

INSTITUTO SUPERIOR TECNOLÓGICO

VIDA NUEVA



**Diseño y Construcción de un Sistema Automático para la Dosificación de
Granos Secos**

Presentado por:

Guaña Molina Leonardo Daniel

Tecnología Superior en Electromecánica

Tutor:

Ing. Machay Gomez Edwin Vinicio Mg.

Julio 2022

Quito – Ecuador

Tecnología Superior en Electromecánica**Certificación del Tutor**

En mi calidad de Tutor del Proyecto de Aplicación Práctica con el tema: “ Diseño y Construcción de un Sistema Automático para la Dosificación de Granos Secos”, presentado por el ciudadano Guaña Molina Leonardo Daniel, para optar por el título de Tecnólogo Superior en Electromecánica, certifico que dicho proyecto ha sido revisado en todas sus partes y considero que reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal examinador que se designe.

En la ciudad de Quito, del mes de julio de 2022.

Tutor: Ing. Machay Gómez Edwin Vinicio Mg.
C.I.: 0503646275

Tecnología Superior en Electromecánica**Aprobación del Tribunal**

Los miembros del tribunal aprueban el Proyecto de Aplicación Práctica, con el tema: “Diseño y Construcción de un Sistema Automático para la Dosificación de Granos Secos”, presentado por el ciudadano: Guaña Molina Leonardo Daniel, facultado en la carrera Tecnología Superior en Electromecánica.

Para constancia firman:

ING.

DOCENTE ISTVN

ING.

DOCENTE ISTVN

ING.

DOCENTE ISTVN

Tecnología Superior en Electromecánica**Cesión de Derechos de Autor**

Yo, Guaña Molina Leonardo Daniel portador de la cédula de ciudadanía 1725510539, facultado de la carrera Tecnología Superior en Electromecánica, autor de esta obra, certifico y proveo al Instituto Superior Tecnológico Vida Nueva, usar plenamente el contenido de este Proyecto de Aplicación Practica con el tema “ Diseño y Construcción de un Sistema Automático para la Dosificación de Granos Secos”, con el objeto de aportar y promover la lectura e investigación, autorizando la publicación de mi proyecto en la colección digital del repositorio institucional bajo la licencia de Creative Commons: Atribución-NoComercial-SinDerivadas.

En la ciudad de Quito, del mes de julio de 2022.

Guaña Molina Leonardo Daniel

C.I.:1725510539

Dedicatoria

Dedico este proyecto de aplicación práctica primero a Dios, ya que me permitió tener a tres personas importantes en mi vida, mi padre Luis Guaña, mi madre Nancy Molina y mi hermana Samantha Guaña, quienes son mi motivación para esforzarme día tras día; también supieron aconsejarme y darme ánimos para no rendirme a lo largo de mi vida estudiantil.

LEONARDO GUAÑA

Agradecimiento

El más sincero agradecimiento a Dios por haberme dado a mis padres que siempre me apoyaron para llegar a cumplir con este sueño que es de ser un hombre de bien y con una profesión, a mis amigos que me acompañaron en los momentos difíciles de mi vida estudiantil y también a mi docente tutor, Ingeniero Vinicio Machay que me supo aconsejar y corregirme en la ejecución del proyecto.

Índice de Contenido

Resumen	14
Abstract	15
Introducción	16
Planteamiento del Problema	17
Justificación	18
Objetivos.....	19
Objetivo General.....	19
Objetivos Específicos	19
Capítulo I.....	20
Marco Teórico.....	20
Antecedentes	20
Bases Teóricas	23
Sistema de Control	23
Sistema Automático.....	24
Dosificador de Alimentos	24
Automatización	25
Programación	25
Tipos de Arduinos	25
Arduino Mega	26
Fin de Carrera.....	27
Servomotor.....	28
Sensor Infrarrojo	29

Driver L298N (Puente H)	30
Módulo HX711	31
Motor de Corriente Continua	32
Granos Secos.....	33
Capitulo II.....	34
Metodología y Desarrollo del Proyecto	34
Diseño Metodológico	34
Señalamiento de Variables	34
Variable Independiente.....	34
Variable Dependiente	34
Enfoque de la Investigación.....	35
Alcance de la Investigación	35
Desarrollo	36
Desarrollo de Montaje Mecánico	36
Capitulo III	43
Propuesta	43
Análisis del Sistema	43
Diseño Mecánico.....	43
Diseño de la Tolva.....	45
Diseño del Túnel de Caída.....	45
Diseño de la Base para el Motor de 12 VCC	46
Diseño del Túnel Dosificador	47
Diseño de las Bases del Túnel Dosificador	47

Diseño de los Cortes para la Estructura.....	48
Diseño Eléctrico	49
Cálculos para el Diseño Mecánico	51
Cálculo para la Escala de la Báscula con Módulo HX711	53
Cálculos para la Tolva	53
Volumen 2 de la Tolva	53
Volumen Total de la Tolva	54
Capacidad de la Tolva	54
Cálculo para la Escala de la Báscula	54
Selección de Componentes	55
Acrílico	55
Placa Arduino Mega	55
Servomotor MG 996R	56
Báscula de 1KG con Módulo HX 711	57
Driver L298N	57
Sensor Infrarrojo	58
Motor de CC de 12V	59
Actuador Lineal a 12 VCC	59
Fuente de 12V	60
Programación del Arduino Mega	61
Calibración de la Báscula	61
Calibración del Sensor Infrarrojo	62
Programación del Servomotor	63

Programación Final	65
Conclusiones	67
Recomendaciones	68
Referencias Bibliográficas.....	69
Anexos.....	71

Índice de Figuras

Figura 1 Dosificador de granos secos	20
Figura 2 Dosificador de granos secos	21
Figura 3 Dosificador de granos secos	23
Figura 4 Dosificador de alimentos.....	24
Figura 5 Placa Arduino Uno.....	26
Figura 6 Placa Arduino Mega.....	27
Figura 7 Tipos de servomotores	28
Figura 8 Tipos de sensores infrarrojos	29
Figura 9 Sensor infrarrojo	30
Figura 10 Driver L298N.....	31
Figura 11 Módulo HX711	32
Figura 12 Motor de corriente continua.....	33
Figura 13 Tolva.....	36
Figura 14 Túnel de caída	37
Figura 15 Base de la tolva	37
Figura 16 Armado de la estructura.....	38
Figura 17 Base para el servomotor	39
Figura 18 Unión al actuador lineal.....	39
Figura 19 Unión de bases	40
Figura 20 Celda de carga con su base	41
Figura 21 Estructura del dosificador armado	41
Figura 22 Túnel de caída	42

Figura 23 Software Fusion y AutoCAD.....	44
Figura 24 Tolva.....	45
Figura 25 Diseño del túnel de caída.....	46
Figura 26 Base para el motor.....	47
Figura 27 Túnel dosificador	47
Figura 28 Bases para el túnel dosificador	48
Figura 29 Diseño de partes del dosificador	48
Figura 30 Software para construcción de circuitos de control	49
Figura 31 Simulación del funcionamiento	50
Figura 32 Software para construcción de circuitos neumáticos.	51
Figura 33 Acrílico	55
Figura 34 Placa Arduino Mega.....	56
Figura 35 Servomotor MG 996R.....	56
Figura 36 Software para construcción de circuitos neumáticos	57
Figura 37 Driver L298N o Puente H.....	58
Figura 38 Software para construcción de circuitos neumáticos	58
Figura 39 Motor de 12 VCC.....	59
Figura 40 Actuador linear a 12 VCC	60
Figura 41 Fuente de 12VCC.....	60
Figura 42 Primera programación en el software Arduino para la báscula	61
Figura 43 Segunda programación en el software Arduino para la báscula	62
Figura 44 Programación del sensor infrarrojo	63
Figura 45 Programación del servomotor.....	64

Figura 46 Programación del Display	65
Figura 47 Primera programación final	66

Resumen

El presente proyecto tiene como propósito implementar un sistema automático para la dosificación de granos secos y así permitir a los estudiantes del Instituto Superior Tecnológico Vida Nueva una máquina o sistema donde puedan practicar y aplicar todo lo aprendido en la carrera de Electromecánica y de esa forma mejorar sus conocimientos. Para el proyecto se usó una metodología aplicada, ya que esta investigación tiene como objetivo recolectar información sobre materiales y softwares que se usaron en la construcción del sistema. Los resultados obtenidos son los siguientes: los dispositivos electrónicos usados son servomotores, un actuador lineal, una placa Arduino Mega, una báscula con módulo HX711, un sensor infrarrojo y un driver para el cambio de giro de un motor de DC; para el diseño mecánico se usó Fusion 360 y AutoCAD.

El sistema está elaborado en función a los conocimientos adquiridos en la carrera de Electromecánica mediante las materias de Programación y Simulación, Control industrial. Y de esa forma se consigue un sistema automático para la dosificación de granos secos mediante el control de un Arduino Mega.

Palabras clave: Sistema de control, sistemas automáticos, dosificación, Arduino, automatización.

Abstract

The purpose of this project is to implement an automatic system for the dosing of dry grains and thus allow the students of the Instituto Superior Tecnológico Vida Nueva a machine or system where they can practice and apply all they have learned in the Electro mechanics career and improve their knowledge. An applicative methodology was used for the project, as this research aims to collect information on materials and software used in the construction of the system. The obtained results are: the electronic devices used are servomotors, a linear actuator, an Arduino Mega board, a scale with HX711 module, an infrared sensor, and a driver for the rotation change of a DC motor; 360° Fusion, and AutoCAD were used for the mechanical design.

The system is developed based on the knowledge acquired in the Electro mechanics career through the subjects of Programming and Simulation, Industrial Control. And thus, an automatic system for the dosing of dry grains is achieved through the control of an Arduino Mega.

Keywords: Control system, automatic systems, dosing, Arduino, automation.

Introducción

En el presente documento se redacta una breve explicación acerca del sistema automático para la dosificación de granos secos. El proyecto se ha diseñado con el propósito de reforzar el aprendizaje de los estudiantes sobre procesos automáticos industriales y de esa forma lograr un mejoramiento en sus conocimientos. En el máquina se podrá encontrar varios elementos que simularán los componentes usados en la industria, también se encontrará otros elementos los cuales reforzarán conocimientos en programación. La dosificación tiene como principal objetivo de racionar cualquier material. Dosificación es el acto y el efecto de determinar una dosis: una porción o cantidad de algo. (Petriz. R, 2021.p. 1).

El material que se usó para la estructura del proyecto fue acrílico y en la tolva se utilizó plástico. Como elementos de control se utilizó circuito electrónico para el encendido y apagado de motores paso a paso, actuador lineal y Arduino Mega. Lo primero que se hizo para la ejecución del proyecto fue diseñar la estructura del sistema de dosificación mediante el uso del software Fusión 360 y como último paso se realizó la construcción de piezas y estructuras donde se instalaron los elementos electrónicos y se realizó la programación en el software Arduino IDE. La prueba de funcionamiento de cada uno de los elementos se la hizo por separado, para luego unir cada componente con su respectiva programación para finalizar se realizó la prueba de funcionamiento se incluyeron los granos secos y la corrección de lecturas de peso, considerando el peso específico.

Planteamiento del Problema

El transcurso de los años se presentaron avances tecnológicos significativos. Hoy en día estos avances se aplican para casi todo, además la sociedad está basada en las tecnologías, tanto para la diversión, trabajo y comunicación. (Almudena, 2018, p. 4).

Considerando el párrafo anterior los avances tecnológicos se aplican para casi todo y la industria no es la excepción ya que en la industria se implementan estos avances para que su producción aumente su eficiencia. Por la falta de instrumentos tecnológicos en talleres los estudiantes no pueden desarrollar conocimiento y habilidades en el ámbito práctico, por esta razón es favorable la implementación de este tipo de proyecto que ayudan al fortalecimiento de dichas actitudes que permitirán a los estudiantes de la carrera de electromecánica ser más competitivos tal como menciona Yáñez C. (2018) es notable el uso que se les da a los instrumentos tecnológicos en un mundo globalizado, en diferentes espacios de la vida, ya sean socioculturales, políticos o educativo.

Las instituciones Educativas de nivel superior que no implementan esta tecnología impide que sus estudiantes tengan conocimientos sobre automatización, control, programación y simulación.

Debido a la innovación el Instituto Superior Tecnológico Vida Nueva en la carrera de Electromecánica se ha visto la necesidad de la creación de un sistema automático para dosificación de granos secos, lo cual permitirá el mejoramiento del proceso de enseñanza aprendizaje.

Justificación

En la actualidad los avances en la tecnología se aplican en todos lados y la industria no es una excepción, ya que buscan la mayor eficacia en la producción y de esa forma evitar pérdidas; por ello los estudiantes necesitan tener conocimiento sobre procesos industriales automáticos.

Al hablar de los procesos automáticos de la industria, la dosificación es una rama de estos, por lo cual mediante el proyecto permitirá al estudiante interactuar y crear una simulación de los diferentes procesos industriales, por ello a través de la construcción del dosificador automático de granos secos mediante una programación en Arduino Mega, se procura reforzar el conocimiento práctico, de esta manera permitir que el estudiante desarrolle de mejor manera los conocimientos teóricos y prácticos. Además, buscando ayudar a la institución para que aumente su prestigio y posea muchas más máquinas con las que los estudiantes puedan interactuar y al momento de graduarse, puedan salir mejor preparados en sistemas automáticos.

Uno de los beneficios que ofrece el sistema es su versatilidad, puesto que permite mezclar diferentes granos secos, también otro beneficio que ofrece, es la facilidad de cambiar la programación, de aumentar más sensores como pueden ser de temperatura o humedad por lo cual permitirá impulsar al estudiante en mejorar su creatividad y su capacidad de resolver problemas relacionados a los procesos industriales automáticos y semiautomáticos.

La máquina dosificadora tiene la capacidad de soltar una cantidad pequeña y una porción grande de granos secos, el peso llega a variar ya que solo es una maqueta que trata de simular cómo funciona un sistema automático.

Objetivos

Objetivo General

Construir un sistema de dosificación automático de granos secos mediante la programación de un Arduino, para que los estudiantes refuercen los conocimientos teóricos en la práctica.

Objetivos Específicos

- Investigar cómo funciona un dosificador, las partes que lo componen y el material de construcción adecuado.
- Analizar los resultados de la investigación para luego diseñar la estructura y los circuitos que tendrá el sistema
- Ensamblar cada parte del sistema de dosificación, incluyendo la programación de Arduino.
- Realizar las pruebas de funcionamiento del sistema automático de dosificación de granos secos.

Capítulo I

Marco Teórico

Antecedentes

Como punto de partida se tomó la tesis del Sr. Oviedo. N, (2018) con el nombre diseño y construcción de un dosificador de granos secos, el cual detalla los pasos para el diseño y construcción de dosificadoras.

Figura 1

Dosificador de granos secos



Nota. En la figura se muestra los tipos de dosificadores de granos secos. Obtenido del repositorio Digital de la Pontificia Universidad Católica del Ecuador Sede Ambato, por O. Navas, 2018, <https://repositorio.pucesa.edu.ec>.

Morales, E (2018) menciona en su investigación sobre el sistema automático de dosificación que:

Un sistema automático de dosificación es un conjunto de procesos y operaciones en los cual tiene la función de entregar la cantidad correcta de producto, con la participación mínima del humano y con la cantidad correcta de producto. La producción se obtiene en

porciones, cuando se completa la porción el equipo debe volver a llenar el producto y procesar la siguiente porción. (p.16).

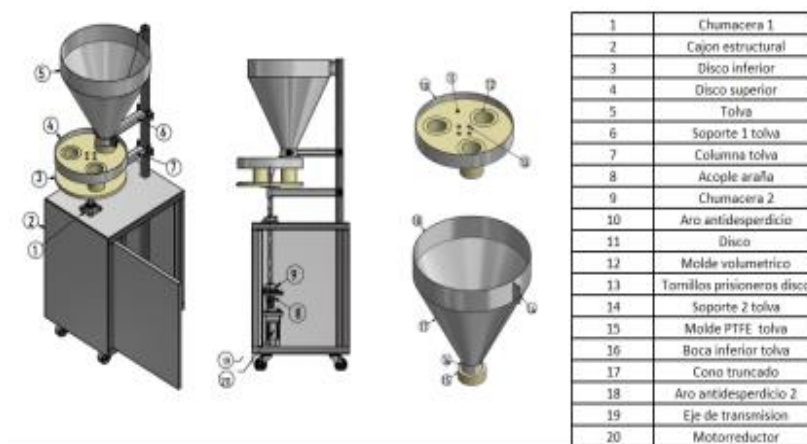
La automatización se ha vuelto sumamente importante en los procesos industriales, como lo afirma Sarzosa Anabel, Pasuña Ivan, Tahhan Francisco (2018) en su tesis:

La automatización juega un papel importante en la industria, puesto que de esta manera se puede lograr una producción en masa mayor a la obtenida en los procesos manuales. Este proyecto se constituye en un aporte tecnológico para los restaurantes, bares o cualquier tipo de industria que actualmente cuenta con un proceso de dosificación manual, que genera riesgos de derrames, fallos o cualquier error provocado por el trabajador (p.2).

En la tesis de Gómez David y Parra Julián sobre el Diseño e implementación de una dosificadora de condimentos para la línea de producción de la compañía Trébol Rojo S.A.S, la cual está dirigida netamente para los procesos industriales.

Figura 2

Dosificador de granos secos



Nota. En la figura se observa dosificador de condimentos. Obtenido del Repositorio Digital de la Universidad Antonio Mariño, por D. Gómez y J. Parra, 2021, <http://repositorio.uan.edu.co/handle/123456789/1730>.

Las máquinas dosificadoras de granos ya han sido implementadas para productores pequeños o para industrias grandes, como lo menciona Cazares Erik, Garrido Orlando (2019) en su investigación para su tesis:

La máquina dosificadora de grano pequeño para pequeñas industrias es el inicio de dicho proceso, se creó un diseño propio tomando como base lo que ya se ha desarrollado en otros países, para ajustarlo a las necesidades y al desarrollo de nuestro país dependiendo de la necesidad industrial (p.5).

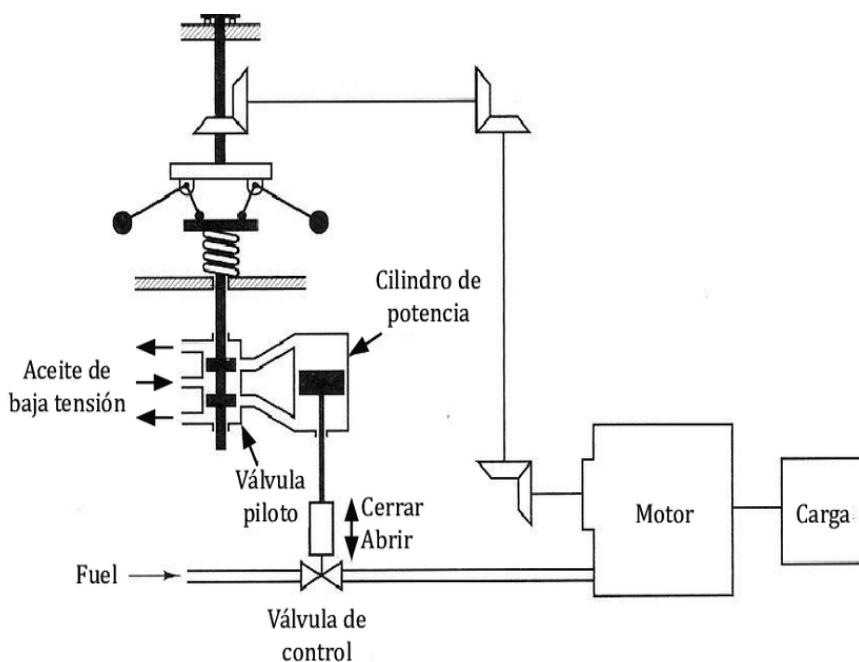
Bases Teóricas

Sistema de Control

El sistema de control es un conjunto de componentes eléctricos, electrónicos, mecánicos, hidráulicos los cuales comandan un proceso. “Un sistema de control es aquel en el que las variables de salida se comportan según las órdenes dadas por las variables de entrada.” (Branete, Segundo, Herrero, 2018, p. 15).

Figura 3

Dosificador de granos secos



Nota. En la figura se muestran los componentes que posee un sistema de control. Extraído de ResearchGate, por C. Timaure, 2016, https://www.researchgate.net/figure/Componentes-basicos-de-un-sistema-de-control-13-Ejemplos-de-Sistemas-de-Control-131_fig18_314074155.

Sistema Automático

El sistema automático es un conjunto de componentes conectados y relacionados con el fin de sustituir al humano de los diferentes procesos industriales. Castilla. J, (2019) en su investigación menciona que:

Un sistema automático de control es un conjunto de componentes físicos conectados o relacionados entre sí, de manera que regulen o dirijan su actuación por sí mismos, es decir sin intervención de agentes exteriores (incluido el factor humano), corrigiendo además los posibles errores que se presenten en su funcionamiento (p.3).

Dosificador de Alimentos

Un dosificador es una máquina o sistema el cual se encarga de una cantidad específica de cualquier material, granos. “Un dosificador tiene la siguiente definición un aparato o medidor con el cual se administran dosis específicas de algún producto, con el fin de evitar el desperdicio” (Carillo. M, 2015, p. 6).

Figura 4

Dosificador de alimentos



Nota. En la figura se muestra un tipo de dosificador. Extraído de CollingWood Packaging Machinery, 2016, https://www.coretamp.com/?gclid=EAIaIQobChMIi8TwuZKe-QIVmIrICh2m3AyfEAAYASAAEgIdQfD_BwE.

La principal función de la máquina es entregar una cantidad específica de cualquier tipo de grano seco. “La función del dosificador es entregar o suministrar de forma ágil la cantidad de material o insumo necesario para la realización de un sistema” (Carrillo. M, 2015, p. 6).

Automatización

La automatización es la implementación de diferentes elementos en algunos procesos industriales con el fin de hacerlos automáticos y evitar la intervención del humano

Según Agudelo, Tano, Vargas (2018) menciona que:

La automatización es un concepto que suele utilizarse en el ámbito de la industria con referencia al sistema que permite que una máquina desarrolle ciertos procesos o realice tareas sin intervención del ser humano. Es empleada con frecuencia por su capacidad de ahorrar tiempo y dinero (p.1).

Programación

La programación es la acción de codificar diferentes series de instrucciones en un tipo de lenguaje diferente. Cáceres. L (2019) menciona que:

Es un proceso mediante el cual se codifica una serie de instrucciones, en un determinado lenguaje, para ser posteriormente decodificados y ejecutados por un sistema computacional, todo ello con el fin de resolver un problema. Es decir, implementar desde un algoritmo hacia un lenguaje de programación y dar solución a un problema (p. 9).

Tipos de Arduinos

Arduino es una plataforma en la cual se introduce la programación para luego pasarla a la placa Arduino. Según algunos autores, “Arduino tiene la siguiente definición, es una plataforma de código abierto de prototipos electrónicos que se basa en hardware y software flexibles y

fáciles de usar que ponen al alcance de cualquier persona la construcción de circuitos electrónicos/robots” (Muñoz. A, Córcoles. S, p. 19).

Figura 5

Placa Arduino Uno



Nota. La figura muestra la placa Arduino uno. Extraído de Arduino.CC, por Placa Arduino UNO, 2015, <https://www.arduino.cc>.

Arduino Mega

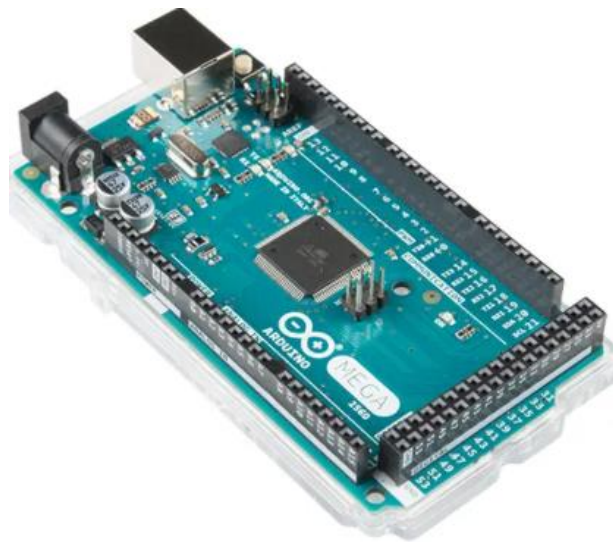
Arduino Mega es parecido al Arduino UNO, sin embargo, hay una diferencia notable entre las dos placas las cuales son el número de entradas y salidas que poseen. Como lo afirma Gonzales, J (2018) en su libro:

Arduino Mega es una placa microcontrolador basada en el microprocesador Atmega2560. Tiene 54 entradas/salidas digitales (de las cuales 15 pueden utilizarse para salidas PWM), 16 entradas analógicas, 4 UARTs (puertos serie por hardware), un oscilador de 16MHz, una conexión USB, entrada de corriente, conector ICSP y botón de reset (p.24).

Para el dosificador de granos secos la cantidad de entradas y salidas que tiene Arduino Mega son suficientes, ya que posee varios sensores, motores, displays y pulsadores. Además si se quisiese mejorar al dosificador aún habría pines libres.

Figura 6

Placa Arduino Mega



Nota. En la figura se muestra la placa Arduino usada para el proyecto. Extraído de Arduinio.cl, por Placa Arduino Mega, s.f., <https://arduino.cl/producto/arduino-mega-2560/>.

Fin de Carrera

En la automatización industrial existen multitudes de elementos eléctricos que facilitan los procesos de producción. El fin de carrera es un sensor que permite saber la posición de un elemento móvil mediante el accionamiento mecánico. “Las salidas del sensor arrojan respuestas binarias, un 1 o un 0. El final de carrera o sensor de contacto (también conocido como "interruptor de límite") o limit switch, son dispositivos eléctricos, neumáticos o mecánicos situados al final del recorrido de un elemento móvil” (Cruz, 2018, p. 1).

Servomotor

El servomotor es un motor especial de DC, ya que permite controlar la posición del eje mediante el uso de determinada cantidad de ángulos. “La definición de servomotor es un tipo especial de motor que permite controlar la posición del eje en un momento dado. Está diseñado para moverse determinada cantidad de grados y luego mantenerse fijo en una posición”

(González, 2018, p. 7).

Figura 7

Tipos de servomotores



Nota. En la figura se muestra los dos tipos de servomotores, los primeros usados en las industrias y la segunda para modelismos. Extraído de Electrónica JR, tipos de Servomotores, 2018, <http://electronicajunior.com.mx/articulo5.html>.

Sensor Infrarrojo

Primero definamos que es un sensor, es un dispositivo capacitado para captar acciones o estímulos externos y responder en consecuencia, “el sensor es el nombre más utilizado en control de autómatas para referirse al dispositivo que mide una magnitud física” (Branete. Segundo. Herrero, 2018, p. 1).

Figura 8

Tipos de sensores infrarrojos



Nota. En la figura se muestra los diferentes sensores infrarrojos industriales. Extraído de SEPIA, tipos de sensores infrarrojos, 2018, <https://suelpla.com/marcas/sick/>.

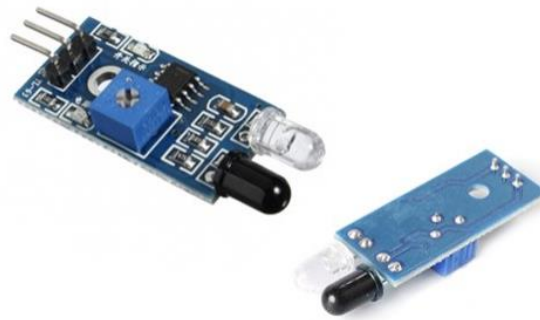
Para poder verificar si hay o no hay granos secos en la tolva se usó un sensor infrarrojo, el cual es un dispositivo opto electrónico, que posee componentes electrónicos como un led infrarrojo y un fototransistor capaces de medir la radiación electromagnética infrarroja de los cuerpos en su campo de visión. Bandala, R (2018) en su investigación define a los sensores infrarrojos como:

Dispositivos opto-eléctricos capaces de medir la radiación electromagnética de los cuerpos en su campo de división, es decir que cada uno de los cuerpos humanos emiten una cierta cantidad de radiación que resulta invisible ante la vista de estos cuerpos (p.5).

Una de las principales ventajas del sensor, es su compatibilidad con la placa Arduino, ya que no necesita ningún dispositivo o adaptador para su funcionamiento.

Figura 9

Sensor infrarrojo



Nota. En la figura se muestra el sensor infrarrojo usado en el proyecto. Extraído de Arduino.CC, sensor infrarrojo, s.f., <https://www.arduino.cc>.

Driver L298N (Puente H)

El puente H o Driver L298N se usa generalmente para el control de inversión de giro o la velocidad de los motores de corriente continua.” Este módulo basado en el chip L298N te permite controlar dos motores de corriente continua o un motor paso a paso bipolar de hasta 2 amperios” (ElectroniLab, 2018, p. 1).

Figura 10*Driver L298N*

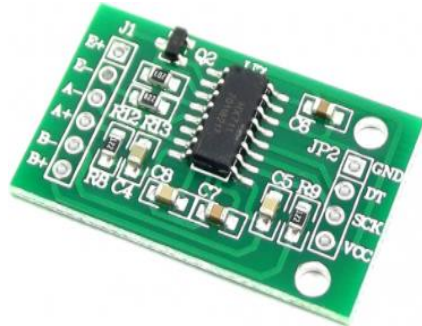
Nota. En la figura se muestra el Driver L298N o puente H usado para el dosificador. Extraído de NAYLAMP, Driver L298N, s.f., <https://naylampmechatronics.com/drivers/11-driver-puente-h-l298n.html>.

Módulo HX711

El módulo HX711 es un convertidor analógico a la digital que permite pesar algún material sólido y líquido. “El módulo HX711 es un transmisor entre las celdas de carga y un microcontrolador como Arduino, permitiendo leer el peso en la celda de manera sencilla. Es compatible con las celdas de carga de 5 kg, 20 kg y 50 kg” (Código Electrónica, 2018, p. 1).

Figura 11

Módulo HX711



Nota. En la figura se muestra el módulo HX711 usado para la celda de carga de 2 Kg. Extraído de NAYLAMP, Modulo HX711, s.f., <https://naylampmechatronics.com/fuerza-peso/147-modulo-hx711-transmisor-de-celda-decarga.html>.

Motor de Corriente Continua

El motor de corriente continua es una máquina giratoria capaz de transformar la energía eléctrica en energía mecánica. La funcionalidad está basada en el principio de inducción, la cual crea una fuerza electromagnética a partir de una corriente de entrada, que a su vez crea un movimiento giratorio. “El motor eléctrico permite la transformación de energía eléctrica en energía mecánica, esto se logra mediante la rotación de un campo magnético alrededor de una espira o bobinado que toma diferentes formas” (Jorge. L, Patiño. V, 2018, p. 3).

Figura 12*Motor de corriente continua*

Nota. La figura muestra un motor de DC. Extraído de Made in China, Motor de corriente continua, s.f., https://es.made-in-china.com/co_leisonmotor/product_230V-AC-Motor-HVDC-Motor-for-Mixer-and-Hand-Blender_herrroguy.html.

Granos Secos

Los granos secos son semillas las cuales han perdido la capacidad de germinar plantas para su producción. Hay varios tipos de granos secos como el maíz, garbanzo, arvejas, habas, morocho, lentejas, mote, chochos. El frijol usado para dosificar es el Canario un tipo de frijol ecuatoriano el cual es bastante resistente a la Antracnosis, el tamaño del grano es mediano y llega a pesar alrededor de 0.3 y 0.45 gramos cada uno.

Capítulo II

Metodología y Desarrollo del Proyecto

Diseño Metodológico

Para esta investigación se aplica un enfoque cualitativo y cuantitativo por las diferentes investigaciones y las pruebas de funcionamiento que se harán al sistema de dosificación automático. En cuanto al alcance del proyecto son los estudiantes del Instituto Superior Tecnológico Vida Nueva, ya que para ellos está dirigido el proyecto. Lozada, J (2018) menciona que:

La investigación aplicada busca la generación de conocimiento con aplicación directa a los problemas de la sociedad o el sector productivo. Esta se basa fundamentalmente en los hallazgos tecnológicos de la investigación básica, ocupándose del proceso de enlace entre la teoría y el producto (p.5).

Se utilizará una investigación aplicada con el objetivo de recolectar información sobre los materiales y softwares que se usaron para la construcción del sistema, a continuación, se realizó el presupuesto para luego verificar los alcances técnicos, aportes académicos para la investigación y construcción del sistema.

Señalamiento de Variables

Variable Independiente

Diseño y construcción de dosificador automático de granos secos para el taller de programación y simulación, control industrial.

Variable Dependiente

El aprendizaje dentro de las áreas de control industrial y programación. La investigación toma como variable independiente el diseño y construcción de dosificador automático de granos

secos, lo cual abarca la automatización, la programación, el control industrial. Mientras que la variable dependiente se centra en la mejora de los procesos de aprendizaje dentro de las áreas de control industrial.

Enfoque de la Investigación

La investigación se basa en un enfoque cualitativo y cuantitativo. El enfoque cualitativo se centra directamente en las técnicas, parámetros y especificaciones que se usaron al momento de realizar la programación en Arduino Mega porque posee versatilidad al implementar elementos y programarlos. Por otro lado, el enfoque cuantitativo se centrará específicamente en las pruebas de funcionamiento de la programación, de la medición del peso, de los números de vueltas que tenga que dar el motor paso a paso.

Alcance de la Investigación

El alcance del proyecto se basa en el análisis de la investigación en donde se pueda identificar el tipo de lenguaje de programación a usar, los elementos eléctricos y mecánicos más convenientes para el sistema, el material adecuado y el correcto diseño de planos eléctricos y mecánicos; de esta forma evitaremos retrasos en la ejecución del proyecto. Por otro lado, el alcance del proyecto son los estudiantes ya que el proyecto está diseñado para dosificar un peso menor a lo que se usa en las industrias.

Desarrollo

Desarrollo de Montaje Mecánico

Para el desarrollo del dosificador de granos secos, primero se comenzó por armar todo el sistema que tiene la tolva. La tolva está conformada por tres elementos, el primero es una hélice, la cual tiene el propósito de permitir que caigan los granos secos y a su vez evitar que se riegan los granos, el material de la hélice es de caucho, el segundo elemento es el acople de plástico que va dentro de la hélice y se sujeta al último elemento, el cual es la tolva que está construida de plástico que ayuda a preservar los granos secos.

Figura 13

Tolva



Nota. En la figura de muestra las partes de la tolva.

Luego de que la tolva se armó, se empezó con el armado del túnel de caída, que contiene 3 pedazos de acrílico en su interior, estos tienen el propósito de hacer que los granos caigan de una forma ordenada y en dirección a la báscula, ya que, si no cae de forma vertical, los granos llegan a rebotar y afectaría al momento del pesado. El material que se usó para unir cada pieza es Silicona Blanca ABRD 1200.

Figura 14*Túnel de caída*

Nota. En la figura se muestra el ensamblaje del túnel de caída.

Luego de que el túnel de caída está totalmente armado, se armó la base de la tolva, la base es una plancha de acrílico y plástico. La base de la tolva va puesta en la parte superior del túnel dosificador.

Figura 15*Base de la tolva*

Nota. En la figura se muestra el ensamblaje de la base para la tolva.

Como siguiente paso para el armado de la estructura del dosificador, se unieron las partes coartadas para el túnel. En la parte frontal va pagada una rampa para que los granos secos puedan caer directo al recipiente y evitar que los granos secos se rieguen fuera del recipiente.

Figura 16

Armado de la estructura



Nota. La figura muestra el ensamblaje de las partes del túnel dosificador.

Seguidamente se procedió a armar la base para el servomotor que se encarga de abrir y cerrar la compuerta que sostiene a los granos secos y evita que se caigan. La base está hecha metal la cual está acoplada con pernos y tuercas al túnel dosificador. Para unir la compuerta al servomotor se usó el adaptador circular que vino incluido en la funda con el servomotor y este se sujetó con tres tornillos.

Figura 17

Base para el servomotor



Nota. En la figura se muestra la compuerta acoplada al servomotor.

Después se adaptó un cuadrado de acrílico a la punta del pistón del actuador lineal. La razón de esta unión es que se pueda empujar los granos secos por el túnel dosificador. El actuador se acopla dentro del túnel.

Figura 18

Unión al actuador lineal



Nota. En la figura se muestra la unión de la placa de acrílico con el pistón.

Después se armó las bases del túnel que servirán para anclarlas a la tabla de madera que será donde se asienta toda la estructura del sistema de dosificación. El material de las bases es Grillon de 3 cm de diámetro y de altura tiene 10 cm de altura. Se usaron planchas de acrílico para anclar el túnel y las bases para el anclaje se utiliza pernos de 1" de longitud

Figura 19

Unión de bases



Nota. En la figura se muestra las partes de la base para el túnel dosificador.

Ya como penúltimo paso se armó las partes que conforman la celda de carga. Para esto se utilizó 6 cuadrados de acrílico los cuales tienen las siguientes medidas: 4 cm de ancho y 8 cm de largo. Se usó 3 planchas de acrílico en la parte superior de la báscula y 3 planchas en la parte inferior de las mismas medidas que se mencionaron anteriormente. Además, se utilizó dos pedazos de 10 cm de altura de Grillon para anclar la celda de carga a la base de madera para que tenga un apoyo y no se llegue a caer con los granos secos.

Figura 20

Celda de carga con su base



Nota. En la figura se muestra la báscula armada.

Ya como último paso en el armado del sistema, se unieron y pegaron todas las partes armadas para luego acoplar con la tabla de madera donde se asentó la estructura del dosificador.

Figura 21

Estructura del dosificador armado



Nota. En la figura se muestra la estructura del dosificador totalmente armado.

Diseño del Túnel de Caída. El túnel de caída está diseñado con las siguientes medidas, de ancho tiene 10 cm, de altura tiene 11 cm y de largo tiene 10.5 cm, además tiene 3 cuadrados en su interior de 10 cm de ancho y de largo tiene 5 cm. Tiene como principal función permitir la caída de los granos secos de una forma ordenada a la báscula evitando el rebote de los granos al actuador lineal y de esa forma evitar fallas o atascamientos en el funcionamiento óptimo.

Figura 22

Túnel de caída



Nota. En la figura se muestra el diseño interno y externo del túnel de caída.

Capítulo III

Propuesta

Análisis del Sistema

El sistema de dosificación automático de granos secos controlado por un Arduino Mega se encuentra basado en diferentes normativas de construcción y programación tales como: El código de dibujo técnico mecánico, la normativa UNE-IEC 60364-7 la cual está enfocada a las instalaciones eléctricas de baja tensión y la norma UNE 60617 que se refiere a la simbología, código de letras para los circuitos de control industrial.

Este dosificador funciona con voltaje continuo que es suministrado por una fuente de 12 V, la cual es alimentada por 110 voltios alternos. La fuente de 12 V alimenta al actuador lineal y al motor de la tolva. La placa Arduino es alimentada con una fuente de 5V, que a su vez también alimenta al display y al servomotor. Los únicos elementos que usan 110 V son del circuito de control que consta de un pulsador de color verde, un pulsador tipo seta, un relé de 110 v, una luz piloto roja y una verde, este circuito solamente se usa para energizar a todo el dosificador o para apagarlo ya sea por un accidente o porque ya no se utilizara.

Diseño Mecánico

Para el diseño de cada parte del dosificador se basó en el Código de Dibujo Técnico Mecánico, en el cual se diseñó todas las piezas de ensamblaje como las bases, las partes del túnel de dosificación, el túnel de caída, la rampa de caída para los granos secos, la base, las bases para cada actuador. Para la realización de los planos mecánicos se usó Autodesk AutoCAD 2022 (Licencia para estudiantes con duración para un año) y Autodesk Fusion 360 (Licencia para estudiantes con duración para un año). con su función más avanzada de diseño en 3D, la cual se

usó para hacer un prototipo del sistema de dosificación automático completo y también para cada pieza del mismo.

Figura 23

Software Fusion y AutoCAD



Nota. En la figura se muestra los dos programas usados para el diseño mecánico del dosificador.

AutoCAD es un software de diseño asistido por computadora, que se usa para dibujar, diseñar y modelar en 2D y 3D de forma precisa con sólidos, superficies, objetos de malla, características de documentación, además AutoCAD es un programa que permite el desarrollo de

proyectos de índole arquitectónico, industrial, mecánicos, eléctricos, de diseño gráficos y de ingeniería.

Fusion 360 es un software CAD, CAM y de circuitos impresos de modelado 3D basados en la nube para el diseño y la manufactura de productos, este software diseña y proyecta productos para garantizar su ajuste, estética, forma y función.

Diseño de la Tolva

La primera pieza que se realizó para el diseño del dosificador fue la tolva. La tolva está construida con plástico y tiene la función de almacenar los granos secos, la tolva tiene la capacidad máxima de almacenamiento de 2 kg. Si el peso aumento le motor no podrá dosificar

Figura 24

Tolva



Nota. La figura muestra el diseño de la tolva con 30 cm de alto.

Diseño del Túnel de Caída.

El túnel de caída está diseñado con las siguientes medidas, de ancho tiene 10 cm, de altura tiene 11 cm y de largo tiene 10.5 cm, además tiene 3 cuadrados en su interior de 10 cm de ancho y de largo tiene 5 cm. Tiene como principal función permitir la caída de los granos secos

de una forma ordenada a la báscula, evitando el rebote de los granos al actuador lineal y de esa forma evitar fallas o atascamientos en el funcionamiento óptimo

Figura 25

Diseño del túnel de caída



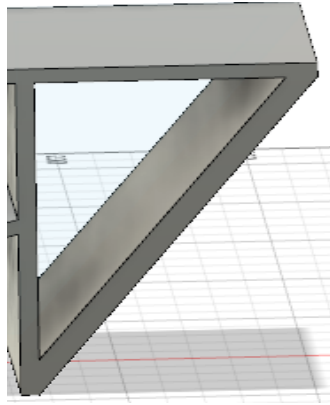
Nota. En la figura se muestra el diseño interno del túnel de caída.

Diseño de la Base para el Motor de 12 VCC

La base para el motor está junto al túnel de caída, esta parte del sistema cumple la función de sujetar con firmeza el motor. Las medidas de la base son: de ancho 7.5 cm, de largo 10.5 cm y de altura tiene 10.5 cm.

Figura 26

Base para el motor



Nota. La figura muestra una vista lateral de la base para el motor.

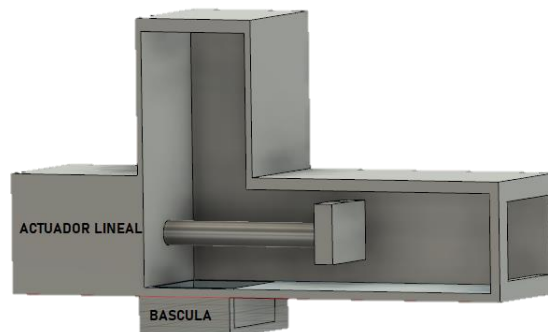
Diseño del Túnel Dosificador

Luego de diseñar la tolva, se diseñó el túnel donde caerán y se pesarán a los granos secos.

El túnel dosificador cumple la función de evitar que se rieguen los granos secos.

Figura 27

Túnel dosificador



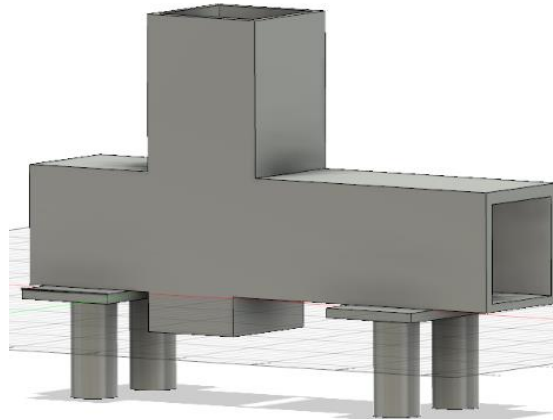
Nota. La figura muestra el diseño del túnel de dosificación.

Diseño de las Bases del Túnel Dosificador

Después del diseño del túnel se procedió con el modelado de las bases considerando las bases que sostendrá la tolva.

Figura 28

Bases para el túnel dosificador



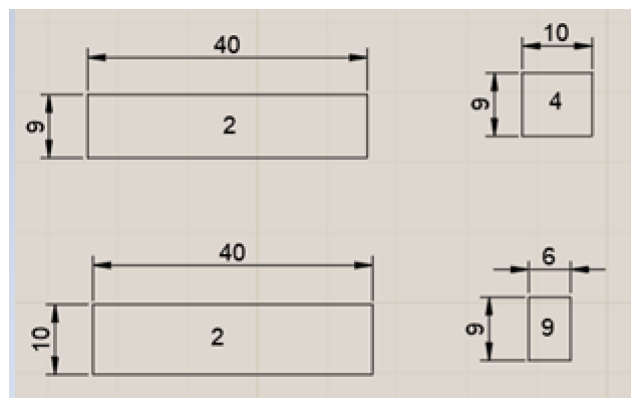
Nota. La figura muestra el diseño de las bases del túnel de dosificación.

Diseño de los Cortes para la Estructura

Después de que cada parte del sistema está diseñada, se procedió a diseñar en el software AutoCAD los cortes que para la plancha de acrílico

Figura 29

Diseño de partes del dosificador



Nota. La figura muestra las partes a cortar. AutoCAD.

Diseño Eléctrico

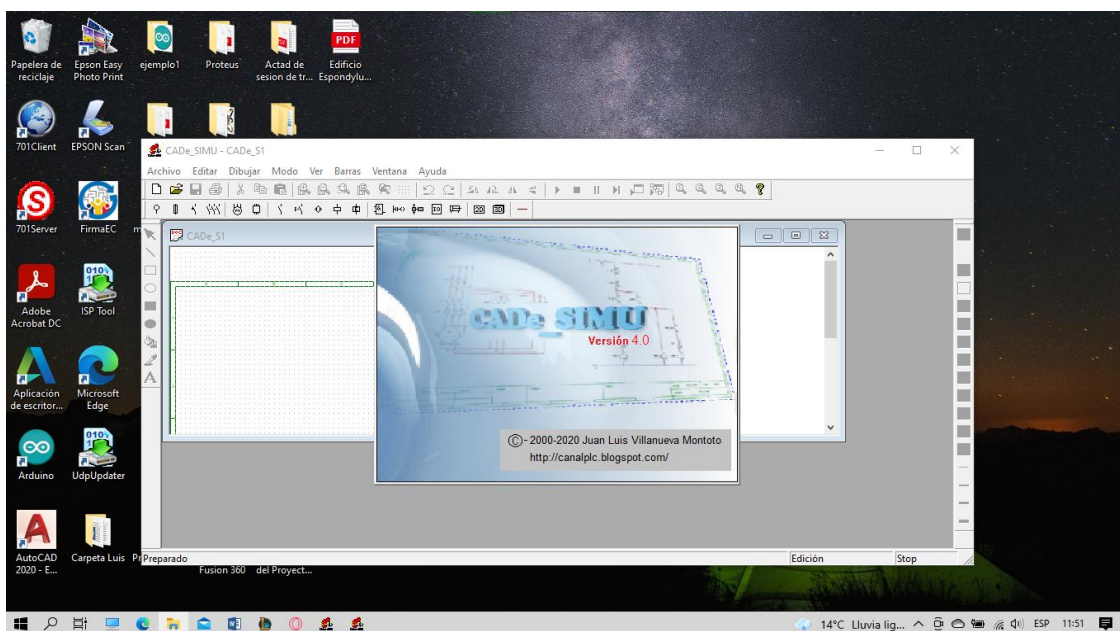
A continuación, se diseña el circuito de control por el que todo el sistema se energiza. Tomando en cuenta la norma UNE 60617, norma que señala la simbología de cada elemento que conforma un circuito.

Un circuito de control industrial es básicamente el conjunto de componentes eléctricos y electrónicos. “Un circuito de control es el conjunto de componentes eléctricos o electrónicos, conectados mediante conductores adecuados, que desarrollan cierta función de control” (Sigüenza. G, 2017, p. 15).

Para el diseño del circuito de control se utiliza el software CADe SIMU con licencia, permitiendo realizar tanto conexiones, simulaciones y comprobaciones del circuito. Seguidamente se muestra el diseño eléctrico.

Figura 30

Software para construcción de circuitos de control



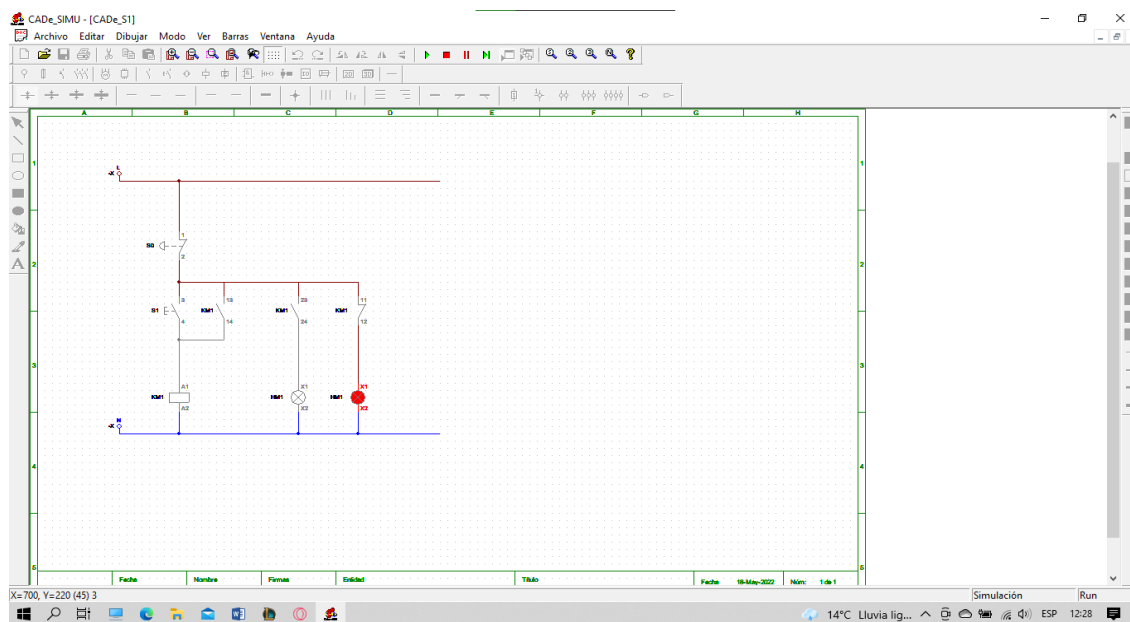
Nota. Esta figura representa el software utilizado para la elaboración de circuitos de control.

El funcionamiento del circuito, complementado por esquemas del mismo permite observar el funcionamiento de todos los elementos empleamos para el mismo, siendo el color rojo para líneas energizadas, gris para líneas desconectadas y azul para las líneas neutras.

- El circuito es energizado por 110 v, ya sea de un enchufe doméstico o tomado desde los disyuntores.
- Al momento de accionar el voltaje recorre por el pulsador normalmente abierto, esperando a que sea pulsado para energizar la bobina del relé de 110 V. Mientras el pulsador verde siga en su estado natural, una luz piloto de color rojo estará encendida, informando al operado que aún no se acciona el relé.

Figura 31

Simulación del funcionamiento

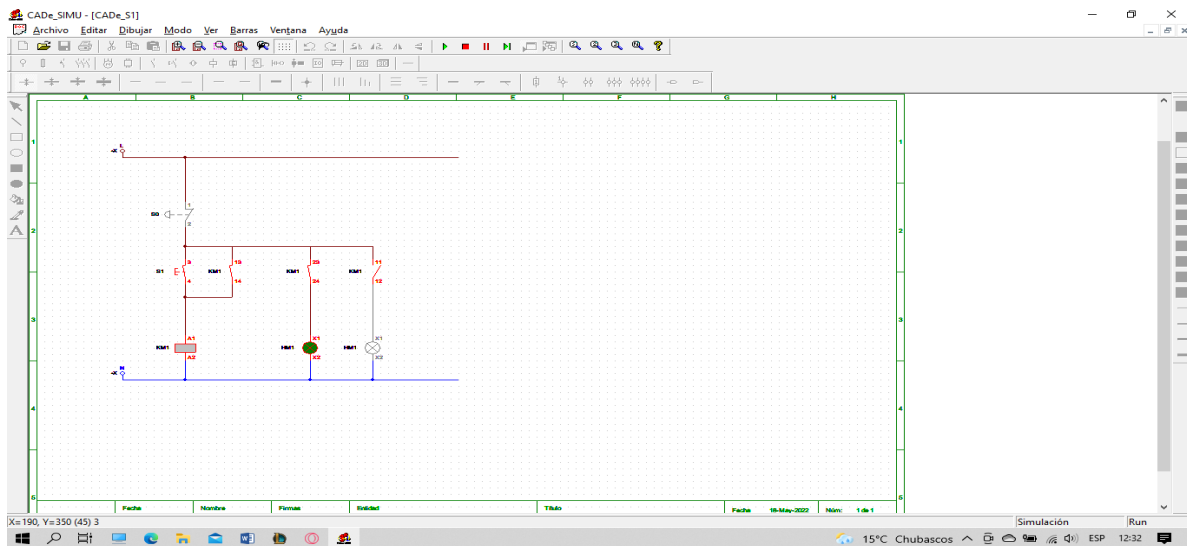


Nota. Esta figura representa el estado inicial del circuito de control.

- Para el accionamiento de la bobina del relé se usó un contacto normalmente abierto del mismo, este hará la función de memoria lo que permite que se mantenga encendida la bobina. Al momento de accionar el pulsador verde se apagará la luz piloto roja y se encenderá la luz piloto verde.

Figura 32

Software para construcción de circuitos neumáticos



Nota. Esta figura representa el circuito de control funcionando.

Cálculos para el Diseño Mecánico

Cada ecuación presentada es todas las necesarias para que la tolva sea diseñada, empezamos sacando datos generales que serán reemplazados en las fórmulas

Datos generales

Diámetro: 17 cm

Altura: 21,5 cm

Altura 2: 9.5 cm

Diámetro 2: 7 cm

Densidad de frejol: 14.4 g/cm³

Volumen 1 de la tolva. - para el cálculo del volumen se necesita la siguiente fórmula:

$$V = \frac{\pi}{4} * D^2 * H \quad \text{Ecu 1}$$

Donde

D = Diámetro 1

H = Altura 1

Volumen 2 de la tolva.

$$V2 = \frac{\pi}{12} * h (D^2 + d^2 + D * d) \quad \text{Ecu 2}$$

Donde

D = Diámetro 1

d = Diámetro 2

h = Altura 2

Volumen total. - Se aplicó la siguiente fórmula

$$Vt = V1 + V2 \quad \text{Ecu 3}$$

Donde

V1 = Volumen 1

V2 = Volumen 2

Capacidad de la tolva. – la siguiente fórmula se usó para conocer la capacidad que tiene la tolva

$$C = Vt * \rho * e \quad \text{Ecu 4}$$

Donde

Vt = Volumen total

p = densidad

e = error

Cálculo para la Escala de la Báscula con Módulo HX711

Para la calibración de esta balanza, primero se tiene que programar en el software Arduino. Luego de la programación se abre el Monitor Serie y con los valores que arroja se hace el cálculo aplicando con la siguiente fórmula.

$$Escala = \frac{Valor\ de\ Lectura}{Peso\ Real} \quad \text{Ecu 5}$$

Cálculos para la Tolva

Utilizando la ecuación 1 se procede a realizar cálculo de la tolva considerando los siguientes datos:

Diámetro: 17 cm

Altura: 21,5 cm

Altura 2: 9.5 cm

Diámetro 2: 7 cm

Densidad de frejol: 14.4 g/cm³

Volumen 1 de la tolva

Para calcular el volumen se usó la siguiente fórmula.

$$V = \frac{\pi}{4} * D^2 * H$$

$$V = \frac{\pi}{4} * 17^2 cm * 21.5 cm$$

$$V = 4800.07 cm^3$$

$$V = 0,00488 m^3$$

Volumen 2 de la Tolva

Seguidamente se utilizó la ecuación 2 para obtener el valor del segundo volumen.

$$V2 = \frac{\pi}{12} * h (D^2 + d^2 + D * d)$$

$$V2 = \frac{\pi}{12} * 9.5 \text{ cm} (17^2 \text{ cm} + 7^2 \text{ cm} + 17 \text{ cm} * 7 \text{ cm})$$

$$V2 = 1136,60 \text{ cm}^3$$

$$V2 = 0,00113 \text{ m}^3$$

Volumen Total de la Tolva

Para conocer el volumen total se suman el volumen 1 y 2, aplicando la aplicación 3.

$$Vt = V1 + V2$$

$$Vt = 0.00488 \text{ m}^3 + 0,00113 \text{ m}^3.$$

$$Vt = 0.00601 \text{ m}^3$$

Capacidad de la Tolva

Para determinar la capacidad de la tolva se utilizará la ecuación 4.

$$C = Vt * \rho * e$$

$$C = 0.00601 \text{ m}^3 * 242 \frac{\text{kg}}{\text{m}^3} * 0.92$$

$$C = 1,33 \text{ Kg}$$

Cálculo para la Escala de la Báscula

Es necesario calcular la escala de la báscula para que pueda entregar valores de masa reales, para ello se usó la siguiente fórmula

$$\text{Escala} = \frac{\text{Valor de Lectura}}{\text{Peso Real}}$$

$$\text{Escala} = \frac{162236}{5}$$

$$\text{Escala} = 32447.2$$

Selección de Componentes

Los materiales que se usaron para la realización del dosificador de granos secos son:

Acrílico

Se usó el acrílico como material para la estructura con el propósito de poder observar todo el proceso de dosificación, además el acrílico llega a ser un material liviano.

Figura 33

Acrílico



Nota. Esta figura muestra el material usado para la estructura del dosificador. Tomado de Acrílicos B13, Acrílico, s.f., <https://acrilicosb13.com.co/portfolio-item/acrilico/>.

Placa Arduino Mega

Se seleccionó la placa Arduino mega debido a que posee 53 pines digitales y 16 pines analógicos, lo que la convierte en una placa completa para el control de muchos elementos electrónicos. Además, Arduino puede llegar a ser muy versátil para la programación, y posee muchos elementos compatibles con la placa.

Figura 34

Placa Arduino Mega



Nota. Esta figura muestra la placa Arduino que se utilizó.

Servomotor MG 996R

Se seleccionó el servomotor MG 996R debido a la facilidad que posee al controlar el eje. La única función que cumple es abrir y cerrar la compuerta para que los granos puedan caer del túnel de dosificación. No se realizó ningún cálculo para la selección del servomotor, ya que no hace mayor esfuerzo y el material que levanta es ligero.

Figura 35

Servomotor MG 996R



Nota. Esta figura muestra el servomotor usado para el proyecto.

Báscula de 1KG con Módulo HX 711

La selección de la báscula de 1 kg se debe a que el proyecto se diseñó con la capacidad de dosificar onzas y máximo libras, ya que, si se dosifica mucho más que una libra, los granos secos causarían fallas en el actuador y en la báscula. El módulo HX 711 vino incluido con la celda de carga.

Figura 36

Software para construcción de circuitos neumáticos



Nota. Esta figura muestra el módulo HX 771 y la celda de carga de 1Kg que se utilizó para el proyecto.

Driver L298N

El driver L298N o el puente H se selección por dos razones, la primera es que se necesitaba un driver para realizar el cambio de giro del actuador lineal y del motor encargado de realizar la dosificación, la segunda razón es que el puente H tiene la capacidad de poder manejar dos motores al mismo tiempo. lo que ayuda a comandar a las dos máquinas. Además, el driver llega a ser compatible con Arduino, lo que facilita al momento de realizar la programación en el software Arduino.

Figura 37

Driver L298N o Puente H



Nota. Esta figura muestra el driver que se utilizó para el proyecto.

Sensor Infrarrojo

El principal motivo por el que se seleccionó el sensor infrarrojo fue porque dio mejor resultado en las pruebas para comprobar si la tolva está llena o no. Además, el sensor es compatible con Arduino y posee varios ejemplos que pueden servir de guía para realizar la respectiva programación.

Figura 38

Software para construcción de circuitos neumáticos



Nota. La imagen muestra el sensor infrarrojo usado para el proyecto.

Motor de CC de 12V

El motor fue seleccionado mediante prueba y error, al inicio se pensó usar un motor paso a paso, pero no tenía la suficiente fuerza para poder mover el mecanismo de la tolva. El motor de 12 v posee la suficiente fuerza para poder mover el mecanismo

Figura 39

Motor de 12 VCC



Nota. La figura muestra el motor usado para el proyecto Extraído de Ardunel, Motor de 12 VCC, s.f., https://ventas-lapaz.ardunel.com.bo/index.php?id_product=134&rewrite=motor-6v-dc-1360rpm-&controller=product.

Actuador Lineal a 12 VCC

Es un tipo de actuador que crea movimiento en línea recta y posee la potencia suficiente para poder empujar los granos que caen de la tolva, esa es el principal motivo por lo que se seleccionó el actuador lineal.

Figura 40

Actuador lineal a 12VCC



Nota. La figura muestra el actuador usado para el proyecto. Extraído de Luis Llama, Actuador líneas a 12 VCC, s.f., <https://www.luisllamas.es/arduino-actuador-lineal/>.

Fuente de 12V

El principal motivo por lo que se escogió la fuente es porque el actuador lineal y el motor funcionan a 12V

Figura 41

Fuente de 12VCC



Nota. La figura muestra la fuente de 12VCC que se usó para el proyecto.

Programación del Arduino Mega

Calibración de la Báscula

El primer elemento electrónico que se programa es la celda de carga con el módulo respectivo. Con la primera programación se obtiene datos mediante Monitor Serie. Con los valores obtenidos se hace un cálculo para encontrar la escala, después de obtener el resultado del cálculo, se usa una segunda programación introduciendo el resultado del cálculo

Figura 42

Primera programación en el software Arduino para la báscula

The screenshot shows the Arduino IDE interface. On the left, the sketch editor displays the following code:

```

1 #include "HX711.h"
2 const int DOUT=A1;
3 const int CLK=A0;
4
5
6 HX711 balanza;
7 void setup() {
8   Serial.begin(9600);
9   balanza.begin(DOUT, CLK);
10  Serial.print("Lectura del valor del ADC:");
11  Serial.println(balanza.read());
12  Serial.println("No ponga ningún objeto sobre la balanza");
13  Serial.println("Destazando...");
14  balanza.set_scale(); //La escala por defecto es 1
15  balanza.tare(20); //El peso actual es considerado Tara.
16  Serial.println("Coloque un peso conocido");
17 }
18
19 void loop() {
20
21  Serial.print("Valor de lectura: t");
22  Serial.println(balanza.get_value(10,0));
23  delay(100);
24 }

```

On the right, the Serial Monitor window shows the following output:

```

Valor de lectura: t-162835
Valor de lectura: t-162611
Valor de lectura: t-162615
Valor de lectura: t-162653
Valor de lectura: t-162159
Valor de lectura: t-162428
Valor de lectura: t-162236
Valor de lectura: t-162835
Valor de lectura: t-162561
Valor de lectura: t-162660
Valor de lectura: t-162439
Valor de lectura: t-162706
Valor de lectura: t-162791
Valor de lectura: t-162709
Valor de lectura: t

```

At the bottom of the IDE, a status bar indicates: "El Sketch usa 4694 bytes (14%) del espacio de almacenamiento de programa. El máximo es 32256 bytes. Las variables Globales usan 359 bytes (17%) de la memoria dinámica, dejando 1689 bytes para las variables locales. El máximo es 2048 bytes."

Nota. En la figura se muestra la primera programación para la báscula.

Para terminar de calibrar la báscula, se hizo una programación más para colocar el resultado de la conversión para la escala y de esa forma lograr que los valores arrojados en el monitor serial sean en gr, onza o Kg. Como último se realiza una prueba con pesos conocidos para poder comprobar que los valores sean correctos.

Las pruebas se realizaron con cuatro fundas de granos secos de pesos diferentes, la primera es 50 gramos, las segundas de 100 gramos, las terceras de 150 gramos y la cuarta de 200 gramos.

Figura 43

Segunda programación en el software Arduino para la báscula



```

sketch_mar12a Arduino 1.0.13
Archivo Editor Programa Herramientas Ayuda
sketch_mar12a.g
1 #include "HX711.h"
2
3 const int DOUT=A1;
4 const int CLK=A0;
5
6 HX711 balanza;
7
8 void setup() {
9   Serial.begin(9600);
10  balanza.begin(DOUT, CLK);
11  Serial.println("Lectura del valor del ADC: ");
12  Serial.println(balanza.read());
13  Serial.println("No ponga ningun objeto sobre la balanza");
14  Serial.println("Destarando...");
15  Serial.println("...");
16  balanza.set_scale(32447.4); // Establecemos la escala
17  balanza.tare(20); //El peso actual es considerado Tara.
18
19  Serial.println("Listo para pesar");
20 }
21
22 void loop() {
23   Serial.println("Peso: ");
24   Serial.println(balanza.get_units(20),3);
25   Serial.println(" onzS");
26   delay(500);
27 }

```

Compartido

El Sketch usa 3256 bytes (16%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 343 bytes (16%) de la memoria dinámica, dejando 1705 bytes para las variables locales. El máximo es 2048 bytes.

27 Arduino Uno - COM5

Nota. En la figura se muestra la programación final para la calibración de la báscula.

Calibración del Sensor Infrarrojo

Luego de la calibración de la báscula, se programó el sensor infrarrojo con una distancia inicial de 4 cm hasta 30 cm de profundidad y de esa forma verificar la distancia correcta que tiene que tener el sensor infrarrojo y los granos secos para que al momento de verificar si hay o no hay producto en la tolva pueda arrojar los datos correctos y poder para o continuar el proceso de dosificación. Para programar el sensor infrarrojo no se instaló ninguna librería, ya que Arduino es compatible con este tipo de sensor.

Figura 44

Programación del sensor infrarrojo



```

1 const int sensorPin = 9;
2
3 void setup() {
4   Serial.begin(9600); //iniciar puerto serie
5   pinMode(sensorPin , INPUT); //definir pin como entrada
6 }
7
8 void loop() {
9   int value = 0;
10  value = digitalRead(sensorPin); //lectura digital de pin
11
12  if (value == HIGH)
13  {
14    Serial.println("NO HAY OBSTACULO");
15  }
16  delay(1000);
17
18  if (value == LOW)
19  {
20    Serial.print("DETECTANDO OBSTACULO");
21  }
22  delay(1000);
23 }

```

Compilado:
El Sketch usa 2022 bytes (6%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 226 bytes (11%) de la memoria dinámica, dejando 1922 bytes para las variables locales. El máximo es 2048 bytes.

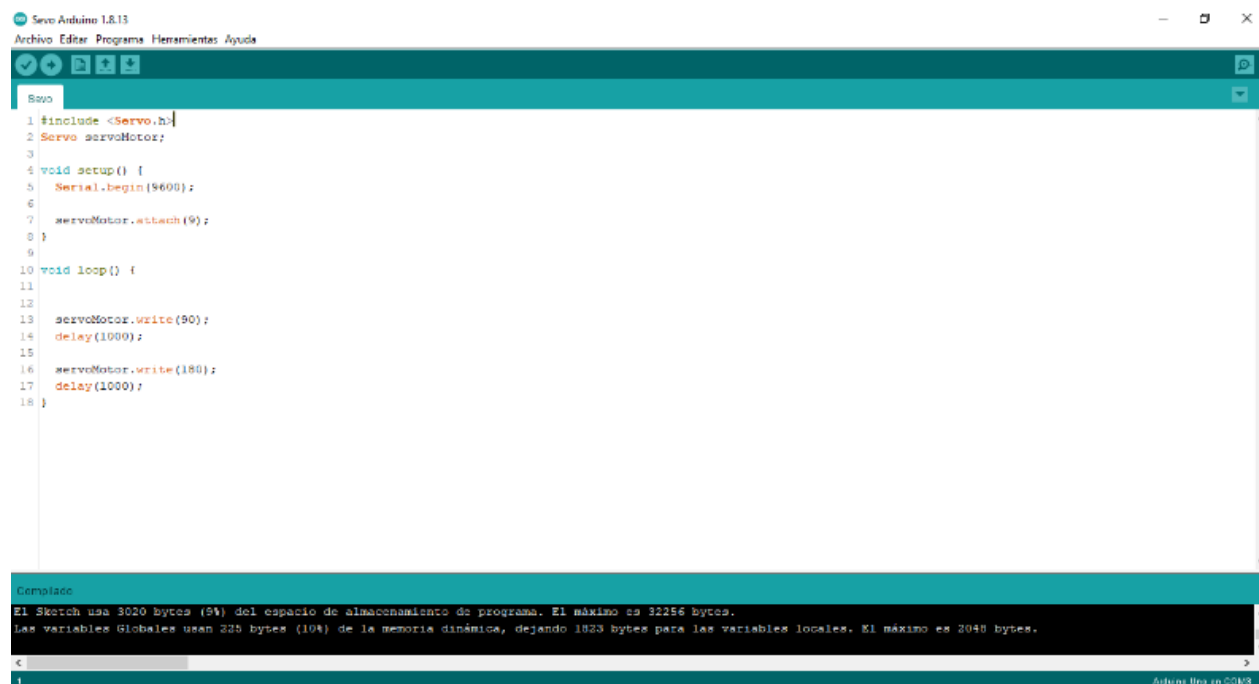
Nota. En la figura se muestra la programación para la calibración del sensor infrarrojo.

Programación del Servomotor

Después se programó el movimiento del servomotor para que se abra o se cierre la compuerta del túnel dosificador y de esa forma permitir que los granos secos caigan al recipiente. De igual forma la compuerta ayuda a que no se desparrame el producto al momento de que se están pesando los granos caídos a la báscula. Los ángulos que se usaron para que la compuerta pueda abrir son de 90° para la posición cerrada y para la posición contraria se usó un Angulo de 180°. El pin usado para el servomotor es el 9, el tiempo que se estableció para que la compuerta se abra o se cierre es de 1000 milisegundos o un segundo. La librería usada para programar el motor es Servo.h, la cual ya viene incluida en el software de Arduino. Tal como en la siguiente figura:

Figura 45

Programación del servomotor



```

1 #include <Servo.h>
2 Servo servoMotor;
3
4 void setup() {
5   Serial.begin(9600);
6
7   servoMotor.attach(9);
8 }
9
10 void loop() {
11
12
13   servoMotor.write(90);
14   delay(1000);
15
16   servoMotor.write(180);
17   delay(1000);
18 }

```

Compiace

El Sketch usa 3020 bytes (9%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
 Las variables Globales usan 225 bytes (10%) de la memoria dinámica, dejando 1823 bytes para las variables locales. El máximo es 2048 bytes.

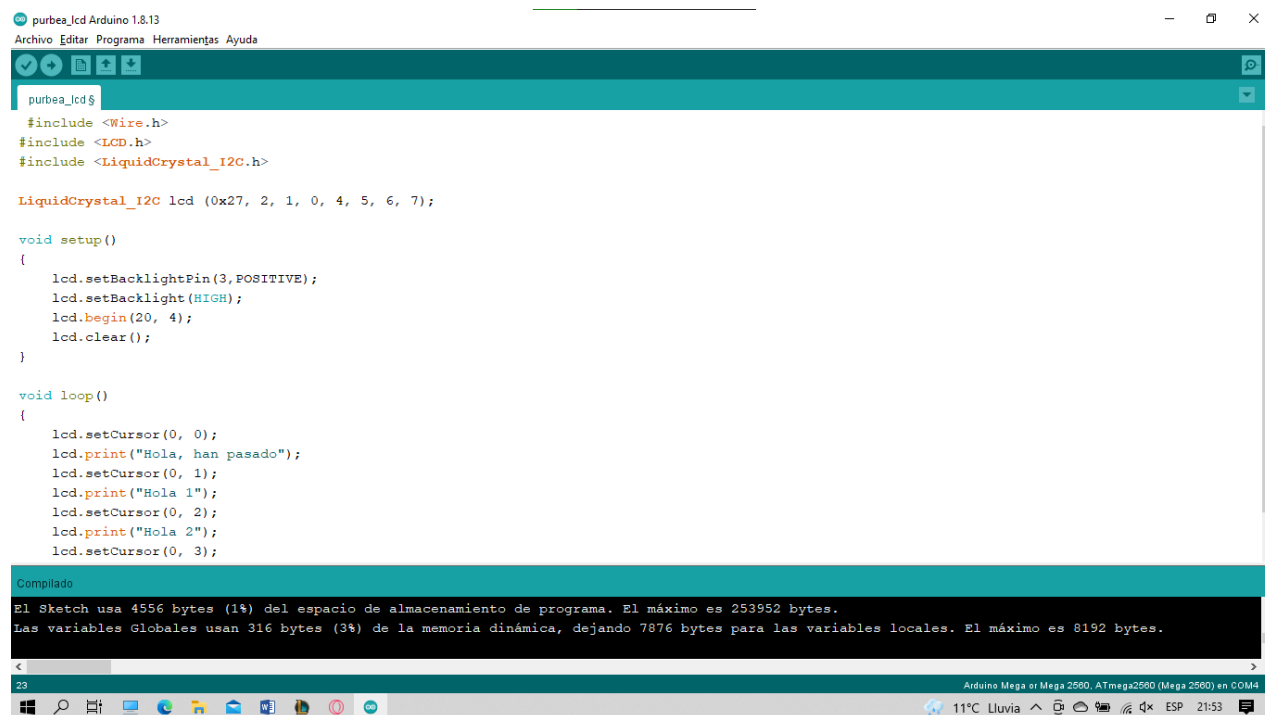
Arduino Uno en COM3

Nota. En la figura se muestra la programación realizada para la programación del servomotor.

Por último, para la programación de la LCD se instalaron 3 librerías ya que se usó el adaptador para el display y de esa forma evitar el uso excesivo de pines de la placa Arduino Mega. Las librerías instaladas son Wire.h, LCD.h y LiquidCrystal.h. Antes de realizar la programación final para la pantalla, primero se hizo un programa básico para comprobar que las librerías, el adaptador y la LCD funcionen correctamente. Se inició con un texto de Hola Mundo, para luego avanzar con variables más complejas relacionadas con el proyecto de aplicación práctica y por último se estructuró la carátula que aparece al momento de iniciar el sistema, el aviso cuando el sensor detecte que no hay producto en la tolva y el peso que arroja la celda de carga con el módulo HTX711.

Figura 46

Programación del Display



```

purbea_lcd Arduino 1.8.13
Archivo Editar Programa Herramientas Ayuda

purbea_lcd$
#include <Wire.h>
#include <LCD.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd (0x27, 2, 1, 0, 4, 5, 6, 7);

void setup()
{
  lcd.setBacklightPin(3, POSITIVE);
  lcd.setBacklight(HIGH);
  lcd.begin(20, 4);
  lcd.clear();
}

void loop()
{
  lcd.setCursor(0, 0);
  lcd.print("Hola, han pasado");
  lcd.setCursor(0, 1);
  lcd.print("Hola 1");
  lcd.setCursor(0, 2);
  lcd.print("Hola 2");
  lcd.setCursor(0, 3);
}

Compilado
El Sketch usa 4556 bytes (1%) del espacio de almacenamiento de programa. El máximo es 253952 bytes.
Las variables Globales usan 316 bytes (3%) de la memoria dinámica, dejando 7876 bytes para las variables locales. El máximo es 8192 bytes.

```

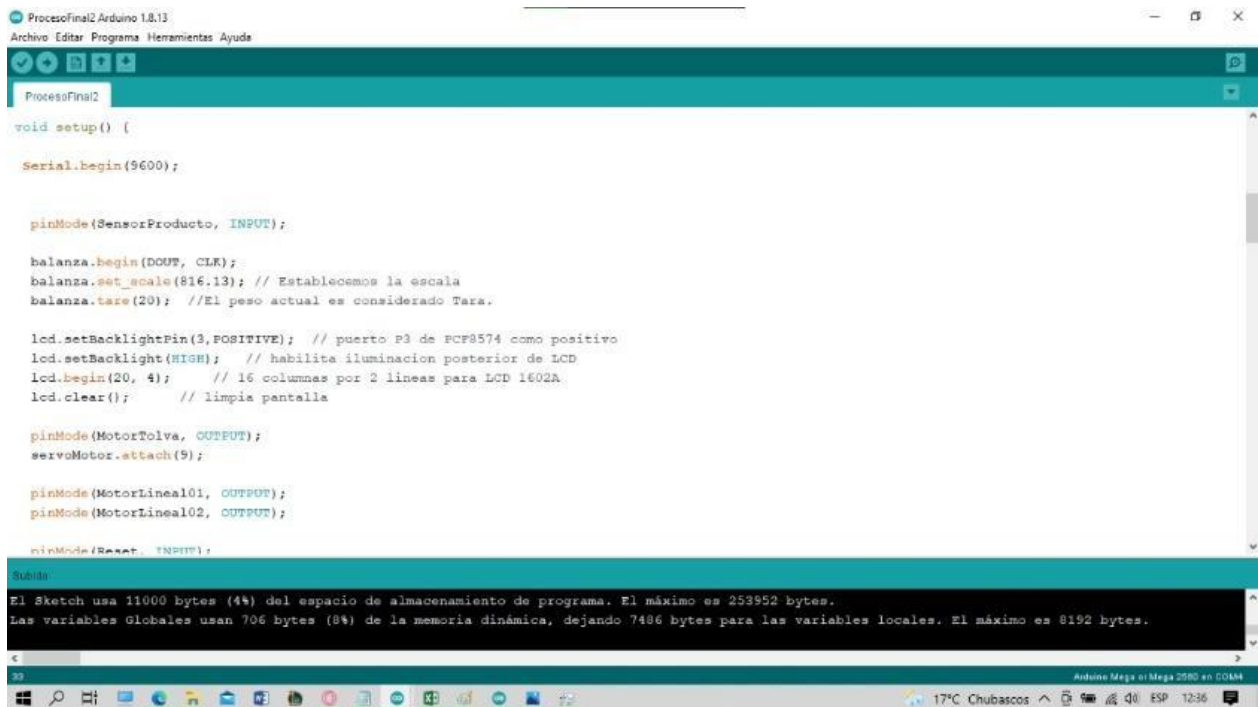
Nota. En la figura se muestra la programación realizada para la programación del display.

Programación Final

Ya como última programación se unió las anteriores partes realizadas de cada elemento electrónico. Al momento de realizar la compilación del código, saltaron errores, por lo que se tuvo que hacer cambios en la programación del Servomotor, de la celda de carga y el módulo HX711, el motor encargado de dosificar y el actuador lineal, además se tuvo que incluir dos fines de carrera por dos razones, la primera es para evitar el uso de la función Delay, y la segunda razón para que el sistema llegue a ser más exacto al momento de dosificar, pesar, empujar el producto y abrir o cerrar la compuerta. Además, se usaron funciones para que se pueda reflejar las diferentes variables del sistema y las alertas.

Figura 47

Primera programación final



```
ProcesoFinal2 Arduino 1.8.13
Archivo Editar Programa Herramientas Ayuda

ProcesoFinal2

void setup() {

  Serial.begin(9600);

  pinMode(SensorProducto, INPUT);

  balanza.begin(DOUT, CLK);
  balanza.set_scale(816.13); // Establecemos la escala
  balanza.tare(20); //El peso actual es considerado Tara.

  lcd.setBacklightPin(3, POSITIVE); // puerto P3 de PCF8574 como positivo
  lcd.setBacklight(HIGH); // habilita iluminacion posterior de LCD
  lcd.begin(20, 4); // 16 columnas por 2 lineas para LCD 1602A
  lcd.clear(); // limpia pantalla

  pinMode(MotorTolva, OUTPUT);
  servoMotor.attach(9);

  pinMode(MotorLineal01, OUTPUT);
  pinMode(MotorLineal02, OUTPUT);

  pinMode(Reset, INPUT);
}

Subida
El Sketch usa 11000 bytes (4%) del espacio de almacenamiento de programa. El máximo es 253952 bytes.
Las variables Globales usan 706 bytes (8%) de la memoria dinámica, dejando 7486 bytes para las variables locales. El máximo es 8192 bytes.
```

Nota. En la figura se muestra la programación usada para el funcionamiento del sistema. Para más información ir a Anexos.

Conclusiones

Para el diseño del sistema dosificador se utilizaron motores reductores de 35 rpm a 25 N/m el cual permite el libre movimiento de los granos secos a través del túnel garantizando así el correcto funcionamiento del sistema.

Para la visualización del peso de granos secos se utilizó una LCD de cristal líquido con interfaz I2C la cual permite la codificación de los datos obtenidos por la báscula que serán enviados a los pines SDA y SCL.

En el ensamblaje de las partes de la estructura se presentó un contratiempo, ya que el material usado para unir cada parte no fue bien seleccionado, este inconveniente fue resuelto mediante consultas acerca del material. En la programación no hubo ningún problema, debido a que se siguieron las instrucciones de cada elemento electrónico.

En las pruebas de funcionamiento la báscula no pesaba correctamente debido a que se cometieron errores de cálculo, esto fue solucionado calibrando una vez más a la báscula y realizando de nuevo los cálculos. Otro elemento que presentó problemas fue el actuador lineal, al momento de empujar los granos secos, lo hacía de forma rápida y no caían todos los granos secos en el recipiente. Esto fue solucionado cambiando valores en la programación.

Recomendaciones

Al inicio el proyecto se diseñó sin fórmulas, por lo que presentó varios problemas en la capacidad de almacenar granos secos. Al momento de cargar los granos secos, hubo mucho peso que afectó el funcionamiento de la hélice encargada de soltar los granos.

Es recomendable realizar cálculos de volumen para conocer el peso y de esta manera seleccionar la fuerza y torque necesario para el motor con el fin de garantizar el correcto funcionamiento de los sistemas de dosificación, evitando atascamientos por el sobre esfuerzo.

Para realizar estructuras en acrílico es necesario conocer las características y propiedades del material ya que de esta manera se podrá conocer los adhesivos para la unión y fijación, por esta razón es recomendable utilizar un pegamento AD-CRYL PG plus evitando el manchado ya que son pegamentos transparentes.

Al momento de iniciar con las pruebas de funcionamiento la báscula arrojaba datos erróneos al momento de pesar, por lo que se recomienda verificar las conexiones, también verificar más de dos veces la fórmula usada y los cálculos realizados, ya que estos factores pueden afectar mucho al momento del funcionamiento.

Referencias Bibliográficas

- Agudelo N. Tano G. Vargas C. (2018). Historia de la automatización. Bogotá Mathieu M. (2014). Instrucción a la programación. México
- Bandala. R. (2018). Sensor infrarrojo. Emisor y receptor.
https://www.academia.edu/16922918/SENSOR_INFRRARROJO_Teoria_y_practica
- Branete A, Segundo P, Herrero R. (2018). Automatización Industrial. España. Madrid
- Cáceres L. (2019). Introducción a la programación, fundamentos de la programación, metodología de la programación, aplicaciones. Lima. Perú.
- Castilla, J. (2019). Sistemas de Control Automático.
<http://ieshuelin.com/huelinwp/download/Tecnologia/Tecnologia%20industrial/3-SISTEMAS-AUTOMÁTICOS-DE-CONTROL-ampliacion.pdf>
- Cazares E. Garrido O. (2018). Báscula dosificadora de café. México. Puebla
- Cruz. E. (2018). Finales de Carrera. <https://es.scribd.com/document/73025528/Finales-de-Carrera>
- Gómez D. Parra J (2021). Diseño e implementación de una dosificadora de condimentos para la línea de producción de la compañía Trébol Rojo S.A.S.
[456789/3098/1/2021BrayanJulianParraTorres.pdf](https://www.scribd.com/document/456789/3098/1/2021BrayanJulianParraTorres.pdf)
- Gonzales, A (2018). ¿Qué es y cómo funciona un servomotor?
<https://www.bolanosdj.com.ar/MOVIL/ARDUINO2/IntroServos.pdf>

Jorge L, Patiño V (2018). Máquinas Eléctricas. Motores de corriente continua.

<file:///C:/Users/Pc%20->

[%20Personal/Downloads/MÓDULO%20N°1%20DE%20MÁQUINAS%20ELÉCTRICAS\(TALLER%20V\)%20MOTORES%20DE%20C.C-2018%20\(1\).pdf](#)

Lozada. J (2018). Investigación aplicada. Definición, propiedad intelectual e industrial.

<https://dialnet.unirioja.es/servlet/articulo?codigo=6163749>

Morales. E (2018). Diseño y construcción de un dosificador de producto alimenticio molido y

seco de 50 GR. Ecuador. Latacunga Naylamp Mechatronic (2018). Calibración de
bascula con módulo HX7711. https://naylampmechatronics.com/blog/25_tutorial-trasmisor-de-celda-de-carga-hx711-balanza-digital.html

Oviedo A. (2018). Diseño y construcción de un dosificador de granos secos.

<https://repositorio.pucesa.edu.ec/bitstream/123456789/684/1/80123.pdf>

Sarzosa A. Pasuña I. Tahhan F. (2018) Dosificador. UTE Ecuador.

Sigüenza G (2018). Curso Básico de Circuitos Eléctricos de Control Industrial.

<https://www.industrialtijuana.com/pdf/cursogratis/F03PRESENTCIRCUITOS.pdf>

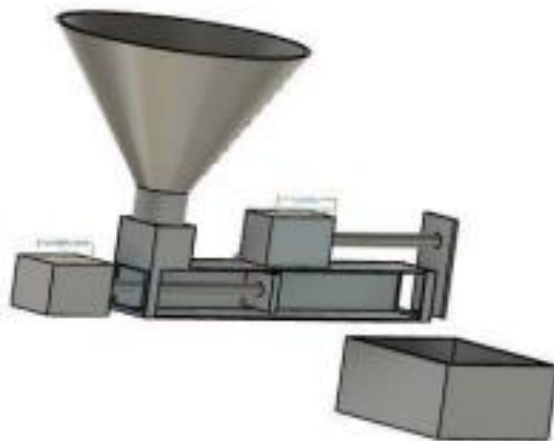
Vallejo M. (2020). Introducción a los sensores.

https://tableroalparque.weebly.com/uploads/5/1/6/9/51696511/introducciÓn_a_los_sensores.pdf

Anexos

Anexo 1

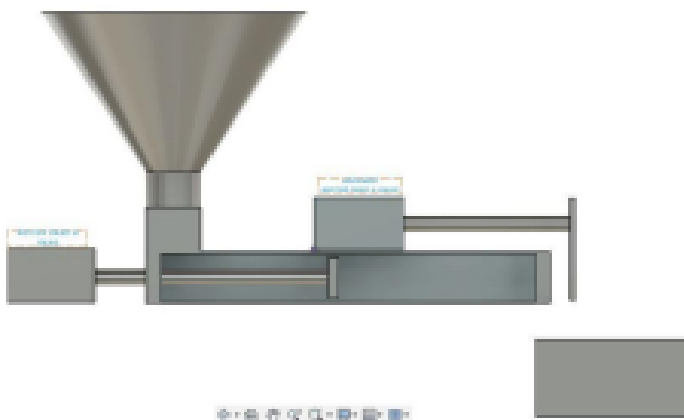
Prototipo de dosificador



Nota. La figura muestra el prototipo que se va a realizar.

Anexo 2

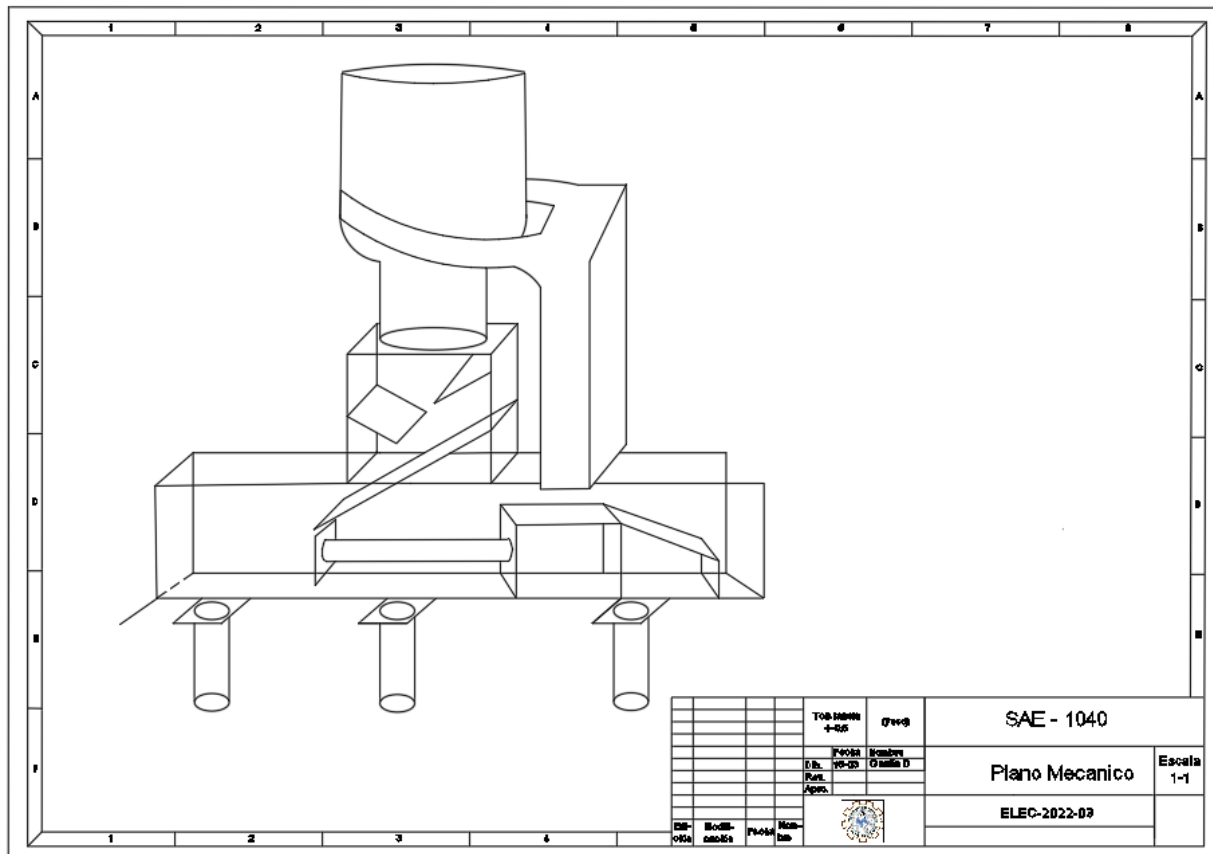
Prototipo de dosificador



Nota. La figura muestra el prototipo que se va a realizar y muestra una vista lateral.

Anexo 3

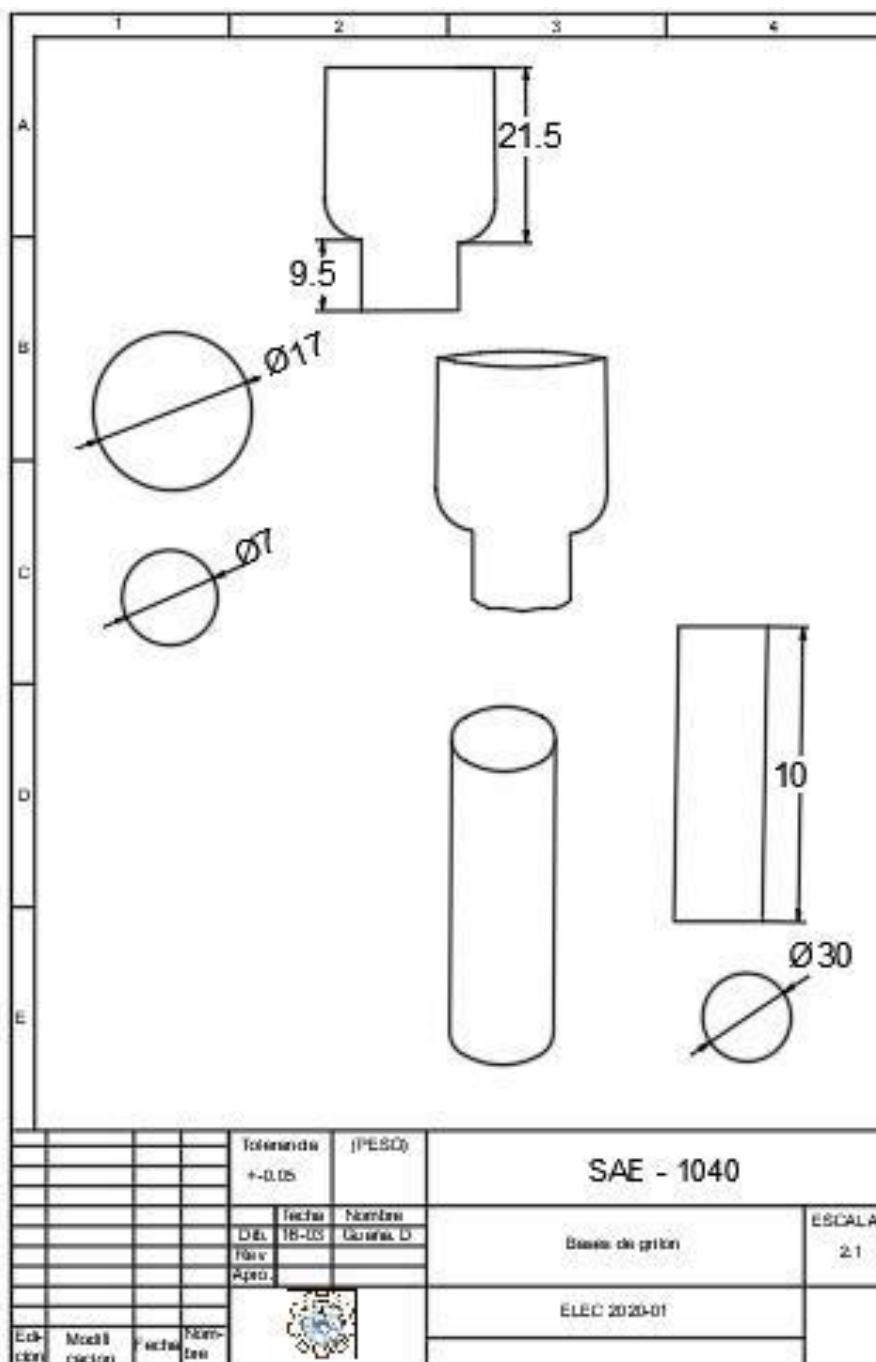
Plano mecánico del dosificador



Nota. En la figura se muestra el plano mecánico de la dosificadora.

Anexo 4

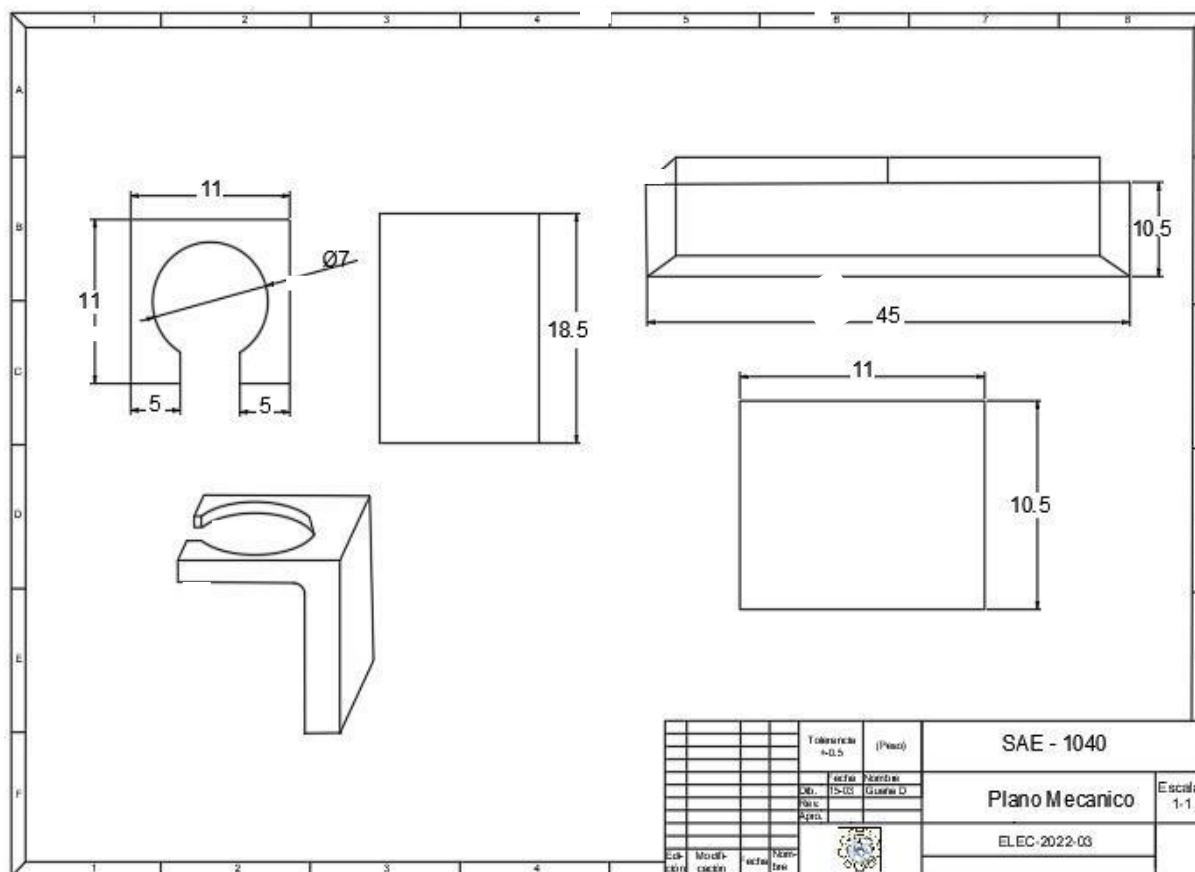
Medidas de las bases de Grillon y tolva



Nota. En la figura se muestra las medidas de las bases de Grillon y las medidas de la tolva.

Anexo 5

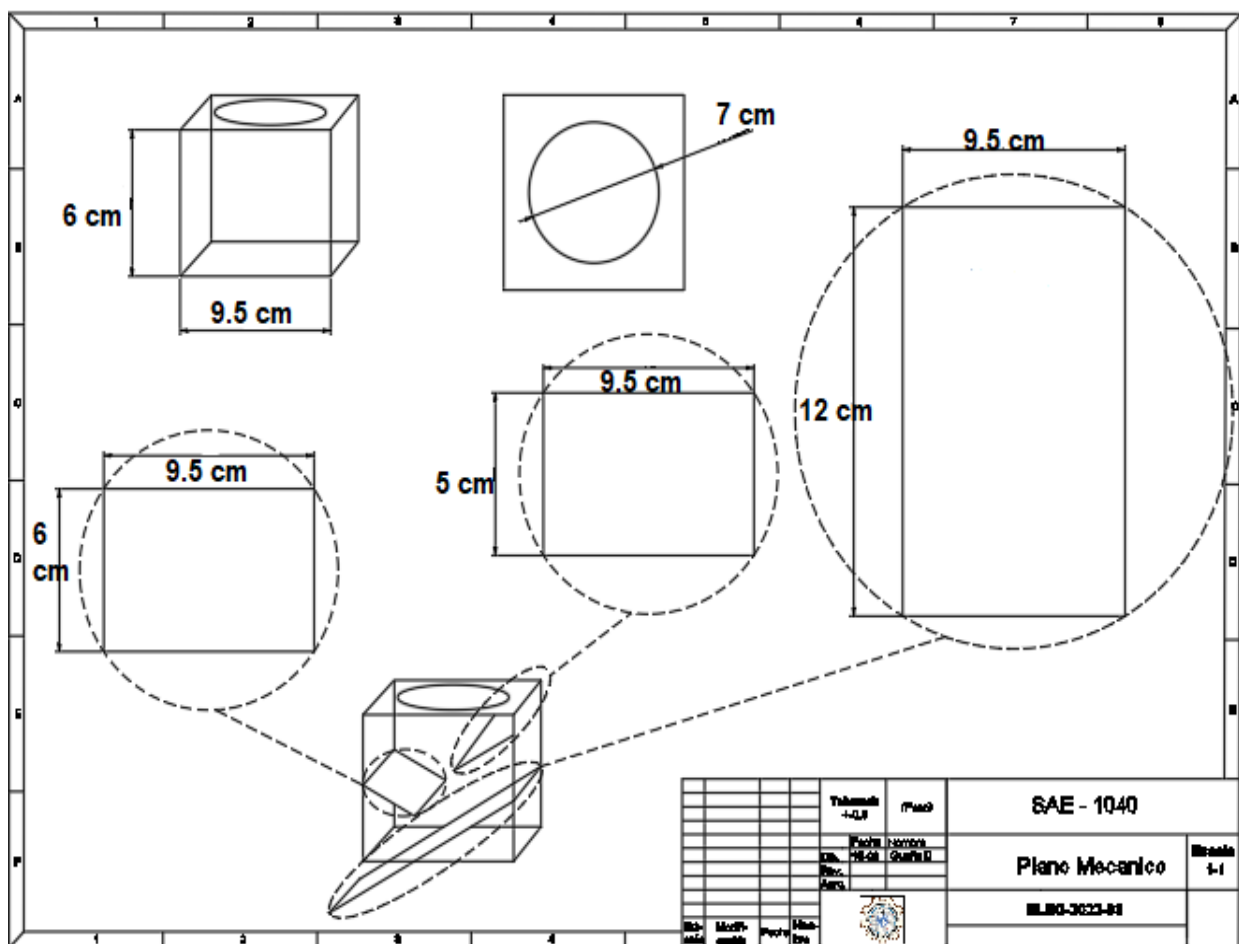
Tolva con medidas



Nota. En la imagen se muestra las medidas de la base de la tolva y el túnel dosificador.

Anexo 6

Medidas del túnel de caída



Nota. En la figura se muestran las medidas del túnel de caída.

Anexo 7

Programación Final

```

ProcesoFinal2 Arduino 1.8.13
Archivo Editar Programa Herramientas Ayuda

ProcesoFinal2
int sensor1=40;
int sensor2=42;

//variables

int Vreset=0;
int Vsensor1;
int Vsensor2;

void setup() {

  Serial.begin(9600);

  pinMode(SensorProducto, INPUT);

  balanza.begin(DOUT, CLK);
  balanza.set_scale(816.13); // Establecemos la escala
  balanza.tare(20); //El peso actual es considerado Tara.

  lcd.setBacklightPin(3,POSITIVE); // puerto P3 de PCF8574 como positivo
  lcd.setBacklight(HIGH); // habilita iluminacion posterior de LCD
  lcd.begin(20, 4); // 16 columnas por 2 lineas para LCD 1602A

#include <Servo.h>
#include "HX711.h"

#include <Wire.h> // libreria de comunicacion por I2C
#include <LCD.h> // libreria para funciones de LCD
#include <LiquidCrystal_I2C.h> // libreria para LCD por I2C

const int DOUT=A1;
const int CLK=A0;

HX711 balanza;
Servo servomotor;

LiquidCrystal_I2C lcd (0x27, 2, 1, 0, 4, 5, 6, 7); // DIR, E, RW, RS, D4, D5, D6, D7

int SensorProducto =30;
int VsensorProducto;

int MotorTolva=10;
int a1=1,a2=1,a3=1,a4=1,a5=1;
int MotorLineal01=11;
int MotorLineal02=12;
int Reset=48;
  lcd.clear(); // limpia pantalla

  pinMode(MotorTolva, OUTPUT);
  servomotor.attach(9);

  pinMode(MotorLineal01, OUTPUT);
  pinMode(MotorLineal02, OUTPUT);

  pinMode(Reset, INPUT);

  //inicio de actuadores
  servomotor.write(50);

  digitalWrite(MotorLineal01, HIGH); // turn the LED on (HIGH is the voltage level)
  digitalWrite(MotorLineal02, LOW);

  delay(3000);
}

void loop() {

VsensorProducto=digitalRead(SensorProducto);

```

```
lcd.clear(); // limpia pantalla

pinMode(MotorTolva, OUTPUT);
servoMotor.attach(9);

pinMode(MotorLineal01, OUTPUT);
pinMode(MotorLineal02, OUTPUT);

pinMode(Reset, INPUT);

//inicio de actuadores
servoMotor.write(90);

digitalWrite(MotorLineal01, HIGH); // turn the LED on (HIGH is the voltage level)
digitalWrite(MotorLineal02, LOW);

delay(3000);

}

void loop() {

VsensorProducto=digitalRead(SensorProducto);

//VA A PESAR -- TIEMPO PARA ESTABILIZARSE
delay(7000);

lcd.setCursor(0, 0);
lcd.print("Peso: ");
lcd.print(balanza.get_units(20),3);

Serial.print("Peso: ");
Serial.print(balanza.get_units(20),3);
Serial.println(" g");
delay(500);

//FIN DE PESADO

//SALE EL PISTON

digitalWrite(MotorLineal01, LOW);
digitalWrite(MotorLineal02, HIGH);

delay(2000);

//ABRE LA COMPUERTA
servoMotor.write(180);

while(a4==1){

Vsensor1=digitalRead(sensor1);
Vsensor2=digitalRead(sensor2);
delay(500);

if(Vsensor1==0&&Vsensor2==1){
a4=2;
}

}

digitalWrite(MotorLineal01, HIGH);
digitalWrite(MotorLineal02, LOW);

while(a5==1){

Vsensor1=digitalRead(sensor1);
```

```

    Vsensor1=digitalRead(sensor1);
    Vsensor2=digitalRead(sensor2);
    delay(500);

    if(Vsensor1==1&&Vsensor2==0){
        a5=2;
    }

}

a1=4;

} //FIN DEL WHILE CUANDO a1 == 3

if(a5==2){

    Serial.print("hola3");
    servoMotor.write(90);
    delay(2000);
    a5=3;

}

Vreset=digitalRead(Reset);

if(Vreset==0&&a1==1){

    Serial.println("hola00");
    a1=2;
    a2=1;
    a3=1;
    a4=1;
    a5=1;
    lcd.clear();
    //a1=1;

}

if(a5==3){

    Serial.println("hola20");
    a1=2;
    a2=1;
    a3=1;
    a4=1;
    a5=1;
    lcd.clear();
    //a1=1;

}

}

//digitalWrite(MotorTolva, HIGH); // turn the LED on (HIGH is the voltage level)
// delay(10000);

}

Subido
El Sketch usa 11000 bytes (4%) del espacio de almacenamiento de programa. El máximo es 253952 bytes.
Las variables Globales usan 706 bytes (8%) de la memoria dinámica, dejando 7486 bytes para las variables locales. El máximo es 8192 bytes.

```

Nota. En la figura se muestra toda la programación realizada para el dosificador.