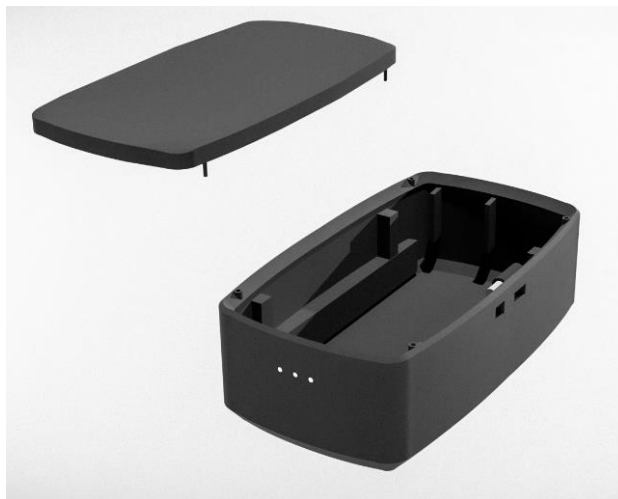
La ubicación de los elementos al interior de la carcasa permite fácil acceso a la batería para cargar, a la PCB para validar señales y al GPS para comunicación con sus satélites sin obstáculos físicos, por ello, se realiza la siguiente disposición de componentes.

Figura 54. Modelo 3D explosionado del sistema integrado [Ver anexo 11].



Con la finalidad de agregar un acabado estético al dispositivo, se realiza un diseño en el software de acceso libre Blender, en el cual se agregan bordes curvilíneos, conectores externos de instalación rápida y agujeros para visualización de pilotos para conocer el estado del dispositivo en tiempo real [Ver anexo 8,9,10 y 11].

Figura 55. Modelo 3D de la carcasa del dispositivo.



Por último, la ubicación del dispositivo en la motocicleta es un punto clave, pues este debe estar fijado a una de las piezas de la motocicleta, no debe ser llevado por el conductor de la motocicleta o el pasajero en su defecto, sin embargo, las superficies en la motocicleta son curvas o simplemente son tubos que no permiten fijación.

Como solución a este problema se plantea ubicar el dispositivo sobre el depósito de combustible de la motocicleta fijado con una ventosa la cual crea vacío al interior de la chupa que fija el dispositivo a la superficie curva del tanque.

Como refuerzo redundante de seguridad se ensamblan cuatro imanes de neodimio que impiden el desplazamiento lateral de la carcasa en el tanque, a la vez que impiden el desprendimiento de esta última en caso de que la ventosa falle.

5.8.2. Materiales

Se desarrolla una tabla ponderada con las alternativas de materiales posibles para la construcción de la cubierta.

Tabla 18. Puntajes ponderados para selección de material.

Nombre	Peso	Facilidad de manufactura	Resistencia	Precio	Puntaje	Ranking
PLA	0.9	0.9	0.7	0.9	85	1
Acero	0.5	0.7	1	0.6	70.5	4
Aluminio	0.7	0.8	0.8	0.7	74.5	3
Madera	0.8	0.9	0.7	1	83.5	2
Peso (0-100)	25	35	25	15		

Dado que el PLA recibe la mayor puntuación, se exporta el archivo hacia el software para impresión 3D CURA para imprimir la cubierta. El tiempo de impresión fue de 3 horas en PLA gris para la base y el botón de encendido y un tiempo de 6 horas con PLA blanco para la tapa, asegurando la calidad de impresión.

Figura 56. Impresión 3D de la carcasa del dispositivo.



5.9. INTEGRACIÓN DE LOS SUBSISTEMAS

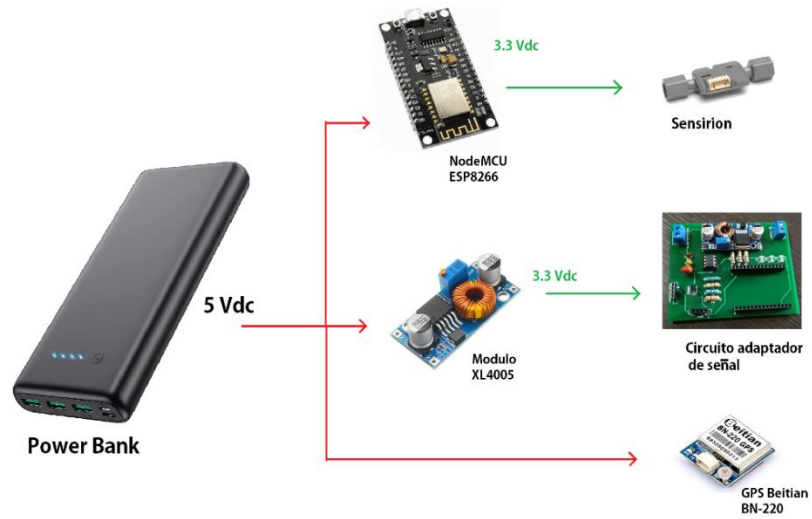
5.9.1. Esquema eléctrico

Todo el sistema eléctrico necesita de energía proveniente de una batería. El dispositivo requiere de dos suministros: el primero de 5 Vdc para conectar por medio de la entrada mini usb a la placa NodeMCU y otro correspondiente a la alimentación del GPS y del circuito adaptador de señal de velocidad angular.

Además, se requiere una alimentación de 3.3 Vdc. Sin embargo, en el mercado no es habitual una batería con dos salidas de diferente valor de tensión, por lo cual se utiliza un módulo regulador de voltaje encargado de ajustar una de las salidas de 5 volts de la batería.

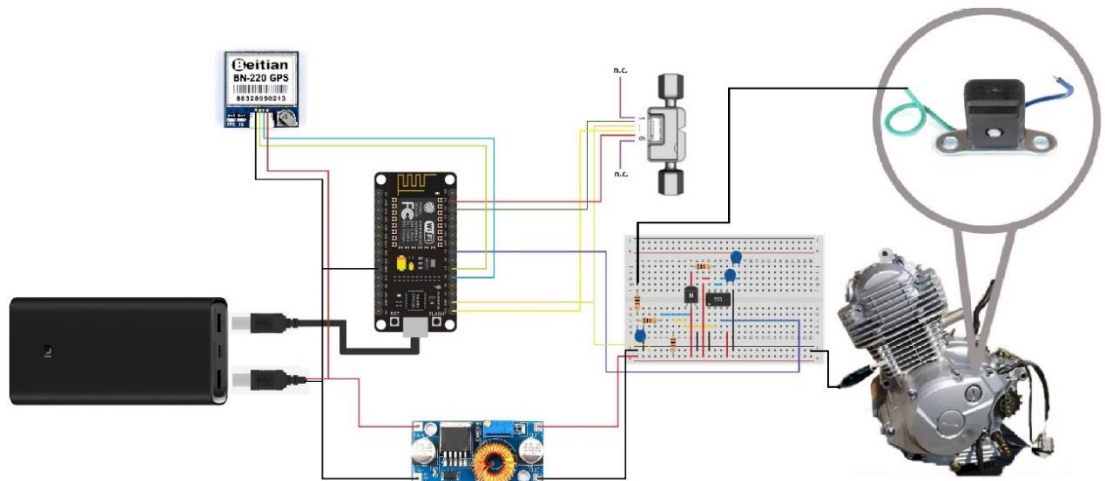
La fuente de alimentación de todo el sistema es una batería portátil de 8000 Mah con dos puertos de salida USB a 5V y 2.1 A. La utilidad y facilidad del dispositivo es su carga rápida. Además, los elementos implicados en la electrónica de todo el sistema no exceden el Amperio de corriente esto permitela batería es una buena elección.

Figura 57. Esquema eléctrico de alimentación de energía del sistema.



La incorporación de todas las conexiones eléctricas y electrónicas se muestran en el siguiente esquema:

Figura 58. Esquema de conexiones eléctricas del sistema.



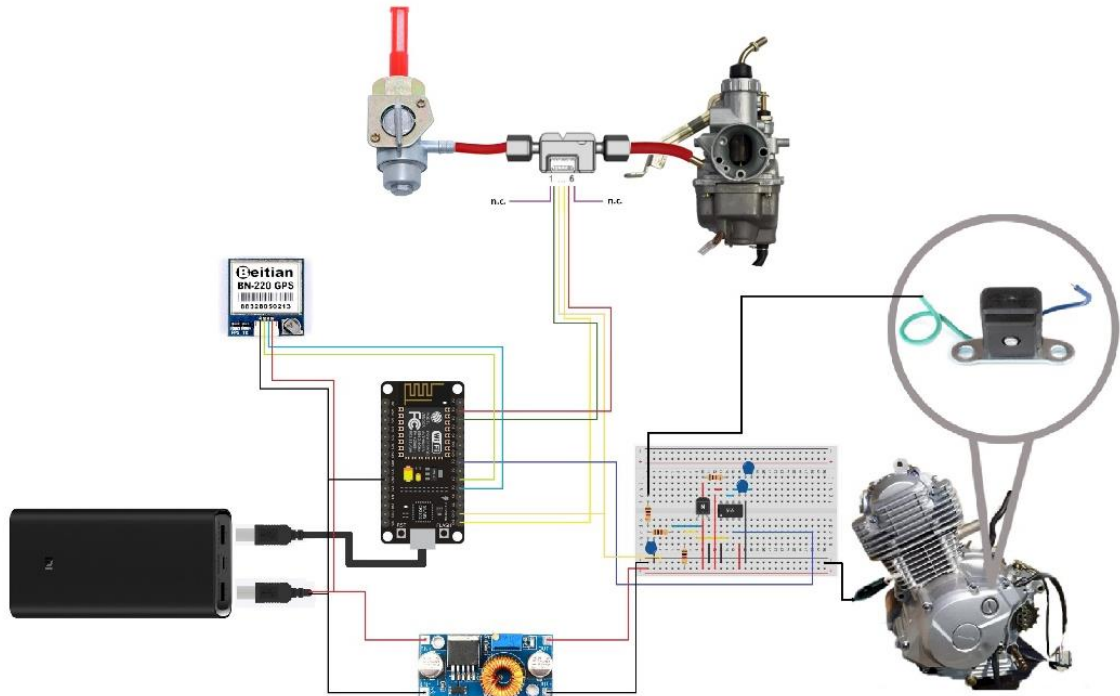
El resultado de la integración de todos los subsistemas que componen el proyecto es un dispositivo compacto con dimensiones 175 x 95 x 70.5 [mm]

Figura 59. Sistema Integrado.



La siguiente figura muestra de manera general todas las conexiones hidráulicas y eléctricas:

Figura 60. Esquema general de conexiones del sistema de monitoreo.



6. RESULTADOS Y EVIDENCIAS

Según la metodología planteada, la validación del funcionamiento del dispositivo se realiza verificando el correcto funcionamiento de cada uno de los subsistemas, y posteriormente se verifica el funcionamiento de todo el sistema integrado utilizando conceptos de error relativo, desviación estándar y validación de la robustez del sistema a partir de los datos recolectados bajo diferentes circunstancias aleatorias como condiciones climáticas, de tráfico, aspectos del terreno, entre otros.

La validación de componentes, subsistemas y funcionalidad del dispositivo se realiza en condiciones estáticas utilizando el error relativo como medida de la precisión de los instrumentos para determinar la magnitud deseada, bajo condiciones dinámicas se realiza un tratamiento de datos para obtener información relevante a partir de los datos en bruto recopilados por el sistema de monitoreo.

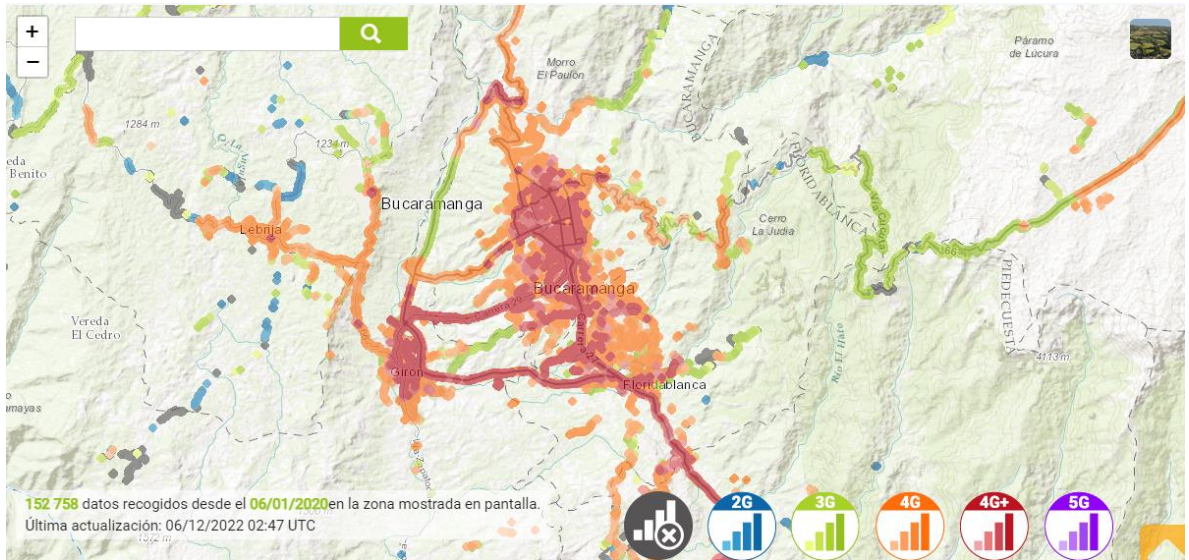
6.1. CONECTIVIDAD

La validación de conectividad del dispositivo hace referencia al tiempo de subida de información hacia la base de datos y la aplicación móvil, así como a la calidad y estabilidad de la conexión.

En Colombia la cobertura de red abarca la totalidad de áreas urbanas llegando a ser la red 4G LTE+ última tecnología implementada hasta la fecha, aunque ya se realizan pilotos para implementación de 5G.

La red 4G LTE+ ofrece una velocidad de subida máxima de 40 Mbps [39], aunque las zonas urbanas tienen cobertura parcial de esta red y entre diferentes áreas de la ciudad existe cobertura 3G, 4G y 4G LTE y esta depende del operador de telefonía móvil contratado.

Figura 61. Mapa de cobertura de red móvil de Claro Colombia en Bucaramanga.



Tomada de Nperf [38].

Un sistema IoT requiere de velocidades de subida sobre los 100 Mbps para cumplir su objetivo de monitoreo en tiempo real, sin embargo, estas velocidades se alcanzan normalmente con redes cableadas o redes de última generación 5G con velocidades de subida hasta cuatro veces mayores [40].

Con la infraestructura de red disponible actualmente en Colombia el Sistema de Monitoreo desarrollado alcanza un tiempo de subida de datos de 2 segundos con tiempos muertos de 10 segundos en promedio a una tasa de 0.15 desconexiones por minuto lo cual garantiza la estabilidad del dispositivo bajo diferentes condiciones climáticas y de tráfico.

Figura 62. Tiempo de subida a la base de datos.

Date	Time
6/1/2022	10:00:15
6/1/2022	10:00:18
6/1/2022	10:00:20
6/1/2022	10:00:22
6/1/2022	10:00:24
6/1/2022	10:00:27
6/1/2022	10:00:29
6/1/2022	10:00:31
6/1/2022	10:00:33
6/1/2022	10:00:36

6.2. PRUEBAS ESTÁTICAS

En esta sección se muestran las evidencias de las pruebas estacionarias realizadas para validación de componentes y de subsistemas. El desarrollo de las pruebas ejecutadas en la motocicleta se divide en las siguientes categorías:

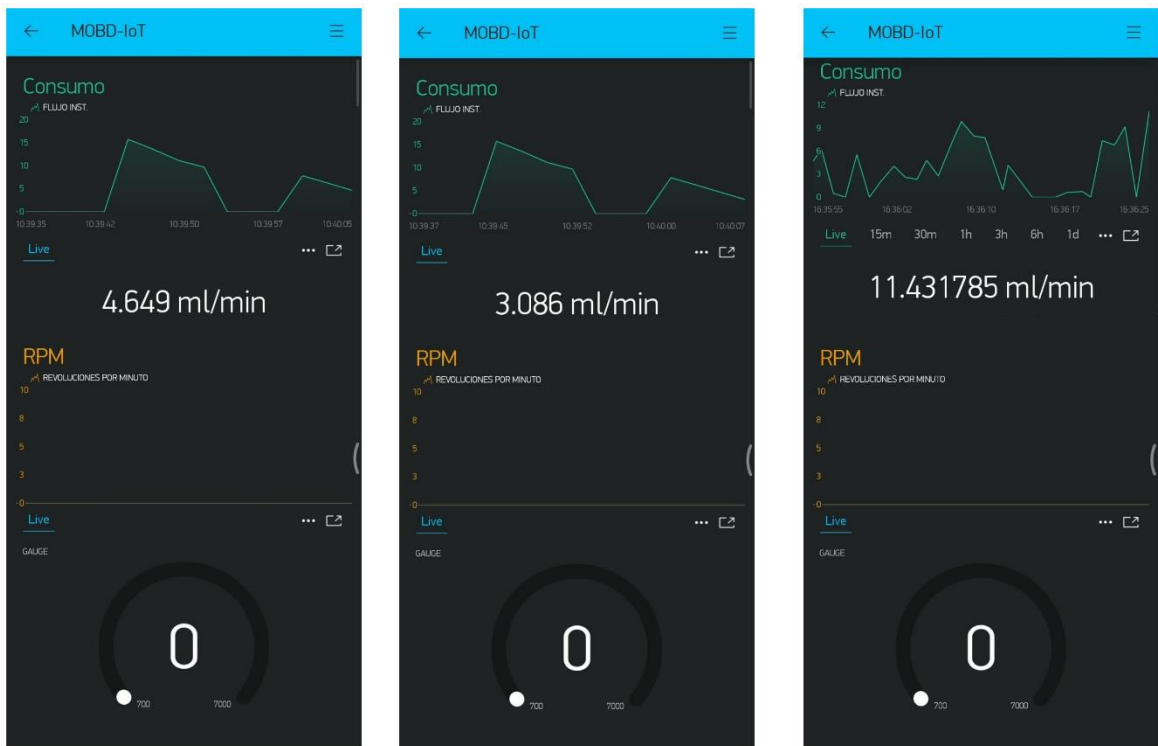
6.2.1. Consumo de Combustible

La validación de consumo de combustible requiere de un dinamómetro de chasis, equipo con el cual no se cuenta durante la ejecución del presente proyecto.

Se realiza comprobación de las mediciones utilizando el sistema de calibración del inciso 5.2.1. con los mismos resultados de precisión de medida.

Se realiza validación del envío de datos a la aplicación de visualización con resultados óptimos de seguimiento según el régimen del motor, pues la relación entre el requisito de velocidad representado en la apertura del manillar de aceleración y el consumo instantáneo de combustible se cumple.

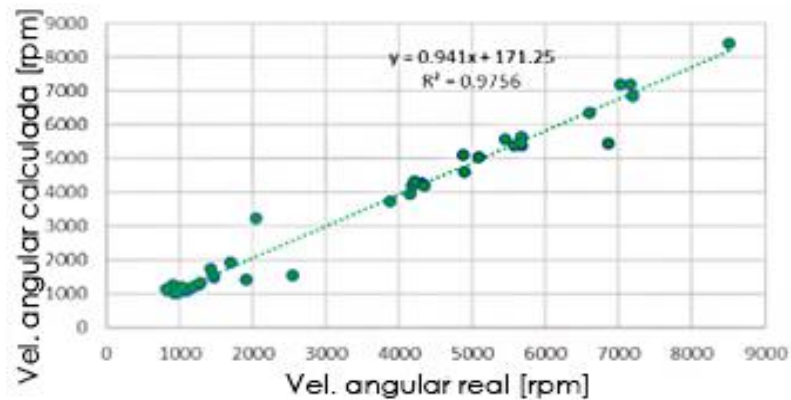
Figura 63. Consumo instantáneo de combustible visualizado en la aplicación móvil.



6.2.2. Velocidad Angular del Motor

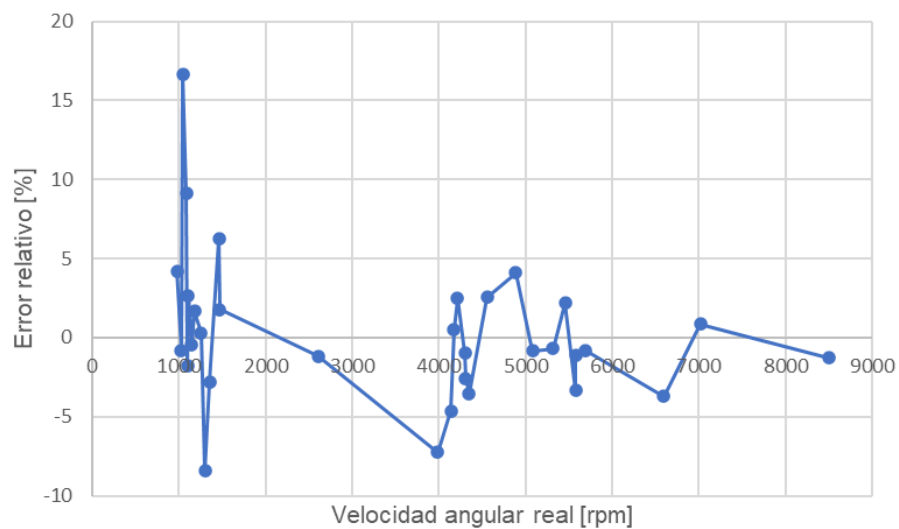
Se realizan validaciones del circuito adaptador de señal comparando con los valores entregados de un tacómetro comercial. Posteriormente, se desarrolla un análisis estadístico-comparativo de los datos obtenidos con el instrumento junto con los datos recopilados por el circuito adaptador de señal. Los resultados son los siguientes:

Figura 64. Velocidad angular calculada vs. Velocidad angular real en rpm.



El error relativo de las medidas recopiladas se encuentra en el rango de -10 % y 15 %, con un promedio es de 0.32%.

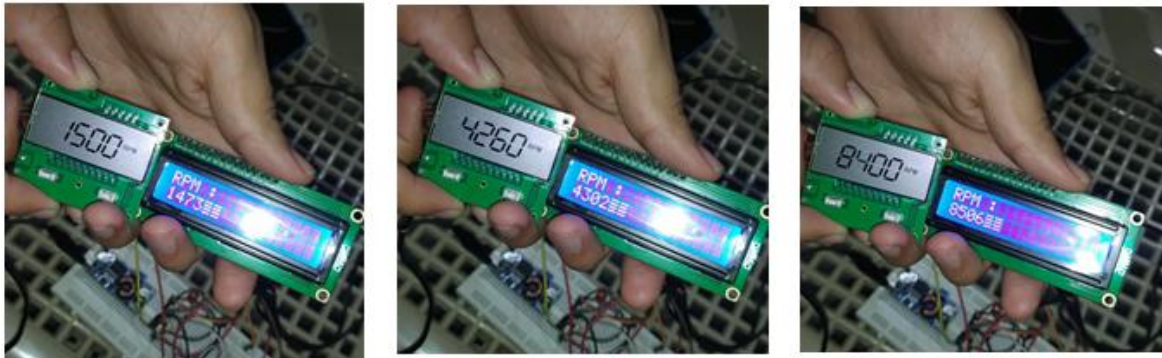
Figura 65. Error relativo vs. Velocidad angular real.



El tacómetro comercial Kebidumei tiene la capacidad de indicar la velocidad angular del motor de la motocicleta en revoluciones por minuto con una resolución de 10 rpm.

La comparación de las mediciones recopiladas se observa en las siguientes imágenes:

Figura 66. Validación de velocidad angular del motor en rpm.



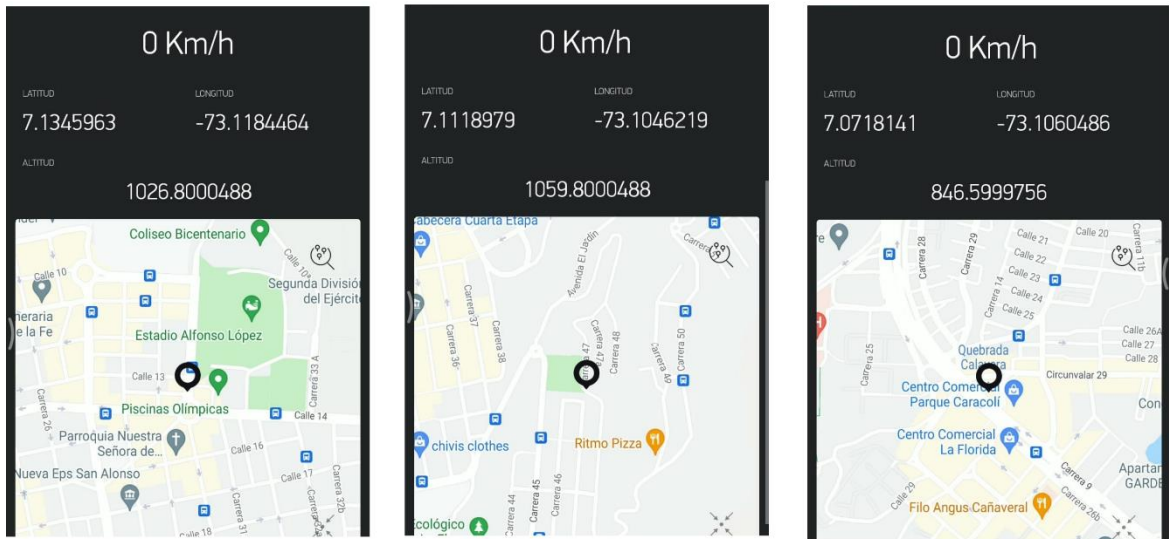
6.2.3. Posición y Velocidad

El acceso a los 72 canales de búsqueda del módulo Beitian BN-220 permiten la rápida conexión del receptor GPS con los satélites aún en áreas urbanas con un alto número de edificaciones, llegando a alcanzar la conexión y sincronización con hasta 12-15 satélites.

La prueba de posición se realiza en diferentes puntos de la ciudad de modo que se validan las coordenadas entregadas y visualizadas en la aplicación con la posición física real en el momento del muestreo.

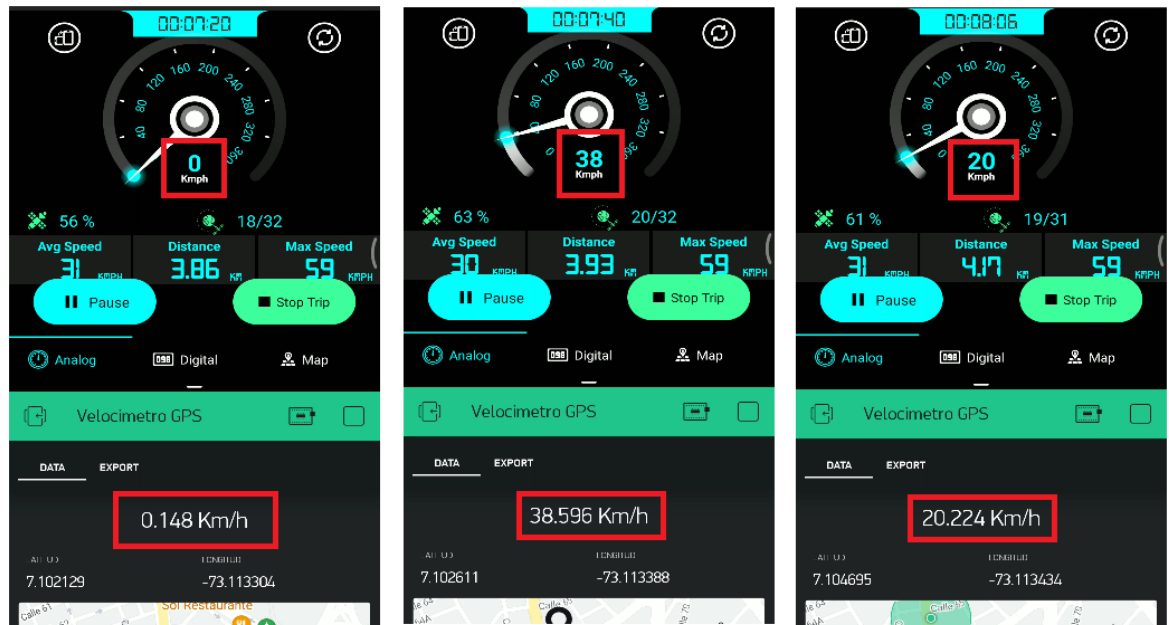
Los puntos de ubicación son: el Estadio Alfonso López, CSU UNAB y el Centro Comercial Caracolí en Cañaveral que se evidencian en las siguientes imágenes:

Figura 67. Validación de posición real con el sistema integrado.



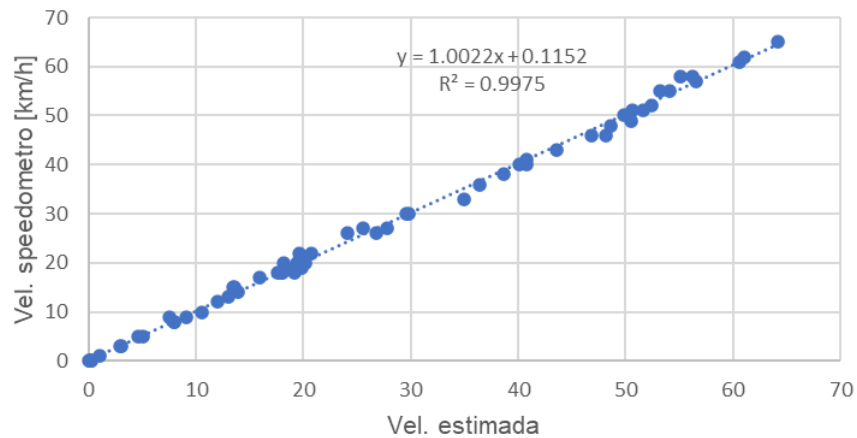
Ahora bien, las validaciones de la medición de velocidad con el GPS se realizan comparando con los valores recopilados mediante la app móvil “GPS Speedometer” en un smartphone. El funcionamiento de dicha app es bajo los mismos principios de un sistema de localización, pero con el sistema GPS integrado del smartphone.

Figura 68. Validación de velocidad con la herramienta GPS Speedometer.



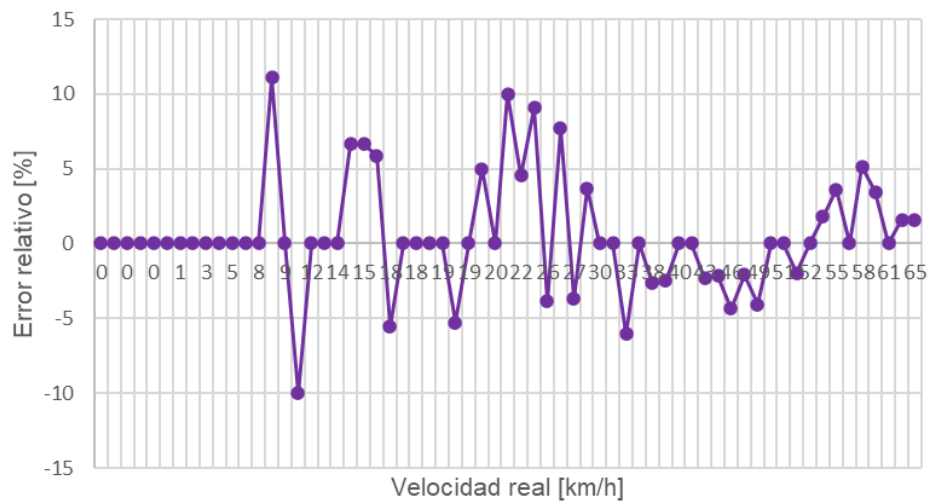
Se lleva a cabo un análisis estadístico-comparativo de los datos recopilados con el GPS Beitian BN-220 junto con los datos recopilados con la app “GPS Speedometer”. Es necesario aclarar que, en esta medición, aunque se hace independientemente de las otras mediciones, es evidente que se realiza en pruebas dinámicas, es decir, la motocicleta en movimiento.

Figura 69. Velocidad real vs Velocidad estimada.



Del análisis se obtienen errores relativos que en el rango de -10% y 10 %, con una media de error relativo de 0.49%.

Figura 70. Error relativo de velocidad vs Velocidad real.



6.2.4. Sistema Integrado

Los subsistemas de las variables medidas en la motocicleta se integran para ser visualizados en la aplicación diseñada en un solo programa ejecutado en la NodeMCU ESP8266. Se obtienen respuestas con un tiempo de actualización de 1 segundo por parte del sensor y de la señal procesada proveniente de la bobina pulsante. La ubicación de la motocicleta coincide con las coordenadas de localización y es dependiente de la velocidad de recepción de las señales de satélite, lo que quiere decir que puede tardar más tiempo su actualización, pero sin presencia de valores atípicos en los resultados.

Figura 71. Validación del sistema integrado en pruebas estacionarias.



Figura 72. Validación de datos recolectados en la aplicación durante pruebas estáticas.



Una vez se habilita la base de datos para la recopilación de las variables monitoreadas, estos se almacenan correctamente en la base de datos creada con un tiempo de subida de 2 segundos.

Figura 73. Resultados de subida de datos a la base de datos en Google Sheets.

	A	B	C	D	E	F	G
1	Date	Time	Ang. Vel. [rpm]	Latitude [°]	Longitude [°]	Altitude [m]	Inst. Flow [ml/min]
2							
3							
4	5/31/2022	09:37:06	2857.142822	7.10216856	-73.12139893	979.80	0.00958
5	5/31/2022	09:37:08	3333.333496	7.10216856	-73.12139893	979.80	9.06823
6	5/31/2022	09:37:10	2142.856934	7.10216856	-73.12139893	979.80	5.85003
7	5/31/2022	09:37:13	2068.965576	7.10216856	-73.12139893	979.80	19.42945
8	5/31/2022	09:37:15	1200	7.10215950	-73.12139130	978.50	0.00000
9	5/31/2022	09:37:18	3000	7.10215950	-73.12139130	978.50	0.00000
10	5/31/2022	09:37:21	1250	7.10215950	-73.12139130	978.50	3.01219
11	5/31/2022	09:37:23	2307.692139	7.10215950	-73.12139130	978.50	0.00000
12	5/31/2022	09:37:26	2068.965576	7.10215950	-73.12139130	978.50	15.06169
13	5/31/2022	09:37:28	1935.484009	7.10215950	-73.12139130	978.50	0.00000
14	5/31/2022	09:37:31	1935.484009	7.10215950	-73.12139130	978.50	3.79878
15	5/31/2022	09:37:33	1935.484009	7.10215950	-73.12139130	978.50	2.53729

6.3. PRUEBAS DINÁMICAS

Como lo indica la metodología del proyecto se realiza la toma de datos y análisis del desempeño del sistema bajo diferentes condiciones para verificar características como autonomía del sistema, calidad de los datos en rutas con diferentes condiciones de terreno, climáticas, de tráfico y de operación para evaluar el cumplimiento de la totalidad de los objetivos planteados.

6.3.1. Sistema Integrado

Tabla 19. Tabla de componentes.


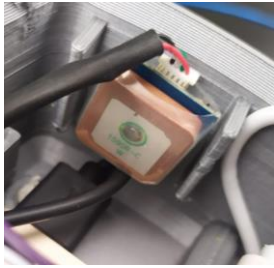

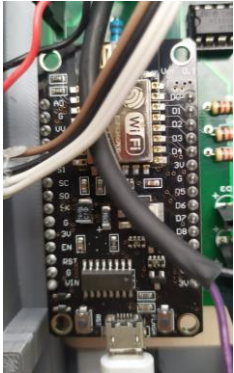

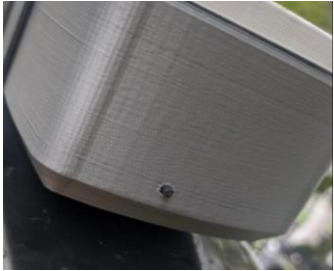
Componente	Imagen	Componente	Imagen
Cable 4p		GPS Beitian BN-220	
Cable 2 P		NodeMCU	
Conector 2p		Botón Encendido	

Tabla 19. Continuación.



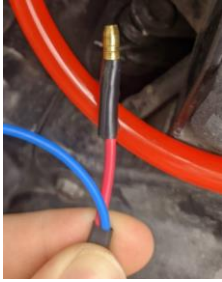


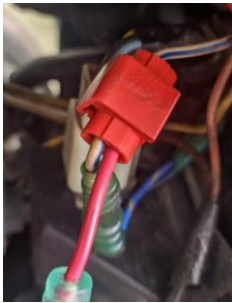










Componente	Imagen	Componente	Imagen
Terminal Faston Hembra		Mangueras Tanque-Flujómetro-Carburador	
Terminal Faston Macho		Válvula de paso	
Terminal Tipo Pinza		Conector de autodesforre	
Conector 2p Hembra		Entrada carburador	

Tabla 19. Continuación.

Componente	Imagen	Componente	Imagen
Racores		Ventosa e imanes	
Flujómetro y montura de fijación		Correas De Sujeción	
Modulo Regulador XL4005		Tapa	
PCB		Caja	

6.4. PROCEDIMIENTO DE INSTALACIÓN, ENCENDIDO Y APAGADO

El dispositivo es parcialmente portable, pues para su instalación se requieren dos actividades previas referentes a la ubicación del sensor de flujo Sensirion SLF3S-1300S [Véase sección 5.2.4] y a la instrumentación para realizar la medición de la velocidad angular del motor de la motocicleta [véase sección 5.3.2.4].

Los resultados de la instalación del dispositivo en la motocicleta y la puesta en servicio del sistema son:

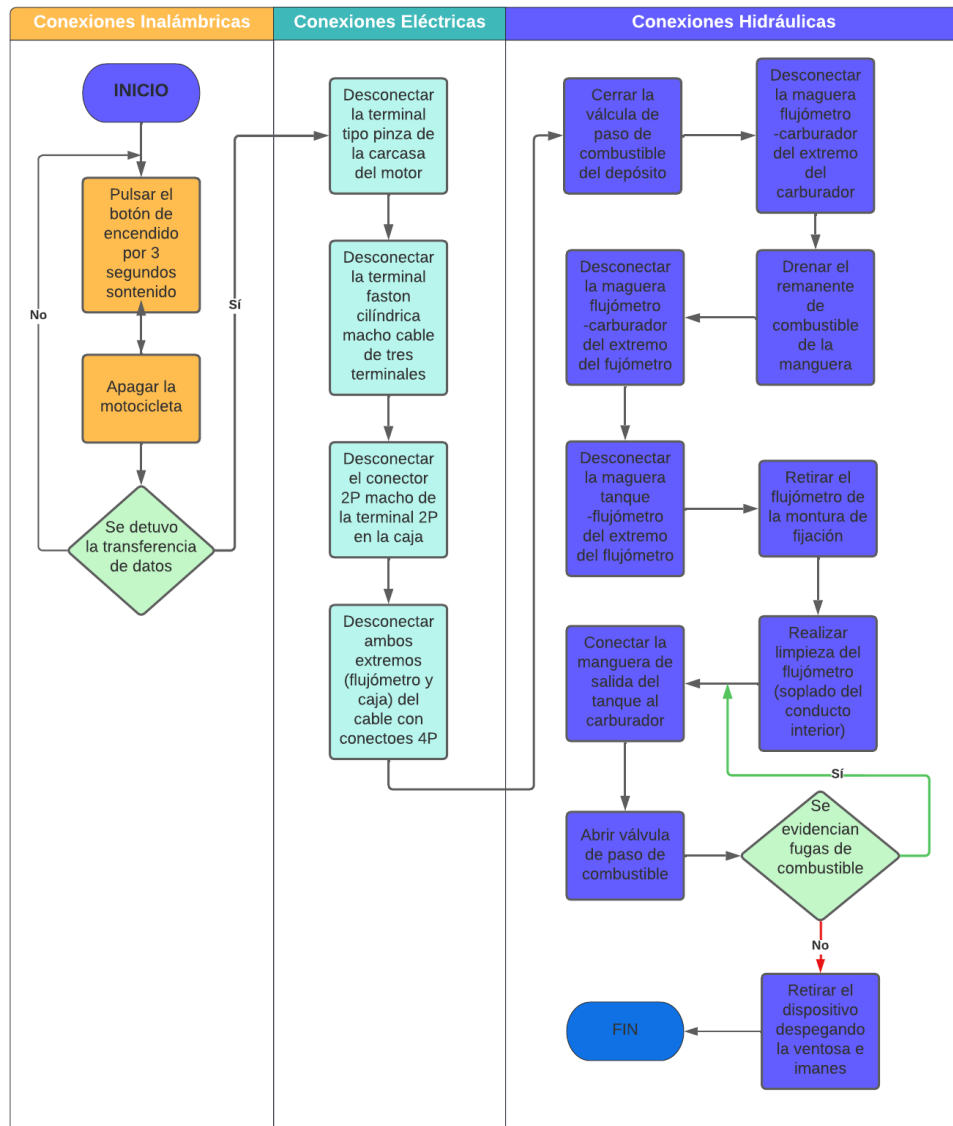
Figura 74. Ubicación del sistema de monitoreo en la motocicleta



6.4.2. Procedimiento de desconexión del sistema

Los pasos para realizar la desconexión del sistema después de finalizada la toma de datos durante una ruta siguen el orden inverso a la puesta en servicio, es decir, primero se desconecta el dispositivo de la red, luego se desconectan las conexiones eléctricas y por último se desconectan las conexiones hidráulicas.

Figura 76. Procedimiento de desconexión del sistema de monitoreo [Ver anexo 13].



6.5. RESULTADOS DE LAS PRUEBAS DINÁMICAS

Las pruebas dinámicas consisten en la instalación, puesta en marcha y monitoreo de las variables de interés capturadas por el sistema durante rutas cotidianas en el área metropolitana de Bucaramanga, Colombia.

Las rutas monitoreadas recorren la ciudad desde puntos estratégicos en zonas comerciales y residenciales con diferentes condiciones de tráfico y climáticas como autopistas, zonas urbanas con alto número de edificaciones, avenidas principales y vías secundarias.

Se recopilaron 25 horas de pruebas en las siguientes rutas:

- Ruta 1: Real de minas – Cañaveral
- Ruta 2: Cañaveral – Real de minas
- Ruta 3: Estoraques - Real de minas
- Ruta 4: Real de minas – Estoraques
- Ruta 5: Real de minas – CSU UNAB.
- Ruta 6: CSU UNAB – Real de minas
- Ruta 7: UNAB – UIS
- Ruta 8: UIS – Real de minas
- Ruta 9: UIS – C.C Cacique
- Ruta 10: C.C Cacique – El Carmen
- Ruta 11: El Carmen – Real de minas
- Ruta 12: Provenza – UNAB
- Ruta 13: UNAB – Cañaveral
- Ruta 14: Real de minas – Estadio Alfonso Lopez
- Ruta 15: Estadio Alfonso Lopez – Real de minas

Los datos en bruto recopilados de estas rutas se pueden visualizar en la base de datos en Google Sheets siguiendo el siguiente enlace.

https://docs.google.com/spreadsheets/d/1I3HbMTIju_9oTkg1Sj-ZSt8WFibEWNaG6gwAfPn0CO0/edit?usp=sharing

Las rutas se pueden visualizar en el mapa utilizando GPS Tracker o Google Maps exportando los datos de las columnas de latitud y altitud de la base de datos, estas se pueden visualizar siguiendo el siguiente enlace:

<https://www.google.com/maps/d/u/2/edit?mid=15rfV-TLy89JzVEIQXfFPjXqsFFNSC-g&usp=sharing>

Se toman dos muestras de dos rutas específicas, estas son las rutas entre las zonas Cañaveral –Real de Minas y Real de Minas – Estoraques para desarrollar los análisis de los datos recopilados.

6.5.1. Ruta Cañaveral – Ciudadela Real de Minas

La Ciudadela Real de Minas es una zona residencial de la ciudad de Bucaramanga con un alto volumen vehicular lo cual sumado a sus calles estrechas resulta en una alta congestión vehicular, por su parte, Cañaveral es una de las zonas comerciales más destacadas del área metropolitana de Bucaramanga, la ruta entre estas dos áreas de la ciudad comprende la Calle 61, viaducto Benjamín García Cadena y la autopista Bucaramanga-Floridablanca.

La calle 61 es una de las avenidas principales del suroccidente de Bucaramanga, la cual comunica la Ciudadela Real de Minas con la autopista Bucaramanga-Floridablanca, es una calle con pendiente nula en el 80% del recorrido y ligeros aumentos de pendiente en cortos intervalos de 30 metros, por su parte el viaducto García Cadena es un puente elevado de diez carriles de 650 metros de longitud total con pendiente ascendente en el sentido de ruta.

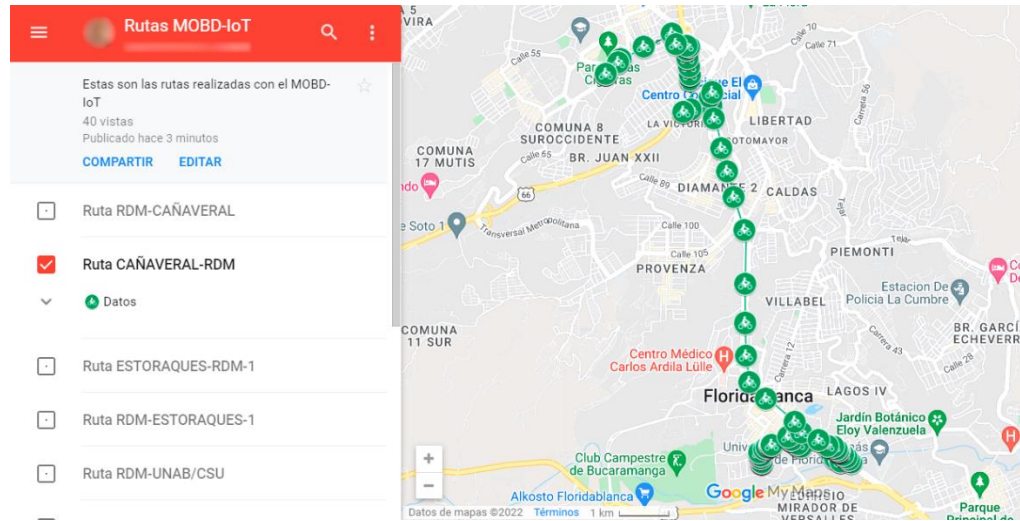
La ruta se monitoreó sobre las 10:00 a.m. por lo cual el volumen vehicular es moderado y los semáforos presentes se ubican en la Ciudadela Real de Minas y en el tramo de la autopista se recorrió el carril rápido. El tiempo total de la ruta fue de aproximadamente 17 minutos.

Figura 77. Aplicación de visualización durante pruebas dinámicas.



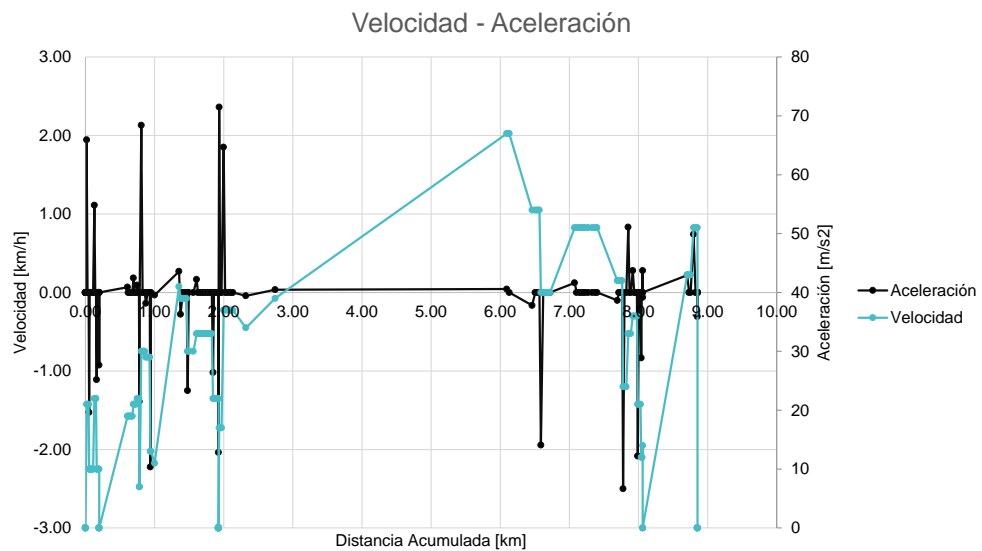
La trayectoria realizada en la motocicleta es la siguiente:

Figura 78. Ruta monitoreada Cañaverál – Ciudadela Real de Minas en Google Maps.



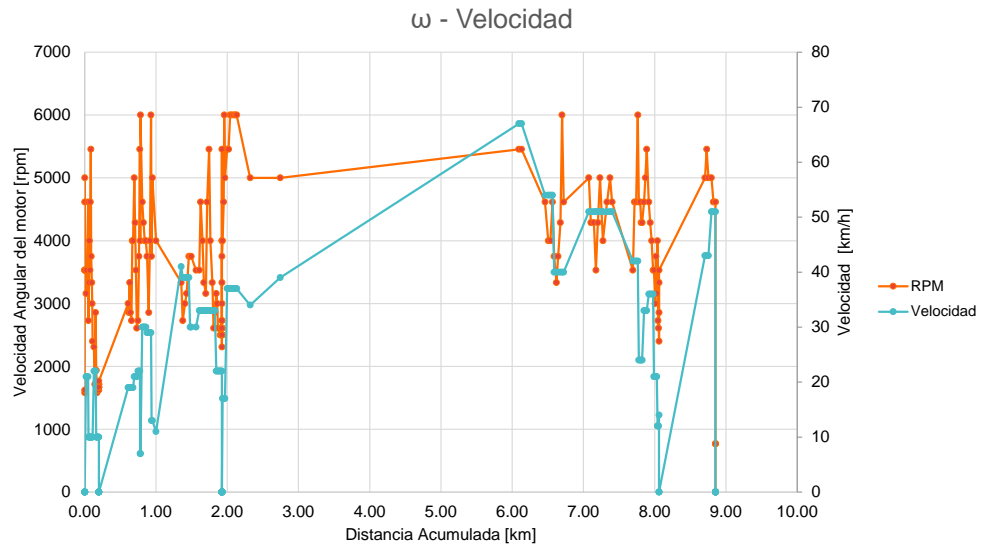
Con el conjunto de datos registrados se realizan validaciones para verificar la coherencia y validez de la base de datos. Además, con los resultados de la información recopilada se llevan a cabo diferentes análisis gráficos. De modo que, de la figura [79] se puede afirmar que en cada pico de aceleración o desaceleración coincide con el aumento y disminución de velocidad.

Figura 79. Curvas de velocidad y aceleración en ruta Cañaverál-Real de Minas.



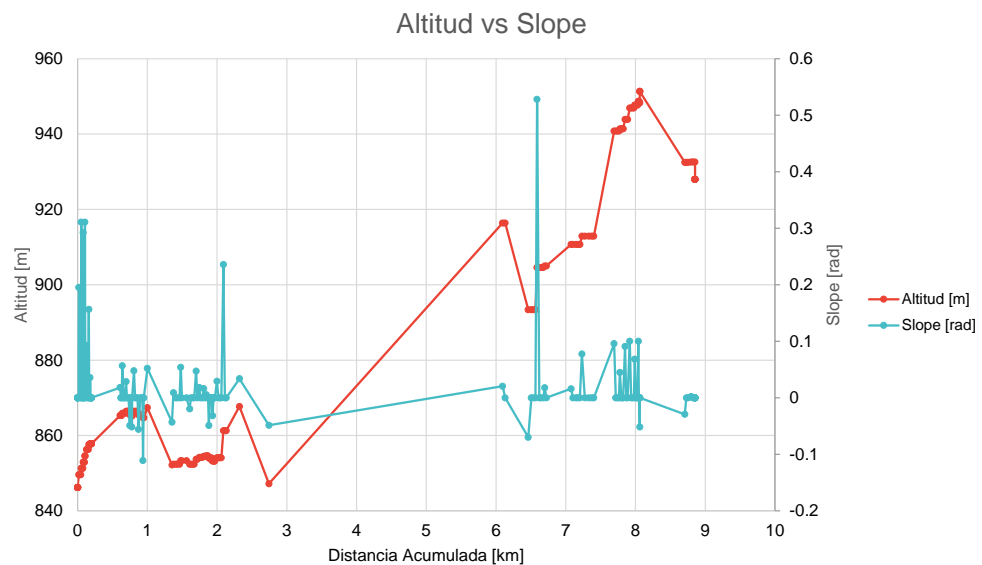
La relación entre la velocidad de la motocicleta y la velocidad angular de la motocicleta es lineal, sin embargo, debido al cambio de marcha los coeficientes no son los mismos, esto se observa en la gráfica con el seguimiento de ambas curvas y las variaciones de velocidad angular durante los cambios de marcha.

Figura 80. Curvas de velocidad angular y velocidad en ruta Cañaverál-Real de minas.



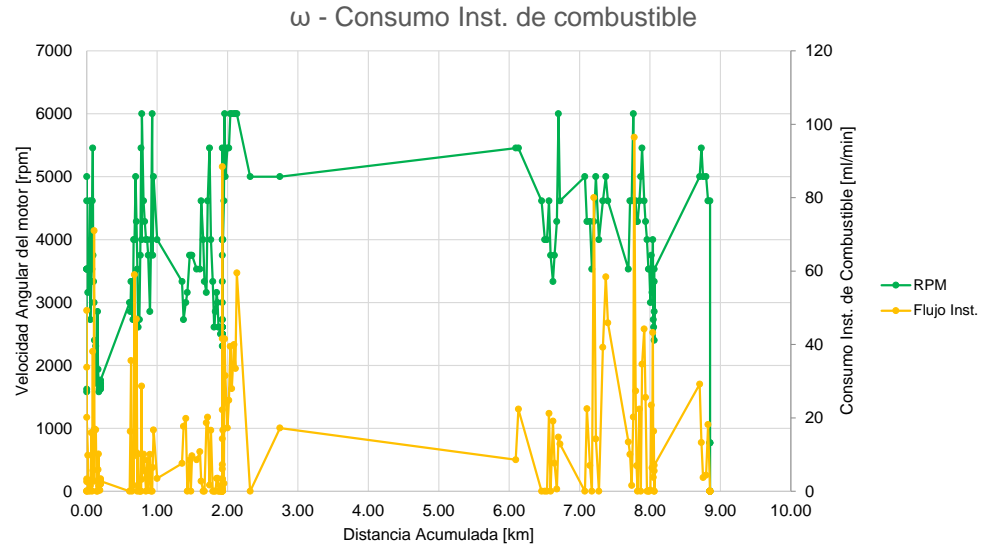
Los puntos de pendiente tienen coherencia cuando existe un cambio de altitud, como se demuestra en la siguiente grafica.

Figura 81. Curvas de altitud y pendiente en ruta Cañaverál-Real de minas.



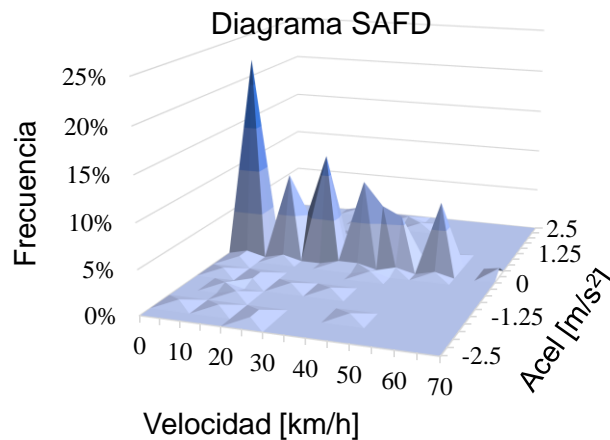
La relación entre la velocidad la velocidad angular de la motocicleta y el consumo no es lineal, ya que depende de varios factores. Sin embargo, debido al cambio de marcha los coeficientes no son los mismos. Similar al grafico anterior, en la figura [82] se observa el seguimiento de ambas curvas y las variaciones de flujo instantáneo durante los cambios de marcha.

Figura 82. Curvas de velocidad angular y Consumo de combustible en ruta Cañaverál-Real de minas.



Por último, un diagrama SAFD (Speed Acceleration Frequency Diagram) del cual se puede analizar aspectos mucho más avanzados en materia de ciclos de conducción o llevar a cabo otros estudios como el modelo de Markov para optimizar procesos en la moto, entre otras cosas más.

Figura 83. SAFD de la ruta Cañaverál-Real de minas..



6.5.2. Ruta Ciudadela Real de Minas - Estoraques

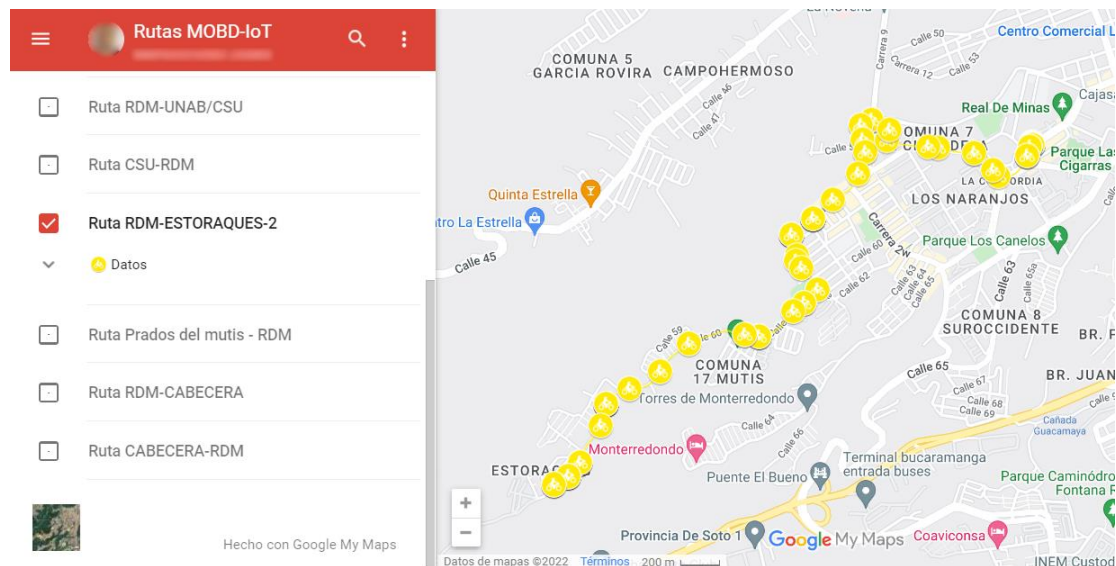
Estoraques es uno de los sectores populares de la ciudad de Bucaramanga, ubicado al suroccidente de la ciudad.

La ruta recorrida comprende las avenidas principales de la ciudadela Real de Minas las cuales son calles de entre uno y dos carriles con alto volumen vehicular debido a la estrechez de la carretera. La pendiente de la carretera tiene tendencia nula, aunque con ligeros ascensos y descensos de entre 30 y 60 metros.

Los datos se recopilaban sobre las 7:00 p.m. finalizando la hora pico de congestión vehicular, por lo cual no existen paradas considerables, adicionalmente la ruta cubierta no posee alto número de semáforos. El tiempo total para esta ruta fue de aproximadamente 10 minutos.

La trayectoria realizada en la motocicleta es la siguiente:

Figura 84. Ruta monitoreada Real de Minas – Estoraques en Google Maps.

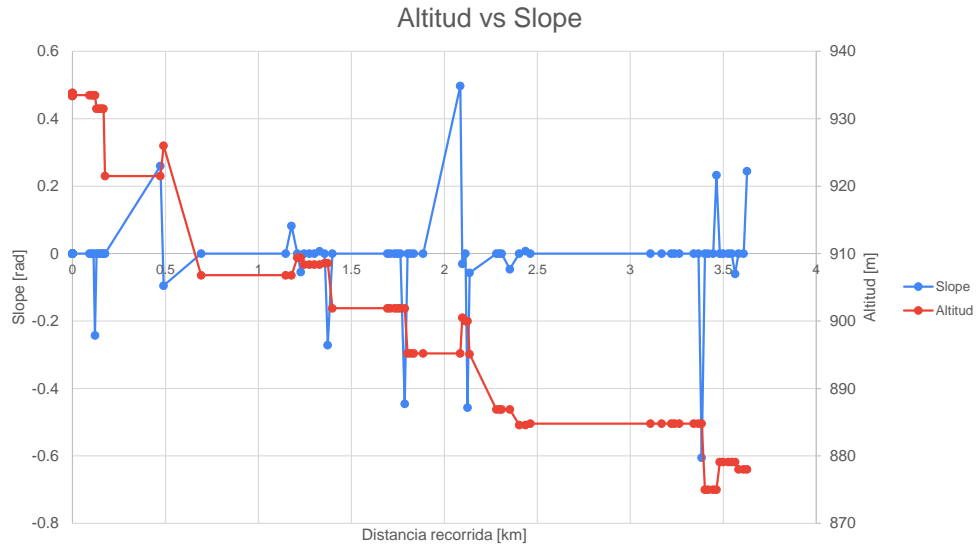


Así como en la ruta Cañaveral – Real de minas, se realiza un análisis con los datos registrados para verificar la misma coherencia, eficiencia y validez de la base de datos.

Se puede afirmar nuevamente en esta ruta que en cada pico de aceleración o desaceleración coincide con el aumento y disminución de velocidad.

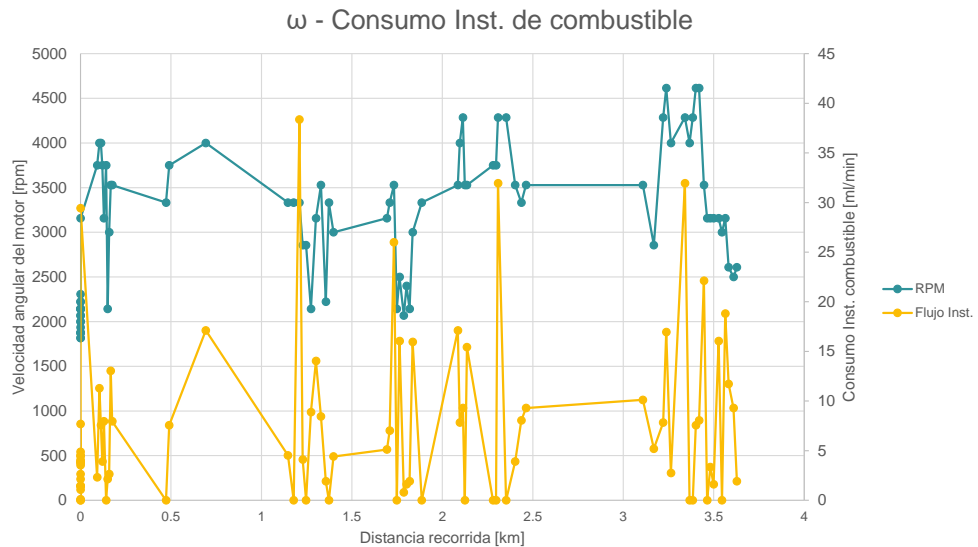
La pendiente cambia cuando existe un cambio de altitud, como se muestra en la siguiente grafica

Figura 87. Curvas de altitud y pendiente en ruta Real de minas - Estoraques.



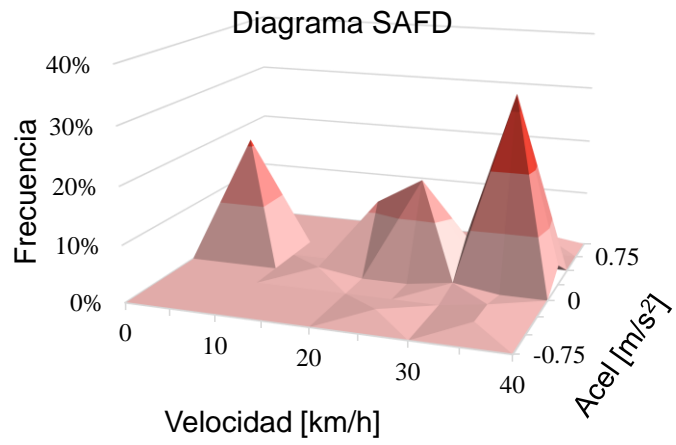
Se observa el seguimiento de ambas curvas y las variaciones de flujo instantáneo durante los cambios de marcha, los datos son consistentes durante las diferentes rutas monitoreadas.

Figura 88. Curvas de velocidad angular y consumo de combustible en ruta Real de minas - Estoraques.



Por último, el diagrama SAFD correspondiente a la ruta Real de minas – Estoraques.

Figura 89. Curvas de velocidad y aceleración en ruta Real de minas - Estoraques.



7. CONCLUSIONES

El sistema instrumental de monitoreo desarrollado se caracteriza por ser un dispositivo semi portable poco invasivo con la motocicleta, pues para su instalación se requiere la ubicación de dos elementos de instrumentación, el flujómetro y el sensor de velocidad angular del motor. El flujómetro al ser un dispositivo montado en línea genera una caída de presión en el fluido debido a la reducción de la sección transversal en su interior, sin embargo, los efectos no son considerables y no afectan el correcto funcionamiento de la motocicleta pues no se observa variación en el tiempo de respuesta ante el requisito de potencia inducido por el manillar de aceleración. Por su parte el sensor de velocidad angular si se instala correctamente no interviene en el proceso de sincronización del sistema eléctrico el cual es el principal riesgo, pues si el cable de la bobina pulsante se abre la bujía no genera la chispa en el tiempo de potencia del motor.

Estas características de portabilidad, sumado a la manufactura por impresión 3D utilizada, facilita la replicabilidad del sistema de monitoreo y la validación realizada en condiciones reales en el área metropolitana de Bucaramanga, permiten ubicar al dispositivo en el nivel de madurez de la tecnología *TRL 7* referente a la demostración de la funcionalidad en un entorno operacional real, aportándole el título de dispositivo desarrollado a escala semi industrial.

La autonomía del sistema instrumental es de 20 horas certificadas o 40 recorridos monitoreados debido a la capacidad de la batería utilizada de 8000 mAh y al circuito desarrollado el cual consume una potencia de 0.75 W validado en laboratorio, las rutas recorridas durante la etapa de validación del sistema se realizaron sin cargas intermedias de la batería llegando a descargarse sobre la cuadragésima ruta monitoreada.

El tiempo de actualización de los datos recopilados durante un recorrido es de 2 segundos en promedio con una tasa de 0.15 desconexiones por minuto, estas dos propiedades dependen de las características del terreno, las condiciones climáticas y la facilidad de acceso a los satélites de la constelación GPS. Sin embargo, el mayor detractor de la disminución del tiempo de respuesta del sistema es la cobertura de red móvil presente en la zona ya que en Colombia a la fecha la infraestructura de red permite como máximo la conexión 4G LTE, la cual permite la subida de datos a la nube en donde la unidad de procesamiento realiza la solicitud *https* a los servidores de Google para escribir los datos recolectados, esto se observa con mayor claridad cuando se deshabilita la subida de datos a Google Sheets obteniendo un tiempo de actualización de 1 segundo en promedio.

La tasa de actualización de los datos de posición y velocidad provenientes del GPS dependen de las características de la zona en la cual se realiza el monitoreo, en el momento de una eventualidad con el sistema de posicionamiento este no se refleja como un valor atípico, pues la unidad de procesamiento envía el último valor guardado en memoria, sumado a la tecnología del módulo GPS utilizado el cual se reinicia y en un tiempo de 5 a 10 segundos y retorna a su estado operacional contribuyen a la calidad de los datos recolectados.

Los datos en bruto recopilados durante la etapa de validación del sistema mediante pruebas dinámicas cumplen con el objetivo de ser la materia prima para la obtención de información útil referente al monitoreo de los ciclos de conducción utilizando herramientas como el diagrama SAFD y las curvas características de las variables de interés para realizar los análisis correspondientes de rendimiento del vehículo y el impacto de sus emisiones contaminantes en el medio ambiente.

La versatilidad que aporta la implementación de tecnologías de la cuarta revolución industrial como el internet de las cosas en el presente proyecto, brindan al usuario la capacidad de capturar los datos para el desarrollo de los ciclos de conducción, al mismo tiempo que uno o más usuarios pueden visualizar el desarrollo de las pruebas en tiempo real a través de la aplicación móvil desarrollada *MOBD-IoT* y la base de datos almacenada en la nube de Google Sheets.

Los requisitos de instalación del dispositivo permiten abarcar un amplio sector del parque automotor en Colombia, a pesar de que los requisitos de instalación permiten abarcar únicamente motocicletas carburadas de menos de 250 cc de cilindrada con bobina pulsante como elemento sincronizador del sistema eléctrico en lugar de sensores CKP y CMP, adicionalmente se restringe la motocicleta al uso de gasolina convencional como combustible.

8. RECOMENDACIONES Y TRABAJOS FUTUROS

El desarrollo de los ciclos de conducción sigue un estándar predeterminado, por lo tanto, el dispositivo está diseñado para ser utilizado por un conductor que viaja sin acompañante, aunque se recomienda en lo posible tener un acompañante que indique el estado en tiempo real de la prueba.

Los requisitos de instalación del sistema instrumental de monitoreo para obtener resultados significativos son:

- Motocicletas carburadas de cilindradas menores a 250 cc.
- Bobina pulsante monopulso por ciclo como elemento sincronizador del sistema eléctrico.
- Utilización de gasolina convencional como combustible.
- Smartphone con datos móviles contratados por un proveedor de servicios de red con infraestructura de red mínimo 4G.

Se recomienda modificar el nombre del punto de red hotspot del smathphone con el nombre MOBD IoT para mantener una conexión estable en caso de alguna eventualidad que induzca desconexiones del sistema a la red.

Se recomienda la implementación de un sistema de almacenamiento físico para obtener un tiempo de actualización de datos mayor con posterior subida a la nube si el requerimiento del usuario es velocidades de subida aún mayores.

Se recomienda realizar una validación de laboratorio del consumo instantáneo de combustible utilizando un dinamómetro de chasis para incrementar la precisión del instrumento de medida.

Se recomienda validar el desempeño de la medición de velocidad y posición utilizando un GPS profesional y un sistema de medición de velocidad en tiempo real con pruebas de distancia-tiempo controladas, así como la instalación de un odómetro para mayor precisión.

Se recomienda para efectos de distribución de la aplicación móvil la subida a las diferentes plataformas de distribución de aplicaciones como Play Store de Android o App Store de Apple debido a que estas son las plataformas con un mayor número de usuarios.

Se recomienda enriquecer el diseño de la carcasa del dispositivo para agregar mayor seguridad de los componentes internos, así como una mayor facilidad de instalación con sistemas mecánicos.

9. CRONOGRAMA

Figura 90. Diagrama de Gantt

Actividad		Sin iniciar (0%-50%)	Iniciada (50%-99%)	Finalizada(99%-100%)														
Código de colores																		
Objetivo Específico	Actividad	Semanas																
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	Recopilación Bibliográfica																	
	Estudio del funcionamiento del motor de combustión interna de una motocicleta carburada																	
	Estudio del funcionamiento de los módulos GPS, Bluetooth y demás librerías necesarias para el microcontrolador																	
	Sustentación de Anteproyecto																	
2	Selección de materiales e instrumentación del sistema																	
3	Diseño electrónico y mecánico del dispositivo en software especializado (Proteus, Solidworks, Labview)																	
	Primera Sustentación																	
	Adquisición de los componentes	PERIODO INTERSEMESTRAL																
4	Desarrollo de la interfaz para la visualización de los datos y la subida a la nube																	
Objetivo Específico	Actividad	Semanas																
		18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
3	Calibración de los componentes																	
5	Construcción del sistema instrumentado																	
	Verificación del funcionamiento de cada uno de los subsistemas para asegurar la integración con el microcontrolador																	
	Implementación del sistema en condiciones de laboratorio																	
6	Validación del funcionamiento del dispositivo en laboratorio																	
5	Implementación del dispositivo para monitorear la motocicleta en condiciones reales																	
6	Validación del dispositivo semi-industrial y adquisición de datos en condiciones reales de ruta																	
Objetivo general	Entrega final																	

10. RECURSOS

Tabla 20. Recursos económicos del proyecto

Subsistema	Elemento	Valor	Total
Banco de calibración	Driver A4988	\$20,000	\$490,000
	BSS 138	\$30,000	
	Cargador	\$15,000	
	Uniones hidráulicas	\$40,000	
	Cables	\$35,000	
	Encoder óptico	\$35,000	
	Sistema Carro Actuator CNC	\$300,000	
	Soporte	\$15,000	
Medición y procesamiento de variables	Flujómetro SLF3S-1300F	\$1,007,926	\$1,550,926
	Racores	\$40,000	
	Cables JST	\$15,000	
	Elementos circuito adaptador de señal	\$25,000	
	NodeMCU ESP8266	\$38,000	
	GPS Beitian BN-220	\$150,000	
	PCB	\$50,000	
	Batería Portable	\$100,000	
	Impresión 3D	\$50,000	
	Energía Blynk	\$25,000	
	Otros	\$50,000	
	TOTAL		

11. REFERENCIAS

- [1] Tong, H. Y.; Hung, W. T., & Cheung, C. S. (2000). On-road motor vehicle emissions and fuel consumption in urban driving conditions. *Journal of the Air & Waste Management Association* (1995), 50(4), 543–554.
- [2] Ligterink, N. E., & Eijk, A. R. A. (2014). Update analysis of real-world fuel consumption of business passenger cars based on Travelcard Nederland fuelpass data. July, 25. [Consultado: 12 de agosto de 2021]. Disponible: <https://doi.org/10.13140/RG.2.1.3212.5040>
- [3] Andemos, D. P. P. (2020). Informe Motos Junio Colombia, 2020. Recuperado 11 de agosto de 2021, de Andemos.org. [Consultado: 8 de agosto de 2021]. Disponible: <https://andemos.org/wp-content/uploads/2020/07/Informe-Motos-2020-6.pdf>
- [4] Vanguardia. Bucaramanga. 5 de agosto de 2021. 147% aumentó en Bucaramanga y su área el parque automotor. [Consultado: 3 de agosto de 2021]. Disponible: <https://www.vanguardia.com/area-metropolitana/bucaramanga/147-aumento-en-bucaramanga-y-su-area-el-parque-automotor-CX4088712>
- [5] Alcaldía de Bucaramanga. 20 de abril de 2021. El parque automotor de Bucaramanga y el área metropolitana asciende a los 760.746 vehículos. Disponible: <https://www.bucaramanga.gov.co/noticias/el-parque-automotor-de-bucaramanga-y-el-area-metropolitana-asciende-a-los-760-746-vehiculos/>
- [6] REGISTRO ÚNICO NACIONAL DE TRANSITO RUNT. Parque automotor registrado en RUNT [en línea]. [Consultado: 30 de septiembre de 2021]. Disponible: <https://www.runt.com.co/runt-en-cifras/parque-automotor>
- [7] LAS MOTOCICLETAS EN COLOMBIA. Cámara de la Industria Automotriz de la Asociación Nacional de Empresarios de Colombia, ANDI (2017). [Consultado: 11 de agosto de 2021]. Disponible: <http://www.andi.com.co/Uploads/LasMotocicletasEnColombia.pdf>
- [8] Mantilla, Ó. J. R. (2020). ¿Qué tan seguras son las 10 motos más vendidas en Colombia? [en línea]. (23 de marzo de 2020). [Consultado: 11 de agosto de 2021]. Disponible: <https://www.elcarrocolombiano.com/dos-ruedas/que-tan-seguras-son-las-10-motos-mas-vendidas-en-colombia/>
- [9] Tzeng, G.-H., & Chen, J.-J. (1998). Developing a taipei motorcycle driving cycle for emissions and fuel economy. *Transportation Research Part D: Transport and Environment*, 3(1), 19–27.

- [10] Chen, K. S., Wang, W. C., Chen, H. M., Lin, C. F., Hsu, H. C., Kao, J. H., & Hu, M. T. (2003). Motorcycle emissions and fuel consumption in urban and rural driving conditions. *Science of the Total Environment*, 312(1–3), 113–122. Disponible: [https://doi.org/10.1016/S0048-9697\(03\)00196-7](https://doi.org/10.1016/S0048-9697(03)00196-7)
- [11] Beusen, B., Broekx, S., Denys, T., Beckx, C., Degraeuwe, B., Gijsbers, M., ... & Panis, L. I. (2009). Using on-board logging devices to study the longer-term impact of an eco-driving course. *Transportation research part D: transport and environment*, 14(7), 514-520
- [12] Johnson, D. A., & Trivedi, M. M. (2011). Driving style recognition using a smartphone as a sensor platform. In 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC) (pp. 1609-1615). IEEE.
- [13] Santamaria, A., Gamboa, H. A. B., & Aguilar, Á. A. P. (2014). Diseño e implementación del control electrónico de dosificación de combustible para un motor de motocicleta. Ecuador: Escuela Politécnica Nacional.
- [14] Satiennam, T., Seedam, A., Radpukdee, T., Satiennam, W., Pasangtiyo, W., & Hashino, Y. (2017). Development of on-road exhaust emission and fuel consumption models for motorcycles and application through traffic microsimulation. *Journal of Advanced Transportation*, 2017.
- [15] Duc, K. N., Nguyen, Y. L. T., Duy, T. N., Nghiem, T. D., Le, A. T., & Huu, T. P. (2020). A robust method for collecting and processing the on-road instantaneous data of fuel consumption and speed for motorcycles. *Journal of the Air & Waste Management Association*, 71(1), 81-101.
- [16] Kaiser, C., Stocker, A., Festl, A., Petrovic, M. D., Papatheocharous, E., Wallberg, A., Ezquerro, G., Orbe, J. O., Szilagyi, T., & Fellmann, M. (2020). A vehicle telematics service for driving style detection: Implementation and privacy challenges. *VEHITS 2020 - Proceedings of the 6th International Conference on Vehicle Technology and Intelligent Transport Systems*, July, 29–36.
- [17] Heywood, J. B. (1988). Internal Combustion Engine (ICE) Fundamentals. In *Handbook of Clean Energy Systems* (Vol. 21). [Consultado: 28 de agosto de 2021]. Disponible: <https://doi.org/10.1002/9781118991978.hces07>
- [18] Andemos, T. Inventario de emisiones vehículos de Colombia [en línea]. Bogotá. (25 de enero de 2017).[Consultado: 11 de agosto de 2021].Disponible: <https://andemos.org/wp-content/uploads/2018/12/Parque-Automotor-Colombia-a-2017.pdf>

- [19] Andemos, T. Inventario de Emisiones Vehículos Colombia [en línea]. Bogotá. (25 de enero de 2017). [Consultado: 11 de agosto de 2021]. Disponible: <https://andemos.org/wp-content/uploads/2017/03/Presentacion-ANDEMOS-Emisiones-01.25.2017-v8.pdf>
- [20] De la A León, Israel K. Diseño y construcción de un módulo medidor de velocidad y aceleración del vehículo por Arduino y GPS. Ibarra, Ecuador: Universidad Internacional del Ecuador. Facultad de ingeniería. Departamentos de ingeniería. 2017.116 p. [Consultado: 20 de agosto de 2021]. Disponible: <http://repositorio.utn.edu.ec/handle/123456789/6826>
- [21] Quitama, F., & Tulcán, H. (2017). Sistema De Control Electrónico De Velocidad Máxima Y Gps En Un Vehículo. Ibarra, Ecuador: Universidad técnica del norte. Facultad de ingeniería en ciencias aplicadas. Departamento de ingeniería. 2017.116 p. [Consultado: 20 de agosto de 2021]. Disponible: http://repositorio.utn.edu.ec/bitstream/123456789/6826/1/04MAU022TRABAJO_DE_GRADO.pdf
- [22] Pozo Ruz, A., Ribeiro, A., García-Alegre, M. C., García, L., Guinea, D., & Sandoval, F. (2000). Sistema De Posicionamiento Global (Gps): Descripción, Análisis De Errores, Aplicaciones Y Futuro. ETS Ingenieros de Telecomunicaciones. Universidad de Malaga, 174. [Consultado: 9 de septiembre de 2021]. Disponible: <http://www.oocities.org/es/forogps/infografia/gps5.pdf>
- [23] Felipe, C., & Bonells, U. (2015). La ventaja de la motocicleta en su velocidad media de recorrido. Épsilon, 2015, 197–214. [Consultado: 10 de octubre de 2021]. Disponible: <https://docplayer.es/19138358-La-ventaja-de-la-motocicleta-en-su-velocidad-media-de-recorrido.html>
- [24] Society of Automotive Engineers, SAE. (1983). Surface Vehicle Recommend Practice. Electric Speedometer Specification – On Road, 4970(724), 1–9.
- [25] Diario Oficial de la Unión Europea. (2018). Actos adoptados por órganos creados mediante acuerdos internacionales. 78, 30–59.
- [26] Van Basshuysen, & Schäfer, F. (2004). Internal combustion engine handbook: basics, components, systems, and perspectives. SAE International. Disponible: <https://www.sae.org/publications/books/content/r-434/>
- [27] Revista moto: ¿Sabes cómo funciona el sistema de inyección electrónica en tu moto? Medellín: Julio,2017. [Consultado: 27 de agosto de 2021]. Disponible en: https://revistamoto.com/wp_rm/sabes-como-funciona-el-sistema-de-inyeccion-electronica-en-tu-moto/

- [28] KYES, John. Geotab: ¿Qué significa GPS?. Canadá (Mayo 22 de 2020). [Consultado: 30 de Agosto del 2021]. Disponible: <https://bit.ly/3NPRTIO>
- [29] YAMAHA.YBR125 Manual del propietario [en línea]. [Consultado: 2 de diciembre de 2021]. Disponible: <https://www.yamaha-motor.com.ar/descargar-manual-de-servicio/ybr125-ed>
- [30] OPM INTEGRAL. Servicios integrales de gestión organizacional de proyectos [en línea]. [Consultado:28 de noviembre de 2021]. Disponible: <https://opmintegral.com/>
- [31] Revista Gestión, Competitividad e Innovación [en línea]. Politécnico de la Costa Atlántica, Claustro Santa Bernardita, Barranquilla, Colombia, Julio-diciembre, 2018. ISSN:2322-7184. Disponible: <https://pca.edu.co/editorial/revistas/index.php/gci/article/view/45/43>
- [32] LINK LABS. Indoor asset tracking system [en línea]. [Consultado: 30 de noviembre de 2021]. Disponible: <https://www.link-labs.com/>
- [33] BESERVICES. Integración de entornos on premise, híbridos y multicloud [en línea]. [Consultado: 1 de diciembre de 2021]. Disponible: <https://www.beservices.es/>
- [34] DATASHEET SENSIRION. Integración de entornos on premise, híbridos y multicloud [en línea]. [Consultado: 2 de diciembre de 2021]. Disponible: <https://sensirion.com/>
- [35] WEBINAR. More tan Flow: Thermal conductivity with the SLF3C-1300F [en línea]. [Consultado: 26 de septiembre de 2021]. Disponible: <https://app.gotowebinar.com/unified/index.html#/webinar/5624794040462639883/attend/5567376445935915534>
- [36] INTERNET ARCHIVE. Datasheet de GPS [en línea]. [Consultado: 30 de agosto de 2021]. Disponible: <https://archive.org/details/bn-220-gps-antenna>
- [37] SENSIRION.Tutorial: Arduino interfacefor SF06 liquid Flow sensors with level shifter[en línea].[Consultado:25 de agosto de 2021]. Disponible: <https://developer.sensirion.com/archive/tutorials/arduino-interface-for-sf06-liquid-flow-sensors-with-level-shifter/>
- [38] NPERF. Mapa de cobertura 3G/4G/5G en Bogotá, Colombia [en línea]. [Consultado: 5 de septiembre de 2021]. Disponible: <https://www.nperf.com/es/map/CO/3688689.Bogota/-/signal/>
- [39] T. Daengsi, P. Ungkap & P. Wuttidittachotti. A Study of 5G Network Performance: A Pilot Field Trial at the Main Skytrain Stations in Bangkok: International Conference on Artificial Intelligence and Computer Science Technology

- (ICAICST),2021. pp. 191-195, [Consultado: 15 de marzo de 2022]: Disponible: <https://ieeexplore.ieee.org/abstract/document/9497810>
- [40] T. Daengsi, P. Ungkap and P. Wuttidittachotti. 5G Network Performance: A Study using Stationary and Mobility Tests on Sukhumvit Line – BTS Skytrain in Bangkok, 2021. 4th International Conference of Computer and Informatics Engineering (IC2IE), 2021, pp. 447-450. Disponible: <https://ieeexplore.ieee.org/abstract/document/9649216>
- [41] DATASHEET SENSIRION SLQ-HC60. Milliliter Flow Meter for Hydrocarbons [en línea]. [Consultado: 1 de diciembre de 2021]. Disponible: <https://sensirion.com/>
- [42] DATASHEET SENSIRION SLQ-HC60. SLI Liquid Flow Meter Series [en línea]. [Consultado: 1 de diciembre de 2021]. Disponible: <https://sensirion.com/>
- [43] DATASHEET SENSIRION LS32-1500. Compact Flow Meter for Low Flow Rates [en línea]. [Consultado: 1 de diciembre de 2021]. Disponible: <https://sensirion.com/>
- [44] DATASHEET SENSIRION LG16. Compact Liquid Flow Meter for OEM Applications [en línea]. [Consultado: 1 de diciembre de 2021]. Disponible: <https://sensirion.com/>
- [45] ALIEXPRESS. Mini turbina Digital medidor de flujo de combustible Diesel [en línea]. [Consultado: 31 de noviembre de 2021]. Disponible: <https://shortest.link/1sEy>
- [46] SENTRONICS. Datasheet FlowSonic®LF Low-Flow Sensor [en línea]. [Consultado: 29 de noviembre de 2021]. Disponible: <https://www.sentrionics.com/wpcontent/uploads/2018/06/FlowSonicDataSheetLFD S.pdf>
- [47] AMAZON. Tacómetro Trail Tech 752-118 [en línea]. [Consultado: 1 de noviembre de 2021]. Disponible: https://www.amazon.com/Trail-Tech-752-118-tac%C3%B3metro-1987-2019/dp/B07BYS2M56?ref=ast_sto_dp&th=1&psc=1
- [48] AMAZON. Tacómetro Trail Tech 752-118 [en línea]. [Consultado: 1 de noviembre de 2021]. Disponible: https://www.amazon.com/Trail-Tech-752-118-tac%C3%B3metro-1987-2019/dp/B07BYS2M56?ref=ast_sto_dp&th=1&psc=1
- [49] ALIBABA. Sensor Winston CS-SM12-3002NB [en línea]. [Consultado: 1 de noviembre de 2021]. Disponible: <https://acortar.link/CbX6D3>
- [50] OMRON Electronics LLC. Encoder E6B2-CWZ6C Datasheet [en línea]. [Consultado: 1 de noviembre de 2021]. Disponible: <https://acortar.link/7beMgk>

[51] Manual Sunding SD-563B. Odómetro de bicicleta SD-563B [en línea]. [Consultado: 1 de noviembre de 2021]. Disponible: <https://www.manuales.com.co/sunding/sd-563a/manual>

[52] MercadoLibre. Sensor de Cable Imán Velocímetro Digital Odómetro Tacómetro ATV Motocicleta [en línea]. [Consultado: 1 de noviembre de 2021]. Disponible: <https://acortar.link/JxM2jO>

[53] MercadoLibre. Sensor de Cable Imán Velocímetro Digital Odómetro Tacómetro ATV Motocicleta [en línea]. [Consultado: 1 de noviembre de 2021]. Disponible: <https://acortar.link/JxM2jO>

[54] AMAZON. ELING Velocímetro GPS digital [en línea]. [Consultado: 5 de noviembre de 2021]. Disponible: <https://acortar.link/O9nU77>

[55] U-blox. NEO-6 u-blox 6 GPS Modules [en línea]. [Consultado: 5 de noviembre de 2021]. Disponible: <https://acortar.link/8SjD7h>

[56] ISO/IEC. Norma Internacional ISO/IEC 17025 Segunda Edición, 2005 [en línea]. [Consultado: 15 de febrero de 2022]. Disponible: <http://integra.cimav.edu.mx/intranet/data/files/calidad/documentos/externos/ISO-IEC-17025-2005.pdf>

12. ANEXOS

Anexo 1. Código Arduino Sistema Bombeo

```
int encoder_pin = 2;           //Pin 2, donde se conecta el encoder
int estado = 0;
int dirButn = 9;
int pare = 10;
float rpm = 0;                 // Revoluciones por minuto calculadas.
float flujo = 0;               //Velocidad en [Km/h]
float timeold = 0; // Tiempo
static volatile unsigned long debounce = 0; // Tiempo del rebote.
float time18D = 0;
float promRPM = 0;

// Define pin connections & motor's steps per revolution
const int dirPin = 5;
const int stepPin = 6;
const int stepsPerRevolution = 800;

// V E L O C I T Y   C O N T R O L
int delayTime(4000);

///// Configuración del Arduino /////
void setup() {
  Serial.begin(9600); // Configuración del puerto serie
  pinMode(encoder_pin, INPUT); // Configuración del pin n°2
  pinMode(dirPin, INPUT_PULLUP);
  pinMode(12, INPUT_PULLUP);
  rpm = 0;
  timeold = 0;

  // Declare pins as Outputs
  pinMode(stepPin, OUTPUT);
  pinMode(dirPin, OUTPUT);
  pinMode(dirButn, INPUT);
  pinMode(pare, INPUT);
}

///// Programa principal /////
void loop() {
  estado = digitalRead(dirButn);
  if (estado == HIGH) {
    digitalWrite(dirPin, LOW);
  }
  else {
    digitalWrite(dirPin, HIGH);
  }

  while (digitalRead(pare) == HIGH) {
    for (int x = 0; x < stepsPerRevolution; x++) {
      if (digitalRead(pare) == LOW) {
```


Anexo 2. Código Arduino Flujómetro

```
#include <Wire.h> // Arduino library for I2C

// -----
// Sensor specific settings, adjust if needed:
// -----

const int ADDRESS = 0x08; // Sensor I2C Address
const float SCALE_FACTOR_FLOW = 500.0; // Scale Factor for flow rate
measurement
const float SCALE_FACTOR_TEMP = 200.0; // Scale Factor for temperature
measurement
const char *UNIT_FLOW = " ml/min"; //physical unit of the flow rate
measurement
const char *UNIT_TEMP = " deg C"; //physical unit of the temperature
measurement
float add_flow = 0.0;
float flow_total = 0.0;
float cont = 0.0;

// -----
// Arduino setup routine, just runs once:
// -----
void setup() {
  int ret;

  Serial.begin(9600); // initialize serial communication
  Wire.begin(); // join i2c bus (address optional for master)

  do {
    // Soft reset the sensor
    Wire.beginTransaction(0x00);
    Wire.write(0x06);
    ret = Wire.endTransmission();
    if (ret != 0) {
      Serial.println("Error while sending soft reset command,
retrying...");
      delay(500); // wait long enough for chip reset to complete
    }
  } while (ret != 0);

  delay(50); // wait long enough for chip reset to complete
}

// -----
// The Arduino loop routine runs over and over again forever:
// -----
```

```

void loop() {
  int ret;
  uint16_t aux_value;
  uint16_t sensor_flow_value;
  uint16_t sensor_temp_value;
  int16_t signed_flow_value;
  int16_t signed_temp_value;
  float scaled_flow_value;
  float scaled_temp_value;
  float fuel_flow
  ;
  byte aux_crc;
  byte sensor_flow_crc;
  byte sensor_temp_crc;

  // To perform a measurement, first send 0x3608 to switch to continuous
  // measurement mode (H2O calibration), then read 3x (2 bytes + 1 CRC
byte) from the sensor.
  // To perform a IPA based measurement, send 0x3615 instead.
  // Check datasheet for available measurement commands.
  Wire.beginTransmission(ADDRESS);
  Wire.write(0x36);
  Wire.write(0x08);
  ret = Wire.endTransmission();
  if (ret != 0) {
    Serial.println("Error during write measurement mode command");
  } else {
    delay(100);

    for (int i = 0; i < 10; ++i) {
      delay(100); // Delay de Medidas
      Wire.requestFrom(ADDRESS, 9);
      if (Wire.available() < 9) {
        Serial.println("Error while reading flow measurement");
        continue;
      }

      sensor_flow_value = Wire.read() << 8; // read the MSB from the
sensor
      sensor_flow_value |= Wire.read(); // read the LSB from the
sensor
      sensor_flow_crc = Wire.read();
      sensor_temp_value = Wire.read() << 8; // read the MSB from the
sensor
      sensor_temp_value |= Wire.read(); // read the LSB from the
sensor
      sensor_temp_crc = Wire.read();
      aux_value = Wire.read() << 8; // read the MSB from the
sensor
      aux_value |= Wire.read(); // read the LSB from the
sensor
      aux_crc = Wire.read();

```



```

    signed_flow_value = (int16_t) sensor_flow_value;

    scaled_flow_value = ((float) signed_flow_value) /
SCALE_FACTOR_FLOW;

    if (scaled_flow_value < 0) {
        fuel_flow = 0.0;
    } else if (scaled_flow_value >= 0.0 && scaled_flow_value <= 1.8) {
        fuel_flow = 6.8154 * scaled_flow_value * scaled_flow_value +
14.353 * scaled_flow_value - 1.9071;
    } else if (scaled_flow_value > 1.8) {
        fuel_flow = -6.1579 * scaled_flow_value * scaled_flow_value +
68.145 * scaled_flow_value - 60.529;
    }

    if (fuel_flow < 0.0) {
        fuel_flow = 0;
    }

    Serial.print("Fuel flow value step: ");
    Serial.print(fuel_flow);
    Serial.print(UNIT_FLOW);

    add_flow = add_flow + fuel_flow;
    cont = cont + 1.0;

    if (cont == 10) {
        flow_total = add_flow / cont;

        add_flow = 0.0;
        cont = 0.0;
    }

    Serial.print(", Fuel flow value: ");
    Serial.print(fuel_flow);
    Serial.print(UNIT_FLOW);
    Serial.println("");
}
// To stop the continuous measurement, first send 0x3FF9.
Wire.beginTransaction(ADDRESS);
Wire.write(0x3F);
Wire.write(0xF9);
ret = Wire.endTransmission();
if (ret != 0) {
    Serial.println("Error during write measurement mode command");
}
}

delay(100); // milliseconds delay between reads (for demo purposes)
}

```

Anexo 3. Código Google Script

```
function doGet(e) {
  Logger.log( JSON.stringify(e) );
  var result = 'OK';
  if (e.parameter == 'undefined') {
    result = 'No Parameters';
  }

  else {
    var sheet_id = '1I3HbMTIju_9oTkg1Sj-
ZSt8WFibEWNaG6gwAfPn0C00'; // Spreadsheet ID
    var sheet = SpreadsheetApp.openById(sheet_id).getActiveSheet();
    var newRow = sheet.getLastRow()+1;
    var rowData = [];
    var Curr_Date = new Date();
    rowData[0] = Curr_Date; // Date in column A
    var Curr_Time = Utilities.formatDate(Curr_Date, "Asia/Jakarta", 'HH:mm:ss');
    rowData[1] = Curr_Time; // Time in column B
    for (var param in e.parameter) {
      Logger.log('In for loop, param=' + param);
      var value = stripQuotes(e.parameter[param]);
      Logger.log(param + ':' + e.parameter[param]);
      switch (param) {
        case 'rpm':
          rowData[2] = value; // Temperature in column C
          result = 'RPM Written on column C';
          break;
        case 'lat':
          rowData[3] = value; // Humidity in column D
          result += ' ,Latitude Written on column D';
          break;
        case 'lon':
          rowData[4] = value; // Humidity in column D
          result += ' ,Longitude Written on column D';
          break;
        case 'alt':
          rowData[5] = value; // Humidity in column D
          result += ' ,Altitud Written on column D';
          break;
        case 'flo':
          rowData[6] = value; // Humidity in column D
          result += ' ,Flow Written on column D';
          break;
        case 'spe':
```

```

        rowData[7] = value; // Humidity in column D
        result += ' ,Speed Written on column D';
        break;
    default:
        result = "unsupported parameter";
    }
}
Logger.log(JSON.stringify(rowData));
var newRange = sheet.getRange(newRow, 1, 1, rowData.length);
newRange.setValues([rowData]);
}
return ContentService.createTextOutput(result);
}
function stripQuotes( value ) {
    return value.replace(/^[\'"]|[\']$/g, "");
}
}

```

Anexo 4. Código NodeMCU

```
//https://script.google.com/macros/s/AKfycbybErizw30tmJdTOO6tHQ2kt1M4vaQy  
CXgGstNn8TiHLUgtG1LYbeu7dj4bFsdLG1lF/exec
```

```
/* GENERAL INFO
```

```
PROJECT DEVELOPED BY: RAY AMAYA - SANTIAGO URIBE  
UNAB - MECHATRONICS ENGINEERING  
COLOMBIA 2022
```

```
VIRTUAL PINS (CLOUD)
```

```
V0 --> MAP GPS
```

```
V1 --> LATITUDE
```

```
V2 --> LONGITUDE
```

```
V3 --> SPEED
```

```
V6 --> Flow
```

```
V7 --> RPM
```

```
V13 --> ALTITUDE
```

```
V8 --> LATITUDE EXPORT
```

```
V9 --> LONGITUDE EXPORT
```

```
V10 --> SPEED EXPORT
```

```
V11 --> Flow EXPORT
```

```
V12 --> RPM EXPORT
```

```
NODEMCU PINS
```

```
D5 --> RPM
```

```
D1 --> SCL Flow
```

```
D2 --> SDA Flow
```

```
D7 --> TX GPS
```

```
D8 --> RX GPS
```

```
*/
```

```
// LIBRARIES
```

```
#include <TinyGPS++.h>
```

```
#include <SoftwareSerial.h>
```

```
#include <BlynkSimpleEsp8266.h>
```

```
#define BLYNK_PRINT Serial
```

```
#include <Wire.h>
```

```
#include <WiFiClientSecure.h>
```

```
// VARIABLES
```

```
// GENERAL
```

```
bool conected = false;
```

```
// LEDS
```

```
#define LEDWIFI 16
```

```
#define LEDFLOW 0
```

```
#define LEDGPS 2
```

```
// EXPORT DATA
```

```

const char* host = "script.google.com";
const int httpsPort = 443;
WiFiClientSecure client; //--> Create a WiFiClientSecure object.

String GAS_ID = "AKfycbxyXhlZdZztRobQgN426LZl7weN-
8KdqO3Z_5ufvnV3qkoA9hp6gt8InCiH3ZeZqmr-"; //--> spreadsheet script ID

//FLOW
const int ADDRESS =
0x08; // Sensor I2C
Address
const float SCALE_FACTOR_FLOW =
500.0; // Scale Factor for flow
rate measurement
const float SCALE_FACTOR_TEMP =
200.0; // Scale Factor for
temperature measurement
const char *UNIT_FLOW = "
ml/min"; // Physical unit of
the flow rate measurement
const char *UNIT_TEMP = " deg
C"; // Physical unit of the
temperature measurement
float fuel_flow;

// RPM
float rpm =
0; // Rpm
calculated
float rpm_filter =
rpm; // Rpm
filtered
float alpha = 0.06;
static volatile unsigned long debounce =
0; // Debounce time.
float timeold = 0;
float time18D = 0;
float rpm_past = 0.0;
int cuenta = 0;

// GPS
static const int RXPin = 13, TXPin = 15;
static const uint32_t GPSBaud = 9600;
float
spd; /
/ Variable to store the speed
float
latitude; /
/ Storing the Lat. and Lon.
float longitude;
float altitud;
unsigned int move_index =
1; // fixed location for
now

```

```

float lastLong, lastLat;
TinyGPSPlus gps;
WidgetMap myMap(V0);

SoftwareSerial ss(RXPin,
TXPin); // The serial
connection to the GPS device
BlynkTimer
timer; //
Timer for write and read data

char auth[] = "FL0fpeVfKIWeB8ectTlRamtjLW-
z5Qdh"; // Authentication key
char ssid[] = "MOBD-
IoT"; // Wifi Name
char pass[] = "mobdiot";

void setup() {

  Serial.begin(115200);
  client.setInsecure();
  Wire.begin();

  // START INTERNET
  Blynk.begin(auth, ssid, pass);
  delay(1500);
  Serial.println("WiFi Connected!");
  delay(1500);

  int ret;
  do {
    Wire.beginTransaction(0x00);
    Wire.write(0x06);
    ret = Wire.endTransmission();
    if (ret != 0) {
      delay(500);
    }
  } while (ret != 0);

  delay(50);

  //INTERRUPCIONES RPM
  pinMode(14, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(14), calculateRPM, RISING);

  // START GPS
  ss.begin(GPSBaud);
  delay(50);

  // READ DATA AND PUSH IT TO CLOUD
  timer.setInterval(2000L, readAndWrite);

  pinMode(LEDFLOW, OUTPUT);
  pinMode(LEDWIFI, OUTPUT);

```

```

pinMode(LEDGPS, OUTPUT);

digitalWrite(LEDFLOW, LOW);
digitalWrite(LEDWIFI, LOW);
digitalWrite(LEDGPS, LOW);

do {
  latitude = (gps.location.lat());
  longitude = (gps.location.lng());
  altitud = (gps.altitude.meters());
  smartDelay(150);
} while (!ss.available());
}

void loop() {
  Blynk.run();
  timer.run();

  if (rpm_past = rpm) {
    cuenta = cuenta + 1;
  }
  rpm_past = rpm;
  if (cuenta >= 3) {
    rpm = 0.0;
    cuenta = 0;
  }

  delay(150);
}

void readAndWrite() {
  readRPM();
  readGPS();
  readFlow();
  WriteData();
}

void WriteData() {

  // Write RPM
  Blynk.virtualWrite(V7, String(int(rpm)));

  // Write LONGITUED & LATITUDE & DISTANCE
  Blynk.virtualWrite(V1, String(latitude, 7));
  Blynk.virtualWrite(V2, String(longitude, 7));
  Blynk.virtualWrite(V13, String(altitud, 7));

  // Write SPEED
  Blynk.virtualWrite(V3, int(spd));

  // Write Flow
  Blynk.virtualWrite(V6, String(fuel_flow, 3));
}

```

```

// Write MAP
myMap.location(move_index, latitude, longitude, "GPS_Location");

// Export data
sendData(latitude, longitude, altitud, spd, rpm, fuel_flow);
}

void readRPM() {
    rpm_filter = rpm_filter;
}

void readGPS() {

    latitude = (gps.location.lat());
    longitude = (gps.location.lng());
    altitud = (gps.altitude.meters());
    spd = (gps.speed.kmph());
    lastLat = latitude;
    lastLong = longitude;
    smartDelay(150);
}

void readFlow() {

    int ret;
    uint16_t aux_value;
    uint16_t sensor_flow_value;
    uint16_t sensor_temp_value;
    int16_t signed_flow_value;
    int16_t signed_temp_value;
    float scaled_flow_value;
    float scaled_temp_value;
    byte aux_crc;
    byte sensor_flow_crc;
    byte sensor_temp_crc;

    Wire.beginTransaction(ADDRESS);
    Wire.write(0x36);
    Wire.write(0x08);
    ret = Wire.endTransmission();
    if (ret != 0) {
    } else {
        delay(100);

        Wire.requestFrom(ADDRESS, 9);
        sensor_flow_value = Wire.read() << 8; // read the MSB from the
sensor
        sensor_flow_value |= Wire.read(); // read the LSB from the
sensor
        sensor_flow_crc = Wire.read();
        sensor_temp_value = Wire.read() << 8; // read the MSB from the
sensor

```



```

    sensor_temp_value |= Wire.read();          // read the LSB from the
sensor
    sensor_temp_crc   = Wire.read();
    aux_value         = Wire.read() << 8; // read the MSB from the
sensor
    aux_value         |= Wire.read();          // read the LSB from the
sensor
    aux_crc           = Wire.read();

    signed_flow_value = (int16_t) sensor_flow_value;
    scaled_flow_value = ((float) signed_flow_value) / SCALE_FACTOR_FLOW;

    if (scaled_flow_value < 0) {
        fuel_flow = 0.0;
    } else if (scaled_flow_value >= 0.0 && scaled_flow_value <= 40) {
        fuel_flow = 6.8154 * scaled_flow_value * scaled_flow_value + 14.353
* scaled_flow_value - 1.9071;
    } else if (scaled_flow_value > 40) {
        fuel_flow = -3.9493 * scaled_flow_value * scaled_flow_value +
52.444 * scaled_flow_value - 37.27;
    }

    if (fuel_flow < 0.0) {
        fuel_flow = 0;
    }

    Wire.beginTransaction(ADDRESS);
    Wire.write(0x3F);
    Wire.write(0xF9);
    ret = Wire.endTransmission();
    if (ret != 0) {
    }
}
}

ICACHE_RAM_ATTR void calculaterPM() {
    if ( (micros() - debounce > 100) ) {
        debounce = micros();
        timeold = millis(); // Almacenamos el tiempo actual.

        float t = (timeold - time18D) / 1000;

        rpm = 360 / (t * 6);
        rpm_filter = alpha * rpm + (1.0 - alpha) * rpm_filter;

        time18D = timeold;
    }
}

static void smartDelay(unsigned long ms)
{
    unsigned long start = millis();
    do
    {

```

```

        while (ss.available())
            gps.encode(ss.read());
    } while (millis() - start < ms);
}

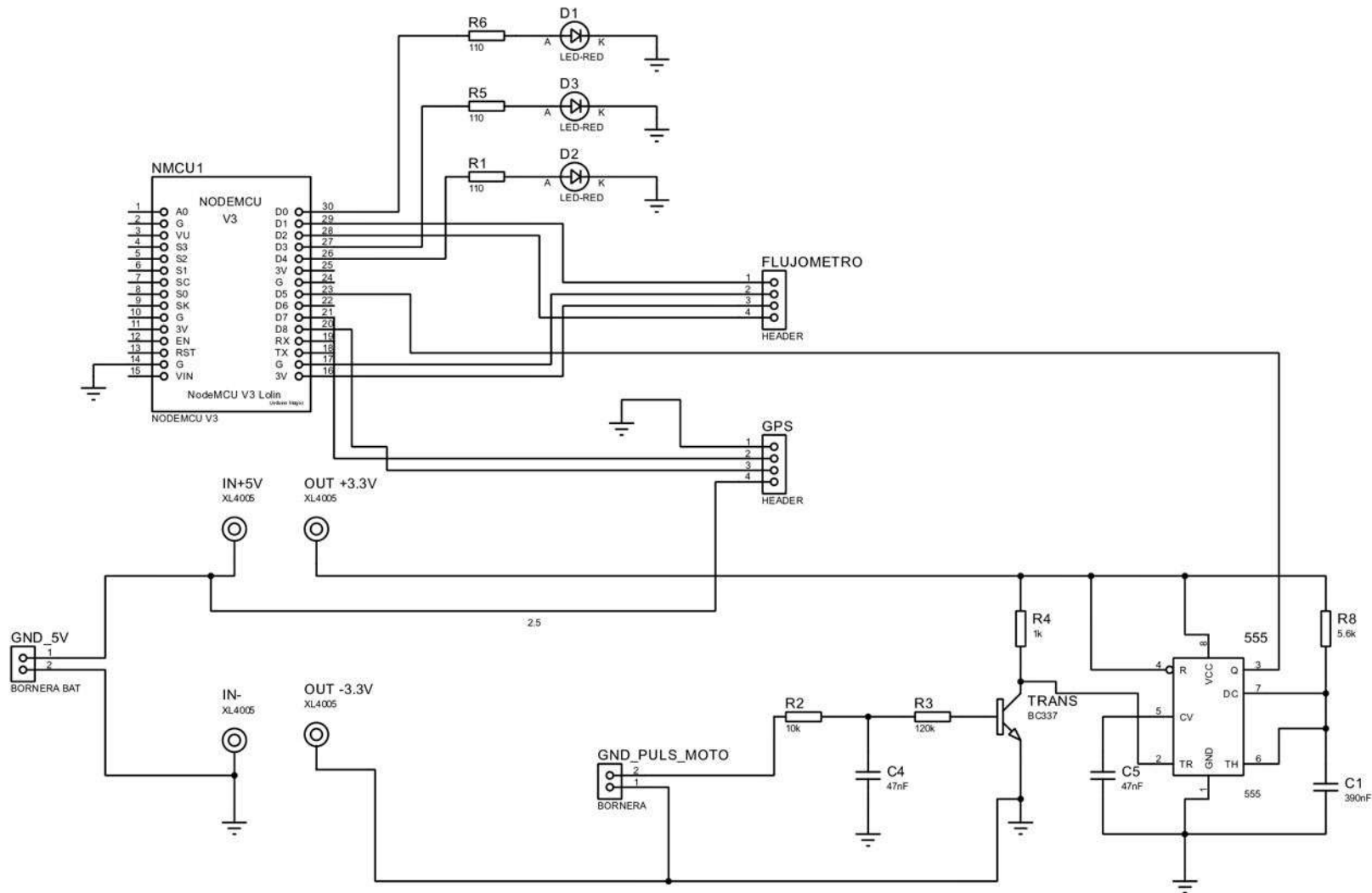
void sendData(float latitude, float longitude, float altitud, int spd,
float rpm, float flow) {
    if (!client.connect(host, httpsPort)) {
        return;
    }
    String string_latitude = String(latitude, DEC);
    String string_altitud = String(altitud, DEC);
    String string_longitude = String(longitude, DEC);
    String string_spd = String(spd, DEC);
    String string_rpm = String(rpm, DEC);
    String string_flow = String(flow, DEC);

    String url = "/macros/s/" + GAS_ID + "/exec?rpm=" + string_rpm +
"&lat=" + string_latitude + "&lon=" + string_longitude + "&alt=" +
string_altitud + "&flo=" + string_flow + "&spe=" + string_spd;

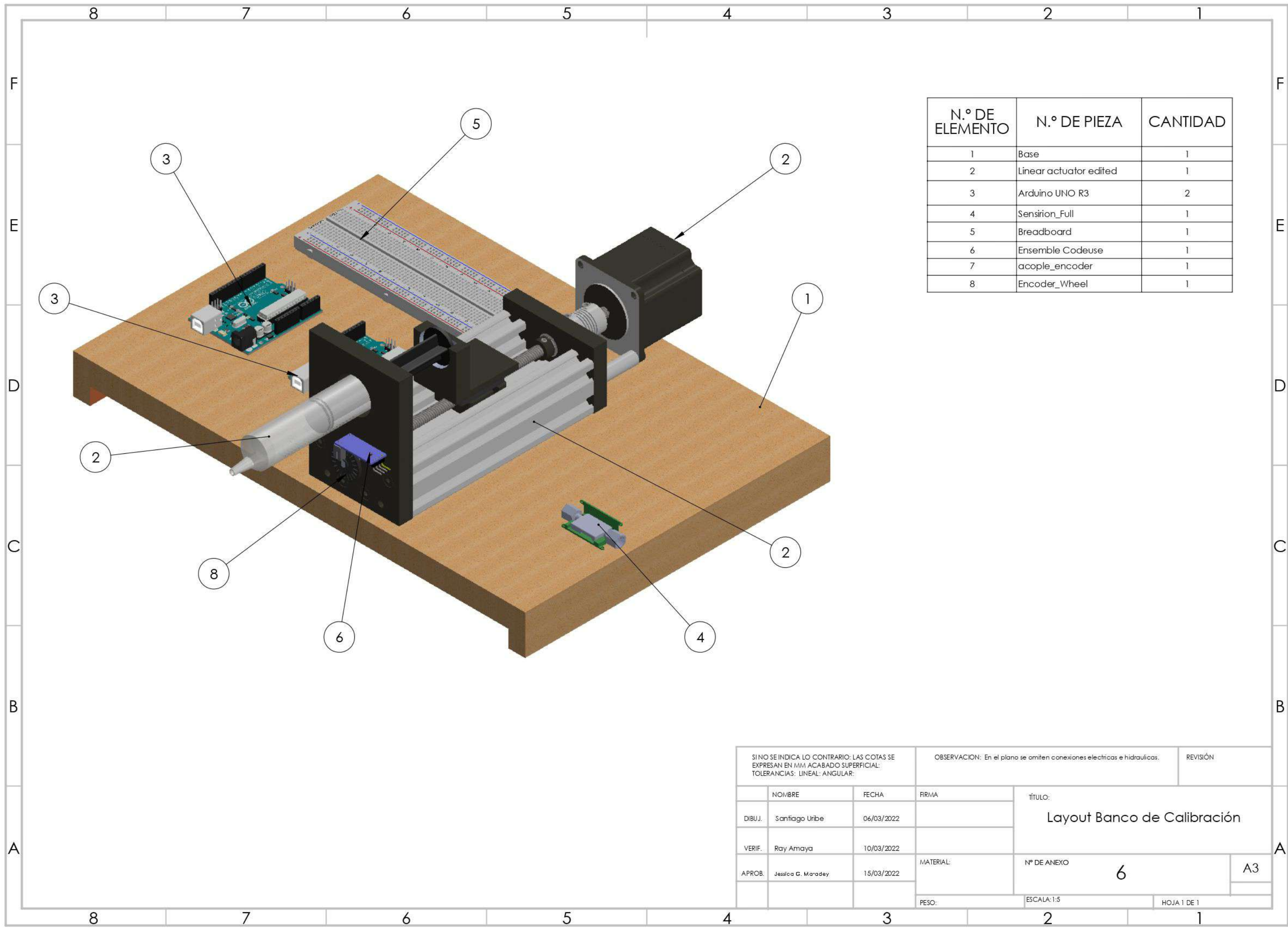
    client.print(String("GET ") + url + " HTTP/1.1\r\n" +
        "Host: " + host + "\r\n" +
        "User-Agent: BuildFailureDetectorESP8266\r\n" +
        "Connection: close\r\n\r\n");

    delay(100);
}

```

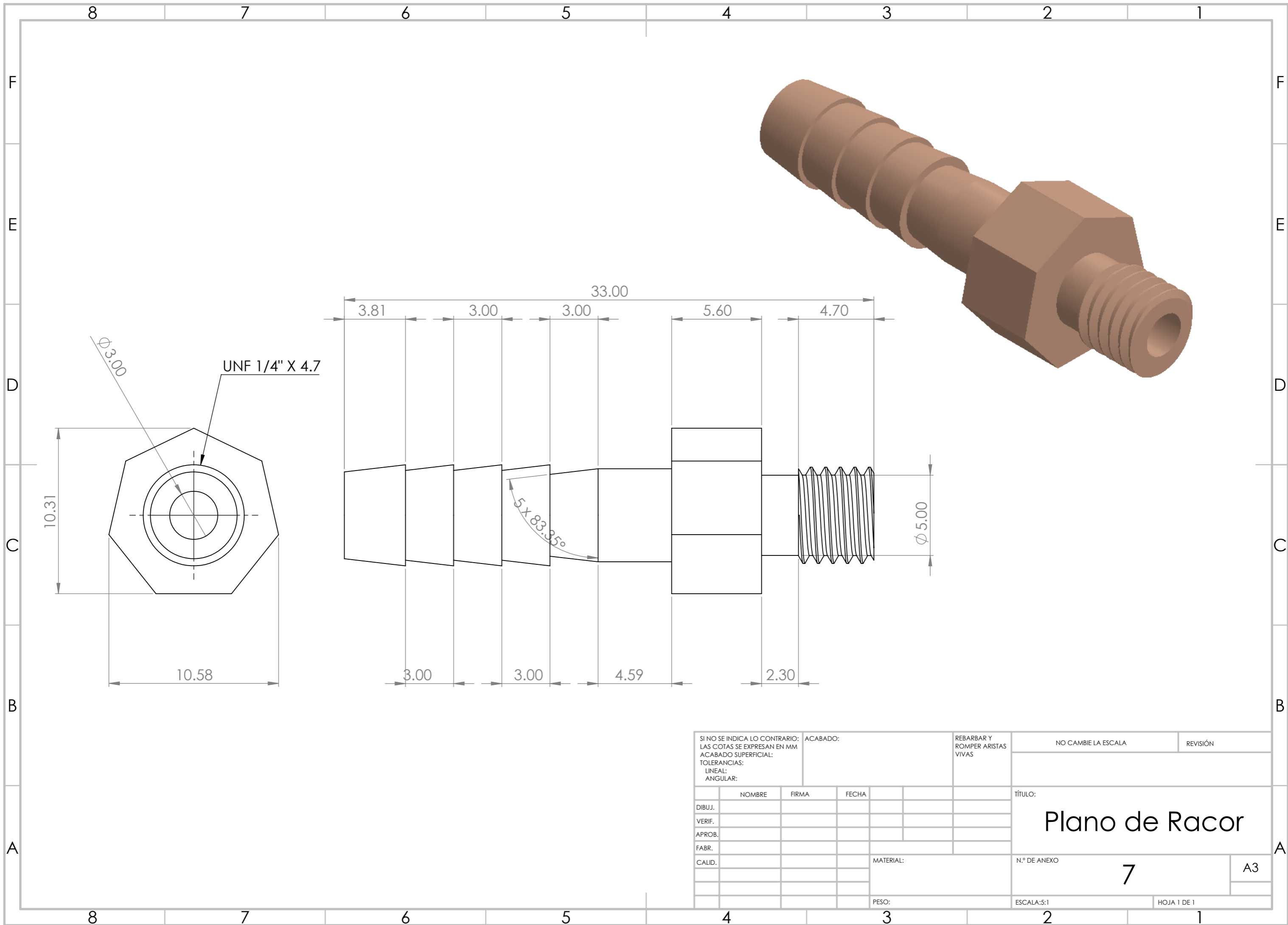


FILE NAME: Placa_MOBD-IoT	DATE: 6/10/2022
DESIGN TITLE: Diagrama Electrónico MOBD-IoT	PAGE: 0 of 1
BY: Santiago Uribe - Ray Amaya	REV: Jessica M.
	TIME: 8:08:04 PM



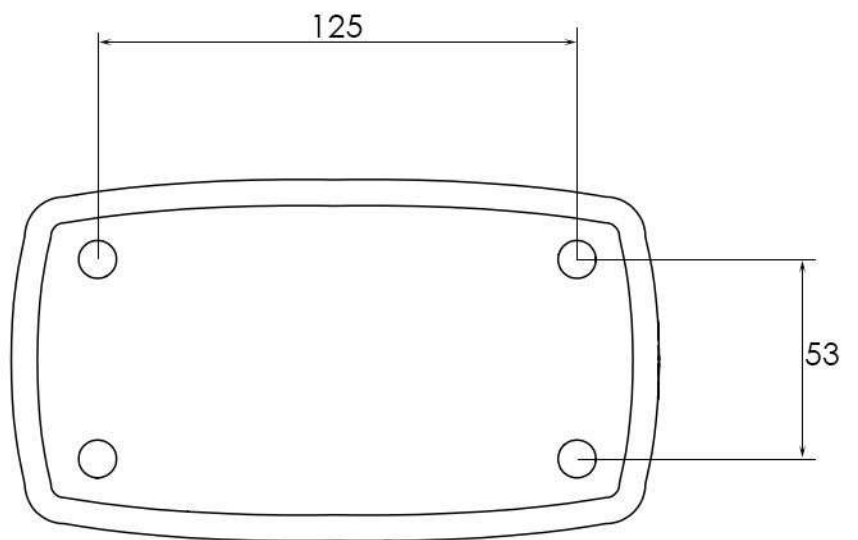
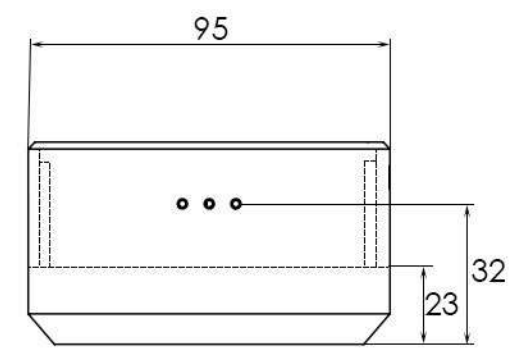
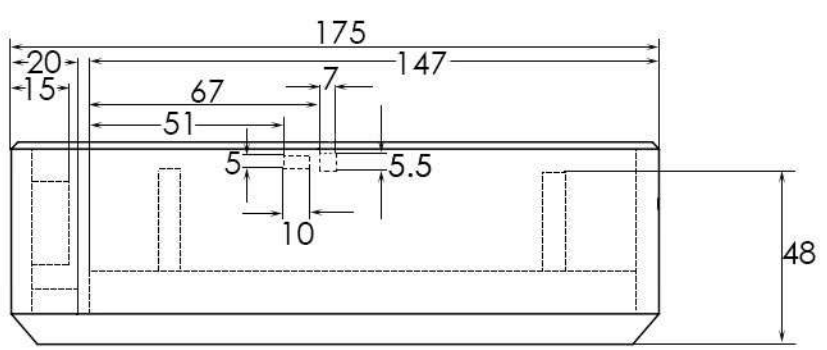
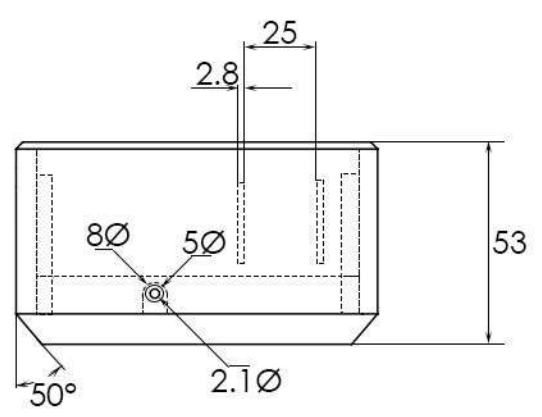
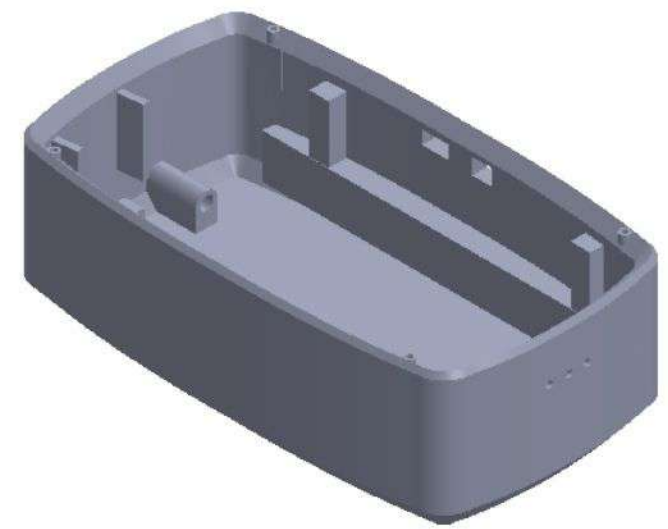
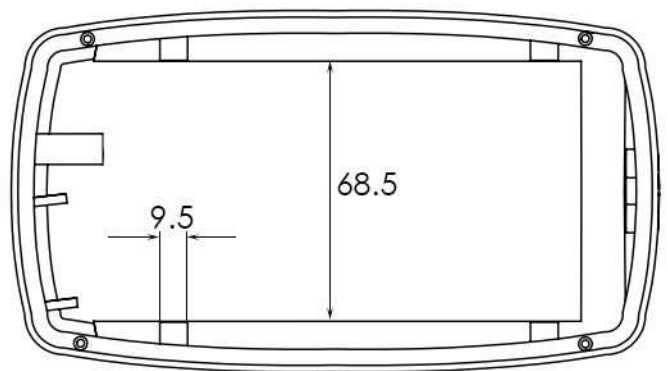
N.º DE ELEMENTO	N.º DE PIEZA	CANTIDAD
1	Base	1
2	Linear actuator edited	1
3	Arduino UNO R3	2
4	Sensirion_Full	1
5	Breadboard	1
6	Ensemble Codeuse	1
7	acople_encoder	1
8	Encoder_Wheel	1

SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			OBSERVACION: En el plano se omiten conexiones electricas e hidraulicas.		REVISIÓN
NOMBRE	FECHA	FIRMA	TÍTULO:		
DIBUJ. Santiago Uribe	06/03/2022		Layout Banco de Calibración		
VERIF. Ray Amaya	10/03/2022		Nº DE ANEXO		A3
APROB. Jessica G. Maradey	15/03/2022	MATERIAL:	6		
		PESO:	ESCALA: 1:5	HOJA 1 DE 1	

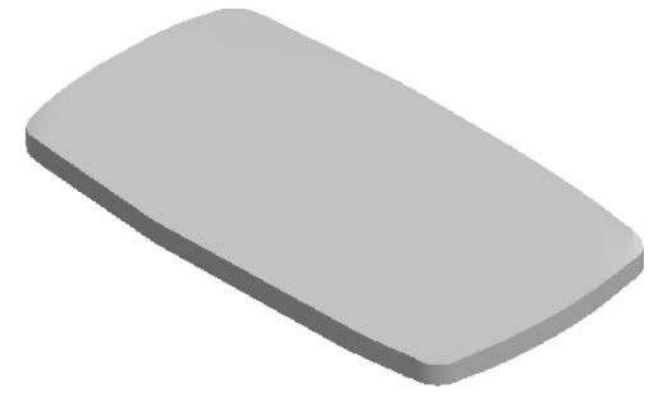
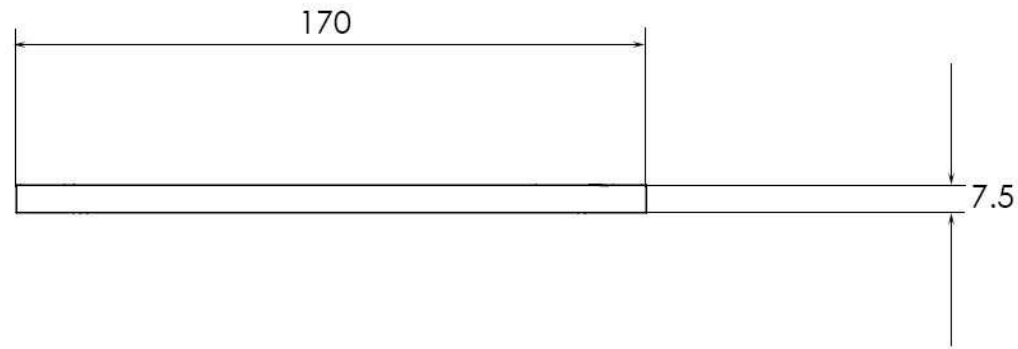
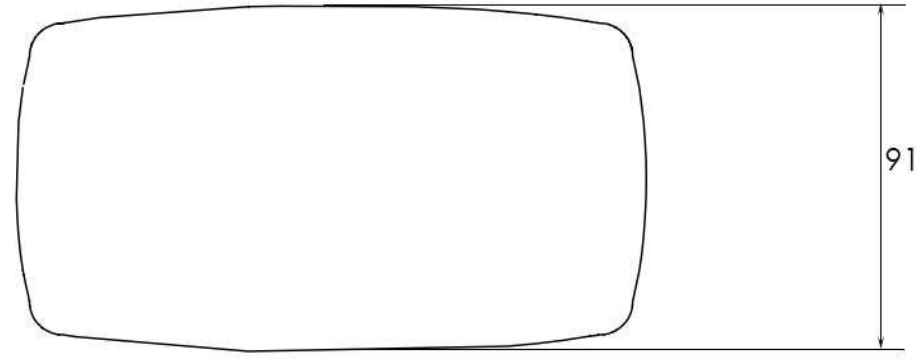


SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			ACABADO:		REBARBAR Y ROMPER ARISTAS VIVAS		NO CAMBIE LA ESCALA		REVISIÓN		
					TÍTULO: Plano de Racor						
					MATERIAL:		N.º DE ANEXO 7			A3	
					PESO:		ESCALA:5:1			HOJA 1 DE 1	

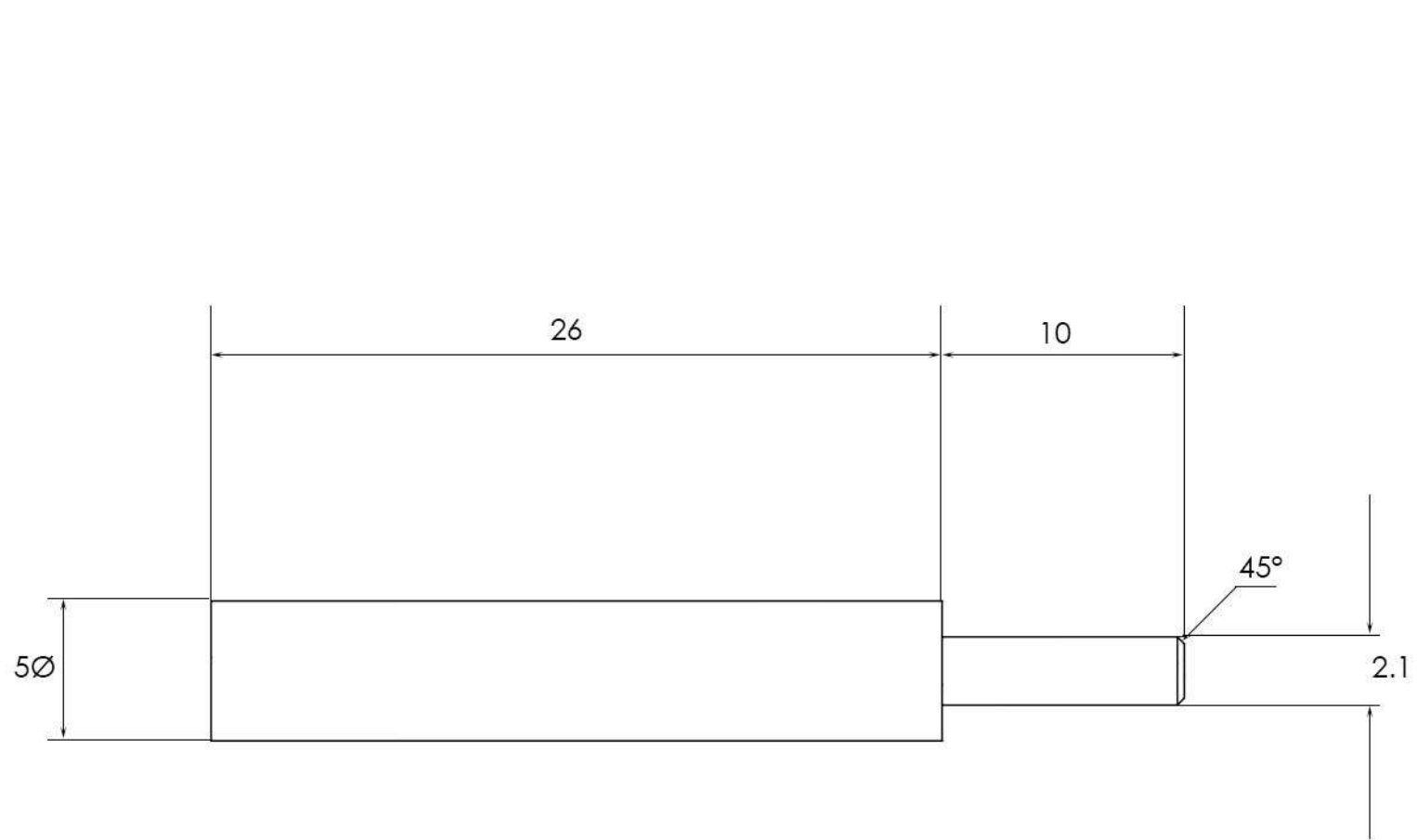
	NOMBRE	FIRMA	FECHA		
DIBUJ.					
VERIF.					
APROB.					
FABR.					
CALID.					



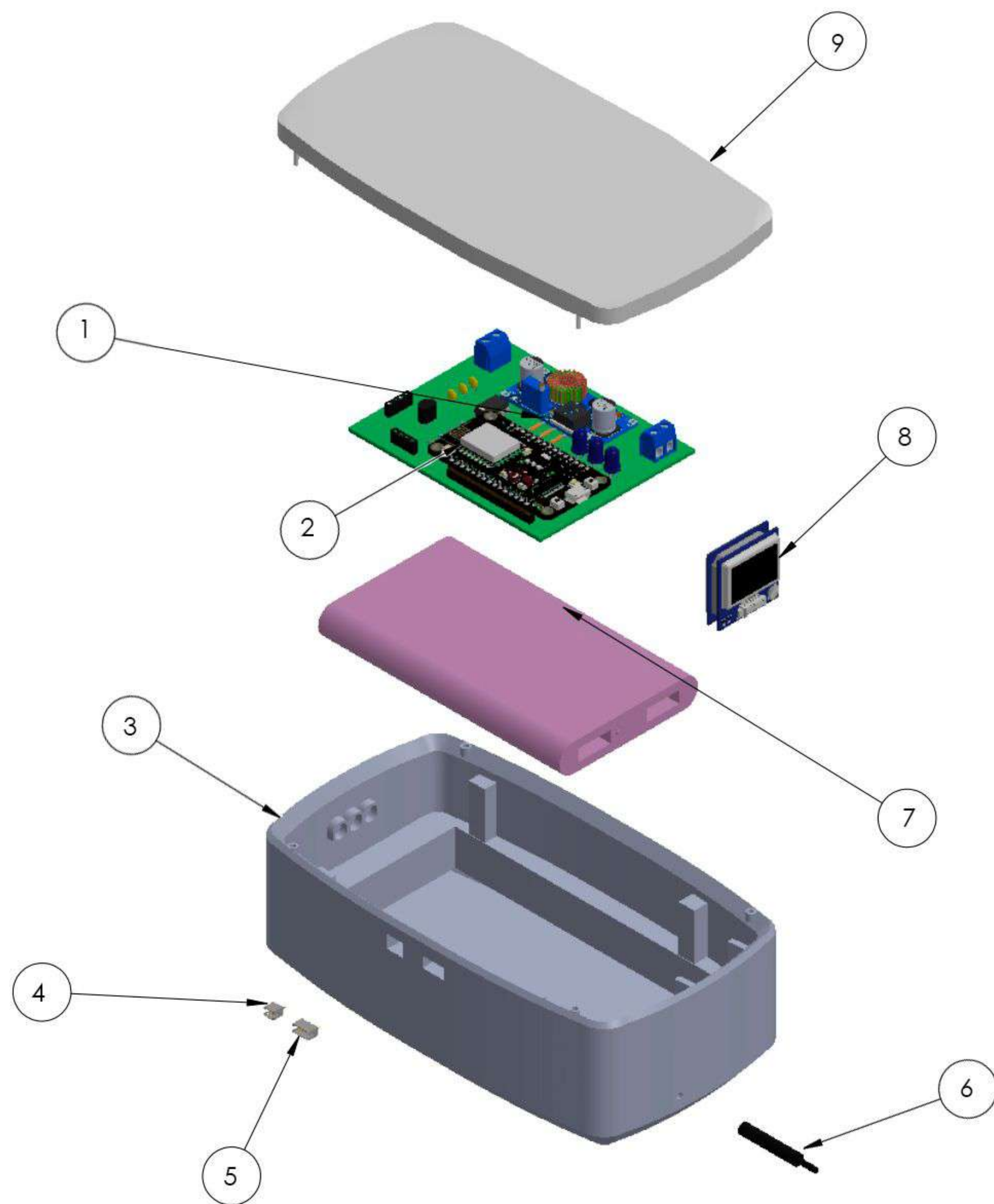
SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			OBSERVACION: En el plano se omiten conexiones electricas e hidraulicas.		REVISIÓN
	NOMBRE	FECHA	FIRMA	TÍTULO:	
DIBUJ.	Santiago Uribe	06/03/2022		Plano de Caja	
VERIF.	Ray Amaya	10/03/2022		N° DE ANEXO	
APROB.	Jessica G. Maradey	15/03/2022	MATERIAL:	8	A3
			PESO:	ESCALA: 1:2	HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			OBSERVACION: En el plano se omiten conexiones electricas e hidraulicas.		REVISIÓN
	NOMBRE	FECHA	FIRMA	TÍTULO:	
DIBUJ.	Santiago Uribe	06/03/2022		Plano de Tapa	
VERIF.	Ray Amaya	10/03/2022		N° DE ANEXO	
APROB.	Jessica G. Maradey	15/03/2022		9	A3
			PESO:	ESCALA: 1:2	HOJA 1 DE 1



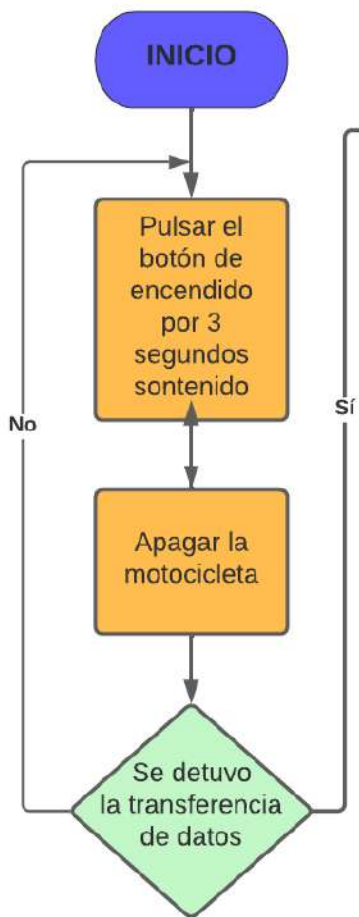
SI NO SE INDICA LO CONTRARIO, LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			OBSERVACION: En el plano se omiten conexiones electricas e hidraulicas.		REVISIÓN
	NOMBRE	FECHA	FIRMA	TÍTULO:	
DIBUJ.	Santiago Uribe	06/03/2022		Plano de Botón	
VERIF.	Ray Amaya	10/03/2022		Nº DE ANEXO	
APROB.	Jessica G. Maradey	15/03/2022	MATERIAL:	10	A3
			PESO:	ESCALA: 2:1	HOJA 1 DE 1



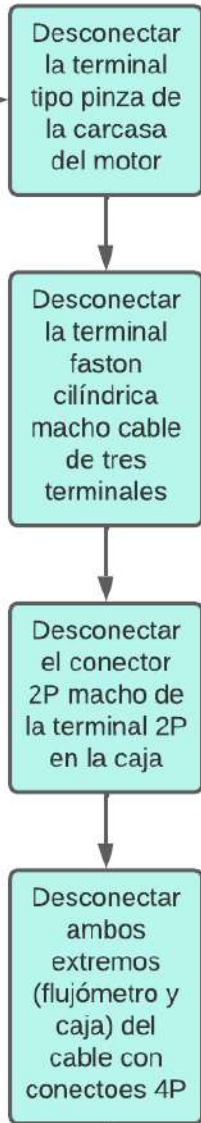
N.º DE ELEMENTO	N.º DE PIEZA	CANTIDAD
1	Placa de Circuito Impreso	1
2	NodeMCU ESP8266	1
3	Caja	1
4	Header 2P rpm	1
5	Header 4P Flujo	1
6	Botón	1
7	Batería	1
8	GPS Beitian BN-220	1
9	Tapa	1

SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			OBSERVACION: En el plano se omite el cableado y la disposición de los LEDs.		REVISIÓN
NOMBRE	FECHA	FIRMA	TÍTULO:		
DIBUJ. Santiago Uribe	06/03/2022		Plano General Explosionado		
VERIF. Ray Amaya	10/03/2022		Nº DE ANEXO		A3
APROB. Jessica G. Maradey	15/03/2022	MATERIAL:	11		
		PESO:	ESCALA: 1:5	HOJA 1 DE 1	

Conexiones Inalámbricas



Conexiones Eléctricas



Conexiones Hidráulicas

