

**UNIVERSIDAD MICHUACANA
DE SAN NICOLAS DE
HIDALGO**



FACULTAD DE INGENIERIA ELECTRICA

Tesis

**"Automatización de un sistema de cambios de
velocidades para una bicicleta utilizando la
plataforma Arduino"**

**Que para obtener el grado de:
Ingeniero en Electrónica**

Presenta:

German Esteban Tapia

Asesor:

Doctor en ingeniería eléctrica Roberto Tapia Sánchez

Agradecimientos

Le doy gracias a dios por brindarme los medios y las herramientas necesarias para llegar a este momento, por darme la fuerza para continuar en aquellos momentos de flaqueza, por darme la dicha existir y rodearme de seres maravillosos que fueron parte de este logro.

Le doy las gracias a mi asesor el Dr. Roberto Tapia Sánchez por su apoyo total de principio a fin en este proyecto de tesis, gracias por confiar en mí, por preocuparse desinteresadamente de este proyecto, gracias por su infinita paciencia pero sobre todo ¡Gracias por ser un buen amigo!

Gracias a mis padres Rosa Tapia y Martin Esteban y a mi esposa Blanca Esthela Fuentes por creer en mí, porque siempre me hicieron creer que era un persona capaz, gracias por su apoyo incondicional, por no dudar que este que escribe alcanzaría la meta.

Gracias a mi tío Jaime Tapia por enseñarme el camino a seguir y por apoyarme desde el principio de mi formación.

Gracias a mi facultad por hacer de mí una mejor persona, a todos aquellos maestros y compañeros con los que compartí aula y que me hicieron crecer y madurar con su buen ejemplo y perseverancia.

Dedicatoria

Dedico esta tesis a mi madre Rosa Tapia López por su cariño y comprensión, por ser una persona entregada a su hijo, me llenaste de buenos pensamientos y me alejaste de las malas costumbres, me diste todo cuanto pudiste sin dudar de mí. Desde que tengo memoria mi educación fue tu prioridad, hiciste lo imposible para que nada me faltara, me motivaste para seguir adelante, me diste todo cuanto pudiste sin dudar de mí. Sé que nada te hace más feliz que verme cumplir mis metas y por eso te dedico este y todos mis logros.

¡Dios te bendiga Madre!

Resumen

En el presente proyecto de tesis, se presenta la automatización de un sistema de cambio de velocidades de una bicicleta utilizando un microcontrolador y un interruptor magnetico.

En la realización del presente proyecto se consideran tres etapas principales, las cuales son descritas a continuación:

Primera etapa: Se realiza una selección de los componentes (Hardware) necesarios para el desarrollo e implementación del prototipo propuesto.

Segunda etapa: Planeación y diseño del algoritmo que es implementado en el microcontrolador.

Tercera etapa: Pruebas e implementación del prototipo propuesto.

Palabras claves: Sistema de cambios, automatización, microcontrolador.

Abstract

This thesis project presents an automatic bicycle gear shift system implemented by using a microcontroller and a magnetic switch.

For the implementation are considered three main stages, which are described below:

First stage: The Hardware elements are selected in order to the development and implementation of the proposed prototype.

Phase Two: The algorithm developed and designed algorithms are implemented in the microcontroller.

Third stage: Implementation and tests of the proposed prototype.

Keywords: Gear shift system, automation, microcontroller.

Contenido

Agradecimientos	I
Dedicatoria	III
Resumen	V
Abstract	VII
Capítulo 1. Introducción	1
1.1 Introducción	1
1.2 Estado del Arte	3
1.3 Breve historia de la bicicleta	4
1.4 Justificación	7
1.5 Descripción del sistema	8
1.5.1 Bicicleta	9
1.5.2 Sistema de cambios	10
1.5.3 Funcionamiento correcto del sistema de cambios de velocidades	12
1.6 Objetivo	15
1.7 Organización de la tesis.....	16
Capítulo 2. Implementación del algoritmo	18
2.1 Planeación.....	18
2.2 Criterios para el diseño del sketch:	22
2.2.1 Calculo de la Velocidad	23
2.2.2 Pulgadas de Engranaje	25
2.2.3 Cuantificación	28
2.3 Estructura del Sketch.....	31
2.4 Diagramas de flujo.....	33
2.4.1 Diagrama de flujo “Control del sistema”	34
2.4.2 Diagrama de flujo de la función “Velocidad”	35
2.4.3 Diagrama de flujo de las funciones “cadencia1” y “cadencia2”	36
2.4.4 Diagrama de flujo de las funciones “subirMarcha” y “bajarMarcha”	38
2.5 Presentación del algoritmo	40
2.5.1 Justificación del código.....	40
2.5.2 Código	42
Capítulo 3. Diseño del circuito general.....	51
3.1 Justificación del Circuito	51
3.2 Capacidad de la batería.....	52

3.3	Descripción de los elementos.....	54
3.3.1	Servomotor.....	54
3.3.2	Interruptor Magnético	57
3.3.3	Interruptor de cadencia	58
3.3.4	Placa Arduino Leonardo	59
3.3.5	Dínamo	60
3.3.6	Regulador de voltaje	61
3.3.7	Círculo General	62
Capítulo 4.	Implementación y pruebas.....	65
4.1	Introducción.....	65
4.2	Prueba 1, Cadencia 1 – velocidad aumento – terreno regular.	66
4.3	Prueba 2, Cadencia 1 – velocidad aumento-disminución – terreno regular.	68
4.4	Prueba 3, Cadencia 2 – velocidad aumento – terreno regular.	70
4.5	Prueba 4, Cadencia 2 – velocidad aumento-disminución – terreno regular.	72
4.6	Prueba 5, Cadencia 1 y 2 – terreno regular.	73
4.7	Prueba 6, Cadencia 1 y 2 – terreno irregular.	75
Capítulo 5.	Conclusiones y trabajos futuros.....	78
5.1	Conclusiones.....	78
5.2	Trabajos futuros	79
Anexo A.....		81
A.1.	Medidas de seguridad Shimano	82
A.2.	Hoja de datos Lm2596t-5.....	83
BIBLIOGRAFIA		84

Capítulo 1. Introducción

En el presente capítulo se aborda el planteamiento del tema de investigación. Son descritos los objetivos del trabajo, así como una breve descripción del estado del arte. La justificación del presente trabajo es descrita y la organización de la tesis es presentada.

1.1 Introducción

La bicicleta es un medio de transporte que ha sido utilizado desde hace varios siglos. Este medio de transporte se ha presentado a lo largo de su historia como una alternativa en cuanto al transporte urbano se refiere. Esto tomando en cuenta que no se tienen grandes distancias a recorrer dentro de los desplazamientos que se realizan en la urbe. A pesar de ser un medio de transporte alternativo para las zonas urbanas, su desarrollo e investigación no ha sido explotado del todo.

La bicicleta posee innumerables beneficios en lo que a tráfico en grandes ciudades se refiere, debido a que es silenciosa, no emite contaminantes y en consecuencia permite disminuir la concentración de contaminantes en las ciudades y, probablemente la más importante, permite la realización de actividad física de las personas, contribuyendo a la disminución de problemas de salud i.e. sobrepeso, colesterol, diabetes etc.

Si bien, la bicicleta es una alternativa interesante a tomar en cuenta como medio de transporte popular, el hecho de que no se tenga una cultura del uso de este medio de transporte, puede radicar en diferentes factores, tales como: el respeto vial o preferencial al ciclista por parte de los conductores de autos o motocicletas, la asignación específica de espacios en las vías para este medio de transporte, o también el hecho de que no se han realizado avances o innovaciones en este medio de transporte.

Debido a lo expuesto anteriormente, el presente proyecto de tesis plantea la utilización de una bicicleta ordinaria que cuente con un sistema de cambios convencional, a partir de la cual se crea un sistema electrónico que permita brindarle al usuario una experiencia más agradable pero al mismo tiempo hacer que éste haga un uso correcto y óptimo de esta herramienta sin tener la necesidad de realizar algún movimiento (en el sistema de cambios), ni la necesidad de contar con conocimientos previos de ciclismo o de su sistema de cambios de velocidades.

El presente proyecto se encuentra enfocado en el uso urbano de la bicicleta, aunque puede ser aplicado de manera profesional, es decir, el sistema puede ser utilizado en competencias profesionales de carreras o para entrenamiento de los competidores. Su uso de manera profesional requeriría de por lo menos una variable más (sensor) a considerar, además de que un incremento de medio kilogramo en el peso total de la bicicleta podría representar una desventaja para el competidor.

Por lo tanto se puede decir que se crea un sistema electromecánico en donde la parte mecánica se encargará de transmitir la energía del ciclista a las ruedas y el sistema electrónico (la parte de mayor interés en este proyecto) se encargará de realizar los cambios de velocidades correctas en el momento justo, considerando que la cadencia de pedaleo se mantenga constante y así la inercia de movimiento de la bicicleta se aproveche de la mejor manera, esto optimiza la transferencia de energía para aprovechar al máximo la fuerza ejercida en el pedaleo.

La optimización de la transferencia de energía se puede ver desde el punto de vista que durante una sesión de ciclismo se tienen varias situaciones en las que la cadencia de pedaleo se vuelve incomoda, es decir que el pedaleo se vuelva pesado o que el pedaleo sea muy ligero pero la velocidad no se vea afectada. Esta situación se hace presente cuando se pasa de un llano a una subida, de un llano a una bajada, de una bajada a una subida o de una subida a una bajada. Debido a estas situaciones, el sistema electrónico permitirá evitar derroches de energía ya sea por el retardo en la ejecución del cambio o por el cambio incorrecto de las marchas.

El presente capítulo se encuentra organizado de la siguiente manera: Primeramente se presenta el estado del arte, la justificación, objetivos y descripción del sistema, para finalmente hacer una breve descripción de los capítulos posteriores.

1.2 Estado del Arte

Los sistemas de cambios automáticos para bicicletas, es un tema en el cual no se han realizado estudios formales y algunos prototipos aplicados son presentados. Se tienen antecedentes de este tipo de sistemas en años recientes (2013), sin embargo la fuente de información no puede ser verificada de manera formal.

En especial, se encontraron antecedentes de 2 sistemas [1], [2]. Para el primer sistema [1], se tiene que unos estudiantes de la universidad de Carnegie Mellon en Pittsburgh, Pensilvania, desarrollaron un sistema de cambios automáticos para un concurso de desarrollo organizado por el departamento de ingeniería eléctrica y computacional. En dicho proyecto, los estudiantes utilizaron sensores de efecto hall e imanes. El objetivo del sistema es el de mantener la cadencia del ciclista en 80 rpm, y hacer el cambio del engrane al estar por debajo de las 60 rpm, llevando la cadena a un engrane más alto. Si la cadencia pasaba las 100 rpm se baja de engrane para regresar a la cadencia preferida. El sistema utiliza un Arduino Mega, y servomotores para realizar el cambio de engranaje.

El sistema ya comercializado que se presenta en [2], tiene un sistema de procesamiento (del cual no se dan detalle), encargado de controlar pequeños motores (probablemente servomotores), y el hecho de contar con una masa trasera del fabricante NuVinci [3] que le permite realizar el cambio internamente. Esto con el objetivo de mantener el confort del ciclista. Además, cuenta con un dínamo delantero, que permite recargar al sistema y alimentar a los motores.

Si bien estos dos proyectos presentan la misma idea, aplicada con algunos componentes diferentes, la información y detalle de ellos es muy ambigua y en ocasiones de difícil acceso. Por otra parte, no se tienen antecedentes precisos que revelen el surgimiento o la fundamentación de este tipo de sistemas pero eso no quiere decir que jamás se había pensado en este proyecto, de hecho ya se han hecho algunos prototipos enfocados en ideas similares (Eléctrico-Mecánico) pero con mayor interés en crear sistemas híbridos.

En las últimas décadas varias empresas de bicicletas, empresas de tecnología y sobre todo automotrices se han empeñado en diseñar las mejores bicicletas, pero en

la mayoría de los casos se trata de bicicletas híbridas, es decir, que cuentan con sistemas que combinan energía mecánica y energía eléctrica para desplazarse. Entre las más destacadas se encuentran los siguientes modelos: Audi con su e-bike, la ipedelec de BMW, Yamaha y su YPJ-01, la Faraday de Ideo y Rock Lobster Cycles, La Ford E-Bike concept, etc. [4] Por otra parte también es interesante mencionar la rueda Copenhagen, la cual es una rueda motor que convierte una bicicleta ordinaria en una bicicleta híbrida eléctrica y se controla mediante una aplicación que se instala en el teléfono celular. Esta rueda reemplaza a la rueda trasera de la bicicleta en la que se usa un sistema de toma decisiones con respecto a la fuerza que ejerce el ciclista en el pedaleo. [5]

¿Y por qué no utilizar esa información como antecedentes? La respuesta es muy simple, los diseños fueron hechos por empresas que tienen la finalidad de competir en la actualidad en el mercado con sus similares y lo último que harían es compartir su diseño y experiencia con el mundo. Por tal motivo expuesto en el párrafo anterior en este subtema se limitará a mencionar los diseños conocidos que tengan relación con este diseño. Es por eso que se presentarán antecedentes del sistema de cambios y la bicicleta que son de gran importancia para el lector antes de sumergirse en el desarrollo del diseño.

1.3 Breve historia de la bicicleta

El antecedente más antiguo del que se tiene registro pertenece a la obra “Codex Atlanticus” de Leonardo Da Vinci (Figura 1.1) donde se bosqueja un diseño de la primera bicicleta y tiene la peculiaridad de usar transmisión de cadena la cual es usada en la bicicleta moderna más no en sus antecesoras contemporáneas.

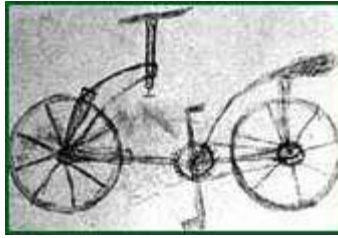


Figura 1.1 Boceto de Leonardo Da Vinci [6]

Después, en 1690 el conde francés Mede de Sivrac inventó el “Celífero” que consistía en un bastidor con ruedas, una almohadilla en el asiento pero sin manillar. El Celífero se propulsaba y dirigía con los pies.

En 1816 un noble alemán agregó un dispositivo de dirección al Celífero y lo llamo máquina Draisiana (Figura 1.2), un año más tarde en Inglaterra lo mejoraron pero en 1819 se patentó la idea en Estados Unidos.



Figura 1.2 Maquina Draisiana [6]

En 1839 el escoces Kirkpatrick Macmillan añadió a la Draisiana palancas de conducción y pedales conectados con barras a la llanta trasera (Figura 1.3).



Figura 1.3 Primera bicicleta con pedales [6]

Ernest Michaux añade en 1861 los pedales a la llanta delantera pero el equilibrio era complicado (Figura 1.4). El cuadro y las ruedas eran de madera mientras que los neumáticos eran de hierro.



Figura 1.4 Velocípedo Machauz [6]

En 1869 se implementan los neumáticos de goma maciza montados sobre el acero, y es aquí cuando se le da el nombre de bicicleta.

Hacia 1873 James Starley, un inventor inglés produjo la bicicleta de rueda alta donde la rueda delantera era tres veces más grande que la trasera (Figura 1.5).



Figura 1.5 Bicicleta de rueda alta [6]

En 1880 surge la máquina segura, las ruedas eran casi del mismo tamaño y los pedales estaban unidos a la rueda trasera mediante una transmisión de cadena (Figura 1.6). Cinco años después John Kemp Starley mejora el diseño, añade frenos y la nombra “Bicicleta de seguridad”. Este diseño es el más parecido al modelo actual.



Figura 1.6 Bicicleta de seguridad [6]

En 1888 John Boyd Dunlop se da cuenta de que es un martirio viajar en bicicleta por calles empedradas y desarrolla los neumáticos con cámara de aire para amortiguar los impactos.

Y finalmente a principios del siglo pasado aparecen los primeros indicios de lo que son hoy los sistemas de transmisión con múltiples piñones y platos, fue cuando las competencias de ciclismo se hicieron muy populares y los competidores buscaban la forma de hacer su bicicleta más rápida, más ligera y con menos fricción en sus partes móviles (sistema de transmisión). Al principio los competidores no confiaron en sistemas de cambios donde la cadena tuviera que saltar de un piñón a otro pues según ellos esto ocasionaba más fricción aparte de que se agregaban más partes móviles al buje trasero, así pues estos diseños fueron utilizados solo en bicicletas convencionales y de turismo mientras que los competidores utilizaron un sistema peculiar el cual consistía en colocar un piñón en cada lado de la rueda trasera (que era fijada con palomas para desmontarla fácilmente), un piñón era diseñado para uso regular mientras que el otro se utilizaba para ascensos pronunciados, así pues cuando el ciclista se encontraba con una subida prolongada bajaba de la bicicleta, desmontaba la rueda trasera, la giraba y montaba de nuevo para seguir su recorrido y al terminar la ascensión repetía el mismo procedimiento para regresar la rueda a su posición más usual. [6], [7], [8].

1.4 Justificación

En la actualidad, los cambios en la tecnología son muy notables y cada vez más grandes, esto se debe en gran parte a la incursión de la electrónica en un gran número de tareas, procesos, y herramientas de la vida cotidiana. Sin embargo hay campos que han sido explorados en menor medida, tal es el caso de algunos sistemas mecánicos como la bicicleta. A este medio de transporte (que es puramente mecánico), se le han agregado herramientas electrónicas como cronómetros, medidor de pulsos cardíacos, lámparas, GPS (por sus siglas en inglés Global Positioning System); sin embargo, ninguno de estos elementos se encuentra enfocado en su sistema mecánico, el cual es la parte principal de la bicicleta.

Es por esta razón, que el presente proyecto se enfoca en la creación de un sistema electro-mecánico que interactúe con el usuario y permita obtener un mejor rendimiento del uso del sistema.

La idea principal surge de la necesidad de tener una certeza en que se están utilizando el sistema de cambios de velocidades en la bicicleta de la mejor manera posible, tanto para el usuario, como también para el propio sistema. Esto debido a que la mayoría de los usuarios no tienen el conocimiento de cuál es la forma correcta de usar los cambios de velocidades de una bicicleta.

Como se explicó en la sección anterior, existen algunos sistemas que realizan esta función, pero resultan costosos y de difícil acceso para la mayoría de las personas, otros como el sistema mostrado en [1], el cual tiene una semejanza muy cercana con el propuesto en el presente trabajo, sin embargo la manera en que es abordado es probable que no sea la mejor, debido a que está basado en mantener el nivel de la cadencia tomando como referencia el dato proporcionado por un club de ciclismo.

Es por eso que surge la motivación en el presente proyecto de desarrollar un sistema que esté al alcance de todas las personas, sin importar su nivel socio-económico, y que al mismo tiempo sea funcional, efectivo y que permita superar en rendimiento a un sistema de cambios de velocidades manual.

1.5 Descripción del sistema

Este sistema estará conformado por varios subsistemas los cuales son: la bicicleta, el sistema de cambios de la bicicleta, el sistema eléctrico y el sistema de procesamiento.

La bicicleta es el sistema en estudio, al cual se le agregará una etapa eléctrica que se encargará de realizar los cambios en el sistema mecánico, esta etapa estará conformada por tres servomotores que actuarán sobre las palancas de velocidades y las guías de cadena. En el sistema de procesamiento se aplicará un control de lazo cerrado, mediante el monitoreo de la velocidad angular de la llanta trasera de la bicicleta, dicho sistema debe ser capaz de ejecutar las acciones necesarias y transmitírselas al sistema eléctrico.

El sistema de procesamiento contará además con un subsistema de alimentación ubicado en el mismo sistema eléctrico con el objetivo de contar con un diseño sustentable.

Se tomará en cuenta el diseño para un sistema de cambios de velocidades que contenga dos platos y nueve piñones, debido a que esta configuración es la más comercial, permitiendo de esta manera un beneficio de mayor penetración desde el punto de vista de comercialización y beneficio para una mayor cantidad de usuarios.

En la siguiente sección se describen los elementos básicos que conforman al sistema estudiado.

1.5.1 Bicicleta

La bicicleta, coloquialmente llamada “bici” [9], es un vehículo de transporte personal de propulsión humana, es decir, por el propio viajero y/o usuario. Los componentes básicos de dicho medio de transporte son:

- i. Dos ruedas, generalmente de igual diámetro y dispuestas en línea.
- ii. Un sistema de transmisión a pedales.
- iii. Un cuadro metálico que le da la estructura e integra los componentes.
- iv. Un manillar para controlar la dirección.

- v. Un sillín para sentarse.

El desplazamiento se obtiene al girar con las piernas la caja de los pedales que se encuentran conectados a un plato, y por medio de una cadena a un piñón que a su vez hace girar la rueda trasera sobre el pavimento.

El diseño y configuración básicos de la bicicleta han cambiado poco desde el primer modelo de transmisión de cadena desarrollado alrededor de 1885.

1.5.2 Sistema de cambios

El sistema de cambios de velocidades de una bicicleta puede estar configurado de diferentes formas (con respecto a la cantidad de platos y piñones así como su número de dientes), esto depende de las características que requiera el usuario. A pesar de esto, dicho sistema, se encuentra conformado con el mismo principio.

Los sistemas de cambios de una bicicleta pueden variar desde 3 ó 5 velocidades hasta 27, pero la relación de cambio (la relación de cambio es la relación que existe entre el número de dientes de los platos y los piñones) entre la velocidad más baja y la más alta de un sistema a otro varía muy poco [10]. Las partes que describen al sistema de cambios de velocidades se pueden observar en el esquema de la Figura 1.7. Dichas partes son descritas a continuación.

Piñones: Es cada una de las ruedas dentadas unidas a la rueda trasera y a través de las cuales se transmite la fuerza a la misma (ver Figura 1.7). Se utilizan para dar mayor o menor tracción a la rueda trasera, según el piñón seleccionado. Hay sistemas con un único piñón, con tres, cinco, seis, siete, ocho y nueve. El número de piñones de una bicicleta depende del tipo de bicicleta y del terreno sobre el que se use. Los piñones están ordenados por tamaño encontrándose el mayor de ellos más cerca del eje axial de la bicicleta, es decir, más cerca de la rueda.

Platos: Los platos son las ruedas dentadas unidas a los pedales y su utilidad es similar a la de los piñones. Un uso adecuado de estos puede hacer que el

aprovechamiento de energía sea muy bueno y por lo tanto habrá un derroche menor de energía para el ciclista.

Las bicicletas pueden tener de uno a tres platos, los platos están ordenados de forma inversa a los piñones, en este caso el plato más pequeño se encuentra en parte interior mientras que el plato más grande se encuentra en la parte exterior del conjunto de platos.

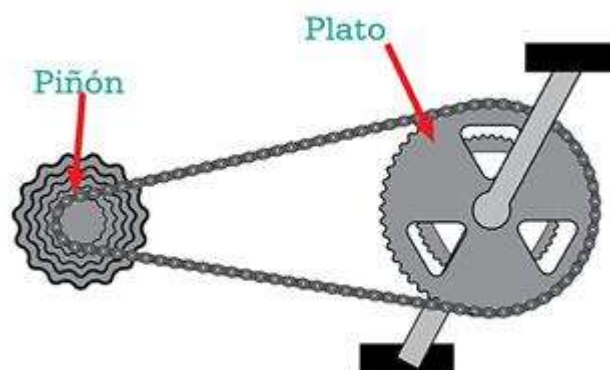


Figura 1.7 Esquema del sistema de cambios de la bicicleta [10]

Cadena: Es la pieza que transmite la fuerza que ejercemos al pedalear desde los platos, hasta los piñones. La cadena se compone de eslabones entrelazados y formados a su vez por dos piezas de metal unidas por dos cojinetes.

Desviadores: Los desviadores o guías de cadena son puentes metálicos que llevan instalado un chicote el cual está conectado en el otro extremo a los mandos de la bicicleta, tienen la función de cambiar la cadena de un piñón a otro o de un plato a otro dependiendo de la instrucción que se ordene con los mandos, una función secundaria es el mantener la cadena en su posición de trabajo evitando que se desborde o brinque a otra posición que no le corresponde. Normalmente las bicicletas cuentan con 2 desviadores, una ubicada junto a los platos y la otra junto a los piñones y suelen ser de distinta forma.

Velocidades: Son cada una de las combinaciones posibles entre platos y piñones. Es decir, si una bicicleta tiene un sólo plato y tres piñones será de tres velocidades; una de tres platos y seis piñones sería de dieciocho, etcétera. La cantidad de velocidades adecuada para cada usuario dependerá de la finalidad para la que se use y por supuesto, el tipo de terreno sobre el que rodara esta.

Mandos: La selección de velocidades se logra generalmente a través del control de la tensión de chicotes conectados a los desviadores o cadena indicadora (para el caso de los cambios internos) por medio de los mandos. Existen también mandos hidráulicos y eléctricos pero estos son menos comunes y más costosos. Los mandos se encuentran ubicados en el manillar o manubrio de la bicicleta y generalmente existen dos tipos de mandos los cuales son de palanca y de puño.

1.5.3 Funcionamiento correcto del sistema de cambios de velocidades

En la mayoría de los casos las personas no tienen una noción clara de la combinación correcta de platos y piñones que debe ser utilizada en los distintos escenarios que se pueden presentar en un recorrido e.g. subidas, pendientes, etc.... El algoritmo de programación que se planea en el presente proyecto debe ser capaz de reconocer la combinación correcta en cada escenario, es decir, debe enviar la acción y/o información necesaria a los actuadores de los cambios de velocidades para aprovechar al máximo la potencia transmitida por el ciclista hacia las ruedas. Como se mencionó anteriormente, el proyecto se encuentra diseñado para un ambiente urbano, por tal motivo, los escenarios que pueden estar presentes en dicho ambiente son tomados en consideración.

A continuación se explica brevemente la forma de elegir adecuadamente la combinación ideal de los cambios de velocidades y los criterios sobre los que se debe basar el usuario para dicha selección.

Como es sabido, el sistema de cambios puede tener hasta tres platos y nueve piñones y el orden de estos se encuentra de forma opuesta. El ordenamiento de los platos y piñones es lo primero que debemos tener en cuenta ya que combinar el piñón

más grande con el plato más grande o el plato más chico con el piñón más chico puede provocar que la cadena quede en una posición inadecuada en la que se exige al máximo sus eslabones y como consecuencia se puede tener un menor tiempo de vida útil en el sistema de cambios de velocidades (ver hoja de datos en Anexo A.1).

En la Figura 1.8, se muestra un esquema de las combinaciones que representan un mayor desgaste en los piñones y platos, debido a la tensión que ejerce la cadena. Estas combinaciones de velocidades no son recomendadas por parte de los fabricantes de este tipo de transmisiones [10].

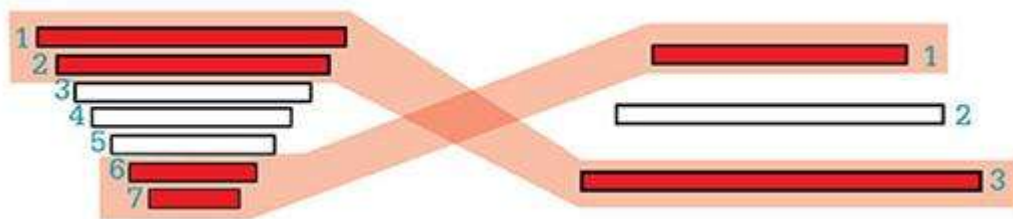


Figura 1.8 Uso no recomendado del sistema de cambios [10]



Figura 1.9 Combinaciones recomendadas [10]

La figura 1.9 muestra las combinaciones más seguras y que por lo tanto están lejos de representar un problema para el sistema, pero ojo que no quiere decir que sean todas las posibles combinaciones útiles del sistema. Se observa que los 3 piñones (3, 4 y 5) y el plato en color verde (2) dan como resultado 3 posibles marchas,

pero si se agregan las combinaciones de los piñones con los platos 1 y 3, se tienen en total 13 marchas.

Ahora bien debemos saber que la combinación del plato más grande el cual supongamos que tiene 50 dientes con el piñón más chico el cual suponemos que tiene 10 dientes, representa el mayor esfuerzo físico exigido debido a que 1 vuelta completa del plato equivale a 5 vueltas del piñón que está sujeto a la rueda trasera. Por el contrario la combinación que representa menos esfuerzo físico sería la del plato más pequeño con el piñón más grande. Así que para el caso de un sistema de 1 plato y 3 piñones existen tres posibles combinaciones mientras, que para un sistema de 2 platos y 9 piñones contamos con 18 marchas o como se conoce comúnmente 18 velocidades. [10]

La mayoría de las personas cuando empiezan a utilizar los cambios cometen errores como arrancar en una posición muy dura o colocar una posición suave cuando se rueda en una bajada o se alcanza una velocidad alta. La mejor manera de arrancar es usando una combinación que requiera poco esfuerzo es decir, escoger un plato pequeño y un piñón grande, mientras la velocidad se incrementa también se debe de cambiar la combinación gradualmente de acuerdo a la tabla de combinaciones que se calcula para cada bicicleta.

Existen varias formas de crear una tabla para observar la fuerza de cadencia de cada combinación o bien la potencia transmitida, incluso hay una calculadora llamada “calculadora de Sheldon Brown” la cual toma en cuenta todos los aspectos que tiene la bicicleta como el tamaño de las ruedas, el tipo de platos, el tipo de piñones así como los dientes de cada uno [12]. Esta calculadora es muy útil debido a que se puede saber si dos combinaciones diferentes de plato-piñón proporcionan la misma fuerza y de esta manera el rango de velocidades se reduce un poco y el sistema se hace menos complejo. A continuación se muestra como ejemplo en la Tabla 1.1 las posibles combinaciones para un sistema de cambios de tres platos y ocho piñones.

Tabla 1.1 Ejemplo de posibles combinaciones de la relación Plato-Piñón [10]

potencia 1	=	plato 1	x	piñón 1
potencia 2	=	plato 1	x	piñón 2
potencia 3	=	plato 1	x	piñón 3
potencia 4	=	plato 1	x	piñón 4
potencia 5	=	plato 1	x	piñón 5
potencia 6	=	plato 1	x	piñón 6
potencia 7	=	plato 1	x	piñón 7
potencia 8	=	plato 1	x	piñón 8
potencia 2	=	plato 2	x	piñón 1
potencia 4	=	plato 2	x	piñón 2
potencia 6	=	plato 2	x	piñón 3
potencia 8	=	plato 2	x	piñón 4
potencia 10	=	plato 2	x	piñón 5
potencia 12	=	plato 2	x	piñón 6
potencia 14	=	plato 2	x	piñón 7
potencia 16	=	plato 2	x	piñón 8
potencia 3	=	plato 3	x	piñón 1
potencia 6	=	plato 3	x	piñón 2
potencia 9	=	plato 3	x	piñón 3
potencia 12	=	plato 3	x	piñón 4
potencia 15	=	plato 3	x	piñón 5
potencia 18	=	plato 3	x	piñón 6
potencia 21	=	plato 3	x	piñón 7
potencia 24	=	plato 3	x	piñón 8

En la Tabla 1.1 se pueden observar las veinticuatro posibles combinaciones de plato-piñón, en color negro se muestran las combinaciones útiles, en color rojo las combinaciones que se deberán evitar para no dañar el sistema de transmisión y en naranja las combinaciones que producen el mismo resultado, es decir se aplica la misma fuerza para pedalear con dos o más combinaciones distintas.

1.6 Objetivo

En este proyecto se tiene la finalidad de crear un sistema de cambios de velocidades automático para una bicicleta. El sistema debe realizar los cambios en el momento preciso aprovechando de la mejor manera la energía que se ejerce al pedalear.

El diseño del sistema está planteado para poder ser aplicado en cualquier bicicleta convencional que cuente con un sistema de transmisión, con la posibilidad de ser adaptable de acuerdo a la necesidad del ciclista.

Con el desarrollo del presente proyecto se pretende demostrar las capacidades y destrezas adquiridas a lo largo de la carrera asimilando, comprendiendo y adoptando una nueva tecnología y un lenguaje de programación distinto a lo habitual pero que es muy eficiente y versátil.

1.7 Organización de la tesis

En el capítulo 2 se describe la implementación del código, teniendo en cuenta los criterios planteados en el capítulo 1. Para este capítulo se plantean tres secciones principales descritas a continuación: primeramente se aborda la planeación, en segundo lugar la evaluación del problema mediante diagramas de flujo, y finalmente se presenta el algoritmo propuesto.

En el capítulo 3 se presenta el modelado del circuito general en el cual se diseña todo el sistema eléctrico que consta del sistema de alimentación y el sistema de control. Aquí se podrán encontrar los subtemas de planeación, diseño y acoplamiento de todas las partes del sistema y correcciones a posteriori del sistema. Al final de este capítulo se darán a conocer los resultados del proyecto, las observaciones hechas en el proceso y durante la experimentación del diseño y algunas recomendaciones enfocadas al funcionamiento del sistema.

El capítulo 4 fue destinado a probar el funcionamiento del proyecto mediante seis distintas pruebas las cuales fueron registradas, tabuladas y graficadas para un análisis más sencillo. En dichas pruebas se registraron los parámetros de velocidad, marcha anterior, marcha nueva y tiempo.

Finalmente en el capítulo 5 se abordan las conclusiones del proyecto, una autocrítica del diseño y se habla de los trabajos futuros a los que el proyecto se puede extender.

Capítulo 2. Implementación del algoritmo

En el presente capítulo se presenta la implementación del algoritmo utilizado para la realización del cambio de velocidades automático. Primeramente se presenta la planeación de su obtención, así como también las condiciones tomadas en cuenta para su desarrollo, para posteriormente presentar los diagramas de flujo del código implementado.

2.1 Planeación

Dentro de la concepción del proyecto, en un inicio se idearon diferentes soluciones a implementar para resolver la problemática expuesta anteriormente. De todas las ideas iniciales, se optó por elaborar un diseño simple y económico, pero que cuente con la misma calidad que los diseños descartados.

Por otra parte, el costo de implementar un diseño más simple en hardware tiene un precio más alto en el software, es decir, la programación del microcontrolador se vuelve un poco más compleja debido a que se tiene que lograr el mismo objetivo, con menos información.

El primer paso en este proceso consiste en identificar todas las partes y variables que conforman el proyecto, las cuales tiene relación directa con el algoritmo, tales como: las entradas del sistema, las salidas y la velocidad máxima de la bicicleta, etc. Como se mencionó en el párrafo anterior, en un principio el proyecto se planteó para hacer uso de una mayor cantidad de variables de entrada (las cuales representan las variables físicas a medir). Se tenían contemplados sensores de presión, posición e infrarrojos como soporte para calcular la velocidad; sin embargo, en la inspección del problema se observó que algunas variables proporcionaban información redundante, tal es el caso de los sensores de presión y posición, los cuales serán descritos más adelante.

Finalmente, en el sistema se consideró únicamente una sola variable, que en este caso es la velocidad angular que se mide con la ayuda de un interruptor magnético, el cual sustituye al sensor infrarrojo, debido a su precisión y confiabilidad de respuesta.

La posición, fuerza y velocidad son tres de las variables (asociadas a sensores) que pueden ser utilizadas para realizar la implementación del sistema. Con estas variables se pretendía conocer los datos de inclinación de la bicicleta, para identificar si la bicicleta rueda por una subida o una bajada; la velocidad de la bicicleta y la fuerza que ejerce el ciclista en el pedaleo.

Además, se pretendía medir la masa corporal del ciclista con la ayuda de un par de sensores de fuerza ubicados en los soportes de las ruedas y un sensor infrarrojo en el asiento para conocer la estatura de la persona que montaba la bicicleta, tomando como referencia la altura del asiento, sin embargo esta idea fue descartada desde el inicio, debido a que se decidió hacer un diseño sencillo, fácil de implementar pero que fuera eficiente.

Realizando un análisis del problema con el uso de las tres variables o sensores en conjunto, se deduce que con el uso del sensor de fuerza y el de velocidad, se puede saber si la bicicleta rueda por una pendiente de subida (positiva) o de bajada (negativa), por lo tanto se concluye que el sensor de posición (giroscopio) no es indispensable para el objetivo que tiene el sistema.

Debido al uso de solo dos sensores (variables), el sistema se simplifica en cuanto a disminución de componentes utilizados, así como también en una disminución de las instrucciones que contiene el código. Dicho de otra manera, esta selección permite obtener un ahorro de energía, y aumenta la velocidad de respuesta y procesamiento en el sistema.

De manera similar, como en el caso anterior, se procede a rectificar el diseño del proyecto. Sin embargo, al realizar una inspección más detallada y con la ayuda del diagrama de flujo de la Figura 2.1, se observa que el sensor de fuerza también puede ser omitido en el diseño, debido a que el cambio de velocidad en intervalos de tiempo predefinidos es suficiente para conocer el estado de la bicicleta. Esto significa que el sistema puede seleccionar el estado correcto únicamente con el censado de velocidad y puede prescindir de cualquier otra variable.

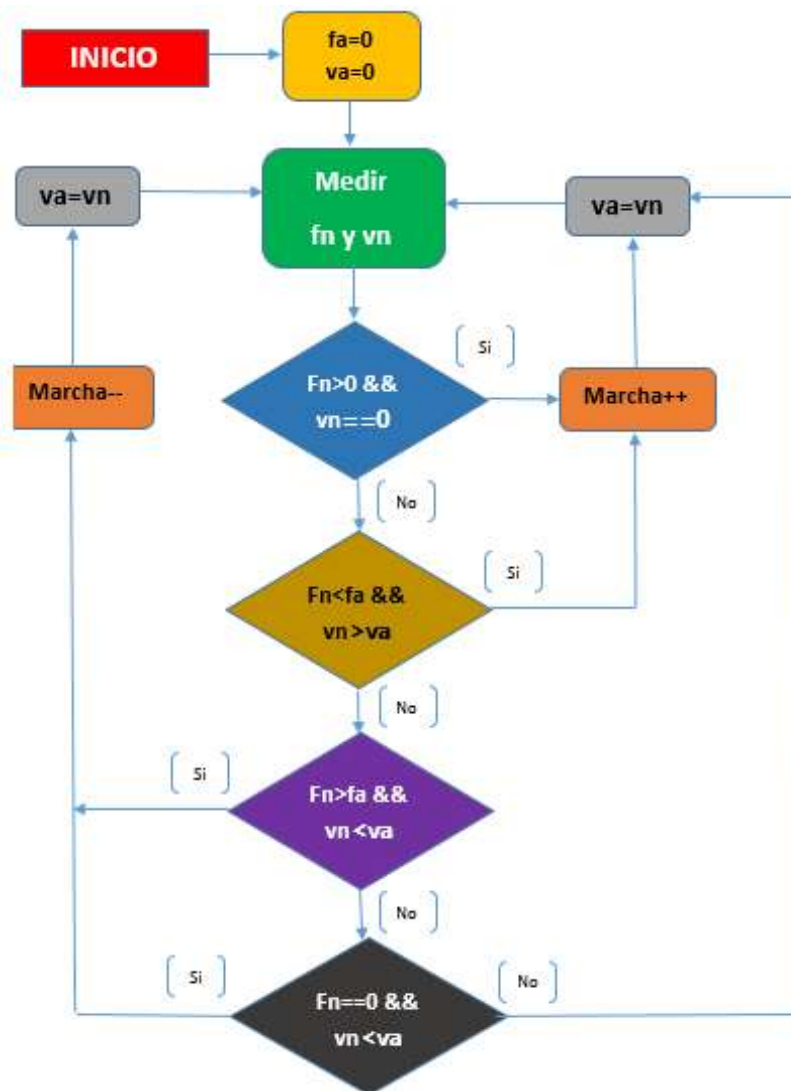


Figura 2.1 Diagrama de flujo para la eliminación de la variable de fuerza

Donde:

fa = fuerza nueva

fn = fuerza anterior

vn = velocidad nueva

va = velocidad anterior

En la Figura 2.1 se presenta el diagrama de flujo en el cual se hace evidente la omisión de la variable de fuerza.

El diagrama comienza con la medición de las variables de fuerza y velocidad para después tomar una decisión con una de las 4 condiciones definidas. Estas 4 condiciones representan 4 posibles estados:

1. El primero ($fn > 0 \ \&\& \ vn == 0$), indica que el ciclista está en posición de arranque pero la velocidad es cero, y por lo tanto la marcha cambia a una posición más suave (si la posición de la marcha estuviera en la posición más dura el sistema sigue cambiando hasta la marcha más baja).
2. La segunda condición ($fn < fa \ \&\& \ vn > va$), indica que se ejerce menos fuerza en el pedaleo y la velocidad incrementa, por lo tanto se aplica al incrementar la marcha por una más dura.
3. La tercera condición ($fn > fa \ \&\& \ vn < va$), indica que la fuerza ejercida en el pedaleo aumenta pero la velocidad disminuye, entonces se entiende que la bicicleta rueda por una subida y por lo tanto se deberá de cambiar a una marcha más suave.
4. La cuarta condición ($fn == 0 \ \&\& \ vn < va$), considerada la más importante, indica que el ciclista dejó de ejercer fuerza en los pedales pero la velocidad sigue cambiando, esto quiere decir que la bicicleta pierde velocidad ya sea de manera natural o por acción del freno.

La última condición deberá disminuir la marcha por una más suave, debido a que si el ciclista decide volver a pedalear, el sistema deberá de proporcionarle una cadencia moderada. Además, esta cuarta condición también se puede aplicar para el caso en que no detecte fuerza en los pedales y la velocidad incremente debido a una bajada y por lo tanto el sistema deberá de proporcionar una cadencia adecuada a la velocidad aumentada.

Es con la cuarta condición que se puede llegar a la conclusión de suprimir el sensor de fuerza ya que el sistema puede tomar decisiones con tan solo detectar cambios en la velocidad.

Finalmente, teniendo en cuenta lo expuesto anteriormente, en el diseño se puede crear un criterio de toma de decisiones basado en una sola variable de velocidad, tomando como referencia el cambio de la velocidad en determinados intervalos de tiempo, es decir, toma como referencia la aceleración del sistema.

El hecho de utilizar una sola variable física tiene un costo, debido a la pérdida de información que se pueda tener. Tal es el caso de conocer la posición de la cadena, debido a que el sistema no identificará en realidad que la cadena está en la posición correcta, simplemente aplicara la tensión indicada a los chicotes que controlan la posición de esta. Entonces, esto quiere decir que será necesario calibrar el sistema de cambios y evitar la pérdida de tensión en los chicotes para que el sistema funcione de forma satisfactoria.

Tomando en cuenta lo expuesto en la presente sección, para el desarrollo del algoritmo de control, se nombran de manera definitiva una variable de entrada y dos variables como salida: Velocidad (V) y dos Voltajes de salida (servo1 y servo2), respectivamente.

La Tabla 2.1 resume las variables utilizadas en la gestión del algoritmo.

Tabla 2.1 Variables de entrada y salida

Variables del sistema	
Entrada	Salida
Velocidad (V)	Servo1
	Servo2

Como se puede apreciar en la Tabla 2.1 las variables son pocas, lo que permite que el sistema sea simple y de bajo consumo de recursos. Sin embargo, esto implica un mayor esfuerzo en el diseño del algoritmo, debido a que los criterios de programación deben estar bien formulados para no dejar pasar algún detalle no considerado debido a la limitación de información.

2.2 Criterios para el diseño del sketch:

Los criterios considerados para realizar la programación del sketch son los siguientes:

- i. Se establecen las condiciones iniciales $marchaA=1$ y $marchaN=1$
- ii. El análisis del algoritmo debe ser simple y ordenado.

- iii. Definir el tiempo mínimo de espera para hacer el siguiente cambio de marcha.
- iv. Cuantificación de la relación velocidad-marcha.
- v. Definir las combinaciones de plato-piñón útiles con el método “Pulgadas de engranaje”.

Utilizando estos criterios, se formula el algoritmo de programación.

Algunas de las variables necesarias para la implementación de dicho algoritmo son descritas a continuación.

2.2.1 Cálculo de la Velocidad

El cálculo de la velocidad se realiza mediante un interruptor magnético de circuito abierto ubicado en la parte delantera del cuadro de la bicicleta, éste es colocado de tal forma que se encuentra ubicado de manera paralela a la rueda delantera, donde se instaló un imán permanente de neodimio, el campo magnético del imán activa el interruptor cada vez que pase por el frente de este.

La técnica para hacer buen uso de este método es conocer la longitud del campo magnético del imán (que representa la distancia) y censar el tiempo que dura este campo magnético cuando pasa frente al interruptor para completar la fórmula de velocidad $v = d/t$, donde d representa la distancia y t el tiempo transcurrido.

Para identificar la longitud del campo magnético del imán permanente, se utiliza el siguiente método:

1. Se instala el interruptor magnético y el imán permanente en la ubicación mencionada en el párrafo anterior. Un led indicador conectado en serie con el sensor se enciende cada vez que el imán pasa frente al interruptor.
2. Se hace rodar la bicicleta y se mide la distancia recorrida mientras el led se encuentra encendido.
3. La prueba se realiza varias veces para contemplar la variación entre mediciones y verificar la confiabilidad del método.

Para obtener la variable de tiempo se utiliza el interruptor magnético en conjunto con el sketch donde se implementa una función que cuenta el tiempo en que el interruptor se encuentra cerrado.

2.2.1.1 Validación de la medición de velocidad

Para corroborar el funcionamiento de la medición de velocidad se plantea el registro de una serie de pruebas en las que se registra la velocidad máxima con dos métodos simultáneamente. Los métodos usados son el propuesto en este proyecto con el interruptor magnético y otro método ya comprobado como lo es un sistema GPS implementado en un teléfono celular.

Los registros obtenidos se presentan en la Tabla 2.2.

Tabla 2.2 Registro de las pruebas de velocidad

Prueba	Método	Interruptor magnético	GPS
Prueba 1		22 km/h	22.4 Km/h
Prueba 2		23 km/h	23.6 km/h
Prueba 3		20 km/h	20.7 km/h

El método con el interruptor magnético registra valores redondeados al entero más bajo mientras que el método GPS registra valores flotantes debido a esto se tiene una diferencia de décimas, sin embargo los valores registrados son muy similares y con esto se puede aprobar el funcionamiento del método implementado en este proyecto.

Nota: El algoritmo utilizado para estas pruebas es la función de velocidad que se utiliza en el sketch principal además de la comunicación serial para registrar los datos en un servidor portátil.

2.2.2 Pulgadas de Engranaje

Como se mencionó en el capítulo 1, es necesario calcular el trabajo realizado por cada combinación de plato-piñón del sistema de transmisión. Esto se hace con la finalidad de omitir diferentes combinaciones de plato-piñón, ya sea por prevención (evitar desgaste del sistema de transmisión) o porque se repiten los resultados con distintas combinaciones (se obtienen el mismo par aplicado en el sistema). Para poder identificar las diferentes combinaciones, es necesario realizar algunos cálculos, los cuales son descritos a continuación.

Para identificar las combinaciones que se deben omitir por cuestiones preventivas no es necesario realizar algún cálculo, debido a que el fabricante del sistema de cambios proporciona la información de dichas combinaciones que deben suprimirse.

En el caso de las combinaciones que deben evitarse debido a su repetición, es necesario elegir algún método que permita realizar su identificación. En este caso se eligió el método de pulgadas de engranaje (Gear inches) [13] el cual calcula el trabajo realizado al pedalear donde el valor de cada combinación es directamente proporcional con la fuerza requerida para dar un giro a la biela de la bicicleta. El cálculo de las pulgadas de engranaje se lleva a cabo con la expresión (2.1).

$$G_i = d \left(\frac{n}{m} \right) \quad (0.1)$$

Donde:

G_i = Pulgadas de engranaje

d = diámetro de la rueda trasera

n = número de dientes del plato

m = número de dientes del piñón

Las ruedas más comunes son de 26 pulgadas, mientras que el sistema de transmisión usado es el de 2 platos y 9 piñones. El sistema de transmisión considerado para el desarrollo del prototipo, tiene las siguientes características:

- Se consideran dos platos. El plato 1 (el plato ubicado en la parte interior) tiene 24 dientes y el plato 2 cuenta con 34 dientes.
- Se consideran 9 piñones. El número de dientes de los piñones empezando con el piñón 1 terminando con el piñón 9 es de: 12,14, 16, 18, 21,24, 28, 32 y 36,

Utilizando la expresión (2.1) y los datos proporcionados, se formula la Tabla 2.3.

Tabla 2.3 Calculo de las marchas mediante el método de Pulgadas de Engranaje

Piñones\Platos	24	34
12	54	76.5
14	46.28	65.57
16	40.5	57.37
18	36	51
21	30.85	43.71
24	27	38.25
28	23.14	32.78
32	20.25	28.68
36	18	25.5

Como se puede observar en la Tabla 2.3, se tienen 18 posibles combinaciones, de las cuales, las de color rojo son descartadas debido a que estas combinaciones son perjudiciales para el sistema (información dada por el fabricante del sistema de cambios Shimano). [11]

Por otra parte, las combinaciones en color azul y verde son resaltadas de esta manera, debido a que presentan un valor muy cercano y su funcionamiento será muy parecido. Debido a esto, una de las dos opciones en cada uno de los casos es descartada.

Para realizar la elección de las marchas útiles, se eligen las combinaciones que ayuden a no cortar la secuencia o por lo menos a no hacerlo de manera brusca, por

ejemplo las combinaciones marcadas en verde que tienen las combinaciones 24-21 y 34-28 (plato-piñón) donde se ve claramente que si se elimina la combinación 24-21, se cortaría la secuencia de cambios, ya que el proceso se saltaría de la combinación 24-24 a la 24-18; mientras que la combinación 34-28 no tiene combinación que le anteceda (sabiendo que las marchas marcadas en rojo no están siendo tomadas en cuenta) y no genera un desbalance en la secuencia. Aplicando este criterio se obtiene un patrón sencillo en el seguimiento de la secuencia. Para una mayor comprensión se presenta la Tabla 2.4 donde se muestran solo las marchas útiles y se omiten las combinaciones que ya fueron descartadas.

Tabla 2.4 Sistema reducido a 12 Marchas

Piñones\Platos	24	34
12	-	76.5
14	-	65.57
16	40.5	57.37
18	36	51
21	30.85	43.71
24	27	-
28	23.14	-
32	20.25	-
36	18	-

Comparando las Tablas 2.3 y 2.4, se tiene que el sistema de transmisión se reduce de 18 a 12 marchas y se puede observar que no existen cambios bruscos en la secuencia. Sin embargo, solamente se tienen 2 casos especiales inevitables que se presentan cuando se cambia de plato: el caso cuando se pasa de la marcha 24-16 a la marcha 34-21, en el cual se debe de cambiar del plato de 24 dientes al plato de 34 y del piñón de 16 dientes al de 21, esto significa que habrá 1 cambio para los platos y 2 para los piñones en el cambio de una marcha a otra y lo mismo pasa cuando se hace el cambio de la marcha 34-21 a la marcha 24-16, pero fuera de estos 2 casos especiales en todos los demás casos solamente habrá un cambio de piñón ya sea al incrementar la marcha o al disminuirla.

Aunado a la selección de las diferentes combinaciones de marcha, se debe identificar también una combinación de marcha “base” o de arranque, a partir de la cual

se comenzaran a realizar los diferentes cambios de marcha, siempre regresando a esta marcha de arranque cuando se comience a utilizar la bicicleta.

La marcha de arranque seleccionada, es la combinación de plato-piñón 24-36. Esta marcha de arranque es identificada como la combinación 1 (ver Tabla 2.5).

De las posibles combinaciones identificadas en la Tabla 2.5, se puede observar que la marcha que representa un mayor trabajo es la que corresponde a la combinación 12 (relación plato-piñón de 34-36).

Tabla 2.5 Orden de las Marchas

Combinación	Plato-Piñón
1	24-36
2	24-32
3	24-28
4	24-24
5	24-21
6	24-18
7	24-16
8	34-21
9	34-24
10	34-28
11	34-32
12	34-36

2.2.3 Cuantificación

La cuantificación del sistema se calcula con dos parámetros: el número de combinaciones útiles y la velocidad máxima de la bicicleta en un ambiente urbano y tiene la finalidad de ponderar cada una de las marchas a su respectivo rango de velocidad para optimizar el desempeño del sistema de cambios y proporcionar una cadencia promedio en cualquier valor de velocidad.

La ecuación (2.2) describe la cuantificación C_i calculada, en relación a la velocidad máxima V_{max} y el número de combinaciones útiles M_{utiles} .

$$C_t = \frac{V_{\max}}{M_{\text{útiles}}} \quad (0.2)$$

Donde la velocidad máxima urbana de referencia es $V_{\max} \approx 35 \text{ km/h}$, y el número de combinaciones útiles $M_{\text{útiles}} = 12$ (de la Tabla 2-4).

$$C_t \approx \frac{35 \frac{\text{km}}{\text{h}}}{12 \text{ marcha}} \approx 2.916 = 3 \text{ km/h} * \text{marcha}$$

Por lo tanto, el valor de cada escalón está dado por:

$$\text{Escalon}_M = M * C_t = 3 * M \quad (2.3)$$

Donde M representa el número de marcha (del 1 al 12) asignado en la Tabla 2.4.

La Tabla 2.6, derivada de los cálculos anteriores, muestra las relaciones a considerar para identificar la combinación de los cambios en la bicicleta.

Tabla 2.6 Relación Marcha-Cuantificación

M	Plato-Piñón	C_t
1	24-36	1-3
2	24-32	4-6
3	24-28	7-9
4	24-24	10-12
5	24-21	13-15
6	24-18	16-18
7	24-16	19-21
8	34-21	22-24
9	34-24	25-27
10	34-28	28-30
11	34-32	31-33
12	34-36	34-36

2.2.4 Control de cadencia

El control de cadencia está pensado para la comodidad del ciclista y tiene el objetivo de poder brindar la opción de elegir entre un pedaleo más suave o uno más duro, de acuerdo a la superficie sobre la que se encuentre el ciclista. En este trabajo, se realiza el diseño de solamente 2 cadencias, sin embargo, en teoría se pueden implementar los niveles de cadencia que se deseen.

En la sección 2.2.3 se presentó el cálculo de la cuantificación de la cadencia principal del sistema, para un promedio máximo de velocidad y un determinado número de marchas, dando como resultado un escalón de $3 \text{ km/h} * \text{marcha}$. Sin embargo, si se cambia el tamaño del escalón se cambia también el nivel de cadencia indirectamente, ya que el sistema realizará los cambios (más rápido o más lento) dependiendo del tamaño del escalón, por ejemplo, si se tiene un escalón de $3 \text{ km/h} * \text{marcha}$ y una velocidad de 11 km/h , el sistema asignará a dicha velocidad la $M=4$, pero si se cambia el valor del escalón a $4 \text{ km/h} * \text{marcha}$ la misma velocidad será asignada a la $M=3$. Este cambio en la cuantificación presenta la ventaja de poder seleccionar en plena actividad el nivel de cadencia con tan solo un Switch (interruptor).

Es este proyecto se manejan dos cadencias diferentes, las cuales son nombradas simplemente como cadencia 1 y cadencia 2. La cadencia 1, corresponde a los datos presentados en la Tabla 2.6, los cuales consideran un escalón de $3 \text{ km/h} * \text{marcha}$. La cadencia 2, tiene un escalón de $4 \text{ km/h} * \text{marcha}$, por lo tanto el nivel de cadencia será más suave debido a lo explicado anteriormente.

La Tabla 2.7, presenta los datos utilizados para la cadencia 2.

Teniendo en cuenta el análisis presentado anteriormente, estas combinaciones son utilizadas para diseñar el algoritmo de control.

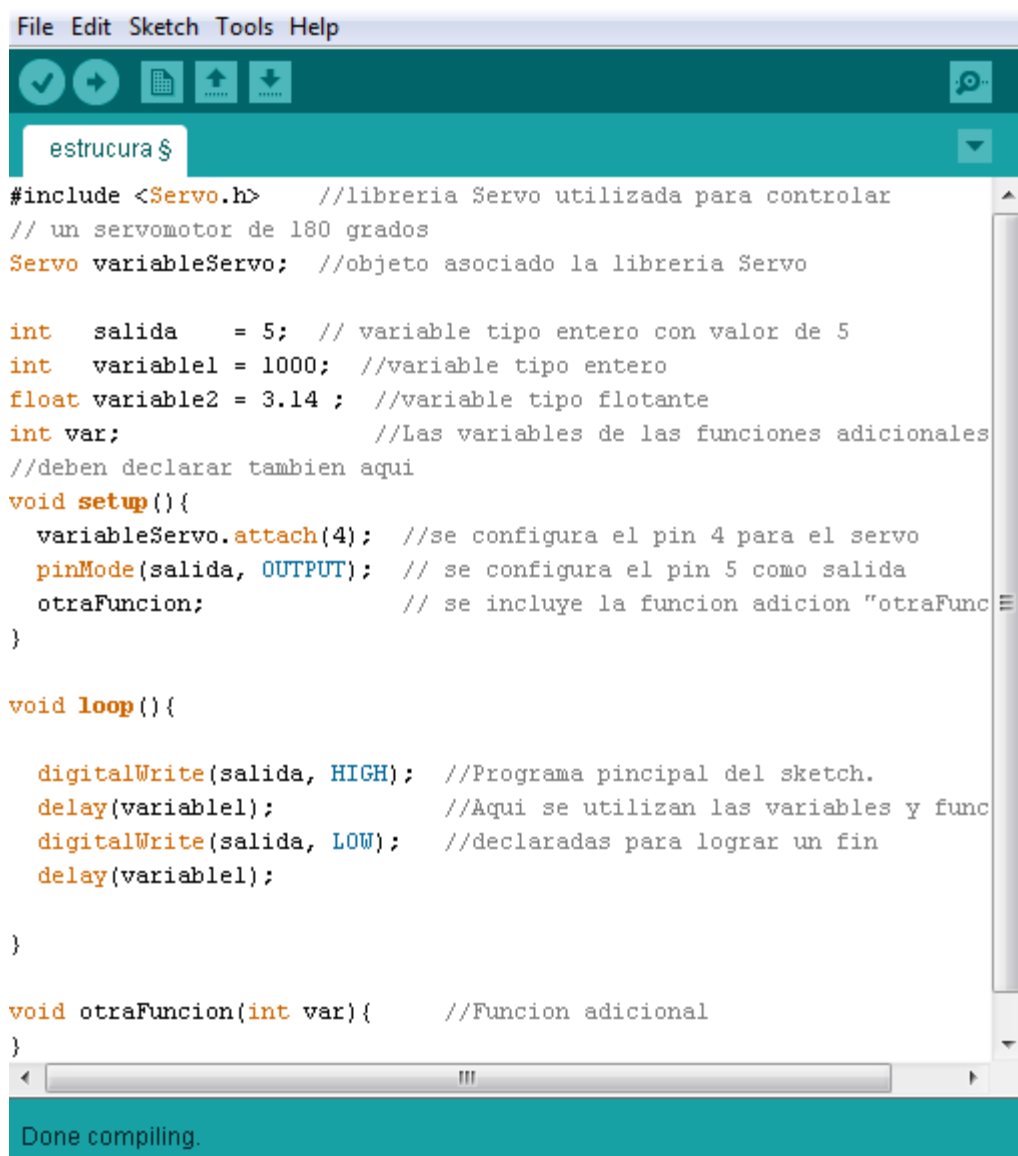
Tabla 0.7 Relación Marcha – Cuantificación Cadencia 2.

M	Plato-Piñón	C_t
1	24-36	1-4
2	24-32	5-8
3	24-28	9-12
4	24-24	13-16
5	24-21	13-15
6	24-18	16-18
7	24-16	19-21
8	34-21	22-24
9	34-24	25-27
10	34-28	28-30
11	34-32	31-33
12	34-36	34-36

2.3 Estructura del Sketch

La plataforma Arduino tiene una estructura bien definida para la construcción de cualquier sketch, cuenta con su propio lenguaje de programación basado en C++ y los sketches pueden ser creados y cargados desde su entorno de desarrollo IDE (Integrated development environment). La estructura se encuentra conformada básicamente de un espacio dedicado a la declaración de variables y/o librerías y de dos funciones indispensables llamadas *setup* y *loop* además se pueden agregar funciones por el programador con la finalidad de darle más claridad a la estructura del sketch (estas últimas pueden ubicarse antes o después de las funciones *loop* y *setup*).

La Figura 2.2 muestra la estructura que presenta el sketch [14].



```
File Edit Sketch Tools Help

estructura$

#include <Servo.h> //libreria Servo utilizada para controlar
// un servomotor de 180 grados
Servo variableServo; //objeto asociado la libreria Servo

int salida = 5; // variable tipo entero con valor de 5
int variable1 = 1000; //variable tipo entero
float variable2 = 3.14 ; //variable tipo flotante
int var; //Las variables de las funciones adicionales
//deben declarar tambien aqui
void setup(){
    variableServo.attach(4); //se configura el pin 4 para el servo
    pinMode(salida, OUTPUT); // se configura el pin 5 como salida
    otraFuncion; // se incluye la funcion adiccion "otraFuncion"
}

void loop(){

    digitalWrite(salida, HIGH); //Programa principal del sketch.
    delay(variable1); //Aquí se utilizan las variables y funciones
    digitalWrite(salida, LOW); //declaradas para lograr un fin
    delay(variable1);

}

void otraFuncion(int var){ //Funcion adicional
}

Done compiling.
```

Figura 2.2 Estructura del Sketch

De la Figura 2.2, se puede observar que el sketch se inicia con la declaración de las librerías, seguidas de la declaración de las variables que serán utilizadas en el código y de todas las funciones que lo conformen.

Después, debe seguir la declaración de la función *setup*, que tiene como objetivo configurar los pines (conectores físicos) de la placa Arduino y reconocer las funciones adicionales del sketch, las cuales se declaran después de la función *loop*.

Una vez definida la función *setup*, se define la función *loop*, la cual tiene como objetivo el uso de todas las variables y librerías incluidas al inicio del código.

Finalmente, una vez declarada la función *loop*, pueden ser declaradas o formuladas las funciones adicionales que se requieran para el correcto funcionamiento del programa.

2.4 Diagramas de flujo

El código o Sketch cuenta con 7 funciones, primero se encuentran las 2 funciones principales predefinidas ya mencionadas (*loop* y *setup*) donde la función *loop* es definida por Arduino como la función principal, sin embargo, en el presente trabajo se define a esta función como función “Control del sistema”. Las demás funciones son las siguientes: “Velocidad”, “cadencia1”, “cadencia2”, “subirMarcha” y “bajarMarcha”. La finalidad de crear estas funciones es en seguimiento a uno de los criterios expuestos, el cual propone presentar un código simple y de fácil comprensión.

La función “Control del sistema”, inicia calculando la velocidad con la ayuda de la función con el mismo nombre, en seguida detecta con un ciclo **if** el nivel de cadencia (cadencia 1 o cadencia 2) revisando el estado de la entrada digital 12 (low = cadencia 1 y high = cadencia 2), cuando entra en la cadencia correspondiente asigna a la velocidad calculada el valor de su marcha correspondiente (usando las funciones “cadencia 1” o “cadencia 2” según sea el caso), después revisa si la marcha aumentó o decreció para usar la función “subirMarcha” o “bajarMarcha” según sea el caso, para que finalmente estas últimas funciones manden la señal correspondiente a los servomotores para que ejecuten los cambios necesarios en el sistema.

Como se mencionó anteriormente, la elaboración de los diagramas de flujo facilitan la inspección del problema y proporcionan un enfoque más claro que permite comprender el problema de forma más rápida y sencilla. Los diagramas de flujo pueden ayudar a detectar errores, así como procesos obsoletos y procesos que hacen falta, es por eso que el presente trabajo se respalda en la elaboración de dos diagramas de flujo como herramienta fundamental para el diseño del sistema. A continuación se muestran los diagramas de flujo y las descripciones de cada una de las dos funciones.

2.4.1 Diagrama de flujo “Control del sistema”

La Figura 2.3 muestra el diagrama de flujo del algoritmo para la función “Control del sistema”.

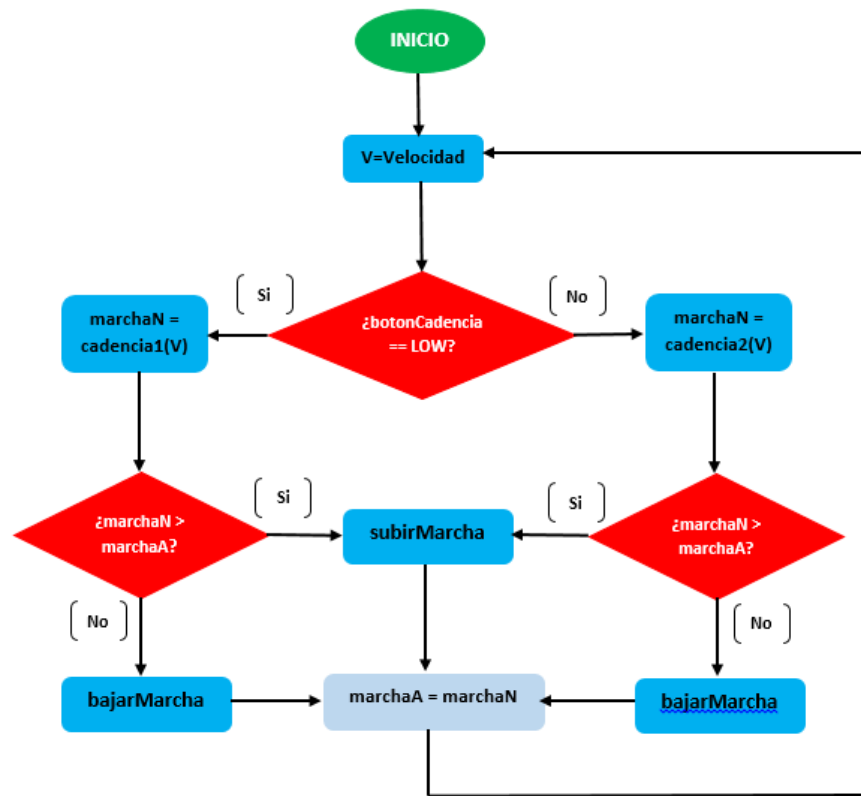


Figura 2.3 Diagrama de flujo de la función Control del sistema

La función central *loop* o “Control del sistema”, inicia calculando la velocidad llamando a la función “Velocidad” y guarda la información una variable de asignación denotada como V. Después, pregunta en qué estado se encuentra la entrada digital correspondiente a la selección de cadencia, si el estado es LOW o bajo (0 volts), el programa utiliza la cadencia 1 para asignarle el número de marchar correspondiente a la velocidad previamente calculada. En el caso de tener un estado HIGH o alto (5 volts), se usara la cadencia 2.

El diagrama continúa con la identificación de la velocidad en aumentó o disminución, lo cual se realiza con la ayuda de un ciclo *if*, donde su condición pregunta si la marchaN (marcha nueva) es mayor a marchaA (marcha anterior), y en caso de ser

mayor ($marchaN > marchaA$) representará un aumento de velocidad y en el caso contrario un decremento. La función “subirMarcha” se activa cuando se identifica un incremento, mientras que la función “bajarMarcha” responde a la disminución de velocidad. Después del paso anterior, el sistema actualiza la variable *marchaA* y vuelve a comenzar el ciclo.

Finalmente, en el caso de que no se detecte un cambio en la velocidad, el programa omite el proceso de subir o bajar la marcha y solamente actualiza la variable *marchaA* para posteriormente volver a comenzar el ciclo.

2.4.2 Diagrama de flujo de la función “Velocidad”

La función de “Velocidad” como su nombre lo indica, permite realizar el cálculo de la velocidad de la bicicleta, para posteriormente hacer una selección correcta del cambio de velocidad. La Figura 2.4 muestra el diagrama de flujo de la función “Velocidad”.

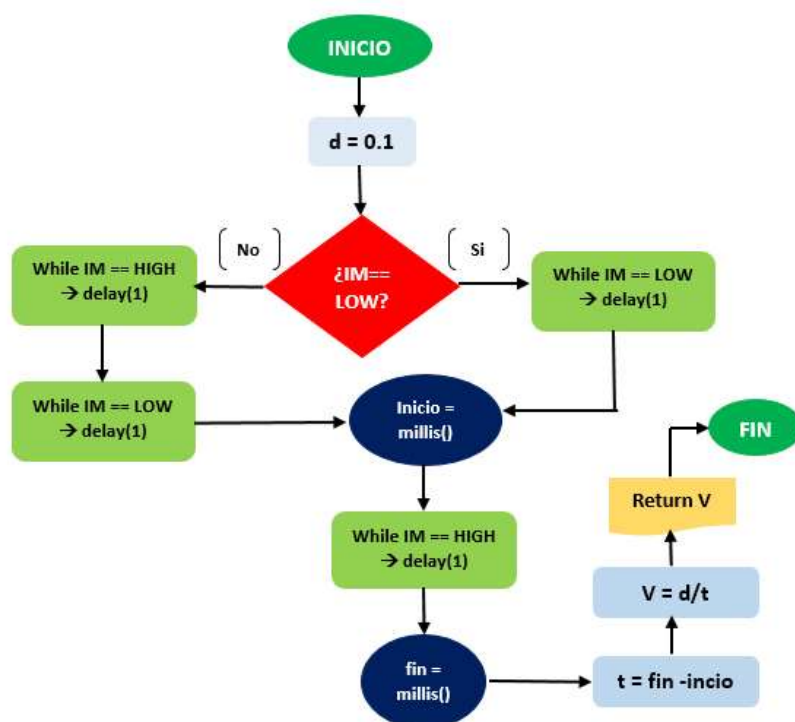


Figura 2.4 Diagrama de flujo de la función “Velocidad”

En el diagrama de la Figura 2.4, primeramente se define la longitud dada por el campo magnético del imán, cuyo cálculo ya fue descrito en la sección 2.2.1. Dicha longitud tiene el valor de 0.1 metros. Continuando con el proceso, en el siguiente punto se trata de identificar si el imán se encuentra o no frente al interruptor magnético (IM) en el instante de tiempo en que se llama a la función, es decir, saber si el interruptor está cerrado o abierto (5v o 0v respectivamente).

Si el interruptor en ese instante está abierto (IM=LOW) simplemente se espera (con el ciclo **while** que precede a la sentencia **if**) el cambio de estado (IM=HIGH) y en ese mismo instante el programa registra el tiempo inicial con una función predefinida por Arduino, llamada `millis()`. La función `millis()` es un cronometro de tiempo en milisegundos que los procesadores Atmel ejecutan implícitamente cuando se arranca el sistema. Una vez obtenido el valor de la función `millis()`, este es almacenado o guardado en la variable *Inicio*, posteriormente, cuando se detecta de nuevo un cambio de estado (IM=LOW) se registra el tiempo final en la variable *Fin*, de esta manera se tiene un tiempo final y un tiempo inicial y el tiempo transcurrido será la diferencia de la variable *Fin* menos la variable *Inicio* [15]. El caso opuesto a este es cuando el primer estado detectado es HIGH, aquí el programa tiene que esperar a que cambie a LOW para que el conteo de tiempo sea registrado correctamente, y a partir de aquí repite el mismo proceso que el caso anterior. Por último se realiza la operación correspondiente para el cálculo de la velocidad, se guarda el resultado en la variable *V* y se entrega el resultado.

2.4.3 Diagrama de flujo de las funciones “cadencia1” y “cadencia2”

Para la selección de la marcha, es necesario conocer en que condición de cadencia se encuentra el sistema. Debido a que los cambios de marcha son diferentes en cada una de las funciones de cadencia seleccionada, se hace necesario el tener una función para cada cadencia.

La Figura 2.5 muestra el diagrama de flujo para la función de “cadencia1”.

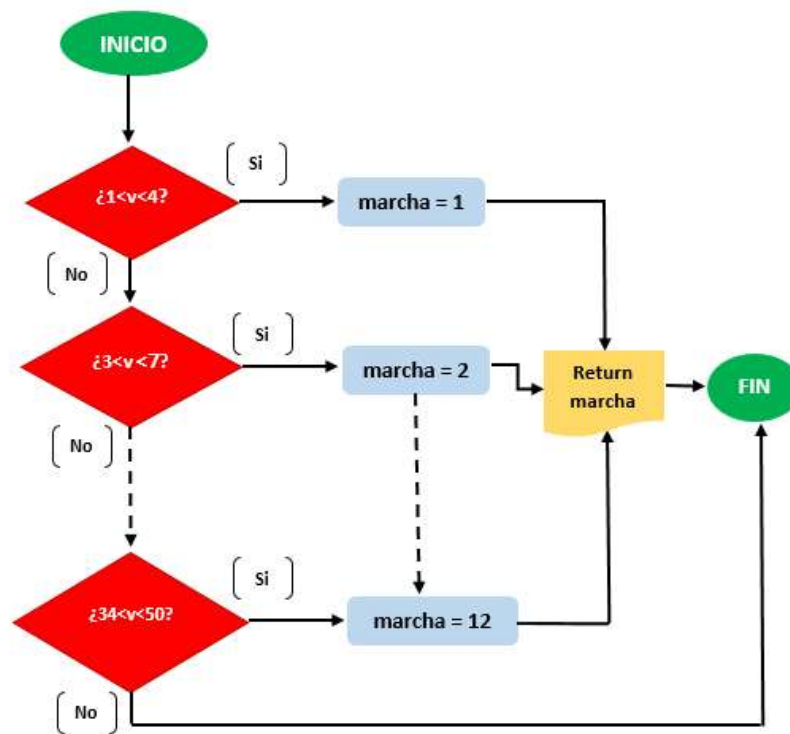


Figura 2.5 Diagrama de flujo de la función “cadencia1”

Observando el diagrama de la Figura 2.5, se puede notar que las condiciones de marcha son seleccionadas por medio de condiciones (ciclos **if**) en el intervalo de velocidad que le corresponde, dependiendo del cálculo previamente realizado.

Una vez obtenido del valor de la marcha correcta, la función regresa dicho valor en la variable denotada como *marcha*.

El diagrama de flujo de la función “cadencia2” (ver Figura 2.6), es exactamente igual al de la función “cadencia1”, con la diferencia de que los intervalos considerados dentro de los ciclos de selección presentan un rango diferente. Esto con el objetivo de manejar una cadencia distinta.

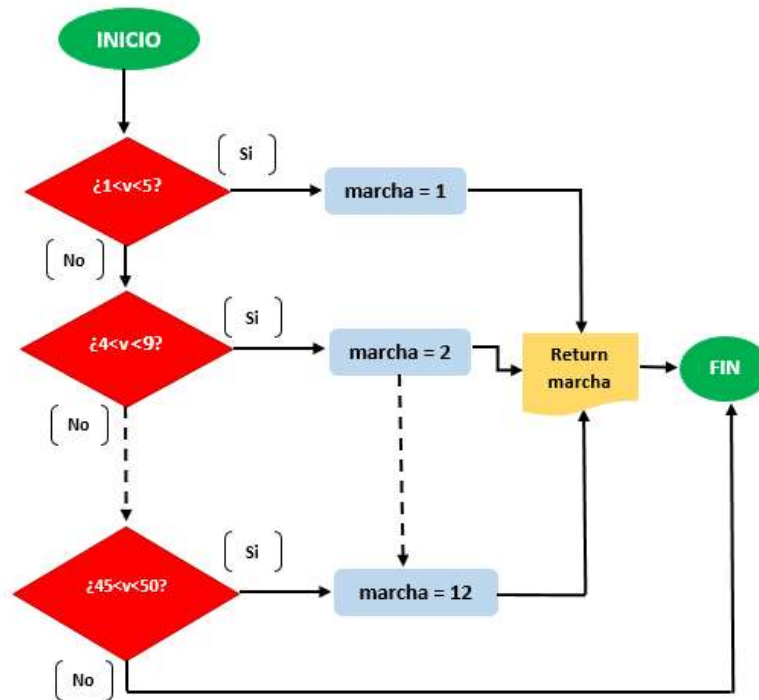


Figura 2.6 Diagrama de flujo de la función “cadencia2”

2.4.4 Diagrama de flujo de las funciones “subirMarcha” y “bajarMarcha”

Los diagramas de flujo para las funciones de “subirMarcha” y “bajarMarcha”, mostrados en la Figura 2.7 y 2.8, respectivamente, básicamente tienen una función muy similar.

El diagrama de subir marcha comienza con la identificación del valor de la marcha, en caso de ser igual a 8 representa que el sistema hizo la transición de la marcha 7 a la 8 (Tablas 2.6 y 2.5) y esto corresponde a un caso “especial”, debido a que es el único cambio de marcha en el cual el sistema realiza 3 cambios en lugar de uno (cambia el plato 24 al plato 34 y el piñón 16 primero cambia al piñón 18 y después al 21). En caso de que la variable marcha no sea igual a 8, el sistema hace simplemente un cambio de piñón (se cambia al piñón más pequeño).

Los delays o retardos mostrados en el diagrama, tienen la finalidad de darle o proporcionarle tiempo a la respuesta de los servos, al sistema de transmisión y al

ciclista, y poder ejecutar el cambio para evitar un posible traslape entre un cambio y otro.

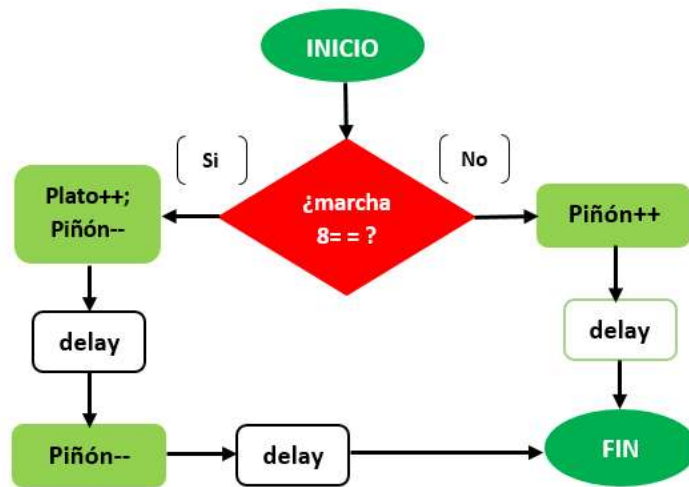


Figura 2.7 Diagrama de flujo de la función "subirMarcha"

Por otra parte, para la función "bajarMarcha" (Figura 2.8) se tiene un funcionamiento muy similar a la función anterior, con la diferencia de que el valor de la marcha correspondiente al caso especial sucede cuando se tiene la marcha 7. Esto significa que el sentido de los cambios en la transmisión es opuesto a la de "subirMarcha".

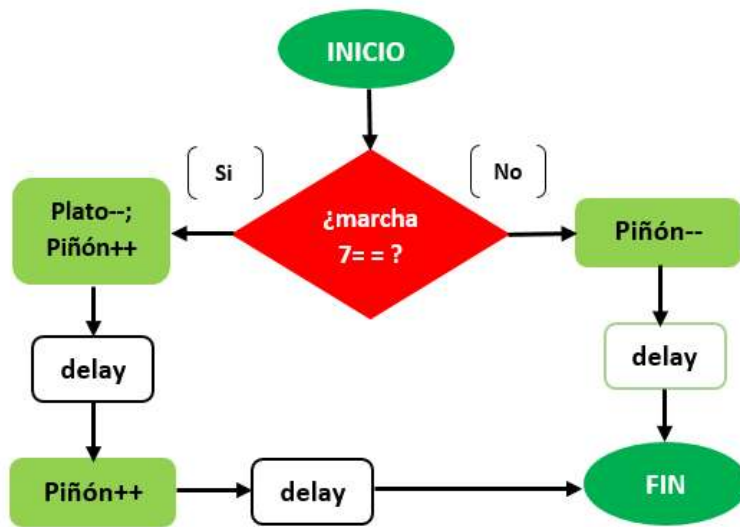


Figura 2.8 Diagrama de flujo de la función “bajarMarcha”

2.5 Presentación del algoritmo

En esta sección se presenta el código implementado en la plataforma Arduino, teniendo en consideración los diagramas de flujo de la sección anterior.

2.5.1 Justificación del código

En esta etapa se deben de considerar tres aspectos muy importantes para que el sistema funcione correctamente.

El primero es la velocidad máxima de la bicicleta en un ambiente urbano el cual es de 35 km por hora (Este dato fue proporcionado por ciclistas que usan dispositivos GPS en su móvil).

El segundo aspecto es el modelo del sistema de cambios que se utilizara (El más usado o el más comercial). En este caso se utilizara un sistema de cambios común en bicicletas comerciales el cual cuenta con 2 platos y 9 piñones, por lo tanto se cuenta con un sistema de 18 velocidades.

Y por último el diámetro de las llantas ya que en función de este parámetro se podrá conocer la medida de pulgadas de engranaje de cada marcha. El diámetro de las ruedas de la bicicleta es de 27 pulgadas (68.5 cm) a presión nominal.

El diseño del sistema se realiza, tomando en cuenta los datos anteriores.

Para el cálculo de la velocidad se tiene:

$$\begin{aligned} \text{Velocidad Maxima} &= 35 \frac{KM}{H} \\ &= \frac{35 km}{1 H} * \frac{35000m}{1 km} * \frac{1 H}{3600 s} = 9.72 \frac{m}{s} \end{aligned}$$

$$\text{Diametro de las llantas} = 0.685m$$

$$\text{Perimetro de la llanta} = \pi * D$$

$$= 3.14 * 0.685m$$

$$= 2.15 m$$

Esto significa que idealmente la bicicleta recorre una distancia de 2.15 metros con cada giro completo de las ruedas.

En seguida se calculara el tiempo que tardan las ruedas en dar una vuelta completa cuando se tiene la velocidad máxima.

$$\begin{aligned} \text{Tiempo de Vuelta} &= \frac{\text{Distancia}}{\text{Velocidad}} \\ &= \frac{2.041 m}{8.33 m/s} = 0.245 S \end{aligned}$$

$$= 245 \text{ mS}$$

El tiempo de respuesta de un interruptor magnético no es mayor a 100ms, por lo tanto no se presenta ningún problema con el uso del interruptor seleccionado. Adicionalmente, también se sabe que en el lenguaje de Arduino:

1 segundo = Delay (1000)

1 milisegundo = Delay (1)

2.5.2 Código

En esta subsección, es presentado el código implementado en el Arduino, el cual está basado en los algoritmos presentados anteriormente.

Como primera parte del código, se declaran de las diferentes variables y constantes globales del sketch. La Figura 2.9 la parte de código implementada para la declaración de las variables y constantes utilizadas.

```

#include <Servo.h>           //libreria de arduino

Servo servoPinon;
Servo servoPlato;

//Variables
int marchaN=1;              //Condiciones iniciales
int marchaA=1;
double v;
int evento = 1;             //Variable usada para el conteo de eventos

//constantes
int sensor=6;
int botonCadencia=12;
double rangoSensor=0.075;   //Longitud del campo magnetico del iman dado en metros
int subirPinon = 180;        //Mueve el servo para cambiar a un piñon mas grande
int bajarPinon = 20;         //Mueve el servo para cambiar a un piñon mas pequeño
int mid = 90;                //Coloca los servo en una posicion neutral
int subirPlato = 0;          //Mueve el servo para cambiar a un plato mas grande
int bajarPlato =140;         //Mueve el servo para cambiar a un plato mas pequeño
int tiempoCambio=1000;
int tiempoInicial;

```

Figura 2.9 Declaración de variables y constantes

La función `setup()`, donde son configurados los pines de entrada y salida, así como la comunicación con otros dispositivos es configurada con el código que se muestra en la Figura 2.10.

```

void setup(){
  Serial.begin(9600);        //Inicio de la comunicacion serial
  servoPinon.attach(7);      //Se asigna la salida digital 7 al servo de los piñones
  servoPlato.attach(8);      //Se asigna la salida digital 8 al servo de los platos
  pinMode(sensor, INPUT);    //Se configura el pin digital 6 como entrada para el interruptor
  pinMode(botonCadencia, INPUT); //Se configura el pin digital 12 como entrada
}

```

Figura 2.10 Configuración de la función Setup

Se puede apreciar, que en la función `setup()`, son utilizados los pines 7 y 8 como salidas para enviar los comandos a los servos que controlan los piñones y platos, respectivamente. Además, los pines 6 y 12 son configurados como entradas, y se encargan de monitorear el estado de la cadencia seleccionada, así como el sensor de velocidad. La función principal “Control del sistema” (*loop*) queda configurada de la forma en que se presenta en la Figura 2.11 y 2.12.

```

void loop(){
  Serial.print("Evento #");Serial.println(evento); //Registro de eventos
  v=Velocidad(sensor);                               //Calculo de la velocidad
  //-----Cadencia 1-----
  if(digitalRead(botonCadencia)==LOW){

    Serial.println("Cadencia 1"); //Variables enviadas a un servidor para
    Serial.print("Velocidad =");Serial.println(v); //su el registro de pruebas
    marchaN=cadencia1(v);
    Serial.print("marcha Anterior =");Serial.println(marchaA);
    Serial.print("marcha Nueva =");Serial.println(marchaN);

    if(marchaN>marchaA){
      int i = marchaA+1;
      for(int c=i;c<=marchaN;c++){ //Este for hace el mismo numero de
        subirMarcha(c);           //cambios que el contador de marchas
        delay(tiempoCambio); }    //Tiempo necesario para que los servos actuen
      marchaA=marchaN;           //Se actualiza el valor de la marcha anterior
    }
    else if(marchaN<marchaA){
      int i = marchaA-1;
      for(int c=i;c>=marchaN;c--){ //Lo mismo que el ciclo for anterior
        subirMarcha(c);
        delay(tiempoCambio); }
      marchaA=marchaN;           //Se actualiza el valor de la marcha anterior
    }
  }
  //-----

```

Figura 2.11 Función Control del sistema parte 1

La Figura 2.11 muestra la primera parte del código, en la cual se puede apreciar primeramente la declaración de la habilitación para la escritura o registro de los eventos mediante la comunicación serial, y en la segunda línea de código, la función del cálculo de la velocidad mediante la función “velocidad” la cual es mostrada en la Figura 2.12.

La función velocidad() mostrada en la Figura 2.12 (véase sección 2.4.2) calcula la velocidad de la bicicleta y guarda la información en una variable local llamada V la cual es retornada para ser asignada a la variable global v. Seguido de dicha declaración, el ciclo que conforma la cadencia 1 es presentado. Dicho ciclo es habilitado con un estado bajo del interruptor de selección de la cadencia.


```

////////////////////////////////////
//Funcion Velocidad - Regresa el valor de la velocidad calculada
////////////////////////////////////
double Velocidad(int interruptor){
//Constantes y variables
double makm=3.6;           //Constante para convertir metros a kilometros
double vo;
double tiempo;
//Primer caso, cuando el interruptor no se encuentra frente al iman
if(digitalRead(interruptor)==LOW){ //Si el sensor no detecta el campo
    while(digitalRead(interruptor)==LOW){ //magenito espera hasta que lo detecta
        delay(50); }
//Tiempo
double inicio=millis();           //Cuando detecta el iman empieza
while(digitalRead(interruptor)==HIGH){ // a contar el tiempo en ms
    delay(1); }
double fin=millis();
tiempo=(fin-inicio);
tiempo /= 1000.0;                 //Calcula el tiempo en ms
//Tiempo en Segundos
vo= rangoSensor/tiempo;
vo *= makm;                       //Conversion de m/s a km/h
int V=vo;
return V;
}
// En caso de que la funcion entre a la mitad del campo magnetico
else {
    while(digitalRead(interruptor)==HIGH){ //Espera a que pase el iman
        delay(1); }
    while(digitalRead(interruptor)==LOW){ //Espera a que detecte el iman
        delay(50); }
//Tiempo
double inicio=millis();           //Cuando detecta el iman empieza
while(digitalRead(interruptor)==HIGH){ // a contar el tiempo en ms
    delay(1); }
double fin=millis();
tiempo=(fin-inicio);
tiempo /= 1000.0;                 //Calcula el tiempo en ms
//Tiempo en Segundos
vo= rangoSensor/tiempo;
vo *= makm;                       //Conversion de m/s a km/h
int V=vo;
return V;
}
}
}

```

Figura 2.12 Función Velocidad

Para la cadencia 1, se puede apreciar de manera general, que primeramente las variables son almacenadas, para posteriormente entrar en los ciclos del control de la marcha, en el cual se detecta si la marcha tiene que aumentar o disminuir. Para el ciclo anterior, es necesario introducir los retardos ("tiempoCambio") para esperar a que los servomotores actúen.

```

//-----Cadencia 2-----
else if(digitalRead(botonCadencia)==HIGH){

    Serial.println("Cadencia 2");           //Variables enviadas a un servidor para
    Serial.print("Velocidad =");Serial.println(v); //su el registro de pruebas
    marchaN=cadencia2(v);
    Serial.print("marcha Anterior =");Serial.println(marchaA);
    Serial.print("marcha Nueva =");Serial.println(marchaN);

    if(marchaN>marchaA){
        int i = marchaA+1;
        for(int c=i;c<=marchaN;c++){        //Este for hace el mismo numero de
            subirMarcha(c);                  //cambios que el contador de marchas
            delay(tiempoCambio); }           //Tiempo necesario para que los servos actuen
        marchaA=marchaN;                    //Se actualiza el valor de la marcha anterior
    }
    else if(marchaN<marchaA){
        int i = marchaA-1;
        for(int c=i;c>=marchaN;c--){        //Lo mismo que el ciclo for anterior
            subirMarcha(c);
            delay(tiempoCambio); }
        marchaA=marchaN;                    //Se actualiza el valor de la marcha anterior
    }
}
//-----
evento++;
int tiempoEvento=millis(); //Se registra y envia a un servidor el tiempo
tiempoEvento /= 1000;      //transcurrido cada vuelta de ciclo
Serial.print("tiempo=");Serial.println(tiempoEvento);
}

```

Figura 2.13 Función Control del sistema parte 2

La Figura 2.13 muestra la continuación del código de la función de control de sistema, en la cual se puede apreciar, que básicamente consiste del ciclo de control de cadencia 2. La función de cadencia 2, es básicamente la misma que la de la cadencia 1.

Adicionalmente, se tiene unas líneas de código con las cuales es detectado o medido el tiempo transcurrido en cada vuelta de ciclo.

Como cada una de las cadencias presenta un nivel de marcha distinto, dos funciones son declaradas. La Figura 2.14 muestra la función de selección de marcha para la cadencia 1.

```

////////////////////
// Funcion "cadencia1"
////////////////////

double cadencia1(double vN){
    if((vN>=1)&&(vN<4)){           //Marcha 1
        marchaN = 1; }
    else if((vN>3)&&(vN<7)){       //Marcha 2
        marchaN = 2; }
    else if((vN>6)&&(vN<10)){      //Marcha 3
        marchaN = 3; }
    else if((vN>9)&&(vN<13)){      //Marcha 4
        marchaN = 4; }
    else if((vN>12)&&(vN<16)){     //Marcha 5
        marchaN = 5; }
    else if((vN>15)&&(vN<19)){     //Marcha 6
        marchaN = 6; }
    else if((vN>18)&&(vN<22)){     //Marcha 7
        marchaN = 7; }
    else if((vN>21)&&(vN<25)){     //Marcha 8
        marchaN = 8; }
    else if((vN>24)&&(vN<28)){     //Marcha 9
        marchaN = 9; }
    else if((vN>27)&&(vN<31)){     //Marcha 10
        marchaN = 10;}
    else if((vN>30)&&(vN<34)){     //Marcha 11
        marchaN = 11;}
    else if((vN>33)&&(vN<50)){     //Marcha 12
        marchaN = 12;}
    return marchaN;
}

```

Figura 2.14 Función cadencia 1

Como se explicó anteriormente, esta función únicamente toma en cuenta el valor de la velocidad en la cual se encuentra para realizar la selección de la marcha adecuada.

La Figura 2.15 muestra el código de la selección de marcha para cuando se selecciona la cadencia 2.

```

////////////////////
// Funcion "cadencia2"
////////////////////

double cadencia2(double vN){
    if((vN>=1)&&(vN<5)){           //Marcha 1
        marchaN = 1; }
    else if((vN>4)&&(vN<9)){       //Marcha 2
        marchaN = 2; }
    else if((vN>8)&&(vN<13)){      //Marcha 3
        marchaN = 3; }
    else if((vN>12)&&(vN<17)){     //Marcha 4
        marchaN = 4; }
    else if((vN>16)&&(vN<21)){     //Marcha 5
        marchaN = 5; }
    else if((vN>20)&&(vN<25)){     //Marcha 6
        marchaN = 6; }
    else if((vN>24)&&(vN<29)){     //Marcha 7
        marchaN = 7; }
    else if((vN>28)&&(vN<33)){     //Marcha 8
        marchaN = 8; }
    else if((vN>32)&&(vN<37)){     //Marcha 9
        marchaN = 9; }
    else if((vN>36)&&(vN<41)){     //Marcha 10
        marchaN = 10; }
    else if((vN>40)&&(vN<45)){     //Marcha 11
        marchaN = 11; }
    else if((vN>44)&&(vN<50)){     //Marcha 12
        marchaN = 12; }
    return marchaN;
}

```

Figura 2.15 Función cadencia 2

Finalmente, se presenta el código de la implementación de las funciones de “subirMarcha” y “bajarMarcha” encargadas de enviar la señal apropiada a los servomotores para aumentar o disminuir la marcha del sistema de transmisión.

La Figura 2.16 muestra el código referente a la función “subirMarcha”.


```

////////////////////
//Funcion "subirMarcha"
////////////////////

void subirMarcha(int nMarcha){
    if (nMarcha==8){          //Esta condicion es para cambiar de la marcha 7 a la 8

        servoPlato.write(subirPlato);    //Se hace un cambio en los platos y un cambio
        servoPinon.write(bajarPinon);    //el primer cambio en los piñones
        delay(2000);                     //Los delays de 2s y 0.5s son para dar tiempo
                                         //de actuar a los servos
        servoPlato.write(mid);            //Se regresa el mando a su posicion neutral
        servoPinon.write(mid);
        delay(500);

        servoPinon.write(bajarPinon);    //Segundo cambio en los piñones
        delay(2000);
        servoPinon.write(mid);
        delay(500);

    }
    else{
        servoPinon.write(subirPinon);    //Si la marcha es diferente de 7 se
        delay(2000);                     //sube una posicion en los piñones
        servoPinon.write(mid);
        delay(500);
    }
}
}

```

Figura 2.16 Función SubirMarcha

Como se explicó en la sección 2.2.4, la marcha 8 (caso especial) tiene que ser detectada, para posteriormente realizar el cambio de los platos y piñones con su respectivo retardo. Cada vez que se realiza un cambio, el código regresa el mando a su posición neutral. Esto último se realiza, debido a que el sistema de cambios de la bicicleta funciona de dicha manera, es decir, que para realizar un cambio únicamente se realiza un desplazamiento de la palanca, hacia abajo o arriba (dependiendo si se aumenta o disminuye una marcha), para posteriormente regresar la palanca a su posición original.

El código de la función “bajarMarcha” (Figura 2.17), es similar, simplemente se tiene que tomar en cuenta la condición de bajar la marcha con la marcha 7.

```

////////////////////////////////////
//Funcion "bajarMarcha"
////////////////////////////////////

void bajarMarcha(int nMarcha){
  if (nMarcha==7){ //Esta condiciones es para cambiar de la marcha 8 a la 7

    servoPlato.write(bajarPlato); //Se hace un cambio en los platos y un cambio
    servoPinon.write(subirPinon); //el primer cambio en los piñones
    delay(2000);

    servoPlato.write(mid); //Se regresa el mando a su posicion neutral
    servoPinon.write(mid);
    delay(500);

    servoPinon.write(subirPinon); //Segundo cambio en los piñones
    delay(2000); //Los delays de 2s y 0.5s son para dar tiempo
    servoPinon.write(mid); //de actuar a los servos
    delay(500);

  }
  else{
    servoPinon.write(bajarPinon); //Si la marcha es diferente de 7 se
    delay(2000); //baja una posicion en los piñones
    servoPinon.write(mid);
    delay(500);
  }
}
}

```

Figura 2.17 Función BajarMarcha

Una vez expuesto el código implementado en el Arduino, se procede a describir cada uno de los componentes necesarios para la implementación del prototipo, esto será descrito en el siguiente capítulo.

Capítulo 3. Diseño del circuito general

El objetivo de este capítulo es de agrupar todos los elementos del sistema y visualizar todos los detalles técnicos que pueden alterar su funcionamiento.

3.1 Justificación del Circuito

En el primer capítulo se explicó el funcionamiento de los tres subsistemas que componen el sistema, los cuales aunados al sistema de procesamiento descrito en el capítulo 2 conforman el sistema completo del sistema de cambios automáticos. El objetivo de este capítulo es el de integrar todos estos subsistemas para conformar el sistema completo.

La Tabla 3.1, muestra un resumen de los diferentes subsistemas utilizados y el elemento asociado (Hardware) a cada uno.

Tabla 3.1 Elementos del circuito general

No.	Subsistema	Elemento	Cantidad
1	Sistema eléctrico	Servomotor Tower pro mg995	2
2	Sistema eléctrico	Batería de celular	1
3	Sistema electrónico	Interruptor magnético	1
4	Sistema eléctrico	Dinamo	1
5	Sistema electrónico	Regulador de voltaje	1
6	Sistema de cambios	Mandos	2
7	Sistema de procesamiento	Placa Arduino nano	1

El circuito general deberá de contar con estos nueve elementos y tendrá que cumplir con los requerimientos necesarios, primeramente el consumo total de potencia para poder rendir de forma apropiada.

3.2 Capacidad de la batería

La capacidad de la batería debe ser superior al total de la potencia consumida por todos los componentes del circuito juntos (considerando el consumo máximo), debido a que esta deberá suministrar la energía suficiente al sistema para el caso de que no se cuente con un dispositivo generador (puede ser por ausencia de generador o por falla del mismo) como lo es el dínamo.

Cuando ya se tienen definidos todos los componentes principales es sencillo calcular la capacidad de la batería, la forma ideal es considerar la potencia máxima que puede consumir cada elemento para después hacer la suma de todos los componentes y finalmente agregar una tolerancia como medida de prevención.

El primer elemento a considerar son los servomotores, debido a que pueden llegar a requerir una corriente de hasta 800mA dependiendo de la carga a la que se someta, y opera con un voltaje nominal de 5V, por lo tanto ambos servos pueden llegar a consumir una potencia de 8W. En este proyecto se utilizan dos servomotores, uno para controlar la palanca de piñones y otro para controlar la palanca de los platos. Por otro lado el interruptor opera a 5V y puede requerir una corriente de 50mA.

Por su parte la placa Arduino trabajara a 5V, consume una corriente de hasta 50mA por pin de salida, por lo tanto dependerá de la tarea que realice la placa. En este caso se tienen dos salidas utilizadas para mandar la señal de control a los servos, por lo tanto se necesitan 100mA, a eso le agregan 50mA de la placa Arduino (debido al regulador de la placa principalmente), entonces la corriente máxima total requerida es de 150mA y su potencia consumida puede llegar a 0.75W.

Tomando en cuenta estos valores se tiene que el sistema completo trabajara con 5V, requerirá una corriente total de 1.8 A, y consumirá una potencia de 5W. Entonces la batería requerida para este sistema podrá ser de 5V y que proporcione una corriente mayor a 1A.

En la Tabla 3.2 se pueden observar los resultados calculados.

Tabla 3.2 Capacidad de la batería

Elemento	V nominal	I _{max}	P _{max}
Servomotores	5 v	1.6 A	8W
Interruptor	5 v	50 mA	0.25 W
Placa Arduino	5 v	150 mA	0.75 W
Total	5 v	1.8 Amp	9 W

Al total de la corriente máxima se agrega una tolerancia del 10% como medida de prevención y finalmente la capacidad de la batería queda aproximadamente de 2A, esto quiere decir que cualquier batería que proporcione 2 o más amperes será útil para este proyecto. En este proyecto se utiliza una batería con capacidad de 2.5A, con salidas de voltaje de 5V por lo tanto su potencia máxima es de 12.5 W. Esta batería cumple bien con los requisitos necesarios del sistema y tiene un pequeño margen para posibles sobrecargas.

La Figura 3.1 muestra la batería real utilizada en el prototipo.

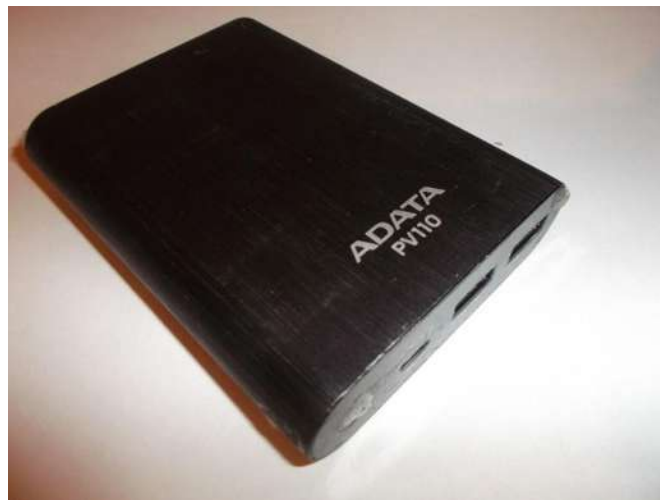


Figura 3.1 Batería utilizada en la implementación

Para el montaje de la batería en la bicicleta, se fabrica una pequeña caja de metal, la cual es colocada debajo del asiento del conductor. Dicha caja, permite

adicionalmente colocar otros componentes. La Figura 3.2 muestra la imagen de la caja donde es colocada la batería.



Figura 3.2 Montaje de la caja para la ubicación de la batería

3.3 Descripción de los elementos

En esta sección se abordan los diferentes elementos (Hardware) necesarios para la implementación del prototipo.

3.3.1 Servomotor

Un servomotor es un motor de corriente continua que tiene la capacidad de ser controlado en posición. Es capaz de ubicarse en cualquier posición dentro de un rango de operación y mantenerse estable en dicha posición.

Los servos disponen de tres cables: dos cables de alimentación (Figura 3.3) que suministran un voltaje 4.8-6V y un cable de control que indica la posición deseada al circuito de control mediante señales PWM (“Pulse Width Modulation”).

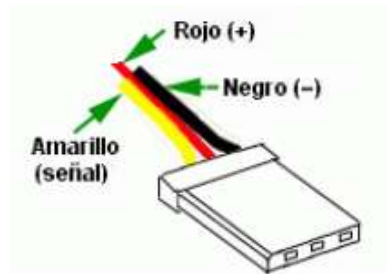


Figura 3.3 Cables de alimentación y Control [14]

Las señales PWM utilizadas para controlar los servos están formadas por pulsos positivos cuya duración es proporcional a la posición deseada del servo y que se repiten cada 20 ms (50Hz). El servomotor Tower Pro MG995 tiene un rango de movimiento de 180 grados que puede controlarse con periodos que van de 0.5 a 2.5ms como se muestra en la Figura 3.4. [14]

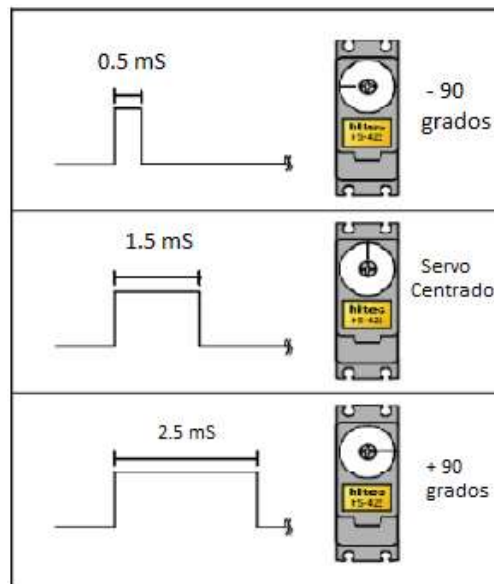


Figura 3.4 Control del Servomotor [14]

Según las recomendaciones de Arduino la conexión del servo con cualquier modelo de placa de Arduino debe de ser la siguiente (Figura 3.5).

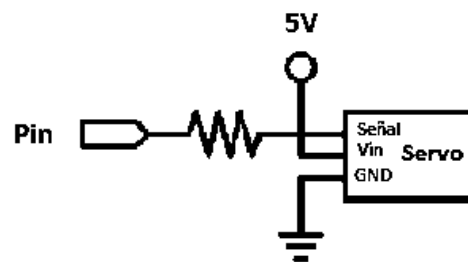


Figura 3.5 Conexión del Servomotor con el Arduino [14]

La implementación de los servomotores en el sistema es realizada directamente en los mandos de cambios con los que cuenta originalmente la bicicleta.

Como se expuso en la descripción del código para la función de “subirMarcha”, dichos mandos realizan el cambio de posición con un movimiento de la palanca hacia abajo o hacia arriba de su posición original, siempre regresando a la posición intermedia entre las dos condiciones. Debido a este funcionamiento, se aprovecha esta condición, donde el servomotor únicamente tendrá que hacer giros de una posición neutral, la cual es colocada a los 90° , hacia una de 0° (por ejemplo bajar palanca) o una de 180° (por ejemplo subir palanca). La Figura 3.6 muestra el montaje de un servomotor en el mando de cambios.



Figura 3.6 Montaje del servomotor en un mando de cambios

3.3.2 Interruptor Magnético

Los interruptores magnéticos también conocidos como “reed”, son dispositivos conformados por dos patillas de metal colocadas a una distancia muy pequeñas entre sí (menor a 1 mm), ambas son colocadas dentro de un encapsulado de cristal (ver Figura 3.7).



Figura 3.7 Interruptor magnético

Su funcionamiento es muy simple, al estar cerca de un campo magnético sus patillas metálicas se unen cerrando el circuito.

Existen de varios tamaños y de distintas capacidades eléctricas, su velocidad de respuesta es buena y tienen más tolerancia a la temperatura, pero con la desventaja de tener una estructura de cristal que los hace vulnerables a los golpes, por tal motivo se protegen generalmente con un porta fusibles de plástico.

El sensor es colocado en la rueda delantera, específicamente en la horquilla. Permitiendo detectar un cambio o pulso, cuando el imán permanente de neodimio pasa por enfrente del sensor.

La Figura 3.8 muestra la colocación de dicho sensor.



Figura 3.8 Montaje del interruptor magnético

Si bien, algún otro sistema comercial pudo haberse utilizado para medir la velocidad, se optó por realizarlo de esta manera, debido a su facilidad de implementación y costos.

3.3.3 Interruptor de cadencia

El interruptor o switch es utilizado para controlar el paso de corriente. Generalmente consta de 2 patillas, de las cuales una es fija y la otra es móvil para permitir el control manual del flujo de corriente. El interruptor utilizado es mostrado en la Figura 3.9.



Figura 3.9 Interruptor para la selección de la cadencia

El montaje en la bicicleta, es mostrado en la Figura 3.10.



Figura 3.10 Montaje del interruptor de selección de cadencia

3.3.4 Placa Arduino Leonardo

La tecnología Arduino está basada en microcontroladores Atmel, en este caso la placa Leonardo cuenta con un Atmega32u4. Esta versión tiene 20 pines de entra/salida digitales de los cuales se pueden configurar 7 como salidas PWM y 12 como entradas analógicas. Además, cuenta con un oscilador de cristal de 16 Mhz. [15]

Se recomienda alimentar la placa con 5V. En sus pines de entrada es recomendable trabajar en un rango de 5 a 12V y la corriente máxima con la que puede trabajar, tanto en sus pines de entrada como de salida, es de 50mA. Adicionalmente, cuenta con una memoria flash de 32 Kb, una SRAM de 2.5 Kb y una memoria EEPROM de 1 kb. Puede ser alimentado mediante USB o por una fuente de alimentación externa.

La Figura 3.11 muestra la placa Arduino utilizada.



Figura 3.11 Placa Arduino Leonardo

La placa Arduino es colocada dentro de la caja mostrada en la Figura 3.2.

3.3.5 Dínamo

El Dínamo es un dispositivo generador que convierte la energía mecánica en energía eléctrica, por medio de la rotación de conductores a través de un campo magnético producido por un imán permanente.

La utilización de dicho dispositivo en el proyecto, tiene como objetivo el recargar la batería aprovechando la energía mecánica de la bicicleta, y así prolonga considerablemente el rendimiento de la batería.

El dínamo instalado en la bicicleta (ver Figura 3.12) proporciona un voltaje de 12V y una corriente continua hasta 1.2A, el cual es suficiente para retribuir al menos un 50% del gasto de energía.



Figura 3.12 Dinamo

En la Figura 3.13, es mostrado el montaje del dínamo en la bicicleta.



Figura 3.13 Colocación del dínamo en la bicicleta

3.3.6 Regulador de voltaje

Debido a que el dínamo proporciona 12V, y la placa Arduino requiere solamente 5V, se hace necesario la adición de un circuito regulador de voltaje.

Dentro del tipo de reguladores que pueden ser utilizados se encuentran los reguladores de tipo lineal y los conmutados. El regulador de tipo lineal como lo puede ser el Lm7805, no proporciona la corriente requerida para la sustentabilidad del sistema. Por tal motivo, se hace necesario el uso de un regulador conmutado como lo es el Lm2596t-5 (ver hoja de datos en Anexo A.2), el cual presenta un armado muy similar al de un regulador lineal con la diferencia de que este incluye después del circuito integrado una configuración de convertidor tipo buck que permite incrementar la corriente mediante una bobina y un diodo volante.

El diseño del regulador conmutado al igual que el regulador lineal es un circuito bien definido, sin embargo su costo puede llegar a ser hasta 6 veces mayor que el de un regulador lineal.

El circuito regulador conmutado que se utiliza se presenta en la Figura 3.14.

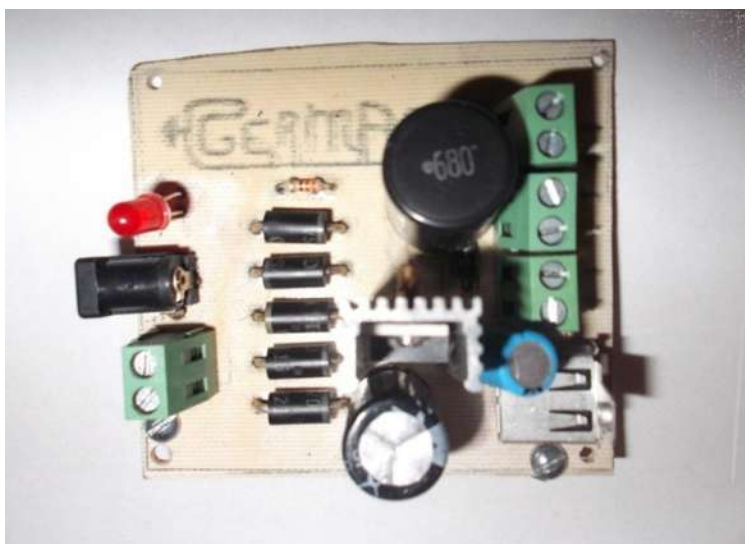


Figura 3.14 Regulador conmutado de 5V

3.3.7 Circuito General

Este esquema fue elaborado con la ayuda del Software Fritzing que es un programa diseñado para elaborar circuitos impresos con Tarjetas Arduino. La Figura 3.15

muestra el circuito de las conexiones físicas realizadas con los dispositivos descritos en las secciones anteriores.

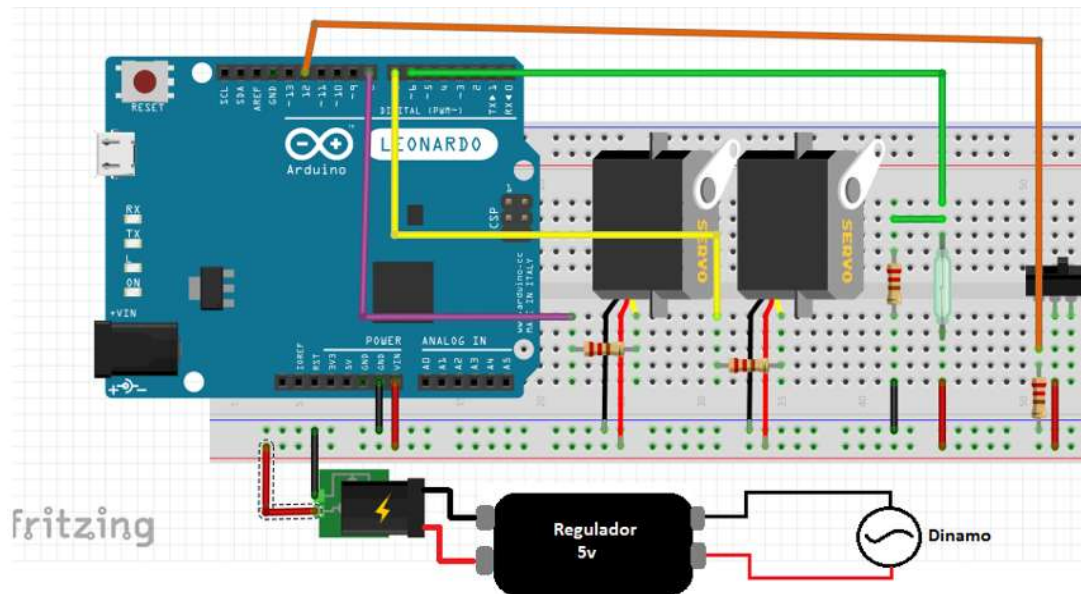


Figura 3.15 Circuito general físico

Adicionalmente, el mismo software genera un diagrama de diseño esquemático (ver Figura 3.16).

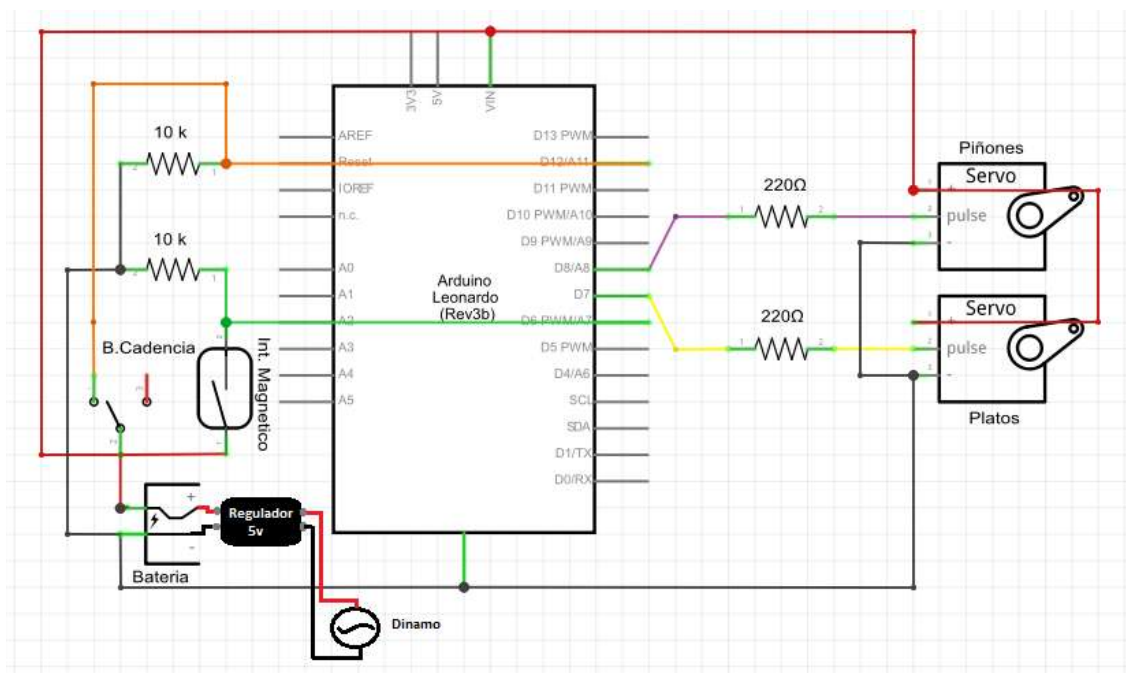


Figura 3.16 Circuito general esquemático

Tanto en el circuito general como en el esquemático, los elementos adicionales necesarios para el correcto funcionamiento del sistema son presentados, tales como las resistencias utilizadas entre el Arduino y los servomotores, así como también las del interruptor de selección de la cadencia.

Finalmente la Figura 3.17 muestra el sistema completo, en el cual se resaltan los diferentes elementos presentados anteriormente. La ubicación de los diferentes elementos se seleccionó de tal manera que no se afectara al ciclista y al diseño mismo de la bicicleta, con el fin de no entorpecer el funcionamiento y la comodidad que se tenía antes de la implementación del sistema.

De la Figura 3.17, los componentes resaltados corresponde a:

1. Servomotores acoplados a los mandos del sistema de transmisión
2. Caja de soporte para el Arduino, el regulador de voltaje y la batería
3. Interruptor magnético
4. Imán permanente de neodimio
5. Dínamo
6. Interruptor de selección de cadencia
7. Módulo de conectores para comunicar los servomotores con la placa Arduino.



Figura 3.17 Montaje de los dispositivos en la bicicleta

En el siguiente capítulo son presentadas las pruebas del sistema propuesto.

Capítulo 4. Implementación y pruebas

El objetivo de este capítulo es poner a prueba el Sistema bajo diversas circunstancias, para analizar su comportamiento y estimar su efectividad con los 2 modos de cadencia seleccionables.

4.1 Introducción

Una vez descrito todas las partes del sistema, así como también la integración de estas, es necesario realizar las pruebas que permitan conocer las ventajas y desventajas que posee el sistema de selección de cambios automáticos para la bicicleta. Debido a esto, seis pruebas diferentes del sistema de control de cambios de velocidades son realizadas, con el objetivo de mostrar su rendimiento bajo diferentes condiciones.

Debido a que el sistema tiene que ser probado bajo condiciones de uso, los resultados obtenidos son tomados de las variables de velocidad, de tiempo transcurrido entre cada cambio de estado (cambio de una velocidad a otra), la cadencia utilizada o modo del sistema, y el cambio en las marchas.

Para realizar la obtención de las variables mencionadas anteriormente, se tienen las líneas de comando descritas en el capítulo 2 donde fue descrito el código del programa.

La obtención de los datos es realiza por medio del puerto USB de comunicación entre la computadora y la placa Arduino. Los datos que son registrados se almacenan en un archivo de texto, los cuales posteriormente son graficado, con el objetivo de mostrar los resultados.

Es importante mencionar que todas las pruebas son desarrolladas en campo, es decir sobre un terreno real y bajo condiciones de conducción común de la bicicleta.

Las seis pruebas utilizadas para comprobar el comportamiento del sistema son las siguientes:

1. Cadencia 1, velocidad en aumento constante, sobre terreno regular.
2. Cadencia 1, velocidad aumento-disminución, sobre terreno regular.
3. Cadencia 2, velocidad en aumento constante, sobre terreno regular.
4. Cadencia 2, velocidad aumento-disminución, sobre terreno regular.
5. Combinación cadencia 1 – cadencia 2, sobre terreno regular.
6. Combinación cadencia 1 – cadencia 2, sobre terreno irregular.

Cabe mencionar que las pruebas de la 1 a la 5, se realizan sobre el mismo recorrido, en un terreno regular (refiérase como terreno regular a una superficie plana que no presenta pendientes prolongadas mayores a los 10°), mientras que la prueba 6 es realizada sobre un terreno irregular (terreno irregular es referido a una superficie que presenta pendientes prolongadas mayores a los 10°).

En las siguientes subsecciones son presentados los resultados de cada una de las pruebas realizadas.

4.2 Prueba 1, Cadencia 1 – velocidad aumento – terreno regular.

La prueba 1 consiste en seleccionar la cadencia 1 como modo de marcha y mantener una velocidad constante en aumento sobre un terreno regular.

Los datos obtenidos son mostrados en la Tabla 4.1.

Tabla 4.1 Datos obtenidos para la prueba 1

Evento	Cadencia	Velocidad (Km/h)	Marcha Anterior	Marcha Nueva	Tiempo transcurrido (Seg)
1	1	0	1	1	0
2	1	2	1	1	2
3	1	3	1	1	3
4	1	7	1	3	7
5	1	13	3	5	12
6	1	6	5	2	21
7	1	8	2	2	22
8	1	287	2	2	22
9	1	13	2	5	29
10	1	11	5	4	32
11	1	15	4	5	36
12	1	287	5	5	36
13	1	22	5	8	47
14	1	23	8	8	47
15	1	28	8	10	52
16	1	95	10	10	52
17	1	22	10	8	58
18	1	9	8	3	76
19	1	5	3	2	81
20	1	1	2	1	84

De la Tabla 4.1, se puede observar que se registran 20 eventos. Todos los eventos son para el modo de Cadencia 1. Se puede observar la velocidad registrada por el sensor en cada uno de los eventos registrados.

De la Tabla 4.1, se puede observar que el evento 7 y 8 presentan una diferencia muy considerable en cuanto a la velocidad, de hecho una velocidad de 287 Km/ es imposible de lograr con una bicicleta.

El dato o evento 8 es un dato erróneo, debido a que como se puede observar, se presenta en el mismo segundo que el evento 7 (22 segundos), esto es debido a que su registro es menor de 4ms. Adicionalmente, no se registra un cambio en los datos de marcha anterior y marcha nueva, debido a que se trata de un “bug” o error.

El caso de la lectura errónea en la velocidad se presenta también en el evento 12. Ambos eventos no son tomados en cuenta para realizar la gráfica de los datos.

La Figura 4.1 muestra la gráfica de los datos de la Tabla 4.1, graficados contra el tiempo.

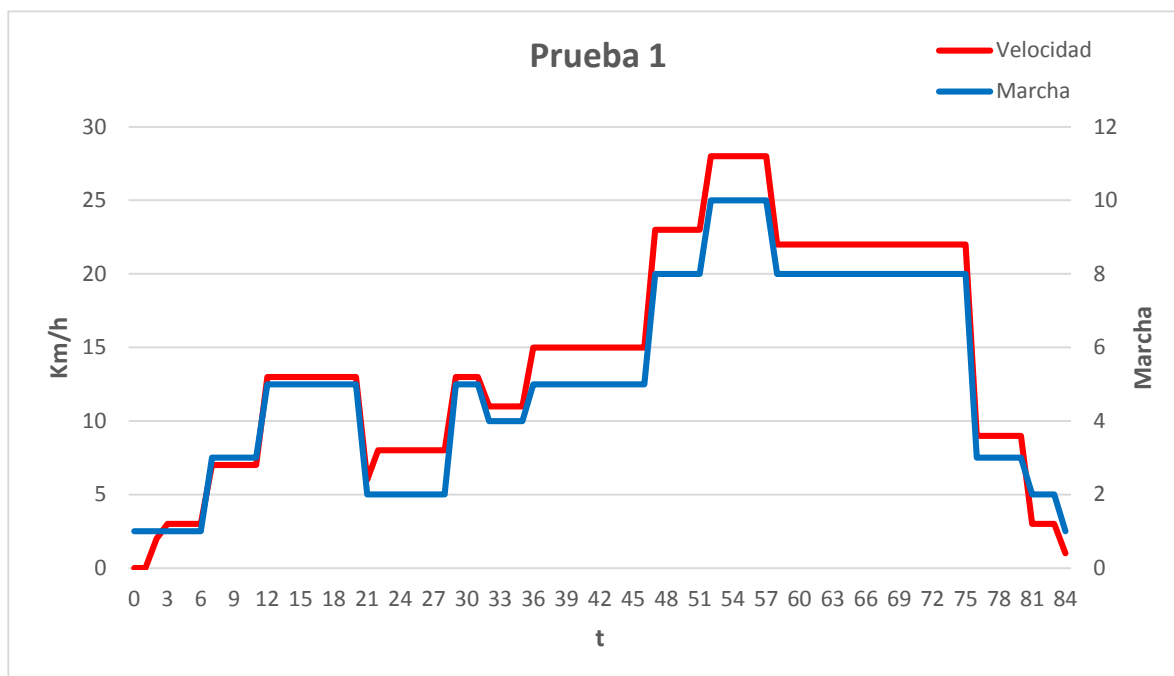


Figura 4.1 Respuesta de la Cadencia 1 sobre un terreno regular y velocidad en aumento

En la Figura 4.1 se muestra el desenvolvimiento del sistema sobre un terreno regular, donde se alcanza una velocidad máxima de 28 km/h en 53 segundos y la marcha más alta que se utiliza es la 11, en el instante de 53 segundos aproximadamente. El incremento en la cadencia es irregular mientras que su descenso es regular y notablemente más breve, además se efectuaron 12 cambios en la marcha en un lapso de 84 segundos para concluir con un promedio de 1 cambio de marcha cada 7 segundos.

4.3 Prueba 2, Cadencia 1 – velocidad aumento-disminución – terreno regular.

Para la prueba 2, se toman en cuenta las mismas condiciones que para la prueba anterior, solamente que el cambio de la velocidad se realiza de una manera ascendente y descendente, en el transcurso de la prueba.

La Tabla 4.2 muestra el valor de los datos adquiridos. Dichos datos, como se puede apreciar, corresponden a la cadencia 1 y además es visible que el valor de la velocidad y marcha (tanto nueva como anterior) presentan una variación que va de un valor menor a mayor y viceversa.

Es importante remarcar que los datos de la Tabla 4.2, así como de las subsecuentes pruebas, son tomados del algoritmo donde se reajustaron los parámetros de las funciones cadencia1 y cadencia2, para evitar registrar dicho “bug”. Específicamente se reajustaron los valores de la sentencia **if** para la asignación de la marcha 12, donde se estableció un límite de velocidad de 50 km/h para así evitar las mediciones erróneas.

Gracias al reajuste de las funciones cadencia1 y cadencia2, los datos no son registrados, debido a que se verifica si el cambio de velocidad excede considerablemente al dato anterior, con lo cual no se guarda dicho dato. Lo mismo sucede, cuando no se registra un cambio de la marcha nueva con la anterior.

Tabla 4.2 Datos obtenidos para la prueba 2

Evento	Cadencia	Velocidad (km/h)	Marcha Anterior	Marcha Nueva	Tiempo transcurrido (Seg)
1	1	0	1	1	0
2	1	4	1	2	4
3	1	12	2	4	9
4	1	24	4	8	21
5	1	30	8	10	30
6	1	32	10	11	33
7	1	30	11	10	36
8	1	19	10	7	45
9	1	13	7	5	52
10	1	12	5	4	55
11	1	4	4	2	61
12	1	1	2	1	66

Para tener una mejor interpretación de los datos obtenidos para la prueba 2, la gráfica de la Figura 4.2 es presentada.

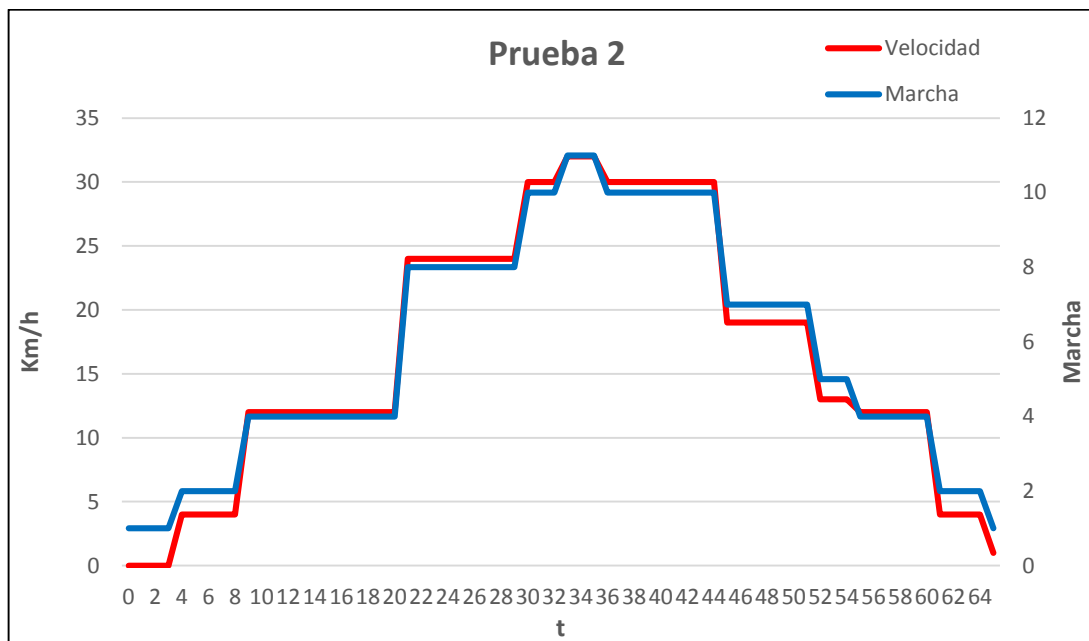


Figura 4.2 Respuesta de la Cadencia 1 sobre un terreno regular y velocidad en aumento y disminución

En el caso de la Figura 4.2 se presenta un incremento de velocidad sin caídas (como lo sucedido en la prueba 1 en los segundos 21 y 32), debido a que el pedaleo fue constante con un ritmo de aproximadamente 0.5 revoluciones por segundo.

La velocidad máxima alcanzada es de 32 km/h, y se logró dicha velocidad al utilizar la marcha 11. Por último, se registraron 11 cambios en 65 segundos teniendo un promedio de 1 cambio de marcha cada 6 segundos.

4.4 Prueba 3, Cadencia 2 – velocidad aumento – terreno regular.

Para esta prueba, el interruptor de selección de cadencia es cambiado, es decir, que ahora el sistema se encuentra funcionando con la cadencia 2 en vez de la cadencia 1.

La prueba se realiza de la misma manera que la prueba 1, se experimenta sobre terreno regular y se elige un aumento de velocidad constante.

La Tabla 4.3 muestra los valores obtenidos, y la Figura 4.3 su respectiva gráfica.

Tabla 4.3 Datos obtenidos para la prueba 3

Evento	Cadencia	Velocidad (Km/h)	Marcha Anterior	Marcha Nueva	Tiempo transcurrido (Seg)
1	2	0	1	1	0
2	2	4	1	2	5
3	2	7	2	2	7
4	2	8	2	2	8
5	2	8	2	2	9
6	2	11	2	3	12
7	2	17	3	5	19
8	2	19	5	5	20
9	2	11	5	3	28
10	2	6	3	2	32
11	2	9	2	3	36
12	2	7	3	2	39
13	2	10	2	3	43
14	2	7	3	2	48
15	2	8	2	2	49
16	2	7	2	2	50
17	2	8	2	2	51
18	2	2	2	1	55

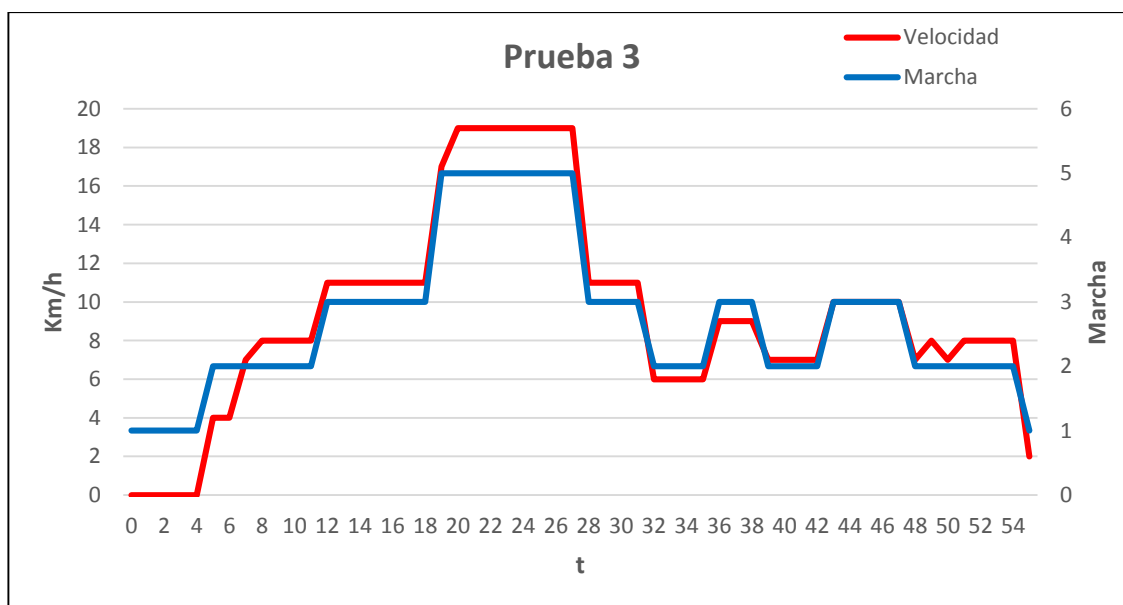


Figura 4.3 Respuesta de la Cadencia 2 sobre un terreno regular y velocidad en aumento

En la Figura 4.3 se observa un constante cambio de la aceleración y a la desaceleración como lo sería muy comúnmente en un ambiente urbano. La velocidad máxima alcanzada es de 19 km/h y por último se registraron 10 cambios en un tiempo de 55 segundos para terminar con un promedio de 1 cambio en el sistema cada 5.5 segundos.

4.5 Prueba 4, Cadencia 2 – velocidad aumento-disminución – terreno regular.

Los resultados de la selección de una cadencia 2, con una velocidad en aumento constante sobre un terreno regular, son presentados en esta sección.

Al igual que las pruebas anteriores, los datos adquiridos del ensayo son presentados en la Tabla 4.4.

Tabla 4.4 Datos obtenidos para la prueba 4

Evento	Cadencia	Velocidad (Km/h)	Marcha Anterior	Marcha Nueva	Tiempo transcurrido (Seg)
1	2	0	1	1	0
2	2	3	1	1	5
3	2	5	1	2	8
4	2	10	2	3	12
5	2	10	3	3	13
6	2	13	3	4	16
7	2	17	4	5	20
8	2	18	5	5	21
9	2	20	5	5	22
10	2	22	5	6	25
11	2	14	6	4	33
12	2	10	4	3	37
13	2	16	3	4	41
14	2	13	4	4	42
15	2	10	4	3	46
16	2	9	3	3	47
17	2	2	3	1	54

Para esta prueba, se puede apreciar en los datos, que durante el intervalo de 25 a 33 segundos no se tiene ningún cambio en la velocidad, es decir, esta se mantiene constante. Así mismo, posterior a este instante, se presenta una baja

considerable en la velocidad y cambio de marcha, debido al frenado que se realizó con la bicicleta. Lo anterior, es apreciado de una mejor manera en al grafica de la Figura 4.4.

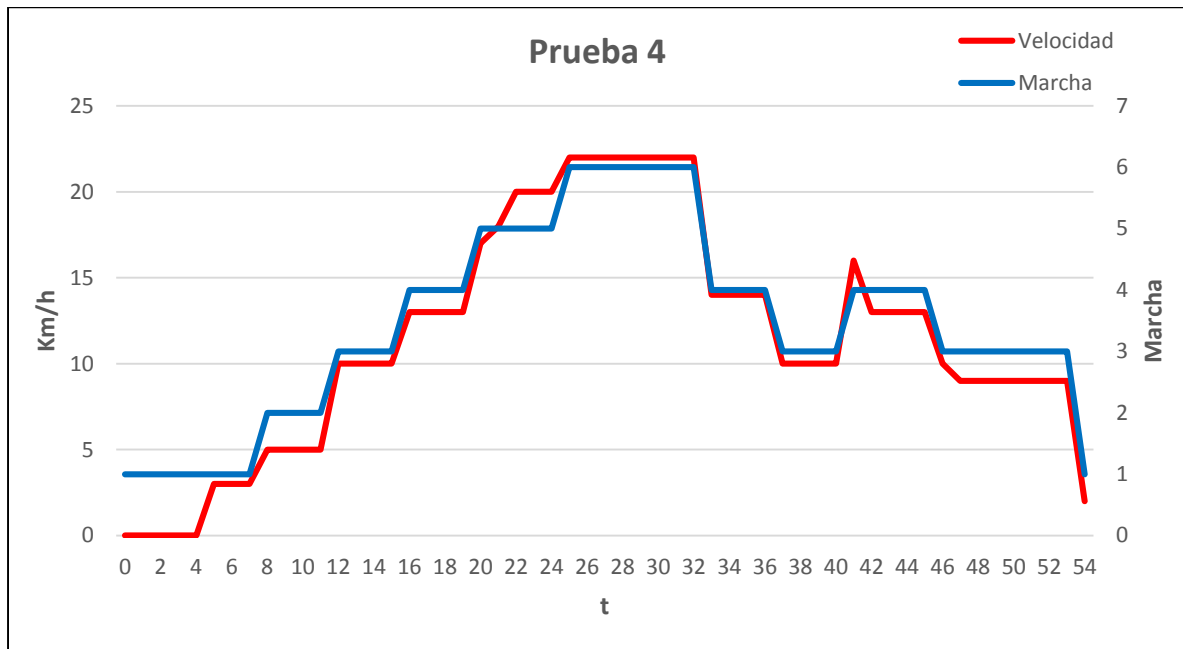


Figura 4.4 Respuesta de la Cadencia 2 sobre un terreno regular y velocidad en aumento y disminución

Para la prueba 4, se registró una velocidad máxima de 22 km/h y se ejecutaron 10 cambios de marcha en un tiempo de 54 segundos, para concluir con promedio de 1 cambio de marcha cada 5.4 segundos.

4.6 Prueba 5, Cadencia 1 y 2 – terreno regular.

Una vez realizadas las pruebas con una selección de la cadencia de manera fija, los resultados de un cambio de cadencia sobre la prueba, se hace evidente.

Es por tal motivo que en esta prueba se realiza un cambio en la selección de la cadencia, durante un instante en el transcurso de la prueba.

La Tabla 4.5 muestra los valores de la cadencia, marcha anterior y nueva y velocidad durante el instante de tiempo que ocurren.

Tabla 4.5 Datos obtenidos para la prueba 5

Evento	Cadencia	Velocidad (Km/h)	Marcha Anterior	Marcha Nueva	Tiempo transcurrido (Seg)
1	1	0	1	1	0
2	1	8	1	3	5
3	1	8	3	3	6
4	1	9	3	3	7
5	1	13	3	5	15
6	1	22	5	8	25
7	1	22	8	8	26
8	1	21	8	7	30
9	1	18	7	6	33
10	1	14	6	5	36
11	2	11	5	4	40
12	2	12	4	4	42
13	2	14	4	5	45
14	2	15	5	5	46
15	2	17	5	6	50
16	2	12	6	3	60
17	2	8	3	2	63
18	2	9	2	3	67
20	2	4	3	1	74

La grafica en la Figura 4.5 y los datos de la Tabla 4.5, muestran el uso combinado de las dos cadencias. La primera parte y hasta el segundo 39 corresponde al uso de la cadencia 1, mientras que la segunda parte (a partir del segundo 40) corresponde a la cadencia 2.

En el momento de la transición (en el segundo 40) se observa una pequeña disminución de la velocidad debido a que el sistema reduce la marcha, lo cual tiene el efecto de perder tracción en los pedales y por consecuencia la bicicleta no recibe el mismo impulso antes del cambio de cadencia.

Adicionalmente, se observa que la velocidad máxima de la cadencia 1 es de 22 km/h, mientras que para la cadencia 2 es de 17 km/h.

Se registraron 13 cambios de marcha en 74 segundos para obtener un promedio de 1 cambio de marcha cada 5.69 segundos.

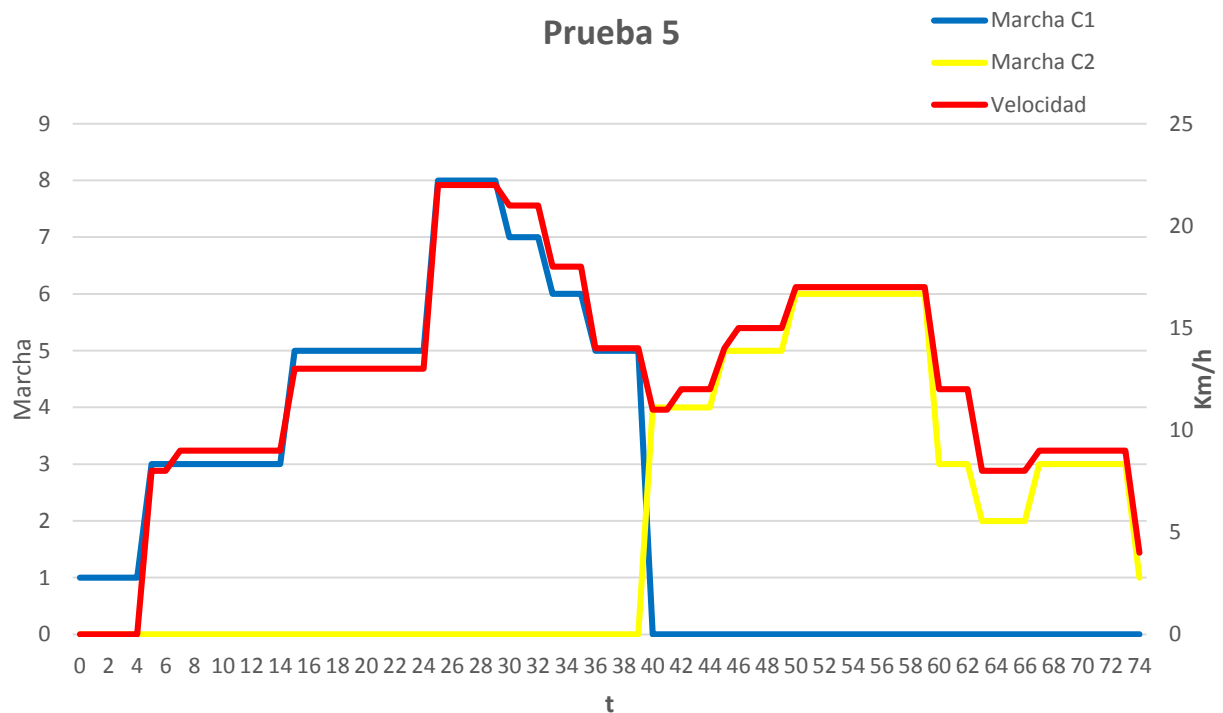


Figura 4.5 Respuesta de la Cadencia 1 y 2 sobre un terreno regular

4.7 Prueba 6, Cadencia 1 y 2 – terreno irregular.

En esta prueba, se toma en cuenta las mismas condiciones que para la prueba anterior, solamente que ahora se elige un terreno irregular.

La Figura 4.6 muestra el diagrama de la forma del terreno irregular. Para dicho terreno irregular son tomadas en cuenta las pendientes mayores a 10° de inclinación.

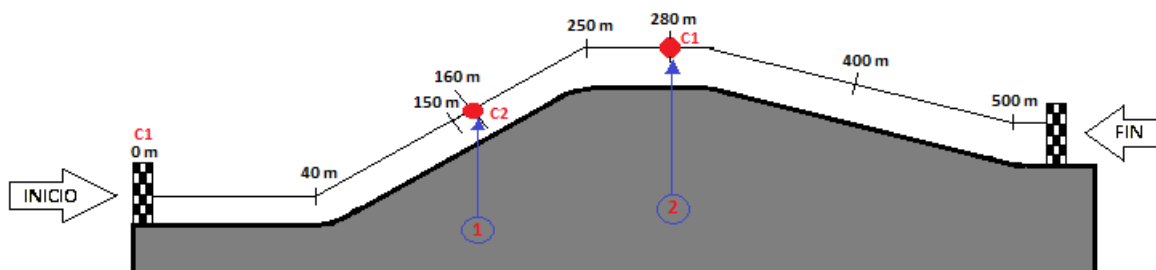


Figura 4.6 Diagrama del terreno irregular

La distancia recorrida total es de 500 metros y en los puntos 1 y 2 (indicados en la gráfica), se realiza el cambio de la cadencia seleccionada.

La Tabla 4.6 y la Figura 4.6 muestran los resultados obtenidos.

Tabla 4.6 Datos obtenidos para la prueba 6

Evento	Cadencia	Velocidad (Km/h)	Marcha Anterior	Marcha Nueva	Tiempo transcurrido (Seg)
1	1	0	1	1	0
2	1	2	1	1	1
3	1	3	1	1	2
4	1	7	1	3	9
5	1	10	3	4	12
6	1	13	4	5	15
7	1	23	5	8	25
8	2	23	8	6	32
9	2	13	6	4	39
10	2	12	4	3	42
11	2	16	3	4	45
12	1	20	4	8	60
13	1	22	8	8	60
14	1	28	8	10	66
15	1	22	10	8	74
16	1	9	8	3	89
17	1	5	3	2	93

La prueba es iniciada con la cadencia 1 seleccionada, y se realiza la primera transición a la cadencia 2 a los 160 metros aproximadamente (en el punto de referencia 1 mostrado en la Figura 4.6) para hacer notar la ventaja de ésta, en la cual se adopta

una marcha más suave que brinda un mayor confort en la subida y permite aplicar menos fuerza al pedaleo a cambio de un sacrificio de velocidad.

La cadencia 2 es seleccionada a los 280 metros aproximadamente (en el punto de referencia 2 de la Figura 4.6). Debido a que la cadencia 1 es más productiva en terrenos planos y pendientes cuesta abajo, el resultado en la velocidad es notable debido a que es aquí, donde se alcanza la velocidad máxima de la prueba.

Al final, solo se disminuye la velocidad y termina la prueba. La Figura 4.7 muestra la gráfica de los resultados.

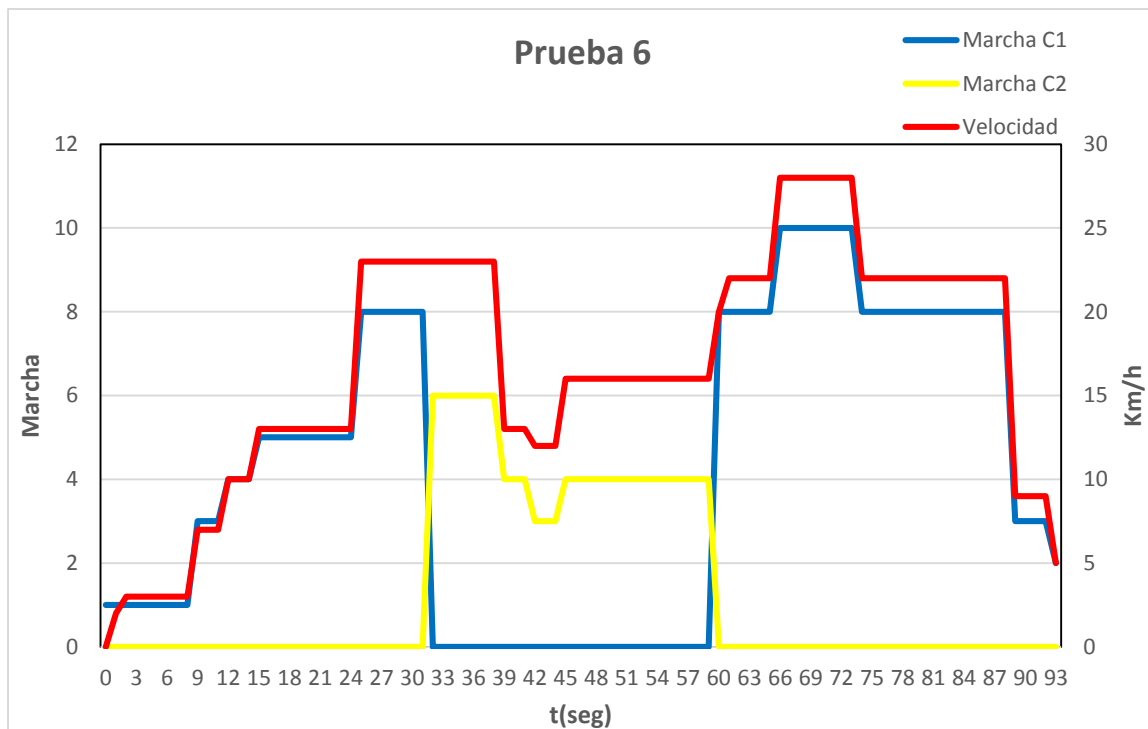


Figura 4.7 Respuesta de la Cadencia 1 y 2 sobre un terreno irregular

De los resultados, se tiene que la velocidad máxima de la prueba es de 28 km/h, hubo 2 transiciones de cadencia y se contaron 14 cambios de marcha en 93 segundos para finalmente concluir con un promedio de 1 cambio de marcha cada 6.64 segundos.

Capítulo 5. Conclusiones y trabajos futuros

El objetivo de este capítulo es proporcionar las conclusiones y trabajos futuros que pueden ser realizados para el mejoramiento del sistema propuesto en el presente trabajo.

5.1 Conclusiones

Las conclusiones que pueden ser derivadas del presente trabajo de tesis, son descritas a continuación:

- Se logró implementar un sistema automático en una bicicleta, el cual permite obtener un mejor aprovechamiento de su sistema de transmisión. Para lo anterior se utilizaron diferentes componentes electrónicos y herramientas básicas de programación.
- Se desarrollaron diferentes funciones utilizando la plataforma Arduino. Dichas funciones permiten obtener un esquema de programación, en el cual son tomadas en cuenta la simplicidad del código, aunado a una optimización del uso de los recursos disponibles en el sistema, tanto de componentes como de ejecución del algoritmo.
- Con el desarrollo del presente prototipo el sistema de transmisión de una bicicleta es usado de una manera eficiente, sin necesidad de que un usuario tenga conocimientos previos del buen uso de dicho sistema.
- Las condiciones de uso del sistema se encuentran enfocadas en un ambiente o uso urbano, sin embargo este puede ser fácilmente adaptado a otro tipo de ambiente, por ejemplo en el uso profesional.
- La ventaja del sistema, es que se trata de un sistema que puede ser montado en cualquier tipo de bicicleta que cuente con un sistema de transmisión. Para realizar esto, se hace necesario ajustar las tablas de pulgada de engranaje, así como las

funciones “subirMarcha” y “bajarMarcha”, para cuando el número de platos y piñones sea diferente al utilizado en el presente proyecto.

- El sistema tiene una pérdida de transferencia de energía muy pequeña a la hora de realizar una transición de marcha, el tiempo promedio por transición a partir de que se detecta un cambio de velocidad es de aproximadamente 3 segundos.
- En este proyecto se demostró que se puede ejecutar una acción de control de manera correcta y precisa con el número mínimo de variables a censar.
- El hecho de utilizar una sola variable para conocer el estado de la bicicleta tiene un costo bajo y contribuye al buen funcionamiento de la bicicleta en terrenos regulares. Sin embargo, cuando el terreno es irregular no se obtienen los datos necesarios para conocer el estado exacto de la bicicleta y las condiciones bajo las que opera, por lo tanto el sistema de control de cambios creará que se está operando bajo condiciones normales y no ejecutara alguna acción específica para aligerar la cadencia. La solución propuesta en la tesis es la implementación de un botón para el cambio de cadencia, lo cual permite al usuario hacer más suave la cadencia en caso de ser necesario.

5.2 Trabajos futuros

Como trabajos futuros se visualizan las siguientes mejoras al sistema:

1.- Al sistema se le puede adicionar un sistema de ubicación remota (GPS), el cual permita realizar el registro de rutas tomadas durante el uso de la bicicleta. Adicionalmente, una vez contando con el GPS, es evidente la adición de una pequeña pantalla que permita realizar el trazado de rutas, previas al recorrido.

2.- Si bien, el sistema presentado en el presente trabajo está enfocado en utilizar el menor número de variables a medir, para lograr una optimización del código. Se pueden adicionar algunas otras que permitan una mejora en el prototipo.

3.- Un sistema de medición de ritmo cardiaco puede ser adicionado. Dicho sistema se colocaría en los mandos de la bicicleta para medir la actividad cardiaca.

4.- La implementación de un acelerómetro puede funcionar como variable del sistema y además como medida de auxilio para el ciclista. Medir la inclinación de la bicicleta proporciona al sistema información sobre el estado de la bicicleta pero si a esto se le agrega un sistema GPS se puede crear un sistema de auxilio el cual pueda tener la capacidad de valorar si la bicicleta ha sufrido un percance o no, dependiendo de la posición y del tiempo que dure la bicicleta en ese estado.

Anexo A

A.1. Medidas de seguridad Shimano

A.2. Hoja de datos Lm2596t-5

A.1. Medidas de seguridad Shimano

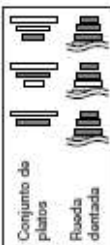
Información general de seguridad

ADVERTENCIA

- Tenga cuidado que su ropa no sea atrapada por la cadena cuando monte la bicicleta, de lo contrario se podría caer de la misma.
- Verifique que la tensión de la cadena sea correcta y que la cadena no está dañada. Si la tensión no es suficiente o la cadena está dañada, deberá cambiar la cadena. De lo contrario, la cadena se puede romper y se podría caer de la bicicleta.
- Los dos pernos de montaje de la biela izquierda deben ser apretados alternadamente de a poco en vez de ser apretados completamente de a uno. Use una llave de torque para verificar que los pernos de apriete finales se encuentran en el rango de 12 - 15 N.m.
- Además, después de andar aproximadamente 100 km (60 millas), use una llave de torque para volver a verificar los pernos de apriete.
- También es importante verificar periódicamente los pares de apriete. Si los pares de apriete son demasiado débiles o si los pernos de montaje no son apretados adecuadamente en pasos, la biela izquierda se podría salir y se podría caer de la bicicleta.
- Verifique que no haya rajaduras en los brazos de la biela antes de montar la bicicleta. Si existen rajaduras, el brazo de la biela se podría romper y se podría caer de la bicicleta.
- Océngala y lea cuidadosamente las instrucciones al instalar las partes. Una parte floja, gastada o dañada puede resultar heridas para el ciclista.
- Recomendamos usar sólo repuestos genuinos de Shimano.
- Lea estas instrucciones de servicio técnico cuidadosamente, y manténgala en un lugar seguro para futuras consultas.

Nota

- Asegúrese de que la combinación de platos está de acuerdo con la configuración de dientes del conjunto de platos en el cuadro de especificaciones del producto. Si se usan otras combinaciones, la distancia entre los platos será incorrecta y la cadena se podría caer y quedar atrapada entre ellos.
- Cuando la cadena se encuentra en la posición indicada en la figura, la cadena y el conjunto de platos o el desviador de cambios delantero pueden tocarse y generar ruido. Si el ruido es un problema, cambie la cadena a la rueda dentada más grande siguiente o la siguiente a esta.
- Si la envoltura del pedalier no están paralelas, el rendimiento de los cambios no será bueno.
- Verifique que no haya holgura en las juntas o conexiones antes de montar la bicicleta. (BB-FC, FC-PD)
- Si siente que el eje del juego pedalier está flojo, deberá cambiar el juego pedalier.
- Además, si el rendimiento de pedaleo no parece normal, verifique eso una vez más.
- No lave el juego pedalier con chorros de alta presión de agua.
- Engrasar el juego pedalier antes de instalarlo.
- Para asegurar el mejor rendimiento, asegúrese de usar sólo el tipo de cadena especificado. No se puede usar una cadena ancha.
- Si la cadena se sale de los platos durante el uso, cambie los platos y la cadena.
- Debe lavar periódicamente los platos usando un detergente neutro y luego volver a lubricarlos. Además, limpiando la cadena con detergente neutro y lubricarla puede ser una manera efectiva de extender la vida útil de los platos y la cadena.
- Las piezas no están garantizadas contra el desgaste natural o el deterioro resultante del uso normal.
- Si tiene alguna duda respecto al uso, mantenimiento, ajuste o instalación, por favor consultar a una tienda especializada en bicicletas.



Nota: Las especificaciones pueden cambiar por razones de precio sin previo aviso. (Reservado)

Instrucciones de servicio técnico

SI-1F30H

FC-7800 FC-7803

Conjunto de platos

Especificaciones

Modelo	FC-7800	FC-7803
Combinación de dientes de la rueda de cadena anterior	A-43 / 42T (9-43 / 30T) B-42 / 38T (9-42 / 30T) 54 / 42T (55 / 42T) 56 / 44T	52 - 38 - 30T
Diámetro del círculo del perno	130 mm	130 mm / 92 mm
Largo de biela	169, 167.5, 170, 172.5, 169, 167.5, 170, 172.5, 175, 177.5, 180 mm	169, 167.5, 170, 172.5, 175, 177.5, 180 mm
Roscas de los pedaliers	B.C. 9/16" x 20TPI (rosca inglesa)	
Cadena aplicable	CN-7801	
Línea de la cadena	43.5 mm	45 mm
Ancho del manijunto (Dimensiones de la rosca)	68 mm (1.37 X 24 TPI) 70 mm (M36 X 24 TPI)	

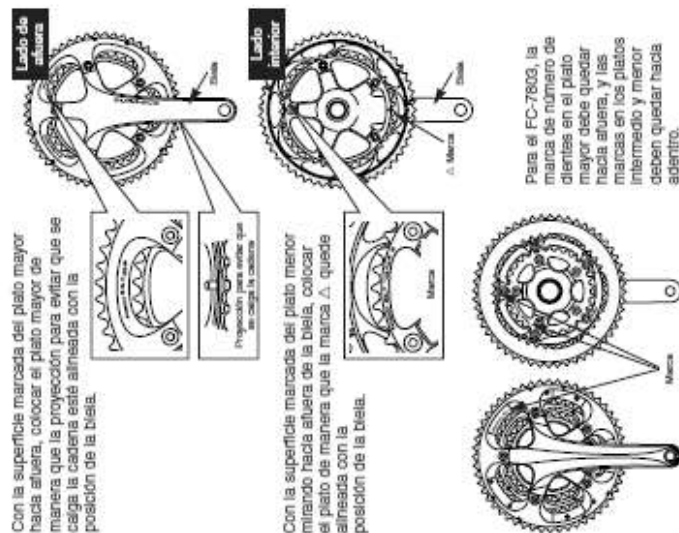
Instalación de los platos

Si los platos no están bien instalados no será posible hacer los cambios con suavidad, asegúrese de instalar los platos en las posiciones correctas.

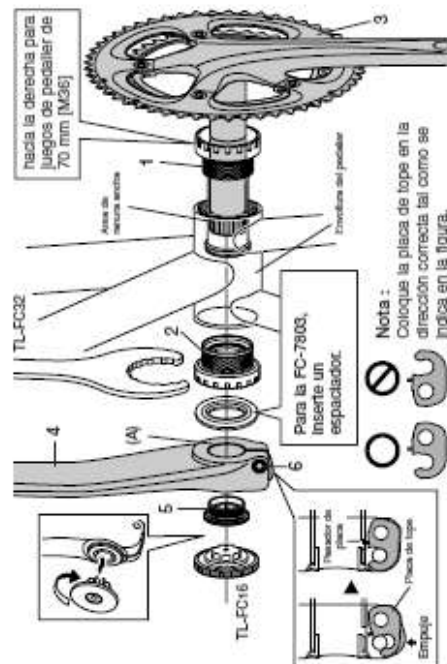
Instalación del conjunto de platos

Instalar usando el procedimiento indicado en la figura.

- 1, 2 Usar la herramienta especial TL-FC32 para instalar la adaptador derecha (rosca hacia la izquierda) y la adaptador izquierda (rosca hacia la derecha).
- Par de apriete: 35 - 50 N.m (350 - 500 kgf-cm)
- 3 Insertar la leva derecha.
- 4 Colocar la sección A de la leva izquierda en el eje de la leva derecha donde la ranura es más ancha. (Inserte un espaciador cuando instale la unidad de biela izquierda FC-7803.)
- 5 Use el TL-FC16 para apretar la tapa.
Par de apriete: 0.7 - 1.5 N.m (7 - 15 kgf-cm)
- 6 Empuje la placa de tope y verifique que el pasador de placa está bien en su lugar, y luego apriete el perno de la biela izquierda.
Nota : Cada uno de los pernos debe ser apretado bien parejo e igual a 12 - 15 N.m (120 - 150 kgf-cm)



Para el FC-7803, la marca de número de dientes en el plato mayor debe quedar hacia afuera, y las marcas en los platos intermedio y menor deben quedar hacia adentro.



Nota : Coloque la placa de tope en la dirección correcta tal como se indica en la figura.

A.2. Hoja de datos Lm2596t-5

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

Check for Samples: [LM2596](#)

FEATURES

- 3.3V, 5V, 12V, and Adjustable Output Versions
- Adjustable Version Output Voltage Range, 1.2V to 37V $\pm 4\%$ Max Over Line and Load Conditions
- Available in TO-220 and TO-263 Packages
- Ensured 3A Output Load Current
- Input Voltage Range Up to 40V
- Requires Only 4 External Components
- Excellent Line and Load Regulation Specifications
- 150 kHz Fixed Frequency Internal Oscillator
- TTL Shutdown Capability
- Low Power Standby Mode, I_Q Typically 80 μ A
- High Efficiency
- Uses Readily Available Standard Inductors
- Thermal Shutdown and Current Limit Protection

APPLICATIONS

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

DESCRIPTION

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

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

The LM2596 series operates at a switching frequency of 150 kHz thus allowing smaller sized filter components than what would be needed with lower frequency switching regulators. Available in a standard 5-lead TO-220 package with several different lead bend options, and a 5-lead TO-263 surface mount package.

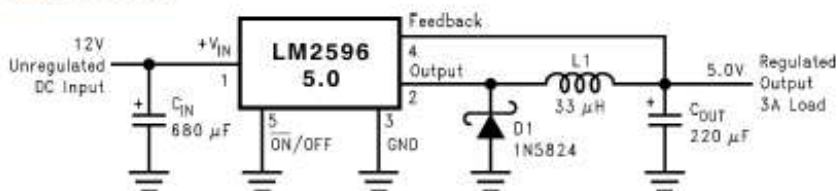
A standard series of inductors are available from several different manufacturers optimized for use with the LM2596 series. This feature greatly simplifies the design of switch-mode power supplies.

Other features include a ensured $\pm 4\%$ tolerance on output voltage under specified input voltage and output load conditions, and $\pm 15\%$ on the oscillator frequency. External shutdown is included, featuring typically 80 μ A standby current. Self protection features include a two stage frequency reducing current limit for the output switch and an over temperature shutdown for complete protection under fault conditions. ⁽¹⁾

(1) $\uparrow$ Patent Number 5,382,918.

Typical Application

(Fixed Output Voltage Versions)



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

SIMPLE SWITCHER is a registered trademark of Texas Instruments. All other trademarks are the property of their respective owners.

PRODUCTION DATA Information is current as of publication date. Products conform to specifications per the terms of the Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1999–2013, Texas Instruments Incorporated

Connection Diagrams

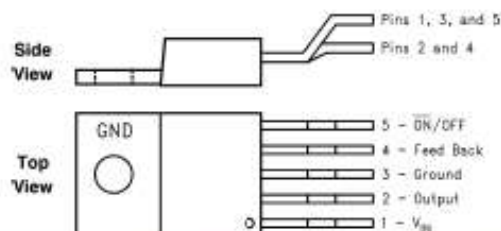


Figure 1. 5-Lead Bent and Staggered Leads, Through Hole TO-220 (T) Package
See Package Number NDH0005D

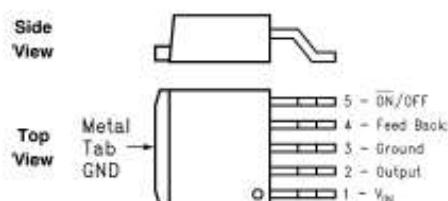


Figure 2. 5-Lead DDPAK/TO-263 (S) Package
See Package Number KTT0005B



These devices have limited built-in ESD protection. The leads should be shorted together or the device placed in conductive foam during storage or handling to prevent electrostatic damage to the MOS gates.

Absolute Maximum Ratings ⁽¹⁾⁽²⁾

Maximum Supply Voltage	45V
ON/OFF Pin Input Voltage	$-0.3 \leq V \leq +25V$
Feedback Pin Voltage	$-0.3 \leq V \leq +25V$
Output Voltage to Ground (Steady State)	-1V
Power Dissipation	Internally limited
Storage Temperature Range	-65°C to +150°C
ESD Susceptibility	
Human Body Model ⁽³⁾	2 kV
Lead Temperature	
DDPAK/TO-263 Package	
Vapor Phase (60 sec.)	+215°C
Infrared (10 sec.)	+245°C
TO-220 Package (Soldering, 10 sec.)	+260°C
Maximum Junction Temperature	+150°C

- (1) Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. Operating Ratings indicate conditions for which the device is intended to be functional, but do not ensure specific performance limits. For ensured specifications and test conditions, see the Electrical Characteristics.
- (2) If Military/Aerospace specified devices are required, please contact the Texas Instruments Sales Office/ Distributors for availability and specifications.
- (3) The human body model is a 100 pF capacitor discharged through a 1.5k resistor into each pin.

Operating Conditions

Temperature Range	$-40^{\circ}\text{C} \leq T_J \leq +125^{\circ}\text{C}$
Supply Voltage	4.5V to 40V

LM2596-3.3 Electrical Characteristics

Specifications with standard type face are for $T_J = 25^\circ\text{C}$, and those with **boldface type** apply over full Operating Temperature Range

LM2596-3.3					
Symbol	Parameter	Conditions	LM2596-3.3		Units (Limits)
			Typ (1)	Limit (2)	
SYSTEM PARAMETERS ⁽³⁾ Test Circuit Figure 20					
V _{OUT}	Output Voltage	4.75V ≤ V _{IN} ≤ 40V, 0.2A ≤ I _{LOAD} ≤ 3A	3.3	3.168/3.135 3.432/3.465	V V(min) V(max)
η	Efficiency	V _{IN} = 12V, I _{LOAD} = 3A	73		%

(1) Typical numbers are at 25°C and represent the most likely norm.

(2) All limits specified at room temperature (standard type face) and at temperature extremes (bold type face). All room temperature limits are 100% production tested. All limits at temperature extremes are ensured via correlation using standard Statistical Quality Control (SQC) methods. All limits are used to calculate Average Outgoing Quality Level (AOQL).

(3) External components such as the catch diode, inductor, input and output capacitors, and voltage programming resistors can affect switching regulator system performance. When the LM2596 is used as shown in the [Figure 20](#) test circuit, system performance will be as shown in system parameters of Electrical Characteristics section.

LM2596-5.0 Electrical Characteristics

Specifications with standard type face are for $T_J = 25^\circ\text{C}$, and those with **boldface type** apply over full Operating Temperature Range

Symbol	Parameter	Conditions	LM2596-5.0		Units (Limits)
			Typ (1)	Limit (2)	
SYSTEM PARAMETERS ⁽³⁾ Test Circuit Figure 20					
V _{OUT}	Output Voltage	7V ≤ V _{IN} ≤ 40V, 0.2A ≤ I _{LOAD} ≤ 3A	5.0	4.800/4.750 5.200/5.250	V V(min) V(max)
η	Efficiency	V _{IN} = 12V, I _{LOAD} = 3A	80		%

(1) Typical numbers are at 25°C and represent the most likely norm.

(2) All limits specified at room temperature (standard type face) and at temperature extremes (bold type face). All room temperature limits are 100% production tested. All limits at temperature extremes are ensured via correlation using standard Statistical Quality Control (SQC) methods. All limits are used to calculate Average Outgoing Quality Level (AOQL).

(3) External components such as the catch diode, inductor, input and output capacitors, and voltage programming resistors can affect switching regulator system performance. When the LM2596 is used as shown in the [Figure 20](#) test circuit, system performance will be as shown in system parameters of Electrical Characteristics section.

LM2596-12 Electrical Characteristics

Specifications with standard type face are for $T_J = 25^\circ\text{C}$, and those with **boldface type** apply over full Operating Temperature Range

Symbol	Parameter	Conditions	LM2596-12		Units (Limits)
			Typ (1)	Limit (2)	
SYSTEM PARAMETERS ⁽³⁾ Test Circuit Figure 20					
V _{OUT}	Output Voltage	15V ≤ V _{IN} ≤ 40V, 0.2A ≤ I _{LOAD} ≤ 3A	12.0	11.52/11.40 12.48/12.60	V V(min) V(max)
η	Efficiency	V _{IN} = 25V, I _{LOAD} = 3A	90		%

(1) Typical numbers are at 25°C and represent the most likely norm.

(2) All limits specified at room temperature (standard type face) and at temperature extremes (bold type face). All room temperature limits are 100% production tested. All limits at temperature extremes are ensured via correlation using standard Statistical Quality Control (SQC) methods. All limits are used to calculate Average Outgoing Quality Level (AOQL).

(3) External components such as the catch diode, inductor, input and output capacitors, and voltage programming resistors can affect switching regulator system performance. When the LM2596 is used as shown in the [Figure 20](#) test circuit, system performance will be as shown in system parameters of Electrical Characteristics section.

LM2596-ADJ Electrical Characteristics

Specifications with standard type face are for $T_J = 25^\circ\text{C}$, and those with **boldface type** apply over full Operating Temperature Range

Temperature Range					
Symbol	Parameter	Conditions	LM2596-ADJ		Units (Limits)
			Typ (1)	Limit (2)	
SYSTEM PARAMETERS ⁽³⁾ Test Circuit Figure 20					
V _{FB}	Feedback Voltage	4.5V ≤ V _{IN} ≤ 40V, 0.2A ≤ I _{LOAD} ≤ 3A V _{OUT} programmed for 3V. Circuit of Figure 20	1.230	1.193/1.180 1.267/1.280	V V(min) V(max)
η	Efficiency	V _{IN} = 12V, V _{OUT} = 3V, I _{LOAD} = 3A	73		%

(1) Typical numbers are at 25°C and represent the most likely norm.

(2) All limits specified at room temperature (standard type face) and at temperature extremes (bold type face). All room temperature limits are 100% production tested. All limits at temperature extremes are ensured via correlation using standard Statistical Quality Control (SQC) methods. All limits are used to calculate Average Outgoing Quality Level (AOQL).

(3) External components such as the catch diode, inductor, input and output capacitors, and voltage programming resistors can affect switching regulator system performance. When the LM2596 is used as shown in the [Figure 20](#) test circuit, system performance will be as shown in system parameters of Electrical Characteristics section.

All Output Voltage Versions Electrical Characteristics

Specifications with standard type face are for $T_J = 25^\circ\text{C}$, and those with **bold type** apply over full Operating Temperature Range. Unless otherwise specified, $V_{IN} = 12\text{V}$ for the 3.3V, 5V, and Adjustable version and $V_{IN} = 24\text{V}$ for the 12V version. $I_{LOAD} = 500\text{mA}$

LM2596-XX					
Symbol	Parameter	Conditions	LM2596-XX		Units (Limits)
			Typ (1)	Limit (2)	
DEVICE PARAMETERS					
I _b	Feedback Bias Current	Adjustable Version Only, V _{FB} = 1.3V	10	50/100	nA nA (max)
f _o	Oscillator Frequency	See (3)	150	127/110 173/173	kHz kHz(min) kHz(max)
V _{SAT}	Saturation Voltage	I _{OUT} = 3A (4) (5)	1.16	1.4/1.5	V V(max)
DC	Max Duty Cycle (ON)	See (3)	100		%
	Min Duty Cycle (OFF)	See (3)	0		
I _{CL}	Current Limit	Peak Current (4)(5)	4.5	3.6/3.4 6.9/7.5	A A(min) A(max)
I _L	Output Leakage Current	Output = 0V (4)(6)		50	μA(max)
		Output = -1V (7)	2	30	mA mA(max)
I _Q	Quiescent Current	See (3)	5	10	mA mA(max)

(1) Typical numbers are at 25°C and represent the most likely norm.

(2) All limits specified at room temperature (standard type face) and at temperature extremes (bold type face). All room temperature limits are 100% production tested. All limits at temperature extremes are ensured via correlation using standard Statistical Quality Control (SQC) methods. All limits are used to calculate Average Outgoing Quality Level (AOQL).

(3) The switching frequency is reduced when the second stage current limit is activated.

(4) No diode, inductor or capacitor connected to output pin.

(5) Feedback pin removed from output and connected to 0V to force the output transistor switch ON.

(6) Feedback pin removed from output and connected to 12V for the 3.3V, 5V, and the ADJ. version, and 15V for the 12V version, to force the output transistor switch OFF.

(7) $V_{IN} = 40\text{V}$.

All Output Voltage Versions Electrical Characteristics (continued)

Specifications with standard type face are for $T_J = 25^\circ\text{C}$, and those with **boldface type** apply over full Operating Temperature Range. Unless otherwise specified, $V_{IN} = 12\text{V}$ for the 3.3V, 5V, and Adjustable version and $V_{IN} = 24\text{V}$ for the 12V version. $I_{LOAD} = 500\text{ mA}$.

Symbol	Parameter	Conditions	LM2596-XX		Units (Limits)
			Typ (1)	Limit (2)	
I _{STBY}	Standby Quiescent Current	ON/OFF pin = 5V (OFF) ⁽⁷⁾	80	200/250	μA μA(max)
θ _{JC}	Thermal Resistance	TO-220 or TO-263 Package, Junction to Case	2		°C/W
θ _{JA}		TO-220 Package, Junction to Ambient ⁽⁸⁾	50		°C/W
θ _{JA}		TO-263 Package, Junction to Ambient ⁽⁹⁾	50		°C/W
θ _{JA}		TO-263 Package, Junction to Ambient ⁽¹⁰⁾	30		°C/W
θ _{JA}		TO-263 Package, Junction to Ambient ⁽¹¹⁾	20		°C/W
ON/OFF CONTROL Test Circuit Figure 20					
V _{IH} V _{IL}	ON /OFF Pin Logic Input Threshold Voltage	Low (Regulator ON) High (Regulator OFF)	1.3	0.6 2.0	V V(max) V(min)
I _H	ON /OFF Pin Input Current	V _{LOGIC} = 2.5V (Regulator OFF)	5	15	μA μA(max)
I _L		V _{LOGIC} = 0.5V (Regulator ON)	0.02	5	μA μA(max)

(8) Junction to ambient thermal resistance (no external heat sink) for the TO-220 package mounted vertically, with the leads soldered to a printed circuit board with (1 oz.) copper area of approximately 1 in^2 .

(9) Junction to ambient thermal resistance with the TO-263 package tab soldered to a single printed circuit board with 0.5 in^2 of (1 oz.) copper area.

(10) Junction to ambient thermal resistance with the TO-263 package tab soldered to a single sided printed circuit board with 2.5 in^2 of (1 oz.) copper area.

(11) Junction to ambient thermal resistance with the TO-263 package tab soldered to a double sided printed circuit board with 3 in^2 of (1 oz.) copper area on the LM2596S side of the board, and approximately 16 in^2 of copper on the other side of the p-c board. See [Application Information](#) in this data sheet and the thermal model in Switchers Made Simple™ version 4.3 software.

BIBLIOGRAFIA

- [1] http://build18.ece.cmu.edu/wiki/index.php/Automatic_Bicycle_Transmission. Consultado el 01 de Septiembre del 2014.
- [2] <http://evolvethetbike.com/>. Consultado el 01 de Septiembre del 2014.
- [3] <http://www.fallbrooktech.com/nuvinci-technology>. Consultado el 01 de Septiembre del 2014.
- [4] <https://www.electricbike.com/category/reviews/electric-bikes/>. Consultado el 04 de Septiembre del 2014
- [5] <https://www.superpedestrian.com/>. Consultado el 06 de Septiembre del 2014
- [6] <http://bicihome.com/la-historia-de-las-bicicleta/>. Consultado el 12 de Septiembre del 2014
- [7] Paco N., Javier R., Akex F., Oriol A., Cristina G., Jordi J., Miguel Ángel R. (2010). La ingeniería de la bicicleta. España: Fundación ESTEYCO. Consultado el 15 de Septiembre del 2014
- [8] <http://historiaybiografias.com/bicicleta/>. Consultado el 18 de Septiembre del 2014
- [9] <http://lema.rae.es/drae/?val=bici>. Consultado el 18 de Septiembre del 2014
- [10] <http://labicikleta.com/>. Consultado el 01 de Noviembre del 2014
- [11] <http://si.shimano.com/>. Consultado el 10 de Noviembre del 2014
- [12] <http://sheldonbrown.com/>. Consultado el 11 de Noviembre del 2014
- [13] Richard Grant, Richard Ballantine. (1992). El gran libro de la bicicleta. Madrid: El país Aguilar. Consultado el 11 de Noviembre del 2014
- [14] José Manuel Ruiz Gutiérrez. (2007). Manual de Programación Arduino. San Francisco, California: Crative Commons. Consultado el 15 de Noviembre del 2014.
- [15] <http://www.arduino.cc/>. Consultado el 17 de Noviembre del 2014.
- [16] Muhammad H. Rashid. (1993). Electronica de Potencia. Mexico:Prentice Hall Hispanoamerica, S.A. Consultado el 22 de noviembre del 2014.