



UNIVERSIDAD DEL PAPALOAPAN

Campus Loma Bonita

INGENIERÍA EN COMPUTACIÓN

**PROGRAMACIÓN Y DISEÑO DE UN SISTEMA
AUTOMÁTICO PARA EL PESADO Y ENROLLADO
DE QUESO TIPO OAXACA**

**TESIS PROFESIONAL PARA OBTENER EL GRADO DE
INGENIERO EN COMPUTACIÓN**

PRESENTA:

FLOR ENGRACIA SÁNCHEZ ZANATTA

ASESOR DE TESIS:

M. C. José Luis Najera Sánchez

LOMA BONITA, OAXACA.

JULIO 2022.



Universidad del Papaloapan

FECHA:	07 de Junio del 2022
AREA:	Vice-Rectoría Académica
OFICIO NUMERO:	UNPA/VRA/103/2022
ASUNTO:	Autorización de Impresión de tesis.

**C. FLOR ENGRACIA SANCHEZ ZANATTA
PRESENTE:**

En base al artículo 120 del reglamento de alumnos, por medio de la presente se aprueba la impresión de la tesis titulada **“Programación y Diseño de un Sistema Automático para el Pesado y Enrollado de Queso Tipo Oaxaca”** así como la programación del examen profesional bajo la dirección del M.C. José Luis Nájera Sánchez.

Sin más por el momento aprovecho la ocasión para enviarle un cordial saludo.

Atentamente.
terra ubérrima, mens aperta
Bou Lo-tama, chi ji jú


MC. HÉCTOR LÓPEZ ARJONA
Vice-Rector Académico.



C.c.p. Dr. Sergio Fabián Ruiz Paz Jefe de Carrera de la Ingeniería en Computación
C.c.p. L.P. Yesenia Barrientos Arenal. Jefa del Departamento de Servicios Escolares
C.c.p. M.C. José Luis Nájera Sánchez. Director de tesis.
C.c.p. Archivo.



UNIVERSIDAD DEL PAPALOAPAN

Campus Loma Bonita

Oficio: ICOM/06/002/22
Loma Bonita, Oaxaca, a 14 de junio de 2022

M.E. Yesenia Barrientos Arenal
Jefa del Departamento de Servicios Escolares
PRESENTE

Mediante la presente, le informo que la jefatura de carrera a mi cargo, con visto bueno de la Vice-Rectoría Académica, ha designado a los siguientes profesores como sinodales para examen profesional de la alumna **C. Flor Engracia Sánchez Zanatta**, quien defenderá su trabajo de tesis titulado **“PROGRAMACIÓN Y DISEÑO DE UN SISTEMA AUTOMÁTICO PARA EL PESADO Y ENROLLADO DE QUESO TIPO OAXACA”**, para obtener el título de Licenciada en Ingeniería en Computación.

Titulares:

Presidente: **M.C. Rafael Fernando González Zarate**
Secretario: **M.C. Luis Alberto Hernández Zuccolotto**
Vocal: **M.C. José Luis Nájera Sánchez**

Suplentes:

Dr. Hiram Netzahualcóyotl García Lozano
Dra. Nancy Pérez Castro



Atentamente
terra uberrima, mens aperta
Rou l'ozama, chi, ji, iú

Dr. Sergio Fabián Ruiz Paz
Jefe de la Carrera de **INGENIERIA EN COMPUTACION**
Computación

M.C. Héctor López Arjona
Vice-Rector Académico

c.c.p. M.C. Héctor López Arjona. Vice-Rector académico. Para su conocimiento
c.c.p. Archivo

DEDICATORIA

Gracias a Dios por la vida de mis seres queridos, porque cada día bendice mi vida con la hermosa oportunidad de estar y disfrutar al lado de las personas que más me aman, y a las que yo sé que más amo en la vida.

El amor recibido, la dedicación y la paciencia con la que cada día se preocupaban mis padres, esposo, hijos y hermanos por mí avance y el desarrollo de esta tesis, es simplemente único y se refleja en la vida.

Gracias a la vida por este nuevo triunfo, gracias a todas las personas que me apoyaron y creyeron en la realización de esta tesis.

A mi esposo que fue el ingrediente perfecto para poder haber terminado esta tesis con éxito y poder disfrutar el privilegio de su compañía, que se preocupó por mí en cada momento y que siempre quiere lo mejor para mi porvenir.

A mis hijos son el mejor regalo que haya podido recibir de Dios, son mi mayor tesoro y fuente de inspiración, por eso quiero agradecer cada momento de felicidad con el que llenan mi vida.

No ha sido sencillo el camino hasta ahora, pero gracias a sus aportes, a su amor, a su inmensa bondad y apoyo, lo complicado de lograr esta meta se ha notado menos. Les agradezco, y hago presente mi gran afecto hacia ustedes, mi hermosa familia.

AGRADECIMIENTOS

Gracias a la universidad, por haber permitido formarme en ella, gracias a todas las personas que fueron partícipes en este proceso, ya sea de manera directa o indirecta.

Quiero agradecer a los profesores por haber tomado la decisión de enseñar, de compartir sus conocimientos, con todo aquel que lo requiera, por creer en la educación y el desarrollo de la sociedad a través de la enseñanza.

Gracias a todos ustedes que fueron los responsables de realizar su gran aporte, en especial a mí director de tesis M. C. José Luis Najera Sánchez a mis asesores de tesis M. C. Rafael Fernando González Zarate y al Dr. Hiram Netzahualcóyotl García Lozano quienes me apoyaron en todo momento, que se tomaron el tiempo de ayudarme en las dudas que surgían y en encontrar una solución para cada una de ellas.

A todas estas personas y a las que me faltó por mencionar pero que las llevo en mis recuerdos.

RESUMEN

En esta tesis se realiza la programación y diseño de un sistema automático para el pesado y enrollado queso tipo Oaxaca. La realización de este trabajo se llevó a cabo en los laboratorios de mecatrónica, de la Universidad del Papaloapan Campus Loma Bonita. Este sistema estará constituido por sensores, mecanismos de movimientos, por una celda de carga, cuyo diseño mecánico se realizó bajo la plataforma CAD de SolidWorks. En el primer capítulo habla brevemente de los inicios del queso Oaxaca y como se han ido desarrollando muchas empresas y aún sigue sin existir una máquina automática para el desarrollo de este proceso. En el segundo capítulo aquí se explica, de manera sencilla, conceptos básicos, clasificación de los sensores, distintos tipos de sistemas embebidos, etapas de potencias, para el diseño del sistema del prototipo. A continuación, en el tercer capítulo, se realizan las pruebas para el sistema de pesado, las conexiones necesarias entre la celda de carga, el módulo HX711, Arduino y el LCD. Así como el diseño de la placa fenólica donde se le montan los elementos necesarios para el funcionamiento. Por último, en el cuarto capítulo, se describen las pruebas realizadas para verificar el funcionamiento del prototipo y su programación. Este capítulo se centra en el ensamble de las piezas impresas con la finalidad de verificar el correcto funcionamiento. Se finaliza con breves conclusiones, en las cuales se presenta el resultado obtenido.

ABSTRACT

In this thesis, the programming and design of an automatic system for weighing and rolling Oaxaca-type cheese is carried out. This work was carried out in the mechatronics laboratories of the University of Papaloapan Campus Loma Bonita. This system will consist of sensors, movement mechanisms, a load cell, whose mechanical design was carried out under the SolidWorks CAD platform. In the first chapter he briefly talks about the beginnings of Oaxaca cheese and how many companies have been developing and there is still no automatic machine for the development of this process. The second chapter here explains, in a simple way, basic concepts, classification of sensors, different types of embedded systems, power stages, for the design of the prototype system. Next, in the third chapter, the tests for the weighing system, the necessary connections between the load cell, the HX711 module, Arduino and the LCD are carried out. As well as the design of the phenolic plate where the necessary elements for the operation are mounted. Finally, in the fourth chapter, the tests carried out to verify the functioning of the prototype and its programming are described. This chapter focuses on the assembly of the printed parts in order to verify correct operation.

It ends with brief conclusions, in which the obtained result is presented.

ÍNDICE GENERAL

DEDICATORIA.....	I
AGRADECIMIENTOS	II
RESUMEN.....	III
ABSTRACT.....	IV
ÍNDICE DE FIGURAS	VI
ÍNDICE DE TABLAS	VIII
ÍNDICE DE DIAGRAMAS.....	VIII
1. INTRODUCCIÓN.....	1
1.1 PLANTEAMIENTO DEL PROBLEMA.	3
1.2 JUSTIFICACIÓN.....	4
1.3 ALCANCE Y LIMITACIÓN DEL PROBLEMA.	5
1.4 OBJETIVOS.....	6
1.4.1 OBJETIVO GENERAL.....	6
1.4.2 OBJETIVOS ESPECÍFICOS.....	6
1.5 HIPÓTESIS.....	7
2. MARCO TEÓRICO.....	8
2.1 SENSORES.....	8
2.1.1 CLASIFICACIÓN DE LOS SENSORES.....	8
2.1.2 PRINCIPIO DE LOS SENSORES.....	10
2.2 SISTEMAS EMBEBIDOS.....	19
2.2.1 APLICACIÓN DE LOS SISTEMAS EMBEBIDOS.....	21
2.2.2 EJEMPLOS DE SISTEMAS EMBEBIDOS.....	22
2.3 ETAPA DE POTENCIA.....	23
2.3.1 RELEVADORES.....	25
2.3.2 TRANSISTOR.....	26
2.2.3 ENGRANES.....	28
2.3.4 MOTORREDUCTORES.....	31
2.4 MECANISMO.....	33

2.4.1 CLASIFICACIÓN DE LOS MECANISMOS.....	34
3. DESARROLLO.....	40
3.1 PRUEBA DE SISTEMA DE PESAJE.....	40
3.1.1 FORMA DE INSTALAR UNA CELDA DE CARGA.....	42
3.1.2 TRANSMISOR DE CELDA DE CARGA.....	42
3.1.3. CONEXIONES ENTRE LA CELDA DE CARGA, MÓDULO.....	45
HX711 Y ARDUINO.....	45
3.1.4. CONEXIÓN DEL LCD (DISPLAY) Y ARDUINO.....	51
3.1.5. DISEÑO DE LA TARJETA DE CIRCUITOS IMPRESA.....	57
3.1.6 CONEXIÓN ENTRE EL MÓDULO HX711, EL DISPLAY Y.....	58
ARDUINO.....	58
3.2 DISEÑO MECÁNICO.....	61
3.3 FUNCIONAMIENTO DEL PROTOTIPO.....	64
3.4 PROGRAMACIÓN E IMPLEMENTACIÓN DEL SISTEMA.....	69
4. PRUEBAS.....	74
4.1 DISEÑO PLACA FENÓLICA.....	74
4.2 PROTOTIPO.....	84
5. RESULTADOS Y CONCLUSIONES.....	85
APÉNDICE: PROGRAMA.....	86

ÍNDICE DE FIGURAS

Figura 1. Transductor resistivo.....	12
Figura 2. Transductores de inductancia variable.	13
Figura 3. Transductor de reluctancia variable.....	14
Figura 4. Transductor capacitivo.....	15
Figura 6. Galga cementada.....	16
Figura 5. Galga sin cementar.....	16
Figura 7. Puente de Wheatstone para galga extensométrica.	17
Figura 8. Transductor de presión de silicio difundido.....	18
Figura 9. Construcción básica y símbolo del Triac.....	24
Figura 10. Construcción básica y símbolo del SCR.....	25
Figura 11. Funcionamiento del Relay.....	25
Figura 12. Transistor tipo PNP y transistor tipo NPN.....	26
Figura 13. Interface digital de relevador con transistor.....	27
Figura 14. Engranajes cónicos de dientes rectos.....	30
Figura 15. Mecanismo biela-manivela.....	31
Figura 16. Sistema manivela-torno para elevación de cargas.....	35
Figura 17. Sistema piñón-cremallera.....	36
Figura 18. Gato de un coche (ejemplo del mecanismo huisillo-tuerca).....	37
Figura 19. Leva y excéntrica.....	38
Figura 20. Sistema biela-manivela.....	39
Figura 21. Cigüeñal.....	39
Figura 22. Celda de carga.....	40
Figura 23. Esquema de instalación de la celda de carga.....	42
Figura 24. Módulo HX711.....	43
Figura 25. Periodo de conversión.....	43
Figura 26. Conexión entre la celda de carga, el módulo HX711 y Arduino...	47
Figura 27. CD o Display.....	52
Figura 28. Esquema de los pines del LCD o Display.....	53
Figura 29. Prototipo de máquina.....	63
Figura 30. Engrane recto.....	64
Figura 31. Engrane recto.....	65
Figura 32. Biela-manivela.....	65
Figura 33. Sostén de biela (sistema de guiado).....	66
Figura 34. Engrane unido a un soporte angular.....	67
Figura 35. Proceso de diseño de una pieza con una impresora 3d.....	69
Figura 36. Engrane base.....	70
Figura 37. Engrane paralelo.....	70
Figura 38. Balero.....	71
Figura 39. Manivela.....	71

Figura 40. Sistema de guiado.	72
Figura 41. Arillo.....	72
Figura 42. Soporte angular.	73
Figura 43. Base tubular.....	73
Figura 44. Base del prototipo.....	74
Figura 45. Base del motorreductor.....	75
Figura 46. Celda de carga.	75
Figura 47. Prototipo final.....	84

ÍNDICE DE TABLAS

Tabla 1. Conexión entre la celda de carga y el módulo HX711.	45
Tabla 2. Conexión entre el módulo HX711 y Arduino.	46

ÍNDICE DE DIAGRAMAS

Diagrama 1. Conexión entre el LCD y Arduino.	54
Diagrama 2. Diagrama esquemático para la conexión entre Arduino, el LCD y el Módulo HX711.	59
Diagrama 3. Tarjeta de circuito impresa diseñada con el software EAGLE. .	60

1. INTRODUCCIÓN.

El queso Oaxaca, también se conoce con el nombre de quesillo, queso de hebra y queso asadero. Como su nombre lo indica originario de Oaxaca, se obtiene mediante una técnica cuidadosa de elaboración, de consistencia semidura para darle una forma elástica, misma que se puede enrollar hasta formar una pelota.

Su historia se remonta al año 1885, cuando una niña llamada Leobarda Castellanos, de tan solo 14 años de edad, estaba encargada de cuidar la leche cuajada para hacer el queso, en un descuido, a la pequeña se le olvidó atender su enmienda, dándose cuenta que la "cuajada" ya se había pasado del punto para elaborar el queso. Para que sus padres no la descubrieran, vació agua caliente sobre dicha masa, dando esto como resultado una fundición y obteniéndose una mezcla chiclosa a la cuál llamó quesillo.

Los padres de la pequeña probaron dicha mezcla, notando que tenía un sabor exquisito, empezaron a elaborarlo gracias a que el "quesillo" tenía una gran aceptación entre la población.

Este producto enriqueció la gastronomía local y tiempo después sirvió para dar a conocer a Reyes Etlá (población a sólo 15 minutos de la Capital de Oaxaca) por muchos países del mundo como "La Cuna del Quesillo".

Se elabora con leche de vaca, con leche entera permite obtener un quesillo de sabor más apreciado, mientras que con leche descremada se obtiene un quesillo de sabor notable únicamente para los paladares conocedores.

Su elaboración requiere de destreza y conocimiento, puesto que presenta ciertos puntos críticos cuyo control es indispensable, por ejemplo, la acidez adecuada de la leche, la acidificación de la cuajada, la determinación del punto de hebra, el amasado de la pasta, la fundición con agua caliente (65° a 70°), luego el estirado y el enfriado de las hebras y dentro de este proceso uno de los puntos principales es el enrollado; se pueden hacer rollos de distinto peso.

El queso Oaxaca goza de la mayor popularidad entre los consumidores mexicanos, tanto en las clases populares como en las de mayores ingresos, se vende tanto en mercados locales como en tiendas de autoservicio con diferentes presentaciones.

Debido a su aptitud para fundir, se consume frecuentemente acompañando a los platillos tradicionales de la cocina mexicana y específicamente los típicos “antojitos” (1).

1.1 PLANTEAMIENTO DEL PROBLEMA.

Para una pequeña o mediana empresa, adquirir maquinaria para el proceso de la elaboración de queso de hebra es algo incosteable. Debido a que los volúmenes de producto que se procesa al día es pequeño, la inversión que se tendría que realizar para la adquisición de equipo sofisticado sería injustificable. Ante esta problemática surge la idea de poder desarrollar tecnología al alcance de las industrias locales que se dedican a esta actividad. Por lo que se pretende desarrollar un sistema embebido que pese y enrolle queso de manera semiautomática.

La máquina desarrollada estará controlada por una tarjeta Arduino. Esta tarjeta consiste en una plataforma de hardware de código abierto, (una placa con entradas y salidas, analógicas y digitales) con un entorno de desarrollo (IDE) que está basado en un lenguaje de programación de fácil comprensión (2).

1.2 JUSTIFICACIÓN.

Al desarrollar una máquina controlada por un sistema embebido, que pese y enrolle queso de forma semiautomática, se proporcionará a los productores de la región una herramienta tecnológica, que les permita realizar los procesos mencionados de manera más eficiente.

1.3 ALCANCE Y LIMITACIÓN DEL PROBLEMA.

En estas empresas, la necesidad de colocar en el mercado un producto de buena calidad y que llegue al consumidor en el menor tiempo posible, representa una mayor utilidad. Por lo tanto, es pertinente desarrollar tecnología al alcance de empresas que manejen volúmenes pequeños de subproductos lácteos. Tomando en cuenta que la textura y calidad de estos productos no tiene por qué ser diferente ni cambiar su sabor natural. De este modo, el empleo de una máquina mejora la higiene y reduce el tiempo en que este producto puede llegar al consumidor. Esta es la motivación de diseñar una maquina capaz de realizar el pesado y dar la presentación final a este tipo de queso.

Alcances:

- Se realizará un programa en el que la tarjeta Arduino procesa la información recibida del sensor de peso y lo muestra en un LCD.
- El motor será controlado por la tarjeta Arduino a través de una programación, que le indicará cuando detenerse.

1.4 OBJETIVOS.

1.4.1 OBJETIVO GENERAL.

Desarrollar el algoritmo, para un sistema embebido, capaz de estimar el peso, así como de realizar el enrollado, de forma semiautomática, de queso tipo Oaxaca.

1.4.2 OBJETIVOS ESPECÍFICOS.

- Desarrollar un sistema embebido para llevar a cabo la conexión entre la tarjeta Arduino, el LCD y el sensor de peso.
- Realizar una interfaz que indique el peso estimado.
- Utilizar un programa especial para realizar el diseño 3d de la máquina.

1.5 HIPÓTESIS.

Es posible automatizar el pesado y enrollado de queso mediante una máquina controlada por un sistema embebido.

2. MARCO TEÓRICO.

2.1 SENSORES.

Los sensores son una de las partes esenciales, para que una máquina funcione en forma automática. Estos están diseñados para recibir información de una magnitud del exterior y transformarla en otra magnitud (en nuestro caso eléctrica) que sea posible cuantificar y manipular.

Existen diferentes tipos de sensores en función al tipo de variable que se tenga que medir, ya sean de presión, fuerza, temperatura, humedad, movimiento, etc.

2.1.1 CLASIFICACIÓN DE LOS SENSORES.

Los sensores se clasifican según el aporte de energía, según la señal de salida, según el modo de funcionamiento y según la relación de entrada/salida.

A continuación, se describe dicha clasificación.

Según aporte de energía

- Moduladores o activos. La energía de la señal de salida procede mayormente de una fuente de energía auxiliar y la entrada solo se encarga de controlar la salida.
- Generadores o pasivos. La energía de salida es producida por la energía de entrada, tomando únicamente la energía del medio donde se miden.

Según la señal de salida

- Analógicos. La salida varía de forma continua, la información se encuentra dentro de la amplitud, en el caso de que la información se encuentre en la frecuencia se denominan "cuasi-digitales".
- Digitales. La salida varía en forma de saltos o pasos discretos, no requiere conversión A/D y la transmisión de salida es menos compleja, normalmente presentan mejor fidelidad, fiabilidad y exactitud.

Según el modo de funcionamiento

- Deflexión. Describe el cambio de posición de un área de un objeto, como resultado de una carga.
- Comparación. Mediante la aplicación de un efecto opuesto al generado por la magnitud medida se intenta mantener nula la deflexión.

Según la relación entrada/salida

- El tipo de orden está relacionado con el número de elementos almacenadores de energía independiente que incluye el sensor y repercute en su exactitud y velocidad de respuesta. Esta clasificación es de gran importancia cuando el sensor forma parte de un sistema de control en lazo cerrado. (4)

Anteriormente, las máquinas funcionaban sin sensores, en la actualidad gracias a la tecnología, estos sensores se ocupan como captadores de información. Se encargan de recibir la información del exterior y la transforman en otra magnitud, de manera que pueda ser procesada y utilizada. En el desarrollo de este diseño se requiere de un sensor, existen distintos tipos y cada uno funciona de diferente forma, lo que se conoce como principio de funcionamiento del sensor. A continuación, se presentan algunos de los muy diversos principios de funcionamiento de los sensores.

2.1.2 PRINCIPIO DE LOS SENSORES.

El principio fundamental de los sensores se basa en la forma en que estos reciben la información del medio. Dentro de este se pueden encontrar los elementos electromecánicos de presión. Estos utilizan un elemento mecánico elástico combinado con un transductor que genera la señal

eléctrica correspondiente. Según su funcionamiento los elementos electromecánicos de presión se clasifican en los siguientes tipos:

- Transmisores electrónicos de equilibrio de fuerzas. Su principal característica es tener un movimiento pequeño de la barra de equilibrio, poseen realimentación, buena elasticidad y un nivel alto en la señal de salida. Por su constitución mecánica presentan un ajuste cero, alcance complicado, alta sensibilidad a vibraciones y estabilidad en el tiempo es de media a baja.
- Transductores resistivos. Su señal de salida es bastante potente como para proporcionar una corriente de salida suficiente para el funcionamiento de los instrumentos indicadores sin necesidad de amplificación. Son insensibles a pequeños movimientos producidos por el contacto del cursor, muy sensibles a vibraciones y presentan una estabilidad baja en el tiempo (figura 1).

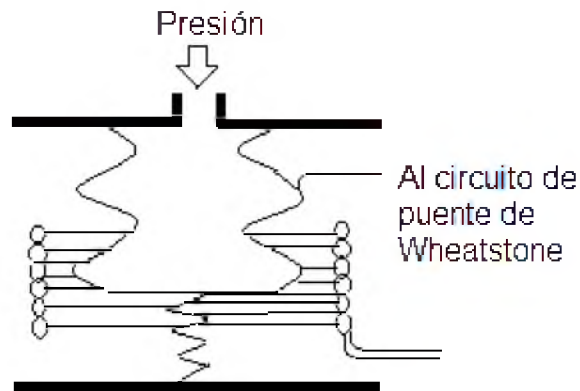


Figura 1. Transductor resistivo.

- Transductores magnéticos. Según el principio de funcionamiento se clasifican en dos grupos.

a) Transductores de inductancia variable. El desplazamiento de un núcleo móvil dentro de una bobina aumenta la inductancia de ésta en forma casi proporcional a la porción metálica del núcleo contenido dentro de la bobina (figura 2).

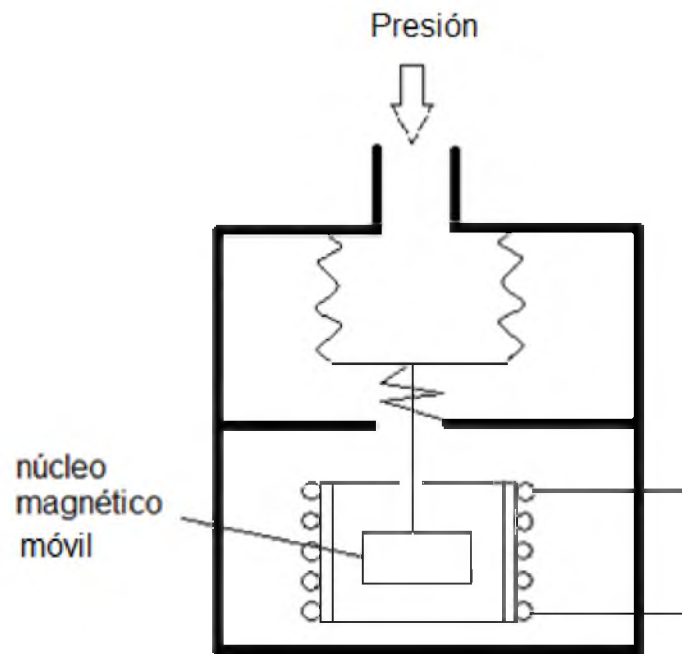


Figura 2. Transductores de inductancia variable.

b) Transductores de reluctancia variable. Mediante un imán permanente o un electroimán crean un campo magnético para dentro de él mover una armadura de material magnético (figura 3).

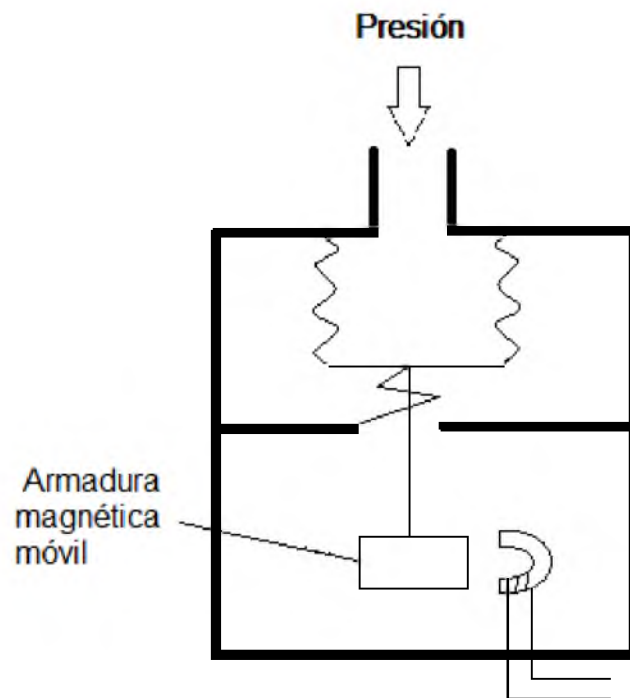


Figura 3. Transductor de reluctancia variable.

- Transductores capacitivos. Se basan en la variación de capacidad que se produce en un condensador al desplazarse una de sus placas por la aplicación de presión (figura 4). La placa móvil tiene forma de diafragma y se encuentra situada entre dos placas fijas. De este modo se tienen dos condensadores uno de capacidad fija o de referencia y el otro de capacidad variable, que pueden compararse en circuitos oscilantes o bien en circuitos de puente de Wheatstone alimentados con corriente alterna.

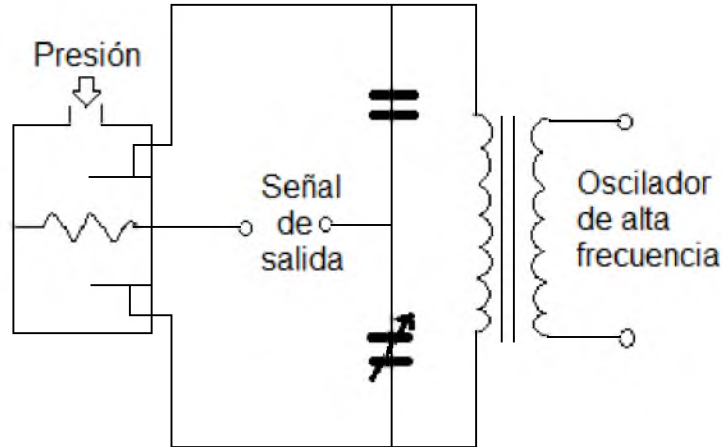


Figura 4. Transductor capacitivo.

- Transductores piezoeléctricos. Son materiales cristalinos que, al deformarse físicamente por la acción de una presión, generan una señal eléctrica. Dos materiales típicos en los transductores piezoeléctricos son el cuarzo y el titanato de bario, capaces de soportar temperaturas del orden de 150° C en servicio continuo y de 230° C en servicio intermitente.
- Galgas extensométricas. Se basan en la variación de longitud y de diámetro, y por lo tanto de resistencia, que tiene lugar cuando un hilo de resistencia se encuentra sometido a una tensión mecánica por la acción de una presión.

Existen dos tipos de galgas extensométricas:

- a) Galgas cementadas formadas por varios bucles de hilo muy fino que están pegados a una hoja base de cerámica, papel o plástico (figura 5).
- b) Galgas sin cementar en las que los hilos de resistencia descansan entre un armazón fijo y otro móvil bajo una ligera tensión inicial (figura 6).

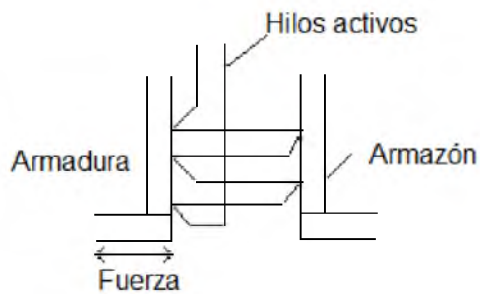


Figura 6. Galga sin cementar.

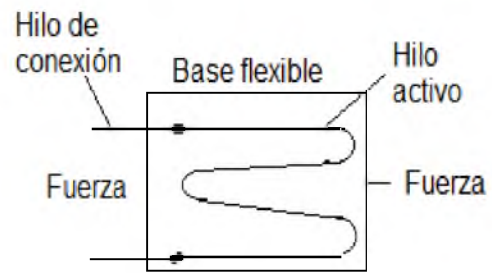


Figura 5. Galga cementada.

En las dos galgas extensométricas, la aplicación de presión estira o comprime los hilos según sea la disposición que el fabricante haya adoptado, modificando la resistencia de los mismos. La galga forma parte de un puente de Wheatstone (figura 7) y cuando está sin tensión tiene una resistencia eléctrica determinada. Se aplica al circuito una tensión nominal, tal que la pequeña corriente que circula por la resistencia crea una caída de tensión en la misma y el puente se equilibra para estas condiciones.

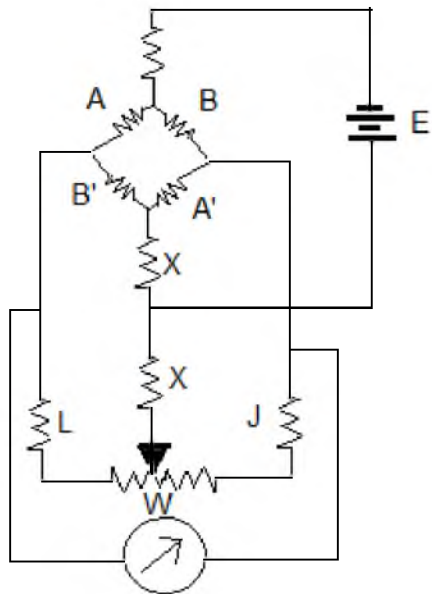


Figura 7. Puente de Wheatstone para galga extensométrica.

Los transductores de presión de silicio difundido son una innovación de la galga extensométricas. Consisten en un elemento de silicio situado dentro de una cámara, conteniendo silicona, que está en contacto con el proceso a través de un diafragma flexible (figura 8).

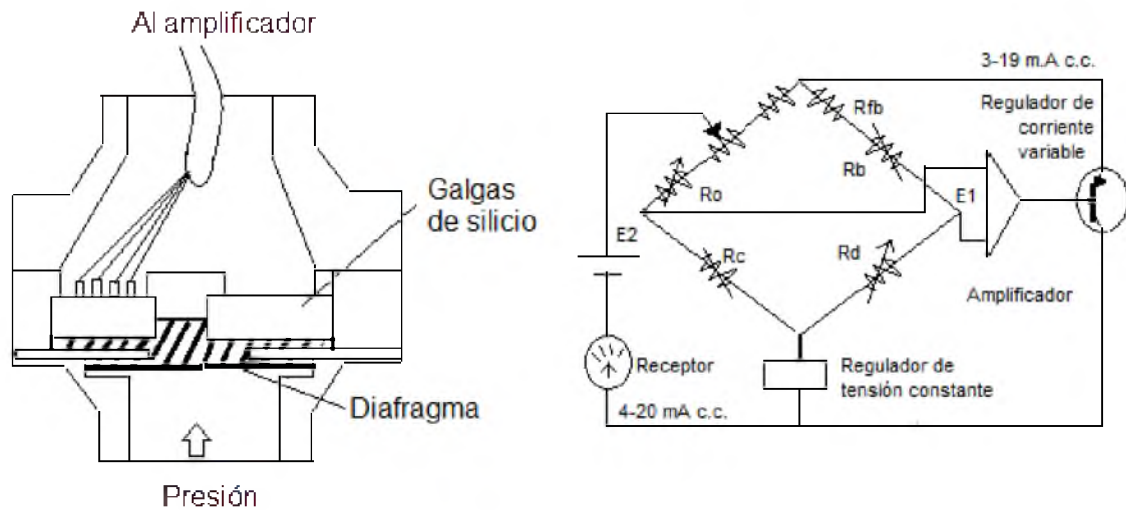


Figura 8. Transductor de presión de silicio difundido.

Las galgas extensométricas pueden alimentarse con c.c. o c.a. y son utilizadas en medidas estáticas y dinámicas (5).

Para la realización de este proyecto se utilizarán galgas extensométricas, debido a que estas satisfacen las necesidades del prototipo a desarrollar, además de encontrarse fácilmente en el mercado y tener un costo accesible.

Como se explicó anteriormente, la galga responde a una deformación física con una señal eléctrica proporcional (en el rango indicado por el fabricante). Sin embargo, para que sea de utilidad, esta señal eléctrica debe ser procesada. Una forma de procesar esta información es por medio de un sistema embebido. En la siguiente sección se describen las características principales de este tipo de sistemas.

2.2 SISTEMAS EMBEBIDOS.

Los circuitos electrónicos digitales capaces de realizar operaciones que sirven para cumplir una tarea específica en un producto, están relacionados con un sistema embebido.

Los sistemas embebidos no son equivalentes a los sistemas de cómputo usados en laptops o en computadoras de escritorio. Estos suelen tener recursos limitados y aplicaciones específicas siendo sumamente útiles en múltiples ambientes, por ejemplo, en el campo automotriz (sistemas de inyección de gasolina, alarmas contra robos, control de climatización, sistema de frenado ABS), en teléfonos y dispositivos móviles, refrigeradores, alarmas de casa, lavadoras, cámaras fotográficas, instrumentación industrial, equipos médicos, entre otros.

El elemento principal de un sistema embebido es un microprocesador, el cual es capaz de ejecutar ciertas instrucciones a determinada velocidad (controlada por una señal de reloj). Esta define la potencia de procesamiento, la cual es la máxima frecuencia de operación, definida en unidades de MIPS (millones de instrucciones por segundo).

Al conjunto de instrucciones y programas que realizan la función de ejecutar e interpretar rutinas de programación, que permiten comunicarse con las partes tangibles del equipo (hardware), se le conoce como software.

La combinación de software y hardware puede ser integrada, en muchos casos, por un circuito que realice la misma tarea, conocido como sistema embebido, una de las ventajas de los sistemas embebidos es su flexibilidad, al realizar alguna modificación en las líneas de código.

El programa que ejecuta un sistema embebido es por lo general elaborado en lenguajes como el ensamblador, ANSI C, C++ o Basic. Algunos sistemas embebidos tienen la capacidad de ejecutar sistemas operativos limitados, conocidos como RTOS (del inglés Real Time Operating Systems), que permiten que el procesador ejecute diversas tareas, asignándoles una prioridad y un orden de ejecución. Algunos de estos sistemas operativos son el MQX de Freescale, μ C/OS-II, FreeRTOS, Linux, Android entre los más conocidos.

Una de las particularidades de todo sistema embebido es el bajo consumo de energía, a tal punto que algunas de las aplicaciones pueden ser portátiles, por lo que el suministro de energía es por medio baterías (6).

2.2.1 APLICACIÓN DE LOS SISTEMAS EMBEBIDOS.

Los primeros computadores que contaba con circuitos integrados y que fueron considerados como los primeros sistemas embebidos surgieron en la década de los 60. Los primeros sistemas operativos en esa época eran capaces de realizar multitareas de hasta 8 tareas algo que en la actualidad sería normal, pero en su momento fue un gran avance.

El primer sistema embebido reconocido fue el sistema de guía de Apolo desarrollado por el laboratorio de desarrollo del MIT para las misiones Apolo hacia la luna. Cada vuelo hacia la luna tenía dos de estos sistemas. La función era manejar el sistema de guía inercial de los módulos de excursión lunar. En un comienzo fue considerado como el elemento que más riesgo presentaba en el proyecto Apolo. Este sistema de cómputo fue el primero en utilizar circuitos integrados y utilizaba una memoria RAM magnética, con un tamaño de palabra de 16 bits. El software fue escrito en el lenguaje ensamblador propio y constituía en el sistema operativo básico, pero capaz de soportar hasta ocho tareas simultáneas.

El primer sistema embebido producido en masa, fue el computador guía del misil norteamericano Minuteman II en 1962. El principal aspecto de diseño del computador del Minuteman, es que además de estar construido con circuitos integrados, permitía reprogramar los algoritmos de guía del misil

para la reducción de errores, y permitía realizar pruebas sobre el misil ahorrando así el peso de los cables y conectores.

Los sistemas embebidos desarrollados en masa permitieron ahorrar tiempo, esfuerzo y dinero. El modo de realizar tareas donde se necesitaba de mucho personal cambio. En la actualidad se utiliza solo una persona para dar la función a la máquina.

2.2.2 EJEMPLOS DE SISTEMAS EMBEBIDOS.

- Decodificadores y set-top boxes para la recepción de televisión. Cada vez existe un mayor número de operadores de televisión que aprovechando las tecnologías vía satélite y de red de cable ofrecen un servicio de televisión de pago diferenciado del convencional. En primer lugar, envían la señal en formato digital MPEG-2 con lo que es necesario un procesado para decodificarla y mandarla al televisor. Viaja cifrada para evitar que la reciban en claro usuarios sin contrato, lo que requiere descifrarla en casa del abonado. También ofrecen un servicio de televisión interactiva o web-TV que necesita de un software específico para mostrar páginas web y con ello un sistema basado en procesador con salida de señal de televisión.

- Sistemas radar de aviones. El procesado de la señal recibida o reflejada del sistema radar embarcado en un avión requiere alta potencia de cálculo

además de ocupar poco espacio, pesar poco y soportar condiciones extremas de funcionamiento (temperatura, presión atmosférica, vibraciones, etc.). (7)

Una vez mencionados los sistemas embebido su funcionamiento y algunos ejemplos pasaremos a la etapa de potencia donde se explicará el manejo de las partes de motor y sus componentes.

2.3 ETAPA DE POTENCIA.

Con la utilización de los sistemas embebidos cuyos voltajes de salida son de niveles reducidos (de 0 a 5 v), así como corrientes pequeñas (mA) son necesarios circuitos que puedan encender y apagar sistemas que manejan potencias mayores, a estos circuitos se les denomina como etapa de potencia.

Existen etapas de potencia de tipo electromecánicas como el relevador, los arrancadores, como dispositivos de tipo semiconductor como son los transistores y otros denominados tiristores.

Los tiristores son una familia de dispositivos semiconductores de cuatro capas que se utilizan para controlar grandes cantidades de corrientes

mediante circuitos electrónicos de bajo consumo de potencia. Dentro de la familia de los tiristores se encuentran los Triac y SCR's.

El triac (figura 9) es un dispositivo semiconductor de tres terminales que se usa para controlar el flujo de corriente promedio a una carga, con la particularidad de que conduce en ambos sentidos y puede ser bloqueado por inversión de la tensión o al disminuir la corriente por debajo del valor de mantenimiento.

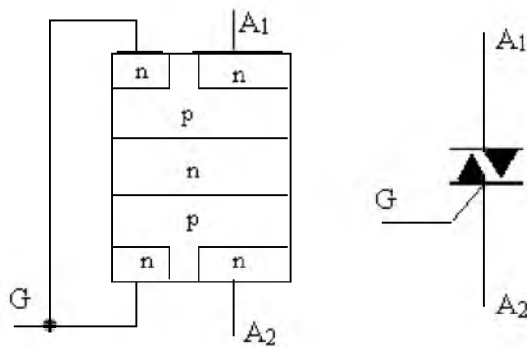


Figura 9. Construcción básica y símbolo del Triac.

El SCR (figura 10) es un dispositivo de cuatro capas posee tres terminales: ánodo, cátodo y puerta, presenta dos estados de operación: abierto y cerrado, como si se tratase de un interruptor.

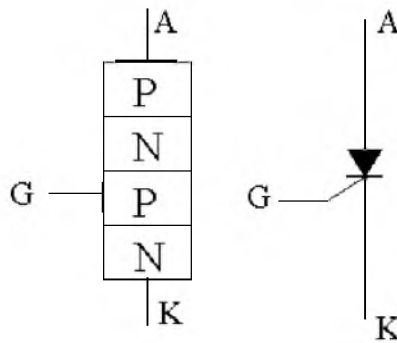


Figura 10. Construcción básica y símbolo del SCR.

2.3.1 RELEVADORES.

Los relevadores son interruptores que operan magnéticamente, se activa o desactiva dependiendo de la conexión. El encargado de activar o desactivar la conexión del relevador es un electroimán que es energizado el recibir un voltaje entre sus terminales.

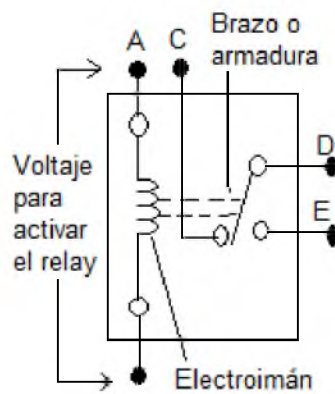


Figura 11. Funcionamiento del Relay.

Si el electroimán está activo atrae el brazo (armadura) y conecta los puntos C y D. Si el electroimán se desactiva, conecta los puntos D y E. De esta manera se puede conectar un dispositivo, cuando el electroimán está activo, y otro diferente, cuando está inactivo como se muestra en la figura 11.

2.3.2 TRANSISTOR.

El transistor (figura 12) es un dispositivo semiconductor que funciona como un elemento amplificador de señal. Otra función es permitir que las señales eléctricas puedan ser transmitidas a partir de señales de control. Al encontrarse el transistor abierto no permite el paso de la corriente y si se encuentra cerrado la corriente fluye.

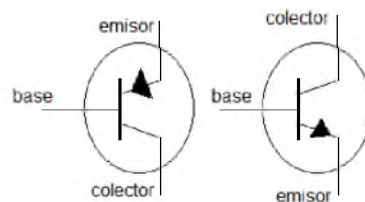


Figura 12. Transistor tipo PNP y transistor tipo NPN.

Los transistores se componen esencialmente de tres conexiones, cada uno encargado de una labor diferente; a continuación, se describe la función de cada uno de estos:

- Emisor. Deja entrar el flujo eléctrico al interior encapsulado del transistor.
- Base. Controla el flujo entre emisor y colector.
- Colector. Deja fluir la corriente una vez que ha sido controlada por la base.

Son dos las funciones principales de un transistor como parte de un circuito eléctrico.

- Como interruptor. Corta el flujo eléctrico a partir de una pequeña señal de control.
- Como amplificador. Recibe una pequeña señal eléctrica que, al salir del transistor, se habrá convertido en una de mayor magnitud.

En la figura 13 se muestra la forma de manejar la bobina de un relé mediante una salida TTL o CMOS utilizando un transistor de propósito general como elemento impulsor o driver.

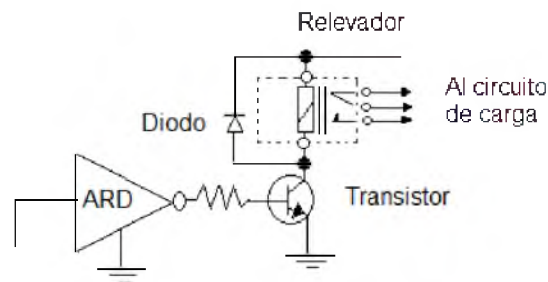


Figura 13. Interface digital de relevador con transistor.

Cuando la salida del inversor es alta, el transistor conduce y energiza la bobina del relé (k). Como resultado, el contacto normalmente abierto (NA) se cierra y normalmente cerrado (NC) se abre. Este efecto puede utilizarse para conectar y desconectar una carga externa, por ejemplo, un motor.

Cuando la salida del inversor es baja el transistor no conduce y no se energiza la bobina. En consecuencia, los contactos NA y NC permanecen en sus posiciones originales o retornan a estas.

Una vez que el motor se encuentre funcionando se requiere de un sistema de engranajes para regular la velocidad (reductor de velocidad), debido a que las máquinas no siempre funcionan de acuerdo con las velocidades que les ofrece el motor, (por ejemplo, hay maquinas que funcionan, a 1.800, 1.600 o 3.600 rpm).

2.2.3 ENGRANES.

El sistema de engranes, está constituido por un conjunto de ruedas o cilindros dentados, empleados para transmitir un movimiento giratorio o alternativo desde una parte de la máquina a otra. Estos cilindros son conocidos como engranes. Un conjunto de dos o más engranajes que transmite el movimiento de un eje a otro se denomina tren de engranaje. Los

engranajes se utilizan sobre todo para transmitir movimiento giratorio, pero usando engranes apropiados y piezas dentadas planas se pueden transformar el movimiento alternativo y giratorio o viceversa.

Existen diferentes tipos de engranes, se clasifican según la disposición de los ejes de rotación y los tipos de dentado. Algunos de ellos son:

- Ejes paralelos cilíndricos de dientes rectos, cilindro de dientes helicoidales y doble helicoidales.
- Ejes perpendiculares: helicoidales cruzados, cónicos de dientes rectos, cónicos de dientes helicoidales, cónicos hipoides, rueda y tornillo sinfín; su transmisión puede ser mediante cadena o polea dentada.

Los engranes cónicos de dientes rectos (figura 14) efectúan la transmisión de movimiento de ejes que se cortan en un mismo plano, generalmente en ángulo recto, por medio de superficies cónicas dentadas. Los dientes convergen en el punto de intersección de los ejes. Son utilizados para efectuar reducción de velocidad con ejes en 90° .

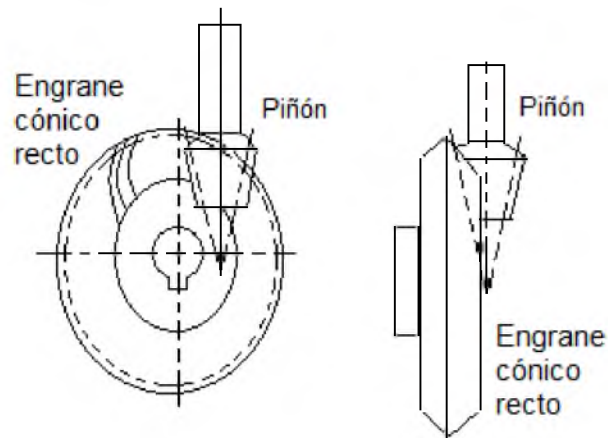


Figura 14. Engranes cónicos de dientes rectos.

El piñón va conectado a una leva que es una pieza con una forma circular que gira solidariamente con un eje. Se emplea para convertir el movimiento de giro en un movimiento alternativo. La leva, al girar, hace subir y bajar a la manivela que sirve como operador para hacer girar un eje con menos esfuerzo (figura 15); la manivela va conectada a una biela que se emplea para convertir un movimiento de giro en un movimiento de avance y retroceso, es decir, moviendo una biela hacia adelante y hacia atrás se puede conseguir que un cuerpo gire. La biela también puede servir para transmitir el movimiento giratorio de una rueda a otra. La biela y la manivela suelen utilizarse juntas, formando un sistema de biela-manivela. Uno de los ejemplos más conocidos es el de las ruedas de los trenes de vapor.

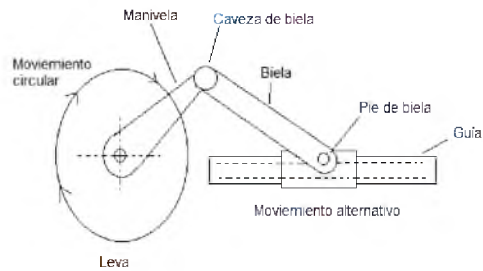


Figura 15. Mecanismo biela-manivela.

Para este diseño se utilizará los engranes cilíndricos rectos que se encuentra en la clasificación de engranes de ejes perpendiculares. Estos cuentan con las características necesarias para el diseño del sistema a desarrollar, son el tipo de engranaje que se utilizan generalmente para velocidades pequeñas y medias; su mecanismo será a través de un piñón.

Los conjuntos de piezas o elementos que ajustados entre si y empleando energía mecánica hacen un trabajo esto es un mecanismo

2.3.4 MOTORREDUCTORES.

Los motorreductores son sistemas formados por engranajes que hacen que los motores eléctricos funcionen a distintas velocidades.

Son creados a base de engranajes, mecanismos circulares o cerrados con geometrías diferentes, según su tamaño y la función en cada motor eléctrico.

Se implementan en cada una de las maquinas que usamos en el día a día, desde el más pequeño motorreductor capaz de cambiar y combinar velocidades de giro en un reloj de pulsera, cambiar velocidades en un automóvil, hasta enormes motorreductores capaces de dar tracción en buques de carga, molinos de cemento, grandes maquinas cavadoras de túneles o bien en molinos de cañas para la fabricación de azúcar.

La característica principal de un motorreductor es que transmiten la fuerza del motor del eje de entrada al eje de salida. Su transmisión está basada en si un reductor reduce o aumenta la velocidad del motor (velocidad de entrada).

En la mayor parte de aplicaciones, el reductor reduce la velocidad de giro transmitiendo simultáneamente pares significativamente más altos que los que el motor eléctrico por sí solo podría suministrar. El diseño del reductor determina si un motorreductor es adecuado para cargas ligeras, medias o pesadas y para tiempos de corta o larga duración.

El siguiente ejemplo muestra como encender o apagar un motorreductor.

```
/*Programa que prueba el encendido y apagado el motor*/  
int motorPin=6;                               //Esta conectado al pin número 6  
void setup(){
```

```

        pinMode(motorPin, OUTPUT);           //Se inicializa el pin.
    }
    void loop(){
        digitalWrite(motorPin, HIGH);       //Motor encendido
        delay(1000);                        //Retardo
        digitalWrite(motorPin, LOW);        //Motor apagado
        delay(1000);                        //Retardo
    }

```

2.4 MECANISMO.

El resultado de conjuntar elementos fijos y móviles que permiten reducir esfuerzo en la realización de trabajos y tareas es una serie de dispositivos y mecanismos llamados máquinas.

Un mecanismo es un dispositivo que transforma el movimiento en un patrón deseable. Por lo general desarrollan fuerzas muy bajas y transmite poca potencia. Una máquina en general contiene mecanismos que están diseñados para producir y transmitir fuerzas. No existe una clara línea divisoria entre mecanismos y máquina.

Los mecanismos son los elementos de una máquina destinados a transmitir y transformar las fuerzas y movimientos desde un elemento motriz, llamado

motor a un elemento receptor; permitiendo al ser humano realizar trabajos con mayor comodidad y menor esfuerzo.

En todo mecanismo resulta indispensable un elemento motriz que origine el movimiento (que puede ser un muelle, una corriente de agua, nuestros músculos, un motor eléctrico). El movimiento originado por el motor se transmite a través de los mecanismos a los elementos receptores (ruedas, brazos mecánicos) realizando, así, el trabajo para el que fueron construidos.

2.4.1 CLASIFICACIÓN DE LOS MECANISMOS.

Según su función los mecanismos se pueden clasificar en dos grupos:

Mecanismo de transmisión del movimiento

Estos mecanismos se encargan de transmitir el movimiento, la fuerza y la potencia producidos por un elemento motriz (motor) a otro punto, sin transformarlo.

1. Mecanismo de transmisión lineal.
 - Palancas. Primer grado, segundo grado y tercer grado.
 - Poleas. Polea fija, polea móvil y polea compuesta.
2. Mecanismos de transmisión circular.
 - Sistemas de ruedas o poleas. Ruedas de fricción, sistemas de poleas con correas, relación de transmisión.

- Engranés y sistemas de engranes.
- Tornillo sin fin-corona.

Mecanismos de transformación de movimiento.

Son aquellos que cambian el tipo de movimiento, de lineal a circular (o a la inversa), o de alternativo a circular (o a la inversa).

- Conjunto manivela-torno.

Una manivela es una barra unida a un eje al que hace girar. La fuerza que se necesita para girar este eje es menor que el que haría falta aplicar directamente. Consiste en un cilindro horizontal (figura 16) sobre el que se enrolla (o desenrolla) una cuerda o cable cuando le comunicamos un movimiento giratorio a su eje.

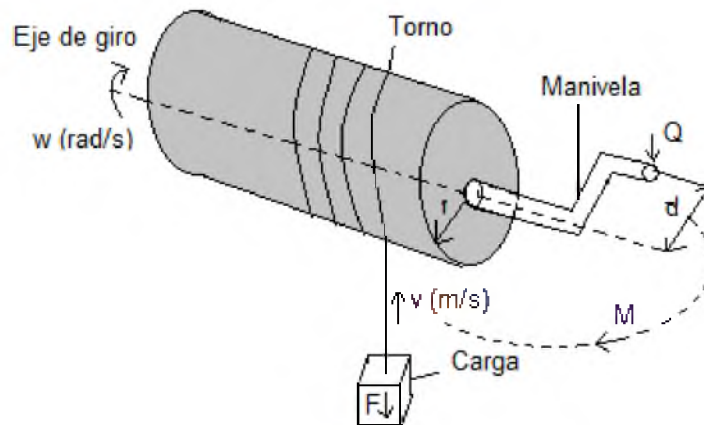


Figura 16. Sistema manivela-torno para elevación de cargas.

Se emplea para:

Obtención de un movimiento lineal a partir de uno giratorio. En grúas (accionado por un motor eléctrico en vez de una manivela), barcos (para recoger las redes de pesca, izar o arriar velas, levar anclas), pozos de agua (elear el cubo desde el fondo).

Obtención de un movimiento giratorio a partir de uno lineal. En peonzas (trompos), arranque de motores fuera-borda, accionamiento de juguetes sonoros para bebés.

- Piñón cremallera.

Este mecanismo (figura 17) está formado por una rueda dentada (piñón) que engrana con una barra también dentada llamada cremallera. Permite transformar el movimiento circular del piñón en movimiento rectilíneo en la cremallera (o viceversa). Se emplea en el sistema de dirección de los automóviles, columnas de taladradoras, trípodes, sacacorchos, puertas de garajes.

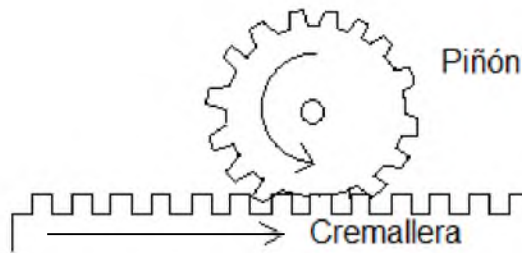


Figura 17. Sistema piñón-cremallera.

- Tornillo-tuerca.

Mecanismo compuesto por un eje roscado (husillo) y una tuerca con la misma rosca que el eje. Si se gira la tuerca, ésta se desplaza linealmente sobre el husillo y viceversa.

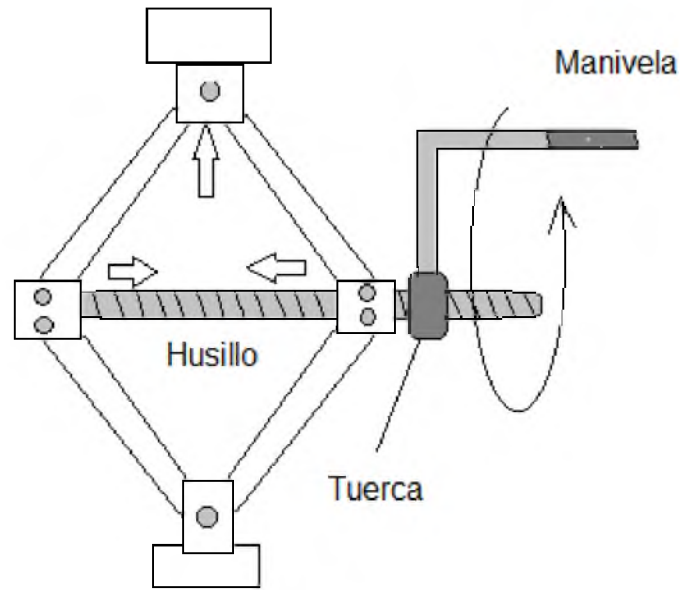


Figura 18. Gato de un coche (ejemplo del mecanismo husillo-tuerca).

Así por ejemplo en el gato de los coches (figura 18), podemos conseguir un movimiento lineal (perpendicular al suelo) a partir de un movimiento circular (al girar la manivela). Otras aplicaciones son las uniones, grifos, compases de rosca, tapones de rosca.

- Excéntrica y leva.

Una rueda excéntrica es una rueda que gira sobre un eje que no pasa por su centro (figura 19). Estos sistemas se componen de una pieza de contorno especial (leva) o de una rueda excéntrica que recibe el movimiento rotativo a

través del eje motriz y de un elemento seguidor que está permanentemente en contacto con la leva por la acción de un muelle.

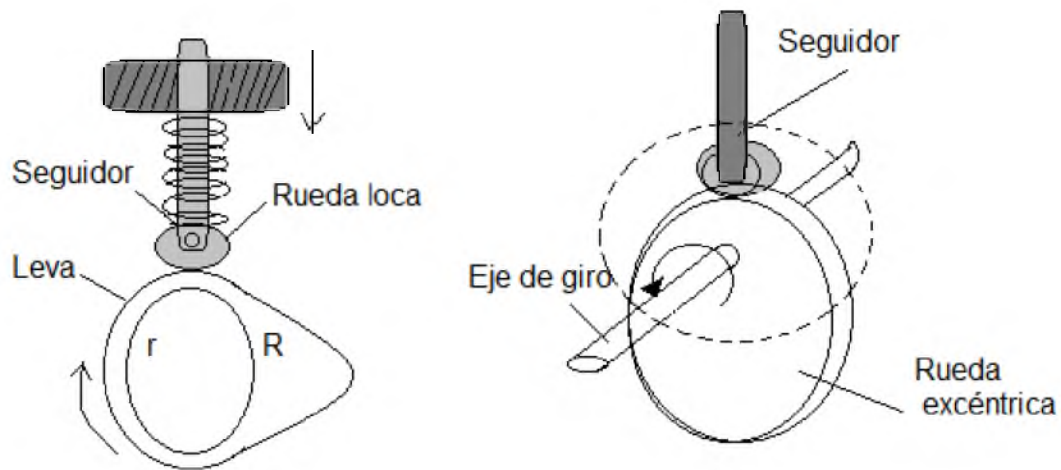


Figura 19. Leva y excéntrica.

Ambos son mecanismos que permiten convertir un movimiento rotativo en un movimiento lineal (pero no viceversa). De este modo, el giro del eje hace que el contorno de la leva o excéntrica mueva o empuje al seguidor que realizará un recorrido ascendente y descendente (movimiento lineal alternativo).

Se emplea en cerraduras, carretes de pesca, depiladoras, motores de automóviles.

- Biela-manivela

Está formado por una manivela que tiene un movimiento circular y una barra llamada biela (figura 20). La biela está unida con articulaciones por un extremo a la manivela, y por el otro a un sistema de guiado (un pistón o émbolo encerrado en unas guías) que describe un movimiento rectilíneo

alternativo en ambos sentidos. Sirve para transformar un movimiento circular en lineal o viceversa, ya que es reversible. Se empleó en la locomotora de vapor, empleándose en motores de combustión interna, herramientas mecánicas, máquinas de coser.

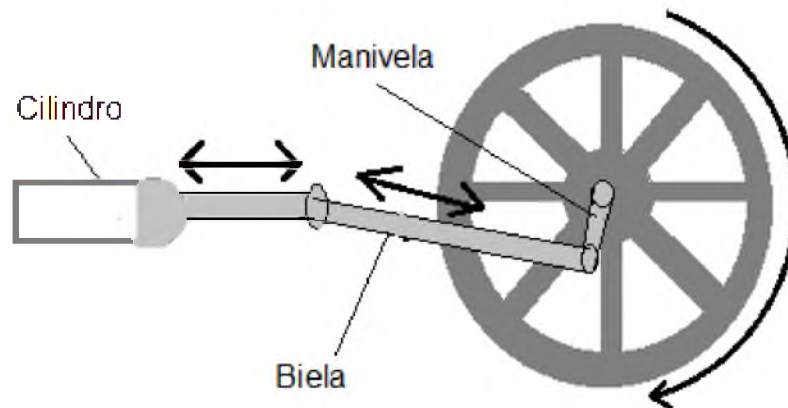


Figura 20. Sistema biela-manivela.

- Cigüeñal.

En el caso de los motores se colocan una serie de bielas en un mismo eje acodado, donde cada uno de los codos del eje hace las veces de manivela (figura 21). A este conjunto se le denomina cigüeñal. El cigüeñal transforma el movimiento de las bielas en un movimiento de rotación en el eje.

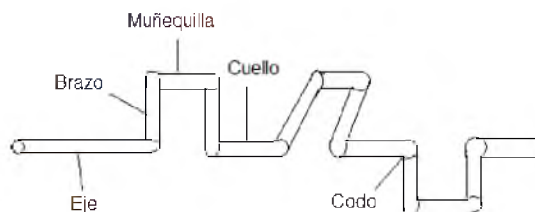


Figura 21. Cigüeñal.

3. DESARROLLO.

En esta sección se presentan los elementos eléctricos, electrónicos mecánicos y de programación para el correcto mantenimiento del sistema propuesto. Sensores, actuadores, etapas de potencia, mecanismos, programas aquí son explicados con detalle y precisión, así como el diseño de tarjetas electrónicas y piezas mecánicas.

3.1 PRUEBA DE SISTEMA DE PESAJE.

La celda de carga es uno de los elementos más importantes de una báscula electrónica. Esta se encarga de traducir la fuerza en una señal de voltaje (celda de carga analógica) o en un valor digital (celda de carga digital). La celda de carga analógica con galgas extensométricas es la más utilizada (figura 22).



Figura 22. Celda de carga.

La celda de carga se divide en dos partes: un elemento rígido y una galga extensométrica. El elemento rígido consiste en un metal que sufre una deformación, conforme se le aplica una fuerza. Este metal se calcula para soportar un rango de fuerza (que va desde cero hasta la capacidad máxima).

La fuerza aplicada puede ser tensión, compresión o ambos. La deformación se realiza en la "parte elástica", esto es lo que limita la capacidad de una celda de carga. Al sobrepasar la capacidad elástica del metal, la celda sufre una deformación permanente (no detecta cero fuerzas), de la misma forma que un resorte no regresa a su forma inicial al estirarse más allá de sus límites.

Al elemento rígido se le adhieren galgas extensométricas; y se conectan en un arreglo de puente de wheatstone, de tal forma que cuando se flexionan emiten una señal de voltaje (mili volts) proporcional a la fuerza aplicada.

Para poder ser utilizada dicha señal se lleva de la celda a un convertidor análogo-digital. Una vez que se tiene en forma de un valor digital, este valor se multiplica por un factor de conversión para ser representado en kg, lb, u otras unidades de peso. El principio de las basculas está fundamentado en la ecuación Fuerza = Masa x aceleración ($F = m * a$) de este modo las básculas miden la fuerza que genera un objeto y la aceleración es una constante (la

gravedad de la tierra) por lo tanto; se puede decir que la Masa es directamente proporcional a la Fuerza.

3.1.1 FORMA DE INSTALAR UNA CELDA DE CARGA.

Para instalar la celda de carga es importante resaltar que debe existir una distancia entre la celda y la base donde se desea colocar, dejando la parte central libre como se muestra en la figura 23. Así mismo se debe considerar que la dirección de la flecha grabada en la celda indica la dirección del peso a medir.

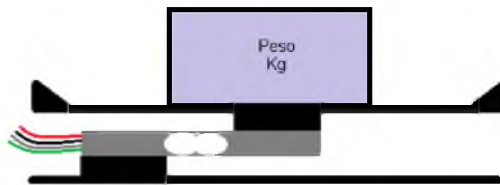


Figura 23. Esquema de instalación de la celda de carga.

3.1.2 TRANSMISOR DE CELDA DE CARGA.

En la figura 24 se muestra el módulo HX711 el cual es una interface entre la celda de carga y el microcontrolador. Internamente se encarga de la lectura del puente wheatstone, convirtiendo la lectura analógica en digital con su convertidor A/D interno de 24 bits.

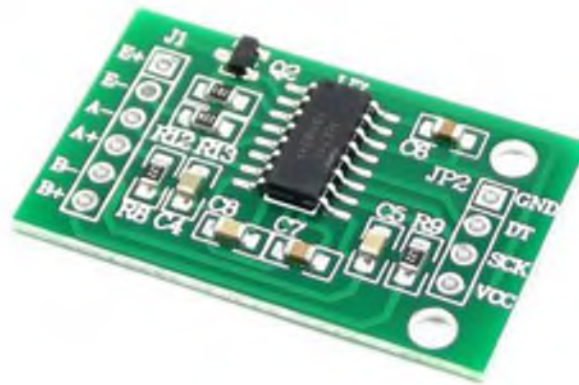


Figura 24. Módulo HX711.

El transmisor de celda de carga se comunica con el microcontrolador mediante los siguientes pines:

El Pin PD_SCK: Línea de la señal de reloj.

El pin DOUT: Línea de transmisión de datos.

Estos funcionando en un protocolo de comunicación serial. La siguiente grafica ejemplifica un periodo de conversión que utiliza este protocolo.

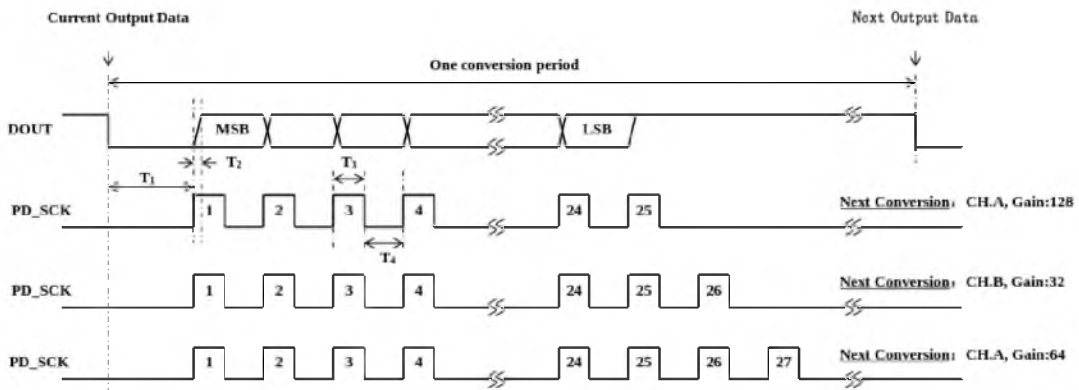


Figura 25. Periodo de conversión.

El pin de salida digital DOUT se encuentra a nivel alto cuando los datos no están listos para ser transmitidos, mientras que la entrada del reloj serial PD_SCK debe estar a nivel bajo. Cuando DOUT pasa a nivel bajo, indica que los datos están listos para su transmisión, cada impulso PD_SCK desplaza un bit a la vez, hasta que todos los 24 bits son desplazados por la línea.

En el pulso número 25 de la línea PD_SCK, la línea de datos DOUT pasará nivel alto. La entrada y la selección de ganancia son controladas por el número de impulsos PD_SCK de entrada. Los impulsos de reloj PD_SCK no deben ser menores de 25 o más de 27 dentro de un período de conversión, para evitar causar un error de comunicación serie.

3.1.3. CONEXIONES ENTRE LA CELDA DE CARGA, MÓDULO HX711 Y ARDUINO.

En la tabla 1 se muestra la forma en cómo se realizan las conexiones entre la celda de carga y el módulo HX711.

Celda De Carga	Módulo HX711
Cable Rojo	Pin E+
Cable Negro	Pin E-
Cable Verde	Pin A-
Cable Blanco	Pin A+

Tabla 1. Conexión entre la celda de carga y el módulo HX711.

En la tabla 2 se muestra cómo se realizan las conexiones entre el módulo HX711 y Arduino.

Módulo HX711	Arduino
Pin GND	Pin GND
Pin DT	Pin A1
Pin SCK	Pin A0
Pin VCC	Pin 5V

Tabla 2. Conexión entre el módulo HX711 y Arduino.

La conexión física se muestra en la figura 26.

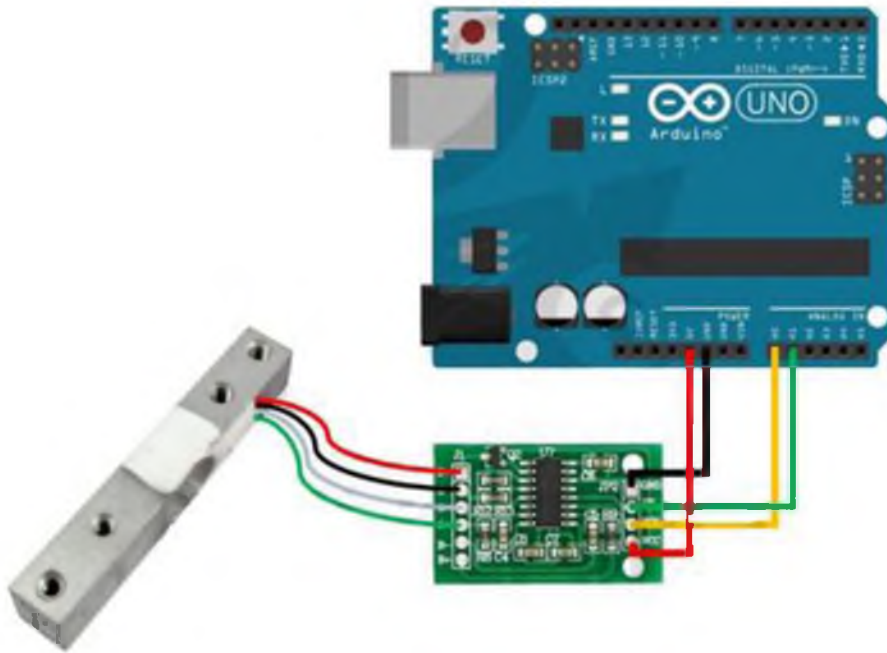


Figura 26. Conexión entre la celda de carga, el módulo HX711 y Arduino.

Con los dispositivos conectados, la celda de carga, el módulo HX711 y Arduino, debe calibrarse la báscula. Este proceso consiste en establecer el valor de la escala que se usará mediante un factor de conversión. El factor de conversión es una constante de proporcionalidad, mediante la cual un valor de lectura en volts se convierte en un valor en unidades de peso.

Para calibrar la báscula debe colocarse sobre la balanza un objeto con peso conocido, y cargar al Arduino el programa, (sketch) que a continuación se muestra:

*/*PROGRAMA PARA CALIBRAR LA BALANZA*/*

```
#include "HX711.h"           // Las librerías hacen referencia al módulo  
HX711  
  
#define DOUT A1           //para llevar acabo la conexión entre la celda de  
  
#define CLK A0           //carga con el microcontrolador.  
  
  
HX711 balanza(DOUT, CLK);   //Constructor del módulo HX711  
  
void setup() {  
  
    Serial.begin(9600); //Abre el puerto serie y fija la velocidad para la trans.  
  
    Serial.println(balanza.read()); // Espera hasta que el dispositivo esté listo  
  
                                //y devuelve la lectura del ADC del HX711.  
  
    Serial.println("No ponga ningún objeto sobre la balanza");  
  
    Serial.println("Destarando...");  
  
    balanza.set_scale();     /* Establece el valor de la escala, que es el  
  
                            factor de conversión para convertir valor de lectura  
  
                            en un valor con unidades de peso. La escala  
  
                            por defecto es 1*/  
  
    balanza.tare(20);       //El peso actual es considerado Tara.
```

```

    Serial.println("Coloque un peso conocido:");
}

void loop() {

    Serial.print("Valor de lectura: ");

    Serial.println(balanza.get_value(10),0); //Devuelve el peso actual restando

                                         // el peso de tara

    delay(100);
}

```

Al iniciar el programa debe correr sin peso, primero calcula la tara que es el peso del contenedor o empaque sin incluir el peso del producto. Después de poner el peso del producto en la balanza el LCD se mostrará las lecturas del peso. Estas lecturas muestran el peso sin escalar, por lo que aparecen valores grandes.

Con los datos se calcula el valor de la escala con la siguiente formula:

$$\text{Escala} = \frac{\text{Valor de lectura}}{\text{Peso real}}$$

El valor del peso del producto deberá estar en las unidades con las que trabajará la balanza, por ejemplo: Kg o g para Kilogramo o gramos respectivamente.

Entonces, el valor de la escala a usar es:

$$Escala = \frac{1757721}{4} = 439430.20$$

El valor obtenido se sustituye en el sketch que se usará para pesar.

```
/* PROGRAMA PARA PESAR*/

#define DOUT A1

#define CLK A0

HX711 balanza(DOUT, CLK);

void setup() {

    Serial.begin(9600);

    Serial.println(balanza.read());

    Serial.println("No ponga ningún objeto sobre la balanza");

    Serial.println("Destarando...");

    balanza.set_scale(439430.25); // Establece la escala

    balanza.tare(20); //El peso actual es considerado Tara.

    Serial.println("Listo para pesar");
```

```

}

void loop() {

    Serial.print("Peso:t");

    Serial.print(balanza.get_units(20),3); /*Devuelve el valor actual restado del
                                           Peso de tara y dividido por la escala*/

    Serial.println("kg");

    delay(500);

}

```

El Arduino debe estar encendido antes de colocar el objeto que se desea pesar, de lo contrario el peso que esté sobre la balanza se considerará como tara.

3.1.4. CONEXIÓN DEL LCD (DISPLAY) Y ARDUINO.

Los Display's o LCD's (Liquid Crystal Display; Pantalla de Cristal Líquido) *son* dispositivos diseñados para mostrar información. Las pantallas LCD vienen unidas a una placa de circuito y poseen pines de entrada/salida de datos.

Existen diferentes tamaños de LCD, en la figura 27 se muestra un LCD de 2 filas por 16 columnas, esto quiere decir que se puede imprimir caracteres en 2 filas, en las cuales caben 16 caracteres.

El Arduino es capaz de utilizar las pantallas LCD para desplegar información. La librería Liquid Cristal se encarga de enviarle los datos al circuito integrado de una pantalla LCD desde Arduino para ser visualizados.



Figura 27. CD o Display.

A continuación se describe la organización de los pines que integran el LCD:

IMAGEN	PIN	ABREV	DESCRIPCIÓN
	1	GND	Negativo o <u>ground</u> .
	2	VDD	Alimentación principal de la pantalla 5 voltios.
	3	VE	Contraste de la pantalla: Se conecta a un potenciómetro de 10k <u>ohms</u> .
	4	RS	Selector de registro: El microcontrolador le comunica al LCD, si quiere mostrar caracteres, cambiar posición del cursor o borrar la pantalla.
	5	RW	Lectura/escritura: Siempre estará conectado a GND para que escriba.
	6	E	<u>Enable</u> : Habilita la pantalla para recibir información.
	7	D0	No se utilizan, la pantalla tiene un bus de datos de 8 bits.
	8	D1	
	9	D2	
	10	D3	
	11	D4	Sirven para establecer las líneas de comunicación por donde se transfieren los datos.
	12	D5	
	13	D6	
	14	D7	
	15	VDD	Pines de led de la luz de fondo de la pantalla.
	16	GND	

Figura 28. Esquema de los pines del LCD o Display.

En el diagrama 1 se muestra las conexiones entre el LCD y Arduino.

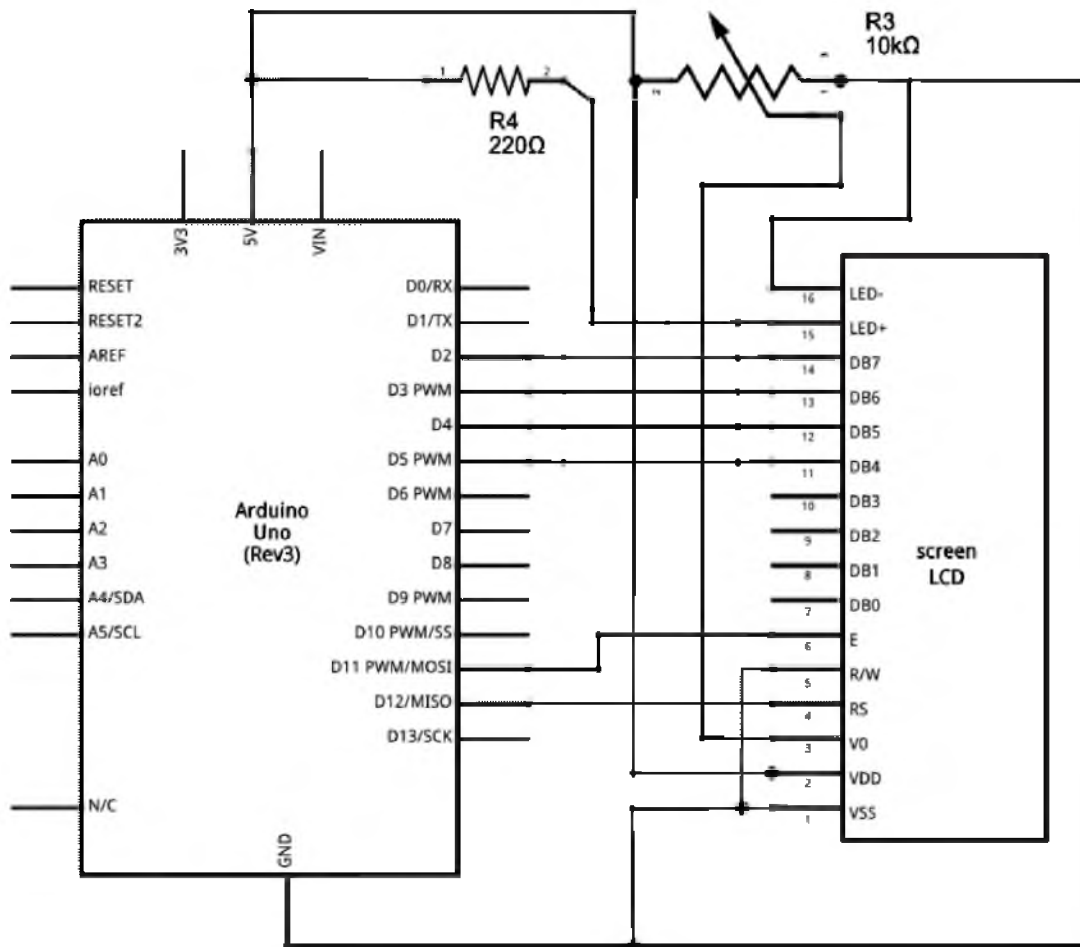


Diagrama 1. Conexión entre el LCD y Arduino.

Para probar el funcionamiento del LCD se desarrolló un programa de prueba en el ARDUINO. Para realizar las conexiones necesarias se utilizó el

siguiente material: protoboard, Arduino UNO, cables de protoboard, potenciómetro y display o LCD de 16x2.

La conexión del potenciómetro de ajuste se realiza conectando un extremo del potenciómetro a GND y el otro a VDD mientras que, el del centro se conectará al pin 3 del LCD. Una vez realizada la conexión, al girar el potenciómetro se modificará el contraste del cristal líquido mostrando en la posición de cada carácter un bloque oscuro.

Los pines: Selector de Registro (RS), pin de Lectura y Escritura (RW) y el pin que habilita la pantalla (E) así como la conexión de los 4 pines de datos D7, D6, D5, y D4 se describe a continuación:

RS, LCD pin 4 a Arduino pin 7

RW, LCD pin 5 a GND

E, LCD pin 6 a Arduino pin 8

DB7, LCD pin 14 a Arduino pin 12

DB6, LCD pin 13 a Arduino pin 11

DB5, LCD pin 12 a Arduino pin 10

DB4, LCD pin 11 a Arduino pin 9

Una vez establecidas las conexiones de forma física el siguiente paso es cargar el sketch al Arduino.

El siguiente código gestiona la comunicación entre el Arduino y el LCD.

```
/*PROGRAMA PARA CONECTAR ARDUINO AL DISPLAY*/

/*La biblioteca de LiquidCrystal es la encargada de realizar las instrucciones de bajo nivel.*/

#include <LiquidCrystal.h>

// Inicializar la biblioteca con los números de los pines de interfaz

LiquidCrystal lcd ( 12 , 11 , 5 , 4 , 3 , 2 );

Void setup ( ) {

    lcd.begin(16, 2);      // configura el número de columnas y filas de la LCD

    lcd.noDisplay();     //Se apaga el display durante 1000ms

    delay(1000);

}

Void loop ( ){

    lcd.display();      // Enciende el LCD

    delay(500);

    lcd.print("Hola");  // imprimir la palabra hola durante 500ms

}
```

Con la finalidad de obtener una conexión del LCD permanente, se realiza un diseño PCB para una placa fenólica.

3.1.5. DISEÑO DE LA TARJETA DE CIRCUITOS IMPRESA.

Un diseño de PCB (printed circuit board, Tarjeta de Circuitos Impresos), se realiza en dos etapas. Primeramente, se diseña un diagrama esquemático de conexiones, para después generar la PCB propiamente dicha. Esta se compone por “pistas” trazadas sobre una placa fenólica (sobre el lado conductor), equivalente a los cables.

La PCB proporciona las interconexiones entre los dispositivos, así como conexiones fijas, mayor inmunidad a ruido, menor probabilidad de fallas, entre otras.

Para realizar los diseños PCB, es necesario un software especializado. Dentro de las opciones existentes en el mercado, se encuentra el software Eagle, presentando la ventaja de poder ser instalado con una licencia tipo “free”. Este tipo de licencia nos permite utilizar la mayoría de las opciones sin costo alguno.

Eagle cuenta con las siguientes características:

- Editor de esquemas.
- Editor de diseño.

- Módulos de biblioteca.
- Editor de interfaces de usuario.
- Servicio de consulta gratuito con expertos dedicados.
- Compatible con Windows ®, Linux ®, Mac ®.
- Varios idiomas disponibles para trabajar.
- Funciones personalizadas, tales como: secuencias de instrucciones individuales, simulación, los datos de exportación e importación.

3.1.6 CONEXIÓN ENTRE EL MÓDULO HX711, EL DISPLAY Y ARDUINO.

En el diagrama 2 (diagrama esquemático) se muestran las conexiones entre el módulo HX711, el LCD y Arduino, al tener estas conexiones a partir del diagrama esquemático se genera la PCB.

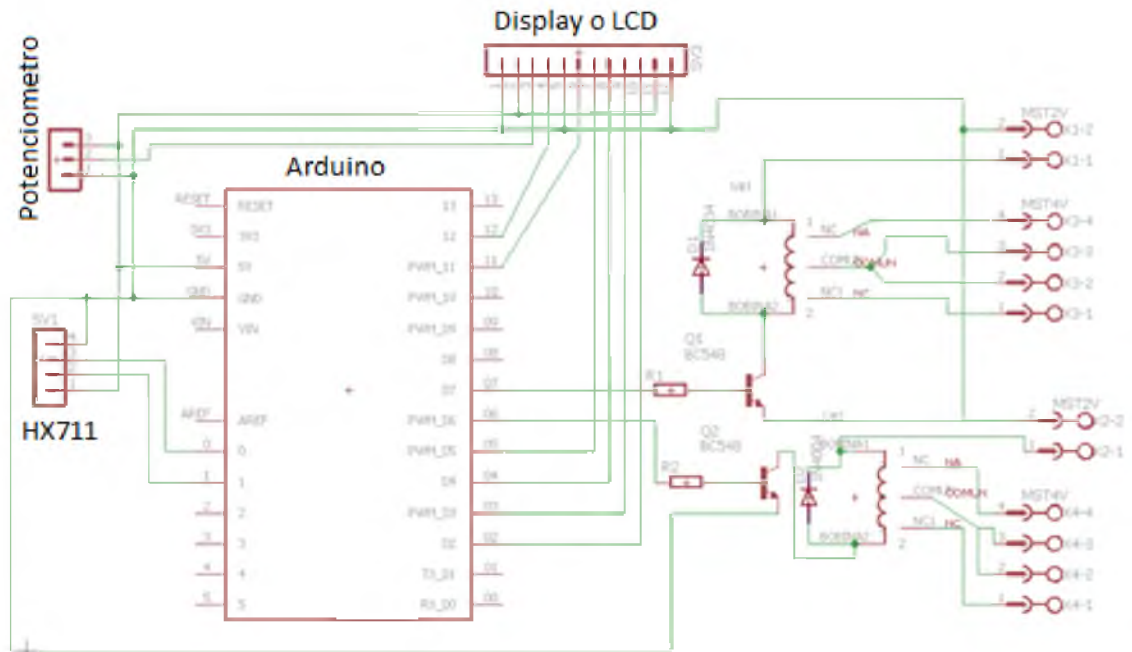


Diagrama 2. Diagrama esquemático para la conexión entre Arduino, el LCD y el Módulo HX711.

Las net, que son el trazo de cobre, sirven para realizar la conexión entre dos o más elementos. En el diagrama 3 se presenta la tarjeta PCB donde se visualizan las net.

Para asegurar el buen funcionamiento del circuito construido es necesario tomar en cuenta la geométrica y los espaciamientos entre las líneas, nets y demás elementos del PCB.

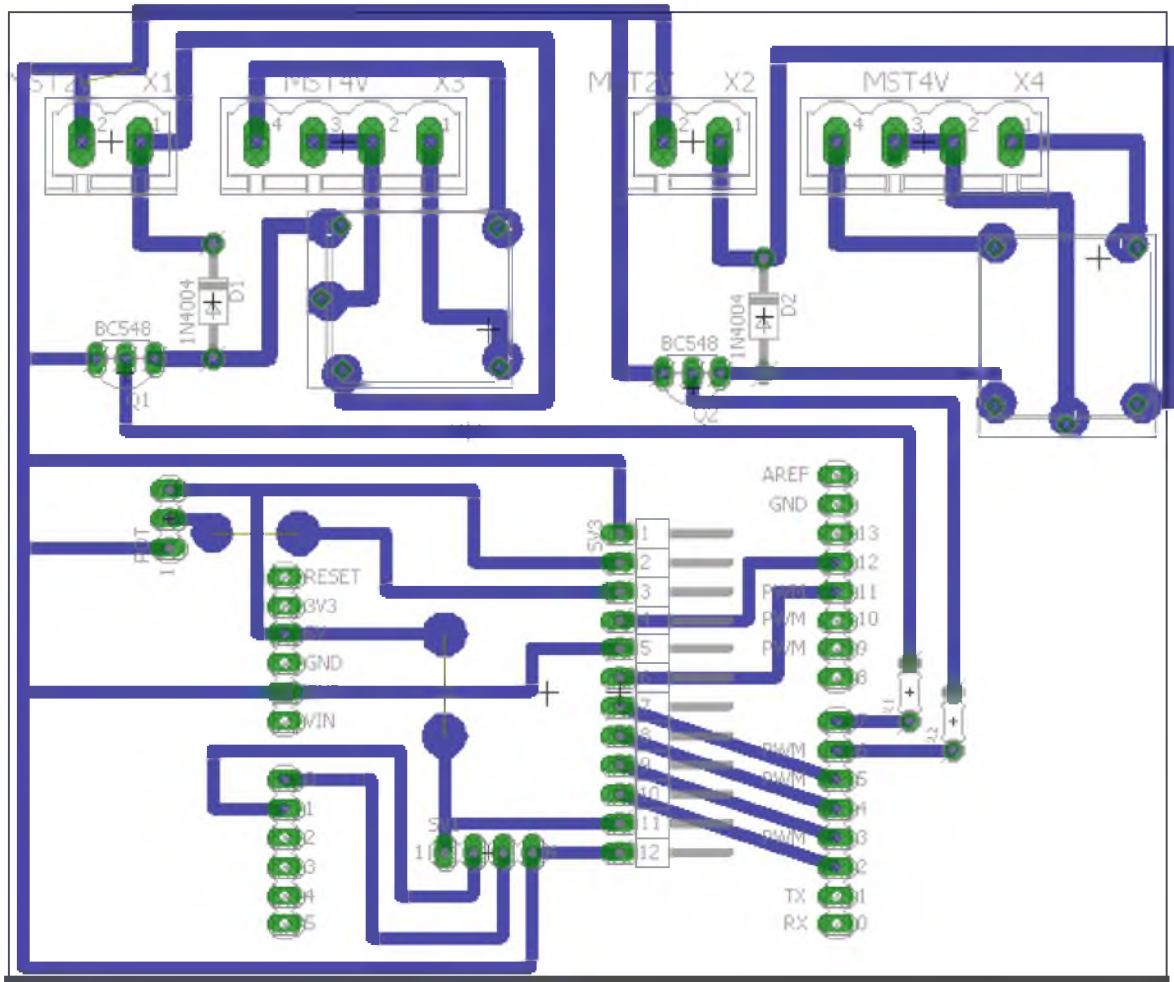


Diagrama 3. Tarjeta de circuito impresa diseñada con el software EAGLE.

El diseño presentado en el diagrama 3, debe ser realizado físicamente transfiriendo el diseño anterior a este procedimiento se realiza una placa fenólica. Teniendo la placa lista con el circuito impreso se montan los elementos; Arduino, el módulo HX711, display, el relevador, potenciómetro, la salida hacia el motorreductor y los botones.

3.2 DISEÑO MECÁNICO.

SolidWorks es un software de diseño mecánico en tercera dimensión (3D) que utiliza un entorno gráfico. Su primera versión fue lanzada al mercado en 1995 por su fundador Jon Hirschtick, con el propósito de hacer la tecnología CAD (diseño asistido por computadora) más accesible.

Este software está conformado por tres módulos: pieza, ensamblaje y dibujo:

- Módulo pieza. Constituye un entorno de trabajo donde se pueden diseñar modelos mediante el empleo de herramientas de diseño. El conjunto de funciones e iconos permiten crear modelos tridimensionales (3D) partiendo de geometrías de croquis bidimensionales (2D) y obtener sólidos, superficies, estructuras metálicas, piezas de chapas, piezas multicuerpo, entre otras.
- Módulo de ensamblaje. Está formado por un entorno de trabajo preparado para crear ensambles mediante la inserción de los modelos 3D creados en el módulo de piezas.
- Módulo dibujo. Permite obtener proyecciones ortogonales, secciones y cortes, perspectivas, acotaciones, lista de materiales, vistas explosionadas entre otras funciones.

El software de diseño SolidWorks provee las herramientas que ayudan a innovar y diseñar en menos tiempo y a un menor costo. Se puede aplicar en diseño mecánico, sistemas mecatrónicos, cinemática, ensamble de robótica, diseño automotriz y aeroespacial, biomecánica, diseño de dispositivos.

Sirve para diseñar piezas mecánicas en 3D, evaluar ensambles de varias piezas y producir dibujos de fabricación para el taller, se pueden manejar los datos de diseño en su sistema de administración y llevar un control de las versiones de dibujos. (2)

La característica que hace que SolidWorks sea una herramienta competitiva, ágil y versátil es su capacidad de ser paramétrico, variacional y asociativo, además de usar las Funciones Geométricas Inteligentes y emplear un Gestor de Diseño que permite visualizar, editar, eliminar y actualizar cualquier operación realizada en una pieza de forma bidireccional entre todos los documentos asociados.

Se aplica para la creación de tubos, cañerías, cables eléctricos o colectores de cable de forma automática y precisa; es útil en el diseño de máquinas, instalaciones o cualquier elemento que emplee enrutamientos. Convierte las rutas 2D en 3D de forma automática y ofrece la lista de materiales creados, así como información sobre sus conexiones.

Actualmente la herramienta de diseño mecánico SolidWorks se emplea en sectores tan diversos como el Aeroespacial, Automoción, Defensa, Educación y Universidades, Ingeniería civil, Ingeniería industrial, Ingeniería marina, Ingeniería mecánica Investigación y desarrollo y en Simulación, entre otros campos. (1)

Gracias al entorno de trabajo que maneja SolidWorks es posible diseñar las piezas necesarias para después realizar el ensamblaje para obtener la máquina a desarrollar. Con el diseño de la máquina diseñada en SolidWorks se pueden realizar las pruebas para verificar el funcionamiento antes de construir el prototipo.

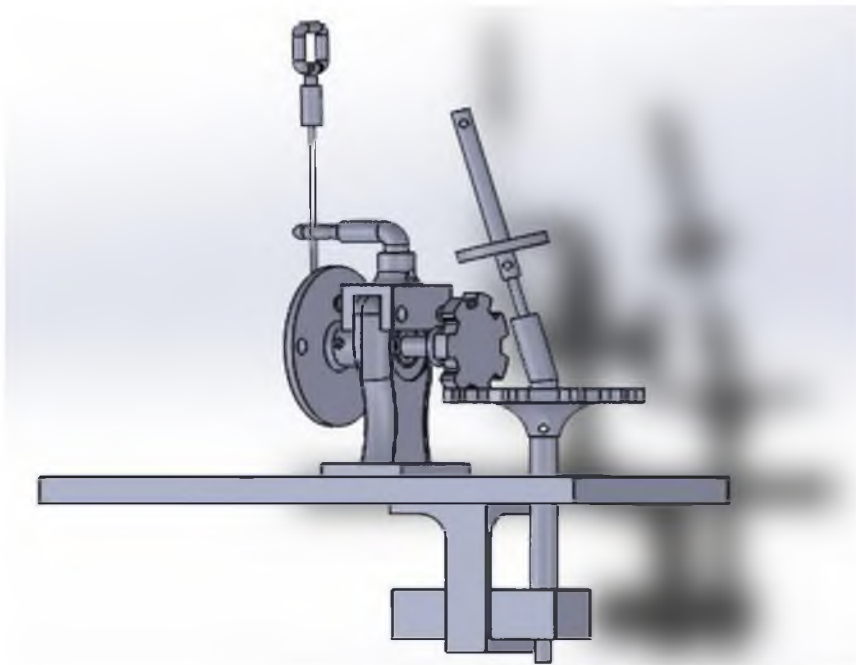


Figura 29. Prototipo de máquina.

3.3 FUNCIONAMIENTO DEL PROTOTIPO.

El mecanismo empieza a funcionar cuando gira el disco engrane. Este movimiento será transmitido por medio del engrane al tuvo, el cual realizara el movimiento para el enrollado del producto figura 29.

A continuación, se describen las piezas diseñadas en SolidWorks:

El engrane recto se muestra en la figura 30. Este es un sistema de rueda que poseen dientes que encajan entre sí, de este modo una rueda transmite el movimiento al siguiente engrane.

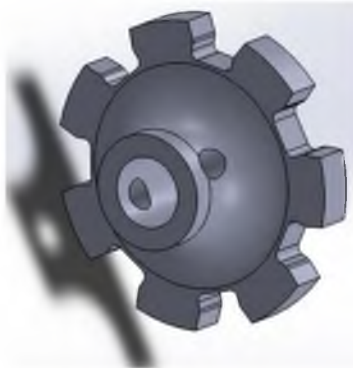


Figura 30. Engrane recto.

Los engranajes adoptan distintas formas, pueden ser cilíndricos, de dientes rectos, helicoidales, cónicos o como en este caso, se muestra en la figura 31. El engrane recto entran en los orificios del disco formando así un tren de engranes (sistema de dos o más engranes). Todos los dientes de los

engranajes en contacto han de tener la misma forma y tamaño para que encajen.

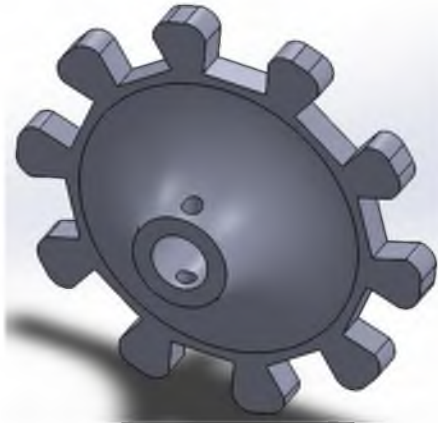


Figura 31. Engrane recto.

La relación de transmisión entre las velocidades de giro depende del tamaño relativo de los engranajes y de la relación entre el número de dientes.

El movimiento del tren de engranajes forma parte de lo que se conoce como biela manivela la cual se muestra en la figura 32.

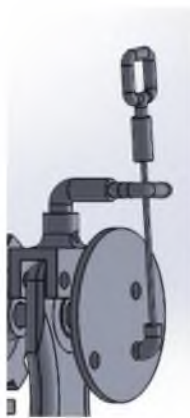


Figura 32. Biela-manivela.

Este mecanismo está formado por una manivela que tiene un movimiento circular y una barra llamada biela. La biela está unida con articulaciones por un extremo a la manivela, y por el otro a un sistema de guiado figura 33 que describe un movimiento vertical.



Figura 33. Sostén de biela (sistema de guiado).

Al girar la rueda, la manivela transmite el movimiento circular a la biela, que experimenta un movimiento de vaivén provocando que la biela sube y baja. El engrane recto provoca el movimiento a la biela manivela, mientras que el sostén de la biela lo mantiene en una sola dirección.



Figura 34. Engrane unido a un soporte angular.

De igual forma el engrane recto, al mantenerse en movimiento, hará girar el disco del engrane. A este engrane estará unido un tubo inclinado, figura 34, donde ira enrollado el producto.

Teniendo el diseño terminado, en SolidWorks, el archivo se convierte en formato STL para poder visualizarlo en el programa CURA. Este no es el único formato que maneja el programa sin embargo es el que se utilizó para este diseño.

Cura es un software encargado de procesar los archivos de diseño en 3D y generar un archivo GCode con la información de cada capa que será depositada por el extrusor de una impresora 3D.

Dispone de opciones avanzadas para el control del relleno y las velocidades de movimiento, impresión y retracción. Ha sido desarrollado por Ultimaker, es gratuito y de código abierto. Se adapta a la impresora que se está utilizando.

Una impresora 3D es una máquina capaz de imprimir figuras con volumen (ancho, largo y alto) a partir de un diseño hecho por computadora.

Las impresoras 3D utilizan múltiples tecnologías de fabricación (ABS, PLA, entre otros).

En la siguiente figura se muestran 3 imágenes, en las cuales se presentan el proceso de diseño de una pieza con una impresora 3d. En la primera se realiza un bosquejo en papel, intentando respetar en la manera de lo posible las dimensiones del objeto real. Después, con un programa de CAD (SolidWorks) se diseña ese objeto en la computadora, el que se presenta en la segunda imagen. En este diseño el objeto debe ser construido con las medidas exactas necesarias. Por último, se separa este objeto en capas para imprimir capa por capa en la impresora de 3d (tercera imagen).



Figura 35. Proceso de diseño de una pieza con una impresora 3d.

Las impresoras 3d utilizan 3 tipos de formas de imprimir, lo que da lugar a 3 tipos de impresoras 3d diferentes. Aunque todos los tipos de impresoras 3d utilizan el proceso aditivo, son formas diferentes de construir el objeto.

3.4 PROGRAMACIÓN E IMPLEMENTACIÓN DEL SISTEMA.

El diseño de las piezas se realizó en SolidWorks tomando en cuenta que las medidas deben de ser precisas para que cada una de las piezas coincidan de manera precisa una entre otra.

Las piezas que componen esta máquina diseñada por medio de SolidWorks son las siguientes:

Engrane base: está compuesta por 10 dientes cada diente mide 8.6mm de ancho, 5.5 mm de alto, entre cada diente hay una distancia de 10.10mm, tiene un radio de 32.5mm figura 36.

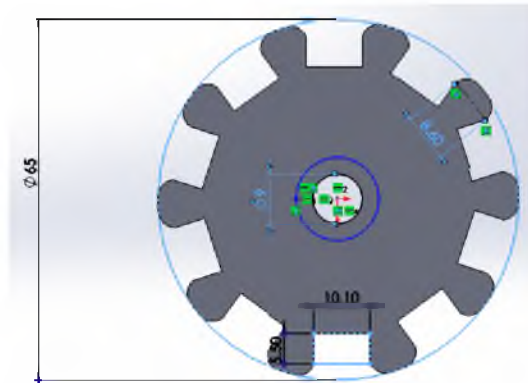


Figura 36. Engrane base.

Engrane paralelo: cuenta con 7 diente cada uno con una medida de 8.79 mm de ancho, 5mm de alto y 6mm de distancia entre cada diente, tiene un radio de 20mm figura 37.

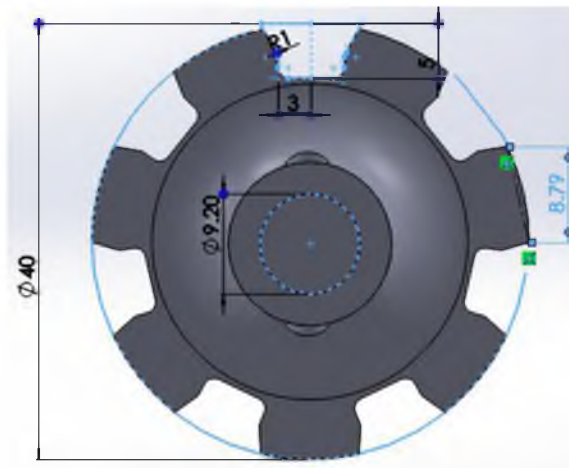


Figura 37. Engrane paralelo.

Este engrane va unido a un eje que esta sostenido por un balero (figura 36) también conocido como rodaje, rolinera, bolillero o ruleman sirve como apoyo a un eje y sobre el cual esta gira. Esta pieza está formada por un par de cilindros concéntricos, separados por una corona de rodillos o bolas que

giran de manera libre. Existen diferentes clases de rodamientos de acuerdo al tipo de esfuerzo que deben soportar en su funcionamiento.

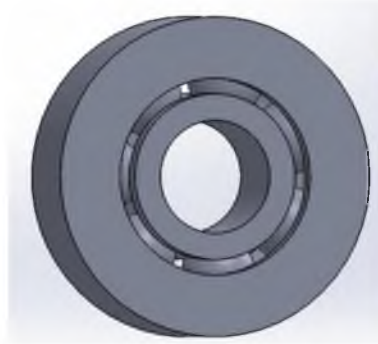


Figura 38. Balero.

El eje en un extremo tiene el engrane y en el otro extremo tiene una rueda excéntrica (figura 39) esta rueda excéntrica cuenta con tres agujeros de 3 y 8mm de distancia uno con otro esto es con el fin de que la biela-manivela tenga diferentes medidas de altura.

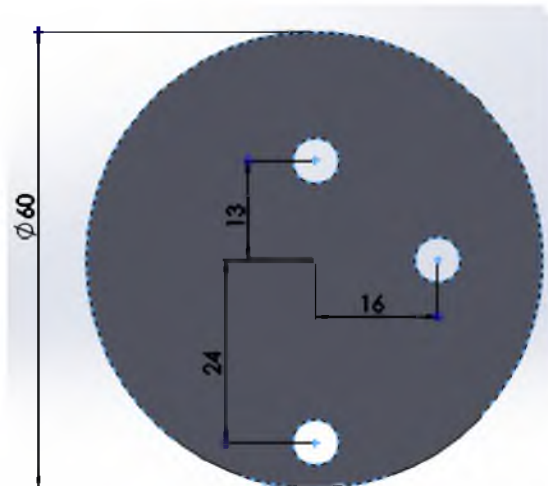


Figura 39. Manivela.

La biela-manivela (figura 32) está sujeta a un sistema de guiado (figura 40) para que siga un mismo movimiento, en el extremo de la biela tiene un arillo (figura 41) por donde va a pasar el producto.

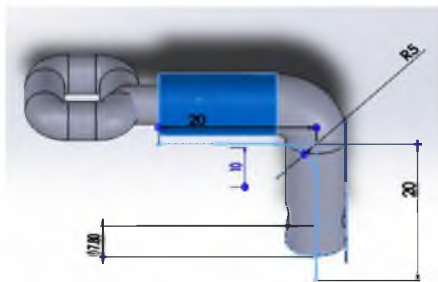


Figura 40. Sistema de guiado.



Figura 41. Arillo.

En la parte de arriba del engrane base se encuentra un tubo (figura 42) que es donde será enrollado el producto con una inclinación de 163° con el fin de que se produzca un efecto de subida y bajada. Sirve para que la base tubular tenga el ángulo de inclinación necesario para que pueda ser enrollado.

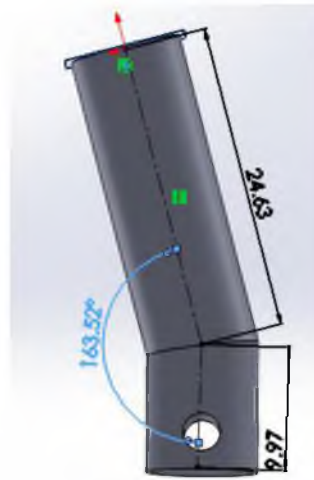


Figura 42. Soporte angular.

El objetivo de la siguiente pieza (figura 43) es proporcionar soporte para que, sobre ella de inicio al enrollado, la parte central sirve para que se enrolle el producto mientras que la parte de abajo sirve para dar firmeza a la bola.

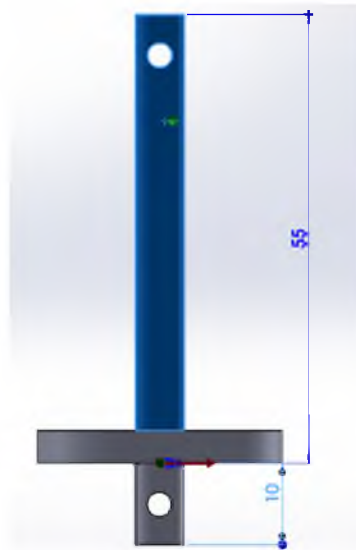


Figura 43. Base tubular.

4. PRUEBAS.

En la siguiente sección se describen las pruebas realizadas para verificar el funcionamiento del diseño y la programación.

4.1 DISEÑO PLACA FENÓLICA.

Placa fenólica con los elementos montados (figura 44).



Figura 44. Base del prototipo.

El Motorreductor está montado en una pieza impresa para sujetarlo a la base del prototipo (figura 45). Un cilindro sirve de cople (impreso para este fin) entre la flecha del motor y el eje del engrane base del sistema. Para evitar fricción y desgaste, este eje está montado sobre un balero.



Figura 45. Base del motorreductor.

Con la finalidad de evitar falsas mediciones se comprueba que la celda de carga tenga la distancia correcta con la base donde va a estar sujeta, dejando la parte central libre (figura 46).

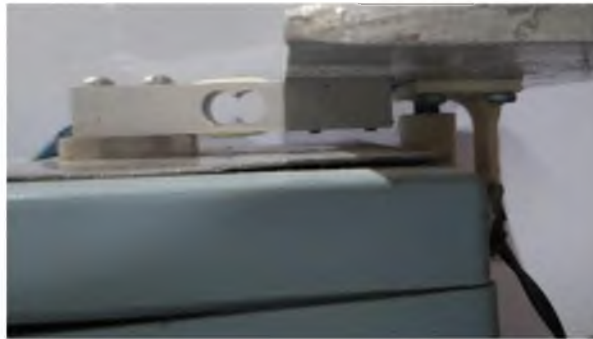


Figura 46. Celda de carga.

Se llevaron a cabo diferentes pruebas, entre las cuales está el programa del pesado y enrollado de queso. En dicho programa se utilizan librerías especializadas (bibliotecas externas del programa), como Arduino.h, LiquidCrystal.h, y HX711.h, este último asociado al módulo de carga.

```
#include <LiquidCrystal.h> //Libreria de LiquidCrystal  
  
#include "Arduino.h" //Libreria de arduino  
  
#include "HX711.h" //Libreria de la balanza  
  
#define DOUT A1 //Puerto de la balanza  
  
#define CLK A0 //Puerto de la balanza
```

En la librería LiquidCrystal se indican los pines de conexión con el Arduino como el LCD, utilizado consta de dos filas por 16 columnas lo cual se indica con la instrucción `lcd.begin(16,2)`, se muestran las conexiones a usar en el `lcd`, `*`, `**` y `***`.

El motorreductor en este caso se inicializa el pin:

* Pin 11 (Pin de habilitación / Control): Este pin debe mantenerse alto para ejecutar el proceso de lectura / escritura, y está conectado a la unidad del microcontrolador y se debe mantener alto constantemente.

** Pin 12 (Selección de registro / Pin de control): Este pin alterna entre comando o registro de datos, se utiliza para conectar un pin de unidad de microcontrolador y obtiene 0 o 1. (0 = modo de datos y 1 = modo de comando).

*** Pin 2 - 5 (Pines de datos): Estos pines se utilizan para enviar datos a la pantalla. Estos pines están conectados en modo de 4 bits.

```
LiquidCrystal lcd(12,11,5,4,3,2);  
  
void setup() {  
  
  pinMode(motorPin, OUTPUT);  
  
  lcd.begin(16, 2);  
  
  }
```

A continuación, se declaran las variables que serán utilizadas dentro del programa.

```
int lcd_key = 0; //Variable que toma datos de los botones  
  
int adc_key_in = 0; //Variable que toma datos de los botones  
  
#define btnizquierdo 1 //Boton izquierdo avanza  
  
#define btncentro 2 //Boton centro selecciona  
  
#define btnderecho 3 //Boton derecho atras  
  
int teclado_lcd = 0; //Almacena el resultado del teclado  
  
int seccion = 1; //Variable auxiliar en el menu  
  
int valor = 1; //Variable auiliar para el movimiento del menu con los
```


botones derecha izquierda.

```
int i;           //Variable auxiliar en el switch

int cant;       //Variable auxiliar en la cantidad del producto

float peso;     //Variable auxiliar en peso del producto

int motorPin=6; //Variable del motor y puerto

float auxbalanza; //Variable auxiliar en el peso de la balanza

int conteo;
```

Se definen tres botones que permiten navegar dentro del menú.

“Maquina Programada”

Modo a operar:

1.- Tarar bascula.

No ponga ningún objeto sobre la balanza.

Listo para pesar.

2.- Cantidad Producto.

1 producto.

2 productos.

3 productos.

3.- Peso producto.

100 gr. (0.10).

200 gr. (0.20).

300 gr. (0.30).

4.- Iniciar.

Dentro del menú existen 4 funciones: Tarar báscula, Cantidad producto, peso producto, iniciar.

- ❖ Primera función: Tarar báscula, es la función para descontar el peso que existe sobre la balanza, si no lo tomaría como valor nulo.

```
void tarar_bascula(){ //Función de tarar bascula  
  
while (seccion == 1){  
  
    balanza.set_scale(43943.025); // Establece la escala  
  
    balanza.tare(20); //El peso actual es considerado Tara.  
  
    lcd.setCursor(0,0);  
  
    lcd.print("No ponga ningun ");  
  
    lcd.setCursor(0,1);  
  
    lcd.print("Objeto balanza ");  
  
    delay(2000); //Retardo  
  
    lcd.setCursor(0,0);  
  
    lcd.print("Listo para pesar");  
  
    lcd.setCursor(0,1);
```

```
lcd.print("      ");

teclado_lcd = read_boton();    //Toma la captura del botón

cambio_menu();                //Realiza la acción

}

}
```

❖ Segunda función: Cantidad producto, aquí da la opción de elegir de cuanto producto se va a producir. En este caso se le dio la opción de 1, 2 o 3 productos. Si se eligió solo un producto la báscula solo pesara una vez y se detendrá el motor, si se eligió 2 productos después de pesar el primer producto se detendrá un momento el motor, para quitar el producto que esta sobre la báscula y continuará pesando el segundo producto y volverá a inicia el motor. De igual forma si se eligió pesar 3 productos el motor se detiene espera a que se quite el producto que esta sobre la báscula y una vez vacía se inicia nuevamente el proceso.

❖

```

void cantidad_producto(){           //Toma la cantidad de productos a
realizar

while (seccion == 1){

    lcd.setCursor(0,0);

    lcd.print("Cantidad");

    lcd.setCursor(10,0);

    lcd.print("<");

    lcd.setCursor(15,0);

    lcd.print(">");

    if (valor > 0 && valor <4 ){ //Condición para elegir 1, 2 o 3 cantidades

        lcd.setCursor(12,0);

        lcd.print(valor);           //Imprime el valor que se eligio
    }

    cant=valor;                     //Asigna el valor seleccionado a cant

    teclado_lcd = read_boton();    //Toma la captura del botón

    cambio_menu();                 //Realiza la acción
}

```

```
}
```

- ❖ Tercera función: Peso producto, aquí se muestra la opción donde se elige de que peso va hacer el producto.

```
void peso_producto(){           //Función donde tomas de que peso sera
el producto

while (seccion == 1 ){

    lcd.setCursor(0,0);

    lcd.print("Peso");

    lcd.setCursor(7,0);

    lcd.print("<");

    lcd.setCursor(15,0);

    lcd.print(">");

    if (peso = 100 && peso < 500) { // El peso puede ser 100, 200 o 300 gr.

        peso=valor * .100;

    }

}
```

```
lcd.setCursor(9 ,0);  
  
lcd.print(peso);  
  
teclado_lcd = read_boton();  
  
cambio_menu();  
  
}  
  
}
```

4.2 PROTOTIPO.



Figura 47. Prototipo final.

La figura 47, muestra el prototipo final. El diseño electromecánico, así como del sistema embebido permite realizar en forma semi-automática el enrollado de una sección transversal de queso de hebra para un peso específico.

Si bien el prototipo está desarrollado a escala, tomando una sección transversal de queso suficientemente pequeño el prototipo puede enrollarla de manera regular.

5. RESULTADOS Y CONCLUSIONES.

Después de haber realizado las pruebas pertinentes, en el desarrollo del prototipo, el resultado obtenido ha sido favorable. La máquina es controlada a través de la tarjeta Arduino, esta indica el momento en que el motor debe detenerse al llegar al peso indicado. El peso se obtendrá a través de la celda de carga. Las pruebas realizadas se efectuaron con diferentes volúmenes de queso, comprobando el funcionamiento correcto de la celda de carga.

El resultado de esta tesis promueve la vinculación entre el sector académico de la Universidad y el sector Productivo de la región. En este trabajo de tesis se construyó un sistema semi-automático que enrolle y pese el queso por medio de un sistema embebido.

APÉNDICE: PROGRAMA

```
/******Prototipo de máquina enrolladora de queso de hebra. *****/

/***Programa que realiza la función del pesado y enrollado del queso*/

/**dando la opción de la cantidad de producto a enrollar y el peso de casa
enrollado*/

#include <LiquidCrystal.h> //Libreria de LiquidCrystal

LiquidCrystal lcd(12,11,5,4,3,2);//Puertos a usar en el LCD

#include "Arduino.h" //Libreria de arduino

#include "HX711.h" //Libreria de la valanza

#define DOUT A1 //Puerto de la valanza

#define CLK A0 //Puerto de la valanza

int lcd_key = 0; //Variable que toma datos de los botones

int adc_key_in = 0; //Variable que toma datos de los botones

#define btnizquierdo 1 //Boton izquierdo avanza
```

```

#define btncentro 2 //Boton centro selecciona

#define btnderecho 3 //Boton derecho atras

int teclado_lcd = 0; //Almacena el resultado del teclado

int seccion = 1; //Variable auxiliar en el menu

int valor = 1; //Variable auiliar para el movimiento del menu con los
botones derecha izquierda.

int i; //Variable auxiliar en el switch

int cant; //Variable auxiliar en la cantidad del producto

float peso; //Variable auxiliar en peso del producto

int motorPin=6; //Variable del motor y puerto

float auxbalanza; //Variable auxiliar en el peso de la valanza

int conteo;

HX711 balanza(DOUT,CLK); //Balanza

void setup() {

    pinMode(motorPin, OUTPUT); //Inicializa el motor

    lcd.begin(16, 2); //Inicializa el LCD

    Serial.begin(9600);

```

```

}

void loop() {

    while (seccion == 1){          //seccion permite cambiar de menu, 1 es la
seccion del menu inicial de informacion,

                                   //la variable valor permite, con el uso de la
subrutina cambiomenu(), cambia la informacion

                                   //al presionar las teclas derechas izquierda.

        if (valor == 1) {          //El numero asignado a la variable valor
presente en menu definido

            lcd.setCursor (0,0);    //Mueve el cursor al primer caracter de la
primera linea

            lcd.print("  MAQUINA  "); //Imprime maquina

            lcd.setCursor(0,1);     //Mueve el cursor al primer caracter de la
segunda linea

            lcd.print(" PROGRAMADA  "); //Imprime programada

        }

        if (valor == 2) {          //El numero asignado a la variable valor
presente en menu definido

```

```
    lcd.setCursor (0,0);          //Mueve el cursor al primer caracter de la
primera linea
```

```
    lcd.print(" POR MEDIO DE "); //Imprime lo que esta en el parentesis
```

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

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

```
}
```

```
    if (valor == 3) {           //El numero asignado a la variable valor
presente en menu definido
```

```
        lcd.setCursor (0,0);          //Mueve el cursor al primer caracter de la
primera linea
```

```
        lcd.print(" Para mover el ");
```

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

```
        lcd.print("menu use < >");
```

```
}
```

```
if (valor == 4){
```

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

```
lcd.print("seleccione el ");  
  
lcd.setCursor(0,1);  
  
lcd.print(" modo a operar ");  
  
}
```

```
if (valor == 5){  
  
    lcd.setCursor(0,0);  
  
    lcd.print("cuando aparezca ");  
  
    lcd.setCursor(0,1);  
  
    lcd.print(" la opcion ");  
  
}
```

```
if (valor == 6){  
  
    lcd.setCursor(0,0);  
  
    lcd.print("deseada presione");  
  
    lcd.setCursor(0,1);  
  
    lcd.print("      =      ");  
  
}
```

```

    teclado_lcd = read_boton();    //Funcion que captura el boton que se
selecciono

    cambio_menu();                //Función
}

while(seccion == 2){            //seccion permite cambiar de menu, 2 es la
seccion de seleccionar la opción,

                                //la variable valor permite, con el uso de la subrutina
cambiomenu(), cambia la informacion

                                //al presionar las teclas derechas izquierda.

    if (valor == 1){            //cuando en el menu se encuentra la opcion
deseada se presiona selecccionar

        lcd.setCursor(0,0);

        lcd.print(" Modo a operar ");

        lcd.setCursor(0,1);

        lcd.print("Tarar bascula "); //Entra a la función de tarar bascula

    }

    if (valor == 2){

```

```

    lcd.setCursor(0,1);

    lcd.print("Cantidad producto"); //Entra a la función de cantida de
producto

}

if (valor == 3){

    lcd.setCursor(0,1);

    lcd.print("Peso producto "); //Entra a la función de la cantidad de peso

}

if (valor == 4){

    lcd.setCursor(0,1);

    lcd.print("Iniciar "); //Inicia a pesar

}

if (valor > 4){

    valor = 1;

}

teclado_lcd = read_boton();

cambio_menu();

}

```

```
while (seccion == 3){ //Dependiendo de la seleccion realizada se llama a
la subrutina correspondiente
```

```
    switch(i){ //En caso que se seleccione una opcion entra en ese
menu
```

```
        case 1:{ //Al apretar seleccionar entra a la sig. función
```

```
            seccion=1;
```

```
            tarar_bascula(); //codigo de tarar la bascula
```

```
            break;
```

```
        }
```

```
        case 2:{
```

```
            seccion= 1;
```

```
            cantidad_producto(); //Funcion para tomar la cantidad de productos
```

```
            break;
```

```
        }
```

```
        case 3:{
```

```
            seccion = 1;
```



```

        peso_producto());    //Manda a llamar la función para elegir el peso del
producto

        break;

    }

    case 4:{

        seccion=1;

        iniciar();    //Realiza la funcion para pesar y cantidad deseada

        break;

    }

}

}

}

```

```

void tarar_bascula(){    //Función de tarar bascula

    while (seccion == 1){

        balanza.set_scale(43943.025); // Establece la escala

        balanza.tare(20);    //El peso actual es considerado Tara.
    }
}

```

```

    lcd.setCursor(0,0);

    lcd.print("No ponga ningun ");

    lcd.setCursor(0,1);

    lcd.print("Objeto balanza ");

    delay(2000);          //Retardo

    lcd.setCursor(0,0);

    lcd.print("Listo para pesar");

    lcd.setCursor(0,1);

    lcd.print("      ");

    teclado_lcd = read_boton();    //Toma la captura del botón

    cambio_menu();                //Realiza la acción
}

}

void cantidad_producto(){        //Toma la cantidad de productos a
realizar

    while (seccion == 1){

        lcd.setCursor(0,0);

```

```

lcd.print("Cantidad");

lcd.setCursor(10,0);

lcd.print("<");

lcd.setCursor(15,0);

lcd.print(">");

if (valor > 0 && valor <4 ){ //Condición para elegir 1, 2 o 3 cantidades

    lcd.setCursor(12,0);

    lcd.print(valor);        //Imprime el valor que se eligio

}

cant=valor;                //Asigna el valor seleccionado a cant

teclado_lcd = read_boton(); //Toma la captura del botón

cambio_menu();            //Realiza la acción

}

}

void peso_producto(){      //Función donde tomas de que peso sera el
producto

while (seccion == 1 ){

```

```

    lcd.setCursor(0,0);

    lcd.print("Peso");

    lcd.setCursor(7,0);

    lcd.print("<");

    lcd.setCursor(15,0);

    lcd.print(">");

    if (peso = 100 && peso < 500 ){ // El peso puede ser 100, 200 o 300 gr.

        peso=valor * .100;

    }

    lcd.setCursor(9 ,0);

    lcd.print(peso);

    teclado_lcd = read_boton();

    cambio_menu();

}

}

void iniciar(){

    while (seccion == 1){

```

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

lcd.print("Cant=");

lcd.setCursor(5,0);

lcd.print(cant);

lcd.setCursor(0,1);

lcd.print("Peso=");

lcd.setCursor(5,1);

lcd.print(peso);

for(conteo=1; conteo<=cant; conteo++){

    lcd.setCursor(12,0);

    lcd.print(conteo);

    auxbalanza=(balanza.get_units(20));

    lcd.setCursor(11,1);           //ponte en la linea 1 posicion 9

    lcd.print(balanza.get_units(20),3);

    delay(500);

    while (auxbalanza <= peso){

        auxbalanza=(balanza.get_units(20));

        Serial.println(auxbalanza);
```

```

Serial.println(balanza.get_units(20));

digitalWrite(motorPin, HIGH);

Serial.print(balanza.get_units(20),3);

lcd.setCursor(11,1);           //ponte en la linea 1 posicion 9

lcd.print(balanza.get_units(20),3);

auxbalanza=(balanza.get_units(20));

}

digitalWrite(motorPin, LOW);

delay(5000);

}

while(seccion==1){

    lcd.setCursor(0,0);

    lcd.print("Proceso Terminado");

    delay (200);

    lcd.setCursor(0,1);

    lcd.print("Presione = para continuar");

    teclado_lcd = read_boton();

    cambio_menu();

```

```
    }  
  }  
}
```

```
int read_boton() { //leer boton
```

```
    adc_key_in = analogRead(2);
```

```
    if (adc_key_in < 630) return btnizquierdo;
```

```
    if (adc_key_in < 670) return btncentro;
```

```
    if (adc_key_in < 850) return btnderecho;
```

```
}
```

```
void cambio_menu(){ // Este programa se utiliza para mover los botones  
adelante o atras del menu que se despliega del LCD
```

```
    teclado_lcd = read_boton(); // Lee los botones
```

```
    switch (teclado_lcd)      // Eleccion de boton.
```

```
{
```

```
    case btnderecho:        //En caso de que se presione adelante.
```

```
{
```

```

    valor = valor + 1;    //Valor incrementa 1 y avanza

    delay(1000);        //El retardo es para evitar rebotes

    break;

}

case btnizquierdo:    //En caso que se presione atras se pregunta
primero si el valor es 1

{

    valor = valor -1;    //si no es 1 se decrementa el valor

    delay(1000);        // El retardo es para evitar rebotes

    break;

    if (valor = 1){

        break;        //Si es uno termina la seleccion.

    }

}

case btncentro:

{

    lcd.clear();

```



```
        seccion = seccion + 1; //cuando se oprime el boton seleccion se  
limpia el LCD  
  
        i = valor;          //aumentando la variable seccion, se utiliza en el sig.  
menu  
  
        valor = 1;         // se reinicia la variable indice  
  
        delay(1000);      // se espera un seg para evitar rebotes  
  
    }  
  
}  
  
}
```

REFERENCIA.

(1) Quesillo Oaxaca

http://www.bedri.es/Comer_y_beber/Queso/Quesos_del_mundo/Mejic_o/Quesillo_Oaxaca.htm Consulta en línea: 29/08/2016.

(2) ¿Qué es Arduino? <http://arduino.cl/que-es-arduino/> Consulta en línea: 01/09/2016.

(3) Celda de cargas. <http://www.pesaje-agropecuario.com.mx/celdas-de-carga.html> Consulta en línea: 05/09/2016.

(4) Libro: Sensores ya acondicionadores de señal

(5) Principio de los sensores. (2) Libro: Instrumentación industrial

(6) Libro: Diseño de maquinaria

(7) Sistemas Embebidos. <http://www.ikerlan.es/es/que-investigamos/sistemas-embebidos> Consulta en línea: 01/10/2016.

(8) Sistemas embebidos <http://server-die.alc.upv.es/asignaturas/PAEEES/2005-06/A07%20-%20Sistemas%20Embebidos.pdf>

(9) Sistemas embebidos

<http://repositorioacademico.upc.edu.pe/upc/bitstream/10757/579143/1/ Todo+sobre+sistemas+embebidos+-+1er+cap.pdf>

(10) Relevadores

<http://unicrom.com/rele-relay-relevador-interruptor-operado-magneticamente/>

- (11) Transistores
<https://iesvillalbahervastecnologia.files.wordpress.com/2008/04/transistores.pdf>
- (12) Engranés
http://electronicaestudio.com/docs/1550_Tutorial_de_ENGRANES.pdf
- (13) Mecanismos
<http://www.edu.xunta.gal/centros/iesfelixmuriel/system/files/M%C3%A1quinas%20y%20mecanismos.pdf>
- (14) Balanza electrónica usando un sensor de presión resistivo utilizando arduino.
http://www.cortoc.com/2012/07/balanza-electronica-utilizando-un_21.html Consulta en línea: 05/09/2016.
- (15) http://www.basculaspoise.com/Soporte/Celdas_de_Carga.html
http://www.naylampmechatronics.com/blog/25_Tutorial-trasmisor-de-celda-de-carga-HX711-Ba.html
- (16) https://hetpro-store.com/images/Tutoriales/pcb_eagle/hetpro_tutorial_pcb_eagle.pdf
- (17) Solidword.
http://www.marcombo.com/Descargas/9788426714589-SolidWorks/descarga_primer_capitulo_libro_solidworks.pdf Consulta en línea 06/02/2018.
- (18) Impresora 3D. <https://www.ecured.cu/SolidWork>

<http://www.atecnologia.com/informatica/impresoras-3d.html> Consulta en

línea 06/02/2018.