

Instituto Tecnológico de Costa Rica
Área Académica de Ingeniería Mecatrónica



**Diseño de un sistema de verificación de peso de productos adquiridos en
comercios.**

**Informe de Proyecto de Graduación para optar por el título de Ingeniero en
Mecatrónica con el grado académico de Licenciatura**

Sergio Andrés Rodríguez Montero

Cartago, 25 de Noviembre de 2022



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/).

Declaro que el presente Proyecto de Graduación ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos propios.

En los casos en que he utilizado bibliografía, he procedido a indicar las fuentes mediante las respectivas citas bibliográficas.

En consecuencia, asumo la responsabilidad total por el trabajo de graduación realizado y por el contenido del correspondiente informe final.

Cartago, 04 de noviembre del 2022


Sergio Andrés Rodríguez Montero
Céd: 3 0506 0013

INSTITUTO TECNOLÓGICO DE COSTA RICA
PROGRAMA DE LICENCIATURA EN INGENIERÍA MECATRÓNICA
PROYECTO FINAL DE GRADUACIÓN
ACTA DE APROBACIÓN

El profesor asesor del presente trabajo final de graduación, indica que el documento presentado por el estudiante cumple con las normas establecidas por el programa de Licenciatura en Ingeniería Mecatrónica del Instituto Tecnológico de Costa Rica para ser defendido ante el jurado evaluador, como requisito final para aprobar el curso Proyecto Final de Graduación y optar así por el título de Ingeniero(a) en Mecatrónica, con el grado académico de Licenciatura.

Estudiante: Sergio Andrés Rodríguez-Montero

Proyecto: Diseño de un sistema de verificación de peso de productos adquiridos en comercios

X Iván Araya Meneses

MSc. -Ing. Iván Araya Meneses

Asesor

Cartago, 25 Noviembre 2022

INSTITUTO TECNOLÓGICO DE COSTA RICA
PROGRAMA DE LICENCIATURA EN INGENIERÍA MECATRÓNICA
PROYECTO FINAL DE GRADUACIÓN
ACTA DE APROBACIÓN

Proyecto final de graduación defendido ante el presente jurado evaluador como requisito para optar por el título de Ingeniero(a) en Mecatrónica con el grado académico de Licenciatura, según lo establecido por el programa de Licenciatura en Ingeniería Mecatrónica, del Instituto Tecnológico de Costa Rica.

Estudiante: Sergio Andrés Rodríguez Montero

Proyecto: Diseño de un sistema de verificación de peso de productos adquiridos en comercios

Miembros del jurado

X



MSc. -Ing. Marta Vilchez Monge

Jurado

X



MSc. -Ing. Felipe Meza Obando

Jurado

X



Ing. Ana María Murillo Morgan

Jurado

Los miembros de este jurado dan fe de que el presente proyecto final de graduación ha sido aprobado y cumple con las normas establecidas por el programa de Licenciatura en Ingeniería Mecatrónica.

Cartago, 25 Noviembre 2022

Resumen

En el siguiente informe de Proyecto Final de Graduación se presenta el diseño e implementación de un sistema de pesaje, para la comprobación de pesos de productos en comercios con el objetivo de implementar un sistema de cobro automatizado y agilizar este proceso en los comercios.

Los componentes que integran el sistema desarrollado son componentes mecánicos, como el diseño de una plataforma de pesaje, la cual se validó por medio de simulación, un factor de seguridad adecuado. Componentes eléctricos, como la implementación de un sistema eléctrico de pesaje con una repetibilidad de medición no mayor a 50g, con celdas de carga y un controlador PLC para la adquisición de los pesos medidos en tiempo real vía comunicación serial ASCII con un tiempo de respuesta que no supera los 1.7 s y finalmente la componente de software de acceso a la web, el cual fue programado en C# y se utilizó el servicio de las bases de datos para poner a disposición la información leída del hardware vía comunicación Modbus TCP hacía la red de internet.

Se planteó un análisis económico derivado del beneficio del proyecto en donde se consideran beneficios suaves y duros e indicadores financieros para evaluar la rentabilidad del proyecto, como lo son el TIR Y VAN.

Palabras clave: Factor de seguridad, Celdas de carga, PLC, Comunicación ASCII, Comunicación Modbus TCP, Bases de Datos, API.

Abstract

The following Final Graduation Project report presents a design and implementation of a weighing system for checking product weights in stores with the aim of implementing an automated collection system and streamlining this process in stores.

The components that make up the developed system are mechanical components, such as the design of a weighing platform, which was validated by means of simulation, an adequate safety factor. Electrical components, such as the implementation of an electrical weighing system with a measurement repeatability of no more than 50g, with load cells and a PLC controller for the acquisition of the weights measured in real time via ASCII serial communication with a response time that does not exceed 1.7 s and finally the web access software component, which was programmed in C # and the database service was used to make available the information read from the hardware via Modbus TCP communication to the internet network.

An economic analysis derived from the benefit of the project was proposed, where soft and hard benefits and financial indicators are considered to evaluate the profitability of the project, such as the TIR and VAN.

Keywords: Safety factor, Load cells, PLC, ASCII communication, Modbus TCP communication, Databases, API.

INDICE DE CONTENIDOS

| | |
|--|-----|
| INDICE DE FIGURAS | iii |
| INDICE DE TABLAS | v |
| LISTA DE ABREVIACIONES | vii |
| 1.1 Entorno del proyecto | 1 |
| 1.2 Descripción del problema..... | 1 |
| 1.3 Síntesis del problema | 2 |
| 1.4 Objetivos | 3 |
| Objetivo General..... | 3 |
| Objetivos Específicos | 3 |
| 2. Marco Teórico | 3 |
| 2.1. Sistemas de pesaje..... | 3 |
| 2.2. Protocolos de comunicación | 6 |
| Serial ASCII | 8 |
| Modbus TCP/IP | 9 |
| 2.3 Servicios a la web y bases de datos | 11 |
| 3. Metodología | 13 |
| 4. Propuesta de diseño | 31 |
| 4.1 Estructura mecánica de pesaje..... | 31 |
| 4.2 Sistema eléctrico de pesaje | 34 |
| Selección de componentes..... | 35 |
| Detalles de implementación..... | 36 |
| 4.3. Implementación del software de automatización para la adquisición de datos de peso..... | 40 |
| 4.4. Implementación de software de enlace a la web | 48 |
| 5. Resultados y Análisis | 51 |
| 5.1. Validación | 51 |
| Plataforma de pesaje..... | 51 |
| Sistema de pesaje | 54 |
| Automatización de adquisición de pesos..... | 59 |
| Software de enlace a la web..... | 62 |

| | |
|--|----|
| 5.2. Análisis Económico | 64 |
| Costo de desarrollo..... | 64 |
| Beneficios del proyecto..... | 67 |
| Análisis financiero..... | 68 |
| 6. Conclusiones y recomendaciones..... | 70 |
| 7. Referencias bibliográficas | 71 |
| 8. Anexos | 73 |

INDICE DE FIGURAS

| | |
|--|----|
| Figura 1. Diagrama de esfuerzo- deformación para material dúctil (acero) [1]..... | 4 |
| Figura 2. Celda de Carga Extensométrica [3] | 6 |
| Figura 3. Modelo de aplicación OSI [5] | 7 |
| Figura 4. Topologías de red de dispositivos [6]. | 9 |
| Figura 5. Encapsulado Modbus sobre TCP [7]..... | 10 |
| Figura 6. Descripción gráfica de un API REST [9]. | 12 |
| Figura 7. Obtención de subproblemas | 17 |
| Figura 8. Ensamble de la Celda de carga | 31 |
| Figura 9. Estructura de la plataforma | 32 |
| Figura 10. Iteración de estructura de la plataforma | 32 |
| Figura 11. Análisis de diagrama de momento en vigas (Software: MDSolids) | 34 |
| Figura 12. Menú de calibración en el transmisor de peso [13]..... | 37 |
| Figura 13. Entradas y salidas del transmisor de peso [13]..... | 38 |
| Figura 14. Entradas y salidas de la celda de carga de 6 hilos [13] | 38 |
| Figura 15. Menú de configuración de comunicación en el transmisor de peso [13] | 39 |
| Figura 16. Conexión Halfdúplex mediante puerto serial RS-485 [14]..... | 40 |
| Figura 17. Puerto serial Halfduplex del transmisor de peso [13] | 40 |
| Figura 18. Datos de salida de transmisión [13] | 40 |
| Figura 19. Programación en escalera para generación de señal biestable | 42 |
| Figura 20. Rutina de lectura del puerto RS485 del PLC..... | 42 |
| Figura 21. Procesado de lectura ASCII | 44 |
| Figura 22. Procesado de lectura ASCII | 44 |
| Figura 23. Procesado de lectura ASCII | 44 |
| Figura 24. Procesado de lectura ASCII | 45 |
| Figura 25. Configuración de IP del PLC | 46 |
| Figura 26. Configuración de puerto de comunicación del PLC | 47 |
| Figura 27. Tabla de direcciones Modbus | 47 |
| Figura 28. Salidas y entradas del Servidor Modbus dentro del software del PLC | 48 |
| Figura 29. Diagrama de trabajo del software de enlace a la web..... | 49 |

| | |
|---|----|
| Figura 30. Base de datos | 50 |
| Figura 31. Parametros de simulación y gráfica escala de Factor de seguridad (Software: SolidWorks)..... | 53 |
| Figura 32. Estructura de simulación (Software: SolidWorks) | 53 |
| Figura 33. Características de acero ANSI 1020 (Software: SolidWorks)..... | 54 |
| Figura 34. Estudio de deflexión (Software: SolidWorks) | 54 |
| Figura 35. Sistema eléctrico de pesaje | 55 |
| Figura 36. Gráfica de medición de tiempo de respuesta (Software: Excel)..... | 60 |
| Figura 37. Interfaz de usuario | 63 |
| Figura 38. Prueba de tiempo de ejecución..... | 64 |

INDICE DE TABLAS

| | |
|--|----|
| Tabla 1. Códigos de función para mensajes Modbus TCP/IP | 11 |
| Tabla 2. Interpretación de necesidades | 13 |
| Tabla 3. Establecimiento de prioridad | 15 |
| Tabla 4. Búsqueda externa e interna de soluciones. | 17 |
| Tabla 5. Características del tipo de controlador | 19 |
| Tabla 6. Características del tipo de sensor de peso | 20 |
| Tabla 7. Características de lenguaje de programación | 21 |
| Tabla 8. Características de protocolos de comunicación | 22 |
| Tabla 9. Concepto 1 | 22 |
| Tabla 10. Concepto 2 | 23 |
| Tabla 11. Concepto 3 | 23 |
| Tabla 12. Concepto 4 | 24 |
| Tabla 13. Métricas Objetivo..... | 25 |
| Tabla 14. Calificación y filtrado de conceptos | 26 |
| Tabla 15. Selección de concepto | 27 |
| Tabla 16. Resumen de pruebas de validación | 29 |
| Tabla 17. Prueba de validación 1 | 29 |
| Tabla 18. Prueba de validación 2 | 30 |
| Tabla 19. Prueba de validación 3 | 30 |
| Tabla 20. Prueba de validación 4 | 30 |
| Tabla 21. Componentes principales | 36 |
| Tabla 22. Sensibilidades específicas de las celdas de carga..... | 36 |
| Tabla 23. Mediciones de pesos..... | 56 |
| Tabla 24. Mediciones de pesos..... | 57 |
| Tabla 25. Tiempos de respuesta | 60 |
| Tabla 26. Tiempos de respuesta en segundos | 61 |
| Tabla 27. Recursos necesarios | 65 |
| Tabla 28. Componentes eléctricos utilizados | 65 |
| Tabla 29. Materiales utilizados | 66 |

| | |
|--|----|
| Tabla 30. Mano de obra requerida | 66 |
| Tabla 31. Flujo de caja | 69 |

LISTA DE ABREVIACIONES

| | |
|----------------------------|--|
| FOS | Factor de seguridad |
| OSI | Interconexión de Sistemas Abiertos |
| PLC | Controlador Lógico Programable |
| TCP | Protocolo de Control de Transmisión |
| API | Interfase de Programación de Aplicaciones |
| HTTP | Protocolo de Transferencia de Hipertexto |
| SQL | Lenguaje de Consulta Estructurada |
| SGBD | Sistema de Gestión de Base de Datos |
| SD | Sistema Diseñado |
| CAD | Diseño Asistido por Computadora |
| ORM | Mapeo Objeto-Relación |
| σ | Desviación de la muestra |
| S | Desviación Estándar |
| Z | Estadístico de distribución normal |
| BD | Base de Datos |
| JSON | Notación de Objeto Javascript |
| TIR | Tasa Interna de Retorno |
| VAN | Valor Actual Neto |
| IFC | Inversión Fija en Componentes |
| | Inversión Fija en Materiales |

| | |
|--------------|--|
| IFM | Mano de Obra |
| MO | Mano de Obra Fija |
| MOF | Mano de Obra No Fija |
| MONF | Código Estándar Americano para Intercambio de Información |
| ASCII | |

1. Introducción

1.1 Entorno del proyecto

La entidad para la que se desarrolló el presente proyecto es una empresa del sector privado llamada Creativa RAAL Industrial S.A. Esta compañía actualmente desarrolla proyectos electromecánicos y mecatrónicos a la medida para compañías con procesos de producción mayoritariamente dentro del sector de dispositivos médicos.

Esta empresa realiza diseño, mejora y replicación de máquinas automatizadas para procesos productivos y necesidades puntuales del sector industrial, cuenta con un departamento de mecanizado, departamento de diseño mecánico y departamento de control eléctrico. Dicha empresa inició como una empresa de servicios electromecánicos y a lo largo del tiempo se fue especializando en alta tecnología como la automatización de maquinaria de líneas de producción de todo tipo de compañía como Abbott Vascular, Boston Scientific. La alta tecnología es un factor de desarrollo muy importante dentro de esta empresa, se desarrollan soluciones con alta tecnología como lo son sistemas de visión, brazos robóticos y controladores de alto desempeño. Así como contar con profesionales especializados en los procesos de diseño mecánico en la ingeniería, en control automático y sistemas de visión, así como profesionales de la metalmecánica que trabajan los subsistemas mecánicos y estructurales de los proyectos casi en su totalidad dentro de la empresa.

Dicha empresa actualmente tiene la visión de iniciar un sistema de producción en serie, es decir, no solo desarrollar soluciones puntuales a las compañías que contratan el servicio, sino que buscar un producto el cual se pueda desarrollar con la visión de que algún sector productivo lo adquiera en más grandes cantidades.

1.2 Descripción del problema

El problema que se solucionó tiene lugar en el sector comercial, específicamente a nivel de comercios y supermercados, con respecto a la venta de productos. Se detectó la necesidad de un sistema con versátil disposición para la verificación de peso de

productos vendidos en supermercados, con el objetivo de implementar la primera etapa de un sistema automático de cobro de productos. Algunas causas de esta necesidad son por ejemplo la importancia de disminuir pérdidas por incongruencia en la cantidad de ventas, también para evitar errores de cobro de productos y acelerar el método de cobro de los productos.

Por otro lado, el presente proyecto tiene la característica de contribuir en la búsqueda de un producto el cual sea posible colocar en el mercado y desarrollar en serie por parte de la empresa. Desarrollar este proyecto a manera de prototipo es viable económicamente para la empresa, ya que un prototipo de no muy alto costo pero que demuestre que se puede desarrollar un sistema que automatice el cobro de productos en supermercados, funciona como herramienta de marketing y exposición del producto a futuros clientes y compradores del sistema diseñado.

Se considera que desarrollar un mecanismo de verificación de peso de productos, viene a contribuir en el resguardo de la viabilidad de negocio, con el hecho de disminuir pérdidas por error humano en el cobro de productos u otros factores, facilitar y evitar problemas en el inventariado de productos vendidos o cierres contables, así como agilizar el método de cobro de productos. Por otro lado, con la característica de versátil disposición se obtiene una característica que diferencia el producto de otros productos que se puedan utilizar para el mismo fin, por ejemplo sistemas ya existentes para cobro de productos colocándolos en una báscula, en este caso se busca un pesaje sin necesidad de retirar los productos del carro o canasta de compras.

1.3 Síntesis del problema

Actualmente no todos los supermercados a nivel nacional cuentan con un sistema de cobro ágil y automatizado, el problema inicialmente surge con la implementación de un sistema de verificación del peso automatizado de los productos que los clientes adquirieren del supermercado, con el fin de desarrollar la primera etapa de un sistema de cobro automatizado.

1.4 Objetivos

Objetivo General

Diseñar un dispositivo encargado de la verificación del peso de los productos adquiridos por clientes en los comercios, para el desarrollo de un sistema de cobro de productos automatizado.

Objetivos Específicos

- Diseñar una plataforma para el sistema de pesaje, con un factor de seguridad de al menos 1.5
- Construir un prototipo a nivel eléctrico de un sistema de pesaje de productos, con una capacidad máxima de medición de 200 Kilogramos y una repetibilidad no mayor a 50 gramos.
- Automatizar la toma de mediciones de pesos y el procesado/envío de datos adquiridos con un tiempo de respuesta no mayor a 1.7 segundos.
- Programar software para la adquisición de datos de peso en tiempo real y para el procesado y envío de los datos hacia un servicio web.
- Validar el sistema diseñado mediante pruebas funcionales y de concepto.
- Desarrollar un análisis económico derivado del beneficio del proyecto.

2. Marco Teórico

2.1. Sistemas de pesaje

Existen gran cantidad de sistemas de pesaje, desde sistemas de pesaje mecánicos a sistemas electrónicos de pesaje, con distintos mecanismos para medir el peso de un objeto. En este trabajo al mencionar sistema de pesaje, se hace referencia a un sistema de pesaje tipo báscula, se tiende a llamar báscula a sistemas de pesaje con alta capacidad de medición como los usados para pesar ganado etc.

Todo sistema de pesaje cuenta con una estructura mecánica que soporta la carga que se desea pesar, cuando se trata de sistemas de pesaje para cargas altas, el análisis de la estructura y del material es un aspecto importante para garantizar una exitosa medición. Al elegir un material se debe tomar en cuenta sus propiedades mecánicas, en el diagrama de esfuerzo-deformación para materiales dúctiles de la figura 1 se muestran estos valores de interés.

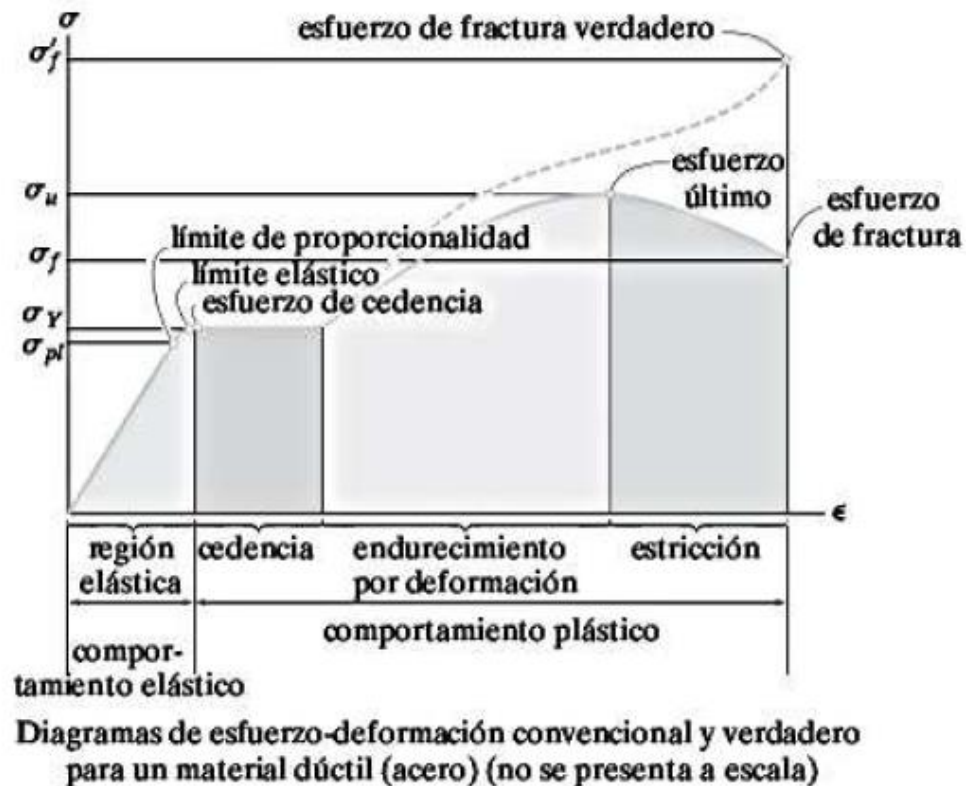


Figura 1. Diagrama de esfuerzo- deformación para material dúctil (acero) [1]

Si se requiere diseñar una estructura robusta se debe trabajar en la región elástica que se muestra en la figura 2, incluso en el inicio de esta zona por la aplicación requerida. Esta zona se caracteriza por estar antes del esfuerzo de fluencia o cedencia del material (σ_Y) esto garantiza que el material no se va a deformar de manera permanente ni fracturar. En el diseño se habla de factor de seguridad (FoS) esto es aplicar un cociente entre el esfuerzo que se desea limitar σ_Y entre el esfuerzo sometido y tiene que ser mayor a 1 para estar en un rango seguro.

Como lo menciona P. Otero [2] en su trabajo de desarrollo e investigación, un sistema de pesaje electrónico o digital es un sofisticado dispositivo hardware y software con la capacidad de procesamiento de los datos durante la operación de pesaje, en donde se obtiene una señal electrónica que es tratada y transmitida a otro sistema para ser procesada o visualizada.

Un tipo de sistema de pesaje que es de importancia definir para este trabajo, es la Báscula de Sobresuelo, como su nombre lo describe, son básculas que tienen la plataforma de pesaje en el suelo, ya que no resulta práctico para su aplicación llevar los objetos hacia una mesa de pesaje. La complejidad a nivel de hardware es mayor en comparación a básculas pequeñas, ya que generalmente se utiliza para pesar objetos con dimensiones grandes, lo cual requiere una plataforma grande de pesaje y con esto es necesario una mayor cantidad de sensores de peso colocados en las cuatro esquinas de la plataforma, para ejemplificar un posible diseño.

Con respecto a los sensores de peso, dentro de la gran variedad de sensores de peso que existen en el mercado, para este trabajo es de interés definir los sensores de peso llamados Celdas de Carga Extensiométricas y Celdas de Carga Hidráulicas; como posteriormente se describirán, la naturaleza de funcionamiento de estos dos tipos de sensores son distintas.

Las Celdas de Carga según J. Estrada [2] de manera general son transductores que convierten la fuerza que se le aplica en una señal de salida medible, internamente se distinguen del fundamento físico de medición (Neumático, Hidráulico, Eléctrico). Las Celdas de Carga que generan una tensión de salida son las Celdas de Carga Extensiométricas, ya que según Prometec [3] cuentan con Galgas Extensiométricas que son básicamente resistencias eléctricas que se adhieren a un material formando un circuito Puente de Wheatstone, y de acuerdo a su deformación por flexión, tensión, compresión o cizalladura generan una tensión eléctrica medible proporcional a la fuerza aplicada, tal y como se muestra en la Figura 2. Por otro lado, las Celdas de Carga Hidráulicas miden un cambio de presión en un fluido interno de llenado, la cual se transfiere a un pistón que a su vez comprime un diafragma.

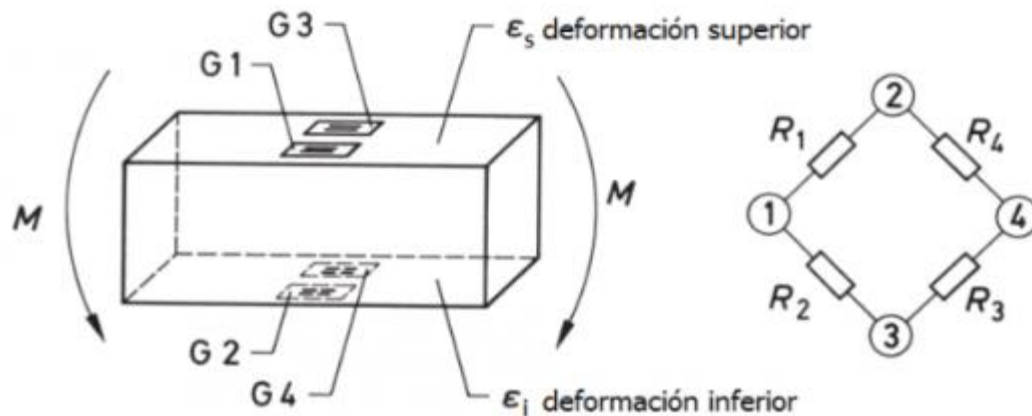


Figura 2. Celda de Carga Extensométrica [3]

Con respecto a la parte electrónica de los sistemas de pesaje, estos cuentan con un transmisor de peso que según ZemiC Europe [4], se encargan de interpretar las señales provenientes de los sensores de peso (2-3 mV/V) y mediante un convertidor digital obtener una medición de peso en una salida de 0-20mA, 0-10V o enviar la información mediante comunicación Serial, Modbus o Ethernet. Los transmisores de peso generalmente entregan la alimentación eléctrica hacia las celdas de carga que soporte.

2.2. Protocolos de comunicación

Los protocolos de comunicación son un aspecto bastante importante para este trabajo y es de importancia conocer que existen gran cantidad de protocolos de comunicación industriales y que son muy utilizados en distintas aplicaciones. La idea de implementar un protocolo de comunicación es lograr llevar la información de un punto a otro con un formato de datos específico. Como lo menciona F. & L. J.Cedillo [5], los protocolos de comunicación son un conjunto de reglas que gobiernan el intercambio de datos dentro de una red, para que exista comunicación entre dispositivos, generalmente

se tiene un dispositivo maestro y un dispositivo esclavo, y estos deben estar configurados adecuadamente para el mismo protocolo de comunicación.

Existe un lineamiento estructural para el intercambio de información entre computadoras y terminales llamado Interconexión de Sistemas Abiertos o bien el modelo OSI (Open System Interconnection), este lineamiento está compuesto por una arquitectura de comunicación de 7 capas, como se muestra en la figura 3.

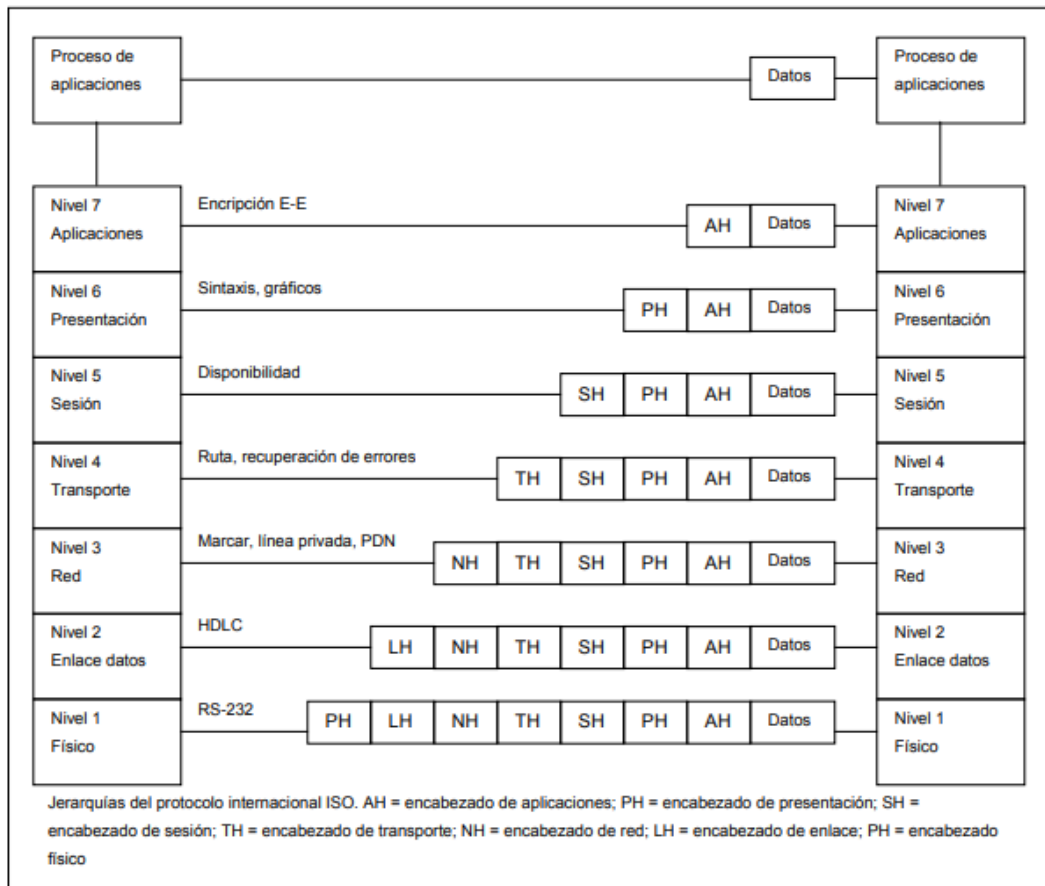


Figura 3. Modelo de aplicación OSI [5]

Los niveles 4, 5, 6 y 7 son las capas que permiten la comunicación con los dispositivos en la red de forma directa y las tres capas inferiores representan la mecánica real de la trasmisión de los datos (a nivel de bit) de una máquina a otra.

La capa física es la de más bajo nivel y establece las consideraciones físicas para la transmisión de los datos, como lo es un medio de transmisión RS-232, RS-485, ASCII, RJ-45. La capa de enlace de datos proporciona la envoltura final de la trama de información y la capa de red determina la configuración de red adecuada según el protocolo deseado. Las capas superiores son capas a nivel de software que se encargan de garantizar la calidad de conexión y la presentación de los datos.

Los protocolos que interesan definir para este trabajo son los protocolos de comunicación Serial ASCII y el protocolo de comunicación Modbus TCP/IP.

Serial ASCII

La comunicación serial es de las primeras tecnologías implementadas para la transmisión de información en telecomunicaciones, es un protocolo de comunicación en que se transmite y se envía información en bits, uno a la vez en donde se puede una comunicación de solo envío o recepción (Simplex), de recepción y transmisión pero uno a la vez (Half duplex) y en ambos sentidos a la vez (Full duplex). Se han implementado gran cantidad de normas técnicas que especifican las características eléctricas de la transmisión de la información, tal y como se explicó anteriormente con respecto a la primera capa del modelo OSI. Un ejemplo de interés es el protocolo RS-485 de comunicación serial.

Según Bob Perrin [6] en su artículo relacionado al RS-485, esta norma técnica es aceptada desde el año 1983 y es una comunicación muy utilizada por la facilidad de conexión, al ser simplemente dos cables para la transmisión halfduplex y por admitir distintas topologías de conexión con otros dispositivos como se muestra en la figura 4.

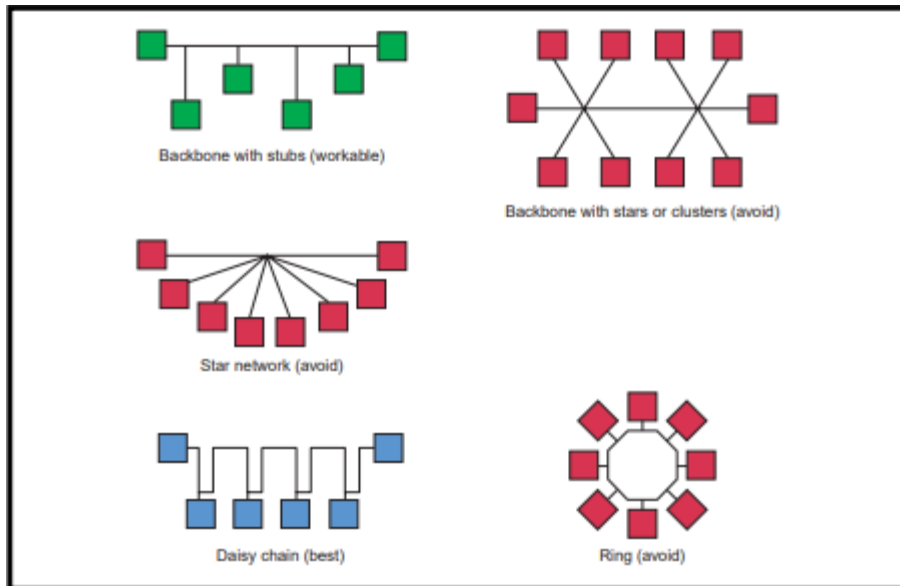


Figura 4. Topologías de red de dispositivos [6].

La topología más confiable según Bob Perrin [6] es la Daisy Chain o topología en cadena, sin embargo es importante tener en cuenta que cuando se tiene solo un dispositivo periférico, una conexión Punto a Punto es la conexión más confiable, este tipo de topología es simplemente la conexión directa entre dos dispositivos.

Con respecto al formato de los bits transmitidos, en este caso se presenta en código ASCII (American Standard Code for Information Interchange) el cual es un código que permite representar 128 caracteres con 7 bits, aunque generalmente se extiende a 8 bits para agregar un bit de paridad.

Modbus TCP/IP

Según A. Ruíz, A. Barandicar, F. Guerrero [7], Modbus/TCP es un protocolo de comunicación diseñado en su mayoría para la conexión, supervisión y control de equipos industriales de automatización como Controladores Lógicos Programables (PLCs), computadores, drivers para motores y tipos de dispositivos físicos de entrada/salida. El termino TCP/IP, primero hace referencia a que TCP es el entorno Intranet o Internet utilizado, es decir, las conexiones a nivel físico están soportadas o

encapsuladas en una red de internet ya desarrollada, en este caso utilizando el puerto 502 por default de esta red. El término IP hace referencia a que este protocolo se comunica por medio de direcciones IP a través de los puertos de comunicación física Ethernet.

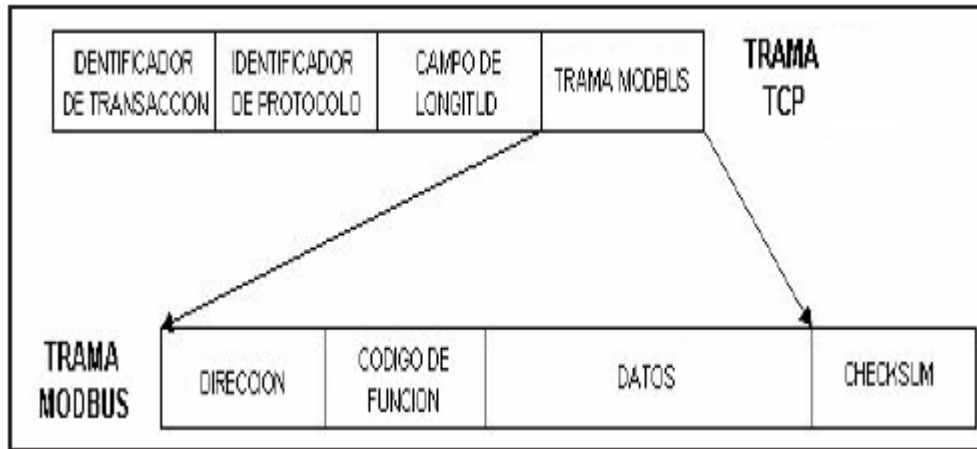


Figura 5. Encapsulado Modbus sobre TCP [7]

Modbus es un protocolo maestro/esclavo con una comunicación orientada a la conexión, esto quiere decir que antes de iniciar la transferencia de datos se debe iniciar o abrir un canal de conexión, esta conexión se establece en la capa de aplicación del modelo OSI, la cual según Cedillo. J, Rafael.F & Linares.L [5], es la capa que da acceso a lo que ofrece las capas de bajo nivel por medio de software, es decir, es un canal de conexión iniciado por medio de software.

Este protocolo tiene un modelo de datos, este modelo especifica las direcciones para las cuales se lee o se escriben en distintos tipos de variables, esto se muestra en la estructura de un dato Modbus (tabla 1), los tipos de variable se da con el código de función, especificados a continuación, basado en la Guía de Modbus Organization INK [8].

Tabla 1. Códigos de función para mensajes Modbus TCP/IP

| Función | Código de función |
|------------------------------|--------------------------|
| Leer bit entrada discreta | 02 |
| Leer bobina | 01 |
| Escribir bobina | 05 |
| Escribir multiples bobinas | 15 |
| Leer registro de entrada | 04 |
| Leer Holding Registers | 03 |
| Escribir un solo registro | 06 |
| Escribir multiples registros | 16 |

2.3 Servicios a la web y bases de datos

Basado en lo mencionado por M. Gallegos [9], un servicio a la web es un componente software localizado en el internet que proporciona datos y servicios a otras aplicaciones software. La comunicación de un servicio web con una aplicación es a través de protocolos web como HTTP (Hypertext Transfer Protocol) o interfaces de programación de aplicaciones (API). La información se transmite codificada o con estructura de datos específica, como por ejemplo XML, JSON, entre otros.

Un ejemplo de lo mencionado anteriormente es una base de datos en la web, la cual puede almacenar datos y proporcionarlos a través de internet, siendo posible la implementación de un API como intermediario para escribir y leer información de la base de datos. Un API REST (Representational State Transfer) quiere decir que es un API en el que entre dos llamadas no guarda los datos, es decir, se realizan peticiones no

dependientes de las anteriores y por otro lado, definen a las interfaces que hacen uso del protocolo HTTP para obtener o manejar los datos en formatos como XML o JSON [10].

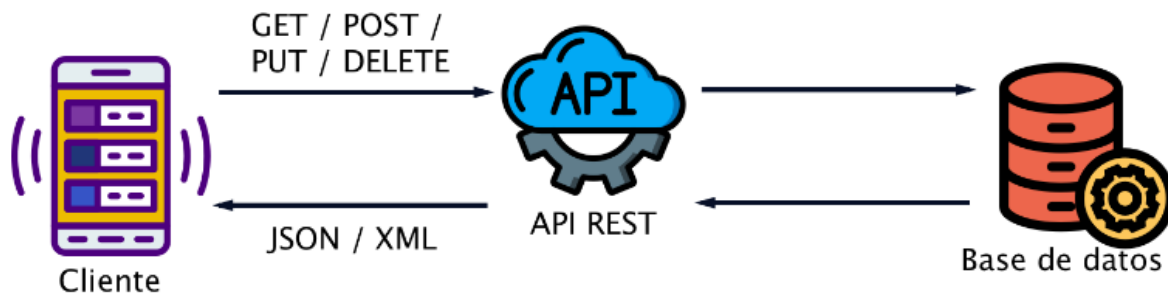


Figura 6. Descripción gráfica de un API REST [9].

La forma en que el API y la Base de datos interactúan con el cliente es a través de funciones llamadas Endpoints (Get, Post, Put , Delete) estas funciones son las encargadas de escribir información a la base de datos (Post) o también leer información de la base de datos (Get) [10] .

Con respecto a las bases de datos, básicamente una base de datos es un conjunto de datos almacenados en memoria externa y que estos están organizados mediante una estructura de datos. Las bases de datos se crean una sola vez y se utilizan al mismo tiempo por distintos usuarios [11]. La gestión de las bases de datos se realiza mediante Aplicaciones o Sistemas de Gestión de Bases de Datos (SGBD), estas aplicaciones permiten a los desarrolladores crear, definir y mantener la base de datos con acceso controlado. Los SGBD permiten definir la base de datos mediante el lenguaje de definición de datos, el cual define la estructura, el tipo y las restricciones de los datos [11].

Un lenguaje de gestión de interés para este proyecto es el SQL (Structured Query Language), el cual es un lenguaje estándar para la gestión de bases de datos relacionales; las bases de datos relacionales están formadas por un conjunto de

relaciones y a estas relaciones en lenguaje SQL se le denominan Tablas. Este lenguaje proporciona comandos como CREATE TABLA, para crear tablas de datos con columnas (Atributos) con su respectivo tipo de dato (Int, String, Real, Entre otros) y luego se tienen las filas, que son las tuplas con los distintos atributos que se pueden consultar, modificar o borrar [11].

3. Metodología

La etapa 1 corresponde a la determinación de necesidades, las cuales el Sistema Diseñado (SD) tiene que contemplar para la generación de conceptos de solución del problema. Las necesidades determinadas se muestran en la tabla 2, las cuales fueron obtenidas del proceso de interpretación de necesidades de los clientes del proyecto, luego en la tabla 3 se muestran las necesidades ordenadas por importancia.

Tabla 2. Interpretación de necesidades

| Tema | Enunciado del cliente | Necesidad Interpretada |
|--------------------------------------|---|---|
| Verificación de productos adquiridos | Para automatizar el cobro se debe pesar los productos o identificar cuales lleva de manera sencilla y rápida | 1. El SD verifica los productos adquiridos de manera rápida. |
| | Se requiere indicadores o señalizaciones visuales de si los productos que lleva son los mismos, para dejarlo pagar o no | 2.El SD indica visualmente al cliente si procede sin problemas el pago o no |

| | | |
|------------------------------------|---|---|
| Adquisición y envío de información | Se requiere que una persona con un software externo valide el pago de los productos | 3.El SD envía la información a un software de validación del pago |
| | Para marketing del producto se requiere mostrar el funcionamiento del sistema diseñado sin tener el software que utiliza el cliente | 4. El SD demuestra la capacidad de enviar los datos a un software, también recibir información para la etapa de indicadores visuales. |
| Prototipado | Que el dispositivo sea fácil de instalar y en distintos lugares | 5.El SD es versátil de instalar |
| | La capacidad de carga tiene que ser acorde a un carro de compras lleno de compras usuales en supermercados o mayor | 6. El SD bajo en concepto de compras en supermercado, tiene alta capacidad de carga |
| | Prototipo desarrollado debe demostrar el funcionamiento completo, por ejemplo si necesita pasar los productos, que se implemente el sistema de pesaje | 11. El SD cuenta con error de medición adecuado |
| | Costo de componentes eléctricos no mayor a 1500 dólares | 10. El SD no sobrepasa los 1500 en componentes eléctricos aproximadamente |

| | | |
|--------------------|---|--|
| Mantenimiento | Que el mantenimiento del sistema sea económico y no muy frecuente | 7. El SD es robusto y de mantenimiento económico. |
| Consumo energético | Dispositivo de bajo consumo energético | 8. El SD es de bajo consumo energético 9. El SD funciona eléctricamente a tensiones de control de 24V |

Tabla 3. Establecimiento de prioridad

| Nivel. descripción | Necesidades |
|---|---|
| 5. La función es de importancia crítica. No consideraría un producto sin ella | 3. El SD envía la información a un software de validación del pago 11. El SD cuenta con error de medición adecuado |
| 4. La función es altamente deseable, pero consideraría un producto sin ella | 1. El SD verifica los productos adquiridos de manera rápida. 2. El SD indica visualmente al cliente si procede sin problemas el pago o no 6. El SD bajo en concepto de compras el supermercado, tiene alta capacidad de carga 7. El SD es robusto y de mantenimiento económico |
| 3. Sería bueno tener esa característica, pero no es necesaria | 4. El SD demuestra la capacidad de enviar los datos a un software, también recibir información para la etapa de indicadores visuales |

| | |
|---|--|
| | <p>5. El SD es versátil de instalar</p> <p>8. El SD es de bajo consumo energético</p> <p>10. El SD no sobrepasa los 1500 en componentes eléctricos aproximadamente</p> |
| 2. La función no es importante, pero no me importaría tenerla | 9. El SD funciona eléctricamente a tensiones de control de 24V |
| 1. La función no es deseada | |

La etapa 2 se muestra en el diagrama de flujo de la figura 7, esta etapa corresponde a la división en subproblemas que constituyen el SD final. Esta etapa fue importante porque se divide el problema en problemas más pequeños, generalmente se trata de dividir en etapas de trabajo y en componentes de cada etapa. Para la etapa de Pesaje, se tienen subproblemas como la estructura mecánica que soporta las cargas por pesar y el método de pesaje. Para la Automatización de medición se busca un tipo de controlador y un respectivo método de comunicación con el sistema mecánico de mediciones y para el Software de adquisición, interesa conocer el lenguaje de programación y el método de comunicación con el controlador de la etapa anterior, para así desarrollar una báscula de pesaje automatizada.

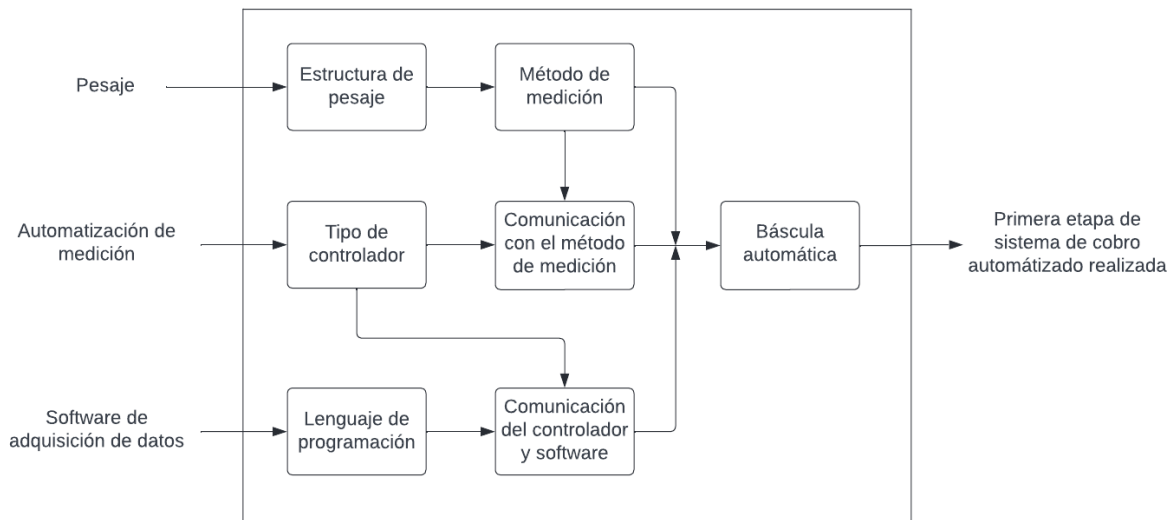


Figura 7. Obtención de subproblemas

En la etapa 3 se realizó una búsqueda de información externa de candidatos para la solución o bien una elección de candidato basada en conocimiento o experiencia propia. En la tabla 4 se presentan las soluciones individuales que interesa analizar en conjunto posteriormente.

Tabla 4. Búsqueda externa e interna de soluciones.

| Generación de conceptos para cada sub problema | |
|---|---|
| Estructura de pesaje | Estructura de acero inoxidable |
| | Estructura de acero estructural de bajo carbono |
| | Estructura de aluminio |
| | Celdas de carga extensométricas |

| | |
|---------------------|-----------------------------|
| Método de medición | Celdas de carga hidráulicas |
| Tipo de controlador | PLC |
| | Arduino Mega |

| | |
|--|-----------------|
| Comunicación del transductor de medición | Modbus TCP/IP |
| | ASCII serial |
| | Ethernet TCP/IP |
| Comunicación del controlador y software | Modbus TCP/IP |
| | Ethernet TCP/IP |
| Lenguaje de programación software | Python |
| | C#, C |

La etapa 4 corresponde a la combinación y calificación de candidatos, es decir para cada subproblema se elige un candidato y se forman combinaciones de soluciones al problema principal. Los cuatro candidatos que se obtuvieron se muestran en la tabla 9, 10,11 y 12.

Para la calificación de conceptos primero se determinan unas métricas adecuadas para cada subsistema y se busca información de referencia para tener el criterio necesario para calificar cada concepto, en las casillas que aparece (-) significa que no se encontró información.

Tabla 5. Características del tipo de controlador

| Número Métrica | Número Necesidad | Métrica | Imp | Unidad | PLC MICRO 820 | Arduino Mega |
|-----------------------|-------------------------|--|------------|---------------|----------------------|---------------------|
| 1 | 7 | Frecuencia de mantenimiento | 3 | Mes | - | - |
| 2 | 7 | Vida útil de componentes principales | 5 | Años | 15-20 | 3-5 |
| 3 | 3 | Protocolo de comunicación del sistema de pesaje, con estándares industriales | 5 | Binario | SI | NO |
| 4 | 8 | Consumo energético | 3 | Watts | 6-10 | 4-6 |
| 5 | 6 | Capacidad de carga y medición | 4 | Kg | N/A | N/A |
| 6 | 6 | Factor de seguridad del esfuerzo permisible de la plataforma | 4 | N/A | N/A | N/A |
| 7 | 10 | Precio | 3 | Dólares | 400 | 60 |
| 8 | 3,1 | Velocidad de transmisión de datos | 4 | Bps | 38400 max | 115200 max |
| 9 | 3,1 | Tiempo de respuesta del sensor | 4 | ms | N/A | N/A |

| | | | | | | |
|----|----|-------------------|---|----|-----|-----|
| 10 | 11 | Error de medición | 5 | Kg | N/A | N/A |
|----|----|-------------------|---|----|-----|-----|

Tabla 6. Características del tipo de sensor de peso

| Número Métrica | Número Necesidad | Métrica | Imp | Unidad | Celdas de carga extensométricas | Celda de carga hidráulicos |
|-----------------------|-------------------------|---|------------|---------------|--|-----------------------------------|
| 1 | 7 | Frecuencia de mantenimiento | 3 | Mes | - | Mayor |
| 2 | 7 | Vida útil de componentes principales | 5 | Años | - | 5 |
| 3 | 3 | Protocolo de comunicación del sistema de pesaje, con estándares industriales. | 5 | Binario | N/A | N/A |
| 4 | 8 | Consumo energético | 3 | Watts | 0.56 | - |
| 5 | 6 | Capacidad de carga y medición | 4 | Kg | 75<x<300 | 100 <x< 1000 |
| 6 | 6 | Factor de seguridad del esfuerzo permisible de la plataforma | 4 | N/A | N/A | N/A |
| 7 | 10 | Precio | 3 | Dólares | 80-120 | 100-150 |

| | | | | | | |
|---|-----|--------------------------------|---|----|-------|-------|
| 8 | 3,1 | Tiempo de respuesta del sensor | 4 | ms | - | <1 |
| 9 | 11 | Error de medición | 5 | % | ±0.03 | ±0.25 |

Para comparar las características de la etapa de software de computación, es decir, comparar los lenguajes de programación elegibles para la aplicación, se generó una tabla comparativa con información externa. Tomando en cuenta la necesidad 3 y 4, es necesario enviar la información de peso obtenida de la báscula a tiempo real a un software externo o un servicio web, por eso es necesario un lenguaje rápido para transferir la información. De acuerdo a las características, se elige de ante mano un software programado en C# o en su defecto C++ el cual tiene características muy similares.

Tabla 7. Características de lenguaje de programación

| Métrica | Python | C#,C++ |
|---------------------------|----------------------------|------------------|
| Tipo de lenguaje | Interprete línea por línea | Compilado |
| Uso | Uso general | Uso con hardware |
| Velocidad | Lento | Rápido |
| Programación por hilos | Permitido | Permitido |
| Librerías de comunicación | Si | Si |

Para comparar las características de los protocolos de comunicación elegibles para transmitir la información del transmisor de peso hacia el controlador se generó una

tabla comparativa entre los protocolos de comunicación por Serial y por Ethernet. Se debe tomar en cuenta que los datos seriales se deben procesar a nivel de tipos de datos para obtener los datos deseados, sin embargo cualquiera de estos dos protocolos puede desempeñar la tarea deseada.

Tabla 8. Características de protocolos de comunicación

| Métrica | Serial | Modbus TCP |
|--------------------------|------------------|-------------------|
| Modo de transmisión | Bit por bit | Paquetes o tramas |
| Conexión a red LAN | No | Si |
| Velocidad de transmisión | Hasta 115200 bps | Por petición |
| Procesado de datos | Mayor | Menor |

Tabla 9. Concepto 1

| Subproblema | Concepto | Boceto/Representación |
|--|-----------------|------------------------------|
| Estructura de pesaje: acero inoxidable | Sistema A | |
| Método de medición: Celdas de carga extensométricas | | |
| Tipo de controlador: Arduino Mega | | |
| Comunicación del transductor de medición: ASCII Serial | | |

| | | |
|--|--|--|
| Comunicación del controlador y software: Modbus TCP/IP | | |
|--|--|--|

Tabla 10. Concepto 2

| Subproblema | Concepto | Boceto/Representación |
|---|-----------------|------------------------------|
| Estructura de pesaje: acero inoxidable | Sistema B | |
| Método de medición: Celdas de carga hidráulicos | | |
| Tipo de controlador: PLC | | |
| Comunicación del transductor de medición: Ethernet TCP/IP | | |
| Comunicación del controlador y software: Ethernet TCP/IP | | |

Tabla 11. Concepto 3

| Subproblema | Concepto | Boceto/Representación |
|--|-----------------|------------------------------|
| Estructura de pesaje: Estructura de acero estructural de bajo carbono | Sistema C | |
| Método de medición: Celdas de carga extensométricas | | |

| | | |
|--|--|--|
| Tipo de controlador: PLC | | |
| Comunicación del transductor de medición: ASCII Serial | | |
| Comunicación del controlador y software: Modbus TCP/IP | | |

Tabla 12. Concepto 4

| Subproblema | Concepto | Boceto/Representación |
|--|-----------------|------------------------------|
| Estructura de pesaje: Aluminio | Sistema D | |
| Método de medición: Celdas de carga hidráulicas | | |
| Tipo de controlador: Arduino Mega | | |
| Comunicación del transductor de medición: Modbus TCP/IP | | |
| Comunicación del controlador y software: Ethernet TCP/IP | | |

Con la información obtenida anteriormente y además utilizando como referencia una báscula comercial modelo: PFK988-C300 de Mettler Toledo, se determinaron las

características ideales y marginales para el sistema diseñado, tal y como se muestra en la tabla 13.

Tabla 13. Métricas Objetivo

| Número Métrica | Número Necesidad | Métrica | Imp | Unidades | Valor marginal | Valor ideal |
|-----------------------|-------------------------|---|------------|-----------------|-----------------------|--------------------|
| 1 | 7 | Frecuencia de mantenimiento de componentes críticos | 3 | Mes | 6 | 12 |
| 2 | 7 | Vida útil de componentes principales | 5 | Mes | 5-8 | 5-10 |
| 3 | 3 | Protocolo de comunicación del sistema de pesaje, con estándares industriales. | 5 | Binario | Si | Si |
| 4 | 8 | Consumo energético | 3 | Watts | <15 | <10 |
| 5 | 6 | Capacidad de carga y medición | 4 | Kg | <= 200 kg | <= 250Kg |
| 6 | 6 | Factor de seguridad del esfuerzo permisible de la plataforma | 4 | N/A | >1 | >=2 |
| 7 | 10 | Precio de desarrollo | 3 | Dólares | <2000 | <1800 |
| 9 | 11 | Repetibilidad | 5 | g | <=50 | <=50 |

Antes de seleccionar un concepto definitivamente, primero se realiza una comparación con un dispositivo comercial, para ya sea descartar algún concepto desde ahora u observar alguna tendencia al respecto, al compararlo con la báscula comercial; teniendo en cuenta que aún no se conoce con exactitud parámetros como el consumo o métricas que tengan que ver con el error o velocidad de medición de los conceptos propuestos, se observa una tendencia de que dos de los conceptos están por encima de los otros dos.

Tabla 14. Calificación y filtrado de conceptos

| Criterios de selección | Báscula PFK988-C300 | A | B | C | D |
|---|----------------------------|----------|----------|----------|----------|
| Precio de desarrollo | 0 | + | + | + | + |
| Consumo energético | 0 | + | 0 | 0 | + |
| Frecuencia de mantenimiento de componentes críticos | 0 | - | + | + | - |
| Vida útil de componentes principales | 0 | - | 0 | 0 | - |
| Protocolo de comunicación con estándares industriales | 0 | 0 | 0 | 0 | 0 |
| Capacidad de carga y medición | 0 | - | - | - | - |
| Velocidad de transmisión de datos | 0 | 0 | 0 | 0 | 0 |

| | | | | | |
|----------------------------------|---|----|----|----|----|
| Error de medición(Repetibilidad) | 0 | - | - | - | - |
| Tiempo de respuesta | 0 | - | 0 | 0 | - |
| Suma + | 0 | 2 | 2 | 2 | 2 |
| Suma 0 | 9 | 2 | 5 | 5 | 2 |
| Suma - | 0 | 5 | 2 | 2 | 5 |
| Evaluación Neta | 0 | -3 | 0 | 0 | -3 |
| ¿Continuar? | | No | Si | Si | No |

Tabla 15. Selección de concepto

| Criterios de selección | Peso (%) (0% a 25%) | Conceptos | | | | | | | |
|---|------------------------|--------------|----------------------|--------------|----------------------|--------------|----------------------|--------------|----------------------|
| | | A | | B | | C | | D | |
| | | Calificación | Evaluación ponderada | Calificación | Evaluación ponderada | Calificación | Evaluación ponderada | Calificación | Evaluación ponderada |
| Precio de desarrollo | 15% | 5 | 0.75 | 3 | 0.45 | 4 | 0.6 | 5 | 0.75 |
| Consumo energético | 15% | 4 | 0.6 | 3 | 0.45 | 3 | 0.45 | 4 | 0.6 |
| Frecuencia de mantenimiento de componente | 15% | 4 | 0.6 | 5 | 0.75 | 5 | 0.75 | 3 | 0.45 |

| | | | | | | | | | |
|---|-----|---|----------------|---|----------------|---|----------------|---|----------------|
| s críticos | | | | | | | | | |
| Vida útil de componentes principales | 25% | 2 | 0.5 | 5 | 1.25 | 5 | 1.25 | 2 | 0.5 |
| Protocolo de comunicación con estándares industriales | 25% | 3 | 0.75 | 5 | 1.25 | 5 | 1.25 | 3 | 0.75 |
| Capacidad de carga y medición | 20% | 4 | 0.80 | 5 | 1 | 4 | 0.8 | 3 | 0.6 |
| Velocidad de transmisión de datos | 20% | 4 | 0.80 | 4 | 0.8 | 4 | 0.8 | 4 | 0.8 |
| Error de medición | 25% | 4 | 1 | 3 | 0.75 | 4 | 1 | 3 | 0.75 |
| Tiempo de respuesta | 20% | 4 | 0.8 | 3 | 0.6 | 4 | 0.8 | 3 | 0.6 |
| Total puntos Lugar Continuar | | | 6.6 3 No | | 7.3 2 Sí | | 7.7 1 Si | | 5.8 4 No |

La etapa final de la metodología de diseño corresponde a la validación de conceptos, en esta etapa del proceso se busca evaluar el concepto seleccionado, mediante una serie de pruebas estrictamente ligadas a las métricas establecidas en el proyecto, las cuales permiten obtener una percepción de cómo se comporta el sistema diseñado. A continuación se van a describir la serie de pruebas que se van a realizar, cuyos resultados se mostrarán en el capítulo de resultados, siempre y cuando los resultados son los deseados para validar las métricas y los respectivos objetivos, de no

ser así se indican los respectivos cambios del sistema diseñado para la repetición de la prueba.

Tabla 16. Resumen de pruebas de validación

| Número de prueba | Necesidad | Objetivo | Variable de muestra | Criterios | Factores de influencia |
|------------------|--|---|--|---|---|
| 1 | El SD bajo el concepto de compras en supermercado, tiene alta capacidad de carga | Plataforma con factor de seguridad mayor a 1.5 para 200 kg de carga | Seguridad ante fluencia o fallo del material | Factor de seguridad a la fluencia | Carga, Resistencia ultima del material, forma de estructura |
| 2 | El SD cuenta con error de medición adecuado | Repetibilidad de medición menor a 50 g | Detección confiable de productos | Robustez de la plataforma | Alta deflexión, dimensión del perfil del material |
| 3 | El SD verifica los productos adquiridos de manera rápida. | Tiempo de respuesta | Rapidez del sistema | Transmisor de peso y programación efectivos | Software del controlador |
| 4 | El SD envía la información a un software de validación del pago | Tiempo de ejecución del software de envío de datos | Rapidez del sistema | Programación y comunicación efectiva | Recursos computacionales |

Tabla 17. Prueba de validación 1

| | |
|---|---------------------|
| Objetivo: Factor de seguridad de plataforma | Número de prueba: 1 |
| Descripción de la prueba: La comprobación del factor de seguridad para una carga de 200 Kg se realiza mediante simulación con el software SolidWorks. Para esto es necesario utilizar la estructura de la plataforma diseñada y realizar un análisis estático con las herramientas que ofrece el software. | |

Naturaleza de la prueba: Simulación
Software: SolidWorks

Tabla 18. Prueba de validación 2

| | |
|---|---------------------|
| Objetivo: Repetibilidad medición | Número de prueba: 2 |
| Descripción de la prueba: Para esta prueba se realiza un conjunto de mediciones de peso con la báscula diseñada con un solo peso, para conocer que tanto varían las mediciones. El número de mediciones se determina mediante la cantidad mínima de una muestra para garantizar un 90% de confiabilidad o más. | |
| Naturaleza de la prueba: Generación y análisis de un muestra Peso: Peso de productos, manejable y significativo (35 – 40)kg | |

Tabla 19. Prueba de validación 3

| | |
|--|---------------------|
| Objetivo: Tiempo de respuesta | Número de prueba: 3 |
| Descripción de la prueba: Para determinar el tiempo de respuesta que tiene la báscula, considerando la etapa de adquisición de los datos mediante el software del controlador, se utiliza la herramienta TREND del mismo controlador utilizado para leer las mediciones. Esta herramienta es para monitorear en el tiempo una variable de interés en el software del controlador. Se monitorea la variable de peso cuando se colocan objetos para el pesaje, se exportan los datos a Excel y se toma el tiempo de respuesta que tiene el sistema. Se coloca un peso por 5 segundos y se mide cuanto tiempo registra el sistema, debería ser mayor a 5 segundos, al hacer la resta se conoce el tiempo de respuesta. | |

Tabla 20. Prueba de validación 4

| | |
|--|---------------------|
| Objetivo: Tiempo de ejecución del software de envió de datos | Número de prueba: 4 |
| Descripción de la prueba: Mediante relojes que se programan en el mismo software programado, se realiza la medición de tiempo que tarda el software en realizar la tarea deseada de lectura y envió de los datos. | |
| Naturaleza de la prueba: Prueba de funcionamiento y tiempo requerido. Software: C# | |

4. Propuesta de diseño

4.1 Estructura mecánica de pesaje

El proceso de diseño inició con el modelado CAD de la estructura base de la báscula, en donde se busca generar los planos de construcción de la plataforma (Anexo1) y validar el material y la topología de la estructura mediante simulación, buscando un Factor de Seguridad mayor a 1.5 para un peso máximo de 200 kg.

La estructura diseñada tiene el objetivo de poder colocar los sensores de peso en una correcta posición de acuerdo al manual de usuario [12] de los sensores elegidos, para que no existan malas mediciones de peso. La plataforma transfiere la totalidad del peso sobre ella a los sensores, al estar los sensores colocados en las esquinas de la estructura soportando todo el peso por medio de las propias patas de la plataforma.

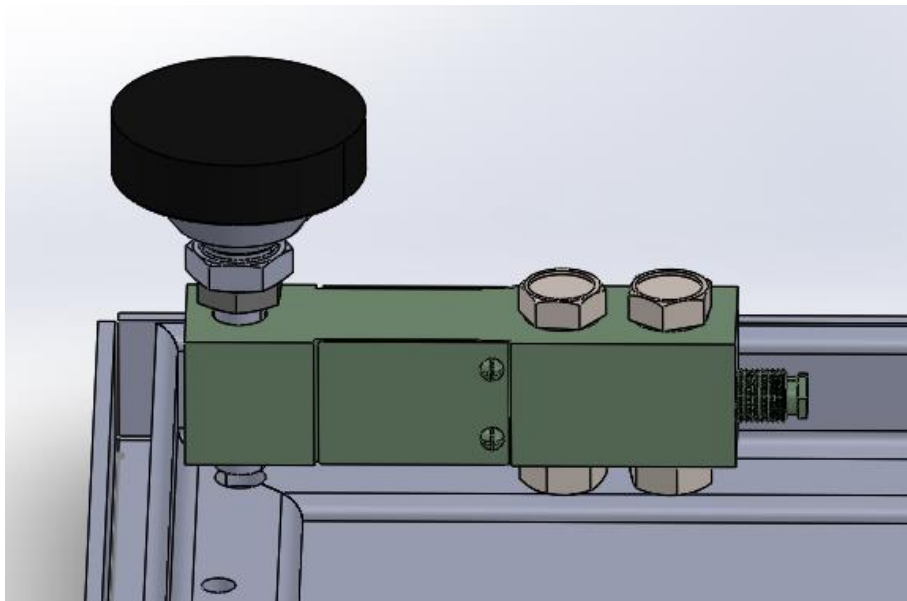


Figura 8. Ensamblaje de la Celda de carga

Inicialmente se diseñó una estructura de la forma que se observa en la figura 10, donde cumplía con el factor de seguridad meta, sin embargo al llevarla a lo físico y probar con cargas y mediciones de los sensores, se observó un fenómeno interesante, ya que el sistema quedaba levemente cargado y flexionado al retirar el peso dispuesto y no

volvía a una posición inicial, lo cual generaba una flexión en los sensores y una medición de entre 100 – 300 gramos en lugar de volver a 0 kg como se desea.



Figura 9. Estructura de la plataforma



Figura 10. Iteración de estructura de la plataforma

Se tomó la decisión de iterar esta componente del proyecto y diseñar una estructura más robusta, con más robusta se hace referencia a utilizar el mismo material

(ANSI 1020) pero con perfiles de mayor dimensión, colocar un perfil que pase de lado a lado, perpendicular a los perfiles que conforman la parte interna de la estructura figura 10, con el objetivo de disminuir la deflexión, ya que como menciona Hibbler en su libro de Resistencia de Materiales, “Con frecuencia, debe limitarse la deflexión de una viga o eje con el fin de proporcionar integridad y estabilidad a una estructura o máquina...” [1]

La ecuación de la deflexión para una viga con sus extremos fijos se muestra en la ecuación 1, donde se puede observar que la deflexión depende de factores del material como el Momento de Inercia (I), el Módulo de Elasticidad (E), Fuerza aplicada (F) y un parámetro geométrico como el largo de la viga (L), el cual si se disminuye, disminuye la deflexión, de aquí se puede intuir que si se coloca una viga perpendicular, se disminuye la deflexión.

$$v = \frac{-FL^3}{48EI} \quad (1)$$

Otra forma de ver el fenómeno de reducción de la deflexión, es con el diagrama de Momento, según la teoría una curva cóncava hacia abajo en el diagrama de momento genera una curva cóncava hacia arriba de la curva elástica o de deflexión de una viga.

En las siguientes imágenes simuladas en el software MDSolids, se muestra que la viga que no tiene una restricción o fuerza en el medio, genera una curva cóncava hacia abajo en el diagrama de momento y la viga que tiene una fuerza en el medio, genera dos curvas cóncavas hacia abajo en el diagrama de momento pero de menor magnitud, esto demuestra la reducción en la deflexión de cada viga de la estructura diseñado, teniendo en cuenta que se idealizó sus extremos fijos, sin embargo realmente los extremos dependen de la deflexión de las vigas que constituyen el marco de la estructura.

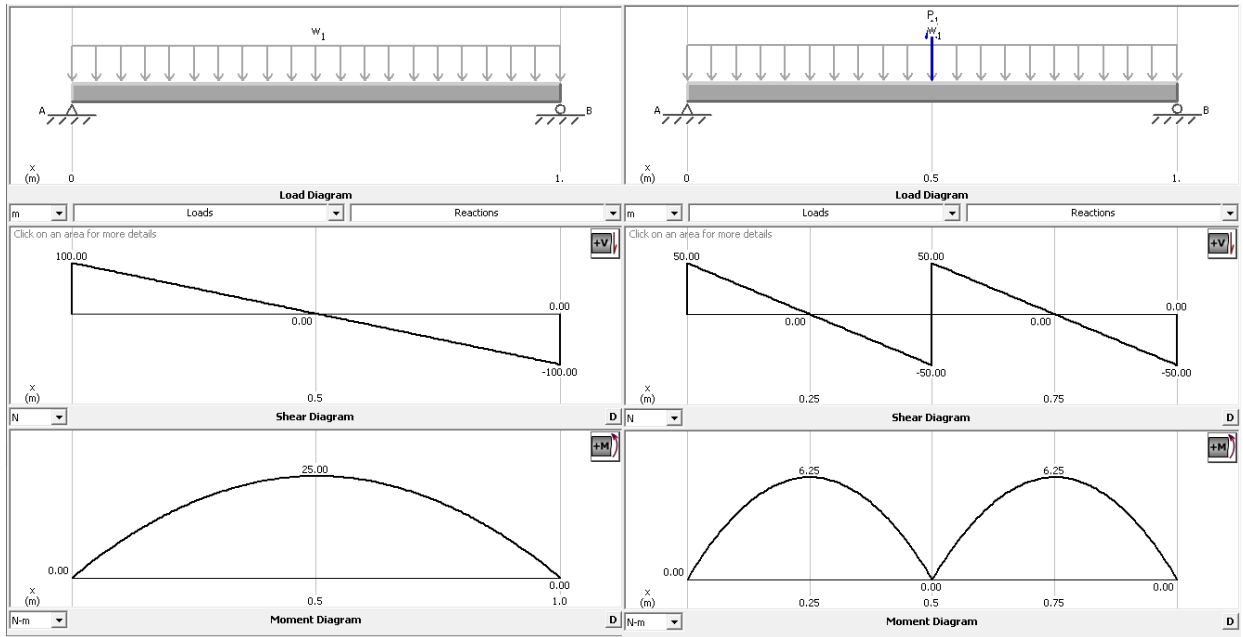


Figura 11. Análisis de diagrama de momento en vigas (Software: MDSolids)

4.2 Sistema eléctrico de pesaje

La segunda etapa de implementación y a su vez el segundo objetivo del proyecto, fue la de construir el sistema de pesaje a nivel eléctrico y de sensores. Esto se desarrolló seguido del diseño de la estructura, ya que era necesario esta componente del proyecto para generar las primeras pruebas de la plataforma, lo cual llevó a iteración como se explicó anteriormente.

El objetivo de esta componente del proyecto es poner a trabajar y calibrar los sensores o celdas de carga extensométricas y transmisor de peso, el transmisor de peso cuenta con su propia visualización del peso medido, por lo que aún no era necesario implementar la componente software de toma de datos y automatización del peso (objetivo específico 3), sin embargo en el circuito eléctrico del sistema de pesaje (Anexo 2) se agregó el hardware necesario para las siguientes etapas del proyecto (PLC entre otros).

Selección de componentes

El criterio de selección del tipo de celda de carga extensométrica se basó en la capacidad de pesaje individual de cada celda, la sensibilidad de medición del instrumento y el tipo de esfuerzo para la medición.

Como se requiere pesar hasta 200 Kg y que según el manual de usuario de la celda no se debe exceder un 75% de la capacidad de carga que se reporta, se considera que 4 celdas de carga de 75kg colocadas en cada esquina de la plataforma, cumplen con los requerimientos de diseño, como se muestra no se supera los 225 Kg permisibles.

$$75 \cdot 4 \cdot 0.75 = 225 \text{ Kg} \quad \text{Carga Permissible} \quad (2)$$

$$200 + 22 = 222 \text{ Kg} \quad \text{Carga máxima de medición} + \text{peso plataforma} \quad (3)$$

Para la aplicación, la cual es pesar productos pequeños o grandes, se considera necesario elegir una celda de carga con una alta sensibilidad de medición, como por ejemplo 2mV/V como la de la celda de carga elegida.

Con respecto al tipo de carga de medición, se eligió una celda de carga de flexión, ya que la celda de carga está colocada en lo alto de la plataforma sin contacto con el piso o alguna base espacial en el diseño mecánico, por ejemplo una celda de carga de compresión no sería la correcta porque sería necesario el contacto completo de la celda con el piso y una base.

El criterio de selección para el transmisor de peso fue el método de comunicación con otro dispositivo y que permita una resolución adecuada para la aplicación, se eligió una transmisión de los datos mediante comunicación Serial ASCII, ya que era necesario transmitir solo el valor de peso continuamente por el puerto serial y el dispositivo logra transmitir el peso medido con hasta 0.0001Kg de resolución.

Con respecto al controlador, en el proceso de generación de concepto se concluyó que se iba a utilizar un PLC por robustez industrial entre otros aspecto, sin embargo con

el modelo de PLC no fue posible generar el proceso de selección de componente, ya que fue un dispositivo que se encontraba adquirido para el proyecto y no fue necesario adquirir otro dispositivo como el Arduino que se planteó en la etapa de filtrado de concepto.

Tabla 21. Componentes principales

| Número de Componente | Nombre de componente | Número de Parte |
|-----------------------------|-----------------------------|------------------------|
| 1 | PLC Micro 820 | 2080LC24QBB |
| 2 | Transmisor de peso | SCT20-NA |
| 3 | Celda de carga | LAUMAS - FTP75 |

Detalles de implementación

La calibración del sistema se realiza de manera teórica en el transmisor de peso, ya que cada celda de carga tiene registrada su propia sensibilidad y esta sensibilidad es un dato confiable que provee el desarrollador del sensor. En la siguiente tabla se muestra la sensibilidad de cada celda de carga utilizada.

Tabla 22. Sensibilidades específicas de las celdas de carga

| Número de sensor | Sensibilidad (mV/V) |
|-------------------------|----------------------------|
| 1 | 2.0043 |
| 2 | 2.0025 |
| 3 | 2.0025 |
| 4 | 2.0017 |

El procedimiento de calibración según el manual de usuario del transmisor de peso [13] se resume en el siguiente diagrama.

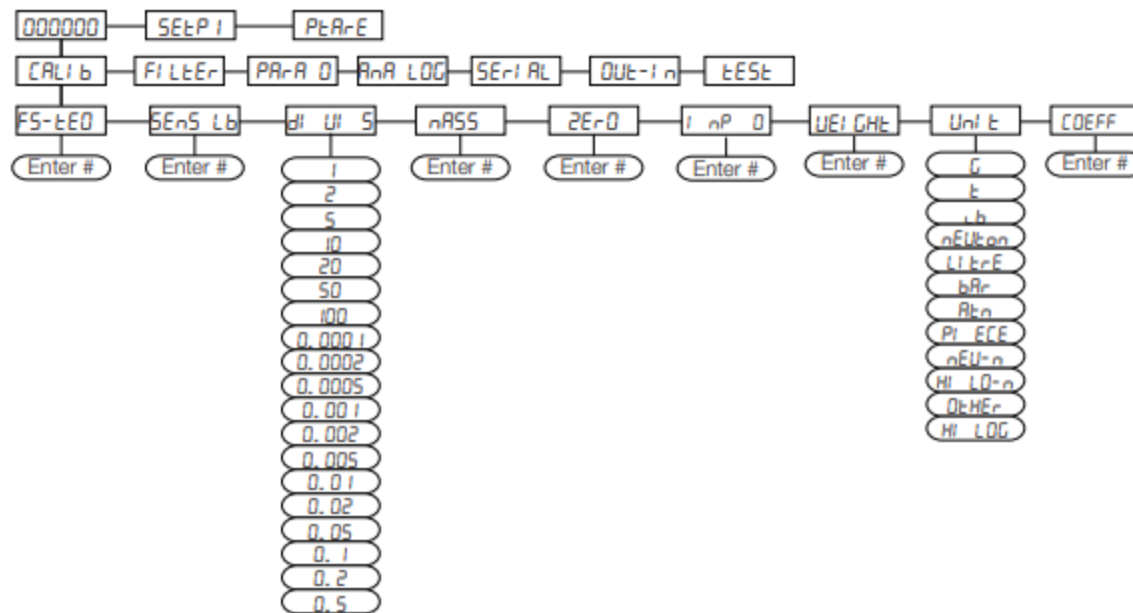


Figura 12. Menú de calibración en el transmisor de peso [13]

Los parámetros más importantes de esta calibración son la especificación del sistema (FS-tE0) en donde se especifica la capacidad máxima del conjunto de celdas empleadas y la sensibilidad (SEnSLb) que es un promedio de la sensibilidad del conjunto de celdas empleadas y la resolución (dl UI S) la cual se configuró en 0.001.

$$75 \cdot 4 = 300 \text{ Kg} \quad (FS - tE0) \quad (4)$$

$$\frac{2.0043 + 2.0014 + 2.0024 + 2.0024}{4} = 2.0025 \quad (SEnSLb) \quad (5)$$

La conexión física entre las celdas de carga y el transmisor de peso se realizó de la siguiente manera, en donde ± Excitación es la tensión (5VCD) de alimentación de las celdas y ± Señal es la salida de las celdas:

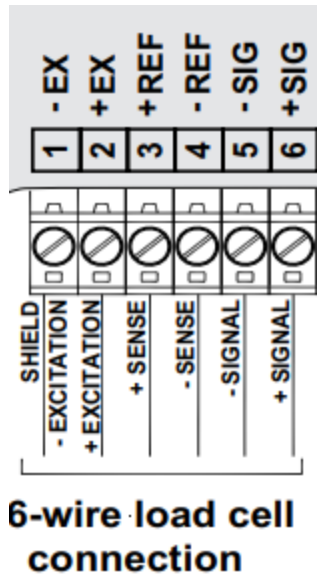


Figura 13. Entradas y salidas del transmisor de peso [13]

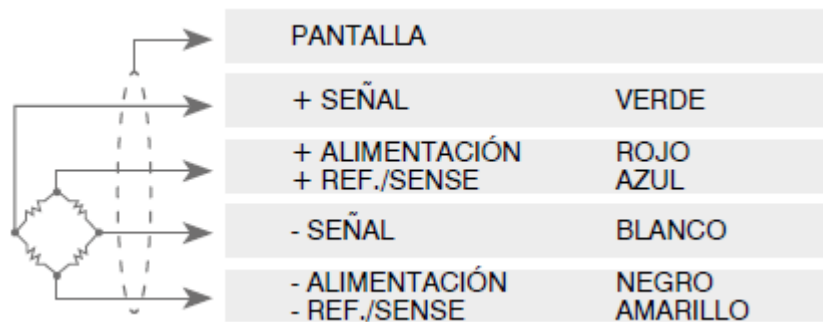


Figura 14. Entradas y salidas de la celda de carga de 6 hilos [13]

La configuración del protocolo de comunicación del transmisor de peso se realizó como se muestra la siguiente figura, en donde se hace uso de la salida para puerto serial RS-485. Los parámetros más importantes son: el modo (Cont In) que es una salida ASCII continua del peso medido con una velocidad de 9600 baudios y un address de 1 para identificar el dispositivo.

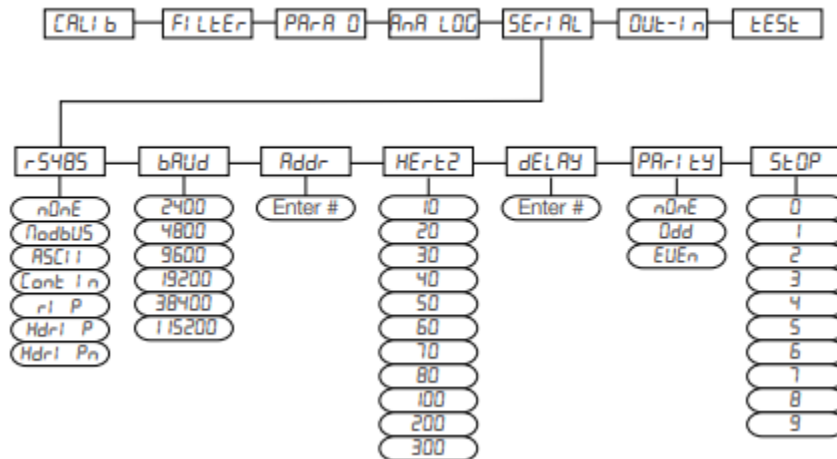


Figura 15. Menú de configuración de comunicación en el transmisor de peso [13]

La conexión eléctrica de la comunicación del transmisor con el controlador se realizó mediante una topología punto a punto entre estos dispositivos y una conexión de tipo Halfduplex para un puerto RS-485 de ambos dispositivos de la red, como se muestra en la siguiente figura, a diferencia que solamente se está utilizando un dispositivo cliente (controlador).

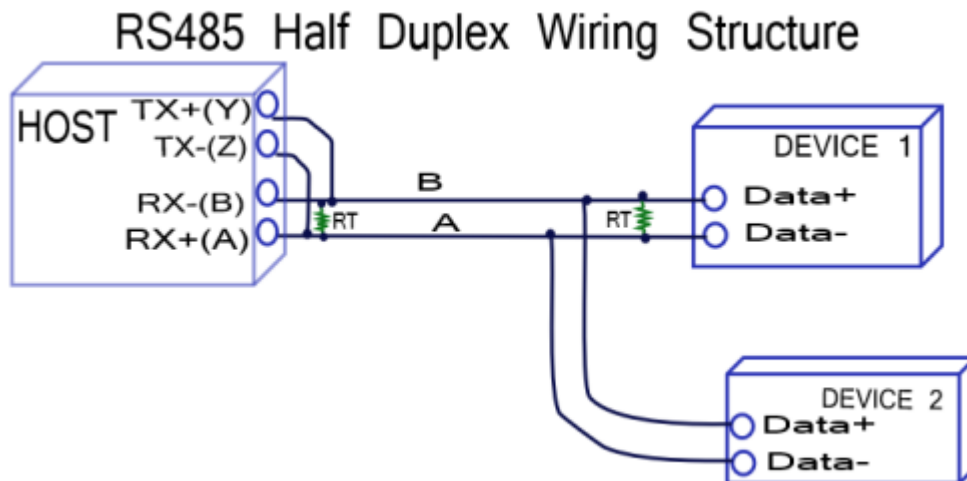


Figura 16. Conexión Halfdúplex mediante puerto serial RS-485 [14]

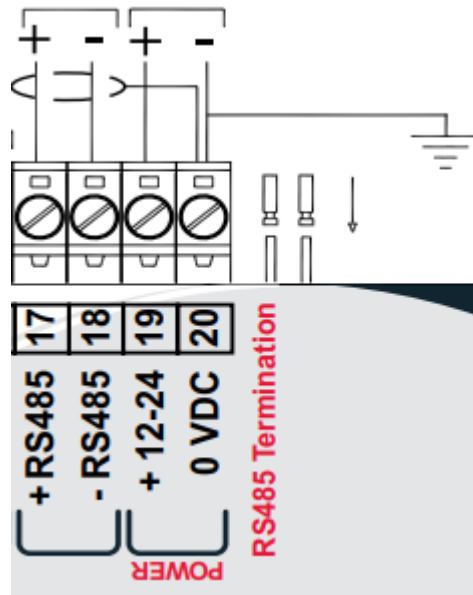


Figura 17. Puerto serial Halfduplex del transmisor de peso [13]

4.3. Implementación del software de automatización para la adquisición de datos de peso.

Para implementar el software de automatización o de software PLC es necesario conocer la estructura de los datos que se van a recibir. El transmisor de peso en modo de transmisión serial continua ordena los datos como se muestra en la siguiente imagen.

| String Character | Definition |
|------------------|---|
| xxxxxx | 6 ASCII characters for gross weight (48 ÷ 57 ASCII) |
| CR | 1 character of carriage return (13 ASCII) |
| LF | 1 character of line feed (10 ASCII) |

Figura 18. Datos de salida de transmisión [13]

Conocer esto es necesario porque se debe indicar por parte del software PLC, cuantos caracteres ASCII se requieren leer para tener la lectura completa del peso, en este caso como se observa en la figura 18, la magnitud del peso se transmite en 6 caracteres ASCII. Luego para lograr procesar esos datos se debe conocer los equivalentes de esos caracteres en decimal, para esto se utilizó la tabla de equivalencias ASCII que se muestra en el anexo 3. Como para esta aplicación solamente nos interesa leer datos numéricos, si se nota en la tabla de equivalencias, los número del 1 al 9 para lograr convertirlos a decimal es simplemente restarle 48 al valor del carácter ASCII. Por ejemplo si se recibe un 50 en ASCII al restarle 48 se obtiene que este carácter en decimal equivale a 2, esto fue parte de la lógica utilizada para el procesado de los datos en el software PLC.

Para la comunicación con el transmisor de peso y leer su salida de peso serial, se utilizaron dos bloques de función definidos por el software de programación Connect Componect Workbench (CCW), en la figura 20 se muestran estos bloques llamados ARD para la lectura del puerto Serial RS485 del PLC y el bloque ACL para la limpieza del puerto serial.

Para lograr una rutina constante de lectura del puerto serial es necesario limpiar el puerto cada vez que se lee con el bloque de lectura ARD, por lo tanto, para lograr esto de manera constante y rápida se implementó una señal biestable o señal de clock que se encargue de habilitar el bloque de lectura cuando se encuentra en estado alto y habilitar el bloque de limpieza y deshabilitar el bloque de lectura cuando la señal se encuentre en bajo. Para esto se utilizó; como se muestra en la figura 19, dos temporizadores de retardo a la conexión, donde la salida del primero (1ms después de iniciado el escaneo del software) habilita el segundo timer, y la salida de este segundo timer deshabilita el primer timer, logrando una señal biestable con 1ms en alto y 1ms en bajo.



Figura 19. Programación en escalera para generación de señal biestable

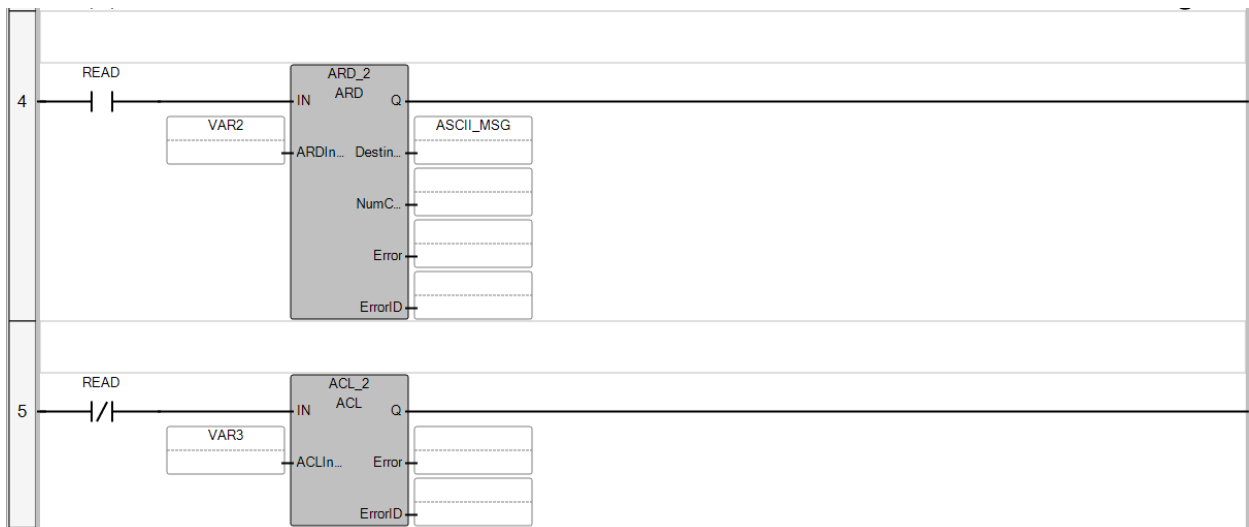


Figura 20. Rutina de lectura del puerto RS485 del PLC

Una vez obtenida una lectura constante en el registro llamado ASCII_MSG [6], el cual contiene las 6 primeras palabras del mensaje transmitido, se prosiguió con el procesamiento del mensaje. Como se muestra en las figuras 21,22 ,23 y 24 se lee cada palabra del registro individualmente, en la palabra más significativa del mensaje (ASCII_MSG [1]) se encuentra el valor de las centenas del peso medido, en la ASCII_MSG [2] se encuentran las decenas y así sucesivamente hasta ASCII_MSG [6] la cual contiene las milésimas del peso medido.

El procesamiento que se realizó fue restarle 48 a cada palabra y luego multiplicar por el factor correspondiente de la palabra en el peso medido, en otras palabras, a las

centenas se multiplica por 100, decenas por 10, unidades por 1, decima dividir por 10, centésima dividir por 100 y milésima dividir por 1000. Con esto realizado se pueden sumar las 6 cantidades obtenidas y obtener un peso total en decimal (figura 23).

En la figura 21 fila 1, se observa que cuando el valor en ASCII de la palabra es 45, se convierte en 48, esto se realizó para evitar errores a la hora de que las celdas de carga proporcionaran un valor negativo de peso como -0.001 Kg, ya que el valor 45 en ASCII es el signo de menos (-) y al restar 45-48 se presentaban problemas. Entonces al convertirlo a 48 la resta da cero y no hay problemas al sumar cero en el resultado final.

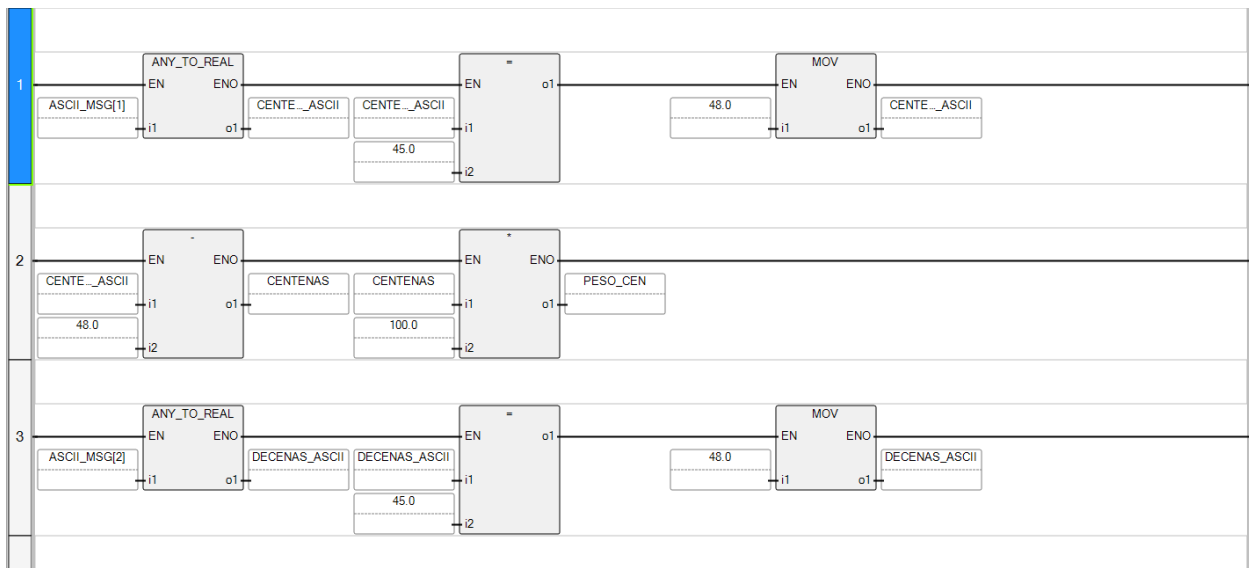


Figura 21. Procesado de lectura ASCII

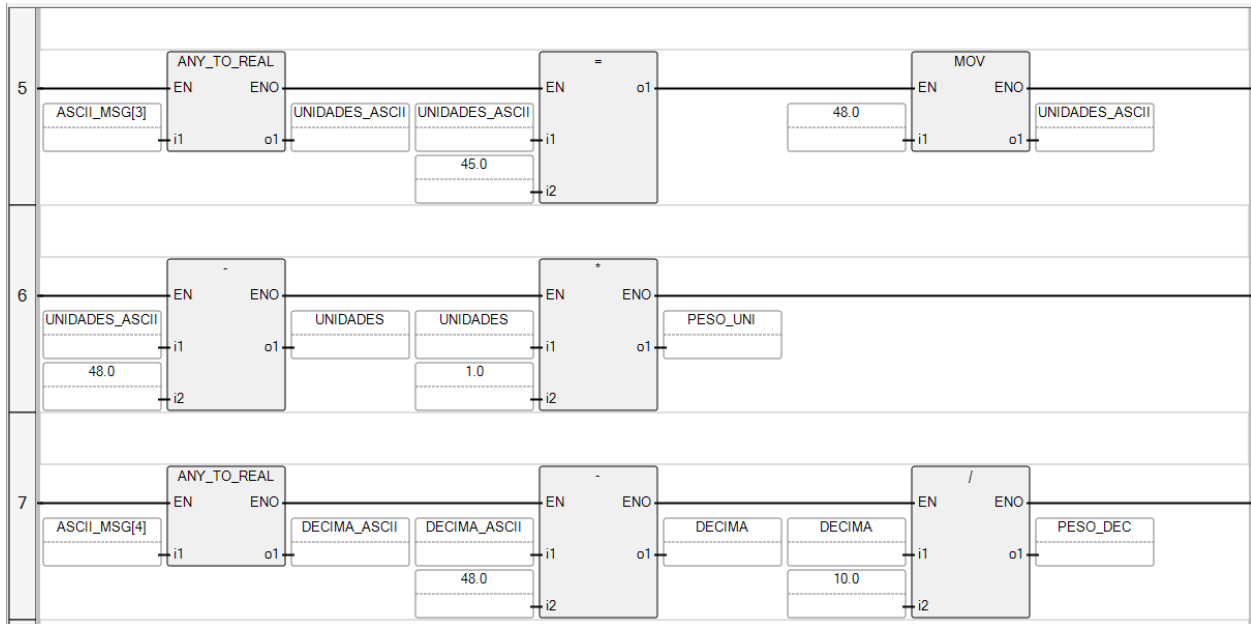


Figura 22. Procesado de lectura ASCII

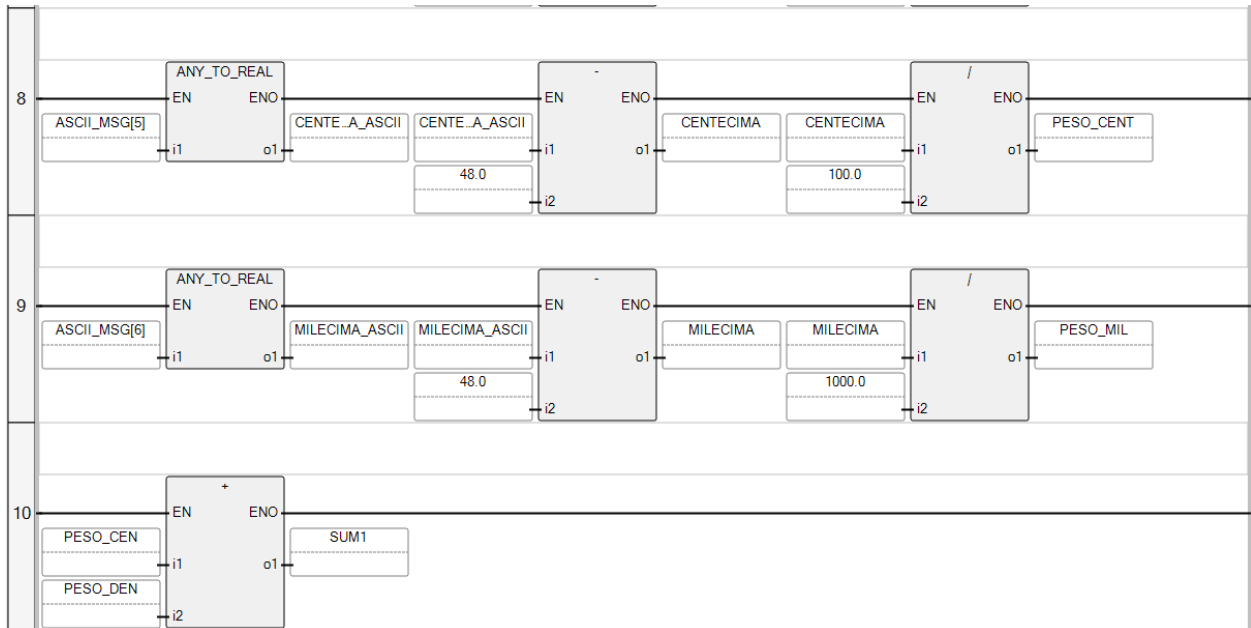


Figura 23. Procesado de lectura ASCII



Figura 24. Procesado de lectura ASCII

Por otro lado, el software PLC funciona como un servidor Modbus, para que el software C# de la siguiente etapa sea capaz de leer datos del PLC o bien escribir en el mismo PLC. Para esto se debe realizar una configuración dentro del PLC y habilitar las direcciones Modbus según su función, como se muestra en la tabla 1. Específicamente debe configurar la IP del PLC, configurar el driver a tipo ASCII, tipo de puerto serial RS-485 y tabla de direcciones Modbus. Según el protocolo Modbus TCP/IP, las direcciones por defecto para registros de palabras son las direcciones 4000x y las direcciones de bits de salida son las 0000x y las de bit de entrada son las 1000x, tal y como se muestra en la figura 27.

Controller - Ethernet

Port Settings

Port State: Enabled Disabled

Auto-Negotiate Speed and Duplex Mode

Internet Protocol (IP) Settings

Obtain IP address automatically using DHCP

Configure IP address and settings

IP Address:

Subnet Mask:

Gateway Address:

Detect duplicate IP address

Figura 25. Configuración de IP del PLC

Controller - Serial Port

Common Settings

Driver:

Baud Rate:

Parity:

Protocol Control

Media:

Control Line:

Deletion Mode:

Data Bits: XON/XOFF

Stop Bits: Echo Mode

Append Chars: ,

Termination Chars: ,

Figura 26. Configuración de puerto de comunicación del PLC

Controller - Modbus Mapping

⚠ To communicate via Ethernet, ensure Modbus TCP 'Server State' is 'Enabled' on [Ethernet Configuration Page](#).

Add Duplicate Import Export

| Variable Name | Data Type | Address | Addresses Used |
|-----------------------|-------------|---------|----------------|
| MODBUS_DIGITAL_OUTPUT | BOOL[1..10] | 00001 | 00001 - 00010 |
| MODBUS_DIGITAL_INPUT | BOOL[1..10] | 10001 | 10001 - 10010 |
| MODBUS_ANALOG_OUTPUT | WORD[1..10] | 40001 | 40001 - 40010 |

Figura 27. Tabla de direcciones Modbus

En la figura 28 se muestra la programación de las variables Modbus, la variable W_PESO contiene el peso total, el cual se obtuvo al realizar la suma SUM2 + SUM 4 de la figura 23, este valor de peso se mueve a la variable MODBUS_DIGITAL_OUTPUT que es un registro de salida de la tabla Modbus, un bit de peso estable (STABLE) que se mueve a los bit Modbus de salida y por último tres bits de entrada al PLC, para que el software C# escriba al PLC y encienda las luces verde, amarilla o roja según corresponda, por ejemplo, si los pesos de los productos coinciden con el peso de los productos que el cliente selecciona que lleva adquiridos, registrados en una base de datos (etapa no abarcada en el presente proyecto), entonces se enciende la luz verde la cual indica que se puede proceder al pago.

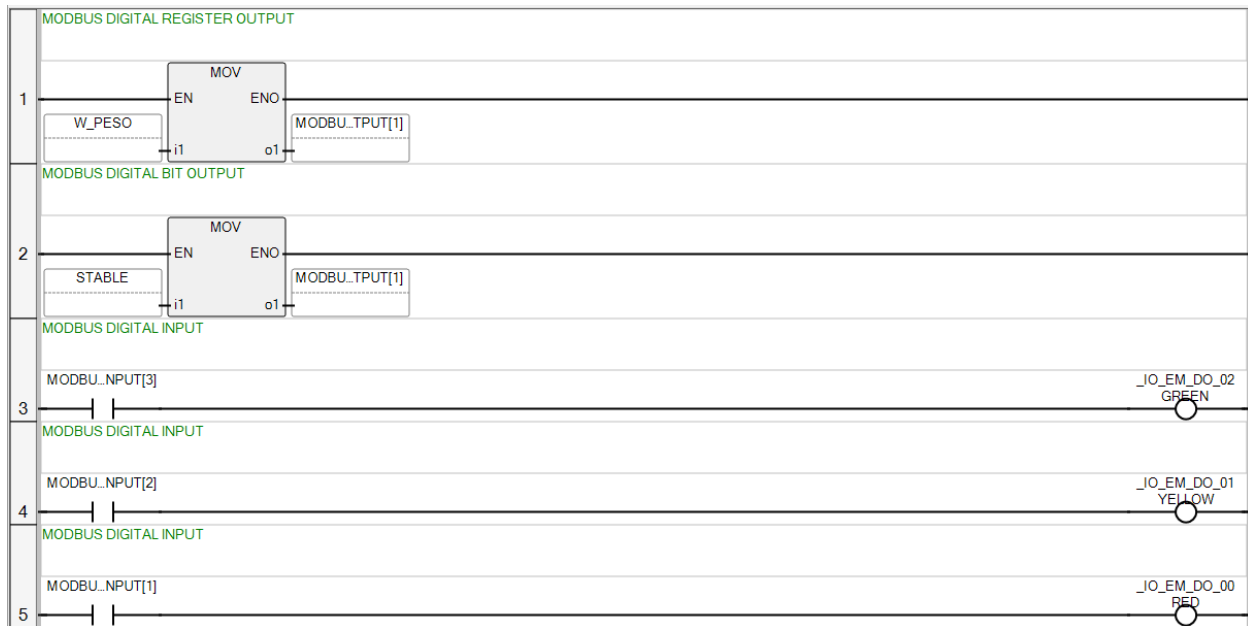


Figura 28. Salidas y entradas del Servidor Modbus dentro del software del PLC

4.4. Implementación de software de enlace a la web

El software desarrollado se describe gráficamente en la figura 29, en donde se muestra que se compone resumidamente de 3 partes. El BackEnd que corresponde al API encargado de recibir los datos provenientes del hardware con formato JSON y de toda la estructura necesaria para enviar los datos a la base de datos, la cual corresponde a la segunda parte del software, la cual almacena y pone a disposición la información del hardware para que cualquier dispositivo pueda adquirir los datos. La tercera parte del software es una interfaz o FrontEnd para pruebas del alcance del software desarrollado.

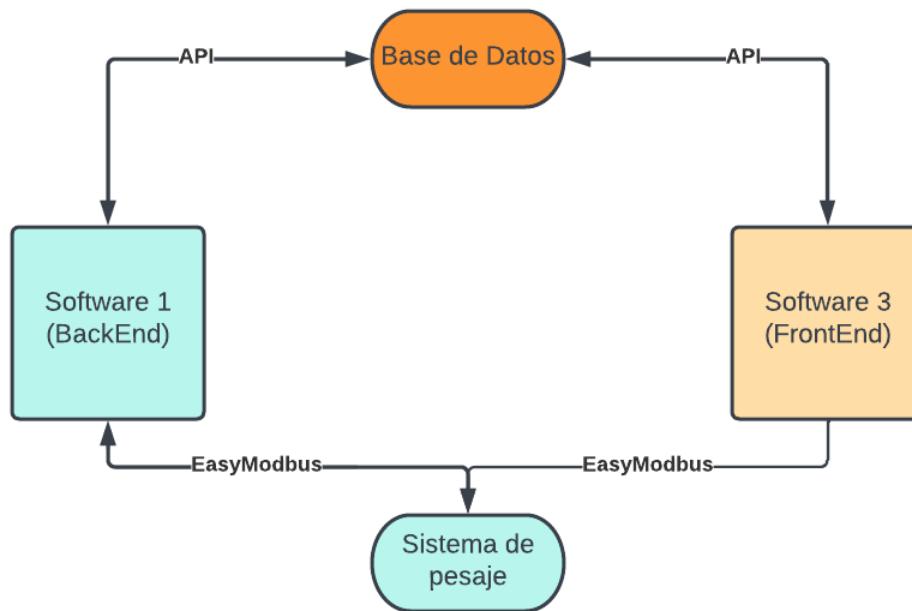


Figura 29. Diagrama de trabajo del software de enlace a la web.

El alcance del proyecto fue enlazar a internet el dispositivo diseñado, de lo mostrado en la figura 29, los encargados de cumplir con ese alcance son el BackEnd y la base de datos, para la etapa de FrontEnd se desarrolló una interfaz gráfica, para realizar pruebas del software, es decir, mostrar en pantalla que es posible leer los datos de peso en tiempo real y poder encender las indicaciones (luces) del hardware.

Esta interfaz desarrollada no corresponde a la interfaz final del proyecto terminado, ya que no se encontraba dentro de los alcances del proyecto, sin embargo para probar lo desarrollado de la mejor manera, se creó una interfaz capaz de lograr parte de lo requerido (escribir y leer de la base de datos el peso, mostrar la conexión con el hardware para enviar información de encender las indicaciones o luces de este).

La base de datos utilizada se creó mediante un SQL Server con el software Microsoft SQL Server Management Studio 18, su código de implementación se muestra en la figura 30. En donde se crea una tabla con las variables que se van a llenar con la información del hardware.

```
BD_BASCULA.sql - D:\EEUKF\Sergio (51)
create database BD_Bascula

use BD_Bascula

create table BasaculaInf(
  Id int Identity(1,1) not null,
  Action int not null,
  Turn bit not null,
  Stable bit null,
  Size decimal (18,0) null
)
```

Figura 30. Base de datos

Id: Identificador del consecutivo de datos escritos

Action: 0,1 y 2 para identificar las tres luces del semáforo de indicaciones

Turn: Bit para encender las luces de indicación

Stable: Bit de peso estabilizado en la báscula

Size: Decimal del peso registrado en tiempo real en la báscula.

Como se observa, la base de datos interactúa con información encapsulada en formato de objeto JSON, para evitar la complejidad de trabajar con archivos tipo TEXT. Para mapear esta información y para establecer conexión con la base de datos se utilizó un ORM (Object Relations Mappers) llamado Dapper y un ORM llamado Entity Framework Core 6, básicamente el objetivo de la utilización de un ORM fue para facilitar la interacción con las tablas de la base de datos, sin la necesidad de preocuparse por las filas y las columnas de las bases de datos. Por otro lado para lograr la interacción Base de Datos – Cliente, se implementó una Interfaz de Programación de Aplicaciones (API). Estos conceptos fueron desarrollados en el marco teórico del proyecto.

Para visualizar los Endpoints (Get, Post, Delete, etc) utilizados para interactuar con la base de datos, se utilizó una herramienta de código abierto llamada Swagger, utilizada comúnmente para documentación de proyectos o servicios web diseñados y utilizados. Esta herramienta fue de utilidad para documentar las etapas de lectura de información provenientes del hardware del proyecto y demostrar el correcto funcionamiento de esta etapa software. En la figura x se muestra esta herramienta y se muestran los Endpoint creados para efectos de este proyecto, en donde en la etapa de resultados se muestran estos Endpoints con información leída de la base de datos y con información que se llevó del hardware hacia la base de datos en tiempo real.

Para extraer información del hardware se utilizó una librería de C# llamada EasyModbus, la cual permite leer y escribir los registros de información compartida en una red de datos con protocolo Modbus TCP/IP, indicando la dirección IP y el puerto (502 por default en Modbus) del dispositivo hardware en la red, es posible leer sus registros (Figura 27).

En el anexo 6 se muestra la totalidad del código C# implementado, el cual se va a describir de manera general cada script de código por su nombre en la sección de resultados.

5. Resultados y Análisis

5.1. Validación

Plataforma de pesaje

Para la validación de la plataforma de pesaje, la cual corresponde al objetivo 1 del proyecto, se realizó una simulación de la estructura de la plataforma, con la herramienta de simulación de SolidWorks, con el objetivo de determinar un factor de seguridad mayor a 1.5 para el peso máximo de medición de 200 Kg.

Luego también se realizó una simulación de la deflexión que se presenta idealmente para los 200 Kg, lo cual es importante de conocer para verificar que no sufre

una deflexión a un punto que se entre en el área de deformación permanente para el material utilizado (ANSI 1020) o un punto cercano que genere que el sistema de medición tenga un tiempo de respuesta muy lento, es decir, que la deformación de la estructura dure un tiempo considerable en volver a su condición normal y se obtengan falsas mediciones de peso durante ese tiempo de volver a cero.

Los resultados obtenidos se observan en las siguientes figuras.

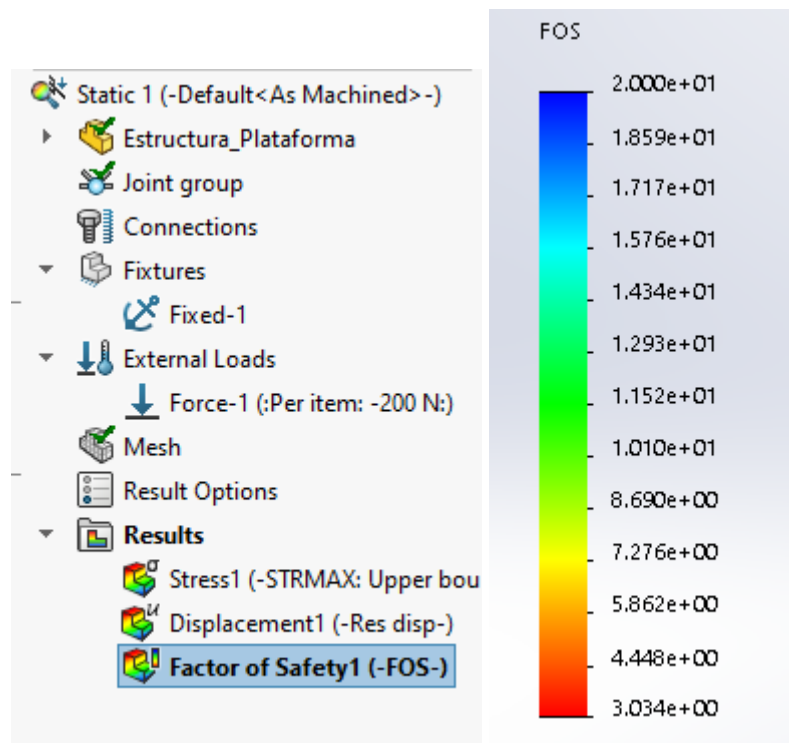


Figura 31. Parametros de simulación y gráfica escala de Factor de seguridad (Software: SolidWorks)

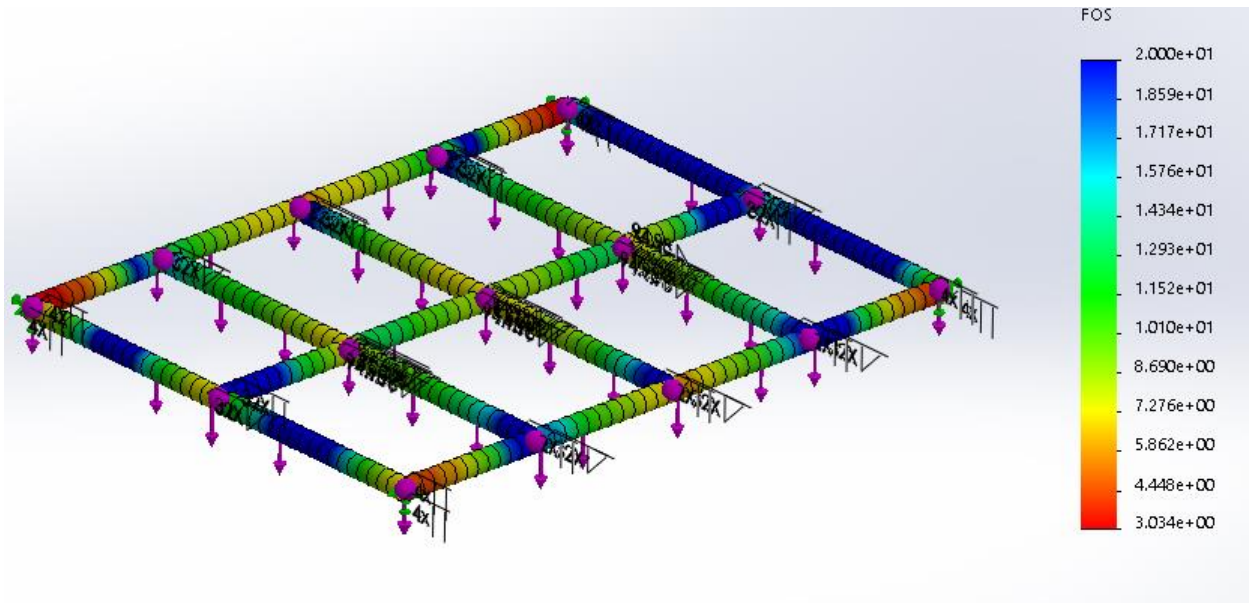


Figura 32. Estructura de simulación (Software: SolidWorks)

Como se observa en la gráfica de escala de factores de seguridad de la simulación, los elementos de color rojo son los elementos más críticos con respecto al esfuerzo elástico del material (ANSI 1020), el cual es de 351.571 Mpa. Estos elementos en rojo presentan un factor de seguridad de 3 y los elementos por encima de esos en la escala presentan un factor de seguridad aún mayor, por ejemplo los de color azul que presenta un factor de seguridad de 10.

| Property | Value | Units |
|-------------------------------|---------|-------------------|
| Elastic Modulus | 200000 | N/mm ² |
| Poisson's Ratio | 0.29 | N/A |
| Shear Modulus | 77000 | N/mm ² |
| Mass Density | 7900 | kg/m ³ |
| Tensile Strength | 420.507 | N/mm ² |
| Compressive Strength | | N/mm ² |
| Yield Strength | 351.571 | N/mm ² |
| Thermal Expansion Coefficient | 1.5e-05 | /K |

Figura 33. Características de acero ANSI 1020 (Software: SolidWorks)

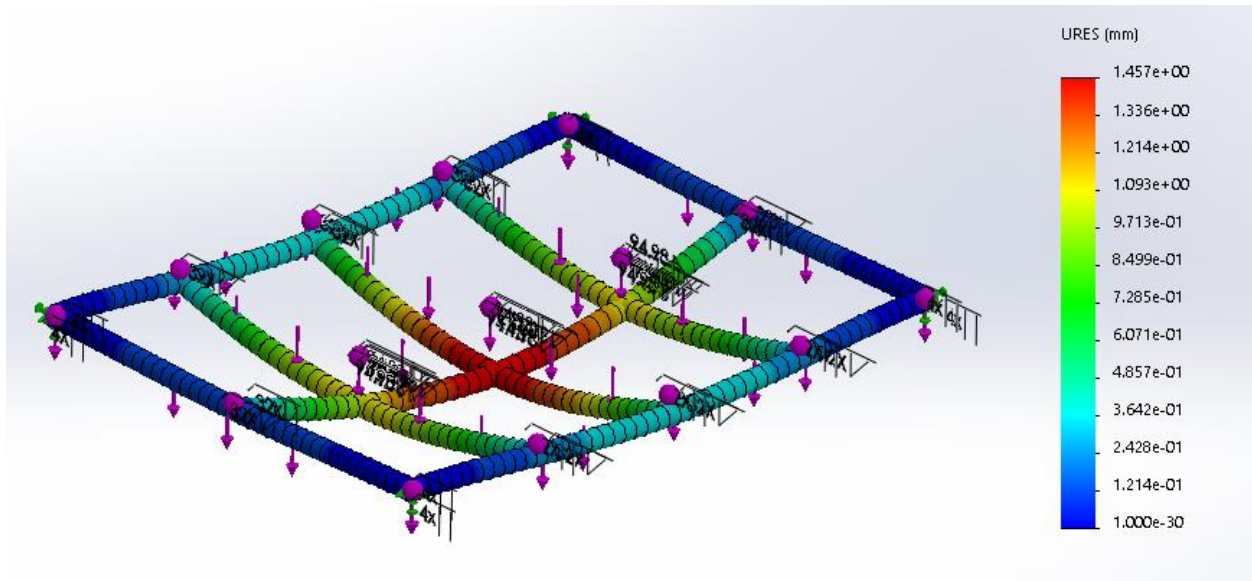


Figura 34. Estudio de deflexión (Software: SolidWorks)

Con respecto al estudio de deformaciones, se realizó una simulación de desplazamiento o deflexión de la plataforma, en donde se muestra que la deflexión mayor se da en el centro de la plataforma, con un desplazamiento de 1.457 mm. Lo cual es un desplazamiento bastante pequeño, como se pretendía al diseñar una plataforma más robusta.

Sistema de pesaje

En la siguiente figura se muestra el gabinete implementado para el sistema eléctrico de pesaje, el cual contiene el transmisor de peso, en donde llegan las señales de las celdas de carga y el PLC utilizado para la etapa posterior.

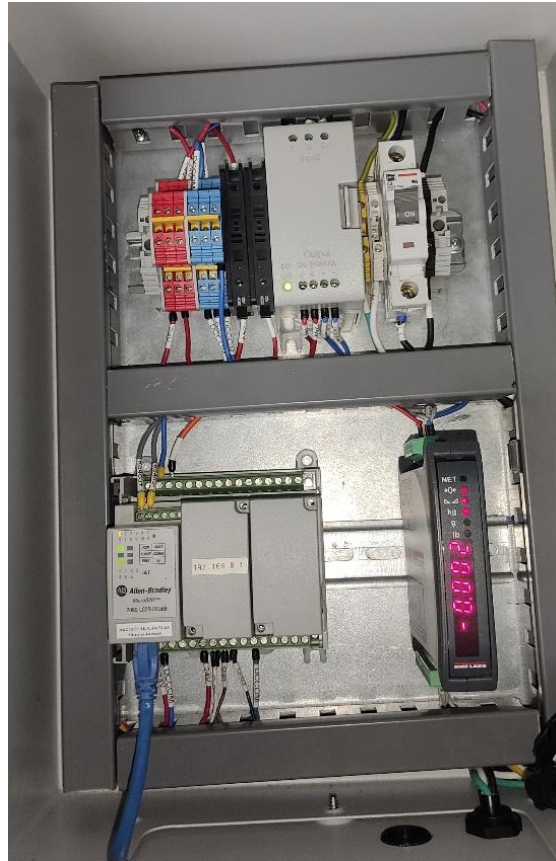


Figura 35. Sistema eléctrico de pesaje

Para la validación del sistema eléctrico de pesaje, como se declara en la metodología, se realizaron una serie de mediciones con un peso fijo, de aproximadamente 38 Kg. El objetivo de estas mediciones fue caracterizar el sistema de pesaje con respecto a que tanto varía la medición de un mismo peso, el valor marginal máximo que se permite para validar a lo que corresponde el objetivo 2 del presente proyecto es de 50 g ó 0.050 Kg de variación de la medición de un mismo peso con una confianza del 95% de que la media de las mediciones no tengan un error mayor a 0.02.

La muestra de pesos que se utilizó para conocer cuántas mediciones realizar para tener un 95% de confianza fue de 35 mediciones de pesos (Anexo 4), el tamaño de la muestra se eligió basándose en lo mencionado en el libro de Probabilidad y Estadística para Ingeniería y Ciencia. En donde se dice que para poblaciones de las cuales no se

conoce la varianza de la población pero se desea conocer el número de mediciones para cierta confianza, utilizar la desviación de la muestra (S).

“En términos estrictos, la fórmula del teorema 9.2 solo será aplicable si se conoce la varianza de la población de la cual se seleccionó la muestra. Si no contamos con esa información, podríamos tomar una muestra preliminar de tamaño $n \geq 30$ para proporcionar una estimación de σ . Después, usando S como aproximación para σ en el teorema 9.2, podemos determinar aproximadamente cuantas observaciones necesitamos para brindar el grado de precisión deseado.” [15]

Como se muestra en el anexo 4, la media de la muestra inicial fue de 38.513 Kg y la desviación estándar de 0.023 Kg. El resultado que se obtuvo al utilizar el teorema 9.2 fue de 6 mediciones redondeando, para una confianza de 95% de que el error no será mayor a 0.02 como se muestra en la ecuación 6. Donde e es el error de 0.02, σ la desviación estándar y $Z_{\frac{\alpha}{2}}$ es un estadístico de tablas que se obtiene del porcentaje de confianza que se desea, el cual es de 1.96 para una confianza de 95%.

$$n = \left(\frac{Z_{\frac{\alpha}{2}} \cdot \sigma}{e} \right)^2 = \left(\frac{1.96 \cdot 0.023}{0.02} \right)^2 = 5.08 = 6 \quad (7) \text{ Teorema 9.2}$$

Al utilizar las 6 nuevas mediciones para determinar la desviación de los datos y la media, se obtuvo como resultado una media de peso medido de 38.503 Kg con una desviación de 0.011 Kg.

Tabla 23. Mediciones de pesos

| Número de medición | Medición en Kg | $\sum_1^n (x_i - \bar{x})^2$ |
|--------------------|----------------|------------------------------|
| 1 | 38.501 | 5.44444E-06 |

| | | |
|------|--------|-------------|
| 2 | 38.495 | 6.94444E-05 |
| 3 | 38.489 | 0.000205444 |
| 4 | 38.52 | 0.000277778 |
| 5 | 38.512 | 7.51111E-05 |
| 6 | 38.503 | 1.11111E-07 |
| Suma | 231.02 | 0.000633 |

$$\bar{x} = \left(\sum_1^n M_n \right) \frac{1}{n} = \frac{231.02}{6} = 38.503 \quad (\text{Media})(8)$$

$$S = \sqrt{\frac{\sum_1^n (x_i - \bar{x})^2}{n-1}} = \sqrt{\frac{0.000633}{5}} = 0.011 \quad (\text{Desviación Estandar de la muestra})(9)$$

$$\bar{x} \pm Z_{\alpha} \frac{S}{\sqrt{n}} = 38.495 < \bar{x} < 38.511 \quad (\text{Intervalo de confianza 95%})(10)$$

Para lograr un resultado más representativo para la aplicación, se realizaron 6 mediciones más pero con un peso mayor, para compararlo con la desviación obtenida para el peso de la tabla 23. Se utilizó un peso representativo para una aplicación promedio del dispositivo, es decir, unos 80 o 90 Kg que se considera una cantidad de productos estándar de compras y nos permite comprar con un peso mayor y evaluar el error para dos extremos de funcionamiento.

Tabla 24. Mediciones de pesos

| Número de medición | Medición en Kg | $\sum_1^n (x_i - \bar{x})^2$ |
|--------------------|----------------|------------------------------|
| 1 | 86.596 | 2.5E-05 |
| 2 | 86.613 | 0.000144 |

| | | |
|------|---------|----------|
| 3 | 86.587 | 0.000196 |
| 4 | 86.609 | 6.4E-05 |
| 5 | 86.606 | 2.5E-05 |
| 6 | 86.595 | 3.6E-05 |
| Suma | 519.606 | 0.000490 |

$$\bar{x} = \left(\sum_1^n M_n \right) \frac{1}{n} = \frac{519.606}{6} = 86.601 \text{ Kg (Media)}(11)$$

$$S = \sqrt{\frac{\sum_1^n (x_i - \bar{x})^2}{n - 1}} = \sqrt{\frac{0.000490}{5}} = 0.0098 \text{ Kg (Desviación Estandar de la muestra)}(12)$$

$$E = 10.9 \% \text{ (Error entre dos mediciones)} (13)$$

Las diferencia entre las ecuaciones 9 y 12 es de 0.0012 Kg, lo cual representa un 10.9 % del resultado de la ecuación 9, sin embargo, si se observa la desviación para esta muestra de datos es menor a la muestra anterior, por lo que el error de 10.9%, no es un error que afecte al garantizar una repetibilidad de 11 g con 95 de confianza.

Se consultó el libro Elementos de Muestreo de Mendelhall, especializado en muestreo estadístico, con el objetivo de fundamentar el criterio expuesto anteriormente, de que para utilizar el teorema 9.2 con la utilización de la distribución $Z_{\frac{\alpha}{2}}$ se debe tener una muestra mayor a 30. La bibliografía consultada menciona que efectivamente, las estimaciones del intervalo de confianza de medias, normalmente se basa en la distribución t de Student, en lugar de en la distribución normal z, pero que para el caso de aplicaciones reales a ser los tamaños de muestra grandes, que el valor de z funciona bien [16].

Automatización de adquisición de pesos

Como se ha mencionado, automatizar la toma de mediciones hace referencia a adquirir los datos en tiempo real o bien adquirir lo más rápido posible esta información proveniente de las celdas de carga y transmisor de peso, considerando en el proceso de adquisición, todo el procesamiento de la información de acuerdo al formato que esta viene encapsulada. Se buscó caracterizar la etapa de automatización de medición con el tiempo que le toma al controlador desarrollar la lectura y el procesamiento de las mediciones.

Los resultados que se obtuvieron, los cuales corresponden a la validación del objetivo 3 del presente proyecto, se muestran a continuación:

De la misma forma en que se trabajó para la validación del objetivo 2, se determinó una muestra de 35 mediciones del tiempo de respuesta o procesamiento de los datos de peso que ingresan al controlador por el puerto serial RS-485, con el objetivo de determinar la cantidad de mediciones que garantizan una confianza del 95% de que el error de la media de los datos no supera 0.2 segundos.

Las mediciones realizadas se muestran en el anexo 5 y como se mencionó brevemente en la metodología, para la realización de estas mediciones se utilizó una herramienta llamada "Trend" del propio controlador, que lo que hace es monitorear en el tiempo una variable contenida en el programación del controlador, en este caso se monitoreó la variable que contiene el valor del peso en decimal, y se tiene la posibilidad de exportar los datos a formato .CSV.

Para lograr aproximar el tiempo de respuesta, luego de varias formas e intentos para lograrlo, lo que se realizó fue colocar pesos manejables para la cantidad de pruebas realizadas, de aproximadamente 2.8, 5.5, 12.8 Kg en la báscula, esto antes de iniciar el Trend. Luego se inicia el Trend, a los 5 segundos transcurridos se retira el peso, generando un gráfico como la que se muestra en la figura 36. El objetivo de esto es conocer el tiempo que tarda el sistema en registrar que la medición es 0 Kg, ya que teóricamente el peso se colocó por 5 segundos pero las mediciones muestran lo contrario, ya que se considera el tiempo de respuesta del sistema, que se obtiene al

realizar la resta del tiempo experimental transcurrido T_{Exp} menos los 5 segundos T_{Real} que realmente estuvo el peso en la báscula.

$$T_{Exp} - T_{Real} = T_{Resp} \quad (\text{Tiempo de respuesta}) \quad (14)$$

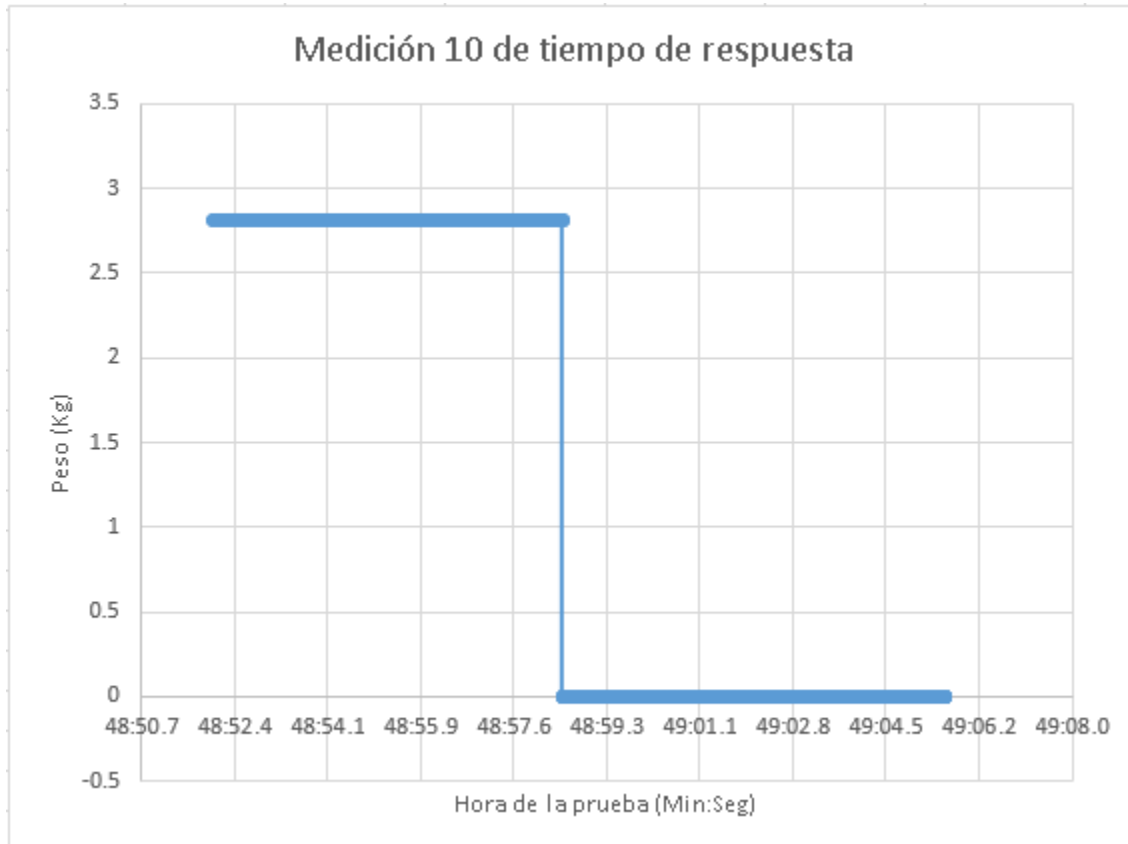


Figura 36. Gráfica de medición de tiempo de respuesta (Software: Excel)

Tabla 25. Tiempos de respuesta en segundos

| | | | | |
|-----|-----|-----|-----|-----|
| 2.1 | 1.1 | 0.6 | 1.1 | 1.2 |
| 1.1 | 1.5 | 1.9 | 1.0 | 1.5 |
| 1.9 | 1.1 | 1.5 | 0.9 | 2.0 |
| 1.2 | 0.9 | 1.5 | 1.5 | 1.0 |
| 2.0 | 1.5 | 1.1 | 1.5 | 1.6 |
| 1.7 | 1.7 | 1.5 | 1.6 | 1.1 |
| 1.7 | 1.9 | 1.2 | 1.2 | 1.1 |

La media y desviación estándar de los datos fue de 1.4 y 0.369 respectivamente, al aplicar la ecuación 7, da como resultado 13.077, lo cual se debe redondear al entero superior para conocer la cantidad de mediciones extra para que la media de los datos no tenga un error mayor a 0.2 segundos con una confianza del 95%.

$$n = \left(\frac{Z_{\alpha/2} \cdot \sigma}{e} \right)^2 = \left(\frac{1.96 \cdot 0.369}{0.2} \right)^2 = 13.077 = 14 \quad (15)$$

Al realizar las 14 nuevas mediciones, se obtuvo una media del tiempo de respuesta del sistema de 1.58 segundos y una desviación de 0.207 segundos con una confianza del 95% de que la media del tiempo de respuesta tenga un error menor a 0.2 segundos.

Tabla 26. Tiempos de respuesta en segundos

| | | | |
|-----|-----|-----|-----|
| 1.1 | 1.4 | 1.7 | 1.7 |
| 1.6 | 1.8 | 1.6 | 1.5 |
| 1.6 | 1.5 | 2.0 | 1.5 |
| 1.7 | 1.5 | | |

$$\bar{x} = \left(\sum_1^n M_n \right) \frac{1}{n} = \frac{22.2}{14} = 1.58 \text{ s} \quad (\text{Media})(16)$$

$$S = \sqrt{\frac{\sum_1^n (x_i - \bar{x})^2}{n - 1}} = \sqrt{\frac{0.000633}{13}} = 0.207 \text{ s} \quad (\text{Desviación Estandar de la muestra})(17)$$

$$\bar{x} \pm Z_{\alpha/2} \frac{S}{\sqrt{n}} = 1.472 < T_{Resp} < 1.688 \quad (\text{Intervalo de confianza 95%})(18)$$

Las 35 y 14 mediciones realizadas se muestran en los anexos 5 y 6 respectivamente, esta información obtenida de Excel fue recortada para no mostrar la

totalidad de los datos, ya que el muestreo de los datos fue en milisegundos y serían muchas páginas para mostrarlo, por lo que se muestra el inicio en la columna izquierda y el final de los datos en el columna derecha. Otro aspecto importante de esta información es el eje del tiempo, el cual es la hora en que se tomaron los datos con el formato (Minutos: Segundos) para mostrar la realidad y confiabilidad de la cantidad de mediciones que se realizaron. Las mediciones realizadas con resultados en bruto están disponibles en el siguiente enlace: [LibrosExcel](#)

Software de enlace a la web

Como se mencionó anteriormente, los scripts de código C# desarrollados se muestran en el anexo 7.

1. Create.cs: corresponde al código que encapsula la información para realizar la petición de ingresar datos a la Base de Datos (BD).

2. Find.cs: es la contraparte del anterior, este determina como y que se quiere buscar en la BD.

3. BascualInf.cs: Corresponde al objeto modelo que encapsula la información leída de la BD.

4. SemaforoController.cs: Corresponde al controlador del API, este controlador instancia los Endpoints que se van a utilizar, para este caso se utiliza GET y POST.

5. DbContext .cs y 6. IDbContext.cs: Corresponde a un contrato de conexión con la BD, es decir, instancia las funciones de conexión y luego se tiene la interface (IDbContext.cs) en donde se muestra lo que realiza cada función instanciada utilizada.

7. ReadConnection.cs y 8. IReadConnection.cs: Corresponde al contrato de lectura de la BD, las funciones que se utilizan para leer la BD y su interfaz, que corresponde a la lógica de cada función utilizada.

9. WriteConnection.cs y 10. IWriteConnection.cs: Corresponde al contrato de escritura de la BD, las funciones que se utilizan para escribir en la BD y su interfaz, que corresponde a la lógica de cada función utilizada.

11. Program.cs: Corresponde al script principal del BackEnd, este levanta el servidor, instancia el servicio Swagger basado en HTTP.

12. appsettings.json: Establece la conexión con la BD creada, donde se le pasa la dirección local de la BD y el nombre de la tabla de la BD.

13. Form1.cs: Corresponde a la interfaz de usuario para realizar pruebas de validación, en donde también se genera la conexión modbus con el hardware, y se crean los modelos tipo JSON para enviar y recibir la información de la BD.

En la siguiente figura se observa ejecutada la interfaz de usuario, en donde se observan los botones para iniciar comunicación con el hardware, el botón para enviar la información del hardware a la BD y luego leer desde la BD hacia la pantalla. Se muestran también los botones de instrucción, los cuales se encuentran configurados para que el usuario encienda las luces del hardware a manera de prueba, ya que no estaba en los alcances enviar información de la interfaz al software, solamente generar la comunicación y estructura en la BD para lograrlo.



Figura 37. Interfaz de usuario

En el siguiente enlace queda a disposición un corto video del funcionamiento detallado del sistema, en donde se muestra la colocación de pesos, la lectura del peso con el software PLC y la lectura del PLC con el software C# y por último el ingreso exitoso de los datos a la base de datos: [Video](#)

Finalmente para caracterizar este software se agregó a la interfaz una caja de texto donde se muestra el tiempo que tarda el sistema en guardar y leer la información en la Base de Datos, esto se realizó tomando el tiempo actual antes de realizar el Post con la siguiente línea de código: `DateTime inicio = DateTime.Now;` luego, al finalizar el Get, se mide el tiempo que transcurrió en ambas tareas haciendo una resta de los dos tiempos capturados, con la siguiente línea de código: `double time = DateTime.Now.Subtract(inicio).Milliseconds;`

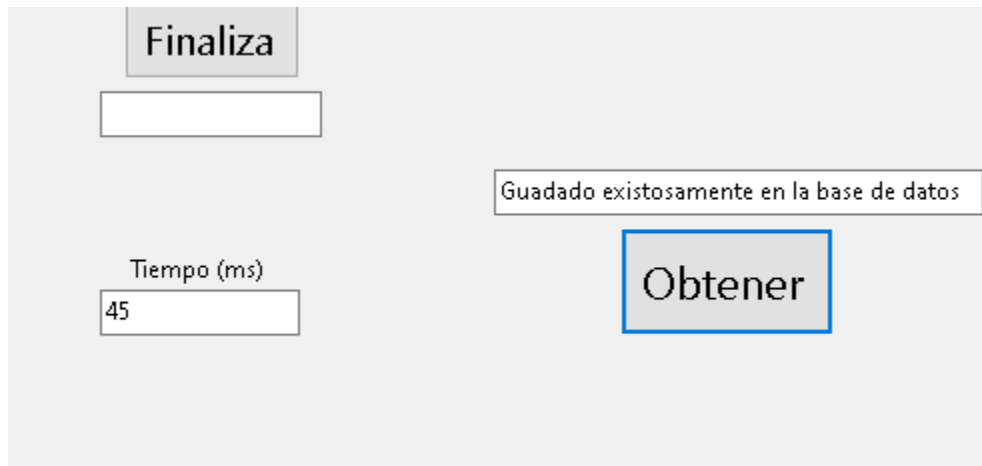


Figura 38. Prueba de tiempo de ejecución

Se realizaron 5 mediciones para generar un promedio y desviación de esta métrica.

$$T_m = \frac{45 + 51 + 48 + 61 + 76}{5} = 56.2 \text{ ms} \quad (19)$$

$$S = \sqrt{\frac{\sum_1^n (x_i - \bar{x})^2}{n - 1}} = \sqrt{\frac{634.8}{4}} = 12.59 \text{ ms} \quad (20)$$

5.2. Análisis Económico

Costo de desarrollo

Inicialmente se desglosan los costos de los recursos utilizados para el desarrollo del proyecto, en donde no se incluyen costos de licencias de software pero si componentes eléctricos y materiales para el mecanizado y ensamble del proyecto, así

como las horas de mano de obra de soldadura, ensamble y programación de los distintos software implementados.

Tabla 27. Recursos necesarios

| Número de recurso | Recursos |
|--------------------------|-------------------------------------|
| 1 | Software |
| 2 | Componentes Eléctricos |
| 3 | Materiales de mecanizado y cableado |
| 4 | Mano de obra |

Algunos software utilizados fueron de libre uso como C# y NanoCAD, sin embargo, los software de programación del controlador y software de diseño como Connect Component Workbench (CCW) y SolidWorks si son software con su respectiva licencia, cuyo costo no se considera en este análisis.

El costo de la totalidad de los materiales para desarrollar una unidad del dispositivo se consideran debidamente, a pesar de que algunos componentes y materiales ya se encontraban disponibles para implementar el prototipo del dispositivo, se va a considerar el costo como si se fuera a implementar de forma definitiva para producción.

Tabla 28. Componentes eléctricos utilizados

| Componentes Eléctricos | Costo (Dólares c/u) |
|--|----------------------------|
| PLC | 225,00 |
| Celdas de carga | 848,00 (4 unidades) |
| Transmisor de peso | 550,00 |
| Fuente de poder | 49,320 |
| Protecciones Eléctricas | 42,00 |
| Interruptor Selector | 13.70 |
| Faro de luminarias | 26,990 |
| Inversión Fija en Componentes (IFC) | 1,755.1 |

Tabla 29. Materiales utilizados

| Materiales de Mecanizado y Cableado | Costo (Dólares) |
|---|------------------------|
| Perfil Acero 1020 | 63.052 |
| Lámina de Acero Inoxidable | 140,00 |
| Tornillería (Tornillos, Arandelas, Tuercas) | 11,50 |
| Material para soldadura | 32,00 |
| Cable eléctrico AWG 12 Verde | 11,97 |
| Cable eléctrico AWG 12 Naranja | 25,95 |
| Cable eléctrico AWG 12 Azul | 12,85 |
| Cable eléctrico AWG 12 Negro | 17,80 |
| Borneras y terminales | 27,99 |
| Gabinete | 149,600 |
| Inversión fija en Materiales (IFM) | 492.722 |

Por último, el costo de mano de obra de mecanizado, ensamble y programación es de mucha importancia para establecer el costo de producción inicial y para establecer el estimado de costo de venta del dispositivo, ya que por ejemplo, mano de obra de diseño mecánico, eléctrico y de programación, es un costo inicial del proyecto porque en caso de no necesitar modificación, solo se diseña una vez y para el costo de venta del dispositivo se debe considerar un costo de diseño pero no precisamente las horas de desarrollo de estos criterios por unidad desarrollada.

Tabla 30. Mano de obra requerida

| Mano de obra | Específico | Horas | Costo (Dólares) |
|----------------------|----------------------------------|--------------|------------------------|
| Mecánica | Diseño | 18 | 88,00 |
| | Corte y mecanizado | 2 | 8,15 |
| | Soldadura | 6 | 24,50 |
| | Ensamble | 4 | 16,30 |
| Eléctrica | Diseño | 16 | 78,30 |
| | Cableado | 4 | 19,50 |
| Programación | Software CCW | 28 | 137,00 |
| | Configuración Transmisor de peso | 1.5 | 7,33 |
| | Software C# | 36 | 176,00 |
| Desarrollo de planos | Planos Eléctricos | 4 | 19,50 |

| | | | |
|---|--------------------|----|--------|
| | Planos Mecánicos | 12 | 58,7 |
| Caracterización de equipo | Toma de mediciones | 3 | 14,7 |
| | Validaciones | 3 | 14,7 |
| Inversión en mano de obra (MO) | | | 662,68 |
| Inversión mano de obra fija por unidad (MOF) | | | 105,18 |
| Inversión mano de obra no fija por unidad (MONF) | MO - MOF | | 557,5 |

Realizando la suma de Inversión fija por unidad en materiales y componentes (IFC, IFM) y la Inversión inicial en mano de obra (MO), se obtuvo el costo total de inversión en implementación, cuyo valor es de 2,910.502 dólares.

Para obtener el costo de producción de cada unidad se sumó la Inversión fija por unidad en materiales y componentes (IFC, IFM), la Inversión fija en mano de obra (MOF) y 10% de la Inversión no fija en mano de obra (MONF), cuyo costo de producción es de 2,408.75 dólares, se considera el 10% para en base a este costo calcular el precio de venta.

Beneficios del proyecto

Para el proyecto desarrollado se habla de dos tipos de beneficios, los beneficios suaves y los beneficios duros. Los suaves son beneficios que genera a otros externos a la empresa que implementa y vende el proyecto y los duros son los beneficios económicos directamente de la empresa que vende el dispositivo.

Los beneficios duros que recibe la empresa con el desarrollo del proyecto es iniciar la venta en serie de un producto, en donde anualmente se tenga prevista la cantidad de unidades que se van a desarrollar y vender a las distintas localidades, esto trae el beneficio a la empresa de mantener un flujo tanto de trabajo como de ganancias constante, y no depender de proyectos que entren a la empresa de manera intermitente. Los beneficios duros se miden en ingresos económicos, para esto se debe considerar el

costo de producción, el costo de servicio después de la venta de cada dispositivo y el costo de venta del dispositivo, posteriormente en el análisis financiero se detalla una aproximación del ingreso por unidad vendida.

Los beneficios suaves que se pretenden ofrecer con la adquisición del dispositivo son, inicialmente brindar al cliente un servicio de asistencia en la instalación del dispositivo, así como un servicio de garantía en caso de anomalías, mal funcionamientos y averías del dispositivo el primer año de vendido. Este beneficio suave, aparte de generar confianza al cliente en la adquisición del dispositivo, genera relaciones comerciales sólidas entre ambas partes.

Otros beneficios suaves, se derivan de la visión que se tuvo para iniciar con el desarrollo del dispositivo, es decir, con un dispositivo que permite automatizar el cobro de productos se agiliza el flujo de compras en las localidades comerciales, por otro lado, al disminuir el factor humano con posibles errores a la hora del cobro de productos, se está contribuyendo en generar una sostenibilidad de negocio más óptima, ya que se disminuyen las pérdidas económicas.

Análisis financiero

Para realizar un flujo de caja del proyecto, se considera el año cero en el que se desarrollan cierta cantidad de unidades y se considera que en un plazo de 3 años se venda la cantidad desarrollada. Para este análisis financiero y viabilidad del proyecto se supone a manera de ejercicio que se realizó un estudio de mercado y de 20 unidades desarrolladas en el año 0, se venden 4, 6 y 10 unidades en los tres siguientes años respectivamente.

El egreso por unidad en el año cero es el costo total de implementación de los 20 dispositivos y los ingresos son el precio de venta del dispositivo, es decir, el costo de producción más un 50% de costo de oportunidad (3,613.125 dólares) menos los egresos de servicio que se les brinda a los clientes por concepto de seguimiento, instalación y garantía, que se estima de 200 dólares por unidad.

Tabla 31. Flujo de caja

| Año | Unidades | Ingreso (Dólares) | Egreso por desarrollo (Dólares) | Egreso de servicio por unidades vendidas(Dólares) | Ingreso Neto (Dólares) |
|-----|----------|-------------------|---------------------------------|---|------------------------|
| 0 | 20 | - | -48,676.752 | - | - |
| 1 | 4 | 14,452.5 | - | -800 | 13,652.5 |
| 2 | 6 | 21,678.75 | - | -1200 | 20,478.75 |
| 3 | 10 | 36,131.25 | - | -2000 | 34,131.25 |

Con el flujo de efectivo realizado es posible determinar la tasa interna de retorno del proyecto (TIR), el valor del TIR determina que la ganancia es igual a la inversión, es decir, que el valor presente neto es igual a cero ($VAN = 0$). El valor presente neto básicamente es llevar al año cero el flujo de caja o visto de otra forma, a cuanto equivalen los ingresos de los siguientes años al año que se invirtió [17].

$$Inversión = \sum_{i=0}^n \frac{Ingreso_i}{(1 + TIR)^i} \quad (19)$$

Donde:

$$n = 3 \text{ años}$$

$$Inversión = 48,676.752$$

$Ingreso_i$: Son los ingresos de cada año

$$TIR = 16\%$$

Con el valor del TIR, es posible determinar la viabilidad del proyecto, ya que, si luego de una investigación de mercado se determina que inversiones similares rentan por ejemplo un 10%, entonces como nuestro TIR es de 16%, quiere decir que el proyecto es viable, porque los ingresos llevados al año de la inversión generan ganancias con respecto a la inversión inicial.

6. Conclusiones y recomendaciones

Se validó el diseño de la plataforma metálica de pesaje implementada en el proyecto por medio de simulación, con un factor de seguridad mínimo de 3, tal y como se observa en la figura 32, lo cual cumple con el indicador meta del objetivo 1 del presente proyecto.

Se implementó un sistema eléctrico de pesaje con una repetividad o bien una desviación promedio de la mediación de un mismo peso no mayor a 11 gramos, con una confiabilidad del 95%, lo cual cumple la validación del objetivo 2 en donde se estableció una desviación de una misma medición no mayor a 50 gramos.

Se diseñó un sistema software de lectura del hardware y procesado de las mediciones de peso con una confiabilidad del 95% de que el tiempo de respuesta no supera 1.688 segundos, lo cual cumple la validación del objetivo 3 en donde se estableció un tiempo de respuesta no mayor a 1.7 segundos.

Se diseñó exitosamente un software para dejar a disposición los datos medidos del hardware en la red de internet, y se caracterizó el sistema con un tiempo de 56.2 ms en que realiza el posteo y lectura de la información a la base de datos.

Se recomienda que para la calibración del sistema de pesaje (Celdas de Carga y Transmisor de Peso), se realice una calibración teórica como la realizada en este proyecto si no se tiene un peso calibrado de mínimo la mitad del peso máximo de funcionamiento de la báscula, es caso contrario realizar la calibración real como indica el manual [13].

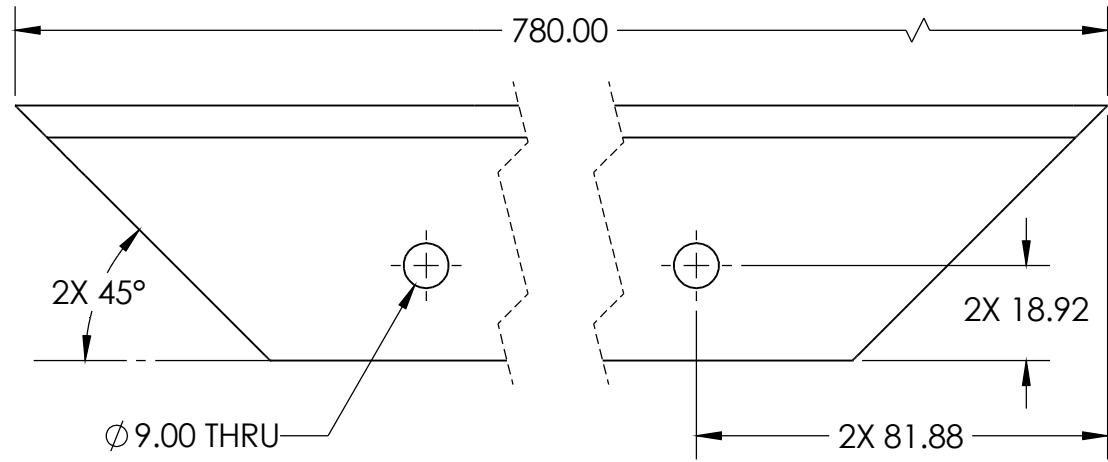
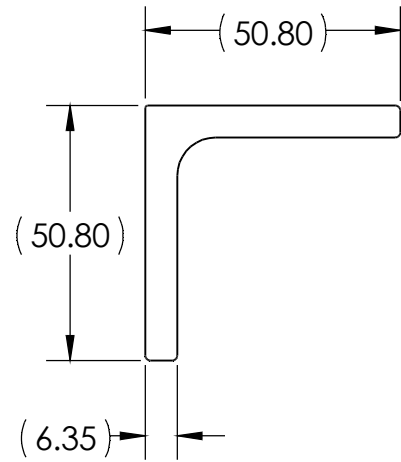
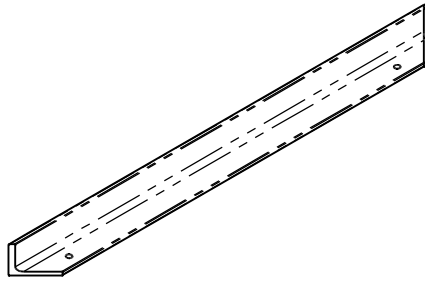
7. Referencias bibliográficas

- [1] R. Hibbeler, *Mecánica de Materiales*, México: PERSON EDUCATION, 2011.
- [2] P. Otero, "Diseño y construcción de una balanza electrónica con interfaz gráfica de visualización," Escuela Politécnica Nacional, Quito, 2007.
- [3] Prometec, "Galgas y Celulas de Carga," www.Prometec.net, [Online].
- [4] ZemicEurope, "Transmisores de Peso," [Online].
- [5] J.Cedillo, F.Rafael & L.Salas, "Optimización de ancho de banda para sistemas GSM," DSpace Repository, 2012.
- [6] B. Perrin, "The Art & Science of RS-485," Circuit Cellar Online, 1999.
- [7] A. Ruíz, A. Barandicar, F. Guerrero, "Implementación de un red ModbusTCP," *Ingeniería y Competitividad*, vol. Vol 6, no. No 2, 2004.
- [8] Modbus Organization INK, "MODBUS Application Protocol Specification V1.1b3," www.modbus.org, 2012.
- [9] M. Gallegos, "Introducción a los Servicios Web," in *Los Servicios Web XML*, 2011, pp. 72-97.
- [10] Vallés, P, "DISEÑO E IMPLEMENTACIÓN DE UNA BASE DE DATOS Y UNA APLICACIÓN," Universitat Politècnica de València, Valencia, España, 2019.
- [11] Marqués, M, "Bases de Datos," Sapientia, Universidad Saume, 2011.

- [12] LAUMAS Electrónica, "Normas para la correcta instalación," [Online]. Available: <https://www.laumas.com/es/productos/celulas-de-carga-y-accesorios-de-montaje/celulas-de-carga-de-flexion/>. [Accessed 15 08 2022].
- [13] Rice Lake Weighing Systems, "Global Weighing Systems," 22 Marzo 2022. [Online]. Available: <https://www.globalwsystems.com/productos/sct20>. [Accessed 20 Agosto 2022].
- [14] J. Sonnenberg, "Serial Communications RS485,RS422,RS232," Raveon Technologies Corp, 2018.
- [15] R. Walpole, R. Myers, S. Myers & Keying Ye, Probabilidad y estadística para ingeniería y ciencias. Novena edición, México: PEARSON EDUCACION, 2012.
- [16] W. Mendenhall, R. Scheaffer, R. Lyman, Elementos de muestreo, sexta edición, Madrid, España : Thomson , 2012.
- [17] Blank. L, Tarquin. A, "Ingeniería Económica," MacGrawHill, México, 2012.
- [18] J. Estrada, «Celdas de Carga,» Logicbus, México, 2018.

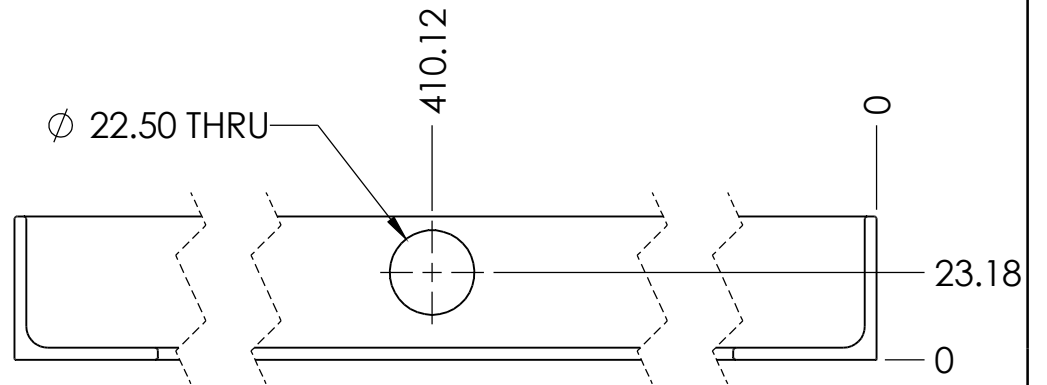
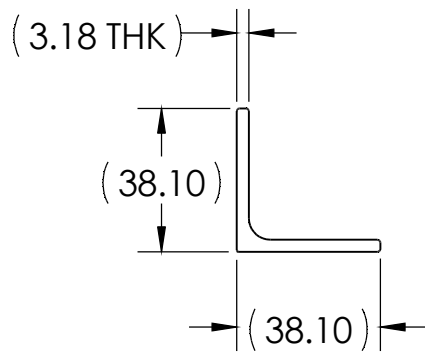
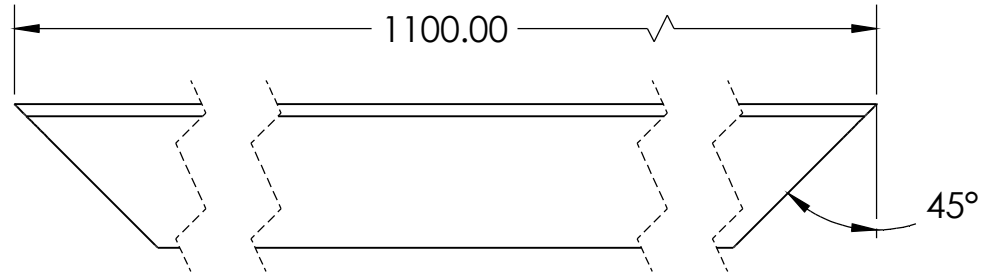
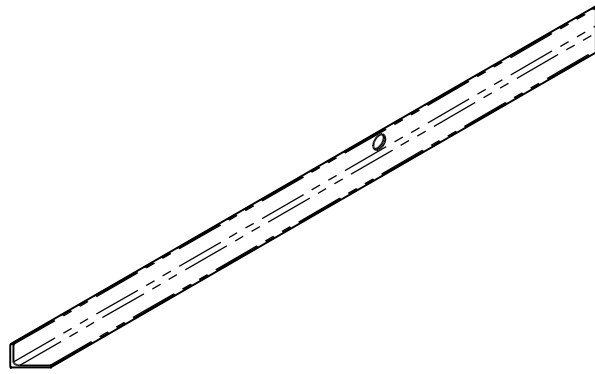
8. Anexos

ANEXO 1

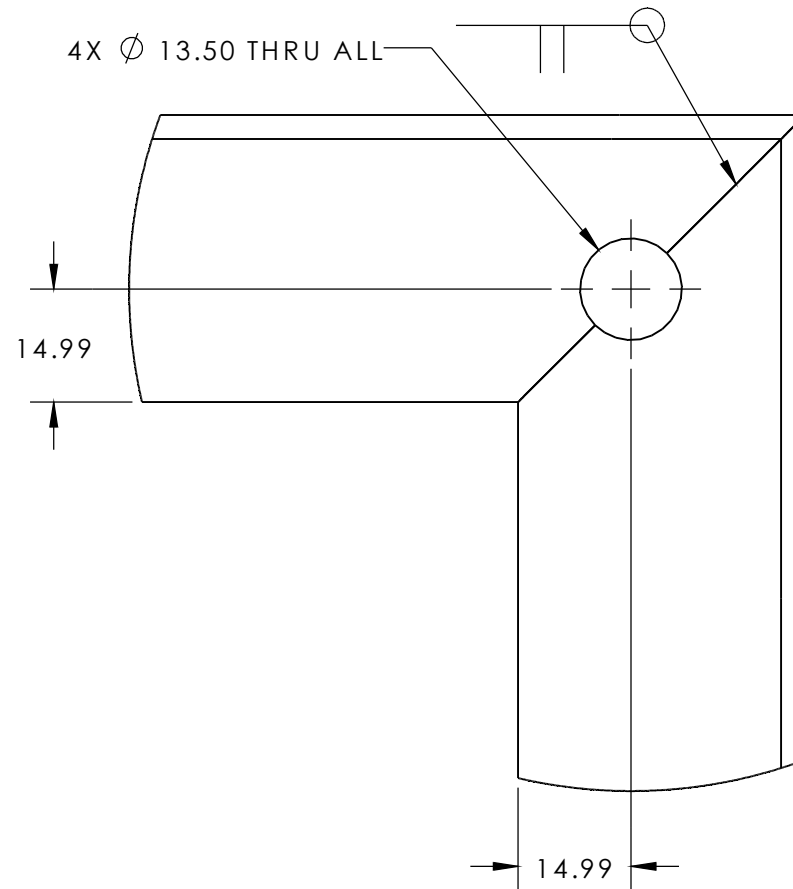
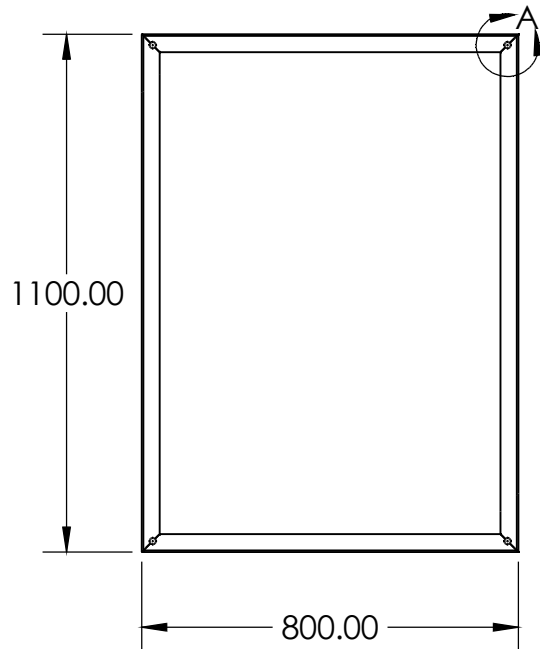


| | | | |
|---|-------------------------------|----------------------|----------------------|
| Creativa. <small>RAAL INDUSTRIAL S.A.</small> | NÚMERO DE DOCUMENTO | MATERIAL: | ACABADO SUPERFICIAL: |
| | CI - 2609 | ANSI 1020 | Pintado |
| | TÍTULO: | TRATAMIENTO TÉRMICO: | SUPERFICIE: |
| | ANGULAR CORTO | | |
| | CLIENTE: | DIBUJADO POR: | REV/VER |
| | | S. Rodríguez | A/A |
| | TERCER ANGLULO DE PROYECCION: | OP: | FECHA: |
| | | 169 | 19/08/22 |
| | | CANTIDAD: | REVISADO POR: |
| | | mm | Pendiente |
| | | | HOJA |
| | | | 1:8 |

| TOLERANCIAS | |
|--------------------|-------------|
| DIMENSIONES EN MM. | |
| INCH | MM |
| X ±.1 | X ± 0.25 |
| XX ±.01 | X ± 0.1 |
| XXX ±.005 | XX ± 0.05 |
| XXXX ±.0005 | XXX ± 0.010 |
| ANGULO ±2° | |



| | | | | | | | | | | | | | |
|---|------------------------------|----------------------|----------------------|----------|---------|---------|-----------|-----------|-------------|-------------|--|--|------|
| Creativa. <small>RAAL INDUSTRIAL S.A.</small> | NÚMERO DE DOCUMENTO | MATERIAL: | ACABADO SUPERFICIAL: | | | | | | | | | | |
| | CI - 2609 | ANSI 1020 | Pintado | | | | | | | | | | |
| | TÍTULO: | TRATAMIENTO TÉRMICO: | SUPERFICIE: | | | | | | | | | | |
| | ANGULAR LARGO | | | | | | | | | | | | |
| | CLIENTE: | DIBUJADO POR: | REV/VER | | | | | | | | | | |
| | | S. Rodríguez | A/A | | | | | | | | | | |
| | TERCER ANGULO DE PROYECCION: | OP: | FECHA: | | | | | | | | | | |
| | | 169 | 19/08/22 | | | | | | | | | | |
| | | CANTIDAD: | REVISADO POR: | | | | | | | | | | |
| | | mm | Pendiente | | | | | | | | | | |
| TOLERANCIAS <small>DIMENSIONES EN MM.</small> <table border="0"> <tr> <td>INCH</td> <td>MM</td> </tr> <tr> <td>X ±.1</td> <td>X ± 0.25</td> </tr> <tr> <td>XX ±.01</td> <td>X ± 0.1</td> </tr> <tr> <td>XXX ±.005</td> <td>XX ± 0.05</td> </tr> <tr> <td>XXXX ±.0005</td> <td>XXX ± 0.010</td> </tr> </table> ANGULO ±2° | INCH | MM | X ±.1 | X ± 0.25 | XX ±.01 | X ± 0.1 | XXX ±.005 | XX ± 0.05 | XXXX ±.0005 | XXX ± 0.010 | | | HOJA |
| INCH | MM | | | | | | | | | | | | |
| X ±.1 | X ± 0.25 | | | | | | | | | | | | |
| XX ±.01 | X ± 0.1 | | | | | | | | | | | | |
| XXX ±.005 | XX ± 0.05 | | | | | | | | | | | | |
| XXXX ±.0005 | XXX ± 0.010 | | | | | | | | | | | | |
| | | | 2:8 | | | | | | | | | | |



4X DETAIL A
SCALE 1 : 1

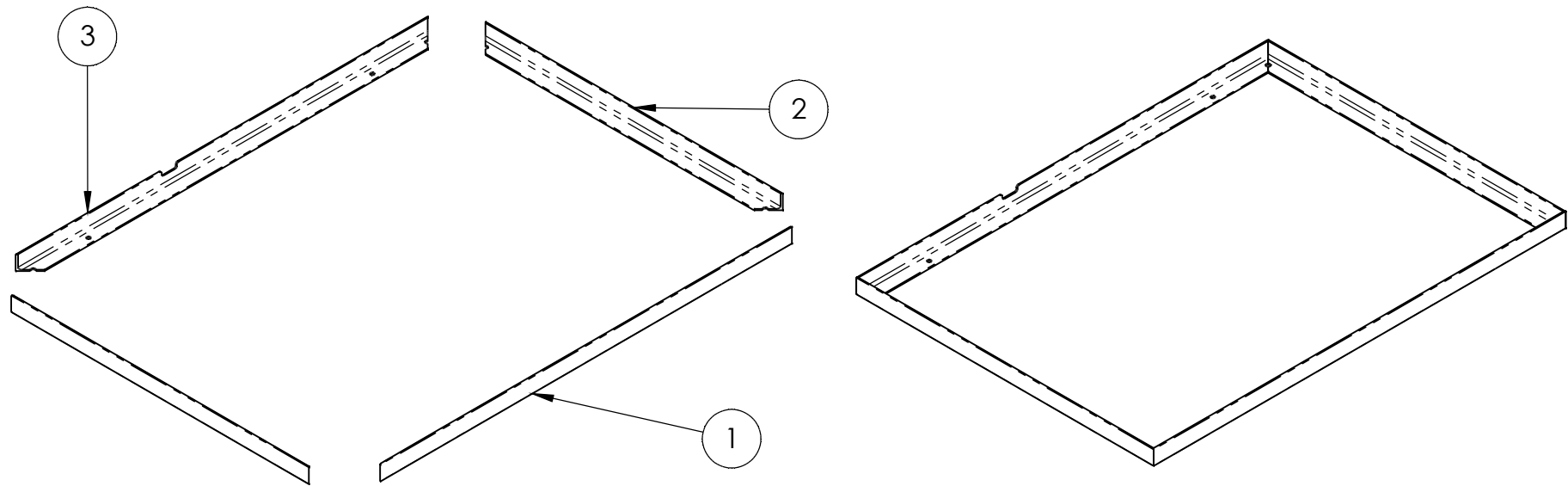
NOTAS:

- LAS UNIONES DEBEN SOLDARSE EN TODA LA PERIFERIA
- PERFORAR AGUJEROS LUEGO DE SOLDAR

| | | | |
|---|-------------------------------|----------------------|----------------------|
| Creativa. <small>RAAL INDUSTRIAL S.A.</small> | NÚMERO DE DOCUMENTO | MATERIAL: | ACABADO SUPERFICIAL: |
| | CI - 2609 | ANSI 1020 | |
| | TÍTULO: | TRATAMIENTO TÉRMICO: | SUPERFICIE: |
| | MARCO ESTRUCTURA | | |
| | CLIENTE: | DIBUJADO POR: | REV/VER |
| | | S. Rodríguez | A/A |
| | TERCER ANGLULO DE PROYECCION: | OP: | FECHA: |
| | | 169 | 19/08/22 |
| | | CANTIDAD: | REVISADO POR: |
| | | mm | Pendiente |
| | | | HOJA |
| | | | 3:8 |

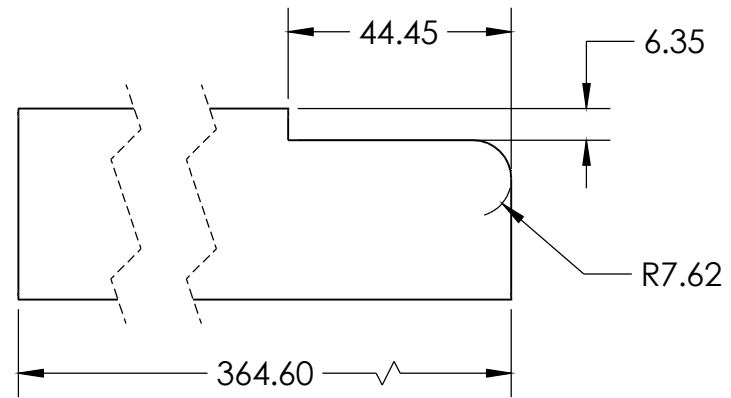
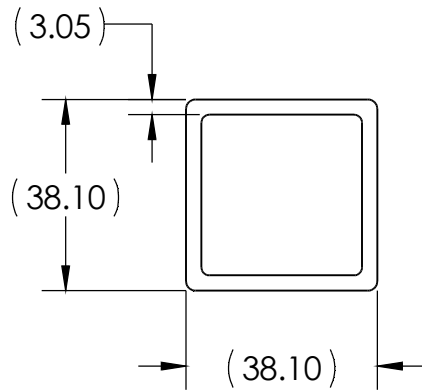
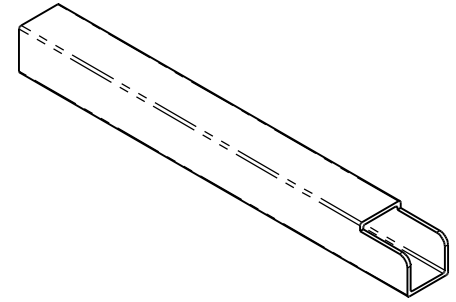
| TOLERANCIAS | |
|--------------------|--------------|
| DIMENSIONES EN MM. | |
| INCH | MM |
| X ±.1 | X ± 0.25 |
| .XX ±.01 | X ± 0.1 |
| .XXX ±.005 | .XX ± 0.05 |
| .XXXX ±.0005 | .XXX ± 0.010 |
| ANGULO ±2° | |

| ITEM NO. | PART NUMBER | DESCRIPTION | QTY. |
|----------|----------------|--------------------------|------|
| 1 | CI-2609-P005 | ANGULAR LARGO - SUPERIOR | 1 |
| 2 | CI-2609-P006 | ANGULAR CORTO - SUPERIOR | 2 |
| 3 | CI-2609-P005.2 | ANGULAR LARGO - SUPERIOR | 1 |



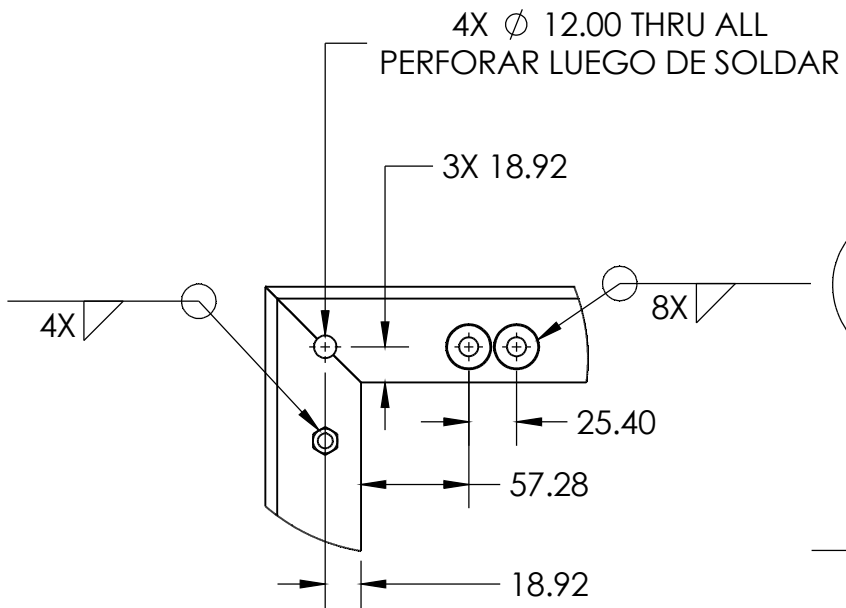
| | | | |
|---|-----------------------------|----------------------|----------------------|
| Creativa. <small>RAAL INDUSTRIAL S.A.</small> | NÚMERO DE DOCUMENTO | MATERIAL: | ACABADO SUPERFICIAL: |
| | CI-2609 | ANSI 1020 | |
| | TÍTULO: | TRATAMIENTO TÉRMICO: | SUPERFICIE: |
| | MARCO ESTRUCTURA | | |
| | CLIENTE: | DIBUJADO POR: | REV/VER |
| | | S. Rodríguez | A/A |
| | TERCER ÁNGULO DE PROYECCIÓN | OP: | FECHA: |
| | | 169 | 19/08/22 |
| | | CANTIDAD: | REVISADO POR: |
| | | mm | Pendiente |
| | | | HOJA |
| | | | 4:8 |

| TOLERANCIAS | |
|--------------------|-------------|
| DIMENSIONES EN MM. | |
| INCH | MM |
| X ±.1 | X ± 0.25 |
| XX ±.01 | X ± 0.1 |
| XXX ±.005 | XX ± 0.05 |
| XXXX ±.0005 | XXX ± 0.010 |
| ÁNGULO ±2° | |

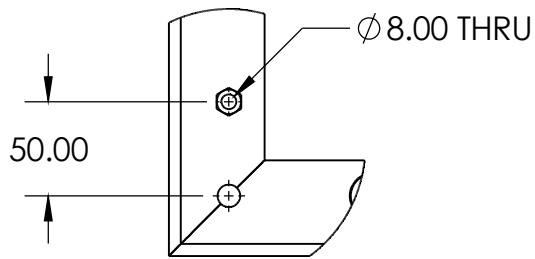


| | | | |
|---|-----------------------------|----------------------|----------------------|
| Creativa. <small>RAAL INDUSTRIAL S.A.</small> | NÚMERO DE DOCUMENTO | MATERIAL: | ACABADO SUPERFICIAL: |
| | CI-2609 | ANSI 1020 | |
| | TÍTULO: | TRATAMIENTO TÉRMICO: | SUPERFICIE: |
| | SOPORTE TRANSVERSAL | | |
| | CLIENTE: | DIBUJADO POR: | REV/VER |
| | | S. Rodríguez | A/A |
| | TERCER ÁNGULO DE PROYECCIÓN | OP: | FECHA: |
| | | 169 | 19/08/22 |
| | | CANTIDAD: | REVISADO POR: |
| | | mm | Pendiente |
| | | | HOJA |
| | | | 5:8 |

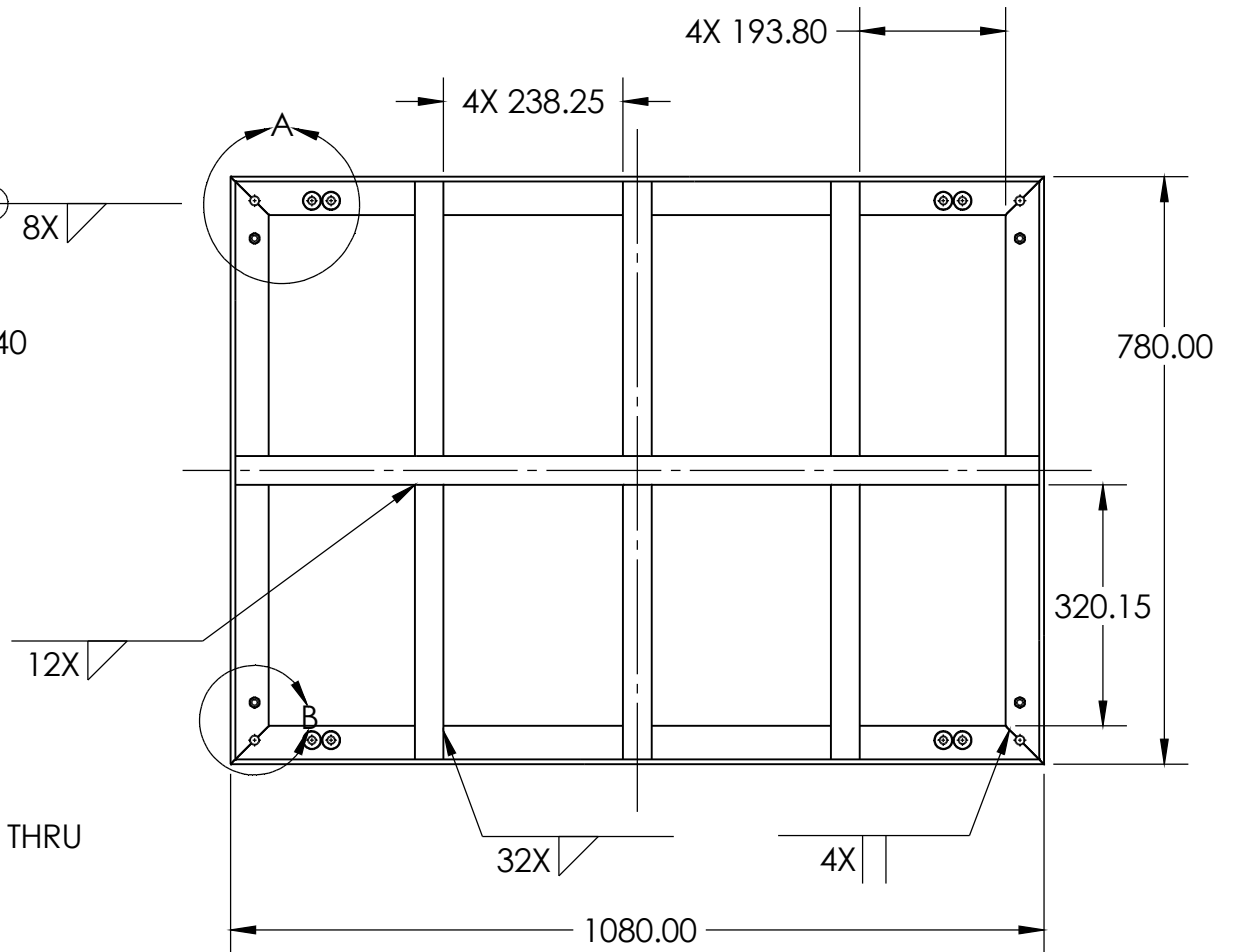
| TOLERANCIAS | |
|--------------------|--------------|
| DIMENSIONES EN MM. | |
| INCH | MM |
| X ±.1 | X ± 0.25 |
| .XX ±.01 | X ± 0.1 |
| .XXX ±.005 | .XX ± 0.05 |
| .XXXX ±.0005 | .XXX ± 0.010 |
| ANGULO ±2° | |



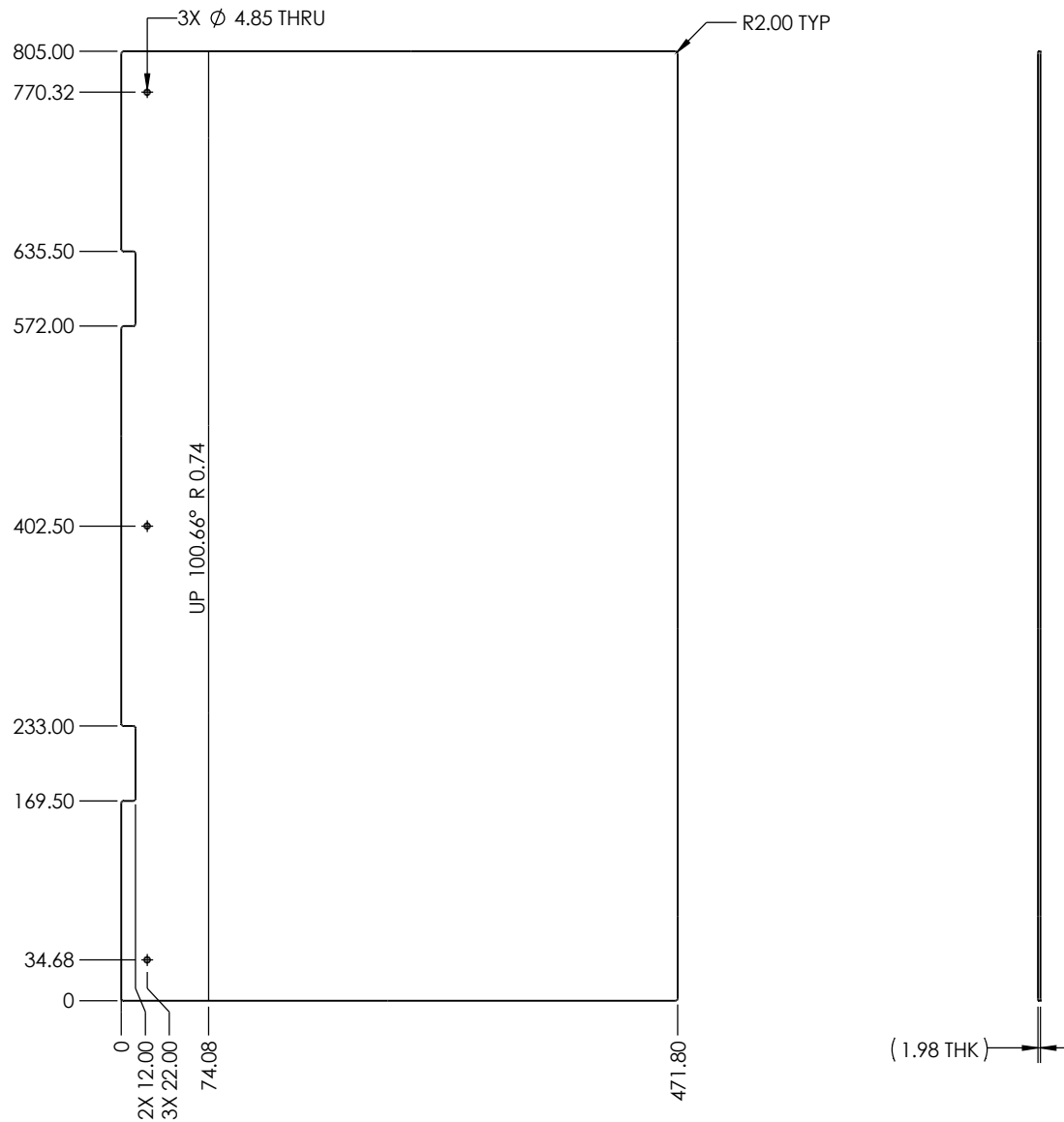
4X DETAIL A
SCALE 1 : 4



DETAIL B
SCALE 1 : 4

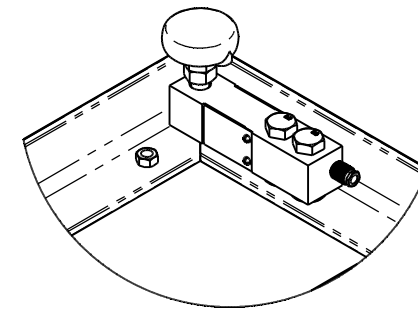
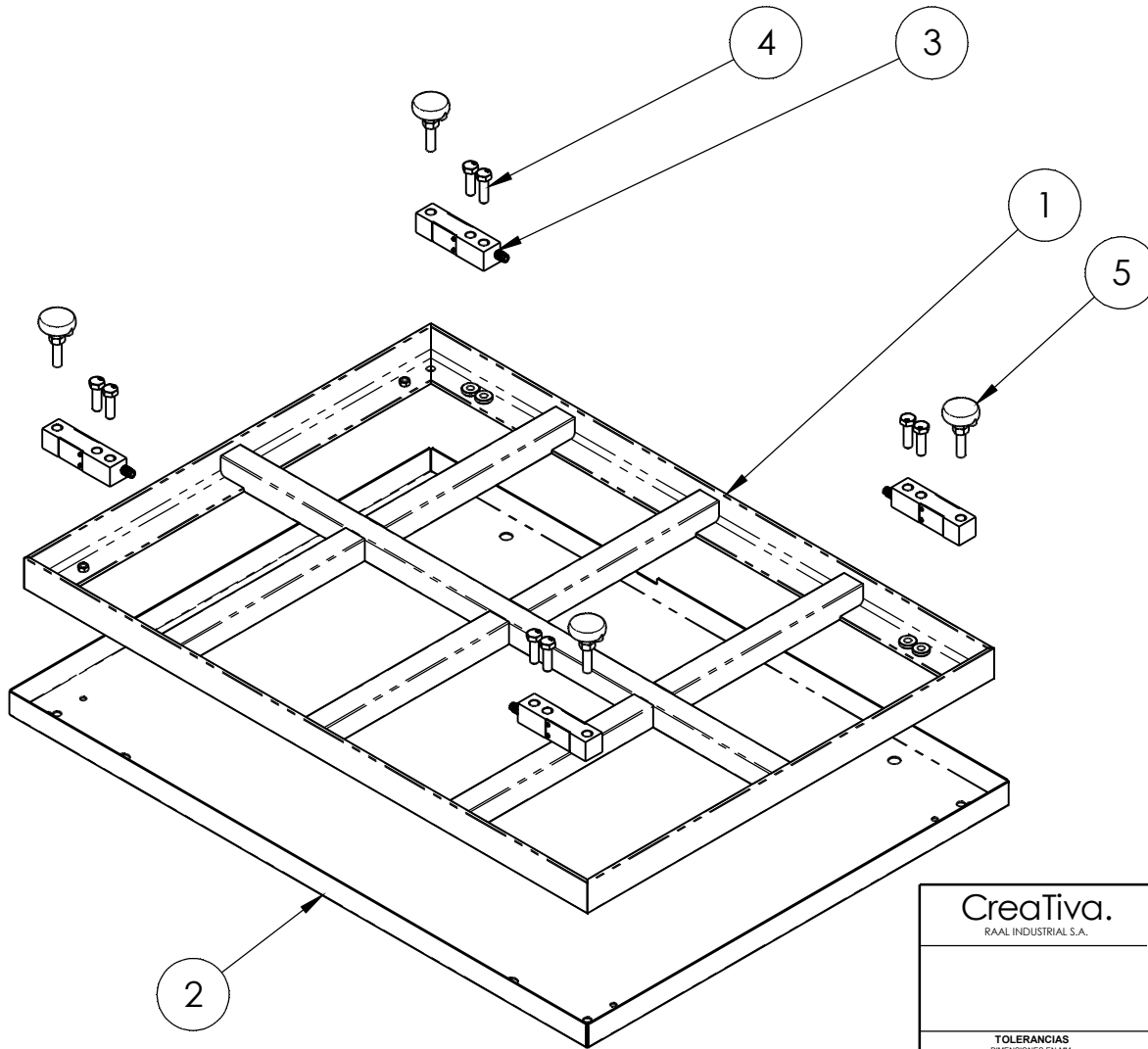
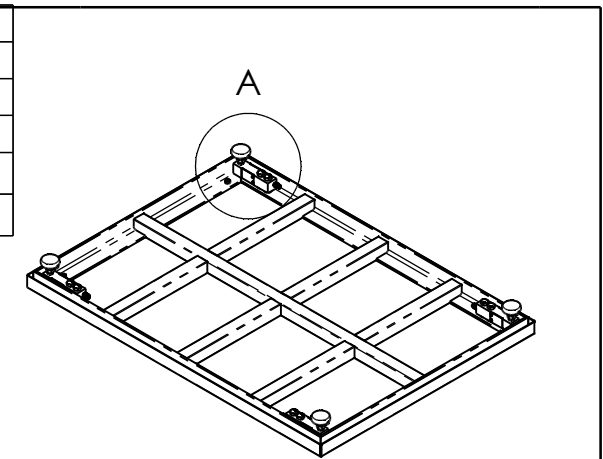


| Creativa. <small>RAAL INDUSTRIAL S.A.</small> | NÚMERO DE DOCUMENTO | MATERIAL: | ACABADO SUPERFICIAL: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|----------------------|----------------------|--------------------|--|------|--|----|--|---|-----|---|--------|----|------|---|-------|-----|-------|----|--------|------|--------|-----|---------|--|--------|--|-----|-----|-----------|---------------|
| | CI-2609 | ANSI 1020 | Pintado | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | TÍTULO: | TRATAMIENTO TÉRMICO: | SUPERFICIE: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | ESTRUCTURA | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | CLIENTE: | DIBUJADO POR: | REV/VER | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | S. Rodríguez | A/A | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | TERCER ANGLULO DE PROYECCION: | OP: | FECHA: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | <table border="1"> <thead> <tr> <th colspan="2">TOLERANCIAS</th> <th colspan="2">DIMENSIONES EN MM.</th> </tr> <tr> <th>INCH</th> <th></th> <th>MM</th> <th></th> </tr> </thead> <tbody> <tr> <td>X</td> <td>±.1</td> <td>X</td> <td>± 0.25</td> </tr> <tr> <td>XX</td> <td>±.01</td> <td>X</td> <td>± 0.1</td> </tr> <tr> <td>XXX</td> <td>±.005</td> <td>XX</td> <td>± 0.05</td> </tr> <tr> <td>XXXX</td> <td>±.0005</td> <td>XXX</td> <td>± 0.010</td> </tr> <tr> <td></td> <td>ANGULO</td> <td></td> <td>±2°</td> </tr> </tbody> </table> | TOLERANCIAS | | DIMENSIONES EN MM. | | INCH | | MM | | X | ±.1 | X | ± 0.25 | XX | ±.01 | X | ± 0.1 | XXX | ±.005 | XX | ± 0.05 | XXXX | ±.0005 | XXX | ± 0.010 | | ANGULO | | ±2° | 169 | CANTIDAD: | REVISADO POR: |
| TOLERANCIAS | | DIMENSIONES EN MM. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| INCH | | MM | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| X | ±.1 | X | ± 0.25 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| XX | ±.01 | X | ± 0.1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| XXX | ±.005 | XX | ± 0.05 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| XXXX | ±.0005 | XXX | ± 0.010 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | ANGULO | | ±2° | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | mm | Pendiente | HOJA | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | 6:8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |



| | | | | | | | | | | | | | | | | | | | | | |
|--|-----------|--------------------|----------------------|---|----|----|------|----|-----|-------|-----|------|--------|------|-------|---------|-------|--------|---|---|-----|
| CreaTiva. S.A. INDUSTRIAL S.A. | CI-2609 | MATERIA: ANSI 1020 | ACABADO SUPERFICIAL: | | | | | | | | | | | | | | | | | | |
| | | ESTADO TECNICO: | SUPERFICIE: | | | | | | | | | | | | | | | | | | |
| SOBRE-DESARROLLO | | | | | | | | | | | | | | | | | | | | | |
| TOLERANCIAS INDICADAS EN MM | | | S. Rodríguez | | | | | | | | | | | | | | | | | | |
| <table border="1"> <tr> <td>INCH</td> <td>±</td> <td>MM</td> </tr> <tr> <td>.X</td> <td>± .1</td> <td>.X</td> </tr> <tr> <td>.XX</td> <td>± .01</td> <td>.XX</td> </tr> <tr> <td>.XXX</td> <td>± .005</td> <td>.XXX</td> </tr> <tr> <td>.XXXX</td> <td>± .0005</td> <td>.XXXX</td> </tr> <tr> <td>ANGULO</td> <td>±</td> <td>°</td> </tr> </table> | | | INCH | ± | MM | .X | ± .1 | .X | .XX | ± .01 | .XX | .XXX | ± .005 | .XXX | .XXXX | ± .0005 | .XXXX | ANGULO | ± | ° | A/A |
| INCH | ± | MM | | | | | | | | | | | | | | | | | | | |
| .X | ± .1 | .X | | | | | | | | | | | | | | | | | | | |
| .XX | ± .01 | .XX | | | | | | | | | | | | | | | | | | | |
| .XXX | ± .005 | .XXX | | | | | | | | | | | | | | | | | | | |
| .XXXX | ± .0005 | .XXXX | | | | | | | | | | | | | | | | | | | |
| ANGULO | ± | ° | | | | | | | | | | | | | | | | | | | |
| OP: | CANTIDAD: | FECHA: | HOJA: | | | | | | | | | | | | | | | | | | |
| 169 | mm | 21/08/22 | 7:8 | | | | | | | | | | | | | | | | | | |
| REVISADO POR: | | | Pendiente | | | | | | | | | | | | | | | | | | |

| ITEM NO. | PART NUMBER | DESCRIPTION | QTY. |
|----------|-----------------|---------------------------------|------|
| 1 | CI-2609-A101 V2 | ESTRUCTURA | 1 |
| 2 | CI-2609-P009 | SOBRE | 1 |
| 3 | FTP75 | CELDA DE CARGA | 4 |
| 4 | HEX M12 X 40 | PERNO CABEZA HEX M12 | 8 |
| 5 | 6992K14 | BOLT-DOWN SWIVEL LEVELING MOUNT | 4 |



DETAIL A
SCALE 1 : 5

| | | | |
|---|-------------------------------|------------------|----------------------|
| Creativa. <small>RAAL INDUSTRIAL S.A.</small> | NÚMERO DE DOCUMENTO | MATERIAL: | ACABADO SUPERFICIAL: |
| | CI-2609 | ANSI 1020 | |
| | | AMIENTO TÉRMICO: | SUPERFICIE: |
| EXPLOSIONADO BÁSCULA | | | |
| | CLIENTE: | DIBUJADO POR: | A/A |
| | | S. Rodríguez | |
| | TERCER ANGLULO DE PROYECCION: | OP: | FECHA: |
| | | 169 | 19/08/22 |
| | CANTIDAD: | | REVISADO POR: |
| | mm | | Pendiente |
| | | | HOJA |
| | | | 8:8 |

| TOLERANCIAS | |
|--------------------|-------------|
| DIMENSIONES EN MM. | |
| INCH | MM |
| X ±.1 | X ± 0.25 |
| XX ±.01 | X ± 0.1 |
| XXX ±.005 | XX ± 0.05 |
| XXXX ±.0005 | XXX ± 0.010 |
| | ANGULO ±2° |

Anexo 2

WEIGHT SYSTEM PROTOTYPE


CreaTiva.

| UTILITY USAGE | RATINGS |
|---------------------|---------|
| ELECTRICAL | 120 VAC |
| COMPRESSED AIR | N/A |
| PLANT WATER | N/A |
| RECIRC. CHILL WATER | N/A |
| STEAM | N/A |
| NATURAL GAS | N/A |
| OTHER | N/A |

| SHEET # | DESCRIPTION OF SHEET |
|---------|-----------------------|
| 1 | FEATURES |
| 2 | ELECTRICAL COMPONENTS |
| 3 | POWER AND PLC OUTPUTS |
| 4 | ELECTRICAL CONEXION |
| 5 | ELECTRICAL CONEXION |
| 6 | NOMENCLATURE |

Designed by : CreaTiva Raal Industrial.
 Sabana Grande, Tejar el Guarco
 Cartago, Costa Rica
 Info@creativa.cr
 Teléfono: (506) 2573-3242
 Fax: (506) 2573-3838

CI -2609
 DT -169



| | | | |
|---|--|---|--|
| WWW.CREATIVACR / CARTAGO, COSTA RICA | |  | |
| STANDARD TOLERANCES UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES | | | |
| INCH ±.1 XX ±.05 XXX ±.005 | | DATE: 16/09/2022 | |
| X ±.125 XX ±.063 XXX ±.030 | | TITLE: COVER PAGE | |
| ANGLES ±2° | | REV BY: ING. IVAN MORA | |
| ADDITIONAL INFORMATION | | DOCUMENT NUMBER CI-2609 / DT-169 | |
| REVISIONS | | REV/VER 1.0 | |
| BY: | | SHEET 1 OF 6 | |
| | | DO NOT SCALE | |

ELECTRICAL COMPONENTS

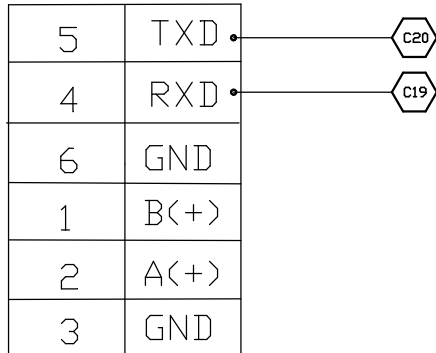
| TAG | QTY | PART NUMBER | DESCRIPTION | MANUFACTURER | LOCATION |
|----------------|-----|-------------------|---|---------------|----------|
| SW-01 | 1 | ZBE-102 | NO CONTACT BLOCK | ALLEN BRADLEY | GABINETE |
| BK-01 | 1 | WMSIC02 | BREAKER 1 POLOS 2 AMP | CUTLER HAMMER | GABINETE |
| FU-01 | 1 | 1492H6 | FUSE ENCLOSURE TO 2A FUSE | ALLEN BRADLEY | GABINETE |
| PS-01 | 1 | 2080-PSI20-240VAC | POWER SUPPLY 100-240VAC/24VDC | OMRON | GABINETE |
| PLC-01 | 1 | 2080LC20-20QBB | CONTROLADOR MICRO 820, 12 ENTRADAS 24VDC/AC, 8 SALIDAS A | ALLEN BRADLEY | GABINETE |
| TR-01 | 1 | SCT20-NA | WEIGHT TRANSMISOR SERIAL COMMUNICATION | SCT | GABINETE |
| SN-01/02/03/04 | 4 | FTP | EXTENSOMETRIC LOAD CELL | LAUMAS | OUTSIDE |
| LT-01 | 1 | SPT5-T-D | LIGHT INDICATOR LAMP | YXQ | OUTSIDE |
| - | 1 | TFFN18VE | Green Cable 18 awg | VIACON | GABINETE |
| - | 1 | TFFN18NE | BLACK CABLE 18 AWG | VIACON | GABINETE |
| - | 1 | 1170157 | ORANGE CABLE 18 AWG | TOPFLEX | GABINETE |
| - | 1 | 4520051 | WHITE CABLE 18 AWG | LAPPKABEL | GABINETE |

NOTE:

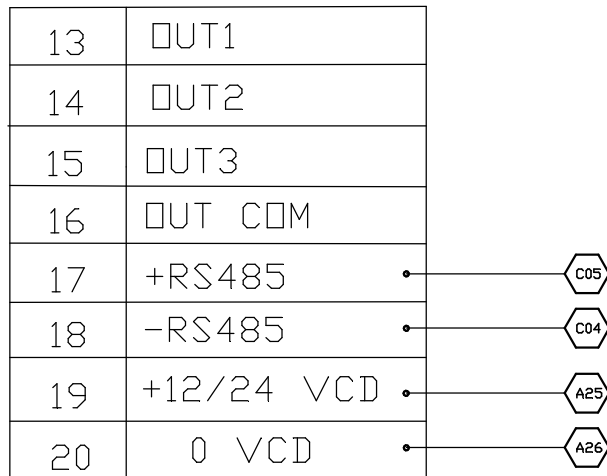
1. NAME LISTED IN DRAWN BY FIELD IS EMPLOYEE OF CREATIVA RAAL INDUSTRIAL SA

| | | | |
|--|------------|---|---|
| WWW.CREATIVACR / CARTAGO, COSTA RICA | |  | |
| STANDARD TOLERANCES UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES. | | | |
| INCH | FR | TITLE | |
| X ±.1 | X ±.125 | ELECTRICAL COMPONENTS | |
| .XX ±.01 | .X ±.01 | | |
| .XXX ±.005 | .XX ±.005 | | |
| .XXX ±.005 | .XXX ±.000 | | |
| ANGLES ±2° | | REV BY: ING. IVAN MORA | ENG. SIZE |
| ADDITIONAL INFORMATION | | C |  THIRD ANGLE PROJECTION |
| NORMAL CHANGES | | | |
| | | CI-2609 / 0T-169 | 1.0 |
| | | SHEET | 2 OF 6 |

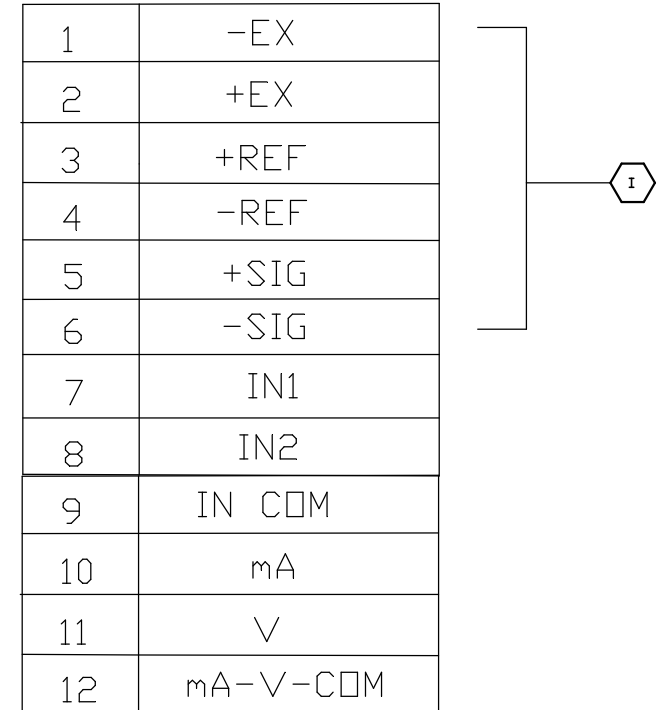
C
PLC EMBELBED SERIAL PORT
RS485-RS242



TRANSMISOR 8 PIN



I
TRANSMISOR 12 PIN

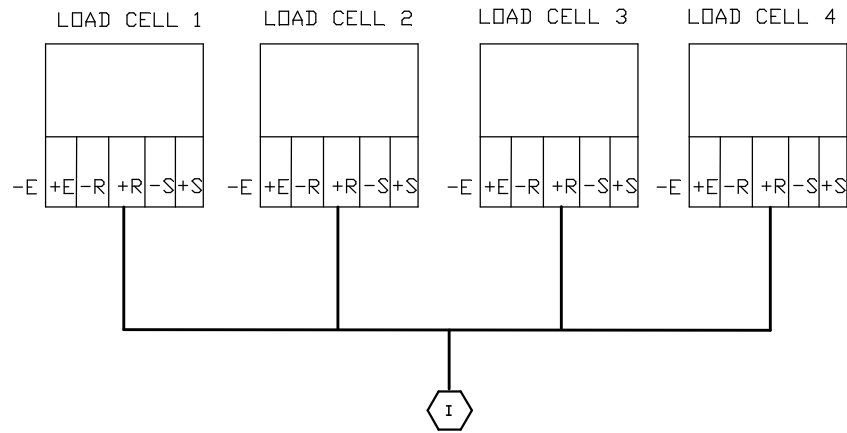


NOTE:

1. NAME LISTED IN DRAWN BY FIELD IS EMPLOYEE OF CREATIVA RAAL INDUSTRIAL S.A

| | | | | | | | | | | | | | | | | | | | |
|---|--------|--|--------|---|-------|----|------|---|------|-----|-------|----|-------|------|--------|-----|--------|-------------------------------------|--|
| WWW.CREATIVA.CR / CARTAGO, COSTA RICA | | Creativa. | | | | | | | | | | | | | | | | | |
| STANDARD TOLERANCES UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES: | | DRAWN BY: S. RODRIGUEZ | | | | | | | | | | | | | | | | | |
| <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 10%;">X</td><td style="width: 10%;">±0.1</td><td style="width: 10%;">X</td><td style="width: 10%;">±0.25</td></tr> <tr><td>XX</td><td>±0.1</td><td>X</td><td>±0.1</td></tr> <tr><td>XXX</td><td>±0.05</td><td>XX</td><td>±0.05</td></tr> <tr><td>XXXX</td><td>±0.025</td><td>XXX</td><td>±0.025</td></tr> </table> | | X | ±0.1 | X | ±0.25 | XX | ±0.1 | X | ±0.1 | XXX | ±0.05 | XX | ±0.05 | XXXX | ±0.025 | XXX | ±0.025 | DATE: 16/09/2022 | |
| X | ±0.1 | X | ±0.25 | | | | | | | | | | | | | | | | |
| XX | ±0.1 | X | ±0.1 | | | | | | | | | | | | | | | | |
| XXX | ±0.05 | XX | ±0.05 | | | | | | | | | | | | | | | | |
| XXXX | ±0.025 | XXX | ±0.025 | | | | | | | | | | | | | | | | |
| <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 10%;">X</td><td style="width: 10%;">±0.1</td><td style="width: 10%;">X</td><td style="width: 10%;">±0.25</td></tr> <tr><td>XX</td><td>±0.1</td><td>X</td><td>±0.1</td></tr> <tr><td>XXX</td><td>±0.05</td><td>XX</td><td>±0.05</td></tr> <tr><td>XXXX</td><td>±0.025</td><td>XXX</td><td>±0.025</td></tr> </table> | | X | ±0.1 | X | ±0.25 | XX | ±0.1 | X | ±0.1 | XXX | ±0.05 | XX | ±0.05 | XXXX | ±0.025 | XXX | ±0.025 | TITLE ELECTRICAL CONEXION | |
| X | ±0.1 | X | ±0.25 | | | | | | | | | | | | | | | | |
| XX | ±0.1 | X | ±0.1 | | | | | | | | | | | | | | | | |
| XXX | ±0.05 | XX | ±0.05 | | | | | | | | | | | | | | | | |
| XXXX | ±0.025 | XXX | ±0.025 | | | | | | | | | | | | | | | | |
| ADDITIONAL INFORMATION | | REV BY: ING. IVAN MORA | | | | | | | | | | | | | | | | | |
| MINOR CHANGES | | DO NOT SCALE | | | | | | | | | | | | | | | | | |
| SHEET 4 OF 6 | | DOCUMENT NUMBER CI-2609 / DT-169 | | | | | | | | | | | | | | | | | |
| REV/VER 1.0 | | DO NOT SCALE | | | | | | | | | | | | | | | | | |





C



NOTA: LAS 6 SEÑALES DE LAS 4 CELDAS DE CARGA SE CONECTAN EN PARALELO HACIA EL CORRESPONDIENTE BORNE DEL TRANSMISOR

NOTE:

1. NAME LISTED IN DRAWN BY FIELD IS EMPLOYEE OF CREATIVA RAAL INDUSTRIAL S.A

| | | | | | | | | | | | | | |
|--|-----------------------------------|--|------|---|-----------------------------------|--------------|--------------|---------|---------|-------|---------|------------------|--|
| WWW.CREATIVACR / CARTAGO, COSTA RICA | |  | | | | | | | | | | | |
| STANDARD TOLERANCES UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES | | | | DRAWN BY: S. RODRIGUEZ | | | | | | | | | |
| <table border="1"> <tr> <td>INCH</td> <td>MM</td> </tr> <tr> <td>.1</td> <td>± .025</td> </tr> <tr> <td>.01</td> <td>± .001</td> </tr> <tr> <td>.001</td> <td>± .0005</td> </tr> <tr> <td>.0005</td> <td>± .0002</td> </tr> </table> | | INCH | MM | .1 | ± .025 | .01 | ± .001 | .001 | ± .0005 | .0005 | ± .0002 | DATE: 16/09/2022 | |
| INCH | MM | | | | | | | | | | | | |
| .1 | ± .025 | | | | | | | | | | | | |
| .01 | ± .001 | | | | | | | | | | | | |
| .001 | ± .0005 | | | | | | | | | | | | |
| .0005 | ± .0002 | | | | | | | | | | | | |
| <table border="1"> <tr> <td>ANGLES</td> <td>± 2°</td> </tr> </table> | | ANGLES | ± 2° | TITLE: ELECTRICAL CONEXION | | | | | | | | | |
| ANGLES | ± 2° | | | | | | | | | | | | |
| ADDITIONAL INFORMATION | | REV BY: ING. IVAN MORA | | | | | | | | | | | |
| MINOR CHANGES | | <table border="1"> <tr> <td>  THIRD ANGLE PROJECTION </td> <td>DOCUMENT NUMBER: CI-2609 / DT-169</td> <td>REV/VER: 1.0</td> </tr> <tr> <td>DO NOT SCALE</td> <td>SHEET 5</td> <td>OF 6</td> </tr> </table> | |  THIRD ANGLE PROJECTION | DOCUMENT NUMBER: CI-2609 / DT-169 | REV/VER: 1.0 | DO NOT SCALE | SHEET 5 | OF 6 | | | | |
|  THIRD ANGLE PROJECTION | DOCUMENT NUMBER: CI-2609 / DT-169 | REV/VER: 1.0 | | | | | | | | | | | |
| DO NOT SCALE | SHEET 5 | OF 6 | | | | | | | | | | | |

SYMBOL TABLE

| Electrical Symbol | Description |
|-------------------|---------------------------|
| | FUSE, PROTECTIVE |
| | PLUG AND SOCKET |
| | SOLENOID COIL |
| | SELECTOR SWITCH |
| | INDICATING LIGHT |
| | PROXIMITY SWITCH |
| | PUSH BUTTON N.O. |
| | PUSH BUTTON N.C. |
| | PUSH BUTTON MUSHROOM HEAD |
| | LIQUID LEVEL SWITCH |
| | FLOW SWITCH |
| | PRESSURE SWITCH |
| | TEMPERATURE SWITCH |
| | LIMIT SWITCH |
| | CONTROL RELAY COIL |
| | CONTACTOR COIL |
| | MOTOR STARTER COIL |
| | TIMER COIL |
| | ELECTROMECHANICAL COUNTER |
| | CIRCUIT BREAKER |
| | TERMINALS (REFERENCE) |
| | CONTACTOR CONTACT CLOSE |

NOTE:

1. NAME LISTED IN DRAWN BY FIELD IS EMPLOYEE OF CREATIVA RAAL INDUSTRIAL S.A

SYMBOL TABLE

| | |
|--|--------------------------------------|
| | CONTACTOR CONTACT OPEN |
| | RELAY CONTACT CLOSED |
| | RELAY CONTACT OPEN |
| | TIMED CONTACT, N.O. -ON DELAY (TDE) |
| | TIMED CONTACT, N.C. -ON DELAY (TDE) |
| | TIMED CONTACT, N.C. -OFF DELAY (TDD) |
| | TIMED CONTACT, N.O. -OFF DELAY (TDD) |
| | SELECTOR SWITCH |

| STANDARD TOLERANCES UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES. | | DRAWN BY: S. RODRIGUEZ DATE: 16/09/2022 | | |
|--|------------|--|------------------|--|
| INCH | MM | TITLE | REV BY: | |
| X ±.1 | X ±.25 | ELECTRIC NOMENCLATURE | ING. IVAN NEIRA | |
| XX ±.01 | XX ±.11 | | | |
| XXX ±.005 | XXX ±.015 | | | |
| XXXX ±.0025 | XXXX ±.008 | | | |
| ANGLES ± 2° | | REV BY: | ING. IVAN NEIRA | |
| ADDITIONAL INFORMATION | | DWG. SIZE | C | |
| MANUAL DIMENSIONS | | THIRD ANGLE PROJECTION | | |
| | | DOCUMENT NUMBER | CI-2609 / DT-169 | |
| | | REV/VER | 1.0 | |
| | | SHEET | 6 OF 6 | |

Anexo 3

Código ASCII: Tabla de códigos imprimibles y no imprimibles.

Los caracteres ASCII son aquellos que por cuestiones de configuración e idioma no pueden tipearse directamente, por lo tanto se encuentran codificados:

Ejemplo: para el carácter ñ debe mantenerse apretada la tecla **alt** y tipear **164**.

Mantener apretada la tecla "ALT" y tipear el número azul, para crear el código buscado.

| | | | | | | | | | |
|----|----|-----|---|-----|----|-----|---|-----|----|
| 32 | | 77 | M | 122 | z | 167 | o | 212 | ḷ |
| 33 | ! | 78 | N | 123 | { | 168 | ı | 213 | ƒ |
| 34 | " | 79 | O | 124 | | 169 | ¬ | 214 | π |
| 35 | # | 80 | P | 125 | } | 170 | ¬ | 215 | ‡ |
| 36 | \$ | 81 | Q | 126 | ~ | 171 | ½ | 216 | ‡ |
| 37 | % | 82 | R | 127 | - | 172 | ¼ | 217 | ↓ |
| 38 | & | 83 | S | 128 | Ç | 173 | i | 218 | Γ |
| 39 | ' | 84 | T | 129 | ü | 174 | « | 219 | ■ |
| 40 | (| 85 | U | 130 | é | 175 | » | 220 | ■ |
| 41 |) | 86 | V | 131 | â | 176 | ▒ | 221 | ▒ |
| 42 | * | 87 | W | 132 | ä | 177 | ▒ | 222 | ▒ |
| 43 | + | 88 | X | 133 | à | 178 | ▒ | 223 | ■ |
| 44 | , | 89 | Y | 134 | ã | 179 | | 224 | α |
| 45 | - | 90 | Z | 135 | ç | 180 | † | 225 | β |
| 46 | . | 91 | [| 136 | ê | 181 | ‡ | 226 | Γ |
| 47 | / | 92 | | 137 | ë | 182 | ‡ | 227 | π |
| 48 | 0 | 93 |] | 138 | è | 183 | π | 228 | Σ |
| 49 | 1 | 94 | ^ | 139 | ï | 184 | ¶ | 229 | σ |
| 50 | 2 | 95 | _ | 140 | î | 185 | ‡ | 230 | μ |
| 51 | 3 | 96 | ` | 141 | ì | 186 | | 231 | τ |
| 52 | 4 | 97 | a | 142 | Ä | 187 | ¶ | 232 | Φ |
| 53 | 5 | 98 | b | 143 | Å | 188 | ¶ | 233 | θ |
| 54 | 6 | 99 | c | 144 | É | 189 | ¶ | 234 | Ω |
| 55 | 7 | 100 | d | 145 | æ | 190 | ¶ | 235 | δ |
| 56 | 8 | 101 | e | 146 | Æ | 191 | ¶ | 236 | ∞ |
| 57 | 9 | 102 | f | 147 | ô | 192 | L | 237 | Ø |
| 58 | : | 103 | g | 148 | ö | 193 | ⊥ | 238 | ε |
| 59 | ; | 104 | h | 149 | ò | 194 | T | 239 | ∩ |
| 60 | < | 105 | i | 150 | û | 195 | † | 240 | ≡ |
| 61 | = | 106 | j | 151 | ù | 196 | - | 241 | ± |
| 62 | > | 107 | k | 152 | ÿ | 197 | † | 242 | ≥ |
| 63 | ? | 108 | l | 153 | Ö | 198 | † | 243 | ≤ |
| 64 | @ | 109 | m | 154 | Ü | 199 | ‡ | 244 | |
| 65 | A | 110 | n | 155 | ç | 200 | ℒ | 245 | |
| 66 | B | 111 | o | 156 | £ | 201 | ƒ | 246 | ÷ |
| 67 | C | 112 | p | 157 | ¥ | 202 | ≡ | 247 | ≈ |
| 68 | D | 113 | q | 158 | Pt | 203 | π | 248 | ° |
| 69 | E | 114 | r | 159 | f | 204 | ‡ | 249 | · |
| 70 | F | 115 | s | 160 | á | 205 | = | 250 | · |
| 71 | G | 116 | t | 161 | í | 206 | ‡ | 251 | √ |
| 72 | H | 117 | u | 162 | ó | 207 | ≡ | 252 | n |
| 73 | I | 118 | v | 163 | ú | 208 | ⊥ | 253 | ² |
| 74 | J | 119 | w | 164 | ñ | 209 | π | 254 | ■ |
| 75 | K | 120 | x | 165 | Ñ | 210 | π | | |
| 76 | L | 121 | y | 166 | a | 211 | ℒ | | |

Anexo 4

| PESOS MEDIDOS (Kg) | Media(Kg) | Desv Est (Kg) |
|--------------------|---|---------------|
| 38.497 | 38.513 | 0.023 |
| 38.533 | | |
| 38.507 | <div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: fit-content;"> Intervalo de confianza </div> $\bar{x} \pm z_{\alpha/2} \frac{s}{\sqrt{n}}$ | |
| 38.502 | | |
| 38.531 | | |
| 38.482 | | |
| 38.525 | | |
| 38.45 | | |
| 38.525 | | |
| 38.522 | | |
| 38.446 | | |
| 38.51 | | |
| 38.545 | | |
| 38.522 | | |
| 38.517 | | |
| 38.524 | | |
| 38.497 | | |
| 38.526 | | |
| 38.483 | | |
| 38.53 | | |
| 38.541 | | |
| 38.501 | | |
| 38.531 | | |
| 38.535 | | |
| 38.515 | | |
| 38.512 | | |
| 38.502 | | |
| 38.545 | | |
| 38.512 | | |
| 38.523 | | |
| 38.518 | | |
| 38.521 | | |
| 38.485 | | |
| 38.518 | | |
| 38.532 | | |

Anexo 5

Sample Rate Value Units
10 Milliseconds
Capture Size 1000
Trigger Trigger Type Allow Overflow
Manual True

| | |
|---------|--------|
| T Real | T Resp |
| 00:05.6 | 0.6 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 45:38.6 | 12.81 | 45:44.1 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.6 | 12.81 | 45:44.2 12.81 |
| 45:38.7 | 12.81 | 45:44.2 12.81 |
| 45:38.7 | 12.81 | 45:44.2 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.7 | 12.81 | 45:44.3 0 |
| 45:38.8 | 12.81 | 45:44.3 0 |
| 45:38.8 | 12.81 | 45:44.3 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.8 | 12.81 | 45:44.4 0 |
| 45:38.9 | 12.81 | 45:44.4 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:38.9 | 12.81 | 45:44.5 0 |
| 45:39.0 | 12.81 | 45:44.5 0 |
| 45:39.0 | 12.81 | 45:44.5 0 |

Sample Rate Value Units
 10 Milliseconds
 Capture Size 1000
 Trigger Trigger Type Allow Overflow
 Manual True

| | |
|---------|--------|
| T Real | T Resp |
| 00:06.0 | 1 |

| Time | Micro820 Port : 3811-PESO | Final de los datos | |
|---------|---------------------------|--------------------|-------|
| 53:03.1 | 2.817 | 53:08.9 | 2.817 |
| 53:03.1 | 2.817 | 53:09.0 | 2.817 |
| 53:03.1 | 2.817 | 53:09.0 | 2.817 |
| 53:03.1 | 2.817 | 53:09.0 | 2.817 |
| 53:03.1 | 2.817 | 53:09.0 | 2.817 |
| 53:03.1 | 2.817 | 53:09.0 | 2.817 |
| 53:03.1 | 2.817 | 53:09.0 | 2.817 |
| 53:03.1 | 2.817 | 53:09.0 | 2.817 |
| 53:03.1 | 2.817 | 53:09.0 | 2.817 |
| 53:03.2 | 2.817 | 53:09.0 | 2.817 |
| 53:03.2 | 2.817 | 53:09.0 | 2.817 |
| 53:03.2 | 2.817 | 53:09.1 | 2.817 |
| 53:03.2 | 2.817 | 53:09.1 | 2.817 |
| 53:03.2 | 2.817 | 53:09.1 | 2.817 |
| 53:03.2 | 2.817 | 53:09.1 | 0 |
| 53:03.2 | 2.817 | 53:09.1 | 0 |
| 53:03.2 | 2.817 | 53:09.1 | 0 |
| 53:03.2 | 2.817 | 53:09.1 | 0 |
| 53:03.2 | 2.817 | 53:09.1 | 0 |
| 53:03.3 | 2.817 | 53:09.1 | 0 |
| 53:03.3 | 2.817 | 53:09.1 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.3 | 2.817 | 53:09.2 | 0 |
| 53:03.4 | 2.817 | 53:09.2 | 0 |
| 53:03.4 | 2.817 | 53:09.2 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.4 | 2.817 | 53:09.3 | 0 |
| 53:03.5 | 2.817 | 53:09.3 | 0 |
| 53:03.5 | 2.817 | 53:09.3 | 0 |

Sample Rate Value Units
 10 Milliseconds
 Capture Size 1000
 Trigger Trigger Type Allow Overflow
 Manual True

| | |
|---------|--------|
| T Real | T Resp |
| 00:07.0 | 2 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 53:41.9 | 2.816 | 53:48.8 2.816 |
| 53:41.9 | 2.816 | 53:48.8 2.816 |
| 53:41.9 | 2.816 | 53:48.8 2.816 |
| 53:41.9 | 2.816 | 53:48.8 2.816 |
| 53:41.9 | 2.816 | 53:48.8 2.816 |
| 53:41.9 | 2.816 | 53:48.8 2.816 |
| 53:41.9 | 2.816 | 53:48.8 2.816 |
| 53:41.9 | 2.816 | 53:48.8 2.816 |
| 53:41.9 | 2.816 | 53:48.9 2.816 |
| 53:41.9 | 2.816 | 53:48.9 2.816 |
| 53:42.0 | 2.816 | 53:48.9 0.001 |
| 53:42.0 | 2.816 | 53:48.9 0.001 |
| 53:42.0 | 2.816 | 53:48.9 0.001 |
| 53:42.0 | 2.816 | 53:48.9 0.001 |
| 53:42.0 | 2.816 | 53:48.9 0.001 |
| 53:42.0 | 2.816 | 53:48.9 0.001 |
| 53:42.0 | 2.816 | 53:48.9 0.001 |
| 53:42.0 | 2.816 | 53:48.9 0.001 |
| 53:42.0 | 2.816 | 53:49.0 0.001 |
| 53:42.0 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.0 0.001 |
| 53:42.1 | 2.816 | 53:49.1 0.001 |
| 53:42.1 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.1 0.001 |
| 53:42.2 | 2.816 | 53:49.2 0.001 |
| 53:42.2 | 2.816 | 53:49.2 0.001 |
| 53:42.3 | 2.816 | 53:49.2 0.001 |
| 53:42.3 | 2.816 | 53:49.2 0.001 |

Sample Rate Value Units
 10 Milliseconds
 Capture Size 1000
 Trigger Trigger Type Allow Overflow
 Manual True

| | |
|---------|--------|
| T Real | T Resp |
| 00:06.1 | 1.1 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 54:28.8 | 2.817 | 54:34.6 2.817 |
| 54:28.8 | 2.817 | 54:34.6 2.817 |
| 54:28.8 | 2.817 | 54:34.7 2.817 |
| 54:28.8 | 2.817 | 54:34.7 2.817 |
| 54:28.8 | 2.817 | 54:34.7 2.817 |
| 54:28.8 | 2.817 | 54:34.7 2.817 |
| 54:28.8 | 2.817 | 54:34.7 2.817 |
| 54:28.8 | 2.817 | 54:34.7 2.817 |
| 54:28.8 | 2.817 | 54:34.7 2.817 |
| 54:28.8 | 2.817 | 54:34.7 2.817 |
| 54:28.9 | 2.817 | 54:34.7 2.817 |
| 54:28.9 | 2.817 | 54:34.7 2.817 |
| 54:28.9 | 2.817 | 54:34.7 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:28.9 | 2.817 | 54:34.8 2.817 |
| 54:29.0 | 2.817 | 54:34.8 2.817 |
| 54:29.0 | 2.817 | 54:34.8 2.817 |
| 54:29.0 | 2.817 | 54:34.8 2.817 |
| 54:29.0 | 2.817 | 54:34.9 2.817 |
| 54:29.0 | 2.817 | 54:34.9 2.817 |
| 54:29.0 | 2.817 | 54:34.9 2.817 |
| 54:29.0 | 2.817 | 54:34.9 2.817 |
| 54:29.0 | 2.817 | 54:34.9 2.817 |
| 54:29.0 | 2.817 | 54:34.9 0.001 |
| 54:29.0 | 2.817 | 54:34.9 0.001 |
| 54:29.1 | 2.817 | 54:34.9 0.001 |
| 54:29.1 | 2.817 | 54:34.9 0.001 |
| 54:29.1 | 2.817 | 54:34.9 0.001 |
| 54:29.1 | 2.817 | 54:35.0 0.001 |
| 54:29.1 | 2.817 | 54:35.0 0.001 |
| 54:29.1 | 2.817 | 54:35.0 0.001 |
| 54:29.1 | 2.817 | 54:35.0 0.001 |
| 54:29.1 | 2.817 | 54:35.0 0.001 |
| 54:29.1 | 2.817 | 54:35.0 0.001 |
| 54:29.1 | 2.817 | 54:35.0 0.001 |
| 54:29.1 | 2.817 | 54:35.0 0.001 |
| 54:29.2 | 2.817 | 54:35.0 0.001 |
| 54:29.2 | 2.817 | 54:35.0 0.001 |

Sample Rate Value Units
10 Milliseconds
Capture Size 1000
Trigger Trigger Type Allow Overflow
Manual True

| | |
|---------|--------|
| T Real | T Resp |
| 00:06.7 | 1.7 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 55:54.3 | 5.52 | 56:00.9 5.52 |
| 55:54.3 | 5.52 | 56:00.9 5.52 |
| 55:54.3 | 5.52 | 56:00.9 5.52 |
| 55:54.3 | 5.52 | 56:00.9 5.52 |
| 55:54.3 | 5.52 | 56:00.9 5.52 |
| 55:54.3 | 5.52 | 56:00.9 5.52 |
| 55:54.3 | 5.52 | 56:00.9 5.52 |
| 55:54.3 | 5.52 | 56:01.0 5.52 |
| 55:54.3 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.0 5.52 |
| 55:54.4 | 5.52 | 56:01.1 0.001 |
| 55:54.4 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.1 0.001 |
| 55:54.5 | 5.52 | 56:01.2 0.001 |
| 55:54.5 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.2 0.001 |
| 55:54.6 | 5.52 | 56:01.3 0.001 |
| 55:54.6 | 5.52 | 56:01.3 0.001 |
| 55:54.7 | 5.52 | 56:01.3 0.001 |
| 55:54.7 | 5.52 | 56:01.3 0.001 |

Sample Rate Value Units
 10 Milliseconds
 Capture Size 1000
 Trigger Trigger Type Allow Overflow
 Manual True

| | |
|---------|--------|
| T Real | T Resp |
| 00:06.7 | 1.7 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 58:00.1 | 5.52 | 58:06.7 5.52 |
| 58:00.1 | 5.52 | 58:06.7 5.52 |
| 58:00.1 | 5.52 | 58:06.7 5.52 |
| 58:00.1 | 5.52 | 58:06.7 5.52 |
| 58:00.1 | 5.52 | 58:06.8 5.52 |
| 58:00.1 | 5.52 | 58:06.8 5.52 |
| 58:00.1 | 5.52 | 58:06.8 5.52 |
| 58:00.1 | 5.52 | 58:06.8 5.52 |
| 58:00.2 | 5.52 | 58:06.8 5.52 |
| 58:00.2 | 5.52 | 58:06.8 5.52 |
| 58:00.2 | 5.52 | 58:06.8 5.52 |
| 58:00.2 | 5.52 | 58:06.8 5.52 |
| 58:00.2 | 5.52 | 58:06.8 5.52 |
| 58:00.2 | 5.52 | 58:06.8 5.52 |
| 58:00.2 | 5.52 | 58:06.8 5.52 |
| 58:00.2 | 5.52 | 58:06.9 0 |
| 58:00.2 | 5.52 | 58:06.9 0 |
| 58:00.2 | 5.52 | 58:06.9 0 |
| 58:00.2 | 5.52 | 58:06.9 0 |
| 58:00.3 | 5.52 | 58:06.9 0 |
| 58:00.3 | 5.52 | 58:06.9 0 |
| 58:00.3 | 5.52 | 58:06.9 0 |
| 58:00.3 | 5.52 | 58:06.9 0 |
| 58:00.3 | 5.52 | 58:06.9 0 |
| 58:00.3 | 5.52 | 58:06.9 0 |
| 58:00.3 | 5.52 | 58:07.0 0 |
| 58:00.3 | 5.52 | 58:07.0 0 |
| 58:00.3 | 5.52 | 58:07.0 0 |
| 58:00.3 | 5.52 | 58:07.0 0 |
| 58:00.4 | 5.52 | 58:07.0 0 |
| 58:00.4 | 5.52 | 58:07.0 0 |
| 58:00.4 | 5.52 | 58:07.0 0 |
| 58:00.4 | 5.52 | 58:07.0 0 |
| 58:00.4 | 5.52 | 58:07.0 0 |
| 58:00.4 | 5.52 | 58:07.0 0 |
| 58:00.4 | 5.52 | 58:07.0 0 |
| 58:00.4 | 5.52 | 58:07.1 0 |
| 58:00.4 | 5.52 | 58:07.1 0 |
| 58:00.4 | 5.52 | 58:07.1 0 |
| 58:00.4 | 5.52 | 58:07.1 0 |
| 58:00.5 | 5.52 | 58:07.1 0 |
| 58:00.5 | 5.52 | 58:07.1 0 |
| 58:00.5 | 5.52 | 58:07.1 0 |

Sample Rate Value Units
10 Milliseconds
Capture Size 1000
Trigger Trigger Type Allow Overflow
Manual True
Time Micro820 Port : 3811-PESO

| | |
|---------|--------|
| T Real | T Resp |
| 00:06.2 | 1.2 |

| Time | Value | Units | Final de los datos | Value |
|---------|-------|-------|--------------------|-------|
| 59:39.1 | 5.52 | | 59:45.0 | 5.52 |
| 59:39.1 | 5.52 | | 59:45.0 | 5.52 |
| 59:39.1 | 5.52 | | 59:45.0 | 5.52 |
| 59:39.1 | 5.52 | | 59:45.0 | 5.52 |
| 59:39.1 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.1 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.1 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.1 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.1 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.2 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.2 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.3 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.3 | 5.52 |
| 59:39.3 | 5.52 | | 59:45.3 | 5.52 |
| 59:39.4 | 5.52 | | 59:45.0 | 5.52 |
| 59:39.4 | 5.52 | | 59:45.3 | 0 |
| 59:39.4 | 5.52 | | 59:45.3 | 0 |
| 59:39.4 | 5.52 | | 59:45.3 | 0 |
| 59:39.4 | 5.52 | | 59:45.3 | 0 |
| 59:39.4 | 5.52 | | 59:45.3 | 0 |
| 59:39.4 | 5.52 | | 59:45.3 | 0 |
| 59:39.4 | 5.52 | | 59:45.4 | 0 |
| 59:39.4 | 5.52 | | 59:45.4 | 0 |
| 59:39.4 | 5.52 | | 59:45.4 | 0 |
| 59:39.5 | 5.52 | | 59:45.4 | 0 |
| 59:39.5 | 5.52 | | 59:45.4 | 0 |
| 59:39.5 | 5.52 | | 59:45.4 | 0 |

Anexo 6

Sample Rate Value Units
 10 Milliseconds
 Capture Size 1000
 Trigger Trigger Type Allow Overflow
 Manual True

| | |
|-------------|--------|
| Tiempo Real | T Resp |
| 00:06.7 | 1.7 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 20:28.0 | 11.002 | 20:34.5 11.002 |
| 20:28.0 | 11.002 | 20:34.5 11.002 |
| 20:28.0 | 11.002 | 20:34.5 11.002 |
| 20:28.0 | 11.002 | 20:34.5 11.002 |
| 20:28.0 | 11.002 | 20:34.5 11.002 |
| 20:28.0 | 11.002 | 20:34.5 11.002 |
| 20:28.0 | 11.002 | 20:34.5 11.002 |
| 20:28.0 | 11.002 | 20:34.5 11.002 |
| 20:28.0 | 11.002 | 20:34.6 11.002 |
| 20:28.0 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.6 11.002 |
| 20:28.1 | 11.002 | 20:34.7 11.002 |
| 20:28.1 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.7 0 |
| 20:28.2 | 11.002 | 20:34.8 0 |
| 20:28.2 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.8 0 |
| 20:28.3 | 11.002 | 20:34.9 0 |
| 20:28.3 | 11.002 | 20:34.9 0 |
| 20:28.4 | 11.002 | 20:34.9 0 |

Sample Rate Value Units
10 Milliseconds
Capture Size 1000
Trigger Trigger Type Allow Overflow
Manual True

| | |
|-------------|--------|
| Tiempo Real | T Resp |
| 00:06.8 | 1.8 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 21:36.2 | 11.002 | 21:42.7 11.002 |
| 21:36.2 | 11.002 | 21:42.7 11.002 |
| 21:36.2 | 11.002 | 21:42.7 11.002 |
| 21:36.2 | 11.002 | 21:42.7 11.002 |
| 21:36.2 | 11.002 | 21:42.8 11.002 |
| 21:36.2 | 11.002 | 21:42.8 11.002 |
| 21:36.2 | 11.002 | 21:42.8 11.002 |
| 21:36.2 | 11.002 | 21:42.8 11.002 |
| 21:36.2 | 11.002 | 21:42.8 11.002 |
| 21:36.3 | 11.002 | 21:42.8 11.002 |
| 21:36.3 | 11.002 | 21:42.8 11.002 |
| 21:36.3 | 11.002 | 21:42.8 11.002 |
| 21:36.3 | 11.002 | 21:42.8 11.002 |
| 21:36.3 | 11.002 | 21:42.8 11.002 |
| 21:36.3 | 11.002 | 21:42.8 11.002 |
| 21:36.3 | 11.002 | 21:42.9 11.002 |
| 21:36.3 | 11.002 | 21:42.9 11.002 |
| 21:36.3 | 11.002 | 21:42.9 11.002 |
| 21:36.3 | 11.002 | 21:42.9 11.002 |
| 21:36.3 | 11.002 | 21:42.9 11.002 |
| 21:36.3 | 11.002 | 21:42.9 11.002 |
| 21:36.3 | 11.002 | 21:42.9 11.002 |
| 21:36.3 | 11.002 | 21:42.9 11.002 |
| 21:36.4 | 11.002 | 21:42.9 11.002 |
| 21:36.4 | 11.002 | 21:42.9 11.002 |
| 21:36.4 | 11.002 | 21:42.9 11.002 |
| 21:36.4 | 11.002 | 21:42.9 11.002 |
| 21:36.4 | 11.002 | 21:42.9 11.002 |
| 21:36.4 | 11.002 | 21:42.9 11.002 |
| 21:36.4 | 11.002 | 21:43.0 11.002 |
| 21:36.4 | 11.002 | 21:43.0 0.002 |
| 21:36.4 | 11.002 | 21:43.0 0.002 |
| 21:36.4 | 11.002 | 21:43.0 0.002 |
| 21:36.4 | 11.002 | 21:43.0 0.002 |
| 21:36.5 | 11.002 | 21:43.0 0.002 |
| 21:36.5 | 11.002 | 21:43.0 0.002 |
| 21:36.5 | 11.002 | 21:43.0 0.002 |
| 21:36.5 | 11.002 | 21:43.0 0.002 |
| 21:36.5 | 11.002 | 21:43.0 0.002 |
| 21:36.5 | 11.002 | 21:43.1 0.002 |
| 21:36.5 | 11.002 | 21:43.1 0.002 |
| 21:36.5 | 11.002 | 21:43.1 0.002 |
| 21:36.5 | 11.002 | 21:43.1 0.002 |
| 21:36.5 | 11.002 | 21:43.1 0.002 |
| 21:36.6 | 11.002 | 21:43.1 0.002 |
| 21:36.6 | 11.002 | 21:43.1 0.002 |

Sample Rate Value Units
 10 Milliseconds
 Capture Size 1000
 Trigger Trigger Type Allow Overflow
 Manual True

| | |
|-------------|--------|
| Tiempo Real | T Resp |
| 00:06.6 | 1.6 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 23:01.6 | 11 | 23:08.1 11.005 |
| 23:01.6 | 11 | 23:08.1 11.005 |
| 23:01.6 | 11 | 23:08.1 11.005 |
| 23:01.6 | 11 | 23:08.1 11.005 |
| 23:01.6 | 11 | 23:08.1 11.005 |
| 23:01.6 | 11 | 23:08.1 11.005 |
| 23:01.6 | 11 | 23:08.1 11.005 |
| 23:01.6 | 11 | 23:08.1 11.005 |
| 23:01.6 | 11 | 23:08.2 11.005 |
| 23:01.6 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.2 0.002 |
| 23:01.7 | 11 | 23:08.3 0.002 |
| 23:01.7 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.3 0.002 |
| 23:01.8 | 11 | 23:08.4 0.002 |
| 23:01.8 | 11 | 23:08.4 0.002 |
| 23:01.9 | 11 | 23:08.4 0.002 |
| 23:01.9 | 11 | 23:08.4 0.002 |
| 23:01.9 | 11 | 23:08.4 0.002 |
| 23:01.9 | 11 | 23:08.4 0.002 |
| 23:01.9 | 11 | 23:08.4 0.002 |
| 23:01.9 | 11.005 | 23:08.4 0.002 |
| 23:01.9 | 11.005 | 23:08.4 0.002 |
| 23:01.9 | 11.005 | 23:08.5 0.002 |
| 23:01.9 | 11.005 | 23:08.5 0.002 |
| 23:02.0 | 11.005 | 23:08.5 0.002 |

Sample Rate Value Units
 10 Milliseconds
 Capture Size 1000
 Trigger Trigger Type Allow Overflow
 Manual True

| | |
|-------------|--------|
| Tiempo Real | T Resp |
| 00:06.5 | 1.5 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 30:11.4 | 2.813 | 30:17.8 2.813 |
| 30:11.4 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.8 2.813 |
| 30:11.5 | 2.813 | 30:17.9 2.813 |
| 30:11.5 | 2.813 | 30:17.9 2.813 |
| 30:11.5 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:17.9 2.813 |
| 30:11.6 | 2.813 | 30:18.0 2.813 |
| 30:11.6 | 2.813 | 30:18.0 2.813 |
| 30:11.6 | 2.813 | 30:18.0 2.813 |
| 30:11.7 | 2.813 | 30:18.0 0.002 |
| 30:11.7 | 2.813 | 30:18.0 0.002 |
| 30:11.7 | 2.813 | 30:18.0 0.002 |
| 30:11.7 | 2.813 | 30:18.0 0.002 |
| 30:11.7 | 2.813 | 30:18.0 0.002 |
| 30:11.7 | 2.813 | 30:18.0 0.002 |
| 30:11.7 | 2.813 | 30:18.0 0.002 |
| 30:11.7 | 2.813 | 30:18.1 0.002 |
| 30:11.7 | 2.813 | 30:18.1 0.002 |
| 30:11.7 | 2.813 | 30:18.1 0.002 |
| 30:11.8 | 2.813 | 30:18.1 0.002 |
| 30:11.8 | 2.813 | 30:18.1 0.002 |
| 30:11.8 | 2.813 | 30:18.1 0.002 |
| 30:11.8 | 2.813 | 30:18.1 0.002 |
| 30:11.8 | 2.813 | 30:18.1 0.002 |
| 30:11.8 | 2.813 | 30:18.1 0.002 |
| 30:11.8 | 2.813 | 30:18.1 0.002 |
| 30:11.8 | 2.813 | 30:18.2 0.002 |
| 30:11.8 | 2.813 | 30:18.2 0.002 |

Sample Rate Value Units
 10 Milliseconds
 Capture Size 1000
 Trigger Trigger Type Allow Overflow
 Manual True

| | |
|-------------|--------|
| Tiempo Real | T Resp |
| 00:06.7 | 1.7 |

| Time | Micro820 Port : 3811-PESO | Final de los datos |
|---------|---------------------------|--------------------|
| 30:38.6 | 2.815 | 30:45.0 2.815 |
| 30:38.6 | 2.815 | 30:45.0 2.815 |
| 30:38.6 | 2.815 | 30:45.0 2.815 |
| 30:38.6 | 2.815 | 30:45.0 2.815 |
| 30:38.6 | 2.815 | 30:45.1 2.815 |
| 30:38.6 | 2.815 | 30:45.1 2.815 |
| 30:38.6 | 2.815 | 30:45.1 2.815 |
| 30:38.6 | 2.815 | 30:45.1 2.815 |
| 30:38.6 | 2.815 | 30:45.1 2.815 |
| 30:38.6 | 2.815 | 30:45.1 2.815 |
| 30:38.6 | 2.815 | 30:45.1 2.815 |
| 30:38.7 | 2.815 | 30:45.1 2.815 |
| 30:38.7 | 2.815 | 30:45.1 2.815 |
| 30:38.7 | 2.815 | 30:45.1 2.815 |
| 30:38.7 | 2.815 | 30:45.1 2.815 |
| 30:38.7 | 2.815 | 30:45.1 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.7 | 2.815 | 30:45.2 2.815 |
| 30:38.8 | 2.815 | 30:45.2 2.815 |
| 30:38.8 | 2.815 | 30:45.2 2.815 |
| 30:38.8 | 2.815 | 30:45.2 0 |
| 30:38.8 | 2.815 | 30:45.3 0 |
| 30:38.8 | 2.815 | 30:45.3 0 |
| 30:38.8 | 2.815 | 30:45.3 0 |
| 30:38.8 | 2.815 | 30:45.3 0 |
| 30:38.8 | 2.815 | 30:45.3 0 |
| 30:38.8 | 2.815 | 30:45.3 0 |
| 30:38.8 | 2.815 | 30:45.3 0 |
| 30:38.8 | 2.815 | 30:45.3 0 |
| 30:38.9 | 2.815 | 30:45.3 0 |
| 30:38.9 | 2.815 | 30:45.3 0 |
| 30:38.9 | 2.815 | 30:45.3 0 |
| 30:38.9 | 2.815 | 30:45.3 0 |
| 30:38.9 | 2.815 | 30:45.4 0 |
| 30:38.9 | 2.815 | 30:45.4 0 |
| 30:38.9 | 2.815 | 30:45.4 0 |
| 30:38.9 | 2.815 | 30:45.4 0 |
| 30:38.9 | 2.815 | 30:45.4 0 |
| 30:38.9 | 2.815 | 30:45.4 0 |
| 30:38.9 | 2.815 | 30:45.4 0 |
| 30:39.0 | 2.815 | 30:45.4 0 |

1 Create.cs

```
1 using Backend.Database.Contract;
2 using MediatR;
3 using Microsoft.EntityFrameworkCore.Infrastructure;
4 using System.Data.Common;
5
6 namespace Backend.BusinessLogic.Semaforo
7 {
8     public class Create
9     {
10         //Creacion de la clase del objeto que se envia a la Base de datos con herencia de
11         //la interfase de BasaculaInf
12         public class CreateSemaforo : IRequest<Backend.Model.BasaculaInf>
13         {
14             public int Action { get; set; } // Action es para escoger cual luz se quiere
15             //escribir (0,1 o 2)
16             public int Turn { get; set; } // Turn es para encender la luz que se escogio
17             //con Action
18             public decimal Size { get; set; } // Size es el Peso registrado
19             public int Stable { get; set; } // Stable es el bit de peso estable en la
20             //plataforma
21         }
22
23         public class Mediator : IRequestHandler<CreateSemaforo, Backend.Model.BasaculaInf>
24         {
25             //Estructura mediadora para encapsular las peticiones
26
27             private IDbContext _dbContext { get; }
28             private IWriteConnection _writeDbConnection { get; }
29             public Mediator(IDbContext dbContext, IWriteConnection writeDbConnection)
30             {
31                 _dbContext = dbContext;
32                 _writeDbConnection = writeDbConnection;
33             }
34
35             public async Task<Model.BasaculaInf> Handle(CreateSemaforo request,
36                 CancellationToken cancellationToken)
37             {
38                 _dbContext.Connection.Open();
39                 using (var transaction = _dbContext.Database.BeginTransaction())
40                 {
41                     try
42                     {
43                         //Estructura de una peticion
44                         var query = $"INSERT INTO [dbo].[BasaculaInf]([Action],[Turn],[
45                             Size],[Stable]) VALUES(" +request.Action+", "+request.Turn+", "
46                             +request.Size+", "+request.Stable+"");
47                         Model.BasaculaInf result = await _writeDbConnection.
48                             QueryFirstOrDefaultAsync<Model.BasaculaInf>(query, new Model.
49                             BasaculaInf { }, transaction: ((IInfrastructure<DbTransaction>
50                             >)_dbContext.Database.CurrentTransaction).Instance);
51
52                         //Commmit a la base de datos
53                         transaction.Commit();
54
55                         return result;
56                     }
57                     catch (Exception ex)
58                     {
59                         //
60                     }
61                 }
62             }
63         }
64     }
65 }
```



```

38         var resultado = await _readDbConnection.QuerySingleAsync<Model.
           BasaculaInf>("SELECT * FROM [BD_Bascula].[dbo].[BasaculaInf]
           WHERE [BD_Bascula].[dbo].[BasaculaInf].[Id] = (SELECT MAX ([
           Id]) FROM [BD_Bascula].[dbo].[BasaculaInf])", new Model.
           BasaculaInf { }, cancellationToken: cancellationToken);
39         return resultado;
40     }
41     catch (Exception ex)
42     {
43         throw;
44     }
45     finally
46     {
47         connection.Close();
48     }
49 }
50 }
51 }
52 }
53 }

```

3 BasaculaInf.cs

```

1 namespace Backend.Model
2 {
3     public class BasaculaInf //Modelo u objeto que se envia a la base de datos
4     {
5         public int Id { get; set; }
6         public int Action { get; set; }
7         public bool Turn { get; set; }
8         public decimal Size { get; set; }
9         public bool Stable { get; set; }
10    }
11 }

```

4 SemaforoController.cs

```

1 using Backend.BusinessLogic.Semaforo;
2 using Backend.Database.Contract;
3 using Backend.Model;
4 using MediatR;
5 using Microsoft.AspNetCore.Mvc;
6 using Microsoft.EntityFrameworkCore;
7
8 namespace Backend.Controllers
9 {
10    [ApiController] //Controladores para modificar la Base de Datos con los
           contrato de servicio API
11    [Route("[controller]")]
12    public class SemaforoController : ControllerBase
13    {
14        private IDbContext _dbContext { get; }
15        private IMediator _mediator;
16        public SemaforoController(IDbContext dbContext, IMediator mediator)
17        {

```

```

18         _dbContext = dbContext;
19         _mediator = mediator;
20     }
21     [HttpPost("insertar")] //
22     // Instancia el Endpoint 'post' del API
23     public async Task<BasaculaInf> TurnLightAsync(Create.CreateSemaforo request) //
24     // Retorna un objeto BasaculaInf & recibe la peticion CREATE
25     {
26         return await _mediator.Send(request); //Se hace el request para insertar
27         // informacion encapsulada con mediador
28     }
29
30     [HttpGet("obtener")] //
31     // Instancia el Endpoint 'get' del API
32     public async Task<BasaculaInf> Get() //Extrae el objeto BasaculaInf
33     {
34         return await _mediator.Send(new Find.FindSemaforo()); //Usa la Peticion
35         // FIND
36     }
37 }

```

5 DbContext.cs

```

1     using Backend.Database.Contract;
2     using Backend.Model;
3     using Microsoft.EntityFrameworkCore;
4     using Microsoft.EntityFrameworkCore.Infrastructure;
5     using System.Data;
6
7     namespace Backend.Database
8     {
9         //Contrato 1: contrato de conexion con la BD
10        // Se tiene herencia de un Framework para crear la estructura del proyecto y usar
11        // ciertos comandos
12        public class DbContext : Microsoft.EntityFrameworkCore.DbContext, IDbContext
13        {
14            public DbContext(DbContextOptions<DbContext> options) : base(options)
15            {
16            }
17            public IDbConnection Connection => Database.GetDbConnection();
18            public DbSet<BasaculaInf> semaforo { get; set; }
19        }
20    }
21 }

```

6 IDbContext.cs

```

1     using Backend.Model;
2     using Microsoft.EntityFrameworkCore;
3     using Microsoft.EntityFrameworkCore.Infrastructure;

```

```

4 using System.Data;
5
6 namespace Backend.Database.Contract
7 {
8     //Interfase de las funciones de conexion con la BD
9     public interface IDbContext
10    {
11        public IDbConnection Connection { get; }
12        DatabaseFacade Database { get; }
13        public DbSet<BasaculaInf> semaforo { get; set; }
14        Task<int> SaveChangesAsync(CancellationTokent cancellationToken);
15    }
16 }
17 }

```

7 ReadConnection.cs

```

1 using Backend.Database.Contract;
2 using Dapper;
3 using Microsoft.Data.SqlClient;
4 using System.Data;
5
6 namespace Backend.Database
7 {
8
9     //Contrato 2: Contrato de lectura a la BD
10    public class ReadConnection : IReadConnection, IDisposable
11    {
12        private readonly IDbConnection connection;
13        public ReadConnection(IConfiguration configuration)
14        {
15            this.connection = new SqlConnection(configuration.GetConnectionString("
16                DefaultConnection"));
17        }
18        public void Dispose()
19        {
20            connection.Dispose();
21        }
22
23        //Funciones de lectura de BD
24
25        public async Task<List<T>> QueryAsync<T>(string sql, object param = null,
26            IDbTransaction transaction = null, CancellationTokent cancellationToken =
27            default)
28        {
29            return (await connection.QueryAsync<T>(sql, param, transaction)).AsList();
30        }
31
32        public async Task<T> QueryFirstOrDefaultAsync<T>(string sql, object param = null,
33            IDbTransaction transaction = null, CancellationTokent cancellationToken =
34            default)
35        {
36            return (await connection.QueryFirstAsync<T>(sql, param, transaction));
37        }
38    }
39 }

```

```

35     public async Task<T> QuerySingleAsync<T>(string sql, object param = null,
36         IDbTransaction transaction = null, CancellationToken cancellationToken =
37         default)
38     {
39         return await connection.QuerySingleAsync<T>(sql, param, transaction);
40     }

```

8 IReadConnection.cs

```

1     using System.Data;
2
3     namespace Backend.Database.Contract
4     {
5
6         //Interfase de las funciones de lectura utilizadas
7
8         public interface IReadConnection
9         {
10             Task<List<T>> QueryAsync<T>(string sql, object param = null, IDbTransaction
11                 transaction = null, CancellationToken cancellationToken = default);
12             Task<T> QueryFirstOrDefaultAsync<T>(string sql, object param = null,
13                 IDbTransaction transaction = null, CancellationToken cancellationToken =
14                 default);
15             Task<T> QuerySingleAsync<T>(string sql, object param = null, IDbTransaction
16                 transaction = null, CancellationToken cancellationToken = default);
17         }
18     }

```

9 WriteConnection.cs

```

1     using Backend.Database.Contract;
2     using Dapper;
3     using System.Data;
4
5     namespace Backend.Database
6     {
7
8         //Contrato 3: Escritura a la BD
9         public class WriteDbConnection : IWriteConnection
10        {
11            private readonly IDbContext context;
12            public WriteDbConnection(IDbContext context)
13            {
14                this.context = context;
15            }
16
17            //Funciones de escritura
18            public async Task<int> ExecuteAsync(string sql, object param = null,
19                IDbTransaction transaction = null, CancellationToken cancellationToken =
20                default)

```

```

21         return await context.Connection.ExecuteAsync(sql, param, transaction);
22     }
23
24     public async Task<List<T>> QueryAsync<T>(string sql, object param = null,
25         IDbTransaction transaction = null, CancellationToken cancellationToken =
26         default)
27     {
28         return (await context.Connection.QueryAsync<T>(sql, param, transaction)).
29             AsList();
30     }
31
32     public async Task<T> QueryFirstOrDefaultAsync<T>(string sql, object param = null,
33         IDbTransaction transaction = null, CancellationToken cancellationToken =
34         default)
35     {
36         return await context.Connection.QueryFirstOrDefaultAsync<T>(sql, param,
37             transaction);
38     }
39
40     public async Task<T> QuerySingleAsync<T>(string sql, object param = null,
41         IDbTransaction transaction = null, CancellationToken cancellationToken =
42         default)
43     {
44         return await context.Connection.QuerySingleAsync<T>(sql, param, transaction);
45     }
46 }

```

10 IWriteConnection.cs

```

1     using System.Data;
2
3     namespace Backend.Database.Contract
4     {
5
6         //Interfase de la funcion de escritura utilizada
7         public interface IWriteConnection : IReadConnection
8         {
9             Task<int> ExecuteAsync(string sql, object param = null, IDbTransaction
10                 transaction = null, CancellationToken cancellationToken = default);
11     }

```

11 Program.cs

```

1     using Backend.BusinessLogic.Semaforo;
2     using Backend.Database;
3     using Backend.Database.Contract;
4     using MediatR;
5     using Microsoft.EntityFrameworkCore;
6
7     var builder = WebApplication.CreateBuilder(args);
8
9     // Servicios para levantar el servidor
10

```

```

11 builder.Services.AddControllers();
12 // Learn more about configuring Swagger/OpenAPI at https://aka.ms/aspnetcore/swashbuckle
13 builder.Services.AddEndpointsApiExplorer();
14 builder.Services.AddSwaggerGen();
15 builder.Services.AddDbContext<Backend.Database.DbContext>(options =>
16 {
17     options.UseSqlServer(builder.Configuration.GetConnectionString("DefaultConnection"));
18 });
19 builder.Services.AddScoped<IDbContext>(provider => provider.GetService<Backend.Database.
    DbContext>());
20 builder.Services.AddScoped<IWriteConnection, WriteDbConnection>();
21 builder.Services.AddScoped<IReadConnection, ReadConnection>();
22
23 builder.Services.AddMediatR(typeof(Create.Mediator).Assembly);
24
25 var app = builder.Build();
26
27 // Configuracin de la peticin HTTP
28 if (app.Environment.IsDevelopment())
29 {
30     app.UseSwagger();
31     app.UseSwaggerUI();
32 }
33
34 app.UseCors(x => x
35     .AllowAnyOrigin()
36     .AllowAnyMethod()
37     .AllowAnyHeader());
38
39 app.UseHttpsRedirection();
40
41 app.UseAuthorization();
42
43 app.MapControllers();
44
45 app.Run();

```

12 appsettings.json

```

1 {
2     "ConnectionStrings": {
3         "DefaultConnection": "Data Source=DESKTOP-9VEEUKF\\TEW_SQLEXPRESS;Initial Catalog=
        BD_Bascula;Integrated Security=True"
4     },
5     "Logging": {
6         "LogLevel": {
7             "Default": "Information",
8             "Microsoft.AspNetCore": "Warning"
9         }
10    },
11    "AllowedHosts": "*"
12 }

```

13 Form1.cs

```

1 using System.Text;

```

```

2  using Newtonsoft.Json;
3  using System;
4  using System.Collections.Generic;
5  using System.Linq;
6  using System.Net.Http;
7  using System.Threading.Tasks;
8  using System.Windows;
9  using EasyModbus;
10 using Backend.Model;
11 using System.Security.Principal;
12
13 namespace Interfase_FrontEnd
14 {
15
16     public partial class Form1 : Form
17     {
18
19         public static ModbusClient Wago = new ModbusClient(); // Creando el Cliente
                Modbus
20         public int Id_Dispositivo ;
21         public int ID_Instruccion ;
22         public int InstruccionRecibida;
23         public decimal PesoRecolectado;
24         public bool bStableRecolectado;
25         public HttpClient client = new HttpClient(); //Instancia el cliente de la
                base de datos
26         public Bascula item = new Bascula(); //Crear objeto JSON a serializar
27         public GetBascula infoGet = new GetBascula();
28
29
30
31         public Form1()
32         {
33             InitializeComponent();
34         }
35
36         public class Bascula //Creacin de Modelo json para POST
37         {
38
39             public int action { get; set; }
40             public int turn { get; set; }
41             public decimal size { get; set; }
42             public bool stable { get; set; }
43         }
44
45         public class GetBascula //Creacin de Modelo json para GET
46         {
47
48             public int id { get; set; }
49             public int action { get; set; }
50             public bool turn { get; set; }
51             public int size { get; set; }
52             public bool stable { get; set; }
53         }
54
55         public static bool AbrirConexionWago() //Funcin para abrir conexin de
                comunicacin Modbus con el Hardware

```

```

56     {
57         bool estatus_conexion = false;
58
59         try
60         {
61             Wago.Connect();
62             estatus_conexion = true;
63         }
64         catch (Exception ex)
65         {
66             throw new Exception("[WAGO]: " + ex.Message, ex);
67         }
68
69         return estatus_conexion;
70     }
71
72     public static bool CerrarConexionWago() //Funcin para cerrar conexin de
73     comunicacin Modbus con el Hardware
74     {
75         bool estatus_conexion = false;
76
77         try
78         {
79             if (Wago.Connected)
80             {
81                 Wago.Disconnect();
82                 estatus_conexion = false;
83             }
84         }
85         catch (Exception ex)
86         {
87             throw new Exception("[WAGO]: " + ex.Message, ex);
88         }
89
90         return estatus_conexion;
91     }
92
93     private void EnciendeAmarillo()
94     {
95         Wago.WriteMultipleCoils(0, new bool[] { false, true, false }); //Escribe a un
96         registro
97     }
98     private void EnciendeVerde()
99     {
100         Wago.WriteMultipleCoils(0, new bool[] { false, false, true }); //Escribe a un
101         registro
102     }
103     private void EnciendeRojo() //Escribe a un
104     registro
105     {
106         Wago.WriteMultipleCoils(0, new bool[] { true, false, false });
107     }
108     private async void button1_Click(object sender, EventArgs e)
109     {

```

```

109
110     bStableRecolectado = Wago.ReadDiscreteInputs(0, 1)[0]; // Lee bits o bobinas
        de la b scula
111
112     int[] rPeso = Wago.ReadHoldingRegisters(0, 1); // Lee registro de la
        b scula
113
114     PesoRecolectado = rPeso[0]; // El registro cero
        corresponde al peso
115
116
117
118     this.item.action = ID_Instruccion; //Crear objeto JSON a serializar
119     this.item.turn = InstruccionRecibida;
120     this.item.size = PesoRecolectado;
121     this.item.stable = bStableRecolectado;
122
123
124     // Consumir con POST
125     var json = JsonConvert.SerializeObject(item); //Serializa el JSON
126     var stringContent = new StringContent(json, UnicodeEncoding.UTF8, "
        application/json");
127     //Realiza el post request al API enviandole los datos de la bascula
128     HttpResponseMessage response = await client.PostAsync("https://localhost:7101
        /Semaforo/insertar", stringContent);
129
130
131     if (response.IsSuccessStatusCode)
132     {
133         textBox2.Text = "Guadado existosamente en la base de datos";
134     }
135     else
136     {
137         textBox2.Text = "Error al guardar el peso";
138     }
139
140
141
142
143     //Consumir con GET
144     HttpResponseMessage getResponse = await client.GetAsync("https://localhost:71
        01/Semaforo/obtener");
145     if (getResponse.IsSuccessStatusCode)
146     {
147         var readTask = getResponse.Content.ReadAsStringAsync();
148         readTask.Wait();
149
150         GetBascula readBD = JsonConvert.DeserializeObject<GetBascula>(readTask.
            Result);
151
152         textBox1.Text = readBD.size.ToString();
153         if (readBD.stable)
154         {
155             textBox3.Text = "El peso es Estable";
156         }
157     }
158

```

```

159         else
160         {
161             textBox3.Text = "El peso NO es Estable, Repita!!";
162         }
163     }
164     else
165     {
166         textBox1.Text = "Error";
167     }
168 }
169
170 }
171
172 private void button2_Click(object sender, EventArgs e) //Evento de botn Verde
173 {
174     EnciendeVerde();
175 }
176
177 private void button3_Click(object sender, EventArgs e) //Evento de botn Rojo
178 {
179     EnciendeRojo();
180 }
181
182 private void button4_Click(object sender, EventArgs e) // Evento de botn
183     Amarillo
184 {
185     EnciendeAmarillo();
186 }
187
188 private void button5_Click(object sender, EventArgs e) //Botn de finalizar
189 {
190     bool status = CerrarConexionWago();
191
192     if(status)
193     {
194         textBox5.Text = "Finalizado";
195     }
196 }
197
198 private void button6_Click(object sender, EventArgs e) //Botn de iniciar
199 {
200     Wago.IPAddress = "192.168.0.1"; //Establece conexin con la b scula
201     en la red
202     Wago.Port = int.Parse("502"); //Puerto default de Modbus TCP
203     bool conexionEstablecida = AbrirConexionWago(); //Abre conexin Modbus
204
205     if(conexionEstablecida)
206     {
207         textBox4.Text = "Iniciado";
208     }
209 }
210 }
211 }
212 }

```