

Implementación de un sistema GPS para la verificación de tiempos de viaje de un autobús.

Autor-Jeferson PORTILLA¹, Coautor-Fredy ROSERO²

^{1, 2} Carrera de Ingeniería en Mantenimiento Automotriz, FICA, Universidad Técnica del Norte, Av. 17 de Julio, Ibarra, Imbabura, Ecuador

jrportillay_@hotmail.com, farosero@utn.edu.ec

Resumen. *En la actualidad se puede observar que las diferentes cooperativas de transporte interprovincial controlan el tiempo de viaje de cada una de sus unidades únicamente definiendo la hora de salida y llegada a su destino, provocando disgusto en los usuarios.*

El dispositivo es desarrollado a través de la plataforma de hardware libre Arduino, en el que mediante la programación es posible recolectar información brindada durante su funcionamiento, permitiendo que las autoridades pertinentes tengan conocimiento de los lugares en los que el autobús realiza esperas y adelantos, y de esta manera imponer la respectiva sanción al responsable del vehículo. Además, con el registro de la velocidad máxima y mínima de viaje del autobús se podrá brindar mayor seguridad a los usuarios y al chofer de la respectiva unidad.

La hora de cruce por los diferentes lugares de control son registrados a través del funcionamiento del módulo GPS instalado en el dispositivo, tomando en cuenta valores de longitud y latitud. Para realizar un análisis estadístico se desarrolla un programa de análisis de datos en el que se observa información de tiempos totales de adelantos y atrasos, como también información de cada viaje realizado en un determinado tiempo, además de registrar las velocidades máximas y mínimas y de su respectiva representación gráfica a través del programa Power BI.

Palabras Claves

Transporte Interprovincial, GPS, Arduino.

Abstract. *Years ago and even today it can be seen that the different interprovincial transport cooperatives control the travel time of each of its units only by defining the time of departure and arrival at destination, causing that users do not get good service because they do not know the specific time at which the vehicle must pass through different places located between the route to follow, this is because they not have any control device that allows them determine those*

different times. Also, by not having any of those systems it is not possible to record the trips of each bus to allow us to obtain statistical information to determine the time and waiting places made in the different trips. The device is developed through the free hardware platform Arduino in which by programming it is possible to collect information provided during its operation, allowing that the relevant authorities be aware of the places where the bus takes pauses and advances. In this way impose the respective sanction to the responsible of the vehicle. In addition, by registering the maximum and minimum travel speed of bus it may provide greater security to users and the driver of the respective unit. The crossing different control sites are registered through the operation of the GPS module installed on the device, taking into account latitude and longitude values. To perform the statistician analysis it is developed a data analysis program in which it is observed the Total information related to advances and delays, as well as information of each trip in a certain time apart from recording the maximum and minimum speeds and its respective graphical representation through the Power BI program.

Keywords

Interprovincial Transportation, GPS, Arduino.

1. Introducción

Desde hace años atrás las diferentes cooperativas de transporte interprovincial controlan el tiempo de viaje de cada una de sus unidades únicamente definiendo la hora de salida y llegada a su destino, lo que hace que los usuarios no obtengan un buen servicio debido a que no conocen la hora específica a la que debe pasar el vehículo a través de los diferentes lugares ubicados entre la ruta a seguir.

En la actualidad no se encuentran dispositivos que se hayan desarrollado para mejorar este tipo de control y así

brindar un mejor servicio a los usuarios de las diferentes cooperativas. Además, al no contar con alguno de estos sistemas no es posible registrar los viajes realizados por el vehículo que nos permita obtener información estadística para conocer el tiempo y los lugares de espera que realizan los autobuses en los diferentes viajes.

El trabajo de grado se realiza en la Universidad Técnica del Norte y el área de aplicación es en las unidades de transporte interprovincial de la ciudad de Ibarra.

Con la incorporación de un dispositivo que permita controlar el tiempo de viaje de los autobuses de las diferentes cooperativas de transporte interprovincial, los choferes podrán conocer en tiempo real la velocidad y tiempo de atraso o adelanto en los diferentes lugares situados entre la ruta de cada viaje y así brindar un mejor servicio a sus usuarios debido a que podrán conocer el tiempo de llegada a su destino.

El presente trabajo de grado tiene como principal objetivo el diseño de un dispositivo para el control de tiempo de viaje de un autobús, que está desarrollado a través de la plataforma de hardware libre Arduino y de un módulo GPS con el que se registra la hora de cruce del vehículo por los diferentes lugares de control.

2. Materiales y Métodos

La metodología aplicada es la siguiente:

- **Diseño electrónico.** Elaboración de diagramas electrónicos para conocer la forma de conexión de los diferentes elementos utilizados para la elaboración del dispositivo.
- **Optimización.** Mejoras en el funcionamiento del dispositivo para lograr con éxito los objetivos planteados para el trabajo de grado.
- **Recolección de datos.** Realizar viajes a manera de simulación para el almacenamiento de datos y su respectivo análisis en el programa LabVIEW.
- **Adaptación.** Adaptación del GPS, pantalla y módulo Arduino para lograr un trabajo en conjunto y de esta manera mostrar la información respectiva y realizar un correcto almacenamiento de datos.

Las técnicas e instrumentos aplicados son:

- **Análisis de datos.** Realizar el análisis de datos a través de la información almacenada en la tarjeta de memoria del dispositivo durante los viajes realizados por el autobús, para de esta manera poder realizar mejoras en su servicio.

- **Pruebas de funcionamiento.** Realizar pruebas de funcionamiento a través de las diferentes rutas asignadas en la programación del dispositivo para así verificar su utilidad y forma de funcionamiento.

Los materiales utilizados en el diseño del dispositivo son los siguientes; Arduino mega, pantalla táctil de 3.2", módulo GPS.

2.1 Etapas de diseño

Para el diseño del dispositivo se utilizan seis etapas necesarias para conseguir los objetivos planteados que son: diseño del dispositivo, selección de elementos electrónicos, diagramas, conexiones, programación de Arduino, desarrollo del software y pruebas de funcionamiento; cada una de ellas contienen actividades específicas como se indica en el diagrama de la figura 1.



Figura. 1. Etapas de diseño.

2.2 Implementación de hardware y software de Arduino.

La conexión del módulo GPS en la placa Arduino Mega se realiza a través de un shield como se indica en la figura 2. Este módulo es conectado a la comunicación serial I (RX1, TX1) de Arduino para la obtención de los datos proporcionados por el módulo GPS. La conexión de pines se realiza como se detalla a continuación:

- Rx GPS → Tx1 Arduino.
- Tx GPS → Rx1 Arduino.
- GND GPS → GND Arduino.
- 5V GPS → 5V Arduino.

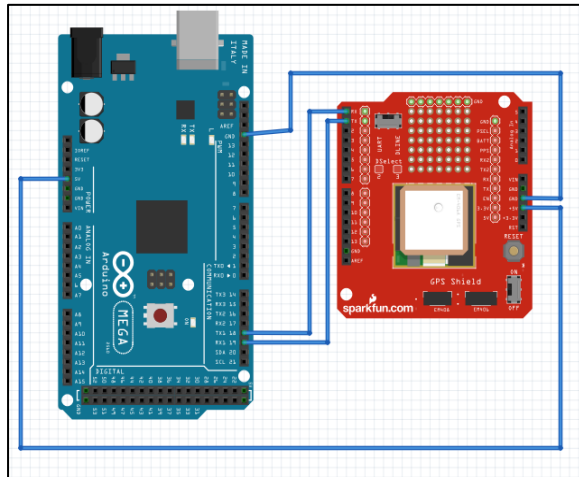


Figura 2. Diagrama de conexión módulo GPS.

El funcionamiento en conjunto del módulo GPS y de la placa Arduino permite establecer las áreas para los lugares de control del tiempo real a la que pasa el autobús. Esto se logra gracias a la biblioteca instalada y a las respectivas instrucciones del código programado de la figura 3, que con los valores de longitud y latitud permiten limitar las áreas para el control de tiempo.

```

codigo2$
512 //TIEMPO REAL 1
513 if(3615>=LAT && LAT>=3602) //IBAPPA TERMINAL
514 {
515     if(781124>=LONG && LONG>=781114)
516     {
517         posicion1++;
518         if(posicion1==1)
519         {
520             horall=h;
521             minutoll=Bus.time.minute();
522             tiempo2=tiempo;
523             diferencial=(3-tiempo2)/60;
524             myGLCD.setFont(BigFont);
525             myGLCD.setBackgroundColor(VGA_LIME);
526             myGLCD.setColor(VGA_BLUE);
527             myGLCD.printNumI(diferencial, 235, 145);
528             if(diferencial<0)
529             {
530                 horal=horall;
531                 minutol=minutoll-abs(diferencial);
532             }
533             else if(diferencial>=0)
534             {
535                 horal=horall;
536                 minutol=minutoll+diferencial;
537             }
538         }
539     }

```

Figura 3. Código limitación de áreas de control.

La velocidad, hora y fecha actual que se indica en la pantalla del dispositivo durante su funcionamiento son

adquiridas a través de los comandos de la librería <TinyGPS++.h>. Para obtener la hora en Ecuador es necesario restar cinco horas a la proporcionada por el GPS debido a que este posee la hora del meridiano cero, tal como se indica en el código de la figura 4.

```

codigo2$
452 int h= Bus.time.hour();
453 h= h-5;//restamos cinco horas para la obtener la hora del ecuador
454 if (h<0)
455 {
456     h= h+24;
457 }
458 long LONG=abs(Bus.location.lng())*10000;
459 long LAT=abs(Bus.location.lat())*10000;
460 tiempo++; //para tiempo de viaje
461 Serial.print(tiempo);
462 myGLCD.setFont(SevenSegNumFont);
463 myGLCD.setBackgroundColor(VGA_BLUE);
464 myGLCD.setColor(VGA_WHITE);
465 myGLCD.printNumI(Bus.speed.kmph(), 140, 30);
466 int velocidad=Bus.speed.kmph();
467 if(Bus.speed.kmph()<2)
468 {
469     velocidad=0;
470 }
471 myGLCD.printNumI(velocidad, 140, 30);
472 if(velocidad<10)
473 {
474     myGLCD.setColor(VGA_BLUE);
475     myGLCD.fillRoundRect(170, 30, 230, 80);
476 }

```

Figura 4. Código de velocidad, hora y fecha.

En la pantalla táctil de 3.2” se indican datos en tiempo real según el funcionamiento del dispositivo, además, dispone de una ranura de tarjeta SD para el respectivo almacenamiento de datos. Para su correcto funcionamiento se realiza la conexión de pines para el control de la pantalla, control táctil, tarjeta SD, tal como se indica en la figura 5; para el control de datos de la pantalla se realiza la conexión de los siguientes pines:

- LCD_RS → D38
- LCD_WR → D39
- LCD_RD → 3.3V
- LCD_CS → D40
- LCD_RST → D41

La TFT tiene incorporada una pantalla táctil de cuatro hilos y la interfaz necesaria para su funcionamiento es como se indica a continuación:

- Pin 34 (DCLK) al pin D6 de Arduino.
- Pin 33 (T_CS) al pin D5 de Arduino.
- Pin 30 (MOSI) al pin D4 de Arduino.
- Pin 29 (MISO) al pin D3 de Arduino.
- Pin 31 (T_PEN) al pin D2 de Arduino.

El módulo SD de la pantalla puede ser utilizada para mostrar imágenes en la pantalla almacenadas en su interior, como también para almacenar datos importantes registrados durante el funcionamiento del dispositivo. Para mostrar las imágenes es necesario transformarlas en formato RAW. La interfaz necesaria para su funcionamiento es como se indica a continuación:

- Pin 34 (SCK) al pin D52 de Arduino.
- Pin 22 (SD_CS) al pin D53 de Arduino.
- Pin 29 (MISO) al pin D50 de Arduino.
- Pin 30 (MOSI) al pin D51 de Arduino.

La pantalla táctil comparte los pines MOSI y MISO con la tarjeta SD, que son controlados por separado con los pines T_CS y SD_CS.

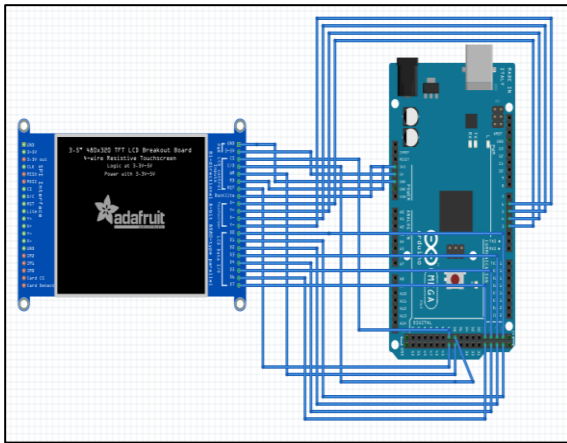


Figura 5. Diagrama de conexión de la pantalla.

El almacenamiento de datos se realiza con la utilización del módulo SD que viene integrado en la pantalla, para ello se utiliza la librería <SD.h> que permite almacenar los datos requeridos para el análisis estadístico de forma rápida y ordenada. El almacenamiento se realiza al momento de dar fin al dispositivo y los datos se guardan en el archivo VIAJES.CSV de la tarjeta de memoria a través de la ejecución del código que se indica en la figura 6.

```

codigo2 $
2231  datos=SD.open("VIAJE.csv",FILE_WRITE);
2232  if(datos)
2233  {
2234    datos.println();
2235    datos.print(titulo);
2236    datos.print(",");
2237    datos.print(",");
2238    datos.print(dia);
2239    datos.print(",");
2240    datos.print(mes);
2241    datos.print(",");
2242    datos.print(a_o);
2243    datos.print(",");
2244    datos.print(",");
2245    if(hora<10){datos.print("0");}
2246    datos.print(hora);
2247    datos.print(",");
2248    if(minuto<10){datos.print("0");}
2249    datos.print(minuto);
2250    datos.print(",");
2251    datos.print(",");
2252    if(hora<10){datos.print("0");}
2253    datos.print(hora);
2254    datos.print(",");
2255    if(minuto<10){datos.print("0");}
2256    datos.print(minuto);
2257    datos.print(",");
2258    datos.print(",");
    
```

Figura 6. Código almacenamiento de datos.

2.3 Definición de los lugares de control.

Para la definición de los lugares de control en cada uno de los viajes es necesario delimitar a través de cuadrantes de aproximadamente 15400 metros cuadrados para asegurar el almacenamiento de datos y la toma de posición del autobús, a través de los valores de longitud y latitud proporcionados por el GPS, tal como se indica en la figura 7.

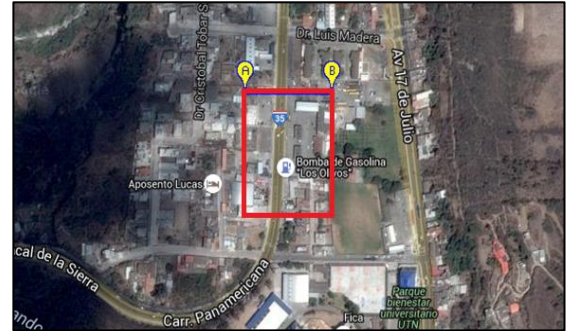


Figura 7. Delimitación de los lugares de control.

En la tabla 1 se indica los diez puntos de control definidos en los cuatro viajes programados en el dispositivo, cada uno de ellos están ubicados en la ruta a seguir por el autobús.

Nº	Viajes			
	Ibarra-Quito	Quito-Ibarra	Ibarra-Tulcán	Tulcán-Ibarra
1	Terminal Ibarra	Terminal Quito	Terminal Ibarra	Terminal Tulcán
2	La Florida	Calderón	Olivo	Control Policía
3	San Antonio	Guallabamba	Salinas	Guagua Negro
4	Atuntaqui	Tabacundo	Juncal	Julio Andrade
5	Otavaló	Cajas	Bolívar	San Gabriel
6	Cajas	Otavaló	San Gabriel	Bolívar
7	Tabacundo	Atuntaqui	Julio Andrade	Juncal
8	Guallabamba	San Antonio	Guagua Negro	Salinas
9	Calderón	La Florida	Control Policía	Olivo
10	Terminal Quito	Terminal Ibarra	Terminal Tulcán	Terminal Ibarra

Tabla 1. Lugares de control.

2.4 Funcionamiento del dispositivo.

El funcionamiento del software está conformado por 5 subrutinas para cumplir con los objetivos planteados, cada una de ellas realiza una función específica para recolectar y almacenar datos para su posterior análisis. Las subrutinas utilizadas son las siguientes:

- Subrutina menú.
- Subrutina viajes.
- Subrutina funcionamiento.
- Subrutina almacenamiento.
- Subrutina home.

El dispositivo está desarrollado con la finalidad de presentar datos de tiempos de atrasos y adelantos en tiempo real como también de su almacenamiento para el respectivo análisis estadístico en el software desarrollado. El funcionamiento general del dispositivo desarrollado se presenta en el flujograma de la figura 8.

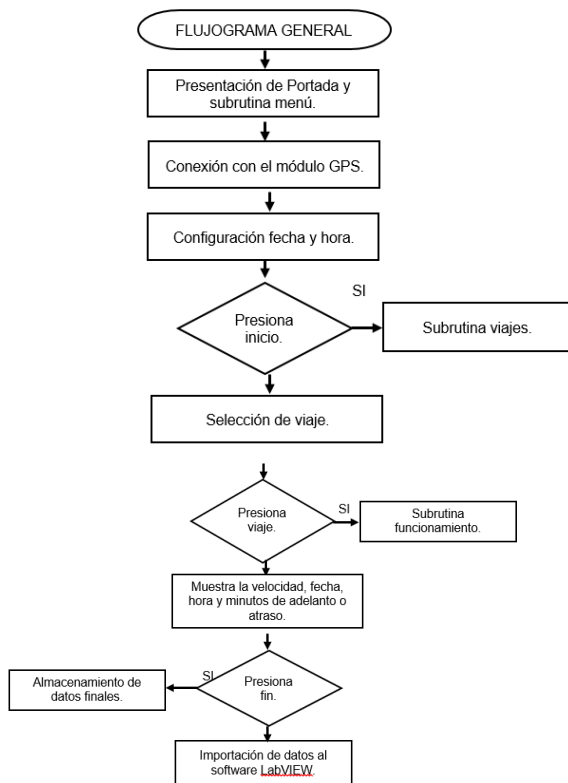


Figura. 8. Flujograma general.

El dispositivo diseñado funciona con una alimentación de cinco voltios proporcionados por un regulador de voltaje que tiene una entrada de corriente de 12 voltios procedentes

de la batería del autobús. Este se instala tal como se indica en la figura 9.



Figura. 9. Instalación del dispositivo.

El dispositivo diseñado cuenta con tres botones para su correcto funcionamiento. Cada uno de ellos desempeña una función específica y se los ha llamado: Inicio, Fin y Home, como se observa en la figura 10. El botón inicio permite visualizar una segunda pantalla con la lista de viajes disponibles; el botón fin permite finalizar el funcionamiento al dispositivo y el botón home regresa a la pantalla principal y de esta manera todas las variables definidas regresan a sus valores iniciales.

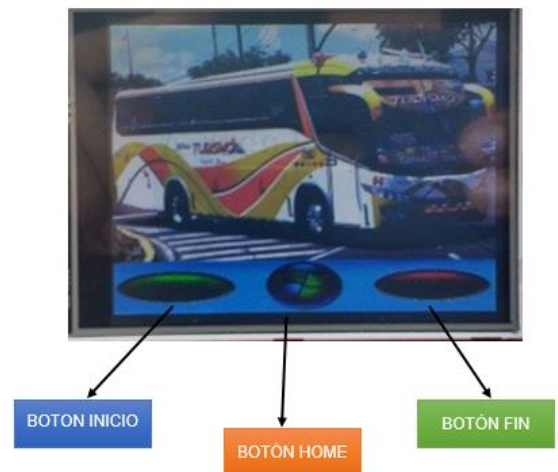


Figura. 10. Botones de control.

Para la selección de viajes se dispone de un cuadro rojo junto al nombre del viaje a elegir, como se observa en la figura 11. Para seleccionar el viaje se presiona el cuadro rojo junto al nombre de cada uno de ellos. Se dispone de cuatro viajes que son: Ibarra - Quito, Quito - Ibarra, Ibarra -Tulcán, Tulcán - Ibarra.



Figura 11. Ventana de selección de viajes.

Al presionar el cuadro se muestra una tercera pantalla que indica datos importantes y en tiempo real durante el funcionamiento del dispositivo. Los datos presentados contienen la siguiente información: nombre del viaje, velocidad en km/h, la hora y los minutos de atraso o adelanto al pasar por cada uno de los puntos de control establecidos, tal como se indica en la figura 12.



Figura 12. Ventana de funcionamiento del dispositivo.

Durante el funcionamiento del dispositivo se muestra la información en la pantalla distribuida en cuadros específicos como se indica en la figura 13, indicando la velocidad en km/h, la hora y fecha actual, así como también los minutos de adelanto y atraso a medida que el autobús pasa por los lugares de control establecidos en la ruta a seguir.



Figura 13. Funcionamiento del dispositivo.

Además, a través de su funcionamiento se almacena datos específicos del viaje que son importantes para el análisis estadístico en el programa de análisis de datos creado. Esta información se almacena en archivos Excel con extensión CSV como se indica en la figura 14.

VIAJE	FECHA	HORA	POSICION	VELOCIDAD MAXIMA	VELOCIDAD MINIMA
IBARRA-QUITO	22/6/2016	17:36	0.338945	-78	90
IBARRA-QUITO	22/6/2016	18:35	0.338776	-78	93
IBARRA-QUITO	22/6/2016	18:38	0.338801	-78	95
IBARRA-QUITO	22/6/2016	18:41	0.338421	-78	101
IBARRA-QUITO	22/6/2016	18:43	0.338041	-78	103
IBARRA-QUITO	22/6/2016	18:46	0.337943	-78	105
IBARRA-QUITO	22/6/2016	18:48	0.337456	-78	103
IBARRA-QUITO	22/6/2016	18:50	0.337270	-78	99
IBARRA-QUITO	22/6/2016	18:52	0.337089	-78	97
IBARRA-QUITO	22/6/2016	18:54	0.336912	-78	94
IBARRA-QUITO	22/6/2016	18:56	0.336292	-78	90
IBARRA-QUITO	22/6/2016	18:58	0.336236	-78	91
IBARRA-QUITO	22/6/2016	18:59	0.336131	-78	94
IBARRA-QUITO	22/6/2016	19:00	0.336123	-78	94
IBARRA-QUITO	22/6/2016	19:01	0.336062	-78	96
IBARRA-QUITO	22/6/2016	19:02	0.336000	-78	97
IBARRA-QUITO	22/6/2016	19:03	0.335936	-78	98
IBARRA-QUITO	22/6/2016	19:04	0.335871	-78	98
IBARRA-QUITO	22/6/2016	19:05	0.335807	-78	97

Figura 14. Archivo viajes.csv.

También se almacenan en dos archivos diferentes la velocidad máxima y velocidad mínima en los archivos Vmax.CSV y Vmin.CSV respectivamente, como se observa en la figura 15. Los datos almacenados son el nombre del viaje, fecha, hora, posición en valores de longitud y latitud, y la velocidad obtenida.

VIAJE	FECHA	HORA	POSICION	VELOCIDAD MAXIMA	VELOCIDAD MINIMA
IBARRA-QUITO	22/6/2016	17:30	0.338945	-78	90
IBARRA-QUITO	22/6/2016	17:30	0.338776	-78	93
IBARRA-QUITO	22/6/2016	17:30	0.338801	-78	95
IBARRA-QUITO	22/6/2016	17:30	0.338421	-78	101
IBARRA-QUITO	22/6/2016	17:30	0.338041	-78	103
IBARRA-QUITO	22/6/2016	17:30	0.337943	-78	105
IBARRA-QUITO	22/6/2016	17:30	0.337456	-78	103
IBARRA-QUITO	22/6/2016	17:30	0.337270	-78	99
IBARRA-QUITO	22/6/2016	17:30	0.337089	-78	97
IBARRA-QUITO	22/6/2016	17:30	0.336912	-78	94
IBARRA-QUITO	22/6/2016	17:35	0.336292	-78	90
IBARRA-QUITO	22/6/2016	17:35	0.336236	-78	91
IBARRA-QUITO	22/6/2016	17:35	0.336131	-78	94
IBARRA-QUITO	22/6/2016	17:35	0.336123	-78	94
IBARRA-QUITO	22/6/2016	17:35	0.336062	-78	96
IBARRA-QUITO	22/6/2016	17:35	0.336000	-78	97
IBARRA-QUITO	22/6/2016	17:35	0.335936	-78	98
IBARRA-QUITO	22/6/2016	17:35	0.335871	-78	98
IBARRA-QUITO	22/6/2016	17:35	0.335807	-78	97

Figura 15. Archivos vmax.csv y vmin.csv.

2.5 Software de análisis de datos.

El programa de análisis de datos es construido a través del software LabVIEW con la utilización de la herramienta Report Generation Toolkit para Microsoft Excel que permite la importación de datos de los archivos almacenados en la tarjeta de memoria del dispositivo.

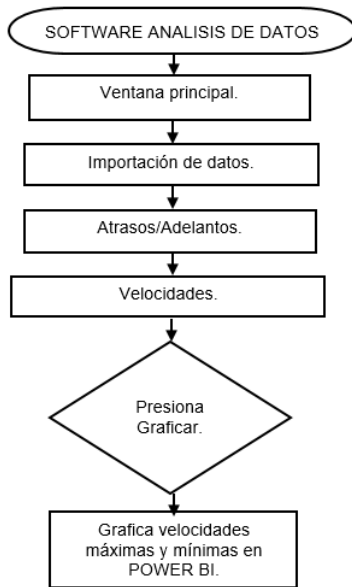


Figura. 16. Flujograma software análisis de datos.

El programa de análisis de datos está diseñado a través de un Tab control con cuatro páginas para mostrar información específica como es: ATRASOS/ADELANTOS, ESTADÍSTICAS Y VELOCIDADES. En la portada se indica el nombre de la institución con el respectivo tema de trabajo de grado, como se indica en la figura 17.



Figura. 17. Portada.

3. Resultados

Durante el funcionamiento del dispositivo se muestra la información en la pantalla distribuida en cuadros específicos como se indica en la figura 18, indicando la velocidad en km/h, la hora y fecha actual, así como también los minutos de adelanto y atraso a medida que el autobús pasa por los lugares de control establecidos en la ruta a seguir.



Figura. 18. Dispositivo en funcionamiento.

El programa de análisis de datos contiene cuatro ventanas, que muestra información de atrasos/adelantos, estadística y velocidades, cada una de ellas con tablas para mostrar los datos importados desde la tarjeta de memoria del dispositivo luego de realizar los cálculos correspondientes.

En la segunda ventana se indica el total de viajes, así como también los minutos de atrasos y adelantos que realiza el autobús en los diferentes viajes, además, frente a cada uno de estos se indica el promedio realizado en cada viaje, como se indica en la figura 19.

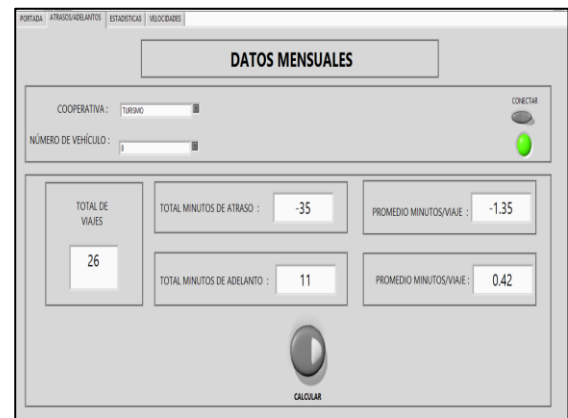


Figura. 19. Ventana de datos generales.

En la tercera ventana se indica la información de cada viaje, luego de seleccionar cada uno de ellos en el filtro establecido al principio de la ventana, como se puede observar en la figura 20. Al final de la tabla se observa un promedio de atrasos o adelantos en cada uno de los lugares de control establecidos en cada viaje.

# VIAJE	FECHA	HYSLEDA	BARRA	LA FLORIDA	SAN ANTONIO	ATATACAZA	OTISUALO	CABO	TABACUNO	GUALAMBAMA	CARCELIN	TERMINAL
1	BARRA-QUITO	20/06/2016	22:03	0	0	0	0	0	0	0	0	0
2	BARRA-QUITO	20/06/2016	18:40	0	0	0	0	0	0	0	0	0
3	BARRA-QUITO	20/06/2016	18:46	0	0	0	0	0	0	0	0	0
4	BARRA-QUITO	20/06/2016	18:46	0	0	1	0	0	0	0	1	2
5	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
6	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
7	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
8	BARRA-QUITO	20/06/2016	18:00	0	0	-1	-1	-3	-3	-3	-3	-1
9	BARRA-QUITO	20/06/2016	18:00	0	0	0	0	0	0	0	0	0
10	BARRA-QUITO	20/06/2016	18:00	0	0	0	0	0	0	0	0	0
11	BARRA-QUITO	20/06/2016	18:40	0	0	0	0	0	0	0	0	0
12	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
13	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
14	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
15	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
16	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
17	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
18	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
19	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
20	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0
21	BARRA-QUITO	20/06/2016	22:39	0	0	0	0	0	0	0	0	0

ATRASOS	0	0	1	1	0	0	0	0	0	0	0	0
ADELANTOS	0	0	0	1	0	0	0	0	0	0	1	2

Figura 20. Ventana adelanto/atraso.

Además, es posible seleccionar viajes en específico para conocer sus adelantos o atrasos en cada punto de control, que al final de cada columna indica un promedio de adelanto o atraso en cada uno de ellos. En la cuarta y última ventana se presenta la información de las velocidades máximas en la primera tabla, mientras que en la segunda tabla las velocidades mínimas, como se observa en la figura 21.

VELOCIDAD MÁXIMA					
VIAJE	FECHA	HORA	POSICION	VELOCIDAD MÁXIMA	
BARRA-QUITO	20/06/2016	21:4	0.30802, -78.18077	101	
BARRA-QUITO	20/06/2016	21:4	0.32752, -78.18076	101	
BARRA-QUITO	20/06/2016	21:5	0.34602, -78.18080	102	
BARRA-QUITO	20/06/2016	21:16	0.36671, -78.18229	102	
BARRA-QUITO	20/06/2016	21:27	0.37752, -78.18080	103	
QUITO-BARRA	20/06/2016	22:30	0.32752, -78.18047	104	
QUITO-BARRA	20/06/2016	22:30	0.32752, -78.18076	101	

VELOCIDAD MÍNIMA					
VIAJE	FECHA	HORA	POSICION	VELOCIDAD MÍNIMA	
BARRA-QUITO	20/06/2016	21:40	0.30802, -78.18081	0	
BARRA-QUITO	20/06/2016	21:40	0.32752, -78.18080	0	
BARRA-QUITO	20/06/2016	21:40	0.34602, -78.18084	25	
BARRA-QUITO	20/06/2016	21:40	0.35864, -78.18079	21	
BARRA-QUITO	20/06/2016	21:40	0.37752, -78.18079	0	
BARRA-QUITO	20/06/2016	21:53	0.34602, -78.18084	12	
BARRA-QUITO	20/06/2016	21:53	0.36672, -78.18022	0	
BARRA-QUITO	20/06/2016	21:58	0.34602, -78.18087	0	
BARRA-QUITO	20/06/2016	21:58	0.36672, -78.18082	21	
BARRA-QUITO	20/06/2016	21:18	0.32752, -78.18080	0	
BARRA-QUITO	20/06/2016	21:18	0.32754, -78.18080	0	
BARRA-QUITO	20/06/2016	21:18	0.32754, -78.18080	7	
BARRA-QUITO	20/06/2016	21:18	0.32754, -78.18072	0	

Figura 21. Ventana de velocidades.

La velocidad máxima o mínima se registra en cada viaje realizado y es almacenada en el respectivo archivo en la tarjeta de memoria del dispositivo. La información almacenada es el nombre del viaje realizado, la fecha, hora, posición y la velocidad registrada en ese instante.

En la misma ventana se dispone de un botón de Graficar que se utiliza para ejecutar el programa PowerBi, en el cual se grafican los lugares en los que el autobús sobrepasa los 100km/h o se mantiene durante 120 segundos en una velocidad menor a 30km/h; estos puntos son graficados como se observa en la figura 22.

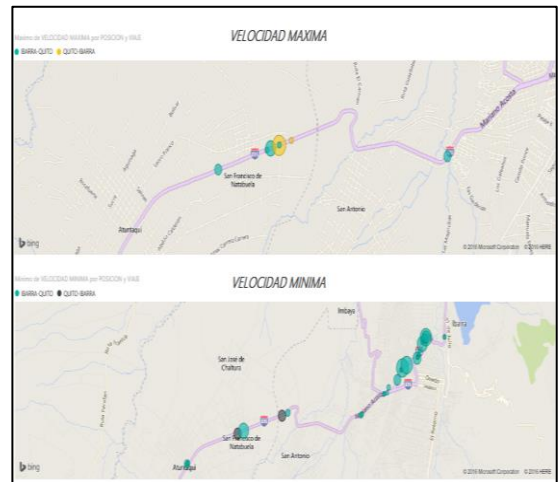


Figura 22. Ventana de representación gráfica.

4. Conclusiones

El dispositivo desarrollado a través de la plataforma de hardware libre Arduino permite conocer los minutos de adelanto y atraso que tiene un autobús durante su viaje, a través de los diferentes lugares de control establecidos en la programación.

El módulo GPS permite identificar los lugares de control de tiempo de viaje mediante las áreas determinadas a través de los valores de longitud y latitud, para así determinar los adelantos y atrasos.

El programa de análisis de datos desarrollado en el software LabVIEW brinda información estadística del total de minutos de atraso y adelanto, como también la información de cada uno de los viajes realizados durante un tiempo determinado.

El dispositivo almacena datos del viaje realizado como: nombre del viaje, fecha y hora de inicio, hora real y de control, tiempo de adelanto o atraso, velocidades máximas y mínimas, longitud y latitud, que permitirán obtener información estadística a través del programa desarrollado en el software LabVIEW.

La alimentación del dispositivo se realiza a través de una conexión de 12 voltios de la batería a un regulador que proporciona 5 voltios de corriente continua necesarios para su funcionamiento.

La característica adicional que brinda el dispositivo desarrollado es el reporte en tiempo real de los minutos de adelanto o atraso en cada uno de los lugares de control establecidos en la ruta a seguir por el autobús, información que permitirá al conductor un mejor control en el tiempo de viaje.

Agradecimientos

A Dios por brindarme la vida protegerme darme fortaleza sabiduría y perseverancia para ser cada día mejor persona, guiándome siempre por un camino de bien, por permitirme cumplir uno de mis mayores objetivos en mi vida universitaria.

A mis padres Narcisa y Manuel por brindarme su apoyo incondicional en todo momento y depositar su confianza total en mí, por enseñarme y demostrarme que todo lo que son y hacen es por el bien y superación de mis hermanos y mío.

A mis abuelitos María Luz y Víctor por sus consejos, apoyo moral y económico, siendo un pilar fundamental para culminar con esta propuesta en mi vida. De igual manera a mi abuelita Anita por brindarme sus mejores deseos oraciones y bendiciones siempre.

A toda mi familia que estuvo pendiente dándome ánimos y apoyo incondicional, de manera especial a mi tía Mercedes y mi prima Fernanda por entregarme su amor infinito, por estar siempre a mi lado porque admiro su fortaleza para ayudar siempre y brindar lo mejor de ellas sin esperar nada a cambio.

Mis más altos sentimientos de gratitud y estima para el Ing. Fredy Rosero Msc. Docente de la Facultad de Ingeniería y Ciencias Aplicadas, director de la investigación quien durante la realización de este presente trabajo supo ofrecerme su mano amiga con paciencia, confianza y dedicación.

Referencias Bibliográficas

- [1] Ó. T. Artero, Arduino: Curso práctico de formación, RC Libros, 2013.
- [2] A. C. Pedrera, Arduino para principiantes, IT Campus Academy, 2014.
- [3] P. Correia, Guía práctica del GPS, Marcombo, 2010.
- [4] L. Letham, GPS fácil. Uso del sistema de posicionamiento global, Paidotribo, 2009.
- [5] M. V. Florez Lopez y G. M. Isaza Posse, Descubre tu GPS, Grupo Editorial Colombia, 2015.
- [6] J. R. Lajara Vizcaíno y J. Pelegrí Sebastián, LabVIEW: Entorno gráfico de programación, Marcombo, 2012.
- [7] Arduino, «Arduino,» 2016. [En línea]. Available: <https://www.arduino.cc/>.
- [8] A. M. Lázaro, LabVIEW 6i: programación gráfica para el control de instrumentación, Paraninfo, 2010.
- [9] J. R. Lequerica, Arduino práctico, Anaya Multimedia-Anaya Interactiva, 2013.
- [10] M. Banzi, Introducción a Arduino, Grupo Anaya Comercial, 2012.
- [11] N. GOILAV y G. LOI, Arduino: Aprender a desarrollar para crear objetos inteligentes, 2016.
- [12] S. Arnalich y J. Urruela, GPS, Google Earth y Cooperación, Arnalich, 2012.
- [13] G. Fesser, Ruedas y el secreto del GPS, Grupo Editorial España, 2013.
- [14] V. Zanini, Macros en excel 2013, Buenos Aires: USERSHOP, 2013.
- [15] L. Morales Costa y L. E. Flores Fernandez, La problemática del transporte interprovincial de pasajeros, SIC, 2009.

Sobre los Autores...

Autor-Jeferson Portilla. Nació en la ciudad de Ibarra el 17 de agosto de 1992. Realizó sus estudios primarios en la escuela San Juan Diego. Sus estudios de ciclo básico en el colegio San Francisco y ciclo diversificado en el colegio La Salle en la especialidad de Físico Matemático. Actualmente es egresado de la Universidad Técnica del Norte en la carrera de Ingeniería en Mantenimiento Automotriz.

Coautor-Fredy ROSERO. Nació en la ciudad de Ibarra el 29 de Diciembre de 1984. Cursó sus estudios secundarios en el Colegio Fisco-misional "San Francisco" en la especialidad Físico Matemático; sus estudios de pregrado los realizó en la Escuela Politécnica del Ejército, obteniendo el título de Ingeniero Automotriz. En los estudios de postgrado obtuvo un Diplomado Superior en Autotróica en la Escuela Politécnica del Ejército, finalmente obtuvo una Maestría en Ingeniería del Transporte en la Pontificia Universidad Católica del Ecuador.

