



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

**“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO PARA EL
CONTROL Y TARIFACIÓN DEL SERVICIO DE TAXI A TRAVÉS
DE SENSORES INERCIALES.”**

TRABAJO DE TITULACIÓN

Tipo: DISPOSITIVOS TECNOLÓGICOS

Para optar al Grado Académico de:

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

AUTORA: REDROBÁN ARANDA LAURA VERÓNICA

TUTOR: Ing. ALTAMIRANO SANTILLÁN EDWIN VINICIO

Riobamba – Ecuador

2018

©2018, Laura Verónica Redrobán Aranda

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

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA ESCUELA DE INGENIERÍA
ELECTRÓNICA EN CONTROL Y REDES INDUSTRIALES**

El Tribunal del trabajo de titulación certifica que: El trabajo de titulación: “DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO PARA EL CONTROL Y TARIFACIÓN DEL SERVICIO DE TAXI ATRAVÈS DE SENSORES INERCIALES” de responsabilidad de la Señorita Laura Verónica Redrobán Aranda, ha sido minuciosamente revisado por los miembros del tribunal del trabajo de titulación, quedando autorizada su presentación.

NOMBRE

FIRMA

FECHA

Dr. Julio Santillán C.

**VICEDECANO DE LA FACULTAD DE
INFORMÁTICA Y ELECTRÓNICA**

Ing. Freddy Chávez V.

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

Ing. Edwin Altamirano S.

**DIRECTOR DEL TRABAJO DE
TITULACIÓN**

Ing. Pablo Lozada Y.

MIEMBRO DEL TRIBUNAL

“Yo, **LAURA VERÓNICA REDROBÁN ARANDA**, soy responsable de las ideas, doctrinas y resultados, expuestos en este trabajo de titulación y el patrimonio intelectual del Trabajo de Titulación pertenece a la **ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**”

Laura Redrobán

DEDICATORIA

Este trabajo de titulación se lo dedico a Dios, a mis padres Wilma y Mauricio quienes son el pilar fundamental en mi vida que con su apoyo, sus consejos y su amor, me ayudaron a cumplir está meta tan anhelada y ser la persona que hoy soy.

A mis hermanos Mayra, Christian y Germán quienes entre llantos, peleas y risas siempre estamos juntos sorteando las pruebas que la vida nos pones, son los mejores hermanos que la vida me pudo dar.

A mis abuelitos Anita, Laurita, Eduardo y Humberto quienes siempre han estado pendientes de mí en todo este largo proceso que algún día lo vi imposible y hoy es una realidad.

A mis tíos Anita y Cristian quienes de una u otra manera me apoyaron durante toda la carrera y mis prácticas pre profesionales.

A toda mi familia ya que cada uno de ellos a portaron con un granito de arena para que este sueño ahora sea un realidad. Todo esto es por ustedes y para ustedes.

Verónica

AGRADECIMIENTO

A Dios por haber bendecido con unos padres perseverantes quienes con su apoyo moral y económico fueron los que permitieron este logro, y unos hermanos que con su amor y paciencia eh sabido salir adelante.

A la persona que ha estado conmigo en las buenas y en las malas, con quien eh vivido momentos únicos, a pesar que no esté cerca en estos momentos siempre está pendiente de mis logros, que con su apoyo y consejos se convirtió en mi apoyo incondicional.

A mis amigos Denise, Jessica, Ana, Brenda y Byron con quienes recorrí toda esta aventura llamada Ingeniería Electrónica en Control y Redes Industriales.

A la Escuela Superior Politécnica de Chimborazo por haberme brindado la oportunidad de formarme profesionalmente es sus aulas, a los docentes quienes compartieron sus conocimientos de manera especial al Ingeniero Edwin Altamirano quien se convirtió en mi guía durante todo el tiempo que duro el desarrollo del proyecto de titulación.

Verónica.

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	x
ÍNDICE DE FIGURAS.....	xi
ÍNDICE DE GRÁFICOS.....	xiii
ÍNDICE DE ECUACIONES	xiv
ÍNDICE DE ANEXOS	xv
ÍNDICE DE ABREVIATURA	xvi
INTRODUCCIÓN	1
CAPITULO I	
1. MARCO TEÓRICO REFERENCIAL	4
1.1 Taxímetro.....	4
<i>1.1.1 Evolución del Taxímetro.....</i>	<i>4</i>
1.2. Tipos de taxímetros	5
<i>1.2.1 Taxímetros con Impresora</i>	<i>6</i>
<i>1.2.2 Taxímetros GPS</i>	<i>6</i>
1.3 Taxímetros Autorizados en el Ecuador.....	7
<i>1.3.1 Centrodyne S700</i>	<i>8</i>
<i>1.3.2 Sems Platinium.....</i>	<i>9</i>
<i>1.3.3 Intelligent Taximeter D10.....</i>	<i>10</i>
1.4 Funcionamiento del Taxímetro.....	10
<i>1.4.1 Principio de Operación</i>	<i>11</i>
<i>1.4.2 Funciones Principales del Taxímetro.....</i>	<i>11</i>
<i>1.4.3 Tarifas.....</i>	<i>12</i>
<i>1.4.3.1 Valor Inicial o Arranque.....</i>	<i>12</i>
<i>1.4.3.2 Carrera Mínima</i>	<i>12</i>
<i>1.4.3.3 Tarifa Diurna</i>	<i>12</i>
<i>1.4.3.4 Tarifa Nocturna.....</i>	<i>13</i>

1.5	Beneficio de la Electrónica en el área de tarificación.	13
1.6	Tarjetas de Desarrollo Electrónicas	13
1.6.1	Tarjeta de Desarrollo Electrónico Arduino	15
1.6.1.1	<i>Tipos de Tarjeta de Desarrollo Electrónico Arduino</i>	16
1.6.1.2	<i>Características Tarjetas de desarrollo Arduino</i>	16
1.6.1.3	<i>Software de Programación para Tarjetas de desarrollo Arduino</i>	17
1.7	Sensores Inerciales (IMU)	18
1.7.1	Sensor Inercial IMU GY- 521	19
1.7.2.1	<i>Características de módulo MPU – 6050</i>	20
1.7.2.2	<i>Estructura Interna del Módulo MPU - 6050</i>	21
1.7.2.3	<i>Ventajas del Módulo MPU - 6050</i>	22
1.8	Pantalla de Cristal Líquido (LCD)	22
1.8.1	Memorias del LCD	23
1.8.1.1	<i>Memoria DDRAM</i>	23
1.8.1.2	<i>Memoria CGROM y CGRAM</i>	24
1.8.2	Características y Pines de Conexión	24
CAPITULO II		
2.	MARCO METODOLÓGICO	26
2.1	Selección de Dispositivos Electrónicos	27
2.1.1	Módulo MPU 60-50	27
2.1.1.1	<i>Características del acelerómetro</i>	28
2.1.1.2	<i>Condiciones de funcionamiento del módulo MPU 60-50</i>	28
2.1.2	Placa de desarrollo electrónico Arduino UNO	28
2.1.2.1	<i>Características técnicas de la tarjeta de desarrollo electrónico Arduino Uno</i>	29
2.1.2.2	<i>Pines de la tarjeta de desarrollo electrónico Arduino UNO</i>	29
2.2	Calibración del módulo MPU 60-50	31
2.2.1	Comunicación entre módulo MPU 60-50 y placa Arduino Uno	31
2.2.2.1	<i>Bus de comunicación I2C</i>	31
2.2.3	Lectura de valores RAW del MPU 60-50	32

2.2.4	Calibración del módulo MPU 60-50	34
2.3.	Diseño del software del funcionamiento del prototipo de tarificación	36
2.3.1	Lectura de los datos del Módulo MPU 60-50	36
2.3.1.1	<i>Cambio de unidades de los valores del MPU 60-50</i>	37
2.3.1.2	<i>Filtro para el módulo MPU 60-50</i>	38
2.3.2	Implementación del Reloj	40
2.3.3	Cálculo de Distancia Recorrida	41
2.3.4	Cálculo de Tiempo de Espera	45
2.3.5	Tarifación	45
2.3.5.1	<i>Cálculo de la Tarifa Diurna</i>	46
2.3.5.2	<i>Cálculo de la Tarifa Nocturna</i>	47
2.4	Diseño de Hardware	48
2.4.1	Conexión de Dispositivos Electrónicos	48
2.4.1.1	<i>Conexión módulo MPU 60-50 con placa Arduino UNO</i>	48
2.4.1.2	<i>Conexión pantalla de cristal líquido LCD de 16x2</i>	49
2.4.1.3	<i>Conexión del botón de 4 pines</i>	51
2.4.2	Diseño y construcción de la carcasa del prototipo	52
2.4.2.1	<i>Diseño de la Carcasa</i>	52
2.4.2.2	<i>Corte y Armado de la Carcasa</i>	49
2.5	Implementación del Prototipo	54
2.5.1	Implementación del prototipo en el vehículo	54
2.6	Costos del Prototipo de Tarificación	56
CAPÍTULO III		
3	PRUEBAS Y RESULTADOS	58
3.1	Análisis Ensayo y Error	58
3.1.1	<i>Análisis de las tarifas de modalidad diurna</i>	59
3.1.2	<i>Análisis de las tarifas de modalidad nocturna</i>	61
3.2	Comparación de costos entre los taxímetros comerciales y el prototipo implementado	64

3.3 Comparación de requerimientos y parámetros de tarificación entre taxímetro comercial y prototipo diseñado.....	64
CONCLUSIONES.....	66
RECOMENDACIONES.....	68
BIBLIOGRAFÍA	
ANEXOS	

ÍNDICE DE TABLAS

Tabla 1 - 1:	Taxímetros Autorizados por el SRI y Homologados por la ANT	8
Tabla 2 – 1:	Cuadro Tarifario Diurno	13
Tabla 3 – 1:	Cuadro Tarifario Nocturno.....	13
Tabla 4 – 1:	Resumen de características técnicas de la placa Arduino Uno	20
Tabla 5 – 1:	Sensores inerciales existentes en el mercado	20
Tabla 6 – 1:	Características del MPU – 6050.....	20
Tabla 7 – 1:	Características del acelerómetro del MPU – 6050	21
Tabla 8 – 1:	Pines de conexión de la LCD	24
Tabla 1 – 2:	Condiciones de Funcionamiento del Módulo MPU 60.50.....	28
Tabla 2 – 2:	Pines especiales de la placa Arduino Uno.....	30
Tabla 3 – 2:	Tipos de direcciones I2C.....	31
Tabla 4 – 2:	Librerías para la comunicación entre el MPU 60-50 y la placa Arduino Uno	32
Tabla 5 – 2:	Tabla de cambio de unidades	37
Tabla 6 – 2:	Valores Máximos y Mínimos de lectura del MPU 60-50	37
Tabla 7 – 2:	Tarifas en función de la hora.....	40
Tabla 8 - 2:	Tarifa del Servicio de Taxi Modalidad Diurna	46
Tabla 9 - 2:	Tarifa del Servicio de Taxi Modalidad Nocturna.....	46
Tabla 10 - 2:	Pines de conexión entre Arduino y MPU 60-50	49
Tabla 11 - 2:	Pines de Conexión del LCD con Arduino UNO	50
Tabla 12 - 2:	Medidas de la carcasa diseñada.....	51
Tabla 13 - 2:	Costo de fabricación del prototipo	56
Tabla 14 - 2:	Costo porcentual de la construcción del prototipo	56
Tabla 1 - 3:	Tarifas marcadas por el taxímetro comercial y el prototipo en la modalidad diurna.....	58
Tabla 2 - 3:	Tarifas marcadas por el taxímetro comercial y el prototipo en la modalidad nocturna.....	59
Tabla 3 - 3:	Márgenes de errores obtenidos en las 15 carrera realizada modalidad diurna....	59
Tabla 4 - 3:	Márgenes de errores obtenidos en las 15 carrera realizada modalidad nocturna	60
Tabla 5 - 3:	Márgenes de errores obtenidos en cada carrera realizada	62
Tabla 6 - 3:	Costo de los taxímetros reconocidos por la ANT y el SRI.....	63
Tabla 7 - 3:	Parámetros y requisitos con los que cumple el prototipo	64

ÍNDICE DE FIGURAS

Figura 1 - 1:	Primer taxímetro comercial.....	5
Figura 2 - 1:	Taxímetro con impresora.....	6
Figura 3 - 1:	Aplicación Taxímetro con GPS.....	7
Figura 4 - 1:	Taxímetro CENTRODYNE S700.....	9
Figura 5 - 1:	Taxímetro SEMSA PLATINIUM.....	9
Figura 6 - 1:	Taxímetro INTELIGENTE TAXIMETER D10.....	10
Figura 7 - 1:	Transductor típico de medición de distancia.....	16
Figura 8 - 1:	Tarjeta de Desarrollo Electrónica Arduino.....	16
Figura 9 - 1:	Entorno de desarrollo Arduino.....	18
Figura 10 - 1:	Sensor Inercial MPU – 6050.....	19
Figura 11 – 1:	Funcionamiento del Acelerómetro.....	21
Figura 12 - 1:	Funcionamiento del Giroscopio.....	22
Figura 13 - 1:	Pantalla de Cristal Líquido de 2x16.....	23
Figura 15 - 1:	Memoria CGROM de LCD.....	24
Figura 1 – 2:	Tarjeta de desarrollo Arduino UNO.....	29
Figura 2 – 2:	Distribución de pines de la placa Arduino UNO.....	30
Figura 3 – 2:	Ejemplo comunicación I2C.....	32
Figura 4 – 2:	Distribución de los datos en brutos adquiridos con el módulo MPU60-50.....	34
Figura 5 – 2:	OFFSETS de inicialización del acelerómetro del MPU 60-50.....	35
Figura 6 – 2:	Valores calibrados del módulo MPU 60-50.....	35
Figura 7 – 2:	Diagrama de Flujo del Funcionamiento del Taxímetro.....	36
Figura 8 – 2:	Valores leídos por el MPU60-50, cambiados de unidades.....	38
Figura 9 – 2:	Valores y señal sin filtrar del módulo MPU 60 - 50.....	38
Figura 10 – 2:	Valores y señal filtrada del módulo MPU 60 - 50.....	38
Figura 11 - 2:	Diagrama de Flujo del Reloj.....	40
Figura 12 - 2:	Diagrama de Flujo para el Cálculo de la Distancia Recorrida.....	43
Figura 13 - 2:	Diagrama de Flujo para el Cálculo del Tiempo de Espera.....	45
Figura 14 - 2:	Cálculo de la Tarifa Diurna.....	46
Figura 15 - 2:	Cálculo de la Tarifa Nocturna.....	47
Figura 16 – 2:	Módulo MPU 60-50 soldado en posición horizontal.....	48
Figura 17 - 2:	Diagrama de conexión MPU60-50 y Arduino UNO.....	49

Figura 18 - 2:	Diagrama de conexión de LCD de 16x2 con Arduino UNO	50
Figura 19 - 2:	Conexión del botón con Arduino UNO.....	50
Figura 20 - 2:	Diseño esquemático del circuito completo.....	50
Figura 21 - 2:	Carcasa diseñada en el software Solidwork	51
Figura 22 - 2:	Corte de la Carcasa.....	52
Figura 23 - 2:	Carcasa para el prototipo.....	52
Figura 24 - 2:	Prototipo de control y tarifación.....	54
Figura 25 - 2:	Instalación del módulo MPU60-50	55
Figura 26 - 2:	Prototipo instalado en el interior de vehículo.....	55

ÍNDICE DE GRÁFICOS

Gráfico 1 - 2:	Etapas del diseño e implementación del prototipo	26
Gráfico 2 – 2:	Estructura del Prototipo de Control y Tarifación	27
Gráfico 1 - 3:	Porcentaje de error de la tarifa marcada en modalidad diurna1	60
Gráfico 2 - 3:	Porcentaje de error de la tarifa marcada en modalidad nocturna	61
Gráfico 3 – 3:	Comparación de costos entre taxímetros comerciales y el prototipo realizado...	64

ÍNDICE DE ECUACIONES

Ecuación 1 - 2:	Ecuación para la conversión de unidades de [g] a [m/s ²].....	38
Ecuación 2 – 2:	Filtro Exponencial EMA con Filtro Pasa Bajo y Pasa Alto para Arduino ..	39
Ecuación 3 - 2:	Cálculo de Distancia con Velocidad Media	41
Ecuación 4 - 2:	Cálculo de Distancia conociendo la Aceleración	41
Ecuación 5 - 2:	Cálculo de Aceleración	43
Ecuación 6 - 2:	Cálculo del Tiempo de la Aceleración Sensada por el MPU 60-50.....	44
Ecuación 7 - 2:	Cálculo de la Distancia Recorrida con la Aceleración Sensada.....	44
Ecuación 1 – 3:	Cálculo del margen de error	59
Ecuación 2 – 3:	Media Aritmética	62

ÍNDICE DE ANEXOS

- Anexo A:** DataSheet del MPU60-50 ó IMU GY-521
- Anexo B:** Programa de Calibración del Módulo MPU 60-50
- Anexo C:** Programa del Prototipo de tarifación
- Anexo D:** Partes del Prototipo de Control y Tarifación
- Anexo E:** Prueba de funcionamiento del prototipo de control y tarifación
- Anexo F:** Recibos de las carreras de prueba realizadas

ÍNDICE DE ABREVIATURA

ANT	Agencia Nacional de Tránsito
INEN	Instituto Ecuatoriano de Normalización
IMU	Unidad de medición inercial
MPU	Unidad de múltiples procesos
LCD	Pantalla de cristal líquido
RS	Sección de registro de datos
R/W	Lectura / Escritura
F	Habilitación
D0	Bit menos significativo
X	Distancia recorrida
Vmed	Velocidad Media
t	Tiempo de aceleración
as	Aceleración sensada
deg/s	Grado por segundo
RPM	Revoluciones por minuto
Ms	Milisegundos
tsa	Tiempo de aceleración sensada
VDC	Voltaje de corriente continua
USB	Universal serial plus
MHZ	Mega Hertz
a	Aceleración
mA	Miliamperios
KΩ	Kilohmios
Hz	Hertz
F	Frecuencia
T	Tiempo
Pull up	Circuito en reposo

RESUMEN

Se desarrolló un prototipo para el control y tarifación del servicio de taxi utilizando sensores inerciales. Se analizaron las diferentes tecnologías para el diseño e implementación del prototipo usando el método heurístico. El prototipo se lo implemento en cinco etapas, la primera se seleccionaron los dispositivos que cumplan las especificaciones de funcionalidad como es el módulo MPU 60 – 50 y la placa arduino uno, en la segunda se calibró del módulo MPU 60 – 50, donde se establecen los valores iniciales de funcionamiento a partir de la ubicación donde se va a utilizar, en la tercera se diseñó el software para la interacción entre la placa arduino y el módulo. El diseño del programa dispone de una función de reloj de 24h que permite verificar el tipo de modalidad que se va a tarifar, el cálculo de la distancia recorrida por el vehículo con los datos de la aceleración proporcionadas por el módulo y el tiempo de aceleración del vehículo, para este cálculo se utilizó el método usado en la fórmula uno. El tiempo de espera que se produce mientras el vehículo está parado, es cuando el sensor no envía datos por lo que se inicializa un contador para el tiempo de espera y tiene un costo monetario extra. La cuarta etapa es el diseño de la carcasa en solidworks, una vez armado el prototipo se instaló el MPU 60 -50 en la parte baja del vehículo y el tarifado en la parte interna sin obstaculizar la visibilidad del conductor. Las pruebas y resultados analizados permitieron comprobar la funcionalidad y eficiencia del prototipo con un margen de error del 0,86% en la tarifa marcada en comparación al dispositivo comercial. Para que el prototipo funcione correctamente se debe tomar en cuenta que no debe exceder la velocidad de 50km/h.

PALABRAS CLAVES: <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <INGENIERÍA Y TECNOLOGÍA ELECTRÓNICA>, <SERVICIO DE TRANSPORTE>, <TARIFACIÓN>, <DISPOSITIVO DE TARIFACIÓN>, <ARDUINO (SOFTWARE-HARDWARE)>, <SENSOR INERCIAL (IMU)>, <SIMULACIÓN>.

SUMMARY

A prototype was developed for the control and pricing of the taxi service using inertial sensors. It was analyzed the different technologies for the desing and implementation of the prototype using the Heuristic method. The prototype was implemented in five stages, the first was selected devices that meet the functionality specifications such as the MPU 60 – 50 module and the Arduino UNO board, the second was calibrated from the MPU 60 – 50 module, where values are set in initials of operation from the location where it will be used, in the third the software was designed for the interaction between the Arduino board and the module. The design of the program has a 24h clock function that allows to verify the type of modality that is going to rate, the calculation of the distance traveled by the vehicle with the data of the acceleration provided by the module and the acceleration time of the vehicle, the method used in Formula One was used for this calculation. The waiting time that accurs while the vehicle is stopped, it is when the sensor does not send data so it initializes a times for the timeout and has an extra monetary cost. The fourth stage is the desing of the housing in SolidWorks, once the prototype was assembled the MPU 60 – 50 were installed in the lower part of the vehicle and the rate on the inside without hindering the visibility of the driver. The tests and results analyzed allowed checking the functionality and efficiency of the prototype wiht an error margin of 1.63% in the rate marked compared to the comercial device. For the prototype to work properly, it should be taken into account that it should not exceed the speed of 50km/h.

KEY WORDS: <TECHNOLOGY AND SCIENCES ENGINEER>, < ENGINEERING AND ELECTRONIC TECHNOLOGY>, < TRANSPORT SERVICE>, < PRICING>, <CHARGING DEVICE>, < ARDUINO (SOFTWARE – HARDWARE)>, < (IMU) INERTIAL SENSOR>, < SIMULATION>.

INTRODUCCIÓN

ANTECEDENTES

Hoy en día con la evolución de la tecnología se ha dado paso a la creación de varias herramientas que dan solución a necesidades de la vida diaria. El taxímetro es una de estas herramientas, que permiten el cálculo de tarifas de servicios de taxi, con el cual se realiza el cobro justo por este servicio, logrando que el usuario como el prestador del mismo esté de acuerdo con la tarifa establecida.

Debido a que los taxímetros que están siendo utilizados en la ciudad de Riobamba son propensos a sufrir alguna alteración, ya sea directamente en la programación del dispositivo, incorporando un pulsador para el aumento del número de revoluciones, o daño del sensor del cual se obtiene los datos para este cálculo.

FORMULACIÓN DEL PROBLEMA

Según el análisis las diferentes necesidades que presentan los señores prestadores del servicio de taxi se encuentran la exactitud del cobro de la tarifa del servicio, ya que este dispositivo es muy propenso a la alteración de los datos que se obtienen, es decir el sobre tarifando del valor de la carrera provocando inconformidad de los usuarios de este servicio.

Si en determinado momento el sensor que genera el dato necesario para el cálculo de la distancia para determinar la tarifa sufre algún daño este dejaría de funcionar, lo que afectaría directamente en la economía del prestador del servicio por la falta de la tarifa real de la distancia recorrida.

Es por tal motivo que se plantea este proyecto como una alternativa de solución para la tarifación del servicio de taxi de manera exacta.

SISTEMATIZACIÓN DEL PROBLEMA

¿Qué tipos de taxímetros existen en el Ecuador?

¿El prototipo de taxímetro cumple con los requerimientos necesarios para dar solución al problema planteado?

¿Qué beneficios prestará el taxímetro al usuario y dueño del vehículo?

JUSTIFICACIÓN TEÓRICA

Por medio de este proyecto de investigación se pretende diseñar e implementar un nuevo taxímetro que presente los mismos beneficios y a la vez cumpla con todas las características de los comerciales que se usan actualmente. La tarifa será calculada de manera justa de acuerdo a la distancia recorrida tomando en cuenta todos los parámetros de monto de arranque, costo de kilómetro recorrido y costo del minuto de espera que platea la Agencia Nacional de Transito (ANT) y el Servicio de Rentas Internas (SRI). Evitando así alteraciones o manipulación de los datos necesarios para este cálculo.

JUSTIFICACIÓN APLICATIVA

Actualmente los taxímetros existentes calculan la distancia recorrida de manera indirecta mediante un sensor que va ubicado en el neumático del vehículo el cuál cuenta el número de revoluciones que realiza durante el recorrido, con el cual se determina la distancia recorrida en función al diámetro del neumático y posteriormente se determina la tarifa final.

A la vez se pretende reducir el costo al momento de la implementación del prototipo, para que esté al alcance de todos los prestadores del servicio de taxi.

Este proyecto se implementará en un vehículo de servicio público, se utilizará una placa de desarrollo electrónica y un sensor inercial (IMU) para la obtención de los datos necesarios para el cálculo de la distancia recorrida y el costo de la carrera, la misma que será visualizada en un LCD.

OBJETIVOS

OBJETIVO GENERAL

Diseñar e implementar un prototipo para el control y tarificación del servicio de taxi a través de sensores inerciales.

OBJETIVOS ESPECIFICOS

- Investigar los tipos de taxímetros existentes en el mercado.
- Estudiar las diferentes tecnologías implementadas en los taxímetros.
- Seleccionar los dispositivos electrónicos para la construcción del prototipo.
- Implementar el prototipo con componentes hardware y software.
- Realizar las pruebas y analizar los resultados del funcionamiento del prototipo.

CAPITULO I

1. MARCO TEÓRICO REFERENCIAL

En el presente capítulo se detalla la manera en que la electrónica aportado en el desarrollo del servicio de transporte para el cálculo del costo exacto del mismo, por medio de la implementación de un prototipo para el cobro y tarifación del servicio de taxi con sensores inerciales.

En el año 2013 en la Facultad de Mecánica de la Escuela Superior Politécnica de Chimborazo el Sr. Edgar García realizó un taxímetro como proyecto de titulación, el cual fue implementado en una bicicleta, utilizando un micro controlador en este caso un PIC.

NITAX es un taxímetro inteligente que calcula las tarifas en función del tiempo y distancia a partir de una medición física real obtenida de sensores magnéticos del taxi tiene un diseño electrónico diferente de manera que no puede ser manipulado en algún taller, fue creado por la empresa Mexicana Nitax. (NITAX, 2017, <http://www.nitax.net/taxiacutemetros.html>)

Actualmente existen aplicaciones móviles las cuales brindan realiza la misma función que un taxímetro físico. Easytaxi es uno de ellos que realiza la tarifación del servicio mediante GPS luego de seleccionar la ciudad donde se encuentre el usuario.

1.1 Taxímetro

El término Taxímetro proviene de las palabras griegas “taxi” que significa tasa y “metro” que significa medir; es decir mide la tasa o costo que tiene el servicio. Es un dispositivo de medida parecido a un odómetro, pero en este caso mide la tarifa a cobrar por el servicio prestado en base a la distancia recorrida y al tiempo transcurrido. (TORRES, 2013, <http://www.segurosautos.com.mx/sabias-que/historia-taximetro.html>. 1.)

1.1.1 Evolución del taxímetro.

El primer taxímetro se utilizó en la antigua Roma, el cual funcionaba por medio de un mecanismo formado por un eje de una carreta que iba soltando pequeñas bolas de madera o

piedra en un saco. Al terminar el trayecto el usuario cancelaba el valor del servicio en función al número de bolas liberadas en el saco. Fue creado por el ingeniero inglés Wilhelm Bruhn en el año de 1891.

El primer taxi equipado con este dispositivo fue el Daimler Victoria fabricado por Gottlieb en 1897. A partir de los años 2000 se utilizó el mismo principio de funcionamiento, solo que se aumentó una variable más que es el tiempo, ya que no se calcula el valor a pagar solo en función de los kilómetros recorridos sino también en función del tiempo transcurrido. (TORRES, 2013, <http://www.segurosautos.com.mx/sabias-que/historia-taximetro.html>. 1.).

El la Figura 1 - 1 se muestra el primer taxímetro comercial.



Figura 1 - 1: Primer taxímetro comercial.

Fuente: <https://img.webme.com/pic/t/taximetrosgama/taximetro.jpg>

1.2. Tipos de taxímetros

Los tipos de taxímetros más comerciales son los de impresora y GPS, los mismos que se detallan a continuación.

1.2.1 Taxímetros con Impresora

Los taxímetros con el paso del tiempo se fueron mejorando de acuerdo a las necesidades de los usuarios del servicio como de los prestadores del mismo. Entre estas innovaciones se tiene la incorporación de una pequeña impresora térmica que está colocada junto al taxímetro, la cual permite la impresión de pequeños recibos con datos puntuales del servicio brindado como es la hora, distancia recorrida en km, datos del vehículo y total a pagar, lo que permite llevar una contabilidad de lo trabajado en el día.

En la figura 2 - 1 se muestra el taxímetro con impresora incluida.



Figura 2 - 1: Taxímetro con impresora

Fuente: <http://p.globalsources.com/IMAGES/PDT/S1154571834/Taximeter.jpg>

1.2.2 Taxímetros con GPS

Los taxímetros siguen innovando, en este caso con la implementación del Sistema de Posicionamiento Global (GPS), el cual permite hacer el cálculo de la tarificación según la distancia que separa al punto de partida del punto de llegada por medio de una aplicación móvil.

En la figura 3 – 1 se muestra un taxímetro con GPS.



Figura 3 - 1: Taxímetro con GPS

Fuente: <http://blog.valejet.com/nao-fique-perdido-saiba-como-funciona-um-gps/>

1.3 Taxímetros autorizados en el Ecuador

En el Ecuador existen una sobre oferta de vehículos que brindan el servicio de taxi, para regularizar el cobro de este servicio la Agencia Nacional de Transito del Ecuador dispuso que desde el 26 de Diciembre del 2013 se inicie una campaña de uso del taxímetro obligatorio por parte de los señores choferes como de los usuarios de este servicio.

La ANT dispuso que no se pueden instalar cualquier tipo de taxímetro, sino solo los que cumplan con los requerimientos de homologación los cuales se encuentran en la INEN 2336 Taxímetros. Requisitos metrológicos y técnicos, estos requerimientos son:

- Incluyan un reloj.
- Un contador de eventos.
- Interfaz de usuario.
- Los parámetros deben ser programables.
- Número de identificación del taxi.
- Sensor que proporcione datos para el cálculo de la distancia recorrida.
- Programa legalmente relevante. (INEN2336, 2013)

Los taxímetros que cumplen con estos requisitos se los muestra en la tabla 1-1.

Tabla 1 - 1: Taxímetros Autorizados por el SRI y Homologados por la ANT

N°	Marca	Modelo
1	CENTRODYNE	S700
2	SEMSA	PLATINUM
3	LAKEDRIVER	SLIM 800
4	INTELLIGENT TAXIMETER	D10
5	TAXSYM	M-09
6	DIGITAX 550	M-07
7	DIGITAX 550	M11
8	FUL MAR	TANGO XP
9	STALIN_TAX	ST-11
10	OPTRONIC	TX-10
11	VIRTUALTEC	VIRLOC10
12	ARIEL TAX	MILENIO

Fuente: <http://www.sri.gob.ec/DocumentosAlfrescoPortlet/descargar/423a4d99-9f48-4838-b2>

Realizado por: REDROBÁN, Laura, 2017

En el Ecuador existen aplicaciones móviles disponibles las cuales son utilizadas comúnmente para solicitar un taxi. Uber es una de ellas, que además realiza el cálculo del costo del servicio mediante GPS, y el cobro del mismo mediante una cuenta vinculada con esta aplicación, y en ciertas ciudades permite el cobro en efectivo. (UBER, 2016, <https://www.uber.com/es-EC/>)

Cabify es otra de las aplicaciones móviles permitidas en el país la cual tiene las mismas funciones que Uber, pero en este caso el cobro del servicio se lo realiza mediante tarjeta de crédito o débito introduciendo los datos de la tarjeta antes de iniciar la carrera, ya que al finalizar se realizará de manera automática el débito del monto marcado. (CABYFY, 2017, <https://www.autocasion.com/actualidad/reportajes/como-funciona-cabify-el-chofer-a-domicilio>)

1.3.1 Centrodyne S700

El taxímetro, "S700", perteneciente a la marca CENTRODYNE es uno de los autorizados en el SRI para su utilización en el cálculo y cobro del servicio de taxi en el país, como se muestra en la figura 4 - 1. Sus principales características son:

- Cuenta con una interfaz de 6 botones pulsadores
- Una pantalla aún más intensa

- Intensidad del brillo de la pantalla regulable día / noche (CENTRODYNE, 2013, <http://www.centrodyne.com/es/products/taximeters/s700.1.>)



Figura 4 - 1: Taxímetro CENTRODYNE S700
Fuente: <http://www.centrodyne.com/images/stories/S700.jpg>

1.3.2 Semsá Platinum

El taxímetro SEMSA PLATINIUM es uno de los que se puede instalar en las unidades de taxi del país. Tiene algunas características relevantes que son:

- Tamaño 157x55x42 mm
- Tarifa diurna y nocturna
- Resistencia mínima a los cambios de temperatura
- Consumo de 500 mA (AGENCIA NACIONAL DE TRANSITO, 2013, <http://www.ant.gob.ec/index.php/descargable/file/415-semsa-platinum.1.>)



Figura 5 - 1: Taxímetro SEMSA PLATINIUM
Fuente: <http://semsataxímetros.com/images/img0022.png>

1.3.3 Intelligent Taximeter D10

Este taxímetro viene con la comunicación integrada a una pequeña impresora, como se muestra en la figura 6 - 1. Sus características son:

- Voltaje de entrada 9 – 16 V
- Consumo nominal 200 mA
- Reloj y fecha internos activados
- Almacena 12 registros por mes (AGENCIA NACIONAL DE TRANSITO, 2013, <http://www.ant.gob.ec/index.php/descargables/file/413-intelligent-taximeter-d10>)



Figura 6 - 1: Taxímetro INTELIGENTE TAXIMETER D10

Fuente: <http://www.ecuavisa.com/sites/default/files/fotos/2014/02/11/taximetros1.jpg>

1.4 Funcionamiento del taxímetro.

El taxímetro ha llegado a dar solución a la problemática del cobro excesivo de tarifas del servicio de taxi. Este dispositivo tiene un funcionamiento el cual pretende tener un mínimo margen de error, ya que el margen de error permitido para los taxímetros comerciales impuesto por la Organización Internacional de Metrología Legal es del 2% del valor real, este valor aumento con respecto al 1% que se establecía en la normativa anterior. (Organización Internacional de Metrología Legal, 2009, <https://www.facua.org/es/noticia.php?Id=4375>)

Para el cálculo adecuado se toma en cuenta dos variables la duración y la distancia del recorrido. El taxímetro tiene dos funcionamientos especiales cuando el vehículo está en movimiento calcula la distancia recorrida y cuando está parado el tiempo de espera ya sea este por el tráfico o semáforos. (TAXIMETER, 2016, <https://examentaxi.com/funcionamiento-de-un-taximetro.1.>)

1.4.1 Principio de operación

En el taxi se instala un transductor el cual mide la distancia y la convierte en pulsos eléctricos los cuales son enviados al taxímetro. Con la señal medida del tiempo junto con la de distancia se calcula: el costo, los totales, y se despliega los posibles registros del resultado debidos al viaje del taxi, basados en tarifas especificadas y/o la distancia y/o la duración del viaje. (INEN2336, 2013)

Un transductor analógico envía impulsos eléctricos con una frecuencia proporcional a la velocidad del taxímetro, el cual es controlado por el programa que calcula la velocidad y transmite el valor de la cantidad en un datagrama a través de un bus de datos para el taxímetro. El transductor que utilizan los taxímetros comerciales se visualiza en la figura 7 – 1.

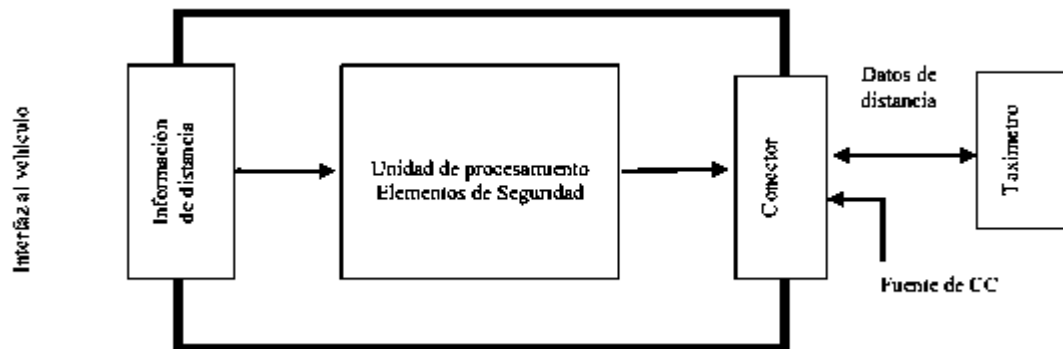


Figura 7 – 1: Transductor típico de medición de distancia.
Fuente: INEN2336

1.4.2 Funciones principales del taxímetro

Las principales funciones del taxímetro son:

a) Etapa “Libre”

Este proceso se da cuando el taxímetro no está realizando ningún cálculo, porque en el interior del vehículo no existe ningún usuario que esté haciendo uso del servicio.

b) Etapa “Ocupado”

El funcionamiento ocupado empieza cuando un usuario hace uso del servicio, se activa el taxímetro y empieza a generar los datos de tiempo y distancia para el cálculo de la tarifa final a pagar, tomando en cuenta el costo inicial y final del servicio.

c) Etapa “Pagar”

En esta etapa el taxímetro determina y visualiza el valor a pagar tomando en cuenta la tarifación establecida por el GAD Municipal de Riobamba.

1.4.3 Tarifas

La regulación de la tarifa del servicio de taxi es competencia del Gobierno Autónomo Descentralizado Municipal de Riobamba, luego de análisis de factores socio económico de la ciudad, técnicos y políticas establecidas por el organismo competente. La coordinación con la autoridad encargada y el control operativo de las vías para el correcto cumplimiento de la disposición. (GAD MUNICIPAL DE RIOBAMBA, 2016)

Dentro de las regulaciones por parte del GAD Municipal de Riobamba se tiene un cuadro tarifario del costo de arranque, kilómetros recorridos, minuto de espera, carrera mínima, tarifa diurna y nocturna; las mismas que se detallan a continuación.

1.4.3.1 Valor Inicial o Arranque

Valor monetario en cual arranca el cálculo del costo final del servicio prestado, está varia en el día y la noche.

1.4.3.2 Carrera Mínima

Es el costo mínimo que el usuario de este servicio debe cancelar por transportarse de un lugar a otro.

1.4.3.3 Tarifa Diurna

El taxímetro tiene incorporado un reloj el cual permitirá identificar la tarifa que se va a cobrar en este caso la tarifa diurna entra en vigencia desde las 06:00 am hasta las 06:59 pm, donde se registrará los rubros de la tabla 2 -1.

Tabla 2 – 1: Cuadro Tarifario Diurno

Diurno			
Arrancada	Km recorrido	Minuto de espera	Carrera mínima
USD 0,40	USD 0,32	USD 0,06	USD 1,20

Fuente: <http://www.gadmriobamba.gob.ec/index.php/transparencia/lotaip2016/ordenanzas-2016?download=658:ordenanza-009-2016->

Realizado por: REDROBÁN, Laura, 2017

1.4.3.4 Tarifa Nocturna

La tarifa nocturna empieza a correr desde las 07:00 pm hasta las 05:59 am, los costos están establecidos en la tabla 3 – 1.

Tabla 3 – 1: Cuadro Tarifario Nocturno

Nocturno			
Arrancada	Km recorrido	Minuto de espera	Carrera mínima
USD 0,45	USD 0,35	USD 0,06	USD 1,30

Fuente: <http://www.gadmriobamba.gob.ec/index.php/transparencia/lotaip2016/ordenanzas-2016?download=658:ordenanza-009-2016-tarifas-taxis>

Realizado por: REDROBÁN, Laura, 2017

1.5 Beneficio de la electrónica en el área de tarifación.

La electrónica ha sido el punto clave para el desarrollo de diferentes áreas, dando solución a varias problemáticas de la vida cotidiana, entre ellas está la tarifación del servicio de transporte público enfocándose en los taxis. El mayor problema presentado en este servicio fue el cobro irregular por el traslado de un lugar a otro a los usuarios, produciendo cobros excesivos. Gracias al aporte de la electrónica se desarrolló un dispositivo que permitió la tarifación del servicio en función de la distancia recorrida y el tiempo de duración del mismo, lo que ayudó a la regulación del costo de este servicio.

1.6 Tarjetas de desarrollo electrónicas

Las tarjetas de desarrollo electrónicas llegaron a dar un giro en la electrónica, ya que estas fueron creadas para el desarrollo de prototipos y proyectos que necesitan mayor rapidez al momento de su ejecución. Estas tarjetas están compuestas por varios elementos entre los cuales se encuentran los microprocesadores y microcontroladores en una sola placa.

Para la selección de una tarjeta de desarrollo electrónica, el procesador que este posee no lo es todo, ya que en muchos de los proyectos hay que tomar en cuenta la simplicidad, el precio, la velocidad de entrega y facilidad en la programación. Para el desarrollo de este proyecto la principal característica para la selección de la tarjeta es que debe tener comunicación I2C y aceptar datos digitales, las tarjetas que cumplen son:

a) Arduino

Esta tarjeta tiene un procesador con ATmega16U2, también posee periféricos y protocolos de comunicación UART, I2C, SPI, GPIO, PWM, ADC y comparador, interrupciones. Tiene una gran gama de dispositivos compatibles con esta tarjeta. Su costo es bajo y este varía de acuerdo del modelo que se elija.

b) Raspberry pi

Esta tarjeta tiene una CPU ARM® Cortex™ A53 de cuatro núcleos de 1,2 GHz y 64 bits, incluye una tarjeta LAN inalámbrica 802.11n integrada y capacidad Bluetooth 4.1, consta con 4 puertos USB los cuales son para conectar teclado y mouse, conector HMI. Para trabajar con esta tarjeta se requiere un almacenamiento extra, es decir una tarjeta microSD y posee periféricos y protocolos de comunicación UART, I2C, SPI, GPIO, PWM, USB, Ethernet, WiFi, HDMI. Su costo es de \$69.

c) BeagleBone Black

Esta tarjeta posee un procesador AM3358 ARM Cortex™ A8 de 1 GHz que proporciona potencia de procesamiento para la mayoría de las necesidades de computación, al igual que 512 MB de DDR3L DRAM y 4 GB de memoria flash integrada. Es usada en el desarrollo de interfaces de expansión. Incluye Ethernet, host USB y OTG, ranura de tarjeta microSD, puertos en serie, JTAG, microHDMI integrado, eMMC, ADC, I2C, SPI, PWM y LCD. Su costo es de \$ 89.00.

d) Joule 570X de Intel

Es la placa de desarrollo más poderosa hasta la fecha. La CPU Atom T5700 de cuatro núcleos tiene un reloj base de 1,7 GHz, pero puede usar el modo turbo y alcanzar los 2,4 GHz. Se puede combinar con 4 GB de memoria LPDDR4 y 16 GB de memoria flash NAND eMMC, y además admite la reproducción y captura de video 4K. Si bien la placa Joule puede ofrecer rendimiento para cualquier diseño, fue creada para ocupar un nicho específico en el mercado de placas de desarrollo ya que está dirigida a desarrolladores avanzados de aplicaciones de visión artificial y video de mayor rendimiento. Posee comunicación 2 USB3, USB2, PCIe, x4 DSI, x4 CSI, SPI, 3 I2C, I2S, DMIC, t2 UART, HDMI. Su costo es de \$ 512.00. (HETPRO, 2018, <https://hetpro-store.com/TUTORIALES/arduino-vs-raspberry-pi/>)

Luego de analizar cada una de las tarjetas de desarrollo electrónico mencionadas anteriormente, todas son aptas para ser usadas en este proyecto, ya que posee comunicación I2C, conexión a dispositivos de visualización, lectura de datos digitales, pines analógicos y digitales.

Se escoge la placa arduino por su bajo precio, ya que uno de los principales objetivos es la implementación del prototipo a bajo costo. La placa arduino ejecuta 2 millones de líneas de código cada segundo y su velocidad de transferencia de datos es variable de acuerdo a las necesidades de la utilidad que se lo dé.

1.6.1 Tarjeta de desarrollo electrónico Arduino

La tarjeta Arduino es una placa de hardware libre que su componente principal es un microcontrolador reprogramable, el cual puede funcionar de manera autónoma o controlada por medio de un software de ejecución. Consta de pines hembra que son entradas y salidas, análogas y digitales.

Arduino fue diseñado con la intención de facilitar la creación de nuevos prototipos, objetos, entornos de interacción con personas, ya que la programación se lo realiza en un software libre donde la programación es sencilla. Las características de cada arduino dependen de la utilidad que se lo vaya a dar. (TORRENTE, 2013, p.64)

En la figura 8 - 1 se muestra una tarjeta de desarrollo electrónica Arduino.



Figura 8 - 1: Tarjeta de Desarrollo Electrónica Arduino

Fuente: <https://blog.arduino.cc/wp-content/uploads/2011/01/10356-01b.jpg>

1.6.1.1 Tipos de Tarjeta de Desarrollo Electrónico Arduino

Actualmente existen varias tarjetas arduino las cuales se diferencian entre sí por el número de E/S, tipo de memorias que se van utilizar y la aplicación que se le va a dar. Entre las más usadas se encuentran:

- Arduino Mega 2560
- Arduino ADK
- Arduino Ethernet
- Arduino Fio
- Arduino Nano
- Arduino Pro Mini
- Arduino Uno
- Arduino Leonardo

1.6.1.2 Características de las tarjetas de desarrollo Arduino

Las características generales de la placa arduino son:

- Simplifica el proceso de trabajar con microcontroladores.
- Su costo es bajo y accesible para los usuarios en comparación con otras placas.
- Es compatibles con los sistemas operativos de Windows y Linux.

- La programación es simple y fácil de usar, ya que está basado en el entorno de programación processing.
- Es de licencia libre, por lo que existen muchos códigos abiertos. (SILVA, 2016, <https://edgardosilvi.wordpress.com/2016/02/29/acamica-ventajas-y-desventajas-de-arduino/>)

Las características técnicas que posee la placa arduino Uno se especifican en la tabla 4 - 1.

Tabla 4 – 1: Resumen de características técnicas de la placa Arduino Uno

Microcontrolador	Atmega 328
Voltaje de operación	5V
Voltaje de entrada (Recomendado)	7 – 12V
Voltaje de entrada (Límite)	6 – 20V
Pines para entrada – salida digital	14 (6 pueden usarse como salida PWM)
Pines de entrada analógicas	6
Corriente continua por pin IO	40 Ma
Corriente continua en el pin 3.3V	50 Ma
Memoria Flash	32 KB (0,5 KB ocupados por el bootloader)
SRAM	2 KB
EEPROM	1 KB
Frecuencia de reloj	16 MHz

Fuente: <http://www3.gobiernodecanarias.org/medusa/ecoblog/ralvgon/files/2013/05/Caracter%C3%ADsticas-Arduino.pdf>

Realizado por: REDROBÁN, Laura, 2018

1.6.1.3 Software de Programación para Tarjetas de desarrollo Arduino

Un programa es el conjunto de varias instrucciones en secuencia que son ejecutadas por una computadora o microcontrolador. El software que se utiliza para el desarrollo, compilación y carga de los programas en el arduino es el IDE (Integrated Development Environment), el cual está disponible en diferentes versiones y varios sistemas operativos. El lenguaje de programación está basado en el lenguaje C.

En la figura 9 - 1 se muestra el entorno de desarrollo Arduino.

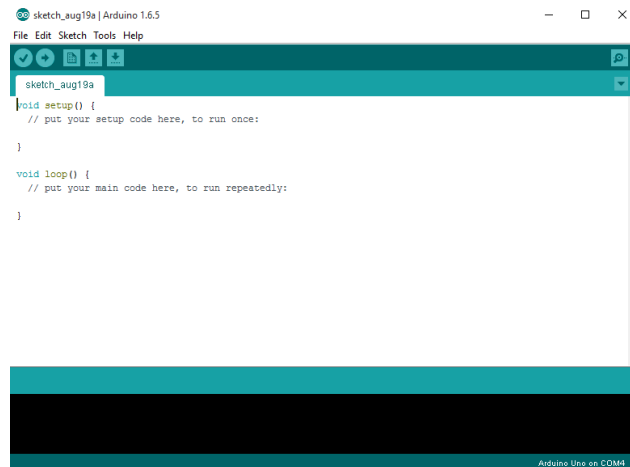


Figura 9 - 1: Entorno de desarrollo Arduino

Fuente: http://k2.arduino.vn/img/2016/08/15/0/3042_882450-1471265969-0-arduino-ide-screenshot.png

1.7 Sensores Inerciales (IMU)

Estos dispositivos surgieron en los años 20's en diferentes aplicaciones como la navegación, control de aeronaves y en el área militar. Los primeros IMU eran grandes, pesados y consumían grandes cantidades de energía y elevados costos, por lo que no eran utilizados muy a menudo, existía un campo extenso en los cuales se podía utilizar, pero se requería que estos fueran de menos peso y tamaño.

Una unidad de medida inercial o IMU detecta la aceleración mediante uno o varios acelerómetros, mide variaciones de velocidad perpendicularmente a la superficie del sensor, proporcionando una tensión proporcional a la aceleración que sufre el sensor. A la vez detecta el cambio de orientación usando uno o más giroscopios, estos miden las variaciones de ángulo en un eje específico, entregando una tensión proporcional al ángulo de giro que sufre el sensor.

(MARTINEZ, 2013,

https://www.researchgate.net/profile/Rigoberto_Martinez_Mendez/publication/263198667_Uso_de_sensores_inerciales_en_la_medicion_y_evaluacion_de_movimiento_humano_para_aplicaciones_en_la_salud/links/5783bfdd08ae37d3af6c0210/Uso-de-sensores-inerciales-en-la-)

En el mercado existe una gran cantidad de sensores inerciales los cuales varían en el número de acelerómetros giroscopios y el tercer sensor que varía de acuerdo al IMU seleccionado. Los más conocidos se muestran en la tabla 5 - 1.

Tabla 5 – 1: Sensores Inerciales existentes en el mercado.

Sensor	Estructura Interna
MPU 60 - 50	Un acelerómetro de 3 ejes, un giroscopio de 3 ejes y un sensor de temperatura.
MPU 91 – 50	Incorpora una MPU 60 – 50 y un magnetómetro AK8975.
MPU 92 - 50	Incorpora un MPU 65 – 00, un acelerómetro de 3 ejes, un giroscopio de 3 ejes y un magnetómetro AK8963.

Fuente: <https://www.invensense.com/products/motion-tracking/6-axis/mpu-6500/>

Realizado por: REDROBÁN, Laura, 2018

Para el desarrollo de este prototipo se elige el MPU 60 – 50 ya que tiene incorporado un acelerómetro de 3 ejes para adquirir los datos de la aceleración lineal, un giroscopio, un sensor de temperatura, y posee comunicación directa con la placa arduino por medio del bus de comunicación I2C.

1.7.1 Sensor Inercial IMU GY- 521 o MPU 60-50

Entre los sensores inerciales está el IMU GY – 521 o MPU 6050 es uno de los módulos compatibles con Arduino con el cual se puede determinar el movimiento, orientación y posición espacial con referente al plano.

El módulo MPU 60 – 50 se lo muestra en la figura 10 - 1.



Figura 10 - 1: Sensor Inercial MPU – 6050

Fuente: <http://playground.arduino.cc/uploads/Main/mpu-6050.jpg>

1.7.1.1 Características de módulo MPU – 6050

El MPU 60 – 50 es uno de los sensores más exactos existentes en el mercado, ya que tiene 16 bits de resolución, lo que permite dividir el rango dinámico en 65536 fracciones, lo cual aplica para cada eje x, y, z tanto para la aceleración como para la velocidad angular. Este sensor es ideal para el diseño de controles en la robótica, medición de vibraciones, también se lo puede usar como sensor de velocidad y distancia, entre otras aplicaciones. Este sensor tiene varias características que lo hacen sobre salir del resto, ya que tiene un precio muy bajo y funciona con arduino que es un hardware y software libre. (COLFIRE, 2013, <https://www.coldfire-electronica.com/esp/item/154/55/acelerometro-giroscopio-6-dof-imu-gy-521-mpu-6050>).

Las principales características de este sensor se las detalla en la tabla 6 – 1.

Tabla 6 – 1: Características del MPU – 6050

Características MPU – 6050	
Módulo	MPU-6050 (giroscopio de tres ejes + acelerómetro triaxial)
Chip	MPU – 6050
Alimentación	3-5V (incluye un regulador interno LDO)
Grados de libertad	6
Comunicación	Digital I2C
Convertidor	analógico a digital de 16 bit en todos los ejes
El giroscopio dispone escalas programables	$\pm 250, \pm 500, \pm 1000$ y ± 2000 ° / seg (dps).
Sensibilidad del Giroscopio	De hasta 131 LBSs/dps
La escala del acelerómetro es programables	$\pm 2g, \pm 4g, \pm 8g$ y $\pm 16g$.
Sensor integrado	Temperatura este sensor es de salida digital
Pines de paso estandar	2,54 mm
Tamaño	2 x 1.6 x 0. 1mm

Fuente: <https://www.coldfire-electronica.com/esp/item/154/55/acelerometro-giroscopio-6-dof-imu-gy-521-mpu-6050>

Realizado por: REDROBÁN, Laura, 2017

1.7.1.2 Estructura Interna del Módulo MPU - 6050

Dentro de la estructura básica de los sensores inerciales se encuentra como mínimo un acelerómetro y un giroscopio para medir la aceleración y la velocidad angular en los tres ejes respectivamente. También se encuentra un microprocesador que es capaz de almacenar los datos receptados por estos sensores para transmitirlo de manera exacta a un ordenador por medio del protocolo de comunicación que este posee, donde serán procesados según la necesidad del usuario.

El acelerómetro es el encargado de medir la aceleración lineal en los tres ejes x, y, z, están integrados en silicio, utilizando la tecnología MEMS⁶, ya que es necesario reducir su tamaño para poder incorporar en el módulo.

Calcula la aceleración mediante voltaje generado entre dos placas, donde una varía su posición dependiendo del movimiento del acelerómetro, como se muestra en la figura 11 - 1.

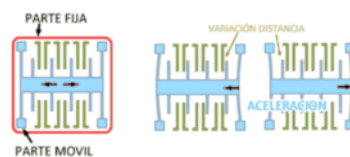


Figura 11 – 1: Funcionamiento del Acelerómetro

Fuente: <https://www.luisllamas.es/como-usar-un-acelerometro-arduino/>

El acelerómetro que está integrado en el MPU 60-50 tiene varias características que se dan a conocer en la tabla 7 - 1:

Tabla 7 - 1: Características del Acelerómetro del MPU 60-50

Características Acelerómetro
Tiene 3 ejes con un rango de escala programable de ± 2 gramo, ± 4 gramo, ± 8 gramo y ± 16 gramo
Tiene integrado ADCs de 16 bits que permite el muestreo simultaneo
Funciona a $500\mu\text{A}$
En baja potencia el acelerómetro actúa a $10\mu\text{A}$ en 1.25Hz, $20\mu\text{A}$ a 5 Hz, 20 Hz a 60mA , $110\mu\text{A}$
Interrupciones programables por el usuario

Fuente: https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf

Realizado por: REDROBÁN, Laura, 2018

El giroscopio mide la posición basándose en los principios del momento angular, este sensor indica en grados por segundos la velocidad angular, como se muestra en la figura 12 - 1. (VARA, 2014, p,39)

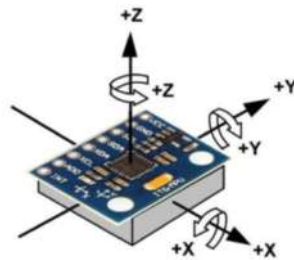


Figura 12 - 1: Funcionamiento del Giroscopio

Fuente: <https://i0.wp.com/www.ingmecafenix.com/wp-content/uploads/2017/05/Ejes>

1.7.1.3 Ventajas del Módulo MPU - 6050

- Su costo es bajo en comparación a otros sensores inerciales.
- Tiene un diseño muy compacto.
- Interactúa con varios sensores digitales no inerciales.
- Es compatible con la familia MPU – 30X0
- Tiene acelerómetro, giroscopio y un sensor de temperatura integrados.
- Es muy exacto en sus mediciones.
- Se usa la comunicación I2C para transmitir datos utilizando arduino.
- Usa alimentación menor a 5V. (HISPAVILA, 2015, <https://www.hispavila.com/el-sensor-mpu-6050/>)

1.8 Pantalla de Cristal Líquido (LCD)

Las pantallas de cristal líquido (LCD) son dispositivos electrónicos de visualización, en los cuales se puede mostrar información, caracteres o figuras de manera clara dependiendo la aplicación que se lo va a dar. Este dispositivo permite la comunicación máquina humano, ya que este puede mostrar mensajes de funcionamiento de las mismas.

La LCD se muestra en la figura 13 - 1.

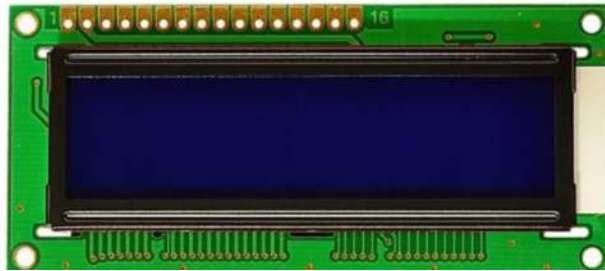


Figura 13 - 1: Pantalla de Cristal Líquido de 2x16

Fuente: http://site.gravitech.us/MicroResearch/Others/LCD-16x2B/LCD-16x2B_2R.jpg

Para el desarrollo de este prototipo se elige un lcd ya que en ellos se puede visualizar mayor número de caracteres con claridad y se optimiza el número de pines de conexión. Para la programación existe una librería específica para el uso de lcd.

1.8.1 Memorias del LCD

Las pantallas de cristal líquido cuentan con dos memorias internas que son: la memoria DDRAM y la memoria CGROM.

1.8.1.1 Memoria DDRAM

La memoria DDRAM (Data Display Ram) es la encargada de almacenar los datos que se van a visualizar en la pantalla de cristal líquido siendo estos caracteres, símbolos o figuras con sus respectivas posiciones.

En la figura 14 – 1 se muestra los caracteres visualizados en la pantalla usando memoria DDRAM.



Figura 14 - 1: Memoria DDRAM del LCD

Fuente: <http://4.bp.blogspot.com/-xtIcab43uk/UQPv-SB63EI/AAAAAAAAAC0g/sPz8FpPj730/s1600/LCD.jpg>

1.8.1.2 Memoria CGROM y CGRAM

La memoria CGROM es la memoria donde se almacena una tabla con los posibles caracteres que se pueden visualizar en la pantalla de cristal líquido, y la memoria CGRAM (Character generator Ram) es en la que puede almacenar datos propios, como se muestra en la figura 15 - 1. (SALAS, 2013, <http://todoelectrodo.blogspot.com/2013/02/lcd-16x2.html>)

4 Higher bits of address			
0000	0001	0010	0011
0000 0000	00P`P	-P3αp	
0000 0001	!1AQa9	α 7#4äq	
0000 0010	"2BRbr	「イツpθ	
0000 0011	#3CScs	」ウチεε	
0000 0100	\$4DTdt	、イトμΩ	
0000 0101	%5EUeu	・オナ1σÜ	
0000 0110	&6FVfv	ヲカニヨρΣ	
0000 0111	?7GW9w	ヲキヌラ9π	
0000 1000	<8HXhx	イウネリrε	
0000 1001	>9IYiy	ウケルル'y	
0000 1010	*:JZjz	エコハレjチ	
0000 1011	+;Klk<	オザヒロ*チ	
0000 1100	,<L¥ll	チシフワチ再	
0000 1101	-=Mlm)	ユスヘンもチ	
0000 1110	.>N^n^	ヨセホ^ñ	
0000 1111	/?0_0+	ツツマ^ö	

Figura 15 - 1: Memoria CGROM de LCD

Fuente: <https://www.mikroe.com/ebooks/pic-microcontrollers-programming-in-basic/additional-components>

1.8.2 Características y Pines de Conexión

Cada pin existente en la pantalla tiene una funcionalidad la cual se especifica en la tabla 8 - 1:

Tabla 8 – 1: Pines de conexión de la LCD

Pin	Símbolo	Descripción
1	Vss	Tierra de alimentación GND
2	Vdd	Alimentación de +5V cc
3	Vo	Ajuste del contraste de CL (0 a +5V)
4	RS	Sección de registro de datos
5	R/W	Lectura / Escritura LCD
6	F	Habilitación
7	D0	Bit menos significativo
8	D1	
9	D2	
10	D3	
11	D4	
12	D5	
13	D6	
14	D7	Bit más significativo
15	A	Alimentación del back light +3,5V o +5V
16	K	Tierra GND de back light

Fuente: Función de cada pin del LCD. (REYES, 2006, p.79)

Realizado por: REDROBÁN, Laura, 2017

CAPITULO II

2. MARCO METODOLÓGICO

En el presente capítulo se especifica los pasos a realizar para el diseño e implementación de un prototipo para el control y tarificación del servicio de taxi a través de sensores inerciales, dentro del cual se analiza las ventajas que se obtiene al utilizar las diferentes tecnologías existentes en el país. En el gráfico 1 - 2 se detalla las etapas que se sigue secuencialmente para la realización del prototipo.

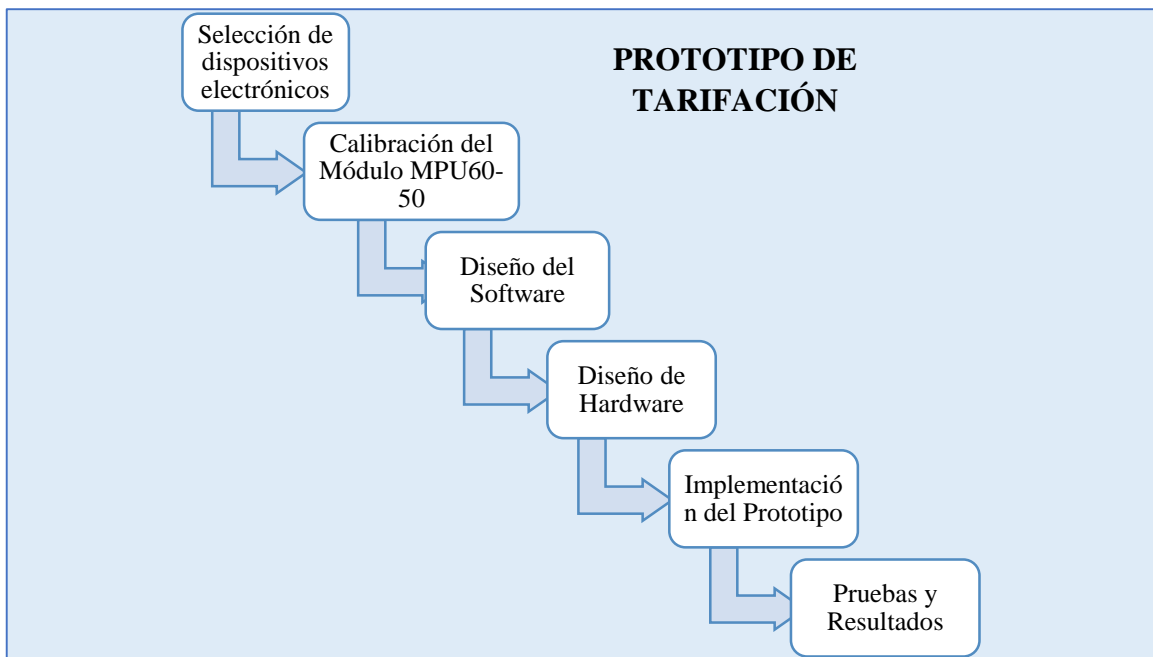


Gráfico 1 - 2: Etapas del diseño e implementación del prototipo
Realizado por: REDROBÁN, Laura, 2018

En el gráfico 2 - 1 se detalla la estructura del diseño e implementación del prototipo para el control y tarificación del servicio de taxi con sensores inerciales.

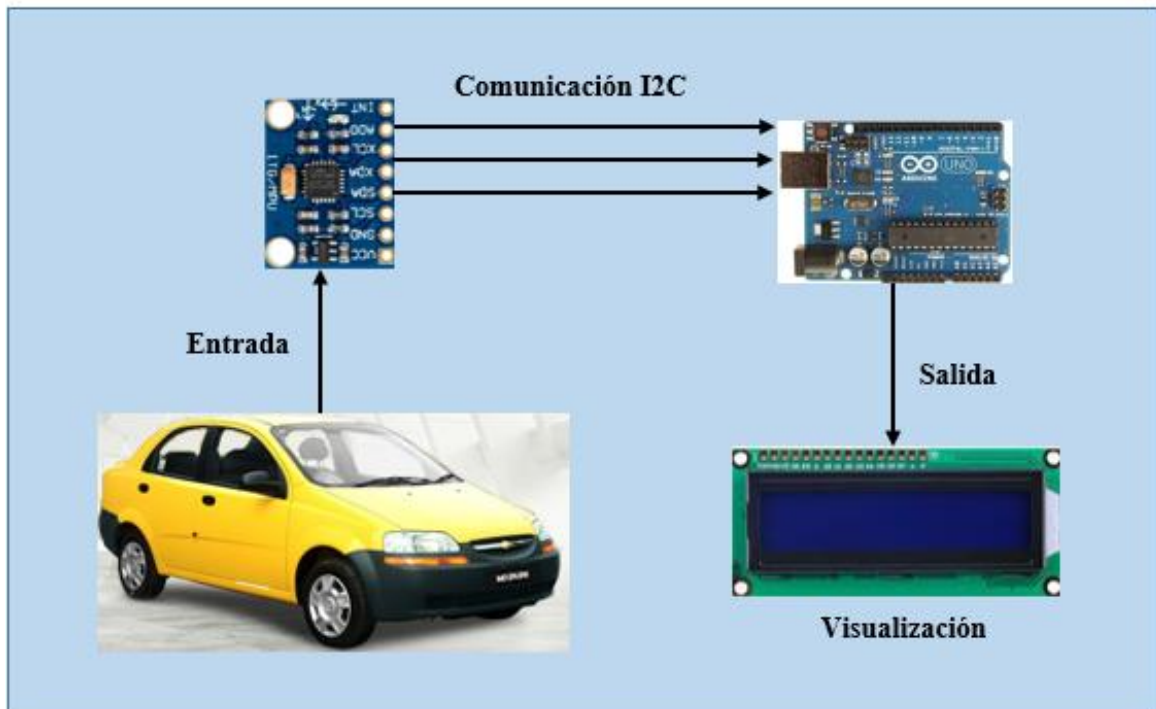


Gráfico 2 – 2: Estructura del Prototipo de Control y Tarifación
 Realizado por: REDROBÁN, Laura, 2018

2.1 Selección de Dispositivos Electrónicos

La parte principal para el desarrollo del proyecto es la selección de la mejor tecnología en dispositivos y elementos electrónicos para el desarrollo del prototipo. Esta etapa está estructurada por un módulo MPU 60-50 y una tarjeta de desarrollo electrónico Arduino Uno. A continuación se especifica el uso de cada uno de ellos.

2.1.1 Módulo MPU 60-50

Durante la selección de los diferentes dispositivos para la elaboración del prototipo se tomó en cuenta algunas especificaciones, donde este módulo cumple con ellas, las cuales son:

- Reducido tamaño
- Bajo costo
- Datos confiables
- Compatibilidad con las tarjetas de desarrollo electrónico

2.1.1.1 Condiciones de funcionamiento del módulo MPU 60-50

Para una obtención de datos fiables del módulo MPU 60-50, se debe evitar ponerlo en funcionamiento en condiciones máximas durante largos períodos de tiempo, ya que esto ocasionaría daños en el dispositivo. Las condiciones para el correcto funcionamiento se muestran en la tabla 1 - 2.

Tabla 1 – 2: Condiciones de Funcionamiento del Módulo MPU 60.50

Condición	Valor min / máx.
Tensión de Alimentación, Vss	-0,5 V a + 6V
VLOGIC Nivel de entrada de voltaje	-0,5 V a 0,5 V + VDD
REGOUT	-0.5V a 2V
Nivel de voltaje de entrada (CLKIN, AUX_DA, AD0, FSYNC, INT, SCL, SDA)	-0,5 V a 0,5 V + VDD
CPOUT ($2.5V \leq VDD \leq 3.6V$)	-0,5V a 30V
Aceleración (cualquier eje, sin alimentación)	10.000 g durante 0,2 ms
Rango de temperatura de funcionamiento	- 40 ° C a + 105 ° C
Temperatura de almacenamiento	- 40 ° C a + 125 ° C
La descarga electrostática (ESD)	2 kV (HBM); 250V (MM)

Fuente: https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf

Realizado por: REDROBÁN, Laura, 2018

2.1.2 Placa de desarrollo electrónico Arduino UNO

Para el desarrollo del proyecto se analizó los diferentes tipos de tarjetas de desarrollo electrónico Arduino existentes en el mercado, siendo la mejor opción la placa Arduino Uno, ya que esta tarjeta cumple con las especificaciones necesarias las cuales son:

- Tamaño pequeño
- Bajo costo
- Rapidez en la transferencia de datos
- Memoria extensa para el almacenamiento de datos
- Compatibilidad con el módulo MPU 60-50

- Comunicación con la pantalla LCD

Además las características técnicas lo avalan como una tarjeta apta para el desarrollo de este proyecto. La placa Arduino Uno se muestra en la figura 1 - 2:

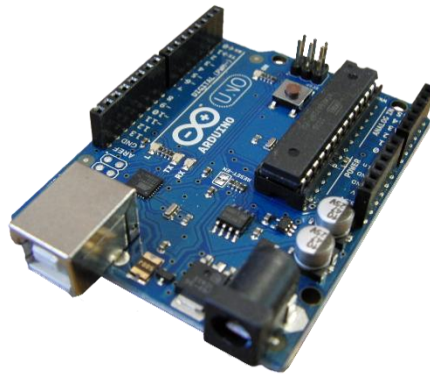


Figura 1 – 2: Tarjeta de desarrollo Arduino UNO

Fuente: <https://upload.wikimedia.org/wikipedia/commons/7/71/Arduino-uno-perspective-transparent.png>

La tarjeta de desarrollo electrónico Arduino Uno, utiliza el microcontrolador ATmega16U2 para el manejo de USB, lo que permite tener una transferencia de datos más rápida y una mayor memoria para el almacenamiento de los mismos. Incluye los pines SDA y SCL los cuales permiten la conexión con el módulo MPU 60-50. Para el uso de esta tarjeta no se necesita la instalación de drivers extras para los sistemas operativos de Windows, Linux o Mac. (THAYER, 2017, <http://arduino.cl/arduino-uno/>)

2.1.1.2 Pines de la tarjeta de desarrollo electrónico Arduino UNO

La placa Arduino Uno consta con 14 pines los cuales se puede configurar según la necesidad del usuario como entradas o salidas, en los que se puede conectar cualquier tipo de dispositivo que puede recibir o transmitir señales de 0 y 5V.

Además, dispone de entradas y salidas analógicas, donde las entradas sirven para adquirir datos de variaciones continuas de un voltajes proporcionados por sensores y traslada las señales a un convertor análogo / digital de 10 bit. Las salidas se las utiliza para enviar señales de control PWM.

Los pines especiales que posee la tarjeta Arduino se detallan en la tabla 2 - 2:

Tabla 2 – 2: Pines especiales de la placa Arduino Uno

PIN	DETALLE
RX y TX	Pines de transmisión de señales TTL
Interrupciones internas	Pin 2 y 3 generan interrupciones en el Atmega
PWM	Genera señal PWM de hasta 8 bit
SPI	Los pines 10, 11, 12 y 13 pueden utilizarse para llevar a cabo comunicaciones SPI, que permiten trasladar información full dúplex en un entorno Maestro/Esclavo
I2C	Permite establecer comunicaciones a través de un bus I 2C.

Fuente: <http://www3.gobiernodecanarias.org/medusa/ecoblog/ralvgon/files/2013/05/Caracter%C3%ADsticas-Arduino.pdf>

Realizado por: REDROBÁN, Laura, 2018

La distribución de los pines de la placa Arduino Uno se lo muestra en la figura 2 - 2:

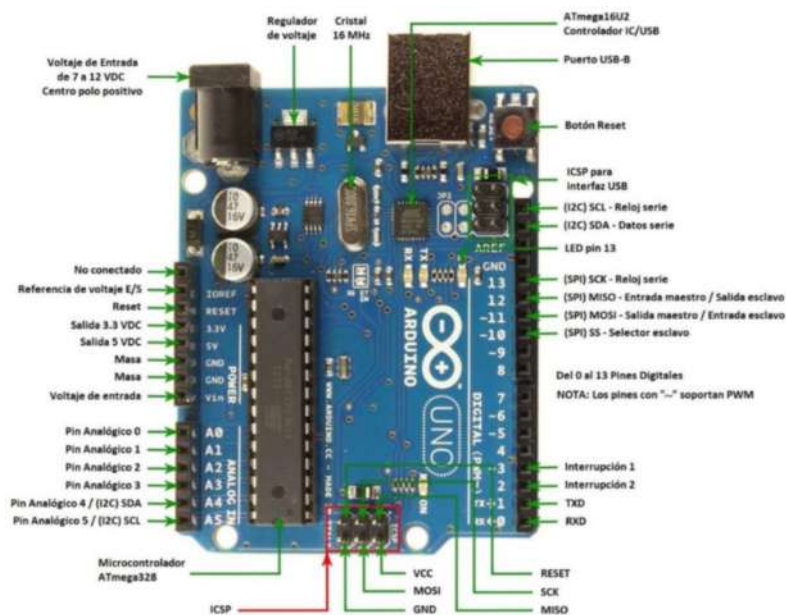


Figura 2 – 2: Distribución de pines de la placa Arduino UNO

Fuente: <https://3.bp.blogspot.com/-Rt8U1UsCuO/U/VmsUCzB4njI/AAAAAAAAAAs/T9fTSSWWPjPA/s1600/Pinout.jpg>

2.2 Calibración del módulo MPU 60-50

La calibración del módulo MPU60-50 es necesaria, ya que se determina los valores de referencia para las mediciones realizadas por el mismo. Para esto se realiza la conexión del módulo con la placa arduino para la calibración del mismo, para una correcta conexión se debe revisar el DataSheet, el cual se encuentra en el Anexo A.

2.2.1 Comunicación entre módulo MPU 60-50 y placa Arduino Uno

La comunicación entre la tarjeta de desarrollo electrónico Arduino y el módulo MPU 60 - 50 es por medio del bus I2C. Los pines SCL y SDA, permiten una conexión directa con la tarjeta Arduino. (NAYLAMP, 2016, http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html)

2.2.2.1 Bus de comunicación I2C

El bus I2C es considerado como una comunicación en serie diseñado por Philips, tiene una velocidad estandar de 100kbit/s. Este tipo de comunicación es muy utilizado en el medio industrial, ya que este bus permite comunicar microcontroladores y dispositivos de sistemas integrados.

Una de las principales características que posee el I2C es que usa dos líneas de transmisión que son:

- SDA: Transmite los datos
- SCK: Transmite la señal de reloj

Al momento de la transmisión, si existen dos o más señales que están siendo transmitidas, estas pueden causar conflictos, lo que desencadenaría en el envío de datos erróneos. Existe una dirección única para cada dispositivo conectado a este bus, estos pueden funcionar como

maestro o esclavos dependiendo la utilidad que se lo va a dar. Un ejemplo de conexión se puede visualizar en la figura 3 - 2. (WORDPRESS, 2016, <https://aprendiendoarduino.wordpress.com/category/i2c/>)

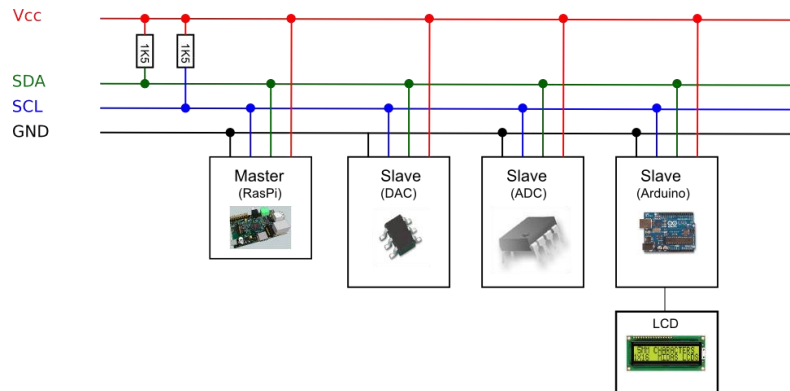


Figura 3 – 2: Ejemplo comunicación I2C

Fuente: <http://www.arduinoob.com/wp-content/uploads/2014/12/i2c-bus-arduino.png>

Para realizar la comunicación entre el módulo MPU60 - 50 y la tarjeta de desarrollo electrónico Arduino Uno existe dos tipos de direcciones I2C, las cuales se muestran en la tabla 3 - 2:

Tabla 3 – 2: Tipos de direcciones I2C

Pin AD0	Dirección I2C
AD0 = HIGH (5V)	0x69
AD0 = LOW (GND o NC)	0x68

Fuente: http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html

Realizado por: REDROBÁN, Laura, 2018

En la parte interna del módulo MPU 60-50, el pin ADDR tiene una resistencia a GND, si este pin no se conecta, el módulo por defecto toma la dirección 0x68. A la vez el módulo posee un regulador de voltaje a 3.3V, lo que permite ser alimentado directamente con los 5 V que proporciona la placa arduino. (NAYLAMP, 2016, http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html)

2.2.3 Lectura de valores RAW del MPU 60-50

Los valores RAW (datos sin procesar) permite verificar la correcta comunicación entre en módulo MPU 60 - 50 y la placa arduino.

Para obtener estos datos y verificar la comunicación entre el módulo y la placa arduino, primero se debe instalar la interfaz de programación de arduino con las librerías necesarias para la conexión. Las librerías requeridas son dos, las cuales se puede descargar de la página de oficial de Arduino. Las librerías se detallan en la tabla 4 - 2.

Tabla 4 – 2: Librerías para la comunicación entre el MPU 60-50 y la placa Arduino Uno

Librería	Detalle
I2Cdev.h	Controla el funcionamiento del MPU 60-50
MPU60-50.h	Leer los datos

Fuente: <https://www.arduino.cc/en/Main/Software#>

Realizado por: REDROBÁN, Laura, 2018

Una vez instalada las librerías, en el IDE de programación se abre el ejemplo “MPU60-50_raw”, este programa permite la obtención de los datos en bruto para comprobar la correcta comunicación, el cual por defecto aparece luego de colocar la librería “MPU60-50.h”. Este ejemplo se lo compila y carga en la placa arduino.

Si se establece una correcta comunicación entre el módulo MPU 60 - 50 y la placa Arduino UNO, en la pantalla de monitoreo en tiempo real se visualizaran los datos en bruto que proporciona el módulo. Hay que tomar en cuenta que el módulo trabaja a una velocidad de 57600 de transmisión de los datos en serie. Al momento de inicializar el módulo los rangos de medición son:

- Acelerómetro de -2g a +2g
- Giroscopio de -250°/sec a +250°/sec

Los datos adquiridos se muestran en la figura 4 - 2, donde se observa 6 columnas numéricas, donde las 3 primeras son los datos de la aceleración de los ejes x, y, z y las otras 3 son los datos del giroscopio en los tres ejes x, y, z.

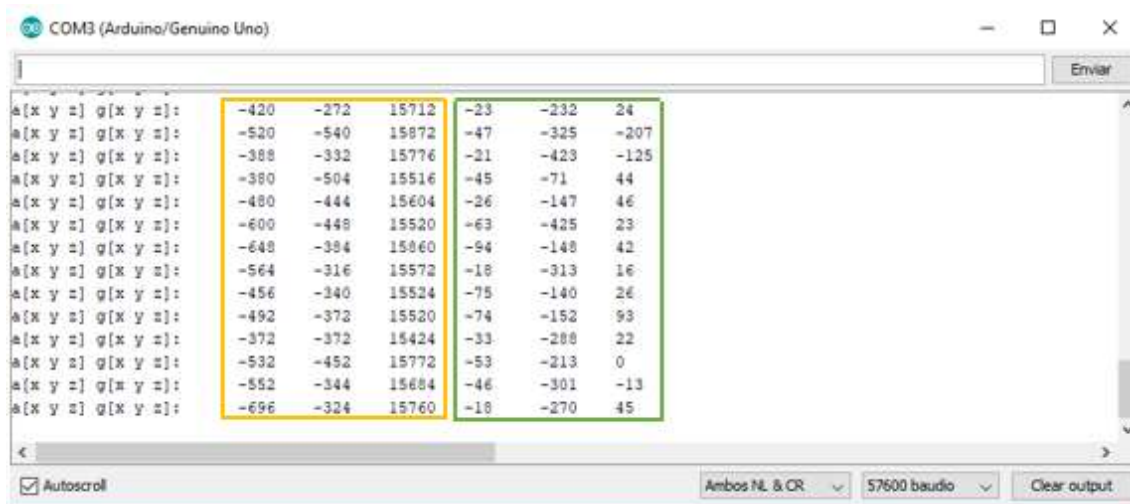


Figura 4 – 2: Distribución de los datos en brutos adquiridos con el módulo MPU60-50
Fuente: REDROBÁN, Laura, 2018

2.2.4 Calibración del módulo MPU 60-50

Para poder utilizar el módulo MPU 60 - 50, este hay que calibrarlo dependiendo en la posición que se lo vaya a ubicar, para tener un correcto sentido de los datos y evitar un margen de error elevado.

Para realizar la calibración del sensor se utiliza el comando OFFSETS, el cual lee los datos iniciales y compensa los errores obtenidos durante la lectura, el programa se lo adjunta en el anexo B.

El programa diseñado tiene la finalidad de calibrar los valores de inicialización del acelerómetro y giroscopio del módulo, en base al DataSheet del módulo los valores son $a_x=0$, $a_y=0$, $a_z=1g$, y $g_x=0$, $g_y=0$, $g_z=0$, ya que para el desarrollo del proyecto solo se usará el acelerómetro del módulo para el cálculo de la distancia recorrida y el giroscopio para determinar si el vehículo está en movimiento, por lo que es necesario calibrarlo.

Al momento de mandar a ejecutar el programa, se debe tomar en cuenta que el módulo debe ser colocado en la posición que se lo va utilizar, para que los offsets lean los valores actuales con los que va ser calibrado. Una vez colocado se debe enviar un carácter cualquiera, con el cual se dará inicio a la calibración, como se muestra en la figura 5 - 2.

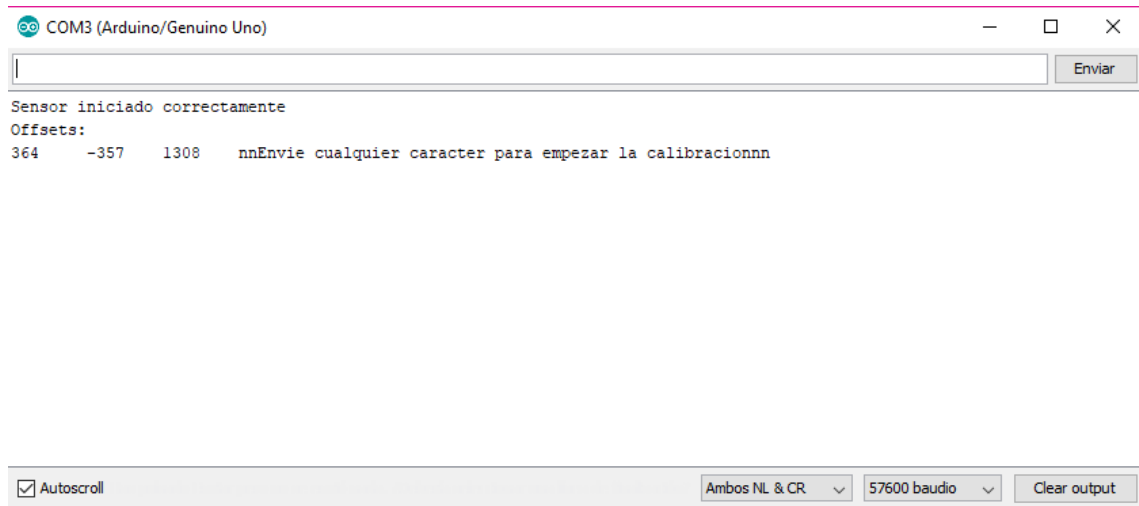


Figura 5 – 2: OFFSETS de inicialización del acelerómetro del MPU 60-50
Fuente: REDROBÁN, Laura, 2018

Al enviar el carácter que da inicio a la calibración del módulo, permite que cada 100 lecturas se compruebe si los datos están cercanos a los valores deseados, siendo estos:

- ax=0
- ay=0
- az=+16384

Una vez visualizados en el monitor en tiempo real del IDE de arduino, valores cercanos a los mencionados anteriormente se debe desconectar el módulo para que la calibración finalice, y volver a cargar el programa de adquisición de datos inicial, así empezará a leer los datos a partir de los calibrados. Los valores en los cuales se calibró el módulo se muestran en la figura 6 - 2:

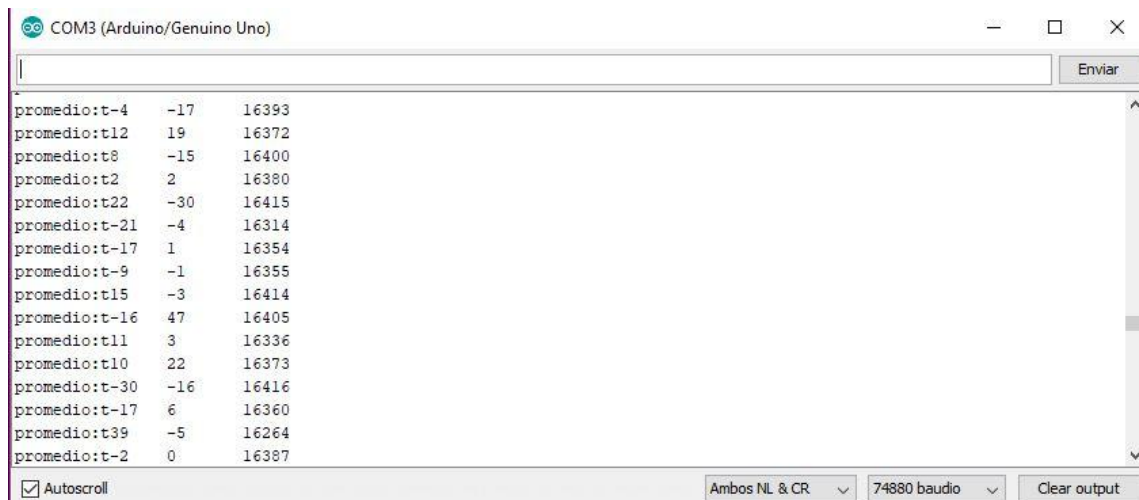


Figura 6 – 2: Valores calibrados del módulo MPU 60-50
Fuente: REDROBÁN, Laura, 2018

2.3. Diseño del software del funcionamiento del prototipo de tarificación

El diseño del Software que realiza el control, cálculo y tarificación del servicio de taxi, se lo ejecuta a partir de un diagrama de flujo. El cual se lo detalla de manera general, ya que dentro de la rutina existen subrutinas, las mismas que se las explica en los siguientes literales. El diagrama de flujo se muestra en la figura 7 - 2.

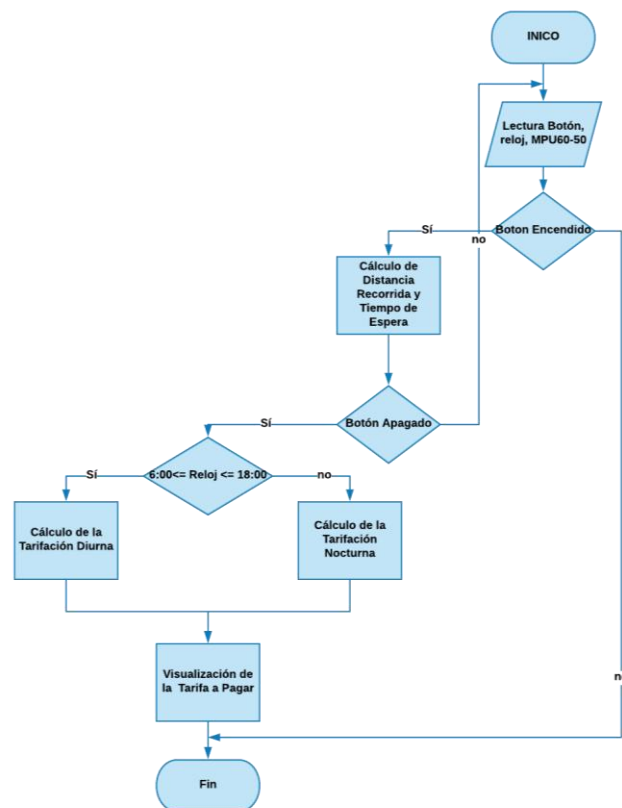


Figura 7 – 2: Diagrama de Flujo del Funcionamiento del Taxímetro
Fuente: REDROBÁN, Laura, 2018

2.3.1 Lectura de los datos del Módulo MPU 60-50

Al momento de la calibración del módulo MPU 60-50, se obtienen datos por medio de la técnica de observación directa que son proporcionados por el módulo. Los cuales están en unidades diferentes a las necesarias, por lo que se realizará una conversión de unidades y la implementación de un filtro para estabilizar la lectura.

2.3.1 .1 Cambio de unidades de los valores del MPU 60-50

El módulo MPU 60-50 al momento de realizar la lectura y transmisión de los valores, lo hace en las unidades que vienen predeterminadas por cada sensor que se encuentra en la parte interna del módulo. Las unidades que vienen predeterminadas y a las que se quiere hacer la conversión se las detalla en la tabla 5 - 2:

Tabla 5 – 2: Tabla de cambio de unidades

Sensor	Unidad Predeterminada	Unidad Convertida
Acelerómetro	G	m/s ²
Giroscopio	°/s	deg/s

Realizado por: REDROBÁN, Laura, 2018

Para el cambio de unidades hay que tomar en cuenta los rangos en los cuales está configurado el módulo, para el acelerómetro son 2g/4g/8g/16g y para el giroscopio 250/500/1000/2000(°/s). Siguiendo estos rangos los valores máximos y mínimos de medición del módulo MPU 60-50 los cuales se detallan en la tabla 6 - 2:

Tabla 6 – 2: Valores Máximos y Mínimos de lectura del MPU 60-50

Variable	Valor Mínimo	Valor Central	Valor Máximo
Aceleración	-2g	0	+2g
Velocidad Angular	-250°/s	0°/s	+250°/s

Fuente: http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html

Realizado por: REDROBÁN, Laura, 2018

El cambio de unidad se va a realizar solo para la aceleración, ya que es el dato necesario para el cálculo de la distancia recorrida. La ecuación 1 - 2 es la que se utiliza para la conversión de la aceleración en cada eje, donde 9,81 m/s² es el valor de la gravedad y 16384 es un valor constante predeterminado dentro de las especificaciones del DataSheet del módulo.

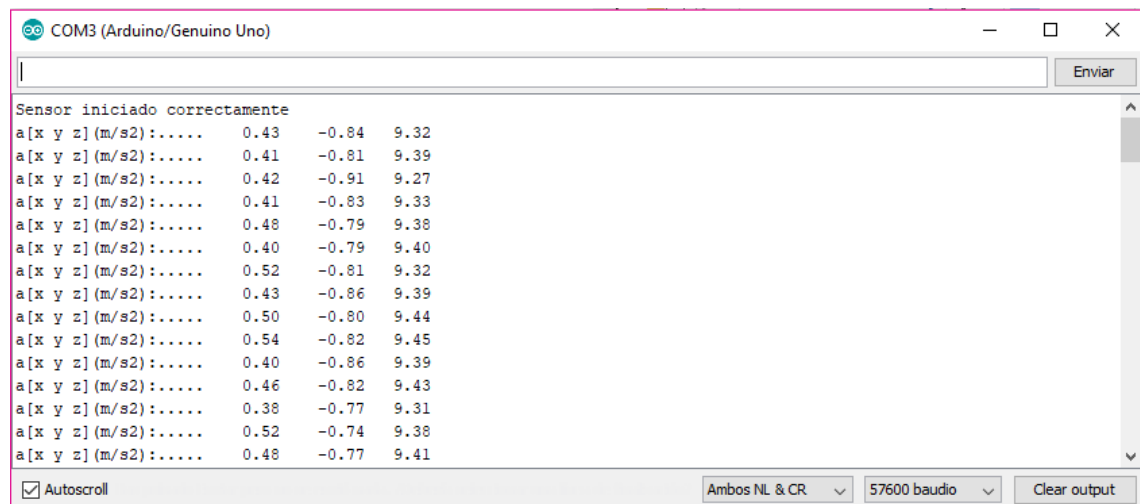
Ecuación 1 - 2: Ecuación para la conversión de unidades de [g] a [m/s²]

$$ax \left[\frac{m}{s^2} \right] = ax * \left(\frac{9,81}{16384,0} \right)$$

$$ay \left[\frac{m}{s^2} \right] = ay * \left(\frac{9,81}{16384,0} \right)$$

$$az \left[\frac{m}{s^2} \right] = az * \left(\frac{9,81}{16384,0} \right)$$

Una vez ingresada la ecuación 1 - 2 en el programa, y cargarlo en la placa arduino, los valores leídos por el módulo y con su respectiva conversión se visualizan en la figura 8 - 2:



The screenshot shows a serial monitor window titled "COM3 (Arduino/Genuino Uno)". The output text is as follows:

```
Sensor iniciado correctamente
a[x y z] (m/s2):..... 0.43 -0.84 9.32
a[x y z] (m/s2):..... 0.41 -0.81 9.39
a[x y z] (m/s2):..... 0.42 -0.91 9.27
a[x y z] (m/s2):..... 0.41 -0.83 9.33
a[x y z] (m/s2):..... 0.48 -0.79 9.38
a[x y z] (m/s2):..... 0.40 -0.79 9.40
a[x y z] (m/s2):..... 0.52 -0.81 9.32
a[x y z] (m/s2):..... 0.43 -0.86 9.39
a[x y z] (m/s2):..... 0.50 -0.80 9.44
a[x y z] (m/s2):..... 0.54 -0.82 9.45
a[x y z] (m/s2):..... 0.40 -0.86 9.39
a[x y z] (m/s2):..... 0.46 -0.82 9.43
a[x y z] (m/s2):..... 0.38 -0.77 9.31
a[x y z] (m/s2):..... 0.52 -0.74 9.38
a[x y z] (m/s2):..... 0.48 -0.77 9.41
```

At the bottom of the window, there are settings: "Autoscroll" is checked, "Ambos NL & CR" is selected, "57600 baudio" is selected, and there is a "Clear output" button.

Figura 8 – 2: Valores leídos por el MPU60-50, cambiados de unidades

Fuente: REDROBÁN, Laura, 2018

2.3.1.2 Filtro para el módulo MPU 60-50

Una vez calibrado el módulo MPU 60 - 50, hay que tratar de corregir las interferencias que este puede tener por factores externos como: el ruido, el aire y la gravedad.

Existen diferentes tipos de algoritmos de filtros que se pueden utilizar, dependiendo la utilidad del módulo, para este caso se utiliza el filtro conocido como “ Filtro Complementario “ el cual

se lo implementa en Arduino de manera sencilla, es de fácil utilización, bajo costo de procesamiento y es muy preciso. El filtro complementario es la unión de dos filtros uno paso alto que actúa sobre el giroscopio y un pasa bajo que actúa en el acelerómetro. En la figura 9 – 2 se visualiza los datos y la señal del módulo sin la implementación del filtro. (TRANSDUCTOR, 2014, <https://robologs.net/2014/10/15/tutorial-de-arduino-y-mpu-6050/>)

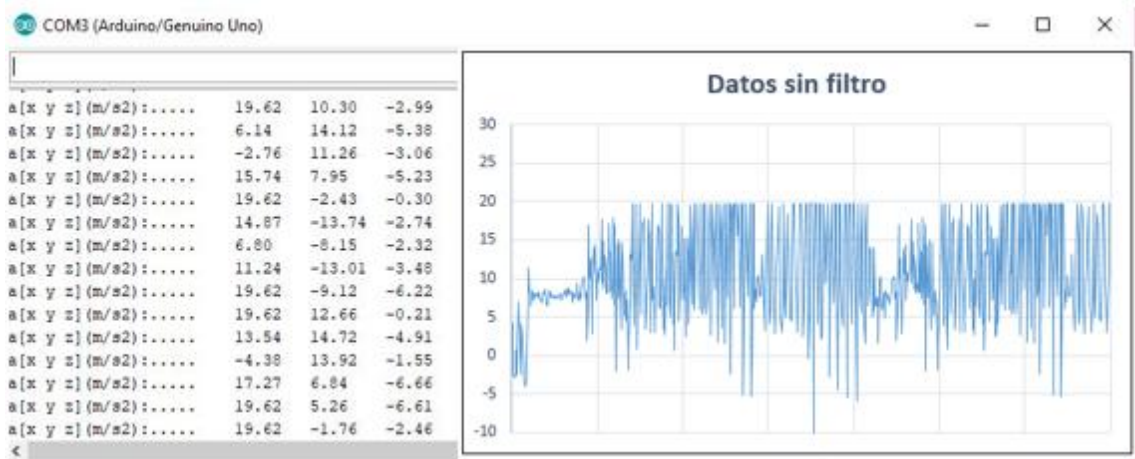


Figura 9 – 2: Valores y señal sin filtrar del módulo MPU 60-50
Fuente: REDROBÁN, Laura, 2018

Para filtrar los datos, se debe declarar las variables a usar e ingresar en la programación la expresión del filtro, la cual se detalla en la ecuación 2 - 2:

Ecuación 2 – 2: Filtro Exponencial EMA con Filtro Pasa Bajo y Pasa Alto para Arduino

$$A_n = A_{n-1} + \alpha M(1 - \alpha)$$

Donde:

A_n : Valor Filtrado

A_{n-1} : Valor Filtrado Anterior

M: Valor Muestrado de la señal a filtrar

α : Factor entre 0 y 1

En la figura 10 – 2 se muestra los datos y la gráfica filtrada del módulo MPU 60 – 50.

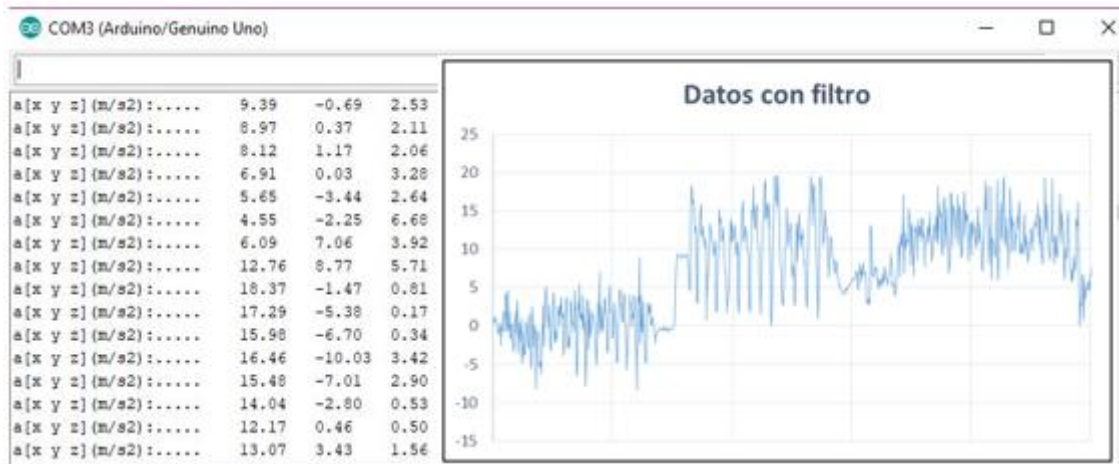


Figura 10 – 2: Valores y señal filtrada del módulo MPU 60-50

Fuente: REDROBÁN, Laura, 2018

Las líneas de código del cambio de unidades y diseño del filtro se encuentran en el Anexo C.

2.3.2 Implementación del Reloj

La implementación de un reloj para la elaboración del prototipo es de vital importancia, ya que este permite saber que tarifa se debe calcular dependiendo la hora en la que inicie el servicio del recorrido del taxi. Este servicio tiene dos tipos de tarifación en función de la hora, las cuales se indican en la tabla 7 - 2:

Tabla 7 – 2: Tarifas en función de la hora

Hora	Tarifa
06:00 a 18:59	Diurna
19:00 a 05:59	Nocturna

Fuente: <http://www.gadmriobamba.gob.ec/index.php/transparencia/lotaip2016/ordenanzas-2016?download=658:ordenanza-009-2016-tarifas-taxis>

Realizado por: REDROBÁN, Laura, 2018

Para el diseño del programa del reloj, se lo realiza en el IDE del arduino. Mediante un diagrama de flujo en el que se explica el funcionamiento.

El diagrama de flujo se muestra en a figura 11 - 2:

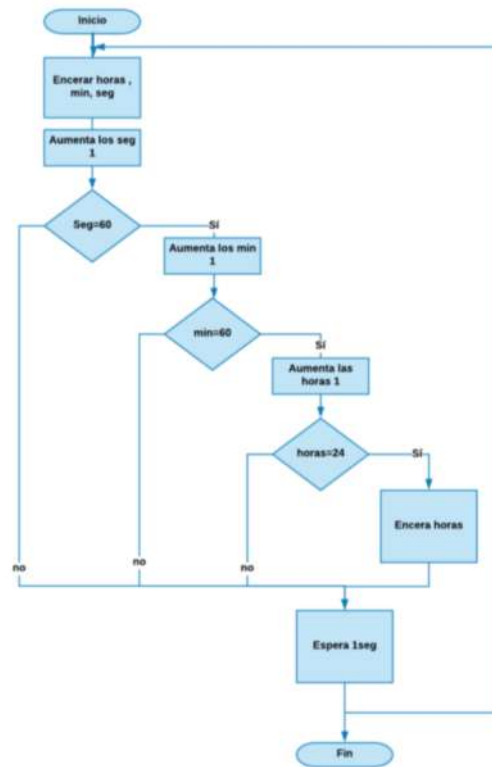


Figura 11 - 2: Diagrama de Flujo del Reloj
Fuente: REDROBÁN, Laura, 2018

El reloj diseñado es uno de 24 horas, por medio de condiciones if que permite la repetición del ciclo. El resultado se visualiza en una pantalla lcd de 16x2. Las líneas de código del programa del prototipo se lo encuentran en el anexo D.

2.3.3 Cálculo de Distancia Recorrida

Para realizar el cálculo de la tarifa a pagar por el servicio de taxi es necesario conocer la distancia recorrida por el vehículo en ese trayecto. Al utilizar el módulo MPU 60-50 el dato que se obtiene es la aceleración, por lo que para determinar la distancia se utiliza el método empleado en los carros de fórmula uno.

Al ser este un movimiento acelerado es complicado, ya que durante el recorrido que realiza el vehículo hay aceleraciones y frenados. Para resolver este movimiento existen dos maneras de encontrar la distancia recorrida.

El primer método es conociendo la velocidad media con la que se desplaza el vehículo y el tiempo que demora en llegar a esa velocidad. Al conocer esos datos se utiliza la ecuación 3 - 2:

Ecuación 3 - 2:Cálculo de Distancia con Velocidad Media

$$X = V_{med} * t$$

X: distancia recorrida

V_{med} : velocidad media

t: tiempo

El segundo método que existe y es el cual se va utiliza, es el que depende de la aceleración y el tiempo de aceleración del carro, con estos datos se aplica la ecuación 4 - 2: (VALENZUELA, 2016, <https://sites.google.com/site/pielagosfq4/home/aceleracion/-que-distancia-se-recorre-con-un-movimiento-acelerado>)

Ecuación 4 - 2:Cálculo de Distancia conociendo la Aceleración

$$X = X_i + V_i * t + \frac{1}{2} * a * t^2$$

X: distancia recorrida

X_i : distancia inicial

V_i : Velocidad inicial

t: tiempo de aceleración

a: aceleración

Todos los vehículos estándar comerciales entre ellos el Chevrolet Aveo Family, año 2014, tienen un tiempo de aceleración de 10 segundos para llegar de 0 a 100km/h, es decir 10km/h en 1 segundo, con la cual se calcula la aceleración que tiene el vehículo. Para la conversión de unidades de km/h*s a m/s², como se muestra en la ecuación 5 - 2.

Ecuación 5 - 2: Cálculo de Aceleración

$$10 \frac{\text{km}}{\text{h}} * 1000 \frac{\text{m}}{\text{km}} * \frac{1\text{h}}{3600 \text{ s}} = 2,77 \frac{\text{m}}{\text{s}^2}$$

Se encuentra la aceleración en las unidades de $\frac{\text{m}}{\text{s}^2}$ por facilidad, ya que el módulo MPU 60-50 está calibrado para sensor en esta unidades.

Una vez conocida la aceleración del vehículo y el tiempo de aceleración, se diseña el diagrama de flujo el cual registrá la rutina de cálculo de distancia. El diagrama se lo visualiza en la figura 12 -2:



Figura 12 - 2: Diagrama de Flujo para el Cálculo de la Distancia Recorrida.

Fuente: REDROBÁN, Laura, 2018

El diagrama de flujo detalla secuencialmente como se va a realizar la rutina. Esta empieza a ejecutarse cuando ingrese la señal del pulsador de inicio, en ese momento el modulo empieza a sensor las aceleraciones cada segundo. Está es positiva cuando el vehículo aumenta la velocidad, negativa cuando el vehículo pierde velocidad y tiene una aceración de $0 \frac{\text{m}}{\text{s}^2}$ cuando el vehículo alcance una velocidad constante.

Tomando en cuenta estos parámetros se hace el cálculo de la distancia recorrida utilizando la ecuación 4 – 2, si la aceleración es positiva se reemplaza los valores únicamente, pero si esta es una aceleración negativa se toma el valor absoluto y se realiza el mismo calculo como si fuese aceleración positiva.

Si la aceleración es $0 \frac{m}{s^2}$, esto significa que el vehículo está parado o alcanzo una velocidad constante, para determinar esto se utiliza el eje y del giroscopio, ya que si este tiene una variación menor a -3 significa que el vehículo esta con velocidad constante y se realiza el cálculo con la última aceleración sensada.

Para el cálculo del tiempo de la aceleración sensado y posteriormente el cálculo de la distancia recorrida se lo hace mediante una regla de tres, como se detalla en la ecuación 6 - 2 y 7 - 3.

Ecuación 6 - 2: Cálculo del Tiempo de la Aceleración Sensada por el MPU 60-50

$$t_{as} = \frac{a_s * 1[s]}{2,77 \left[\frac{m}{s^2} \right]}$$

t_{as} : tiempo de la aceleracion sensada

a_s : aceleracion sensada

Ecuación 7 - 2: Cálculo de la Distancia Recorrida con la Aceleración Sensada

$$x_r = \frac{1}{2} a_s * t_{as}^2$$

x_r : distancia recorrida

t_{as} : tiempo de la aceleracion sensada

a_s : aceleracion sensada

Las líneas de código con las que se implementó el programa del prototipo se encuentra en el anexo D.

2.3.4 Cálculo de Tiempo de Espera

Otra de las variables utilizadas en tarificación del servicio de taxi es el tiempo de espera, es decir el tiempo en que el vehículo se encuentra parado esperando el cambio de color de un semáforo, a causa del tráfico o al momento de cruzar las calles transversales.

Según lo impuesto por el GAD Municipal de Riobamba cada minuto de espera tiene un costo monetario adicional, para saber el tiempo que el vehículo permanece en reposo se utiliza el diagrama de flujo de la figura 13 - 2.



Figura 13 - 2: Diagrama de flujo para el cálculo del tiempo de espera

Fuente: REDROBÁN, Laura, 2018

Cuando el módulo MPU 60 – 50 envía datos de 0 y ha comprobado que el vehículo está en reposo, se inicializa un contador donde cada 10 segundos se incrementa USD 0,01 al valor de arranque.

Las líneas de código del diagrama de flujo de la figura 11 - 2, se encuentran en el anexo D.

2.3.5 Tarificación

Los taxímetros comerciales utilizados actualmente en la ciudad de Riobamba tienen dos modalidades de funcionamiento diurna y nocturna, las cuales dependen de la hora de inicio del servicio de taxi.

2.3.5.1 Cálculo de la Tarifa Diurna

La tarifa diurna es la que rige en el día, para la cual las tarifas impuestas por el GAD Municipal de Riobamba al servicio de taxi se detallan en la tabla 8 - 2.

Tabla 8 - 2: Tarifa del Servicio de Taxi Modalidad Diurna

Concepto	Tarifa Diurna
Costo de Arranque	USD 0,40
Costo por kilómetro recorrido	USD 0,32
Costo minuto de espera	USD 0,06
Carrera Mínima	USD 1,20

Fuente: <http://www.gadmriobamba.gob.ec/index.php/transparencia/lotaip2016/ordenanzas-2016?download=658:ordenanza-009-2016-tarifas-taxis>

Realizado por: REDROBÁN, Laura, 2018

Según la tabla 15-2 el valor por cada km recorrido es de USD 0,32, es decir que cada 31,25 m se incrementa USD 0,01 al valor de arranque, sabiendo esto se diseña del diagrama de flujo que se visualiza en la figura 14 - 2.

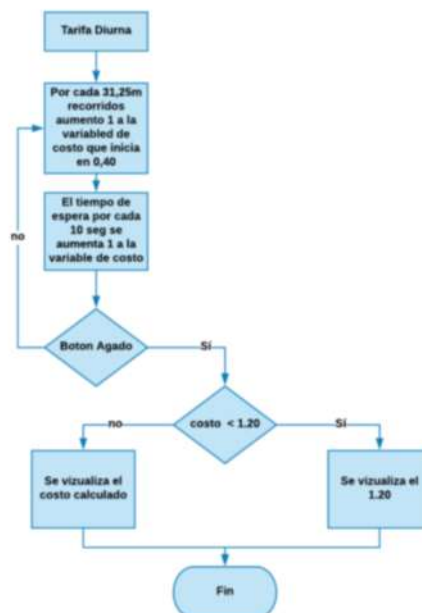


Figura 14 - 2: Cálculo de la Tarifa Diurna

Fuente: REDROBÁN, Laura, 2018

Una vez terminada la rutina se compara si el costo calculado es menor a USD 1.20, se visualiza el costo mínimo impuesto que es USD 1.20, y si es mayor se visualiza el costo calculado. Las líneas de código que realizan la rutina se encuentran en el anexo D.

2.3.5.2 Cálculo de la Tarifa Nocturna

La tarifa nocturna rige para las noches, también tiene su tabla con los costos mínimos del servicio de taxi, los cuales se detallan en la tabla 9 - 2.

Tabla 9 - 2: Tarifa del Servicio de Taxi Modalidad Nocturna

Concepto	Tarifa Nocturna
Costo de Arranque	USD 0,45
Costo por kilómetro recorrido	USD 0,35
Costo minuto de espera	USD 0,06
Carrera Mínima	USD 1,30

Fuente: <http://www.gadmriobamba.gob.ec/index.php/transparencia/lotaip2016/ordenanzas-2016?download=658:ordenanza-009-2016-tarifas-taxis>

Realizado por: REDROBÁN, Laura, 2018

Según la tabla 16-2 el valor por cada km recorrido es de USD 0,35, es decir que cada 38,57 m se incrementa USD 0,01 al valor de arranque, sabiendo esto se procese al diseño del diagrama de flujo que se visualiza en la figura 15 - 2.

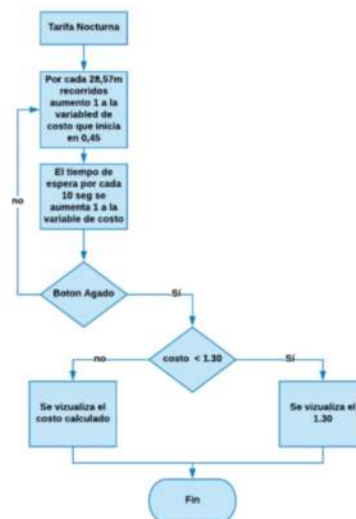


Figura 15 - 2: Cálculo de la Tarifa Nocturna

Fuente: REDROBÁN, Laura, 2018

Una vez terminada la rutina se compara si el costo calculado es menor a USD 1.30, se visualiza el costo mínimo impuesto que es USD 1.30, y si es mayor se visualiza el costo calculado. Las líneas de código que realizan la rutina se encuentran en el anexo D.

2.4 Diseño de Hardware

En el diseño de hardware se explica todas las conexiones entre los dispositivos y elementos electrónicos para su correcto funcionamiento, y el diseño de la carcasa que protegerá al prototipo de tarificación.

2.4.1 Conexión de Dispositivos Electrónicos

En el siguiente apartado se especifica las conexiones con sus respectivos diagramas.

2.4.1.1 Conexión módulo MPU 60-50 con placa Arduino UNO

Hay que tomar en cuenta que los pines o espadines del módulo MPU 60-50 viene separados, es decir hay que soldarlos al módulo de acuerdo a la posición en las que se va a utilizar, en este caso se lo suelda para que el módulo quede en posición horizontal. Como se lo puede apreciar en la figura 16 - 2:

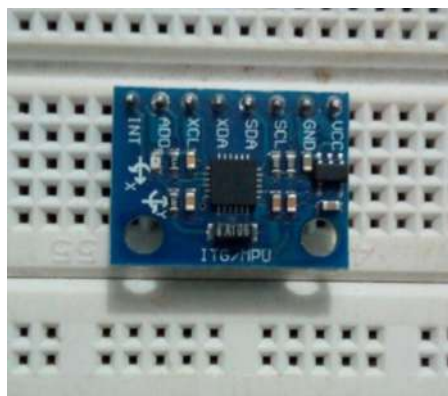


Figura 16 – 2: Módulo MPU 60-50 soldado en posición horizontal

Fuente: REDROBÁN, Laura, 2018

El módulo tiene 8 pines, de los cuales solo se utiliza 5 pines para la conexión con arduino. Los pines que se usa en la conexión se detallan en la tabla 10 - 2:

Tabla 10 - 2: Pines de conexión entre Arduino y MPU 60-50

MPU 60-50	Arduino UNO
Vcc	5V
GND	GND
SCL	Pin Analógico 5
SDA	Pin Analógico 4
INT	Pin Digital 2

Fuente: http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html

Realizado por: REDROBÁN, Laura, 2018

Tomando en cuenta las especificaciones de la tabla 10 - 2, se diseñó el diagrama de conexión, el cual se muestra en la figura 17 - 2:

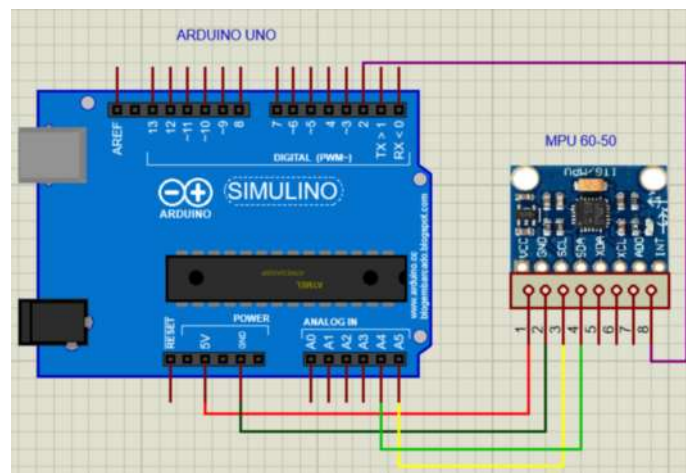


Figura 17 - 2: Diagrama de conexión MPU60-50 y Arduino UNO

Fuente: REDROBÁN, Laura, 2018

2.4.1.2 Conexión pantalla de cristal líquido LCD de 16x2

El uso de las pantallas de cristal líquido da un mejor aspecto a los proyectos, a la vez estas son de bajo costo y consumo. Al momento de conectar hay que tener cuidado ya que tiene varios pines.

En la tabla 11 - 2 se especifica la conexión de los pines.

Tabla 11 - 2: Pines de Conexión del LCD con Arduino UNO

Pines Lcd	Pines Arduino Uno
Vss	5V
VDD	GND
VEE	Pin12
RS	Pin11
RW	Pin10
E	Pin 9
D2	Pin 8
D3	GND
D4	Pin 7
D5	Pin central del potenciómetro
D6	5V
D7	GND

Fuente: <https://www.prometec.net/displays-lcd/#>

Realizado por: REDROBÁN, Laura, 2018

Una vez conocidos los pines de conexión, se diseña el diagrama, el cual se muestra en la figura 18 - 2.

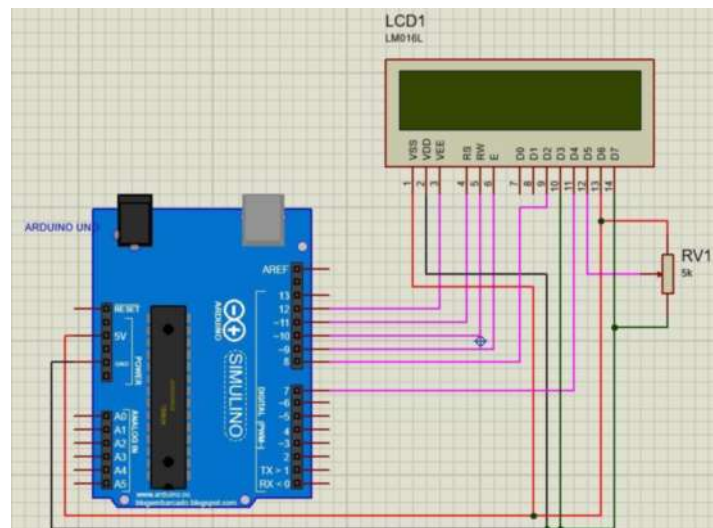


Figura 18 - 2: Diagrama de conexión de LCD de 16x2 con Arduino UNO

Fuente: REDROBÁN, Laura, 2018

2.4.1.3 Conexión del botón de 4 pines

El botón tiene la función de iniciar y finalizar la tarificación, es decir el chofer del vehículo al momento de iniciar una carrera debe pulsarlo y de igual manera al terminar para que el usuario visualice el valor que debe cancelar por el servicio prestado. El diagrama electrónico de conexión del botón se visualiza en la figura 19 - 2.

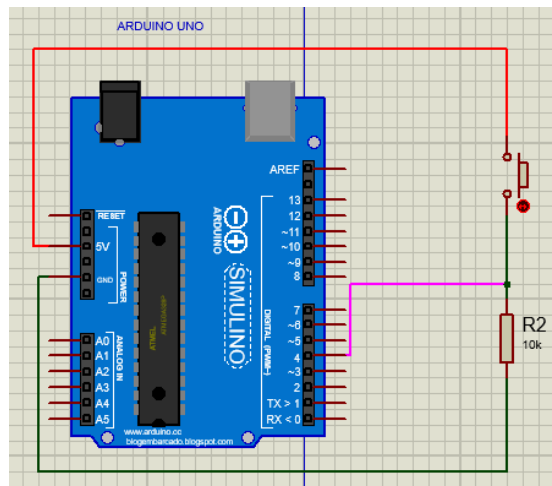


Figura 19 - 2: Conexión del botón con Arduino UNO
Fuente: REDROBÁN, Laura, 2018

Los botones de 4 pines se los conecta igual que los de 2 pines. El diseño esquemático completo del circuito para el prototipo de control y tarificación del servicio de taxi con sensores inerciales se detalla en la figura 20 - 2.

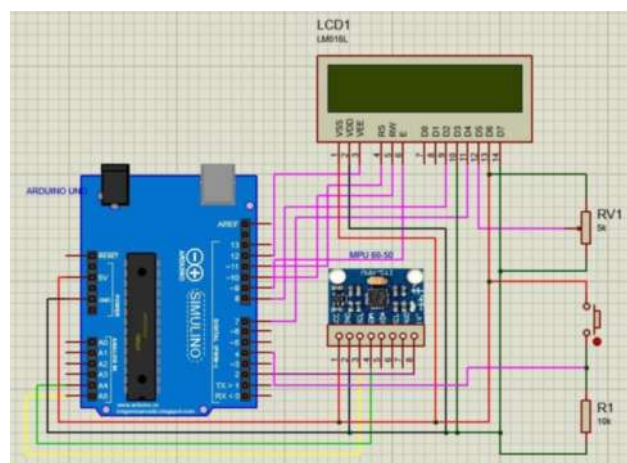


Figura 20 - 2: Diseño esquemático del circuito completo
Fuente: REDROBÁN, Laura, 2018

2.4.2 Diseño y construcción de la carcasa del prototipo

Se diseña una carcasa en la cual se pueda implementar el prototipo de control y tarificación del servicio de taxi, dándole una mejor estética visual para el usuario.

2.4.2.1 Diseño de la Carcasa

Para obtener una carcasa con medidas exactas, el diseño se realiza en el software Solidwork. Las medidas del diseño se muestran en la tabla 12 – 2 y la carcasa terminada en la figura 21 - 2.

Tabla 12 - 2: Medidas de la carcasa diseñada

Parte	Medida
Tapa Superior	7x10cm
Tapa Inferior	7x10cm
Tapa Lateral Derecha	7x7cm
Tapa Lateral Izquierda	7x7cm
Pantalla LCD	7,2x2,5cm
Botones	0,7x0,7cm
Potenciómetro	0,7cm diámetro
Alimentación	1,5x1,5cm

Realizado por: REDROBÁN, Laura, 2018

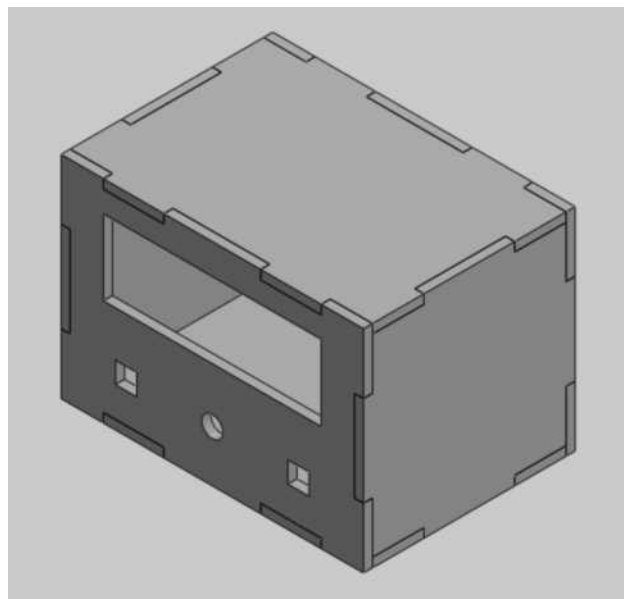


Figura 21 - 2: Carcasa diseñada en el software Solidwork
Fuente: REDROBÁN, Laura, 2018

2.4.2.2 Corte y Armado de la Carcasa

Una vez terminado el diseño se lo convierte se lo guarda en un formato compatible con Illustrator, ya que con este se realiza los cortes con láser para mayor precisión como se visualiza en la figura 22 - 2. Se utiliza MDF de 4mm.



Figura 22 - 2: Corte de la Carcasa
Fuente: REDROBÁN, Laura, 2018

Uniendo todas las piezas cortadas, el resultado final de la carcasa se lo visualiza en la figura 23 - 2:



Figura 23 - 2: Carcasa para el prototipo
Fuente: REDROBÁN, Laura, 2018

2.5 Implementación del Prototipo

El prototipo en sí tiene dos tipos de implementación, una la implementación del prototipo de control y tarifado del servicio de taxi, y la implementación del sistema completo en el vehículo que consta del prototipo y el módulo MPU 60-50.

Finalizada la elaboración de la carcasa, se procede al armado interno, es decir la colocación de todos los elementos en la parte interna y sellado. Dando por terminado la construcción del prototipo, el resultado final se muestra en figura 24 - 2.



Figura 24 - 2: Prototipo de control y tarifación
Fuente: REDROBÁN, Laura, 2018

El prototipo está alimentado con 5V, los cuales se los toma directamente desde el encendedor del vehículo, el cual proporciona 12V y se regula al voltaje necesario por medio de un adaptador que sirve para carga los celulares, ya que el cable de alimentación es USB.

2.5.1 Implementación del prototipo en el vehículo

El prototipo completo para el correcto funcionamiento del proyecto consta del MPU 60-50 y el prototipo de tarifación.

El módulo MPU 60-50 va instalado en la parte externa del vehículo en un lugar donde tenga fácil acceso para el sensado de la aceleración, en este caso se lo coloca en la parte baja del vehículo como se muestra en la figura 25 - 2:

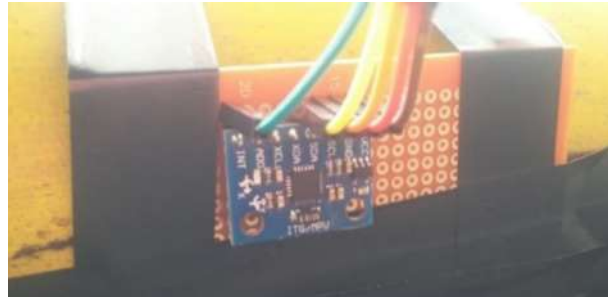


Figura 25 - 2: Instalación del módulo MPU60-50
Fuente: REDROBÁN, Laura, 2018

El prototipo de tarificación se lo coloca en la parte interna del vehículo donde este visible tanto para el usuario como para el prestador del servicio de taxi. Tomando en cuenta que este no tiene que interferir con la visibilidad del conductor, por esto se lo coloca en la parte superior derecha del parabrisas delantero, como se muestra en la figura 26 - 2.



Figura 26 - 2: Prototipo instalado en el interior de vehículo
Fuente: REDROBÁN, Laura, 2018

2.6 Costos del Prototipo de Tarifación

Para el diseño y la implementación del proyecto se utilizó varios elementos y dispositivos electrónicos, con dicho costo se puede realizar una comparación con dispositivos que realizan la misma tarea existente en el país. En la tabla 13-2 se detalla el costo de fabricación del prototipo.

Tabla 13 - 2: Costo de fabricación del prototipo

Cantidad	Materiales	Valor Unitario	Valor Total
1	Arduino UNO	8.00	8.00
1	MPU60-50	7.00	7.00
1	Pantalla LCD	8.00	8.00
2	Botones	0.20	0.40
1	Potenciómetro	0.15	0.15
1	Resistencia	0.10	0.10
1	Baquelita	2.00	2.00
30	Cables	0.15	4.50
1	Corte Laser	7.00	7.00
1	Otros	20.00	20.00
1	Investigación y Conocimiento		42,85
TOTAL			100.00

Realizado por: REDROBÁN, Laura, 2018

Utilizando Excel se realiza una comparación porcentual entre todos los materiales por medio de regla de tres, para constatar el de mayor influencia. Los porcentajes se detallan en la tabla 16 - 2.

Tabla 16 - 2: Costo porcentual de la construcción del prototipo

Materiales	Costo	%Costo
Arduino UNO	8.00	8
MPU60-50	7.00	7
Pantalla LCD	8.00	8
Botones	0.40	0.4
Potenciómetro	0.15	0.15
Resistencia	0.10	0.10
Baquelita	2.00	2
Cables	4.50	4.50
Corte Laser	7.00	7
Otros	20.00	20
Investigación y Conocimiento	42,85	42,85
	100.00	100

Realizado por: REDROBÁN, Laura, 2018

Se observa que el campo de la investigación y del análisis corresponde al 42.85%, siendo el valor más alto obtenido del cálculo de los distintos componentes del prototipo para el control y tarifado del servicio de taxi a través sensores inerciales.

CAPÍTULO III

3 PRUEBAS Y RESULTADOS

Completado el diseño e implementación del prototipo para el control y tarifado del servicio de taxi a través de sensores inerciales, se realizó las pruebas de funcionamiento del prototipo, comparándolo con un taxímetro comercial y avalado por la Agencia Nacional de Transito de Ecuador.

3.1 Análisis Ensayo y Error

Se realizó un análisis de las tarifas marcadas por un taxímetro homologado por la ANT en este caso el Inteligente Taximeter D10 y el prototipo diseñado, estas pruebas se las realizó en un vehículo Chevrolet Aveo Family del año 2014, que presta el servicio del taxi en la Cooperativa “San Nicolás” disco 10-24, de la ciudad de Riobamba. Las pruebas se las realizó de manera manual, donde los dos dispositivos se los pone en marcha simultáneamente durante la duración de una ruta establecida, recorriendo la misma distancia, y a una velocidad que no sobrepase a los 50 km/h. Los datos obtenidos en la modalidad diurna se detallan en la tabla 1 - 3 y la modalidad nocturna en la tabla 2 – 3 respectivamente.

Tabla 1 - 3: Tarifas marcadas por el taxímetro comercial y el prototipo en la modalidad diurna

Tarifa Diurna Arranque \$ 0,40					Taxímetro Comercial		Prototipo	
Carrera #	Hora	Duración (min)	Distancia (km)	Aceleración Prom (m/s ²)	Marcado	TOTAL	Marcado	TOTAL
1	13:22	6	2,02	2,05	\$1,08	\$1,20	\$1,07	\$1,20
2	13:41	9	3,80	1,99	\$1,57	\$1,57	\$1,56	\$1,56
3	14:02	5	2,10	2,64	\$1,18	\$1,20	\$1,18	\$1,20
4	14:35	11	4,15	3,71	\$1,98	\$1,98	\$1,97	\$1,99
5	14:56	7	3,43	2,84	\$1,45	\$1,45	\$1,44	\$1,44
6	15:09	8	3,78	2,44	\$1,75	\$1,75	\$1,73	\$1,73
7	15:20	13	4,85	1,2	\$2,36	\$2,36	\$2,36	\$2,36
8	15:35	6	3,02	3,38	\$1,33	\$1,33	\$1,34	\$1,34

9	15:50	12	4,35	3,47	\$2,10	\$2,10	\$2,10	\$2,10
10	16:02	10	3,90	2,16	\$1,67	\$1,67	\$1,65	\$1,65
11	16:19	18	2,57	1,98	\$1,71	\$1,71	\$1,72	\$1,72
12	16:25	3	1,85	1,55	\$0,99	\$1,20	\$1,00	\$1,20
13	16:34	7	3,22	2,82	\$1,43	\$1,43	\$1,45	\$1,45
14	17:00	5	2,72	2,27	\$1,28	\$1,28	\$1,27	\$1,27
15	17:16	8	3,56	1,96	\$1,67	\$1,67	\$1,66	\$1,66

Realizado por: REDROBÁN, Laura, 2018

Tabla 2 - 3: Tarifas marcadas por el taxímetro comercial y el prototipo en la modalidad nocturna

Tarifa Nocturna Arranque \$ 0,45					Taxímetro Comercial		Prototipo	
Carrera #	Hora	Duración (min)	Distancia (km)	Aceleración Prom(m/s ²)	Marcado	TOTAL	Marcado	TOTAL
1	19:37	6	3	2,7	\$1,50	\$1,50	\$1,48	\$1,48
2	19:43	2	0,77	0,65	\$0,72	\$1,30	\$0,74	\$1,30
3	19:47	7	4,10	3,92	\$2,13	\$2,13	\$2,12	\$2,12
4	19:54	9	3,80	2,54	\$1,57	\$1,57	\$1,56	\$1,56
5	20:03	2	0,72	0,89	\$0,70	\$1,30	\$0,71	\$1,30
6	20:11	7	3,43	2,81	\$1,45	\$1,45	\$1,44	\$1,44
7	20:29	11	4,15	3,79	\$1,98	\$1,98	\$1,97	\$1,99
8	20:43	4	1,75	1,24	\$1,12	\$1,30	\$1,09	\$1,30
9	20:59	3	1,13	1,12	\$0,92	\$1,30	\$0,92	\$1,30
10	21:13	4	3,84	2,38	\$1,56	\$1,56	\$1,55	\$1,55
11	21:32	8	3,90	2,26	\$1,67	\$1,67	\$1,65	\$1,65
12	21:49	5	2,72	2,19	\$1,28	\$1,30	\$1,27	\$1,30
13	22:06	7	3,22	2,82	\$1,43	\$1,43	\$1,45	\$1,45
14	22:18	10	4,85	3,04	\$2,36	\$2,36	\$2,36	\$2,36
15	22:32	8	3,78	2,76	\$1,75	\$1,75	\$1,73	\$1,73

Realizado por: REDROBÁN, Laura, 2018

3.1.1 Análisis de las tarifas en modalidad diurna

Para analizar cada una de las tarifas marcadas en las diferentes carreras realizadas por el vehículo, se realizó el cálculo del margen de error que posee cada una de estas. Se utilizó la ecuación 1 – 3 para realizar el cálculo.

Ecuación 1 – 3: Cálculo del margen de error para cada prueba.

$$\%ERROR = 100\% - \frac{Tarifa\ del\ Prototipo * 100\%}{Tarifa\ del\ comercial}$$

Una vez aplicada la ecuación 1- 3 para cada carrera realizada se obtienen los márgenes de error detallados en la tabla 3 – 3.

Tabla 3 - 3: Márgenes de error de las 15 pruebas realizada en modalidad diurna

Carrera #	Margen de Error (%)
1	0,93
2	0,64
3	0,00
4	0,51
5	0,69
6	1,14
7	0,00
8	0,75
9	0,00
10	1,20
11	0,58
12	1,01
13	1,40
14	0,78
15	0,60

Realizado por: REDROBÁN, Laura, 2018

En el gráfico 1 -3 se visualiza todos los márgenes de error en la modalidad nocturna, donde se notó que el margen oscila entre 0% y 1,4%.



Gráfico 1 - 3: Porcentajes de error de las carreras en modalidad diurna
Realizado por: REDROBÁN, Laura, 2018

3.1.2 Análisis de las tarifas de modalidad nocturna

Para analizar cada una de las tarifas marcadas en las diferentes carreras realizadas por el vehículo, se realizó una regla de tres entre el valor marcado por el taxímetro comercial con respecto al prototipo, así se calculó su efectividad y porcentaje error que representa. Para lo cual se realizaron 15 pruebas.

Una vez aplicada la ecuación 1- 3 para cada carrera realizada se obtienen los márgenes de error detallados en la tabla 4 – 3.

Tabla 4 - 3: Márgenes de error de las 15 pruebas realizada en modalidad nocturna.

Carrera #	Margen de Error (%)
1	1,33
2	2,78
3	0,47
4	0,64
5	1,43
6	0,69
7	0,51
8	2,68
9	0,00
10	0,64
11	1,20
12	0,78
13	1,40
14	0,00
15	1,14

Realizado por: REDROBÁN, Laura, 2018

En el gráfico 2 -3 se visualiza todos los márgenes de error en la modalidad nocturna, donde se notó que el margen oscila entre 0% y 2,58%.

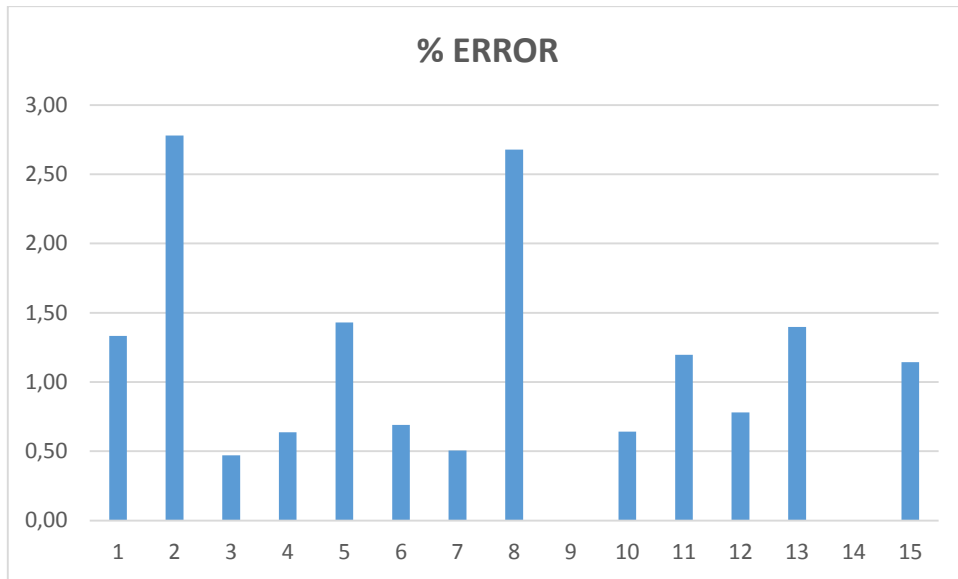


Gráfico 2 - 3: Porcentajes de error de las carreras en modalidad nocturna
Realizado por: REDROBÁN, Laura, 2018

3.2.3 Efectividad del prototipo de control y tarifación

Una vez realizado el análisis y a la vez encontrado el margen de error que tiene el prototipo con respecto al taxímetro comercial en cada carrera realizada, se procedió a realizar una prueba de efectividad con los resultados encontrados. Para este caso se utilizó la media aritmética.

Los márgenes de error existentes en cada recorrido se detallan en la tabla 5 - 3.

Tabla 5 - 3: Márgenes de errores obtenidos en cada carrera realizada

Carrera #	Margen de Error (%)
1	0,93
2	0,64
3	0,00
4	0,51
5	0,69
6	1,14
7	0,00
8	0,75
9	0,00
10	1,20
11	0,58
12	1,01
13	1,40

14	0,78
15	0,60
16	1,33
17	2,78
18	0,47
19	0,64
20	1,43
21	0,69
22	0,51
23	2,68
24	0,00
25	0,64
26	1,20
27	0,78
28	1,40
29	0,00
30	1,14

Realizado por: REDROBÁN, Laura, 2018

Para obtener el porcentaje general de error los datos de la tabla 5 – 3 se los remplazó en la ecuación 2 – 3.

Ecuación 2 – 3: Media Aritmética

$$\mu = \frac{\sum xi}{N}$$

$$\%ERROR = \frac{25,8}{30}$$

$\%ERROR = 0,86.\%$ Efectivo

Luego del cálculo del error general que posee el prototipo para el control y tarifación del servicio de taxi con respecto al taxímetro comercial instalado en la unidad que presta este servicio es de 0,86% lo que representa un valor bajo, es decir el prototipo es igual de efectivo que los dispositivos comerciales ya que tiene un margen de error bajo.

3.2 Comparación de costos entre los taxímetros comerciales y el prototipo implementado.

En el país existen varios modelos de taxímetro que son reconocidos por la Agencia Nacional de Transito, para el análisis de comparación de precios se tomó los taxímetros que son utilizados más comúnmente en el ámbito del transporte público. Los modelos con sus respectivos precios se detallan en la tabla 6 - 3.

Tabla 6 - 3: Costo de los taxímetros reconocidos por la ANT y el SRI

N°	Marca	Modelo	Costo (Dólares)	Costo %
1	CENTRODYNE	S700	160	57,14
2	SEMSA	PLATINUM	180	64,29
3	LAKEDRIVER	SLIM 800	140	50,00
4	INTELLIGENT TAXIMETER	D10	200	71,43
5	TAXSYM	M-09	230	82,14
6	DIGITAX 550	M-07	170	60,71
7	DIGITAX 550	M11	180	64,29
8	FUL MAR	TANGO XP	280	100,00
9	PROTOTIPO		100	35,71

Realizado por: REDROBÁN, Laura, 2018

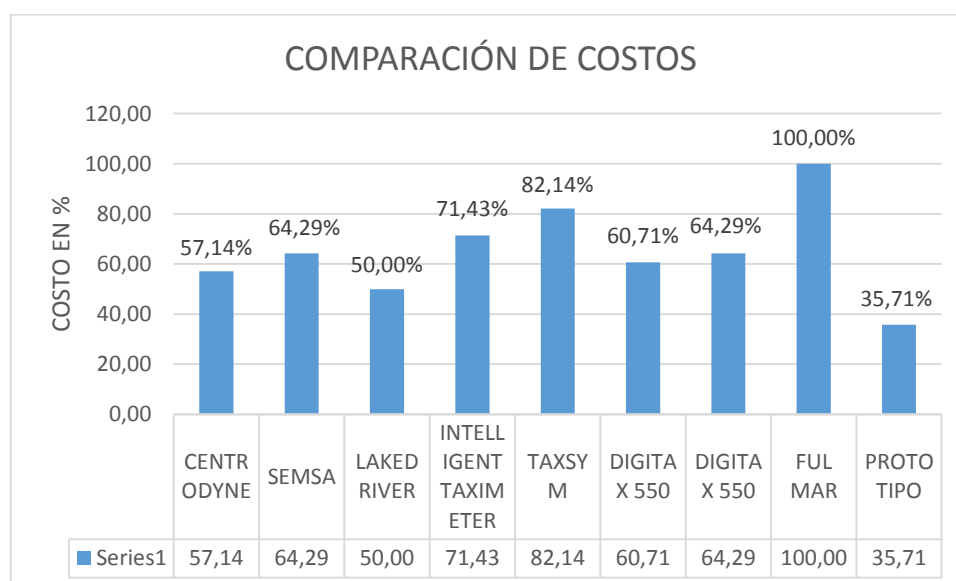


Gráfico 3 – 3: Comparación de costos entre taxímetros comerciales y el prototipo realizado
Realizado por: REDROBÁN, Laura, 2018

Según el gráfico 3 – 3 el prototipo diseñado e implementado para el control y tarificación del servicio de taxis, presenta el menor costo monetario con un 35,71% en relación a los taxímetros comerciales, esto se debe a la utilización de elementos de bajo costo pero con mayor exactitud al momento de realizar las mediciones.

3.3 Comparación de requerimientos y parámetros de tarificación entre taxímetro comercial y prototipo diseñado.

En la tabla 7 – 3 se detalla los requisitos impuestos por la ANT, SRI, y los parámetros de tarificación impuestos por el GAD Municipal de Riobamba con los que cumple el prototipo.

Tabla 7 - 3: Parámetros y requisitos con los que cumple el prototipo.

Requisitos		Parámetros de Tarificación	
Incluye un reloj	✓	Tarifa diurna	✓
Contador de eventos	✓	Tarifa nocturna	✓
Interfaz de usuario	✓	Monto de arranque	✓
Parámetros programables	✓	Kilómetro recorrido	✓
Número de identificación del taxi	✓	Minuto de espera	✓
Sensor que proporcione datos para el cálculo de distancia	✓	Carrera mínima	✓
Programa legalmente relevante			

Realizado por: REDROBÁN, Laura, 2018

El prototipo diseñado cumple con los parámetros que se detallan en la norma INEN 2336, exceptuando el requisito del programa legalmente relevante ya que este es válido cuando es revisado minuciosamente por las entidades competentes para su posterior legalización.

CONCLUSIONES

- En nuestro país existen taxímetros que realizan la tarificación por medio de un odómetro lo que facilita la alteración del cálculo, por lo cual se diseñó e implementó un prototipo para el control y tarificación del servicio de taxi con sensores inerciales, que cumpla con las mismas funciones que un taxímetro comercial existente en el país, a bajo costo y fiable.
- Del análisis de las tecnologías existentes en el país, se realiza el estudio, diseño e implementación de un prototipo de control y tarificación con sensores inerciales, que por sus características son considerados como unos de los sensores más exacto existentes en el mercado a bajo costo y fácil de adquirirlo en el país.
- En el servicio de taxi una de los procesos más importantes es el cálculo del costo exacto del recorrido realizado al usuario, el dispositivo que realiza esta función tiene un costo elevado para los prestadores del servicio por lo que se diseña un prototipo mucho más económico y con las mismas prestaciones.
- La utilización de la tarjeta de desarrollo arduino uno en el diseño del prototipo para el control y tarificación del servicio de taxi con sensores inerciales ha sido de gran importancia, ya que por sus características de software como hardware permiten reprograma el prototipo las veces necesarias para cumplir con los requerimientos de los usuarios.
- Al realizar el diseño del circuito de conexión en el entorno Ares del software Proteus, y por sus características de construcción permite la conexión entre los distintos componentes del prototipo de una manera ordenada, lo que permite que sea un dispositivo transportable y de fácil manipulación por su reducido tamaño.
- El módulo MPU 60 – 50 sensa un valor de 0 m/s^2 significa que el vehículo está parado o alcanza una velocidad constante, por lo que se utiliza una señal extra que permite identificar la posición del vehículo.
- Del análisis de las pruebas del prototipo realizadas en las dos modalidades tanto diurna y nocturna en diferentes velocidades respectivamente, se obtuvo un error del 0,86% lo que está dentro del margen de error que permite el sensor inercial, teniendo un sistema estable.

- De las pruebas realizadas entre el prototipo y un dispositivo comercial se observa que el error aumenta cuando el vehículo sobrepasa los 50km/h que es la velocidad máxima de circulación permitida en la ciudad de Riobamba y normada por la Agencia Nacional de Transito, por lo que el prototipo está diseñado para este límite de velocidad.
- El prototipo de control y tarifación del servicio de taxi diseñado, tiene un valor económico de fabricación menor a los dispositivos utilizados actualmente, lo que le hace más accesible para los usuarios que desee disponer de este prototipo.

RECOMENDACIONES

- Se recomienda no exceder el límite de velocidad establecida por la ANT de 50km/h en las zonas urbanas para evitar un margen de error mayor al establecido en la tarifación al final de un recorrido.
- Al momento de calibrar el módulo MPU 60 – 50, se lo coloque en la posición y lugar donde se lo va a utilizar para obtener una correcta calibración de los valores de inicialización.
- La tarjeta de desarrollo arduino uno sea programada según los parámetros de tarifación impuestos para el prototipo desarrollado, para este caso en particular como el GAD Municipal de Riobamba.
- Se recomienda tomar en cuenta que la aceleración es cero cuando el vehículo alcanza una velocidad constante, por lo que hay que utilizar una señal extra para identificar este caso.
- Al momento de realizar las conexiones de la pantalla LCD, se lo haga con cuidado ya que al tener varios pines estos pueden causar confusión y quemar la pantalla.
- Se recomienda utilizar un regulador de voltaje que proporcione una alimentación entre 3,3V a 5V para el correcto funcionamiento del prototipo, evitando daños irreparables en los dispositivos del prototipo.

BIBLIOGRAFIA

Acelerometro, Giroscopios, IMU. [En línea] 2013. [Citado el: 27 de Diciembre de 2017.] Recuperado de : <https://www.coldfire-electronica.com/esp/item/154/55/acelerometro-giroscopio-6-dof-imu-gy-521-mpu-6050>.

AGENCIA NACIONAL DE TRANSITO, ECUADOR. *Inventario Tecnológico de Dispositivos y Sistemas de Control. Sems platinum.* [En línea]. 2013 [Citado el: 12 de Octubre de 2017.] Disponible en: <http://www.ant.gob.ec/index.php/descargable/file/415-sems-platinum>.

AGENCIA NACIONAL DE TRANSITO, ECUADOR. *Inventario tecnológico de dispositivos y sistemas de control. Intelligent taximeter.* [En línea] 2013. [Citado el: 12 de Octubre de 2017.] Disponible en: <http://www.ant.gob.ec/index.php/descargables/file/413-intelligent-taximeter-d10>. 1.

Aprendiendo Arduino. I2C. [En línea] 2016. [Citado el: 10 de Febrero de 2018.] Recuperado de: <https://aprendiendoarduino.wordpress.com/category/i2c/>.

ARDUINO. *Arduino.* [En línea] 2018. [Citado el: 10 de Febrero de 2018.]. Recuperado de: <https://www.arduino.cc/en/Main/Software#>.

CABYFY. *Funcionamiento de CABYFY.* [En línea] 2017. [Citado el: 18 de Marzo 2018]. Recuperado de: <https://www.autocasion.com/actualidad/reportajes/como-funciona-cabify-el-chofer-a-domicilio>

CENTRODYNE. *Taximetro S700.* [En línea] 2013. [Citado el: 2017 de OCTUBRE de 2017.] Recuperado de : <http://www.centrodyne.com/es/products/taximeters/s700>. 1.

GOBIERNO AUTOMONO DECENTRALIZADO MUNICIPAL DE RIOBAMBA. *Ordenanza 009-2016. Ordenanza para el fortalecimiento de la calidad y la fijación de la tarifa por la prestación del servicio de transporte terrestre comercial en taxi del cantón riobamba.* [En línea] 2016.[Citado el: 17 de Agosto de 2017.] Disponible en: <http://www.gadmriobamba.gob.ec/index.php/transparencia/lotaip2016/ordenanzas-2016?download=658:ordenanza-009-2016-tarifas-taxis>.

INEN 2336. *Normas Técnicas Taxímetros.*

InvenSense. *MPU-6000 and MPU-6050 Product Specification.* [En línea] 2013. [Citado el: 10 de Febrero de 2018.] Disponible en: https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf. 3.4.

LLAMAS, LUIS. *Ingeniería, Informática y Diseño. Filtro Paso Bajo y Paso Alto en Arduino.* [En línea] 2017. [Citado el: 17 de Febrero de 2018.] Recuperado de: <https://www.luisllamas.es/medir-la-inclinacion-imu-arduino-filtro-complementario/>

MARTINEZ, RIGOBERTO. *Uso de sensores inerciales en la medición y evaluación de movimiento humano para aplicaciones en la salud.* [En línea] 2013. [Citado el: 26 de Diciembre de 2017.] Disponible en: https://www.researchgate.net/profile/Rigoberto_Martinez_Mendez/publication/263198667_Uso_de_sensores_inerciales_en_la_medicion_y_evaluacion_de_movimiento_humano_para_aplicaciones_en_la_salud/links

NAYLAMP MECHATRONICS. *MPU6050, Acelerómetro y Giroscopio.* [En línea] 2016. [Citado el: 10 de Febrero de 2018.] Recuperado de: http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html.

NITAX. *Taxímetro Inteligente.* [En línea] 2016. [Citado el: 20 de Marzo de 2018] Recuperado de: <http://www.nitax.net/taxiacutemetros.html>

ORGANIZACIÓN INTERNACIONAL DE METROLOGÍA LEGAL. *Errores máximos permitidos en los taxímetros.* [En línea] 2009. [Citado el: 20 de Marzo de 2018] Recuperado de: <https://www.facua.org/es/noticia.php?Id=4375>

PROMETEC. *DISPLAY LCD. Usando displays de texto LCD 16x2 con Arduino.* [En línea] 2018. [Citado el: 22 de 02 de 2018.] Recuperado de: <https://www.prometec.net/displays-lcd/#>.

REYES, CARLOS. *Microcontroladores Programación en BASIC.* Quito - Ecuador : Rispergraf, 2006. Segunda Edición. p.79

SALAS, JOSE *Memorias de la LCD.* [En línea] 2013. [Citado el: 30 de Diciembre de 2017.] Recuperado de: <http://todoelectrodo.blogspot.com/2013/02/lcd-16x2.html>.

Sensor MPU - 6050. [En línea] 29 de Noviembre de 2015. [Citado el: 27 de Diciembre de 2017.] Recuperado de: <https://www.hispavila.com/el-sensor-mpu-6050/>.

SILVA, EDGARDO. *Herramientas del Pensamiento Computacional*. [En línea] 2016. [Citado el: 26 de Diciembre de 2017.] Recuperado de: <https://edgardosilvi.wordpress.com/2016/02/29/acamica-ventajas-y-desventajas-de-arduino/>.

SERVICIO DE RENTAS INTERNAS, ECUADOR *Taxímetros calificados y autorizados por el Servicio de Rentas Internas. Taxímetros autorizados y homologados*. [En línea] 2014. [Citado el: 12 de Octubre de 2017.] Recuperado de: <http://www.sri.gob.ec/DocumentosAlfrescoPortlet/descargar/423a4d99-9f48-4838-b281-63fa261c0abb/LISTADO+DE+TAX%CDMETROS+octubre+2014.pdf>. 1.

TAXIMETER. *Funcionamiento del taxímetro*. [En línea] 2017. [Citado el: 17 de Diciembre de 2017.] Recuperado de: <https://examentaxi.com/funcionamiento-de-un-taximetro>.

THAYER, LUIS. *MCI Electronics. Arduino*. [En línea] 2017. [Citado el: 10 de Febrero de 2018.] Recuperado de: <http://arduino.cl/arduino-uno/>.

TORRENTE, OSCAR. *ARDUINO Curso Práctico de Formación*. Mexico : Alfaomega Grupo Editor, 2013. Primera Edición. pp.63-64

TOYOTA. *Cómo funciona el control de velocidad de cruceo y cuándo usarlo para ganar en seguridad*. [En línea] 2016. [Citado el: 14 de Marzo de 2018.] Recuperado de: <https://www.motorpasion.com/espaciotoyota/como-funciona-el-control-de-velocidad-de-cruceo-y-cuando-usarlo-para-ganar-en-seguridad>.

Tutorial de Arduino y MPU 60-50. [En línea] 2014. [Citado el: 15 de Febrero de 2018.] Recuperado de: <https://robologs.net/2014/10/15/tutorial-de-arduino-y-mpu-6050/>.

UBER. *Funcionamiento de UBER*. [En línea] 2016. [Citado el: 20 de Marzo de 2018] Recuperado de: <https://www.uber.com/es-EC>

VALENZUELA, PEDRO. *ReleabilityWeb. Movimiento Acelerado*. [En línea] 2016. [Citado el: 21 de Febrero de 2018.] Recuperado de: <https://sites.google.com/site/pielagosfq4/home/aceleracion/-que-distancia-se-recorre-con-un-movimiento-acelerado>.

VARA, DAVID. *Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones. COMPONENTES DE UNA IMU*. [En línea] 2014. [Citado el: 27 de Diciembre de 2017.] Disponible en: <https://uvadoc.uva.es/bitstream/10324/12884/1/TFG-P-161.pdf>.

ANEXOS

Anexo A: DataSheet del MPU60-50 ó IMU GY-521



SYNACORP TRADING & SERVICES

No.9, 1st Floor, Lrg 1/SS2, Bandar Tasek Mutiara, 14120 Simpang Ampat, 5.Prai (S), Penang
Tel : +604.504.1617 Hunting Line : 012.4033.474 Fax : +604.502.1726
(Website) <http://www.synacorp.my> (Email) sales@synacorp.com.my

MPU 6050 GY-521 3 Axis Gyro Accelerometer Sensor Module Arduino



The MPU-6050 sensor module contains an accelerometer and a gyro in a single chip. It is very accurate, as it contains 16-bits analog to digital conversion hardware for each channel. Therefore it captures the x, y, and z channel at the same time. The sensor uses the I2C-bus to interface with the Arduino.

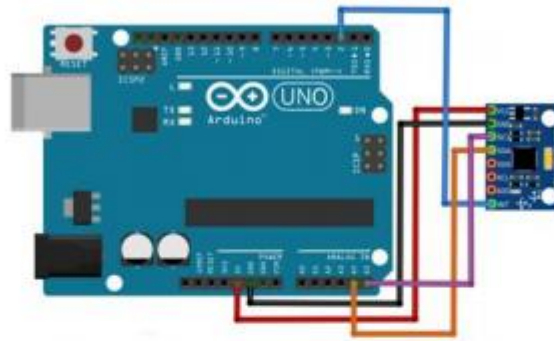
Application:

- Motion-enabled game and application framework
- Location based services, points of interest
- Handset and portable gaming
- Motion-based game controllers
- Wearable sensors for health, fitness and sports
- Toys

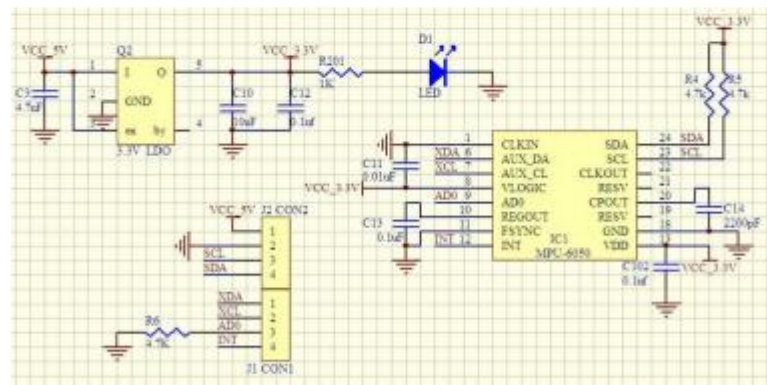
Features:

- Use the chip: MPU-6050.
- Power supply: 3-5v (internal low dropout regulator).
- Communication modes: standard IIC communications protocol.
- Chip built-in 16bit AD converter, 16-bit data output.
- Immersion Gold PCB machine welding process to ensure quality.
- Tri-Axis angular rate sensor (gyro) with a sensitivity up to 131 LSBs/dps and a full-scale range of ± 250 , ± 500 , ± 1000 , and ± 2000 dps
- Tri-Axis accelerometer with a programmable full scale range of $\pm 2g$, $\pm 4g$, $\pm 8g$ and $\pm 16g$

Pin Assignment:



Schematic:



Anexo B: Programa de Calibración del Módulo MPU 60-50

```
#include "I2Cdev.h"

#include "MPU6050.h"

#include "Wire.h"

int ax, ay, az;

int ax_o, ay_o, az_o;

void setup() {

  Serial.begin(57600);

  Wire.begin();

  sensor.initialize();

  if (sensor.testConnection()) Serial.println("Sensor iniciado correctamente");

  ax_o=sensor.getXAccelOffset();

  ay_o=sensor.getYAccelOffset();

  az_o=sensor.getZAccelOffset();

  Serial.println("Offsets:");

  Serial.print(ax_o); Serial.print("\t");

  Serial.print(ay_o); Serial.print("\t");

  Serial.print(az_o); Serial.print("\t");

  Serial.println("\n\nEnvie cualquier caracter para empezar la calibracionnn");

  Serial.println("Calibrando, no mover IMU");
```

```
}  
  
void loop() {  
  
  sensor.getAcceleration(&ax, &ay, &az);  
  
  if (counter==100){  
  
    Serial.print(p_ax); Serial.print("\t");  
  
    Serial.print(p_ay); Serial.print("\t");  
  
    Serial.println(p_az); //Serial.print("\t");  
  
  
    if (p_ax>0) ax_o--;  
  
    else {ax_o++;}  
  
    if (p_ay>0) ay_o--;  
  
    else {ay_o++;}  
  
    if (p_az-16384>0) az_o--;  
  
    else {az_o++;}  
  
  
    sensor.setXAccelOffset(ax_o);  
  
    sensor.setYAccelOffset(ay_o);  
  
    sensor.setZAccelOffset(az_o);  
  
  }  
  
  counter++;  
  
}
```

Anexo C: Programa del Cambio de Unidades y Diseño de Filtro

```
#include "I2Cdev.h"
#include "MPU6050.h"
#include <TimerOne.h>
#include <LiquidCrystal.h>
#define pulsador 4

LiquidCrystal lcd(7, 8, 9, 10, 11, 12);

#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
    #include "Wire.h"
#endif

MPU6050 accelgyro;

int ss=00;
int mm=45;
int hh=19;

int16_t ax, ay, az;
int16_t gx, gy, gz;
int d=0;
int t=0;
int T=45;
int x=0;
int d1=0;
int t1=0;
int T1=0;

#define OUTPUT_READABLE_ACCELGYRO
#define LED_PIN 13
bool blinkState = false;

void setup() {
    lcd.begin(16,2);
```

```

pinMode (pulsador,INPUT);
#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
    Wire.begin();
#elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
    Fastwire::setup(400, true);
#endif

Serial.begin(38400);

Serial.println("Initializing I2C devices...");
accelgyro.initialize();

Serial.println("Testing device connections...");
Serial.println(accelgyro.testConnection() ? "MPU6050 connection successful" : "MPU6050
connection failed");
pinMode(LED_PIN, OUTPUT);
}

void loop() {

    accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);

#ifdef OUTPUT_READABLE_ACCELGYRO

    Serial.print("a:\t");
    Serial.print(ax); Serial.print("\t");
    Serial.print(ay); Serial.print("\t");
    Serial.println(az);
#endif

#ifdef OUTPUT_BINARY_ACCELGYRO

    Serial.write((uint8_t)(ax >> 8)); Serial.write((uint8_t)(ax & 0xFF));
    Serial.write((uint8_t)(ay >> 8)); Serial.write((uint8_t)(ay & 0xFF));

```



```
Serial.write((uint8_t)(az >> 8)); Serial.write((uint8_t)(az & 0xFF));

#endif

blinkState = !blinkState;
digitalWrite(LED_PIN, blinkState);

lcd.setCursor(0 ,0);
lcd.print("COOP.SAN_NICOLAS");

lcd.setCursor(2 ,1);
lcd.print("UNIDAD");

lcd.setCursor(9 ,1);
lcd.print("10-24");
delay(1000);
lcd.clear();
inicio=digitalRead(pulsador);
fin=digitalRead(pulsador);
fin=1-fin;

if( inicio && fin ==0 )
if((hh>6)&&(hh<19))

for (ss = 0; ss < 61; ss++){
inicio=0;

fin=digitalRead(pulsador);
if(fin==0){

ss=ss;
t=t+1;

if(t>3)
{
```

```
t=0;
diu=diu+1;
}
```

```
if(diu>99){
    diu=0;
    dol=dol+1;
}
if(ss > 59)
{
    ss = 00;
    mm=mm+1;
}
```

```
if(mm > 59)
{
    mm=00;
    hh=hh+1;
}
```

```
if(hh >= 24)
{
    hh=00;
    mm=00;
    ss=00;
}
```

```
lcd.clear();
lcd.setCursor(0 ,0);
lcd.print("Hora");
```

```
lcd.setCursor(5 ,0);
lcd.print(hh);
```

```
lcd.setCursor(7 ,0);
lcd.print(":");
```

```
lcd.setCursor(8 ,0);
```

```
lcd.print(mm);
```

```
lcd.setCursor(0 ,1);
```

```
lcd.print("PAGO_TOTAL:");
```

```
lcd.setCursor( 11,1);
```

```
lcd.print(dol);
```

```
s=dol;
```

```
lcd.setCursor(12 ,1);
```

```
lcd.print(".");
```

```
lcd.setCursor(13 ,1);
```

```
lcd.print(diu);
```

```
delay(1000);
```

```
}
```

```
if(fin==1){
```

```
fin=digitalRead(pulsador);
```

```
if(fin==0){
```

```
    // read raw accel/gyro measurements from device
```

```
    #if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
```

```
        Wire.begin();
```

```
    #elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
```

```
        Fastwire::setup(400, false);
```

```
    #endif
```

```
MPU6050 accelgyro;
```

```
Serial.begin(38400);
```

```
accelgyro.initialize();
```

```
accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
```

```

if( ((ax*(9.81/16384.0))>-0.5)&&((ax*(9.81/16384.0)<0.5)))
{
    c=0;
    T=T+1;}

Serial.print("Parado:\t");

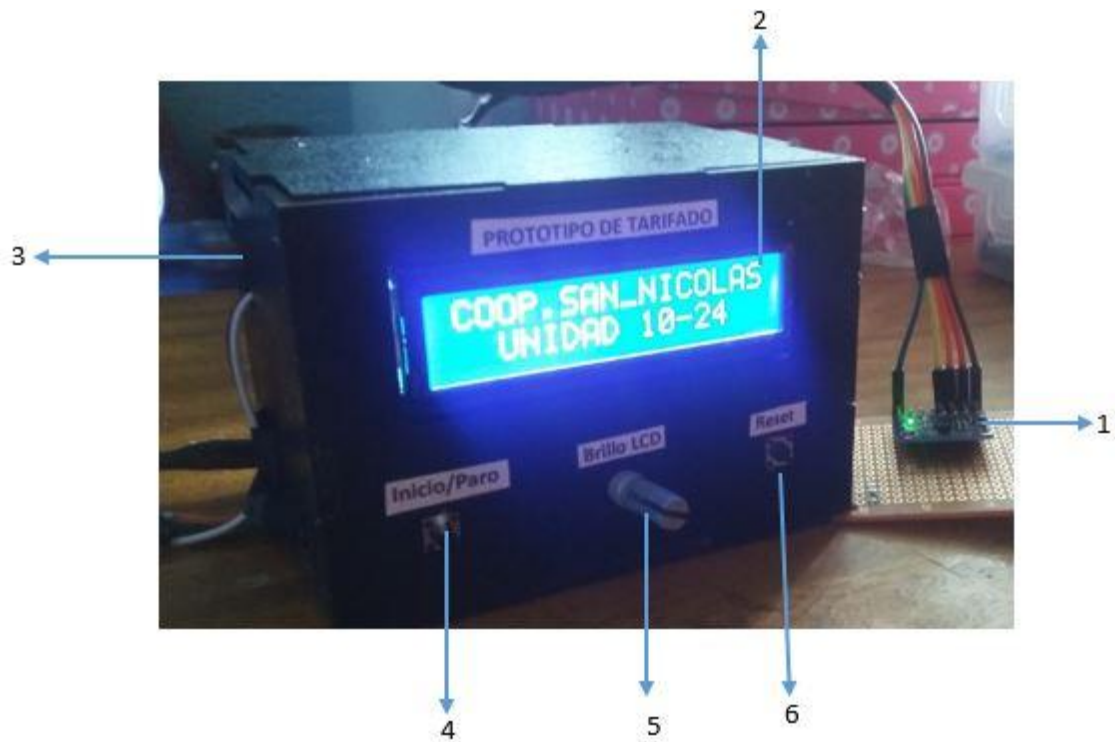
        Serial.print("00"); Serial.print("\t");
    Serial.println(T);
}
else{
    if ((ax*(9.81/16384.0))>0.5){
        t=(ax*(9.81/16384.0))/2,77;
        x=0.5*(ax*(9.81/16384.0))*t*t;
        d=d+x;
        Serial.print("a+:\t");
        Serial.print(ax*(9.81/16384.0)); Serial.print("\t");
        Serial.print(d); Serial.print("\t");
        Serial.println(T);

        if (d>31.25)
        {
            T=T+1;
            d=d-31.25;
        }

        else
        9.81/16384.0))*t1*t1;
            d1=d1-31.25;
        }
        Serial.print("a-:\t");
        Serial.print(ax*(9.81/16384.0)); Serial.print("\t");
        Serial.print(d1); Serial.print("\t");
        Serial.println(T);
    }}
delay(1000);

```

Anexo D: Partes del Prototipo de Control y Tarifación



1. Módulo MPU 60-50
2. Pantalla de cristal líquido LCD
3. Alimentación 5V
4. Pulsador de Inicio / Paro
5. Perilla para regular el brillo del pantalla LCD
6. Pulsador de Reset

Anexo E: Prueba de funcionamiento del prototipo de control y tarificación





Anexo F: Recibos de las carreras de prueba realizadas

COOPERATIVA
 COOP. SAN NICOLAS
 RIOBAMBA - ECUADOR
 RUC: 0602207318001
 Aut. SRI: 1121415559
 Vence en: 12/09/2018

REDROBAN QUIROZ GERMAN MAURICIO
 Dir: RIOBAMBA NORTE
 Tel: 0980546856
 Placa: HAA-2203 / Disco#: 1024
 Taximetro Serie: H1459
 Nombre/Razon Social: -----

CI/RUC: -----

Ticket Original No: 01135
 Fecha de emision: 04-03-19
 Hora: 20:03 a 20:05
 Concepto: TRANSPORTE.

.....Tarifa Nocturna.....
 Arranque: \$ 0.45
 Distancia: .77Km
 Tarifa \$ 0.35/Km \$ 0.27

Espera: 003 seg
 Tarifa \$ 0.06/Min \$ 0.00
 SUBTOTAL - - - \$ 0.72
 Minima: \$ 1.30
 IVA 00% \$ 000.00
 TOTAL - - - - US\$ 1.30

Intelligent Taximeter D10
 Programa Sumitrag 05-12-2017
 Vers: 1.01 2011-000
 SUMITRAG - HECHO EN ECUADOR

COOPERATIVA
 COOP. SAN NICOLAS
 RIOBAMBA - ECUADOR
 RUC: 0602207318001
 Aut. SRI: 1121415559
 Vence en: 12/09/2018

REDROBAN QUIROZ GERMAN MAURICIO
 Dir: RIOBAMBA NORTE
 Tel: 0980546856
 Placa: HAA-2203 / Disco#: 1024
 Taximetro Serie: H1459
 Nombre/Razon Social: -----

CI/RUC: -----

Ticket Original No: 01127
 Fecha de emision: 04-03-19
 Hora: 16:34 a 16:41
 Concepto: TRANSPORTE.

.....Tarifa Diurna.....
 Arranque: \$ 0.40
 Distancia: 3.22Km
 Tarifa \$ 0.32/Km \$ 1.03

Espera: 008 seg
 Tarifa \$ 0.06/Min \$ 0.00
 SUBTOTAL - - - \$ 1.43
 IVA 00% \$ 000.00
 TOTAL - - - - US\$ 1.43

Intelligent Taximeter D10
 Programa Sumitrag 05-12-2017
 Vers: 1.01 2011-000
 SUMITRAG - HECHO EN ECUADOR

COOPERATIVA
 COOP. SAN NICOLAS
 RIOBAMBA - ECUADOR
 RUC: 0502207318001
 Aut. SRI: 1121415559
 Vence en: 12/09/2018

REDROBAN QUIROZ GERMAN MAURICIO
 Dir: RIOBAMBA NORTE
 Telf: 0980546856
 Placa: HAA-2203 / Disco#: 1024
 Taximetro Serie: H1459

Nombre/Razon Social:

CI/RUC:

Ticket Original No: 01130
 Fecha de emision: 04-03/18
 Hora: 17:00 a 17:05
 Concepto: TRANSPORTE.
Tarifa Diurna.....

Arranque:		\$	0.40
Distancia:	2.72Km		
Tarifa \$	0.32/Km	\$	0.87
Espera:	017 seg		
Tarifa \$	0.05/Min	\$	0.01
SUBTOTAL		\$	1.28
IVA 00%		\$	000.00
TOTAL		US\$	1.28

Intelligent Taximeter OIG #
 Programa Sumitrag 09-12-2017
 INEVI FVM 2011-004
 SUMITRAG - HECHO EN ECUADOR

COOPERATIVA
 COOP. SAN NICOLAS
 RIOBAMBA - ECUADOR
 RUC: 0502207318001
 Aut. SRI: 1121415559
 Vence en: 12/09/2018

REDROBAN QUIROZ GERMAN MAURICIO
 Dir: RIOBAMBA NORTE
 Telf: 0980546856
 Placa: HAA-2203 / Disco#: 1024
 Taximetro Serie: H1459

Nombre/Razon Social:

CI/RUC:

Ticket Original No: 01123
 Fecha de emision: 02-03/18
 Hora: 16:02 a 16:12
 Concepto: TRANSPORTE.
Tarifa Diurna.....

Arranque:		\$	0.40
Distancia:	3.90Km		
Tarifa \$	0.32/Km	\$	1.24
Espera:	033 seg		
Tarifa \$	0.05/Min	\$	0.03
SUBTOTAL		\$	1.67
IVA 00%		\$	000.00
TOTAL		US\$	1.67

Intelligent Taximeter OIG #
 Programa Sumitrag 09-12-2017
 INEVI FVM 2011-004
 SUMITRAG - HECHO EN ECUADOR