



**ESCUELA SUPERIOR POLITÉCNICA DE
CHIMBORAZO**

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

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

**“DISEÑO E IMPLEMENTACIÓN DE UN CONTROL DE VELOCIDAD PARA EL
BUS DE LA FIE EN LA CIUDAD DE RIOBAMBA”**

TESIS DE GRADO

Previo a la obtención del título de:

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

Presentado por:

DARWIN ROBERTO POMAQUERO SANGA

ROBERTO CARLOS OÑATE LOPEZ

Riobamba – Ecuador

2013

AGRADECIMIENTO

A nuestro Dios por habernos guiado por el camino de la felicidad hasta ahora. Y a todos aquellos maestros y amigos quienes han estado ahí en los momentos de alegrías y tristezas a lo largo de nuestra vida estudiantil por todo ello es mi más profundo agradecimiento.

DEDICATORIA

A mis padres Fidel Pomaquero y Ana Sanga que me apoyaron incondicionalmente y de sobremanera en lo largo de mi vida estudiantil, gracias por sus consejos y por el inmenso amor que siempre recibo de ustedes.

A mis hermanos Arturo y Caleb por ser los motores de mi fortaleza para enfrentar todos los retos que se me hayan presentado y que vendrán, por extrañarme en la distancia y ser mis camaradas en los momentos difíciles.

Darwin

A Dios y a mis padres Enrique Oñate y Ana López por su inmenso amor y por haber sembrado en mí la semilla con la que crecen los hombres de bien, brindándome siempre su mano amiga y buenos consejos a parte del sustento económico para lograr alcanzar mis objetivos en el ámbito estudiantil.

A mi esposa Alexandra Hernández y a mi hija Génesis Oñate por su total entrega y comprensión en los momentos que no podía estar con Ustedes, y por todo el amor y aliento que me fortalecía en esos momentos difíciles.

Su apoyo incondicional ha sido un el pilar fundamental para alcanzar esta instancia de mi vida, ya que esto que he logrado es para todos Ustedes.

Roberto

FIRMAS DE RESPONSABLES Y NOTA

NOMBRE	FIRMA	FECHA
Ing. Iván Menes DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
Ing. Paúl Romero DIRECTOR DE LA ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES INDUSTRIALES
Ing. Jorge Luis Paucar DIRECTOR DE TESIS.
Ing. Jhony Vizueté MIEMBRO DEL TRIBUNAL
Tec. Carlos Rodríguez Carpio DIRECTOR DPTO DOCUMENTACIÓN
NOTA DE LA TESIS	

TEXTO DE RESPONSABILIDAD

“Nosotros DARWIN ROBERTO POMAQUERO SANGA Y ROBERTO CARLOS OÑATE LOPEZ, somos responsables de las ideas, doctrinas y resultados expuestos en esta tesis; y, el patrimonio intelectual de la Tesis de Grado pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”.

.....

DARWIN ROBERTO POMAQUERO SANGA

.....

ROBERTO CARLOS OÑATE LOPEZ

INDICE DE ABREVIATURAS

NA	Normalmente Abierto
NC	Normalmente Cerrado
LDR	Light Dependent Resisto
PTC	Positive Temperature Coefficient
NTC	Coeficiente de Temperatura Negativo
LED	Diodo Emisor de Luz
VDC	Voltaje Corriente Directa
DSP	Procesador Digital de Señal
ROM	Read Only Memory
RAM	Memoria de Acceso Aleatorio
CAN	Controller Area Network
UCP	Unidad Central de Proceso
POR	Power On Reset
VSS	Vehicle Speed Sensor
ECM	Modulo de Control del Motor
CA	Corriente Alterna
RISC	Computador de Juego de Instrucciones Reducido
CAD	Conversores Analógico/Digital
CDA	Conversores Digital/Analógico
UCP	Unidad Central de Procesamiento
LCD	Liquid Crystal Display
RISC	Reduced Instruction Set Computer
PWM	Pulse Wide Modulation

ÍNDICE GENERAL

PORTADA

AGRADECIMIENTO

DEDICATORIA

FIRMAS DE RESPONSABLES Y NOTA

TEXTO DE RESPONSABILIDAD

ÍNDICE DE ABREVIATURAS

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ÍNDICE DE ANEXOS

CAPITULO I

<i>1. GENERALIDADES</i>	<i>15</i>
1.1. Antecedentes	15
1.2. Justificación.....	16
1.3. Objetivos	17
1.3.1. Objetivo General.....	17
1.3.2. Objetivos Específicos	17
1.4. Hipótesis.....	17

CAPITULO II

<i>2. MARCO TÉORICO</i>	<i>18</i>
2.1. Sensores.....	18
2.1.1. Clasificación	19
2.1.2. Tipos de Sensores	20
2.2. Relés.....	23
2.2.1. Definición	23
2.3. Sistemas neumáticos	24

2.3.1.	Actuadores Neumáticos:	24
2.3.2.	Cilindros de simple efecto	26
2.3.3.	Consumo de aire en el cilindro	26
2.3.4.	Producción de aire comprimido	27
2.3.5.	Recomendaciones para el montaje de cilindros neumáticos.....	27
2.4.	Electroválvula	28
2.4.1.	Electroválvula 5/2 Vías Monoestable	29
2.5.	Sensor de velocidad.....	29
2.5.1.	Sensor tipo interruptor Reed Switch	30
2.5.2.	Sensor de Velocidad Tipo VSS	31
2.6.	Módulos Arduino	35
2.6.2.	Comunicación vía puerto Serial:.....	37
2.6.3.	Instalación del software Arduino	38
2.6.4.	Programación en Arduino	45
2.6.5.	Lenguaje de texto estructurado	45

CAPITULO III

3.	<i>DISEÑO DEL SISTEMA DE CONTROL DE VELOCIDAD</i>	47
3.1.	INTRODUCCIÓN	47
3.5.	Diagrama de bloques del sistema de control.....	50
3.5.1.	Bloque de entrada de datos	51
3.5.2.	Bloque de control.....	52
3.5.3.	Bloque de actuador	53
3.6.	Diseño de la Etapa de potencia en el Bloque del Actuador	57

CAPITULO IV

4.	PRUEBAS Y RESULTADOS.....	76
4.1.	MANUAL TÉCNICO	78

4.1.1. Especificaciones Técnicas de manejo de la Tarjeta Arduino Uno:.....	78
4.2. MANUAL DE USUARIO	81

CONCLUSIONES

RECOMENDACIONES

RESUMEN

SUMARY

BIBLIOGRAFÍA

BIBLIOGRAFÍAS DE INTERNET

ANEXO

ÍNDICE DE FIGURAS

Figura II. 1: Sensor de contacto	21
Figura II. 2: Bloque de función de un sensor.....	22
Figura II. 3: Sensor magnético.....	22
Figura II. 4: Estructura de un relé	23
Figura II. 5: Relé de 24 VDC.....	23
Figura II. 6: Cilindro de doble efecto	25
Figura II. 7: Electroválvula 5/2 Monoestable	29
Figura II. 8: Sensor Tipo de Interruptor Reed Switch	30
Figura II. 9: Ubicación del sensor de velocidad	32
Figura II 10: Tipo de Onda	34
Figura II. 11: Sistema decodificador y control de señal binaria	35
Figura II. 12: Trama de Bits.....	36
Figura II. 13: Descarga de Software Arduino.....	38
Figura II. 14: Ejecutable del Paquete de Instalación	39
Figura II. 15: Ejecutar en modo de administrador	39
Figura II. 16: Exploración del Software	40
Figura II. 17: Comprobación del Puerto Serial.....	40
Figura II. 18: Ingreso al administrador de dispositivos	41
Figura II. 19: Verificación en el puerto COM	41
Figura II. 20: Ejecución del Software	42
Figura II. 21: Entorno de desarrollo.....	42
Figura II. 22: Revisión de iconos en Arduino 10.5.....	43
Figura II. 23: Crear nuevo proyecto.....	43
Figura II. 24: Reconocimiento del entorno de programación	44
Figura II. 25: Entorno de trabajo Arduino 10.5	44

Figura II. 26: Programación en texto estructurado	46
Figura III. 27: Diagrama de bloques del sistema	51
Figura III. 28: Sensor de velocidad integrado en el Bus de la FIE	52
Figura III. 29: Sensor control de velocidad tipo Vss	52
Figura III. 30: Simulación del sistema de bloqueo del acelerador.....	54
Figura III. 31: Diseño de Bloqueo al sistema de aceleración	55
Figura III. 32: Adaptación de bloqueo al acelerador	56
Figura III. 33: Construcción del sistema de bloqueo	56
Figura III. 34: Simulación divisor de voltaje	57
Figura III. 35: Diseño etapa de potencia.....	58
Figura III. 36: Simulación del reductor de velocidad	59
Figura III. 37: Diseño del reductor de velocidad	59
Figura IV. 38: Bus de la Fie.....	77
Figura IV. 39: Pines de alimentación de la Tarjeta Arduino Uno	79
Figura IV. 40: Reconocimiento de dispositivos.....	81
Figura IV. 41: Ingreso del usuario	81
Figura IV. 42: Ingreso al menú de Usuario	82
Figura IV. 43: Selección del tipo de usuario	82
Figura IV. 44: Ingreso de clave	83
Figura IV. 45: Digitación de la clave de acceso	83
Figura IV. 46: Visualización de clave incorrecta	84
Figura IV. 47: Visualización de clave correcta.....	84
Figura IV. 48: Selección del tipo de recorrido del vehículo.....	85
Figura IV. 49: Selección de zona Urbana, Límite máximo permitido.....	85
Figura IV. 50: Selección de zona Perimetral, Límite máximo permitido.....	86
Figura IV. 51: Descripción del entorno de desarrollo	86

ÍNDICE DE TABLAS

Tabla II. 1: Clasificación de sensores	19
Tabla III. 2: Designación de Pines de la tarjeta Arduino.....	53
Tabla IV. 3: Tiempos de desarrollo de las prácticas.....	78
Tabla IV. 4: Parametros de Funcionamiento de la Tarjeta Arduino.....	79

ÍNDICE DE ANEXOS

ANEXO 1:

Diagrama de funcionamiento del sistema de control de velocidad

ANEXO 2:

Plano de conexiones del sistema neumático

ANEXO 3:

Plano de conexiones del sistema de entrada del sensor

ANEXO 4:

Plano de conexiones del sistema de frenado

ANEXO 5:

Plano de la unidad de control

ANEXO 6:

Plano de las conexiones del sistema de control de velocidad

INTRODUCCIÓN

El principal reto que se presenta en los sistemas de automatización es obtener el control y supervisión de los mismos, optimizando los recursos humanos, técnicos y económicos a la vez reduciendo los costos de implementación. Utilizando tecnología de última generación, se implementará un sistema sencillo de utilizar y programar por cualquier persona.

El diseño e implementación para el control de velocidad nos ayudará a no sobre pasar los límites de velocidad establecidos por la comisión nacional de tránsito, se realizará el diseño de un prototipo implementado en el bus de la Facultad de Informática y Electrónica el cual actuara directamente en paralelo al sistema de frenado, a la par se enviara un sistema visual y de audio que nos indicara que se ha excedido del límite de velocidad de acuerdo al tipo de transporte y lugar de circulación.

El control de velocidad se lo ha diseñado basándonos en sistemas ya probados pero con un costo elevado. Se mejorará el diseño bajando los costos de implementación siendo accesible en el mercado nacional.

En la actualidad los accidentes de tránsito por exceso de velocidad no se han reducido debido a la irresponsabilidad de los conductores que infringen los límites de circulación vehicular al no respetarla. Todo esto ha demandado en una constante búsqueda a realizar un sistema que se presenta a continuación, utilizando modelos de prototipos y técnicas de trabajo.

Demostrando que nuestro sistema es factible realizar para controlar de forma eficiente, segura y confiable la circulación vehicular, utilizando una adecuada tarjeta de adquisición y una programación estructurada ideal.

CAPITULO I

1. GENERALIDADES

1.1. Antecedentes

La situación que enfrenta las Agencia Nacional de Tránsito en todo el país es dificultosa, cada vez se producen más accidentes en las vías que dejan no sólo un saldo económico irremediable, sino psicológico para quienes han sobrevivido a un hecho de éste tipo y lo que es más crítico un saldo de vidas humanas; familias devastadas por el dolor, tragedia y sobretodo la gran premisa de los ciudadanos de qué hacer para evitarlo. A pesar de que en la actualidad rigen leyes de Tránsito¹ que se consideran más estrictas en sus sanciones esto no basta para frenar la irresponsabilidad de los señores choferes que aún a sabiendas de los límites de velocidad que son necesarios alcanzar en sus travesías los exceden.

¹ LEY GENERAL DE TRANSPORTE Y TRÁNSITO TERRESTRE N° 2718 PUBLICADO EL 18-01-2003

La estadísticas según la Dirección Nacional de Tránsito nos dice que la mayoría de los accidentes de tránsito son por exceso de velocidad a tal razón nos permitimos realizar un control de velocidad de tal forma poder disminuir los accidentes de tránsito.

1.2. Justificación

El número de población del mundo en el que se vive, es ascendente, por lo tanto las necesidades de estos también han crecido a la par. El mundo globalizado en el que se desenvuelven las personas exige que las mismas se interrelacionen, que no permanezcan estáticos en un solo lugar, sino que se movilicen, es decir se trasporten para realizar sus actividades cotidianas y satisfacer sus requerimientos.

Se considera que el transporte público es el medio por el que la mayoría de población se moviliza. De ahí que el alto número de accidentes que se ha presentado en el último tiempo, ha dejado un saldo trágico de muchas personas muertas, personas heridas, traumatizadas, y muchas familias destrozadas por la imprudencia de algunos de los conductores.

A pesar de los múltiples controles que se han implementado en calles, terminales y demás, no se ha logrado disminuir el número de accidentes, incluso se han ido incrementando poco a poco.

Es evidente que para dar una mejor solución y así minimizar el número de accidentes, se necesita realizar una intervención en cada una de las unidades de transporte. En tal virtud el sistema de control de velocidad automático, ayudará a disminuir el exceso de la velocidad ya que al estar destinado para buses interprovinciales permitirá llegar solo hasta una determinada velocidad.

Salvaguardando la integridad de las personas que en el transporte se movilizan y almacenando información para un posterior monitoreo de manera que se pueda sancionar estrictamente a los infractores.

1.3. Objetivos

1.3.1. Objetivo General

- Diseñar e implementar de un control de velocidad para el bus de la FIE en la ciudad de Riobamba

1.3.2. Objetivos Específicos

- Diseñar el módulo de lectura de velocidad del autobús.
- Diseñar el módulo electrónico neumático de frenado.
- Diseñar el sistema de control de velocidad del autobús.
- Integración de los módulos por medio de una tarjeta Arduino para ejercer el control de velocidad.

1.4. Hipótesis

El diseño e implementación de un dispositivo para el control de velocidad en el bus de la FIE permitirá controlar la velocidad, de tal forma que no pueda exceder el límite de velocidad permitido para vehículos de transporte de pasajeros.

CAPITULO II

2. MARCO TÉORICO

2.1. Sensores

Un sensor es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, etc. Una magnitud eléctrica puede ser una resistencia eléctrica, una capacidad eléctrica como en un sensor de humedad, una Tensión eléctrica, una corriente eléctrica (como en un fototransistor). Un sensor es un tipo de transductor que transforma la magnitud que se quiere medir o controlar. Pueden ser de indicación directa (e.g. un termómetro de mercurio) o pueden estar conectados a un indicador (posiblemente a través de un convertidor analógico a digital, un computador y un display) de modo que los valores detectados puedan ser leídos por un humano.

2.1.1. Clasificación

Aunque es un poco complicado realizar una clasificación única, debido a la gran cantidad de sensores que existen actualmente, los siguientes son las clasificaciones más generales.

➤ Según el tipo de señal de salida

Analógicos. Dan como salida un valor de tensión o corriente variables en forma continua dentro del campo de medida.

Digitales. Dan como salida una señal en forma de una palabra digital.

➤ Según la magnitud física a detectar

Posición, velocidad, fuerza y par, presión, caudal, proximidad, etc.

MAGNITUD FÍSICA	TRANSDUCTOR	CARACTERÍSTICAS
Posición (Lineal o Angular)	Potenciómetro	Analógico
	Encoder	Digital
Pequeños Desplazamientos	Transformador Diferencial	Analógico
	Galga Extensiométrica	Analógico
Velocidad (Lineal o Angular)	Dinamo Tacométrica	Analógico
	Encoder	Digital
	Detectro inductivo u óptico	Digital
Aceleración	Acelerómetro	Analógico
	Sensor de velocidad + calculador	Digital
Fuerza y Par	Galga Extensiométrica	Analógico
Temperatura	Termopar	Analógico
	Resistencias PT 100	Analógico
	Termistores CTN	Analógico
	Termistores CTP	Todo-Nada
Sensores de Presencia o Proximidad	Inductivos	Analógico o Todo-Nada
	Capacitivos	Todo-Nada
	Opticos	Analógico o Todo-Nada

FUENTE: <http://www.unet.edu.ve/~ielectro/sensores.pdf>

Tabla II. 1: Clasificación de sensores

2.1.2. Tipos de Sensores

Existen diferentes tipos de sensores, en función del tipo de variable que tengan que medir o detectar:

- De contacto.
- Ópticos.
- Térmicos.
- De humedad.
- Magnéticos.
- De infrarrojos.

2.1.2.1 Sensores de Contacto

Se emplean para detectar el final del recorrido o la posición límite de componentes mecánicos. Por ejemplo: saber cuándo una puerta o una ventana que se abren automáticamente están ya completamente abiertas y por lo tanto el motor que las acciona debe pararse.

Los principales son los llamados fines de carrera (o finales de carrera). Se trata de un interruptor que consta de una pequeña pieza móvil y de una pieza fija que se llama NA, normalmente abierto, o NC, normalmente cerrado.

La pieza NA está separada de la móvil y sólo hace contacto cuando el componente mecánico llega al final de su recorrido y acciona la pieza móvil haciendo que pase la corriente por el circuito de control.

La pieza NC hace contacto con la móvil y sólo se separa cuando el componente mecánico llega al final de su recorrido y acciona la pieza móvil impidiendo el

paso de la corriente por el circuito de control. Según el tipo de fin de carrera, puede haber una pieza NA, una NC o ambas.



FUENTE: http://recursostic.educacion.es/secundaria/edad/4esotecnologia/quincena11/4quincena11_contenidos_3b.htm

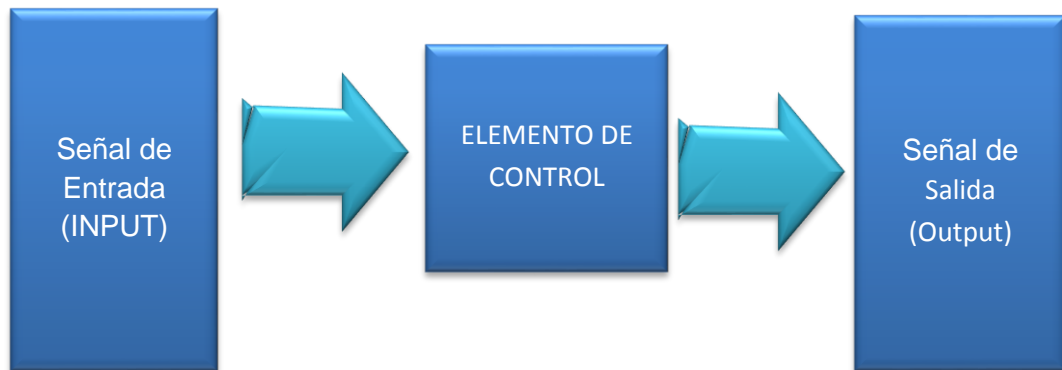
Figura II. 1: Sensor de contacto

2.1.2.2 Sensores ópticos

Detectan la presencia de una persona o de un objeto que interrumpen el haz de luz que le llega al sensor. Recordemos que se trataba de resistencias cuyo valor disminuía con la luz, de forma que cuando reciben un haz de luz permiten el paso de la corriente eléctrica por el circuito de control. Cuando una persona o un obstáculo interrumpen el paso de la luz, la LDR aumenta su resistencia e interrumpe el paso de corriente por el circuito de control.

Las LDR son muy útiles en robótica para regular el movimiento de los robots y detener su movimiento cuando van a tropezar con un obstáculo o bien disparar alguna alarma. También sirven para regular la iluminación artificial en función de la luz natural. El circuito que aparece en la imagen superior derecha nos permitiría controlar la puesta en marcha de una alarma al disminuir la intensidad luminosa que incide sobre un LDR.

CIRCUITO AUTOMÁTICO



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura II. 2: Bloque de función de un sensor

2.1.2.3 Sensores Magnéticos

Detecta los campos magnéticos que provocan los imanes o las corrientes eléctricas. El principal es el llamado interruptor Reed; consiste en un par de láminas metálicas de materiales ferromagnéticos metidas en el interior de una cápsula que se atraen en presencia de un campo magnético, cerrando el circuito.



FUENTE: http://recursostic.educacion.es/secundaria/edad/4esotecnologia/quincena11/4_quincena11_contenidos_3b.htm

Figura II. 3: Sensor magnético

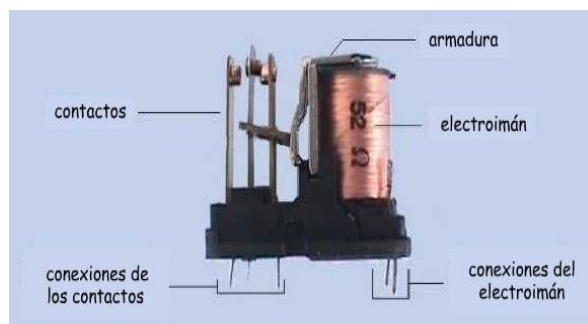
El interruptor Reed puede sustituir a los finales de carrera para detectar la posición de un elemento móvil, con la ventaja de que no necesita ser empujado

físicamente por dicho elemento sino que puede detectar la proximidad sin contacto directo.

2.2. Relés

2.2.1. Definición

Los Relés son interruptores o dispositivos de conmutación activados por señales, lo cual los hace extremadamente funcionales para que controlen cosas cuando se les manda una señal. Los relés están formados por un contacto móvil o polo y por un contacto fijo, pero también hay relés que funcionan como un conmutador, porque disponen de un polo (contacto móvil) y dos contactos fijos. Pueden ser de tipo electromecánico o totalmente electrónico, en cuyo caso carece de partes móviles.



FUENTE: <http://cpc.farnell.com/1/1/3687-relay-pcb-16a-spc0-24vdc-4061-7024>

Figura II. 4: Estructura de un relé

En la figura II.5 siguiente se muestra la imagen de un relé de 24 Vdc de corriente continua.



FUENTE: <http://eljonablog.blogspot.com/2009/12/el-rele-es-un-dispositivo.html>

Figura II. 5: Relé de 24 VDC

2.3. Sistemas neumáticos

Introducción a la neumática

La neumática es la tecnología que emplea el aire comprimido como modo de transmisión de la energía necesaria para mover y hacer funcionar mecanismos. El aire es un material elástico y por tanto, al aplicarle una fuerza, se comprime, mantiene esta compresión y devolverá la energía acumulada cuando se le permita expandirse, según la los gases ideales.

2.3.1. Actuadores Neumáticos:

Un actuador es un dispositivo inherentemente mecánico cuya función es proporcionar fuerza para mover o “actuar” otro dispositivo mecánico. La fuerza que provoca el actuador proviene de tres fuentes posibles: Presión neumática, presión hidráulica, y fuerza motriz eléctrica (motor eléctrico o solenoide).

Dependiendo del origen de la fuerza el actuador se denomina “neumático”, “hidráulico” o “eléctrico”.

El actuador más común es el actuador manual o humano. Es decir, una persona mueve o actúa un dispositivo para promover su funcionamiento.

Con el tiempo, se hizo conveniente automatizar la actuación de dispositivos, por lo que diferentes dispositivos hicieron su aparición. Actualmente hay básicamente dos tipos de actuadores.

- Lineales
- Rotatorios

En esta sección nos centraremos en los actuadores lineales debido a su importancia en el sistema de control de velocidad.



FUENTE: <http://www.ec.all.biz/cilindros-neumaticos-bgg1059016>

Figura II. 6: Cilindro de doble efecto

Los cilindros neumáticos son unidades que transforman la energía potencial del aire comprimido en energía cinética o en fuerzas prensoras.

Básicamente consisten en un recipiente cilíndrico provisto de un émbolo o pistón.

Al introducir un determinado caudal de aire comprimido, éste se expande dentro de la cámara y provoca un desplazamiento lineal.

Si se acopla al émbolo un vástago rígido. Este mecanismo es capaz de empujar algún elemento, o simplemente sujetarlo.

La fuerza de empuje es proporcional a la presión del aire y a la superficie del pistón:

Ecuación 2.1. Fuerza que ejerce el cilindro

$$F = p \cdot A$$

Dónde: F = Fuerza

p = Presión manométrica

A = Área del émbolo o pistón

2.3.2. Cilindros de simple efecto

Uno de sus movimientos está gobernado por el aire comprimido, mientras que el otro se da por una acción antagonista, generalmente un resorte colocado en el interior del cilindro, entre el pistón y su tapa trasera (con resorte trasero).

Realiza trabajo aprovechable sólo en uno de los dos sentidos, y la fuerza obtenible es algo menor a la que da la expresión $F = P \times A$, pues hay que descontar la fuerza de oposición que ejerce el resorte. Uno de sus movimientos está gobernado por el aire comprimido, mientras que el otro se da por una acción antagonista, generalmente un resorte colocado en el interior del cilindro. Este resorte podrá situarse opcionalmente entre el pistón y tapa delantera (con resorte delantero).

Realiza trabajo aprovechable sólo en uno de los dos sentidos, y la fuerza obtenible es algo menor a la que da la expresión $F = P \times A$, pues hay que descontar la fuerza de oposición que ejerce el resorte.

2.3.3. Consumo de aire en el cilindro

El cálculo del consumo de aire en cilindro neumático es muy importante cuando se requiere conocer la capacidad del compresor necesario para abastecer la demanda de una instalación.

Puede calcularse con la siguiente fórmula:

Ecuación 2.2. Consumo de aire en cilindro neumático

$$Q = (\pi/4) \cdot d^2 \cdot c \cdot n \cdot P \cdot N \cdot 10^{-6}$$

Dónde:

Q = Consumo de aire (Nl/min)

d = Diámetro del cilindro (mm)

c = Carrera del cilindro (mm)

n = Número de ciclos completos por minuto

P = Presión absoluta=Presión relativa de trabajo + 1 bar

N = Número de efectos del cilindro

$N=1$ para simple efecto, $N=2$ para doble efecto

2.3.4. Producción de aire comprimido

El aire comprimido, por el hecho de comprimirse, comprime también todas las impurezas que contiene, tales como polvo, hollín, suciedad, hidrocarburos, gérmenes y vapor de agua. A estas impurezas se suman las partículas que provienen del propio compresor, tales como polvo de abrasión por desgaste, aceites y aerosoles y los residuos y depósitos de la red de tuberías, tales como óxido, residuos de soldadura.

Estas impurezas pueden crear partículas más grandes (polvo +aceite) por lo que dan origen muchas veces a averías y pueden conducir a la destrucción de los elementos neumáticos.

Es vital eliminarlas en los procesos de producción de aire comprimido, en los compresores y en el de preparación para la alimentación directa de los dispositivos neumáticos. Por otro lado, desde el punto de vista de prevención de los riesgos laborales.

2.3.5. Recomendaciones para el montaje de cilindros neumáticos

- a. Los cilindros neumáticos están diseñados para transmitir esfuerzos axiales. La presencia de esfuerzos radiales¹ o laterales sobre los vástagos conducirán a un desgaste prematuro de las guarniciones y de sus guías.

¹**ESFUERZO RADIALES:** Esfuerzos de dirección al centro de su eje

Por lo tanto, deberán analizarse detenidamente los tipos de montaje más adecuados para cada aplicación a efectos de anular dichos esfuerzos.

- b. Toda vez que se utilice un montaje basculante para el cilindro (en cualquiera de sus formas), deberá preverse un equivalente en el extremo del vástago. La combinación de montajes rígidos con basculantes resulta un contrasentido técnico que origina esfuerzos radiales sobre el vástago.
- c. Debe evitarse el montaje rígido del cilindro con el elemento a mover. En caso que sea inevitable, fijar suavemente el actuador y operarlo a baja presión de modo que entre y salga libremente y pueda auto alinearse.

2.4. Electroválvula

Una electroválvula tiene dos partes fundamentales: el solenoide y la válvula. El solenoide convierte energía eléctrica en energía mecánica para actuar la válvula. Una electroválvula es una válvula electromecánica, diseñada para controlar el flujo de un fluido a través de un conducto como puede ser una tubería. La válvula está controlada por una corriente eléctrica a través de una bobina solenoide. Las electroválvulas resultan del acoplamiento de un sistema electromecánico (solenoide–electroimán de accionamiento).

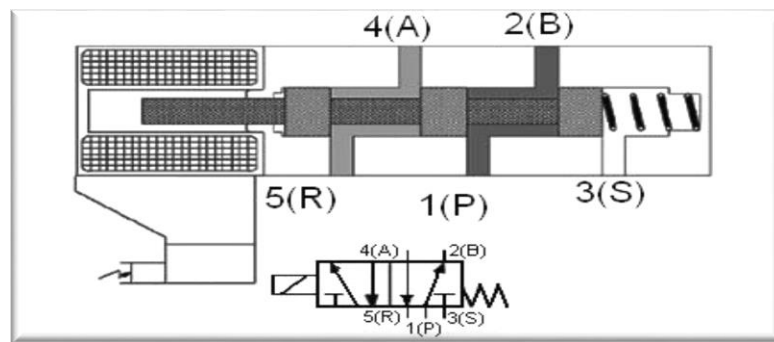
Estas válvulas se utilizan cuando la señal proviene de un temporizador eléctrico, un final de carrera eléctrico, presostatos o mandos electrónicos. En general, se elige el accionamiento eléctrico para mandos con distancias extremadamente largas y cortos tiempos de conexión.

Las electroválvulas o válvulas electromagnéticas se dividen en válvulas de mando directo o indirecto.

Al conectar el imán, el núcleo (inducido) es atraído hacia arriba venciendo la resistencia del muelle.

2.4.1. Electroválvula 5/2 Vías Monoestable

Cumple las mismas funciones que la de 4/2 vías y simplemente tiene otro sistema constructivo. Este tipo es de tipo corredera a diferencia de las de tipo asiento².



FUENTE: <http://es.scribd.com/doc/23721358/electroneumatica-basica>

Figura II. 7: Electroválvula 5/2 Monoestable

2.5. Sensor de velocidad

Hay al menos dos sensores de velocidad distintos en un vehículo. El primero es un generador de señal magnética que, al girar, genera un impulso eléctrico que es utilizado por la computadora. El sensor de velocidad en una transmisión de unidad de llanta trasera se encuentra en el alojamiento de la cola trasera de la transmisión.

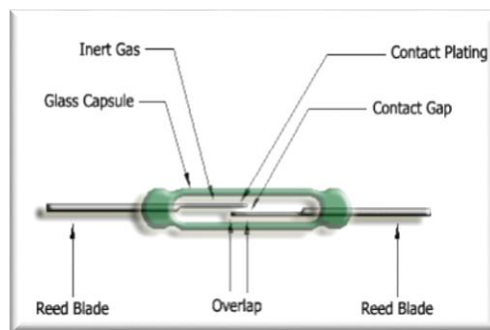
Es impulsado por un engranaje situado en el alojamiento de la caja de transmisión.

² (Manual Aeroquip., 2008)

En un vehículo de unidad de llanta frontal, el sensor de velocidad está situado en la parte superior del alojamiento diferencial sobre la transmisión, cerca del eje de accionamiento.

2.5.1. Sensor tipo interruptor Reed Switch

El tipo de interruptor de línea es impulsado por el cable del velocímetro. Los componentes principales son un imán, interruptor de láminas, y el cable del velocímetro. Conforme el imán gira, los contactos de interruptor de láminas se abren y cierran cuatro veces por vuelta. Esta acción produce cuatro pulsos por revolución. Con el número de pulsos emitido por la VSS. No puede manejar grandes valores de tensión lo que provoca chispas en su interior que afectan su vida útil.



FUENTE: http://es.wikipedia.org/wiki/Reed_switch

Figura II. 8: Sensor Tipo de Interruptor Reed Switch

Cuando el sensor falla provoca lo siguiente:

- Marcha mínima variable.
- Mucho consumo de combustible.
- Pérdida de la información de los kilómetros recorridos en un viaje, el kilometraje por galón, todo esto pasa en la computadora.
- El control de la velocidad de cruce pueda funcionar con irregularidad.

2.5.2. Sensor de Velocidad Tipo VSS

a) DESCRIPCIÓN

El sensor de velocidad del vehículo VSS (Vehicle Speed Sensor) es un captador magnético, se encuentra ubicado en el eje de la salida de la corona. El VSS proporciona una señal de corriente alterna al ECM la cuál es interpretada como velocidad del vehículo.

Lo que hace este sensor es determinar por el número de vueltas del neumático la velocidad del vehículo. Se generan de 4 a 8 ciclos por cada vuelta del neumático, la Computadora determina mediante un algoritmo y de acuerdo al diámetro de la llanta la velocidad a la que va el vehículo.

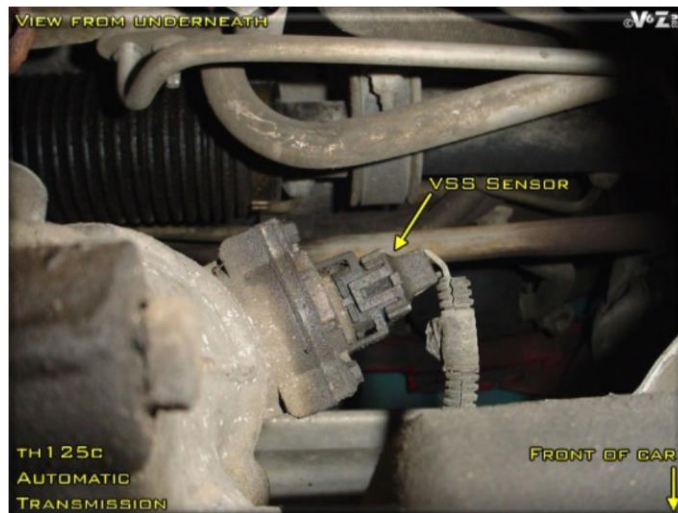
Al aumentar la velocidad del vehículo la frecuencia y el voltaje aumentan, entonces el ECM convierte ese voltaje en Km/h, el cual usa para sus cálculos. Los Km/h pueden leerse.

El VSS se encarga de informarle al ECM³ de la velocidad del vehículo para controlar el velocímetro, el acople del embrague convertidor de torsión (TCC) transmisiones automáticas, en algunos casos se utiliza como señal de referencia de velocidad para el control de cruceo y controlar el moto ventilador de dos velocidades del radiador.

Tiene en su interior un imán giratorio que genera una onda senoidal de corriente alterna directamente proporcional a la velocidad del vehículo.

Por cada vuelta del eje genera 8 ciclos, su resistencia debe ser de 190 a 240 Ohmios.

³ ECM: Módulo de Control del Motor



FUENTE: http://e-auto.com.mx/manual_detalle.php?manual_id=223

Figura II. 9: Ubicación del sensor de velocidad

Si es del tipo Hall, por cada 8 inversiones de campo magnético significa una vuelta, la ECM determina mediante un algoritmo la velocidad a la que va el vehículo considerando el diámetro de la llanta.

El sensor de velocidad del vehículo se encarga en enviar la velocidad a la cual se desplaza el vehículo a la unidad de control.

La mayoría de los nuevos sensores de velocidad de vehículos son del tipo de imán permanente, y la función es muy parecida a la del sensor del árbol de levas o el sensor del cigüeñal

El VSS se encarga de informarle al ECM de la velocidad del vehículo para controlar Control de la rotación en marcha lenta, Enriquecimiento de combustible durante la aceleración, Corte de combustible durante la desaceleración

FUNCIONAMIENTO

Los voltajes que proporciona el sensor a la computadora son interpretados para:

- La velocidad de la marcha mínima
- El embrague de convertidor de torsión
- Información para que marque la velocidad, al tablero.
- Para la función del sistema de control de la velocidad de crucero.

El VSS se encarga de informarle al ECM (Módulo de Control del Motor) de la velocidad del vehículo para controlar el velocímetro y el odómetro, el acople del embrague convertidor de torsión transmisiones automáticas.

La señal que emite este sensor también es utilizada para:

- Transmisión del auto
- Frenos ABS
- Componentes del tablero

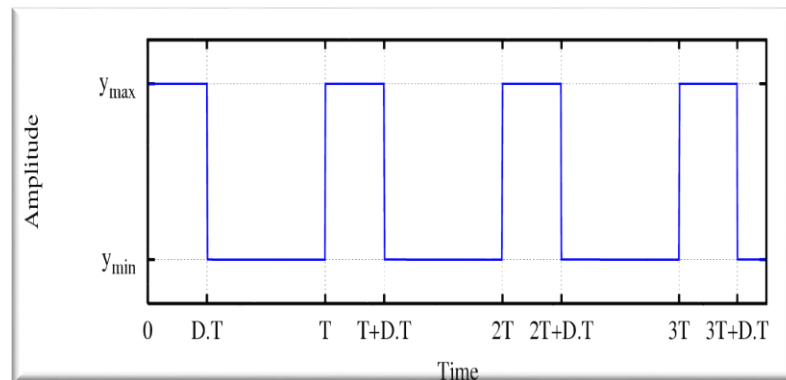
Este sensor se encuentra ubicado en la caja de transmisión. Al inclinar uno de los neumáticos del vehículo en un ángulo de 20° , si conectamos un multímetro a las terminales del tacómetro este debe marcar un voltaje de salida de 3.2V.

b) TIPO DE COMUNICACIONES (TREN DE PULSOS)

Los dispositivos o equipos industriales generan un patrón de pulsos desfasados en un ángulo de 90° , Una representación de esta señal son los Encoder mediante este patrón de pulsos pueden informar la velocidad de rotación y el sentido de giro. Otro dispositivo menos conocido son las ruedas dentadas que se encuentran dentro de un mouse. Este circuito permite crear este mismo patrón de pulsos desfasados, para fines de prueba y simulación.

Se denomina señal a una tensión que varía su potencial en el tiempo, el medio por el que viaja o se propaga es un factor que ejerce una resistencia al paso de la señal, dependiendo del medio, así se verá influenciada la señal, llegando a disminuir hasta potencial cero, este efecto se llama atenuación, cuando hablamos de trenes de pulsos, nos referimos a una serie de pulsos continuados por un intervalo de tiempo. Dos factores muy importantes en un tren de pulsos, es la frecuencia de repetición y su nivel. Con estos dos factores, se puede conocer su frecuencia.

La modulación por ancho de pulsos (también conocida como PWM, siglas en inglés de pulse-width modulation) de una señal o fuente de energía es una técnica en la que se modifica el ciclo de trabajo de una señal periódica una senoidal o una cuadrada.



FUENTE: http://es.wikipedia.org/wiki/Onda_cuadrada

Figura II 10: Tipo de Onda

Ecuación 2.3. Ciclo de trabajo

$$D = \frac{T}{T}$$

D : Es el ciclo de trabajo

T : Es el tiempo en que la función es positiva (ancho del pulso)

T : Es el período de la función

2.6. Módulos Arduino

Definición de Arduino

Arduino es una herramienta para controlar el mundo físico a través del ordenador personal, es una plataforma de desarrollo de computación física (physical computing) de código abierto, basada en una placa con un sencillo microcontrolador y un entorno de desarrollo para crear software (programas).

Los módulos de Arduino en sus diferentes versiones sirven para crear objetos interactivos, puede leer una gran variedad de datos, ejemplo interruptor, sensores. En sus salidas puede controlar diferentes dispositivos como luces, motores y otros actuadores físicos.

Los proyectos de Arduino pueden ser autónomos o comunicarse con un programa (software) que se ejecuten desde un ordenador. La placa se puede adquirir desmontada para ensamblarla o comprarla ya lista para usar, el software de desarrollo es abierto y se puede descargar gratis.



FUENTE: <http://ro-botica.com/es/tienda/ARDUINO/>

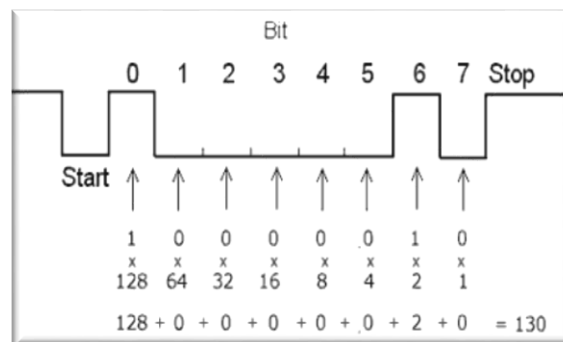
Figura II. 11: Sistema decodificador y control de señal binaria

2.6.1. Comunicación Básica

La comunicación de la tarjeta Arduino se realiza por medio del puerto serial, interactuando con un ordenador o con otros dispositivos a través de esta interfaz. Los pines que se usan para la comunicación por puerto serial vienen marcados en la placa con las siglas RX y TX, que corresponden al pin por el que se enviarán y recibirán los datos.

Series de pulsos

La comunicación puerto serial (asincrónica¹, 8 bits, más un bit de parada) envía un tren de 8 pulsos de voltaje, un tipo de lenguaje máquina como una serie de 8 bit (1 ó 0):



FUENTE: <http://arduino.cc/es/Guide/Introduction>

Figura II. 12: Trama de Bits

Si Arduino lee en un sensor analógico un valor de 65, equivalente a la serie binaria 01000001 esta será enviada, según el modificador, como:

Dato	Modificador	Envío (pulsos)
65	---DEC----	(“6” y “5” ASCIIs 54–55) 000110110–000110111
65	---HEX----	(“4” Y “1” ASCIIs 52–49) 000110100–000110001

1 Asincrónica: Acceso a la red de manera no simultánea

2.6.2. Comunicación vía puerto Serial:

La tarjeta Arduino puede establecer comunicación serie (recibir y enviar valores codificados en ASCII) con un dispositivo externo, a través de una conexión por un cable/puerto USB o cable/puerto serie RS-232.

Igual que para la descarga de los programas, sólo será necesario indicar el número de puerto de comunicaciones que estamos utilizando y la velocidad de transferencia en baudios. También hay que tener en cuenta las limitaciones de la transmisión en la comunicación serie, que sólo se realiza a través de valores con una longitud de 8-bits (1 Byte) (ver `Serial.write ()` o `Serial.read(c)`), mientras el A/D (Convertidor) de Arduino tiene una resolución de 10-bits. (Enlace)

Dentro del interfaz Arduino, disponemos de la opción "Monitorización de Puerto Serie", que posibilita la visualización de datos procedentes de la tarjeta.

Para definir la velocidad de transferencia de datos, hay que ir al menú "Herramientas" (Tools) y seleccionar la etiqueta "Velocidad de monitor Serie"(Tools). La velocidad seleccionada, debe coincidir con el valor que hemos determinado o definido en nuestro programa y a través del comando `Serial.begin()`. Dicha velocidad es independiente de la velocidad definida para la descarga de los programas.

La opción de "Monitorización de puerto serie" dentro del entorno Arduino, sólo admite datos procedentes de la tarjeta.

También se pueden utilizar otros programas para enviar y recibir valores ASCII o establecer una comunicación con Arduino: Processing (enlace), Pure Data (enlace),

Director (enlace), la combinación o paquete serial proxy + Flash (enlace), MaxMSP (enlace), etc.

2.6.3. Instalación del software Arduino

Funciona en Windows, Mac OS X y Linux. El entorno está escrito en Java y basado en Processing, avr-gcc y otros programas también de código abierto.

PASO 1: DESCARGA DEL SOFTWARE ARDUINO

Se puede ingresar a la Pagina Web: <http://arduino.cc/en/Main/Software> y descargarnos el instalador de la página Arduino, no se necesita pagar por la descarga es totalmente gratis.



FUENTE: <http://www.arduino.cc/es/>

Figura II. 13: Descarga de Software Arduino

PASO 2: ABRIR LA CARPETA COMPRIMIDA

Se debe tener instalado el desempaquetado de archivos WinRAR.

Nombre	Fecha de modifica...	Tipo	Tamaño
drivers	17/05/2013 22:24	Carpeta de archivos	
examples	17/05/2013 22:26	Carpeta de archivos	
hardware	17/05/2013 22:26	Carpeta de archivos	
java	17/05/2013 22:26	Carpeta de archivos	
lib	17/05/2013 22:26	Carpeta de archivos	
libraries	17/05/2013 22:25	Carpeta de archivos	
reference	17/05/2013 22:26	Carpeta de archivos	
tools	17/05/2013 22:25	Carpeta de archivos	
arduino.exe	17/05/2013 22:26	Aplicación	840 KB
cygiconv-2.dll	17/05/2013 22:24	Extensión de la apl...	947 KB
cygwin1.dll	17/05/2013 22:24	Extensión de la apl...	1.829 KB
libusb0.dll	17/05/2013 22:24	Extensión de la apl...	43 KB
revisions.txt	17/05/2013 22:24	Documento de tex...	38 KB
ntxSerial.dll	17/05/2013 22:24	Extensión de la apl...	76 KB

FUENTE: <http://www.arduino.cc/es/>

Figura II. 14: Ejecutable del Paquete de Instalación

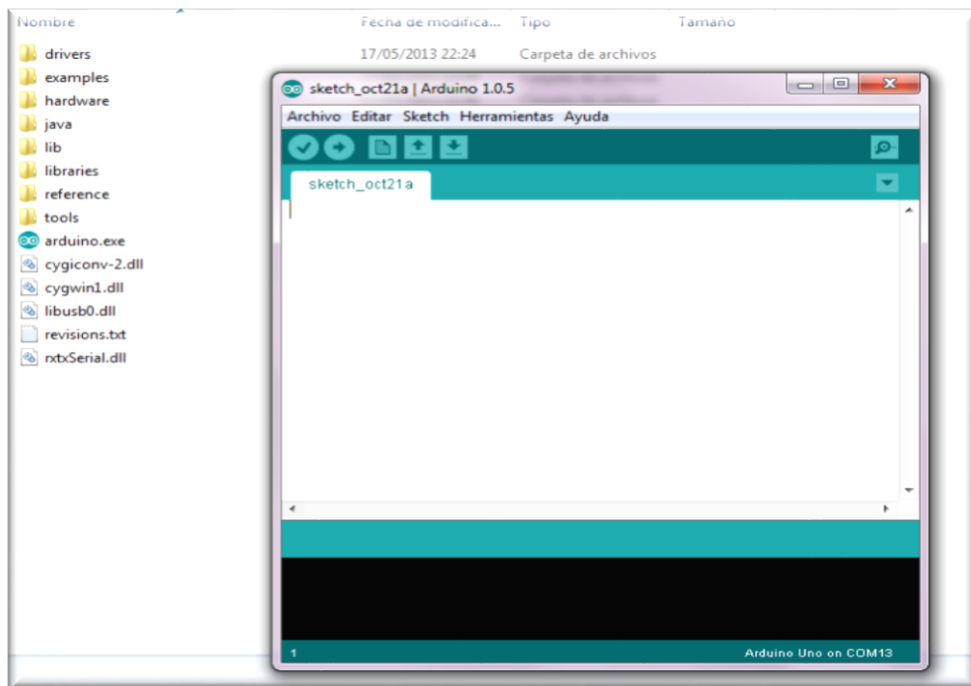
PASO 3: EJECUTAR EL INSTALADOR (arduino.exe)

Es importante ejecutar Arduino.exe como administrador.



FUENTE: <http://www.arduino.cc/es/>

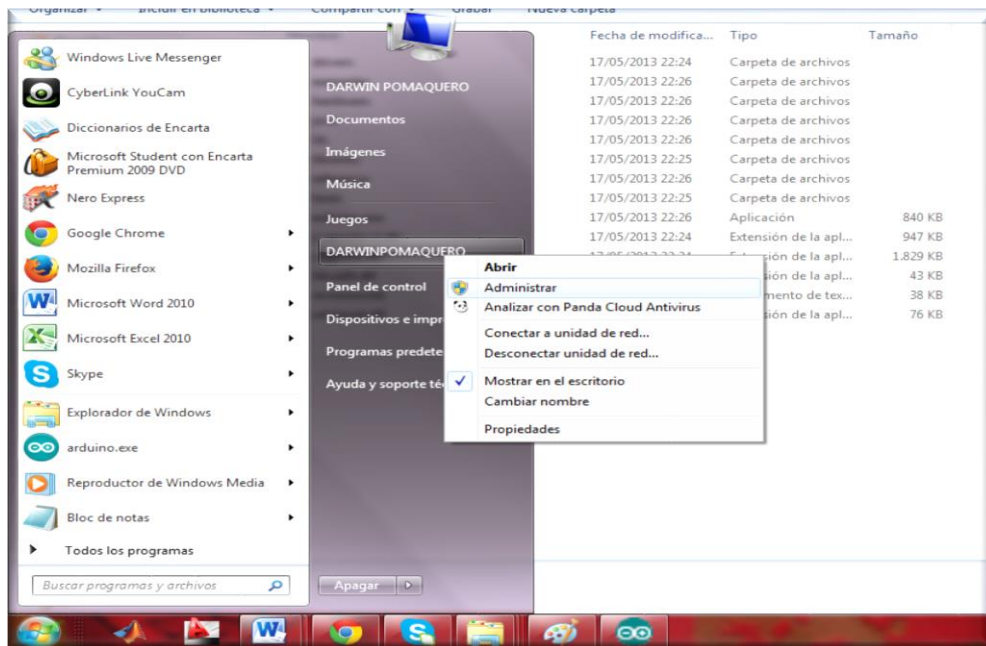
Figura II. 15: Ejecutar en modo de administrador



FUENTE: <http://www.arduino.cc/es/>

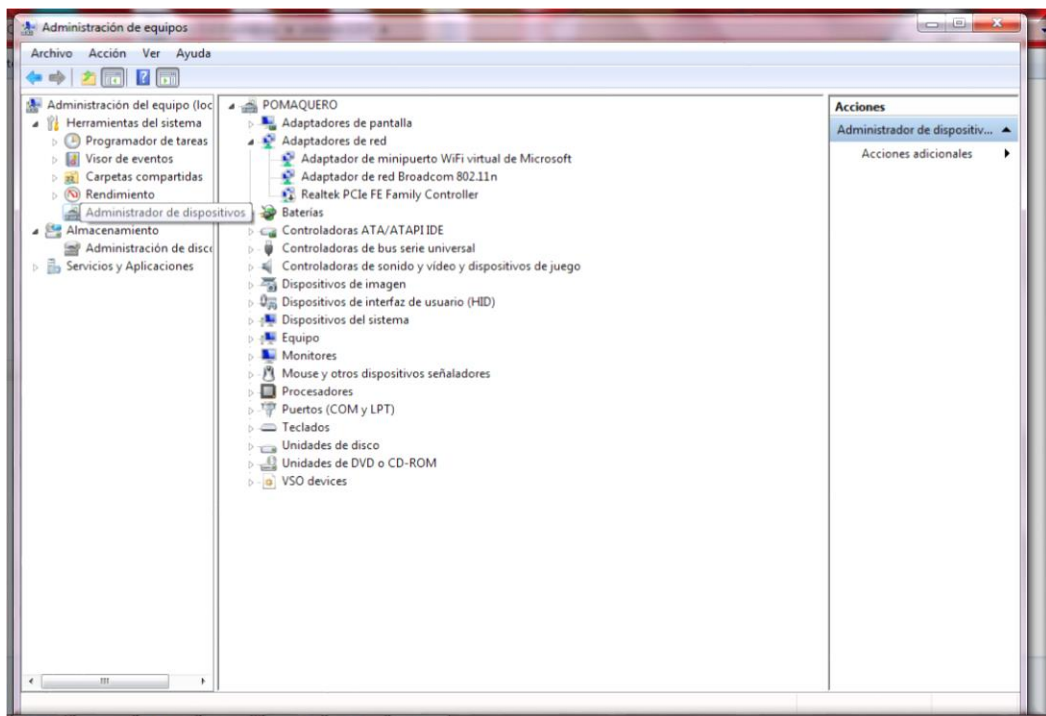
Figura II. 16: Exploración del Software

PASO 4: COMPROBAR EL PUERTO SERIAL



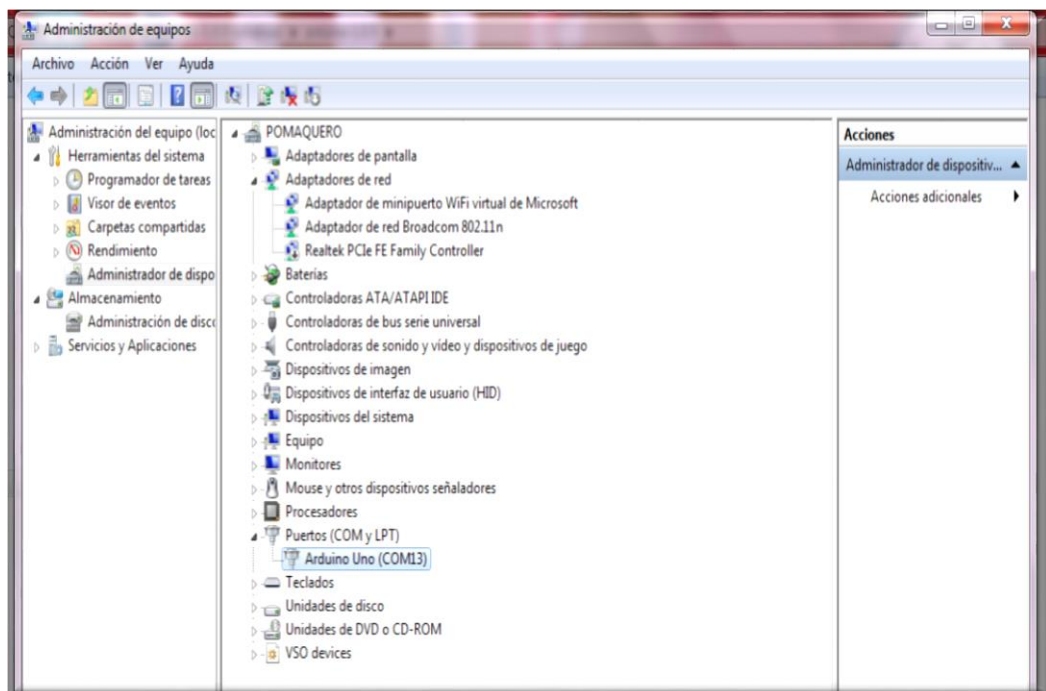
FUENTE: <http://www.arduino.cc/es/>

Figura II. 17: Comprobación del Puerto Serial



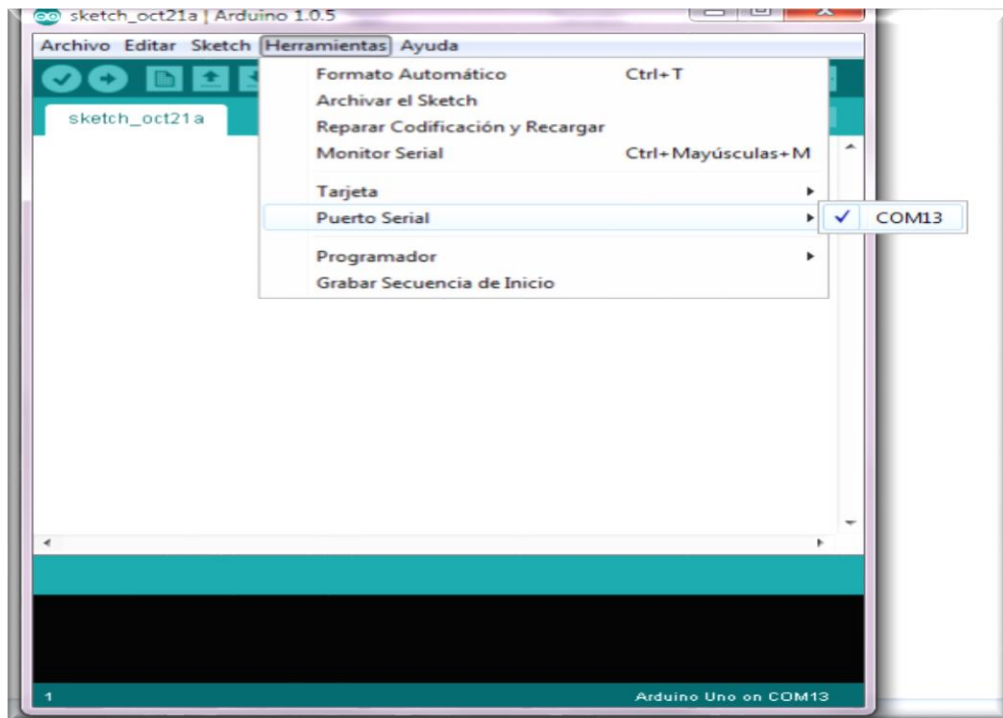
FUENTE: <http://www.arduino.cc/es/>

Figura II. 18: Ingreso al administrador de dispositivos



FUENTE: <http://www.arduino.cc/es/>

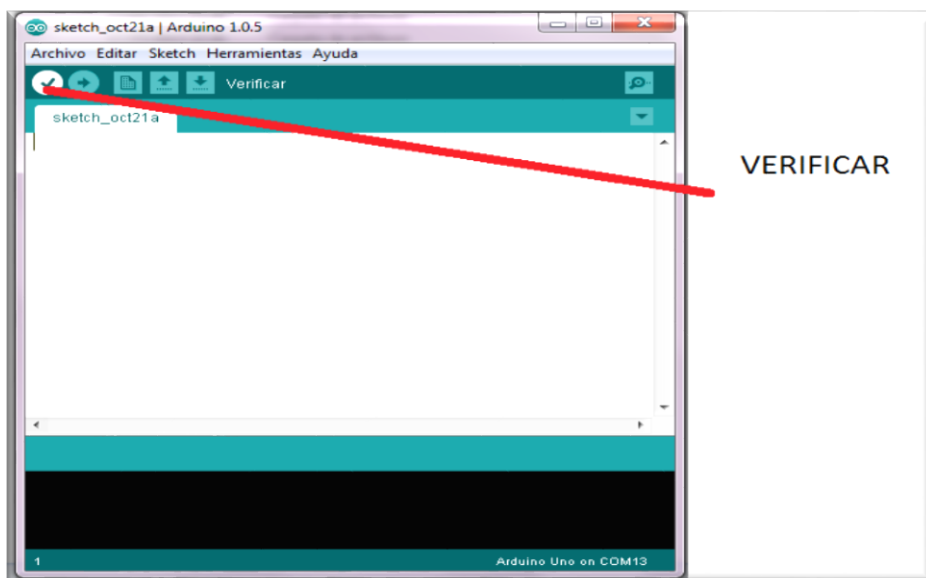
Figura II. 19: Verificación en el puerto COM



FUENTE: <http://www.arduino.cc/es/>

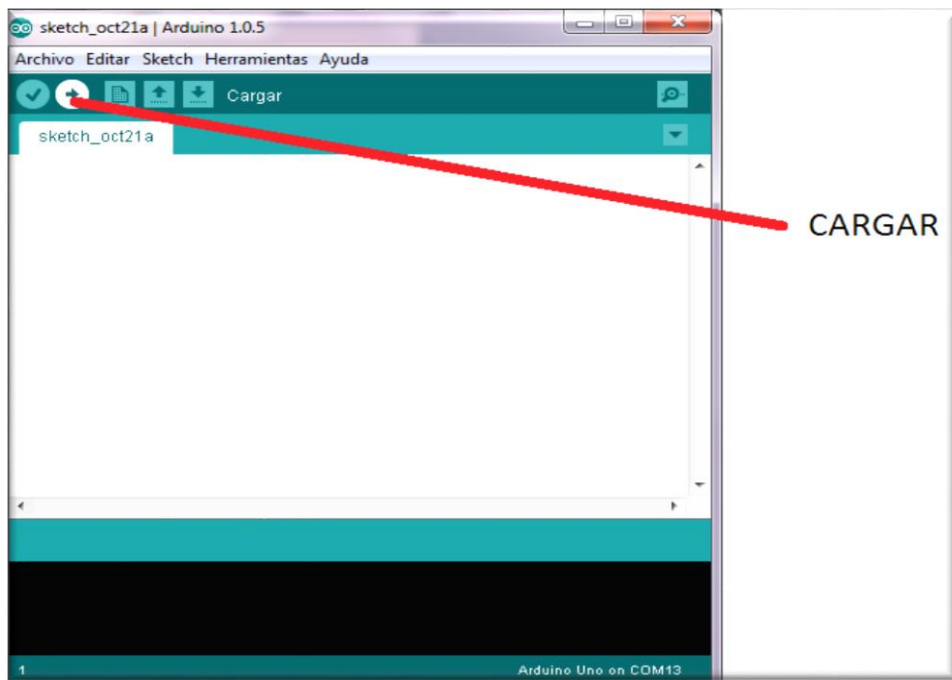
Figura II. 20: Ejecución del Software

PASO 5: ENTORNO DE DESARROLLO



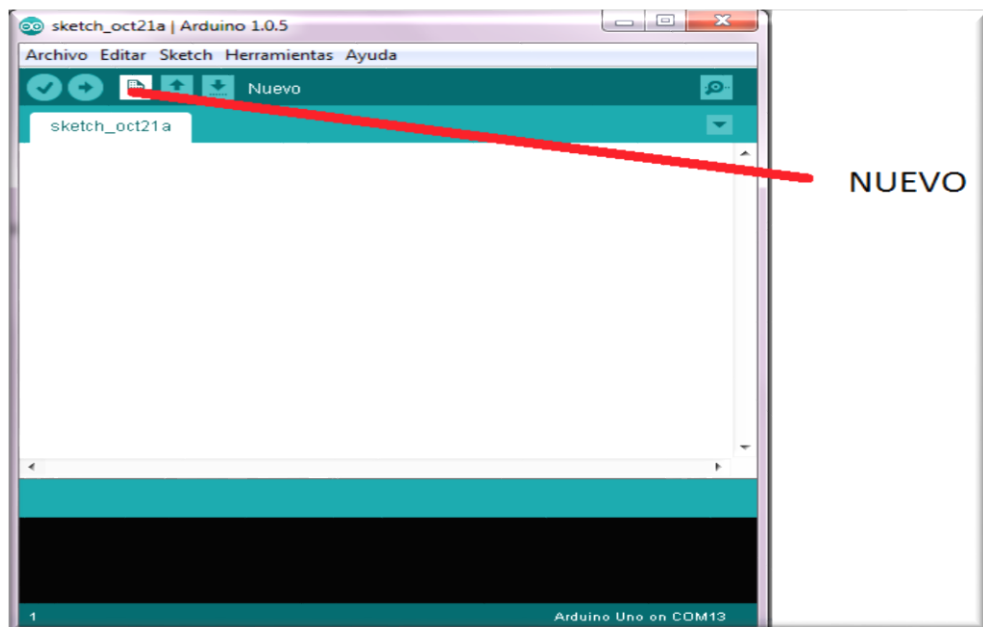
FUENTE: <http://www.arduino.cc/es/>

Figura II. 21: Entorno de desarrollo



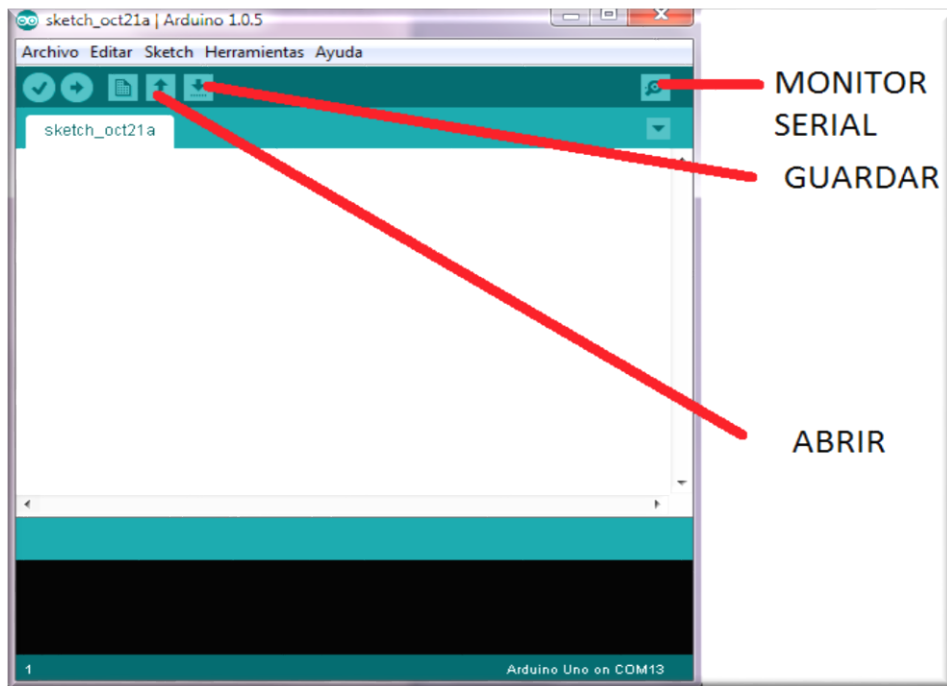
FUENTE: <http://www.arduino.cc/es/>

Figura II. 22: Revisión de iconos en Arduino 10.5



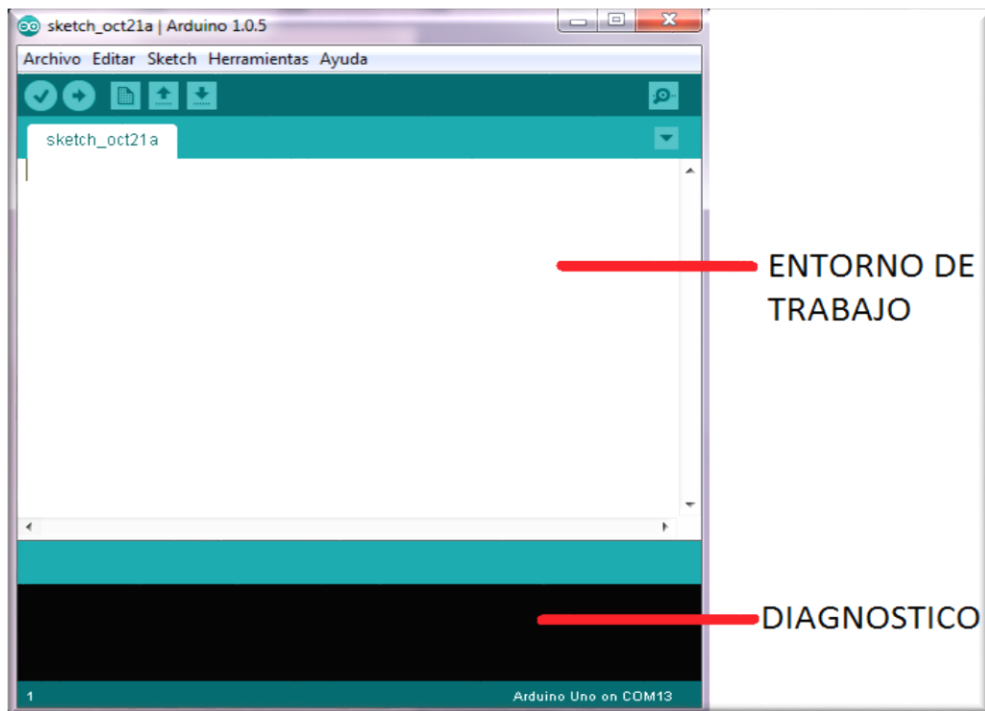
FUENTE: <http://www.arduino.cc/es/>

Figura II. 23: Crear nuevo proyecto



FUENTE: <http://www.arduino.cc/es/>

Figura II. 24: Reconocimiento del entorno de programación



FUENTE: <http://www.arduino.cc/es/>

Figura II. 25: Entorno de trabajo Arduino 10.5

2.6.4. Programación en Arduino

Una mirada en la referencia de Arduino permitirá constatar que las funciones `print` y `println` (lo mismo que la anterior pero con salto de renglón) tienen opcionalmente un modificador que puede ser de varios tipos:

```
Serial.print(data, DEC); // decimal en ASCII
```

```
Serial.print(data, HEX); // hexadecimal en ASCII
```

```
Serial.print(data, OCT); // octal en ASCII
```

```
Serial.print(data, BIN); // binario en ASCII
```

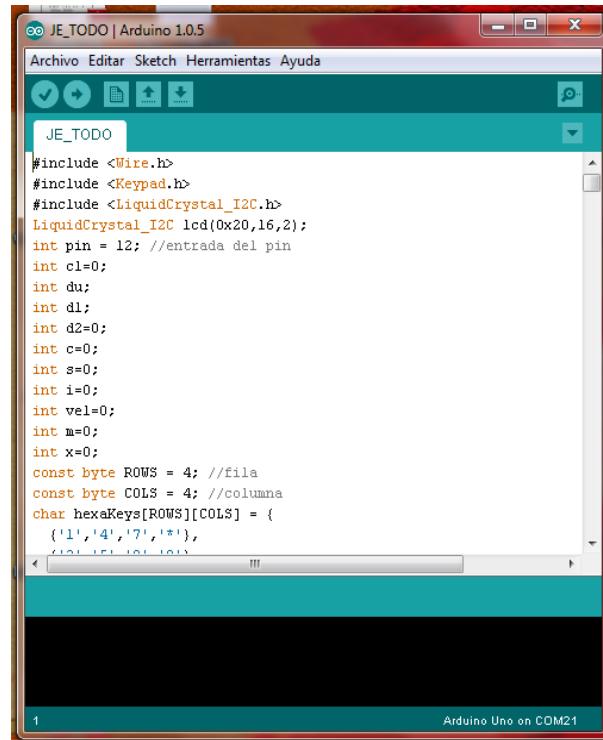
```
Serial.print(data, BYTE); // un Byte
```

Si no utilizamos ningún modificador para el `Serial.println` es lo mismo que si utilizáramos el modificador `DEC`. Así que no estamos utilizando el modo más eficiente pero sí el más fácil de leer en el mismo Arduino. Al ejecutar este programa podremos inmediatamente abrir el monitor serial del software Arduino (último botón a la derecha).

2.6.5. Lenguaje de texto estructurado

Es un lenguaje de programación de alto nivel similar a Basic, Fortran, Pascal y "C". Se lo utiliza en procesos mayormente cuando se requiere hacer un cálculo matemático complejo el cual requiere exactitud en los datos. Utilizado por usuarios o programadores avanzados.

Los programas hechos con Arduino se dividen en tres partes principales: estructura, valores (variables y constantes), y funciones. El Lenguaje de programación Arduino se basa en C/C++.



```

JE_TODO | Arduino 1.0.5
Archivo Editar Sketch Herramientas Ayuda
JE_TODO
#include <Wire.h>
#include <Keypad.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x20,16,2);
int pin = 12; //entrada del pin
int c1=0;
int du;
int d1;
int d2=0;
int c=0;
int s=0;
int i=0;
int vel=0;
int m=0;
int x=0;
const byte ROWS = 4; //fila
const byte COLS = 4; //columna
char hexaKeys[ROWS][COLS] = {
  {'1','4','7','*'},
  {'2','5','8','0'},
  {'3','6','9','-'},
  {'0','1','2','3'}
};
1 Arduino Uno on COM21

```

FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura II. 26: Programación en texto estructurado

Estructuras de control

- if (comparador si-entonces)
- if...else (comparador si...sino)
- for (bucle con contador)
- switch case (comparador múltiple)
- while (bucle por comparación booleana)
- do... while (bucle por comparación booleana)
- break (salida de bloque de código)
- continue (continuación en bloque de código)
- return (devuelve valor a programa)

CAPITULO III

3. DISEÑO DEL SISTEMA DE CONTROL DE VELOCIDAD

3.1. INTRODUCCIÓN

En este capítulo detallaremos el diseño e implementación de un control de velocidad para el bus de la FIE en la ciudad de Riobamba, especificando las diferentes partes que la conforman. Este sistema puede ser visto por el chofer del vehículo cuando la velocidad se va incrementando, al llegar al límite establecido nos dará una señal audible y visible denotando que el conductor ha llegado a su límite, este límite dependerá del recorrido establecido que lo asignara el administrador, para circulación en zonas urbanas será 40 km/h o en perimetral a 90 km/h.

Si la velocidad del vehículo excede los límites establecidos automáticamente se enviara una señal al solenoide quien activara la electroválvula 5/2 para que se accione el cilindro neumático para que interrumpa la aceleración inmediatamente, se desactivara

cuando la velocidad del vehículo sea reducida hasta un rango inferior del límite de velocidad.

Realizaremos un modelo particular que puede ser implementado en cualquier automotor a diésel, siendo diferente el modelo a gasolina.

3.2. Por qué instalar una señales de alerta

El cuerpo humano reacciona ante los estímulos externos, tanto físicos como químicos o electromagnéticos, que desencadenan reacciones funcionales en el organismo. Para recibir esos estímulos el cuerpo se vale de los sentidos: los oídos, la nariz, la boca, los ojos y la piel.

Por tal razón era necesario instalar dispositivos audibles (chicharra) y visibles (luces) para alertar al usuario cuando este se esté excediendo el límite de velocidad, para indicar que el sistema de control de velocidad accionara el freno y el bloqueo del acelerador.

3.3. Por qué utilizar un Actuadores Neumático

Aunque en esencia los actuadores neumáticos e hidráulicos son idénticos, los Neumáticos tienen un mayor rango de compresión y además existen diferencias.

En cuanto al uso y estructura. Considerando que la presión necesaria para la activación el cilindro neumático es de 65 PSI, con la que podemos fácilmente elevar el pedal del acelerador que dando de tal manera bloqueada; Una vez localizado los tanques de aire del autobús los cuales generan una presión de 220 PSI(Libras por pulgada cuadrada), decidimos utilizar una electroválvula con su respectivo cilindro neumático.

3.4. Por qué se utilizó la tarjeta Arduino Uno

Hay muchos otros microcontroladores y plataformas con microcontroladores disponibles para la computación física. Parallax Basic Stamp, BX-24 de Netmedia, Phidgets, Handyboard del MIT, y muchos otros ofrecen funcionalidades similares. Todas estas herramientas organizan el complicado trabajo de programar un microcontrolador en paquetes fáciles de usar.

Arduino, además de simplificar el proceso de trabajar con microcontroladores, ofrece algunas ventajas respecto a otros sistemas a profesores, estudiantes y amateurs:

Asequible - Las placas Arduino son más asequibles comparadas con otras plataformas de microcontroladores.

Multi-Plataforma - El software de Arduino funciona en los sistemas operativos Windows, Macintosh OSX y Linux. La mayoría de los entornos para microcontroladores están limitados a Windows.

Software ampliable y de código abierto- El software Arduino está publicado bajo una licencia libre y preparado para ser ampliado por programadores experimentados. El lenguaje puede ampliarse a través de librerías de C++, y si se está interesado en profundizar en los detalles técnicos, se puede dar el salto a la programación en el lenguaje AVR C en el que está basado. De igual modo se puede añadir directamente código en AVR C en tus programas si así lo deseas.

Hardware ampliable y de Código abierto - Arduino está basado en los microcontroladores ATMEGA168, ATMEGA328 y ATMEGA1280. Los planos de los módulos están publicados bajo licencia Creative Commons, por lo que diseñadores de

circuitos con experiencia pueden hacer su propia versión del módulo, ampliándolo u optimizándolo.

Incluso usuarios relativamente inexpertos pueden construir la versión para placa de desarrollo para entender cómo funciona y ahorrar algo de dinero.

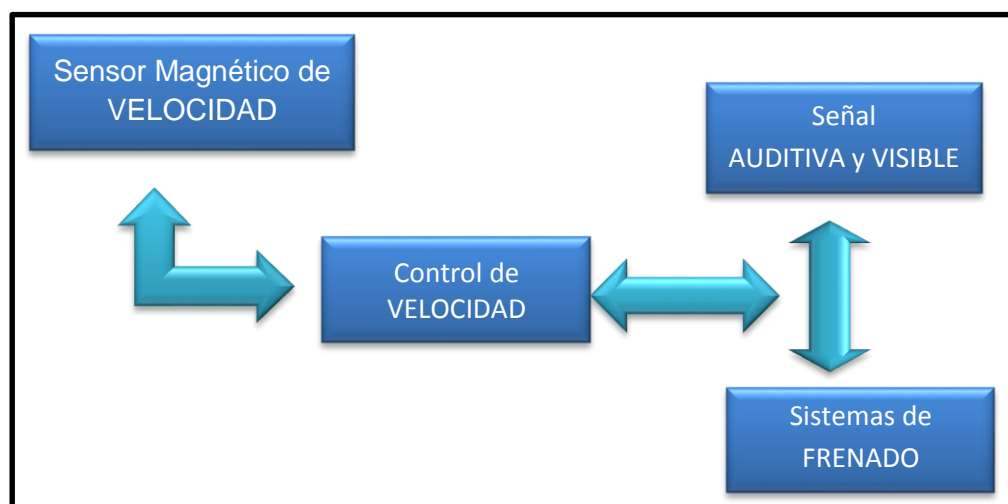
3.5. Diagrama de bloques del sistema de control

El diagrama de bloques de la figura.V.26 muestra todas las etapas que debe cumplir el sistema de control de velocidad. Cuenta con las etapas de entrada, control, actuador.

Etapas del Sistema de control de velocidad:

La entrada de datos será mediante un sensor digital que enviara un tren de pulsos cuando se incremente la velocidad del vehículo, entregándome un tren de pulsos para nuestra tarjeta de adquisición que ejercerá en control de velocidad, la cual enviará una señal a nuestro circuito de potencia que realizará el frenado del bus mediante el freno mecánico a través del circuito de potencia.

Si excede el límite permitido paralelo al sistema de frenado se encenderá un sistema audible (pito) y una señal visible una luz intermitente.



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 27: Diagrama de bloques del sistema

3.5.1. Bloque de entrada de datos

Hemos definido como entrada de datos a la señal binaria que nos enviará el sensor VSS colocado en el vehículo a prueba. La generación de señales binarias es importante en el desarrollo de cualquier dispositivo electrónico, es así que el generador de señales se utiliza para proporcionar condiciones de prueba conocidas para la evaluación y verificación de las señales faltantes en sistemas como el de control de velocidad.

Para la generación de estas señales existen varios tipos de generadores de señales, los cuales tienen diversas características en común:

- Primero la frecuencia de la señal debe ser estable y conocerse con exactitud.
- Segundo se ha de controlar la amplitud, desde valores muy pequeños hasta relativamente altos. Por último, la señal debe estar libre de distorsión.

La entrada de datos nos enviara el sensor de velocidad del vehículo (VSS), que en nuestro caso actuará en paralelo con nuestro sistema, para modificar las funciones del motor y poner en marcha rutinas de diagnóstico.



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 28: Sensor de velocidad integrado en el Bus de la FIE

Tiene en su interior un imán giratorio que genera una onda senoidal de corriente alterna directamente proporcional a la velocidad del vehículo. Por cada vuelta del eje genera 8 ciclos, su resistencia debe ser de 190 a 240 Ohmios. Este sensor nos enviara datos en paralelo tanto para el tacómetro como para nuestro sistema. Con un voltímetro de corriente alterna se verifica la tensión a la salida del sensor, estando desconectado y poniendo a girar una de las ruedas motrices a una velocidad de 40 km/h.



FUENTE: <http://expoypracticass2bmmarturo.blogspot.com/2012/06/sensor-vss-sensor-de-velocidad-el-auto.html>

Figura III. 29: Sensor control de velocidad tipo Vss

El voltaje a la salida del sensor de velocidad deberá estar en un rango de 7.2 a 12 V.

3.5.2. Bloque de control

En el bloque de control la señal binaria será decodificada por nuestra tarjeta arduino de adquisición de datos, la señal que ingresa es de 12 Vdc, por cada vuelta del eje genera 8 ciclos. Se debe conectar la entrada de la señal del sensor Vss al pin 12 de la tarjeta Arduino. En la etapa de control se maneja señales de entradas y salidas, las señales del sensor que ingresan son decodificadas por la tarjeta Arduino. Por medio del pin 12

recibe la señal de frecuencia (Tren de pulsos), mediante la relación de la tabla. VI.3 se enviara la señal de control a la electroválvula y freno de máquina.

Se puede utilizar los pines del Arduino a donde están conectados los Reles:

Pin Arduino	Relé
5	RB0
6	RB1
7	RB2
8	RB3
9	RB4
10	RB5
11	RB6
12	RB7

FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

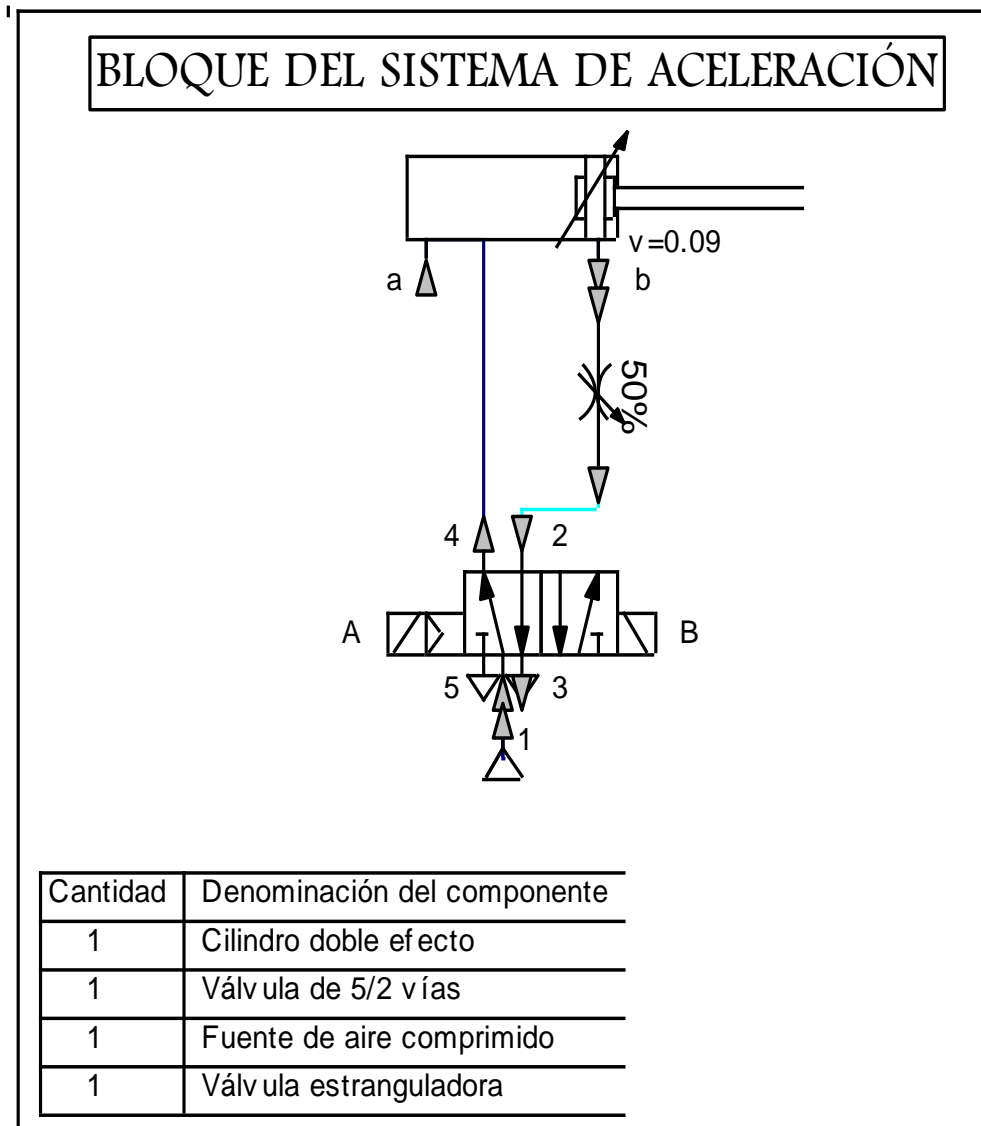
Tabla III. 2: Designación de Pines de la tarjeta Arduino

Esta señal es tomada para realizar la comparación a los cambios de velocidad. Mediante una comparación se realiza la variable a visualizar, se activa los actuadores, Utilizamos el pin 13 para el control del freno de máquina mediante un relé de 12 Vdc, por medio del pin 10 controlamos luces de aviso cuando se ha excedido del límite de velocidad. Por medio del pin 11 controlamos el bloqueo del acelerador, el cual se desactivara cuando la velocidad se haya reducido al límite permitido.

3.5.3. Bloque de actuador

En este bloque se realizará el control de frenado para reducir la velocidad del bus, con los parámetros antes mencionados, cuando se exceda del límite de velocidad la etapa de potencia enviará una señal al relé para active el freno de máquina hasta llegar a

una velocidad inferior del límite establecido. La desactivación será inmediata cuando el sistema llegue a una velocidad inferior del 5 % del límite de sanción



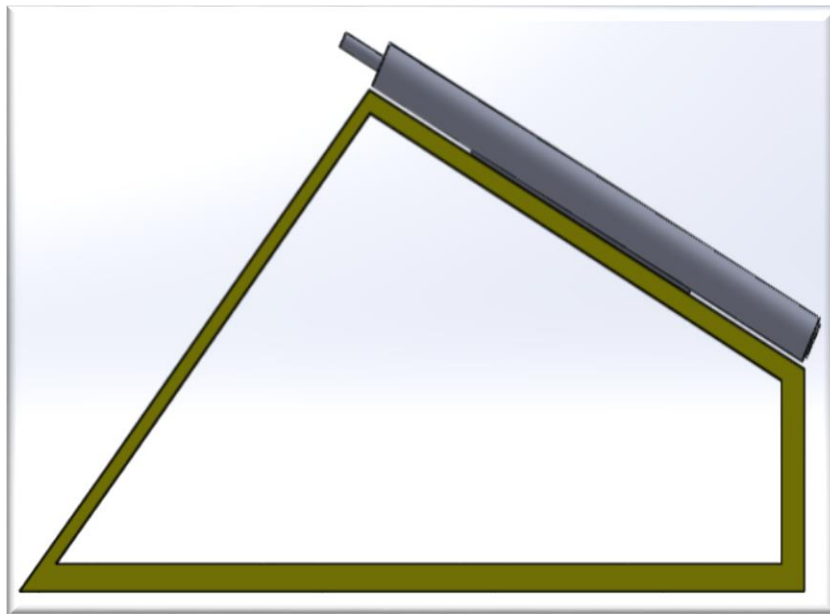
FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 30: Simulación del sistema de bloqueo del acelerador

Se debe bloquear a la vez el sistema de aceleración para que no exista interferencia con el sistema de freno y pueda desarrollarse de acuerdo a las especificaciones técnicas. Mediante una válvula neumática 5/2 se realizará el control de paso del aire,

esta válvula se activará cuando se exceda del límite de velocidad, habilitando el paso de aire al cilindro neumático, bloqueando el acelerador de forma inmediata.

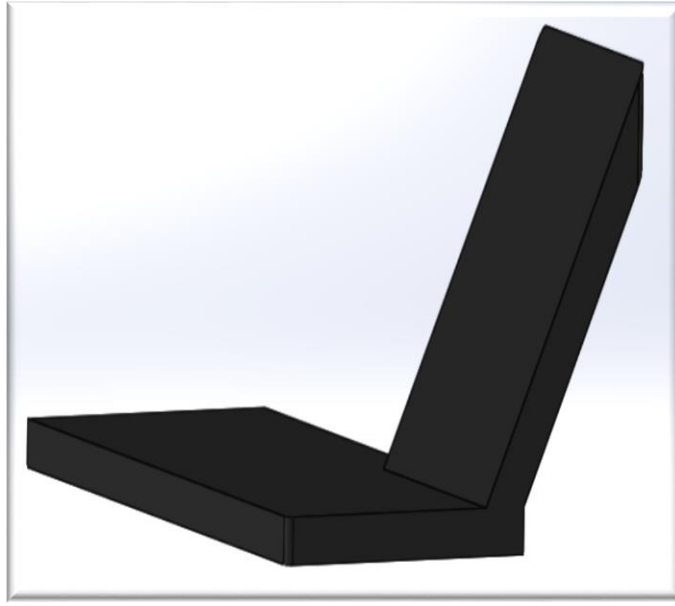
Para establecer el control de la velocidad en el sistema propuesto, se realizará por medio del freno de máquina, cuando se active este control el acelerador debe estar totalmente bloqueado, La señal transmitida desde la tarjeta arduino en el bloque de control, permitirá que la electroválvula se accione, entrando en funcionamiento el cilindro neumático, levantando el pie del conductor del acelerador y bloqueando asta llegar a una velocidad permitida. En paralelo se activa el freno de maquina del bus.



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 31: Diseño de Bloqueo al sistema de aceleración

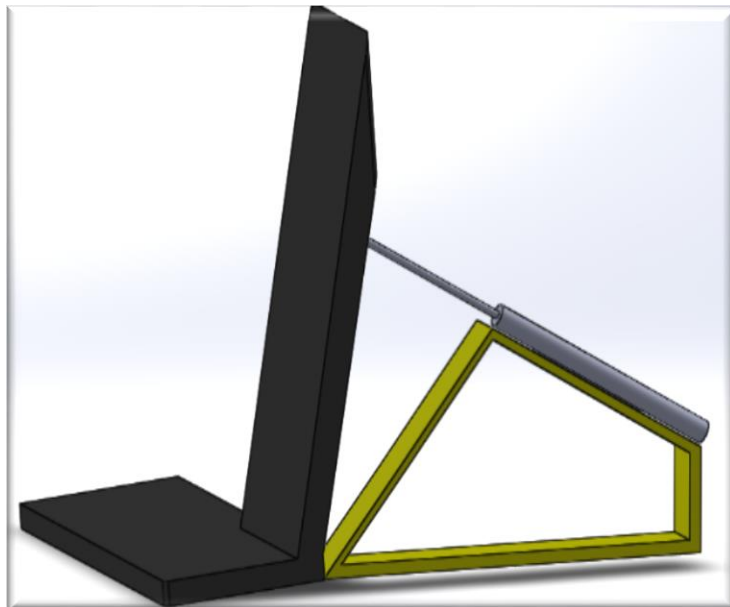
Para obtener un diseño final del sistema de bloqueo, se realizó diferentes pruebas de posicionamiento, debemos tomar en cuenta el grado de inclinación al momento de realizar el diseño de bloqueo del acelerador, es importante tomar las dimensiones entre la base y el acople del acelerador para seleccionar el actuador neumático adecuado, con esto evitaremos la alteración del diseño original en el sistema de aceleración.



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 32: Adaptación de bloqueo al acelerador

Bloqueo final del sistema de aceleración. Es importante tener presión en el cilindro neumático para que se bloquee el acelerador



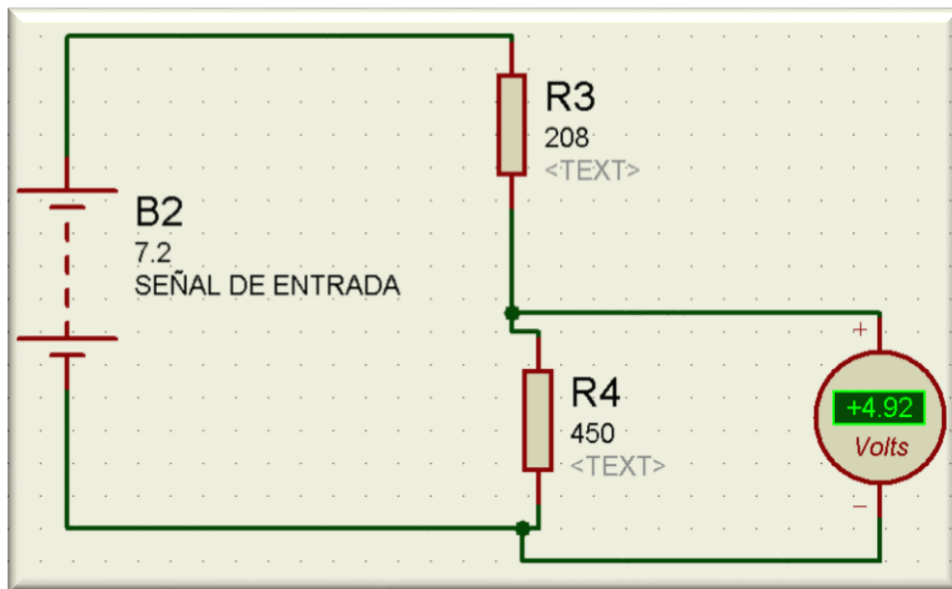
FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 33: Construcción del sistema de bloqueo

3.6. Diseño de la Etapa de potencia en el Bloque del Actuador

3.6.1. Diseño del divisor de voltaje

Para alimentar al sistema de control de velocidad, se ha tomado la alimentación suministrada por el banco de baterías del bus, el cual consta de dos baterías de 12 VDC. Conectadas en serie dotará un voltaje de 24 VDC. Se aplicó un divisor de voltaje en la entrada de alimentación al sistema, para suministrar la tensión requerida del sistema. En el divisor de voltaje la salida dependerá de la resistencia de la carga que alimenta.



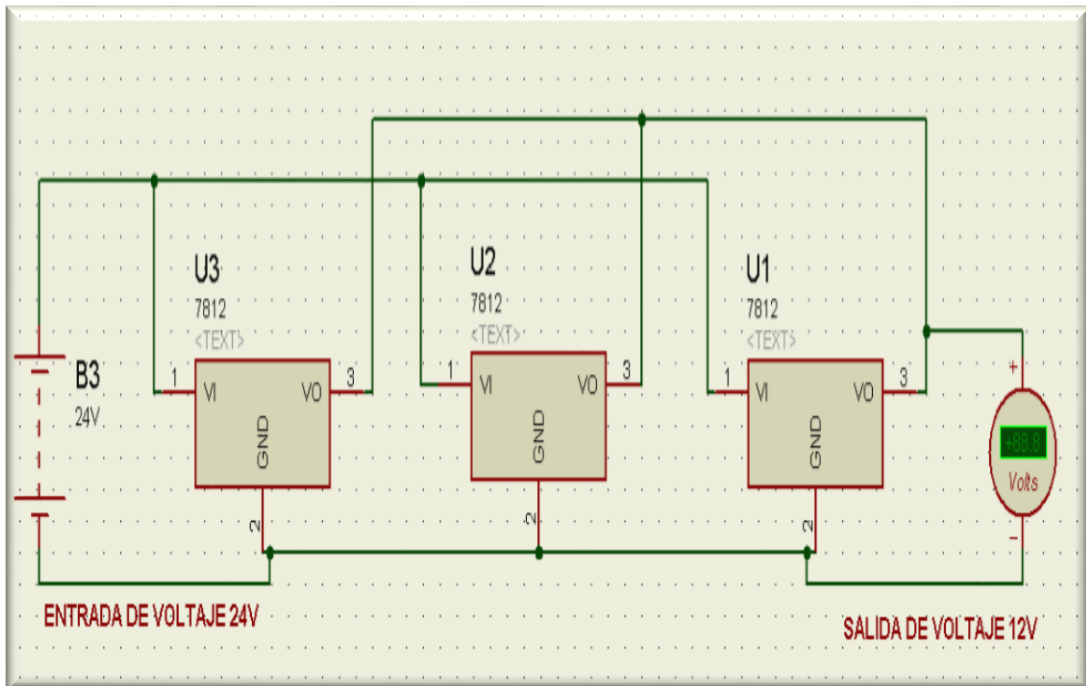
FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 34: Simulación divisor de voltaje

Divisor de voltaje: Se tiene una fuente $V_{Input} = 7.2$ VDC. Para proporcionar la energía y dos resistencias R1 y R2. . Cualquier combinación de R1=208 con R2=450 formara un divisor de voltaje.

Divisor de voltaje solo sirve para analizar dos resistencias, con la finalidad de obtener una tensión menor a partir de una mayor.

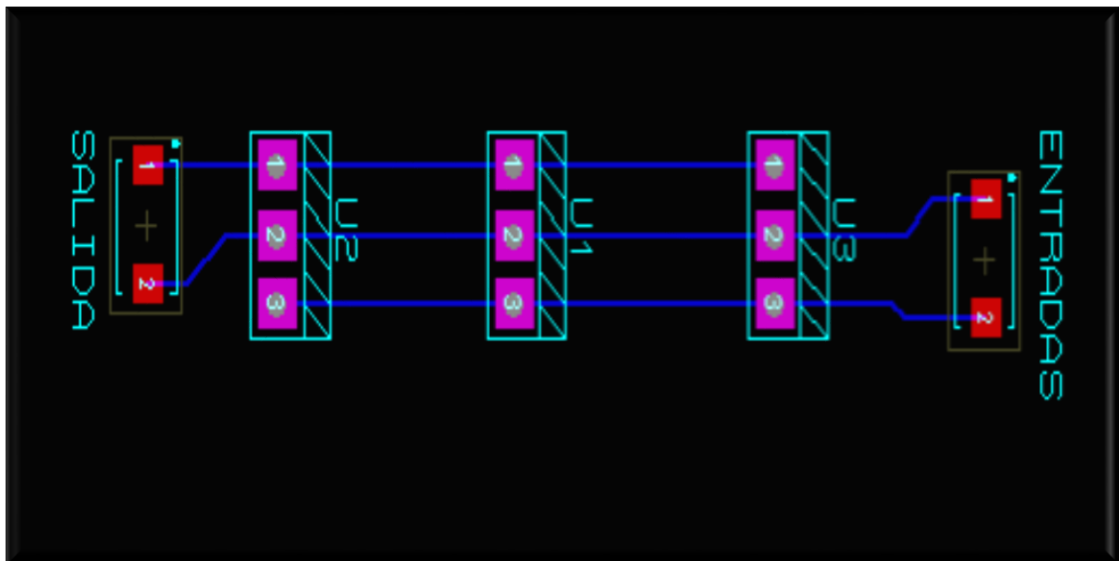
SIMULACIÓN DEL REDUCTOR DE VOLTAJE



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 36: Simulación del reductor de velocidad

DISEÑO DE LA PLACA DEL REDUCTOR DE VOLTAJE



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura III. 37: Diseño del reductor de velocidad

DIAGRAMA DE FLUJO

PASO 1: Debemos verificar que las librerías que vienen por defecto nos sean útiles, se necesitamos alguna librería adicional tenemos que descargarlos y validar con nuestro software de programación

//////////////////// DECLARACIÓN DE LAS VARIABLES Y LIBRERIAS //////////////////////

```
#include <Wire.h>
#include <Keypad.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x20,16,2);
int pin = 12; //entrada del pin
int du;
int d1;
int d2=0;
int c=0;
int s=0;
int i=0;
int vel=0;
int m=0;
int x=0;
const byte ROWS = 4; //fila
const byte COLS = 4; //columna
char hexaKeys[ROWS][COLS] = {
    {'1','4','7','*'},
    {'2','5','8','0'},
```

```

    {'3','6','9','#'},
    {'A','B','C','D'}
};

byte rowPins[ROWS] = {3, 2, 9, 8}; //fila

byte colPins[COLS] = {7, 6, 5, 4}; //columna

Keypad customKeypad = Keypad(makeKeymap(hexaKeys), rowPins,
colPins, ROWS, COLS);

```

PASO 2: Debemos seleccionar los puertos que vamos a utilizar para nuestra programación ya sea de entrada como de salida. Se recomienda verificar que los puertos no estén averiados para obtener una toma de datos seguros y confiables.

```

////////////////////////////////// ENTRADA DE SEÑAL //////////////////////////////////
    du = pulseIn (pin,HIGH);      //LEE UN PULSO ALTO EN UN PIN
    d1=du/10;                      //DIVIDE LA LECTURA DEL PULSO
    if((d1>=1767)&&(d1<=1835)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
    {
        d2=5;                      //SI ES IGUA LA VARIABLE ES 5Km
    }
    if((d1>=1100)&&(d1<=1200)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
    {
        d2=10;                      //SI ES IGUA LA VARIABLE ES 10Km
    }
    if((d1>=800)&&(d1<=850)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
    {
        d2=15;                      //SI ES IGUA LA VARIABLE ES 15Km
    }
    if((d1>=615)&&(d1<=620)) //CONTROL DE LA VARIABLE DE
                                ENTRADA

```

```
{
d2=20;           //SI ES IGUA LA VARIABLE ES 20Km
}

if((d1>=488)&&(d1<=490)) //CONTROL DE LA VARIABLE DE
                        ENTRADA

{
d2=25;           //SI ES IGUA LA VARIABLE ES 25Km
}

if((d1>=418)&&(d1<=420)) //CONTROL DE LA VARIABLE DE
                        ENTRADA

{
d2=30;           //SI ES IGUA LA VARIABLE ES 30Km
}

if((d1>=355)&&(d1<=360)) //CONTROL DE LA VARIABLE DE
                        ENTRADA

{
d2=35;           //SI ES IGUA LA VARIABLE ES 35Km
}

if((d1>=310)&&(d1<=315)) //CONTROL DE LA VARIABLE DE
                        ENTRADA

{
d2=40;           //SI ES IGUA LA VARIABLE ES 40Km
}

if((d1>=275)&&(d1<=280)) //CONTROL DE LA VARIABLE DE
                        ENTRADA

{
d2=45;           //SI ES IGUA LA VARIABLE ES 45Km
```

```

}
if((d1>=245)&&(d1<=250)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
{
d2=50;                            //SI ES IGUA LA VARIABLE ES
                                50Km
}
if((d1>=220)&&(d1<=225)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
{
d2=55;                            //SI ES IGUA LA VARIABLE ES
                                55Km
}
if((d1>=210)&&(d1<=215)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
{
d2=60;                            //SI ES IGUA LA VARIABLE ES
                                60Km
}
if((d1>=195)&&(d1<=200)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
{
d2=65;                            //SI ES IGUA LA VARIABLE ES 65Km
}
    if((d1>=175)&&(d1<=180)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
    {
d2=70;                            //SI ES IGUA LA VARIABLE ES
                                70Km
    }
if((d1>=173)&&(d1<=174)) //CONTROL DE LA VARIABLE DE
                                ENTRADA
{
d2=75;                            //SI ES IGUA LA VARIABLE ES
                                75Km
}

```

```

}
if((d1>=156)&&(d1<=161)) //CONTROL DE LA VARIABLE DE
                          ENTRADA
{
d2=80;                    //SI ES IGUA LA VARIABLE ES
                          80Km

}

if((d1>=150)&&(d1<=155)) //CONTROL DE LA VARIABLE DE
                          ENTRADA
{
d2=85;                    //SI ES IGUA LA VARIABLE ES
                          85Km

}

if((d1>=140)&&(d1<=145)) //CONTROL DE LA VARIABLE DE
                          ENTRADA
{
d2=90;                    //SI ES IGUA LA VARIABLE ES
                          90Km

```

PASO 3: En este bloque de programación, se recomienda una presentación amigable con el usuario, en el que cualquier persona pueda entender y ejecutar el a cabalidad la instrucciones propuestas por el programador del sistema.

//////////////////////////////////// VISUALIZACIÓN //////////////////////////////////////

```

lcd.setCursor(0, 0); //POSICIÓN DEL CURSOR
lcd.print("V:");    //ESCRIBA EN LA PANTALLA LA LETRA
lcd.setCursor(2, 0); //POSICIÓN DEL CURSOR
lcd.print(d2);      //ESCRIBA EN LA PANTALLA LA VARIABLE
lcd.print("Km/h");  //ESCRIBA EN LA PANTALLA LAS LETRAS
lcd.setCursor(10, 0); //POSICIÓN DEL CURSOR
lcd.print("Max:");  //ESCRIBA EN LA PANTALLA LAS LETRAS
lcd.print(vel);     //ESCRIBA EN LA PANTALLA LA VARIABLE

```



```

if(vel==40){          // CONTROL DE LA VARIABLE
                      VELOCIDAD_MAX

lcd.setCursor(5, 1);

lcd.print("URBANO");

}

if(vel==90){         // CONTROL DE LA VARIABLE
                      VELOCIDAD_MAX

lcd.setCursor(3, 1);

lcd.print("PERIMETRAL");

}

```

//////////////////////////////////// ADMINISTRADOR //////////////////////////////////////

```

if(customKey=='1'){ // CONTROL DE LA VARIABLE_TECLEADA
s=0;                //INICIALIZACION DEL CONTADOR

lcd.clear();        //LIMPIA LA PANTALLA

while (s!=2){      //BUCLE HASTA CUMPLIR SENTENCIA

lcd.setCursor(0, 0); //POSICIÓN DEL CURSOR

lcd.print("INGRESE_CLAVE");//ESCRIBA EN LA PANTALLA

char customKey = customKeypad.getKey();//LEER _ ENTRAD
                      _TECLADO

if (customKey != NO_KEY){ //CONTROL_BARRIDO_TECLEADA

lcd.setCursor(i,1);  //POSICIÓN DEL CURSOR

lcd.print("*");

i=i+1;              //SUMA EL CONTADOR

if(customKey=='D'){ //CONTROL _VARIABLE

i=0;                //INICIALIZACION_CONTADOR

```

```
s=2;           //INICIALIZACION_CONTADOR
lcd.clear();   //LIMPIA PANTALLA
```

```
////////////////////////////////// CONTROL DE LA VARIABLE CLAVE //////////////////////////////////////
```

```
if(customKey=='2'){ //CONTROL_VARIABLE
c=1;           //INICIALIZACIÓN_CONTADOR
}
if(customKey=='4'){ //CONTROL_VARIABLE
if(c==1){     //CONTROL_VARIABLE
c=2;         //CONTADOR
}
}
if(customKey=='5'){ //CONTROL_VARIABLE
if(c==2){     //CONTROL_VARIABLE
c=3;         //CONTADOR
}
}
if(customKey=='1'){ //CONTROL_VARIABLE
if(c==3){
c=4;         //CONTADOR
i=0;        //CONTADOR
}
if(c==4){     //CONTROL_VARIABLE
lcd.clear();  //LIMPIA PANTALLA
lcd.setCursor(4, 0); //POSICION CURSOR
lcd.print("CORRECTO");//ESCRIBA EN PANTALLA
```

```

c=0;           //INICIALIZA CONTADOR
delay(1500);  //PAUSA
lcd.clear();   //LIMPIAR PANTALLA
while (s!=2){  /BUCLE HASTA CUMPLIR SENTENCIA
lcd.setCursor(0, 0); //POSICIÓN DEL CURSOR
lcd.print("1._URBANO");//ESCRIBA EN PANTALLA
lcd.setCursor(0, 1);
lcd.print("2._PERIMETRAL");
char customKey = customKeypad.getKey();//LEER TECLADO
if (customKey != NO_KEY){//CONTROL DEL TECLADO

```

//////////////////////////////////// TIPO DE CARRETERAS //////////////////////////////////////

```

if(customKey=='1'){ //CONTROL DE LA VARIABLE
lcd.clear();        //LIMPIA PANTALLA
lcd.setCursor(5, 0); //POSICIÓN DEL CURSOR
lcd.print("URBANO"); //ESCRIBE EN PANTALLA
lcd.setCursor(0, 1); //POSICIÓN DEL CURSOR
lcd.print("40 Km/h"); //ESCRIBE EN PANTALLA
delay(2000);        //PAUSA
s=2;                //CONTADOR
vel=40;             //VARIABLE VELOCIDAD_MAX
lcd.clear();        //LIMPIA PANTALLA
if(customKey=='2'){ //CONTROL DE VARIABLE
lcd.clear();
lcd.setCursor(3, 0);
lcd.print("PERIMETRAL");

```

```

lcd.setCursor(0, 1);
lcd.print("90 Km/h");
delay(2000);          //PAUSA
s=2;                  //CONTADOR
vel=90;               //VARIABLE VELOCIDAD_MAX
lcd.clear();
}

```

//////////////////////////////////// CLAVE INCORRECTA //////////////////////////////////////

```

if(i==4){//CONTROL DE VARIABLE
lcd.clear();
lcd.setCursor(3, 0);//POSICIÓN DEL CORSOR
lcd.print("INCORRECTO");
delay(1000); //PAUSA
i=0;
}

```

PASO 4:

//////////////////////////////////// USUARIO //////////////////////////////////////

```

if(customKey=='2'){ //CONTROL DE LA VARIABLE
s=2;                  //CONTADOR
lcd.clear();          //LIMPIA PANTALLA
}

```

PASO 5:

//////////////////////////////////// CONTROL DEL FRENADO //////////////////////////////////////

```

if((d2==40) &&(vel==40)){ // CONTROL DE LA VARIABLE
digitalWrite(11, 1);      // ACTIVACIÓN BLOQUEO ACELERADOR
digitalWrite(10, 1);     // ACTIVACIÓN ALARMA
digitalWrite(13, 1);    // ACTIVACIÓN FRENO
}
if((d2==35) &&(vel==40)){ //CONTROL DE LA VARIABLE
digitalWrite(11, 1);    // DESACTIVACIÓN BLOQUEO
                        // ACELERADOR
digitalWrite(10, 0);   //DESACTIVACIÓN ALARMA
digitalWrite(13, 0);   //DESACTIVACIÓN FRENO
}
if((d2==90) &&(vel==90)){ //CONTROL DE LA VARIABLE
digitalWrite(11, 1);   // ACTIVACIÓN BLOQUEO ACELERADOR
digitalWrite(10, 1);   //ACTIVACIÓN ALARMA
digitalWrite(13, 1);   //ACTIVACIÓN FRENO
}
if((d2==85) &&(vel==90)){ //CONTROL DE LA VARIABLE
digitalWrite(11, 1);   // DESCTIVACIÓN BLOQUEO

```

PASO 6:

//////////////////////////////////// CUERPO //////////////////////////////////////

```

void loop () ///SENTENCIA INICIO
{
s=0;      //INICIALIZA EL CONTADOR

```

```

d2=0;    //INICIALIZA EL CONTADOR

if((d2==40) &&(vel==40)){ // CONTROL DE LA VARIABLE
digitalWrite(13, 1);//////////BLOQUEO DEL ACELERADOR
digitalWrite(10, 1);//////////BLOQUEO DEL ACELERADOR
}

if((d2==35) &&(vel==40)){ // // CONTROL DE LA VARIABLE
digitalWrite(10, 0);//////////FRENO DE MAQUINA
digitalWrite(13, 0);//////////BLOQUEO DEL ACELERADOR
}

if((d2==90) &&(vel==90)){ // CONTROL DE LA VARIABLE
digitalWrite(13, 1);//////////BLOQUEO DEL ACELERADOR
digitalWrite(10, 1);//////////FRENO DE MÁQUINA
}

if((d2==85) &&(vel==90)){ // CONTROL DE LA VARIABLE
digitalWrite(13, 0);//////////BLOQUEO DEL ACELERADOR
digitalWrite(10, 0);//////////FRENO DE MÁQUINA
}

lcd.setCursor(0, 0);    //POSICIÓN DEL CURSOR
lcd.print("V:");        //ESCRIBA LA VARIABLE

lcd.setCursor(2, 0);
lcd.print(d2);

lcd.print("Km/h");

lcd.setCursor(10, 0);

lcd.print("Max:");

lcd.print(vel);    //ESCRIBE LA VARIABLE VELOCIDAD_MAX

if(vel==40){    //CONTROL DE LA VARIABLE VELOCIDAD_MAX

```

```

lcd.setCursor(5, 1);
lcd.print("URBANO");
}
if(vel==90){ //CONTROL DE LA VARIABLE VELOCIDAD_MAX
lcd.setCursor(3, 1);
lcd.print("PERIMETRAL");
}
char customKey = customKeypad.getKey();//LEE TECLADO
if (customKey != NO_KEY){//CONTROL DEL TECLADO
if(customKey=='D'){ //CONTROL DE LA VARIABLE
lcd.clear();
while (s!=2){//BUCLE DE SENTENCIA
lcd.setCursor(0, 0); //POSICION DEL CURSOR
lcd.print("1._ADMINISTRADOR");//ESCRIBA EN PANTALLA
lcd.setCursor(0, 1);//POSICIÓN DEL CURSOR
lcd.print("2._USUARIO");
char customKey = customKeypad.getKey();//LEE TECLADO
if (customKey != NO_KEY){//CONTROL TECLADO
if((customKey=='D')||(d2==0)){ //CONTROL DE
CONDICIÓN_VARIABLE
s=2; //CONTADOR
lcd.clear();//LIMPIAR PANTALLA
}
if(customKey=='1'){ //CONTROL_VARIABLE
s=0; //INICIALIZA CONTADOR
lcd.clear();
while (s!=2){//BUCLE DE SENTENCIA

```

```

lcd.setCursor(0, 0); //POSICIÓN DEL CURSOR
lcd.print("INGRESE_CLAVE"); //ESCRIBA EN PANTALLA

char customKey = customKeypad.getKey(); //LEER TECLADO

if (customKey != NO_KEY) { //CONTROL TECLADO

lcd.setCursor(i, 1); //POSICIÓN DEL CURSOR

lcd.print("*"); //ESCRIBE EN PANTALLA

i=i+1; //SUMA CONTADOR

if(customKey=='D') { //CONTROL DE LA VARIABLE

i=0; //INICIALIZA CONTADOR

s=2; //INICIALIZA CONTADOR

lcd.clear();

}

if(customKey=='2') { //CONTROL DE CLAVE

c=1; //CONTADOR

}

if(customKey=='4') { //CONTROL DE CLAVE

if(c==1){

c=2; //CONTADOR

}

}

if(customKey=='5') { //CONTROL DE CLAVE

if(c==2){

c=3; //CONTADOR

}

}

if(customKey=='1') { //CONTROL DE CLAVE

if(c==3){ //CONTROL DE LA VARIABLE CONTADOR

c=4; //CONTADOR

```



```

i=0;           //ENCERA CONTADOR
}
}
if(c==4){     //CONTROL DE LA VARIABLE CONTADOR
lcd.clear();  //LIMPIAR PANTALLA
lcd.setCursor(4, 0); //POSICIÓN DEL CURSOR
lcd.print("CORRECTO");
c=0;         //INICIALIZA CONTADOR
delay(1500); //PAUSA
lcd.clear(); //LIMPIAR PANTALLA
while (s!=2){ //BUCLE DE SENTENCIA
lcd.setCursor(0, 0); //POSICIÓN DEL CURSOR
lcd.print("1._URBANO");
lcd.setCursor(0, 1); //POSICIÓN DEL CURSOR
lcd.print("2._PERIMETRAL");
char customKey = customKeypad.getKey();//LEER TECLADO
if (customKey != NO_KEY){//CONTROL TECLADO
if(customKey=='1'){ //CONTROL DE LA VARIABLE
lcd.clear(); //LIMPIAR PANTALLA
lcd.setCursor(5, 0); //POSICIÓN DEL CURSOR
lcd.print("URBANO"); //ESCRIBA EN PANTALLA
lcd.setCursor(0, 1); //POSICIÓN DEL CURSOR
lcd.print("40 Km/h"); //ESCRIBA EN PANTALLA
delay(2000); //PAUSA
s=2; //CONTADOR
vel=40; //VARIABLE_VELOCIDAD_MAX

```

```

lcd.clear();          //LIMPIAR PANTALLA
}
}

if(customKey=='2'){  //CONTROL DE LA VARIABLE
lcd.clear();        //LIMPIAR PANTALLA
lcd.setCursor(3, 0); //POSICIÓN DEL CURSOR
lcd.print("PERIMETRAL");//ESCRIBA EN PANTALLA
lcd.setCursor(0, 1); //POSICIÓN DEL CURSOR
lcd.print("90 Km/h"); //ESCRIBA EN PANTALLA
delay(2000);        //PAUSA
s=2;                //CONTADOR
vel=90; VARIABLE_VELOCIDAD_MAX
lcd.clear();        //LIMPIAR PANTALLA
} } }

if(i==4){           //CONTROL DE VARIABLE
lcd.clear();        //LIMPIAR PANTALLA
lcd.setCursor(3, 0); //POSICIÓN DEL CURSOR
lcd.print("INCORRECTO");//ESCRIBA EN PANTALLA
delay(1000);        //PAUSA
i=0;                //INICIALIZA CONTADOR
if(customKey=='2'){ //CONTROL DE VARIABLE
s=2;//CONTADOR

lcd.clear();        //LIMPIAR PANTALLA

} }

if(customKey=='2'){ //CONTROL DE VARIABLE

```

```
s=2;//CONTADOR
```

PASO 7:

```
////////// INICIALIZACION DE LOS PUERTOS DE ENTRADAS Y SALIDAS //////////
```

```
void setup ()//SENTENCIA QUE SE CUMPLE UNA SOLA VEZ
{
  Serial.begin(9600);//FRECUENCIA DE TRABAJO
  pinMode (pin, INPUT); //HABILITA ENTRADAS
  pinMode(13, OUTPUT);//HABILITA SALIDAS
  pinMode(11, OUTPUT); //HABILITA SALIDAS
  pinMode(10, OUTPUT); //HABILITA SALIDAS
  lcd.init(); //INICIALIZA TECLADO
  lcd.backlight();//DECLARACIÓN TECLADO
  vel=40;//INICIALIZA VARIABLE_VELOCIDAD_MAX
}
```

PASO 8:

```
//////////////////////////////////// MENÚ //////////////////////////////////////
```

```
char customKey = customKeypad.getKey();//LEE TECLDO
if (customKey != NO_KEY){//CONTROL DE TECLADO
if(customKey=='D'){ //CONTROL DE LA VARIABLE
  lcd.clear();//LIMPIA PANTALLA
  while (s!=2){//BUCLE DE SENTENCIA
  lcd.setCursor(0, 0);//POSICIÓN DEL CURSOR
```

CAPITULO IV

4. PRUEBAS Y RESULTADOS

Como resultado de la investigación se obtuvo el diseño e implementación de un sistema de control de velocidad para un bus de la FIE. El cual cuenta con un manual de prácticas y guía de usuario donde se detallan los pasos y procedimientos que hay que seguir para poner en operación de una manera segura y correcta el sistema de control de velocidad, por lo que se debe tener una copia de este documento para la correcta operación del módulo de proceso, ya que además contiene las especificaciones técnicas de cada dispositivo utilizado en el sistema de control.

Una vez diseñado el sistema e instalado en bus de la Facultad de Informática y Electrónica se procedió a verificar que los datos entregados por el sensor de velocidad

que sean reales, mediante un tacómetro colocado en la llanta trasera. Se estableció una relación velocidad vs frecuencia mediante la Tabla.IV.38 se realizó una relación entre el tren de Bit y la velocidad. Estableciendo de manera significativa el tiempo real de recorrido. Demostrando que es factible la implementación del sistema automatizando el sistema de frenado. Tomando la señal en paralelo al sensor de velocidad tipo Vss.



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)
Figura IV. 38: Bus de la Fie

Cuando se sobre pase del límite de circulación permitida, se activará los sistemas de alerta en un tiempo de 2 Segundos. La velocidad se reduce en menos 5 Km/h al límite máximo establecido en un tiempo de 5 a 8 Segundos. Se puede circular en zona urbana a un límite de 0 km/h hasta los 38 km/h. Zona Perimetral de 0km/h hasta los 88 km/h, si se excede los límites el sistema se activara inmediatamente.

Mediante un análisis exhaustivo se llegó a la conclusión que es más seguro y confiable activar el frenado de aire que el frenado de tambor para controlar la velocidad al límite permitido.

ACOPLAMIENTO DE LA SEÑAL DEL SENSOR VSS

RELACION TREN DE PULSOS DEL SENSOR CON VELOCIDAD			
RANGOS TREN DE PULSOS		VELOCIDAD	
1767 - 1835	Bit	5	Km/h
110 - 1200	Bit	10	Km/h
800 - 850	Bit	15	Km/h
615 - 620	Bit	20	Km/h
488 - 490	Bit	25	Km/h
418 - 420	Bit	30	Km/h
355 - 360	Bit	35	Km/h
310 - 315	Bit	40	Km/h
275 - 280	Bit	45	Km/h
245 - 250	Bit	50	Km/h
220 - 225	Bit	55	Km/h
210 - 215	Bit	60	Km/h
195 - 200	Bit	65	Km/h
175 - 180	Bit	70	Km/h
173 - 174	Bit	75	Km/h
156 - 161	Bit	80	Km/h
150 - 155	Bit	85	Km/h
140 -145	Bit	90	Km/h

FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Tabla IV. 3: Tiempos de desarrollo de las prácticas

Es importante que las variables de control en el sistema diseñado coincidan con las variables originales aplicadas al freno de máquina del bus.

4.1. MANUAL TÉCNICO

4.1.1. Especificaciones Técnicas de manejo de la Tarjeta Arduino Uno:

El pin de 5V puede suministrar menos de cinco voltios y la junta puede ser inestable. Si se utiliza más de 12Vdc, el regulador de voltaje se puede sobrecalentar y dañar la placa. El rango recomendado es de 7 a 12Vdc.

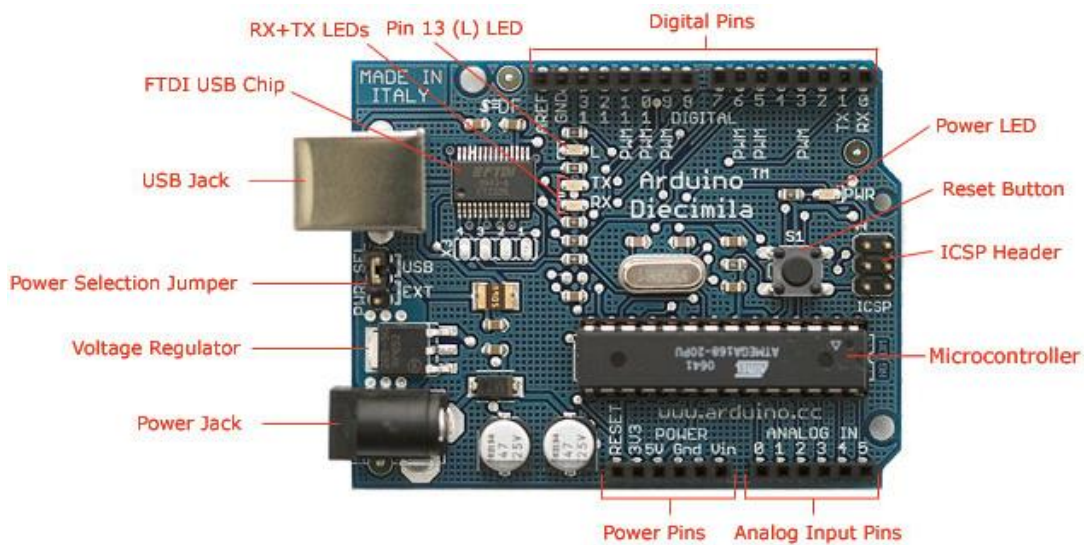
Se recomienda tener en cuenta los siguientes parámetros:

Voltaje de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límites)	6-20V
Pines E / S digitales	14 (de los cuales 6 proporcionan PWM)
Pines de entrada analógica	6
DC Corriente por I / O Pin	40 mA
Corriente CC para Pin 3.3V	50 mA
Memoria Flash	32 KB (ATmega328) de los cuales 0,5 KB utilizado por gestor de arranque
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Velocidad del reloj	16 MHz

FUENTE: <http://arduino.cc/en/Main/arduinoBoardUno>

Tabla IV. 4: Parametros de Funcionamiento de la Tarjeta Arduino

4.1.2. Pines de alimentación de la Tarjeta Arduino Uno



FUENTE: <http://arduino.cc/en/Main/arduinoBoardUno>

Figura IV. 39: Pines de alimentación de la Tarjeta Arduino Uno

- El pin 5v de la tarjeta debe estar regulada desde el tablero. Se puede utilizar el conector USB.
- El suministro de tensión se lo puede realizar a través de los Pines 5V o 3.3Vdc.
- El pin 3.3V específicamente de estar alimentado con 3,3 voltios para evitar daños en la tarjeta. El amperaje máximo es de 50 mA.
- GND. patillas de tierra.
- IOREF. Este pin de la placa Arduino proporciona la referencia de tensión con la que opera el microcontrolador. Un escudo configurado puede leer el voltaje pin IOREF y seleccione la fuente de alimentación adecuada o habilitar traductores voltaje en las salidas para trabajar con los 5V o 3.3V.

Cada uno de los 14 pines digitales en la tarjeta Arduino Uno se puede utilizar como una entrada o salida, funcionan a 5 voltios.

Cada pin puede proporcionar o recibir un máximo de 40 mA y tiene una resistencia pull-up interna (desconectada por defecto) de 20 a 50 kΩ. Además, algunos pines tienen funciones especializadas:

- Serie: Se utiliza para recibir (R_X) y transmitir (T_X) datos serie TTL $R_x = 0$ y $T_x = 1$.
- PWM: 3, 5, 6, 9, 10, 11 y proporcionar una salida de PWM de 8 bits
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Estos pasadores de soporte de la comunicación SPI utilizando la biblioteca de SPI .

4.2. MANUAL DE USUARIO

PASO 1: CONTROL DEL SISTEMA

- Contiene un teclado matricial y un display lcd



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura IV. 40: Reconocimiento de dispositivos

PASO 2: VISUALIZACIÓN

- Contiene un teclado matricial y un display lcd



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura IV. 41: Ingreso del usuario

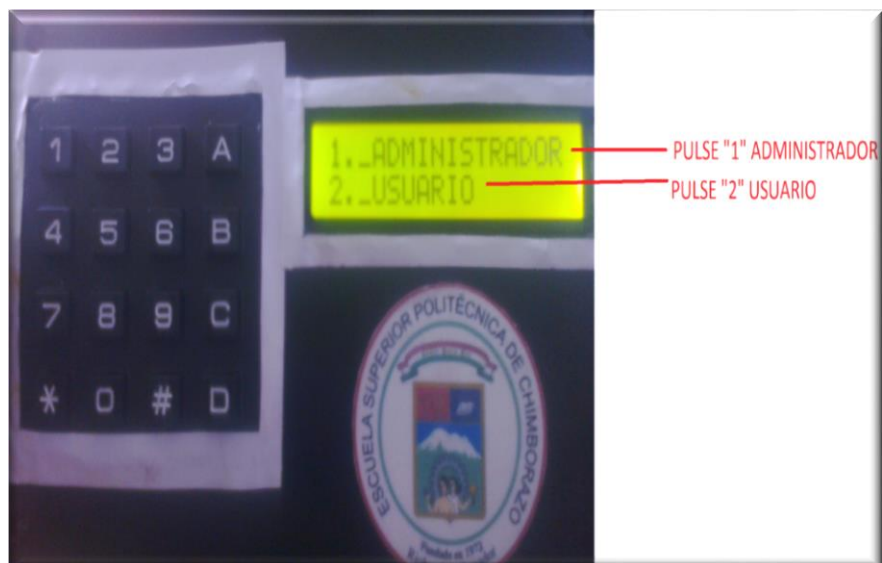
- Se debe presionar la tecla de para ingresar al menú del sistema.



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)
Figura IV. 42: Ingreso al menú de Usuario

PASO 3: MENU

- En el menú del sistema tenemos 2 opciones; Administrador y Usuario



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)
Figura IV. 43: Selección del tipo de usuario

PASO 4: ADMINISTRADOR

- Ingrese la clave correcta del administrador



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura IV. 44: Ingreso de clave

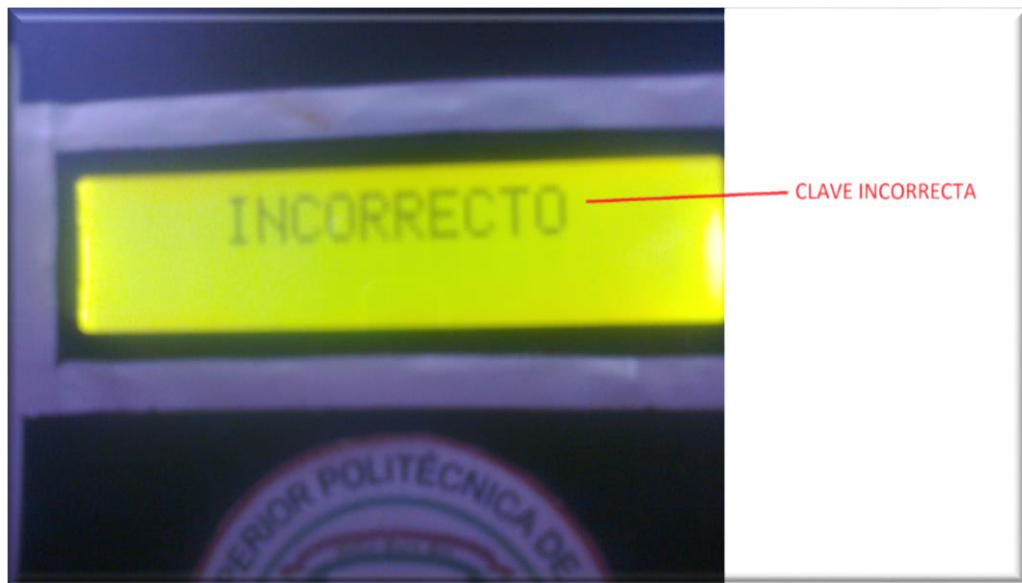
- La clave del Administrador es “ 2451”



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura IV. 45: Digitación de la clave de acceso

- Mensaje en caso de ingresar la clave errónea



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura IV. 46: Visualización de clave incorrecta

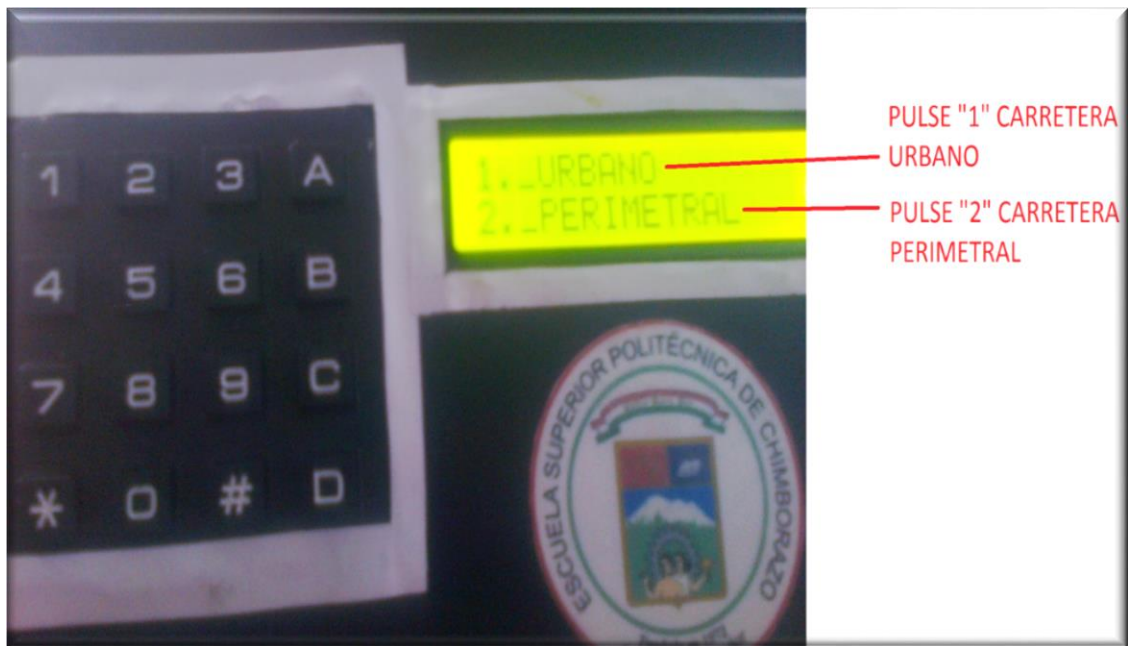
- Mensaje de ingreso clave correcta



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura IV. 47: Visualización de clave correcta

- Selección el tipo de carretera a circular



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura IV. 48: Selección del tipo de recorrido del vehículo

- Tipo de carretera Urbana 40 km/h



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)

Figura IV. 49: Selección de zona Urbana, Límite máximo permitido

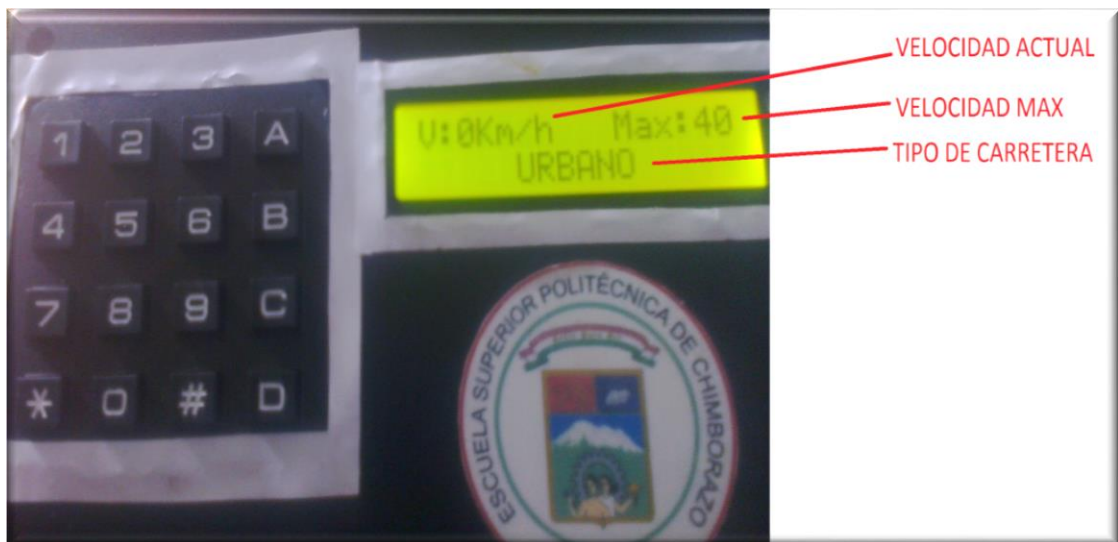
- Tipo de carretera perimetral 90 km/h



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)
Figura IV. 50: Selección de zona Perimetral, Límite máximo permitido

PASO 5: USUARIO

- Selección de opciones y dispositivo configurado



FUENTE: Darwin R. Pomaquero S., Roberto C. Oñate L. (Autores)
Figura IV. 51: Descripción del entorno de desarrollo

CONCLUSIONES

- ✓ Con la implementación del sistema se controla de forma eficiente, segura y confiable, la velocidad óptima del bus en zonas urbana y perimetral.
- ✓ La elección del sistema a utilizar se realizó previo un análisis de costos y soporte técnico.
- ✓ El sistema está diseñado con una presentación amigable e intuitiva para que cualquier persona pueda entender con facilidad.
- ✓ Como indicador de seguridad se colocó un sistema audible una chicharra y en paralelo un sistema visible con una luz intermitente multicolor (rojo-azul).
- ✓ Se reforzó la seguridad de conducir mediante la incorporación de un sistema de frenado automático al exceder los límites de circulación vehicular indicado.
- ✓ El sistema es administrado por una central de operaciones, acorde al lugar de circulación vehicular, mediante la selección por teclado del tipo de circulación.
- ✓ El sistema es reprogramable y actualizado en los parámetros del límite de velocidad, dependiendo de la reglamentación de circulación vehicular vigente.
- ✓ Se obtuvo el voltaje necesario de la señal de entrada mediante un divisor de voltaje.
- ✓ Se logró la adaptación en el bus del sistema neumático para el bloqueo del acelerador.
- ✓ Se observó el correcto funcionamiento del bus con el sistema de control de velocidad instalada.

RECOMENDACIONES

- ✓ Se debe seleccionar una adecuada tarjeta de adquisición de datos para obtener los datos en tiempo real y que sea amigable con nuestro lenguaje de programación seleccionado.
- ✓ Debemos analizar qué tipo de frenado se adapte en un vehículo, esto dependerá del modelo y sistema de combustión.
- ✓ Se debe promover e incentivar el uso de este sistema para evitar sanciones por exceso de velocidad y lo más importante salvaguardar la vida nuestra y la de los demás.
- ✓ Leer el manual de usuario para entender el sistema, en especial si se actúa como Administrador del sistema.
- ✓ Se debe realizar una tabla de relación de velocidad exacta, para que no exista desfases con el sistema original.
- ✓ Se recomienda trabajar a una presión de 65 PSI para que el sistema de bloqueo del acelerador funcione y el cilindro neumático pueda actuar.
- ✓ Se recomienda colocar disipador de calor para la fuente de energía.
- ✓ No exceder en la manipulación de los racores (Conector Neumático), tienden a dañarse y por ende generan pérdidas de presión.
- ✓ Se recomienda utilizar una fuente de energía alterna para evitar el apagado inesperado e inapropiado del sistema.
- ✓ En caso de falla se recomienda apagar el sistema mediante un botón de emergencia, el cual desactiva todo el sistema.

RESUMEN

Se diseñó e implementó un sistema de control de velocidad para el bus de la Facultad de Informática y Electrónica de la Escuela Superior Politécnica de Chimborazo.

Mediante el método deductivo se pudo partir de un análisis general de sistemas de automatización vehicular y se llegó a un modelo particular que puede ser implementado en cualquier automotor a diesel, utilizando una tarjeta de adquisición de datos, un sensor de velocidad, circuito de potencia, cilindro eléctrico, computador entre otros elementos electrónicos. Basándonos en reglamentos de tránsito de circulación vehicular programaremos nuestra tarjeta para el control de velocidad para una conducción segura y confiable. El control de velocidad se lo ha diseñado basándonos en sistemas ya probados pero con un costo elevado. Se mejorará el diseño bajando los costos para la implementación, siendo accesible en el mercado nacional. El administrador vehicular seleccionara el tipo de recorrido que realice ya sea urbano o perimetral, el sistema permitirá obtener resultados en el momento que el conductor intente sobrepasar el nivel de velocidad permitido por la Ley de tránsito, es decir, el sistema evitara las sanciones por exceso de velocidad y además permitirá un viaje tranquilo y seguro, evitando así en un 80% los accidentes de tránsito ocasionados por el exceso de velocidad.

Podemos concluir que con la implementación de dicho sistema es factible controlar de forma eficiente, segura y confiable la circulación vehicular, utilizando una adecuada tarjeta de adquisición y una programación estructurada ideal en lenguaje de programación C++. Se recomienda realizar un sistema independiente para el sistema de frenado diferente de fabricación existente para los diferentes vehículos, así el sistema funcionara correctamente y no existirá interferencia con el sistema original.

SUMMARY

It is designed and implemented a speed control system for the Faculty of Informatics and Electronics at The Escuela Superior Politécnica de Chimborazo (Higher education).

Through the deductive method from a general analysis of vehicle automation systems and is reached to a particular model that can be implemented in any diésel automotive, using a data collection card, a sensor speed, power circuit, electric cylinder, computer among other electronic elements. Based in transit regulations of vehicular movement the card will be programmed for the speed control for a secure and reliable driving.

The speed control has been designed based in proven systems but with a high cost. The design will be improved lowering costs for the implementation, being accesible in the national market.

The vehicular administrator will select the type of path it is realized whether it is urban and rural, the system will allow to get results in the momento that the driver try to exceed the speed level allowed by the transit law, it means, the system will avoid the sanctions for exceed the speed besides will allow a confortable and secure ride, avoiding the transit accidents in 80 % caused by speeding.

It is conclude the implementation makes the system feasible to control the vehicular movement in an efficient way, secure and reliable, using an adequate card of acquisition and an ideal structured programming with C++ programming language.

It is recommended to carry out an independent system for the different breaking system of existing manufacturing for different vehicles, thus the system will work correctly and there will not be interference with the original system.

BIBLIOGRAFÍA

- 1. BOLTON, W.,** Mecatrónica: Sistemas de Control Electrónico en la Ingeniería Mecánica y Eléctrica., 4^a.ed., México DF-México., ALFAOMEGA., 2010., Pp. 65-80
- 2. CREUS, A.,** Instrumentación Industrial., 8^a.ed., Madrid-España., McGrill., 2011., Pp. 30-68
- 3. KATSUIKO OGATA.,** Sistemas de control Electrónico., 2^a.ed., México DF-México ., Prentice Hall Hispanoamérica SA., 1996., Pp. 150-310
- 4. MULLER, R.,** Pneumatics: Theory and Applications., 1^a.ed., Berlín-Germany., DITZINGEN., 1998., Pp. 20-44
- 5. REYES, C.,** Microcontroladores: Programación en Basic., 3^a.ed., Quito-Ecuador., RISPERGRAF., 2008., Pp. 80-154

BIBLIOGRAFÍA DE INTERNET

1. ACCESORIOS NEUMÁTICOS

<http://www.taringa.net/posts/apuntes-y-monografias/5426137/>

2013/02/10

<http://www.sicontrol.com/racores.htm>

2013/02/28

2. CILINDRO NEUMÁTICO

<http://www.microautomacion.com/catalogo/Actuadores.pdf>

2013/03/01

3. MANUAL PARA PROGRAMAR E INSTALAR EL CONTROL DE VELOCIDAD

<http://www.microediacar.com/UserFiles/File/MANUAL%20DE%20INSTALACION%20CONTROL%20DE%20VELOCIDAD.>

2013/04/17

4. MICROCONTROLADORES

http://web.ing.puc.cl/~mtorrest/downloads/pic/tutorial_pic.pdf

2013/05/11

5. MODULOS ARDUINO

<http://arduino.cc/es/Reference/Modulo>

2013/05/22

6. RELÉ

<http://cpc.farnell.com/1/1/3687-relay-pcb-16a-spc0-24vdc-4061>

2013/06/07

7. SENSOR DE VELOCIDAD

http://e-auto.com.mx/manual_detalle.php?manual_id=223

2013/06/11

8. SISTEMAS NEUMATICOS

<http://cursos.aiu.edu/PDF/Tema%202.pdf>

2013/09/08

9. TIPOS DE SEÑALES

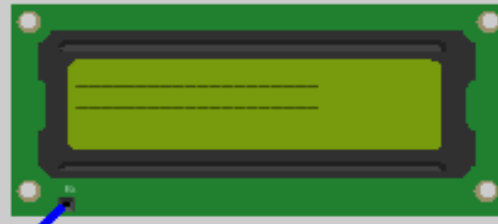
http://es.wikipedia.org/wiki/Modulaci%C3%B3n_por_ancho_de_pulsos

2013/10/19

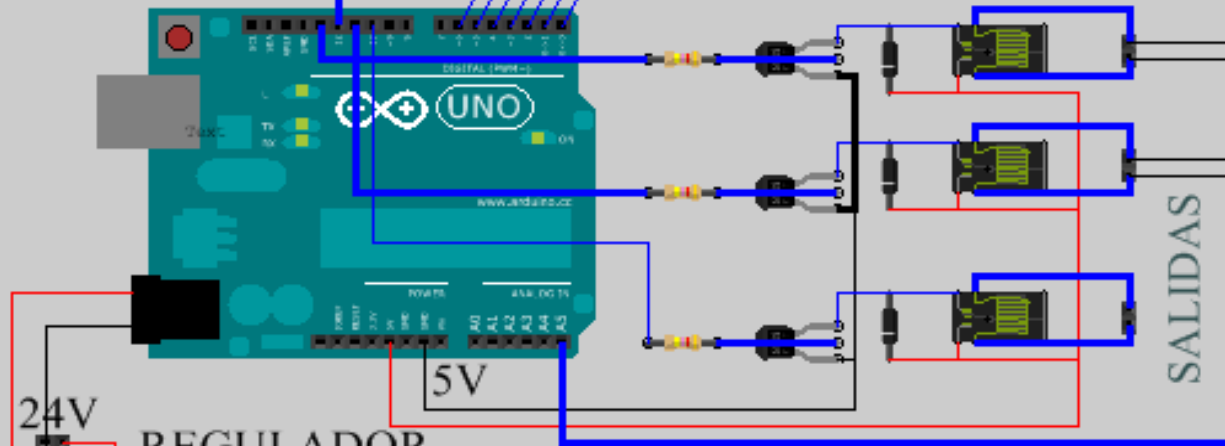
ANEXO 1

Diagrama de funcionamiento del sistema de control de velocidad

SEÑAL DE ENTRADA



ETAPA DE POTENCIA



24V

REGULADOR

5V

12V

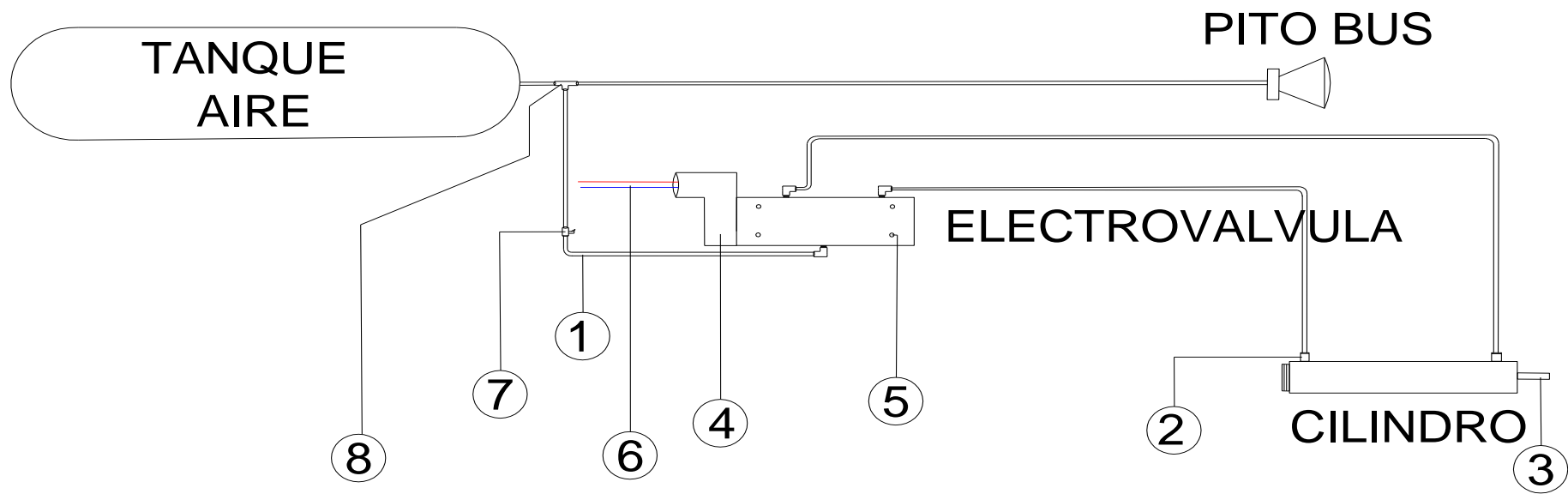
SALIDAS

- NEGATIVO
- POSITIVO
- SEÑAL



ANEXO 2

Plano de conexiones del sistema neumático



① MANGERA NEUMÁTICA

⑤ ESCAPE NEUMÁTICO

② RECORES NEUMÁTICO

⑥ ALIMENTACIÓN DE LA BOBINA

③ VASTAGO

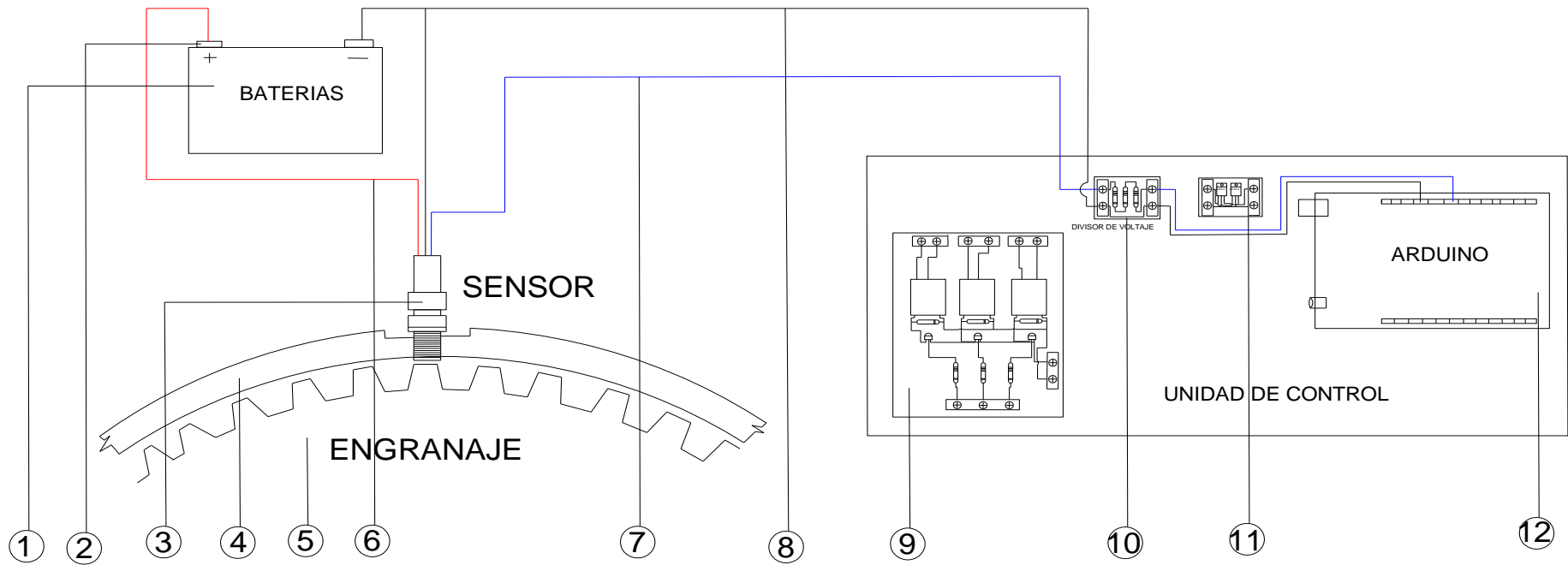
⑦ LLAVES DE PASO NEUMÁTICO

④ BOBINA

⑧ UNIÓN T NEUMÁTICO

ANEXO 3

Plano de conexiones del sistema de entrada del sensor



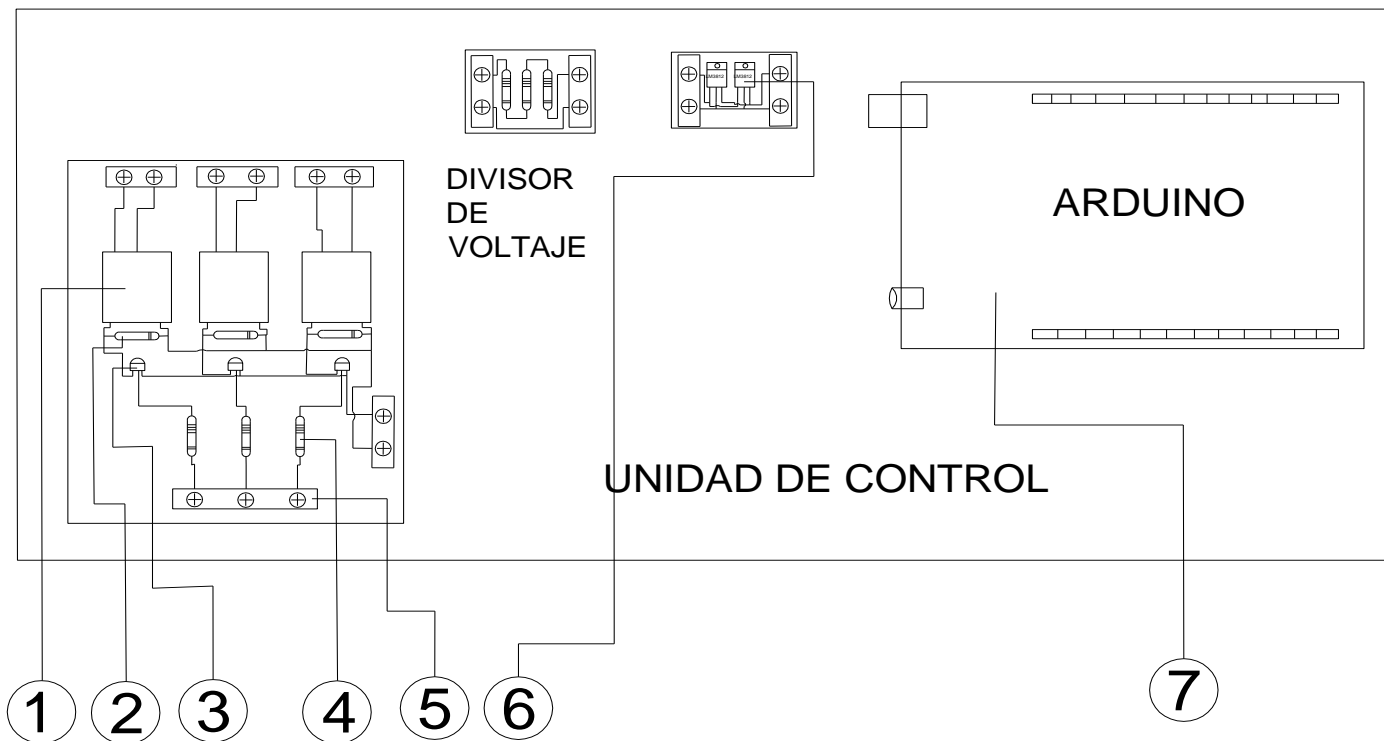
- | | |
|------------------------------|-----------------------------|
| ① BATERÍA DEL BUS | ⑦ SEÑAL DEL SENSOR |
| ② CONECTOR DE BATERÍAS | ⑧ ALIMENTACIÓN NEGATIVO |
| ③ SENSOR MAGNÉTICO VSS | ⑨ ETAPA DE POTENCIA |
| ④ CAJA DE TRANSMISIÓN | ⑩ ETAPA DIVISOR DE VOLTAJE |
| ⑤ ENGRANAJE | ⑪ ETAPA REDUCTOR DE VOLTAJE |
| ⑥ ALIMENTACIÓN POSITIVO(24V) | ⑫ ETAPA ARDUINO |

ANEXO 4

Plano de conexiones del sistema de frenado

ANEXO 5

Plano de la unidad de control



① RELAY 5V

② DIODO RECTIFICADOR

③ TRANSISTOR 2N3904

④ RESISTENCIA 4.7K

⑤ BORNERAS

⑥ RECTIFICADOR DE VOLTAJE

⑦ MODULO ARDUINO

ANEXO 6

Plano de las conexiones del sistema de control de velocidad

