



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO
CARRERA DE INGENIERÍA ELECTRÓNICA**

**IMPLEMENTACIÓN DE UN GEMELO DIGITAL UTILIZANDO SISTEMAS
EMBEBIDOS Y MODELAMIENTO 3D DEL MÓDULO SORTING
APORTANDO EN LA ENSEÑANZA DE LA UPS**

Trabajo de titulación previo a la obtención del
Título de Ingeniero Electrónico

AUTOR: Edison Rodrigo Llumiquinga Lagla
Pablo Enrique Parreño Escobar

TUTOR: Gustavo Javier Caiza Guanochanga

Quito-Ecuador
2022

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Edison Rodrigo Llumiquinga Lagla con documento de identificación N.º 0503511578 y Pablo Enrique Parreño Escobar con documento de identificación N.º 1712636941; manifestamos que:

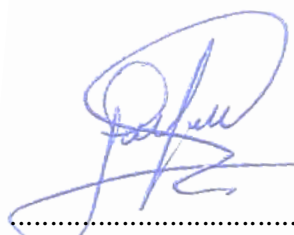
Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 03 de agosto del año 2022

Atentamente,



.....
Edison Rodrigo Llumiquinga Lagla
0503511578



.....
Pablo Enrique Parreño Escobar
1712636941

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Nosotros, Edison Rodrigo Llumiquinga Lagla con documento de identificación N.º 0503511578 y Pablo Enrique Parreño Escobar con documento de identificación N.º 1712636941, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico : " Implementación de un gemelo digital utilizando sistemas embebidos y modelamiento 3d del Módulo Sorting aportando en la enseñanza de la UPS ", el cual ha sido desarrollado para optar por el título de: Ingeniero Electrónico en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Quito, 03 de agosto del año 2022

Atentamente,



.....
Edison Rodrigo Llumiquinga Lagla
0503511578



.....
Pablo Enrique Parreño Escobar
1712636941

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Gustavo Javier Caiza Guanochanga con documento de identificación N° 1721192191, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: IMPLEMENTACIÓN DE UN GEMELO DIGITAL UTILIZANDO SISTEMAS EMBEBIDOS Y MODELAMIENTO 3D DEL MÓDULO SORTING APORTANDO EN LA ENSEÑANZA DE LA UPS, realizado por Edison Rodrigo Llumiquinga Lagla con documento de identificación N° 0503511578 y Pablo Enrique Parreño Escobar con documento de identificación N° 1712636941, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 03 de agosto del año 2022

Atentamente,



Ing. Gustavo Javier Caiza Guanochanga, Mgs.

1721192191

DEDICATORIA

El presente proyecto de titulación va dirigido a mis padres, Luis Llumiquinga y Carmen Lagla ya que sin su apoyo incondicional no hubiese sido posible llegar hasta esta etapa de mi vida, ya que ellos han infundido en mí, valores morales y son mi fuente más importante de inspiración para seguir adelante cada día de mi vida y llegar a ser una persona responsable lo que me permitió culminar con éxito este gran primer paso en mi vida profesional.

Rodrigo

A mi familia, ya que sin su apoyo, consejos y bendición que me brindan día a día no lo hubiese logrado, gracias a su esfuerzo y trabajo son mi motivo más grande para seguir adelante.

Y sin dejar atrás a mis amigos por su gran aporte por confiar en mí, he hicieron que esto se haya logrado.

Pablo

AGRADECIMIENTOS

Agradezco a Dios quién supo darme la sabiduría, la capacidad ya que él ha sido mi guía en el camino del bien y no ha permitido tropezar en el transcurso de mi vida.

A mi familia por su apoyo para poder lograr la conclusión de mi carrera universitaria a pesar de todos los obstáculos que se han presentado en vida he sabido salir adelante con su bendición que me brindan día con día.

Rodrigo

Primero agradezco a Dios, por su bondad y amor infinito, ya que me ha permitido una alegría más a mis logros que son el resultado de su ayuda, y cuando he caído ha sido para ponerme a prueba y así aprender de mis errores y mejorar como persona, y crecer de distintas maneras.

El presente trabajo de tesis ha llegado a ser una gran bendición en todo sentido a lo largo de mi vida universitaria; he logrado concluir con éxito un proyecto que al iniciarlo parecería interminable, gracias a mis padres y amigos.

Pablo

ÍNDICE DE CONTENIDOS

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN	ii
CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA	iii
CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN	iv
DEDICATORIA	v
AGRADECIMIENTOS.....	vi
ÍNDICE DE CONTENIDOS	vii
ÍNDICE DE FIGURAS	x
ÍNDICE DE TABLAS.....	xi
RESUMEN.....	xii
ABSTRACT	xiii
INTRODUCCIÓN	xiv
CAPÍTULO I.....	15
ANTECEDENTES.....	15
Planteamiento del problema.....	15
Justificación.....	15
Objetivos	16
Objetivo General	16
Objetivos Específicos	16
CAPÍTULO 2	18
FUNDAMENTACIÓN TEÓRICA	18
2.1 Industria 4.0	18
2.2 Digital Twins o Gemelo Digital	19
2.3 Módulo Sorting (Estación de Clasificación).....	19
2.4 SolidWorks	19
2.5 Unity 3D	20
2.6 Visual Studio.....	20
2.7 Arduino.....	21
2.8 TIA Portal	21
2.9 Protocolo de Ethernet Industrial	22
2.10 Modbus TCP/IP.....	23

2.11 Ethernet Shield	24
2.12 Módulo Digital SM 1223	25
2.13 Switch.....	25
CAPÍTULO 3	26
3.1 Descripción del diseño del Gemelo Digital Módulo Sorting (Estación de Clasificación).....	26
3.2 Parametrización del Módulo Sorting del Sistema Modular de Producción (MPS)..	27
3.2.1 Proceso básico del Módulo Sorting (Estación de Clasificación)	27
3.2.2 Modelamiento de los componentes del Módulo Sorting (Estación de Clasificación) en SolidWorks	28
3.2.3 Identificación del controlador Módulo Sorting (Estación de Clasificación)	31
3.2.3.1 PLC S7 – 300 CPU 313C.....	33
3.2.4 Descripción de los elementos seleccionados para el control del Módulo Sorting (Estación de Clasificación)	34
3.2.4.1 PLC S7 – 1200 CPU 1212C	34
3.2.4.2 Comparativa entre el PLC S7 – 300 y PLC S7 – 1200	35
3.2.4.3 Comparativa del direccionamiento de entradas y salidas del PLC S7 – 300 al PLC S7 – 1200.....	35
3.2.4.4 Módulo Digital SM 1223 (6ES7 223 – 1BL32 – 0XB0)	37
3.2.4.5 Arduino Mega 2560.....	37
3.2.4.6 Ethernet Shield	38
3.2.4.7 Fuente Conmutada.....	38
3.2.5 Esquema eléctrico del Módulo de Control	39
3.2.6 Diseño del Módulo de Control	39
3.3 Programación en Arduino	40
3.4 Programación en TIA Portal.....	42
3.5 Programación en Unity	44
3.6 Comunicación entre el Módulo Sorting (Estación de Clasificación) y el Módulo virtualizado en 3D.....	44
CAPÍTULO 4	46
ANÁLISIS Y RESULTADOS.....	46
4.1 Comunicación	46
4.1.1 Comunicación entre Hardware y Software.....	46
4.2 Tipos de Comunicación	46

4.2.1 Comunicación Tipo 1.....	46
4.2.2 Comunicación Tipo 2.....	47
4.3 Protocolos de Red Ethernet.....	48
4.4 Protocolo de Comunicación Modbus TCP/IP.....	48
4.5 Análisis del tiempo de ejecución entre los Modelo Físico y Modelo Digital	49
CONCLUSIONES	53
RECOMENDACIONES	54
REFERENCIAS.....	55
ANEXOS.....	59

ÍNDICE DE FIGURAS

Figura 2.1: Trama Modbus en TCP/IP.....	24
Figura 3.1: Pasos a seguir para implementar un Gemelo Digital	26
Figura 3.2: Proceso de la Estación de Clasificación (Módulo Sorting)	27
Figura 3.3: Diseño en 3D del Módulo Sorting (Estación de Clasificación) en SolidWorks	31
Figura 3.4: Diagrama de Bloques del Diseño del Módulo de Control	40
Figura 3.5: Inclusión de las librerías Ethernet y Modbus.....	40
Figura 3.6: Declaraciones de constantes en Arduino	41
Figura 3.7: Configuración de la red Ethernet	41
Figura 3.8: Inicialización de los registros de envío y recepción de datos.....	42
Figura 3.9: Creación del bloque de datos para el envío de datos	42
Figura 3.10: Creación del bloque de datos para la recepción de datos	43
Figura 3.11: Creación del bloque de datos para el envío de datos PLC hacia Arduino .	43
Figura 3.12: Creación del bloque de datos para la recepción de datos Arduino hacia PLC	44
Figura 3.13: Proceso del envío de datos del Gemelo Digital – Módulo Sorting	45
Figura 3.14: Proceso del envío de datos del módulo Sorting – Gemelo Digital	45
Figura 4.1: Comunicación de un entorno virtual y un entorno físico.....	46
Figura 4.2: Comunicación bidireccional entre Unity y Arduino.....	47
Figura 4.3: Comunicación entre Arduino y PLC S7 – 1200.....	47
Figura 4.4: Comunicación de la Red Ethernet.....	48
Figura 4.5: Envío y recepción de datos de un punto A hacia un punto B.....	49
Figura 4.6: Grafica de Análisis de Comunicación entre el Arduino y el PLC S7-1200	50
Figura 4.7: Grafica de Análisis de Comunicación entre el PLC S7-1200 y el Arduino	50
Figura 4.8: Grafica de Análisis de Comunicación entre TIA PORTAL y el PLC S7-1200	51
Figura 4.9: Grafica de Análisis de Comunicación	52
Figura 4.10: Resumen de Captura de Datos WireShark.....	52

ÍNDICE DE TABLAS

Tabla 3.1: Tabla comparativa. Estación de Clasificación vs Modelo 3D	28
Tabla 3.2: Entradas y Salidas del Módulo 1(Módulo Sorting)	32
Tabla 3.3: Entradas y Salidas del Módulo 2 (Módulo Sorting)	32
Tabla 3.4: Características del PLC S7 – 300 CPU 313C.....	33
Tabla 3.5: Características del PLC S7 – 1200 CPU 1212C.....	34
Tabla 3.6 Diferencias de PLC S7 – 300 y PLC S7 – 1200.....	35
Tabla 3.7: Comparativa de entradas y Salidas del Módulo 1(Módulo Sorting)	35
Tabla 3.8: Comparativa de entradas y salidas del Módulo 2 (Módulo Sorting).....	36
Tabla 3.9: Características del Módulo digital SM 1223	37
Tabla 3.10: Características del Arduino Mega 2560.....	37
Tabla 3.11: Características de la placa Ethernet Shield	38
Tabla 3.12: Características de la Fuente Conmutada	39
Tabla 4.1: Direccionamiento de Componentes.....	49

RESUMEN

En la actualidad el mundo atraviesa por una pandemia global, la cual ha provocado estragos económicos a varios sectores del Ecuador, de igual manera en el sector privado y público, con esto ha provocado cambios en las actividades que solían hacer de forma normal, como es el caso en el aprovechamiento de los laboratorios de la UPS, lo que permita acceder a un conocimiento adecuado de parte de los docentes hacia los estudiantes.

El presente proyecto de titulación describe la ejecución de un gemelo digital del Módulo Sorting del MPS, en este proceso se involucran dos partes la primera es la digitalización del Módulo Sorting por medio de un software de diseño CAD que permite la simulación y la segunda parte es la comunicación entre el diseño digital y el físico, esto en conjunto permite el análisis del comportamiento de los diferentes componentes como son: actuadores y sensores que son obligatorios para el normal funcionamiento del gemelo digital.

El gemelo digital recrea un entorno ideal para el aprendizaje y aprovechamiento del laboratorio de MPS, por parte de los estudiantes y a su vez por docentes quienes han sido en gran parte afectados por la pandemia, la cual ha modificado el aprendizaje tradicional a uno virtual y por medio de este diseño permita lograr una enseñanza acorde a estos cambios que atravesamos y contribuir en el aprendizaje de los futuros profesionales que brinda la UPS.

En el presente trabajo se logró conectar la estación de clasificación (Sorting) y su modelo digital utilizando el protocolo de comunicación ethernet con un retraso entre los modelos relativamente pequeño y poder observar la ejecución en los dos modelos simultáneamente.

ABSTRACT

At present, the world is going through a global pandemic, which has caused economic havoc in various sectors of Ecuador, in the same way in the private and public sectors, with this it has caused changes in the activities that they used to do normally, such as the case in the use of UPS laboratories, which allows access to adequate knowledge from teachers to students.

This degree project describes the execution of a digital twin of the MPS Sorting Module, two parts are involved in this process, the first is the digitization of the Sorting Module by means of CAD design software that allows simulation and the second part is the communication between the digital and the physical design, this together allows the analysis of the behavior of the different components such as: actuators and sensors that are mandatory for the normal operation of the digital twin.

The digital twin recreates an ideal environment for learning and taking advantage of the MPS laboratory, by students and, in turn, by teachers who have been largely affected by the pandemic, which has modified traditional learning to a virtual one and for Through this design, it allows us to achieve teaching according to these changes that we are going through and contribute to the learning of future professionals offered by UPS.

In the present work, it was possible to connect the sorting station and its digital model using the ethernet communication protocol with a small delay between the models and to be able to observe the execution in the two models simultaneously.

INTRODUCCIÓN

En este trabajo de titulación se describe el desarrollo y ejecución de un gemelo digital de un Módulo Sorting del laboratorio MPS de la UPS, tiene una gran ventaja para el aprendizaje de los estudiantes y obtener el conocimiento necesario, en los diversos procesos industriales que brinda el laboratorio de MPS y es un apoyo a los docentes que imparten la materia. La virtualización en 3D del Módulo Sorting se procederá comunicar con su parte física que se entrelazaran de tal forma que ejecutaran procesos programados y a su vez observar el funcionamiento en si del gemelo digital.

El capítulo uno está compuesto del planteamiento del problema, justificación y los objetivos general y específicos que permiten la implementación del gemelo digital.

El capítulo dos describe en detalle los conceptos necesarios que se utilizara para la implementación del proyecto y descripción del Módulo Sorting y los softwares necesarios en la implementación del gemelo digital.

En el capítulo tres comprende el proceso de la utilización de los softwares, necesarios para la virtualización 3D del Módulo Sorting que se utilizara para la implementación del gemelo digital y su correcto funcionamiento.

El capítulo cuatro se analiza el funcionamiento del gemelo digital y además de obtener como resultado conclusiones y recomendaciones observadas en todo el proceso de la implementación del gemelo digital del Módulo Sorting.

CAPÍTULO I

ANTECEDENTES

Planteamiento del problema

En la actualidad los laboratorios de la Universidad Politécnica Salesiana, no se encuentran en uso de manera normal, por causa de una pandemia global, por tal motivo existe la necesidad de encontrar una forma para que el aprovechamiento de los laboratorios de forma remota sea completo, esto implica que los docentes y los estudiantes, obtendrían tanto el conocimiento teórico y práctico, que hoy en día es muy importante en la industria, que está en constante cambio y requiere el aprendizaje obtenido en la universidad.

Por lo tanto, en el diseño de un gemelo digital tiene un gran impacto en los procesos industriales, ya que da inicio para próximos proyectos y con ello permite el monitoreo de las diversas señales, tanto de entrada y salida que conforman la estación modular del MPS.

A través del proyecto se debe dar ciertas condiciones de trabajo en los laboratorios de la UPS, con el fin de precautelar el bienestar de los docentes y los estudiantes a la hora de utilizar los laboratorios de forma remota.

Con ello se permitirá relacionar la parte física y digital de los distintos procesos que se encuentran en el laboratorio de MPS de la UPS. Y garantizar un aprendizaje que los estudiantes necesitan para enfrentarse en la industria que alienta a seguir en un cambio constante, que hoy en día en un mundo globalizado.

Justificación

Ante la situación actual que se encuentra, en el mundo y la no disponibilidad de realizar actividades que se efectuaba en los distintos centros educativos de forma normal y la repentina necesidad de adoptar el aprendizaje en línea, entre docentes y estudiantes se encuentra en la búsqueda de transición del sistema educativo a distancia o en línea. Esto permite elaborar programas de educación de calidad los cuales se adapten para satisfacer las necesidades de los estudiantes y de igual manera para los docentes, quienes a su vez deben afrontar en una educación innovadora y retos en su vida profesional. (Santuario, Educación y pandemia: Una visión académica, 2020)

En el presente proyecto técnico surge de la necesidad de diseñar un gemelo digital, esto permitirá realizar una digitalización de procesos industriales o de ingeniería, que se encuentran estrechamente vinculados, tanto en un entorno educativo o industrial, con ello permite garantizar un avance considerado en la optimización de diversos procesos industriales. (Rodríguez N. , Uso de nuevas tecnologías para la competitividad del país, 2018)

El proyecto busca recabar información necesaria para posteriores proyectos que permita desarrollar proyectos que se encuentren vinculados a la comunidad universitaria y a su vez privadas o públicas con ello lograr grandes avances en la digitalización industrial.

En la actualidad en nuestro país no hay suficiente información sobre la digitalización de procesos industriales, el proyecto se encamina en afianzar un conocimiento en dicho campo de desarrollo, esto permitirá en la recopilación de información necesaria, para futuros proyectos técnicos y contribuirá en analizar las ventajas y las desventajas de implementar dicho proyecto en las entidades públicas o privadas de nuestro país. El trabajo tiene como finalidad motivar a futuras investigaciones en el campo de desarrollo que se está planteando en este proyecto, como sería el caso efectuar in estudio de los procesos de Digital Twins, tanto en el campo universitario como en la industria del país. (Rodríguez N. , Uso de nuevas tecnologías para la competitividad del país, 2018)

Objetivos

Objetivo General

Implementar un Gemelo Digital del Módulo Sorting de Festo Didactics para que satisfaga las necesidades de aprendizaje en un Sistema Modular de Producción (MPS) de la UPS mediante el uso de software para modelado diseño y sistemas embebidos para su comunicación.

Objetivos Específicos

Realizar una investigación bibliográfica para establecer la comunicación entre un modelo físico y virtual usando como base el protocolo de comunicación Ethernet.

Diseñar el modelo 3D del Módulo Sorting de MPS mediante el uso de software de modelado 3D para establecer la comunicación con su modelo físico.

Implementar la conectividad de la red entre el Módulo Sorting del MPS y su modelado 3D mediante el uso del protocolo Ethernet y sistemas embebidos para comprobar su comunicación.

Analizar costos y resultados en la implementación del Gemelo Digital que permita lograr futuras investigaciones para la digitalización de los procesos industriales y a su vez permita comprobar el presente diseño con un modelo físico.

CAPÍTULO 2

FUNDAMENTACIÓN TEÓRICA

2.1 Industria 4.0

El termino 4.0 se manifestó por primera vez en el 2011 y su alcance ha crecido exponencialmente. De lo que no cabe duda es que su inicio es en Alemania. (Jaume Martínez, 2019)

El conocimiento de Industria 4.0 se compone de una gran cantidad de tecnologías en las cuales se basa la reinención del sector industrial, se puede notar que existen en la actualidad varias que son esenciales entre ellas se puede nombrar los datos masivos, los sistemas integrados, Internet de las Cosa, además de ciberseguridad. Todas estas tecnologías junto con tantas otras se relacionan con ellas, tal como la fabricación aditiva, la realidad virtual, la implementación de sensores en los sistemas, con lo que se ha conseguido llevar a cabo proyectos transformadores en distintos sectores de la industria. Entre los sectores donde se puede observar el mayor cambio en sus sistemas de fabricación están el automotriz y aeronáutico. Estos son líderes de transformación ya que el nivel de producción es grande y bastante costosa cuando se habla de aeronáutica. Entre los principales cambios se puede notar la inclusión de sistemas de realidad virtual que se utilizan para garantizar la ausencia de fallos en el ensamblado, así también en este proceso aporta de una manera positiva el concepto de fabricación aditiva. (Losada, 2017)

Para comprender de mejor manera el concepto de fabricación aditiva se puede mencionar que es el proceso de adicionar material capa a capa de manera controlada una manera más practica de verlo es el sistema de impresión 3d en el cual se pueden producir formas geométricas personalizadas en función de las necesidades de cada sector. (Edimar, 2019)

La industria 4.0 promueve la plena integración de la información y el conocimiento en cada una de las etapas del ciclo de un producto en una respuesta a una serie de tendencia del mercado hacia las personas y el uso de nuevas y variadas tecnologías. Entonces se enumeran las siguientes, tomadas en cuenta el nivel de significancia en la actualidad: Inteligencia Artificial mediante el Aprendizaje Profundo, análisis de datos, Internet de las cosas, Simulación, Fabricación aditiva, Robots autónomos. (Jaume Martínez, 2019)

2.2 Digital Twins o Gemelo Digital

Es una tecnología 3D capaz de crear un modelo digital de una entidad real. El gemelo digital le permite capturar información en tiempo real del modelo físico e interactuar con él a través de los sistemas digitales. Cuando hablamos de un modelo digital del mundo real, a menudo nos encontramos con el término mundo espejo, el mundo que se puede encontrar en la nube y las tecnologías necesarias para crear y visualizar bienes, los miles de millones de polígonos que deben tener los modelos 3D realistas deben tener. Una vez que el entorno alcanza un cierto tamaño. (Takahashi, 2022)

2.3 Módulo Sorting (Estación de Clasificación)

Según el estándar VDI 2860, el modo de clasificación se realiza a través de subcategorías de las funciones del procesador. Según los tipos de los componentes se incluyen diferentes adaptadores. Los componentes por manejar deben cargarse específicamente para que no obstruyan con el funcionamiento de la válvula. En el módulo de clasificación, las piezas a procesar se clasifican por material y color. Las funciones de la estación de clasificación incluyen: Clasificar y diferenciar las piezas a procesar según sus propiedades. El módulo de clasificación envía las piezas distribuyéndolas en tres planos inclinados. Los segmentos colocados en la parte superior del portador son detectados por un tabique óptico ramificado.

La mordaza neumática sujeta las piezas para establecer sus características. Los sensores que se encuentran en el módulo de detección de piezas determinan el material y el color de las piezas (negro, rojo, metálico). Para dirigir las piezas a sus rampas inclinadas se utilizan actuadores electromagnéticos. El sensor de reflexión directa supervisa el número de segmentos en el plano inclinado. (Didactic, 2015)

2.4 SolidWorks

Es un software de diseño para modelamiento 3D que engloba un ambiente gráfico intuitivo y de manera fácil de utilizarlo. Lo que permite dar soluciones tridimensionales, con lo cual acceder a una gran ventaja, a la hora de integrar un sinnúmero de funciones, para ayudar en el modelado de piezas, y con ello crear ensambles y enlazar entre diferentes operaciones de software.

SolidWorks se caracteriza por ser un software versátil a la hora de asociar las distintas funciones, que se emplean en el proceso del diseño y a su vez acceder en la visualización, editar y eliminar las distintas operaciones realizadas de manera bidireccional, por ello es

un software de diseño 3D más completo en el mundo del diseño CAD. (Gómez González, 2008)

2.5 Unity 3D

Unity fue lanzado al mercado el 1 de junio 2005, por una empresa Danesa Unity Technologies, es un motor para la creación de videojuegos, sea convertido muy popular desde su lanzamiento a nivel mundial, que permite desarrollar un sinfín de posibilidades en animación, diseño y audio con lo cual permite acceder a varios componentes, en el desarrollo de videojuegos. Como son el caso de importar archivos 3D's, Cinema4D, Blender, FBX los cuales permiten en la creación de escenarios para los proyectos que los desarrolladores necesiten para el videojuego. (Lidon, Unity 3D, 2019)

Unity se caracteriza por tener un editor de lenguaje de programación en scripts, intuitivo para los futuros desarrolladores de videojuegos, lo cual no debe ser un experto programador para lograr utilizarlo y con la herramienta que incluye Unity en el desarrollo MonoDevelop, que permite el acceso a los scripts en C#, JavaScript, Python entre otros. Una gran ventaja es que Unity tiene una API, que se puede consultar cualquier duda que se presente durante el progreso de la aplicación a crear. (Ouazzani, 2012)

2.6 Visual Studio

El software de desarrollo integrado Microsoft Visual Studio puede ser usado en distintos sistemas Windows y MacOS. Este entorno de desarrollo es compatible con múltiples plataformas de desarrollo que comprenden los lenguajes de programación mas usados en la actualidad además de ser una de las herramientas mas usadas en la actualidad ya que contiene herramientas intuitivas para la rápida codificación y que facilita el proceso de desarrollo, además se puede sumar a todas estas características las capacidades de conectividad en línea a través de Windows Azure.

Este editor también incorpora herramientas suficientes para el desarrollo Web siendo una de las herramientas mas usadas para el desarrollo de sitios y aplicaciones Web en la actualidad. Visual Studio en la actualidad además posee la capacidad de crear aplicaciones que se comuniquen entre estaciones de trabajo, dispositivos embebidos, consolas de videojuegos, entre otros. (Microsft, 2022)

2.7 Arduino

Arduino pertenece a un grupo de plataformas de creación de dispositivos electrónicos de código abierto, la cual se basa en hardware, es gratuita además flexible y fácil de usar para aprendices y desarrolladores. Esta plataforma aporta con el desarrollo de algoritmos capaces de determinar la creación de dispositivos.

Arduino se concibe como un proyecto el cual no se basa en una placa en específico lo que incentiva a los desarrolladores a la creación de sus propios modelos a partir de los diseños básicos y es por lo que se pueden encontrar en el mercado diferentes tipos de placas que va desde usos simples hasta grandes usos para resolver complicados problemas.

Arduino está orientado a IoT o impresión 3D y por ello en base a estas funcionalidades se puede encontrar varios tipos de precios. (Fernández Y. , 2020)

2.8 TIA Portal

Portal Totalmente Integrado de Automatización es la principal plataforma de ingeniería de Siemens que suministra soluciones de automatización en todos los sectores industriales, la cual integra en su plataforma varios procesos de automatización de los procesos industriales. Además, por ser un sistema modular se pueden añadir nuevas funcionalidades en función de las necesidades específicas de cada campo.

Tia Portal en su versión más reciente reúne características que siguen innovando con la realidad actual de los procesos industriales además de incluir nuevas características de seguridad y conectividad que garantizan la compatibilidad con los componentes miembros de la industria.

Las principales nuevas características son las siguientes:

- STEP 7 V17 donde se desarrolla, dimensiona y selecciona los componentes para el proyecto.
- WINCC herramienta útil para el desarrollo tanto para las interfaces hombre-máquina para sistemas SCADA, como para paneles de operación.
- SINAMICS Startdrive Componente clave para automatización de máquinas, provee la interacción conveniente entre los drives y el controlador para las aplicaciones de control de movimiento con los controladores Simatic y los variadores Sinamics.
- SIMCode

- Servidor de Proyectos de TIA
 - Almacenamientos de proyectos en la red.
 - Administración de accesos y usuarios central
 - Permite revisiones de proyectos.
 - Registro de cambios para los objetos sincronizados.
 - Ejecución de tareas con Scripts.
- Nuevas Librerías de Tia Portal
- Mejora de seguridad para comunicación PG/HMI
- Administración de Usuarios y Control de Acceso
- Add-Ins que permiten la funcionalidad de Tia Portal usando API.

A parte de las funcionalidades mencionadas Tia Portal V17 incorpora características para IoT que permiten interactuar de manera más sencilla con estos dispositivos.

TIA Portal es mejor conocido por programar la serie de PLC's Simatic de Siemens, formada por los autómatas programables S7 – 1200 y su hermano mayor S7 – 1500.

TIA Portal permite la integración de varias aplicaciones de software industrial en el que intervienen procesos de fabricación industriales en una sola interfaz, lo que facilita considerablemente el aprendizaje, comunicación y operación. Ya se trate de la programación de consolas, la configuración de pantallas HMI o la configuración de parámetros del variador; esta nueva arquitectura de software permite tanto a los usuarios como los nuevos profesionales interactuar de una manera automática y eficiente porque no es necesario perder el tiempo ya que los sistemas tienen diferentes orígenes.

Además, TIA Portal también está diseñado para detectar y administrar fácilmente los errores de programación en línea, lo que ayuda a reducir o en algunos casos eliminar el tiempo de inactividad de la línea de producción y aumentar la disponibilidad de la planta. (Festo, 2022)

2.9 Protocolo de Ethernet Industrial

Protocolo Industrial Ethernet es un esquema de red de comunicaciones el cual es capaz de soportar datos de gran tamaño a velocidades de 10 Mbps o 100 Mbps y hasta 1500 bytes por paquete.

En la industrial, es particularmente útil cuando se utiliza en aplicaciones de control. Las principales ventajas de este tipo de red es que es muy fácil de configurar, expandirse y montar ya que en la actualidad es uno de los protocolos más utilizados.

Este tipo de tecnología se utiliza con computadoras personales, robots, periféricos y conmutadores de entrada/salida, controladores lógicos programables (PLC) y otros periféricos. Toda esta estandarización está sujeta y respaldada por Industrial Ethernet Association (IEA), ControlNet International (CI) y Open Network Equipment Vendors Association (ODVA).

Como anteriormente este protocolo se a convertido en una de las principales tendencias para la migración de datos en aplicaciones industriales. Esto es porque este protocolo se a convertido en uno de los mas confiables y de alto rendimiento por lo que se ha convertido fácilmente en el preferido para los sistemas de control y automatización.

Como una explicación básica de que es el protocolo Ethernet Industrial y cómo funciona seria simple mencionar que son los protocolos industriales usados anteriormente encapsulados dentro del protocolo Ethernet de forma que se garantice el envío y recepción de la información correcta para realizar una operación específica. (aula21, 2022)

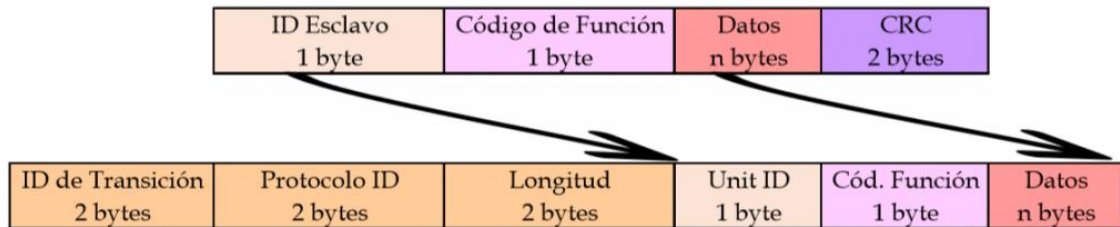
2.10 Modbus TCP/IP

Ethernet no es un protocolo. TCP/IP es un mecanismo de transporte no una capa de aplicación. Por lo tanto, sin un acuerdo de formato de datos para los procesos, automatización o dispositivos integrados, Ethernet sería un caos.

Esto se está resolviendo mediante la portabilidad ya en protocolos de automatización a Ethernet, existen cuatro formatos principales entre ellos esta Modbus. Los perfiles de los protocolos se están portando a Ethernet TCP/IP.

Modbus es el más predominante protocolo serial en automatización, estos paquetes Modbus son fácilmente encapsulados en TCP/IP. (Sink, 2001)

Figura 2.1: Trama Modbus en TCP/IP



Fuente: Sothis 2021

2.11 Ethernet Shield

Arduino Ethernet Shield se acopla a las placas de desarrollo Arduino con lo que puede a una determinada red Ethernet.

El dispositivo tiene como principal componente el chip Wiznet W5100 ethernet. El cual proporciona una red (IP). Se utiliza una biblioteca desarrollada para Ethernet lo que mediante sketches permite la conexión a internet.

El Shield tiene un estándar de conexión RJ-45, además de contar con el soporte de Power Over Ethernet que permite la alimentación del dispositivo a través de la red, además consta de una ranura para tarjetas microSD, que puede ser usada para almacenar archivos.

El módulo Shield además cuenta con luces Informativas:

- PWR: indica que tanto el Arduino y como el módulo Shield están alimentados y encendidos.
- ENLACE: indica que el dispositivo esta conectado a una red.
- FULLD: indica que la conexión es full dúplex.
- 100M: indica la presencia de una conexión de 100Mb/s
- RX: módulo shield recibe datos.
- TX: módulo shield envía datos.

El módulo shield es de gran ayuda en la actualidad ya que permite la conexión a las redes locales y globales con esto la integración a nuevas tecnologías como Internet de las Cosas, Domótica, entre otros. (Gutiérrez, 2013)

2.12 Módulo Digital SM 1223

Es un módulo de expansión que se acopla a la CPU de un controlador S7-1200, aumentando las características de este con las siguientes características:

- 8 entradas digitales a 24 (VDC)
- 8 salidas digitales tipo relé que soportan hasta 2 (A)

Estas características permiten al controlador expandir su funcionalidad y controlar modularmente desde una pequeña a gran industria. (Rodríguez & Quinchiguango De la cruz, 2021)

2.13 Switch

Los Switch son dispositivos que permiten la interconexión de varios dispositivos dentro de una misma red Ethernet, además estos se pueden catalogar como los componentes activos de la red que se encargan de distribuir de forma activa la información entre los dispositivos conectados en ella, con esto se puede decir que los switches son dispositivos que junto con el cableado constituyen las redes de área local. (Gonzales, 2013)

CAPÍTULO 3

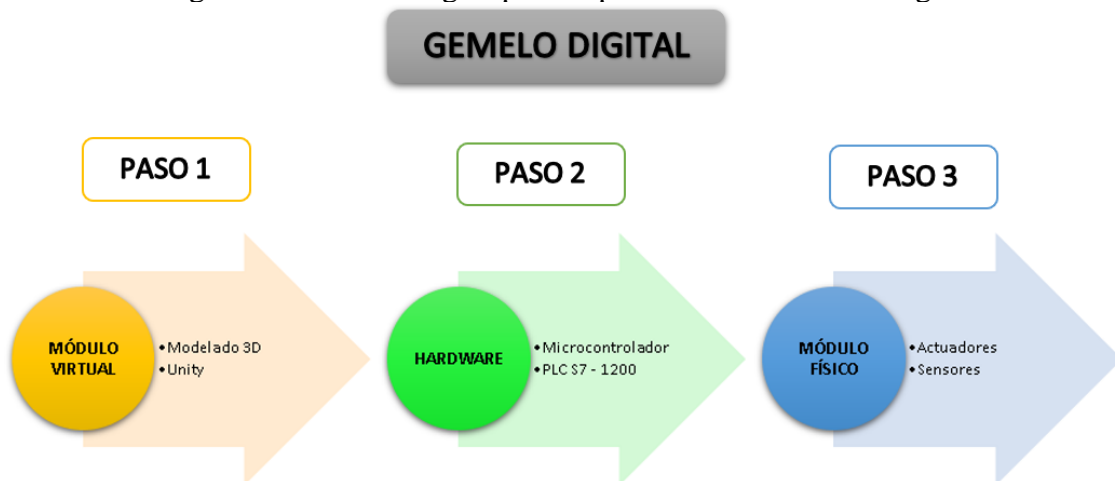
DISEÑO E IMPLEMENTACIÓN DE UN GEMELO DIGITAL DEL MÓDULO SORTING

3.1 Descripción del diseño del Gemelo Digital Módulo Sorting (Estación de Clasificación)

La implementación del Gemelo Digital consiste en el diseño de un modelo virtual que se asemeje en lo más posible al físico y además que puedan interactuar e intercambiar información entre ellos.

Se busca replicar un pequeño proceso del funcionamiento del Módulo Sorting (Estación de Clasificación) para lo cual es importante el intercambio de ciertos parámetros como son los sensores y actuadores. Para resumirlo se enumera en 3 simples pasos para entenderlo de la manera más simple. (ver Figura 3.1)

Figura 3.1: Pasos a seguir para implementar un Gemelo Digital



Elaborado por: Autores

- Primer paso tomar las medidas de componentes que comprenden el Módulo Sorting, tanto sensores y actuadores, para su diseño en 3D por ello se utilizó SolidWorks, y para ensamblar las piezas se debe utilizar un motor de juegos para este caso se utilizó Unity.
- Segundo paso se utilizará un microcontrolador Arduino y el controlador PLC S7 – 1200, en conjunto con otros dispositivos, es el hardware necesario para enviar y recibir los del Gemelo Digital, la parte física y su parte virtualizada.
- Tercer paso es el conjunto de actuadores, sensores y otros componentes, que forman parte del Módulo Sorting (Estación de Clasificación), en este paso el

Módulo Sorting (Parte Física) envía información a través de la red al Arduino el cual a su vez es encargado de decodificar y comunicarse con su Modelo Virtual, y con ello poder visualizar dicho proceso en el software Unity.

A continuación, se describirá cada uno de los pasos mencionados anteriormente con más detalle para la implementación del Gemelo Digital del Módulo Sorting (Estación de Clasificación).

3.2 Parametrización del Módulo Sorting del Sistema Modular de Producción (MPS)

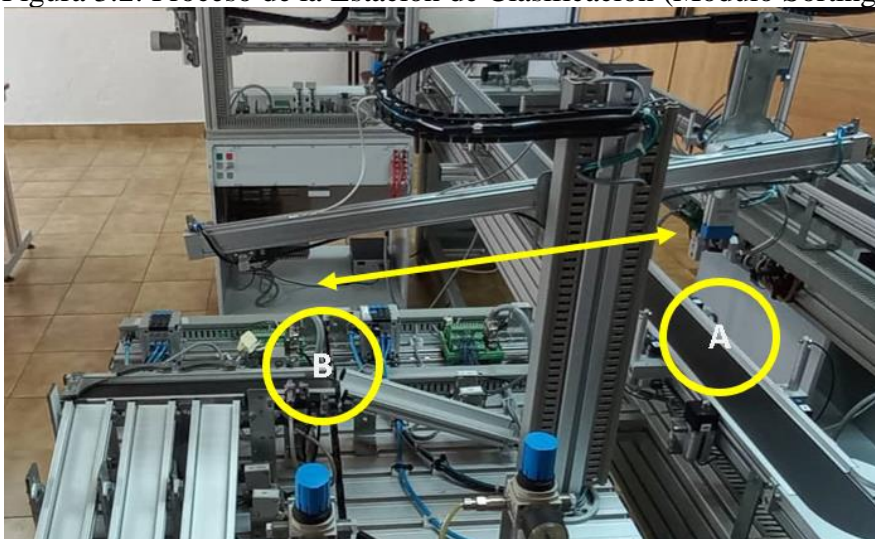
Se realizó el diseño CAD del modelo físico tomando como referencia las medidas reales del mismo.

Se definió las características de movimiento de los componentes del modelo físico para que el modelo virtual realice las funciones de forma idéntica con el modelo físico.

3.2.1 Proceso básico del Módulo Sorting (Estación de Clasificación)

El proceso del módulo con la que se va a trabajar inicia desde que la Pinza Neumática en el Módulo PicAlfa recoge la ficha de la posición A (ver Figura 3.2), el brazo en el que se encuentra la Pinza Neumática se desplaza de manera lineal con la finalidad de soltar la pieza en la posición B (ver Figura 3.2), donde se encuentra un sensor inductivo que detecta piezas con características metálicas, un sensor de reflexión directa la cual detecta la presencia de una ficha y un sensor óptico que detecta todas las piezas excepto la de color negro debido a que este no refleja el componente óptico.

Figura 3.2: Proceso de la Estación de Clasificación (Módulo Sorting)





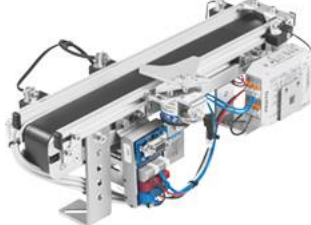
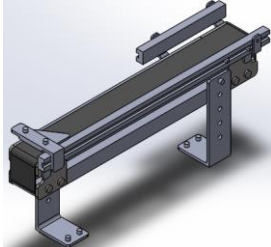
Elaborado por: Autores


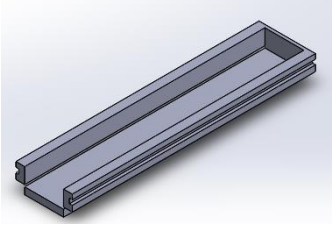

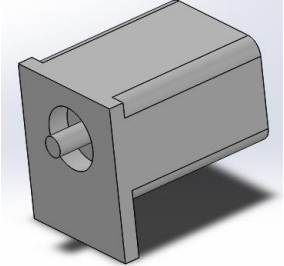
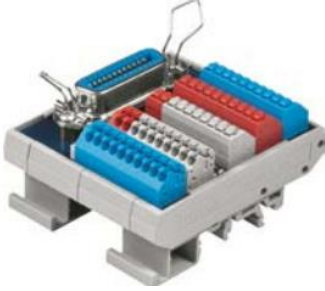
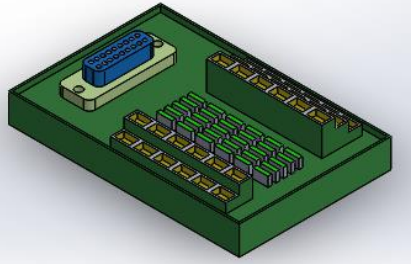

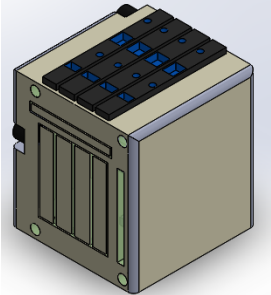

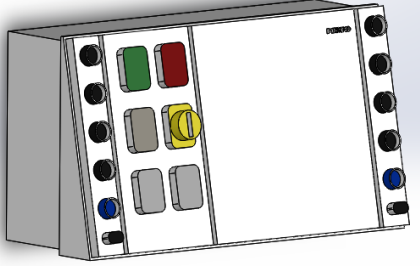
Una vez en la posición B se encuentra la pieza, el brazo de la pinza retorna a la posición A, a continuación, se activa en módulo de tope, que permite evaluar por los tres sensores a la pieza, se activara el módulo transportador, lo cual se desplazé la pieza de acuerdo con la programación, de igual manera los módulos derivadores, se activaran para que la pieza sea clasificada en uno de los módulos rampa que se encuentran en el Módulo Sorting (Estación de Clasificación).

3.2.2 Modelamiento de los componentes del Módulo Sorting (Estación de Clasificación) en SolidWorks

Mediante el software de modelamiento SolidWorks, se realiza el diseño de los distintos componentes que forman parte del Módulo Sorting (Estación de Clasificación), con la finalidad de ser utilizado en Unity. A continuación, se realiza la comparativa de los elementos físicos con los elaborados en SolidWorks (ver Tabla 3.1).

Tabla 3.1: Tabla comparativa. Estación de Clasificación vs Modelo 3D

Módulo Sorting (Estación de Clasificación)	Módulo Físico	Módulo 3D
Módulo PicAlfa		
Módulo transportador		

<p>Módulo de rampa</p>		
<p>Módulo de tope</p>		
<p>Terminal digital E/S SysLink</p>		
<p>Terminales de válvulas CP</p>		
<p>Consola de control MPS</p>		

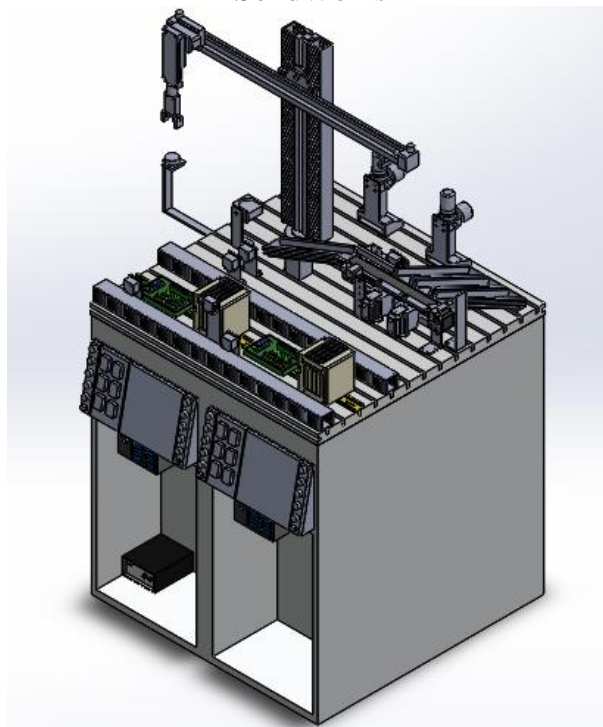
<p>Válvula de interrupción con filtro y regulador</p>		
<p>Módulo de puerta clasificadora neumática</p>		
<p>Sensor de reflexión directa</p>		
<p>Sensor de proximidad inductivo</p>		
<p>Sensor óptico</p>		



Componentes del Módulo Sorting (Estación de Clasificación) desarrollado en el software SolidWorks, Elaborado por: Autores

Al finalizar con las distintas partes del Módulo Sorting (Estación de Clasificación) en SolidWorks, se procede a ensamblar todos sus componentes, así se obtiene (ver Figura 3.3) como resultado, el Módulo Sorting completado que posteriormente será utilizado en el software Unity.

Figura 3.3: Diseño en 3D del Módulo Sorting (Estación de Clasificación) en SolidWorks



Elaborado por: Autores

3.2.3 Identificación del controlador Módulo Sorting (Estación de Clasificación)

Se realizó la caracterización del Módulo Sorting del laboratorio MPS en el que se observa que la estación de clasificación consta de dos bloques a los cuales los llamaremos Modulo 1 en el que se encuentra el Módulo PicAlfa y Modulo 2 en donde se encuentra la banda

transportadora, cada uno de estos Módulos consta de un controlador PLC S7 – 300, CPU 313C (6ES7313 – 5BG04 – 0AB0), a su vez se obtuvo las direcciones de entrada y salida de cada uno de los controladores que otorga el movimiento de actuadores y recibe los datos de sensores. Una vez identificadas las direcciones se puede apreciar en la Tabla 3.2, las direcciones del Módulo 1 y en la Tabla 3.3 las direcciones del Módulo 2.

Tabla 3.2: Entradas y Salidas del Módulo 1 (Módulo Sorting)

Entradas Módulo 1	Piezas
I124.0	Sensor de reflexión directa
I124.1	Sensor izquierdo barra
I124.2	Sensor medio barra
I124.3	Sensor final barra
I124.4	Sensor abajo pinza
I124.5	Sensor arriba pinza
I124.6	Sensor pinza abre/cierra
Salidas Módulo 1	Piezas
Q124.0	Actuador izquierdo brazo
Q124.1	Actuador derecho brazo
Q124.2	Actuador pistón pinza
Q124.3	Actuador garra pinza

Direcciones de entradas y salidas del Módulo 1 (Estación de Clasificación), Fuente: (FESTO, 2022)

Tabla 3.3: Entradas y Salidas del Módulo 2 (Módulo Sorting)

Entradas Módulo 2	Piezas
I124.0	Sensor óptico
I124.1	Sensor de proximidad inductivo
I124.2	Sensor de reflexión directa
I124.3	Sensor de retro reflexión
Salidas Módulo 2	Piezas
Q124.0	Banda transportadora pequeña

Q124.1	Módulo de compuerta clasificadora neumática 1
Q124.2	Módulo de compuerta clasificadora neumática 2
Q124.3	Módulo de borde

Direcciones de entradas y salidas del Módulo 2 (Estación de Clasificación), Fuente: (FESTO, 2022)

3.2.3.1 PLC S7 – 300 CPU 313C

El controlador Siemens SIMATIC PLC S7 – 300 CPU 313C (6ES7313 – 5BG04 – 0AB0), que se encuentra en el Módulo Sorting (Estación de Clasificación), controla una parte del módulo y por ello necesita otro para controlar el otro módulo, en la Tabla 3.4, se describe los parámetros más relevantes del dispositivo, de esta manera lograr controlar el módulo por completo.

Tabla 3.4: Características del PLC S7 – 300 CPU 313C

Siemens PLC S7 – 300 CPU 313C	
Características Técnicas	Valores o Rangos
Voltaje de alimentación	24 VDC
Voltaje de entrada señal digital	24 VDC
Voltaje de salida señal digital	24 VDC
Intensidad de alimentación	650 mA
Intensidad de entrada señal digital	80mA
Intensidad de salida señal digital	50mA
Cantidad de entradas digitales	24
Cantidad de salidas digitales	16
Interfaz	Interfaz RS 485
Protocolo	MPI

Características de operación del Simatic Siemens PLC S7 – 300 con el CPU 313C (6ES7313 – 5BG04 – 0AB0), Fuente: (SIEMENS, 2022)

3.2.4 Descripción de los elementos seleccionados para el control del Módulo Sorting (Estación de Clasificación)

Para realizar el Gemelo Digital del Módulo Sorting, es entender la importancia del funcionamiento de cada uno de los elementos y mecanismos, que ayudara a la hora del modelamiento 3D y animar en Unity.

A continuación, se procederá a describir a cada uno de los componentes más relevantes para el funcionamiento del Módulo Sorting (Estación de Clasificación) físico y el Gemelo Digital del Módulo.

3.2.4.1 PLC S7 – 1200 CPU 1212C

El controlador PLC S7 – 1200 CPU 1212C (6ES7 212 – 1AE40 – 0XB0), es versátil para el control del Gemelo Digital tanto en el Módulo Sorting (Estación de Clasificación) físico y el Módulo en 3D, las características más relevantes del controlador se pueden apreciar en la Tabla 3.5.

Tabla 3.5: Características del PLC S7 – 1200 CPU 1212C

Siemens PLC S7 – 1200 CPU 1212C	
Características Técnicas	Valores o Rangos
Voltaje de alimentación	24 VDC
Voltaje de entrada señal digital	24 VDC
Voltaje de salida señal digital	5 VDC
Intensidad de alimentación	400 mA
Nivel de entrada señal digital	2.5 mA
Nivel de salida señal digital	1mA
Cantidad de entradas digitales	8
Cantidad de salidas digitales	6
Interfaz	RJ45 Ethernet
Protocolo	Ethernet TCP/IP

Características de operación del Simatic Siemens PLC S7 – 1200 con el CPU 1212C (6ES7 212 – 1AE40 – 0XB0), Fuente: (Masvoltaje, 2020)

3.2.4.2 Comparativa entre el PLC S7 – 300 y PLC S7 – 1200

En el momento de la obtención las direcciones del Módulo Sorting (Estación de Clasificación), se obtuvo que el PLC S7 – 300 CPU 313C (6ES7313 – 5BG04 – 0AB0), dispone de una interfaz RS 485, la cual no es compatible para realizar la comunicación del Gemelo Digital del módulo físico y el virtual, por ello se buscó un controlador que contenga las prestaciones necesarias, una interfaz RJ45 Ethernet entradas y salidas digitales a 24VDC, el controlador que cumple estas características es el PLC S7 – 1200 CPU 1212C (6ES7 212 – 1AE40 – 0XB0), (ver Tabla 3.6) por ello se optó por adquirir el controlador para la implementación del Gemelo Digital.

Tabla 3.6 Diferencias de PLC S7 – 300 y PLC S7 – 1200

PLC S7 – 300 CPU 313C		PLC S7 – 1200 CPU 1212C	
Interfaz	Interfaz RS 485	Interfaz	RJ45 Ethernet
Protocolo	MPI	Protocolo	Ethernet TCP/IP

Diferencias entre interfaces y protocolos de comunicación, Elaborado por: Autores

3.2.4.3 Comparativa del direccionamiento de entradas y salidas del PLC S7 – 300 al PLC S7 – 1200

En la Tabla 3.7 y Tabla 3.8 se aprecia el direccionamiento del módulo 1 y módulo 2 de los dos PLC S7 – 300, en el que se observa sus entradas y salidas, de igual manera muestra el direccionamiento para el PLC S7 – 1200 con su módulo de expansión (SM 1223 6ES7 223 – 1BL32 – 0XB0) de entradas y salidas digitales, lo cual permite optimizar el Módulo de Control diseñado.

Tabla 3.7: Comparativa de entradas y Salidas del Módulo 1(Módulo Sorting)

Entradas Módulo 1 (PLC S7-300)	Entradas Módulo de Control (PLC S7-1200)	Piezas
I124.0	I9.0	Sensor de reflexión directa
I124.1	I9.1	Sensor izquierdo barra
I124.2	I9.2	Sensor medio barra
I124.3	I9.3	Sensor final barra
I124.4	I9.4	Sensor abajo pinza
I124.5	I9.5	Sensor arriba pinza
I124.6	I9.6	Sensor pinza abre/cierra

Salidas Módulo 1 (PLC S7-300)	Salidas Módulo de Control (PLC S7-1200)	Piezas
Q124.0	Q9.0	Actuador izquierdo brazo
Q124.1	Q9.1	Actuador derecho brazo
Q124.2	Q9.2	Actuador pistón pinza
Q124.3	Q9.3	Actuador garra pinza

Direcciones de entradas y salidas del Módulo 1 y su equivalencia para el Módulo de Control, Elaborado por: Autores

Tabla 3.8: Comparativa de entradas y salidas del Módulo 2 (Módulo Sorting)

Entradas Módulo 2 (PLC S7-300)	Entradas Módulo de Control (PLC S7-1200)	Piezas
I124.0	I8.0	Sensor óptico
I124.1	I8.1	Sensor de proximidad inductivo
I124.2	I8.2	Sensor de reflexión directa
I124.3	I8.3	Sensor de retro reflexión
Salidas Módulo 2 (PLC S7-300)	Salidas Módulo de Control (PLC S7-1200)	Piezas
Q124.0	Q8.0	Banda transportadora pequeña
Q124.1	Q8.1	Módulo de puerta clasificadora neumática 1
Q124.2	Q8.2	Módulo de puerta clasificadora neumática 2
Q124.3	Q8.3	Módulo de tope

Direcciones de entradas y salidas del Módulo 2 y su equivalencia del Módulo de Control, Elaborado por: Autores

3.2.4.4 Módulo Digital SM 1223 (6ES7 223 – 1BL32 – 0XB0)

En la Tabla 3.9 se aprecia las características de los rangos de valores del Módulo digital SM 1223 (6ES7 223 – 1BL32 – 0XB0), en donde serán conectadas las direcciones (Entradas y Salidas), del Módulo Sorting, esto es a través de los interfaces SysLink (MPS).

Tabla 3.9: Características del Módulo de expansión digital SM 1223

Módulo de expansión digital SM 1223	
Características Técnicas	Valores o Rangos
Voltaje de alimentación	24 VDC
Intensidad de entrada	185 mA
Voltaje para señal de salida “0”	0.1 VDC
Voltaje para señal de salida “1”	20 VDC
Intensidad para señal de salida “0”	0.1 VDC
Intensidad para señal de salida “1”	20 VDC
Entradas digitales	16
Salidas digitales	16

Características del Módulo Digital SM 1223 modelo 16ED/SD, Fuente: (Masvoltaje, 2020)

3.2.4.5 Arduino Mega 2560

El microcontrolador Arduino Mega 2560 es muy importante en este proyecto ya que es el medio de comunicación entre el Modelo Virtual y la Red Modbus TCP/IP de donde se recibe los datos del PLC S7-1200.

En la Tabla 3.10 se puede observar las características de este Microcontrolador.

Tabla 3.10: Características del Arduino Mega 2560

Arduino Mega 2560	
Características Técnicas	Valores o Rangos
Microcontrolador	ATmega2560
Voltaje de trabajo	5V

Voltaje de entrada (recomendada)	7 – 12 V
Voltaje de entrada (límite)	6 – 20 V
Terminales digitales I/O	54 (15 proporcionan salida PWM)
Terminales de entradas analógicas	16
Memoria flash	256 kb
SRAM	8 kb
Memoria EEPROM	4 kb
Velocidad de reloj	16 MHz

Principales características del microcontrolador ATmega 2560, Fuente: (Arduino, 2022)

3.2.4.6 Ethernet Shield

La placa Ethernet Shield tiene la capacidad de conectar con distintos modelos de Arduino, para este caso es un Mega 2560, con ello para crear una red ethernet protocolo TCP/IP, esto permitirá comunicar entre el PLC S7 – 1200. Para ello se debe tener presente las características de la placa como se puede observar en la Tabla 3.11.

Tabla 3.11: Características de la placa Ethernet Shield

Ethernet Shield	
Características Técnicas	Valores o Rangos
Voltaje de operación	5 VDC
Chip Ethernet	Wiznet W5100
Velocidad de Conexión	10/100 Mbps
Estándar	RJ45
Interfaz	SPI
Compatibilidad	Arduino Uno/Mega/Leonardo

Principales características de la placa Ethernet Shield, Fuente: (Naylamp, <https://naylampmechatronics.com/ardusystem-shields/12-shield-ethernet-w5100.html>, 2022)

3.2.4.7 Fuente Conmutada

Es una fuente de alimentación conmutada, encargada de convertir voltaje alterno (VAC) a voltaje continuo (VDC), la cual permitirá en dar funcionamiento al Módulo de Control en su totalidad, como se puede observar en la Tabla 3.12, las características que necesita

el Módulo de Control para dar funcionamiento el Módulo Sorting (Estación de Clasificación).

Tabla 3.12: Características de la Fuente Conmutada

Fuente Conmutada	
Características Técnicas	Valores o Rangos
Voltaje de entrada	200V – 240VAC (60 Hz)
Tensión de salida	24 V
Corriente de salida máxima	10 a
Energía máxima	50 W
Eficiencia de salida	83%
Temperatura de operación	-10 ⁰ C hasta +60 ⁰ C

Principales características de la Fuente Conmutada, Fuente: (UNIT, 2022)

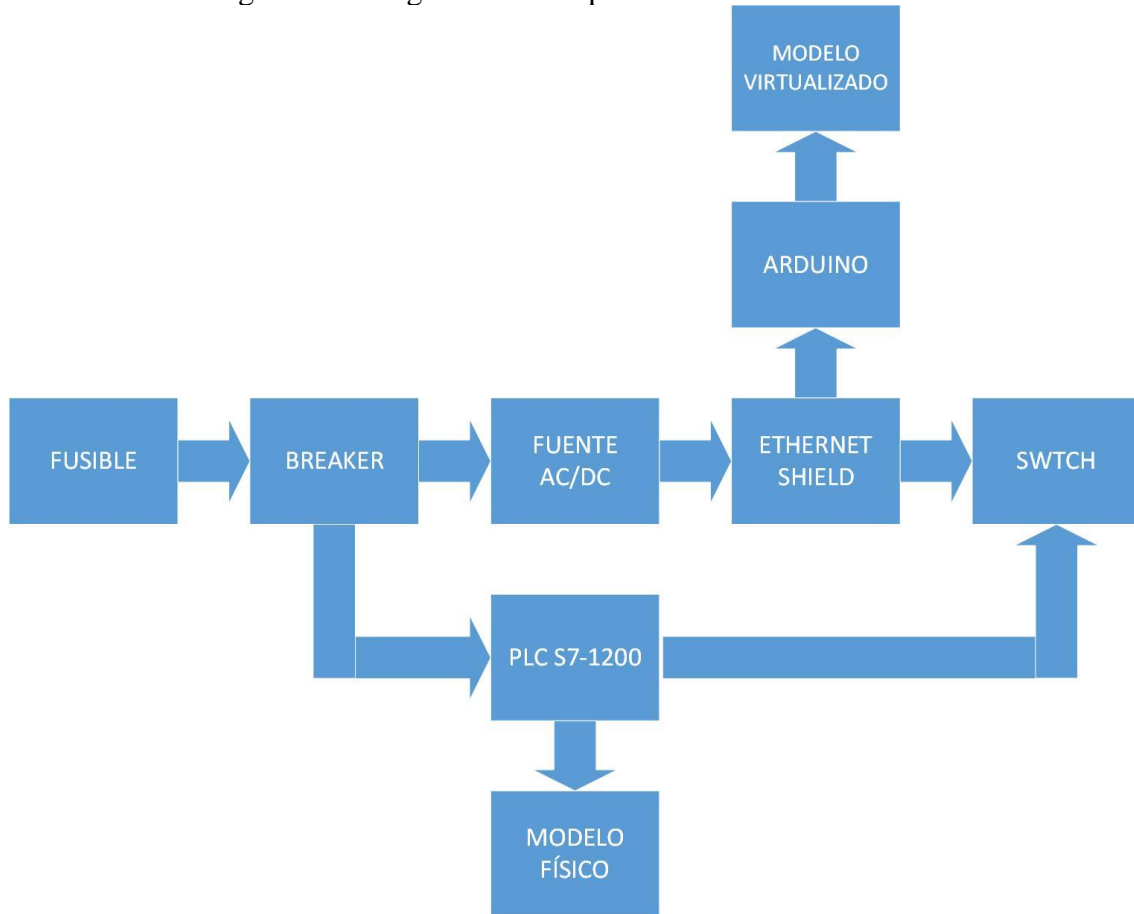
3.2.5 Esquema eléctrico del Módulo de Control

La representación eléctrica (ver Anexo 1) se puede apreciar la conexión de la fuente de alimentación (110 VAC), que va conectado a un break para poder proteger el Módulo de Control, si se produjera una sobrecarga en el suministro de energía, a continuación, se conecta a una fuente de alimentación de CA/CC (110 VAC/24 VDC), ya que el PLC S7 – 1200 y Módulo Digital SM 1223, funcionan a 24VDC, y para finalizar la conexión, los Interfaces SysLink (MPS), en el Módulo de expansión (Módulo Digital SM 1223).

3.2.6 Diseño del Módulo de Control

El diseño del Módulo de Control (ver Figura 3.4) comprende un conjunto de componentes que son necesarios, para el funcionamiento del Módulo Sorting, y a su vez permite enviar los datos que se obtienen por los sensores y actuadores los que posteriormente serán utilizados en el control del modelo virtualizado del Módulo Sorting en el software Unity3D.

Figura 3.4: Diagrama de Bloques del Módulo de Control



Elaborado por: Autores

3.3 Programación en Arduino

El microcontrolador Arduino va a ser el encargado de la codificación y decodificación de los paquetes Modbus TCP/IP, para ello es necesario seguir ciertos pasos para que pueda realizar esta función, los cuales se describe a continuación.

- Primer paso es la inclusión de la librería Ethernet, y Modbus en el software de Arduino como se puede apreciar en la Figura 3.5.

Figura 3.5: Inclusión de las librerías Ethernet y Modbus

```
#include <SPI.h>
#include <Ethernet.h>
#include <Modbus.h>
#include <ModbusIP.h>
```

Elaborado por: Autores

- Segundo paso se realiza la creación de variables, con su respectiva dirección del Protocolo Modbus, se puede apreciar en la Figura 3.6, donde el recuadro de color

rojo es para el envío de datos y de igual manera el recuadro de color verde es para la recepción de datos.

Figura 3.6: Declaraciones de constantes en Arduino

```
const int ENVIAR_DT1=2;  
const int ENVIAR_DT2=3;  
const int ENVIAR_DT3=4;  
const int ENVIAR_DT4=5;  
const int ENVIAR_DT5=6;  
const int ENVIAR_DT6=7;  
const int ENVIAR_DT7=8;  
const int ENVIAR_DT8=9;  
  
const int RECIB_DT1= 10;  
const int RECIB_DT2= 11;  
const int RECIB_DT3= 12;  
const int RECIB_DT4= 13;  
const int RECIB_DT5= 14;  
const int RECIB_DT6= 15;  
const int RECIB_DT7= 16;
```

Elaborado por: Autores

- Tercer paso se puede apreciar en la Figura 3.7, la asignación de la dirección MAC de la placa Ethernet Shield, la dirección ip de la red, la dirección Gateway es la puerta de enlace para el Switch de la red y por último la máscara de la subred.

Figura 3.7: Configuración de la red Ethernet

```
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};  
IPAddress ip(192, 168, 100, 88);  
IPAddress gateway(192, 168, 100, 1);  
IPAddress subnet(255, 255, 255, 0);
```

Elaborado por: Autores

- Cuarto paso se inicializa los registros en la función void setup (), como se puede apreciar en la Figura 3.8, el recuadro de color azul son los valores de cada registro serán almacenados que posteriormente enviaran de Unity hacia el PLC, y de igual manera el recuadro de color café son los registros que son recepción de datos del PLC hacia Unity.

Figura 3.8: Inicialización de los registros de envío y recepción de datos

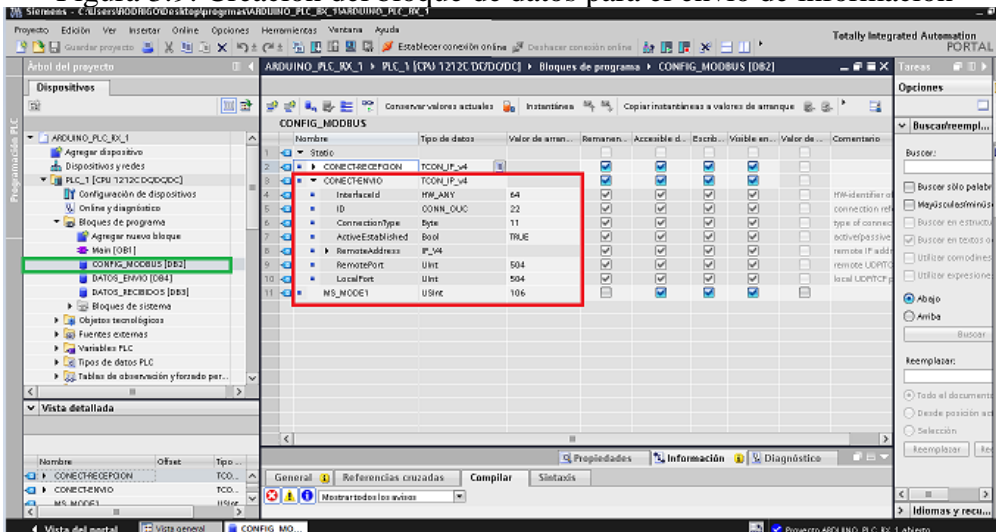
<pre>mb.addHreg(ENVIAR_DT1); delay(200); mb.addHreg(ENVIAR_DT2); delay(200); mb.addHreg(ENVIAR_DT3); delay(200); mb.addHreg(ENVIAR_DT4); delay(200); mb.addHreg(ENVIAR_DT5); delay(200); mb.addHreg(ENVIAR_DT6); delay(200); mb.addHreg(ENVIAR_DT7); delay(200); mb.addHreg(ENVIAR_DT8); delay(200);</pre>	<pre>mb.addHreg(RECIB_DT1); delay(200); mb.addHreg(RECIB_DT2); delay(200); mb.addHreg(RECIB_DT3); delay(200); mb.addHreg(RECIB_DT4); delay(200); mb.addHreg(RECIB_DT5); delay(200); mb.addHreg(RECIB_DT6); delay(200); mb.addHreg(RECIB_DT7); delay(200);</pre>
--	---

Elaborado por: Autores

3.4 Programación en TIA Portal

En el algoritmo del software TIA Portal, se realiza la configuración de un bloque de datos (CONFIG_MOBUS), en el que se configuran los parámetros de la comunicación Modbus, (ver Figura 3.9), como se puede apreciar en el recuadro de color rojo, la pestaña CONECT-ENVIO.

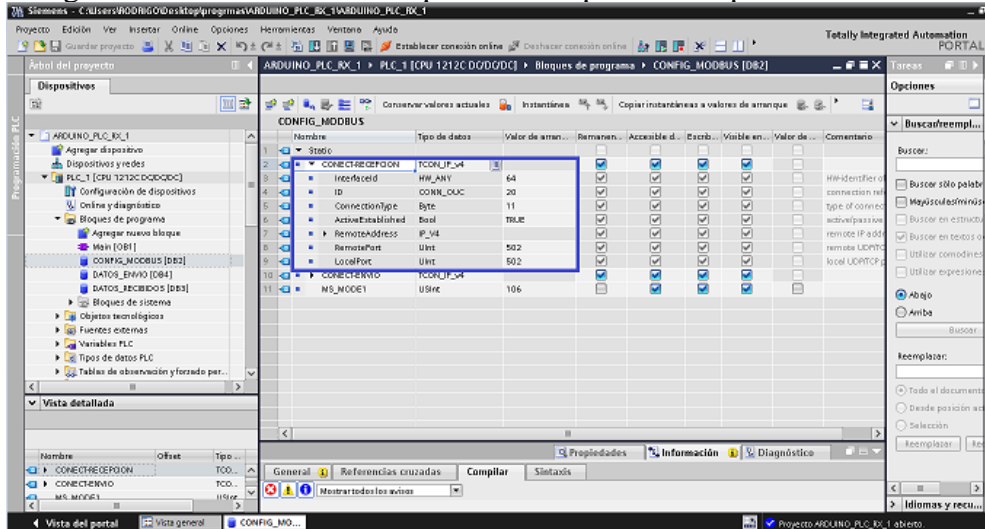
Figura 3.9: Creación del bloque de datos para el envío de información



Elaborado por: Autores

Como se puede observar en la Figura 3.10, en recuadro de color azul la pestaña CONET – RECEPCION, de igual manera se establecen los parámetros para la declaración Modbus.

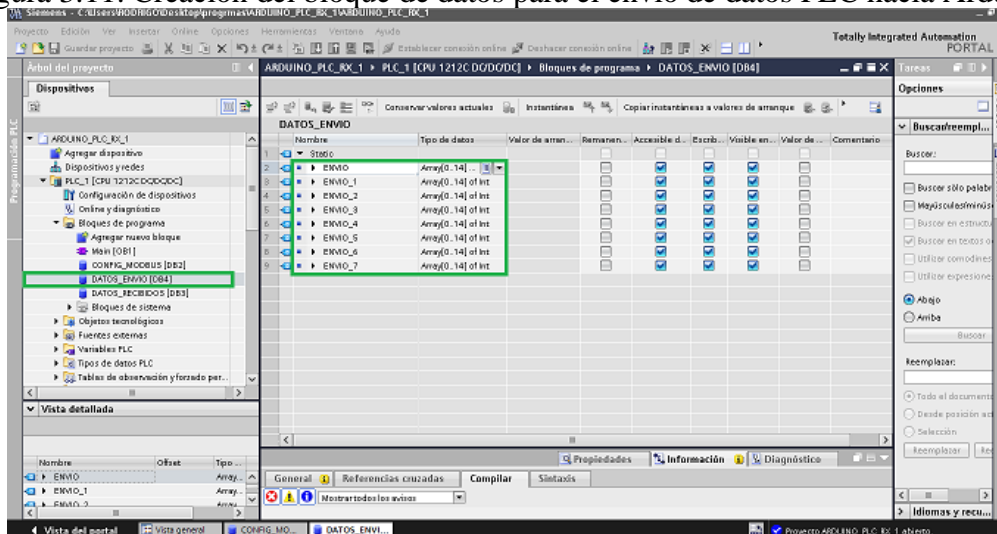
Figura 3.10: Creación del bloque de datos para la recepción información



Elaborado por: Autores

En la Figura 3.11 podemos notar dos recuadros de color verde, en donde se configura un bloque de datos (DB), el cual es el encargado de enviar los datos obtenidos del PLC S7 – 1200 para que puedan ser procesados por el microcontrolador Arduino Mega 2560, donde serán guardados en cada una de las direcciones y utilizadas en el programa.

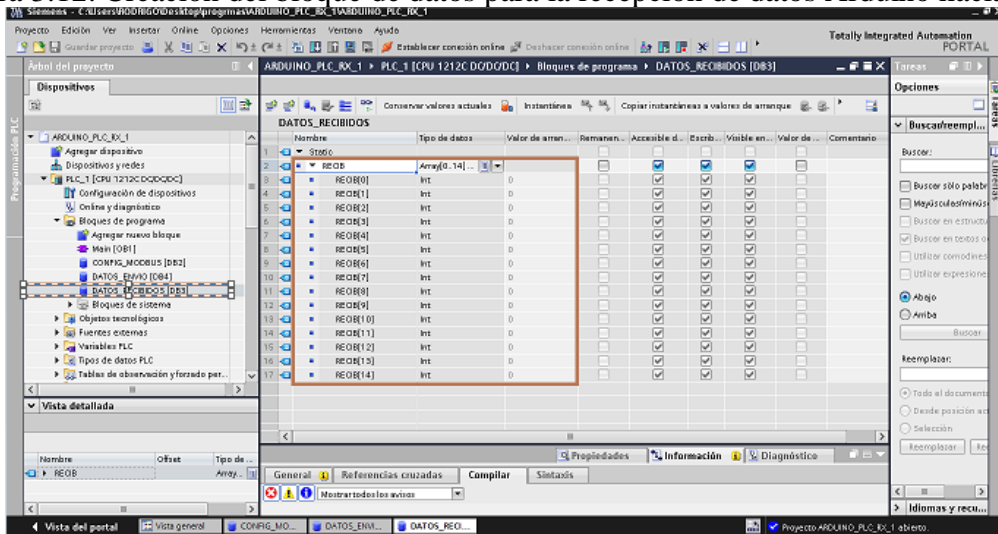
Figura 3.11: Creación del bloque de datos para el envío de datos PLC hacia Arduino



Elaborado por: Autores

En la Figura 3.12 el bloque de DATOS-RECIBIDOS (DB), que se encuentra remarcado de color café, son las direcciones que se encargan en recibir y guardar los datos del microcontrolador Arduino Mega 2560 y posteriormente enviados hacia el PLC S7 – 1200, para ser utilizados por el programa.

Figura 3.12: Creación del bloque de datos para la recepción de datos Arduino hacia PLC



Elaborado por: Autores

3.5 Programación en Unity

En este apartado se efectúa la programación en Unity, a través del MonoDevelop Visual Studio, se inicia en la creación de una variable SerialPort para la asignación del COM de Arduino y velocidad de la comunicación del puerto, y por último la declaración de variables donde almacenaran los datos enviados del PLC a Unity.

En la función void Start (), es aquí donde se comprueba si el puerto serial de Arduino está abierto o cerrado y el tiempo de espera, para el envío de datos y de recepción.

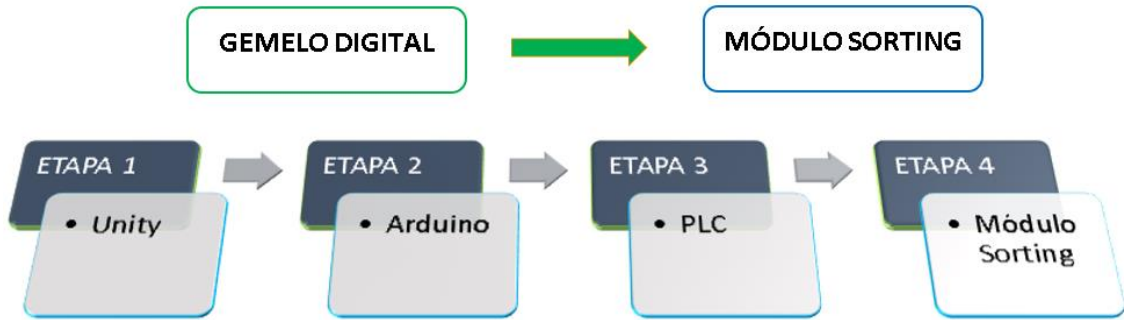
Y por último la función void Update (), efectúa un ciclo repetitivo constantemente, comprobando si el puerto serial está abierto, el recuadro de color rojo indica la recepción de todos los datos enviados del PLC, en el recuadro de color verde se realiza la separación de los datos y posteriormente ser utilizados en el proceso.

3.6 Comunicación entre el Módulo Sorting (Estación de Clasificación) y el Módulo virtualizado en 3D

Aquí se explica el proceso del envío y recepción de datos del Gemelo Digital hacia el Módulo Sorting físico, las distintas etapas que conlleva la comunicación entre ambos módulos.

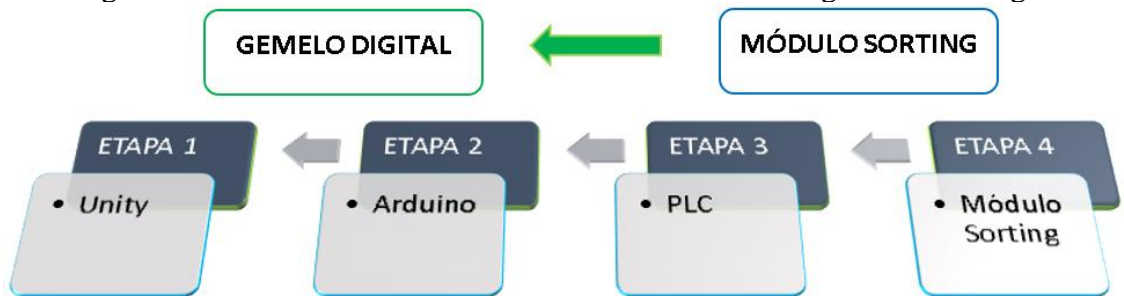
A continuación, en las Figuras 3.13 y 3.14, se describe cada una de las etapas que conlleva lograr la comunicación del Gemelo Digital y el Módulo Sorting.

Figura 3.13: Proceso del envío de datos del Gemelo Digital – Módulo Sorting



Fuente: Autores

Figura 3.14: Proceso del envío de datos del módulo Sorting – Gemelo Digital



Fuente: Autores

CAPÍTULO 4

ANÁLISIS Y RESULTADOS

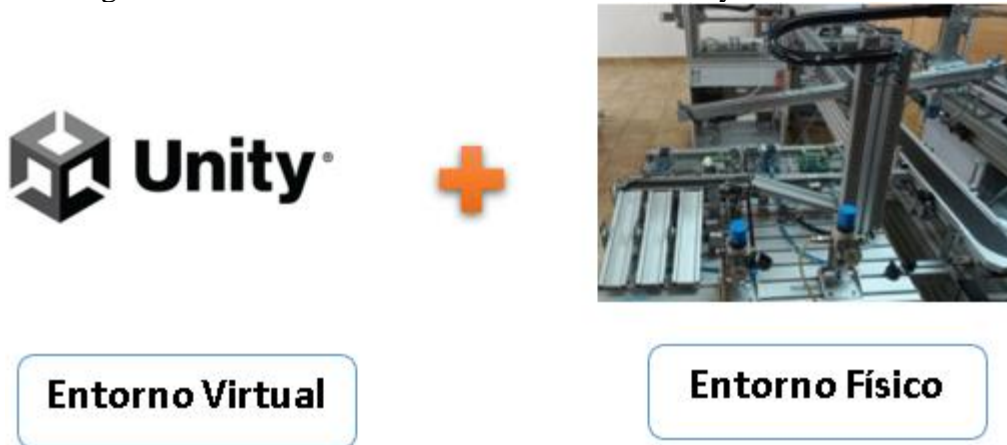
4.1 Comunicación

En el capítulo 4, se abordará el tema referente a las comunicaciones empleadas para el desarrollo e implantación del gemelo digital para el módulo Sorting, a continuación, se detalla los procesos y protocolos empleados durante el desarrollo del proyecto.

4.1.1 Comunicación entre Hardware y Software

Se define como Hardware al soporte físico, y como Software al soporte operacional, (ver Figura 4.1) siendo estos dos elementos indispensables en el funcionamiento de los dispositivos electrónicos empleados en el presente proyecto.

Figura 4.1: Comunicación entre el entorno virtual y el entorno físico



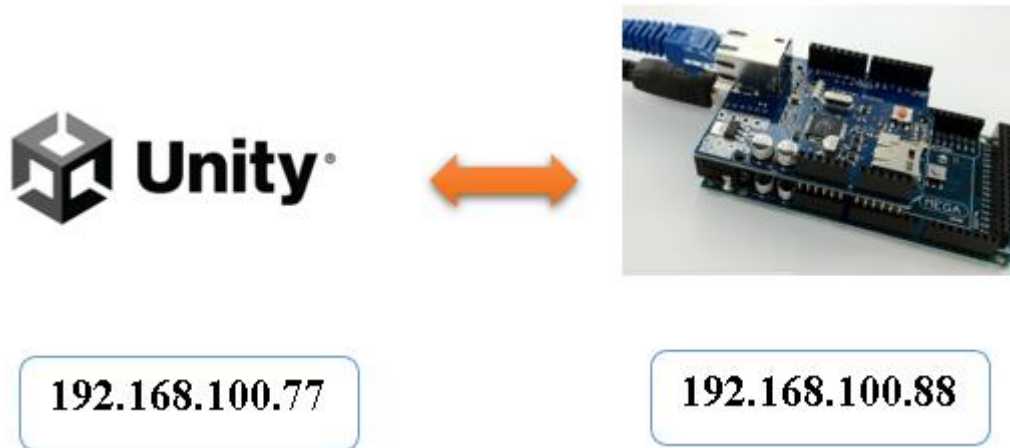
Fuente: Autores

4.2 Tipos de Comunicación

4.2.1 Comunicación Tipo 1

Entre el software (Unity 3D) y la tarjeta de desarrollo (Arduino Mega), desarrollada la interfaz gráfica para la interpretación de los movimientos en Unity 3D, usando como desarrollador visual studio dentro del código de lenguaje de programación C#, siendo Arduino el encargado de procesar la información de recepción y envió.

Figura 4.2: Comunicación bidireccional entre Unity y Arduino

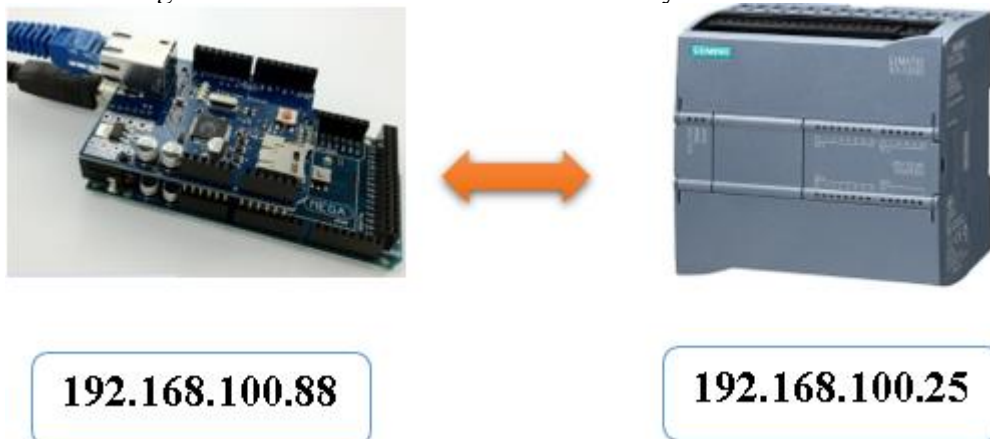


Fuente: Autores

4.2.2 Comunicación Tipo 2

Para la comunicación entre el controlador (PLC S7 – 1200) y la tarjeta de desarrollo (Arduino Mega), de la información de adquisición de sensores y la operación de actuadores, programadas en lenguaje Ladder para el PLC, se enviarán al microcontrolador Arduino el cual se encargará de realizar la lectura y escritura de los registros de información almacenadas en las direcciones modbus, para la operación general del sistema.

Figura 4.3: Comunicación entre Arduino y PLC S7 – 1200

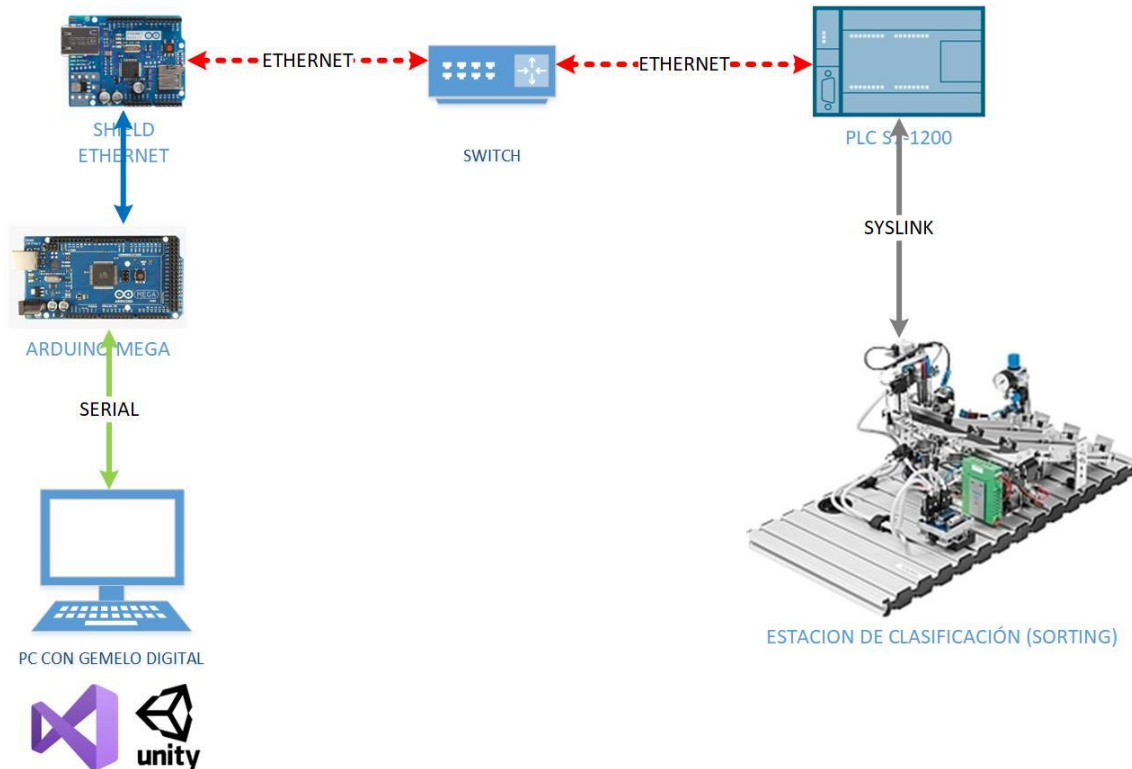


Fuente: Autores

4.3 Protocolos de Red Ethernet

Ethernet estándar de comunicación entre los dispositivos físicos, dentro del área local. Empleada en el envío y recepción de información, entre el hardware y el software, permitiendo acceder y controlar a nivel industrial el sistema automático.

Figura 4.4: Comunicación de la Red Ethernet

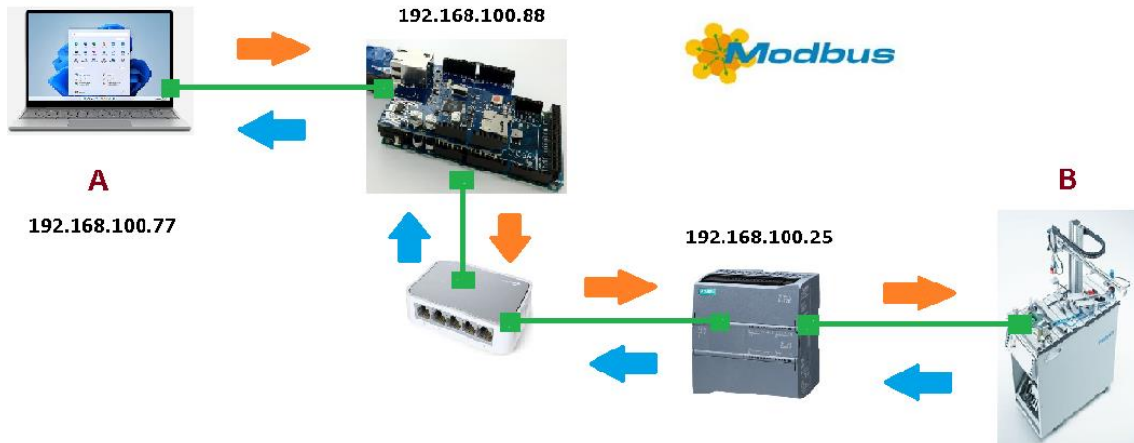


Fuente: Autores

4.4 Protocolo de Comunicación Modbus TCP/IP

Ya que Modbus es un protocolo de comunicación abierto, utilizado en la transmisión de información en redes industriales a dispositivos conectados. Realizando la comunicación en etapas entre los dispositivos, esto quiere decir que durante la comunicación se actúa de forma unidireccional, cuando A envía B solo recibe y terminada esta etapa B envía y A solo recibe.

Figura 4.5: Envío y recepción de datos de un punto A hacia un punto B



Fuente: Autores

Tabla 4.1: Direccionamiento de Componentes

Componente	Campo	Valor
PLC S7-1200	Dirección IP:	192.168.100.85
	Mascara de Red:	255.255.255.0
	Puerta de Enlace:	192.168.100.1
Arduino Mega (Shield Ethernet)	Dirección IP:	192.168.100.85
	Mascara de Red:	255.255.255.0
	Puerta de Enlace:	192.168.100.1

Elaborado por: Autores

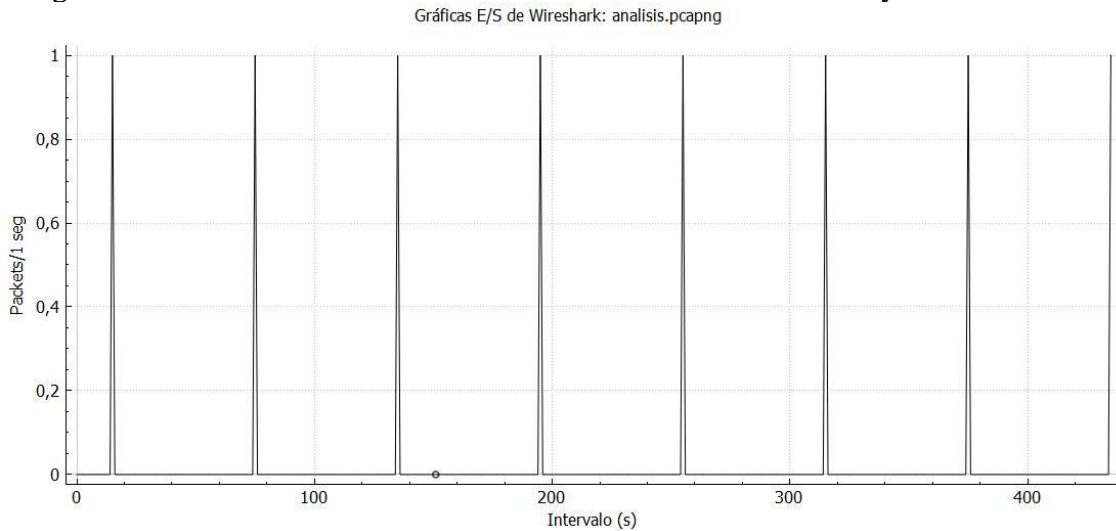
4.5 Análisis del tiempo de ejecución entre los Modelo Físico y Modelo Digital

Se realizó capturas de paquetes durante la transmisión de la información entre los modelos utilizando el software WireShark, el cual fue instalado en el computador donde se encuentra el módulo virtual, el computador se conecta a la red a través del puerto Ethernet para ser parte de la misma y que el software WireShark pueda realizar la captura de paquetes en toda la red, además se realizó la conexión de esta manera ya que se deseaba realizar la visualización del funcionamiento de las variables del PLC en tiempo real en el software TIA PORTAL, realizada la conexión de esta manera se pudo obtener los tiempos de entrega de paquetes entre los distintos elementos de la red Ethernet.

Para poder realizar el análisis se realizó la captura de paquetes durante 8 minutos, pasado este tiempo se detuvo la captura de paquetes para poder realizar su análisis, esto quiere

decir identificar el origen y destino de los paquetes con esto clasificar los mismos y generar graficas que permitan interpretar de manera más sencilla el tiempo que tardan en intercambiar información entre los distintos componentes de la red, en la Figura 4.6 se puede observar el intervalo de tiempo que toma en transportar un paquete entre el sistema microcontrolador (Arduino) y el controlador (PLC S7-1200) el cual es equivalente a un segundo.

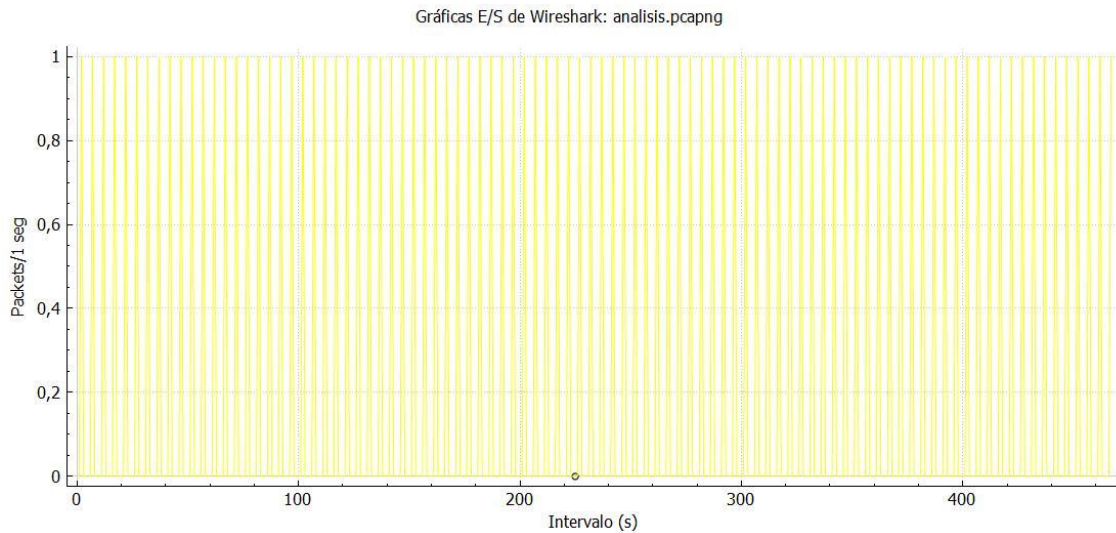
Figura 4.6: Grafica de Análisis de Comunicación entre el Arduino y el PLC S7-1200



Fuente: Autores

En otro análisis se puede observar (Figura 4.7) la comunicación que se establece en la otra dirección esto desde el PLC S7-1200 hacia el módulo Arduino, en el que se puede observar que el intervalo de tiempo es el mismo (1 segundo) pero el envío de información es más constante.

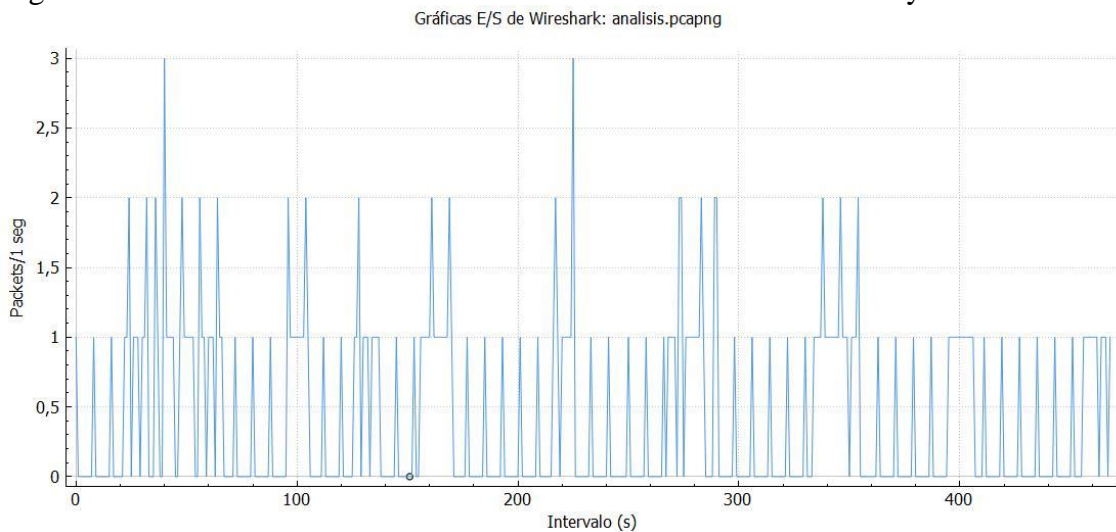
Figura 4.7: Grafica de Análisis de Comunicación entre el PLC S7-1200 y el Arduino



Fuente: Autores

En la Figura 4.8 que se muestra ac continuación se puede apreciar el tráfico de paquetes que fueron compartidos entre el computador y el PLC S7-1200 ya que estaba activo el software TIA PORTAL para observar la ejecución del programa en tiempo real, se nota que existen picos de comunicación de hasta 3 segundos ya que el flujo de datos que se comparte es mayor.

Figura 4.8: Grafica de Análisis de Comunicación entre TIA PORTAL y el PLC S7-1200

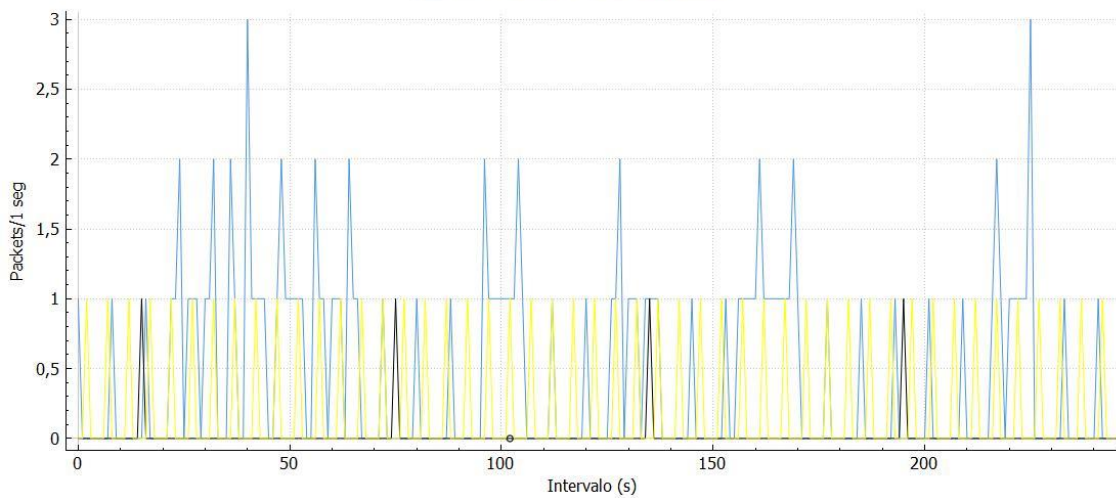


Fuente: Autores

Por último, se observa en la Figura 4.9 la comparación entre todos los paquetes que se compartieron durante la captura y se puede concluir que el mayor tiempo se encuentra en el intercambio entre el computador y el PLC S7-1200.

Figura 4.9: Grafica de Análisis de Comunicación

Gráficas E/S de Wireshark: analisis.pcapng



Fuente: Autores

En la Figura 4.10 que se muestra se observa un resumen global de la captura donde se puede apreciar el promedio del tamaño de los paquetes que se enviaron a la red.

Figura 4.10: Resumen de Captura de Datos WireShark

Estadísticas

<u>Medida</u>	<u>Capturado</u>	<u>Mostrado</u>
Paquetes	297	297 (100.0%)
Espacio de tiempo, s	468.041	468.041
Promedio pps	0.6	0.6
Promedio de tamaño de paquete, B	161	161
Bytes	47941	47941 (100.0%)
Promedio de bytes/s	102	102
Promedio de bits/s	819	819

Fuente: Autores

CONCLUSIONES

Se pudo implementar un Gemelo Digital del Modelo Sorting utilizando el protocolo de comunicación Ethernet, con el uso del software Unity 3D y Visual Studio se pudo dar movimiento al modelo digital además de recibir y enviar señales desde y hacia el modelo físico en este caso el Módulo Sorting de la MPS, para la gestión de la estación física se conectó el PLC S7-1200, gracias a las características del controlador se pudo lograr la comunicación mediante el protocolo ethernet conectándose mediante un switch y llegando al módulo digital a través de un Arduino Mega dotado de una Shield Ethernet que le permite ser parte de la red y recibir los datos del controlador PLC S7-1200.

Se pudo establecer la comunicación bidireccional entre los modelos físico y digital usando el protocolo de comunicación Ethernet Industrial y el uso del sistema embebido (Arduino Mega) para interpretar los mensajes enviados desde y hacia el controlador PLC S7-1200.

Utilizando el software WireShark se tomaron muestras del intercambio de información en la red con lo que se pudo establecer que el tiempo de traslado de los paquetes es de aproximadamente 1 segundo y el tamaño promedio de los paquetes es de 161 Bytes, también se pudo notar que la comunicación es más activa de parte del controlador ya que envía constantemente el estado de las variables (sensores y actuadores) hacia el modelo virtual.

RECOMENDACIONES

Usando la base de esta investigación se podría generar un sistema en el que se tengan algunas herramientas como los que se pueden observar partes de distintas estaciones del MPS con el fin de realizar prácticas u observar más a fondo las distintas formas de funcionamiento de las partes y adjuntarlas para simular una fábrica más completa.

Se podría además tomar esta investigación con el fin de crear sistemas SCADA que permitan la supervisión del estado de la fábrica y el reporte de errores mucho más detallados se puede también realizar la gestión vía remoto de partes básicas del funcionamiento de la fábrica o establecer soporte remoto más sencillo.

En la presente investigación lo que se pretende es mostrar el estado actual de los sensores y actuadores conectados pero además de esto se podría guardar fallas o ciertos datos que ayuden al diagnóstico en caso de que exista una falla esto mediante la implementación de bases de datos en donde se guarden datos críticos que alerten con cierto tiempo que es necesario la intervención de un operario el agendamiento de un mantenimiento preventivo, lo que reduciría el gasto en reparaciones catastróficas.

REFERENCIAS

- Anderson, C. (2018). Industria 4.0: La transformación digital en la industria. En J. M. Aguiló, *Industria 4.0: La transformación digital en la industria*.
- Arduino. (6 de Junio de 2022). *Arduino*. Obtenido de <https://store.arduino.cc/products/arduino-mega-2560-rev3>:
<https://store.arduino.cc/products/arduino-mega-2560-rev3>
- aula21. (5 de Mayo de 2022). *aula21 Centro de formación técnica para la industria*. Obtenido de <https://www.cursosaula21.com/>: <https://www.cursosaula21.com/>
- Betts, R. (2016). *Architecting for the Internet of Things* (1 ed.). New York: O'Reilly Media, Inc.
- Bollaín, M. (2018). *Ingeniería de Instrumentación de Plantas de Proceso*. España: DIAZ DE SANTOS.
- Boquera, M. C. (2005). *Comunicaciones Ópticas* (2 ed.). Madrid: Ediciones Díaz de Santos, S. A.
- Butler, J. (2013). *Redes Inalámbricas en los Países en Desarrollo* (4 ed.). Copenhagen: Perason Educación.
- Cabezas, A. F. (2014). *Sistemas de comunicaciones ópticas* (2 ed.). Bogotá D.C.
- Castillo, J. (2019). *Infraestructuras comunes de telecomunicación en viviendas y edificios*. Mexico: Editex.
- Castro, M. A. (2005). *Sistemas de Cableado Estructurado* (1 ed.). Madrid: Ra-Ma S.A. Editorial y Publicaciones.
- Chomycz, B. (1998). *Instalaciones de fibra óptica :fundamentos, técnicas y aplicaciones* (2 ed.). Madrid: McGraw-Hill.
- Córcoles, S., & Muñoz, A. (2018). *Arduino Curso Práctico*. Madrid: Ra-Ma.
- Cordero Pedraza, C. (2007). *Fundamentos de redes* (4 ed.). México: McGraw-Hill/Interamericana Editores, S.A. de C.V.
- Didactic, F. (Noviembre de 2015). https://ip.festo-didactic.com/InfoPortal/DataSheets/INT/8046325_deenesfr_v2.01_LP8046391_MPS-D_Sorting_station_Manual.pdf. Obtenido de https://ip.festo-didactic.com/InfoPortal/DataSheets/INT/8046325_deenesfr_v2.01_LP8046391_MPS-D_Sorting_station_Manual.pdf
- Dumenjo, S. S. (2012). *Guía de Sistemas de Cableado Estructurado* (2 ed.). México: Experiencias Ediciones.

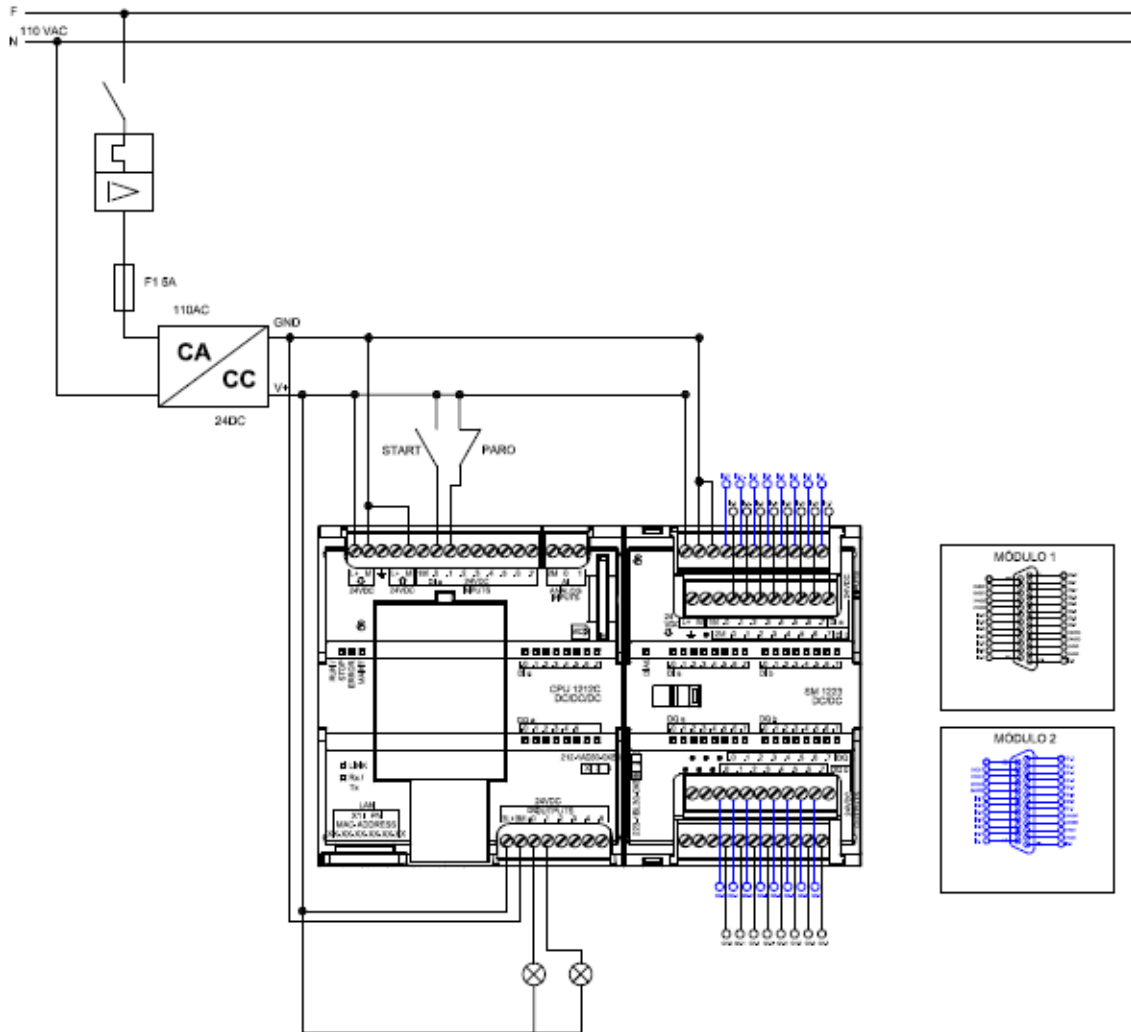
- Fernández, R. (2009). *Redes Inalámbricas de Sensores Teoría y aplicación práctica* (1 ed.). España: Grupo de investigación EDMANS.
- Fernández, Y. (3 de Agosto de 2020). *Xataka*. Obtenido de <https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>: <https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>
- Festo. (5 de Mayo de 2022). *Festo*. Obtenido de https://www.festo.com/es/es/e/tendencias/tia-portal-id_828990/: https://www.festo.com/es/es/e/tendencias/tia-portal-id_828990/
- FESTO. (4 de Junio de 2022). <https://www.festo-didactic.com/es-es/productos/?fbid=ZXMuZXMuNTQ3LjE0LjE5LjMzMzMDA>. Obtenido de FESTO: <https://www.festo-didactic.com/es-es/productos/?fbid=ZXMuZXMuNTQ3LjE0LjE5LjMzMzMDA>
- Gómez González, S. (2008). *El gran libro de Solidworks*. Barcelona - España: Marcombo, S.A.
- Gonzales, M. (08 de 11 de 2013). *Redes Telemáticas*. Obtenido de Redes Telemáticas: <https://redestelematicas.com/el-switch-como-funciona-y-sus-principales-caracteristicas/>
- Guerrero, V., Martínez, L., & Yuste, R. L. (2009). *Comunicaciones Industriales*. Barcelona: MARCOMBO, S.A.
- Gutiérrez, J. M. (2013). *Arduino + Ethernet Shield* .
- Hanes, D. (2017). *IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things* (2 ed.). USA: Cisco Press.
- Hu, F. (2016). *Security and Privacy in Internet of Things (IoTs)* (1 ed.). New York: Taylor & Francis Group.
- Jaume Martínez, A. (2019). *Industrial 4.0: La transformación digital en la industria*. México: Alfaomega.
- Kranz, M. (2017). *Internet of Things* (1 ed.). Madrid: Lid Editorial.
- Kurose, J. (1995). *Redes de Computadoras* (5 ed.). México: Pearson Educación.
- Lidon, M. (2019). *Unity 3D*. México: Alfaomega.
- Lidon, M. (2019). *UNITY 3D*. Barcelona: Alfaomega.
- Masvoltaje, T. (28 de Junio de 2020). *Masvoltaje Tu tienda de electricidad*. Obtenido de <https://masvoltaje.com/>: <https://masvoltaje.com/>

- McEwen, A. (2014). *Designing the internet of things* (1 ed.). USA: Chichester Wiley.
- Merkel, A. (2018). Industrial 4.0: La transformación digital en la industria. En J. M. Aguiló, *Industrial 4.0: La transformación digital en la industria*. Alfaomega.
- Microsoft. (21 de 06 de 2022). *Microsoft Docs*. Obtenido de <https://docs.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2022>
- Mountford, P. (24 de 2 de 2019). *Riverbed Modeler*. Obtenido de <https://www.riverbed.com/mx/products/steelcentral/steelcentral-riverbed-modeler.html>
- Naylamp, M. (6 de Junio de 2022). <https://naylampmechatronics.com/ardusystem-shields/12-shield-ethernet-w5100.html>. Obtenido de Naylamp Mechatronics: <https://naylampmechatronics.com/ardusystem-shields/12-shield-ethernet-w5100.html>
- Naylamp, M. (6 de Junio de 2022). <https://naylampmechatronics.com/fuentes-switching-ac-dc/795-fuente-de-alimentacion-conmutada-acdc-50w-24v-2a-wode.html>. Obtenido de Naylamp Mechatronics: <https://naylampmechatronics.com/fuentes-switching-ac-dc/795-fuente-de-alimentacion-conmutada-acdc-50w-24v-2a-wode.html>
- Ouazzani, I. (2012). *Manual de creación de videojuegos con Unity3D*. Madrid: Universidad Carlos III de Madrid.
- Peñaloza, A. G. (2015). *Diseño y Planificación de Proyectos de Cableado Estructurado* (1 ed.). Madrid: Kindle.
- Rodríguez, L. D., & Quinchiguango De la cruz, L. E. (2021). *IMPLEMENTACIÓN DE I/O ANALÓGICAS Y DIGITALES EN LOS MÓDULOS DIDÁCTICOS PLC'S DEL LTI-ESFOT*. Quito.
- Rodríguez, N. (2018). Uso de nuevas tecnologías para la competitividad del país. *TEC*, 33.
- Rodríguez, N. (2018). Uso de nuevas tecnologías para la competitividad del país. *TEC*, 33.
- Santuario, A. (2020). Educación y pandemia: Una visión académica. *Universidad Nacional Autónoma de México*, 75-82.
- Santuario, A. (2020). Educación y pandemia: Una visión académica. *Universidad Nacional Autónoma de México*, 75 - 82.
- SIEMENS. (10 de Junio de 2022). <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7313-5BG04-0AB0>. Obtenido de SIEMENS: <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7313-5BG04-0AB0>

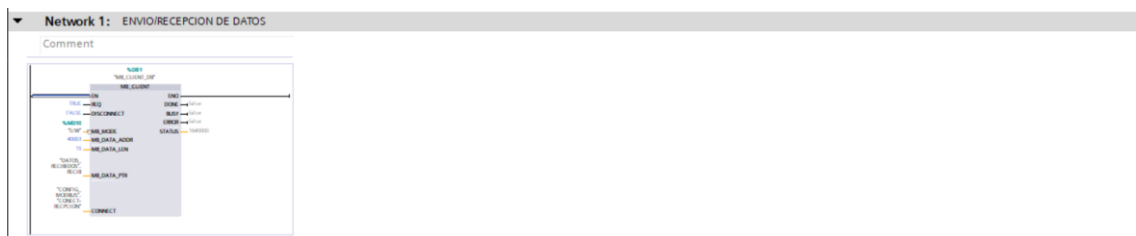
- Sink, P. (2001). Industrial Ethernet for the real world takes shape: The convergence of embedded devices, TCP/IP, and factory automation. *CompactPCI Systems*, 2.
- Stallings, W. (2016). *Foundations of Modern Networking SDN, NFV, QoE, IoT, and Cloud* (1 ed.). USA: Pearson Education.
- Stallings, William. (2016). *Foundations of Modern Networking SDN, NFV, QoE, IoT, and Cloud*. USA: Pearson Education.
- Takahashi. (2022). La Nueva Revolución Digital para Principiantes: Usos prácticos del Metaverso. En *La Nueva Revolución Digital para Principiantes: Usos prácticos del Metaverso*. Comprehensive Technological Research.
- Tanenbaum, A. (2012). *Redes de Computadoras* (5 ed.). México: Pearson Educación.
- UNIT, E. (6 de Junio de 2022). <https://uelectronics.com/producto/fuente-conmutada-24v-10a/>. Obtenido de UNIT ELECTRONICS: <https://uelectronics.com/producto/fuente-conmutada-24v-10a/>
- Vázquez, R. N. (5 de 6 de 2018). *beBee producir*. Obtenido de <https://www.bebec.com/producir/@rafael-nunez-vazquez/10-consideraciones-en-la-construccion-de-edificios-inteligentes-con-internet-de-las-cosas-iot>

ANEXOS

Anexo 1

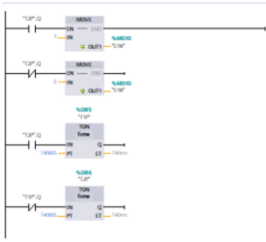


Anexo 2



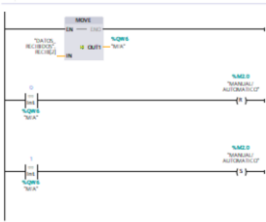
Network 2: TIEMPO DE ENVIO

Comment



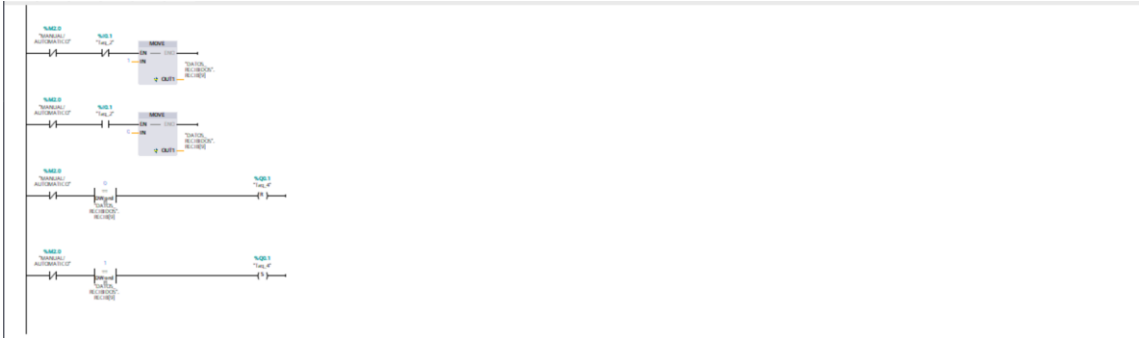
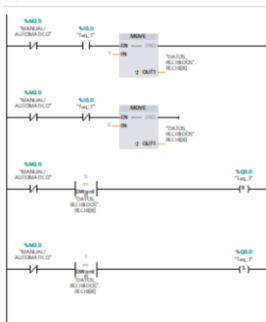
Network 3: SELECTOR MODO DE OPERACION MANUAL / AUTOMATICO

Comment



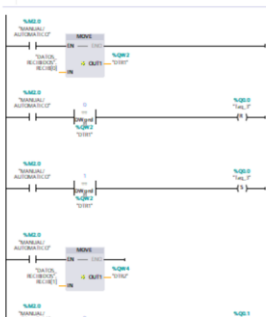
Network 4: ENVIO DE DATOS AL ARDUINO

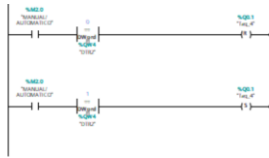
Comment



Network 5: RECEPCION DE DE DATOS DESDE ARDUINO

Comment





Anexo 3

CONFIG_MODBUS									
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	CONECT-RECPCION	TCON_IP_v4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	CONECT-ENVIO	TCON_IP_v4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	MS_MODE1	USInt	106	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

DATOS_RECIBIDOS									
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	RECIB	Array[0..14] ...		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Anexo 4

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using System.IO.Ports;
5
6  public class Comunicacion : MonoBehaviour
7  {
8      SerialPort stream = new SerialPort("COM11", 115200);
9
10     public int I0_MD1;
11     public int I1_MD1;
12     public int I2_MD1;
13     public int I3_MD1;
14     public int I4_MD1;
15     public int I5_MD1;
16     public int I6_MD1;
17     public int Q0_MD1;
18     public int Q1_MD1;
19     public int Q2_MD1;
20     public int Q3_MD1;
21
22     public int I0_MD2;
23     public int I1_MD2;
24     public int I2_MD2;
25     public int I3_MD2;
26     public int Q0_MD2;
27     public int Q1_MD2;
28     public int Q2_MD2;
29     public int Q3_MD2;
30
31     public int cont;
32     public int MDPT;
33     public int aux;
34     public int aux1;
35

```

```

36     // Start is called before the first frame update
37     void Start()
38     {
39         stream.Open(); //Abrimos una nueva conexión de puerto serie
40         stream.ReadTimeout = 1; //Establ
41     }
42
43     // Update is called once per frame
44     void Update()
45     {
46         if (stream.IsOpen) //comprobamos que el puerto esta abierto
47         {
48             try //utilizamos el bloque try/catch para detectar una posible excepción.
49             {
50                 string value = stream.ReadLine(); //leemos una línea del puerto serie y la almacenamos en un string
51                 string[] vec6 = value.Split(';'); //Separamos el String leído valiendonos
52
53                 I0_MD1 = int.Parse(vec6[0]); //de las comas y almacenamos los valores en un array.
54                 I1_MD1 = int.Parse(vec6[1]);
55                 I2_MD1 = int.Parse(vec6[2]);
56                 I3_MD1 = int.Parse(vec6[3]);
57                 I4_MD1 = int.Parse(vec6[4]);
58                 I5_MD1 = int.Parse(vec6[5]);
59                 I6_MD1 = int.Parse(vec6[6]);
60
61                 Q0_MD1 = int.Parse(vec6[7]);
62                 Q1_MD1 = int.Parse(vec6[8]);
63                 Q2_MD1 = int.Parse(vec6[9]); // print(value); //printamos la línea leída para verificar que leemos el dato que manda nuestro Arduino
64                 Q3_MD1 = int.Parse(vec6[10]);
65
66                 I0_MD2 = int.Parse(vec6[11]); //de las comas y almacenamos los valores en un array.
67                 I1_MD2 = int.Parse(vec6[12]);
68                 I2_MD2 = int.Parse(vec6[13]);
69                 I3_MD2 = int.Parse(vec6[14]);
70
71

```

```

71
72
73     Q0_MD2 = int.Parse(vec6[15]);
74     Q1_MD2 = int.Parse(vec6[16]);
75     Q2_MD2 = int.Parse(vec6[17]);           // print(value); //printemos la línea leída para verificar que leemos el dato que manda nuestro Arduino
76     Q3_MD2 = int.Parse(vec6[18]);
77
78     //print(value);
79
80     if (cont == 1) /* selector modo de operacion */
81     {
82         MDPT = 1;
83     }
84     if (cont == 0)
85     {
86         MDPT = 0;
87     }
88
89     catch
90     {
91     }
92
93 }
94
95
96 public void brazoderecha()
97 {
98     if (stream.IsOpen)
99     {
100         stream.Write("A");
101         stream.Write("A");
102         stream.Write("A");
103         stream.Write("A");
104         aux = 1; //boton derecha click
105         aux1 = 0;
106     }
107 }
108

```

```

108
109 public void brazoizquierda()
110 {
111     if (stream.IsOpen)
112     {
113         stream.Write("B");
114         stream.Write("B");
115         stream.Write("B");
116         stream.Write("B");
117         aux1 = 1; //boton izquierda click
118         aux = 0;
119     }
120 }
121
122 public void selectManuAuto()
123 {
124     //print(cont);
125     if (stream.IsOpen)
126     {
127         cont++;
128         if (cont == 1)
129         {
130             stream.Write("C");
131             stream.Write("C");
132             stream.Write("C");
133             stream.Write("C");
134         }
135         if (cont == 2)
136         {
137             stream.Write("D");
138             stream.Write("D");
139             stream.Write("D");
140             stream.Write("D");
141             cont = 0;
142         }
143     }
144 }
145

```

Anexo 5

```

#include <SPI.h>
#include <Ethernet.h>
#include <Modbus.h>
#include <ModbusIP.h>

const int ENVIAR_DT1=2;
const int ENVIAR_DT2=3;
const int ENVIAR_DT3=4;
const int ENVIAR_DT4=5;
const int ENVIAR_DT5=6;
const int ENVIAR_DT6=7;
const int ENVIAR_DT7=8;
const int ENVIAR_DT8=9;

const int RECIB_DT1= 10;
const int RECIB_DT2= 11;
const int RECIB_DT3= 12;
const int RECIB_DT4= 13;
const int RECIB_DT5= 14;
const int RECIB_DT6= 15;
const int RECIB_DT7= 16;

int I0=0;
int I1=0;

uint32_t t1;

int input;

ModbusIP mb;

byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
IPAddress ip(192, 168, 100, 88);
IPAddress gateway(192, 168, 100, 1);
IPAddress subnet(255, 255, 255, 0);

void setup() {
    Serial.begin(9600);
    Ethernet.begin(mac, ip, gateway, subnet);
    mb.config(mac, ip);
}

```

```

void setup() {
  Serial.begin(9600);

  Ethernet.begin(mac, ip, gateway, subnet);
  mb.config(mac, ip);

  //ENVIO DE DATOS
  mb.addHreq(ENVIAR_DT1);
  delay(200);
  mb.addHreq(ENVIAR_DT2);
  delay(200);
  mb.addHreq(ENVIAR_DT3);
  delay(200);
  mb.addHreq(ENVIAR_DT4);
  delay(200);
  mb.addHreq(ENVIAR_DT5);
  delay(200);
  mb.addHreq(ENVIAR_DT6);
  delay(200);
  mb.addHreq(ENVIAR_DT7);
  delay(200);
  mb.addHreq(ENVIAR_DT8);
  delay(200);

  //RECEPCION DE DATOS
  mb.addHreq(RECIB_DT1);
  delay(200);
  mb.addHreq(RECIB_DT2);
  delay(200);
  mb.addHreq(RECIB_DT3);
  delay(200);
  mb.addHreq(RECIB_DT4);
  delay(200);
  mb.addHreq(RECIB_DT5);
  delay(200);
  mb.addHreq(RECIB_DT6);
  delay(200);
  mb.addHreq(RECIB_DT7);
  delay(200);

  t0=millis();
}

void loop() {

  if (Serial.available()>0){

    input=Serial.read();
    if (input=='A'){
      mb.Hreq(ENVIAR_DT1,1);
      delay(20);
      mb.Hreq(ENVIAR_DT2,0);
      delay(20);
      mb.Hreq(ENVIAR_DT1,1);
      delay(20);
      mb.Hreq(ENVIAR_DT2,0);
      delay(20);
      mb.Hreq(ENVIAR_DT1,1);
      delay(20);
      mb.Hreq(ENVIAR_DT2,0);
      delay(20);
    }

    if (input=='B'){
      mb.Hreq(ENVIAR_DT2,1);
      delay(20);
      mb.Hreq(ENVIAR_DT1,0);
      delay(20);
      mb.Hreq(ENVIAR_DT2,1);
      delay(20);
      mb.Hreq(ENVIAR_DT1,0);
      delay(20);
      mb.Hreq(ENVIAR_DT2,1);
      delay(20);
      mb.Hreq(ENVIAR_DT1,0);
      delay(20);
    }

    if (input=='C'){
      mb.Hreq(ENVIAR_DT3,1);
      delay(20);
      mb.Hreq(ENVIAR_DT3,1);
      delay(20);
      mb.Hreq(ENVIAR_DT3,1);
      delay(20);
      mb.Hreq(ENVIAR_DT3,1);
      delay(20);
      mb.Hreq(ENVIAR_DT3,1);
      delay(20);
      mb.Hreq(ENVIAR_DT3,1);
      delay(20);
    }
  }
}

```

```
if (input=='D'){
  mb.Hreg(ENVIAR_DT3,0);
  delay(20);
  mb.Hreg(ENVIAR_DT3,0);
  delay(20);
  mb.Hreg(ENVIAR_DT3,0);
  delay(20);
  mb.Hreg(ENVIAR_DT3,0);
  delay(20);
  mb.Hreg(ENVIAR_DT3,0);
  delay(20);
  mb.Hreg(ENVIAR_DT3,0);
  delay(20);
}
}

if(millis()-t0>=200){
  mb.task();

  Serial.flush();
  Serial.println(String(mb.Hreg(RECIB_DT1))+" "+String(mb.Hreg(RECIB_DT2))+" "+String(mb.Hreg(RECIB_DT3))+" "+String(mb.Hreg(RECIB_DT4))+" "+String(mb.Hreg(RECIB_DT5)));
  delay(30);
}
}
```